Coding For Frequency Hopped Spread Spectrum Satellite Communications

Final Report
Period Covered: April 16 1987 to March 31 1988

Prepared for
The Department of Communications of Canada
under DSS Contract No. 27ST. 36001-6-3539.

Department of Electrical and Computer Engineering

THE UNIVERSITY OF VICTORIA
P.O. BOX 1700, VICTORIA, B.C. CANADA V8W 2Y2

Coding For Frequency Hopped Spread Spectrum Satellite Communications

Final Report
Period Covered: April 16 1987 to March 31 1988

Prepared for
The Department of Communications of Canada
under DSS Contract No. 27ST. 36001-6-3539.

by

Q. Wang, T.A. Gulliver, V.K. Bhargava and W.D. Little
Department of Electrical and Computer Engineering
University of Victoria
P.O. Box 1700
Victoria, B.C.
Canada V8W 2Y2

Technical Report ECE-88-2

April 15, 1988

Abstract

The performance of various types of error correcting codes is examined under both partial band noise jamming and multitone jamming using fast frequency hopping, noncoherent M-ary frequency-shift-keying (NCMFSK). A comprehensive study including convolutional codes, binary and nonbinary block codes and concatenated codes has been conducted. The bit error rate (BER) performance of many error correcting codes is presented. These results augment those previously published on the subject.

A fixed data rate is assumed in Chapter 2. However, a practical frequency hopping spread spectrum (SS/FH) system may be limited to a fixed hop rate. This constraint has been given little consideration in previous work on coding for SS/FH systems. In Chapter 3, the performance of error correcting codes is examined under partial band noise jamming and multitone jamming using the same system as in Chapter 2, but with fixed hop rates. These results are compared to those with a fixed data rate. The analysis method used is the well known Chernoff union bound.

The performance of a recently proposed efficient anti-jam communication system is examined in Chapter 4. The system employs frequency hopping, MFSK modulation, diversity, Reed-Solomon (RS) coding, and parallel error-erasure correction decoding. It has previously been shown to be effective in partial band noise jamming. In this report, we evaluate the performance of this system in multitone jamming. An exact method is used rather than a bounding technique. It is shown that in worst case jamming, when the redundancy is not large, multitone jamming tends to be more effective than partial band noise jamming from the jammer's point of view, for nonbinary FSK. The optimum design of the system under worst case jamming is presented in terms of the combination of MFSK, diversity and RS coding, and it is shown that a proper combination with large redundancy can completely nullify worst case multitone jamming.

Chapter 5 describes the design of a CODEC based on the (127,99) four error correcting BCH code. This code was chosen as a compromise between overall performance, as given in Chapter 2, and implementation complexity. The CODEC is designed completely in hardware for implementation with application specific integrated circuits. This approach has a short implementation cycle, requires a very small number of integrated circuit chips, and yields a CODEC that can operate up to about 5Mbits/sec.

Contents

List of Figures

Chapter 1

Introduction

In defence communication systems, spread-spectrum techniques such as frequency-hopping (FH) have been utilized to provide some protection against jamming. However, an intelligent jammer can drastically reduce the effectiveness of such a system. This effectiveness can be regained through the use of error correcting codes. In this report we present the results of a comprehensive study of the performance of various error correcting codes when used in a frequency-hopping system.

1.1 System Model

The system we consider employs the fast frequency-hopping (FFH) noncoherent M-ary frequency-shift-keying (NCFSK) technique. By fast we mean one or more hops per transmitted symbol. Our objective is to investigate the performance of various known error correcting codes in such a system under different kinds of jamming. The system model is shown in Fig. 1.1. System assumptions are as follows.

Transmitted signals are MFSK orthogonal signals which hop over a total spread spectrum bandwith W_{ss}. The channel over W_{ss} is assumed to be uniform. Noncoherent soft energy detection (square-law reception) of each hop is assumed. In practice this can

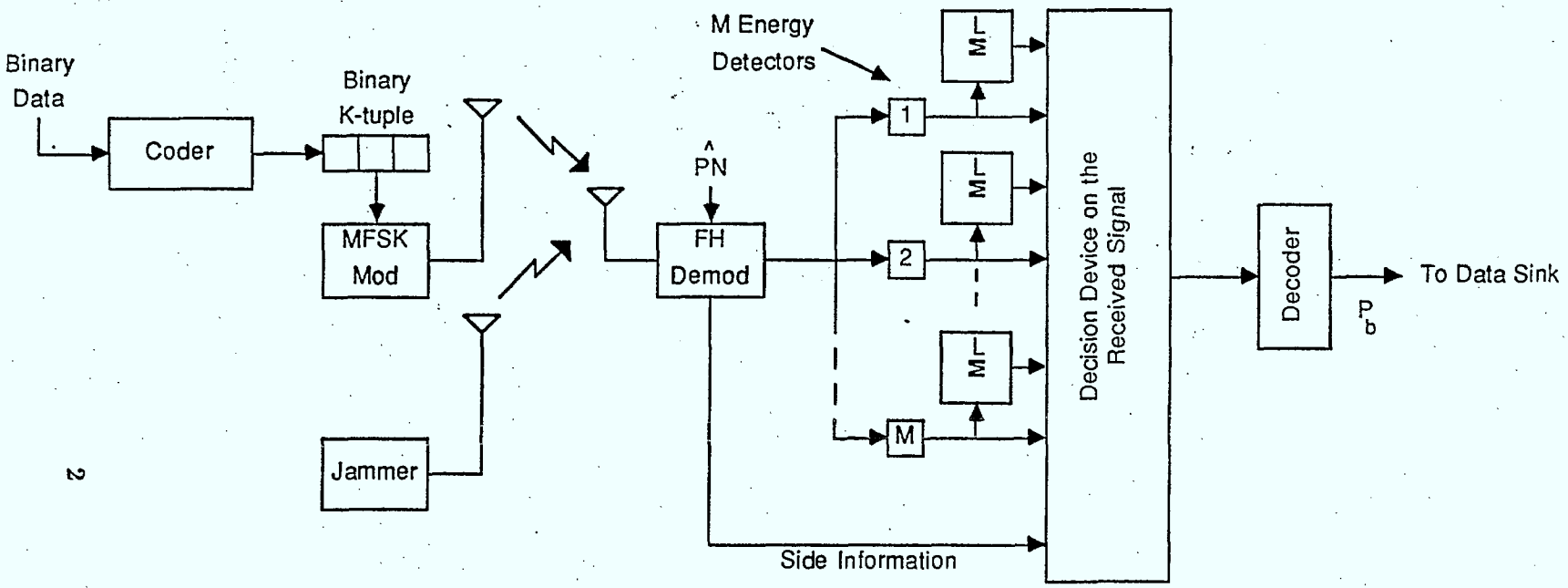


Figure 1.1: Block diagram of a FH/MFSK system in a jamming environment. One of $M = 2^K$ tones is transmitted. The carrier is hopped according to the pattern determined by the PN code. Dehopping requires the derived PN reference ($\widehat{\text{PN}}$). Detection is non-coherent. Side information is assumed to be perfect so that the receiver knows with certainty whether each hop of an M-ary symbol is jammed or not. If any of the L hops is not jammed, an error free M-ary decision can be made based solely on that hop. Otherwise, the largest of the linear combinations of the energy of L hops is selected.

be approximated with finite level quantization. Note that this soft decision is used only in energy detection. While the soft energy decision may be passed to a soft decision decoder such as that for a convolutional code, the decision made after diversity combination can be a hard decision.

One coded M-ary symbol is transmitted in L hops using time diversity. For a given M and error correcting (EC) code, when the energy per symbol E_s is fixed, the energy per bit E_b is fixed. In this case, there is usually an optimum L, denoted as L_{opt}, at which the final bit error rate (BER) can be minimized for a given signal to noise ratio. This is not true when the hop rate R_h is fixed and the information bit rate R_b is variable. In Chapter 2 we evaluate the effectiveness of various EC codes with E_b assumed to be fixed. The case when R_h is fixed is treated in Chapter 3. For convenience, the term bit error rate (BER) is used to denote the probability of bit error. In the analysis of a system with a fixed data rate, L_{opt} provides an indication of how efficient a specific code is. A smaller L_{opt} implies that the code is more efficient against jamming in the sense that less added redundancy is required.

In a jamming environment, the channel is nonstationary, and side information is valuable for efficient data reception. However, a system heavily dependent on side information may not be robust, because the information may not always be obtainable or its quality may vary greatly. One method used in practice to derive the side information is to implement automatic gain control (AGC) in the receiver[1]. Based on this implementation, we may assume, as a good approximation, that the receiver knows with certainty whether each hop of an M-ary symbol is jammed or not. If any of the L hops is not jammed, an error free M-ary decision can be made based solely on that hop (thermal noise is neglected, this will be discussed shortly). Otherwise, the largest of the linear combinations (direct sums) of the energy of L hops is selected, as shown by the decision device inputs from the summation boxes in Fig. 1.1. For the error-erasure correction decoding considered in Chapter 4, we attach an erasure flag to this symbol.

Under strong jamming, the receiver thermal or non-hostile background noise is usually small compared to the jamming, so receiver noise is neglected here. Assume the total jamming power J (referenced to the receiver input) is fixed. Then the effective jamming power spectral density is given by

$$J_O = J/W_{ss}.$$

We consider two types of worst case (WC) intelligent but non-repeat-back jamming, namely partial band noise and multitone interference. For partial band noise (PBN) jamming, J is restricted to a fraction ρ $(0 < \rho \leq 1)$ of the full spread spectrum bandwidth, but in this band the power spectral density is increased to J_O/ρ. Multitone jamming (MT) includes band multitone jamming and independent multitone jamming. It has been shown that worst case multitone jamming tends to have a single jamming tone per jammed band[2], using equal power tones. We consider only this type of worst case multitone jamming. In this case the jammer has one parameter to optimize, namely the ratio of signal power of one hop to the power of the jamming tone, denoted as α.

In this report, we do not consider fading of the signal due to propagation, but in worst case jamming the signal already suffers from a kind of fading in terms of the signal to noise ratio. Thus, results without considering propagation fading may be indicative of the case with propagation fading. This point has been verified in previous work. For instance, a broadband jammer is the worst case noise jammer in a Rayleigh fading channel[3].

In anti-jam communications, a good code should perform well regardless of the type of jamming. Thus in this report we consider good codes to be those with the best performance for the most effective type of jamming, WC MT jamming or WC PBN jamming, at a given low BER, typically $P_b = 10^{-5}$.

For such systems, we can consider three methods to evaluate the BER performance of a code. Monte Carlo simulation is the most universal but most time consuming method. This brute force method is therefore not suitable for a comprehensive preliminary investi-

gation of codes. It may be considered to evaluate the BER performance when a code has been adopted. Note that this method has the highest credibility.

The second method is the exact numerical computation of system performance. While it is not impossible, it is generally difficult and cumbersome to compute the exact performance of systems like the one considered here. Since these results are usually very complex, in most cases they do not yield insights readily.

The third method is the Chernoff union bound method which gives an upper bound for the BER. The computations involved in this method are usually much simpler than the other two methods. Though the general credibility of this method remains controversial, it has been shown to provide useful and reliable information[3] for the systems considered in Chapters 2 and 3. Due to its relative simplicity it is especially suitable for a comprehensive study of various codes, and thus has been chosen as the evaluation method. It provides a unified approach to evaluating performance and provides a clear relationship between the system parameters and BER. This method can be used to select good codes which can then be analysed more accurately using other techniques.

Chapter 5 presents the design of a CODEC based on the (127,99) BCH code. The design is done completely in hardware using application specific integrated circuits.

Chapter 6 provides some directions for future work, primarily in the area of slow frequency hopping systems.

Chapter 2

Performance of Error Correcting Codes for Fast Frequency Hopped Noncoherent *M*FSK Spread Spectrum Communications with a Fixed Data Rate

In this chapter we evaluate the performance of various EC codes assuming a fixed data rate. Although the performance of error correcting codes in FH systems has been widely studied, there is no single reference in the literature providing complete information on the BER performance (rather than other criteria, such as the cutoff rate) of various codes for both PBN and MT jamming. Ma and Poole[4] and Simon et al.[3] are perhaps the most comprehensive. Only partial band noise jamming was considered in [4] and only Reed-Solomon codes and several convolutional and concatenated codes were considered in [3]. These previous results are augmented through this work.

2.1 Formulas for Performance Evaluation

In this section the formulas used to evaluate the BER for block and convolutional codes are presented. They are based on the work reported in [3]. Optimum diversity is assumed in all cases. BER is given vs E_b/J_O, where E_b is the energy per information bit. The signal to noise power ratio is related to E_b/J_O by

$$(S/J)_{dB} = (E_b/J_O)_{dB} - PG$$

where $PG = 10\log_{10}(W_{ss}/R_b)$ is the processing gain. For example, if $W_{ss} = 100\ MHZ$ and $R_b = 2.4\ kb/s$, $PG = 46.2\ dB$. Note that the term processing gain has been given several conflicting definitions in the literature[3]. The definition we use is meaningful for all spread spectrum systems. We consider M-ary signalling, where $M = 2^K$, for K up to 5.

The probability of bit error, P_b, for convolutional codes is upperbounded by the Chernoff union bound[3] as

$$P_b \leq \begin{cases} \frac{1}{2}G(D^L) & \text{PBN jamming;} \\ G(D^L) & \text{MT jamming} \end{cases} \tag{2.1}$$

where

$$D^L = \begin{cases} e^{-L_{opt}} & L_{opt} > 1, \ \text{PBN;} \\ \frac{4e^{-1}}{R'E_b/J_O} & L_{opt} = 1, \ \text{PBN}, \quad \frac{E_b}{J_O} \geq \frac{3}{R'}; \\ \frac{e^{-\frac{\lambda}{1+\lambda}R'\frac{E_b}{J_O}}}{1-\lambda^2} & L_{opt} = 1, \ \text{PBN}, \quad \frac{E_b}{J_O} < \frac{3}{R'}; \\ \frac{1}{R'E_b/J_O} & L_{opt} = 1, \ \text{MT}, \quad K = 1, \ \frac{E_b}{J_O} \geq \frac{2}{R'}; \\ \frac{1}{2} & L_{opt} = 1, \ \text{MT}, \quad K = 1, \ \frac{E_b}{J_O} < \frac{2}{R'}; \\ \frac{\beta K}{R'\frac{E_b}{J_O}} & L_{opt} = 1, \ \text{MT}, \quad K \geq 2, \ \frac{E_b}{J_O} \geq \frac{\alpha_0 M}{R'}; \\ \frac{1}{R'\frac{E_b}{J_O}}\left[\frac{\alpha_{wc}(M-2)}{1-\alpha_{wc}}\right]^{1-\alpha_{wc}} & L_{opt} = 1, \ \text{MT}, \quad K \geq 2, \ \frac{E_b}{J_O} < \frac{\alpha_0 M}{R'}; \end{cases} \tag{2.2}$$

with λ equal to

$$\lambda = \frac{1}{2}\left[\sqrt{1+3R'\frac{E_b}{J_O}+\frac{1}{4}\left(R'\frac{E_b}{J_O}\right)^2}-\frac{1}{2}R'\frac{E_b}{J_O}-1\right]. \tag{2.3}$$

$R' = rK$ where r is the code rate. β is given in Table 2.1. The function $G(D^L)$ in (2.1) varies for different codes and will be given in the following sections. α_{wc} is the worst case α in MT jamming given by

$$\alpha_{wc} = \begin{cases} \alpha_0 & E_b/J_O \geq (L_{opt}\alpha_0 M)/R'; \\ \frac{R'E_b/J_O}{ML_{opt}} & E_b/J_O < (L_{opt}\alpha_0 M)/R'; \end{cases} \tag{2.4}$$

with α_0 given in Table 2.1, where "1_" means a value less than but infinitely close to 1. The worst case ρ in PBN jamming, denoted as ρ_{wc}, is given by

$$\rho_{wc} = \begin{cases} \frac{3L_{opt}}{R'E_b/J_O} & E_b/J_O \geq 3L_{opt}/R'; \\ 1 & E_b/J_O < 3L_{opt}/R'; \end{cases} \tag{2.5}$$

and L_{opt} by

$$L_{opt} = \begin{cases} \frac{R'}{\gamma}\frac{E_b}{J_O} & E_b/J_O \geq \gamma/R'; \\ 1 & E_b/J_O < \gamma/R'; \end{cases} \tag{2.6}$$

with γ defined as

$$\gamma = \begin{cases} 4 & \text{worst case PBN jamming;} \\ \beta Ke & \text{worst case MT jamming.} \end{cases} \tag{2.7}$$

β is given in Table 2.1. L_{opt} in (2.6) can be a noninteger which is not realizable, (and it cannot be less than 1). For the purpose of analysis, however, L_{opt} is informative in this finer form, and so is used for performance evaluation of all EC codes where applicable.

Unless otherwise specified, the following formulas define the probability of bit error,

Table 2.1: Values of α_0 and β.

K	α_0	β
1	1_	1
2	0.683	0.7945
3	0.527	0.8188
4	0.427	0.9583
5	0.356	1.2204

P_b, for block codes. The probability of bit error is given by

$$P_b = \frac{Q}{2(Q-1)} P_s \tag{2.8}$$

where P_s is the Q-ary output symbol error rate of an (n,k) Q-ary block code (with $\log_2 Q = q$ bits per Q-ary symbol), as shown in Fig. 2.1. For binary block codes, we have $Q = 2$, $q = 1$ and $P_b = P_s$. If the minimum distance is d, the number of correctable errors is given by

$$t = \lfloor (d-1)/2 \rfloor$$

where $\lfloor x \rfloor$ denotes the integer part of x. With hard decision decoding, P_s is related to the Q-ary symbol error rate before decoding, P_Q, by the well known formula (e.g. see [4])

$$P_s \approx \frac{1}{n} \sum_{i=t+1}^{n} i \binom{n}{i} P_Q^i (1-P_Q)^{n-i}. \tag{2.9}$$

For M-ary signalling, let P_K be the K-bit transmission symbol error rate (referenced to the point after diversity combination). P_K is given by

$$P_K \leq \begin{cases} \frac{(M-1)}{M} G(D^L) & \text{PBN jamming;} \\ \frac{2(M-1)}{M} G(D^L) & \text{MT jamming} \end{cases} \tag{2.10}$$

with

$$G(D^L) = \frac{M}{2} D^L \tag{2.11}$$

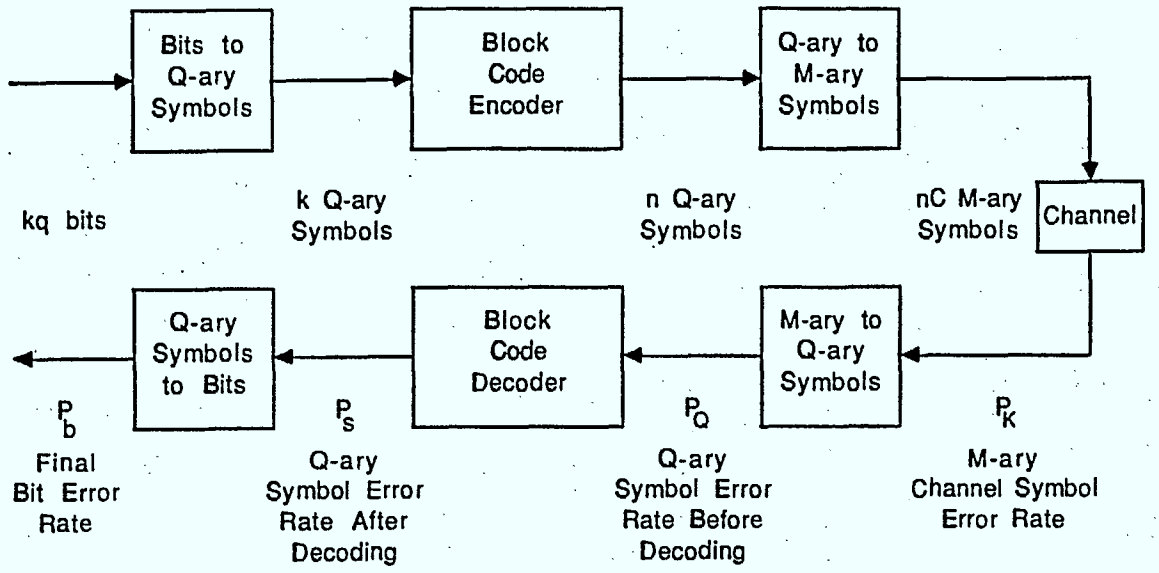


Figure 2.1: Block diagram of a system employing a block coding scheme, showing the symbols used and the alphabet conversions.

and D^L as defined in (2.2). The Q-ary symbol error rate P_Q is

$$P_Q = 1 - (1 - P_K)^C \tag{2.12}$$

where

$$C = \log_2 Q/K \tag{2.13}$$

when $q \geq K$. Fig. 2.1 illustrates the relationships given in (2.8) to (2.13) for a block coding system.

In the following sections we will use these formulas for three kinds of codes: block codes, convolutional codes and concatenated codes. Only selected results are presented.

2.2 Performance of Convolutional Codes

In this section the performance of various convolutional codes is presented. $G(D)$ is determined by the code used. Soft decision Viterbi decoding is assumed for all convolutional

codes.

2.2.1 Trumpis Codes

In [5] Trumpis presents two optimum (over orthogonal channels) constraint length 7 convolutional codes. Actually these codes are (K,1,7) binary convolutional codes, but K bits at the output of the encoder, corresponding to 1 bit at the input, are considered to be one M-ary symbol. The best of these codes is the 4-ary, rate $r = 1/2$ code for which $K = 2$ and

$$G(D) = 7D^7 + 39D^8 + 104D^9 + 352D^{10} + 1187D^{11} + \cdots. \tag{2.14}$$

The other is the 8-ary, rate $r = 1/3$ code. In this case we have $K = 3$ and

$$G(D) = D^7 + 4D^8 + 8D^9 + 49D^{10} + 92D^{11} + \cdots. \tag{2.15}$$

Fig. 2.2 gives the performance of these codes in WC PBN and WC MT jamming. Our results for these codes are the same as those in [3].

2.2.2 Dual-K Convolutional Codes

For all values of K, we have the class of dual-K M-ary convolutional codes with code rate $r = 1/\nu$ over $GF(2^K)$[1,3]. For every M-ary (K-bit) input word, ν M-ary code symbols are generated, where ν is an integer greater than 1. The constraint length is $2K$ which accounts for two binary shift registers in the encoder. Now $R' = K/\nu$ and

$$G(D) = \frac{MD^{2\nu}}{2[1 - \nu D^{\nu-1} - (M - \nu - 1)D^{\nu}]^2}. \tag{2.16}$$

As pointed out in [3], at a low BER, the performance of these codes does not depend on ν, which is verified by our results. However, with binary FSK under WC MT jamming, at a low signal to noise ratio, a low rate code is much more powerful than a high rate code[6]. At a low BER, the best code is that for $K = 2$, which is the same as that shown in [3]. The performance of the Dual-K codes with $\nu = 2$ is given in Figs. 2.3 and 2.4.

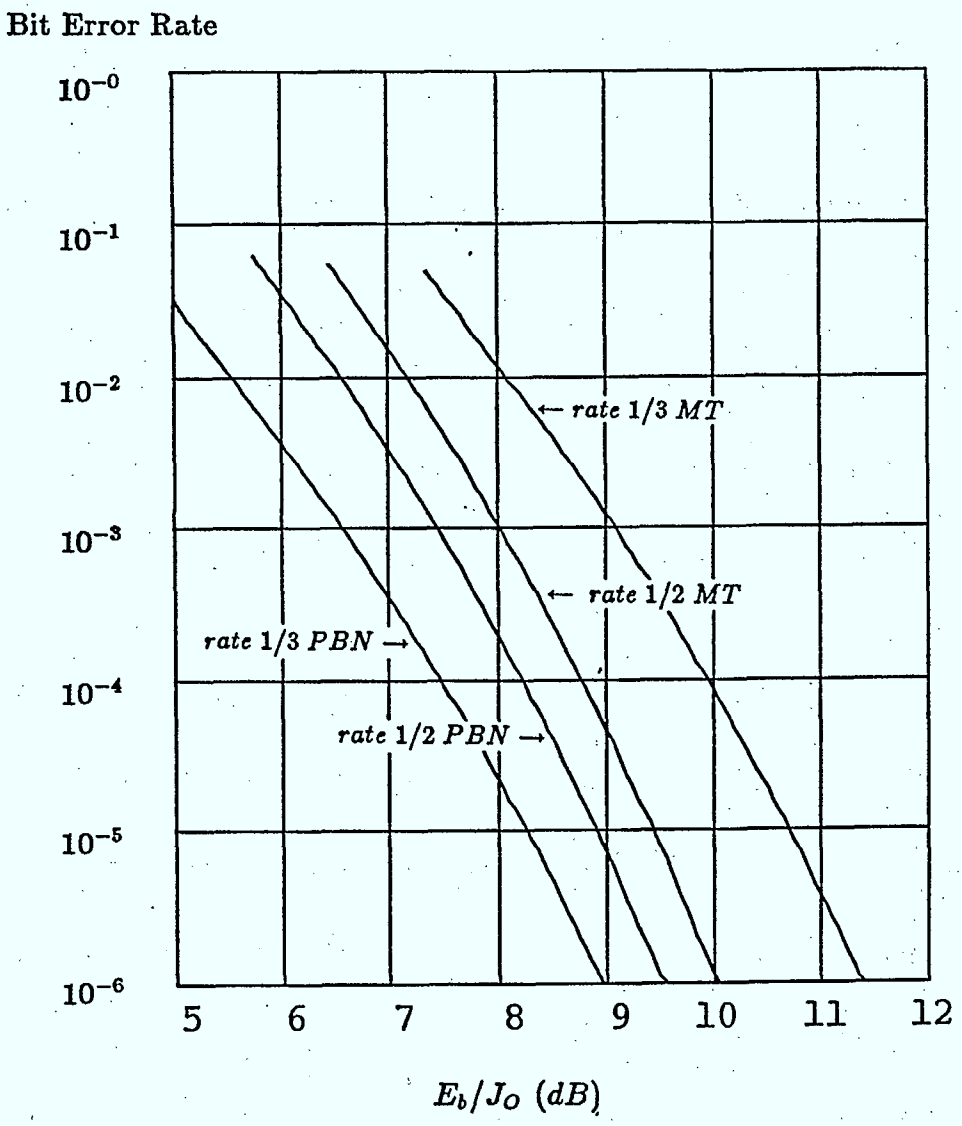


Figure 2.2: BER performance of the rate 1/2 and 1/3 Trumpis convolutional codes for $K = 2, 3$ respectively. The best code is that for $K = 2$.

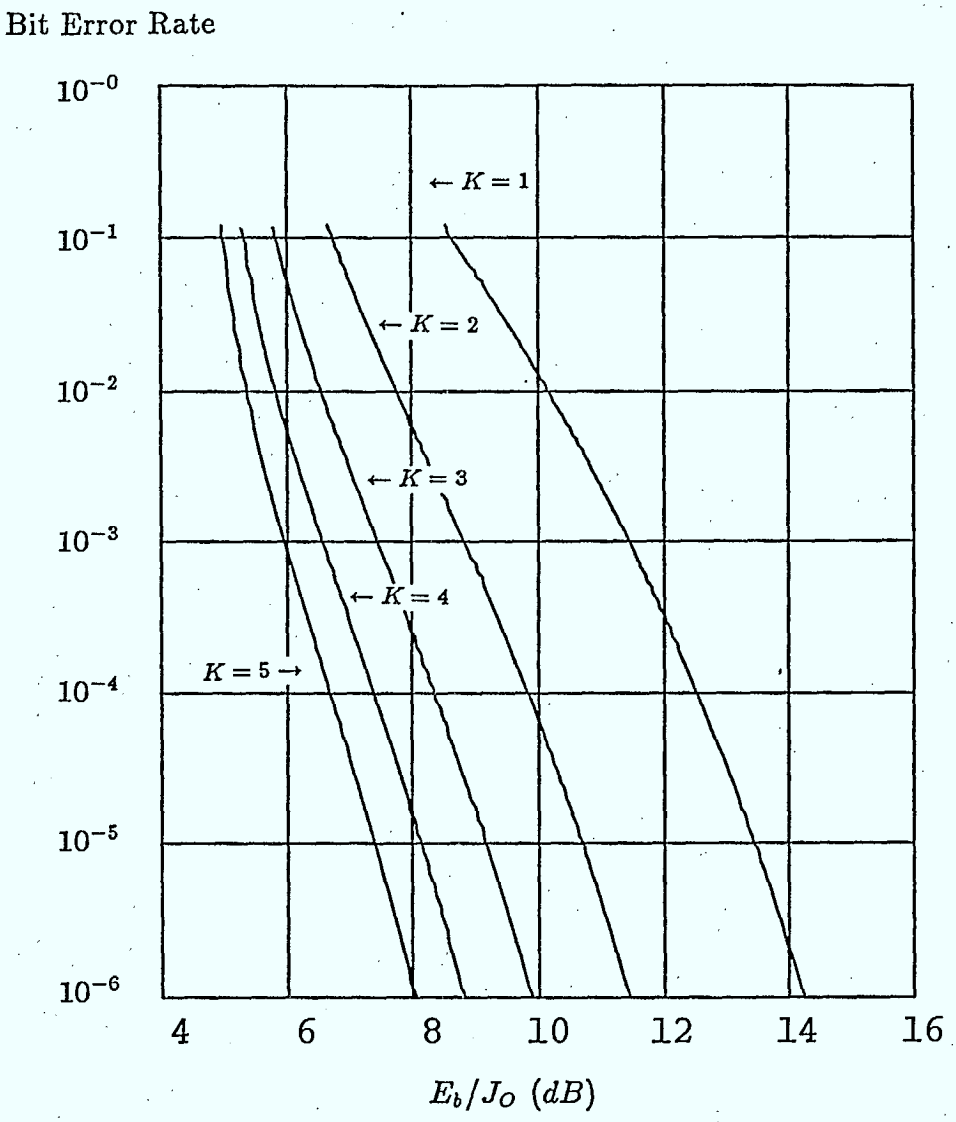


Figure 2.3: BER performance of the dual-K rate 1/2 convolutional codes for K equal to 1 through 5, under WC PBN jamming. The best code is that for $K = 2$, for which $\rho_{wc} = 0.75$ at a BER of 10^{-5}.

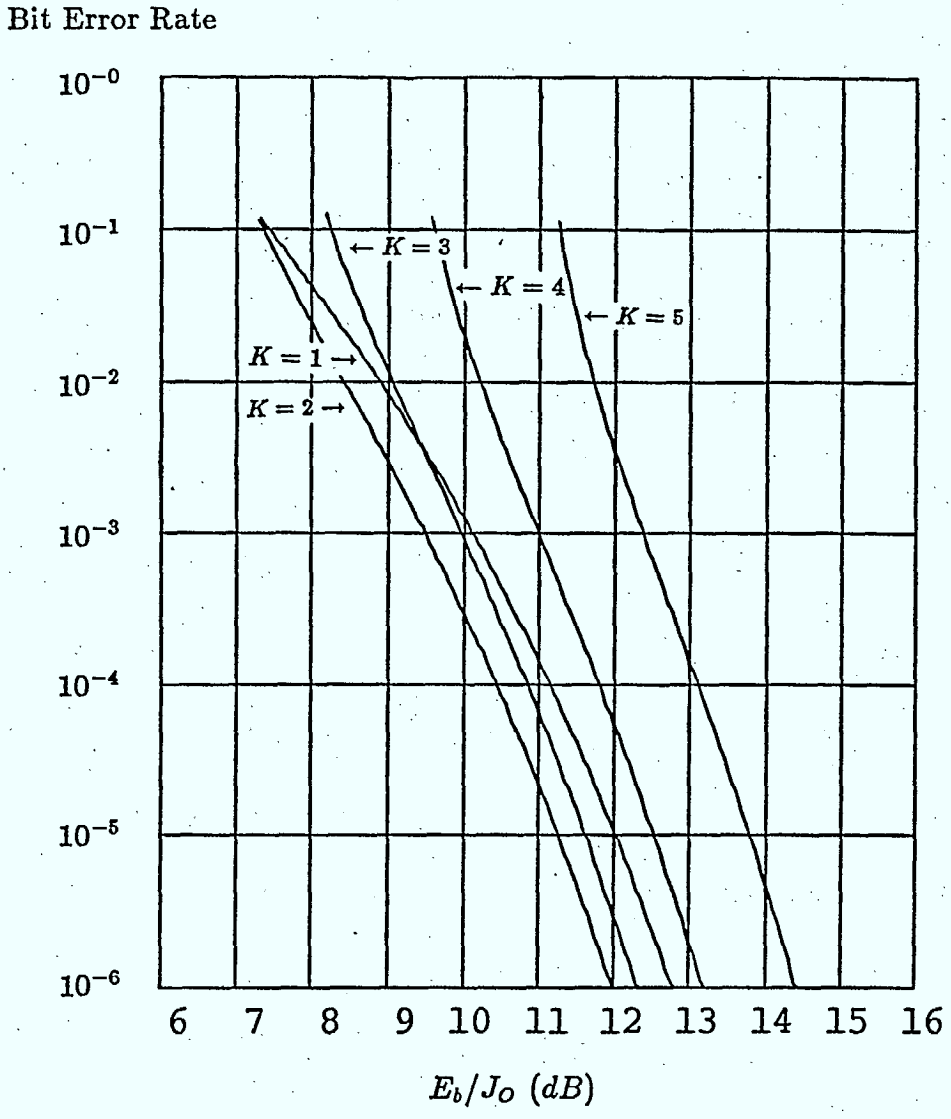


Figure 2.4: BER performance of the dual-K rate 1/2 convolutional codes for K equal to 1 through 5, under WC MT jamming. The best code is that for $K = 2$, for which $\alpha_{wc} = 0.683$ at a BER of 10^{-5}.

2.2.3 Odenwalder Binary Codes

Two commonly used binary convolutional codes are the constraint length 7, rate 1/2 and 1/3 codes discovered by Odenwalder[7]. For $r = 1/2$,

$$\begin{aligned} G(D) &= 36D^{10} + 211D^{12} + 1404D^{14} + 11,633D^{16} \\ &\quad +77,433D^{18} + 502,690D^{20} + 3,322,763D^{22} \\ &\quad +21,292,910D^{24} + 134,365,911D^{26} + \cdots; \end{aligned} \tag{2.17}$$

and for $r = 1/3$,

$$G(D) = D^{14} + 20D^{16} + 53D^{18} + 184D^{20} + \cdots. \tag{2.18}$$

Rate 1/4 and 1/8 codes having the same constraint length, 7, can be derived from the above rate 1/2 code[4]. For $r = 1/4$,

$$\begin{aligned} G(D) &= 36D^{20} + 211D^{24} + 1404D^{28} + 11,633D^{32} \\ &\quad +77,433D^{36} + 502,690D^{40} + 3,322,763D^{44} \\ &\quad +21,292,910D^{48} + 134,365,911D^{52} + \cdots; \end{aligned} \tag{2.19}$$

and for $r = 1/8$,

$$\begin{aligned} G(D) &= 36D^{40} + 211D^{48} + 1404D^{56} + 11,633D^{64} \\ &\quad +77,433D^{72} + 502,690D^{80} + 3,322,763D^{88} \\ &\quad +21,292,910D^{96} + 134,365,911D^{104} + \cdots. \end{aligned} \tag{2.20}$$

The BER performance of the rate 1/3 code under PBN and MT jamming is shown in Fig. 2.5, and of the rate 1/2, 1/4 and 1/8 codes under PBN jamming in Fig. 2.6. In [4], the three curves in Fig. 2.6 are the same as that for the rate 1/2 code. This is incorrect, since in that case, for the rate 1/4 and 1/8 codes, L_{opt} would be smaller than 1 for a BER larger than 10^{-6}, which is not realizable. The BER performance of the rate 1/2 and 1/4 codes under MT jamming is shown in Fig. 2.7. From Figs. 2.5 to 2.7 it can be seen that the $r = 1/3$ code performs best.

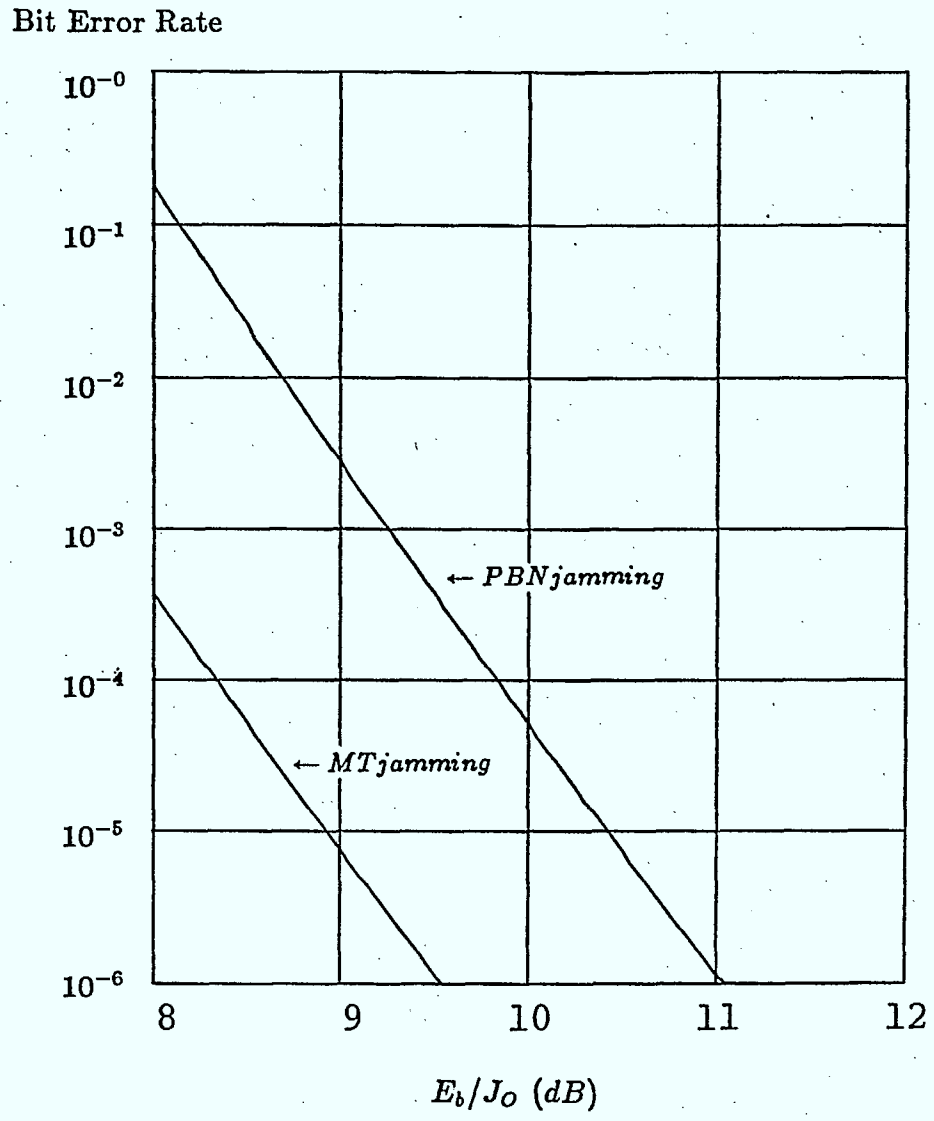


Figure 2.5: BER performance of the constraint length 7 binary Odenwalder convolutional rate 1/3 code, with FH/BFSK. At $P_b = 10^{-5}$, $\rho_{wc} = 0.815$ for WC PBN jamming, and $\alpha_{wc} = 1\_$ for WC MT jamming.

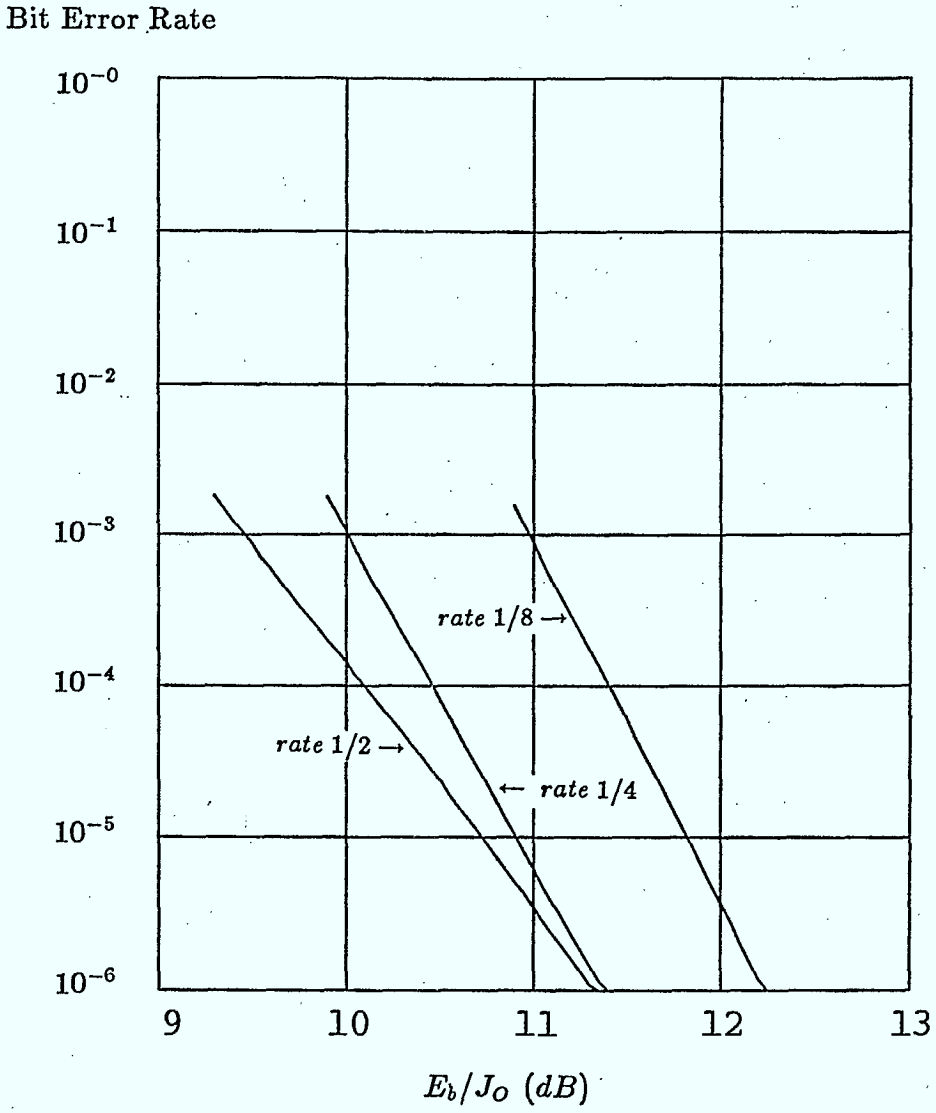


Figure 2.6: BER performance of the constraint length 7 binary Odenwalder convolutional rate 1/2 , 1/4 and 1/8 codes with FH/BFSK, under WC PBN jamming. At $P_b = 10^{-5}$, $\rho_{wc} = 0.75$ for the rate 1/2 code, $\rho_{wc} = 0.975$ for the rate 1/4 code and $\rho_{wc} = 1$ for the rate 1/8 code.

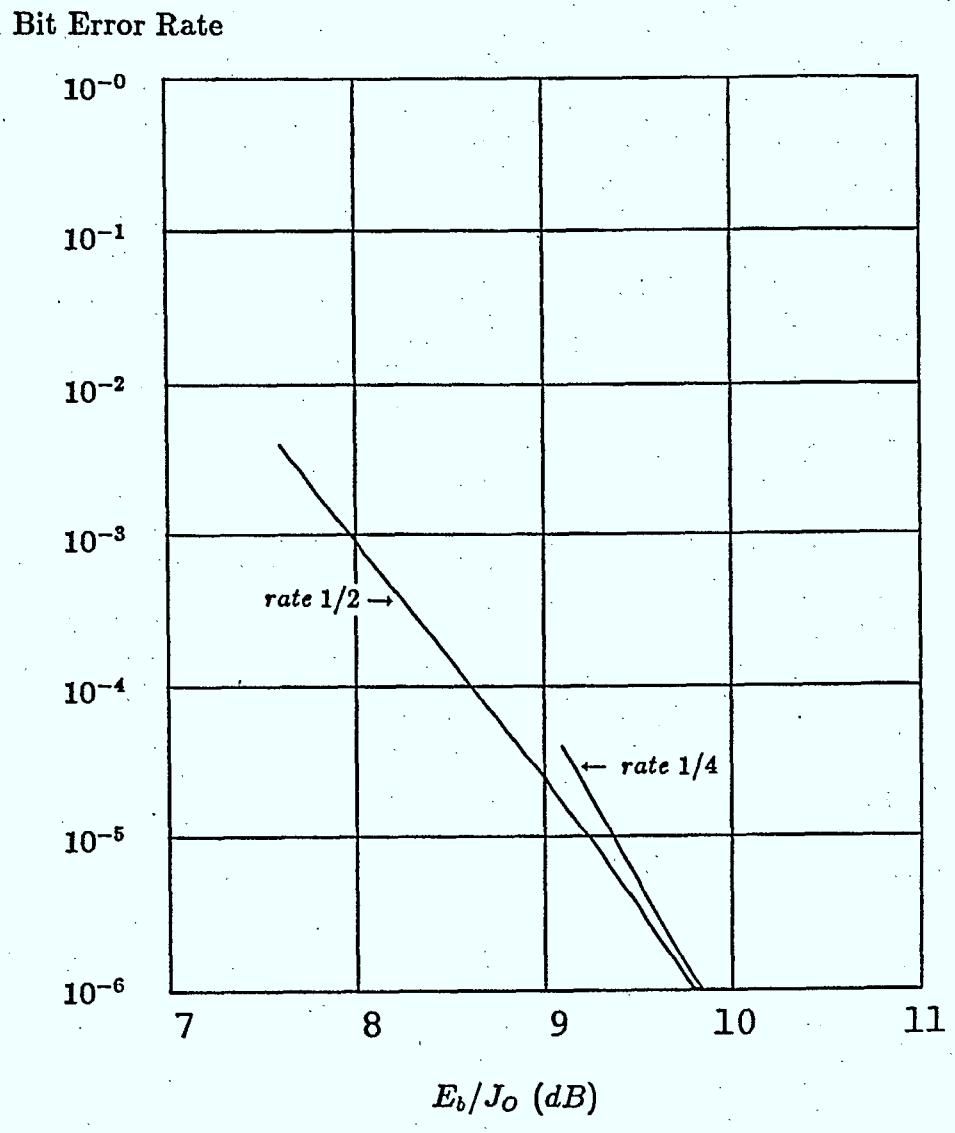


Figure 2.7: BER performance of the constraint length 7 binary Odenwalder convolutional rate 1/2 and 1/4 codes with FH/BFSK, under WC MT jamming. At $P_b = 10^{-5}$, $\alpha_{wc} = 1$ for both the rate 1/2 and rate 1/4 codes.

2.2.4 M-ary Orthogonal Convolutional Codes

M-ary orthogonal convolutional codes are a class of codes with constraint length K[1,8], hence $K \geq 2$. They are $(M, 1, K)$ binary convolutional codes where each M-bit at the output of the encoder, corresponding to 1 bit at the input, is one of M orthogonal binary sequences of dimension M. Thus one such M-bit corresponds to one M-ary symbol. In this case, R' is always equal to 1 and we have

$$G(D) = \frac{D^K(1-D)^2}{(1-2D+D^K)^2}. \tag{2.21}$$

The best code in this class is that for $K = 2$ with BER performance shown in Fig. 2.8. Due to its short constraint length, the performance of this code is relatively poor.

2.2.5 Semi-Orthogonal Convolutional Codes

For $K \geq 3$, we have the class of semi-orthogonal M-ary convolutional codes with constraint length $2K+1$[1,4]. As for the orthogonal convolutional codes, R' is always equal to 1 for the semi-orthogonal convolutional codes. $G(D)$ is given by

$$G(D) \approx D^{2K+1}. \tag{2.22}$$

The BER performance of these codes is shown in Fig. 2.9 for K equal to 3, 4 and 5. The code corresponding to $K = 3$ is the best in this class.

2.3 Performance of Block Codes

2.3.1 Binary Codes

Although binary codes can be used directly with BFSK ($K = 1$, as in [4]), they can be used with any M-ary signalling through interleaving. In this regard, hard decision decoded block codes are easier to deal with than soft decision decoded convolutional codes. Interleaving of binary block codes is used to ensure that the K bits in one M-ary channel

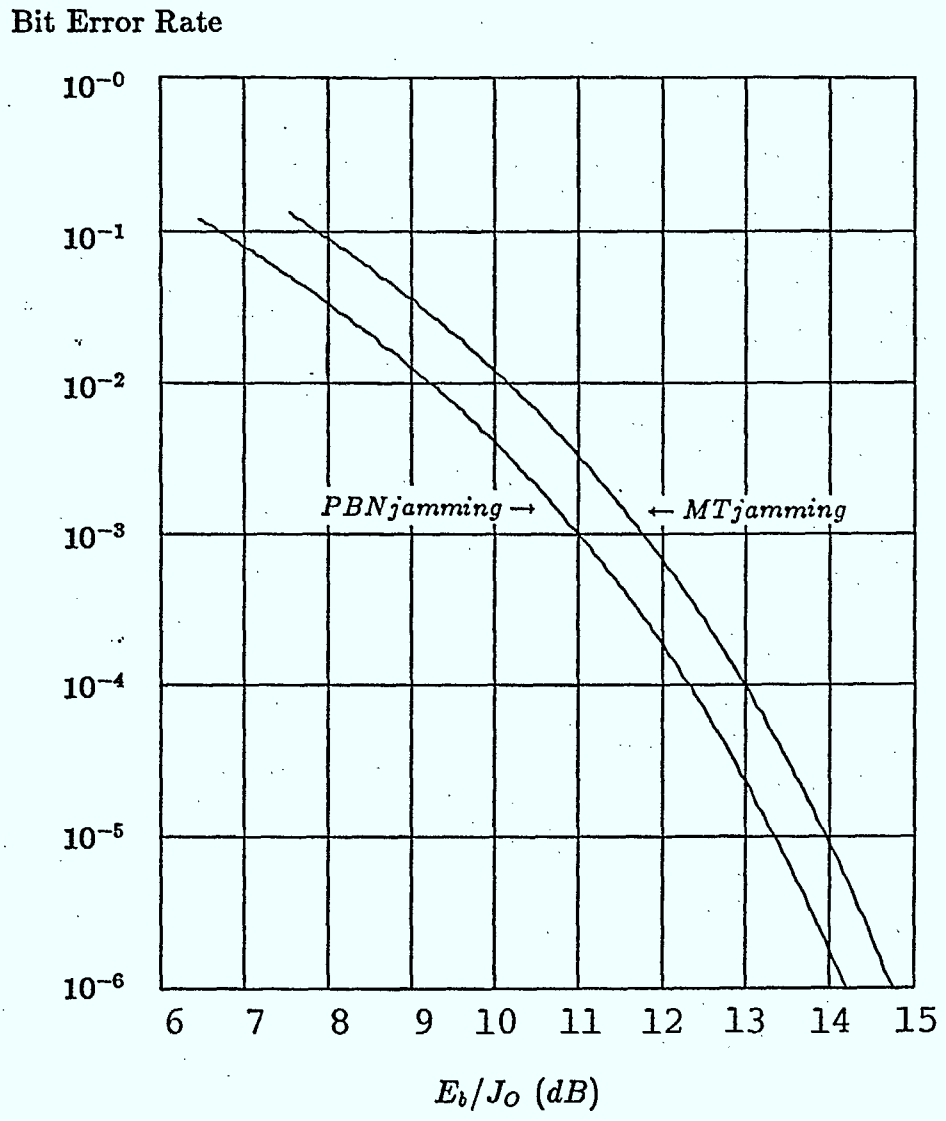


Figure 2.8: BER performance of the 4-ary orthogonal convolutional code with FH/4FSK. At $P_b = 10^{-5}$, $\rho_{wc} = 0.75$ for WC PBN jamming and $\alpha_{wc} = 0.683$ for WC MT jamming.

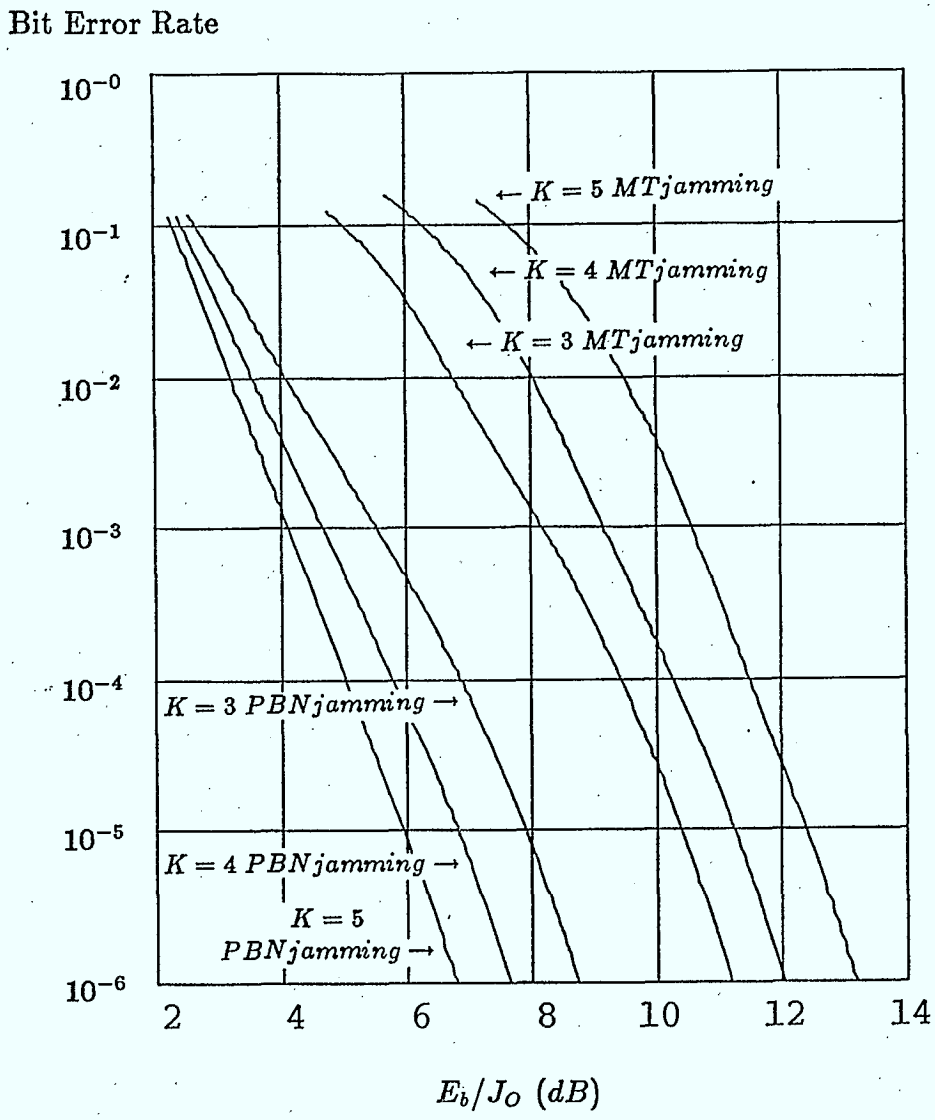


Figure 2.9: BER performance of the semi-orthogonal M-ary convolutional codes for K equal to 3, 4 and 5. The constraint length is $2K+1$. The best code in this class is that for $K = 3$. For this code, at a BER of 10^{-5}, $\rho_{wc} = 0.75$ for WC PBN jamming, and $\alpha_{wc} = 0.527$ for WC MT jamming.

symbol belong to K code words, so that in each code word the errors will not occur in bursts. In this case, (2.8) to (2.11) still apply, but now $K > q$ so that

$$P_Q = \frac{M}{2(M-1)} P_K, \tag{2.23}$$

with $Q = 2$ and $q = 1$.

There are two parameters which can be varied in order to minimize P_b, K and the EC code to be used, i.e., n and k. For any fixed code of rate r, from (2.1) to (2.7) and (2.11), it can be seen that we have in fact an uncoded channel with equivalent channel bit energy rE_b. For this channel, it has been shown that under worst case jamming and with optimum diversity, $K = 2$ is optimum [3]. Thus we can say that for a fixed code with optimum diversity, $K = 2$ is the optimum signalling and in this case WC multitone jamming is worse than WC partial band noise jamming. For a fixed K, if there is an optimum code (among a class of codes), then as argued above the same code will perform best when $K = 2$. Thus all we need to do is to find the optimum code used with 4-ary FSK.

Hamming Codes

We examined the (7,4) and (31,26) Hamming codes which are single error correcting with minimum distance $d = 3$.

Golay Code

We examined the perfect (23,12) Golay code which is triple error correcting with minimum distance $d = 7$.

BCH Codes

We examined the multi-error correcting BCH codes of length $n = 15$, 31, 63, 127 and 255.

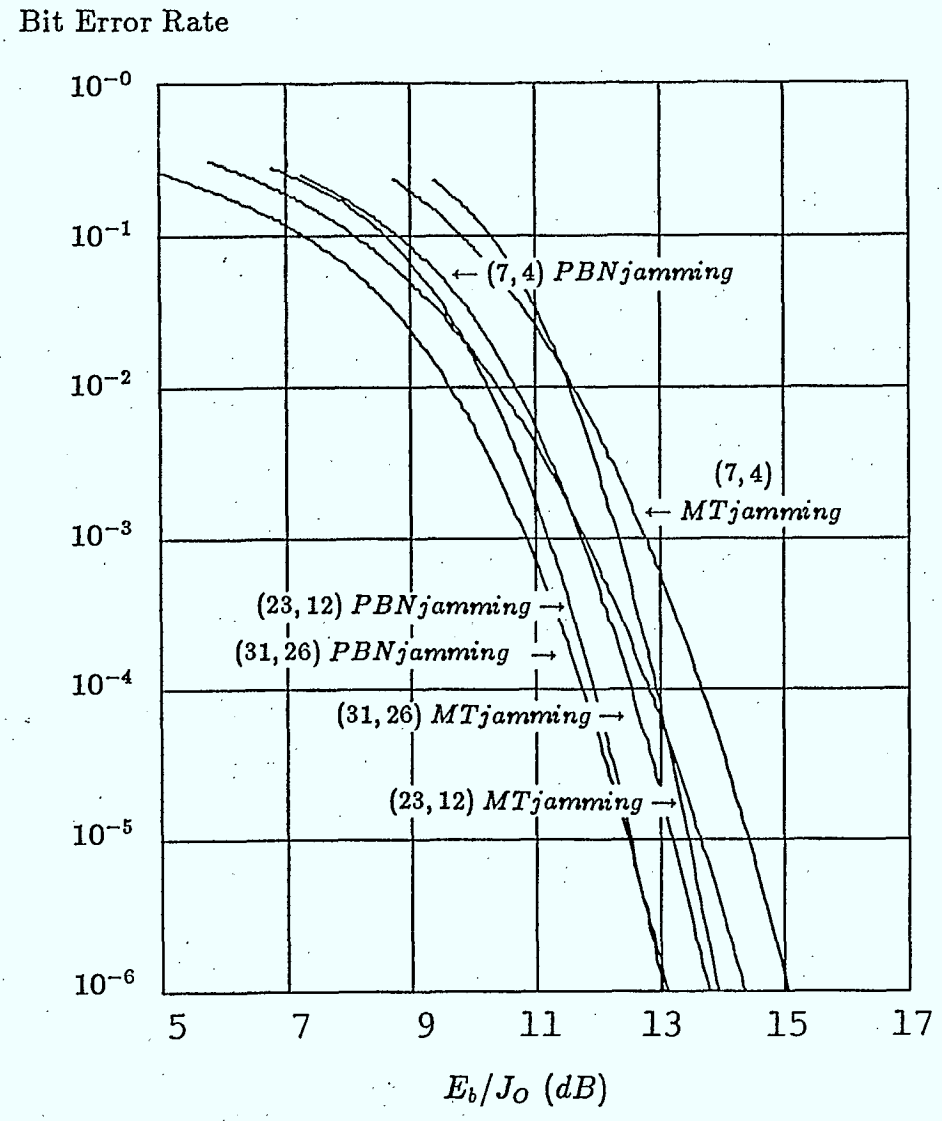


Figure 2.10: BER performance of the (7,4) and (31,26) Hamming codes and the (23,12) Golay code with FH/4FSK under worst case jamming.

Table 2.2: E_b/J_O required for $P_b = 10^{-5}$ with Binary Block Codes under Worst Case Jamming, $K = 2$.

block code	E_b/J_O for WC PBN jamming	E_b/J_O for for WC MT jamming
(7,4)	13.67	14.43
(23,12)	12.54	13.47
(31,26)	12.50	13.22
(15,11)	12.85	13.60
(31,21)	12.37	13.19
(63,51)	11.94	12.73
(127,99)	11.45	12.30
(255,187)	11.01	11.95

Summary of Results

The BER performance of the Hamming codes and the Golay code is shown in Fig. 2.10 and the E_b/J_O required for $P_b = 10^{-5}$ under WC jamming is given in Table 2.2. It is evident that the (31,26) Hamming code is the best of these short block length codes, at $P_b = 10^{-5}$, followed by the (23,12) Golay code (for $K = 1$, the Golay code is better).

In Fig. 2.11 the E_b/J_O required for $P_b = 10^{-5}$ is given for the BCH codes, vs code rate, under WC MT jamming. It is noted that for code lengths larger than 15, there are several codes of a similar rate which give near optimum BER performance. The optimum BCH codes (in the sense that they require the smallest E_b/J_O for $P_b = 10^{-5}$ while having the largest code rate r), are the (15,11), (31,21), (63,51), (127,99) and (255,187) codes for lengths 15, 31, 63, 127 and 255, respectively. The BER performance of these optimum codes is shown in Figs. 2.12 and 2.13 and given in Table 2.2 for $P_b = 10^{-5}$. It can be seen that 0.41 to 0.45 dB can be gained at $P_b = 10^{-5}$ by doubling the length of a BCH code, but at the expense of increased codec complexity. However, the choice of a code must also take into account the specific application and implementation required.

2.3.2 Reed-Solomon Codes

Length $n = Q-1$, $Q = 2^q$, Reed-Solomon (RS) codes are Q-ary codes over $GF(Q)$. For (n,k) RS codes, $d = n-k+1$. These codes can be used directly with M-ary signalling, in which case $Q = M$, or with alphabet conversion so that $Q > M$.

Direct Use

By direct use we mean transmit the symbols of (n,k) RS codes directly over an M-ary channel, $K \geq 2$. In this case $P_Q = P_K$, where P_K is given in (2.10), since C in (2.13) is 1.

Alphabet Conversion

To get a larger minimum distance for RS codes with a fixed code rate, the code length must be increased. This in turn increases the size of the alphabet over which an RS code is defined. Alphabet conversion matches the channel signalling with the RS symbols. A Q-ary symbol is now composed of C M-ary symbols, as defined in (2.12) and (2.13). Note that for small P_K, $P_Q \approx CP_K$. This means that there is a multiplication in the error probability by a factor C due to the conversion.

Summary of Results

The best RS codes were found in the following way. For a given code length we have a given Q, but several choices for K. For each possible Q,K combination, the code rate was optimized for the given code length to give the lowest E_b/J_O at $P_b = 10^{-5}$ under the worst of WC PBN or WC MT jamming. The results of this optimization are given in Table 2.3. For length 7 RS codes, the (7,5) code in direct use with $K = 3$ is the best choice. For code lengths 15, 31, 63, 127, 255, 511 and 1023, the optimum combinations are, (15,11)

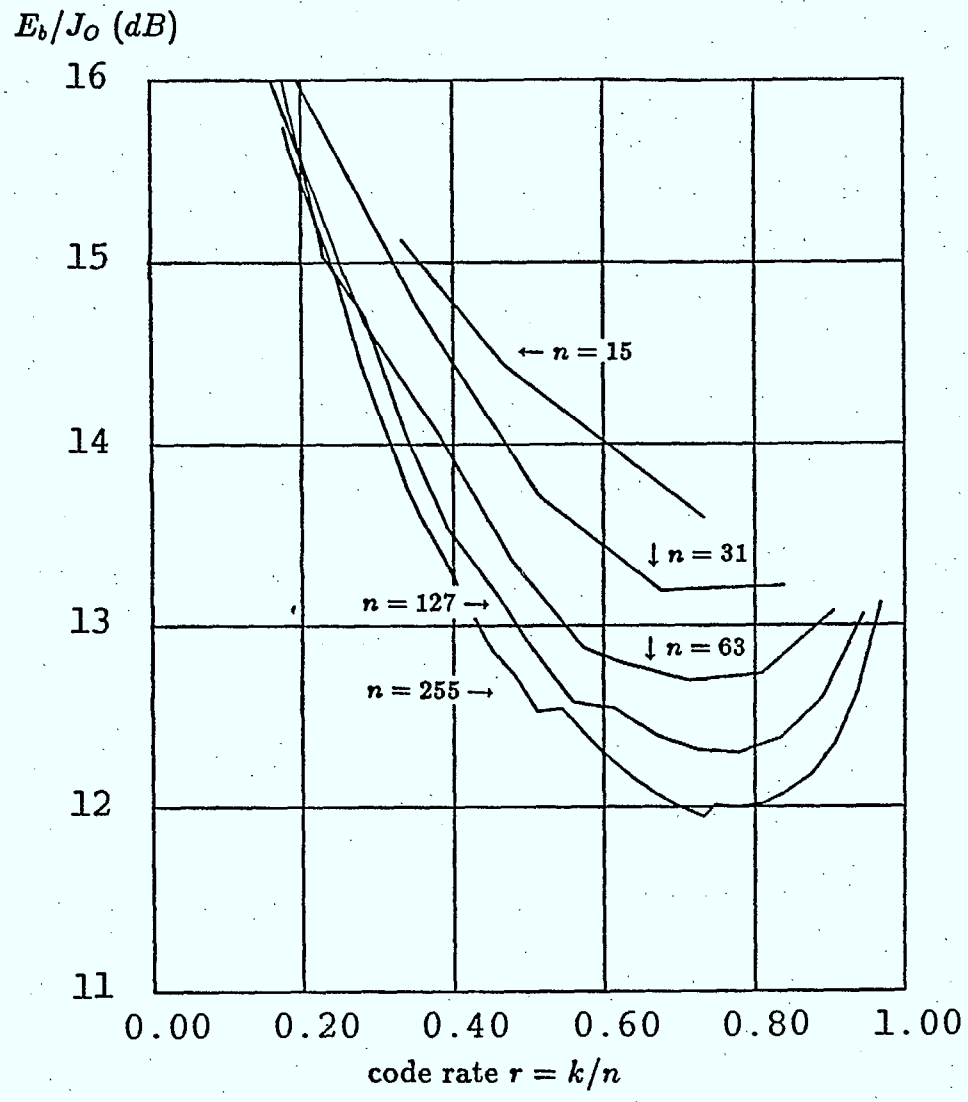


Figure 2.11: E_b/J_O required for $P_b = 10^{-5}$ with BCH codes, under WC MT jamming, vs code rate $r = k/n$, $K = 2$.

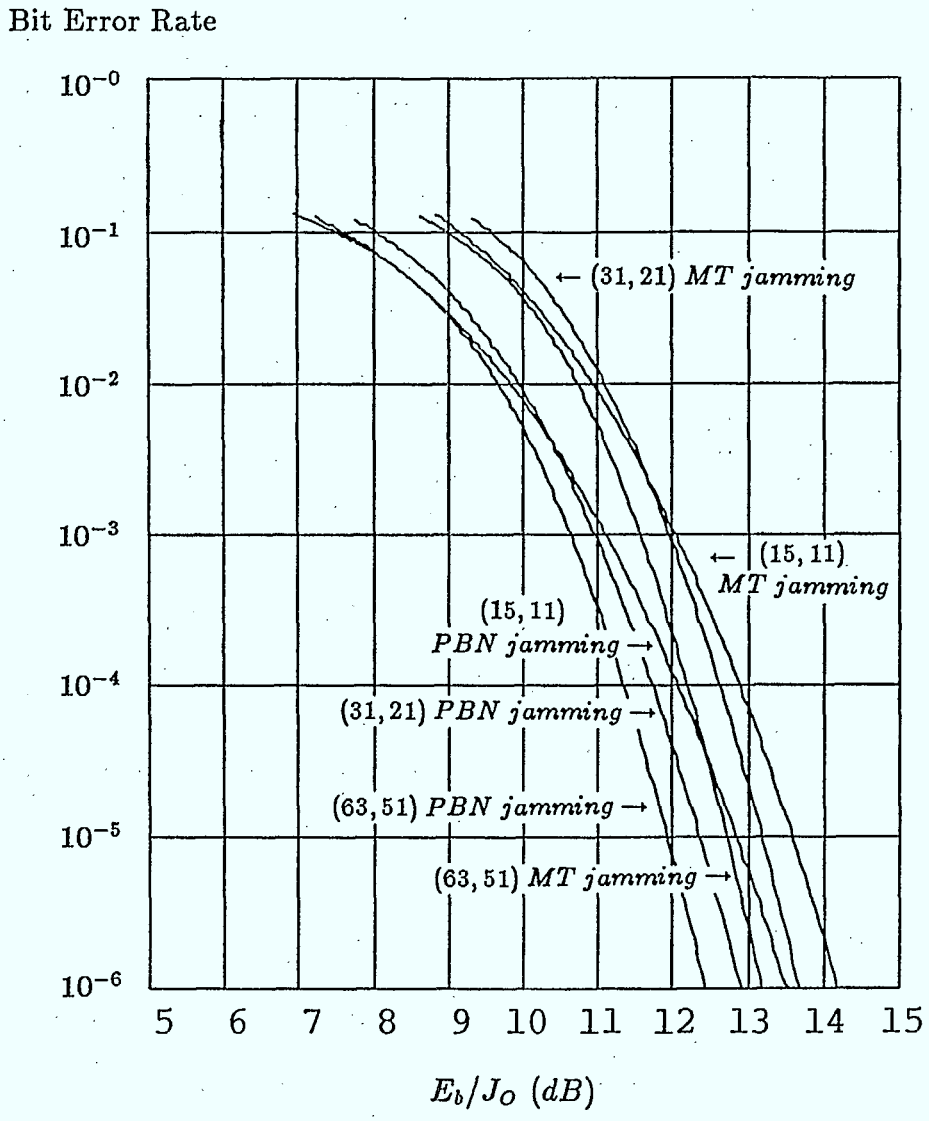


Figure 2.12: BER performance of the (15,11), (31,21) and (63,51) BCH codes with FH/4FSK for WC PBN and WC MT jamming.

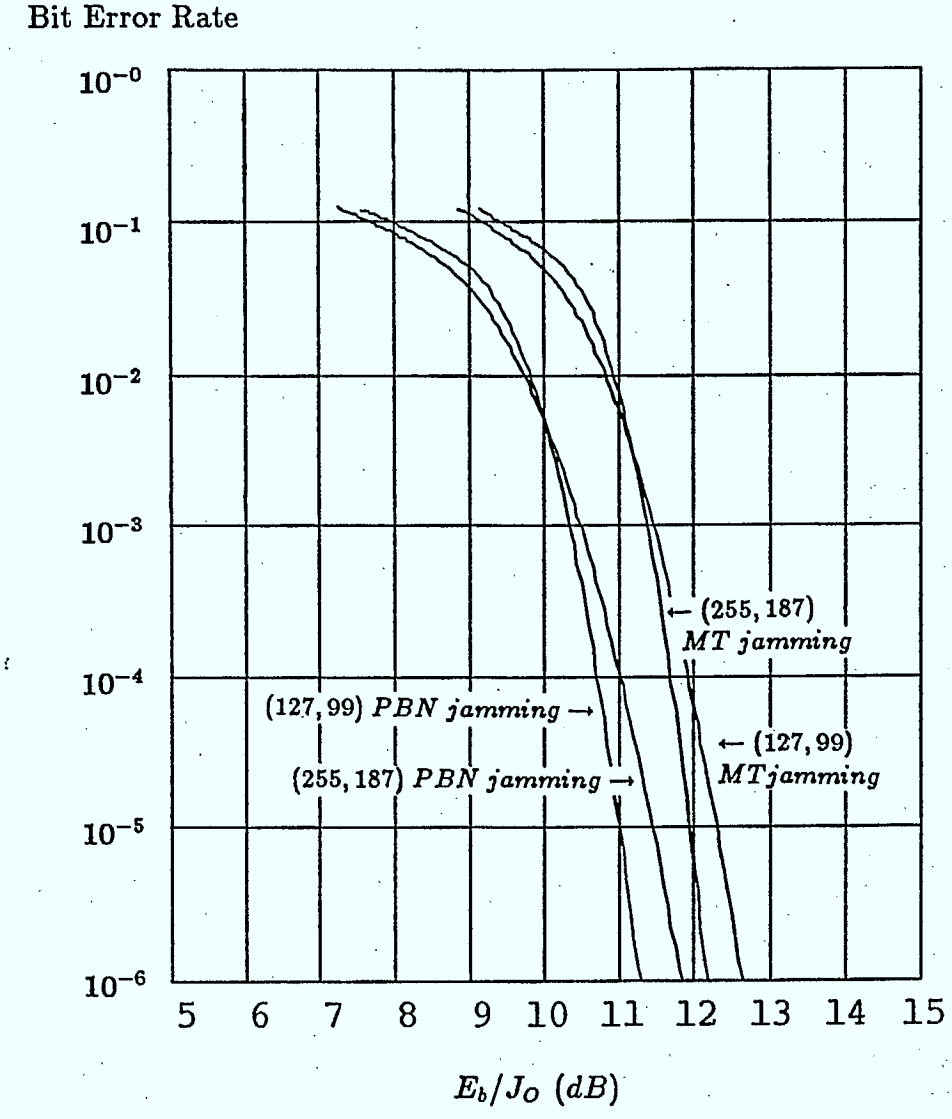


Figure 2.13: BER performance of the (127,99) and (255,187) BCH codes with FH/4FSK for WC PBN and WC MT jamming.

Table 2.3: E_b/J_O (dB) required for $P_b = 10^{-5}$ with Reed-Solomon Codes under Worst Case Jamming.

RS code	q	K	E_b/J_O for WC PBN jamming	E_b/J_O for WC MT jamming
(7,5)	3	3	11.53	14.13
(7,5)	3	1	15.80	14.50
(15,11)	4	4	9.89	14.45
(15,11)	4	2	12.30	13.09
(15,11)	4	1	15.02	13.83
(31,25)	5	5	8.71	15.28
(31,23)	5	1	14.36	13.24
(63,53)	6	6	7.89	14.37
(63,51)	6	3	9.90	12.60
(63,51)	6	2	11.32	12.17
(63,49)	6	1	13.93	12.83
(127,99)	7	1	13.60	12.55
(255,221)	8	4	8.62	13.23
(255,213)	8	2	10.85	11.74
(255,203)	8	1	13.42	12.38
(511,439)	9	3	9.38	12.11
(511,407)	9	1	13.30	12.28
(1023,899)	10	5	7.83	14.44
(1023,861)	10	2	10.68	11.58
(1023,817)	10	1	13.25	12.23

with $K = 2$, (31,23) with $K = 1$, (63,51) with $K = 2$, (127,99) with $K = 1$, (255,213) with $K = 2$, (511,439) with $K = 3$, and (1023,861) with $K = 2$, respectively. It is clear that codes with $K = 2$, which is the optimum signalling for uncoded systems, are the best. $K = 3$ is the second best choice. Q should be chosen so that these K's can be used. The BER performance of the (15,11), (63,51), (255,213) and (1023,861) RS codes with $K = 2$ is given in Fig. 2.14. The BER performance of the (7,5), (63,51) and (511,439) RS codes with $K = 3$ is given in Fig. 2.15. The E_b/J_O for $P_b = 10^{-5}$ is plotted in Fig. 2.16 for all RS codes with $K = 2$ and $K = 3$, vs code rate, under WC MT jamming. This shows that increasing the code length must be done so as to allow a good signalling scheme to be used, otherwise performance may actually decrease. From Table 2.3, it is seen that doubling the code length will not improve the performance for code length increases from 15 to 31, 63 to 127 and 255 to 511. It is obvious that alphabet conversion is essential to the optimal use of RS codes, since except for the length 7 code all other optimum combinations are through alphabet conversion.

2.4 Performance of Concatenated Codes

It is well known that concatenation of an RS outer code with an inner code can form a very powerful error correcting code[11]. An (n,k) Q-ary RS code is used as the outer code and a convolutional code or a block code is used as the inner code. Whenever necessary, interleaving is assumed between the inner code and the outer code so that the input to the outer RS decoder appears to have memoryless Q-ary symbol errors. Now $R' = \frac{k}{n} r_i K$, where r_i is the code rate of the inner code. We observe that there is a threshold effect in the BER performance of all concatenated codes. When E_b/J_O approaches the threshold from above, the BER increases rapidly, and when E_b/J_O approaches the threshold from below, the BER decreases rapidly. This sensitivity should be taken into account in the design of a system. A fast drop in BER as E_b/J_O increases is desireable when an extremely low BER must be

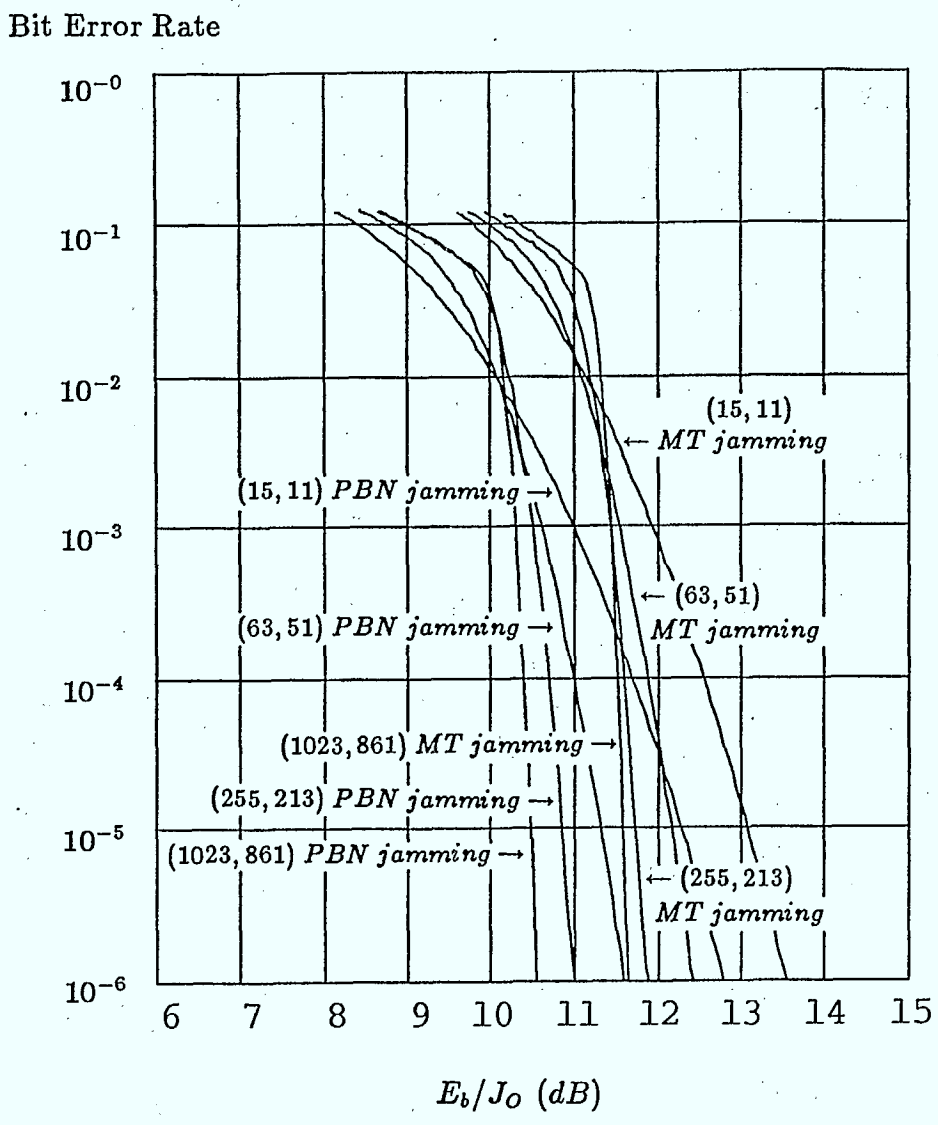


Figure 2.14: BER performance of the (15,11), (63,51), (255,213) and (1023,861) RS codes with $K = 2$ for WC PBN and WC MT jamming.

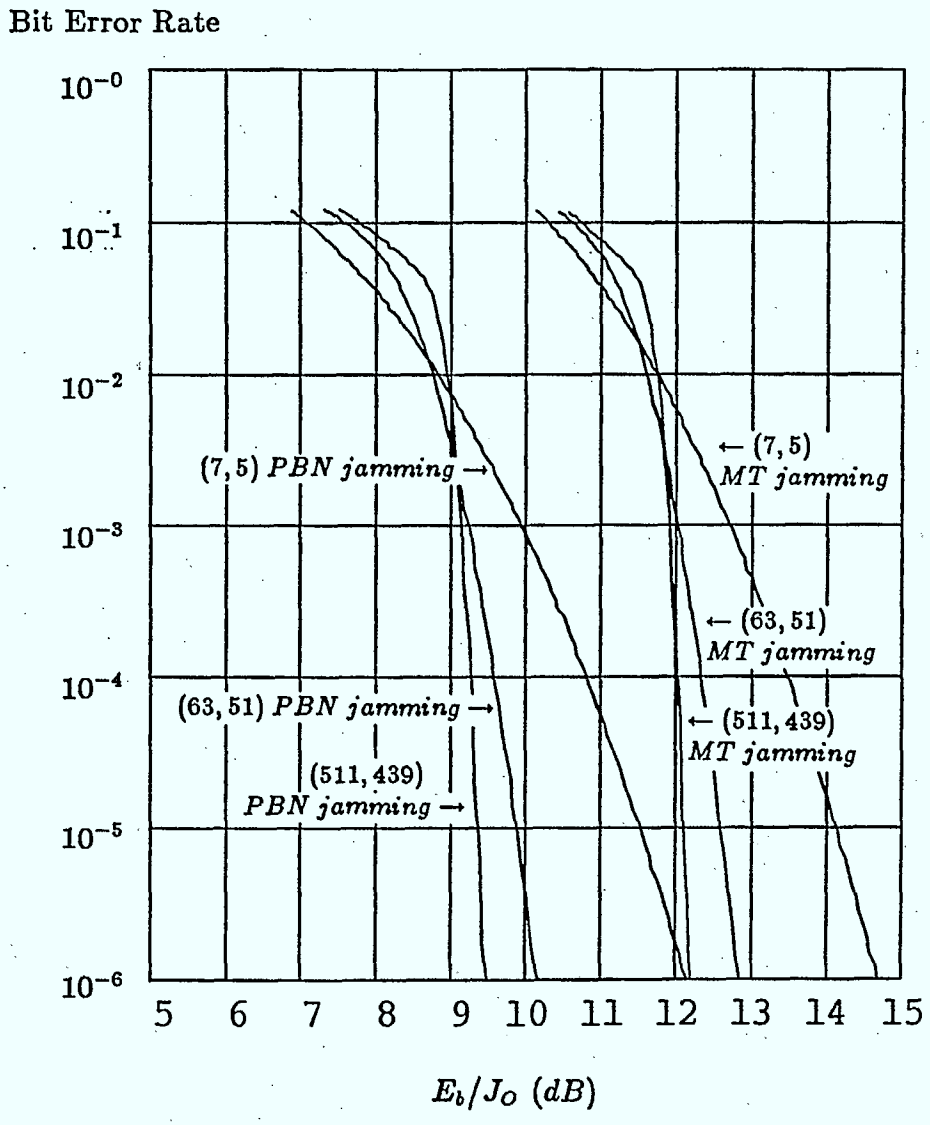


Figure 2.15: BER performance of the (7,5), (63,51) and (511,439) RS codes with $K = 3$ for WC PBN and WC MT jamming.

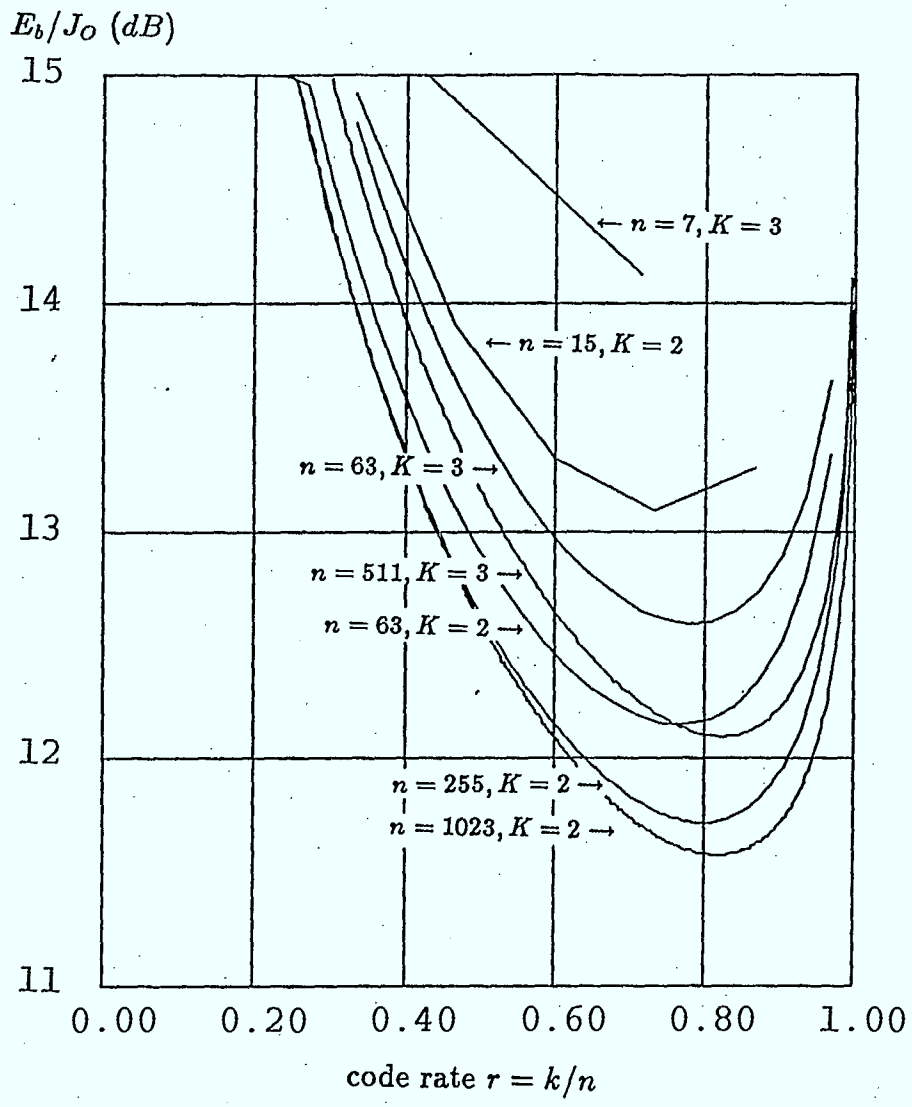


Figure 2.16: E_b/J_O (dB) required for $P_b = 10^{-5}$ with RS coding and $K = 2$ and 3, under WC MT jamming, vs code rate $r = k/n$.

achieved.

2.4.1 Convolutional Inner Codes

We can evaluate the BER performance of an RS code concatenated with a convolutional inner code by evaluating the BER performance of the outer RS code using the formulas developed for block codes, with the exception that $G(D^L)$ in (2.10) is replaced by $G(D^L)$ of the inner convolutional code given previously. Due to the assumption of soft decision decoding of convolutional codes, we only consider convolutional codes in direct use with M-ary signalling, thus there is no alphabet conversion between them. There may be an alphabet conversion between the outer and inner codes, in which case C symbols at the inner decoder output form an outer code symbol.

In [3], dual-K codes, Odenwalder binary codes and Trumpis codes were considered as inner codes. Here we consider M-ary orthogonal convolutional codes and semi-orthogonal convolutional codes as inner codes. In both encoders, one information bit at the input generates one M-ary symbol, Thus the output of the inner decoder is binary, so $C = q$ in (2.13) and P_K is replaced by (2.1), the BER at the output of the inner decoder. This is illustrated in Fig. 2.17. It is easy to see that the optimum K should be the one which is optimum for the convolutional code alone under WC MT jamming. That is, $K = 2$ for M-ary orthogonal and $K = 3$ for semi-orthogonal codes. If we optimize the code rate for each outer code length, we find longer codes always give better performance. This means that the larger error multiplication due to alphabet conversion is always offset by the larger minimum distance due to the code length increase.

M-ary Orthogonal Convolutional Inner Codes

In Table 2.4 we list the optimum RS outer codes for lengths 15, 31, 63, 127 and 255 together with the required E_b/J_O for $P_b = 10^{-5}$, and the improvement over using the convolutional code alone under WC MT jamming. Note that for $K = 2$, the M-ary or-

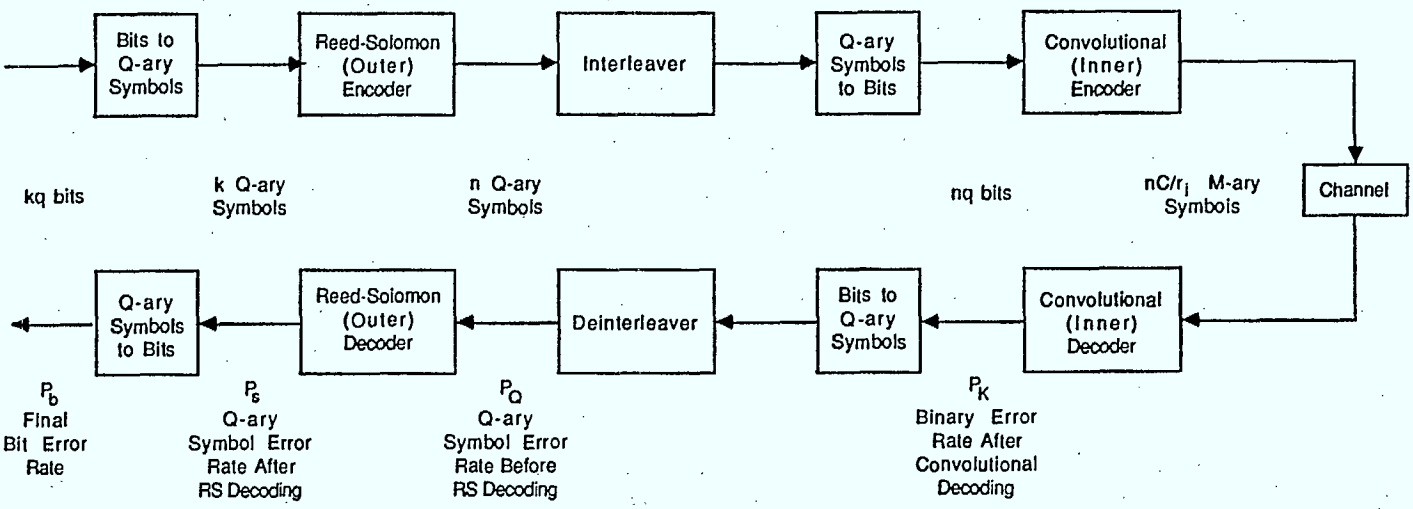


Figure 2.17: Block diagram of a system employing a concatenated coding scheme with a Reed-Solomon outer code and a convolutional inner code.

Table 2.4: E_b/J_O (dB) required for $P_b = 10^{-5}$ with a Reed-Solomon Outer Code and an M-ary Orthogonal Convolutional Inner Code under Worst Case Jamming, $K = 2$.

RS code	q	E_b/J_O for WC PBN jamming	E_b/J_O for WC MT jamming	improvement over inner code used alone
(15,11)	4	12.09	12.88	1.07
(31,23)	5	11.48	12.33	1.62
(63,49)	6	11.07	11.93	2.02
(127,103)	7	10.81	11.68	2.27
(255,211)	8	10.64	11.52	2.53

Table 2.5: E_b/J_O (dB) required for $P_b = 10^{-5}$ with a Reed-Solomon Outer Code and a Semi-Orthogonal Convolutional Inner Code under Worst Case Jamming, $K = 3$.

RS code	q	E_b/J_O for WC PBN jamming	E_b/J_O for WC MT jamming	improvement over inner code used alone
(15,11)	4	6.63	9.29	1.12
(31,25)	5	6.12	8.78	1.63
(63,51)	6	5.75	8.40	2.01
(127,105)	7	5.51	8.17	2.24
(255,217)	8	5.36	8.02	2.39

thogonal code has a very short constraint length. Hence when the code is used alone, the performance is relatively poor, (as seen previously). By concatenation with an RS outer code we can improve the performance by about 1 to 2.5 dB as seen in Table 2.4.

Semi-Orthogonal Convolutional Inner Codes

In Table 2.5 we list the optimum RS outer codes for lengths 15, 31, 63, 127 and 255 together with the required E_b/J_O for $P_b = 10^{-5}$ and $K = 3$, and the improvement over using the convolutional code alone under WC MT jamming. The gain in performance is about 1 to 2.4 dB, making these codes an excellent choice. For example, a (255,217) RS outer code concatenated with the semi-orthogonal code of constraint length 7 (for $K = 3$) over 8-ary FSK performs 0.5 dB better than the Trumpis code with 4-ary FSK concatenated with a RS code of the same length presented in [3]. In fact, it performs better than the Trumpis code with a RS code of length 1023 which is the best code given in [3].

2.4.2 Block Inner Codes

With hard decision decoding, alphabet conversion can be used between the inner block code and the M-ary signalling through interleaving. Only binary block inner codes are considered. In this case, as discussed previously, the 4-ary channel ($K = 2$), is optimum.

Let the inner block code be an (n_i, k_i) block code with minimum distance d_i, and

$$t_i = \lfloor (d_i - 1)/2 \rfloor.$$

Assume hard decision decoding is used for the inner block code. We can then evaluate the final BER performance of the concatenated RS outer code and (n_i, k_i) block inner code by evaluating the BER performance of the outer RS code using (2.8) and (2.9) with P_Q given as follows. A symbol error occurs at the input to the RS decoder only when the inner decoder fails to decode an inner code word correctly. Thus if the inner decoder input BER is small, we may assume (as an approximation so that the weight distribution of the inner code is not required in the calculation) that, conditioned on the decoding failure, the inner code decoder output has bit error rate d_i/n_i. That is,

$$P_Q \approx (1 - (1 - d_i/n_i)^q) \sum_{j=t_i+1}^{n_i} \binom{n_i}{j} P_K^j (1 - P_K)^{n_i - j}, \tag{2.24}$$

with P_K defined by

$$P_K \leq \begin{cases} \frac{1}{2} G(D^L) & \text{PBN jamming;} \\ G(D^L) & \text{MT jamming,} \end{cases} \tag{2.25}$$

and $G(D^L)$ as in (2.11). It is noted that the above result would be the same if derived from the more rigorously proved result in [10] when d_i/n_i and P_K are small.

In Table 2.6 we list the optimum RS outer codes and BCH inner codes for a given code length together with the required E_b/J_O for $P_b = 10^{-5}$ and the improvement over BCH codes used alone under WC MT jamming. We also examined codes with the (23,12) Golay and (7,4) Hamming codes as the inner codes, and these results are given in Table 1.6 as well. The performance of the best code in this class, the (255,241) RS outer code with the (255,207) BCH inner code, is shown in Fig. 2.18. It is clear that if the inner block code has a large minimum distance, so that the BER drops sharply as E_b/J_O increases, then concatenation with an RS code will not give much gain in E_b/J_O for a given P_b. This type of behavior is exhibited in Fig. 2.18. (Imagine moving a nearly vertical line

Table 2.6: E_b/J_O (dB) required for $P_b = 10^{-5}$ with a Reed-Solomon Outer Code and a Binary Block Inner Code under Worst Case Jamming, $K = 2$.

RS outer code	inner block code	E_b/J_O for WC PBN jamming	E_b/J_O for WC MT jamming	improvement over inner code used alone
(15,13)	(15,11)	12.32	13.19	0.41
(31,25)	(15,11)	11.78	12.74	0.86
(63,53)	(15,11)	11.41	12.40	1.20
(15,13)	(31,26)	12.09	12.91	0.31
(31,27)	(31,26)	11.68	12.54	0.88
(63,53)	(31,26)	11.30	12.22	1.00
(127,111)	(31,26)	11.08	12.01	1.21
(31,27)	(63,51)	11.47	12.37	0.36
(63,57)	(63,51)	11.23	12.14	0.59
(127,115)	(63,51)	11.02	11.96	0.77
(255,235)	(63,51)	10.91	11.85	0.88
(63,57)	(127,106)	11.09	12.01	0.37
(127,117)	(127,106)	10.94	11.87	0.51
(255,239)	(127,106)	10.85	11.77	0.61
(63,59)	(255,207)	10.94	11.87	0.15
(127,119)	(255,207)	10.81	11.76	0.26
(255,241)	(255,207)	10.73	11.68	0.34
(7,5)	(7,4)	13.52	14.47	-0.03
(15,13)	(7,4)	12.90	13.82	0.61
(31,25)	(7,4)	12.20	13.26	1.17
(63,51)	(7,4)	11.72	12.86	1.57
(127,107)	(7,4)	11.40	12.56	1.87
(255,217)	(7,4)	11.15	12.34	2.09
(15,13)	(23,12)	12.27	13.33	0.14
(31,27)	(23,12)	11.91	13.02	0.45
(63,55)	(23,12)	11.58	12.75	0.72
(127,115)	(23,12)	11.38	12.56	0.91
(255,227)	(23,12)	11.21	12.41	1.06

downward, the sharper the slope, the less horizontal movement there will be.) This implies a complimentary nature between convolutional codes and RS block codes when they are concatenated, because the BER for a convolutional code usually drops slowly as E_b/J_O increases.

2.5 Concluding Remarks

The best codes considered in this Chapter, and some from [3] are compiled in Table 2.7. Overall, if code complexity is ignored, the concatenated (255,241) RS outer code with the semi-orthogonal convolutional inner code with 8-ary FSK has the best performance. If only non-concatenated codes are considered, the rate 1/2 Trumpis code, followed by the rate 1/3 Odenwalder and $K = 3$ semi-orthogonal codes, are the best. This Trumpis code is 1.41 dB worse than the best concatenated code. Optimum diversity is assumed in all cases, and side information is used to eliminate jammed hops in diversity combination unless all L_{opt} hops of an M-ary symbol are jammed, in which case a linear combination is assumed. Although the optimization of the code design for a specific application must take into account other constraints such as the complexity restriction, the results given here serve as a guide to the performance of EC codes in a FFH/MFSK system. Other codes may be examined using the same basic principles given here.

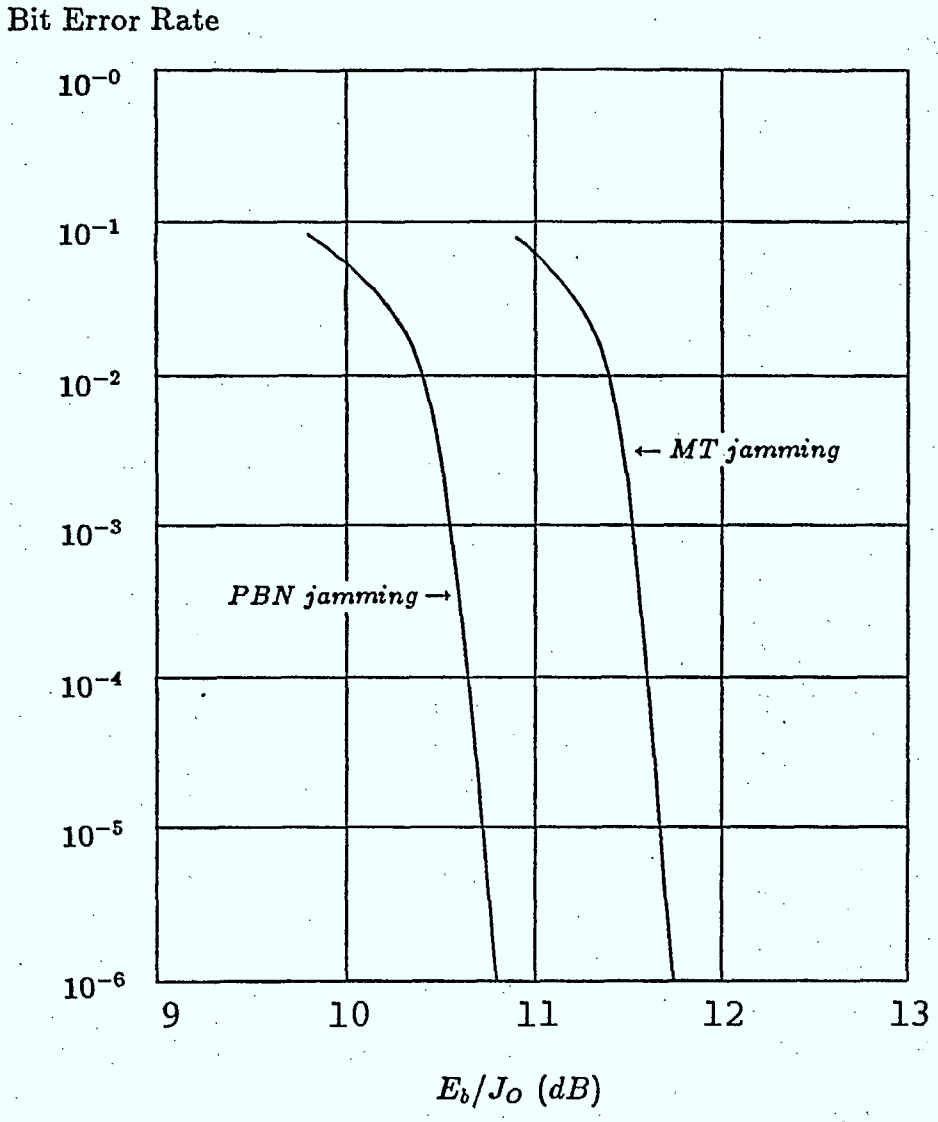


Figure 2.18: BER performance of the concatenated code with a (255,241) RS outer code and a (255,207) BCH inner code with FH/4FSK under worst case jamming.

Table 2.7: E_b/J_O (dB) required for $P_b = 10^{-5}$ with the Best Codes in each Class under Worst Case Jamming.

error correcting codes	K	E_b/J_O for WC PBN jamming	E_b/J_O for WC MT jamming
rate 1/2 Trumpis	2	8.91	9.43
Dual-K M-ary	2	10.70	11.27
rate 1/3 Odenwalder	1	10.38	8.89
M-ary Orthogonal	2	13.35	13.95
Semi-Orthogonal	3	7.91	10.41
(255,187) BCH	2	11.01	11.95
(255,213) RS	2	10.85	11.74
(511,439) RS	3	9.38	12.11
(1023,861) RS	2	10.68	11.58
Outer (255,211) RS Inner M-ary Orthogonal	2	10.64	11.52
Outer (255,217) RS Inner Semi-Orthogonal	3	5.36	8.02
Outer (255,241) RS Inner (255,207) BCH	2	10.73	11.68

Chapter 3

Performance of Error Correcting Codes for Fast Frequency Hopped Noncoherent *M*FSK Spread Spectrum Communications with a Fixed Hop Rate

3.1 Introduction

In Chapter 2, we evaluated the performance of various error correcting (EC) codes in a FFH/*M*FSK system with a fixed data rate. In this case, optimum diversity is assumed and E_b/J_O is the basis for EC code comparison, where E_b is the energy per information bit and J_O is the jamming spectral density. This implies a fixed data rate R_b (the average signal power S is considered to be fixed in all cases), and a variable hop rate R_h. For reasons discussed in [2], we now consider coding performance under the condition that the hop rate R_h is fixed and the data rate R_b is variable. This constraint has been given little consideration in previous work. A fixed hop rate is a practical requirement for satellite communications when multiple users access the same onboard dehopper. This constant hop rate is determined by many factors, such as the response time of a potential repeat-

back jammer and the synchronisation capability of communication receivers. Thus in this Chapter we consider the hop rate fixed. As in Chapter 2, the Chernoff union bound method is used for the performance evaluation.

As mentioned above, the average signal power S is fixed. Since the hop rate R_h is fixed, so is the energy per hop $E_h = S/R_h$. In this case the term optimum diversity is meaningless, because the diversity factor, L (the number of hops per M-ary symbol), is no longer an independent parameter. Specifically, L is given by

$$L = \frac{rK}{R_b/R_h} = \frac{R'}{R_b/R_h} \tag{3.1}$$

where r is the EC code rate and $M = 2^K$. R' is the code rate in data bits per M-ary symbol before diversity. Note that (3.1) must satisfy the restriction $L \geq 1$. This means that R_b/R_h cannot exceed the upper limit R'. From (3.1) it is clear that L varies continuously as R_b/R_h is varied. Although this implies a value of L which may not be an integer, it is useful and convenient to have the results in this finer form. For a given BER, the information bit rate R_b reflects the throughput of the system. A larger R_b means a larger throughput.

E_b/J_O is determined by

$$E_b/J_O = \frac{S}{R_b J_O} = \frac{S/R_h}{(R_b/R_h)J_O} = \frac{E_h/J_O}{R_b/R_h}. \tag{3.2}$$

We can see that for a fixed hop rate, E_b/J_O depends on R_b/R_h and E_h/J_O (which is fixed as mentioned above). Thus we will use E_h/J_O as a basic parameter to evaluate the system performance rather than E_b/J_O. This results in two system performance criteria. One is R_b/R_h, reflecting the system throughput, and the other is the more traditional BER, P_b. It is noted that to determine R_b/R_h for a fixed E_h/J_O, P_b must be fixed. In fact, as mentioned above, only for a given P_b can R_b/R_h reflect the throughput in a meaningful way. On the other hand, to determine P_b, R_b/R_h and therefore L must be given. This method of evaluating the system performance is equivalent to the P_b versus E_b/J_O format, for a given R_b/R_h as given in (3.2). Another useful format is P_b versus R_b/R_h for a given

E_h/J_O, which shows explicitly the tradeoff between them. It is not difficult to see that these formats present the same results in different ways. In this report, we focus on the R_b/R_h versus E_h/J_O format, for a given P_b, chosen to be 10^{-5}. Without further specification, performance will be evaluated by this criteria.

System assumptions are the same as in Chapters 1 and 2, with the critical exception that there is no optimum diversity. In the next section, we present the basic formulas for performance evaluation and the results for uncoded systems. In the following sections we evaluate the performance of various convolutional codes and block codes. The results obtained with a fixed hop rate are compared with those for a fixed data rate, given in Chapter 2.

3.2 Basic Formulas and the Results for Uncoded Systems

As given previously, the probability of bit error, P_b, for convolutional codes is upperbounded by the Chernoff union bound[3] as

$$P_b \leq \begin{cases} \frac{1}{2}G(D^L) & \text{PBN jamming;} \\ G(D^L) & \text{MT jamming,} \end{cases} \tag{3.3}$$

but now

$$D = \begin{cases} \frac{4e^{-1}}{E_h/J_O} & \text{PBN} & \frac{E_h}{J_O} \geq 3; \\ \frac{e^{-\frac{\lambda}{\lambda+1}\frac{E_h}{J_O}}}{1-\lambda^2} & \text{PBN} & \frac{E_h}{J_O} < 3; \\ \frac{1}{E_h/J_O} & \text{MT,} \quad K=1, & \frac{E_h}{J_O} \geq 2; \\ \frac{1}{2} & \text{MT,} \quad K=1, & \frac{E_h}{J_O} < 2; \\ \frac{\beta K}{E_h/J_O} & \text{MT,} \quad K \geq 2, & \frac{E_h}{J_O} \geq \alpha_0 M; \\ \frac{1}{E_h/J_O}\left[\frac{\alpha_{wc}(M-2)}{1-\alpha_{wc}}\right]^{1-\alpha_{wc}} & \text{MT,} \quad K \geq 2, & \frac{E_h}{J_O} < \alpha_0 M \end{cases} \tag{3.4}$$

with λ equal to

$$\lambda = \left[\frac{1}{2}\sqrt{1+3\frac{E_h}{J_O}+\frac{1}{4}\left(\frac{E_h}{J_O}\right)^2} - \frac{1}{2}\frac{E_h}{J_O} - 1\right] \tag{3.5}$$

Note that D should be raised to the power L to get D^L. β is given in Table 2.1. The worst case α, α_{wc}, is now

$$\alpha_{wc} = \begin{cases} \alpha_0 & E_h/J_O \geq \alpha_0 M; \\ \frac{E_h/J_O}{M} & E_h/J_O < \alpha_0 M. \end{cases} \tag{3.6}$$

with α_0 given in Table 2.1. The function $G(D^L)$ in (3.3) varies for different codes, as defined in Chapter 2. The worst case ρ, denoted as ρ_{wc}, is given by

$$\rho_{wc} = \begin{cases} \frac{3}{E_h/J_O} & E_h/J_O \geq 3; \\ 1 & E_b/J_O < 3. \end{cases} \tag{3.7}$$

The formulas given in Chapter 1 and 2 for computing the probability of bit error for block codes are still valid with a fixed hop rate, but in all cases (3.4) to (3.7) must now be used.

Before evaluating the performance of a fixed hop rate system with convolutional and block codes, we first evaluate uncoded systems, i.e., those without EC codes. The reason for this is twofold, first these uncoded results provide an insight into the performance of many coded systems, second the evaluation of some coded systems can be easily related to the evaluation of an uncoded system (as will be shown later).

For uncoded systems over M-ary FSK, we have $R' = K$, and

$$G(D) = \frac{M}{2} D^L. \tag{3.8}$$

From (3.1), (3.3), (3.4) and (3.8), for $E_h/J_O \geq 3$, we can express R_b/R_h for PBN jamming as

$$\frac{R_b}{R_h} = \frac{1}{\ln 2 + (\ln(1/P_b) - \ln 4)/K} \ln \frac{E_h/J_O}{4e^{-1}} \tag{3.9}$$

This means that under PBN jamming, an increase in K(or M) improves the performance, or throughput, (recall that we are concerned with R_b/R_h versus E_h/J_O for a given BER,

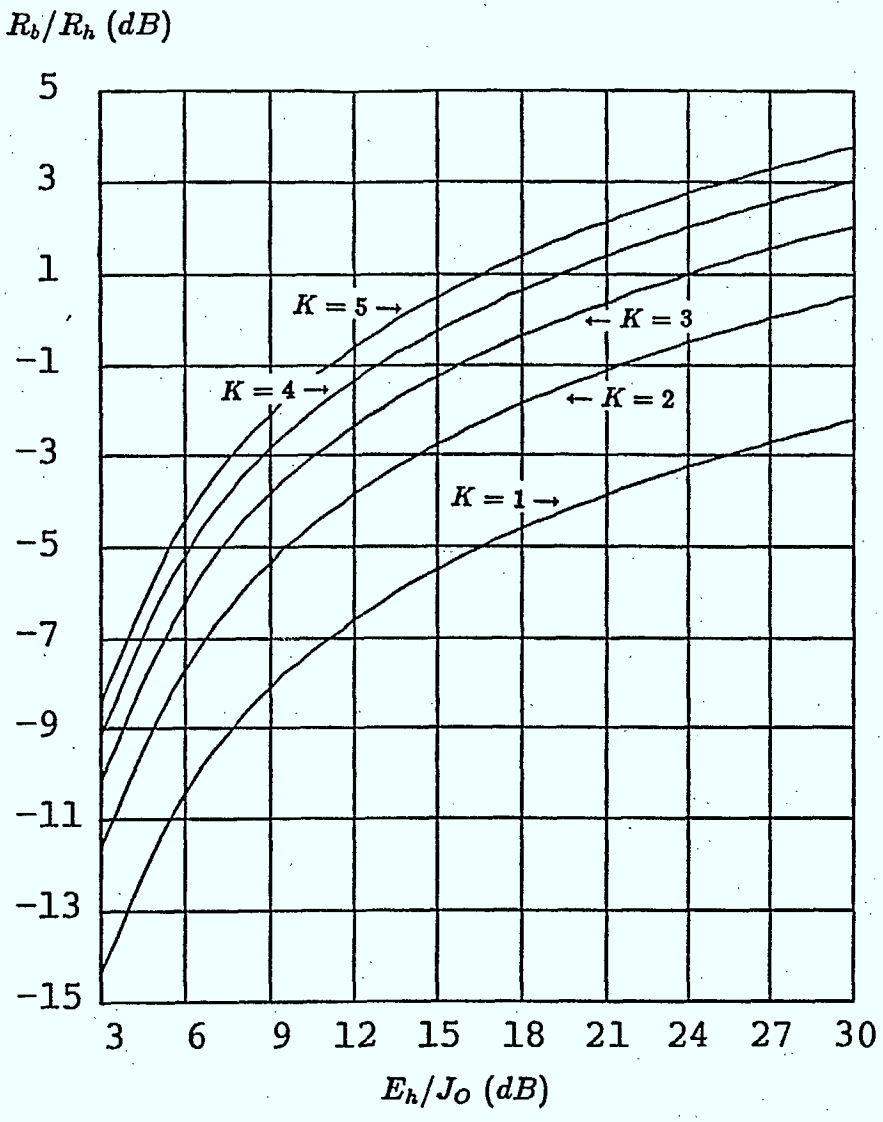


Figure 3.1: Throughput performance of uncoded MFSK with fixed hop rates under WC PBN jamming, for $P_b = 10^{-5}$. $M = 2^K$, $K = 1$ to 5.

P_b). Also, R_b/R_h has a logarithmic dependence on E_h/J_O. The uncoded performance for different K at $P_b = 10^{-5}$, under PBN jamming is shown in Fig. 3.1. To interpret the performance from the figures under any jamming, a larger R_b/R_h for a given E_h/J_O means a larger throughput, which results in better performance. Figs. 3.2 to 3.6 compare the uncoded performance under WC PBN and WC MT jamming for $K = 1$ to 5. From these figures we see that for $K = 1$, regardless of E_h/J_O, PBN jamming is more effective against communications than MT jamming, and the opposite is true for any other K. Thus, for $K = 1$ PBN jamming is the worst case jammer, but for $K = 2$ to 5, MT jamming is the worst. Results similar to these were observed in Chapter 2 for systems with fixed data rates. Fig. 3.7 combines the worst case jamming for each K, i.e., PBN jamming for $K = 1$ and MT jamming for $K = 2$ to 5. From this we can determine the best K for a particular E_h/J_O. Note that $K = 1$ never gives the best R_b/R_h.

In the case of WC MT jamming, the relation between R_b/R_h and E_h/J_O for different K is not as explicit as for PBN jamming. From Figs. 3.2 to 3.7, we see that under WC MT jamming there is an optimum K which is a function of E_h/J_O. For 4.8 dB $< E_h/J_O <$ 8.4 dB, $K = 2$ is optimum, for 8.4 dB $< E_h/J_O <$ 13.1 dB, $K = 3$ is optimum, for 13.1 dB $< E_h/J_O <$ 18.0 dB, $K = 4$ is optimum, and for $E_h/J_O >$ 18.0 dB, $K = 5$ is optimum. The optimum K increases as E_h/J_O increases. This result differs from that for systems with a fixed data rate, for which an increase in K above 2 always gives a poorer performance under MT jamming.

3.3 Performance of Convolutional Codes

As stated previously, the expressions for $G(D)$ for various codes are the same as for a fixed data rate, as given in Chapter 2.

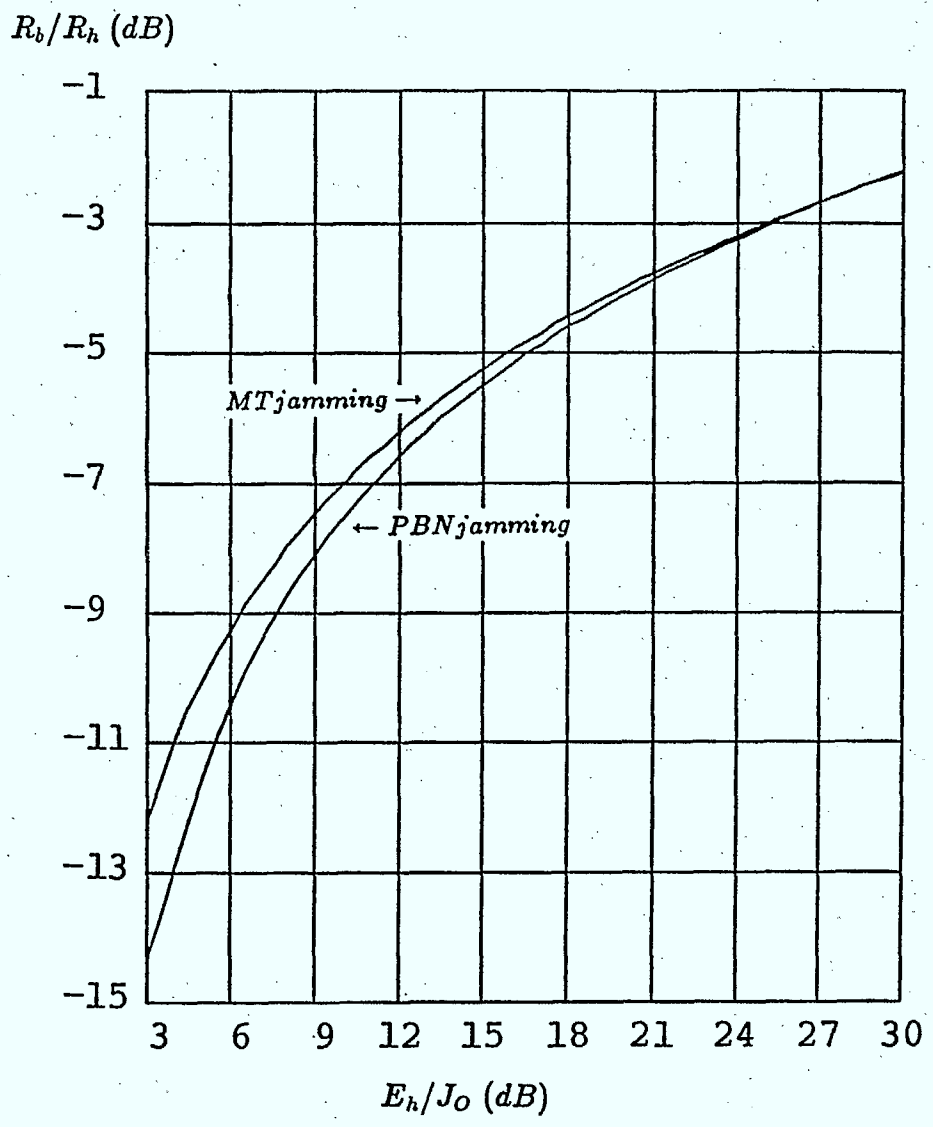


Figure 3.2: Throughput performance of uncoded BFSK with fixed hop rates under WC MT and WC PBN jamming, for $P_b = 10^{-5}$, $K = 1$.

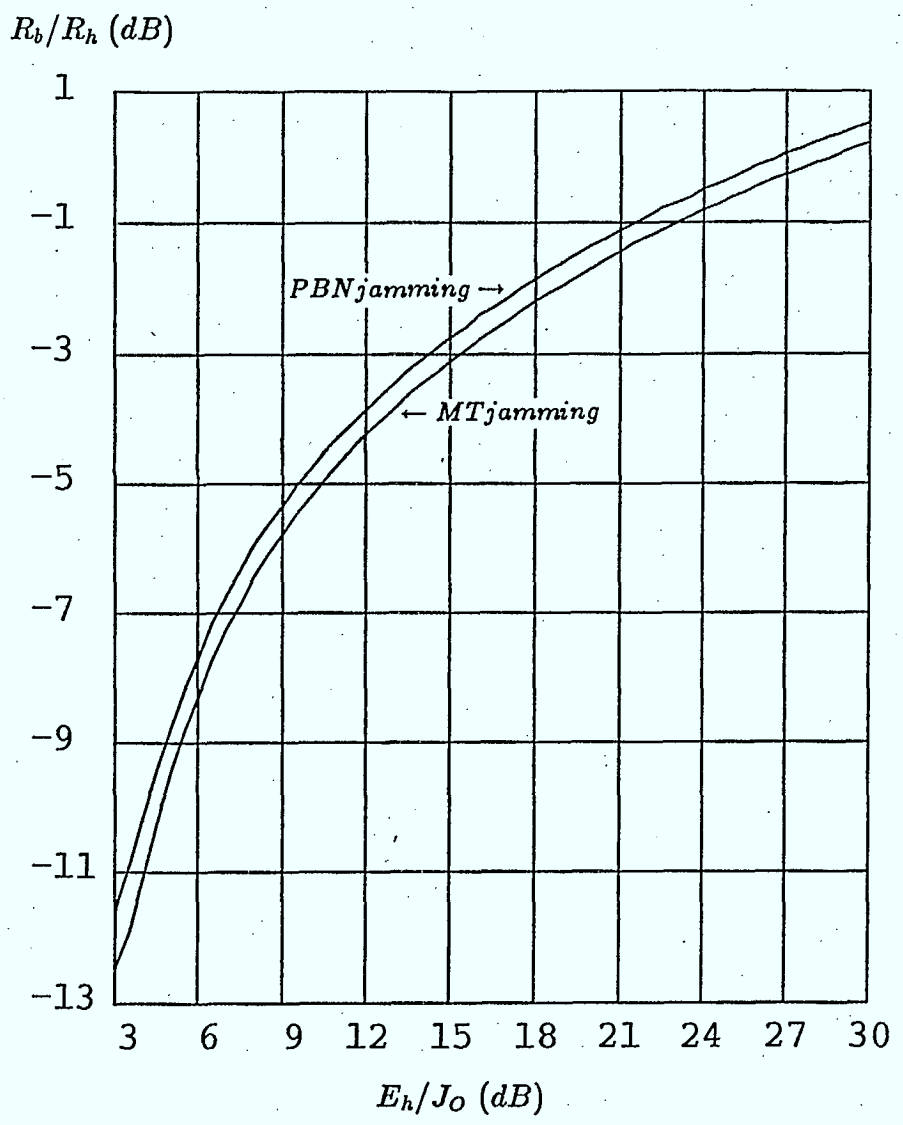


Figure 3.3: Throughput performance of uncoded 4FSK with fixed hop rates under WC MT and WC PBN jamming, for $P_b = 10^{-5}$, $K = 2$.

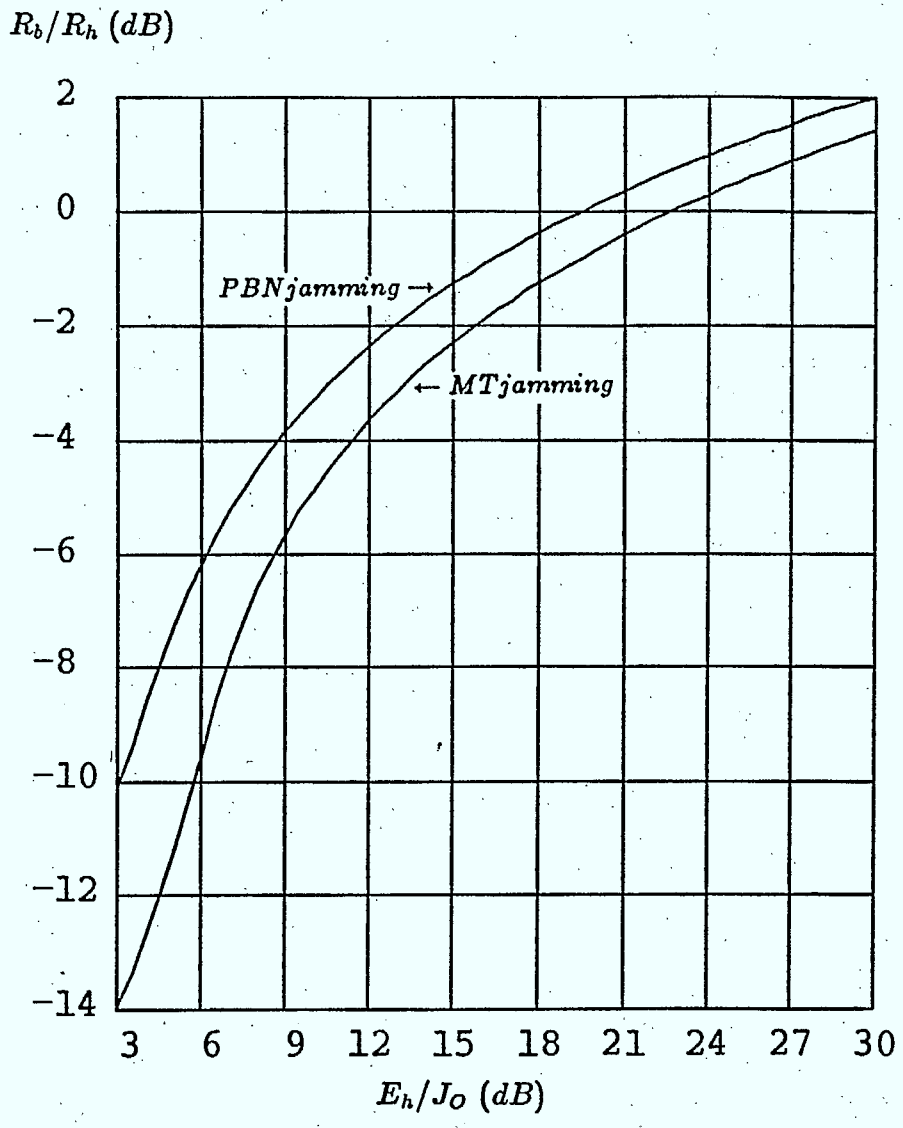


Figure 3.4: Throughput performance of uncoded 8FSK with fixed hop rates under WC MT and WC PBN jamming, for $P_b = 10^{-5}$, $K = 3$.

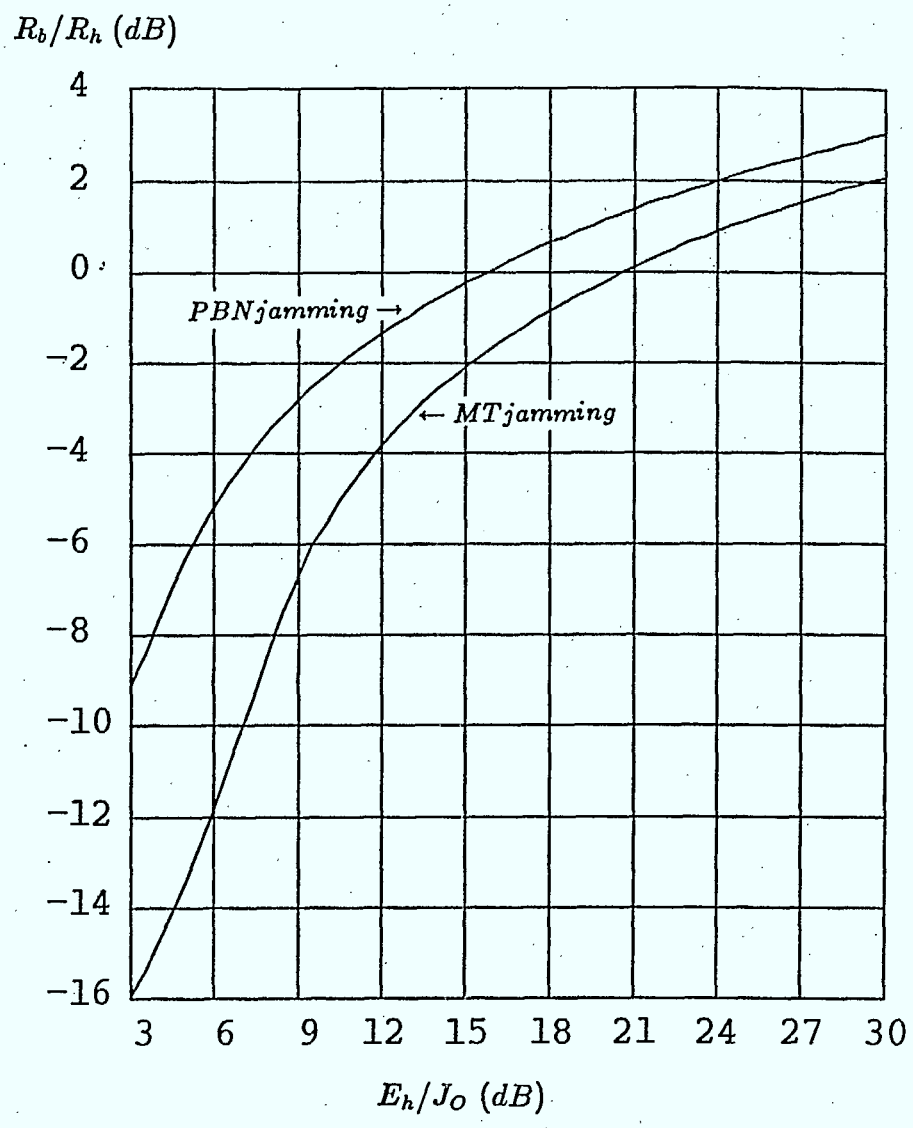


Figure 3.5: Throughput performance of uncoded 16FSK with fixed hop rates under WC MT and WC PBN jamming, for $P_b = 10^{-5}$, $K = 4$.

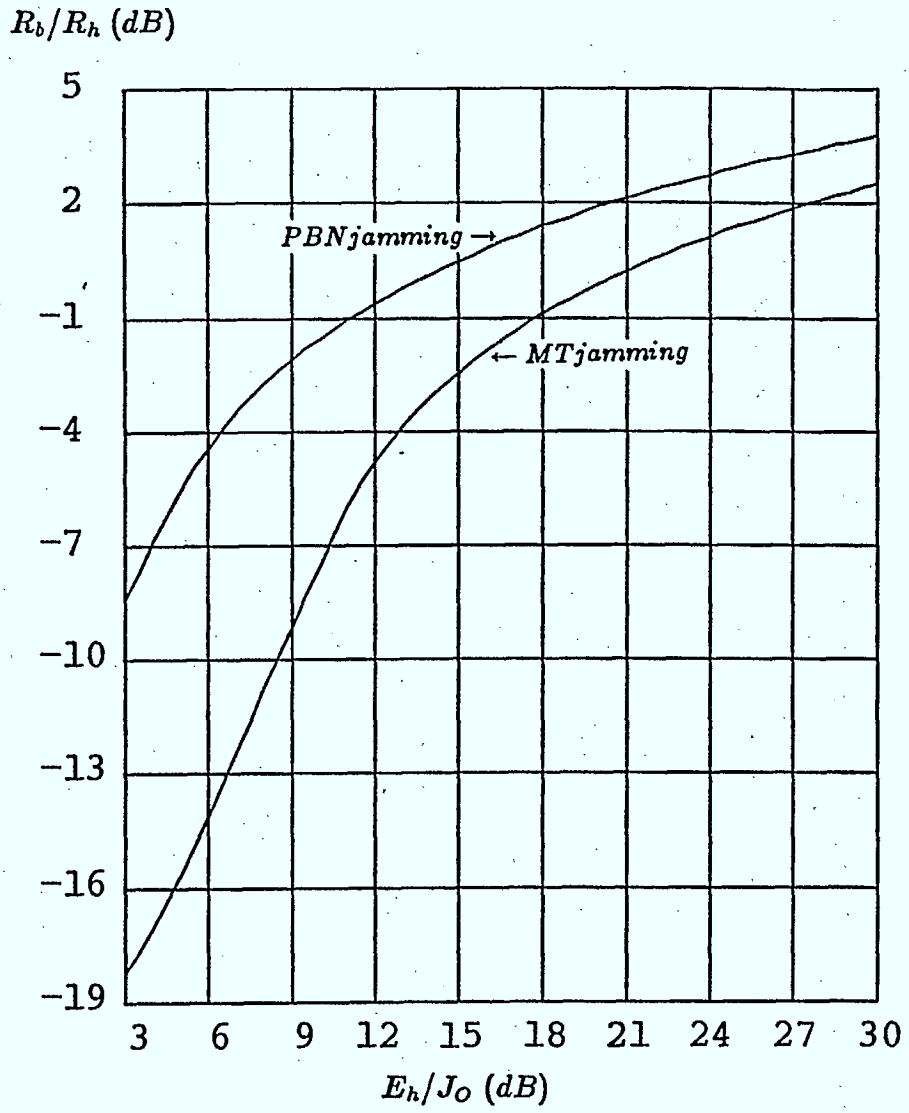


Figure 3.6: Throughput performance of uncoded 32FSK with fixed hop rates under WC MT and WC PBN jamming, for $P_b = 10^{-5}$, $K = 5$.

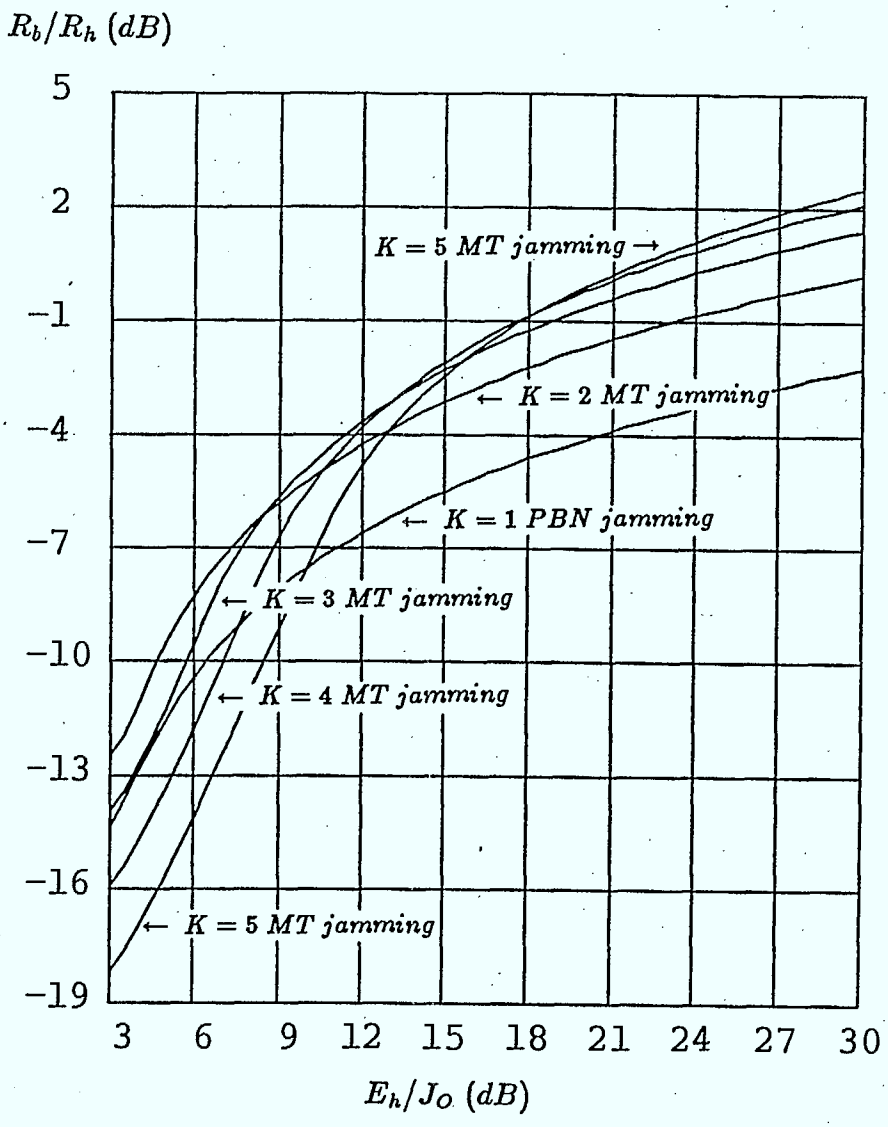


Figure 3.7: Throughput performance of uncoded MFSK with fixed hop rates under the worst jamming, WC MT jamming for $K = 2$ to 5, and WC PBN jamming for $K = 1$, for $P_b = 10^{-5}$.

3.3.1 Odenwalder Binary Codes

If D in (2.17) is replaced with D^2 and D^4, we get (2.19) and (2.20), respectively. Thus from (3.1) it is evident that the rate 1/2, 1/4 and 1/8 codes have the same performance, except that the region of valid R_b/R_h is different, due to different upper limits on R_b/R_h set by R'. Since R' is smaller for lower rate codes, the rate 1/2 code provides the largest upper limit for R_b/R_h. Thus we only consider the rate 1/2 and 1/3 codes.

Figs. 3.8 and 3.9 give the performance under PBN and MT jamming for these codes, respectively. It is evident from these figures that for $K = 1$, PBN jamming is more effective against communications than MT jamming. Thus we need only consider the results under PBN jamming to determine the best code. This result is the same as for fixed data rates. We note that for the rate 1/3 code, the number of terms in (2.18) is insufficient for numerical accuracy in Fig. 3.9. To compare the rate 1/2 and 1/3 codes with numerical accuracy, the throughput performance for $P_b = 10^{-8}$ and 10^{-12} was calculated. This was done because the number of terms in (2.18) is small, thus D has to be small in order to assure accuracy, which implies P_b must be very low. These results confirm that the rate 1/3 code performs slightly better than the rate 1/2 code over the region of valid R_b/R_h. However, this region is 1.76 dB smaller than that for the rate 1/2 code. The rate 1/3 code also has the best performance with a fixed data rate. It is interesting to note that although the redundancy allowed for fixed hop rates is different from that for fixed data rates, the coding scheme best utilizing the redundancy is the same.

3.3.2 Trumpis Codes

Figs. 3.10 and 3.11 give the performance under PBN jamming and MT jamming for these codes over 4FSK and 8FSK, respectively. They show that MT jamming is more effective than PBN jamming for these codes, as was observed for uncoded systems. Also, the rate 1/2 code over 4FSK performs better, which is the same as for systems with a fixed

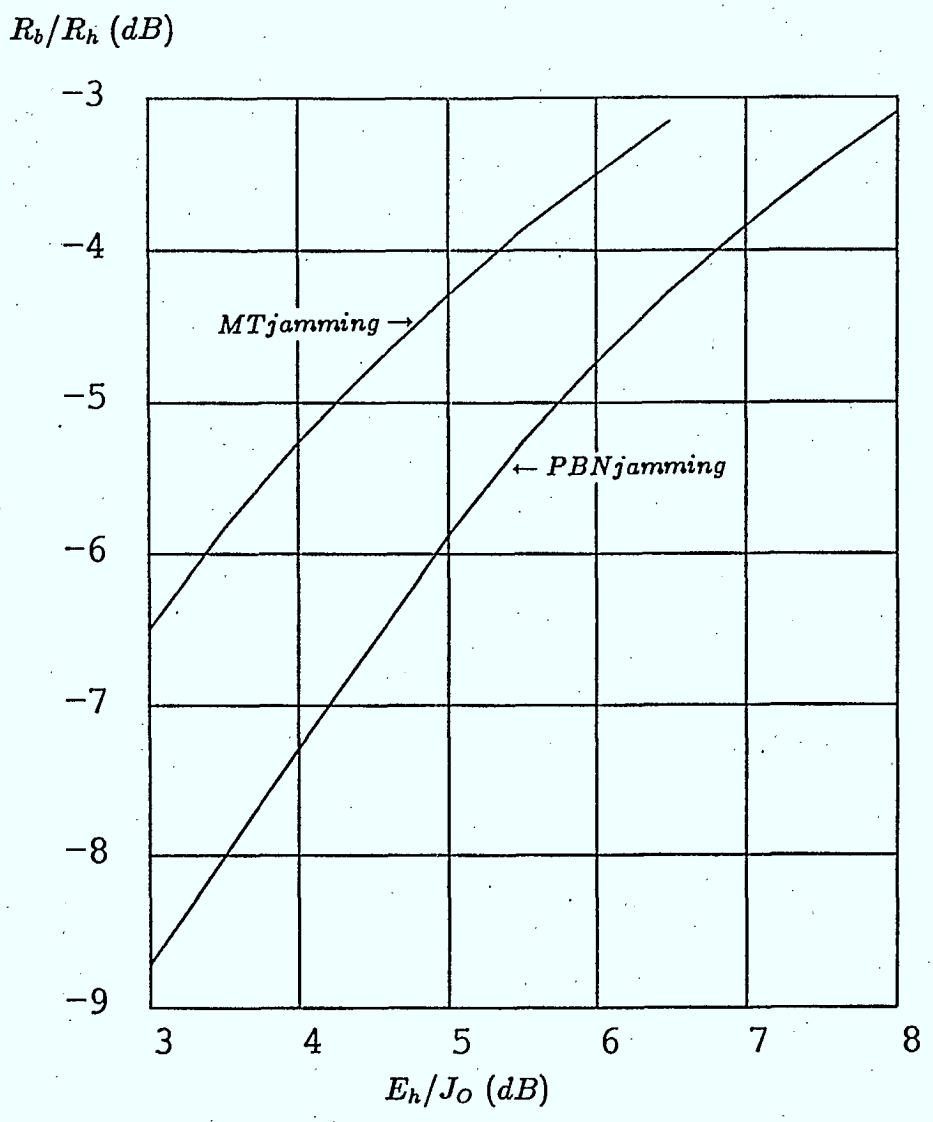


Figure 3.8: Throughput performance of the binary rate 1/2 Odenwalder code with BFSK, $(K = 1)$, and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

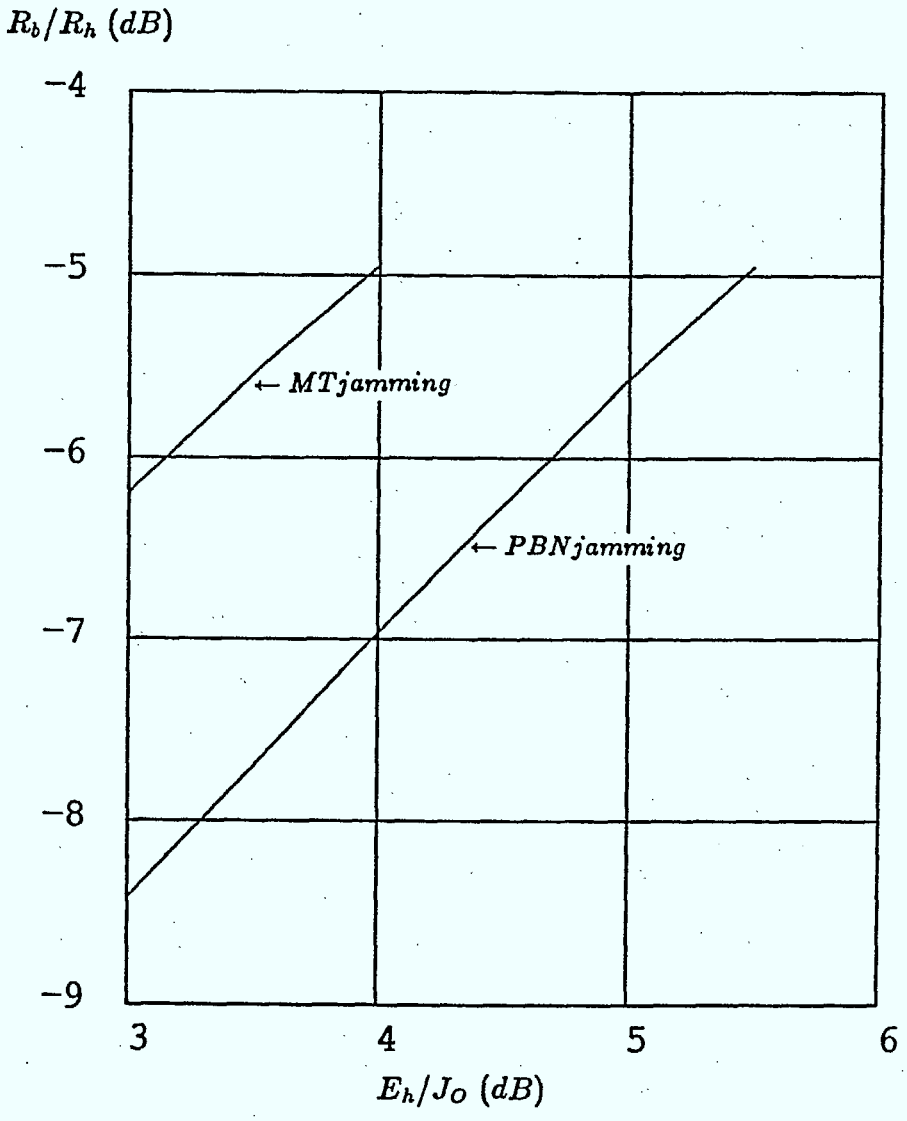


Figure 3.9: Throughput performance of the binary rate 1/3 Odenwalder code with BFSK, $(K = 1)$, and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

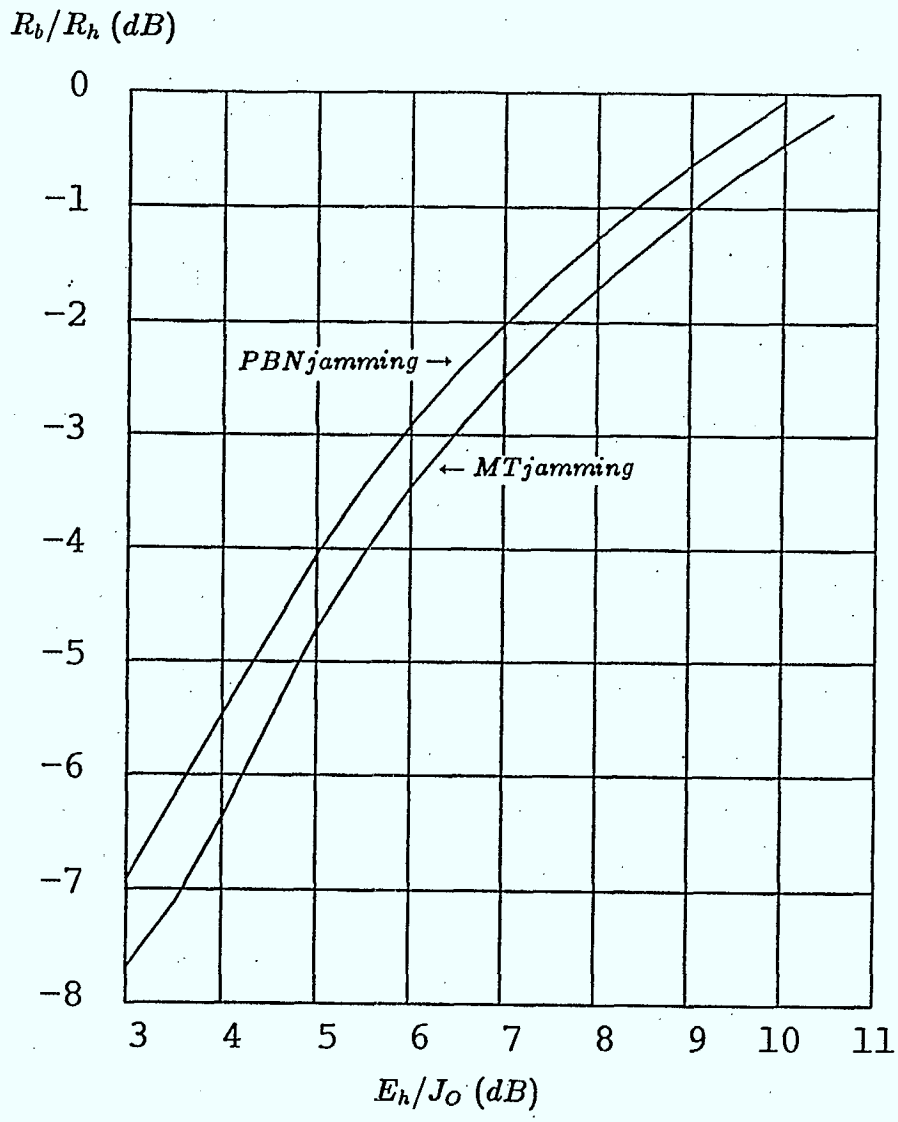


Figure 3.10: Throughput performance of the rate 1/2 Trumpis 4-ary convolutional code with 4FSK and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

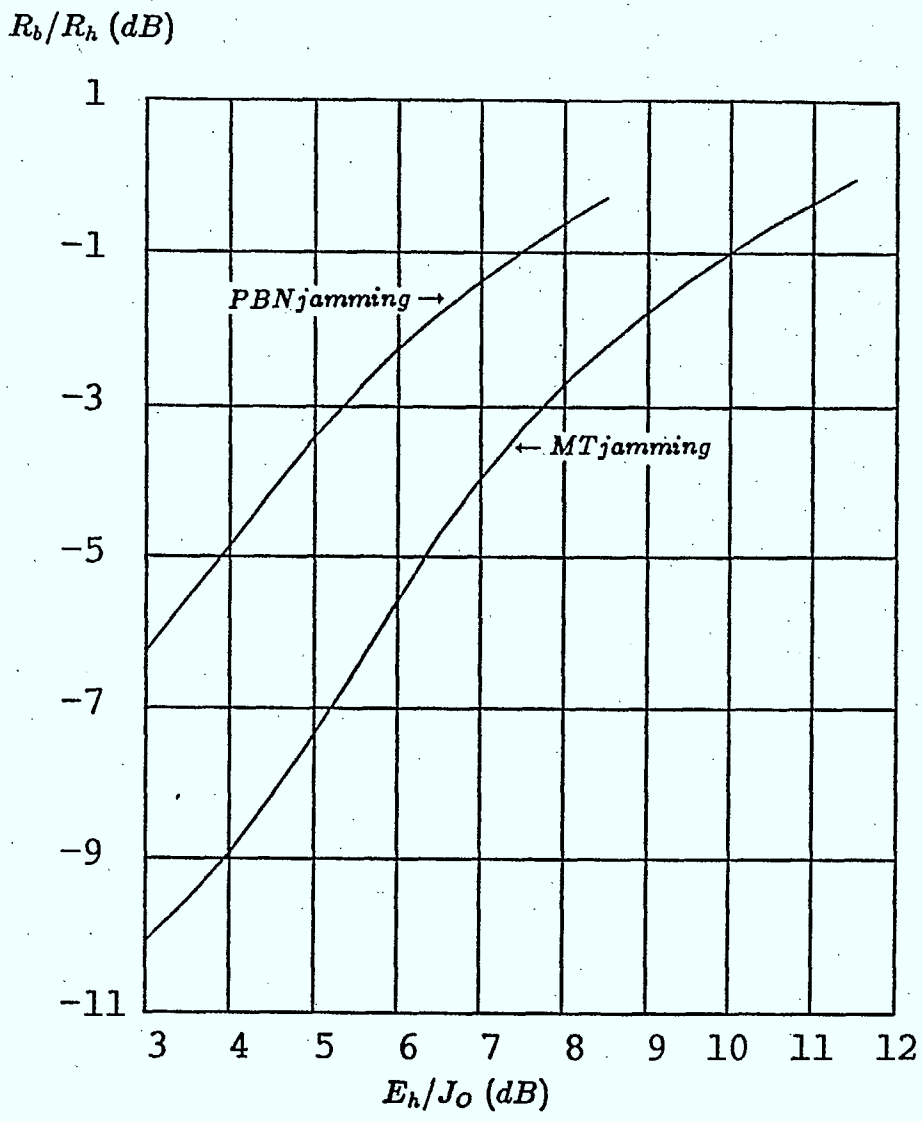


Figure 3.11: Throughput performance of the rate 1/3 Trumpis 8-ary convolutional code with 8FSK and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

data rate and Trumpis coding.

3.3.3 M-ary Orthogonal Convolutional Codes

Figs. 3.12 to 3.15 give the performance under WC PBN jamming and WC MT jamming for $K = 2$ to 5, respectively. From these we observe that MT jamming is more effective than PBN jamming. Thus we need only consider performance under MT jamming. The combined performance curves under MT jamming for $K = 2$ to 5 are given in Fig. 3.16. This figure is similar to that observed for uncoded systems, where there is an optimum K (and thus an optimum code), which is dependant on E_h/J_O. Over the range of E_h/J_O the optimum code varies. This result is different from that for systems with a fixed hop rate and M-ary orthogonal convolutional coding, in which $K = 2$ is always the best signalling.

3.3.4 Dual-K Convolutional Codes

$G(D)$ for these codes is given by (2.16). For small D, we have the approximation

$$G(D^L) \approx \frac{MD^{2L\nu}}{2} = \frac{MD^{2LK/R'}}{2}. \tag{3.10}$$

Noting (3.1), we have

$$G(D^L) \approx \frac{MD^{\frac{2K}{R_b/R_h}}}{2}. \tag{3.11}$$

This means that the performance of the system is independent of ν (the code rate) asymptotically as E_h/J_O increases. In fact, this approximation is valid for most E_h/J_O of interest. However, a smaller ν will give a larger R', thus the range of possible R_b/R_h is larger. Also, comparing (3.11) with (3.8), we find a dual-K coded system has exactly 3 dB gain in throughput, R_b/R_h, over an uncoded system (but with inherent diversity), for the same K and E_h/J_O. Thus all results and discussions presented earlier for uncoded systems apply to dual-K coded systems, except for the 3 dB difference in R_b/R_h and the valid range of R_b/R_h. This is verified by Fig. 3.17, which gives the performance under MT jamming for

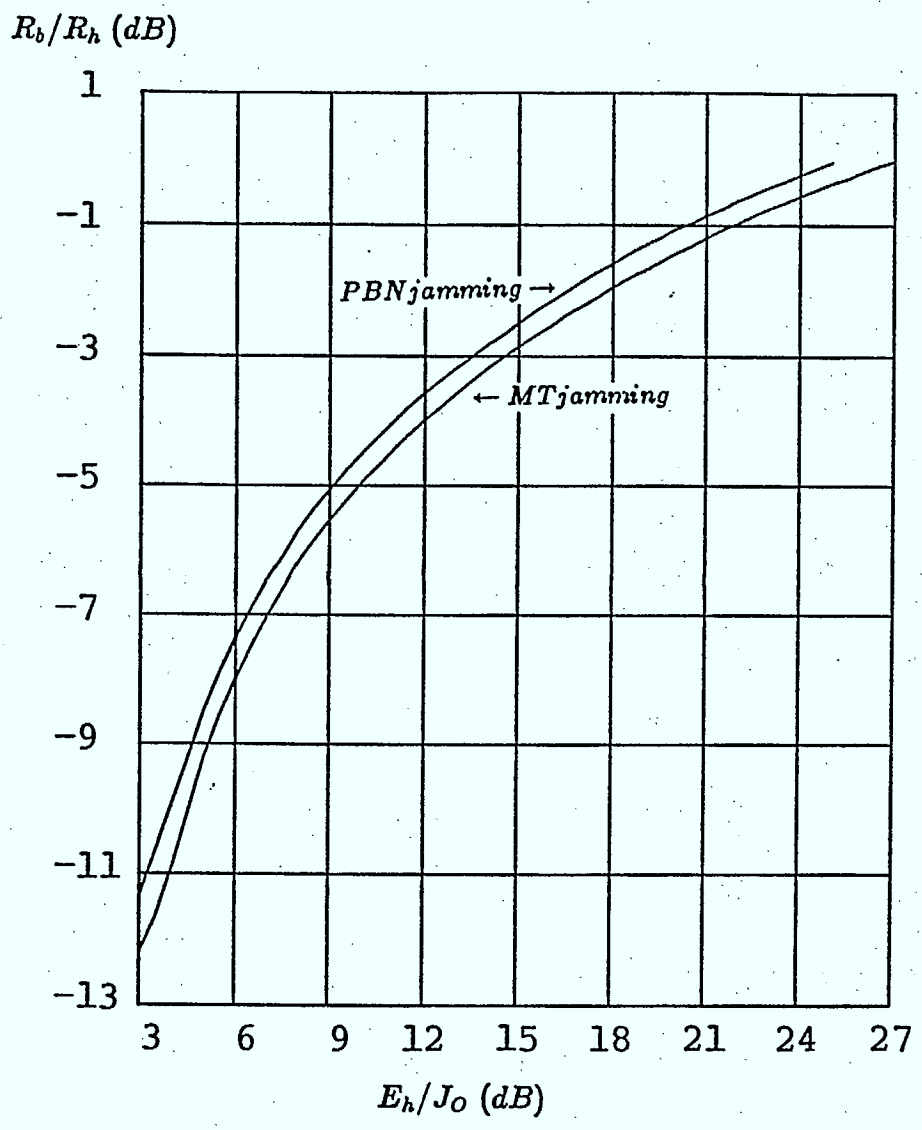


Figure 3.12: Throughput performance of the 4-ary Orthogonal Convolutional code with 4FSK and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

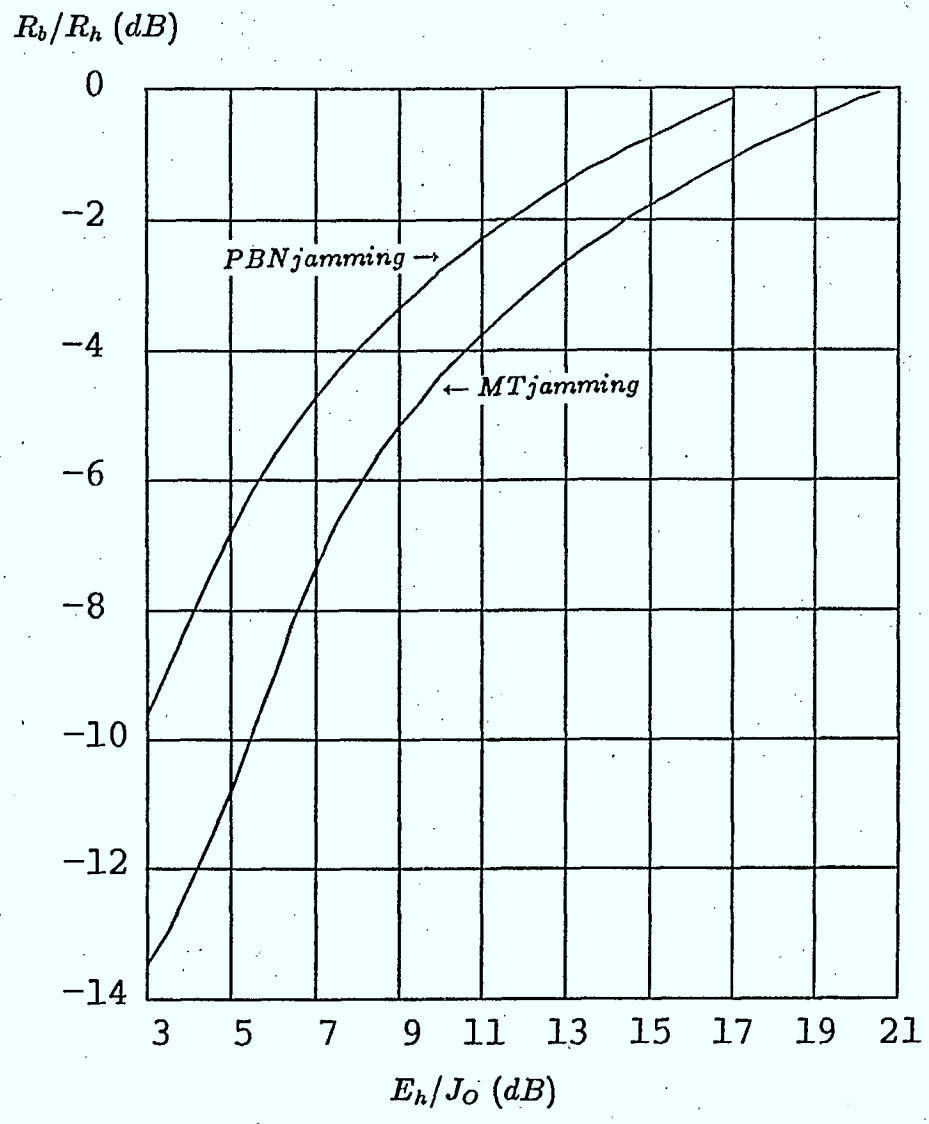


Figure 3.13: Throughput performance of the 8-ary Orthogonal Convolutional code with 8FSK and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

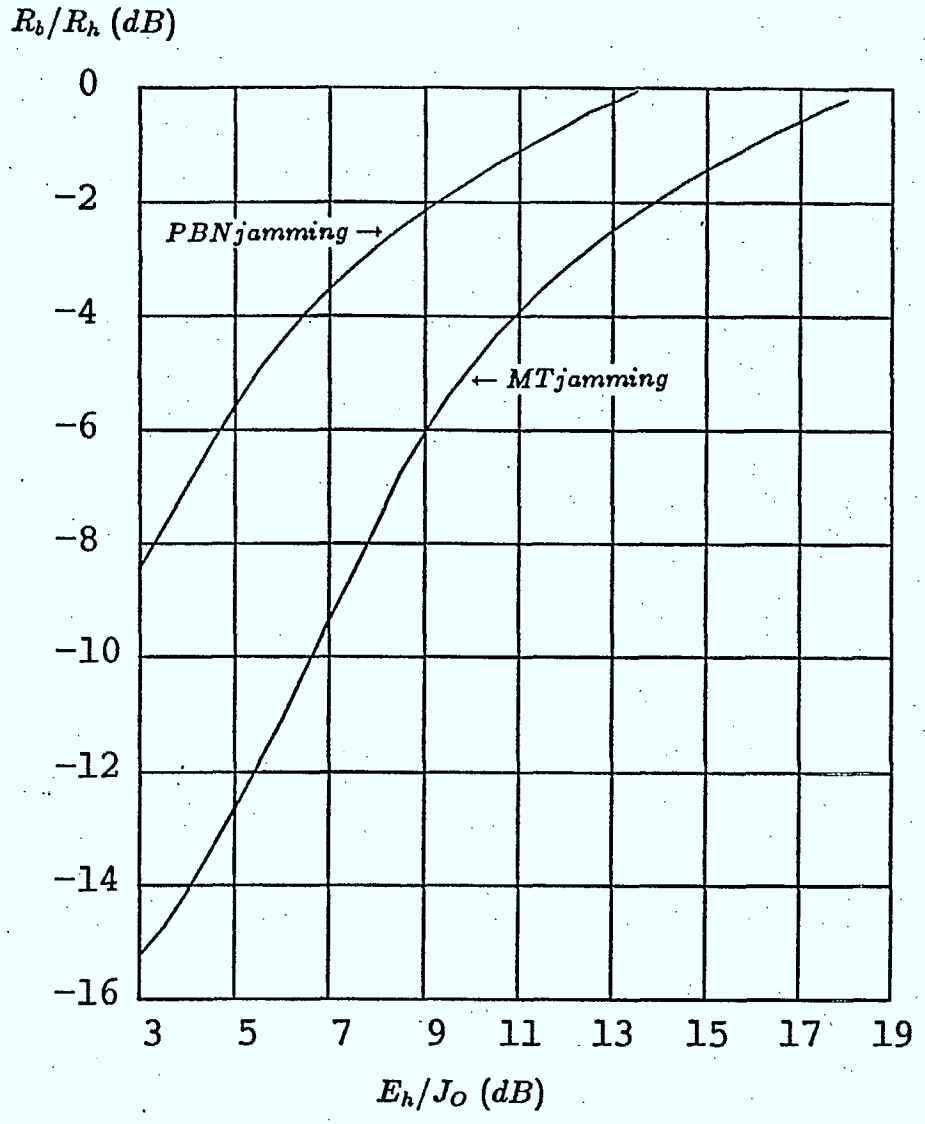


Figure 3.14: Throughput performance of the 16-ary Orthogonal Convolutional code with 16FSK and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

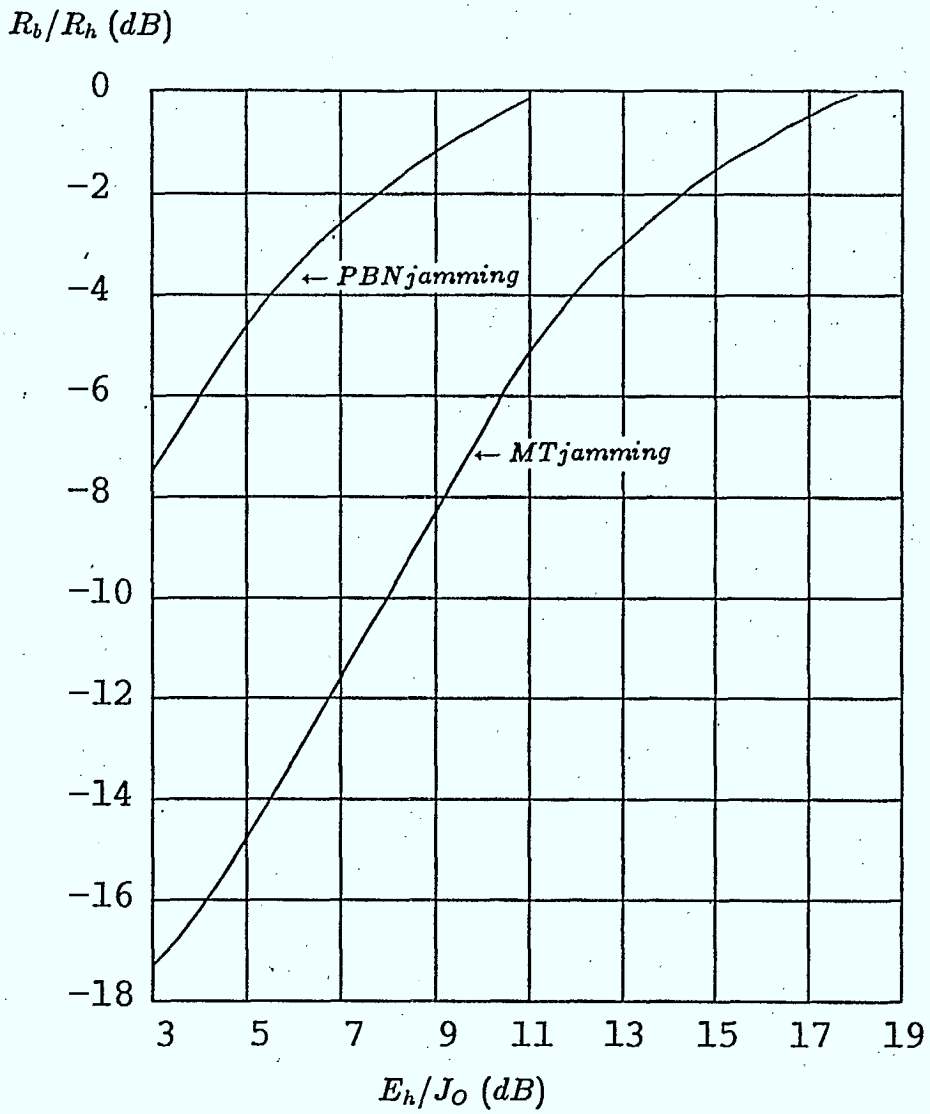


Figure 3.15: Throughput performance of the 32-ary Orthogonal Convolutional code with 32FSK and fixed hop rates under WC PBN and WC MT jamming, for $P_b = 10^{-5}$.

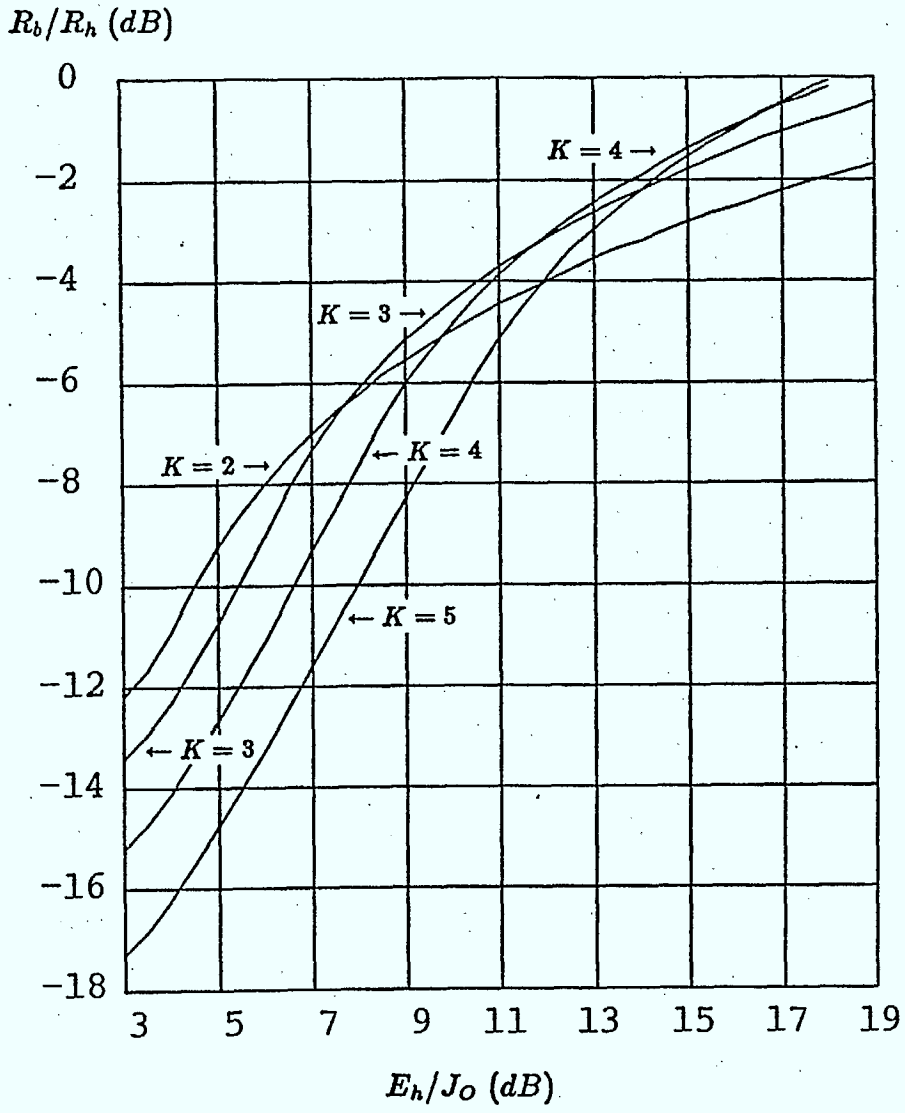


Figure 3.16: Throughput performance of the M-ary Orthogonal Convolutional codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5, for $P_b = 10^{-5}$.

the rate 1/2 dual-K codes, with $K = 2$ to 5.

It should be noted that this asymptotic independence is also true for systems with fixed data rates and optimum diversity, but the proof is more involved. For dual-K coded systems with fixed data rates, $K = 2$ is always the best signalling, but with a fixed hop rate, the optimum K depends on E_h/J_O (this K increases with increasing E_h/J_O).

3.3.5 Semi-Orthogonal Convolutional Codes

Figs. 3.18 to 3.20 give the performance under WC PBN and WC MT jamming for $K = 3$ to 5, respectively. From these figures we see that MT jamming is more effective than PBN jamming for this class of codes. This is the same as observed for uncoded systems, thus we need only consider performance under MT jamming. Fig. 3.21 gives the performance under MT jamming for $K = 3$ to 5, which shows that $K = 3$ produces the best result. This is also the optimum K for systems with a fixed data rate and semi-orthogonal convolutional coding.

3.4 Performance of Block Codes

The analysis for block codes is similar to that for a fixed data rate, except for the changes noted previously.

3.4.1 Binary Codes

As discussed in Chapter 2, (n, k) binary block codes can be used with depth K interleaving, so that an M-ary channel can be employed. Thus there are two parameters which can be varied in order to minimize P_b, K and the EC code to be used, i.e., n and k. Comparing the coded channel we have now with the uncoded systems considered in Section 2, we find two properties of uncoded systems still apply. Regardless of E_h/J_O, for $K = 1$, PBN jamming is more effective against communications than MT jamming, and the opposite is true for any other K. Second, under the worst possible jamming, there is an

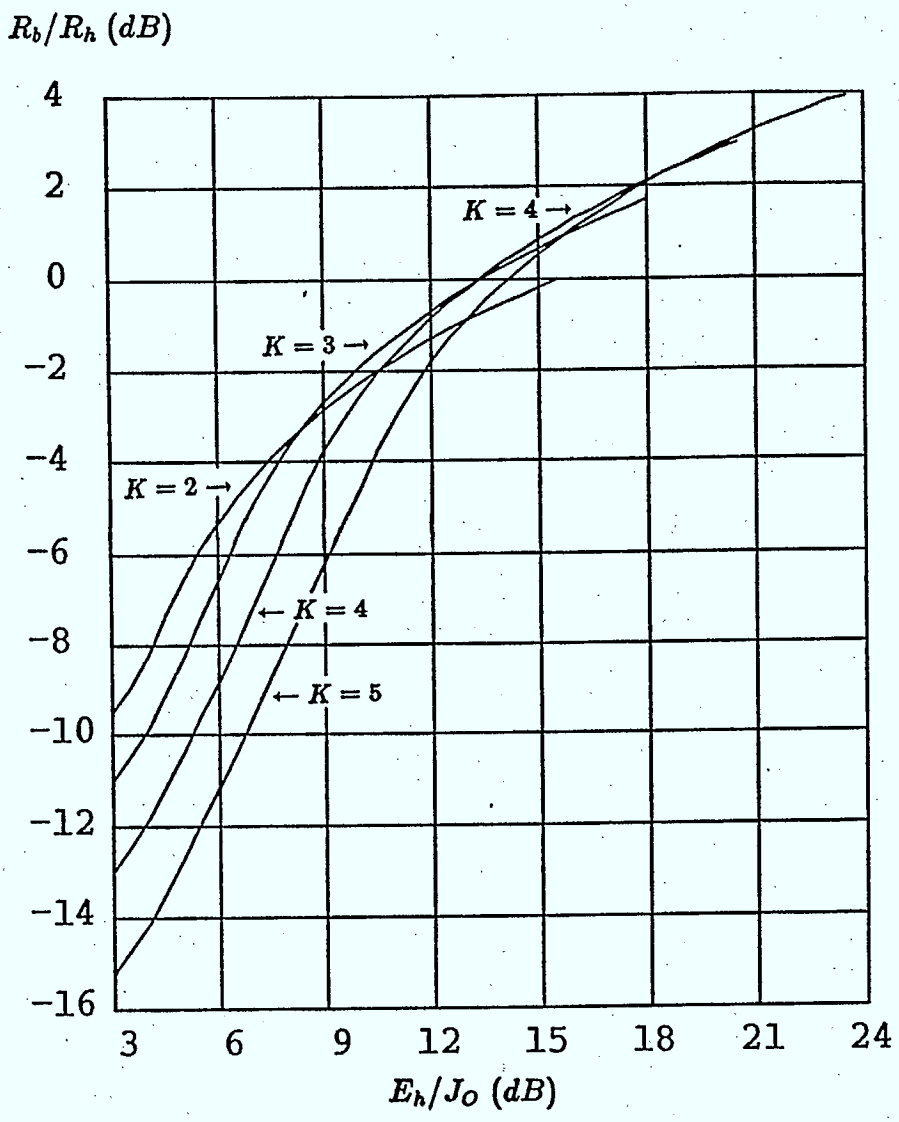


Figure 3.17: Throughput performance of the rate 1/2 Dual-K Orthogonal Convolutional codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5, for $P_b = 10^{-5}$.

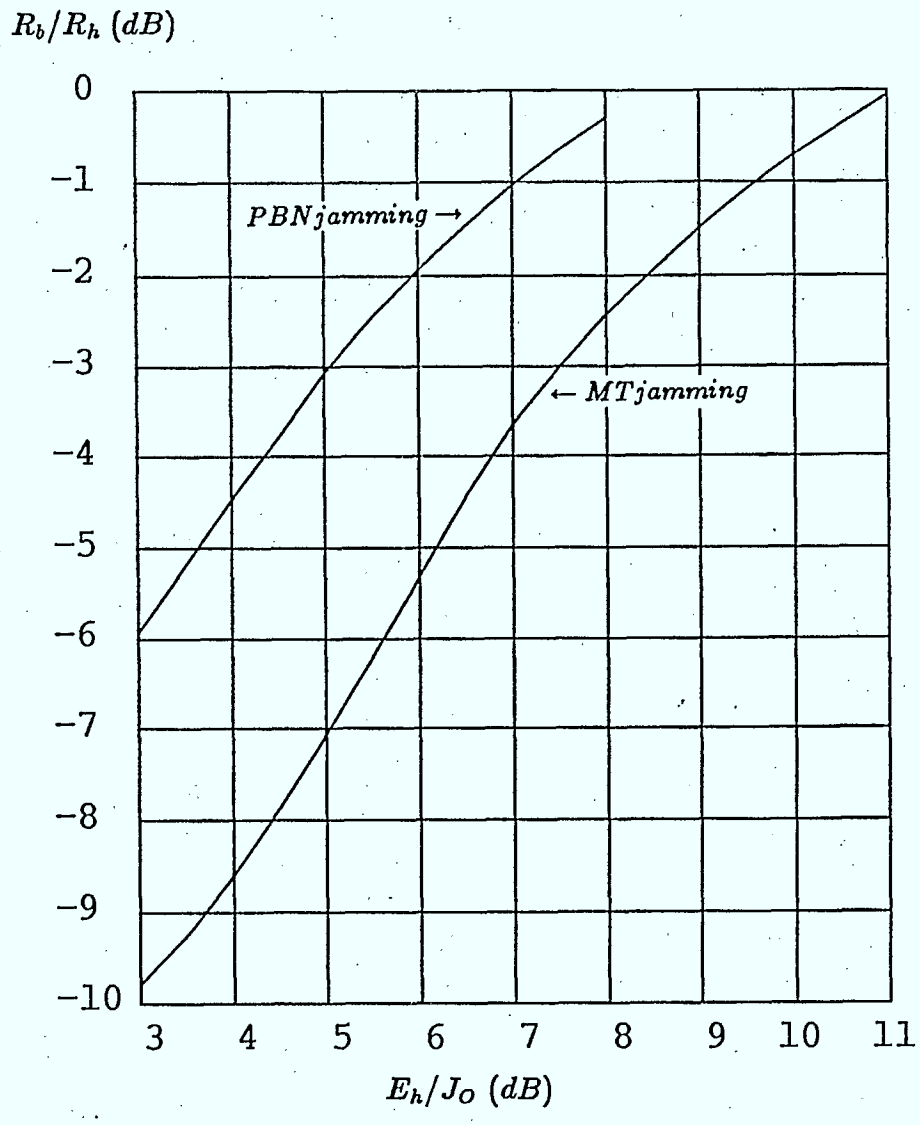


Figure 3.18: Throughput performance of the Semi-Orthogonal Convolutional code under WC PBN and WC MT jamming, $K = 3$, for $P_b = 10^{-5}$.

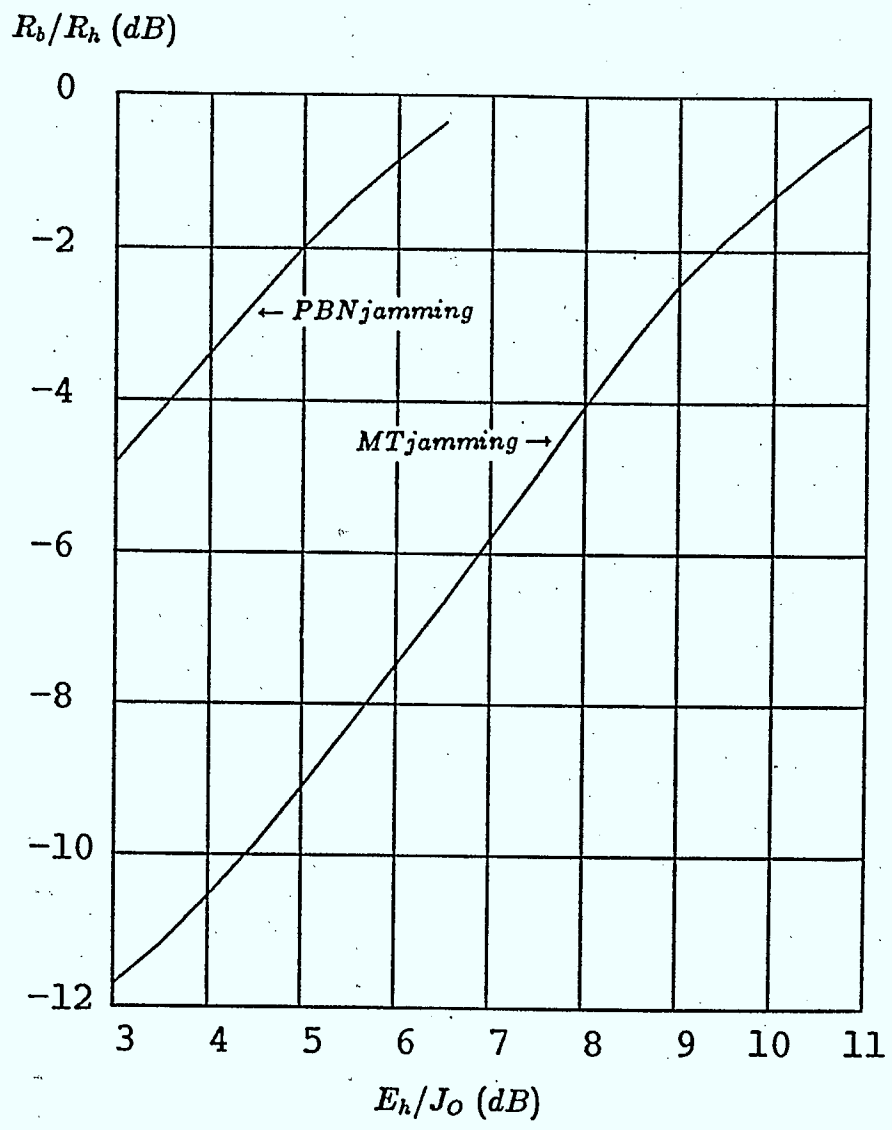


Figure 3.19: Throughput performance of the Semi-Orthogonal Convolutional code under WC PBN and WC MT jamming, $K = 4$, for $P_b = 10^{-5}$.

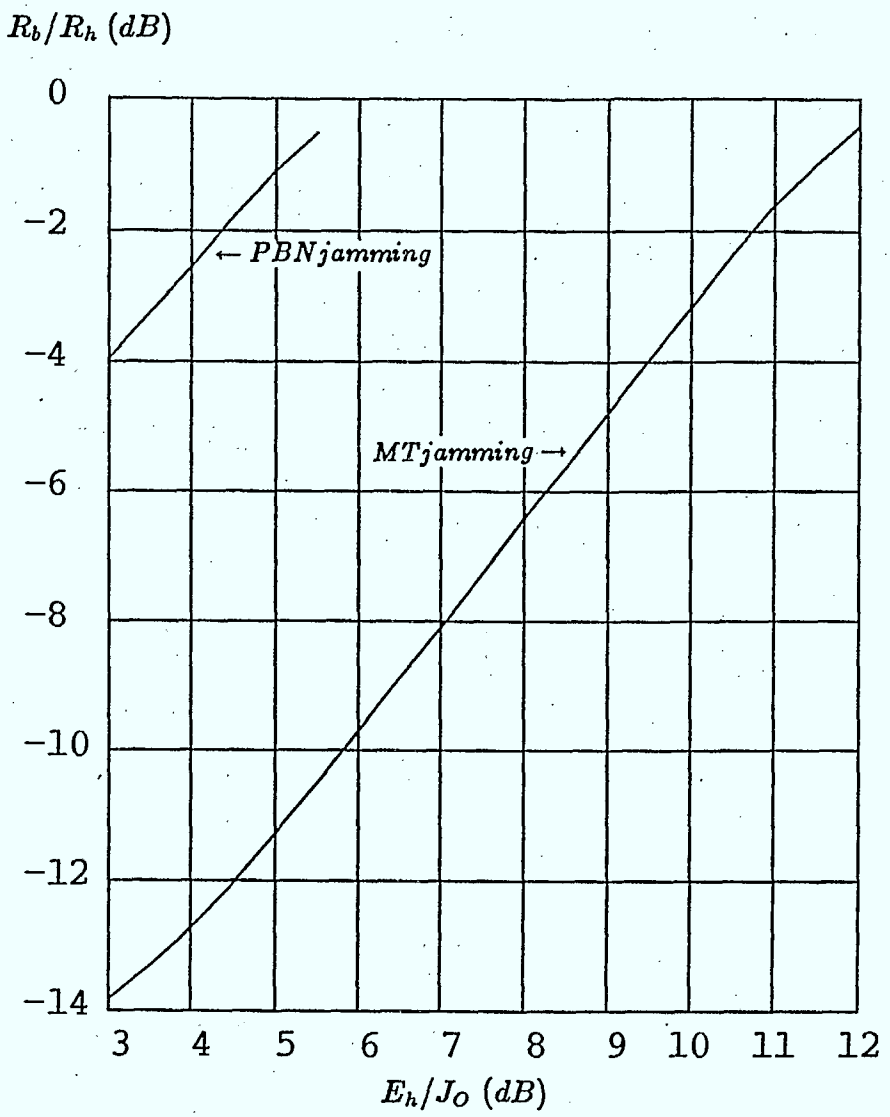


Figure 3.20: Throughput performance of the Semi-Orthogonal Convolutional code under WC PBN and WC MT jamming, $K = 5$, for $P_b = 10^{-5}$.

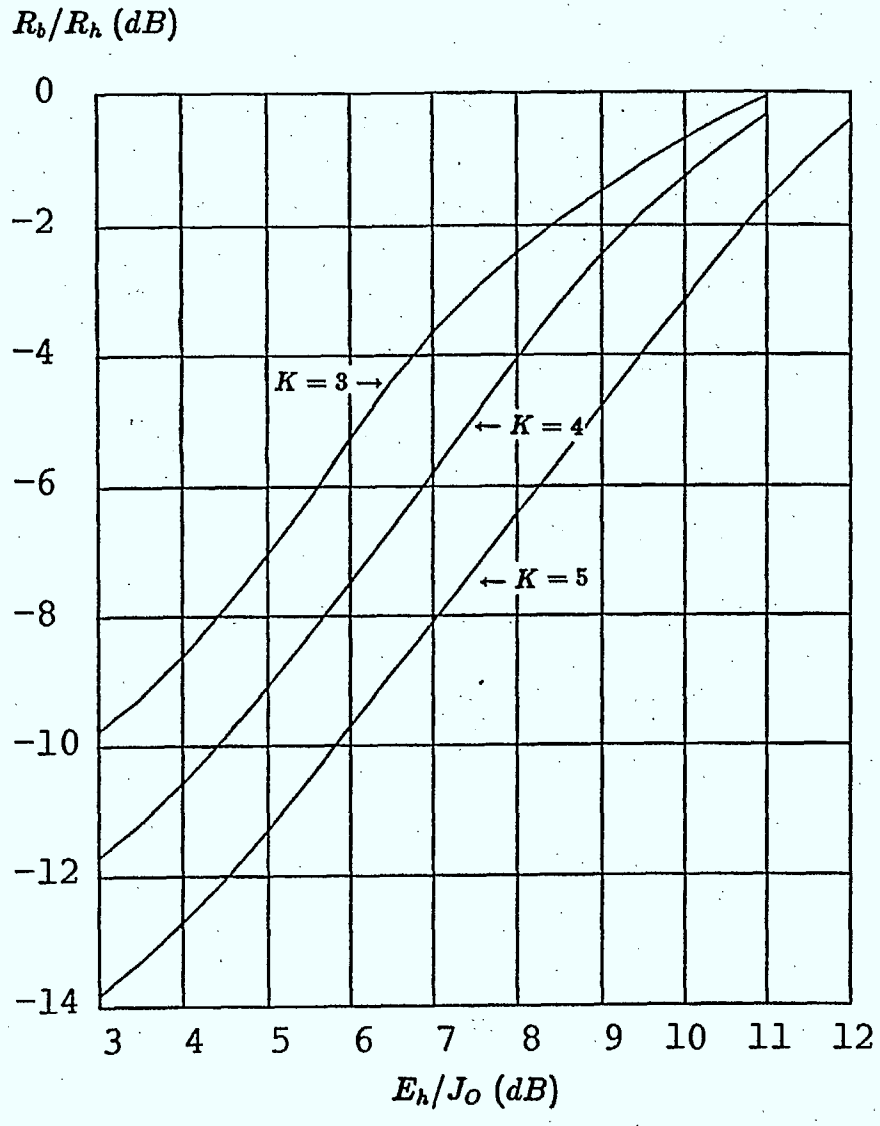


Figure 3.21: Throughput performance of the Semi-Orthogonal Convolutional codes with MFSK and fixed hop rates under WC MT jamming, $K = 3$ to 5, for $P_b = 10^{-5}$.

optimum K which is no less than 2 and is a function of E_h/J_O. This optimum K increases as E_h/J_O increases. Thus all we need do is evaluate codes for $K \geq 2$ under MT jamming. This will give an optimum K which also varies according to the particular code used, and may be different from that for uncoded systems. Thus the results for binary block codes with a fixed hop rate are different from those for systems with a fixed data rate, in which $K = 2$ is always the best signalling.

We examined the (7,4) and (31,26) Hamming codes, the (23,12) Golay code, and the BCH codes of length $n = 15$, 31, 63, 127, and 255, as in Chapter 2.

Summary of Results

In order to simplify the presentation of the results and allow for easy comparison, only the envelope which gives the best R_b/R_h for a given E_h/J_O is shown. Thus the curves representing each code start with $K = 2$ for low E_h/J_O and end with $K = 5$ for high E_h/J_O. The performance of the Hamming codes and the Golay code is shown in Fig. 3.22. It is seen that the (31,26) Hamming code is the best of these short block length codes, as was the case with a fixed data rate. The results for the block length 15 and 31 BCH codes are given in Figs. 3.23 and 3.24, respectively. From these it is obvious that the (15,11) and (31,21) codes are the best of these BCH codes. However, a comparison with Fig. 3.22 shows that the (31,26) code is slightly better than the (31,21) code (0.18 dB in R_b/R_h at E_h/J_O = 30dB). The performance of these two codes is repeated in Fig. 3.25. This result differs from that with a fixed data rate, where the (31,21) code is slightly better than the (31,26). The results for the block length 63, 127 and 255 BCH codes are given in Figs. 3.26, 3.27 and 3.28, respectively. The optimum BCH codes for each block length (in the sense that they provide the largest throughput, R_b/R_h, for a given E_h/J_O at $P_b = 10^{-5}$ and ignoring small crossings in the performance curves), are the (63,51), (127,106) and (255,207) codes. The performance of the 5 best codes (for each block length 15 to 255) are shown in Fig. 3.29. From this figure it is clear that 0.3 dB to 0.6 dB in throughput can be gained at

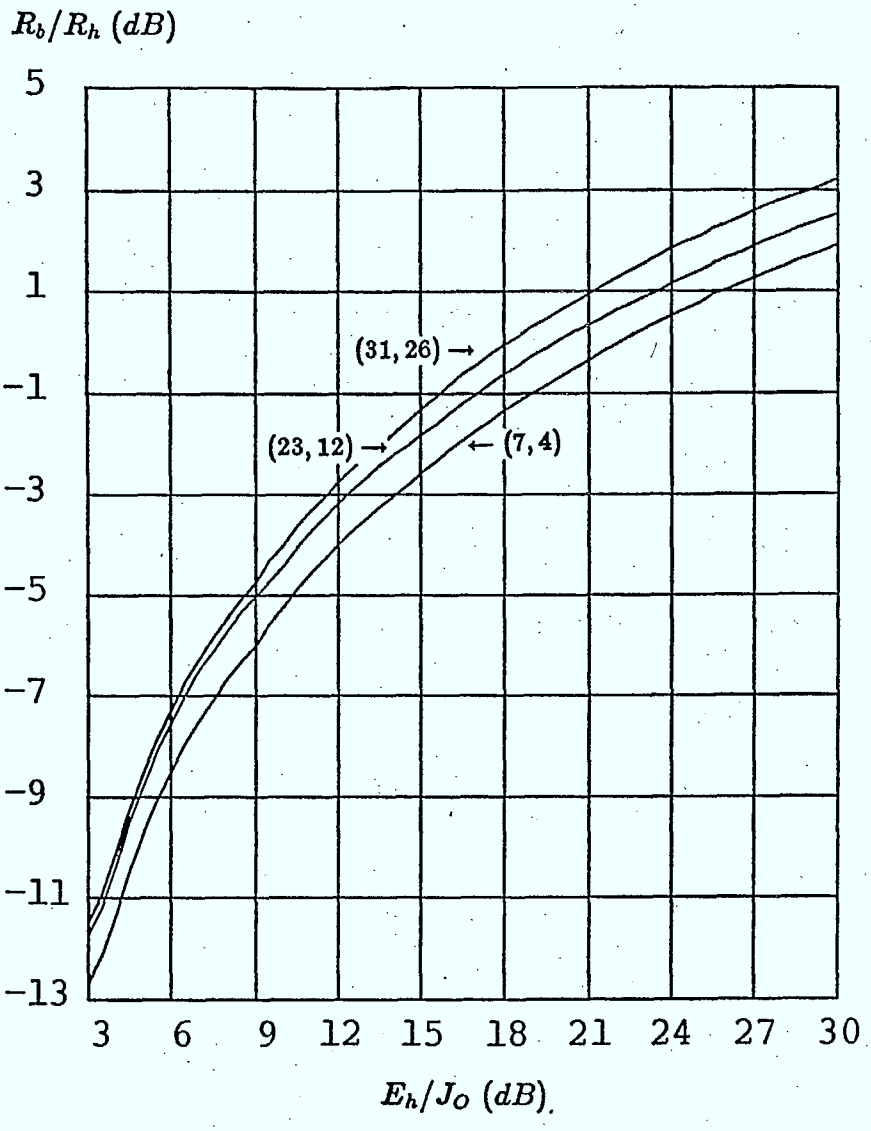


Figure 3.22: Throughput performance of the (23,12) Golay code and (7,4) and (31,26) Hamming codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

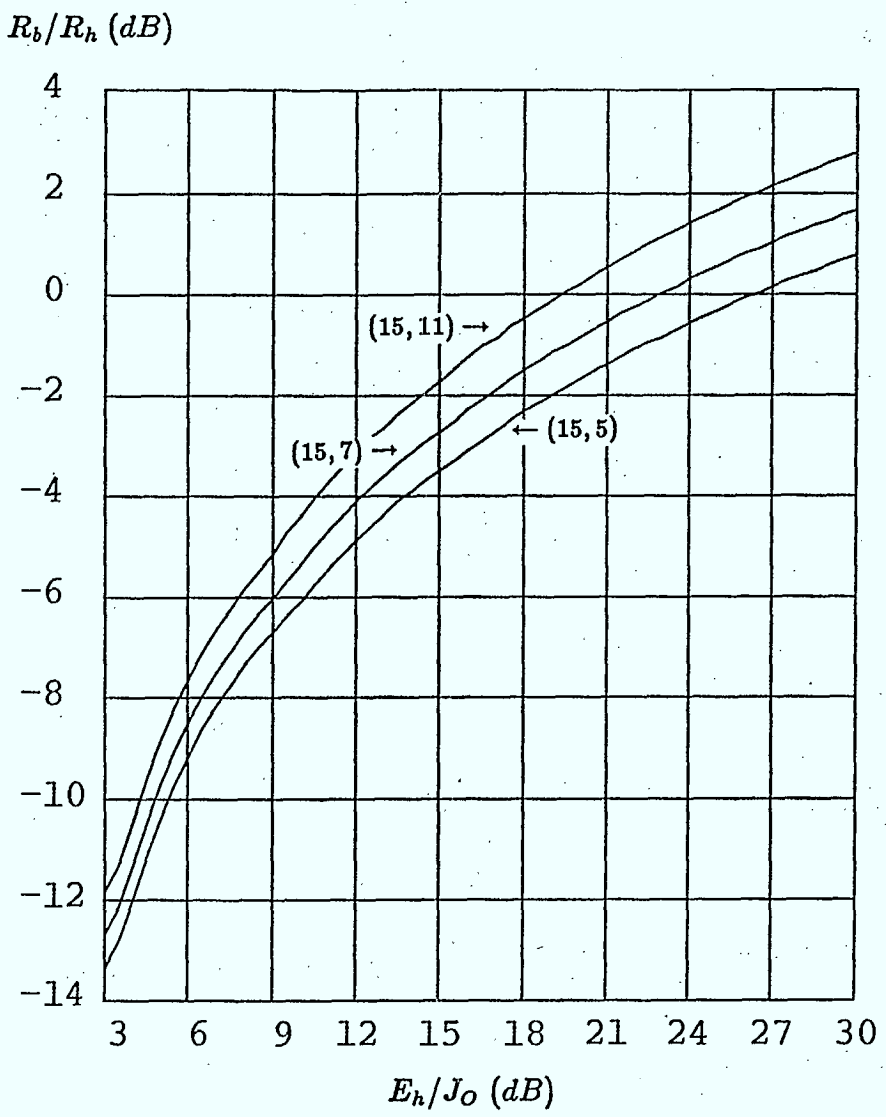


Figure 3.23: Throughput performance of the block length 15 BCH codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

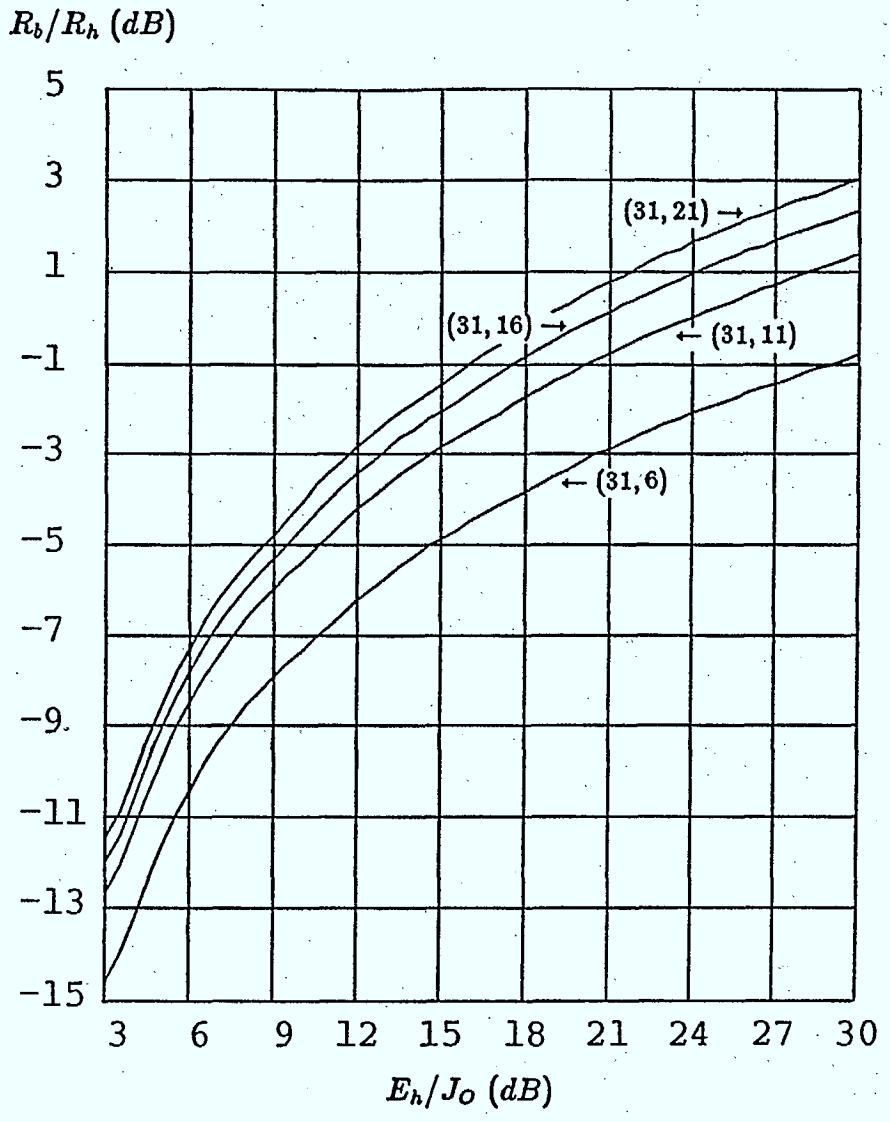


Figure 3.24: Throughput performance of the block length 31 BCH codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

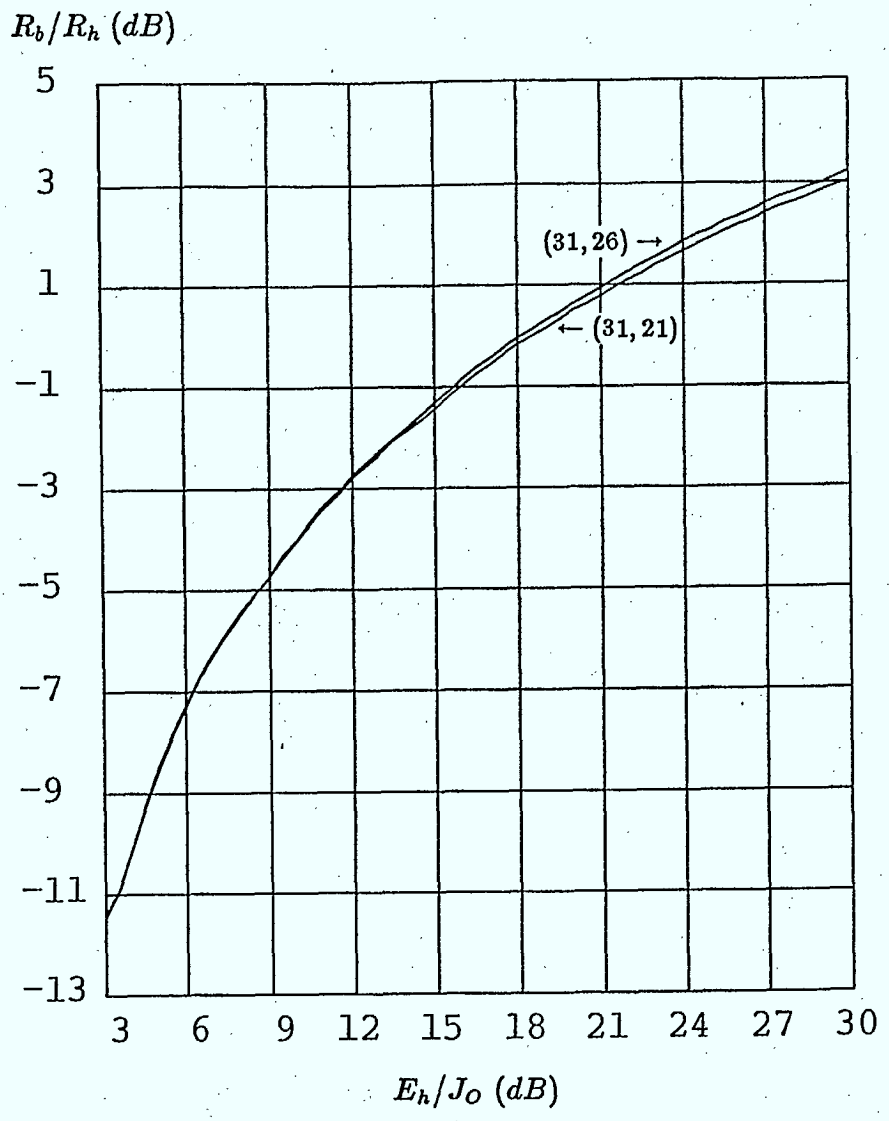


Figure 3.25: Throughput performance of the (31,21) and (31,26) BCH codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

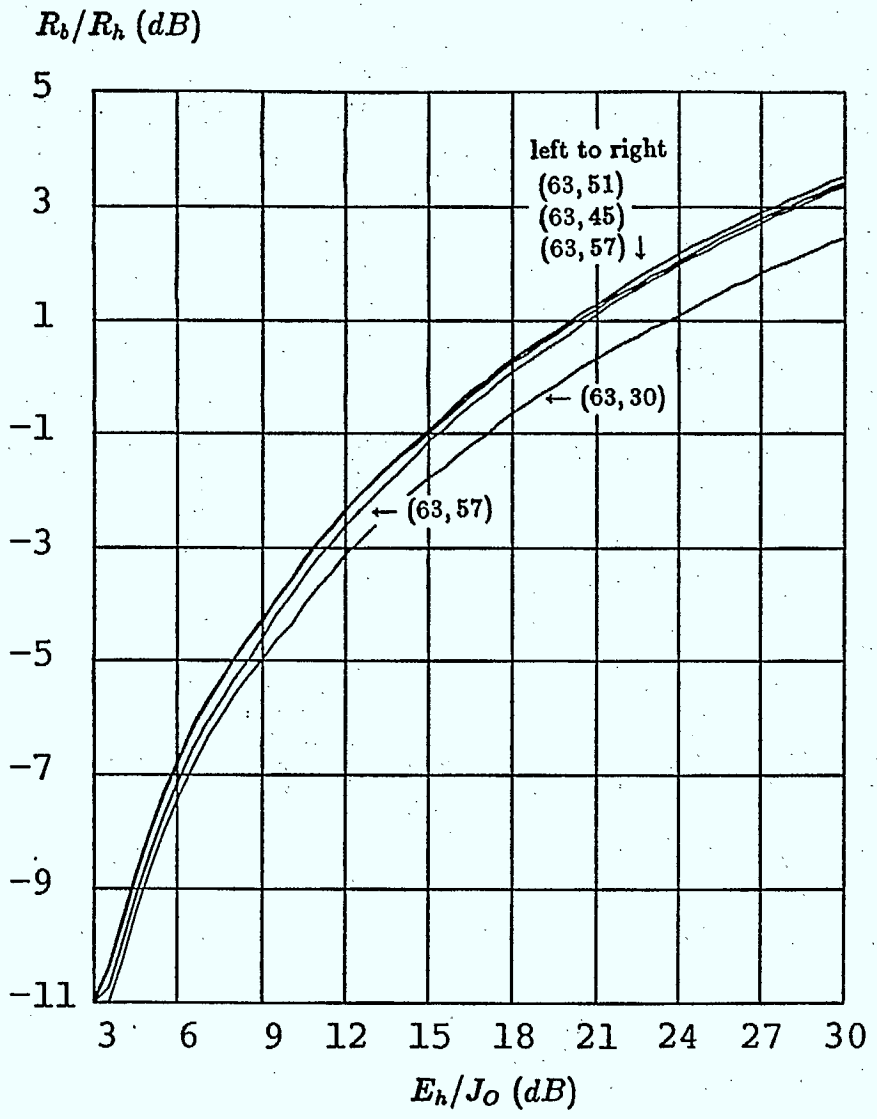


Figure 3.26: Throughput performance of the block length 63 BCH codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

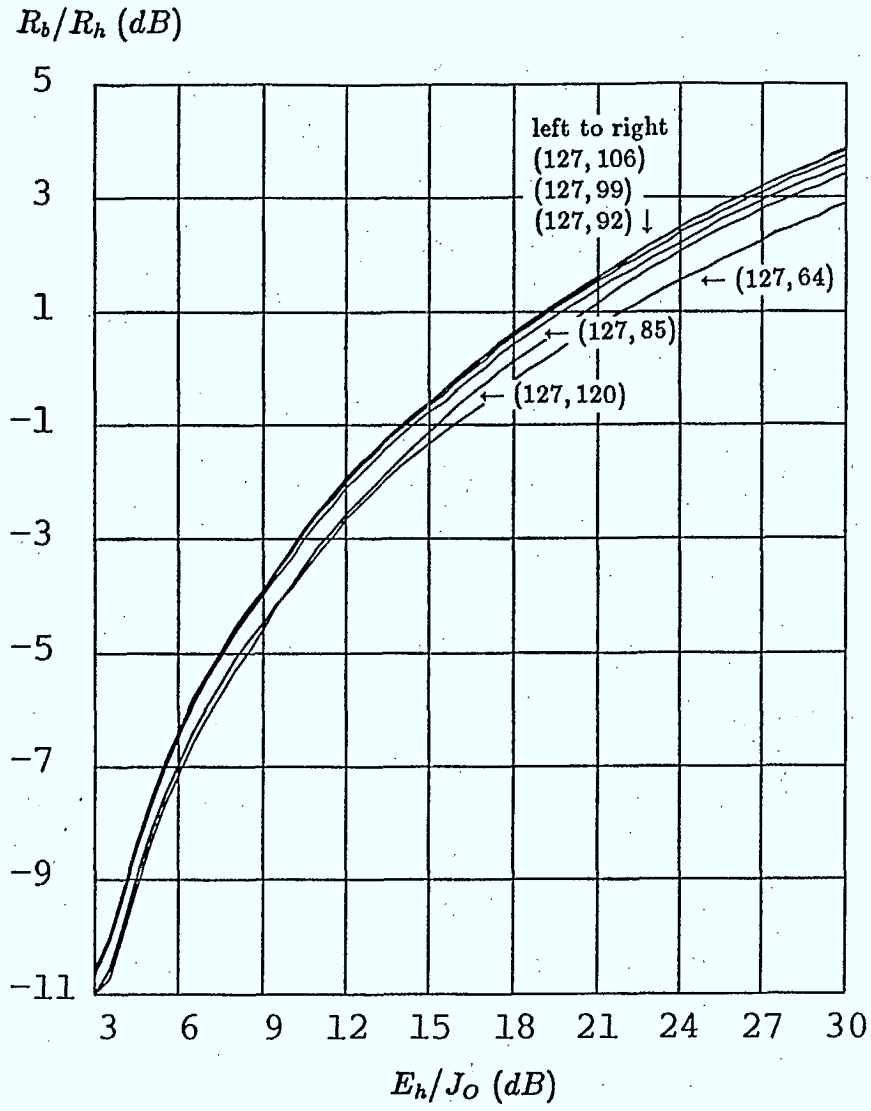


Figure 3.27: Throughput performance of the block length 127 BCH codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

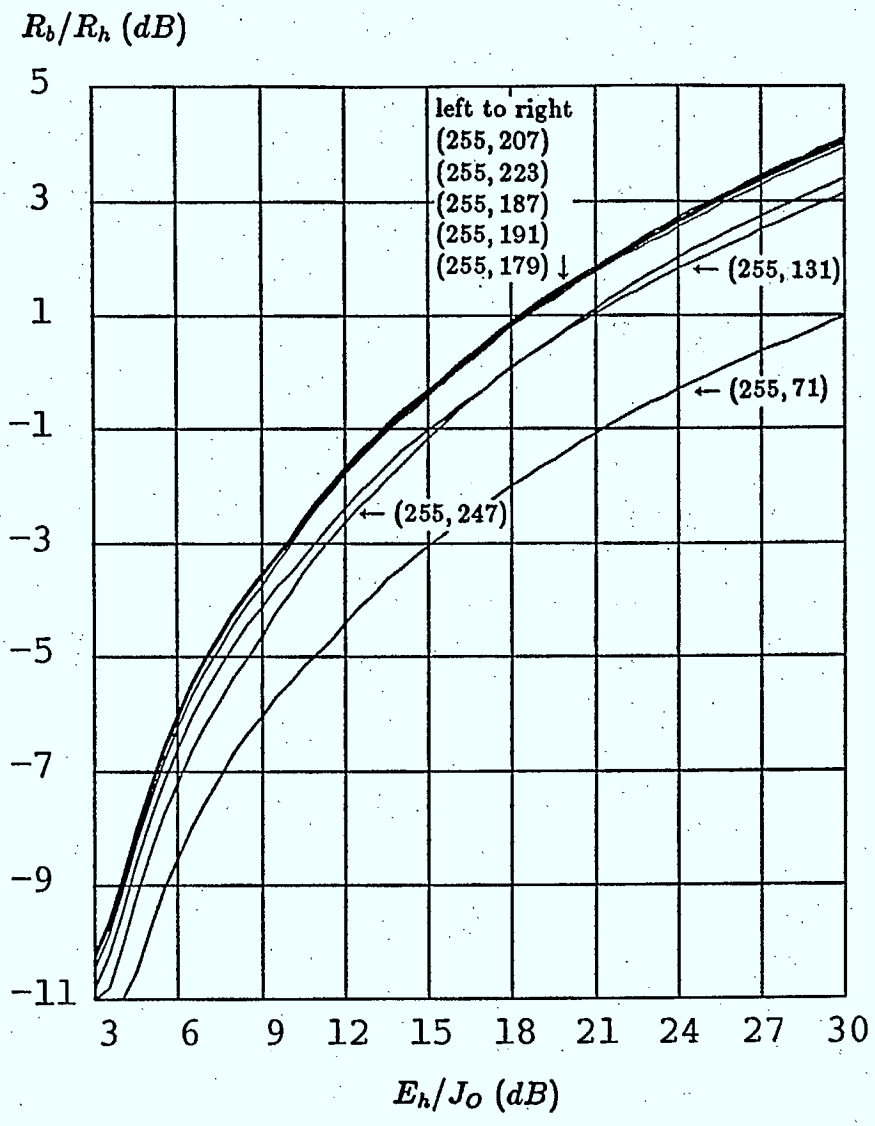


Figure 3.28: Throughput performance of the block length 255 BCH codes with MFSK and fixed hop rates under WC MT jamming, $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

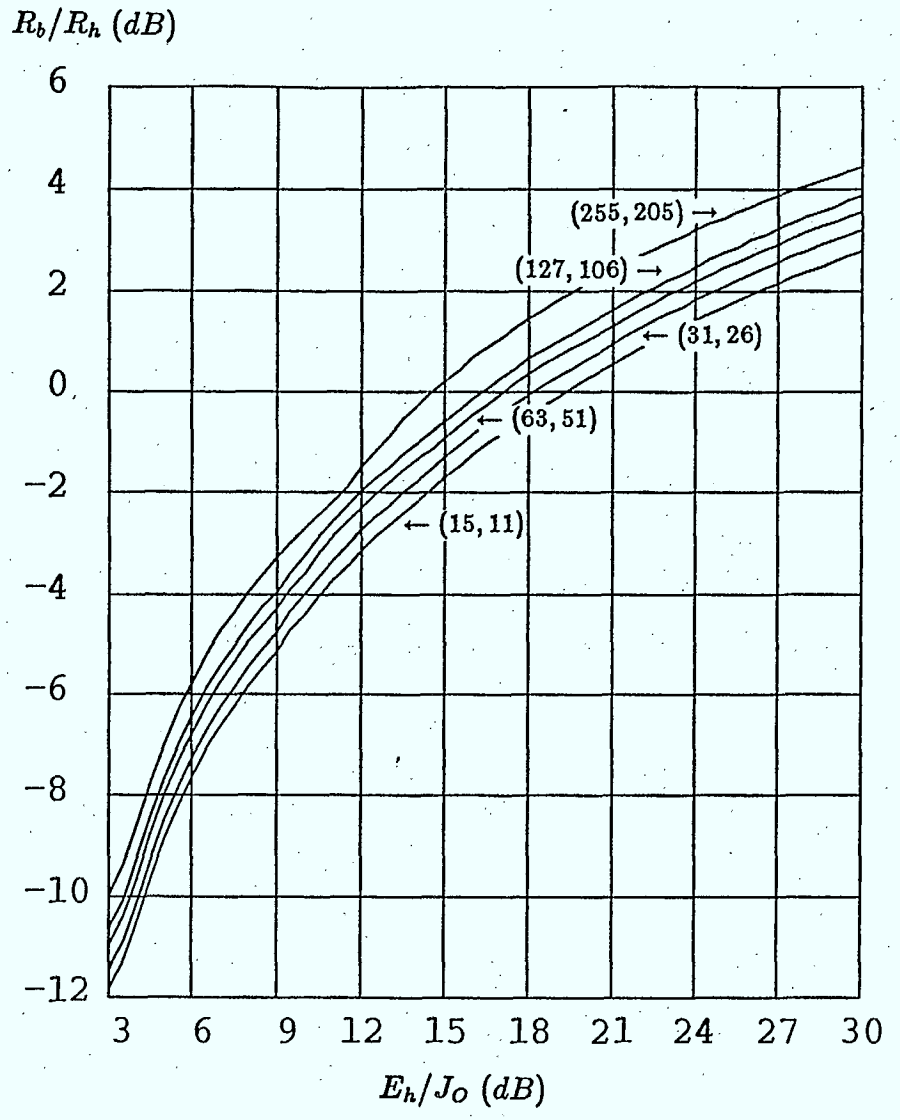


Figure 3.29: Throughput performance of the best block length 15, 31, 63, 127 and 255 BCH codes with MFSK and fixed hop rates under WC MT jamming, $M = 2^K$ and $K = 2$ to 5 and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

$P_b = 10^{-5}$ by doubling the length of the BCH code used. Of course this means an increase in codec complexity. As for fixed data rates, for block lengths larger than 15, there are a number of codes with near optimum performance. This provides a greater choice of codes to implement, which is important when cost and other factors are taken into account.

3.4.2 Reed-Solomon Codes

The Reed-Solomon codes examined in Chapter 2 are re-evaluated with a fixed hop rate. For a given code, we have a given Q, but several choices for K due to alphabet conversion, as discussed in Chapter 2. Thus we have a set of R_b/R_h vs. E_h/J_O curves for each Q with all valid K. The envelope corresponding to this set of curves was computed for each code, as was done for the BCH codes. The code which produced the best throughput for all E_h/J_O, ignoring minor crossings, was chosen as the optimum code for that length. The results are given in Figs. 3.30 to 3.34 for block lengths 15, 31, 63, 127 and 255, respectively. (For $K = 1$ WC PBN jamming is worst, while for $K > 1$ WC MT jamming is the worst.)

The best length 7 RS code is the (7,5) code. For code lengths 15, 31, 63, 127 and 255, we have optimum codes (15,11), (31,25), (63,53), (127,103), (255,221), respectively. These 5 best codes are shown in Fig. 3.35. It is obvious that codes which allow $K = 2$ to 5 produce the best results. Because 7 has no factors, the best block length 127 code has a very poor performance. Although the (255,221) code has the best overall performance, its curve meets that for the (63,53) code at $E_h/J_O = 11$ dB. This is due to the different possible K values and different transition points of these values for the codes. Thus, choosing a Q which allows a K that lies in the optimum region for uncoded systems, for the given E_h/J_O, tends to produce the best coding scheme for RS codes. Otherwise, doubling the code length will not improve the performance (see Fig. 3.35 for code length increases from 15 to 31, 63 to 127, etc.). As for fixed data rate systems, alphabet conversion is essential for optimizing

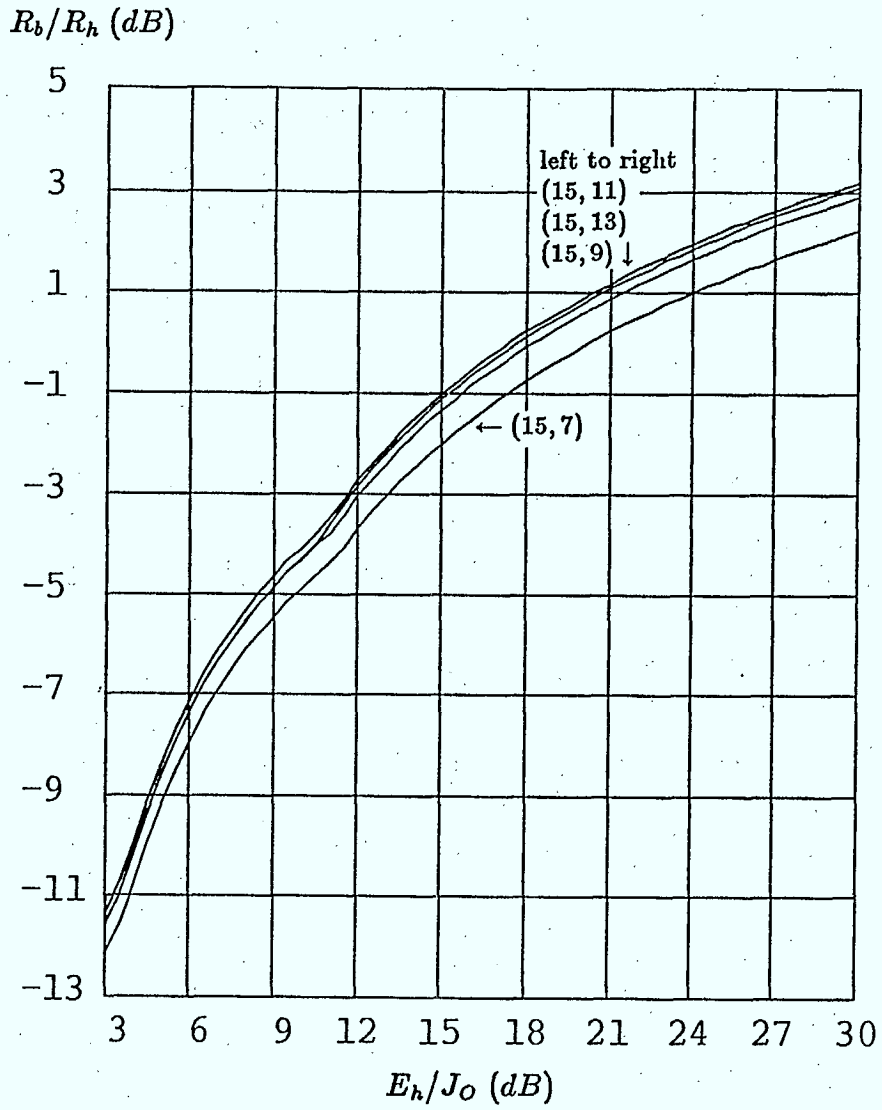


Figure 3.30: Throughput performance of the block length 15 RS codes with MFSK and fixed hop rates under WC MT jamming, $K = 2, 4$ and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

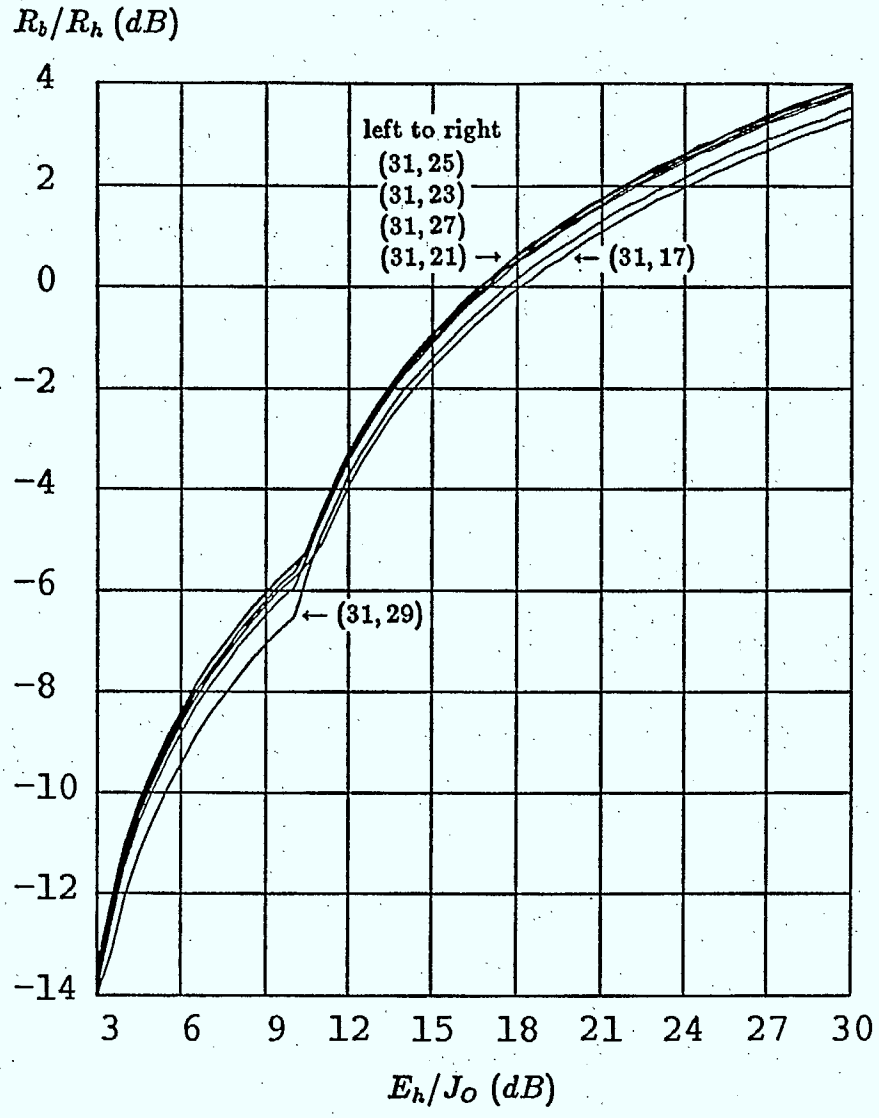


Figure 3.31: Throughput performance of the block length 31 RS codes with MFSK and fixed hop rates, under WC PBN jamming for $K = 1$ and WC MT jamming for $K = 5$, and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

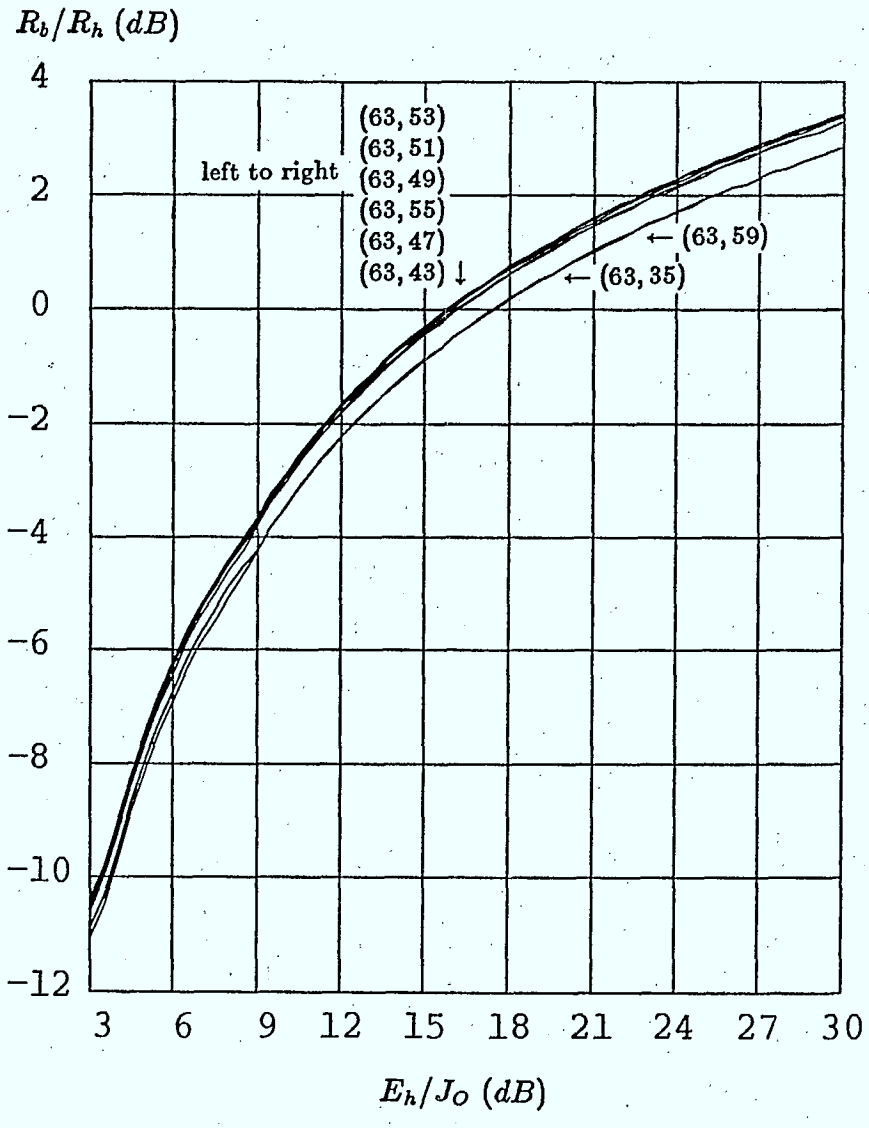


Figure 3.32: Throughput performance of the block length 63 RS codes with MFSK and fixed hop rates, under WC MT jamming, $K = 2, 3$ and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

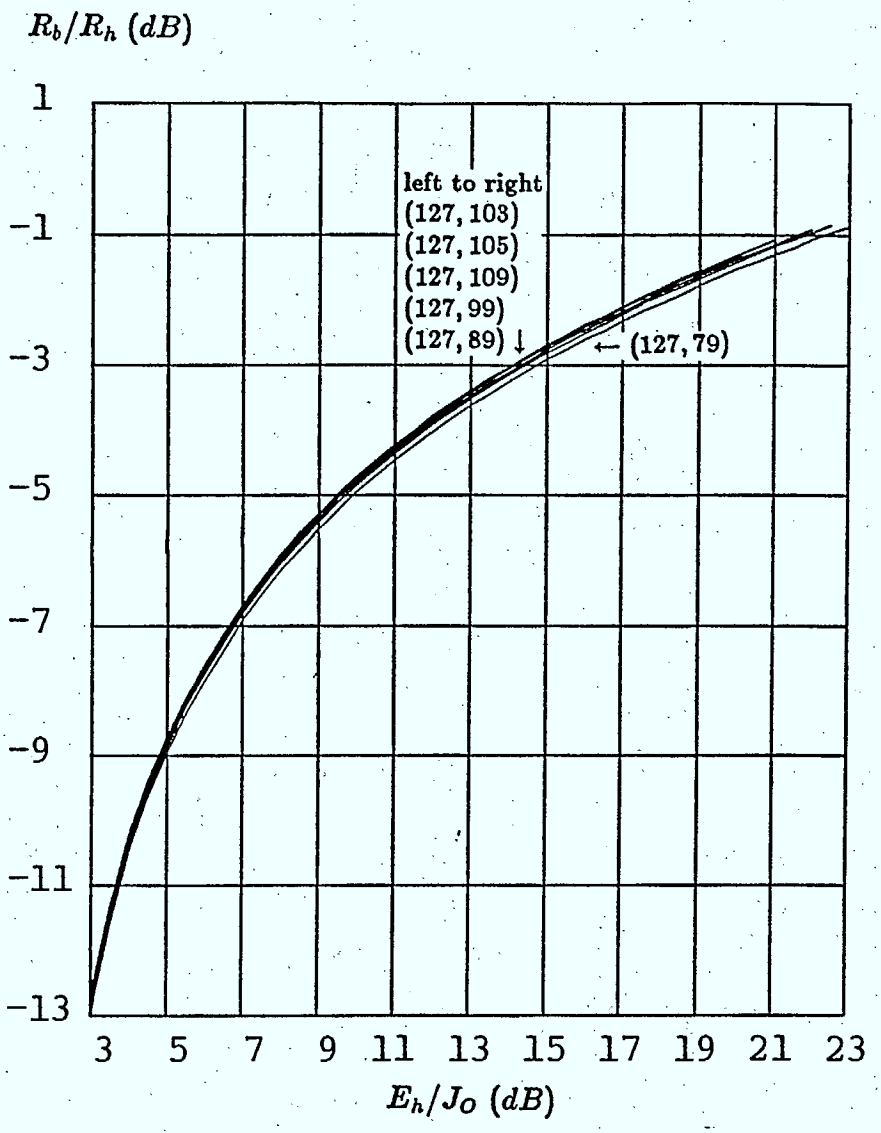


Figure 3.33: Throughput performance of the block length 127 RS codes with MFSK and fixed hop rates under WC PBN jamming, $K = 1$ and $P_b = 10^{-5}$.

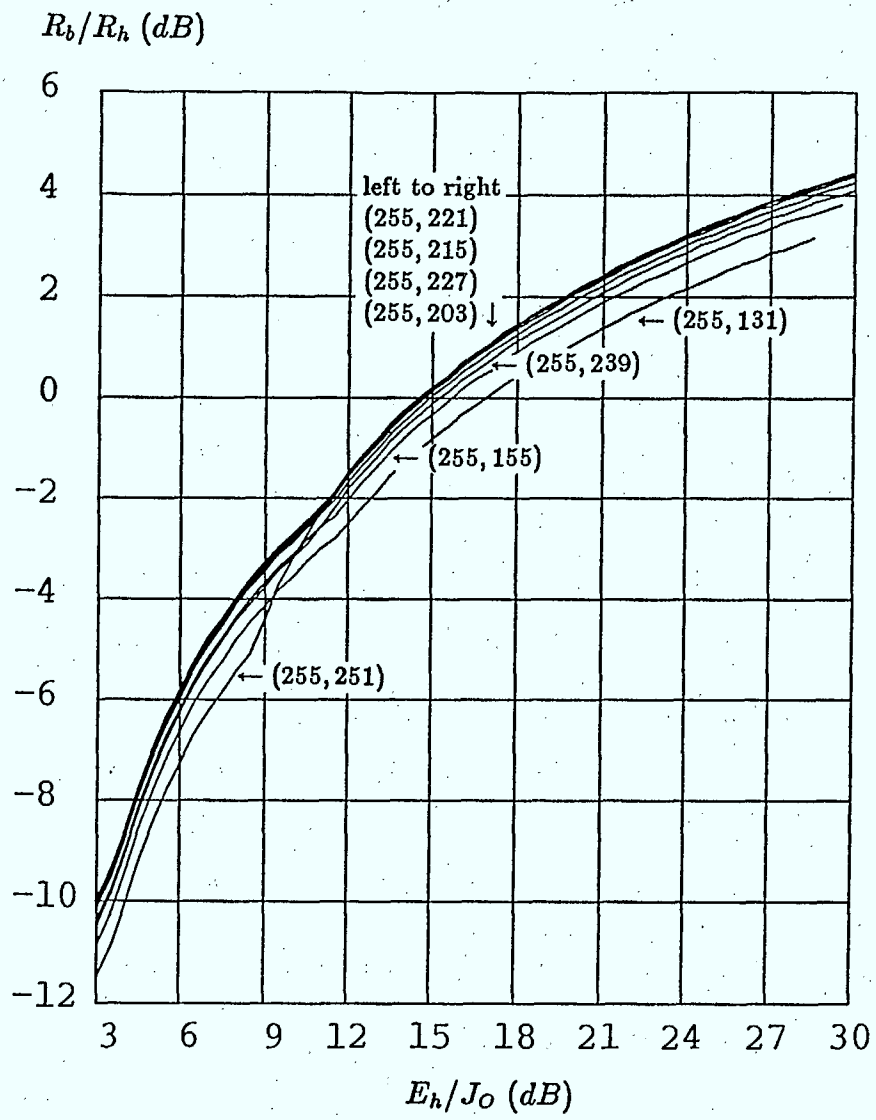


Figure 3.34: Throughput performance of the block length 255 RS codes with MFSK and fixed hop rates under WC MT jamming, $K = 2, 4$ and $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

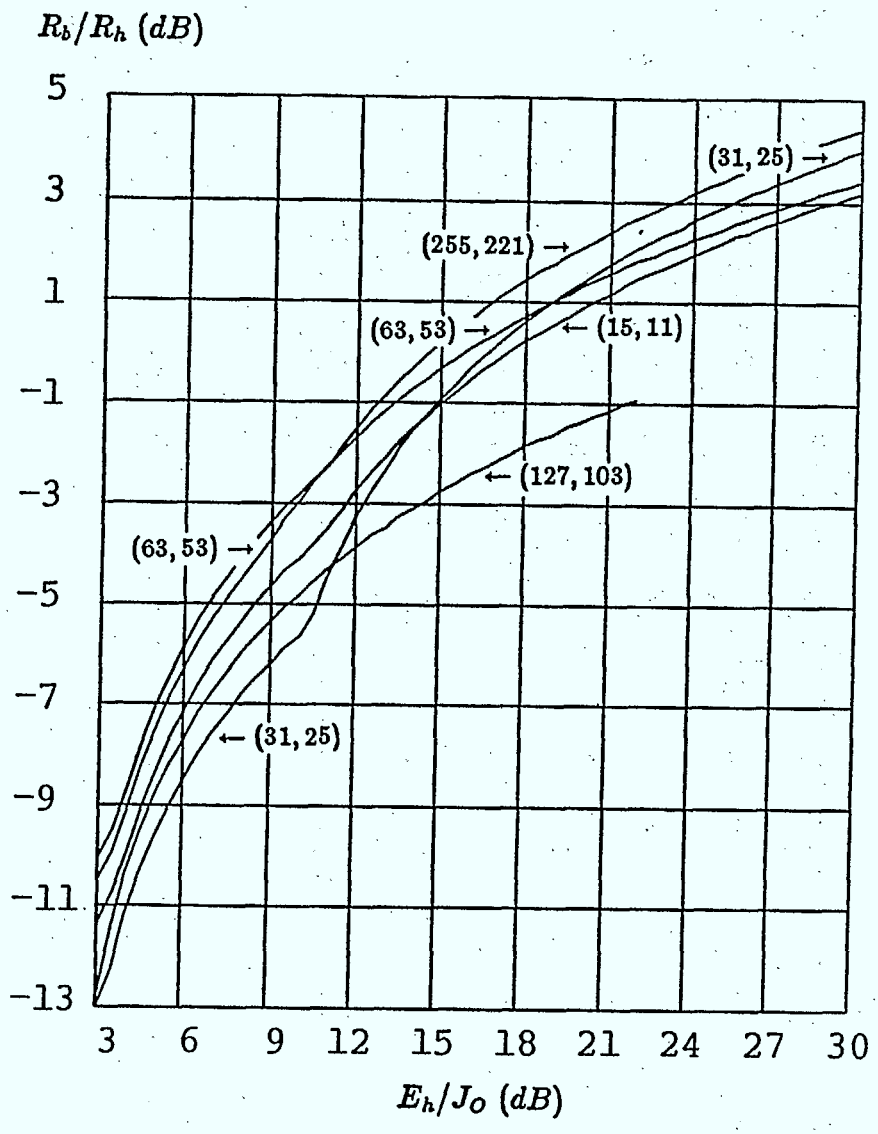


Figure 3.35: Throughput performance of the best block length 15, 31, 63, 127 and 255 RS codes with MFSK and fixed hop rates, under WC PBN jamming for $K = 1$, and WC MT jamming for $K = 2$ to 5, for $P_b = 10^{-5}$. Optimum K is assumed for a given E_h/J_O.

the use of RS codes, as the best coding schemes use this conversion. The best fixed data rate codes are the (15,11), (31,23), (127,99) and (255,203) codes, thus the best fixed hop rate codes have a higher rate than the best fixed data rate codes.

3.5 Concluding Remarks

With a fixed hop rate, the best convolutional codes are the rate 1/2 Trumpis code, the semi-orthogonal code for $K = 3$ and the Dual-K codes. Three best codes are given because although the Trumpis code is the best overall, the valid range of R_b/R_h varies between codes. The best block codes are the (255,221) RS code, the (255,207) BCH code, and the (63,53) RS code, in that order. As with a fixed data rate, the best convolutional codes perform better than the best block codes. The best block codes with a fixed data rate have a code rate which is higher than with a fixed data rate, but the differences are slight, and often negligible.

Other codes may be examined using the same basic principles presented here. Although we have stated that certain EC codes are the best, optimization of the code design should take into account other constraints such as the system complexity. Our results should serve as a guide to error correcting code selection for a particular application.

Chapter 4

Performance of Error-Erasure Correction Decoding of Reed-Solomon Codes for Frequency Hop Communications in Multitone Interference

4.1 Introduction

In this Chapter, we examine a recently proposed combination of spread-spectrum modulation, EC coding, diversity and error-erasure correction decoding to combat partial band noise jamming[13] which is very promising. The system employs non-coherent MFSK, Reed-Solomon (RS) coding and parallel error-erasure correction decoding. It has been shown to be very effective against PBN jamming. However, it is well known that partial band noise jamming is not the only effective type of jamming. In fact, multitone jamming can be more effective against FH/MFSK signals than partial band noise[2,3]. The reason being that continous wave tones are the most efficient way for a jammer to inject energy into

the non-coherent detectors. In the following sections we evaluate the performance of the system with parallel error-erasure correction decoding of RS codes in multitone jamming. As in the previous Chapters, the only multitone jamming considered is one jamming tone per jammed M-ary band, with $0 < \alpha < 1$. We show that when the redundancy is not large, multitone jamming, from the jammer's point of view, tends to be more effective against the system than partial band noise jamming, as M increases. Note that in partial band noise jamming, increasing M improves performance from the communicator's point of view[13]. This implies that for a robust communication system, the multitone jamming threat must be considered. Optimization of the design in worst case multitone jamming is studied in terms of the modulation, code rate and diversity factor.

Performance evaluation in this Chapter was done using an exact method rather than the prevailing bounding methods. The accuracy in results gained by using the exact method reveals details particular to multitone jamming which would not have been evident if a bounding technique were used.

The system we consider is the same as in the previous Chapters except for the generation of erased symbols and the use of parallel error-erasure correction decoding instead of error correction only. Basic assumptions are repeated as follows.

- Non-coherent M-ary FSK is used to transmit M-ary symbols. Orthogonal frequency spacing is assumed.
- Fast frequency-hopping is used so that one M-ary symbol is transmitted over L independent hops, where L is greater than or equal to 1. Perfect side information is assumed available to detect if a hop is jammed. This is a practical assumption in the case of multitone jamming, because a hop is jammed if and only if there are two or more tones observed over the transmitted M-ary band. If at least one of the L hops of an M-ary symbol is not jammed, the M-ary symbol is assumed to be received correctly. Otherwise, a hard decision on which M-ary symbol was transmitted is made

based on the linear combination (direct sum) of the received energy over L hops, and an erasure flag is attached to the M-ary symbol.

- (n, k) Q-ary RS codes are considered, where $Q \geq M$. One Q-ary symbol consists of C M-ary symbols, where C is an integer greater than or equal to 1. Thus $C = \log_2 Q / \log_2 M$. If any of the C M-ary symbols of a Q-ary symbol has an erasure flag attached, the Q-ary symbol will also have an erasure flag attached. Otherwise, the Q-ary symbol is assumed to be correct. Suppose d is the minimum distance of the RS code used. If the number of Q-ary symbols with erasure flags attached in a RS codeword is smaller than d, these symbols are erased and erasure correction is performed. Otherwise error correction is performed.

In the next section, formulas for evaluation of the bit error rate (BER) performance of the system in multitone jamming are given, followed by the determination of the worst case jamming parameter. In Section 3.3, a comparison is made between the performance of the system in worst case partial band noise jamming and in worst case multitone jamming. Optimization of the design is presented in Section 3.4 in terms of the modulation, coding rate and diversity factor.

4.2 Performance Evaluation in Multitone Jamming

Worst case multitone and partial band jamming were described in Chapter 1. Let μ be the probability that an M-ary band contains at least one jamming tone, where $M = 2^K$. It can easily be shown that[3]

$$\mu = \begin{cases} \frac{\alpha M}{E_h/J_O} & E_h/J_O > \alpha M; \\ 1 & otherwise,\ (E_h/J_O = \alpha M) \end{cases} \tag{4.1}$$

where E_h is the energy per hop. Note that when $\mu = 1$, i.e., one jamming tone in every M-ary band, it can be shown that we must have $E_h/J_O = \alpha M$. Suppose an (n, k) Q-ary

RS code, MFSK and diversity L are used, then

$$E_h = E_b \times \frac{Kk/n}{L}.$$

A hop is attached an erasure flag with the probability

$$\delta = \mu \frac{M-1}{M}. \tag{4.2}$$

An M-ary symbol is then attached an erasure flag with the probability δ^L. A Q-ary symbol of an RS codeword, which consists of C M-ary symbols, is attached an erasure flag with the probability

$$\Delta = 1 - (1-\delta^L)^C. \tag{4.3}$$

Since the RS decoder performs error correction decoding only when there are d or more Q-ary symbols in a codeword with erasure flags attached, the probability that the decoder performs error correction is given by

$$P_{error-c} = \sum_{i=d}^{n} \binom{n}{i} \Delta^i (1-\Delta)^{n-i}. \tag{4.4}$$

Under the assumption of perfect side information and no background noise, when the decoder performs erasure correction all errors can be cleared by the decoder. That is, there can be no bit errors at the decoder output in this case. Thus the BER at the output of the parallel error-erasure correction RS decoder is

$$P_b = P_b' P_{error-c} \tag{4.5}$$

where P_b' is the BER at the decoder output when error correction is performed. A reasonable assumption is that at the RS decoder output, an erroneous Q-ary symbol is equally likely to be any of the $Q-1$ possible incorrect Q-ary symbols. Then we have [9]

$$P_b' = P_s' \frac{Q}{2(Q-1)} \tag{4.6}$$

where P_s' is the Q-ary symbol error probability at the decoder output when error correction is performed. $t = \lfloor d-1)/2 \rfloor$ is the number of errors that are guaranteed correctable by the decoder when error correction is performed, where $\lfloor x \rfloor$ is the integer part of x. Then with hard decision error correction decoding, P_s' is related to the Q-ary error probability at the decoder input, P_i, by (see e.g. [4])

$$P_s' \approx \frac{1}{n} \sum_{j=t+1}^{n} j \binom{n}{j} P_i^j (1-P_i)^{n-j}. \tag{4.7}$$

Since a Q-ary symbol consists of C M-ary symbols, P_i is related to the M-ary symbol error probability at the decoder input P_e by

$$P_i = 1-(1-P_e)^C. \tag{4.8}$$

For $M = 2$,

$$P_e = \delta^L = (\mu/2)^L. \tag{4.9}$$

For $M > 2$,

$$P_e = \delta^L \frac{S_{M-1,L}(c)}{(M-1)^L} \tag{4.10}$$

where $c \equiv \lceil L\alpha \rceil$ ($\lceil x \rceil$ is the smallest integer greater than or equal to x) and $S_{M-1,L}(c)$ is a monotonically decreasing function of c given as follows[3]. For $c = 1$ or 2,

$$S_{M-1,L}(c) = \begin{cases} (M-1)^L & c=1; \\ (M-1)^L - \frac{(M-1)!}{(M-1-L)!} & c=2 \quad L \le M-1; \\ (M-1)^L & c=2 \quad L > M-1 \end{cases} \tag{4.11}$$

Otherwise if $L/3 < c \le L$,

$$\begin{aligned} S_{M-1,L}(c) &= (M-1)\sum_{i=c}^{L} \binom{L}{i} (M-2)^{L-i} \\ &- \binom{M-1}{2} \sum_{i=c}^{[L/2]} \binom{L}{i} \binom{L-i}{i} (M-3)^{L-2i} \\ &- (M-1)(M-2) \sum_{i=c}^{[L/2]} \binom{L}{i} \sum_{j=i+1}^{L-i} \binom{L-i}{j} (M-3)^{L-i-j} \end{aligned} \tag{4.12}$$

Table 4.1: Possible Practical Combinations of MFSK ($M = 2^K$) Signalling and Q-ary ($Q = 2^q$) RS coding

K	q					
1	3	4	5	6	7	8
2		4		6		8
3	3			6		
4		4				8
5			5			
6				6		

where the convention is adopted that $\sum_{i=a}^{b} = 0$ if $a > b$. Using (4.11) and (4.12) we can compute any $S_{M-1,L}(c)$ for $L \leq 8$ over the entire range of $1 \leq c \leq L$.

Due to their practicality, certain combinations of MFSK modulation and Q-ary RS coding (with length Q or $Q-1$) are considered in this Chapter, as shown in Table 4.1.

Now we determine the worst case α, from the communicator's point of view, that maximizes the final BER P_b. For a given communication system, i.e., a fixed L, M, RS code and signal to noise ratio (E_b/J_O), from (4.2) to (4.4), it can be observed that $P_{error-c}$ increases as μ increases. From (4.5), it is clear that if P_b' increases as μ increases, maximizing P_b is equivalent to maximizing μ. This condition in turn means, from (4.6) to (4.8), that P_e increases if μ increases. For $M = 2$, from (4.9), this condition is obviously true. For $L = 1$, $S_{M-1,1}(c)$ in (4.10) is a constant, thus this condition is also true. Therefore, in these cases, the worst case α is given by

$$\alpha_{wc} = \begin{cases} 1_- & E_h/J_O \geq M; \\ \frac{E_h/J_O}{M} & E_h/J_O < M \end{cases} \tag{4.13}$$

where x_- means a value less than but infinitely close to x. Note that since $\mu \leq 1$, α is always constrained by

$$\alpha \leq \alpha_b = min\left\{\frac{E_h/J_O}{M}, 1_-\right\}. \tag{4.14}$$

For other M and L, however, P_e is not, in general, a monotonic function of α. To illustrate this, we plot $\beta(\alpha) = \alpha^L S_{M-1,L}(c)/(M-1)^L$ vs. α for $M = 16$ and $L = 2, 4, 6$ and 8 in Fig. 4.1. Within the region of α specified by (4.14), P_e is proportional to $\beta(\alpha)$, ($\beta(\alpha)$ is an intermediate variable created solely to illustrate the relationship clearly). It can be seen that $\beta(\alpha)$ is discontinous and has $L-1$ peaks at $\alpha = \alpha_l = (l/L)_-$ for $l = 1, \cdots, L-1$. By use of the Chernoff union bound, it can be shown that the highest peak is approximately α_0, as given in Table 2.1 [3].

It should be emphasized that for parallel error-erasure correction decoding, the term $P_{error-c}$ plays an important role. Since it is less than 1, P_b is less than P_b'. Also, it increases as α increases, forcing the worst case α away from (larger than) the optimum value (from the jammer's point of view) that would exist if only error correction decoding were used. Therefore, the overall performance of the parallel error-erasure correction decoding is better than that with error correction decoding alone.

For parallel error-erasure correction decoding, and $M > 2$ and $L > 1$, α_{wc} should be chosen from α_l inside the region specified by (4.14) and α_b that maximizes P_b. This is computionally simple, since L is not large. It has been observed that either α_b or the closest α_l, denoted as α_c, tends to be α_{wc}. α_c is given by

$$\alpha_c = ([\alpha_b L]/L)_- \tag{4.15}$$

4.3 Comparison Between Worst Case Partial Band Noise Jamming and Multitone Jamming

In this section, we compare the BER performance of the system in worst case multitone jamming with that in worst case partial band noise jamming. For a fixed RS code, L and M, we compare the E_b/J_O required for $P_b = 10^{-4}$ in the two types of jamming. Selected results are shown in Table 4.2. The results for partial band noise jamming are from [13]. For binary FSK, (4.1) to (4.3), (4.9), etc., show that as long as $E_h/J_O < 2$ (so that

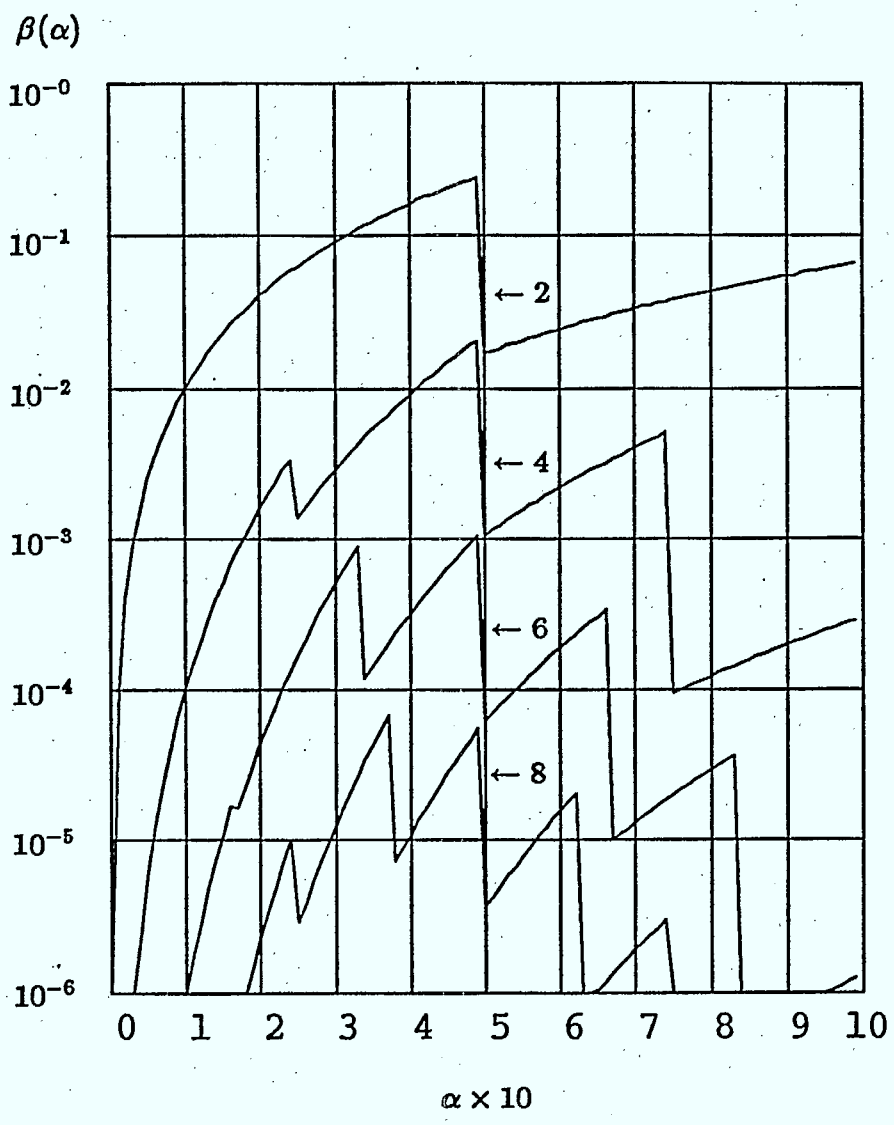


Figure 4.1: β vs. α for $M = 16$ and $L = 2, 4, 6$ and 8.

Table 4.2: Selected results of the comparison between E_b/J_O (dB) required for $P_b = 10^{-4}$ under worst case partial band noise (PBN) jamming and multitone (MT) jamming. * indicates that P_b is always below 10^{-4}.

k	L	n	K	E_b/J_O(PBN)	E_b/J_O(MT)
5	2	32	1	15.98	15.13
10	4	32	1	14.31	*
15	6	32	1	13.77	*
20	8	32	1	13.42	*
10	2	64	1	15.69	15.02
20	4	64	1	14.12	*
30	6	64	1	13.46	*
35	7	64	1	13.35	*
40	8	64	1	13.20	*
30	3	128	1	14.64	*
40	4	128	1	14.04	*
80	4	256	1	14.03	*
5	2	16	4	9.15	13.05
40	1	256	4	9.90	16.75
80	2	256	4	7.97	12.94
120	3	256	4	7.16	12.96
160	4	256	4	6.82	12.50
200	5	256	4	6.81	11.83
8	1	32	5	8.45	17.15
16	2	32	5	6.76	13.75
24	3	32	5	6.63	13.90
27	1	64	6	6.88	18.13
53	2	64	6	7.11	16.45

under worst case multitone jamming $\mu = 1$), P_b is independent of E_b/J_O. Thus increasing L and/or the number of parity check symbols in the RS code, $(n-k)$, will reduce P_b to as small a value as required over all E_b/J_O, because P_b is a monotonically decreasing function of E_b/J_O. This is reflected in Table 4.2, where * indicates that P_b is always smaller than 10^{-4} for the specific combination of RS coding and diversity L. This is true for $M > 2$ also, but a larger redundancy is required. Thus it is not reflected in Table 4.2. From this table it is clear that, from the jammer's point of view, for $M = 2$, partial band jamming is more effective. As M increases, the effectiveness of multitone jamming increases, while that of partial band jamming decreases. Thus for nonbinary FSK, and a redundancy which is not large (as explained in the next section), multitone jamming tends to be more effective than partial band noise jamming. If the communication system is expected to perform well in both partial band and multitone jamming, the system design must consider multitone jamming. In the next section, we optimize the design of the system in multitone jamming in terms of the modulation, diversity and RS coding.

4.4 Optimization of the System Design

In the design of the communication system, there are three parameters to be determined: the M-ary FSK modulation, i.e., M, the diversity factor L, and the RS code size, i.e., n and k. Since the choice of one parameter is related to that of another, they must all be considered during optimization. For each combination the worst case jamming is found using the method described in Section 4.2. Then the BER performance of the different combinations of system parameters is compared and the best one found. One constraint is the RS code length n. Since the minimum distance d of an RS code is proportional to the code length, and the complexity of the codec (linearly or approximately linearly) increases with n [14], it makes sense to make comparisons between codes of fixed length. $1 < k < n-1$ are considered such that $n - k + 1 = d = 2t + 1$.

As discussed in the last section, as long as there is enough redundancy, i.e., a large diversity factor L and/or number of parity check symbols in the RS code, $(n-k)$, the BER can be made arbitrarily small for any E_b/J_O. Suppose we require $P_b \leq 10^{-5}$. The redundancy required can be expressed, for a fixed n and L, in terms of k^*, the largest k such that the (n,k) RS code with diversity L can provide $P_b \leq 10^{-5}$ for any E_b/J_O. Obviously, an increase in L and/or a decrease in k will also result in $P_b \leq 10^{-5}$ for all E_b/J_O. From these results we can conclude that the worst type of multitone jamming, one jamming tone per M-ary band, is completely nullified by a system with large redundancy. Thus the jammer is forced to consider the use of multiple jamming tones per M-ary band, which is worth further study but is not considered here.

For a low redundancy (small L and/or number of parity check symbols), there may not exist a k^*. In this case, for a fixed n, M and L, there is an optimum k, denoted as k', that gives the smallest E_b/J'_O required for a given P_b due to the trade-off between E_h and d. Obviously for some L and n there exists both a k^* and a k'. In this case, k' can be interpreted as the local minimum point. As an example, Fig. 4.2 shows the BER performance of the $(31,k)$ RS codes for k equal to 5, 7, 11, 15, 19, 21, 23 and 27, and $L=3$ and $M=2$, where $k'=21$ and $k^*=5$. Clearly for large L there may not exist a k'. In Tables 4.3 through 4.8 we list k' together with the corresponding α_{wc}, E_b/J'_O required for $P_b \leq 10^{-5}$, and k^* for different M, n and L.

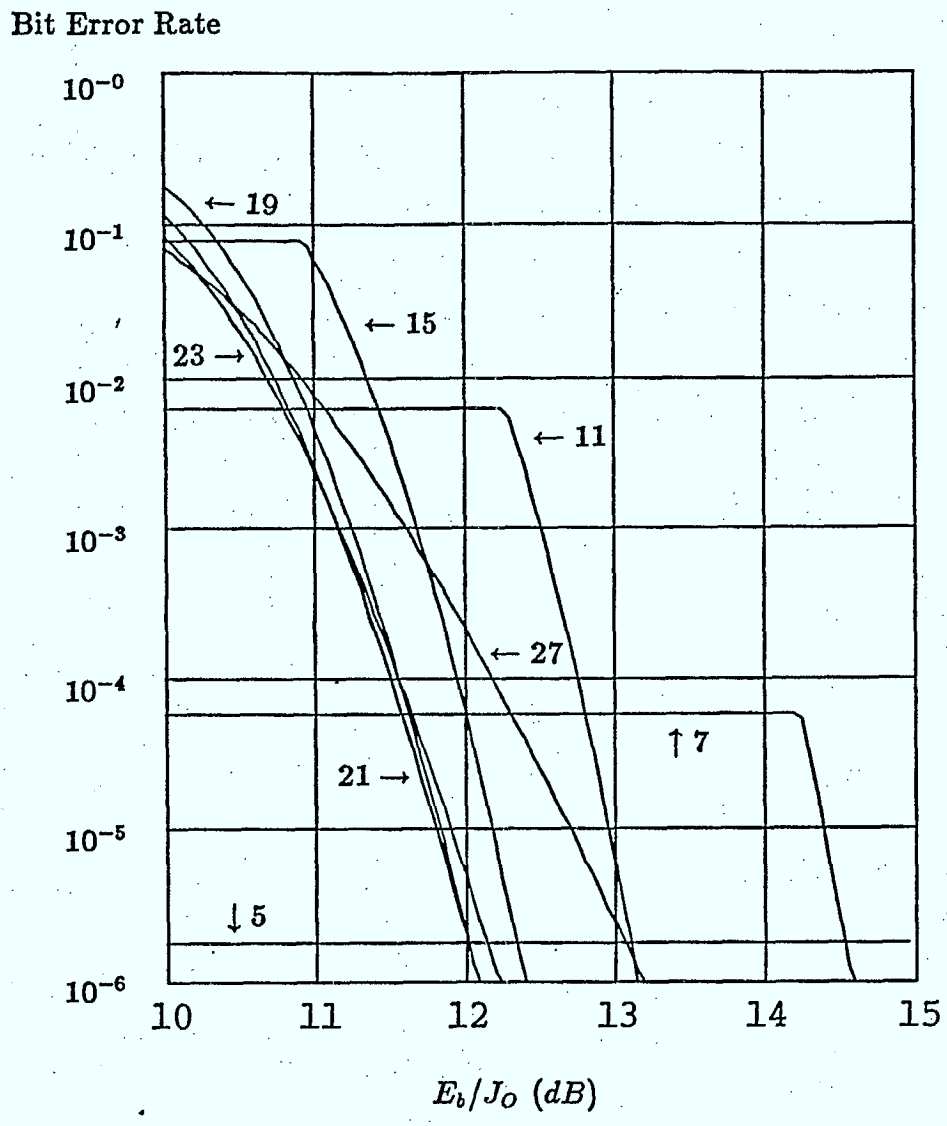


Figure 4.2: BER performance of the $(31, k)$ RS codes, $k = 5, 7, 11, 15, 19, 21, 23$ and 27, with diversity factor $L = 3$ and FH/BFSK in worst case multitone jamming. $k' = 21$ and $k^* = 5$.

Table 4.3: Optimum $(7, k')$ RS codes for different M-ary FSK and diversity factor L, and E_b/J_O' required for $P_b = 10^{-5}$.

	L	1	2	3	4	5	6	7	8
$M = 2$	k'	3	3	5	5	5	5	-	-
	E_b/J_O'	16.98	13.35	12.46	12.15	12.19	12.36	-	-
	α_{wc}	1.0	1.0	1.0	1.0	1.0	1.0	-	-
	k^*	-	-	-	-	3	3	5	5
$M = 8$	k'	3	3	5	5	5	5	5	5
	E_b/J_O'	16.09	13.15	12.22	12.20	12.28	12.43	12.74	12.89
	α_{wc}	1.0	1.0	.66	.74	.79	.66	.71	.62
	k^*	-	-	-	-	-	-	-	-

Table 4.4: Optimum $(15, k')$ RS codes for different M-ary FSK and diversity factor L, and E_b/J_O' required for $P_b = 10^{-5}$.

	L	1	2	3	4	5	6	7	8
$M = 2$	k'	5	9	9	11	11	13	13	-
	E_b/J_O'	16.03	12.82	12.05	11.88	11.96	12.19	12.32	-
	α_{wc}	1.0	1.0	1.0	1.0	1.0	1.0	1.0	-
	k^*	-	-	-	5	9	11	11	13
$M = 4$	k'	5	9	11	11	11	11	11	7
	E_b/J_O'	14.94	12.25	11.63	11.37	11.41	11.71	11.74	11.05
	α_{wc}	1.0	1.0	1.0	1.0	.79	.83	.71	.37
	k^*	-	-	-	-	-	3	5	5
$M = 16$	k'	5	9	11	11	11	11	11	13
	E_b/J_O'	16.25	13.30	13.18	12.82	13.04	13.04	13.00	13.41
	α_{wc}	1.0	1.0	.66	.74	.59	.49	.42	.49
	k^*	-	-	-	-	-	-	-	-

Table 4.5: Optimum $(31, k')$ RS codes for different M-ary FSK and diversity factor L, and E_b/J_O' required for $P_b = 10^{-5}$.

	L	1	2	3	4	5	6	7	8
$M = 2$	k'	11	19	21	23	25	25	27	29
	E_b/J_O'	15.56	12.53	11.81	11.70	11.80	11.95	12.19	12.61
	α_{wc}	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	k^*	-	-	5	13	19	23	25	27
$M = 32$	k'	13	19	11	23	25	21	19	17
	E_b/J_O'	17.16	14.01	13.96	14.11	13.85	13.77	14.11	14.18
	α_{wc}	1.0	.49	.32	.49	.39	.32	.28	.24
	k^*	-	-	-	-	-	-	-	-

Table 4.6: Optimum $(63, k')$ RS codes for different M-ary FSK and diversity factor L, and E_b/J_O' required for $P_b = 10^{-5}$.

	L	1	2	3	4	5	6	7	8
$M = 2$	k'	25	39	45	49	51	51	55	61
	E_b/J_O'	15.28	12.33	11.68	11.58	11.67	11.84	12.08	12.50
	α_{wc}	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	k^*	-	-	13	29	41	49	53	57
$M = 4$	k'	25	39	43	49	53	47	41	39
	E_b/J_O'	14.20	11.90	11.02	10.89	11.17	11.06	11.26	11.18
	α_{wc}	1.0	1.0	1.0	.74	.79	.66	.57	.49
	k^*	-	-	-	7	13	19	25	33
$M = 8$	k'	27	39	45	47	49	47	39	53
	E_b/J_O'	14.57	11.97	11.56	11.34	11.49	11.39	11.71	11.80
	α_{wc}	1.0	1.0	.66	.74	.59	.49	.42	.49
	k^*	-	-	-	-	-	5	9	11
$M = 64$	k'	27	39	41	33	23	55	53	53
	E_b/J_O'	18.53	15.76	14.44	14.28	15.16	15.48	15.36	15.14
	α_{wc}	1.0	.49	.32	.24	.19	.32	.28	.24
	k^*	-	-	-	-	-	-	-	-

Table 4.7: Optimum $(127, k')$ RS codes for different M-ary FSK and diversity factor L, and E_b/J_O' required for $P_b = 10^{-5}$.

	L	1	2	3	4	5	6	7	8
$M = 2$	k'	51	79	95	99	103	107	113	119
	E_b/J_O'	15.10	12.21	11.62	11.49	11.59	11.78	12.05	12.42
	α_{wc}	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	k^*	-	3	29	59	83	101	111	117

These tables provide information for optimum system design under two different conditions: high overall code rate, $r_{all} = kK/(nL)$, and low overall code rate. Constraints on r_{all} may arise from a fixed hop rate R_h requirement [2], because

$$r_{all} = \frac{R_b}{R_h}$$

where R_b is the information bit rate. Hence for a fixed hop rate R_h, a larger r results in a higher throughput. For high r, designs based on the local minimum point k' can be considered. The optimum combinations of M'FSK modulation, (n, k') RS codes and diversity L' are listed in Table 4.9 together with the corresponding E_b/J_O' required for $P_b \leq 10^{-5}$, and α_{wc}. These combinations give the smallest E_b/J_O' for a given code length n. The BER performance of these optimum combinations in worst case multitone jamming are shown in Figs. 4.3 and 4.4. From Table 4.9 it is observed that 4-ary FSK provides the best performance, (recall that in partial band noise jamming, a large M is preferred). The corresponding r_{all} values range from 0.37 to 0.46. The (63,49) code with $L = 4$ is particularly attractive due to its large $r = 0.39$, short code length $n = 63$ and small E_b/J_O. If $M = 4$ does not provide satisfactory BER performance in partial band noise jamming, M can be increased. Tables 4.3 through 4.8 provide the needed information for comparison.

If a low overall code rate r_{all} is tolerable, we can use k^* in the system design. For

Table 4.8: Optimum $(255, k')$ RS codes for different M-ary FSK and diversity factor L, and E_b/J_O' required for $P_b = 10^{-5}$.

	L	1	2	3	4	5	6	7	8
$M = 2$	k'	107	153	187	199	211	215	227	239
	E_b/J_O'	15.08	12.11	11.52	11.43	11.56	11.73	11.99	12.35
	α_{wc}	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	k^*	-	7	57	121	171	205	225	237
$M = 4$	k'	107	167	179	205	205	211	187	167
	E_b/J_O'	14.00	11.90	10.47	10.83	10.79	10.91	10.94	10.99
	α_{wc}	1.0	1.0	1.0	.74	.79	.66	.57	.49
	k^*	-	-	9	31	57	85	113	137
$M = 16$	k'	109	161	181	209	215	165	215	221
	E_b/J_O'	15.29	11.90	12.44	12.33	11.91	11.90	11.39	12.48
	α_{wc}	1.0	.49	.66	.49	.59	.32	.42	.37
	k^*	-	-	-	-	5	9	13	19

Table 4.9: Optimum combinations of M'FSK modulation, (n, k') RS codes and diversity L'. E_b/J_O' and α_{wc} are for $P_b = 10^{-5}$.

n	M'	L'	k'	E_b/J_O'	α_{wc}
7	2	4	5	12.15	1.0
15	4	4	11	11.37	1.0
31	2	4	23	11.70	1.0
63	4	4	49	10.89	.74
127	2	4	99	11.49	1.0
255	4	3	179	10.47	1.0

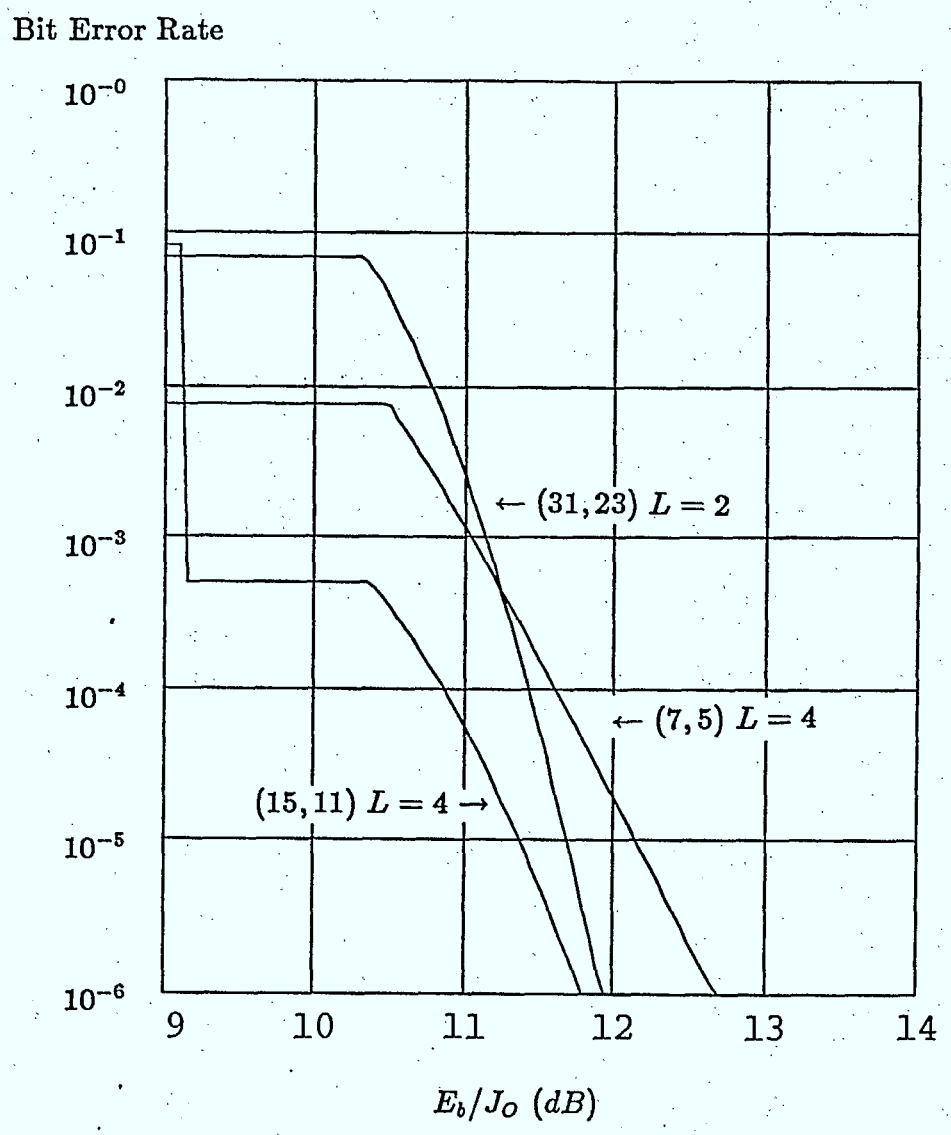


Figure 4.3: The BER performance of the optimum combinations of MFSK, diversity L and (n,k) RS codes for n equal to 7, 15 and 31, in worst case multitone jamming.

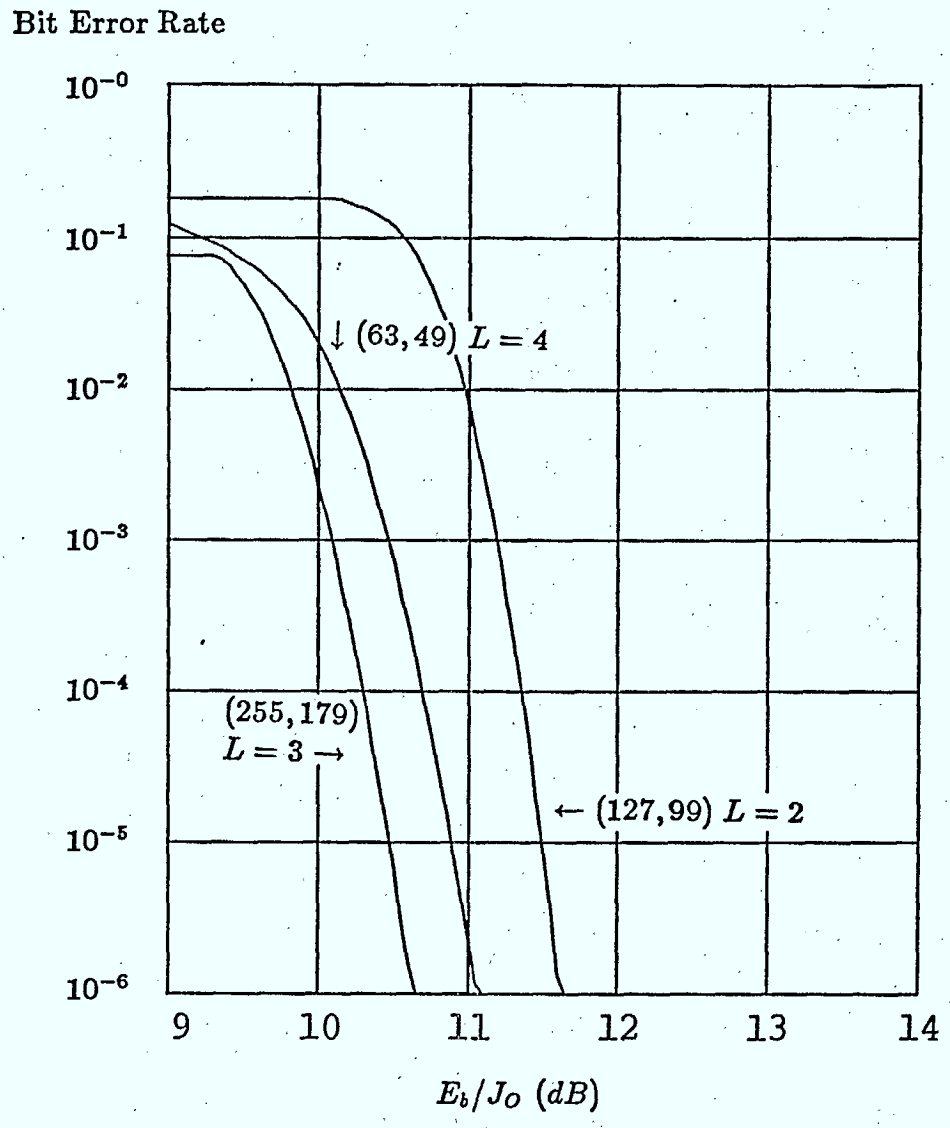


Figure 4.4: The BER performance of the optimum combinations of MFSK, diversity L and (n, k) RS codes for n equal to 63 and 255, in worst case multitone jamming.

Table 4.10: Combinations of MFSK modulation, (n, k) RS codes and diversity L for $P_b \leq 10^{-5}$ and overall code rate greater than 0.12.

n, M	L	5	6	7	8
15 (M=2)	k	9	11	-	-
31 (M=2)	k	19	23	-	-
63 (M=2)	k	39,41	47,49	-	-
63 (M=4)	k	-	-	-	31,33
127 (M=2)	k	77-83	93-101	107-111	-
255 (M=2)	k	153-171	185-205	215-225	-
255 (M=4)	k	-	-	109-113	123-137

instance, suppose $R_h = 20\,khop/s$ and $R_b = 2.4\,kbit/s$, then

$$r_{all} = \frac{R_b}{R_h} = 0.12.$$

We must find all possible combinations for which a k^* exists that will provide an overall code rate greater than or equal to 0.12. From Tables 4.3 through 4.8, we find that there are no such combinations for $n = 7$ and $M = 8$, 16 and 32. The only combinations are listed in Table 4.10. For the same reason as in the high r_{all} case, $M = 4$ is preferred, but now a large diversity factor with low rate RS codes are used.

4.5 Concluding Remarks

We have given a method for evaluating the performance of parallel error-erasure-decoding of RS codes in multitone jamming, using an exact method rather than a bounding technique. The worst case jamming parameter has been determined. We have shown that in worst case jamming, from the jammer's point of view, multitone jamming tends to be more effective than partial band noise jamming for nonbinary FSK. The optimum design of the system in worst case jamming was presented in terms of the combination of MFSK, diversity and RS coding for both high and low redundancy systems. However, these results

are informative rather than deterministic, since in the design of an actual system many other factors must be considered.

Chapter 5

Codec Implementation

5.1 Introduction

During the tenure of this contract, implementation work was done on the (24,12) Golay code (in quasi-cyclic form) and the (127,99) BCH Code. The Golay code was discussed in a previous report[15]. The (127,99) code design and implementation work is described in this report.

The (127,99) four error correcting BCH code was chosen as a a compromise between overall performance, given in Chapter 1, and implementation complexity.

5.2 The (127,99) BCH Code

An excellent description of BCH codes is given in [16]. This section gives the equations and steps that are relevant to the hardware implementation of the (127,99) BCH code. The following parameters apply:

number of bits/block:	$n = 2^m - 1 = 127, \quad m = 7$
number of information bits/block:	$k = 99$
number of parity bits:	$n - k \leq mt = 28$
number of correctable errors:	$t = 4$
minimum distance:	$d_{min} \leq 2t + 1 = 9$

The following polynomials over $GF(2)$ are used to define the code:

information polynomial:	$i(x) = i_0 + i_1 x + \ldots i_{98} x^{98}$
parity polynomial:	$p(x) = p_0 + p_1 x + \ldots p_{27} x^{27}$
code polynomial:	$c(x) = c_0 + c_1 x + \ldots c_{126} x^{126}$
received polynomial:	$r(x) = r_0 + r_1 x + \ldots c_{126} x^{126}$

The code under consideration is generated with the generator polynomial

$$g(x) = 1 + x^3 + x^4 + x^5 + x^7 + x^9 + x^{10} + x^{13} + x^{18} + x^{19} + x^{20} + x^{23} + x^{26} + x^{27} + x^{28}$$

This polynomial has roots $\alpha, \alpha^2, \ldots \alpha^8$ where α is the primitive element of $GF(2^7)$ generated by the primitive polynomial $m(x) = 1 + x^3 + x^7$.

5.2.1 Encoder Algorithm

The parity check polynomial is given by

$$p(x) = i(x)x^{28} \pmod{g(x)}$$

The codeword polynomial, in systematic form, is given by

$$c(x) = i(x)x^{28} + p(x)$$

5.2.2 Decoder Algorithm

Decoding is carried out in three steps:

1. Calculate syndromes S_1, S_3, S_5, S_7 using

$$S_i = r(\alpha^i)$$

2. Use syndromes from (1) to calculate the coefficients $\sigma_1, \sigma_2, \sigma_3, \sigma_4$ of the error-locator polynomial

$$\sigma(x) = 1 + \sigma_1 x + \sigma_2 x^2 + \sigma_3 x^3 + \sigma_4 x^4$$

3. Search for roots of $\sigma(x)$ to locate and correct errors in the received polynomial $r(x)$.

The decoder algorithm in more detailed pseudocode form is given in Fig. 5.1.

5.2.3 Hardware Overview

Many hardware configurations are possible depending upon the interface, cost, speed and power requirements. In this report we describe a low-cost implementation, with simple interfaces, based upon the use of 100 cell logic cell array ASICs from XILINX Inc.[17]. The design gives some consideration to speed, but is more concerned with a clean and simple implementation that uses a minimum number of chips, than with maximizing speed. Some pipelining and multiple buses are used to increase speed, but memory based lookup tables to perform multiplication and division are used to reduce complexity.

5.2.4 Encoder Implementation

The encoder, its interface signals, and relevant timing diagrams are shown in Fig. 5.2. All signals are synchronized with a common clock signal and all transitions occur on the + edge of the clock. The *ready-in* signal is asserted while the 99 serial input bits are applied. *Ready-out* is asserted by the encoder while the 127 code bits are available. At least one clock period must elapse between the output for one block and the input for the next sequential block. A schematic diagram for the encoder is given in Fig. 5.3. While *ready-in* is asserted, *data-in* is multiplexed to the output and sent to the feedback shift register which generates the parity bits. The 28 parity bits are multiplexed to the output following the 99 data bits. The 5-bit counter is used to count the the 28 parity bits.

1. Calculate syndromes S_1, S_3, S_5, S_7

   For i in 1,3,5,7 loop
   $S_i := r(\alpha^i)$;
   endloop;

2. Calculate $\sigma_1, \sigma_2, \sigma_3, \sigma_4$

   If $S_1 = 0$ then
   $\sigma_1 = S_1$; $\sigma_2 = S_5/S_3$; $\sigma_3 = S_3$; $\sigma_4 = \frac{S_3 S_7 + S_5^2}{S_3^2}$; $--x/o = 0$
   elseif $S_3(S_1^3 + S_3) + S_1(S_1^5 + S_5) = 0$ then
   $\sigma_1 = S_1$; $\sigma_2 = \frac{S_3 + S_1^3}{S_1}$; $\sigma_3 = 0$; $\sigma_4 = 0$;

   else

   $\sigma_1 = S_1$; $\sigma_2 = \frac{S_1(S_7 + S_1^7) + S_3(S_1^5 + S_5)}{S_3(S_1^3 + S_3) + S_1(S_1^5 + S_5)}$;
   $\sigma_3 = S_1^3 + S_3 + S_1\sigma_2$; $\sigma_4 = \frac{(S_5 + S_1^3 S_3) + (S_1^3 + S_3)\sigma_2}{S_1}$;
   endif;

3. Correct errors

   For i in 126..0 loop
   if $1 + \sigma_1(\alpha^i) + \sigma_2(\alpha^i)^2 + \sigma_3(\alpha^i)^3 + \sigma_4(\alpha^i)^4 = 0$, then
   $r_i = r_i + 1$;
   endif;
   endloop

Figure 5.1: Decoder Algorithm

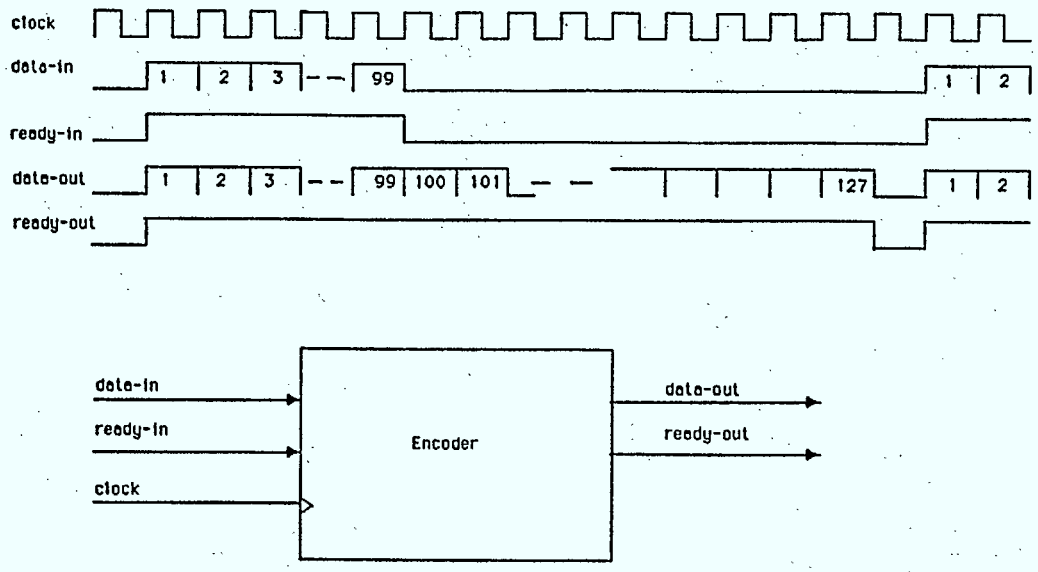


Figure 5.2: Encoder I/O and Timing

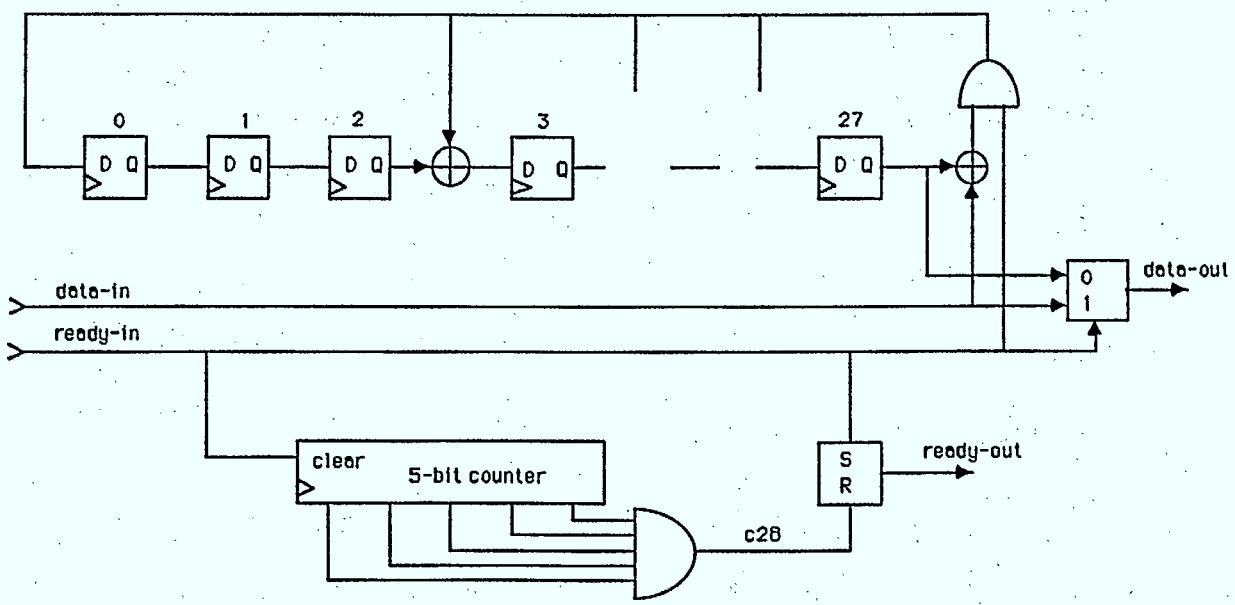


Figure 5.3: Encoder Schematic

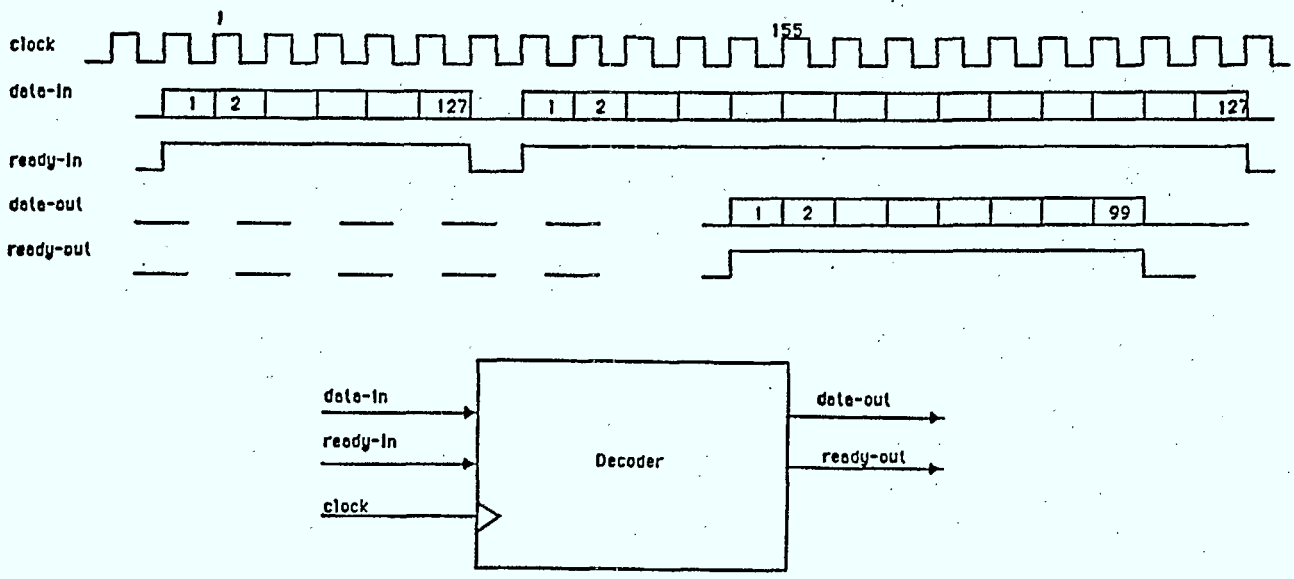


Figure 5.4: Decoder I/O and Timing

5.2.5 Decoder Implementation

The decoder terminal characteristics and timing are shown in Fig. 5.4. *Data-in* arrives in 127 bit blocks. A delay of 28 clock periods exists between the last bit of an input block and the first bit of the decoded output block. A new input block can be initiated one clock period after the last bit of the previous input block.

The decoder is partitioned into 3 stages plus a buffer as shown in Fig. 5.5. Syndrome components $S1, S2 = S1^2, S3, S5$ and $S7$ are calculated in stage 1 and passed to stage 2 under the control of several control signals. Stage 2 calculates the error locator polynomial coefficients $\sigma_1, \sigma_2, \sigma_3$ and σ_4 and sends them sequentially to stage 3 via the 7 signal lines, *sigmai*, and the four, *loadsi*, control signals. Stage 3 sequentially determines the roots of the error locator polynomial to correct data from the buffer as it is shifted to the serial *data-out* line. Fig. 5.6 is a functional partition and data flow diagram for stage 1. Schematic diagrams for each block are given in Figs. 5.7 to 5.10. Syndromes S1, S3, S5 and S7 are

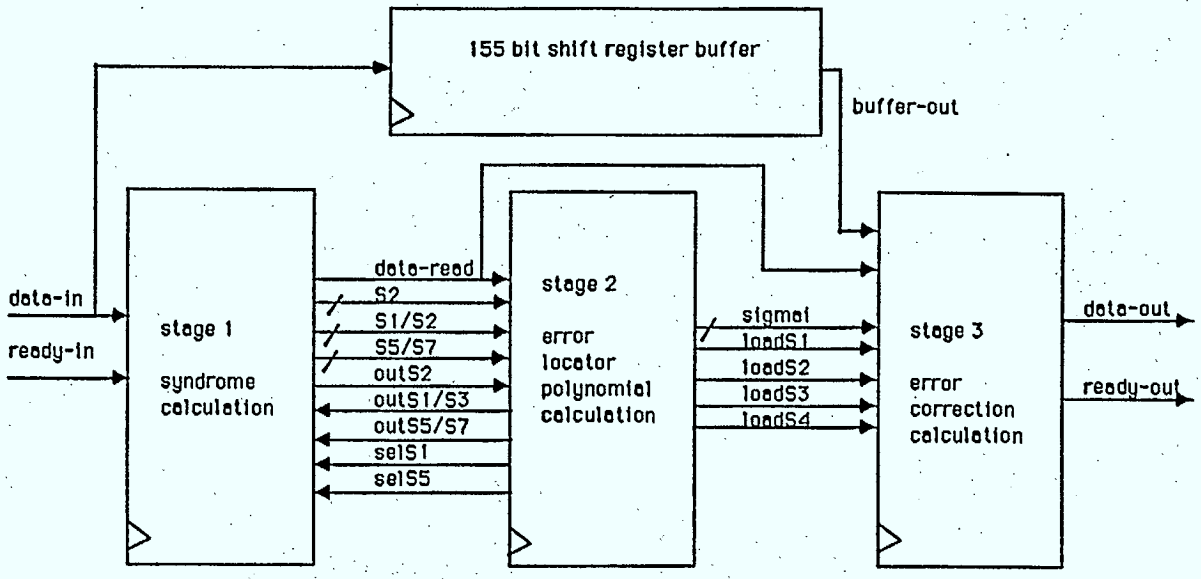


Figure 5.5: Decoder Block Diagram

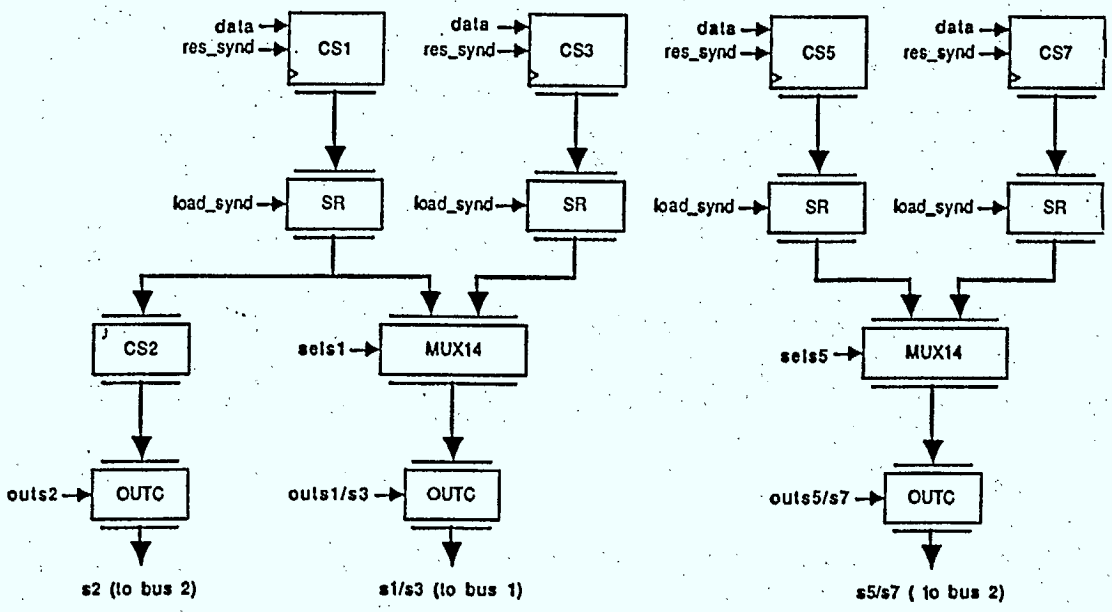


Figure 5.6: Stage 1 Functional Partition and Data Flow Diagram

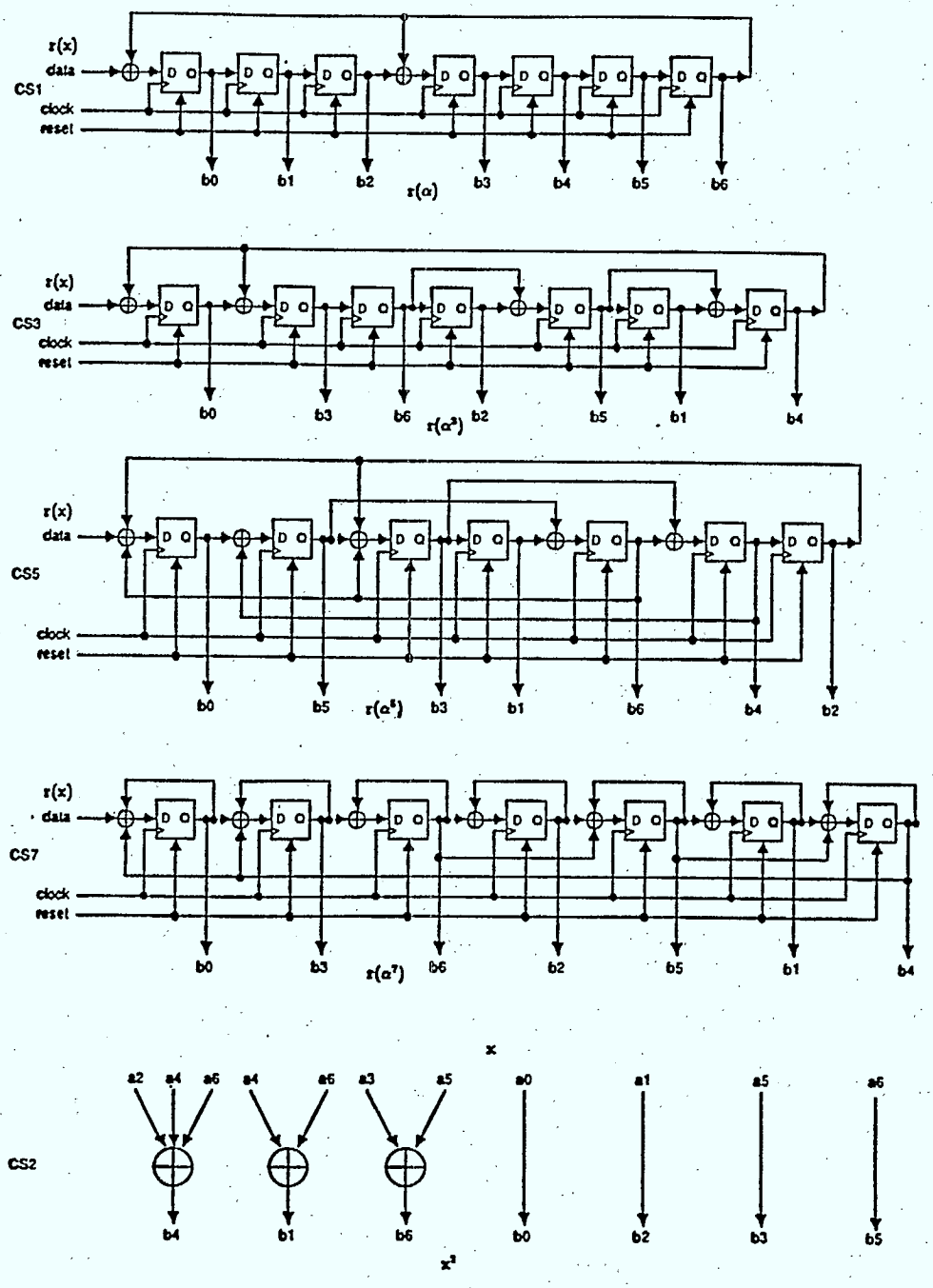


Figure 5.7: Detailed Schematic Diagrams I

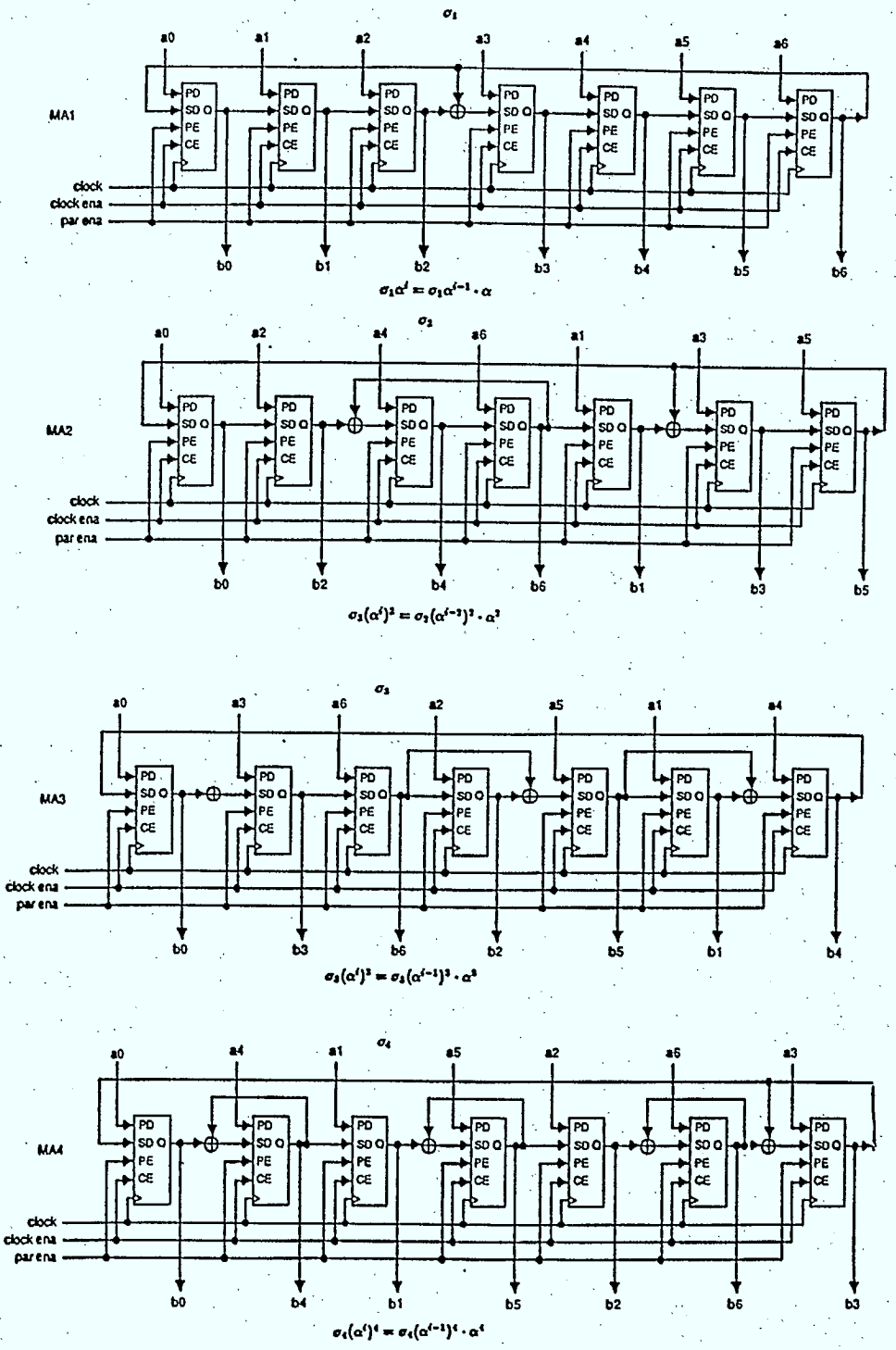


Figure 5.8: Detailed Schematic Diagrams II

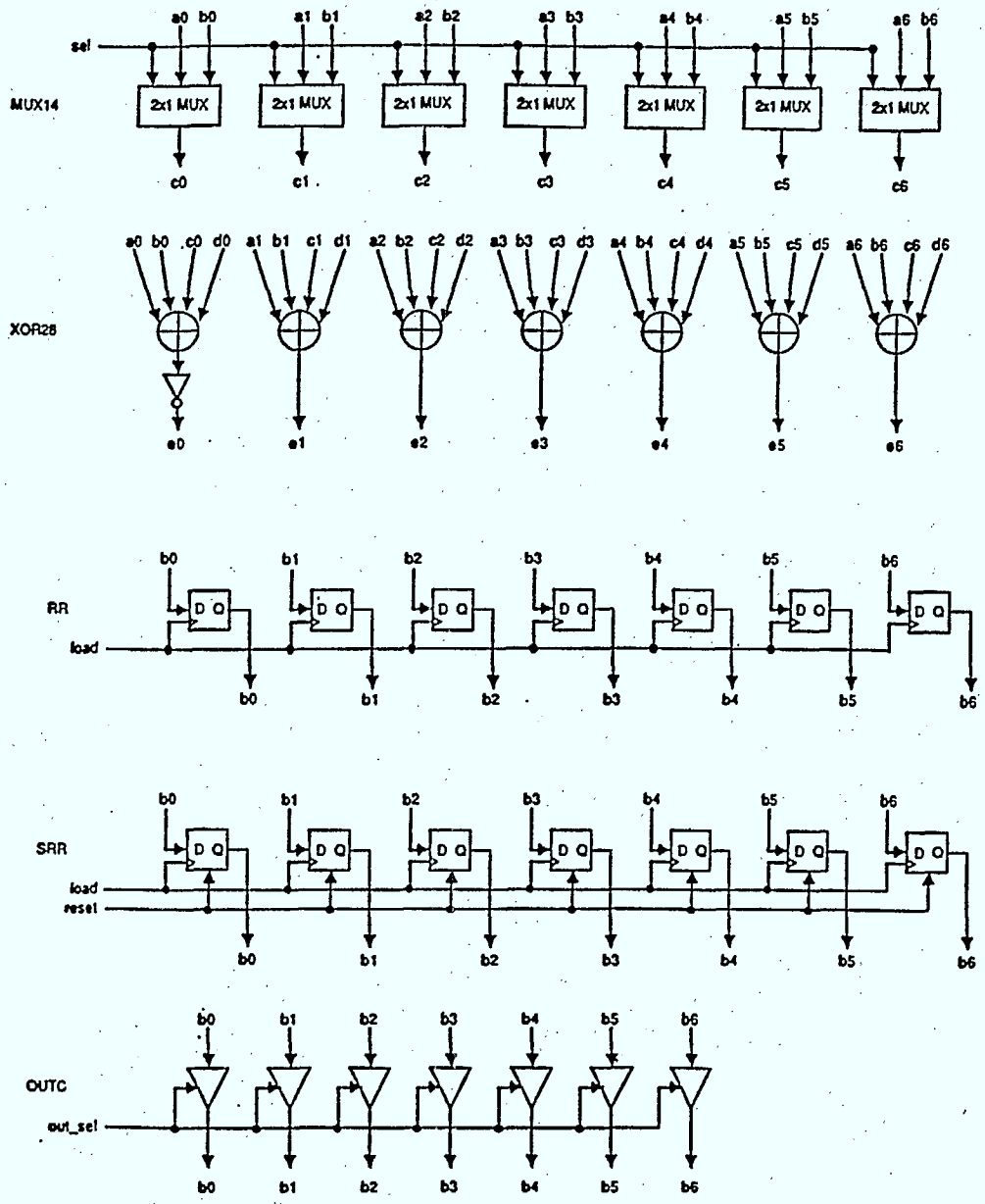


Figure 5.9: Detailed Schematic Diagrams III

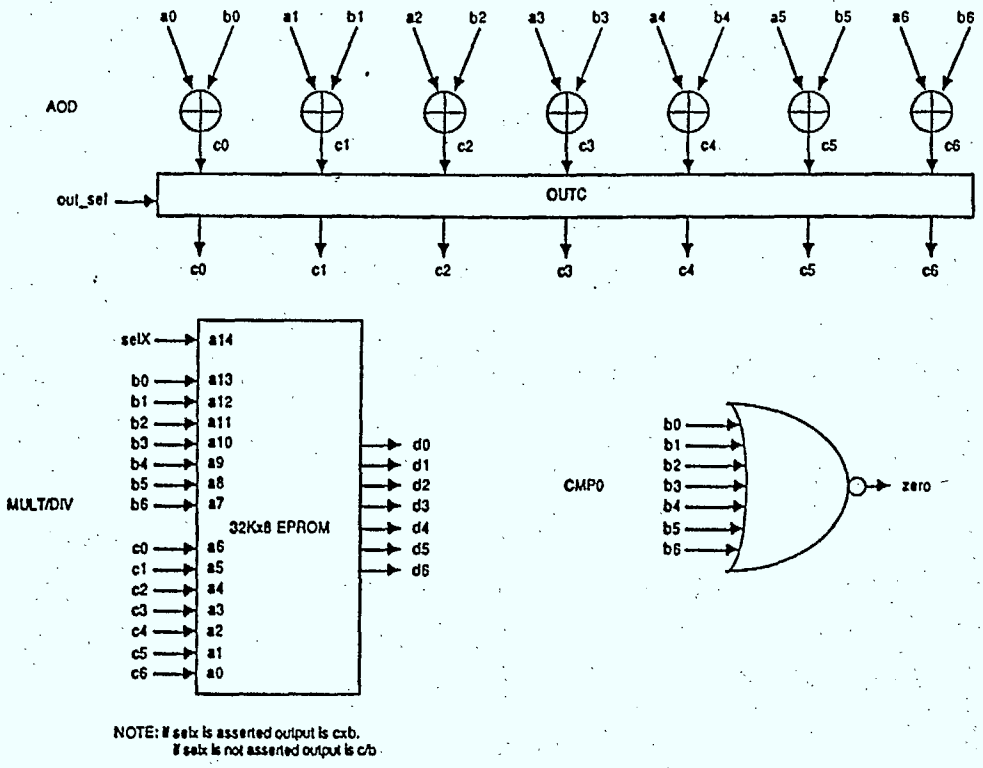


Figure 5.10: Detailed Schematic Diagrams IV

calculated in parallel by feedback shift registers CS1, CS3, CS5 and CS7 and then stored in four 7 bit holding registers. CS2 forms $S_2 = S_1^2$. Output of S1, S2, S3, S5 and S7 to stage 2 is under control of the control signals from stage 2.

A detailed flow diagram for generating the required control signals for stage 1 is given in Fig. 5.11. One flip-flop is required for the two states indicated in the diagram. Since asynchronous loading and resetting of the data flow registers is used, care is taken in the flow diagram to eliminate timing problems. Actual schematic diagrams for the control hardware are not included in this report.

A functional partition and data flow diagram for stage 2 is given in Fig. 5.12. All buses are 7 bits wide. Multiplication and division in $GF(2^7)$ is simply bit-wise exclusive OR. Registers $R1, R2, R3$ and $R4$ are temporary holding registers that were found necessary to carry out the stage 2 calculations.

A detailed flow diagram for stage 2 is given in Fig. 5.13. For clarity, the register transfer operations rather than the required control signals are shown in the data flow diagram. The diagram has 30 states and the longest path through the diagram is 24 clock periods. Thirty states can be realized with a 5-bit state machine and 24 clock periods is sufficiently fast that only one level of buffering is required for the calculations. The multiple data paths shown in the data flow diagram are used so that multiple operations can be performed concurrently to achieve the 24 clock period execution time for stage 2. The stage 3 functional partition and data flow diagram is given in Fig. 5.14. Each feedback shift register operates concurrently to evaluate a term of the error-correcting polynomial. The error-correcting polynomial is evaluated 99 times to correct the 99 information bits. The flow diagram that specifies the controller for stage 3 is given in Fig. 5.15.

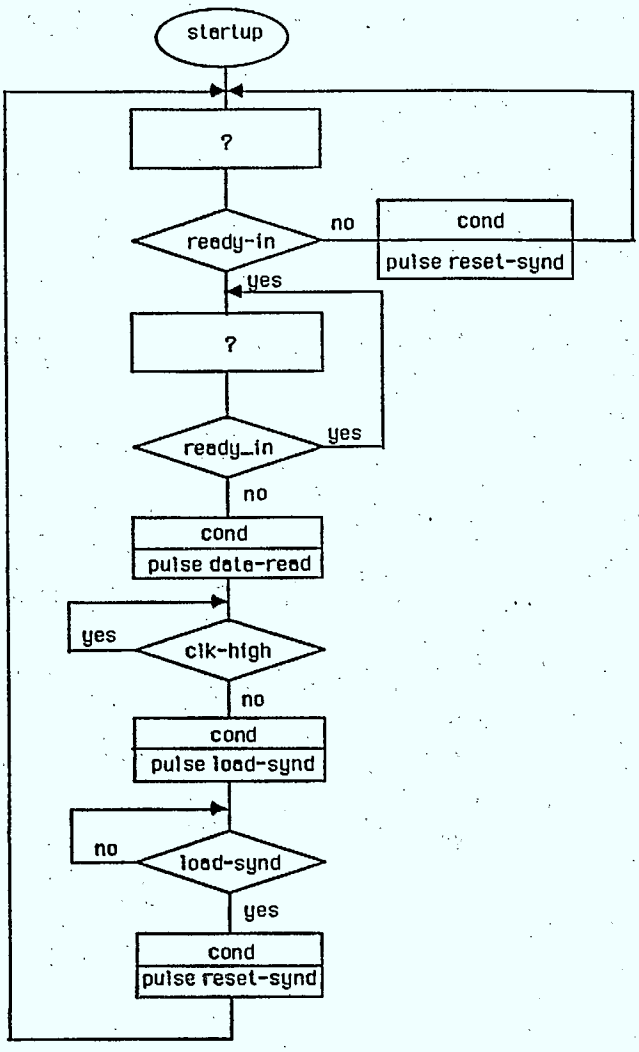


Figure 5.11: Detailed Flow Diagram for Stage 1

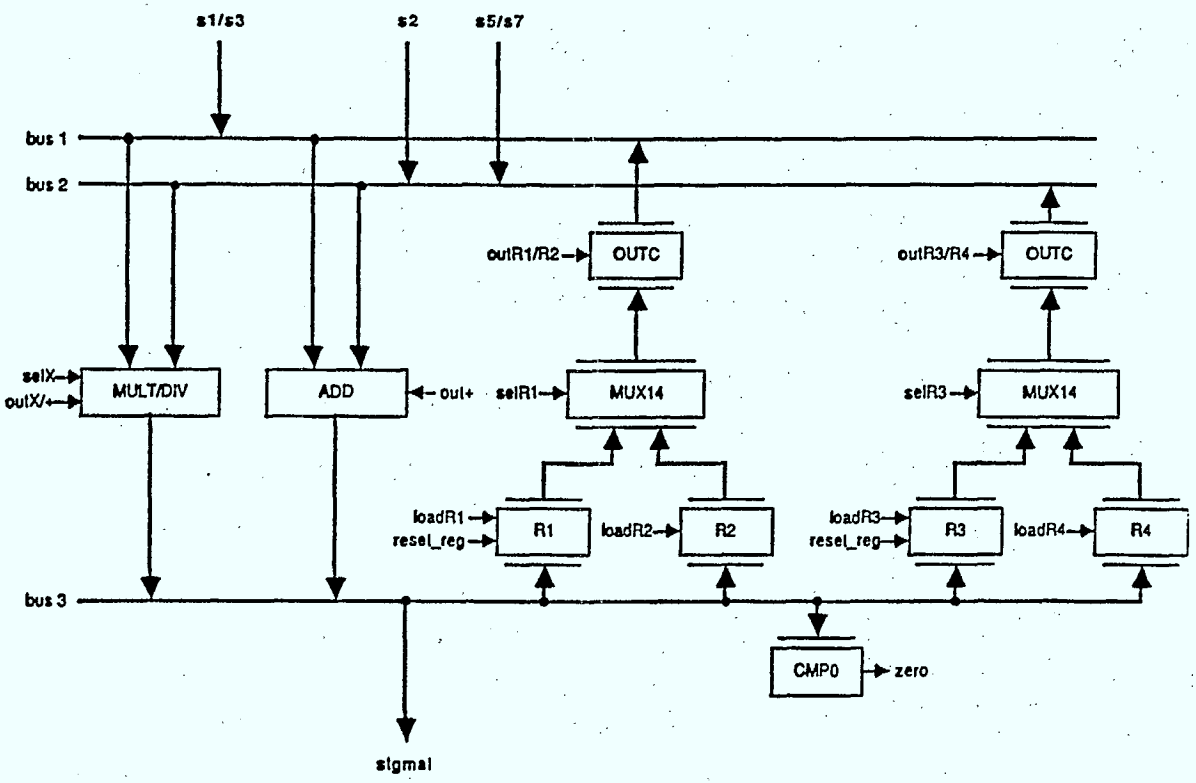


Figure 5.12: Stage 2 Functional Partition and Data Flow Diagram

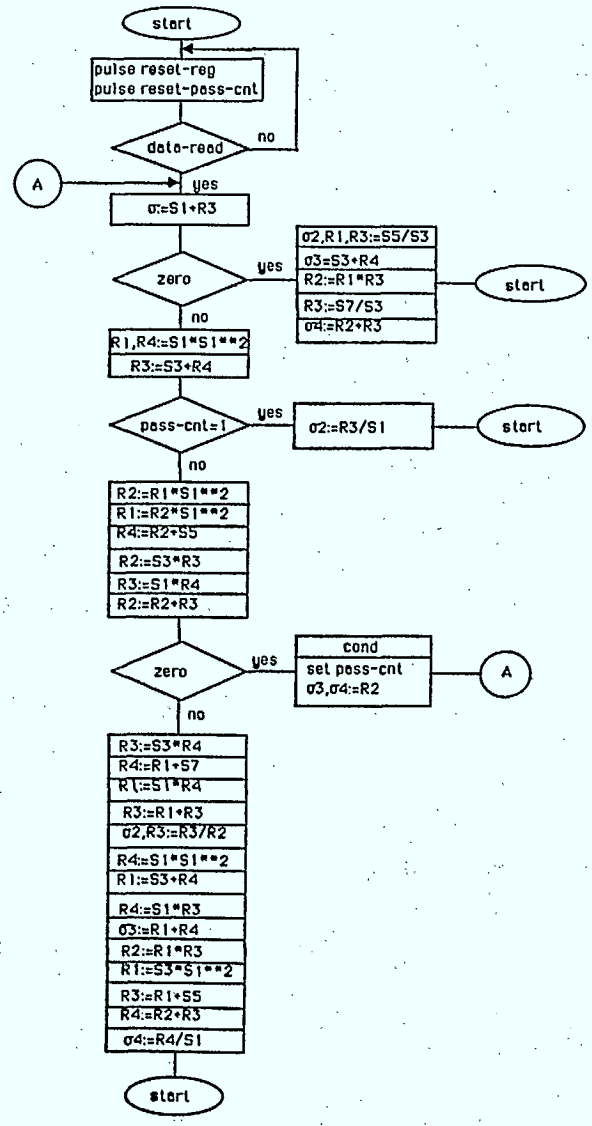


Figure 5.13: Detailed Flow Diagram for Stage 2

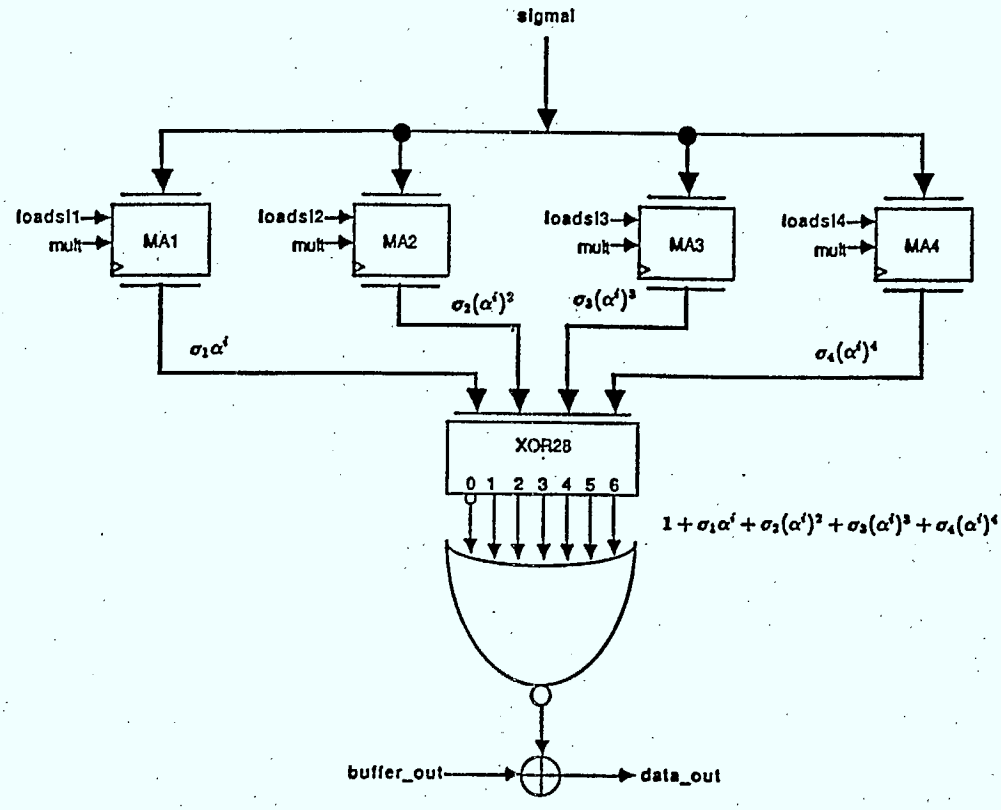


Figure 5.14: Stage 3 Functional Partition and Data Flow Diagram

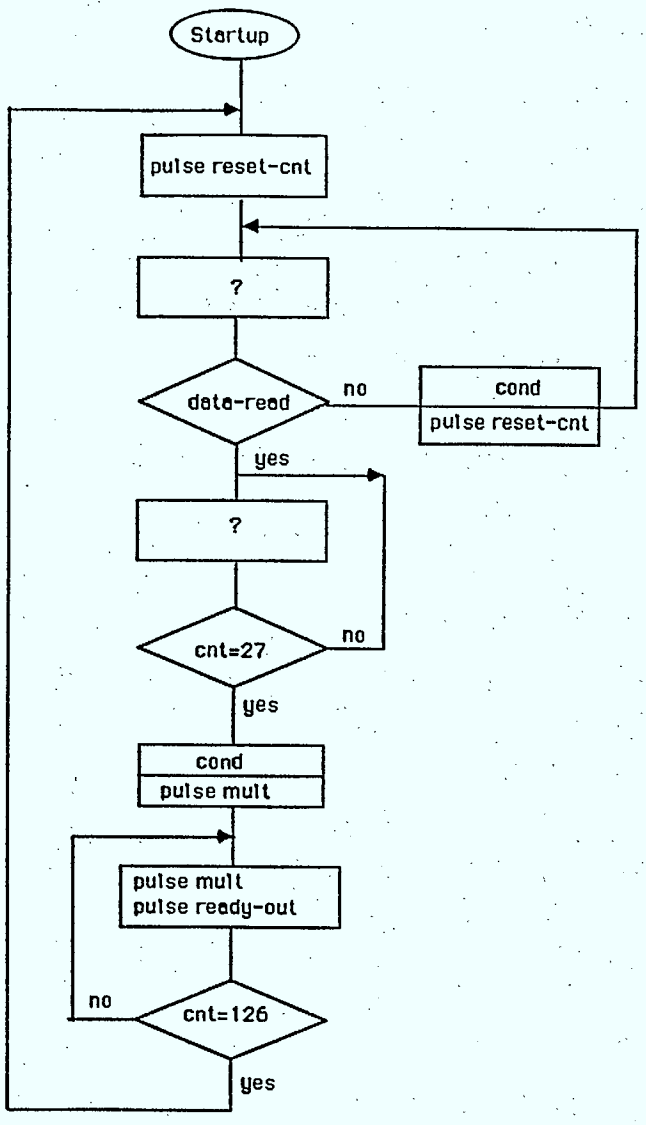


Figure 5.15: Detailed Flow Diagram for Stage 3

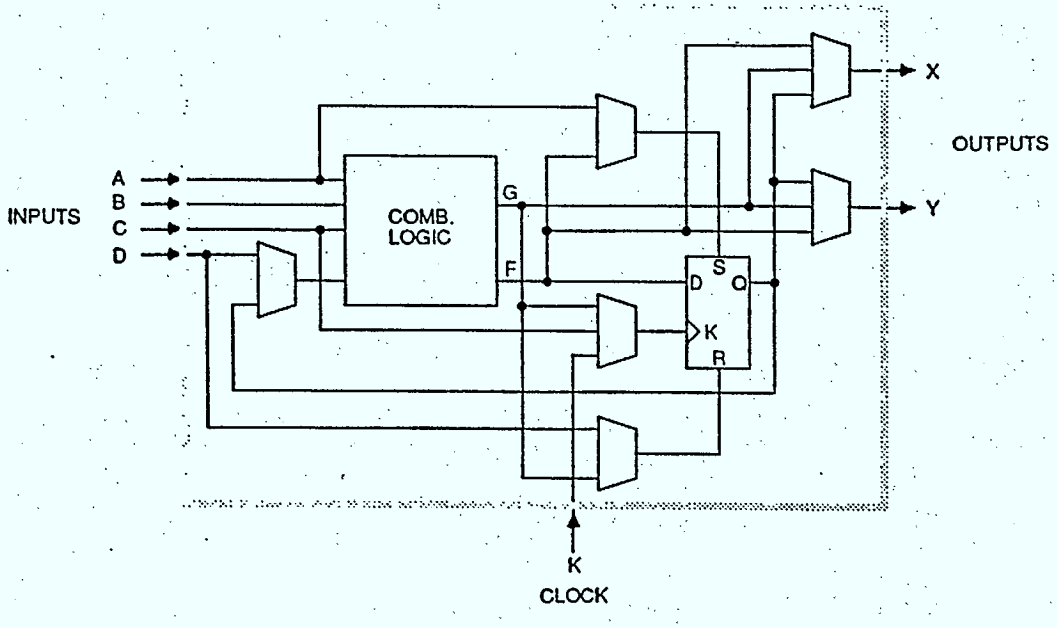
5.3 ASIC Implementation

The encoder and decoder described above were designed to be implemented with XC2018 logic cell array (LCA) ASIC chips from XILINX, Inc.[17]. These chips have 74 user programmable I/O blocks and 100 configurable logic blocks. The interconnections within and between the blocks are stored in on-chip RAM. The chips are low-power CMOS technology that can be clocked up to about 40 MHz.

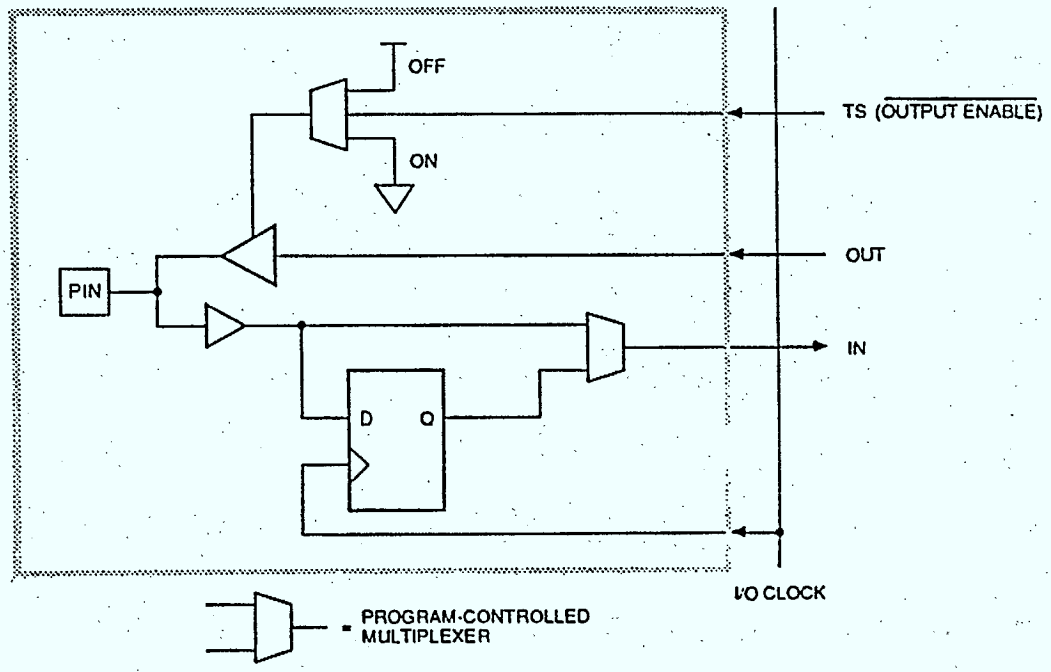
LCA programming is carried out on a personal computer using the XACT computer-aided design software package. The software package includes a design editor, P-Silos Simulator and an in-circuit emulator. An LCA I/O block and a configurable logic block are shown in Fig. 5.16. The I/O pins can be programmed as input, output or tri-state pins. Each I/O block also contains a D flip-flop that can be used to latch inputs or for general-purpose storage. The configurable logic blocks can be programmed to implement any function of 4 input variables or any 2 functions of 3 input variables. The multiplexors within a configurable logic block can be programmed to route signals within the block. The LCA also contains many programmable switches and interconnection buses to interconnect I/O blocks and configurable logic blocks.

The programming of a configurable logic block using XACT is illustrated by the screen image shown in Fig. 5.17. As shown the logic functions of the block can be specified algebraically or via a Karnaugh Map. User names can be specified for all the signals of a block.

The following LCA resources are estimated for the encoder and the 3 stages of the decoder. The resource estimates are well within the 74 I/O blocks and 100 configurable logic blocks of the XILINX XC2018.



Configurable Logic Block



I/O Block

Figure 5.16: Configurable Logic Block and I/O Block

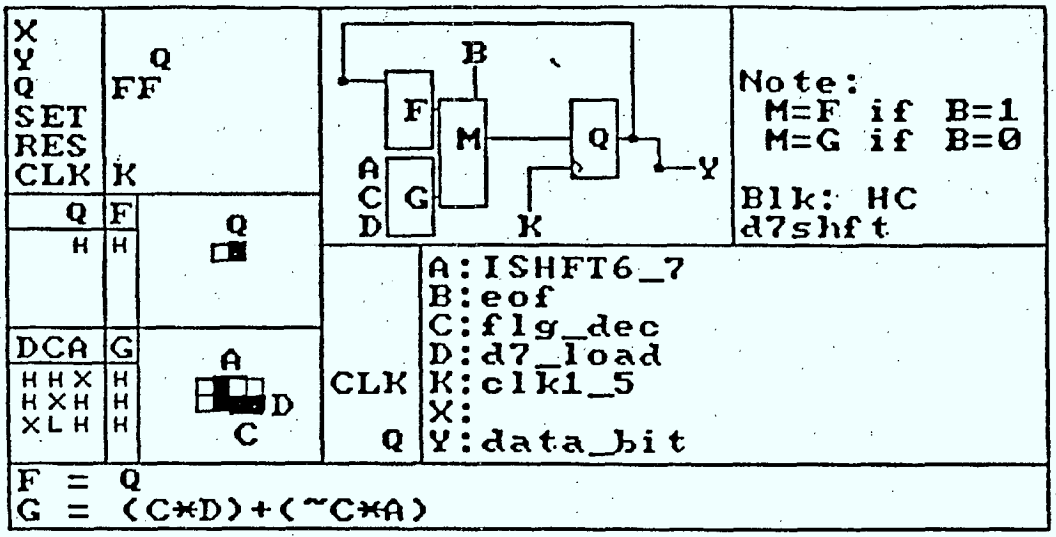


Figure 5.17: XACT CAD Display

	# of I/O Blocks	# of configurable logic blocks
encoder	5	37
stage 1	30	76
stage 2	61	61
stage 3	16	69

5.4 Concluding Remarks

The design of a (127,99) BCH codec suitable for ASIC implementation has been presented. The encoder can be realized with a single XC2018 ASIC whereas the decoder uses 5 chips; 3 XC2018 ASICs, 1 155 bit shift register possibly realized with another XC2018 and a 32K × 8 EPROM. Although actual implementation and testing is not complete, it is believed that a bit rate exceeding 5 Mbits/sec will be easy to achieve.

Implementation has been done using the XC2064 logic cell array system from XILINX. This application specific integrated circuit (ASIC) has 64 cells, uses low power CMOS

technology, and can be clocked up to 40 MHz/sec. For the codes considered, a single XC2064 chip was sufficient for the encoder. The Golay decoder used one XC2064 chip for processing and an EPROM for correction look-up. The (127,99) BCH decoder requires three XC2064 chips for processing and an EPROM for GF (2^7) multiplication by table look-up. The encoders that we have implemented can encode data at rates up to 40 Mbits/sec but the decoders are limited to 2 to 5 Mbits/sec because of the use of relatively slow EPROM memories.

New XILINX products and the use of normal basis representation for Galois Fields open some exciting possibilities for codec implementation. The new XILINX XC3090 has 5 times the number of cells as the XC2064 and about 10 times the number of useable storage elements. Using this new product with an on-board normal basis multiplier should enable the implementation of a single chip (127,99) BCH codec with a processing rate approaching 70 Mbits/sec. A similar performance should also be possible with a Reed-Solomon codecs.

We conclude from our work that XILINX logic cell arrays are very effective for codec implementation. They offer rapid prototyping, high performance, and reasonable cost for low to medium volume applications.

Chapter 6

Suggestions for Future Work

6.1 Coding for Slow Frequency Hopping Systems

The study of slow frequency hopping systems has been given little attention in past research, and we feel that this is the next logical phase of the investigation into coding for FH systems. In slow frequency hopping (SFH) a large number of bits of information are transmitted per hop. For example, with a 20 kbit/sec hop rate and a 1.5 Mbit/sec data rate, there are 75 bits/hop. There is no inherent diversity available as with fast frequency hopping (FFH). SFH signalling is vulnerable to jamming which can cause both long and short bursts of errors. The following sections present three methods of employing error correcting codes in a SFH system.

This proposed work presents a substantial amount of effort in analysis and would probably employ Monte-Carlo simulation.

6.1.1 Error Correcting Codes with Deep Interleaving

In order to use random error correcting codes directly, deep interleaving must be used. For example, using a BCH code with DPSK and C coded bits/hop, interleaving would have to be done to a depth greater than C to guarantee the independence of each bit in a codeword from the others, hence the randomness of errors in a codeword. Thus the cost

of interleaving and the decoding delay are serious concerns. Although this type of coded system may be costly in terms of interleaving cost and delay, it may perform well and is robust as far as the jamming pulse is concerned. Thus it may be used as a benchmark to compare with other coded systems and so is worth further study.

6.1.2 Long Error Correcting Codes to Correct Both Burst and Random Errors

In this technique, a long codeword is continuously transmitted over several independant hops without interleaving. If the jammed bits occur in bursts, (not longer than C), the code should be designed to correct them. However, if the error bursts are short and frequent, (restricted by the "coherent gain", a recently discovered phenomena associated with DPSK systems under jamming which is currently under investigation by us), then the burst error correcting code performance is degraded. Thus the task here is to design a combined code, (e.g., concatenated, parallel, etc.), so that both burst and random errors can be corrected efficiently. This type of coding is even more attractive if the jamming signal level is close to the level of system thermal noise, which causes random errors.

6.1.3 Diversity and Coding

In this method the same coded sequence is transmitted L times over L independant hops. At the receiver, diversity combination is first done using some voting algorithm, followed by the decoding. The basic drawback in this method is the large redundancy introduced by the artificial diversity and coding. This redundancy may be more effectively used if MDPSK were used with a large M. This is based on the fact that under severe jamming, a symbol would be correctly received if it is not jammed, even for large M, and incorrectly received if it is jammed, even for small M. Note that in AWGN, the performance degrades when M increases for MDPSK.

6.1.4 Key Techniques to be Considered

Burst Forecasting

This is described in [18] in conjunction with helical interleaving. This idea can be applied to the techniques described in 6.1.1 and 6.1.3 above. For the technique in 6.1.3, the voting results, (current and perhaps previous), in the diversity combination can be used to forecast burst errors.

Using Detected Differential Phase to Reduce the Effects of Errors

For the techniques in 6.1.1 and 6.1.2, this represents a means of generating erasures without external side information. However, the generation erasures in an optimum way may depend on the type of jamming and the signal to jammer power ratio. This may be more robust if a similar idea is used in conjunction with the technique in 6.1.3.

6.2 Normalized Envelope Detection

To reduce the effects of jamming in FFH/MFSK systems, the real envelope detector can be normalized by the sum of M envelope detector outputs over the M-ary band(M energy bands). Now the normalized output of a jammed hop tends to be smaller than that over a hop which is not jammed. The important feature of this technique is that it does not require external side information and seems to be robust against jamming.

The analysis of this scheme for $M > 2$ may be complex and will become even more so when used in conjunction with error control coding. Different types of codes such as hard decision block and convolutional codes, and soft decision convolutional codes will be analysed in conjunction with this scheme.

6.3 Implementation Aspects of Error Correcting Codes

Completion of the (127,99) BCH code Xilinx CODEC implementation will be followed by the evaluation of a commercial Reed-Solomon chip and a comparison with our Reed-Solomon CODEC design and implementation. A Galois Field Arithmetic processor will be designed if CODEC speed needs to be improved.

6.4 Other Directions

From the results of this report it is evident that the use of M-ary codes is important for FFH/MFSK systems. However, there are few results available for non-binary codes as compared to binary. Thus we propose a search for good M-ary convolutional codes ($M > 8$), and M-ary block codes ($M > 2$). In the case of block codes, the use of M-ary codes will reduce the error multiplication associated with alphabet conversion, as seen previously in the report.

For DPSK under jamming, as nonorthogonal signalling, dedicated codes are very scarce. Trellis codes are dedicated for a class of signals including DPSK in AWGN. However, the performance of the "set partition" principle may be poor under jamming. It may be possible to trade the bandwidth efficiency gained with these codes for power efficiency. This is important since bandwidth efficiency is secondary for SS systems.

Bibliography

[1] Viterbi, A.J. and Jacobs, I.M.: 'Advances in Coding and Modulation for Noncoherent Channels Affected by Fading, Partial Band, and Multiple Access Interference', *Advances in Communication Systems*, Vol. 4, Academic Press, New York, NY, 1975, pp 279-308.

[2] Bird, J.S. and Felstead, E.B.: 'Antijam Performance of Fast Frequency-Hopped M-ary NCFSK – An Overview", *IEEE J. on Selected Areas on Commun.*, Vol. SAC-4, No. 2, Mar. 1986, pp 216-233.

[3] Simon, M.K., Omura, J.K., Scholtz, R.A. and Levitt, B.K.: *Spread Spectrum Communications, Vol. II*, Computer Science Press, Rockville, MD, 1985.

[4] Ma, H.H. and Poole, M.A.: 'Error Correcting Codes Against the Worst-Case Partial-Band Jammer', *IEEE Trans. on Commun.*, Vol. COM-32, No. 2, Feb. 1984, pp 124-133.

[5] Trumpis, B.D.: 'Convolutional Coding for M-ary Channels', *Ph.D. Dissertation*, University of California, Los Angeles, 1975.

[6] Wang, Q., Gulliver, T.A., and Bhargava, V.K.: 'Coding for Fast Frequency Hopped *M*FSK Spread Spectrum Satellite Communications', *Technical Report EE 87-4*, Department of Electrical and Computer Engineering, University of Victoria, Victoria, B.C., Aug. 1987.

[7] Odenwalder, J.P.: 'Optimal Decoding of Convolutional Codes', *Ph.D. Dissertation*, University of California, Los Angeles, 1970.

[8] Viterbi, A.J. and Omura, J.K.: *Principles of Digital Communications and Coding*, McGraw-Hill, New York, NY, 1979.

[9] Ziemer, R.E. and Peterson, R.L.: *Digital Communications and Spread Spectrum Systems*, MacMillan, New York, NY, 1985.

[10] Torrieri, D.J.: 'The Information-Bit Error Rate for Block Codes', *IEEE Trans. on Commun.*, Vol. COM-32, No. 4, Apr. 1984, pp 474-476.

[11] Forney, D.G., Jr.: *Concatenated Codes*, MIT Press, Cambridge, MA, 1966.

[12] Blahut, R.E.: *Theory and Practice of Error Control Codes*, Addison-Wesley, 1983.

[13] Pursley, M.B. and Stark, W.E.: 'Performance of Reed-Solomon Coded Frequency-Hop Spread-Spectrum Communications in Partial-Band Interference', *IEEE Trans. on Commun.*, Vol. COM-33, No. 8, Aug. 1985, pp 767-774.

[14] Brunton, R.S.: 'A Comparative Analysis of Reed-Solomon Decoding Techniques', M.A.Sc. Thesis, University of Waterloo, Waterloo, Ontario, Canada, 1981.

[15] Wang, Q., Gulliver, T.A., Bhargava, V.K. and Little, W.D., *Coding for Frequency Hopped Spread Spectrum Communications*, Annual Report under DSS Contract No. 27ST.36001-6-3539, Apr. 1987.

[16] Lin, S. and Costello, D.J. Jr., *Error Control Coding*, Prentice-Hall, NJ, 1983.

[17] *The Programmable Gate Array Handbook*, XILINX Inc, 2069 Hamilton Ave., San Jose, CA, 1986.

[18] Wu, W.W., Haccoun, D., Peile, R. and Hirata, Y., 'Coding for satellite Communications', *IEEE JSAC*, Vol. SAC-5, No. 4, pp. 724-728, May 1987.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

University of Victoria, Department of Electrical and Computer Engineering, P.O. Box 1700 Victoria, BC, V8W 2Y2

2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable)

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C,R or U) in parentheses after the title.)

Coding for Frequency Hopped Spread Spectrum Satellite Communications (U)

4. AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)

Wang, Q., Gulliver, Aaron T., Bhargava, Vijay K., and Little, Warren D.

5. DATE OF PUBLICATION (month and year of publication of document)

April 1988

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)

135

6b. NO. OF REFS (total cited in document)

18

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Annual Technical Report

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

Communications Research Centre
P.O. Box 11490, Station "H"
Ottawa, Ontario K2H 8S2

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)

32A99

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

SSC #27ST.36001-6-3539

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

Technical Report No. ECE 88-2

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

- (X) Unlimited distribution
- ( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
- ( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
- ( ) Distribution limited to government departments and agencies; further distribution only as approved
- ( ) Distribution limited to defence departments; further distribution only as approved
- ( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availabilty (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

Unlimited

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

Performance of various types of error correcting codes is examined under both partial band noise jamming and multitone jamming using fast frequency hopping, noncoherent M-ary Frequency-shift-keying (NCMFSK) with optimum diversity and a fixed data rate.

A practical frequency hopping spread spectrum system may be limited to a fixed hop rate. Thus the performance of error correcting codes is examined under the same constraints as before but with a fixed hop rate. These results are compared to those with a fixed data rate.

A recently proposed system employing Reed-Solomon coding, and parallel error-erasure-correction decoding is studied under multitone jamming. An exact method, rather than a bounding technique is used for performance evaluation.

Hardware design of a CODEC based on the (127,99) four error-correcting BCH code is described. This code was chosen as a compromise between overall performance and implementation complexity.

The report concludes with some suggestions for future work. They are primarily directed towards the analysis of slow frequency hopping systems.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible, keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

- Frequency hopping
- Spread spectrum
- Error-correcting codes
- Erasure decoding
- Anti-jamming
- M-ary FSK modulation

DATE DUE
DATE DE RETOUR
