DIGITAL BIT SYNCHRONIZATION OF HARD LIMITED BINARY DATA

by

PATRICK G. OGMUNDSON
B.Eng., Lakehead University, 1986

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE
in the Department of
Electrical and Computer Engineering

ACCEPTED
FACULTY OF GRADUATE STUDIES
[redacted]
DEAN
DATE Sept 20, 1988

We accept this thesis as conforming
to the required standard

[redacted]
Supervisor Dr. P. F. Driessen

[redacted]
Dr. V. K. Bhargava

[redacted]
Dr. R. L. Kirlin

[redacted]
Dr. N. J. Dimopoulos

[redacted]
Dr. R. N. Horspool

[redacted]
Dr. A. Olin

Supervisor: Dr. P. F. Driessen

ABSTRACT

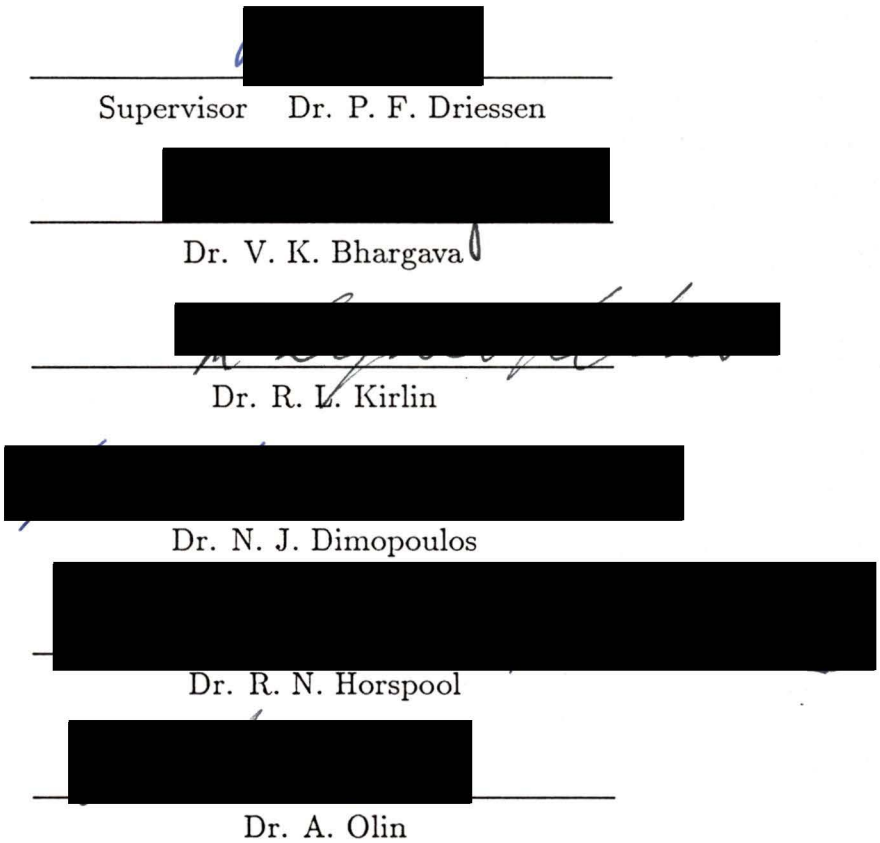
Three new digital bit synchronization techniques are presented. Each synchronizer uses a hard limiter to interface between the received analog data waveform and the digital synchronizer.

The optimum maximum likelihood synchronizers (data aided and non data aided) for hard limited data signals are developed. Linear approximations are made to these optimum synchronizers to simplify implementation. The lower bound on the timing error variance of the hard limited data aided synchronizer is established.

A zero crossing tracking digital phase locked loop (PLL) suitable for bandlimited data transmission systems is proposed. Pattern jitter (self noise) is reduced by a pattern jitter compensation signal in the PLL feedback, thus eliminating the need for a prefilter. The timing jitter variance analysis is shown to be an improvement over a previous result for a zero crossing tracking PLL. Computer simulations and laboratory experiments verify the performance of the pattern jitter compensation technique.

A PLL synchronizer with a simple adaptive bandwidth control is shown to yield a better acquisition/tracking performance tradeoff than a fixed bandwidth PLL. The adaptive bandwidth control operates by monitoring the location of zero crossings measured at the output of a phase detector. Computer simulation and experimental results for some typical PLL parameters are presented.

Examiners:



Supervisor Dr. P. F. Driessen

Dr. V. K. Bhargava

Dr. R. L. Kirlin

Dr. N. J. Dimopoulos

Dr. R. N. Horspool

Dr. A. Olin

Contents

Appendices

List of Tables

List of Figures

Acknowledgements

I would like to thank my thesis supervisor, Dr. Peter F. Driessen, for his encouragement, patience, and advice, for many interesting and helpful discussions, and for help in the preparation of this document. Financial assistance received from Dr. Driessen (through NSERC), and from the University of Victoria, is also gratefully acknowledged.

To my wife, Anita.

Notation

List of Abbreviations

A/D	-	analog to digital
AWGN	-	additive white Gaussian noise
CR	-	Cramér-Rao
CSMA	-	carrier sense multiple access
DA	-	data-aided
DAC	-	digital to analog converter
DPLL	-	digital phase locked loop
DSP	-	digital signal processing
HL	-	hard limited
ISI	-	inter symbol interference
IF	-	intermediate frequency
LPF	-	low pass filter
ML	-	maximum likelihood
NDA	-	non-data-aided
NRZ	-	non return to zero
PLL	-	phase locked loop
psd	-	power spectral density
RMS	-	root mean square
SNR	-	signal to noise ratio
SRC	-	spectral raised cosine
TDMA	-	time division multiple access

List of Principal Symbols

Chapter 2

a_j	=	transmitted binary data bit at $t = jT$, $a_j \in \{-1, 1\}$.
$n(t)$	=	zero mean AWGN.
$\eta(t)$	=	low pass filtered zero mean AWGN.
$x(t)$	=	signalling (symbol) pulse. (2.8).
T	=	data symbol duration.
m	=	number of samples taken by the synchronizer in T seconds.
K	=	number of symbols in the ML observation interval.
$y(t)$	=	data receiver low pass filter output (2.9).
ϵ	=	transmission delay parameter.
$\hat{\epsilon}_{ML}, \hat{\epsilon}_{HL}$	=	synchronizer estimate of ϵ.
B	=	single sided bandwidth of the data receiver LPF (2.10).
σ_η^2	=	variance of the additive noise $\eta(t)$ (2.11).
N_o	=	single sided power spectral density of AWGN
$\mathbf{A}$	=	vector of random variables with elements A_j (2.12).
A_j	=	random variable that may assume a value $a_j \in \{-1, 1\}$
$\mathbf{Y}, \mathbf{Y}_j$	=	vector of discrete time samples of $y(t)$ (2.13)-(2.15).
$\mathbf{X}_j(\epsilon), X_j^i(\epsilon)$	=	vector elements are discrete time samples of $x(t-(j-1)T-\epsilon)$
$\mathbf{N}_j, N_j^i$	=	vector elements are discrete time samples of $\eta(t)$.
$\tau_j(\epsilon)$	=	the j^{th} sub interval in the observation interval (2.26).
T_j	=	duration of $\tau_j(\epsilon)$ (2.27).
E_b	=	energy of $x(t)$.
$v(t)$	=	$\text{sgn}\{y(t)\}$
$\mathbf{V}, \mathbf{V}_j, V_j^i$	=	vector elements are discrete time samples of v(t) (2.40)-(2.41)
P_j^i	=	$P_r[N_j^i > x_j^i]$ (2.47)

Chapter 3

a_k	=	transmitted binary data bit at $t = kT$, $a_k \in \{-1, 1\}$.
$n(t)$	=	zero mean AWGN.
$\eta(t)$	=	low pass filtered zero mean AWGN.
$x(t)$	=	signalling (symbol) pulse.
T	=	data symbol duration.
$s(t)$	=	signal at the LPF_R output (3.5)
l, m	=	number of postcursor and precursor bits respectively in the pattern jitter compensation (3.7).
ϵ, ϵ_k	=	slowly varying delay of the received data.
$\hat{\epsilon}_k$	=	synchronizer estimate of ϵ.
$X(f)$	=	Fourier transform of $x(t)$.
$H_T(f), H_R(f)$	=	frequency response of the LPF_T and LPF_R respectively (figure 3.2).
K_{NCO}, K_{PD}	=	NCO and PD gain constant respectively.
t_o	=	nominal zero crossing location relative to the data sample at $t = kT$.
e_k	=	synchronizer timing error sequence (3.11).
w_k	=	phase detector output.
n_k	=	zero crossing error generated by additive noise (3.13).
c	=	inverse slope of the zero crossing (3.14).
τ_k	=	zero mean ISI generated random variable (3.15).
$\hat{\tau}_k$	=	synchronizer estimate of τ_k (3.16).
b_k	=	data transition detector output (3.17)-(3.18).
$G(z)$	=	PLL feedback transfer function (figure 3.6).
g_n	=	impulse response of $G(z)$.
γ_k	=	$2b_k(\tau_k - \hat{\tau}_k + n_k + \hat{\epsilon}_k)$ (3.22).
$R_{\hat{\epsilon}}(q), R_\gamma(q)$	=	autocorrelation function of $\hat{\epsilon}_k$ and γ_k.
$S_{\hat{\epsilon}}(z), S_\gamma(z)$	=	z transform of $R_{\hat{\epsilon}}(q)$ and $R_\gamma(q)$.
$H(z)$	=	PLL transfer function (3.35).
$S(\omega)$	=	$S(z)\|_{z=e^{j\omega}}$.
$2B_LT$	=	PLL double sided equivalent noise bandwidth (3.40).
$R_I(q), R_N(q), R_D(q)$	=	component of $R_\gamma(q)$ attributable to ISI, $\eta(t)$, and ϵ respectively (3.43).
$S_I(\omega), S_N(\omega), S_D(\omega)$	=	psd of $R_I(q)$, $R_N(q)$, and $R_D(q)$ respectively (3.44).

σ_e^2 = variance of the timing error sequence e_k (3.46).
σ_{pj}^2 = pattern jitter timing error variance (3.47).
σ_{nj}^2 = additive noise timing error variance (3.65), (3.75), (3.78)
σ_η^2 = variance of the additive noise $\eta(t)$.
E_b = bit energy at the input to the LPF_R.
N_o = single sided power spectral density of the AWGN.
β = stochastic gradient algorithm adaptation gain constant.
K = $K_{PD}K_{NCO}$.
α = SRC excess bandwidth parameter (3.81).

Chapter 4

a_k = transmitted binary data bit at $t = kT$, $a_k \in \{-1, 1\}$
$\hat{a}_k$ = estimated binary data bit.
$\eta(t)$ = low pass filtered zero mean AWGN
$x(t)$ = signalling (symbol) pulse.
T = data symbol duration.
ϵ_k = slowly varying delay of the received data.
$\hat{\epsilon}_k$ = synchronizer estimate of ϵ.
e_k = synchronizer timing error (4.2)
b_k = transition detector (4.4)
w_k = PD output sequence.
K_k = adaptive bandwidth PLL loop gain value.
$K^{(i)}$ = $K_k \in \{K^{(i)}\}$ (4.7).
N = number of states (loop gain values) in the Markov chain.
T_e = synchronizer threshold parameter (4.8).
P_o = probability of no data transition detected (4.11).
P_+ = probability of increasing the synchronizer bandwidth (4.12).
P_- = probability of decreasing the synchronizer bandwidth (4.13).
Q_i = steady state probability density of loop gain values $K^{(i)}$ (4.14).
L = number of maximum bandwidth states in the Markov chain.
$2B_LT$ = double sided noise bandwidth of the PLL transfer function (4.17).
E_b = bit energy at the input to the LPF_R.
N_o = single sided power spectral density of the AWGN.

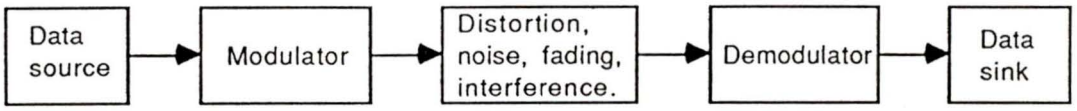
Chapter 1

Introduction

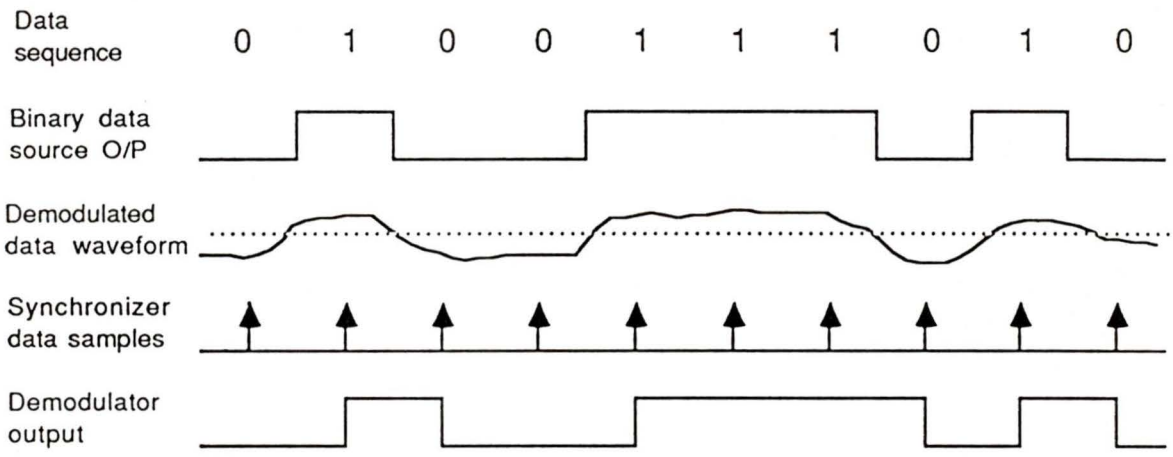
1.1 Introduction

Recently there has been a major increase in the demand for digital radio systems, particularly for land mobile radio services such as computerized radio dispatch, paging, and vehicle location systems. The next generation of cellular telephone systems is almost certain to be digital [1] and will likely provide data services compatible with the emerging Integrated Services Digital Network [2]. The emphasis on future modulator/demodulator (modem) designs may be oriented toward low-cost digital circuit implementations.

A data transmission system demodulator requires a synchronizer to estimates the time of arrival of data symbols. The synchronization process is illustrated in figure 1.1. A typical data transmission system (figure 1.1a) consists of a data source, a modulator, a transmission medium, a demodulator and a data sink. During transmission the transmitted waveform is subjected to such impairments as distortion, noise, channel fading, and



a) Block diagram of a data transmission system.



b) Synchronization timing waveforms.

Figure 1.1: Ilustration of synchronization in a binary data transmission system.

interference. In the demodulator a synchronizer samples the demodulated data waveform and a hard decision is made to estimate the transmitted data bit (figure 1.1b). The location of the synchronizer data samples is selected to minimizing the probability of incorrect bit decisions (i.e. the bit error probability). The need for reliable and accurate synchronization is crucial for minimizing the error probability and maximizing data throughput.

With the growing popularity of digital signal processing techniques, the emphasis of recent synchronizer literature has been oriented towards discrete time sampled implementations involving an analog to digital (A/D) converter and a digital signal processing integrated circuit [3]-[7]. However for timing measurement systems such as synchronizers, amplitude-based systems often "stem from tradition rather than need"[8] and the resulting system may be either inappropriate, overly complex, too slow, or too costly.

For a binary data receiver, a simple, high speed, low cost interface between the received analog data waveform and the signal processing circuitry is a hard limiter, or equivalently a one bit A/D converter. The hard limiting operation assigns one binary number to analog values above a threshold, and the binary complement to analog values below the threshold. A hard limiter eliminates the need for an expensive and usually slow multi-bit A/D converter. Other advantages of hard limiting are that accurate automatic gain control is not required, and the susceptibility to impulse noise is not as great as for amplitude based systems. However hard limited synchronization systems are suboptimal with performance degradations that should be identified.

The focus of this thesis is the development of hard limited binary data

synchronization techniques that are suitable for low cost digital circuit implementation. Three new synchronization techniques are presented

1. Optimum and suboptimum hard limited synchronizer structures for use in conjunction with hard limited binary detectors are developed in chapter 2.

2. A digital phase locked loop (PLL) synchronizer for near Nyquist bandwidth limited operation is proposed and evaluated in chapter 3.

3. A digital PLL synchronizer with rapid acquisition and a small mean square tracking error for time division multiple access (TDMA) and carrier sense multiple access (CSMA) packet radio applications is evaluated in chapter 4.

1.2 Literature Survey in Synchronization

In this section a literature survey and brief description of synchronization techniques is presented. Single chapters dealing with synchronization are contained in the books by Lindsey and Simon [9], Bhargava et al [10], Ziemer and Peterson [11], Holmes [12], Feher [13], and Spilker [14]. A tutorial paper by Franks [15] also contains an overview of synchronization.

Upon reception of the data signal, the synchronizer must quickly and accurately synchronize a locally generated timing waveform with the incoming data stream. The ability of the synchronizer to achieve the initial synchronization determines the "acquisition" performance of the synchronizer. Once synchronization has been achieved, the synchronizer "jitter" or

"tracking" performance measures how well the synchronizer tracks the incoming data. The third synchronization parameter of interest is the "cycle slip" performance, which measures the propensity for the synchronizer (due to noise and/or frequency offsets between the transmitter and receiver) to mistakenly add or delete a data symbol.

Synchronization is most often derived from the incoming data signal directly (data derived synchronization), thereby eliminating the inefficiencies associated with allocating power or bandwidth for a separate synchronization signal.

The problem of binary data synchronization was initially formulated as a maximum likelihood (ML) estimation problem by Wintz and Luecke [16]. Although the resulting optimum synchronization model is generally not considered to be practical to implement, the analysis has provided motivation for several PLL implementations [9,11]. The ML estimation analysis can also be extended to non-binary synchronization applications [4,17].

Two synchronization techniques that use a PLL to either track the zero crossings of a received demodulated data waveform, or track the zeros of the derivative of the waveform are considered in [18].

Digital communications systems that operate close to the minimum (Nyquist) bandwidth are hampered by intersymbol interference (ISI). ISI is caused by the slowly decaying tails of bandwidth limited data symbols interfering with adjacent data symbols [25]. A popular synchronization technique for this application is to pass the received data signal through a nonlinear device to generate a sinusoidal spectral component (at the symbol rate) in the output, and to recover this timing waveform with either a

PLL or a bandpass filter [15,16]. Another technique is to use a delay and multiply circuit preceding either a PLL or bandpass filter [19]. A digital delay and multiply circuit has also been evaluated [20]. A significant problem associated with bandwidth limited synchronization is the presence of a "pattern jitter" or "self noise" component in the recovered timing waveform that is directly related to the ISI [15,21,22]. Appropriate prefiltering techniques may be used to reduce (and for square law nonlinearities possibly eliminate) the pattern jitter in a nonlinearity and PLL (or bandpass filter) synchronizer [23].

The popularity of digital signal processing has led to the development of several digital synchronization techniques. One type of digital synchronizer involves sampling the output of a matched filter at the symbol rate, A/D conversion and nonlinear processing of the sampled value, followed by a digital PLL system [4,5]. Another class of digital synchronizers operate on a timing error estimate generated from a linear combination of the samples and the recovered transmitted data [6,7]. A digital filter equivalent to the nonlinearity and bandpass filter synchronizer has also been investigated [3].

For binary synchronization applications, an attractive synchronization method in terms of both ease of implementation and low cost is the use of a binary quantized digital PLL synchronizer [24] which essentially tracks the zero crossing location of the received data waveform based on binary quantized (lead, lag) phase error measurements. Binary quantized PLL structures require no multiplication operations and therefore are suitable for single chip implementation using microcontrollers or discrete logic.

1.3 Contributions in this Thesis

Three synchronization techniques that operate on a hard limited data signal are considered in this thesis. The synchronizers are appropriate for binary antipodal linear modulation systems, although the timing error detection techniques may may be extended to quadrature modulation with binary modulation in each quadrature path [25].

The attractiveness of hard limiting binary signalling waveforms for digital circuit implementation has led to recent interest in the performance of hard limited binary detection systems as a substitute for the optimum digital matched filter [26]-[32]. In chapter 2 a ML hard limited synchronization scheme that may be used in conjunction with a hard limited binary detector is developed. The optimum (in the ML sense) data-aided and non-data-aided synchronizers are presented, and the Cramér-Rao lower bound on the timing error variance is established. Sub-optimal versions of the optimum synchronizers are derived and their performance is verified with a simulation.

Another hard limited synchronization method that has previously been evaluated is a PLL implementation that tracks the zero crossings of the data waveform [18]. For bandlimited operation ISI will generate a significant pattern jitter component in the recovered timing waveform of a zero crossing tracking synchronizer. In chapter 3, an all digital zero crossing tracking PLL synchronizer is evaluated for bandlimited operation and a new technique to reduce pattern jitter (called pattern jitter compensation) is proposed and evaluated. The timing error variance analysis is shown to be an improvement over a previous result [18] and the new compensation

technique is incorporated into the analysis. Simulation and experimental results confirm the performance of the pattern jitter compensation technique.

In TDMA and CSMA packet radio systems, the delay associated with the initial synchronization is overhead that will reduce the maximum data throughput of the system [33]. Therefore rapid acquisition synchronization is required. For fixed parameter synchronizers there is generally a tradeoff between acquisition and tracking performance. In chapter 4 a digital PLL synchronizer with a simple adaptive bandwidth control is considered. The input to the PLL is a hard limited binary data waveform. The adaptive bandwidth control automatically adjusts the PLL parameters for rapid acquisition when initial synchronization is required, and a low mean square tracking error once synchronization has been achieved. The synchronizer performance is verified with a simulation study and laboratory experiment results.

1.4 Outline of the Thesis

The formulation and solution to the hard limited ML synchronization problem is presented in chapter 2, and a comparison with non-hard limited ML synchronizers is made. The resulting synchronizer structures are simplified with linear approximations. A lower bound on the timing error variance is also established. Computer simulation measurements of the synchronizer timing error variance for the simplified hard limited synchronizer structures are compared to the optimum non-hard limited synchronizers, and are also compared to the lower bound calculation.

Chapter 3 contains the synchronizer description and jitter variance analysis of the pattern jitter compensation PLL synchronizer. Simulation results verify the performance of the proposed synchronization technique. Experimental results are also presented.

The adaptive bandwidth synchronizer is considered in chapter 4. A description of the synchronizer operation is presented, and simulation results for some typical applications are presented. Laboratory experiments confirm the operation of the synchronizer.

Conclusions and some further research recommendations are made in chapter 5.

Chapter 2

Maximum Likelihood Synchronization of Hard Limited Binary Data

2.1 Foreword

Digital bit synchronization of a sampled hard limited binary data waveform is considered. The optimum (in the maximum likelihood sense) open loop synchronizers are presented for both data-aided and non-data-aided synchronization. The optimum hard limited synchronizers may not be practical to implement, and so simpler realizations (based on linear approximations that are valid for a high sampling rate) are derived. These open loop synchronizers can also be adapted for closed loop operation. The Cramér-Rao lower bound on the timing error variance is established for the optimum hard limited synchronizer, and it is shown that the lower bound is a fixed constant ($\pi/2$) higher than the lower bound for the non hard limited optimum synchronizer. Computer simulations of the hard limited synchronizer

verify this result.

2.2 Introduction

A digital implementation of a binary data receiver can be greatly simplified if the received analog waveform is hard limited (HL) before being sampled and processed. The requirement of an expensive and usually slow analog to digital converter is eliminated and a digital multiplier may not be needed. However, associated with these advantages are performance penalties that must be identified. A number of recent papers have analyzed the performance of HL detection systems [26]-[32]. The penalty in terms of probability of bit error performance relative to the optimum digital matched filter has been determined for the optimum hard limited binary detection scheme [28], and also for other suboptimal but more implementable detection schemes [29].

For a system that incorporates a hard limited detector, it is reasonable to expect that the synchronization information will be recovered from the hard limited waveform as well. The performance of a HL synchronization system relative to an optimum system is therefore of interest.

In this chapter the principle of maximum likelihood (ML) estimation is applied to a binary synchronization system that operates entirely on hard limited data. Equations that suggest synchronization algorithms for both non-data-aided (NDA) and data-aided (DA) systems are presented. These algorithms are similar in structure to the optimum open loop synchronization systems [16,34,35], which form the basis for closed loop realizations

[9,11]. It should also be noted that an open loop synchronizer using a parallel search for the ML estimate has superior acquisition performance when compared to a closed loop synchronizer. Often the synchronizer acquisition performance is a very important parameter, for example in a severe fading environment like mobile radio. For these applications the parallel search open loop synchronizer may be appropriate, and recently there has been some interest shown in implementing the open loop structure using parallel digital signal processing techniques [36].

The optimum HL-DA and HL-NDA systems may not be practical to implement, and so the optimum equations are simplified with linear approximations that are valid for a high sampling rate.

A lower bound on the timing error variance is established for the optimum HL-DA synchronizer. A computer simulation is used to compare the performance of the HL-DA and HL-NDA algorithms based on the linear approximations.

This chapter is organized as follows: a brief review of ML estimation as it relates to synchronization is presented in section 2.3. The binary synchronization problem is formulated in section 2.4, and previous ML synchronization results are restated in section 2.5. Section 2.6 is the derivation of the optimum HL synchronization equations and the HL synchronizer timing error variance lower bound calculation. These new results are compared with the previous ML synchronization results. The computer simulation results are presented in section 2.7, and section 2.8 is the chapter summary.

2.3 Maximum Likelihood Estimation

The principle of ML estimation is often used in the development of synchronization systems [11]. ML estimation is discussed in [37], and a description of ML estimation as it relates to synchronization is presented in this section.

In the estimation of a single parameter ϵ, the objective of ML estimation is to generate an estimate $\hat{\epsilon}_{ML}$ from the received data vector $\mathbf{Y}$ such that the conditional density $f_{\mathrm{Y}|\hat{\epsilon}}(\mathbf{y}|\epsilon)$[1] satisfies

$$f_{\mathrm{Y}|\epsilon}(\mathbf{y}|\epsilon)|_{\epsilon=\hat{\epsilon}_{ML}} = \text{maximum} \tag{2.1}$$

The maximum likelihood estimate is particularly useful in situations where little is known about the distribution $f_{\epsilon}(\epsilon)$, which is generally the case in bit synchronization applications. For applications where $f_{\epsilon}(\epsilon)$ is known, the maximum a posteriori (MAP) estimate is applicable [37].

Necessary (but not sufficient) conditions for the ML estimate are

$$\frac{\partial f_{\mathrm{Y}|\epsilon}(\mathbf{y}|\epsilon)}{\partial \epsilon} |_{\epsilon=\hat{\epsilon}_{ML}} = 0 \tag{2.2}$$

and

$$\frac{\partial \ln f_{\mathrm{Y}|\epsilon}(\mathbf{y}|\epsilon)}{\partial \epsilon} |_{\epsilon=\hat{\epsilon}_{ML}} = 0 \tag{2.3}$$

A number of synchronization techniques have been developed that use (2.2) or (2.3) to generate an error signal that is forced to zero in a closed loop feedback system [9].

[1] In this chapter, upper case characters are generally used to represent a random variable, while lower case characters are used to identify a particular value of the random variable.

The variance of $\hat{\epsilon}_{ML}$ may be difficult to determine, but in many cases a lower bound on the variance may be established using the Cramér-Rao (CR) bound. If $\hat{\epsilon}_{ML}$ is unbiased, i.e.

$$E[\hat{\epsilon}_{ML}|\epsilon] = \epsilon \tag{2.4}$$

then the CR bound is

$$E[\hat{\epsilon}_{ML} - \epsilon)^2] \geq -\left\{ E\left[\left(\frac{\partial \ln f_{\mathrm{Y}|\epsilon}(\mathbf{y}|\epsilon)}{\partial \epsilon} \right)^2 \right] \right\}^{-1} \tag{2.5}$$

or equivalently

$$E[\hat{\epsilon}_{ML} - \epsilon)^2] \geq -\left\{ E\left[\left(\frac{\partial^2 \ln f_{\mathrm{Y}|\epsilon}(\mathbf{y}|\epsilon)}{\partial \epsilon^2} \right)^2 \right] \right\}^{-1} \tag{2.6}$$

If (2.5) and (2.6) are satisfied with equality, then $\hat{\epsilon}_{ML}$ is said to be an efficient estimate. It can be shown that if an estimate is efficient it is the ML estimate, and if the observations in $\mathbf{Y}$ are independent, as the number observations in $\mathbf{Y}$ becomes large, ML estimates are Gaussian, unbiased, and efficient [37].

2.4 Problem Formulation

The delay ϵ of a received binary data waveform corrupted by additive white Gaussian noise is to be estimated. It is assumed that ϵ is uniformly distributed over the bit duration $(0, T)$. The estimate $\hat{\epsilon}_{ML}$ is to be determined after an observation interval of KT seconds. The received waveform during

the observation interval is

$$w(t) = \sum_{j=0}^{K} a_j x(t - (j-1)T - \epsilon) + n(t) \tag{2.7}$$

where $a_j \in \{-1, +1\}$ are zero mean statistically independent binary data, $n(t)$ is zero mean additive white Gaussian noise with two sided power spectral density $N_o/2$, and $x(t)$ is the strictly time limited transmitted symbol waveform

$$x(t) = \begin{cases} x(t) & \text{for } t \in (0, T) \\ 0 & \text{for } t \notin (0, T) \end{cases} \tag{2.8}$$

The effects of intersymbol interference (ISI) due to bandlimiting the $x(t)$ pulses are not addressed in this analysis. It is also assumed that the parameter ϵ is constant over KT seconds. The observation interval contains the last ϵ seconds of the $j = 0$ symbol, the next $K - 1$ complete symbols, and the first $T - \epsilon$ seconds of the K^{th} symbol.

The conditional density function in (2.1) requires that the received data be expressed in vector form. This is generally achieved by representing the waveforms $x(t)$ and $n(t)$ by the coefficients of a complete set of orthonormal basis functions defined on $(0, T)$ [37]. For a digital synchronizer implementation it is of interest to use discrete time samples to represent the waveforms rather than orthonormal expansion coefficients. Using this method, the effects of a finite sample size m per bit interval T on the performance of the digital synchronizer can be determined.

Consider the synchronizer model in figure 2.1. The received waveform $w(t)$ is filtered by the low pass filter. It is assumed that the bandwidth B is

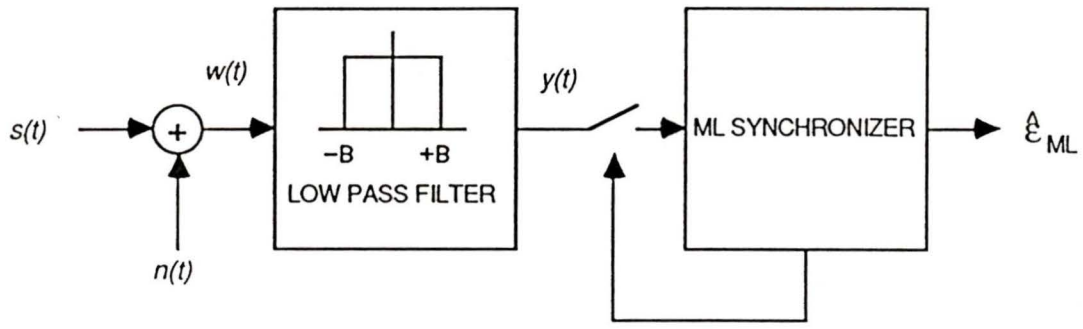


Figure 2.1: Block diagram of a digital maximum likelihood synchronizer.

large enough to ensure $x(t)$ is passed relatively undistorted [29]. The filter output $y(t)$ is

$$y(t) = \sum_{j=0}^{K} a_j x(t - (j-1)T - \epsilon) + \eta(t) \tag{2.9}$$

where $\eta(t)$ is the filtered $n(t)$. The synchronizer samples $y(t)$ at a rate m samples per bit, and after mK samples are taken over a period of KT seconds, the synchronizer generates $\hat{\epsilon}_{ML}$.

The filter bandwidth B is large enough to ensure the noise samples will be uncorrelated, and since they are Gaussian, also independent. This requires a theoretical minimum bandwidth [38]

$$B = \frac{m}{2T} \tag{2.10}$$

and hence the noise variance of the samples is

$$\sigma_\eta^2 = N_o B = \frac{mN_o}{2T} \tag{2.11}$$

The sampled $y(t)$ may be represented by the vector $\mathbf{Y}$, which contains mK elements. Let $\mathbf{A}$ be the vector of random variables representing the transmitted binary data

$$\mathbf{A} = [A_0, \ldots, A_j, \ldots, A_K]^t \tag{2.12}$$

The random variable A_j may assume a particular value $a_j \in \{-1, +1\}$. The vector $\mathbf{Y}$ may be partitioned into

$$\mathbf{Y} = [\mathbf{Y}_0, \ldots, \mathbf{Y}_j, \ldots, \mathbf{Y}_K]^t \tag{2.13}$$

where $\mathbf{Y}_j$ is

$$\mathbf{Y}_j = \begin{bmatrix} Y_j^1 \\ Y_j^2 \\ . \\ . \\ Y_j^{m_j} \end{bmatrix} = A_j \begin{bmatrix} X_j^1(\epsilon) \\ X_j^2(\epsilon) \\ . \\ . \\ X_j^{m_j}(\epsilon) \end{bmatrix} + \begin{bmatrix} N_j^1 \\ N_j^2 \\ . \\ . \\ N_j^{m_j} \end{bmatrix} \tag{2.14}$$

or in vector notation

$$\mathbf{Y}_j = A_j \mathbf{X}_j(\epsilon) + \mathbf{N}_j \tag{2.15}$$

The elements of $\mathbf{X}_j$ and $\mathbf{N}_j$ are the sampled values of the waveforms $x(t - (j-1)T - \epsilon)$ and $\eta(t)$ respectively. The number of elements in $\mathbf{Y}_j$ is $m_j = m$ for $j = 1, 2, \ldots K-1$. For the two outer symbols, m_0 is the number of samples taken in the last ϵ seconds of the $j = 0$ symbol, and m_K is the number of samples taken in the first $(T - \epsilon)$ seconds of the K^{th} symbol.

2.5 Optimum Maximum Likelihood Synchronization

Before developing the HL synchronizer it is useful for comparison to review briefly the derivation of the optimum ML synchronizer [16], and the determination of the CR bound on the timing error variance for this synchronizer.

2.5.1 ML-NDA and ML-DA Synchronization

The conditional density to be maximized for ML-NDA synchronization is (2.1). From the independence assumptions about the noise samples and the

data, the $\mathbf{Y}_j$ vectors in (2.13) are statistically independent, and therefore (2.1) may be written as

$$f_{\mathrm{Y}|\epsilon}(\mathbf{y}|\epsilon) = \prod_{j=0}^{K} f_{\mathrm{Y}_j|\epsilon}(\mathbf{y}_j|\epsilon) \tag{2.16}$$

The elements Y_j^i in (2.14) are Gaussian distributed with mean $a_j X_i^j(\epsilon)$ and variance σ_η^2 given by (2.11). Since the noise samples are statistically independent the conditional density $f_{\mathrm{Y}_j|\epsilon,A_j}(\mathbf{y}_j|\epsilon, a_j)$ is

$$f_{\mathrm{Y}_j|\epsilon,A_j}(\mathbf{y}_j|\epsilon, a_j) = \frac{1}{(2\pi\sigma_\eta)^{m_j/2}} \exp\left\{\frac{[\mathbf{y}_j - a_j\mathbf{x}_j(\epsilon)]^t[\mathbf{y}_j - a_j\mathbf{x}_j(\epsilon)]}{2\sigma_\eta^2}\right\} \tag{2.17}$$

Expanding the exponent in (2.17) and averaging over the probability of a_j yields

$$\begin{aligned} f_{\mathrm{Y}_j|\epsilon}(\mathbf{y}_j|\epsilon) &= \frac{1}{(2\pi\sigma_\eta)^{m_j/2}} \exp\left\{\frac{\|\mathbf{y}_j\| + \|\mathbf{x}_j(\epsilon)\|}{2\sigma_\eta^2}\right\} \cdot \\ &\quad \frac{1}{2}\left\{\exp\left(\frac{\mathbf{y}_j^t\mathbf{x}_j(\epsilon)}{\sigma_\eta^2}\right) + \exp\left(\frac{-\mathbf{y}_j^t\mathbf{x}_j(\epsilon)}{\sigma_\eta^2}\right)\right\} \end{aligned} \tag{2.18}$$

where

$$\|\mathbf{y}_j\| = \mathbf{y}_j^t\mathbf{y}_j \tag{2.19}$$

$$\|\mathbf{x}_j\| = \mathbf{x}_j^t\mathbf{x}_j \tag{2.20}$$

Since

$$\frac{e^z + e^{-z}}{2} = \cosh(z) \tag{2.21}$$

equation (2.18) may be substituted into (2.16) to get

$$f_{Y|\epsilon}(\mathbf{y}|\epsilon) = \prod_{j=0}^{K} \frac{1}{(2\pi\sigma_\eta)^{m_j/2}} \exp\left\{\frac{\|\mathbf{y}_j\| + \|\mathbf{x}_j(\epsilon)\|}{2\sigma_\eta^2}\right\} \cdot \cosh\left(\frac{\mathbf{y}_j^t \mathbf{x}_j(\epsilon)}{\sigma_\eta^2}\right) \tag{2.22}$$

The value of $\|\mathbf{y}_j\|$ is independent of ϵ and can therefore be factored out of the exponent into a multiplicative constant. The value of $\|\mathbf{x}_j(\epsilon)\|$ when summed in the exponent over the entire K symbols is the pulse energy received in the observation interval. Hence $\|\mathbf{x}_j(\epsilon)\|$ is independent of ϵ and can also be factored into a constant. Combining all the constants into a single constant C_1, the conditional density function to be maximized is

$$f_{Y|\epsilon}(\mathbf{y}|\epsilon) = \prod_{j=0}^{K} C_1 \cosh\left(\frac{\mathbf{y}_j^t \mathbf{x}_j(\epsilon)}{\sigma_\eta^2}\right) \tag{2.23}$$

Equivalently the logarithm of (2.23) may be maximized [16,9]

$$\ln f_{Y|\epsilon}(\mathbf{y}|\epsilon) = \sum_{j=0}^{K} \ln\cosh\left(\frac{\mathbf{y}_j^t \mathbf{x}_j(\epsilon)}{\sigma_\eta^2}\right) \tag{2.24}$$

where the constant $\ln(C_1)$ is unimportant in the maximization of (2.24) and therefore is neglected. In the limit as $m_j \to \infty$ the inner product in (2.24) may be written as an integral

$$\lim_{m_j \to \infty} \mathbf{y}_j^t \mathbf{x}(\epsilon) = \frac{m_j}{T_j} \int_{\tau_j(\epsilon)} y(t)x(t-(j-1)T-\epsilon)dt \tag{2.25}$$

where $\tau_j(\epsilon)$ is the interval of integration

$$\tau_j(\epsilon) = \begin{cases} 0 & \leq t < \epsilon & j = 0 \\ (j-1)T + \epsilon & \leq t < jT + \epsilon & j = 1, 2, \ldots (K-1) \\ (K-1)T + \epsilon & \leq t \leq KT & j = K \end{cases} \tag{2.26}$$

and T_j is the duration of the integration

$$T_j = \begin{cases} \epsilon & j = 0 \\ T & j = 1, 2, \ldots (K-1) \\ T - \epsilon & j = K \end{cases} \tag{2.27}$$

Substituting (2.11) and (2.25) into (2.24), the conditional density function to be maximized for the continuous time case is [9]

$$\ln f_{\mathrm{Y}|\epsilon}(\mathbf{y}|\epsilon) = \sum_{j=0}^{K} \ln \cosh \left(\frac{2}{N_o} \int_{\tau_j(\epsilon)} y(t) x(t - (j-1)T - \epsilon) dt \right) \tag{2.28}$$

A digital synchronization algorithm may be interpreted from (2.24) as follows: assume $\hat{\epsilon}_{ML}$ can be quantized to one of q possible values. The value of q would usually be the number of samples per bit m. For each of the m possible delay estimates $\hat{\epsilon}_l$ where $l \in \{1, 2, \ldots m\}$, correlate the samples of the received waveform $y(t)$ with stored samples of the signalling waveform of $x(t - (j-1)T - \hat{\epsilon}_l)$ over each subinterval $\tau_j(\hat{\epsilon}_l)$, calculate the log hyperbolic cosine of the output, and accumulate these results over the $(K+1)$ subintervals. The value of $\hat{\epsilon}_l$ that yields the largest result is selected as $\hat{\epsilon}_{ML}$.

The ML estimate found using (2.28) requires no information about the transmitted data $\mathbf{A}$, and hence is a ML-NDA synchronizer. Often the data

sequence may be known, or can be estimated. For example a data preamble may be used for synchronization, or the recovered data can be fed back to the synchronizer. If $\mathbf{A}$ is known, averaging of the conditional density in (2.17) is not required. Following the same procedure that leads to (2.24) but omitting the averaging, the function to be maximized for a ML-DA synchronizer is

$$\ln f_{\mathrm{Y}|\epsilon,\mathrm{A}}(\mathbf{y}|\epsilon,\mathbf{a}) = \sum_{j=0}^{K} \ln f_{\mathrm{Y}_j|\epsilon,A_j}(\mathbf{y}_j|\epsilon,a_j) \tag{2.29}$$

$$= \sum_{j=0}^{K} \frac{a_j \mathbf{y}_j^t \mathbf{x}_j(\epsilon)}{\sigma_\eta^2} \tag{2.30}$$

In the limit as $m_j \to \infty$ the integral form of (2.30) is [11]

$$\ln f_{\mathrm{Y}|\epsilon,\mathrm{A}}(\mathbf{y}|\epsilon,\mathbf{a}) = \sum_{j=0}^{K} \frac{2a_j}{N_o} \int_{\tau_j(\epsilon)} y(t)x(t-(j-1)T-\epsilon)dt \tag{2.31}$$

The ML-DA algorithm to implement (2.31) is similar to the algorithm to implement (2.28). The log hyperbolic cosine operation is simply replaced with a multiplication by the known (or estimated) a_j.

In general the ML-DA synchronizer will have better performance than the ML-NDA synchronizer, providing the data sequence can be accurately estimated [17]. Note that at high SNR the ML-DA synchronizer and the ML-NDA synchronizer are asymptotically equivalent since

$$\ln\cosh(x) \approx |x|, \qquad |x| >> 1. \tag{2.32}$$

Synchronizers that directly implement the optimum synchronization algorithms are termed open loop synchronizers. The open loop synchronizers

have generally not been considered practical for implementation. For example if $\hat{\epsilon}$ is quantized to q possible values, the synchronizer must evaluate all q possibilities with either a parallel or a serial search. A parallel search may have excessive hardware requirements, while a serial search may require a long delay before $\hat{\epsilon}_{ML}$ is found. Synchronizer structures to implement both the parallel search and serial search synchronizers are illustrated and described in [9].

Several closed loop synchronization systems have been developed based on satisfying the necessary condition for $\hat{\epsilon}_{ML}$ given in (2.3), using either the ML-NDA result (2.28) or the ML-DA result (2.31) [4,9]. The correlation operations may be replaced by matched filters.

2.5.2 ML-DA Synchronizer Cramér-Rao Bound

The derivation of the timing error variance of the ML-DA (optimum situation) synchronizer is reviewed. Synchronizer CR bound calculations are discussed in [11]. From (2.6), the CR lower bound is

$$E[(\hat{\epsilon}_{ML} - \epsilon)^2] \geq -\left\{ E\left[\left(\frac{\partial^2 \ln f_{\mathrm{Y}|\epsilon,\mathrm{A}}(\mathbf{y}|\epsilon, \mathbf{a})}{\partial \epsilon^2} \right)^2 \right] \right\}^{-1} \tag{2.33}$$

Differentiating (2.31) twice with respect to ϵ, and substituting into (2.33) yields

$$E[(\hat{\epsilon}_{ML} - \epsilon)^2] \geq -\left\{ \sum_{j=0}^{K} \frac{2a_j}{N_o} \int_{\tau_j(\epsilon)} E[y(t)] \left[\frac{\partial^2 x(t-(j-1)T-\epsilon)}{\partial \epsilon^2} \right] dt \right\}^{-1} \tag{2.34}$$

The expectation is with respect to $y(t)$ since ϵ and a_j are regarded as being nonrandom. The expectation within the integral of (2.34) is evaluated using (2.9). The CR bound for the ML synchronizer is therefore

$$E[(\hat{\epsilon}_{ML}-\epsilon)^2] \geq -\left\{\sum_{j=0}^{K}\frac{2}{N_o}\int_{\tau_j(\epsilon)} x(t-(j-1)T-\epsilon)x''(t-(j-1)T-\epsilon)dt\right\}^{-1} \tag{2.35}$$

For example, consider $x(t)$ pulses that are half sinusoids

$$x_{hs}(t) = \begin{cases} \sqrt{2E_bT}\sin(\pi t/T) & \text{for } t \in (0,T) \\ 0 & \text{for } t \notin (0,T) \end{cases} \tag{2.36}$$

Substituting the second partial derivative of (2.36) into (2.35), and after performing the integration the lower bound is

$$E[(\hat{\epsilon}_{ML}-\epsilon)^2] \geq \frac{N_o}{E_b}\frac{1}{2K\pi^2} \tag{2.37}$$

2.6 Optimum Hard Limited Synchronization

We now consider a synchronizer that operates only on HL data. In this section we determine the optimum (in the ML sense) HL-DA and HL-NDA synchronizers, and establish the CR lower bound on the HL-DA timing error variance. These results have not been found in the literature. By comparing these results with the previous ML synchronizer results the degradation associated with hard limiting can be determined.

2.6.1 HL-NDA and HL-DA Synchronization

The HL synchronization system is illustrated in figure 2.2. The output of the filter $y(t)$ is passed through a hard limiter before being sampled by the synchronizer. The hard limited signal $v(t)$ is

$$v(t) = \text{ sgn}\{y(t)\} = \sum_{j=0}^{K} \text{sgn}\{a_j x(t-(j-1)T-\epsilon)+\eta(t)\} \tag{2.38}$$

where

$$\text{sgn(z)} \triangleq \begin{cases} +1 & \text{for } z > 0 \\ 0 & \text{for } z = 0 \\ -1 & \text{for } z < 1 \end{cases} \tag{2.39}$$

The sampled $v(t)$, represented by the vector $\mathbf{V}$, may be partitioned into the following form

$$\mathbf{V} = [\mathbf{V}_0, \ldots, \mathbf{V}_j, \ldots, \mathbf{V}_K]^t \tag{2.40}$$

where $\mathbf{V}_j$ is

$$\mathbf{V}_j = \begin{bmatrix} V_j^1 \\ V_j^2 \\ . \\ . \\ V_j^{m_j} \end{bmatrix} = \text{sgn}\left\{ A_j \begin{bmatrix} X_j^1(\epsilon) \\ X_j^2(\epsilon) \\ . \\ . \\ X_j^{m_j}(\epsilon) \end{bmatrix} + \begin{bmatrix} N_j^1 \\ N_j^2 \\ . \\ . \\ N_j^{m_j} \end{bmatrix} \right\} \tag{2.41}$$

Define the ML estimate of ϵ for a HL synchronizer to be $\hat{\epsilon}_{HL}$. From the independence assumptions about A_j and N_j^i, the $\mathbf{V}_j$ are independent, and therefore the conditional density function to be maximized is

$$f_{\mathrm{V}|\epsilon}(\mathbf{v}|\epsilon) = \prod_{j=0}^{K} f_{\mathrm{V}_j|\epsilon}(\mathbf{v}_j|\epsilon) \tag{2.42}$$

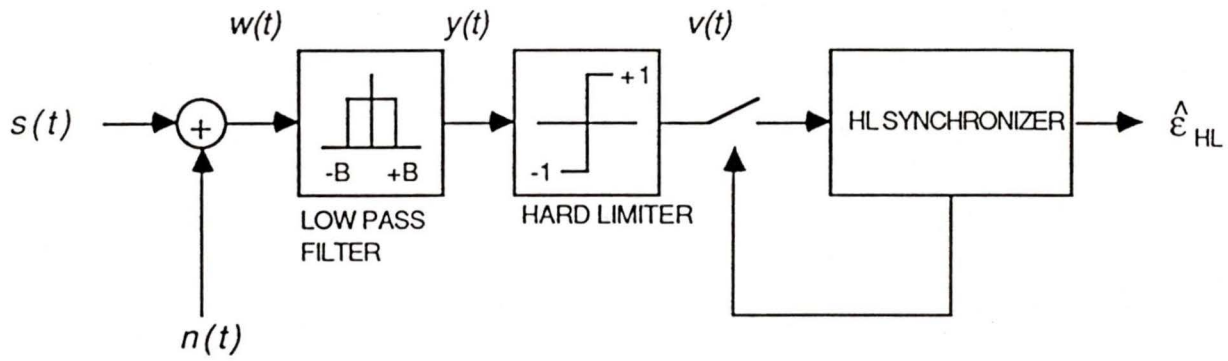


Figure 2.2: Block diagram of a digital hard limited (HL) synchronizer.

where $f_{\mathrm{V}_j|\epsilon}(\mathbf{v}_j|\epsilon)$ is found by averaging the conditional density $f_{\mathrm{V}_j|\epsilon,A_j}(\mathbf{v}_j|\epsilon,a_j)$ over the density function of a_j

$$
\begin{aligned}
f_{\mathrm{V}_j|\epsilon}(\mathbf{v}_j|\epsilon) &= E[f_{\mathrm{V}_j|\epsilon,A_j}(\mathbf{v}_j|\epsilon,a_j)] && (2.43)\\
&= E[\prod_{i=1}^{m_j} f_{\mathrm{V}_j^i|\epsilon,A_j}(v_j^i|\epsilon,a_j)] && (2.44)\\
&= \frac{1}{2}\left\{\prod_{i=1}^{m_j} f_{\mathrm{V}_j^i|\epsilon,A_j}(v_j^i|\epsilon,a_j=1)+\prod_{i=1}^{m_j} f_{\mathrm{V}_j^i|\epsilon,A_j}(v_j^i|\epsilon,a_j=-1)\right\} && (2.45)
\end{aligned}
$$

The conditional density function $f_{\mathrm{V}_j^i|\epsilon,A_j}(v_j^i|\epsilon,a_j)$ is related to $X_j^i(\epsilon)$ and σ_η^2 as follows

$$
f_{\mathrm{V}_j^i|\epsilon,A_j}(v_j^i|\epsilon,a_j)=\left\{1-P_j^i\right\}\delta(v_j^i-a_j)+P_j^i\delta(v_j^i+a_j) \qquad (2.46)
$$

where

$$
\begin{aligned}
P_j^i &= Pr[N_j^i > x_j^i(\epsilon)]\\
&= Q\left(\frac{x_j^i(\epsilon)}{\sigma_\eta}\right)\\
&= Q\left(\sqrt{\frac{2T}{mN_o}}x_j^i(\epsilon)\right) && (2.47)
\end{aligned}
$$

and

$$
Q(z)=\frac{1}{\sqrt{2\pi}}\int_z^\infty e^{-\zeta^2/2}d\zeta \qquad (2.48)
$$

Combining (2.42), (2.45), and (2.46), the conditional density function to be

maximized is

$$f_{\mathrm{V}|\epsilon}(\mathbf{v}|\epsilon) = \prod_{j=0}^{K} \frac{1}{2} \left\{ \prod_{i=1}^{m_j} \left\{ \left(1 - P_j^i\right) \delta(v_j^i - 1) + P_j^i \delta(v_j^i + 1) \right\} + \prod_{i=1}^{m_j} \left\{ \left(1 - P_j^i\right) \delta(v_j^i + 1) + P_j^i \delta(v_j^i - 1) \right\} \right\} \tag{2.49}$$

or equivalently the logarithm of (2.49) can be maximized

$$\ln f_{\mathrm{V}|\epsilon}(\mathbf{v}|\epsilon) = \sum_{j=0}^{K} \ln \left\{ \prod_{i=1}^{m_j} \left\{ \left(1 - P_j^i\right) \delta(v_j^i - 1) + P_j^i \delta(v_j^i + 1) \right\} + \prod_{i=1}^{m_j} \left\{ \left(1 - P_j^i\right) \delta(v_j^i + 1) + P_j^i \delta(v_j^i - 1) \right\} \right\} \tag{2.50}$$

where the constant $\ln(1/2)$ is ignored since it is not important in the maximization.

Assuming σ_η is known, (2.50) can be used as a basis for a HL-NDA synchronization algorithm to determine $\hat{\epsilon}_{HL}$. This algorithm is optimum in the ML sense. For each of the m possible values of $\hat{\epsilon}_{HL}$ the probability of receiving $\mathbf{V}_j$ is determined and the logarithm of these probabilities are accumulated over the $(K+1)$ subintervals. The value of $\hat{\epsilon}_l$ that yields the largest result is selected as $\hat{\epsilon}_{HL}$.

Following the same procedure used to derive (2.50), the function to be

maximized for a HL-DA synchronizer is

$$\ln f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon,\mathbf{a}) = \sum_{j=0}^{K} \ln f_{\mathrm{V}_j|\epsilon,A_j}(\mathbf{v}_j|\epsilon,a_j) \tag{2.51}$$

which from (2.46) is

$$\ln f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon,\mathbf{a}) =$$

$$\sum_{j=0}^{K} \sum_{i=1}^{m_j} \ln\left\{\left(1-P_j^i\right)\delta(v_j^i - a_j) + P_j^i \delta(v_j^i + a_j)\right\} \tag{2.52}$$

The HL-DA synchronization algorithm based on (2.52) can be implemented as follows: for each of the m possible values of $\hat{\epsilon}_{HL}$, the logarithm of the probability of receiving $\mathbf{V}_j$ (given a_j is known or estimated) is determined, and the results are accumulated over the $K+1$ subintervals. The value of $\hat{\epsilon}_l$ that generates the largest result is selected as $\hat{\epsilon}_{HL}$.

Both the HL-NDA and HL-DA synchronizer algorithms generate the ML estimate $\hat{\epsilon}_{HL}$, but neither may be practical to implement. The equations require a large number of numerical operations, evaluation of $Q(z)$, and a priori knowledge of σ_η, which generally isn't available.

We now derive synchronization algorithms (HL-DA and HL-NDA) that are practical to implement. The analysis to follow assumes a large m to simplify the optimum HL-DA result (2.52) with linear approximations.

From (2.47), the argument of the Q function approaches zero for large m, and therefore may be approximated by the first two terms of a power series

$$Q(\zeta) \approx \frac{1}{2} - \frac{\zeta}{\sqrt{2\pi}}, \qquad \zeta << 1. \tag{2.53}$$

With this approximation, (2.52) becomes

$$\ln f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon,\mathbf{a}) \approx$$

$$\sum_{j=0}^{K}\sum_{i=1}^{m_j}\ln\left\{\left(\frac{1}{2}+C_2x_j^i(\epsilon)\right)\delta(v_j^i-a_j)+\left(\frac{1}{2}-C_2x_j^i(\epsilon)\right)\delta(v_j^i+a_j)\right\}$$

$$= \sum_{j=0}^{K}\sum_{i=1}^{m_j}\ln\left\{\frac{1}{2}+C_2a_jv_j^ix_j^i(\epsilon)\right\} \tag{2.54}$$

where the constant C_2 is

$$C_2=\sqrt{\frac{T}{mN_o\pi}} \tag{2.55}$$

For large m

$$C_2x_j^i(\epsilon) << \frac{1}{2} \qquad m >> 1. \tag{2.56}$$

Using

$$\ln\left(\frac{1}{2}+\zeta\right) \approx \ln\left(\frac{1}{2}\right)+2\zeta, \qquad \zeta << 1/2. \tag{2.57}$$

and (2.56) in (2.54), the logarithm of the conditional density function to be maximized for a HL-DA synchronizer becomes

$$\ln f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon,\mathbf{a}) \approx 2C_2\sum_{j=0}^{K}a_j\sum_{i=1}^{m_j}v_j^ix_j^i(\epsilon)$$

$$= 2C_2\sum_{j=0}^{K}a_j\mathbf{v}_j^t\mathbf{x}_j(\epsilon) \tag{2.58}$$

where the constant associated with $\ln(1/2)$ is omitted. In the limit as $m \to \infty$, (2.58) may be expressed in integral form

$$\ln f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon,\mathbf{a}) = C_3 \sum_{j=0}^{K} a_j \int_{\tau_j(\epsilon)} v(t)x(t-(j-1)T-\epsilon)dt \tag{2.59}$$

where C_3 is a scaling constant.

Comparing (2.59) to (2.31), it is readily apparent that the HL-DA synchronization algorithm (based on the linear approximations) and the ML-DA synchronization algorithm are identical. A parallel search open loop synchronizer structure to implement (2.58) is shown in figure 2.3.

A HL-NDA synchronizer may be derived from (2.58) by exponentiating both sides of the equation, averaging over the density of a_j, and taking the logarithm of the result. The result is an expression similar to (2.24)

$$\ln f_{\mathrm{V}|\epsilon}(\mathbf{v}|\epsilon) \approx \sum_{j=0}^{K} \ln\cosh\left\{2C_2\mathbf{v}_j^t\mathbf{x}_j(\epsilon)\right\} \tag{2.60}$$

To simplify the implementation, a useful approximation is $\ln\cosh(z) \approx |z|, |z| >> 1$ [9,16]. This approximation is appropriate for a large value of E_b/N_o and/or a large m. Using this approximation (2.60) becomes

$$\ln f_{\mathrm{V}|\epsilon}(\mathbf{v}|\epsilon) = 2C_2 \sum_{j=0}^{K} |\mathbf{v}_j^t\mathbf{x}_j(\epsilon)| \tag{2.61}$$

and in integral form ($m_j \to \infty$)

$$\ln f_{\mathrm{V}|\epsilon}(\mathbf{v}|\epsilon) = C_3 \sum_{j=0}^{K} \left| \int_{\tau_j(\epsilon)} v(t)x(t-jT-\epsilon)dt \right| \tag{2.62}$$

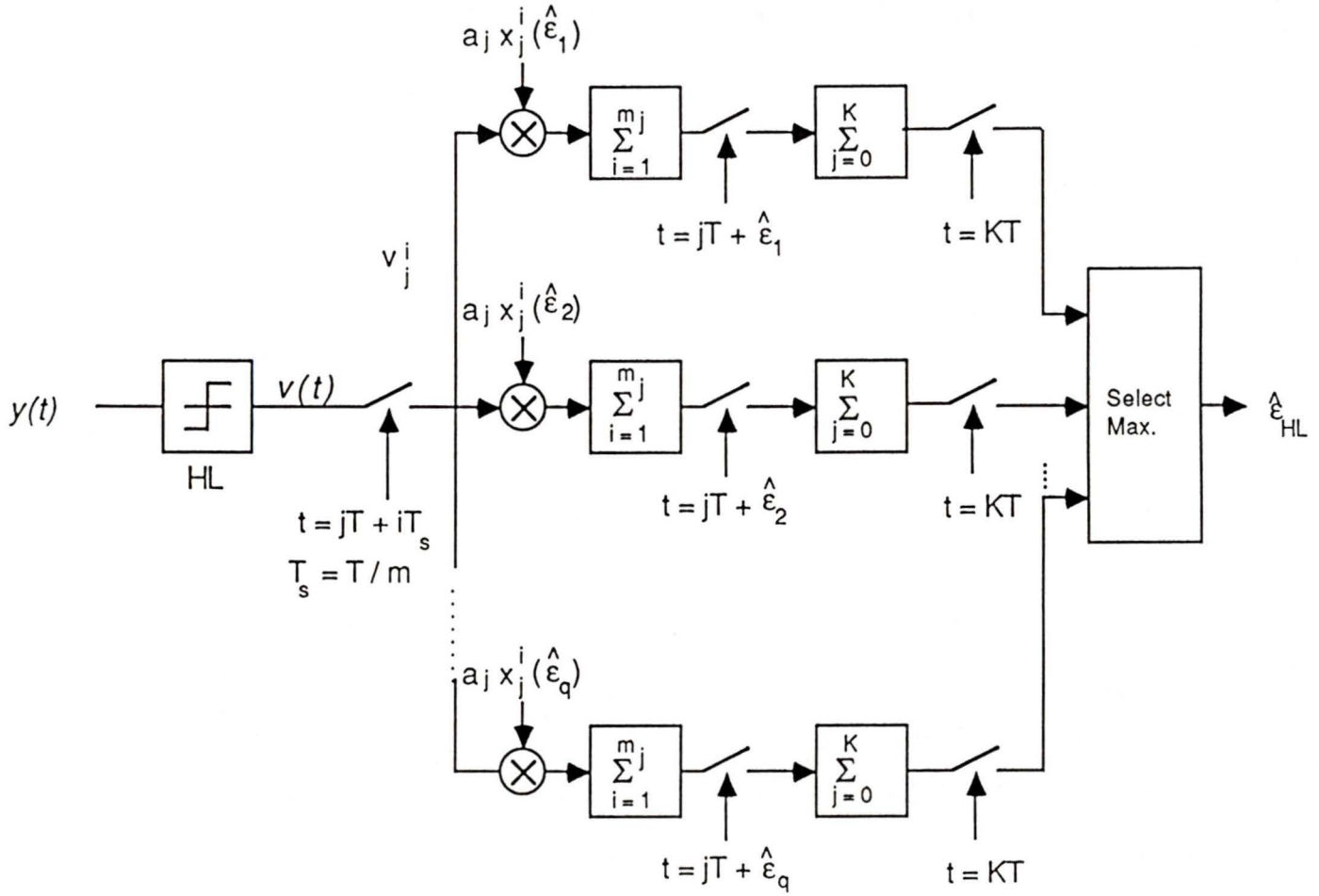


Figure 2.3: Block diagram of a HL-DA open loop synchronizer.

The open loop HL-DA synchronizer in figure 2.3 may be modified to implement the HL-NDA algorithm (2.61) by removing the a_j multiplication from each correlator, and including an absolute value function after the correlators.

For a digital circuit implementation, the major mathematical operation in both (2.58) and (2.61) is the inner product calculation $\mathbf{v}_j^t \mathbf{x}_j(\epsilon)$. Remembering that the elements $v_j^i \in \{-1, +1\}$, the hardware implementation is a simple summation of the m_j elements in $\mathbf{x}_j(\epsilon)$ with the polarity of each element determined by v_j^i.

The necessary condition for the ML estimate defined by (2.3) can be applied to either the HL-DA or HL-NDA equations to derive a closed loop realization of the HL synchronizer. In the case of the HL-DA synchronizer, applying (2.3) to (2.58) yields

$$\frac{\partial \ln f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon, \mathbf{a})}{\partial \epsilon} = 2C_2 \sum_{j=0}^{K} a_j \mathbf{v}_j^t \left[\frac{\partial \mathbf{x}_j(\epsilon)}{\partial \epsilon}\right] \tag{2.63}$$

$$= 0 \tag{2.64}$$

A possible closed loop implementation of (2.64) is shown in figure (2.4). The operation of the synchronizer may be described as follows: the hard limited waveform $v(t)$ is sampled and correlated with $\mathbf{x}'(\hat{\epsilon})$, a stored replica of the derivative of the sampled signalling waveform $x(t)$. The correlator output is multiplied by the estimated data symbol $\hat{a}_j$, and the resulting sequence is filtered by a loop filter $F(z)$. The timing error estimate $\hat{\epsilon}$ is chosen to force the output of the loop filter to zero. Note that the data decisions $\hat{a}_j$ are determined with a weighted partial decision binary detector

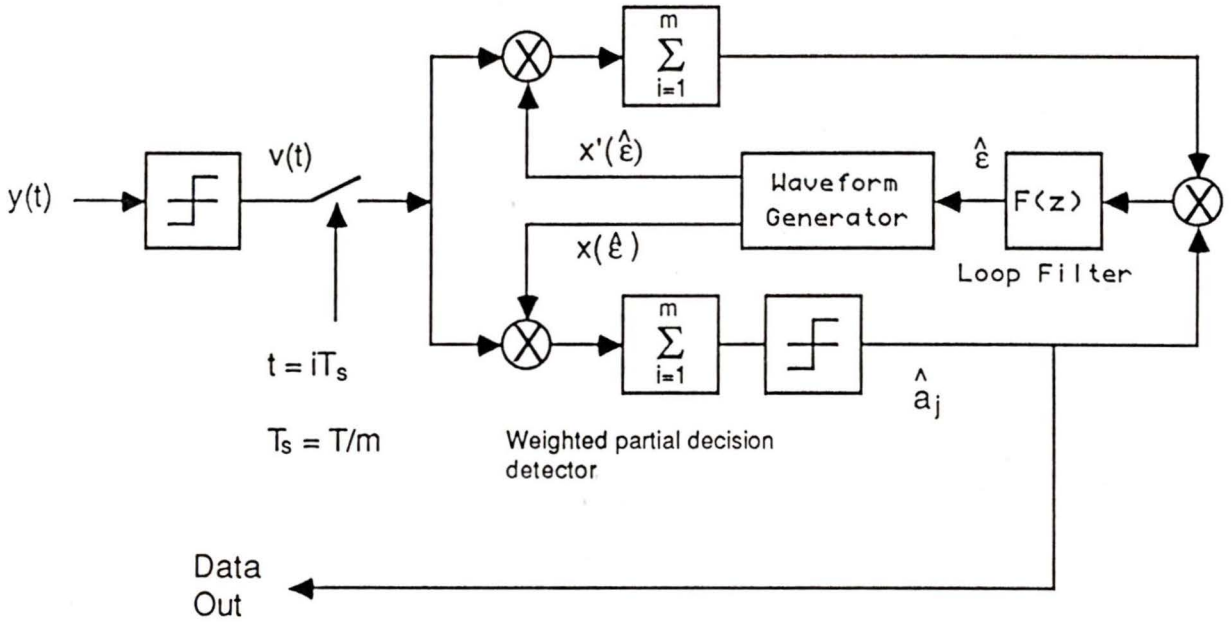


Figure 2.4: Closed loop realization of a HL-DA synchronizer.

[29]. The synchronizer structure is very similar to other non-hard-limited systems [9,11].

2.6.2 HL-DA Synchronizer Cramér-Rao Bound

The lower bound on the timing error variance for the HL-DA synchronizer is to be determined. The linear assumptions made in the derivation of (2.58) are assumed to be correct, i.e. a large sample size m is valid. It is required to find

$$E[(\hat{\epsilon}_{HL} - \epsilon)^2] \geq -\left\{ E\left[\left(\frac{\partial^2 \ln f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon, \mathbf{a})}{\partial \epsilon^2} \right)^2 \right] \right\}^{-1} \tag{2.65}$$

Differentiating (2.58) twice with respect to ϵ and substituting into (2.65) yields

$$E[(\hat{\epsilon}_{HL} - \epsilon)^2] \geq -\left\{ 2C_2 \sum_{j=0}^{K} a_j E[\mathbf{v}_j^t] \left[\frac{\partial^2 \mathbf{x}_j(\epsilon)}{\partial \epsilon^2} \right] \right\}^{-1} \tag{2.66}$$

Since the expectation $E[\cdot]$ is evaluated over the conditional density $f_{\mathrm{V}|\epsilon,\mathrm{A}}(\mathbf{v}|\epsilon, \mathbf{a})$, both ϵ and a_j are regarded as being nonrandom, and therefore

$$E[\mathbf{v}_j] = E[\mathbf{v}_j|\epsilon, a_j] \tag{2.67}$$

The expectation of each element in $\mathbf{v}_j$ can be determined using the linear Q function approximation (2.53)

$$E[v_j^i|\epsilon, a_j] = \sum_{v_j^i} v_j^i Pr[V_j^i = v_j^i|\epsilon, a_j], \quad v_j^i \in \{-1, 1\} \tag{2.68}$$

$$\approx \sum_{v_j^i} v_j^i \left\{ \frac{1}{2} + C_2 a_j v_j^i x_j^i(\epsilon) \right\} \tag{2.69}$$

$$= 2C_2 a_j x_j^i(\epsilon) \tag{2.70}$$

Substituting (2.70) into (2.66), the CR bound for the HL-DA synchronizer is

$$E[(\hat{\epsilon}_{HL} - \epsilon)^2] \geq - \left\{ 4C_2^2 \sum_{j=0}^{K} \mathbf{x}_j^t \left[\frac{\partial^2 \mathbf{x}_j(\epsilon)}{\partial \epsilon^2} \right] \right\}^{-1} \tag{2.71}$$

The integral form of (2.71) as $m_j \rightarrow \infty$ is

$$E[(\hat{\epsilon}_{HL} - \epsilon)^2] \geq$$
$$- \left\{ \sum_{j=0}^{K} \frac{4}{\pi N_o} \int_{\tau_j(\epsilon)} x(t - (j-1)T - \epsilon) x''(t - (j-1)T - \epsilon) dt \right\}^{-1} \tag{2.72}$$

where (2.55) has been substituted for C_2.

Comparing the CR bound for the ML-DA synchronizer (2.35), to the CR bound for the HL-DA synchronizer (2.72), it is easily seen that

$$E[(\hat{\epsilon}_{HL} - \epsilon)^2]_{min} = \left\{ \frac{\pi}{2} \right\} E[(\hat{\epsilon}_{ML} - \epsilon)^2]_{min} \tag{2.73}$$

indicating a penalty factor in terms of timing error variance of $\pi/2$ for the HL-DA synchronizer. It is of interest to note that this is the same factor that has been determined to be the penalty (in terms of required E_b/N_o for equivalent probability of bit error performance) of the weighted

partial decision binary detector with decision weights equal to the sampled waveform $x(t)$ [29].

For a half sinusoid $x(t)$, the CR bound for the HL-DA synchronizer is

$$E[(\hat{\epsilon}_{HL} - \epsilon)^2] \geq \frac{1}{4K\pi} \frac{N_o}{E_b} \tag{2.74}$$

For HL-DA and ML-DA synchronizers that satisfy their respective CR bounds with equality, the two synchronizer will have the equivalent timing error variance performance for a fixed K if

$$\left[\frac{E_b}{N_o}\right]_{HL} = \frac{\pi}{2} \left[\frac{E_b}{N_o}\right]_{ML} \tag{2.75}$$

or for a fixed E_b/N_o

$$[K]_{HL} = \frac{\pi}{2} [K]_{ML} \tag{2.76}$$

2.7 Simulation Results

The HL-DA and HL-NDA synchronization equations, defined by (2.58) and (2.61) respectively, will have the equivalent performance of the optimum HL-DA and HL-NDA synchronizers ((2.52) and (2.50) respectively) as m approaches infinity. It is therefore of interest to simulate these two synchronizers over a range of typical E_b/N_o to determine if practical values of m can be expected to yield timing error variances that are close to the CR lower bound. It is also of interest to determine if the $\pi/2$ variance degradation factor associated with the CR bound for the HL-DA synchronizer is a reasonable approximation over the E_b/N_o and m range.

For the simulations three pulse shapes $x(t)$ of equal energy are considered: a raised cosine pulse, defined by

$$x_{rc}(t) = \begin{cases} \sqrt{\frac{2}{3}P}[1 - \cos(2\pi t/T)] & t \in (0,T) \\ 0 & t \notin (0,T) \end{cases} \tag{2.77}$$

a half sine pulse

$$x_{hs}(t) = \begin{cases} \sqrt{2P}\sin(\pi t/T)] & t \in (0,T) \\ 0 & t \notin (0,T) \end{cases} \tag{2.78}$$

and a square pulse

$$x_{sq}(t) = \begin{cases} \sqrt{P} & t \in (0,T) \\ 0 & t \notin (0,T) \end{cases} \tag{2.79}$$

where P represents the signal power in each case. The square pulse is of special interest since it is the standard non return to zero signalling format. However a meaningful CR bound for this symbol shape cannot be determined due to the discontinuities in the pulse waveform.

The timing error variance σ^2_{HL} for both the HL-DA and HL-NDA synchronizer algorithms, defined by (2.58) and (2.61) respectively, is determined using Monte Carlo techniques. The optimum ML-DA and ML-NDA synchronizer algorithms defined by (2.30) and (2.24) are also simulated to determine the ML synchronizer timing error variance σ^2_{ML}. With the exception of the square pulse simulation results, all results are plotted against the HL-DA CR and ML-DA CR bound defined by (2.72) and (2.35) respectively. The simulation results are presented for: $10 \leq m \leq 100$, which is a practical implementation range for m. Sample sizes of less than 10 are not

considered to ensure that the filter with bandwidth B, defined in (2.10), does not significantly distort the waveform $x(t)$. Also since m determines the quantization of the timing error estimate $\hat{\epsilon}_l$, very small values of m will not yield accurate estimates. A description of the simulation method is given in appendix A.

In figure 2.5, the simulation results are presented for raised cosine pulses with a $K = 10$ symbol observation interval and $E_b/N_o = 5, 10,$ and 15 dB. Both the ML and HL synchronizer timing error variances approach their respective CR bound as the sample size m increases. The ML CR bound is slightly tighter than the HL CR bound. As expected the DA performance is better than the NDA performance in both cases. Comparing the HL and ML curves it is apparent that the $\pi/2$ degradation factor is a reasonable approximation for large m (≈ 100). The HL and ML curves tend to converge for small m (≈ 10).

The simulation results for half sinusoid signalling pulses with $K = 10$ symbols and $E_b/N_o = 5, 10$,and 15dB are presented in figure 2.6. The timing error variance is slightly larger than for the raised cosine pulses, which from the CR calculation is expected. Otherwise the $\pi/2$ degradation factor is apparent for large m, and the ML and HL curves converge for small m.

The simulation results for square signalling pulses, $K = 10$ symbols, and $E_b/N_o = 5, 10$, and 15dB are presented in figure 2.7. In the absence of CR bound information, no conclusions can be drawn about the the square pulse results. pulses. General observations indicate that the the ML and HL curves converge for low m and diverge for large m, although the $\pi/2$

Figure 2.5a: Data-Aided Synch.

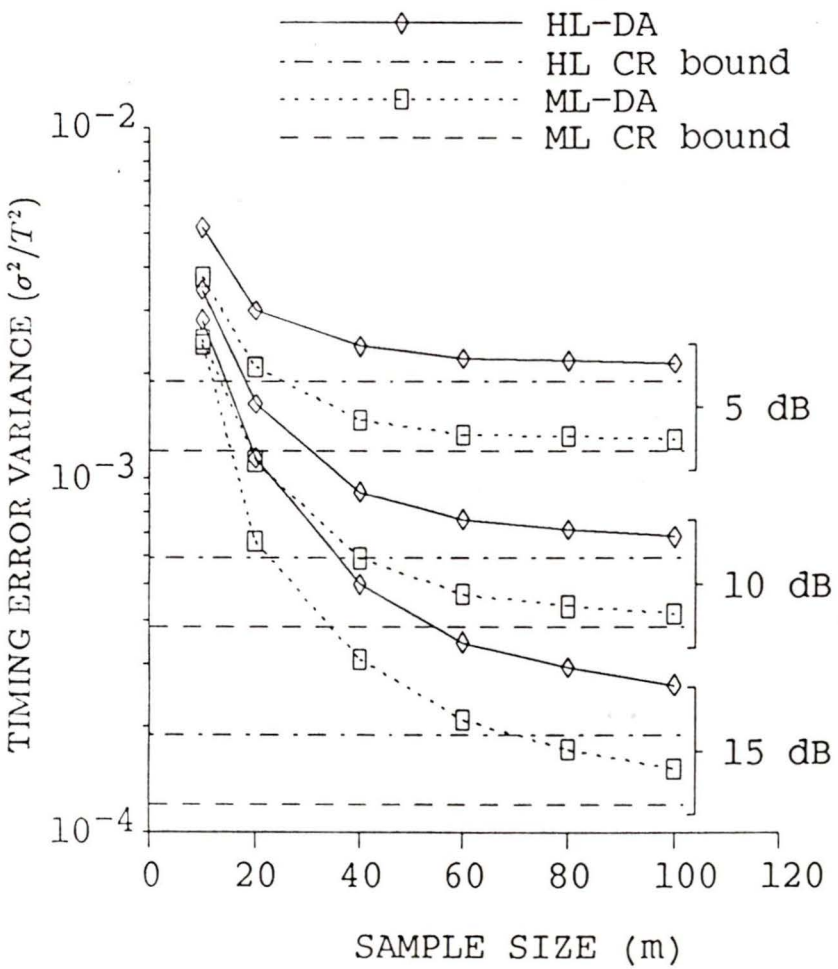


Figure 2.5b: Non-Data-Aided Synch.

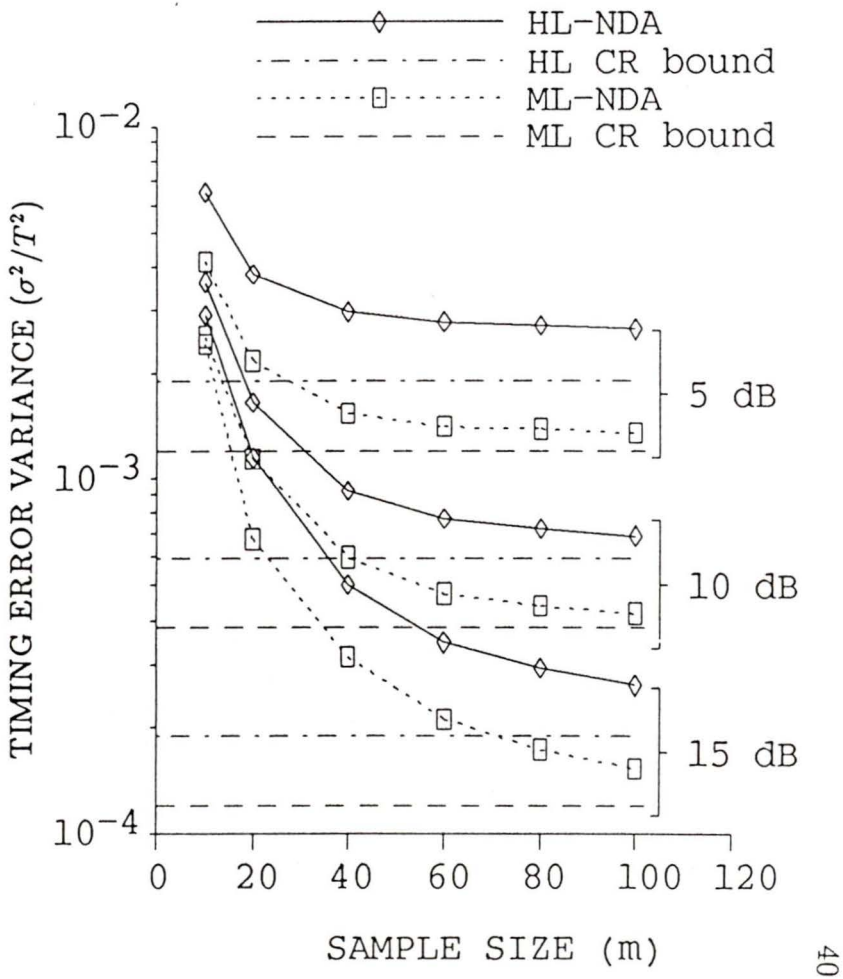


Figure 2.5: Timing error variance versus sample size m for: raised cosine signalling pulses; Eb/No = 5, 10, and 15 dB; K = 10 symbols.

Figure 2.6a: Data-Aided Synch.

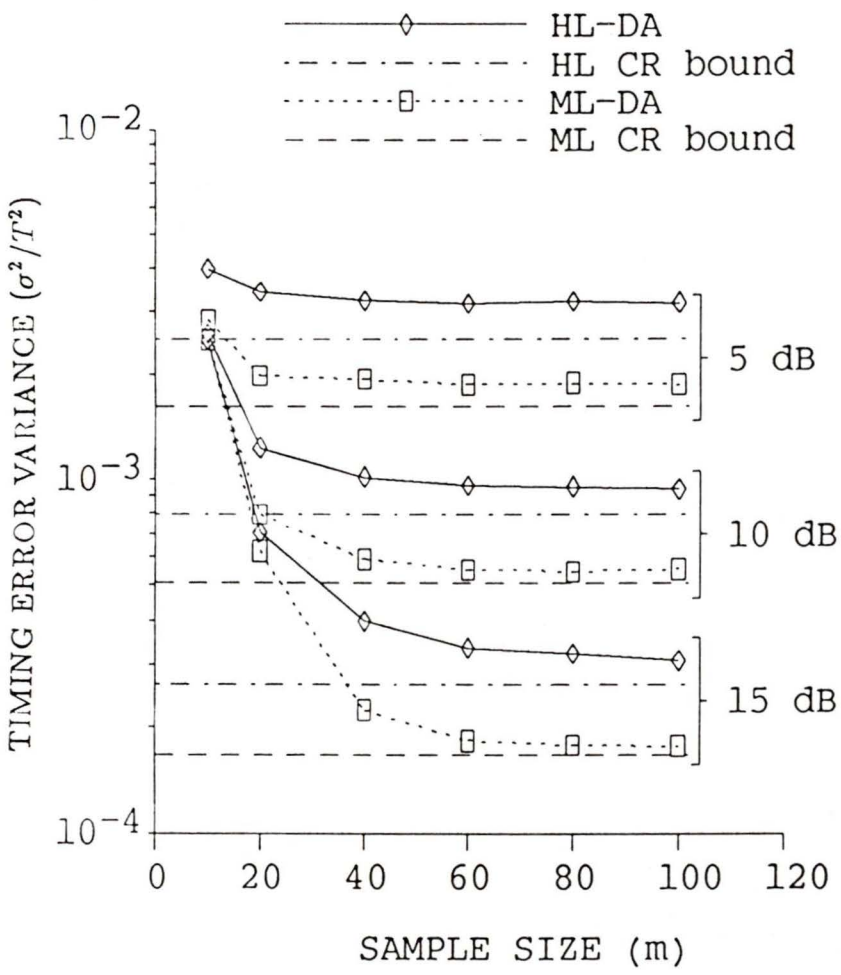


Figure 2.6b: Non-Data-Aided Synch.

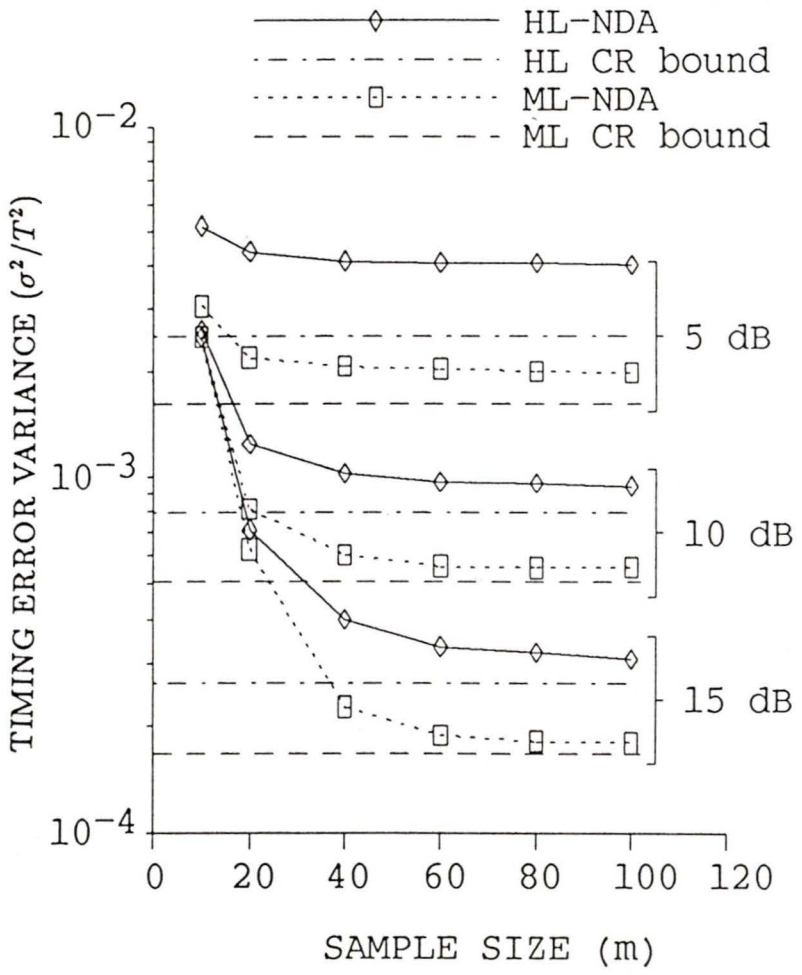


Figure 2.6: Timing error variance versus sample size m for: half sine signalling pulses; Eb/No = 5, 10, and 15 dB; K = 10 symbols.

Figure 2.7a: Data-Aided Synch.

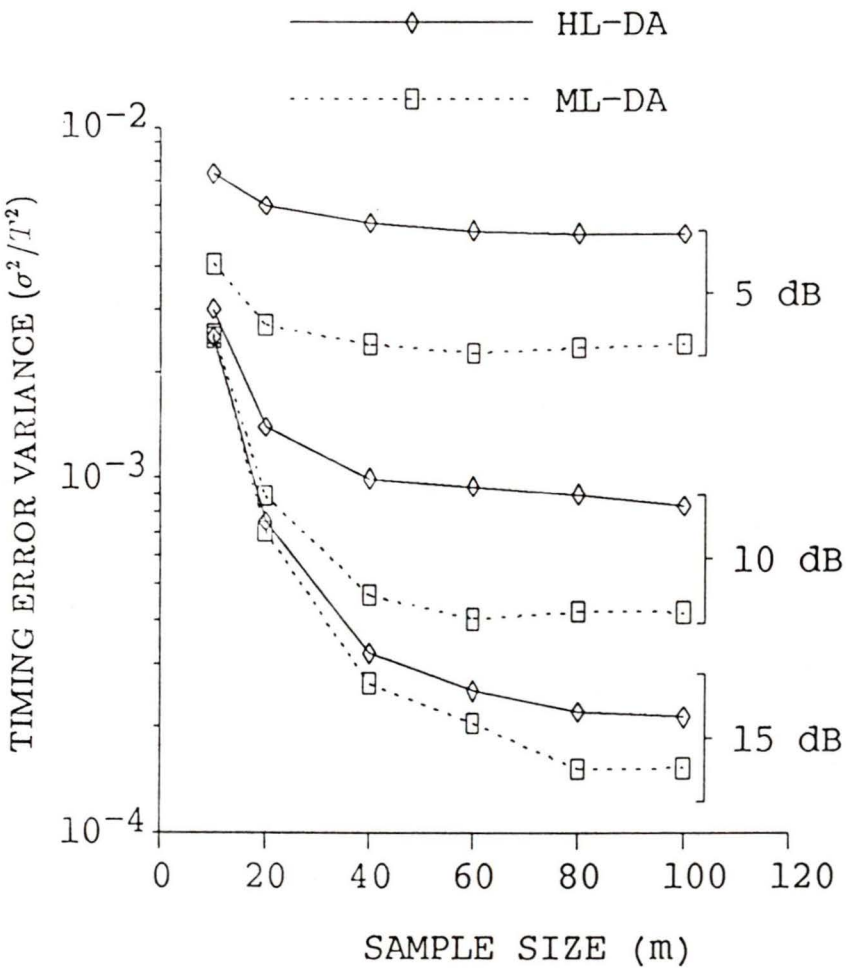


Figure 2.7b: Non-Data-Aided Synch.

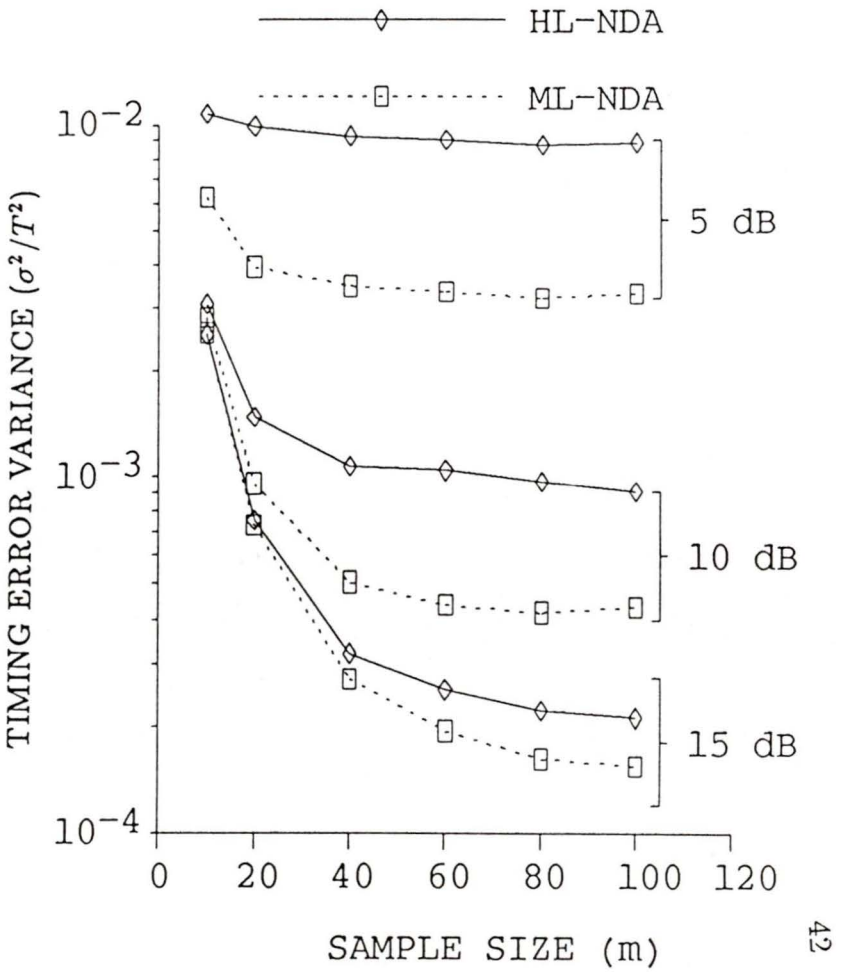


Figure 2.7: Timing error variance versus sample size m for: square signalling pulses; Eb/No = 5, 10, and 15 dB; K = 10 symbols.

factor is not readily apparent.

For $K = 6$ symbols, figures 2.8, 2.9, and 2.10 illustrate the timing error variance results for raised cosine, half sinusoid, and square signalling pulses respectively. For a smaller value of K, the jitter variance increases for both the ML and HL synchronizer, as is expected from the CR bound calculation. Otherwise the performance of the ML and HL synchronizers is relatively unchanged from the $K = 10$ simulations.

Figure 2.8a: Data-Aided Synch.

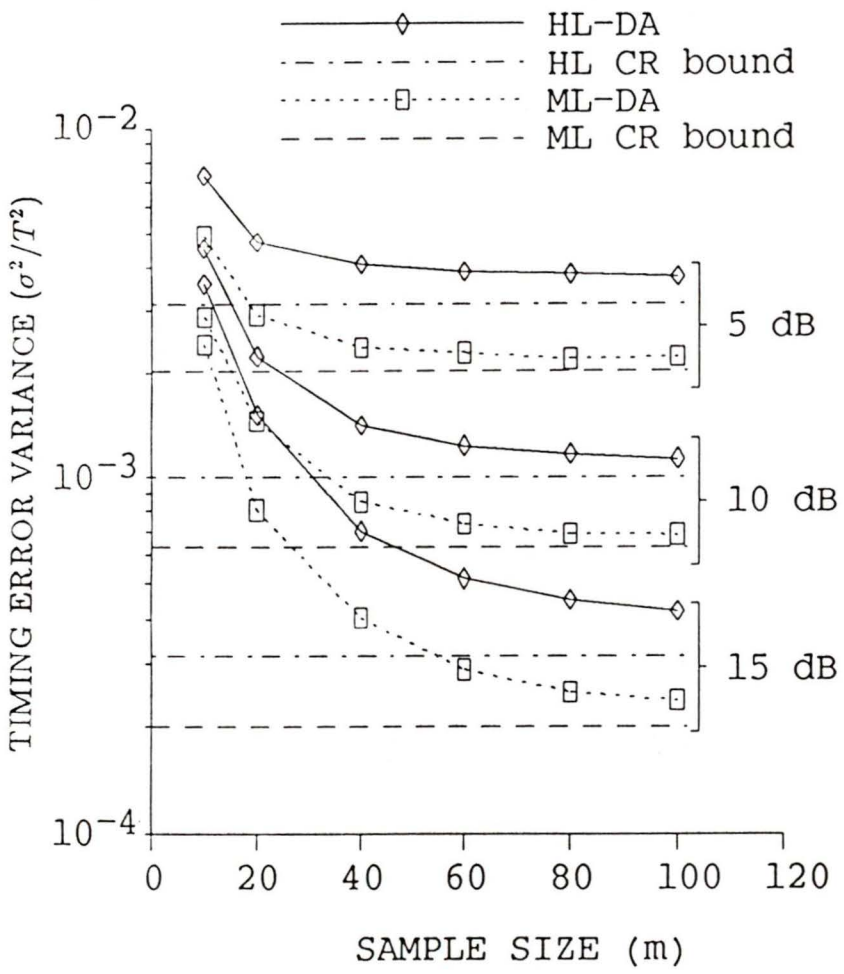


Figure 2.8b: Non-Data-Aided Synch.

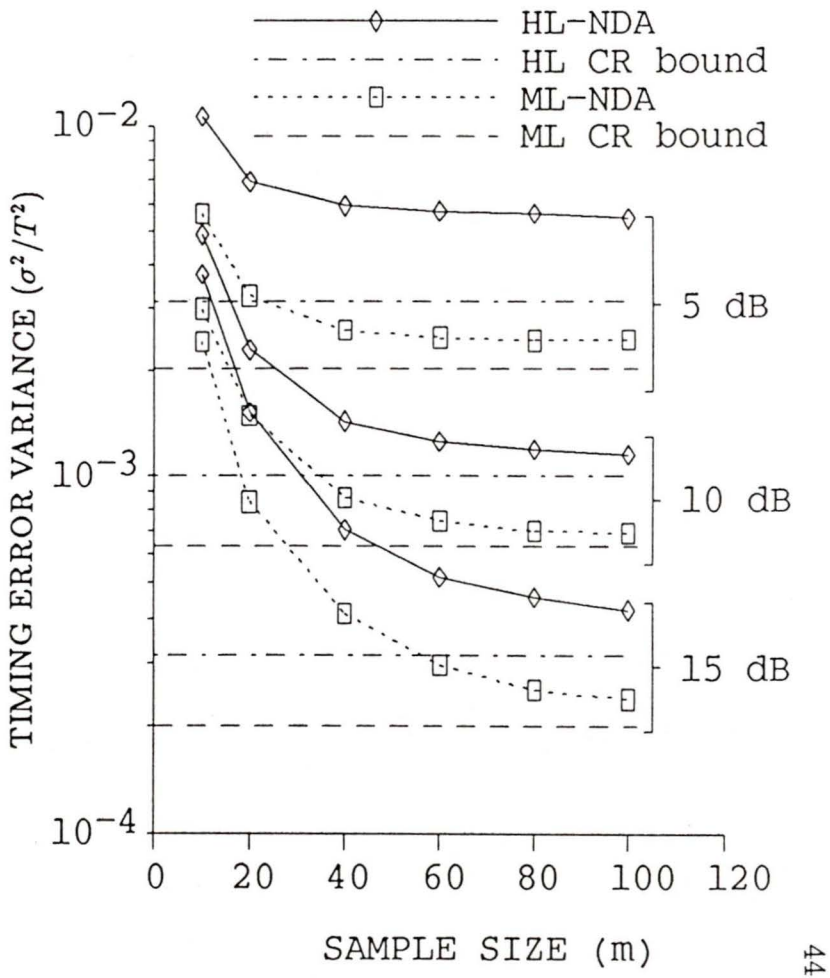


Figure 2.8: Timing error variance versus sample size m for: raised cosine signalling pulses; Eb/No = 5, 10, and 15 dB; K = 6 symbols.

Figure 2.9a: Data-Aided Synch.

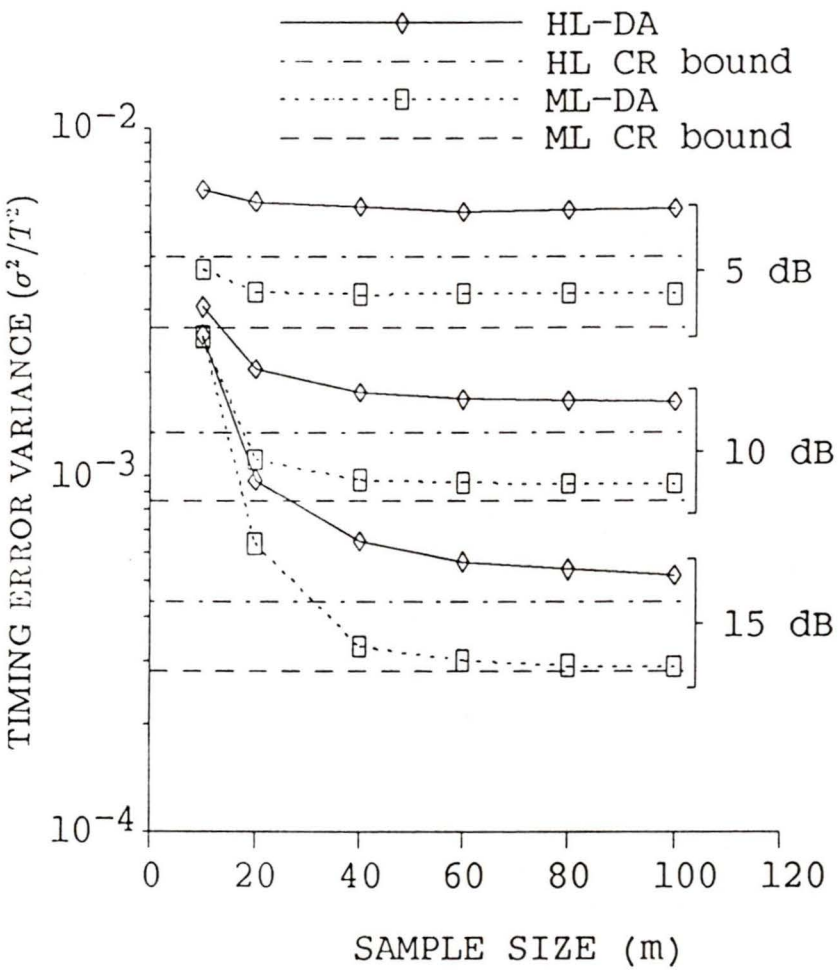


Figure 2.9b: Non-Data-Aided Synch.

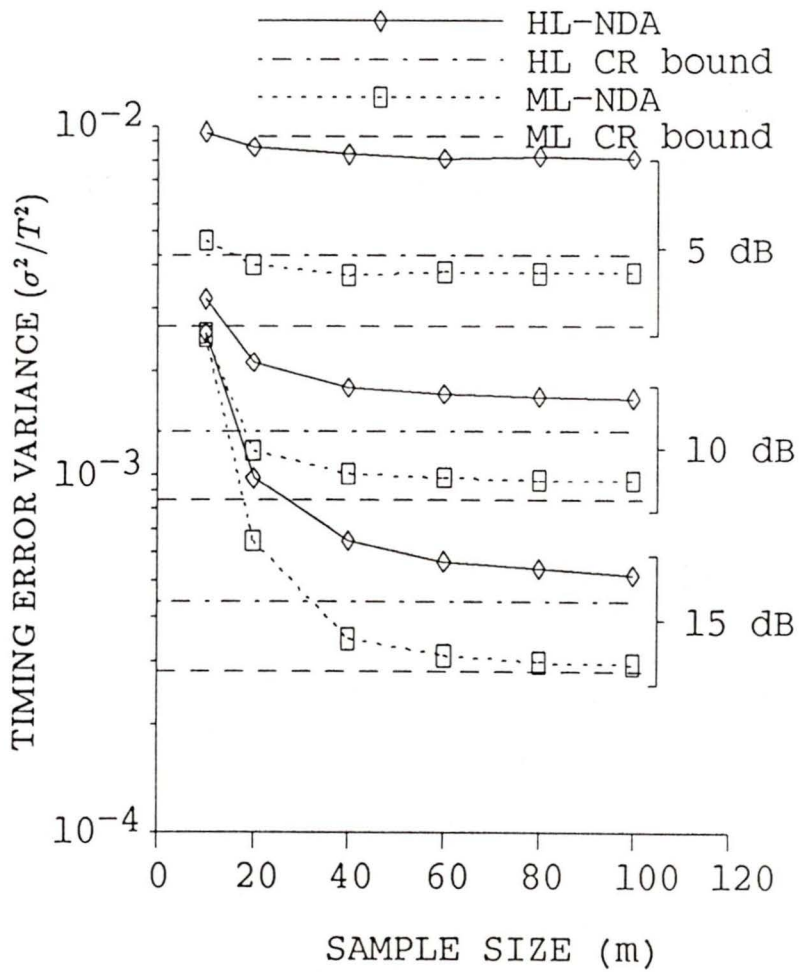


Figure 2.9: Timing error variance versus sample size m for: half sine signalling pulses; Eb/No = 5, 10, and 15 dB; K = 6 symbols.

Figure 2.10a: Data-Aided Synch.

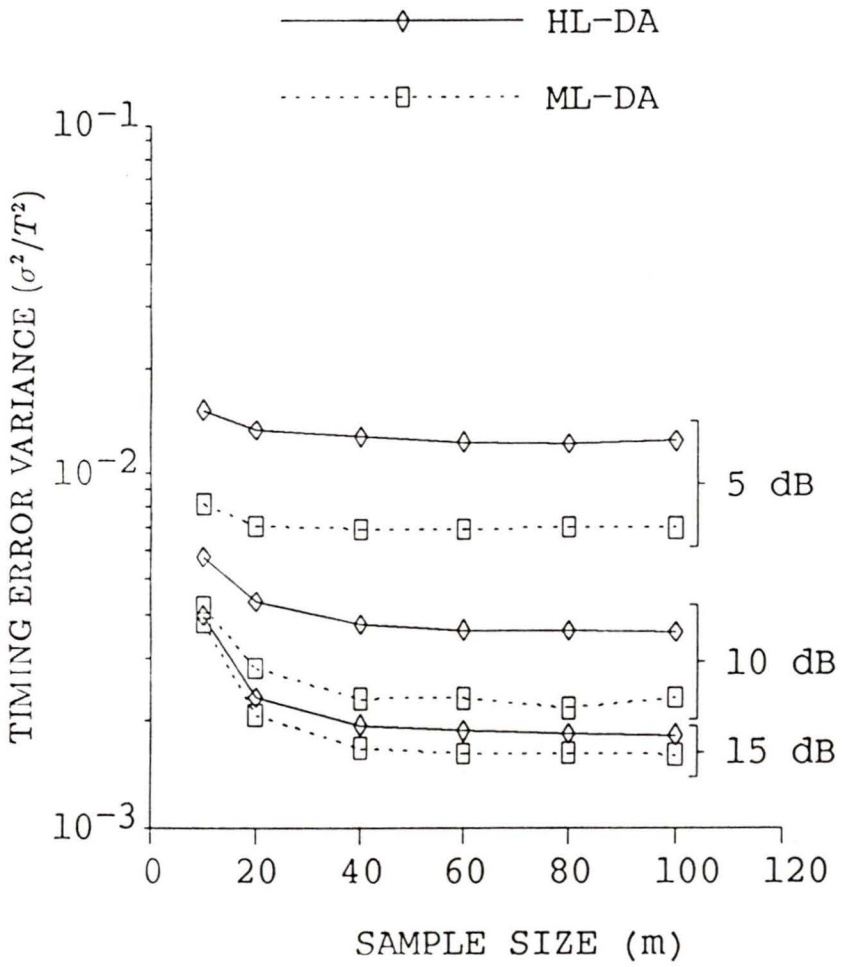


Figure 2.10b: Non-Data-Aided Synch.

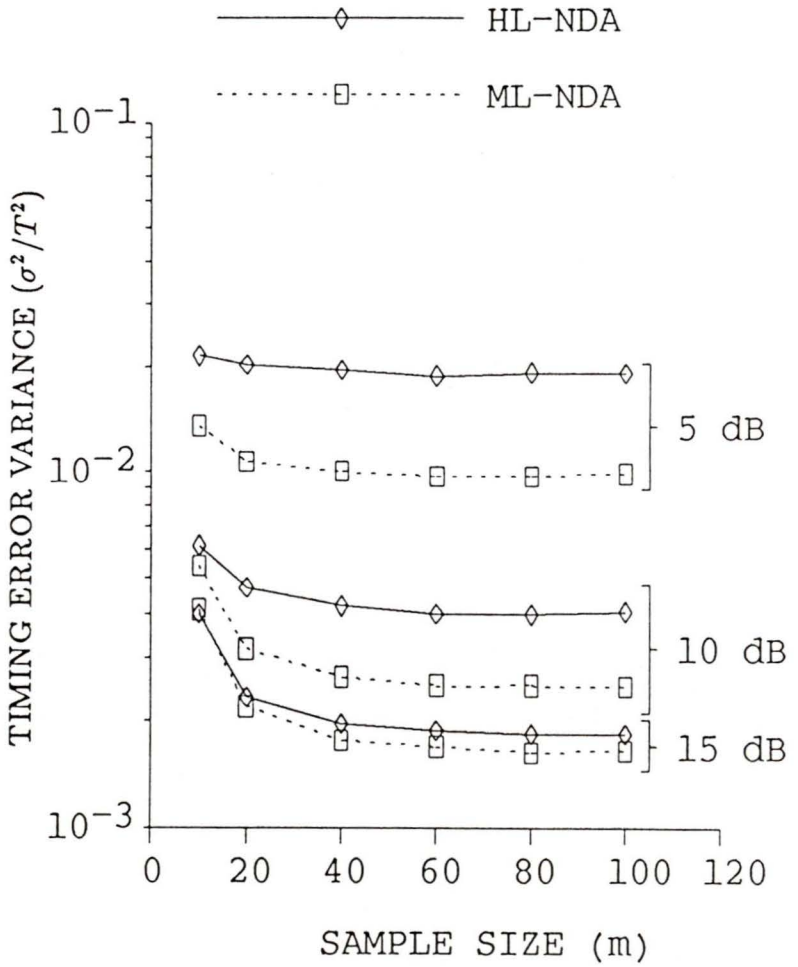


Figure 2.10: Timing error variance versus sample size m for: square signalling pulses; Eb/No = 5, 10, and 15 dB; K = 6 symbols.

2.8 Summary

ML estimation has been applied to the problem of digital binary bit synchronization where the ML estimate is generated from a hard limited data signal. The received data signal is assumed to be binary data employing time limited $(0, T)$ signalling pulses in the presence of additive white Gaussian noise. The analysis follows some of the ideas in [28,29].

The optimum (in the ML sense) HL-DA and HL-NDA synchronizers have been derived in the context of a sampled waveform digital circuit implementation. Neither synchronizer may be easily implemented, and so simpler HL-DA and HL-NDA synchronizers are derived using linear approximations that are valid for a large samle size m.

The CR lower bound for the HL-DA synchronizer is a fixed constant $(\pi/2)$ higher than the CR bound for the ML-DA synchronizer. The computer simulation results tend to confirm the $\pi/2$ degradation for a large m. At low values of m the HL and ML variance measurements tend to converge, indicating that the degradation associated with the hard limiting is less than the $\pi/2$ factor for small m.

The synchronization performance of square signalling pulses is somewhat inconclusive due to the lack of CR bound information. Since most practical communications channel have finite bandwidth, the square signalling pulse is generally not feasible. Therefore further work is required before a conclusion can be reached on the relative HL synchronization degradation for square signalling pulses filtered by a finite bandwidth.

Several other areas of HL synchronization research are left to be in-

vestigated. The optimum HL synchronizer algorithms suggest closed loop realizations based on the necessary condition for the ML estimate, given by (2.3). It is of interest to determine if the $\pi/2$ timing error variance degradation will be translated to a closed loop realization when compared to other closed loop synchronizer structures.

The effects of filtering the additive noise on the HL synchronizer performance is also of interest, since often an IF filter will band limit the noise prior to data detection and synchronization. With bandlimited noise the noise samples N_j^i are no longer independent. As the filter bandwidth decreases, ISI also becomes a concern and the synchronizer performance in the presence of ISI must also be considered. The synchronizer performance for a very small number of samples ($m < 10$) is also to be determined.

Chapter 3

Pattern Jitter Compensation Synchronizer

3.1 Foreword

A digital phase locked loop (DPLL) bit synchronizer for bandlimited binary signalling using a hard limited receiver is proposed. The synchronizer tracks the zero crossings of the received waveform. A new technique to reduce "pattern jitter" or "self noise" with a compensation signal in the feedback loop of the synchronizer is considered. In this chapter analytical results are derived for the timing jitter variance (additive noise and self noise) of the synchronizer. The analytical result shows an accuracy improvement over a previous result [18] determined for a zero crossing tracking synchronizer, and the analysis is adapted to incorporate the pattern jitter compensation technique. Computer simulations verify the effectiveness of the pattern jitter compensation technique for a synchronizer operating with both spectral raised cosine signalling pulses as well as for signalling pulses generated by a realizable filter network. The performance of a pattern jitter compensation

is confirmed with laboratory measurements as well.

The pattern jitter compensation method can be implemented in an adaptive synchronizer structure for applications where little is known about the signalling pulse waveform.

3.2 Introduction

In this chapter, the effects of bandlimiting the received data signal are briefly discussed to illustrate the potential problems associated with the synchronization of hard limited signals in a bandlimited environment. A hard limited synchronization technique appropriate for bandlimited systems is proposed, and its performance is analyzed.

The analysis of the synchronizer is confined to determining the additive noise and self noise timing error (jitter) variances. The pattern jitter compensation technique is incorporated into the analysis. For applications where little is known about the system impulse response, an adaptive pattern jitter compensation technique is also presented. Computer simulations results are presented for synchronization of both spectral raised cosine (SRC) signalling pulses [39] and for pulses that can be generated by a realizable filter network [40]. The computer simulations verify the synchronization technique for application in systems that operate fairly close to the Nyquist minimum bandwidth. Laboratory measurements of a digital signal processor implementation of a pattern jitter compensation synchronizer confirm the effectiveness of the technique.

This chapter is organized as follows: section 3.3 is a review of bandlimited digital communications, synchronization problems in bandlimited

digital communications, and bandlimited synchronization techniques. The proposed pattern jitter compensation method is described in section 3.4. Section 3.5 contains the major analytical result of this chapter, the timing jitter variance analysis of the proposed DPLL synchronizer for both additive noise and pattern jitter (self noise). The computer simulation results are presented in section 3.6. Section 3.7 contains laboratory measurements of a pattern jitter compensation synchronizer. Section 3.8 is the chapter summary.

3.3 Bandlimited Digital Communications

For many communications systems employing frequency division multiplexing, optimum spectral efficiency may require each user's baseband spectrum to occupy a (single sided) bandwidth approaching the Nyquist limit of $1/2T$, where T is the data symbol duration [39]. A system is considered bandlimited if the operating bandwidth is less than $1/T$. It is well known that a signal cannot be both bandlimited and time limited, and so bandlimited signals will have a non-zero value outside the data symbol interval, which may potentially interfere with adjacent symbols. The interference with adjacent symbols is referred to as intersymbol interference, or ISI.

Consider a received bandlimited data signal in the absence of noise

$$s(t) = \sum_k a_k x(t - kT) \tag{3.1}$$

where a_k is the data value, and $x(t)$ is the transmission system impulse response. The Nyquist criterion to detect $s(t)$ ISI free at the decision time

$t = kT$ requires

$$x(nT) = \begin{cases} A & \text{if } n = 0 \\ 0 & \text{if } n = \pm 1, \pm 2, \pm 3 \ldots \end{cases} \tag{3.2}$$

where A is an arbitrary constant. A typical Nyquist pulse is illustrated in figure 3.1. The main pulse occupies a duration of 2 symbol intervals. Prior to the main pulse lobe are the precursor tails, and after are the postcursor tails.

The generation of a Nyquist pulse requires $X(f)$, the Fourier transform of $x(t)$, to satisfy Nyquist's vestigial symmetry condition [39]

$$X(f) = \begin{cases} 1 + E(f) & |f| \leq f_N \\ E(f) & f_N < |f| < 2f_N \\ 0 & \text{elsewhere} \end{cases} \tag{3.3}$$

where $f_N = 1/2T$ is the Nyquist limit, and

$$E(f_N - f) \;=\; -E(f_N + f) \quad 0 < f < f_N \tag{3.4}$$

The function $E(f)$ is chosen to determine the excess bandwidth the system will operate with. For example $E(f) = 0$ is the ideal "brick wall filter" with a cutoff at $f = f_N$. The synchronizability of a pulse generally improves as the excess bandwidth increases (ISI decreases). At zero excess bandwidth, the magnitude of the ISI is infinite, and so the signalling pulses are not synchronizable.

A truly bandlimited signal is a physical impossibility, requiring filters of infinite order with infinite delay associated with the impulse response. Approximations to the ideal pulse shape can be generated either by a low

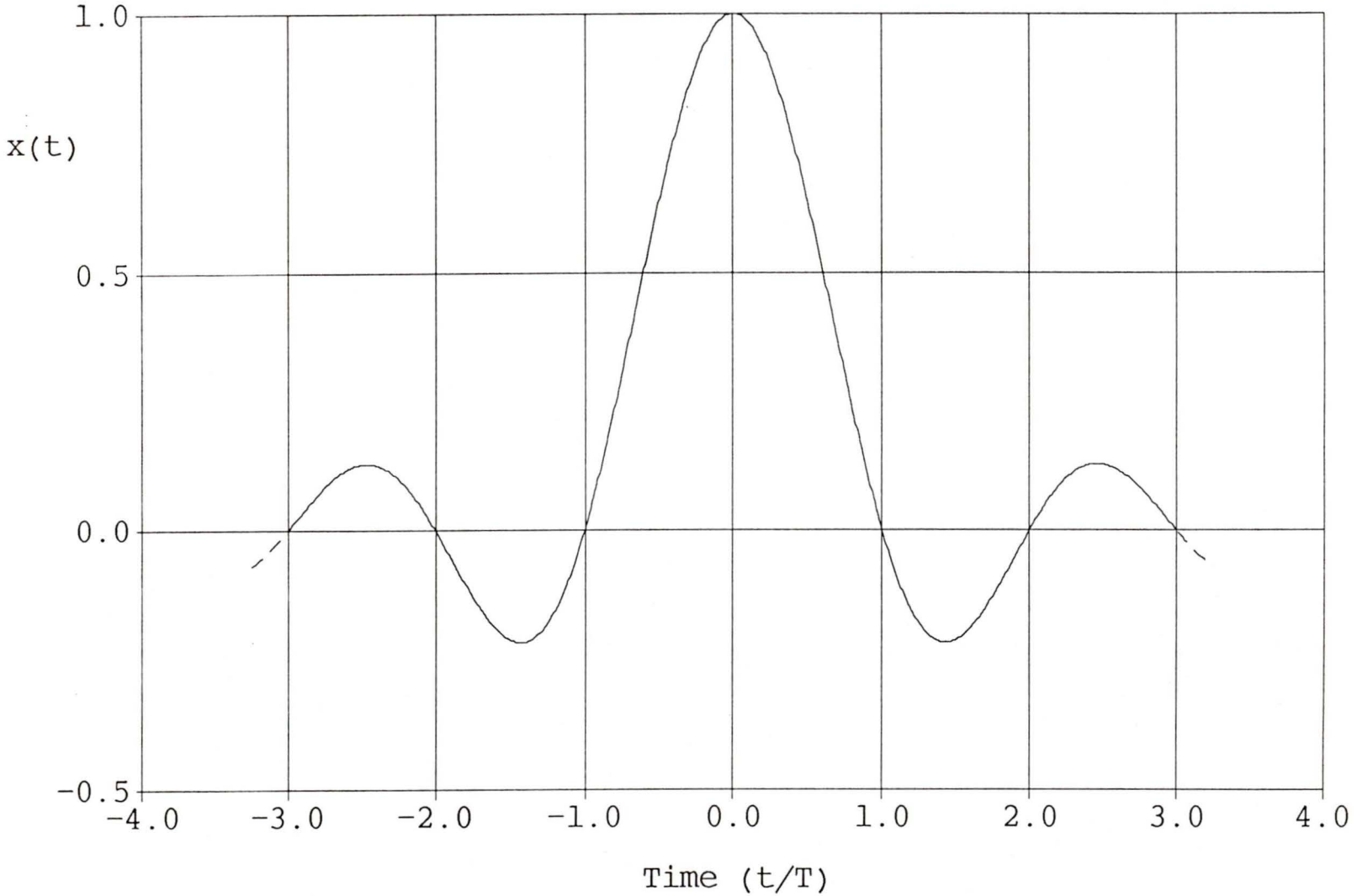


Figure 3.1: A typical Nyquist signalling pulse.

pass filter realization specifically designed to minimize ISI effects [40], or by generating the desired waveform directly with a nonrecursive filter [41]. A nonrecursive implementation requires the designer to truncate the transmitted pulse waveform to a finite duration with the assumption that the truncated portion of the waveform is negligible.

Data sampled at the ideal sampling time $t = kT$ will be unaffected by ISI. However small timing errors due to imperfect timing recovery will cause sampling at points of non-zero ISI, resulting in degraded bit error performance.

The presence of ISI will cause the synchronizer to generate a timing jitter component that is statistically independent of the timing jitter generated by the additive noise. This jitter component is termed "pattern jitter", or "self noise" [22,23] (also called pattern dependent jitter). The term pattern jitter is appropriate as the ISI is dependent on data patterns. At high signal to noise ratios the jitter performance is bounded by the pattern jitter. The synchronizer pattern jitter performance is of special concern in a regenerative synchronizer environment, where the timing signal is successively recovered along a chain of digital regenerators [42]. The maximum number of regenerators in the chain is determined by the maximum allowable cumulative pattern jitter before the error probability is not acceptable [43]. Reducing the pattern jitter to a minimum at each regenerator is required to maximize the number of regenerators that can be used.

A synchronization technique for bandlimited systems is to pass the received signal through a nonlinear device to generate a discrete spectral

line component at $1/T$, which is recovered by either a bandpass filter or a PLL [15,21,22]. This synchronizer will have a pattern jitter component present in the recovered timing signal. The pattern jitter can be reduced by reducing the bandwidth of the bandpass filter (or PLL) [44], at the expense of acquisition time [45].

It has been shown that for a synchronizer operating with a square law nonlinearity, the use of a bandpass prefilter with conjugate symmetry about $1/2T$ prior to the nonlinear device will eliminate pattern jitter [23]. For other nonlinearities a nearly optimum prefilter has been defined [46]. A disadvantage of prefiltering is that the performance is very susceptible to small frequency misadjustments [44].

3.4 Pattern Jitter Compensation

A new approach to pattern jitter reduction, based on knowledge of the impulse response $x(t)$, is presented in this chapter. For application where the impulse response is not accurately known, an adaptive implementation will be discussed in section 3.6. A zero crossing tracking digital phase locked loop is used in conjunction with a compensation signal that estimates the ISI from the received data decisions. The digital phase locked loop has several advantages over the analog systems. First, the nonlinear operation on the received data is a simple hard limiter. Second, the system is suitable for low cost digital circuit implementation in that only the hard limited signal is processed. Third, it will be shown that pattern jitter can be effectively eliminated for most practical signal to noise ratios (SNR) without

the requirement of an analog or digital prefilter.

The Nyquist criterion for zero ISI guarantees no ISI at the decision times $t = kT$. However at the zero crossing locations ISI can be quite significant, and most often is near maximum. For zero ISI at the location of the zero crossings of Nyquist filtered signalling pulses, the system bandwidth determining function (3.3) must be chosen to ensure an excess bandwidth extending to at least $1/T$ [39]. As the bandwidth of the channel is reduced below $1/T$, ISI will distort the location of the zero crossings. As the bandwidth is decreased the peak to peak amplitude of the zero crossing distortion will increase, as the tails of $x(t)$ will have a slower rate of decay [47]. A synchronizer that tracks the zero crossings will therefore have a significant pattern jitter component in the recovered timing signal.

Consider a binary transmission system in the absence of noise. The received signal is

$$s(t) = \sum_k a_k x(t - kT) \tag{3.5}$$

where $a_k \in \{-1, +1\}$ are independent zero mean binary numbers. In the interval between the a_k and a_{k+1} bits, a zero crossing will occur if and only if a_k and a_{k+1} are of opposite polarity. The transition will occur at

$$t = kT + t_o + \tau_k \tag{3.6}$$

where t_o is the nominal zero crossing location and τ_k is a zero mean random variable generated by the ISI. If the entire sequence $\{a_k\}$ is known *a priori* a synchronizer can calculate τ_k exactly and subtract it out.

If the tails of the pulse decay quickly, a synchronizer can use a finite

number of received data decisions to estimate τ_k. The received sequence

$$\{\hat{a}_k\} = \{\hat{a}_{k-l}, \ldots, \hat{a}_k, \hat{a}_{k+1}, \ldots, \hat{a}_{k+m+1}\} \tag{3.7}$$

can be used to determine $\hat{\tau}_k$, the estimate of τ_k, by solving

$$\sum_{i=-l}^{m+1} \hat{a}_{k+i} x(\hat{\tau}_k + t_o - iT) = 0 \tag{3.8}$$

As m future bits relative to a_{k+1} are required in $\{\hat{a}_k\}$, an m bit delay is required to calculate $\hat{\tau}_k$. The m bits will correspond to the expected length of the precursor, and for most applications the precursor will not be significantly longer than one or two bits [40]. In a practical system the values of $\hat{\tau}_k$ can be determined for all combinations of $\{\hat{a}_k\}$ and stored in memory. During operation the synchronizer can retrieve the stored values of $\hat{\tau}_k$. It will be shown in section 3.6 that the $\hat{\tau}_k$ values may also be determined with an adaptive synchronization structure.

3.5 Synchronizer Operation

3.5.1 Synchronizer Description

A typical baseband equivalent model of a bandlimited binary communications system is illustrated in figure 3.2 [25]. The system consists of a transmit low pass filter (LPF_T), a receive low pass filter (LPF_R), and a synchronizer to generate the appropriate samples to recover the transmitted data. The $H_T(f)$ and $H_R(f)$ filters form a matched filter pair [25]. It is assumed that the channel frequency characteristic is flat, or that fixed equalization is used to compensate for the channel characteristic.

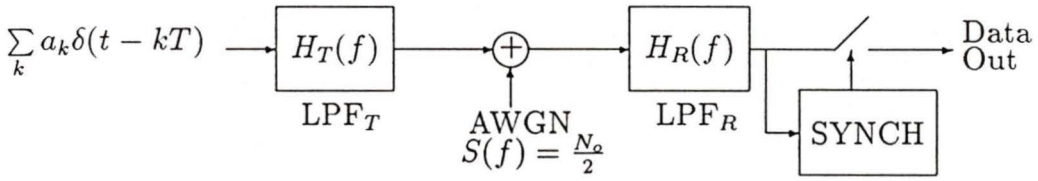


Figure 3.2: Bandlimited baseband binary data transmission system.

The received signal and noise at the LPF_R output is

$$y(t) = s(t - \epsilon) + \eta(t) \tag{3.9}$$

where $s(t)$ is defined by (3.5), ϵ is the slowly varying delay of the received signal, and $\eta(t)$ is the filtered additive white Gaussian noise. The pulse shape $x(t)$ is the impulse response of the matched filter cascade.

The synchronizer used to analyze the performance of the proposed compensation technique is illustrated in figure 3.3. The synchronizer is a zero crossing tracking phase locked loop synchronizer [18] with the addition of the pattern jitter compensation. The hard limiter reveals the zero crossings in the received signal. The values of $\hat{a}_k$ are determined by sampling at

$$t = kT + \hat{\epsilon}_k \tag{3.10}$$

where $\hat{\epsilon}_k$ is the synchronizer's estimate of ϵ_k. The synchronizer timing error e_k is

$$e_k = \epsilon_k - \hat{\epsilon}_k \tag{3.11}$$

The received sequence $\{\hat{a}_{k-m}\}$ is stored in a shift register, the values of which are used to locate in memory the correct value of $\hat{\tau}_{k-m}$. The phase detector (PD) generates an output proportional to the time difference between the location of the zero crossing, and the the predicted location, (i.e. $t = kT + t_o + \hat{\epsilon}_k$). The remainder of the synchronizer consists of the m bit delay, a loop filter F(z), and a numerically controlled oscillator (NCO).

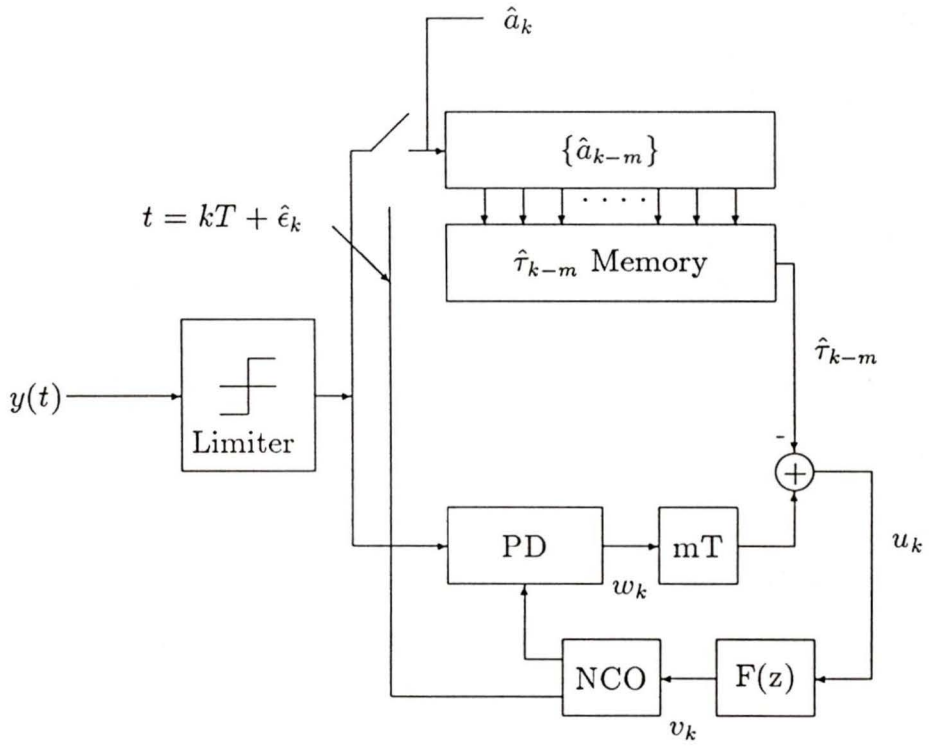


Figure 3.3: Synchronizer with pattern jitter compensation.

3.5.2 Analysis Model

In this section the model of the synchronizer used for the jitter variance analysis is discussed and analysis assumptions are stated.

For analysis purposes we follow the assumption made in [18] that the zero crossing locations are sufficiently close to the nominal locations $t = kT + t_o + \epsilon_k$ and use a two term Taylor series expansion about the nominal zero crossings as a linear approximation. Some comments about this assumption should be made. We would not expect the assumption to be valid for very small excess bandwidth systems where the ISI is too large for a linear approximation to apply. At zero excess bandwidth the magnitude of ISI at the nominal zero crossing is infinite and the assumption is certainly no longer valid. The second case where the assumption would fail is if the noise $\eta(t)$ is not sufficiently bandlimited and the SNR is too low to assume either a single zero crossing in a bit interval, or zero crossings close to the nominal location. These comments should always be considered when interpreting the following results. A comparison of the analysis in [18] and the analysis presented here is contained in appendix B.

Following these assumptions [18], if $\hat{a}_k \neq \hat{a}_{k+1}$ a zero crossing in the received waveform $y(t)$ occurs at

$$t = kT + t_o + \epsilon_k + n_k + \tau_k \tag{3.12}$$

where ϵ_k is the delay parameter, n_k is due to the additive noise, and τ_k is due to ISI. The terms ϵ_k, n_k and τ_k are all statistically independent. A typical zero crossing waveform is illustrated in figure 3.4.

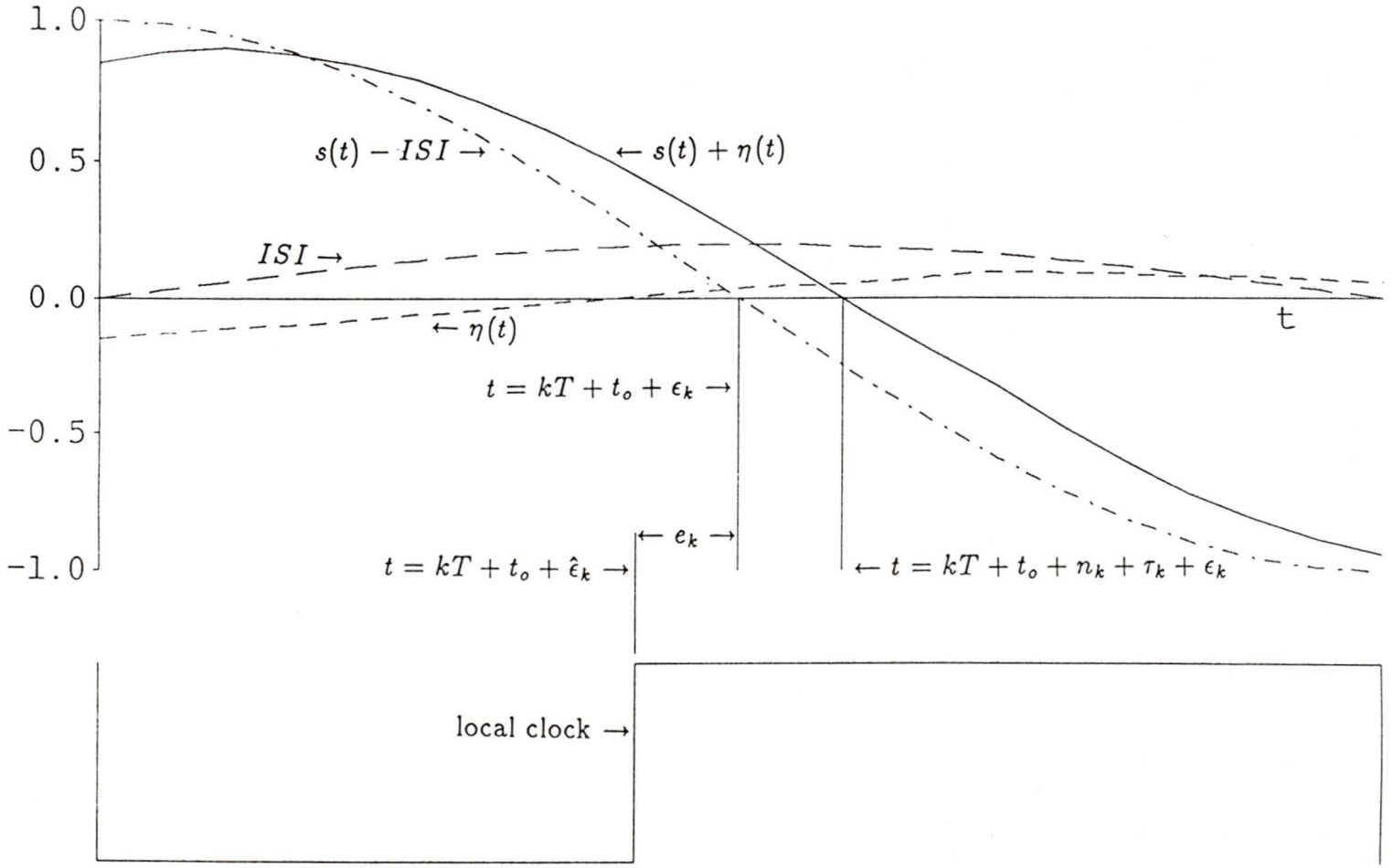


Figure 3.4: Zero crossing distortion generated by additive noise and ISI.

The linear value of n_k is

$$n_k = c \cdot a_k \eta(kT + t_o + \epsilon_k) \tag{3.13}$$

where c is the expected inverse slope of the zero crossing [18]

$$c = \frac{1}{x'(t_o) - x'(t_o - T)} \tag{3.14}$$

Similarly a linear approximation to τ_k can be made by determining the amplitude of the ISI at the nominal zero crossing location and multiplying by c [18]

$$\tau_k = c \cdot a_k \sum_{i \neq 0,1} a_{k+i} x(t_o - iT) \tag{3.15}$$

Ignoring the possibility of errors in the received data sequence, the linearized value of $\hat{\tau}_k$ is the summation in (3.15), including only the terms in the compensation sequence (3.7)

$$\hat{\tau}_k = c \cdot a_k \sum_{\substack{i=-l \\ i \neq 0,1}}^{m+1} a_{k+i} x(t_o - iT) \tag{3.16}$$

A discrete time model of the synchronizer is shown in figure 3.5. The structure is very similar to a digital phase locked loop [48], with the exception of the multiplier and the pattern jitter compensation signal in the feedback. The three statistically independent timing error sequences, n_k, τ_k, and ϵ_k, are applied to the input.

The value of b_k, the input to the multiplier, is determined by the presence of a transition [18]

$$b_k = \begin{cases} 1 & \text{if} \quad a_k \neq a_{k+1} \\ 0 & \text{if} \quad a_k = a_{k+1} \end{cases} \tag{3.17}$$

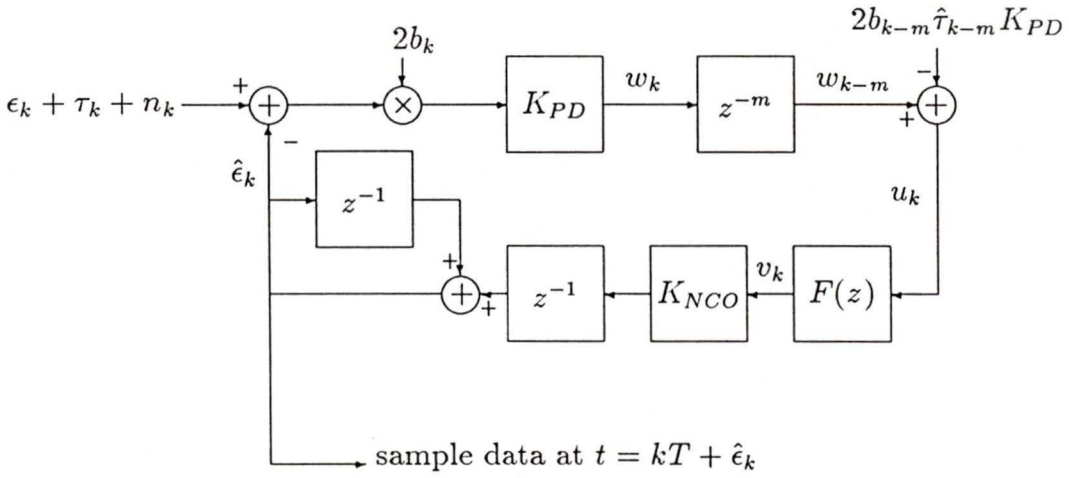


Figure 3.5: Discrete time synchronizer model.

and so the multiplier input $2b_k$ may be written as

$$2b_k = (1 - a_k a_{k+1}) \tag{3.18}$$

In figure 3.5 the output of the multiplier is scaled by the PD gain constant K_{PD}. The PD output w_k is delayed m bits and the pattern jitter compensation signal $2b_{k-m}\hat{\tau}_{k-m}K_{PD}$ is subtracted from w_{k-m}. The resulting sequence u_k is filtered by $F(z)$, and the filtered output sequence v_k is used to control the NCO. Timing adjustment are made according to

$$\hat{\epsilon}_{k+1} = \hat{\epsilon}_k + K_{NCO}v_k \tag{3.19}$$

The discrete time synchronizer model of figure 3.5 may be redrawn as shown in figure 3.6, where the transfer function $G(z)$ is

$$G(z) = \frac{K_{PD}K_{NCO}F(z)z^{-(m+1)}}{1 - z^{-1}} \tag{3.20}$$

Referring to figure 3.5, the pattern jitter compensation signal $2b_{k-m}\hat{\tau}_{k-m}K_{PD}$ may be transferred backwards through the m bit delay, the phase detector gain constant K_{PD}, and the transition multiplier b_k, and therefore may be represented at the input by $\hat{\tau}_k$ as shown in figure 3.6.

The sequence $\hat{\epsilon}_k$ is determined by

$$\hat{\epsilon}_k = \sum_{j=-\infty}^{k} g_{k-j}\{\gamma_j - 2b_j\hat{\epsilon}_j\} \tag{3.21}$$

where g_j is the impulse response of $G(z)$, and

$$\gamma_k \triangleq 2b_k\{\tau_k - \hat{\tau}_k + n_k + \epsilon_k\} \tag{3.22}$$

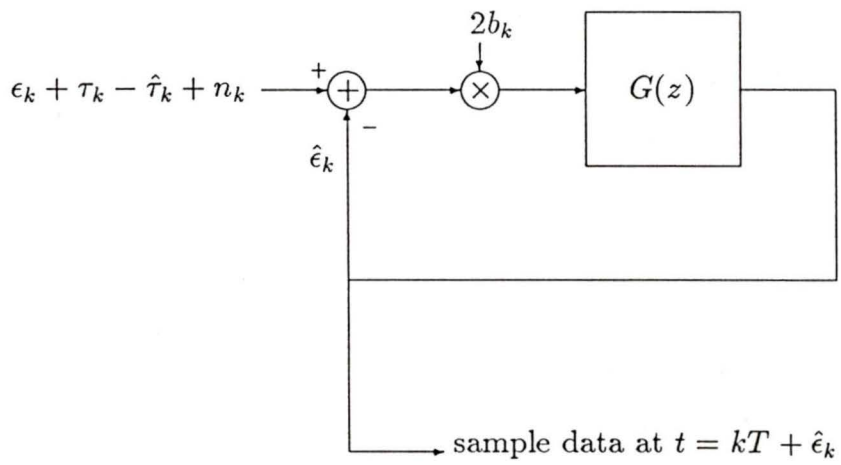


Figure 3.6: Discrete time synchronizer represented as a single transfer function feedback system.

3.5.3 Outline of Analysis

Due to the presence of the multiplier in the feedback path, a transfer function for the PLL cannot be directly derived. Saltzberg [18] has analyzed the performance of an analog synchronizer, based on the linear two term Taylor series approximation stated above. This analysis is adapted here for the discrete time model (see appendix B) with the pattern jitter compensation technique incorporated into the analysis.

Saltzberg's result for the jitter variance [18] is accurate for narrow bandwidth synchronizers. In this analysis a new result for the timing jitter variance is derived. The result may be applied to any arbitrary signalling pulse $x(t)$ and filtered noise $\eta(t)$. It is shown that if the sampled noise sequence $\eta(kT + t_o + \epsilon_k)$ is a discrete time white noise process (which for example is the case for any Nyquist matched filter system (appendix C)) the additive noise jitter variance result is correct regardless of the synchronizer bandwidth. For the pattern jitter variance analysis this analysis is not strictly correct, but computer simulation results indicate that the analysis assumptions introduce only small approximations, and the analysis is still more accurate than in [18]. In the limit of an infinitely narrow bandwidth synchronizer, the analysis presented here is equivalent to [18].

The intent of this analysis is to derive an expression for the synchronizer timing error variance (jitter variance) σ_e^2. The two components of the timing jitter variance of interest are the pattern jitter variance σ_{pj}^2 and the the noise jitter variance σ_{nj}^2, and expressions for each are derived from a general result.

The following procedure is used to determine σ_e^2: first the autocorrelation function $R_{\hat{\epsilon}}(q)$ of the sequence defined in (3.21) is evaluated. After manipulation and substitution, two assumptions are made to simplify two of the less dominant cross correlation expressions in the resulting equation. After applying the assumptions, the z transform of the simplified equation is taken, from which $R_{\hat{\epsilon}}(0)$ is found. The assumption of a zero mean $\hat{\epsilon}_k$ is made without any loss in generality to this analysis, and it is therefore shown that $R_{\hat{\epsilon}}(0) = \sigma_e^2$. The general result for $R_{\hat{\epsilon}}(0)$ is given in (3.38).

3.5.4 Jitter Variance Analysis

Rearranging (3.21) yields

$$\hat{\epsilon}_k + 2 \sum_{j=-\infty}^{k} g_{k-j} b_j \hat{\epsilon}_j = \sum_{j=-\infty}^{k} g_{k-j} \gamma_j \tag{3.23}$$

The autocorrelation of (3.23) is

$$E\{[\hat{\epsilon}_k + 2 \sum_{i=-\infty}^{k} g_{k-i} b_i \hat{\epsilon}_i][\hat{\epsilon}_{k+q} + 2 \sum_{j=-\infty}^{k+q} g_{k-j} b_j \hat{\epsilon}_j]\}$$

$$= \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} R_\gamma(i-j) \tag{3.24}$$

where $R_\gamma(q)$ is the autocorrelation function of γ_k. Expanding (3.24) yields

$$R_{\hat{\epsilon}}(q) + 2 \sum_{i=-\infty}^{k} g_{k-i} E[b_i \hat{\epsilon}_i \hat{\epsilon}_{k+q}] + 2 \sum_{j=-\infty}^{k+q} g_{k+q-j} E[b_j \hat{\epsilon}_j \hat{\epsilon}_k]$$

$$+ \quad 4 \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} E[b_i \hat{\epsilon}_i b_j \hat{\epsilon}_j]$$

$$= \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} R_\gamma(i-j) \tag{3.25}$$

Substituting (3.18) for b_i and b_j in (3.25) after rearranging gives

$$R_{\hat{\epsilon}}(q) + \sum_{i=-\infty}^{k} g_{k-i} \{R_{\hat{\epsilon}}(k+q-i) - E[a_i a_{i+1} \hat{\epsilon}_i \hat{\epsilon}_{k+q}]\}$$

$$+ \sum_{j=-\infty}^{k+q} g_{k+q-j} \{R_{\hat{\epsilon}}(j-k) - E[a_j a_{j+1} \hat{\epsilon}_j \hat{\epsilon}_k]\}$$

$$+ \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} E[(1 - a_i a_{i+1})(1 - a_j a_{j+1}) \hat{\epsilon}_i \hat{\epsilon}_j]$$

$$= \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} R_\gamma(i-j) \tag{3.26}$$

Replacing $\hat{\epsilon}_i$ and $\hat{\epsilon}_j$ with (3.21)

$$R_{\hat{\epsilon}}(q) + \sum_{i=-\infty}^{k} g_{k-i} R_{\hat{\epsilon}}(k+q-i) + \sum_{j=-\infty}^{k+q} g_{k+q-j} R_{\hat{\epsilon}}(j-k)$$

$$+ \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} \{R_{\hat{\epsilon}}(i-j) - E[a_i a_{i+1} \hat{\epsilon}_i \{\gamma_j - 2 b_j \hat{\epsilon}_j\}]$$

$$- E[a_j a_{j+1} \hat{\epsilon}_j \{\gamma_i - 2 b_i \hat{\epsilon}_i\}] - E[a_i a_{i+1} \hat{\epsilon}_i \hat{\epsilon}_j] - E[a_j a_{j+1} \hat{\epsilon}_i \hat{\epsilon}_j]$$

$$+ E[a_i a_{i+1} a_j a_{j+1} \hat{\epsilon}_i \hat{\epsilon}_j]\} = \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} R_\gamma(i-j) \tag{3.27}$$

Substituting (3.18) for b_i and b_j in (3.27), and after some manipulation the result is

$$\begin{aligned} & R_{\hat{\epsilon}}(q) + \sum_{i=-\infty}^{k} g_{k-i} R_{\hat{\epsilon}}(k+q-i) + \sum_{j=-\infty}^{k+q} g_{k+q-j} R_{\hat{\epsilon}}(j-k) \\ & + \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} \{ R_{\hat{\epsilon}}(i-j) - E[a_i a_{i+1} a_j a_{j+1} \hat{\epsilon}_i \hat{\epsilon}_j] \\ & \quad - E[a_i a_{i+1} \hat{\epsilon}_i \gamma_j] - E[a_j a_{j+1} \hat{\epsilon}_j \gamma_i] \} \\ & \quad = \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} R_{\gamma}(i-j) \end{aligned} \tag{3.28}$$

In appendix D it is demonstrated with reasonable assumptions that

$$E[a_i a_{i+1} a_j a_{j+1} \hat{\epsilon}_i \hat{\epsilon}_j] \approx R_{\hat{\epsilon}}(0) \delta(i-j) \tag{3.29}$$

and

$$E[a_i a_{i+1} \hat{\epsilon}_i \gamma_j] \approx 0 \tag{3.30}$$

The approximation in (3.29) is satisfied with equality under the following conditions

1. No ISI (i.e. $\tau_k = \hat{\tau}_k = 0$).

2. The delay parameter ϵ_k assumes a constant value (i.e. very little clock offset between the local and incoming timing references).

For the approximation in (3.30) to be satisfied with equality, the two conditions stated above must be true, and additionally the sequence $\eta(kT + t_o)$ must be discrete white noise, which from appendix C is the case for a

Nyquist matched filter. Most realizable matched filter approximations should have very little correlation associated with noise samples separated by the symbol duration T, and therefore the white noise assumption is a reasonable assumption for most applications. Note that these approximations are required only to simplify the two cross correlation expressions (3.29) and (3.30). For the other expressions in (3.28) (found to be the dominant terms by computer simulation) no restrictions are placed on the ISI, $\eta(kT + t_o + \epsilon_k)$, or ϵ_k. In summary, the two assumptions (3.29) and (3.30) will introduce small approximations into the pattern jitter variance result. If $\eta(kT + t_o + \epsilon_k)$ is not white noise, then the two assumptions represent small approximations in the additive noise jitter variance result as well.

Using (3.29) and (3.30) in (3.28)

$$R_{\hat{\epsilon}}(q) + \sum_{i=-\infty}^{k} g_{k-i} R_{\hat{\epsilon}}(k+q-i) + \sum_{j=-\infty}^{k+q} g_{k+q-j} R_{\hat{\epsilon}}(j-k)$$

$$+ \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} \{R_{\hat{\epsilon}}(i-j) - \delta(i-j) R_{\hat{\epsilon}}(0)\}$$

$$= \sum_{i=-\infty}^{k} \sum_{j=-\infty}^{k+q} g_{k-i} g_{k+q-j} R_{\gamma}(i-j) \tag{3.31}$$

The z transform can be applied to each term in (3.31)

$$S_{\hat{\epsilon}}(z)(1+G(z))(1+G(1/z)) - G(z)G(1/z)R_{\hat{\epsilon}}(0) = G(z)G(1/z)S_{\gamma}(z) \tag{3.32}$$

where the z transform of the autocorrelation function $R_{\hat{\epsilon}}(q)$ is defined to be

$$S_{\hat{\epsilon}}(z) = \sum_{q} R_{\hat{\epsilon}}(q) z^{-q} \tag{3.33}$$

with a similar definition for $S_\gamma(z)$. Rearranging (3.32) yields

$$S_{\hat{\epsilon}}(z) = \frac{G(z)G(1/z)S_\gamma(z)}{(1+G(z))(1+G(1/z))} - \frac{G(z)G(1/z)R_{\hat{\epsilon}}(0)}{(1+G(z))(1+G(1/z))} \tag{3.34}$$

Defining the feedback transfer function $H(z)$ to be

$$H(z) \triangleq \frac{G(z)}{1+G(z)} \tag{3.35}$$

equation (3.34) is

$$S_{\hat{\epsilon}}(z) = H(z)H(1/z)S_\gamma(z) + H(z)H(1/z)R_{\hat{\epsilon}}(0) \tag{3.36}$$

The mean squared value of the discrete random variable $\hat{\epsilon}_k$ is [50]

$$R_{\hat{\epsilon}}(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{\hat{\epsilon}}(\omega)d\omega \tag{3.37}$$

Using (3.36) and rearranging we find

$$R_{\hat{\epsilon}}(0) = \frac{\frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2 S_\gamma(\omega)d\omega}{1-2B_LT} \tag{3.38}$$

where

$$S(\omega) = S(z)|_{z=e^{j\omega}} \tag{3.39}$$

and $2B_LT$ is the equivalent noise bandwidth of the transfer function $H(z)$ [48,49]

$$2B_LT = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2 d\omega \tag{3.40}$$

Note that ω is normalized such that $T = 1$.

Comparing (3.38) to the result obtained for $R_{\hat{\epsilon}}(0)$ by Saltzberg [18] [1] it is evident that the two equations differ in the sign of $2B_LT$ in the denominator. The two results for $R_{\hat{\epsilon}}(0)$ are compared in appendix B; for small values of $2B_LT$ they are approximately equivalent.

The power spectral density (psd) $S_\gamma(\omega)$ is determined from the autocorrelation function $R_\gamma(q)$. From statistical independence, the sequence γ_k defined in (3.22) can be partitioned into the sum of three mutually uncorrelated sequences

$$\gamma_k = 2b_k(\tau_k - \hat{\tau}_k) + 2b_k n_k + 2b_k \epsilon_k \tag{3.41}$$

with the following autocorrelation function

$$\begin{aligned} R_\gamma(q) &= 4E[b_k(\tau_k - \hat{\tau}_k)b_{k+q}(\tau_{k+q} - \hat{\tau}_{k+q})] + 4E[b_k n_k b_{k+q} n_{k+q}] \\ &\quad +4E[b_k \epsilon_k b_{k+q} \epsilon_{k+q}] \qquad (3.42) \\ &= R_I(q) + R_N(q) + R_D(q) \qquad (3.43) \end{aligned}$$

Note that $R_I(q)$ is related to the ISI disturbance, $R_N(q)$ the additive noise, and $R_D(q)$ the channel delay. The power spectrum $S_\gamma(\omega)$ is the Fourier transform of (3.43)

$$S_\gamma(\omega) = S_I(\omega) + S_N(\omega) + S_D(\omega) \tag{3.44}$$

The autocorrelation function $R_D(q)$ is determined by ϵ, the slowly varying delay of the received data signal and can be used to model frequency offsets. Since we are concerned with the additive noise and pattern jitter

[1] Equation (62) of [18], with appropriate symbol substitutions and ignoring the psd due to pilot tone recovery.

variances only, we may assume without any loss of generality that $\epsilon_k = 0$, and therefore neglect $S_D(\omega)$ in the jitter variance calculation. With this assumption, and substituting (3.44) into (3.38), the mean square value of $\hat{\epsilon}$ is

$$R_{\hat{\epsilon}}(0) = \frac{\frac{1}{2\pi}\int_{-\pi}^{\pi}|H(\omega)|^2[S_I(\omega)+S_N(\omega)]d\omega}{1-2B_LT} \tag{3.45}$$

Both $S_I(\omega)$ and $S_N(\omega)$ are generated by zero mean random variables, and so $\hat{\epsilon}_k$ is zero mean. Therefore the jitter variance is

$$\sigma_e^2 = R_{\hat{\epsilon}}(0) = \sigma_{pj}^2 + \sigma_{nj}^2 \tag{3.46}$$

3.5.5 Pattern Jitter Variance

In this section we evaluate the pattern jitter variance σ_{pj}^2 using (3.45) with $S_N(\omega) = 0$

$$\sigma_{pj}^2 = \frac{\frac{1}{2\pi}\int_{-\pi}^{\pi}|H(\omega)|^2 S_I(\omega)d\omega}{1-2B_LT} \tag{3.47}$$

The autocorrelation function $R_I(q)$ is required to find the psd of the ISI disturbance $S_I(\omega)$. From (3.42) and (3.43), $R_I(q)$ is

$$R_I(q) = 4E[b_k(\tau_k - \hat{\tau}_k)b_{k+q}(\tau_{k+q} - \hat{\tau}_{k+q})] \tag{3.48}$$

Substituting (3.15), (3.16) and (3.18) for τ_k, $\hat{\tau}_k$, and b_k respectively

$$R_I(q) = c^2 \sum_{i\notin P}\sum_{j\notin P} E[(a_k - a_{k+1})(a_{k+q} - a_{k+q+1})a_{k-i}a_{k+q-j}]x(t_o - iT)x(t_o - jT) \tag{3.49}$$

where P is the sequence of terms eliminated by the compensation $\hat{\tau}_k$

$$P = \{-l, \ldots, 0, 1, \ldots, m+1\} \tag{3.50}$$

Retaining only the non-zero expectation terms in (3.49),

$$R_I(0) = 2c^2 \sum_{i \notin P} x^2(t_o - iT) \tag{3.51}$$

$$R_I(\pm 1) = -c^2 \{ \sum_{i \notin P \cup Q} x(t_o - iT)x(t_o - (i+1)T) - A_2 B_{-2} \} \tag{3.52}$$

$$R_I(q) = c^2[A_q - B_q][A_{-q} - B_{-q}] \qquad \text{for } q \neq 0, \pm 1 \tag{3.53}$$

where

$$\begin{aligned} A_q &= \begin{cases} x(t_o - qT) & \text{for } q \notin P \\ 0 & \text{for } q \in P \end{cases} \\ B_q &= \begin{cases} x(t_o - (q+1)T) & \text{for } q \notin Q \\ 0 & \text{for } q \in Q \end{cases} \end{aligned} \tag{3.54}$$

and Q is the sequence

$$Q = \{-(l+1), \ldots, -1, 0, \ldots, m\} \tag{3.55}$$

The psd of the ISI disturbance, $S_I(\omega)$, is the Fourier transform of $R_I(q)$, found from (3.51), (3.52), and (3.53)

$$S_I(\omega) = R_I(0) + 2 \sum_{q=1}^{\infty} R_I(q) cos(\omega q) \tag{3.56}$$

The psd $S_I(\omega)$ may be used in (3.47) to determine σ_{pj}^2. However we now show that for a strictly bandlimited $x(t)$ the two infinite summations in the expressions for $R_I(0)$ and $R_I(\pm 1)$ defined in (3.51) and (3.52) respectively may be reduced to a finite summation to simplify $S_I(\omega)$.

Consider the waveform $v(t)$ which is the product of two time shifted versions of $x(t)$

$$v(t) \triangleq x(t_1 - t)x(t_2 - t) \tag{3.57}$$

The Fourier transform of $v(t)$ is

$$V(f) = e^{j2\pi f t_1} \int_{-\infty}^{\infty} X^*(\zeta)X(\zeta + f)e^{j\pi v(t_2 - t_1)}d\zeta \tag{3.58}$$

where $X^*(f)$ is the complex conjugate of $X(f)$. Sampling $v(t)$ at $t = iT$ yields the sequence

$$v(iT) = x(t_1 - iT)x(t_2 - iT) \tag{3.59}$$

From the Poisson summation formula

$$\sum_i v(iT) = \frac{1}{T}\sum_p V(-p/T) \tag{3.60}$$

Using (3.58), the infinite summation is therefore

$$\sum_i v(iT) = \frac{1}{T}\sum_p e^{-j2\pi p t_1/T} \int_{-\infty}^{\infty} X^*(f)X\left(f + \frac{p}{T}\right)e^{j2\pi f(t_2 - t_1)}df \tag{3.61}$$

This general result may be applied to any $x(t)$. For the case where $x(t)$ is bandlimited, $X(f)$ is confined to the interval $[-1/T, 1/T]$. Hence the only non-zero terms in (3.61) are $p = -1, 0, 1$. For an $x(t)$ that is not bandlimited, more terms are required in the summation.

Consider the specific case where $x(t)$ is bandlimited, and the nominal zero crossing location $t_o = T/2$. Equation (3.51) may then be written as

$$R_I(0) \;=\; 2c^2\{\, \sum_i x^2(T/2 - iT) - \sum_{i \in P} x^2(T/2 - iT)\,\} \tag{3.62}$$

The infinite summation in (3.62) is found using the three non-zero terms in (3.61), with $t_1 = t_2 = T/2$, and thus $R_I(0)$ is given by the finite sum

$$R_I(0) = 2c^2\{\frac{1}{T}\sum_{p=-1}^{1}(-1)^p\int_{-\infty}^{\infty}X^*(f)X\left(f+\frac{p}{T}\right)df - \sum_{i\in P}x^2(T/2-iT)\} \tag{3.63}$$

Similarly, letting $t_1 = T/2$, and $t_2 = -T/2$ in (3.61), and substituting for the infinite summation in (3.52), the resulting finite sum is

$$\begin{aligned} R_I(\pm 1) &= -c^2\left\{\sum_i x(T/2-iT)x(-T/2-iT)\right. \\ &\quad \left. - \sum_{i\in P\cup Q} x(T/2-iT)x(-T/2-iT) - A_2B_{-2}\right\} \\ &= -c^2\left\{\frac{1}{T}\sum_{p=-1}^{1}(-1)^p\int_{-\infty}^{\infty}X^*(f)X\left(f+\frac{p}{T}\right)e^{-j2\pi fT}df\right. \\ &\quad \left. - \sum_{i\in P\cup Q} x(T/2-iT)x(-T/2-iT) - A_2B_{-2}\right\} \end{aligned} \tag{3.64}$$

The two finite sum expressions (3.62) and (3.64) simplify the calculation of $S_I(\omega)$, defined in (3.56). The pattern jitter variance σ_{pj}^2 is found using (3.56) in (3.47).

3.5.6 Additive Noise Jitter Variance

In this section we determine the noise jitter variance σ_{nj}^2 using 3.45

$$\sigma_{nj}^2 = \frac{\frac{1}{2\pi}\int_{-\pi}^{\pi}|H(\omega)|^2 S_N(\omega)d\omega}{1-2B_LT} \tag{3.65}$$

where $S_N(\omega)$ is determined from the autocorrelation of the additive noise disturbance $R_N(p)$, which from (3.43) is

$$R_N(q) = 4E[b_k n_k b_{k+q} n_{k+q}] \tag{3.66}$$

Using the linear approximation (3.13) for n_k and (3.18) for b_k

$$\begin{aligned} R_N(q) &= c^2 E[(a_k - a_{k+1})(a_{k+q} - a_{k+q+1}) \\ &\quad \eta(kT + t_o + \epsilon)\eta((k+q)T + t_o + \epsilon)] \end{aligned} \tag{3.67}$$

and since the data sequence $\{a_k\}$ is statistically independent of the additive noise $\eta(t)$

$$R_N(q) = c^2 E[(a_k - a_{k+1})(a_{k+q} - a_{k+q+1})]R_\eta(qT) \tag{3.68}$$

where $R_\eta(\tau)$ is the continuous autocorrelation function of the noise $\eta(t)$. The elements of the sequence $\{a_k\}$ are zero mean and statistically independent, and so evaluation of the expectation in (3.68) yields the previously derived result [18]

$$R_N(0) = 2c^2 R_\eta(0) \tag{3.69}$$

$$R_N(\pm 1) = -c^2 R_\eta(\pm T) \tag{3.70}$$

$$R_N(q) = 0 \qquad \text{for } q \neq 0, \pm 1 \tag{3.71}$$

The psd $S_N(w)$ is the Fourier transform of $R_N(q)$

$$S_N(\omega) = R_N(0) + 2R_N(1)cos(\omega) \tag{3.72}$$

$$= 2c^2\{\sigma_\eta^2 - R_\eta(T)cos(\omega)\} \tag{3.73}$$

where $R_\eta(0) = \sigma_\eta^2$ is the variance of the zero mean $\eta(t)$. For any LPF_R filter that satisfies $R_\eta(\pm T) = 0$) (for example a Nyquist matched filter system from (C.6)) the psd $S_N(\omega)$ is a discrete time white noise process

$$S_N(\omega) = 2c^2\sigma_\eta^2 \tag{3.74}$$

Note that if the autocorrelation of the noise $\eta(t)$ at the LPF_R output does not satisfy (C.6), then $S_N(\omega)$ is not white noise, and (3.73) is required to calculate $S_N(\omega)$.

Assuming $S_N(\omega)$ satisfies (3.74), from (3.65) the jitter variance σ_{nj}^2 due to the additive noise $\eta(t)$ is

$$\sigma_{nj}^2 = \frac{2c^2\sigma_\eta^2 2B_L T}{1 - 2B_L T} \tag{3.75}$$

We now make some assumptions to allow us to express (3.75) in a more recognizable form. First define the normalized pulse $\tilde{x}(t) \triangleq x(t)/A$, where A is the amplitude of the received data signal $s(t - \epsilon)$ at the optimum data sample location $t = kT + \epsilon_k$. Assuming $x(t)$ is reasonably symmetrical about $t = 0$, the nominal zero crossings are located midway between the data samples. From (3.14), c may then be approximated by

$$c \approx \frac{1}{2A\tilde{x}'(T/2)} \tag{3.76}$$

For optimal matched filter detection of binary antipodal signals it is known that [25]

$$\frac{A^2}{\sigma_\eta^2} = \frac{2E_b}{N_o} \tag{3.77}$$

where E_b is the transmitted bit energy at the LPF_R input, and N_o is the single sided power spectral density of the AWGN. Using (3.76) and (3.77) in (3.75)

$$\sigma_{nj}^2 \approx \frac{N_o B_L T}{2E_b \tilde{x}'(T/2)(1 - 2B_L T)} \tag{3.78}$$

The noise jitter variance may be reduced by increasing E_b/N_o, reducing the PLL bandwidth $2B_L T$ (at the expense of acquisition performance), or by increasing the slope of the waveform $\tilde{x}(T/2)$. Increasing $\tilde{x}'(T/2)$ implies increasing the high frequency content of the waveform $x(t)$ and therefore an increase in the bandwidth of the system.

In appendix E the additive noise jitter variance is rederived for a special case synchronizer, specifically the synchronizer used in the computer simulations. This analysis result clearly demonstrates the effect of the m bit delay in the feedback path on the noise jitter performance.

3.6 Adaptive Pattern Jitter Compensation

The analysis of the pattern jitter compensation technique assumes that $x(t)$ is known at the synchronizer. Often due to filter imperfections and non ideal channel equalization $x(t)$ may not be accurately known. It is therefore of interest to examine an adaptive pattern jitter compensation

technique based on the previous analysis.

Consider the linearized value of the compensation sequence $\hat{\tau}_k$ defined in (3.16). If $x(t)$ is not known, the value of $x(t_o - iT)$ may be replaced by an estimated value $\hat{x}_{k,i}$. The unknown scaling constant c may also be lumped in with this estimate. The adaptive estimate of $\hat{\tau}_k$ may be written as

$$\hat{\tau}_k = a_k \sum_{\substack{i=-l \\ i \neq 0,1}}^{m+1} a_{k+i} \hat{x}_{k,i} \tag{3.79}$$

Figure 3.7 shows the implementation of an adaptive synchronizer.

The estimates $\hat{x}_{k,i}$ may be recursively generated using adaptive filtering techniques. A useful algorithm for simple implementation purposes is the stochastic gradient algorithm [51]. Applying the stochastic gradient algorithm to this application yields the following coefficient update equation

$$\hat{x}_{k+1,i} = \hat{x}_{k,i} + \beta u_k a_{k-m} a_{k+i} \tag{3.80}$$

where u_k is from figure 3.7 and β is the update equation gain constant. The value of β should be chosen to be small enough to ensure that the adaptation does not interfere with the acquisition performance of the synchronizer.

3.7 Simulation Results

A computer simulation of the pattern jitter compensation synchronizer is required to verify that the assumptions made in the analysis are reasonable, and to determine the effectiveness of the proposed synchronization technique.

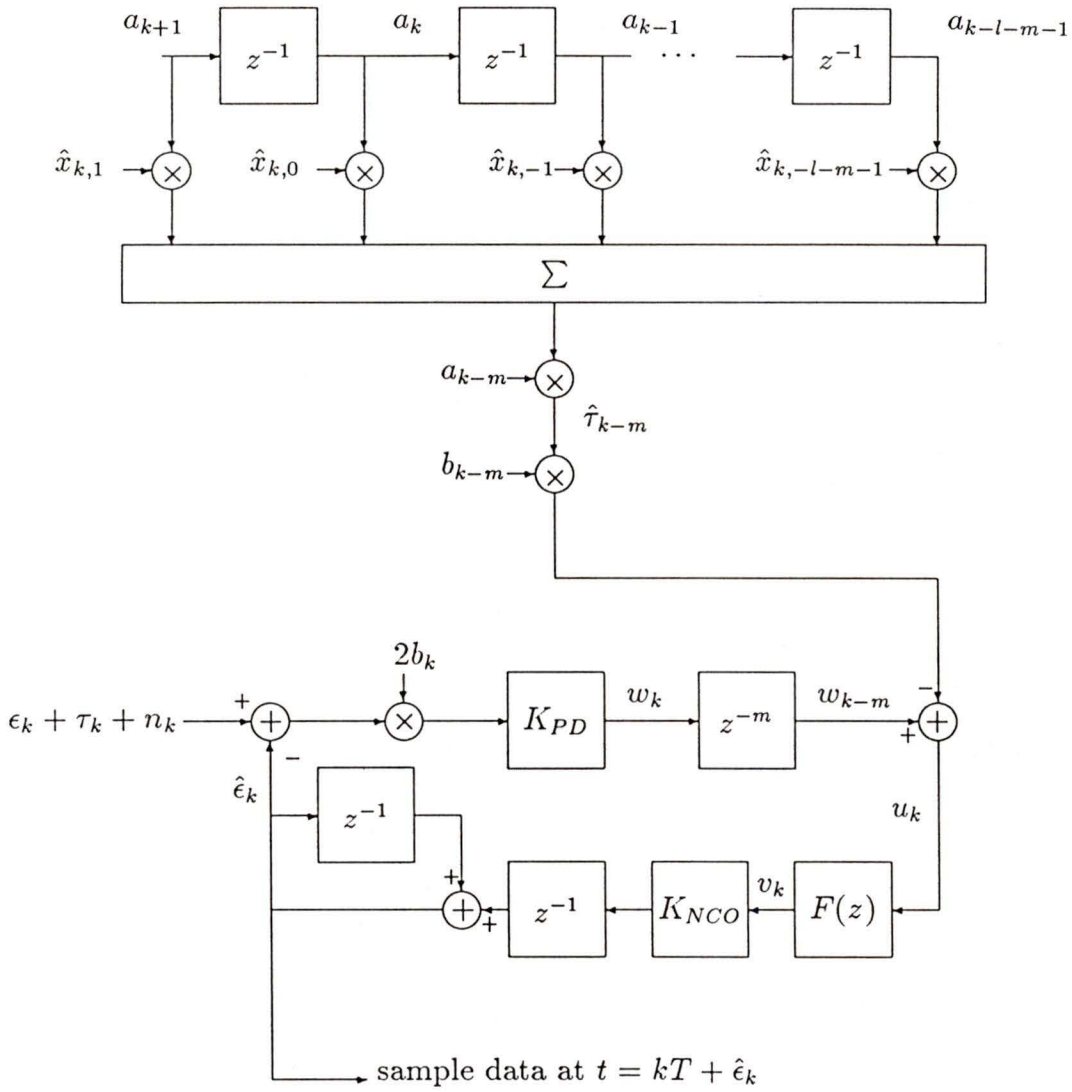


Figure 3.7: Adaptive pattern jitter compensation synchronizer.

The synchronizer used in the simulations operates with $F(z) = 1$, and with a loop gain parameter $K \triangleq K_{PD}K_{NCO}$. Two simulation methods, a linear simulation and an actual simulation, are used. The linear simulation uses the linearized values of n_k, τ_k, and $\hat{\tau}_k$ from (3.13), (3.15), and (3.16) respectively to determine the timing jitter variance σ_e^2 using Monte Carlo methods. The actual simulation results are based on timing jitter variance estimates from the generation of actual signal and noise waveforms. The simulation methods are described in appendix F.

Simulation results are presented for both spectral raised cosine (SRC) signalling pulses [25], and for a realizable data transmission filter [40]. Both a linear simulation and an actual simulation is done for the SRC pulses, while a linear simulation is done for the data transmission filter. The two pulses used for $x(t)$ are described in the following.

SRC Signalling

The SRC filter may be used as a benchmark for evaluation and analysis of bandlimited communications systems. For the synchronizer application described here, use of the SRC filter allows the synchronizer performance to be tested over the bandlimited frequency range ($0-100\%$ excess bandwidth relative to $f_N = 1/2T$) with a relatively simple filter specification. The filter impulse response $x(t)$ for a SRC filter is [25]

$$x(t) = \frac{sin(\pi t/T)}{\pi t/T}\left[\frac{cos(\alpha\pi t/T)}{1-4\alpha^2(t/T)^2}\right] \tag{3.81}$$

The parameter α determines the percent excess bandwidth of the system where $0 \leq \alpha \leq 1$.

It is interesting to observe the ISI spectrum $S_I(\omega)$ from (3.56) for the SRC pulse $x(t)$ to determine how the spectrum is altered as the number of terms in compensation sequence (3.50) length is increased. Figure 3.8 shows $S_I(\omega)$ for $\alpha = 0.3$, with the number of terms in the compensation sequence P as a parameter. The psd $S_I(\omega)$ is clearly not white, and tends to zero as ω approaches zero. Increasing the length of the compensation sequence reduces the magnitude of the psd across the frequency range.

The psd $S_N(\omega)$ from (3.74) is assumed to be white for the SRC filter.

Data Transmission Filter

In practice a signalling pulse must be an approximation to the ideal bandlimited pulse. It is therefore of interest to evaluate the performance of the synchronizer on a data transmission system using signalling pulses that can be generated by a realizable filter network. A design technique for data transmission filters that minimizes the mean square value of ISI at the optimum sampling time is given in [40]. A seventh order filter with an impulse response that has significant ISI is defined by the Laplace transfer function poles and zeros listed in table 3.1. The impulse response of the filter is shown in figure 3.9. The postcursor has the expected gradually decaying tails. The precursor length is $1.5T$ and consequently the longest delay required in the feedback path of the synchronizer is $m = 1$. The computer generated eye pattern associated with the impulse response of the filter is shown in figure 3.10. The eye pattern has only a very small

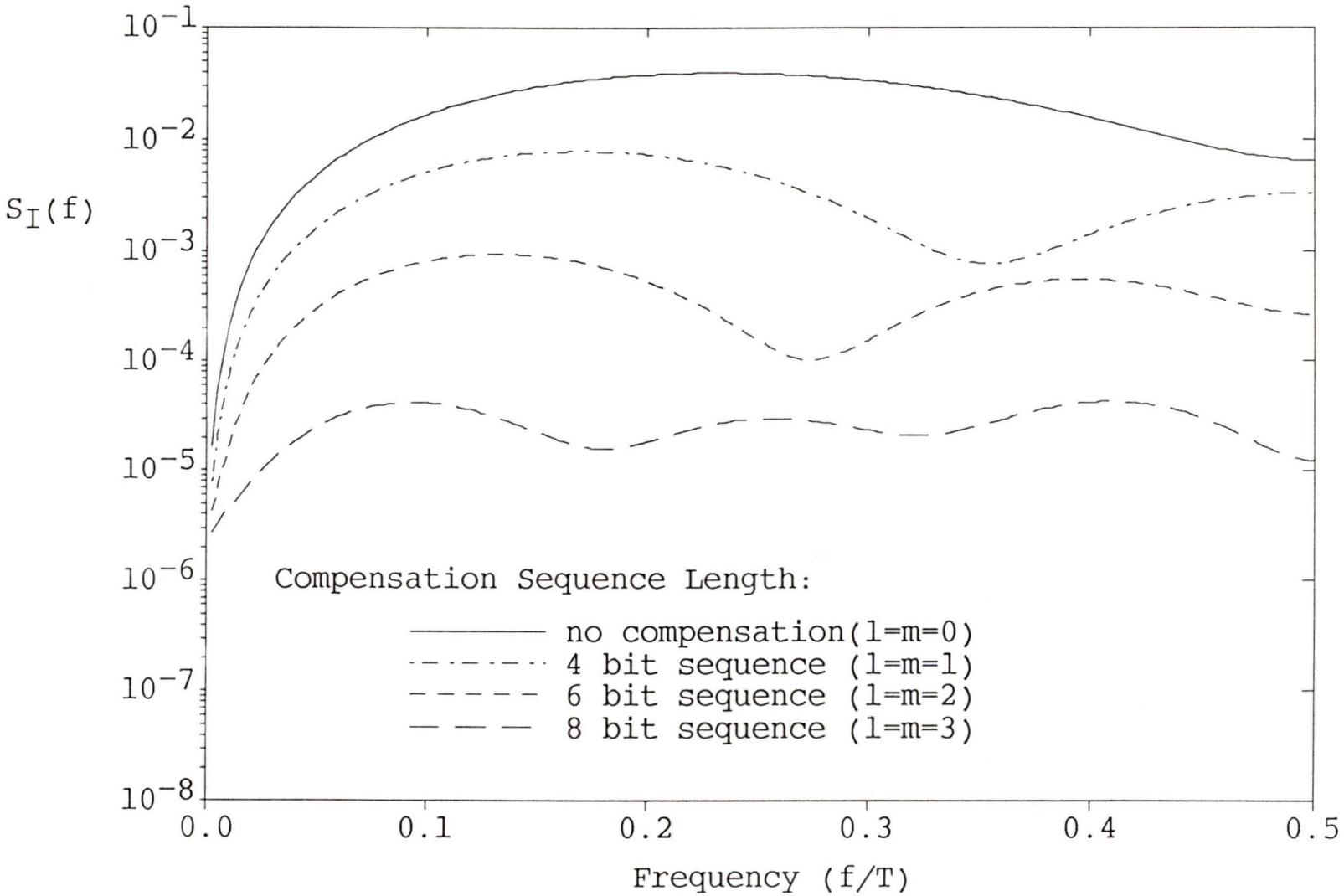


Figure 3.8: Power spectral density $S_I(f)$ of a spectral raised cosine x(t) (alpha = 0.3) for various length compensation sequences.

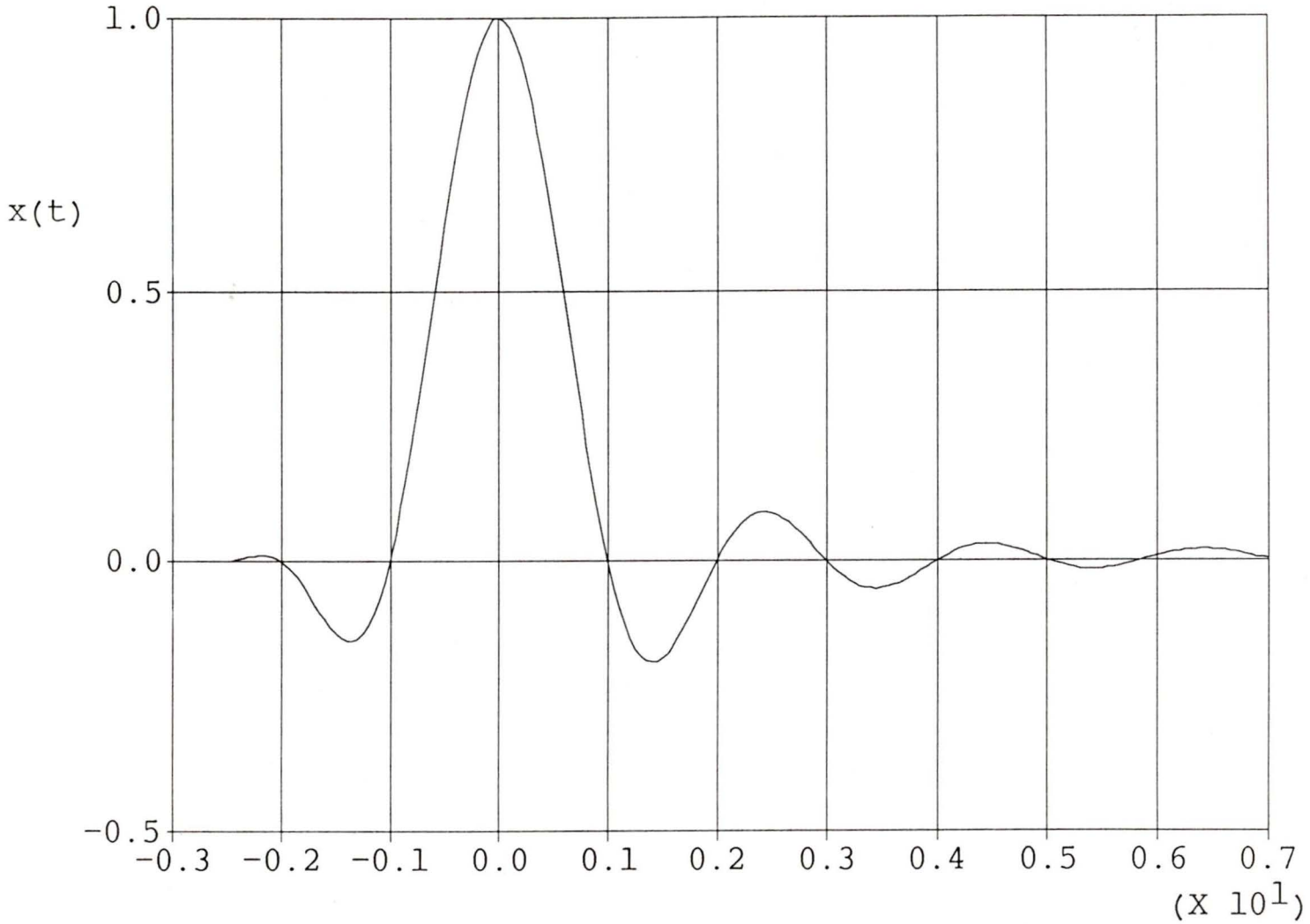


Figure 3.9: Impulse response of the seventh order data transmision filter.

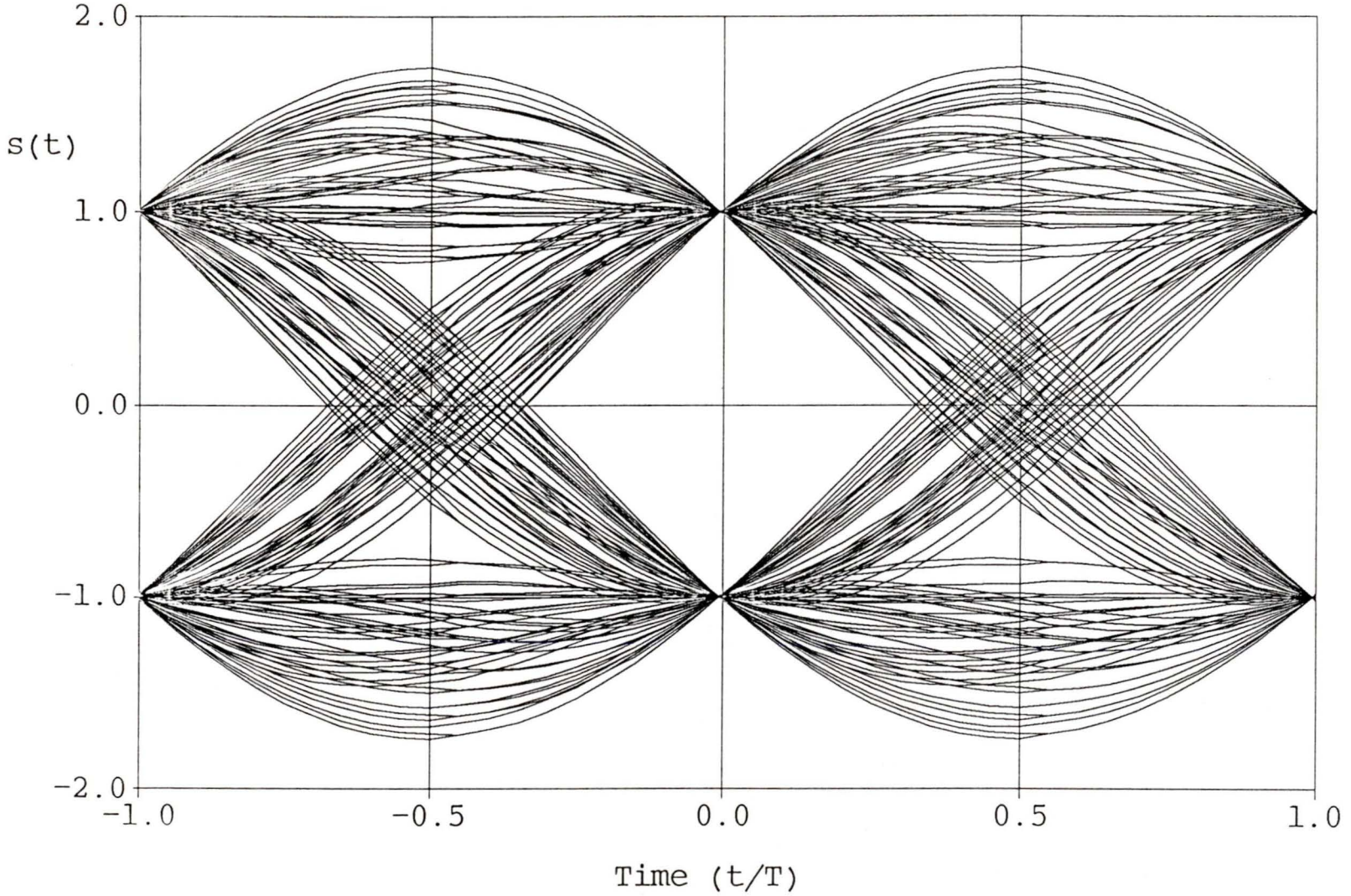


Figure 3.10: Eye pattern generated by the seventh order data transmission filter.

$p_{1,2}$	$-0.4236 \pm j3.01469$
$p_{3,4}$	$-.71633 \pm j2.13020$
$p_{5,6}$	$-.77845 \pm j1.05600$
p_7	$-.76847$
$z_{1,2}$	± 2.12453
$z_{3,4}$	± 4.74191

Table 3.1: Poles and zeros of the data transmission filter.

amount of ISI at the sample time. Also apparent is the large distortion of the zero crossings due to the ISI, which for this filter has a peak to peak value of $0.37T$. The filter frequency response shown in figure 3.11 indicates an attenuation of about $6dB$ at f_N, and greater than $40dB$ at $2f_N$.

The data transmission model shown in figure 3.2 requires that the response of the transmission filter be divided into a matched filter pair. The pattern jitter result does not depend on matched filter operation but only on the final impulse response $x(t)$ of the cascade of the two filters. Optimum probability of error performance requires matched filter operation. Since we are only concerned with a linear simulation, the variance σ_η^2 is required to generate the linearized sequence n_k defined in (3.13). For this pulse shape it is assumed that the variance σ_η^2 is equal to the optimum for matched filter operation [25], and is determined according to (3.77). From (C.6), the samples $\eta(kT + t_o + \epsilon_k)$ in (3.13) are assumed to be independent from sample to sample. The correctness of these two assumptions is dependent on how accurately the filter can be partitioned into an approximate matched filter cascade. For simplicity the optimum performance is assumed.

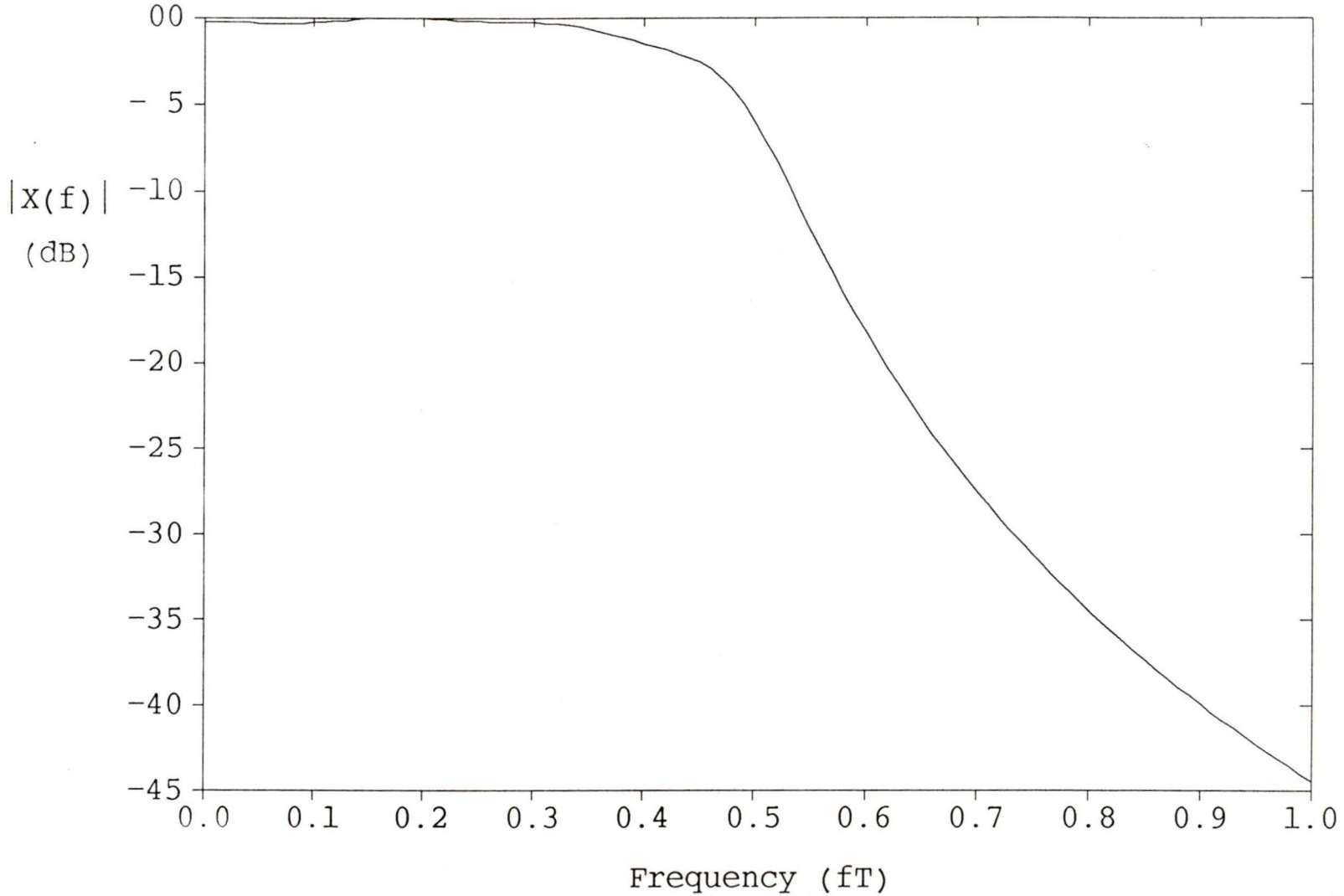


Figure 3.11: Frequency response of the seventh order data transmission filter.

In the following, the simulation results are compared to the calculated σ_{nj}^2 and σ_{pj}^2, determined by numerical integration of (3.75) and (3.47) respectively.

3.7.1 Simulations with a Spectral Raised Cosine Filter

The performance of the pattern jitter compensation technique with SRC signalling pulses is compared to the uncompensated synchronizer for the following number of terms in pattern jitter compensation estimate $\hat{\tau}_k$: 4 bits ($l = m = 1$), 6 bits ($l = m = 2$), and 8 bits ($l = m = 3$).

Linear Simulation

In this section we present the jitter variance results from a computer simulation of the synchronizer based on the linearized values for n_k, τ_k and $\hat{\tau}_k$ from (3.13), (3.15), and (3.16) respectively, and a SRC $x(t)$.

Figure 3.12 compares the calculated and linear simulation results for σ_{pj}^2. The value of K is adjusted such that $2B_LT = 0.1$ for the uncompensated synchronizer ($m = 0$). It is apparent that at very small values of α, the data sequence lengths are not sufficient to yield a significant improvement. The improvement for $\alpha \geq 0.2$ is quite good. As stated previously, the assumptions (3.29) and (3.30) introduce slight approximations into the jitter variance analysis, which accounts for the slight discrepancy. Note that at some specific values of α the jitter variance of the longer length sequence is worse than that of the shorter sequence. This can be explained by

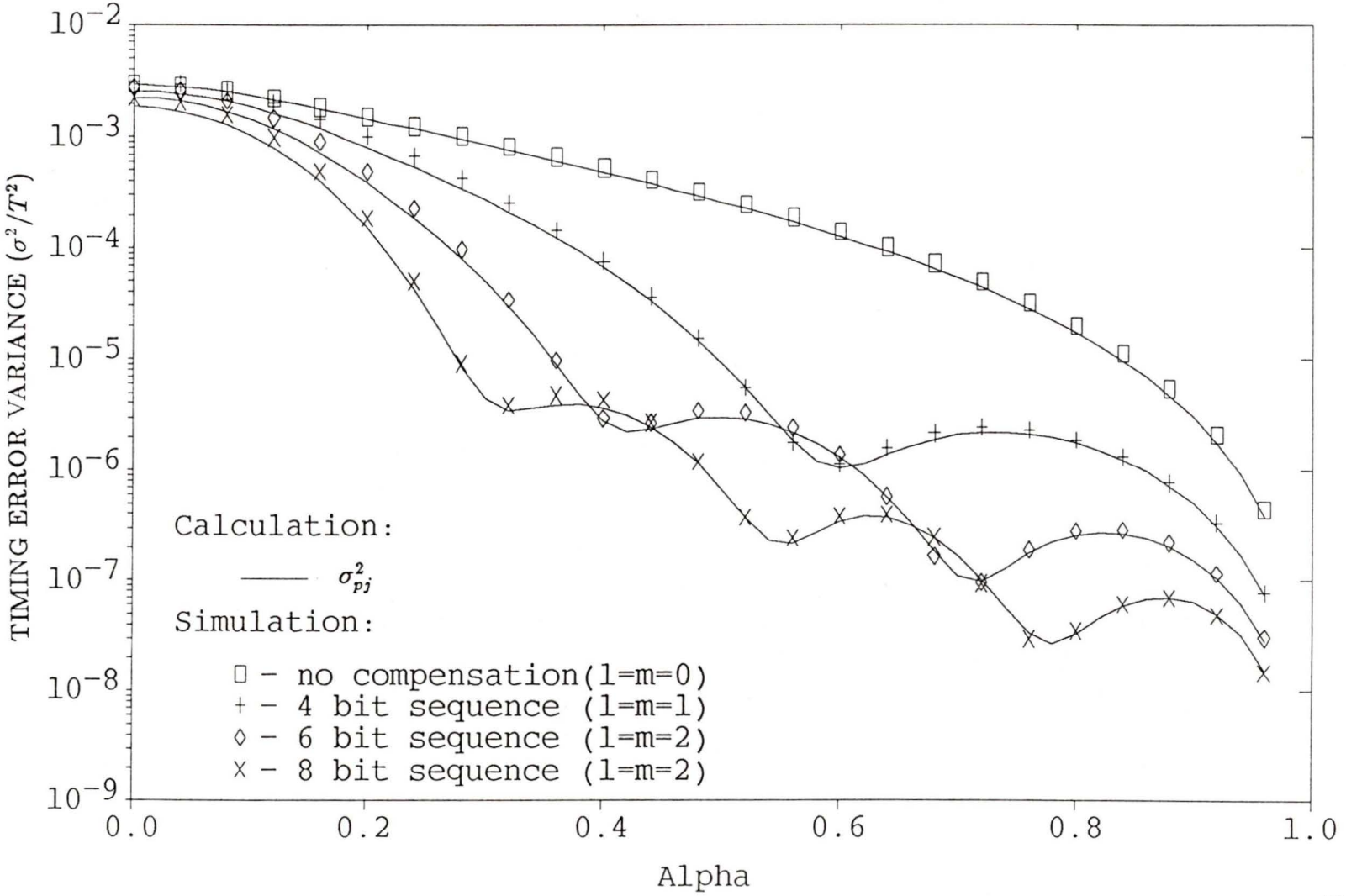


Figure 3.12: Linear simulation of the synchronizer pattern jitter variance versus alpha for 2BLT = 0.1.

recognizing the fact that at these values of α, the precursor and postcursor tails of the longer sequence pass through zero at $t = t_o - (m+1)T$, and $t = t_o + lT$ respectively. Therefore the longer sequence does not eliminate ISI, but performance will be degraded due to the extra delay m in the feedback loop of the PLL. Figure 3.13 illustrates σ_{pj}^2 at a loop bandwidth $2B_LT = .025$. As expected the jitter is reduced with the lower $2B_LT$, but the performance of the compensation mechanism is relatively unchanged. The agreement between analysis and calculation is improved at the lower bandwidth.

The performance of the compensation technique in the presence of additive noise is illustrated in figure 3.14. The value of α is 0.3 and $2B_LT = 0.10$. The simulation points are compared to the calculated timing error variance $\sigma_e^2 = \sigma_{pj}^2 + \sigma_{nj}^2$, where σ_{nj} and σ_{pj}^2 are determined by (3.75) and (3.47), respectively, and are also compared to the calculated σ_{nj}^2 alone. The simulation points are in close agreement with the analytical results. At low E_b/N_o the jitter variance σ_e^2 for different values of $l = m$ is asymptotic to σ_{nj}^2 for the corresponding feedback delay m. The jitter reduces as E_b/N_o increases as expected, but is lower bounded by the residual pattern jitter. For $E_b/N_o \leq 35dB$, the performance of the 8 bit compensation sequence synchronizer has a jitter variance only slightly larger than that of σ_{nj}^2. Therefore pattern jitter is effectively eliminated from the synchronizer output for all practical E_b/N_o. The penalty paid in term of noise jitter performance due to the m bit delay in the feedback path is illustrated by the difference between the $m = 0$ and $m = 3$ curves calculated for σ_{nj}^2.

For a lower loop bandwidth $2B_LT = .025$, (see figure 3.15) the simu-

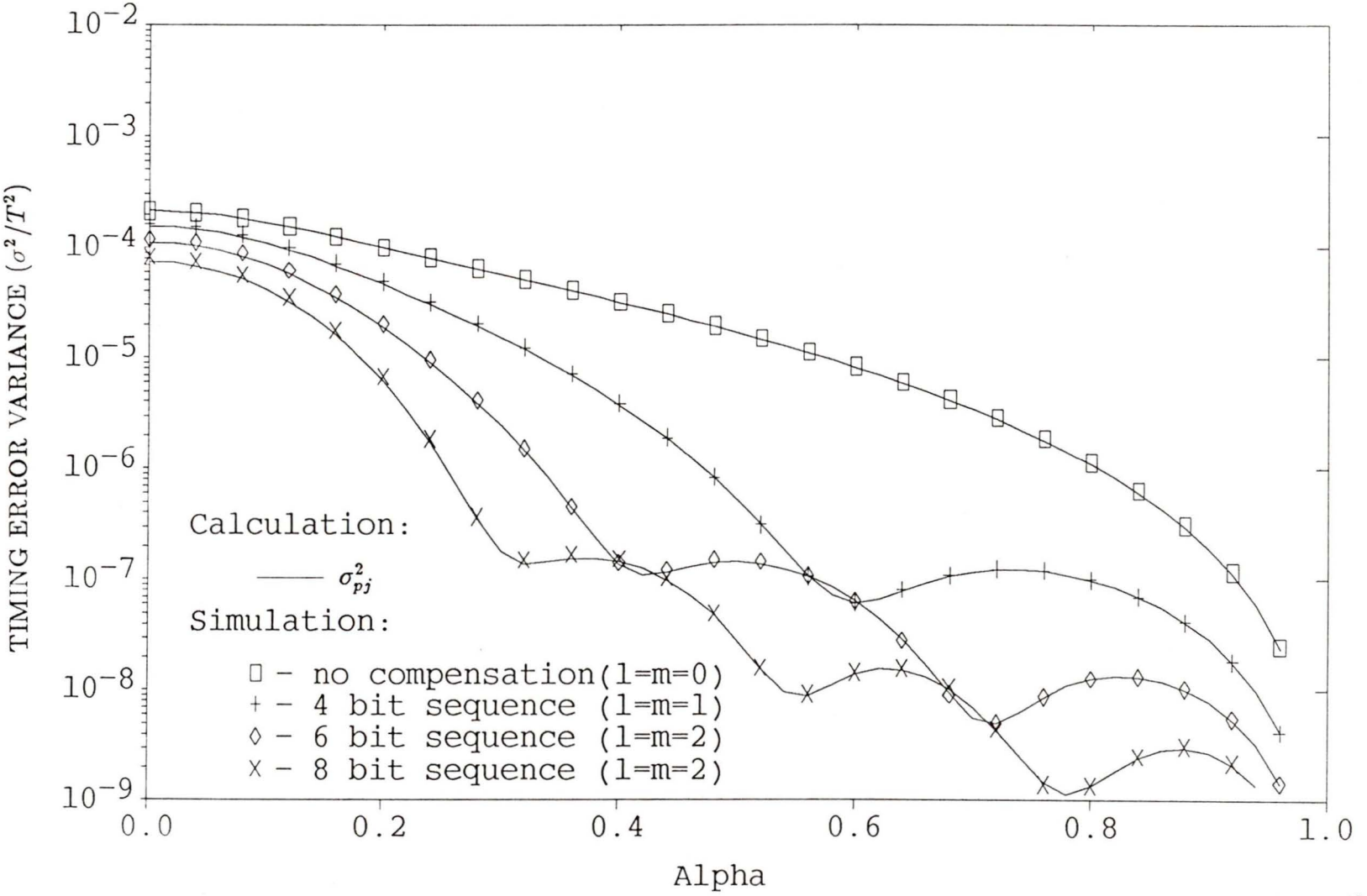


Figure 3.13: Linear simulation of the synchronizer pattern jitter variance versus alpha for 2BLT = 0.025.

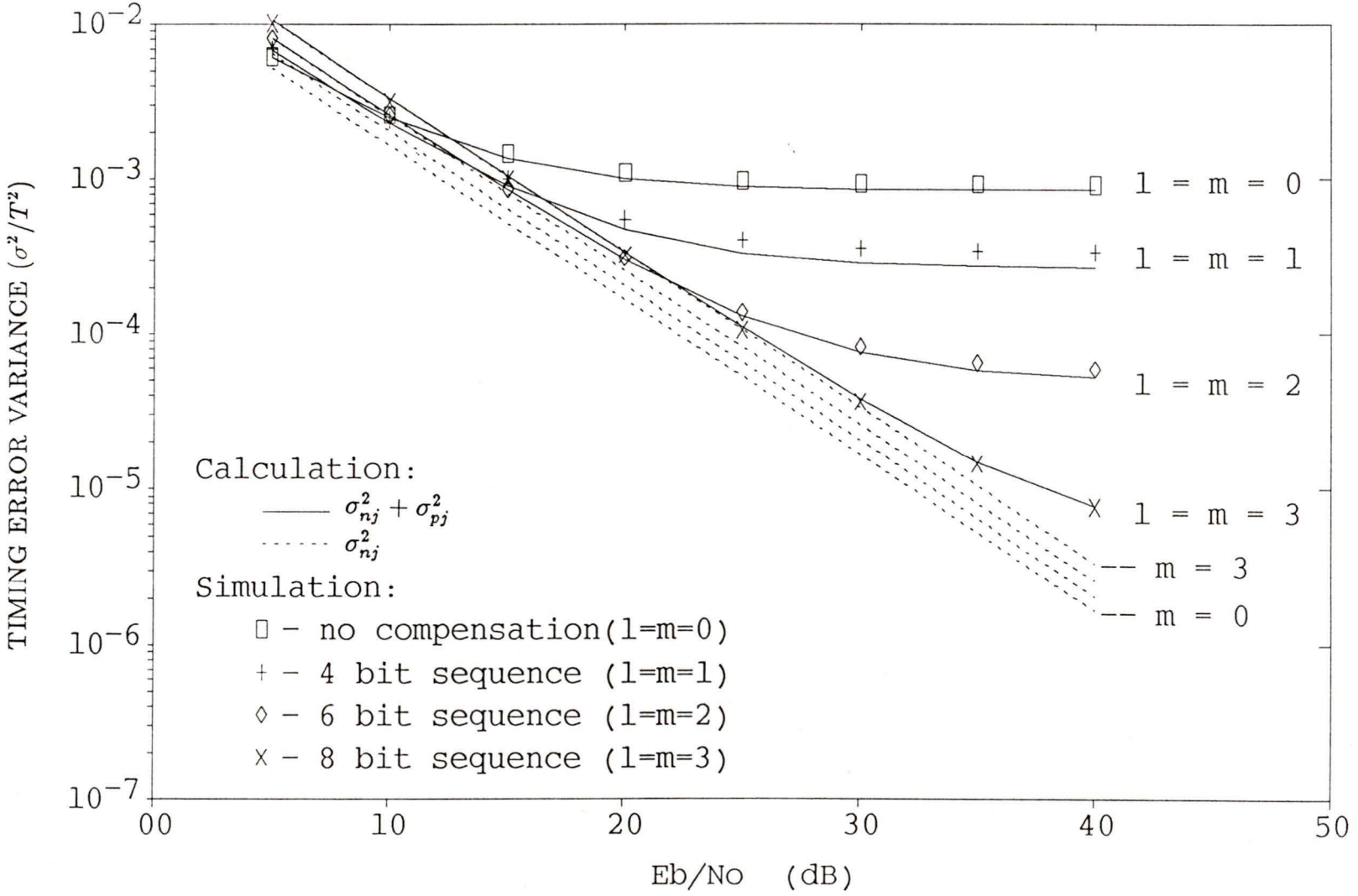


Figure 3.14: Linear simulation of the synchronizer jitter variance versus Eb/No with alpha = 0.3; 2BLT = 0.1.

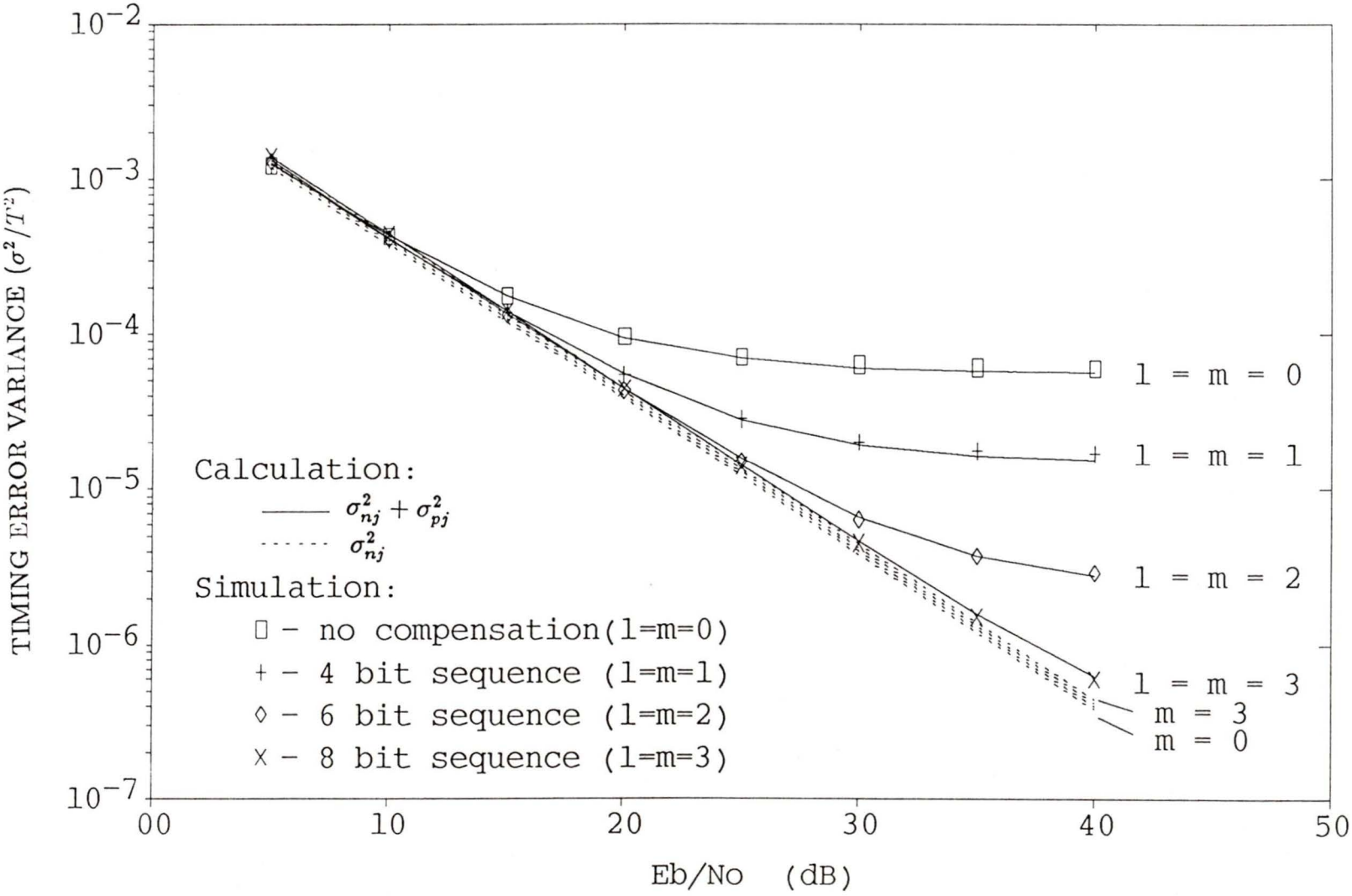


Figure 3.15: Linear simulation of the synchronizer jitter variance versus Eb/No with alpha = 0.3; 2BLT = 0.025.

lation points are in very close agreement with the calculated curves. As determined in appendix E, the noise jitter degradation due to the feedback delay is much less for the lower loop bandwidth, as can be seen by the closeness of the calculated σ_{nj}^2 curves for $m = 0 \rightarrow 3$.

For larger values of α the required sequence length to effectively eliminate pattern jitter will decrease. From figure 3.16, with $\alpha = 0.5$, a sequence length of only 6 bits ($l = m = 2$) gives performance nearly equivalent to the synchronizer operating in the absence of ISI. Again the simulation results are in close agreement with the analysis.

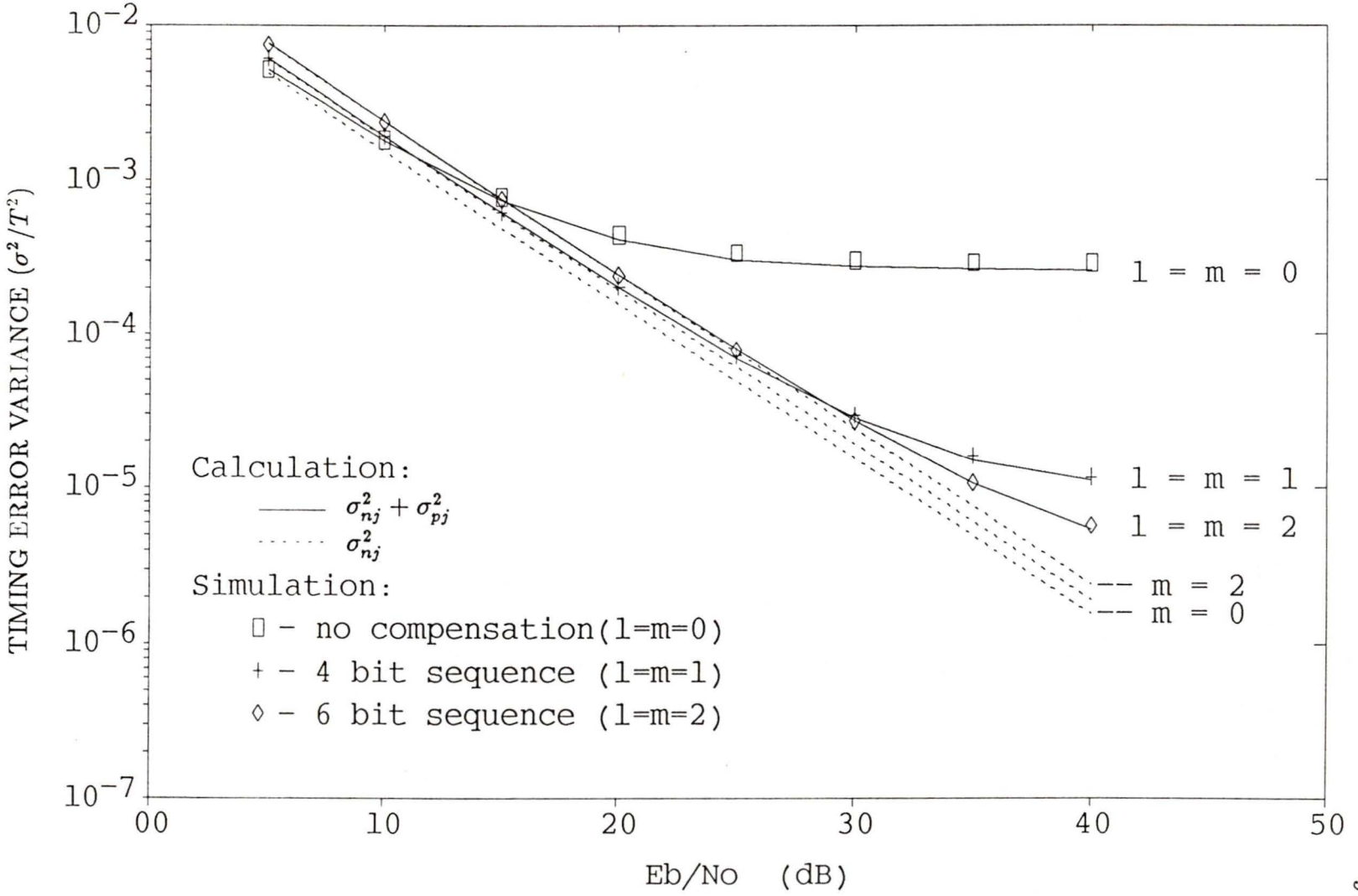


Figure 3.16: Linear simulation of the synchronizer jitter variance versus Eb/No with alpha = 0.5; 2BLT = 0.1.

Actual Simulation

The proposed synchronizer is now evaluated using actual SRC pulses, defined by (3.81), and a filtered noise waveform. The simulation method is described in appendix F.

Figure 3.17 compares the the simulated and calculated σ_{pj}^2 for different α. It is apparent that the actual simulation yields results very close to the linear analysis and the linear simulation (figure 3.12), confirming that for a SRC system, the linear assumptions made in the analysis are a reasonable approximation to the actual case. For very small values of α, there is some divergence between the linear and actual curves, where the ISI is too large for the linear analysis to apply.

The simulation results for the jitter variance of an $\alpha = 0.3$ and $\alpha = 0.5$ system with additive noise and ISI are plotted as a function of E_b/N_o in figures 3.18 and 3.19 respectively. The actual simulation results are in close agreement with both the calculated performance and with the linear simulation (figures 3.14 and 3.16). For the $\alpha = 0.3$ system the 8 bit compensation sequence has almost equivalent performance to the synchronizer operating in the absence of ISI, confirming the conclusion reached from the linear simulation (figure 3.14). Similarly the choice of a 6 bit compensation sequence for the $\alpha = 0.5$ system determined by the linear simulation (figure 3.16) is confirmed with the actual simulation. It should be noted that for values of $E_b/N_o < 5dB$ the actual synchronizer simulation results diverged from the linear model quite rapidly.

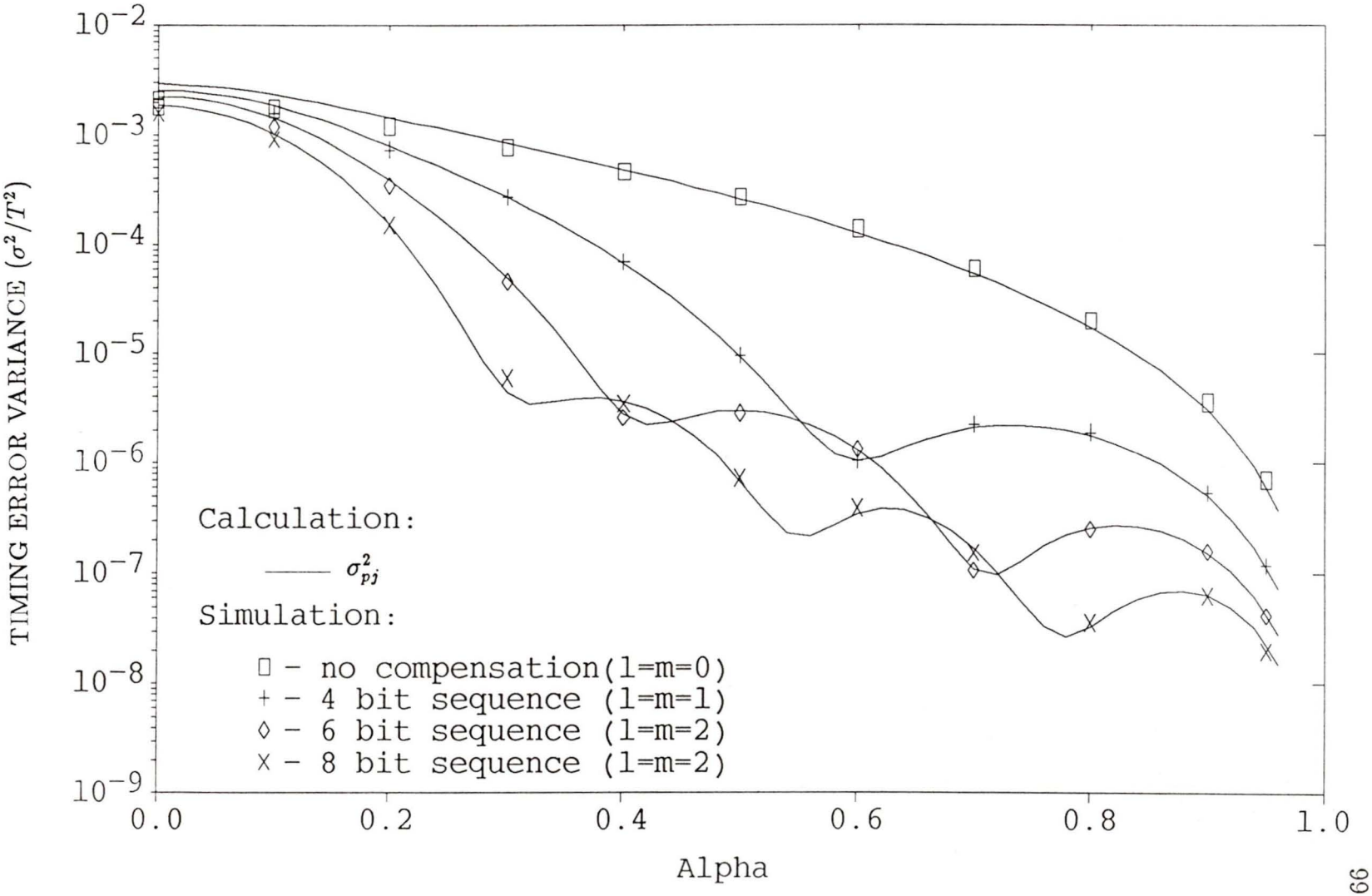


Figure 3.17: Actual simulation of the synchronizer pattern jitter variance versus alpha for 2BLT = 0.1.

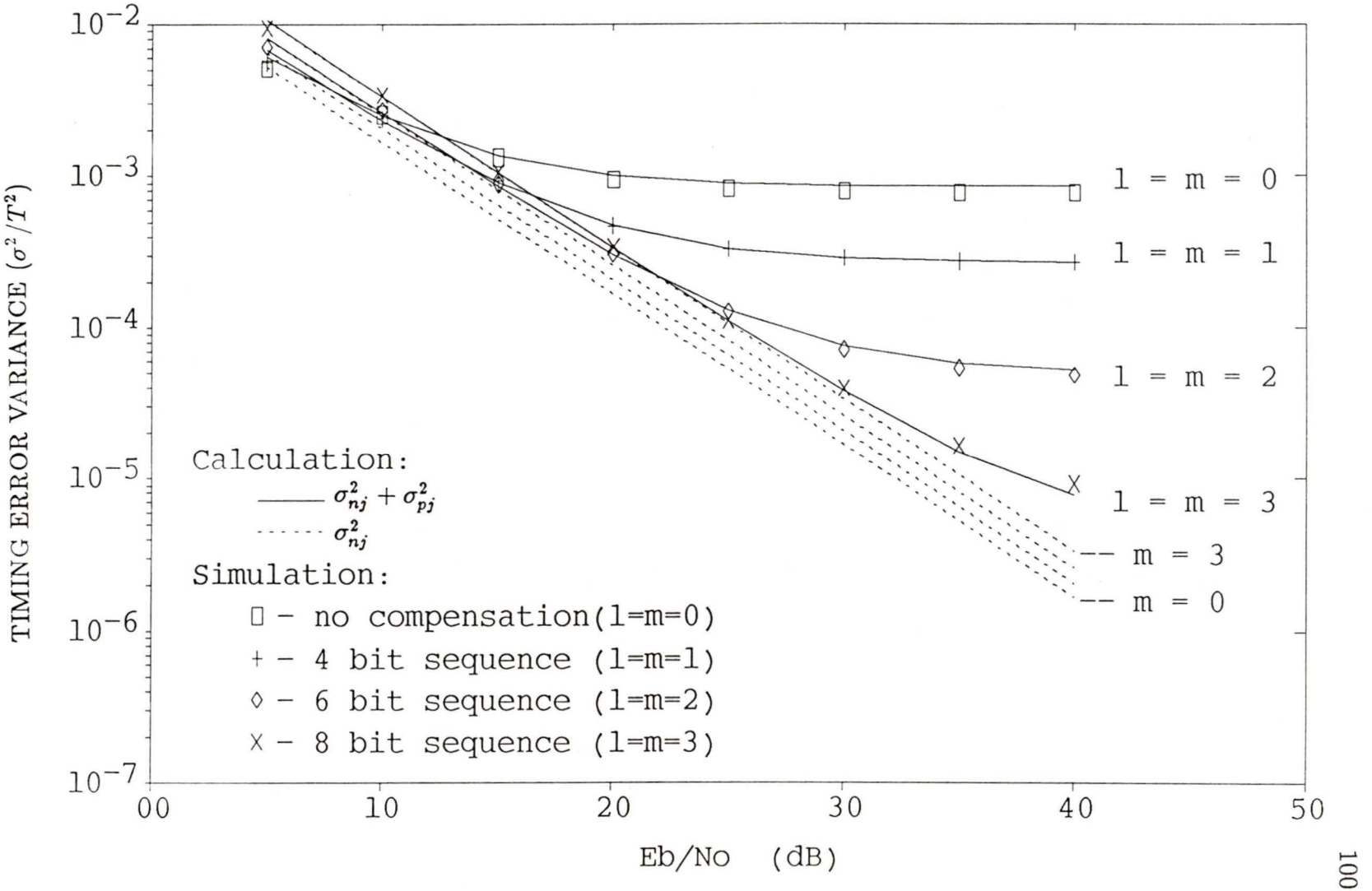


Figure 3.18: Actual simulation of the synchronizer jitter variance versus Eb/No with alpha = 0.3; 2BLT = 0.10.

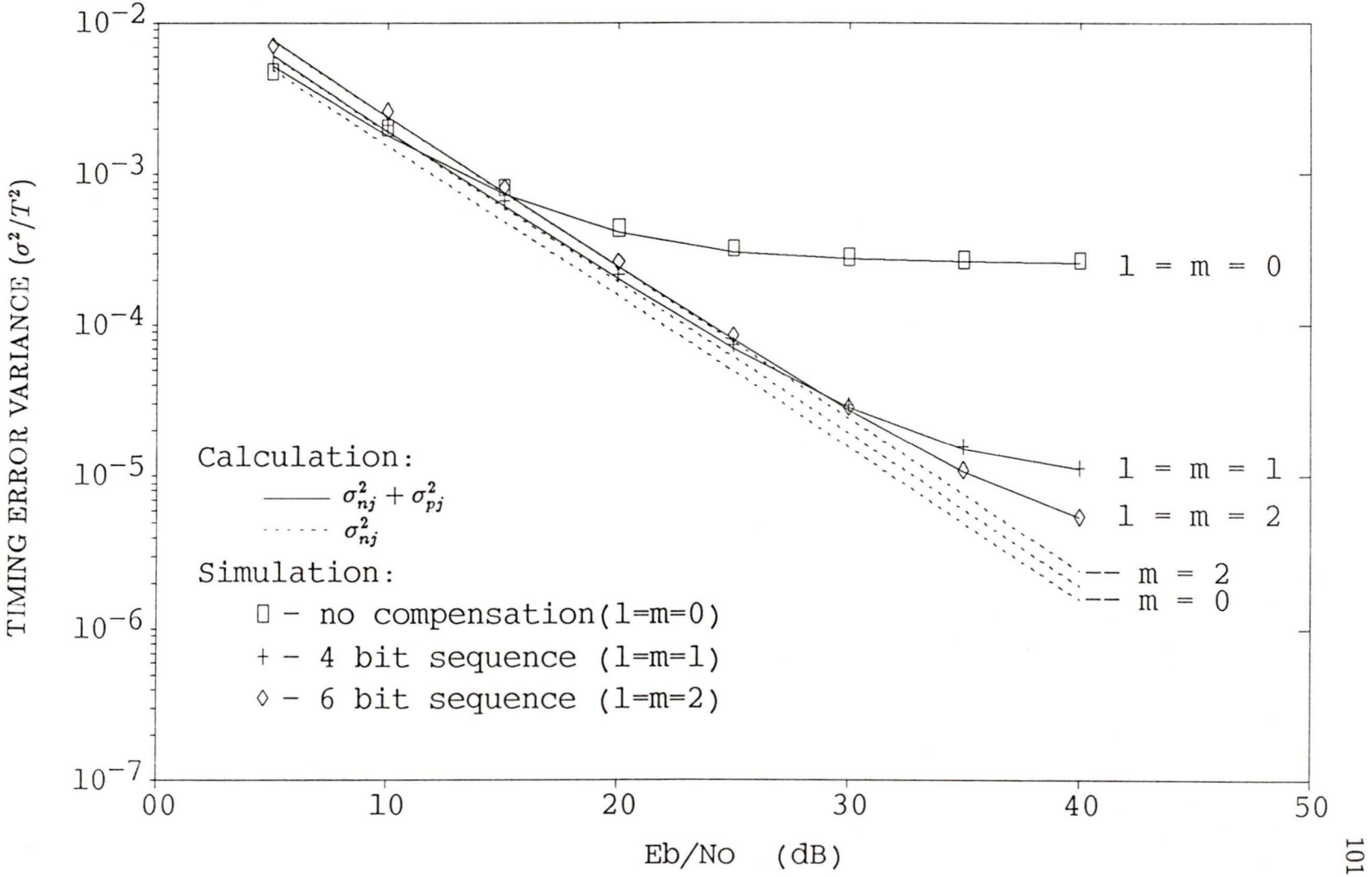


Figure 3.19: Actual simulation of the synchronizer jitter variance versus Eb/No with alpha = 0.5; 2BLT = 0.1.

3.7.2 Simulations with a Realizable Data Transmission Filter

The jitter variance of a synchronizer operating on signalling pulses generated by the seventh order transmission filter is evaluated in this section. The jitter variances σ_{nj}^2 and σ_{pj}^2 are determined by (3.75) and (3.47) respectively, and are comared with the linear simulation results. The linear simulation method is described in appendix F.

The calculated and simulated timing jitter variance $\sigma_e^2 = \sigma_{nj}^2 + \sigma_{pj}^2$ as a function of E_b/N_o is illustrated in figure 3.20. The bandwidth of the synchronizer is $2B_LT = 0.1$. Again the agreement between the calculated and simulated results is quite good. Also apparent is that a 9 bit compensation sequence ($l = 1, m = 6$) will effectively eliminate the pattern jitter for most practical E_b/N_o.

3.7.3 Adaptive Synchronizer Simulations

The adaptive synchronization technique described in section 3.6 is verified in this section using a linear simulation. Figure 3.21 illustrates the performance of an $\alpha = .3$ SRC system as a function of E_b/N_o. The stochastic gradient adaptation gain parameter $\beta = .001$. No analysis of the adaptation speed of convergence versus β is carried out. The simulation indicates a steady state timing error variance that is only slightly degraded relative to the performance of the synchronizer where $x(t)$ is known.

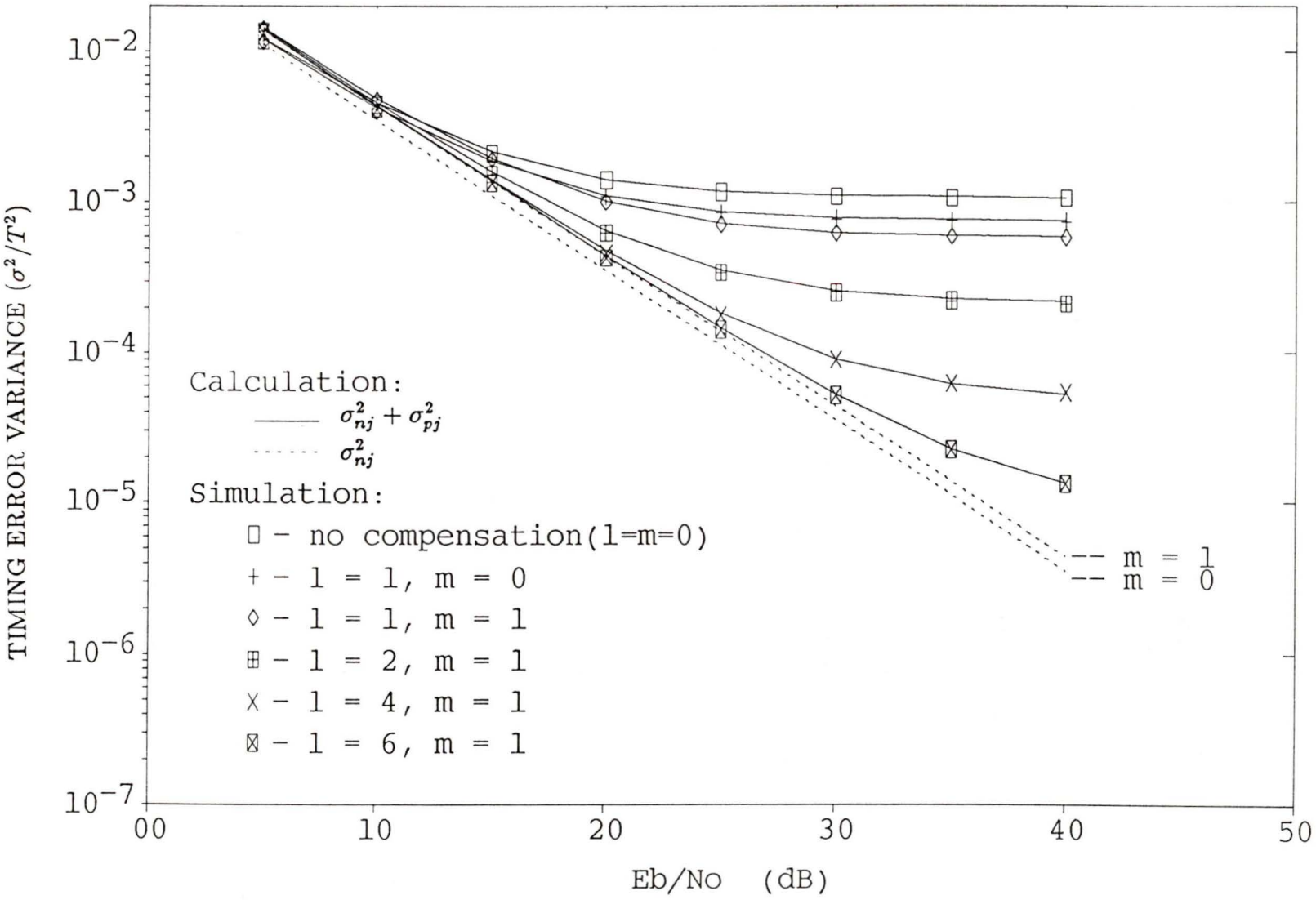


Figure 3.20: Linear simulation of the synchronizer jitter variance for the seventh order data transmission filter; 2BLT = 0.1.

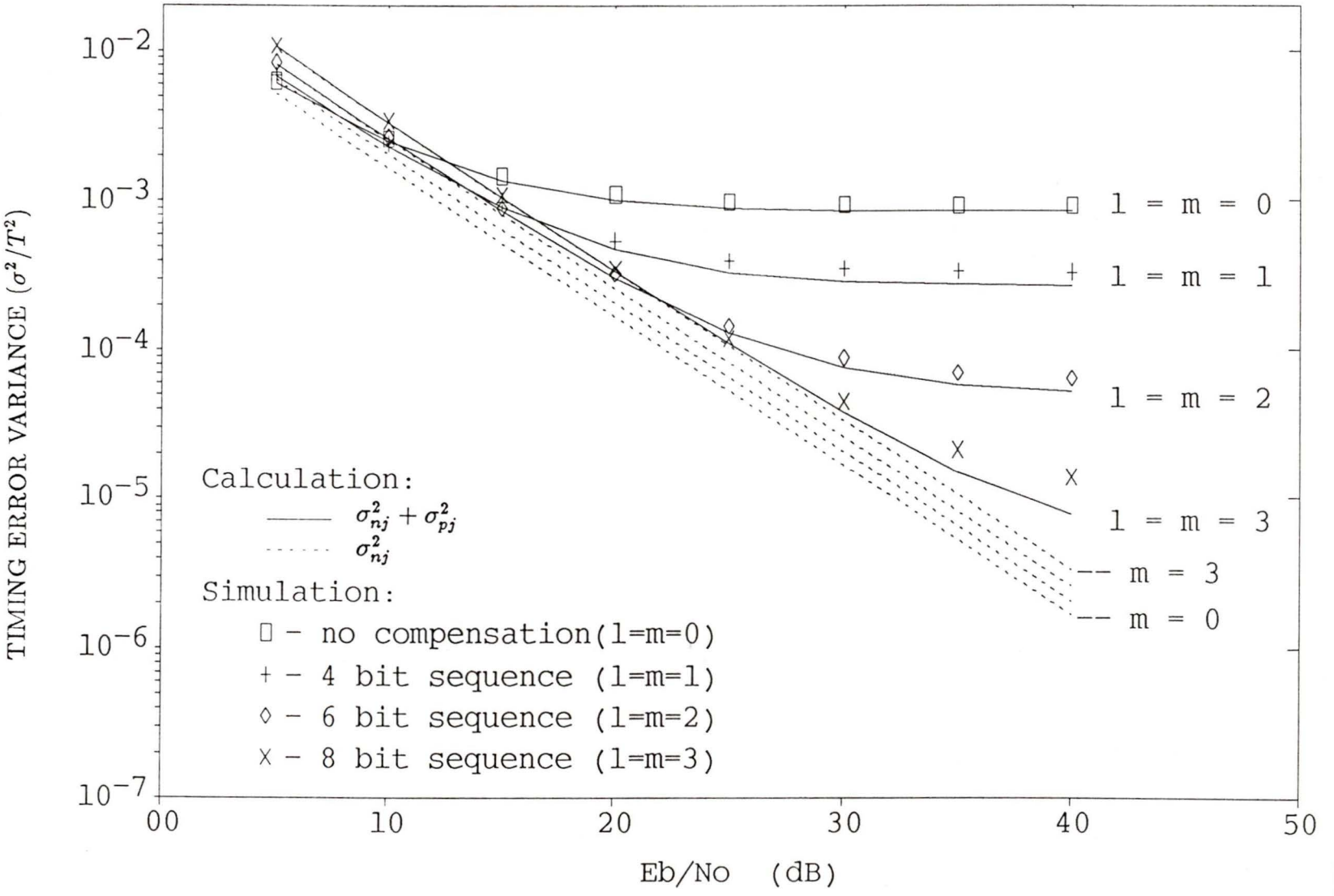


Figure 3.21: Linear simulation of an adaptive synchronizer jitter variance with alpha = 0.3; beta = .001; 2BLT = 0.1.

3.8 Implementation and Experimental Measurements

To complete the evaluation of the proposed synchronizer, experiment results of a pattern jitter compensation synchronizer are presented in this section.

The synchronizer is implemented using a TMS32020 digital signal processing integrated circuit. The digital synchronizer and the jitter variance measurement technique is described in appendix G.

The realizable transmission filter used in the simulations is not fabricated, and so we are unable to compare the simulation results to experimental data. Instead nonreturn to zero (NRZ) pulses are filtered by a readily available fourth order Butterworth filter with a 3 dB cutoff frequency $f_o = 1/2T$ to simulate the received eye pattern. The synchronizer jitter variance measurements are compared to the analysis results.

The eye pattern is illustrated in figure 3.22. Also shown in figure 3.22 are the data samples generated by the uncompensated synchronizer to illustrate the pattern jitter in the recovered timing signal. The compensation sequence values $\hat{\tau}_k$ are measured on an oscilloscope by observing the zero crossing locations of the eye pattern, and therefore the exact pulse shape $x(t)$ is not required. The pulse response had a negligible precursor, and hence no feedback delay is required ($m = 0$). For implementation reasons, a value of $K = 1/8$ is used. From (E.14) the bandwidth $2B_LT = .067$.

Figure 3.23 shows the measured jitter variance of an uncompensated synchronizer and a compensated synchronizer with a compensation sequence length 4, ($l = 2, m = 0$). The measured results are compared to the

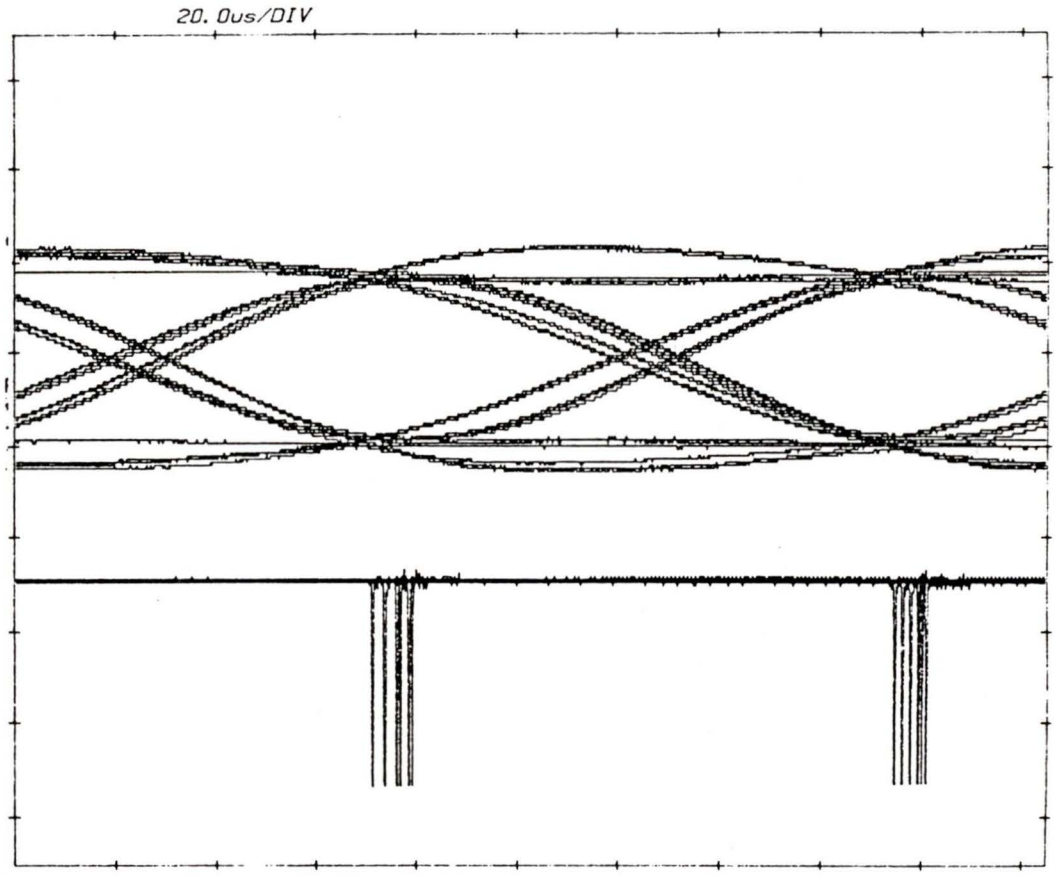


Figure 3.22: Bandlimited eye pattern and synchronizer data samples. $2B_LT = 0.067$.

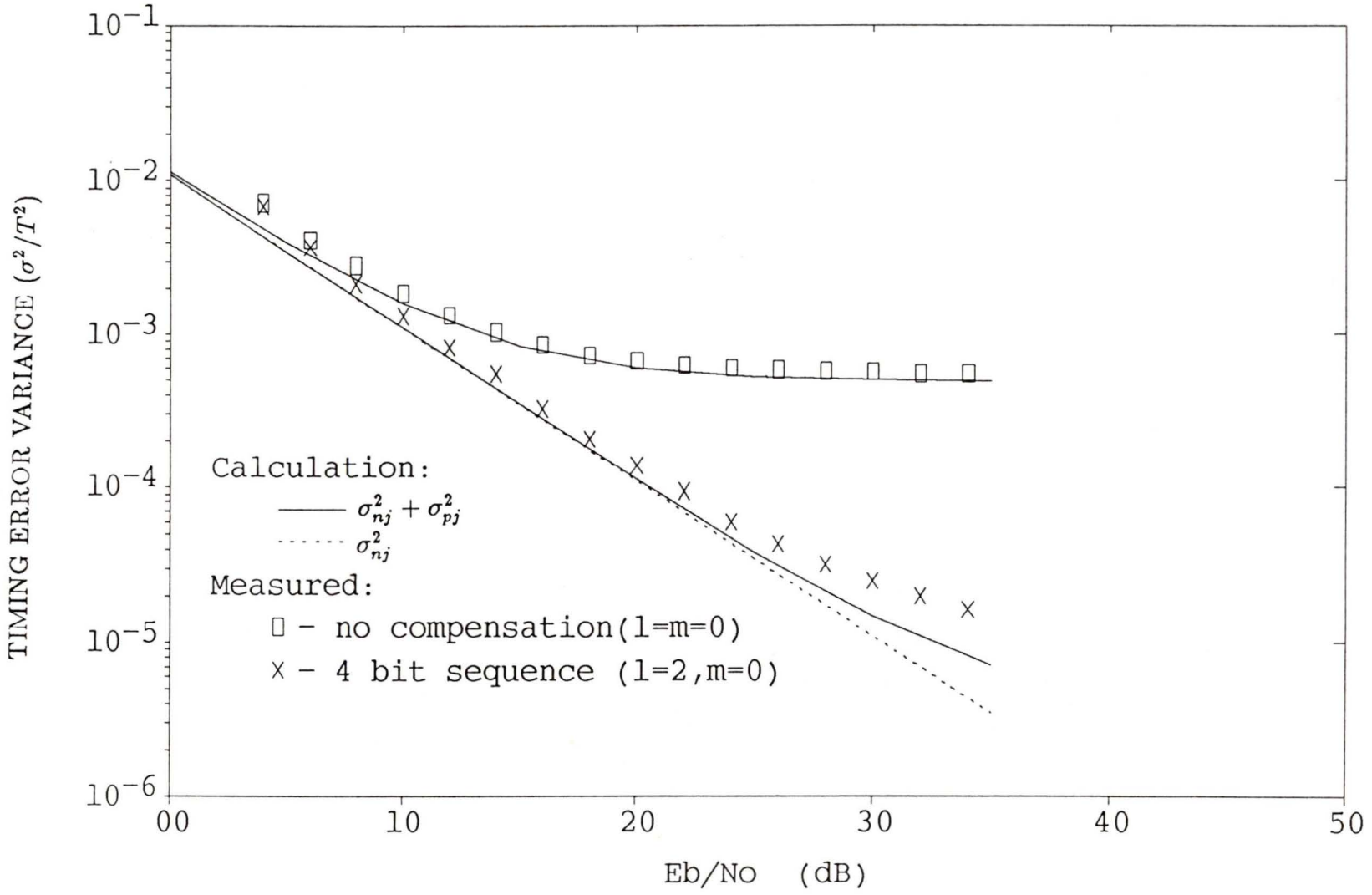


Figure 3.23: Experimental measurement of the synchronizer jitter variance versus Eb/No. 2BLT = 0.067.

calculated σ_{nj}^2 from (3.75) and σ_{pj}^2 from (3.47). As expected the pattern jitter is the lower bound on the uncompensated synchronizer jitter variance. The compensated synchronizer does not have a significant pattern jitter component over the range of E_b/N_o. The divergence of the calculated and measured values at low E_b/N_o can be attributed to the linear approximations in the analysis no longer being valid. The timing error variance measurement equipment is not capable of measuring normalized variances less than $\approx 10^{-5}$, which would account for the divergence at high E_b/N_o for the compensated synchronizer. Also slight inaccuracies in measuring $\hat{\tau}_k$ would also contribute to the divergence.

The recovered timing signals for the uncompensated and compensated synchronizers are compared in figure 3.24. The pattern jitter is readily apparent in the uncompensated synchronizer. The compensated synchronizer indicates only a very slight amount of pattern jitter.

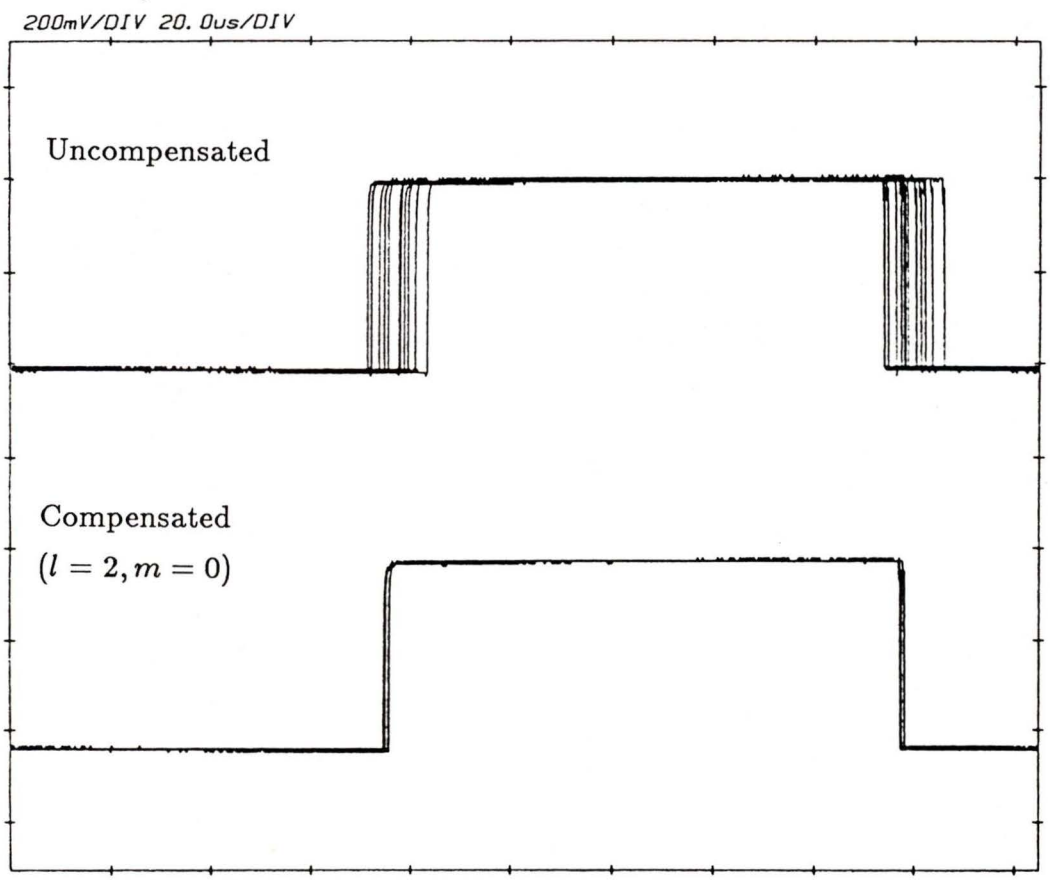


Figure 3.24: Timing waveforms of the uncompensated and compensated ($l = 2, m = 0$) synchronizers for $E_b/N_o = 40$ dB.

3.9 Summary

The pattern jitter component of a zero crossing tracking PLL synchronizer may be reduced well below the noise jitter component by using a short sequence of received data bits to predict the effect of ISI on the zero crossing. Pattern jitter compensation can be based on a known impulse response, measured zero crossing locations, or in an adaptive implementation. The number of bits in the compensation sequence depends on the rate at which the system impulse response decays, which is generally determined by the operating bandwidth of the system.

Three methods are presented to determine the timing jitter variance performance of the synchronizer. First, analytical equations based on linear approximations are derived. The analysis more correct than a previous result [18], and it also includes the pattern jitter compensation. The new result is exact for white noise disturbances, and simulation results indicate that it is more accurate than [18] for non white disturbances, specifically ISI. Second, the linear assumptions made in the analysis can be used to generate a recursive timing equation which can be evaluated using Monte Carlo techniques to determine the jitter variances. The third technique is to directly simulate the synchronizer by generating the required waveforms, and recovering the timing information. Any of the above methods can be used by designers to estimate the compensation sequence lengths to reduce the pattern jitter below a required level.

For operation with SRC signalling pulses, both the linear and actual simulation results are in close agreement with the analytical results. The

analysis is also effective in predicting the jitter variance of a synchronizer operating on pulse waveforms generated by a seventh order transmission filter.

The simulation results and analysis also verified the effectiveness of the proposed pattern jitter compensation technique. The results demonstrate that the technique can be used on a SRC signalling pulses to greatly reduce the pattern jitter component of the synchronizer. For a small excess bandwidth $\alpha = 0.3$, the pattern jitter is essentially eliminated for $E_b/N_o < 35$ dB using an 8 bit compensation sequence. An excess bandwidth of $\alpha = 0.5$ requires only a 6 bit sequence.

The seventh order data transmission filter required a sequence length of 9 bits, with only one bit feedback delay ($m = 1$). The performance of the synchronizer on the realizable pulse shape indicates that the synchronizer may find application in low cost digital synchronizer implementations for a bandlimited binary data communications system, where self noise in the synchronizer is expected to be a concern.

For applications where the impulse response is not accurately known, the adaptive pattern jitter compensation technique may be appropriate. Also the zero crossings may be measured and fixed compensation sequence values used.

Laboratory measurements of a pattern jitter compensation synchronizer indicate that for NRZ pulses filtered by a fourth order Butterworth filter, pattern jitter is essentially eliminated using a four bit compensation sequence with no feedback delay.

The performance of the compensation technique in the presence of a

frequency offset has not been addressed in the analysis or simulations, and is an area for further study.

Chapter 4

Adaptive Bandwidth Phase Locked Loop Synchronizer

4.1 Foreword

A digital PLL synchronizer for time division multiple access (TDMA) and carrier sense multiple access(CSMA) applications is proposed. The PLL phase detector (PD) input is a hard limited binary data signal. A simple adaptive bandwidth control monitors the output of the PD and moves the synchronizer to a large bandwidth when rapid acquisition is required. When acquisition has been achieved, the synchronizer moves to a lower bandwidth to improve the tracking performance. The operation of the synchronizer is verified for typical PLL parameters with both computer simulation results and hardware experimental measurements.

4.2 Introduction

In this chapter we consider a phase locked loop (PLL) synchronizer with an adaptive bandwidth control for application in a carrier sense multiple access (CSMA) digital radio system.

In TDMA and CSMA packet radio systems multiple users share a common channel, and transmit information contained within data packets. An important system parameter is the portion of the data packet assigned for synchronization. The delay associated with the initial synchronization will reduce the maximum data throughput of the system [33]. The minimum required preamble duration is determined by the acquisition performance of the data receiver's synchronizer.

A synchronizer for these applications must have rapid acquisition characteristics. Unfortunately rapid acquisition generally implies poor tracking and cycle slip performance. What is presented in this chapter is a simple hard limited zero crossing tracking PLL synchronizer with an adaptive bandwidth control. The bandwidth control initially forces the synchronizer into a wide bandwidth (acquisition) mode. When synchronization has been achieved the bandwidth of the synchronizer is progressively lowered to improve the tracking of the synchronizer (tracking mode). The narrow bandwidth may allow the synchronizer to remain locked through rapid fades, as is required in mobile radio applications [52].

The adaptive bandwidth control requires only a small amount of hardware and for microprocessor implementations can be completely incorporated into software.

The evaluation of the proposed synchronizer consists of a computer simulation and laboratory experiment results. The synchronizer parameters are not optimized to any particular application. The investigation is confined to measuring the operating performance of the synchronizer for typical PLL parameters over a reasonable range of signal to noise ratios (SNR).

This chapter is organized as follows: section 4.3 is a description of the proposed synchronizer. The description and analysis of the adaptive loop gain control is presented in section 4.4. Computer simulation results for some typical applications are contained in section 4.5. In section 4.6, laboratory experiments verify the synchronizer operation. The results of this chapter are summarized in section 4.7.

4.3 Synchronizer Description

The proposed synchronizer is illustrated in figure 4.1. The structure is a first order digital PLL. The received signal $y(t)$ at the low pass filter (LPF_R) output is

$$y(t) = \sum_i a_k x(t - kT - \epsilon) + \eta(t) \tag{4.1}$$

where $a_k \in \{-1, +1\}$ are zero mean binary data, $x(t)$ is the signalling pulse shape, ϵ is the slowly varying delay of the received waveform, and $\eta(t)$ is filtered additive white Gaussian noise (AWGN). It is assumed that the excess bandwidth of the LPF_R (relative to the Nyquist limit) is sufficient to ensure negligible intersymbol interference (ISI) at the zero crossings [39].

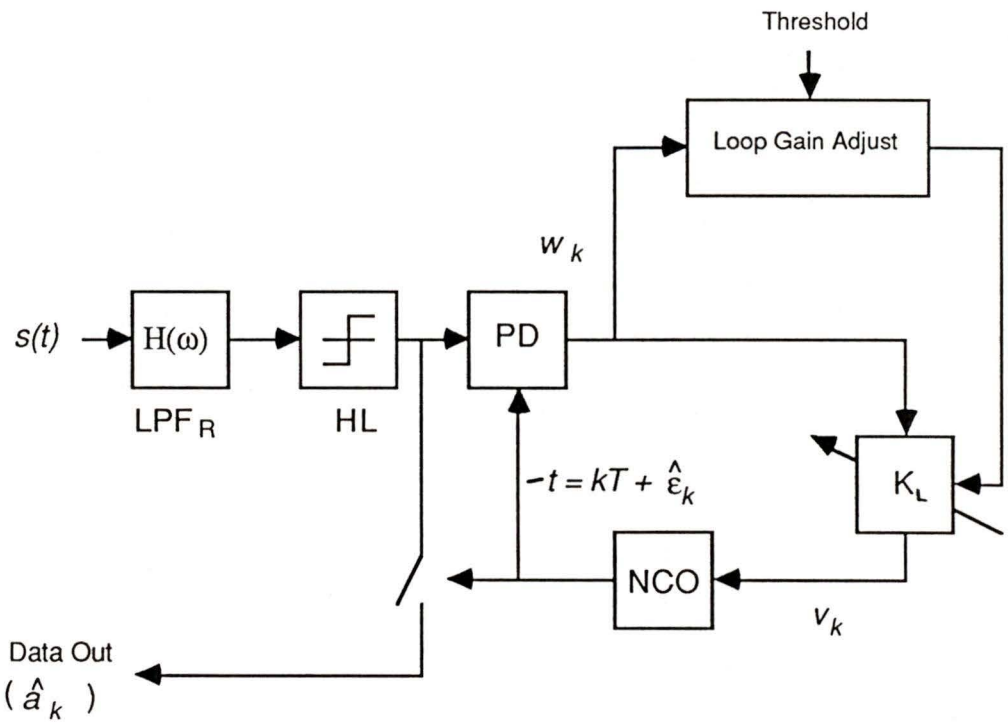


Figure 4.1: Block diagram of the adaptive bandwidth synchronizer.

The waveform $y(t)$ is hard limited, and the data decisions $\hat{a}_k$ are made by sampling at $t = kT + \hat{\epsilon}_k$, where $\hat{\epsilon}_k$ is the synchronizer's estimate of ϵ_k. The synchronizer timing error e_k is

$$e_k = \epsilon_k - \hat{\epsilon}_k, \qquad e_k \in (-T/2, T/2). \tag{4.2}$$

The phase detector (PD) generates an estimate of the phase error of the incoming signal relative to the local timing reference, generated by the numerically controlled oscillator (NCO). For this application we consider a binary counter PD implementation. The operation of this PD is discussed in appendix G and a description of counter phase detectors is given in [48]. In the absence of noise the PD characteristic, defined as $E[w_k|e_k]$, is sawtooth shaped

$$E[w_k|e_k] = \begin{cases} 2b_k K_{PD} e_k & e_k \in (-T/2, T/2) \\ 0 & e_k \notin (-T/2, T/2) \end{cases} \tag{4.3}$$

where K_{PD} is the PD gain constant, and $2b_k$ is the transition detector output

$$2b_k = 1 - \hat{a}_k \hat{a}_{k+1} \tag{4.4}$$

The transition detector ensures no timing error adjustments are made when no transition is detected (i.e. $\hat{a}_k = \hat{a}_{k+1}$).

A disadvantage of PLL synchronizers for CSMA applications, is a phenomenon called "hangup". A PLL occasionally will dwell or hangup at an unstable equilibrium for a long duration before proper acquisition is achieved [45]. PLL's with analog multiplier phase detectors are known to be significantly hampered by hangup problems [45], and as a result some synchronizer designers have adopted the nonlinearity and bandpass filter

synchronization approach to avoid PLL hangup [53]. However for the saw-tooth PD characteristic with discontinuities at $e_k = \pm T/2$, it is known that PLL hangup is not as significant a problem, although it is still a concern [54].

The PD output depends critically on the relative zero crossing locations of the input signal and the local (NCO) clock. For a reasonable SNR and bandlimited noise, in a bit interval where a data transition is present (i.e. $a_k \neq a_{k+1}$) one zero crossing is expected. However if multiple zero crossings occur (for example at low SNR and/or a wide bandwidth noise disturbance) the counter PD will calculate the mean zero crossing location and output the difference between mean zero crossing location and the local (NCO) estimate.

The PD output w_k is scaled by the feedback gain parameter K_L, the output of which is used to control the NCO. The value of K_L is determined by the adaptive loop gain control. The NCO has an internal gain parameter K_{NCO}. Timing adjustments are made according to

$$\hat{\epsilon}_{k+1} = \hat{\epsilon}_k + K w_k \tag{4.5}$$

where we define the loop gain constant $K \triangleq K_L K_{NCO}$, which we will use to control the bandwidth. For an adaptive synchronizer: $K = K_k$. The PD gain is normalized to be $K_{PD} = 1$. Therefore the range

$$w_k \in (-T, T) \tag{4.6}$$

follows from (4.3).

4.4 Analysis of the Adaptive Loop Bandwidth Control

The objective of the loop gain control is to maintain the loop in a wide bandwidth state when no signal is present at the input (i.e. only hard limited noise). When signal is present the synchronizer should move toward a narrow bandwidth after synchronization is achieved. A simple method to control the synchronizer bandwidth is to monitor the variation of zero crossings, measured by the PD output w_k. With only noise present at the input, the zero crossing locations will have a large variation. When signal is present with a reasonable SNR the zero crossings will cluster about regularly spaced T second intervals. Therefore an appropriate bandwidth control technique is to adjust K_k according to the history of the consistency of zero crossings recorded at the PD output.

The technique chosen to modify K_k is illustrated in figure 4.2, where the progression of K_k is modeled as a Markov chain [56]. K_k is selected from N values

$$K_k \in \{K^{(i)} : 1 \leq i \leq N\} \tag{4.7}$$

The $K^{(i)}$ values are a nonincreasing function of i. The largest value, $K^{(1)}$, is determined by stability considerations and by the desired transient response. The minimum value, $K^{(N)}$, is determined by the lowest required bandwidth. The state transition probabilities $P_{i|j}$ are determined by the SNR, and by an external threshold parameter T_e, where

$$T_e \in (0, T) \tag{4.8}$$

The state transitions may be described as follows: assume $K_k = K^{(i)}$;

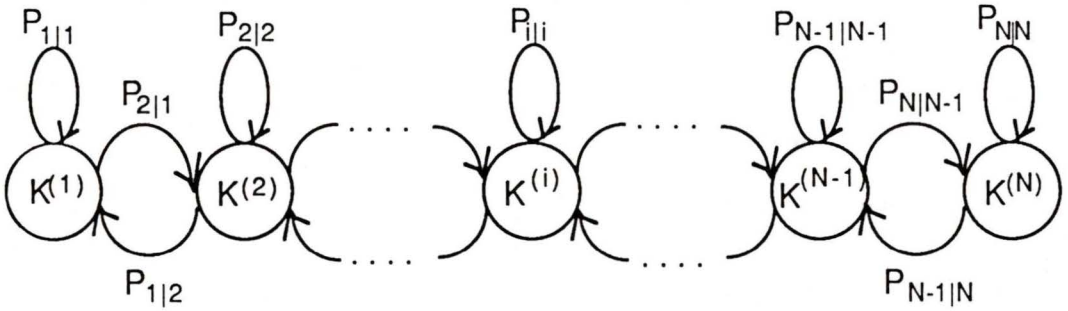


Figure 4.2: Markov chain model of the adaptive bandwidth control.

choose K_{k+1} according to

$$K_{k+1} = \begin{cases} K^i & \text{if } b_k = 0 & \text{(no data transition)} \\ K^{(i+1)} & \text{if } b_k = 1,\ |w_k| < T_e,\ i \neq N & \text{(decrease the bandwidth)} \\ K^{(i-1)} & \text{if } b_k = 1,\ |w_k| \geq T_e,\ i \neq 1 & \text{(increase the bandwidth)} \\ K^N & \text{if } b_k = 1,\ |w_k| < T_e,\ i = N & \text{(minimum bandwidth state)} \\ K^1 & \text{if } b_k = 1,\ |w_k| \geq T_e,\ i = 1 & \text{(maximum bandwidth state)} \end{cases} \tag{4.9}$$

Therefore the position in the chain, and thus the overall bandwidth, is determined by the previous history of w_k, the PD measured timing errors.

The state transition probabilities are:

$$P_{i|j} = \begin{cases} P_o + P_+ & i = j = 1 \\ P_o & i = j \neq 1, N \\ P_+ & i = j - 1 \\ P_- & i = j + 1 \\ P_o + P_- & i = j = N \\ 0 & \text{elsewhere} \end{cases} \tag{4.10}$$

where

$$P_o = Pr[b_k = 0] \tag{4.11}$$

$$P_+ = Pr[b_k = 1 \cap |w_k| \geq T_e] \tag{4.12}$$

$$P_- = Pr[b_k = 1 \cap |w_k| < T_e] \tag{4.13}$$

Let Q_i be the steady state probability that the Markov chain is in the i^{th} state. For independent w_k values

$$Q_i = \sum_{j=1}^{N} P_{i|j} Q_j \tag{4.14}$$

The independence assumption may or may not apply, depending on the properties of the noise and the signal. In chapter 3, the PLL analysis is carried out with the assumption that zero crossing events recorded by the zero crossing PD are independent from bit to bit, and some justification based on linear approximations is presented in appendix C. These assumptions are adopted for this analysis, but caution must be used when interpreting the results. Further investigation may be required to identify where the assumptions might fail.

An iterative solution to (4.14) is

$$Q_i = \begin{cases} \frac{(1-P_o-P_+)Q_1}{P_B} & i = 2 \\ \frac{(1-P_o)Q_{i-1}-P_-Q_{i-2}}{P_+} & i = 3, \ldots, N \end{cases} \tag{4.15}$$

A procedure to find Q_i is to assume $\tilde{Q}_1 = 1$, find $\tilde{Q}_i$ for $i = 2, \ldots, N$ using (4.15), and normalize the results such that $\sum_{i=1}^{N} Q_i = 1$

$$Q_i = \frac{\tilde{Q}_i}{\sum_{j=1}^{N} \tilde{Q}_j} \tag{4.16}$$

The flexibility in the design technique allows the engineer to tailor the operation of the loop to specific applications. For example, the threshold parameter T_e largely determines the SNR for which the synchronizer bandwidth begins to decrease. The length of the chain N and the $K^{(i)}$ values are important in determining the acquisition performance of the synchronizer. A particular value of $K^{(i)}$, for example the smallest value, may be used as a carrier detect signal. When this value of $K^{(i)}$ is reached the designer may

wish to force the loop to remain in the low bandwidth (tracking mode). An external system input may be used to force the synchronizer out of the low bandwidth condition, for example when an end of packet flag is decoded, or if the SNR indicates a deep fade that greatly exceeds the error correcting capability of the receiver.

In the following we present computer simulations and hardware measurements for typical design examples to verify the adaptive synchronization technique.

4.5 Simulation Results

For the computer simulations, a spectral raised cosine (SRC) matched filter system with rolloff factor $\alpha = 1$ is used. This filter ensures zero crossings at $t = kT + T/2 + \epsilon_k$ and zero intersymbol interference (ISI) at zero crossings in the absence of noise [39]. The simulation program generates 32 samples per bit. The simulation program also uses linear interpolation in determining the zero crossing location to increase the resolution of the phase detector beyond $\pm T/32$ for measuring very small timing error variances. The simulation method is described in more detail in appendix H.

The analysis for the timing jitter variance performance is the same as the analysis in chapter 3, section 3.5.6. The bandwidth $2B_LT$ of the first order loop from (E.14) is

$$2B_LT = \frac{K_k}{2 - K_k} \tag{4.17}$$

Although the Markov chain elements $K^{(i)}$ may assume any value, for im-

plementation purposes we consider the following values

$$K^{(i)} = \begin{cases} \frac{1}{8} & 1 \leq i \leq L \\ \frac{1}{8}\left(\frac{1}{2}\right)^{i-L} & L+1 \leq i \leq L+3 \\ \frac{1}{128} & L+4 \leq i \leq N \end{cases} \tag{4.18}$$

Three modes of synchronizer operation can be identified from (4.18)

1. For $1 \leq i \leq L$ the synchronizer operates at its maximum bandwidth (acquisition mode).

2. For $L+1 \leq i \leq L+3$ the synchronizer is in transition from maximum bandwidth to minimum bandwidth.

3. For $L+4 \leq i \leq N$ the synchronizer operates at its minimum bandwidth (tracking mode).

The maximum value $\{K_k\}_{max} = 1/8$ is the largest value that was measured to give a reasonably reliable and rapid acquisition performance at low SNR. The minimum value $\{K_k\}_{min} = 1/128$ is the smallest value that we could use and still reliably measure phase jitter variances in the laboratory. From (4.17), the maximum and minimum bandwidth values are

$$2B_L T_{max} = .067 \tag{4.19}$$

$$2B_L T_{min} = .0039 \tag{4.20}$$

From (4.15) the determination of the steady state probability distribution Q_i requires the event probabilities P_o, P_+, and P_-. If the input to the hard limiter is bandlimited noise (SRC filter, $\alpha = 1$), the measured distribution of w_k (figure 4.3) indicates two reasonable assumptions: first the output

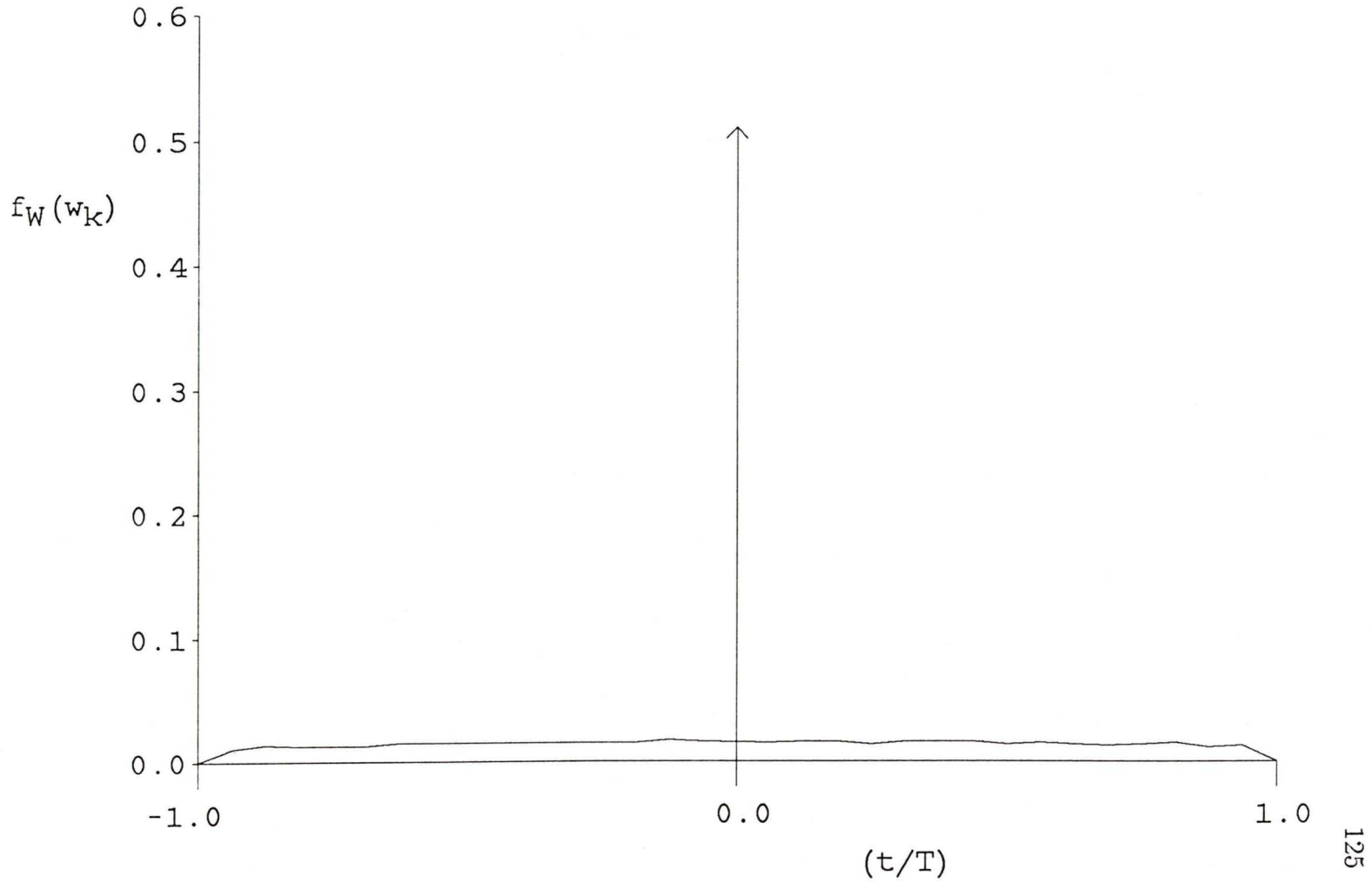


Figure 4.3: Probability density of the simulation program PD output w_k. The PD input is hard limited filtered AWGN.

of the phase detector w_k is uniformly distributed over its range $(-T, T)$, and second the transition detector probability is $Pr[b_k = 0] = Pr[b_k = 1] = 1/2$. The required probabilities follow from these two assumptions, and (4.11)-(4.13)

$$P_o = 1/2 \tag{4.21}$$

$$P_+ = (1/2)(1 - T_e/T) \tag{4.22}$$

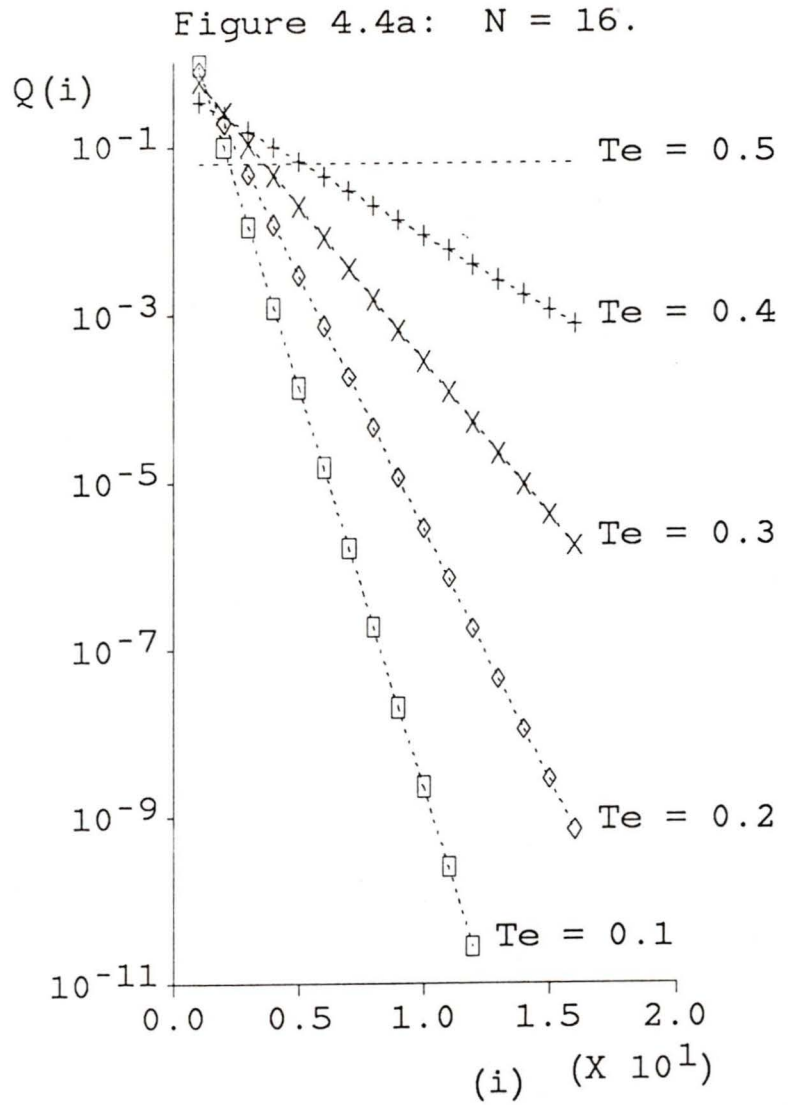
$$P_- = (1/2)(T_e/T) \tag{4.23}$$

$$\tag{4.24}$$

The steady state distribution Q_i is illustrated in figure 4.4 for $N = 16$ and $N = 32$. As expected, as T_e is increased the tendency is for the loop to move toward the lower bandwidth states. However for all $T_e \leq 0.3T$ the probability of being in a large bandwidth state is quite high, and the probability of moving to the minimum bandwidth state is low ($\leq 10^{-5}$).

The first objective of the simulation is to observe the tracking and acquisition performance of the fixed bandwidth synchronizer for both the maximum and minimum bandwidth values. The adaptive synchronizer simulation results will be compared to the fixed synchronizer results to determine how well the adaptive synchronizer achieves the acquisition performance of the maximum bandwidth synchronizer without sacrificing the tracking performance of the minimum bandwidth synchronizer.

To compare the acquisition performance of the synchronizers we measure the probability distribution of acquisition time, $Pr[T_{acq} \leq t]$, where the acquisition time T_{acq} is defined to be the time from when the data signal is initially received until the timing error e_k is reduced to within the smallest



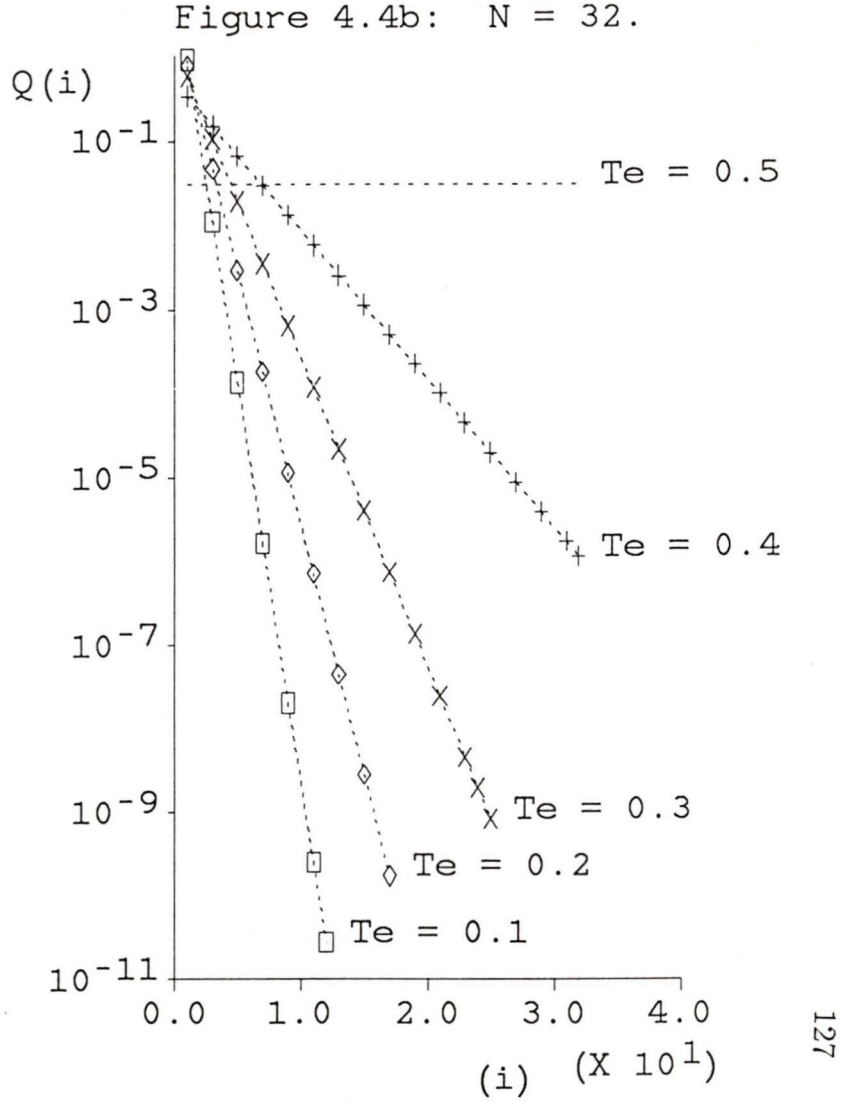


Figure 4.4: Steady state probability density Qi with hard limited filtered AWGN applied to the PD input.

value the simulation program is able to measure ($|e_k| \leq T/32$) [55]. It is desired to maximize the probability $Pr[T_{acq} \leq t]$ for all $t > 0$ to optimize the acquisition performance. A packet radio synchronization preamble data sequence $\{a_k\}$ is generally an alternating $\{\ldots, +1, -1, +1, \ldots\}$ sequence to maximize the number of data transitions. Hence acquisition is measured using this preamble sequence. Each distribution is based on 10^3 T_{acq} measurements.

The probability distribution of acquisition time T_{acq} for the maximum bandwidth ($2B_LT_{max}$) synchronizer is illustrated in figure 4.5. The fast acquisition performance of the $2B_LT_{max}$ synchronizer relative to the minimum bandwidth ($2B_LT_{min}$) synchronizer (figure 4.6) readily illustrates the advantage of a large bandwidth during acquisition.

The small tracking error advantage of the minimum bandwidth synchronizer is illustrated by the timing jitter variance measurements in figure 4.7. The calculated timing jitter variance σ_{nj}^2 is found from (3.75). The simulated timing jitter variance measurement is made with $\{a_k\}$ a pseudo-random data sequence.

Implementing the adaptive algorithm requires the selection of the parameters T_e, N, and L. After some experimentation the following general observations for parameter selection may be stated:

- Values of $T_e > 0.4$ are generally too large to ensure a high probability that the synchronizer will be at its maximum bandwidth state when the data signal is initially received.

- Values of $T_e \leq 0.1T$ allow the synchronizer to converge to the mini-

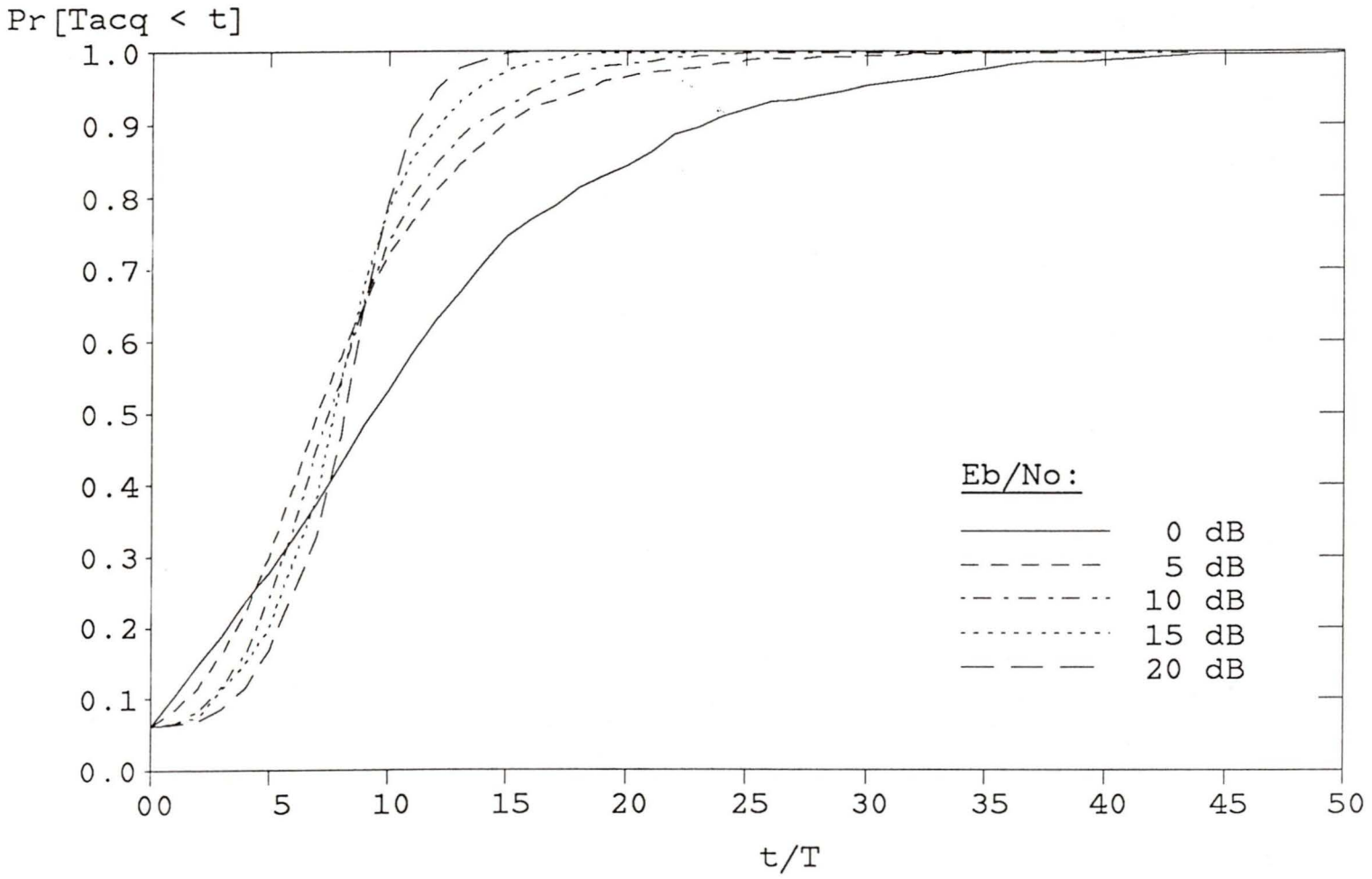


Figure 4.5: Simulation measurement of the acquisition time (Tacq) probability distribution function for the maximum bandwidth synchronizer ($K = 1/8$, $2B_L T_{max} = 0.067$).

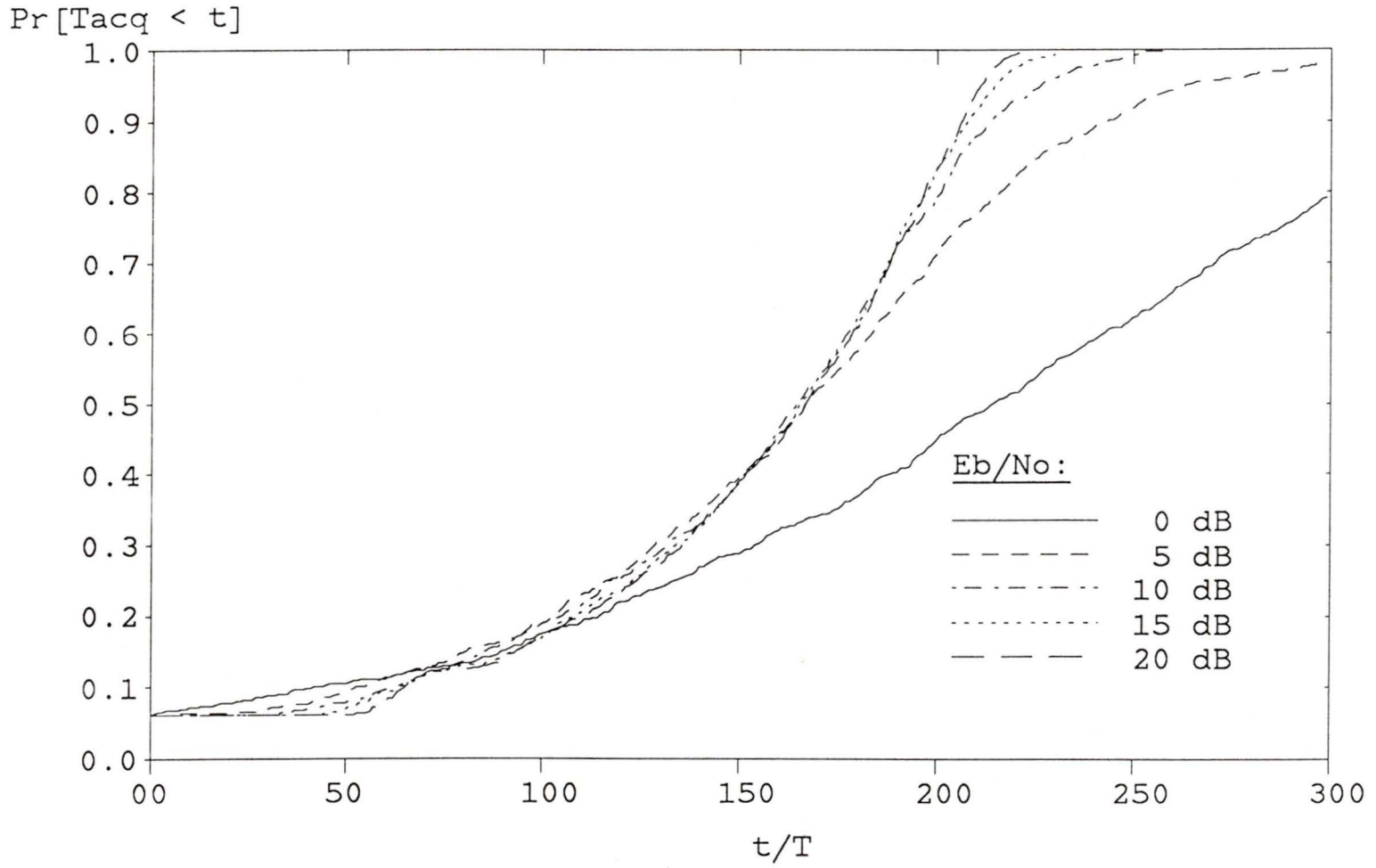


Figure 4.6: Simulation measurement of the acquisition time (Tacq) probability distribution function for the minimum bandwidth synchronizer ($K = 1/128$, $2B_LT_{min} = .0039$).

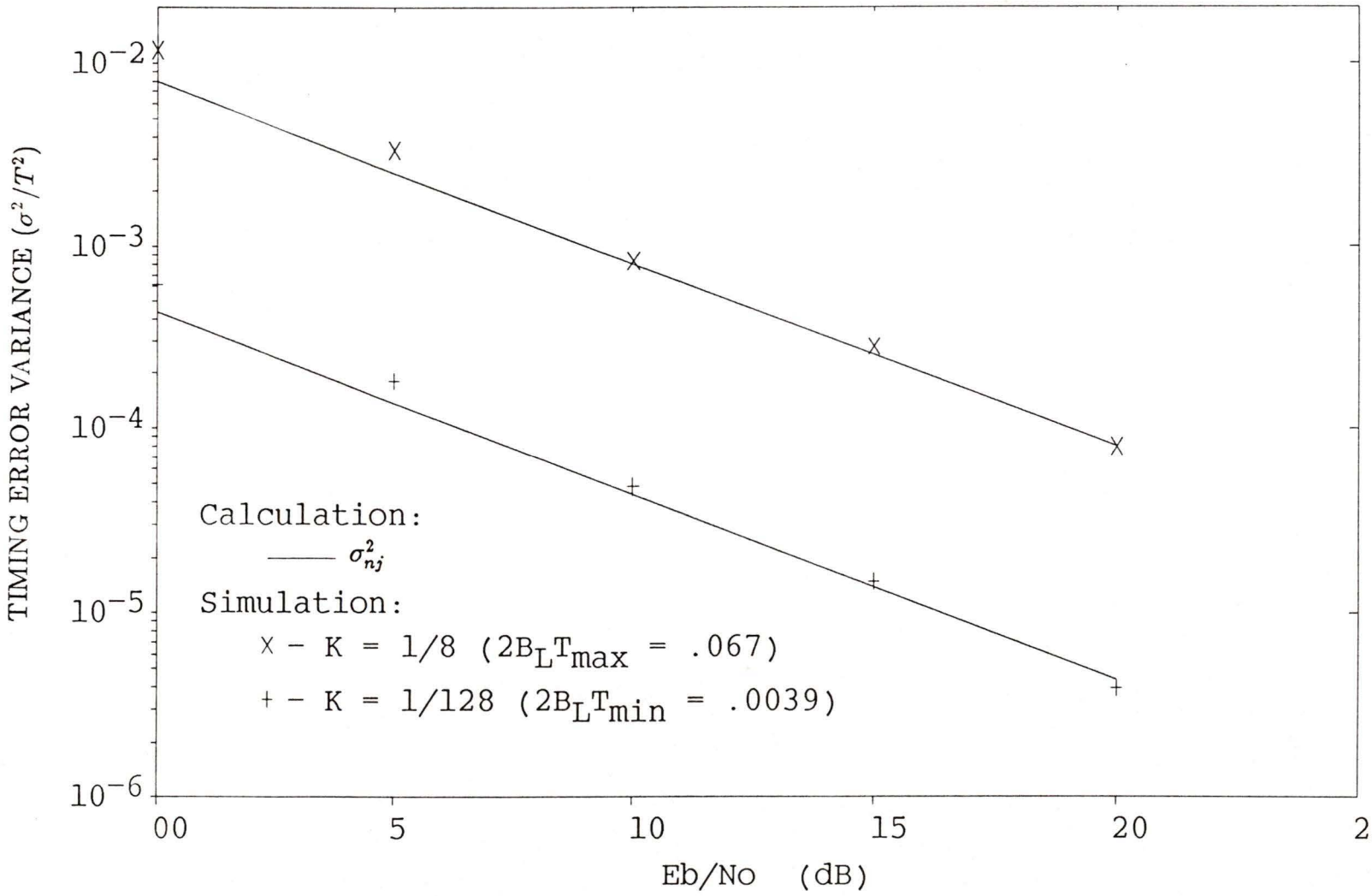


Figure 4.7: Simulated jitter variance vs. Eb/No of the maximum bandwidth ($2B_LT_{max}$) and minimum bandwidth ($2B_LT_{min}$) synchronizers.

mum bandwidth only at high SNR.

- The value of L should be chosen to be in the neighborhood of the number of preamble bits the maximum bandwidth synchronizer requires to achieve acquisition with a reasonably high probability over the required SNR range.

With these rough guidelines stated, the performance of a typical adaptive synchronizer may be evaluated.

Figure 4.8 illustrates the tracking performance of an adaptive synchronizer ($N = 16, L = 8$). As expected there is a transition region where the the synchronizer moves from acquisition mode to tracking mode. The range of E_b/N_o where the transition occurs decreases as T_e increases. For this synchronizer if $T_e \geq 0.2T$, the synchronizer has the tracking performance of the minimum bandwidth synchronizer over most of the indicated E_b/N_o range. The corresponding acquisition performance of the adaptive synchronizer for $T_e = 0.1T, 0.2T, 0.3T$, and $0.4T$ is shown in figures 4.9, 4.10, 4.11, and 4.12 respectively. The acquisition performance degrades as T_e increases, although even at $T_e = 0.4T$ the acquisition performance is still dramatically improved relative to the minimum bandwidth ($2B_L T_{min}$) synchronizer. The acquisition performance degradation can be attributed to the synchronizer converging to the minimum bandwidth before the acquisition has been achieved.

The acquisition performance can be improved by increasing both N and L, although a price is paid in terms of speed of convergence to the minimum bandwidth. As an example, consider the ($N = 32$, $L = 8$) adaptive synchronizer. The transition from acquisition mode to tracking

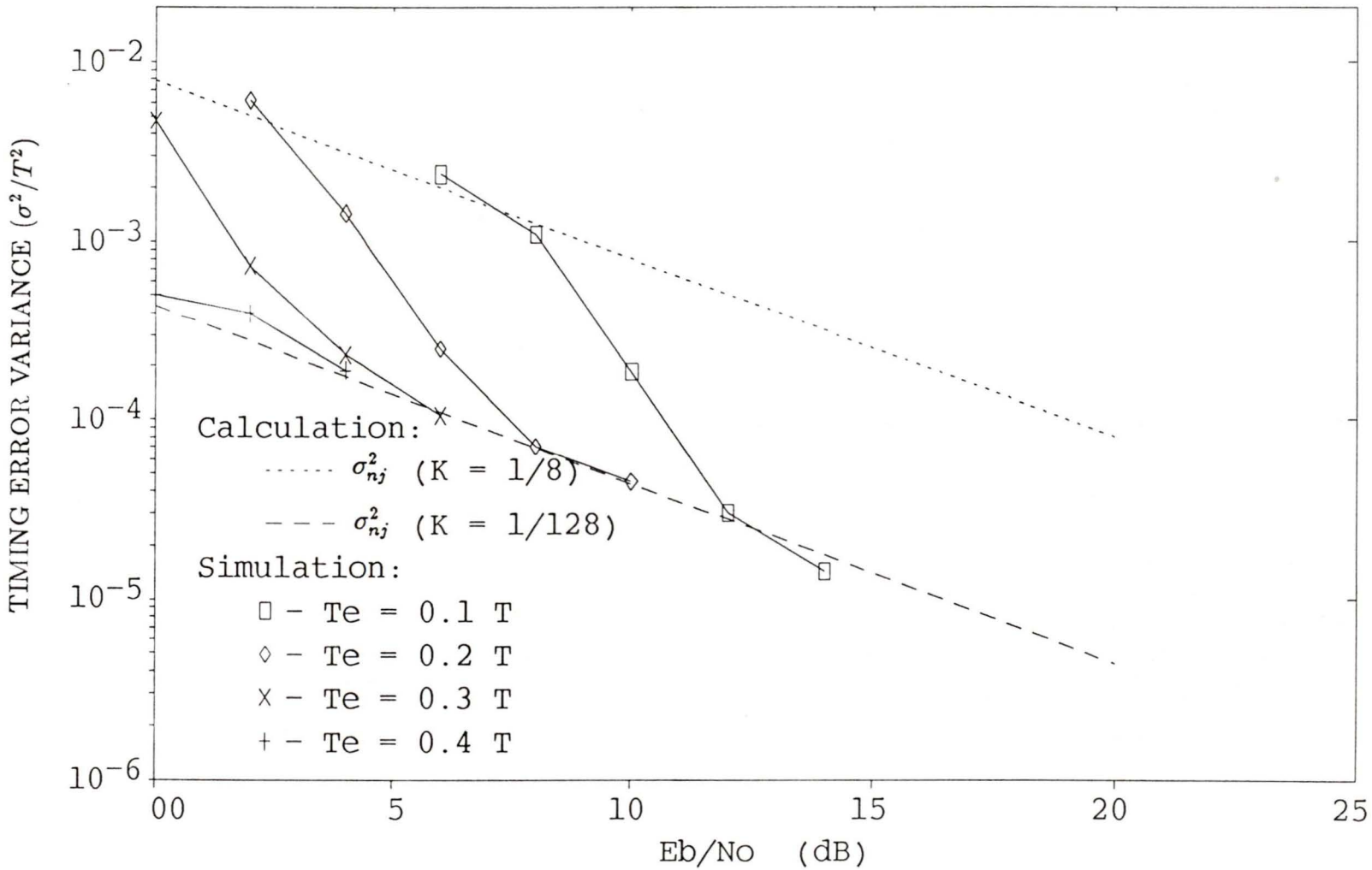


Figure 4.8: Simulation measurement of the jitter variance of an adaptive synchronizer (N = 16, L = 8). Simulation measurements in the transition region from acquisition mode to tracking mode are shown.

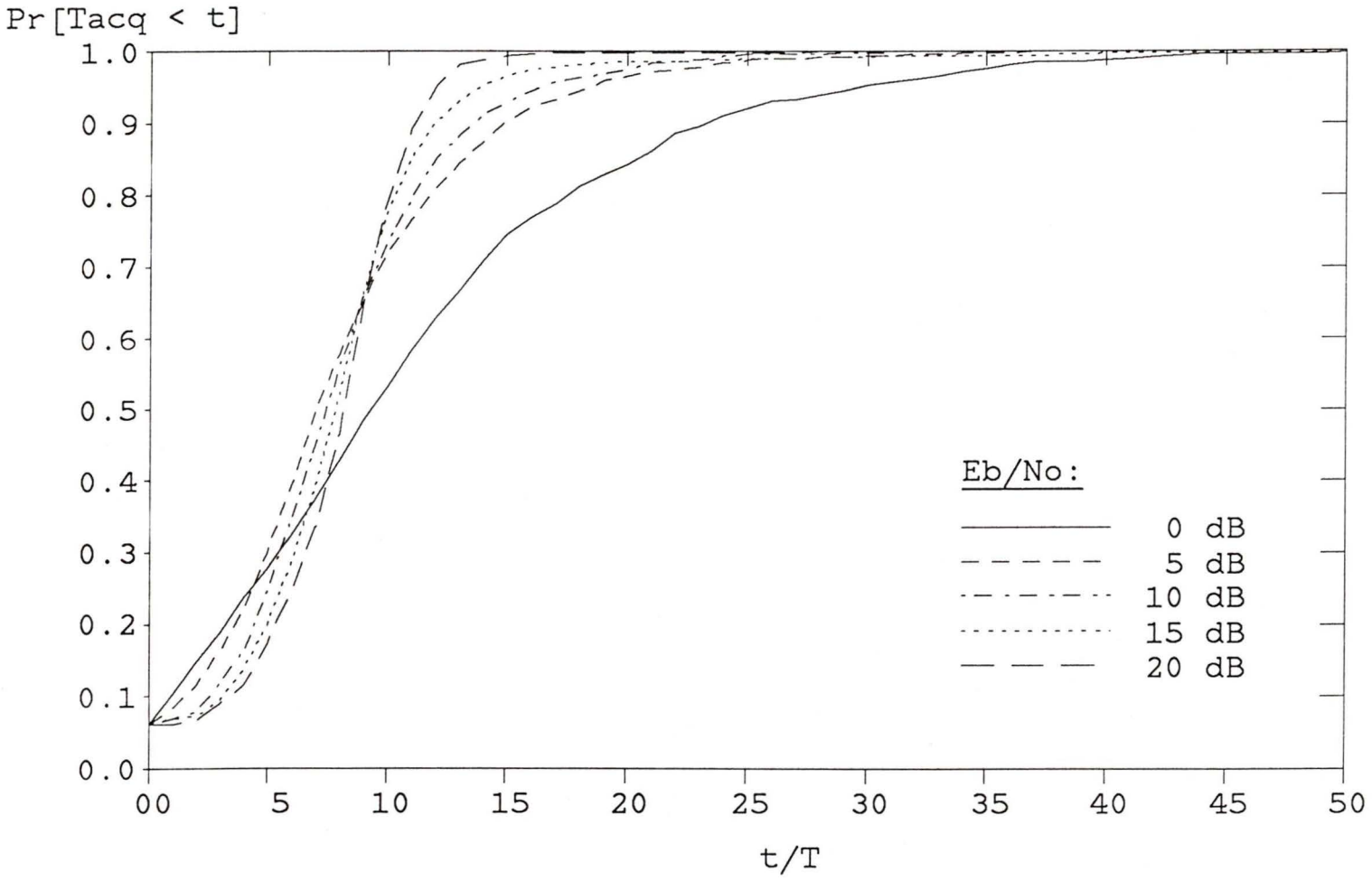


Figure 4.9: Simulation measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.1T).

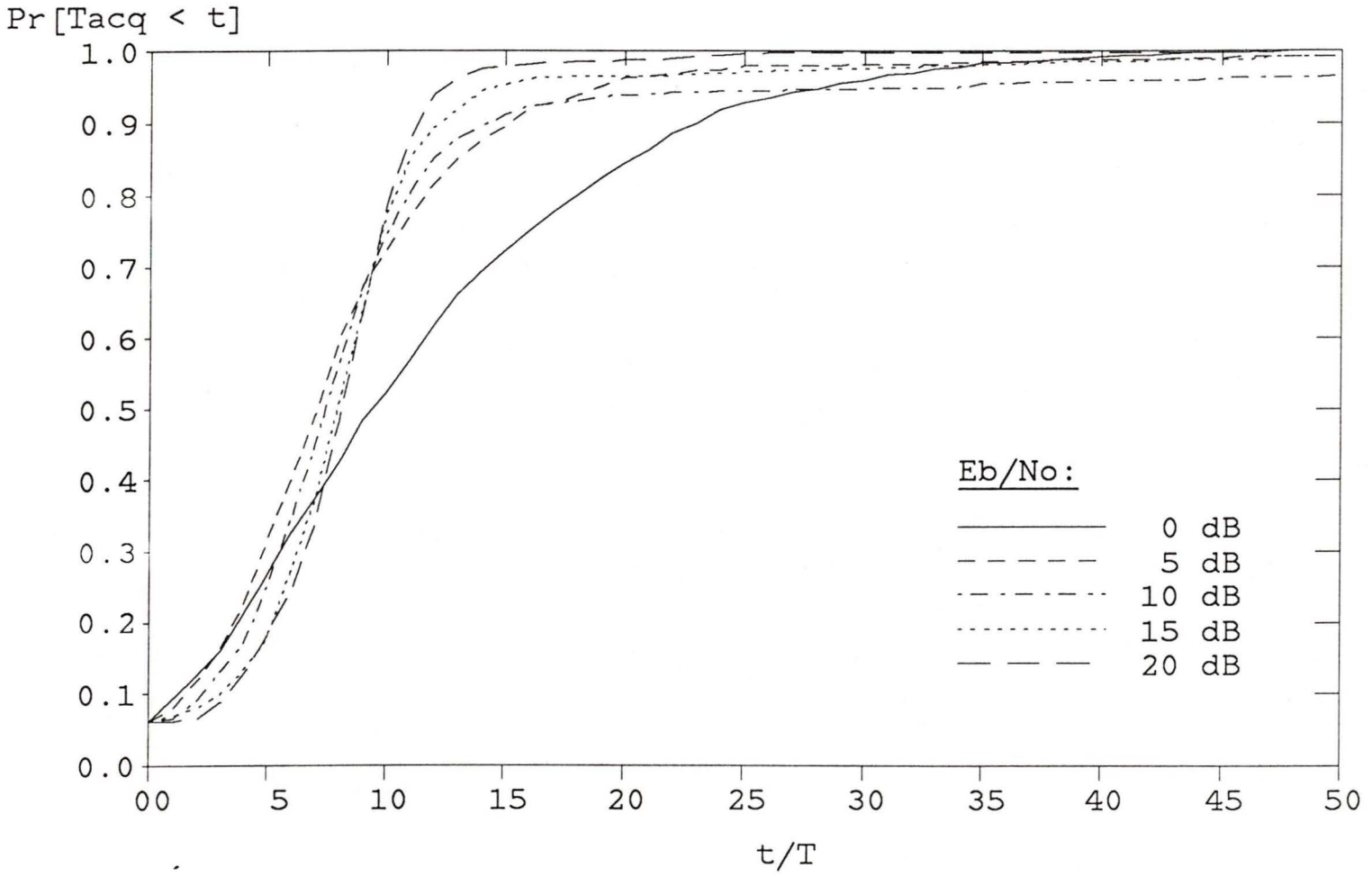


Figure 4.10: Simulation measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.2T).

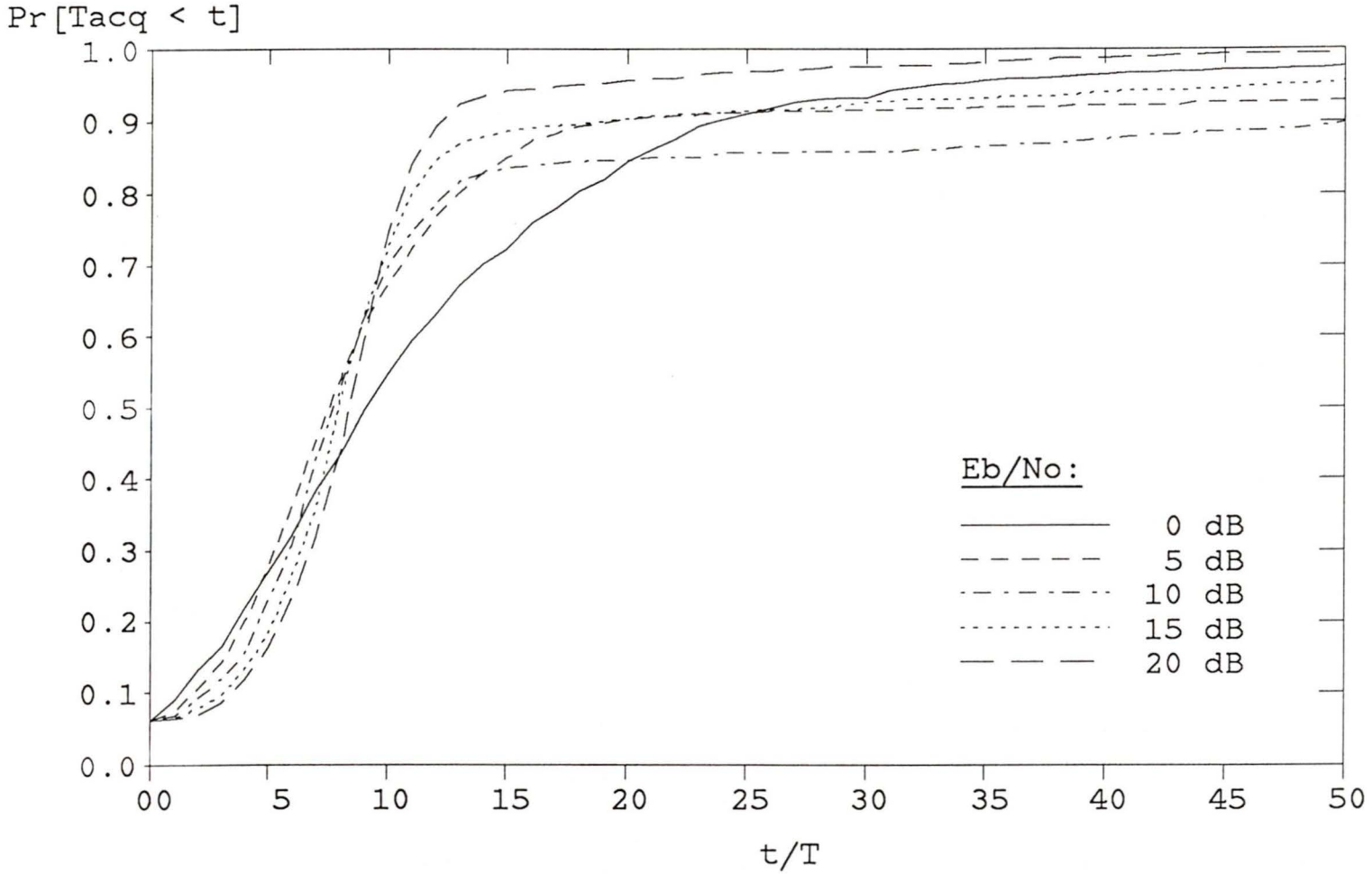


Figure 4.11: Simulation measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.3T).

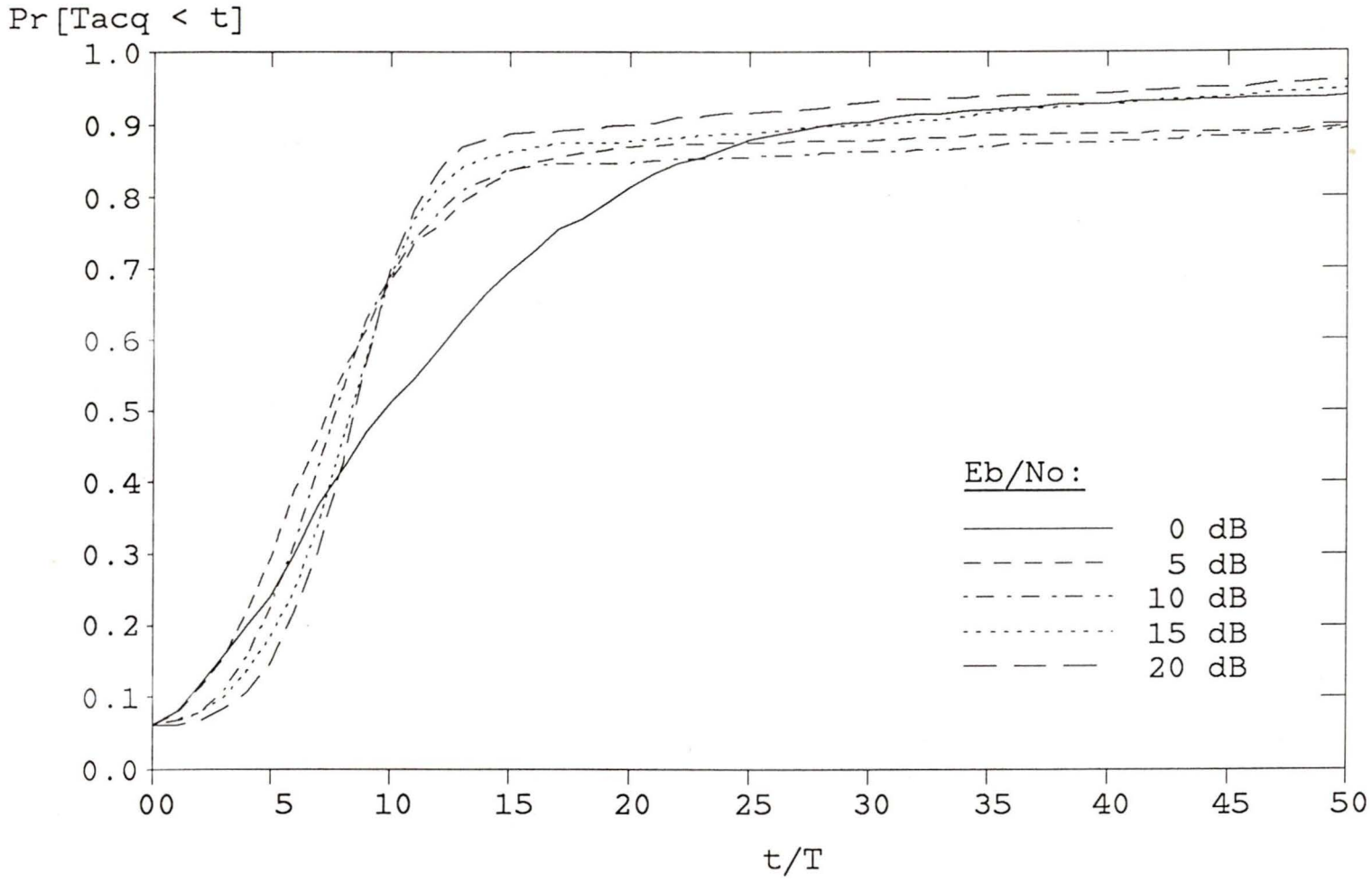


Figure 4.12: Simulation measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.4T).

mode (figure 4.13) when compared to the ($N = 16, L = 8$) result (figure 4.8) indicates that increasing N and L moves the transition region to a slightly lower E_b/N_o range. From figure 4.14, where $T_e = 0.4T$, the degradation in acquisition performance associated with the early convergence to minimum bandwidth is not as prevalent as it was for the ($N = 16, L = 8$) synchronizer (figure 4.12). As expected the acquisition performance improves for the larger L and N values.

The T_{acq} distribution function is a useful indication of the number of bits required to achieve synchronization with a high probability. Another test, which may be more valid, is to determine the bit error probability P_b as a function of the preamble bit number k. To measure $P_b(k)$, first the density function of the synchronizer timing error $f_{e_k}(e_k)$ is measured for $k = \{1, 2, \ldots\}$ using the simulation program. The average probability of error $P_b(k)$ is determined by averaging the conditional bit error probability $Pr[\hat{a}_k \neq a_k | e_k = e_i)$ over the density $f_{e_k}(e_k)$ [9]

$$P_b(k) = \sum_{i=1}^{NS} Pr[\hat{a}_k \neq a_k | e_k = e_i] Pr[e_k = e_i] \tag{4.25}$$

where $NS = 32$ is the number of discrete e_k locations in the simulation program. $P_b(k)$ is dependent on the pulse shape $x(t)$. This method of estimating the average bit error probability requires less computation than measuring and averaging actual bit errors. Within the available simulation time, we are unable to present accurate error probabilities $P_b \leq 10^-3$.

Figure 4.15 compares $P_b(k)$ for the maximum bandwidth ($2B_L T_{max}$) synchronizer and the adaptive ($N = 16, L = 8$) synchronizer. At high E_b/N_o

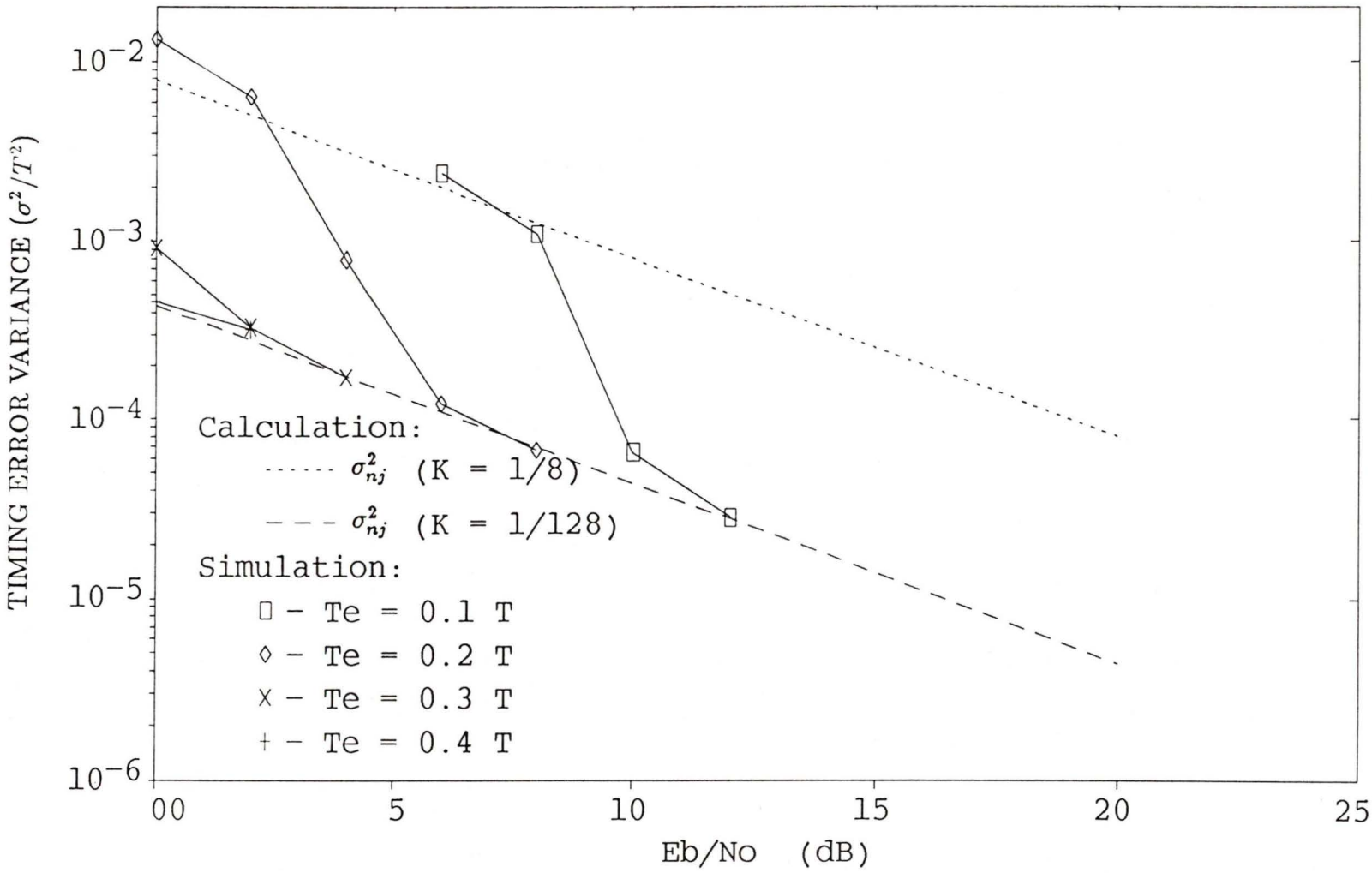


Figure 4.13: Simulation measurement of the jitter variance of an adaptive synchronizer (N = 32, L = 16). The transition region from acquisition to tracking mode is moved to a lower Eb/No range by increasing N and L.

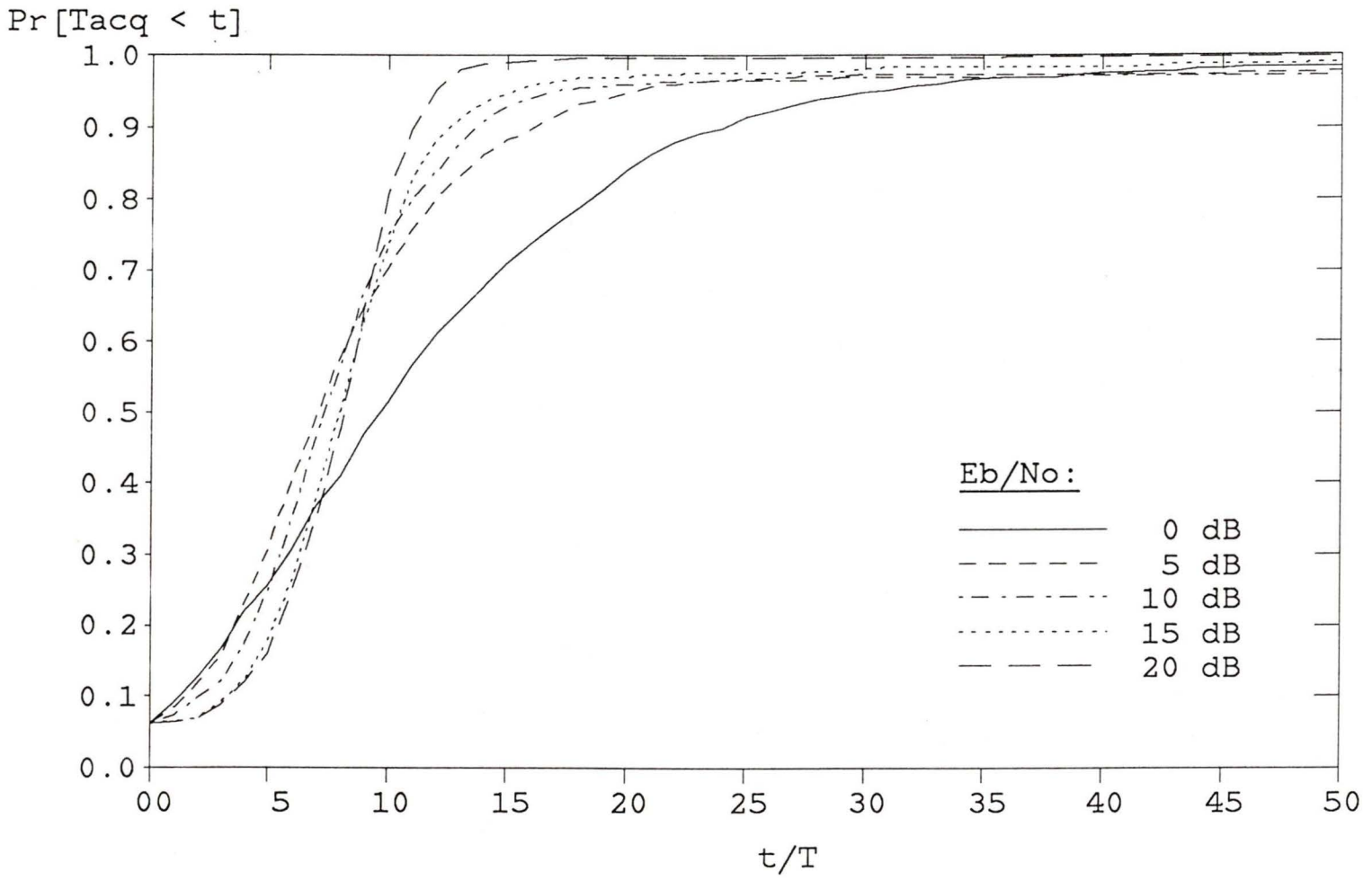


Figure 4.14: Simulation measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 32, L = 16, T_e = 0.4T).

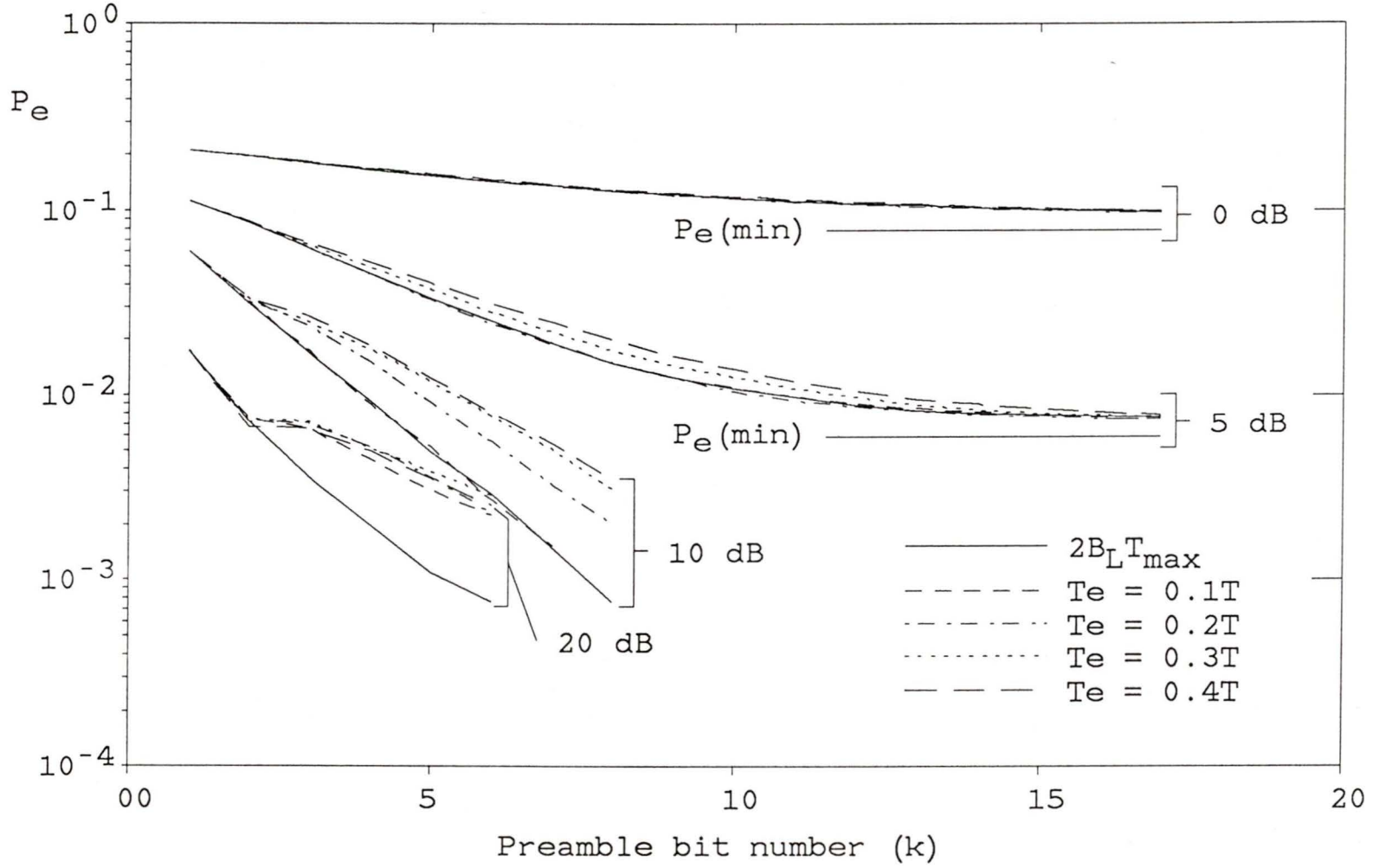


Figure 4.15: Simulation measurement of the preamble bit error probability with Eb/No (in dB) and Te as parameters (N = 16, L = 8).

and large T_e the adaptive synchronizer bit error performance degrades relative to the maximum bandwidth synchronizer. The degradation can again be attributed to the early convergence to minimum bandwidth. Increasing N and L, which reduces the probability of early convergence to minimum bandwidth, reduces the probability of bit error degradation (figure 4.16).

4.6 Implementation and Experimental Measurements

To verify this synchronization technique with experimental results, a hardware prototype system is implemented using a TMS32020 digital signal processing chip [63]. The hardware implementation and jitter and acquisition test equipment are described in appendix F.

The SRC matched filter system is approximated as follows: non return to zero (NRZ) signalling pulses are filtered by a fourth order Butterworth filter ($f_{3dB} = 1/T$) at the data source, and the LPF_R is approximated by a second order Butterworth filter ($f_{3dB} = 1/2T$). The recovered eye pattern (figure 4.17) appears to be a reasonable approximation to the $\alpha = 1$ SRC eye pattern [47]. For example there is very little ISI at the zero crossings.

The tracking jitter variance σ_{nj}^2 of the two fixed bandwidth ($2B_LT_{max}$ and $2B_LT_{min}$) synchronizers (figure 4.18) is in close agreement with both simulation (figure 4.7) and with theory (3.75).

The acquisition performance of the $2B_LT_{max}$ and $2B_LT_{min}$ synchronizers is illustrated in figures 4.19 and 4.20 respectively. The distribution functions are based on 150 acquisition measurements. The acquisition results

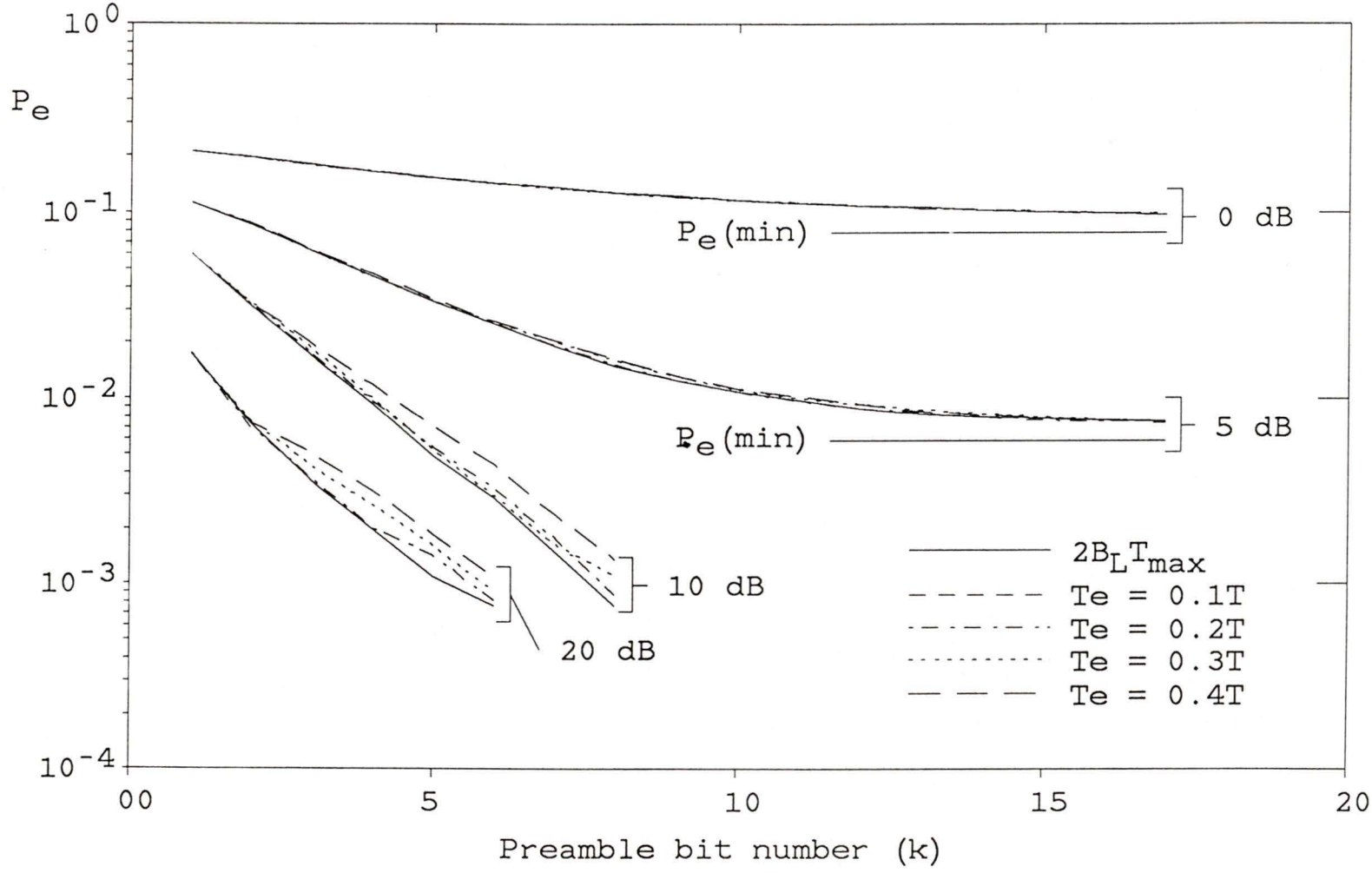


Figure 4.16: Simulation measurement of the preamble bit error probability with Eb/No (in dB) and Te as parameters (N = 32, L = 16).

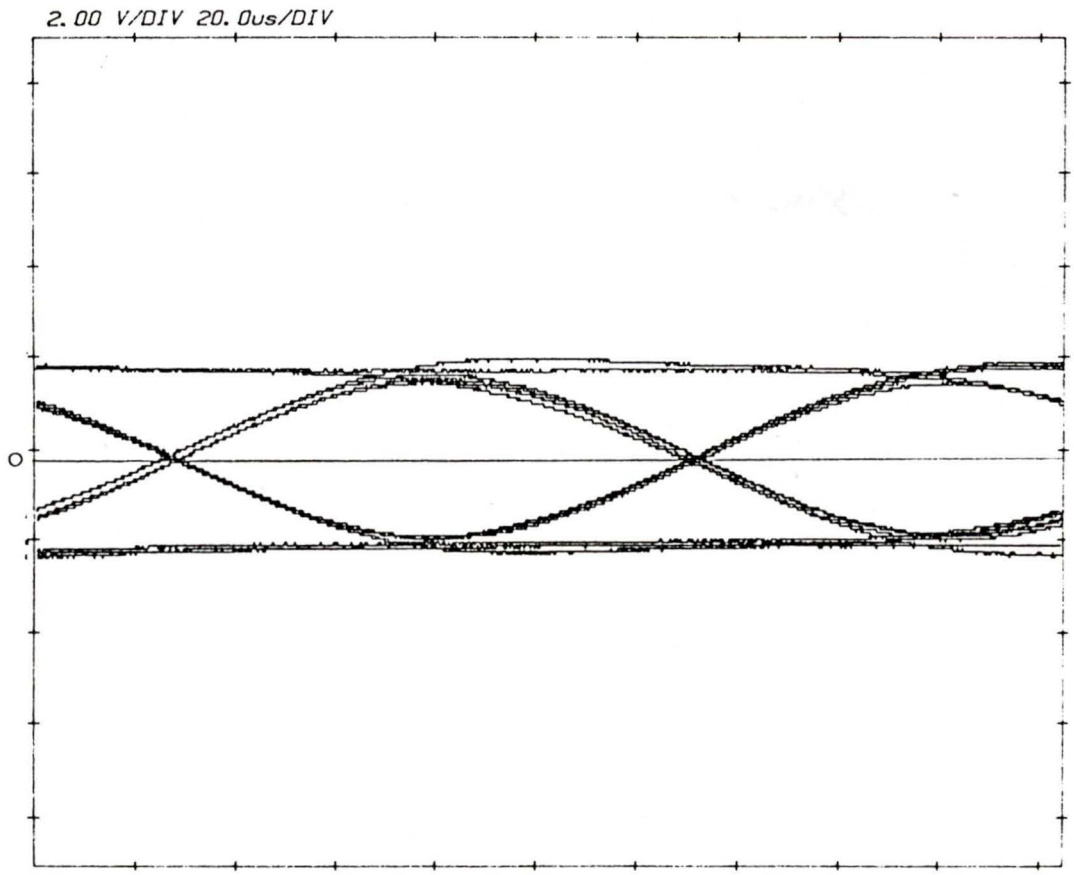


Figure 4.17: Recovered eye pattern at the LPF_R output.

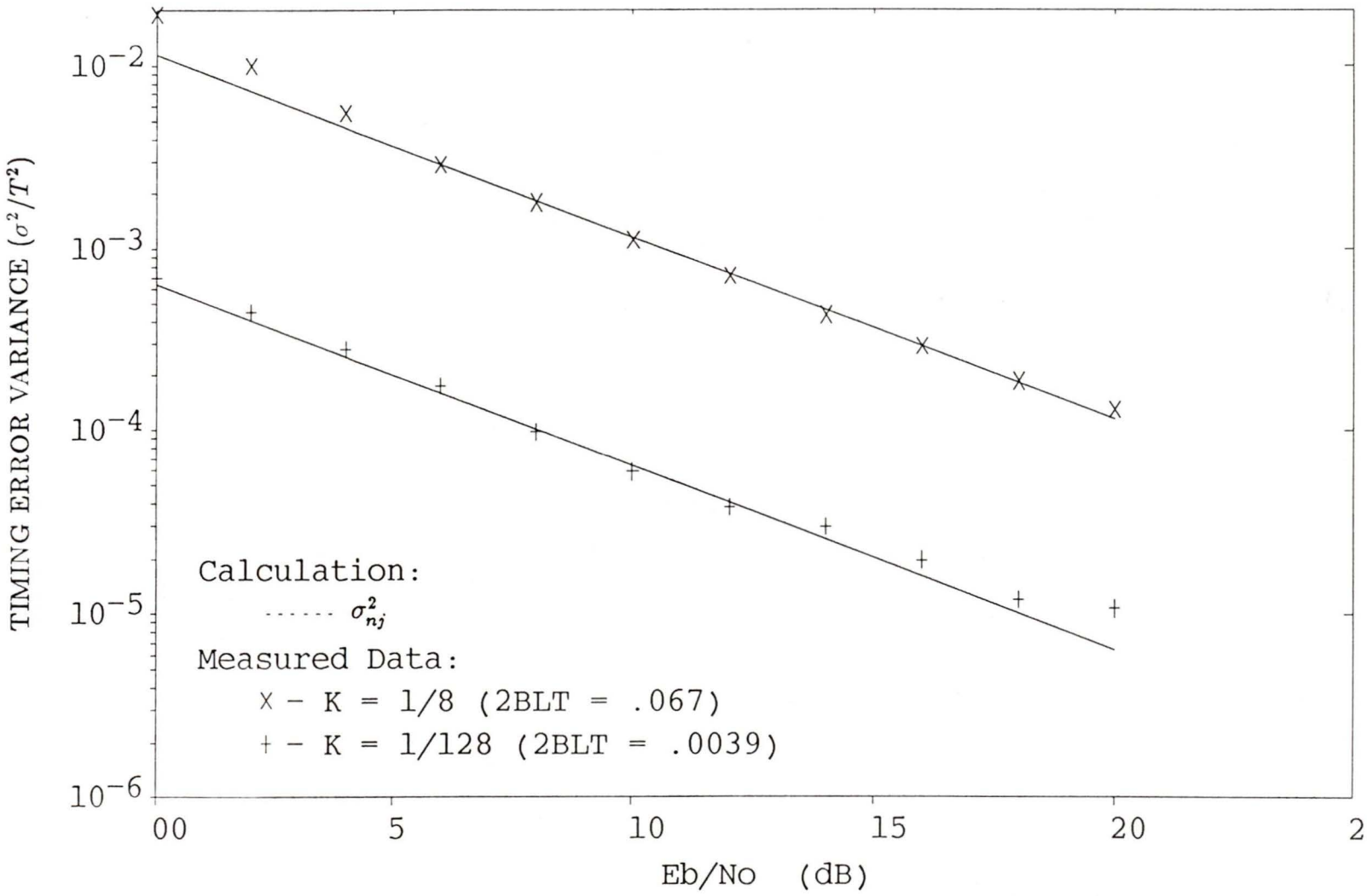


Figure 4.18: Measured jitter variance of the maximum bandwidth ($2BLT_{max}$) and the minimum bandwidth ($2BLT_{min}$) synchronizers.

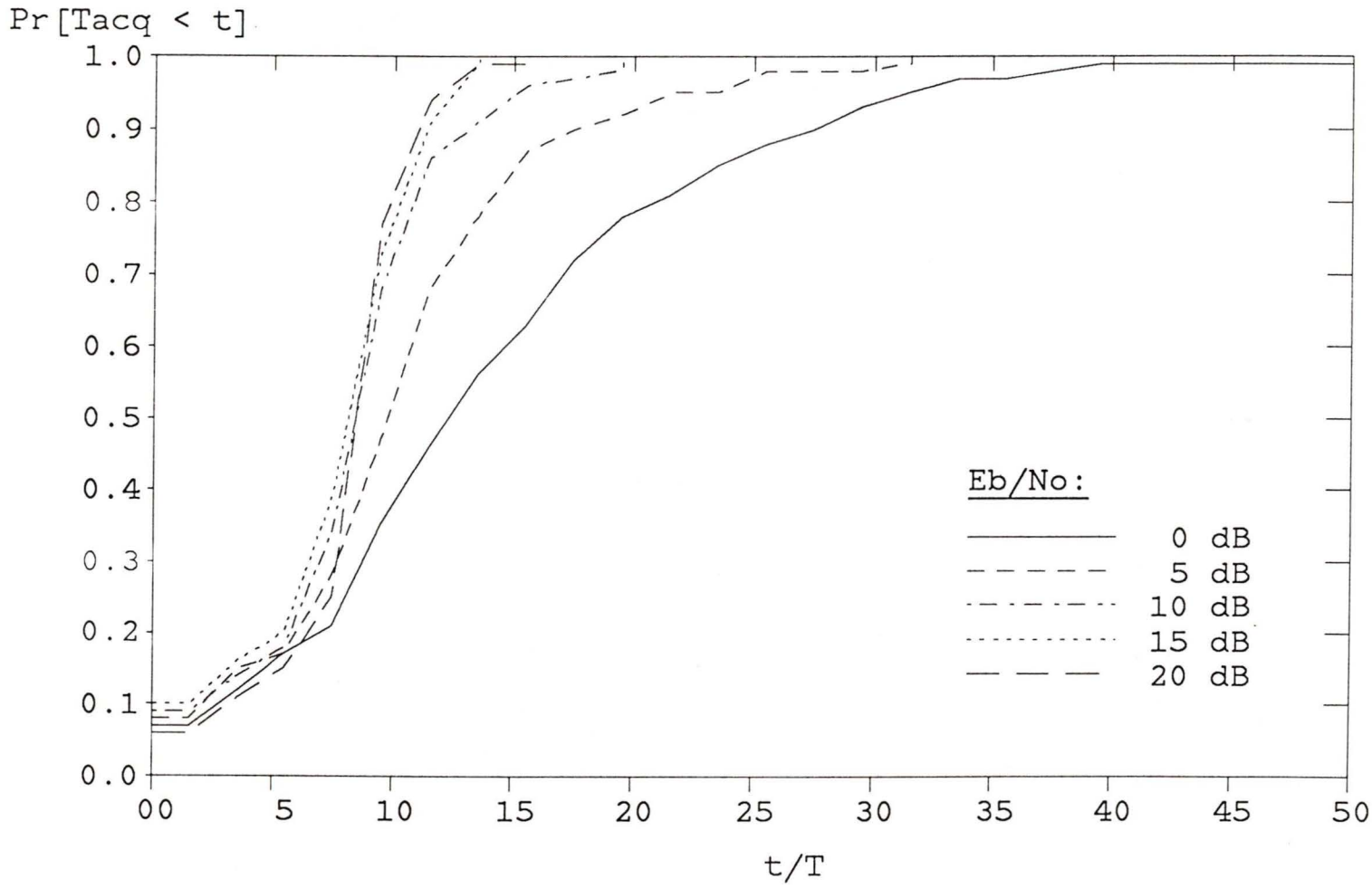


Figure 4.19: Experimental measurement of the acquisition time (Tacq) probability distribution function for the maximum bandwidth synchronizer (K = 1/8, $2B_LT_{max}$ = 0.067).

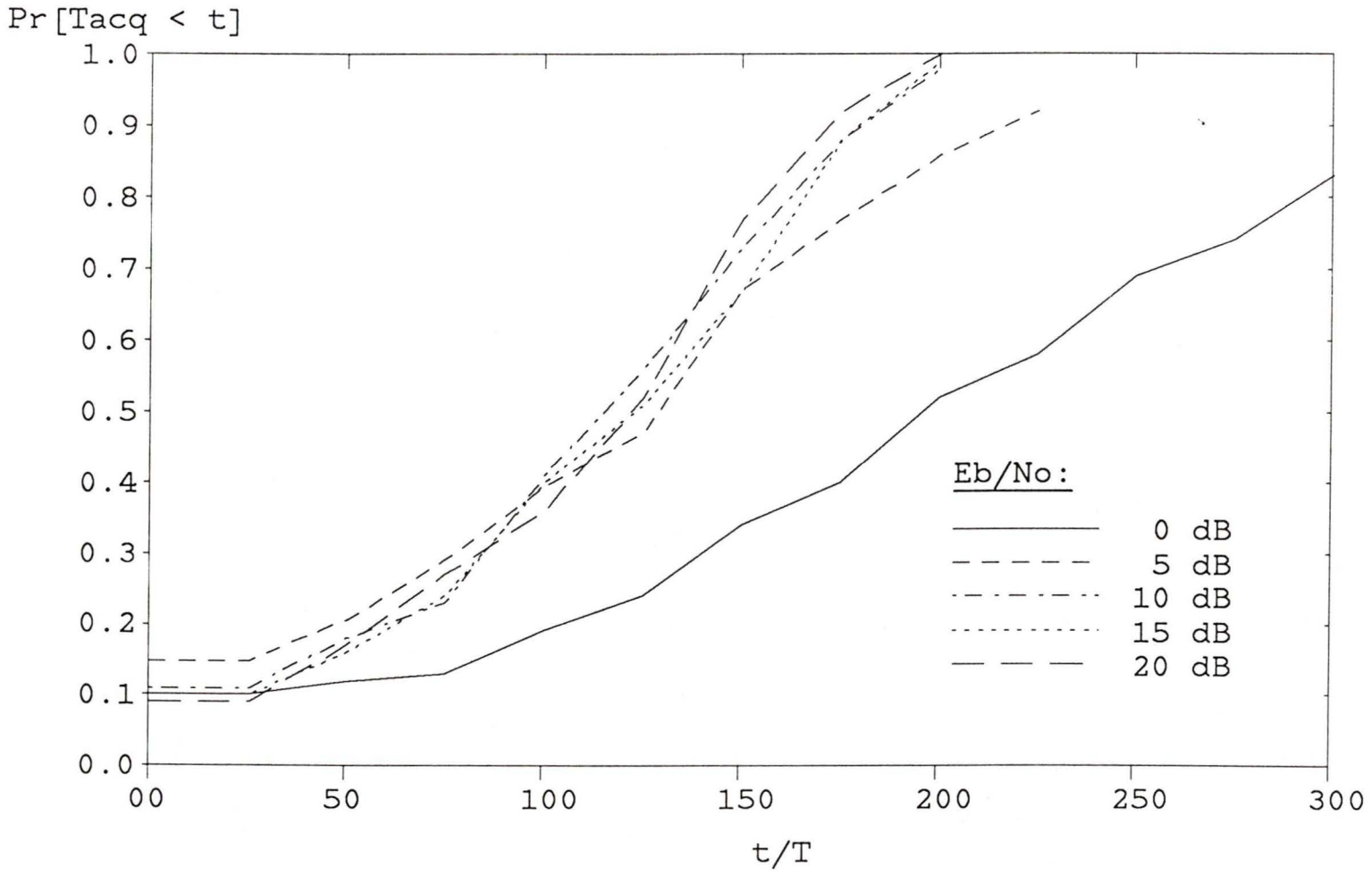


Figure 4.20: Experimental measurement of the acquisition time (Tacq) probability distribution function for the minimum bandwidth synchronizer ($K = 1/128$, $2B_L T_{min} = .0039$).

are similar to the simulation results (figures 4.5 and 4.6).

For the adaptive synchronizer, the transition from acquisition mode to tracking mode for the ($N = 16, L = 8$) synchronizer (figure 4.21) is similar to the simulation result (figure 4.8). Note that a direct comparison is not appropriate since the received experimental data signal is only an approximation to the SRC signal used in the simulation.

The acquisition performance of the adaptive synchronizer ($N = 16, L = 8$) for $T_e = 0.1T$, $T_e = 0.2T$, $T_e = 0.3T$ and $T_e = 0.4T$ is illustrated in figures 4.22, 4.23, 4.24, and 4.25 respectively. The nature of the synchronizer acquisition is essentially the same as has been determined in the simulations. The degradation associated with the early convergence to minimum bandwidth is readily apparent in the $T_e = 0.4T$ results (figure 4.25). For $T_e \leq 0.3T$, the acquisition performance is only slightly degraded from the maximum bandwidth acquisition performance (figure 4.19).

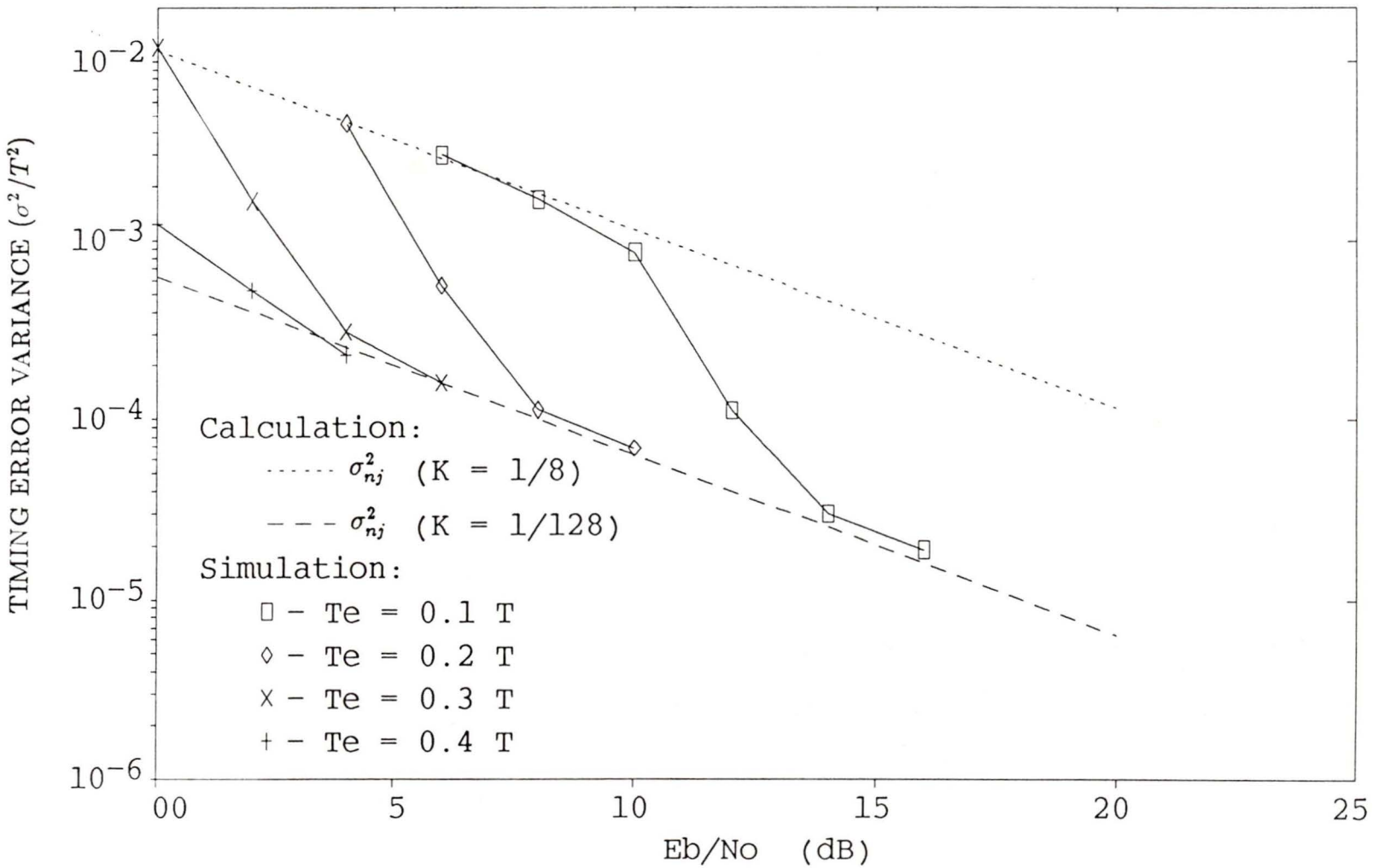


Figure 4.21: Experimental measurement of the jitter variance of an adaptive synchronizer (N = 16, L = 8). Measurements in the transition region from acquisition mode to tracking mode are shown.

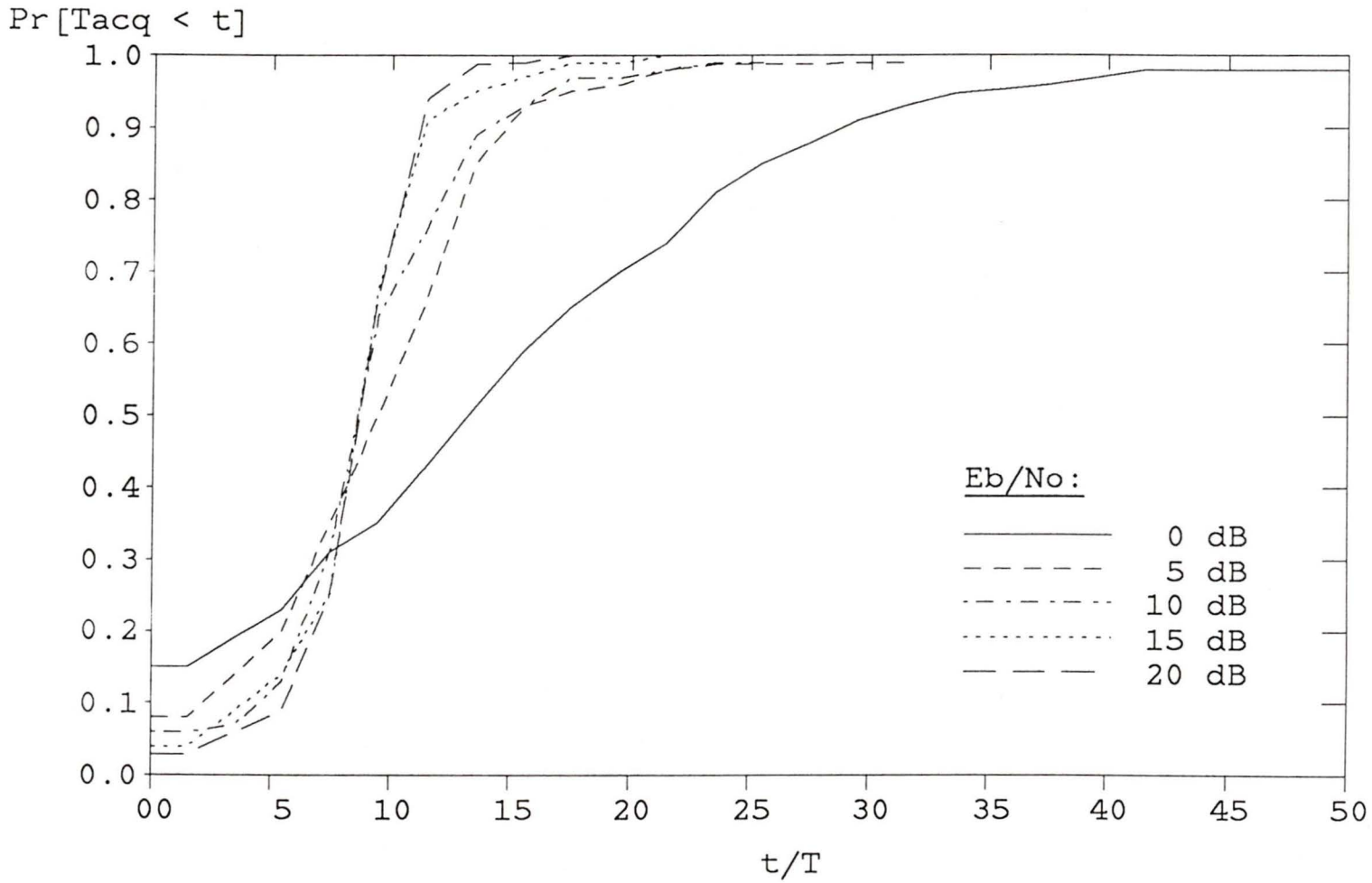


Figure 4.22: Experimental measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.1T).

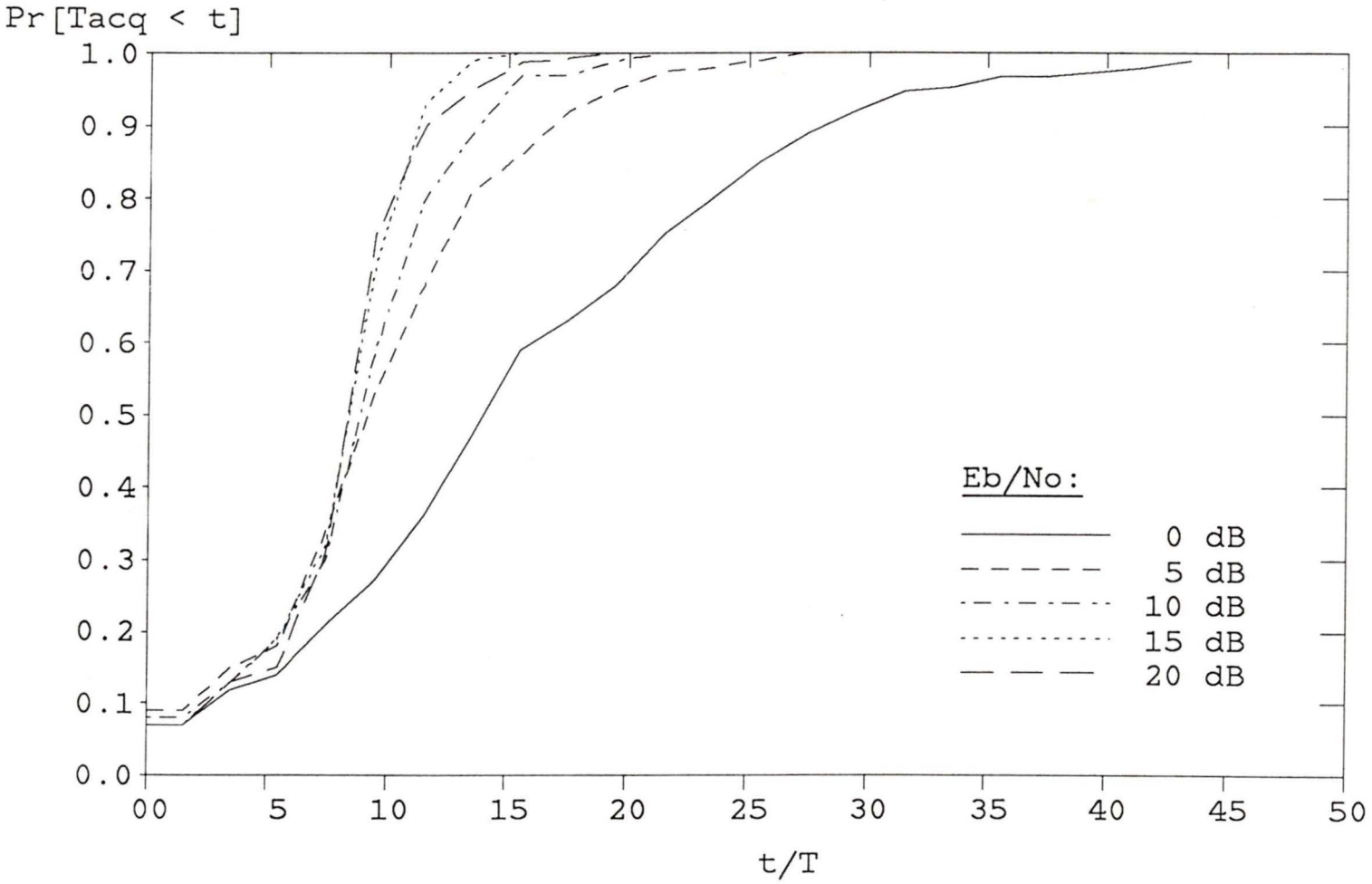


Figure 4.23: Experimental measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.2T).

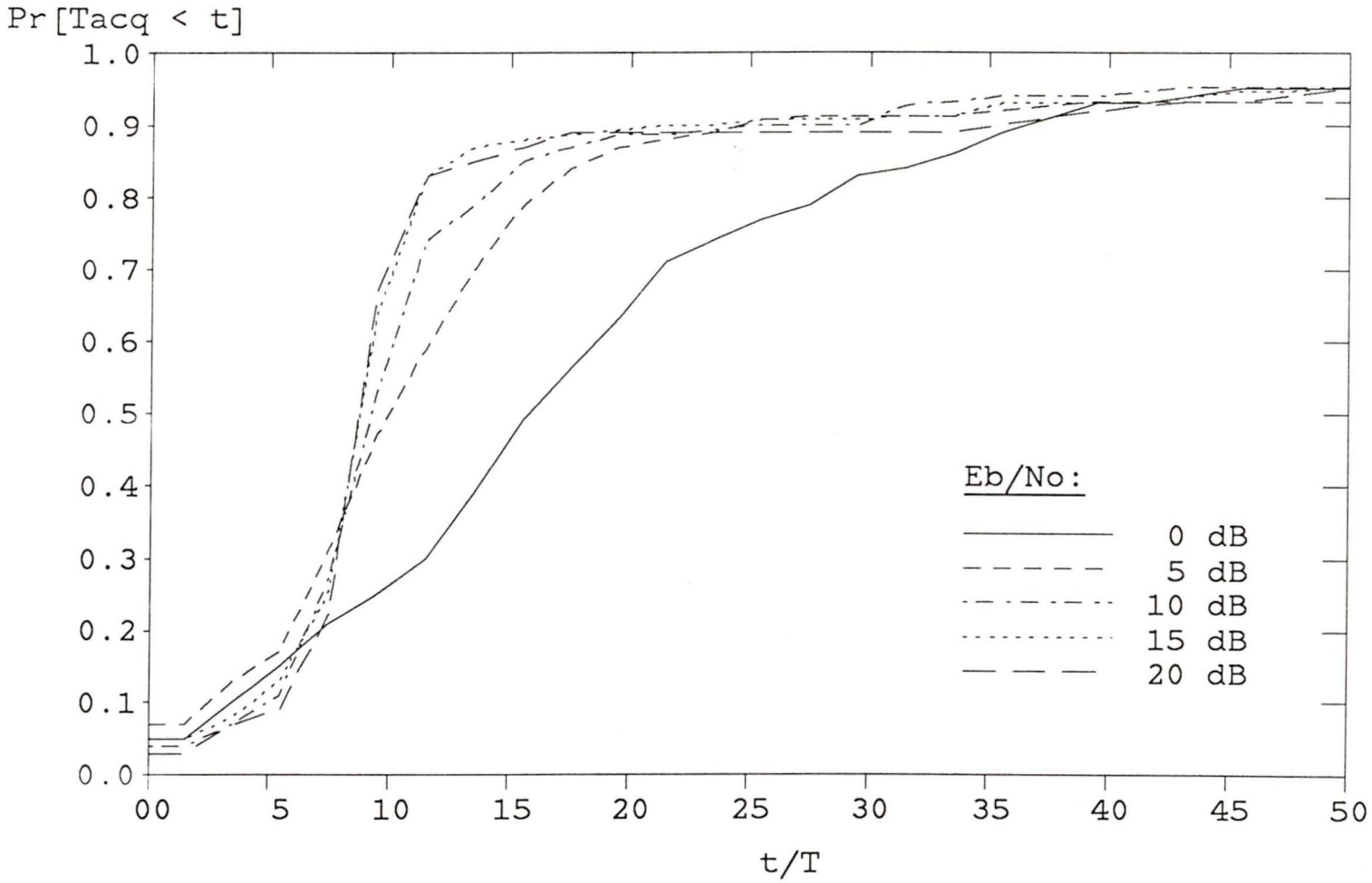


Figure 4.24: Experimental measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.3T).

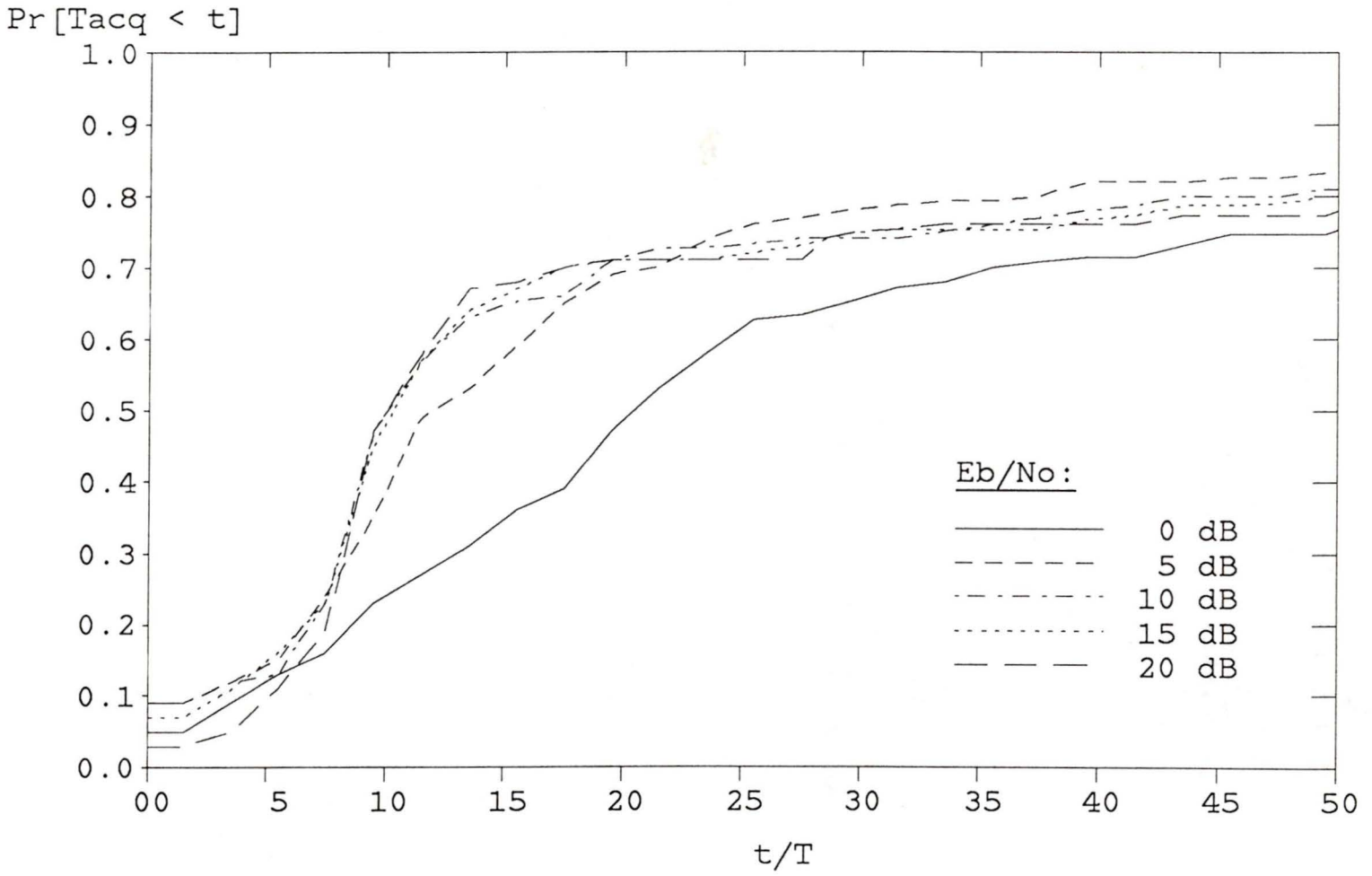


Figure 4.25:Experimental measurement of the acquisition time (Tacq) probability distribution function for an adaptive synchronizer (N = 16, L = 8, T_e = 0.4T).

4.7 Summary

In this chapter we have proposed a simple technique to adaptively adjust the bandwidth of a digital PLL, where the adaptation is based on the consistency of measured timing errors at the output of a simple counter type PD. The PD input is a hard limited binary data signal. The adaptive bandwidth control maintains the synchronizer at maximum bandwidth when acquisition is required and reduces the bandwidth during tracking operation to reduce the timing jitter variance. A Markov chain model of the adaptive bandwidth control is discussed to illustrate the bandwidth adaptation procedure, and analytical results to predict the probability distribution of loop gain values $K^{(i)}$ are given.

A simulation study of the adaptive PLL over the SNR range where error probability is of typical interest ($0dB \leq E_b/N_o \leq 20dB$) indicates that the tracking performance of a narrow bandwidth ($2B_LT = .0039$) synchronizer can be obtained without significantly sacrificing the acquisition performance of a wide bandwidth ($2B_LT = .067$) synchronizer. The adaptive PLL parameters T_e, N, and L may be used to tailor the performance of the synchronizer to specific applications. Increasing T_e extends the E_b/N_o range over which the synchronizer will have the tracking performance of the minimum bandwidth synchronizer. However large values of T_e converge faster to the minimum bandwidth, and acquisition performance is therefore degraded. The early convergence to minimum bandwidth for large T_e values can be counteracted by increasing N, the total number of states in the Markov chain, and L, the number of states in the Markov chain where the synchronizer operates at maximum bandwidth. However this will result

in a slower convergence to minimum bandwidth. The simulation results also indicate that the early convergence to minimum bandwidth slightly increases (relative to the maximum bandwidth synchronizer) the average bit error probability during acquisition. The average probability of bit error probability degradation may be reduced by increasing both N and L.

Laboratory experimental measurements of a hardware prototype synchronizer verify the synchronizer acquisition and tracking performance. We were able to achieve the tracking performance of the minimum bandwidth synchronizer over most of the Eb/No range without significantly sacrificing the acquisition performance of the maximum bandwidth synchronizer.

The evaluation of the proposed synchronizer in this chapter is restricted to a simulation study and some laboratory measurement results. The PLL parameters have been chosen based on hardware measurement constraints and ease of implementation rather than being optimized for specific applications. The evaluation of the proposed synchronizer should be continued for specific applications where the adaptive bandwidth control would be of interest. As mentioned previously, the adaptive bandwidth control may be useful in a fading environment such as a mobile radio channel. The ability of the synchronizer to remain synchronized through deep signal fades when locked into the minimum bandwidth state should be determined. For an adaptive synchronizer that remains locked in the minimum bandwidth state once it is reached, the effects of false lock probability on TDMA or CSMA system performance could be investigated. Algorithms to rapidly switch from tracking mode to acquisition mode at the end of a data packet should also be considered.

Chapter 5

Conclusions and Further Research Recommendations

In chapter 2 the optimum data-aided and non-data-aided maximum likelihood (ML) hard limited synchronizers are presented. The synchronizers may not be feasible to implement, but using linear approximations data-aided and non-data-aided synchronizers that are more suitable for implementation, for example in closed loop operation, are developed. The lower bound on the timing error variance is established and it is shown to be a fixed constant ($\pi/2$) higher than the non hard limited optimum synchronizer. Simulation results indicate that the lower bound calculation is a reasonable approximation for a large number of samples per bit (≈ 100), while for a small number of samples per bit (≈ 10) the hard limited and non-hard limited results tend to converge. This indicates that for a small number of samples per bit, the degradation associated with hard limiting may be less than the $\pi/2$ factor the lower bound indicates. Samples sizes of less than 10 samples per bit will violate analysis assumptions, and therefore further work is required to confirm this result for smaller sample sizes.

There are many extensions to the work presented in chapter 2 yet to be investigated. For example the performance of hard limited closed loop systems relative to non hard limited systems is of interest.

The analysis of chapter 2 assumes additive white Gaussian noise and a prefilter with a bandwidth large enough to ensure both no intersymbol interference and statistically independent noise samples. Often a filter is present which will bandwidth limit the signal such that both of these assumptions are violated. A re-evaluation of the ML synchronization problem including the effects of filtering should be considered.

The optimum hard limited ML synchronizer equations are simplified using high sampling rate assumptions and linear approximations. As might be expected the resulting expressions involves weighting the sampled hard limited waveform by the sampled signalling waveform $x(t)$. It may be appropriate to investigate weighting functions that more closely approximate the optimum result for fixed sampling rates and high and low signal to noise ratios, as has been done for the hard limited detection problem [29].

Chapter 3 presents a new technique for reducing pattern jitter within the feedback path of a PLL synchronizer. This synchronizer does not require an analog or digital prefilter [46]. The new timing error variance result (3.45) is an improvement over the previous result [18], especially for wide bandwidth synchronizers. The pattern jitter compensation technique is also incorporated into the analysis.

The simulation and experiment results indicate that the technique may be used to effectively eliminate pattern jitter from the recovered timing signal for some typical signalling pulses and most practical signal to noise

ratios. For applications where the impulse response of the transmission system is not known, an adaptive pattern jitter compensation technique has been proposed.

The concept of reducing pattern jitter in the feedback path of a PLL synchronizer diverges from the usual prefiltering approach. The pattern jitter compensation technique can be extended to any digital PLL synchronizer, and is not restricted to hard limited signals or binary signalling. If the output of a phase detector contains a data dependent disturbance, it follows that a combination of data values and compensation values (either known or adaptively estimated) may be used to compensate the disturbance. Such a system may find application in ISDN data receivers, where prefilter techniques are currently used [57]. The adaptation speed of convergence of the adaptive compensation technique should be investigated. Also the effects of frequency offsets should be considered.

The results of chapter 4 indicate that for TDMA and CSMA applications the tradeoff between acquisition performance and tracking performance of a synchronizer can be significantly reduced by utilizing a relatively simple adaptive bandwidth control. The system parameters (error threshold T_e, Markov chain length N, and the chain values $K^{(i)}$) may be used to tailor the synchronizer performance to specific applications.

The adaptive synchronization technique may be useful for packet mobile radio applications, where rapid acquisition during the synchronization preamble is required, and it is also required to have a very narrow bandwidth synchronizer to maintain synchronization through deep fades in the received signal strength. The performance of the adaptive synchronizer in

a mobile radio environment is therefore of interest.

Improvements to the bandwidth adaptation algorithm may be necessary to improve both the acquisition and tracking performance of the synchronizer. An adaptive threshold value T_e may be required to prevent the convergence to minimum bandwidth before acquisition is complete.

References

[1] H. Pfannschmidt and W. Schmidt, "Evolution of public land mobile systems toward digital voice and data transmission using narrowband schemes," *Nordic Conference on Digital Land Mobile Radiocommunication*, Helsinki, Feb. 1985, pp.35-44.

[2] CCITT I-series recommendations.

[3] M. Oerder and H. Meyr, "Digital filter and square timing recovery," *IEEE Trans. Commun.,* vol. COM-36, pp. 605-612, May, 1988.

[4] M. Moeneclaey, "Two maximum-likelihood symbol synchronizers with superior tracking performance", *IEEE Trans. Commun.,* vol. COM-28, pp. 1178-1185, Nov. 1984.

[5] M. Moeneclaey, "A simple lower bound on the linearized performance of practical symbol synchronizer," *IEEE Trans. Commun.,* vol. COM-31, pp. 1029-1032, Sept. 1983.

[6] A. Jennings and B. R. Clarke, "Data-sequence selective timing recovery for PAM systems," *IEEE Trans. Commun.,* vol. COM-33, pp. 729-731, Jul. 1985.

[7] K. H. Mueller and M. Müller, "Timing recovery in digital synchronous data receivers," *IEEE Trans. Commun.,* vol. COM-24, pp. 516-531, May 1976.

[8] D. Chu, "Phase digitizing sharpens timing measurements," *IEEE Spectrum*, vol. 25, No. 7, pp. 28-32, July 1988.

[9] W. C. Simon and M. K. Lindsey, *Telecommunications Systems Engineering,* Englewood Cliffs N. J.: Prentice Hall, 1973, ch. 9.

[10] V. K. Bhargava, D. Haccoun, R. Matyas, and P. P. Nuspl, *Digital Communication by Satellite,* New York: Wiley, 1981, ch. 5.

[11] R. E. Ziemer and R. L. Peterson, *Digital Communications and Spread Spectrum Systems*, New York: MacMillan, 1985, ch. 6.

[12] J. K. Holmes, *Coherent Spread Spectrum Systems*, New York: Wiley, 1982, ch. 12.

[13] K. H. Feher, *Digital Communications,* Englewood Cliffs: Prentice Hall, 1983, ch. 7.

[14] J. J. Spilker, *Digital Communications by Satellite*, Englewood Cliffs: Prentice-Hall, ch. 14.

[15] L. E. Franks, "Carrier and bit synchronization in data communication," *IEEE Trans. Commun.,* vol. COM-28, pp. 1107-1121, Aug. 1980.

[16] P. A. Wintz and E.J. Luecke, "Performance of self bit synchronization systems," School of Electrical Engineering, Purdue University, Lafayette Ind., Tech. Rept. TR-EE68-1, January 1968.

[17] M. Moeneclaey, "Synchronization Problems in PAM systems," *IEEE Trans. Commun.,* vol. COM-28, pp. 1130-1136, Aug. 1980.

[18] B. R. Saltzberg, "Timing recovery for synchronous binary data transmission", *Bell System Technical Journal*, pp. 593-623, March 1967.

[19] A. N. D'Andrea, "Performance analysis of the delay line-line clock regenerator," *IEEE Trans. Commun.,* vol. COM-34, pp. 321-328, Apr. 1986.

[20] T. L. Le-Ngoc and K. Feher, "A digital approach to symbol timing recovery," *IEEE Trans. Commun.,* vol. COM-28, pp. 1993-1999, Dec. 1980.

[21] Y. Takasaki, "Timing extraction in baseband pulse transmission", *IEEE Trans. Commun.,* vol. COM-20, pp. 877-884, Oct 1972.

[22] F. Gardner, "Self noise in synchronizers", *IEEE Trans. Commun.,* vol. COM-28, pp. 1159-1163, Aug. 1980.

[23] L. E. Franks and J. P. Bubrouski, " Statistical properties of timing jitter in a PAM recovery scheme," *IEEE Trans. Commun.,* vol. COM-22, pp. 913-920, July 1974.

[24] A. E. Payzin, "Analysis of a digital bit synchronizer," *IEEE Trans. Commun.,* vol. COM-31, pp. 554-560, April 1983.

[25] R. E. Ziemer and R. L. Peterson, *Digital Communications and Spread Spectrum Systems*, New York: MacMillan, 1985, ch. 3.

[26] V. Milutinovic, "A comparison of suboptimal detection algorithms applied to the additive mix of orthogonal sinusoidal signals," *IEEE Trans. Commun.,* vol. COM-36, pp. 538-543, May 1988.

[27] N. C. Beaulieu, "Penalties of sample and sum and weighted partial decision detectors in Gaussian noise," *IEEE Trans. Commun.,* vol. COM-35, pp. 777-785, Aug. 1987.

[28] N. C. Beaulieu and C. Leung, "Optimal detection of hard limited data signals in different noise environments," *IEEE Trans. Commun.,* vol. COM-34, pp. 619-622, June 1986.

[29] N. C. Beaulieu and C. Leung, "On the performance of three suboptimum detection schemes for binary signalling," *IEEE Trans. Commun.,* vol. COM-33, pp. 241-245, March 1985.

[30] V. Milutinovic, "Suboptimum procedure based on the weighting of partial decisions," *IEE Electron. Lett,,* vol. 16, pp. 237-238, March 1980.

[31] V. Milutinovic, "Comparison of three suboptimum detection procedures," *IEE Electron. Lett,,* vol. 16, pp. 681-683, Aug. 1980.

[32] V. Milutinovic, "Performance comparison of three suboptimum detection procedures in real environment," *IEE Proc.,* vol. 131, pp. 341-344, Mar 1984.

[33] J. F. Hayes, *Modeling and Analysis of Computer Communications Networks* New York: Plenum, 1984, ch. 8.

[34] P. A. Wintz and E.J. Luecke, "Performance of Optimum and Suboptimum Synchronizers," *IEEE Trans. Commun.,* vol. COM-17, pp. 380-389, June 1969.

[35] A. L. McBride and A. P. Sage, "Optimum Estimation of Bit Synchronization," *IEEE Trans. on Aerosp. Electron. Syst.,* vol. AES-5, pp. 525-536, May 1969.

[36] G. Ascheid and H. Meyr, "Maximum likelihood detection and synchronization by parallel digital signal processing," *Globecom Conference Proc.* pp. 32.2.1-5, 1984.

[37] H.L. Van Trees, *Detection Estimation, and Modulation Theory, Part 1.* New York: Wiley, 1968.

[38] M. C. Schwartz and L. Shaw, *Discrete Spectral Analysis, Detection, and Estimation,* New York: McGraw-Hill, 1975.

[39] W. R. Bennet and J. R. Davey, *Data Transmission,* New York: McGraw-Hill, 1965, ch. 5.

[40] S. E. Nader and L. F. Lind, "Optimal data transmission filters," *IEEE Trans. on Circuits and Systems*, vol. CAS-26, pp. 36-45, Jan. 1979.

[41] K. H. Feher, *Digital Communications.* Englewood Cliffs: Prentice Hall, 1983, ch. 3.

[42] U. Mengali and G. Pirani, "Jitter accumulation in PAM systems using PLL's or resonant circuits for timing extraction," *National Telecommunications Conference Proc.*, pp. 38.1.1-5, 1980.

[43] S. Siu et al, "Analysis of the maximum tolerable input jitter in a chain of digital regenerators," *Global Communications Conference Proc.*, pp. 30.5.1-7, 1986.

[44] N. A. D'Andrea, and U. M. Mengali, "A simulation study of clock recovery in QPSK and 9QPRS systems", *IEEE Trans. Commun.,* vol. COM-33, pp 1139-1142, Oct. 1985.

[45] H. Meyr and L. Popken, "Phase acquisition statistics for phase locked loops", *IEEE Trans. Commun.*, vol. COM-28, pp. 1365-1372, Aug. 1980.

[46] N. A. D'Andrea, and U. M. Mengali, " Nearly optimal prefiltering in clock recovery", *IEEE Trans. Commun.,* vol. COM-34, pp. 1081-1088, Oct. 1986.

[47] J. C. Y. Huang, K. Feher, and M. Gendron, "Techniques to generate ISI and jitter-free bandlimited Nyquist signals and a method to analyze jitter effects", *IEEE Trans. Commun.,* vol. COM-27, pp. 1700-1711, Nov. 1979.

[48] W.C. Lindsey and Chak Ming Chie, "A survey of digital phase locked loops", *Proc. IEEE,* vol. 69, pp. 410-431.

[49] W.C. Lindsey, *Synchronization Systems in Communication and Control*, Englewood Cliffs: Prentice Hall, 1972, ch. 4.

[50] S.J. Orfandis, *Optimum Signal Processing*, New York: MacMillan, 1985, ch. 1.

[51] M. L. Honig and D. G. Messerschmitt, *Adaptive Filters: Structures, Algorithms, and Applications,* Boston: Kluwer, 1984, ch. 3.

[52] W. C. Y. Lee, *Mobile Communications Engineering*, New York: McGraw-Hill, 1982, ch. 6.

[53] F. M. Gardner, "Hangup in phase-lock loops", *IEEE Trans. Commun.*, vol. COM-25, pp. 1210-1214, Oct 1977.

[54] F. M. Gardner, "Equivocation as a cause of PLL hangup", *IEEE Trans. Commun.*, vol. COM-30, pp. 2242-2243, Oct 1977.

[55] W. C. Lindsey and J. Seyl, "Characterization and measurement of phase locked loop performance", *National Telecommunications Conference Proc.*, pp. E2.5.1-6, 1982.

[56] J. F. Hayes, *Modeling and Analysis of Computer Communications Networks,* New York: Plenum, 1984, appendix B.

[57] S. G. Brophy and D. D. Falconer, "Investigation of synchronization parameters in a digital subscriber loop transmission system", *IEEE JSAC.,* vol. SAC-4, No. 4 , pp. 1312-1316, Nov. 1986.

[58] IMSL Library, vol 2, IMSL inc., 1985, ch. D.

[59] IMSL Library, vol 2, IMSL inc., 1985, ch. G.

[60] M. J. Levin, "Generation of a sampled Gaussian time series having a specified correlation function", *IRE Trans. on Information Theory*, vol. IT-6, pp. 545-548, Dec. 1960.

[61] J. E. Freund, *Mathematical Statistics*, Englewood Cliffs: Prentice-Hall, 1971, ch. 9.

[62] A. Antoniou, *Digital Filters: Analysis and Design*, New York: McGraw-Hill, 1979, ch. 9.

[63] Texas Instruments Inc., *TMS32020 Users Guide*, 1986.

[64] Hewlett Packard Co., *Product Note/Specification Guide, HP5371A Frequency and Time Interval Analyzer*, Nov. 1987.

[65] P. Z. Peebles, *Probability, Random Variables and Random Signal Principles,* New York: McGraw-Hill, 1987, appendix F.

Appendix A

Chapter 2 Simulation Methods

The simulation methods for determining the timing error variances of both the ML and HL synchronizers are described in this appendix.

The random data a_k are generated by a uniform $(0, 1)$ random number generator [58] where $a_k = 1$ if the random number is $\geq .5$, otherwise $a_k = -1$. At the receiver the m stored sampled values of $x(t)$ are calculated directly, and the samples are evenly spaced over the bit duration. The delay ϵ associated with the transmitted data can assume one of ten evenly spaced locations in the bit interval, and a small delay is additionally added to the transmitted data to simulate the fractional sample delay. This small delay is uniformly distributed over $(-T_s/2, T_s/2)$ where $1/T_s = m/T$ is the sampling rate. The statistically independent Gaussian noise samples are simulated with a Gaussian random number generator [58].

The HL and ML synchronizer programs both evaluate all m possible locations of $\hat{\epsilon}$ serially to determine $\hat{\epsilon}_{ML}$ and $\hat{\epsilon}_{HL}$. The timing error magnitude $|e| = |\hat{\epsilon} - \epsilon|$ is reduced mod$(.5T)$ since timing errors cannot exceed $\pm .5T$. After each iteration a new delay value $\hat{\epsilon}$ is selected (including a new

fractional sample delay), $K+1$ statistically independent data values a_j are generated, and the search for $\hat{\epsilon}_{ML}$ and $\hat{\epsilon}_{HL}$ is repeated.

The program is numerically intensive and the simulations require a large amount of computer time. The simulation time is $\approx KNm \times 10^{-3}$ seconds per simulation point on a Sun 3/280 computer with a 20 MHz floating point coprocessor, where N is the number of iterations. The variance measurements are based on $N = 25 \times 10^3$ iterations. A comment on the accuracy of the measurements should be made. The square signalling pulse variance measurements do not converge as quickly as do the other signalling pulses. This can be attributed to the probability of the event $K+1$ consecutive data symbols a_j being equal (either all $1's$ or all $-1's$) in an iteration. For this event, no synchronization information is available since the received signal is a constant value plus noise. The resulting estimate (either $\hat{\epsilon}_{ML}$ or $\hat{\epsilon}_{HL}$) is uniformly distributed over the bit interval $(0, T)$. Therefore standard variance estimation confidence interval tests [61] based on equal variance samples are not appropriate, since the variance of the timing error estimate for each iteration is dependent on the data pattern.

Appendix B

Comments on the Jitter Variance Analysis in Reference [18]

In this appendix, it is shown that the zero (threshold) crossing synchronizer model developed by Saltzberg [18] and the discrete time synchronizer model of chapter 3 are equivalent. To aid the comparison, the symbols used in [18] and the symbols of chapter 3 are summarized in table B.1.

In [18] the output of the phase detector is a series of impulses with magnitude proportional to the difference between the location of the zero crossing and the synchronizers estimate of the zero crossing[1]

$$e(t) = \sum_k K_1(\alpha_k - \gamma)\delta(t - kT) \tag{B.1}$$

where it is assumed that the effects of the variation of the position of the impulse in the bit interval are negligible [18, page 599]. A two term Taylor

[1] Equation (21) of [18].

Ref. [18]	Chapter 3	Comment
$f(t)$	$x(t)$	signalling pulse
a_k	a_k	zero mean binary data
$n(t)$	$\eta(t)$	filtered noise
βT	ϵ_k	delay parameter
γT	$\hat{\epsilon}_k$	synchronizer estimate of the delay
$\alpha_k T$	$\epsilon_k + n_k + \tau_k$	zero crossing relative to local clock
K_1	$2K_{PD}T$	phase detector gain
K_2	K_{NCO}	VCO (NCO) gain
$T/2$	t_o	
$e(t)$	w_k	phase detector output
d_k	b_k	transition detector $= 0.5(1 - a_k a_{k+1})$
$G_2(\omega)$	$G(z)$	filter transfer function
$g_2(k)$	g_k	transfer function impulse response
b	$1/c$	slope of the zero crossing

Table B.1: Comparison of the symbols used in ref. [18] and the symbols of chapter 3.

series expansion of (B.1) about the nominal zero crossing $(\alpha_k - \beta)$ yields[2]

$$e(t) = K_1 \sum_k d_k \left\{ \beta - \gamma + \frac{a_k}{bT} [\sum_{i \neq 0,-1} a_{k-i} f(iT + T/2) + n(t)] \right\} \delta(t - kT) \tag{B.2}$$

Using table B.1, substitution of chapter 3 symbols into (B.2) yields

$$w_k = K_{PD} 2 b_k \{ \epsilon_k - \hat{\epsilon}_k + c \cdot a_k \sum_{i \neq 0,1} a_{k+i} x(t_o - iT) + c \cdot a_k \eta(kT) \} \tag{B.3}$$

$$= K_{PD} 2 b_k \{ \epsilon_k - \hat{\epsilon}_k + \tau_k + n_k \} \tag{B.4}$$

[2]Equation (30) of [18].

where w_k is the phase detector output sequence from figure 3.5, τ_k and n_k are defined in (3.15) and (3.13) respectively. Since $\eta(t)$ is stationary we may define the discrete noise disturbance n_k to be either $c \cdot a_k \eta(kT)$ (from (B.3)) or as $c \cdot a_k \eta(kT + t_o + \epsilon_k)$ as it is defined in (3.13) (again assuming a nearly constant ϵ_k). We use the latter definition as it is more consistent with the two term Taylor series approximation about the nominal zero crossing locations $t = kT + t_o + \epsilon_k$.

The analysis in [18] is done in two parts: first the mean (static) timing error $\bar{\gamma} - \bar{\beta}$ is determined, and then the timing error jitter variance is determined from the autocorrelation function of the zero mean component of the the output sequence $\gamma(n)$. We now show that the timing error estimate sequence $\gamma(n)$ used in the derivation of the mean timing error and the timing jitter variance in [18] is equivalent to the timing error estimate sequence $\hat{\epsilon}_k$ from (3.21) used in the jitter variance analysis of chapter 3.

In [18], the input to the filter $G_2(w)$ is[3]

$$e_2(n) = 2d_n \left\{ \beta - \sum_{k=-\infty}^{n} g_2(n-k) e_2(k) + \frac{a_n}{b} [\sum_{k \neq 0,-1} f(k + 1/2) + n(t)] \right\} \tag{B.5}$$

where

$$e_2(t) = \sum_n e_2(n) \delta(t - nT) \tag{B.6}$$

The synchronizer delay estimate $\gamma(n)$ is the convolution of $e_2(n)$ with the

[3]Equation (40) of [18], where $T = 1$, and we ignore the pilot tone recovery parameter τ.

impulse response $g_2(n)$

$$\begin{aligned}\gamma(n) &= \sum_{k=-\infty}^{n} g_2(n-k)e_2(k) \\ &= \sum_{k=-\infty}^{n} g_2(n-k)2d_n \left\{ \beta + + \frac{a_n}{b}[\sum_{k\neq 0,-1} f(k+1/2) + n(t)] - \gamma(k) \right\}\end{aligned} \tag{B.7}$$

From table B.1, (B.7) may be expressed using chapter 3 symbols

$$\hat{\epsilon}_n = \sum_{k=-\infty}^{n} g_{n-k} 2b_n \{\epsilon_k + \tau_k + n_k - \hat{\epsilon}_k\} \tag{B.8}$$

where n_k and τ_k are defined in (3.13) and (3.15) respectively. Equation (B.8) is equal to (3.21), where the latter is used to determine the timing jitter variance result of chapter 3.

It can be shown that for zero mean timing errors, the timing error variance analysis of [18] is identical to the chapter 3 analysis up until the evaluation of the autocorrelation function of the delay estimate sequence $\gamma(n)$[4]. Saltzburgs assumptions in the autocorrelation function are not stated explicitly. A careful study shows that if a narrow bandwidth synchronizer is used, a reasonable assumption is that the random variable b_k is uncorrelated with all other random variables within each expectation in (3.25). This assumption leads to a result consistent with [18][5]. This assumption

[4]Compare equation (3.24) with equation (55) of [18], ignoring the autocorrelation function due to pilot tone recovery in the latter.

[5]Equation (62) of [18].

is not made in the analysis of chapter 3 so the present analysis is more correct.

In the present work, a new analysis of the autocorrelation function (3.24) is carried out and the assumptions for the timing jitter variance analysis are explicitly stated. Simulations results and the analysis for the special case synchronizer in appendix E indicate the analysis in chapter 3 to be more accurate than the result presented in [18].

Appendix C

Autocorrelation of Additive Noise at the Output of a Nyquist Matched Filter

In this appendix the autocorrelation of filtered additive white Gaussian noise at the output of a Nyquist matched filter receiver is reviewed to justify assumptions made in the jitter variance analysis of chapter 3.

The autocorrelation function $R_\eta(\tau)$ of the filtered AWGN waveform $\eta(t)$ may be determined from the inverse Fourier transform of $S_\eta(f)$, the psd at the output of $H_R(f)$, the Nyquist receive low pass filter of figure 3.2. The psd of the noise at the filter output is

$$S_\eta(f) = \left(\frac{N_o}{2}\right)|H_R(f)|^2 \tag{C.1}$$

$$= \left(\frac{N_o}{2}\right)H_R(f)H_R^*(f) \tag{C.2}$$

where $N_o/2$ is the two sided psd of the AWGN. If $H_R(f)$ and $H_T(f)$ form a matched filter pair $H_T(f) = H_R^*(f)e^{j2\pi t_d}$, where t_d is a filter delay parame-

ter. Thus

$$S_\eta(f) = \left(\frac{N_o}{2}\right) H_R(f) H_T(f) e^{-j2\pi f t_d} \tag{C.3}$$

$$= \left(\frac{N_o}{2}\right) X(f) \tag{C.4}$$

where $X(f)$ is the Fourier transform of $x(t)$, the impulse response of the matched filter cascade. The autocorrelation function $R_\eta(\tau)$ is the inverse Fourier transform of (C.4)

$$R_\eta(\tau) = \left(\frac{N_o}{2}\right) x(\tau) \tag{C.5}$$

Since $x(\tau)$ satisfies the Nyquist criterion in (3.2), then from (3.70)

$$R_\eta(pT) = \begin{cases} R_\eta(0) & \text{for } p = 0 \\ 0 & \text{for } p = \pm 1, \pm 2, \ldots \end{cases} \tag{C.6}$$

Therefore the elements of a sequence obtained by sampling $\eta(t)$ at $t = kT + \zeta$, where ζ is any constant, are zero mean and uncorrelated. The power spectrum of such a sequence will be flat (white noise), with power equal to the variance σ_η^2 [50]. Since the uncorrelated samples have a Gaussian distribution, the samples are also independent [38].

Appendix D

Jitter Variance Analysis Assumptions

The assumptions stated in (3.29) and (3.30) are to be justified. A general analysis for both the additive noise and ISI disturbances proved elusive, and based on numerical results unnecessary. Therefore only the additive noise case is considered.

For the analysis to follow it is assumed that the delay parameter ϵ_k is a constant value. It is easy to confirm the following results for an arbitrary constant value of ϵ_k, but to simplify the notation we also assume $\epsilon_k = 0$. For a synchronizer operating in the presence of additive noise only (i.e. $\tau_k = \epsilon_k = 0$), the sequence γ_k from (3.22) is

$$\gamma_k = 2b_k n_k \tag{D.1}$$

Using (3.18) and (3.13) for b_k and n_k respectively, equation (D.1) may be written

$$\gamma_k = c(a_k - a_{k+1})\eta(kT + t_o) \tag{D.2}$$

$$= \quad c(a_k - a_{k+1})\tilde{\eta}_k \tag{D.3}$$

where $\tilde{\eta}_k \triangleq \eta(kT + t_o)$ is the sampled value of the additive noise at the output of the LPF_R of figure 3.2. Substituting (D.3) into (3.21), the sequence $\hat{\epsilon}_i$ is

$$\hat{\epsilon}_i = \sum_{l=1}^{\infty} g_l(1 - a_{i-l}a_{i-l+1})\{ca_{i-l}\tilde{\eta}_{i-l} - \hat{\epsilon}_{i-l}) \tag{D.4}$$

Note that the summation begins at $l = 1$ since it can be shown from (3.20) that $g_o = 0$.

The first assumption is (3.29)

$$E[a_i a_{i+1} a_j a_{j+1} \hat{\epsilon}_i \hat{\epsilon}_j] = R_{\hat{\epsilon}}(0)\delta(i - j) \tag{D.5}$$

For $i = j$, the result is straightforward. Since the elements of the sequence $\{a_k\}$ are statistically independent zero mean random variables, if $i \neq j$ a non-zero expectation requires the product $\hat{\epsilon}_i\hat{\epsilon}_j$ to contain at least one $a_{i+1}a_{j+1}$ term. From (D.4), $\hat{\epsilon}_i$ does not contain a_{i+1}, and for $\hat{\epsilon}_j$ to contain a_{i+1}

$$j > i \tag{D.6}$$

Similarly for $\hat{\epsilon}_j$ to contain a_{i+1}

$$i > j \tag{D.7}$$

Since it is not possible to satisfy (D.6) and (D.7) simultaneously, the expectation in (D.5) is zero if $i \neq j$.

The second assumption (3.29) is

$$E[a_i a_{i+1} \hat{\epsilon}_i \gamma_j] = 0 \tag{D.8}$$

The expectation in (D.8) may be non-zero only if the product $\hat{\epsilon}_i \gamma_j$ contains at least one a_{i+1} term. Using (D.3) in (D.8) we have

$$E[a_i a_{i+1} \hat{\epsilon}_i \gamma_j] = cE[a_i a_{i+1} \hat{\epsilon}_i (a_j - a_{j+1}) \tilde{\eta}_j] \tag{D.9}$$

From inspection of (D.4), it is apparent that $\hat{\epsilon}_i$ does not contain a_{i+1}. Therefore j must equal i or $i+1$ in (D.9) for a non-zero expectation. From (D.4), if $j = i$ or $i+1$ $\hat{\epsilon}_i$ cannot contain $\tilde{\eta}_j$. If it is assumed that the autocorrelation function of the additive noise at the output of the LPF_R satisfies (C.6) (for example if a Nyquist matched filter is used) sampled noise values $\tilde{\eta}_k$ are statistically independent. Hence the zero mean random variable $\tilde{\eta}_j$ is uncorrelated with the other terms within the expectation of (D.9), and therefore the expectation is zero.

Appendix E

Noise Jitter Analysis of a Special Case Pattern Jitter Compensation Synchronizer.

An expression for σ_{nj}^2 is to be derived for a special case pattern jitter compensation synchronizer, specifically the synchronizer used in the computer simulations of Chapter 3. It will be shown that the result presented in this analysis demonstrate the effect of the m bit delay in the feedback on the noise jitter variance.

Consider the synchronizer of figure 3.5 with loop filter $F(z) = 1$ and $K \triangleq K_{PD}K_{NCO}$ operating in the absence of ISI ($\tau_k = \hat{\tau}_k = 0$) and delay ($\epsilon_k = 0$). Timing adjustments are made according to

$$\hat{\epsilon}_{k+1} = \hat{\epsilon}_k + 2Kb_{k-m}(n_{k-m} - \hat{\epsilon}_{k-m}) \tag{E.1}$$

where $n_k = a_k c\eta(kT + t_o)$ is defined in (3.13). The sampled value of the noise $\eta(kT+t_o)$ is assumed to be a discrete white noise process. Justification for this assumption is given in appendix C. The mean value of (E.1) is zero,

and so the noise jitter variance σ_{nj}^2 is equal to the mean square value of (E.1)

$$\begin{aligned} E[\hat{\epsilon}_{k+1}^2] &= E[\hat{\epsilon}_k^2] + 4K\{E[b_{k-m}n_{k-m}\hat{\epsilon}_k] - E[b_{k-m}\hat{\epsilon}_{k-m}\hat{\epsilon}_k]\} \\ &\quad +4K^2\{E[b_{k-m}n_{k-m}^2] - 2E[b_{k-m}n_{k-m}\hat{\epsilon}_{k-m}] + E[b_{k-m}\hat{\epsilon}_{k-m}^2]\} \end{aligned} \tag{E.2}$$

Assuming stationarity, and substituting (3.18), (3.13), and (E.1) for b_k, n_k and $\hat{\epsilon}_k$ respectively, straightforward evaluation of the expectations in (E.2) yields the following

$$E[\hat{\epsilon}_{k+1}^2] = E[\hat{\epsilon}_k^2] = R_{\hat{\epsilon}}(0) \tag{E.3}$$

$$E[b_{k-m}\hat{\epsilon}_{k-m}\hat{\epsilon}_k] = E[b_{k-m}n_{k-m}\hat{\epsilon}_{k-m}] = 0 \tag{E.4}$$

$$2E[b_{k-m}\hat{\epsilon}_{k-m}\hat{\epsilon}_k] = R_{\hat{\epsilon}}(m) \tag{E.5}$$

$$4E[b_{k-m}n_{k-m}^2] = 2c^2\sigma_\eta^2 \tag{E.6}$$

$$2E[b_{k-m}\hat{\epsilon}_{k-m}^2] = R_{\hat{\epsilon}}(0) \tag{E.7}$$

Substituting (E.3)-(E.7) into (E.2) and rearranging yields the jitter variance σ_{nj}^2

$$\sigma_{nj}^2 = R_{\hat{\epsilon}}(0) = \frac{Kc^2\sigma_\eta^2}{\rho_{\hat{\epsilon}}(m) - K} \tag{E.8}$$

where $\rho_{\hat{\epsilon}}(m) = R_{\hat{\epsilon}}(m)/R_{\hat{\epsilon}}(0)$. To determine $R_{\hat{\epsilon}}(m)$ equation (E.1) is substituted into $E[\hat{\epsilon}_k\hat{\epsilon}_{k-m}]$

$$\begin{aligned} R_{\hat{\epsilon}}(m) &= E[\hat{\epsilon}_k\hat{\epsilon}_{k-m}] \\ &= E[\hat{\epsilon}_{k+1}\hat{\epsilon}_{k-m+1}] \end{aligned}$$

$$
\begin{aligned}
&= E[(\hat{\epsilon}_k + 2Kb_{k-m}(n_{k-m} - \hat{\epsilon}_{k-m}))\hat{\epsilon}_{k-m+1}] \\
&= R_{\hat{\epsilon}}(m-1) + 2KE[b_{k-m}n_{k-m}\hat{\epsilon}_{k-m+1}] \\
&\quad -2KE[b_{k-m}\hat{\epsilon}_{k-m}\hat{\epsilon}_{k-m}] \qquad \text{(E.9)}
\end{aligned}
$$

Since $E[b_{k-m}n_{k-m}\hat{\epsilon}_{k-m+1}] = 0$, and $2E[b_{k-m}\hat{\epsilon}_{k-m}\hat{\epsilon}_{k-m+1}] = R_{\hat{\epsilon}}(1)$, then (E.9) is

$$R_{\hat{\epsilon}}(m) = R_{\hat{\epsilon}}(m-1) - KR_{\hat{\epsilon}}(1) \qquad \text{(E.10)}$$

This procedure is repeated m times to generate the following set of equations

$$R_{\hat{\epsilon}}(m-p) = R_{\hat{\epsilon}}(m-p-1) - KR_{\hat{\epsilon}}(p+1) \quad p = 0, 1, \ldots, m-1 \qquad \text{(E.11)}$$

Solutions to $R_{\hat{\epsilon}}(m)$ for $m \leq 3$ are

$$\rho_{\hat{\epsilon}}(m) = \frac{R_{\hat{\epsilon}}(m)}{R_{\hat{\epsilon}}(0)} = \begin{cases} 1 & m = 0 \\ \frac{1}{1+K} & m = 1 \\ \frac{1-K}{1+K(1-K)} & m = 2 \\ \frac{1-K(1+K)}{1+K+K(1-K(1+K))} & m = 3 \end{cases} \qquad \text{(E.12)}$$

Note that since $\rho_{\hat{\epsilon}}(m) \leq 1$, from (E.8) the m bit delay will generally degrade the noise performance of the synchronizer. However from inspection of (E.12) for small values of K and m, $\rho_{\hat{\epsilon}}(m) \approx 1$, and the degradation will not be significant. This degradation in terms of mean square jitter can intuitively be explained by realizing that corrections for timing errors at the input are delayed m bits in the feedback, and therefore one would expect the mean square timing error to be larger than in the absence of delay.

Recall from (3.75)

$$\sigma_{nj}^2 = \frac{2c^2\sigma_\eta^2 2B_L T}{1 - 2B_L T} \tag{E.13}$$

Solving (E.8) and (E.13) for $2B_L T$ we find

$$2B_L T = \frac{K}{2\rho_{\hat{\epsilon}}(m) - K} \tag{E.14}$$

The bandwidth of the synchronizer $2B_L T$ may be also be evaluated using (3.40). For example, consider the first order $m = 0$ synchronizer. From (3.20)

$$G(z) = \frac{Kz^{-1}}{1 - z^{-1}} \tag{E.15}$$

From (3.35), the transfer function $H(z)$ is

$$H(z) = \frac{Kz^{-1}}{1 + (K-1)z^{-1}} \tag{E.16}$$

Substituting the magnitude squared frequency response

$$|H(\omega)|^2 = H(z)H(1/z)|_{z=e^{j\omega}} \tag{E.17}$$

into (3.40) and evaluating the integral yields

$$2B_L T = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2 \, d\omega \tag{E.18}$$

$$= \frac{K}{2 - K} \tag{E.19}$$

which is identical to (E.14) for $m = 0$.

Either (3.40) or (E.14) may be used to determine $2B_L T$, although the latter expression is generally easier to evaluate.

Appendix F

Chapter 3 Simulation Methods

The methods of computer simulation and numerical analysis used in chapter 3 are outlined in this appendix.

Numerical Integration

The numerical integration of (3.47) and (3.65) is done using an iterative integration routine, from the IMSL subroutine library [58].

Pulse Length Truncation

For simulation purposes, the pulse $x(t)$ must be truncated to a finite duration, with some justification given that the truncation is a reasonable approximation to the infinite duration pulse. To evaluate the effects of the pulse truncation, consider the mean square value of the ISI disturbance $R_I(0)$, which is defined in (3.51). For an infinite duration SRC pulse defined

by (3.81), the value of $R_I(0)$ from (3.63) is

$$R_I(0) = 2c^2 \left\{ \sum_{p=-1}^{1} (-1)^p \int_{-\infty}^{\infty} X(f) X\left(f - \frac{p}{T}\right) df - \sum_{i \in P} x^2(T/2 - iT) \right\} \tag{F.1}$$

where the equation has been simplified by noting that $x(t) = x(-t)$ and so $X(f) = X^*(f)$. Evaluation of (F.1) yields

$$R_I(0) = 2c^2\{1 - .5\alpha - \sum_{i \in P} x^2(T/2 - iT)\} \tag{F.2}$$

For an uncompensated synchronizer ($l = m = 0$), $R_I(0)$ may be found by substituting (3.81) into (F.2)

$$R_I(0)|_{l=m=0} = 2c^2\{1 - .5\alpha - 2x^2(T/2)\} \tag{F.3}$$

$$= 2c^2 \left\{ 1 - .5\alpha - \left(\frac{2cos(\alpha\pi/2)}{\pi(1-\alpha^2)} \right)^2 \right\} \tag{F.4}$$

Equation (F.4) clearly demonstrates the effect of α on the magnitude of the ISI disturbance. The value of $R_I(0)$ is maximum for $\alpha = 0$, and $R_I(0) = 0$ for $\alpha = 1$.

Figure F.1 compares $R_I(0)$ for the infinite duration pulse defined by equation (F.4) to that of various length symmetrically truncated pulses, calculated from (3.51). As expected the curves for the truncated pulses diverge from the infinite duration pulse for small values of α. The $14T$ duration pulse has a value of $R_I(0)$ greater than 98% that of an infinite duration pulse for all $\alpha > 0.1$, and therefore was chosen as the truncation length for the simulation.

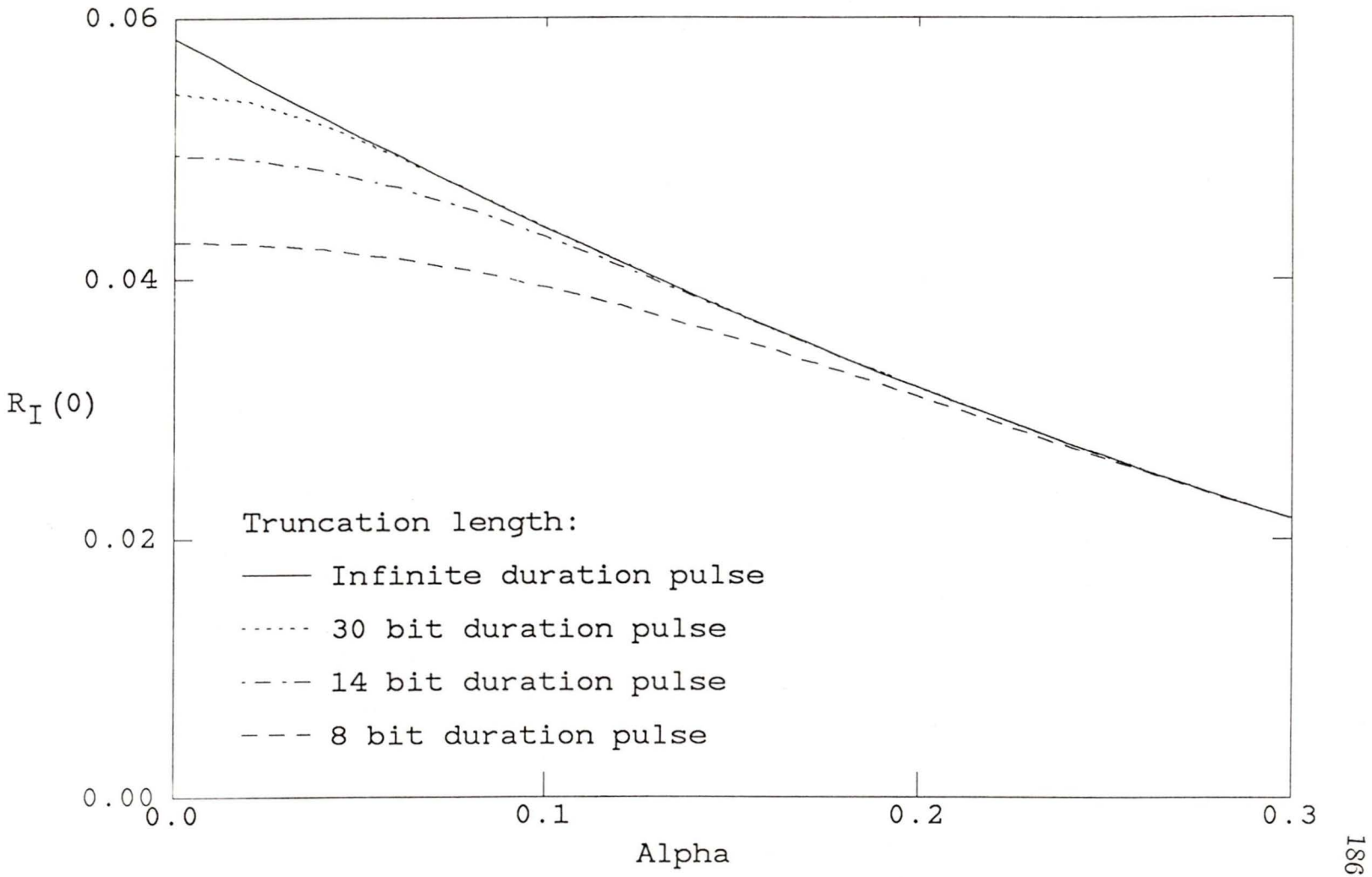


Figure F.1: Mean square value of the ISI disturbance for various length symmetrically truncated spectral raised cosine pulses.

For the seventh order data transmission filter, the signal is not strictly bandlimited, and therefore (3.63) does not apply. The value of $R_I(0)$ for the uncompensated synchronizer ($l = m = 0$) is determined from 3.51

$$R_I(0) = 2c^2 \sum_{i \neq 0,1} x^2(t_o - iT) \tag{F.5}$$

This summation must converge for any finite energy $x(t)$. A comparison of the simulation results for truncation lengths of $14T$ and $18T$ yielded no significant difference, and therefore it is assumed that the value of $R_I(0)$ for an $18T$ duration pulse is sufficiently close to the infinite duration pulse for these measurements.

Linear Simulation

The linear synchronizer model of figure 3.5 is simulated to confirm the analysis based on the linear assumptions. The recursive equation (3.19) is evaluated by Monte Carlo techniques to determine σ_e^2. The values of τ_k and $\hat{\tau}_k$ are calculated from (3.15) and (3.16) respectively. The noise disturbance n_k is generated according to (3.13) where the sequence $\eta(kT + t_o)$ are Gaussian distributed random numbers [59]. The noise samples are independent from bit to bit in accordance with (C.6).

Actual Simulation

For the actual simulation, the SRC pulse $x(t)$ is calculated directly from equation (3.81). The compensation values $\hat{\tau}_k$ are found by solving (3.8) using a root finding program and are stored in computer memory. The

noise $\eta(t)$ is simulated by filtering independent random Gaussian numbers with a finite impulse response (FIR) filter. A recursive filter is computationally more efficient, and a method to generate Gaussian sequences with a specified autocorrelation using a recursive filter is presented in [60]. The FIR filter is used here as it was found to be easier to approximate the required autocorrelation properties and frequency response. To simplify the simulations, a single filter is for all values of α. The filter is a truncated Fourier series approximation [62] to a SRC filter with $\alpha \approx 0.5$. The filter length is 100, with delay elements $T_d = .05T$. The discrete autocorrelation function of the noise sequence at the output of the the FIR filter is given by [50]

$$R_\eta(pT_d) = \frac{\sigma_i^2}{2\pi j} \oint_{u.c.} H_f(z) H_f(1/z) z^p dz \tag{F.6}$$

where $H_f(z)$ is the transfer function of the FIR filter, and σ_i^2 is the variance of a discrete white noise input. This autocorrelation function is illustrated in figure F.2. For simplicity, the magnitudes $R_\eta(pT_d)$ are joined together rather than representing the correlation values as impulses. The result indicates that the filter should be a reasonably good approximation to a continuous function. The filter is designed to ensure $R_\eta(\pm T) = 0$, in accordance with (C.6). From the additive noise jitter variance analysis, equation (3.71) indicates that the very small non-zero correlation at $\pm 2T$, and $\pm 3T$ will not alter $S_N(\omega)$, and therefore should not alter the variance measurements.

The phase detector determines the location of the zero crossing with an accuracy of $\pm 2.5 \times 10^{-5} T$. In practice a typical digital phase detector will

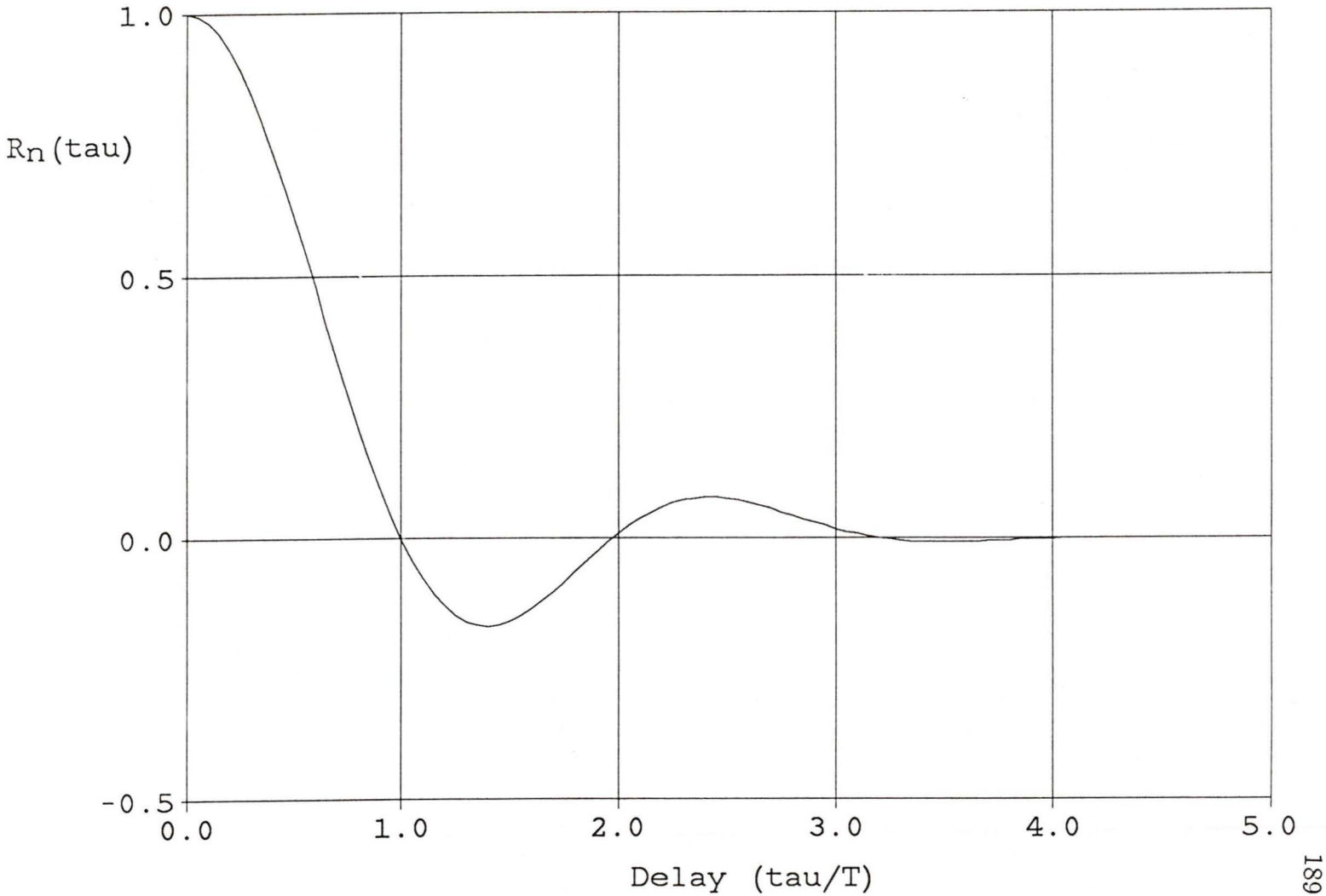


Figure F.2: Autocorrelation function of the simulation program's filtered noise.

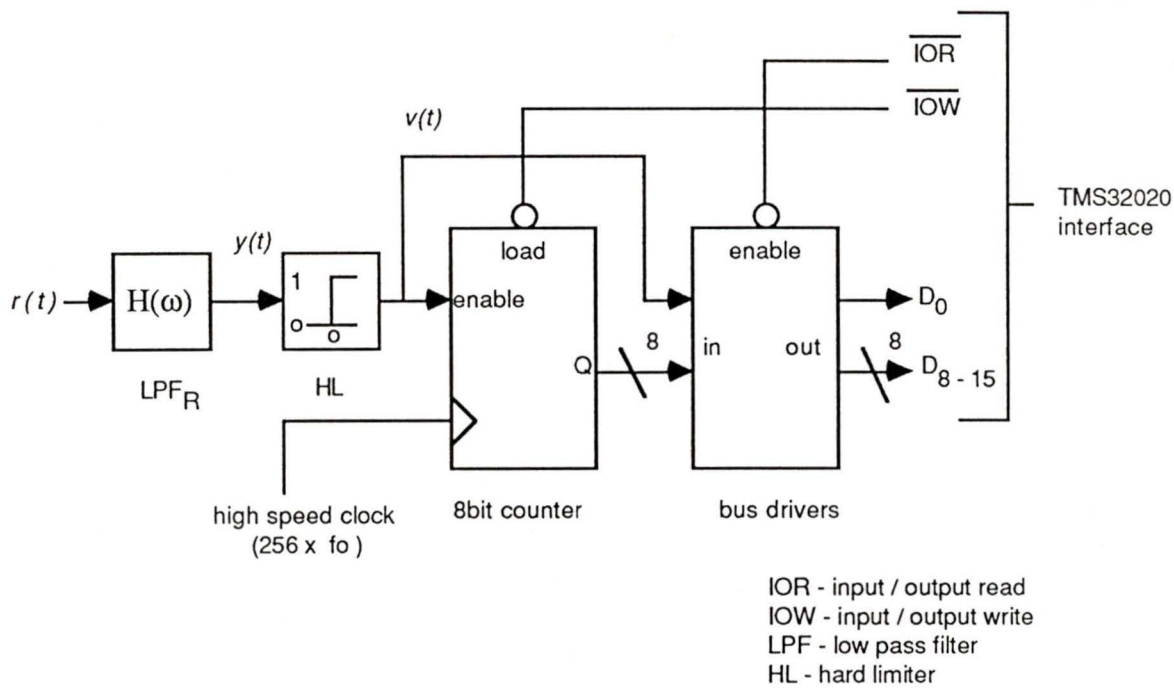
not have this resolution, but it is required here for comparison with the very small jitter values determined by the analysis and linear simulation. To measure the very small timing errors, it is assumed that the noise $\eta(t)$ is a constant value over T_d. This appears to be a reasonable assumption, since from figure F.2 $R_\eta(\pm T_d) = .995 R_\eta(0)$, indicating that a continuous random variable would have a nearly constant value over the T_d sample duration. Of course $x(t)$ (being deterministic) is not assumed to be constant over this duration. The number of iterations per point in the figures is 10^4. Measurements of some curves for 10^5 iterations did not alter variance measurements by more than .5%.

Appendix G

DSP Synchronizer Implementation

To test the synchronization techniques discussed in chapters 3 and 4, a prototype synchronizer that utilizes a Texas Instruments TMS32020 digital signal processing (DSP) chip [63] is used. The block diagram of the hardware interface to the TMS32020 is illustrated in figure G.1. The system is designed to operate nominally at 9600 bits per second.

The operation of the circuit is described in the following. The data eye pattern and noise at the output of the low pass filter (LPF) is hard limited. The hard limited binary output $(0, 1)$ is used to enable the 8 bit counter. The input/output write (I0W) line from the TMS32020 development system interface loads the counter with the most negative (1's complement) 8 bit number. After 256 cycles the input/output read (IOR) line (the data strobe) reads the counter value $c_k \in \{-128, \ldots, 127\}$ into the processor on the upper 8 bits of the data bus. The estimated binary data value $\hat{a}_{k+1}$ is read in on the least significant bit of the data bus. The value of the count



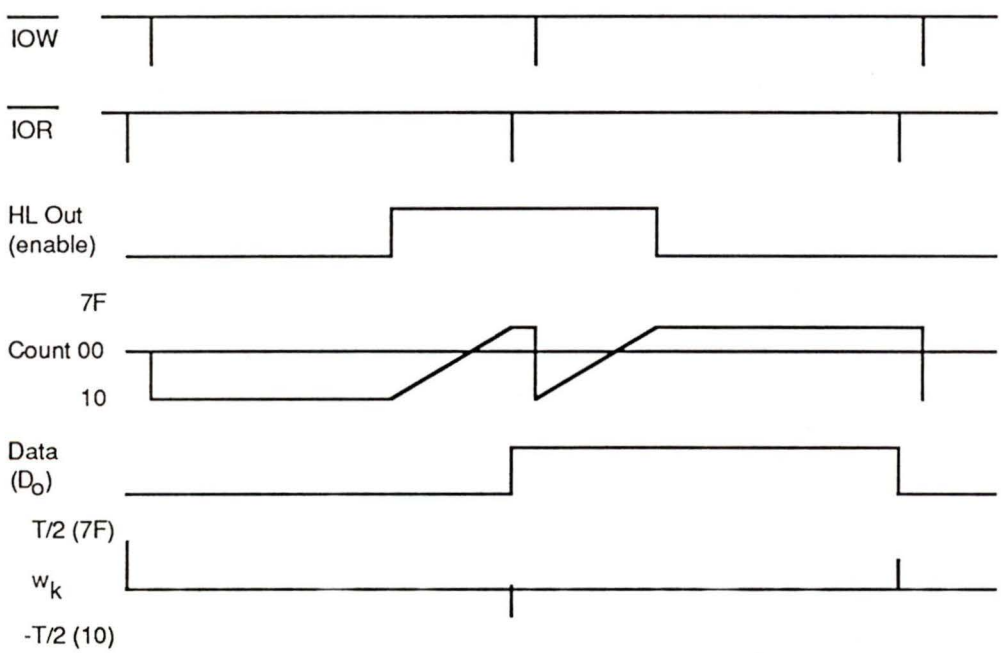


Figure G.1: Block diagram and timing waveforms of the hardware interface to the TMS32020 development system.

c_k is a function of the duration the hard limited binary data output enables the counter. An internal DSP algorithm is used to convert c_k into the phase detector output w_k

$$w_k = \hat{a}_k b_k c_k \qquad w_k \in \{-128, \ldots, 127\} \tag{G.1}$$

where $\hat{a}_k \in \{-1, 1\}$, and $b_k = 1 - \hat{a_k}\hat{a}_{k+1}$ is the data transition detector. It is readily apparent that the hardware complexity of the interface is considerably simpler than an analog to digital converter.

The DSP synchronization algorithm uses a discrete phase control to generate the IOW and IOR strobe pulses. The resolution of the phase control is $\pm T/128$. The block diagram of the internal synchronizer algorithm is illustrated in figure G.2. The PD measured error w_k is added to a remainder term r_k, and the resulting sum v_k is scaled by the loop gain constant K

$$u_k = Kv_k \tag{G.2}$$

$$= K(w_k + r_k) \tag{G.3}$$

The synchronizer delay estimate $\hat{\epsilon}_k$ is

$$\hat{\epsilon}_k = \text{sgn}(u_k)\lfloor |u_k| \rfloor \tag{G.4}$$

where $\lfloor(\cdot)\rfloor$ denotes the smallest integer less than $(\cdot)$. The integer output $\hat{\epsilon}_k$ determines the number of cycles to add or delete from the nominal 128 cycles before the next data sample. The new remainder $r_{k+1} = v_k \text{mod}(1/K)$ is the fractional portion of u_k which is truncated from the output by the $\lfloor(\cdot)\rfloor$ operation. It should be noted that choosing $K = 1/2^i$ where i is any

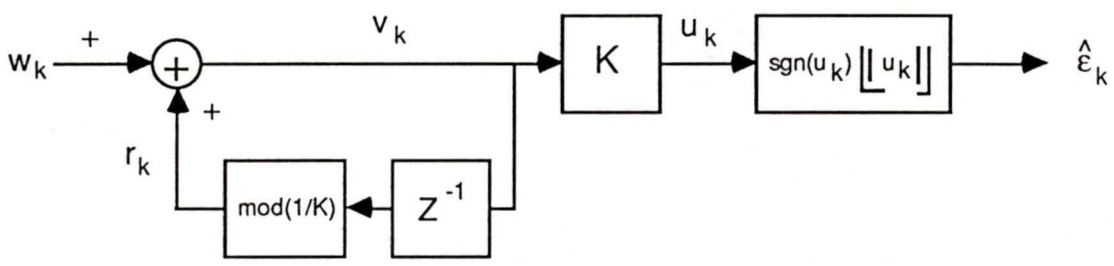


Figure G.2: Block diagram of the DSP internal synchronizer implementation.

integer greatly simplifies the implementation. Multiplication by K may be achieved by simply shifting an internal register of the processor, and the mod$(1/K)$ and $\lfloor(\cdot)\rfloor$ operations are accomplished internally with logical AND operations. In fact the TMS32020 internal multiplier was not used in this implementation. Therefore this synchronizer may be implemented using most standard microprocessors or microcontrollers, or using programmable logic arrays.

The hardware to estimate the timing error e_k for jitter variance and acquisition measurements is illustrated in figure G.3. The 8 bit binary counter is clocked by a high speed ($f = 256/T$) clock synchronized to the incoming data. The IOR synchronizer data strobe pulse is aligned in time such that the 8 bit flip flop latches the counter output at the count value 00 HEX when the timing error $e_k = 0$. If the IOR strobe is late, the counter output is positive. If the IOR strobe is early the output is negative. The digital to analog converter (DAC) translates the fluctuations in IOR strobe locations into an analog waveform. The RMS value of the DAC output is the RMS timing jitter (within an appropriate scaling constant). This technique of phase measurement is described in [8], and commercial test equipment has recently become available to measures timing errors using this technique [64].

The lock detect output (figure G.3) for acquisition measurements decodes the upper 5 bits of the 8 bit flip flop output to determine when the timing error is less than the specified $\pm 1/32T$.

The minimum measurable timing jitter variance is the mean square value of a random variable uniformly distributed over the minimum phase

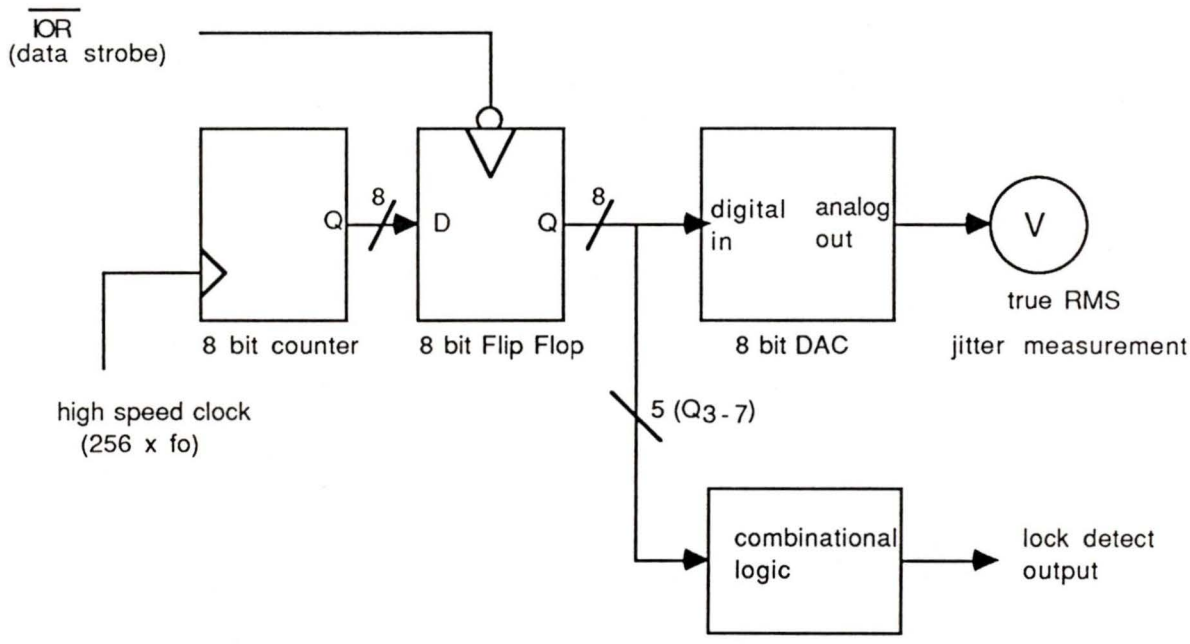


Figure G.3: Block diagram of the jitter measurement and lock detection test hardware.

adjustment duration [65].

$$\{\sigma_e^2/T^2\}_{min} = \frac{(1/128)^2}{12} = 5.08 \times 10^{-6} \tag{G.5}$$

To ensure that the lower limit does not introduce significant errors into the measured data, it is assumed that timing jitter variances $\sigma_e^2 \leq 10^{-5}$ are not accurate.

Appendix H

Chapter 4 Simulation Methods

The simulation techniques used in chapter 4 are discussed in this appendix.

Signalling Pulse Shape and Noise Filtering

It is required to simulate a spectral raised cosine (SRC) matched filter system with $\alpha = 1.0$. Since ISI is not a significant concern for a large excess bandwidth, the signalling pulse $x(t)$ is symmetrically truncated to a duration $4T$ to reduce computation. The number samples per bit is 32.

The filtered AWGN $\eta(t)$ is approximated by filtering zero mean independent Gaussian random numbers with a length 64 nonrecursive filter. The filter delay elements delay elements are $T_d = T/32$. The filter is an approximation to the LPF_R shown in figure 4.1. The filter coefficients are the sampled values of the impulse response of a SRC ($\alpha = 1.0$) matched filter receiver, often referred to as a $\sqrt{\alpha}$ filter since the magnitude of the frequency response is the square root of the overall SRC frequency response. The LPF_R filter is symmetrically truncated about the main lobe.

Synchronizer Description

As mentioned in chapter 4, the phase detector (PD) is implemented using a counter arrangement. The phase detector counts the NS (nominally $NS = 32$) hard limited signal plus noise samples $h_i \in \{-1, 1\}$ over the bit interval. The PD timing error w_k is

$$w_k = a_k b_k \frac{T}{NS} \sum_{i=1}^{NS} h_i \tag{H.1}$$

where $\hat{a}_k$ is the estimated data value, and $b_k = 1 - \hat{a}_k \hat{a}_{k+1}$ is the transition detector output. Timing adjustments are made according to

$$\hat{\epsilon}_k = \hat{\epsilon}_k + K_k w_k \tag{H.2}$$

where the value of K_k is selected according to the adaptive algorithm described in chapter 4. It should be noted that the timing error adjustment are made in discrete steps of $\pm nT/32$. The fractional portion of $K_k w_k$ which is truncated by the discrete timing adjustment operation is stored and added to the phase detector output w_{k+1}. This is discussed in more detail in the synchronizer hardware implementation description in appendix G.

Jitter Variance Measurement

The jitter variance σ_e^2 is determined via Monte Carlo techniques. The number of iterations per measurement is 2×10^4. The data sequence $\{a_j\}$ is a psuedorandom sequence (length = 255). For comparison with the hardware measurements, it was required to increase the resolution of the PD beyond

$\pm T/32$. The hardware synchronizer PD has a resolution of $\pm T/256$. To measure very small jitter variances, the zero crossing location is determined using linear interpolation. The zero crossing is interpolated from the two sampled values of signal plus noise that straddle the zero crossing. This is a compromise to avoid increasing the number of samples per bit. This increased resolution phase detector is not required for acquisition measurements.

Acquisition Measurement

The initial delay ϵ at the beginning of each acquisition measurement is assumed to be a constant value and is uniformly distributed over the bit interval $(-T/2, T/2)$.

For the adaptive synchronizer, the initial value of the loop gain is $K_0 = K^{(i)}$, where i is selected according to the steady state probability density Q_i, defined in (4.14). It is assumed that prior to synchronization, hard limited noise is present for a sufficient period of time to ensure the steady state conditions apply. The probabilities P_o, P_+, and P_- are chosen according to (4.21), (4.22), and (4.23).

The acquisition time (T_{acq}) distribution measurements are based on 10^3 acquisition tests. T_{acq} is defined here to be the time required to reduce the synchronizer error e_k to within $|e_k| \leq T/32$.

VITA

Surname: **OGMUNDSON** Given Names: **PATRICK GERARD**

Place of Birth: Port Alberni, B. C. Date of Birth: May 7, 1960

Educational Institutions Attended, with Dates of Entering and Leaving:

UNIVERSITY OF VICTORIA, B.C.	1986 to 1988
LAKEHEAD UNIVERSITY, THUNDER BAY ONT.	1984 to 1986
BRITISH COLUMBIA INSTITUTE OF TECHNOLOGY	1978 to 1980

Degrees, Diplomas, Etc., Awarded, with Dates and Names of Institutions:

B.Eng.	1986	Lakehead University.
Diploma of Technology	1980	B. C. Institute of Technology.

Honours and Awards:

NSERC Postgraduate Scholarship, 1988

Lakehead University Course Scholarship, 1985

Publications:

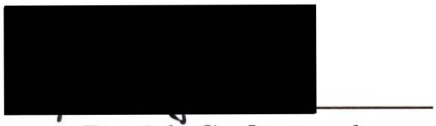
1. Patrick G. Ogmundson and Peter F. Driessen, "Digital Bit Synchronizer with Pattern Jitter Compensation", *IEEE Pacific Rim Conference on Computers, Communications and Signal Processing*, Victoria B. C., pp. 275-278, June 1987.

PARTIAL COPYRIGHT LICENSE

I hereby grant the right to lend my thesis to users of the University of Victoria Library, and to make single copies only for such users or in response to a request from the Library of any other university, or similar institution, on its behalf or for one of its users. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by me or a member of the University designated by me. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Title of Thesis:

Digital Bit Synchronization of Hard Limited Binary Data



Patrick G. Ogmundson

September 16, 1988

ISBN 0-315-50180-4