1972

# Performance of self bit synchronizers for binary overlapping signals

Chung-Tao David Wang



Department: Electrical and Computer Engineering

# PERFORMANCE OF SELF BIT SYNCHRONIZERS

# FOR BINARY OVERLAPPING SIGNALS

by

Chung-Tao David Wang, 1943-

DISSERTATION

Presented to the Faculty of the Graduate School of the

UNIVERSITY OF MISSOURI-ROLLA

In Partial Fulfillment of the Requirement for the Degree

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

1972



_______________ Advisor

Maxwell E. Engelhardt

R. E. Zimmer

Eddie R. Fowler

E. C. Bertnolli

## ABSTRACT

This research is concerned with the optimum and suboptimum ways of providing self bit synchronization for binary overlapping signals.

A maximum likelihood synchronizer is derived as the optimum approach. The suboptimum ways are those employing decision-directed feedback and matched derivative filter techniques which are proposed for treating the overlapping signals. Combining the two suboptimum techniques along with the bandlimited version of the overlapping signal, a suboptimum synchronizer is derived. The performance of the above synchronizer is evaluated by Monte Carlo simulation techniques.

Finally, a bit synchronizer using nonlinear filtering theory is considered. The performance of a nonlinear bit synchronizer is discussed.

# ACKNOWLEDGEMENT

The author wishes to express his sincere appreciation to all members of his committee, but above all to his advisor, Dr. Thomas L. Noack, for his guidance and assistance throughout this work.

He also wishes to thank Dr. Roger E. Ziemer for his helpful discussions and constructive criticism.

# TABLE OF CONTENTS

Page

# LIST OF ILLUSTRATIONS

Figure | Page

# LIST OF TABLES

# I. INTRODUCTION

## A. Synchronization

A binary communication system can be described as transmitting and receiving a sequence of binary symbols of predetermined duration and form from one point of space to a second. In the transmission process, the duration and form of the symbols may be altered, some symbols may overlap with the others, and further the symbol sequence may be perturbed by noise of various kinds. The receiver is designed to recover or detect the original binary data upon receiving the distorted, noise perturbed signals.

It is well known that if the epoch and the duration of each symbol are known at the receiver and the noise assumed additive and Gaussian, an optimum receiver called the matched filter or correlation detector achieves the minimum error probability. The matched filter following the coherent detector consists of an integrator, a sampler and a reset configuration. Bit synchronization timing is required in the matched filter to sample and discharge the integrator properly. If the bit synchronization timing for the matched filter is inaccurate, then sampling may take place too soon or too late, thereby reducing the probability of making the correct decision.

Basically, there are three levels of synchronization. Bit synchronization is necessary for the optimum (matched) filter in order to make bit by bit decisions. Word synchronization is also needed to sort out bits into their appropriate words. Finally, frame synchronization is required by the data user if one is to distinguish between frames. Further discussion about the types of synchronization are given by Stiffler[4] and by Golomb, et. al.[7]

Two different approach have been derived for obtaining bit synchronization in binary communications. In transmitted reference (TR) systems, a separate synchronizing signal is transmitted through a separate channel solely for the purpose of achieving synchronization. The immediate disadvantage of this approach is that the transmitted power must be shared by the synchronizing signal and the data signals. In self synchronized (SS) systems, the receiver consists of a epoch estimation portion which provides the necessary timing. The timing is extracted directly from the information bearing signals. In this study, only self bit synchronization is investigated.

## B. The Problem

This research considers the process of self bit synchronization as it applies to binary overlapping Non-Return-to-Zero (NRZ) signals. For convenience, it will be called the binary overlapping signal in the sequel. The difference between overlapping signals and ordinary NRZ signals is that the former, through channel-induced distortion, have had their symbol waveforms stretched out so that each symbol may overlap with symbols in the preceding and following intervals. The resulting signal is therefore no longer anti-correlated. A formal description of such signals will be presented in Chapter III.

Considerable research concerning self bit synchronization exists in the literature. In Chapter II, a brief review of those self bit synchronization techniques which are related to the development of this study is presented.

In Chapter III, a maximum likelihood synchronizer for binary overlapping signals is derived. The synchronizer structure consists of matched filters, a transition detector, and an accumulator. The form of the optimum synchronizer does not however, lead itself to a simple implementation. A more practical approach called decision-directed feedback is proposed in Chapter IV. The basic idea of this approach is to detect a particular symbol by properly subtracting the overlapping head from the preceding symbol and the overlapping tail from the following symbol. The result is the Decision-Directed (DD) detector. The performance of this detector is discussed.

In Chapter V, synchronization using the matched derivative filter (MDF) and the transition detector (TD) is proposed particularly to deal with overlapping signals. The resulting synchronizer is called the Absolute-Value Bit Synchronizer (AVBS). The performance is evaluated by Monte Carlo simulation techniques.

In Chapter VI, we use the techniques developed in Chapter IV and Chapter V with bandlimited overlapping signals to obtain a sub-optimum synchronizer. The basic idea of this approach is to find the estimate of the phase by subtracting the DD detector output from the bandlimited signal output. The result is then multiplied by the derivative of the signal. For comparison, the same technique is applied to PCM/split-phase signals.

Another approach to bit synchronization problems employs non-linear filtering theory. The general nonlinear filtering approach is to find the optimum estimate of the bit sequence by solving the stochastic differential equation for the conditional probability density function. The advantage of this approach is that it can be used to treat more general communication problems for various shapes of symbol wave-forms. A bit synchronizer structure using this approach is considered in Chapter VII.

# II. LITERATURE REVIEW

## A. Self Synchronization systems

Basically, the problem can be separated into two parts, the epoch estimation portion and the bit detection portion. Various types of signals and approaches are proposed. Probability of bit error versus signal-to-noise ratio for different systems are used as measures of system performance.

Wintz and Hancock[1] considered the problem of determining the performance of an epoch-correlation detector system for an M-ary alphabet. The phase of the carrier is assumed known and no specific form is given for the epoch estimator. Probability of detection error is computed for binary signaling with a prescribed autocorrelation function.

Van Horn[2] presents a correlation strategy for self bit synchronization which uses a bank of correlators similar to the detection scheme used in some radar systems. No claims are made about the optimality of this system or about its practicality.

Stiffler[3,4] proposed a maximum likelihood procedure for estimation of synchronization position which requires the knowledge of the infinite past or at least enough of the past so that truncation errors are negligible.

Wintz and Luecke[5,6] proposed a maximum likelihood (ML) synchronizer for anti-correlated signals. The ML synchronizer structure involves evaluating the log-hyperbolic cosine of the correlation function for each time interval. For easier implementation, a suboptimum system is considered. This system composed of a cascade of a lowpass filter, a square-law nonlinearity and a bandpass filter centered at the symbol rate. Monte Carlo simulations of the optimum synchronizer and of the analytical and experimental performance of the suboptimum systems are presented.

The problem is closely related to the problem of obtaining the required reference signal for coherent detection of PSK signals. Van Trees[7] proposed a synchronizer which consists of transmitted reference carrier and a squaring loop for the self generation of a reference signal. He showed that optimum performance occured when the carrier power was decreased to zero and only the self synchronizer was used. Lindsey[8] has evaluated the error probability for such a system with the assumption that the necessary bit timing was available for the correlation detector.

Simon[9,10] developed the steady-state phase-noise performance of an absolute value type of early-late-gate bit synchronizer with the use of the Fokker-Planck method. The results show that this system is better than two other commonly used synchronizers.

Proakis, et. al.[11] investigates the effect of using baud Decision to Direct the phase Measurement process (DDM) in order to obtain a less noisy measurement which in turn acts on the decisions. By Monte Carlo simulation on orthogonal and on anti-correlated signals, the results show the DDM approach yields a lower probability of error than that of the non-DDM method, at all signal-to-noise ratios.

Oberst and Schilling[12,13] derived upper and lower bounds on the probability of error for self-synchronized binary PSK systems. In addition to Decision Feedback (DF) and Phase Doubling (PD) systems, a new maximum likelihood (ML) system is derived and studied. Simulation results show that ML-PSK is the best and PD-PSK is the worst in comparing the probability of error in each case.

For a PCM/NRZ signal, unlimited in bandwidth and in the presence of white Gaussian noise, the "integrate-and-dump" circuit is equivalent to the matched filter[14] and is an optimum detector, which is defined as the detector that achieves the lowest probability of bit error $P_E$ for a given signal-to-noise ratio (SNR). In general, however, restrictions in bandwidth for various reasons degrade the performance of the matched filter or correlation detector because of intersymbol interference. The influence of bandwidth restriction on performance of a PCM/NRZ signal is considered by Martinides and Reijns[15]. The detector used in the investigation contains a device that integrates the signal over the bit period. The theoretical results were obtained by a Fourier analysis of the bandwidth restricted signals and by an autocorrelation analysis of the bandwidth restricted noise. It is shown that the theoretical and experiment results are in good agreement.

Park[16] considered the PCM/NRZ systems operating in the bandlimited channel with two types of bit detector, integrate-and-dump, and bandlimit-and-sample. The results show that the integrate-and-dump is superior to the bandlimit-and-sample method. For a nearly optimum performance, the bandlimiting should not exceed about 0.7 of the bit rate.

Shehadeh and Tu[17,18] indicated that there is a fundamental error in Park's approach and the optimum receiving bandwidth should be set 0.9 of the bit rate of the transmitted NRZ signals in order to eliminate the effect of intersymbol interference due to bandwidth restriction. Further, Shehadeh and Tu[19] investigate the PCM/split-phase signals using an integrate-and-dump filter. The results, compared with those obtained for PCM/NRZ signals, indicate that the former signals require about twice as much bandwidth to have the same $P_E$ under same value of SNR.

B. Nonlinear Filtering Theory

When random disturbances occur and only noise corrupted measurements are available, it is well known that all the information provided by such measurements about the condition or "state" of a system is contained in the probability density function of the state conditioned on the entire history of the measurement. This density function thus becomes a prime object for study. Many authors have considered the problem of deriving a stochastic differential equation for finding the density function when the system disturbances and the measurement noise are both jointly Gaussian and white. This was first done in 1960 by Stratonovich[35] and later by Kushner[34] and Bucy[36] and many others.

Kushner[34] presented an expression for the probability density of the state conditioned upon the observation as well as the initial data in the continuous case. He then derived a partial differential equation satisfied by this conditional density function. This equation is of great usefulness in dealing with many communications and control problems.

Bucy[36] approaches the problem more rigorously than do Stratonovich and Kushner, but his approach is restricted to Gaussian distributions and the measurement noise which are statistically independent of each other.

Wonham[40] has used the theory of stochastic differential equations and a representation theorem from Doob[24] to obtain a stochastic differential equation for the conditional probability function when the system state is a scalar generalized Poisson process with a known transition probability. His measurement vector is linear in the state variable and contains additive white noise.

Fisher and Stear[41] presented a new approach to the formation of the multidimensional optimal nonlinear filtering problem. They unified and generalized the results of the previous authors through use of characteristic functions and theory of independent increment processes.

In Part II of this paper, the Gaussian independent increment noise restriction is relaxed and therefore a more general equation for the density function is found.

Since most signal and noise processes appearing in practice and in the literature are represented as solutions to stochastic differential equations with a white noise forcing function, the questions of uniqueness and existence of solutions to these equations have been examined by several authors. A rigorous mathematical theory of these equations was given by Ito[38]. Certain seemingly undesirable properties of the Ito formulation have caused Stratonovich[35] to define a symmetric stochastic integral and to derive nonlinear filters using this interpretation of the state and observation equation. Although the form of the filters appear different depending on which interpretation is made of the state equation, Stratonovich showed that both results are equivalent under a suitable transformation.

Lee and Komo[43] formed an optimal nonlinear sequential estimator for the case of square on-off anti-correlated pulses. The estimator structure is a function of jitter dynamics, coding statistics and signal-to-noise ratio. The development of this research in the nonlinear filtering approach is closely related to their work.

# III. MAXIMUM LIKELIHOOD SYNCHRONIZER

The basic problem in obtaining self synchronization is that of finding the epoch of each symbol. In this chapter, the maximum likelihood parameter estimation technique for deriving the structure of the optimum synchronizer is considered. The signal at the receiver is the overlapped version of the PCM/NRZ signal sequences. The case where the signal is not overlapped has been investigated by Wintz and Luecke[5,6].

## A. Basic Assumptions

The basic symbol has a duration one time unit (here, for convenience, one second is assumed in the sequel). It is assumed that m seconds are observed at the input of the synchronizer. The binary NRZ symbols and the binary overlapping symbols are shown in Fig. 3.1(a) and 3.1(b), respectively. It can be seen that the symbols at the transition instants are overlapped which causes various problems when estimating the epoch of the received symbols. The analytical expression for the overlapping symbol is

$$S_p(t) = \begin{cases} 1/2 + t/2\alpha, & |t| \leq \alpha \\ 1, & |t| \leq 1-\alpha \\ 1/2 + (1-t)/2\alpha, & |1-t| \leq \alpha \\ 0, & \text{otherwise} \end{cases} \qquad (3.1)$$

where $\alpha$ is defined as the overlapping parameter and is in the range

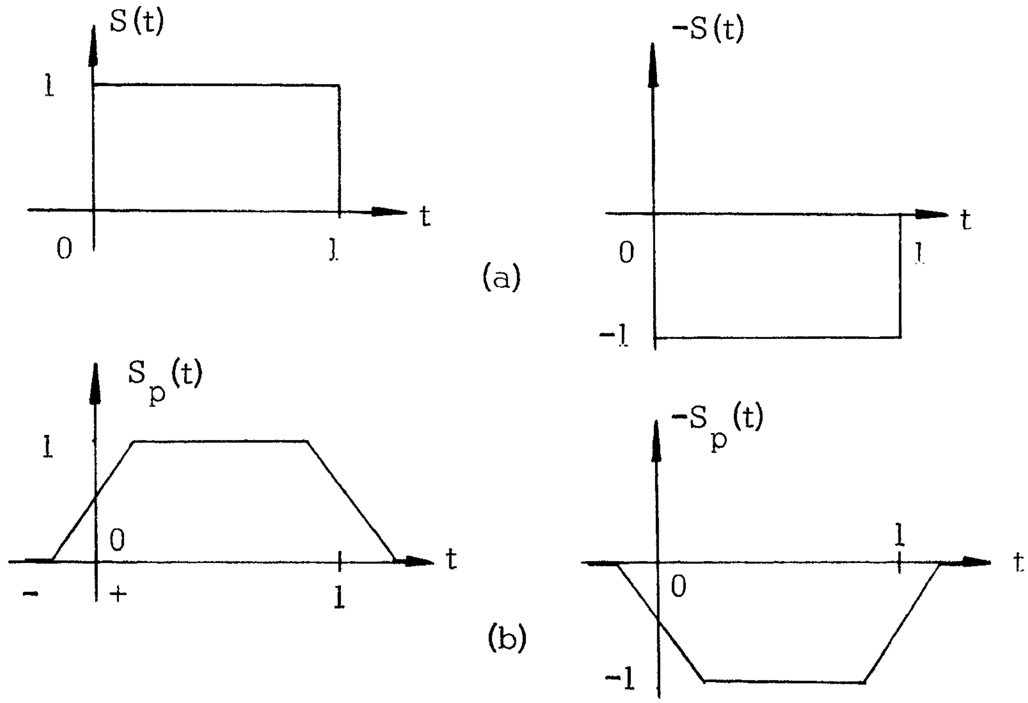



Fig. 3.1(a) Binary NRZ symbols
(b) Binary overlapping symbols

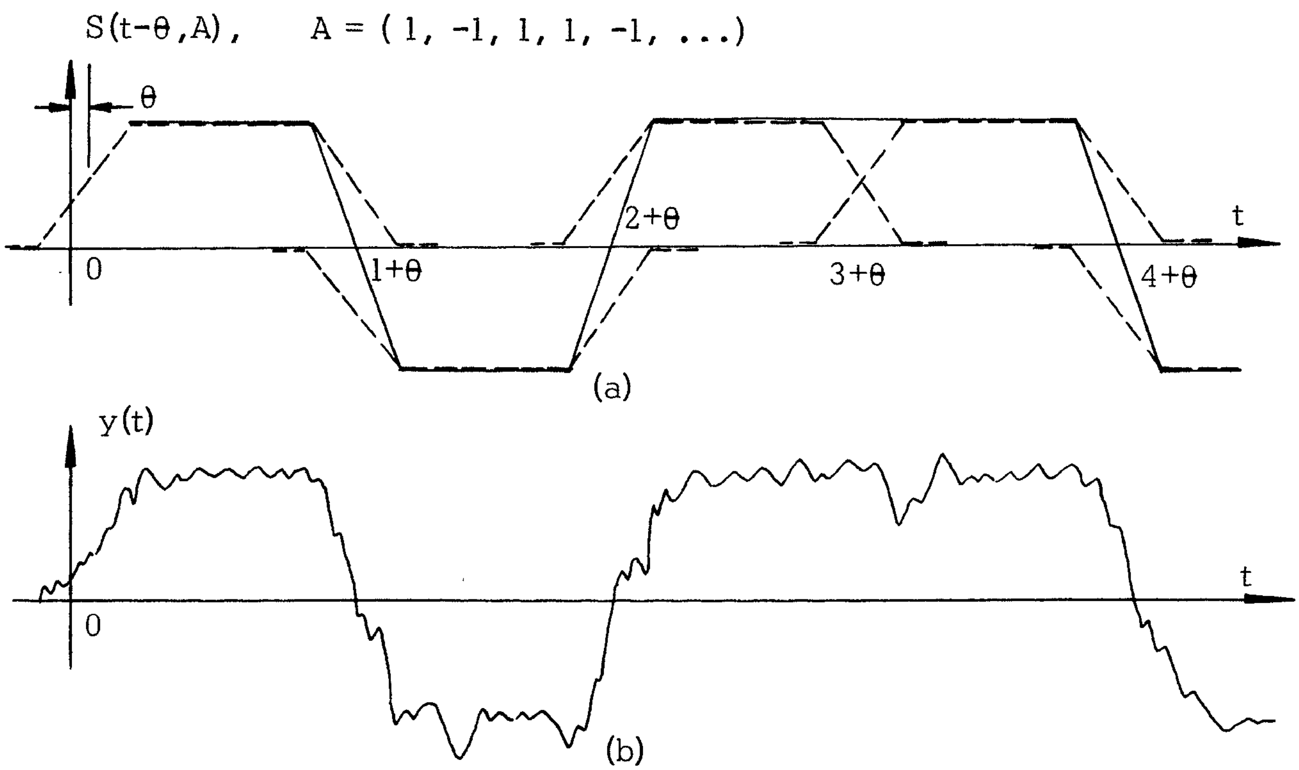



Fig. 3.2(a) Received signal without noise
(b) Received signal with noise

from -0.5 to +0.5. The received signal waveforms are indicated by Fig. 3.2(a) for the noiseless case, and by Fig. 3.2(b) for the noisy case. The dotted line in the figure shows the individual overlapping symbol and the solid line is the actual received waveform.

The received signal is perturbed by an additive noise, n(t), which is assumed to be a sample function from a Gaussian random process with zero mean and known variance. The input to the synchronizer is of the following form:

$$y(t) = S(t;\theta,A) + n(t), \qquad 0 \leqslant t < m \tag{3.2}$$

where $A = (a_1, a_2, \ldots, a_m)$,

$a_j$ = +1 or -1 with equal probability, and

$\theta$ = epoch to be estimated which is assumed uniformly distributed between -1/2 and +1/2.

B. Derivation of Optimum Synchronizer

In order to find the maximum likelihood (ML) solution, the conditional probability function $P(Y|\theta)$ is required. Owing to the presence of the random variable A, the following expression is considered.

$$P(Y|\theta) = \int_A P(Y|\theta,A)\, P(A)\, dA \tag{3.3}$$

We now find the conditional probability density function by using the sampling method. That is first take N samples in each interval. Then let the samples become dense and obtain an integral form of $P(Y|\theta,A)$. For each sample i, $y(i) = S(i;\theta,A) + n(i)$, $i=1,2,\ldots,mN.$, where n(i) is Gaussian distributed with zero mean and variance $B_oN$. The conditional joint pdf, $P(Y\ \theta,A)$ for mN samples is written as:

$$P(Y|\theta,A) = \prod_{i=0}^{mN} (1/\sqrt{2\pi B_o N}) \ \exp\left\{-(1/2B_oN)\left[y(i) - S(i;\theta,A)\right]^2\right\} \quad (3.4)$$

When N becomes very large, (3.4) can be written as an integral as follows:

$$P(Y|\theta,A) = K_1 \ \exp\left\{-(1/2B_o)\int_0^m \left[y(t) - S(t;\theta,A)\right]^2 dt\right\} \quad (3.5)$$

where $K_1$ is a constant. Because of the overlapping situation, the set of random variables $a_j$, $j=1,2,\ldots,m$, are correlated with each other. Thus $P(Y|\theta,A)$ cannot be expressed as the product of the conditional pdf of individual symbols, namely, $P(Y|\theta,a_j)$, $j=1,2,\ldots,m$. In order to proceed, we group the signal sequences as follows:

$$S(t;\theta,A) = \sum_{j=1}^{m-1}\left( a_j S_p(t-j;\theta) + a_{j+1} S(t-j+1;\theta)\right) \quad (3.6)$$

and integrate each interval from (j- 1/2) to (j+ 1/2), so that (3.5) can be further simplified as follows:

$$P(Y|\theta,A) = K_1 \prod_{j=1}^{m-1} \exp\left\{-(1/2B_o)\int_{j-1/2}^{j+1/2}\left[y(t) - a_j S_p(t-j;\theta)\right.\right.$$

$$\left.\left.- a_{j+1} S_p(t-j-1;\theta)\right]^2 dt\right\} \tag{3.7}$$

There are six terms to be considered if we expand out the above expression. However, three of them, involving the integration of squared terms, are actually constants. Hence their product can be combined with $K_1$ to form another constant $K_2$. Since $K_2$ is not a function of $\theta$, it will not enter into the maximizing process. We simply ignore this quantity for awhile. Note that if $\theta$ is also assumed to be stationary, we can write $S(t;\theta) = S(t-\theta)$, for $-1/2 \leqslant \theta \leqslant 1/2$. Therefore, (3.7) is reduced to

$$P(Y|\theta,A) \doteq \prod_{j=1}^{m-1} \exp\left\{a_j B_j(\theta) + a_{j+1} C_j(\theta) - a_j a_{j+1} D\right\} \tag{3.8}$$

where

$$B_j(\theta) = (1/B_o) \int_{j-1/2}^{j+1/2} y(t)\, S_p(t-\theta-j)\, dt \tag{3.9}$$

$$C_j(\theta) = (1/B_o) \int_{j-1/2}^{j+1/2} y(t)\, S_p(t-\theta-j-1)\, dt \tag{3.10}$$

$$D = (1/B_o) \int_{j-1/2}^{j+1/2} S_p(t-\theta-j)\, S_p(t-\theta-j-1)\, dt \qquad (3.11)$$

D can be calculated immediately to be $\alpha/3$ and is not a function of $\theta$. The calculation of $B_j(\theta)$ and $C_j(\theta)$ proceeds as follows. When there is no transition in the interval $(j-1/2,\ j+1/2)$,

$$B_j(\theta) = ((a_j + a_{j+1})/2B_o)\ (1/2) \qquad (3.12)$$

$$C_j(\theta) = B_j(\theta). \qquad (3.13)$$

When there is a transition in the interval $(j-1/2,\ j+1/2)$, the situation is described by Fig. 3.3 for the case $a_j = 1$, and $a_{j+1} = -1$. After integration, the values for $B_j(\theta)$ and $C_j(\theta)$ can be found as follows.

$$B_j(\theta) = ((a_j - a_{j+1})/2B_o) \int_{-1/2}^{1/2} y(t)\, S(t-\theta)\, dt$$

$$= t_j \cdot (1/2 - (2\alpha/3) - (\theta^2/2\alpha) + (\theta^3/12\alpha^2)), \text{ for } \theta \geq 0 \qquad (3.14)$$

$$C_j(\theta) = t_j \cdot (-1/2 - (2\alpha/3) + (\theta^2/2\alpha) - (\theta^3/12\alpha^2)), \text{ for } \theta \geq 0 \qquad (3.15)$$

where $t_j = (a_j - a_{j+1})/2B_o$.

For the case $\theta \leqslant 0$, it can be found that the values for $B_j(\theta)$ and $C_j(\theta)$ are the same as those indicated by (3.14) and (3.15) but with a different sign. Fig. 3.4 shows the case for finding $B_j(\theta)$ and $C_j(\theta)$ when $\theta \leqslant 0$. In summary, $B_j(\theta)$ and $C_j(\theta)$ can be tabulated in Table 3.1 :

Table 3.1 $B_j(\theta)$ and $C_j(\theta)$ for various $\theta$'s

| | | without transition | with transition |
|---|---|---|---|
| $\theta > 0$ | $B_j(\theta)$ | $(a_j+a_{j+1})/\ 4B_o$ | $((a_j-a_{j+1})/2B_o)\ x$ |
| | $C_j(\theta)$ | $(a_j+a_{j+1})/\ 4B_o$ | $((a_j-a_{j+1})/2B_o)\ (-x)$ |
| $\theta < 0$ | $B_j(\theta)$ | $(a_j+a_{j+1})/\ 4B_o$ | $((a_j-a_{j+1})/2B_o)\ y$ |
| | $C_j(\theta)$ | $(a_j+a_{j+1})/\ 4B_o$ | $((a_j-a_{j+1})/2B_o)\ (-y)$ |
| where $x = 1/2 - (2\alpha/3) - \theta^2/2\alpha + \theta^3/12\alpha^2$, $y = 1/2 - (2\alpha/3) - \theta^2/2\alpha - \theta^3/12\alpha^2$. | | | |

The next problem is to maximize $\int P(Y\,|\,\theta, A)\ P(A)\ dA$ to obtain the optimum estimate. Owing to the overlapping situation, the exponential term in (3.8) consists of both the symbol $a_j$ and the

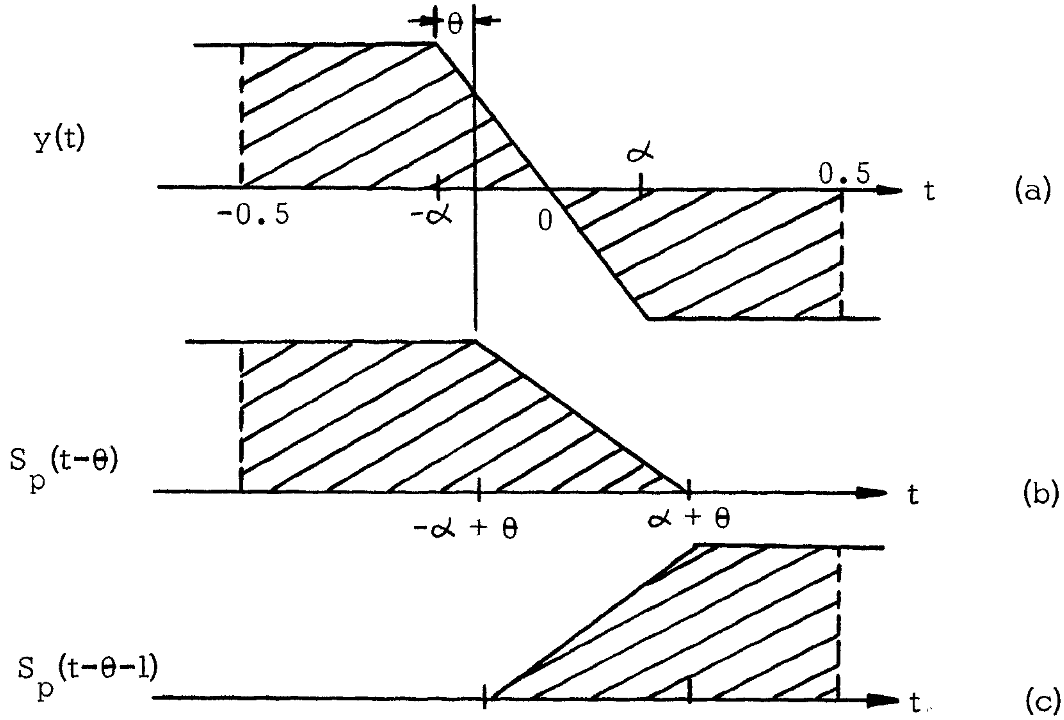



Fig. 3.3 On finding $B_j(\theta)$ and $C_j(\theta)$ when $\theta \geq 0$

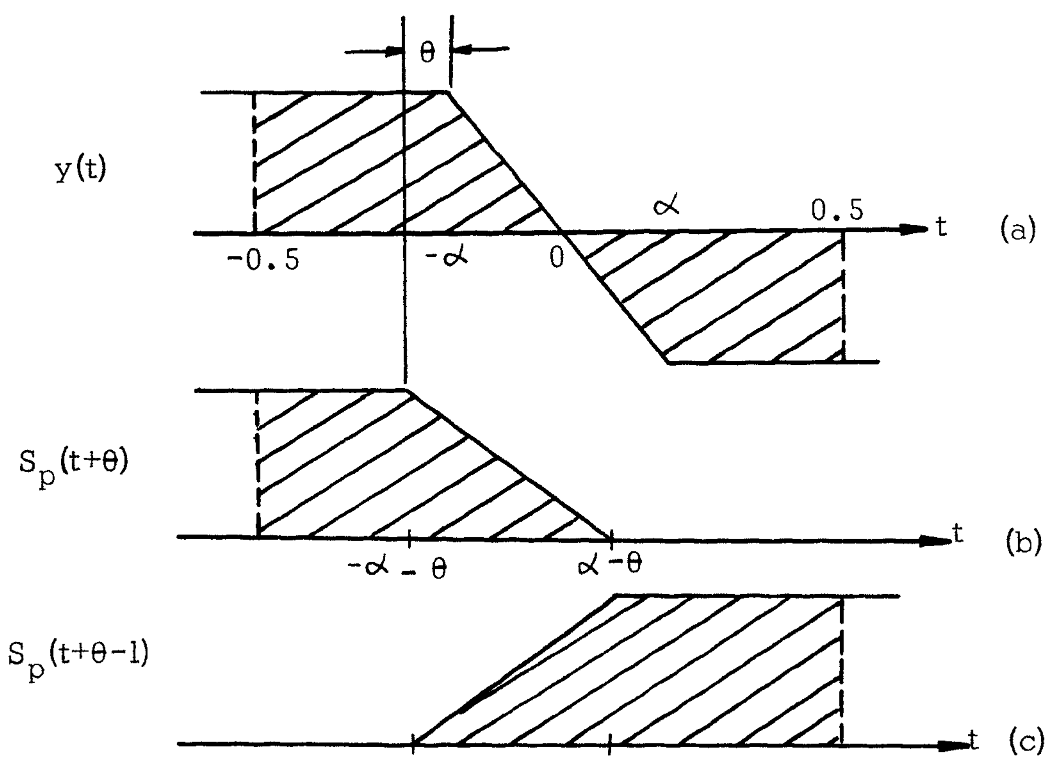



Fig. 3.4 On finding $B_j(\theta)$ and $C_j(\theta)$ when $\theta \leq 0$

following symbol $a_{j+1}$. The averaging process is therefore of a recursive nature. Further, the random variable $a_j$ is equally likely to be +1 or -1 so that four different cases should included. For convenience, let

$$Q(\theta, a_j, a_{j+1}) = \exp(\, a_j B_j(\theta) + a_{j+1} C_j(\theta) - a_j a_{j+1} D), \tag{3.16}$$

and let

$Q(\theta, a_j{=}1, a_{j+1}{=}1)$ be represented by simply $Q(1,1)$. If we use the subscripts 0 and 1 to represent the symbol -1 and +1, respectively, the probability of (j+1)th stage can be written as a function of the jth stage as follows:

$$\begin{cases} P_1(j+1) = P(Y|\theta, j+1, 1) = P(Y|\theta, j, -1)\ Q(-1,1) + P(Y|\theta, j, 1)\ Q(1,1) & \text{(3.17a)} \\ P_0(j+1) = P(Y|\theta, j+1, -1) = P(Y|\theta, j, -1)\ Q(-1,-1) + P(Y|\theta, j, 1)\ Q(1,-1) & \text{(3.17b)} \end{cases}$$

Or, if we write the above expressions in matrix form,

$$\begin{pmatrix} P_0(j+1) \\ P_1(j+1) \end{pmatrix} = \begin{pmatrix} Q(-1,-1) & Q(1,-1) \\ Q(-1,1) & Q(1,1) \end{pmatrix} \begin{pmatrix} P_0(j) \\ P_1(j) \end{pmatrix} \tag{3.18}$$

The next step is to compute the average of $P_0(j+1)$ and $P_1(j+1)$ by following equation:

$$\begin{pmatrix} P_2(j+1) \\ P_3(j+1) \end{pmatrix} = (1/2) \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} P_0(j+1) \\ P_1(j+1) \end{pmatrix}$$

$$= (1/2)\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} Q(-1,-1) & Q(1,-1) \\ Q(-1,1) & Q(1,1) \end{pmatrix}\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}\begin{pmatrix} P_2(j) \\ P_3(j) \end{pmatrix} \quad (3.19)$$

To write (3.19) in a matrix form, we have

$$P(j+1) = H_j(\theta) \ P(j) \quad (3.20)$$

where $H_j(\theta)$ is a two by two matrix having the following elements;

$$\begin{cases} h_{11}(\theta) = e^{-C_j} \cosh(D+B_j) + e^{C_j} \cosh(B_j-D) \\ h_{12}(\theta) = -e^{-C_j} \sinh(D+B_j) + e^{C_j} \sinh(D-B_j) \\ h_{21}(\theta) = e^{C_j} \cosh(D+B_j) - e^{-C_j} \cosh(D-B_j) \\ h_{22}(\theta) = -e^{C_j} \sinh(D+B_j) - e^{+C_j} \sinh(D-B_j) \end{cases} \quad (3.21)$$

The associated synchronizer structure is shown in the next section. The Monte Carlo simulation program , written in FORTRAN language , will be presented in Section 3.D.

## C. The Synchronizer Structure

The maximum likelihood estimate $\theta_{ML}$ is the value of $\theta$ that maximizes the likelihood function, $P(Y|\theta)$. Mathematically, the ML estimate corresponds to the limiting case of a MAP estimate in which a priori knowledge of $\theta$ approaches zero, or the variance of

$\theta$ approaches infinity. In this study, the maximum likelihood estimation is regarded as the optimum approach.

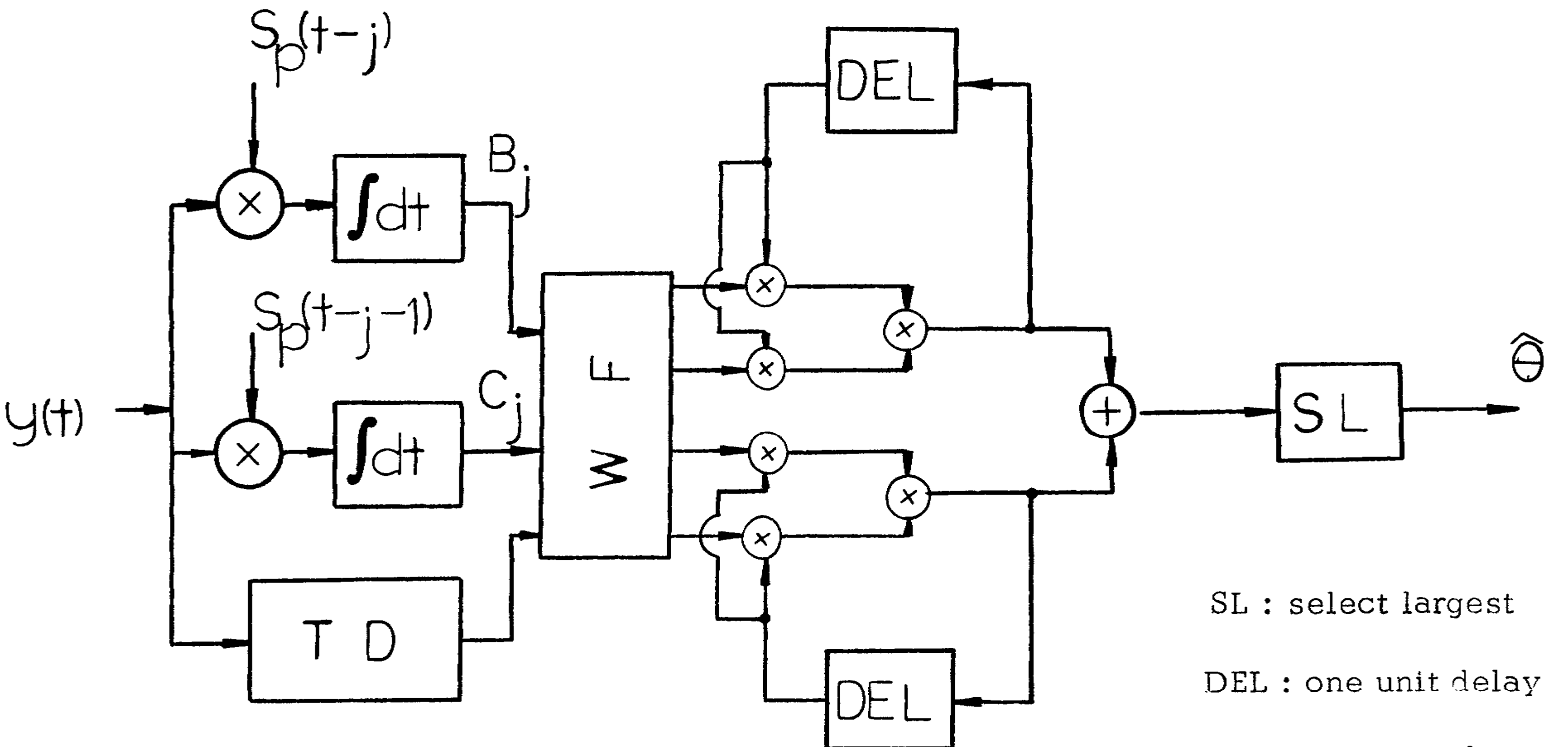
The synchronizer structure by using the analysis of Section B consists of matched filters, a transition device, a weighting function and a feedback circuit. It is shown in Fig. 3.5. The transition detector is a device which examines $a_j$ and $a_{j+1}$ in the presence of noise, and records an output $t_j$ according to the following rules:

$$\begin{cases} \text{If} \quad a_j = a_{j+1}, & t_j = (a_j + a_{j+1})/2. \\ \text{If} \quad a_j \neq a_{j+1}, & t_j = (a_j - a_{j+1})/2. \end{cases} \tag{3.22}$$

The weighting function is used to compute the Q-function defined by (3.16) after receiving messages from the matched filters and the transition detector. The feedback circuitry is used here to generate the conditional probability density function recursively.

The intepretation of the ML synchronizer proceeds as follows. To obtain the maximum value of the density function for a given received y(t), y(t) is first correlated with the the overlapping symbol $S_p(t)$ and $S_p(t-1)$ separately in the time interval ( 1/2, 3/2 ). At the same time y(t) is passed through a transition detector to record output $t_j$. The output of the matched filters and the transition detector are then passed to a weighting function which computes the four Q -functions. Then the conditional probability density function for each stage is calculated and stored. This is continued until the mth time interval is processed.

At the end of the mth symbol, the output statistics in each stage are compared, and the largest statistic is announced as the estimate of the correct synchronization position.




SL : select largest

DEL : one unit delay

TD : transition detector

WF : weighting function

Fig. 3.5 ML synchronizer

D. Monte Carlo Simulation Program and Results

In order to find the exact value of the conditional probability density function $P(Y|\theta)$, we again examine (3.7):

$$P(Y|\theta,A) = \prod_{i=1}^{m-1} K_1 \exp\left\{-(1/2B_o) \int_{i-1/2}^{i+1/2} \left[y(t) - a_i S_p(t-i-\theta) - a_{i+1} S_p(t-i-1-\theta)\right]^2 dt\right\}$$

$$= \prod_{i=1}^{m-1} K_1 \exp\left\{K_5 + a_i B_i(\theta) + a_{i+1} C_i(\theta) - a_i a_{i+1} D\right\}. \quad (3.23)$$

where $K_1 = 1/\sqrt{2\pi B_o}$. (3.24)

$K_5$ can be written as the sum of three terms:

$$K_5 = K_2 + K_3(\theta) + K_4(\theta) \quad (3.25)$$

where $K_2 = -(1/2B_o) \int_{i-1/2}^{i+1/2} y^2(t)\ dt$

$$= \begin{cases} -(3-4\alpha)/6B_o, & \text{if there is a transition} \\ -(1/2B_o), & \text{if there is no transition} \end{cases} \quad (3.26)$$

$$K_3(\theta) = -(1/2B_o) \int_{i-1/2}^{i+1/2} S_p^{\,2}(t-\theta-i)\ dt$$

$$K_3(\theta) = -(1/2B_o) \int_{-1/2}^{1/2} S^2(t-\theta)\ dt$$

$$= -(1/2B_o) \left\{ -\alpha/3 + 1/2 + \theta \right\} \qquad (3.27)$$

$$K_4(\theta) = -(1/2B_o) \int_{-1/2}^{1/2} S^2(t-\theta-1)\ dt$$

$$= -(1/2B_o) \left\{ -\alpha/3 + 1/2 - \theta \right\} \qquad (3.28)$$

At first glance, $K_3(\theta)$ and $K_4(\theta)$ are functions of $\theta$ so that they ought to be included in Section 3.B. However, their sum indicates that $K_5$ is not a function of $\theta$. That is

$$K_5 = K_2 + K_3(\theta) + K_4(\theta)$$

$$= \begin{cases} -((1 - 2\alpha)/B_o), & \text{if there is a transition} \\ -((3 - \alpha)/3B_o), & \text{if there is no transition} \end{cases} \qquad (3.29)$$

The flow chart of the ML synchronizer simulation program is shown in Fig. 3.7. The program is listed in Table 3.2. The input bit stream for the simulation program is generated by a uniform random number generator. The description of this subroutine is given in Appendix A. The signal-to-noise ratio (SNR) in the program is the power ratio indicated by numbers. $\theta$ in the simulation program is the true value.

For 40 bits as the input data to the ML synchronizer, the probability density function versus different $\theta$ are plotted in Fig. 3.7, using SNR as a parameter. The graphs appear to be nearly Gaussianly distributed with center around $\theta = 0$. At high SNR, the curve tends to peak up at the origin and to flatten out rapidly as the SNR decreases.

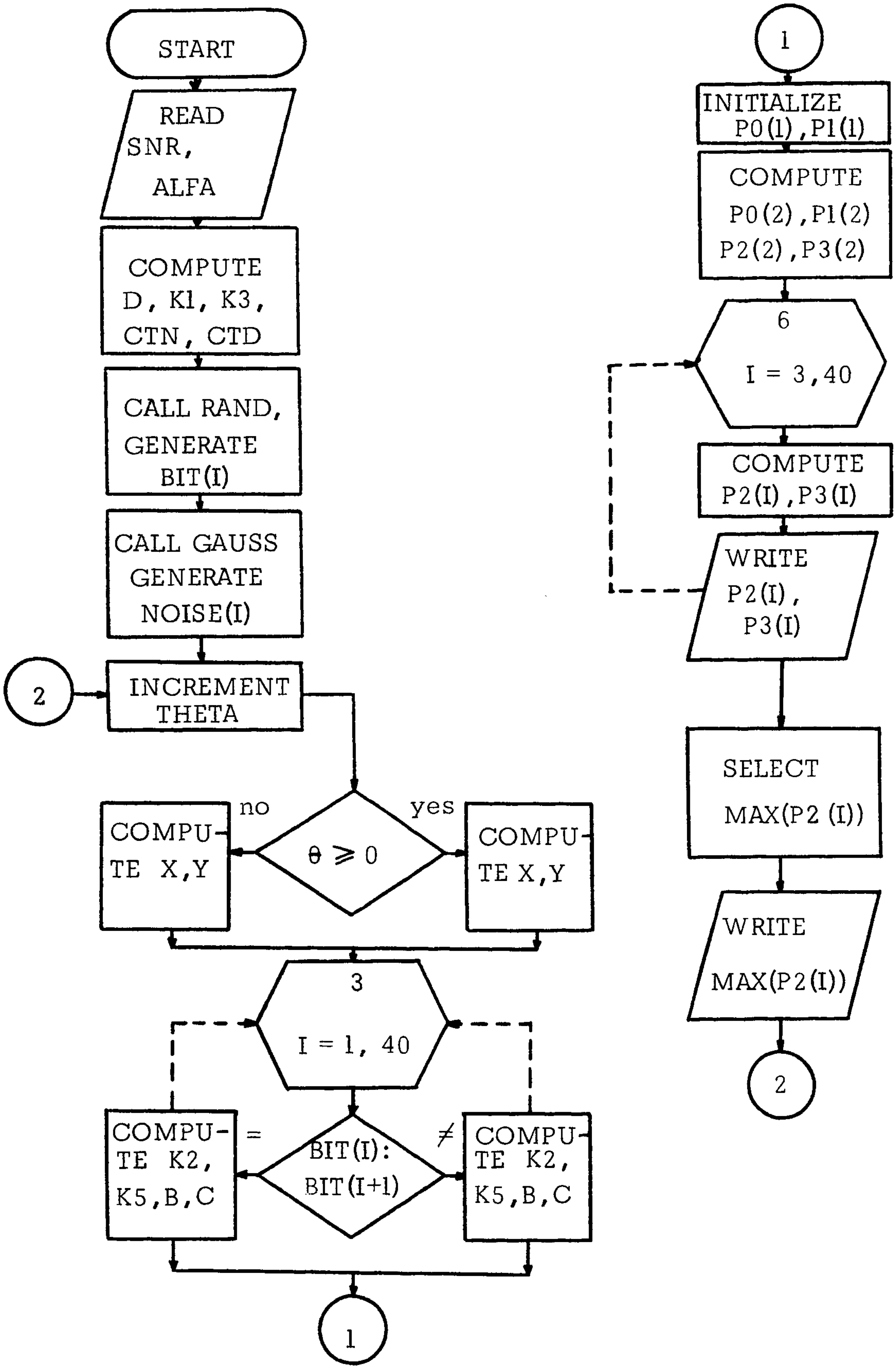



Fig. 3.6 Flow chart of ML synchronizer simulation program

Table 3.2 ML synchronizer simulation program

```
C  MLSYNC SIMULATION PROGRAM
        REAL  BIT(41),B(40),C(40),K1,K2(40),K3,K5(40)
        REAL  P0(2),P1(2),P2(40),P3(40),NOISE(40),LARGE
        SNR = 5.
        IX = 213711
        S = SQRT(1./SNR)
        ALFA = 0.1
        D = ALFA/3.*SNR
        CTN = SQRT(2.*(1.-2.*ALFA) + 2.*ALFA/3.)
        CTD = 2.*(1.-2.*ALFA) + 2.*ALFA/3.
        K1 = SQRT(SNR/(2.*3.1416*40.))
C
        DO 1  I=1,41
        CALL RAND(IX,IY,YFL)
  1     BIT(I) = SIGN(1.,YFL-.5)
C
        DO 2  I=1,40
  2     NOISE(I) = CTN*GAUSS(S)
C
        THETA = 0.0
  23    K3 = (ALFA/3.-.5)*SNR
        IF(THETA .LT. 0.0) GO TO 13
        X = .5-ALFA/1.5-THETA**2/(2.*ALFA)+THETA**3/(12.*
       1   ALFA**2)
        Y = -X
        GO TO 10
  13    X = .5-ALFA/1.5-THETA**2/(2.*ALFA)-THETA**3/(12.*
       2   ALFA**2)
        Y = -X
C
  10    DO 3  J=1,40
        IF(BIT(J)-BIT(J+1)) 4,5,4
  5     K2(J) = -(1.+NOISE(J))*.5*SNR
        K5(J) = K2(J) + K3
        B(J) = ((BIT(J)+BIT(J+1))*.5*.5+NOISE(J))*SNR
        C(J) = B(J)
        GO TO 3
  4     K2(J) = -((3.-4.*ALFA)/3.+NOISE(J))*.5*SNR
        K5(J) = K2(J)+K3
```

(continued on the next page)

```
  4     K2(J) = -((3.-4.*ALFA)/3.+NOISE(J))*.5*SNR
        K5(J) = K2(J)+K3
        B(J) = ((BIT(J)-BIT(J+1))*.5*X+NOISE(J))*SNR
        C(J) = ((BIT(J)-BIT(J+1))*.5*Y+NOISE(J))*SNR
  3     CONTINUE
C
        P0(1) = 0.5
        P1(1) = 0.5
        P2(1) = 0.0
        P3(1) = 0.0
        P0(2) = EXP(-C(1))*(P0(1)*EXP(-B(1)-D)+P1(1)*EXP(B(1)+D))*
       1   EXP(K5(2))*K1
        P1(2) = EXP( C(1))*(P0(1)*EXP(D-B(1))+P1(1)*EXP(B(1)-D))*
       2   EXP(K5(2))*K1
        P2(2) = 0.5*P0(2)+0.5*P1(2)
        P3(2) = 0.5*P0(2)-0.5*P1(2)
C
        DO 6  I=3,40
        J = I-1
        T = EXP(-C(J))*COSH(D+B(J))+EXP(C(J))*COSH(B(J)-D)
        U = -EXP(-C(J))*SINH(D+B(J))+EXP(C(J))*SINH(D-B(J))
        V = EXP(-C(J))*COSH(D+B(J))-EXP(C(J))*COSH(D-B(J))
        W = -EXP(-C(J))*SINH(D+B(J))-EXP(C(J))*SINH(D-B(J))
        P3(I) = (P2(J)*V+P3(J)*W)*EXP(K5(I))*K1
        P2(I) = (P2(J)*T+P3(J)*U)*EXP(K5(I))*K1
  6     CONTINUE
        WRITE(3,107)  (P2(I), I=1,40)
 107    FORMAT(//5X,'THE PROBABILITY P2 IS'/(10X(5E14.7)))
C
        LARGE = P2(1)
        J = 1
        DO 15  I=2,40
        IF(P2(I) .LE. LARGE) GO TO 15
        LARGE = P2(I)
        J = I
  15    CONTINUE
        WRITE(3,110) J,LARGE,THETA
 110    FORMAT(//5X,'N=',I2,5X,'P='F16.7,5X,'THETA='F7.3)
C
        IF(THETA .LT. 0.0) GO TO 14
        THETA = THETA+0.02
        IF(THETA .LE. ALFA) GO TO 23
```

(continued on the next page)

```
C
          THETA = 0.0
   14     THETA = THETA-0.02
          IF(THETA .GE. -ALFA)  GO TO 23
  999     STOP
          END
```

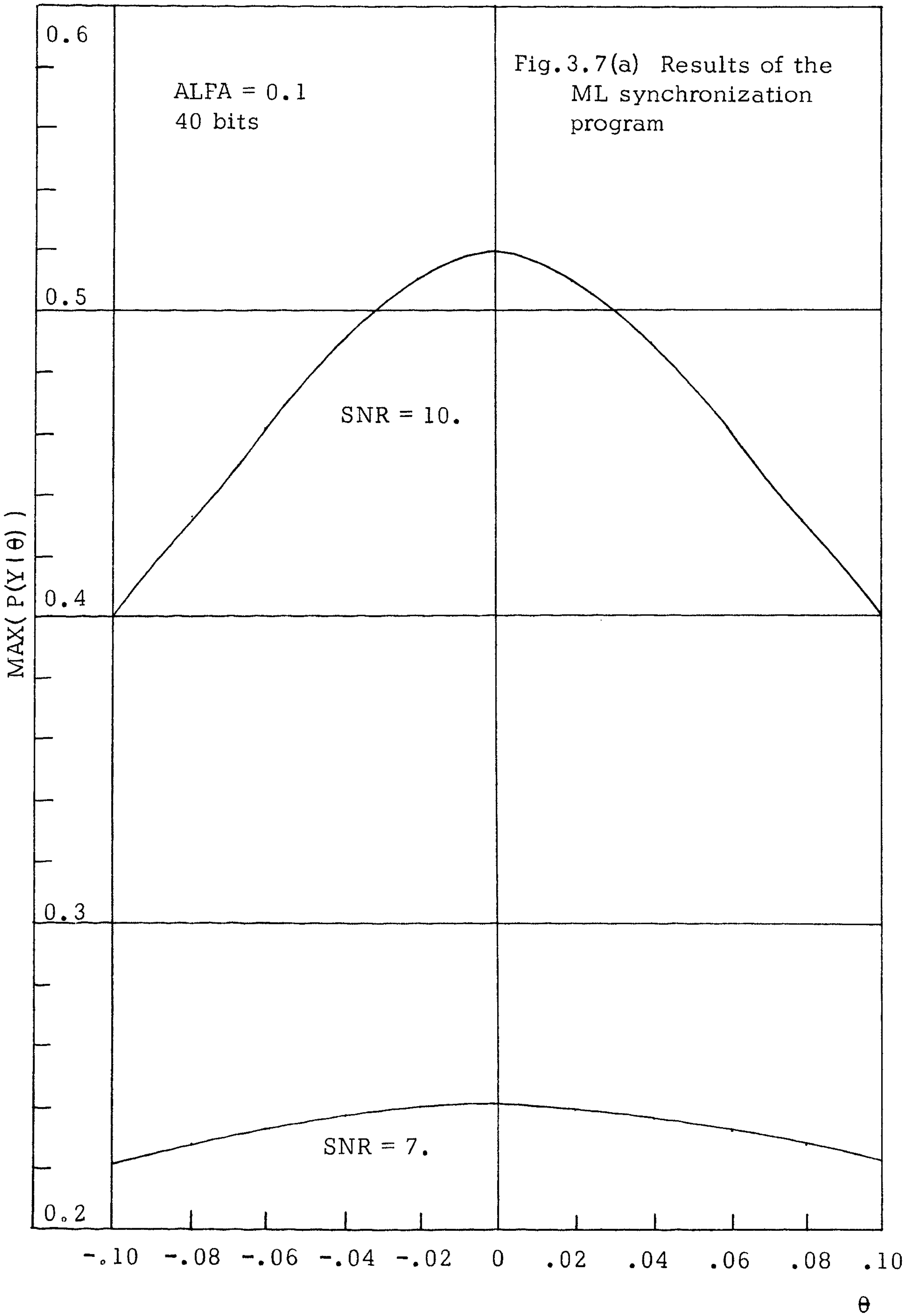
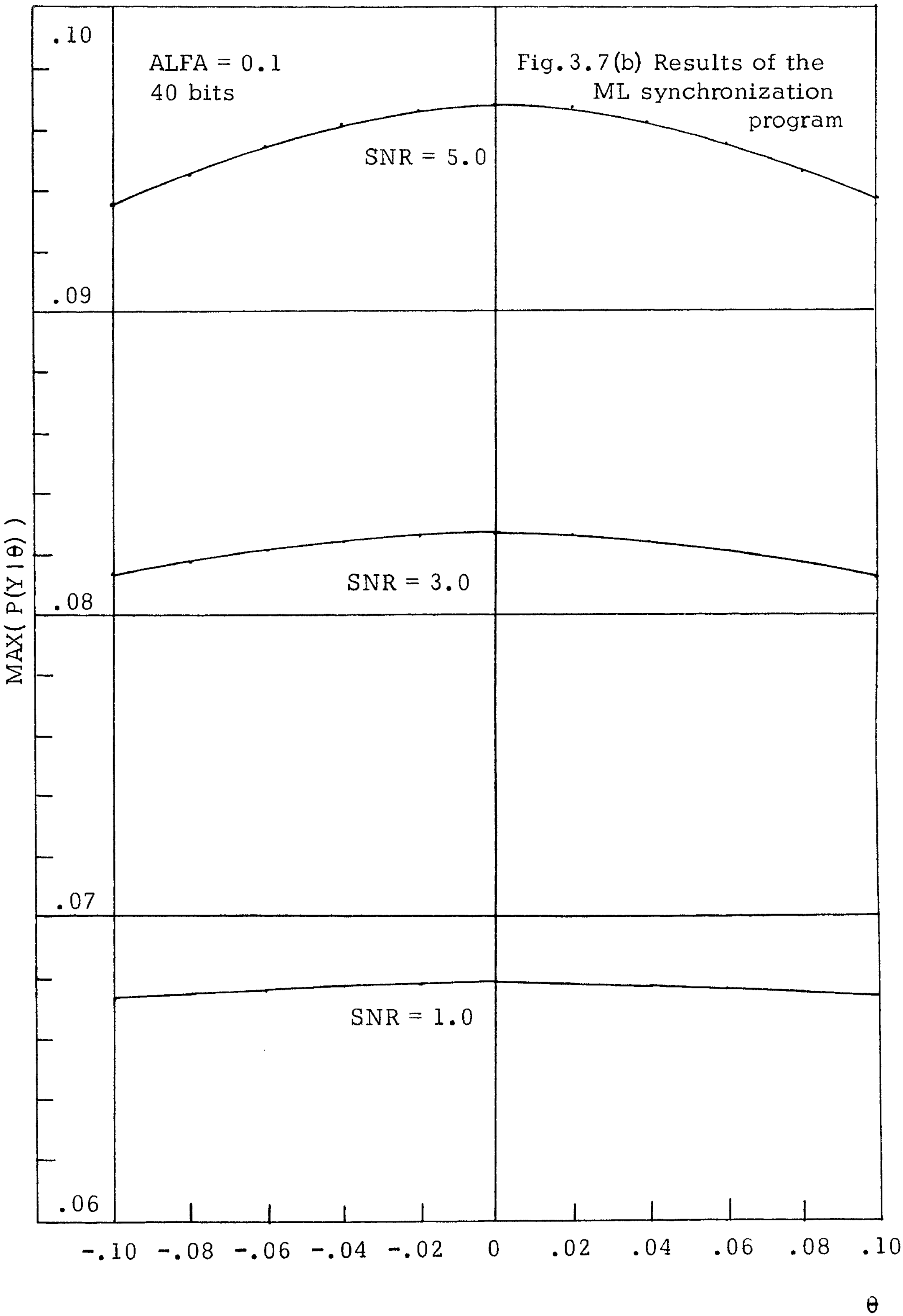



Fig. 3.7(a) Results of the ML synchronization program



ALFA = 0.1
40 bits
Fig. 3.7(b) Results of the ML synchronization program
SNR = 5.0
SNR = 3.0
SNR = 1.0
.10
.09
.08
.07
.06
MAX( P(Y|θ) )
-.10 -.08 -.06 -.04 -.02 0 .02 .04 .06 .08 .10
θ

# IV. DECISION-DIRECTED DETECTOR

## A. Introduction

The optimum synchronizer for the overlapping signals considered in Chapter III is difficult to implement without the aid of a special purpose digital computer. This chapter presents a practical approach called the decision-directed (DD) technique. By Monte Carlo simulation on a digital computer, the probability of error, $P_E$, versus signal-to-noise ratio (SNR) for the DD detector can be evaluated.

Basically, two steps are involved in this approach. First, the symbol most likely received is determined as a primary decision. Then, this decision is used to direct the detection process to obtain a better estimate of the symbol which will yield less probability of error. For overlapping signals, the DD technique can be roughly summarized in Fig.4.1. The post-bit detector consists of an ordinary matched filter and a sampler. The input signals are processed serially, one after the other, and the output is the primary detected value, $\hat{a}_p$. The function of the block "SHAPE" is to maintain a constant level until the next sampling instant. The output is then used to subtract the overlapping tail resulting from the next bit. With the subtraction of the overlapping head from the preceding bit, a present-bit detector is followed to find the final decision. The overlapping head and tail are illustrated by Fig.4.2.

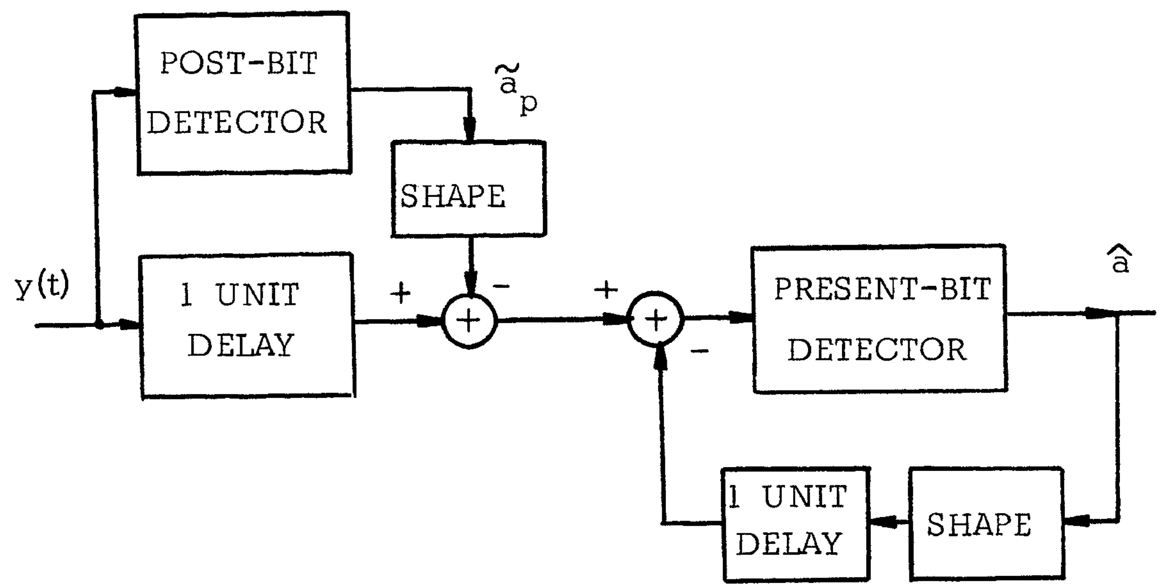



Fig.4.1 Decision-directed detector for overlapping signals

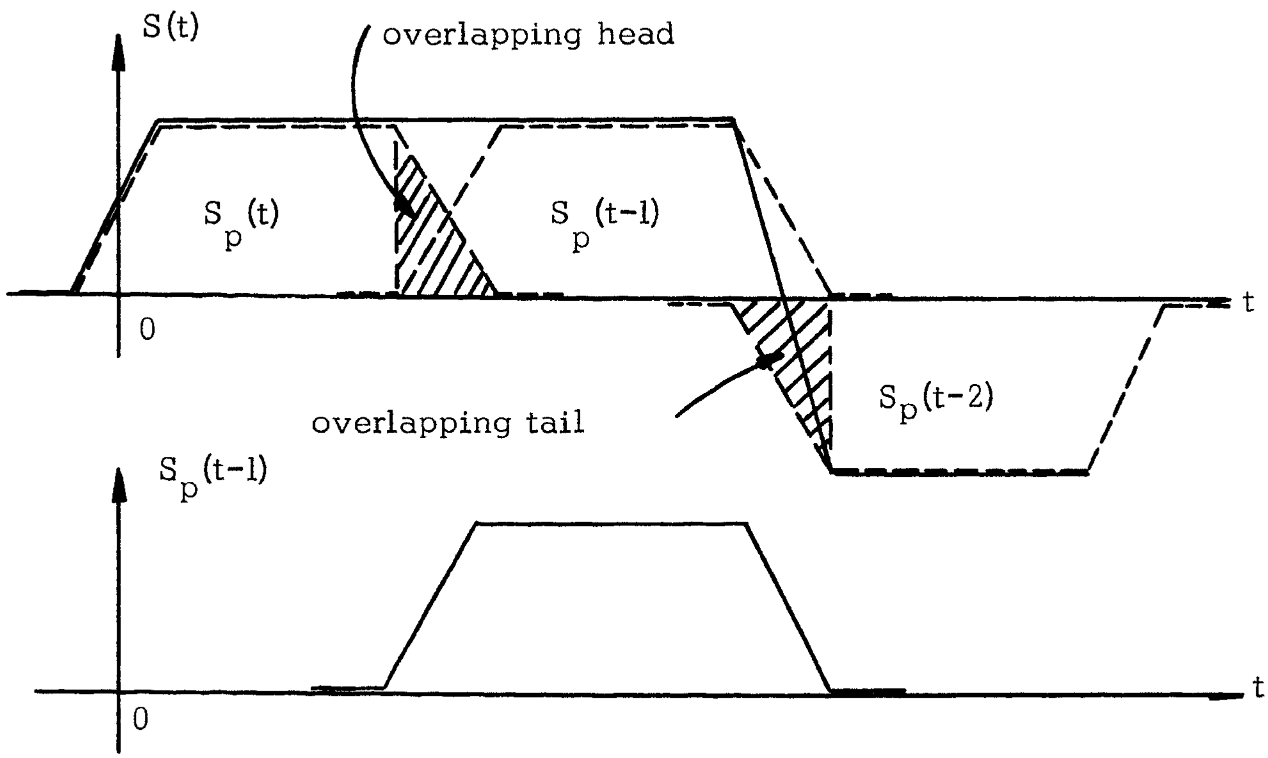



Fig.4.2 Received signal and the overlapping symbol

Since both detectors in Fig. 4.1 perform the same function, they can be placed in the front as a detector. The modified structure for the DD detector is shown in Fig. 4.3. Here the decision devices are replaced by two hard limiters. The constants $K_2$ and $K_3$ will be introduced in the following section.

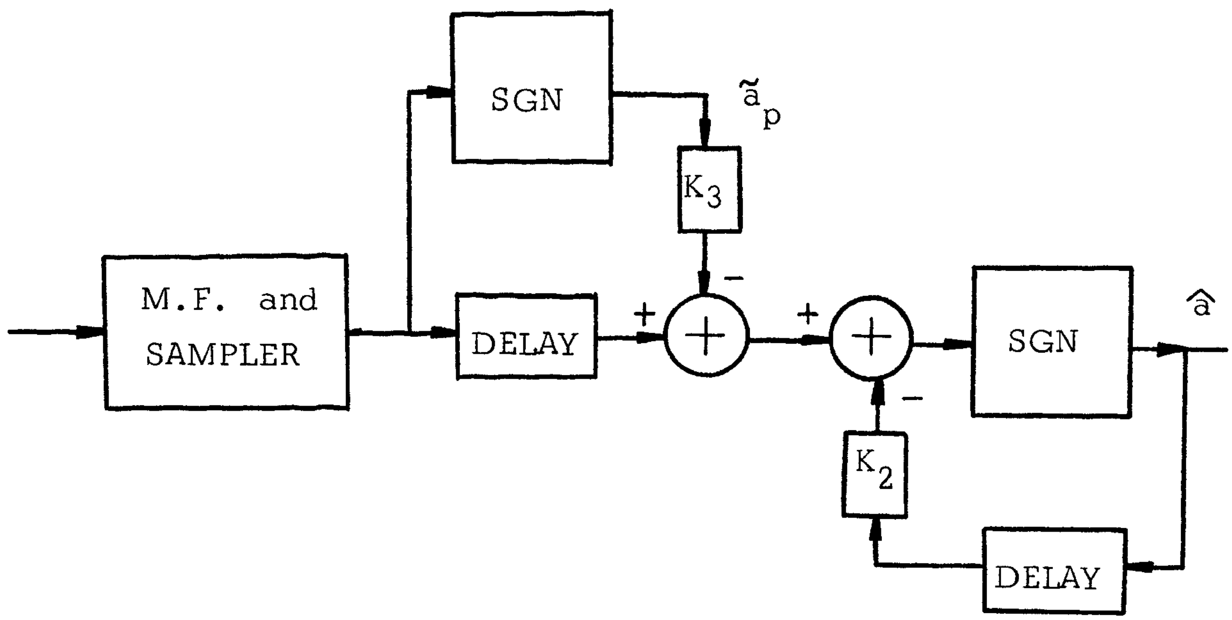



Fig. 4.3 Modified decision-directed detector

B. Mathematical Derivation

The binary communication system under consideration consists of two symbols, $S_p(t)$ and $-S_p(t)$, defined in Chapter III. The received signal is

$$y(t) = \sum_{k=1}^{m} a_k S_p(t-k) + n(t) \tag{4.1}$$

The matched filter has an impulse response

$$h(t) = S_p(1-t)$$

Thus the output of the matched filter is

$$z(t) = y(t) * h(t)$$

$$= \sum_{k=1}^{m} a_k S_p(t-k) * S_p(1-t) + n(t) * S_p(1-t) \qquad (4.2)$$

At the sampling instant $t = i$,

$$z(i) = \sum_{k=1}^{m} a_k S_p(i-k) * S_p(1-i) + n(i) * S_p(1-i) \qquad (4.3)$$

Due to the overlapping situation , $z(i)$ can be further written as

$$z(i) = K_2 a_{i-1} + K_1 a_i + K_3 a_{i+1} + K_4 n(t) \qquad (4.4)$$

where $K_2$ and $K_3$ are the areas when dealing with overlap of the ith bit with the (i-1)th bit and with the (i+1)th bit, respectively. $K_1$ is the area when the ith bit convolves with itself. $K_4$ is the noise coefficient to be defined later. These constants are functions of $\theta$, and are required in the simulation program. The calculation of $K_1$, $K_2$, and $K_3$ proceeds as follows.

$$K_2 = \int_{k-\alpha}^{k+1+\alpha} S_p(t-k)\, S_p(t-k+1)\, dt$$

$$= \int_{-\alpha}^{1+\alpha} (1/2 + t/2\alpha)\,(1/2 - t/2\alpha)\, dt = \alpha/3.$$

(4.5)

$$K_1 = \int_{-\alpha}^{1+\alpha} S_p^2(t)\, dt = \alpha/3 + (1 - 2\alpha) + \alpha/3$$

$$= 1 - 2\alpha/3, \text{ and} \qquad (4.6)$$

$$K_3 = K_2 = \alpha/3. \qquad (4.7)$$

To find $K_4$, we consider the root mean square of the correlator output when the signal is correlated with noise. That is

$$E\left\{\left[\int_{-\alpha}^{1+\alpha} S_p(t)\, n(t)\, dt\right]^2\right\}$$

$$= \iint_{-\alpha}^{1+\alpha} S_p(t_1)\, S_p(t_2)\, E\left\{n(t_1)\, n(t_2)\right\} dt_1\, dt_2$$

$$= B \int_{-\alpha}^{1+\alpha} S_p^2(t)\, dt = B(1 - 2\alpha/3) \qquad (4.8)$$

Where B is taken as the reciprocal of the input signal-to-noise ratio in the simulation program. The noise coefficient $K_4$ is then

$$K_4 = \sqrt{1 - 2\alpha/3} \ .$$

C. Simulation Program

In general, the desired output is a graph of $P_E$ versus SNR with the overlapping parameter $\alpha$ as a parameter. In order to examine how much improvement can be achieved by the DD technique, two error probabilities are tabulated in parallel. That is, for each bit, a primary decision by non-decision-directed measurement is made. At the end of m bits, the probability of error of the primary decision is computed. Then the final decision on each bit using the DD measurement is found. $P_E$ in this program is approximated by the ratio of the total number of erroneous bits to the total number of input bits.

The final detail to be considered is the DD algorithm. First, the primary decision stream from the matched filter is called ADETP(i), i=1,2,...,m. The next step is to combine ADETP(i+1) with the last final estimate, ADET(i-1), to form the present final estimate, ADET(i). Note that the first final estimate should be set to zero in order to start the feedback relationship.

A flow chart of the above procedure and the associated variable names used in this program are shown in Fig.4.4, and Table 4.1, respectively. The FORTRAN simulation program is shown in Table 4.2.

D. Results

In the program shown in the last section, 500 bits are generated by the subroutine RAND as the input to the DD detector. The resulting error counts are shown in Fig.4.5 for various $\alpha$'s. It can be seen that when SNR is greater than five, this system will give a better performance. When the overlapping parameter $\alpha$ is large, or the overlapping situation becomes worse, the system tends to correct more errors.

Table 4.1 Variable names used in DD detector simulation program

| | |
|---|---|
| SNR | Signal-to-Noise power Ratio |
| ALFA | the overlapping parameter, ranged from 0.1 to 0.5 |
| IX | starting value of the subroutine RAND, an odd integer (213711 is used in this program) |
| RAND | uniform random number generator subroutine which generates random numbers uniformly distributed from 0 to 1 |
| BIT | random bit stream, input data words |
| MFSIG | matched filter output sampled at the end of the bit |
| S | variance value of the subroutine GAUSS(S) |
| GAUSS | Gaussian random number generator subroutine with S as the desirable standard deviation |
| ADETP | primary detected value of the bit stream |
| ADET | final detected value of the bit stream |
| P1 | primary probability of bit error |
| P2 | final probability of bit error |

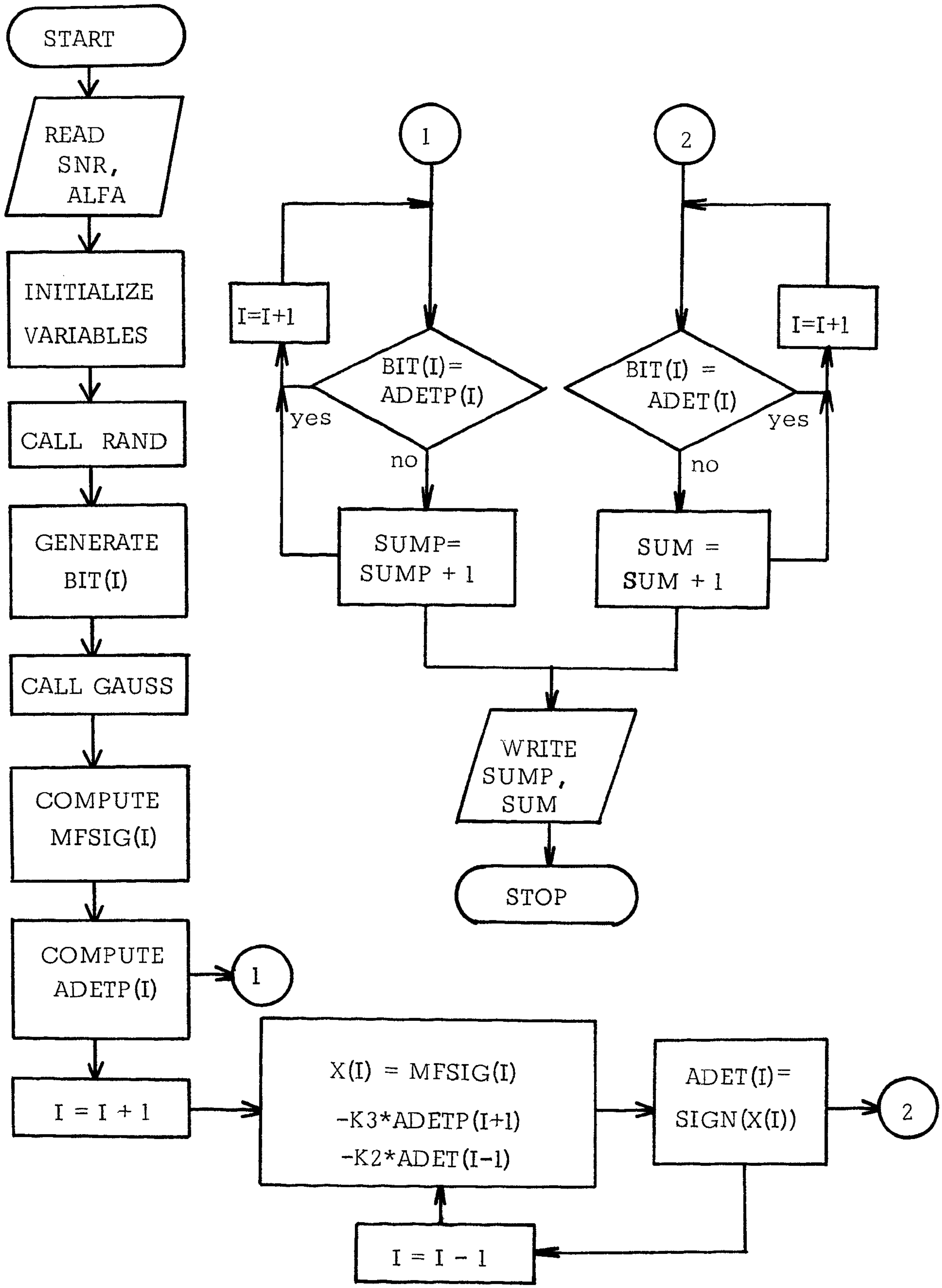



Fig.4.4 Flow chart of DD detector simulation program

Table 4.2 DD detector simulation program

```
      REAL MFSIG(503),K1,K2,K3,K4,BIT(503),ADETP(503),ADET(
     1    503)
      IX = 213711
 10   READ(1,99,END=999) SNR,ALFA
 99   FORMAT(2F5.2)
C
      K1 = 1.-2.*ALFA/3.
      K2 = ALFA/3.
      K3 = K2
      K4 = SQRT(1.-ALFA/1.5)
      S = SQRT(1./SNR)
C
      BIT(1) = 0.
      DO 1  I = 2,503
      CALL RAND(IX,IY,YFL)
 1    BIT(I) = SIGN(1., YFL-.5)
C
      MFSIG(1) = 0.
      DO 2  I=2,502
 2    MFSIG(I)=K2*BIT(I-1)+K1*BIT(I)+K3*BIT(I+1)+K4*GAUSS(S)
C
      ADETP(1) = 0.
      ADET(1) = 0.
      ADETP(502) = SIGN(1., MFSIG(502))
      DO 3  I=2,502
 3    ADETP(I) = SIGN(1., MFSIG(I))
C
      DO 4  I=2,501
 4    ADET(I) = SIGN(1., MFSIG(I)-K3*ADETP(I+1)-K2*ADET(I-1))
C
      SUMP = 0.
      SUM = 0.
      DO 5  I=2,501
      IF(BIT(I) .EQ. ADETP(I)) GO TO 6
      SUMP = SUMP+1
 6    IF(BIT(I) .EQ. ADET(I)) GO TO 5
      SUM = SUM+1.
 5    CONTINUE
      P1 = SUMP/500.
      P2 = SUM/500.
      WRITE(3,106)  SNR,ALFA,P1,P2
 106  FORMAT(//5X,'SNR='F5.2,5X,'ALFA='F5.2,5X,'P1='F10.5,
     2    5X,'P2='F10.5)
 999  STOP
      END
```

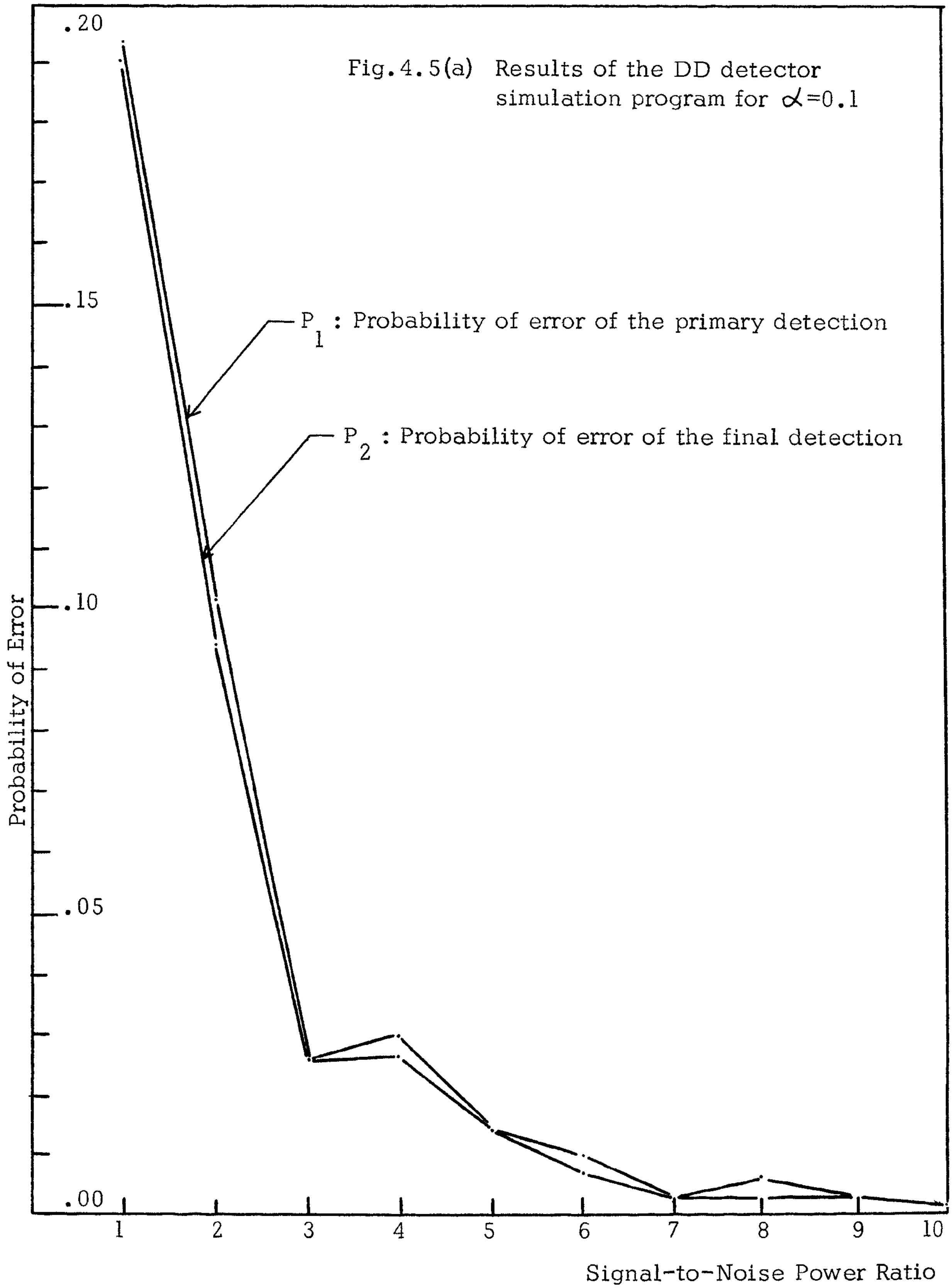



Fig.4.5(a) Results of the DD detector simulation program for $\alpha=0.1$

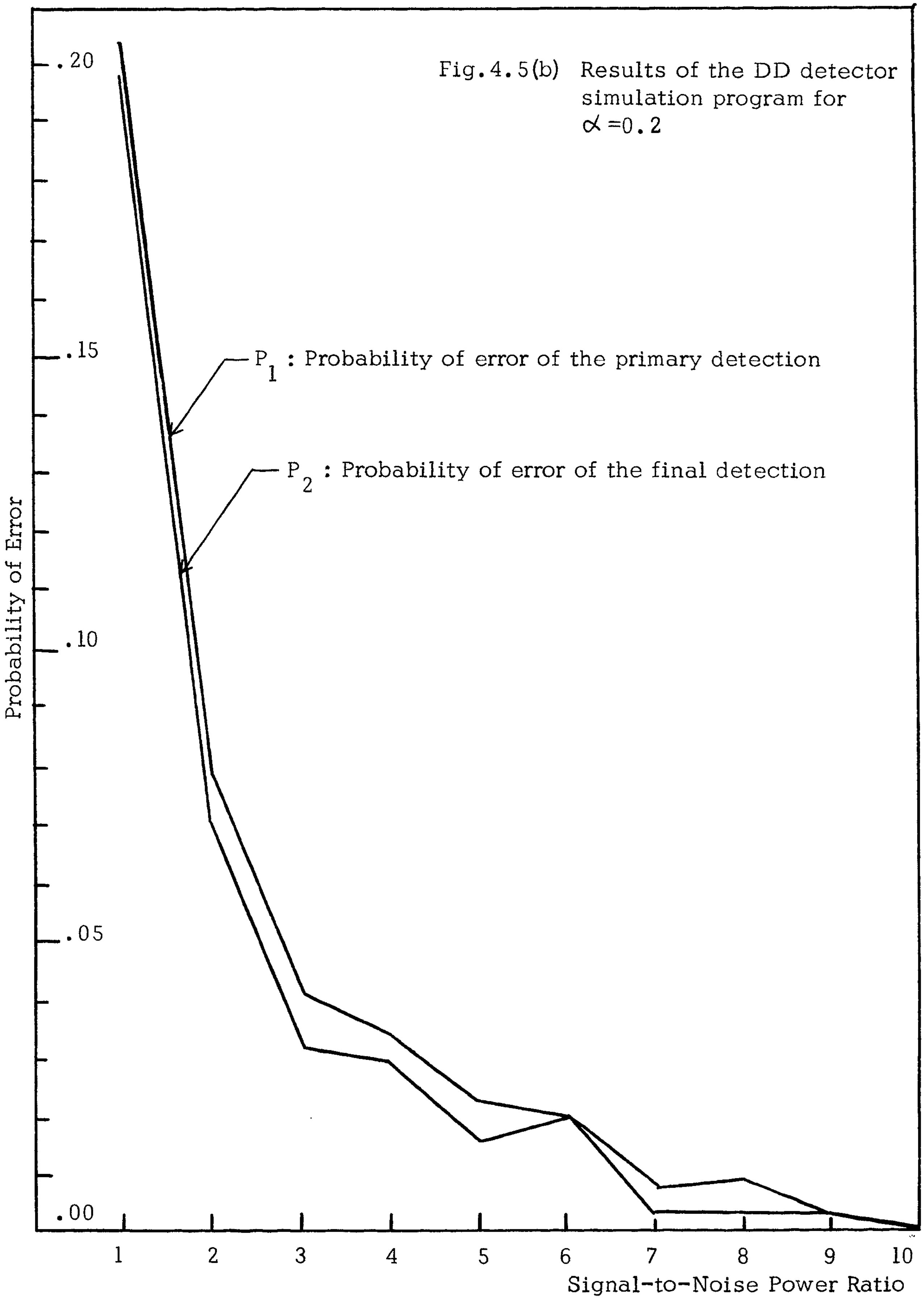



Fig.4.5(b) Results of the DD detector simulation program for $\alpha=0.2$

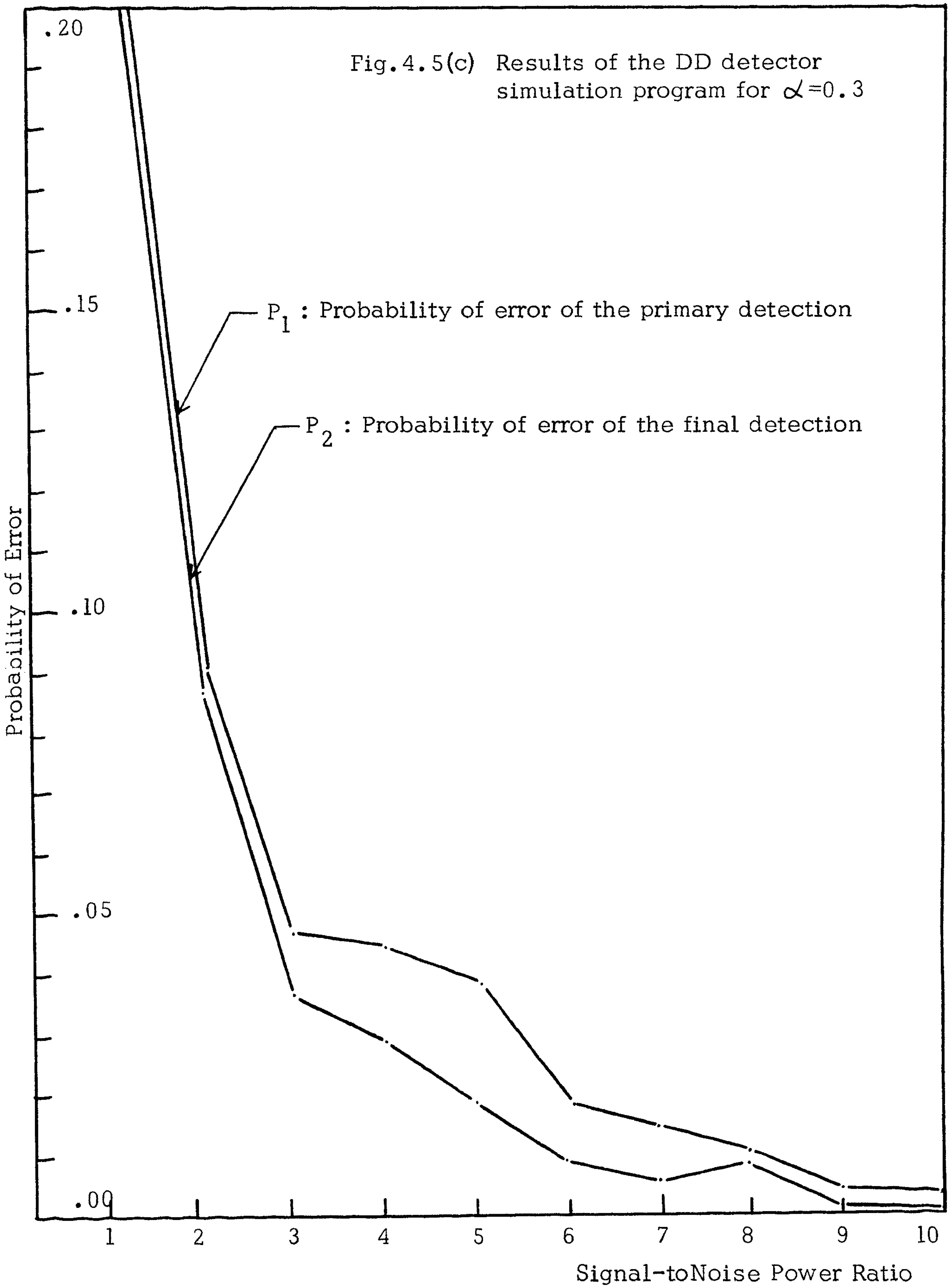



Fig. 4.5(c) Results of the DD detector simulation program for $\alpha=0.3$

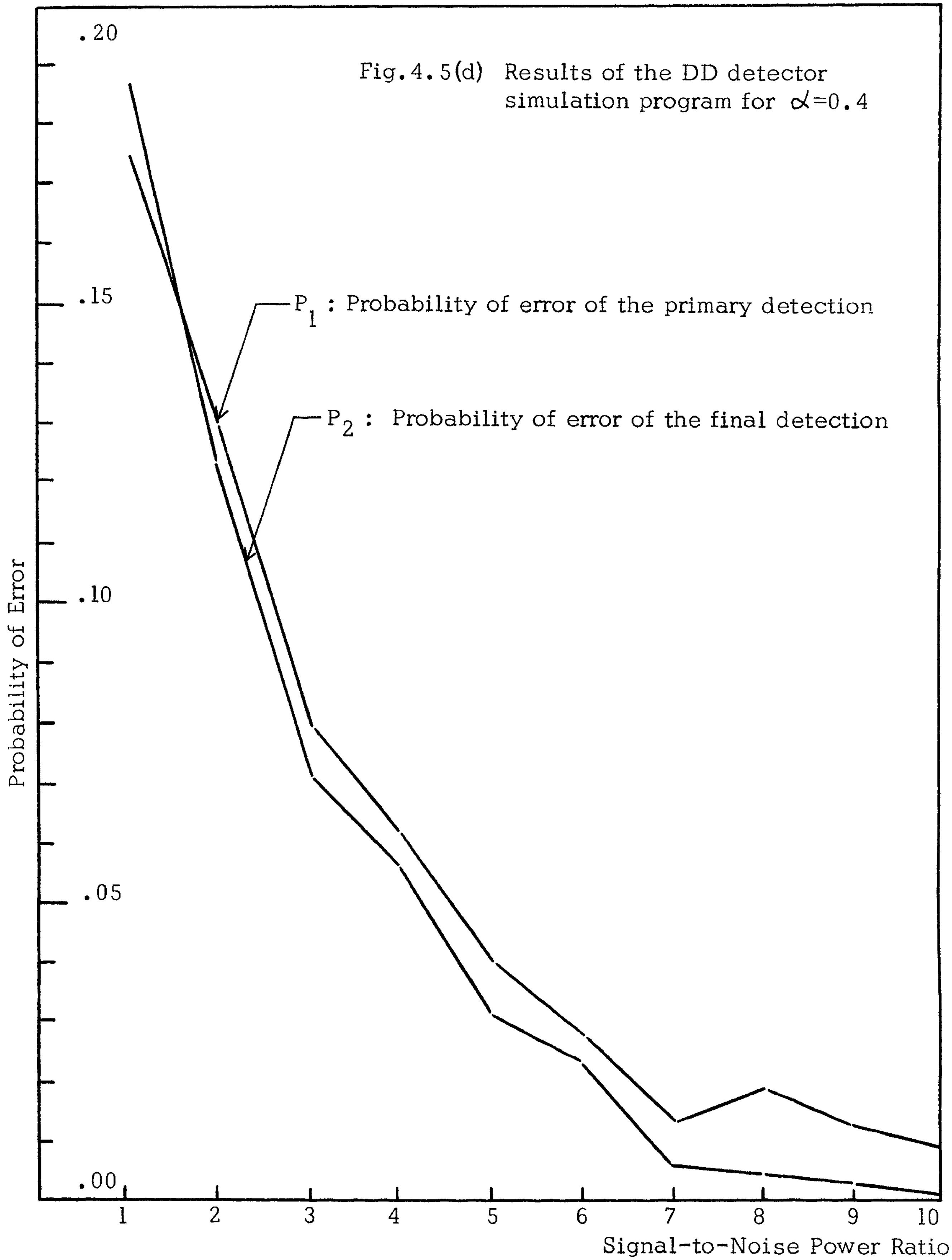



Fig. 4.5(d) Results of the DD detector simulation program for $\alpha$=0.4

E. Analytical Results

In this section, calculation of $P_E$ for the DD detector is considered. The block diagram of a DD detector is redrawn in Fig. 4.6.

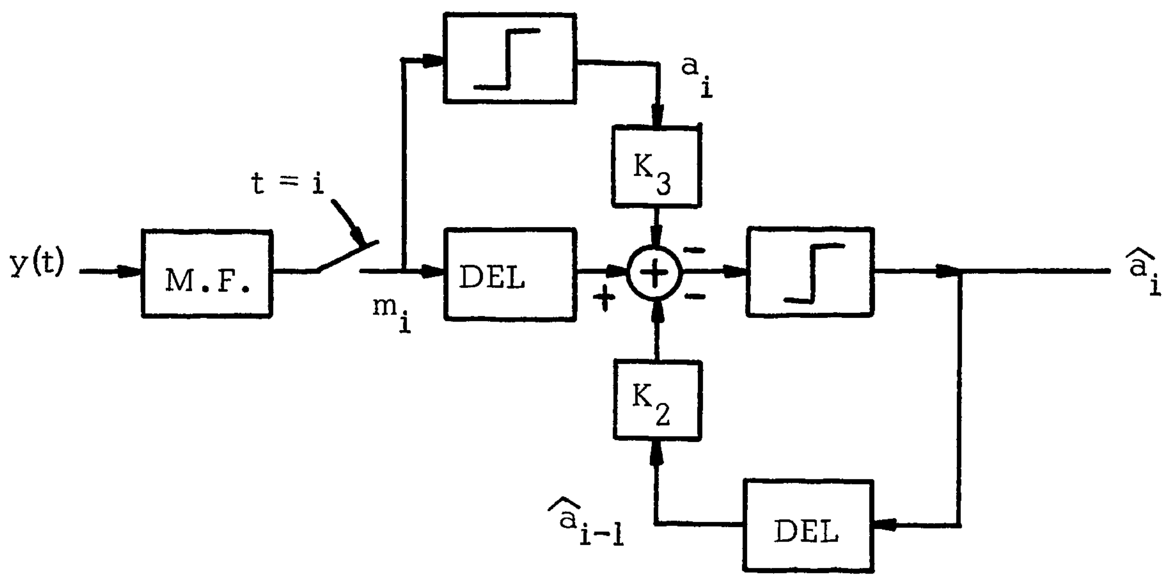



Fig. 4.6. Block diagram of the DD detector

The output of the matched filter and sampler is

$$m_i = K_2 a_{i-1} + K_1 a_i + K_3 a_{i+1} + K_4 n_i \tag{4.9}$$

where
$$\begin{cases} K_1 = 1 - (2/3)\alpha \\ K_2 = K_3 = (1/3)\alpha \text{ , and} \\ K_4 = \sqrt{1 - (2/3)\alpha} \text{ .} \end{cases}$$

Because of the overlapping situation, three symbols are involved in determining the probability of bit error of the primary detection. Let $A = (a_{i-1}, a_i, a_{i+1})$ ; the eight possible combinations of the sequence are tabulated in Table 4.3. The $P_E$ of the primary detection is

$$\tilde{P} = P\left\{m_i<0 \,\middle|\, a_i=1\right\} P\left\{a_i=1\right\} + P\left\{m_i>0 \,\middle|\, a_i=-1\right\} P\left\{a_i=-1\right\} \qquad (4.10)$$

Table 4.3 Eight possible sequences and mean values for finding the primary $P_E$

| | $a_{i-1}$ | $a_i$ | $a_{i+1}$ | $E(m_i) = \mu_i$ | $var(m_i) = \sigma$ |
|---|---|---|---|---|---|
| $A_1$ | 1 | 1 | 1 | 1 | |
| $A_2$ | -1 | 1 | 1 | $1 - (2/3)\alpha$ | |
| $A_3$ | 1 | 1 | -1 | $1 - (2/3)\alpha$ | $\sigma = \sqrt{B(1-(2/3)\alpha)}$ |
| $A_4$ | -1 | 1 | -1 | $1 - (4/3)\alpha$ | |
| $A_5$ | 1 | -1 | 1 | $-1 + (4/3)\alpha$ | $B = 1/SNR$ |
| $A_6$ | -1 | -1 | 1 | $-1 + (2/3)\alpha$ | |
| $A_7$ | 1 | -1 | -1 | $-1 + (2/3)\alpha$ | |
| $A_8$ | -1 | -1 | -1 | -1 | |

When we consider $a_i=1$, there are four possible sequences, $(A_1, A_2, A_3, A_4)$, involved in finding the term $P\left\{m_i<0 \mid a_i=1\right\}$. Thus,

$$P\left\{m_i<0 \mid a_i=1\right\} = (1/4) \sum_{j=1}^{4} \int_{-\infty}^{0} (1/\sqrt{2\pi}\sigma) \exp\left\{-\frac{(x-\mu_j)^2}{2\sigma^2}\right\} dx,$$

where $\mu_i \triangleq E\left\{m_i\right\}$. Treating the case for $a_i=-1$ in the same manner, we have

$$\widetilde{P} = (1/2)\left\{(1/4) \sum_{j=1}^{4} \int_{-\infty}^{0} (1/\sqrt{2\pi}\sigma) \exp\left[-\frac{(x-\mu_j)^2}{2\sigma^2}\right] dx\right.$$

$$\left. + (1/4) \sum_{j=5}^{8} \int_{0}^{\infty} (1/\sqrt{2\pi}\sigma) \exp\left[-\frac{(x-\mu_j)^2}{2\sigma^2}\right] dx\right\}$$

$$= (1/8)\left\{\sum_{j=1}^{4} \phi(-\mu_j/\sigma) + \sum_{j=5}^{8} \phi(\mu_j/\sigma).\right\} \tag{4.11}$$

$$\text{where} \quad \phi(a) = \int_{-\infty}^{a} (1/\sqrt{2\pi}) \cdot \exp(-x^2/2)\, dx \tag{4.12}$$

Substituting mean values, (4.11) can be simplified as follows.

$$\tilde{P} = (1/4)\left\{ \phi(-1/\sigma) + 2\phi(-(1-2\alpha/3)/\sigma) + \phi(-(1-3\alpha/4)/\sigma) \right\}$$

(4.13)

The probability of the final detection is found as follows. Since the final decision $\hat{a}_i$ depends on the previous final decision $\hat{a}_{i-1}$ and on the primary decision on (i+1)th bit, $\tilde{a}_{i+1}$ , we have

$$\hat{a}_i = m_i - K_3 \tilde{a}_{i+1} - K_2 \hat{a}_{i-1} \qquad (4.14)$$

where $m_i = K_2 a_{i-1} + K_1 a_i + K_3 a_{i+1} + K_4 n_i$

Hence, the probability of error is described by a recursive relation. Let us define

$$P_i \triangleq \left\{ P\ \hat{a}_i < 0 \,\middle|\, a_i, \tilde{a}_{i+1}, \hat{a}_{i-1} \right\} P\left\{a_i\right\} P\left\{\tilde{a}_{i+1}\right\} P\left\{\hat{a}_{i-1}\right\} \qquad (4.15)$$

Rewriting (4.14) by substituting $m_i$ in the equation, we have

$$\hat{a}_i = K_2 (a_{i-1} - \hat{a}_{i-1}) + K_1 a_i + K_3 (a_{i+1} - \tilde{a}_{i+1}) + K_4 n_i \qquad (4.16)$$

If the primary decision on the symbol $a_i$ is correct, the final output of the detector will be just a constant, $K_1$, times $a_i$. It can be seen from the simulated results that at high SNR the probability of error for the final detection decreases faster than that of the primary detection.

Table 4.4 Possible sequences and mean values for finding the final $P_E$

| $a_i$ | | $E\left[\hat{a}_i\right] \triangleq \eta_i$ |
|---|---|---|
| 1 | $a_{i-1} = \hat{a}_{i-1}, \ a_{i+1} = \tilde{a}_{i+1}$ | $\eta_1 = K_1$ |
| 1 | $a_{i-1} \neq \hat{a}_{i-1}, \ a_{i+1} = \tilde{a}_{i+1}$ | $\eta_2 = 2K_2 a_{i-1} + K_1$ |
| 1 | $a_{i-1} = \hat{a}_{i-1}, \ a_{i+1} \neq \tilde{a}_{i+1}$ | $\eta_3 = K_1 + 2K_3 a_{i+1}$ |
| 1 | $a_{i-1} \neq \hat{a}_{i-1}, \ a_{i+1} \neq \tilde{a}_{i+1}$ | $\eta_4 = 2K_2 a_{i-1} + K_1 + 2K_3 a_{i+1}$ |
| -1 | $a_{i-1} = \hat{a}_{i-1}, \ a_{i+1} = \tilde{a}_{i+1}$ | $\eta_5 = -K_1$ |
| -1 | $a_{i-1} \neq \hat{a}_{i-1}, \ a_{i+1} = \tilde{a}_{i+1}$ | $\eta_6 = 2K_2 a_{i-1} - K_1$ |
| -1 | $a_{i-1} = \hat{a}_{i-1}, \ a_{i+1} \neq \tilde{a}_{i+1}$ | $\eta_7 = -K_1 + 2K_3 a_{i+1}$ |
| -1 | $a_{i-1} \neq \hat{a}_{i-1}, \ a_{i+1} \neq \tilde{a}_{i+1}$ | $\eta_8 = 2K_2 a_{i-1} - K_1 + 2K_3 a_{i+1}$ |

The probability of error for the final detection, $P_i$, is

$$
\begin{aligned}
P_i = \; & P\left\{\hat{a}_i < 0 \,\middle|\, a_i=1, \tilde{a}_{i+1}=a_{i+1}, \hat{a}_{i-1}=a_{i-1}\right\} (1/2)\ (1-\tilde{P})\ (1-P_{i-1}) \\
& + P\left\{\hat{a}_i < 0 \,\middle|\, a_i=1, \tilde{a}_{i+1}=a_{i+1}, \hat{a}_{i-1}\neq a_{i-1}\right\} (1/2)\ (1-\tilde{P})\ P_{i-1} \\
& + P\left\{\hat{a}_i < 0 \,\middle|\, a_i=1, \tilde{a}_{i+1}\neq a_{i+1}, \hat{a}_{i-1}=a_{i-1}\right\} (1/2)\ \tilde{P}\ (1-P_{i-1}) \\
& + P\left\{\hat{a}_i < 0 \,\middle|\, a_i=1, \tilde{a}_{i+1}\neq a_{i+1}, \hat{a}_{i-1}=a_{i-1}\right\} (1/2)\ \tilde{P}\ P_{i-1} \\
& + \text{four terms for } a_i = -1. \qquad (4.17)
\end{aligned}
$$

Setting $1 - P_{i-1} = Q$, $P_{i-1} = P$, $1 - \tilde{P} = \tilde{Q}$, we have

$$P_i = (Q\tilde{Q}/2) \int_{-\infty}^{0} N(\eta_1, \sigma)\, dx + (P\tilde{Q}/2) \int_{-\infty}^{0} N(\eta_2, \sigma)\, dx$$

$$+ (\tilde{P}Q/2) \int_{-\infty}^{0} N(\eta_3, \sigma)\, dx + (P\tilde{P}/2) \int_{-\infty}^{0} N(\eta_4, \sigma)\, dx$$

$$+ (Q\tilde{Q}/2) \int_{0}^{\infty} N(\eta_5, \sigma)\, dx + (P\tilde{Q}/2) \int_{0}^{\infty} N(\eta_6, \sigma)\, dx$$

$$+ (Q\tilde{P}/2) \int_{0}^{\infty} N(\eta_7, \sigma)\, dx + (P\tilde{Q}/2) \int_{0}^{\infty} N(\eta_8, \sigma)\, dx. \tag{4.18}$$

where the notation $N(\eta_i, \sigma)$ indicates the density function of a Gaussian distribution having a mean value $\eta_i$, and a variance $\sigma^2$. Now we can substitute the values for $\eta_i$, $i = 1, 2, \ldots, 8$. and write $P_i$ as the function of the $\emptyset$-function defined by (4.12). Note that $\emptyset(-x) = 1 - \emptyset(x)$. (4.18) can be further simplified by using this identity.

Therefore,

$$P_i = \left[Q\tilde{Q}\right] \phi(-K_1/\sigma) + (P\tilde{Q}/2)\left[\phi(-(K_1-2K_2)/\sigma) + \phi(-(K_1+2K_2)/\sigma)\right]$$

$$+ (Q\tilde{P}/2)\left[\phi(-(K_1 - K_3)/\sigma) + \phi(-(K_1 + K_3)/\sigma)\right]$$

$$+ (P\tilde{P}/2)\left[\phi(-(K_1 - 2K_2 - 2K_3)/\sigma) + \phi(-(K_1 + 2K_2 + 2K_3)/\sigma)\right.$$

$$\left. + \phi(-(K_1 - 2K_2 + 2K_3)/\sigma) + \phi(-(K_1 + 2K_2 - 2K_3)/\sigma)\right] \tag{4.19}$$

The analytical results of the primary $P_E$ and the final $P_E$ can be evaluated by a computer program shown in Table 4.5. The results are plotted with the simulation data found in the last section to see how closely they are related. It is seen from Fig. 4.7 that when $\alpha$=0.3 and 0.4, the analytical results and the simulation results are in good agreement.

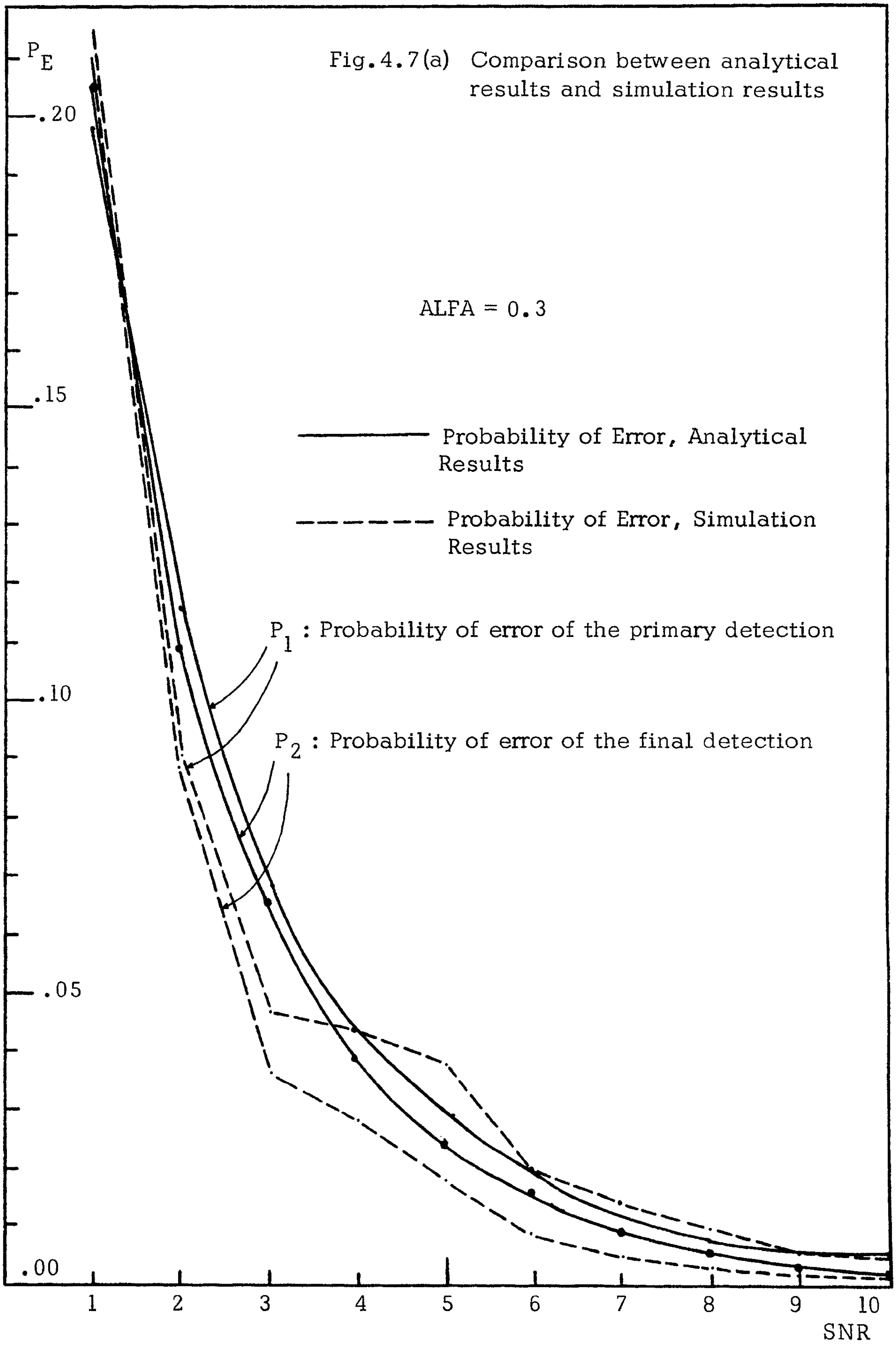
Table 4.5 Program for evaluating the primary $P_E$ and the final $P_E$

```
      REAL K1,K2,K3,P(10),Q(10)
 99   READ(1,100,END=999) SNR,ALFA
 100  FORMAT(2F5.2)
C
      K1 = 1.-2.*ALFA/3.
      K2 = ALFA/3.
      K3 = K2
      SIGMA = SQRT((1.-2.*ALFA/3.)/SNR)
      A = -1./SIGMA
      B = -K1/SIGMA
      C = -(1.-4.*ALFA/3.)/SIGMA
C COMPUTE THE PRIMARY PROBABILITY OR ERROR
      P1 = 0.25*(PHI(A) + 2.*PHI(B) + PHI(C))
      WRITE(3,101) SNR,ALFA,P1
 101  FORMAT(/5X'SNR='F5.2,5X,'ALFA='F5.2,5X,'PRIMAY P(E) IS',
     1   F16.7)
C
      Q1 = 1.-P1
      P(1) = 0.0
      Q(1) = 1.-P(1)
      C1 = PHI(-K1/SIGMA)
      C2 = PHI(-(K1-2.*k2)/SIGMA) + PHI(-(K1+2.*K2)/SIGMA)
      C3 = PHI(-(K1-K3)/SIGMA) + PHI(-(K1+K3)/SIGMA)
      C4 = PHI(-(K1-2.*K2-2.*K3)/SIGMA) + PHI(-(K1+2.*K2+2.*K3)
     2    /SIGMA) + PHI(-(K1-2.*K2+2.*K3)/SIGMA)+PHI(-(K1+2.*K2
     3    -2.*K3)/SIGMA)
C COMPUTE THE FINAL PROBABILITY OF ERROR
      DO 2 I=2,10
      P(I) = Q1*Q(I-1)*C1 + Q1*P(I-1)*0.5*C2 + P1*Q(I-1)*0.5*C3 +
     4     Q1*P(I-1)*0.25*C4
      Q(I) = 1.-P(I)
      WRITE(3,102) I,P(I)
 102  FORMAT(/15X,'P(',I2,')=',F16.7)
 2    CONTINUE
      GO TO 99
 999  STOP
      END
```

(continue on the next page)

(Table 4.5 continued)

```
      FUNCTION  PHI(X)
      AX = ABS(X)
      T = 1.0/(1.0+0.2316419*AX)
      D = 0.3989423*EXP(-X*X/2.0)
      P = 1.0-D*T*((((1.330274*T-1.821256)*T+1.781478)*T-
     1   0.3565638)*T+0.3193815)
      PHI = P
      IF(X) 1,2,2
 1    PGI = 1.0-P
 2    RETURN
      END
/DATA
/END
```



P_E
.20
.15
.10
.05
.00
Fig.4.7(a) Comparison between analytical results and simulation results
ALFA = 0.3
Probability of Error, Analytical Results
Probability of Error, Simulation Results
P1 : Probability of error of the primary detection
P2 : Probability of error of the final detection
1
2
3
4
5
6
7
8
9
10
SNR

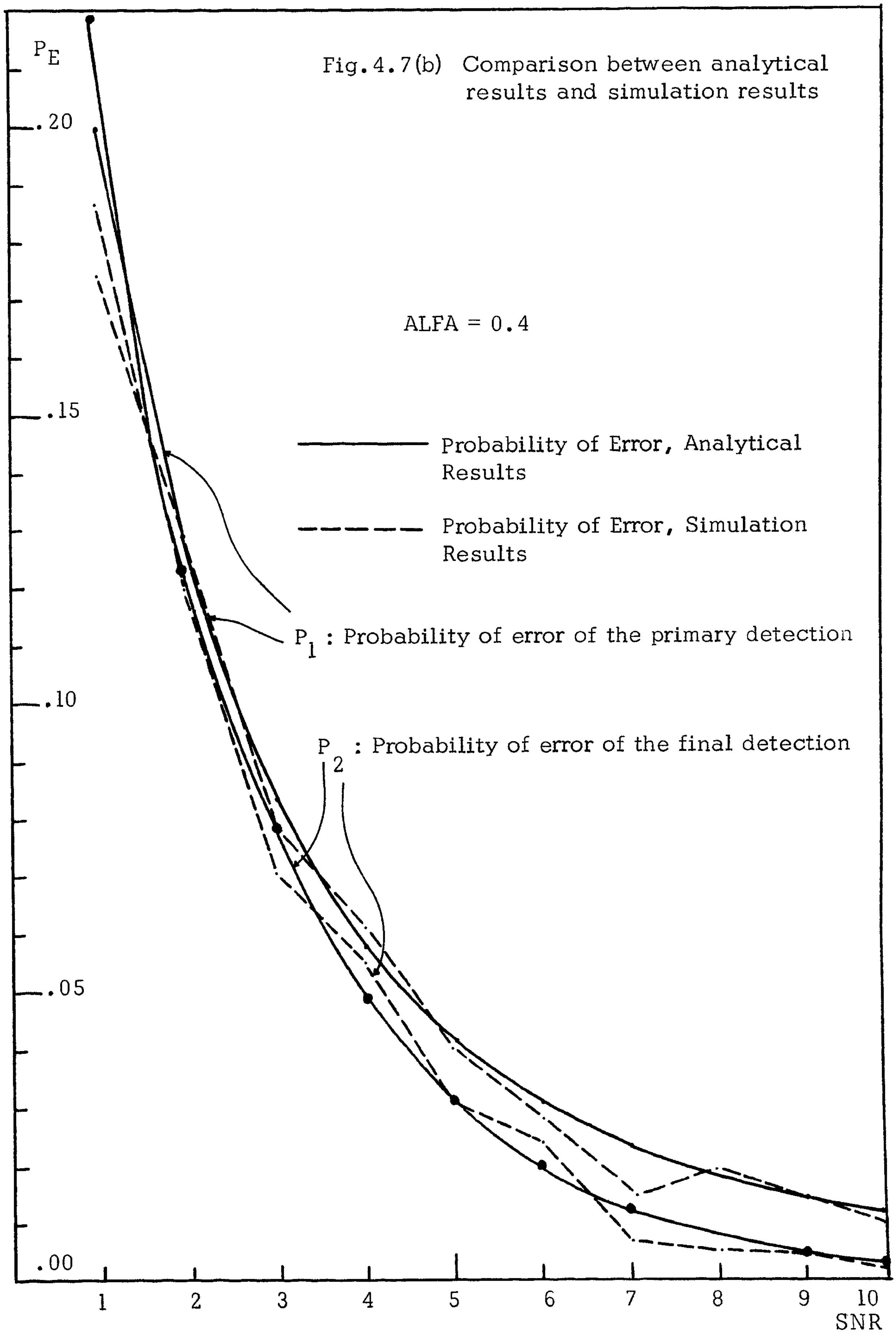



Fig.4.7(b) Comparison between analytical results and simulation results

# V. ABSOLUTE VALUE BIT SYNCHRONIZER

## A. Introduction

The purpose of this chapter is to present the steady-state phase-noise performance of an Absolute Value Bit Synchronizer (AVBS) for overlapping signals. A functional block diagram of AVBS is given by Fig. 5.1, which includes following portions: (1) a Matched Derivative Filter (MDF), which is a filter matched to the derivative of the received signal, (2) a Transition Detector (TD) as defined in Chapter III, and (3) an Averager (A) which computes the average of the output samples of the multiplier.

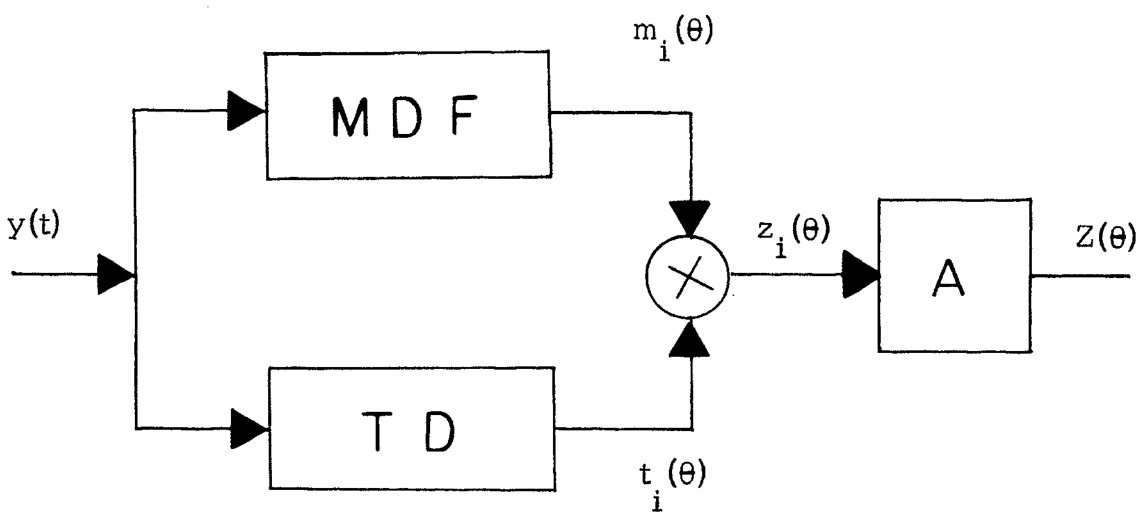



Fig. 5.1 AVBS sturcture

The proposed circuitry is based on the fact that the derivative of the overlapping signals can be used to estimate the phase of the received signal. An error signal $z_i(\theta)$ is generated by the product of the MDF output, $m_i(\theta)$, and the TD output, $t_i(\theta)$. Ideally, in the absence of noise, the operation of this system is as follows. For a given phase offset $\theta$, (1) the MDF output is $\pm\, m_i(\theta)$, a function of the phase offset $\theta$, when there is a transition; the output is $\pm 2\alpha$, when there is no transition. (2) the TD output is 1, or -1 if there is a transition and the TD output is zero when there is no transition. The sign of the above expressions depend on whether the transition is trom a positive pulse to a negative or vice versa. Fig. 5.3(a),(b), and (c) show the results of the system when different $\theta$'s are assumed at the input.

After averaging the error signals over many bit intervals, the output of the Averager is sent to a voltage-controlled oscillator (VCO). The output of the VCO is an estimate of the phase $\theta$. An equivalent phase-locked loop (PLL) for AVBS on overlapping signals can be found as shown in Fig. 5.2.

In the following section calculation of the MDF output as a function of $\theta$ is presented. Monte Carlo simulations of the AVBS system is investigated in Section C.

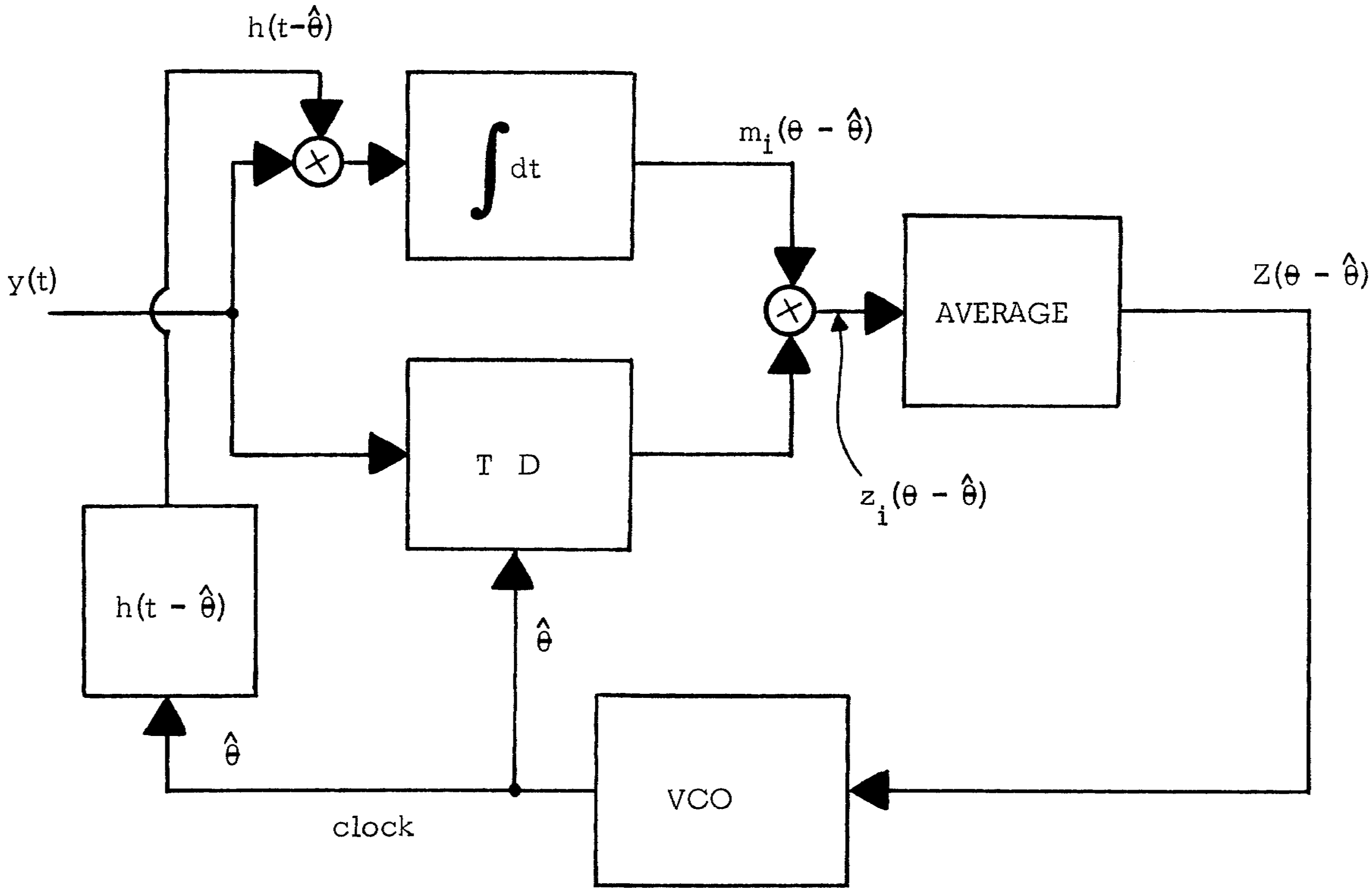



Fig.5.2 Equivalent PLL structure

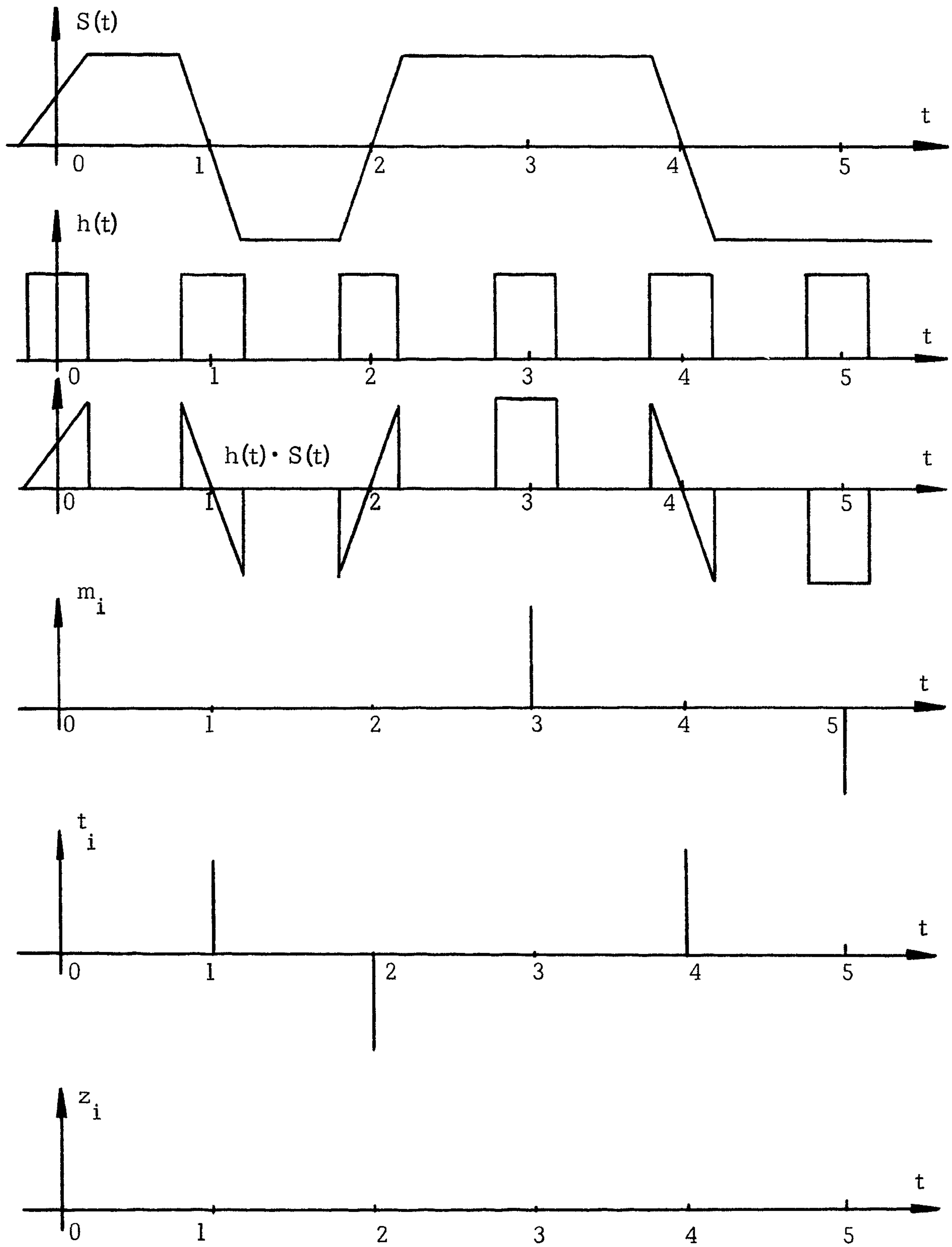



Fig. 5.3(a) AVBS output when $\theta = 0$

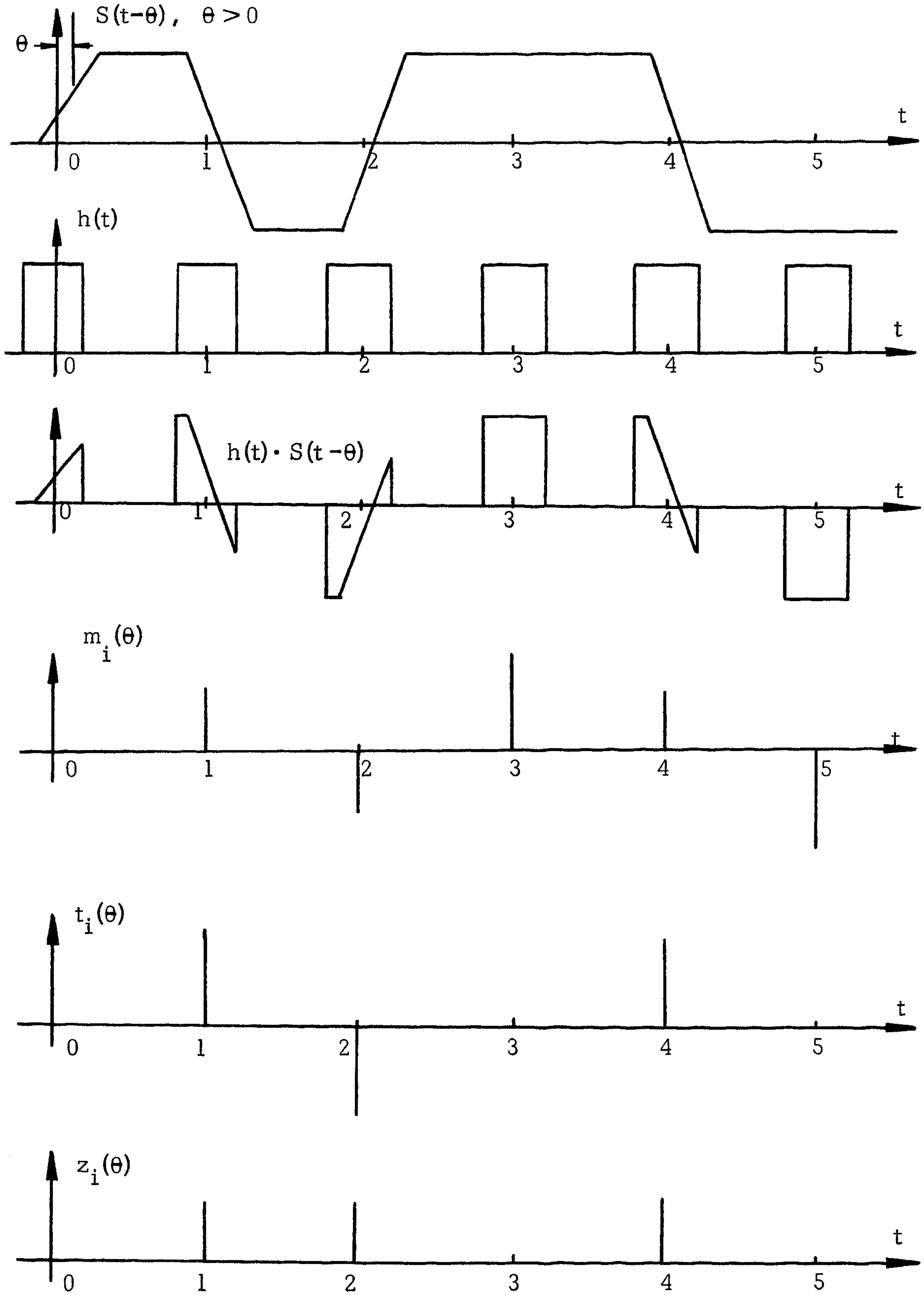



Fig 5.3(b) AVBS output when $\theta > 0$

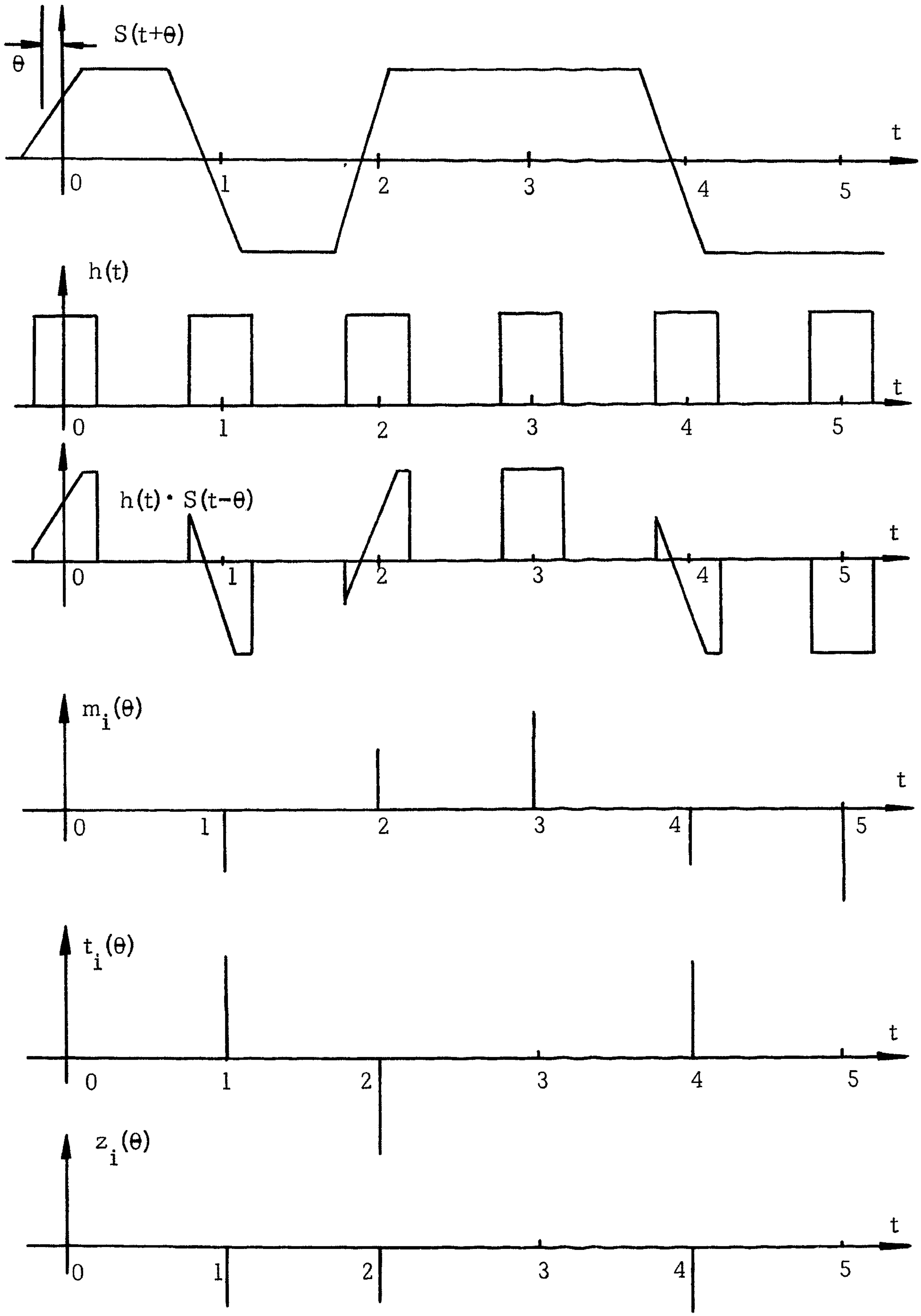



Fig. 5.3(c) AVBS output when $\theta < 0$

## B. Mathematical Derivation

This section presents an expression of the output of the MDF for a given $\theta$. The output of the MDF is considered in one of the two following cases:

1. When $\theta = 0$ and in the absence of noise,

$$m_i(\theta) = \int_{-.5+i}^{.5+i} S(t-\theta)\ h(t)\ dt$$

$$= a_i \int_{-.5+i}^{.5+i} S_p(t-i)\ h(t)\ dt + a_{i-1} \int_{-.5+i}^{.5+i} S_p(t-i+1)\ h(t)\ dt$$

$$= a_i \int_{-.5}^{.5} (.5 + t/2\alpha)\ dt + a_{i-1} \int_{-.5}^{.5} (-.5 + t/2\alpha)\ dt$$

$$= (a_i + a_{i-1})\,\alpha. \tag{5.1}$$

2. When $\theta \neq 0$ and in the absence of noise,

$$m_i(\theta) = (a_i - a_{i-1})\ (\theta^2 - 4\alpha\theta)/2\alpha. \tag{5.2}$$

There are four different cases to compute for (5.2) since $\theta$ can be either positive or negative and the transition can be either from a positive pulse to a negative pulse or vice versa. However, the resulting expressions are the same.

To find the noise coefficient, the integration limits are taken from $-\alpha$ to $\alpha$. The noise coefficient for the MDF output is

$$C_{MN} = \sqrt{E\left[\int_{-\alpha}^{\alpha} h(t)\, n(t)\, dt\right]^2} = \sqrt{2\alpha}. \qquad (5.3)$$

For the TD portion, the output is followed by the following rule:

$$t_i = \mathrm{Sgn}\left[(a_{i-1} - a_i)\, C_{TD} + n_i\right] \qquad (5.4)$$

where

$$C_{TD} = \int_{-1+2\alpha}^{1-2\alpha} a_{i-1} S_p(t-i+1-\theta)dt + \int_{-1+2\alpha}^{1-2\alpha} a_i S_p(t-i-\theta)\, dt$$

$$= 2(1 - 2\alpha) + 2\alpha/3 \qquad (5.5)$$

and the noise coefficient for the TD is

$$C_{TN} = \sqrt{2(1 - 2\alpha) + 2\alpha/3} \qquad (5.6)$$

C. Simulation Program and Results

The expressions derived in the last section are used in the simulation program to be presented in this section. A flow chart of the AVBS system is given in Fig. 5.4 and the simulation program is shown in Table 5.1.

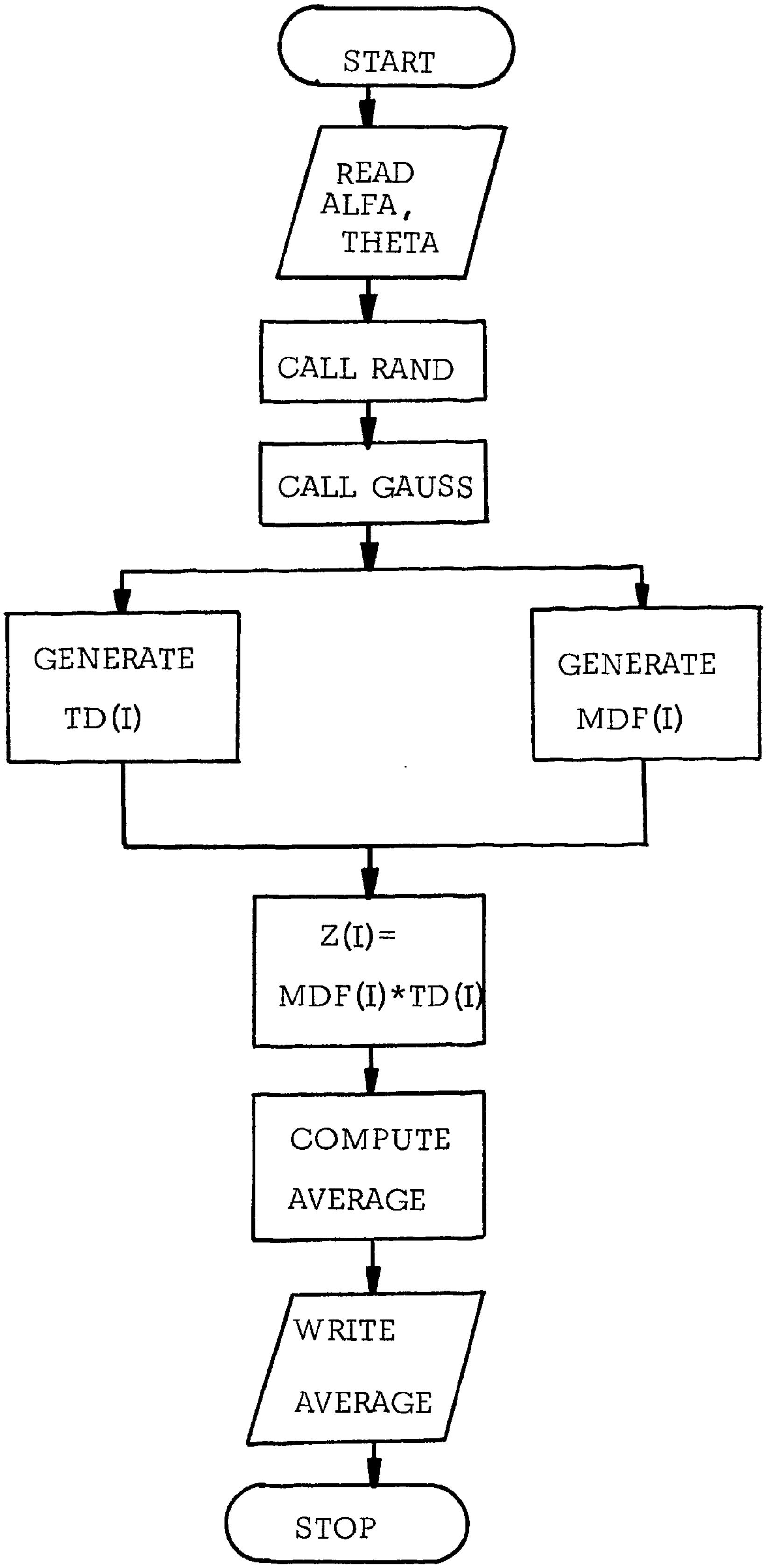



Fig. 5.4 Flow chart of AVBS simulation program

Table 5.1 AVBS simulation program

```
C
      REAL  BIT(101),MDF(100),TD(100),THEHAT(100)
      IX = 213711
 99   READ(1,100,END=999) SNR,ALFA
 100  FORMAT(2F5.2)
      THETA = 0.
      S = SQRT(1./SNR)
      CTN = SQRT(2.*(1.-2.*ALFA) + 2.*ALFA/3.)
      CMN = SQRT(2.*ALFA)
      CTD = 2.*(1.-2.*ALFA) + 2.*ALFA/3.
 9    CMDF = (4.*ALFA*THETA - THETA**2)/(2.*ALFA)
C
      DO 1  I=1,101
      CALL RAND(IX,IY,YFL)
 1    BIT(I) = SIGN(1., YFL-.5)
C
      G = 0.
      DO 2  I=1,100
      MDF(I) = (BIT(I)-BIT(I+1))*CMDF + CMN*GAUSS(S)
      TD(I) = SIGN(1., ((BIT(I)-BIT(I+1))*CTD+CTN*GAUSS(S)))
      IF(BIT(I)-BIT(I+1)) 8,2,8
 8    G = G+1
 2    THEHAT(I) = MDF(I)*TD(I)
C
      SUM =0.
      DO 3  I=1,100
 3    SUM = SUM+THEHAT(I)
      SUM = SUM/G
      WRITE(3,101)  SNR,ALFA,THETA,SUM
 101  FORMAT(///5X,'SNR='F5.2,5X,'ALFA='F5.2,5X,'THETA='
     1   F10.5,5X,'AVBS OUTPUT=',F16.7)
C
      IF(THETA .LE. 0.) GO TO 11
      THETA = THETA+0.02
      IF(THETA  .LE. ALFA) GO TO 9
      THETA = 0.
 11   THETA = THETA - 0.02
      IF(THETA .GE. -ALFA) GO TO 9
      GO TO 99
 999  STOP
      END
```

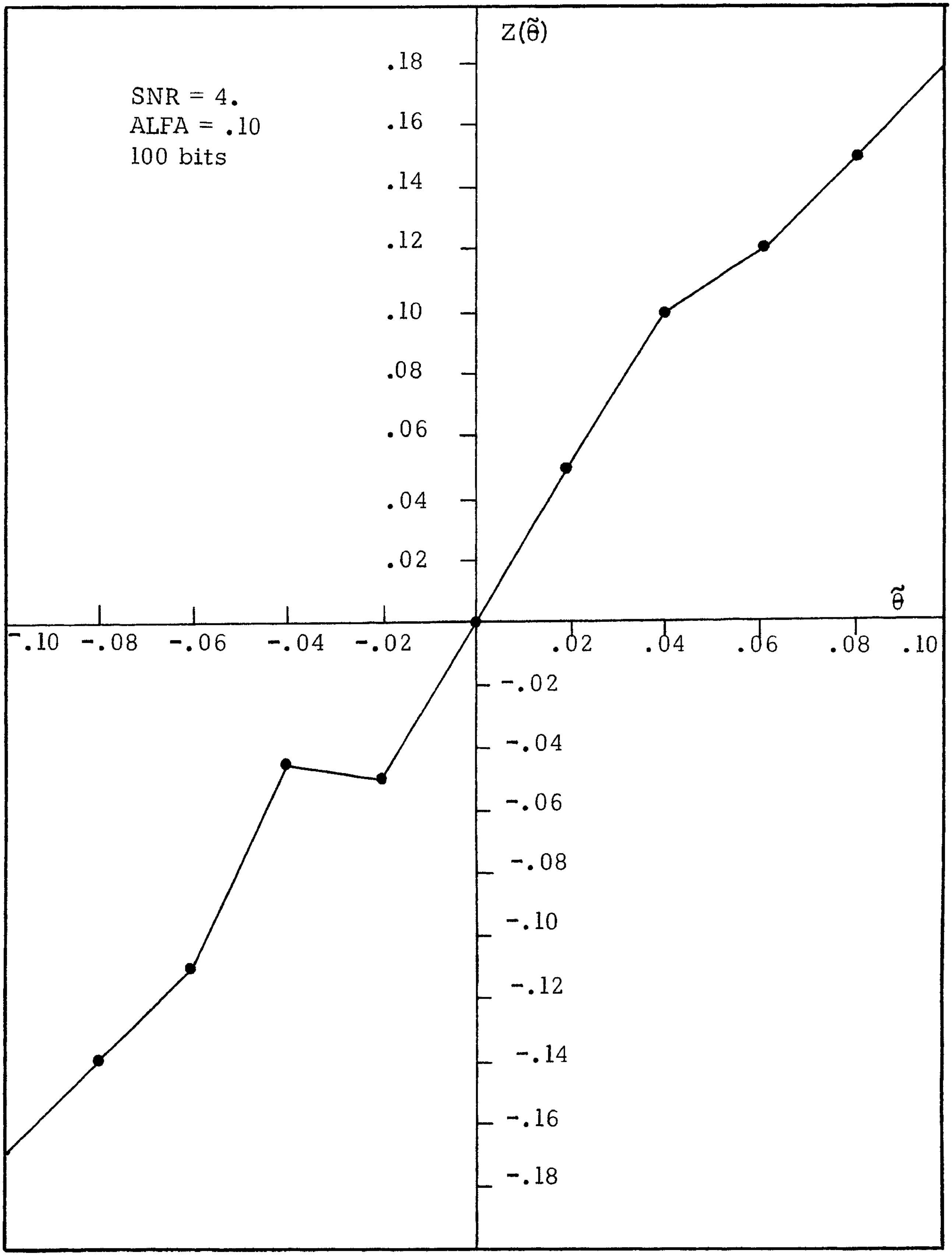



Fig. 5.5(a) AVBS simulation results

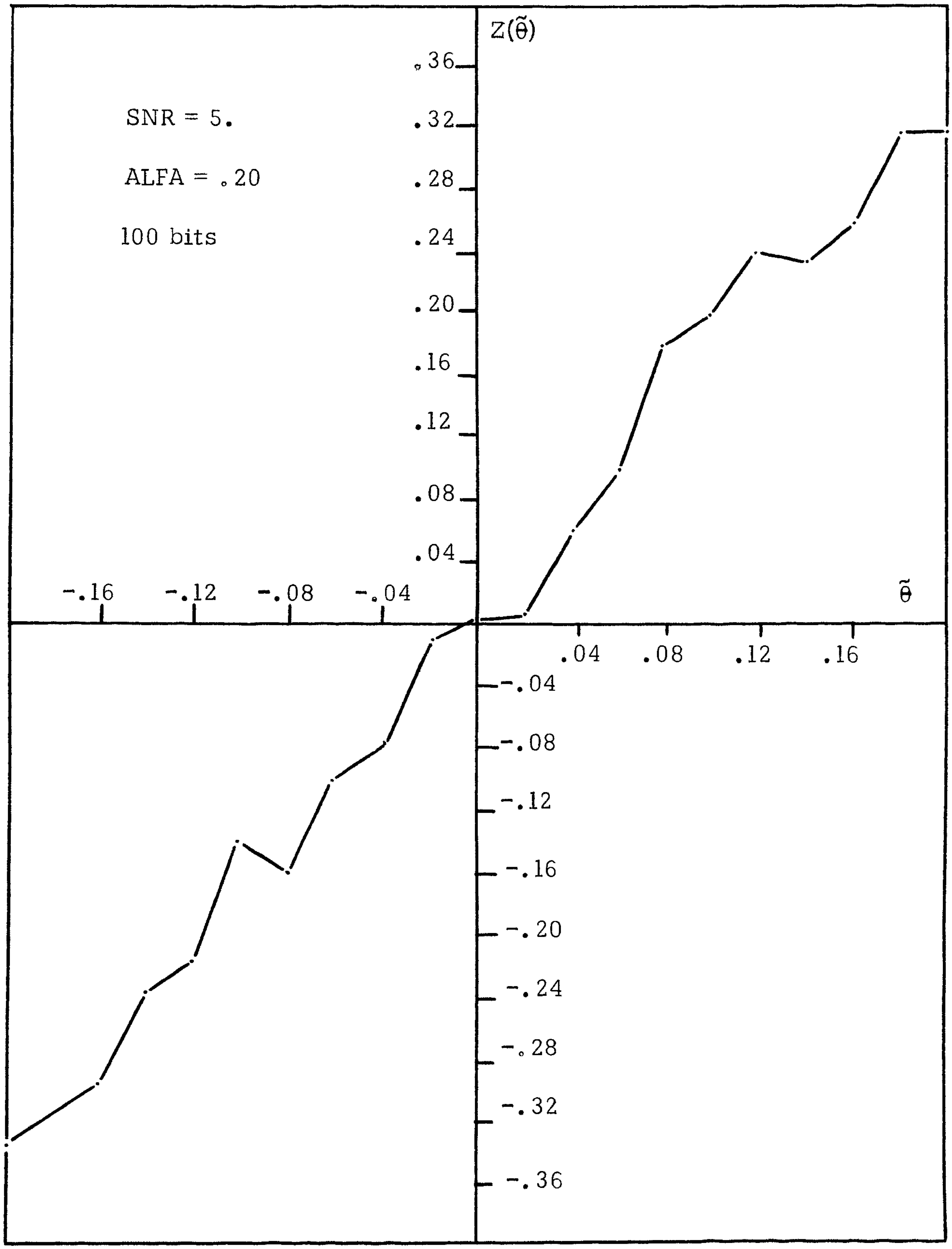



Fig. 5.5(b) AVBS simulation results

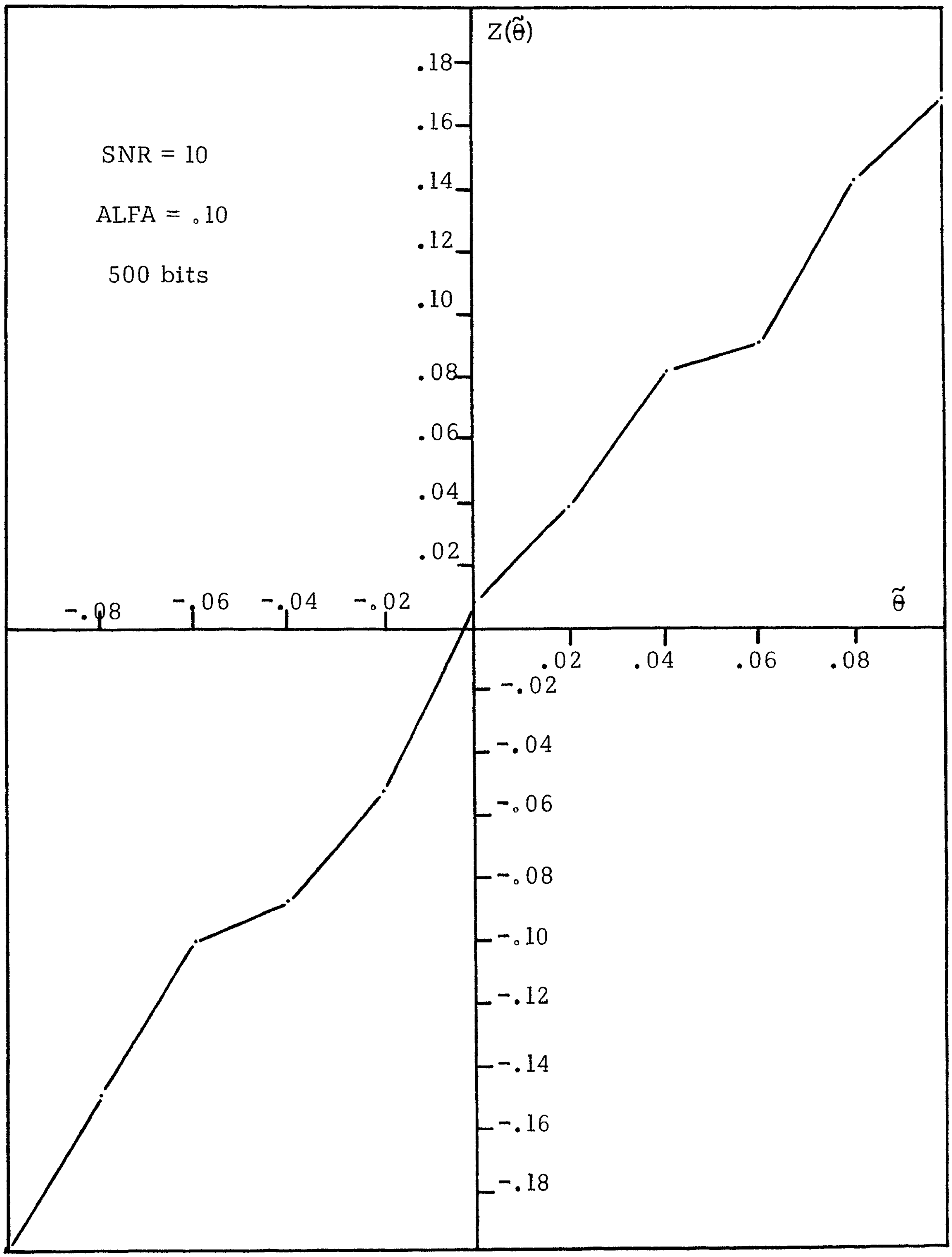



Fig. 5.5(c) AVBS simulation results

# VI. SYNCHRONIZER USING BANDLIMITED OVERLAPPING SIGNALS

## A. Introduction

In this chapter, we combine the techniques developed in Chapter VI and Chapter V along with the bandlimited version of the overlapping signals to obtain a suboptimum bit synchronizer. Suppose we receive following signals:

$$y(t) = \sum_{-\infty}^{\infty} a_n S(t-\theta-n) + n(t) \tag{6.1}$$

and pass it through an ideal lowpass filter having transfer function H(f),

$$H(f) = \begin{cases} 1 & -B \leq f \leq B \\ 0 & \text{elsewhere} \end{cases} \tag{6.2}$$

and let the output be $y^*(t)$, where

$$y^*(t) = \sum_{-\infty}^{\infty} b_n(t) + n_1(t) \tag{6.3}$$

The conditional probability density function of $y^*(t)$ given synchronization error $\theta$, and the signal sequence A is expressed as follows.

$P(y^*(y) \mid \theta, A)$

$$= K_1 \exp\left\{-(1/2B_o) \sum_{n=1}^{N} \left[y^*(n) - \sum_{k=1}^{N} a_k S_p(n-k-\theta)\right]^2\right\} \qquad (6.4)$$

We now use the fact that the overlapping signals with synchronization error $\theta$, can be approximated by the following linear relationship:

$$S(t - \theta) = S(t) - \theta \ S'(t) \qquad (6.5)$$

The associated waveforms are shown in Fig. 6.1. For small $\theta$, the difference between the two curves is small enough to be neglected. Using this approximation in the conditional probability density function,

$P(y^*(t) \mid \theta, A)$

$$= K_1 \exp\left\{-(1/2B_o) \sum_{n=1}^{N} \left[(y^*(n) - \sum_{k=1}^{N} a_k S_p(n-k)) + \theta \sum_{k=1}^{N} a_k S'_p(n-k)\right]^2\right\} \qquad (6.6)$$

To find the optimum estimate of $\theta$, we set

$$\left.\frac{\partial \ln P(y^*(t)|\theta, A)}{\partial \theta}\right|_{\theta = \hat{\theta}_{ML}} = 0$$

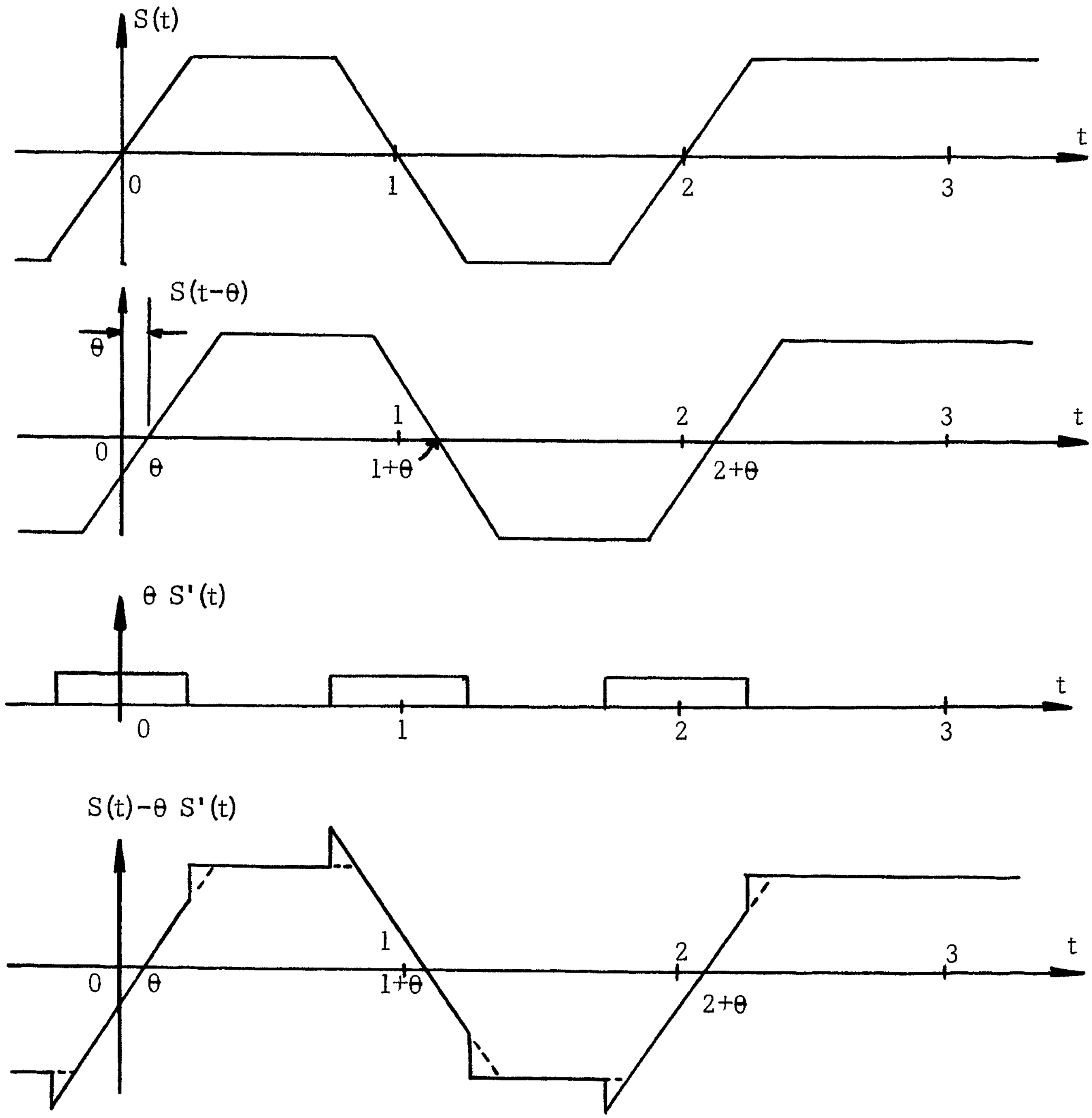



Fig. 6.1 Linear approximation of the overlapping signal

Thus,

$$\sum_{n=1}^{N} \left[ y^*(n) - \sum_{k=1}^{N} a_k S_p(n-k) + \theta \sum_{k=1}^{N} a_k S'_p(n-k) \right] \cdot \sum_{k=1}^{N} a_k S'_p(n-k) \Bigg|_{\theta = \hat{\theta}_{ML}} = 0. \tag{6.7}$$

Solving for $\theta$, we find the following result:

$$\theta_{ML} = -\sum_{n=1}^{N} \frac{\left[ y^*(n) - \sum_{k=1}^{N} a_k S_p(n-k) \right] \left[ \sum_{k=1}^{N} a_k S'_p(n-k) \right]}{\sum_{k=1}^{N} \sum_{j=1}^{N} a_k a_j S'_p(n-k) S'_p(n-j)} \tag{6.8}$$

$$= K \sum_{n=1}^{N} \left[ y^*(n) - \sum_{k=1}^{N} a_k S_p(n-k) \right] \left[ \sum_{k=1}^{N} a_k S'_p(n-k) \right] \tag{6.9}$$

The above derivation leads to a synchronizer which is roughly sketched in Fig. 6.2.

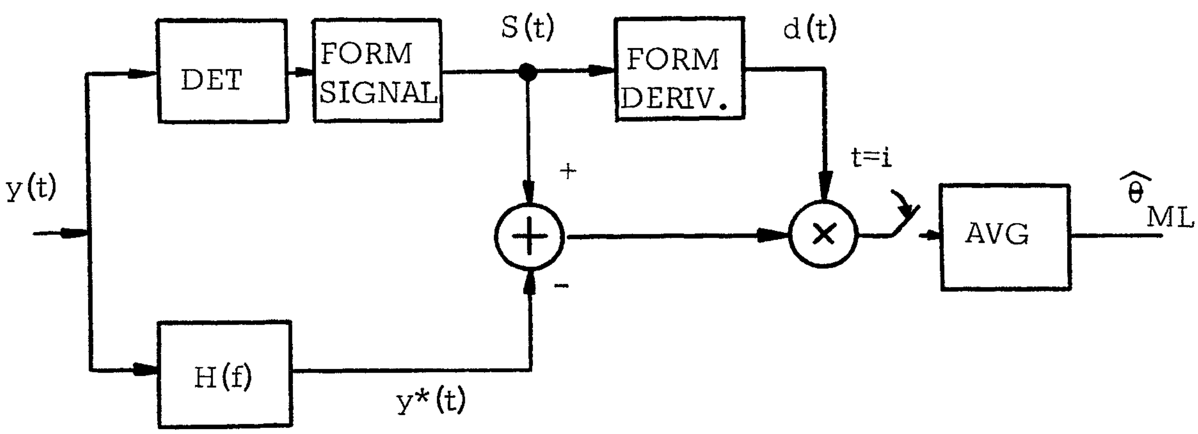



Fig.6.2 Block diagram of the suboptimum synchronizer

The received signal is first passed through a detector to form the original overlapping signal and then the derivative of the signal. Meanwhile, it is bandlimited by passing through a lowpass filter with bandwidth 2B. After forming the signal, we subtract the two waveforms, and the difference is further multiplied by the derivative of the original signal, d(t). The final block is an accumulator. The operation of the synchronizer is shown in Fig.6.3(a) and 6.3(b). With the input signal having different delays, the estimated values of $\theta$'s are shown as functions of $\theta$'s.

Furthermore, the detector portion is replaced by the DD detector investigated in Chapter VI. The overall block diagram is given in Fig.6.4. In order to simulate this system, we need to have the theoretical expression of the bandlimited signal by Fourier analysis and the bandlimited noise by autocorrelation analysis. This will be considered in the next section.

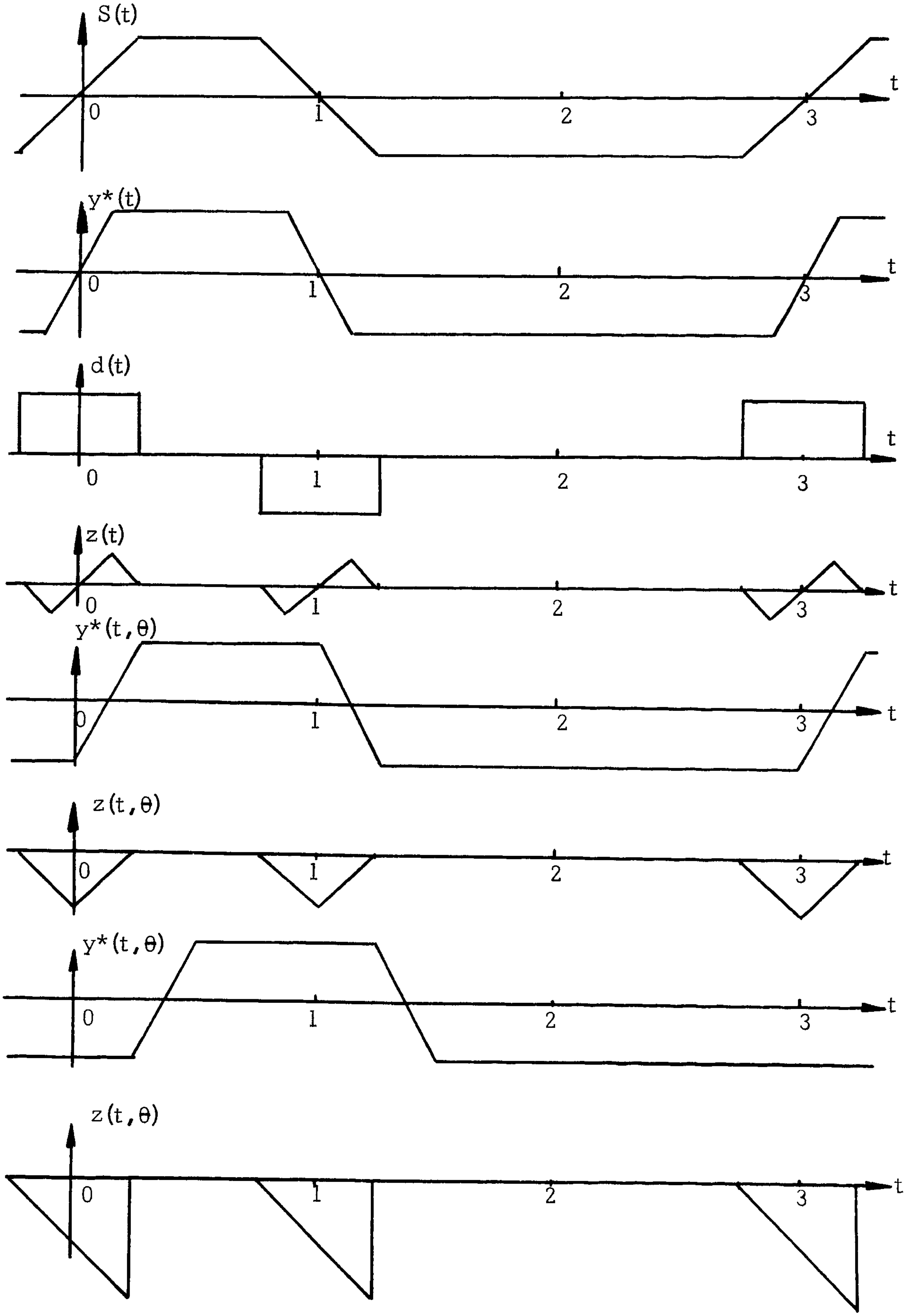



Fig. 6.3(a) Output waveforms of the suboptimum synchronizer for $\theta \geqslant 0$

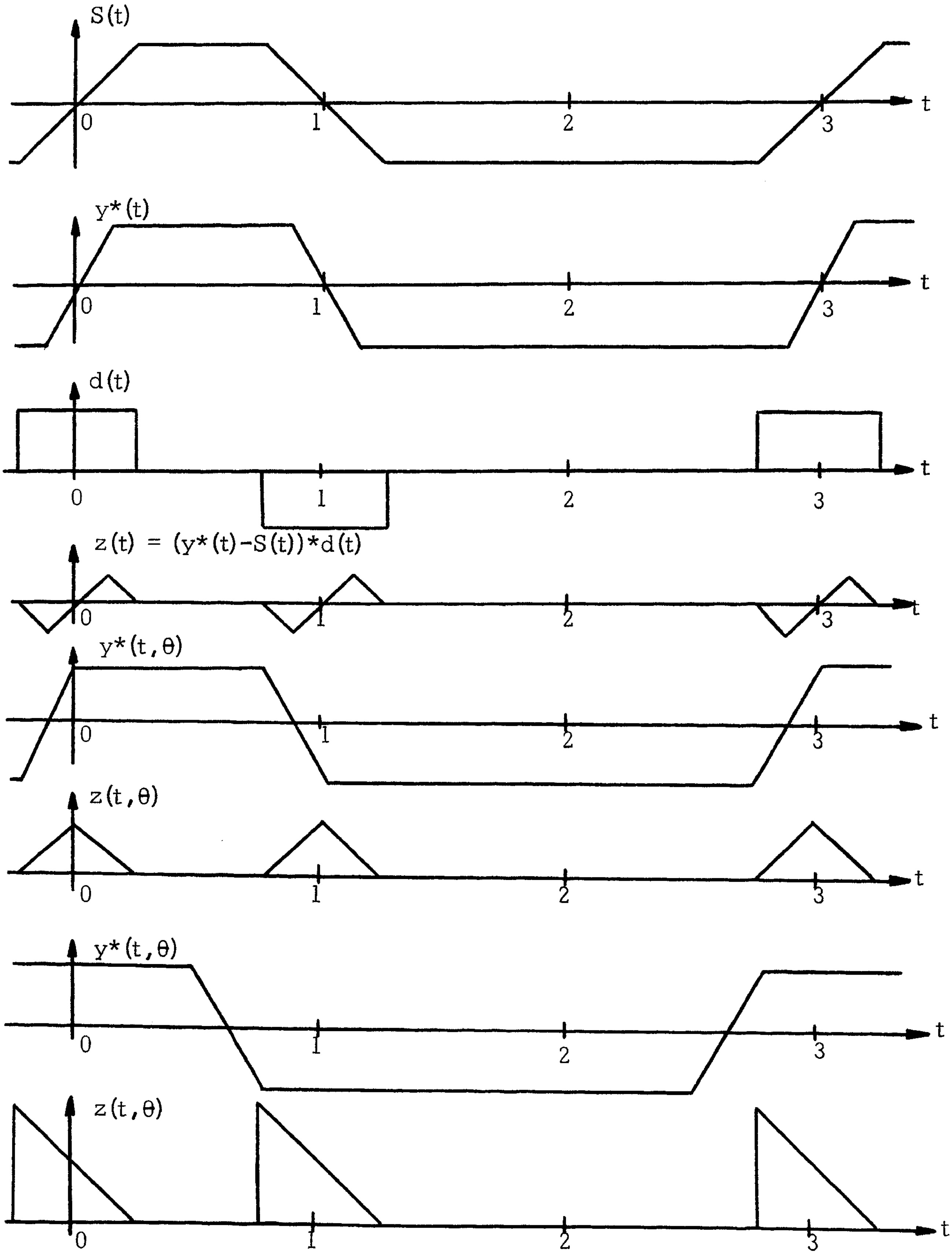



Fig. 6.3(b) Output waveforms of the suboptimum synchronizer for $\theta \leqslant 0$

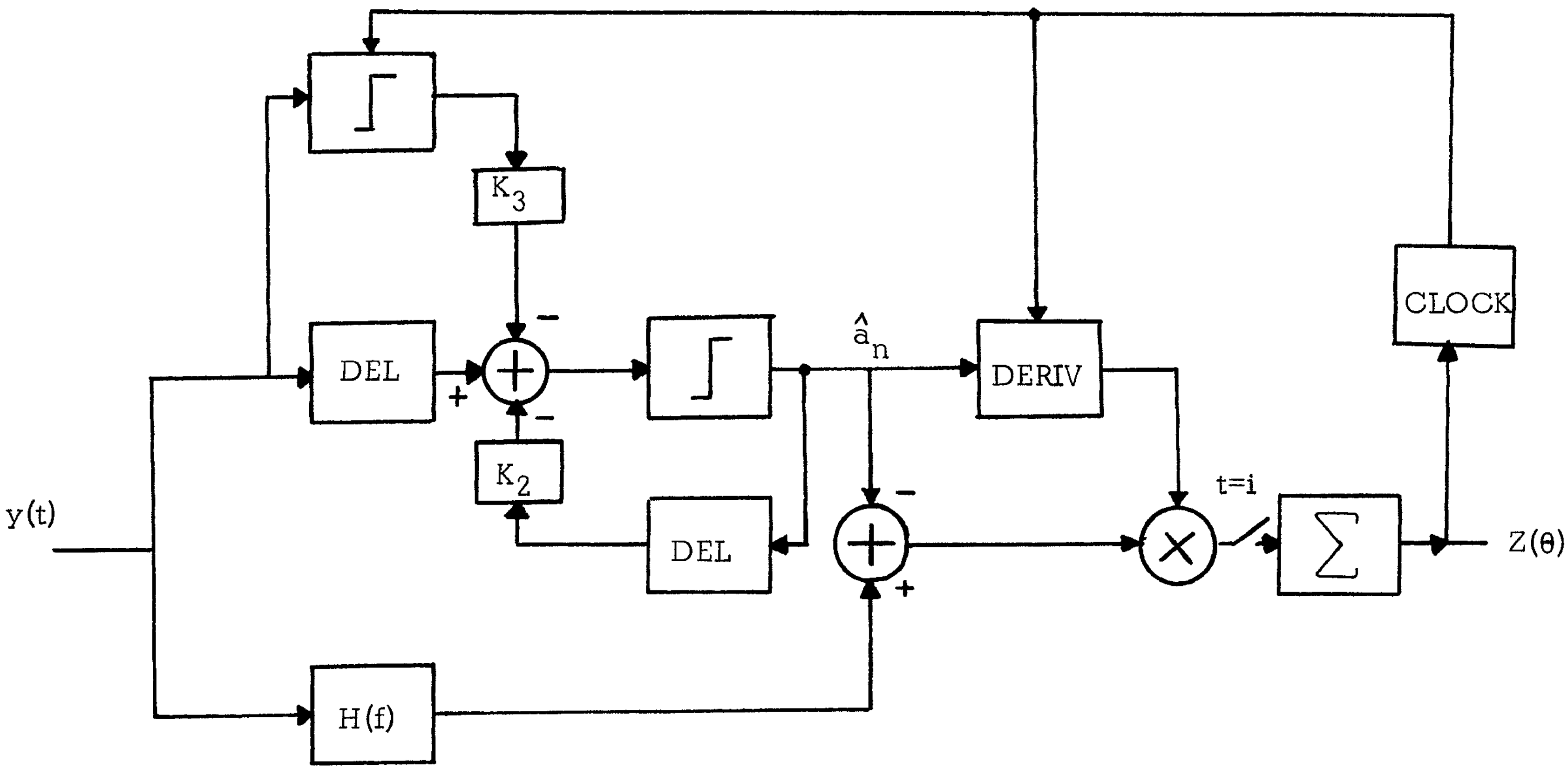



Fig.6.4 Suboptimum synchronizer structure

B. Bandlimiting and Sampling of the Overlapping Signals

To find the output of the filter, we shall first find the Fourier transform of the overlapping symbol. It is clear that if the symbol is differentiated with respect to time twice, a sequence of impulses can be obtained. The transform of the impulses is readily found. Let the symbol be centered at the origin and be called f(t). It is evident from Fig. 6.5(c) that

$$d^2f / dt^2 = (1/2\alpha) \Big[\delta(t+1/2+\alpha) - \delta(t+1/2-\alpha) - \delta(t-1/2+\alpha) + \delta(t-1/2-\alpha)\Big] \qquad (6.10)$$

Using the Fourier time shift theorem, we have

$$(jw)^2 F(w) = (1/2\alpha) \Big[\exp\big[jw(1/2+\alpha)\big] - \exp\big[jw(1/2-\alpha)\big] - \exp\big[-jw(1/2-\alpha)\big] + \exp\big[jw(1/2+\alpha)\big]\Big]$$

Thus,

$$F(w) = (1/\alpha w^2) \Big[\cos(1/2-\alpha)w - \cos(1/2+\alpha)w\Big] \qquad (6.11)$$

The time shift theorem is used again to obtain $S_p(w)$, the Fourier transform of the overlapping symbol, $S_p(t)$.

$$S_p(w) = (1/\alpha w^2) \Big[\cos(1/2-\alpha)w - \cos(1/2+\alpha)w\Big] \exp(-jw(1/2))$$

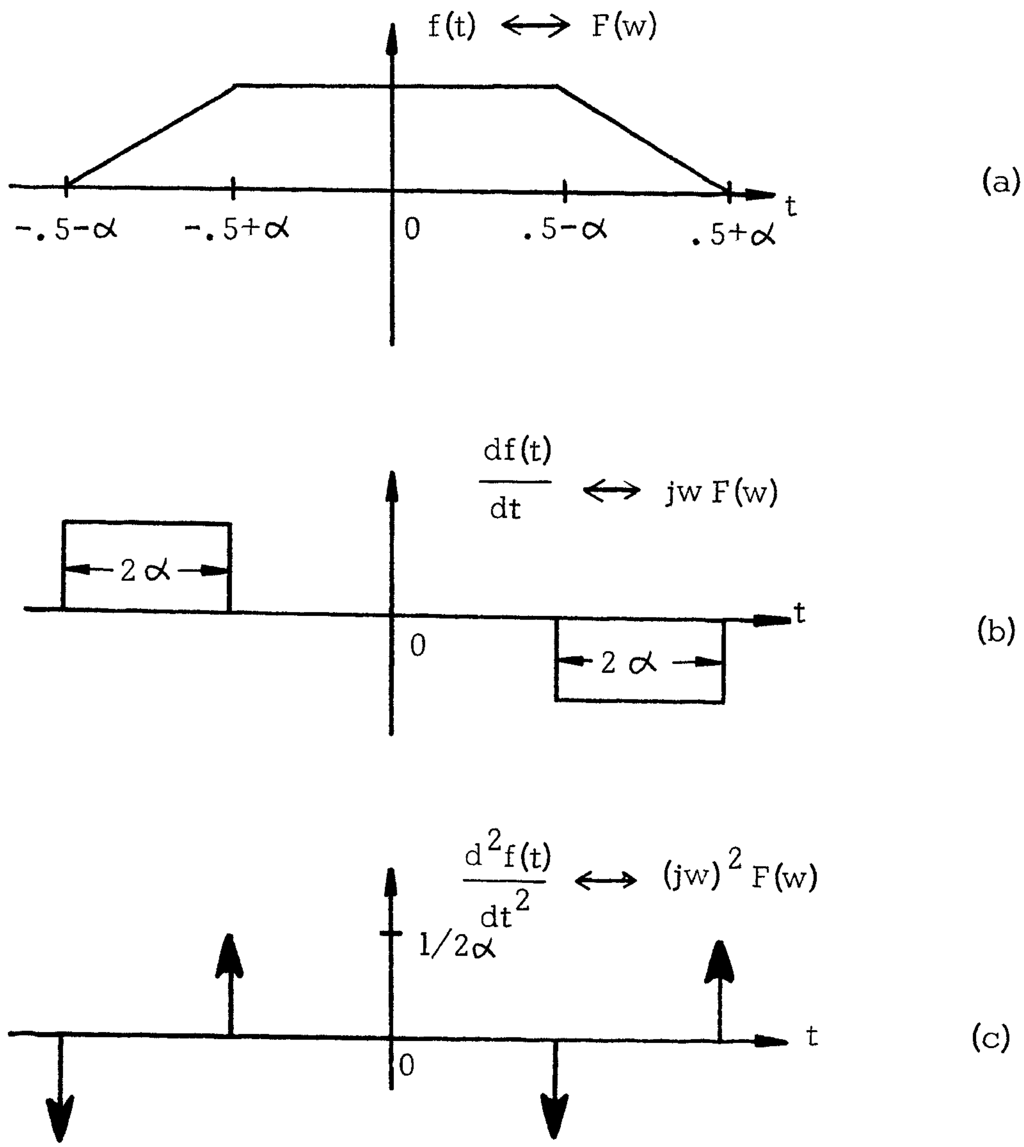



Fig. 6.5 Fourier transform of a trapezoidal function f(t)

$$S_p(w) = (1/\alpha w^2)\ 2 \sin(w/2) \sin(\alpha w) \cdot \exp(-jw/2)$$

$$= Sa(w/2)\ Sa(\alpha w) \cdot \exp(-jw/2), \tag{6.12}$$

where $Sa(x) = \sin x / x$.

Let the output of the filter be

$$y^*(t) = \sum_{n=1}^{N} b_n(t) + n_1(t).$$

Then the Fourier transform of the nth bit is

$$B_n(f) = \begin{cases} a_n\, Sa(\pi f)\, Sa(2\pi\alpha f) \cdot \exp(-j\pi f(1+2n)) & -B \leqslant f \leqslant B \\ 0\ , & \text{elsewhere} \end{cases} \tag{6.13}$$

The time response $b_n(t)$ is

$$b_n(t) = \int_{-B}^{B} B_n(f) \cdot \exp(j2\pi ft)\ df \tag{6.14}$$

$$= \int_{-B}^{B} a_n\ Sa(\pi f)\ Sa(2\pi\alpha f) \cdot \exp(-j\pi f(1+2n-2t))\ df$$

Substituting $\pi f = x$, we have

$$b_n(t) = a_n\ (2/\pi) \int_0^{\pi B} Sa(x)\ Sa(2\pi\alpha x)\ \cos(1+2n-2t)x\ dx \tag{6.15}$$

The response of the sample due to an infinite bit train can be expressed as

$$y^*(t) = \sum_{n=-\infty}^{\infty} b_n(t) + n_1(t)$$

$$= a_o (2/\pi) \int_0^{\pi B} Sa(x)\, Sa(2\alpha x) \cos(1-2t)x \; dx$$

$$+ \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} a_n (2/\pi) \int_0^{\pi B} Sa(x)\, Sa(2\alpha x) \cos(1+2n-2t)x \; dx$$

$$+ n_1(t) \qquad (6.16)$$

The first term is the desired signal and is peaked at $t = 1/2$, for $B \leqslant 1$. The second term is the intersymbol interference due to band-limiting the signals. Thus, sampled at $t = 1/2$, the response can be simplified to give

$$y^*(t = 1/2) = a_o S(B,0) + \sum_{n=-\infty}^{\infty} a_n S(B,n) + n_1(t) \qquad (6.17)$$

where

$$\begin{cases} S(B,0) = (2/\pi) \int_0^{\pi B} Sa(x)\, Sa(2\alpha x) \; dx \\ S(B,n) = (2/\pi) \int_0^{\pi B} Sa(x)\, Sa(2\alpha x) \cos(2nx) \; dx \end{cases}$$

The filtered noise has the following variance:

$$\sigma^2 = (N_o/2) \int_{-B}^{B} \left| H(f) \right|^2 df$$

$$= N_o B. \qquad (6.18)$$

In the following section, the bandlimited overlapping symbol is evaluated by numerical integration. Here we assume the bandwidth is properly chosen so that the effect of intersymbol interference is very small.

C. Simulation Program and Results

The bandlimited signal as the output of the ideal lowpass filter is

$$y_s^*(t) = (2/\pi) \int_0^{\pi B} Sa(x)\, Sa(2\alpha x) \cos(1 - 2t)x \; dx \qquad (6.19)$$

where $Sa(x) = \sin x / x$, and the subscript "s" indicates that only the signal portion without intersymbol interference is considered. We shall first numerically integrate the function in (6.19) and store the result for the main simulation program. By Simpson's rule on integration the function $y_s^*(t)$ is indicated in Fig. 6.6. Here are some prior statistics to the computer program:

Sample per bit = 8,

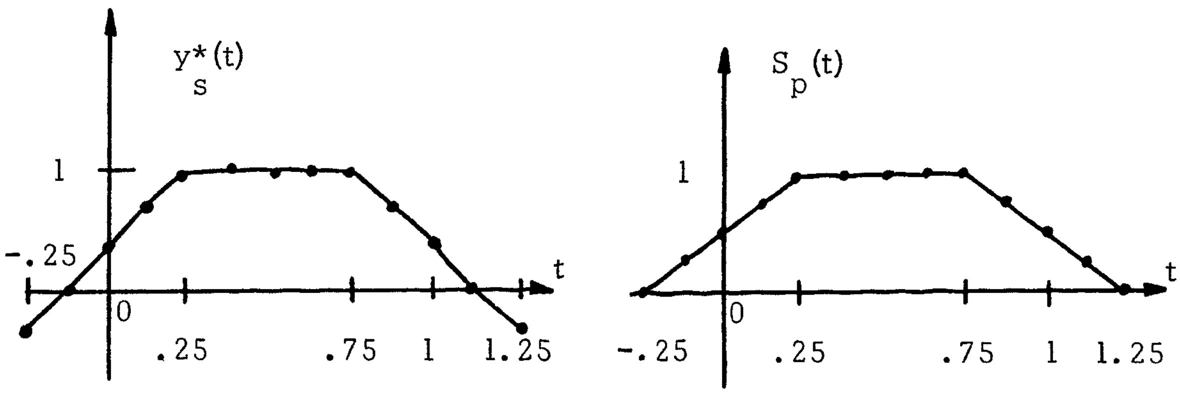
ALFA = 0.25,

Total sample calculated (due to overlap) for one bit = 13,

SNR = 5.0, and

Number of bits as the input stream = 12.

The subroutine BANDO(Y,L,BW) in the simulation program generates the required function $y_s^*(t)$. The results are shown in Fig. 6.6. The flow chart of the simulation program is drawn in Fig. 6.7 and the program is listed in Appendix B. The results are plotted in Fig. 6.8.




| t | $y_s^*(t)$ | $S_p(t)$ |
|---|---|---|
| -.250 | -.3306 | .000 |
| -.125 | -.0001 | .250 |
| .000 | .3333 | .500 |
| .125 | .6667 | .750 |
| .250 | .9972 | 1.000 |
| .375 | 1.0000 | 1.000 |
| .500 | .9999 | 1.000 |
| .625 | 1.0000 | 1.000 |
| .750 | .9972 | 1.000 |
| .875 | .6667 | .750 |
| 1.000 | .3333 | .500 |
| 1.125 | -.0001 | .250 |
| 1.250 | -.3306 | .000 |

Fig. 6.6 Bandlimited signal $y_s^*(t)$ and the overlapping symbol $S_p(t)$

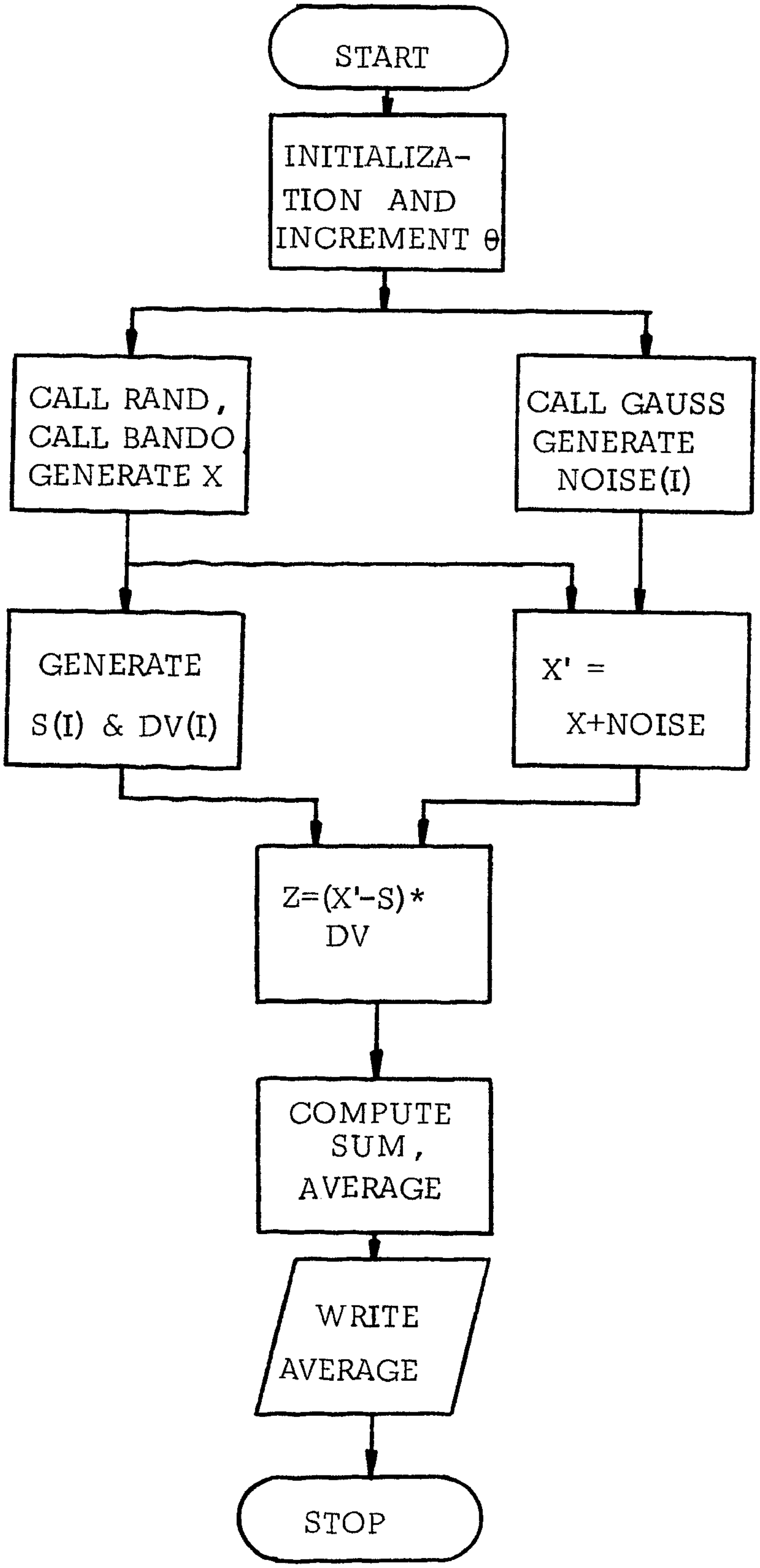



Fig.6.7 Flow chart of the suboptimum synchronizer simulation program

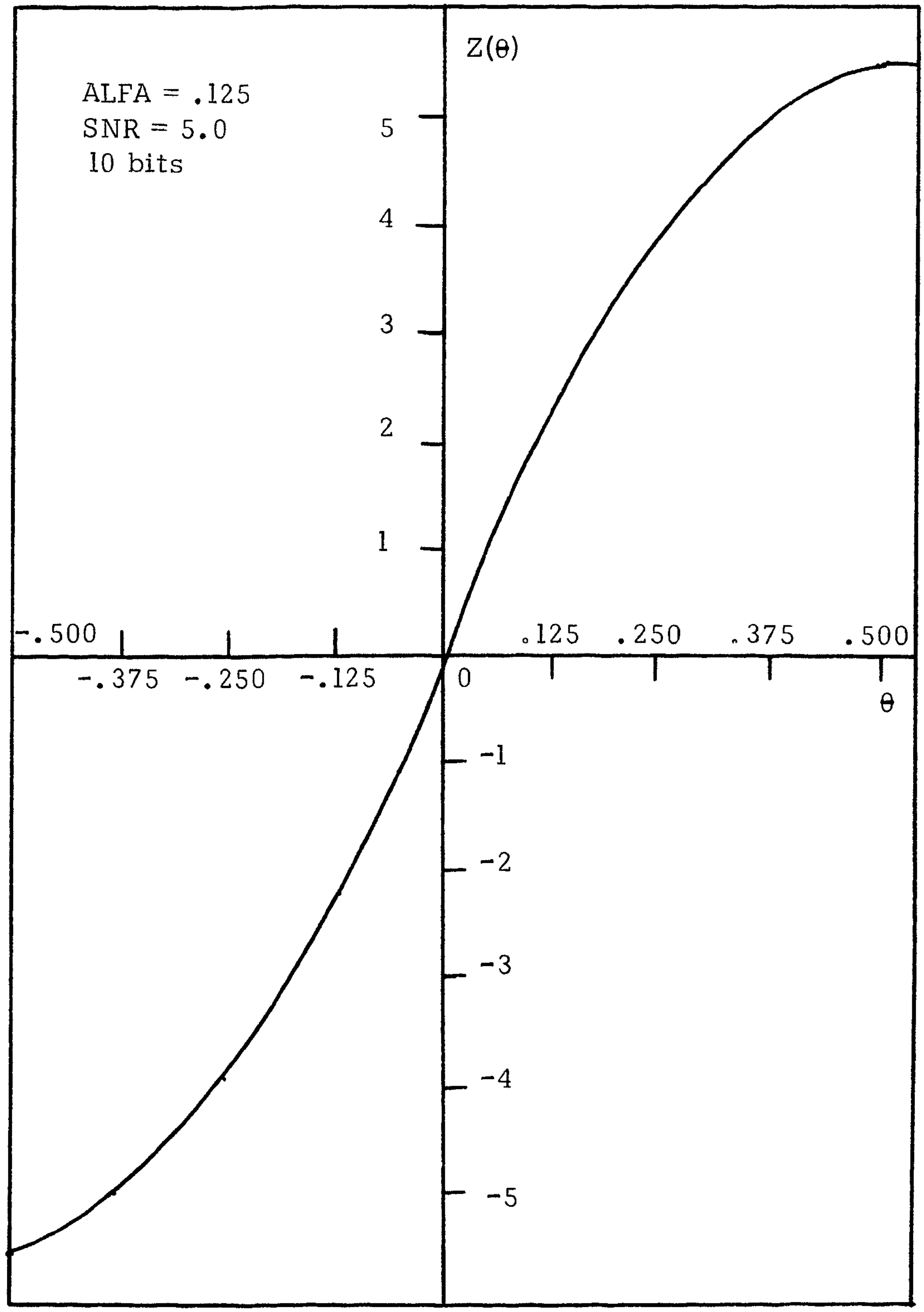



Fig. 6.8 Results of the suboptimum synchronizer simulation program

## D. Synchronizer for Overlapping Split-Phase Signals

### 1. Bandlimiting and sampling

Using the same technique in Section B, the Fourier transform of the overlapping split phase (SØ) symbol g(t) as shown in Fig. 6.9 is written as

$$\frac{d^2g}{dt^2} = (1/2\alpha)\left[\delta(t+1/2+\alpha) - \delta(t+1/2-\alpha) + \delta(t-1/2+\alpha) - \delta(t-1/2-\alpha)\right] + (1/\alpha)\left[-\delta(t+\alpha) + \delta(t-\alpha)\right] \qquad (6.20)$$

Thus, using the transform pairs, we have

$$\begin{aligned}(jw)^2\, G(w) &= (1/2\alpha)\Big[\exp(jw(1/2+\alpha)) - \exp(+jw(1/2-\alpha)) \\ &\quad + \exp(-jw(1/2-\alpha)) - \exp(-jw(1/2+\alpha))\Big] \\ &\quad + (1/\alpha)\left[-\exp(jw\alpha) + \exp(-jw\alpha)\right] \\ &= (j/\alpha)\left[\sin(1/2+\alpha)w - \sin(1/2-\alpha)w\right] - (2j/\alpha)\cdot \\ &\quad \sin(\alpha w). \\ &= (2j/\alpha)\left[\cos(w/2)\sin(\alpha w) - \sin(\alpha w)\right]. \end{aligned} \qquad (6.21)$$

Thus, $G(w) = 2j\left[\sin(\alpha w)/\alpha w^2\right]\cdot\left[1-\cos(w/2)\right]$ (6.22)

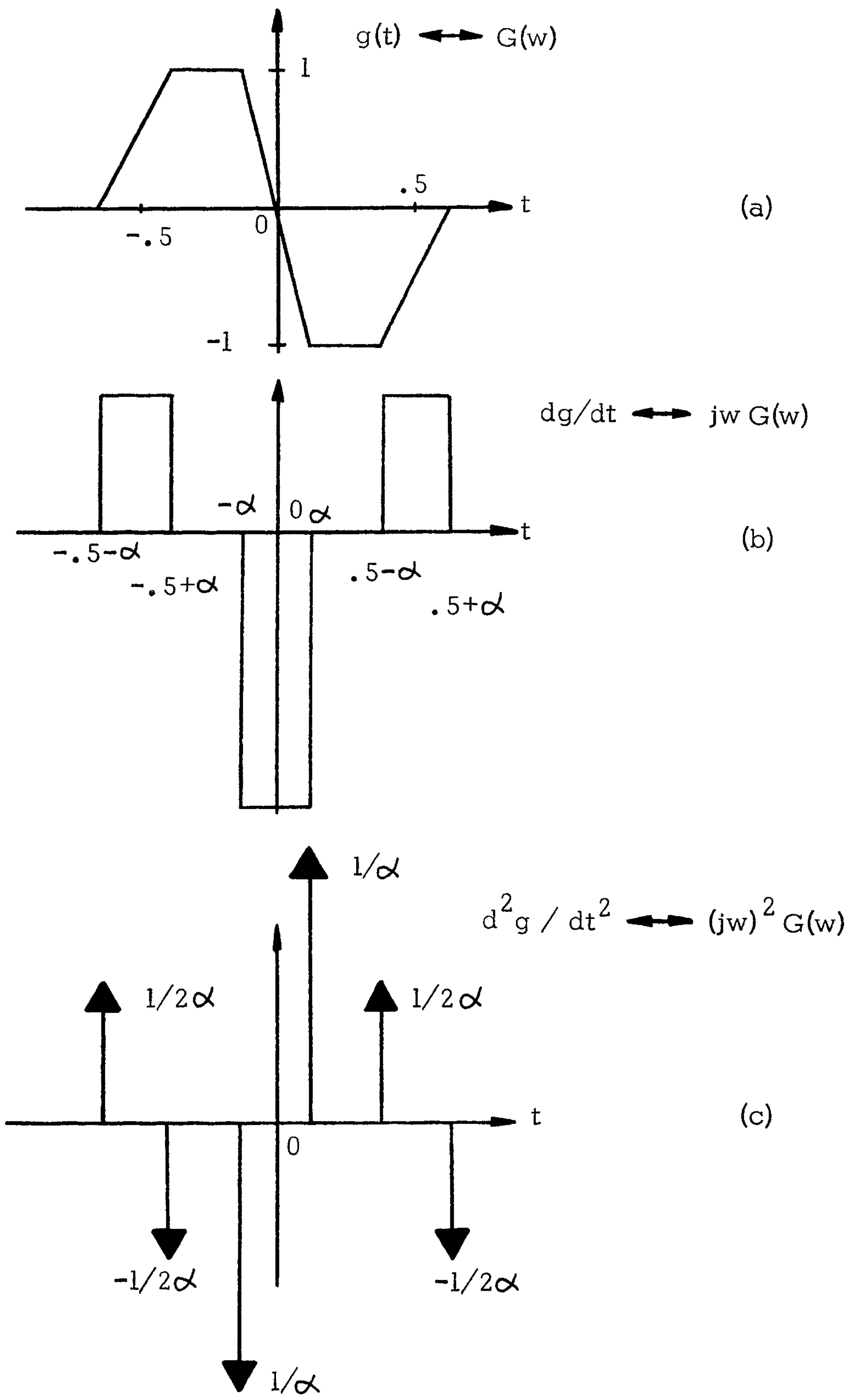



Fig. 6.9 Fourier transform of a overlapping split-phase symbol

Then the Fourier transform of the overlapping SØ symbol, $S_{\phi}(t)$, is

$$S_{\phi}(w) = G(w) \exp(-jw/2) \tag{6.23}$$

Let $w = 2\pi f \triangleq 2x$,

$$S_{\phi}(f) = (j/x) \; (\sin(2\alpha x)/\, 2\alpha x) \; (1 - \cos x) \; \exp(-jx)$$

$$= j \, Sa(2\alpha x) \left[(1 - \cos x)/x\right] \exp(-jx) \tag{6.24}$$

If the output of the LPF is

$$y^{*}_{\phi}(t) = \sum_{n=-\infty}^{\infty} b_n(t) + n_1(t),$$

The Fourier transform of $b_n(t)$ can be written as

$$B_n(f) = a_n \, j \, Sa(2\alpha x) \left[(1 - \cos x)/x\right] \exp(-jx(1+2n)). \tag{6.25}$$

Thus,

$$b_n(t) = \int_{-B}^{B} B_n(f) \exp(j2\pi ft) \, df$$

$$= a_n \frac{j}{\pi} \int_{-\pi B}^{\pi B} Sa(2\alpha x) \left[(1 - \cos x)/x\right] \exp(-jx(1+2n-2t) \, dx$$

$$= a_n \, (2/\pi) \int_{0}^{\pi B} Sa(2\alpha x) \left[(1-\cos x)/x\right] \sin\overline{(1+2n-2t)x} \, dx$$

The signal portion is found as follows,

$$y_s^*(t) = (2/\pi) \int_0^{\pi B} Sa(2\alpha x) \left[(1-\cos x)/x\right] \sin\overline{(1-2t)x}\ dx \qquad (6.27)$$

Using simpson's rule for integration, the function $y_s^*(t)$ is evaluated by the subroutine BANDO(Y,L,BW) and is shown in Fig. 6.10.

## 2. Simulation program and results

The simulation program for the overlapping split phase signals is more complicated than that of the NRZ case for two reasons: 1) twice as many samples as before should be taken in a bit interval, and 2) on the average, the number of transitions is increased by a factor of 3. In the program we have following prior statistics:

Samples per bit = 16,

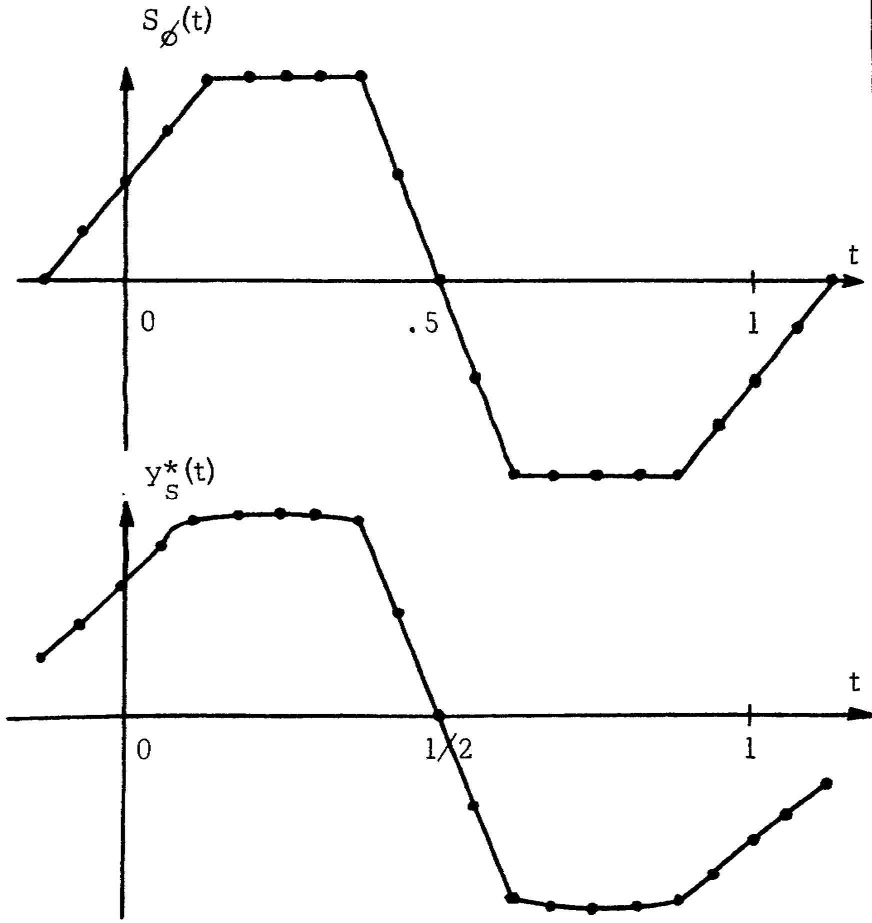
ALFA = 0.125

Total samples calculated (due to overlap) for one bit = 21

SNR = 5.0

Number of bits as the input stream = 12

The program is listed in Appendix C. The results are plotted in Fig. 6.12. It is seen that the maximum value for the estimate, $\hat{\theta}$, occured at $\theta$ = 0.375. From then on, the estimated values go down quite rapidly. Graphically, this situation is shown in Fig. 6.11, where for illustration purposes, the bandlimited output is replaced by the signal itself. At $\theta = 0.375$, the synchronizer output z(t) decreases and it is very small at $\theta = 0.5$. The above results indicate that the values for $\theta$ are restricted in the region (-0.375, 0.375). In the overlapping NRZ case we have assumed that $\theta$ is in the range (-0.5, 0.5).




| $t$ | $y_s^*(t)$ | $S_\phi(t)$ |
|---|---|---|
| -.1250 | .336035 | .0000 |
| -.0625 | .499868 | .2500 |
| .0000 | .666537 | .5000 |
| .0625 | .833284 | .7500 |
| .1250 | .997285 | 1.0000 |
| .1875 | 1.000225 | 1.0000 |
| .2500 | 1.000338 | 1.0000 |
| .3125 | 1.000336 | 1.0000 |
| .3750 | .991879 | 1.0000 |
| .4375 | .500147 | .5000 |
| .5000 | .000000 | .0000 |
| .5625 | -.500147 | -.5000 |
| .6250 | -.991879 | -1.0000 |
| .6875 | -1.000336 | -1.0000 |
| .7500 | -1.000338 | -1.0000 |
| .8125 | -1.000225 | -1.0000 |
| .8750 | -.997285 | -1.0000 |
| .9375 | -.833284 | -.7500 |
| 1.0000 | -.666537 | -.5000 |
| 1.0625 | -.499868 | -.2500 |
| 1.1250 | -.336035 | .0000 |

Fig. 6.10 Bandlimited symbol $y_s^*(t)$ and the overlapping split-phase symbol $S_\phi(t)$

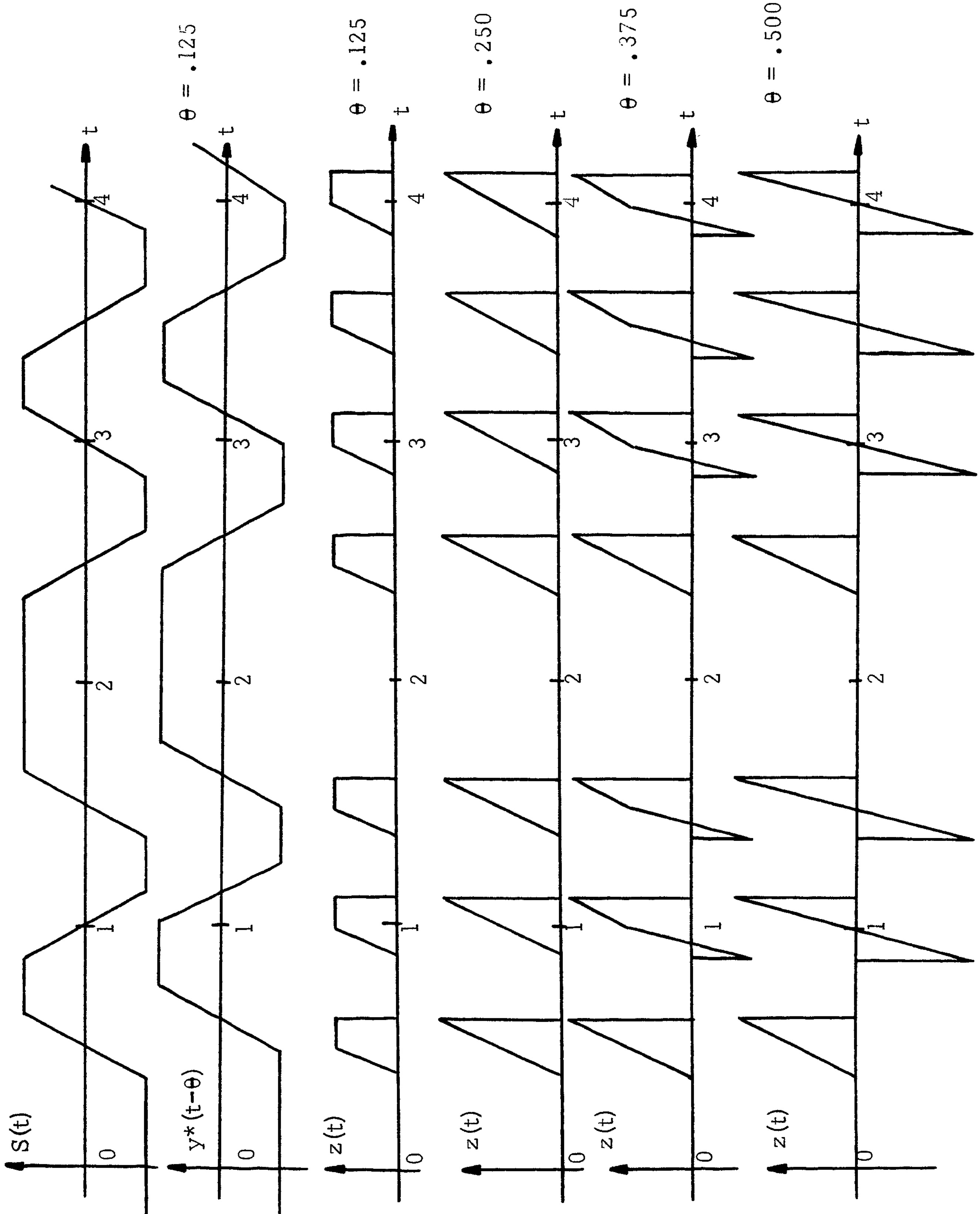



Fig. 6.11 Output waveform of the split-phase suboptimum synchronizer

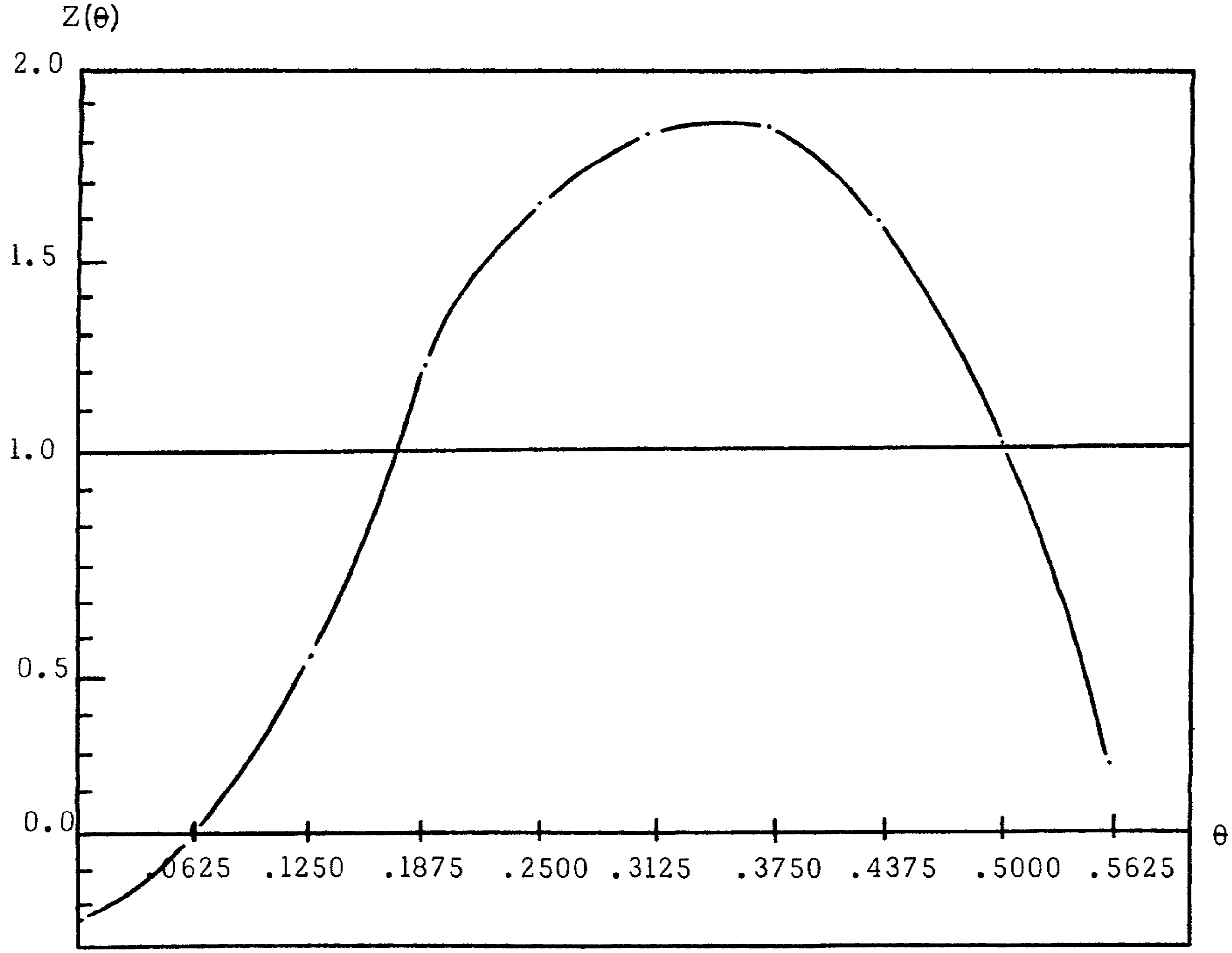



Fig. 6.12 Phase estimation from the split-phase bit synchronizer simulation program

# VII. NONLINEAR BIT SYNCHRONIZER

## A. Introduction

The general nonlinear filtering problem formulated for the time continuous case by Kushner[34] and formulated for the time discrete case by Stratonovich[35] applies to various types of communication problems. In this chapter, the nonlinear filtering technique is used to solve the bit synchronization and detection problems when dealing with overlapping signals. The message and observation models in this study are described by the following pair of stochastic differential equations.

$$d\underline{x} = \underline{f}(\underline{x})\,dt + d\underline{w} \tag{7.1}$$

$$d\underline{y} = \underline{h}(\underline{x})\,dt + d\underline{v} \tag{7.2}$$

where $\underline{x}$ represents the state and $d\underline{y}$ is the observation. $\underline{w}$ and $\underline{v}$ are independent Wiener processes.

The general nonlinear filtering problem is the determination of $P\left\{\underline{x}(t)\,\middle|\,d\underline{y}(t),\ 0 \leq t \leq T\right\}$, which is the probability density function of $\underline{x}(t)$ conditioned upon the observations $d\underline{y}$ on the interval $(0, T)$.

Similar results are available for the case where the observation is called $\underline{y}$ and the model is

$$\underline{y} = \underline{h}(\underline{x}) + \underline{n}(t) \tag{7.3}$$

where $\underline{n}(t)$ represents the white noise. Thus the equivalent problem for the observation equation (7.3) is the determination of

$P\left\{ \underline{x}(t) \mid \underline{y}(t),\ 0 \leq t \leq T \right\}$. This approach is used by Stratonovich.

The time continuous case is analyzed by solving the following filtering equation for the conditional probability density function P:

$$dP = L^{+}\left\{P\right\} dt + P\left\{ d\underline{y} - E\underline{h}(\underline{x})\, dt \right\}^{T} V_{v}^{-1} \left\{ \underline{h}(\underline{x}) - E\underline{h}(\underline{x}) \right\} \qquad (7.4)$$

where $P = P\left\{ \underline{x}(t) \mid d\underline{y}(t),\ 0 \leq t \leq T \right\}$,

$$E h(x) = \int_{-\infty}^{\infty} \cdots \int \underline{h}(\underline{x})\ P\left\{ \underline{x}(t) \mid d\underline{y},\ 0 \leq t \leq T \right\} d\underline{x}(t) \qquad (7.5)$$

and $L^{+}$, Kolmogorov's diffusion operator, is defined as

$$L^{+}\left\{\cdot\right\} = -\sum_{i=1}^{m} \frac{\partial}{\partial x_i} f_i \left\{\cdot\right\} + (1/2) \sum_{i=1}^{m} \sum_{j=1}^{m} \frac{\partial^2 \left\{\cdot\right\}}{\partial x_i \partial x_j}$$

(7.6)

The filter equation for the discrete case is similar and can be found from the results of Stratonovich,

$$dP_i = \sum_{j=1}^{m} a_{ij} P_j\, dt + P_i \left\{ dy - E h(x)\, dt \right\} V_v^{-1} \left\{ h(s_i) - E h(x) \right\}$$

(7.7)

$$\text{where} \quad P_i = P\left\{ x(t) = s_i(t) \;\middle|\; dy(t),\ 0 \leqslant t \leqslant T \right\}, \tag{7.8}$$

$$Eh(x) = \sum_{i=1}^{m} h(s_i(t)) \cdot P_i \tag{7.9}$$

and the corresponding $L^+$ can be described by a matrix whose elements are the transition probabilities:

$$a_{ij} = \lim_{\Delta t \to 0} \quad Pr\left\{ x(t+\Delta t) = s_j(t+\Delta t) \;\middle|\; x(t) = s_i(t) \right\} / \Delta t \tag{7.10}$$

$$a_{ii} = -\lim_{\Delta t \to 0} \quad 1 - Pr\left\{ x(t+\Delta t) = s_i(t+\Delta t) \;\middle|\; x(t) = s_i(t) \right\} / \Delta t \tag{7.11}$$

B. An Example

Suppose we have received a sequence of binary NRZ signals with synchronization error $\theta$ and noisy observations. Find the filtering equation for the probability density function and the nonlinear bit synchronizer structure.

For the noiseless case, the observed symbol in the interval

$$\left[ (n-1) + \theta,\ n+\theta \right], \quad n = 1, 2, \ldots, N$$

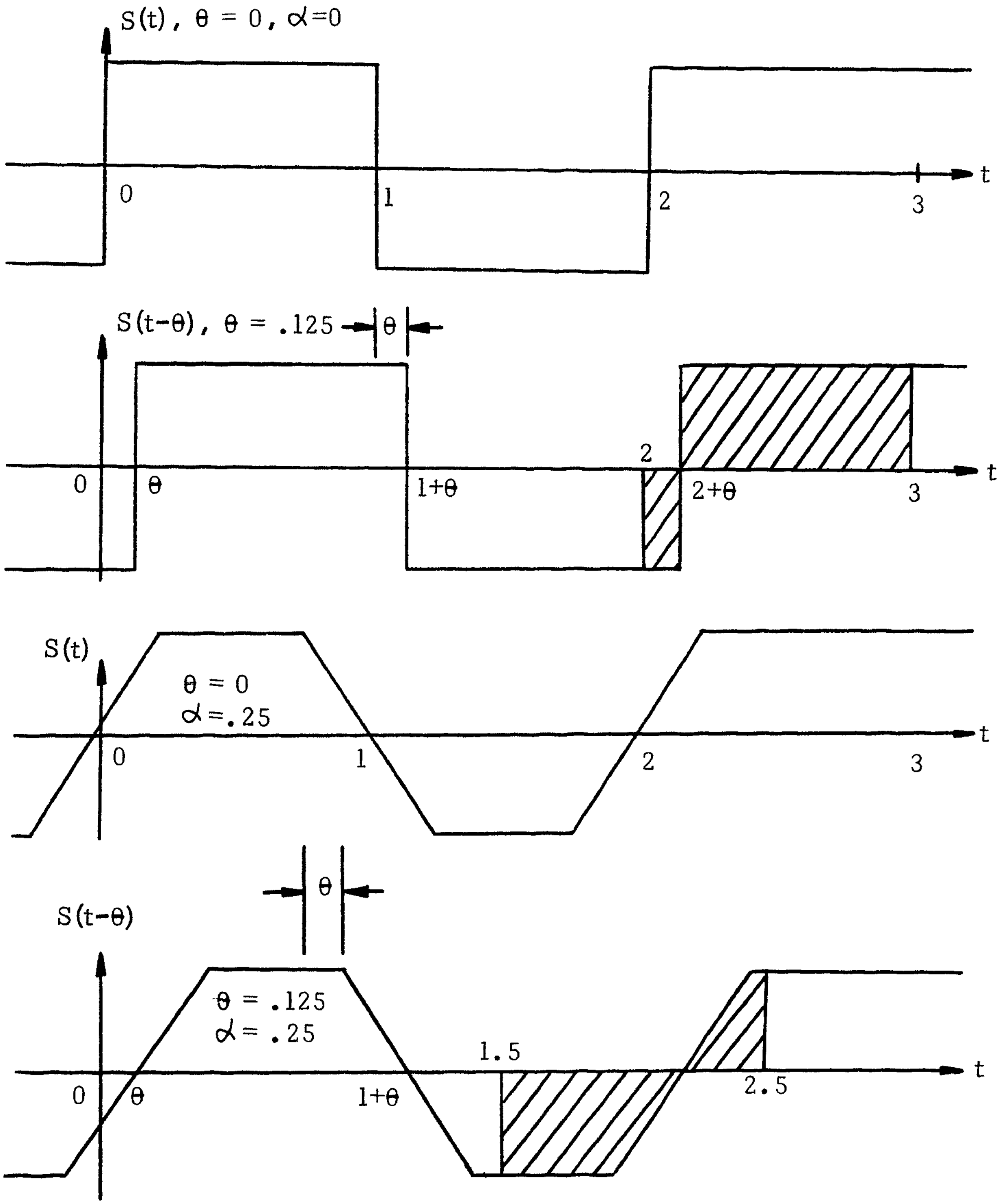



Fig.7.1(a) NRZ signal
(b) Overlapping signal

is $$s^n(t) = a_n \tag{7.11}$$

A typical received signal waveform is indicated in Fig.7.1(a). To formulate the filtering equation, let us first define the following,

$$s_{ij}^n = \begin{cases} i, & \text{for } n < t \leq n + \theta \\ & \\ j, & \text{for } n + \theta < t \leq n + 1 \end{cases} \qquad i,j = -1,1. \tag{7.12}$$

and the probabilities,

$$P_{ij}^n(t,\theta)$$

$$= \Pr\left\{ s^n(t)=i, \text{ for } (n-1)\leq t\leq(n-1+\theta)\ ;\ s^n(t)=j, \text{ for } (n-1+\theta)\leq t\leq n \right\}. \tag{7.13}$$

where $i,j = -1, 1$ $\quad n = 1, 2, \ldots, N.$

Let $$\dot{y} = s^n(t) + dv / dt \tag{7.14}$$

be the observation on the interval $(n-1, n)$ and $E(dv^2) = B_o dt$ where $B_o$ is the equivalent spectral density of a white noise.

Using the result of Kushner[14], the probability $P_{ij}^n(t,\theta)$ must satisfy the following filtering equation:

$$\dot{P}_{ij}^n(t,\theta) = L^+\left\{P_{ij}^n(t,\theta)\right\} + P_{ij}^n(t,\theta)\ (\dot{y} - m_\theta(t))\ (s_{ij}^n - m_\theta(t))/B_o \tag{7.15}$$

where

$$L^{+} P^{n}_{ij}(t,\theta) = - \sum_{i=1}^{n} \frac{\partial}{\partial x_i} f_i(x) P^{n}_{ij}(t,\theta)$$

$$+ (1/2) \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j} P^{n}_{ij}(t,\theta) \qquad (7.16)$$

and $$m_\theta(t) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} \int_{-\infty}^{\infty} s^{n}_{ij}(t,\theta) P^{n}_{ij}(t,\theta) \, dt$$

(7.17)

If $\theta$ is assumed to be constant at least for several bit periods, the term $L^{+} P^{n}_{ij}(t,\theta)$ is zero and the equation reduces to

$$\dot{P}^{n}_{ij}(t,\theta) = P^{n}_{ij}(t,\theta) \, (\dot{y} - m_\theta(t)) \, (s^{n}_{ij} - m_\theta(t)) / B_o \qquad (7.18)$$

If the symbols are independent and equally probable, the following relations can be found.

$$P^{n}_{1,-1}(n,\theta) = P^{n}_{1,1}(n,\theta) = (1/2) P^{n-1}_{-1,1}(n,\theta) + (1/2) P^{n-1}_{1,1}(n,\theta)$$

(7.19)

$$P^{n}_{-1,1}(n,\theta) = P^{n}_{-1,-1}(n,\theta) = (1/2) P^{n-1}_{1,-1}(n,\theta) + (1/2) P^{n-1}_{-1,-1}(n,\theta)$$

(7.20)

$$P^n_{ij}(n+1) = \sum_{\theta} P^n_{ij}(n+1,\theta) \tag{7.21}$$

where the summation is over all possible values of $\theta$.

Althought both bits $a_n$ and $a_{n+1}$ could be estimated in each interval, it is clear that only part of the second bit has been observed and thus a better estimation can be made in the next interval. Hence the optimum estimate of the first bit in the interval $n < t < n+1$ is determined by checking whether or not the following inequality holds.

$$P^n_{1,-1}(n+1) + P^n_{1,1}(n+1) < P^n_{-1,1}(n+1) + P^n_{-1,-1}(n+1) \tag{7.22}$$

We decide $a_n = 1$ was sent if the above inequality does hold; if not, we decide $a_n = -1$. The nonlinear bit synchronizer is shown in Fig. 7.2. It is obtained by solving the filtering equation, (7.18), and by using the relation described by (7.22). Thus it represents a combined synchronizing and estimation scheme which is optimum in the sense that it makes bit by bit decisions conditioned upon all observations up to that time. This technique can be implemented with analog computers since the only nonlinear elements required are multipliers.

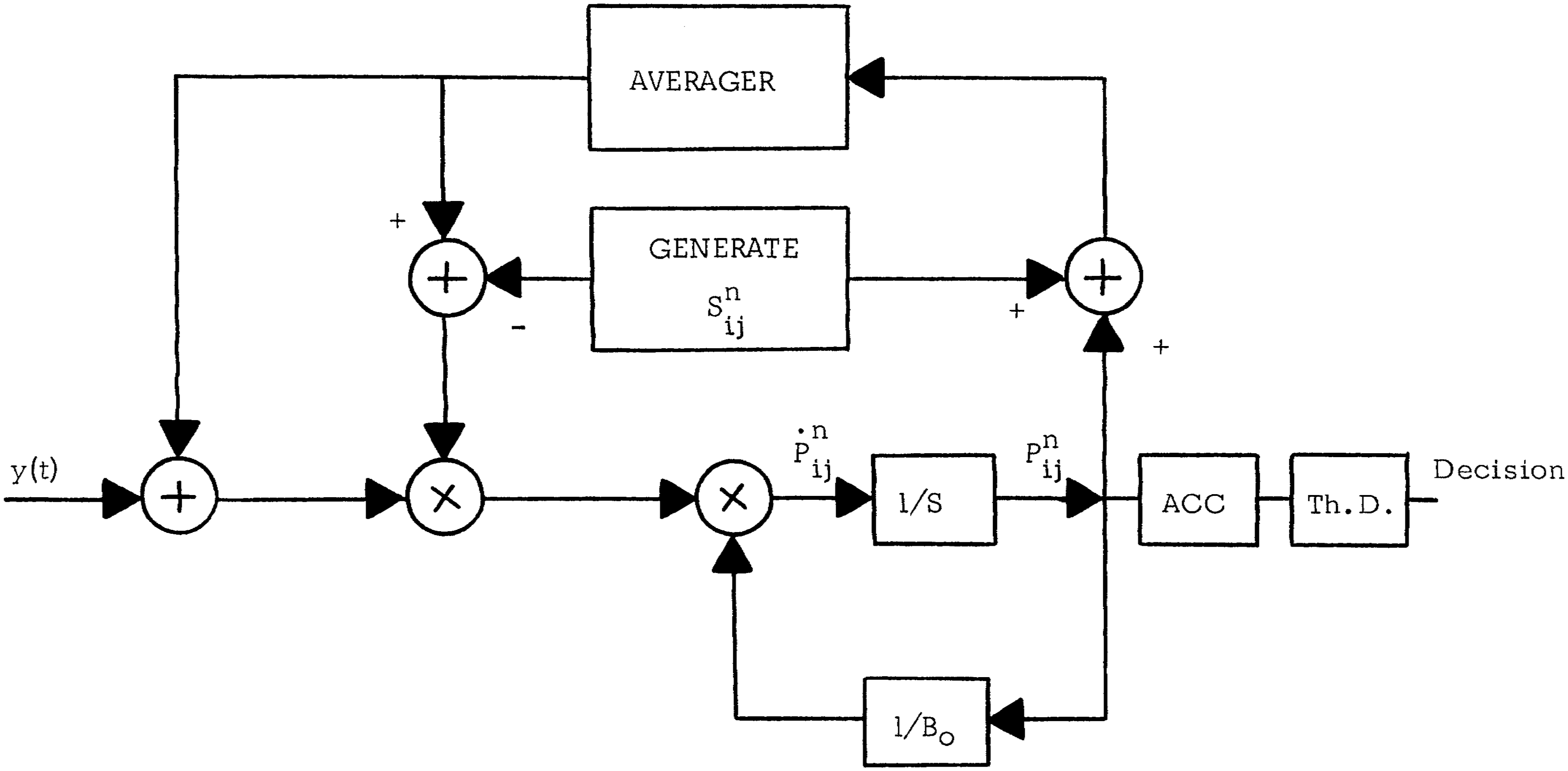



Fig.7.2 Nonlinear bit synchronizer for NRZ signals

C. Nonlinear Bit Synchronizer for Overlapping Signals

For the overlapping signal case as shown in Fig.7.1(b), the message and observation models are modified in order to set up a filtering equation. Let the observed symbol in the interval

$$\left[(n-1)-\alpha+\theta,\ n+\alpha+\theta\right],\quad n=1,2,\ldots,N$$

be $s^n(t) = a_n S_p(t)$. (7.23)

where $S_p(t)$ is the overlapping symbol defined in Chapter III.

Using the same approach as Eq. (3.6), we again consider the following set of intervals,

$$\left[n-(1/2),\ n+(1/2)\right],\quad n=1,2,\ldots,N.$$

and define the following functions:

$$S^n_{ij}(t,\theta) = \begin{cases} a_n S_p(t-\theta-n) & \text{for } (n-.5)\leq t\leq(n+\theta+\alpha) \\ a_{n+1} S_p(t-1-\theta-n) & \text{for } (n+\theta-\alpha)< t\leq(n+.5) \end{cases} \quad (7.24)$$

where $i \Leftrightarrow a_n$ , $j \Leftrightarrow a_{n+1}$, $n = 1, 2, \ldots, N$.

The related waveform for $a_n = 1$ and $a_{n+1} = -1$ is indicated by Fig.7.3 in the interval $(n-1/2), (n+1/2)$. We also need to define the following probabilities:

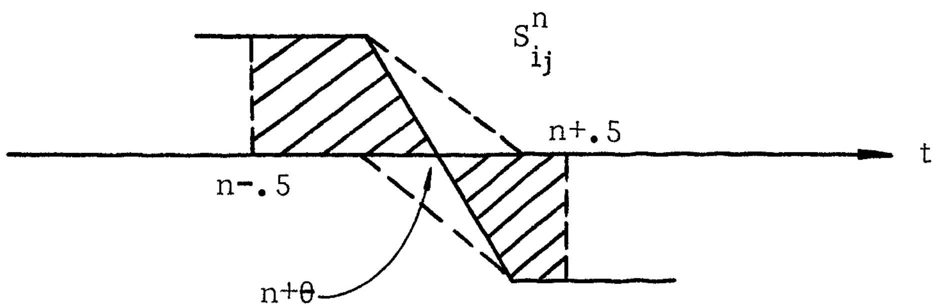



Fig.7.3 $S^n_{ij}$ and the signal waveform in the interval ( n-.5, n+.5)

$$P^n_{ij}(t,\theta) = \Pr\left\{\begin{array}{ll} s^n(t)=a_n S_p(t-\theta-n), & \text{for } (n-.5)\leq t\leq (n+\theta+\alpha) \ ; \\ s^n(t)=a_{n+1} S_p(t-\theta-n-1), & \text{for } (n+\theta-\alpha) < t \leq (n+.5) \end{array}\right\} \tag{7.25}$$

where $a_n \Longleftrightarrow i$, $a_{n+1} \Longleftrightarrow j$, $i,j = 1, -1$.

Then the observation model on the interval (n-.5, n+.5) can be written as follow.

$$\dot{y}(t) = S^n_{ij}(t,\theta) + dv/dt , \tag{7.26}$$

where $E(dv^2) = B_o dt$.

Substituting (7.24) into (7.26), we have

$$\dot{y}(t) = \left\{ a_n S_p(t-\theta-n) + a_{n+1} S_p(t-\theta-n-1) \right\} + dv/dt$$

The filtering equation can be found as follows.

$$\dot{P}^{n}_{ij}(t,\theta) = L^{+} \; P^{n}_{ij}(t,\theta) + P^{n}_{ij}(t,\theta) \; ( \dot{y} - m_{\theta}(t) ) \; (S^{n}_{ij}(t,\theta) - m_{\theta}(t))/B_{o} \tag{7.27}$$

where

$$m_{\theta}(t) = \sum_{i=-1}^{1} \sum_{i=-1}^{1} \int_{-\infty}^{\infty} S^{n}_{ij}(t,\theta) \; P^{n}_{ij}(t,\theta) \; dt \tag{7.28}$$

The bit synchronizer structure by solving (7.27) is shown in Fig.7.4. The received signal is passed through a transition detector to form the function $S^{n}_{ij}(t,\theta)$. The rest of the structure is similar to the synchronizer developed in Section B of this chapter.

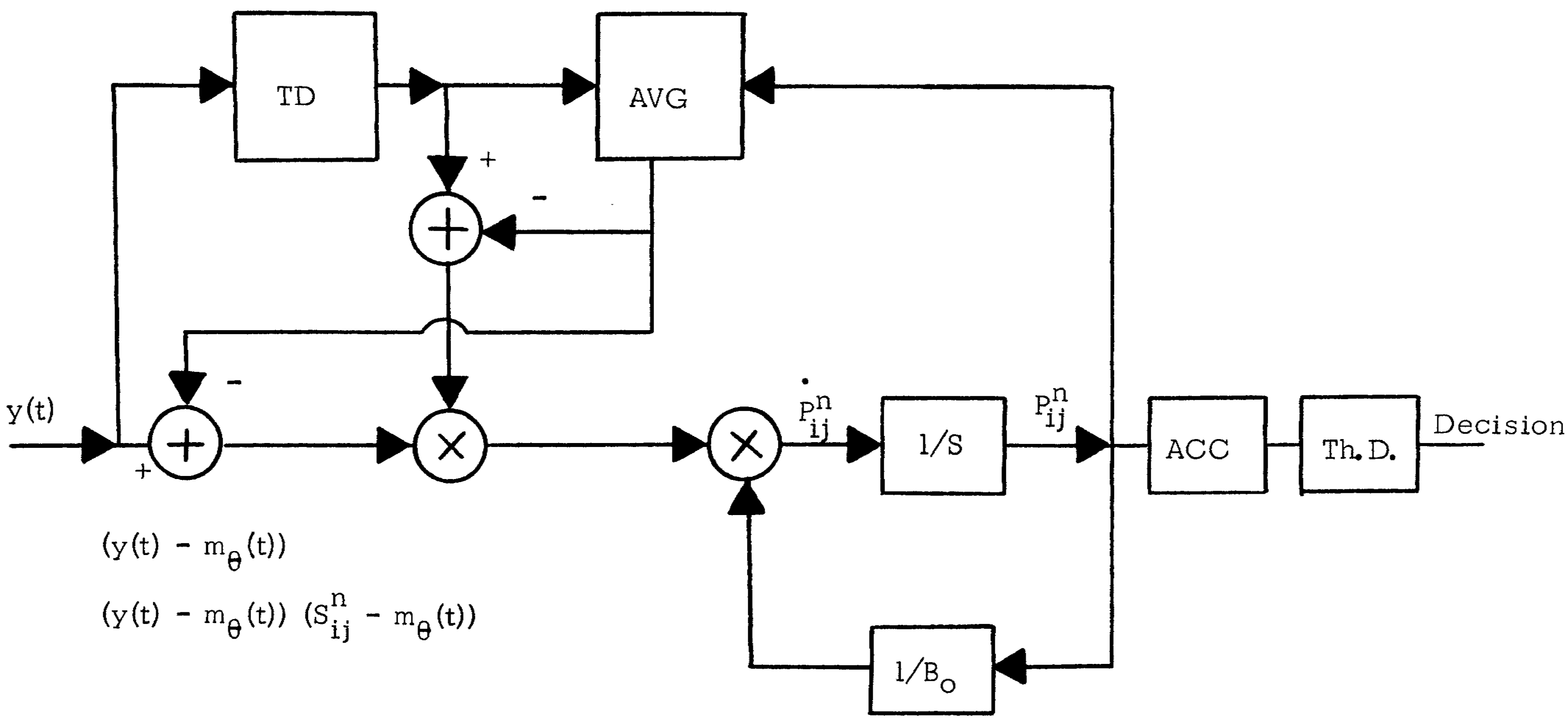



Fig. 7.4 Nonlinear bit synchronizer for overlapping signals

# VIII. CONCLUSIONS AND SUGGESTIONS FOR FURTHER STUDY

Self Bit synchronization techniques for overlapping signals are proposed in this study. The major significance lies in the different approaches used in estimation of the epoch when receiving jittered signals with additive Gaussian white noise. Previous work has concentrated on finding bit synchronizers for anti-correlated signals for which the symbol in one time interval does not overlap with the others.

With the concept of maximum likelihood principle, the ML synchronizer for estimating epoch was derived in a recursive manner. For practical purposes, the Decision-Directed (DD) feedback technique and the matched derivative filter (MDF) technique are proposed particularly for overlapping signals. The DD technique is applicable for large overlapping parameter, $\alpha$, and at high SNR. It is shown that the analytical and simulational performance of the DD detector are in good agreement. The matched derivative filter technique and the transition detector technique are very useful when the received signal has a large $\alpha$.

All the techniques developed in this study can be easily modified to apply to PCM/Split-phase signals. For split-phase signals, one can expect three times as many transition instants, on the average, than the NRZ case. Thus, if a long sequence of logical zero or logical ones is received, the transition detector for the split-phase case can provide more knowledge to the synchronizer.

The overlapping parameter, $\alpha$, is assumed smaller than half of the bit interval so that one symbol will only be overlapped with the two adjacent bits. If this restriction on the parameter $\alpha$ is relaxed, one symbol can overlap with four other symbols. For the optimum synchronizer, a large $\alpha$ can lead into a much more complicated mechanization. Hence, the suboptimum approaches are preferred in this case. For the DD detector portion, the technique should be modified to subtract the overlapping head from the two preceding bits and the overlapping tail from the two following bits. Further work in this area is suggested.

BIBLIOGRAPHY


1. Wintz, P. A., and Hancock, J. C., "An Adaptive Receiver Approach to the Time Synchronization Problem", IEEE Trans. on Comm. Tech., vol. COM-13, No. 1, pp. 90-96, March 1965.

2. Van Horn, J. H., "A Theoretical Synchronization System for Use with Noisy Digital Signals", IEEE Trans. on Comm. Tech., vol. COM-12, No. 3, pp. 82-90, September 1964.

3. Stiffler, J. J., "Maximum Likelihood Symbol Synchronization", JPL Space Programs Summary, vol. IV, No. 3735, pp. 349-357, October 1965.

4. Stiffler, J. J., Theory of Synchronous Communications, Prentice Hall, 1971.

5. Wintz, P. A., and Luecke, E. J., "Performance of Self Bit Synchronization Systems", Report of the School of Electrical Engineering, Purdue University, June 1968.

6. Wintz, P. A., and Luecke, E. J., "Performance of Optimum and Suboptimum Synchronizers", IEEE Trans. on Comm. Tech., vol. COM-17, No. 3, pp. 380-389, June 1966.

7. Golomb, S. W., Davey, J. R., Reed, I. S., Van Trees, H. L., Stiffler, J. J., " Synchronization", IEEE Trans. on Comm. Syst., vol. CS-11, No. 4, pp. 481-491, December 1963.

8. Lindsey, W. C., "Phase-Shifted-Keyed Signal Detection with Noisy Reference Signals", IEEE Trans. on Aero. and Elec. Syst., vol. AES-2, No. 4, pp. 393-401, July 1966.

9. Simon, M. K., "An Analysis of the Steady State Phase Noise Performance of a Digital Data-Transition Tracking Loop", JPL Internal Document, November 21, 1968.

10. Simon, M. K., "Nonlinear Analysis of an Absolute Value Type of Early-Late-Gate Bit Synchronizer", JPL Internal Docu., December 4, 1968.

11. Proakis, J. G., Drouilhet, P. R., Jr., and Price, R., "Performance of Coherent Detection Systems Using Decision-Directed Channel Measurement", IEEE Trans. on Comm. Syst., pp. 54-63, March 1964.

12. Oberst, J. F., "Binary Phase-Shifted-Keyed Communication Systems", Ph.D. Dissertation, Polytechnic Institute of Brooklyn, June 1969.

13. Oberst, J. F., and Schilling, P. R., "Performance of Self-Synchronized Phase-Shifted-Keyed Systems", IEEE Trans. on Comm. Tech., vol. COM-17, No. 6, pp. 664-669, December 1969.

14. Wozencraft, J. M., and Jacobs, I. M., Principles of Communication Engineering, Wiley, January 1967.

15. Martinides, H. F., and Reijns, G. L., "Influence of Bandwidth Restrictions on the Signal-to-Noise Performance on the PCM/NRZ Signal", IEEE Trans. on Aero. and Elec. Syst., vol. AES-4, pp. 35-40, January 1968.

16. Park, J. H., "Effects of Band Limiting on the Detection of Binary Signals", IEEE Trans. on Aero. and Elec. Syst., vol. AES-5, pp. 867-869, September 1969.

17. Shehadeh, N. M., and Tu, K., "The Effect of Band Limiting of a PCM/NRZ Signal on the Bit-Error Probability Using a Sample Detector", Proc. IEEE (letters), vol. 58, pp. 1400-1401, September 1970.

18. Shehadeh, N. M., and Tu, K., Comments on "Effects of Band Limiting on the Detection of Binary Signals", IEEE Trans. on Aero. and Elec. Syst., vol. AES-7, No. 4, July 1971.

19. Tu, K., and Shehadeh, N. M., "Effects of Band Limiting on the Detection of PCM/Split-Phase Signals", Proc. IEEE, pp. 91-93, January 1971.

20. Saltzberg, B. R., and Kurz, L., "Design of Band Limited Signals for Binary Communication Using Simple Correlation Detection", BSTJ, vol. 44, pp. 235-252, February 1965.

21. Saltzberg, B. R., "Intersymbol Interference Error Bounds with Application to Ideal Bandlimited Signaling", IEEE Trans. on Information Theory, vol. IT-14, pp. 563-568, July 1968.

22. Lucky, R. W., "A Functional Analysis Relating Delay Variation and Intersymbol Interference in Data Transmission", BSTJ, vol. 42, pp. 2427-2483, September 1963.

23. Hartman, H. P., "Degration of Signal-to-Noise Ratio Due to IF Filtering", IEEE Trans. on Aero. and Elec. Syst., vol. AES-5, pp. 22-32, January 1969.

24. Mengali, U., "A Self Bit Synchronizer Matched to the Signal Shape", IEEE Trans. on Aero. and Elec. Syst., vol. AES-7, No. 4, July 1971.

25. Bennett, W. R., and Davey, J. R., Data Transmission, McGraw Hill, pp. 119-120, 1965.

26. Lucky, R. W., Salz, J., and Weldon, E. J., Jr., Principles of Data Communication, McGraw Hill, 1968.

27. Webb, A. R., "Split-Phase Pulse Code Modulation", M.S. Thesis, University of Missouri-Rolla, 1970.

28. Hu, K. C., "A Study of Early-Late Gate Type SØ-PCM Signal Bit Synchronizers", M.S. Thesis, University of Missouri-Rolla, 1971

29. Van Trees, H. L., Detection, Estimation, and Modulation Theory, Part II, Wiley, 1971.

30. Kaneko, H., "A Statistical Analysis of the Synchronization of a Binary Receiver", IEEE Trans. on Comm. Syst., pp. 498-501, December 1963.

31. Wintz, P. A., "Optimum Adative Reception for Binary Sequences", IEEE Trans. on Aero. and Elec. Syst., vol. AES-6, No. 3, May 1970.

32. Viterbi, A. J., "Phase-Locked Loop Dynamics in the Presence of Noise by Fokker-Planck Techniques", Proc. IEEE, pp. 1737-1753, December 1963.

33. Simon, M. K., "On the Equivalence in Performance of Several Phase-Locked Loop Configurations", IEEE Trans. on Comm. Tech., pp. 449-452, August 1970.

34. Kushner, H. J., "On the Differential Equations Satisfied by Conditional Probability Densities of Markov Processes with Applications", SIAM Journal on Control, No. 2, 1964.

35. Stratonovich, R. L., Conditional Markov Processes and Their Application to the Theory of Optimal Control, American Elsevier Publishing Company, New York, 1968.

36. Kalman, R. E., and Bucy, R. C., "New Results in Linear Filtering and Prediction Theory", Trans. ASME, J. of Basic Engineering, 83D, 1961.

37. Bucy, R. S., "Nonlinear Filtering Theory", IEEE Trans. on Automatic Control (Correspondence), vol. AC-10, p. 198, April 1965.

38. Ito, K., "Stochastic Integral", Proc. Imp. Acad., v. 20, 1944.

39. Mortenson, R. E., "Mathematical Problems of Modeling Stochastic Nonlinear Dynamic Systems", J. of Stat. Physics, v. 1, pp. 271-296, March 20, 1969.

40. Wonham, W. M., "Some Applications of Stochastic Differential Equations to Optimal Nonlinear Filtering", J. Soc. Ind. Appl. Math., vol. 2, Series A (Control), pp. 347-396, 1964.

41. Fisher, J. R., and Stear, E. B., "Optimal Nonlinear Filtering for Independent Increment Processes", IEEE Trans. on Information Theory, vol. IT-3, No. 4, October 1967.

42. Lee, G. M., "Nonlinear Filtering with Applications to Communication Theory", D. Sc. Dissertation, Dept. of Elec. Engr., Washington University, St. Louis, Missouri, June 1968.

43. Lee, G. M., and Komo, J. K., "PCM Bit Synchronization and Detection by Nonlinear Filtering Theory", IEEE Trans. on Comm. Tech., vol. COM-18. No. 6, December 1970.

44. Doob, J. L., Stochastic Processes, Wiley, New York, 1953.

45. Sage, A. P., and Melsa, J. L., Estimation Theory with Applications to Communications and Control, McGraw Hill, 1971.

46. Sage, A. P., and McBride, A. L., "Optimum Estimation of Bit Synchronization", IEEE Trans. on Aero. and Elec. Syst., pp. 525-536, May 1969.

47. Viterbi, A. J., Principles of Coherent Communication, McGraw Hill, 1966.

48. Dillard, G. M., "Generating Random Numbers Having Probability Distributions Occurring in Signal Detection Problems", IEEE Trans. on Information Theory, pp. 616-617, October 1967.

49. Hull, T. E., and Dobell, A. R., "Random Number Generators", SIAM Rev., vol. 4, pp. 230-254, July 1962.

# VITA

Chung-Tao David Wang was born on December 3, 1943 in Szuchan Province, China. He received the B.S. degree and the M.S. degree both in Electrical Engineering from National Cheng Kung University, Tainan, Taiwan, Republic of China, in 1966 and from the University of Missouri at Rolla, Rolla, Missouri, in 1970, respectively.

He has been enrolled in the graduate school of University of Missouri at Rolla since 1968, and has been a Research Assistant in Electrical Engineering Department for the period from January 1970 to June 1971.

# APPENDIX A

## Random Number Generator Program

Normal random numbers can be generated by a digital computer. There is a large amount of literature on the generation of random numbers[12,27,48]. In general, a uniform distribution on the interval (0,1) is first generated, and it is invaluable in the generation of other distributions. The FORTRAN subroutine RAND (IX,IY,YEL) is used in this study to generate a new random number from the previous one. An initial value is required to start the recurrence relation. The next number is randomly drawn from the finite population of the integers that the computer can produce. At some point a number that has already occurred will be produced thus forming a closed-loop sequence, which continuously cycles from that point on. The length of this sequence is called the period of the generator. It is of the order of the total integer population of the machine.

The subroutine GAUSS(S) in the simulation program uses the result of the subroutine RAND to generate a normal distribution with zero mean and a desired variance "S". The two subroutine are listed in Table A.1.

Table A.1 Subroutine RAND and subroutine GAUSS

```
      SUBROUTINE  RAND(IX,IY,YFL)
      IY=IX*65539
      IF(IY) 1,2,2
    1 IY=IY+2147483647+1
    2 YFL=IY
      YFL=YFL*.4656613E-9
      IX=IY
      RETURN
      END
```

```
      FUNCTION  GAUSS(S)
      INTEGER IX/213711/
      A=-.6
      DO 1  I=1,12
      IF(IY) 5,5,6
    5 IY=IY+2147483647+1
    6 X=IY
      RANDU=X*.4656613E-9
      A=A+RANDU
    1 IX=IY
      GAUSS=A*S
      RETURN
      END
```

# APPENDIX B

## Suboptimum Synchronizer Simulation Program for Overlapping NRZ Signals

```
C   SUBOPTIMUM SYNCHRONIZER SIMULATION PROGRAM, NRZ CASE
          COMMON C, ALFA, T
          REAL X(93), S(93), Z(93), SUM(11), DV(93)
          REAL Y(13), BIT(12), NOISE(93)
          IX = 213711
          SNR = 5.
          BW = 50.
          BLVAR = SQRT(1./(SNR*BW*2.))
          ALFA = 0.25
          SPB = 10.
          L = 13
          G = 0.0
C
          DO 1  I=1,12
          CALL RAND(IX,IY,YFL)
    1     BIT(I) = SIGN(1., YFL-.5)
          WRITE(3,100)  (BIT(I), I=1,12)
  100     FORMAT(/5X,'INPUT DATA=', 12F5.0)
C
          DO 20  I=1,11
          IF(BIT(I)-BIT(I+1)) 20,21,20
   21     G = G+1.
   20     CONTINUE
          WRITE(3,99) G
   99     FORMAT(/5X,'NUMBER OF TRANSITIONS IS', F3.0)
C
          CALL BANDO(Y,L,BW)
          WRITE(3,102)  (Y(I), I=1,13)
  102     FORMAT(/5X,'OUTPUT SAMPLES OF THE FILTER'/(10X,5(5X,E14.7)))
```

```
C
        DO 3  I=1,93
  3     NOISE(I) = GAUSS(BLVAR)
C
        S(1) = BIT(1)
        S(90) = BIT(12)
        S(91) = BIT(12)*.5
        S(92) = BIT(12)*.25
        S(93) = 0.
C
        DO 30  I=2,12
        L = (I-2)*8+2
        K = L+7
        DO 30  J = L,K
        IF(BIT(I)-BIT(I-1)) 32,33,32
  32    IF(J .GT. L+1) GO TO 34
        S(J) = BIT(I-1)
        GO TO 30
  34    IF(J .GT. L+4) GO TO 35
        S(J) =(BIT(I-1)-BIT(I))*.5*((L+3)-J)*.5
        GO TO 30
  35    S(J) = BIT(I)
        GO TO 30
  33    S(J) = BIT(I-1)
  30    CONTINUE
        WRITE(3,1000)  (S(J), J=1,93)
 1000   FORMAT(/5X,'SIGNAL SAMPLES='/(10X,10(F10.5)))
C
        DV(1) = 0.
        DV(90) = BIT(12)
        DV(91) = BIT(12)
```

```
      DV(92) = BIT(12)
      DV(93) = BIT(12)
      DO 60  I = 2,12
      L = (I-2) * 8 + 2
      K = L + 7
      DO 60  J=L,K
      IF(J.LT.L+1 .OR. J.GE.K-1) GO TO 61
      DV(J) =(BIT(I)-BIT(I-1))*.5
      GO TO 60
 61   DV(J) = 0.
 60   CONTINUE
      WRITE(3,999)  (DV(I), I=1.93)
 999  FORMAT(/5X,'DERIV. OUTPUT='/(10X,20(F4.1)))
C
      X(1) = Y(7)*BIT(1)
      X(90) = BIT(12)*Y(10)
      X(91) = BIT(12)*Y(11)
      X(92) = BIT(12)*Y(12)
      X(93) = Y(13)
      DO 40  I=2,12
      K = (I-1)*8+1
      L = (I-2)*8+2
      M = 3
      DO 40  J=L,K
      IF(BIT(I)-BIT(I-1)) 42,43,42
 42   IF(J .GT. L) GO TO 44
      X(J) = BIT(I-1)*Y(8)
      GO TO 40
 44   IF(J .GT. K-2) GO TO 45
      X(J) = BIT(I)*Y(M-2) + BIT(I-1)*Y(M+6)
      M = M+1
```

```
      GO TO 40
   45 X(J) = BIT(I)*Y(M-2)
      M = M+1
      GO TO 40
   43 X(J) = BIT(I)
   40 CONTINUE
      WRITE(3,1001)  (X(J), J=1,93)
 1001 FORMAT(/5X,'BANDLIMITED SIGNAL SAMPLES='/(10X,5(E15.7)))
C
      THETA = 0.
      Z(1) = 0.
      Z(2) = 0.
      Z(3) = 0.
      Z(4) = 0.
      DO 51  J=1,5
C
      DO 50  I=5,89
      K = J-1
   50 Z(I) = (S(I)-(X(I-K)+NOISE(I-K)))*DV(I)
      WRITE(3,1002)  (Z(I), I=5,89)
 1002 FORMAT(/5X,'OUTPUT SAMPLES OF THE SYNCHRONIZER='/(10X,5(E15.7)))
C
      DO 52  II=2,11
      M = (II-2)*8+3
      N = M+4
      SUM(II) = 0.
      DO 53  JJ=M,N
   53 SUM(II) = SUM(II) + Z(JJ)
   52 CONTINUE
      WRITE(3,1003)  (SUM(II), II=2,11)
 1003 FORMAT(/5X,'SUM OF OUTPUT SAMPLES IN EACH INTERVAL='/(10X,5(E15.7)))
```

```
C
      EST = 0.
      DO 54 II=3,11
   54 EST = EST + SUM(II)
      EST = EST/(G-1.)
      WRITE(3,1004) THETA,EST
 1004 FORMAT(/5X,'THETA=',F10.5,5X,'ESTIMATED VALUE OF THETA IS' E14.7)
C
      THETA = THETA+0.125
   51 CONTINUE
      STOP
      END

      FUNCTION F(X)
      COMMON C,ALFA,T
      F = C*(SIN(X)/X)*(SIN(2.*ALFA*X)/(2.*ALFA*X))*COS(X*(1.-2.*T))
      RETURN
      END

      SUBROUTINE  BANDO(Y,L,BW)
      COMMON C,ALFA,T
      REAL Y(13)
      T = 0.
      N2 = 100
      A = 0.
      B = 3.1416*BW
      C = 2./3.1416
      N = N2/2
      H = (B-A)/FLOAT(N2)
      M = L-2
      DO 2  J=1,M
      S = C+4.*F(A+H)
```

```
      N1 = N-1
      DO 3  I=1,N1
3     S = 2.*F(A+H*FLOAT(2*I)) + 4.*F(A+H*FLOAT(2*I+1)) + S
      S = H*(F(B)+S)/3.0
      K = J+2
      Y(K) =S
      T = T+0.125
2     CONTINUE
      Y(1) = Y(13)
      Y(2) = Y(12)
      RETURN
      END

      FUNCTION GAUSS(S)
      INTEGER  IX/213711/
      A = -6.
      DO 1  I=1,12
      IY = IX*65539
      IF(IY) 5,5,6
5     IY = IY+2147483647+1
6     X = IY
      RANDU = X*.4656613E-9
      A = A+RANDU
1     IX = IY
      GAUSS = A*S
      RETURN
      END

      SUBROUTINE RAND(IX,IY,YFL)
      IY =IX*65539
      IF(IY) 1,2,2
```

```
  1       IY = IY+2147483647+1
  2       YFL = IY
          YFL = YFL*.4656613E-9
          IX = IY
          RETURN
          END
/DATA
/END
```

# APPENDIX C

## Suboptimum Synchronizer Simulation Program for Overlapping Split-Phase Signals

```
C  SUBOPTIMUM SYNCHRONIZER SIMULATION PROGRAM,SPLIT-PHASE CASE
      REAL  BIT(12), Y(21), S(177),DV(177),X(177),Z(177),NOISE(190)
      COMMON  C,ALFA,T
      IX = 213711
      ALFA = 0.125
      SNR = 5.
      BW = 50.
      BLVAR = SQRT(1./(SNR*BW*2.))
      L = 21
      S(1) = 0.
      G = 0.
C
      DO 1  I=1,12
      CALL RAND(IX,IY,YFL)
 1    BIT(I) = SIGN(1., YFL-.5)
      WRITE(3,100)  (BIT(I), I=1,12)
100   FORMAT('1', 'INPUT SEQUENCE=',12F8.0)
C
      DO 20  I=2,11
      IF(BIT(I)-BIT(I+1)) 21,22,21
 21   G = G+1.
      GO TO 20
 22   G = G+2.
 20   CONTINUE
      WRITE(3,99) G
 99   FORMAT(/5X,'NUMBER OF TRANSITIONS IS'5X,F4.0)
C
      DO 2  I=1,190
 2    NOISE(I) = GAUSS(BLVAR)
C
      CALL BANDO(Y,L,BW)
      WRITE(3,102)  (Y(I), I=1,21)
102   FORMAT(//5X,'BANDLIMITED SYMBOL SAMPLES'/(10X,5(5X,E14.7)))
```

```
C
      DO 30  I=2,12
      L = (I-2)*16+2
      K = L+15
      DO 30  J=L,K
      IF(BIT(I)-BIT(I-1)) 32,33,32
   32 NN = J-L+1
      GO TO (34,35,35,35,35,35,35,35,35,35,35,35,35,35,34,36),NN
   34 S(J) = -BIT(I-1)*.5
      GO TO 30
   35 S(J) = -BIT(I-1)
      GO TO 30
   36 S(J) = 0.
      GO TO 30
   33 MM = J-L+1
      GO TO (4,5,5,5,5,5,4,6,7,8,8,8,8,8,7,6),MM
    4 S(J) = -BIT(I)*.5
      GO TO 30
    5 S(J) = -BIT(I)
      GO TO 30
    6 S(J) = 0.
      GO TO 30
    7 S(J) = BIT(I)*.5
      GO TO 30
    8 S(J) = BIT(I)
   30 CONTINUE
      WRITE(3,1000)  (S(J), J=1,177)
 1000 FORMAT(/5X,'SIGNAL SAMPLES='/(10X,10(F10.2)))
C
      DV(1) = -BIT(1)
      DO 60  I=2,12
      L = (I-2)*16+2
```

```
      K = L+15
      DO 60  J=L,K
      IF(J .GT. L+1) GO TO 61
      DV(J) = -BIT(I-1)
      GO TO 60
   61 IF(J .LT. L+5 .OR. J .GT. L+9) GO TO 62
      DV(J) =(BIT(I)+BIT(I-1))*.5
      GO TO 60
   62 IF(J .LT. L+13) GO TO 63
      DV(J) = -BIT(I)
      GO TO 60
   63 DV(J) = 0.
   60 CONTINUE
      WRITE(3,999)  (DV(I), I=1,177)
  999 FORMAT(/5X,'DERIVATIVE OUTPUT=',/(10X,10(F10.2)))
C
      X(1) = Y(11)
      DO 40  I=2,12
      L = (I-2)*16+2
      K = L+15
      M = 1
      DO 40  J=L,K
      MM = J-L+1
      GO TO (41,41,41,41,41,42,42,42,42,42,43,43,43,43,43,43),MM
   41 X(J) = Y(MM+11)*(-BIT(I-1))
      GO TO 40
   42 X(J) = -BIT(I-1)*Y(MM+11) + BIT(I)*Y(M)
      M = M+1
      GO TO 40
   43 X(J) = Y(M)*BIT(I)
      M = M+1
   40 CONTINUE
```

```
      WRITE(3,1004)  (X(I), I=1,177)
 1004 FORMAT(/5X,'OUTPUT SAMPLES OF THE FILTER'/(10X,5(5X,E14.7)))
C
      THETA = 0.
      DO 51  J=1,10
C
      DO 50  I=10,177
      K = J-1
 50   Z(I) = (S(I)-(X(I-K)+NOISE(I-K)))*DV(I)
      WRITE(3,1005)  (Z(I), I=10,177)
 1005 FORMAT('1',5X,'DETECTED SAMPLE='/(10X,5(5X,E14.7)))
C
      SUM = 0.
      DO 70  JJ=10,177
 70   SUM = SUM+Z(JJ)
      SUM = SUM/G
      WRITE(3,1006)  THETA, SUM
 1006 FORMAT(///5X,'THETA=',F10.5,5X,'ESTIMATE OF THETA=',E15.7)
C
 51   THETA = THETA+0.0625
      STOP
      END

      SUBROUTINE RAND(IX,IY,YFL)
      IY = IX*65539
      IF(IY) 1,2,2
 1    IY = IY+2147483647+1
 2    YFL = IY
      YFL = YFL*.4656613E-9
      IX = IY
      RETURN
      END
```

```
      SUBROUTINE BANDO(Y,L,BW)
      COMMON C,ALFA,T
      REAL Y(21)
      L = 21
      ALFA = 0.125
      BW = 50
      T = 0.
      N2 = 100
      A = 0.
      B = 3.1416*BW
      C = 2./3.1416
      N = N2/2
      H = (B-A)/FLOAT(N2)
      M = L-2
      DO 2  J=1,M
      F0 = 0.
      S = F0 + 4.*F(A+H)
      N1 = N-1
      DO 3  I=1,N1
3     S = 2.*F(A+H*FLOAT(2*I))+4.*F(A+H*FLOAT(2*I+1)) + S
      S = H*(F(B)+S)/3.0
      K = J+2
      Y(K) = S
      T = T+0.0625
2     CONTINUE
      Y(1) = -Y(21)
      Y(2) = -Y(20)
      RETURN
      END
```

```
      FUNCTION  F(X)
      COMMON C,ALFA,T
      F = C*(SIN(2.*ALFA*X)/(2.*ALFA*X))*((1.-COS(X))/X)*SIN(X*(1.-2.*T))
      RETURN
      END

      FUNCTION GAUSS(S)
      INTEGER IX/213711/
      A = -6.
      DO 1  I=1,12
      IY = IX*65539
      IF(IY) 5,5,6
  5   IY = IY + 2147483647+1
  6   X = IY
      RANDU = X*.4656613E-9
      A = A + RANDU
  1   IX = IY
      GAUSS = A*S
      RETURN
      END
/DATA
/END
```