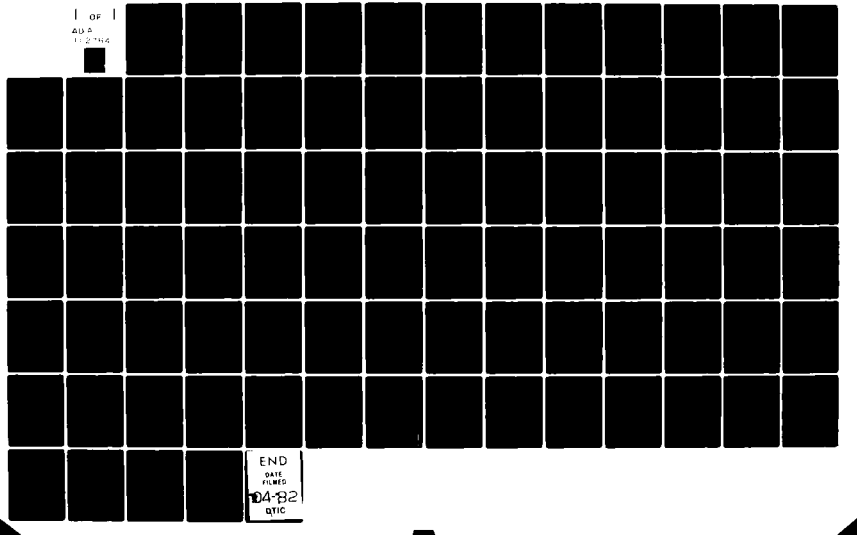
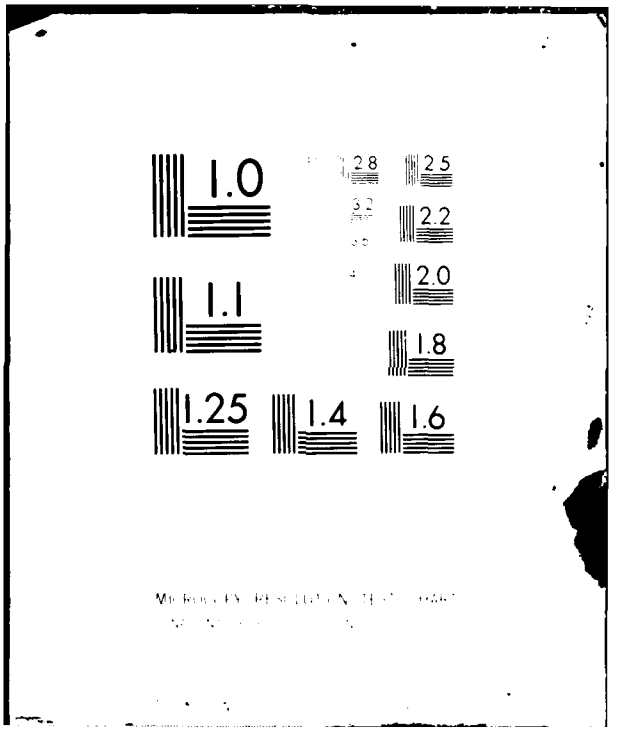
AD-A112 784 OHIO STATE UNIV COLUMBUS ELECTROSCIENCE LAB F/G 17/2
USE OF AN ADAPTIVE ARRAY IN A FREQUENCY-SHIFT KEYED COMMUNICATI--ETC(U)
AUG 80 E C HUDSON N00019-80-C-0181
UNCLASSIFIED ESL-712684-1 NL



END
DATE FILMED
04-82
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5

(1)

OSU

The Ohio State University

USE OF AN ADAPTIVE ARRAY IN A FREQUENCY-SHIFT KEYED COMMUNICATION SYSTEM

Erwin C. Hudson

The Ohio State University

ElectroScience Laboratory

Department of Electrical Engineering
Columbus, Ohio 43212

Technical Report 712684-1

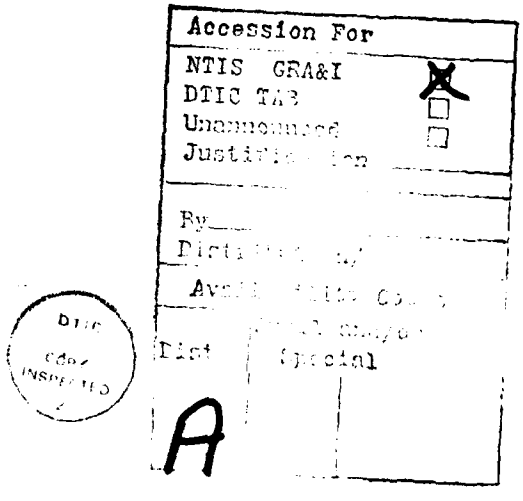
August 1980

Contract N00019-80-C-0181

Department of the Navy
Naval Air Systems Command
Washington, D.C. 20361

NOTICES

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.



SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1 REPORT NUMBER	2. GOVT ACCESSION NO. AD-A112784	3. RECIPIENT'S CATALOG NUMBER
4 TITLE (and Subtitle) USE OF AN ADAPTIVE ARRAY IN A FREQUENCY-SHIFT KEYED COMMUNICATION SYSTEM		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER ESL 712684-1
7 AUTHOR(s) Erwin C. Hudson		8. CONTRACT OR GRANT NUMBER(s) N00019-80-C-0181
9 PERFORMING ORGANIZATION NAME AND ADDRESS The Ohio State University ElectroScience Laboratory, Department of Electrical Engineering, Columbus, Ohio 43212		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Project N00019-80-PR-RJ001
11 CONTROLLING OFFICE NAME AND ADDRESS Department of the Navy Naval Air Systems command Washington, D.C. 20361		12. REPORT DATE August 1980
		13. NUMBER OF PAGES 82
14 MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

The work reported in this report was also used as a thesis submitted to the Department of Electrical Engineering, The Ohio State University as partial fulfillment for the degree Master of Science.

19 KEY WORDS (Continue on reverse side if necessary and identify by block number)

Adaptive Arrays
Communications
Frequency Shift Keying
Interference Protection

20 ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report is a study of a frequency-shift keyed (FSK) communication system with an LMS adaptive array included for interference protection. The major problem addressed is the generation of a reference signal for the adaptive array. The reference signal generation system presented involves converting the FSK bit stream to a Markov sequence, allowing bit predictions to be made at the adaptive array. The bit predictions are used by a feedback circuit to form

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20.

the reference signal. A computer simulation program is presented and simulation results are used to demonstrate the operation of the overall FSK system. Also included is a study of the effects of bit prediction errors and reference signal phase error.

TABLE OF CONTENTS

CHAPTER I
INTRODUCTION

An adaptive array is an antenna system that can automatically adjust its antenna pattern. The antenna pattern is adjusted to track desired signals and at the same time, to steer nulls toward interfering signals. Such antennas are especially useful for protecting communication links from interference or jamming, and have both commercial and military applications.

Several techniques for controlling the weights in an adaptive array have been studied. These include the LMS array[1], the Applebaum array[2], and the Shor array[3].

This thesis is concerned with the use of the LMS adaptive array in a frequency-shift keyed (FSK) communication system[4]. The LMS array has previously been used in several types of communication systems. For example, the LMS array was used in an early adaptive system using amplitude moduation[5]. In another experimental system, binary information was biphase modulated and received by an LMS array[6].

The communication system that we will examine is a binary frequency shift keyed system[4]. We assume a sequency of bits, 1's and 0's, is to be transmitted. As illustrated in Figure I-1, a 0 is transmitted by sending a burst of signal at frequency ω_0, and a 1 by a burst signal at frequency ω_1. In either case, the time duration of a bit is T. Thus the transmitted signal shifts back and forth between the two frequencies, ω_0 and ω_1. At the receiver, bit detection is done by determining, during each bit interval, whether the received signal has frequency ω_0 or ω_1.

The adaptive array that we will consider is the LMS array. The LMS array adjusts its antenna pattern to achieve least mean-square error between the received signal and a reference signal. Ideally, the reference signal should be an exact replica of the desired signal. However, in an actual communication system, the received signal is not known in advance. Therefore, the reference signal is obtained by estimating the desired signal from the array output.

The FSK system studied in this thesis uses bit predictions to generate a reference signal. In Chapter II, a coding procedure is presented which allows the desired signal bits to be predicted at

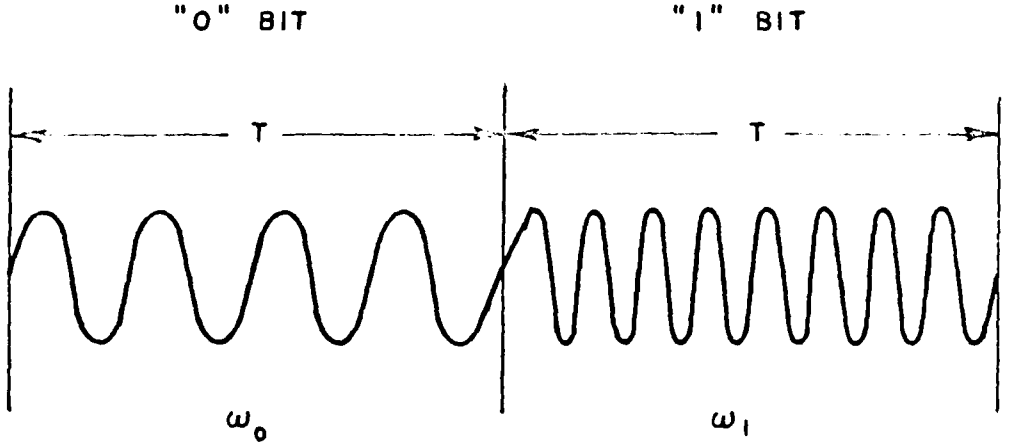


Figure I-1. FSK modulation.

the receiver. Each bit is predicted from the previous received bit. A processing loop, described in Chapter IV, uses the bit predictions to process the array output and to form a reference signal. A computer simulation program has been used to demonstrate the performance of the entire system. Finally, Chapter V considers some possible problems associated with the FSK system.

CHAPTER II
THE LMS ADAPTIVE ARRAY

A. Introduction

In this chapter, we will discuss the LMS algorithm used in many adaptive arrays. This discussion will provide some background information on adaptive arrays and also establish notation to be used later. We will begin with the configuration of the adaptive array. Then we will derive the LMS algorithm used to control the array weights. In the next part of this chapter, we will introduce an analytic signal model of the LMS array. The use of analytic signals makes the notation more compact and easier to work with. We will use the analytic signal notation in the following chapters to study an adaptive array in an FSK communication system.

B. The LMS Algorithm

A diagram of the LMS adaptive array is shown in Figure II-1. The signal received by each element of the array is processed by a quadrature hybrid (QH). A QH is a power divider which separates its input signal into quadrature components. We will refer to one QH output signal as the in-phase component, $X_{Ii}(t)$, and the other as the quadrature component, $X_{Qi}(t)$. The subscript i indicates the antenna element with which the signal is associated. We assume N antenna elements and therefore i=1,2,$\cdots$,N. The signals are used to define a signal vector

$$X_r(t) = \begin{pmatrix} X_{I1} \\ X_{Q1} \\ X_{I2} \\ X_{Q2} \\ \vdots \\ X_{IN} \\ X_{QN} \end{pmatrix} . \qquad \text{(II-1)}$$

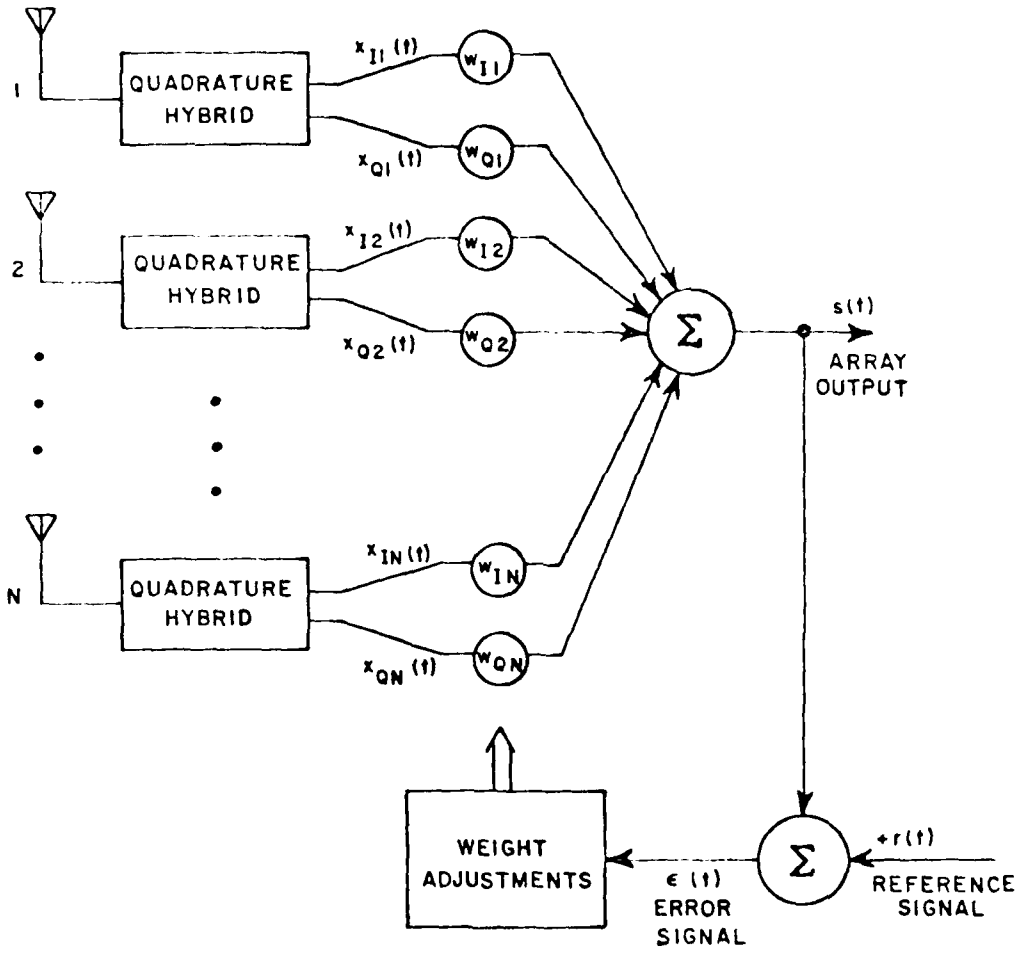


Figure II-1. The LMS array.

The signal vector is subscripted with an r to indicate a vector of real components. Later, when we introduce analytic signals, the signal vector will have complex components.

Each of the QH output signals is multiplied by a weight. Let $X_{Ii}(t)$ be multiplied by weight w_{Ii} and $X_{Qi}(t)$ be multiplied by weight w_{Qi}. The weights are used to define a weight vector,

$$W_r = \begin{pmatrix} w_{I1} \\ w_{Q1} \\ w_{I2} \\ w_{Q2} \\ \vdots \\ w_{IN} \\ w_{QN} \end{pmatrix} \quad . \tag{II-2}$$

The weighted signals are summed to produce the array output, s(t):

$$s(t) = W_r^T X_r(t) = X_r^T(t) W_r \quad . \tag{II-3}$$

The LMS array requires a reference signal, r(t) in Figure II-1, which is an estimate of the incoming desired signal. The array output is subtracted from the reference signal to form an error signal,

$$\epsilon(t) = r(t) - s(t) \quad . \tag{II-4}$$

The feedback network adjusts the weights to minimize the mean square value of the error signal, $E\{\epsilon^2(t)\}$ where $E\{\cdot\}$ denotes the expectation.

The weight vector that minimizes the mean square error satifies the system of equations,

$$\nabla_{Wr} E\{\epsilon^2(t)\} = 0 \quad , \tag{II-5}$$

where ∇_{Wr} denotes the gradient with respect to the weight vector, W_r. Substituting Equations (II-3) and (II-4) into Equation (II-5) yields,

$$\nabla_{W_r}[r^2(t) - 2W_r^T E\{X_r(t)r(t)\} + W_r^T E\{X_r(t)X_r^T(t)\}W_r] = 0 \quad . \tag{II-6}$$

The equation may be simplified be defining the Covariance Matrix,

$$\Phi_r = E\{X_r(t)X_r^T(t)\} \quad , \tag{II-7}$$

and the Reference Correlation Vector,

$$S_r = E\{X_r(t)r(t)\} \quad . \tag{II-8}$$

Using these definitions, Equation (II-6) becomes,

$$\nabla_{W_r}[r^2(t)-2W_r^T S_r+W_r^T \Phi_r W_r] = 0 \quad . \tag{II-9}$$

Taking the gradient as indicated gives,

$$-2S_r + 2\Phi_r W_{r_{opt}} = 0 \quad . \tag{II-10}$$

Therefore, the optimum weights with respect to minimum mean square error are given by,

$$W_{r_{opt}} = \Phi^{-1} S_r \quad . \tag{II-11}$$

To make the adaptive array, we need a feedback network which will make the weights converge to $W_{r_{opt}}$. In this work, we assume the weights are controlled by the LMS algorithm[1]. This algorithm implements the gradient technique given by,

$$\frac{dw_{Ii}}{dt} = - k \frac{\partial}{\partial w_{Ii}} \epsilon^2(t) \quad , \tag{II-12}$$

and

$$\frac{dw_{Qi}}{dt} = - k \frac{\partial}{\partial w_{Qi}} \epsilon^2(t) \quad , \tag{II-13}$$

where k is a gain constant.

In practice, k must be chosen small enough that w_{Ii} and w_{Qi} do not respond to the instantaneous fluctuations in $\epsilon^2(t)$, but only to the

average value of $\epsilon^2(t)$[7]. Hence we may approximate Equations (II-12) and (II-13) by introducing expected values,

$$\frac{dw_{Ii}}{dt} = -k \frac{\partial}{\partial w_{Ii}} E\{\epsilon^2(t)\} \quad , \tag{II-14}$$

and

$$\frac{dw_{Qi}}{dt} = -k \frac{\partial}{\partial w_{Qi}} E\{\epsilon^2(t)\} \quad . \tag{II-15}$$

Equations (II-14) and (II-15) can be combined with vector notation into

$$\frac{dW_r}{dt} = -k \nabla_{W_r} E\{\epsilon^2(t)\} \quad . \tag{II-16}$$

Taking the gradient as was done for Equation (II-9) gives

$$\frac{dW_r}{dt} = -k[-2S_r+2\Phi_r W_r] \quad , \tag{II-17}$$

or

$$\frac{dW_r}{dt} + 2k\ \Phi_r W_r = 2k\ S_r \quad . \tag{II-18}$$

In steady state, when $dW_r/dt=0$, we have,

$$2k\ \Phi_r W_{r_{ss}} = 2k\ S_r \quad , \tag{II-19}$$

or

$$W_{r_{ss}} = \Phi_r^{-1}\ S_r \quad . \tag{II-20}$$

We see that the steady state weights in the LMS algorithm equal the optimum weights calculated in Equation (II-14)

The feedback circuit that implements the LMS algorithm is easily found. Substituting Equations (II-4) into Equations (II-12) yields

$$\frac{dw_{Ii}}{dt} = -k \frac{\partial}{\partial w_{Ii}} [r^2(t)-2r(t)s(t)+s^2(t)] \quad , \tag{II-21}$$

which becomes,

$$\frac{dw_{Ii}}{dt} = -k [-2r(t)X_{Ii}(t)-2 X_{Ii}(t)s(t)] \quad . \tag{II-22}$$

Factoring out the $X_{Ii}(t)$ term, we have,

$$\frac{dw_{Ii}}{dt} = 2k X_{Ii}(t)\varepsilon(t) \quad . \tag{II-23}$$

A similar result is found for the quadrature weights,

$$\frac{dw_{Qi}}{dt} = 2k X_{Qi}(t)\varepsilon(t) \quad . \tag{II-24}$$

From either Equation (II-23) or Equation (II-24), we find the feedback circuit of Figure II-2. A feedback circuit of this type is required for each weight in the adaptive array.

C. Analytic Signal Notation

As was noted earlier, it is useful to describe the LMS array with analytic signal notation. To do this, we observe that the two outputs of a quadrature hybrid are related by the equation,

$$\hat{X}_{Ii} = X_{Qi} \quad , \quad \hat{X}_{Qi} = -X_{Ii} \tag{II-25}$$

where "^" indicates the Hilbert transform[8],

$$\hat{g}(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{g(t')dt'}{t-t'} \quad . \tag{II-26}$$

Hence we may combine the two outputs of each quadrature hybrid into an analytic signal,

$$\tilde{X}_i(t) = X_{Ii} + j X_{Qi} = X_{Ii} + j \hat{X}_{Ii} \quad . \tag{II=27}$$

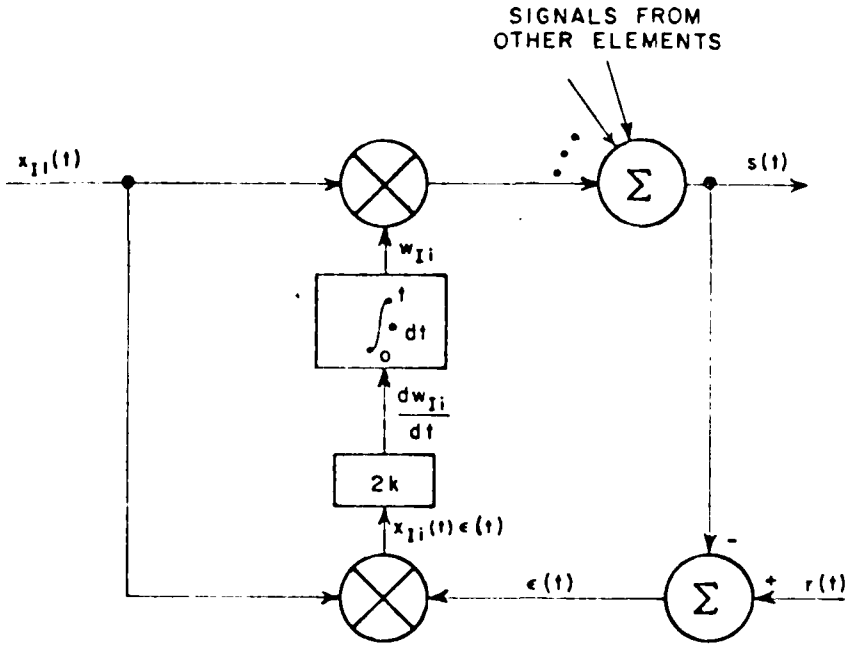


Figure II-2. The LMS array feedback loop.

We also combine each pair of quadrature weights into a single complex weight,

$$w_i = w_{Ii} - j\, w_{Qi} \quad . \tag{II-28}$$

A minus sign is used here to reduce the number of conjugates required in the resulting equations. We also define the analytic signals,

$$\tilde{r}(t) = r(t) + j\, \hat{r}(t) \quad , \tag{II-29}$$

$$\tilde{\epsilon}(t) = \epsilon(t) + j\, \hat{\epsilon}(t) \quad , \tag{II-30}$$

and

$$\tilde{s}(t) = s(t) + j\, \hat{s}(t) \quad , \tag{II-31}$$

for the reference signal, the error signal and the array output.

The analytic signals $\tilde{X}_i(t)$ may be combined into a complex signal vector X,

$$X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{pmatrix} \quad . \tag{II-32}$$

Similarly, the complex weight vector is defined to be

$$W = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{pmatrix} \quad . \tag{II-33}$$

By writing out the real and imaginary parts, one can verify that the analytic signal corresponding to the array output is given by

$$\tilde{s}(t) = W^T X = X^T W \quad . \tag{II-34}$$

The differential equation for the LMS array in terms of analytic signals is easily derived. First we combine Equations (II-23) and (II-24) into the complex equation

$$\frac{dw_{Ii}}{dt} - j\frac{dw_{Qi}}{dt} = 2k[E\{X_{Ii}(t)\varepsilon(t)\} - j\,E\{X_{Qi}(t)\varepsilon(t)\}] \quad . \tag{II-35}$$

Using the equality [9]

$$E\{X_{Ii}(t)\varepsilon(t)\} = E\{\hat{X}_{Ii}(t)\hat{\varepsilon}(t)\} = E\{X_{Qi}(t)\hat{\varepsilon}(t)\} \tag{II-36}$$

and the similar relation for $E\{X_{Qi}(t)\varepsilon(t)\}$, we can rewrite Eq. (II-35) as

$$\frac{d}{dt}(W_{Ii} - jW_{Qi}) = kE\{X_{Ii}(t)\varepsilon(t) + X_{Qi}(t)\hat{\varepsilon}(t) - jX_{Qi}(t)\varepsilon(t) + jX_{Ii}(t)\hat{\varepsilon}(t)\} \tag{II-37}$$

Now recognizing that

$$E\{X_{Ii}(t)\epsilon(t) + X_{Qi}(t)\hat{\epsilon}(t) - jX_{Qi}(t)\epsilon(t) + jX_{Ii}(t)\hat{\epsilon}(t)\} = E\{X_i^*(t)\tilde{\epsilon}(t)\} \quad \text{(II-38)}$$

we find

$$\frac{dw_i}{dt} = k\, E\{X_i^*(t)\tilde{\epsilon}(t)\} \quad . \quad \text{(II-39)}$$

In vector notation, Equation (II-39) becomes

$$\frac{dW}{dt} = k\, E\{X^*(t)\tilde{\epsilon}(t)\} \quad . \quad \text{(II-40)}$$

If we replace $\tilde{\epsilon}(t)$ in Equation (II-40) with

$$\tilde{\epsilon}(t) = \tilde{r}(t) - \tilde{s}(t) = \tilde{r}(t) - X^T W \quad , \quad \text{(II-41)}$$

we find that,

$$\frac{dW}{dt} = k\,[E\{X^*(t)\tilde{r}(t)\} - E\{X^*(t)X^T(t)\}W] \quad . \quad \text{(II-42)}$$

In this complex notation, we define the covariance matrix to be

$$\Phi = E\{X^*(t)X^T(t)\} \quad , \quad \text{(II-43)}$$

and the reference correlation vector to be

$$S = E\{X^*(t)\tilde{r}(t)\} \quad . \quad \text{(II-44)}$$

These definitions allow us to write Equation (II-42) in the form,

$$\frac{dW}{dt} + k\Phi W = kS \quad . \quad \text{(II-45)}$$

The complex steady state weights, found when dW/dt=0 are given by

$$W_{ss} = \Phi^{-1} S \quad . \tag{II-46}$$

It is sometimes convenient to break the signal vector into three components,

$$X = X_d + X_i + X_n \quad , \tag{II-47}$$

where X_d is due to desired signal, X_i is due to interference, and X_n is due to noise. If we assume that the three components are uncorrelated, we find

$$\Phi = E\{X^*X^T\} = E\{X_d^*X_d^T\} + E\{X_i^*X_i^T\} + E\{X_n^*X_n^T\} \quad . \tag{II-48}$$

Using the notation,

$$\Phi_d = E\{X_d^*X_d^T\} \quad , \tag{II-49}$$

$$\Phi_i = E\{X_i^*X_i^T\} \quad , \tag{II-50}$$

and

$$\Phi_n = E\{X_n^*X_n^T\} \quad , \tag{II-51}$$

we can write the covariance matrix,

$$\Phi = \Phi_d + \Phi_i + \Phi_n \quad . \tag{II-52}$$

We see that dividing the signal vector into three uncorrelated components allows us to write the covariance matrix as the sum of three matrices, each involving one signal only. This notation will be useful in Chapter V.

The analytic signal representation of the LMS array can be visualized in terms of a block diagram. Rewriting Equation (II-39) without the expectation operator, we have,

$$\frac{dw_i}{dt} = k\, X_i^*(t)\varepsilon(t) \quad . \tag{II-53}$$

This equation corresponds to the feedback circuit of Figure II-3.

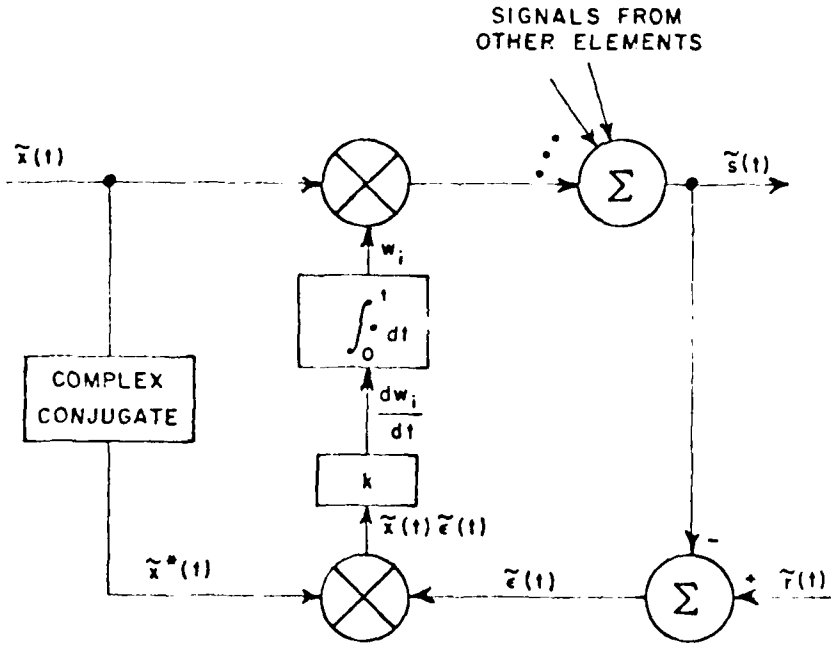


Figure II-3. Complex LMS loop.

The advantage of the analytic signal notation can be seen by comparing Equations (II-20) and (II-46). To calculate steady state weights with real notation, one must find the inverse of Φ_r, a 2N by 2N matrix for an N element array. However, in the analytic signal notation, Φ is an NxN matrix and its inverse is much easier to compute.

D. Summary

The LMS adaptive array was introduced in this chapter as background for the following chapters. We noted that the LMS array minimizes the mean-square error between the array output and a so-called reference signal. In the remaining chapters, we address the problem of how to obtain a suitable reference signal in a binary FSK communication system.

CHAPTER III
PROPERTIES OF THE DESIRED SIGNAL BIT STREAM

A. Introduction

The LMS array compares the array output signal to a reference signal and minimizes the mean square error. We now turn to the problem of obtaining a reference signal in a binary FSK communication system.

The technique we will study assumes the transmitted bits in the FSK signal come from a first-order Markov source[10]. We will take advantage of the Markov bit transition probabilities to make a one bit prediction of the desired signal that can be used to form the reference signal.

In this chapter, we will describe the overall system concept and present a method of modifying the transmitted bit stream to convert it to a Markov source. In Chapter IV, we will show how a bit prediction scheme may be used to form a reference signal for the adaptive array.

B. The Desired Signal

Figure III-1 shows a block diagram of the communication system envisioned in this study. A source of binary information is assumed to be given. The information bits from this source are sent to a receiver equipped with an adaptive array. As shown in Figure III-1, the information bits will first be coded into a new set of bits and then the new coded bits will be transmitted to the adaptive array and receiver. At the receiver, the array output is first demodulated to regain the coded bits, and these are then passed through a decoder to recover the original information bits.

The purpose of coding the transmitted bits is to cause them to have the properties of a Markov source[10]. These properties allow the generation of a reference signal at the receiver. The coding which converts the information bits to a Markov source will enable us to make bit predictions at the adaptive array. Bit predictions are necessary because the reference signal in the adaptive array must be a real-time estimate of the desired signal. Since a received bit is not detected until the end of its bit interval, knowledge of this bit

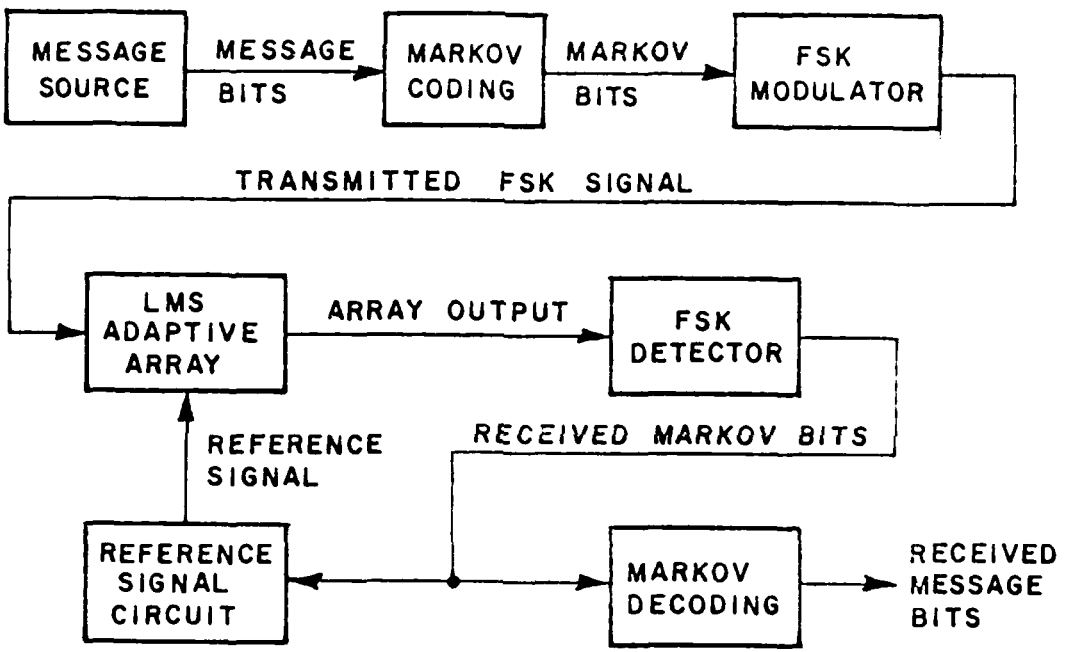


Figure III-1. Block diagram of the FSK communication system.

is not available until it is too late for use in the reference signal. However, if each bit is used to predict the next bit to be received (i.e., one bit in advance), an estimate of the desired signal can be available in time for use in the reference signal. The purpose of coding the information bits into a Markov source is to make it possible to make such a one bit prediction, with a better than average probability of being correct. Of course, if useful information is to be transmitted and received, we cannot predict every bit correctly. The coding process will allow us to keep the number of bit prediction errors low enough to produce a suitable reference signal in the FSK system.

C. A Markov Source

To describe the coding procedure for the FSK bits, we need to look at some statistics of the FSK bit stream. Therefore, we will introduce notation for the bit probabilities. $P_n(0)$ will indicate the probability of a 0 during the n^{th} bit interval. Likewise, $P_n(1)$ will be the probability of a 1 on the n^{th} bit. The notation $P_n(0/1)$, will denote the probability of a 0 on the n^{th} bit interval given that a 1 occured on the $(n-1)^{th}$ bit. The other conditional probabilities, $P_n(0/0)$, $P_n(1/1)$ and $P_n(1/0)$, are defined in the same manner.

The coding procedure that we will use at the transmitter converts the information bit sequence into a Markov bit sequence[10]. In an n^{th} order Markov sequence, the probability of the next bit is a function of the previous n bits. We will consider in particular, a first order Markov bit sequence. For this case, the probability of a 1 or 0 on each bit interval depends only on which bit was transmitted previously. This relationship may be visualized in terms of the transition diagram as shown in Figure III-2. The arrows in the diagram represent bit transitions from the $(n-1)^{th}$ bit interval to the n^{th} bit interval. We can add the probabilities and write the equations,

$$P_n(0) = P_n(0/0)P_{n-1}(0) + P_n(0/1)P_{n-1}(1) \qquad \text{(III-1)}$$

and

$$P_n(1) = P_n(1/0)P_{n-1}(0) + P_n(1/1)P_{n-1}(1) \quad . \qquad \text{(III-2)}$$

We will further assume that the first order Markov sequence is stationary and symmetric. By stationary, we mean that the probabilities do not change from bit interval to bit interval. Therefore, we can drop the subscripts from Equations (III-1) and (III-2) and write,

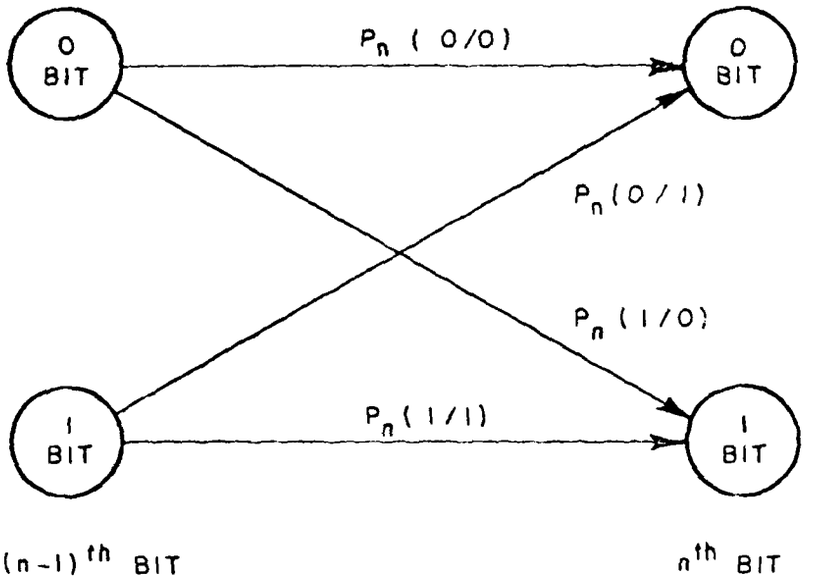


Figure III-2. Transition diagram for a first-order Markov source.

$$P(0) = P(0/0)P(0) + P(0/1)P(1) \qquad \text{(III-3)}$$

and

$$P(1) = P(1/0)P(0) + P(1/1)P(1) \quad . \qquad \text{(III-4)}$$

The assumption that the Markov sequence is symmetric means,

$$P(0/1) = P(1/0) = q \qquad \text{(III-5)}$$

and

$$P(0/0) = P(1/1) = 1-q \quad . \qquad \text{(III-6)}$$

Substituting Equations(III-5) and (III-6) into Equations (III-3) and (III-4) gives,

$$P(0) = P(1) = \frac{1}{2} \quad . \tag{III-7}$$

In summary, a first order stationary and symmetric Markov bit sequence is described by the equations,

$$P(0) = P(1) = \frac{1}{2} \quad , \tag{III-8}$$

$$P(\text{bit change}) = q \quad , \tag{III-9}$$

and

$$P(\text{no bit change}) = 1-q \quad . \tag{III-10}$$

At the receiver, the Markov nature of the desired signal allows us to make bit predictions. Assume for example that the probability of a bit change is greater than one half,

$$P(1/0) = P(0/1) > \frac{1}{2} \quad . \tag{III-11}$$

Then we can predict the next bit by assuming it to be the opposite of the last bit. The probability of a correct prediction is equal to q, the bit change probability. The probability of a bit prediction error is equal to 1-q.

D. Coding and Decoding of the Message Bits

As noted above, we want to transmit a Markov bit stream to the adaptive array. Therefore to send an arbitrary binary message, we must first convert the message bits into Markov bits. We will now illustrate, in the form of an example, how an arbitrary bit sequence may be converted into a Markov bit sequence.

A method of forming a Markov bit stream from an arbitrary bit sequence with equally likely bit transitions is illustrated in Figure III-3. In this illustration, each group of 3 message bits is mapped into a group of four Markov bits for transmission to the adaptive array. The mapping is done by first listing all possible 4 bit words and counting the number of bit transitions in each word. Then we choose the 8 words which have the highest number of bit transitions. The set of 8 words chosen are marked with a check in the figure. In Figure III-3(b) we list all possible combinations of 3 bits which are the message words. Then to each of these 3 bit message words, we assign one of the 4 bit words marked with a check. These assignments

4 BIT WORDS	NUMBER OF TRANSITIONS
0000	0
0001	1
0010	2 ✓
0011	1
0100	2 ✓
0101	3 ✓
0110	2 ✓
0111	1
1000	1
1001	2 ✓
1010	3 ✓
1011	2 ✓
1100	1
1101	2 ✓
1110	1
1111	0

(a)

MESSAGE BITS	TRANSMITTED BITS
000	0010
001	0100
010	0101
011	0110
100	1001
101	1010
110	1011
111	1101

(b)

Figure III-3. Markov coding example.

are used at the transmitter so that for each 3 message bits, 4 Markov bits are transmitted. We find for the Markov bits,

$$q = P(\text{bit transition}) = 0.69 \quad . \tag{III-12}$$

We note that by transmitting the Markov bits instead of the message bits, the number of transmitted bits is increased by the ratio 4/3. If the message bit rate is held constant, the coding process will require an increased transmission bit rate. Consequently, the bandwidth of the FSK signal must be increased as well.

The bandwidth increase can be related to the transition probabilities of the Markov bits. Let H_M denote the entropy of the message bits and H_C the entropy of the coded Markov bits. The coded bit stream is a symmetric first order Markov sequence and therefore[11]

$$H_C = q \log_2 \frac{1}{q} + (1-q) \log_2 \frac{1}{1-q} \quad . \tag{III-13}$$

Let B_M denote the bandwidth of the FSK signal when the message bits are transmitted without being coded into Markov bits. B_C will be the bandwidth of the FSK signal when the Markov bits are transmitted. Since the message bit rate is held constant, we can write the required bandwidth of the transmitted Markov bits as

$$B_{C_{min}} = B_M \frac{H_M}{H_C} \quad . \tag{III-14}$$

Also we can define the minimum bandwidth spreading factor,

$$F = \frac{H_M}{H_C} \quad . \tag{III-15}$$

If we assume that the message bits have unit entropy,

$$H_M = 1 \quad , \tag{III-16}$$

we can write,

$$F = \frac{1}{q \log_2 \frac{1}{q} + (1-q) \log_2 \frac{1}{1-q}} \quad . \tag{III-17}$$

A plot of Equation (III-17) is shown in Figure III-4. We note that as a result of doubling the bandwidth of the transmitted FSK signal, we can make bit predictions with a 0.9 probability of being correct.

Equation (III-17) also allows us to compare the coding example of Figure III-3 to theoretical results. In the example, a bandwidth spreading ratio of 4/3 resulted in a bit change probability of 0.7. From the plot in Figure III-4, we see that the maximum theoretical bit change probability for the example case is 0.8. Although the coding scheme in the example does not give optimum results, it does demonstrate the feasibility of the idea. In this report, we will not consider the general problem of converting a bit stream into a Markov bit stream of given characteristics. Rather, we will just use this example to illustrate that it can be done.

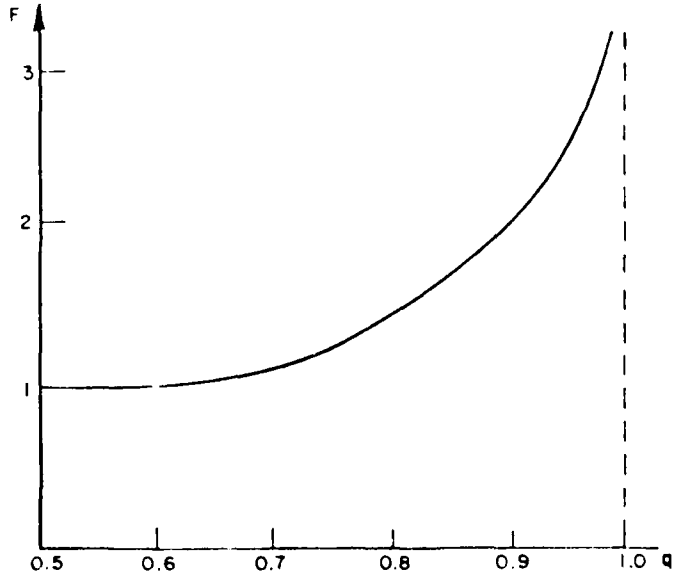


Figure III-4. Bandwidth spread versus prediction probability.

CHAPTER IV
REFERENCE SIGNAL GENERATION

A. Introduction

The previous chapter presented a coding procedure for the bit stream. The bits are coded to make the bit stream first-order Markov. By detecting the last bit received and knowing the bit transition probabilities of the Markov code, we can make bit predictions at the adaptive array. In this chapter, we will describe the operation of the adaptive array reference signal generation circuity and will show how the bit predictions are used in forming a reference signal from the array output. Next we will demonstrate the formation of a reference signal when the adaptive array receives CW interference in addition to desired signal. The examples in this chapter will assume that perfect bit predictions are made at the receiver. In the next chapter, we will look at the effects of bit prediction errors on the performance of the system.

B. The Adaptive Array and the FSK Detector

Figure IV-1 shows a block diagram of an adaptive array with its reference signal generation circuitry. The array is the LMS array described in Chapter II. An incoherent FSK bit detector is connected to the array output and produces the received bit stream. A prediction circuit looks at the received bits and makes bit predictions according to the Markov properties of the bits. The array output and the bit predictions are used by the reference signal processing loop to produce a reference signal.

The detector behind the LMS adaptive array is assumed to be incoherent. The block diagram of an incoherent FSK detector[12] is shown in Figure IV-2. The signal is split two channels, each containing a bandpass filter and an envelope detector. In the upper or 0 channel, the filter has center frequency ω_0 corresponding to the 0 frequency of the FSK signal. The filter in the lower channel has center frequency ω_1 corresponding to the 1 frequency. The output of each filter is then envelope detected. If, for example, at the end of each bit interval, the output of the envelope detector in the 0 channel is greater in magnitude than that in the 1 channel, a 0 bit

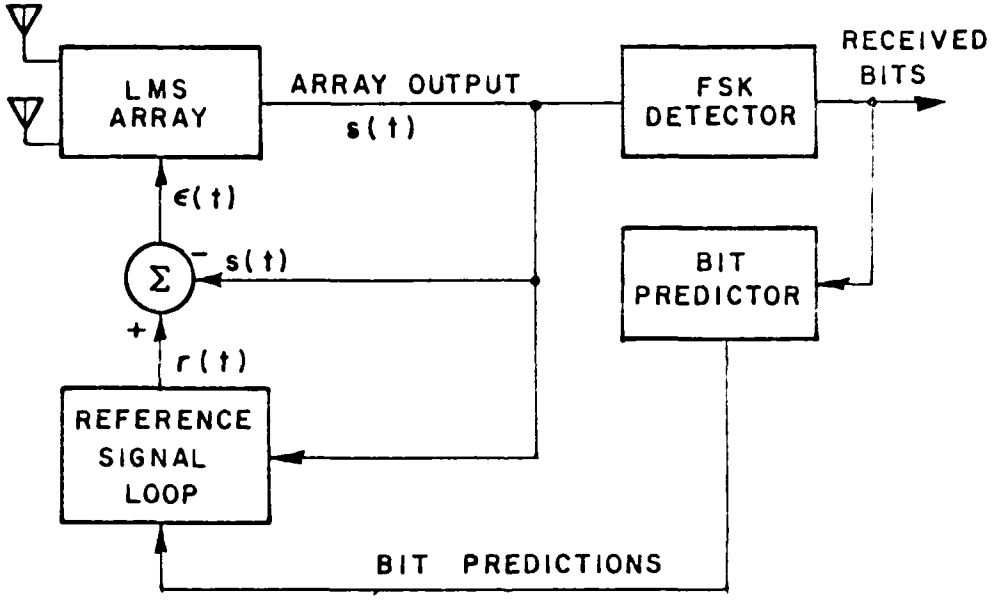


Figure IV-1. Reference signal generator circuitry.

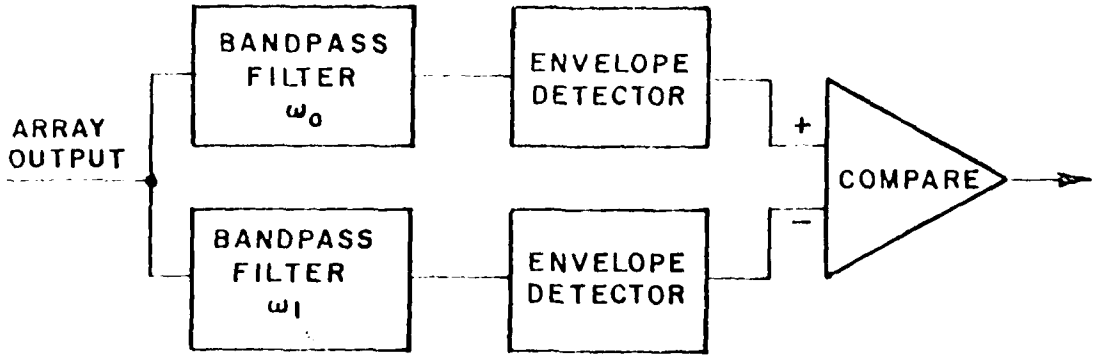


Figure IV-2. Incoherent FSK bit detector.

selected. Likewise, when the output of the 1 channel is greater a 1 bit is detected.

During a particular bit interval, one channel of the FSK detector contains both signal and noise while the other channel contains only noise. Bit detection errors are made whenever the magnitude of the noise in the noise only channel exceeds the magnitude of the signal plus noise in the other channel. We will assume temporarily in this section that no bit detection errors are made. Later, we will look at the effects of bit detection errors on the performance of the system.

C. The Reference Signal Processing Loop

A block diagram of the reference signal processing loop is shown in Figure IV-3. The array output and the bit predictions are inputs to the loop and the loop output is the reference signal. In the following discussion, we will look individually at the bandpass filter, the local oscillator, and the limiter. Then we will show how the loop

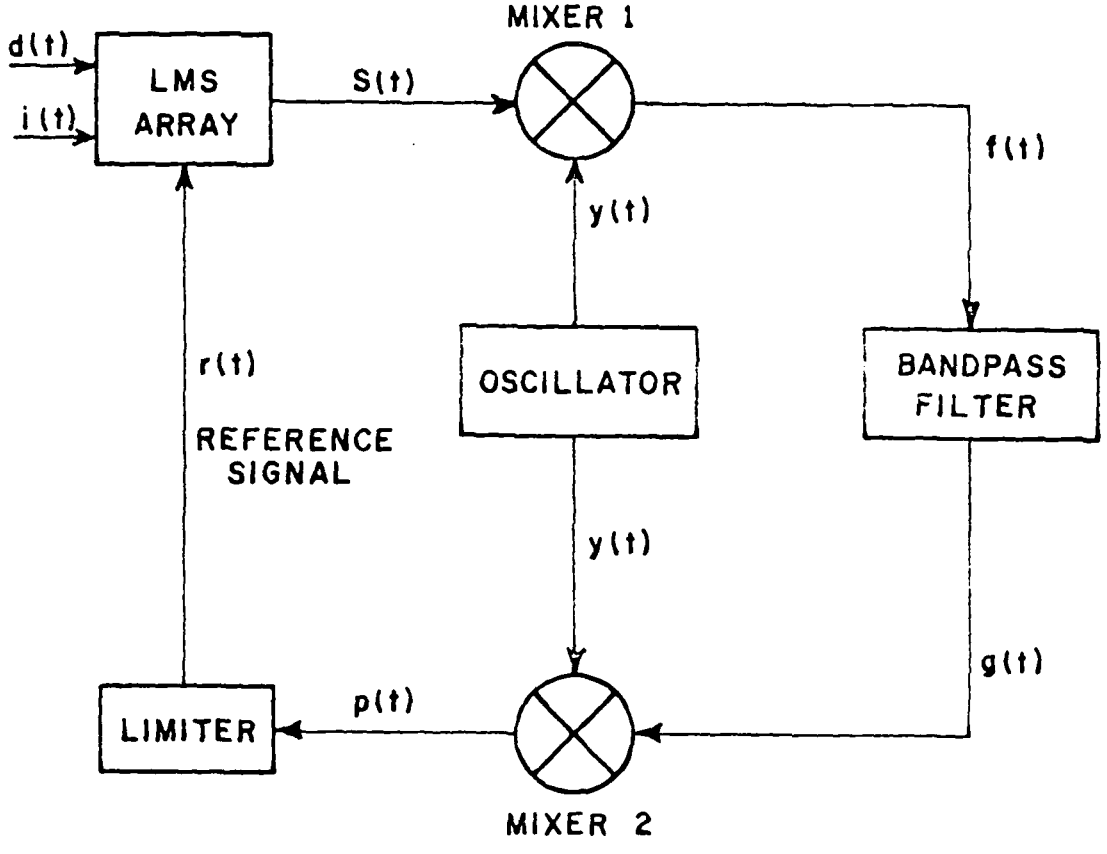


Figure IV-3. Reference signal loop circuitry.

operates when the array output consists of both desired signal and interference.

The bandpass filter is used in the loop to separate the desired and the undesired components of the array output. A typical amplitude response of the filter is illstrated in Figure IV-4. The filter has a center frequency of ω_c and its half power bandwidth is B.

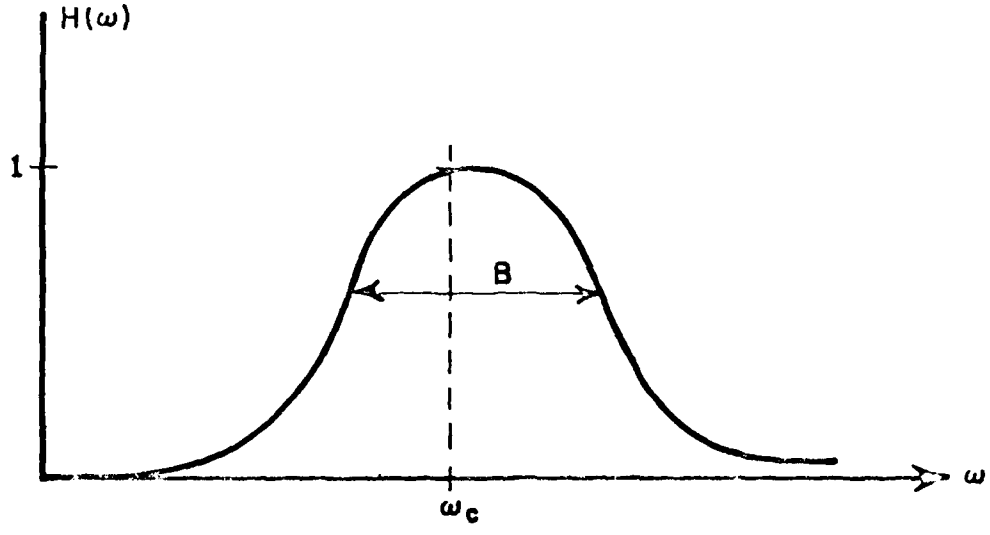


Figure IV-4. Transfer function of the reference loop bandpass filter.

The oscillator shifts the frequency of the array output to the center of the bandpass filter. Ideally, if ω_{osc} is the frequency of the oscillator signal,

$$\omega_{osc} = \begin{cases} \omega_c - \omega_0 & \text{during a 0 bit} \\ \omega_c - \omega_1 & \text{during a 1 bit} \end{cases} . \qquad \text{(IV-1)}$$

We do not know the bit being received, however, until the end of the bit interval. Therefore, we set the oscillator frequency according to the predicted bit for the current bit interval. The oscillator frequency then is

$$\omega_{osc} = \begin{cases} \omega_c - \omega_0 & \text{when a 0 is predicted} \\ \omega_c - \omega_1 & \text{when a 1 is predicted} \end{cases} \quad . \qquad \text{(IV-2)}$$

A limiter is included in the loop so the reference signal will have fixed amplitude. A typical transfer function for a limiter is shown in Figure IV-5. Below the saturation voltage R, the device behaves like a linear amplifier with a gain of G. When the device is saturated, the output signal is clipped and remains below voltage R.

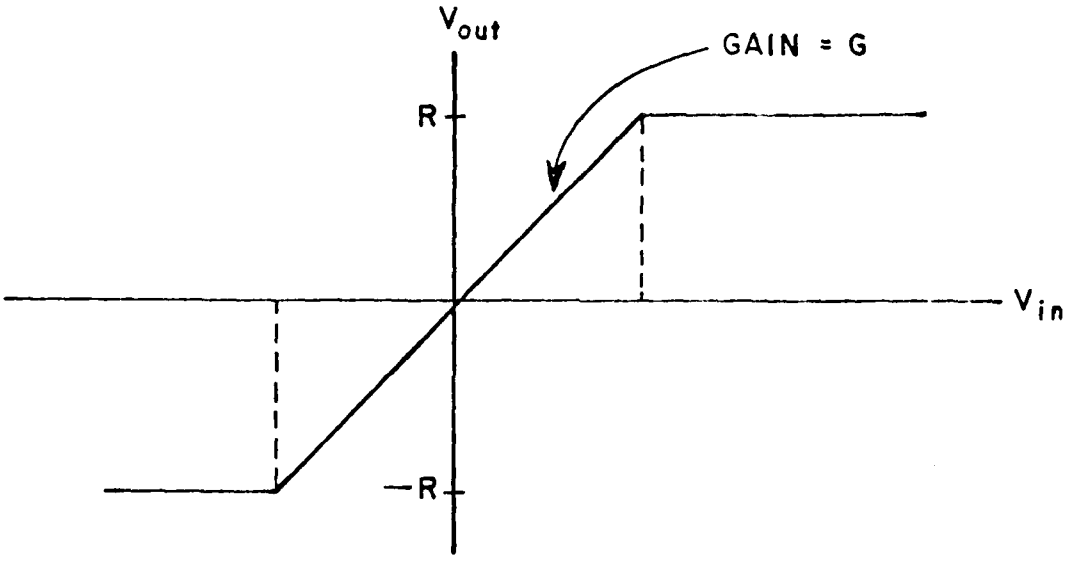


Figure IV-5. Transfer function of the limiter.

D. Operation of the Reference Signal Loop

We will assume in our discussion of the reference signal loop that the array receives both an FSK desired signal and a CW interference signal. The incoming desired signal at the array elements, in real notation, is given by,

$$d(t) = \sqrt{2}\, Ad \cos(\omega_d t + \psi_d) \quad , \qquad \text{(IV-3)}$$

where

$$\omega_d = \begin{cases} \omega_0 & \text{during a 0 bit} \\ \omega_1 & \text{during a 1 bit} \end{cases} \quad . \qquad \text{(IV-4)}$$

We use $\sqrt{2}$ Ad for the signal amplitude so that the outputs of the quadrature hybrids will have amplitude Ad. The phase angle, ψ_d, is assumed uniformly distributed,

$$p(\psi_d) = \begin{cases} \frac{1}{2\pi} & -\pi < \psi_d < \pi \\ 0 & \text{elsewhere} \end{cases} \quad . \qquad \text{(IV-5)}$$

The desired signal component of the array output will be of the form,

$$s_d(t) = B_d \cos(\omega_d t + \gamma_d) \quad , \qquad \text{(IV-6)}$$

where the amplitude, B_d, and the phase, γ_d, incorporate the gain and phase shift caused by the array weights. Similarly, we assume the interference signal is given by,

$$i(t) = \sqrt{2}\, A_i \cos(\omega_i t + \psi_i) \quad . \qquad \text{(IV-7)}$$

The magnitude of the interference is $\sqrt{2}\, A_i$ and ω_i is the interference frequency. The phase angle, ψ_i, is assumed uniformly distributed,

$$p(\psi_i) = \begin{cases} \frac{1}{2\pi} & -\pi < \psi_i < \pi \\ 0 & \text{elsewhere} \end{cases} \quad . \qquad \text{(IV-8)}$$

The array output component associated with the interference is,

$$s_i(t) = B_i \cos(\omega_i t + \gamma_i) \quad . \qquad \text{(IV-9)}$$

As with the desired signal, the amplitude B_i and the phase angle γ_i indicate the effects of the array weights. The total array output due to desired signal, interference, and noise is,

$$s(t) = s_d(t)+s_i(t)+s_n(t) \quad . \tag{IV-10}$$

We will assume that the desired signal to noise ratio is high enough to neglect noise in the reference loop. This assumption, however, does not imply that noise is not a factor in the bit detection process in the receiver. Neglecting noise, we can write the array output,

$$s(t) = B_d \cos(\omega_d t+\gamma_d) + B_i \cos(\omega_i t+\gamma_i) \quad . \tag{IV-11}$$

Refering to Figure IV-3, we see that the array output enters the reference signal loop and is mixed with the oscillator signal. For now, we will assume that perfect bit predictions are made and therefore the oscillator frequency is

$$\omega_{osc} = \omega_c - \omega_d \quad . \tag{IV-12}$$

The problems caused by bit prediction errors will be discussed in Chapter V.

The output of the loop oscillator is given by,

$$y(t) = \cos[(\omega_c-\omega_d)t+\gamma_{osc}] \quad , \tag{IV-13}$$

where the amplitude is assumed to be unity and γ_{osc} is the phase angle.

The output of Mixer 1 is the product of the oscillator signal and the array output. The mixer output is

$$f(t) = [B_d \cos(\omega_d t+\gamma_d)+B_i \cos(\omega_i t+\gamma_i)] \cos[(\omega_c-\omega_d)t+\gamma_{osc}] \quad . \tag{IV-14}$$

Equation (IV-14) can be expanded into

$$f(t) = \frac{B_d}{2} \cos[\omega_c t+\gamma_d+\gamma_{osc}] + \frac{B_d}{2} \cos[(\omega_c-2\omega_d)t-\gamma_d+\gamma_{osc}]$$
$$+ \frac{B_i}{2} \cos[(\omega_c+\omega_i-\omega_d)t+\gamma_i+\gamma_{osc}] + \frac{B_i}{2} \cos[(\omega_c-\omega_i-\omega_d)t-\gamma_i+\gamma_{osc}]. \tag{IV-15}$$

The amplitude response, $H(\omega)$, and the phase response, $\theta(\omega)$, of the bandpass filter are shown in Figure IV-4. We will assume that the frequencies $\omega_c-\omega_d$ and $\omega_c-(\omega_i+\omega_d)$ are outside the bandpass of the filter. Therefore, we can drop these components from the filter input and write,

$$f'(t) = \frac{B_d}{2} \cos[\omega_c t+\gamma_d-\gamma_{osc}] + \frac{B_i}{2} \cos[(\omega_c+\omega_i-\omega_d)t+\gamma_i+\gamma_{osc}] \quad . \tag{IV-16}$$

The filter output signal is given by,

$$g(t) = \frac{B_d}{2} H(\omega_c)\cos(\omega_c t+\gamma_d+\gamma_{osc}+\theta(\omega_c)) + \frac{B_i}{2} H(\omega_c+\omega_i-\omega_d)\cos((\omega_c+\omega_i-\omega_d)t+\gamma_i+\gamma_{osc}+\theta(\omega_c+\omega_i-\omega_d)) \quad . \tag{IV-17}$$

At Mixer 2, the output of the bandpass filter is multiplied by the oscillator signal. The second mixer output is given by,

$$p(t) = \left\{ \frac{B_d}{2} H(\omega_c)\cos[\omega_c t+\gamma_d+\gamma_{osc}+\theta(\omega_c)] + \frac{B_i}{2} H(\omega_c+\omega_i-\omega_d)\cos[(\omega_c+\omega_i-\omega_d)t+\gamma_i+\gamma_{osc}+\theta(\omega_c+\omega_i-\omega_d)] \right\} \cdot \cos[(\omega_c+\omega_d)t+\gamma_{osc}] \quad . \tag{IV-18}$$

We will neglect the terms in Equation (IV-18) with frequencies of the order of $2\omega_d$ under the assumption that the transient response of the adaptive array is too slow to respond to them. If this assumption were not valid, we would add a zonal filter at the output of Mixer 2. In either case, we can drop the high frequency terms and write the second mixer output,

$$p(t) = \frac{B_d}{4} H(\omega_c)\cos[\omega_d t+\gamma_d+\theta(\omega_c)] + \frac{B_i}{4} H(\omega_c+\omega_i-\omega_d)\cos[\omega_i t+\gamma_i+\theta(\omega_c+\omega_i-\omega_d)] \quad . \tag{IV-19}$$

The adaptive array, in the absence of noise, will make the magnitude of the array output track the magnitude of the reference signal. If the reference signal amplitude is greater than the array output amplitude, the weights and the array output will continuously increase. A limiter is included in the reference signal loop to keep the array output at a fixed level, inside the dynamic range of the electronics. The limiter is connected to the output of Mixer 2, and its input is given by Equation (IV-19). As illustrated in Figure IV-5, the limiter behaves like a linear amplifier below saturation.

We will assume that both terms in Equation (IV-19) fall below the limiter cutoff voltage. This assumption is valid because of the relationship between the magnitudes of the reference signal and the array output. If the limiter were in the clipping region and if the reference signal magnitude were smaller than the array output magnitude, the array output would be decreased. The array output would continue to be reduced until the limiter input signal reaches the knee of the transfer function. Therefore, we can write the limiter output, which is the reference signal, as,

$$r(t) = \frac{B_d}{4} G\, H(\omega_c)\cos[\omega_c t+\gamma_d+\theta(\omega_c)] + \frac{B_d}{4} G\, H(\omega_c+\omega_i-\omega_d)\cos[\omega_i t+\gamma_i+\theta(\omega_c+\omega_i-\omega_d)] \quad . \qquad \text{(IV-20)}$$

The reference signal, given by Equation (IV-20), consists of both a desired and an undesired component. The loop gain for the desired component, measured between the array output and the reference signal, is,

$$L_d = \frac{G}{4} H(\omega_c) \quad . \qquad \text{(IV-21)}$$

The loop gain for the interference component of the reference signal is given by,

$$L_i = \frac{G}{4} H(\omega_c+\omega_i-\omega_d) \quad . \qquad \text{(IV-22)}$$

Since we are temporarily considering the case of perfect bit predictions, the loop gain for the desired signal is not a function of which bit was transmitted. The loop gain for the interference component, however, contains an ω_d term and is therefore a function of the transmitted bit stream. The average loop gain for the interference can be written,

$$L_i = P(0)\,\frac{G}{4}\,H(\omega_c+\omega_i-\omega_o)+P(1)\,\frac{G}{4}\,H(\omega_c+\omega_i-\omega_1) \quad . \qquad \text{(IV-23)}$$

For the special case where the bits are equally probable,

$$L_i = \frac{G}{8}\,[H(\omega_c+\omega_i-\omega_o) + H(\omega_c+\omega_i-\omega_1)] \quad . \qquad \text{(IV-24)}$$

The relationship between the amplitude of the reference signal and the amplitude of the array output signal is an important parameter of the reference loop. This parameter is the loop gain measured from the array output around the reference loop to the reference signal port. To explain the importance of the reference loop gain, we will consider the case where the signals passing through the loop are below the limiter cutoff amplitude. In this situation, we want the interference signal passing through the reference loop to be attenuated so we make the loop gain for the interference less than one,

$$L_i = \frac{G}{8}\,[H(\omega_c+\omega_i-\omega_o) + H(\omega_c+\omega_i-\omega_1)] < 1 \quad . \qquad \text{(IV-25)}$$

At the same time, we want desired signal passing through the reference loop to be amplified so we require

$$L_d = \frac{G}{4}\,H(\omega_c) > 1 \quad . \qquad \text{(IV-26)}$$

If Equations (IV-25) and (IV-26) are satisfied, the desired part of the array output will increase up to the limiter cutoff voltage while the interference magnitude is decreased. Therefore the array will steer a beam maximum toward the desired signal and form a null on the interference.

E. Computer Simulation of the FSK System

A computer simulation program was written to study the performance of the reference signal processing loop. The program simulates a two element LMS array, a bit prediction circuit, and a reference signal loop. In this section, we will describe the simulation and present some results. A description of the program and a copy of the Fortran code are included in the Appendix at the end of this thesis.

The geometry of the two element array is shown in Figure IV-6. The antennas are assumed to have omnidirectional element patterns and

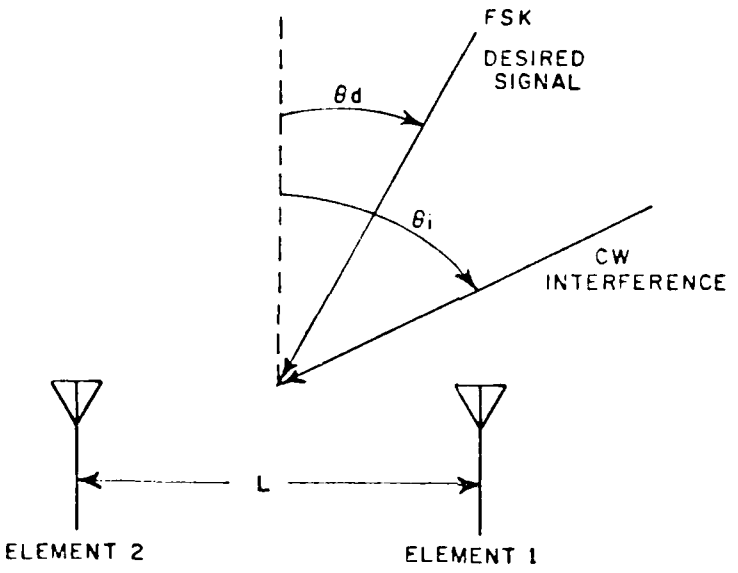


Figure IV-6. The geometry of the two-element adaptive array.

to be separated by a distance of L meters. Both incoming signals are assumed to be in a plane containing the array elements. The incident FSK signal, in complex notation, is given by

$$\tilde{d}(t) = \sqrt{2}\, A_d\, e^{j(\omega_d t + \psi_d)} \quad . \qquad \text{(IV-7)}$$

The desired signal arrives at an angle θ_d measured from broadside. Since the frequency of the desired signal changes from bit to bit, the interelement phase shift depends on which bit is being received. For a 0 bit, the interelement phase shift is

$$\psi_d(\omega_0) = \frac{\omega_0 L}{c} \sin\theta_d \quad , \qquad \text{(IV-8)}$$

and for a 1 bit,

$$\phi_d(\omega_1) = \frac{\omega_1 L}{c} \sin\theta_d \quad , \tag{IV-29}$$

where L is the antenna element separation and c is the speed of light. We assume that the two FSK frequencies are close enough together so that we can average the two phase shifts and use

$$\phi_d = \frac{\phi_d(\omega_o)+\phi_d(\omega_1)}{2} \tag{IV-30}$$

for the calculations in this chapter. The complex form of the interference signal notation, is given by,

$$i(t) = \sqrt{2}\, A_i\, e^{j(\omega_i t+\psi_i)} \quad , \tag{IV-31}$$

and the interelement phase shift is written,

$$\phi_i = \frac{2\pi L}{\omega_i} \sin\theta_i \quad . \tag{IV-32}$$

We also assume that each received signal component contains a noise term with power σ^2.

As defined in Chapter II, the signal vector due to the desired signal is

$$X_d = A_d \begin{pmatrix} 1 \\ e^{-j\phi_d} \end{pmatrix} e^{j(\omega_d t+\psi_d)} \quad . \tag{IV-33}$$

The interference component of the signal vector is given by,

$$X_i = A_i \begin{pmatrix} 1 \\ e^{-j\phi_i} \end{pmatrix} e^{j(\omega_i t+\psi_i)} \quad , \tag{IV-34}$$

and the noise component is

$$X_n = \begin{pmatrix} \tilde{n}_1(t) \\ \tilde{n}_2(t) \end{pmatrix} \quad , \tag{IV-35}$$

where

$$E\{\tilde{n}_1(t)\tilde{n}_1^*(t)\} = E\{\tilde{n}_2(t)\tilde{n}_2^*(t)\} = \sigma^2 \quad , \tag{IV-36}$$

and

$$E\{\tilde{n}_1(t)\tilde{n}_2^*(t)\} = 0 \quad . \tag{IV-37}$$

The results of the simulation program consist of plots of each of the four weights versus time and an antenna pattern taken at the end of the simulation period. We will present the results of two simulations in this section. First will be for the case of an FSK desired signal, perfect bit predictions, and no interference. In the second simulation, we will include CW interference directly on one of the two FSK frequencies.

A list of input data for the first simulation example is shown in Figure IV-7. The FSK desired signal is incident from broadside and has unit magnitude. The selection of frequencies in the simulation is rather arbitrary since the simulation is not done in real time. Scaling a frequency up or down merely changes the scale on the time axis in the results. Desired signal frequencies of 990 hertz for a 0 and 1010 hertz for a 1 were chosen for the simulation. The center frequency of the bandpass filter in the reference signal loop was chosen to be 2000 hertz. Also, at the center of the bandpass filter,

$$H(\omega_c) = 1 \quad . \tag{IV-38}$$

The limited gain is set to 2,

$$G = 2 \quad , \tag{IV-39}$$

so that the loop gain for the desired portion of the reference signal is

$$L_d = 2 \quad . \tag{IV-40}$$

The antennas are separated by one-half of the average wavelength of the desired signal. Finally, the array convergence factor, k, was adjusted so that the weights would reach steady-state in approximately one-half of the simulation period.

The results of the first simulation example are presented in Figure IV-8. The four weight curves are identical because the desired signal is at broadside and therefore both antenna elements receive

```
        ADAPTIVE ARRAY SIMULATION PROGRAM INPUT DATA

INCOMING SIGNALS:
     DESIRED FSK SIGNAL
        "0" BIT FREQUENCY=990 HERTZ
        "1" BIT FREQUENCY=1010 HERTZ
        POWER=0 DB
        ANGLE=0 DEGREES FROM BROADSIDE
        BIT RATE=10 BITS/SECOND
        MARKOV TRANSITION PROBABILITY=1.

     INTERFERENCE SIGNAL
        NONE

     NOISE
        POWER/ELEMENT=-10 DB

ADAPTIVE ARRAY
        ELEMENTS=2
        SEPARATION=0.5 DESIRED SIGNAL WAVELENGTHS
        CONVERGENCE FACTOR(K)=0.00005

REFERENCE SIGNAL LOOP
     OSCILLATOR
        FREQUENCY FOR "0" BIT PREDICTION=1010 HERTZ
        FREQUENCY FOR "1" BIT PREDICTION=990 HERTZ
        OUTPUT AMPLITUDE=0 DB
     LOOP BANDPASS FILTER
        CENTER FREQUENCY=2000 HERTZ
        BANDWIDTH=10 HERTZ
        CENTER FREQUENCY GAIN=0 DB
     LIMITER
        CUTOFF VOLTAGE=0 DB
        GAIN BELOW CUTOFF=3 DB
```

Figure IV-7. Simulation input data for the case of no interference.

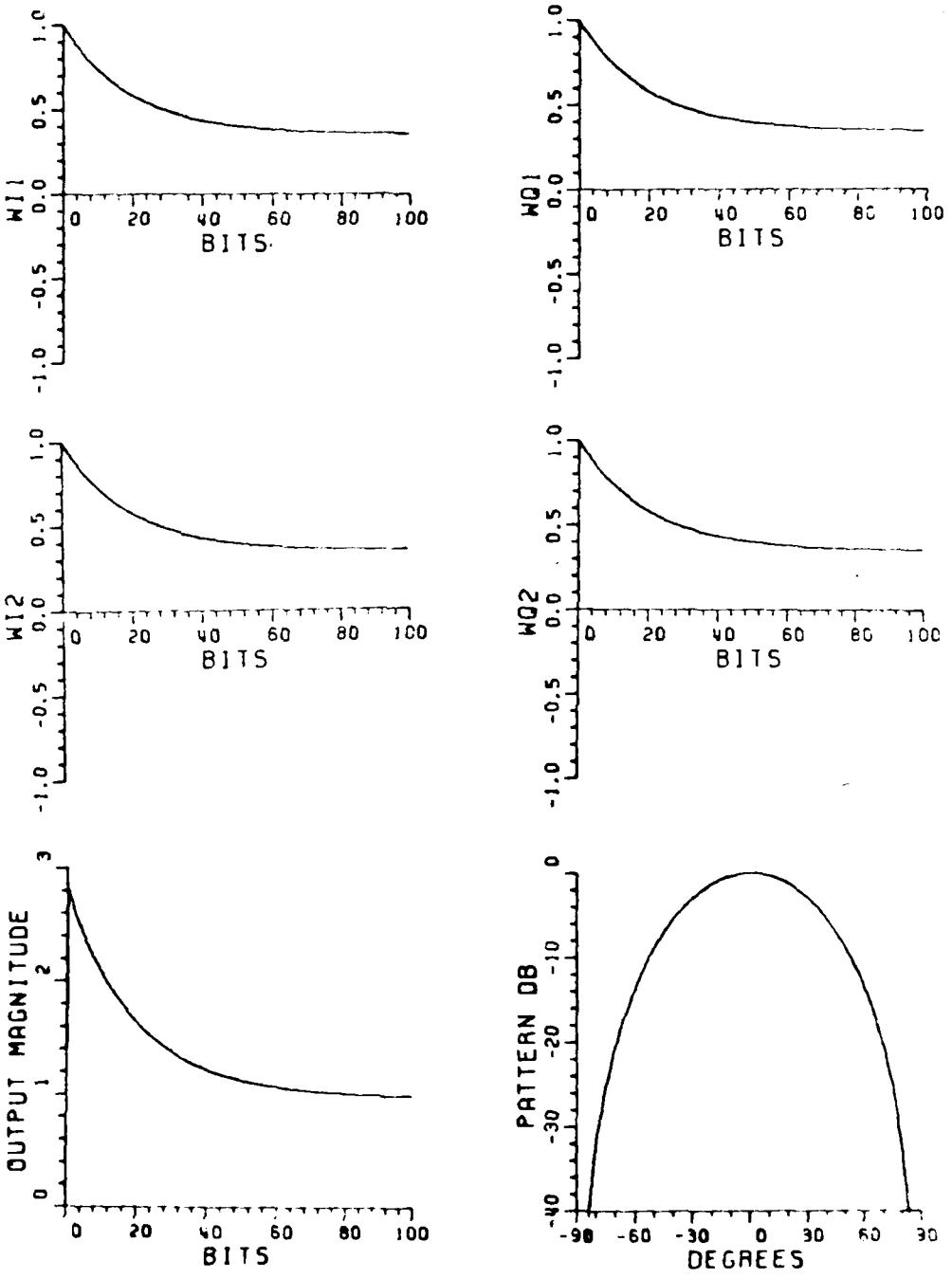


Figure IV-8. Simulation results corresponding to the input data of Figure IV-7.

the same signal. The steady-state antenna pattern has its main beam in the direction of the desired signal indicating proper operation of the adaptive array.

For the second simulation example, we will add CW interference directly on one of the two FSK bits. A list of input data for the second simulation is shown in Figure IV-9. Except for the addition of the interference, all the input parameters are unchanged from the previous case.

The addition of the interfernece affects not only the adaptive array response, but also the bit detection and bit prediction processes. During lock-up, the array output component due to interference may have much higher power than the desired component. As a result, the FSK detector will not be able to detect the desired signal bits. We assume in the simulation program however that bit detection and timing acquisition continue even when the desired signal power is below the interference power at the array output. In Chapter V, we will look at the problems of timing acquisition and bit detection during lock-up.

The results of the system simulation with interference included are shown in Figure IV-10. Note that with the addition of interference, the weights are not smooth curves. The roughness in the weight curves appears because the desired signal and the interference sometimes look alike to the adaptive array. When a 0 bit is received along with CW interference on the 0 bit frequency, the array tries to acquire both signals. During the early part of the adaptation period, the tendency to acquire both signals results in abrupt changes in the weights. As the null forms and the array output decreases, the weight curves become smoother. In steady-state, the interference is practically eliminated from the array output and therefore the weight curves become smooth.

The antenna pattern taken at the end of the simulation period is also shown in Figure IV-10. A null of about 40 dB in the direction of the interference shows that the system is functioning properly.

```
        ADAPTIVE ARRAY SIMULATION PROGRAM INPUT DATA

INCOMING SIGNALS:
     DESIRED FSK SIGNAL
        "0" BIT FREQUENCY=990 HERTZ
        "1" BIT FREQUENCY=1010 HERTZ
        POWER=0 DB
        ANGLE=0 DEGREES FROM BROADSIDE
        BIT RATE=10 BITS/SECOND
        MARKOV TRANSITION PROBABILITY=1.

     INTERFERENCE SIGNAL
        FREQUENCY=990 HERTZ
        POWER=10 DB
        ANGLE=45 DEGREES FROM BROADSIDE

     NOISE
        POWER/ELEMENT=-10 DB

ADAPTIVE ARRAY
        ELEMENTS=2
        SEPARATION=0.5 DESIRED SIGNAL WAVELENGTHS
        CONVERGENCE FACTOR(K)=0.00005

REFERENCE SIGNAL LOOP
     OSCILLATOR
        FREQUENCY FOR "0" BIT PREDICTION=1010 HERTZ
        FREQUENCY FOR "1" BIT PREDICTION=990 HERTZ
        OUTPUT AMPLITUDE=0 DB
     LOOP BANDPASS FILTER
        CENTER FREQUENCY=2000 HERTZ
        BANDWIDTH=10 HERTZ
        CENTER FREQUENCY GAIN=0 DB
     LIMITER
        CUTOFF VOLTAGE=0 DB
        GAIN BELOW CUTOFF=3 DB
```

Figure IV-9. Simulation input data for the case of CW interference.

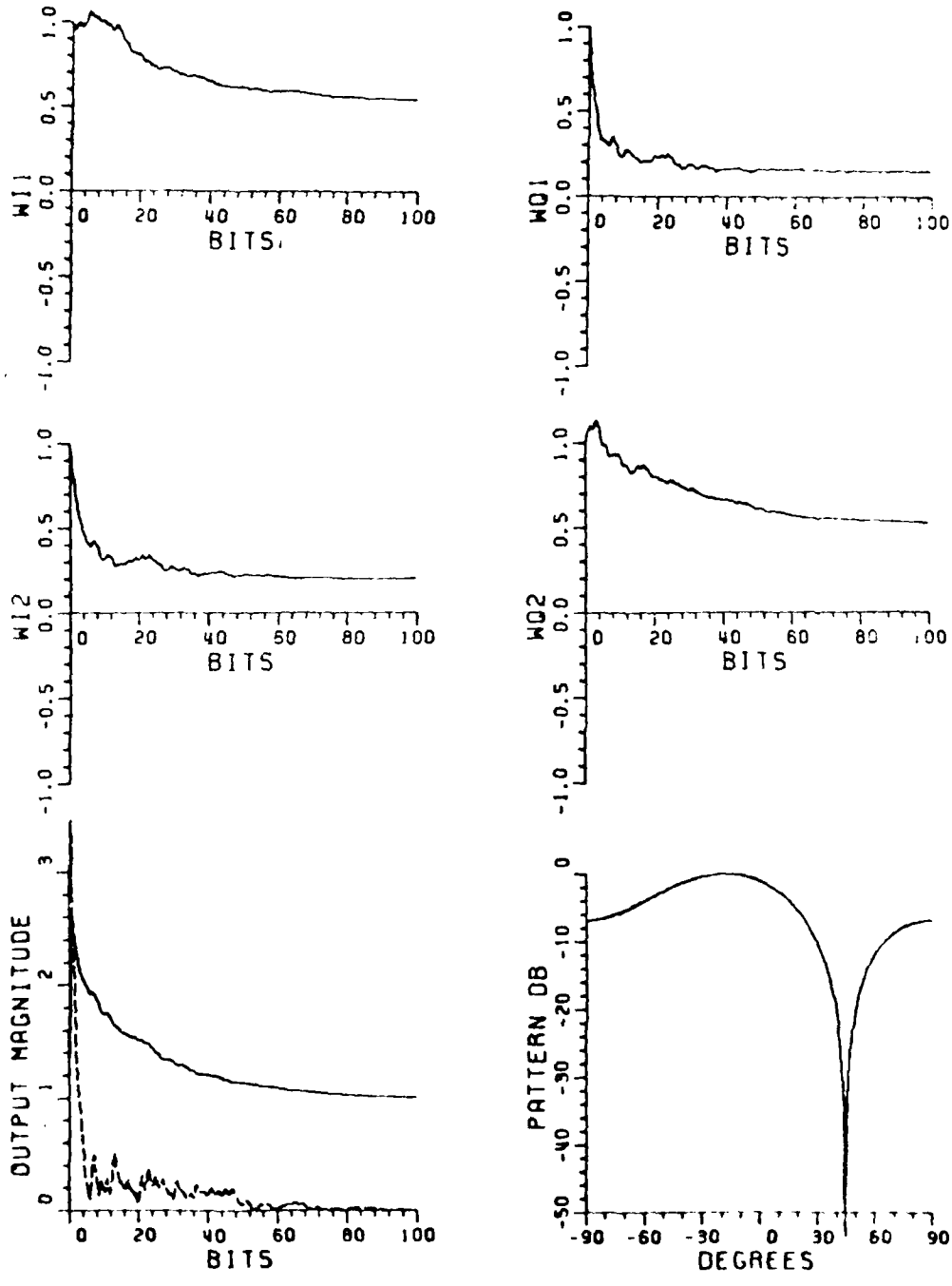


Figure IV-10. Simulation results corresponding to the input data of Figure IV-9.

CHAPTER V
REFERENCE LOOP DESIGN FACTORS

A. Introduction

In this chapter, we will look at some aspects of the reference signal loop which can affect the performance of the adaptive array. First we will consider the problem of bit prediction errors. The simulation program, introduced in the previous chapter, will be used to show the effects of bit prediction errors on the array weights. The second problem that we will consider occurs when there is a phase shift between the array output and the reference signal. We will show that this phase error causes the array weights to oscillate. Also, we will show that in the presence of interference, the signal to interference plus noise ratio (SINR) at the array output decreases with increasing phase error. Finally, we will look briefly at the problem of synchronizing the bit detector and the bit predictions with the received bit stream.

B. Bit Prediction Errors

We demonstrated, in the previous chapter, the performance of the reference signal loop for the case of perfect bit predictions. If useful information is to be received however, prediction errors cannot be avoided. Therefore, we are interested in the effect of bit prediction errors on the performance of the FSK system.

A direct calculation of the effects of bit prediction errors on the system performance would be a difficult task. All the different parameters involved in bit detection and prediction, reference loop design and adaptive array combine together to form a very complicated problem. Therefore we will use computer simulation results to study the effects of bit prediction errors. The system simulation program introduced in the previous chapter will be used to demonstrate the effects of several bit prediction error probabilities.

To see the results of a single prediction error, we modified the system simulation program so a prediction error would occur on the 14^{th} bit. A list of the program input data is given in Figure V-1. Except for the prediction error, the input data is the same as for the last example in Chapter IV. The FSK desired signal is incident

```
        ADAPTIVE ARRAY SIMULATION PROGRAM INPUT DATA

INCOMING SIGNALS:
     DESIRED FSK SIGNAL
        "0" BIT FREQUENCY=990 HERTZ
        "1" BIT FREQUENCY=1010 HERTZ
        POWER=0 DB
        ANGLE=0 DEGREES FROM BROADSIDE
        BIT RATE=10 BITS/SECOND
        MARKOV TRANSITION PROBABILITY=1.

     INTERFERENCE SIGNAL
        NONE

     NOISE
        POWER/ELEMENT=-10 DB

ADAPTIVE ARRAY
        ELEMENTS=2
        SEPARATION=0.5 DESIRED SIGNAL WAVELENGTHS
        CONVERGENCE FACTOR(K)=0.001

REFERENCE SIGNAL LOOP
     PREDICTION ERRORS
        ONE ERROR ON BIT NUMBER 15
     OSCILLATOR
        FREQUENCY FOR "0" BIT PREDICTION=1010 HERTZ
        FREQUENCY FOR "1" BIT PREDICTION=990 HERTZ
        OUTPUT AMPLITUDE=0 DB
     LOOP BANDPASS FILTER
        CENTER FREQUENCY=2000 HERTZ
        BANDWIDTH=10 HERTZ
        CENTER FREQUENCY GAIN=0 DB
     LIMITER
        CUTOFF VOLTAGE=0 DB
        GAIN BELOW CUTOFF=3 DB
```

Figure V-1. Simulation input data for a case containing a single bit prediction error.

from broadside and a CW interference signal is incident at 45° from broadside.

The program output for the case of one bit prediction error is shown in Figure V-2. Included with the weight curves and antenna pattern is a plot of the magnitudes of the desired and undesired parts of the array output. During the bit interval in which the prediction error occurs, the desired signal appears as interference to the adaptive array. Therefore, the weights change quickly in an attempt to null the desired signal. After the bit interval containing the error is over, the weights begin to return exponentially back to their steady state values. We note that the array output is amplitude modulated with an envelope similar to the weight curves as a result of the prediction error. The amount the weights and the array output are affected by a prediction error depends on the time constant of the adaptive array. If the array time constant is several bit intervals long, the array weights will not change significantly during a prediction error. Therefore, with respect to prediction errors, the time constant of the adaptive array should be made as long as possible. On the other hand, the array time constant must be short enough so that adaptation occurs in a reasonable period of time. Therefore, the choice of an array time constant is a compromise between good tolerance to prediction errors and rapid array response.

In the next set of simulation results, we allow prediction errors to occur with a probability of 0.1. The CW interference is still present on the 0 frequency and all other input parameters are the same as in the previous case. A complete list of the input information is given in Figure V-3 and the program output is given in Figure V-4.

We can see in Figure V-4 that bit prediction errors make the weights less smooth. The slope of each weight curve changes abruptly whenever a prediction error occurs. From the plot of the magnitudes of the array output components, we see that the bit prediction errors tend to amplitude modulate the received signal. FSK bit detection, however, is relatively unaffected by amplitude modulation because FSK bits are distinguished by frequency. The antenna pattern taken at the end of the simulation period shows that in spite of the prediction errors, the array is able to adapt properly. We see that the adaptive array has formed a null in the direction of the interference.

The following three sets of simulation results are for prediction error probabilities of 0.2, 0.3 and 0.4. Except for the prediction error probabilities, the input parameters are unchanged from the previous case. In Figure V-5 is a list of input data for the case with a 0.2 prediction error probability and the corresponding output is given in Figure V-6. The input data in Figure V-7 and the simulation results in Figure V-8 are for a 0.3 prediction error probability. Notice that as the prediction error probability increases, the amplitude distortion of the array output becomes greater. Also we note

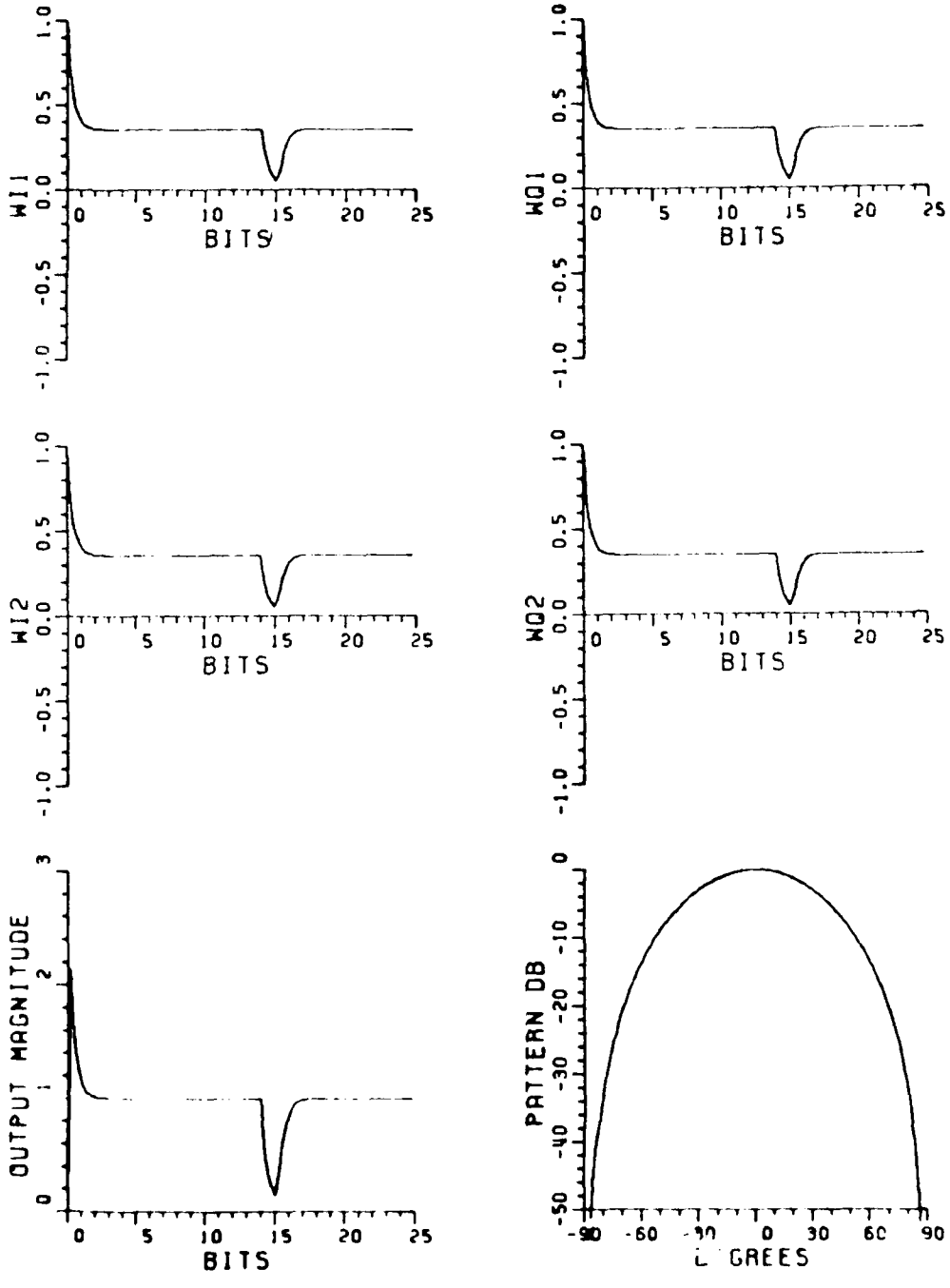


Figure V-2. Simulation results corresponding to the input data of Figure V-1.

```
        ADAPTIVE ARRAY SIMULATION PROGRAM INPUT DATA

INCOMING SIGNALS:
     DESIRED FSK SIGNAL
        "0" BIT FREQUENCY=990 HERTZ
        "1" BIT FREQUENCY=1010 HERTZ
        POWER=0 DB
        ANGLE=0 DEGREES FROM BROADSIDE
        BIT RATE=10 BITS/SECOND
        MARKOV TRANSITION PROBABILITY=.9

     INTERFERENCE SIGNAL
        FREQUENCY=990 HERTZ
        POWER=10 DB
        ANGLE=45 DEGREES FROM BROADSIDE

     NOISE
        POWER/ELEMENT=-10 DB

ADAPTIVE ARRAY
        ELEMENTS=2
        SEPARATION=0.5 DESIRED SIGNAL WAVELENGTHS
        CONVERGENCE FACTOR(K)=0.00005

REFERENCE SIGNAL LOOP
     PREDICTION ERRORS
        PREDICTION ERROR PROBABILITY=.1
     OSCILLATOR
        FREQUENCY FOR "0" BIT PREDICTION=1010 HERTZ
        FREQUENCY FOR "1" BIT PREDICTION=990 HERTZ
        OUTPUT AMPLITUDE=0 DB
     LOOP BANDPASS FILTER
        CENTER FREQUENCY=2000 HERTZ
        BANDWIDTH=10 HERTZ
        CENTER FREQUENCY GAIN=0 DB
     LIMITER
        CUTOFF VOLTAGE=0 DB
        GAIN BELOW CUTOFF=3 DB
```

Figure V-3. Simulation input data for a prediction error probability of 0.1.

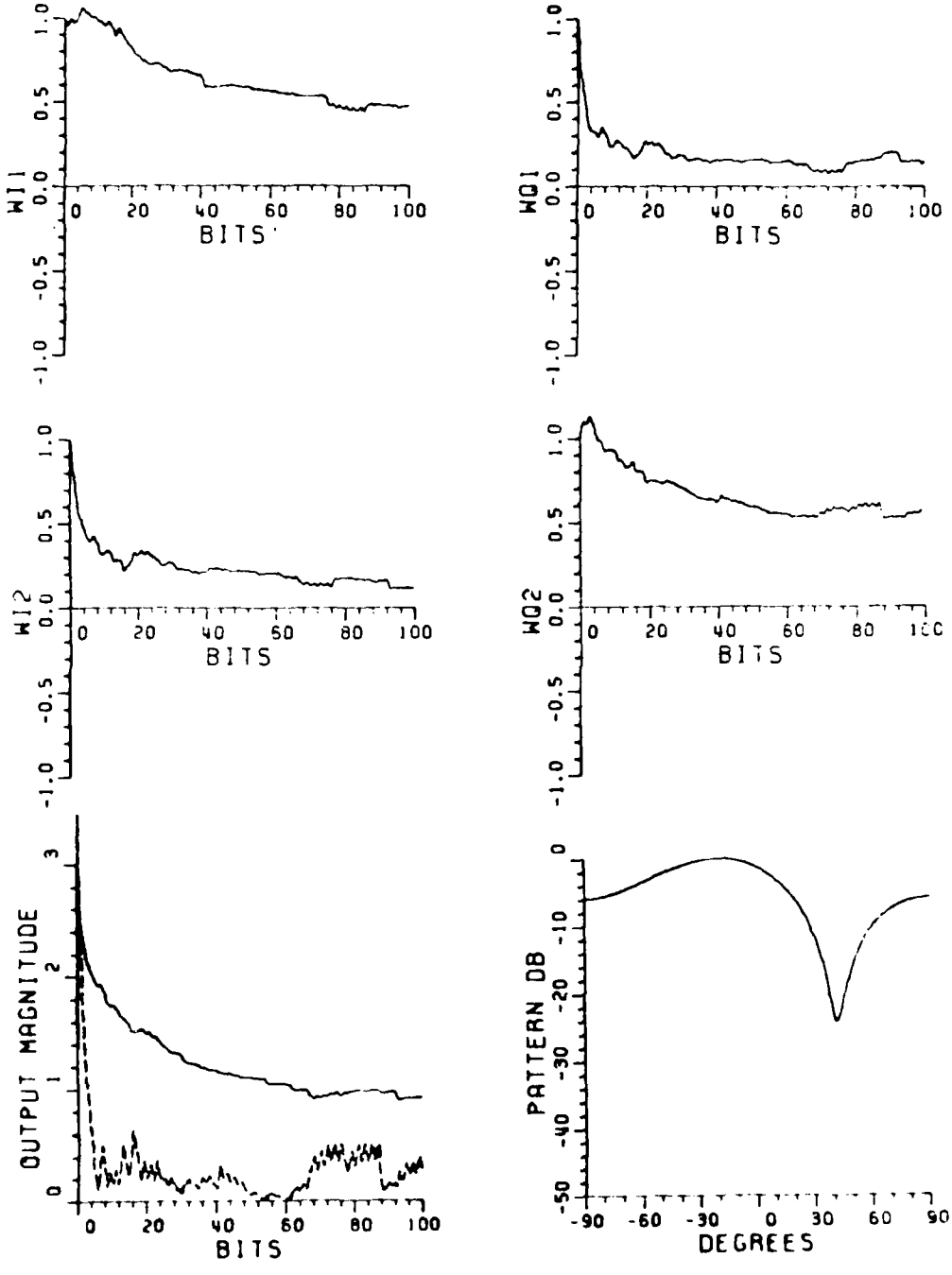


Figure V-4. Simulation results corresponding to the input data of Figure V-3.

```
        ADAPTIVE ARRAY SIMULATION PROGRAM INPUT DATA

INCOMING SIGNALS:
     DESIRED FSK SIGNAL
        "0" BIT FREQUENCY=990 HERTZ
        "1" BIT FREQUENCY=1010 HERTZ
        POWER=0 DB
        ANGLE=0 DEGREES FROM BROADSIDE
        BIT RATE=10 BITS/SECOND
        MARKOV TRANSITION PROBABILITY=.8

     INTERFERENCE SIGNAL
        FREQUENCY=990 HERTZ
        POWER=10 DB
        ANGLE=45 DEGREES FROM BROADSIDE

     NOISE
        POWER/ELEMENT=-10 DB

ADAPTIVE ARRAY
        ELEMENTS=2
        SEPARATION=0.5 DESIRED SIGNAL WAVELENGTHS
        CONVERGENCE FACTOR(K)=0.00005

REFERENCE SIGNAL LOOP
     PREDICTION ERRORS
        PREDICTION ERROR PROBABILITY=.2
     OSCILLATOR
        FREQUENCY FOR "0" BIT PREDICTION=1010 HERTZ
        FREQUENCY FOR "1" BIT PREDICTION=990 HERTZ
        OUTPUT AMPLITUDE=0 DB
     LOOP BANDPASS FILTER
        CENTER FREQUENCY=2000 HERTZ
        BANDWIDTH=10 HERTZ
        CENTER FREQUENCY GAIN=0 DB
     LIMITER
        CUTOFF VOLTAGE=0 DB
        GAIN BELOW CUTOFF=3 DB
```

Figure V-5. Simulation input data for a prediction error probability of 0.2.

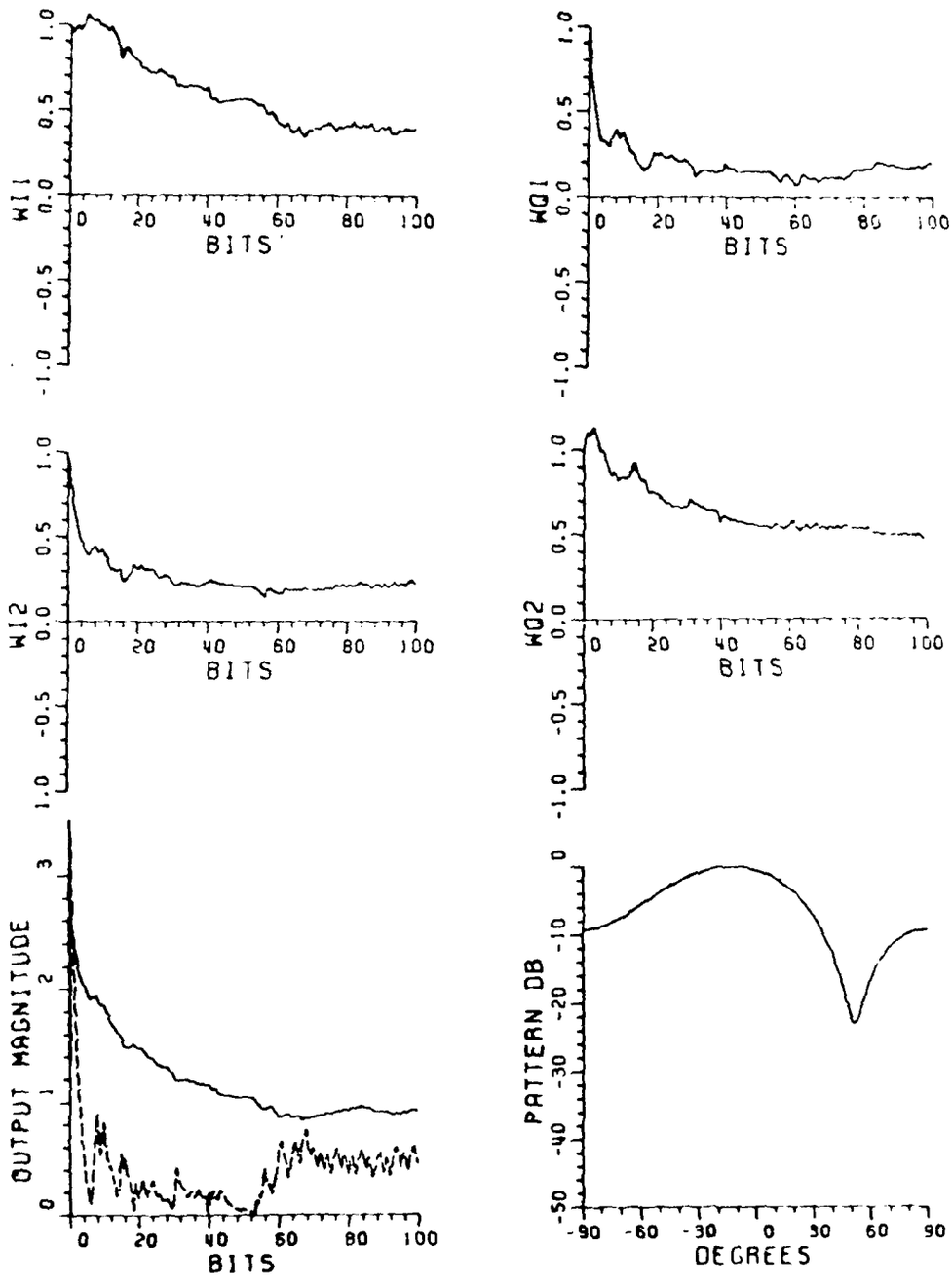


Figure V-6. Simulation results corresponding to the input data of Figure V-5.

```
        ADAPTIVE ARRAY SIMULATION PROGRAM INPUT DATA

INCOMING SIGNALS:
     DESIRED FSK SIGNAL
        "0" BIT FREQUENCY=990 HERTZ
        "1" BIT FREQUENCY=1010 HERTZ
        POWER=0 DB
        ANGLE=0 DEGREES FROM BROADSIDE
        BIT RATE=10 BITS/SECOND
        MARKOV TRANSITION PROBABILITY=.7

     INTERFERENCE SIGNAL
        FREQUENCY=990 HERTZ
        POWER=10 DB
        ANGLE=45 DEGREES FROM BROADSIDE

     NOISE
        POWER/ELEMENT=-10 DB

ADAPTIVE ARRAY
        ELEMENTS=2
        SEPARATION=0.5 DESIRED SIGNAL WAVELENGTHS
        CONVERGENCE FACTOR(K)=0.00005

REFERENCE SIGNAL LOOP
     PREDICTION ERRORS
        PREDICTION ERROR PROBABILITY=.3
     OSCILLATOR
        FREQUENCY FOR "0" BIT PREDICTION=1010 HERTZ
        FREQUENCY FOR "1" BIT PREDICTION=990 HERTZ
        OUTPUT AMPLITUDE=0 DB
     LOOP BANDPASS FILTER
        CENTER FREQUENCY=2000 HERTZ
        BANDWIDTH=10 HERTZ
        CENTER FREQUENCY GAIN=0 DB
     LIMITER
        CUTOFF VOLTAGE=0 DB
        GAIN BELOW CUTOFF=3 DB
```

Figure V-7. Simulation input data for the prediction error probability of 0.3.

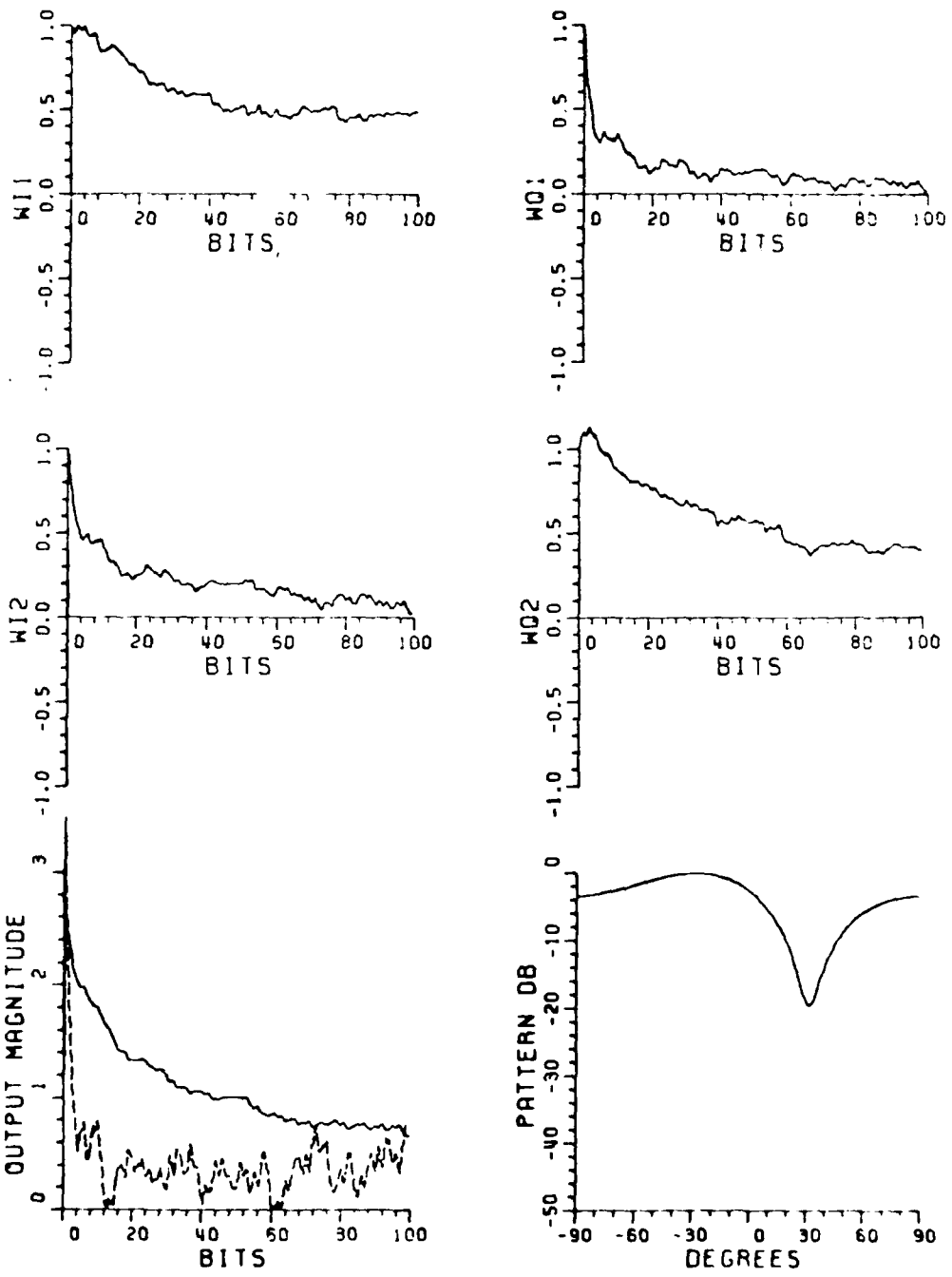


Figure V-8. Simulation results corresponding to the input data of Figure V-7.

```
        ADAPTIVE ARRAY SIMULATION PROGRAM INPUT DATA

INCOMING SIGNALS:
     DESIRED FSK SIGNAL
        "0" BIT FREQUENCY=990 HERTZ
        "1" BIT FREQUENCY=1010 HERTZ
        POWER=0 DB
        ANGLE=0 DEGREES FROM BROADSIDE
        BIT RATE=10 BITS/SECOND
        MARKOV TRANSITION PROBABILITY=.6

     INTERFERENCE SIGNAL
        FREQUENCY=990 HERTZ
        POWER=10 DB
        ANGLE=45 DEGREES FROM BROADSIDE

     NOISE
        POWER/ELEMENT=-10 DB

ADAPTIVE ARRAY
        ELEMENTS=2
        SEPARATION=0.5 DESIRED SIGNAL WAVELENGTHS
        CONVERGENCE FACTOR(K)=0.00005

REFERENCE SIGNAL LOOP
     PREDICTION ERRORS
        PREDICTION ERROR PROBABILITY=.4
     OSCILLATOR
        FREQUENCY FOR "0" BIT PREDICTION=1010 HERTZ
        FREQUENCY FOR "1" BIT PREDICTION=990 HERTZ
        OUTPUT AMPLITUDE=0 DB
     LOOP BANDPASS FILTER
        CENTER FREQUENCY=2000 HERTZ
        BANDWIDTH=10 HERTZ
        CENTER FREQUENCY GAIN=0 DB
     LIMITER
        CUTOFF VOLTAGE=0 DB
        GAIN BELOW CUTOFF=3 DB
```

Figure V-9. Simulation input data for a prediction error probability of 0.4.

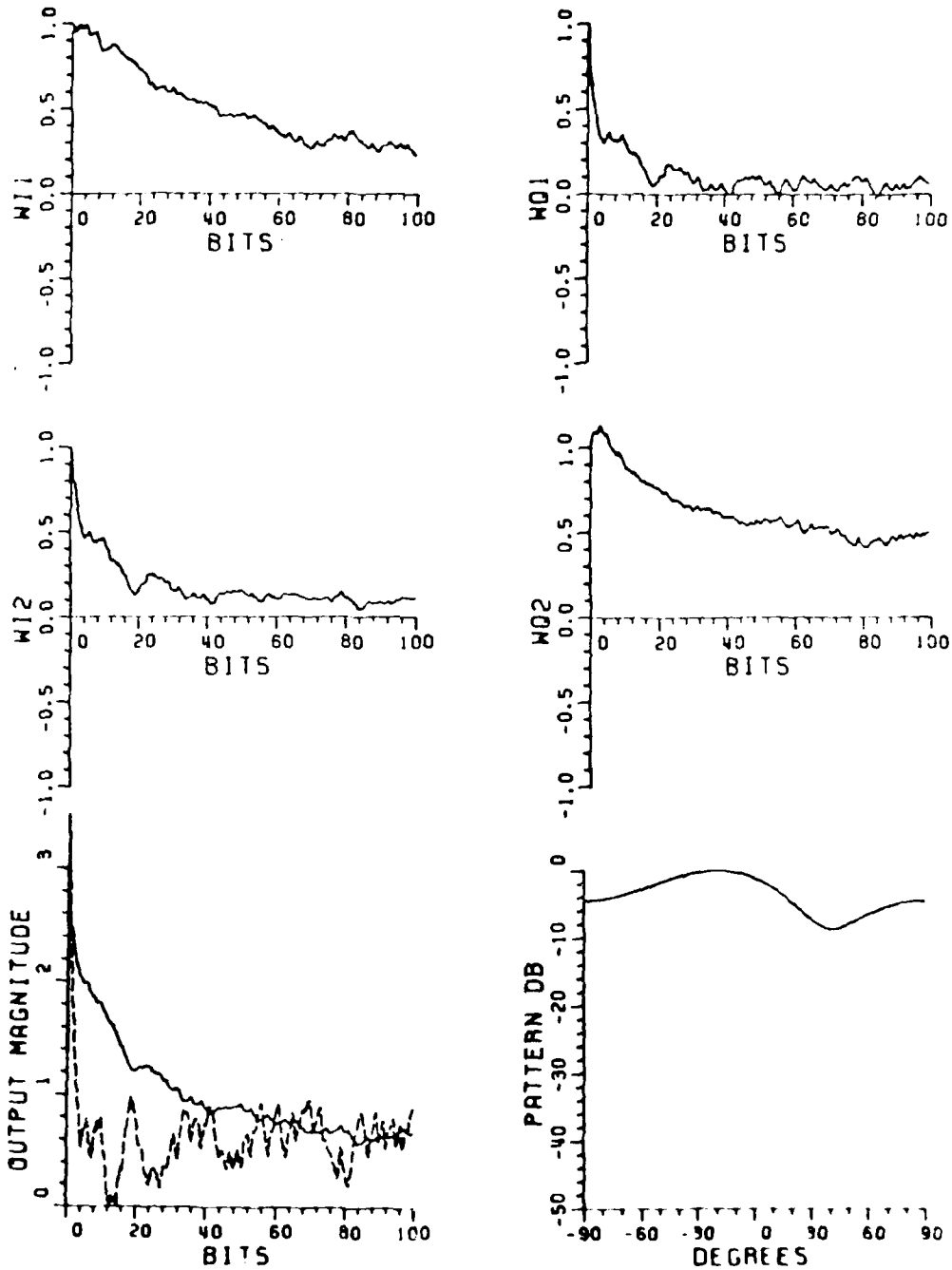


Figure V-10. Simulation results corresponding to the input data of Figure V-9.

that with the increase in prediction errors, the magnitude of the interference at the array output increases. In the input data of Figure V-9, the prediction error probability is set to 0.4. The corresponding output, in Figure V-10, shows that the null in the antenna pattern is steered about halfway between the desired and interference signals. With such a high prediction error probability, the array is no longer able to distinguish the desired signal from the interference. Therefore, the null in the antenna pattern is not steered directly toward the interference.

Looking at the array output curves in Figures V-3 and V-5, we see that the average interference at the array output decreases with time throughout the simulation period. In Figures V-7 and V-9, for higher prediction error probabilities, the average value of the interference remains just below the average value of the desired signal. Thus we conclude that for this particular simulation of the system, the prediction error should be made 0.2 or less. This result, of course, applies only to the particular set of system parameters chosen for the simulations.

The bit prediction error rate is an important parameter in the design of the FSK system. As explained in Chapter III, the prediction error rate determines the bandwidth spreading factor for the FSK signal transmitted to the adaptive array. The prediction error probability becomes smaller as the bandwidth of the FSK signal increases. Therefore, the choice of the bit prediction error probability is a trade-off between good adaptive array performance and bandwidth use.

C. Reference Signal Phase Error

Reference signal phase error can be a problem in adaptive arrays that form the reference signal from the array output. By phase error, we mean a phase shift between the array output and the reference signal. The phase shift around the reference signal loop is a function of frequency. Therefore, Doppler shift of the received signal or oscillator frequency drift can cause the reference signal to have a phase error.

It was shown by DiCarlo and Compton[13] that in a one or two element adaptive array, a reference signal phase error causes the weights to cycle. A subsequent study by DiCarlo[14] extended the results to include an adaptive array with any number of elements. The study showed that without interference, the cyclic weights cause no change in the signal to noise ratio at the array output.

In this report, we have been studying the performance of a two-element FSK system in the presence of CN interference. We are therefore interested in the effect of a reference signal phase error on the output SINR in a two element adaptive array.

A model of the reference signal phase shift problem is shown in Figure V-11. We are using the same notation introduced in Chapter II. In the model, an ideal reference signal processor removes the desired signal component from the array output and produces a reference signal

$$r(t) = A_r \begin{bmatrix} s_d(t) \\ \overline{s_d(t)} \end{bmatrix} \quad . \tag{V-1}$$

Also in the model is a phase shifter which introduces a phase error, θ_e, in the reference signal

$$r(t) = A_r \begin{bmatrix} s_d(t) \\ \overline{s_d(t)} \end{bmatrix} e^{j\theta_e} \quad . \tag{V-2}$$

Using the equality,

$$s_d(t) = X_d{}^T W = W^T X_d \quad , \tag{V-3}$$

we find

$$r(t) = \frac{A_r X_d{}^T W e^{j\theta_e}}{\sqrt{(W^T X_d^*)^* X_d{}^T W}} \quad , \tag{V-4}$$

which becomes

$$r(t) = \frac{A_r X_d{}^T W e^{j\theta_e}}{\sqrt{W^{*T} \Phi_d W}} \quad . \tag{V-5}$$

The reference signal correlation vector is given by

$$S = E\{X^* R(t)\} = \frac{A_r \Phi_d W e^{j\theta_e}}{\sqrt{W^{*T} \Phi_d W}} \quad . \tag{V-6}$$

The differential equation for the LMS array, given by Equation (II-45) in Chapter II, becomes

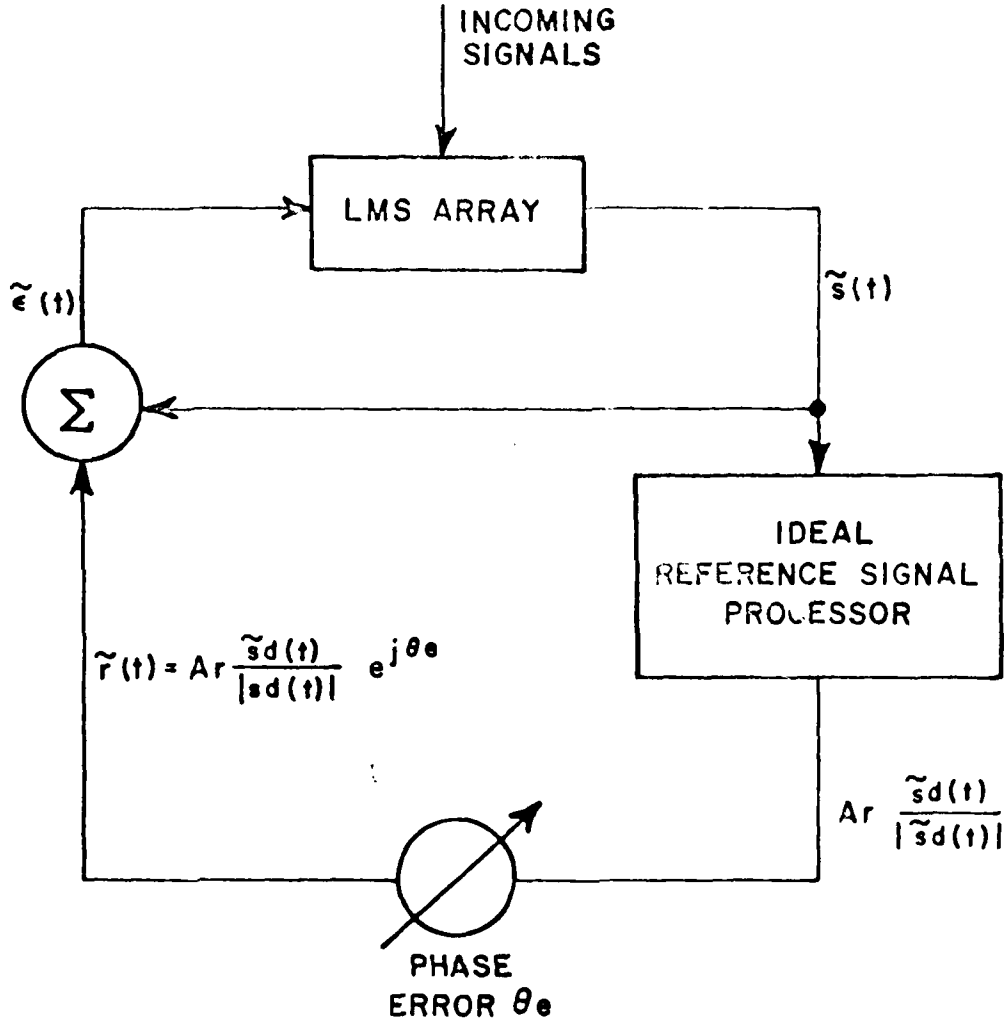


Figure V-11. Mathematical model used for computing the effects of reference signal loop phase error.

$$\frac{dW}{dt} + k\phi w = \frac{Ar\ \Phi_d\ W\ e^{j\theta_e}}{\sqrt{W^{*T}\ \Phi_d\ W}} \quad . \qquad \text{(V-7)}$$

To get the SINR at the array output, we will find the steady state solution to the above equation for the two element case.

Again using notation introduced in Chapter II, we calculate the covariance matrices for the two element array,

$$\Phi_d = A_d^2 \begin{pmatrix} 1 & e^{-j\phi_d} \\ e^{j\phi_d} & 1 \end{pmatrix} \quad , \qquad \text{(V-8)}$$

$$\Phi_i = A_i^2 \begin{pmatrix} 1 & e^{-j\phi_i} \\ e^{j\phi_i} & 1 \end{pmatrix} \quad , \qquad \text{(V-9)}$$

and

$$\Phi_n = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix} \quad . \qquad \text{(V-10)}$$

The desired signal covariance matrix, Φ_d, has one non-zero eigenvalue given by

$$\lambda_d = 2A_d^2 \quad . \qquad \text{(V-11)}$$

The diagonal matrix of eigenvalues for Φ_d is defined

$$\Lambda_d = \begin{pmatrix} 0 & 0 \\ 0 & 2A_d^2 \end{pmatrix} \quad . \qquad \text{(V-12)}$$

We also define a coordinate rotation matrix, R_d, such that

$$R_d^{*T}\,\Phi_d\,R_d = \Lambda_d \qquad . \tag{V-13}$$

For the two element adaptive array, we find

$$R_d = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 & e^{-j\phi_d} \\ e^{j\phi_d} & 1 \end{pmatrix} \qquad . \tag{V-14}$$

A new set of rotated weights,

$$V = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \qquad , \tag{V-15}$$

are defined by the equation,

$$W = R_d V \qquad . \tag{V-16}$$

Substituting the above definition into Equation (V-7) gives

$$R_d \frac{dV}{dt} + k[\phi_d+\phi_i+\phi_n]R_d V = \frac{k\,A_r \Phi_d R_d V\, e^{j\theta_e}}{\sqrt{V^{*T}R_d^{*T}\Phi_d R_d V}} \qquad . \tag{V-17}$$

After some manipulation, we have

$$\frac{dV}{dt} = \frac{kA_r R_d^{*T}\Phi_d R_d V\, e^{j\theta_e}}{\sqrt{V^{*T}R_d^{*T}\Phi_d R_d V}} - k\,R_d^{*T}\Phi_d R_d V - k\,R_d^{*T}\Phi_i R_d V - k\Phi_n V \qquad . \tag{V-18}$$

Equation (V-18) can be expanded further by using Equations (V-12) and (V-13) into

$$\frac{d}{dt}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \sqrt{2}\,kA_d A_r \begin{pmatrix} 0 \\ v_2 \\ |v_2| \end{pmatrix} e^{j\theta_e} - 2kA_d^2 \begin{pmatrix} 0 \\ v_2 \end{pmatrix} - k[\sigma^2 I + R_d^{*T}\Phi_i R_d] \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \tag{V-19}$$

Using Equations (V-9) and (V-14), we can expand the term

$$R_d^{*T}{}_{\phi_i} R_d = A_i^2 \begin{bmatrix} 1-\cos\phi & -j\sin\phi \\ j\sin\phi & 1+\cos\phi \end{bmatrix} , \qquad \text{(V-20)}$$

where

$$\phi = \phi_d - \phi_i \quad . \qquad \text{(V-21)}$$

Now, substituting Equation (V-20) into Equation (V-19), we find a pair of differential equations for the rotated weights

$$\frac{dv_1}{dt} = -k\sigma^2 v_1 - kA_i^2(1-\cos\phi)v_1 + jkA_i^2 \sin\phi v_2 \qquad \text{(V-22)}$$

$$\frac{dv_2}{dt} = \sqrt{2}\, kA_d A_r \frac{v_2}{|v_2|} e^{j\theta} e - 2k A_d^2 v_2 - k\sigma^2 v_2$$

$$-jkA_i^2 \sin\phi v_1 - kA_i^2(1+\cos\phi)v_2 \quad . \qquad \text{(V-23)}$$

At this point, we will assume a sinusoidal steady-state solution for Equations (V-22) and (V-23). The assumed solution takes the form

$$V = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} V_1 e^{j\gamma} \\ V_2 \end{pmatrix} e^{j\omega_c t} \quad , \qquad \text{(V-24)}$$

where V_1 and V_2 are the magnitudes of the cyclic weights, γ is the phase angle between the two weights, and ω_c is the weight cycling frequency.

Substituting the assumed form of the solution into Equations (V-22) and (V-23), we obtain the pair of equations

$$j\omega_c V_1 e^{j\gamma} = -k(\sigma^2 + A_i^2 - A_i^2\cos\phi)V_1 e^{j\gamma} + jkA_i^2 \sin\phi V_2 \qquad \text{(V-25)}$$

$$j\omega_c V_2 = \sqrt{2}\, kA_d A_r\, e^{j\theta_e} - k(\sigma^2+2A_d^2+A_i^2+A_i^2\cos\phi)V_2$$

$$-j\, kA_i^2 \sin\phi V_1 e^{j\gamma} \quad . \qquad \text{(V-26)}$$

Each equation above can be divided into real and imaginary parts and the following four equations result:

$$\omega_c V_1 \cos\gamma = -k(\sigma^2+A_i^2-A_i^2\cos\phi)V_1\sin\gamma + kA_i^2\sin\phi V_2 \qquad \text{(V-27)}$$

$$-\omega_c V_1 \sin\gamma = -k(\sigma^2+A_i^2-A_i^2\cos\phi)V_1\cos\gamma \qquad \text{(V-28)}$$

$$\omega_c V_2 = \sqrt{2}\, k\, A_d A_r \sin\theta_e - kA_i^2\sin\phi V_1\cos\gamma \qquad \text{(V-29)}$$

$$0 = \sqrt{2}\, kA_d A_r \cos\theta_e - k(\sigma^2+2A_d^2+A_i^2+A_i^2\cos\phi)V_2$$

$$+ kA_i^2\sin\phi V_1\sin\gamma \quad . \qquad \text{(V-30)}$$

In the following calculations, we will solve the four equations above by first solving for the cycling frequency, ω_c. Then we will be able to write the other three variables, V_1, V_2, and γ, in terms of ω_c.

We begin with Equation (V-29) and obtain the relation

$$\tan\gamma = \frac{k(\sigma^2+A_i^2-A_i^2\cos\phi)}{\omega}\cos\gamma \quad . \qquad \text{(V-31)}$$

Now using Equation (V-28) and Equation (V-31), we find

$$\cos\gamma = \frac{k\omega A_i^2\sin\phi}{\omega^2+k^2(\sigma^2+A_i^2-A_i^2\cos\phi)^2}\,\frac{V_2}{V_1} \quad . \qquad \text{(V-32)}$$

Substituting Equation (V-32) into Equation (V-30), and solving for V_2 yields

$$V_2 = \frac{\sqrt{2}\ kA_dA_r\sin\theta_e[\omega_c^2+k^2(\sigma^2+A_i^2-A_i^2\cos\phi)^2]}{\omega_c[\omega_c^2+k^2(\sigma^2+A_i^2-A_i^2\cos\phi)^2+k^2A_i^4\sin^2\phi]} \quad . \qquad (V\text{-}33)$$

We substitute Equations (V-31) and (V-32) into Equation V-30), again solve for V_2 obtaining

$$V_2 = \frac{\sqrt{2}\ kA_dA_r\cos\theta_e(\omega^2+k^2(\sigma^2+A_i^2-A_i^2\cos\phi)^2)}{[\omega^2+k^2(\sigma^2+A_i^2-A_i^2\cos\phi)^2][\sigma^2+2A_d^2+A_i^2\cos\phi-kA_i^4\sin^2\phi}$$

$$(\sigma^2+a_i^2-A_i^2\cos:)] \quad . \qquad (V\text{-}34)$$

Equating Equation (V-33) and Equation (V-34) gives us an expression for ω_c,

$$\omega_c^3-\omega_c^2\ k\tan\theta_e(\sigma^2+2A_d^2+A_i^2+A_i^2\cos\phi)$$

$$+\ \omega_c\{k^2(\sigma^2+A_i^2-A_i^2\cos\phi)^2 + k^2A_i^4\sin^2\phi\}$$

$$-\ k^3\tan\theta_e(\sigma^2+A_i^2-A_i^2\cos\phi)\{(\sigma^2+2A_d^2+A_i^2+A_i^2\cos\phi)$$

$$(\sigma^2+A_i^2-A_i^2\cos\phi) - A_i^4\sin^2\phi\} = 0 \quad . \qquad (V\text{-}35)$$

Now by solving Equation (V-35) above for the cycling frequency, we can use Equation (V-31) to get γ, Equation (V-33) to find V_2 and Equation V-32) to solve for V_1. The normalized weights then are substituted into Equation (V-16) to obtain the unrotated complex weight vector W.

Equation (V-35) is a cubic with real coefficients. Therefore it will either have one real root and a conjugate pair of complex roots or three real roots. In the case of three real roots, the weights will oscillate in one of three different frequencies or modes The mode in which the weights fall into depends upon the initial conditions.

Equation (V-35) was solved for several values of interference power and the results are shown in Figure V-12. The curves in the

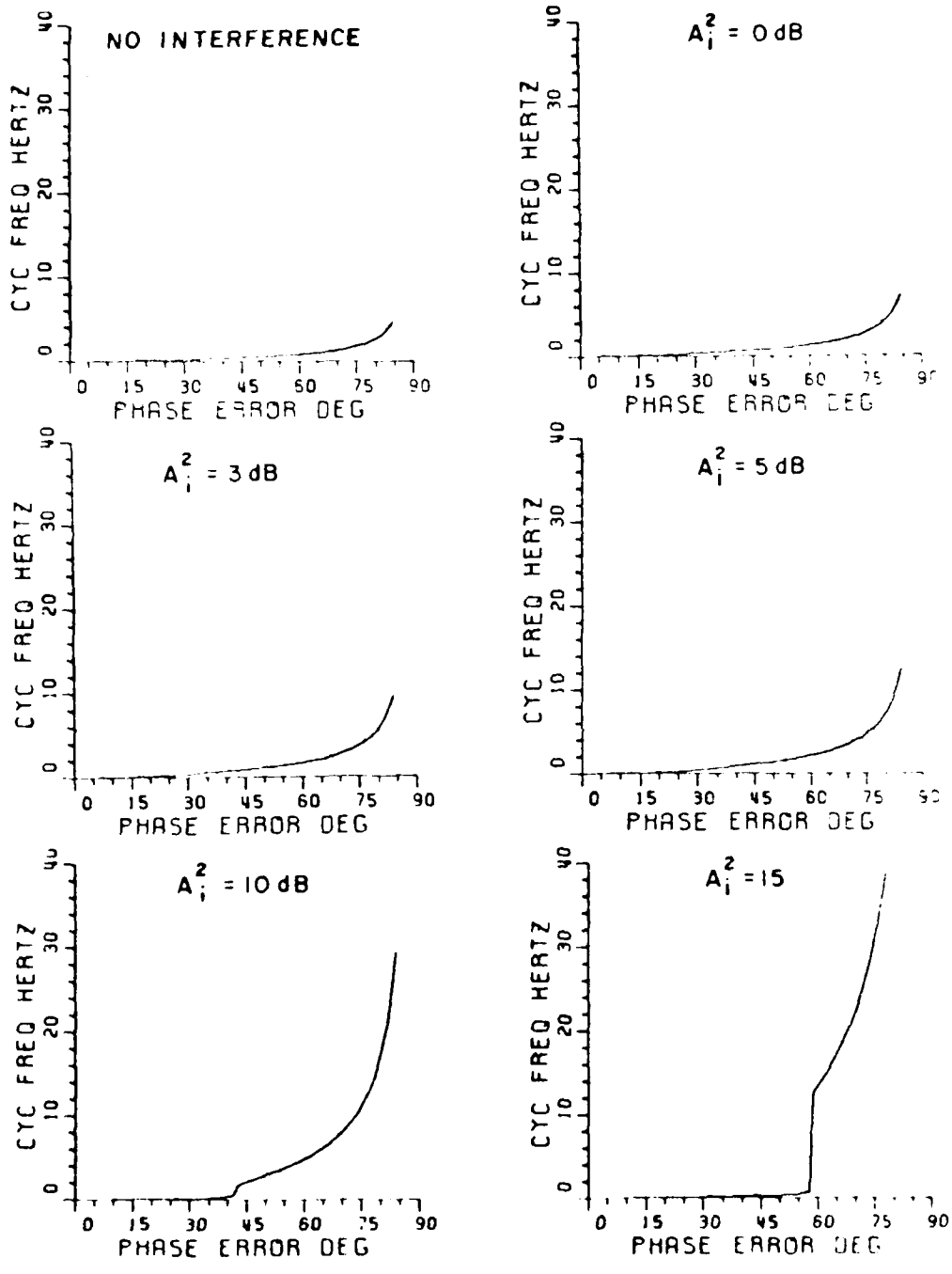


Figure V-12. Cycling frequencies for various levels of interference power. For each curve, A_d^2 = 0 dB and σ^2 = 0 dB.

figure were drawn by computing the real roots of the cubic equation. For the points where the equation has three real roots, the root with the highest absolute value was chosen. At the origin, each curve has only one real root. The case of three real roots does not occur for any value of phase error when $A_j^2=5$ or less. In the other two curves however, as the phase error is increased, a point is reached where the cubic equation has three real roots. This point appears in the curves as a slope discontinuity. The discontinuity appears when three real roots occur because one of the three roots has larger magnitude than the single root at the previous point. Since we are plotting the largest root, a slope discontinuity occurs in the curve.

We will now study in more detail a case where the weights have three cycling frequencies. From Figure V-12, we see that for $A_j^2=10$, a slope discontinuity occurs near a phase error of 40°. We choose the point, $\theta_e=41.65°$, and calculate the three weight cycling frequencies from Equation (V-35). For each cycling frequency, we can find a set of steady state complex weights using Equations (V-31), (V-32) and (V-33). A list of the assumptions, the cycling frequencies, and the weight calculations are shown in Figure V-13.

Using a fourth-order Runge-Kutta algorithm[15], we get a numerical solution for the array weights from Equation (V-7). For the case outlined in Figure V-13, we calculate the initial weights, at t=0, to use for starting points for the Runge-Kutta algorithm. Each of the three sets of initial conditions results in weights with different cycling frequencies. The weight curves for the lowest cycling frequency are shown in Figure V-14. Using the initial conditions corresponding to the second cycling frequency, we obtain the results of Figure V-15. Finally, the solution for the highest cycling frequency is illustrated in Figure V-16.

The relationship between the reference signal phase error and the array output SINR is examined in Figure V-17. The curves were drawn by first computing the cycling frequencies and weights for each value of phase error. Then the components of the array output due to desired signal, interference, and noise were calculated. For the cases where three cycling frequencies occurred, the frequency resulting in minimum SINR was chosen. In Figure V-17, we see the array output SINR in decibels for several values of interference power. For the case of no interference, we see that the SINR remains constant with increasing phase error. When interference is added, however, an increase in phase error causes a reduction in SINR. As more interference power is added, the reduction in SINR becomes more pronounced. In the last two curves of Figure V-17, there is an almost vertical drop in SINR as the phase error is increased past a certain point. This SINR drop occurs at the point where the solution to the equation for the weight cycling frequency goes from one real root to three real roots. These curves show that for high interference levels, a large

Mode 1. $\omega_{c_1} = 2.865$ rad/sec

$V_1 = 0.1554$

$V_2 = 0.1116$

$\gamma = 59.903°$

$W_1 = (0.09801\text{-}j0.09508)\, e^{j\omega_{c_1}t}$

$W_2 = (0.0238\text{-}j0.09508)\, e^{j\omega_{c_1}t}$

Mode 2. $\omega_{c_2} = 5.560$ rad/sec

$V_1 = 0.08433$

$V_2 = 0.07885$

$\gamma = 41.460°$

$W_1 = (0.01119\text{-}j0.03962)\, e^{j\omega_{c_2}t}$

$W_2 = (-0.10032\text{-}j0.03962)\, e^{j\omega_{c_2}t}$

Mode 3. $\omega_{c_3} = 8.527$ rad/sec

$V_1 = 0.05388$

$V_2 = 0.06674$

$\gamma = 30.107°$

$W_1 = (0.01423\text{-}j0.01911)\, e^{j\omega_{c_3}t}$

$W_2 = (-0.08015\text{-}j0.01911)\, e^{j\omega_{c_3}t}$

Figure V-13. Data and results for a three-mode example.

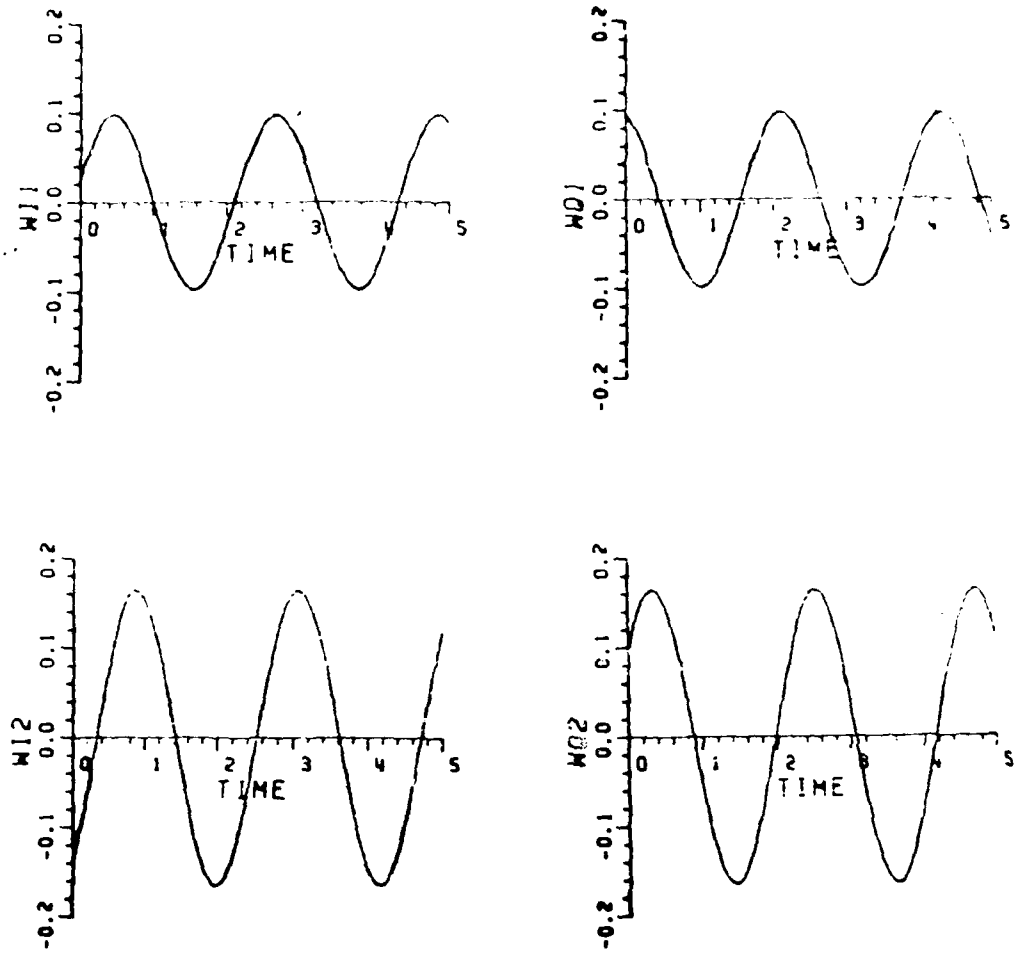


Figure V-14. Weight curves corresponding to the lowest cycling frequency, ω_c=2.865 rad/sec.

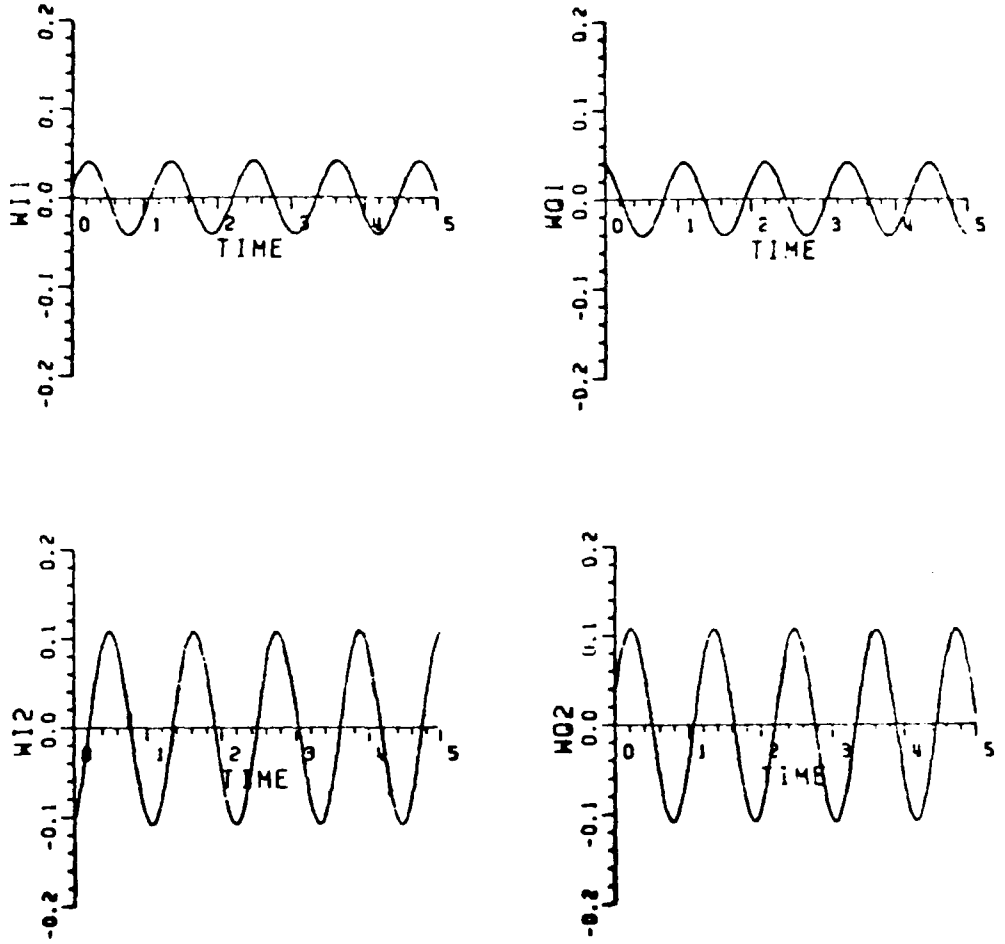


Figure V-15. Weight curves corresponding to a cycling frequency of ω_c=5.559 rad/sec.

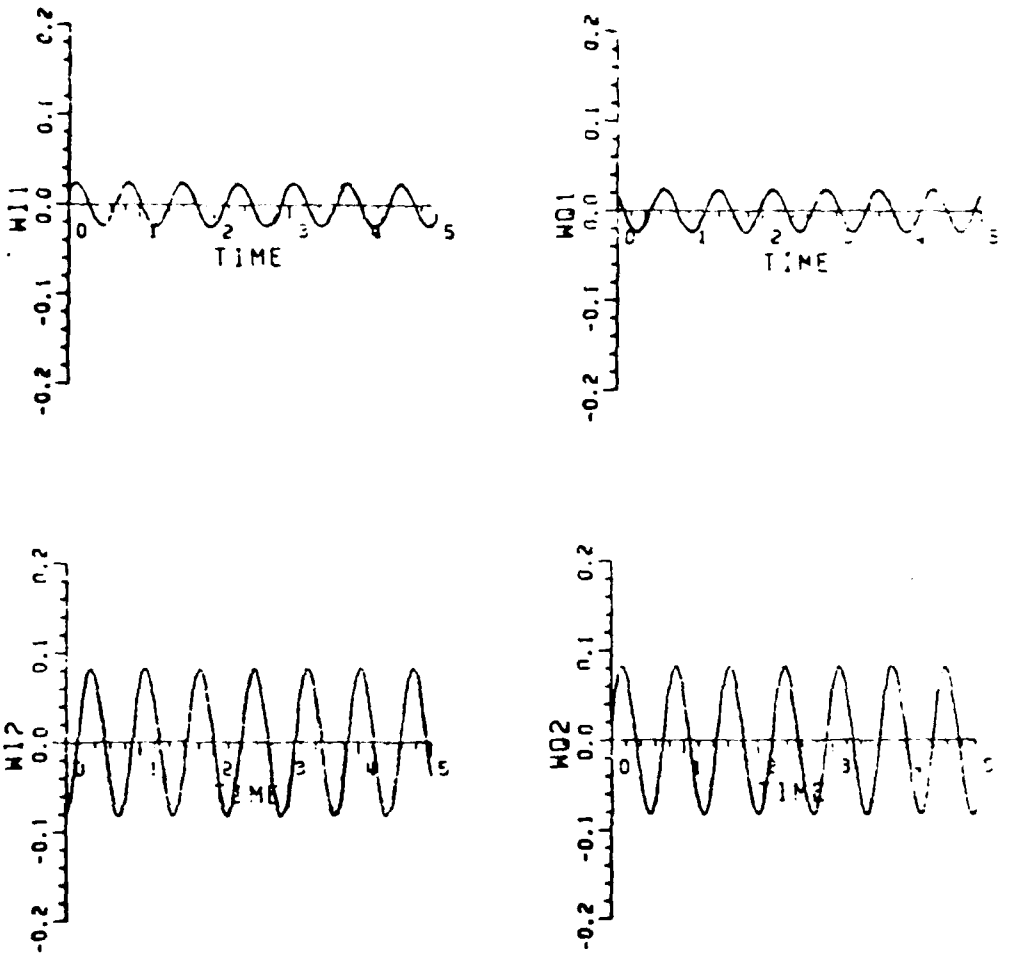


Figure V-16. Weight curves for a cycling frequency, ω_c=8.5247 rad/sec.

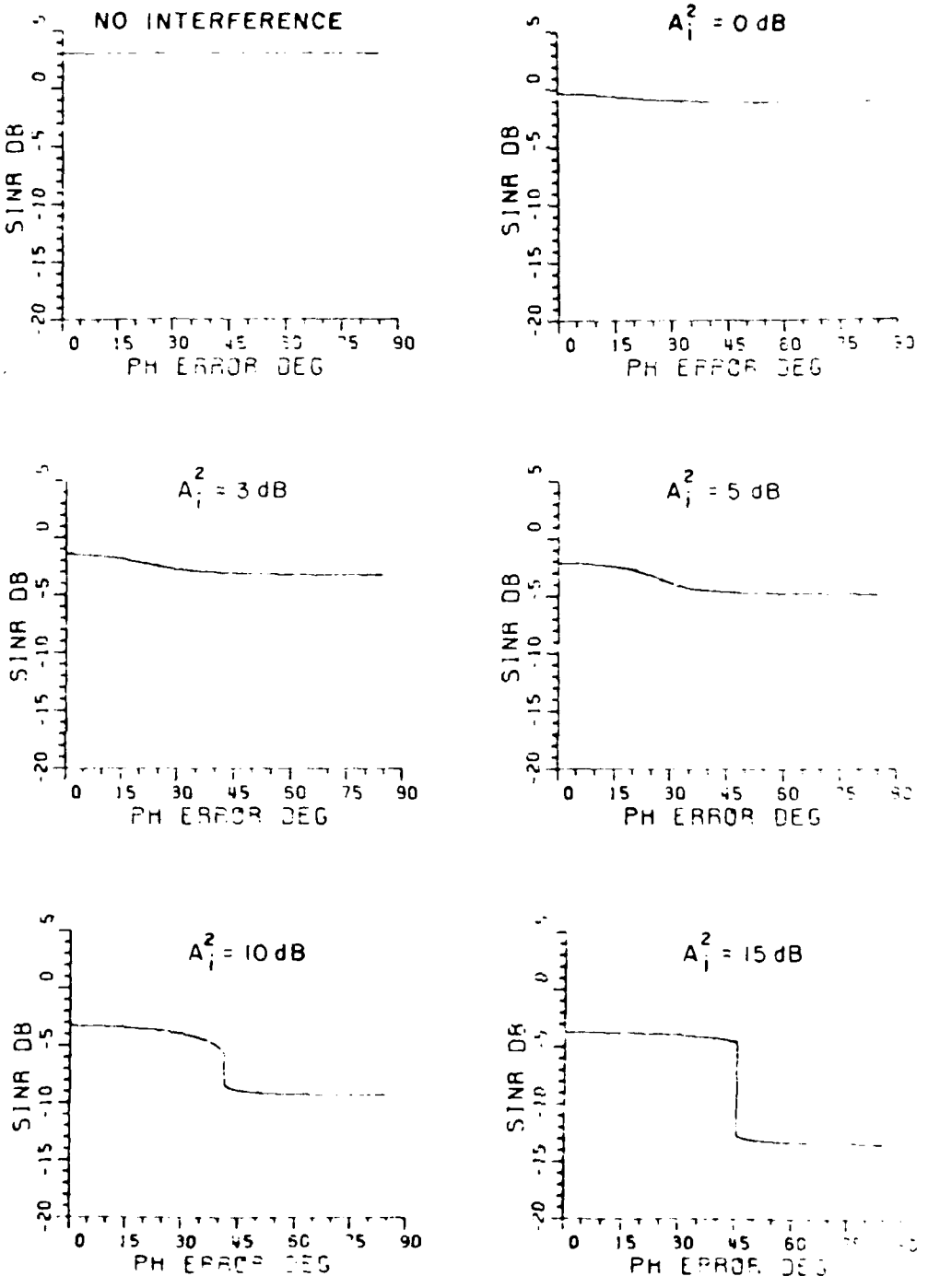


Figure V-17. SINR versus reference signal phase error for several interference levels. In each case, A_d^2=0 dB and σ^2=0 dB.

reference signal phase error can cause a serious reduction in the performance of the adaptive array.

We have seen, in this section the effects of a reference signal phase error in the adaptive array. First, we showed that a phase error causes the steady state array weights to cycle. Under some conditions there is only one cycling frequency but sometimes Equation (V-35) gives three real cycling frequencies. When no interference is present, we showed that the phase error has no effect on SINR. However, we showed that in the presence of interference, the phase error a significant reduction in SINR. Therefore we conclude that in designing the reference signal loop, it is important to keep the phase error as small as possible.

D. Signal Acquisition and Synchronization

We assumed in our discussion of the reference signal loop in Chapter IV that bit predictions were available during the array lock-up period. We also assumed that the bit predictions were synchronized with the incoming desired signal bits. These two assumptions imply that we can detect the desired signal and extract timing information during the array lock-up period. In this section we will show how the array responds in the early part of the acquisition period, before the bit predictions and the reference signal are available. It will be shown that even without the reference signal present, the array maintains a good SINR during the initial lock-up period.

The reference signal loop requires properly timed bit predictions to produce a reference signal that correlates with the desired signal. During the early part of lock-up, the interference power at the array output may exceed the desired signal power. Therefore, it may not be possible to detect the desired signal bits and make bit predictions. When the bit predictions are not available, we will assume that the output of the reference signal loop does not correlate with the desired signal. Therefore the reference signal correlation vector becomes

$$S = E\{X(t)r(t)\} = 0 \quad . \tag{V-36}$$

When the reference signal does not correlate with the desired signal, the differential equation for the LMS array is

$$\frac{dW}{dt} = -k\,\Phi W \quad . \tag{V-37}$$

Figure V-18 shows curves of SINR, output desired signal power (P_d), and output interference power (P_i) for the period where no bit predictions are available. The curves were obtained by using a Runge-Kutta algorithm to solve Equation (V-37) for the array weights. The weights were then used to compute the signal powers at the array output.

We see in Figure V-18 that the adaptive array does power inversion when no reference signal is present. By power inversion, we mean that after the initial transient,

$$\frac{P_d}{P_i} = \frac{A_i^2}{A_d^2} \quad . \tag{V-38}$$

Therefore, when the incident interference power is much greater than the incident desired signal power, the array will quickly give a good SINR at the array output. The absolute signal levels however decrease with time. Therefore acquisition of the signal and timing must take place after a good SINR is established and before the signals become too small to be detected.

The FSK detector used in the system described here is assumed to be noncoherent. The problem of bit synchronization for noncoherent SFK has been studied by Resso [16]. We will not go into the problem of bit timing in this report. The reader is referred to Resso for details.

We have seen that in the presence of large interference levels, the power inversion property of the LMS array gives us a time period in which to acquire the desired signal and timing. We will assume that this time period is long enough that bit detection and synchronization will not be a serious problem.

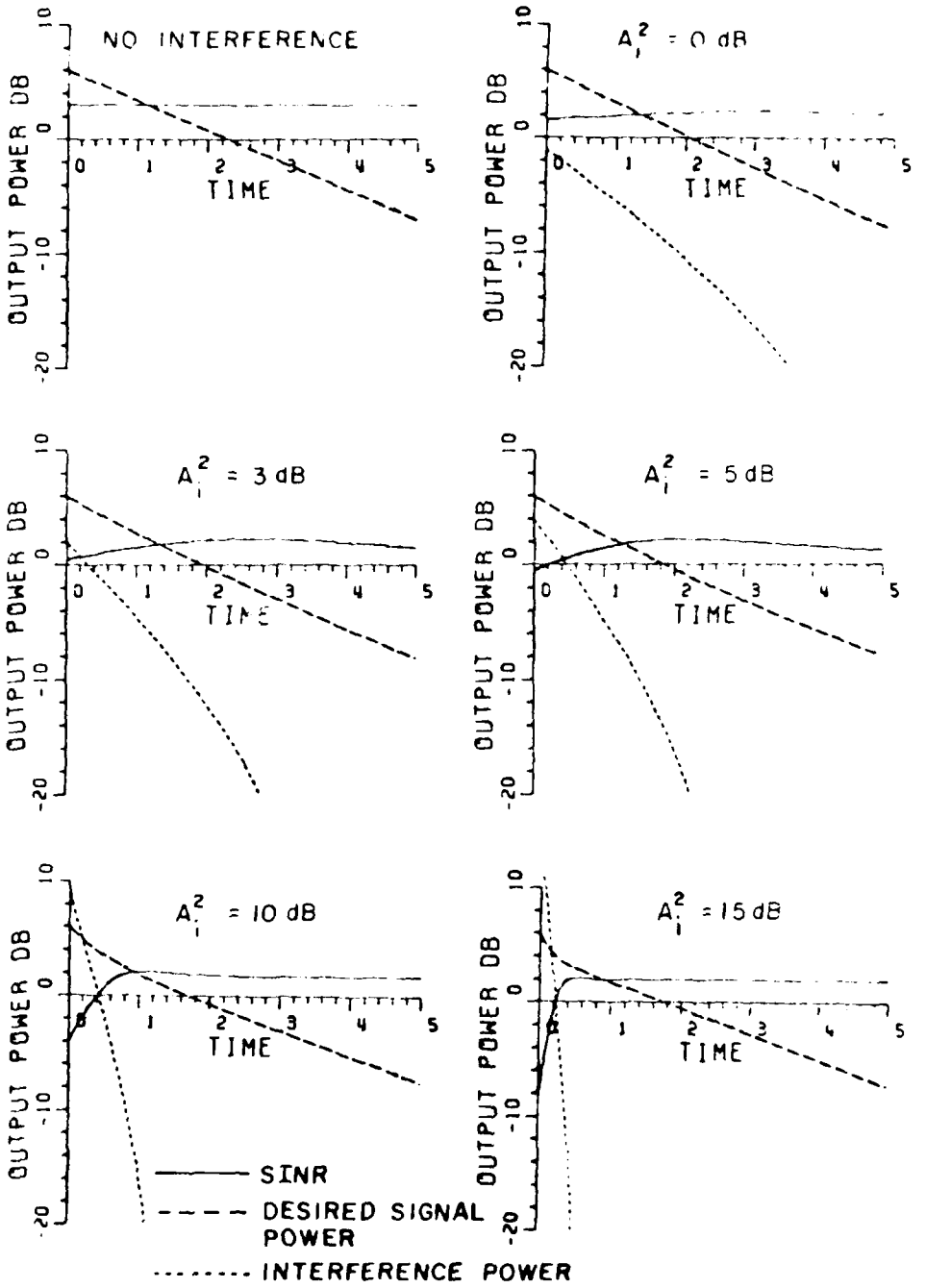


Figure V-18. SINR, desired signal power and interference power at the array output during lock-up for A_d^2=0 dB and σ^2=0 dB.

CHAPTER IV
CONCLUSION

We have examined the use of an adaptive array in an FSK communications system. The adaptive array is used to protect the communications system from interference. Interference protection is especially valuable in military communications for example, where deliberate jamming may be a factor.

The FSK system presented in this report uses an LMS adaptive array. As explained in Chapter II, the LMS array minimizes the mean square error between the array output and a reference signal. Therefore, to use the LMS array in an FSK system, we must generate an FSK reference signal.

We described a method for forming an FSK reference signal based on the use of bit predictions. In Chapter III we showed that by making the desired bit stream first order Markov, bit predictions can be made at the FSK receiver.

In Chapter IV, we described a reference signal loop which uses the array output and the bit predictions to form a reference signal. When the reference signal loop operates properly, the desired signal component of the array output is fed back to the adaptive array as the reference signal.

Several possible problems associated with the FSK system were discussed in Chapter IV. We first looked at the effects of bit prediction errors. Then we considered the problems of reference signal phase error and the resulting cyclic weights. Finally we looked at signal acquisition and bit synchronization during lock-up.

In conclusion, we have presented an adaptive array system for use in FSK communications. The system operation was demonstrated with the use of a computer simulation program. We found that within the limits discussed in this report, the FSK system developed here is a viable system for use in FSK communications.

REFERENCES

1. B. Widrow, P.E. Mantey, L.J. Griffiths and B.B. Goode, "Adaptive Antenna Systems," Proc. IEEE, 55, 12 (December 1967), 2143.

2. S.R. Applebaum, "Adaptive Arrays," IEEE Trans., AP-24, 5 (September 1976), 585.

3. S.W. Shor, "Adaptive Technique to Discriminate Against Coherent Noise in a Narrow-Band System," Journal of the Acoustical Society of America, 39, 1 (January 1966), 74.

4. M. Schwartz, Information Transmission, Modulation, and Noise, McGraw-Hill Book Co., Inc., New York, 1970: Section 4.2.

5. R.L. Riegler and R.T. Compton, Jr., "An Adaptive Array for Interference Rejection," Proc. IEEE, 61, 6 (June 1973), 748.

6. R.T. Compton, Jr., "An Adaptive Array in a Spread-Spectrum Communication System," Proc. IEEE, 66, 3 (March 1978), 289.

7. A.J. Berni, "Weight Jitter Phenomena in Adaptive Array Control Loops," IEEE Transactions, AES-13, 7 (July 1977), 335.

8. A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill Book Co., New York, 1965, 356.

9. T.C. Kincaid, "The Complex Representation of Signals," Report No. R67EMH5, October 1966, General Electric Col, HMED Technical Publications, Box 1122 (LeMoyne Avenue), Syracuse, New York 13201.

10. N. Abramson, Information Theory and Coding, Mc-Graw-Hill Book Co., New York, 1963, 85.

11. N. Abramson, op. cit.

12. W.C. Lindsey and M.K. Simon, Telecommunication System Engineering, Prentice-Hall, New Jersey, 1973.

13. D.M. DiCarlo and R.T. Compton, Jr., "Reference Loop Phase Shift in Adaptive Arrays," IEEE Transactions, AES-14, 4 (July 1978), 599.

14. D.M. DiCarlo, "Reference Loop Phase Shift in an N-Element Adaptive Array," IEEE Transactions, AES-15, 4 (July 1979), 576.

15. J.D. Lambert, Computational Methods in Ordinary Differential Equations, John Wiley and Sons Ltd., London, 1973; Chapter 4.

16. F. Russo, "Bit Synchronization for Phase-Noncoherent Reception of Binary FSK Signals," IEEE Transactions, AES-12, 3 (May 1976), 362.

APPENDIX

```
0010   C   SIMULATION PROGRAM FOR AN ADAPTIVE ARRAY SYSTEM USING
0020   C   FSK MODULATION.
0030   C   THIS PROGRAM SIMULATES A TWO ELEMENT LMS ADAPTIVE ARRAY,
0040   C   A MARKOV SOURCE FOR THE DESIRED SIGNAL BITS, A BIT PRED-
0050   C   ICTOR CIRCUIT, AND A REFERENCE SIGNAL LOOP.
0060   C   THE SIMULATION IS DONE IN DISCRETE TIME BY SAMPLING EACH
0070   C   SIGNAL AT A RATE EQUAL TO FIVE TIMES THE CENTER FREQUENCY
0080   C   OF THE DESIRED FSK SIGNAL.
0090   C   THE PROGRAM CALCULATES AND STORES THE WEIGHT SAMPLES AND
0100   C   THE OUTPUT CONSISTS OF PLOTS OF THE ARRAY WEIGHTS AND AN
0110   C   ANTENNA PATTERN CALCULATED AT THE FINAL TIME SAMPLE.
0120   C
0130           DIMENSION TA(500),W1(500),W2(500),W3(500),W4(500),TAP(500)
0140           DIMENSION ASOUT(500),AIOUT(500)
0150           DIMENSION AOT(5),SN(5),SO(5)
0160           DIMENSION ANG(361),DBS(361)
0170           PI=3.14159265359
0180   C
0190   C   INPUT DATA
0200   C
0210   C       DESIRED SIGNAL PARAMETERS
0220            FS0=990.
0230            FS1=1010.
0240            AS=1.0
0250            THS=0.
0260   C       CW INTERFERENCE PARAMETERS
0270            AI=1.
0280            FI=990.
0290            THI=45.
0300   C       MARKOV BIT CHANGE PROBABILITY
0310            PA=1.
0320   C       SAMPLING RATE
0330            FSAM=5000.
0340   C       OSCILLATOR PARAMETERS
0350            FOS0=1010.
0360            FOS1=990.
0370            DOS=0.
0380   C       LIMITER PARAMETERS
0390            GL=2.
0400            COFF=1.
```

```
0410   C      REFERENCE LOOP BANDPASS FILTER PARAMETERS
0420           FC=2000.
0430           BW=5.
0440   C      LMS ARRAY FEEDBACK LOOPS GAIN CONSTANT
0450           CONV=0.00005
0460   C      ANTENNA SEPARATION IN DESIRED SIGNAL WAVELENGTHS
0470           AL=0.5
0480   C      THERMAL NOISE POWER
0490           S2=.1
0500   C
0510   C   INITIAL VALUES
0520   C
0530   C      SEED FOR THE RANDOM NUMBER ROUTINE
0540           IX=4021947
0550   C      MEAN VALUE OF THE THERMAL NOISE
0560           AM=0.0
0570   C      INITIAL TRANSMITTED BIT
0580           MSIG=1
0590   C      INITIAL BIT PREDICTION
0600           MOSC=1
0610   C      INITIAL WEIGHT VALUES
0620           W1(1)=1.0
0630           W2(1)=1.0
0640           W3(1)=1.0
0650           W4(1)=1.0
0660   C      INITIAL TIME VALUE
0670           TA(1)=0.
0680   C      INITIAL VALUES FOR TRIG FUNCTIONS.  TRIG VALUES ARE ITTERATED
0690   C      TO PREVENT ERROR CAUSED BY COMPUTING TRIG FUNCTIONS OF VERY
0700   C      LARGE VALUES.
0710           OSIN0=0.
0720           OCOS0=1.
0730           OSIN1=0.
0740           OCOS1=1.
0750           DSIN0=0.
0760           DCOS0=1.
0770           DSIN1=0.
0780           DCOS1=1.
0790           SINWI=0.
0800           COSWI=1.
0810   C   UNIT CONVERSIONS
0820           WS0=2.*PI*FS0
0830           WS1=2.*PI*FS1
0840           WI=2.*PI*FI
0850           DT=1./FSAM
0860           WOS0=2.*PI*FOS0
0870           WOS1=2.*PI*FOS1
0880           AOS=PI*DOS/180.
0890           WC=2.*PI*FC
0900           WBW=2.*PI*BW
```

```
0910   C    INITIALIZE STACK VECTORS
0920            DO 10 I=1,5
0930            AOT(I)=0.
0940            SN(I)=0.
0950    10      SO(I)=0.
0960            T=0.0
0970            IWT=1
0980   C    INITIALIZE SCa vEIGHTS
0990            WP1=W1(1)
1000            WP2=W2(1)
1010            WP3=W3(1)
1020            WP4=W4(1)
1030   C    SIGNAL SPACE ANGLES
1040            WSC=(WS0+WS1)/2.
1050            BTS0=2.*PI*AL*SIN(THS*PI/180.)*WS0/WSC
1060            BTS1=2.*PI*AL*SIN(THS*PI/180.)*WS1/WSC
1070            BTSA=(BTS0+BTS1)/2.
1080            BTI=2.*PI*AL*SIN(THI*PI/180.)*WI/WSC
1090   C    BIT COUNTER
1100            DO 40 M=1,100
1110   C    PREDICT OPPOSITE OF LAST BIT
1120            IF(MSIG.EQ.1) MOSC=0
1130            IF(MSIG.EQ.0) MOSC=1
1140   C    MARKOV SOURCE;  FIRST ORDER SYMMETRIC
1150            CALL RANDU(IX,IY,RNUM)
1160            IX=IY
1170            IF(RNUM.LT.PA.AND.MSIG.EQ.0) MSIGP=1
1180            IF(RNUM.LT.PA.AND.MSIG.EQ.1) MSIGP=0
1190            MSIG=MSIGP
1200   C    RANDOM PHASE FOR INTERFERENCE BIT
1210            CALL RANDU(IX,IY,RNP)
1220            IX=IY
1230            RNP=RNP*2.*PI
1240   C    WEIGHTS SAVED PER BIT COUNTER
1250            DO 35 I=1,5
1260   C    ITTERATIONS PER WEIGHT SAVED COUNTER
1270            DO 30 J=1,100
1280   C    DESIRED SIGNAL CALCULATIONS
1290            DSIN0P=DSIN0*COS(WS0*DT)+DCOS0*SIN(WS0*DT)
1300            DCOS0P=DCOS0*COS(WS0*DT)-DSIN0*SIN(WS0*DT)
1310            DSIN1P=DSIN1*COS(WS1*DT)+DCOS1*SIN(WS1*DT)
1320            DCOS1P=DCOS1*COS(WS1*DT)-DSIN1*SIN(WS1*DT)
1330            IF(MSIG.EQ.1) GO TO 20
1340            XS1=AS*DCOS0P
1350            XS2=AS*DSIN0P
1360            XS3=AS*(DCOS0*COS(WS0*DT-BTS0)-DSIN0*SIN(WS0*DT-BTS0))
1370            XS4=AS*(DSIN0*COS(WS0*DT-BTS0)+DCOS0*SIN(WS0*DT-BTS0))
1380            GO TO 13
1390    20      XS1=AS*DCOS1P
```

```
1400            XS2=AS*DSIN1P
1410            XS3=AS*(DCOS1*COS(WS1*DT-BTS1)-DSIN1*SIN(WS1*DT-BTS1))
1420            XS4=AS*(DSIN1*COS(WS1*DT-BTS1)+DCOS1*SIN(WS1*DT-BTS1))
1430     13     DSIN0=DSIN0P
1440            DCOS0=DCOS0P
1450            DSIN1=DSIN1P
1460            DCOS1=DCOS1P
1470    C    INTERFERENCE SIGNAL CALCULATIONS
1480            XI1=AI*(COSWI*COS(WI*DT+RNP)-SINWI*SIN(WI*DT+RNP))
1490            XI2=AI*(SINWI*COS(WI*DT+RNP)+COSWI*SIN(WI*DT+RNP))
1500            XI3=AI*(COSWI*COS(WI*DT-BTI+RNP)-SINWI*SIN(WI*DT-BTI+RNP))
1510            XI4=AI*(SINWI*COS(WI*DT-BTI+RNP)+COSWI*SIN(WI*DT-BTI+RNP))
1520            SINWIP=SINWI*COS(WI*DT)+COSWI*SIN(WI*DT)
1530            COSWIP=COSWI*COS(WI*DT)-SINWI*SIN(WI*DT)
1540            SINWI=SINWIP
1550            COSWI=COSWIP
1560    C    RANDOM NOISE
1570            CALL GAUSS(IX,S2,AM,RNOS1)
1580            CALL GAUSS(IX,S2,AM,RNOS2)
1590            CALL GAUSS(IX,S2,AM,RNOS3)
1600            CALL GAUSS(IX,S2,AM,RNOS4)
1610    C   COMPOSITE SIGNALS
1620            X1=XS1+XI1+RNOS1
1630            X2=XS2+XI2+RNOS2
1640            X3=XS3+XI3+RNOS3
1650            X4=XS4+XI4+RNOS4
1660    C   ARRAY OUTPUT
1670            ASIG=WP1*X1+WP2*X2+WP3*X3+WP4*X4
1680            WFACS=SQRT(WP1**2+WP2**2+WP3**2+WP4**2+2.*(WP1*WP3+WP2*WP4)
1690         &BTSA)+2.*(WP2*WP3-WP1*WP4)*SIN(BTSA))
1700            WFACI=SQRT(WP1**2+WP2**2+WP3**2+WP4**2+2.*(WP1*WP3+WP2*WP4)
1710         &BTI)+2.*(WP2*WP3-WP1*WP4)*SIN(BTI))
1720            THWI=ATAN2(WP2+WP3*SIN(BTI)+WP4*COS(BTI),
1730         &WP1+WP3*COS(BTI)-WP4*SIN(BTI))
1740            THW0=ATAN2(WP2+WP3*SIN(BTS0)+WP4*COS(BTS0),
1750         &WP1+WP3*COS(BTS0)-WP4*SIN(BTS0))
1760            THW1=ATAN2(WP2+WP3*SIN(BTS1)+WP4*COS(BTS1),
1770         &WP1+WP3*COS(BTS1)-WP4*SIN(BTS1))
1780    C   BEGIN CALCULATIONS FOR LIMITER OUTPUT.  THE FOLLOWING CODE
1790    C   ALLOWS FOR WEAKER SIGNAL ATTENUATION IN THE LIMITER.
1800    C   REFERENCE COMPONENT DUE TO INTERFERENCE
1810            AIP=AI*WFACI
1820            WLI=WI+W0S0
1830            IF(MOSC.EQ.1) WLI=WI+W0S1
1840            CALL LFILT(AIFAC,WLI,WC,WBW)
1850            AIP=AIP*AIFAC*GL
1860    C   REFERENCE SIGNAL COMPONENT DUE TO SIGNAL
1870            ASP=AS*WFACS
1880            IF(MSIG.EQ.1) GO TO 61
1890            WLS=WS0+W0S0
```

```
1900            IF(MOSC.EQ.1) WLS=WS0+WOS1
1910            CALL LFILT(ASFAC,WLS,WC,WBW)
1920            ASP=ASP*ASFAC*GL
1930            GO TO 62
1940     61     WLS=WS1+WOS0
1950            IF(MOSC.EQ.1) WLS=WS1+WOS1
1960            CALL LFILT(ASFAC,WLS,WC,WBW)
1970            ASP=ASP*ASFAC*GL
1980     62     CONTINUE
1990            COFF2=2.*COFF
2000            IF(ASP.LE.COFF.AND.AIP.LE.COFF) GO TO 78
2010            IF(ASP.LE.COFF.AND.AIP.GE.COFF2) GO TO 63
2020            GO TO 64
2030     63     ASP=ASP/2.
2040            AIP=COFF
2050            GO TO 78
2060     64     CONTINUE
2070            IF(AIP.LE.COFF.AND.ASP.GE.COFF2) GO TO 65
2080            GO TO 66
2090     65     ASP=COFF
2100            AIP=AIP/2.
2110            GO TO 78
2120     66     CONTINUE
2130            IF(AIP.LE.COFF.AND.ASP.GT.COFF.AND.ASP.LT.COFF2) GO TO 67
2140            GO TO 68
2150     67     ASP=COFF
2160            AIP=COFF*AIP/ASP
2170            GO TO 78
2180     68     CONTINUE
2190            IF(ASP.LE.COFF.AND.AIP.GT.COFF.AND.AIP.LT.COFF2) GO TO 69
2200            GO TO 70
2210     69     ASP=COFF*ASP/AIP
2220            AIP=COFF
2230            GO TO 78
2240     70     CONTINUE
2250            RATIO=AIP/ASP
2260            IF(RATIO.LT.0.5) GO TO 71
2270            GO TO 72
2280     71     ASP=COFF
2290            AIP=COFF/2.
2300            GO TO 78
2310     72     CONTINUE
2320            IF(RATIO.GE.0.5.AND.RATIO.LT.1.) GO TO 73
2330            GO TO 74
2340     73     ASP=COFF
2350            AIP=COFF*RATIO
2360            GO TO 78
2370     74     CONTINUE
2380            IF(RATIO.GE.1.AND.RATIO.LT.2.) GO TO 75
2390            GO TO 76
```

```
2400     75     ASP=COFF/RATIO
2410            AIP=COFF
2420            GO TO 78
2430     76     CONTINUE
2440            IF(RATIO.GE.2.) GO TO 77
2450            GO TO 78
2460     77     ASP=COFF/2.
2470            AIP=COFF
2480     78     CONTINUE
2490            REFI=AIP*(COSWI*COS(RNP-THWI)-SINWI*SIN(RNP-THWI))
2500            IF(MSIG.EQ.1) GO TO 79
2510            REFS=ASP*(DCOS0*COS(THW0)+DSIN0*SIN(THW0))
2520            GO TO 80
2530     79     REFS=ASP*(DCOS1*COS(THW1)+DSIN1*SIN(THW1))
2540     80     CONTINUE
2550   C    TOTAL REFERENCE SIGNAL
2560            REF=REFS+REFI
2570   C  ERROR SIGNAL VALUE
2580            ER=ASIG-REF
2590   C  CALCULATE NEW WEIGHTS
2600            WP1=WP1-2.*CONV*ER*X1
2610            WP2=WP2-2.*CONV*ER*X2
2620            WP3=WP3-2.*CONV*ER*X3
2630            WP4=WP4-2.*CONV*ER*X4
2640            T=T+DT
2650     30     CONTINUE
2660   C  SAVE EVERY 20TH WEIGHT VALUE FOR PLOT
2670            IWT=IWT+1
2680            IF(IWT.GT.500) GO TO 35
2690            W1(IWT)=WP1
2700            W2(IWT)=WP2
2710            W3(IWT)=WP3
2720            W4(IWT)=WP4
2730            ASOUT(IWT)=AS*WFACS
2740            AIOUT(IWT)=AI*WFACI
2750            TA(IWT)=T-DT
2760     35     CONTINUE
2770     40     CONTINUE
2780            DO 99 I=1,500
2790     99     TAP(I)=TA(I)*10.
2800   C    PLOT RESULTS
2810   C    GENERAL PLOTTING ROUTINE USED
2820            CALL PLTPKM(TAP,W1,500,500,-1,0,1,.25)
2830            CALL PLTPKM(TAP,W2,500,500,-1,0,1,.25)
2840            CALL PLTPKM(TAP,W3,500,500,-1,0,1,.25)
2850            CALL PLTPKM(TAP,W4,500,500,-1,0,1,.25)
2860            CALL PLTPKM(TAP,ASOUT,500,500,-1,0,2,.25)
2870            CALL PLTPKM(TAP,AIOUT,500,500,-2,0,2,.25)
2880            CALL PATPLT(AL,W1(500),W2(500),W3(500),W4(500),ANG,DBS,361)
2890            CALL EXIT
2900            END
```

```
2910   C
2920   C  SUBROUTINE FOR ANTENNA PATTERN PLOTS
2930   C
2940            SUBROUTINE PATPLT(AL,W1,W2,W3,W4,ANG,DBS,N)
2950            DIMENSION ANG(N),DBS(N)
2960            PI=3.14159265359
2970            NP=N-1
2980            DO 5 I=1,N
2990            ANG(I)=PI*(FLOAT(I-1)/NP-0.5)
3000            BETA=2.*PI*AL*SIN(ANG(I))
3010    5       DBS(I)=W1*W1+W2*W2+W3*W3+W4*W4+2.*(W1*W3+W2+W4)*
3020         &COS(BETA)+2.*(W2*W3-W1*W4)*SIN(BETA)
3030            AMAX=0.
3040            DO 10 I=1,N
3050            IF(DBS(I).LT.AMAX) GO TO 10
3060            AMAX=DBS(I)
3070    10      CONTINUE
3080            DO 15 I=1,N
3090            DBS(I)=10.*ALOG10(ABS(DBS(I)/AMAX+1.0E-30))
3100    15      ANG(I)=ANG(I)*180./PI
3110            MN=-N
3120            CALL PLTPKM(ANG,DBS,N,MN,-1,0,1,.25)
3130            RETURN
3140            END
3150   C
3160   C       SUBROUTINE FOR GAUSSIAN RANDOM NOISE
3170   C
3180            SUBROUTINE GAUSS(IX,S2,AM,V2)
3190            A=0.0
3200            DO 50 I=1,12
3210            CALL RANDU(IX,IY,Y)
3220            IX=IY
3230    50      A=A+Y
3240            V2=(A-6.0)*S2+AM
3250            RETURN
3260            END
3270   C
3280   C   FUNCTION FOR RANDOM NUMBER GENERATION
3290   C
3300            SUBROUTINE RANDU(IX)
3310            IY=IX*65539
3320            IF(IY)5,6,6
3330      5     IY=IY+2147483647+1
3340      6     YFL=IY
3350            YFL=YFL*.4656613E-9
3360            RETURN
3370            END
```

```
C
C   LOOP BANDPASS FILTER SUBROUTINE
C
      SUBROUTINE LFILT(FAC,WL,W0,WBW)
      COMPLEX G,H,S
      AL=0.5*WBW
      S=CMPLX(0.,W0)
      G=S/(S*S+2.*AL*S+AL*AL+W0*W0)
      G=1./G
      AK=CABS(G)
      S=CMPLX(0.,WL)
      H=S*AK/(S*S+2.*AL*S+AL*AL+W0*W0)
      FAC=CABS(H)
      RETURN
      END
```


DATE
FILMED
4-8