DATA RELIABILITY[1]

S. G. POWERS
Radiation
Melbourne, FL

Summary In this paper the problem of achieving reliable digital information transfer in the presence of data errors is addressed. The approach taken is to reject data which is suspected of being in error under the philosophy that it is better to miss data than to receive it incorrectly. To this end, error detection mechanisms are considered and their performance compared for a specific application. The mechanisms are thresholding, error detection coding, waveform error detection and feedback. It is shown that error detection coding is the most effective, followed by feedback, thresholding, then waveform error detection. The results are summarized in Tables I and II. These tables give the undetected word error rate and missed word rates for the techniques considered.

The application which originally inspired this work is the use of time-division multiplexing to transfer mission-critical data on the B-1 aircraft.

Introduction The realization of reliable information transfer is an ever present task of the digital communication engineer. Typical approaches to improving data reliability are to improve the signal-to-noise ratio (SNR) by increasing power or subsystem efficiency, use of source and/or error correction encoding, or a more efficient modulation/detection scheme. The approach addressed in this paper is characterized by a philosophy that it is better to not receive the information than to receive it in error. This is accomplished by utilizing error detection mechanisms to detect the occurrence (or suspected occurrence) of errors. The suspected data is discarded to improve the reliability of the data which is retained. The error detection mechanisms considered are thresholding, error detection coding, waveform error detection and feedback.

The performance improvements realized by these techniques are compared by example. The performance is measured in terms of the probability of an incorrect data word being accepted, P_e, and the probability of data being missed or rejected, p_m. These results are presented as a function of the channel bit error rate (BER) and various parameters which effect a tradeoff between p_e and p_m. In general, the error rate may be decreased at the expense of missing more data.

[1] This work was supported, in part, by the Air Force Avionics Laboratory under Contract No. F33615-71-C-1917.

No attempt is made to consider all possible error detection schemes. Rather, a few attractive candidates are selected which offer a broad range of performance. In the selection of these techniques, major emphasis has been placed on simplicity of implementation.

Thresholding. The first approach to be considered is thresholding. The principle is to use a threshold to measure the quality of bit decisions and isolate the data most likely to be in error. By varying the threshold, a tradeoff is effected between p_m and p_e. The threshold scheme considered utilizes a null zone detector. A null zone detector, as illustrated in Figure 1, makes one of three decisions, depending on the amplitude of its input at the decision time. If the input is above the threshold, th, a data ZERO decision is made; if below -th a data ONE decision is made; and if between -th and th a null decision is made. Any word containing a null zone decision is rejected.

A simple bit detector, illustrated in Figure 2, is used. The analysis p_e and p_m for various values of threshold are contained in the Appendix. The results for fourteen bit words are summarized in Tables I and II in terms of the fractional threshold, η, which is the threshold as a fraction of the input data level. From the tables it is seen that at a BER of 10^{-4}, with no null zone, a 1.4×10^{-3} error rate occurs. The error rate decreases to 2.7×10^{-4} with $\eta = 0.1$ and 3.4×10^{-5} at $\eta = 0.2$. At the same time the miss rate increases from 0, with no null zone, to 7.0×10^{-3} at η 0. 1, and $3.9 \text{ X } 10^{-2}$ at $\eta = 0.2$.

Error Detection Coding Channel encoding improves the data reliability by adding parity bits to overcome the interference at the expense of bandwidth or transmission rate. Common codes used for this purpose are block codes (e.g., simple parity check codes, Hamming codes, and BCH codes) and convolution codes. A particularly simple (though inefficient) form involves repetitive transmission of the data message a specified number of times. Coding can be used entirely for correction of errors or entirely for detection of errors, or a combination of both correction and detection. However, in terms of improving the undetected word error rate, error detection alone is more effective. This is clear when one realizes that for every error a code can correct, it can detect two. Furthermore, with a few exceptions, the implementation of error detection is far less complex than error correction. Only error detection and its implementation are considered here.

The application of error detection to enhance data reliability is best demonstrated with an example. For this example, consider the use of error detection coding to determine when data should be rejected. Assume for purposes of comparison that the data consists of fourteen bit words. The objective here is to avoid acceptance of code words which have been altered due to the effects of noise. It will be assumed that bit errors are made randomly and independently with probability p (which is c function of the SN R and detector implementation).

With no coding, any single error will cause one data word to be mistaken for another, resulting in a word error. The objective in choosing code words is to choose a set of words which all differ in at least N_d bit positions. Thus, N_d, or more, bit errors would be required to cause a word error. The codes considered are a (19,14) shortened single error correcting BCH[2] code and a (24,14) shortened double error correcting BCH code. The (19,14) code has $N_d = 3$ so that all double bit errors can be detected. The (24, 14) code has $N_d = 5$ so that all combinations of four random bit errors can be detected. These codes also have the ability to detect all bursts of errors of length (n-k) or less. The (19, 14) code can detect bursts of errors of up to five bits long and the (24,14) code bursts of up to ten bits long. Note that data is only missed when an error has been detected so that no words without errors are lost, minimizing the missed message rate. The analysis of the performance of these codes as well as the single parity check (15,14) code is contained in the Appendix. The results are presented in Tables I and II. The tables show the word error rate, p_e, and the missed word rate, p_m, corresponding to the uncoded BER for the three codes. Consider a BER of 10^{-4}. In order of increasing code length, the codes achieve word error rates of 2.7×10^{-6}, 2.3×10^{-9} and $8.9 \times 10\text{-}^{13}$. The corresponding miss probabilities are 2.4×10^{-3}, 8.4×10^{-3} and 4.3×10^{-2}. Each code requires a different bit rate to transmit the fourteen bits of data plus the parity check bits in the same amount of time. This will result in a SNR decrease from the uncoded case. This was accounted for in obtaining the data in the tables, assuming the detector of Figure 2 with no null zone ($\eta = 0$).

It was stated previously that error detection coding is simply implemented. The codes are generated with a feedback shift register. The technique for determining the feedback connections are obtained from any standard reference on coding. An encoder for the (31,26) BCH code is shown in Figure 3. It consists of five shift register stages and modulo-two adders (exclusive OR gates). The procedure for encoding is to shift the information word (26 bits) into the shift register, then open the Feedback path and shift five more times. The five bits which result are the parity check bits. The output is the code word consisting of the twenty-six information bits followed by the five parity check bits. To detect errors, the some feedback shift register is used. All thirty-one bits are shifted into the register. The register contents after thirty-one shifts will be zero if no errors have occurred. If the contents are not zero, then an error has been detected. For the shortened (19,14) code, only the nineteen-bitcode word is shifted into the feedback register and the contents tested for zero.

Waveform Error Detection There are several ways in which the waveform characteristics can be used to detect errors. Two are discussed here. First, there exists a number of signaling waveforms which have a structural restriction which will be violated

---

[2] BCH codes, named after their discoverers, Bose, Chaudhuri, and Hocquenghem, are, as a class, the most efficient algebraic block codes known for correcting random errors. The (19,14) code is not a true BCH code but is derived from the (31,26) BCH code by a process called shortening.

by errors. Indicative of these waveforms is bipolar NRZ. Bipolar NRZ, as indicated in Figure 4, is a three-level scheme whereby a data ZERO is represented by a zero level and a data ONE is represented by equal-magnitude pulses that alternate in polarity and are one bit period wide. Any single bit error will violate the alternating pulse property.

The second approach is to make sub-bit decisions and compare the decisions for illegal structure. For example, a matched filter detector for the NRZ-L waveform shown in Figure 4 has a well-known performance if perfect timing and additive white Gaussian noise are assumed. IF the detector was matched to only half-bit segments the SNR would be decreased by 3 dB. The two half-bit decisions for each bit can be compared to each other. If they are the same, the bit is accepted; if different, the bit is rejected.

Using half-bit decisions to detect errors in fourteen-bit words, just as for thresholding, the two approaches can be compared. The analysis is performed in the Appendix with the results shown in Tables I and II. The results indicate that half-bit decisions improve the word error rate by about an order of magnitude but results in a relatively large amount of missed data. Clearly, this is not as good as thresholding.

Error Detection Coding With Thresholding The techniques discussed so far can be combined to complement each other. The combination of error detection coding and thresholding will be investigated to demonstrate the obtainable performance and available tradeoffs. The approach considered subjects all received data decisions to a null zone test and rejects all words with null decisions. The data which passes is then checked with an error detection decoder. The BER after thresholding becomes the input BER for the error detection decoder. The missed word rote is the sum of pm due to thresholding and then coding at the threshold output BER. The results of the sections on coding and thresholding are, thus, directly applicable. Tables I and 11 present the results for comparison. The different rates required for the codes have been taken into account.

Feedback Still another means of achieving reliable data transfer is through the appropriate use of feedback. A straightforward approach to using feedback is as follows. The source transmits a data word and stores the same word. The receiver stores the received data word and also transmits the same word back to the source. The source compares the feedback word to its stored word. If they are the same, the source transmits an accept data command. If they differ, the word is retransmitted and the above procedure repeated. For data to be accepted, the data word, the feedback and the accept command must all be received correctly. This technique will be referred to as feedback verification (FV). Tables I and II give the performance of FV for a range of channel BER’s. The analysis for these results is performed in the Appendix. At a BER of 10^{-4} feedback verification can be seen to result in an undetected error rate of 2.4×10^{-7} and a missed data rate of 7.2×10^{-5}.

There are a number of ways of improving the performance of Feedback systems using the error detection mechanisms described. These combinations will not be considered here except to note that if the receiver has a means of detecting errors, it is necessary to feed back only a retransmission request, and only when an error has been detected. It can be recognized that this is the means of recovering the data discarded by the techniques discussed previously. Since the missed data rate is usually much less than one, the impact on transmission rate is negligible (the impact on complexity may be significant).

Conclusions An effort has been made in this paper to show alternatives and tradeoffs and allow the reader to draw conclusions relative to his application. As such, it is difficult to draw conclusions of a general nature. However, For the specific application considered throughout this paper, it was found that error detection coding is the most effective in terms of minimizing the undetected word error rate and the amount of lost data. Feedback was found to be the second most effective followed by thresholding and waveform error detection. The results are contained in Tables I and II.

Appendix

Analysis for Thresholding The null zone detector has been analyzed and its performance curves are given in Figures A-1 and A-2. The performance is expressed in terms of p, the undetected BER, and p_{mb}, the missed bit rate. In the analysis additive white Gaussian noise and perfect timing were assumed. The optimum single pole RC low-pass filter was used, the effects of intersymbol interference were taken into account. The results are presented as a function of the SNR (E_b/N_o) and the null zone threshold, η (as a fraction of the input signal level).

An n bit word is missed if one or more bits are missed. The probability of this event is

$$p_m = P\left\{\text{at least one bit miss}\right\}$$

$$= \sum_{i=1}^{n} \binom{n}{i} p_{mb}^{i} (1 - p_{mb})^{n-i}$$

$$\approx np_{mb} \text{ for } np_{mb} < .01.$$

An n bit word is accepted in error if one or more bit errors are made and no bits are missed. Thus,

$$p_e = P\left\{\text{at least one undetected bit error (u.b.e.) in } n \text{ bits and no missed bits}\right\}$$

$$= \sum_{i=1}^{n} (\#\text{ of ways to make } i \text{ u.b.e.'s})\ P\left\{\text{exactly } i \text{ u.b.e.'s}\right\}$$

$$\times P\left\{\text{no missed bits in } n - i \text{ bits}\right\}$$

$$\approx np \text{ for } np,\ np_{mb} < .01.$$

For the case considered in the text of this paper n = 14.

Analysis for Error Detecting Coding With no coding, any single error can cause one data word to be mistaken for another, resulting in a word error. For each word there are fourteen words which differ by only one bit, $\binom{14}{2} = 91$ words which differ by only two bits, and, in general, $\binom{14}{i}$ which differ by i bits, so that the probability of a word, p_e, is

$$p_e = \sum_{i=1}^{14} \binom{14}{i} p^i (1-p)^i = 1 - (1-p)^{14} \approx 14p \text{ for } 14p < .01$$

The simplest code is a single parity check code. This code has minimum distance two[3] so that two bit errors are required for a word error. The distance structure of the code words is symmetric, that is, each code word has the some number of code words at any given distance. Thus, the performance is readily computed once the distance structure is known. The number of words that is distance i from any other code word can be shown to be

$$N_i = \binom{n}{i} \quad i \text{ even}$$

$$= 0 \quad i \text{ odd}$$

where n is the word length (n = 15 for this code). The probability of exactly i errors is

$$p_i = p^i (1-p)^{n-i}$$

so that the probability of a word error, p_e, is

$$p = P\left\{\text{at least two undetected errors}\right\}$$

$$= \sum_{\substack{i=2 \\ i \text{ even}}}^{n} \binom{n}{i} P\left\{\text{exactly } i \text{ errors}\right\} = \sum_{\substack{i=2 \\ i \text{ even}}}^{15} \binom{15}{i} p^i (1-p)^{15-i}$$

$$\approx 105p^2 \text{ for } 15p < .01.$$

3 This is referred to as the Hamming distance and is the number of bit positions in which two code words are different.

A codeword will be missed whenever an error is detected. There are $\binom{n}{i}$ ways of making i errors and all will result in word errors for i even, so the rest are detected. Thus,

$$p_m = \sum_{\substack{i=1 \\ i \text{ odd}}}^{15} \binom{15}{i} p^i (1-p)^{15-i}$$

$$\approx 15p \text{ for } 15p < .01.$$

The BCH codes are symmetric also. Table A-1 gives the distance structure for the (19,14) and (24,14) codes. The performance is found just as for the (15,14) code. In general, for an (n k) code with minimum distance d, the probability of exactly i errors is

$$p_i = p^i (1-p)^{n-i}$$

so that the probability of a word error is

$$p_e = P\left\{\text{at least } N_d \text{ undetected errors}\right\}$$

$$= \sum_{i=N_d}^{n} N_i \, P\left\{\text{exactly } i \text{ errors}\right\} = \sum_{i=N_d}^{n} N_i \, p^i (1-p)^{n-i}$$

$$\approx N_d \, p^d \text{ for } np < .01.$$

A code word is missed whenever an error is detected. There are $\binom{n}{i}$ ways of making exactly i errors and only N_i will result in word errors so that the rest are detected. Thus,

$$p_m = \sum_{i=1}^{n} \left[\binom{n}{i} - N_i\right] p^i (1-p)^{n-i}$$

$$\approx np \text{ for } np < .01.$$

It would be unfair to compare the codes on the basis of the some BER because of the different rates required for each code. For instance, the (24, 14) code requ ires a 1.77 times higher bit rate than the uncoded channel to transmit the some fourteen information bits. To make a fair comparison, the detector curves of Figures A-1 and A-2 are used to determine the BER for each code at the appropriate SNR. Had a different detector been used, the results would be different.

Analysis for Waveform Error Detection To make an error, both halves must be in error; the bit is rejected if the half-bit decisions differ. Thus,

$$p = p_{1/2}^2$$

$$p_{mb} = 2p_{1/2} - 2p_{1/2}^2$$

where p is the undetected biterrorrate, $p_{1/2}$ is the BER of the half-bit detector and p_{mb} is the missed bit rate.

The word error rate and missed word rate are found as follows:

$$p_m = P\left\{\text{at least one missed bit decision in an } n \text{ bit word}\right\}$$

$$= \sum_{i=1}^{n} \binom{n}{i} p_{mb}^{\ i} \ (1 - p_{mb})^i$$

$$\approx np_{mb} \text{ for } np_{mb} < .01.$$

$$p_e = P\left\{\begin{matrix}\text{at least one undetected bit error (u.b.e.) in } n \text{ bits and no missed bits} \\ \text{in the rest}\end{matrix}\right\}$$

$$= \sum_{i=1}^{n} \binom{n}{i} p^i (1 - p)^{n-i} \left[1 - \sum_{k=1}^{n-i} \binom{n-i}{k} p_{mb}^{\ k} (1 - p_{mb})^{n-i-k}\right]$$

$$\approx np \text{ for } np, np_{mb} < .01.$$

Analysis for Feedback Verification An error will occur in an n bit data word if at least one bit error is made in the original transmission, the same errors are made during the feedback transmission and the data accept command is received correctly. The probability of this occurring, p_e, is given by

$$p_e = P\left\{\begin{matrix}\text{at least one} \\ \text{error at} \\ \text{the decoder}\end{matrix} \quad \text{and} \quad \begin{matrix}\text{same errors} \\ \text{at} \\ \text{verification}\end{matrix} \quad \text{and} \quad \begin{matrix}\text{data accept} \\ \text{command correctly} \\ \text{received}\end{matrix}\right\}$$

Assume that bit errors occur independently with probability p, in both directions and the data accept word is only one bit. Then

$$p_e = \sum_{i=1}^{n} \binom{n}{i} p^i (1 - p)^{n-i} \ p^i (1 - p)^{n-i} \ (1 - p)$$

$$\approx np^2 \text{ for } np < .01.$$

Data is received correctly if the decoder correctly receives all bits, all bits are correctly verified, and the accept command is received.

The probability of correct reception, p_c, is given by

$$p_c = P\left\{\begin{matrix}\text{all bits decoded}\\\text{correctly}\end{matrix} \text{ and } \begin{matrix}\text{all bits}\\\text{verified}\end{matrix} \text{ and } \begin{matrix}\text{accept command}\\\text{correctly received}\end{matrix}\right\}$$

$$= (1 - p)^{2n + 1}$$

Only three things can happen when a data word is sent; it is correctly received, an undetected error occurs or errors occur and are detected. These events are mutually exclusive and exhaustive so that the probability of missed data, p_m , is given by

$$p_m = 1 - p_c - p_e \approx 1 - (1 - p)^{2n + 1} - np^2 \approx (2n + 1)\, p \text{ for } 2np < .01.$$

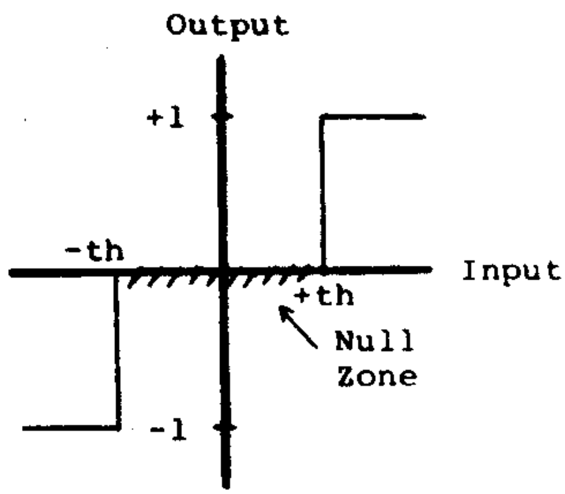

Figure 1. Null Zone Device

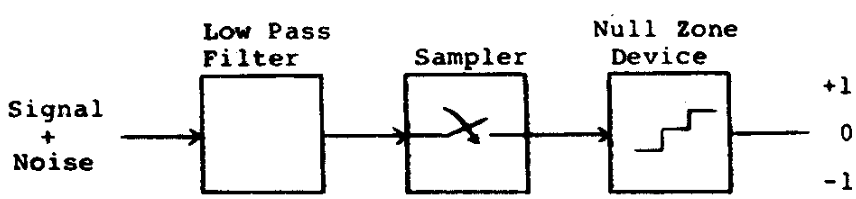

Figure 2. Simple Null Zone Detector

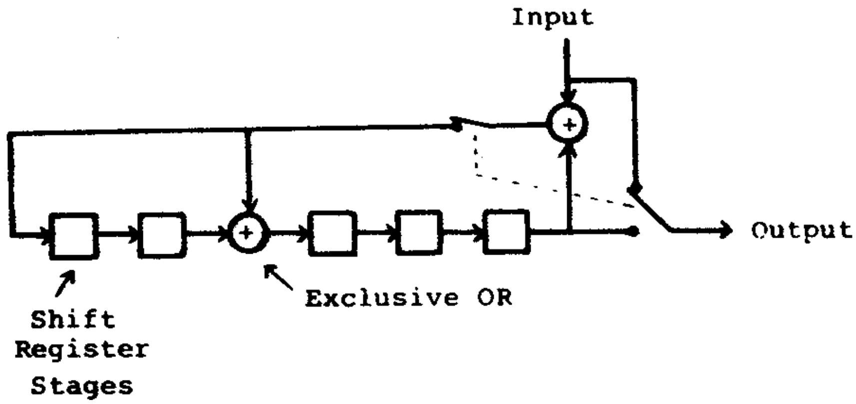

Figure 3. Feedback Shift Register Encoder/Error Detector For the (31,26) BCH Double Error Detecting Code

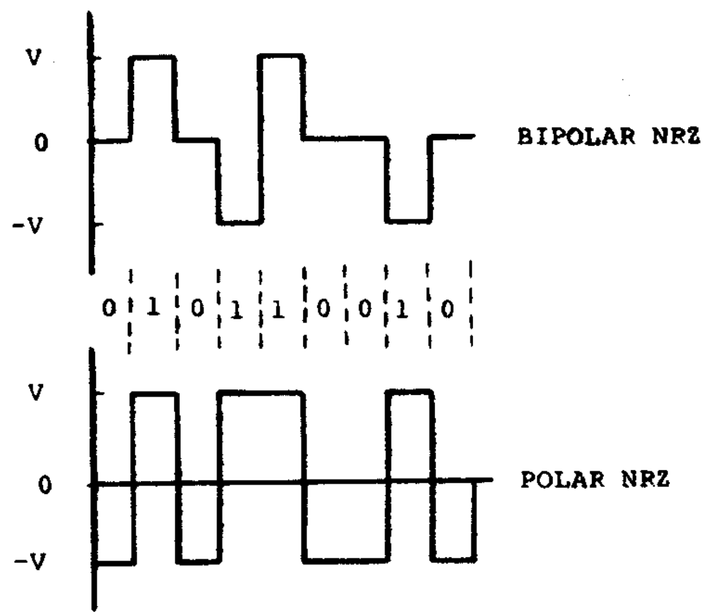

Figure 4. Waveforms Representing Typical Bipolar NRZ and Polar NRZ Signals.

Table I. Summary of Missed Word Rate for the Techniques Considered

BER	Thresholding			Error Detection Coding			Waveform Error Detection	Thresholding and Coding				Feedback Verification
								(15, 14)		(19, 14)		
	$\eta = 0$	$\eta = 0.1$	$\eta = 0.2$	(15, 14)	(19, 14)	(24, 14)		$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.1$	$\eta = 0.2$	
10^{-3}	0	3.8×10^{-2}	1.3×10^{-1}	2.0×10^{-2}	6.7×10^{-2}	1.9×10^{-1}	2.7×10^{-1}	4.1×10^{-2}	1.3×10^{-1}	1.4×10^{-1}	3.2×10^{-1}	2.8×10^{-2}
3×10^{-4}	0	1.5×10^{-2}	6.7×10^{-2}	6.6×10^{-3}	1.9×10^{-2}	8.6×10^{-2}	1.6×10^{-1}	1.2×10^{-2}	7.2×10^{-2}	5.7×10^{-2}	1.6×10^{-1}	8.7×10^{-3}
10^{-4}	0	7.0×10^{-3}	3.9×10^{-2}	2.4×10^{-3}	8.4×10^{-3}	4.3×10^{-2}	9.8×10^{-2}	7.8×10^{-3}	4.2×10^{-2}	3.0×10^{-2}	1.1×10^{-1}	2.9×10^{-4}
3×10^{-5}	0	2.5×10^{-3}	2.0×10^{-2}	7.8×10^{-4}	3.8×10^{-3}	2.1×10^{-2}	5.1×10^{-2}					8.7×10^{-5}
10^{-5}	0	9.1×10^{-4}	1.1×10^{-2}	2.7×10^{-4}	1.2×10^{-3}	1.1×10^{-2}	3.5×10^{-2}					2.9×10^{-6}
3×10^{-6}	0	4.1×10^{-4}	6.3×10^{-3}	8.7×10^{-5}	4.4×10^{-4}	5.0×10^{-3}	1.8×10^{-2}					8.7×10^{-7}
10^{-6}	0	2.0×10^{-4}	3.5×10^{-3}	3.3×10^{-5}	2.1×10^{-4}	2.4×10^{-3}	9.9×10^{-3}					2.9×10^{-8}
3×10^{-7}				9.6×10^{-6}	1.1×10^{-4}	1.2×10^{-3}	5.2×10^{-3}					8.7×10^{-9}
10^{-7}				3.9×10^{-6}	6.1×10^{-5}	6.0×10^{-4}	2.9×10^{-3}					2.9×10^{-10}
3×10^{-8}				1.1×10^{-6}	2.5×10^{-5}	3.1×10^{-4}	1.6×10^{-3}					8.7×10^{-11}
10^{-8}				4.5×10^{-7}	1.1×10^{-5}	1.5×10^{-4}	8.8×10^{-4}					2.9×10^{-12}

Table II. Summary of Undetected Word Error Rate for the Techniques Considered

BER	Thresholding			Error Detection Coding			Waveform Error Detection	Coding and Thresholding				Feedback Verification
								(15, 14)		(19, 14)		
	$\eta = 0$	$\eta = 0.1$	$\eta = 0.2$	(15, 14)	(19, 14)	(24, 14)		$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.1$	$\eta = 0.2$	
10^{-3}	1.4×10^{-2}	4.0×10^{-3}	9.5×10^{-4}	1.8×10^{-4}	1.2×10^{-6}	1.4×10^{-9}	2.0×10^{-3}	8.8×10^{-6}	4.9×10^{-8}	7.4×10^{-8}	2.5×10^{-9}	1.4×10^{-5}
3×10^{-4}	4.2×10^{-3}	9.2×10^{-4}	1.8×10^{-4}	2.0×10^{-5}	2.7×10^{-8}	2.8×10^{-11}	6.2×10^{-4}	4.6×10^{-7}	1.8×10^{-8}	1.5×10^{-10}	7.8×10^{-12}	1.3×10^{-6}
10^{-4}	1.4×10^{-3}	2.7×10^{-4}	3.4×10^{-5}	2.7×10^{-6}	2.3×10^{-9}	8.9×10^{-13}	2.1×10^{-4}	3.8×10^{-8}	6.0×10^{-10}	3.6×10^{-11}	2.5×10^{-13}	1.4×10^{-7}
3×10^{-5}	4.2×10^{-4}	5.6×10^{-5}	5.0×10^{-6}	2.8×10^{-7}	2.2×10^{-10}	2.2×10^{-14}	5.4×10^{-5}					1.3×10^{-8}
10^{-5}	1.4×10^{-4}	1.4×10^{-5}	9.8×10^{-7}	3.4×10^{-8}	7.1×10^{-12}	7.8×10^{-16}	2.6×10^{-5}					1.4×10^{-9}
3×10^{-6}	4.2×10^{-5}	3.2×10^{-6}	1.5×10^{-7}	3.5×10^{-9}	3.3×10^{-13}	1.9×10^{-17}	6.2×10^{-6}					1.3×10^{-10}
10^{-6}	1.4×10^{-5}	7.8×10^{-7}	3.5×10^{-8}	5.1×10^{-10}	3.5×10^{-14}	4.7×10^{-19}	1.8×10^{-6}					1.4×10^{-11}
3×10^{-7}	4.2×10^{-6}			4.3×10^{-11}	5.8×10^{-15}	1.5×10^{-20}	5.2×10^{-7}					1.3×10^{-12}
10^{-7}	1.4×10^{-6}			7.1×10^{-12}	8.8×10^{-16}	4.6×10^{-22}	1.6×10^{-7}					1.4×10^{-13}
3×10^{-8}	4.2×10^{-7}			5.1×10^{-13}	5.9×10^{-17}	1.7×10^{-23}	4.7×10^{-8}					1.3×10^{-14}
10^{-8}	1.4×10^{-7}			2.8×10^{-14}	5.8×10^{-18}	4.6×10^{-25}	1.6×10^{-8}					1.4×10^{-15}

Table A-1. Distance Structure of the (19, 14) and (24, 14) BCH Codes

Distance	N_1 (19,14)	N_1 (24,14)
0	1	1
1	0	0
2	0	0
3	29	0
4	109	0
5	286	47
6	671	144
7	1276	346
8	1943	750
9	2490	1226
10	2743	1824
11	2546	2522
12	1943	2732
13	1234	2368
14	677	1872
15	308	1310
16	100	729
17	22	326
18	5	128
19	1	46
20		12
21		1
22		0
23		0
24		0

Figure A-1. Probability of making a bit error with a null zone detector for several null zone thresholds.

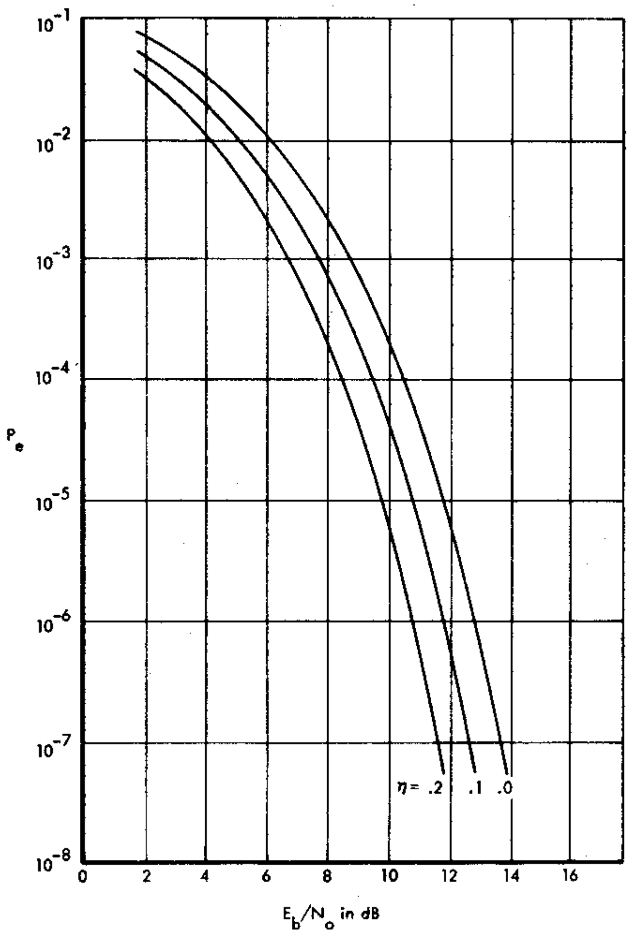

Figure A-2. Probability of missing a bit with a null zone detector for several null zone thresholds.