Michigan
N 56-2

MINIMUM-ERROR DEMODULATION OF

BINARY PCM SIGNALS

by

Earl F. Smith

1963

# MINIMUM-ERROR DEMODULATION OF BINARY PCM SIGNALS

by

Earl F. Smith

1963

## Abstract

There are at least two practical motivations for the development of theoretically optimum methods of performing an operation such as demodulation. The first is to determine, by calculation, the performance of the optimum system for comparison with that of existing or proposed practical systems, and the second is to determine the feasibility of implementing the optimum system or some approximation of it. The former is particularly useful in making decisions concerning the amount of effort which should be expended on improving existing systems, and the latter gives direction to that effort. The optimization must be with respect to some specified performance parameter.

The operation studied here is that of demodulation of noisy, binary, pulse-code-modulated waveforms. The performance parameter for which the operation is optimized is error-probability. The minimum-error demodulation operation is determined for waveforms with and without inter-bit dependence. It is assumed that this dependence results from statistical dependence between data samples represented by the PCM "words" or code groups, and that the demodulation decisions are made one-word-at-a-time, but utilizing n statistically dependent, received noisy words. The noise is assumed independent of the transmitted signal and additive and the number of words utilized, n, is arbitrary.

For the special case of no inter-bit dependence the minimum attainable error probabilities may be calculated directly for any signal-to-noise condition, assuming band-limited white gaussian noise. A comparison of these theoretical results with experimental results obtained independently in two different laboratories indicates that for bit-error probabilities lower than about 0.05 the minimum error demodulator offers no significant improvement over conventional demodulators if no inter-bit dependence exists.

For independent, additive, band-limited white gaussian noise, a method is developed for simulating with a digital computer the minimum-error demodulation with statistical dependence between data samples. Minimum error probabilities are then computed, by a monte-carlo method, for gaussian data and for $n = 2$ and $n = 1$. This computation is done for 3-bit and 6 bit code words. The results are applicable regardless of the waveforms used to represent the binary digits (or bits). These results indicate that for word-error probabilities less than about 0.1, no very significant power gains accrue from the use of statistical dependence in the data unless the correlation coefficients between data samples is large (i.e., 0.98 or greater) for a large number of transmitted samples. However, the results also indicate that the effect of using the statistical dependence in the data is to reduce errors in the high order (most significant) bits of the code. Therefore the error amplitude reduction may be considerable even if the reduction in error probability is not.

Possible implementations of the minimum-error demodulator as well as some simpler approximate implementations are discussed briefly.

# MINIMUM-ERROR DEMODULATION OF BINARY PCM SIGNALS

by

Earl F. Smith

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in the
University of Michigan

1963

Doctoral Committee:

Professor Lawrence L. Rauch, Chairman
Associate Professor Elmer G. Gilbert
Professor William L. Root
Professor Norman R. Scott

## Acknowledgements

The author would like to thank the members of his doctoral committee, Professors L. L. Rauch, E. G. Gilbert, W. L. Root, and N. R. Scott, for their patience and helpful criticisms during the course of this investigation. A special debt of gratitude is due Professor L. L. Rauch for his many helpful suggestions and fruitful discussions without which this investigation would not have been started, much less completed, and for his guidance and encouragement during difficult phases of the effort.

Special thanks also to Mr. D. D. McRae of Radiation Incorporated for the numerous discussions we have had which helped clarify concepts formulated during this study, to Dr. J. W. Young of Radiation Incorporated for his excellent help in formulating efficient logic for the computer programs, and to the University of Michigan Computing Center for assistance in utilizing their modern electronic computing facilities.

The cooperation of Radiation Incorporated in supporting a large part of the research effort is gratefully acknowledged.

Thanks are also extended to Mrs. Helen Wilkie for her excellent help in preparing and typing the manuscript.

## TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

## LIST OF TABLES

## LIST OF FIGURES

## INTRODUCTION

In demodulation, as in filtering and other operations on noise-contaminated signals, a question of prime concern when contemplating improvements in the operation is, "How much improvement is theoretically possible?" Some measure of the "goodness" of the operation (e. g., rms error, error probability, etc.) must be selected if a quantitative answer to this question is desired. For any selected measure of goodness (or performance parameter") there is generally a fundamental limitation on how well the operation can be performed with given signal power and noise conditions. Efforts to determine these limitations have led to the development of theories of optimum operations such as the Wiener filter theory (Reference 13) and theories regarding optimum detection of signals in noise (see, for example, Reference 3). Such theoretical treatments provide an optimum system (based on optimizing some performance parameter such as rms error or probability of error) whose performance may be calculated and compared with existing practical systems in order to determine the improvement theoretically possible.

In Rauch's report on improved demodulation (Reference 11), the maximum-likelihood demodulator is derived, assuming gaussian data and additive gaussian noise, for a large class of modulation operators (e. g., FM, PM, AM, PAM, PDM, etc. ). However, as pointed out in that report, the results are not applicable to PCM (Pulse Code Modulation) even when the distributions of the data and the noise random processes can be assumed gaussian. Yet it appears that the maximum-likelihood criterion for demodulation should be a very meaningful one for PCM communications since maximizing the probability of selecting correct transmitted signals at the receiver is equivalent to minimizing the probability of error.

This dissertation treats the PCM demodulation problem by deriving the minimum-error demodulator, assuming independent additive noise. The error probabilities (vs. signal-to-noise ratios) attained by this demodulator for band-limited white gaussian noise are then calculated. This is done first for completely random PCM signals (i. e., without considering data statistics) and then for gaussian data with dependence between data samples. We thus obtain the lowest possible error probabilities attainable under the assumed conditions, which we can compare with error probabilities obtained with existing or proposed practical demodulation schemes.

It should be noted that minimizing error probability does not in general result in minimizing other cost functionals or performance parameters such as the mean square error of the quantity represented by the code words. If error probability is not considered a satisfactory measure of goodness, then it is desirable to optimize the demodulation operation with respect to some performance parameter which is a satisfactory measure of goodness. Other performance parameters, or cost functionals, most often considered are statistical measures of error amplitude such as mean-square-error or mean-absolute-error. The statistical dependence between data samples may be viewed as redundancy in the data. Since this statistical dependence is a statistical constraint upon the relative amplitudes of the data samples it seems intuitively that the data redundancy could be used more effectively for reducing some statistical measure of error amplitude than for reducing the probability of error without regard to error amplitude. The results obtained in this dissertation indicate this to be true. The results also indicate that the minimum-error demodulation operation developed here will be more effective in reducing statistical measures of error amplitude than for reducing error probability even though the operation is optimized for the latter.

The interpretation of the meaning of a minimum-error-probability criterion for demodulation is not necessarily unique. It might be interpreted as the criterion of selecting the most probable sequence of data sample values over some interval of time including many data samples, or it might be interpreted as the criterion of selecting the most probable value for each sample in the sequence. These do not in general give the same result. The latter interpretation is used in this dissertation since selection of the most probable value for each sample, or PCM "word," yields the lowest possible word-error probability.

The theoretical optimum system may be impractical to construct and use, but with sufficient intuition a practical "near-optimum" system might be devised (possibly by modification of the optimization process) whose performance can then be compared with that of existing systems to determine the improvement afforded, if any, and compared with the theoretical optimum to determine whether further significant improvement is possible. Two such systems are briefly discussed in Chapter 6.

## Chapter 1

## MINIMUM-ERROR DEMODULATION OF RANDOM BINARY PCM WAVEFORMS

Let us first consider the minimum-error demodulation, or estimation, of binary PCM signals which have no dependence between bits, and which have been contaminated by independent additive band-limited white gaussian noise. The signal notation used is indicated in Figure 1. A binary PCM signal of duration T is divided into equal intervals of time, $T_B$, called "bit-times" during each of which the signal may assume either of two specified waveforms. The binary character of the transmitted signals need not be restricted to the use of two signal levels, but might be represented by two specified functions of time, one of which represents a "yes" bit and the other of which represents a "no" bit. For idealized PCM/AM, for example, the two specified functions of time representing "yes" bits and "no" bits are sinusoids of the same frequency and phase but of different amplitudes.

If we wish to minimize the probability of error in estimating transmitted signals we must select, for each received signal z(t), the transmitted signal, y(t), which maximizes the inverse conditional probability distribution of the transmitted signal, y(t), given the received signal, z(t):

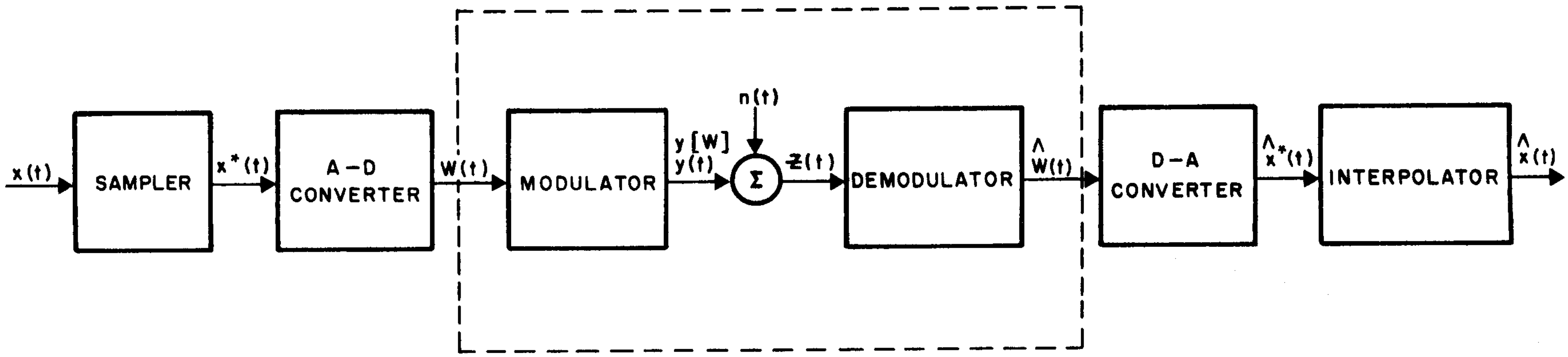
$$r(y \mid z) = \frac{p(z \mid y)\, f(y)}{g(z)}$$

where: $p(z \mid y)$ is the conditional probability density function of z(t) when y(t) is known.

$f(y)$ is the marginal probability distribution of the y(t) waveforms.

$g(z)$ is the marginal probability density function of the z(t) waveforms.

The probability density functions, $p(z \mid y)$ and $g(z)$ are interpreted as probability per unit volume of z-waveform space. We may consider the




35151

Figure 1 System Block Diagram

co-ordinate values in the waveform space to be the time samples of a band-limited waveform where the sampling rate is taken arbitrarily high but finite. Then for finite T the waveform space is of finite dimension. If dz is an infinitesimal hyper-volume in the waveform space, the relative probabilities of the z-waveforms will be $p(z \mid y)dz$ (a posteriori) or $g(z)dz$ (a priori). In the above equation the dz's of numerator and denominator have been omitted since they cancel each other.

It has been shown (see Reference 14, page 69) that for independent, additive, band-limited white gaussian noise of one-sided power spectral height $K^2$:

$$p(z \mid y) = K_1 \exp\left( - \frac{1}{K^2} \int_T (z(t) - y(t))^2 dt \right) \qquad (1.1)$$

where $K_1$ is a (normalizing) constant and the integration is over the interval of time, T, occupied by the signal. The term "band-limited white" here means that the power spectrum is of uniform height, $K^2$, from zero frequency to some arbitrarily large but finite frequency, W, and of zero height for all higher frequencies.

For the random signals being considered in this chapter, all of the possible binary PCM signals in the interval T are assumed to have equal a priori probabilities. That is, f(y) is the same for all possible y's. Then since g(z) is fixed for any specified z we may write

$$r(y \mid z) = K_2(z) \exp\left( - \frac{1}{K^2} \int_T (z(t) - y(t))^2 dt \right) \qquad (1.2)$$

where $K_2(z)$ is constant with respect to y.

Hence the most probable transmitted signal, y(t), is the one which gives the smallest value for the integral in (1.2). But if during any bit-time the PCM signal is one of two known waveforms, $f_1(t)$ or $f_2(t)$, then the y(t) which gives the smallest value for this integral must give the smallest value over each separate bit-interval, $T_B$. Therefore, the most probable y(t) can be determined one-bit-at-a-time (assuming synchronization) by selecting for each bit-time the transmitted signal, $y_B(t)$, (given received signal $z_B(t)$) which maximizes the inverse conditional probability function:

$$r(y_B \mid z_B) = K_3(z) \exp\left( - \frac{1}{K^2} \int_{T_B} (z_B(t) - y_B(t))^2 dt \right) \quad (1.3)$$

(We assume that the bandwidth, W, of the band-limited white noise is large compared to $1/T_B$.) Since such a selection has a greater probability of being correct than does any other method of selection, it has a lower probability of being in error than does any other method of selection. A device which makes such selection is therefore the minimum-error demodulator for random binary PCM waveforms with independent, additive, band-limited white gaussian noise.

It has been shown (see, for example, Reference 7) that when the binary character of the transmitted PCM signals is represented during each bit-time by either of two known waveforms, $f_1(t)$ and $f_2(t)$, the bit-error probability for such a device is:

$$P_E = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\beta\sqrt{1-\alpha}} e^{-x^2/2} dx \quad (1.4)$$

where $\beta^2$ is the ratio of average signal power to the noise power in a bandwidth equal to the bit rate or, equivalently, the ratio of signal energy per bit to noise power per unit bandwidth.

$$\beta^2 = \frac{1}{2K^2} \int_{T_B} (f_1(t)^2 + f_2(t)^2)\, dt \quad (1.5)$$

and

$$\alpha = \frac{1}{K^2 \beta^2} \int_{T_B} f_1(t) f_2(t)\, dt \quad (1.6)$$

If $f_1(t)$ and $f_2(t)$ have the same energy (i.e., $\int_{T_B} f_1(t)^2 dt = \int_{T_B} f_2(t)^2 dt$)

then eqn. (1.3) can be written:

$$r(y_B \mid z_B) = K_4(z) \exp\left(\frac{2}{K^2} \int_{T_B} z_B(t) y_B(t)\, dt\right) \qquad (1.7)$$

where $K_4(z)$ is constant with respect to $y_B$.

Hence the minimum-error demodulation for this case can be accomplished by correlating the received signal $z_B(t)$ during each bit-time with each of the two possible transmitted signals, $f_1(t)$ and $f_2(t)$, and selecting the transmitted signal which correlates best with $z_B(t)$.

For the case where the binary character of the transmitted PCM signals is represented by two signal levels, e.g., $f_1(t) = +S$ and $f_2(t) = -S$, the bit-by-bit correlation detection is equivalent to filtering the received waveform with an aperture filter of aperture $T_B$ and sampling its output at the end of each bit-time. If the sample is greater than zero the corresponding bit of the transmitted signal is assumed to have been $+S$, and if the sample is less than zero the corresponding bit is assumed to have been $-S$.

The minimum bit-error probabilities can be obtained for random binary PCM using the waveforms $f_1(t)$ and $f_2(t)$ corresponding to idealized PCM/AM, PCM/PM and PCM/FM. We assume that there are many cycles of the carrier occurring in one bit-time. That is $T_B \gg 2\pi/\omega_c$. For 100% modulated PCM/AM we have $f_1(t) = 2\,S \sin(\omega_c t + \phi)$ and $f_2(t) = 0$ ($S^2$ = average transmitted signal power). And from (1.5) and (1.6) we get (if $T_B \gg 2\pi/\omega_c$)

$$\beta = \frac{S}{K\sqrt{B}} \qquad \text{where } B = 1/T_B = \text{bit rate}$$

and $$\alpha = 0$$

and from (1.4)

$$\left(P_E\right)_{PCM/AM} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\beta} e^{-x^2/2}\, dx \qquad (1.8)$$

For PCM/PM we have $f_1(t) = \sqrt{2}S \sin(\omega_c t + \phi_1)$ and $f_2(t) = \sqrt{2}S \sin(\omega_c t + \phi_2)$, and therefore, if $T_B \gg 2\pi/\omega_c$

$$\beta = S/K\sqrt{B}$$
$$\alpha = \cos(\phi_1 - \phi_2)$$

The optimum value of phase deviation, $\phi_1 - \phi_2$, is 180 degrees since this gives the lowest possible value (i.e. - 1) for $\alpha$ and consequently the lowest possible value for $P_E$. Binary phase modulation with total phase deviation of 180 degrees is called "phase-shift-keying" (PSK) and is exactly equivalent to suppressed carrier PCM/AM. The minimum bit error attainable for this type of modulation is, from (1.4):

$$\left(P_E\right)_{PCM/PSK} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\sqrt{2}\beta} e^{-x^2/2}\, dx \qquad (1.9)$$

For PCM/FM it is often erroneously presumed that a higher deviation ratio will result in a lower error probability. This conclusion is usually based on the fact that the (filtered) video signal-to-noise ratio resulting from reception of PCM/FM by a receiver with conventional discriminator can be either calculated or observed on an oscilloscope to be higher for higher deviation ratio, and hence the conclusion that the error probability should decrease for higher deviation ratio. The invalidity of the conclusion is, of course, due to the implicit assumption that the video noise amplitude distribution is independent of deviation. (This is discussed further in Appendix III.)

That the bit-error probability cannot be made arbitrarily small by any means other than increasing the signal power (or decreasing system noise) is apparent from (1.4) since the minimum value for $\alpha$ is -1. Hence the error probability for PCM/FM or any other binary PCM transmission system can never be less than that for PCM/PSK (or suppressed carrier PCM/AM) for which $\alpha = -1$. What then is the minimum error probability attainable for PCM/FM? For this case we have

$$f_1(t) = \sqrt{2}S \sin(\omega_1 t + \phi_1)$$

$$f_2(t) = \sqrt{2}S \sin(\omega_2 t + \phi_2)$$

and $\alpha$ is a function of $\omega_1$, $\omega_2$, $\phi_1$, and $\phi_2$. PCM/FM is treated in Appendix I where it is found that for conventional PCM/FM, in which a single oscillator is frequency modulated by the PCM waveform, the optimum (minimum error probability) deviation ratio, D (D = ratio of total deviation to bit rate) is 0.715. This is in very good agreement with optimum deviation ratios determined experimentally (References 15 and 16) using conventional PCM/FM receiving equipment. It is also interesting to note that, due to previous heuristic considerations and experimental results, deviation ratios in the vicinity of 0.7 have previously been recommended for PCM/FM.

From Appendix I equation (I-4), the corresponding bit-error probability is:

$$\left(P_E\right)_{\substack{PCM/FM \\ opt}} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-1.1\beta} e^{-x^2/2} \, dk \qquad (1.10)$$

The treatment in Appendix I derives bit-error probability for PCM/FM directly in terms of deviation ratio rather than using the parameter $\alpha$, since deviation ratio is a more familiar parameter. The value of $\alpha$ for the above case is, however, 0.21, which when substituted in (1.4) gives the same result.

The minimum-error probability (corresponding to optimum deviation ratio) of (1.10) is plotted as a function of $\beta$ in Figure 2 along with some experimentally measured results[1] from References 15 and 16. The

---

1 A fourth measured curve, using phase-lock loop and D = .75, reported in Reference 15, crosses the theoretical optimum curve and indicates about 1 db better than optimum at $P_E = 10^{-6}$.

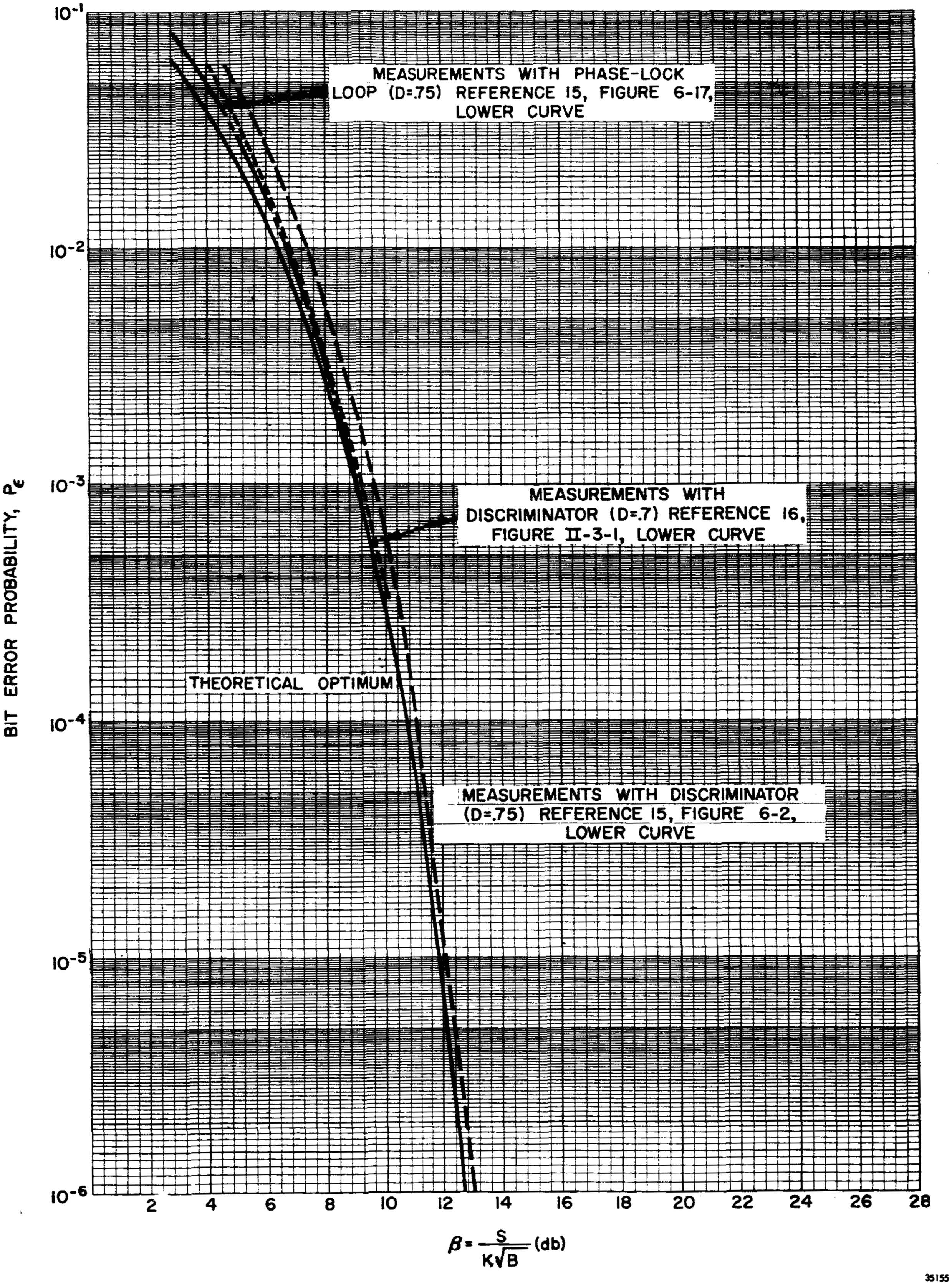



Figure 2 Bit Error Probability vs $\beta$ For PCM/FM

theoretical and experimental results are seen to be in very good agreement, which indicates that if we do not make use of the a priori statistics of the data (i. e., if we assume no dependence between bits) we cannot devise demodulaticn schemes for PCM/FM which will reduce the bit-error probability appreciably below that obtainable with conventional PCM/FM receiving equipment with good synchronization. This remark applies only for bit-error probabilities less than about .05 since no experimental results were obtained for higher bit-error probabilities.

Some conclusions regarding PCM/FM and PCM/AM may be drawn from the analytical results. By comparison of equations (1.8) and (1.10) we see that, for a given noise spectrum and bit rate, PCM/AM requires 0.8 db more power to accomplish a given bit-error probability than does PCM/FM (both using "optimum" demodulation). Also, by comparison of equations (1. 10) and (1. 9) we see that PCM/FM requires 2. 2 db more power than PCM/PSK (or suppressed carrier PCM/AM) for the same bit-error probability.

For "switched PCM/FM," i. e. PCM/FM resulting from non-synchronous switching between two oscillators of frequencies $\omega_1$ and $\omega_2$, we note from Appendix I that the lowest error probability is obtained with $\omega_2 - \omega_1 = 2\pi n B$ where n is any integer. And, from equation (I-11), the resulting bit-error probability is the same as for PCM/AM. Hence "switched PCM/FM" requires 0.8 db more power to accomplish a given bit-error probability than does conventional PCM/FM.

But the most important conclusion concerning further research in improved PCM/FM demodulation is that, in view of the excellent comparison between the experimental and theoretically optimum results of Figure 2, if we do not make use of data statistics in the demodulation process we cannot devise demodulation schemes for obtaining bit-error probabilities less than .05 with significantly less transmitted power than that required by conventional PCM/FM receiving equipment with good synchronization.

Similar results and conclusions could surely be obtained for PCM/AM and PCM/PSK. But PCM/FM has been emphasized here because of availability of measured results from independent laboratories and because most of the existing PCM telemetry systems utilize PCM/FM. The gains available by use of a priori data statistics in demodulation of binary PCM signals are investigated in the following chapters.

## Chapter 2

# MINIMUM-ERROR DEMODULATION OF BINARY PCM WAVEFORMS WITH DEPENDENCE BETWEEN DATA SAMPLES

The notation used here is again that of Figure 1. The x(t), y(t), and z(t) symbols represent signals existing over a time interval, T, at a data source (transducer) output, transmitter output, and receiver input respectively. For simplicity we treat a single data source. The extension of our results to multiple data sources will be apparent. The transmitted signal, y(t), is the serial binary PCM code waveform representing the amplitudes of successive samples of the data, x(t). The received signal, z(t), is the sum of y(t) and channel noise, n(t).

In Chapter 1, it was assumed that y(t) during any bit interval was independent of y(t) during any other bit interval. But in many cases there may be, for example, considerable statistical dependence between words (or coded samples of data) one or more frames apart. In this case, the probability distribution of the "possible" PCM signals over the interval T is not constant as assumed in Chapter 1, but is higher for waveforms which exhibit this "periodic dependence" than for waveforms which do not. Hence the optimum demodulator derived for completely random PCM waveforms is not necessarily optimum for these more realistic waveforms, and considerable improvement in sensitivity might be gained by taking advantage of this statistical dependence.

For data power spectra which extend from zero frequency to some finite frequency, $f_I$, and are zero for all higher frequencies we theoretically can sample the data at $2f_I$ samples per second and recover it with zero interpolation error. The data samples in this case will be uncorrelated if the spectrum amplitude is constant from zero frequency to $f_I$. But for a non-idealized data spectrum whose amplitude decreases less abruptly for higher frequencies (e. g., inversely proportional to some power of the frequency) any finite sample rate results in a non-zero interpolation error and, in general, a non-zero correlation between samples.

The types of non-ideal power spectra most realistic and useful for representing data spectra are not known since the determination of typical data spectra has received little attention in the past. But if for purposes of illustration, and for maintaining simplicity, (and since some results are available in Reference 8 for this case) we assume a 3rd order Butterworth spectrum, $S(f) = \dfrac{k}{1 + \left(\dfrac{f}{f_I}\right)^6}$ --- that is, the spectrum produced by passing white noise through a 3rd order, low-pass Butterworth filter (the ideal data mentioned above is of "infinite order") --- we find that if we sample it rapidly enough so that we can interpolate with 1% rms interpolation error we have a correlation coefficient of 0.905 between adjacent samples. This is determined as follows:

If we interpolate the samples of 3rd order Butterworth data with a Wiener optimum (minimum rms error) interpolation filter, the sampling rate, $f_s$, must be approximately ten times the break frequency, $f_I$, of the data spectrum if the rms interpolation error is to be one percent (see Figure 5, Part 1 of Reference 8). The normalized autocorrelation function for 3rd order Butterworth data is (see Table 3, Part 2, of Reference 8):

$$\rho(\tau) = 1/2 \exp(-\omega_I |\tau|) + \exp\left(-\frac{\omega_I |\tau|}{2}\right) \cos(.866\, \omega_I |\tau| - \pi/3)$$

and for $\tau = T_s = 1/f_s = 2\pi/\omega_s$ we get

$\rho(T_s) = .905$

similarly we find

$\rho(2T_s) = .677$

$\rho(3T_s) = .403$

etc.

Although little is known about the spectral characteristics of typical measured or telemetered data, the possibility of such high correlation between data samples raises the prospect of making use of this correlation in the demodulation of telemetered data. If the correlation coefficient between two samples were unity (i. e., the probability of the two samples being identical is one), and if we used the finite-time-correlation demodulation techniques of Chapter 1 (see equation (1.7) and the paragraph following it) we could, by using both samples, obtain a specified error probability with half the signal power required by using only one sample. This is due to the fact that when two noisy identical signals (noises uncorrelated, but from same random process) are operated on by identical finite time correlators whose outputs are summed, the signal component of the sum is twice that of either correlator, but the rms value of the noise component of the sum is only $\sqrt{2}$ times that of either correlator. Hence the ratio of signal amplitude to rms noise for the sum will be $\sqrt{2}$ times greater than for a single correlator. Since the same result would be obtained by using one sample with twice the signal power, a power gain of 3 db is made possible by using two data samples with unity correlation coefficient. We now investigate the power gains made possible by correlation coefficients less than unity between data samples.

The difficulty in deriving a maximum-likelihood demodulator for PCM considering data statistics is a consequence of the difficulty in analytically relating the statistics of the modulated signals that are transmitted with the statistics of the data (assumed known). The difficulty in relating these statistics is due to the fact that a PCM signal is not a continuous function of the modulating (data) signal. Hence, for deriving a maximum-likelihood demodulator for PCM, it is more convenient to consider the maximum-likelihood receiver to be that which determines the most likely PCM signal rather than the most likely data signal. This is a reasonable approach since there is a unique (though non-analytic) correspondence between a PCM signal and the data samples which it represents. That is, the coding and modulation operations are deterministic and reversible. Hence, determining the most likely PCM signal is equivalent to determining the most likely sequence of quantized data samples, and the interpolation of the sample values need not be considered as part of the demodulation process (see Figure 1). However, with this approach it is not at all clear how one might make use of a priori data statistics such as autocorrelation of the data (or correlation between data samples) in the demodulation process. It appears that we must somehow be able to make use of the PCM signal statistics resulting from these data statistics.

For the case treated in Chapter 1, minimum-error demodulation has been regarded as the process of deciding which of all possible signals extending over some time interval, T, is most likely to have been transmitted, given a received signal with independent, additive, band-limited white gaussian noise. This requires signal probability distributions for entire signals (elements of a random process ensemble). The concept of signal space is very convenient for envisioning entire signals as the elements of a probability distribution. But useful mathematical expressions for such distributions cannot always be obtained. For a gaussian random process, such expressions may be obtained as in Reference 11, but PCM signals do not represent a gaussian random process; and although the modulating data signals might be assumed to come from a gaussian random process, the treatment of Reference 11 places requirements on the modulation operator which are not satisfied by PCM as pointed out in Reference 11. But if we consider each transmitted y-waveform extending over time interval T to be a sequence of binary PCM words, $y_1$, $y_2$, ---, $y_n$, and each received z-waveform to be a sequence of noisy binary PCM words, $z_1$, $z_2$, ---, $z_n$, then the inverse probability distribution of the transmitted waveforms may be written:

$$q(y_1, ---, y_n \mid z_1, ---, z_n) = \frac{r(z_1, ---, z_n \mid y_1, ---, y_n) g(y_1, ---, y_n)}{s(z_1, ---, z_n)}$$

where $q(y_1, ---, y_n \mid z_1, ---, z_n)$ and $g(y_1, ---, y_n)$ are probability distributions of the y-waveforms, and $r(z_1, ---, z_n \mid y_1, ---, y_n)$ and $s(z_1, ---, z_n)$ are probability density functions of the z-waveforms. The remarks made on page 4 regarding the interpretation of probability density functions for waveforms applies to the above probability density functions.

Since the a priori joint probability distribution $g(y_1, ---, y_n)$ of the word waveforms $y_1, ---, y_n$ is equivalent to the a priori joint probability distribution $f(Y_1, ---, Y_n)$ of the corresponding quantized data samples $Y_1, ---, Y_n$ (we assume $f(Y_1, ---, Y_n)$ to be known), and since $r(z_1, ---, z_n \mid y_1, ---, y_n)$ could be obtained from the assumed distribution, $h(n)$, of the additive, independent, band-limited white gaussian noise, we could evaluate $g(y_1, ---, y_n \mid z_1, ---, z_n)$ for each possible set of $y_i$'s with a given set of $z_i$'s, and choose the set of $y_i$'s which gives the greatest value. But this process would not necessarily yield the lowest possible word error probability since the most probable sequence of words is not necessarily the sequence of most probable words in each word-position. If we wish to minimize word error probability we should choose the most probable word in each word-position (or "frame").

Let us assume that we have stored n frames of received signal, and let $y_i$ be the signal (word) transmitted in the $i^{th}$ frame and $z_i$ the received signal in the $i^{th}$ frame. We wish to make a maximum probability estimate of $y_j$, knowing $z_1, ---, z_j, ---, z_n$. That is we wish to choose the $y_j$ waveform which maximizes $p(y_j \mid z_1, ---, z_n)$. We may obtain $p(y_j \mid z_1, ---, z_n)$ by summing the joint probability distribution $q(y_1, ---, y_n \mid z_1, ---, z_n)$ over all $y_i$ except $y_j$:

$$p(y_j \mid z_1, ---, z_n) = \sum_{y_1 \in U} --- \sum_{y_{j-1} \in U} \sum_{y_{j+1} \in U} --- \sum_{y_n \in U} q(y_1, ---, y_n \mid z_1, ---, z_n)$$

$$= \sum_{y_1 \in U} --- \sum_{y_{j-1} \in U} \sum_{y_{j+1} \in U} --- \sum_{y_n \in U} \frac{r(z_1, ---, z_n \mid y_1, ---, y_n) g(y_1, ---, y_n)}{s(z_1, ---, z_n)} \tag{2.1}$$

where U = the set of all possible transmitted PCM waveforms during one word-time, $T_W$.

The $y_i$'s and $z_i$'s now represent waveforms of duration $T_W$.

If the noise is independent of the transmitted signal y then:

$$r(z_1, ---, z_n \mid y_1, ---, y_n) = h(z_1 - y_1, ---, z_n - y_n)$$

If the noise is also independent from frame to frame:

$$r(z_1, ---, z_n \mid y_1, ---, y_n) = h(z_1 - y_1) --- h(z_n - y_n) \tag{2. 2}$$

Here h is used to represent both joint and marginal distribution for the noise random process. Then:

$$p(y_j \mid z_1, ---, z_n) = K_5(z) \sum_{y_1 \in U} --- \sum_{y_{j-1} \in U} \sum_{y_{j+1} \in U} --- \sum_{y_n \in U} h(z_1 - y_1) h(z_2 - y_2) ---$$

$$--- h(z_n - y_n)\, g(y_1, ---, y_n) \tag{2. 3}$$

where $K_5(z) = \dfrac{1}{s(z_1, ---, z_n)}$ and is independent of $y_j$.

Since $g(y_1, ---, y_n)$ is equivalent to $f(Y_1, ---, Y_n)$, equation (2. 3) expresses $p(y_j \mid z_1, ---, z_n)$ in terms of known functions of the received z's and all possible combinations of transmitted y's. Therefore, in principle the problem of computing $p(y_j \mid z_1, ---, z_n)$ for any specified $z_1, ---, z_n$ is solved. However, in practice the problem still appears quite formidable because of the number of operations required for the computation with a reasonable number, m, of bits per word and a reasonable number, n, of received words to be considered. For example, if m = 6 and n = 5 the multiple summation involves more than sixteen million terms for each possible $y_j$ waveform, of which there are $2^m$; and the complete computation must be made for each word to be demodulated. Hence it appears that the only feasible method for making the computations is by use of a high speed digital computer.

The significance of equation (2. 3) lies in the fact that if the estimated $y_j$ is that which maximizes (2. 3), the probability of being wrong (i. e., probability of error) is the lowest obtainable by any method of estimation which makes use of only the n received words, $z_1, ---, z_n$. If we can calculate this error probability as a function of signal-to-noise power ratio for any assumed n and data statistics we have calculated the lowest possible error probability attainable for the assumed conditions. No explicit expression for this error probability has been obtained. Consequently calculation of the error probability must be accomplished by a model-sampling or "monte-carlo" technique, which in its simplest form would consist of selecting sets of z's (noisy waveforms) from the proper distribution and operating on them as indicated by equation (2. 3) to select the $y_j$ which maximizes (2. 3). This must be repeated, observing the frequency with which errors are made in selecting $y_j$, until an estimate can be made, with reasonable confidence, of the average error rate or error probability. The means for implementing this operation in a high speed digital computer must now be considered.

## Chapter 3

## SIMULATION OF MINIMUM-ERROR DEMODULATION WITH A DIGITAL COMPUTER

The expression (2. 3) has a numerical (probability) value for any set of waveforms $y_j, z_1, ---, z_n$. In order to determine these numerical values we must first be able to determine numerical values for the factors of the form $h(z_i - y_i)$ for any waveforms $z_i$ and $y_i$. These factors are the values of the noise probability density function, $h(n)$, for $n = z_i - y_i$. We assume the noise waveform to be a band-limited white gaussian waveform of duration $T_W$, in which case it can be shown that (see Reference 14):

$$h(z_i - y_i) = K_6 \exp \left( - \frac{1}{K^2} \int_{T_W} (z_i - y_i)^2 dt \right) \tag{3. 1}$$

where $K^2$ is the one sided spectral height of the noise and $K_6$ is a constant.

As in Chapter 1, the term "band-limited white" here means that the power spectrum is of uniform height, $K^2$, from zero frequency to some arbitrarily large but finite frequency, W, and of zero height for all higher frequencies.

The integration in (3. 1) is over the $i^{th}$ word-time. This equation can be reduced to:

$$h(z_i - y_i) = F(E_{z_i}, E_{y_i}) \exp \left( \frac{2}{K^2} \int_{T_W} z_i y_i dt \right) \tag{3. 2}$$

where $E_{z_i} = \int_{T_W} z_i^2 dt$

$E_{y_i} = \int_{T_W} y_i^2 dt$

For symmetrical PCM waveforms, $E_{y_i}$ is the same for all $y_i \in U$. Since the $z_i$'s are all given and are unchanged for calculating $p(y_j | z_1, ---, z_n)$ for any $y_j \in U$, the $E_{z_i}$'s are also unchanged. Hence $F(E_{z_i}, E_{y_i})$ may be considered a constant and, for purposes of maximizing $p(y_j | z_1, ---, z_n)$ with respect to $y_j$, equation (3. 2) may be written:

$$h(z_i - y_i) = K_7(z) \exp\left(\frac{2}{K^2} \int_{T_W} z_i y_i dt\right)$$

The multiple summation of equation (2. 3) is taken over all $y_1 \in U$, $y_2 \in U$, etc. for all $y_i$ except $y_j$. For m-bit words there are $2^m$ possible waveforms for each $y_i$. That is, the set U is made up of $2^m$ different PCM waveforms. (The results are easily extendable to redundant codes in which there are less than $2^m$ possible transmitted waveforms). Each of these waveforms will be distinguished in our y-notation by a second subscript. For example the $p^{th}$ waveform, from some ordered arrangement of the waveforms, is $y_{i(p)}$. (The most obvious "ordered arrangement" for ordinary binary PCM waveforms is in the numerical order of their binary number representation.) Using this more explicit notation, the above equation becomes:

$$h(z_i - y_{i(p)}) = K_7(z) \exp\left(\frac{2}{K^2} \int_{T_W} z_i y_{i(p)} dt\right) \tag{3. 3}$$

Furthermore, the value of the first bit of $y_{i(p)}$ will be represented by $y_{i(p)1}$, the second bit by $y_{i(p)2}$, etc. Then the exponential in equation (3. 3) is:

$$\frac{2}{K^2} \int_{T_W} z_i y_{i(p)} dt = \sum_{r=1}^{m} \left(\frac{2}{K^2} \int_{T_B} z_{ir} y_{i(p)r} dt\right) \tag{3.4}$$

where the integration of the $r^{th}$ term of the summation is taken over the $r^{th}$ bit-time (of duration $T_B$). We assume here that the bandwidth, W, of the band-limited white noise is large compared to $1/T_B$.

Let the binary character of the transmitted PCM signals be represented during each bit-time by either of two known waveforms[1], $g_1(t)$ and $g_2(t)$ (e.g., $g_1(t)$ representing "yes" bits and $g_2(t)$ representing "no" bits). We must require $g_1(t)$ and $g_2(t)$ to have equal energies since we have already made that assumption implicitly by assuming that $E_{y_i}$ is the same for all $y_i \in U$.

That is, we must require that

$$\int_{T_B} g_1(t)^2 dt = \int_{T_B} g_2(t)^2 dt = E_g = S^2 T_B = \frac{S^2}{B} \qquad (3.5)$$

where $S^2$ = average transmitted signal power. (We will later extend our results to include any two waveforms $f_1(t)$ and $f_2(t)$ without the equal energy requirement - see Chapter 5.)

The finite-time correlation coefficient for $g_1(t)$ and $g_2(t)$ is:

$$\lambda = \frac{1}{E_g} \int_{T_B} g_1(t) g_2(t)\, dt \qquad (3.6)$$

It can have any value from -1 to +1.

Let $n_r(t)$ be the noise waveform during the $r^{th}$ bit time. That is, $n_r(t)$ is the noise waveform which is added to the transmitted signal, $y_{i(s)r}$ to give the received signal $z_{ir}$ during the $r^{th}$ bit time. Now consider the integral

$$N_r = \frac{2}{K^2} \int_{T_B} n_r(t) y_{i(p)r}(t)\, dt$$

where $y_{i(p)r}(t)$ is either $g_1(t)$ or $g_2(t)$

1 In the treatment following it is assumed that the pair of waveforms, $g_1(t)$ and $g_2(t)$, is known for each bit-time independently of the waveform existing during any other bit-time. This assumption does not include PCM/FM with non-integer deviation ratio (see first paragraph of Appendix I).

Since $n_r(t)$ is a sample waveform from a band-limited white stationary random gaussian process of zero mean and two-sided spectral height $K^2/2$, $N_r$ is a random gaussian variable of zero mean and mean square value

$$\overline{N_r^{\,2}} = \frac{4}{K^4} \overline{\int_{T_B}\int_{T_B} g(t_1)g(t_2)n_r(t_1)n_r(t_2)\,dt_1dt_2}$$

$$= \frac{4}{K^4} \int_{T_B}\int_{T_B} g(t_1)g(t_2)R_n(t_1-t_2)\,dt_1dt_2$$

where $R_n(t_1 - t_2) = \overline{n_r(t_1)n_r(t_2)} = \left(K^2/2\right)\delta\,(t_1-t_2)$

($\overline{x}$ is used here to indicate the ensemble average, or expected value, of x.)

Therefore:

$$\overline{N_r^{\,2}} = \frac{2}{K^2}\int_{T_B} g(t)^2 dt = \frac{2}{K^2}\,E_g = \frac{2S^2}{K^2B}$$

For a specific $n_r(t)$ waveform, $N_r$ will have one value, $N_{r1}$, for $y_{i(p)r}(t) = g_1(t)$ and (in general) another value, $N_{r2}$, for $y_{i(p)r}(t) = g_2(t)$. $N_{r1}$ and $N_{r2}$ are random gaussian variables of zero mean, variance $\overline{N_r^{\,2}}$, and correlation coefficient $\rho_N$ where:

$$\rho_N = \frac{\overline{N_{r1}N_{r2}}}{\overline{N_r^{\,2}}} = \frac{4}{K^4\,\overline{N_r^{\,2}}} \overline{\int_{T_B}\int_{T_B} g_1(t_1)g_2(t_2)n(t_1)n(t_2)\,dt_1dt_2}$$

$$= \frac{4}{K^4\,\overline{N_r^{\,2}}} \int_{T_B}\int_{T_B} g_1(t_1)g_2(t_2)R_n(t_1-t_2)\,dt_1dt_2$$

$$\rho_N = \frac{2}{K^2 \overline{N_r^2}} \int_{T_B} g_1(t) g_2(t)\, dt = \lambda$$

Returning now to equation (3.4), if $y_{i(s)}$ is the transmitted waveform during the $i^{th}$ word-time the $r^{th}$ term of the summation of (3.4) is

$$\frac{2}{K^2} \int_{T_B} z_{ir} y_{i(p)r} dt = \frac{2}{K^2} \int_{T_B} \left[ y_{i(s)r}(t) + n_r(t) \right] y_{i(p)r}(t)\, dt$$

$$= \begin{cases} \dfrac{2S^2}{K^2 B} + N_r & \text{for matched bits (i.e., } y_{i(s)r} = y_{i(p)r}) \\ \lambda \dfrac{2S^2}{K^2 B} + N_r & \text{for unmatched bits} \end{cases} \qquad (3.7)$$

where: $N_r = N_{r1}$ if $y_{i(p)r} = g_1(t)$ (e.g., a "yes" bit)

$N_r = N_{r2}$ if $y_{i(p)r} = g_2(t)$ (e.g., a "no" bit)

Now the actual minimum-error demodulator, having available only the received noisy waveforms, $z_i$, would determine the $h(z_i - y_{i(p)})$ values by correlating the received $z_i$'s with each possible transmitted waveform, $y_{i(p)}$, and then exponentiating the results as indicated in (3.3). But for purposes of simulating the operation in a digital computer (e.g., for a monte carlo method) where we must manipulate digital quantities rather than waveforms, the transmitted signals (y's) must be generated by the computer and therefore it has the information needed to determine for any assumed waveform, $y_i$, the bits which "match" the actual transmitted waveform, $y_{i(s)}$, and those which are "unmatched." Therefore the computer can calculate the $h(z_i - y_{i(p)})$ values of equation (3.3) by use of (3.4) and (3.7). Hence the only use that the computer need make of the noise portion of the received signal waveform is to determine $N_{r1}$ and $N_{r2}$. Since $N_{r1}$ and $N_{r2}$ are simply two correlated random gaussian variates with variance and correlation coefficient determined by $\frac{S^2}{K^2B}$ and $\lambda$, we can, for computer simulation purposes, select $N_{r1}$ and $N_{r2}$ directly from the proper 2-dimensional gaussian distribution and use them in (3.7).

Note that the only characteristics of the waveforms $g_1(t)$ and $g_2(t)$ used in this operation are the mean square value, $S^2$, (assumed the same for both waveforms) and the finite-time correlation coefficient, $\lambda$. Hence results obtained for any pair of waveforms apply directly to any other pair (with equal energies) having the same $\lambda$.

The basic general computer procedure then for generating sets of z's and evaluating the factors of the form $h(z_i - y_{i(p)})$ for a sample calculation is to first select n data samples having the appropriate correlation between samples, code these samples in m-bit binary code and store these. For each code bit, select two random numbers, $N_{r1}$ and $N_{r2}$, from a two-dimensional gaussian distribution with variance $\frac{2S^2}{K^2B}$ and correlation coefficient $\lambda$ (determined from the assumed $g_1(t)$ and $g_2(t)$). Evaluate each $h(z_i - y_{i(p)})$ by use of (3.7), (3.4), and (3.3).

The required calculation procedure is made clearer if we discuss it in terms of specific waveforms for $g_1(t)$ and $g_2(t)$. The simplest possible waveforms of equal energy are $g_1(t) = +S$ and $g_2(t) = -S$ illustrated in Figure 3. For these waveforms $\lambda = -1$, and therefore $N_{r1} = -N_{r2}$. This facilitates the generation of bit-noise in the computer and therefore the computer calculations were carried out for these particular waveforms. As mentioned above, the results apply directly to any equal-energy waveforms having the same $\lambda$. We will see later that the results can easily be applied to any waveforms whatever. The procedure discussed in the following paragraphs for these specific simple waveforms is slightly different from the

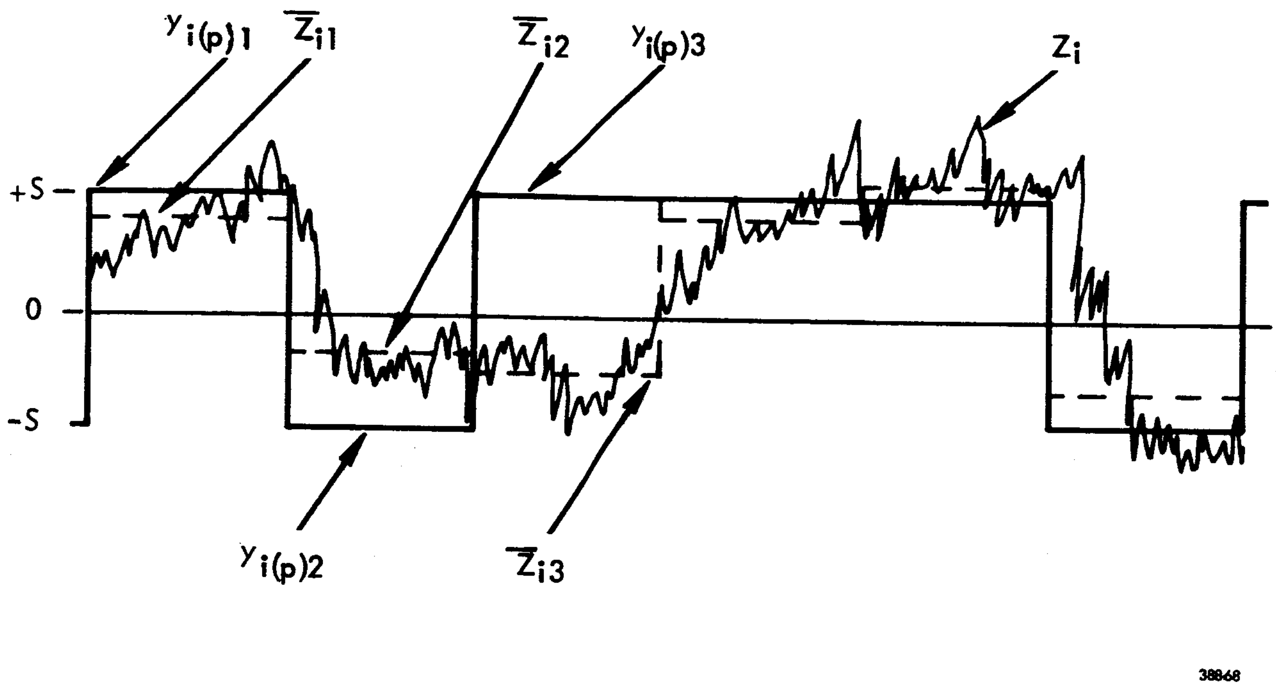



Figure 3 Typical Waveforms

general procedure outlined in the above paragraph since it is not necessary to use equation (3.7) in this simple case. Equation (3.4) can be used as follows.

Consider the waveforms shown in Figure 3.

$y_{i(p)}$ is the (solid-line) symmetrical PCM waveform

$z_i$ is the (solid-line) noisy waveform

The dotted lines represent the average values of $z_i$ over individual bit-times and this average value over the $r^{th}$ bit time is designated by $\bar{z}_{ir}$. Equation (3.4) may now be written:

$$\frac{2}{K^2} \int_{T_W} z_i y_{i(p)} dt = \frac{2T_B}{K^2} \sum_{r=1}^{m} y_{i(p)r} \, \bar{z}_{ir} \qquad (3.8)$$

Note that $y_{i(p)r}$ can be either $+S$ or $-S$, but can have no other values. Therefore, since $z_i$ is some $y_i$ to which band-limited white gaussian noise of one-sided spectral height $K^2$ has been added:

$\bar{z}_{ir} = \pm S$ + the average over one bit-time, $T_B$, of some band-limited white gaussian noise of zero mean and of spectral height $K^2$

$= \pm S + N_{ir}$

where $N_{ir}$ is a sample of a random gaussian variable of zero mean, and variance

$$= \frac{K^2}{2T_B}$$

We now define $\zeta_{ir}$ as a normalized $\bar{z}_{ir}$:

$$\zeta_{ir} = \frac{\bar{z}_{ir}}{\frac{K}{\sqrt{2T_B}}} = \frac{\pm S}{\frac{K}{\sqrt{2T_B}}} + \frac{N_{ir}}{\frac{K}{\sqrt{2T_B}}} = \frac{\pm S}{\frac{K}{\sqrt{2T_B}}} + \nu_{ir} \qquad (3.9)$$

where $\nu_{ir} = \frac{N_{ir}}{\frac{K}{\sqrt{2T_B}}}$ is a sample of a random gaussian variable of zero mean and unity variance. These samples may be assumed independent from one bit-time to another since the bandwidth, W, of the white noise is assumed large compared to $1/T_B$.

If we define normalized PCM waveforms:

$$\eta_i = \frac{y_i}{\frac{K}{\sqrt{2T_B}}}$$

Then:

$$\eta_{i(p)r} = \frac{y_{i(p)r}}{\frac{K}{\sqrt{2T_B}}} = \frac{\pm S}{\frac{K}{\sqrt{2T_B}}} \tag{3.10}$$

Equation (3.8) becomes:

$$\frac{2}{K^2} \int_{T_W} z_i y_{i(p)} dt = \sum_{r=1}^{m} \eta_{i(p)r} \, \xi_{ir} \tag{3.11}$$

Equation (3.3) then becomes:

$$h(z_i - y_{i(p)}) = K_7(z) \exp \left( \sum_{r=1}^{m} \eta_{i(p)r} \, \xi_{ir} \right) \tag{3.12}$$

All of the factors of equation (2.3) are of this form except for the factor $g(y_1, ---, y_n)$ which is the joint probability distribution of the PCM waveforms $y_1, ---, y_n$. But each waveform, $y_i$, is a (binary) representation of a particular quantized data value or sample, $Y_i$, and each set of waveforms, $y_1, ---, y_n$, represents a corresponding set of quantized data samples $Y_1, ---, Y_n$. Therefore, the probability of occurrence of a particular set of waveforms is the same as the probability of occurrence of the corresponding set of quantized data samples. Hence the joint probability distribution, $g(y_1, ---, y_n)$, of the waveforms can be replaced by the joint probability distribution, $f(Y_1, ---, Y_n)$, of the corresponding quantized data samples.

For any given set of z's, equation (2.3) may be evaluated for each $y_j$ waveform, $y_{j(1)}, y_{j(2)}, ---$ etc. or, in general, for $y_{j(p)}$. Using equation (3.12) and replacing $g(y_1, ---, y_n)$ by $f(Y_1, ---, Y_n)$ we may express the right hand side of equation (2.3) in terms of numbers rather than waveforms so that a purely numerical evaluation is possible:

$$p(y_{j(p)} | z_1, ---, z_n) = K_8(z) \sum_{\substack{\eta_i \in V \\ i = 1, --- n \\ i \neq j}} \exp\left( \sum_{r=1}^{m} \eta_{ir} \xi_{ir} + --- + \eta_{nr} \xi_{nr} \right) f(Y_1, ---, Y_n) \tag{3.13}$$

where V is the set of all possible normalized PCM word waveforms.

It may be noted again that the first subscript, 1, 2, ---, j, etc. refers to the word-time-slot; the subscript in parenthesis, (p), (q), ---, refers to the particular m-bit PCM waveform; and the other subscript, r, refers to the bit-time-slot.

Since evaluation of (3.13) involves well defined manipulations of numbers, we may simulate the operation with a digital computer.

# Chapter 4

## PROCEDURES FOR COMPUTATION OF ERROR PROBABILITIES

The computer simulation of minimum-error demodulation will produce the same average error rate, or average error probability, as would an actual minimum-error demodulator provided the sets of z's used by the computer are representative of those which the actual device would operate upon. We can satisfy this requirement by selecting the z's from the proper z distribution, a process sometimes called "model sampling." Since each z is formed by the addition of a PCM waveform, y, and a noise waveform, n, we may insure that the z's are from the proper distribution by selecting the y's and n's from the proper distributions. The probability distribution $g(y_1, ---, y_n)$ of a set of y's is, as previously discussed, uniquely determined by the equivalent distribution $f(Y_1, ---, Y_n)$ of the quantized data samples. The probability density function of the noise waveforms is h(n), but in Chapter 3 we found that we do not need to make use of the complete noise waveforms in our calculations. We make use only of $N_{ir}$'s, the average values of the noise over one bit-time, $T_B$. We found in Chapter 3 that these average values are simply random numbers selected from a gaussian distribution of variance $= \frac{K^2}{2T_B}$. Hence we may, for purposes of these calculations, generate sets of z's from the proper distribution by selecting a set of Y's from the proper joint distribution, code these in binary PCM code of amplitude $\pm \frac{S}{K\sqrt{B/2}}$, and add to the amplitude of each bit of the codes constant, independent values (i.e., the $\nu$'s of equation (3.9) ) selected from a gaussian distribution of zero mean and unity variance. The resulting numbers are the $\zeta$'s required in equation (3.13). We shall henceforth refer to the ratio $\frac{S}{K\sqrt{B/2}}$ simply as the "signal-to-noise ratio," S/N.

We assume the data to be from a gaussian random process with specified power spectrum (or autocorrelation function), mean $m_y$ and variance $\sigma_y^2$. If the data quantization intervals are such that the probability density function of the data amplitude does not change appreciably over the quantization intervals, then for calculation of the joint probabilities of data samples we may use the data value at the center of a quantization interval for any data sample falling in that interval (see Figure 4). But an m-bit binary code can represent only $2^m$ distinct levels or quantization intervals. Hence only a finite range of data amplitude can be represented by such a code. In Figure 4 (shown for m = 6) this finite range has been chosen to be 5.2 $\sigma_y$ (or $m_y \pm 2.6\, \sigma_y$). The probability of the data occurring outside this amplitude range is less than 1%. When data samples outside this range do occur they are coded as "zero" (if below this range) or "full scale" (if above this range). The joint probability distribution, $f(Y_1, ---, Y_n)$ of quantized data samples $Y_1, ---, Y_n$ is then essentially an $n^{th}$ order gaussian distribution with correlation coefficients $\rho_{hi}$ equal to the values of the normalized autocorrelation function of the data, $\rho(\tau_{hi})$, where $\tau_{hi}$ is the time between samples $Y_h$ and $Y_i$:

$$f(Y_1, ---, Y_n) = K_9 \exp\left( - \frac{1}{2\sigma_y^2 |\rho|} \sum_{h=1}^{n} \sum_{i=1}^{n} |\rho|_{hi} (Y_h - m_y)(Y_i - m_y) \right)$$

$$= K_{10} \exp\left( - \frac{1}{2|\rho|} \sum_{h=1}^{n} \sum_{i=1}^{n} |\rho|_{hi} X_h X_i \right)$$

where $X = \dfrac{Y - m_y}{\sigma_y}$

$|\rho|$ = determinant of correlation matrix, $[\rho]$

$|\rho|_{hi}$ = cofactor of element $\rho_{hi}$ of $|\rho|$

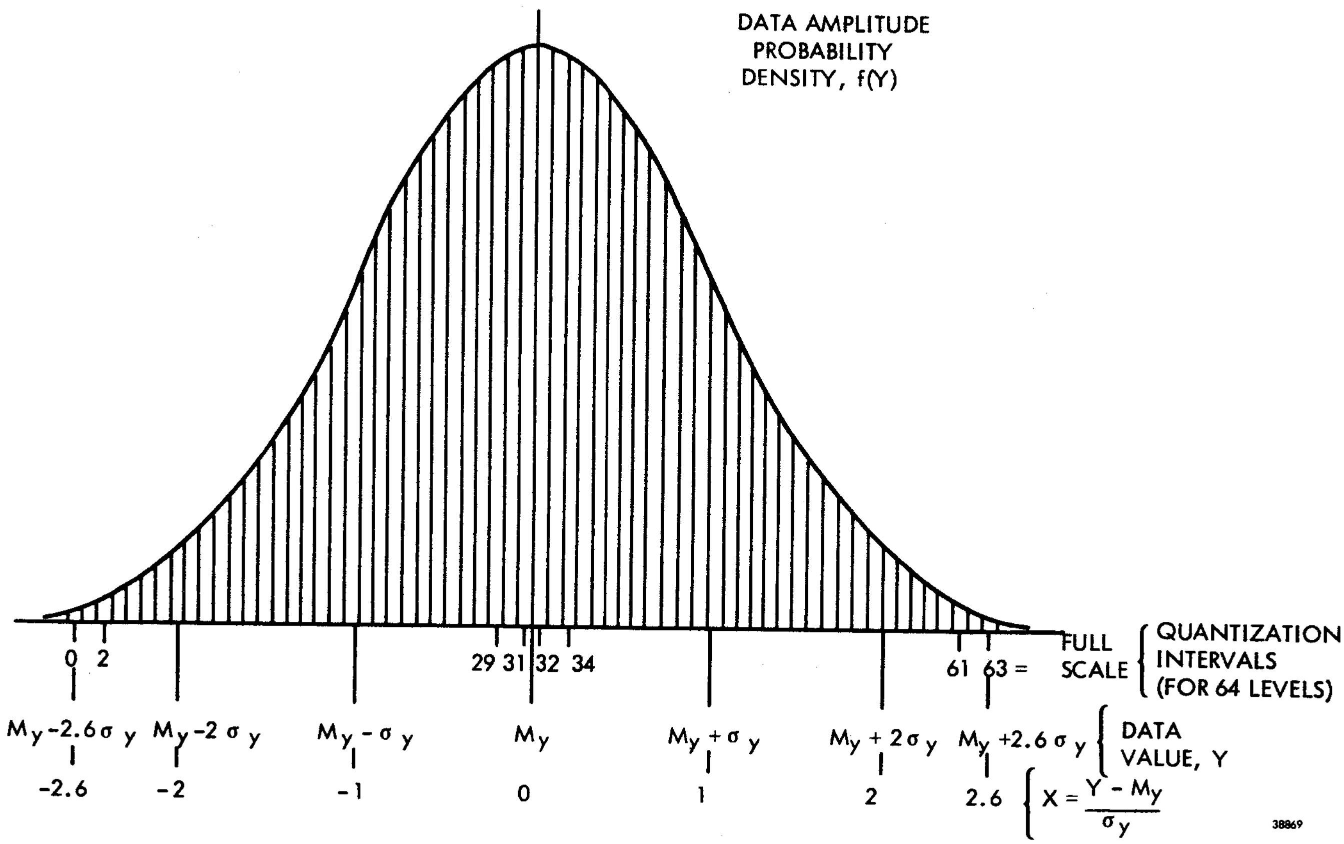



Figure 4 Assumed Data Amplitude Probability Density

The generation of independent gaussian numbers is not difficult. One way in which such numbers may be generated in a digital computer is by first generating groups of random numbers (say M numbers in each group) uniformly distributed between zero and unity, and then adding the numbers in each group. If M is fairly large (say 10 or more) these sums are, for practical purposes, random independent gaussian variates of mean M/2 and variance $\frac{M}{12}$. But we want Y's which are <u>dependent</u> (or correlated) gaussian variates with correlation matrix $[\rho]$. We may, however, generate n independent variates and then transform them into n correlated variates with correlation matrix $[\rho]$ as follows:

Let $u_1, u_2, ---, u_n$ be uncorrelated gaussian variates, with zero mean and unity variance which we wish to transform into correlated gaussian variates $Y_1, Y_2, ---, Y_n$ with zero mean and unity variance and with the specified correlation matrix:

$$[\rho] = \begin{bmatrix} 1 & \rho_{12} & ----- & \rho_{1n} \\ \rho_{21} & 1 & ----- & \rho_{2n} \\ \vdots & \vdots & & \vdots \\ \rho_{n1} & \rho_{n2} & ----- & 1 \end{bmatrix}$$

If we find the eigen values, $\Lambda_1, \Lambda_2, ---, \Lambda_n$, and the eigen vectors, $e_1, e_2, ---, e_n$, of the matrix, $[\rho]$, the desired transformation is: (see Appendix IV):

$$[Y] = [P]\ [L]\ [U] \qquad (4.1)$$

where $[P]$ is the (orthogonal) matrix whose column elements are the eigen vector components, and

$$[L] = \begin{bmatrix} \sqrt{\Lambda_1} & 0 & ----- & 0 \\ 0 & \sqrt{\Lambda_2} & ----- & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & ----- & \sqrt{\Lambda_n} \end{bmatrix}$$

$\left[U\right]$ is column matrix with elements $u_1, u_2, ---, u_n$

$\left[Y\right]$ is column matrix with elements $Y_1, Y_2, ---, Y_n$

We now have all the formulations and assumptions necessary to make error probability calculations with a digital computer using the model-sampling (or monte-carlo) technique. But the computing time required increases very rapidly as m and n get large. We can learn a great deal, however, about the gains available by use of data correlation from consideration of the gain for n = 2 -- that is, we examine 2 received (correlated) words in estimating the transmitted waveform for one of them. The cases investigated for this report are m = 6 and 3, n = 1 and 2. For m = 6 and n = 2 the time required by an IBM 7090 for each simulated demodulation is approximately 0.007 minute. For this case the data correlation transformation, (4.1), becomes (see Appendix IV):

$$Y_1 = \sqrt{\frac{1+\rho_{12}}{2}}\, u_1 + \sqrt{\frac{1-\rho_{12}}{2}}\, u_2 \qquad (4.2a)$$

$$Y_2 = \sqrt{\frac{1+\rho_{12}}{2}}\, u_1 - \sqrt{\frac{1-\rho_{12}}{2}}\, u_2 \qquad (4.2b)$$

and equation (3.13) becomes:

$$p(y_{2(p)} \mid z_1, z_2) = K_a(z) \exp\left(\sum_{r=1}^{6} \eta_{2(p)r}\,\xi_{2r}\right) \sum_{q=0}^{63} \exp\left(\sum_{r=1}^{6} \eta_{1(q)r}\,\xi_{1r}\right)$$

$$\exp\left(-\frac{X_{1(q)}^{2} - 2\rho_{12} X_{1(q)} X_{2(p)} + X_{2(p)}^{2}}{2(1-\rho_{12}^{2})}\right) \qquad (4.3)$$

A simple model-sampling estimation of error probability would be to note, for each simulated demodulation, whether or not an error was made and take the ratio of total errors to total demodulations as an estimate of the word-error probability, $P_W$. For this simple technique we can calculate, for any desired confidence level, the number of total demodulations required (and hence the computation time) as follows. Consider a random variable, U, which has a value of unity when a demodulation results in an error, and a value of zero when no error is made. The mean (expected) value of U is then the word-error probability, $P_W$, which we wish to estimate. For L demodulations the random variable

$$P_1 = \frac{1}{L} \sum_{i=1}^{L} U_i$$

is our estimate of $P_W$. But $P_1$ has a binomial distribution of mean $P_W$ and standard deviation $\sigma_{P_1} = \sqrt{P_W(1-P_W)/L}$. The 95% confidence interval is (for reasonably large N) approximately $\pm\, 2\,\sigma_{P_1}$. So if we require 95% confidence that $P_1$ is within 5% of $P_W$ we must have $2\,\sigma_{P_1} / P_W = 0.05$. Therefore for $P_W = 0.1$ (for example) we get L = 14,400 requiring (14,400)(0.007) $\cong$ 100 minutes of computing time on the 7090 for each combination of data correlation coefficient and signal-to-noise ratio. For $P_W = 0.3$ we find that L = 3,700. We need to reduce these by a factor of twenty or thirty for a reasonable total computation time on the 7090 (say around one hour). To accomplish this we must modify our technique so that our estimate of $P_W$ requires fewer simulated demodulations (i.e., fewer "samples") in order to converge to $P_W$ with reasonable confidence.

One such modification of our technique would be to find a better statistic for estimating $P_W$ than the estimate, $P_1$, used above. Such a statistic must have an expected value (or mean) equal to $P_W$ and, in order to be "better" for our purposes, its variance about this mean must be acceptably low with fewer samples (simulated demodulations) than required for $P_1$. Such an estimate can be obtained as follows.

To accomplish each simulated demodulation we calculate, from equation (3. 13), the value of $p(y_{2(p)} | z_1, z_2)$ for each of the $2^m$ possible $y_2$ waveforms and choose the $y_2$ corresponding to the largest value of $p(y_{2(p)} | z_1, z_2)$. Let this largest value be $p_M$. Then $p_M$ is the probability that we have chosen, for the particular set of z's used in that simulated demodulation, the correct $y_2$. Then $Q = 1-p_M$ is the probability, for that set of z's, that we will choose the wrong $y_2$ -- that is, that we will make an error. Therefore if we choose our sets of z's from the proper distribution (which we shall do), Q is an unbiased estimate of $P_W$, and hence so is

$P_2 = \frac{1}{L} \sum_{i=1}^{L} Q_i$ where L is the number of samples, or simulated demodulations, used in the estimate. Since the terms in the summation of $P_1$ must be either zero or unity while the terms in the summation of $P_2$ may have any values from zero to unity, we might expect that $P_2$ is a better estimator of $P_W$ than is $P_1$. This turns out to be true. For $P_W = 0.3$, $P_2$ gives a good estimate of $P_W$ with $L = 300$. But for $P_W$ of 0. 1 or less (corresponding to signal-to-noise ratio of about 2 or greater), the number of samples required is undesirably high and we must seek further means of modifying our model sampling technique so that fewer samples are required.

A method, applicable to our problem, for modifying the model sampling (or monte carlo) technique so that fewer samples may be required has been suggested by Kahn and Marshall (see Reference 5). The philosophy of this method (which is called "importance sampling") is as follows. Consider the random variable $Q = 1-p_M$ defined above. The randomness of Q derives from two sources: the randomness of the transmitted data values, Y, and the randomness of the noise, N. The amplitude of the noise is assumed gaussianly distributed as discussed earlier. Now consider $P_W$ as the expected value of Q:

$$P_W = \iint Q(Y, N)p(Y, N)dYdN \tag{4.4}$$

where p(Y, N) is the joint probability distribution of Y and N.

Our estimate, $P_2$, of $P_W$ is

$$P_2 = \frac{1}{L} \sum_{i=1}^{L} Q(Y_i, N_i) \tag{4.5}$$

where the Q's are calculated using $Y_i$'s and $N_i$'s selected from the distribution $p(Y, N)$.

If the integrand of (4.4) is multiplied and divided by some arbitrary probability distribution, $p^*(Y,N)$, it becomes:

$$P_W = \int\int \left( Q(Y,N) \frac{p(Y,N)}{p^*(Y,N)} \right) p^*(Y,N)\, dYdN \tag{4.6}$$

So if we now select $Y_i$'s and $N_i$'s from the "modified" distribution, $p^*(Y,N)$, (rather than from $p(Y,N)$), an unbiased estimate of $P_W$ is

$$P_3 = \frac{1}{L} \sum_{i=1}^{L} \left( Q(Y_i, N_i) \frac{p(Y_i, N_i)}{p^*(Y_i, N_i)} \right) \tag{4.7}$$

That is, we calculate the Q's as before but now we "weight" them by $\dfrac{p(Y_i, N_i)}{p^*(Y_i, N_i)}$ before summing them. As pointed out by Kahn and Marshall, $p^*(Y,N)$ can be selected such that the variance of $P_3$ is less than the variance of $P_2$ and consequently the sample size, L, can be smaller for $P_3$ than for $P_2$. In fact, $p^*(Y,N)$ can theoretically be selected such that the variance of $P_3$ is zero thus permitting us to take $L = 1$. This can be seen by noting that if $p^*(Y,N)$ is selected such that

$$p^*(Y, N) = \frac{Q(Y,N)\, p(Y,N)}{P_W}$$

equation (4.7) would become

$$P_3 = \frac{1}{L} \sum_{i=1}^{L} P_W = P_W \text{ for any L.}$$

Of course we cannot determine this optimum $p^*(Y, N)$ since it's determination requires us to know the very thing we are trying to estimate. But knowing that such a $p^*(Y,N)$ exists encourages us to attempt to find, by intuition or good luck, a $p^*(Y,N)$ close enough to the optimum so that the variance of $P_3$ is less than that of $P_2$, and hence requires fewer samples for satisfactory convergence to $P_W$. We may accomplish this in this case as follows.

Since Y and N are independent we may write $p(Y, N)$ as $r(Y)\, h(N)$. We may reason intuitively that since changes in the variance of our estimate $P_2$ result primarily from changes in the rms value of the noise, that the variance is largely due to the randomness resulting from N rather than Y. Therefore the difference between the original distribution, $p(Y, N)$, and the optimum "modified" distribution, $p^*(Y, N)$, should be largely due to modification of $h(N)$ rather than $r(Y)$. Hence we restrict our $p^*(Y, N)$ to be of the form $p^*(Y, N) = r(Y)\, h^*(N)$. With this restriction, equations (4.6) and (4.7) become:

$$P_W = \int\int \left( Q(Y, N) \frac{h(N)}{h^*(N)} \right) p^*(Y, N)\, dY dN \quad (4.8)$$

$$P_3 = \frac{1}{L} \sum_{i=1}^{L} \left( Q(Y_i, N_i) \frac{h(N_i)}{h^*(N_i)} \right) \quad (4.9)$$

and the optimum (zero variance) $h^*(N)$ is:

$$h^*(N) = \frac{Q(Y, N)\, h(N)}{P_W} \quad (4.10)$$

As pointed out in Chapter 3, the noise selected (by the computer) for each bit of the received signal is simply a number, $N_r$, selected from a gaussian distribution, $g(N_r)$. We may consider any noise, N, to be a set of independent bit-noises, $N_r$, and hence the probability density h(N) of any N is the product of the probability densities $g(N_r)$ for the corresponding set of $N_r$'s. Therefore, we may modify h(N) by modifying $g(N_r)$ to obtain $g^*(N_r)$. According to equation (4. 10) we should modify h(N) by multiplying it by Q(Y, N). Ignoring the variation of Q with Y we may reason that the variation of Q with the bit-noise, $N_r$, is such that Q is largest when the amplitude of $N_r$ is equal to the peak signal amplitude, S, (since then the signal plus noise may be halfway between + S and - S in which case we have the greatest uncertainty as to which signal, + S or - S, was transmitted) and decreases for larger or smaller $N_r$, the decrease being more rapid for higher signal-to-noise ratios. Hence to obtain $g^*(N_r)$ we should multiply the $g(N_r)$ of Figure 5(a) by a function shaped something like that of 5(b), with the result shown in 5(d). The result must, of course, be normalized so that it is indeed a probability density function. Somewhat the same result can be obtained by <u>adding</u> to $h(N_r)$ of 5(a) a function such as that of 5(c). This latter is very easy to accomplish in the computer by simply selecting some of the $N_r$'s from a flat distribution $f(N_r)$ such as 5(c) and some of them from $g(N_r)$ of 5(a). Which of these distributions is used for a particular selection is determined by playing an auxiliary game of chance, and the percentages of time that each should be used, as well as the half-range, B, of 5(c), are determined (for minimum variance of the estimate $P_3$) by experimentation. This technique of reducing the variance of our estimate, $P_3$, worked very well for $P_W$ of around 0. 1 (signal-to-noise ratio of 2). For this case, B of Fig. 5(c) was 2. 5 and the flat distribution $f(N_r)$ of Fig. 5(c) was used (randomly) for selecting 20% of the bit noise values. With these values, $P_3$ gives a good estimate of $P_W$ with L = 300. For $P_W$'s less than 0. 1 (signal-to-noise ratios of 3 or greater) this simple technique did not reduce the

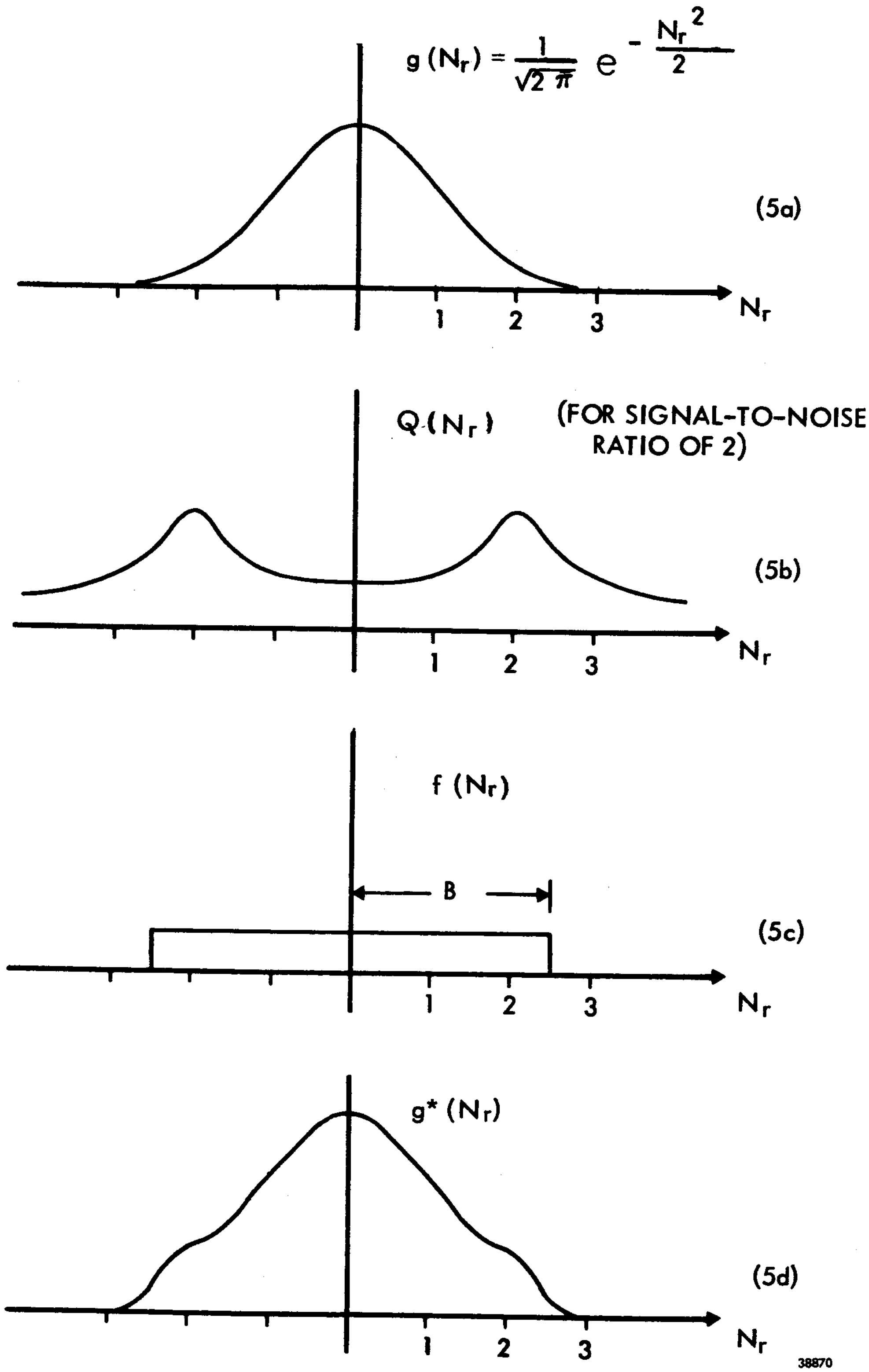



Figure 5 Waveforms for Development of Modified Noise Distribution

variance sufficiently to make the monte carlo calculations practical. Other techniques for modifying the noise distribution could surely be found to make calculations for higher signal-to-noise ratios practical. But this was not pursued further because the trend of the results is apparent without these further calculations.

The "weighting factors," $h(N_i)/h^*(N_i)$ of equation (4.9) are calculated as follows (for n = 2, m = 6). Let A be the fraction of the bit-noises, $N_r$, selected (on the average) from the flat distribution $f(N_r)$ of Figure 5(c), and 1-A the fraction selected from the gaussian distribution $g(N_r)$ of Figure 5(a). Then

$$g^*(N_r) = (1-A)g(N_r) + Af(N_r) \qquad \text{and}$$

$$\frac{g^*(N_r)}{g(N_r)} = (1-A) + A \frac{f(N_r)}{g(N_r)}$$

$$= \begin{cases} \dfrac{A}{B}\sqrt{\pi/2}\left[\dfrac{1-A}{A}\sqrt{\dfrac{2}{\pi}}\, B + e^{N_r^2/2}\right] & |N_r| < B \\[2ex] \dfrac{A}{B}\sqrt{\pi/2}\left[\dfrac{1-A}{A}\sqrt{\dfrac{2}{\pi}}\, B\right] & |N_r| > B \end{cases}$$

and:

$$\frac{h^*(N)}{h(N)} = \prod_{k=1}^{12} \frac{g^*(N_{rk})}{g(N_{rk})}$$

The index of the multiple product goes to 12 because we are considering the case of two 6-bit words (i.e., n = 2, m = 6).

The IBM 7090 computer programs used for the computation of error probabilities are presented and discussed in Appendix V.

## Chapter 5

## INTERPRETATION OF RESULTS

Computed results are presented in Appendix VI. As discussed in Chapter 3, these are the minimum attainable word-error probabilities for $n = 2$ when any two waveforms, $g_1(t)$ and $g_2(t)$, with equal energies (see equation (3.5)) and with finite-time correlation coefficient, $\lambda$, of -1 (see equation (3.6)) are used to represent the binary character of the PCM codes. These results can easily be used to determine the minimum attainable word-error probabilities when <u>any</u> two waveforms, $f_1(t)$ and $f_2(t)$, are used to represent this binary character of the codes. It is neither necessary for $f_1(t)$ and $f_2(t)$ to have the same energies nor for their finite-time correlation coefficient, $\lambda$, to be -1. In fact, their correlation coefficient, $\lambda$, need not even be determinate.

Consider two arbitrary waveforms, $f_1(t)$ and $f_2(t)$ of duration $T_B$, used for representing the binary character of the PCM waveforms. There is associated with any such pair of waveforms a "correlation" parameter, $\alpha$, defined as follows:

$$\alpha = \frac{2\int_{T_B} f_1(t)\, f_2(t)\, dt}{\int_{T_B} \left[f_1(t)^2 + f_2(t)^2\right] dt} \tag{5.1}$$

This parameter is identical to $\lambda$ when the two waveforms have equal energies. If we subtract from both $f_1(t)$ and $f_2(t)$ the waveform $s(t) = 1/2\left[f_1(t) + f_2(t)\right]$ we obtain new functions, $g_1(t)$ and $g_2(t)$, which have equal energies and a $\lambda$ (or $\alpha$) of -1.

$$g_1(t) = \frac{1}{2}\left[f_1(t) - f_2(t)\right]$$

$$g_2(t) = -\frac{1}{2}\left[f_1(t) - f_2(t)\right]$$

This is a linear reversible operation which could be performed on the received noisy waveforms to convert the signal portions of the received waveforms from $f_i(t)$ to $g_i(t)$, but without altering the (added) noise portion of the waveforms. Then we could operate on these converted (noisy) waveforms with the minimum-error demodulator and obtain the error probabilities of Appendix VI. Since the conversion operation is reversible, these error probabilities represent the minimum attainable error probabilities for the original, received waveforms vs. the signal-to-noise ratio of the <u>converted</u> waveforms. But since the conversion does not alter the noise, the change in signal-to-noise ratio due to the conversion operation is just the square root of the ratio of the average power in the signal portions of the original and converted waveforms. We assume that "yes" bits and "no" bits (i. e., $f_1(t)$ and $f_2(t)$) occur with equal frequency since this assumption is implicit in the results of Appendix VI as a consequence of assuming the data distribution of Figure 4. Then the average power of the original signal is

$$S_f^2 = \frac{1}{2}\left\langle f_1(t)^2 + f_2(t)^2 \right\rangle$$ where $\langle X \rangle$ indicates the average over one bit time, $T_B$, of X.

The average power of the converted signal is

$$S_g^2 = \frac{1}{2}\left\langle g_1(t)^2 + g_2(t)^2 \right\rangle = \frac{1}{4}\left\langle \left[f_1(t) - f_2(t)\right]^2 \right\rangle$$

$$= \frac{1}{4}\left\langle f_1^2(t) - 2\, f_1(t)\, f_2(t) + f_2^2(t) \right\rangle = \frac{1}{2}\left(S_f^2 - \alpha S_f^2\right)$$

and hence:

$$\frac{\text{original signal-to-noise ratio}}{\text{converted signal-to-noise ratio}} = \left(\frac{2}{1-\alpha}\right)^{1/2}$$

Therefore the results of Appendix VI may be applied to any binary PCM waveforms whatever by simply multiplying the S/N values by $\sqrt{\dfrac{2}{1-\alpha}}$.

The computed results for n = 2, m = 6 are presented graphically in Figure 6. This figure shows the minimum-attainable word-error probability, vs. signal-to-noise ratio (normalized by $\sqrt{1-\alpha}$) when two noisy received (6-bit) binary PCM code words are used in the demodulation of one of them if the correlation coefficient between the data samples represented by the code words is $\rho$ and the data has the gaussian amplitude distribution of Figure 4. Results are shown for $\rho = 0$, 0.5, 0.7, 0.9, 0.95, and 0.98. Results are also shown for completely random bits - i.e., no interbit dependence. (Note that for gaussian data there is interbit dependence due to the data amplitude distribution, even for $\rho = 0$.)

The results for n = 2, m = 3, presented graphically in Figure 7, indicate greater available gains from high data correlation for three-bit words than those indicated in Figure 6 for six-bit words. This is not surprising when we consider, for these two cases, the relative probabilities of two correlated data samples falling in the same quantization interval. Consider the conditional probability density function, $p(Y_2 | Y_1)$, of the second sample, $Y_2$, when the value of the first sample, $Y_1$, is given. If $Y_1$ and $Y_2$ are gaussian variates with zero mean, variance $\sigma^2$, and correlation coefficient $\rho$, then:

$$p(Y_2 | Y_1) = \frac{1}{\sqrt{2\pi}\,\sigma_1} \exp\left(- \frac{(Y_2 - \rho Y_1)^2}{2\sigma_1^2}\right) \qquad (5.2)$$

where $\sigma_1^2 = \sigma^2(1-\rho^2)$

For the assumed distribution of Figure 4 we have

$$\sigma = \frac{2^m}{5.2} I$$ where I is the width of a quantization interval.

Hence the mean of the gaussian conditional distribution of $Y_2$, given $Y_1$, is $\rho Y_1$, and the variance is:

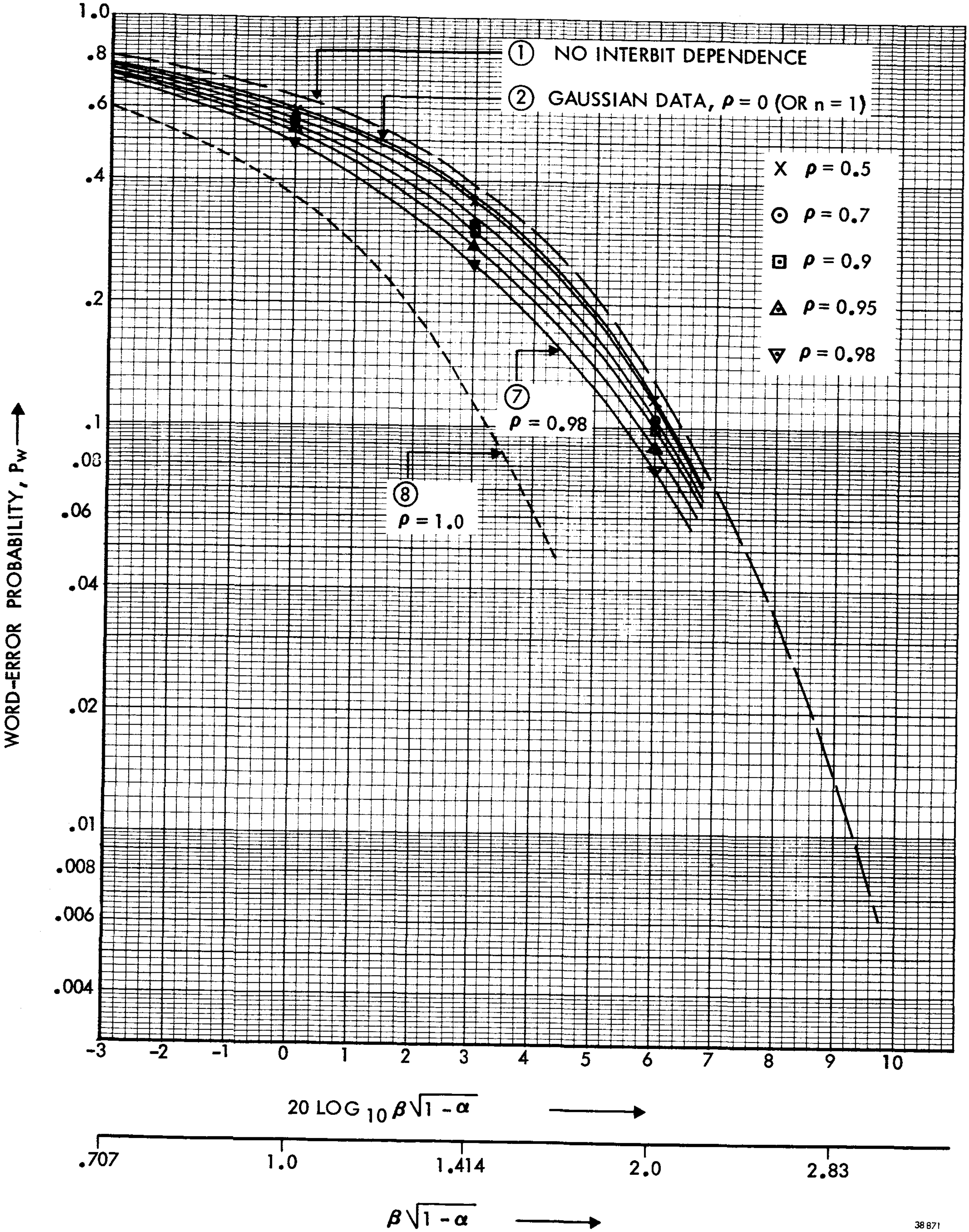



Figure 6 Word Error Probabilities vs Signal-to-Noise Ratio for n = 2, m = 6

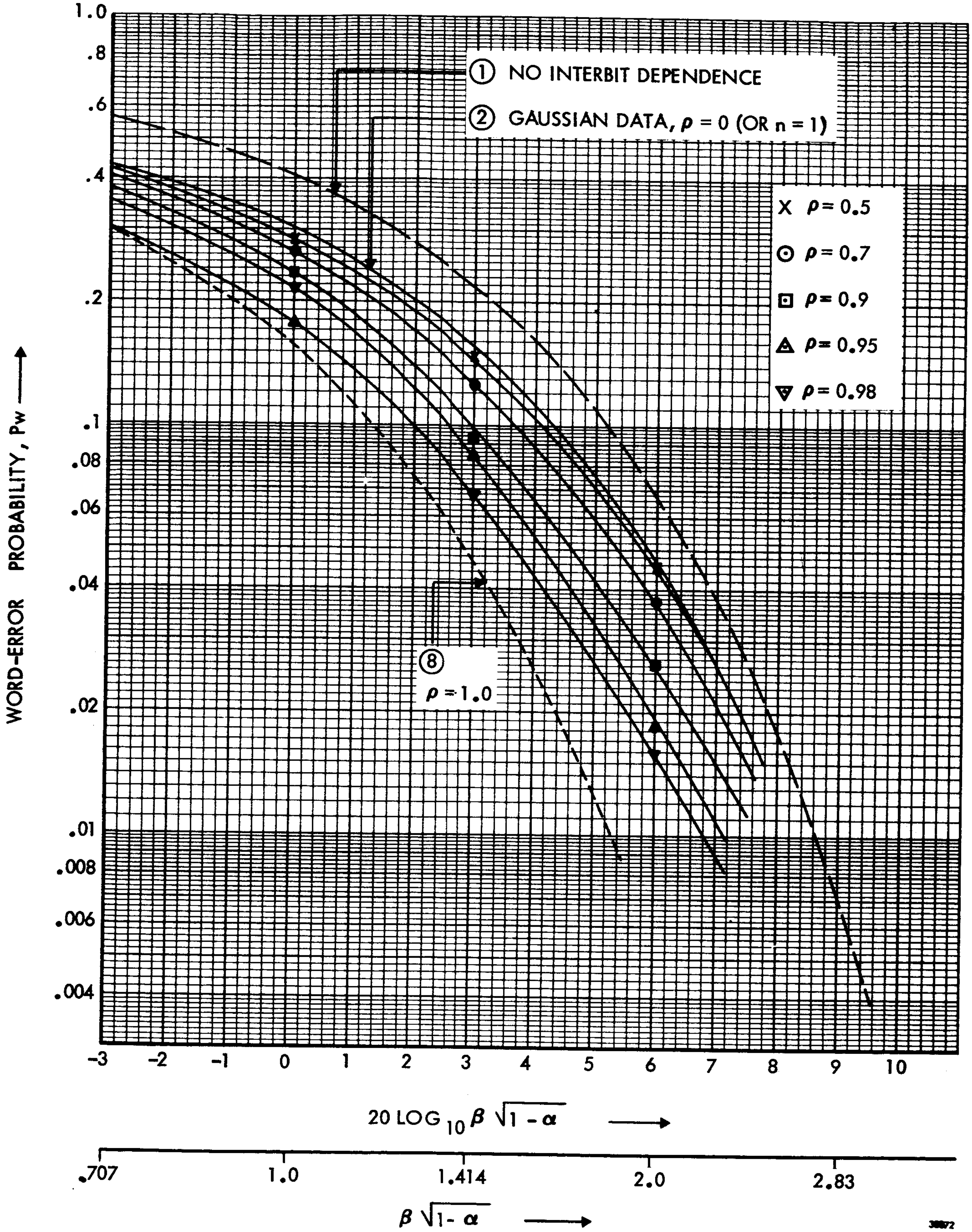



Figure 7 Word Error Probabilities vs Signal-to-Noise Ratio for n = 2, m = 3

for m = 3: $\sigma_1 = 1.54 \sqrt{1-\rho^2}\ I = 0.67\ I$ for $\rho = .9$
$= 0.31\ I$ for $\rho = .98$

for m = 6; $\sigma_1 = 12.3 \sqrt{1-\rho^2}\ I = 5.4\ I$ for $\rho = .9$
$= 2.5\ I$ for $\rho = .98$

Therefore the probability of $Y_2$ and $Y_1$ falling in the same quantization interval is considerably greater for three-bit quantization than for 6-bit quantization. Or, in general, following the same reasoning, the relative dependence between quantized samples is less for finer quantization (i.e., higher m). Consequently the gains available by use of such dependence would be expected to decrease for increased m as is indicated by comparing Figures 6 and 7.

Curve 1 of Figure 6 is for completely random PCM waveforms. Curve 2 is for data with the gaussian amplitude distribution of Figure 4, but with no correlation between data samples. For this case, the inverse probability $p(y_{2(p)} | z_1, z_2)$ becomes:

$$p(y_{2(p)} | z_1, z_2) = K_b(z) \exp\left(\frac{2}{K^2} \int_{T_W} z_2(t) y_{2(p)}(t)\, dt\right) f\left(Y_{2(p)}\right) \quad (5.3)$$

where $K_b(z)$ is constant with respect to $y_2$.

This expression is identical to that obtained for $p(y_{(p)} | z)$ for n = 1. Therefore curve 2 also represents the word-error probability for n = 1, m = 6. The factor $f(Y_{2(p)})$ represents the relative a priori probabilities of the quantized data values, Y, and is not dependent upon signal power or noise power. Therefore by the same reasoning as that used in Chapter 2 (page 16) we may conclude that by using two data samples with unity correlation coefficient (i.e., $\rho = 1.0$) a power gain of 3 db is made possible relative to that required for one word (or for $\rho = 0$). The resulting curve is curve 8 of Figure 6.

The results of Figure 6 for correlation coefficient of 0.5, 0.7, 0.9, 0.95, and 0.98 do not indicate as much power gain due to use of such high data correlation as might have been anticipated. This appears particularly true for word-error probabilities less than 0.1. For $\rho = 0.98$ the power gain is a little more than 1 db for $P_W$ of 0.4; it is approximately 1 db for $P_W$ of 0.2; it is less than 1 db for $P_W$ of 0.1, and appears to continue to decrease as $P_W$ decreases.

But in determining the power gains available by use of data statistics we have assumed that the word-error probability is the "performance parameter" which is specified (i.e., fixed). If some other performance parameter such as rms error is fixed, the available gains may be different from those determined with fixed word-error probability. Of course, as discussed in the Introduction, the (demodulation) operation should be optimized for the performance parameter of interest. But it is not unlikely that an operation optimized for one performance parameter may give considerably improved performance for some other performance parameter. More specifically, the optimum demodulation operation which we have optimized for word-error probability may give considerably improved rms error performance. Some qualitative indication of this may be obtained as follows from the computed word-error probabilities for $n = 2$, $m = 3$ (presented graphically in Figure 7).

The results of Figure 7 for three-bit words may be interpreted as the minimum probabilities of error in the three most significant bits (i.e., the "upper half") of the six-bit words when use is made of only the "upper half" of the received words in the demodulation. But these error probabilities cannot be lower than the minimum probabilities of error in the three most significant bits when use is made of the entire received words. Hence the minimum probabilities of error in the three most significant bits of six-bit words is equal to or less than the error probabilities obtained from Figure 7.

Therefore, for $\beta\sqrt{1-\alpha} = 2$ for example, from Figure 6 we see that a correlation coefficient of 0.98 reduces the probability of error in a six bit word from 0.13 to 0.078 - that is, by a factor of almost 2. But from Figure 7 we see that the same correlation coefficient reduces the probability of error in the three most significant bits by a factor of 4 or more. Since errors in the most significant bits result in larger error amplitudes than do errors in the least significant bits, the above indicates that the gains afforded by use of a priori data statistics may be greater for a specified error amplitude parameter (such as rms error) than for specified error probability.

The above consideration also suggests a method for obtaining approximate results for $m = m_2$ from computed results for $m = m_1$ where $m_1 < m_2$. From equation (5.2) and the discussion following it we see that the correlation coefficient, $\rho$, can be as high as 0.9 without inducing much dependence between the lowest order bits of the 6-bit codes representing the two words. This suggests that for $\rho \leq 0.9$, errors probably occur in the low order bits with about the same frequency as with no interbit dependence. If we assume this to hold for the three lowest order bits and use the computed error probabilities for three-bit words (Figure 7) for the three highest order bits we obtain:

$$P_{W\rho 6} = 1-(1-P_E)^3 (1-P_{W\rho 3}) \qquad (5.4)$$

where $P_{W\rho 6}$ = minimum word-error probability for two 6-bit words ($n = 2$, $m = 6$) with correlation coefficient, $\rho$.

$P_{W\rho 3}$ = minimum word-error probability for two 3-bit words ($n = 2$, $m = 3$) with correlation coefficient, $\rho$.

$P_E$ = minimum bit-error probability with no interbit dependence.

Or, more generally, for estimating the minimum word-error probabilities $P_W$ for two $\ell$-bit words from computed results for two m-bit words:

$$P_{W\rho\ell} = 1 - (1-P_E)^{\ell-m} (1-P_{W\rho m}) \qquad (5.5)$$

The relation (5.4) matches very well with the computed results for 3-bit and 6-bit words. Equation (5.5) should give even better results for predicting word error probabilities for $\ell > 6$ and $m = 6$ since the interbit dependence of bits of orders lower than those of the six most significant bits of each word is almost completely unaffected by data correlation coefficients of 0.98 or less.

The reasoning employed to conclude that a 3 db power gain is made possible by using two words (i.e., $n = 2$) when $\rho = 1.0$ can be employed to conclude that an additional 3 db gain is made possible each time n is doubled. The resultant total gain, $G_r$, for n words with $\rho = 1.0$ is:

$$G_r = G_o \left[ 1 + \sum_{i=1}^{n-1} (G_i - 1) \right]$$

where $G_o$ is the gain available from use of the a priori amplitude distribution of the data values. (That is, the power gain between curves 1 and 2 of Figure 6.)

$G_i$ is the power gain, relative to the power required for $n = 1$ (or equivalently for $\rho = 0$), for each of the n-1 additional words if used one-at-a-time. That is, $G_i = 2$ for $\rho = 1$.

This expression also applies, of course, for $\rho = 0$ in which case $G_i = 1$. Whether or not it applies for $0 < \rho < 1$ (i.e., $1 < G_i < 2$) is not known. If it were applicable it could be used in conjunction with the results of Figure 6 to estimate the total gain available by making use in the demodulation process of an arbitrarily large number of received words with any specified data correlation. For example if the data samples have the correlation coefficients determined in Chapter 2 for 3rd order Butterworth data and the word error probability is specified to be 0.1, we would find that the available gain for very large n is about 1 db. This assumes that, as indicated in Figure 6, $G_i$ approaches unity rapidly as $\rho$ decreases.

Since monte-carlo computations involve model sampling and statistical estimation of results, rather than rigorous calculation of results, an associated difficulty is that of establishing confidence in the monte-carlo results. A direct calculation of confidence intervals such as that performed in Chapter 4 for the simple estimate, $P_1$, is usually not possible because the probability distribution of an estimate is generally not known. It would be possible to use the computed values of Q (see Appendix VI, Tables 7, 8, 9) to establish confidence to some extent as follows: first, estimate the variance, $\sigma_Q^2$, of the Q's by the estimate $s^2 = \frac{1}{L-1} \sum_{i=1}^{L} (Q_i - P_2)^2$. (Replace $P_2$ by $P_3$ and Q by Q $h/h^*$ when $P_3$ is used as the estimate of $P_W$). The confidence of this estimate can be established by assuming $s^2$ to have a "chi-square" distribution of L-1 degrees of freedom (see Reference 2, Chapters 6 and 8), but the estimate should be very close to $\sigma_Q^2$ for the large L's used here. Then the variance $\sigma_P^2$ of the estimate $P_2$ (or $P_3$) is

$$\sigma_P^2 = \frac{\sigma_Q^2}{L} \simeq \frac{s^2}{L}$$

Then, by <u>assuming</u> a probability distribution for $P_2$ (or $P_3$) we can establish confidence intervals as was done for $P_1$ in Chapter 4. But for the results obtained in this report, it is believed that confidence can best be established simply by noting the consistency of the results as presented in Figure 6 (or Figure 7). That is, since the results are known to be unbiased, the consistent spacing and consistent trends of the curves connecting the computed points serves to establish a more meaningful confidence in the results than would a formal treatment such as the one outlined above.

For specified word-error probability less than about 0.1, the computed results indicate that for significant gains to accrue from the use of data redundancy, the correlation coefficients between data samples must be large (i.e., 0.98 or greater) for large numbers of samples. This apparently becomes truer as the specified word-error probability becomes lower. Whether or not sufficient redundancy is present in transmitted data must be determined by examining typical data and error requirements. There are surely cases where data sample rates are high enough so that sufficient redundancy is present even though such high sampling rates may not be necessary for the data recovery accuracies required. In such cases the existence of the high redundancy in the data may not even be recognized. But data redundancy which is not known a priori at the receiver cannot be used to improve the demodulation operation.

## Chapter 6

## ALTERNATE TECHNIQUES AND POSSIBLE FURTHER INVESTIGATIONS

The results obtained here and discussed in Chapter 5 indicate only in a general way the values of data correlation and specified error probabilities for which the use of a priori data statistics and minimum-error demodulation may be of significant value. Investigation of data statistics for classes of sampled data measurements of general, or specific, interest may reveal very high correlation coefficients (greater than 0.98) between samples. In such cases the results should be extended to include the higher values of $\rho$. At any rate there may be circumstances where, due perhaps to very high data correlation or high tolerable error probability, the use of a practical implementation of the minimum-error demodulator is desired.

The most obvious implementation, suggested by combining equations (2.3) and (3.3), is to use $n2^m$ finite-time correlators (i.e., one for each of the possible $2^m$ transmitted word waveforms in each of the n word time slots considered) to evaluate the factors of the form $h(z_i - y_{i(p)})$, then combine these factors and $g(y_1, ---, y_n)$ by appropriate exponentiation, summing, and weighting, in accordance with (2.3) for each $y_{j(p)}$, and select the $y_{j(p)}$ giving the largest result. But, as indicated by equation (3.4), the correlation operations may be done bit-by-bit. And since the $z_{ir}$'s are given and $y_{i(p)r}$ has only two possible forms, i.e. $f_1(t)$ or $f_2(t)$, all of the required correlations can be performed by one pair of correlators with references $f_1(t)$ and $f_2(t)$. These correlators may then operate on the $z_{ir}$'s to produce the terms on the right hand side of equation (3.4), which are of the form

$$\int_{T_B} z_{ir} y_{i(p)r} \, dt$$

But since there are only two possible $y_{i(p)r}$'s (i.e., $f_1(t)$ and $f_2(t)$) and only mn (given) $z_{ir}$'s involved in a demodulation, there are only 2mn possible results for the above operation. Hence these 2mn resulting values may be stored, with appropriate indexing, and combined by digital computer or similar device according to equations (3.4), (3.3), and (2.3) to produce evaluations of $p(y_{j(p)} | z_1, ---, z_n)$ for each p. Therefore it is not a completely unreasonable task to realize such a demodulator in principle. But with present computing speeds only a few words (demodulations) per second can be handled for $n = 2$, and even fewer for larger n.

Because of this limitation in operating speed and the complexity of this "optimum" (minimum-error) demodulator, a simpler "approximate" implementation of the demodulator may be desirable. One such approximation may be derived by a modification of the optimization procedure as follows. Suppose that instead of being given the n received noisy signals, $z_1, ---, z_n$, we are given the received signal, $z_n$, and the n-1 previous transmitted signals, $y_1, ---, y_{n-1}$, and we wish to make a minimum-error probability estimate of $y_n$. That is, we wish to select the $y_{n(p)}$ which maximizes the conditional probability distribution, $p(y_{n(p)} | y_1, ---, y_{n-1}, z_n)$. The $y_{n(p)}$ which maximizes this conditional distribution would be a <u>better</u> estimate of the transmitted $y_{n(s)}$ than would the estimate which maximizes $p(y_{n(p)} | z_1, ---, z_n)$ since the y's are not contaminated by noise as are the z's, but $y_1, ---, y_{n-1}$ are not known at the receiver. However, we may assume that estimates of these are known from the previous n-1 demodulations and will be used instead of the actual y's. (Since the objective here is to derive a simple implementation of a "nearly optimum" demodulator or estimator, we have tacitly assumed that use is made of $z_n$ and estimates of y's occurring in time <u>prior</u> to $z_n$, but not after $z_n$. This is not a necessary limitation since by performing successive (iterative) demodulation operations, use can be made of previous estimates of y's occurring both before and after $z_n$ if desired.) Therefore we must determine the operations on $y_1, ---, y_{n-1}, z_n$ necessary to evaluate $p(y_{n(p)} | y_1, ---, y_{n-1}, z_n)$ so that we may select the $y_{n(p)}$ which for any given $y_1, ---, y_{n-1}, z_n$ gives the maximum value.

$$p(y_{n(p)} | y_1, ---, y_{n-1}, z_n) = \frac{s(y_1, ---, y_{n-1}, y_{n(p)}, z_n)}{w(y_1, ---, y_{n-1}, z_n)}$$

$$= \frac{u(y_1, ---, y_{n-1}, y_{n(p)})\, v(z_n | y_1, ---, y_{n-1}, y_{n(p)})}{w(y_1, ---, y_{n-1}, z_n)}$$

$$= \frac{g(y_{n(p)} | y_1, ---, y_{n-1})\, v\,(z_n | y_1, ---, y_{n-1}, y_{n(p)})}{r(z_n | y_1, ---, y_{n-1})} \qquad (6.1)$$

But $v(z_n | y_1, ---, y_{n-1}, y_{n(p)}) = v\,(z_n | y_{n(p)}) = h\,(z_n - y_{n(p)})$

where $h(n)$ is the probability density of the additive noise, $n(t)$. Since $r(z_n | y_1, ---, y_{n-1})$ is not dependent upon $y_{n(p)}$ it can be considered a constant, $1/K_c$ for the maximization with respect to $y_{n(p)}$. And since each set of waveforms, $y_1, ---, y_n$, represents uniquely a corresponding set of quantized data samples, $Y_1, ---, Y_n$, the conditional probability distribution $g(y_{n(p)} | y_1, ---, y_{n-1})$ may be replaced by $f\,(Y_{n(p)} | Y_1, ---, Y_{n-1})$. Hence, for additive band-limited gaussian noise of spectral height $K^2$ (see page 69 of Reference 14):

$$p(y_{n(p)} | y_1, ---, y_{n-1}, z_n) = K_c(z)\, h(z_n - y_{n(p)})\, f\,(Y_{n(p)} | Y_1, ---, Y_{n-1})$$

$$= K_d(z) \exp\left(-\frac{1}{K^2}\int_{T_W} y^2_{n(p)}\, dt\right) \exp\left(\frac{2}{K^2}\int_{T_W} y_{n(p)} z_n\, dt\right)$$

$$f(Y_{n(p)} | Y_1, ---, Y_{n-1}) \qquad (6.2)$$

If the two waveforms, $f_1(t)$ and $f_2(t)$, used to represent the binary code have equal energies, the first exponential factor in (6.2) will be the same for any $y_{n(p)}$ and can be absorbed into the constant, $K_d(z)$. ($K_d(z)$ is a function of $z_n$, but not of $y_{n(p)}$). The exponent of the second exponential factor can be obtained for any $y_{n(p)}$ by a finite-time correlator with reference $y_{n(p)}$ and input $z_n$. It should be noted that the amplitude of the reference $y_{n(p)}$ must be equal to the amplitude of the signal component of $z_n$. Therefore, signal amplitude as well as noise spectral height, $K^2$, (proportional to receiver noise figure) must be determined in order to obtain the exponent of the second factor. The last factor has, in general, a different value for each $y_{n(p)}$. It is a value whose logarithm must be added to the exponent obtained by the finite-time correlator with reference $y_{n(p)}$ (or, equivalently, a value which must multiply, or "weight," the result obtained by exponentiating the correlator output). If the values of the first exponential factor of (6.2) are not constant for all $y_{n(p)}$, they must be determined for each $y_{n(p)}$ and then treated in the same way as the values of the last factor.

The values of the last factor, $f(Y_{n(p)} \mid Y_1, ---, Y_{n-1})$, can be obtained to very good accuracy, for gaussian data, by evaluating, for each $Y_{n(p)}$, a gaussian probability density function with mean and variance determined by $Y_1, ---, Y_{n-1}$. Expressions for determining both the mean and variance of the gaussian density function may be obtained in either of two ways. The first method is to use the technique outlined in Chapter 9 of Reference 9 for calculating the "regression function" of $Y_{n(p)}$, which is simply the mean of $f(Y_{n(p)} \mid Y_1, ---, Y_{n-1})$ and is a linear function of $Y_1, ---, Y_{n-1}$, the coefficients being determined by the correlation coefficients of the data samples $Y_1, ---, Y_{n-1}$. Also outlined in Reference 9, is a technique which can be used to calculate the variance of $f(Y_{n(p)} \mid Y_1, ---, Y_{n-1})$ from the correlation coefficients between data samples.

The second method for getting expressions for the mean and variance of $f(Y_{n(p)} \mid Y_1, ---, Y_{n-1})$ for gaussian data is by optimum linear prediction to minimize mean-square-error, noting that for gaussian signals the optimum linear predictor is the optimum predictor (see page 275 of Reference 6) and that the prediction which minimizes mean-square error must be the mean of the (conditional) probability distribution. Consider the problem of linearly predicting a value, $Y_2$, of a signal, $Y(t)$, when some value, $Y_1$, is known and when $Y_1$ and $Y_2$ are separated in time by an interval $\tau$. Let the predicted value of $Y_2$ be $\hat{Y}_2$. Then $\hat{Y}_2 = WY_1$ where W may be a function of $\tau$, but not of $Y_1$. The prediction error is $e = Y_2 - \hat{Y}_2 = Y_2 - WY_1$ and the mean-square error is

$$\overline{e^2} = \overline{Y_2^2 - 2WY_1Y_2 + W^2Y_1^2}$$

Let:
$$\overline{Y_1} = \overline{Y_2} = 0$$

$$\overline{Y_1^2} = \overline{Y_2^2} = \sigma^2$$

$$\overline{Y_1Y_2} = R(\tau) = \rho(\tau)\,\sigma^2$$

Then:
$$\overline{e^2} = \sigma^2 - 2W\rho\,\sigma^2 + w^2\sigma^2$$

To minimize $\overline{e^2}$ with respect to W

$$\frac{\partial \overline{e^2}}{\partial W} = -2\,\rho\,\sigma^2 + 2\,W\sigma^2 = 0$$

or $W = \rho(\tau)$

Therefore $\rho(\tau)\,Y_1$ is the mean value of the conditional probability distribution $f(Y_2 \mid Y_1)$. The variance of this distribution is the mean-square-error of this prediction or:

$$\overline{e^2} = \sigma^2 - 2\rho^2\sigma^2 + \rho^2\sigma^2 = \sigma^2(1-\rho^2)$$

This procedure can be applied for any number of known points, $Y_1, ---, Y_{n-1}$.

The expression for $p(y_{n(p)} \mid y_1, ---, y_{n-1}, z_n)$ given in equation (6.2) indicates that we can determine the $y_{n(p)}$ which maximizes this expression by cross correlating the received waveform, $z_n(t)$ with each $y_{n(p)}$, obtain an exponential of each correlator output, and weight each of these with the corresponding conditional probability $f(Y_{n(p)} | Y_1, ---, Y_{n-1})$. (Additional weighting is necessary if the $y_{n(p)}$'s have unequal energies.) The maximum result thus obtained corresponds to the most likely $y_{n(p)}$ for any given $z_n$ and $y_1, ---, y_{n-1}$.

Error probabilities for this "near-optimum" demodulator have not been computed, but their computation by the monte-carlo technique would be considerably simpler and require much less computer time than for the minimum-error demodulator, and the computation time does not increase so rapidly as n gets large.

Another "approximation" of minimum-error demodulation[1] which makes use of data correlation would be the detection of individual bits (rather than words) after averaging the signals for that bit-time over an appropriate number of sequential samples. The "appropriate number " of samples used would be different for bits of different order as may be seen from the following qualitative discussion. It will also vary with signal-to-noise ratio and with correlation between data samples.

Certain bits, or binary digits, (such as the most significant bit) of ordinary binary PCM code words representing data are not likely to change for several sequential samples if there is high correlation between data samples. If the detection of these bits, contaminated by additive gaussian noise of zero mean, employs some linear smoothing process, such as a correlation detector, the noise component out of the smoothing device is gaussian and its mean-square value decreases as the smoothing time increases. Therefore if a bit does not change for several samples, a better detection could be accomplished by smoothing that bit over several samples. But if the bit is smoothed over too many samples it will probably change once or more during the "smoothing time" and the resulting detection may be poorer than for smoothing over one sample only. There will

---

1 This approximation was suggested by Professor Rauch.

in general be an optimum number of samples over which each bit of a PCM code should be smoothed for given data correlation and signal-to-noise ratio. A feasible method for determining these optimum numbers of samples is not readily apparent. One conceivable, but crude, approach is to assume values of correlation coefficients $\rho(kT_s)$ for samples separated by k sample periods $T_s$, and of smoothed output signal-to-noise ratio, S/N (for smoothing over a single sample). Then calculate the probability of error for each bit position of the code (i. e., most significant bit, second most significant bit, etc.) for smoothing over two samples, three samples, etc., to determine the optimum number of samples for minimizing the probability of error.

The bit error probability for the $r^{th}$ bit after smoothing over n samples is

$$P_{Enr} = \sum_{j=1}^{2^n} P_{nj} Q_{jr}$$

Where $Q_{jr}$ is the probability of occurrence of the $j^{th}$ combination of bit values (1's and 0's) for the $r^{th}$ bit in the n (coded) samples, and $P_{nj}$ is the bit-error probability after smoothing over this $j^{th}$ combination of bit values.

For example, for n = 2 there are four possible combinations of bit values for any given bit. These are 00; 01; 10; 11, for which we let j = 0; 1; 2; 3 respectively. Let S/N be the ratio of peak signal to rms noise out of the smoothing device (e. g., aperture filter, or finite-time correlator) after smoothing over one sample only. Then for bit combinations 00 and 11 the value of $P_{2j}$ after smoothing over two samples is (for gaussian noise):

$$P_{20} = P_{23} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\sqrt{2}\, S/N} e^{-x^2/2}\, dx$$

since peak signal out remains the same while rms noise decreases by $\sqrt{2}$. These values can be found from gaussian tables for any S/N. For bit combinations 01 and 10 the value of $P_{2j}$ after smoothing over two samples is $P_{21} = P_{22} = 0$ since peak signal out is zero.

The corresponding values of $Q_{jr}$ are more difficult to calculate. They depend upon the statistical dependence between data samples and will be different for different bit positions, r. For the most significant bit (r = 1) the probabilities of occurrence for 00 and 11 (i. e. for j = 0 and 3) are:

$$Q_{01} = \int_{-\infty}^{Y_h} \int_{-\infty}^{Y_h} p(Y_1, Y_2)\, dY_1\, dY_2$$

$$Q_{31} = \int_{Y_h}^{\infty} \int_{Y_h}^{\infty} p(Y_1, Y_2)\, dY_1 dY_2$$

where $Y_1$ and $Y_2$ are the first and second data sample values, $Y_h$ is the half-scale data value, and $p(Y_1, Y_2)$ is the joint probability density for data samples $Y_1$ and $Y_2$. If the data is assumed gaussian, the $Q_{jr}$'s will be determined by the correlation coefficient, $\rho(T_s)$. The probabilities of occurrence for 01 and 10 (i. e. for j = 1 or 2) are:

$$Q_{11} = \int_{Y_h}^{\infty} \int_{-\infty}^{Y_h} p(Y_1, Y_2)\, dY_1 dY_2$$

$$Q_{21} = \int_{-\infty}^{Y_h} \int_{Y_h}^{\infty} p(Y_1, Y_2) \qquad dY_1\, dY_2$$

For the second most significant bit ($r = 2$) the double integration must cover four separate regions of the $Y_1 Y_2$ space for each $Q_{j2}$. That is, for $Q_{02}$ the integration must cover the regions where the $Y_1$ and $Y_2$ values are either less than quarter-scale or between half-scale and three-quarter-scale (i. e., the regions where the second most significant bit of both coded samples is "0"); for $Q_{12}$ the integration must cover the region where the $Y_1$ value is either less than quarter-scale or between half-scale and three-quarter scale while the $Y_2$ value is either greater than three-quarter scale or between quarter-scale and half-scale (i. e., the regions where the second most significant bits of the first and second coded samples are "0" and "1" respectively); etc.

In principle this approach can be extended to any number of samples and any number of bits, but the computations rapidly become very complex. A serious attack upon the problem of determining the optimum number of samples over which the bits should be smoothed might well yield a better approach. The optimum number for any bit (order) would surely be dependent upon S/N and the $\rho$'s, thus requiring knowledge at the receiver of signal-to-noise ratios and data correlation. But such knowledge is also required for the optimum (minimum-error) demodulator and for the "near-optimum" demodulator discussed earlier. Such knowledge appears to be necessary in order to combat thresholding.

Thresholding is considered here to be the phenomena which causes the mean-square error of the demodulated signal to increase faster than the input mean-square-noise-to-signal ratio. For systems employing both pulse-code-modulation and some form of rf-modulation (i. e., AM, PM, FM) thresholding will, in general, result from both. But the minimum-error demodulator completely avoids thresholding due to rf-modulation. This is apparent since different types of rf-modulation merely require different waveforms, $f_1(t)$ and $f_2(t)$, for representing the binary character of the coded signals. But we have found in Chapter 5 that the statistical results obtained by the minimum-error demodulator (for error probability <u>or</u> error amplitudes) is dependent only upon the parameter $\alpha$, defined by equation (5.1); and for any $\alpha$ (between -1 and +1) we can find waveforms $f_1(t)$ and $f_2(t)$ corresponding to an AM-modulation, which exhibits no thresholding due to the rf-modulation.

That thresholding due to the pulse-code-modulation is combatted somewhat by the minimum-error demodulator can be seen qualitatively from the result (also discussed in Chapter 5) that this demodulator reduces errors primarily in the high order bits of the codes. Quantitative data on PCM-thresholding could be obtained by monte-carlo computations similar to those reported here, but computing error amplitudes, (rather than error probabilities) from which any error amplitude parameters such as rms-error or mean-absolute error could be obtained. Such results would also be of value for determining gains available from minimum-error demodulation for specified error amplitude parameters rather than specified error probability. Of course these results would not necessarily represent the maximum theoretical gains available for the specified error amplitude parameter. The determination of such maximum gains would require the derivation of a demodulation technique for optimizing the specified error amplitude parameter.

Another worthwhile extension of the results reported here would be the determination of a simpler method for calculating the low values of word-error probability $P_W$ (e.g., for $P_W < 0.1$) since these computations require very large computation time if the monte-carlo method is used. An approximate analytical method might be determined by presuming that for low error probabilities most of the erroneous words will have only one bit in error. Another approach might be to presume that the results obtained for the "near-optimum" demodulator discussed above are, for low error probability, very close to the results for the optimum demodulator. This seems a reasonable assumption since for low error probability most of the "previous estimates" used in the near-optimum demodulator should be correct, particularly if the word error probability is low when no use is made of data statistics. Monte-carlo computations for the near-optimum demodulator would require much less time than for the optimum demodulator. This would permit extension of the results of Figures 6 and 7 to higher values of $\beta \sqrt{1 - \alpha}$.

## APPENDIX I

## MINIMUM BIT-ERROR PROBABILITIES FOR DEMODULATION OF RANDOM IDEALIZED PCM/FM

The statement made in Chapter 1 (page 6 ) that "---the most probable y(t) can be determined one-bit-at-a-time ---" assumes that during any bit time the PCM signal is one of two waveforms, $f_1(t)$ or $f_2(t)$, which are known independently of the waveforms existing during other bit times. For conventional PCM/FM in which a single oscillator is frequency modulated by a binary signal, this assumption is not true for non-integer deviation ratios since the phase of the sinusoid during any bit-time depends on the binary modulating signal during other bit-times. (Deviation ratio is the ratio of total frequency deviation to bit rate.) In such a case, the most probable y(t) waveform is not necessarily that determined by selecting the most probable waveform, $y_B(t)$, for each bit-time. But the latter is the selection which minimizes bit-error probability if the entire received waveform, z(t), is used in making each bit decision --- that is if we choose, for each bit-time, the $y_B$-waveform which maximizes $p(y_B | z)$. However, if we assume that the phase of the two possible sinusoids is known for each bit-time, then $p(y_B | z) = p(y_B | z_B)$ and we may make minimum-bit-error-probability decisions one-bit-at-a-time and without reference to the received waveform, z(t), during other bit-times. The bit-error probabilities thus obtained can not be greater than those obtained without the assumption of known phase for the two possible sinusoids. Hence a lower limit on bit-error probability can be established by assuming the phase to be known for each bit-time.

Therefore, for purposes of calculating a lower limit on bit-error probability we may consider the minimum-error PCM/FM demodulator (making no use of dependence between bits) to consist of two sampled, finite-time correlators whose sampled outputs are subtracted. The correlation time is one bit-time and the outputs are sampled at the ends of the correlation times, which are assumed to be coincident with the ends of the bit times. The reference for correlator 1 is a sinusoid of frequency $f_1$ phase coherent with transmitted waveforms of frequency $f_1$, and the reference for correlator 2 is a sinusoid with the same amplitude but of frequency $f_2$ phase coherent with transmitted waveforms of frequency $f_2$. We will let the sampled output of correlator 2 be subtracted (in a comparator) from the sampled output of correlator 1. When this difference (i.e., the comparator output) is greater than zero, the demodulator will indicate that the transmitted waveform was of frequency $f_1$. When the difference is less than zero it will indicate that the transmitted waveform was of frequency $f_2$. We now calculate the probability that this indication will be in error when the transmitted signal is contaminated by independent, additive white gaussian noise of single-sided spectral height $K^2$ volts$^2$/cps (or two-sided spectral height $k^2 = \frac{K^2}{2}$ ).

A finite-time correlator of the type discussed above is a device which forms the product of a reference waveform and an input waveform, and averages this product over the correlation time which in this case is equal to the bit time $T_B$. Since this is a linear operation on the input, we may consider separately the signal component, $y(t)$, and the noise component, $n(t)$, of the input, $z(t) = y(t) + n(t)$. Let us consider the correlator with reference $R_1(t) = C \cos(\omega_1 t + \phi_1)$. The signal component, $y(t)$, of the input may have either of the forms: $y_1(t) = A \cos(\omega_1 t + \phi_1)$ or $y_2(t) = A \cos(\omega_2 t + \phi_2)$. If it is $y_1(t)$, the resulting sampled correlator output will be:

$$S_m = \frac{1}{T_B} \int_o^{T_B} R_1(t) y_1(t)\, dt = \frac{AC}{T_B} \int_o^{T_B} \cos^2(\omega_1 t + \phi_1)\, dt$$

$$S_m = \frac{AC}{2}\left(1 + \frac{1}{T_B}\int_0^{T_B} \cos 2(\omega_1 t + \phi_1)\, dt\right)$$

$$\cong \frac{AC}{2} \qquad \text{if } \frac{\omega_1}{2\pi} \gg \frac{1}{T_B}$$

The output due to signal $y_2(t)$ is:

$$S_u = \frac{1}{T_B}\int_0^{T_B} R_1(t) y_2(t)\, dt = \frac{AC}{T_B}\int_0^{T_B} \cos(\omega_1 t + \phi_1)\cos(\omega_2 t + \phi_2)\, dt$$

$$= \frac{AC}{2T_B}\left(\int_0^{T_B} \cos\left[(\omega_1 + \omega_2) t + (\phi_1 + \phi_2)\right] dt + \int_0^{T_B} \cos\left[(\omega_2 - \omega_1) t + (\phi_2 - \phi_1)\right] dt\right)$$

Similarly we find that for the correlator with reference $R_2(t) = C \cos(\omega_2 t + \phi_2)$ the sampled output due to signal $y_2(t)$ is $S_m$ (with $\omega_1$ replaced by $\omega_2$, and $\phi_1$ by $\phi_2$) and the output due to $y_1(t)$ is $S_u$. That is, for either correlator, the output resulting from the "matched" signal is $S_m$ while that resulting from the "unmatched" signal is $S_u$. Hence, the output of the comparator due to the signal is $S_o = \pm (S_m - S_u)$.

$$\text{But } (S_m - S_u) \cong \frac{AC}{2}\left(1 - \frac{1}{T_B}\int_0^{T_B} \cos\left[(\omega_1 + \omega_2) t + (\phi_1 + \phi_2)\right] dt - \frac{1}{T_B}\int_0^{T_B} \cos\left[(\omega_2 - \omega_1) t + (\phi_2 - \phi_1)\right] dt\right)$$

$$(S_m - S_u) \cong \frac{AC}{2}\left(1 - \frac{1}{T_B}\int_o^{T_B} \cos\ (\omega_2 - \omega_1)\,t + (\phi_2 - \phi_1)\ dt\right)$$

$$\text{if}\quad \frac{\omega_1 + \omega_2}{2\pi} >> \frac{1}{T_B}$$

We now calculate the output due to white gaussian noise.

The sampled output of either correlator due to noise n(t) is:

$$N(T_B) = \frac{1}{T_B}\int_o^{T_B} R(t)\, n(t)\, dt$$

and the noise output of the comparator is:

$$N_m(T_B) - N_u(T_B) = \frac{1}{T_B}\int_o^{T_B} \left[R_1(t) - R_2(t)\right] n(t)\ dt$$

Hence, since $R_1$ and $R_2$ are not random, the variance of the noise output of the comparator is:

$$\sigma_N^{\ 2} = \overline{\left[N_m(T_B) - N_u(T_B)\right]^2}$$

$$= \frac{1}{T_B^{\ 2}}\int_o^{T_B}\int_o^{T_B} \left[R_1(\tau_1) - R_2(\tau_1)\right]\left[R_1(\tau_2) - R_2(\tau_2)\right]\overline{n(\tau_1)n(\tau_2)}\ d\tau_1 d\tau_2$$

$$= \frac{k^2}{T_B^{\ 2}}\int_o^{T_B} \left[R_1^{\ 2}(\tau) + R_2^{\ 2}(\tau) - 2R_1(\tau)\, R_2(\tau)\right]\ d\tau$$

since

$$\overline{\cdot\,(\tau_1)\ n(\tau_2)} = k^2\ \delta(\tau_1 - \tau_2)$$

where $k^2$ is the two sided spectral height of the noise. ($\bar{x}$ is used here to indicate the ensemble average of x.) Therefore:

$$\sigma_N^2 = \frac{k^2}{T_B^2} \int_0^{T_B} \left[ C^2 \cos^2(\omega_1\tau + \phi_1) + C^2 \cos^2(\omega_2\tau + \phi_2) - 2C^2 \cos(\omega_1\tau + \phi_1)\cos(\omega_2\tau + \phi_2) \right] d\tau$$

$$= \frac{C^2k^2}{2T_B^2} \int_0^{T_B} \Big( 2 + \cos 2(\omega_1\tau + \phi_1) + \cos 2(\omega_2\tau + \phi_2) - 2\cos\left[(\omega_1 + \omega_2)\tau + (\phi_1 + \phi_2)\right] - 2\cos\left[(\omega_2 - \omega_1)\tau + (\phi_2 - \phi_1)\right] \Big) d\tau$$

$$\cong \frac{C^2k^2}{T_B} \left( 1 - \frac{1}{T_B} \int_0^{T_B} \cos\left[(\omega_2 - \omega_1)\tau + (\phi_2 - \phi_1)\right] d\tau \right)$$

$$\text{if } \frac{\omega_1}{2\pi} \gg \frac{1}{T_B}$$

$$\text{and } \frac{\omega_2}{2\pi} \gg \frac{1}{T_B}$$

The conditions on the $\omega$'s are satisfied in practice since a bit-interval contains many cycles of the carrier. (The consequence of these conditions not being satisfied is discussed in Appendix II.) Hence the ratio of signal amplitude to rms noise out of the comparator is:

$$\frac{S_o}{\sigma_N} = \frac{S_m - S_u}{\sigma_N} = \frac{A\sqrt{T_B}}{2k} \sqrt{1 - \frac{1}{T_B} \int_0^{T_B} \cos\left[(\omega_2 - \omega_1)\tau + (\phi_2 - \phi_1)\right] d\tau}$$

$$\frac{S_o}{\sigma_N} = \frac{S}{K\sqrt{B}} \sqrt{1 - \left(\frac{\sin(2\pi D + \delta\phi) - \sin\delta\phi}{2\pi D}\right)} \qquad \text{(I-1)}$$

where: $S = A/\sqrt{2}$ = rms value of input signal

$B = \dfrac{1}{T_B}$ = bit rate

$D = \dfrac{\omega_2 - \omega_1}{2\pi B}$ = ratio of total deviation (in cps) to bit rate

$\delta\phi = \phi_2 - \phi_1$ = difference in phase of references at beginning of bit interval

$K^2 = 2k^2$ = one-sided spectral height of noise

The quantity "D" is often defined as the deviation ratio for PCM/FM. The quantity "$\delta\phi$" might, of course, be different for different bit intervals, but it is zero for conventional PCM/FM systems in which a single oscillator is frequency modulated by the binary PCM waveform. For this case, we wish to find the deviation ratio, D, which maximizes the ratio of signal amplitude to rms noise, $S_o/\sigma_N$, out of the comparator, hence minimizing the error probability. The usual maximization procedure yields the optimum deviation ratio.

$$D_{opt} = 0.715$$

The resulting $S_o/\sigma_N$ is:

$$\left(\frac{S_o}{\sigma_N}\right)_{\substack{\text{opt.}\\ \text{PCM/FM}}} = 1.1 \left(\frac{S}{K\sqrt{B}}\right) \qquad \text{(I-2)}$$

An error will be made when the noise out of the comparator exceeds the amplitude of the signal out of the comparator and is of opposite algebraic sign from the signal. Since all operations on the input are linear, and hence the noise remains gaussian, the bit error probability is then

$$P_E = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\frac{S_o}{\sigma_N}} e^{-x^2/2}\, dx \qquad \text{(I-3)}$$

Hence, for optimum deviation ratio of 0.715:

$$\left(P_E\right)_{\substack{\text{opt.}\\ \text{PCM/FM}}} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-1.1\beta} e^{-x^2/2}\, dx \qquad \text{(I-4)}$$

where $\beta = \dfrac{S}{K\sqrt{B}}$

Note from equation (I-1) that if the deviation ratio is any integer, the output signal-to-noise ratio is (with $\delta\phi = 0$):

$$\left(\frac{S_o}{\sigma_N}\right)_{D=n} = \beta \qquad \text{(I-5)}$$

The corresponding bit-error probability is:

$$\left(P_E\right)_{D=n} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\beta} e^{-x^2/2}\, dx \qquad \text{(I-6)}$$

Therefore, the minimum obtainable bit-error probability is the same for any integer value of deviation ratio. It may also be noted that the optimum combination of values for $\delta\phi$ and D (i.e., the combination of values which maximizes I-1) is:

$\delta\phi = \pi$

$D = 0$

This corresponds to phase shift keying (PSK) with phase shift of $\pi$ radians, which is exactly equivalent to suppressed-carrier PCM/AM. The resulting output signal-to-noise ratio and error probability are:

$$\left(\frac{S_o}{\sigma_N}\right)_{PCM/PSK} = \sqrt{2}\,\beta \tag{I-7}$$

$$\left(P_E\right)_{PCM/PSK} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\sqrt{2}\,\beta} e^{-x^2/2}\, dx \tag{I-8}$$

Since 50% of the power in a PCM/AM signal with 100% modulation is in the carrier, the output signal-to-noise ratio and error probability for correlation detection (i. e., maximum-likelihood demodulation) of PCM/AM (non-suppressed-carrier) are:

$$\left(\frac{S_o}{\sigma_N}\right)_{PCM/AM} = \beta \tag{I-9}$$

$$\left(P_E\right)_{PCM/AM} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\beta} e^{-x^2/2}\, dx \tag{I-10}$$

For PCM/FM with random $\delta\,\phi$ resulting from non-synchronous switching between two oscillators of frequencies $\omega_1$ and $\omega_2$, $\delta\,\phi$ is uniformly distributed between $+\pi$ and $-\pi$ radians; and from equation (I-1) we see that

$\left(\frac{S_o}{\sigma_N}\right)^2$ is a function of $\delta\,\phi$ and hence is a random variable, $\left(\frac{S_o}{\sigma_N}(\delta\,\phi)\right)^2$, with symmetrical distribution (that of a sinusoid) about its mean value of $\beta^2$. It may be seen that the maximum amplitude of the variation of $\left( \frac{S_o}{\sigma_N} (\delta \phi) \right)^2$ about $\beta^2$ depends upon the deviation ratio, D, and is zero for D equal to any integer. We may consider the error probability, $P_E (\delta \phi)$, to be a function of $\delta \phi$ and hence also a random variable:

$$P_E (\delta \phi) = \frac{1}{\sqrt{2 \pi}} \int_{-\infty}^{-\frac{S_o}{\sigma_N} (\delta \phi)} e^{-x^2/2} \, dx$$

Now from this expression, since $S_o/\sigma_N$ must be positive, any two values of $\left[ S_o/\sigma_N (\delta \phi) \right]^2$ symmetrically spaced about its mean value, $\beta^2$, will yield two error probabilities whose average is easily shown to be greater than the error probability for $\left[ S_o/\sigma_N (\delta \phi) \right]^2 = \beta^2$. Therefore any symmetrical variation of $\left[ S_o/\sigma_N (\delta \phi) \right]^2$ about its mean, $\beta^2$, will result in a higher average value of $P_E$ than if the variation of $\left[ S_o/\sigma_N \right]^2$ were zero. Hence, for PCM/FM resulting from non-synchronous switching between two oscillators, the lowest possible error probability is obtained with D equal to any integer, and this error probability is:

$$\left( P_E \right)_{\substack{\text{switched} \\ \text{PCM/FM}}} = \frac{1}{\sqrt{2 \pi}} \int_{-\infty}^{-\beta} e^{-x^2/2} \, dx \qquad \text{(I-11)}$$

which is the same as for PCM/AM (with carrier).

## APPENDIX II

## AN UPPER LIMIT ON THE INACCURACY OF PCM/FM RESULTS DUE TO CARRIER FREQUENCY MUCH GREATER THAN BIT RATE

From Appendix I we see that with $\phi_1 = \phi_2 = \phi$ (i. e., conventional PCM/FM), if we do not assume that $\frac{\omega_1}{2\pi} \gg \frac{1}{T_B}$ and $\frac{\omega_2}{2\pi} \gg \frac{1}{T_B}$

then:

$$S_o = S_m - S_u = \frac{AC}{2}\left(1 + \frac{\sin\left[2\omega_1 T_B + 2\phi\right] - \sin 2\phi}{2\omega_1 T_B} - \frac{\sin\left[(\omega_1 + \omega_2)T_B + 2\phi\right] - \sin 2\phi}{(\omega_1 + \omega_2)T_B} - \frac{\sin 2\pi D}{2\pi D}\right)$$

$$= \frac{AC}{2}\left(1 - \frac{\sin 2\pi D}{2\pi D}\right) + \frac{AC}{2}\left(\frac{\sin\left[2\omega_1 T_B + 2\phi\right] - \sin 2\phi}{2\omega_1 T_B} - \frac{\sin\left[(\omega_1 + \omega_2)T_B + 2\phi\right] - \sin 2\phi}{(\omega_1 + \omega_2)T_B}\right)$$

$$= K_1 (G + V)$$

where:

$$K_1 = \frac{AC}{2}$$

$$G = 1 - \frac{\sin 2\pi D}{2\pi D}$$

$$V = \frac{\sin\left[2\omega_1 T_B + 2\phi\right] - \sin 2\phi}{2\omega_1 T_B} - \frac{\sin\left[(\omega_1 + \omega_2) T_B + 2\phi\right] - \sin 2\phi}{(\omega_1 + \omega_2) T_B}$$

Similarly we find that

$$\sigma_N^{\,2} = K_2(G + W)$$

where $K_2 = \dfrac{C^2 k^2}{T_B}$

$$\text{and } W = \frac{\sin\left[2\omega_1 T_B + 2\phi\right] - \sin 2\phi}{4\omega_1 T_B} + \frac{\sin\left[2\omega_2 T_B + 2\phi\right] - \sin 2\phi}{4\omega_2 T_B} - \frac{\sin\left[(\omega_1 + \omega_2) T_B + 2\phi\right] - \sin 2\phi}{(\omega_1 + \omega_2) T_B}$$

therefore

$$\frac{S_o}{\sigma_N} = \frac{K_1(G + V)}{\sqrt{K_2\,(G + W)}} = K_3 \; \frac{G + V}{\sqrt{G + W}}$$

Now in Appendix I, by neglecting the terms V and W we obtained a simplified expression for $S/\sigma_N$:

$$\left(\frac{S}{\sigma_N}\right)_s = \frac{K_1\, G}{\sqrt{K_2\, G}} = K_3 \sqrt{G}$$

We wish to find an upper limit (as a function of $\omega_1 T_B$ and D) on the (positive) value of:

$$F = \frac{\left(\dfrac{S_o}{\sigma_N}\right)_s - \dfrac{S_o}{\sigma_N}}{\left(\dfrac{S_o}{\sigma_N}\right)_s} = 1 - \frac{\dfrac{S_o}{\sigma_N}}{\left(\dfrac{S_o}{\sigma_N}\right)_s} = 1 - R$$

This quantity must be non-negative since $\left(\frac{S_o}{\sigma_N}\right)_s \geq \frac{S_o}{\sigma_N}$

This follows from the following reasoning: Error probability is a monotonic decreasing function of output signal-to-noise ratio. Hence we have shown in Appendix I that if the carrier frequencies, $\frac{\omega_1}{2\pi}$ and $\frac{\omega_2}{2\pi}$, are arbitrarily high, $\left(\frac{S_o}{\sigma_N}\right)_s$ is the highest obtainable value of output signal-to-noise ratio. If the carrier frequencies are not high (relative to the bit rate) the highest obtainable output signal-to-noise ratio is $\frac{S_o}{\sigma_N}$.

But since any signal with low frequency carrier could be obtained by multiplying a signal with arbitrarily high carrier by a sinusoid and filtering to eliminate the sum frequencies (although the inverse is not necessarily possible), $\frac{S_o}{\sigma_N}$ cannot be greater than $\left(\frac{S_o}{\sigma_N}\right)_s$. Also, since both $\frac{S_o}{\sigma_N}$ and $\left(\frac{S_o}{\sigma_N}\right)_s$ must be real and positive, so must their ratio, R.

Therefore if we can find a lower limit on $R^2$ (i.e. $(R^2)_{min}$ then an upper limit on F is established since $F \leq 1 - \sqrt{(R^2)_{min}}$.

$$R^2 = \frac{(G+V)^2}{G(G+W)} = 1 + \frac{V}{G} + \frac{V-W}{G} + \frac{(V-W)^2}{G^2 + GW} \geq 1 + \frac{V}{G} + \frac{V-W}{G} \tag{II-1}$$

Let $\omega_1 T_B = x$ and $\omega_2 T_B = x + 2\pi D$

then:

$$V = \frac{\sin(2x + 2\phi)}{2x} - \frac{\sin(2x + 2\pi D + 2\phi)}{2x + 2\pi D} + \frac{\sin 2\phi}{2x + 2\pi D} - \frac{\sin 2\phi}{2x}$$

$$= \frac{2x\left[\sin(2x + 2\phi) - \sin(2x + 2\pi D + 2\phi)\right] + 2\pi D\left[\sin(2x + 2\phi) - \sin 2\phi\right]}{2x\,(2x + 2\pi D)}$$

$$= \frac{4x \sin \pi D\left[\cos(2x + 2\phi + \pi D)\right] + 4\pi D\left[\cos(x + 2\phi)\sin x\right]}{2x\,(2x + 2\pi D)}$$

$$V_{peak} \leq \frac{x \sin \pi D + \pi D}{x\,(x + \pi D)} \triangleq P$$

Similarly:

$$V\text{-}W = \frac{\sin(2x + 2\phi) - \sin 2\phi}{4x} - \frac{\sin(2x + 4\pi D + 2\phi) - \sin 2\phi}{4\,(x + 2\pi D)}$$

$$= \frac{2x \sin 2\pi D\left[\cos(2x + 2\phi + 2\pi D)\right] + 4\pi D\left[\cos(x + 2\phi)\sin x\right]}{4x\,(x + 2\pi D)}$$

$$(V\text{-}W)_{peak} \leq \frac{x \sin 2\pi D + 2\pi D}{2x\,(x + 2\pi D)} \triangleq Q$$

A liberal lower limit for $R^2$ is obtained by using the negative peak value limits for V and V-W in equation (II-1) i.e.,

$$R^2 \geq 1 - \frac{|P|}{G} - \frac{|Q|}{G}$$

and hence a liberal upper limit for F is

$$F \leq 1 - \sqrt{1 - \frac{|P|}{G} - \frac{|Q|}{G}} = 1 - \sqrt{1 - U}$$

$$= \frac{1}{2}U + \frac{1\cdot 1}{2\cdot 4}U^2 + \frac{1\cdot 1\cdot 3}{2\cdot 4\cdot 6}U^3 + \frac{1\cdot 1\cdot 3\cdot 5}{2\cdot 4\cdot 6\cdot 8}U^4 + \ldots$$

where: $U \triangleq \frac{|P|}{G} + \frac{|Q|}{G}$

This upper limit for F is plotted as a function of deviation ratio, D, for $\frac{\omega_1}{2\pi} \geq \frac{10}{T_B}$ in Figure II-1. (This limit is obviously very liberal, at least for low values of D, since the true limit is zero for D = 0.) Hence, if the lower carrier frequency, $\frac{\omega_1}{2\pi}$, is at least ten times the bit rate, we see from Figure II-1 that the theoretical curve of Figure 2 (i.e. for D = .715) cannot be in error by more than 0.8%, or .007 db, on the $\beta$ axis.

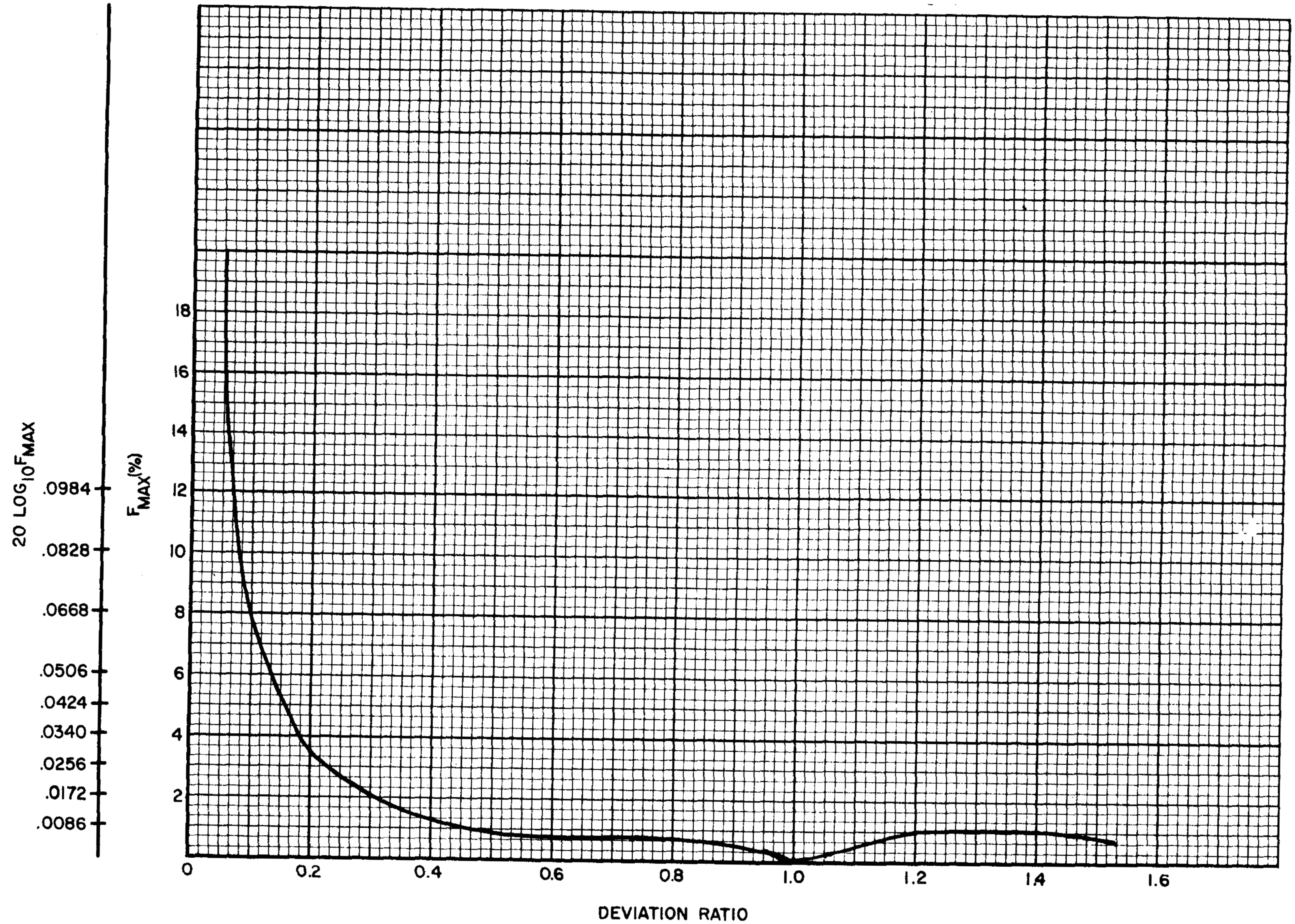



Figure II-1 Upper Limit, $F_{max}$, on the Inaccuracy Incurred by

Assuming $\frac{\omega_1}{2\pi} \gg \frac{1}{T_B}$ When Actually $\frac{\omega_1}{2\pi} \geq \frac{10}{T_B}$

($\omega_2$ greater than $\omega_1$)

# APPENDIX III

## FM RECEIVER OUTPUT NOISE

The development of a theoretical treatment of PCM/FM reception (with a receiver using a conventional discriminator) which would yield results consistent with experimental measurement would require an expression for the amplitude distribution of the filtered video noise for any deviation. Although no such expression has been derived, one of Rice's results (Reference 12 equation 5.4) can be interpreted as the amplitude distribution of unfiltered video noise for any static deviation. In an effort to determine whether or not Rice's result might be used in some way to predict PCM/FM error probability, the theoretical amplitude distribution of unfiltered video noise was calculated (using Rice's result) for a 60 kc static deviation (measured from the center of the discriminator characteristic which is assumed aligned with the center of the I. F. ) using the actual measured I. F. bandpass characteristic of an available Nems-Clarke FM receiver (effective noise bandwidth of 126 kc and measured noise figure of 7 db) and an input signal to noise power ratio of 6.8 (8.3 db).

The video noise amplitude distribution was then measured experimentally, with the same parameters, (with both static deviation and square wave modulation) using the Nems-Clarke FM receiver. The experimental set-up was that shown in Figure III-1. Theoretical and experimental results are presented in Figure III-2. The video noise amplitude in these plots is in units of equivalent deviation so that the video "bit decision level" or "slice level" is at -60 kc on the plots (since the deviation due to the signal is ± 60 kc). Therefore, the value of the cumulative probability of the noise at - 60 kc is just the bit-error probability. From the curves of Figure III-2 it is seen that the bit-error probability measured for the static case (Curve B) is almost identical with that calculated (Curve A). Also, the measured and calculated cumulative probability distributions agree very well. The

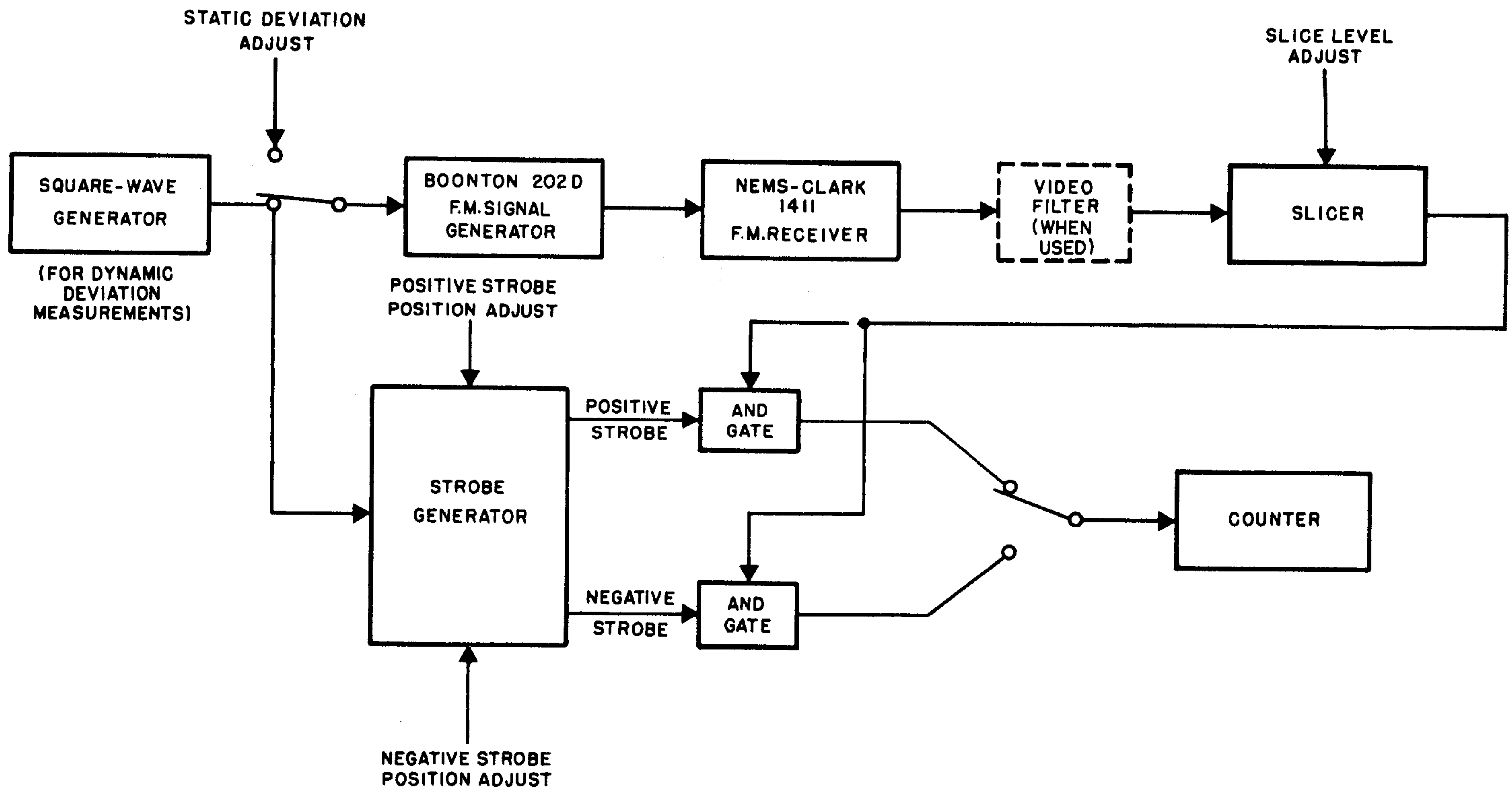
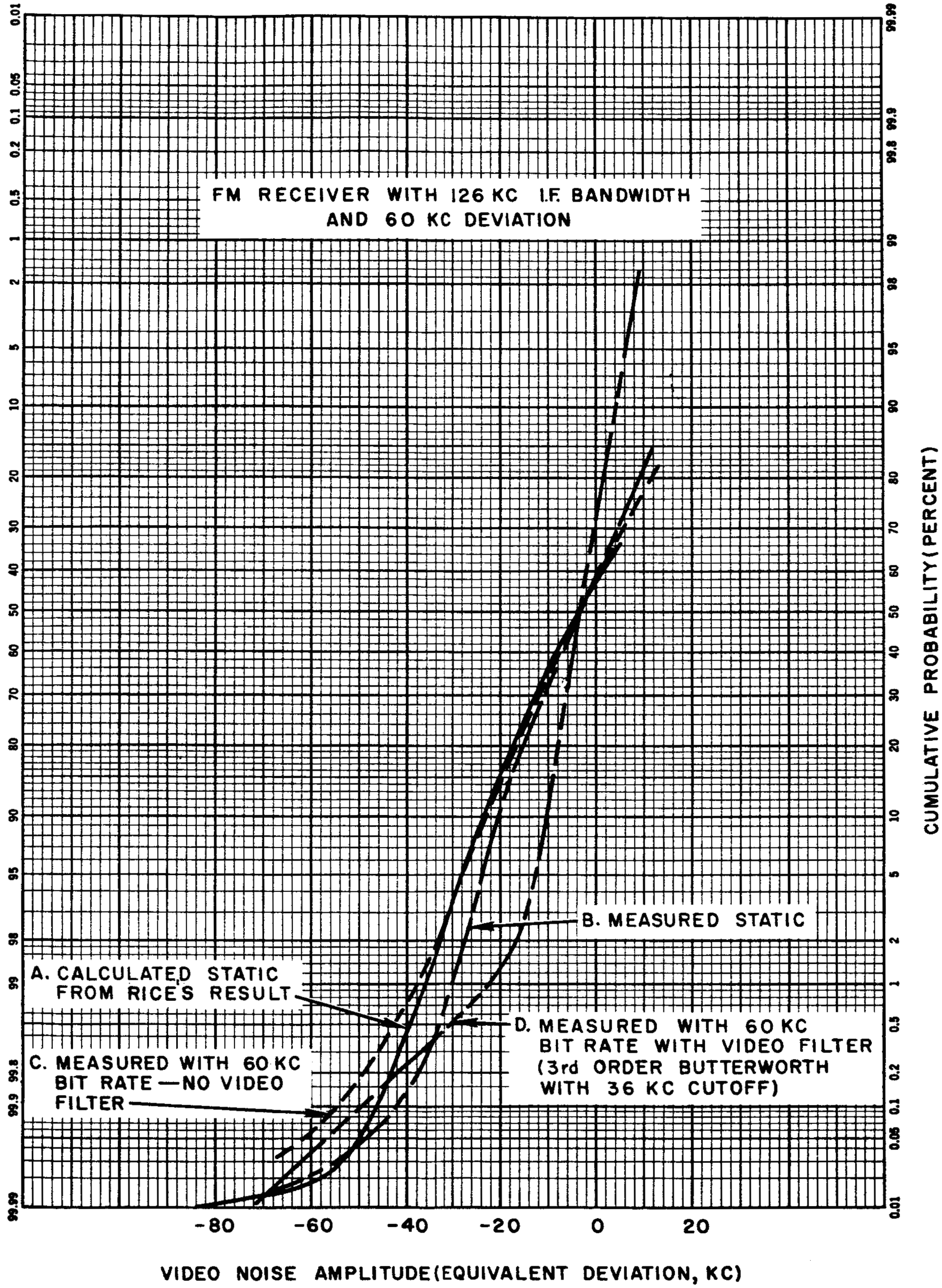



Figure III-1 Experimental Test Set-Up




35154

Figure III-2 Video Noise Amplitude Cumulative Probability Distribution For F.M. Discriminator measured dynamic curve with 60 kc bit rate and no video filter (Curve C) agrees fairly well with the calculated static curve, giving only slightly higher bit error probability. Hence it would seem that if a video filter cutting off at six tenths the bit rate were now used, it should decrease the bit-error probability significantly since the rms video noise can be calculated or observed to be considerably less with the video filter while the signal amplitude is not appreciably affected. Curve D shows, however, that the bit-error probability is hardly affected at all by the video filter, although the shape of the noise amplitude distribution is modified quite a lot. Measurements were made using other combinations of bit-rate and deviation and, in general, the addition of video filtering does not appear to have much effect on error probability unless both deviation and bit-rate are low (1/4 or less) compared to the I. F. bandwidth. This conclusion may also be drawn from the results of the measurements of Reference 15.

These results indicate that there are at least two separate phenomena causing bit-error in a conventional PCM/FM receiver. One seems to be due to the occasional occurrence of I. F. noise amplitudes large enough to cause errors. This phenomenon appears to be a gradual (rather than abrupt) occurrence of the "improvement threshold" phenomenon. Errors resulting from this phenomenon seem unaffected by video filtering. The other phenomenon causing bit-error is the familiar video noise with parabolic power spectrum (Reference 10 page 52-54) which is affected by video filtering, but does not contribute significant error unless the deviation is low compared to I. F. bandwidth.

## APPENDIX IV

## GENERATION OF SETS OF VARIATES FROM AN N-DIMENSIONAL GAUSSIAN DISTRIBUTION WITH SPECIFIED CORRELATION COEFFICIENTS

The generation of random gaussian variates is a fairly routine task, particularly when a digital computer is used. Standard computer subroutines are available for the generation of sets of independent variates from a gaussian distribution with any specified mean and variance. But it is sometimes necessary to generate sets of variates from an n-dimensional gaussian distribution with specified correlation coefficients, $\rho_{ij}$, for each pair of variates. This can be accomplished by first generating a set of n independent gaussian variates and performing a linear transformation of them. The required transformation equations for a specified covariance matrix are derived below.

The joint probability density function for n gaussian variates, $y_1$, $y_2, ---, y_n$, with zero means may be expressed in matrix form as (see Reference 1 Section 8-3):

$$g(y_1, y_2, ---, y_n) = \frac{1}{(2\pi)^{n/2} \left| M \right|^{1/2}} \exp\left( -\frac{1}{2} Y^T M^{-1} Y \right) \tag{IV-1}$$

where M is the (square) covariance matrix

$$M = \begin{bmatrix} \lambda_{11} & \lambda_{12} & --- & \lambda_{1n} \\ \lambda_{21} & \lambda_{22} & --- & \lambda_{2n} \\ --- & --- & --- & --- \\ \lambda_{n1} & \lambda_{n2} & --- & \lambda_{nn} \end{bmatrix}$$

in which $\lambda_{nm} = \overline{y_n y_m} = \sigma_n \sigma_m \rho_{nm}$

$\sigma_i^2$ = variance of $y_i$

and where $|M|$ is the determinant of M

$M^{-1}$ is the inverse of M

Y is a column matrix with elements $y_1, y_2, ---, y_n$.

$Y^T$ is the transpose of Y

For uncorrelated gaussian variates, $v_1, v_2, ---, v_n$, the covariance matrix is a diagonal matrix, D:

$$D = \begin{bmatrix} d_1 & 0 & --- & 0 \\ 0 & d_2 & --- & 0 \\ --- & --- & --- & --- \\ 0 & 0 & --- & d_n \end{bmatrix}$$

where $d_1, d_2, ---, d_n$ are the variances of $v_1, v_2, ---, v_n$; and the joint probability density function is:

$$g(v_1, v_2, ---, v_n) = \frac{1}{(2\pi)^{n/2} \, |D|^{1/2}} \exp\left(-\frac{1}{2} V^T D^{-1} V\right) \quad \text{(IV-2)}$$

Now a square matrix, M, can be reduced to a diagonal matrix such as D by the matrix operation (see Reference 4, Chapter III, Art. 10):

$$D = P^{-1} M P \quad \text{(IV-3)}$$

where P is the "modal"-matrix whose column elements are the components of the unit eigen vectors, $e_1, e_2, ---, e_n$, of the matrix M. These eigen vectors are the normalized solutions of the matrix equation:

$$M \, e_i] = \Lambda_i \; e_i]$$

where $e_i]$ is a column matrix with elements $e_{i1}, e_{i2}, --- e_{in}$.

and the $\Lambda_i$'s (eigen values of M) are solutions of:

$$\left| M - \Lambda_i I \right| = 0$$

I is the identity matrix, $\begin{bmatrix} 1 & 0 & --- & 0 \\ 0 & 1 & --- & 0 \\ -- & -- & --- & -- \\ 0 & 0 & --- & 1 \end{bmatrix}$

Since M is a symmetric matrix, the existence of a nonsingular modal matrix, P, is guaranteed. Furthermore, P is an orthogonal matrix and therefore its transpose is equal to its inverse: $P^T = P^{-1}$.

We seek a transformation which will transform any set of v's (uncorrelated gaussian variates) into an equivalent set of y's (gaussian variates with specified covariance matrix, M). That is, if $g(v_1, v_2, ---, v_n)$ of (IV-2) is evaluated for any set of v's, we wish to determine the set of y's which will give the same value for $g(y_1, y_2, ---, y_n)$ of (IV-1). The factors in front of the exponential are the same since

$$\left| D \right| = \left| P^{-1} \right| \cdot \left| M \right| \cdot \left| P \right| = \left| P^{-1} \right| \cdot \left| P \right| \cdot \left| M \right| = \left| P^{-1} P \right| \cdot \left| M \right| = \left| M \right|$$

Hence we seek the transformation between V and Y which will make

$$V^T D^{-1} V = Y^T M^{-1} Y \qquad \text{(IV-4)}$$

But since $D = P^{-1} MP$ we have

$$D^{-1} = \left[ MP \right]^{-1} P = P^{-1} M^{-1} P$$

and therefore:

$$V^T D^{-1} V = V^T P^{-1} M^{-1} PV = \left[ V^T P^T \right] M^{-1} \left[ PV \right] = \left[ PV \right]^T M^{-1} \left[ PV \right] \qquad \text{(IV-5)}$$

For the right hand sides of (IV-4) and (IV-5) to be equal we must have

$$Y = PV \qquad \text{(IV-6)}$$

Hence this is the required transformation between V and Y. But we must note that (see Reference 4, page 115):

$$D = P^{-1}MP = \begin{bmatrix} \Lambda_1 & 0 & --- & 0 \\ 0 & \Lambda_2 & --- & 0 \\ -- & -- & --- & -- \\ 0 & -- & --- & \Lambda_n \end{bmatrix}$$

so that the variance of $v_1$ is $\Lambda_1$

the variance of $v_2$ is $\Lambda_2$

etc.

Therefore the v's all have, in general, different variances. But when we are generating sets of variates it is more convenient to select all of our uncorrelated variates from the same distribution and hence the same variance. We may select such variates, $u_1, u_2, ---, u_n$, from a gaussian distribution of unity variance and transform them, to get the required v's, with the transformation:

$$V = L\,U$$

where

$$L = \begin{bmatrix} \sqrt{\Lambda_1} & 0 & --- & 0 \\ 0 & \sqrt{\Lambda_2} & --- & 0 \\ --- & --- & --- & --- \\ 0 & 0 & --- & \sqrt{\Lambda_n} \end{bmatrix}$$

Then the required transformation from u's to y's is:

$$Y = P\,L\,U \qquad \text{(IV-7)}$$

For the case n = 2 with $\lambda_{11} = \lambda_{22} = \sigma^2 = 1$, and $\lambda_{12} = \lambda_{21} = \sigma^2\rho = \rho$

we have:

$$M = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

and

$$\left| M - \Lambda_i I \right| = \begin{vmatrix} 1 - \Lambda_i & \rho \\ \rho & 1 - \Lambda_i \end{vmatrix} = 0$$

or $\Lambda_1 = 1 + \rho$

$\Lambda_2 = 1 - \rho$

then $M \, e_i] = \Lambda_i \, e_i]$ becomes

$$e_{i1} + \rho e_{i2} = \Lambda_i \, e_{i1}$$

$$\rho e_{i1} + e_{i2} = \Lambda_i \, e_{i2}$$

yielding:

$$e_{11} = \frac{1}{\sqrt{2}} \qquad e_{12} = \frac{1}{\sqrt{2}}$$

$$e_{21} = \frac{1}{\sqrt{2}} \qquad e_{22} = -\frac{1}{\sqrt{2}}$$

Hence

$$PL = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \cdot \begin{bmatrix} \sqrt{1+\rho} & 0 \\ 0 & \sqrt{1-\rho} \end{bmatrix} = \begin{bmatrix} \sqrt{\frac{1+\rho}{2}} & \sqrt{\frac{1-\rho}{2}} \\ \sqrt{\frac{1+\rho}{2}} & -\sqrt{\frac{1-\rho}{2}} \end{bmatrix}$$

and the trnasformation equations are:

$$y_1 = \sqrt{\frac{1+\rho}{2}} \, u_1 + \sqrt{\frac{1-\rho}{2}} \, u_2$$

$$y_2 = \sqrt{\frac{1+\rho}{2}} \, u_1 - \sqrt{\frac{1-\rho}{2}} \, u_2 \qquad \text{(IV-8)}$$

## APPENDIX V

## THE COMPUTER PROGRAMS

The computer program for simulating the minimum-error demodulation and for computing (by model-sampling) the resulting error probabilities was written in Fortran language. It was, in fact, written with a fairly limited version of Fortran since it was initially written for, and the program logic checked out on, an IBM 1620 which has a more limited Fortran vocabulary than does the IBM 7090 used to obtain the final results.

The simulation of the minimum-error demodulation (for two 6-bit words, i. e., $n = 2$, $m = 6$) is accomplished by evaluating (for any given $z_1$ and $z_2$) the expression $p(y_{2(p)} \mid z_1, z_2)/K_t$ of equation(4. 3) for each of the $2^6$ possible $y_{2(p)}$'s, and selecting the largest of the $2^6$ values so obtained. This largest value is then normalized (by dividing it by the sum of all the $2^6$ values) so that the result represents the probability, $p_M$, of the most probable $y_{2(p)}$. Then $Q = 1 - p_M$ is stored for use in obtaining the estimate $P_2$ of equation (4. 5), or the estimate $P_3$ of equation (4. 9). This must be done for many sets of z's which must themselves be generated by the computer. Actually, the $\zeta$'s of (4. 3) are generated rather than the z's since the $\zeta$'s represent the only attributes of the z's used in the simulated demodulation. To insure that the $\zeta$'s are selected from the proper distribution we select data samples, $Y_1$ and $Y_2$, from the specified data distribution, code the sample values in 6-bit binary PCM code of amplitude $\pm$ S/N (+ for "yes" bits and - for "no" bits) (S/N = signal-to-noise ratio = $S/K \sqrt{B/2}$) and add to the amplitude of each bit of the codes, constant independent "noise" values selected from a gaussian distribution of zero mean and unity variance (see first paragraph of Chapter 4). The resulting values are the $\zeta$'s.

For signal-to-noise ratios less than 2 the estimate $P_2$ of (4. 5) is used, but for signal-to-noise ratio of 2 or more the variance of this estimate becomes too large and hence $P_3$ of (4. 9) is used. In this case the noise distribution must be modified and the Q's "weighted" as discussed in Chapter 4. The Fortran (source) programs are presented in Tables 1 and 2 for both of these cases for $n = 2$ and $m = 6$ (i. e., for two 6-bit words). The modifications required for $n = 1$ and/or $m = 3$ are apparent. They consist primarily of changing the appropriate indices and, for the case with modified noise distribution, using a different weighting factor for weighting the Q's (see Chapter 4 for method of calculating the weighting factors). However, for convenience, the programs for $n = 1$, $m = 6$ are presented in Tables 3 and 4.

To facilitate explanation of programs 1 and 2 they have been broken into eleven divisions which have been labeled A through K (see Tables 1 and 2). Given below for each of these divisions is a brief statement of the purpose of that portion of the program followed by more detailed explanation where necessary.

A. Store required constants. The insertion of a "CALL FTRAP(O)" statement following the dimension statements may be necessary to avoid automatic computer stoppage due to underflow arising from the randomness of numbers appearing in a later portion of the program.

B. Generate and store 6-bit binary PCM codes corresponding to each possible transmitted word. The bit values are "+ 1" and "-1" rather than "1" and "0". Indexing of bit values is such that JC(I, K) is the $K^{th}$ bit (K = 1 for least significant bit) of the binary representation of I-1. These, when multiplied by S/N, become the $\eta$'s of equation (4. 3).

C. Read input data and print identification of output data. Here SN represents signal-to-noise ratio S/N, and RH represents the correlation coefficient, $\rho$, between data samples.

D. Calculate required constants which depend upon input data, and set initial values (of CT, CTM, and PSI). RRSCR and RRSC2 are used later for calculating the joint probabilities $f(Y_1, Y_2)$ (see E). C1 and C2 are the coefficients used to transform the independent data samples into correlated data samples. For program 2, "D" and "G" are needed for calculating the weighting factors required when the modified noise distribution is used.

E. Calculate and store the $(64)^2$ values of joint probability $p(Y_1, Y_2)$. These are indexed such that WT(I, M) is $p(Y_1, Y_2)$ where $Y_1$ is a quantized data sample of amplitude I-1 and $Y_2$ is a quantized data sample of amplitude M-1 (see Figure 4).

F. Generate two quantized correlated data samples, $Y_1$ and $Y_2$, represented by JYT(1) and JYT(2). Statements 80 and 95 generate two independent samples, W(1) and W(2), from the proper gaussian distribution and the other two statements perform the transformation represented by equations (4. 2).

G. Generate binary code words representing JYT(1) and JYT(2), and add noise from the proper distribution to each bit of these codes. For program 1 the "proper" noise distribution is gaussian while for program 2 it is the modified distribution discussed in Chapter 4. Values of JYT(1) less than zero are set equal to zero, and values greater than 63 are set equal to 63. This is accomplished by statements up through statement 130. The inner "DO 140" loop of program 1 and the inner "DO 144" loop of program 2 generate the "bit noises," US, ($\nu$'s of equation (3. 9)) and add them to the bits of JYT(1) and JYT(2) ($\eta$'s corresponding to transmitted waveforms) to produce the noisy received bits, CU(I, K)

( $\zeta$ 's of equation (4.3)). For program 2 the two statements immediately preceding statement 131 represent the "auxiliary game of chance" referred to in Chapter 4 for determining whether the noise is selected from a gaussian or flat distribution (i.e., Figure 5a or 5c). Statement 131 generates noise values from a flat distribution of zero mean and range 2B while statement 132 generates noise values from a gaussian distribution of zero mean and unity variance. Statements 134 through 137 calculate the reciprocal, (represented by RFN) of the appropriate weighting factor, $\frac{h(N)}{h^*(N)}$, which multiplies Q (Y, N) of equation 4.9 to compensate for the modification of the noise distribution.

H. Pre-calculate and store the factors required for evaluating $p(y_{2(p)} \mid z_1, z_2)$ for each $y_{2(p)}$. These factors are: $\sum_{r=1}^{6} \eta_{2(p)r} \zeta_{2r}$ (of equation (4.3)) evaluated for each p (these values are represented by PR2(I) where I = p + 1), and $\exp\left(\sum_{r=1}^{6} \eta_{1(q)r} \zeta_{1r}\right)$ evaluated for each q (these values are represented by F1(M) where M = q + 1). Also, statement 165 and the statement preceding it select the largest value of F1(M) for use in the re-scaling immediately following.

I. Re-scale factors to utilize the full range of computer capability to reprint numbers. This avoids overflow and minimizes underflow due to wide range of values of factors encountered due to random sampling.

J. Evaluate $p(y_{2(p)} \mid z_1, z_2)/K_t$ of equation (4.3) (represented by "P" in the program) for each $y_{2(p)}$ (i.e., each "I" in the "DO 180" loop) and choose the largest value obtained (represented by "PP" in the program). Also, the sum (i.e., "PS") of all of the values except the largest is computed.

K. Normalize the largest value, PP, by dividing it by the sum of all values, PS + PP, so that the result represents the probability, $p_M$, of the most probable $y_{2(p)}$. Then calculate the error probability estimate $P_2$, or $P_3$ (see Chapter 4). In program 1, $p_M$ is represented by PEM and $P_2$ is represented by AVRM. In program 2, Q is represented by PEM and $P_3$ is represented by AVRI. In program 2, the statement "PEI = FN * PEM" represents the weighting of the Q's indicated in equation (4. 9). AVRB represents an auxiliary estimate of error probability. It was calculated merely for experimental purposes and can be ignored here.

The remaining portion of the program is for printing the results and establishing formats and is self explanatory.

Programs 3 and 4 are similar, but somewhat simpler than programs 1 and 2. They require no additional explanation. Compilation time required is slightly less than a minute for programs 3 and 4, and a little more than a minute for programs 1 and 2.

**Table 1 Computer Program No. 1, n = 2, m = 6,**

**Gaussian Noise Distribution**

```
      DIMENSION JC(64,6),XM(64),XMS(64),CU(2,6),W(2),JYT(2)
      DIMENSION WT(64,64),PR2(64),F1(64),PRP(64)
      XMY=31.5                                              A
      XMY05=32.0
      SY=12.115769
      RSY=1.0/SY
      DO 30 K=1,6
   30 JC(1,K)=-1
      DO 62 I=1,63
      DO 40 K=1,6
   40 JC(I+1,K)=JC(I,K)
      DO 60 K=1,6                                           B
      JC(I+1,K)=JC(I+1,K)+1
      IF(JC(I+1,K)) 50,50,60
   50 JC(I+1,K)=1
      GO TO 62
   60 JC(I+1,K)=-1
   62 CONTINUE
   70 READ INPUT TAPE 7,1,SN
      READ INPUT TAPE 7,1,RH
      WRITE OUTPUT TAPE 6,5
      WRITE OUTPUT TAPE 6,2,SN,RH                           C
      WRITE OUTPUT TAPE 6,3
      WRITE OUTPUT TAPE 6,4
      WRITE OUTPUT TAPE 6,3
      RRSC=1.0/(1.0-RH**2)
      RRSCR=SQRT (RH*RRSC)
      RRSC2=0.5*RRSC
      C1=SQRT (0.5*(1.0+RH))                                D
      C2=SQRT (0.5*(1.0-RH))
      CT=1.0
      PSM=0.0
      DO 72 I=1,64
      Y=I-1
      X=(Y-XMY)*RSY
      XM(I)=X*RRSCR
   72 XMS(I)=X**2*RRSC2                                     E
      DO 75 I=1,64
      DO 75 M=1,64
   75 WT(I,M)=EXP (XM(I)*XM(M)-XMS(I)-XMS(M))
   80 DO 95 I=1,2
   95 CALL NDRN1B(SY,0.0,W(I))                              F
      JYT(1)=C1*W(1)+C2*W(2)+XMY05
      JYT(2)=C1*W(1)-C2*W(2)+XMY05
      DO 140 I=1,2                                          G
```

Table 1 (Continued)

```
      IF(JYT(I)) 99,130,100
   99 JYT(I)=0
      GO TO 130
  100 IF(JYT(I)-63) 130,130,110
  110 JYT(I)=63
  130 J=JYT(I)+1                                             G
      DO 140 K=1,6
      CALL NDRN1B (1.0,0.0,US)
      IF(JC(J,K)) 136,136,138
  136 CU(I,K)=US-SN
      GO TO 140
  138 CU(I,K)=US+SN
  140 CONTINUE  - - - - - - - - - - - - - - - - - - - - - - -
      PPR1=0.0
      DO 170 I=1,64
      PWR2=0.0
      PWR1=0.0
      DO 160 K=1,6
      IF(JC(I,K))145,145,150
  145 PWR2=PWR2-CU(2,K)
      PWR1=PWR1-CU(1,K)
      GO TO 160                                              H
  150 PWR2=PWR2+CU(2,K)
      PWR1=PWR1+CU(1,K)
  160 CONTINUE
      PR2(I)=PWR2*SN
      PWR1=PWR1*SN
      IF(PWR1-PPR1) 170,170,165
  165 PPR1=PWR1
  170 F1(I)=PWR1
      SKP1=80.0-PPR1  - - - - - - - - - - - - - - - - - - - -
      DO 171 I=1,64
  171 F1(I)=EXP(F1(I)+SKP1)
      PPRP=0.0
      DO 175 I=1,64
      S=0.0
      DO 172 M=1,64
  172 S=S+F1(M)*WT(I,M)
      IF(S) 173,173,273                                      I
  173 PRPI=-(10.0**30)
      GO TO 175
  273 PRPI=PR2(I)+ELOG(S)
      IF(PRPI-PPRP) 175,175,174
  174 PPRP=PRPI
  175 PRP(I)=PRPI
```

Table 1 (Continued)

```
                                                             I
      SKPR=80.0-PPRP
      PP=0.0
      PS=0.0
      DO 180 I=1,64
      P=EXP(PRP(I)+SKPR)
      IF(P-PP) 176,176,177
  176 PS=PS+P                                                J
      GO TO 180
  177 PS=PS+PP
      PP=P
  180 CONTINUE
      PEM=PP/(PS+PP)
      PSM=PSM+PEM                                            K
      AVRC=PSM/CT
      AVRM=1.0-AVRC
      YT1=JYT(1)
      YT2=JYT(2)
      WRITE OUTPUT TAPE 6,1,CT,YT1,YT2,PEM,AVRM
      CT=CT+1.0
      IF(CT-100.0) 80,80,70
    1 FORMAT (E15.8,E15.8,E15.8,E15.8,E15.8)
    2 FORMAT (4H SN=E15.8,5X,4HRHO=E15.8)
    3 FORMAT (1H0)
    4 FORMAT (5X,5HCOUNT,11X,3HYT1,12X,3HYT2,12X,3HPEM,11X,4HAVRM)
    5 FORMAT (1H1)
      END
```

Table 2 Computer Program No. 2, n = 2, m = 6,

Modified Noise Distribution

```
      DIMENSION JC(64,6),XM(64),XMS(64),CU(2,6),W(2),JYT(2)
      DIMENSION WT(64,64),PR2(64),F1(64),PRP(64)
      XMY=31.5
      XMY05=32.0                                              A
      SY=12.115769
      RSY=1.0/SY
      DO 30 K=1,6
   30 JC(1,K)=-1
      DO 62 I=1,63
      DO 40 K=1,6
   40 JC(I+1,K)=JC(I,K)
      DO 60 K=1,6
      JC(I+1,K)=JC(I+1,K)+1                                   B
      IF(JC(I+1,K)) 50,50,60
   50 JC(I+1,K)=1
      GO TO 62
   60 JC(I+1,K)=-1
   62 CONTINUE
   70 READ INPUT TAPE 7,1,SN,B
      READ INPUT TAPE 7,1,RH
      READ INPUT TAPE 7,1,A
      WRITE OUTPUT TAPE 6,5                                   C
      WRITE OUTPUT TAPE 6,2,SN,B,RH,A
      WRITE OUTPUT TAPE 6,3
      WRITE OUTPUT TAPE 6,4
      WRITE OUTPUT TAPE 6,3
      RRSC=1.0/(1.0-RH**2)
      RRSCR=SQRT (RH*RRSC)
      RRSC2=0.5*RRSC
      C1=SQRT (0.5*(1.0+RH))
      C2=SQRT (0.5*(1.0-RH))
      CT=1.0                                                  D
      CTM=0.0
      PSI=0.0
      PI2RB=0.79788455*B
      D=(A/PI2RB)**12
      G=(1.0-A)*PI2RB/A
      DO 72 I=1,64
      Y=I-1
      X=(Y-XMY)*RSY
      XM(I)=X*RRSCR                                           E
   72 XMS(I)=X**2*RRSC2
      DO 75 I=1,64
      DO 75 M=1,64
   75 WT(I,M)=EXP (XM(I)*XM(M)-XMS(I)-XMS(M))
   80 DO 95 I=1,2
   95 CALL NDRN1B(SY,0.0,W(I))                                F
```

Table 2 (Continued)

```
      JYT(1)=C1*W(1)+C2*W(2)+XMY05
      JYT(2)=C1*W(1)-C2*W(2)+XMY05
      RFN=D
      DO 144 I=1,2
      IF(JYT(I)) 99,130,100
   99 JYT(I)=0
      GO TO 130
  100 IF(JYT(I)-63) 130,130,110
  110 JYT(I)=63
  130 J=JYT(I)+1
      DO 144 K=1,6
      FP=RAM2B(0)
      IF(FP-A) 131,132,132
  131 U=RAM2B(0)
      UM=2.0*B*U-B
      GO TO 136
  132 CALL NDRN1B(1.0,0.0,UM)
      IF(UM)134,134,143
  134 Q=-UM
      GO TO 135
  143 Q=UM
  135 IF(Q-B) 136,136,137
  136 RFN=RFN*(G+EXP(0.5*UM**2))
      GO TO 138
  137 RFN=RFN*G
  138 IF(JC(J,K)) 139,139,140
  139 CU(I,K)=UM-SN
      GO TO 144
  140 CU(I,K)=UM+SN
  144 CONTINUE
      PPR1=0.0
      DO 170 I=1,64
      PWR2=0.0
      PWR1=0.0
      DO 160 K=1,6
      IF(JC(I,K))145,145,150
  145 PWR2=PWR2-CU(2,K)
      PWR1=PWR1-CU(1,K)
      GO TO 160
  150 PWR2=PWR2+CU(2,K)
      PWR1=PWR1+CU(1,K)
  160 CONTINUE
      PR2(I)=PWR2*SN
      PWR1=PWR1*SN
      IF(PWR1-PPR1) 170,170,165
  165 PPR1=PWR1
  170 F1(I)=PWR1
```

F

G

H

Table 2 (Continued)

```
      SKP1=80.0-PPR1
      DO 171 I=1,64
  171 F1(I)=EXP(F1(I)+SKP1)
      PPRP=0.0
      DO 175 I=1,64
      S=0.0
      DO 172 M=1,64
  172 S=S+F1(M)*WT(I,M)                                 I
      IF(S) 173,173,273
  173 PRPI=-(10.0**30)
      GO TO 175
  273 PRPI=PR2(I)+ELOG(S)
      IF(PRPI-PPRP) 175,175,174
  174 PPRP=PRPI
  175 PRP(I)=PRPI
      SKPR=80.0-PPRP
      PP=0.0
      PS=0.0
      DO 180 I=1,64
      P=EXP(PRP(I)+SKPR)
      IF(P-PP) 176,176,177                              J
  176 PS=PS+P
      GO TO 180
  177 PS=PS+PP
      PP=P
  180 CONTINUE
      PEM=1.0-PP/(PS+PP)
      YT1=JYT(1)
      YT2=JYT(2)
      FN=1.0/RFN
      PEI=FN*PEM                                        K
      PSI=PSI+PEI
      AVRI=PSI/CT
      CTM=CTM+FN
      AVRB=PSI/CTM
      WRITE OUTPUT TAPE 6,6,CT,YT1,YT2,FN,PEI,PEM,AVRI,AVRB
      CT=CT+1.0
      IF(CT-100.0) 80,80,70
    1 FORMAT (E15.8,E15.8,E15.8,E15.8,E15.8)
    2 FORMAT (4H SN=E15.8,5X,2HB=E15.8,5X,4HRHO=E15.8,5X,2HA=E15.8)
    3 FORMAT (1H0)
    4 FORMAT (6X,2HCT,10X,3HYT1,10X,3HYT2,11X,2HFN,10X,3HPEI,10X,3HPEM,
     110X,4HAVRI,9X,4HAVRB)
    5 FORMAT (1H1)
    6 FORMAT (1X,E12.6,1X,E12.6,1X,E12.6,1X,E12.6,1X,E12.6,1X,E12.6,
     11X,E12.6,1X,E12.6)
      END
```

Table 3 Computer Program No. 3, n = 1, m = 6,

Gaussian Noise Distribution

```
      DIMENSION JC(64,6),XS2(64),CU(6),PR(64)
      XMY=31.5
      XMY15=XMY+1.5
      SY=XMY/2.6
      RSY=1.0/SY
      DO 10 I=1,64
      Y=I-1
   10 XS2(I)=0.5*(((Y-XMY)*RSY)**2)
      DO 30 K=1,6
   30 JC(1,K)=-1
      DO 70 I=1,63
      DO 40 K=1,6
   40 JC(I+1,K)=JC(I,K)
      DO 60 K=1,6
      JC(I+1,K)=JC(I+1,K)+1
      IF(JC(I+1,K)) 50,50,60
   50 JC(I+1,K)=1
      GO TO 70
   60 JC(I+1,K)=-1
   70 CONTINUE
   75 READ INPUT TAPE 7,1,SN
      WRITE OUTPUT TAPE 6,5
      WRITE OUTPUT TAPE 6,2,SN
      WRITE OUTPUT TAPE 6,3
      WRITE OUTPUT TAPE 6,4
      WRITE OUTPUT TAPE 6,3
      CT=1.0
      PSM=0.0
   80 CALL NDRN1B (1.0,0.0,US)
      J=XMY15+SY*US
      IF(J-1) 100,130,110
  100 J=1
      GO TO 130
  110 IF(J-64) 130,130,120
  120 J=64
  130 DO 150 K=1,6
      CALL NDRN1B (1.0,0.0,US)
      IF(JC(J,K)) 144,144,146
  144 CU(K)=US-SN
      GO TO 150
  146 CU(K)=US+SN
  150 CONTINUE
      PPWR=0.0
      DO 162 I=1,64
      PWR=0.0
```

Table 3 (Continued)

```
      DO 160 K=1,6
      IF(JC(I,K)) 154,154,156
  154 PWR=PWR-CU(K)
      GO TO 160
  156 PWR=PWR+CU(K)
  160 CONTINUE
      PWR=PWR*SN-XS2(I)
      IF(PWR-PPWR) 162,162,161
  161 PPWR=PWR
  162 PR(I)=PWR
      SK=80.0-PPWR
      PP=0.0
      PS=0.0
      DO 180 I=1,64
      P=EXP(PR(I)+SK)
      IF(P-PP) 170,170,175
  170 PS=PS+P
      GO TO 180
  175 PS=PS+PP
      PP=P
  180 CONTINUE
      PEM=1.0-PP/(PS+PP)
      PSM=PSM+PEM
      AVRM=PSM/CT
      YJ=J-1
      WRITE OUTPUT TAPE 6,6,CT,YJ,PP,PS,PEM,AVRM
      CT=CT+1.0
      IF(CT-100.0) 80,80,75
    1 FORMAT (E15.8,E15.8,E15.8,E15.8)
    2 FORMAT (4H SN=E15.8)
    3 FORMAT (1H0)
    4 FORMAT (6X,2HCT,11X,2HYJ,11X,2HPP,11X,2HPS,10X,3HPEM,9X,4HAVRM)
    5 FORMAT (1H1)
    6 FORMAT (1X,E12.6,1X,E12.6,1XE12.6,1X,E12.6,1X,E12.6,1X,E12.6)
      END
```

Table 4 Computer Program No. 4, n = 1, m = 6,

Modified Noise Distribution

```
      DIMENSION JC(64,6),XS2(64),CU(6),PR(64)
      XMY=31.5
      XMY15=XMY+1.5
      SY=XMY/2.6
      RSY=1.0/SY
      DO 10 I=1,64
      Y=I-1
   10 XS2(I)=0.5*(((Y-XMY)*RSY)**2)
      DO 30 K=1,6
   30 JC(1,K)=-1
      DO 70 I=1,63
      DO 40 K=1,6
   40 JC(I+1,K)=JC(I,K)
      DO 60 K=1,6
      JC(I+1,K)=JC(I+1,K)+1
      IF(JC(I+1,K)) 50,50,60
   50 JC(I+1,K)=1
      GO TO 70
   60 JC(I+1,K)=-1
   70 CONTINUE
   75 READ INPUT TAPE 7,1,SN,B
      READ INPUT TAPE 7,1,A
      WRITE OUTPUT TAPE 6,5
      WRITE OUTPUT TAPE 6,2,SN,B,A
      WRITE OUTPUT TAPE 6,3
      WRITE OUTPUT TAPE 6,4
      WRITE OUTPUT TAPE 6,3
      CT=1.0
      CTM=0.0
      PSI=0.0
      PI2RB=0.7978455*B
      D=(A/PI2RB)**6
      G=(1.0-A)*PI2RB/A
   80 CALL NORM1B (1.0,0.0,US)
      J=XMY15+SY*US
      IF(J-1) 100,130,110
  100 J=1
      GO TO 130
  110 IF(J-64) 130,130,120
  120 J=64
  130 RFN=D
      DO 144 K=1,6
      FP=RAM2B(0)
      IF(FP-A) 131,132,132
  131 U=RAM2B(0)
```

Table 4 (Continued)

```
      UM=2.0*B*U-B
      GO TO 136
  132 CALL NDRN1B(1.0,0.0,UM)
      IF(UM)134,134,143
  134 Q=-UM
      GO TO 135
  143 Q=UM
  135 IF(Q-B) 136,136,137
  136 RFN=RFN*(G+EXP(0.5*UM**2))
      GO TO 138
  137 RFN=RFN*G
  138 IF(JC(J,K)) 139,139,140
  139 CU(K)=UM-SN
      GO TO 144
  140 CU(K)=UM+SN
  144 CONTINUE
      PPWR=0.0
      DO 162 I=1,64
      PWR=0.0
      DO 160 K=1,6
      IF(JC(I,K)) 154,154,156
  154 PWR=PWR-CU(K)
      GO TO 160
  156 PWR=PWR+CU(K)
  160 CONTINUE
      PWR=PWR*SN-XS2(I)
      IF(PWR-PPWR) 162,162,161
  161 PPWR=PWR
  162 PR(I)=PWR
      SK=80.0-PPWR
      PP=0.0
      PS=0.0
      DO 180 I=1,64
      P=EXP(PR(I)+SK)
      IF(P-PP) 170,170,175
  170 PS=PS+P
      GO TO 180
  175 PS=PS+PP
      PP=P
  180 CONTINUE
      PEM=1.0-PP/(PS+PP)
      FN=1.0/RFN
      PEI=FN*PEM
      PSI=PSI+PEI
      AVRI=PSI/CT
```

Table 4 (Continued)

```
      CTM=CTM+FN
      AVRB=PSI/CTM
      YJ=J-1
      WRITE OUTPUT TAPE 6,6,CT,YJ,FN,PEI,PEM,AVRI,AVRB
      CT=CT+1.0
      IF(CT-100.0) 80,80,75
    1 FORMAT (E15.8,E15.8,E15.8,E15.8)
    2 FORMAT (4H SN=E15.8,5X,2HB=E15.8,5X,2HA=E15.8)
    3 FORMAT (1H0)
    4 FORMAT (6X,2HCT,11X,2HYJ,11X,2HFN,10X,3HPEI,10X,3HPEM,9X,4HAVRI,
     19X,4HAVRB)
    5 FORMAT (1H1)
    6 FORMAT (1X,E12.6,1X,E12.6,1X,E12.6,1X,E12.6,1X,E12.6,1X,E12.6,
     11X,E12.6,1X,E12.6)
      END
```

# APPENDIX VI

## COMPUTATION RESULTS

The minimum attainable word error probabilities with inter-bit dependence were obtained by monte-carlo computation using the IBM 7090 computer of the University of Michigan Computing Center. The results for 6-bit codes are presented in Table 5 and the results for 3-bit codes are presented in Table 6. The following symbols are used in these tables:

| | |
|---|---|
| S/N | = Signal-to-noise ratio = $S/K\sqrt{B/2}$ |
| $\rho$ | = Correlation coefficient between samples |
| L | = Number of simulated demodulations performed |
| T | = Execution time required on computer |
| P | = Monte-carlo estimate ($P_2$ or $P_3$ of Chapter 4) of word-error probability, $P_W$. |

As indicated in these tables, several hundred simulated demodulations were used (in most cases) for each estimate of word-error probability. (More demodulations were required for 3-bit codes than for 6-bit codes due to the courseness of quantization in the 3-bit case. But, fortunately, more can be tolerated since the computation time per demodulation is much less for the 3-bit case.)

It is of interest to observe the behavior of the estimates as the number of demodulations used in the estimate increases. Typical detailed results obtained for individual demodulations and for averaging of the results of from one to one hundred demodulations are presented in Tables 7, 8, and 9 for three different combinations of S/N and $\rho$. These results are presented in the computer format where $.d_1d_2d_3$----E $b_1b_2$ represents $(.d_1d_2d_3----)10^{b_1b_2}$. That is, for example, .367010E-02 represents $(.367010)10^{-2}$ = .00367010. If the number in the first column (count or CT) is N, the last column gives the estimate, $P_2$ or $P_3$, obtained with the preceding N demodulations.

Monte-carlo computations are not necessary with no inter-bit dependence. For this case, bit-error probabilities, $P_E$, may be determined directly by the use of gaussian tables and equation (1.4). Corresponding word-error probabilities, $P_W$, for m-bit words can then be determined from:

$$P_W = 1 - (1 - P_E)^m$$

Results for m = 6 and for m = 3 are presented in Table 10.

Graphical summaries and discussions of all results are presented in Chapter 5.

Table 5

Computed Word-Error Probabilities for 6-Bit Words, n = 2

| S/N | $\rho$ | L | T(sec.) | P |
|---|---|---|---|---|
| 0.707 | 0 | 100 | 11 | 0.778 |
| 0.707 | 0.5 | 200 | 82 | 0.750 |
| 0.707 | 0.7 | 200 | 82 | 0.743 |
| 0.707 | 0.9 | 200 | 82 | 0.730 |
| 0.707 | 0.95 | 200 | 82 | 0.719 |
| 0.707 | 0.98 | 200 | 82 | 0.685 |
| | | | | |
| 1.0 | 0 | 100 | 11 | 0.589 |
| 1.0 | 0.5 | 300 | 123 | 0.600 |
| 1.0 | 0.7 | 300 | 123 | 0.563 |
| 1.0 | 0.9 | 300 | 123 | 0.556 |
| 1.0 | 0.95 | 300 | 123 | 0.538 |
| 1.0 | 0.98 | 300 | 123 | 0.489 |
| | | | | |
| 1.414 | 0 | 200 | 22 | 0.366 |
| 1.414 | 0.5 | 300 | 123 | 0.351 |
| 1.414 | 0.7 | 300 | 123 | 0.311 |
| 1.414 | 0.9 | 300 | 123 | 0.300 |
| 1.414 | 0.95 | 400 | 164 | 0.275 |
| 1.414 | 0.98 | 400 | 164 | 0.250 |

Table 5 (Continued)

| S/N | $\rho$ | L | T(sec.) | P |
|---|---|---|---|---|
| 2.0 | 0 | 300 | 35 | 0.116 |
| 2.0 | 0.5 | 300 | 129 | 0.1126 |
| 2.0 | 0.7 | 300 | 129 | 0.1040 |
| 2.0 | 0.9 | 300 | 129 | 0.0986 |
| 2.0 | 0.95 | 300 | 129 | 0.0865 |
| 2.0 | 0.98 | 300 | 129 | 0.0783 |

Total Time = 2446 sec. = 40.77 min.

## Table 6

## Computed Word-Error Probabilities For 3-Bit Words, n = 2

| S/N | $\rho$ | L | T(sec.) | P |
|---|---|---|---|---|
| 0.707 | 0 | 700 | 18 | .416 |
| 0.707 | 0.5 | 400 | 15.6 | .415 |
| 0.707 | 0.7 | 400 | 15.6 | .415 |
| 0.707 | 0.9 | 400 | 15.6 | .367 |
| 0.707 | 0.95 | 400 | 15.6 | .356 |
| 0.707 | 0.98 | 400 | 15.6 | .298 |
| | | | | |
| 1.0 | 0 | 700 | 18 | .302 |
| 1.0 | 0.5 | 300 | 11.7 | .281 |
| 1.0 | 0.7 | 300 | 11.7 | .268 |
| 1.0 | 0.9 | 300 | 11.7 | .235 |
| 1.0 | 0.95 | 300 | 11.7 | .219 |
| 1.0 | 0.98 | 300 | 11.7 | .177 |
| | | | | |
| 1.414 | 0 | 1500 | 42 | .153 |
| 1.414 | 0.5 | 400 | 15.6 | .146 |
| 1.414 | 0.7 | 400 | 15.6 | .124 |
| 1.414 | 0.9 | 400 | 15.6 | .0925 |
| 1.414 | 0.95 | 400 | 15.6 | .0860 |
| 1.414 | 0.98 | 400 | 15.6 | .0666 |

Table 6 (Continued)

| S/N | $\rho$ | L | T (sec.) | P |
|---|---|---|---|---|
| 2.0 | 0 | 1500 | 42 | .0477 |
| 2.0 | 0.5 | 400 | 19.4 | .0449 |
| 2.0 | 0.7 | 400 | 19.4 | .0372 |
| 2.0 | 0.9 | 400 | 19.4 | .0262 |
| 2.0 | 0.95 | 400 | 19.4 | .0187 |
| 2.0 | 0.98 | 400 | 19.4 | .0159 |

Total Time = 431.5 sec. = 7.19 min.

Table 7

Detailed Results For S/N = 1.0, $\rho = 0.9$

| CT = No. of Demodulations | PEM = $p_M$ = 1-Q | AVRM = $P_2$ |
|---|---|---|
| 1 | .37659343E 00 | .62340657E 00 |
| 2 | .66810828E 00 | .47764915E 00 |
| 3 | .43912477E 00 | .50539118E 00 |
| 4 | .13976964E 00 | .59410097E 00 |
| 5 | .23474571E 00 | .62833164E 00 |
| 6 | .13800351E 00 | .66727578E 00 |
| 7 | .38229535E 00 | .66019420E 00 |
| 8 | .21535571E 00 | .67575046E 00 |
| 9 | .29519363E 00 | .67897889E 00 |
| 10 | .34163614E 00 | .67691739E 00 |
| 11 | .57835601E 00 | .65371072E 00 |
| 12 | .47914741E 00 | .64263921E 00 |
| 13 | .48944151E 00 | .63247915E 00 |
| 14 | .39589334E 00 | .63045255E 00 |
| 15 | .69615895E 00 | .60867845E 00 |
| 16 | .64513706E 00 | .59281498E 00 |
| 17 | .48543736E 00 | .58821190E 00 |
| 18 | .51775474E 00 | .58232487E 00 |
| 19 | .34181543E 00 | .58631749E 00 |
| 20 | .57160394E 00 | .57842142E 00 |
| 21 | .22904922E 00 | .58758949E 00 |
| 22 | .20530006E 00 | .59700360E 00 |
| 23 | .31439529E 00 | .60085583E 00 |
| 24 | .28460506E 00 | .60562829E 00 |
| 25 | .53445961E 00 | .60002477E 00 |
| 26 | .61289126E 00 | .59183570E 00 |
| 27 | .78004861E 00 | .57806221E 00 |

Table 7 (Continued)

| CT = No. of Demodulations | PEM = $p_M$ = 1-Q | AVRM = $P_2$ |
|---|---|---|
| 28 | .45510860E 00 | .57687754E 00 |
| 29 | .24858699E 00 | .58289601E 00 |
| 30 | .49126271E 00 | .58042405E 00 |
| 31 | .36532862E 00 | .58217397E 00 |
| 32 | .21306201E 00 | .58857284E 00 |
| 33 | .33297767E 00 | .59095010E 00 |
| 34 | .42444362E 00 | .59049734E 00 |
| 35 | .42559066E 00 | .59003769E 00 |
| 36 | .45955130E 00 | .58866022E 00 |
| 37 | .53134900E 00 | .58541673E 00 |
| 38 | .33658686E 00 | .58746926E 00 |
| 39 | .39330582E 00 | .58796222E 00 |
| 40 | .66824244E 00 | .58155711E 00 |
| 41 | .45655217E 00 | .58062761E 00 |
| 42 | .17261373E 00 | .58650283E 00 |
| 43 | .47958811E 00 | .58496583E 00 |
| 44 | .35108143E 00 | .58641930E 00 |
| 45 | .58004863E 00 | .58272001E 00 |
| 46 | .23956098E 00 | .58658347E 00 |
| 47 | .30785554E 00 | .58882946E 00 |
| 48 | .62420302E 00 | .58439128E 00 |
| 49 | .43415309E 00 | .58401282E 00 |
| 50 | .74533676E 00 | .57742584E 00 |
| 51 | .28134660E 00 | .58019501E 00 |
| 52 | .15484132E 00 | .58529046E 00 |
| 53 | .35085720E 00 | .58649523E 00 |
| 54 | .81592303E 00 | .57904304E 00 |
| 55 | .57752253E 00 | .57619639E 00 |

Table 7 (Continued)

| CT = No. of Demodulations | PEM = $p_M$ = 1-Q | AVRM = $P_2$ |
|---|---|---|
| 56 | .65355828E 00 | .57209363E 00 |
| 57 | .31350663E 00 | .57410064E 00 |
| 58 | .69666915E 00 | .56943220E 00 |
| 59 | .49165632E 00 | .56839680E 00 |
| 60 | .44067775E 00 | .56824556E 00 |
| 61 | .76408508E 00 | .56279752E 00 |
| 62 | .76661388E 00 | .55748443E 00 |
| 63 | .72934165E 00 | .55293164E 00 |
| 64 | .17451296E 00 | .55719032E 00 |
| 65 | .24180973E 00 | .56028263E 00 |
| 66 | .67112135E 00 | .55677652E 00 |
| 67 | .35238759E 00 | .55813228E 00 |
| 68 | .44011146E 00 | .55815810E 00 |
| 69 | .51846281E 00 | .55704766E 00 |
| 70 | .35828228E 00 | .55825723E 00 |
| 71 | .83712418E 00 | .55268848E 00 |
| 72 | .31466457E 00 | .55453081E 00 |
| 73 | .61190619E 00 | .55225085E 00 |
| 74 | .42824259E 00 | .55251446E 00 |
| 75 | .30549955E 00 | .55440761E 00 |
| 76 | .74486751E 00 | .55046978E 00 |
| 77 | .40557059E 00 | .55104069E 00 |
| 78 | .56534718E 00 | .54954854E 00 |
| 79 | .81449752E 00 | .54494036E 00 |
| 80 | .43801969E 00 | .54515336E 00 |
| 81 | .45683171E 00 | .54512887E 00 |
| 82 | .41739225E 00 | .54558592E 00 |

Table 7 (Continued)

| CT = No. of Demodulations | PEM = $p_M$ = 1-Q | AVRM = $P_2$ |
|---|---|---|
| 83 | .29496267E 00 | .54750703E 00 |
| 84 | .17571313E 00 | .55080203E 00 |
| 85 | .45933786E 00 | .55068274E 00 |
| 86 | .91628870E 00 | .54525284E 00 |
| 87 | .53854378E 00 | .54428966E 00 |
| 88 | .41588623E 00 | .54474221E 00 |
| 89 | .38474651E 00 | .54553448E 00 |
| 90 | .34859136E 00 | .54671086E 00 |
| 91 | .23439055E 00 | .54911634E 00 |
| 92 | .15104613E 00 | .55237544E 00 |
| 93 | .30322347E 00 | .55392815E 00 |
| 94 | .44172963E 00 | .55397435E 00 |
| 95 | .26579398E 00 | .55587152E 00 |
| 96 | .70673721E 00 | .55313601E 00 |
| 97 | .37986711E 00 | .55382670E 00 |
| 98 | .78444912E 00 | .55037491E 00 |
| 99 | .23153388E 00 | .55257785E 00 |
| 100 | .13079321E 00 | .55574415E 00 |

Table 8

Detailed Results For S/N = 2.0, $\rho = 0.98$

| CT = No. of Demodulations | PEI = $Q \frac{h}{h^*}$ | PEM = Q | AVRI = $P_3$ |
|---|---|---|---|
| 1 | .887244E-02 | .121695E-01 | .887244E-02 |
| 2 | .122609E-01 | .877047E-02 | .105667E-01 |
| 3 | .105259E-01 | .396113E-01 | .105531E-01 |
| 4 | .167040E-03 | .175603E-03 | .795658E-02 |
| 5 | .248927E-02 | .436259E-02 | .686312E-02 |
| 6 | .195185E-02 | .945304E-02 | .604457E-02 |
| 7 | .196621E 00 | .367094E 00 | .332697E-01 |
| 8 | .339279E-02 | .300901E-02 | .295351E-01 |
| 9 | .200014E 00 | .413194E 00 | .484772E-01 |
| 10 | .384768E 00 | .328324E 00 | .821063E-01 |
| 11 | .923001E-02 | .105449E-01 | .754812E-01 |
| 12 | .298305E-01 | .462027E-01 | .716770E-01 |
| 13 | .219430E-03 | .126049E-03 | .661802E-01 |
| 14 | .356502E-03 | .264585E-03 | .614785E-01 |
| 15 | .258250E-03 | .266649E-03 | .573972E-01 |
| 16 | .354243E-02 | .161722E-02 | .540313E-01 |
| 17 | .916238E-02 | .534345E-02 | .513919E-01 |
| 18 | .286576E 00 | .212577E 00 | .644577E-01 |
| 19 | .922903E-03 | .100849E-02 | .611137E-01 |
| 20 | .123828E 00 | .137772E 00 | .642494E-01 |
| 21 | .659648E-02 | .118151E-01 | .615041E-01 |
| 22 | .238620E-01 | .203225E-01 | .597931E-01 |
| 23 | .247468E-02 | .223776E-02 | .573009E-01 |
| 24 | .474423E-01 | .246194E-01 | .568902E-01 |
| 25 | .129946E-04 | .137761E-04 | .546151E-01 |
| 26 | .132678E 00 | .211830E 00 | .576175E-01 |
| 27 | .237831E-01 | .565135E-01 | .563644E-01 |

Table 8 (Continued)

| CT = No. of Demodulations | PEI = Q $\frac{h}{h^*}$ | PEM = Q | AVRI = $P_3$ |
|---|---|---|---|
| 28 | .393097E-02 | .399855E-02 | .544918E-01 |
| 29 | .184877E 00 | .458020E 00 | .589878E-01 |
| 30 | .165717E 00 | .109082E 00 | .625455E-01 |
| 31 | .521823E-01 | .327755E 00 | .622112E-01 |
| 32 | .863306E-01 | .398227E 00 | .629649E-01 |
| 33 | .184107E-01 | .801113E-02 | .616148E-01 |
| 34 | .459159E-02 | .420560E-02 | .599376E-01 |
| 35 | .411653E 00 | .571774E 00 | .699866E-01 |
| 36 | .185737E 00 | .101851E 00 | .732019E-01 |
| 37 | .446248E-02 | .496939E-02 | .713441E-01 |
| 38 | .101331E 00 | .139497E 00 | .721332E-01 |
| 39 | .556297E-02 | .234813E-01 | .704263E-01 |
| 40 | .565422E-02 | .251931E-01 | .688070E-01 |
| 41 | .171904E-01 | .133176E-01 | .675480E-01 |
| 42 | .226738E 00 | .306709E 00 | .713383E-01 |
| 43 | .645816E-02 | .123730E-01 | .698294E-01 |
| 44 | .326665E-01 | .549806E-01 | .689848E-01 |
| 45 | .208019E-02 | .222058E-02 | .674980E-01 |
| 46 | .147161E-01 | .820509E-02 | .663506E-01 |
| 47 | .501775E-01 | .334110E-01 | .660065E-01 |
| 48 | .338054E-03 | .238009E-03 | .646384E-01 |
| 49 | .616801E-01 | .122926E 00 | .645780E-01 |
| 50 | .227685E-01 | .226173E-01 | .637418E-01 |
| 51 | .686395E-02 | .677930E-02 | .626266E-01 |
| 52 | .297635E 00 | .369936E 00 | .671460E-01 |
| 53 | .273606E 00 | .305751E 00 | .710414E-01 |
| 54 | .457260E-02 | .245342E-02 | .698105E-01 |
| 55 | .198230E-01 | .390167E-01 | .689017E-01 |

Table 8 (Continued)

| CT = No. of Demodulations | PEI = $Q\,\frac{h}{h^*}$ | PEM = Q | AVRI = $P_3$ |
|---|---|---|---|
| 56 | .290463E-03 | .397258E-03 | .676765E-01 |
| 57 | .694242E-02 | .385579E-02 | .666110E-01 |
| 58 | .190171E-01 | .238055E-01 | .657904E-01 |
| 59 | .479549E-02 | .451062E-02 | .64756E-01 |
| 60 | .192055E 00 | .606313E 00 | .668782E-01 |
| 61 | .190934E 00 | .341179E 00 | .689119E-01 |
| 62 | .164464E-01 | .112258E-01 | .680657E-01 |
| 63 | .768579E-01 | .434691E-01 | .682053E-01 |
| 64 | .169490E-02 | .163029E-02 | .671660E-01 |
| 65 | .584541E 00 | .253737E 00 | .751256E-01 |
| 66 | .538745E-01 | .315460E 00 | .748037E-01 |
| 67 | .909178E-01 | .546219E-01 | .750442E-01 |
| 68 | .231297E 00 | .274067E 00 | .773420E-01 |
| 69 | .444353E-01 | .120110E 00 | .768651E-01 |
| 70 | .227451E-01 | .254028E-01 | .760919E-01 |
| 71 | .157747E-01 | .137936E-01 | .752424E-01 |
| 72 | .181212E 00 | .413611E 00 | .767142E-01 |
| 73 | .364642E-04 | .256747E-04 | .756638E-01 |
| 74 | .817412E-01 | .110805E 00 | .757460E-01 |
| 75 | .158362E 00 | .269423E 00 | .768475E-01 |
| 76 | .436904E-02 | .277011E-01 | .758938E-01 |
| 77 | .273002E-02 | .198131E-02 | .749437E-01 |
| 78 | .318554E-01 | .187426E-01 | .743912E-01 |
| 79 | .228797E-01 | .212824E-01 | .737392E-01 |
| 80 | .238722E 00 | .146937E 00 | .758015E-01 |
| 81 | .692196E-01 | .302460E-01 | .757202E-01 |
| 82 | .785796E-05 | .226572E-04 | .747969E-01 |

Table 8 (Continued)

| CT = No. of Demodulations | PEI = $Q \frac{h}{h^*}$ | PEM = Q | AVRI = $P_3$ |
|---|---|---|---|
| 83 | .419095E-01 | .195260E-01 | .744007E-01 |
| 84 | .143300E-01 | .144276E-01 | .736855E-01 |
| 85 | .150250E-01 | .358517E-01 | .729954E-01 |
| 86 | .373712E 00 | .445742E 00 | .764921E-01 |
| 87 | .443849E-02 | .269791E-02 | .756639E-01 |
| 88 | .563731E 00 | .423997E 00 | .812101E-01 |
| 89 | .109039E-01 | .109618E 00 | .804202E-01 |
| 90 | .196468E 00 | .558084E 00 | .817096E-01 |
| 91 | .150477E-02 | .319158E-02 | .808282E-01 |
| 92 | .311043E-02 | .113983E-01 | .799835E-01 |
| 93 | .208261E-03 | .145949E-03 | .791257E-01 |
| 94 | .964372E-02 | .101269E-01 | .783865E-01 |
| 95 | .142038E-01 | .800990E-02 | .777109E-01 |
| 96 | .198894E-01 | .443432E-01 | .771086E-01 |
| 97 | .237481E 00 | .480541E 00 | .787619E-01 |
| 98 | .521661E-02 | .722164E-02 | .780114E-01 |
| 99 | .693649E-01 | .275221E-01 | .779241E-01 |
| 100 | .813598E-03 | .225443E-02 | .771530E-01 |

Table 9

Detailed Results For S/N = 1.414, $\rho = 0$ (or n = 1)

| CT = No. of Demodulations | PEM = Q | AVRM = $P_2$ |
|---|---|---|
| 1 | .838906E 00 | .838906E 00 |
| 2 | .496383E-01 | .444272E 00 |
| 3 | .758264E 00 | .548936E 00 |
| 4 | .763433E-01 | .430788E 00 |
| 5 | .146204E 00 | .373871E 00 |
| 6 | .382893E 00 | .375375E 00 |
| 7 | .292061E 00 | .363473E 00 |
| 8 | .662838E 00 | .400893E 00 |
| 9 | .230133E 00 | .381920E 00 |
| 10 | .602794E-01 | .349756E 00 |
| 11 | .281032E 00 | .343508E 00 |
| 12 | .555666E-01 | .319513E 00 |
| 13 | .205799E 00 | .310766E 00 |
| 14 | .282232E-01 | .290584E 00 |
| 15 | .430902E 00 | .299939E 00 |
| 16 | .381355E 00 | .305027E 00 |
| 17 | .385803E 00 | .309779E 00 |
| 18 | .408660E 00 | .315272E 00 |
| 19 | .653269E 00 | .333062E 00 |
| 20 | .208512E 00 | .326834E 00 |
| 21 | .805447E 00 | .349625E 00 |
| 22 | .493588E 00 | .356169E 00 |
| 23 | .488309E 00 | .361914E 00 |
| 24 | .211028E 00 | .355627E 00 |
| 25 | .200154E 00 | .349408E 00 |
| 26 | .566857E-01 | .338150E 00 |
| 27 | .324982E 00 | .337662E 00 |

Table 9 (Continued)

| CT = No. of Demodulations | PEM = Q | AVRM = $P_2$ |
|---|---|---|
| 28 | .114252E 00 | .329683E 00 |
| 29 | .189953E 00 | .324865E 00 |
| 30 | .604415E-01 | .316051E 00 |
| 31 | .431587E 00 | .319778E 00 |
| 32 | .336366E 00 | .320296E 00 |
| 33 | .135335E 00 | .314691E 00 |
| 34 | .194643E 00 | .311160E 00 |
| 35 | .410337E 00 | .313994E 00 |
| 36 | .580940E 00 | .321409E 00 |
| 37 | .289787E 00 | .320555E 00 |
| 38 | .290345E 00 | .319760E 00 |
| 39 | .122533E 00 | .314702E 00 |
| 40 | .259594E 00 | .313325E 00 |
| 41 | .211219E 00 | .310834E 00 |
| 42 | .726527E 00 | .320732E 00 |
| 43 | .466344E 00 | .324118E 00 |
| 44 | .433417E-01 | .317737E 00 |
| 45 | .642568E 00 | .324955E 00 |
| 46 | .400129E 00 | .326590E 00 |
| 47 | .451392E 00 | .329245E 00 |
| 48 | .101668E-01 | .322597E 00 |
| 49 | .516780E 00 | .326560E 00 |
| 50 | .657325E 00 | .333176E 00 |
| 51 | .262141E 00 | .331783E 00 |
| 52 | .420233E 00 | .333484E 00 |
| 53 | .445581E 00 | .335599E 00 |
| 54 | .121936E 00 | .331642E 00 |
| 55 | .456685E 00 | .333916E 00 |
| 56 | .487427E 00 | .336657E 00 |

Table 9 (Continued)

| CT = No. of Demodulations | PEM = Q | AVRM = $P_2$ |
|---|---|---|
| 57 | .806038E 00 | .344892E 00 |
| 58 | .400024E 00 | .345842E 00 |
| 59 | .423723E 00 | .347162E 00 |
| 60 | .446713E 00 | .348821E 00 |
| 61 | .605045E 00 | .353022E 00 |
| 62 | .590971E 00 | .356860E 00 |
| 63 | .285580E 00 | .355728E 00 |
| 64 | .371234E 00 | .355970E 00 |
| 65 | .562408E 00 | .359146E 00 |
| 66 | .826602E-01 | .354957E 00 |
| 67 | .294445E-01 | .350099E 00 |
| 68 | .370704E 00 | .350402E 00 |
| 69 | .182951E 00 | .347975E 00 |
| 70 | .422965E 00 | .349046E 00 |
| 71 | .519226E 00 | .351443E 00 |
| 72 | .618235E 00 | .355149E 00 |
| 73 | .275129E 00 | .354052E 00 |
| 74 | .678303E 00 | .358434E 00 |
| 75 | .278614E 00 | .357370E 00 |
| 76 | .294648E 00 | .356545E 00 |
| 77 | .251829E-01 | .352241E 00 |
| 78 | .695051E 00 | .356636E 00 |
| 79 | .193674E 00 | .354573E 00 |
| 80 | .790583E 00 | .360024E 00 |
| 81 | .153728E 00 | .357477E 00 |
| 82 | .547047E 00 | .359789E 00 |
| 83 | .702893E-01 | .356301E 00 |
| 84 | .731346E 00 | .360765E 00 |
| 85 | .689068E 00 | .364628E 00 |

Table 9 (Continued)

| CT = No. of Demodulations | PEM = Q | AVRM = $P_2$ |
|---|---|---|
| 86 | .266668E 00 | .363489E 00 |
| 87 | .360057E 00 | .363449E 00 |
| 88 | .708612E-01 | .360124E 00 |
| 89 | .359297E 00 | .360115E 00 |
| 90 | .706486E-01 | .356899E 00 |
| 91 | .244928E 00 | .355668E 00 |
| 92 | .396508E 00 | .356112E 00 |
| 93 | .759087E-01 | .353099E 00 |
| 94 | .472791E 00 | .354373E 00 |
| 95 | .312459E 00 | .353931E 00 |
| 96 | .599942E 00 | .356494E 00 |
| 97 | .238840E 00 | .355281E 00 |
| 98 | .665262E 00 | .358444E 00 |
| 99 | .662327E 00 | .361514E 00 |
| 100 | .215951E 00 | .360058E 00 |

Table 10

Calculated Word-Error Probabilities For No Inter-Bit Dependence

| $S/N = \dfrac{S}{K\sqrt{B/2}}$ | | $P_E$ | $P_W$ (3-bits) | $P_W$ (6-bits) |
|---|---|---|---|---|
| .707 | (-3 db) | .24 | .561 | .808 |
| 1.0 | (0 db) | .159 | .404 | .645 |
| 1.414 | (+3 db) | .079 | .218 | .39 |
| 2.0 | (+6 db) | .0228 | .067 | .13 |
| 3.0 | (+9.54 db) | .00135 | .00405 | .0081 |
| 4.0 | (+12 db) | .000032 | .000096 | .00019 |

## BIBLIOGRAPHY


| Reference Number | |
|---|---|
| 1 | Davenport, Wilbur B., Jr., and Root, William L. An Introduction to the Theory of Random Signals and Noise. New York: McGraw-Hill Book Company, Inc., 1958. |
| 2 | Dixon, Wilfrid J., and Massey, Frank J., Jr. Introduction to Statistical Analysis. New York: McGraw-Hill Book Company, Inc., 1957. |
| 3 | Dwork, B.M. "Detection of a Pulse Superimposed on Fluctuation Noise." Proceedings of the IRE, vol. 38 (July 1950), pp. 771-774 |
| 4 | Guillemin, E. A., The Mathematics of Circuit Analysis. New York: John Wiley & Sons, Inc., 1949. |
| 5 | Kahn, H. and Marshall, A. W. "Methods of Reducing Sample Size in Monte Carlo Computations." Res. Soc. of Amer. I, 1953, pp. 263-271. |
| 6 | Laning, Halcombe J., Jr., and Battin, Richard H. Random Processes in Automatic Control. New York: McGraw-Hill Book Company, Inc., 1956. |
| 7 | Lawton, John G. "Comparisons of Binary Data Transmission Systems" Proceeding of 2nd National Conference on Military Electronics. 1958, pp 54-61. |
| 8 | McRae, D.D., and Smith, E.F. Computer Interpolation. Technical Report No. 2, Parts 1 and 2, Advanced Communications Group, Radiation Incorporated, Melbourne, Florida, Part 1-15 December 1961, Part 2-10 May 1962. |
| 9 | Mood, Alexander M., Introduction to the Theory of Statistics. New York: McGraw-Hill Book Company, Inc., 1950. |
| 10 | Nichols, M.H., and Rauch, L. L. Radio Telemetry New York: John Wiley & Sons, Inc., Second Edition, 1956. |
| 11 | Rauch, L. L. Improved Demodulation and Predetection Recording. Research Report under USAF Contract No. AF 33 (616)-5796, Task No. 82048, Project No. 4107, University of Michigan, 28 February, 1961. |

| Reference Number | |
|---|---|
| 12 | Rice, S. O. "Properties of a Sine Wave Plus Random Noise." Bell System Technical Journal vol. 27 (January 1948) pp. 109-157. |
| 13 | Wiener, N. The Extrapolation, Interpolation, and Smoothing of Stationary Time Series With Engineering Applications. New York: John Wiley & Sons, Inc., 1949. |
| 14 | Woodward, P. M. Probability and Information Theory with Applications to Radar. New York: McGraw Hill Book Co., Inc., 1953. |
| 15 | Experimental Determination of Signal-to-Noise Relationships in PCM/FM and PCM/PM Transmission. Interim Report for NASA, GSFC, Contract NAS 5-505 Electro-Mechanical Research, Inc., Sarasota, Florida, 20 October 1961. |
| 16 | Telemetry System Study Final Report, Volume II of III. "Experimental Evaluation Program." U.S. Army Signal Research and Development Laboratories Contract No. DA-36-039 SC-73182 Aeronutronic; Newport Beach, California, 18 December, 1959. |