THE UNIVERSITY OF ALBERTA

ERROR CORRECTING CODES FOR ASYMMETRIC PATHS

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF MASTER OF SCIENCE

DEPARTMENT OF MATHEMATICS

by

MRS. C. MARGUERITE FENYVESI

CALGARY, ALBERTA

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "Error Correcting Codes for Asymmetric Path" submitted by Mrs. C. Marguerite Fenyvesi in partial fulfilment of the requirements for the degree of Master of Science.

J. E. L. Peck ....
Supervisor

....

Leo Moser ....

Nobuo Nobusawa ....

Jas. P. Hanton ....
External Examiner

Date ....

INTRODUCTION

This thesis deals with error correcting codes to be used on a binary asymmetric independent data tronsmission channel. Much work has been done on constructing and analyzing error correcting codes in the past few years. However, these works, as far as the author is aware, have been restricted to the assumption of a symmetric channel, where in the binary case no distinction is made between the transmission of a 0 or a 1. There are a very small number of papers on error detection codes for a completely asymmetric path or channel. However, for reflected signals; that is encoded radar signals with picture to picture memory; or over-horizon communication systems with encoded signals and encoded messages, where the signals are radiated upward and then reflected back from the trails of meteorites, the probability that a 0 becomes a 1 is much larger than the probability that a 1 becomes a 0. Therefore, it is necessary to consider error correction for binary asymmetric paths and this thesis discusses certain types of codes and the probability of correct decoding if they are transmitted on an asymmetric path.

Chapter I is an introduction to the basic concepts in coding. Chapter II is a survey of algebraic definitions and theorems, some without proof, which is required for the development of group codes. Chapter III discusses the previous work done on group codes for the symmetric path. Then the theory is extended by the author, for single-error correcting group codes, to the asymmetric path. Chapter IV develops the theory for certain non-group codes transmitted on an asymmetric path. In Chapter V, five example codes are introduced to illustrate the theory developed in the previous chapters. The probability of correct decoding of these codes is calculated, both for the symmetric and asymmetric path, and these calculations are shown in the Appendix.

ACKNOWLEDGEMENTS

The writer would like to thank Dr. J.E.L. Peck for the supervision of this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION TO CODING

1. Communication Channels

The accurate transmission of data at high speeds over communication channels is becoming extremely important with the reliability needed for use with computers and automatic equipment. It has been apparent from the earliest days of communication systems that the signals were not exactly the same at the input and output side of a black box or path. The electrical and electromagnetic disturbances which alter the transmitted signals have been called noise. Many existing communication systems are inherently noisy. Error is introduced by imperfect medium as magnetic tape or telephone lines, or by circuits using relays, diodes, or transistors which have a probability of error. Until quite recently the efforts to improve the accuracy of transmission were to reduce the noise by increasing the signal to noise ratio through an increase in the power of the signal, or by improving the circuit and its components. In recent years with the growing need for reliable communication, error correcting codes have been introduced.

A block diagram of a digital communication system is shown in Figure 1.1.

The source usually consists of binary, digital or alphabetic information. The encoder changes the

BLOCK DIAGRAM OF DIGITAL

COMMUNICATION SYSTEM

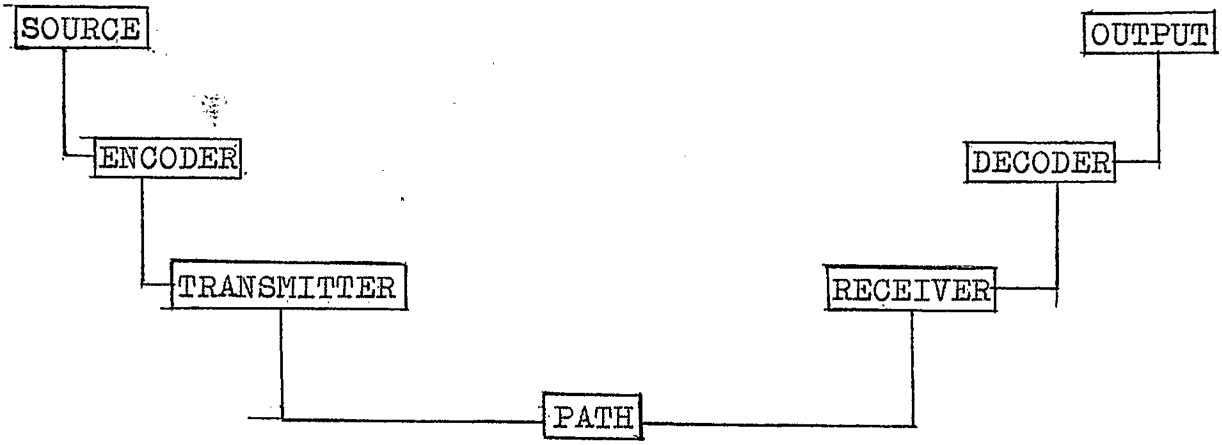


FIGURE 1.1

information into a valid code word of the system and in this paper the code words are assumed to be binary words. The encoding usually involves adding some redundancy to the information in the form of mathematical structure or check bits to be used later to detect or correct possible errors. The transmitter changes the code words into signals acceptable to the channel. With binary words there are exactly two messages which can be denoted by "0" and "1". The transmitter or modulator converts these messages into distinct waveforms, for example, a 1 would become a pulse and a 0 no pulse. At the receiver, the channel signals are put back in the form of binary words, and the decoder makes use of the added redundancy to make a decision as to what information actually was sent or to indicate that an error has occurred. The output will be binary information or it may be changed to digital or alphabetic form.

The previous work done in error correcting took into consideration the type of error which occurs in the operation of transistors, diodes and relays. These errors have a Gaussian shaped probability distribution and the probability, for example, of switching or not switching; that is, the probability that a 1 becomes a 0 or that a 0 becomes a 1, is equal. But it is necessary to consider black boxes where this statement is not true. For example, in returning radar signal pulses from a long path, and where these are partly buried in noise, the probability that a 0 becomes a 1 is much larger than the probability that a 1 becomes a 0. This is true for reflected signals, that is, encoded radar signals, with picture to picture memory, or over-horizon communication systems, with encoded signals and encoded messages, where the signals are radiated upward and then reflected back from the trails of meteorites or from passive communication satellites. In these cases, all the necessary factors to use and apply asymmetric error correcting systems exist. It can be shown that the reflected amplitude will be comparable to the integrated noise power in a given bandwidth.

It can also be shown that each pulse will be affected individually by the noise, that is the noise is non-coherent. There is a possibility of long noise bursts and there are burst error correcting codes especially designed to correct errors resulting from these noise bursts, providing the burst is not longer than one-third of the word time. However, long noise bursts will not be considered here because systems could easily be designed so that transmission automatically stops when a long noise burst occurs, and after the transmission of a probing sequence and the establishment of a new path, the system automatically repeats the last code word. Therefore, it will be assumed the noise affects each symbol independently.

One can also consider a black box with the characteristics that the probability that a 1 becomes a 0 is much larger than the probability that a 0 becomes a 1. An example of this is interplanetary narrow laser beam communication system (where narrow is applied to the frequency band) where there is sudden absorption from meteorites (path blocking).

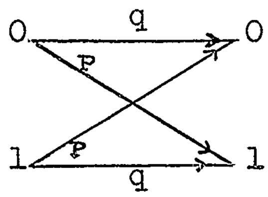
In this thesis the case considered is of weak reflected signals in noisy media and so the probability that a 0 becomes a 1 is larger than the probability that a 1 becomes a 0. At some future time, consideration will be given to the application of the theory developed here to reflected infra-red communication and guidance systems.

The codes discussed in this paper will be block codes.

Definition 1.1 [7, page 4] A block code is defined as a code that uses sequences of n channel symbols, or n-tuples. Only certain of these n-tuples will be transmitted and these are called code words. To predict the performance of a code it is necessary to have some knowledge of the path or channel. Three communication channels are shown in Figure 1.2, Figure 1.3 and Figure 1.4.

Actually the symmetric and completely asymmetric channels are special cases of the general asymmetric channel. If $q_1 = q_2$ and $p_1 = p_2$ then the asymmetric channel becomes what has been called the symmetric channel and if $q_1 = 1$ and $p_1 = 0$, then the asymmetric channel becomes the completely asymmetric channel.

THE BINARY SYMMETRIC PATH



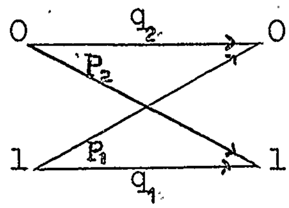
Pr (1 received | 1 sent)=q   Pr (0 received | 1 sent) =p

Pr (0 received | 0 sent)=q   Pr (1 received | 0 sent) =p

$$p+q=1$$

FIGURE 1.2

THE BINARY ASYMMETRIC PATH



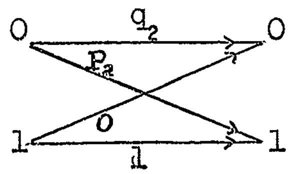
Pr (1 received | 1 sent) $=q_1$ Pr (0 received | 1 sent) $=p_1$

Pr (0 received | 0 sent)$=q_2$ Pr (1 received | 0 sent) $=p_2$

$$q_i+p_i = 1 \quad i=1,2$$

FIGURE 1.3

THE BINARY COMPLETELY ASYMMETRIC PATH



Pr (1 received | 1 sent) =1 Pr (0 received | 1 sent) =0

Pr (0 received | 0 sent) $=q_2$ Pr (1 received | 0 sent) $=p_2$

$$q_2+p_2 = 1$$

FIGURE 1.4

2. Error Probabilities in Transmission

A channel which is to be used can be tested in the following way. [5, page 6] A long string of 0's, say N symbols, would be transmitted. A 0 received would indicate no error or a success and a 1 received would be an error or a failure. If the statistical properties of the noise do not change with time, the probability of correct transmission for the channel is No/N where No is the number of 0's received. No/N is the estimate of the probability that a 0 will be received, given that a 0 was sent. This is a conditional probability and can be written:

$$\Pr(0 \text{ received}/0 \text{ sent}) = No/N = q_2$$

$$\Pr(1 \text{ received}/0 \text{ sent}) = (N-No)/N = 1-q_2 = p_2$$

Similarly, a long string of 1's, say N symbols, can be transmitted and the number of 1's received denoted by N_1. Then the conditional probability can be written:

$$\Pr(1 \text{ received}/1 \text{ sent}) = N_1/N = q_1$$

$$\Pr(0 \text{ received}/1 \text{ sent}) = (N-N_1)/N = 1-q_1 = p_1$$

If the channel is symmetric q_1 will be equal to q_2 or so close that no appreciable error will result from taking the probability of correct transmission to be $(q_1 + q_2)/2 = q$.

In this paper it will be assumed that

$$q > p \text{ and } q_i > p_i \quad i = 1,2$$

Since in the case considered here the probability that a 0 becomes a 1 is larger than or equal to the probability that a 1 becomes a 0 the following inequalities can be written:

$$q_1 \geq q_2 > p_2 \geq p_1$$

For any code of length n with w 1's the probability that no error will occur is $q_1^w q_2^{n-w}$. The probability that one error will occur in a specified position of the code word is $p_1 q_1^{w-1} q_2^{n-w}$ if the error is the type such that a 1 becomes a 0, and it is $p_2 q_1^w q_2^{n-w-1}$ if the error is such that a 0 becomes a 1. With a symmetric channel the probability of no error is q^n and the probability of one error in a specified place is pq^{n-1}. In general the probability that the received word differs from the transmitted word in j positions is $p^j q^{n-j}$. Since $q>p$, the probability that no error occurs in transmission is most likely, the probability that one error occurs is more likely than the probability that two errors occur and so on. Therefore, in the symmetric channel the best decision at the receiving end is to decode into the code word which differs from the received word in the fewest positions. This is called maximum likelihood decoding. In the asymmetric channel there are exceptions to this rule and these will be discussed in Chapters III and IV.

3. Error Detection and Correction

Useful concepts in discussing the error correcting ability of codes are the Hamming weight and Hamming distance. [4]

Definition 1.2 The Hamming weight of a code word v, denoted $w(v)$ is defined to be the number of non-zero components.

Definition 1.3 The Hamming distance between two words v_1 and v_2, denoted $d(v_1, v_2)$ is defined to be the number of positions in which they differ.

Thus, it can be seen from [4] and [7, page 7-8] that a single error results in a Hamming distance of one between the transmitted word and the received word. If a code designed for error detection had a minimum distance of $(d+1)$ between the code words, every possible pattern of d or fewer errors could be detected. Of course, if more than d errors occurred in the transmission of a code word, the errors could go undetected as it would be possible for one code word to be received as another code word, or to be at least as close to another code word as to the transmitted one. Similarly, it is possible to correct all patterns of t or fewer errors if and only if the minimum distance between code words is at least $(2t+1)$ since any received word with $t' \leq t$ errors differs from the transmitted word in t' places, but it differs from all other possible code words in at least $(2t+1) - t' > t'$ places and so using maximum likelihood decoding would be correctly decoded. However, if the minimum distance between code words was less than (2t+1), there would be at least one case where t errors would result in a received word being at least as close to another code word as to the transmitted one. Also, it is possible to decode such that all combinations of t or fewer errors are corrected, and simultaneously d or fewer additional errors are detected for $d \geq t$, if and only if the minimum distance between code words is (t+d+1). This can be seen as $d \geq t$, then $(t+d+1) \geq (2t + 1)$, so any combination of t or fewer errors can be corrected. Also, as the minimum distance between code words is (t+d+1), then (t+d) errors can be detected, or an additional d errors can simultaneously be detected. In the special case of the completely asymmetric channel where only one type of error occurs, these rules are slightly modified.

CHAPTER II

SURVEY OF ALGEBRA

This chapter deals with definitions and theorems of algebraic systems which are needed in the development of codes with algebraic structure.

Definition 2.1 [6, page 9] Let A be a given set. A binary operation "$\circ$" on A is a correspondence that associates with each ordered pair (a,b) of elements of A, a uniquely determined element $a_\circ b$ of A.

Definition 2.2 A non-empty set G on which there is defined a binary operation "$\circ$" is called a group (with respect to this operation), provided the following properties are satisfied:

(i) If $a,b,c \in G$, then $(a_\circ b)_\circ c = a_\circ(b_\circ c)$ (associative law).

(ii) There exists an element 1 of G such that $1_\circ a = a_\circ 1 = a$ for every element a of G (existance of an identity)

(iii) If $a \in G$, there exists an element x of G such that $a_\circ x = x_\circ a = 1$ (existance of inverses)

The inverse of a is frequently written a^{-1}.

Theorem 2.1 The identity element of a group is unique, and the inverse element of each group element is unique.

Proof The identity element is unique, for if there were two identity elements, 1 and $1'$ then $1_\circ 1' = 1'$ and $1_\circ 1' = 1$, so $1 = 1'$. Similarly, inverses are unique, for if a group element were to have two inverses g^{-1} and g_1^{-1}, then:

$$\begin{aligned} g^{-1} &= 1 \circ g^{-1} \\ &= (g_1^{-1} \circ g) \circ g^{-1} \\ &= g_1^{-1} \circ (g \circ g^{-1}) \\ &= g_1^{-1} \circ 1 \\ &= g_1^{-1} \end{aligned}$$

Definition 2.3 A group is said to be Abelian or commutative if it satisfies the following property:

(i) If $a, b \varepsilon G$, then $a \circ b = b \circ a$

Definition 2.4 A ring R is a non-empty set of elements on which there is defined two binary operations. One is called addition and denoted $a+b$, and the other is called multiplication and denoted ab. In order for R to be a ring the following axioms must be satisfied:

(i) The set R is an Abelian group under addition.

(ii) If $a, b \varepsilon R$, then ab is defined (and is an element of R)(closure)

(iii) If $a, b, c \varepsilon R$, then $a(bc) = (ab)c$ (associative law).

(iv) If $a, b, c, \varepsilon R$, then $a(b+c) = ab + ac$ and $(b+c)a = ba + ca$ (distributive law).

Definition 2.5 A ring is called commutative if its multiplication operation is commutative; that is, if $a, b \varepsilon R$ then $ab=ba$.

Definition 2.6 A field F, is defined as a commutative ring with a unit element (multiplicative identity) in which every non-zero element has a multiplicative inverse.

Definition 2.7 A subset of elements of a group G is called a subgroup H if it satisfies all the axioms of a group itself.

This paper will be concerned only with finite groups.

Definition 2.8 [7, page 17] A coset of a finite group can be constructed as follows:

Suppose that the elements of a group G are g_1, g_2, ... and the elements of a subgroup H are h_1, h_2, ... and an array is formed with the first row consisting of the elements of a subgroup with the identity element h_1, in the left hand position. The first element in the second row is any element not appearing in the first row and the rest of the elements are obtained by multiplying each subgroup element by this first element. Similarly, other rows are formed, each with a previously unused element in the first column, until all the group elements appear somewhere in the array, as shown in Figure 2.1

$$\begin{array}{llllll}
h_1 = 1 & h_2 & h_3 & \cdot & \cdot\ \cdot\ \cdot & h_n \\
g_1 h_1 = g_1 & g_1 h_2 & g_1 h_3 & & \cdot\ \cdot\ \cdot & g_1 h_n \\
g_2 h_1 = g_2 & g_2 h_2 & g_2 h_3 & & \cdot\ \cdot\ \cdot & g_2 h_n \\
\vdots & \vdots & \vdots & & & \vdots \\
g_m h_1 = g_m & g_m h_2 & g_m h_3 & & \cdot\ \cdot\ \cdot & g_m h_n
\end{array}$$

Figure 2.1

The set of elements in a row of this array is called a left coset and the element appearing in the first column is called a coset leader. Right cosets could be similarly formed.

The property that the inverse of a product is the product of the inverses in reverse order can be seen from the following:

$$(ab)(b^{-1}a^{-1})$$
$$= a(bb^{-1})a^{-1}$$
$$= a1a^{-1}$$
$$= aa^{-1}$$
$$= 1$$

Therefore $(ab)^{-1} = b^{-1}a^{-1}$

<u>Theorem 2.2</u> [7, Theorem 2.3] Two elements g and g' of a group G are in the same left coset of a subgroup H of G if and only if $g^{-1}g'$ is an element of H.

<u>Proof</u> (i) If g and g' belong to the coset whose leader is g_i, then they can be written in the form:

$$g = g_i h_j \text{ for some } j$$
$$g' = g_i h_k \text{ for some } k$$

and

$$g^{-1}g' = (g_i h_j)^{-1}(g_i h_k)$$
$$= h_j^{-1} g_i^{-1} \quad g_i h_k$$
$$= h_j^{-1} h_k$$

$h_j^{-1} h_k$ is in H by the properties of groups and subgroups.

(ii) If $g = g_i h$ where g_i is the coset leader and if $g^{-1}g' = h'$, then $g' = gh' = g_i hh'$ which is in the same coset, since hh' is in the subgroup.

<u>Theorem 2.3</u> [7, Theorem 2.4] Every element of a group G is in one and only one coset of the subgroup H.

<u>Proof</u> Every element appears at least once by the construction of the array.

It must be shown that each element appears only once in the array.

(i) First suppose that two elements in the same row are equal, that is:

$$g_i h_j = g_i h_k$$

Multiplying each on the left by g_i^{-1} gives

$$h_j = h_k$$

This is a contradiction since each subgroup element was assumed to appear only once in the first row.

(ii) Now suppose the two equal elements appear in different rows, for example:

$g_i h_j = g_k h_l$ and suppose $i > k$

Multiplying on the right by h_j^{-1} gives

$$g_i = g_k h_l h_j^{-1}$$

Since $h_l h_j^{-1}$ is in the subgroup this says that g_i is in kth coset. However, this contradicts the rule of construction that coset leaders should be previously unused.

<u>Definition 2.9</u> A set V of elements is called a <u>vector space</u> over a field F if it satisfies the following axioms:

(i) The set V is an Abelian group

(ii) If $a \varepsilon F$, $u \varepsilon V$, then au is a uniquely determined element of V

(iii) $a(u+v) = au + av$

$a \varepsilon F, u, v \varepsilon V$ (distributive law)

(iv) $(a+b)u = au + bu$, $a, b, \varepsilon F$, $u \varepsilon V$

(distributive law)

(v) $a(bu) = (ab)u$, $a, b, \varepsilon F$, $u \varepsilon V$

(associative law)

(vi) $1u = u$ 1 is unity of F, $u \varepsilon V$

A subset of a vector space is called a subspace if it satisfies the axioms for a vector space. To check whether a subset of a vector space is a subspace, it is necessary only to check for closure under addition and scalar multiplication.

Definition 2.10 [7, page 20] An n-tuple over a field is an ordered set of n field elements, and is denoted $(a_1, a_2, \ldots, a_n)$, where each a_i is an element of the field. Addition of n-tuples is defined as follows:

$$(a_1, a_2, \ldots, a_n) + (b_1, b_2, \ldots, b_n)$$
$$= (a_1 + b_1, a_2 + b_2, \ldots, a_n + b_n)$$

Multiplication of an n-tuple by a field element is defined as follows:

$$c(a_1, a_2, \ldots, a_n) = (ca_1, ca_2, \ldots, ca_n)$$

With these two definitions it can be shown that the set of all n-tuples over a field form a vector space.

Multiplication of n-tuples can also be defined as follows:

$$(a_1, a_2, \ldots, a_n)(b_1, b_2, \ldots, b_n)$$
$$= (a_1 b_1, a_2 b_2, \ldots, a_n b_n)$$

In the set of all n-tuples $0 = (0, \ldots, 0)$ and the context makes clear whether the symbol 0 means a vector or a scalar.

Definition 2.11 A linear combination of a vector is defined as a sum of the form

$$u = a_1 v_1 + a_2 v_2 + \ldots + a_n v_n$$
$$a_i \varepsilon F, \quad v_i \varepsilon V$$

Theorem 2.4 [7, theorem 2.5] The set of all linear combinations of a set of vectors $v_1, \ldots, v_n$ of a vector space V is a subspace of V.

Proof Every linear combination of vectors of V is also a vector of V so the set of all linear combinations of vectors is a subset of V. Let the set of all linear combinations of $v_1, v_2, \ldots, v_n$ be called S. If w =

$= b_1 v_1 + \ldots + b_n v_n$ and $u = c_1 v_1 + \ldots + c_2 v_n$ are any two elements of S then, $w + u = (b_1 + c_1)v_1 + \ldots + (b_n + c_n)v_n$ is in S and the subset is closed under addition. Also, any scalar multiple of w, $aw = ab_1 v_1 + \ldots + ab_n v_n$, is in S, so S is closed under multiplication by scalars. Therefore S is a subspace of V.

<u>Definition 2.12.</u> A set of vectors $v_1, v_2, \ldots v_n$, is <u>linearly dependent</u> if and only if there are scalars $c_1, \ldots, c_n$ not all zero such that

$$c_1 v_1 + c_2 v_2 + \ldots + c_n v_n = 0.$$

A set of vectors is <u>linearly independent</u> if it is not linearly dependent.

<u>Definition 2.13</u> A set of vectors is said to <u>span</u> a vector if every vector in the vector space equals a linear combination of the vectors of the set.

<u>Theorem 2.5</u> [7, Theorem 2.6] If a set of n vectors $v_1, \ldots, v_n$ spans a vector space that contain a set of m linearly independent vectors $u_1, \ldots, u_m$ then $n \geq m$.

<u>Proof</u> Since $v_1, \ldots, v_n$ span the space, u_1 can be expressed as a linear combination of the v_i. This equation can be solved for one of the v_i, say v_k, in terms of u_1 and the rest of the v_i. Therefore the set consisting of u_1 and the rest of the v_i spans the vector space, since any linear combination of the v_i becomes a linear combination of u_1 and all the v_i, $i \neq k$, as the expression for v_k in terms of u_1, and the other v_i is used to eliminate v_k. Then u_2 can be expressed as a linear combination of u_1 and all the v_i, $i \neq k$. Since the u_i are linearly independent

some v_i must have a non-zero coefficient, and therefore this v_i can be expressed in terms of u_1, u_2 and the remaining $(n-2)$ v_i. These n vectors span the space. This process can be continued until all m of the u_i vectors are used. Since at each stage one v_i vector is replaced, the number of vectors v_i must be at least as great as the number of vectors u_i, that is $n \geq m$.

Theorem 2.6 [7, Theorem 2.7] If two sets of linearly independent vectors span the same space, there are the same number of vectors in each set.

Proof If there are m vectors in one set and n in the other, then by Theorem 2.5 $m \geq n$ and $n \geq m$, and thus $n = m$.

Definition 2.14 The dimension of a space is defined as the number of linearly independent vectors that span the space.

Definition 2.15 A basis of a space is defined as a set of n linearly independent vectors spanning an n-dimensional vector space.

Definition 2.16 An inner-product or dot-product of two n-tuples is a scalar and is defined as follows:

$$(a_1, \ldots, a_n) \cdot (b_1, \ldots, b_n) = a_1 b_1 + \ldots + a_n b_n$$

It can be shown that $u \cdot v = v \cdot u$ and $w \cdot (u+v) = w \cdot u + w \cdot v$. If the inner product of two vectors is zero, they are said to be orthogonal.

Definition 2.17 [7, page 22] An nxm matrix, M, is an ordered set of nm elements in a rectangular array of n rows and m columns.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} = [a_{ij}]$$

The elements of the matrix will be elements of a field. The rows or columns of a matrix M can be thought of as vectors. The row (column) space of a matrix is the set of all linear combinations of the row (column) vectors. The dimension of the row (column) space is called the row (column) rank.

Definition 2.18 [2, page 271] There is a set of elementary row operations defined for matrices as follows:

(i) The interchange of any two rows.

(ii) The multiplication of a row by a scalar $C \neq 0$, in F.

(iii) The addition of one row to another row.

Definition 2.19 [7, page 23] Elementary row operations can be used to rearrange a matrix and put it in a standard form, echelon canonical form, which is defined as follows:

(i) Every leading term, that is first non-zero term, of a non-zero row is one.

(ii) Every column containing such a leading term has all its other entries zero.

(iii) The leading term of any row is to the right of leading terms in every preceding row. All zero rows are below all non-zero rows. The non-zero rows of a matrix in echelon canonical form are linearly independent, and thus the number of non-zero rows is the dimension of the row space.

<u>Definition 2.20</u> [7, page 25] The <u>transpose</u> of an nxm matrix M is an mxn matrix denoted M^T, whose rows are the columns of M, and whose columns are the rows of M. The transpose of $[a_{ij}]$ is $[a_{ji}]$.

Two nxm matrices can be added, element by element. This addition can be written as:

$$[a_{ij}] + [b_{ij}] = [a_{ij} + b_{ij}]$$

An nxk matrix $[a_{ij}]$ and a kxm matrix $[b_{ij}]$ can be multiplied to give an nxm matrix $[c_{ij}]$ by the rule.

$$c_{ij} = \sum_{l=1}^{k} a_{il} b_{lj}$$

<u>Theorem 2.7</u> [7, Theorem 2.13] The set of all n-tuples orthogonal to a subspace V_1 of n-tuples forms a subspace V_2 of n-tuples. This subspace V_2 is called the <u>null space</u> of V_1.

<u>Proof</u> Let V_1 be a subspace of the vector space of all n-tuples over a field. Let V_2 be the set of all vectors orthogonal to every vector in V_1. Let v be any vector in V_1 and u_1 and u_2 any vectors in V_2.

Then $v \cdot u_1 = v \cdot u_2 = 0$

and $v \cdot u_1 + v \cdot u_2 = 0 = v(u_1 + u_2)$

Therefore $(u_1 + u_2)$ is in V_2.

Also $v \cdot (cu_1) = c\,(v \cdot u_1) = 0$

Therefore cu_1 is in V_2.

Thus, V_2 is a subspace.

<u>Theorem 2.8</u> [7, Theorem 2.14] If a vector is orthogonal to every vector of a set which spans V_1, it is in the null space of V_1.

<u>Proof</u> If $v_1, \ldots, v_n$ span V_1, then every vector of V_1 can be expressed in the form:

$$v = c_1 v_1 + \ldots + c_n v_n$$

Then $v \cdot u = (c_1 v_1 + \ldots + c_n v_n) \cdot u$

$$= c_1 v_1 \cdot u + \ldots + c_n v_n \cdot u$$

and if it is orthogonal to each v_i, it is orthogonal to v. Therefore u is in the null space of V_1.

The null space of the row space of a matrix is called the null space of the matrix. A vector is in the null space of a matrix, if it is orthogonal to each row of the matrix. If the n-tuple v is considered to be a $1 \times n$ matrix, v is in the null space of an $m \times n$ matrix M, if and only if, $vM^T = 0$.

It can be shown that if the dimension of a subspace of n-tuples is k, the dimension of the null space is n-k. If V_2 is a subspace of n-tuples, and V_1 is the null space of V_2, then V_2 is the null space of V_1.

CHAPTER III

GROUP CODES

1. Definition and Matrix Representation

In this chapter one type of error correcting code will be introduced.

Definition 3.1 [7, page 30] A set of n-tuples or vectors is called a linear code if, and only if, it is a subspace of the space of all n-tuples. The term group code is the common terminology for binary linear codes.

Since the Hamming distance between two vectors v_1 and v_2 is the number of positions in which they differ, the distance between v_1 and v_2 is equal to $w(v_1 - v_2)$. Since the set of all code vectors is a vector space, if v_1 and v_2 are code vectors of a linear code then $v_1 - v_2$ is also a code vector. Therefore the distance between any two code vectors must be equal to the weight of some third code vector, and the minimum weight of the nonzero code vectors will be the minimum distance for the linear code.

As the paper is concerned only with binary codes, the elements in the vector belong to the field of two elements, denoted by 0 and 1.

The following set of vectors of length $n = 5$ form a vector space V_1, and hence a binary group code.

(00000)	(00101)
(10011)	(10110)
(01010)	(01111)
(11001)	(11100)

The minimum weight, and hence the minimum distance is two. This code will be used as an example throughout most of this chapter, and is the example used by Peterson. [7, page 30]

Linear or group codes can be describbd by matrices. A matrix G, called a generator matrix of V, can be formed by using any set of basis vectors of the linear code V, as rows of the matrix. A vector is a code word if and only if it is a linear combination of the rows of G. G will have k rows, where k is the dimension of the vector space V. Since the rows must be linearly independent, k equals the rank of G. Each distinct linear combination of the rows of G gives a distinct code vector, since if any two linear combinations were equal, there would be a dependence relation among rows of G. Since there are k coefficients and in the binary case, two possible values for each, there are 2^k code vectors in V. Such a code is called an (n,k) code. The advantage of the matrix description is that it is much more compact than a list of code vectors. A generator matrix for the code V_1, of the previous example, is the matrix: [7, page 31]

$$\begin{bmatrix} 10011 \\ 01010 \\ 00101 \end{bmatrix}$$

There is an alternate description of codes using matrices. Again using the notation of Peterson [7, page 31], if V is a subspace of dimension k, its null space is a vector space V' of dimension (n-k). A matrix, H, can be formed whose rows are a basis for V'. It will have rank (n-k) and its row space will be V'. Then V is the null space of V', and a vector v is in V if and only if it is orthogonal to every row of H. That is, if and only if

$$vH^T = 0 \qquad (3.1)$$

If $v = (a_1, a_2, \ldots, a_n)$ and h_{ij} is the element in the i^{th} row and j^{th} column of H, then Equation (3.1) can be written:

$$\sum_j a_j h_{ij} = 0 \text{ for each } i, i=1 \ldots, (n-k) \quad (3.2)$$

Therefore Equation (3.1) means that the components of v must satisfy a set of (n-k) independent equations. Also since v is orthogonal to every vector in V', any linear combination of Equation (3.2) gives an equation that the components of v must satisfy. These equations are called parity checks and H is called a parity-check matrix of V.

In the example, [7, page 31-32] the null space V_2 of the vector space V_1 consists of the four vectors:

(00000) (10101)

(11010) (01111)

V_2 is the row space of the matrix:

$$\begin{bmatrix} 11010 \\ 10101 \end{bmatrix}$$

The code V_1 is the null space of this matrix, and to each vector of V_2 there is an equation that the components of every code vector must satisfy. For example, corresponding to the vector (01111) of V_2, is the equation

$$0a_1 + 1a_2 + 1a_3 + 1a_4 + 1a_5 = 0$$

which must be satisfied by every code point (a_1, a_2, a_3, a_4, a_5). For binary codes this is equivalent to having an even parity check on the last four components.

V and V′ are called dual codes [7, page 32] and if V is an (n,k) code, V′ is an (n,n-k) code. If a code is the row space of a matrix, its dual is the null space.

<u>Theorem 3.1</u> [7, Theorem 3.1] Let V be a linear code which is the null space of a matrix H. Then for each code word of weight w, ($w \neq 0$) there is a linear dependence relation among w columns of H, and conversely, for each linear dependence relation involving w ($w \neq 0$) columns of H, there is a code word of weight w.

<u>Proof</u>: A vector $v = (a_1, a_2, \ldots, a_n)$ is a code word if and only if

$$vH^T = 0$$

or if h_i is the i^{th} column vector of H

$$\sum_{i=1}^{n} a_i h_i = 0$$

This is exactly a linear dependence relation among columns of H, and the number of columns of H which appear with non-zero coefficients is the number of non-zero components of v, which is w.

Similarly, the coefficients of any dependence relation among w columns of H are components of a vector that must be in the null space of H and so there is a code word of weight w.

For a channel with independent errors, two codes which differ only in the arrangement of symbols have the same probability or error and are called equivalent. By row operations on a generator matrix G, a combinatorially equivalent matrix, G', in echelon canonical form can be obtained. G and G' will generate the same code. Then, the k columns that contain the leading 1's of each row can be arranged by column permutation to form a k x k identity matrix, resulting in a combinatorially equivalent matrix G'' for an equivalent code. It has the form shown in (3.3) and can be called reduced-echelon form [7, pge. 33] There is a reduced-echelon matrix G'' combinatorially equivalent to every generator matrix G and every code is equivalent to the row space of some matrix in reduced-echelon form.

$$G''' = \begin{bmatrix} 1 & 0 & \cdots & 0 & p_{11} & \cdots & p_{1\,n-k} \\ 0 & 1 & \cdots & 0 & p_{21} & \cdots & p_{2\,n-k} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 & p_{k,1} & \cdots & p_{k,n-k} \end{bmatrix} \tag{3.3}$$

Let $v = (a_1, a_2, \ldots, a_k)$ be an arbitrary k-tuple, and consider the vector u, which is a linear combination of rows of G'' with a_i as the ith coefficient. [7, page 34]

$$u = vG''' = (a_1, a_2, \ldots, a_k, c_1, c_2, \ldots c_{n-k}) \tag{3.4}$$

where $$c_j = \sum_{i=1}^{k} a_i p_{ij} \tag{3.5}$$

Thus the first k components of the code vector, called information symbols, can be chosen arbitrarily, and each of the last n-k components, called check or redundancy symbols is a linear combination of the first k components. A code of this type is called a systematic code. [4]

Theorem 3.2 [7, Theorem 3.4] If V is the row space of the matrix $G = [I_k P]$ where I_k is a $k \times k$ identity matrix and P is a $k \times (n-k)$ matrix, then V is the null space of $H = [-P^T I_{n-k}]$ where I_{n-k} is an $(n-k) \times (n-k)$ identity matrix.

Proof It can easily be verified that $GH^T = 0$, and as their ranks are k and (n-k) respectively, they are dual codes, and the row space of G is the null space of H.

If $u = (a_1, a_2, \ldots, a_k, c_1, c_2, \ldots c_{n-k})$ is a code vector, then

$$uH^T = 0 = -\sum a_i p_{ij} + c_j$$

which is the same as Equation (3.5) [7, page 34] .

For the code used in previous examples the generator matrix is in reduced-echelon form and is written [7, page 34-35]

$$G = \begin{bmatrix} 10011 \\ 01010 \\ 00101 \end{bmatrix} = [I_3 P]$$

If

$$H = \begin{bmatrix} 11010 \\ 10101 \end{bmatrix} = [-P^T I_2]$$

then $GH^T = HG^T = 0$ and the row space of each is the null space of the other. In each code word $(a_1, a_2, \ldots, a_5)$ the first three components can be chosen arbitrarily and

and the other two are parity-check symbols with

$$a_4 = a_1 + a_2$$
$$a_5 = a_1 + a_3 \qquad (3.6)$$

Since every code word is orthogonal to each row of H, from the first row

$$1a_1 + 1a_2 + 0a_3 + 1a_4 + 0a_5 = 0$$

and from the second row

$$1a_1 + 0a_2 + 1a_3 + 0a_4 + 1a_5 = 0$$

and these equations can be solved for a_4 and a_5 to give equations (3.6)

2. Decoding for the Symmetric Path

Let V be an (n,k) linear code, h_1 be the identity element and $h_2, h_3, \ldots, h_{2^k}$ be the other code vectors. [7, page 35] A decoding table called a standard array, can be formed using the method in Figure 2.1. The elements $g_1, g_2, \ldots$ were chosen to be any previously unused element. However, for this decoding table they will be chosen to be the elements most likely to be received if the identity element is transmitted. Thus the rows are cosets and the vectors in the first column are coset leaders. If the vector u is transmitted and a vector v is received, then v-u is called the error pattern.

Theorem 3.3 [7, Theorem 3.5] If the standard array is used as a decoding table, then a received vector v will be decoded correctly into the transmitted vector u, if and only if the error pattern v-u is a coset leader.

Proof: If $v-u = g_i$ the coset leader of the i^{th} coset, then $v = g_i + u$, and v must appear in the standard array in the i^{th} coset, under the code vector u and will be decoded correctly.

If, on the other hand, $v-u$ is not a coset leader, v must still be in some coset, say the j^{th}, with coset leader g_j. Then v is in the j^{th} row, but not under u for $v \neq g_j + u$.

Definition 3.2 [7, page 36] Let the linear code be the null space of an $r \times n$ matrix H, whose rows may, but need not, be linearly independent. For any received vector v, the r component vector

$$S = vH^T$$

is called the syndrome.

Since the code is the null space of H, a vector is a code word if, and only if, its syndrome is zero.

Theorem 3.4 Two vectors v_1 and v_2 are in the same coset if and only if their syndromes are equal.

Proof Two group elements v_1 and v_2 are in the same coset if and only if $(-v_2) + v_1 = v_1 - v_2$ is an element of the subgroup, which in this case is the code vector space. If the code space is the null space of H, then $(v_1 - v_2)$ is in the code space if and only if

$$(v_1 - v_2)H^T = 0$$

Since the distributive law holds for multiplication of matrices

$$(v_1 - v_2)H^T = v_1 H^T - v_2 H^T = 0$$

so $(v_1 - v_2)$ is a code vector if and only if the syndromes of v_1 and v_2 are equal.

Now to decode, a table is formed which shows the coset leader and the syndrome for each of the 2^{n-k} cosets. For each received vector the syndrome is calculated and the coset leader is looked up in the table.

The coset leader is the presumed error pattern, and subtracting it from the received vector gives the code vector that is assumed to have been sent. This decoding is the same as that using the standard array, but it requires less memory space and so is useful especially when n is large.

<u>Theorem 3.5</u> [7, Theorem 3.7] Let V be an (n,k) linear binary code to be used with the binary symmetric channel, and assume that all the code vectors are equally likely to be transmitted. Then the average probability of correct decoding is as large as possible for this code if the standard array, with each coset leader chosen to have minimum weight in its coset, is used as a decoding table.

<u>Proof</u> Let v_{ij} be the vector in the i^th^ row and j^th^ column of the decoding table. Denote by v_{oj} the code words placed at the top of the column. Denote by d_{ij} the Hamming distance between a received v_{ij} and the code word into which it is decoded, v_{oj}. Then the probability of correct decoding if the code word v_{oj} is transmitted is:

$$\sum_{i=0}^{2^{n-k}-1} p^{d_{ij}} q^{n-d_{ij}}$$

where p is the channel probability of error and q=1-p as shown in Figure 1.2.

Since there are 2^k code words which are assumed equally probable, in averaging the probability of correct decoding, the weighting factor 2^{-k} is used:

$$P_r \text{ (correct decoding)} = 2^{-k} \sum_{i,j} p^{dij} q^{n-dij}$$

There is one term in the sum for each possible received vector of binary symbols and that term is maximized in each case if that particular vector is decoded into the closest code vector in the Hamming sense, since $p^{dij}q^{n-dij}$ is a monotone decreasing function of d_{ij}. Therefore the probability of correct decoding will be maximized if each vector is decoded into the closest code vector.

Suppose that a particular vector v appears in the decoding table under the code vector u, which is at a Hamming distance w. Suppose that the closest code vector u_1 is at distance w_1. Let g denote the coset leader of the coset that contains v. Then g = v-u has weight w. The element $v-u_1 = g + (u - u_1)$ has weight w_1 and is in the same coset. However, it was assumed that g has minimum weight in its coset so $w_1 \geq w$ and therefore v is at least as close to u as to u_1.

By assuming a binary symmetric channel Theorem 3.5 can be applied to the previous example. The standard array for the code V_1 is [7, page 38].

00000	10011	01010	11001	00101	10110	01111	11100
00001	10010	01011	11000	00100	10111	01110	11101
00010	10001	01000	11011	00111	10100	01101	11110
10000	00011	11010	01001	10101	00110	11111	01100

A word will be correctly decoded if either the received word is exactly the transmitted code word or it lies in the column of the transmitted code word in the standard array. Using Theorem 3.5 the probability of correct decoding is

$$Pr = 2^{-3} \sum_{i,j} p^{d_{ij}} q^{5-d_{ij}}$$

where $d_{ij} = 0$, $i = 0$

$d_{ij} = 1$, $i = 1, \ldots, 2^{n-k}-1 = 3$

$$Pr = 2^{-3} \sum_{j=1}^{8} [p^0 q^5 + 3p^1 q^4]$$

$$Pr = 2^{-3}\ 8\ [\ q^5 + 3pq^4\ ]$$

$$Pr = q^5 + 3pq^4$$

For the code words both the parity checks of Equation 3.6 are satisfied. Using the formula in Definition 3.2 the syndromes for the next three cosets are 01, 10 and 11 respectively.

This code corrects only three of the five possible single error patterns. For example, vectors (00001) and (00100) are in the same coset in the decoding table with coset leader (00001). Since the error pattern is assumed to be the coset leader, if (00100) is received it will be decoded into code word (00101) even though the received vector could also have been obtained by a single error in code word (00000). A similar result occurs whenever there is a code word of weight two, as three is the minimum weight which is necessary and sufficient for correcting all single errors.

3. Decoding for the Asymmetric Path

Theorem 3.5 has been proved for a group code with a binary symmetric path. This can be extended, and under certain conditions, a similar theorem proved, for a binary asymmetric path. The standard array will be used as the decoding table and Theorem 3.3 will be assumed. Consider that the code word v_{o_j} with Hamming weight w_{o_j}, is transmitted. The probability that this code word is received is the probability that the w_{o_j} 1's remain 1's, which is q_1, times the probability that the $(n-w_{o_j})$ 0's remain 0's which is q_2. Therefore the probability, that the transmitted code word v_{o_j} is received, is $q_1^{w_{o_j}} q_2^{n-w_{o_j}}$. If an error occurs during transmission on the asymmetric channel it may be one of two kinds: a 1 may become a 0 or a 0 may become a 1. If the error is the $(0 \to 1)$ type, the probability of correct decoding is the product of the probability that the 1's remain 1's, that one 0 becomes a 1, and that the balance of the 0's remain 0's. This probability is written as follows:

$$q_1^{w_{o_j}} p_2 q_2^{n-w_{o_j}-1}$$

Similarly if the error is the $(1 \to 0)$ type the probability that the received word will be decoded correctly is written.

$$p_1 q_1^{w_{o_j}-1} q_2^{n-w_{o_j}}$$

Of these two, $q_1^{w_{o_j}} p_2 q_2^{n-w_{o_j}-1} > p_1 q_1^{w_{o_j}-1} q_2^{n-w_{o_j}}$.

The probability of correct decoding for the code of the previous example, transmitted on a binary asymmetric channel, can be written down by considering each entry in the standard array separately as follows:

column 1: $w^{o}_{01} = 0$ $\quad Pr1 = q_2^5 + p_2 q_2^4 + p_2 q_2^4 + p_2 q_2^4$

column 2: $w_{02} = 3$ $\quad Pr2 = q_1^3 q_2^2 + p_1 q_1^2 q_2^2 + p_1 q_1^2 q_2^2 + p_1 q_1^2 q_2^2$

column 3: $w_{03} = 2$ $\quad Pr3 = q_1^2 q_2^3 + q_1^2 p_2 q_2^2 + p_1 q_1^2 q_2^2 + p_1 q_1^2 q_2^2$

column 4: $w_{04} = 3$ $\quad Pr4 = q_1^3 q_2^2 + p_1 q_1^2 q_2^2 + q_1^3 p_2 q_2 + p_1 q_1^2 q_2^2$

column 5: $w_{05} = 2$ $\quad Pr5 = q_1^2 q_2^3 + p_1 q_1 q_2^3 + q_1^2 p_2 q_2^2 + q_1^2 p_2 q_2^2$

column 6: $w_{06} = 3$ $\quad Pr6 = q_1^3 q_2^2 + q_1^3 p_2 q_2 + p_1 q_1^2 q_2^2 + p_1 q_1^2 q_2^2$

column 7: $w_{07} = 4$ $\quad Pr7 = q_1^4 q_2 + p_1 q_1^3 q_2 + p_1 q_1^3 q_2 + q_1^4 p_2$

column 8: $w_{08} = 3$ $\quad Pr8 = q_1^3 q_2^2 + q_1^3 p_2 q_2 + q_1^3 p_2 q_2 + p_1 q_1^2 q_2^2$

By combining these terms the average probability of correct decoding can be written:

$$\begin{aligned} Pr = 2^{-3} \; [& (q_1^4 q_2 + 4q_1^3 q_2 + 2q_1^2 q_2^3 + q_2^5) \\ & + p_1 (2q_1^3 q_2 + 8q_1^2 q_2^2 + 2q_1 q_2^3) \\ & + p_2 (q_1^4 + 4q_1^3 q_2 + 4q_1^2 q_2^2 + 3q_2^4)] \end{aligned}$$

Although there are twelve cases where the type of error is (1→0) and twelve cases where the type of error is (0→1), it is not possible to simplify the above into a general formula. The decoding table can be used to correct three of the five possible single errors; those in the first, fourth and fifth digits. The weight of the code words are known but this does not tell which of the five digits are 1's and which are 0's. For example, the code words in column 2 and column 8 both have weight 3 but the code word in column 2 has 1's in positions one, four and five, and so all the correctable errors are of the type $(1 \to 0)$, while the code word in column 8 has a 1 in position one and 0's in positions four and five so one of the correctable errors is of the type $(1 \to 0)$, and the others are of the type $(0 \to 1)$. However, since it is assumed that noise affects each symbol independently, errors do not occur more frequently in the positions one, four and five than they do in any other position and therefore it does not seem reasonable to be able to correct single errors in some digits and not in others. If a single error correcting code is desired then it should be possible to correct every possible single error. Also it will be seen that for an array which corrects every single error it is possible to write a general formula for the probability of correct decoding. The weight w_{o_j} of a code word is known and so there would be w_{o_j} words with an error of the type $(1 \to 0)$ which could be corrected and $(n-w_{o_j})$ words with an error of the type $(0 \to 1)$ which could be corrected.

This type of error correcting code was introduced by Hamming and is called by his name. The binary Hamming code can be described in terms of its parity-check matrix. Again, using the notation of Peterson [7, page 64-65] a matrix H of 1's and 0's with m rows and 2^m-1 columns can be considered. The column vectors would consist of all possible m-tuples except the 0 m-tuple. As the field considered has two elements, if two vectors add to 0, they must be equal. Therefore, in this matrix no two columns or linear combinations of two columns will add to zero. The code vectors are in the null space of this matrix. They have a minimum weight of 3 and so the code is capable of correcting all single errors. The code vectors have length 2^m-1, with m parity-check symbols and therefore 2^m-1-m information symbols.

<u>Definition 3.3</u> [7, page 48] A linear code that has for some m all patterns of weight m or less and no others as coset leaders, is called a <u>perfect code</u>.

<u>Definition 3.4</u> [7, page 48] A code which for some m has all patterns of weight m or less and some of weight m+1, and none of greater weight as coset leaders, is called <u>quasi-perfect</u>.

The 2^m cosets are made up of the code space and the 2^m-1 single error patterns. This is an example of a perfect code as the coset leaders are the 0 vector, all single error patterns.

If an error occurs in the transmission of a code word u, the received vector is u+e, where e is a vector with a 1 in the error position and 0's in all the other components. The syndrome is

$$(u+e)H^T = uH^T + eH^T = eH^T$$

since the code vector u is in the null space of H. Since e is a vector with a single 1 in the error position, the syndrome eH^T is just the row of H^T corresponding to the error and so the error can be found by comparing the syndrome with the matrix H^T. Hamming did this by letting the i^{th} column of H be the binary representation of the number i and so the syndrome gives the binary representation of the position in error. [4]

From here on this paper will be concerned with codes which are capable of correcting all possible single errors. By the previous notation the length of the code word is n, and there are 2^k code words or columns and 2^{n-k} cosets or rows in the standard array. If a code is to correct all single errors using the standard array as a decoding table then the following relationship must be satisfied:

$$n+1 = 2^{n-k}$$

Consideration can now be given to a theorem similar to Theorem 3.5 for a single error correcting code on the binary asymmetric path.

In attempting to prove such a theorem it must be shown whether or not the standard array will result in maximum likelihood decoding. This was shown for the symmetric channel in Theorem 3.5. Since $p^{d_{ij}}q^{n-d_{ij}}$ is a monotone decreasing function of d_{ij}, the probability of correct decoding is maximized if each received vector is decoded into the closest code vector in the Hamming sense. For a single error correcting code using an asymmetric path the probability of no errors in the transmission of a code word is $q_1^{w_{oj}}q_2^{n-w_{oj}}$ where the variable w_{oj} is the Hamming weight of the code word. If a received word v lies in the table under the code word u, the probability that v is the result of a single error in u is of the form

$p_1 q_1^{w_oj-1} q_2^{n-w_oj}$ or $p_2 q_1^{w_oj} q_2^{n-w_oj-1}$. The word v must differ from all other code words in two or more places. Therefore, the probability that v is the result of two or more errors in some other code word must be a term such that the sum of the powers of p_1 and p_2 is greater than or equal to two. Since $q_i > p_i$ $i = 1,2$ the probability in general, that v is received when u is transmitted is greater than the probability that v is received when some other code word is transmitted. However, there are a few exceptions for certain values of q_1, q_2, p_1 and p_2. For example, if u = 1010101 and u_1 = 0000000 are code words of a group code and v = 0010101 is a received word, then in the standard array v is in the column headed by u. The probability that v is received when u is transmitted is $p_1 q_1^3 q_2^3$. The probability that v is received when some other code word, say u_1, is sent, is $p_2^3 q_2^4$. If the standard array is used, v will be decoded to u. However, the result of subtracting these expressions is:

$$q_2^3 [p_1 q_1^3 - p_2^3 q_2] \qquad (3.7)$$

From the inequalities, $q_1 \geq q_2 > p_2 \geq p_1$, it can be seen that for many values of q_1, q_2, p_1 and p_2 Equation 3.6 is positive and so the standard array results in maximum likelihood decoding. However, if p_1 is sufficiently small compared to p_2, Equation 3.6 becomes negative which means v is more likely the result of errors in u_1 than in u. For most of the possible received words this situation can never happen. It can happen when the received word differs from its column heading by an error of the type ($1 \to 0$) with probability

p_1, and there is some other code word such that the received word differs from it only by errors of the type (0→1) with probability p_2. Therefore, if the standard array is to be used in decoding a group code on an asymmetric path, the path must be limited to those values of q_1, q_2, p_1 and p_2 where a situation as described above does not occur. This places a restriction on the use of the group codes on an asymmetric path. However, it is not as severe a restriction as it may appear because the channel probabilities can be altered to some extent, and so it may be possible in some specific channels to vary q_1, q_2, p_1 and p_2 so they will lie in the required range for the standard array to give maximum likelihood decoding. Therefore, a modified form of Theorem 3.5 can now be proved for asymmetric channels, where q_1, q_2, p_1 and p_2 lie in certain intervals which depend on the particular group code to be used.

Theorem 3.6 Let V be an (n,k) linear binary code to be used with a binary asymmetric channel and assume that all the code vectors are equally likely to be transmitted. If every possible single error is to be corrected then the average probability of correct decoding is as large as possible for the code if the standard array is used as a decoding table, providing q_1, q_2, p_1 and p_2 lie in certain intervals which can be calculated for each particular code.

Proof: Using the notation of Theorem 3.5, let v_{ij} be the vector in the i^{th} row and j^{th} column of the decoding table. The code words, placed at the top of the columns, are denoted v_{oj} and have Hamming weight w_{oj}. Then the probability of correct decoding, if the code word v_{o_j} is transmitted, is:

$$q_1^{w_{o_j}} q_2^{n-w_{o_j}} + w_{o_j} p_1 q_1^{w_{o_j}-1} q_2^{n-w_{o_j}} + (n-w_{o_j})\, p_2\, q_1^{w_{o_j}} q_2^{n-w_{o_j}-1}$$

where q_1, q_2, p_1, p_2 are as designated in Figure 1.3. Since there are 2^k code words which are assumed equally probable, in averaging the probability of correct decoding the weighting factor 2^{-k} is used.

$$\Pr\,(\text{correct decoding}) = 2^{-k} \sum_{j=0}^{2^k-1} \{q_1^{w_{o_j}} q_2^{n-w_{o_j}}$$

$$+ w_{o_j} p_1 q_1^{w_{o_j}-1} q_2^{n-w_{o_j}} + (n-w_{o_j})\, p_2\, q_1^{w_{o_j}} q_2^{n-w_{o_j}-1}\}$$

There is one term in the sum for each possible received vector of binary symbols and that term is maximized in each case if that particular vector is decoded into the closest code vector in the Hamming sense, as if q_1, q_2, p_1, and p_2 lie in intervals calculated for each code, then the probability of no errors is greater than the probability of one error and this is greater than the probability of two errors and so on. Therefore the probability of correct decoding will be maximized if each received vector is decoded into the closest code vector. The remainder of this Theorem follows exactly as in Theorem 3.5.

Theorem 3.5, which was the only part of the preceeding theory to be restricted to a symmetric channel, has now been extended in the case of single error correcting codes to the asymmetric path or channel by Theorem 3.5. Similar theorems could be written for codes correcting double or more errors.

CHAPTER IV

NON - GROUP CODES

1. The Fixed Weight and the Sum Codes

While part of Chapter III has dealt with error correcting codes in general, much of the theory has been restricted to group codes. In this chapter other block codes will be introduced which are not required to satisfy the axioms of a group. A binary operation on the elements of the code will still be defined and can be shown by the following table:

+	0	1
0	0	1
1	1	0

As previously stated, each code word will consist of n digits, each digit being either a 0 or a 1. Also as this thesis takes into consideration only single error correcting codes each code word will be at a minimum Hamming distance of three from all others.

The two block codes introduced in this chapter are the "m-out-of-n code" and the "sum code". These codes are discussed by C.V. Freiman [3] and J.M. Berger [1] in papers dealing with error detection for completely asymmetric channels.

The m-out-of-n code, or (n/2)-out-of-n code as it is frequently written, is called a fixed weight code. An n position binary sequence may be a code word of an m-out-of-n code if and only if it contains exactly m 1's. When n=2m+1, the (n/2)-out-of-n code is taken as either the m-out-of-n code or the (m+1)-out-of-n code [3] . Use has been made in some communication systems of the fixed weight codes. Their adoption has come about mainly because of their error detection advantages in a communication path which is asymmetric to a large degree. The fixed weight codes are perfect error detection codes in completely asymmetric channels or paths, since any error of the type $(0 \rightarrow 1)$ would increase the fixed weight of the code word. In symmetric channels they will detect all odd numbers of error and will only fail to detect those even errors which correspond to an interchange of 0's with 1's. The main disadvantage in using fixed weight codes is that they are nonseparable.

<u>Definition 4.1</u> [1] A <u>separable code</u> is defined to be a code in which the bits or digits of the code word containing the information to be transmitted are distinct from the bits added to the code word to provide the capacity for error detection or correction.

In a fixed weight code it is the pattern or structure which provides the error detection or correction and it is not possible to separate off the redundant bits. Because the structure of the m-out-of-n codes is such that the information bits of the code and the error detection or correction capacity are bound together, modification of the code cannot be simply made. In using a fixed weight code the alphabet of the system would be established and then a fixed weight code, with a sufficient number of valid code word combinations, could be selected. Each code word would then correspond to a particular symbol of the alphabet. This may be a disadvantage in a case where a long string of symbols are transmitted. Since the redundancy is already included in each symbol it is not possible to take advantage of the economies that might be gained by coding over a whole string of symbols. Thus it seems that a nonseparable code may lack the the flexibility of a separable code and also may be less economical in coding over a large block of information.

Definition 4.2 [3] The redundancy R of a block code can be defined as

$$R = \frac{n - \log_2 \text{(number of code words)}}{n}$$

where n is the length of each code word.

It can be proved [3] that the (n/2)-out-of-n code is the least redundant binary block code which permits detection of all errors in a completely asymmetric channel or path.

A class of separate binary block codes have recently been introduced independently by J.M. Berger, H.J. Smith and C.V. Freiman. These codes, like the nonseparable m-out-of-n codes, permit perfect error detection over completely asymmetric paths. They have been called sum codes, and are denoted by $\sum$ (k,n-k). A code word of length n is formed by considering a set of k digits as information bits while the remaining (n-k) digits are used for coding purposes, and are called check bits. These check bits are formed by making them equal to the binary representation of the number of 0's in the k information bits. Thus the number of check-bits (n-k), is equal to the smallest integer that contains $\log_2 k$. It can be seen that the sum code detects all errors in a completely asymmetric path or channel. Since in using this path only 0's can become 1's, the number of 0's in the code word must decrease if an error occurs and so the sum of the number of 0's derived from the received information bits is smaller than the number represented by the check bits. This can be illustrated by an example. Consider (1100011011) to be a code word with n = 10, with k = 7 information bits and with n-k = 3 check bits. In the completely asymmetric path considered here, any error must be of the type (0→1). Suppose an error occurs in the information bits and the word (1100111011) is received. The binary representation of the sum of the number of 0's in the received information bits is 010. The check bits are 011 and it can be seen that 010<011 so an error is detected. Similarly, if an error occurs in the check bits and the word (1100011111), for example, is received then the sum of the number of 0's is 011 while the check bits are 111 and again 011< 111 detecting an error. In any other channel or path the sum code will detect all single errors and a large fraction of multiple errors. In using the sum codes for error detection, the k information bits can be any k digits so there are 2^k code words. Therefore in Definition 4.2, the redundancy R reduces to (n-k)/n. It can be proved [3] that these codes are the least redundant of all separable codes and that they are asymptotically twice as redundant as the (n/2)-out-of-n code.

While the fixed weight codes and sum codes were introduced because of their error detection ability, if the valid code words are restricted to those which are a Hamming distance of three apart, all single errors can be corrected and all other errors detected in a completely asymmetrical channel. For other channels all single errors can be corrected and a large number of multiple errors detected. The fixed weight and sum codes have an advantage over the group codes in that while they all correct single errors the fixed weight and sum codes simultaneously can detect a large number of additional errors.

2. The Standard Array as a Decoding Table for the Asymmetric Path

It has been found that decoding for group codes can be done by using the standard array as a decoding table, or by the use of syndromes, which give the same result. For the fixed weight and sum codes, there are also various procedures for maximum likelihood decoding. However, it will be convenient to define a decoding table for these block codes by extending the idea of a standard array. The number of columns in the decoding table will be the number of code words and the number of rows will be (n+1) where n is the length of each code word. The first row will consist of the code words and each of the other n rows will be the code words with a single error in the first to the nth digits respectively. All this is similar to the standard array decoding table for group codes. However in a general block code the n-tuples or code words do not necessarily satisfy the axioms of a group, nor will the rows of the array be cosets. The code words do not in general have inverses which are also code words, and the identity element is not necessarily a code word. The identity element and the n elements which differ from it by having a single 1 in the first to last digits respectively are called error pattern elements. If a block code has the identity element as a code word, then it will be placed in the left most position in the row of code words, and the error pattern elements will form the first column in the array. The remainder of the columns will be formed by adding the error pattern elements in turn to each code word, and placing the elements so formed in the column under that code word. If the identity element is not a code word for a block code, the error pattern elements will not form part of the array but will be placed in a column just to the left of the array. The array will be formed so that each column will be headed by a code word, and the element under the code word will have an error in the 1st digit, the next element will have an error in the 2nd digit, and the last element in each column will have an error in the nth digit. In other words, the array consists of the code words and the elements formed by adding the error pattern elements successively to each code word, even though the error pattern elements are not actually a part of the array. Therefore, whether the identity element is a code word or not, each element in a decoding table for a block code has the form of a code word plus an error pattern element.

In a group code the decoding table has 2^n words, so every possible received word of length n appears somewhere in the table. However, with general block codes it is possible to have n-tuples which do not appear in the decoding table. If a received word is to be decoded, it must differ from a code word by an error pattern element, and thus will be decoded into the code word heading its column. If for some received word, this is not the case, that is if the received word does not appear in the table, then an uncorrectable error will be detected.

The use, in Chapter III, of a standard array as a decoding table for group codes has now been extended and a similar array has been defined to use as a decoding table for general block codes. Also a theorem similar to Theorem 3.5 and Theorem 3.6 can now be proved.

As in Theorem 3.6 it is necessary to consider whether or not any cases arise, in the use of the standard array, which do not result in maximum likelihood decoding.

It can be shown that for the fixed weight code, the standard array does give maximum likelihood decoding. The probability that a received word, v, is the result of a single error in the code word heading its column, and denoted by u, is;

$$p_2\, q_1^{\,w_{oj}}\, q_2^{\,n-w_{oj}-1} \qquad (4.1)$$

or

$$p_1\, q_1^{\,w_{oj}-1}\, q_2^{\,n-w_{oj}} \qquad (4.2)$$

where the weight, w_{oj}, is constant for all j in each particular code. The word v must differ from all code words, except u in at least two places. Thus, the sum of the powers of p_1 and p_2 must be greater than or equal to two. If v is the result of errors in some code word other than u, this probability will have one of the following forms:

$$p_2^2 q_1^{w_oj} q_2^{n-w_oj-2},\ p_2^3 q_1^{w_oj} q_2^{n-w_oj-3},\ \ldots, \tag{4.3}$$

$$p_1^2 q_1^{w_oj-2} q_2^{n-w_oj},\ p_1^3 q_1^{w_oj-3} q_2^{n-w_oj},\ \ldots, \tag{4.4}$$

$$p_1 p_2 q_1^{w_oj-1} q_2^{n-w_oj-1},\ p_1^2 p_2 q_1^{w_oj-2} q_2^{n-w_oj-1},$$
$$p_1 p_2^2 q_1^{w_oj-1} q_2^{n-w_oj-2},\ p_1^2 p_2^2 q_1^{w_oj-2} q_2^{n-w_oj-2},\ \ldots, \tag{4.5}$$

If the standard array is to give maximum likelihood decoding it must be shown that v is more likely the result of an error in the code word heading its column than the result of errors in some other code word. That is, it must be shown that the probabilities shown in (4.1) and (4.2) are larger than any probabilities of the form (4.3) (4.4) or (4.5).

Suppose the probability that v is received, when u is transmitted, is as shown in (4.1). If the probability that v is received, when some other code word is transmitted is of the form (4.3) then

$$p_2 q_1^{w_oj} q_2^{n-w_oj-1} - p_2^2 q_1^{w_oj} q_2^{n-w_oj-2}$$
$$= p_2 q_1^{w_oj} q_2^{n-w_oj-2} [q_2 - p_2] > 0$$

where equality holds in the rather trivial case where $p_2 = 0$. Similarly, the difference between the term in (4.1) and the other terms in (4.3) is positive (or zero) as a q_2 in the second term of the difference is just replaced by the smaller value p_2.

If the probability of the form (4.4) is considered then

$$p_2 q_1^{w_{oj}} q_2^{n-w_{oj}-1} - p_1^2 q_1^{w_{oj}-2} q_2^{n-w_{oj}}$$
$$= q_1^{w_{oj}-2} q_2^{n-w_{oj}-1} [p_2 q_1^2 - p_1^2 q_2] > 0$$

A similar result occurs if the other terms in (4.4) are considered as this just means a q_1 is replaced by a p_1, decreasing the second term in the difference.

If the probability of the form (4.5) is considered then

$$p_2 q_1^{w_{oj}} q_2^{n-w_{oj}-1} - p_1 p_2 q_1^{w_{oj}-1} q_2^{n-w_{oj}-1}$$
$$= p_2 q_1^{w_{oj}-1} q_2^{n-w_{oj}-1} [q_1 - p_1] \geq 0$$

This result also holds if the other terms in (4.5) are considered as this just means a q_1 is replaced by a p_1, a q_2 by a p_2 or both, thus decreasing the second term in the difference.

Suppose the probability that v is received, when the code word, u, which heads its column, is transmitted, is as shown in (4.2). If the probability that v is received when some other code word is transmitted, is of the form (4.3) then

$$p_1 q_1^{w_{oj}-1} q_2^{n-w_{oj}} - p_2^2 q_1^{w_{oj}} q_2^{n-w_{oj}-2}$$
$$= q_1^{w_{oj}-1} q_2^{n-w_{oj}-2} [p_1 q_2^2 - p_2^2 q_1]$$

This expression and the similar expressions using other terms from (4.3) may be positive, negative or zero, depending on the values of q_1, q_2, p_1 and p_2.

If the probability of the form (4.4) is considered then

$$p_1 q_1^{w_{oj}-1} - p_1^2 q_1^{w_{oj}-2} q_2^{n-w_{oj}}$$

$$= p_1 q_1^{w_o j-2} q_2^{n-w_o j} \;[q_1 - p_1] \geq 0$$

Similarly, if the other terms in (4.4) are used, the second term in the difference decreases as a q_1 is replaced by a p_1.

If the probability of the form (4.5) is considered then

$$p_1 q_1^{w_o j-1} q_2^{n-w_o j} - p_1 p_2 q_1^{w_o j-1} q_2^{n-w_o j-1}$$
$$= p_1 q_1^{w_o j-1} q_2^{n-w_o j-1} \;[q_2 - p_2] \geq 0$$

This result also holds for the other terms in (4.5) as a q_1 is replaced by a p_1, a q_2 by a p_2 in both, thus decreasing the second term in the difference.

It appears that if the standard array is used as a decoding table, it will result in maximum likelihood decoding, with one possible exception. The exception may occur when the probability that v is received when u is transmitted is $p_1 q_1^{w_o j-1} q_2^{n-w_o j}$ and there is some other code word, say u_1 such that the probability that v is received when u_1 is transmitted is of the form (4.3). If such a code word u_1 exists, then for v to be received when u_1 is transmitted there must be no error of the type ($1 \to 0$). This means that for every digit in v which is 0 that digit in u_1 must also be a 0. It can be shown that this cannot happen. In a fixed weight code each code word has w_{oj} 1's and $(n-w_{oj})$ 0's where w_{oj} is a constant for each code. The word v differs from u in one place and the probability of this error is p_1. Therefore, a 1 in u changes to a 0 and v has $(w_{oj} - 1)$ 1's and $(n-w_{oj} + 1)$ 0's. As the code word u described above must have 0's in the same positions where v has, u must have at least $(n-w_{0\,j}+1)$ 0's. This is impossible and therefore the standard array gives maximium likelihood decoding for the fixed weight code.

For the sum code there does not appear to be any general method to show that the standard array results in maximum likelihood decoding. However, for each particular sum code this can be done by comparing, for all the entries in the table, the probability that a received ward was a result of a single error in its column heading with the probability that the received ward was a result of errors in some other code word. This has been done for the

$\sum(7,3)$ code and it has been found that the standard array does give maximum likelihood decoding in this case.

The following theorem is an extension of Theorem 3.6 to non-group codes. It can be applied to fixed weight codes and any block codes for which it can be shown that no cases occur which destroy maximum likelihood decoding.

<u>Theorem 4.1</u> Let V be a binary block code where the length of each word is n. Assume a binary asymmetric path and that all code words are equally likely to be transmitted. If all single errors are to be corrected, the average probability of correct decoding is as large as possible for the code if the above defined standard array is used as a decoding table providing the code is a fixed weight code or a block code, where it can be verified that no exceptions to maximum likelihood decoding can arise.

<u>Proof</u> Using the notation of Theorems 3.5 and 3.6 let

v_{ij} be the n-tuple in the i^{th} row and j^{th} column of the decoding table. The code words, placed at the top of the columns, are denoted by v_{0j} and have Hamming veight w_{0j}. Let the total number of the code words be M. Then the probability of correct decoding, if the code word v_{0j} is transmitted, is:

$$q_1^{w_{0j}} q_2^{n-w_{0j}} + w_{0j} p_1 q_1^{w_{0j}} + (n-w_{0j}) p_2 q_1^{w_{0j}} q_2^{n-w_{0j}-1}$$

where q_1, q_2, p_1 and p_2 are as designated in Figure 1.3. Since there are M code words which are assumed equally probable, in averaging the probability of correct decoding the weighting factor 1/M is used.

$$\Pr(\text{correct decoding}) = \frac{1}{M}\sum_{j=1}^{M} \{q_1^{w_{0j}} q_2^{n-w_{0j}} + w_{0j} p_1 q_1^{w_{0j}-1} q_2^{n-w_{0j}}$$

$$+ (n-w_{0j}) p_2 q_1^{w_{0j}} q_2^{n-w_{0j}-1}\}$$

There is one term in the sum for each possible received n-tuple which differs from some code word in no more than one place. That term is maximized in each case if that particular word is decoded into the closest code vector in the Hamming sense providing the code is a fixed weight code or a block code which satisfies the condition stated in this theorem. Then the probability of no error is greater than the probability of one error and this is greater than the probability of two errors and so on. Therefore, the probability of correct decoding will be maximized if each received n-tuple is decoded into the closest code vector.

Now suppose that a particular word v appears in the decoding table under the code word u, which is at a Hamming distance w. Let u_1 be any other code word in the array. Since Hamming distance is a metric function it must satisfy the relation

$$d(u,u_1) \leq d(u,v) + d(u_1,v)$$

$$d(u,u_1) - d(u,v) \leq d(u_1,v)$$

$$d(u_1,v) \geq d(u,u_1) - d(u,v) \qquad (4.1)$$

In this decoding table $d(u,v) = 0$ or 1.

$d(u,u_1) \geq 3$ as this restriction has been placed on all code words in a single error correcting code.

Therefore, $d(u_1,v) \geq 2$.

That is, v is closer to u than to u_1.

CHAPTER V

DISCUSSION OF PARTICULAR CODES

1. Five Example Codes

In this chapter five codes will be introduced as examples to illustrate the preceeding theory and Theorems 3.5, 3.6, 4.1. In these codes n is chosen to be small. This is because in the use of reflecting meteorites which are rapid and of short duration, it is then possible to utilize the maximum percentage of available time. When n is small the number of code words is also small, so it may be necessary to use combinations of words to generate new characters. However, this not a serious drawback as due to the high carrier frequency of these systems of 200 - 300 Mcs. , an extremely high pulse speed could be used.

The five codes considered here are as follows;

Code I This is a (7,4) group code, that is $n=7$ and $k=4$. The code words and decoding table are shown in Table 5.1. There are $2^k = 2^4 = 16$ code words and $2^{n-k} = 2^3 = 8$ cosets. This is a single error correcting code so each code word is at a distance of at least three from all other code words and the relationship $n + 1 = 2^{n-k} = 8$ is satisfied. Since this is a Hamming code it can be represented by a parity-check matrix.

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

Information can be encoded by taking the first, second and fourth symbols as parity - check symbols, as each of them only occur in one of the parity-check relations. For example [7, page 65] to encode 1100 the three parity check symbols are inserted, $p_1 p_2 1 p_3 100$ and the parity check relations which must be satisfied are:

$$p_3 + 1 = 0$$

$$p_2 + 1 = 0$$

$$p_1 + 1 + 1 = 0$$

Therefore, $p_1 = 0$, $p_2 = 1$ and $p_3 = 1$ and the code word is 0111100. For any received code word the syndrome will be the binary representation of the symbol in error since each column of H is the binary representation of the column number. Using Theorem 3.6 the probability of correct decoding for this code is

$$Pr = 2^{-4} \left[\sum_{j=1}^{16} \{ q_1^{w_{0j}} q_2^{7-w_{0j}} + w_{0j} p_1 q_1^{w_{0j}-1} q_2^{7-w_{0j}} + (7-w_{0j}) p_2 q_1^{w_{0j}} q_2^{7-w_{0j}-1} \} \right]$$

Summing over j and collecting terms this becomes:

$$Pr = 2^{-4} [q_2^7 + 7p_2 q_2^6 + 7\{q_1^3 q_2^4 + 3p_1 q_1^2 q_2^4 + 4p_2 q_1^3 q_2^3\} + 7\{q_1^4 q_2^3 + 4p_1 q_1^3 q_2^3 + 3p_2 q_1^4 q_2^2\} + q_1^7 + 7p_1 q_1^6]$$

In the special case where the probablity of an error is q, and the probablity of no error is p this becomes

$$Pr = 2^{-4} [16\{ q^7 + 7pq^6 \}]$$

<u>Code II</u> This code is a 4-out-of-7 fixed weight

code. It is an (n/2)-out-of-n code, or if n is written as 2m + 1 it is an (m+1)-out-of-n code where n=7 and (m+1)=4. The code words and decoding table are shown in Table 5.2. The code shown here has seven code words but any seven-digit words of weight four could be included as code words, as long as they satisfy the condition for a single-error correcting code that each word is at a Hamming distance of at least three from all other code words. Changes in the code words will not affect the average probability of correct decoding since in a fixed weight code the probability of correct decoding since in a fixed weight code the probability of correct decoding is obviously the same for each column in the talbe. Using Theorem 4.1, with n=7 and the number of code words M=7, the probability of correct decoding is:

$$Pr = \frac{1}{7} \sum_{j=1}^{7} \{ q_1^{w_{0\,j}} q_2^{7-w_{0\,j}} + w_{0\,j} p_1 q_1^{w_{0\,j}-1} q_2^{7-w_{0\,j}} + (7-w_{0\,j}) p_2 q_1^{w_{0\,j}} q_2^{7-w_{0\,j}-1} \}$$

Since $w_{0\,j}=4$ for $j=1,\dots,7$ this becomes

$$Pr = \frac{1}{7} \cdot 7 \,\{ q_1^4 q_2^3 + 4p_1 q_1^3 q_2^3 + 3p_2 q_1^4 q_2^2 \}$$
$$= q_1^4 q_2^3 + 4p_1 q_1^3 q_2^3 + 3p_2 q_1^4 q_2^2$$

Code III This is another fixed weight code, the 5-out-of-9 code. The code words and decoding table are shown in Table 5.3. In Table 5.3 there are sixteen code words but as in Code II any 9 digit word of weight five could be a code word if it was a Hamming distance of at least three from all other code words. Again using Theorem 4.1, the probability of correct decoding is:

$$Pr = \frac{1}{M} \sum_{j=1}^{M} q_1^{w_{0\,j}} q_2^{9-w_{0\,j}} + w_{0\,j}\, p_1 q_1^{w_{0\,j}-1} q_2^{9-w_{0\,j}} + (9-w_{0\,j})\, p_2 q_1^{w_{0\,j}} q_2^{9-w_{0\,j}-1}$$

Since $w_{0\,j} = 5$ for $j=1,\ldots,M$, this becomes:

$$Pr = \frac{1}{M} \cdot M \{q_1^5 q_2^4 + 5p_1 q_1^4 q_2^4 + 4p_2 q_1^5 q_2^4\}$$

$$Pr = q_1^5 q_2^4 + 5p_1 q_1^4 q_2^4 + 4p_2 q_1^5 q_2^3$$

<u>Code IV</u> This is also a fixed weight code, a 5-out-of-10 code. Table 5.4 shows the code words and the decoding table. There are twenty-six code words in the decoding table but as in all fixed weight codes the particular words chosen or the number of words in the code does not affect the average probability of correct decoding. From Theorem 4.1 it follows that:

$$Pr = \frac{1}{M} \sum_{j=1}^{M} \{q_1^{w_{0\,j}} q_2^{10-w_{0\,j}} + w_{0\,j} p_1 q_1^{w_{0\,j}-1} q_2^{10-w_{0\,j}} + (10-w_{0\,j})\, p_2 q_1^{w_{0\,j}} q_2^{10-w_{0\,j}-1}\}$$

Since $w_{0\,j} = 5$ for $j=1,\ldots,M$, then

$$Pr = \frac{1}{M} \cdot M\{q_1^5 q_2^5 + 5p_1 q_1^4 q_2^5 + 5p_2 q_1^5 q_2^4\}$$

$$Pr = q_1^5 q_2^5 + 5p_1 q_1^4 q_2^5 + 5p_2 q_1^5 q_2^4$$

<u>Code V</u> This is a sum code where n=10 and k=7. Therefore, there are 7 information digits and 3 check digits where the check digits are the binary number which corresponds

to the sum of the 0's in the 7 information digits. This code can be denoted by $\sum(3,7)$. The code words and decoding table are shown in Table 5.5. From Theorem 4.1, the probability of correct decoding can be written as follows:

$$Pr = \frac{1}{16} \sum_{j=1}^{16} \{q_1^{w_{0j}} q_2^{10-w_{0j}} + w_{0j} p_1 q_1^{w_{0j}-1} q_2^{10-w_{0j}} +$$

$$(10-w_{0j}) p_2 q_1^{w_{0j}} q_2^{10-w_{0j}-1}\}.$$

In summing over j there is one case where $w_{0j} = 7$, one case where $w_{0j} = 3$ and seven cases where $w_{0j} = 6$ and $w_{0j} = 4$, so the probability can be written as

$$Pr = \frac{1}{16}[q_1^7 q_2^3 + 7p_1 q_1^6 q_2^3 + 3p_2 q_1^7 q_2^2 + q_1^3 q_2^7 + 3p_1 q_1^2 q_2^7 + 7p_2 q_1^3 q_2^6 + 7\{q_1^6 q_2^4 + 6p_1 q_1^5 q_2^4 + 4p_2 q_1^6 q_2^3\} + 7\{q_1^4 q_2^6 + 4p_1 q_1^3 q_2^6 + 6p_2 q_1^4 q_2^5\}].$$

2. <u>Applicability of These Codes to the Asymmetric Path.</u>

As this paper is considering communication by weak reflected signals in a noisy media, it is necessary to consider which codes would be suitable for use in this case, where the probability that a 0 becomes a 1 is greater than the probability that a 1 becomes a 0. If a code is transmitted on a path where $q_1, q_2, p_1,$ and p_2 are as shown in Figure 1.2 and the inequality $q_1 \geq q_2 > p_2 \geq p_1$ holds, the probability of correct decoding for that code can be found by using the equations in Theorems 3.6 and 4.1. This same code could be transmitted on a channel where symmetry was assumed. This is the same as assuming that there is no difference between a 0 and a 1 and therefore the probability of no error occuring is $Pr\binom{1 \text{ rec'd } 1 \text{ sent}}{0 \text{ rec'd } 0 \text{ sent}} = \frac{q_1 + q_2}{2}$ and the probability of an error is $\frac{p_1 + p_2}{2}$, as stated in Chapter I. In this case also the probability of correct decoding could be calculated. In the asymmetric case where 0's and 1's are distinquished between the probability of correct decoding will be denoted Pr(A), while in the symmetric case where the 0's and 1's are not distinquished between this probability will be denoted Pr(S). If for a particular code Pr(A)>Pr(S), then that code would be more suitable for use on an asymmetric path than on a symmetric one. If, however, Pr(S)>Pr(A), the opposite would hold. For a particular code the comparison between Pr(A) and Pr(S) can be done algebraically or numerically.

For the group code, Code I, it can be showed algebraically that $Pr(S) \geq Pr(A)$. As was previously shown for Code I, $Pr(A) = 2^{-4}[q_2^7 + 7p_2 q_2^6 + 7\{q_1^3 q_2^4 + 3p_1 q_1^2 q_2^4 + 4p_2 q_1^3 q_2^3\} + 7\{q_1^4 q_2^3 + 4p_1 q_1^3 q_2^3 + 3p_2 q_1^4 q_2^2\} + q_1^7 + 7p_1 q_1^6]$.

In the case where the probability of no error occuring is $q = \frac{q_1 + q_2}{2}$ and the probability of an error is $p = \frac{p_1 + p_2}{2}$, the probability of correct decoding is:

$$Pr(S) = 2^{-4}[16 (\tfrac{q_1 + q_2}{2})^7 + 7 (\tfrac{p_1 + p_2}{2})(\tfrac{q_1 + q_2}{2})^6 \}]$$

$$= 2^{-4} . 16[\tfrac{1}{128}(q_1 + q_2)^7 + \tfrac{7}{128}(p_1 + p_2)(q_1 + q_2)^6]$$

$$= 2^{-4} . 16[\tfrac{1}{128}\{q_1^7 + \tbinom{7}{1}q_1^6 q_2 + \tbinom{7}{2}q_1^5 q_2^2 + \tbinom{7}{3}q_1^4 q_2^3 + \tbinom{7}{4}q_1^3 q_2^4 + \tbinom{7}{5}q_1^2 q_2^5 + \tbinom{7}{6}q_1 q_2^6 + q_2^7\} + \tfrac{7}{128}\{(p_1 + p_2)(q_1^6 + \tbinom{6}{1}q_1^5 q_2 + \tbinom{6}{2}q_1^4 q_2^2 + \tbinom{6}{3}q_1^3 q_2^3 + \tbinom{6}{4}q_1^2 q_2^4 + \tbinom{6}{5}q_1 q_2^5 + q_2^6\}].$$

$$= 2^{-4}[\tfrac{1}{8}q_1^7 + \tfrac{7}{8}q_1^6 q_2 + \tfrac{21}{8}q_1^5 q_2^2 + \tfrac{35}{8}q_1^4 q_2^3 + \tfrac{35}{8}q_1^3 q_2^4 + \tfrac{21}{8}q_1^2 q_2^5 + \tfrac{7}{8}q_1 q_2^6 + \tfrac{1}{8}q_2^7$$

$$+ p_1(\tfrac{7}{8}q_1^6 + \tfrac{42}{8}q_1^5 q_2 + \tfrac{105}{8}q_1^4 q_2^2 + \tfrac{140}{8}q_1^3 q_2^2 + \tfrac{105}{8} q_1^2 q_2^4 + \tfrac{42}{8}q_1 q_2^5 + \tfrac{7}{8}q_2^6)$$

$$+ p_2(\tfrac{7}{8}q_1^6 + \tfrac{42}{8}q_1^5 q_2 + \tfrac{105}{8}q_1^4 q_2^2 + \tfrac{140}{8}q_1^3 q_2^3 + \tfrac{105}{8}q_1^2 q_2^4 + \tfrac{42}{8}q_1 q_2^5 + \tfrac{7}{8}q_2^6)].$$

In showing that $Pr(S) \geq Pr(A)$, or what is the same thing, that $Pr(A)-Pr(S) \leq 0$ the part of the expression $Pr(A)-Pr(S)$ involving no error can be considered first, then the part involving an error of the type $(1\to 0)$ whose probability is p_1 and then the part involving an error of the type $(0\to 1)$ with probability p_2. The part of $Pr(A)-Pr(S)$ with no error is $q_2^7 + 7q_1^3 q_2^4 + 7q_1^4 q_2^3 + q_1^7 - \frac{1}{8}q_1^7 - \frac{7}{8}q_1^6 q_2 - \frac{21}{8}q_1^5 q_2^2 - \frac{35}{8}q_1^4 q_2^3 - \frac{35}{8}q_1^3 q_2^4 - \frac{21}{8}q_1^2 q_2^5 - \frac{7}{8}q_1 q_2^6 - \frac{1}{8}q_2^7$
$= \frac{7}{8}q_2^7 + \frac{21}{8}q_1^3 q_2^4 + \frac{21}{8}q_1^4 q_2^3 + \frac{7}{8}q_1^7 - \frac{7}{8}q_1^6 q_2 - \frac{21}{8}q_1^5 q_2^2 - \frac{21}{8}q_1^2 q_2^5 - \frac{7}{8}q_1 q_2^6$.

Let $q_1 = kq_2$ where k is a variable, and this becomes

$$\tfrac{1}{8}q_2^7(7 + 21k^3 + 21k^4 + 7k^7 - 7k^6 - 21k^5 - 21k^2 - 7k)$$
$$= \tfrac{7}{8}q_2^7(k^7 - k^6 - 3k^5 + 3k^4 + 3k^3 - 3k^2 - k + 1)$$
$$= \tfrac{7}{8}q_2^7(k-1)^4(k+1)^3.$$

The part of $Pr(A)-Pr(S)$ containing the term p_1 is

$$p_1(21q_1^2 q_2^4 + 28q_1^3 q_2^3 + 7q_1^6 - \tfrac{7}{8}q_1^6 - \tfrac{42}{8}q_1^5 q_2 - \tfrac{105}{8}q_1^4 q_2^2 - \tfrac{140}{8}q_1^3 q_2^3 - \tfrac{105}{8}q_1^2 q_2^4 - \tfrac{42}{8}q_1 q_2^5 - \tfrac{7}{8}q_2^6).$$

Letting $q_1 = kq_2$ and $p_1 = 1-q_1 = 1-kq_2$ this becomes

$$q_2^6(1-kq_2)(\tfrac{49}{8}k^6 - \tfrac{42}{8}k^5 - \tfrac{105}{8}k^4 + \tfrac{84}{8}k^3 + \tfrac{63}{8}k^2 - \tfrac{42}{8}k - \tfrac{7}{8})$$
$$= \tfrac{7}{8}q_2^6(1-kq_2)(7k^6 - 6k^5 - 15k^4 + 12k^3 + 9k^2 - 6k - 1)$$
$$= \tfrac{7}{8}q_2^6(1-kq_2)(k-1)^3(k+1)^2(7k+1).$$

The part of Pr(A)-Pr(S) containing the term p_2 is

$$p_2\ (7q_2^{\,6} + 28q_1^{\,3}q_2^{\,3} + 21q_1^{\,4}q_2^{\,2} - \tfrac{7}{8}q_1^{\,6} - \tfrac{42}{8}q_1^{\,5}q_2 - \tfrac{105}{8}q_1^{\,4}q_2^{\,2} - \tfrac{140}{8}q_1^{\,3}q_2^{\,3} - \tfrac{105}{8}q_1^{\,2}q_2^{\,4} - \tfrac{42}{8}q_1 q_2^{\,5} - \tfrac{7}{8}q_2^{\,6})$$

Letting $q_1 = kq_2$ and $p_2 = 1-q_2$ this becomes

$$q_2^{\,6}(1-q_2)(7 + 28k^3 + 21k^4 - \tfrac{7}{8}k^6 - \tfrac{42}{8}k^5 - \tfrac{105}{8}k^4 - \tfrac{140}{8}k^3 - \tfrac{105}{8}k^2 - \tfrac{42}{8}k - \tfrac{7}{8})$$

$$= -\tfrac{7}{8}q_2^{\,6}(1-q_2)(k-1)^3(k+1)^2(k+7).$$

Adding all these terms

$$\begin{aligned}
\Pr(A)-\Pr(S) &= 2^{-4}\cdot \tfrac{7}{8}q_2^{\,7}(k-1)^4(k+1)^3 + \\
&\quad \tfrac{7}{8}q_2^{\,6}(1-kq_2)(k-1)^3(k+1)^2(7k+1) - \\
&\quad \tfrac{7}{8}q_2^{\,6}(1-q_2)(k-1)^3(k+1)^2(k+7). \\
&= 2^{-4}\cdot \tfrac{7}{8}q_2^{\,6}(k-1)^3(k+1)^2[q_2(k-1)(k+1) + \\
&\quad (1-kq_2)(7k+1) - (1-q_2)(k+7)]. \\
&= 2^{-4}\cdot \tfrac{7}{8}q_2^{\,6}(k-1)^3(k+1)^2[q_2k^2 - q_2 + 7k + 1 - 7k^2q_2 \\
&\quad - kq_2 - k - 7 + q_2k + 7q_2]. \\
&= 2^{-4}\cdot \tfrac{7}{8}q_2^{\,6}(k-1)^3(k+1)^2[6q_2(-k^2+1) + 6(k-1)]. \\
&= 2^{-4}\cdot \tfrac{7}{8}q_2^{\,6}(k-1)^3(k+1)^2[6(k-1)\{1 - q_2(k+1)\}]. \\
&= 2^{-4}\cdot \tfrac{42}{8}q_2^{\,6}(k-1)^4(k+1)^2[1 - q_2(k+1)]. \qquad (4.1)
\end{aligned}$$

To show that $\Pr(A)-\Pr(S) \leq 0$, it is necessary to consider the range of values q_2 and k may take. q_2 is a probability where the inequality $q_i \geq p_i$, $i = 1,2$ holds. Therefore, q_2 must lie in the interval $[\frac{1}{2},1]$. Since $q_1 = kq_2$ and $q_1 \geq q_2$, then $k \geq 1$. k will have its maximum value when q_2 has its minimum value of $\frac{1}{2}$, and for this product the following condition must hold:

$kq_2 = q_1 \leq 1.$

Therefore, $k \leq 2$.

Therefore, $1 \leq k \leq 2$.

Considering Equation 4.1 for q_2 and k in their respective ranges, the first term in the equation, $\frac{42}{8}q_2{}^6(k-1)^4(k+1)^2 \geq 0$. The second term in Equation 4.1, $1-q_2(k+1)$, will be positive if and only if $q_2(k+1) < 1$. If q_2 and k each take their minimum values $q_2(k+1) = \frac{1}{2}(1+1) = 1$. For any other values of q_2 and k in their respective ranges, $q_2(k+1) > 1$. Therefore, $[1-q_2(k+1)] \leq 0$.

Therefore, Equation 4.1 is negative or zero. That is, $Pr(A)-Pr(S) \leq 0$ for Code I. Since the zero occurs in the trivial case where k=0, that is, where $q_1 = q_2$ and Pr(A) and Pr(S) coincide, then if the trivial case is neglected it can be said that for Code I, $Pr(S) > Pr(A)$.

Similar evaluations can be done for Codes II, III, IV and V. However the equations arrived at in these cases do not factor or simplify as the equation for Code I did. However the comparison between Pr(A) and Pr(S) can be done numerically and these results for thefive codes are shown in Tables 5.6, 5.7, 5.8, 5.9, 5.10 respectively. The first four columns in each table are q_1, q_2, p_1 and p_2 where q_2 varies from 1.00 to 0.50 decreasing by increments of 0.02. Since $q_1 \geq q_2$, for each particular q_2, q_1 varies from 1.00 to q_2, decreasing by increments of 0.02. The fifth column gives Pr(A) and Pr(S) alternately where the probability of no error is $\frac{q_1+q_2}{2}$ for Pr(S), with q_1 and q_2 being the probabilities used to calculate the previous Pr(A). Column six gives the difference

Pr(A)-Pr(S) for each pair of these in column five.

Table 5.6 shows that for Code I, $Pr(S) \geq Pr(A)$ which is the same result as was obtained algebraically. The result for Code II and Code III as shown in Table 5.7 and Table 5.8 respectively is that $Pr(A) \geq Pr(S)$. Tables 5.9 and 5.10, which are drawn for Codes IV and V respectively, show that $Pr(A) \geq Pr(S)$ providing q_1 and q_2 are sufficiently large, but for small q_1 and q_2 $Pr(S) > Pr(A)$. In Code IV if $1.00 \leq q_2 \leq 0.88$ and $q_1 \geq q_2$ then $Pr(A) \geq Pr(S)$. Also for $q_2 = 0.86$ and $1.00 \leq q_1 \leq 0.92$, for $q_2 = 0.84$ and $1.00 \leq q_1 \leq 0.96$, for $q_2 = 0.82$ and $1.00 \leq q_1 \leq 0.98$, and for $q_2 = 0.80$ and $q_1 = 1.00$ this same inequality holds. For all other values of q_1 and q_2 the inequality $Pr(S) > Pr(A)$ holds. In Code V, if $1.00 \leq q_2 \leq 0.88$ and $q_1 \geq q_2$ then $Pr(A) \geq Pr(S)$. This is also the case for $q_2 = 0.86$ and $1.00 \leq q_1 \leq 0.92$, for $q_2 = 0.84$ and $1.00 \leq q_1 \leq 0.94$, for $q_2 = 0.82$ and $1.00 \leq q_1 \leq 0.96$, for $q_2 = 0.80$ and $1.00 \leq q_1 \leq 0.98$ and for $q_2 = 0.78$ and $q_1 = 1.00$. Otherwise $Pr(S) > Pr(A)$.

Thus it would appear that of these five codes, Code I would be most suitable for transmission on a channel where it was found that there was no difference in the behavior of a 1 and a 0. Codes II and III appear to be suited for use on an asymmetric channel. If q_1 and q_2 are sufficiently high, Codes IV and V seem to be more suited to an asymmetric path than a symmetric one; however, if the error probabilities are higher the opposite of this holds.

These tables only take into consideration the error correcting capacity of the codes. However, Codes II, III, IV and V also simultaneously detect a large number of errors. With Code I, if a single error occurs, the received word will be in the column of the transmitted word and the error will be corrected. However, if two or more errors occur, it will be in the column of some other code word and will be incorrectly decoded. The decoding table for Code I contains 2^7 words or all possible received words. For the other codes there are a great many possible words which do not appear in the decoding tables. If a single error occurs, it will, of course, be corrected. However, if two or more errors occur, the received word is not necessarily incorrectly decoded. If the errors are such that the received word differs from some other code word in just one place then it will be incorrectly decoded. Otherwise, the received word will not appear in the decoding table and an error will be detected. This tends to give these four codes an advantage over Code I although this is not incorporated in the results of the tables.

A great deal of analysis would be required to investigate the results shown in Tables 5.6, 5.7, 5.8, 5.9 and 5.10. Only some of the general trends seen in these tables will be discussed here. An analysis of this data would make it possible to take advantage of the code characteristics in selecting the best type of code for a particular channel. A trend which can be seen in all these codes is as q_1 approaches q_2, Pr(A) and Pr(S) become closer together, that is, the term $|Pr(A)-Pr(S)|$ decreases. This is to be expected. Another property which can be seen is that since $q_1 \geq q_2$, a word with a high number of 1's will contribute a larger amount to the probability of correct decoding than a word with a high number of 0's. Also it can be seen that the product of a large probability, say q_1, with a smaller probability, say q_2, is smaller than the product of $(\frac{q_1+q_2}{2})$ by $(\frac{q_1+q_2}{2})$. For example, if $q_1=0.9$ and $q_2=0.7$, $q_1 q_2=.63$, but $(\frac{q_1+q_2}{2})^2 = .64$. However, if the term $q_1^4 q_2^3$ is considered for these probabilities, it is found that $q_1^4 q_2^3>(\frac{q_1+q_2}{2})^7$. Therefore, it can be seen that a large number of 1's tend to make Pr(A)>Pr(S) and tend to offset the above. In Code I it has been shown that Pr(S)>Pr(A). A possible explanation is that it is a group code and therefore every word has an inverse. Therefore, for a word of all 1's which would contribute a large amount to Pr(A), there must be a word of all 0's which would contribute a much smaller amount. Similarly for every word with three 1's and four 0's there is a word with four 1's and three 0's, and so the result of a term with a high number of 1's is offset by its inverse with a low number of 1's.

Also it has been shown that for certain values of q_1, q_2, p_1 and p_2 the standard array does not result in maximum likelihood decoding, although for the symmetric case where $q_1=q_2$ the decoding is always maximum likelihood. This means that in using the asymmetric path, a received word may be decoded to some code word which is not the code word most likely to have been transmitted.

For Codes II and III, every word in the decoding table has the number of 1's greater than or at least equal to the number of 0's. This possibly explains why $Pr(A)\geq Pr(S)$ throughout these codes. For Codes IV and V, however, the number of 0's and 1's are equal and although the words do not have exact inverses, for every word of six 1's and four 0's there is a word of four 1's and six 0's. Possibly this is why terms with a large number of 1's are not large enough to consistently make $Pr(A) \geq Pr(S)$, and so as Tables 5.9 and 5.10 show this is true only for sufficiently large q_1 and q_2 and otherwise $Pr(S) \geq Pr(A)$.

However, for a complete analysis of these results many more examples of these three code types would be required and more numerical results.

CODE I

0000000	0101010	1111111	1010101	0011001	0110011	1100110	1001100
1000000	1101010	0111111	0010101	1011001	1110011	0100110	0001100
0100000	0001010	1011111	1110101	0111001	0010011	1000110	1101100
0010000	0111010	1101111	1000101	0001001	0100011	1110110	1011100
0001000	0100010	1110111	1011101	0010001	0111011	1101110	1000100
0000100	0101110	1111011	1010001	0011101	0110111	1100010	1001000
0000010	0101000	1111101	1010111	0011011	0110001	1100100	1001110
0000001	0101011	1111110	1010100	0011000	0110010	1100111	1001101

1110000	1011010	0001111	0100101	1101001	1000011	0010110	0111100
0110000	0011010	1001111	1100101	0101001	0000011	1010110	1111100
1010000	1111010	0101111	0000101	1001001	1100011	0110110	0011100
1100000	1001010	0011111	0110101	1111001	1010011	0000110	0101100
1111000	1010010	0000111	0101101	1100001	1001011	0001110	0110100
1110100	1011110	0001011	0100001	1101101	1000111	0010010	0111000
1110010	1011000	0001101	0100111	1101011	1000001	0010100	0111110
1110001	1011011	0001110	0100100	1101000	1000010	0010111	0111101

FIGURE 5.1

CODE II

Error Pattern							
0000000	0111100	1011010	1101001	1100110	1010101	0110011	0001111
1000000	1111100	0011010	0101001	0100110	0010101	1110011	1001111
0100000	1011100	1111010	1001001	1000110	1110101	0010011	0101111
0010000	0101100	1001010	1111001	1110110	1100101	0100011	0011111
0001000	0110100	1010010	1100001	1101110	1011101	0111011	0000111
0000100	0111000	1011110	1101101	1100010	1010001	0110111	0001011
0000010	0111110	1011000	1101011	1100100	1010111	0110001	0001101
0000001	0111101	1011011	1101000	1100111	1010100	0110010	0001110

FIGURE 5.2

CODE III

Error Pattern								
000000000	111101000	111010100	110110010	101110001	100111100	101011010	110011001	101100110
100000000	011101000	011010100	010110010	001110001	000111100	001011010	010011001	001100110
010000000	101101000	101010100	100110010	011110001	110111100	111011010	100011001	111100110
001000000	110101000	110010100	111110010	100110001	101111100	100011010	111011001	100100110
000100000	111001000	111110100	110010010	101010001	100011100	101111010	110111001	101000110
000010000	111111000	111000100	110100010	101100001	100101100	101001010	110001001	101110110
000001000	111100000	111011100	110111010	101111001	100110100	101010010	110010001	101101110
000000100	111101100	111010000	110110110	101110101	100111000	101011110	110011101	101100010
000000010	111101010	111010110	110110000	101110011	100111110	101011000	110011011	101100100
000000001	111101001	111010101	110110011	101110000	100111101	101011011	110011000	101100111

	110100101	111000011	010101010	010010110	110001110	101001101	100101011	100010111
	010100101	011000011	110101010	110010110	010001110	001001101	000101011	000010111
	100100101	101000011	000101010	000010110	100001110	111001101	110101011	110010111
	111100101	110000011	011101010	011010110	111001110	100001101	101101011	101010111
	110000101	111100011	010001010	010110110	110101110	101101101	100001011	100110111
	110110101	111010011	010111010	010000110	110011110	101011101	100111011	100000111
	110101101	111001011	010100010	010011110	110000110	101000101	100100011	100011111
	110100001	111000111	010101110	010010010	110001010	101001001	100101111	100010011
	110100111	111000001	010101000	010010100	110001100	101001111	100101001	100010101
	110100100	111000010	010101011	010010111	110001111	101001100	100101010	100010110

FIGURE 5.3

CODE IV

Error Pattern							
0000000000	1111010000	1110101000	1101100100	1011100010	0111100001	1110000011	1101001010
1000000000	0111010000	0110101000	0101100100	0011100010	1111100001	0110000011	0101001010
0100000000	1011010000	1010101000	1001100100	1111100010	0011100001	1010000011	1001001010
0010000000	1101010000	1100101000	1111100100	1001100010	0101100001	1100000011	1111001010
0001000000	1110010000	1111101000	1100100100	1010100010	0110100001	1111000011	1100001010
0000100000	1111110000	1110001000	1101000100	1011000010	0111000001	1110100011	1101101010
0000010000	1111000000	1110111000	1101110100	1011110010	0111110001	1110010011	1101011010
0000001000	1111011000	1110100000	1101101100	1011101010	0111101001	1110001011	1101000010
0000000100	1111010100	1110101100	1101100000	1011100110	0111100101	1110000111	1101001110
0000000010	1111010010	1110101010	1101100110	1011100000	0111100011	1110000001	1101001000
0000000001	1111010001	1110101001	1101100101	1011100011	0111100000	1110000010	1101001011

FIGURE 5.4

CODE IV (Cont'd. 2)

1100110001	1011000101	1010110100	1001101001	0111000110	0110110010	0101111000
0100110001	0011000101	0010110100	0001101001	1111000110	1110110010	1101111000
1000110001	1111000101	1110110100	1101101001	0011000110	0010110010	0001111000
1110110001	1001000101	1000110100	1011101001	0101000110	0100110010	0111111000
1101110001	1010000101	1011110100	1000101001	0110000110	0111110010	0100111000
1100010001	1011100101	1010010100	1001001001	0111100110	0110010010	0101011000
1100100001	1011010101	1010100100	1001111001	0111010110	0110100010	0101101000
1100111001	1011001101	1010111100	1001100001	0111001110	0110111010	0101110000
1100110101	1011000001	1010110000	1001101101	0111000010	0110110110	0101111100
1100110011	1011000111	1010110110	1001101011	0111000100	0110110000	0101111010
1100110000	1011000100	1010110101	1001101000	0111000111	0110110011	0101111001

FIGURE 5.4 (Cont'd.)

CODE IV (Cont'd. 3)

0011101100	1100011100	1010011010	1001010110	1000101110	0110011001
1011101100	0100011100	0010011010	0001010110	0000101110	1110011001
0111101100	1000011100	1110011010	1101010110	1100101110	0010011001
0001101100	1110011100	1000011010	1011010110	1010101110	0100011001
0010101100	1101011100	1011011010	1000010110	1001101110	0111011001
0011001100	1100111100	1010111010	1001110110	1000001110	0110111001
0011111100	1100001100	1010001010	1001000110	1000111110	0110001001
0011100100	1100010100	1010010010	1001011110	1000100110	0110010001
0011101000	1100011000	1010011110	1001010010	1000101010	0110011101
0011101110	1100011110	1010011000	1001010100	1000101100	0110011011
0011101101	1100011101	1010011011	1001010111	1000101111	0110011000

FIGURE 5.4 (Cont'd.)

CODE IV (Cont'd. 4)

0101010101	0100101101	0011010011	0010101011	0001100111	0000011111
1101010101	1100101101	1011010011	1010101011	1001100111	1000011111
0001010101	0000101101	0111010011	0110101011	0101100111	0100011111
0111010101	0110101101	0001010011	0000101011	0011100111	0010011111
0100010101	0101101101	0010010011	0011101011	0000100111	0001011111
0101110101	0100001101	0011110011	0010001011	0001000111	0000111111
0101000101	0100111101	0011000011	0010111011	0001110111	0000001111
0101011101	0100100101	0011011011	0010100011	0001101111	0000010111
0101010001	0100101001	0011010111	0010101111	0001100011	0000011011
0101010111	0100101111	0011010001	0010101001	0001100101	0000011101
0101010100	0100101100	0011010010	0010101010	0001100110	0000011110

FIGURE 5.4 (Cont'd.)

CODE V

Error Pattern								
0000000000	1111110001	1111100010	1110011010	1001111010	1010101011	0011011011	0110110011	1101001011
1000000000	0111110001	0111100000	0110011010	0001111010	0010101011	1011011011	1110110011	0101001011
0100000000	1011110001	1011100010	1010011010	1101111010	1110101011	0111011011	0010110011	1001001011
0010000000	1101110001	1101100010	1100011010	1011111010	1000101011	0001011011	0100110011	1111001011
0001000000	1110110001	1110100010	1111011010	1000111010	1011101011	0010011011	0111110011	1100001011
0000100000	1111010001	1111000010	1110111010	1001011010	1010001011	0011111011	0110010011	1101101011
0000010000	1111100001	1111110010	1110111010	1001101010	1010111011	0011001011	0110100011	1101011011
0000001000	1111111001	1111101010	1110001010	1001110010	1010100011	0011010011	0110111011	1101000011
0000000100	1111110101	1111100110	1110010010	1001111110	1010101111	0011011111	0110110111	1101001111
0000000010	1111110011	1111100000	1110011110	1001111000	1010101001	0011011001	0110110001	1101001001
0000000001	1111110000	1111100011	1110011011	1001111011	1010101010	0011011010	0110110010	1101001010

TABLE 5.5

CODE V (Continued)

1110000100	1001100100	0100110100	0011001100	1000011100	0100001101	0010100101	1000000110
0110000100	0001100100	1100110100	1011001100	0000011100	1100001101	1010100101	0000000110
1010000100	1101100100	0000110100	0111001100	1100011100	0000001101	0110100101	1100000110
1100000100	1011100100	0110110100	0001001100	1010011100	0110001101	0000100101	1010000110
1111000100	1000100100	0101110100	0010001100	1001011100	0101001101	0011100101	1001000110
1110100100	1001000100	0100010100	0011101100	1000111100	0100101101	0010000101	1000100110
1110010100	1001110100	0100100100	0011011100	1000001100	0100011101	0010110101	1000010110
1110001100	1001101100	0100111100	0011000100	1000010100	0100000101	0010101101	1000001110
1110000000	1001100000	0100110000	0011001000	1000011000	0100001001	0010100001	1000000010
1110000110	1001100110	0100110110	0011001110	1000011110	0100001111	0010100111	1000000100
1110000101	1001100101	0100110101	0011001101	1000011101	0100001100	0010100100	1000000111

BIBLIOGRAPHY

1. Berger, J.M. A Note on Error Detection Codes for Asymmetric Channels., Information and Control, Vol. 4, No. 1, (1961).

2. Birkhoff, G. and MacLane, S. Introduction to Modern Algebra., Macmillan, New York, (1948).

3. Freiman, C.V. Optimal Error Detection Codes for Completely Asymmetric Binary Channels., Information and Control, Vol. 5, No. 1, (1962).

4. Hamming, R.W. Error Detection and Error Correcting Codes., B.S.T.J., Vol. 29, pp. 147-160, (1950).

5. MacDonald, J.E.Jr. Constructive Coding Methods for the Binary Symmetric Independent Data Transmission Channel., M.S. Thesis, Department of Electrical Engineering Syacuse University, Syacuse, N.Y., (1958).

6. McCoy, N. Introduction to Modern Algebra., Allyn and Bacon, Inc., (1960).

7. Peterson, W.W. Error Correcting Codes., The M.I.T. Press and John Wiley and Sons Inc., (1961).

ADDITIONAL REFERENCES

8. Armstrong, D.B. — General Method of Applying Error Correction to Synchronous Digital Systems., B.S.T.J., Vol. 40, pp. 577-593, (1961).

9. Banerji, R.B. — On Constructing Group Codes., Information and Control, Vol. 4, No. 1, (1961).

10. Berger, J.M. — A Note on Burst Detecting Sum Codes., Information and Control, Vol. 4, No. 2-3, (1961).

11. Bose, R.C. and Kuebler, R.R.Jr. — A Geometry of Binary Sequences Associated with Group Alphabets in Information Theory., The Annals of Mathematical Statistics, Vol. 31, No. 1, (1960).

12. Bose, R.C. and Ray-Chandhuri, D.K. — On a Class of Error Correcting Binary Group Codes., Information and Control, Vol. 3, No. 1, (1960).

13. Bose, R.C. and Ray-Chandhuri, D.K. — Further Results on Error Correcting Binary Group Codes., Information and Control, Vol. 3, No. 3, (1960).

14. Bose, R.C. and Shrikhande, S.S. — A Note on a Result in the Theory of Code Construction., Information and Control, Vol. 2, No. 3, (1959).

15. Brown, D.T. and Peterson, W.W. — Cyclic Codes for Error Detection., Technical Publication, I.B.M., Poughkeepsie, N.Y., (1960).

16. Brown, D.T. and Peterson, W.W. — Cyclic Codes for Error Detection., Proceedings of the I.R.E., Vol. 49, No. 1, (1961).

17. Burt, M.W. and James, D.C. — How Much Does Redundancy Improve Reliability?, Control Engineering, p. 71, June, 1963.

18. (Cherry, Colin) Zierler, N — On Decoding Linear Error-Correcting Codes., Symposium on "Information Theory", Royal Institution, London, (1960).

19. Dimsdale, B. and Weinberg, G.M. — Programmed Error Correction in Project Mercury., Communications of the A.C.M., Vol. 3, No. 12, (1960).

20. Elias, P. — Predictive Coding-Part I., I.R.E. Transactions of the Professional Group on Information Theory, Vol. IT-1 No. 1, (1955).

21. Elias, P. — List Decoding for Noisy Channels., I.R.E. Wescon Convention Record, Part 2, (1957).

22. Elias, P. — Computation in the Presence of Noise., I.B.M. Journal, October, 1958.

23. Elias, P. and Feinstein, A. and Shannon, C. — A Note on the Maximum Flow Through a Network., I.R.E. Transactions of the Professional Group on Information Theory, Vol. IT-2, No. 4, (1956).

24. Epstein, M.A. — Algebraic Decoding for a Binary Erasure Channel., Technical Report 340, Research Laboratory of Electronics M.I.T., Cambridge, Mass., (1958).

25. Fire, P. — A Class of Multiple-Error-Correcting Binary Codes for Non-Independent Errors., Stanford Electronics Laboratories Technical Report No. 55 (1959).

26. Fontaine, A.B. and Gallager, R.G. — Error Statistics and Coding for Binary Transmission over Telephone Circuits., Proceedings of the I.R.E. November, 1960.

27. Fontaine, A.B. and Peterson, W.W. — On Coding for the Binary Symmetric Channel., Transactions Paper. A.I.E.E. No. 58-925, (1958).

28. Gilbert, E.N. — A Comparison of Signaling Alphabets., B.S.T.J., Vol. 31, pp 504-522, (1952)

29. Gilbert, E.N. and Moore, E.F. — Variable Length Binary Encoding., B.S.T.J., Vol. 38, pp. 933-968, (1959)

30. Goldstine, H.H. — Information Theory., Science, Vol. 133, No. 3462, pp. 1395-1399, (1961).

31. Gorenstein, D. and Peterson, W.W. and Zierler, N. Two-Error Correcting Bose-Chaudhuri Codes are Quasi-Perfect., Information and Control. 3. pp. 291-294, (1960).

32. Gorenstein, D. and Zierler, N. A Class of Error Correcting Codes in p^m Symbols., J. Soc. Indust. Appl. Math. Vol. 9, No. 2. June, 1961.

33. Hagelborger, D.W. Recurrent Codes: Easily- Mechanized, Burst-Correcting, Binary Codes., B.S.T.J., Vol. 38, pp. 969-984, (1959).

34. Hartley, R.V.L. Transmission of Information., reprint B-343, Bell Telephone Laboratories (1928).

35. Lerner, A.M. TDM Coding Factors., Instruments and Control Systems, p.121, March, 1963.

36. Levell, R.W. The Application of a Hamming Error Correcting Code to a Standard Teletype Equipment., J. Brit, I.R.E., Vol. 22, No. 5, (1961).

37. Lloyd, S.P. Binary Block Coding., B.S.T.J., Vol, 36, pp. 517-535, (1957).

38. MacDonald, J.E. Design Methods for Maximum Minimum-Distance Error-Correcting Codes., I.B.M. Journal, January, 1960.

39. MacWilliams, J. Error-Correcting Codes for Multiple-Level Transmission., B.S.T.J., Vol. 40, pp. 281-308, (1961).

40. McClusky, E.J.Jr. Error-Correcting Codes - a Linear Programming Approach., Bell Telephone System Technical Publications, Monograph 3399, (1959).

41. Meggitt, J.E. Auto-Programming for Numerically Controlled Machine Tools., Annual Review in Automatic Programming, Vol. I, (1960).

42. Meggitt, J.E. Error Correcting Codes for Bursts of Errors., Communication and Electronics, January, 1961.

43. Meggitt, J.E. Error Correction in Digital Data Transmission., Journal of the Institute of Electrical Engineers, Vol. 7, (1961).

44. Meggitt, J.E. Error Correcting Codes and Their Implementation for Data Transmission Systems., I.R.E. Transactions of the Professional Group on Information Theory, Vol. IT-7, No. 4, (1961).

45. Melas, C.M. A New Group of Codes for Correction of Dependent Errors in Data Transmission., I.B.M. Journal, Jan. 1960.

46. Melas, C.M. A Cyclic Code for Double Error Correction., I.B.M. Journal, Jan. 1960.

47. Neumann, P.G. A Note on Cyclic Permutation Error-Correcting Codes., Bell Telephone System Technical Publication, Monograph 4183, (1962).

48. Nyquist, H. Certain Topics in Telegraph Transmission Theory., reprint B-331; Bell Telephone Laboratories (1928).

49. Patterson, G.W. What is a Code? Communications of the A.C.M., Vol. 3, No. 5, (1960).

50. Peterson, W.W. An Experimental Study of a Binary Code., Engineering Progress at the University of Florida, Vol. XII, No. 11, (1958).

51. Peterson, W.W. Error-correcting Codes., Scientific American, Vol. 206, No. 2, (1962).

52. Root, W.L. Communications through Unspecified Additive Noise., Information and Control, Vol. 4, No. 1, (1961).

53. Rubinoff, M. N-Dimensional Codes for Detieting and Correcting Multiple Errors., Communications of the A.C.M., Vol. 4, No. 12, (1961).

54. Scherer, E.H. An Error-Correction Code for Quaternay Data Transmission., presented at A.I.E.E. Winter General Meeting, New York, (1961).

55. Shannon, C.E. A Mathematical Theory of Communication., B.S.T.J., Vol. 27, pp. 379-423, 623-656, (1948).

56. Shapero, H.S. and Slotnick, D.L. On the Mathematical Theory of Error-Correcting Codes., I.B.M. Journal, January, 1959.

57. Slepian, D. A Class of Binary Signaling Alphabets., Bell Telephone System Technical Publication, Monograph 2563, (1956).

58. Slepian, D. Some Further Theory of Group Codes., Bell Telephone System Technical Publication, Monograph 3693, (1960).

59. Stone, J.J. Multiple Burst Error Correction., Information and Control, Vol. 4, No. 4, (1961).

60. Ulrich, W. Non-binary Error Correction Codes., Bell Telephone System Technical Publications, Mongraph 2909, (1958).

61. Weinburg, G.M. Programmed Error Correction on a Decimal Computer., Communications of the A.C.M., Vol. 3, No. 12 (1960).

62. Wolfe, C.H. — 9 Criteria for Selecting a Digital Code., Control Engineering, p. 128, August, 1962.

63. Zierler, N. — Several Binary-Sequence Generators., M.I.T., Lincoln Laborator Technical Report, No. 95, (1955).

64. Zierler, N. — On a Class of Binary Sequences., Proceeding of the Am. Math. Society, Vol. 7, No. 4; pp. 675-681, (1956).

65. Zierler, N. — On a Variation of the First Order Reed-Muller Codes., Group Report 34-80, Lincoln Laboratory, M.I.T., (1958).

66. Zierler, N. — Linear Recurring Sequences., J. Soc. Indust. Appl. Math., Vol. 7, No. 1, (1959).

67. 17 papers. — 1954 Symposium on Information Theory., Transactions of the I.R.E., M.I.T. Cambridge, Mass. PGIT-4, Sept. 1954.

APPENDIX

PROBABILITIES OF CORRECT DECODING FOR CODE I

TABLE 5.6

q_1	q_2	p_1	p_2	Pr(A),Pr(S)	Pr(A)-Pr(S)
1.00	1.00	.00	.00	1.00000000	
1.00	.98	.00	.02	.99796875	
.99	.99	.01	.01	.99796887	-.00000012
.98	.98	.02	.02	.99214337	
1.00	.96	.00	.04	.99214031	
.98	.98	.02	.02	.99214337	-.00000306
.98	.96	.02	.04	.98290668	
.97	.97	.03	.03	.98290687	-.00000019
.96	.96	.04	.04	.97061956	
1.00	.94	.00	.06	.98289187	
.97	.97	.03	.03	.98290687	-.00001500
.98	.94	.02	.06	.97061675	
.96	.96	.04	.04	.97061956	-.00000281
.96	.94	.04	.06	.95561918	
.95	.95	.05	.05	.95561937	-.00000019
.94	.94	.06	.06	.93822281	
1.00	.92	.00	.08	.97057406	
.96	.96	.04	.04	.97061956	-.00004550
.98	.92	.02	.08	.95560556	
.95	.95	.05	.05	.95561937	-.00001381
.96	.92	.04	.08	.93822018	
.94	.94	.06	.06	.93822281	-.00000263
.94	.92	.06	.08	.91872581	
.93	.93	.07	.07	.91872600	-.00000019
.92	.92	.08	.08	.89740537	
1.00	.90	.00	.10	.95551281	
.95	.95	.05	.05	.95561937	-.00010656
.98	.90	.02	.10	.93818106	
.94	.94	.06	.06	.93822281	-.00004175
.96	.90	.04	.10	.91871331	
.93	.93	.07	.07	.91872600	-.00001269
.94	.90	.06	.10	.89740293	
.92	.92	.08	.08	.89740537	-.00000244
.92	.90	.08	.10	.87451843	
.91	.91	.09	.09	.87451856	-.00000013
.90	.90	.10	.10	.85030556	
1.00	.88	.00	.12	.93801125	
.94	.94	.06	.06	.93822281	-.00021156
.98	.88	.02	.12	.91862837	
.93	.93	.07	.07	.91872600	-.00009763
.96	.88	.04	.12	.89736712	
.92	.92	.08	.08	.89740537	-.00003825
.94	.88	.06	.12	.87450700	
.91	.91	.09	.09	.87451856	-.00001156
.92	.88	.08	.12	.85030337	
.90	.90	.10	.10	.85030556	-.00000219
.90	.88	.10	.12	.82498875	
.89	.89	.11	.11	.82498887	-.00000012
.88	.88	.12	.12	.79877500	
1.00	.86	.00	.14	.91835100	
.93	.93	.07	.07	.91872600	-.00037500
.98	.86	.02	.14	.89721181	
.92	.92	.08	.08	.89740537	-.00019356
.96	.86	.04	.14	.87442943	
.91	.91	.09	.09	.87451856	-.00008913
.94	.86	.06	.14	.85027068	
.90	.90	.10	.10	.85030556	-.00003488
.92	.86	.08	.14	.82497837	
.89	.89	.11	.11	.82498887	-.00001050
.90	.86	.10	.14	.79877300	
.88	.88	.12	.12	.79877500	-.00000200
.88	.86	.12	.14	.77185450	

.87	.87	.13	.13	.77185456	-.00000006
.86	.86	.14	.14	.74440362	
1.00	.84	.00	.16	.89679381	
.92	.92	.08	.08	.89740537	-.00061156
.98	.84	.02	.16	.87417618	
.91	.91	.09	.09	.87451856	-.00034238
.96	.84	.04	.16	.85012918	
.90	.90	.10	.10	.85030556	-.00017638
.94	.84	.06	.16	.82490775	
.89	.89	.11	.11	.82498887	-.00008112
.92	.84	.08	.16	.79874331	
.88	.88	.12	.12	.79877500	-.00003169
.90	.84	.10	.16	.77184506	
.87	.87	.13	.13	.77185456	-.00000950
.88	.84	.12	.16	.74440187	
.86	.86	.14	.14	.74440362	-.00000175
.86	.84	.14	.16	.71658387	
.85	.85	.15	.15	.71658406	-.00000019
.84	.84	.16	.16	.68854412	
1.00	.82	.00	.18	.87358300	
.91	.91	.09	.09	.87451856	-.00093556
.98	.82	.02	.18	.84974812	
.90	.90	.10	.10	.85030556	-.00055744
.96	.82	.04	.18	.82467737	
.89	.89	.11	.11	.82498887	-.00031150
.94	.82	.06	.18	.79861475	
.88	.88	.12	.12	.79877500	-.00016025
.92	.82	.08	.18	.77178106	
.87	.87	.13	.13	.77185456	-.00007350
.90	.82	.10	.18	.74437500	
.86	.86	.14	.14	.74440362	-.00002862
.88	.82	.12	.18	.71657543	
.85	.85	.15	.15	.71658406	-.00000863
.86	.82	.14	.18	.68854243	
.84	.84	.16	.16	.68854412	-.00000169
.84	.82	.16	.18	.66041943	
.83	.83	.17	.17	.66041950	-.00000007
.82	.82	.18	.18	.63233381	
1.00	.80	.00	.20	.84894475	
.90	.90	.10	.10	.85030556	-.00136081
.98	.80	.02	.20	.82413762	
.89	.89	.11	.11	.82498887	-.00085125
.96	.80	.04	.20	.79826868	
.88	.88	.12	.12	.79877500	-.00050632
.94	.80	.06	.20	.77157212	
.87	.87	.13	.13	.77185456	-.00028244
.92	.80	.08	.20	.74425875	
.86	.86	.14	.14	.74440362	-.00014487
.90	.80	.10	.20	.71651762	
.85	.85	.15	.15	.71658406	-.00006644
.88	.80	.12	.20	.68851831	
.84	.84	.16	.16	.68854412	-.00002581
.86	.80	.14	.20	.66041181	
.83	.83	.17	.17	.66041950	-.00000769
.84	.80	.16	.20	.63233237	
.82	.82	.18	.18	.63233381	-.00000144
.82	.80	.18	.20	.60439911	
.81	.81	.19	.19	.60439921	-.00000010
.80	.80	.20	.20	.57671680	
1.00	.78	.00	.22	.82308931	
.89	.89	.11	.11	.82498887	-.00189956
.98	.78	.02	.22	.79753900	

.88	.88	.12	.12	.79877500	-.00123600
.96	.78	.04	.22	.77108287	
.87	.87	.13	.13	.77185456	-.00077169
.94	.78	.06	.22	.74394556	
.86	.86	.14	.14	.74440362	-.00045806
.92	.78	.08	.22	.71632900	
.85	.85	.15	.15	.71658406	-.00025506
.90	.78	.10	.22	.68841350	
.84	.84	.16	.16	.68854412	-.00013062
.88	.78	.12	.22	.66035981	
.83	.83	.17	.17	.66041950	-.00005969
.86	.78	.14	.22	.63231068	
.82	.82	.18	.18	.63233381	-.00002313
.84	.78	.16	.22	.60439228	
.81	.81	.19	.19	.60439921	-.00000693
.82	.78	.18	.22	.57671550	
.80	.80	.20	.20	.57671680	-.00000130
.80	.78	.20	.22	.54937756	
.79	.79	.21	.21	.54937765	-.00000009
.78	.78	.22	.22	.52246307	
1.00	.76	.00	.24	.79621212	
.88	.88	.12	.12	.79877500	-.00256288
.98	.76	.02	.24	.77013243	
.87	.87	.13	.13	.77185456	-.00172213
.96	.76	.04	.24	.74328537	
.86	.86	.14	.14	.74440362	-.00111825
.94	.76	.06	.24	.71588718	
.85	.85	.15	.15	.71658406	-.00069688
.92	.76	.08	.24	.68813137	
.84	.84	.16	.16	.68854412	-.00041275
.90	.76	.10	.24	.66019025	
.83	.83	.17	.17	.66041950	-.00022925
.88	.76	.12	.24	.63221675	
.82	.82	.18	.18	.63233381	-.00011706
.86	.76	.14	.24	.60434581	
.81	.81	.19	.19	.60439921	-.00005340
.84	.76	.16	.24	.57669615	
.80	.80	.20	.20	.57671680	-.00002065
.82	.76	.18	.24	.54937148	
.79	.79	.21	.21	.54937765	-.00000617
.80	.76	.20	.24	.52246193	
.78	.78	.22	.22	.52246307	-.00000114
.78	.76	.22	.24	.49604519	
.77	.77	.23	.23	.49604525	-.00000006
.76	.76	.24	.24	.47018782	
1.00	.74	.00	.26	.76849518	
.87	.87	.13	.13	.77185456	-.00335938
.98	.74	.02	.26	.74208481	
.86	.86	.14	.14	.74440362	-.00231881
.96	.74	.04	.26	.71502900	
.85	.85	.15	.15	.71658406	-.00155506
.94	.74	.06	.26	.68753643	
.84	.84	.16	.16	.68854412	-.00100769
.92	.74	.08	.26	.65979306	
.83	.83	.17	.17	.66041950	-.00062644
.90	.74	.10	.26	.63196368	
.82	.82	.18	.18	.63233381	-.00037013
.88	.74	.12	.26	.60419410	
.81	.81	.19	.19	.60439921	-.00020511
.86	.74	.14	.26	.57661228	
.80	.80	.20	.20	.57671680	-.00010452
.84	.74	.16	.26	.54933013	

.79	.79	.21	.21	.54937765	-.00004752
.82	.74	.18	.26	.52244475	
.78	.78	.22	.22	.52246307	-.00001832
.80	.74	.20	.26	.49603981	
.77	.77	.23	.23	.49604525	-.00000544
.78	.74	.22	.26	.47018681	
.76	.76	.24	.24	.47018782	-.00000101
.76	.74	.24	.26	.44494622	
.75	.75	.25	.25	.44494630	-.00000008
.74	.74	.26	.26	.42036861	
1.00	.72	.00	.28	.74010768	
.86	.86	.14	.14	.74440362	-.00429594
.98	.72	.02	.28	.71355062	
.85	.85	.15	.15	.71658406	-.00303344
.96	.72	.04	.28	.68645475	
.84	.84	.16	.16	.68854412	-.00208937
.94	.72	.06	.28	.65902156	
.83	.83	.17	.17	.66041950	-.00139794
.92	.72	.08	.28	.63143012	
.82	.82	.18	.18	.63233381	-.00090369
.90	.72	.10	.28	.60383874	
.81	.81	.19	.19	.60439921	-.00056047
.88	.72	.12	.28	.57638649	
.80	.80	.20	.20	.57671680	-.00033031
.86	.72	.14	.28	.54919512	
.79	.79	.21	.21	.54937765	-.00018253
.84	.72	.16	.28	.52237033	
.78	.78	.22	.22	.52246307	-.00009274
.82	.72	.18	.28	.49600323	
.77	.77	.23	.23	.49604525	-.00004202
.80	.72	.20	.28	.47017168	
.76	.76	.24	.24	.47018782	-.00001614
.78	.72	.22	.28	.44494150	
.75	.75	.25	.25	.44494630	-.00000480
.76	.72	.24	.28	.42036772	
.74	.74	.26	.26	.42036861	-.00000089
.74	.72	.26	.28	.39649561	
.73	.73	.27	.27	.39649566	-.00000005
.72	.72	.28	.28	.37336171	
1.00	.70	.00	.30	.71120731	
.85	.85	.15	.15	.71658406	-.00537675
.98	.70	.02	.30	.68467331	
.84	.84	.16	.16	.68854412	-.00387081
.96	.70	.04	.30	.65769250	
.83	.83	.17	.17	.66041950	-.00272700
.94	.70	.06	.30	.63045993	
.82	.82	.18	.18	.63233381	-.00187388
.92	.70	.08	.30	.60314850	
.81	.81	.19	.19	.60439921	-.00125071
.90	.70	.10	.30	.57591040	
.80	.80	.20	.20	.57671680	-.00080640
.88	.70	.12	.30	.54887891	
.79	.79	.21	.21	.54937765	-.00049874
.86	.70	.14	.30	.52217001	
.78	.78	.22	.22	.52246307	-.00029306
.84	.70	.16	.30	.49588383	
.77	.77	.23	.23	.49604525	-.00016142
.82	.70	.18	.30	.47010607	
.76	.76	.24	.24	.47018782	-.00008175
.80	.70	.20	.30	.44490937	
.75	.75	.25	.25	.44494630	-.00003693
.78	.70	.22	.30	.42035448	

.74	.74	.26	.26	.42036861	-.00001413
.76	.70	.24	.30	.39649149	
.73	.73	.27	.27	.39649566	-.00000417
.74	.70	.26	.30	.37336094	
.72	.72	.28	.28	.37336171	-.00000077
.72	.70	.28	.30	.35099473	
.71	.71	.29	.29	.35099476	-.00000003
.70	.70	.30	.30	.32941720	
1.00	.68	.00	.32	.68194075	
.84	.84	.16	.16	.68854412	-.00660337
.98	.68	.02	.32	.65558575	
.83	.83	.17	.17	.66041950	-.00483375
.96	.68	.04	.32	.62886218	
.82	.82	.18	.18	.63233381	-.00347163
.94	.68	.06	.32	.60195940	
.81	.81	.19	.19	.60439921	-.00243981
.92	.68	.08	.32	.57504464	
.80	.80	.20	.20	.57671680	-.00167216
.90	.68	.10	.32	.54826470	
.79	.79	.21	.21	.54937765	-.00111295
.88	.68	.12	.32	.52174758	
.78	.78	.22	.22	.52246307	-.00071549
.86	.68	.14	.32	.49560413	
.77	.77	.23	.23	.49604525	-.00044112
.84	.68	.16	.32	.46992947	
.76	.76	.24	.24	.47018782	-.00025835
.82	.68	.18	.32	.44480447	
.75	.75	.25	.25	.44494630	-.00014183
.80	.68	.20	.32	.42029708	
.74	.74	.26	.26	.42036861	-.00007153
.78	.68	.22	.32	.39646349	
.73	.73	.27	.27	.39649566	-.00003217
.76	.68	.24	.32	.37334943	
.72	.72	.28	.28	.37336171	-.00001228
.74	.68	.26	.32	.35099117	
.71	.71	.29	.29	.35099476	-.00000359
.72	.68	.28	.32	.32941653	
.70	.70	.30	.30	.32941720	-.00000067
.70	.68	.30	.32	.30864590	
.69	.69	.31	.31	.30864593	-.00000003
.68	.68	.32	.32	.28869304	
1.00	.66	.00	.34	.65244481	
.83	.83	.17	.17	.66041950	-.00797469
.98	.66	.02	.34	.62641131	
.82	.82	.18	.18	.63233381	-.00592250
.96	.66	.04	.34	.60007460	
.81	.81	.19	.19	.60439921	-.00432461
.94	.66	.06	.34	.57361891	
.80	.80	.20	.20	.57671680	-.00309789
.92	.66	.08	.34	.54720656	
.79	.79	.21	.21	.54937765	-.00217109
.90	.66	.10	.34	.52097945	
.78	.78	.22	.22	.52246307	-.00148362
.88	.66	.12	.34	.49506086	
.77	.77	.23	.23	.49604525	-.00098439
.86	.66	.14	.34	.46955708	
.76	.76	.24	.24	.47018782	-.00063074
.84	.66	.16	.34	.44455877	
.75	.75	.25	.25	.44494630	-.00038753
.82	.66	.18	.34	.42014251	
.74	.74	.26	.26	.42036861	-.00022610
.80	.66	.20	.34	.39637206	

.73	.73	.27	.27	.39649566	-.00012360
.78	.66	.22	.34	.37329963	
.72	.72	.28	.28	.37336171	-.00006208
.76	.66	.24	.34	.35096698	
.71	.71	.29	.29	.35099476	-.00002778
.74	.66	.26	.34	.32940666	
.70	.70	.30	.30	.32941720	-.00001054
.72	.66	.28	.34	.30864286	
.69	.69	.31	.31	.30864593	-.00000307
.70	.66	.30	.34	.28869248	
.68	.68	.32	.32	.28869304	-.00000056
.68	.66	.32	.34	.26956594	
.67	.67	.33	.33	.26956595	-.00000001
.66	.66	.34	.34	.25126800	
1.00	.64	.00	.36	.62284713	
.82	.82	.18	.18	.63233381	-.00948668
.98	.64	.02	.36	.59726448	
.81	.81	.19	.19	.60439921	-.00713473
.96	.64	.04	.36	.57143196	
.80	.80	.20	.20	.57671680	-.00528484
.94	.64	.06	.36	.54552936	
.79	.79	.21	.21	.54937765	-.00384829
.92	.64	.08	.36	.51971448	
.78	.78	.22	.22	.52246307	-.00274859
.90	.64	.10	.36	.49412496	
.77	.77	.23	.23	.49604525	-.00192029
.88	.64	.12	.36	.46887991	
.76	.76	.24	.24	.47018782	-.00130791
.86	.64	.14	.36	.44408154	
.75	.75	.25	.25	.44494630	-.00086476
.84	.64	.16	.36	.41981663	
.74	.74	.26	.26	.42036861	-.00055198
.82	.64	.18	.36	.39615792	
.73	.73	.27	.27	.39649566	-.00033774
.80	.64	.20	.36	.37316550	
.72	.72	.28	.28	.37336171	-.00019621
.78	.64	.22	.36	.35088800	
.71	.71	.29	.29	.35099476	-.00010676
.76	.64	.24	.36	.32936385	
.70	.70	.30	.30	.32941720	-.00005335
.74	.64	.26	.36	.30862220	
.69	.69	.31	.31	.30864593	-.00002373
.72	.64	.28	.36	.28868409	
.68	.68	.32	.32	.28869304	-.00000895
.70	.64	.30	.36	.26956337	
.67	.67	.33	.33	.26956595	-.00000258
.68	.64	.32	.36	.25126752	
.66	.66	.34	.34	.25126800	-.00000048
.66	.64	.34	.36	.23379853	
.65	.65	.35	.35	.23379855	-.00000002
.64	.64	.36	.36	.21715353	
1.00	.62	.00	.38	.59326663	
.81	.81	.19	.19	.60439921	-.01113258
.98	.62	.02	.38	.56825151	
.80	.80	.20	.20	.57671680	-.00846529
.96	.62	.04	.38	.54302875	
.79	.79	.21	.21	.54937765	-.00634890
.94	.62	.06	.38	.51777410	
.78	.78	.22	.22	.52246307	-.00468897
.92	.62	.08	.38	.49264149	
.77	.77	.23	.23	.49604525	-.00340376
.90	.62	.10	.38	.46776476	

.76	.76	.24	.24	.47018782	-.00242306
.88	.62	.12	.38	.44325940	
.75	.75	.25	.25	.44494630	-.00168690
.86	.62	.14	.38	.41922401	
.74	.74	.26	.26	.42036861	-.00114460
.84	.62	.16	.38	.39574198	
.73	.73	.27	.27	.39649566	-.00075368
.82	.62	.18	.38	.37288269	
.72	.72	.28	.28	.37336171	-.00047902
.80	.62	.20	.38	.35070305	
.71	.71	.29	.29	.35099476	-.00029171
.78	.62	.22	.38	.32924859	
.70	.70	.30	.30	.32941720	-.00016861
.76	.62	.24	.38	.30855471	
.69	.69	.31	.31	.30864593	-.00009122
.74	.62	.26	.38	.28864773	
.68	.68	.32	.32	.28869304	-.00004531
.72	.62	.28	.38	.26954593	
.67	.67	.33	.33	.26956595	-.00002002
.70	.62	.30	.38	.25126050	
.66	.66	.34	.34	.25126800	-.00000750
.68	.62	.32	.38	.23379639	
.65	.65	.35	.35	.23379855	-.00000216
.66	.62	.34	.38	.21715315	
.64	.64	.36	.36	.21715353	-.00000038
.64	.62	.36	.38	.20132564	
.63	.63	.37	.37	.20132565	-.00000001
.62	.62	.38	.38	.18630475	
1.00	.60	.00	.40	.56381440	
.80	.80	.20	.20	.57671680	-.01290240
.98	.60	.02	.40	.53947124	
.79	.79	.21	.21	.54937765	-.00990641
.96	.60	.04	.40	.51495226	
.78	.78	.22	.22	.52246307	-.00751081
.94	.60	.06	.40	.49042974	
.77	.77	.23	.23	.49604525	-.00561551
.92	.60	.08	.40	.46605421	
.76	.76	.24	.24	.47018782	-.00413361
.90	.60	.10	.40	.44195625	
.75	.75	.25	.25	.44494630	-.00299005
.88	.60	.12	.40	.41824811	
.74	.74	.26	.26	.42036861	-.00212050
.86	.60	.14	.40	.39502540	
.73	.73	.27	.27	.39649566	-.00147026
.84	.60	.16	.40	.37236844	
.72	.72	.28	.28	.37336171	-.00099327
.82	.60	.18	.40	.35034381	
.71	.71	.29	.29	.35099476	-.00065095
.80	.60	.20	.40	.32900560	
.70	.70	.30	.30	.32941720	-.00041160
.78	.60	.22	.40	.30839666	
.69	.69	.31	.31	.30864593	-.00024927
.76	.60	.24	.40	.28854986	
.68	.68	.32	.32	.28869304	-.00014318
.74	.60	.26	.40	.26948901	
.67	.67	.33	.33	.26956595	-.00007694
.72	.60	.28	.40	.25123006	
.66	.66	.34	.34	.25126800	-.00003794
.70	.60	.30	.40	.23378192	
.65	.65	.35	.35	.23379855	-.00001663
.68	.60	.32	.40	.21714737	
.64	.64	.36	.36	.21715353	-.00000616

.66	.60	.34	.40	.20132391	
.63	.63	.37	.37	.20132565	-.00000174
.64	.60	.36	.40	.18630445	
.62	.62	.38	.38	.18630475	-.00000030
.62	.60	.38	.40	.17207802	
.61	.61	.39	.39	.17207804	-.00000002
.60	.60	.40	.40	.15863040	
1.00	.58	.00	.42	.53459410	
.79	.79	.21	.21	.54937765	-.01478355
.98	.58	.02	.42	.51101541	
.78	.78	.22	.22	.52246307	-.01144766
.96	.58	.04	.42	.48728314	
.77	.77	.23	.23	.49604525	-.00876211
.94	.58	.06	.42	.46356656	
.76	.76	.24	.24	.47018782	-.00662126
.92	.58	.08	.42	.44001332	
.75	.75	.25	.25	.44494630	-.00493298
.90	.58	.10	.42	.41675115	
.74	.74	.26	.26	.42036861	-.00361746
.88	.58	.12	.42	.39388958	
.73	.73	.27	.27	.39649566	-.00260608
.86	.58	.14	.42	.37152157	
.72	.72	.28	.28	.37336171	-.00184014
.84	.58	.16	.42	.34972490	
.71	.71	.29	.29	.35099476	-.00126986
.82	.58	.18	.42	.32856370	
.70	.70	.30	.30	.32941720	-.00085350
.80	.58	.20	.42	.30808969	
.69	.69	.31	.31	.30864593	-.00055624
.78	.58	.22	.42	.28834346	
.68	.68	.32	.32	.28869304	-.00034958
.76	.58	.24	.42	.26935568	
.67	.67	.33	.33	.26956595	-.00021027
.74	.58	.26	.42	.25114810	
.66	.66	.34	.34	.25126800	-.00011990
.72	.58	.28	.42	.23373465	
.65	.65	.35	.35	.23379855	-.00006390
.70	.58	.30	.42	.21712233	
.64	.64	.36	.36	.21715353	-.00003120
.68	.58	.32	.42	.20131213	
.63	.63	.37	.37	.20132565	-.00001352
.66	.58	.34	.42	.18629980	
.62	.62	.38	.38	.18630475	-.00000495
.64	.58	.36	.42	.17207665	
.61	.61	.39	.39	.17207804	-.00000139
.62	.58	.38	.42	.15863015	
.60	.60	.40	.40	.15863040	-.00000025
.60	.58	.40	.42	.14594462	
.59	.59	.41	.41	.14594463	-.00000001
.58	.58	.42	.42	.13400178	
1.00	.56	.00	.44	.50570255	
.78	.78	.22	.22	.52246307	-.01676052
.98	.56	.02	.44	.48296935	
.77	.77	.23	.23	.49604525	-.01307590
.96	.56	.04	.44	.46009600	
.76	.76	.24	.24	.47018782	-.01009182
.94	.56	.06	.44	.43724918	
.75	.75	.25	.25	.44494630	-.00769712
.92	.56	.08	.44	.41457413	
.74	.74	.26	.26	.42036861	-.00579448
.90	.56	.10	.44	.39219616	
.73	.73	.27	.27	.39649566	-.00429950

.88	.56	.12	.44	.37022251	
.72	.72	.28	.28	.37336171	-.00313920
.86	.56	.14	.44	.34874390	
.71	.71	.29	.29	.35099476	-.00225086
.84	.56	.16	.44	.32783598	
.70	.70	.30	.30	.32941720	-.00158122
.82	.56	.18	.44	.30756082	
.69	.69	.31	.31	.30864593	-.00108511
.80	.56	.20	.44	.28796816	
.68	.68	.32	.32	.28869304	-.00072488
.78	.56	.22	.44	.26909670	
.67	.67	.33	.33	.26956595	-.00046925
.76	.56	.24	.44	.25097527	
.66	.66	.34	.34	.25126800	-.00029273
.74	.56	.26	.44	.23362391	
.65	.65	.35	.35	.23379855	-.00017464
.72	.56	.28	.44	.21705488	
.64	.64	.36	.36	.21715353	-.00009865
.70	.56	.30	.44	.20127363	
.63	.63	.37	.37	.20132565	-.00005202
.68	.56	.32	.44	.18627965	
.62	.62	.38	.38	.18630475	-.00002510
.66	.56	.34	.44	.17206730	
.61	.61	.39	.39	.17207804	-.00001074
.64	.56	.36	.44	.15862651	
.60	.60	.40	.40	.15863040	-.00000389
.62	.56	.38	.44	.14594356	
.59	.59	.41	.41	.14594463	-.00000107
.60	.56	.40	.44	.13400161	
.58	.58	.42	.42	.13400178	-.00000017
.58	.56	.42	.44	.12278126	
.57	.57	.43	.43	.12278127	-.00000001
.56	.56	.44	.44	.11226115	
1.00	.54	.00	.46	.47723018	
.77	.77	.23	.23	.49604525	-.01881507
.98	.54	.02	.46	.45541237	
.76	.76	.24	.24	.47018782	-.01477545
.96	.54	.04	.46	.43345974	
.75	.75	.25	.25	.44494630	-.01148656
.94	.54	.06	.46	.41153691	
.74	.74	.26	.26	.42036861	-.00883170
.92	.54	.08	.46	.38978698	
.73	.73	.27	.27	.39649566	-.00670868
.90	.54	.10	.46	.36833333	
.72	.72	.28	.28	.37336171	-.00502838
.88	.54	.12	.46	.34728129	
.71	.71	.29	.29	.35099476	-.00371347
.86	.54	.14	.46	.32671972	
.70	.70	.30	.30	.32941720	-.00269748
.84	.54	.16	.46	.30672255	
.69	.69	.31	.31	.30864593	-.00192338
.82	.54	.18	.46	.28735011	
.68	.68	.32	.32	.28869304	-.00134293
.80	.54	.20	.46	.26865055	
.67	.67	.33	.33	.26956595	-.00091540
.78	.54	.22	.46	.25066101	
.66	.66	.34	.34	.25126800	-.00060699
.76	.54	.24	.46	.23340885	
.65	.65	.35	.35	.23379855	-.00038970
.74	.54	.26	.46	.21691268	
.64	.64	.36	.36	.21715353	-.00024085
.72	.54	.28	.46	.20118348	

.63	.63	.37	.37	.20132565	-.00014217
.70	.54	.30	.46	.18622540	
.62	.62	.38	.38	.18630475	-.00007935
.68	.54	.32	.46	.17203676	
.61	.61	.39	.39	.17207804	-.00004128
.66	.54	.34	.46	.15861080	
.60	.60	.40	.40	.15863040	-.00001960
.64	.54	.36	.46	.14593641	
.59	.59	.41	.41	.14594463	-.00000822
.62	.54	.38	.46	.13399889	
.58	.58	.42	.42	.13400178	-.00000289
.60	.54	.40	.46	.12278050	
.57	.57	.43	.43	.12278127	-.00000077
.58	.54	.42	.46	.11226103	
.56	.56	.44	.44	.11226115	-.00000012
.56	.54	.44	.46	.10241835	
.55	.55	.45	.45	.10241836	-.00000001
.54	.54	.46	.46	.09322885	
1.00	.52	.00	.48	.44926140	
.76	.76	.24	.24	.47018782	-.02092642
.98	.52	.02	.48	.42841817	
.75	.75	.25	.25	.44494630	-.01652813
.96	.52	.04	.48	.40743811	
.74	.74	.26	.26	.42036861	-.01293050
.94	.52	.06	.48	.38648415	
.73	.73	.27	.27	.39649566	-.01001151
.92	.52	.08	.48	.36569768	
.72	.72	.28	.28	.37336171	-.00766403
.90	.52	.10	.48	.34520048	
.71	.71	.29	.29	.35099476	-.00579428
.88	.52	.12	.48	.32509637	
.70	.70	.30	.30	.32941720	-.00432083
.86	.52	.14	.48	.30547274	
.69	.69	.31	.31	.30864593	-.00317319
.84	.52	.16	.48	.28640206	
.68	.68	.32	.32	.28869304	-.00229098
.82	.52	.18	.48	.26794336	
.67	.67	.33	.33	.26956595	-.00162259
.80	.52	.20	.48	.25014347	
.66	.66	.34	.34	.25126800	-.00112453
.78	.52	.22	.48	.23303832	
.65	.65	.35	.35	.23379855	-.00076023
.76	.52	.24	.48	.21665411	
.64	.64	.36	.36	.21715353	-.00049942
.74	.52	.26	.48	.20100838	
.63	.63	.37	.37	.20132565	-.00031727
.72	.52	.28	.48	.18611103	
.62	.62	.38	.38	.18630475	-.00019372
.70	.52	.30	.48	.17196525	
.61	.61	.39	.39	.17207804	-.00011279
.68	.52	.32	.48	.15856845	
.60	.60	.40	.40	.15863040	-.00006195
.66	.52	.34	.48	.14591304	
.59	.59	.41	.41	.14594463	-.00003159
.64	.52	.36	.48	.13398713	
.58	.58	.42	.42	.13400178	-.00001465
.62	.52	.38	.48	.12277530	
.57	.57	.43	.43	.12278127	-.00000597
.60	.52	.40	.48	.11225913	
.56	.56	.44	.44	.11226115	-.00000202
.58	.52	.42	.48	.10241785	
.55	.55	.45	.45	.10241836	-.00000051

.56	.52	.44	.48	.09322878	
.54	.54	.46	.46	.09322885	-.00000007
.54	.52	.46	.48	.08466785	
.53	.53	.47	.47	.08466785	.00000000
.52	.52	.48	.48	.07670995	
1.00	.50	.00	.50	.42187500	
.75	.75	.25	.25	.44494630	-.02307130
.98	.50	.02	.50	.40205520	
.74	.74	.26	.26	.42036861	-.01831341
.96	.50	.04	.50	.38208996	
.73	.73	.27	.27	.39649566	-.01440570
.94	.50	.06	.50	.36214080	
.72	.72	.28	.28	.37336171	-.01122091
.92	.50	.08	.50	.34234783	
.71	.71	.29	.29	.35099476	-.00864693
.90	.50	.10	.50	.32283160	
.70	.70	.30	.30	.32941720	-.00658560
.88	.50	.12	.50	.30369468	
.69	.69	.31	.31	.30864593	-.00495125
.86	.50	.14	.50	.28502335	
.68	.68	.32	.32	.28869304	-.00366969
.84	.50	.16	.50	.26688900	
.67	.67	.33	.33	.26956595	-.00267695
.82	.50	.18	.50	.24934961	
.66	.66	.34	.34	.25126800	-.00191839
.80	.50	.20	.50	.23245105	
.65	.65	.35	.35	.23379855	-.00134750
.78	.50	.22	.50	.21622831	
.64	.64	.36	.36	.21715353	-.00092522
.76	.50	.24	.50	.20070673	
.63	.63	.37	.37	.20132565	-.00061892
.74	.50	.26	.50	.18590303	
.62	.62	.38	.38	.18630475	-.00040172
.72	.50	.28	.50	.17182635	
.61	.61	.39	.39	.17207804	-.00025169
.70	.50	.30	.50	.15847920	
.60	.60	.40	.40	.15863040	-.00015120
.68	.50	.32	.50	.14585831	
.59	.59	.41	.41	.14594463	-.00008632
.66	.50	.34	.50	.13395550	
.58	.58	.42	.42	.13400178	-.00004628
.64	.50	.36	.50	.12275834	
.57	.57	.43	.43	.12278127	-.00002293
.62	.50	.38	.50	.11225091	
.56	.56	.44	.44	.11226115	-.00001024
.60	.50	.40	.50	.10241440	
.55	.55	.45	.45	.10241836	-.00000396
.58	.50	.42	.50	.09322761	
.54	.54	.46	.46	.09322885	-.00000124
.56	.50	.44	.50	.08466756	
.53	.53	.47	.47	.08466785	-.00000028
.54	.50	.46	.50	.07670992	
.52	.52	.48	.48	.07670995	-.00000003
.52	.50	.48	.50	.06932936	
.51	.51	.49	.49	.06932936	.00000000
.50	.50	.50	.50	.06250000	

STOP

PROBABILITIES OF CORRECT DECODING FOR CODE II

TABLE 5.7

q_1	q_2	p_1	p_2	Pr(A),Pr(S)	Pr(A)-Pr(S)
1.00	1.00	.00	.00	1.00000000	
1.00	.98	.00	.02	.99881600	
.99	.99	.01	.01	.99796895	.00084705
.98	.98	.02	.02	.99214346	
1.00	.96	.00	.04	.99532800	
.98	.98	.02	.02	.99214346	.00318454
.98	.96	.02	.04	.98467537	
.97	.97	.03	.03	.98290696	.00176841
.96	.96	.04	.04	.97061966	
1.00	.94	.00	.06	.98963200	
.97	.97	.03	.03	.98290696	.00672504
.98	.94	.02	.06	.97534417	
.96	.96	.04	.04	.97061966	.00472451
.96	.94	.04	.06	.95811614	
.95	.95	.05	.05	.95561947	.00249667
.94	.94	.06	.06	.93822290	
1.00	.92	.00	.08	.98182400	
.96	.96	.04	.04	.97061966	.01120434
.98	.92	.02	.08	.96423479	
.95	.95	.05	.05	.95561947	.00861532
.96	.92	.04	.08	.94413816	
.94	.94	.06	.06	.93822290	.00591526
.94	.92	.06	.08	.92178186	
.93	.93	.07	.07	.91872606	.00305580
.92	.92	.08	.08	.89740540	
1.00	.90	.00	.10	.97200000	
.95	.95	.05	.05	.95561947	.01638053
.98	.90	.02	.10	.95143216	
.94	.94	.06	.06	.93822290	.01320926
.96	.90	.04	.10	.92876047	
.93	.93	.07	.07	.91872606	.01003441
.94	.90	.06	.10	.90420697	
.92	.92	.08	.08	.89740540	.00680157
.92	.90	.08	.10	.87798630	
.91	.91	.09	.09	.87451864	.00346766
.90	.90	.10	.10	.85030560	
1.00	.88	.00	.12	.96025600	
.94	.94	.06	.06	.93822290	.02203310
.98	.88	.02	.12	.93702124	
.93	.93	.07	.07	.91872606	.01829518
.96	.88	.04	.12	.91205778	
.92	.92	.08	.08	.89740540	.01465238
.94	.88	.06	.12	.88556362	
.91	.91	.09	.09	.87451864	.01104498
.92	.88	.08	.12	.85772994	
.90	.90	.10	.10	.85030560	.00742434
.90	.88	.10	.12	.82874120	
.89	.89	.11	.11	.82498895	.00375225
.88	.88	.12	.12	.79877503	
1.00	.86	.00	.14	.94668800	
.93	.93	.07	.07	.91872606	.02796194
.98	.86	.02	.14	.92108693	
.92	.92	.08	.08	.89740540	.02368153
.96	.86	.04	.14	.89410486	
.91	.91	.09	.09	.87451864	.01958622
.94	.86	.06	.14	.86591717	
.90	.90	.10	.10	.85030560	.01561157
.92	.86	.08	.14	.83669316	
.89	.89	.11	.11	.82498895	.01170421

.90	.86	.10	.14	.80659593	
.88	.88	.12	.12	.79877503	.00782090
.88	.86	.12	.14	.77578249	
.87	.87	.13	.13	.77185463	.00392786
.86	.86	.14	.14	.74440371	
1.00	.84	.00	.16	.93139200	
.92	.92	.08	.08	.89740540	.03398660
.98	.84	.02	.16	.90371419	
.91	.91	.09	.09	.87451864	.02919555
.96	.84	.04	.16	.87497644	
.90	.90	.10	.10	.85030560	.02467084
.94	.84	.06	.16	.84533304	
.89	.89	.11	.11	.82498895	.02034409
.92	.84	.08	.16	.81493275	
.88	.88	.12	.12	.79877503	.01615772
.90	.84	.10	.16	.78391878	
.87	.87	.13	.13	.77185463	.01206415
.88	.84	.12	.16	.75242883	
.86	.86	.14	.14	.74440371	.00802512
.86	.84	.14	.16	.72059506	
.85	.85	.15	.15	.71658408	.00401098
.84	.84	.16	.16	.68854414	
1.00	.82	.00	.18	.91446400	
.91	.91	.09	.09	.87451864	.03994536
.98	.82	.02	.18	.88498793	
.90	.90	.10	.10	.85030560	.03468233
.96	.82	.04	.18	.85474726	
.89	.89	.11	.11	.82498895	.02975831
.94	.82	.06	.18	.82387661	
.88	.88	.12	.12	.79877503	.02510158
.92	.82	.08	.18	.79250554	
.87	.87	.13	.13	.77185463	.02065091
.90	.82	.10	.18	.76075874	
.86	.86	.14	.14	.74440371	.01635503
.88	.82	.12	.18	.72875590	
.85	.85	.15	.15	.71658408	.01217182
.86	.82	.14	.18	.69661180	
.84	.84	.16	.16	.68854414	.00806766
.84	.82	.16	.18	.66443617	
.83	.83	.17	.17	.66041955	.00401662
.82	.82	.18	.18	.63233387	
1.00	.80	.00	.20	.89600000	
.90	.90	.10	.10	.85030560	.04569440
.98	.80	.02	.20	.86499310	
.89	.89	.11	.11	.82498895	.04000415
.96	.80	.04	.20	.83349209	
.88	.88	.12	.12	.79877503	.03471706
.94	.80	.06	.20	.80161323	
.87	.87	.13	.13	.77185463	.02975860
.92	.80	.08	.20	.76946834	
.86	.86	.14	.14	.74440371	.02506463
.90	.80	.10	.20	.73716480	
.85	.85	.15	.15	.71658408	.02058072
.88	.80	.12	.20	.70480560	
.84	.84	.16	.16	.68854414	.01626146
.86	.80	.14	.20	.67248930	
.83	.83	.17	.17	.66041955	.01206975
.84	.80	.16	.20	.64030999	
.82	.82	.18	.18	.63233387	.00797612
.82	.80	.18	.20	.60835740	
.81	.81	.19	.19	.60439921	.00395819
.80	.80	.20	.20	.57671680	

1.00	.78	.00	.22	.87609600	
.89	.89	.11	.11	.82498895	.05110705
.98	.78	.02	.22	.84381462	
.88	.88	.12	.12	.79877503	.04503959
.96	.78	.04	.22	.81128564	
.87	.87	.13	.13	.77185463	.03943101
.94	.78	.06	.22	.77860831	
.86	.86	.14	.14	.74440371	.03420460
.92	.78	.08	.22	.74587794	
.85	.85	.15	.15	.71658408	.02929386
.90	.78	.10	.22	.71318595	
.84	.84	.16	.16	.68854414	.02464181
.88	.78	.12	.22	.68061978	
.83	.83	.17	.17	.66041955	.02020023
.86	.78	.14	.22	.64826299	
.82	.82	.18	.18	.63233387	.01592912
.84	.78	.16	.22	.61619519	
.81	.81	.19	.19	.60439921	.01179598
.82	.78	.18	.22	.58449208	
.80	.80	.20	.20	.57671680	.00777528
.80	.78	.20	.22	.55322542	
.79	.79	.21	.21	.54937765	.00384777
.78	.78	.22	.22	.52246308	
1.00	.76	.00	.24	.85484800	
.88	.88	.12	.12	.79877503	.05607297
.98	.76	.02	.24	.82153743	
.87	.87	.13	.13	.77185463	.04968280
.96	.76	.04	.24	.78820266	
.86	.86	.14	.14	.74440371	.04379895
.94	.76	.06	.24	.75492723	
.85	.85	.15	.15	.71658408	.03834315
.92	.76	.08	.24	.72179120	
.84	.84	.16	.16	.68854414	.03324706
.90	.76	.10	.24	.68887117	
.83	.83	.17	.17	.66041955	.02845162
.88	.76	.12	.24	.65624031	
.82	.82	.18	.18	.63233387	.02390644
.86	.76	.14	.24	.62396829	
.81	.81	.19	.19	.60439921	.01956908
.84	.76	.16	.24	.59212135	
.80	.80	.20	.20	.57671680	.01540455
.82	.76	.18	.24	.56076225	
.79	.79	.21	.21	.54937765	.01138460
.80	.76	.20	.24	.52995031	
.78	.78	.22	.22	.52246308	.00748723
.78	.76	.22	.24	.49974137	
.77	.77	.23	.23	.49604526	.00369611
.76	.76	.24	.24	.47018783	
1.00	.74	.00	.26	.83235200	
.87	.87	.13	.13	.77185463	.06049737
.98	.74	.02	.26	.79824646	
.86	.86	.14	.14	.74440371	.05384275
.96	.74	.04	.26	.76431791	
.85	.85	.15	.15	.71658408	.04773383
.94	.74	.06	.26	.73063537	
.84	.84	.16	.16	.68854414	.04209123
.92	.74	.08	.26	.69726489	
.83	.83	.17	.17	.66041955	.03684534
.90	.74	.10	.26	.66426947	
.82	.82	.18	.18	.63233387	.03193560
.88	.74	.12	.26	.63170906	
.81	.81	.19	.19	.60439921	.02730985

.86	.74	.14	.26	.59964062	
.80	.80	.20	.20	.57671680	.02292382
.84	.74	.16	.26	.56811807	
.79	.79	.21	.21	.54937765	.01874042
.82	.74	.18	.26	.53719228	
.78	.78	.22	.22	.52246308	.01472920
.80	.74	.20	.26	.50691113	
.77	.77	.23	.23	.49604526	.01086587
.78	.74	.22	.26	.47731944	
.76	.76	.24	.24	.47018783	.00713161
.76	.74	.24	.26	.44845900	
.75	.75	.25	.25	.44494630	.00351270
.74	.74	.26	.26	.42036861	
1.00	.72	.00	.28	.80870400	
.86	.86	.14	.14	.74440371	.06430029
.98	.72	.02	.28	.77402666	
.85	.85	.15	.15	.71658408	.05744258
.96	.72	.04	.28	.73970611	
.84	.84	.16	.16	.68854414	.05116197
.94	.72	.06	.28	.70579812	
.83	.83	.17	.17	.66041955	.04537857
.92	.72	.08	.28	.67235584	
.82	.82	.18	.18	.63233387	.04002197
.90	.72	.10	.28	.63942981	
.81	.81	.19	.19	.60439921	.03503060
.88	.72	.12	.28	.60706790	
.80	.80	.20	.20	.57671680	.03035110
.86	.72	.14	.28	.57531539	
.79	.79	.21	.21	.54937765	.02593774
.84	.72	.16	.28	.54421493	
.78	.78	.22	.22	.52246308	.02175185
.82	.72	.18	.28	.51380651	
.77	.77	.23	.23	.49604526	.01776125
.80	.72	.20	.28	.48412754	
.76	.76	.24	.24	.47018783	.01393971
.78	.72	.22	.28	.45521275	
.75	.75	.25	.25	.44494630	.01026645
.76	.72	.24	.28	.42709429	
.74	.74	.26	.26	.42036861	.00672568
.74	.72	.26	.28	.39980166	
.73	.73	.27	.27	.39649567	.00330599
.72	.72	.28	.28	.37336171	
1.00	.70	.00	.30	.78400000	
.85	.85	.15	.15	.71658408	.06741592
.98	.70	.02	.30	.74896294	
.84	.84	.16	.16	.68854414	.06041880
.96	.70	.04	.30	.71444201	
.83	.83	.17	.17	.66041955	.05402246
.94	.70	.06	.30	.68048085	
.82	.82	.18	.18	.63233387	.04814698
.92	.70	.08	.30	.64712087	
.81	.81	.19	.19	.60439921	.04272166
.90	.70	.10	.30	.61440120	
.80	.80	.20	.20	.57671680	.03768440
.88	.70	.12	.30	.58235871	
.79	.79	.21	.21	.54937765	.03298106
.86	.70	.14	.30	.55102804	
.78	.78	.22	.22	.52246308	.02856496
.84	.70	.16	.30	.52044153	
.77	.77	.23	.23	.49604526	.02439627

.82	.70	.18	.30	.49062929	
.76	.76	.24	.24	.47018783	.02044146
.80	.70	.20	.30	.46161920	
.75	.75	.25	.25	.44494630	.01667290
.78	.70	.22	.30	.43343682	
.74	.74	.26	.26	.42036861	.01306821
.76	.70	.24	.30	.40610547	
.73	.73	.27	.27	.39649567	.00960980
.74	.70	.26	.30	.37964627	
.72	.72	.28	.28	.37336171	.00628456
.72	.70	.28	.30	.35407797	
.71	.71	.29	.29	.35099477	.00308320
.70	.70	.30	.30	.32941720	
1.00	.68	.00	.32	.75833600	
.84	.84	.16	.16	.68854414	.06979186
.98	.68	.02	.32	.72314025	
.83	.83	.17	.17	.66041955	.06272070
.96	.68	.04	.32	.68860037	
.82	.82	.18	.18	.63233387	.05626650
.94	.68	.06	.32	.65474897	
.81	.81	.19	.19	.60439921	.05034976
.92	.68	.08	.32	.62161678	
.80	.80	.20	.20	.57671680	.04489998
.90	.68	.10	.32	.58923262	
.79	.79	.21	.21	.54937765	.03985497
.88	.68	.12	.32	.55762335	
.78	.78	.22	.22	.52246308	.03516027
.86	.68	.14	.32	.52681394	
.77	.77	.23	.23	.49604526	.03076868
.84	.68	.16	.32	.49682745	
.76	.76	.24	.24	.47018783	.02663962
.82	.68	.18	.32	.46768498	
.75	.75	.25	.25	.44494630	.02273868
.80	.68	.20	.32	.43940578	
.74	.74	.26	.26	.42036861	.01903717
.78	.68	.22	.32	.41200711	
.73	.73	.27	.27	.39649567	.01551144
.76	.68	.24	.32	.38550437	
.72	.72	.28	.28	.37336171	.01214266
.74	.68	.26	.32	.35991101	
.71	.71	.29	.29	.35099477	.00891624
.72	.68	.28	.32	.33523857	
.70	.70	.30	.30	.32941720	.00582137
.70	.68	.30	.32	.31149668	
.69	.69	.31	.31	.30864594	.00285074
.68	.68	.32	.32	.28869304	
1.00	.66	.00	.34	.73180800	
.83	.83	.17	.17	.66041955	.07138845
.98	.66	.02	.34	.69664351	
.82	.82	.18	.18	.63233387	.06430964
.96	.66	.04	.34	.66225590	
.81	.81	.19	.19	.60439921	.05785669
.94	.66	.06	.34	.62866782	
.80	.80	.20	.20	.57671680	.05195102
.92	.66	.08	.34	.59590039	
.79	.79	.21	.21	.54937765	.04652274
.90	.66	.10	.34	.56397306	
.78	.78	.22	.22	.52246308	.04150998
.88	.66	.12	.34	.53290369	
.77	.77	.23	.23	.49604526	.03685843
.86	.66	.14	.34	.50270854	
.76	.76	.24	.24	.47018783	.03252071

.84	.66	.16	.34	.47340226	
.75	.75	.25	.25	.44494630	.02845596
.82	.66	.18	.34	.44499792	
.74	.74	.26	.26	.42036861	.02462931
.80	.66	.20	.34	.41750692	
.73	.73	.27	.27	.39649567	.02101125
.78	.66	.22	.34	.39093914	
.72	.72	.28	.28	.37336171	.01757743
.76	.66	.24	.34	.36530276	
.71	.71	.29	.29	.35099477	.01430799
.74	.66	.26	.34	.34060445	
.70	.70	.30	.30	.32941720	.01118725
.72	.66	.28	.34	.31684921	
.69	.69	.31	.31	.30864594	.00820327
.70	.66	.30	.34	.29404045	
.68	.68	.32	.32	.28869304	.00534741
.68	.66	.32	.34	.27217999	
.67	.67	.33	.33	.26956596	.00261403
.66	.66	.34	.34	.25126800	
1.00	.64	.00	.36	.70451200	
.82	.82	.18	.18	.63233387	.07217813
.98	.64	.02	.36	.66955766	
.81	.81	.19	.19	.60439921	.06515845
.96	.64	.04	.36	.63548335	
.80	.80	.20	.20	.57671680	.05876655
.94	.64	.06	.36	.60230284	
.79	.79	.21	.21	.54937765	.05292519
.92	.64	.08	.36	.57002852	
.78	.78	.22	.22	.52246308	.04756544
.90	.64	.10	.36	.53867151	
.77	.77	.23	.23	.49604526	.04262625
.88	.64	.12	.36	.50824160	
.76	.76	.24	.24	.47018783	.03805377
.86	.64	.14	.36	.47874723	
.75	.75	.25	.25	.44494630	.03380093
.84	.64	.16	.36	.45019558	
.74	.74	.26	.26	.42036861	.02982697
.82	.64	.18	.36	.42259244	
.73	.73	.27	.27	.39649567	.02609677
.80	.64	.20	.36	.39594229	
.72	.72	.28	.28	.37336171	.02258058
.78	.64	.22	.36	.37024834	
.71	.71	.29	.29	.35099477	.01925357
.76	.64	.24	.36	.34551246	
.70	.70	.30	.30	.32941720	.01609526
.74	.64	.26	.36	.32173515	
.69	.69	.31	.31	.30864594	.01308921
.72	.64	.28	.36	.29891563	
.68	.68	.32	.32	.28869304	.01022259
.70	.64	.30	.36	.27705180	
.67	.67	.33	.33	.26956596	.00748584
.68	.64	.32	.36	.25614022	
.66	.66	.34	.34	.25126800	.00487222
.66	.64	.34	.36	.23617617	
.65	.65	.35	.35	.23379855	.00237762
.64	.64	.36	.36	.21715354	
1.00	.62	.00	.38	.67654400	
.81	.81	.19	.19	.60439921	.07214479
.98	.62	.02	.38	.64196763	
.80	.80	.20	.20	.57671680	.06525083
.96	.62	.04	.38	.60835749	
.79	.79	.21	.21	.54937765	.05897984

.94	.62	.06	.38	.57571936	
.78	.78	.22	.22	.52246308	.05325628
.92	.62	.08	.38	.54405796	
.77	.77	.23	.23	.49604526	.04801270
.90	.62	.10	.38	.51337696	
.76	.76	.24	.24	.47018783	.04318913
.88	.62	.12	.38	.48367895	
.75	.75	.25	.25	.44494630	.03873265
.86	.62	.14	.38	.45496546	
.74	.74	.26	.26	.42036861	.03459685
.84	.62	.16	.38	.42723698	
.73	.73	.27	.27	.39649567	.03074131
.82	.62	.18	.38	.40049289	
.72	.72	.28	.28	.37336171	.02713118
.80	.62	.20	.38	.37473156	
.71	.71	.29	.29	.35099477	.02373679
.78	.62	.22	.38	.34995028	
.70	.70	.30	.30	.32941720	.02053308
.76	.62	.24	.38	.32614525	
.69	.69	.31	.31	.30864594	.01749931
.74	.62	.26	.38	.30331165	
.68	.68	.32	.32	.28869304	.01461861
.72	.62	.28	.38	.28144356	
.67	.67	.33	.33	.26956596	.01187760
.70	.62	.30	.38	.26053401	
.66	.66	.34	.34	.25126800	.00926601
.68	.62	.32	.38	.24057499	
.65	.65	.35	.35	.23379855	.00677644
.66	.62	.34	.38	.22155738	
.64	.64	.36	.36	.21715354	.00440384
.64	.62	.36	.38	.20347105	
.63	.63	.37	.37	.20132566	.00214539
.62	.62	.38	.38	.18630476	
1.00	.60	.00	.40	.64800000	
.80	.80	.20	.20	.57671680	.07128320
.98	.60	.02	.40	.61395836	
.79	.79	.21	.21	.54937765	.06458071
.96	.60	.04	.40	.58095305	
.78	.78	.22	.22	.52246308	.05848997
.94	.60	.06	.40	.54898280	
.77	.77	.23	.23	.49604526	.05293754
.92	.60	.08	.40	.51804555	
.76	.76	.24	.24	.47018783	.04785772
.90	.60	.10	.40	.48813840	
.75	.75	.25	.25	.44494630	.04319210
.88	.60	.12	.40	.45925760	
.74	.74	.26	.26	.42036861	.03888899
.86	.60	.14	.40	.43139861	
.73	.73	.27	.27	.39649567	.03490294
.84	.60	.16	.40	.40455604	
.72	.72	.28	.28	.37336171	.03119433
.82	.60	.18	.40	.37872365	
.71	.71	.29	.29	.35099477	.02772888
.80	.60	.20	.40	.35389440	
.70	.70	.30	.30	.32941720	.02447720
.78	.60	.22	.40	.33006040	
.69	.69	.31	.31	.30864594	.02141446
.76	.60	.24	.40	.30721296	
.68	.68	.32	.32	.28869304	.01851992
.74	.60	.26	.40	.28534253	
.67	.67	.33	.33	.26956596	.01577657
.72	.60	.28	.40	.26443874	

.66	.66	.34	.34	.25126800	.01317074
.70	.60	.30	.40	.24449040	
.65	.65	.35	.35	.23379855	.01069185
.68	.60	.32	.40	.22548547	
.64	.64	.36	.36	.21715354	.00833193
.66	.60	.34	.40	.20741110	
.63	.63	.37	.37	.20132566	.00608544
.64	.60	.36	.40	.19025362	
.62	.62	.38	.38	.18630476	.00394886
.62	.60	.38	.40	.17399850	
.61	.61	.39	.39	.17207804	.00192046
.60	.60	.40	.40	.15863040	
1.00	.58	.00	.42	.61897600	
.79	.79	.21	.21	.54937765	.06959835
.98	.58	.02	.42	.58561478	
.78	.78	.22	.22	.52246308	.06315170
.96	.58	.04	.42	.55334476	
.77	.77	.23	.23	.49604526	.05729950
.94	.58	.06	.42	.52215852	
.76	.76	.24	.24	.47018783	.05197069
.92	.58	.08	.42	.49204808	
.75	.75	.25	.25	.44494630	.04710178
.90	.58	.10	.42	.46300480	
.74	.74	.26	.26	.42036861	.04263619
.88	.58	.12	.42	.43501944	
.73	.73	.27	.27	.39649567	.03852377
.86	.58	.14	.42	.40808213	
.72	.72	.28	.28	.37336171	.03472042
.84	.58	.16	.42	.38218236	
.71	.71	.29	.29	.35099477	.03118759
.82	.58	.18	.42	.35730905	
.70	.70	.30	.30	.32941720	.02789185
.80	.58	.20	.42	.33345044	
.69	.69	.31	.31	.30864594	.02480450
.78	.58	.22	.42	.31059421	
.68	.68	.32	.32	.28869304	.02190117
.76	.58	.24	.42	.28872736	
.67	.67	.33	.33	.26956596	.01916140
.74	.58	.26	.42	.26783633	
.66	.66	.34	.34	.25126800	.01656833
.72	.58	.28	.42	.24790690	
.65	.65	.35	.35	.23379855	.01410835
.70	.58	.30	.42	.22892424	
.64	.64	.36	.36	.21715354	.01177070
.68	.58	.32	.42	.21087288	
.63	.63	.37	.37	.20132566	.00954722
.66	.58	.34	.42	.19373678	
.62	.62	.38	.38	.18630476	.00743202
.64	.58	.36	.42	.17749924	
.61	.61	.39	.39	.17207804	.00542120
.62	.58	.38	.42	.16214296	
.60	.60	.40	.40	.15863040	.00351256
.60	.58	.40	.42	.14764999	
.59	.59	.41	.41	.14594464	.00170535
.58	.58	.42	.42	.13400179	
1.00	.56	.00	.44	.58956800	
.78	.78	.22	.22	.52246308	.06710492
.98	.56	.02	.44	.55702183	
.77	.77	.23	.23	.49604526	.06097657
.96	.56	.04	.44	.52560737	
.76	.76	.24	.24	.47018783	.05541954

.94	.56	.06	.44	.49531191	
.75	.75	.25	.25	.44494630	.05036561
.92	.56	.08	.44	.46612239	
.74	.74	.26	.26	.42036861	.04575378
.90	.56	.10	.44	.43802519	
.73	.73	.27	.27	.39649567	.04152952
.88	.56	.12	.44	.41100634	
.72	.72	.28	.28	.37336171	.03764463
.86	.56	.14	.44	.38505140	
.71	.71	.29	.29	.35099477	.03405663
.84	.56	.16	.44	.36014554	
.70	.70	.30	.30	.32941720	.03072834
.82	.56	.18	.44	.33627343	
.69	.69	.31	.31	.30864594	.02762749
.80	.56	.20	.44	.31341936	
.68	.68	.32	.32	.28869304	.02472632
.78	.56	.22	.44	.29156717	
.67	.67	.33	.33	.26956596	.02200121
.76	.56	.24	.44	.27070027	
.66	.66	.34	.34	.25126800	.01943227
.74	.56	.26	.44	.25080162	
.65	.65	.35	.35	.23379855	.01700307
.72	.56	.28	.44	.23185377	
.64	.64	.36	.36	.21715354	.01470023
.70	.56	.30	.44	.21383881	
.63	.63	.37	.37	.20132566	.01251315
.68	.56	.32	.44	.19673843	
.62	.62	.38	.38	.18630476	.01043367
.66	.56	.34	.44	.18053387	
.61	.61	.39	.39	.17207804	.00845583
.64	.56	.36	.44	.16520591	
.60	.60	.40	.40	.15863040	.00657551
.62	.56	.38	.44	.15073494	
.59	.59	.41	.41	.14594464	.00479030
.60	.56	.40	.44	.13710090	
.58	.58	.42	.42	.13400179	.00309911
.58	.56	.42	.44	.12428328	
.57	.57	.43	.43	.12278128	.00150200
.56	.56	.44	.44	.11226116	
1.00	.54	.00	.46	.55987200	
.77	.77	.23	.23	.49604526	.06382674
.98	.54	.02	.46	.52826442	
.76	.76	.24	.24	.47018783	.05807659
.96	.54	.04	.46	.49781562	
.75	.75	.25	.25	.44494630	.05286932
.94	.54	.06	.46	.46850838	
.74	.74	.26	.26	.42036861	.04813977
.92	.54	.08	.46	.44032526	
.73	.73	.27	.27	.39649567	.04382959
.90	.54	.10	.46	.41324852	
.72	.72	.28	.28	.37336171	.03988681
.88	.54	.12	.46	.38726014	
.71	.71	.29	.29	.35099477	.03626537
.86	.54	.14	.46	.36234186	
.70	.70	.30	.30	.32941720	.03292466
.84	.54	.16	.46	.33847514	
.69	.69	.31	.31	.30864594	.02982920
.82	.54	.18	.46	.31564115	
.68	.68	.32	.32	.28869304	.02694811
.80	.54	.20	.46	.29382082	
.67	.67	.33	.33	.26956596	.02425486
.78	.54	.22	.46	.27299481	

.66	.66	.34	.34	.25126800	.02172681
.76	.54	.24	.46	.25314348	
.65	.65	.35	.35	.23379855	.01934493
.74	.54	.26	.46	.23424696	
.64	.64	.36	.36	.21715354	.01709342
.72	.54	.28	.46	.21628509	
.63	.63	.37	.37	.20132566	.01495943
.70	.54	.30	.46	.19923744	
.62	.62	.38	.38	.18630476	.01293268
.68	.54	.32	.46	.18308334	
.61	.61	.39	.39	.17207804	.01100530
.66	.54	.34	.46	.16780180	
.60	.60	.40	.40	.15863040	.00917140
.64	.54	.36	.46	.15337160	
.59	.59	.41	.41	.14594464	.00742696
.62	.54	.38	.46	.13977125	
.58	.58	.42	.42	.13400179	.00576946
.60	.54	.40	.46	.12697896	
.57	.57	.43	.43	.12278128	.00419768
.58	.54	.42	.46	.11497272	
.56	.56	.44	.44	.11226116	.00271156
.56	.54	.44	.46	.10373021	
.55	.55	.45	.45	.10241837	.00131184
.54	.54	.46	.46	.09322886	
1.00	.52	.00	.48	.52998400	
.76	.76	.24	.24	.47018783	.05979617
.98	.52	.02	.48	.49942751	
.75	.75	.25	.25	.44494630	.05448121
.96	.52	.04	.48	.47004422	
.74	.74	.26	.26	.42036861	.04967561
.94	.52	.06	.48	.44181328	
.73	.73	.27	.27	.39649567	.04531761
.92	.52	.08	.48	.41471354	
.72	.72	.28	.28	.37336171	.04135183
.90	.52	.10	.48	.38872379	
.71	.71	.29	.29	.35099477	.03772902
.88	.52	.12	.48	.36382274	
.70	.70	.30	.30	.32941720	.03440554
.86	.52	.14	.48	.33998892	
.69	.69	.31	.31	.30864594	.03134298
.84	.52	.16	.48	.31720076	
.68	.68	.32	.32	.28869304	.02850772
.82	.52	.18	.48	.29543655	
.67	.67	.33	.33	.26956596	.02587059
.80	.52	.20	.48	.27467448	
.66	.66	.34	.34	.25126800	.02340648
.78	.52	.22	.48	.25489259	
.65	.65	.35	.35	.23379855	.02109404
.76	.52	.24	.48	.23606879	
.64	.64	.36	.36	.21715354	.01891525
.74	.52	.26	.48	.21818090	
.63	.63	.37	.37	.20132566	.01685524
.72	.52	.28	.48	.20120659	
.62	.62	.38	.38	.18630476	.01490183
.70	.52	.30	.48	.18512341	
.61	.61	.39	.39	.17207804	.01304537
.68	.52	.32	.48	.16990878	
.60	.60	.40	.40	.15863040	.01127838
.66	.52	.34	.48	.15554002	
.59	.59	.41	.41	.14594464	.00959538

.64	.52	.36	.48	.14199430	
.58	.58	.42	.42	.13400179	.00799251
.62	.52	.38	.48	.12924866	
.57	.57	.43	.43	.12278128	.00646738
.60	.52	.40	.48	.11728005	
.56	.56	.44	.44	.11226116	.00501889
.58	.52	.42	.48	.10606525	
.55	.55	.45	.45	.10241837	.00364688
.56	.52	.44	.48	.09558096	
.54	.54	.46	.46	.09322886	.00235209
.54	.52	.46	.48	.08580372	
.53	.53	.47	.47	.08466785	.00113586
.52	.52	.48	.48	.07670996	
1.00	.50	.00	.50	.50000000	
.75	.75	.25	.25	.44494630	.05505370
.98	.50	.02	.50	.47059599	
.74	.74	.26	.26	.42036861	.05022738
.96	.50	.04	.50	.44236799	
.73	.73	.27	.27	.39649567	.04587232
.94	.50	.06	.50	.41529199	
.72	.72	.28	.28	.37336171	.04193028
.92	.50	.08	.50	.38934399	
.71	.71	.29	.29	.35099477	.03834922
.90	.50	.10	.50	.36450000	
.70	.70	.30	.30	.32941720	.03508280
.88	.50	.12	.50	.34073600	
.69	.69	.31	.31	.30864594	.03209006
.86	.50	.14	.50	.31802800	
.68	.68	.32	.32	.28869304	.02933496
.84	.50	.16	.50	.29635200	
.67	.67	.33	.33	.26956596	.02678604
.82	.50	.18	.50	.27568400	
.66	.66	.34	.34	.25126800	.02441600
.80	.50	.20	.50	.25600000	
.65	.65	.35	.35	.23379855	.02220145
.78	.50	.22	.50	.23727600	
.64	.64	.36	.36	.21715354	.02012246
.76	.50	.24	.50	.21948800	
.63	.63	.37	.37	.20132566	.01816234
.74	.50	.26	.50	.20261200	
.62	.62	.38	.38	.18630476	.01630724
.72	.50	.28	.50	.18662400	
.61	.61	.39	.39	.17207804	.01454596
.70	.50	.30	.50	.17150000	
.60	.60	.40	.40	.15863040	.01286960
.68	.50	.32	.50	.15721600	
.59	.59	.41	.41	.14594464	.01127136
.66	.50	.34	.50	.14374800	
.58	.58	.42	.42	.13400179	.00974621
.64	.50	.36	.50	.13107200	
.57	.57	.43	.43	.12278128	.00829072
.62	.50	.38	.50	.11916400	
.56	.56	.44	.44	.11226116	.00690284
.60	.50	.40	.50	.10800000	
.55	.55	.45	.45	.10241837	.00558163
.58	.50	.42	.50	.09755600	
.54	.54	.46	.46	.09322886	.00432713
.56	.50	.44	.50	.08780800	
.53	.53	.47	.47	.08466785	.00314014
.54	.50	.46	.50	.07873200	
.52	.52	.48	.48	.07670996	.00202203

.52	.50	.48	.50	.07030400	
.51	.51	.49	.49	.06932937	.00097462
.50	.50	.50	.50	.06250000	

STOP

PROBABILITIES FOR CORRECT DECODING FOR CODE III

TABLE 5.8

q_1	q_2	p_1	p_2	Pr(A),Pr(S)	Pr(A)-Pr(S)
1.00	1.00	.00	.00	1.00000000	
1.00	.98	.00	.02	.99766352	
.99	.99	.01	.01	.99656426	.00109926
.98	.98	.02	.02	.98688511	
1.00	.96	.00	.04	.99090432	
.98	.98	.02	.02	.98688511	.00401921
.98	.96	.02	.04	.97404005	
.97	.97	.03	.03	.97184176	.00219829
.96	.96	.04	.04	.95223424	
1.00	.94	.00	.06	.98008912	
.97	.97	.03	.03	.97184176	.00824736
.98	.94	.02	.06	.95793673	
.96	.96	.04	.04	.95223424	.00570249
.96	.94	.04	.06	.93176321	
.95	.95	.05	.05	.92878861	.00297460
.94	.94	.06	.06	.90216202	
1.00	.92	.00	.08	.96557312	
.96	.96	.04	.04	.95223424	.01333888
.98	.92	.02	.08	.93887943	
.95	.95	.05	.05	.92878861	.01009082
.96	.92	.04	.08	.90899514	
.94	.94	.06	.06	.90216202	.00683312
.94	.92	.06	.08	.87643491	
.93	.93	.07	.07	.87294761	.00348730
.92	.92	.08	.08	.84167895	
1.00	.90	.00	.10	.94770000	
.95	.95	.05	.05	.92878861	.01891139
.98	.90	.02	.10	.91716232	
.94	.94	.06	.06	.90216202	.01500030
.96	.90	.04	.10	.88417996	
.93	.93	.07	.07	.87294761	.01123235
.94	.90	.06	.10	.84919566	
.92	.92	.08	.08	.84167895	.00751671
.92	.90	.08	.10	.81262173	
.91	.91	.09	.09	.80883434	.00378739
.90	.90	.10	.10	.77484098	
1.00	.88	.00	.12	.92680192	
.94	.94	.06	.06	.90216202	.02463990
.98	.88	.02	.12	.89306952	
.93	.93	.07	.07	.87294761	.02012191
.96	.88	.04	.12	.85755882	
.92	.92	.08	.08	.84167895	.01587987
.94	.88	.06	.12	.82064712	
.91	.91	.09	.09	.80883434	.01181278
.92	.88	.08	.12	.78268503	
.90	.90	.10	.10	.77484098	.00784405
.90	.88	.10	.12	.74399732	
.89	.89	.11	.11	.74007869	.00391863
.88	.88	.12	.12	.70488367	
1.00	.86	.00	.14	.90319952	
.93	.93	.07	.07	.87294761	.03025191
.98	.86	.02	.14	.86687512	
.92	.92	.08	.08	.84167895	.02519617
.96	.86	.04	.14	.82936412	
.91	.91	.09	.09	.80883434	.02052978
.94	.86	.06	.14	.79098458	
.90	.90	.10	.10	.77484098	.01614360
.92	.86	.08	.14	.75203123	
.89	.89	.11	.11	.74007869	.01195254

.90	.86	.10	.14	.71277632	
.88	.88	.12	.12	.70488367	.00789265
.88	.86	.12	.14	.67347018	
.87	.87	.13	.13	.66955181	.00391837
.86	.86	.14	.14	.63434201	
1.00	.84	.00	.16	.87720192	
.92	.92	.08	.08	.84167895	.03552297
.98	.84	.02	.16	.83884312	
.91	.91	.09	.09	.80883434	.03000878
.96	.84	.04	.16	.79981956	
.90	.90	.10	.10	.77484098	.02497858
.94	.84	.06	.16	.76039577	
.89	.89	.11	.11	.74007869	.02031708
.92	.84	.08	.16	.72081618	
.88	.88	.12	.12	.70488367	.01593251
.90	.84	.10	.16	.68130566	
.87	.87	.13	.13	.66955181	.01175385
.88	.84	.12	.16	.64207015	
.86	.86	.14	.14	.63434201	.00772814
.86	.84	.14	.16	.60329727	
.85	.85	.15	.15	.59947916	.00381811
.84	.84	.16	.16	.56515703	
1.00	.82	.00	.18	.84910672	
.91	.91	.09	.09	.80883434	.04027238
.98	.82	.02	.18	.80922749	
.90	.90	.10	.10	.77484098	.03438651
.96	.82	.04	.18	.76914005	
.89	.89	.11	.11	.74007869	.02906136
.94	.82	.06	.18	.72906092	
.88	.88	.12	.12	.70488367	.02417725
.92	.82	.08	.18	.68918929	
.87	.87	.13	.13	.66955181	.01963748
.90	.82	.10	.18	.64970757	
.86	.86	.14	.14	.63434201	.01536556
.88	.82	.12	.18	.61078190	
.85	.85	.15	.15	.59947916	.01130274
.86	.82	.14	.18	.57256274	
.84	.84	.16	.16	.56515703	.00740571
.84	.82	.16	.18	.53518535	
.83	.83	.17	.17	.53154096	.00364439
.82	.82	.18	.18	.49877037	
1.00	.80	.00	.20	.81920000	
.90	.90	.10	.10	.77484098	.04435902
.98	.80	.02	.20	.77827212	
.89	.89	.11	.11	.74007869	.03819343
.96	.80	.04	.20	.73753178	
.88	.88	.12	.12	.70488367	.03264811
.94	.80	.06	.20	.69715261	
.87	.87	.13	.13	.66955181	.02760080
.92	.80	.08	.20	.65729340	
.86	.86	.14	.14	.63434201	.02295139
.90	.80	.10	.20	.61809868	
.85	.85	.15	.15	.59947916	.01861952
.88	.80	.12	.20	.57969912	
.84	.84	.16	.16	.56515703	.01454209
.86	.80	.14	.20	.54221200	
.83	.83	.17	.17	.53154096	.01067104
.84	.80	.16	.20	.50574171	
.82	.82	.18	.18	.49877037	.00697134
.82	.80	.18	.20	.47038024	
.81	.81	.19	.19	.46696109	.00341915
.80	.80	.20	.20	.43620762	

1.00	.78	.00	.22	.78775632	
.89	.89	.11	.11	.74007869	.04767763
.98	.78	.02	.22	.74621082	
.88	.88	.12	.12	.70488367	.04132715
.96	.78	.04	.22	.70519222	
.87	.87	.13	.13	.66955181	.03564041
.94	.78	.06	.22	.66483593	
.86	.86	.14	.14	.63434201	.03049392
.92	.78	.08	.22	.62526493	
.85	.85	.15	.15	.59947916	.02578577
.90	.78	.10	.22	.58659012	
.84	.84	.16	.16	.56515703	.02143309
.88	.78	.12	.22	.54891071	
.83	.83	.17	.17	.53154096	.01736975
.86	.78	.14	.22	.51231461	
.82	.82	.18	.18	.49877037	.01354424
.84	.78	.16	.22	.47687899	
.81	.81	.19	.19	.46696109	.00991790
.82	.78	.18	.22	.44267046	
.80	.80	.20	.20	.43620762	.00646284
.80	.78	.20	.22	.40974567	
.79	.79	.21	.21	.40658516	.00316051
.78	.78	.22	.22	.37815157	
1.00	.76	.00	.24	.75503872	
.88	.88	.12	.12	.70488367	.05015505
.98	.76	.02	.24	.71326741	
.87	.87	.13	.13	.66955181	.04371560
.96	.76	.04	.24	.67231005	
.86	.86	.14	.14	.63434201	.03796804
.94	.76	.06	.24	.63226840	
.85	.85	.15	.15	.59947916	.03278924
.92	.76	.08	.24	.59323378	
.84	.84	.16	.16	.56515703	.02807675
.90	.76	.10	.24	.55528743	
.83	.83	.17	.17	.53154096	.02374647
.88	.76	.12	.24	.51850088	
.82	.82	.18	.18	.49877037	.01973051
.86	.76	.14	.24	.48293630	
.81	.81	.19	.19	.46696109	.01597521
.84	.76	.16	.24	.44864678	
.80	.80	.20	.20	.43620762	.01243916
.82	.76	.18	.24	.41567683	
.79	.79	.21	.21	.40658516	.00909167
.80	.76	.20	.24	.38406256	
.78	.78	.22	.22	.37815157	.00591099
.78	.76	.22	.24	.35383215	
.77	.77	.23	.23	.35094910	.00288305
.76	.76	.24	.24	.32500615	
1.00	.74	.00	.26	.72129872	
.87	.87	.13	.13	.66955181	.05174691
.98	.74	.02	.26	.67965558	
.86	.86	.14	.14	.63434201	.04531357
.96	.74	.04	.26	.63906527	
.85	.85	.15	.15	.59947916	.03958611
.94	.74	.06	.26	.59959999	
.84	.84	.16	.16	.56515703	.03444296
.92	.74	.08	.26	.56132334	
.83	.83	.17	.17	.53154096	.02978238
.90	.74	.10	.26	.52429064	
.82	.82	.18	.18	.49877037	.02552027
.88	.74	.12	.26	.48854922	

.81	.81	.19	.19	.46696109	.02158813
.86	.74	.14	.26	.45413872	
.80	.80	.20	.20	.43620762	.01793110
.84	.74	.16	.26	.42109140	
.79	.79	.21	.21	.40658516	.01450624
.82	.74	.18	.26	.38943242	
.78	.78	.22	.22	.37815157	.01128085
.80	.74	.20	.26	.35918018	
.77	.77	.23	.23	.35094910	.00823108
.78	.74	.22	.26	.33034655	
.76	.76	.24	.24	.32500615	.00534040
.76	.74	.24	.26	.30293721	
.75	.75	.25	.25	.30033874	.00259847
.74	.74	.26	.26	.27695198	
1.00	.72	.00	.28	.68677632	
.86	.86	.14	.14	.63434201	.05243431
.98	.72	.02	.28	.64557898	
.85	.85	.15	.15	.59947916	.04609982
.96	.72	.04	.28	.60562910	
.84	.84	.16	.16	.56515703	.04047207
.94	.72	.06	.28	.56697310	
.83	.83	.17	.17	.53154096	.03543214
.92	.72	.08	.28	.52965054	
.82	.82	.18	.18	.49877037	.03088017
.90	.72	.10	.28	.49369423	
.81	.81	.19	.19	.46696109	.02673314
.88	.72	.12	.28	.45913055	
.80	.80	.20	.20	.43620762	.02292293
.86	.72	.14	.28	.42597967	
.79	.79	.21	.21	.40658516	.01939451
.84	.72	.16	.28	.39425584	
.78	.78	.22	.22	.37815157	.01610427
.82	.72	.18	.28	.36396763	
.77	.77	.23	.23	.35094910	.01301853
.80	.72	.20	.28	.33511817	
.76	.76	.24	.24	.32500615	.01011202
.78	.72	.22	.28	.30770539	
.75	.75	.25	.25	.30033874	.00736665
.76	.72	.24	.28	.28172231	
.74	.74	.26	.26	.27695198	.00477033
.74	.72	.26	.28	.25715725	
.73	.73	.27	.27	.25484137	.00231588
.72	.72	.28	.28	.23399413	
1.00	.70	.00	.30	.65170000	
.85	.85	.15	.15	.59947916	.05222084
.98	.70	.02	.30	.61123126	
.84	.84	.16	.16	.56515703	.04607423
.96	.70	.04	.30	.57216401	
.83	.83	.17	.17	.53154096	.04062305
.94	.70	.06	.30	.53452259	
.82	.82	.18	.18	.49877037	.03575222
.92	.70	.08	.30	.49832579	
.81	.81	.19	.19	.46696109	.03136470
.90	.70	.10	.30	.46358713	
.80	.80	.20	.20	.43620762	.02737951
.88	.70	.12	.30	.43031500	
.79	.79	.21	.21	.40658516	.02372984
.86	.70	.14	.30	.39851295	
.78	.78	.22	.22	.37815157	.02036138
.84	.70	.16	.30	.36817986	
.77	.77	.23	.23	.35094910	.01723076

.82	.70	.18	.30	.33931014	
.76	.76	.24	.24	.32500615	.01430399
.80	.70	.20	.30	.31189401	
.75	.75	.25	.25	.30033874	.01155527
.78	.70	.22	.30	.28591762	
.74	.74	.26	.26	.27695198	.00896564
.76	.70	.24	.30	.26136329	
.73	.73	.27	.27	.25484137	.00652192
.74	.70	.26	.30	.23820975	
.72	.72	.28	.28	.23399413	.00421562
.72	.70	.28	.30	.21643235	
.71	.71	.29	.29	.21439017	.00204218
.70	.70	.30	.30	.19600323	
1.00	.68	.00	.32	.61628672	
.84	.84	.16	.16	.56515703	.05112969
.98	.68	.02	.32	.57679587	
.83	.83	.17	.17	.53154096	.04525491
.96	.68	.04	.32	.53882377	
.82	.82	.18	.18	.49877037	.04005340
.94	.68	.06	.32	.50237576	
.81	.81	.19	.19	.46696109	.03541467
.92	.68	.08	.32	.46745305	
.80	.80	.20	.20	.43620762	.03124543
.90	.68	.10	.32	.43405275	
.79	.79	.21	.21	.40658516	.02746759
.88	.68	.12	.32	.40216805	
.78	.78	.22	.22	.37815157	.02401648
.86	.68	.14	.32	.37178844	
.77	.77	.23	.23	.35094910	.02083934
.84	.68	.16	.32	.34289986	
.76	.76	.24	.24	.32500615	.01789371
.82	.68	.18	.32	.31548490	
.75	.75	.25	.25	.30033874	.01514616
.80	.68	.20	.32	.28952294	
.74	.74	.26	.26	.27695198	.01257096
.78	.68	.22	.32	.26499033	
.73	.73	.27	.27	.25484137	.01014896
.76	.68	.24	.32	.24186056	
.72	.72	.28	.28	.23399413	.00786643
.74	.68	.26	.32	.22010447	
.71	.71	.29	.29	.21439017	.00571430
.72	.68	.28	.32	.19969040	
.70	.70	.30	.30	.19600323	.00368717
.70	.68	.30	.32	.18058433	
.69	.69	.31	.31	.17880182	.00178251
.68	.68	.32	.32	.16275011	
1.00	.66	.00	.34	.58074192	
.83	.83	.17	.17	.53154096	.04920096
.98	.66	.02	.34	.54244640	
.82	.82	.18	.18	.49877037	.04367603
.96	.66	.04	.34	.50575337	
.81	.81	.19	.19	.46696109	.03879228
.94	.66	.06	.34	.47065234	
.80	.80	.20	.20	.43620762	.03444472
.92	.66	.08	.34	.43712973	
.79	.79	.21	.21	.40658516	.03054457
.90	.66	.10	.34	.40516892	
.78	.78	.22	.22	.37815157	.02701735
.88	.66	.12	.34	.37475041	
.77	.77	.23	.23	.35094910	.02380131

.86	.66	.14	.34	.34585204	
.76	.76	.24	.24	.32500615	.02084589
.84	.66	.16	.34	.31844902	
.75	.75	.25	.25	.30033874	.01811028
.82	.66	.18	.34	.29251418	
.74	.74	.26	.26	.27695198	.01556220
.80	.66	.20	.34	.26801803	
.73	.73	.27	.27	.25484137	.01317666
.78	.66	.22	.34	.24492892	
.72	.72	.28	.28	.23399413	.01093479
.76	.66	.24	.34	.22321320	
.71	.71	.29	.29	.21439017	.00882303
.74	.66	.26	.34	.20283537	
.70	.70	.30	.30	.19600323	.00683214
.72	.66	.28	.34	.18375817	
.69	.69	.31	.31	.17880182	.00495635
.70	.66	.30	.34	.16594280	
.68	.68	.32	.32	.16275011	.00319269
.68	.66	.32	.34	.14934897	
.67	.67	.33	.33	.14780863	.00154034
.66	.66	.34	.34	.13393510	
1.00	.64	.00	.36	.54525952	
.82	.82	.18	.18	.49877037	.04648915
.98	.64	.02	.36	.50834621	
.81	.81	.19	.19	.46696109	.04138512
.96	.64	.04	.36	.47308907	
.80	.80	.20	.20	.43620762	.03688145
.94	.64	.06	.36	.43946453	
.79	.79	.21	.21	.40658516	.03287937
.92	.64	.08	.36	.40744678	
.78	.78	.22	.22	.37815157	.02929521
.90	.64	.10	.36	.37700795	
.77	.77	.23	.23	.35094910	.02605885
.88	.64	.12	.36	.34811816	
.76	.76	.24	.24	.32500615	.02311201
.86	.64	.14	.36	.32074572	
.75	.75	.25	.25	.30033874	.02040698
.84	.64	.16	.36	.29485720	
.74	.74	.26	.26	.27695198	.01790522
.82	.64	.18	.36	.27041752	
.73	.73	.27	.27	.25484137	.01557615
.80	.64	.20	.36	.24739011	
.72	.72	.28	.28	.23399413	.01339598
.78	.64	.22	.36	.22573698	
.71	.71	.29	.29	.21439017	.01134681
.76	.64	.24	.36	.20541886	
.70	.70	.30	.30	.19600323	.00941563
.74	.64	.26	.36	.18639530	
.69	.69	.31	.31	.17880182	.00759348
.72	.64	.28	.36	.16862481	
.68	.68	.32	.32	.16275011	.00587470
.70	.64	.30	.36	.15206491	
.67	.67	.33	.33	.14780863	.00425628
.68	.64	.32	.36	.13667230	
.66	.66	.34	.34	.13393510	.00273720
.66	.64	.34	.36	.12240297	
.65	.65	.35	.35	.12108502	.00131795
.64	.64	.36	.36	.10921229	
1.00	.62	.00	.38	.51002192	
.81	.81	.19	.19	.46696109	.04306083
.98	.62	.02	.38	.47464863	

.80	.80	.20	.20	.43620762	.03844101
.96	.62	.04	.38	.44095841	
.79	.79	.21	.21	.40658516	.03437325
.94	.62	.06	.38	.40891695	
.78	.78	.22	.22	.37815157	.03076538
.92	.62	.08	.38	.37848867	
.77	.77	.23	.23	.35094910	.02753957
.90	.62	.10	.38	.34963661	
.76	.76	.24	.24	.32500615	.02463046
.88	.62	.12	.38	.32232265	
.75	.75	.25	.25	.30033874	.02198391
.86	.62	.14	.38	.29650752	
.74	.74	.26	.26	.27695198	.01955554
.84	.62	.16	.38	.27215097	
.73	.73	.27	.27	.25484137	.01730960
.82	.62	.18	.38	.24921177	
.72	.72	.28	.28	.23399413	.01521764
.80	.62	.20	.38	.22764785	
.71	.71	.29	.29	.21439017	.01325768
.78	.62	.22	.38	.20741637	
.70	.70	.30	.30	.19600323	.01141314
.76	.62	.24	.38	.18847383	
.69	.69	.31	.31	.17880182	.00967201
.74	.62	.26	.38	.17077612	
.68	.68	.32	.32	.16275011	.00802601
.72	.62	.28	.38	.15427863	
.67	.67	.33	.33	.14780863	.00647000
.70	.62	.30	.38	.13893635	
.66	.66	.34	.34	.13393510	.00500125
.68	.62	.32	.38	.12470394	
.65	.65	.35	.35	.12108502	.00361892
.66	.62	.34	.38	.11153581	
.64	.64	.36	.36	.10921229	.00232352
.64	.62	.36	.38	.09938622	
.63	.63	.37	.37	.09826969	.00111653
.62	.62	.38	.38	.08820940	
1.00	.60	.00	.40	.47520000	
.80	.80	.20	.20	.43620762	.03899238
.98	.60	.02	.40	.44149706	
.79	.79	.21	.21	.40658516	.03491190
.96	.60	.04	.40	.40948016	
.78	.78	.22	.22	.37815157	.03132859
.94	.60	.06	.40	.37910670	
.77	.77	.23	.23	.35094910	.02815760
.92	.60	.08	.40	.35033334	
.76	.76	.24	.24	.32500615	.02532719
.90	.60	.10	.40	.32311612	
.75	.75	.25	.25	.30033874	.02277738
.88	.60	.12	.40	.29741051	
.74	.74	.26	.26	.27695198	.02045853
.86	.60	.14	.40	.27317150	
.73	.73	.27	.27	.25484137	.01833013
.84	.60	.16	.40	.25035362	
.72	.72	.28	.28	.23399413	.01635949
.82	.60	.18	.40	.22891105	
.71	.71	.29	.29	.21439017	.01452088
.80	.60	.20	.40	.20879769	
.70	.70	.30	.30	.19600323	.01279446
.78	.60	.22	.40	.18996719	
.69	.69	.31	.31	.17880182	.01116537

.76	.60	.24	.40	.17237302	
.68	.68	.32	.32	.16275011	.00962291
.74	.60	.26	.40	.15596857	
.67	.67	.33	.33	.14780863	.00815994
.72	.60	.28	.40	.14070720	
.66	.66	.34	.34	.13393510	.00677210
.70	.60	.30	.40	.12654230	
.65	.65	.35	.35	.12108502	.00545728
.68	.60	.32	.40	.11342734	
.64	.64	.36	.36	.10921229	.00421505
.66	.60	.34	.40	.10131598	
.63	.63	.37	.37	.09826969	.00304629
.64	.60	.36	.40	.09016210	
.62	.62	.38	.38	.08820940	.00195269
.62	.60	.38	.40	.07991988	
.61	.61	.39	.39	.07898341	.00093646
.60	.60	.40	.40	.07054387	
1.00	.58	.00	.42	.44095312	
.79	.79	.21	.21	.40658516	.03436796
.98	.58	.02	.42	.40902466	
.78	.78	.22	.22	.37815157	.03087309
.96	.58	.04	.42	.37876438	
.77	.77	.23	.23	.35094910	.02781528
.94	.58	.06	.42	.35012330	
.76	.76	.24	.24	.32500615	.02511715
.92	.58	.08	.42	.32305229	
.75	.75	.25	.25	.30033874	.02271355
.90	.58	.10	.42	.29750217	
.74	.74	.26	.26	.27695198	.02055019
.88	.58	.12	.42	.27342374	
.73	.73	.27	.27	.25484137	.01858237
.86	.58	.14	.42	.25076777	
.72	.72	.28	.28	.23399413	.01677364
.84	.58	.16	.42	.22948513	
.71	.71	.29	.29	.21439017	.01509496
.82	.58	.18	.42	.20952679	
.70	.70	.30	.30	.19600323	.01352356
.80	.58	.20	.42	.19084388	
.69	.69	.31	.31	.17880182	.01204206
.78	.58	.22	.42	.17338773	
.68	.68	.32	.32	.16275011	.01063762
.76	.58	.24	.42	.15710993	
.67	.67	.33	.33	.14780863	.00930130
.74	.58	.26	.42	.14196237	
.66	.66	.34	.34	.13393510	.00802727
.72	.58	.28	.42	.12789729	
.65	.65	.35	.35	.12108502	.00681227
.70	.58	.30	.42	.11486735	
.64	.64	.36	.36	.10921229	.00565506
.68	.58	.32	.42	.10282561	
.63	.63	.37	.37	.09826969	.00455592
.66	.58	.34	.42	.09172567	
.62	.62	.38	.38	.08820940	.00351627
.64	.58	.36	.42	.08152165	
.61	.61	.39	.39	.07898341	.00253824
.62	.58	.38	.42	.07216827	
.60	.60	.40	.40	.07054387	.00162439
.60	.58	.40	.42	.06362087	
.59	.59	.41	.41	.06284342	.00077744
.58	.58	.42	.42	.05583550	
1.00	.56	.00	.44	.40742912	
.78	.78	.22	.22	.37815157	.02927755

.98	.56	.02	.44	.37735468	
.77	.77	.23	.23	.35094910	.02640558
.96	.56	.04	.44	.34891237	
.76	.76	.24	.24	.32500615	.02390622
.94	.56	.06	.44	.32204868	
.75	.75	.25	.25	.30033874	.02170994
.92	.56	.08	.44	.29671046	
.74	.74	.26	.26	.27695198	.01975848
.90	.56	.10	.44	.27284488	
.73	.73	.27	.27	.25484137	.01800351
.88	.56	.12	.44	.25039956	
.72	.72	.28	.28	.23399413	.01640543
.86	.56	.14	.44	.22932251	
.71	.71	.29	.29	.21439017	.01493234
.84	.56	.16	.44	.20956223	
.70	.70	.30	.30	.19600323	.01355900
.82	.56	.18	.44	.19106771	
.69	.69	.31	.31	.17880182	.01226589
.80	.56	.20	.44	.17378847	
.68	.68	.32	.32	.16275011	.01103836
.78	.56	.22	.44	.15767457	
.67	.67	.33	.33	.14780863	.00986594
.76	.56	.24	.44	.14267671	
.66	.66	.34	.34	.13393510	.00874161
.74	.56	.26	.44	.12874616	
.65	.65	.35	.35	.12108502	.00766114
.72	.56	.28	.44	.11583489	
.64	.64	.36	.36	.10921229	.00662260
.70	.56	.30	.44	.10389554	
.63	.63	.37	.37	.09826969	.00562585
.68	.56	.32	.44	.09288149	
.62	.62	.38	.38	.08820940	.00467209
.66	.56	.34	.44	.08274686	
.61	.61	.39	.39	.07898341	.00376344
.64	.56	.36	.44	.07344655	
.60	.60	.40	.40	.07054387	.00290268
.62	.56	.38	.44	.06493630	
.59	.59	.41	.41	.06284342	.00209287
.60	.56	.40	.44	.05717270	
.58	.58	.42	.42	.05583550	.00133719
.58	.56	.42	.44	.05011321	
.57	.57	.43	.43	.04947454	.00063866
.56	.56	.44	.44	.04371622	
1.00	.54	.00	.46	.37476432	
.77	.77	.23	.23	.35094910	.02381522
.98	.54	.02	.46	.34660022	
.76	.76	.24	.24	.32500615	.02159407
.96	.54	.04	.46	.32001668	
.75	.75	.25	.25	.30033874	.01967794
.94	.54	.06	.46	.29495729	
.74	.74	.26	.26	.27695198	.01800531
.92	.54	.08	.46	.27136635	
.73	.73	.27	.27	.25484137	.01652498
.90	.54	.10	.46	.24918886	
.72	.72	.28	.28	.23399413	.01519473
.88	.54	.12	.46	.22837055	
.71	.71	.29	.29	.21439017	.01398038
.86	.54	.14	.46	.20885795	
.70	.70	.30	.30	.19600323	.01285472

.84	.54	.16	.46	.19059834	
.69	.69	.31	.31	.17880182	.01179652
.82	.54	.18	.46	.17353981	
.68	.68	.32	.32	.16275011	.01078970
.80	.54	.20	.46	.15763128	
.67	.67	.33	.33	.14780863	.00982265
.78	.54	.22	.46	.14282251	
.66	.66	.34	.34	.13393510	.00888741
.76	.54	.24	.46	.12906409	
.65	.65	.35	.35	.12108502	.00797907
.74	.54	.26	.46	.11630752	
.64	.64	.36	.36	.10921229	.00709523
.72	.54	.28	.46	.10450519	
.63	.63	.37	.37	.09826969	.00623550
.70	.54	.30	.46	.09361039	
.62	.62	.38	.38	.08820940	.00540099
.68	.54	.32	.46	.08357736	
.61	.61	.39	.39	.07898341	.00459395
.66	.54	.34	.46	.07436130	
.60	.60	.40	.40	.07054387	.00381743
.64	.54	.36	.46	.06591838	
.59	.59	.41	.41	.06284342	.00307495
.62	.54	.38	.46	.05820575	
.58	.58	.42	.42	.05583550	.00237024
.60	.54	.40	.46	.05118159	
.57	.57	.43	.43	.04947454	.00170704
.58	.54	.42	.46	.04480511	
.56	.56	.44	.44	.04371622	.00108889
.56	.54	.44	.46	.03903658	
.55	.55	.45	.45	.03851761	.00051897
.54	.54	.46	.46	.03383731	
1.00	.52	.00	.48	.34308352	
.76	.76	.24	.24	.32500615	.01807737
.98	.52	.02	.48	.31686432	
.75	.75	.25	.25	.30033874	.01652558
.96	.52	.04	.48	.29216113	
.74	.74	.26	.26	.27695198	.01520915
.94	.52	.06	.48	.26891598	
.73	.73	.27	.27	.25484137	.01407461
.92	.52	.08	.48	.24707197	
.72	.72	.28	.28	.23399413	.01307784
.90	.52	.10	.48	.22657314	
.71	.71	.29	.29	.21439017	.01218297
.88	.52	.12	.48	.20736458	
.70	.70	.30	.30	.19600323	.01136135
.86	.52	.14	.48	.18939235	
.69	.69	.31	.31	.17880182	.01059053
.84	.52	.16	.48	.17260357	
.68	.68	.32	.32	.16275011	.00985346
.82	.52	.18	.48	.15694639	
.67	.67	.33	.33	.14780863	.00913776
.80	.52	.20	.48	.14236998	
.66	.66	.34	.34	.13393510	.00843488
.78	.52	.22	.48	.12882458	
.65	.65	.35	.35	.12108502	.00773956
.76	.52	.24	.48	.11626146	
.64	.64	.36	.36	.10921229	.00704917
.74	.52	.26	.48	.10463300	
.63	.63	.37	.37	.09826969	.00636331
.72	.52	.28	.48	.09389261	
.62	.62	.38	.38	.08820940	.00568321
.70	.52	.30	.48	.08399483	

.61	.61	.39	.39	.07898341	.00501142
.68	.52	.32	.48	.07489525	
.60	.60	.40	.40	.07054387	.00435137
.66	.52	.34	.48	.06655058	
.59	.59	.41	.41	.06284342	.00370715
.64	.52	.36	.48	.05891865	
.58	.58	.42	.42	.05583550	.00308314
.62	.52	.38	.48	.05195839	
.57	.57	.43	.43	.04947454	.00248385
.60	.52	.40	.48	.04562988	
.56	.56	.44	.44	.04371622	.00191365
.58	.52	.42	.48	.03989431	
.55	.55	.45	.45	.03851761	.00137670
.56	.52	.44	.48	.03471403	
.54	.54	.46	.46	.03383731	.00087671
.54	.52	.46	.48	.03005254	
.53	.53	.47	.47	.02963561	.00041693
.52	.52	.48	.48	.02587450	
1.00	.50	.00	.50	.31250000	
.75	.75	.25	.25	.30033874	.01216126
.98	.50	.02	.50	.28824005	
.74	.74	.26	.26	.27695198	.01128807
.96	.50	.04	.50	.26542080	
.73	.73	.27	.27	.25484137	.01057943
.94	.50	.06	.50	.24398404	
.72	.72	.28	.28	.23399413	.00998991
.92	.50	.08	.50	.22387279	
.71	.71	.29	.29	.21439017	.00948262
.90	.50	.10	.50	.20503125	
.70	.70	.30	.30	.19600323	.00902802
.88	.50	.12	.50	.18740480	
.69	.69	.31	.31	.17880182	.00860298
.86	.50	.14	.50	.17094005	
.68	.68	.32	.32	.16275011	.00818994
.84	.50	.16	.50	.15558480	
.67	.67	.33	.33	.14780863	.00777617
.82	.50	.18	.50	.14128804	
.66	.66	.34	.34	.13393510	.00735294
.80	.50	.20	.50	.12800000	
.65	.65	.35	.35	.12108502	.00691498
.78	.50	.22	.50	.11567205	
.64	.64	.36	.36	.10921229	.00645976
.76	.50	.24	.50	.10425680	
.63	.63	.37	.37	.09826969	.00598711
.74	.50	.26	.50	.09370804	
.62	.62	.38	.38	.08820940	.00549864
.72	.50	.28	.50	.08398079	
.61	.61	.39	.39	.07898341	.00499738
.70	.50	.30	.50	.07503125	
.60	.60	.40	.40	.07054387	.00448737
.68	.50	.32	.50	.06681680	
.59	.59	.41	.41	.06284342	.00397337
.66	.50	.34	.50	.05929605	
.58	.58	.42	.42	.05583550	.00346054
.64	.50	.36	.50	.05242879	
.57	.57	.43	.43	.04947454	.00295425
.62	.50	.38	.50	.04617605	
.56	.56	.44	.44	.04371622	.00245982
.60	.50	.40	.50	.04050000	
.55	.55	.45	.45	.03851761	.00198238
.58	.50	.42	.50	.03536404	
.54	.54	.46	.46	.03383731	.00152673

.56	.50	.44	.50	.03073280	
.53	.53	.47	.47	.02963561	.00109718
.54	.50	.46	.50	.02657205	
.52	.52	.48	.48	.02587450	.00069754
.52	.50	.48	.50	.02284879	
.51	.51	.49	.49	.02251782	.00033097
.50	.50	.50	.50	.01953125	

STOP

PROBABILITIES FOR CORRECT DECODING FOR CODE IV

TABLE 5.9

q_1	q_2	p_1	p_2	Pr(A),Pr(S)	Pr(A)-Pr(S)
1.00	1.00	.00	.00	1.00000000	
1.00	.98	.00	.02	.99615761	
.99	.99	.01	.01	.99573380	.00042381
.98	.98	.02	.02	.98382235	
1.00	.96	.00	.04	.98524201	
.98	.98	.02	.02	.98382235	.00141966
.98	.96	.02	.04	.96578812	
.97	.97	.03	.03	.96549345	.00029467
.96	.96	.04	.04	.94184624	
1.00	.94	.00	.06	.96812871	
.97	.97	.03	.03	.96549345	.00263526
.98	.94	.02	.06	.94280465	
.96	.96	.04	.04	.94184624	.00095841
.96	.94	.04	.06	.91405349	
.95	.95	.05	.05	.91386165	.00019184
.94	.94	.06	.06	.88241199	
1.00	.92	.00	.08	.94563870	
.96	.96	.04	.04	.94184624	.00379246
.98	.92	.02	.08	.91557407	
.95	.95	.05	.05	.91386165	.00171242
.96	.92	.04	.08	.88300570	
.94	.94	.06	.06	.88241199	.00059371
.94	.92	.06	.08	.84838121	
.93	.93	.07	.07	.84827007	.00011114
.92	.92	.08	.08	.81211754	
1.00	.90	.00	.10	.91854000	
.95	.95	.05	.05	.91386165	.00467835
.98	.90	.02	.10	.88475232	
.94	.94	.06	.06	.88241199	.00234033
.96	.90	.04	.10	.84925857	
.93	.93	.07	.07	.84827007	.00098850
.94	.90	.06	.10	.81242753	
.92	.92	.08	.08	.81211754	.00030999
.92	.90	.08	.10	.77460189	
.91	.91	.09	.09	.77455294	.00004895
.90	.90	.10	.10	.73609892	
1.00	.88	.00	.12	.88754914	
.94	.94	.06	.06	.88241199	.00513715
.98	.88	.02	.12	.85095044	
.93	.93	.07	.07	.84827007	.00268037
.96	.88	.04	.12	.81332878	
.92	.92	.08	.08	.81211754	.00121124
.94	.88	.06	.12	.77498373	
.91	.91	.09	.09	.77455294	.00043079
.92	.88	.08	.12	.73619261	
.90	.90	.10	.10	.73609892	.00009369
.90	.88	.10	.12	.69721135	
.89	.89	.11	.11	.69720923	.00000212
.88	.88	.12	.12	.65827504	
1.00	.86	.00	.14	.85333273	
.93	.93	.07	.07	.84827007	.00506266
.98	.86	.02	.14	.81473590	
.92	.92	.08	.08	.81211754	.00261836
.96	.86	.04	.14	.77569532	
.91	.91	.09	.09	.77455294	.00114238
.94	.86	.06	.14	.73644994	
.90	.90	.10	.10	.73609892	.00035102
.92	.86	.08	.14	.69722008	
.89	.89	.11	.11	.69720923	.00001085

.90	.86	.10	.14	.65820802	
.88	.88	.12	.12	.65827504	-.00006702
.88	.86	.12	.14	.61959865	
.87	.87	.13	.13	.61963081	-.00003216
.86	.86	.14	.14	.58155996	
1.00	.84	.00	.16	.81650903	
.92	.92	.08	.08	.81211754	.00439149
.98	.84	.02	.16	.77663402	
.91	.91	.09	.09	.77455294	.00208108
.96	.84	.04	.16	.73680054	
.90	.90	.10	.10	.73609892	.00070162
.94	.84	.06	.16	.69719482	
.89	.89	.11	.11	.69720923	-.00001441
.92	.84	.08	.16	.65798764	
.88	.88	.12	.12	.65827504	-.00028740
.90	.84	.10	.16	.61933485	
.87	.87	.13	.13	.61963081	-.00029596
.88	.84	.12	.16	.58137774	
.86	.86	.14	.14	.58155996	-.00018222
.86	.84	.14	.16	.54424365	
.85	.85	.15	.15	.54429982	-.00005617
.84	.84	.16	.16	.50804643	
1.00	.82	.00	.18	.77764942	
.91	.91	.09	.09	.77455294	.00309648
.98	.82	.02	.18	.73712935	
.90	.90	.10	.10	.73609892	.00103043
.96	.82	.04	.18	.69705143	
.89	.89	.11	.11	.69720923	-.00015780
.94	.82	.06	.18	.65755646	
.88	.88	.12	.12	.65827504	-.00071858
.92	.82	.08	.18	.61877253	
.87	.87	.13	.13	.61963081	-.00085828
.90	.82	.10	.18	.58081541	
.86	.86	.14	.14	.58155996	-.00074455
.88	.82	.12	.18	.54378901	
.85	.85	.15	.15	.54429982	-.00051081
.86	.82	.14	.18	.50778570	
.84	.84	.16	.16	.50804643	-.00026073
.84	.82	.16	.18	.47288688	
.83	.83	.17	.17	.47295884	-.00007196
.82	.82	.18	.18	.43916322	
1.00	.80	.00	.20	.73728000	
.90	.90	.10	.10	.73609892	.00118108
.98	.80	.02	.20	.69666689	
.89	.89	.11	.11	.69720923	-.00054234
.96	.80	.04	.20	.65682076	
.88	.88	.12	.12	.65827504	-.00145428
.94	.80	.06	.20	.61784350	
.87	.87	.13	.13	.61963081	-.00178731
.92	.80	.08	.20	.57982667	
.86	.86	.14	.14	.58155996	-.00173329
.90	.80	.10	.20	.54285188	
.85	.85	.15	.15	.54429982	-.00144794
.88	.80	.12	.20	.50699110	
.84	.84	.16	.16	.50804643	-.00105533
.86	.80	.14	.20	.47230698	
.83	.83	.17	.17	.47295884	-.00065186
.84	.80	.16	.20	.43885329	
.82	.82	.18	.18	.43916322	-.00030993
.82	.80	.18	.20	.40667520	
.81	.81	.19	.19	.40675645	-.00008125

.80	.80	.20	.20	.37580964	
1.00	.78	.00	.22	.69588306	
.89	.89	.11	.11	.69720923	-.00132617
.98	.78	.02	.22	.65565354	
.88	.88	.12	.12	.65827504	-.00262150
.96	.78	.04	.22	.61644828	
.87	.87	.13	.13	.61963081	-.00318253
.94	.78	.06	.22	.57833612	
.86	.86	.14	.14	.58155996	-.00322384
.92	.78	.08	.22	.54137771	
.85	.85	.15	.15	.54429982	-.00292211
.90	.78	.10	.22	.50562573	
.84	.84	.16	.16	.50804643	-.00242070
.88	.78	.12	.22	.47112521	
.83	.83	.17	.17	.47295884	-.00183363
.86	.78	.14	.22	.43791375	
.82	.82	.18	.18	.43916322	-.00124947
.84	.78	.16	.22	.40602192	
.81	.81	.19	.19	.40675645	-.00073453
.82	.78	.18	.22	.37547346	
.80	.80	.20	.20	.37580964	-.00033618
.80	.78	.20	.22	.34628562	
.79	.79	.21	.21	.34637112	-.00008550
.78	.78	.22	.22	.31846939	
1.00	.76	.00	.24	.65389865	
.88	.88	.12	.12	.65827504	-.00437639
.98	.76	.02	.24	.61445947	
.87	.87	.13	.13	.61963081	-.00517134
.96	.76	.04	.24	.57624190	
.86	.86	.14	.14	.58155996	-.00531806
.94	.76	.06	.24	.53928711	
.85	.85	.15	.15	.54429982	-.00501271
.92	.76	.08	.24	.50362981	
.84	.84	.16	.16	.50804643	-.00441662
.90	.76	.10	.24	.46929853	
.83	.83	.17	.17	.47295884	-.00366031
.88	.76	.12	.24	.43631576	
.82	.82	.18	.18	.43916322	-.00284746
.86	.76	.14	.24	.40469831	
.81	.81	.19	.19	.40675645	-.00205814
.84	.76	.16	.24	.37445746	
.80	.80	.20	.20	.37580964	-.00135218
.82	.76	.18	.24	.34559923	
.79	.79	.21	.21	.34637112	-.00077189
.80	.76	.20	.24	.31812462	
.78	.78	.22	.22	.31846939	-.00034477
.78	.76	.22	.24	.29202983	
.77	.77	.23	.23	.29211570	-.00008587
.76	.76	.24	.24	.26730643	
1.00	.74	.00	.26	.61172615	
.87	.87	.13	.13	.61963081	-.00790466
.98	.74	.02	.26	.57341938	
.86	.86	.14	.14	.58155996	-.00814058
.96	.74	.04	.26	.53647891	
.85	.85	.15	.15	.54429982	-.00782091
.94	.74	.06	.26	.50092289	
.84	.84	.16	.16	.50804643	-.00712354
.92	.74	.08	.26	.46676462	
.83	.83	.17	.17	.47295884	-.00619422
.90	.74	.10	.26	.43401268	
.82	.82	.18	.18	.43916322	-.00515054

.88	.74	.12	.26	.40267108	
.81	.81	.19	.19	.40675645	-.00408537
.86	.74	.14	.26	.37273954	
.80	.80	.20	.20	.37580964	-.00307010
.84	.74	.16	.26	.34421356	
.79	.79	.21	.21	.34637112	-.00215756
.82	.74	.18	.26	.31708475	
.78	.78	.22	.22	.31846939	-.00138464
.80	.74	.20	.26	.29134093	
.77	.77	.23	.23	.29211570	-.00077477
.78	.74	.22	.26	.26696633	
.76	.76	.24	.24	.26730643	-.00034010
.76	.74	.24	.26	.24394178	
.75	.75	.25	.25	.24402522	-.00008344
.74	.74	.26	.26	.22224496	
1.00	.72	.00	.28	.56972574	
.86	.86	.14	.14	.58155996	-.01183422
.98	.72	.02	.28	.53283401	
.85	.85	.15	.15	.54429982	-.01146581
.96	.72	.04	.28	.49740713	
.84	.84	.16	.16	.50804643	-.01063930
.94	.72	.06	.28	.46344455	
.83	.83	.17	.17	.47295884	-.00951429
.92	.72	.08	.28	.43094216	
.82	.82	.18	.18	.43916322	-.00822106
.90	.72	.10	.28	.39989233	
.81	.81	.19	.19	.40675645	-.00686412
.88	.72	.12	.28	.37028412	
.80	.80	.20	.20	.37580964	-.00552552
.86	.72	.14	.28	.34210348	
.79	.79	.21	.21	.34637112	-.00426764
.84	.72	.16	.28	.31533331	
.78	.78	.22	.22	.31846939	-.00313608
.82	.72	.18	.28	.28995368	
.77	.77	.23	.23	.29211570	-.00216202
.80	.72	.20	.28	.26594195	
.76	.76	.24	.24	.26730643	-.00136448
.78	.72	.22	.28	.24327294	
.75	.75	.25	.25	.24402522	-.00075228
.76	.72	.24	.28	.22191908	
.74	.74	.26	.26	.22224496	-.00032588
.74	.72	.26	.28	.20185052	
.73	.73	.27	.27	.20192954	-.00007902
.72	.72	.28	.28	.18303540	
1.00	.70	.00	.30	.52822000	
.85	.85	.15	.15	.54429982	-.01607982
.98	.70	.02	.30	.49297129	
.84	.84	.16	.16	.50804643	-.01507514
.96	.70	.04	.30	.45924611	
.83	.83	.17	.17	.47295884	-.01371273
.94	.70	.06	.30	.42702892	
.82	.82	.18	.18	.43916322	-.01213430
.92	.70	.08	.30	.39630170	
.81	.81	.19	.19	.40675645	-.01045475
.90	.70	.10	.30	.36704398	
.80	.80	.20	.20	.37580964	-.00876566
.88	.70	.12	.30	.33923303	
.79	.79	.21	.21	.34637112	-.00713809
.86	.70	.14	.30	.31284392	
.78	.78	.22	.22	.31846939	-.00562547
.84	.70	.16	.30	.28784970	

.77	.77	.23	.23	.29211570	-.00426600
.82	.70	.18	.30	.26422148	
.76	.76	.24	.24	.26730643	-.00308495
.80	.70	.20	.30	.24192860	
.75	.75	.25	.25	.24402522	-.00209662
.78	.70	.22	.30	.22093865	
.74	.74	.26	.26	.22224496	-.00130631
.76	.70	.24	.30	.20121768	
.73	.73	.27	.27	.20192954	-.00071186
.74	.70	.26	.30	.18273033	
.72	.72	.28	.28	.18303540	-.00030507
.72	.70	.28	.30	.16543986	
.71	.71	.29	.29	.16551309	-.00007323
.70	.70	.30	.30	.14930834	
1.00	.68	.00	.32	.48749538	
.84	.84	.16	.16	.50804643	-.02055105
.98	.68	.02	.32	.45406783	
.83	.83	.17	.17	.47295884	-.01889101
.96	.68	.04	.32	.42218829	
.82	.82	.18	.18	.43916322	-.01697493
.94	.68	.06	.32	.39182952	
.81	.81	.19	.19	.40675645	-.01492693
.92	.68	.08	.32	.36296269	
.80	.80	.20	.20	.37580964	-.01284695
.90	.68	.10	.32	.33555743	
.79	.79	.21	.21	.34637112	-.01081369
.88	.68	.12	.32	.30958190	
.78	.78	.22	.22	.31846939	-.00888749
.86	.68	.14	.32	.28500294	
.77	.77	.23	.23	.29211570	-.00711276
.84	.68	.16	.32	.26178614	
.76	.76	.24	.24	.26730643	-.00552029
.82	.68	.18	.32	.23989591	
.75	.75	.25	.25	.24402522	-.00412931
.80	.68	.20	.32	.21929560	
.74	.74	.26	.26	.22224496	-.00294936
.78	.68	.22	.32	.19994758	
.73	.73	.27	.27	.20192954	-.00198196
.76	.68	.24	.32	.18181335	
.72	.72	.28	.28	.18303540	-.00122205
.74	.68	.26	.32	.16485358	
.71	.71	.29	.29	.16551309	-.00065951
.72	.68	.28	.32	.14902826	
.70	.70	.30	.30	.14930834	-.00028008
.70	.68	.30	.32	.13429676	
.69	.69	.31	.31	.13436340	-.00006664
.68	.68	.32	.32	.12061795	
1.00	.66	.00	.34	.44780377	
.83	.83	.17	.17	.47295884	-.02515507
.98	.66	.02	.34	.41633026	
.82	.82	.18	.18	.43916322	-.02283296
.96	.66	.04	.34	.38640025	
.81	.81	.19	.19	.40675645	-.02035620
.94	.66	.06	.34	.35797770	
.80	.80	.20	.20	.37580964	-.01783194
.92	.66	.08	.34	.33102568	
.79	.79	.21	.21	.34637112	-.01534544
.90	.66	.10	.34	.30550641	
.78	.78	.22	.22	.31846939	-.01296298
.88	.66	.12	.34	.28138142	

.77	.77	.23	.23	.29211570	-.01073428
.86	.66	.14	.34	.25861151	
.76	.76	.24	.24	.26730643	-.00869492
.84	.66	.16	.34	.23715692	
.75	.75	.25	.25	.24402522	-.00686830
.82	.66	.18	.34	.21697731	
.74	.74	.26	.26	.22224496	-.00526765
.80	.66	.20	.34	.19803188	
.73	.73	.27	.27	.20192954	-.00389766
.78	.66	.22	.34	.18027943	
.72	.72	.28	.28	.18303540	-.00275597
.76	.66	.24	.34	.16367843	
.71	.71	.29	.29	.16551309	-.00183466
.74	.66	.26	.34	.14818706	
.70	.70	.30	.30	.14930834	-.00112128
.72	.66	.28	.34	.13376334	
.69	.69	.31	.31	.13436340	-.00060006
.70	.66	.30	.34	.12036513	
.68	.68	.32	.32	.12061795	-.00025282
.68	.66	.32	.34	.10795024	
.67	.67	.33	.33	.10800994	-.00005970
.66	.66	.34	.34	.09647648	
1.00	.64	.00	.36	.40936407	
.82	.82	.18	.18	.43916322	-.02979915
.98	.64	.02	.36	.37993653	
.81	.81	.19	.19	.40675645	-.02681992
.96	.64	.04	.36	.35202387	
.80	.80	.20	.20	.37580964	-.02378577
.94	.64	.06	.36	.32558361	
.79	.79	.21	.21	.34637112	-.02078751
.92	.64	.08	.36	.30057313	
.78	.78	.22	.22	.31846939	-.01789626
.90	.64	.10	.36	.27694949	
.77	.77	.23	.23	.29211570	-.01516621
.88	.64	.12	.36	.25466956	
.76	.76	.24	.24	.26730643	-.01263687
.86	.64	.14	.36	.23369010	
.75	.75	.25	.25	.24402522	-.01033512
.84	.64	.16	.36	.21396776	
.74	.74	.26	.26	.22224496	-.00827720
.82	.64	.18	.36	.19545914	
.73	.73	.27	.27	.20192954	-.00647040
.80	.64	.20	.36	.17812088	
.72	.72	.28	.28	.18303540	-.00491452
.78	.64	.22	.36	.16190962	
.71	.71	.29	.29	.16551309	-.00360347
.76	.64	.24	.36	.14678213	
.70	.70	.30	.30	.14930834	-.00252621
.74	.64	.26	.36	.13269534	
.69	.69	.31	.31	.13436340	-.00166806
.72	.64	.28	.36	.11960638	
.68	.68	.32	.32	.12061795	-.00101157
.70	.64	.30	.36	.10747262	
.67	.67	.33	.33	.10800994	-.00053732
.68	.64	.32	.36	.09625174	
.66	.66	.34	.34	.09647648	-.00022474
.66	.64	.34	.36	.08590174	
.65	.65	.35	.35	.08595443	-.00005269
.64	.64	.36	.36	.07638104	
1.00	.62	.00	.38	.37236366	
.81	.81	.19	.19	.40675645	-.03439279

.98	.62	.02	.38	.34503737	
.80	.80	.20	.20	.37580964	-.03077227
.96	.62	.04	.38	.31917744	
.79	.79	.21	.21	.34637112	-.02719368
.94	.62	.06	.38	.29473728	
.78	.78	.22	.22	.31846939	-.02373211
.92	.62	.08	.38	.27167045	
.77	.77	.23	.23	.29211570	-.02044525
.90	.62	.10	.38	.24993075	
.76	.76	.24	.24	.26730643	-.01737568
.88	.62	.12	.38	.22947222	
.75	.75	.25	.25	.24402522	-.01455300
.86	.62	.14	.38	.21024918	
.74	.74	.26	.26	.22224496	-.01199578
.84	.62	.16	.38	.19221623	
.73	.73	.27	.27	.20192954	-.00971331
.82	.62	.18	.38	.17532836	
.72	.72	.28	.28	.18303540	-.00770704
.80	.62	.20	.38	.15954092	
.71	.71	.29	.29	.16551309	-.00597217
.78	.62	.22	.38	.14480966	
.70	.70	.30	.30	.14930834	-.00449868
.76	.62	.24	.38	.13109077	
.69	.69	.31	.31	.13436340	-.00327263
.74	.62	.26	.38	.11834093	
.68	.68	.32	.32	.12061795	-.00227702
.72	.62	.28	.38	.10651733	
.67	.67	.33	.33	.10800994	-.00149261
.70	.62	.30	.38	.09557768	
.66	.66	.34	.34	.09647648	-.00089879
.68	.62	.32	.38	.08548029	
.65	.65	.35	.35	.08595443	-.00047414
.66	.62	.34	.38	.07618405	
.64	.64	.36	.36	.07638104	-.00019699
.64	.62	.36	.38	.06764852	
.63	.63	.37	.37	.06769441	-.00004588
.62	.62	.38	.38	.05983392	
1.00	.60	.00	.40	.33696000	
.80	.80	.20	.20	.37580964	-.03884964
.98	.60	.02	.40	.31175748	
.79	.79	.21	.21	.34637112	-.03461364
.96	.60	.04	.40	.28795701	
.78	.78	.22	.22	.31846939	-.03051238
.94	.60	.06	.40	.26550960	
.77	.77	.23	.23	.29211570	-.02660610
.92	.60	.08	.40	.24436678	
.76	.76	.24	.24	.26730643	-.02293965
.90	.60	.10	.40	.22448068	
.75	.75	.25	.25	.24402522	-.01954454
.88	.60	.12	.40	.20580393	
.74	.74	.26	.26	.22224496	-.01644103
.86	.60	.14	.40	.18828983	
.73	.73	.27	.27	.20192954	-.01363971
.84	.60	.16	.40	.17189227	
.72	.72	.28	.28	.18303540	-.01114313
.82	.60	.18	.40	.15656578	
.71	.71	.29	.29	.16551309	-.00894731
.80	.60	.20	.40	.14226554	
.70	.70	.30	.30	.14930834	-.00704280
.78	.60	.22	.40	.12894742	
.69	.69	.31	.31	.13436340	-.00541598

.76	.60	.24	.40	.11656797	
.68	.68	.32	.32	.12061795	-.00404998
.74	.60	.26	.40	.10508447	
.67	.67	.33	.33	.10800994	-.00292547
.72	.60	.28	.40	.09445493	
.66	.66	.34	.34	.09647648	-.00202154
.70	.60	.30	.40	.08463813	
.65	.65	.35	.35	.08595443	-.00131630
.68	.60	.32	.40	.07559360	
.64	.64	.36	.36	.07638104	-.00078744
.66	.60	.34	.40	.06728168	
.63	.63	.37	.37	.06769441	-.00041273
.64	.60	.36	.40	.05966353	
.62	.62	.38	.38	.05983392	-.00017038
.62	.60	.38	.40	.05270116	
.61	.61	.39	.39	.05274059	-.00003943
.60	.60	.40	.40	.04635740	
1.00	.58	.00	.42	.30328209	
.79	.79	.21	.21	.34637112	-.04308903
.98	.58	.02	.42	.28019701	
.78	.78	.22	.22	.31846939	-.03827238
.96	.58	.04	.42	.25843742	
.77	.77	.23	.23	.29211570	-.03367828
.94	.58	.06	.42	.23795344	
.76	.76	.24	.24	.26730643	-.02935299
.92	.58	.08	.42	.21869599	
.75	.75	.25	.25	.24402522	-.02532923
.90	.58	.10	.42	.20061682	
.74	.74	.26	.26	.22224496	-.02162814
.88	.58	.12	.42	.18366848	
.73	.73	.27	.27	.20192954	-.01826106
.86	.58	.14	.42	.16780437	
.72	.72	.28	.28	.18303540	-.01523103
.84	.58	.16	.42	.15297869	
.71	.71	.29	.29	.16551309	-.01253440
.82	.58	.18	.42	.13914654	
.70	.70	.30	.30	.14930834	-.01016180
.80	.58	.20	.42	.12626385	
.69	.69	.31	.31	.13436340	-.00809955
.78	.58	.22	.42	.11428742	
.68	.68	.32	.32	.12061795	-.00633053
.76	.58	.24	.42	.10317493	
.67	.67	.33	.33	.10800994	-.00483501
.74	.58	.26	.42	.09288495	
.66	.66	.34	.34	.09647648	-.00359152
.72	.58	.28	.42	.08337695	
.65	.65	.35	.35	.08595443	-.00257747
.70	.58	.30	.42	.07461131	
.64	.64	.36	.36	.07638104	-.00176973
.68	.58	.32	.42	.06654930	
.63	.63	.37	.37	.06769441	-.00114511
.66	.58	.34	.42	.05915313	
.62	.62	.38	.38	.05983392	-.00068078
.64	.58	.36	.42	.05238597	
.61	.61	.39	.39	.05274059	-.00035462
.62	.58	.38	.42	.04621190	
.60	.60	.40	.40	.04635740	-.00014549
.60	.58	.40	.42	.04059598	
.59	.59	.41	.41	.04062944	-.00003346
.58	.58	.42	.42	.03550420	
1.00	.56	.00	.44	.27143208	

.78	.78	.22	.22	.31846939	-.04703731
.98	.56	.02	.44	.25043288	
.77	.77	.23	.23	.29211570	-.04168282
.96	.56	.04	.44	.23067355	
.76	.76	.24	.24	.26730643	-.03663288
.94	.56	.06	.44	.21210459	
.75	.75	.25	.25	.24402522	-.03192063
.92	.56	.08	.44	.19467748	
.74	.74	.26	.26	.22224496	-.02756748
.90	.56	.10	.44	.17834468	
.73	.73	.27	.27	.20192954	-.02358486
.88	.56	.12	.44	.16305965	
.72	.72	.28	.28	.18303540	-.01997575
.86	.56	.14	.44	.14877681	
.71	.71	.29	.29	.16551309	-.01673628
.84	.56	.16	.44	.13545162	
.70	.70	.30	.30	.14930834	-.01385672
.82	.56	.18	.44	.12304049	
.69	.69	.31	.31	.13436340	-.01132291
.80	.56	.20	.44	.11150084	
.68	.68	.32	.32	.12061795	-.00911711
.78	.56	.22	.44	.10079108	
.67	.67	.33	.33	.10800994	-.00721886
.76	.56	.24	.44	.09087062	
.66	.66	.34	.34	.09647648	-.00560585
.74	.56	.26	.44	.08169988	
.65	.65	.35	.35	.08595443	-.00425454
.72	.56	.28	.44	.07324027	
.64	.64	.36	.36	.07638104	-.00314077
.70	.56	.30	.44	.06545419	
.63	.63	.37	.37	.06769441	-.00224021
.68	.56	.32	.44	.05830506	
.62	.62	.38	.38	.05983392	-.00152885
.66	.56	.34	.44	.05175730	
.61	.61	.39	.39	.05274059	-.00098329
.64	.56	.36	.44	.04577634	
.60	.60	.40	.40	.04635740	-.00058106
.62	.56	.38	.44	.04032860	
.59	.59	.41	.41	.04062944	-.00030084
.60	.56	.40	.44	.03538152	
.58	.58	.42	.42	.03550420	-.00012268
.58	.56	.42	.44	.03090357	
.57	.57	.43	.43	.03093161	-.00002804
.56	.56	.44	.44	.02686420	
1.00	.54	.00	.46	.24148679	
.77	.77	.23	.23	.29211570	-.05062891
.98	.54	.02	.46	.22252012	
.76	.76	.24	.24	.26730643	-.04478631
.96	.54	.04	.46	.20470153	
.75	.75	.25	.25	.24402522	-.03932369
.94	.54	.06	.46	.18798290	
.74	.74	.26	.26	.22224496	-.03426206
.92	.54	.08	.46	.17231718	
.73	.73	.27	.27	.20192954	-.02961236
.90	.54	.10	.46	.15765844	
.72	.72	.28	.28	.18303540	-.02537696
.88	.54	.12	.46	.14396183	
.71	.71	.29	.29	.16551309	-.02155126
.86	.54	.14	.46	.13118359	
.70	.70	.30	.30	.14930834	-.01812475

.84	.54	.16	.46	.11928106	
.69	.69	.31	.31	.13436340	-.01508234
.82	.54	.18	.46	.10821263	
.68	.68	.32	.32	.12061795	-.01240532
.80	.54	.20	.46	.09793779	
.67	.67	.33	.33	.10800994	-.01007215
.78	.54	.22	.46	.08841706	
.66	.66	.34	.34	.09647648	-.00805941
.76	.54	.24	.46	.07961208	
.65	.65	.35	.35	.08595443	-.00634235
.74	.54	.26	.46	.07148550	
.64	.64	.36	.36	.07638104	-.00489554
.72	.54	.28	.46	.06400105	
.63	.63	.37	.37	.06769441	-.00369336
.70	.54	.30	.46	.05712351	
.62	.62	.38	.38	.05983392	-.00271041
.68	.54	.32	.46	.05081870	
.61	.61	.39	.39	.05274059	-.00192189
.66	.54	.34	.46	.04505348	
.60	.60	.40	.40	.04635740	-.00130391
.64	.54	.36	.46	.03979576	
.59	.59	.41	.41	.04062944	-.00083368
.62	.54	.38	.46	.03501447	
.58	.58	.42	.42	.03550420	-.00048973
.60	.54	.40	.46	.03067957	
.57	.57	.43	.43	.03093161	-.00025204
.58	.54	.42	.46	.02676204	
.56	.56	.44	.44	.02686420	-.00010215
.56	.54	.44	.46	.02323388	
.55	.55	.45	.45	.02325710	-.00002321
.54	.54	.46	.46	.02006813	
1.00	.52	.00	.48	.21349918	
.76	.76	.24	.24	.26730643	-.05380725
.98	.52	.02	.48	.19649323	
.75	.75	.25	.25	.24402522	-.04753199
.96	.52	.04	.48	.18053990	
.74	.74	.26	.26	.22224496	-.04170506
.94	.52	.06	.48	.16559322	
.73	.73	.27	.27	.20192954	-.03633632
.92	.52	.08	.48	.15160838	
.72	.72	.28	.28	.18303540	-.03142702
.90	.52	.10	.48	.13854173	
.71	.71	.29	.29	.16551309	-.02697136
.88	.52	.12	.48	.12635073	
.70	.70	.30	.30	.14930834	-.02295761
.86	.52	.14	.48	.11499401	
.69	.69	.31	.31	.13436340	-.01936939
.84	.52	.16	.48	.10443132	
.68	.68	.32	.32	.12061795	-.01618663
.82	.52	.18	.48	.09462352	
.67	.67	.33	.33	.10800994	-.01338642
.80	.52	.20	.48	.08553257	
.66	.66	.34	.34	.09647648	-.01094391
.78	.52	.22	.48	.07712154	
.65	.65	.35	.35	.08595443	-.00883289
.76	.52	.24	.48	.06935458	
.64	.64	.36	.36	.07638104	-.00702646
.74	.52	.26	.48	.06219693	
.63	.63	.37	.37	.06769441	-.00549748
.72	.52	.28	.48	.05561490	
.62	.62	.38	.38	.05983392	-.00421902

.70	.52	.30	.48	.04957585	
.61	.61	.39	.39	.05274059	-.00316474
.68	.52	.32	.48	.04404822	
.60	.60	.40	.40	.04635740	-.00230918
.66	.52	.34	.48	.03900146	
.59	.59	.41	.41	.04062944	-.00162798
.64	.52	.36	.48	.03440607	
.58	.58	.42	.42	.03550420	-.00109813
.62	.52	.38	.48	.03023360	
.57	.57	.43	.43	.03093161	-.00069801
.60	.52	.40	.48	.02645658	
.56	.56	.44	.44	.02686420	-.00040761
.58	.52	.42	.48	.02304857	
.55	.55	.45	.45	.02325710	-.00020852
.56	.52	.44	.48	.01998413	
.54	.54	.46	.46	.02006813	-.00008400
.54	.52	.46	.48	.01723879	
.53	.53	.47	.47	.01725776	-.00001896
.52	.52	.48	.48	.01478909	
1.00	.50	.00	.50	.18750000	
.75	.75	.25	.25	.24402522	-.05652522
.98	.50	.02	.50	.17236755	
.74	.74	.26	.26	.22224496	-.04987741
.96	.50	.04	.50	.15819080	
.73	.73	.27	.27	.20192954	-.04373874
.94	.50	.06	.50	.14492653	
.72	.72	.28	.28	.18303540	-.03810887
.92	.50	.08	.50	.13253269	
.71	.71	.29	.29	.16551309	-.03298040
.90	.50	.10	.50	.12096843	
.70	.70	.30	.30	.14930834	-.02833991
.88	.50	.12	.50	.11019402	
.69	.69	.31	.31	.13436340	-.02416938
.86	.50	.14	.50	.10017087	
.68	.68	.32	.32	.12061795	-.02044708
.84	.50	.16	.50	.09086152	
.67	.67	.33	.33	.10800994	-.01714842
.82	.50	.18	.50	.08222964	
.66	.66	.34	.34	.09647648	-.01424683
.80	.50	.20	.50	.07424000	
.65	.65	.35	.35	.08595443	-.01171443
.78	.50	.22	.50	.06685844	
.64	.64	.36	.36	.07638104	-.00952260
.76	.50	.24	.50	.06005191	
.63	.63	.37	.37	.06769441	-.00764249
.74	.50	.26	.50	.05378842	
.62	.62	.38	.38	.05983392	-.00604550
.72	.50	.28	.50	.04803701	
.61	.61	.39	.39	.05274059	-.00470358
.70	.50	.30	.50	.04276781	
.60	.60	.40	.40	.04635740	-.00358958
.68	.50	.32	.50	.03795194	
.59	.59	.41	.41	.04062944	-.00267750
.66	.50	.34	.50	.03356156	
.58	.58	.42	.42	.03550420	-.00194264
.64	.50	.36	.50	.02956984	
.57	.57	.43	.43	.03093161	-.00136177
.62	.50	.38	.50	.02595094	
.56	.56	.44	.44	.02686420	-.00091326
.60	.50	.40	.50	.02268000	
.55	.55	.45	.45	.02325710	-.00057710
.58	.50	.42	.50	.01973313	

.54	.54	.46	.46	.02006813	-.00033499
.56	.50	.44	.50	.01708743	
.53	.53	.47	.47	.01725776	-.00017032
.54	.50	.46	.50	.01472091	
.52	.52	.48	.48	.01478909	-.00006818
.52	.50	.48	.50	.01261253	
.51	.51	.49	.49	.01262783	-.00001529
.50	.50	.50	.50	.01074218	

STOP

PROBABILITIES FOR CORRECT DECODING FOR CODE V

TABLE 5.10

q_1	q_2	p_1	p_2	Pr(A), Pr(B)	Pr(A)-Pr(B)
1.00	1.00	.00	.00	1.00000000	
1.00	.98	.00	.02	.99592456	
.99	.99	.01	.01	.99573368	.00019088
.98	.98	.02	.02	.98382231	
1.00	.96	.00	.04	.98446293	
.98	.98	.02	.02	.98382231	.00064062
.98	.96	.02	.04	.96562606	
.97	.97	.03	.03	.96549337	.00013269
.96	.96	.04	.04	.94184618	
1.00	.94	.00	.06	.96668775	
.97	.97	.03	.03	.96549337	.00119438
.98	.94	.02	.06	.94227900	
.96	.96	.04	.04	.94184618	.00043282
.96	.94	.04	.06	.91394800	
.95	.95	.05	.05	.91386156	.00008644
.94	.94	.06	.06	.88241187	
1.00	.92	.00	.08	.94357743	
.96	.96	.04	.04	.94184618	.00173125
.98	.92	.02	.08	.91463937	
.95	.95	.05	.05	.91386156	.00077781
.96	.92	.04	.08	.88268043	
.94	.94	.06	.06	.88241187	.00026856
.94	.92	.06	.08	.84832006	
.93	.93	.07	.07	.84827000	.00005006
.92	.92	.08	.08	.81211750	
1.00	.90	.00	.10	.91602187	
.95	.95	.05	.05	.91386156	.00216031
.98	.90	.02	.10	.88348575	
.94	.94	.06	.06	.88241187	.00107388
.96	.90	.04	.10	.84872081	
.93	.93	.07	.07	.84827000	.00045081
.94	.90	.06	.10	.81225806	
.92	.92	.08	.08	.81211750	.00014056
.92	.90	.08	.10	.77457500	
.91	.91	.09	.09	.77455287	.00002213
.90	.90	.10	.10	.73609881	
1.00	.88	.00	.12	.88482756	
.94	.94	.06	.06	.88241187	.00241569
.98	.88	.02	.12	.84952212	
.93	.93	.07	.07	.84827000	.00125212
.96	.88	.04	.12	.81267975	
.92	.92	.08	.08	.81211750	.00056225
.94	.88	.06	.12	.77475168	
.91	.91	.09	.09	.77455287	.00019881
.92	.88	.08	.12	.73614193	
.90	.90	.10	.10	.73609881	.00004312
.90	.88	.10	.12	.69721018	
.89	.89	.11	.11	.69720918	.00000100
.88	.88	.12	.12	.65827500	
1.00	.86	.00	.14	.85072306	
.93	.93	.07	.07	.84827000	.00245306
.98	.86	.02	.14	.81338212	
.92	.92	.08	.08	.81211750	.00126462
.96	.86	.04	.14	.77510500	
.91	.91	.09	.09	.77455287	.00055213
.94	.86	.06	.14	.73627106	
.90	.90	.10	.10	.73609881	.00017225
.92	.86	.08	.14	.69721812	
.89	.89	.11	.11	.69720918	.00000894
.90	.86	.10	.14	.65824556	
.88	.88	.12	.12	.65827500	-.00002944
.88	.86	.12	.14	.61961638	

.87	.87	.13	.13	.61963081	-.00001443
.86	.86	.14	.14	.58155995	
1.00	.84	.00	.16	.81436368	
.92	.92	.08	.08	.81211750	.00224618
.98	.84	.02	.16	.77563406	
.91	.91	.09	.09	.77455287	.00108119
.96	.84	.04	.16	.73648581	
.90	.90	.10	.10	.73609881	.00038700
.94	.84	.06	.16	.69723393	
.89	.89	.11	.11	.69720918	.00002475
.92	.84	.08	.16	.65815725	
.88	.88	.12	.12	.65827500	-.00011775
.90	.84	.10	.16	.61950093	
.87	.87	.13	.13	.61963081	-.00012988
.88	.84	.12	.16	.58147854	
.86	.86	.14	.14	.58155995	-.00008141
.86	.84	.14	.16	.54427457	
.85	.85	.15	.15	.54429981	-.00002524
.84	.84	.16	.16	.50804642	
1.00	.82	.00	.18	.77633643	
.91	.91	.09	.09	.77455287	.00178356
.98	.82	.02	.18	.73678487	
.90	.90	.10	.10	.73609881	.00068606
.96	.82	.04	.18	.69725681	
.89	.89	.11	.11	.69720918	.00004763
.94	.82	.06	.18	.65800975	
.88	.88	.12	.12	.65827500	-.00026525
.92	.82	.08	.18	.61926991	
.87	.87	.13	.13	.61963081	-.00036090
.90	.82	.10	.18	.58123425	
.86	.86	.14	.14	.58155995	-.00032570
.88	.82	.12	.18	.54407260	
.85	.85	.15	.15	.54429981	-.00022721
.86	.82	.14	.18	.50792956	
.84	.84	.16	.16	.50804642	-.00011686
.84	.82	.16	.18	.47292647	
.83	.83	.17	.17	.47295883	-.00003236
.82	.82	.18	.18	.43916321	
1.00	.80	.00	.20	.73716475	
.90	.90	.10	.10	.73609881	.00106594
.98	.80	.02	.20	.69728456	
.89	.89	.11	.11	.69720918	.00007538
.96	.80	.04	.20	.65780181	
.88	.88	.12	.12	.65827500	-.00047319
.94	.80	.06	.20	.61892270	
.87	.87	.13	.13	.61963081	-.00070811
.92	.80	.08	.20	.58082683	
.86	.86	.14	.14	.58155995	-.00073312
.90	.80	.10	.20	.54366855	
.85	.85	.15	.15	.54429981	-.00063126
.88	.80	.12	.20	.50757896	
.84	.84	.16	.16	.50804642	-.00046746
.86	.80	.14	.20	.47266760	
.83	.83	.17	.17	.47295883	-.00029123
.84	.80	.16	.20	.43902410	
.82	.82	.18	.18	.43916321	-.00013911
.82	.80	.18	.20	.40671991	
.81	.81	.19	.19	.40675645	-.00003654
.80	.80	.20	.20	.37589962	
1.00	.78	.00	.22	.69731243	
.89	.89	.11	.11	.69720918	.00010325
.98	.78	.02	.22	.65753018	

.88	.88	.12	.12	.65827500	-.00074482
.96	.78	.04	.22	.61845736	
.87	.87	.13	.13	.61963081	-.00117345
.94	.78	.06	.22	.58025515	
.86	.86	.14	.14	.58155995	-.00130480
.92	.78	.08	.22	.54306187	
.85	.85	.15	.15	.54429981	-.00123794
.90	.78	.10	.22	.50699437	
.84	.84	.16	.16	.50804642	-.00105205
.88	.78	.12	.22	.47214973	
.83	.83	.17	.17	.47295883	-.00080910
.86	.78	.14	.22	.43860678	
.82	.82	.18	.18	.43916321	-.00055643
.84	.78	.16	.22	.40642753	
.81	.81	.19	.19	.40675645	-.00032892
.82	.78	.18	.22	.37565863	
.80	.80	.20	.20	.37580962	-.00015099
.80	.78	.20	.22	.34633264	
.79	.79	.21	.21	.34637111	-.00003847
.78	.78	.22	.22	.31846937	
1.00	.76	.00	.24	.65718843	
.88	.88	.12	.12	.65827500	-.00108657
.98	.76	.02	.24	.61786946	
.87	.87	.13	.13	.61963081	-.00176135
.96	.76	.04	.24	.57951640	
.86	.86	.14	.14	.58155995	-.00204355
.94	.76	.06	.24	.54225083	
.85	.85	.15	.15	.54429981	-.00204898
.92	.76	.08	.24	.50617481	
.84	.84	.16	.16	.50804642	-.00187161
.90	.76	.10	.24	.47137240	
.83	.83	.17	.17	.47295883	-.00158643
.88	.76	.12	.24	.43791100	
.82	.82	.18	.18	.43916321	-.00125221
.86	.76	.14	.24	.40584271	
.81	.81	.19	.19	.40675645	-.00091374
.84	.76	.16	.24	.37520563	
.80	.80	.20	.20	.37580962	-.00060399
.82	.76	.18	.24	.34602500	
.79	.79	.21	.21	.34637111	-.00034611
.80	.76	.20	.24	.31831445	
.78	.78	.22	.22	.31846937	-.00015492
.78	.76	.22	.24	.29207705	
.77	.77	.23	.23	.29211568	-.00003863
.76	.76	.24	.24	.26730642	
1.00	.74	.00	.26	.61715076	
.87	.87	.13	.13	.61963081	-.00248005
.98	.74	.02	.26	.57860477	
.86	.86	.14	.14	.58155995	-.00295518
.96	.74	.04	.26	.54123151	
.85	.85	.15	.15	.54429981	-.00306830
.94	.74	.06	.26	.50511778	
.84	.84	.16	.16	.50804642	-.00292864
.92	.74	.08	.26	.47033406	
.83	.83	.17	.17	.47295883	-.00262477
.90	.74	.10	.26	.43693591	
.82	.82	.18	.18	.43916321	-.00222730
.88	.74	.12	.26	.40496501	
.81	.81	.19	.19	.40675645	-.00179144
.86	.74	.14	.26	.37445040	
.80	.80	.20	.20	.37580962	-.00135922
.84	.74	.16	.26	.34540961	

.79	.79	.21	.21	.34637111	-.00096150
.82	.74	.18	.26	.31784964	
.78	.78	.22	.22	.31846937	-.00061973
.80	.74	.20	.26	.29176796	
.77	.77	.23	.23	.29211568	-.00034772
.78	.74	.22	.26	.26715352	
.76	.76	.24	.24	.26730642	-.00015290
.76	.74	.24	.26	.24398768	
.75	.75	.25	.25	.24402521	-.00003753
.74	.74	.26	.26	.22224495	
1.00	.72	.00	.28	.57751009	
.86	.86	.14	.14	.58155995	-.00404986
.98	.72	.02	.28	.53999663	
.85	.85	.15	.15	.54429981	-.00430318
.96	.72	.04	.28	.50381815	
.84	.84	.16	.16	.50804642	-.00422827
.94	.72	.06	.28	.46903129	
.83	.83	.17	.17	.47295883	-.00392754
.92	.72	.08	.28	.43567929	
.82	.82	.18	.18	.43916321	-.00348392
.90	.72	.10	.28	.40379306	
.81	.81	.19	.19	.40675645	-.00296339
.88	.72	.12	.28	.37339222	
.80	.80	.20	.20	.37580962	-.00241740
.86	.72	.14	.28	.34448615	
.79	.79	.21	.21	.34637111	-.00188496
.84	.72	.16	.28	.31707478	
.78	.78	.22	.22	.31846937	-.00139459
.82	.72	.18	.28	.29114970	
.77	.77	.23	.23	.29211568	-.00096598
.80	.72	.20	.28	.26669482	
.76	.76	.24	.24	.26730642	-.00061160
.78	.72	.22	.28	.24368733	
.75	.75	.25	.25	.24402521	-.00033788
.76	.72	.24	.28	.22209841	
.74	.74	.26	.26	.22224495	-.00014654
.74	.72	.26	.28	.20189398	
.73	.73	.27	.27	.20192953	-.00003555
.72	.72	.28	.28	.18303541	
1.00	.70	.00	.30	.53853388	
.85	.85	.15	.15	.54429981	-.00576593
.98	.70	.02	.30	.50226693	
.84	.84	.16	.16	.50804642	-.00577949
.96	.70	.04	.30	.46745761	
.83	.83	.17	.17	.47295883	-.00550122
.94	.70	.06	.30	.43413665	
.82	.82	.18	.18	.43916321	-.00502656
.92	.70	.08	.30	.40232386	
.81	.81	.19	.19	.40675645	-.00443259
.90	.70	.10	.30	.37202914	
.80	.80	.20	.20	.37580962	-.00378048
.88	.70	.12	.30	.34325337	
.79	.79	.21	.21	.34637111	-.00311774
.86	.70	.14	.30	.31598923	
.78	.78	.22	.22	.31846937	-.00248014
.84	.70	.16	.30	.29022194	
.77	.77	.23	.23	.29211568	-.00189374
.82	.70	.18	.30	.26593016	
.76	.76	.24	.24	.26730642	-.00137626
.80	.70	.20	.30	.24308660	
.75	.75	.25	.25	.24402521	-.00093861
.78	.70	.22	.30	.22165876	

.74	.74	.26	.26	.22224495	-.00058619
.76	.70	.24	.30	.20160963	
.73	.73	.27	.27	.20192953	-.00031990
.74	.70	.26	.30	.18289818	
.72	.72	.28	.28	.18303541	-.00013723
.72	.70	.28	.30	.16548012	
.71	.71	.29	.29	.16551307	-.00003295
.70	.70	.30	.30	.14930833	
1.00	.68	.00	.32	.50044969	
.84	.84	.16	.16	.50804642	-.00759673
.98	.68	.02	.32	.46560224	
.83	.83	.17	.17	.47295883	-.00735659
.96	.68	.04	.32	.43230007	
.82	.82	.18	.18	.43916321	-.00686314
.94	.68	.06	.32	.40055174	
.81	.81	.19	.19	.40675645	-.00620471
.92	.68	.08	.32	.37035722	
.80	.80	.20	.20	.37580962	-.00545240
.90	.68	.10	.32	.34170870	
.79	.79	.21	.21	.34637111	-.00466241
.88	.68	.12	.32	.31459127	
.78	.78	.22	.22	.31846937	-.00387810
.86	.68	.14	.32	.28898368	
.77	.77	.23	.23	.29211568	-.00313200
.84	.68	.16	.32	.26485896	
.76	.76	.24	.24	.26730642	-.00244746
.82	.68	.18	.32	.24218518	
.75	.75	.25	.25	.24402521	-.00184003
.80	.68	.20	.32	.22092589	
.74	.74	.26	.26	.22224495	-.00131906
.78	.68	.22	.32	.20104085	
.73	.73	.27	.27	.20192953	-.00088868
.76	.68	.24	.32	.18248652	
.72	.72	.28	.28	.18303541	-.00054889
.74	.68	.26	.32	.16521655	
.71	.71	.29	.29	.16551307	-.00029652
.72	.68	.28	.32	.14918232	
.70	.70	.30	.30	.14930833	-.00012601
.70	.68	.30	.32	.13433340	
.69	.69	.31	.31	.13436340	-.00003000
.68	.68	.32	.32	.12061794	
1.00	.66	.00	.34	.46344862	
.83	.83	.17	.17	.47295883	-.00951021
.98	.66	.02	.34	.43015690	
.82	.82	.18	.18	.43916321	-.00900631
.96	.66	.04	.34	.39846722	
.81	.81	.19	.19	.40675645	-.00828923
.94	.66	.06	.34	.36836952	
.80	.80	.20	.20	.37580962	-.00744010
.92	.66	.08	.34	.33984715	
.79	.79	.21	.21	.34637111	-.00652396
.90	.66	.10	.34	.31287748	
.78	.78	.22	.22	.31846937	-.00559189
.88	.66	.12	.34	.28743259	
.77	.77	.23	.23	.29211568	-.00468309
.86	.66	.14	.34	.26347978	
.76	.76	.24	.24	.26730642	-.00382664
.84	.66	.16	.34	.24098218	
.75	.75	.25	.25	.24402521	-.00304303
.82	.66	.18	.34	.21989928	
.74	.74	.26	.26	.22224495	-.00234567
.80	.66	.20	.34	.20018742	

.73	.73	.27	.27	.20192953	-.00174211
.78	.66	.22	.34	.18180029	
.72	.72	.28	.28	.18303541	-.00123512
.76	.66	.24	.34	.16468935	
.71	.71	.29	.29	.16551307	-.00082372
.74	.66	.26	.34	.14880431	
.70	.70	.30	.30	.14930833	-.00050402
.72	.66	.28	.34	.13409348	
.69	.69	.31	.31	.13436340	-.00026992
.70	.66	.30	.34	.12050418	
.68	.68	.32	.32	.12061794	-.00011376
.68	.66	.32	.34	.10798306	
.67	.67	.33	.33	.10800993	-.00002687
.66	.66	.34	.34	.09647647	
1.00	.64	.00	.36	.42768850	
.82	.82	.18	.18	.43916321	-.01147471
.98	.64	.02	.36	.39605586	
.81	.81	.19	.19	.40675645	-.01070059
.96	.64	.04	.36	.36605503	
.80	.80	.20	.20	.37580962	-.00975459
.94	.64	.06	.36	.33766047	
.79	.79	.21	.21	.34637111	-.00871064
.92	.64	.08	.36	.31084182	
.78	.78	.22	.22	.31846937	-.00762755
.90	.64	.10	.36	.28556438	
.77	.77	.23	.23	.29211568	-.00655130
.88	.64	.12	.36	.26178961	
.76	.76	.24	.24	.26730642	-.00551681
.86	.64	.14	.36	.23947562	
.75	.75	.25	.25	.24402521	-.00454959
.84	.64	.16	.36	.21857768	
.74	.74	.26	.26	.22224495	-.00366727
.82	.64	.18	.36	.19904856	
.73	.73	.27	.27	.20192953	-.00288097
.80	.64	.20	.36	.18083906	
.72	.72	.28	.28	.18303541	-.00219635
.78	.64	.22	.36	.16389832	
.71	.71	.29	.29	.16551307	-.00161475
.76	.64	.24	.36	.14817418	
.70	.70	.30	.30	.14930833	-.00113415
.74	.64	.26	.36	.13361360	
.69	.69	.31	.31	.13436340	-.00074980
.72	.64	.28	.36	.12016291	
.68	.68	.32	.32	.12061794	-.00045503
.70	.64	.30	.36	.10776815	
.67	.67	.33	.33	.10800993	-.00024178
.68	.64	.32	.36	.09637533	
.66	.66	.34	.34	.09647647	-.00010114
.66	.64	.34	.36	.08593071	
.65	.65	.35	.35	.08595443	-.00002371
.64	.64	.36	.36	.07638104	
1.00	.62	.00	.38	.39329698	
.81	.81	.19	.19	.40675645	-.01345947
.98	.62	.02	.38	.36339750	
.80	.80	.20	.20	.37580962	-.01241212
.96	.62	.04	.38	.33513614	
.79	.79	.21	.21	.34637111	-.01123497
.94	.62	.06	.38	.30847478	
.78	.78	.22	.22	.31846937	-.00999459
.92	.62	.08	.38	.28337196	
.77	.77	.23	.23	.29211568	-.00874372
.90	.62	.10	.38	.25978328	

.76	.76	.24	.24	.26730642	-.00752314
.88	.62	.12	.38	.23766182	
.75	.75	.25	.25	.24402521	-.00636339
.86	.62	.14	.38	.21695853	
.74	.74	.26	.26	.22224495	-.00528642
.84	.62	.16	.38	.19762260	
.73	.73	.27	.27	.20192953	-.00430693
.82	.62	.18	.38	.17960178	
.72	.72	.28	.28	.18303541	-.00343363
.80	.62	.20	.38	.16284281	
.71	.71	.29	.29	.16551307	-.00267026
.78	.62	.22	.38	.14729159	
.70	.70	.30	.30	.14930833	-.00201674
.76	.62	.24	.38	.13289358	
.69	.69	.31	.31	.13436340	-.00146982
.74	.62	.26	.38	.11959405	
.68	.68	.32	.32	.12061794	-.00102389
.72	.62	.28	.38	.10733829	
.67	.67	.33	.33	.10800993	-.00067164
.70	.62	.30	.38	.09607190	
.66	.66	.34	.34	.09647647	-.00040456
.68	.62	.32	.38	.08574100	
.65	.65	.35	.35	.08595443	-.00021343
.66	.62	.34	.38	.07629238	
.64	.64	.36	.36	.07638104	-.00008866
.64	.62	.36	.38	.06767375	
.63	.63	.37	.37	.06769441	-.00002065
.62	.62	.38	.38	.05983392	
1.00	.60	.00	.40	.36037440	
.80	.80	.20	.20	.37580962	-.01543522
.98	.60	.02	.40	.33225621	
.79	.79	.21	.21	.34637111	-.01411490
.96	.60	.04	.40	.30576228	
.78	.78	.22	.22	.31846937	-.01270709
.94	.60	.06	.40	.28084444	
.77	.77	.23	.23	.29211568	-.01127124
.92	.60	.08	.40	.25745255	
.76	.76	.24	.24	.26730642	-.00985387
.90	.60	.10	.40	.23553464	
.75	.75	.25	.25	.24402521	-.00849057
.88	.60	.12	.40	.21503738	
.74	.74	.26	.26	.22224495	-.00720757
.86	.60	.14	.40	.19590634	
.73	.73	.27	.27	.20192953	-.00602319
.84	.60	.16	.40	.17808626	
.72	.72	.28	.28	.18303541	-.00494915
.82	.60	.18	.40	.16152136	
.71	.71	.29	.29	.16551307	-.00399171
.80	.60	.20	.40	.14615560	
.70	.70	.30	.30	.14930833	-.00315273
.78	.60	.22	.40	.13193288	
.69	.69	.31	.31	.13436340	-.00243052
.76	.60	.24	.40	.11879729	
.68	.68	.32	.32	.12061794	-.00182065
.74	.60	.26	.40	.10669334	
.67	.67	.33	.33	.10800993	-.00131659
.72	.60	.28	.40	.09556613	
.66	.66	.34	.34	.09647647	-.00091034
.70	.60	.30	.40	.08536154	
.65	.65	.35	.35	.08595443	-.00059288
.68	.60	.32	.40	.07602638	
.64	.64	.36	.36	.07638104	-.00035466

.66	.60	.34	.40	.06750855	
.63	.63	.37	.37	.06769441	-.00018585
.64	.60	.36	.40	.05975722	
.62	.62	.38	.38	.05983392	-.00007670
.62	.60	.38	.40	.05272284	
.61	.61	.39	.39	.05274059	-.00001774
.60	.60	.40	.40	.04635740	
1.00	.58	.00	.42	.32899648	
.79	.79	.21	.21	.34637111	-.01737463
.98	.58	.02	.42	.30268489	
.78	.78	.22	.22	.31846937	-.01578448
.96	.58	.04	.42	.27796647	
.77	.77	.23	.23	.29211568	-.01414921
.94	.58	.06	.42	.25478537	
.76	.76	.24	.24	.26730642	-.01252105
.92	.58	.08	.42	.23308478	
.75	.75	.25	.25	.24402521	-.01094043
.90	.58	.10	.42	.21280719	
.74	.74	.26	.26	.22224495	-.00943776
.88	.58	.12	.42	.19389456	
.73	.73	.27	.27	.20192953	-.00803497
.86	.58	.14	.42	.17628870	
.72	.72	.28	.28	.18303541	-.00674671
.84	.58	.16	.42	.15993132	
.71	.71	.29	.29	.16551307	-.00558175
.82	.58	.18	.42	.14476440	
.70	.70	.30	.30	.14930833	-.00454393
.80	.58	.20	.42	.13073026	
.69	.69	.31	.31	.13436340	-.00363314
.78	.58	.22	.42	.11777186	
.68	.68	.32	.32	.12061794	-.00284608
.76	.58	.24	.42	.10583283	
.67	.67	.33	.33	.10800993	-.00217710
.74	.58	.26	.42	.09485775	
.66	.66	.34	.34	.09647647	-.00161871
.72	.58	.28	.42	.08479221	
.65	.65	.35	.35	.08595443	-.00116221
.70	.58	.30	.42	.07558298	
.64	.64	.36	.36	.07638104	-.00079805
.68	.58	.32	.42	.06717813	
.63	.63	.37	.37	.06769441	-.00051628
.66	.58	.34	.42	.05952711	
.62	.62	.38	.38	.05983392	-.00030680
.64	.58	.36	.42	.05258085	
.61	.61	.39	.39	.05274059	-.00015973
.62	.58	.38	.42	.04629189	
.60	.60	.40	.40	.04635740	-.00006550
.60	.58	.40	.42	.04061438	
.59	.59	.41	.41	.04062945	-.00001506
.58	.58	.42	.42	.03550420	
1.00	.56	.00	.44	.29921700	
.78	.78	.22	.22	.31846937	-.01925237
.98	.56	.02	.44	.27471727	
.77	.77	.23	.23	.29211568	-.01739841
.96	.56	.04	.44	.25176514	
.76	.76	.24	.24	.26730642	-.01554128
.94	.56	.06	.44	.23029914	
.75	.75	.25	.25	.24402521	-.01372607
.92	.56	.08	.44	.21025770	
.74	.74	.26	.26	.22224495	-.01198725
.90	.56	.10	.44	.19157941	
.73	.73	.27	.27	.20192953	-.01035012

.88	.56	.12	.44	.17420313	
.72	.72	.28	.28	.18303541	-.00883228
.86	.56	.14	.44	.15806823	
.71	.71	.29	.29	.16551307	-.00744484
.84	.56	.16	.44	.14311476	
.70	.70	.30	.30	.14930833	-.00619357
.82	.56	.18	.44	.12928349	
.69	.69	.31	.31	.13436340	-.00507991
.80	.56	.20	.44	.11651622	
.68	.68	.32	.32	.12061794	-.00410172
.78	.56	.22	.44	.10475576	
.67	.67	.33	.33	.10800993	-.00325417
.76	.56	.24	.44	.09394613	
.66	.66	.34	.34	.09647647	-.00253034
.74	.56	.26	.44	.08403264	
.65	.65	.35	.35	.08595443	-.00192178
.72	.56	.28	.44	.07496199	
.64	.64	.36	.36	.07638104	-.00141905
.70	.56	.30	.44	.06668238	
.63	.63	.37	.37	.06769441	-.00101202
.68	.56	.32	.44	.05914355	
.62	.62	.38	.38	.05983392	-.00069036
.66	.56	.34	.44	.05229686	
.61	.61	.39	.39	.05274059	-.00044373
.64	.56	.36	.44	.04609537	
.60	.60	.40	.40	.04635740	-.00026202
.62	.56	.38	.44	.04049388	
.59	.59	.41	.41	.04062945	-.00013556
.60	.56	.40	.44	.03544896	
.58	.58	.42	.42	.03550420	-.00005524
.58	.56	.42	.44	.03091899	
.57	.57	.43	.43	.03093161	-.00001262
.56	.56	.44	.44	.02686419	
1.00	.54	.00	.46	.27107013	
.77	.77	.23	.23	.29211568	-.02104555
.98	.54	.02	.46	.24837006	
.76	.76	.24	.24	.26730642	-.01893636
.96	.54	.04	.46	.22716007	
.75	.75	.25	.25	.24402521	-.01686514
.94	.54	.06	.46	.20737486	
.74	.74	.26	.26	.22224495	-.01487009
.92	.54	.08	.46	.18894973	
.73	.73	.27	.27	.20192953	-.01297980
.90	.54	.10	.46	.17182091	
.72	.72	.28	.28	.18303541	-.01121450
.88	.54	.12	.46	.15592549	
.71	.71	.29	.29	.16551307	-.00958758
.86	.54	.14	.46	.14120168	
.70	.70	.30	.30	.14930833	-.00810665
.84	.54	.16	.46	.12758886	
.69	.69	.31	.31	.13436340	-.00677454
.82	.54	.18	.46	.11502768	
.68	.68	.32	.32	.12061794	-.00559026
.80	.54	.20	.46	.10346023	
.67	.67	.33	.33	.10800993	-.00454970
.78	.54	.22	.46	.09282998	
.66	.66	.34	.34	.09647647	-.00364649
.76	.54	.24	.46	.08308198	
.65	.65	.35	.35	.08595443	-.00287245
.74	.54	.26	.46	.07416288	
.64	.64	.36	.36	.07638104	-.00221816
.72	.54	.28	.46	.06602100	

.63	.63	.37	.37	.06769441	-.00167340
.70	.54	.30	.46	.05860637	
.62	.62	.38	.38	.05983392	-.00122754
.68	.54	.32	.46	.05187077	
.61	.61	.39	.39	.05274059	-.00086981
.66	.54	.34	.46	.04576780	
.60	.60	.40	.40	.04635740	-.00058959
.64	.54	.36	.46	.04025287	
.59	.59	.41	.41	.04062945	-.00037658
.62	.54	.38	.46	.03528322	
.58	.58	.42	.42	.03550420	-.00022097
.60	.54	.40	.46	.03081800	
.57	.57	.43	.43	.03093161	-.00011361
.58	.54	.42	.46	.02681819	
.56	.56	.44	.44	.02686419	-.00004600
.56	.54	.44	.46	.02324665	
.55	.55	.45	.45	.02325710	-.00001045
.54	.54	.46	.46	.02006813	
1.00	.52	.00	.48	.24457275	
.76	.76	.24	.24	.26730642	-.02273367
.98	.52	.02	.48	.22364510	
.75	.75	.25	.25	.24402521	-.02038011
.96	.52	.04	.48	.20414031	
.74	.74	.26	.26	.22224495	-.01810464
.94	.52	.06	.48	.18599080	
.73	.73	.27	.27	.20192953	-.01593873
.92	.52	.08	.48	.16913026	
.72	.72	.28	.28	.18303541	-.01390515
.90	.52	.10	.48	.15349380	
.71	.71	.29	.29	.16551307	-.01201927
.88	.52	.12	.48	.13901796	
.70	.70	.30	.30	.14930833	-.01029037
.86	.52	.14	.48	.12564085	
.69	.69	.31	.31	.13436340	-.00872255
.84	.52	.16	.48	.11330213	
.68	.68	.32	.32	.12061794	-.00731581
.82	.52	.18	.48	.10194320	
.67	.67	.33	.33	.10800993	-.00606673
.80	.52	.20	.48	.09150711	
.66	.66	.34	.34	.09647647	-.00496936
.78	.52	.22	.48	.08193869	
.65	.65	.35	.35	.08595443	-.00401573
.76	.52	.24	.48	.07318460	
.64	.64	.36	.36	.07638104	-.00319644
.74	.52	.26	.48	.06519330	
.63	.63	.37	.37	.06769441	-.00250110
.72	.52	.28	.48	.05791515	
.62	.62	.38	.38	.05983392	-.00191877
.70	.52	.30	.48	.05130235	
.61	.61	.39	.39	.05274059	-.00143824
.68	.52	.32	.48	.04530905	
.60	.60	.40	.40	.04635740	-.00104834
.66	.52	.34	.48	.03989127	
.59	.59	.41	.41	.04062945	-.00073817
.64	.52	.36	.48	.03500697	
.58	.58	.42	.42	.03550420	-.00049723
.62	.52	.38	.48	.03061601	
.57	.57	.43	.43	.03093161	-.00031560
.60	.52	.40	.48	.02668015	
.56	.56	.44	.44	.02686419	-.00018404
.58	.52	.42	.48	.02316307	
.55	.55	.45	.45	.02325710	-.00009403

.56	.52	.44	.48	.02003029	
.54	.54	.46	.46	.02006813	-.00003783
.54	.52	.46	.48	.01724922	
.53	.53	.47	.47	.01725776	-.00000853
.52	.52	.48	.48	.01478909	
1.00	.50	.00	.50	.21972656	
.75	.75	.25	.25	.24402521	-.02429865
.98	.50	.02	.50	.20053120	
.74	.74	.26	.26	.22224495	-.02171375
.96	.50	.04	.50	.18268380	
.73	.73	.27	.27	.20192953	-.01924573
.94	.50	.06	.50	.16611591	
.72	.72	.28	.28	.18303541	-.01691950
.92	.50	.08	.50	.15076088	
.71	.71	.29	.29	.16551307	-.01475219
.90	.50	.10	.50	.13655384	
.70	.70	.30	.30	.14930833	-.01275449
.88	.50	.12	.50	.12343179	
.69	.69	.31	.31	.13436340	-.01093161
.86	.50	.14	.50	.11133362	
.68	.68	.32	.32	.12061794	-.00928432
.84	.50	.16	.50	.10020014	
.67	.67	.33	.33	.10800993	-.00780979
.82	.50	.18	.50	.08997411	
.66	.66	.34	.34	.09647647	-.00650235
.80	.50	.20	.50	.08060030	
.65	.65	.35	.35	.08595443	-.00535413
.78	.50	.22	.50	.07202536	
.64	.64	.36	.36	.07638104	-.00435568
.76	.50	.24	.50	.06419805	
.63	.63	.37	.37	.06769441	-.00349636
.74	.50	.26	.50	.05706904	
.62	.62	.38	.38	.05983392	-.00276488
.72	.50	.28	.50	.05059105	
.61	.61	.39	.39	.05274059	-.00214954
.70	.50	.30	.50	.04471877	
.60	.60	.40	.40	.04635740	-.00163862
.68	.50	.32	.50	.03940890	
.59	.59	.41	.41	.04062945	-.00122054
.66	.50	.34	.50	.03462009	
.58	.58	.42	.42	.03550420	-.00088410
.64	.50	.36	.50	.03031297	
.57	.57	.43	.43	.03093161	-.00061864
.62	.50	.38	.50	.02645008	
.56	.56	.44	.44	.02686419	-.00041411
.60	.50	.40	.50	.02299590	
.55	.55	.45	.45	.02325710	-.00026120
.58	.50	.42	.50	.01991677	
.54	.54	.46	.46	.02006813	-.00015135
.56	.50	.44	.50	.01718092	
.53	.53	.47	.47	.01725776	-.00007683
.54	.50	.46	.50	.01475837	
.52	.52	.48	.48	.01478909	-.00003071
.52	.50	.48	.50	.01262094	
.51	.51	.49	.49	.01262783	-.00000688
.50	.50	.50	.50	.01074218	

STOP