OKLAHOMA STATE UNIVERSITY

# ANALYSIS OF MULTIPATH FADING EFFECTS

By

RONALD N. ISAAC

Bachelor of Science

University of Balamand

El-Koura, Lebanon

1997

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
MAY, 1999

ANALYSIS OF MULTIPATH FADING EFFECTS

Thesis Approved:

Thesis Advisor

Dean of the Graduate College

# PREFACE

Multipath fading has a severe effect on the performance of wireless communication systems. Because of either refraction from the ionosphere or obstructions in the path of a transmitting and receiving device, received signals do involve varying delays. In assessing the impact of multipath propagation on various wireless systems, a quantitative estimate of the system outage is the figure of merit.

Systems outages are often assessed with equipment signatures (receiver's tolerance to sclective fading) and are highly dependent on the statistics of fading. Multipath fading effects are analyzed from a theoretical perspective. A four state Markov model is derived in parallel with a predefined three-state birth-death model. A formula for reacquisition hyserisis induced outages is shown to yield accurate predictions when compared with actual counts on simulated fades. These predictions are also shown to be more accurate than Ranade's empirical asymptotic result. Outages would be diminished if equalizers can be used to mitigate very large notch depths. A fundamental limit exists, however for the fade notch depths to which an equalizer can remove dispersion without enhancing the thermal noise to an unacceptable amount. A way to deduce the limit is provided and analyzed.

# ACKNOWLEDGMENTS

I wish to extend my sincere appreciation to my advisor, Dr. Scott R. Shepard for his guidance and friendship. I also appreciate all the time he has spared in expressing his thoughts and experiences. I would also like to extend my appreciation to my other committee members Dr. George Scheets, from whom I learnt the basics of communications, Dr. R. Yarlagadda and Dr. Chris Hutchens for all their guidance.

Furthermore, I would like to thank my parents and my Cousin, Nazi, for their love and kind financial support. Not forgetting my beloved fiancée, Claude, for standing by me all through out this period and guiding me through the difficulties. I also appreciate the encouragement and help of my dear friend, Husni Barbar, during this period of study.

Finally, I would like to express my gratitude to the Department of Electrical Engineering and Oklahoma State University as a whole for their friendship and support during these two years of study.

TABLE OF CONTENTS

# LIST OF FIGURES

*Figure* *Page*

# NOMENCLATURE

| | |
|---|---|
| LOS | Line of Sight |
| AMPS | Analog Mobile Phone System |
| PCS | Personal Communication System |
| GSM | Global System for Mobile Communications |
| BER | Bit Error Rate |
| b | Relative amplitude of delayed ray to direct ray |
| a | Amplitude of direct ray |
| $\omega_o$ or $f_0$ | Notch frequency |
| $\tau$ | Channel delay |
| B | Power level of b |
| A | Power level of a |
| C/N or S/N | Signal to Noise ratio |
| IF | Intermediate Frequency |
| RF | Radio frequency |
| PSK | Phase Shift Keying |
| C | Transmitted power correction term |
| q | Reacquisition induced outage |
| PDF | Probability density function |
| T | Low pass filter time constant |

| | |
|---|---|
| $H(f)$ | Transfer function of channel |
| $H_{eq}(f)$ | Transfer function of equalizer |
| $K$ | Equalization amplification factor |
| $G_{si}(f)$ | Power density of input signal |
| $G_{ni}(f)$ | Power density of input noise |
| $G_{nc}(f)$ | Power density of excess noise introduced by channel |
| $G_{so}(f)$ | Power density of output signal |
| $G_{no}(f)$ | Power density of output noise |
| $F$ | Noise figure |
| $S_i$ | Power of input signal |
| $N_i$ | Power of input noise |
| $S_o$ | Power of output signal |
| $N_o$ | Power of output noise |
| $T_1$ | Temperature of input thermal noise |
| $T_2$ | Temperature of channel thermal noise |
| $T_i$ | Logarithmic difference of $T_1$ and $T_2$ |
| $W_s$ | Logarithm of the inverse of 2 times the channel bandwidth |
| $A_e$ | Flat fading noise enhancement term |
| $B_e$ | Selective fading noise enhancement term |

*Chapter 1*

# INTRODUCTION

The growing popularity of worldwide communications has by no means triggered a lot of work in the field. Following the first messages sent by Marconi in 1895 via the air waves across the Atlantic, converged researchers to the introduction of microwave communications[1]. Modulating frequencies up to 4Ghz were used and put into service in 1948[2]. Bringing along a virtual end to the coaxial era, this way of communications provided the advantages of wireless communications.

Major use of such systems was and is still found in line-of-sight(LOS) radio links and is used extensively in the transmission of data, voice and video. Due to the curvature of the earth though, the coverage area of LOS propagation is limited[3] and thus the use of expensive repeaters en route is unavoidable. This drawback brought LOS relay systems to city centers with antennas mounted on tall towers and rooftops.

Within the expanding world, the need for more convenient means of communications systems is inevitable. This wireless technology is being extensively used today in the provision of data and voice services such as the vanishing US wireless telephone system known as the analog mobile phone system(AMPS); the more current and advanced personal communications system(PCS); and several global ones such as the global system for mobile communications(GSM)[4].

Although wireless communications is being used extensively over a wide range of applications, a lot of work has and is still being done on improving the transmission of signals through the unguided air channel. Signals transmitted do involve multipath components with varying delays along the channel. These out of phase components superimpose on each other resulting in severe fading of the received signal. Irregular variations in the refractive index of the atmosphere is the major cause of fading in microwave line-of-sight radio links[5] whereas obstructions in the path of communication between a mobile system and a base station is the major cause of fading in mobile systems[3]. These natural impediments cannot be avoided. Our goal as engineers is to reduce their impacts once they have been identified and properly characterized.

Chapter II introduces the transfer function of a multipath faded channel and describes much of the work done by Rummler et al.[6] in the measurement of equipment signatures, also known as M-curves, from laboratory and field testing. The method used in calculating the probability of outage duration using these curves is also explained. Chapter III introduces Shepard's[7] three state Markov model for calculating reacquisition induced outages due to hysterisis and a four state model is analyzed. Fade notch depths, characterized by exponential distributions, are simulated with a computer random generator and counting is performed on the data and compared to the derived four state model. The only previous attempt to predict hysterisis was done by Ranade[8], who showed an agreement between field data and an empirical formula within the limit of vanishing hysterisis. In chapter V we show that the formula from Shepard's three state model matches the simulated data better than Ranade's asymptote even in the limit of vanishing hysterisis. In chapter IV it will be shown that a

fundamental limit exists for the fade depth to which an equalizer can remove dispersion without enhancing the thermal noise induced by the channel.

Field simulation and testing of multipath fading is very expensive and difficult[9]. An easier approach is the use of hardware built simulators. This piece of hardware characterizes the channel transfer function and serves the very useful purpose of simulating the fading. In Appendix A, a description of the simulator and software/hardware D/A conversion unit is provided. In the future, the simulator/PC interface could be used to measure equipment signatures, described in chapter II. It could also be used to simulate probabilistic fading sequences from which the Markov parameters, described in chapter III, could be extracted to test hysterisis predictions.

# FADE MARGINS AND THE CALCULATION OF OUTAGES

Outage estimation is important in determining the performance of stationary digital radio communication systems as well as in mobile systems. The outage (predominantly due to multipath fading) is defined as the time that the bit error rate (BER) of interest exceeds a threshold. During the time when the system is in outage, it cannot be used for its purpose and hence its name.

Methods of predicting this outage serve as useful tools in this context. Many authors have presented such methods [6][10][11]. In this chapter, however, I will concentrate on the method developed by Rummler[6]. This will serve as a tool in understanding the use of fade margins in plotting what are known as M- curves. It will also present a better understanding of the hysterisis model presented by Shepard[7] and the four state model developed in chapter 3, and its usefulness and comparisons in later chapters.

## 2.1 The Channel Model

In reality, the number of physical rays reaching the receiver is N, the transfer function of such a model could be expressed:

$$H_A(\omega) = H_0 + \sum_{i=1}^{N} a_i e^{-j\omega\tau_i} \qquad (2.1)$$

$H_o$ is taken as a reference where $a_i$ and $\tau_i$ are the respective relative delay and amplitude of the

i $^{th}$ ray. This equation does represent the real life situation but with taking N=2 and $H_o$=1, experimental data fits nicely to it[5].

### 2.1.1 The two-ray model

The model proposed by [12] known as the two-ray model is somehow the most close to describing the transfer function of the multipath fading channel in digital radio LOS. With N=2 and $H_o$=1, as stated above, the general three-ray model results. This is given by:

$$H(\omega) = 1 + a_1 e^{-j\omega\tau_1} + a_2 e^{-j\omega\tau} \qquad \text{, where } \tau_2 = \tau \tag{2.1}$$

The exponential term could be visualized as a rotating vector with radius $a_1$. For $\omega$ fixed at an instant, and $\tau_1$ fixed, the transfer function can be represented by the resultant of the second ray and the resultant of the unity and first ray. By making $\tau_1$ sufficiently small, it is noticed that the phase $\omega\tau_1$ changes slightly as the angular frequency varies over the bandwidth of the carrier. Hence the vector $1 + a_1.e^{-j\omega\tau_1}$ will not vary much.

$$1 + a_1 e^{-j\omega\tau_1} \approx a e^{-j\phi} \text{, where } \phi = \omega_o\tau - \pi \tag{2.2}$$

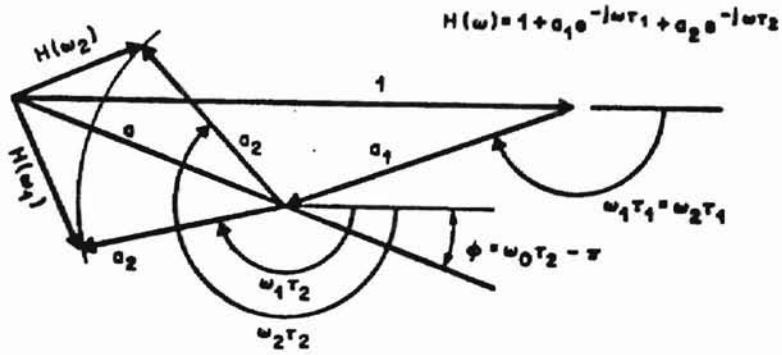



Figure 2.1[12] Vector representation of three rays

The notch frequency is referred to as $\omega_o$ and $\tau$ is the delay difference between the direct and indirect rays as seen reproduced in Figure 2.1. Using the simplified diagram of Figure 2.2 and assuming $b=a_2/a$, the complex voltage transfer function is given by:

$$H(\omega) = a[1 - be^{-j(\omega-\omega_o)\tau}] \tag{2.4}$$

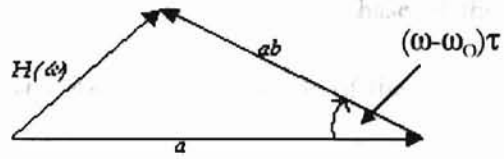



Figure 2.2 Simplified three-path fade

## 2.2 Parameters of the model

Following experiments and statistical fitting of data, $\tau$ the delay in the channel is said to be

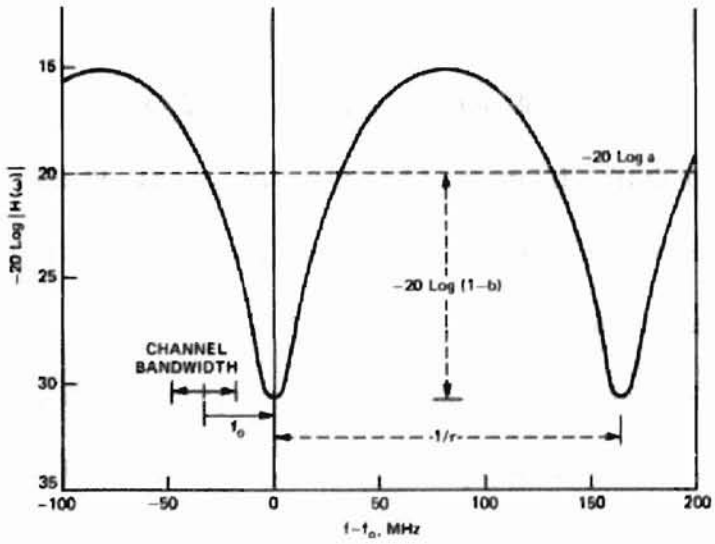



Figure 2.3[6] Power transfer function of channel

fixed at 6.3ns for a 30Mhz bandwidth operating at 6Ghz[6]. The channel power transfer function could be stated as:

$$|H(\omega)|^2 = a^2[1 + b^2 - 2b\cos(\omega - \omega_o)\tau] \quad (2.5)$$

The attenuation introduced by the channel is reproduced in Fig 2.3. Multiplying the delay, fixed at 6.3ns, with the notch frequency results in the phase of the signal. $\omega_o$ refers to the notch angular frequency and $\phi = \tau\omega_o$. After one cycle of the signal, the second null could be seen at $\phi + 2\pi$:

$$\phi = \tau\omega_o;\ \phi' = \phi + 2\pi;$$

$$\omega_o' = \phi' / \tau = \phi / \tau + 2\pi / \tau = \omega_o + 2\pi / \tau;$$

$$f_o' = \omega_o' / 2\pi = f_o + 1/\tau = f_o + 158.4\,Mhz \quad (2.6)$$

Therefore, assuming a 30 MHz bandwidth, the second null exists outside the bandwidth of interest. This model is simplified in the sense that the delay is fixed and does not vary. In reality, the delay varies[9]. The parameters, a and b, control the depth and shape of the curve shown in Fig 2.1. The frequency, $f_o$, denotes the position of the notch. The frequencies of operation and the notch frequency are measured from the center of the 30Mhz channel. The transfer function could be interpreted as the response of a channel to a direct ray with amplitude *a*, and a 6.3ns delayed path with relative amplitude b.

## 2.3 Probability distribution of parameters

Notch depths are best modeled in terms of B = -20log(1-b). From data collected over a period wherein severe fading was observed, the distribution of B for a line-of-sight radio link was best fit in the region between values of 3 and 23 dB. The distribution is uncertain for values of B exceeding 23 dB and occurred only about 15% of the time. The fitted line to the data collected, is modeled with a probability that B exceeded a value $x$ by:

$$P(B > x) = e^{-x/3.8} \tag{2.7}$$

Therefore the probability distribution function could be written as:

$$p_B(x) = \frac{1}{3.8} e^{-x/3.8} \tag{2.8}$$

The distribution of A= -20log a, however, was found to be log-normal with a standard deviation of 5 dB and a mean and variance that is dependent on B. Hence the probability density function is written as:

$$p_{A/B}(Y / B_0) = \frac{1}{\sigma(B_O)\sqrt{2\pi}} e^{-[Y - A_O(B_O)]^2 / 2\sigma^2(B_O)} \tag{2.9}$$

The dependence of the mean of A on B is reproduced in the Figure 2.4. The distribution of fo was found to be independent of A and B. Being easier to work with $\phi=2\pi\tau f_o$, the probability density function can be defined over two levels as seen below:

$$p_{\phi}(\phi) = \begin{cases} \frac{1}{216} & |\phi|<90^{\circ} \\ \frac{1}{1080} & 90\leq|\phi|\geq 180^{\circ} \end{cases} \tag{2.10}$$

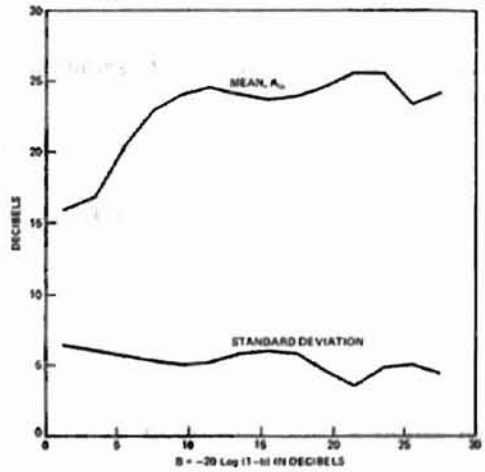



Fig 2.4[6] Variations of the mean and variance of A with B

The probability density functions in equations (2.2) - (2.4) serve as a set of valuable tools in determining the impact of the parameters in the a-b-$f_o$ space. They are also of importance in simulating data for determining outages as will be seen in the coming chapters. In this chapter, Rummler makes use of them in calculating outages from a laboratory setup wherein an a-b-$f_o$ space is plotted.

## 2.3 Outage estimation from experiments

The two-ray model described above can be simulated in the laboratory to determine the response to multipath fading. The idea is to determine critical values of A and B at a specified BER. Since it is practically difficult to fix the BER and the simultaneous effects of A and B are desired to be noticed, B is varied while BER is plotted against carrier-to-noise(C/N) ratio. These critical values yield critical contours for the designated BER as a function of fade notch positions. The probability distributions of the various parameters can then be used on these contours to compute probabilities that A and B lie in a specific region of the plots which correspond to an outage. By varying the notch fade position over a uniform range and computing the corresponding critical contours, the probability of an outage can be calculated.

### 2.3.1 Plotting BER vs. C/N

A pseudo-random data pattern is modulated onto a 6Ghz RF carrier[6]. The RF carrier is further down converted to an intermediate frequency (IF), pre-amplified and passed through an IF fade simulator. The resulting output is then demodulated and sent to a BER test receiver as seen in the diagram below:

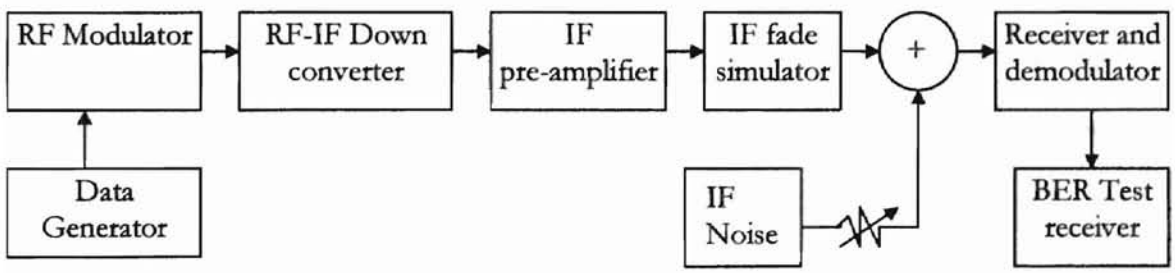



Fig 2.5 Block diagram of laboratory setup

The IF fade simulator in Fig 2.5 is adjusted for different notch positions by simply tuning the attenuators controlling the phase and amplitude of the delayed component. IF thermal noise is added and BERs are plotted versus the C/N for a constant fade notch depth, B. As seen reproduced in Fig 2.6 below, from a laboratory setup, the baseline curve which refers to that curve with B=0, is faded only due to flat fading caused by the IF thermal noise source.

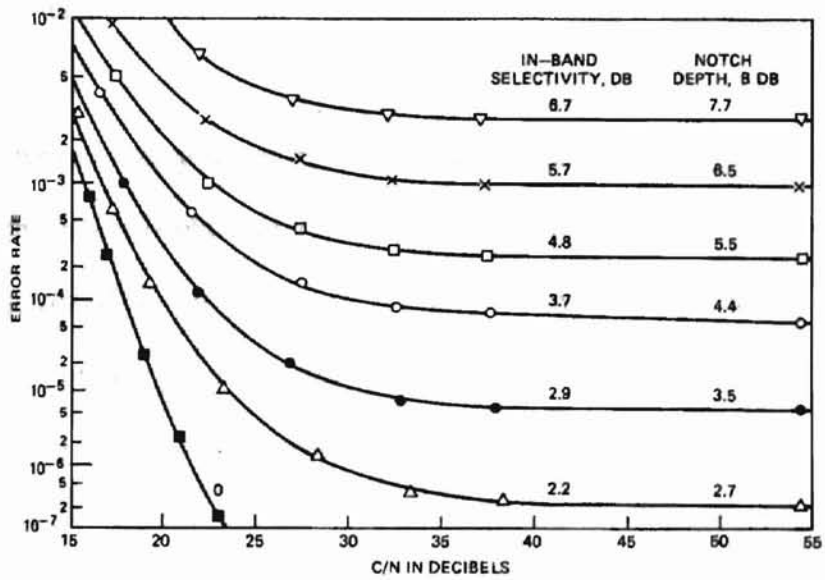



Fig 2.6[6] BER versus C/N

The baseline curve in Fig 2.6 maps a value of C/N = 21.5dB for a BER of $10^{-6}$. For the system setup, the measured flat fade margin for BER= $10^{-6}$ was 40.5dB. This implies that the unfaded IF C/N = 21.5 + 40.5 = 62 dB. Curves similar to that of Fig 2.6 were collected by Rummler[6] for different notch positions along the bandwidth of the channel. In the next section, it will be seen how all of the above contributes to the calculation of outages.

## 2.4 Calculation of outage

The method presented by Rummler[6] in calculating outages, is based on the critical curves of A and B. The next subsection will show their derivation from the BER vs. C/N curves and the following section will give a detailed view of Rummler's approach in calculating outages using the derived critical A-B curves.

### 2.4.1 Derivation of A-B curves

Referring to Figure 2.6 reproduced from [6], the points that cross the critical BER for the six different notch depths are obtained. For the curve corresponding to B=4.4 dB, the C/N ratio corresponding to a BER $=10^{-3}$ is 20.2 dB. The carrier to noise ratio of the unfaded channel is 62 dB and therefore the relative average power loss is 62 – 20.2 = 41.8 dB. Rummler then decides to assume a PSK signal with a rectangular spectrum of width $f_b$ to introduce a correction term in the determination of A. Consequently the average power transmitted by such a signal is:

$$P_{av} = a^2\{1 + b^2 - 2b\cos 2\pi f_o(\frac{\sin \pi f_b \tau}{\pi f_b \tau})\} \tag{2.11}$$

Taking logarithms on both sides,

$$10Log(P_{av}) = 20Loga + 10Log(1 + b^2 - 2b\cos 2\pi f_o(\frac{\sin \pi f_b \tau}{\pi f_b \tau})) \tag{2.12}$$

and if C, a correction term is defined as,

$$C = -10Log(1 + b^2 - 2b\cos 2\pi f_o(\frac{\sin \pi f_b \tau}{\pi f_b \tau})) \quad (2.13)$$

then $L_s$ = -10 Log ($P_{av}$)=A + C, which implies A= $L_s$ – C. For B= 4.4 dB and $f_o$ = -19.8Mhz, C = 2.06 dB and A is found to be equal to 41.8 – 2.1 = 39.7 dB. Carrying out these calculations on the rest of the B curves and $f_o$ = -19.8 Mhz, the critical curve of A versus B is generated for BER =$10^{-3}$. The curve is reproduced in Figure 2.7. A complete set of curves is then generated for different notch frequencies and BERs.

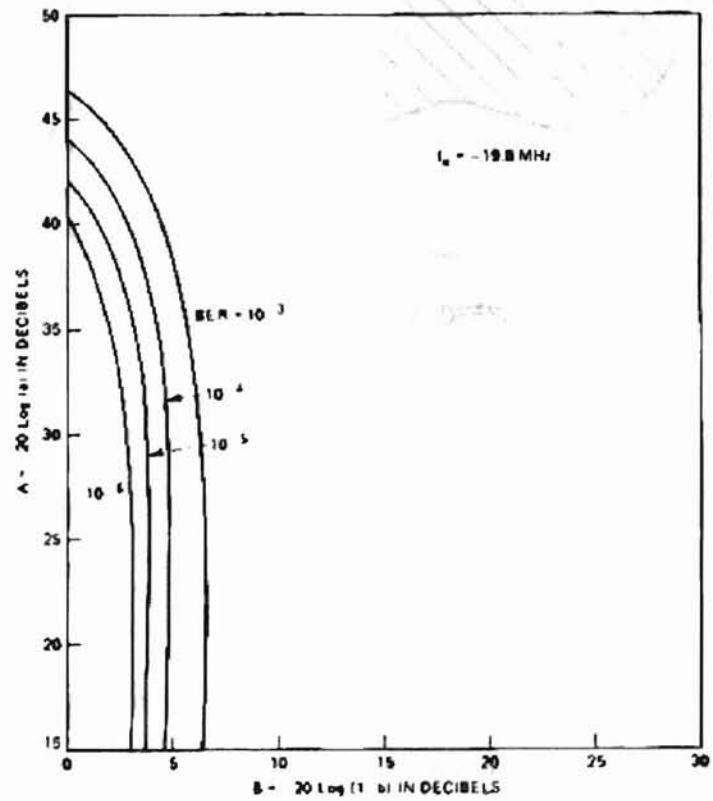



Figure 2.7[6] Critical curves of A and B

### 2.4.2 M-Curves

In later studies, Rummler et al found that the outage time was dominated by the frequency selectivity. Accordingly, they removed the A parameter from consideration and plotted the notch depth, B, versus the notch frequency offset from the center of the channel and is reproduced in Figure 2.8 with a general shape of 'W'.

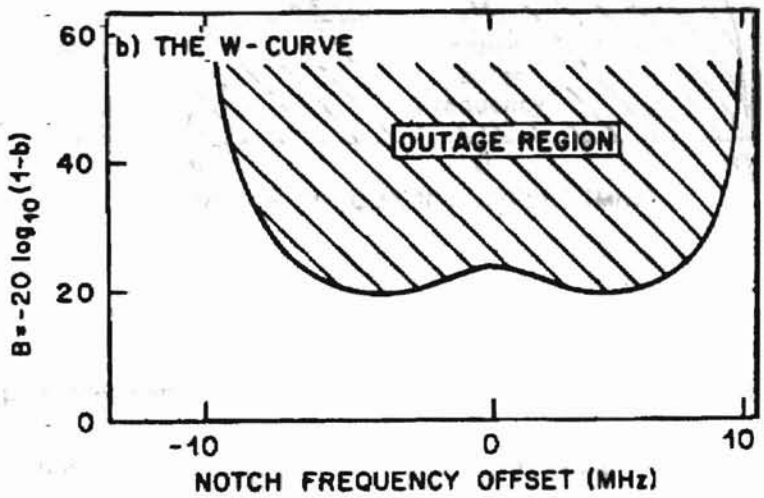



Figure 2.8[14] W-curve of B vs. $\Delta f_o$

The outage as stated in the diagram is calculated as the area bounded by the curve. In [14] the critical values of (1-b) are plotted versus notch frequency, $f_o$ resulting in the general M-shaped curve as shown reproduced in Figure 2.9. It is also noticed that flipping the ordinate axis results in the W-curve. Whatever way is used in plotting these signatures, the outage result is the same and is calculated as:

$$P_{outage} = \int_{-\pi}^{\pi} \int_{B_c}^{\infty} p_{\phi}(\phi).p_B(X).dX\, d\phi \qquad (2.14)$$

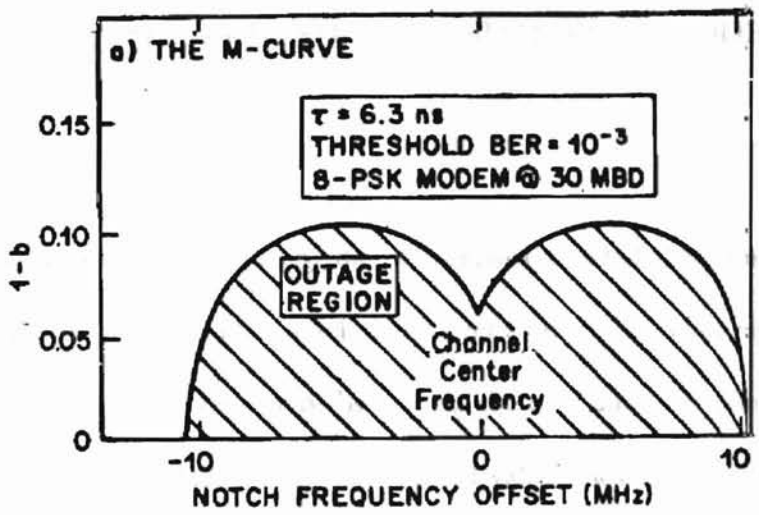



Figure 2.9[14] M-curve of (1-b) versus $f_o$

In the next chapter, the shapes of these equipment signatures will be used in simulating an exponential distribution for the fade notch parameter and then used in estimating outages from the a four state model deduced from that of Shepards[7] three state model.

*Chapter 3*

# REACQUSITION HYSTERISIS AND MARKOV MODEL

The M curves play a vital role in assessing the performance of a faded communication system. They characterize critical values of the notch depth over a notch frequency range and are often used in predicting outages of the system by finding the area under its contour in the B $-f_o$ plane. When these notch depths exceed those critical values, however, the BER increases at which carrier recovery is lost and the system is said to lose lock. The system never reacquires lock until the notch depth becomes small enough that is less than its critical value at which it lost lock. Hence, outages of this type are not accounted by the M-curve alone. Shepard[7] introduced a simple formula of a steady-state analysis Markov model predicting reacquisition induced outages. Reacquisition is fundamentally a three-state problem and the motivation for four-state is to get closer to continuum. In this chapter a four-state model is introduced.

## 3.1 THE HYSTERISIS PROBLEM

"The fact that a receiver does not reacquire lock at the same fade depth at which it loses it", is how Shepard[7] defines reacquisition hysterisis. Considering the M –curves outages alone, adaptive equalizers from field tests fail to deliver the anticipated improvement to their M – curves[13]. Laboratory simulations can be done but it is quite close to impossible to isolate these induced outages from those characterized by the equipment's signatures. Some authors

have decided to ignore this effect probably because of its complexity while others such as Greenstein and Shafi [14] have put it as: "This hysterisis effect is ignored in most existing methods of outage time estimation, the implicit assumption being that such periods are relatively short.". Shepard[7], however, shows that is not the case. In designing the Markov model, the l-curve, the notch depth/notch frequency contour at which the system loses lock, and the r-curve, the contour at which it regains lock, are defined and is as shown in Figure 3.1.

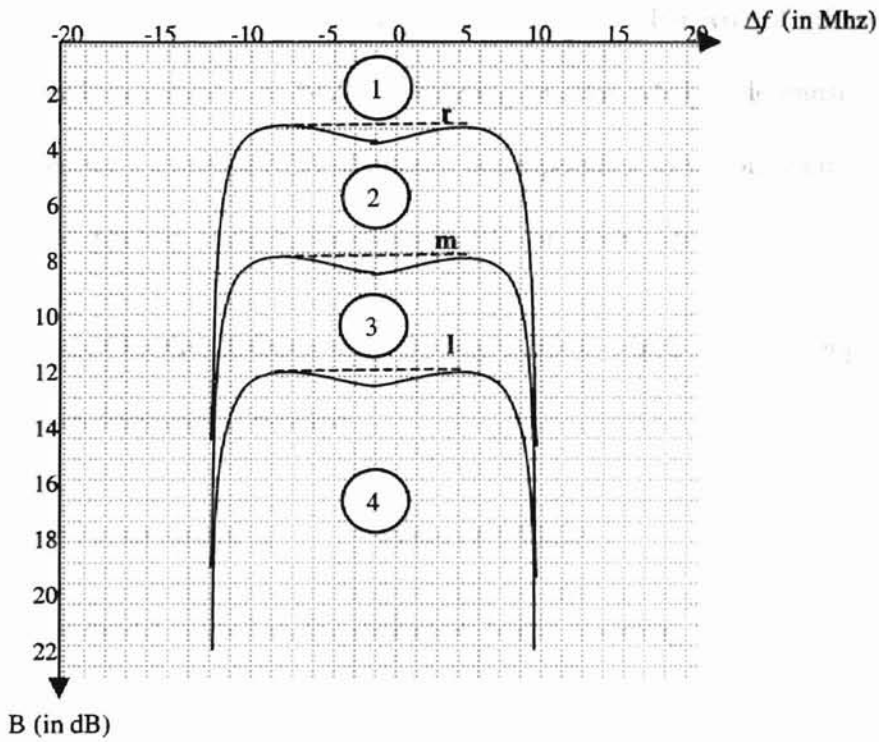



Fig 3.1 Equipment Signatures in the B-$f_o$ plane

As could be seen in Fig 3.1, the curves are partitioned into four regions. The outage is proportional to $P_4$, the probabilities of being in region 4. This is definitely the m- curve outage. The induced outage, however, is defined to be proportional to the conditional probability of being in regions 2 or 3 given that we were in region 4 before going back to

region 1. This conditional probability is denoted as $q$. The total outage, therefore is calculated as $q + P_4$. In order to evaluate the total outage time, it is reasonable to find an accurate estimate for $q$. This is done via a first order, steady state Markov analysis of both a three state and four state model[15].

## 3.2 THE MARKOV MODEL

The choice of a three state or more model has much to do with what type of information is being transferred and how critical is its BER to obtaining an acknowledgeable transmission. M –curves are measured for a constant BER of interest corresponding to factors such as the type of traffic intended for that hop in study, for example, $10^{-3}$ for voice/video or $10^{-6}$ for data.

The model is introduced by first defining the following states in which a fade having a certain notch depth and notch frequency is situated:

$i(k) \equiv$ Being in region $i$ at an instant $k$, where $i \in \{ 1,2,3,4 \}$

A first order Markov model is used in the derivation hence assuming that knowledge of the previous state provides sufficient memory in describing the fading process. Following statistical simulation later in the chapter it will be shown that a higher order Markov model is not needed in this derivation. Although fading is a continuous process, a discrete time model is used with caution that the sampling frequency is fast enough compared to the time intervals of the fading dynamics. With these assumptions in place, a birth death model can then be proposed wherein a fade cannot jump from one region to another without passing through all intermediate stages. The model is depicted graphically in Figure 3.2. After a sufficiently long

fade duration, the process is said to achieve steady-state so that the following equations become valid.

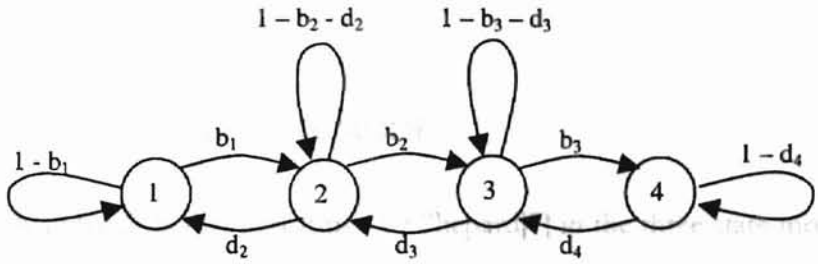



Figure 3.2 Four state Markov Model

$$\frac{dP_1}{dt} = 0 = -P_1 b_1 + P_2 d_2 \tag{3.1}$$

$$\frac{dP_2}{dt} = 0 = -P_1 b_1 - P_2 (b_2 + d_2) + P_3 d_3 \tag{3.2}$$

$$\frac{dP_3}{dt} = 0 = P_2 b_2 - P_3 (b_3 + d_3) + P_4 d_4 \tag{3.3}$$

Shepard[7] decides to reduce his model to a three state model, the reason being that in most important cases, the l- curve is roughly identical to the m- curve in Figure 3.1. Without ruling out the importance of the four state model, he believes it might be of use to situations wherein m- curves are measured with a relatively low BER, $10^{-6}$ for example. Switching over to a three state model, Shepard derives a simple formula for the induced outage $q$ as:

$$q = \frac{P_2 b_2}{(b_2 + d_2)} \tag{3.4}$$

Later in the chapter, fading simulations will be used to predict outage times using this formula compared to actual counts using an exponential probability density distribution(PDF) for fade notch depths.

### 3.2.1 THE FOUR STATE MARKOV MODEL

The analysis in here parallels that performed by Shepard[7] in the three-state model. Referring to Figure 3.2, the state transitions are defined as follows:

$b_i$= birth from state $i$ to state $i+1$ ; $d_i$ = death from state $i+1$ to state $i$

We begin by defining $q$, which seems to be quite complicated in terms of writing:

$$
\begin{aligned}
q = {} & P(3(i) \cap 4(i-1)) + P(3(i) \cap 3(i-1) \cap 4(i-2)) + .... + \\
& P(2(i) \cap 3(i-1) \cap 4(i-2)) + P(2(i) \cap 3(i-1) \cap 3(i-2) \cap 4(i-3)) + ..... + \\
& P(2(i) \cap 2(i-1) \cap 3(i-2) \cap 4(i-3)) + P(2(i) \cap 2(i-1) \cap 3(i-2) \cap 3(i-3) \cap 4(i-4)) + ... + \\
& P(2(i) \cap 2(i-1) \cap 2(i-2) \cap 3(i-3) \cap 4(i-4)) + P(2(i) \cap 2(i-1) \cap 2(i-2) \cap 3(i-3) \cap 3(i-4) \cap 4(i-5)) + ... + \\
& \vdots
\end{aligned}
\tag{3.5}
$$

The analysis in obtaining an estimate of this outage due to reacquisition, $q$, is done by analyzing each and every of the lines in the term for $q$ above. Each line will be referred to as a case in study. I will begin by studying each of the different cases and then sum up all of these cases to reach the estimate. Steady state is assumed for all the analysis to follow.

Case1:

$$q_1 = P(3(i) \cap 4(i-1)) + P(3(i) \cap 3(i-1) \cap 4(i-2)) + .... +$$

$$\textit{and} \qquad P(3(i) \cap 4(i-1)) = P(3(i) \mid 4(i-1))P(4(i-1) = P_4 d_4 \tag{3.6}$$

$$q_2 = P(2(i) \cap 3(i-1) \cap 4(i-2)) + P(2(i) \cap 3(i-1) \cap 3(i-2) \cap 4(i-3)) + \ldots\ldots +$$

*also* $P(3(i) \cap 3(i-1) \cap 4(i-2)) =$

$$P(3(i) \mid 3(i-1) \cap 4(i-2)) P(3(i-1) \cap 4(i-2)) = (1 - b_3 - d_3) P_4 d_4 \quad (3.7)$$

Since Probability of staying in state 3 given after one interval is $1-b_3-d_3$ as seen in Figure 3.2.

The third term can as well be written in terms of the above as:

$$P(3(i) \cap 3(i-1) \cap 3(i-2) \cap 4(i-3)) = [1 - b_3 - d_3]^2 P_4 d_4 \quad (3.8)$$

Likewise the nth term of the sequence is a multiple of the (n-1)th term. Summing all these terms results in the following:

$$q_1 = P_4 d_4 \sum_{n=0}^{\infty} [1 - b_3 - d_3]^n = \frac{P_4 d_4}{(b_3 + d_3)} \quad (3.9)$$

It is worthwhile noting that the above result of case 1 of the four-state model is the exact result of Shepard's simple formula in the three-state model. This however alone does not count for the induced outages in the four-state model, though still contributing greatly to the overall result.

Case2:

The first term in the 2nd line of the expression for $q$ could be written as:

$$P(2(i) \cap 3(i-1) \cap 4(i-2)) = P(2(i) \mid 3(i-1) \cap 4(i-2)) P(3(i-1) \cap 4(i-2)) = P_4 d_4 d_3 \quad (3.10)$$

$$P(2(i)\,|\,3(i-1)\cap 4(i-2)) = d_3$$
$$P(3(i-1)\cap 4(i-2)) = P_4 d_4 \tag{3.11}$$

The second term could as well be written as:

$$P(2(i)\cap 3(i-1)\cap 3(i-2)\cap 4(i-3)) =$$
$$P(2(i)\,|\,3(i-1)\cap 3(i-2)\cap 4(i-3))P(3(i-1)\cap 3(i-2)\cap 4(i-3))$$
$$= d_3 P(3(i-1)\cap 3(i-2)\cap 4(i-3)) = d_3 P(3(i-1)\,|\,3(i-2)\cap 4(i-3))P(3(i-2)\cap 4(i-3))$$
$$= (1-b_3-d_3)P_4 d_4 d_3$$

By inspection, the third term turns out to be equal to $(1\text{-}b_3\text{-}d_3)^2 P_4 d_4 d_3$. Hence the sum of all the terms of case 2 can be put as:

$$q_2 = P_4 d_4 d_3 \sum_{n=0}^{\infty} [1-b_3-d_3]^n = \frac{P_4 d_4 d_3}{(b_3+d_3)} \tag{3.13}$$

Case3:

This line evaluates to a different term from all of the above cases as shown below:

$$q_3 = P(2(i)\cap 2(i-1)\cap 3(i-2)\cap 4(i-3)) + P(2(i)\cap 2(i-1)\cap 3(i-2)\cap 3(i-3)\cap 4(i-4)) + \ldots +$$

$$P(2(i)\cap 2(i-1)\cap 3(i-2)\cap 4(i-3))$$
$$= P(2(i)\,|\,2(i-1)\cap 3(i-2)\cap 4(i-3))P(2(i-1)\cap 3(i-2)\cap 4(i-3)) \tag{3.14}$$
$$= (1-b_2-d_2)P(2(i-1)\,|\,3(i-2)\cap 4(i-3))P(3(i-2)\cap 4(i-3)) = (1-b_2-d_2)d_3 P_4 d_4$$

The second term is written as:

$$
\begin{aligned}
&P(2(i) \cap 2(i-1) \cap 3(i-2) \cap 3(i-3) \cap 4(i-4)) \\
&= P(2(i) \mid 2(i-1) \cap 3(i-2) \cap 3(i-3) \cap 4(i-4)) P(2(i-1) \cap 3(i-2) \cap 3(i-3) \cap 4(i-4)) \\
&= (1-b_2-d_2) P(2(i-1) \mid 3(i-2) \cap 3(i-3) \cap 4(i-4)) P(3(i-2) \cap 3(i-3) \cap 4(i-4)) \\
&= (1-b_2-d_2) d_3 P(3(i-2) \mid 3(i-3) \cap 4(i-4)) P(3(i-3) \cap 4(i-4)) \\
&= (1-b_2-d_2) d_3 (1-b_3-d_3) P_4 d_4
\end{aligned} \tag{3.15}
$$

The third term evaluates to $(1-b_3-d_3)^2 P_4 d_4 d_3 (1-b_2-d_2)$. The sequence is then clearly identified as :

$$
q_3 = P_4 d_4 d_3 (1-b_2-d_2) \sum_{n=0}^{\infty} [1-b_3-d_3]^n = \frac{P_4 d_4 d_3 (1-b_2-d_2)}{(b_3+d_3)} \tag{3.16}
$$

Case4:

The analysis of this line will clarify the solution to a case of infinity as is required in evaluating the $q$ term.

$$
\begin{aligned}
q_4 &= P(2(i) \cap 2(i-1) \cap 2(i-2) \cap 3(i-3) \cap 4(i-4)) + \\
&P(2(i) \cap 2(i-1) \cap 2(i-2) \cap 3(i-3) \cap 3(i-4) \cap 4(i-5)) + \ldots +
\end{aligned} \tag{3.17}
$$

The first term is evaluated as:

$$
\begin{aligned}
&P(2(i) \cap 2(i-1) \cap 2(i-2) \cap 3(i-3) \cap 4(i-4)) \\
&= P(2(i) \mid 2(i-1) \cap 2(i-2) \cap 3(i-3) \cap 4(i-4)) P(2(i-1) \cap 2(i-2) \cap 3(i-3) \cap 4(i-4)) \\
&= (1-b_2-d_2) P(2(i-1) \mid 2(i-2) \cap 3(i-3) \cap 4(i-4)) P(2(i-2) \cap 3(i-3) \cap 4(i-4)) \\
&= (1-b_2-d_2)(1-b_2-d_2) P(2(i-2) \mid 3(i-3) \cap 4(i-4)) P(3(i-3) \cap 4(i-4)) \\
&= (1-b_2-d_2)(1-b_2-d_2) d_3 P_4 d_4
\end{aligned} \tag{3.18}
$$

Similarly the second term turns out to be:

$$
\begin{aligned}
&P(2(i)\cap 2(i-1)\cap 2(i-2)\cap 3(i-3)\cap 3(i-4)\cap 4(i-5))\\
&=P(2(i)|\,2(i-1)\cap 2(i-2)\cap 3(i-3)\cap 3(i-4)\cap 4(i-5))P(2(i-1)\cap 2(i-2)\cap 3(i-3)\cap 3(i-4)\cap 4(i-5))\\
&=(1-b_2-d_2)P(2(i-1)|\,2(i-2)\cap 3(i-3)\cap 3(i-4)\cap 4(i-5))P(2(i-2)\cap 3(i-3)\cap 3(i-4)\cap 4(i-5))\\
&=(1-b_2-d_2)(1-b_2-d_2)P(2(i-2)|\,3(i-3)\cap 3(i-4)\cap 4(i-5))P(3(i-3)\cap 3(i-4)\cap 4(i-5))\\
&=(1-b_2-d_2)(1-b_2-d_2)d_3P(3(i-3)|\,3(i-4)\cap 4(i-5))P(3(i-4)\cap 4(i-5))\\
&=(1-b_2-d_2)(1-b_2-d_2)d_3(1-b_3-d_3)P_4d_4
\end{aligned}
\tag{3.19}
$$

Clearly the trend is obvious again and the sum could be written as:

$$
q_4 = P_4d_4d_3(1-b_2-d_2)^2\sum_{n=0}^{\infty}[1-b_3-d_3]^n = \frac{P_4d_4d_3(1-b_2-d_2)^2}{(b_3+d_3)} \tag{3.20}
$$

At this point, it is interesting to sum up the results brought up so far. The $q$ term could now be written as:

$$
\begin{aligned}
&q = q_1+q_2+q_3+q_4+.....+q_n,\{n\to\infty\}\\
&\Rightarrow q = \frac{P_4d_4}{(b_3+d_3)}+\frac{P_4d_4d_3}{(b_3+d_3)}+\frac{P_4d_4d_3(1-b_2-d_2)}{(b_3+d_3)}+\frac{P_4d_4d_3(1-b_2-d_2)^2}{(b_3+d_3)}+.....
\end{aligned}
\tag{3.21}
$$

Ignoring the first term of equation 3.21 for the moment and glancing at the terms $q_2$ to $q_n$, these terms could be written as the sum of a series, i.e,

$$
q = \frac{P_4d_4}{(b_3+d_3)}+\frac{P_4d_4d_3}{(b_3+d_3)}\sum_{n=0}^{\infty}[1-b_2-d_2]^n = \frac{P_4d_4}{(b_3+d_3)}+\frac{P_4d_4d_3}{(b_3+d_3)(b_2+d_2)} \tag{3.22}
$$

$$
\Rightarrow q = \frac{P_4d_4}{(b_3+d_3)}[1+\frac{1}{(b_2+d_2)}] = \frac{P_4d_4(d_3+b_2+d_2)}{(b_3+d_3)(b_2+d_2)} \tag{3.23}
$$

The total outage, hence is this value of $q$ and $P_4$, the probability of being in region 4. This four state analysis could be thought of as a three state model if regions 2 and 3 are merged into a single region. The analysis would then result in equation 3.4. The key note to realize is that the r and l – curves are the basic blocks of the analysis. Further splitting of the regions in between those two curves should be thought of no more than an attempt of reaching a more accurate result for determining $q$. This is because as the number of regions increase, the model tends to approach a continuous one. Therefore as the model is analyzed for 5,6,7.. regions, the terms or in my study, the *cases* , increase and $q$ is said to approach a more accurate result for the actual duration of induced outages.

## 3.3 FADING SIMULATIONS

With the help of the channel transfer function and the probability density functions described in chapter 2, the model in the previous section is simulated with a computer random generator. The correlation between samples generated by this generator is very small hence making the spectrum as white as possible. To accurately simulate the birth-death model, there has to be some memory between the samples. Thus, if passed through a low pass filter with a specified time constant, T, this problem can somehow be solved. The consequence though is that the PDF with which the samples are obtained is altered. Thanks to the characteristics of a Gaussian distribution which is not altered on filtering.

The fading sequences as described below is according to Shepard's[7]. This simulation will help us prove the integrity of equation 3.22 and provide a tool for adjusting the time constant of the filtering to suit the model.

- A Gaussian process with zero mean and unity variance is generated by sampling a computer random generator *n* times. The autocorrelation of such a sequence with

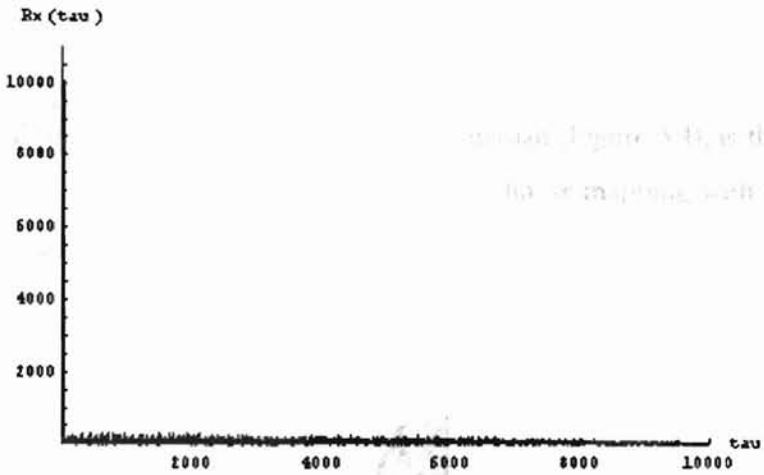


Figure 3.3 Autocorrelation of Gaussian sequence

n=10000 is shown in Figure 3.3

- The sequence is then passed through a low pass filter with T =15, defined below, in continuous form as :

$$\frac{dz(t)}{dt} = \frac{x(t) - z(t)}{RC} \tag{3.24}$$

By utilizing a third order backward difference approximation the following results:

$$\frac{dz(t = i\Delta t)}{dt} \cong \frac{1}{\Delta t}\left[\nabla + \frac{1}{2}\nabla^2 + \frac{1}{3}\nabla^3\right] z_i \tag{3.25}$$

$$\textit{where} \qquad z_i \equiv z(t = i\Delta t); \nabla z_i \equiv z_i - z_{i-1} \tag{3.26}$$

$$z_i = \frac{\left(\frac{x_i}{T} + 3z_{i-1} - \frac{3}{2}z_{i-2} + \frac{1}{3}z_{i-3}\right)}{\frac{11}{6} + \frac{1}{T}} \qquad T \equiv RC/\Delta t \qquad (3.27)$$

- The filtered process, which is an unchanged Gaussian (Figure 3.4), is then standardized to zero mean, unity variance through a memoryless linear mapping with its auto-correlation as seen in Figure 3.5

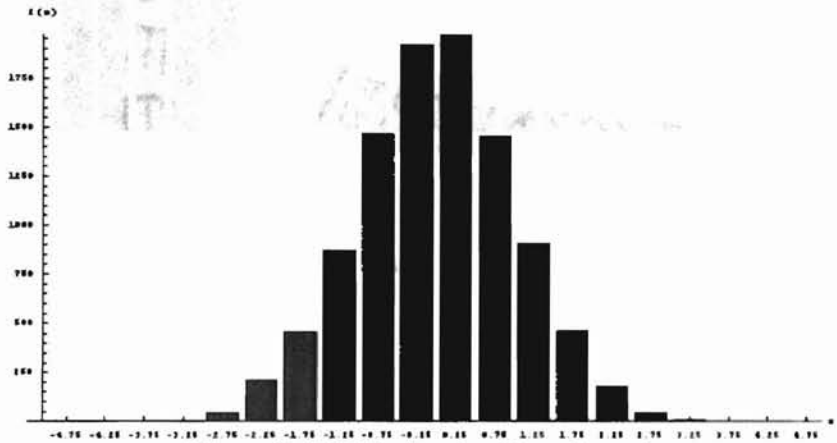


Figure 3.4 PDF of $z$, the filtered $x$ points

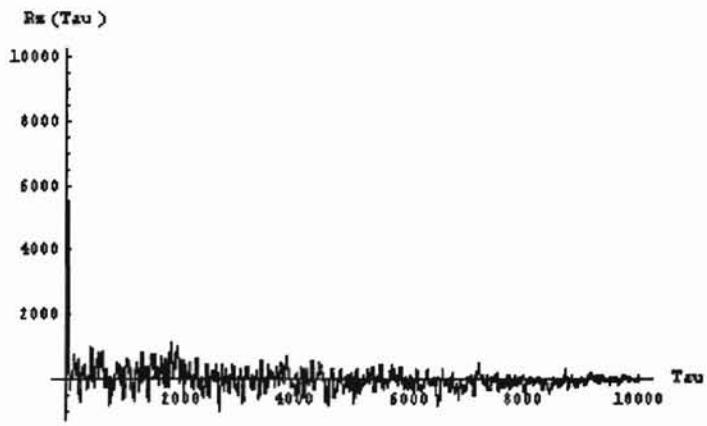



Figure 3.5 Autocorrelation of $z$ sequence

- The standardized sequence, z, is then mapped to the desired PDF in study(Figure 3.6), $w$, which is in this case is an exponential. Its autocorrelation is shown in Figure 3.7

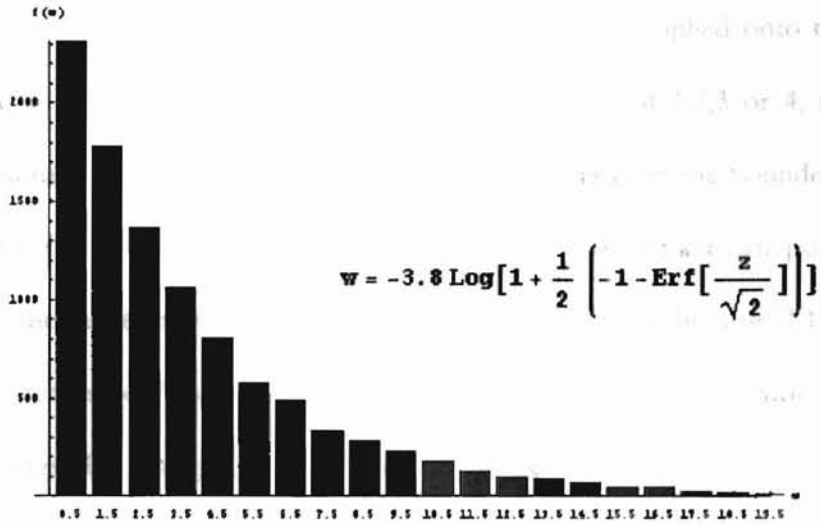



Figure 3.6 PDF of the mapped $z$ data, $w$

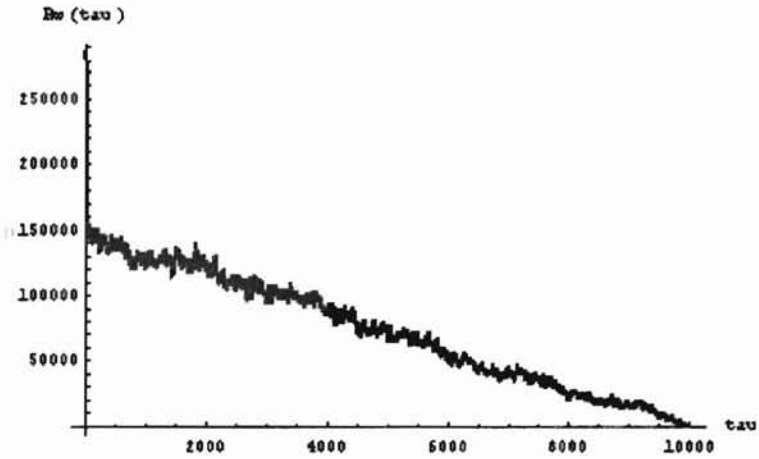



Figure 3.7 Autocorrelation of $w$

The M-curves are said to fall off sharply outside a certain band of frequencies. Also the notch depths do not seem to vary much within those bands. In this way it is possible to approximate the M –curves in Figure 3.1 by the dashed lines above them. Therefore, the signatures could be

defined with their corresponding notch depths bounding the different regions. Since also these signatures cover the same bandwidth, the notch frequency is not a parameter of interest in the simulation, where in fact is the notch depth. By positioning the r and m curves in the B-$f_n$ plane and varying the l-curve (loss of lock), a simple mapping, $y$, is applied onto the sampled and filtered data, $w$ . This function maps the $w$ data into values of 1,2,3 or 4, representing the different regions, according to where the data points belong to in the bounded B-$f_n$ plane. In this simulation r=1 dB, m=3 dB and l varies from 4 to 23 dB. For a $w$ data point equals to 2.4, for example, the function maps it to a value $y$ equals to 2. This is because 2.4 falls in between the r and m curves of Figure 3.1 and hence is assigned the value 2, denoting the region to which it belongs. This mapping is processed on all the $w$ data points and a new set of value points, $y$, result.

### 3.3.1 COUNTING ERROR

This new set, $y$, characterizes the fading dynamics of the system and is the one used in assessing the reacquisition outages as well as the so-called "M-curve" outages. In order to accurately model the system, the $y$ set of points should conform to the birth death model. That is, all transitions between different states can only happen through intermediate states. Therefore, applying to our $y$ set of points, any of the checked transitions in table 3.1 are termed illegal in accurate modeling of the system as a birth death process:

| TO | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1 | ☐ | ☐ | ☒ | ☒ |
| 2 | ☐ | ☐ | ☐ | ☒ |
| 3 | ☒ | ☐ | ☐ | ☐ |
| 4 | ☒ | ☒ | ☐ | ☐ |

Table 3.1 Illegal transitions checked

It is of importance that the $y$ data points consist of as very few illegal transitions as possible so as to allow comparison of simulated counts to that of equation 3.3 for the evaluation of the induced outages. All of the occurring illegal transitions were termed as errors and their number computed for different values, T, the time constant of the low pass filter. It was found that as T increased, the percentage error, scaled over the total number of generated points, also decreased as could be seen in the Figure below for values of T ranging form 1 to 20.

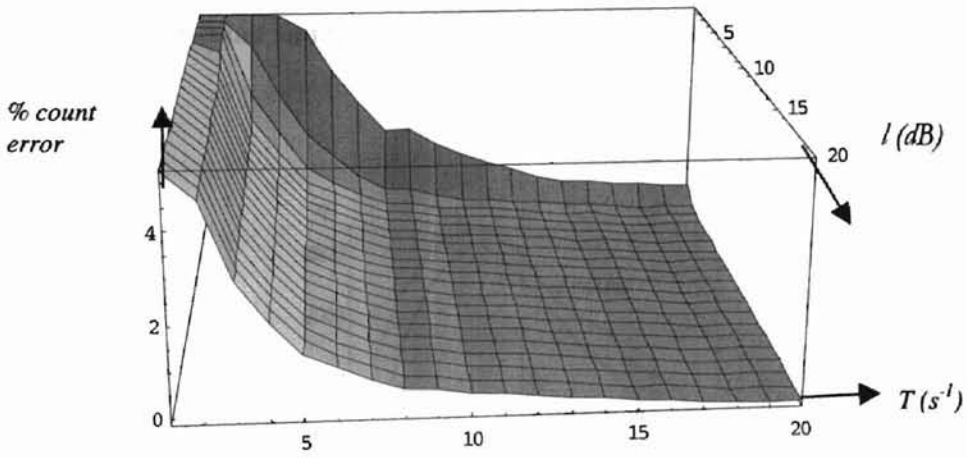



Figure 3.8 Illegal counting error as a function of T

The value of the notch depth does not seem to have an abrupt impact on the error count, as expected, compared to that of T. This is because as the value of T increases, sampling is faster and from equation 3.27, it is intuitively obvious that the filtered $z$ data points become more correlated. This in return shrinks the spectral bandwidth of the generated data points, $x$.

### 3.3.2 RESULTS OF SIMULATION

One of the main objectives out of this analysis was to determine the induced reacquisition outage from the dynamics of the fading channel. With the help of Shepard's three-state model and the presented four-state model in section 3.2.1, an expression, equation 3.23, was presented in determining this outage from a birth-death model. With negligible counting error, direct counting from the dynamics of the computer generated data, should be in well agreement with equation 3.23. With n=10,000 and T=15, Figure 3.9 shows an overlaid plot of both the direct count and the evaluated expression.

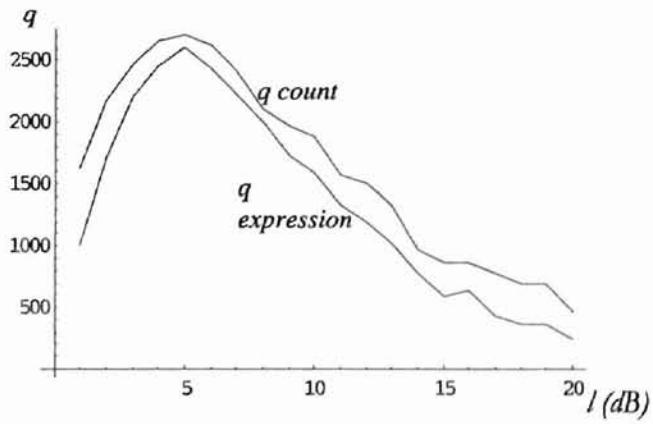



Figure 3.9 Direct count versus expression of $q$

Figure 3.9 shows that equation 3.23 evaluates an accurate approximation of $q$. This result is impressive in the sense that induced outages can be predicted prior to knowledge of a set of equipment signatures for a particular system. A key point worth mentioning is that this four state model could further be developed into an $n$-state model and an expression recomputed. The analysis might be complex though but worth doing if a high level accuracy is essential in determining the induced outage. This brings the model even closer to a continuous

phenomenon which is the actual scenario in real life. The total outage of this four state model is obviously $q + P_4$, where $P_4$ is the total duration of having notch depths in region 4 of Figure 3.1.

*Chapter 4*

# LINEAR NOISE ENHANCEMENT

As already described in the previous chapters, multipath fading has a serious effect on the performance of a digital radio line of sight communication system. Means of determining and predicting outages are crucial in designing such systems. Once these outages have been identified, the next step would be to design countermeasures to these effects. In multipath fading channels, the ideal situation is to have only a direct transmitted ray arriving at the receiver. The channel, characterized by nature, imposes a very difficult challenge to the problem. In fact solutions that would perfectly prevent these delays are quite impossible. A more practical approach is to reduce the effect imposed by the channel through the use of an equalizer. In the coming sections, the transfer function of an ideal equalizer is derived and used in analyzing its effective use in such faded channels.

## 4.1 THE EQUALIZED CHANNEL MODEL

Signal fading causes signal distortion. Carlson[16] defines a distortionless transmission as that wherein the output of a system differs from it's input only by a multiplying constant and a finite time delay. Multipath fading does not impose a finite delay on the output of the channel and hence is definitely a type of distortion. Such systems could be described by the equation:

$$y(t) = Kx(t - t_d) \tag{4.1}$$

where $K$ *and* $t_d$ are constants.

To easily adapt equation 4.1 into the analysis, the output spectrum is calculated as:

$$Y(f) = Ke^{-jwt_d} X(f), \tag{4.2}$$

$$\text{where,} \qquad H(f) = Ke^{-jwt_d} \tag{4.3}$$

Referring to the output spectrum, a distortionless transmission must have constant amplitude response and negative linear phase shift, implying :

$$|H(f)| = K \tag{4.4}$$

$$\arg[H(f)] = -2\pi t_d f \pm m180^{\circ} \tag{4.5}$$

Distortion in general can be classified into two categories: amplitude and phase distortion, as described above, and non-linear distortion. In nonlinear distortion, the transfer function of the system is not defined. Linear distortion refers to both amplitude and delay distortion of a signal, and can to an extent be corrected by the use of equalization. Figure 4.1 shows the block diagram of a system containing both the distorting channel, in this case the faded channel, and an equalizer.

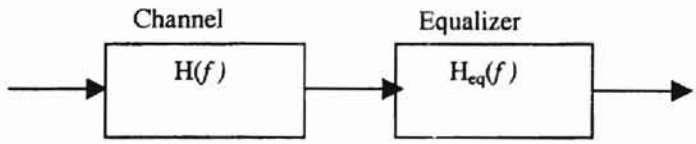



Figure 4.1 Equalized linearly distorted channel

The overall transfer function of the above system is:

$$H(f)H_{eq}(f) \tag{4.6}$$

For distortionless transmission, the overall transfer function in equation 4.6 should be equal to:

$$H(f)H_{eq}(f) = Ke^{-jwt_d} \tag{4.7}$$

and therefore,

$$H_{eq}(f) = \frac{Ke^{-jwt_d}}{H(f)} \tag{4.8}$$

Equation 4.8 is practically difficult to achieve and very seldom can such an equalizer be designed to satisfy it. However, it puts an upper bound on equalization of the channel and can be used in studying the limits to its application.

The transfer function of the channel, as stated in the early chapters is:

$$H(\omega) = a[1 - be^{-j(\omega-\omega_o)\tau}] \tag{4.9}$$

This implies that equation 4.8 could be stated in terms of the transfer function of the channel.

$$H_{eq}(f) = \frac{Ke^{-jwt_d}}{a[1 - be^{-j(\omega-\omega_o)\tau}]} \tag{4.10}$$

$$= \frac{K}{a[1 - be^{-j(\omega-\omega_o)\tau}]\ e^{j\omega t_d}} \tag{4.11}$$

If $\theta$ is substituted for $(\omega-\omega_o)\tau$ in equation 4.11, then the transfer function can be simplified to:

$$H_{eq}(f) = \frac{K}{ae^{j\omega t_d} - abe^{-j(\theta-\omega t_d)}} \tag{4.12}$$

$$= \frac{K}{a[\cos\omega t_d + j\sin\omega t_d] - ab[\cos(\theta - \omega t_d) - j\sin(\theta - \omega t_d)]} \tag{4.13}$$

And regtouping,

$$\Rightarrow H_{eq}(f) = \frac{K}{a[\cos\omega t_d - b\cos(\theta - \omega t_d)] + j[a\sin\omega t_d + ab\sin(\theta - \omega t_d)]} \tag{4.14}$$

In developing the noise model, the power transfer function is of more use than the transfer function. Therefore from the expression for the transfer function of the equalizer:

$$|H_{eq}(f)|^2 = \frac{K^2}{[a\cos\omega t_d - ab\cos(\theta - \omega t_d)]^2 + [a\sin\omega t_d + ab\sin(\theta - \omega t_d)]^2} \tag{4.15}$$

Evaluating and simplifying,

$$|H_{eq}(f)|^2 = \frac{K^2}{[a^2 + a^2b^2 + 2a^2b[\sin\omega t_d \sin(\theta - \omega t_d) - \cos\omega t_d \cos(\theta - \omega t_d)]]} \tag{4.16}$$

$$\text{since} \quad -\cos(A+B) = \sin A \sin B - \cos A \cos B$$

$$\Rightarrow |H_{eq}(f)|^2 = \frac{K^2}{[a^2 + a^2b^2 + 2a^2b[-\cos(\omega t_d + \theta - \omega t_d)]]} \tag{4.17}$$

Simplifying and substituting $(\omega-\omega_o)\tau$ for $\theta$, the final expression for the power transfer function is surprisingly the inverse of the power transfer function of the channel multiplied by a constant $k^2$:

$$|H_{eq}(f)|^2 = \frac{K^2}{[a^2 + a^2b^2 - 2a^2b\cos(\omega\tau - \omega\tau_o)]} \tag{4.18}$$

## 4.2 THE ADDITIVE NOISE CHANNEL MODEL

In any communication system, additive noise is an important factor that limits the performance of such systems. Such noise could be additive from that generated by the electronics and background thermal introduced by wireline and wireless channels. To integrate this effect into the equipment signatures of Figure 3.1, a mathematical model for this additive noise can be established from which one parameter, such as the noise Figure, *F*, can be extracted. The noise-enhanced model is shown below:

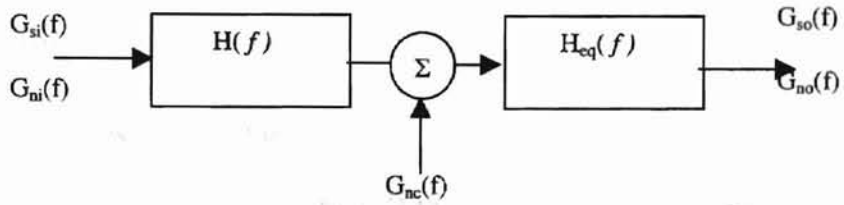



Figure 4.2 Additive noise model

where the spectral densities are defined as follows:

$G_{si}(f)$ = Power density of input signal ; $G_{ni}(f)$ = Power density of input noise ;

$G_{nc}(f)$ = Power density of excess noise introduced by channel ;

$G_{so}(f)$ = Power density of the output signal;

$G_{no}(f)$ = Power density of output noise;

The noise Figure is assessed in terms of the power of the inputs and outputs and is defined as the ratio of the signal to noise ratio of the input to that of the output. That is, the noise Figure, *F*, is :

$$F = \frac{(S/N)_i}{(S/N)_o}$$ ,where subscripts denote input or output (4.19)

The first step is to evaluate the power expressions for the input and output signals and noise. The input signal and noise powers are :

$$S_i = \int_{-W}^{W} G_{si}(f)df \tag{4.20}$$

$$N_i = \int_{-W}^{W} G_{ni}(f)df \tag{4.21}$$

where W = the bandwidth of the system.

In order to reach expressions for the output signal and noise powers, the mathematical model of Figure 4.2 has to be traced to the output. If the output power of the signal and noise after the passage through the channel are referred to as , $S_m$ and $N_m$, the input to the equalizer could be written as:

$$S_m = \int_{-W}^{W} G_{si}(f)|H(f)|^2\, df \tag{4.22}$$

$$N_m = \int_{-W}^{W} [G_{ni}(f)|H(f)|^2 + G_{nc}(f)]df \tag{4.23}$$

Thus, it is clear from Figure 4.2 that the power spectral densities of the intermediate stage can be written as:

$$G_{sm}(f) = G_{si}(f)|H(f)|^2 \tag{4.24}$$

$$G_{nm}(f) = G_{ni}(f)|H(f)|^2 + G_{nc}(f) \tag{4.25}$$

Therefore,

$$S_o = \int_{-W}^{W} [G_{sm}(f)|H_{eq}(f)|^2]df \tag{4.26}$$

$$N_o = \int_{-W}^{W} [G_{nm}(f)|H_{eq}(f)|^2]df \tag{4.27}$$

$$\Rightarrow S_o = \int_{-W}^{W} [G_{si}(f)|H_{eq}(f)|^2|H(f)|^2]df \tag{4.28}$$

$$\Rightarrow N_o = \int_{-W}^{W} [G_{ni}(f)|H_{eq}(f)|^2|H(f)|^2 + G_{nc}(f)|H_{eq}(f)|^2]df \tag{4.29}$$

The noise Figure could be evaluated as follows:

$$F = \frac{\int_{-W}^{W} G_{si}(f)df}{\int_{-W}^{W} G_{ni}(f)df} \cdot \frac{\int_{-W}^{W} [G_{ni}(f)|H_{eq}(f)|^2|H(f)|^2 + G_{nc}(f)|H_{eq}(f)|^2]df}{\int_{-W}^{W} [G_{si}(f)|H_{eq}(f)|^2|H(f)|^2]df} \tag{4.30}$$

$$\text{since} \quad |H_{eq}(f)|^2 = \frac{K^2}{[a^2 + a^2b^2 - 2a^2b\cos(\omega\tau - \omega\tau_o)]} \tag{4.31}$$

$$\text{and} \quad |H(f)|^2 = [a^2 + a^2b^2 - 2a^2b\cos(\omega\tau - \omega\tau_o)] \tag{4.32}$$

$$\Rightarrow |H(f)|^2|H_{eq}(f)|^2 = K^2 \tag{4.33}$$

Therefore the noise Figure can further be simplified into,

$$F = \frac{\int_{-W}^{W} G_{si}(f)df}{\int_{-W}^{W} G_{ni}(f)df} \cdot \frac{\int_{-W}^{W} [G_{ni}(f)K^2 + G_{nc}(f)|H_{eq}(f)|^2]df}{K^2 \int_{-W}^{W} G_{si}(f)df} \tag{4.34}$$

$$\Rightarrow F = \frac{\int_{-W}^{W}[G_{ni}(f)K^2 + G_{nc}(f)|H_{eq}(f)|^2]df}{K^2\int_{-W}^{W}G_{ni}(f)df} \tag{4.35}$$

$$\Rightarrow F = 1 + \frac{\int_{-W}^{W}[G_{nc}(f)|H_{eq}(f)|^2]df}{K^2\int_{-W}^{W}G_{ni}(f)df} \tag{4.36}$$

The thermal noise spectral density from quantum mechanical analysis could be stated as:

$$S_n(f) = \frac{\hbar f}{2(e^{\frac{\hbar f}{kT}} - 1)} \tag{4.37}$$

*where* $\hbar$ = *planck's cons* tan *t*; *k* = *Boltzmann's cons* tan *t*
*T* = *temperature in* deg *ress kelvin*

The spectrum achieves its maximum at $f$=0 and its value is $kT/2$. As $f$ approaches infinity, the spectrum goes to zero at a rate that is very slow. For example, at 300 °K, the spectrum drops off to 90% of its maximum value at about $2*10^{12}$Hz which is beyond the microwave frequencies of interest here. Therefore thermal noise could be modeled as white with its power spectral density equal to $kT/2$. Therefore, the input thermal noise could be modeled at $T_1$ and the channel thermal noise at $T_2$. The noise Figure can then be interpreted as:

$$F = 1 + \frac{\int_{-W}^{W}[kT_1/2|H_{eq}(f)|^2]df}{K^2\int_{-W}^{W}kT_2/2df} \tag{4.38}$$

If K = 1,

$$\Rightarrow F = 1 + \frac{T_1}{2T_2 W} \int_{-W}^{W} \frac{1}{[a^2 + a^2 b^2 - 2a^2 b \cos(\omega\tau - \omega_o \tau)]} df \tag{4.39}$$

$$and \quad F - 1 = \frac{T_1}{T_2} \cdot \frac{1}{a^2} \cdot \frac{1}{2W} \int_{-W}^{W} \frac{1}{[1 + b^2 - 2b \cos(\omega\tau - \omega_o \tau)]} df \tag{4.40}$$

Taking logarithms to the base 10 on both sides and simplifying for $F$ in decibels:

$$\Rightarrow (F)_{dB} = 10Log[10^{Log(\frac{T_1}{T_2}) + Log(\frac{1}{a^2}) + Log(\frac{1}{2W}) + Log(\int_{-W}^{W} \frac{1}{[1+b^2-2b\cos(\omega\tau-\omega_o\tau)]} df)}] \tag{4.41}$$

Also F could be written as:

$$(F)_{dB} = 10Log[10^{T + A_e + W_s + B_e}] \tag{4.42}$$

$$\begin{aligned} where \quad T &= Log(\frac{T_1}{T_2}) \\ W_s &= Log(\frac{1}{2W}) \\ A_e &= Log(\frac{1}{a^2}) \\ B_e &= Log(\int_{-W}^{W} \frac{1}{[1 + b^2 - 2b\cos(\omega\tau - \omega_o\tau)]} df) \end{aligned} \tag{4.43}$$

We see the noise Figure is a function of four terms clearly identified in the raised power of equation 4.42. As one would expect, both flat and selective fading contribute to the noise Figure of the system. The $1/a^2$ term is due to flat fading and if set equals to 1, the noise Figure is left dependant on $B_e$ shown in equation 4.43. This term is said to enhance noise in the system and is referred to as the linear noise enhancement from the equalizer. Although faded channels can to a degree be well equalized, the process also drags along a considerable amount of noise which degrades the output signal to noise ratio. Thus, although equalization reduces inter-symbol interference effects, it also enhances the noise. Figure 4.4 shows a plot of the

noise enhancement terms, $B_e + A_e$, versus A and B, the flat and selective fading levels of the signal in decibels. In any communication system therefore, a trade-off exists between the use of equalization at the expense of degrading signal to noise ratio at the output. In the next section, the noise Figure is used in estimating limits as to how deep of a dispersive notch can be equalized without enhancing the thermal noise to an unacceptable amount.

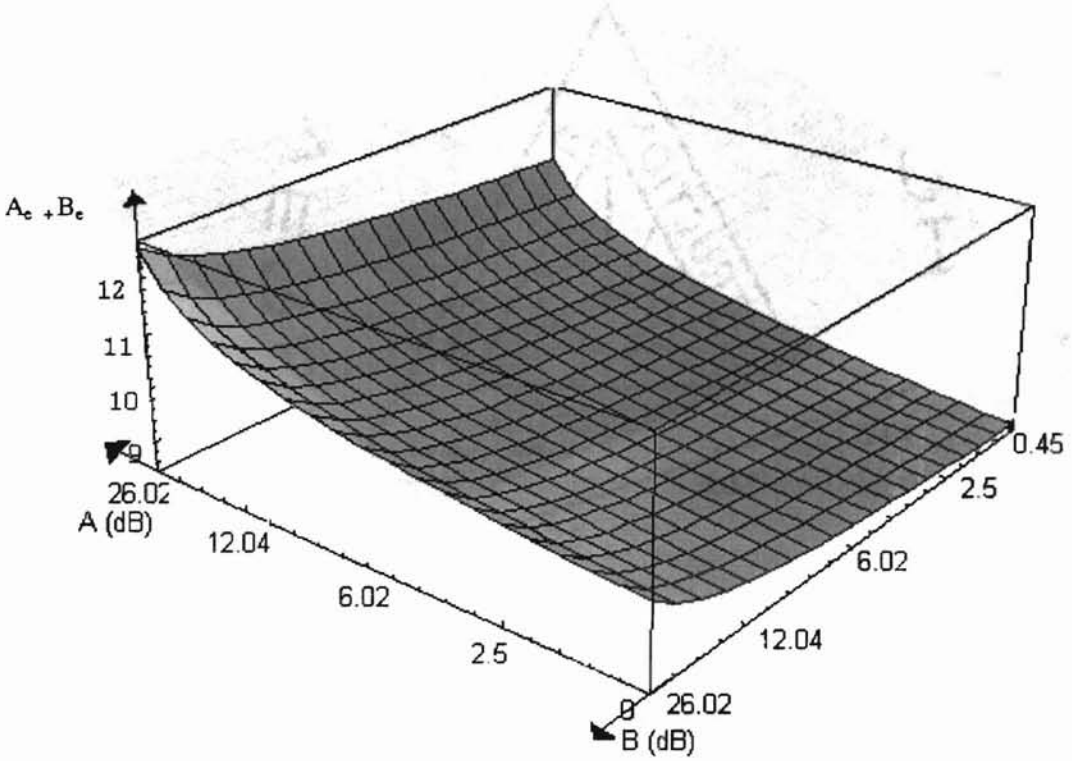



Figure 4.3 Noise enhancement terms versus A and B

## 4.2 ESTIMATING LIMITS ON EQUALIZATION

The aim is to have a system BER greater than or equal to the critical BER during the fading of the channel. Depending on the type of communication system used and the method of modulation employed, a minimum S/N ratio at the output is usually associated with a specific BER. The noise Figure obtained in equation 4.41 can thus be used in estimating the notch

depth limits that can be tolerated without noise excessive noise enhancement given a prior knowledge of the temperatures of the input noise signal and channel. In this calculation

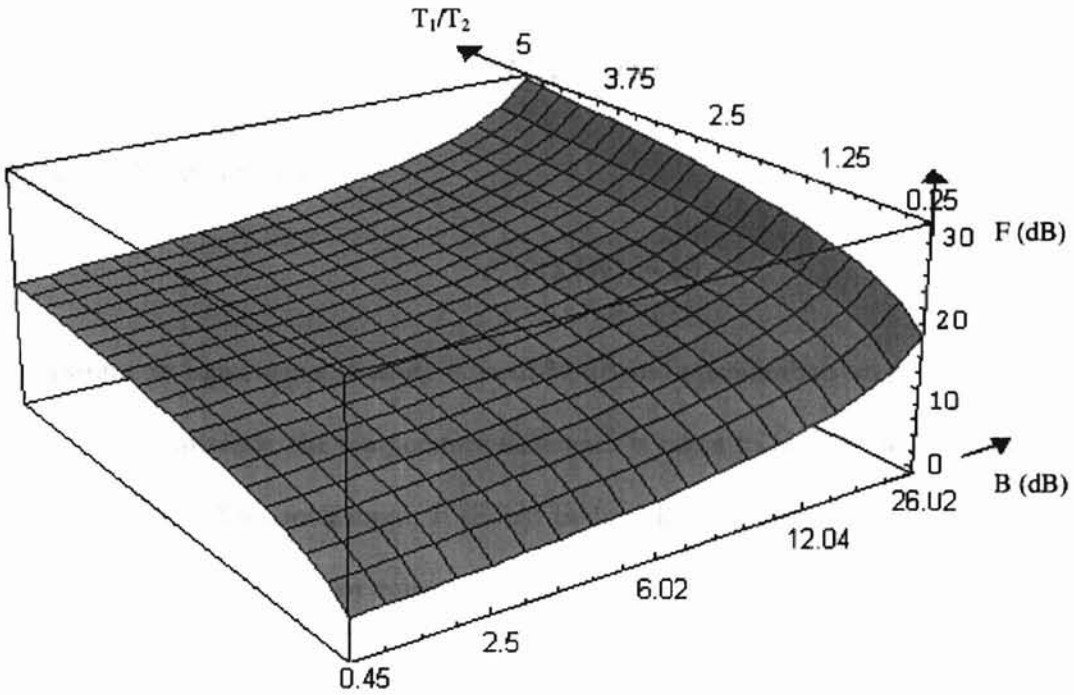



Figure 4.5 3D plot of F versus $T_1/T_2$ and B

the bandwidth of the system is assumed to be 25.3Mhz and fade notch location equals to – 19.8Mhz. The time delay constant is taken as 6.3 ns, as discussed in chapter 2. These values were based on Rummlers[6] experiment but any otherwise realistic values can as well be chosen. Figure 4.5 shows a 3D plot of $F$ versus $T_1/T_2$ and $B$. This plot generally shows the behavioral trend of the system as the temperature difference and notch depths vary. Figure 4.6 depicts a contour plot of Figure 4.5 and is a more useful tool in extracting the limits of the notch depth. It shows an exponential growth of the noise Figure as the notch depth value is increased over a certain temperature difference. An expected behavior since as the notch depth

increases, one would expect to pump in more power at the input to overcome this dispersive fading effect. To illustrate the use of Figure 4.5, let us assume that an output S/N of 20dB or higher, guarantees that the system BER is equal to or greater than $10^{-3}$. Since the noise Figure, $F$, is stated in decibels as:

$$F = (S/N)_i - (S/N)_o \tag{4.12}$$

and $(S/N)_o \geq 20$ dB, it implies :

$$F \leq (S/N)_i - 20 \tag{4.13}$$

We will assume an input signal to noise ratio of 30 dB and operate the system at two different instances. At one instance, the temperature difference is taken to be 0.25 and at another it is 1.5 as in Figure 4.6. With such input, $F$ should be less than 10 dB in order for the system's performance to be higher than or equal to a BER of $10^{-3}$. When $T_1/T_2 = 0.25$, the value of B corresponding to $F = 10$ dB is approximately 13.05 dB. This implies that for notch depths greater than 13.05 dB the system will lose its desired performance and the equalizer would be doing more harms in enhancing the noise than equalizing the channel. For $T_1/T_2 = 1.5$, a limit on B does not exist on the plot. This simply means that the equalizer is useless over any range of notch depths and the solution is in pumping more power at the input. It is obvious from Figure 4.6 that by increasing the signal to noise ratio at the input, the range of operation of the equalizer increases at a certain temperature difference. However, this solution has safety and regulatory restrictions imposed on it[17]. In a very noisy environment where the temperature difference is high, and a tight bound is set on the input power, it is realizable that an equalizer is not that much useful for the recovery of a faded signal.

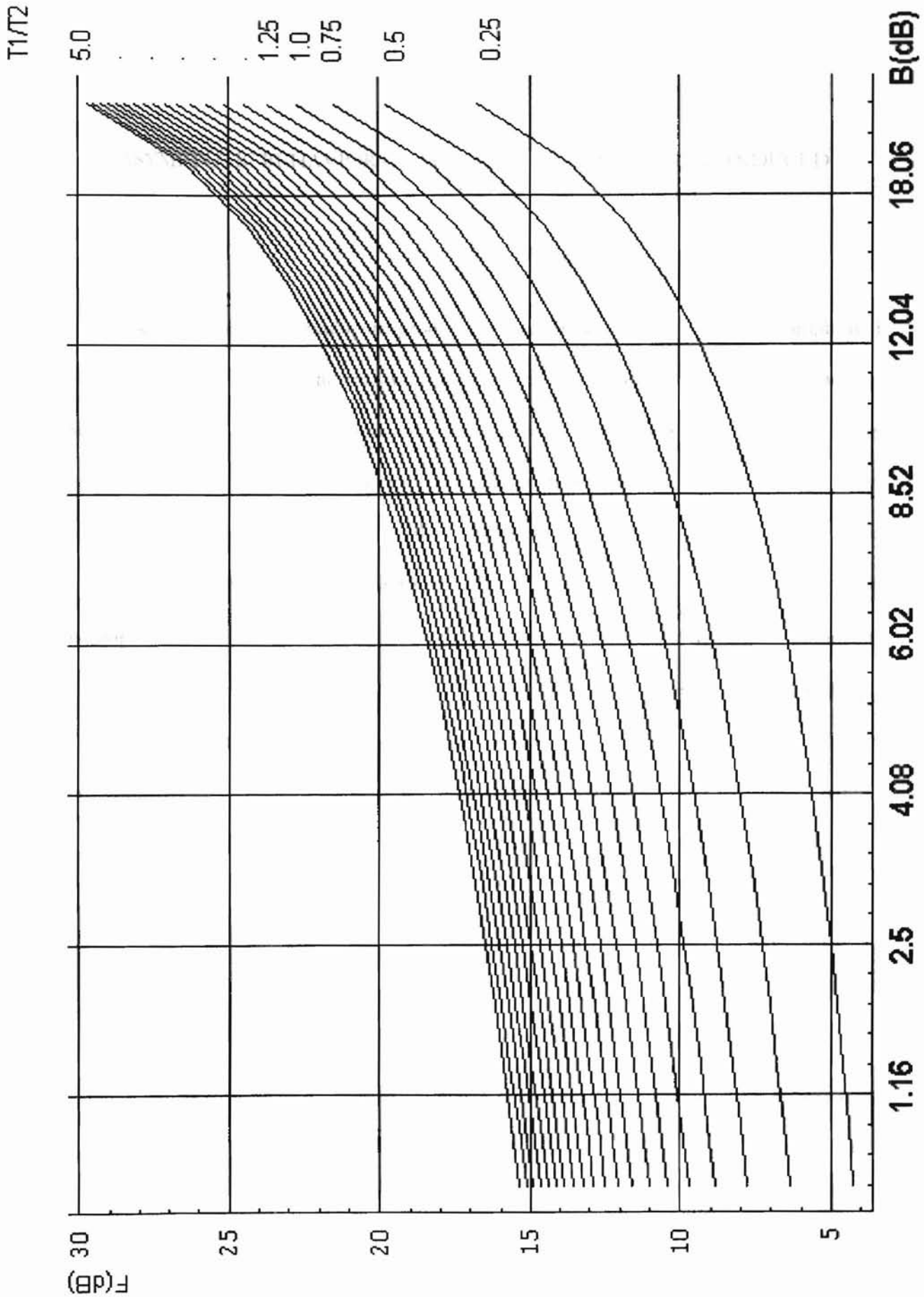



Fig 4.6 Contour plot of F versus $T_1/T_2$ and B

*Chapter 5*

# ASYMPTOTIC BEHAVIOR OF REACQUISITION HYSTERISIS INDUCED OUTAGES

In assessing the impact of multipath propagation on various radio systems the complicated effects of fading dynamics and reacquisition hysterisis are often ignored. As Greenstein and Shafi[14] put it: "A multipath-related component of outage time is the time it takes a receiver to recover synchronization after losing it because of high bit error ratios. This 'hysterisis' effect is ignored in most existing methods of outage time estimation, the implicit assumption being that such periods are relatively short." Shepard has shown however, that this implicit assumption is often not valid — indeed, hysterisis induced outages can actually be dominant, even for rapidly varying fades and for fading events of short duration [7]. Ranade [8] obtained an empirical asymptote for total outage time (including hysterisis) in the limit of vanishing hysterisis. In this chapter we shall compare Ranade's asymptote and a simple formula (from Shepard's three state model) to the outages obtained from simulated fades in this same limit.

## 5.1 COUNTING AND ASYMPTOTE

Using the same method of simulation as described in chapter 3, counted results for the total outage in the limit of vanishing hysterisis is used in the comparison. Two distributions are used in simulating the fading data. While many authors agree that the notch depth has an exponential distribution, at least in digital radio line of sight links[6], many others argue that

the distribution is Rayleigh. Notch depths in mobile channels have been characterized with a Rayleigh distribution[4]. Both distributions will be used here in the comparison. The probability density function of a Rayleigh distribution used in this context is:

$$f(x) = \frac{x}{b^2} e^{(\frac{-x^2}{2b^2})} \tag{5.1}$$

where b is set to 1. Since the data is originally sampled by a Gaussian distribution, the following mapping results in the Rayleigh distribution:

$$w = \sqrt{-2b^2 Log[\frac{1}{2}[1 - Erf[\frac{z}{\sqrt{2}}]]} \tag{5.2}$$

where $z$ is the filtered Gaussian data and Erf is the error function.

Ranade has shown that in the limit that regions 2 and 3 in Figure 3.1 become very small then the formula $\sqrt{(P_2 + P_3 + P_4)P_4}$ is proportional to the total outage time. Obviously in the limit of vanishing hysterisis, the formula evaluates to $\sqrt{(P_4)P_4} = P_4$. The empirical formula could also be stated in terms of the three state model as $\sqrt{(P_2 + P_3)P_3}$ . It could be thought of as regions 2 and 3 converging into one region, say 2. Clearly, region 4 is then named as region 3. An exponential distribution was used for 10,000 simulated data points sampled at T =10 in Figure 5.1. The Figure below shows how this limit is approached with the r-curve approaching the l-curve with no m-curve in Figure 3.1. Figure 5.2 draws the same comparison under the same condition but instead the data is sampled with a Rayleigh distribution. It could be noticed from both studies how the Markov model prediction of outage times closely approximates the counted results for the entire range of increasing $r$. Hence accurate outage estimates could be obtained for random values of $P_2$. Unlike Ranade's

formula, which from both plots is most close towards increasing values of $r$, Shepard's[7] formula approximates well even in the limit of vanishing hysterisis.

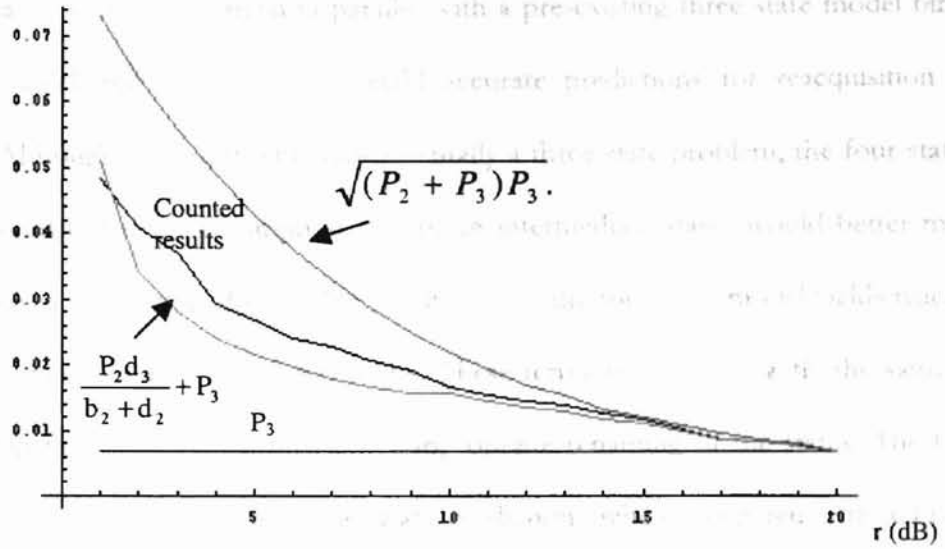



Figure 5.1 Comparison based on Exponential Distribution

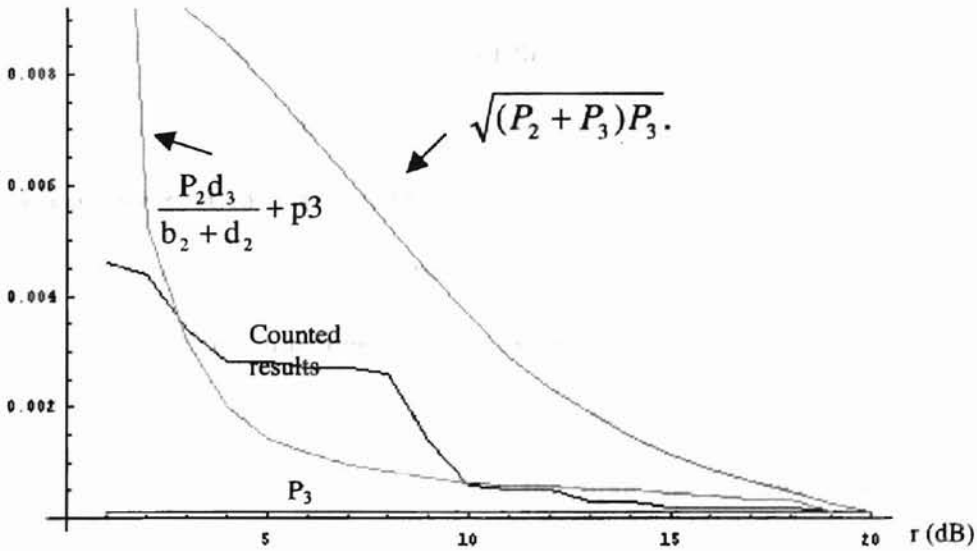



Figure 5.2 Comparison based on Rayleigh Distribution

# CONCLUSION

A four state model was derived in parallel with a pre-existing three state model birth-death Markov model and also shown to yield accurate predictions for reacquisition induced outages. Although reacquisition is fundamentally a three-state problem, the four-state model was explored under the assumption that more intermediate states would better model the continuous fade variations for smaller T. Analysis of the four-state model yields reacquisition induced outages as the sum of two terms. These terms reduce to exactly the same term as obtained in the three-state model under appropriate renaming of the states. The four-state model was found to also provide accurate predictions when compared with actual counts extracted from the simulations.

Most authors have omitted reacquisition induced outage times in their predictions, because of the previous complexity of modeling this effect. Ranade, to the best of my knowledge, was the only author to attempt the calculation of those predictions by obtaining an empirical asymptotic formula (valid in the limit of vanishing hysterisis). The three-state model predictions were however shown to yield more accurate results than that of the asymptotic formula even in the limit of vanishing hysterisis.

Further, it was shown that a fundamental limit exists for fade depths to which an equalizer aligned in a channel distorted by multipath fading can remove the dispersion without enhancing the thermal noise introduced by both the input and channel to an unacceptable amount. These limits were extracted by evaluating an expression for the noise figure, F, of the system in terms of the linear noise enhancement by the equalizer.

## FUTURE POSSIBLE RESEARCH

The utility of computer generated fade simulations of fading is greatly enhanced by running these sequences through hardware built simulators. This piece of hardware characterizes the channel transfer function. In Appendix A, a description of the simulator and software/hardware D/A conversion unit is provided. In the future, the simulator/PC interface could be used to measure equipment signatures, described in chapter II. It could also be used to simulate probabilistic fading sequences from which the Markov parameters could be extracted to test hysterisis predictions.

# REFERENCES


[1] Rizzi, Peter A., Microwave Engineering, Englewood Cliffs, New Jersey, Prentice Hall, 1988.

[2] Govind P. Agrawal, Fiber optic communication systems/2nd edition, New York, New York, John Wiley & Sons, 1997.

[3] John G. Proakis and Masoud Salehi, Communication Systems Engineering, Saddle River, New Jersey, 1994.

[4] Ellen Kayata Wessel, Wireless Multimedia Communications, Reading, Massachusetts, 1997.

[5] A.A.R. Townsend, Digital Line-of-sight Radio Links, London, Prentice Hall, 1988

[6] C.W. Lundgren and W.D. Rummler, " Digital radio outage due to selective fading — observation vs. prediction from laboratory simulation," Bell Syst. Tech. J., vol. 58, no.5, pp. 1073-1100, May-June 1979.

[7] S. R. Shepard, "Reacquisition Induced Outages During Dispersive Fading Events," submitted to IEEE Transactions on Communications (1998).

[8] A. Ranade, " Statistics of the time dynamics of dispersive multipath fading and its effects on digital microwave radios," in Proc. IEEE 1985 Int. Conf. Commun., Cicago, IL, Paper 47.7.

[9] J.A. Hoffmeyer, " Measurement, modeling, and simulation of digital LOS microwave channels with applications to outage prediction," IEEE Trans, Commun., vol. 39, pp. 1295-1305, Sept. 1991.

[10] Di Zenobio, D., " A closed form to predict outage of digital radio", IEEE Transactions on Antennas and Propagation, vol 40, no.1, pp.112-115, Jan 1992.

[11] Kafedziski, V., " QAM Digital Radio Outage Probability in the presence of multipath fading,"Computers and communications, 1992. Conference Proceedings., Eleventh Annual International Phoenix Conference on, pp. 248-254

[12] Rummler, W. D., " A Multipath Channel Model for Line-of-Sight Digital Radio Systems", International Communications Conference, 1978, 3, paper 47.5.

[13] G.L. Fenderson, S.R. Shepard, and M.A. Skinner, " Adaptive transversal equalizer for 90-Mb/s 16-QAM systems in the presence of multipath propagation," in Proc. IEEE 1983 Int. Conf. Commun., Boston, MA, paper C8.7.

[14] L.J. Greenstein and M. Shafi, "Outage calculation methods for microwave digital radio," IEEE Commun. Mag., vol. 25, pp 30-39, Feb. 1987.

[15] A.W. Drake, Fundamentals of Applied Probability theory, New York, New York, McGraw-Hill, 1967.

[16] A.B. Carlson, Communication Systems-An Introduction to Signals and Noise in Electrical Communication, New York, New York, McGraw-Hill, 1968.

[17] FCC Report and Order in ET Docket 93-62, 61 Federal Register 41006 (August 7, 1996), [47 C.F.R. 1.1307(b), 1.1310, 2.1091, 2.1093].

# APPENDIX A

HARDWARE/SOFTWARE CONTROL CIRCUITRY OF IF FADE SIMULATOR

The hardware and software associated with an IF fade simulator. The schematic of the D/A control circuitry from the parallel port of a PC is shown in figure A.1. The control software was written in Visual C++ and it's source code is listed below.

```
// TestDlg.cpp : implementation file
//

#include "stdafx.h"
#include "Test.h"
#include "TestDlg.h"
#include "afxwin.h"
#include "conio.h"
#include "math.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////////
/////
// CAboutDlg dialog used for App About


class CAboutDlg : public CDialog
{
public:
        CAboutDlg();

// Dialog Data
        //{{AFX_DATA(CAboutDlg)
        enum { IDD = IDD_ABOUTBOX };
        //}}AFX_DATA

        // ClassWizard generated virtual function overrides
        //{{AFX_VIRTUAL(CAboutDlg)
        protected:
        virtual void DoDataExchange(CDataExchange* pDX);    // DDX/DDV support
        //}}AFX_VIRTUAL

// Implementation
```

```
protected:
	//{{AFX_MSG(CAboutDlg)
	//}}AFX_MSG
	DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{
	//{{AFX_DATA_INIT(CAboutDlg)
	//}}AFX_DATA_INIT
}

void CAboutDlg::DoDataExchange(CDataExchange* pDX)
{
	CDialog::DoDataExchange(pDX);
	//{{AFX_DATA_MAP(CAboutDlg)
	//}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CAboutDlg, CDialog)
	//{{AFX_MSG_MAP(CAboutDlg)
		// No message handlers
	//}}AFX_MSG_MAP
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
/////
// CTestDlg dialog

CTestDlg::CTestDlg(CWnd* pParent /*=NULL*/)
	: CDialog(CTestDlg::IDD, pParent)
{
	//{{AFX_DATA_INIT(CTestDlg)
	m_Option = -1;
	m_Edit7 = 0.0;
	m_Edit8 = 0.0;
	m_Edit9 = 0.0;
	m_Edit4 = 0.0;
	m_Edit5 = 0.0;
	m_Edit6 = 0.0;
	m_Edit1 = 0.0;
	m_Edit2 = 0.0;
	m_Edit3 = 0.0;
	//}}AFX_DATA_INIT
	// Note that LoadIcon does not require a subsequent DestroyIcon in Win32
	m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}
```

```
void CTestDlg::DoDataExchange(CDataExchange* pDX)
{
	CDialog::DoDataExchange(pDX);
	//{{AFX_DATA_MAP(CTestDlg)
	DDX_Control(pDX, IDC_BUTTON7, m_Da3);
	DDX_Control(pDX, IDC_BUTTON6, m_Upa3);
	DDX_Control(pDX, IDC_BUTTON5, m_Da2);
	DDX_Control(pDX, IDC_BUTTON4, m_Upa2);
	DDX_Control(pDX, IDC_BUTTON3, m_Da1);
	DDX_Control(pDX, IDC_BUTTON2, m_Upa1);
	DDX_Radio(pDX, IDC_RADIO1, m_Option);
	DDX_Text(pDX, IDC_EDIT7, m_Edit7);
	DDX_Text(pDX, IDC_EDIT8, m_Edit8);
	DDX_Text(pDX, IDC_EDIT9, m_Edit9);
	DDX_Text(pDX, IDC_EDIT4, m_Edit4);
	DDX_Text(pDX, IDC_EDIT5, m_Edit5);
	DDX_Text(pDX, IDC_EDIT6, m_Edit6);
	DDX_Text(pDX, IDC_EDIT1, m_Edit1);
	DDX_Text(pDX, IDC_EDIT2, m_Edit2);
	DDX_Text(pDX, IDC_EDIT3, m_Edit3);
	//}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CTestDlg, CDialog)
	//{{AFX_MSG_MAP(CTestDlg)
	ON_WM_SYSCOMMAND()
	ON_WM_PAINT()
	ON_WM_QUERYDRAGICON()
	ON_BN_CLICKED(IDC_BUTTON2, OnButton2)
	ON_BN_CLICKED(IDC_BUTTON4, OnButton4)
	ON_BN_CLICKED(IDC_RADIO1, OnRadio1)
	ON_BN_CLICKED(IDC_RADIO2, OnRadio2)
	ON_BN_CLICKED(IDC_RADIO3, OnRadio3)
	ON_BN_CLICKED(IDC_BUTTON3, OnButton3)
	ON_BN_CLICKED(IDC_BUTTON5, OnButton5)
	ON_BN_CLICKED(IDC_BUTTON6, OnButton6)
	ON_BN_CLICKED(IDC_BUTTON7, OnButton7)
	ON_BN_CLICKED(IDC_BUTTON1, OnButton1)
	//}}AFX_MSG_MAP
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
/////
// CTestDlg message handlers

BOOL CTestDlg::OnInitDialog()
{
	CDialog::OnInitDialog();
```

```
	// Add "About..." menu item to system menu.

	// IDM_ABOUTBOX must be in the system command range.
	ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
	ASSERT(IDM_ABOUTBOX < 0xF000);
	CMenu* pSysMenu = GetSystemMenu(FALSE);
	CString strAboutMenu;
	strAboutMenu.LoadString(IDS_ABOUTBOX);
	if (!strAboutMenu.IsEmpty())
	{
		pSysMenu->AppendMenu(MF_SEPARATOR);
		pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX,
strAboutMenu);
	}

	// Set the icon for this dialog.  The framework does this automatically
	//  when the application's main window is not a dialog
	SetIcon(m_hIcon, TRUE);			// Set big icon
	SetIcon(m_hIcon, FALSE);		// Set small icon

	// TODO: Add extra initialization here
	m_Edit4=0.039;
	m_Edit5=0.078;
	m_Edit6=0.078;
	m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
	m_Edit8=(m_Edit6/m_Edit7);
	m_Edit9=atan2(m_Edit5,m_Edit4);
	m_Upa1.EnableWindow(FALSE);
	m_Da1.EnableWindow(FALSE);
	m_Upa2.EnableWindow(FALSE);
	m_Da2.EnableWindow(FALSE);
	m_Upa3.EnableWindow(FALSE);
	m_Da3.EnableWindow(FALSE);
	UpdateData(FALSE);
	return TRUE;  // return TRUE  unless you set the focus to a control
}

void CTestDlg::OnSysCommand(UINT nID, LPARAM lParam)
{
	if ((nID & 0xFFF0) == IDM_ABOUTBOX)
	{
		CAboutDlg dlgAbout;
		dlgAbout.DoModal();
	}
	else
	{
		CDialog::OnSysCommand(nID, lParam);
```

```
}
}
// If you add a minimize button to your dialog, you will need the code below
//  to draw the icon.  For MFC applications using the document/view model,
//  this is automatically done for you by the framework.

void CTestDlg::OnPaint()
{
        if (IsIconic())
        {
                CPaintDC dc(this); // device context for painting

                SendMessage(WM_ICONERASEBKGND, (WPARAM) dc.GetSafeHdc(),
0);

                // Center icon in client rectangle
                int cxIcon = GetSystemMetrics(SM_CXICON);
                int cyIcon = GetSystemMetrics(SM_CYICON);
                CRect rect;
                GetClientRect(&rect);
                int x = (rect.Width() - cxIcon + 1) / 2;
                int y = (rect.Height() - cyIcon + 1) / 2;

                // Draw the icon
                dc.DrawIcon(x, y, m_hIcon);
        }
        else
        {
                CDialog::OnPaint();
        }
}

// The system calls this to obtain the cursor to display while the user drags
//  the minimized window.
HCURSOR CTestDlg::OnQueryDragIcon()
{
        return (HCURSOR) m_hIcon;
}




void CTestDlg::OnButton2()
{
double temp2,temp21, root1;
int i;
double send[3];
```

```
double d1,d2,d3,ad1,ad2,ad3;
temp2 = m_Edit7;
temp21= m_Edit4;


if( m_Option == 0 )
 {
        m_Edit4 = m_Edit4 + 0.039;
        m_Edit5= (m_Edit4 * tan(m_Edit9));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp2);
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9=atan2(m_Edit5,m_Edit4);
 }
if (m_Option == 2)
{
        m_Edit4 = m_Edit4 + 0.039;
        root1= (temp21*temp21)+(m_Edit5*m_Edit5)-(m_Edit4*m_Edit4);
        m_Edit5 = sqrt(root1);
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp2);
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9=atan2(m_Edit5,m_Edit4);
}

if (m_Edit4 > 9.96 || m_Edit5 <=0 || m_Edit5 > 9.96 || m_Edit6 <=0 || m_Edit6
>9.96 || root1 <0)
UpdateData (TRUE);
else
{


        d1 = (m_Edit4*256)/10;
        d2 = (m_Edit5*256)/10;
        d3 = (m_Edit6*256)/10;

        ad1 = (floor(d1) + ceil(d1))/2 - d1;

        if (ad1 > 0)
        {send[0] = floor(d1);}
        else
        {send[0] = ceil(d1);}

        ad2 = (floor(d2) + ceil(d2))/2 - d2;
        if (ad2 > 0)
        {send[1] = floor(d2);}
        else
        {send[1] = ceil(d2);}
```

```
        ad3 = (floor(d3) + ceil(d3))/2 - d3;
        if (ad3 > 0)
        {send[2] = floor(d3);}
        else
        {send[2] = ceil(d3);}

m_Edit1 = send[0];
m_Edit2 = send[1];
m_Edit3 = send[2];

for (i=0;i<3;i++)
{
        _outp(888,send[i]);
        Sleep(10);
        _outp(890,i);
        Sleep(10);
        _outp(890,3);
}
UpdateData(FALSE);


}
}

void CTestDlg::OnButton4()
{
int i;
double send[3];
double d1,d2,d3,ad1,ad2,ad3;
double temp4,temp41,root;
temp4 = m_Edit7;
temp41=m_Edit5;
if( m_Option == 0 )
  {
        m_Edit5 = m_Edit5 + 0.039;
   m_Edit4= (m_Edit5 / tan(m_Edit9));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp4);
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9=atan2(m_Edit5,m_Edit4);
  }
if (m_Option == 2)
{
        m_Edit5= m_Edit5 + 0.039;
        root = (temp41*temp41)+(m_Edit4*m_Edit4)-(m_Edit5*m_Edit5);
        m_Edit4 = sqrt(root);
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp4);
```

```
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9=atan2(m_Edit5,m_Edit4);
}

if (m_Edit5 > 9.6 || m_Edit4 <=0 || m_Edit4 > 9.6 || m_Edit6 <=0 || m_Edit6 >9.96
|| root <0)
UpdateData (TRUE);
else
{


        d1 = (m_Edit4*256)/10;
        d2 = (m_Edit5*256)/10;
        d3 = (m_Edit6*256)/10;

        ad1 = (floor(d1) + ceil(d1))/2 - d1;

        if (ad1 > 0)
        {send[0] = floor(d1);}
        else
        {send[0] = ceil(d1);}

        ad2 = (floor(d2) + ceil(d2))/2 - d2;
        if (ad2 > 0)
        {send[1] = floor(d2);}
        else
        {send[1] = ceil(d2);}
        ad3 = (floor(d3) + ceil(d3))/2 - d3;
        if (ad3 > 0)
        {send[2] = floor(d3);}
        else
        {send[2] = ceil(d3);}

m_Edit1 = send[0];
m_Edit2 = send[1];
m_Edit3 = send[2];

for (i=0;i<3;i++)
{
        _outp(888,send[i]);
        Sleep(10);
        _outp(890,i);
        Sleep(10);
        _outp(890,3);
}
UpdateData(FALSE);
```

```
}

}

void CTestDlg::OnRadio1()
{
        m_Upa1.EnableWindow(TRUE);
        m_Da1.EnableWindow(TRUE);
        m_Upa2.EnableWindow(TRUE);
        m_Da2.EnableWindow(TRUE);
        m_Upa3.EnableWindow(FALSE);
        m_Da3.EnableWindow(FALSE);
        m_Option = 0;
}

void CTestDlg::OnRadio2()
{
        m_Upa1.EnableWindow(FALSE);
        m_Da1.EnableWindow(FALSE);
        m_Upa2.EnableWindow(FALSE);
        m_Da2.EnableWindow(FALSE);
        m_Upa3.EnableWindow(TRUE);
        m_Da3.EnableWindow(TRUE);
        m_Option = 1;
}

void CTestDlg::OnRadio3()
{
        m_Upa1.EnableWindow(TRUE);
        m_Da1.EnableWindow(TRUE);
        m_Upa2.EnableWindow(TRUE);
        m_Da2.EnableWindow(TRUE);
        m_Upa3.EnableWindow(FALSE);
        m_Da3.EnableWindow(FALSE);
        m_Option = 2;
}

void CTestDlg::OnButton3()
{

int i;
double send[3];
double d1,d2,d3,ad1,ad2,ad3;
double temp3;
double temp31;
temp3 = m_Edit7;
temp31 = m_Edit4;
if( m_Option == 0 )
```

```
{
        m_Edit4 = m_Edit4 - 0.039;
  m_Edit5= (m_Edit4 * tan(m_Edit9));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp3);
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9= atan2(m_Edit5,m_Edit4);
 }
if (m_Option == 2)
{
        m_Edit4 = m_Edit4 - 0.039;
        m_Edit5 = sqrt((temp31*temp31)+(m_Edit5*m_Edit5)-(m_Edit4*m_Edit4));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp3);
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9=atan2(m_Edit5,m_Edit4);
}

if (m_Edit4 <= 0 || m_Edit5 <=0 || m_Edit5 > 9.6 || m_Edit6 <=0 || m_Edit6 >9.96)
UpdateData (TRUE);
else
{


        d1 = (m_Edit4*256)/10;
        d2 = (m_Edit5*256)/10;
        d3 = (m_Edit6*256)/10;

        ad1 = (floor(d1) + ceil(d1))/2 - d1;

        if (ad1 > 0)
        {send[0] = floor(d1);}
        else
        {send[0] = ceil(d1);}

        ad2 = (floor(d2) + ceil(d2))/2 - d2;
        if (ad2 > 0)
        {send[1] = floor(d2);}
        else
        {send[1] = ceil(d2);}
        ad3 = (floor(d3) + ceil(d3))/2 - d3;
        if (ad3 > 0)
        {send[2] = floor(d3);}
        else
        {send[2] = ceil(d3);}

m_Edit1 = send[0];
m_Edit2 = send[1];
```

```
m_Edit3 = send[2];

for (i=0;i<3;i++)
{
        _outp(888,send[i]);
        Sleep(10);
        _outp(890,i);
        Sleep(10);
        _outp(890,3);
}
UpdateData(FALSE);


}


}



void CTestDlg::OnButton5()
{
int i;
double send[3];
double d1,d2,d3,ad1,ad2,ad3;
double temp5,temp51;
temp5 = m_Edit7;
temp51= m_Edit5;
if( m_Option == 0 )
 {
        m_Edit5 = m_Edit5 - 0.039;
  m_Edit4= (m_Edit5 / tan(m_Edit9));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp5);
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9=atan2(m_Edit5,m_Edit4);
 }
if (m_Option == 2)
{
        m_Edit5= m_Edit5 - 0.039;
        m_Edit4 = sqrt((temp51*temp51)+(m_Edit4*m_Edit4)-(m_Edit5*m_Edit5));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit6= (m_Edit7 * m_Edit6)/(temp5);
        m_Edit8=(m_Edit6/m_Edit7);
        m_Edit9=atan2(m_Edit5,m_Edit4);
}

if (m_Edit5 <=0 || m_Edit4 <=0 || m_Edit4 > 9.6 || m_Edit6 <=0 || m_Edit6 >9.96)
```

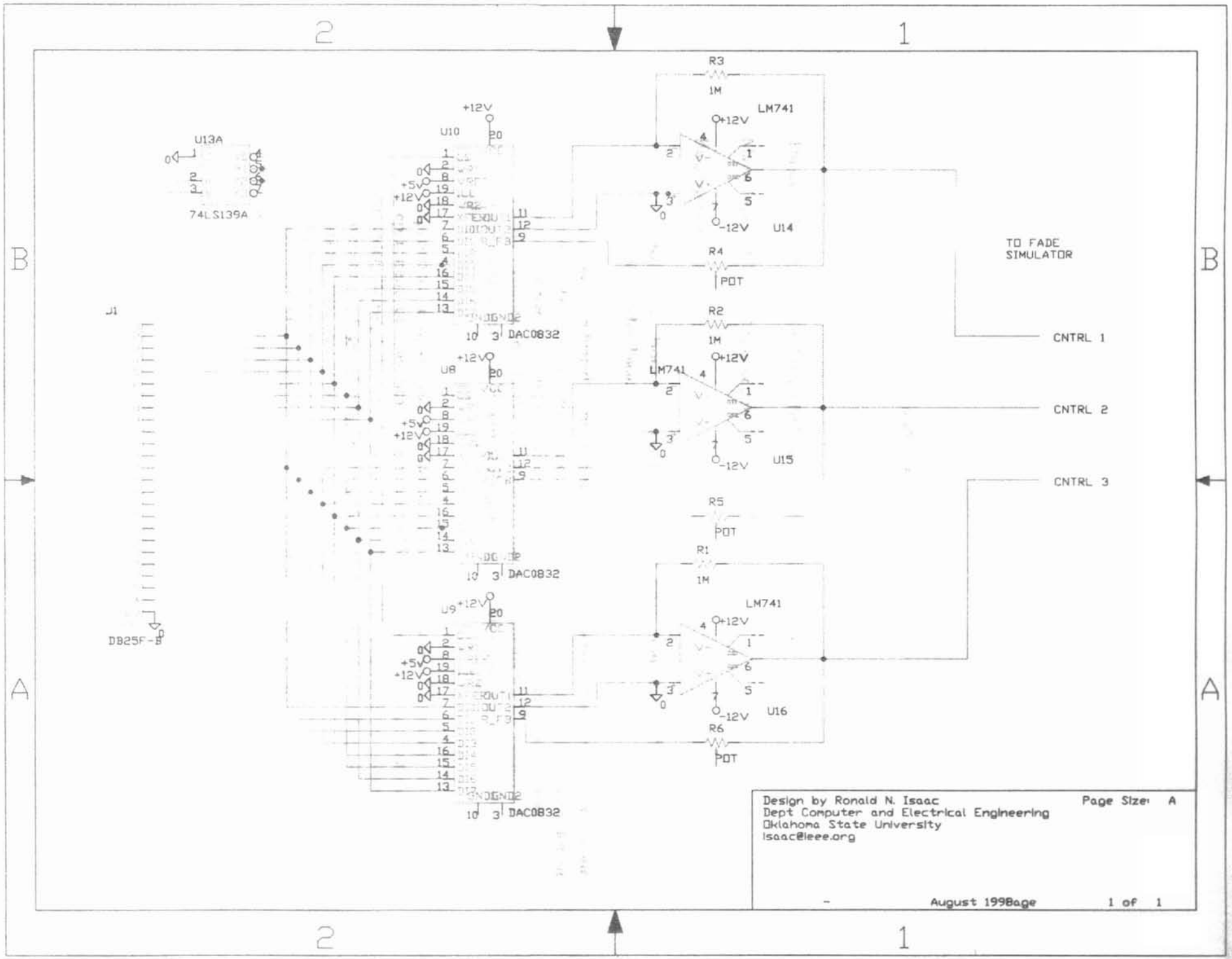
```
UpdateData (TRUE);
else
{

        d1 = (m_Edit4*256)/10;
        d2 = (m_Edit5*256)/10;
        d3 = (m_Edit6*256)/10;

        ad1 = (floor(d1) + ceil(d1))/2 - d1;

        if (ad1 > 0)
        {send[0] = floor(d1);}
        else
        {send[0] = ceil(d1);}

        ad2 = (floor(d2) + ceil(d2))/2 - d2;
        if (ad2 > 0)
        {send[1] = floor(d2);}
        else
        {send[1] = ceil(d2);}
        ad3 = (floor(d3) + ceil(d3))/2 - d3;
        if (ad3 > 0)
        {send[2] = floor(d3);}
        else
        {send[2] = ceil(d3);}

m_Edit1 = send[0];
m_Edit2 = send[1];
m_Edit3 = send[2];

for (i=0;i<3;i++)
{
        _outp(888,send[i]);
        Sleep(10);
        _outp(890,i);
        Sleep(10);
        _outp(890,3);
}
UpdateData(FALSE);


}

}

void CTestDlg::OnButton6()
{
```

```
int i;
double send[3];
double d1,d2,d3,ad1,ad2,ad3;
double temp6;
temp6= m_Edit6;

if( m_Option == 1 )
 {
        m_Edit6 = m_Edit6 + 0.039;
  m_Edit4= (m_Edit5 / tan(m_Edit9));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit8=((m_Edit6*m_Edit8)/temp6);
        m_Edit9=atan2(m_Edit5,m_Edit4);
 }
if (m_Edit6 >9.96 )
UpdateData (TRUE);
else
{


        d1 = (m_Edit4*256)/10;
        d2 = (m_Edit5*256)/10;
        d3 = (m_Edit6*256)/10;

        ad1 = (floor(d1) + ceil(d1))/2 - d1;

        if (ad1 > 0)
        {send[0] = floor(d1);}
        else
        {send[0] = ceil(d1);}

        ad2 = (floor(d2) + ceil(d2))/2 - d2;
        if (ad2 > 0)
        {send[1] = floor(d2);}
        else
        {send[1] = ceil(d2);}
        ad3 = (floor(d3) + ceil(d3))/2 - d3;
        if (ad3 > 0)
        {send[2] = floor(d3);}
        else
        {send[2] = ceil(d3);}

m_Edit1 = send[0];
m_Edit2 = send[1];
m_Edit3 = send[2];

for (i=0;i<3;i++)
{
```

```
        _outp(888,send[i]);
        Sleep(10);
        _outp(890,i);
        Sleep(10);
        _outp(890,3);
}
UpdateData(FALSE);


}

}

void CTestDlg::OnButton7()
{
int i;
double send[3];
double d1,d2,d3,ad1,ad2,ad3;
double temp6;
temp6= m_Edit6;

if( m_Option == 1 )
  {
        m_Edit6 = m_Edit6 - 0.039;
   m_Edit4= (m_Edit5 / tan(m_Edit9));
        m_Edit7= sqrt((m_Edit4*m_Edit4) + (m_Edit5*m_Edit5));
        m_Edit8=((m_Edit6*m_Edit8)/temp6);
        m_Edit9=atan2(m_Edit5,m_Edit4);
  }
if (m_Edit6 <= 0 )
UpdateData (TRUE);

else
{


        d1 = (m_Edit4*256)/10;
        d2 = (m_Edit5*256)/10;
        d3 = (m_Edit6*256)/10;

        ad1 = (floor(d1) + ceil(d1))/2 - d1;

        if (ad1 > 0)
        {send[0] = floor(d1);}
        else
        {send[0] = ceil(d1);}

        ad2 = (floor(d2) + ceil(d2))/2 - d2;
```

```
        if (ad2 > 0)
        {send[1] = floor(d2);}
        else
        {send[1] = ceil(d2);}
        ad3 = (floor(d3) + ceil(d3))/2 - d3;
        if (ad3 > 0)
        {send[2] = floor(d3);}
        else
        {send[2] = ceil(d3);}

m_Edit1 = send[0];
m_Edit2 = send[1];
m_Edit3 = send[2];

for (i=0;i<3;i++)
{
        _outp(888,send[i]);
        Sleep(10);
        _outp(890,i);
        Sleep(10);
        _outp(890,3);
}
UpdateData(FALSE);


}


}

void CTestDlg::OnButton1()
{
        int i;
        double send[3];
        double d1,d2,d3,ad1,ad2,ad3;
        d1 = (m_Edit4*256)/10;
        d2 = (m_Edit5*256)/10;
        d3 = (m_Edit6*256)/10;

        ad1 = (floor(d1) + ceil(d1))/2 - d1;

        if (ad1 > 0)
        {send[0] = floor(d1);}
        else
        {send[0] = ceil(d1);}

        ad2 = (floor(d2) + ceil(d2))/2 - d2;
        if (ad2 > 0)
```

```
        {send[1] = floor(d2);}
        else
        {send[1] = ceil(d2);}
        ad3 = (floor(d3) + ceil(d3))/2 - d3;
        if (ad3 > 0)
        {send[2] = floor(d3);}
        else
        {send[2] = ceil(d3);}

m_Edit1 = send[0];
m_Edit2 = send[1];
m_Edit3 = send[2];

for (i=0;i<3;i++)
{
        _outp(888,send[i]);
        Sleep(1175);
        _outp(890,i);
        Sleep(1175);
        _outp(890,3);
}
UpdateData(FALSE);

}
```


2
1
B
A
R3
1M
LM741
+12V
U10
U13A
74LS139A
J1
DB25F-S
DAC0832
U14
R4
POT
TO FADE
SIMULATOR
R2
U8
U15
CNTRL 1
CNTRL 2
CNTRL 3
R5
R1
U9
U16
R6
Design by Ronald N. Isaac
Dept Computer and Electrical Engineering
Oklahoma State University
Isaac@ieee.org
Page Size: A
August 199Page 1 of 1

VITA 2

Ronald N. Isaac

Candidate for the Degree of

Master of Science

Thesis: ANALYSIS OF MULTIPATH FADING EFFECTS

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Sierra Leone, West Africa, On May 3, 1976, the son of Nadim and Helen Isaac.

Education: Graduated form Lebanese International School, Freetown, Sierra Leone in June 1994 sitting for the London G.C.E 'A' Level exams; received Bachelor of Science degree in Computer Engineering from the University of Balamand, El-Koura, Lebanon in July 1997. Completed the requirements for the Master of Science degree with a major in Electrical Engineering at Oklahoma State University in May, 1999.

Experience: Employed as a network lab assistant by the University of Balamand, Department of Electrical Engineering, 1994; employed by an internet service provider, Beirut, Lebanon, August- December 1997; employed by Oklahoma State University, College of Engineering, Architecture and Technology as a graduate assistant; Oklahoma State University, 1998 to present.

Professional Memberships: Student Member of the Institute of Electrical and Electronics Engineers.