

Copyright © 1986, by the author(s).  
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

PARTIAL RESPONSE CODING IN DIGITAL SUBSCRIBER  
LOOPS

by

Nan-Sheng Lin

Memorandum No. UCB/ERL M86/79

29 September 1986

COUNT PAGE

PARTIAL RESPONSE CODING IN DIGITAL SUBSCRIBER LOOPS

by

Nan-Sheng Lin

Memorandum No. UCB/ERL M86/79

29 September 1986

ELECTRONICS RESEARCH LABORATORY

College of Engineering  
University of California, Berkeley  
94720

TIME PAGE



PARTIAL RESPONSE CODING IN DIGITAL SUBSCRIBER LOOPS

by

Nan-Sheng Lin

Memorandum No. UCB/ERL M86/79

29 September 1986

ELECTRONICS RESEARCH LABORATORY

College of Engineering  
University of California, Berkeley  
94720

# Partial Response Coding in Digital Subscriber Loops

Ph. D.

Nan-Sheng Lin

Department of EECS

David C. Hodges

Chairman of Committee

## Abstract

In this work, *modified duobinary partial response coding (PRC)* for *digital subscriber loops (DSL)* was studied and the design issues concerning algorithms and implementation of echo cancellation, equalization, and timing recovery were addressed. We evaluated the performance of the modified duobinary system and showed that modified duobinary PRC offers a desirable trade-off between system performance and implementation complexity.

In addition, the timing jitter problem was studied. We showed that an interpolation technique used in conjunction with modified duobinary coding can greatly relax the maximum allowable jitter in a digital subscriber loop system. This readily facilitates the design of a digital phase-locked loop.

The performance of modified duobinary PRC in a digital subscriber loop subsystem with timing jitter was measured experimentally. The results were in good agreement with computer simulations.

## ACKNOWLEDGEMENT

I would like to express my deepest appreciation to Professor Hodges for his thoughtful guidance and continuous support during the course of my Ph. D. studies. I am also very grateful to Professor Messerschmitt for his special assistance and advice. It has been my privilege to be able to participate in this project under their guidance. I also wish to thank Dr. Jeremy Tzeng for many suggestions and technical assistance.

The help of many fellow graduate students was very important in making my research effort successful. In particular, the fruitful discussions with Graham Brand is acknowledged. Special thanks goes to Steve Lewis for his valuable suggestions in my thesis writing, and Joey Doernberg for many helpful assistances.

Finally, I would like to express my hearty gratitude to my parents, my wife, my brothers, and my sisters for their endless encouragement, inspiration, and patience. Without them, this work wouldn't have been possible.

This research was supported by grants from Advanced Micro Devices, Bell Communications Research, Fairchild Semiconductor, Harris Semiconductor, National Semiconductor, and Intel Corporation, with matching grants from the University of California's MICRO program.

## TABLE OF CONTENTS

<b>1. INTRODUCTION .....</b>	<b>1</b>
1.1. Evolution of the integrated services digital networks .....	1
1.2. Digital subscriber loops .....	2
<b>2. SYSTEM OVERVIEW .....</b>	<b>6</b>
2.1. Impairments .....	6
2.1.1. White Noise .....	6
2.1.2. Crosstalk .....	6
2.1.3. Impulse noise .....	10
2.1.4. Echo .....	10
2.1.5. Intersymbol Interference .....	17
2.2. System Description .....	27
2.2.1. Line Coding and Pulse Shaping .....	27
2.2.1.1. Spectrum shaping .....	28
2.2.1.2. Timing information .....	31
2.2.1.3. Realization complexity .....	31
2.2.1.4. Length of echo path impulse response .....	32
2.2.2. Filtering and Equalization .....	32
2.2.2.1. Optimum linear equalizer .....	43
2.2.2.2. Slightly nonlinear receiver - DFE .....	47
2.2.2.3. Practical DFE receiver .....	54
2.2.3. Echo Canceller .....	58
2.2.3.1. Linear Echo Canceller .....	59
2.2.3.2. Nonlinear Echo Canceller .....	62
2.2.3.3. Adaptation speed and residual error considerations .....	67
<b>3. PARTIAL RESPONSE CODING .....</b>	<b>74</b>
3.1. Partial Response Coding .....	74
3.1.1. Spectrum shaping by the system polynomial F(D) .....	82
3.1.2. Decoding .....	84
3.1.3. Precoding .....	84
3.2. Modified Duobinary PRC .....	88
<b>4. NOISE PERFORMANCE OF THE MODIFIED DUOBINARY PRC .....</b>	<b>97</b>
4.1. Comparison of Line Codes with zero-forcing LE Design .....	97
4.1.1. Line codes .....	97
4.1.1.1. Biphase .....	98
4.1.1.2. AMI .....	98
4.1.1.3. Modified Duobinary (MDB) .....	98
4.1.1.4. Modified Modified Duobinary (MMDB) .....	102
4.1.1.5. MMS43 .....	102
4.1.1.6. DI43 .....	102
4.1.2. Assumptions .....	103
4.1.2.1. Transmitted Data Stream .....	104
4.1.2.2. Transmitted Pulse .....	104

4.1.2.3. Baud Rate .....	105
4.1.2.4. Line .....	105
4.1.2.5. Equalized Pulse .....	105
4.1.2.6. Transmit Filter .....	109
4.1.3. Comparative Performance .....	113
4.1.3.1. Probability of Error .....	113
4.1.3.2. Near-End Crosstalk .....	116
4.1.3.3. White Noise .....	119
4.1.3.4. Foreign Crosstalk .....	122
4.1.3.5. Transceiver Range .....	123
4.2. Comparison of Line Codes with Optimum DFE Design .....	125
4.2.1. Assumptions .....	125
4.2.1.1. Transmitted Data Stream .....	126
4.2.1.2. Transmitted Pulse .....	126
4.2.1.3. Baud Rate .....	126
4.2.1.4. Transmit Filter .....	126
4.2.1.5. Line .....	127
4.2.1.5.1. Line 1 .....	127
4.2.1.5.2. Line 2 .....	127
4.2.1.5.3. Line 3 .....	127
4.2.1.5.4. Line 4 .....	130
4.2.1.6. Equalized Pulse .....	131
4.2.2. Comparative Performance .....	132
4.2.2.1. Near-End Crosstalk .....	132
4.2.2.2. White Noise .....	134
4.2.2.3. Foreign Crosstalk .....	135
4.2.3. Impact on Echo Cancellation .....	135
4.2.4. Number of Echo Canceler Taps .....	136
4.2.5. MDB Implementation Alternatives .....	138
4.3. Conclusions .....	143
<b>5. TIMING RECOVERY .....</b>	<b>144</b>
5.1. Optimum Sampling Phase .....	146
5.2. Conventional Timing Recovery Techniques .....	148
5.2.1. Maximum-likelihood timing recovery .....	149
5.2.2. Spectral line method .....	150
5.2.3. Threshold-crossing phase detector .....	153
5.2.4. Sampled-derivative detector .....	153
5.2.5. Modified sampled-derivative detector .....	157
5.3. Low Sample-rate Timing Recovery Techniques .....	159
5.3.1. Wave difference method .....	159
5.3.2. Baud-rate sampling .....	167
5.3.2.1. Least mean-square timing recovery technique .....	167
5.3.2.2. Mueller and Muller's timing recovery method .....	171
<b>6. TIMING JITTER .....</b>	<b>181</b>

6.1. Digital phase-locked loop .....	181
6.2. PROBLEM AND SOLUTION FORMULATION .....	186
6.2.1. Problem formulation .....	187
6.2.2. Partial Response Coding (PRC) .....	190
6.2.3. Interpolation .....	194
6.3. Extension to the memory-based echo canceller .....	195
6.4. Simulation Results .....	197
6.5. Hardware Implementation .....	203
6.6. Experimental realizations .....	205
6.6.1. System architecture .....	205
6.6.2. State diagram .....	206
6.6.3. Echo canceller .....	206
6.6.4. System controller and Digital processor .....	210
6.6.5. Digital PLL and timing generation .....	212
6.6.6. Coder, pulse former, and filters .....	216
6.6.7. Analog cancellation .....	216
6.6.8. Experimental results .....	216
6.7. Conclusions .....	219
7. CONCLUSION .....	220

# CHAPTER 1

## Introduction

### 1.1. Evolution of the integrated services digital networks

Two important developments in the telecommunications network in the recent years are digital transmission and digital switching. Based on these two important developments, the idea of an *integrated digital network (IDN)* that integrates the functions of transmission and switching arose[1]. Fig. 1.1 is the block diagram of the IDN. Incoming voice signals are digitized using pulse-code modulation (PCM) and multiplexed using time-division multiplexing (TDM). The intermediate time-division digital switches can switch the individual signals without decoding them. Thus, the voice signals can travel a long range using only a single digital encoding-decoding process near the ends of the connection. In addition, due to time-division multiplexing, the system can accommodate digital signals of virtually any origin. Thus, digital data of various kinds can share the same network with the voice signals. The situation is different for the analog network, in which the incoming voice lines are modulated and frequency division multiplexed (FDM) at the end office and sent out over an FDM line. At the intermediate switching centers of an analog network, the incoming FDM carrier has to be demultiplexed and demodulated, switched by a space-division switch, and multiplexed and modulated before being retransmitted. This process results in noise accumulation as well as cost increase.

Despite all the progress in toll switching and trunking, local telephones are still sending analog signals to the end office, where those analog signals are digitized and then digitally transmitted. Thus, the next step towards a digital network is to extend the digital capability to the local loop and provide a digital service to the end user. The consequences of this extension will not only enable end-to-end digital voice transmission but also open the door for a wide variety of digital data services. This leads to the idea of an *integrated*

*services digital network (ISDN)*. Fig. 1.2 is a block diagram of the ISDN. The telephones and data terminals are connected to the ISDN user interface. Through the *digital subscriber loop (DSL)*, the signals are digitally transmitted to the ISDN central office, where a large number of ISDN subscriber loop signals are connected to the IDN. Thus, in addition to accessing the circuit-switched networks, the users can access packet-switched networks and a wide variety of services.

## 1.2. Digital subscriber loops

One of the most critical elements in the evolution toward ISDN is the digital subscriber loop technology. For economic reasons, the DSL must be able to provide a high data-rate, full-duplex, digital transmission over the conventional twisted pair, which was used originally to transmit voice signals with a bandwidth up to 4 kHz. The preferred capacity of the DSL encompasses two B channels and one D channel. The B channel is the basic user channel, whose data rate is 64 kbps. It is mainly designed to carry one PCM-encoded digital voice, although other uses of the B channel such as high-speed data or low-scan video signal are possible. The D channel, with a data rate of 16 kbps, is used to support a low-speed digital data transmission. The data rate required to carry 2B+D channels is 144 kbps. In addition to these channels, control signals are needed. This requires that the data rate be higher than 144 kbps. One proposed bit rate is 160 kbps, where 16 kbps control signals are inserted.

Two transmission schemes have been proposed to achieve full-duplex two-wire transmission. These two schemes are (1) burst mode (Fig. 1.2.1.a), and (2) echo-cancellation mode (Fig. 1.2.1.b) transmissions. In the burst mode DSL system, transmissions in different directions are separated in time. Thus, blocks of bits (bursts) are sent alternatively in each direction. Due to the propagation delay present in the line, a guard time must be provided between bursts; therefore, the line rate during bursts must be higher than twice the data rate. The determination of the number of bits per burst is a compromise between two requirements. This number should be kept large to reduce the

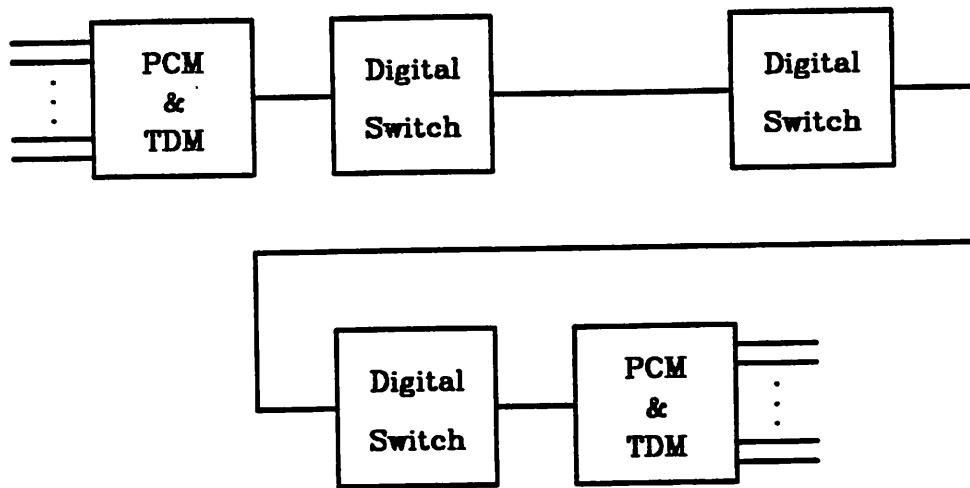


Figure 1.1. Block diagram of the IDN.

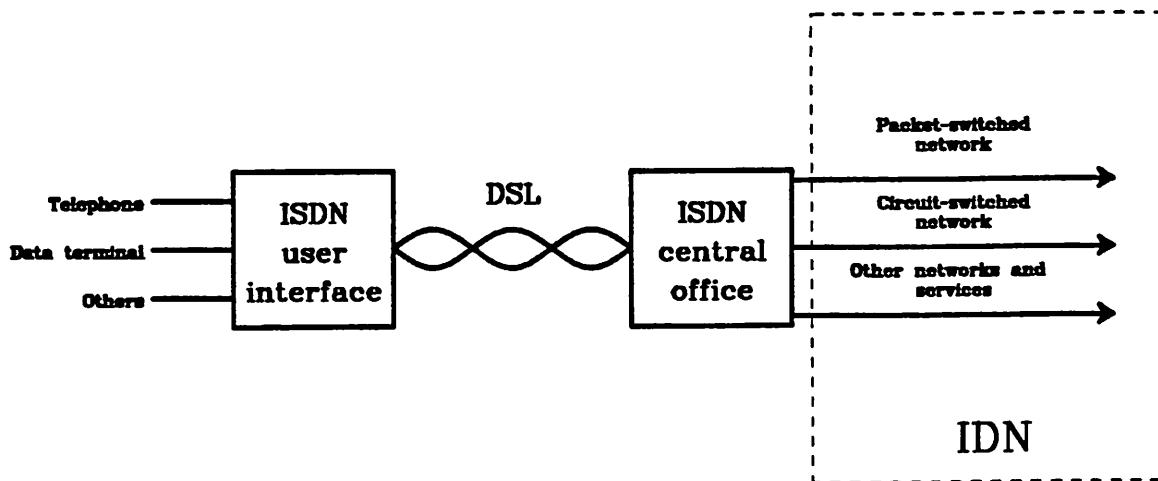


Figure 1.2. Block diagram of the ISDN.

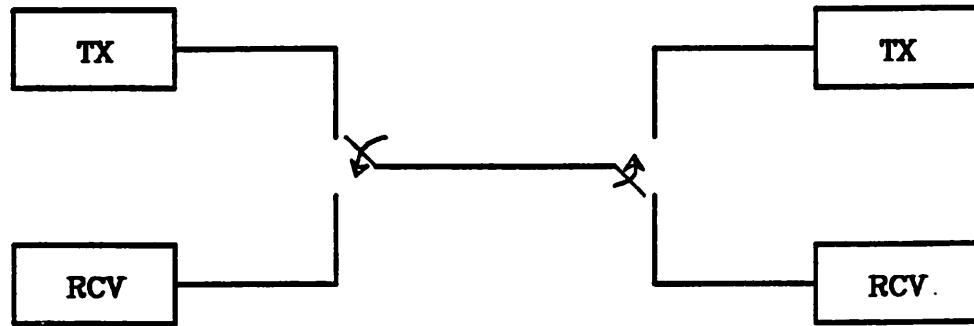


Figure 1.2.1.a. Burst mode data transmission

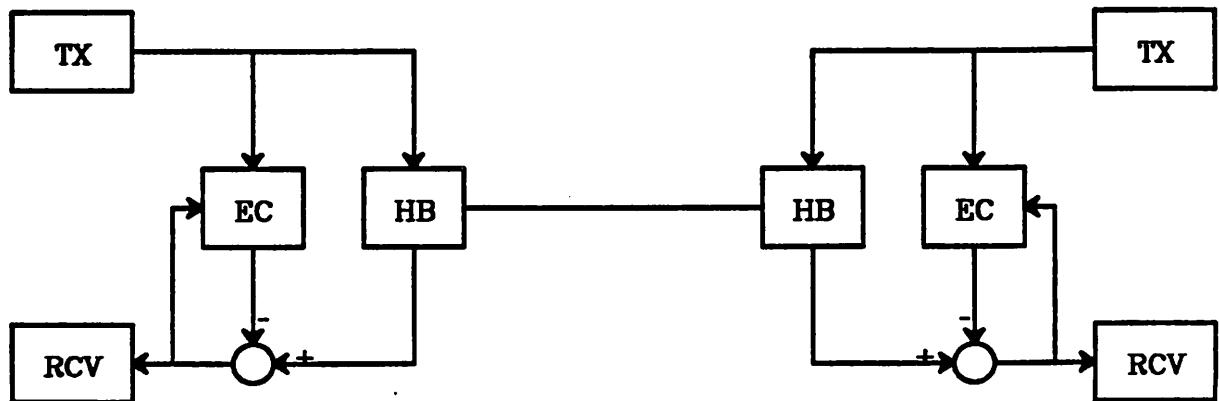


Figure 1.2.1.b. Echo-cancellation mode data transmission

line rate while it should be small enough to avoid long signal delay, which is undesirable for voice traffic. In the echo-cancellation mode transmission system, data flows simultaneously in both directions. A hybrid transformer is used to separate the transmitted signal from the received signal and to perform the two-to-four wire conversion. Due to the imperfect impedance matching of the hybrid transformer to the channel, there will be a leakage of the near-end transmitted signal through the hybrid transformer to the near-end receiver. This unwanted signal is called echo. An echo canceller is used to cancel this unwanted echo. Since data can flow simultaneously in both directions, the line rate is the same as the data rate. Consequently, this mode will offer better performance than the burst mode transmission in the sense that either higher data rate or longer transmission range can be achieved. The hardware complexity of the echo-cancellation mode system is higher than that of the burst mode system mostly due to the complexity of the echo canceller. However, the fast development of integrated-circuit technology has rapidly decreased the cost of the extra hardware, and LSI realization of the echo-cancellation mode system has been proven to be feasible [2]. Thus the echo-cancellation scheme is now preferred over the burst mode transmission scheme[2].

Two of the most important elements in the DSL technology are echo cancellation the timing recovery. There has been extensive work devoted to these two topics[2, 3]. An additional factor that critically affects the DSL performance is the choice of the line code. This thesis starts with an overview of the DSL system (Chapter 2), where the system impairments, design parameters, and individual system components are discussed. In Chapters 3 and 4, a particular partial response code is proposed as the line code and the system performance is evaluated and compared with other systems using different line codes. Chapter 5 gives a discussion on various timing recovery techniques. The timing jitter problem is presented in chapter 6 accompanied by a proposed solution to this problem. A bread-board prototype system was built and the experimental results are also presented.

## CHAPTER 2

### System Overview

In this chapter, various impairments affecting the performance of the digital subscriber loop system are first discussed. Some design parameters are introduced. Section 2.2 describes the individual system components and some design techniques.

#### 2.1. Impairments

The main sources of impairments in the context of a digital subscriber loop (DSL) are *white noise*, *crosstalk*, *echo*, *impulse noise* and *intersymbol interference*.

##### 2.1.1. White Noise

White noise is characterized by its broad and flat power spectrum. It is generally not the dominant impairment. However, since the crosstalk signal to noise ratio is independent of the transmitted signal power, the transmitted signal power is normally kept small to reduce the interference to foreign systems. The reduction in the transmitted signal power will cause a decrease in the white noise signal to noise ratio. In such cases, white noise may become important.

##### 2.1.2. Crosstalk

Crosstalk is one of the most significant limitations in the digital subscriber loops. Its mechanism is primarily due to the capacitive coupling of the signal on one wire to another. The amplitude distribution of crosstalk noise can be adequately modeled by a Gaussian distribution when the number of crosstalk interferers is large. Its power spectral density is not flat and is dependent on

- (1) the power transfer function of the crosstalk coupling, and
- (2) the transmitted signal power spectrum.

There are two kinds of crosstalk, *near-end crosstalk* (NEXT) and *far-end crosstalk* (FEXT). They are shown in Fig. 2.1.2.1. Notice that FEXT suffers the same channel loss as the signal while NEXT does not. Therefore, in the situation where the signals are traveling in both directions within the same cable, such as DSL, NEXT will be the dominant interference compared with FEXT. Much study has been done[4] and the results are that NEXT increases at a rate of 4.5 dB/octave ( $f^{\frac{3}{2}}$ ) while FEXT increases by 6dB/octave ( $f^2$ ).

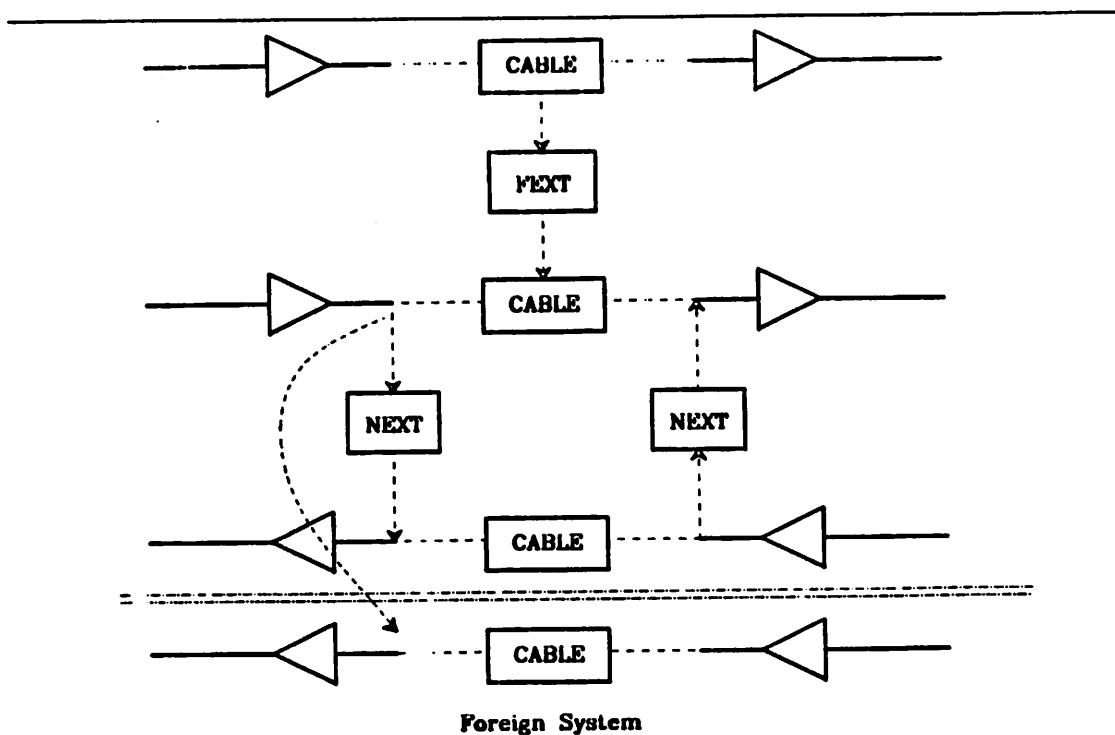


Figure 2.1.2.1. Near-end crosstalk (NEXT) and Far-end crosstalk (FEXT)

In order to analyze the NEXT power, we first define the NEXT power transfer function  $X_{next}$  :

$$|X_{next}|^2 = K \left[ \frac{f}{f_b} \right]^{\frac{3}{2}} \quad (2.1.2.1)$$

where  $K$  is an empirical constant. The power transfer function accounts for the mechanism of the coupling between twisted pairs. The NEXT power is given by

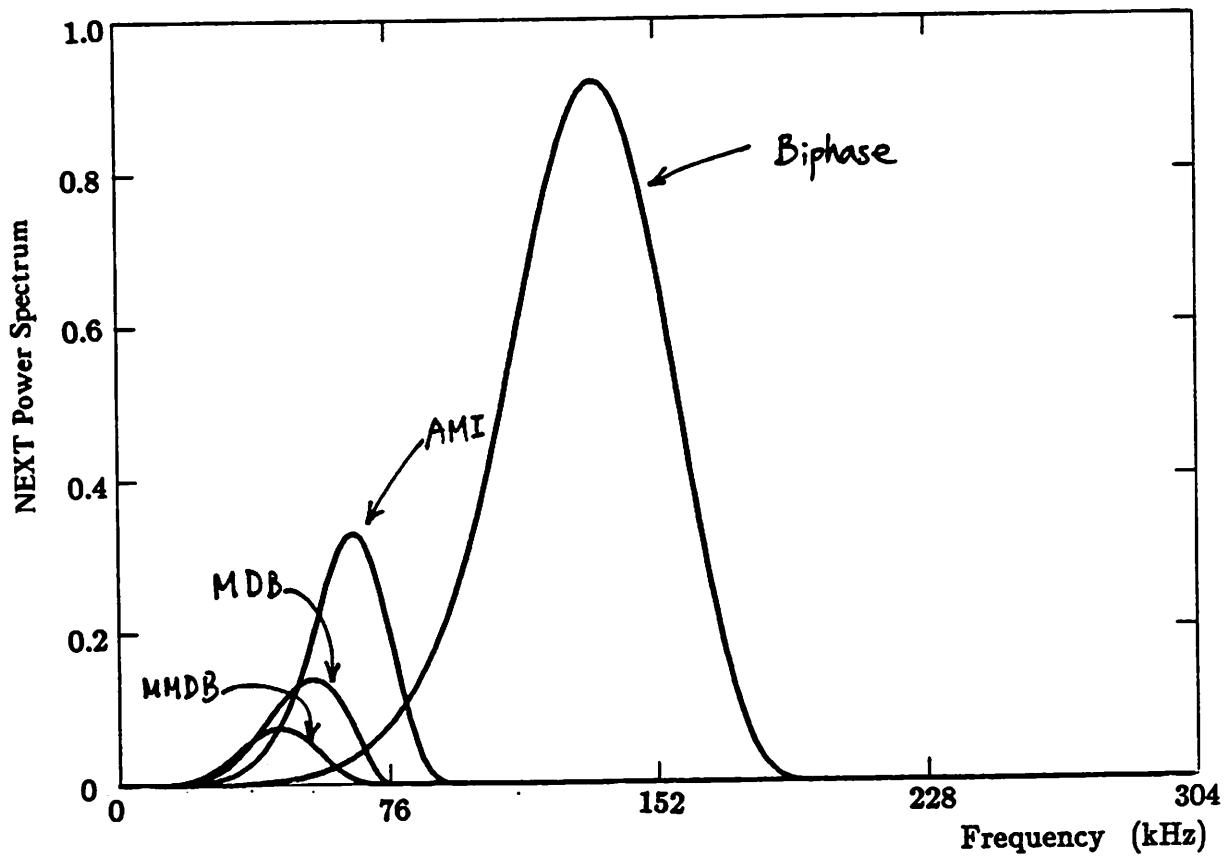
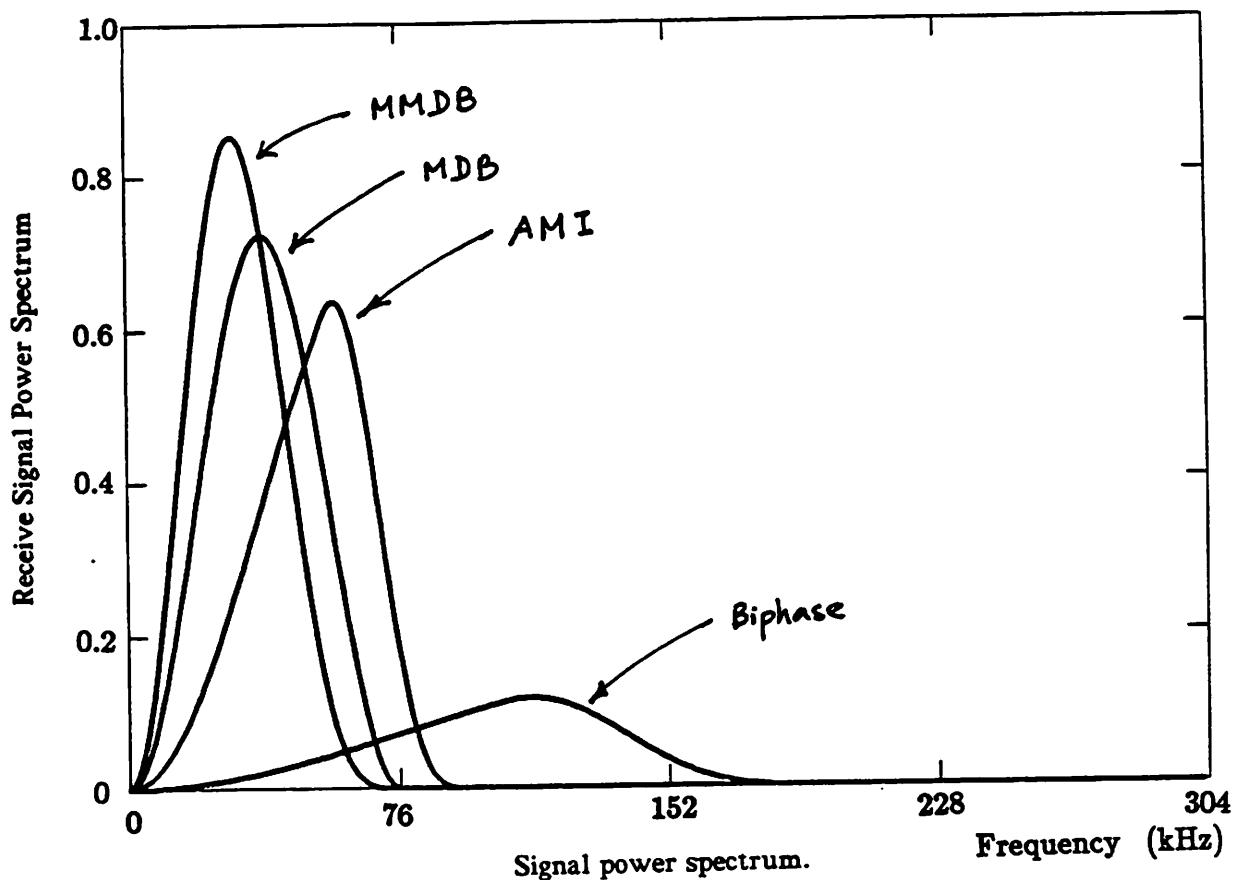
$$\begin{aligned} P_{next} &= K \int S_{inp}(f) |TX(f)|^2 |RCV(f)|^2 f^{\frac{3}{2}} df \\ &= K \int S_{rcv} \frac{f^{\frac{3}{2}}}{|C(f)|^2} df \end{aligned} \quad (2.1.2.2)$$

where  $S_p$  and  $S_{rcv}$  are the spectrum of the input signal to the transmit filter and the received signal spectrum respectively and

$$S_{rcv} = S_{inp} |TX(f)|^2 |C(f)|^2 |RCV(f)|^2 \quad (2.1.2.3)$$

It is interesting to notice that the NEXT SNR is independent of the level of the transmitted signal power because a change in the transmitted signal power level will affect the received signal power equally as much as it affects the NEXT power. Therefore, in DSL, the transmitted signal power can be reduced, with NEXT SNR unchanged, to reduce radio frequency interference (RFI) to a point where the white noise is comparable with NEXT. It is also because of this reason that the white noise, normally unimportant, should also be considered.

From (2.1.2.2) we see that the NEXT power is the integral of the received signal power spectrum weighted by the function  $f^{\frac{3}{2}}/|C(f)|^2$  whose gain increases with frequency at a rate higher than  $f^{\frac{3}{2}}$  due to the low pass nature of the channel. Therefore, to reduce the NEXT, one must design the system such that the received signal has as little power at high frequencies as possible. Fig. 2.1.2.2 shows the received signal spectrum and the NEXT spectrum of AMI system and the modified duobinary partial response system. These line codes will be described in Chapter 3.



NEXT power spectrum.  
Figure 2.1.2.2. Signal and NEXT power spectrum.

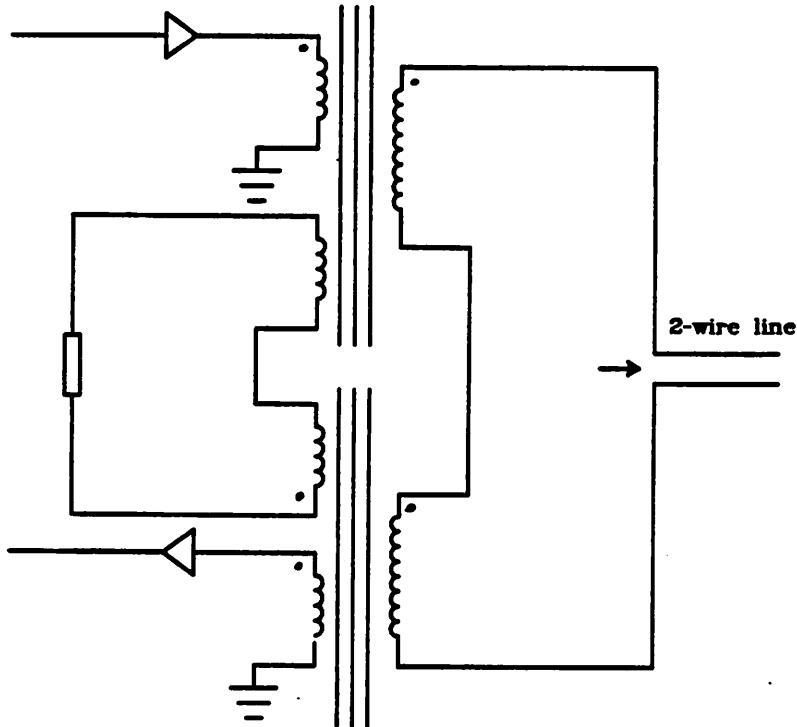
### 2.1.3. Impulse noise

Impulse noise is characterized by infrequent high amplitude bursts of noise and is generally evaluated by the number of events per unit time instead of its statistical distribution. It is mostly caused by central office switching transients and lightning. When it occurs, the signal can be corrupted beyond recognition. However, the use of an interleaved error correction code can prevent noise interruption in reception if an accurate characterization of the noise is used in design of the error-correcting code.

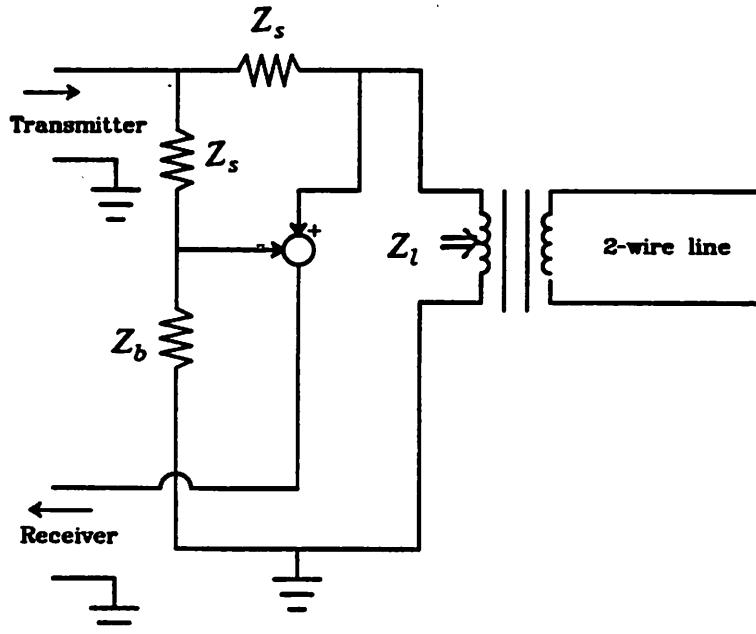
### 2.1.4. Echo

Echo is the leakage of the near-end signal through the hybrid transformer to the near-end receiver. Fig. 2.1.4.1 shows the diagram of a hybrid transformer. If  $Z_b$  is matched to the channel impedance  $Z_L$ , the receiver will be isolated from the transmitter. However, the complete isolation of the receiver from the transmitter requires that  $Z_b$  be matched to the channel impedance  $Z_L$  at all frequencies, which is not possible. In practical situations, variations in cable impedances are strongest at lower frequencies, thus the impedance matching is better at higher frequencies than at lower frequencies. Consequently, a hybrid transformer normally provides better echo attenuation at higher frequencies than at lower frequencies. This results in relatively large echo power at low frequencies, which explains why echo pulses tend to have long tails. An *echo canceller* is used to cancel this unwanted echo. The hybrid transformer will normally provide about 10 dB loss. Therefore, the desired far-end signal to echo noise ratio at the receiver is -30 dB for 40 dB channel attenuation. To provide 20 dB signal to noise ratio (SNR), 50 dB echo cancellation is therefore required.

Since echo power is high at low frequencies, a high-pass filter can be added into the echo path. This will improve signal to echo noise ratio if the 3dB frequency of the high-pass filter is well chosen. In fact, a filter that minimizes the mean-square error



Simplified diagram of a hybrid transformer.



$$\text{hybrid loss} = 20 \log \left| \frac{(Z_l + Z_s)(Z_b + Z_s)}{Z_l Z_b - Z_s Z_b} \right|$$

Equivalent model.

Figure 2.1.4.1. Hybrid transformer.

can be found if the channel response is known. The improvement in signal to echo noise ratio can reduce the requirements on the echo canceller. Furthermore, a high-pass filter can also reduce the length of the echo path impulse response. This is beneficial since the number of the echo canceller taps required depends directly on the length of echo path impulse response.

Now we are going to derive the filter which minimizes the mean-square error. This derivation follows [ 5] closely. Let the system be formulated as shown in Fig. 2.1.4.2. The received signal, composed of the far-end signal, echo, and additive noise, is  $r(t)$ :

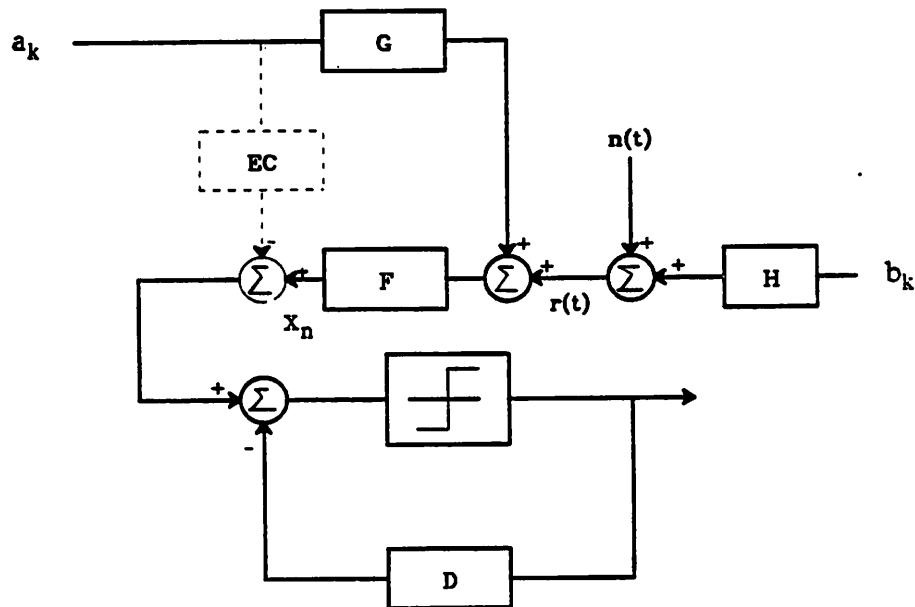


Figure 2.1.4.2. System model for deriving minimum MSE filter.

$$r(t) = \sum_k b_k h(t - kT) + \sum_k a_k g(t - kT) + n(t) \quad (2.1.4.1)$$

where  $h(t)$  is the channel impulse response,  $g(t)$ , the echo impulse response,  $a_k$ , the near-end transmitted data,  $b_k$ , the far-end transmitted data, and  $n(t)$ , the additive noise. Define

$$p_k = \int_{-\infty}^{\infty} f(\tau)h(kT - \tau)d\tau \quad (2.1.4.2)$$

$$q_k = \int_{-\infty}^{\infty} f(\tau)g(kT - \tau)d\tau \quad (2.1.4.3)$$

where  $f(t)$  is the impulse response of the optimum filter. Thus,  $p_k$  and  $q_k$  are the sampled responses of the filter to the channel and echo path respectively. The output sample of the filter  $x_n$  is:

$$x_n = \sum_k b_k p_{n-k} + \sum_k a_k q_{n-k} + \int_{-\infty}^{\infty} f(\tau)n(nT - \tau)d\tau \quad (2.1.4.4)$$

The objective is to find  $f(\tau)$  which minimizes the mean-square error  $\bar{\epsilon}^2$ :

$$\bar{\epsilon}^2 = E[(x_n - \sum_{j=1}^N d_j \delta_{n-j} - b_n)^2] \quad (2.1.4.5)$$

Here we have assumed that a decision feedback equalizer with coefficients  $\{d_k\}$  is used and  $\delta_k$  is the detected data symbol. Assume that  $\delta_{n-j} = b_{n-j}$  and  $a_n, b_n$ , and  $n(t)$  are independent and zero-mean with

$$E[a_n a_j] = E[b_n b_j] = \sigma_b^2 \delta_{n-j}$$

Thus, (2.1.4.5) becomes

$$\begin{aligned} \bar{\epsilon}^2 = & \sigma_b^2 \left[ \sum_{k=-\infty}^{-1} p_k^2 + \sum_{k=N+1}^{\infty} p_k^2 + (1-p_0)^2 + \sum_k q_k^2 \right. \\ & \left. + \sum_{j=1}^N (d_j - p_j)^2 \right] + \int_{-\infty}^{\infty} R_n(\gamma) \int_{-\infty}^{\infty} f(\tau) f(\tau - \gamma) d\tau d\gamma \end{aligned} \quad (2.1.4.6)$$

where  $R_n(\gamma) = E[n(\tau)n(\tau + \gamma)]$  is the autocorrelation function of the noise. The DFE coefficients  $d_j$  are chosen to be equal to  $p_j$  to minimize  $\bar{\epsilon}^2$ . Now, by the method of variations, replace  $f(\tau)$  in (2.1.4.6) by  $f(\tau) + \xi s(\tau)$  and set

$$\frac{\partial}{\partial \xi} \bar{\epsilon}^2|_{\xi=0} = 0$$

We have:

$$\begin{aligned}
 0 = 2\sigma_b^2 &\left\{ \left( \sum_{k=-\infty}^{-1} + \sum_{k=N+1}^{\infty} \right) p_k \int_{-\infty}^{\infty} s(\tau) h(kT - \tau) d\tau \right. \\
 &- (1-p_0) \int_{-\infty}^{\infty} s(\tau) h(-\tau) d\tau + \sum_k q_k \int_{-\infty}^{\infty} s(\tau) g(kT - \tau) d\tau \\
 &\left. + 2 \int_{-\infty}^{\infty} R_n(\gamma) \int_{-\infty}^{\infty} s(\tau) f(\tau - \gamma) d\tau d\gamma \right\} \tag{2.1.4.7}
 \end{aligned}$$

This is true for all  $s(\tau)$ , therefore, we have

$$\begin{aligned}
 0 = \sigma_b^2 &\left\{ \left( \sum_{k=-\infty}^{-1} + \sum_{k=N+1}^{\infty} \right) p_k h(kT - \tau) - (1-p_0) h(-\tau) \right. \\
 &\left. + \sum_k q_k g(kT - \tau) \right\} + \int_{-\infty}^{\infty} R_n(\gamma) f(\tau - \gamma) d\gamma \tag{2.1.4.8}
 \end{aligned}$$

From (2.1.4.8), we can solve for  $f(\tau)$ . This can be done by first taking the Fourier transform of (2.1.4.8), dividing by  $S(w)$ , the Fourier transform of  $R_n(\tau)$ , and then taking the inverse Fourier transform:

$$\begin{aligned}
 f(t) = \sigma_b^2 &\left\{ - \left( \sum_{k=-\infty}^{-1} + \sum_{k=N+1}^{\infty} \right) p_k u(kT - t) + (1-p_0) u(-t) \right. \\
 &\left. - \sum_k q_k v(kT - t) \right\} \tag{2.1.4.9}
 \end{aligned}$$

where

$$u(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H(\omega)}{S(\omega)} e^{j\omega t} d\omega$$

$$v(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{G(\omega)}{S(\omega)} e^{j\omega t} d\omega$$

Notice that  $u(-t)$  and  $v(-t)$  are the impulse responses of the matched filters matched to channel and echo path respectively. Fig. 2.1.4.3 is the structure of the optimum filter  $F(\omega)$ . The two transversal filters have transfer functions:

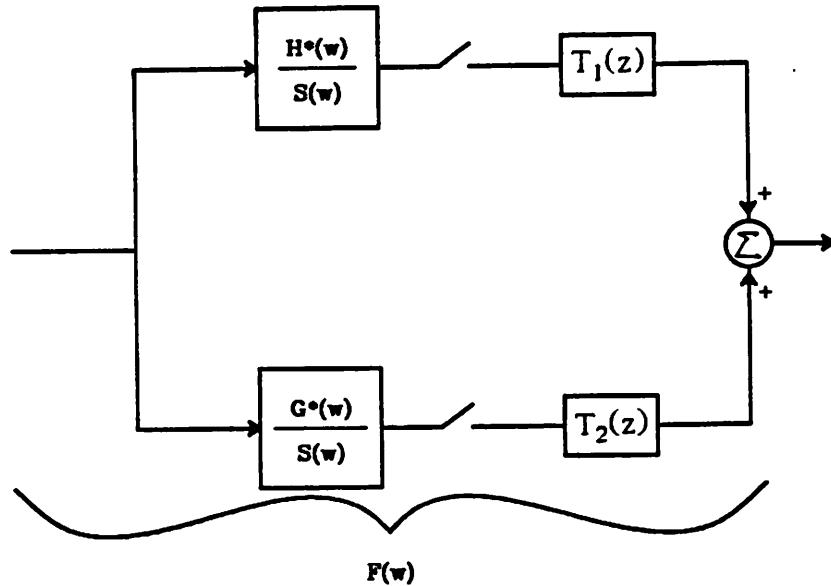


Figure 2.1.4.3. Minimum MSE filter.

$$\begin{aligned}
 T_1(z) &= \sigma_b^2 \left\{ 1 - p_0 - \sum_{k=-\infty}^{-1} p_k z^{-k} - \sum_{k=N+1}^{\infty} p_k z^{-k} \right. \\
 &\quad \left. = 1 - P(z) + \sum_{k=1}^N p_k z^{-k} \right. \tag{2.1.4.10}
 \end{aligned}$$

$$T_2(z) = -\sigma_b^2 \sum_k q_k z^{-k} = \sigma_b^2 Q(z) \tag{2.1.4.11}$$

Since  $p_k$  and  $q_k$  are responses of  $f(t)$  to the channel and echo path, to find  $P(z)$ , multiply (2.1.4.9) by  $h(mT - t)$  and integrate from  $-\infty$  to  $\infty$ . Therefore

$$p_m = \sigma_b^2 \left\{ (1 - p_0) w_m - \left( \sum_{k=-\infty}^{-1} + \sum_{k=N+1}^{\infty} \right) p_k w_{m-k} - \sum_k q_k y_{m-k} \right\} \tag{2.1.4.12}$$

where

$$w_k = \int_{-\infty}^{\infty} u(\tau) h(\tau + kT) d\tau \tag{2.1.4.13}$$

$$y_k = \int_{-\infty}^{\infty} v(\tau)h(\tau + kT)d\tau \quad (2.1.4.14)$$

$w_k$  and  $y_k$  are the responses of the two matched filters to  $h(t)$  sampled at  $t = kT$ .

Therefore  $w_k$  and  $y_k$  can also be expressed as

$$w_k = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{|H(\omega)|^2}{S(\omega)} e^{-j\omega kT} d\omega \quad (2.1.4.15)$$

$$y_k = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H^*(\omega)G(\omega)}{S(\omega)} e^{-j\omega kT} d\omega \quad (2.1.4.16)$$

Similarly, multiplying (2.1.4.9) by  $g(mT - \tau)$  and integrating, we have

$$q_m = \sigma_b^2 \left\{ (1 - p_0) \theta_m - \left( \sum_{k=-\infty}^{-1} + \sum_{k=N+1}^{\infty} \right) p_k \theta_{m-k} - \sum_k q_k \psi_{m-k} \right\} \quad (2.1.4.17)$$

where

$$\theta_k = \int_{-\infty}^{\infty} u(\tau)g(\tau + kT)d\tau \quad (2.1.4.18)$$

$$\psi_k = \int_{-\infty}^{\infty} v(\tau)g(\tau + kT)d\tau \quad (2.1.4.19)$$

are the responses of the two matched filters to  $g(t)$  sampled at  $t = kT$  and

$$\theta_k = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H(\omega)G^*(\omega)}{S(\omega)} e^{-j\omega kT} d\omega \quad (2.1.4.20)$$

$$\psi_k = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{|G(\omega)|^2}{S(\omega)} e^{-j\omega kT} d\omega \quad (2.1.4.21)$$

Taking the z-transform of (2.1.4.12) and (2.1.4.17), we get

$$P(z) = \sigma_b^2 \left\{ (1 + \sum_{k=1}^N p_k z^{-k} - P(z))W(z) - Q(z)Y(z) \right\} \quad (2.1.4.22)$$

$$Q(z) = \sigma_b^2 \left\{ (1 + \sum_{k=1}^N p_k z^{-k} - P(z))\Theta(z) - Q(z)\Psi(z) \right\} \quad (2.1.4.23)$$

Solving (2.1.4.22) and (2.1.4.23) simultaneously and noticing that since  $y_k = \theta_{-k}$ , hence  $\Theta(z) = Y(z)$ ,

$$P(z) = \sigma_b^2 (1 + \sum_{k=1}^N p_k z^{-k}) \frac{A(z)}{1 + \sigma_b^2 A(z)} \quad (2.1.4.24)$$

$$Q(z) = \sigma_b^2(1 + \sum_{k=1}^N p_k z^{-k}) \frac{Y(z^{-1})}{(1 + \sigma_b^2 \Psi(z))(1 + \sigma_b^2 A(z))} \quad (2.1.4.25)$$

where

$$A(z) = \frac{W(z) + \sigma_b^2(W(z)\Psi(z) - Y(z)Y(z^{-1}))}{1 + \sigma_b^2\Psi(z)} \quad (2.1.4.26)$$

To get  $p_n$  and  $q_n$ , we can take inverse z-transform of (2.1.4.25) and (2.1.4.26). But, first we define  $i_k$  to be the inverse z-transform of

$$\frac{\sigma_b^2 A(z)}{1 + \sigma_b^2 A(z)}$$

and  $j_k$  be the inverse z-transform of

$$\frac{\sigma_b^2 Y(z^{-1})}{(1 + \sigma_b^2 \Psi(z))(1 + \sigma_b^2 A(z))}$$

then,

$$p_n = i_k + \sum_{k=1}^N p_k i_{n-k}, \quad 1 \leq n \leq N \quad (2.1.4.27)$$

and

$$q_n = i_k + \sum_{k=1}^N p_k j_{n-k}, \quad 1 \leq n \leq N \quad (2.1.4.28)$$

From (2.1.4.27) and (2.1.4.28), we can solve for  $p_n$  and  $q_k$ .

It is impractical, however, to build an optimum filter since this requires the use of matched filters. In addition, because the channel response changes from one system to another, any design will only be optimum to a specific channel. However, we do find that a high-pass filter with a well-chosen 3dB frequency improves the cancellation of echoes.

## 2.1.5. Intersymbol Interference

For band-limited linear channels with frequency dependent attenuation and delay, pulses are dispersed and span over several baud periods, and therefore interfere with other pulses. This effect is known as intersymbol interference (ISI).

Two commonly used measures of the amount of the ISI are:

- (1) Mean-square ISI:

$$\frac{\sum_{k \neq 0} |h(kT)|^2}{|h(0)|^2} \quad (2.1.5.1)$$

where  $h(t)$  is the effective impulse response and  $T$  is the baud period.

- (2) Peak ISI, which relates to the worst-case intersymbol interference:

$$\sum_{k=-\infty, k \neq 0}^{\infty} |h(kT)| \quad (2.1.5.2)$$

One useful graphical representation of a data signal that allows immediate evaluation of the amount of ISI is an eye diagram, which is closely related to the peak ISI measure. An eye diagram is the superposition of all possible pulses with the length of time axis equals a multiple of a baud period. The vertical eye opening for a 2-level signal can be expressed analytically as:

$$\text{eye opening} = h(\tau) - \sum_{k=-\infty, k \neq 0}^{\infty} |h(\tau + kT)| \quad (2.1.5.3)$$

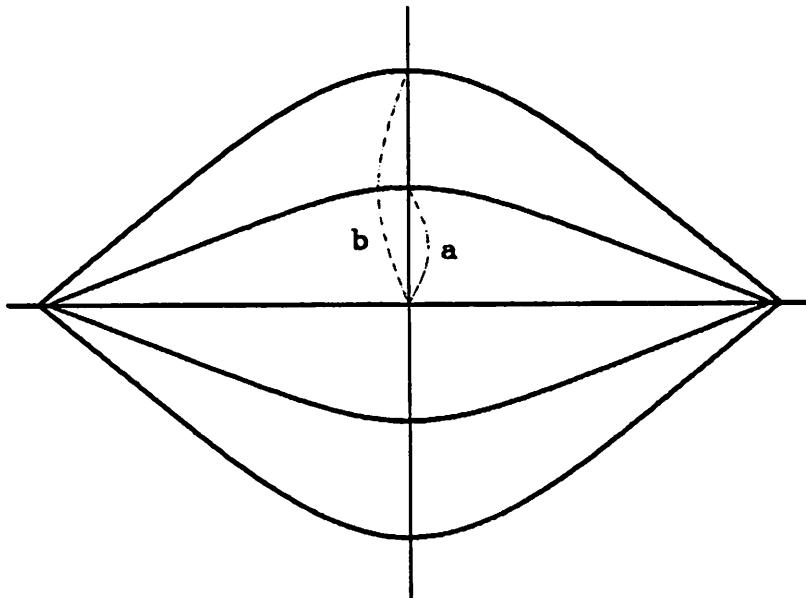
where  $\tau$  is the observation instant. And for a  $k$ -level signal, it is given as:

$$\text{eye opening} = h(\tau) - (k-1) \sum_{k=-\infty, k \neq 0}^{\infty} |h(\tau + kT)| \quad (2.1.5.4)$$

The eye opening can also be specified in terms of percentage, as shown in Fig. 2.1.5.1. It is given analytically as:

$$\% \text{ eye opening} = \frac{h(\tau) - \sum_{k=-\infty, k \neq 0}^{\infty} |h(\tau + kT)|}{h(\tau) + \sum_{k=-\infty, k \neq 0}^{\infty} |h(\tau + kT)|} \quad (2.1.5.5)$$

One other important piece of information that can be obtained from an eye diagram is the sensitivity of error probability of the system to timing jitter. This is related to the horizontal eye opening. The wider the horizontal eye opening is, the better immunity the system to the sampling phase and timing jitter will be.



$$\% \text{ eye opening} = \left( \frac{a}{b} \right) 100$$

Figure 2.1.5.1. Eye diagram.

The criterion for zero ISI is called the *Nyquist criterion*. It can be described as the following: For no ISI,  $h(t)$  should be such that

$$h(kT) = \begin{cases} 1 & k = 0 \\ 0 & k \neq 0 \end{cases}$$

Then

$$\sum_k h(kT) \delta(t - kT) = \delta(t) \quad (2.1.5.6)$$

Let  $H(w)$  be the Fourier transform of  $h(t)$ . Therefore,

$$\frac{1}{T} \sum_k H(\omega + m \frac{2\pi}{T}) = 1 \quad (2.1.5.7)$$

This is the Nyquist criterion. The left hand side of Equation (2.1.5.7),  $\frac{1}{T} \sum_k H(\omega + m \frac{2\pi}{T})$ , is called the folded spectrum. If  $H(w)$  is band limited to less than

two Nyquist bandwidths, i.e.,  $H(w) = 0$  for  $|w| \geq \frac{2\pi}{T}$ , and has a linear phase characteristic, then its amplitude response should have vestigial symmetry about the Nyquist frequency. Also, in order that Equation (2.1.5.7) be satisfied, the bandwidth of  $H(w)$  must be greater or equal to  $\frac{\pi}{T}$ . One set of filters that satisfy the Nyquist criterion are the *raised-cosine filters*. The frequency responses of these filters are given by:

$$H(f) = \begin{cases} T & 0 \leq |f| \leq (1-\beta)/2T \\ \frac{T}{2} \left[ 1 - \sin \left( \frac{\pi T(f - \frac{1}{2T})}{\beta} \right) \right] & (1-\beta)/2T \leq |f| \leq (1+\beta)/2T \\ 0 & \text{elsewhere} \end{cases} \quad (2.1.5.8)$$

where  $\beta$  is called the roll-off parameter, or the excess bandwidth. Their corresponding impulse responses are given by the expression:

$$h(t) = \frac{\sin \frac{\pi t}{T} \cos \frac{\beta \pi t}{T}}{\frac{\pi t}{T} \left[ 1 - \frac{4\beta^2 t^2}{T^2} \right]} \quad (2.1.5.9)$$

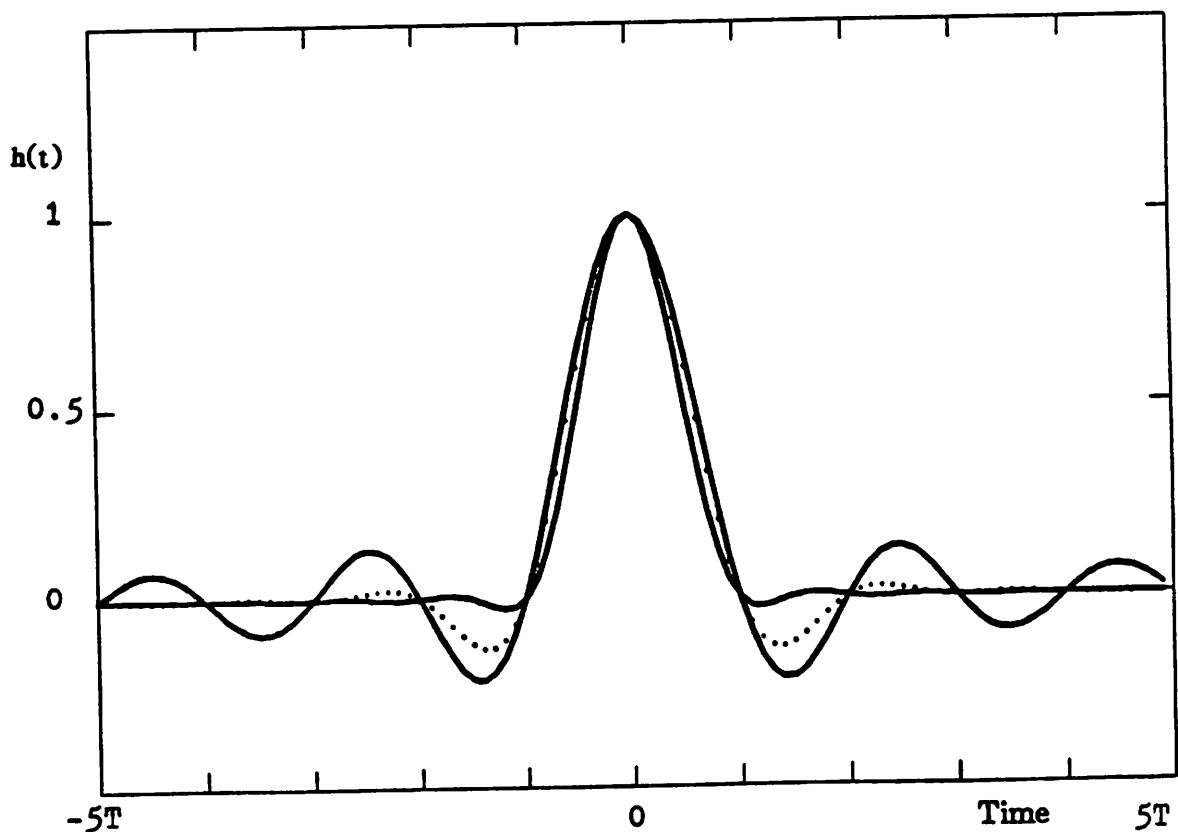
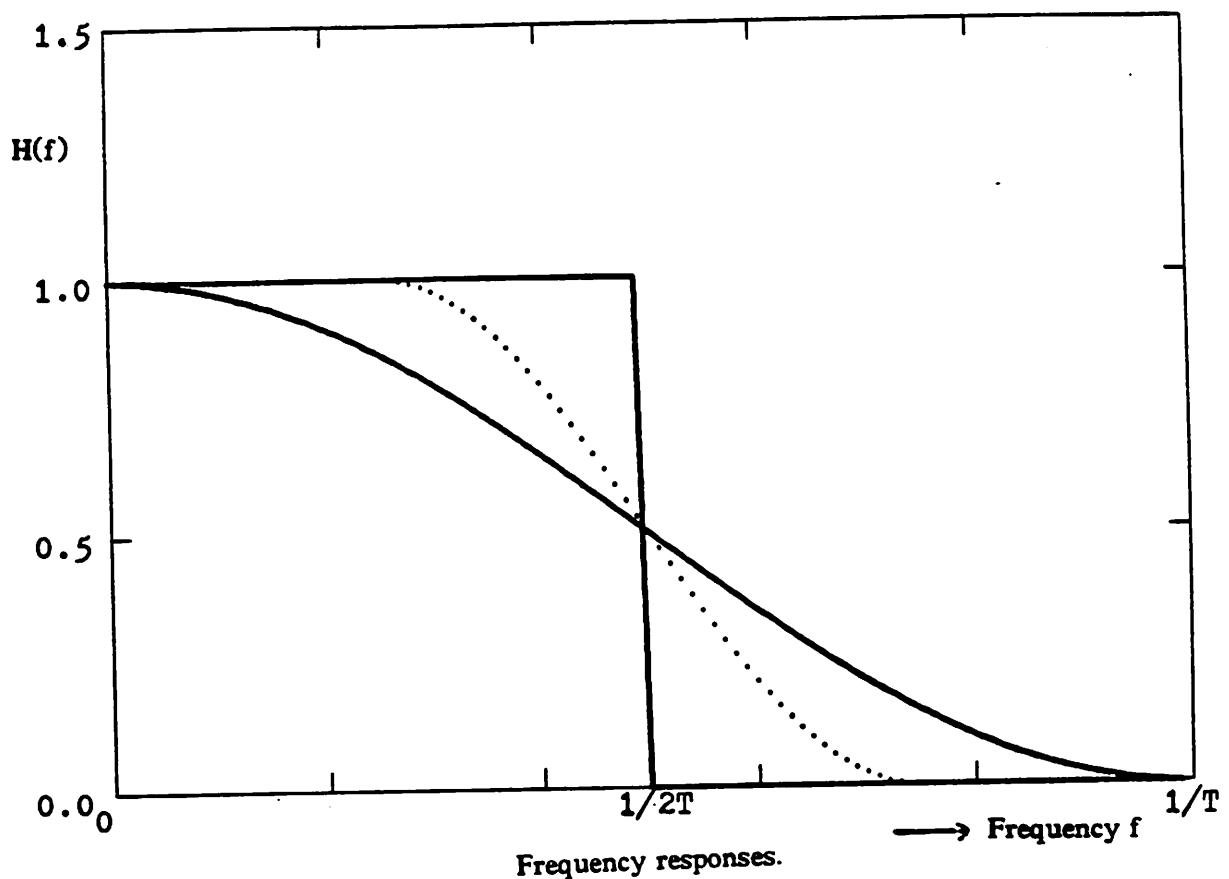
Fig. 2.1.5.2 is the frequency and impulse responses of raised-cosine filters with different excess bandwidths.

For  $\beta$  equals 0,  $H(f)$  becomes

$$H(f) = \begin{cases} T & 0 \leq |f| \leq \frac{\pi}{T} \\ 0 & \text{elsewhere} \end{cases} \quad (2.1.5.10)$$

This is the minimum bandwidth required to achieve zero ISI. The corresponding impulse response is:

$$h(t) = \frac{\sin \frac{\pi t}{T}}{\frac{\pi t}{T}} \quad (2.1.5.11)$$



Impulse responses.  
Figure 2.1.5.2. Frequency and impulse responses of raised-cosine filters.

This filter, having a brick-wall response, is not physically realizable. In addition, it suffers a practical limitation that it doesn't have any horizontal eye opening. This can be seen by examining the peak ISI:

$$\begin{aligned}
 \text{peak ISI} &= \sum_{k=-\infty, k \neq 0}^{\infty} |h(kT + \delta)| \\
 &= \sum_{k=-\infty, k \neq 0}^{\infty} \left| \frac{\sin \frac{\pi}{T}(kT + \delta)}{\frac{\pi}{T}(kT + \delta)} \right| \\
 &= 2 \frac{T}{\pi} \sin\left(\frac{\pi}{T}\delta\right) \sum_{k=1}^{\infty} \frac{1}{kT + \delta}
 \end{aligned} \tag{2.1.5.12}$$

Since the series  $\{ \frac{1}{kT + \delta} \}$  decreases as  $\frac{1}{k}$ , the infinite sum diverges. Thus, even in the absence of noise, the ISI will corrupt the signal so that correct detection is not possible for any sampling phase other than  $\delta = 0$ .

In practical designs, excess bandwidth is required. The larger the excess bandwidth is, the wider the horizontal eye opening will be. Fig. 2.1.5.3 shows the impulse responses and eye diagrams of the AMI system with various excess bandwidth. It is clear that as the excess bandwidth increases, the impulse response decays more rapidly, and therefore results in less ISI. Two factors must be considered to determine the amount of excess bandwidth required for a particular system. Firstly, the amount of excess bandwidth will affect the design of the timing recovery. It is clear that the timing recovery is more critical for narrow horizontal eye openings than for wide openings. The excess bandwidth has to be large enough to give a sufficiently wide horizontal eye opening such that for a given timing recovery technique, a given steady-state sampling phase that may not be at the maxima of the eye, and a given timing jitter variance, the system can achieve the required performance in error probability. Conversely, for a given excess bandwidth, the sampling phase has to fall within a certain range to achieve the required SNR. This gives the restriction to the

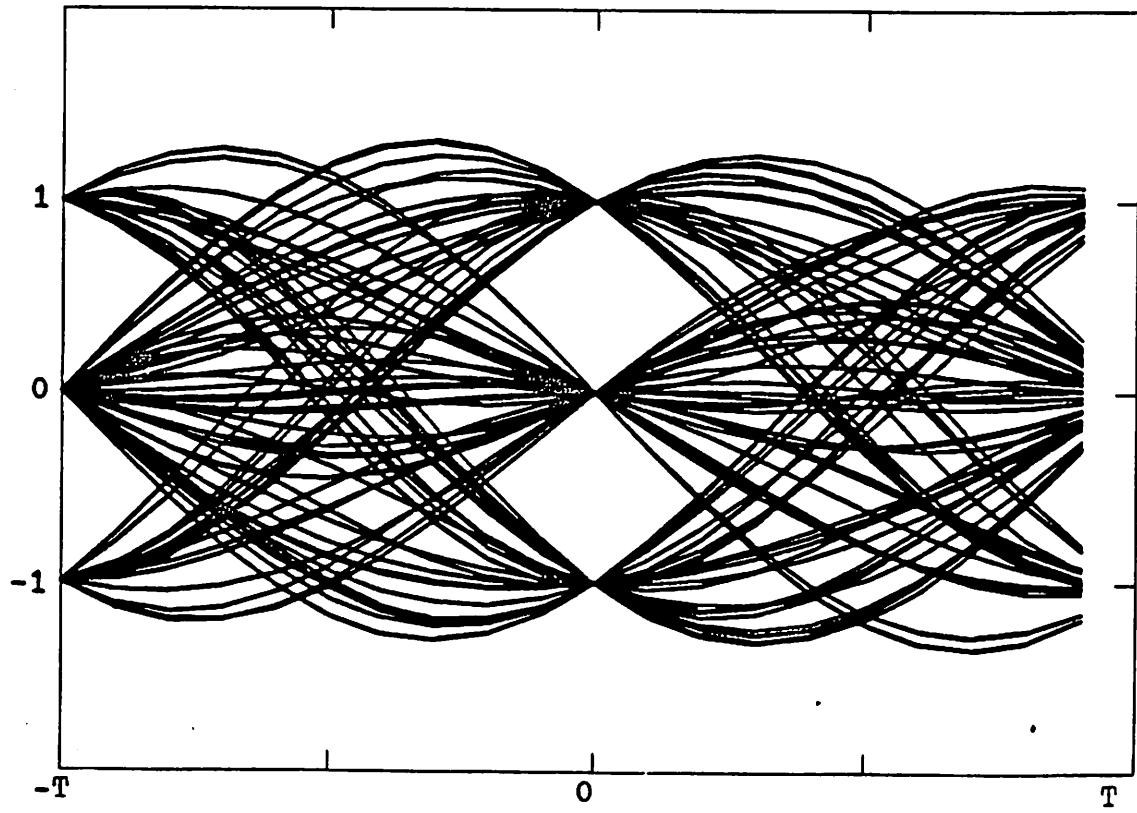
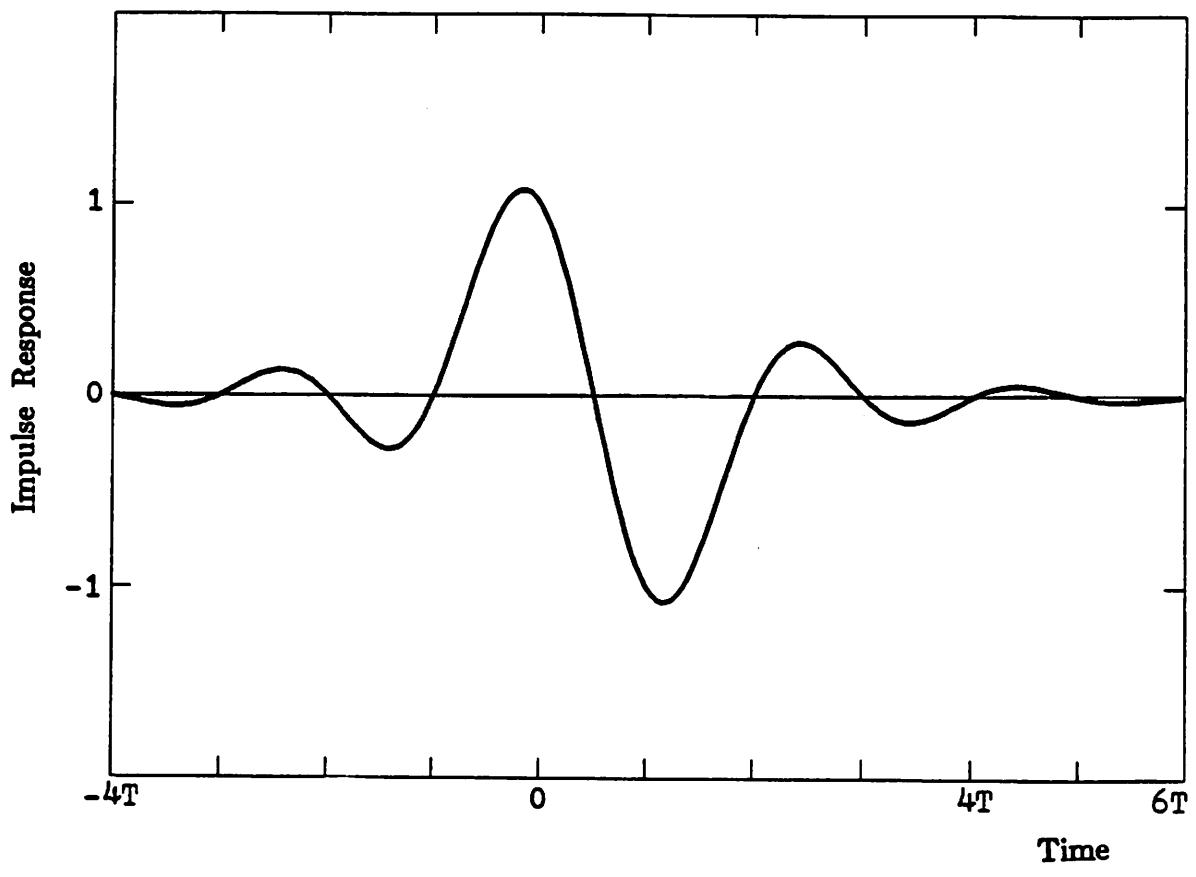
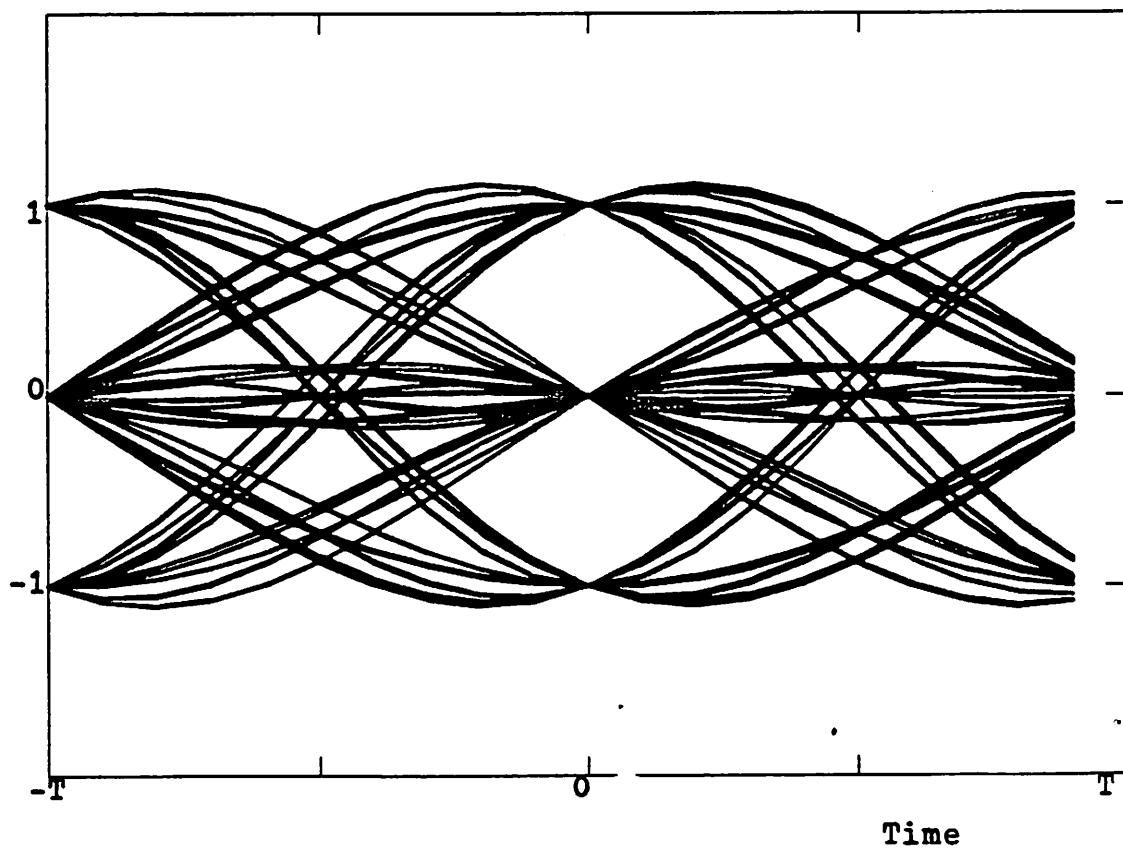
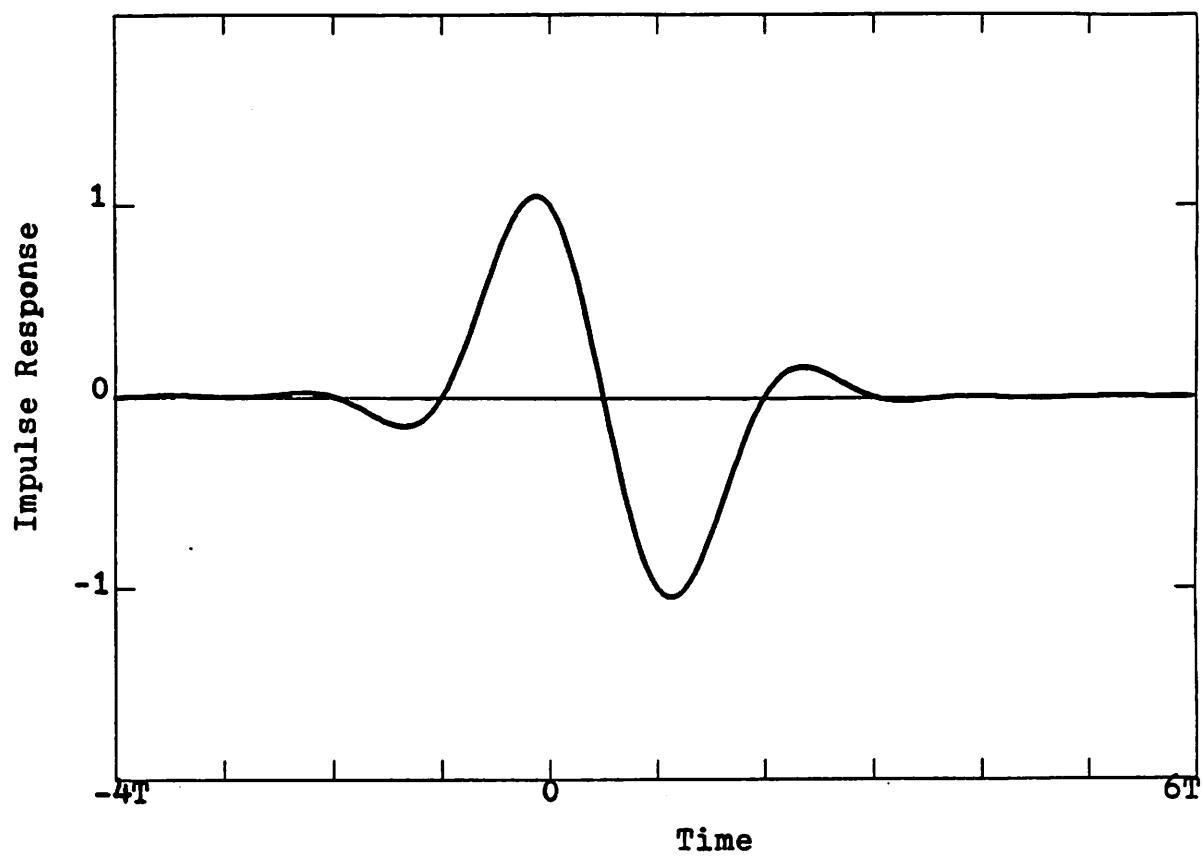


Figure 2.1.5.3. Impulse response and eye diagram  
of AMI system with 25% excess bandwidth.



**Figure 2.1.5.3.** Impulse response and eye diagram  
of AMI system with 50% excess bandwidth.

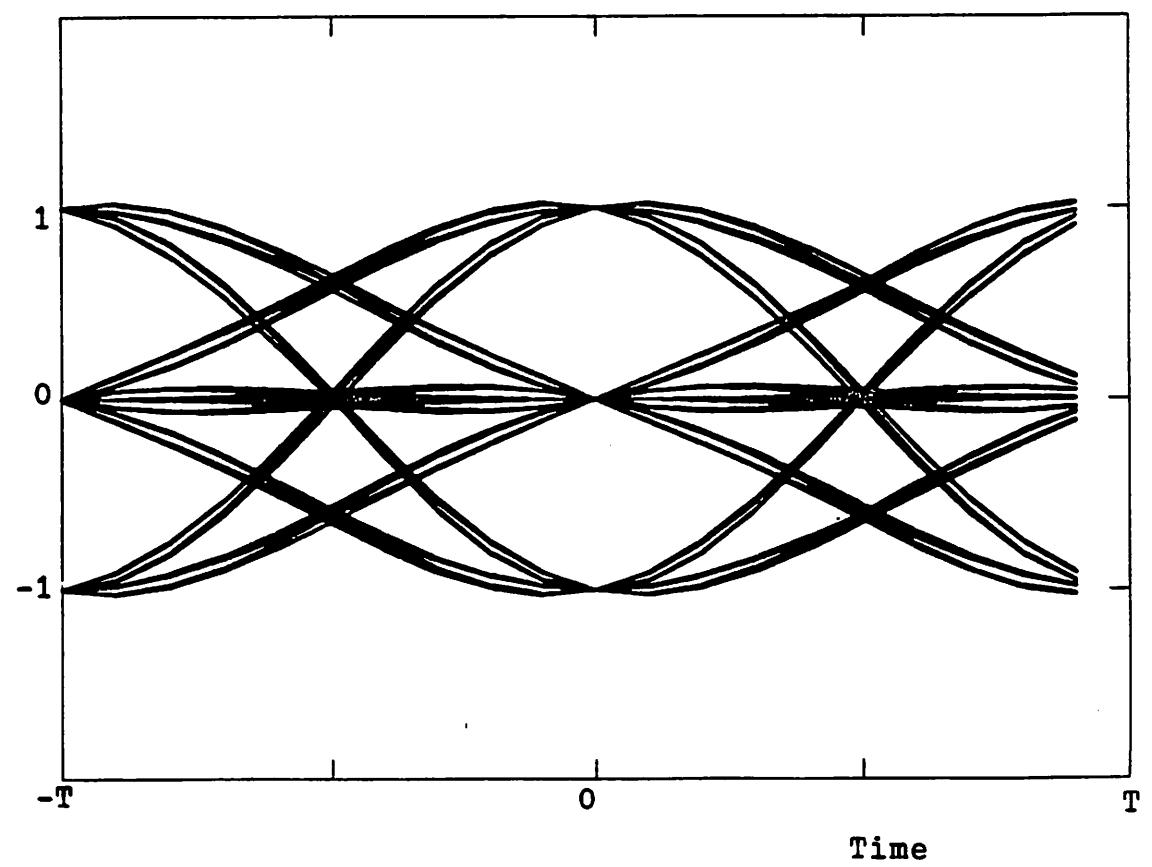
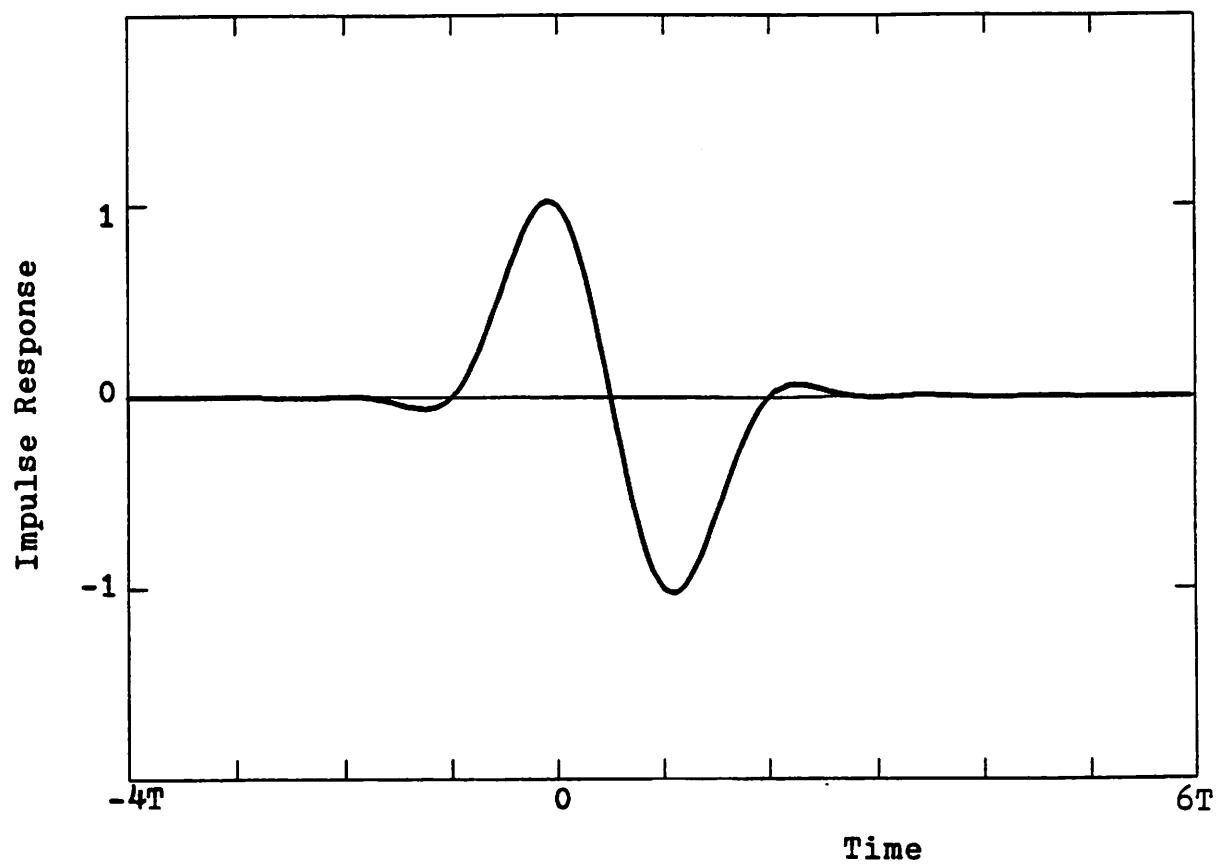


Figure 2.1.5.3. Impulse response and eye diagram  
of AMI system with 75% excess bandwidth.

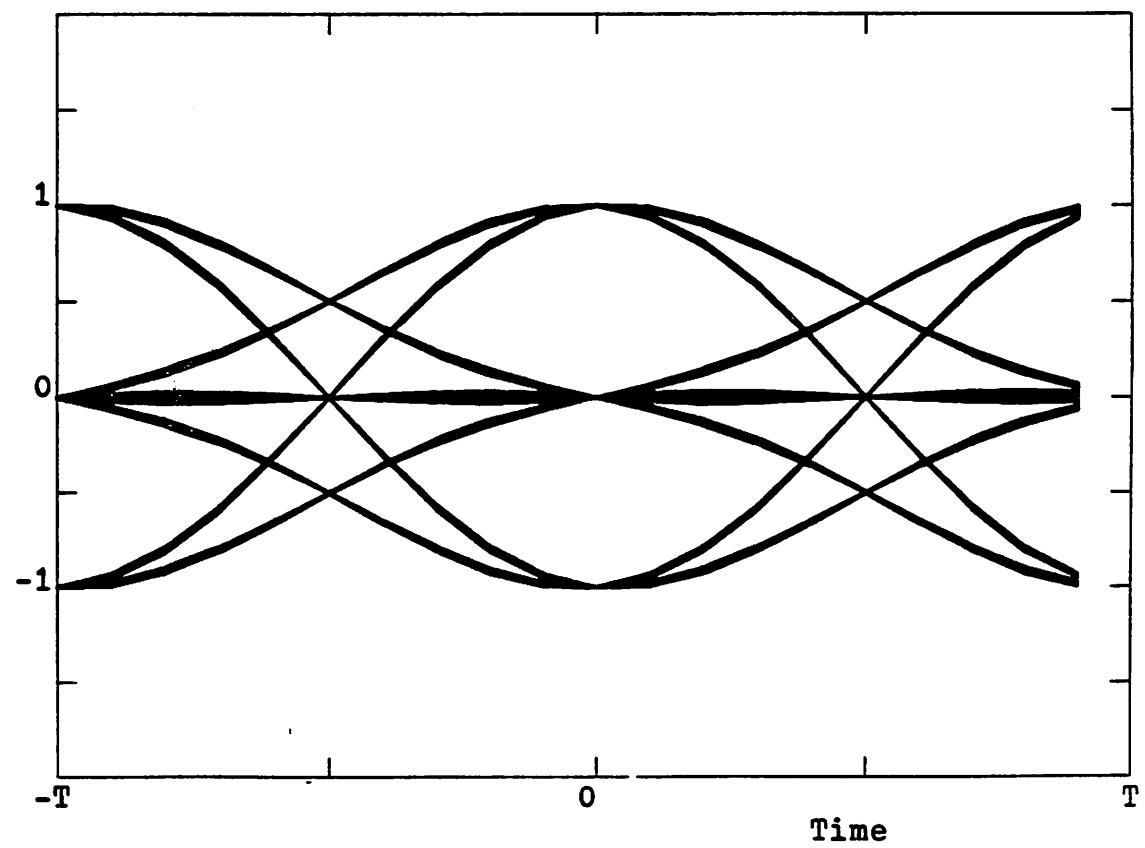
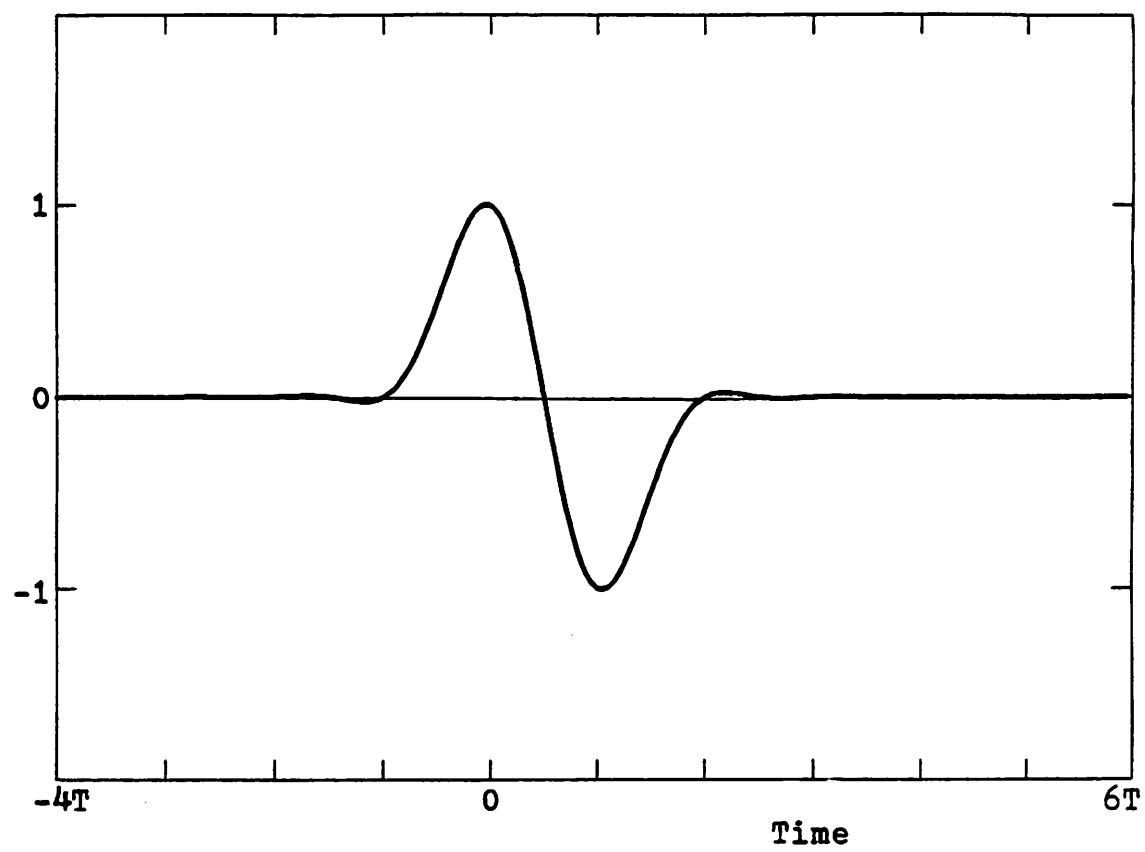


Figure 2.1.5.3. Impulse response and eye diagram  
of AMI system with 100% excess bandwidth.

steady-state sampling phase as well as the maximum allowable timing jitter.

A second factor that has to be considered is the noise penetration. Wider excess bandwidth generally implies more noise penetration into the system. Recall that the NEXT, one of the dominant impairments, has a power spectrum increasing rapidly as frequency increases. The increase in the excess bandwidth thus increases the noise power rapidly.

Up to this point, we have concentrated on the design of systems which give zero ISI. If we allow a controlled amount of ISI, a whole new class of signalings will arise. This extra degree of freedom will be useful in designing more efficient systems while this controlled ISI can still be removed eventually. This new class of signaling is called *partial response signaling* and will be discussed in chap. 3.

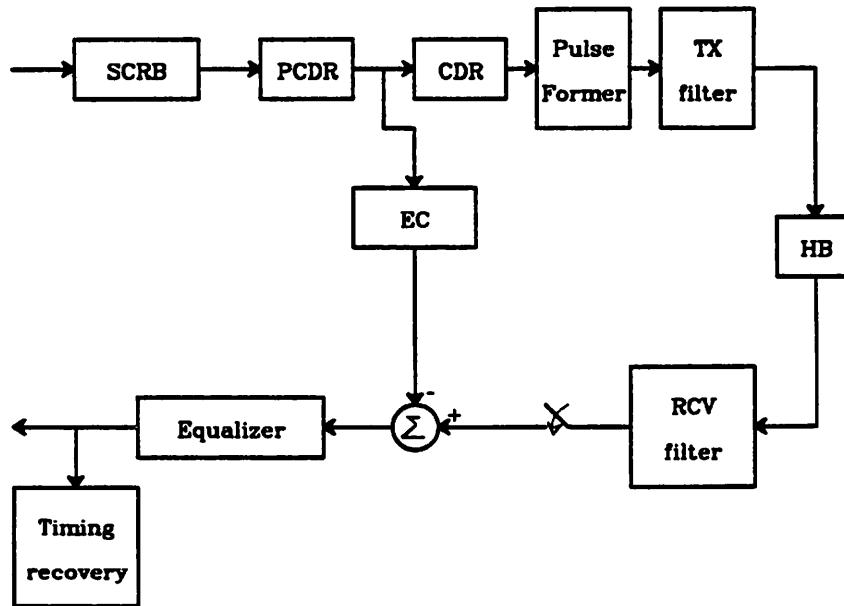
## 2.2. System Description

In this section, the various system components will be discussed. Some design considerations are also presented.

The block diagram of the system is shown in Fig. 2.2.1. The important components are the line coder, the pulse former, the transmit filter, the receive filter and the equalizer, the echo canceller, and the timing recovery circuit. Timing recovery will be treated in chapter 5.

### 2.2.1. Line Coding and Pulse Shaping

Line coding is the mapping between data bits and pulses on the line. Line coding plays an important role in data transmission system. In the DSL environment, the choice of a line code involves a tradeoff among spectral properties, crosstalk performance, information-bearing capability, synchronization (timing information), and realization complexity. In addition to the factors mentioned above, one particularly



SCRB:scrambler, PCDR:precoder, CDR:coder, EC:echo canceller.  
 Figure 2.2.1. Block diagram of the subscriber end of DSL.

important issue is the effect of the line code on the length of the echo path impulse response. This effect is critical because the length of the echo path impulse response determines the size of the echo canceller, which is a critical part in the DSL system. Fig. 2.2.1.1 shows some of the line codes and pulse shapes.

### 2.2.1.1. Spectrum shaping

The first spectral property required is a zero at DC because no DC transmission through hybrid transformers is possible. This requirement can be fulfilled by either line coding or pulse shaping techniques. For example, in biphase coding, where a one bit is transmitted as a 50% duty-cycle positive pulse followed immediately by a 50% duty-cycle negative pulse, and a zero bit is transmitted as the negative of a one bit.

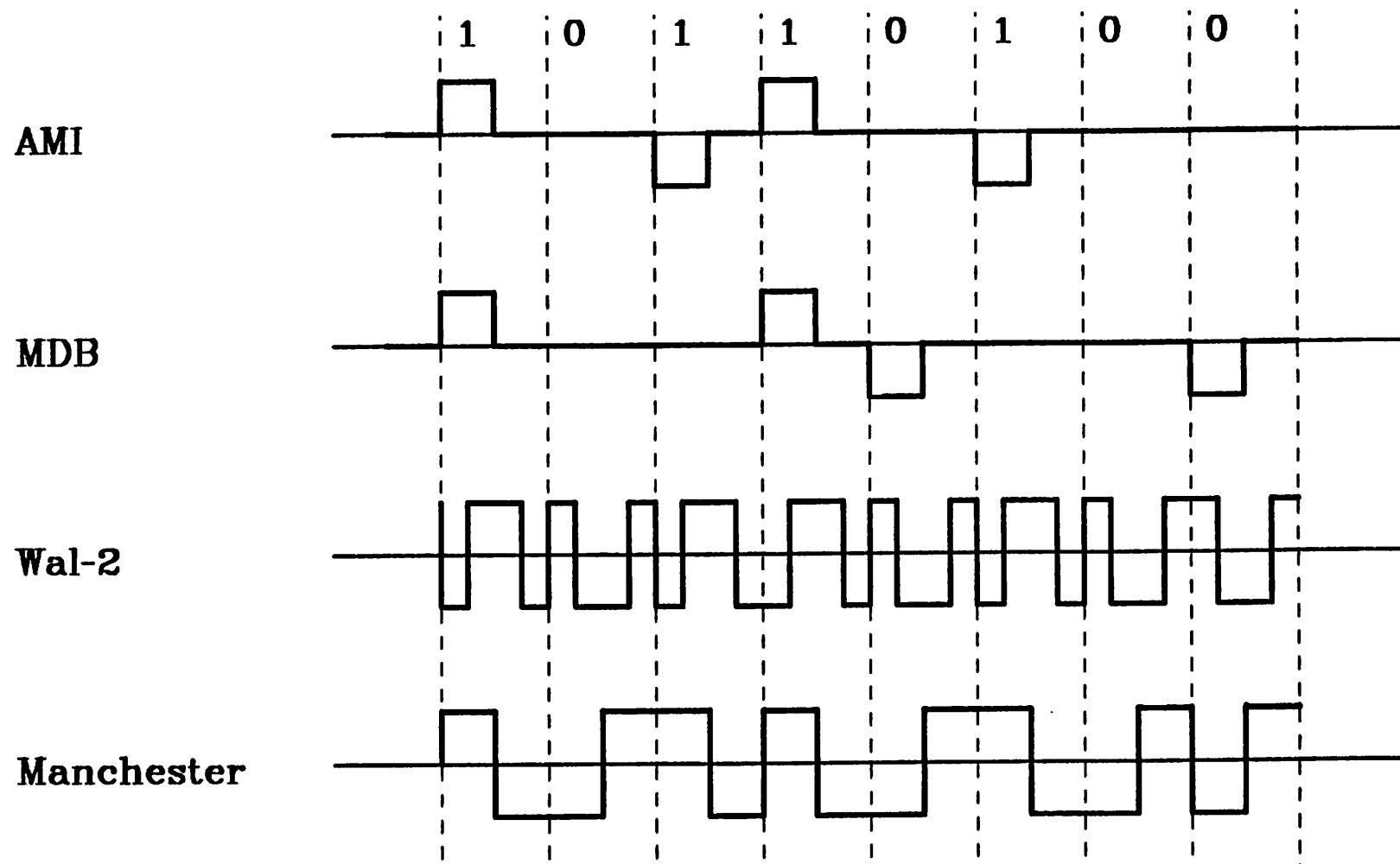


Figure 2.2.1.1 Examples of line coding and pulse shaping.

zero DC is guaranteed by pulse shaping. Some other codes suppress DC component by a coding technique. For example, AMI, a commonly used line code, is a three-level code. Pulses to be transmitted are modulated by 1, 0, or -1. A zero bit is transmitted as a zero pulse, while a one bit is transmitted alternately as a positive pulse and a negative pulse. The alternating positive and negative pulses suppress the DC component. In fact, AMI can be considered as a precoded partial response code with system polynomial  $(1 - D)$ , where  $D$  represents a baud-period delay. This  $(1 - D)$  factor, whose frequency response is periodic in magnitude with period  $f_b$ , inserts a zero at DC. MDB, a class IV partial response code whose system polynomial is  $(1 - D^2) = (1 + D)(1 - D)$ , gives a zero at DC also due to the  $(1 - D)$  factor. MMS43, a particular 4B3T block code, guarantees zero DC by monitoring the running digit sum (RDS) [6]. Zero DC is ensured by keeping the RDS bounded.

The second spectral property desired is small signal power at high frequencies. There are two major reasons behind this. Firstly, the transmission attenuation at high frequencies is more severe than at low frequencies; consequently, more equalization is needed for a system with more signal power at high frequencies. Secondly, crosstalk interference between neighboring pairs increases dramatically at high frequencies. In order to have small crosstalk power, the signal must have small power at high frequencies (See Fig. 2.1.2.2). It is obvious that those codes having small power at high frequencies end up with small near-end-crosstalk power. The reduction in signal power at high frequencies can normally be achieved by the line code. For example, MDB achieves this because of the  $(1 + D)$  factor in its system polynomial. Some block codes control their high frequency power by controlling the running alternating sum (RAS). In fact, keeping the RAS bounded gives a zero at half the baud rate. In addition to this, block codes can normally make better use of the three-level signaling than those codes like AMI and MDB. For example, 4B3T codes, which map four binary bits into three ternary symbols, reduce the baud rate by a factor of 0.75 relative to

AMI, if the framing issue is ignored. Consequently, the high frequency component is reduced.

### 2.2.1.2. Timing information

Depending on the timing recovery technique used, different line coding or pulse shaping techniques will be preferable. For example, biphase coding, which has a zero crossing in every baud period, contains much timing information and is particularly well suited for those timing recovery techniques such as the zero-crossing method. Some timing recovery techniques, such as the spectral-line method and the generalized wave-difference method using a square-law nonlinear device, rely on the power at around half the baud rate and require large enough excess bandwidth to perform timing recovery. In such cases, a zero at the Nyquist frequency is not desirable and excess bandwidth is required. However, the same technique can be used for minimum bandwidth systems if higher-order nonlinear devices are used. On the other hand, minimum excess bandwidth is intuitively beneficial for the class of baud-rate sampling timing recovery techniques since little aliasing distortion is introduced. For example, in the least mean-square timing recovery technique proposed by Qureshi, the accuracy of representing the derivative of the signal at the sampling instant by baud-spaced sampling points relies on the fact that the signal waveform can be uniquely determined by baud-spaced points. The excess bandwidth in this case will contribute aliasing distortion and thus introduce error in the estimation of the signal derivative. Therefore those systems that are realizable under zero excess bandwidth, such as MDB, are intuitively more advantageous in this case.

### 2.2.1.3. Realization complexity

The complexity of the coder of the block codes is generally higher than that of other codes. It will be shown in later chapter that the complexity of the MDB coder is

essentially the same as that of AMI coder. In fact, MDB also offers some desirable alternatives and gives good trade-off between the complexity and system performance. The complexity issue is particularly important in DSL system since the feasibility of the integrated circuit realization of the system is of great concern. Length of echo path impulse response.

#### 2.2.1.4. Length of echo path impulse response

Since the echo canceller is one of the most complicated parts in the DSL, the length of the echo path impulse response, which determines the number of echo canceller taps required, has a major influence on the overall system complexity. In general, those codes with self equalization characteristics tend to cancel the long tails due to the self equalization and thus give shorter impulse response. For example, AMI code ( $1 - D$ ) will have impulse response shorter than MDB, ( $1 - D^2$ ), since the self equalization of the ( $1 - D$ ) is more effective than that of the ( $1 - D^2$ ). Fig. 2.2.1.2 shows the echo path impulse response and frequency response of AMI and MDB systems for a 18kf. gauge AWG24 cable with the balancing impedance of the hybrid transformer adjusted to match the channel impedance at  $f = f_b$ . In fact, as discussed in section 2.1.4, the length of echo path impulse response can also be changed if a high pass filter is inserted into the echo path.

### 2.2.2. Filtering and Equalization

The transmit and receive filters are designed to minimize the ISI and the noise (white noise and crosstalk noise) penetration. Let us assume, for the time being, that the transmission line has an ideal, frequency-independent amplitude response. Then the ISI can be minimized by designing filters satisfying the Nyquist criterion. For example, the filters can be designed to approximate the raised-cosine filter response. Once the overall filter response is determined, one has to partition it between the

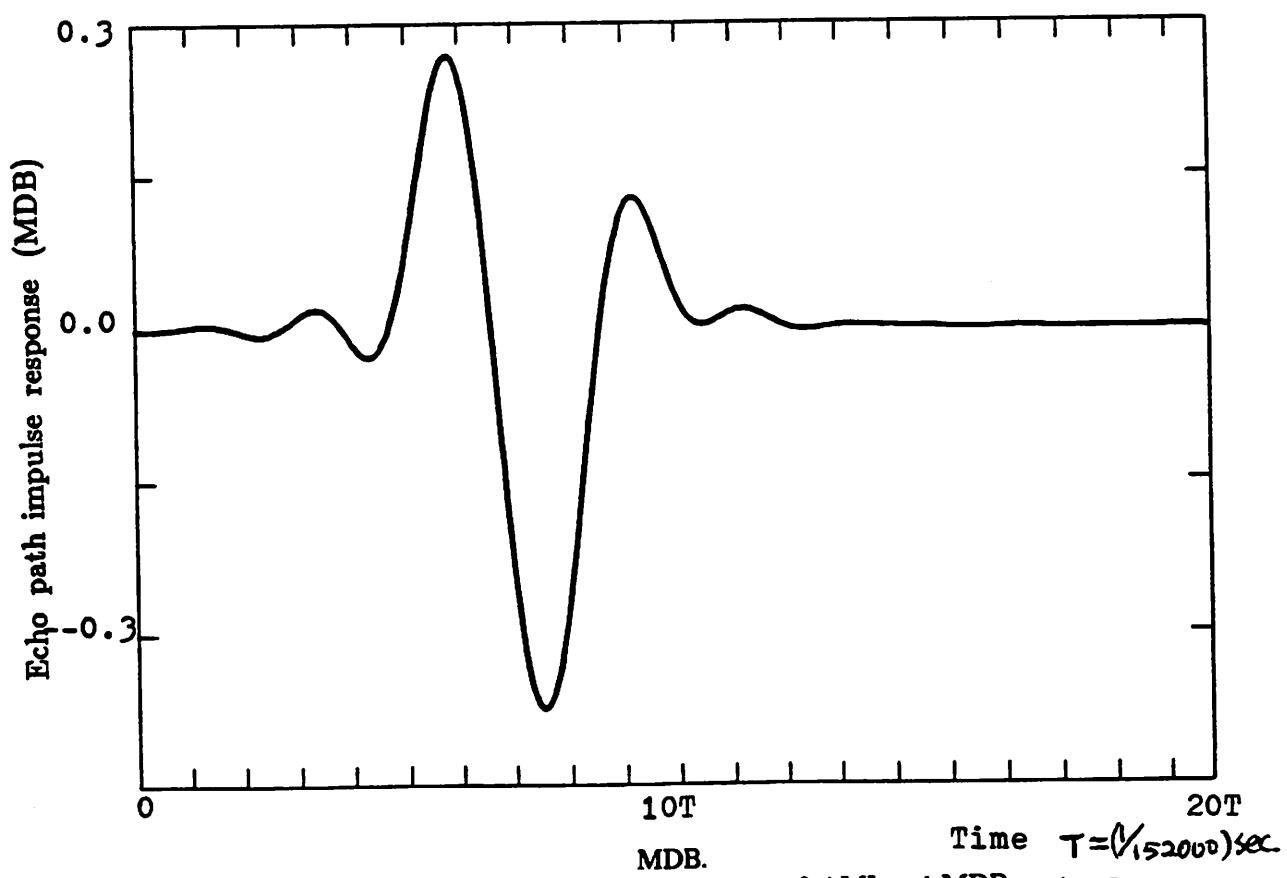
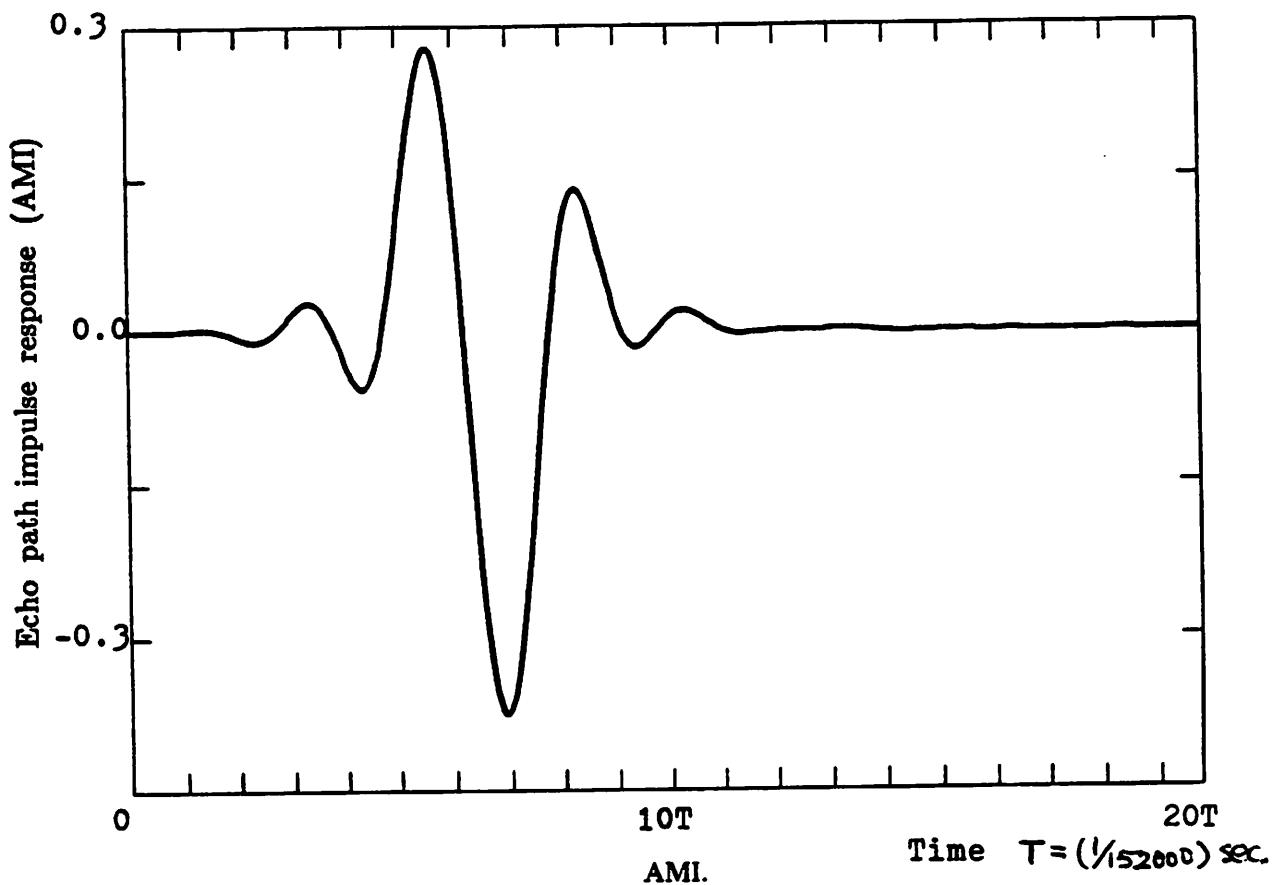


Figure 2.2.1.2. Echo path impulse responses of AMI and MDB systems.

transmitter and the receiver. The partitioning will not affect the crosstalk SNR since both the signal and the crosstalk noise go through the transmit and the receive filters and therefore depends only on the overall filter response. If we assume that the channel is ideal, the white noise SNR will be optimized if the filter response is split equally between the transmitter and the receiver since the receiver will become the matched filter in such a case. Therefore, a straightforward implementation is to partition the overall response equally between the transmitter and the receiver.

Another approach to the filter design is to find the filter whose response minimizes the mean-square ISI measured in the time domain directly, as opposed to matching the filter response in the frequency domain. This approach is shown in [7, 8].

As mentioned before, an ideal channel response is assumed in designing the filter response. However, the channel response in a data transmission system is not flat and varies from line to line. In addition, the responses are normally time varying. Therefore the filters are generally designed for a particular compromised response. The departure of any particular channel response from the compromised one will have to be compensated for by some equalization scheme that can adaptively adjust for the characteristics of various channels.

The frequency response of wire channels in general is dominated by the skin effect. Because of the skin effect, high frequency current tends to flow only in the outer portion of the conductor. This results in an increase in the impedance as frequency increases. The frequency response can roughly be described by:

$$C(f) = C(0) e^{\alpha \sqrt{f}} \quad (2.2.2.1)$$

where  $l$  is the length of the cable,  $C(0)$  is the DC response of the cable and  $\alpha$  is an empirical constant. This corresponds to an attenuation in dBs proportional to the square-root of frequency. Fig. 2.2.2.1 shows the response of a 5 km gauge AWG24 cable.

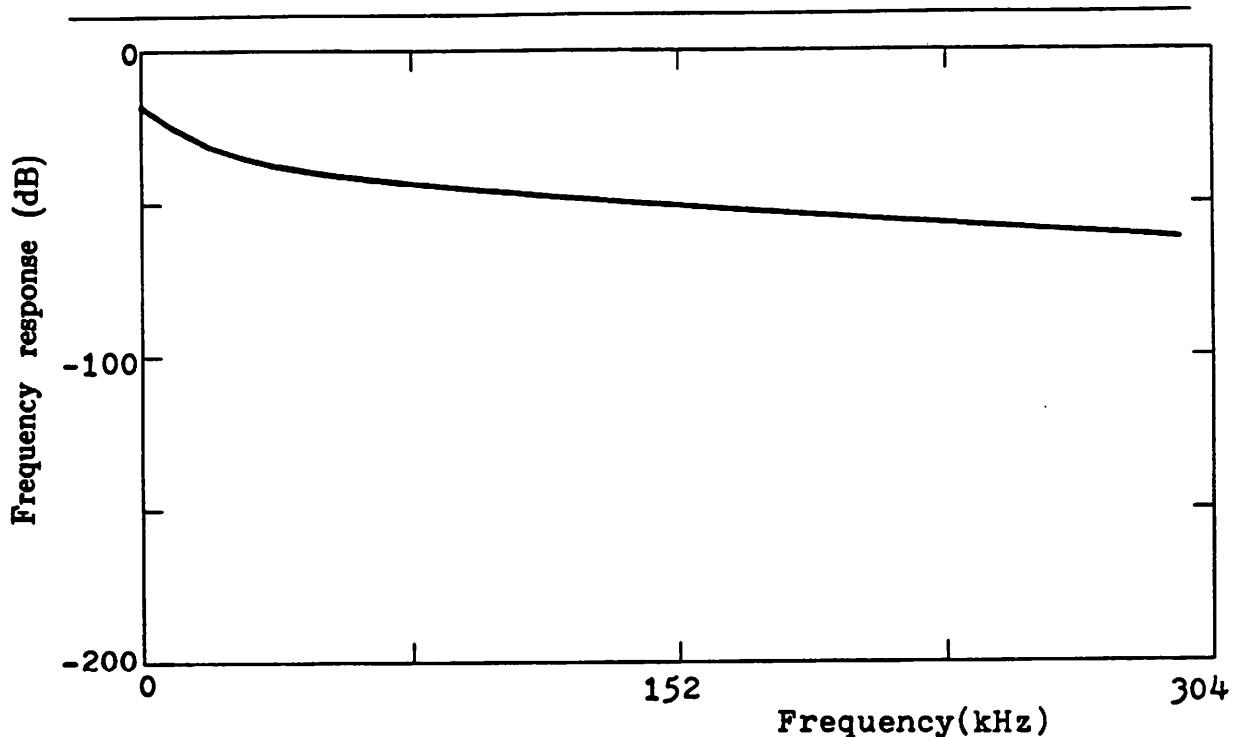
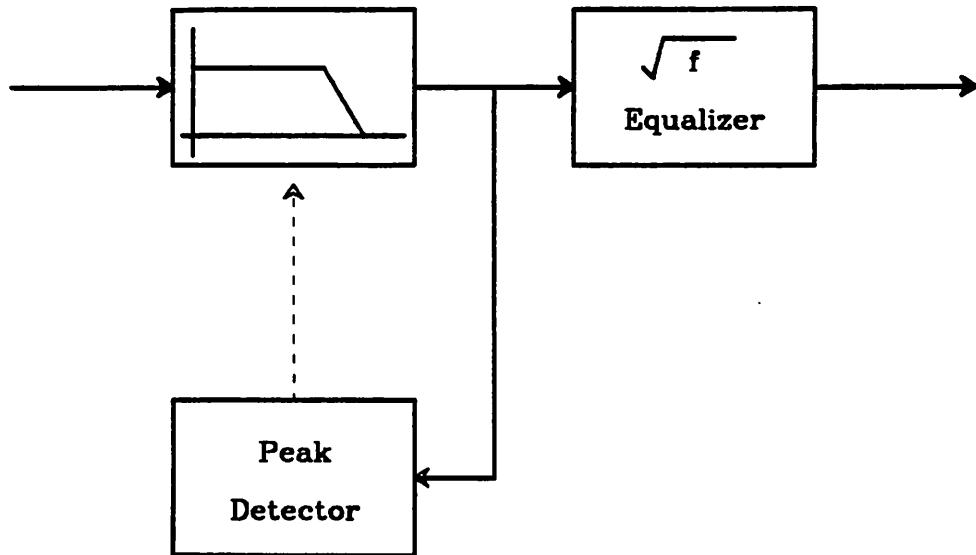


Figure 2.2.2.1. Frequency response of a 5 km gauge AWG24 cable.

Knowing the dominant attenuation factor, a coarse equalization scheme that compensates for the channel attenuation is the  $\sqrt{f}$  equalization. This can be a fixed equalizer whose dB response increases proportional to the square root of frequency. However, the attenuation of the cable depends on its length, and the length of the cable varies in the subscriber loop systems. A purely fixed square-root equalizer alone cannot do the job. One solution is to combine the  $\sqrt{f}$  equalizer with *automatic line build out* (ALBO). This is shown in Fig. 2.2.2.2. In this scheme, the  $\sqrt{f}$  equalizer is designed to equalize the longest line and ALBO is used to compensate for the difference in the line length from the longest line by adjusting the pole locations of a single-pole low-pass filter. In other words, ALBO is adjusted so that the response of the cascade of the cable with ALBO is approximately equal to the response of the longest line. Since the

Figure 2.2.2.2. ALBO and  $\sqrt{f}$  equalizer.

received pulse is spread out and its peak value decreases as the cable length increases. the pole location can be adjusted according to the output of a peak detector.

However, the above equalization ( $\sqrt{f}$  and ALBO) only gives a coarse compensation of the channel response. In addition to that, the cable attenuation also changes with temperature and is heavily affected by bridge taps (BT), open-ended wire pairs attached to the main subscriber loops. The existence of the BT normally affects the trailing part of the impulse response. In fact, the resultant time response is the superposition of the original BT-free pulse with a delayed amplitude-attenuated pulse reflected from the open end of the BT. Fig. 2.2.2.3.a is the time responses of a 3.48km. gauge AWG24. BT-free pulse and the response of the same length, but with 1.579km BT attached cable. In the frequency domain, BT causes dips in the frequency response

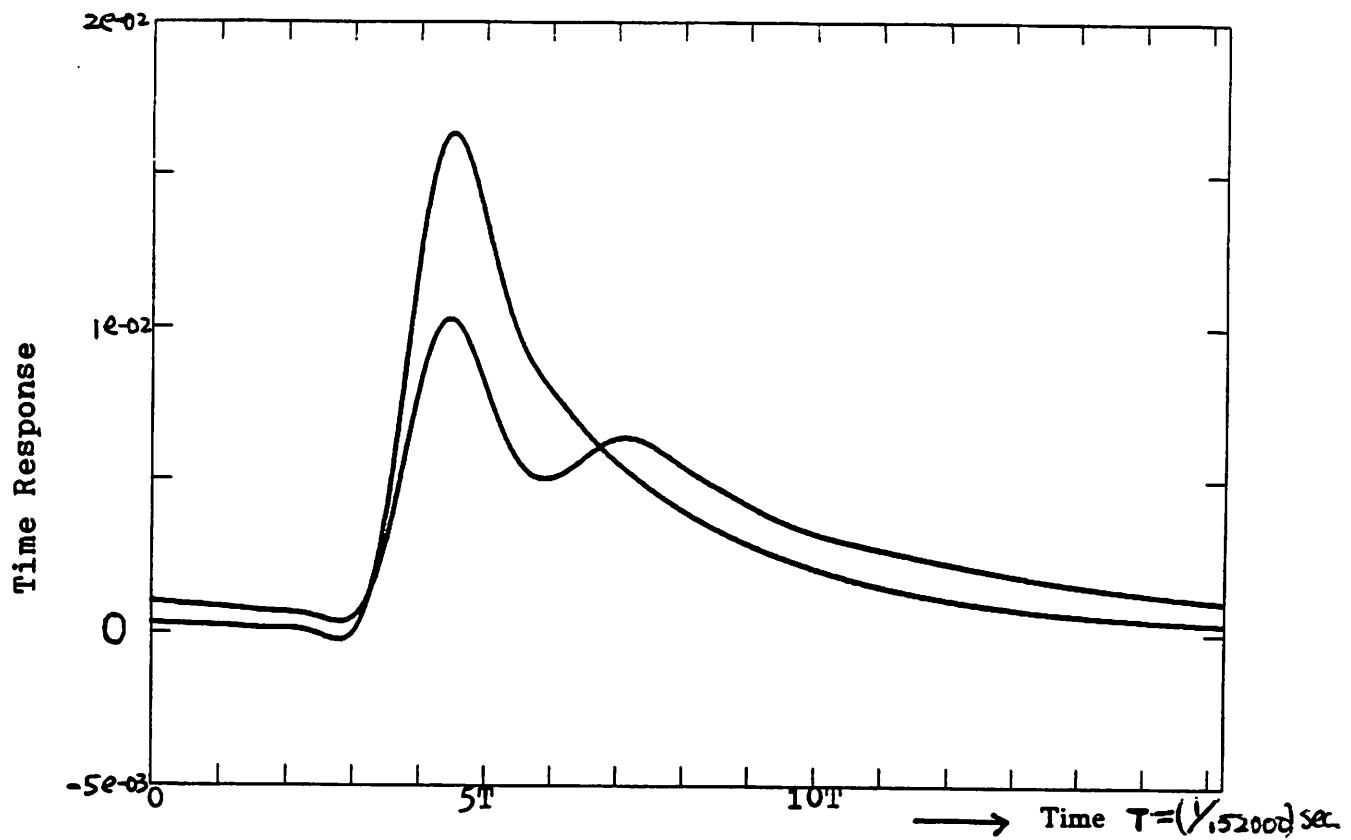


Figure 2.2.2.3.a Time responses of 3.48km gauge AWG24 cable with or without BT.

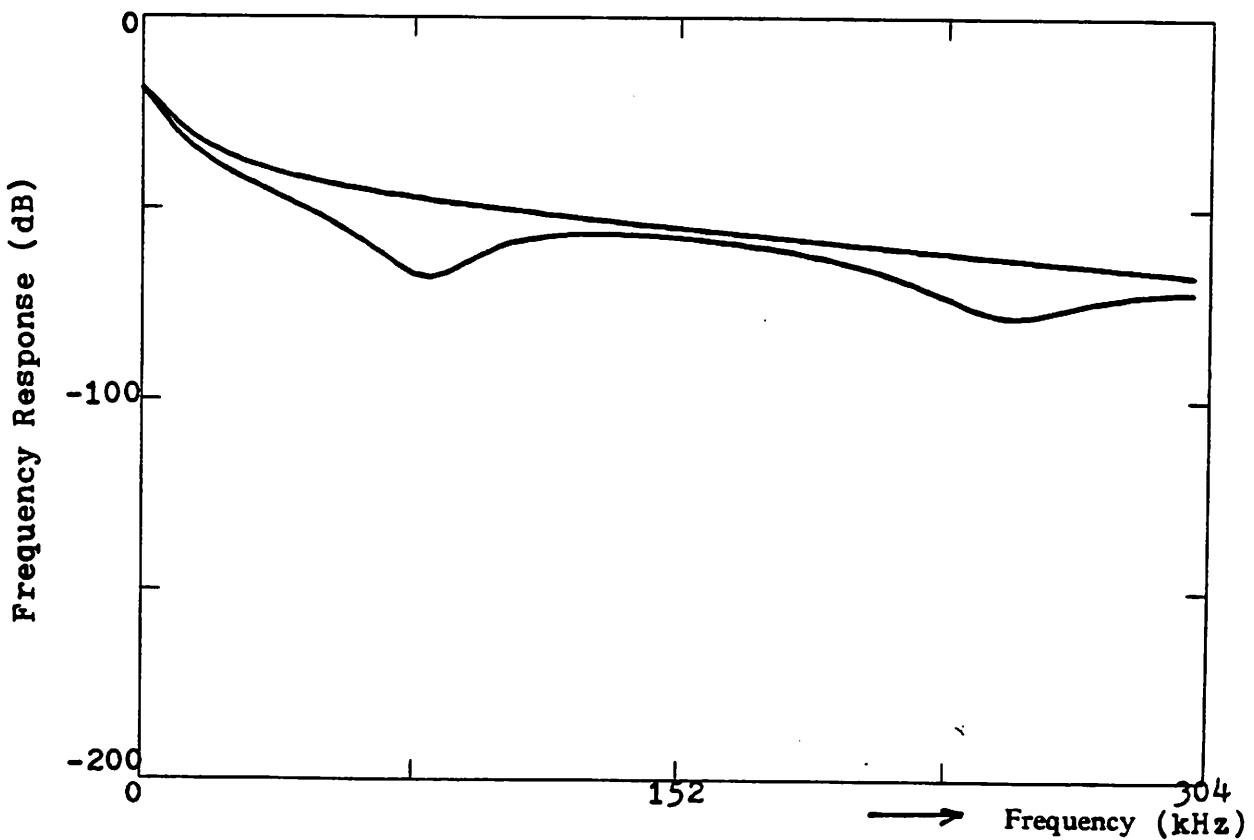


Figure 2.2.2.3.b Frequency responses of 5.48km, gauge AWG24 cable with or without BT.

at certain frequencies. The frequencies affected are determined by the length as well as the location of the BT. Fig. 2.2.2.3.b shows the response of a 5.48km. gauge AWG24 cable and the response of the same cable with two BTs.

More sophisticated and effective equalization schemes are needed in general. Two commonly used adaptive equalization techniques are *linear equalization (LE)* and *decision-feedback equalization (DFE)*.

A linear equalizer compensates for the nonideal channel response linearly. A block diagram of a linear equalizer is shown in Fig. 2.2.2.4 where  $T$  represents one baud-period delay. The output  $y_n$  of the LE given the input sequence  $\{x_n\}$  is

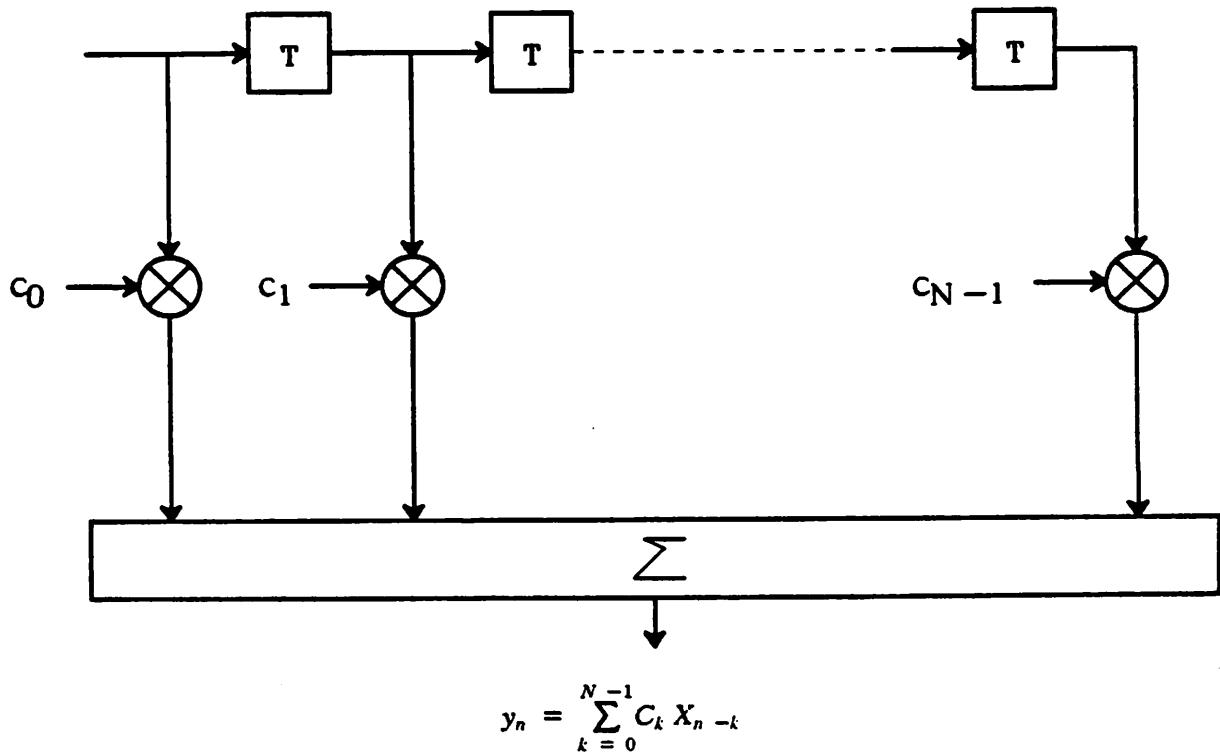


Figure 2.2.2.4. Linear equalizer.

$$y_n = \sum_{k=0}^{N-1} c_k x_{n-k} \quad (2.2.2.2)$$

and its transfer function is:

$$C(\omega) = \sum_{k=0}^{N-1} c_k e^{-j\omega kT} \quad (2.2.2.3)$$

Its response is periodic with period  $\frac{2\pi}{T}$ .

Note that since the input  $x_n$  is an analog signal (or quantized analog signal), there are multiplications involved in performing the convolutions ( $\sum c_k x_{n-k}$ ). The number of multiplications required per baud interval is equal to the number of linear equalizer taps. Since multipliers are expensive in terms of hardware implementation cost, it is undesirable to use a LE with many taps.

Another disadvantage of the LE is the noise enhancement problem. Since the LE compensates for the channel response linearly by having large gain at those frequencies where the channel attenuation is severe, it therefore amplifies the noise. This problem is worse if the channel has a deep in-band value which requires large amplification. This undesired property can be avoided if nonlinear equalizers are used.

The optimum equalizer coefficients can be determined if the channel response is known. We will derive the optimum coefficients in the following subsections. However, the channel response is not known in general and is normally time varying. In such case, the coefficients can be obtained adaptively. For example, in one commonly used adaptive algorithm, the *stochastic gradient algorithm*, the coefficients are adjusted according to the equation:

$$\bar{C}^{(n+1)} = \bar{C}^n + \beta e_n \bar{X}_n \quad (2.2.2.4)$$

where  $\bar{C}^n$  is the coefficient vector at time n:

$$\bar{C}^n = \{c_0^n, \dots, c_{N-1}^n\}$$

$\beta$  is the step size of adaptation,  $\bar{X}_n$  is the received signal vector:

$$\vec{X}_n = \{x_n, \dots, x_{n-N+1}\}$$

and  $e_n$  is the adaptation error:

$$e_n = y_n - \sum_{k=0}^{N-1} g_k \hat{a}_{n-k} \quad (2.2.2.5)$$

and  $\hat{a}$  is the estimated data,  $g_k$  is the desired response. For ordinary transmission,  $g_k = 0$  for any  $k \neq 0$ .

The derivation of the optimum filter coefficients of the linear receiver and the receiver with DFE for minimizing the ISI and the additive Gaussian noise can be found in the literature[9, 10, 11]. Although there are many techniques for reducing the ISI, one common point for all of them is that they start with a filter matched to  $h(t)$ , the overall impulse response seen at the input of the receiver, followed by a sampler operating at the baud rate  $\frac{1}{T}$ . This is shown in Fig. 2.2.2.5.a. Although this matched filter is optimum only for white noise environment, colored noise, such as the crosstalk interference, can be similarly treated as shown in Fig. 2.2.2.5.b. The  $\frac{1}{\sqrt{N(\omega)}}$  filtering whitens the colored noise, and the response of the matched filter, now matching to the response  $\frac{H(\omega)}{\sqrt{N(\omega)}}$ , is  $\left[ \frac{H(\omega)}{\sqrt{N(\omega)}} \right]^* = \frac{H^*(\omega)}{\sqrt{N(\omega)}}$  where  $H^*(\omega)$  denotes the complex conjugate of  $H(\omega)$ . Thus, without loss of generality, we can concentrate on the white noise environment.

The samples at the output of the sampler form a set of sufficient statistics for the estimation of data sequence. Different techniques would differ in the subsequent processing method for estimating the data sequence. First, we look at the optimum linear equalization.

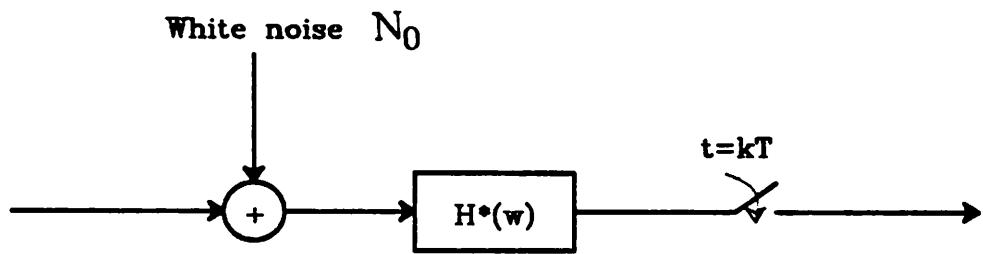


Figure 2.2.2.5.a. Matched filter for white noise.

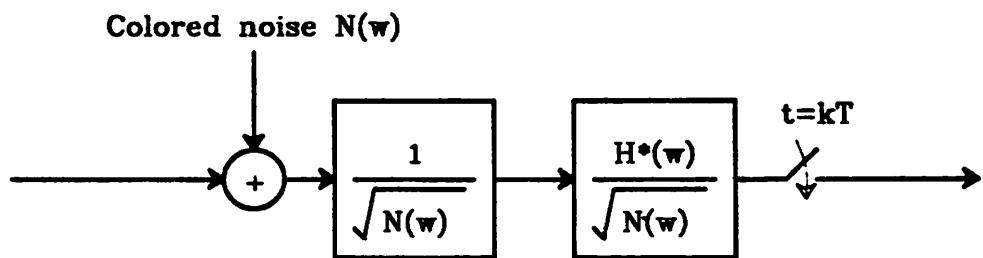


Figure 2.2.2.5.b. Matched filter for colored noise.

### 2.2.2.1. Optimum linear equalizer

Fig. 2.2.2.1.1 is the block diagram of the optimum linear receiver where  $H^*(\omega)$  is the matched filter and

$$C(z) = \sum_{n=0}^{N-1} c_n z^{-n} \quad (2.2.2.1.1)$$

is the response of the linear equalizer with  $z = e^{j\omega T}$ . There are two different criteria for determining the coefficients of the linear equalizer. They are:

- (1) Zero forcing (peak distortion) and
- (2) Mean-square-error criterion.

The objective of the zero-forcing criterion is to choose the coefficients such that the peak ISI distortion is minimized. A natural choice is to satisfy Nyquist criterion. Let  $r_k$  be the kth sample of the sampled impulse response:

$$r_k = \int_{-\infty}^{\infty} h(\tau)h(\tau-kT)d\tau \quad (2.2.2.1.2)$$

and

$$R(z) = \sum_{k=-\infty}^{\infty} r_k z^{-k} \quad (2.2.2.1.3)$$

Thus,  $R(z)|_{z=e^{j\omega T}}$  is the frequency response looking at the output of the sampler and is equal to the folded spectrum of  $h(t)*h(-t)$  where \* represents convolution.

Since the frequency response of  $h(t)*h(-t)$  is  $H(\omega)H^*(\omega) = |H(\omega)|^2$ , we have

$$R(e^{j\omega T}) = \frac{1}{T} \sum_k |H(\omega + k \frac{2\pi}{T})|^2 \quad (2.2.2.1.4)$$

In order to satisfy the Nyquist criterion, the folded spectrum after the linear equalizer should equal 1, i.e.:

$$\begin{aligned} R(z)C(z) &= 1 \\ \Rightarrow C(z) &= \frac{1}{R(z)} \end{aligned} \quad (2.2.2.1.5)$$

The resultant system is shown in Fig. 2.2.2.1.2. As shown in this figure, the impulse response after the matched filter is symmetrical with respect to its peak, and becomes ISI free after the linear equalizer.

Although the choice of  $C(z)$  completely eliminates the ISI, it also enhances the noise. This is because the choice of  $C(z)$  is to equalize the overall response, i.e.,

$$\begin{aligned} C(e^{j\omega T}) &= \frac{1}{R(e^{j\omega T})} \\ &= \frac{T}{\sum_k |H(\omega + k \frac{2\pi}{T})|^2} \end{aligned} \quad (2.2.2.1.6)$$

And the power spectrum of the filtered white noise is now proportional to:

$$\frac{1}{\sum_k |H(\omega + k \frac{2\pi}{T})|^2}$$

Thus, any dip in  $H(\omega)$  will enhance noise.

Another criterion in determining the coefficients is the *mean-square error criterion*. In this criterion, the coefficients are adjusted to minimize the mean-square error.

In Fig. 2.2.2.1.3, let  $C(z)$  be the linear equalizer with the mean square-error criterion. The error at the  $k$ th sample is  $e_k$ :

$$e_k = \hat{a}_k - y_k \quad (2.2.2.1.7)$$

where  $\hat{a}$  is the decision and  $y$  is the input signal to the slicer:

$$y_k = \sum_{j=-\infty}^{\infty} c_j x_{k-j} \quad (2.2.2.1.8)$$

and

$$x_k = \sum r_j a_{k-j} + n_k \quad (2.2.2.1.9)$$

$r_j$ , as defined in Equation (2.2.2.1.2), is overall sampled impulse response before the linear equalizer and

$$n_k = \int n(\tau) h(\tau - kT) d\tau \quad (2.2.2.1.10)$$

is the noise at the input to the linear equalizer. If we assume there is no decision error,

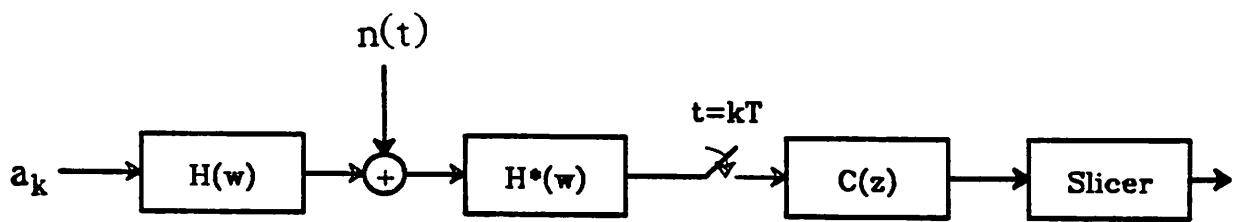
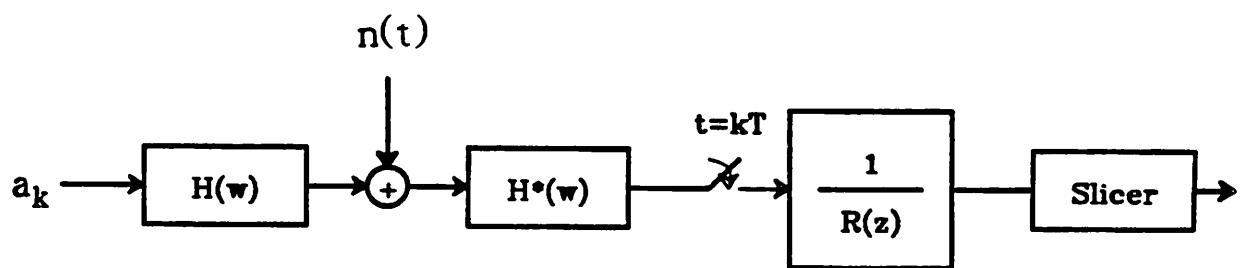
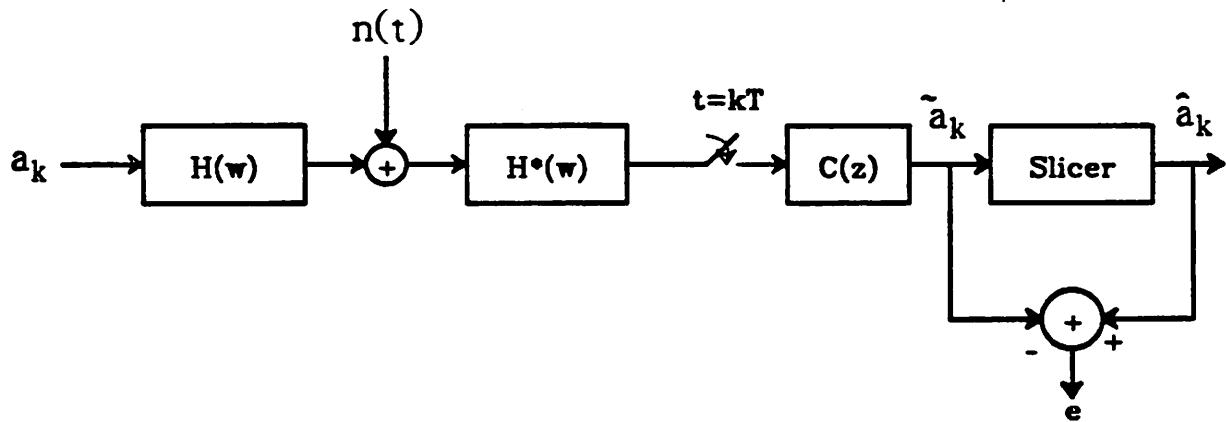


Figure 2.2.2.1.1. Optimum linear receiver.



$$R(e^{j\omega T}) = \frac{1}{T} \sum_k |H(\omega + k \frac{2\pi}{T})|^2$$

Figure 2.2.2.1.2. Optimum zero-forcing linear receiver.



$$C(z) = \frac{1}{R(z) + \frac{N_0}{S}}$$

Figure 2.2.2.1.3. Minimum MSE linear equalizer.

i.e.,  $\hat{a}_k = a_k$ , then Equation (2.2.2.1.7) becomes

$$e_k = a_k - y_k \quad (2.2.2.1.11)$$

By the orthogonality principle,

$$E[e_k x_{k-l}] = 0 \quad -\infty < l < \infty \quad (2.2.2.1.12)$$

$$\Rightarrow E[(a_k - \sum_j c_j x_{k-j}) x_{k-l}] = 0 \quad (2.2.2.1.13)$$

$$\Rightarrow \sum_j c_j E[x_{k-j} x_{k-l}] = E[a_k x_{k-l}] \quad (2.2.2.1.14)$$

To evaluate the moments in (2.2.2.1.14), we use  $r_k$  in (2.2.2.1.9). Thus

$$E[x_{k-j} x_{k-l}] = E[(\sum_i r_i a_{k-j-i} + n'_{k-j})(\sum_m r_m a_{k-l-m} + n'_{k-l})]$$

$$= S \sum_m r_m r_{m+l-j} + N_o r_{j-l} \quad (2.2.2.1.15)$$

where  $S = E[a_k^2]$  and  $N_o = E[n^2(t)]$ . and

$$E[a_k x_{k-l}] = S r_{-l} \quad (2.2.2.1.16)$$

By substitute (2.2.2.1.15) and (2.2.2.1.16) into (2.2.2.1.14) and taking the z transform of both sides, we obtain

$$C(z)[R(z)R(z^{-1}) + \frac{N_o}{S}R(z)] = R(z) \quad (2.2.2.1.17)$$

Therefore, we have

$$C(z) = \frac{1}{R(z) + \frac{N_o}{S}} \quad (2.2.2.1.18)$$

It is interesting to compare (2.2.2.1.18) with (2.2.2.1.5). The only difference between these two criteria is the presence of the  $\frac{N_o}{S}$  term in the denominator of Equation (2.2.2.18). When the noise is small compared with signal, i.e.,  $\frac{N_o}{S} \rightarrow 0$ , the coefficients given by the MSL criterion are close to those given by the peak distortion criterion. In fact, when the noise is small, the only error to be minimized is ISI. Therefore, it makes sense that the criterion minimizing the peak ISI distortion gives the same results as the one minimizing the mean-square error.

### 2.2.2.2. Slightly nonlinear receiver - DFE

Another equalization scheme is called *decision feedback equalization (DFE)*. It is shown in Fig. 2.2.2.2.1. Based on the past estimations of the transmitted data, the predicted ISI =  $\sum_{k=1}^K d_k \hat{a}_{n-k}$  is subtracted from the received signal  $\sum_{k=-\infty}^{\infty} h_k a_{n-k}$ .

This presumably ISI-reduced signal is then passed through the slicer to recover the transmitted data. Several points are worth mentioning here. Since the prediction of the ISI is based on the past decisions, the DFE is effective only in eliminating post-

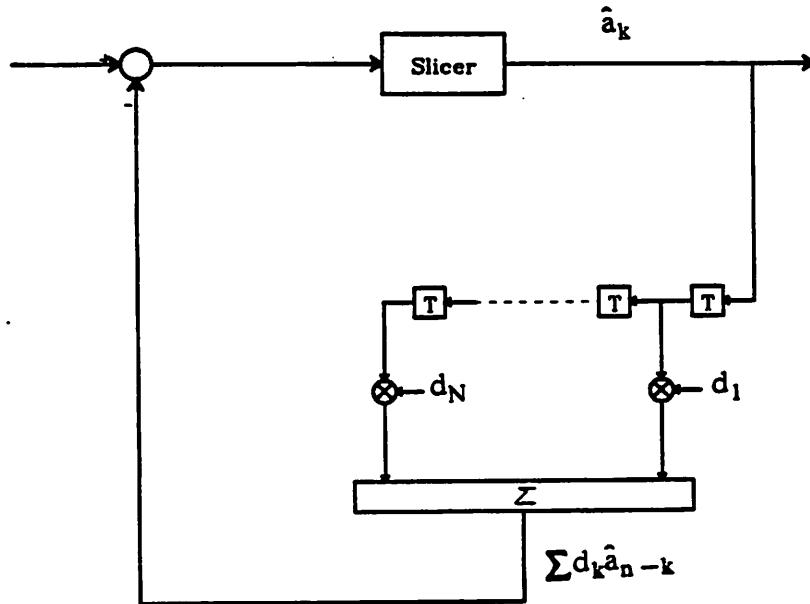


Figure 2.2.2.2.1. Decision feedback equalizer.

cursor ISI and cannot reduce the precursor ISI. Secondly, the signal  $\hat{a}_k$  in the tapped delay line belongs either to the set  $\{1,-1\}$  or  $\{1.0,-1\}$ . Thus no multiplication is needed in computing the predicted ISI  $\sum d_k \hat{a}_{n-k}$ . This is important when we compare this scheme with that of the LE where one multiplication per tap is required. Therefore, the DFE is particularly suitable for modern VLSI implementation. Also note that the signal fed to the tapped delay line of the DFE is the output of the slicer, a nonlinear device. This scheme is therefore a nonlinear equalization scheme. Because of this nonlinearity, the input to the tapped delay line (the recovered far-end data symbols) is noise free if no prediction error is made. Therefore, noise is not being fed back and thus the DFE does not have any filtering effect on the additive white noise. Consequently, there is no noise enhancement, as opposed to the LE case where noise enhance-

ment is one of the major concerns.

The optimum DFE receiver is shown in Fig. 2.2.2.2.2. As in the LE case, the received signal is prefiltered by the matched filter and then sampled at the period  $T$ . A forward equalizer is placed before the DFE to manipulate the overall sampled impulse response such that the response at the output of the forward equalizer becomes precursor ISI free. The remaining post-cursor ISI can be completely removed by the DFE.

The DFE scheme has smaller noise enhancement than the LE scheme. This fact can be explained heuristically by examining the DFE scheme in two parts. First, note that the response after the forward equalizer is spread out in time and corresponds to a narrow frequency bandwidth. The narrow bandwidth associated with the front end

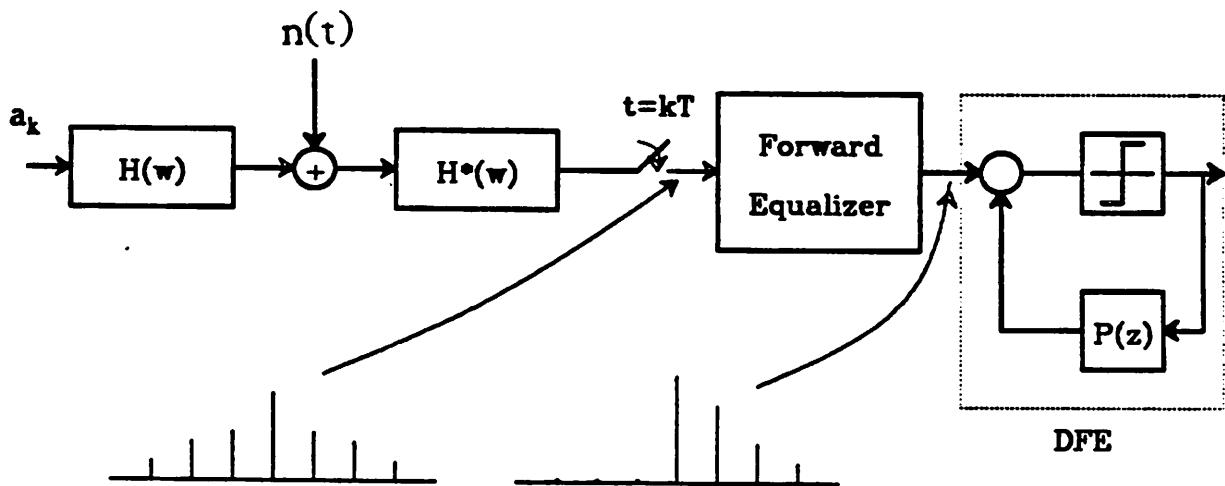


Figure 2.2.2.2.2. Optimum DFE receiver.

of the DFE scheme allows less noise penetration than the wide bandwidth (and narrow time response) associated with the entire LE scheme. Second, as explained before, the remaining portion of the DFE scheme has no noise enhancement due to its nonlinear characteristics. Thus the overall noise enhancement of the DFE scheme is smaller than that of the LE scheme.

As in the LE case, there are two alternative criteria, zero forcing and mean-square-error, for determining the coefficients of the decision feedback equalizer. We now derive the optimum DFE receiver for the zero forcing criterion. Fig. 2.2.2.2.3 again shows the optimum zero-forcing linear receiver, which completely eliminates ISI. Notice however that the noise at the slicer input is not white. Thus, by exploiting the correlation of the noise, we can reduce the noise variance by estimating the future noise based on the knowledge of the past noise samples. This is shown in Fig. 2.2.2.2.4 where the decision  $\hat{a}_k$  is fed back and subtracted from the signal  $a_k + n'_k$ . The result  $n''_k$ , assuming no decision error ( $\hat{a}_k = a_k$ ), is fed to a strictly causal but infinite-tap predictor  $P(z)$ :

$$P(z) = \sum_{i=1}^{\infty} p_i z^{-i}$$

The predicted noise sample is then subtracted from the signal to give a result  $a_k + n''_k$ . Since the number of taps of the predictor is infinite, the resultant noise sample  $n''$  is white. Fig. 2.2.2.2.4 can be rearranged into Fig. 2.2.2.2.5. To determine  $P(z)$ , we use the fact that  $n''$  is white:

$$\begin{aligned} S_n(z)(1 - P(z))(1 - P(z^{-1})) &= S_{n''}(z) \\ &= \text{constant} \end{aligned} \tag{2.2.2.2.1}$$

where  $S_n(z)$  is the spectrum of the noise  $n'$  and  $S_{n''}(z)$  is the spectrum of  $n''$ . Also

$$S_{n''}(z) = S_n(z) \frac{1}{R(z)} \frac{1}{R(z^{-1})}$$

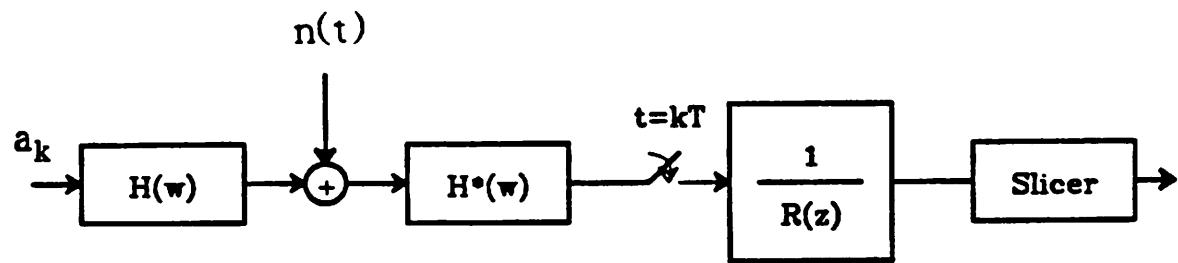


Figure 2.2.2.2.3. Optimum zero-forcing linear receiver.

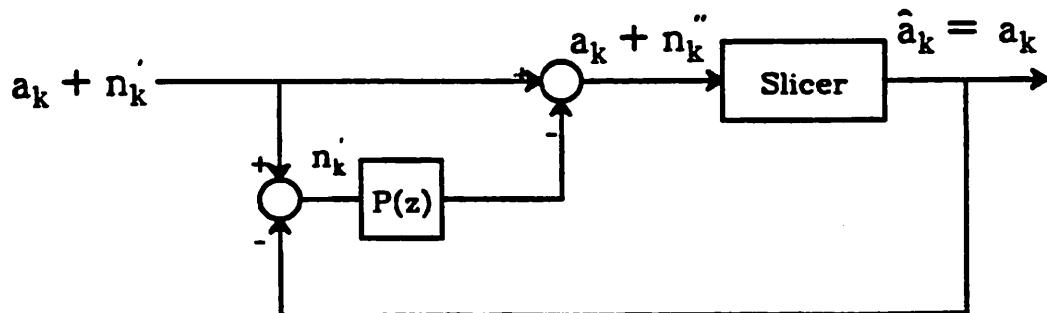
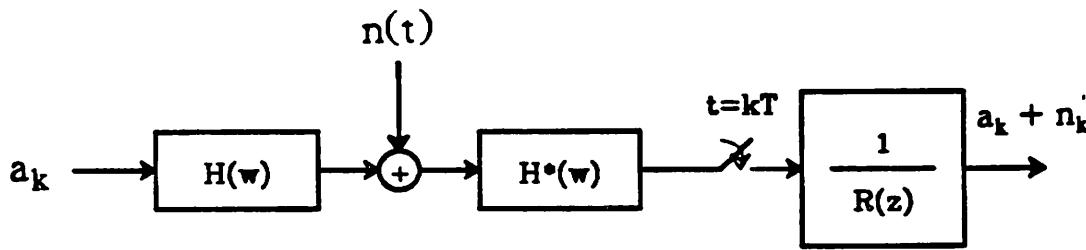


Figure 2.2.2.2.4. Optimum zero-forcing DFE receiver.

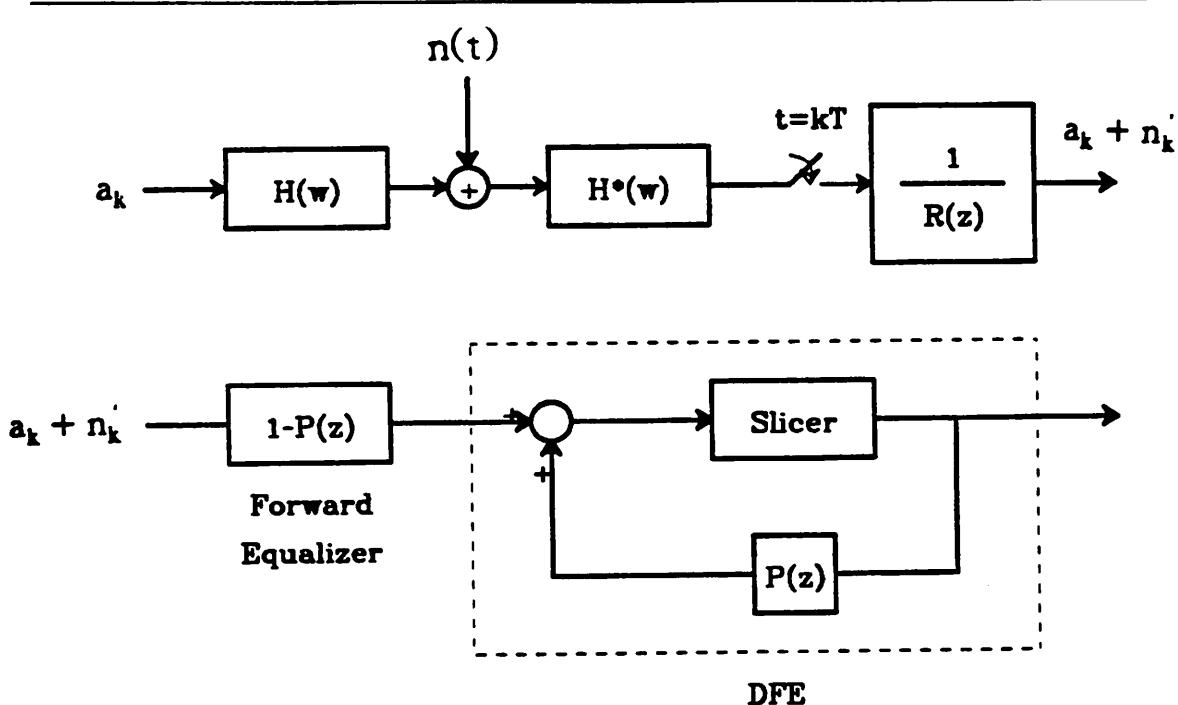


Figure 2.2.2.2.5. Optimum zero-forcing DFE receiver.

$$\begin{aligned}
 &= \frac{N_o}{2} R(z) \frac{1}{R(z)} \frac{1}{R(z^{-1})} \\
 &= \frac{N_o}{2} R^{-1}(z^{-1})
 \end{aligned} \tag{2.2.2.2}$$

Since  $r_k = r_{-k}$ , thus,  $R(z) = R(z^{-1})$ . we have

$$S_n = \frac{N_o}{2} R^{-1}(z) \tag{2.2.2.3}$$

Substituting (2.2.2.2.3) into (2.2.2.2.1), we have

$$(1 - P(z))(1 - P(z^{-1})) = R(z) \tag{2.2.2.4}$$

Since  $R(z) = R(z^{-1})$ ,  $R(z)$  will have poles  $\frac{1}{\rho_1}, \dots, \frac{1}{\rho_N}$  if  $\rho_1, \dots, \rho_N$  are poles of  $R(z)$ . Therefore

$$R(z) = \sum_{j=-\infty}^{\infty} r_j z^{-j}$$

$$= (1 - \sum_{j=1}^{\infty} p_j z^{-j})(1 - \sum_{j=1}^{\infty} p_j z^j) \quad (2.2.2.2.5)$$

and  $P(z)$  is chosen to be strictly causal:

$$P(z) = \sum_{j=1}^{\infty} p_j z^{-j} \quad (2.2.2.2.6)$$

Thus, Fig. 2.2.2.2.5 now becomes Fig. 2.2.2.2.6. Notice that the forward equalizer

$\frac{1}{1-P(z^{-1})}$  is strictly anticausal. This agrees with the previous presumption that the purpose of the forward equalizer is to remove the precursor ISI. Also important is the fact that the cascading of the matched filter and the forward filter is equivalent to an all-pass filter (in the sampled-data domain). This is justified if we recall that the purpose of the filters preceding DFE is to eliminate the precursor ISI without excessive

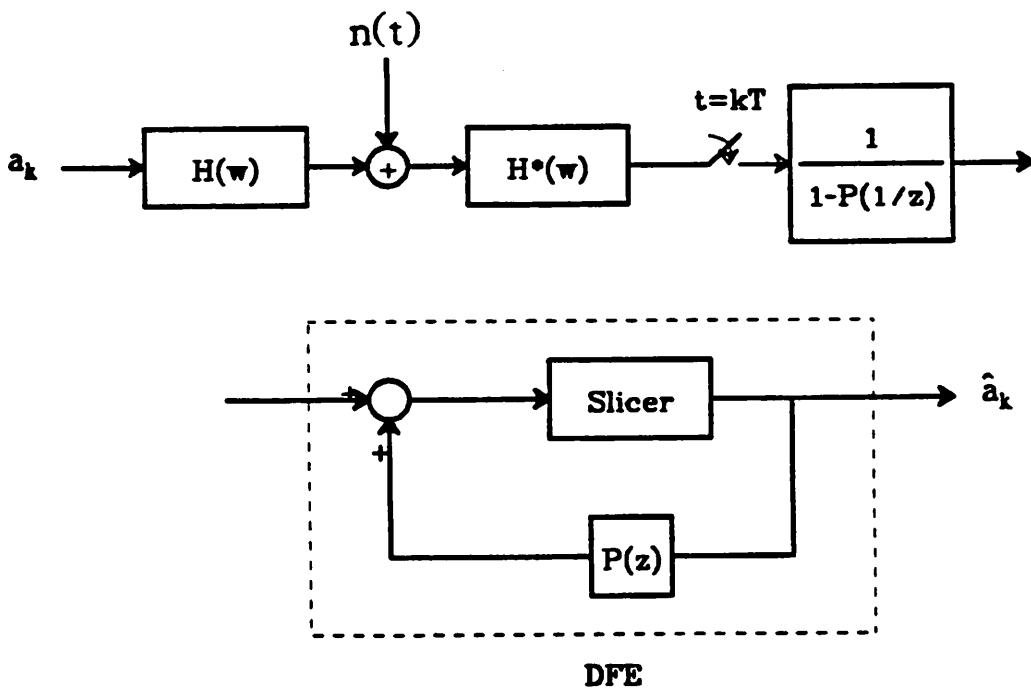


Figure 2.2.2.2.6. Optimum zero-forcing DFE receiver.

noise enhancement. This can be achieved by an all-pass filter with the desired phase response. The minimum noise variance of this scheme is given by:

$$\exp \left\{ \frac{T}{2\pi} \int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} \ln \left( \frac{N_0}{R(e^{j\omega t})} \right) d\omega \right\} \quad (2.2.2.2.7)$$

### 2.2.2.3. Practical DFE receiver

In the previous sections, we focused on the discussion of the optimum LE and DFE schemes. Since an infinite number of taps are required and matched filters are used in these schemes, they are not practical for implementation. In this section, we look at a more practical DFE receiver where the number of taps is finite and a low pass filter is used in place of the matched filter. A detailed discussion can be found in [12]. Let the system be as shown in Fig 2.2.2.2.7. In this system, let the sampled impulse response after the low-pass filter be  $\{x(k), k = 0, 1, \dots, M\}$ . A  $M$ -tap decision feedback equalizer will be needed to eliminate the ISI. However, if the decision feedback equalizer is used solely without a forward equalizer, it will define the first sample, i.e.,  $x(0)$ , as the main lobe and treat those other samples  $x(k), k = 1, 2, \dots, M$ , as post-cursor ISI and eliminate them. This results in large SNR loss since the first sample is normally small because the main lobe is normally preceded by some small-magnitude precursor samples. To overcome this problem, the DFE is preceded by a forward equalizer. Let the number of taps of the forward equalizer be  $N + 1$ . The sampled response at the output of the forward equalizer should be composed of  $M + N + 1$  samples in general (( $M+1$ )-sample response convolves with  $(N+1)$ -tap equalizer). However, we expect the first  $N$  points to be zero and the  $(N + 1)th$  sample to be one (or whatever the desired "1" level is) for an ideal system without noise added, if the equalizers function as desired. This is indeed the case as we will see later.

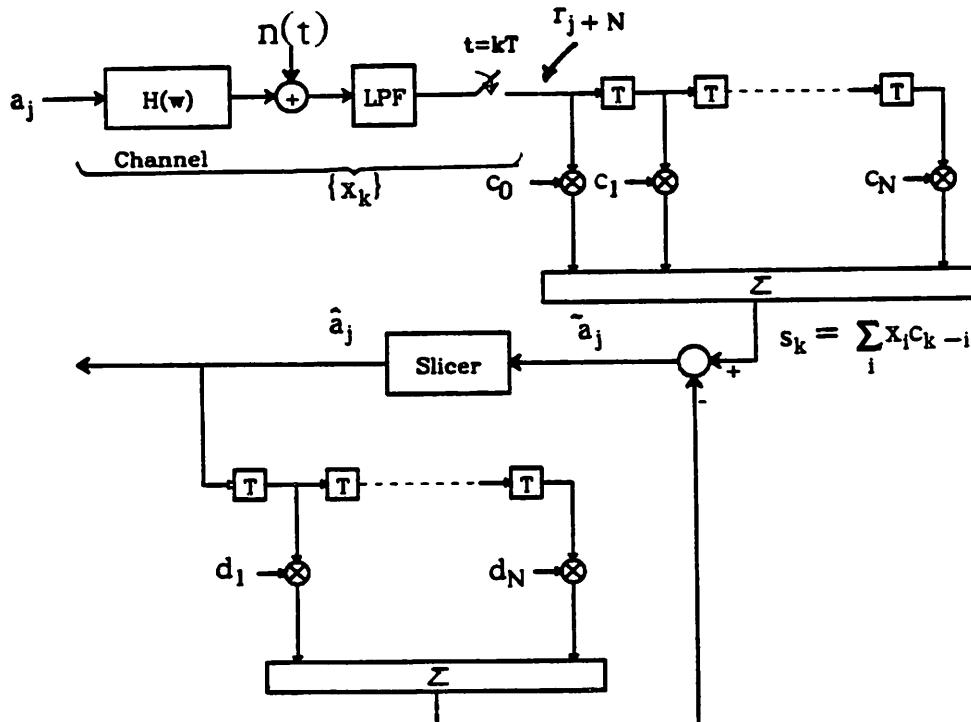


Figure 2.2.2.2.7. Practical DFE receiver.

Let the transmitted data be  $a(j)$  and the sample at the output of sampler be  $r(j)$ :

$$r(j) = \sum_{i=0}^M a(j-i)x(i) + n'(j) \quad (2.2.2.8)$$

where  $n'(j)$  is the sample of the additive noise at the output of the low-pass filter at time  $jT$ . The input to the decision slicer  $\hat{a}(j)$  is:

$$\hat{a}(j) = \sum_{k=0}^N c(k)r(j+N-k) - \sum_{l=1}^M d(l)\hat{a}(j-l) \quad (2.2.2.9)$$

where  $c(k)$  is the coefficient of the forward equalizer and  $d(l)$  is the coefficient of the DFE. The error  $e(j)$  is given by:

$$e(j) \triangleq \hat{a}(j) - a(j) \quad (2.2.2.10)$$

The tap values are chosen to minimize the mean-square error  $E[e^2(j)]$ :

Copyright © 2014, by the author(s).  
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

$$\begin{aligned}
E[e^2(j)] &= E[\{\hat{a}(j) - a(j)\}^2] \\
&= E[\{\sum_{k=0}^N c(k)r(j+N-k) \\
&\quad - \sum_{l=1}^M d(l)\hat{a}(j-l) - a(j)\}^2]
\end{aligned} \tag{2.2.2.2.11}$$

Therefore,  $c(n)$  and  $d(m)$  have to satisfy

$$\begin{aligned}
\frac{\partial E[e^2(j)]}{\partial c(n)} &= 0 \quad n = 0, 1, \dots, N \\
\frac{\partial E[e^2(j)]}{\partial d(m)} &= 0 \quad m = 1, 2, \dots, M
\end{aligned}$$

Thus

$$\begin{aligned}
\frac{\partial E[e^2(j)]}{\partial c(n)} &= 2E[e(j)r(j+N-n)] \\
&= 2E[\{\sum_{k=0}^N c(k)r(j+N-k) \\
&\quad - \sum_{l=1}^M d(l)\hat{a}(j-l) - a(j)\}r(j+N-n)] \\
&= 0, \quad n = 0, 1, \dots, N.
\end{aligned} \tag{2.2.2.2.12}$$

and

$$\begin{aligned}
\frac{\partial E[e^2(j)]}{\partial d(m)} &= -2E[e(j)\hat{a}(j-m)] \\
&= -2E[\{\sum_{k=0}^N c(k)r(j+N-k) \\
&\quad - \sum_{l=1}^M d(l)\hat{a}(j-l) - a(j)\}\hat{a}(j-m)] \\
&= 0, \quad m = 1, 2, \dots, M.
\end{aligned} \tag{2.2.2.2.13}$$

Substituting (2.2.2.2.8) into (2.2.2.2.12) and assuming that  $\hat{a}(j) = a(j)$  and  $E[a(j)a(j+k)] = 0$  if  $k \neq 0$ , we have

$$x(k) = \sum_{i=0}^N \{c(N-i) \sum_{l=0}^M x(l)x(l-i+k) + \phi_n(k-i)\}$$

$$-\sum_{i=1}^M d(i)x(i+k) \quad (2.2.2.2.14)$$

where  $\phi_n(k-i)$  is the autocorrelation function of the noise  $n'(t)$  at time  $(k-i)T$ .

Substitute (2.2.2.2.8) into (2.2.2.2.13) and we get

$$d(k) = \sum_{i=0}^N c(N-i)x(i+k) \quad (2.2.2.2.15)$$

This is the expression for the DFE coefficients. Now substituting (2.2.2.2.15) into (2.2.2.2.14), we have the expression for the coefficients of the forward equalizer:

$$\sum_{i=0}^N c(N-i)\left\{\sum_{l=0}^i x(l)x(l-i+k) + \phi_n(k-i)\right\} = x(k) \quad (2.2.2.2.16)$$

Let's now check the results to see if they agree with intuition, namely, that DFE should cancel all the post-cursor ISI and that the forward equalizer, under the assumption that noise  $n'(j)$  is zero, should completely eliminate the pre-cursor ISI. First, the response at the output of the forward equalizer  $s(k)$  is:

$$s(k) = \sum_{i=0}^N c(i)x(k-i) \quad 0 \leq k \leq N+M+1 \quad (2.2.2.2.17)$$

We expect that the last  $M$  points of the response should be cancelled exactly by DFE.

This is the case when we notice that

$$\begin{aligned} s(N+k) &= \sum_{i=0}^N c(i)x(N+k-i) \\ &= \sum_{i=0}^N c(N-i)x(i+k) \\ &= d(k) \quad 1 \leq k \leq M \end{aligned} \quad (2.2.2.2.18)$$

The next step is to see if the first  $N$  points are zero and the value of the  $(N+1)$ -th point is 1. Evaluating (2.2.2.16) at  $k=0$  and assuming the noise is zero:

$$\begin{aligned} x(0)\left[\sum_{i=0}^N c(i)x(N-i)\right] &= x(0)s(N) \\ &= x(0) \end{aligned} \quad (2.2.2.2.19)$$

Thus,

$$s(N) = 1 \quad (2.2.2.2.20)$$

Similarly, evaluating (2.2.2.2.16) at  $k = 1$ , we have

$$\begin{aligned} x(1) \sum_{i=0}^N c(i)x(N-i) + \sum_{i=0}^N c(i)x(N-1-i) &= x(1) \\ \implies x(1)s(N) + s(N-1) &= x(1) \end{aligned}$$

Therefore,

$$s(N-1) = 0$$

Similarly, by evaluating (2.2.2.2.16) at  $k = 2, 3, \dots, N$ , we can prove that

$$s(N-2) = s(N-3) = \dots = s(1) = s(0) = 0$$

Therefore, in the absence of noise, the forward equalizer can completely remove the pre-cursor ISI.

Although the combination of the forward equalizer and DFE can eliminate the ISI, it is important to notice that DFE is much simpler to implement than forward equalization. Therefore, it is desired that pulse response have as few pre-cursor ISI samples as possible so that the number of taps required in the forward equalizer can be reduced. By combining the pulse shaping and line coding, Jeremy Tzeng proposed a timing recovery technique which gives a sampling phase having minimum precursor ISI [3].

### 2.2.3. Echo Canceller

As mentioned before, due to the imperfect impedance matching of the hybrid transformer, there is a leakage of the near-end signal through the hybrid transformer to the near-end receiver. To cancel this unwanted echo, an echo canceller is used.

An echo canceller is an adaptive filter whose response is adjusted to match the echo path impulse response. Since both the echo path and the echo canceller are driven by the same near-end data signal, their outputs will be the same if the echo canceller has adapted to the same response as that of the echo path.

### 2.2.3.1. Linear Echo Canceller

Fig. 2.2.3.1.1 is the block diagram of the subscriber end. The received signal  $r_n$  is given by:

$$r_n = s_n + e_n + n_n \quad (2.2.3.1.1)$$

where  $e_n$  is the echo,  $n_n$ , the noise, and  $s_n$  is the desired far-end signal:

$$s_n = \sum_{k=0}^{M-1} h_k b_{n-k} \quad (2.2.3.1.2)$$

If the echo path can be modeled as a linear system with impulse response  $g_k$ ,  $0 \leq k \leq N-1$ , then the echo at the sampling instant  $n$  can be expressed as:

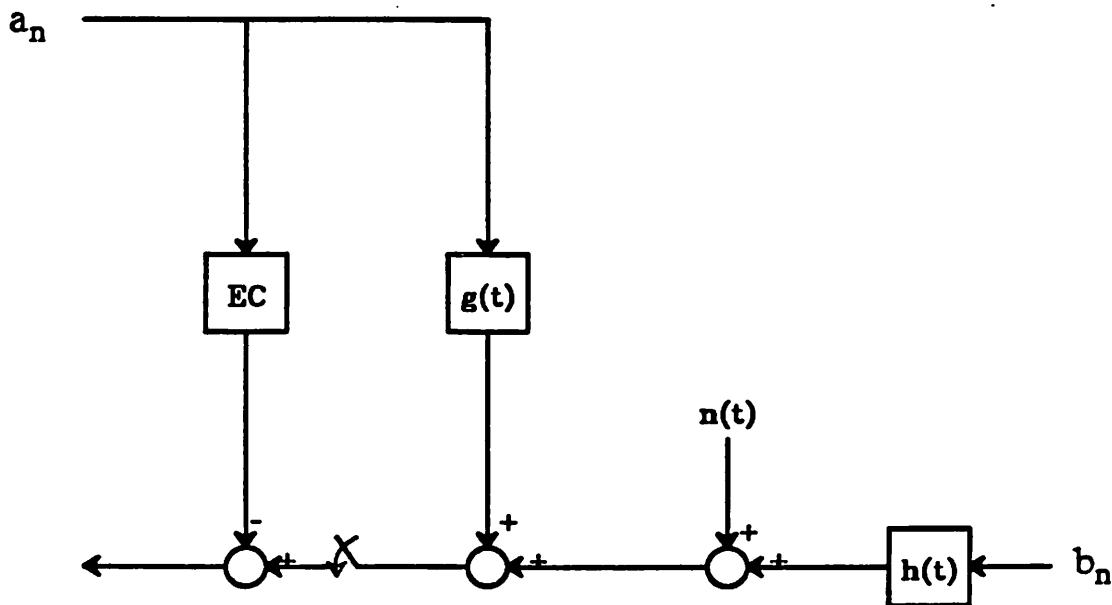


Figure 2.2.3.1.1. Block diagram of the subscriber end.

$$e_n = \sum_{k=0}^{N-1} g_k a_{n-k} \quad (2.2.3.1.3)$$

where  $a_k$  is the near-end data sequence. This echo can be cancelled by a linear echo canceller, an adaptive transversal filter. Let  $c_k$ ,  $0 \leq k \leq N-1$ , be the coefficients of the echo canceller, the output of the echo canceller at instant  $n$ ,  $\hat{e}_n$  is:

$$\hat{e}_n = \sum_{k=0}^{N-1} c_k a_{n-k} \quad (2.2.3.1.4)$$

The residual signal after the echo cancellation is given by:

$$y_n = s_n + n_n + (e_n - \hat{e}_n) \quad (2.2.3.1.5)$$

And the cancellation error is defined as:

$$e_n - \hat{e}_n \quad (2.2.3.1.6)$$

The coefficients of the echo canceller are adjusted to minimize the mean-square cancellation error. This is equivalent to minimizing the residual signal  $y_n$  since both  $r_n$  and  $n_n$  are independent of  $\{c_k\}$ . However, the presence of  $r_n$  and  $n_n$  in  $y_n$  do increase the variance of the noise of the adaptation and result in either increased final cancellation error or reduced adaptation speed.

One algorithm that achieves the minimization of the mean-square cancellation error using the gradient technique is the *minimum mean-square error gradient algorithm*. The coefficients are updated using the expression:

$$\bar{c}^{(n+1)} = \bar{c}^n + \beta E[y_n \bar{a}_n] \quad (2.2.3.1.7)$$

where  $\beta$  is the step size and

$$\bar{c} = (c_0, c_1, \dots, c_{N-1})^T \quad (2.2.3.1.8)$$

$$\bar{a}_n = (a_n, a_{n-1}, \dots, a_{n-N+1})^T \quad (2.2.3.1.9)$$

are coefficient vector and data vector respectively. However, the expectation in the last term of the right hand side of (2.2.3.1.7) is not available. If we simply ignore the expectation and replace it by its noisy estimate  $y_n \bar{a}_n$ , this algorithm becomes

$$\bar{c}^{(n+1)} = \bar{c}^n + \beta y_n \bar{a}_n \quad (2.2.3.1.10)$$

This is known as the *stochastic gradient algorithm*.

Although this algorithm has been widely studied and well developed, some important points are worth mentioning. Let the mean-square cancellation error be

$$\epsilon_n = E[(e_n - \hat{e}_n)^2] \quad (2.2.3.1.11)$$

Then, it can be described by the following difference equation:

$$\epsilon_{n+1} = (1 - 2\beta + \beta^2 N) \epsilon_n + \beta^2 N E[u_n^2] \quad (2.2.3.1.12)$$

where  $u_n$  is the uncancelable signal

$$u_n = s_n + n_n \quad (2.2.3.1.13)$$

and the solution of (2.2.3.1.12) is

$$\frac{\epsilon_n}{E[u_n^2]} = (1 - 2\beta + \beta^2 N)^n \left( \frac{\epsilon_0}{E[u_0^2]} - \frac{\beta N}{1 - \beta N} \right) + \frac{\beta N}{2 - \beta N} \quad (2.2.3.1.14)$$

From (2.2.3.1.14), we have the following conclusions:

(1) Convergence occurs only if

$$|1 - 2\beta + \beta^2 N| < 1 \quad \text{or} \quad 0 < \beta < \frac{2}{N} \quad (2.2.3.1.15)$$

Notice that as  $N$ , the length of the echo path impulse response or the number of the echo canceller taps, increases, the step size has to be reduced to guarantee the convergence.

(2) The ratio of the final cancellation error to the uncancelable signal is

$$\frac{\epsilon_\infty}{E[u_\infty^2]} = \frac{\beta N}{2 - \beta N} \quad (2.2.3.1.16)$$

Therefore, the cancellation error depends on the product of  $N$  and  $\beta$ . For a fixed  $N$ , the final cancellation error increases as the step size increases. Fig. 2.2.3.1.2 gives their relationship.

(3) The rate of convergence is governed by

$$(1 - 2\beta + \beta^2 N) \quad (2.2.3.1.17)$$

For  $\beta \ll \frac{1}{N}$ , rate of convergence increases as the step size increases. Fig. 2.2.3.1.3 shows the number of iterations required to achieve 20dB cancellation for

different step size assuming  $N = 16$ .

### 2.2.3.2. Nonlinear Echo Canceller

The major limitation of the linear echo canceller is that it can only deal with systems whose echo paths are linear. For a general nonlinear echo path, a nonlinear echo canceller has to be used. The nonlinear echo canceller that cancels echos of a general nonlinear echo path is the memory based (table look-up) echo canceller. The echo replica of such an echo canceller can be expressed as

$$\hat{e}_n = f(a_n, a_{n-1}, \dots, a_{n-N+1}) \quad (2.2.3.2.1)$$

$$\triangleq f_{\vec{a}_n}$$

where  $f(\cdot)$  is a general function. To implement such a general nonlinear echo canceller, a memory of size  $2^N$  words is used with each memory location storing the echo replica of a particular data pattern. This is shown in Fig. 2.2.3.2.1 where the current near-end transmitted data vector

$$\vec{a}_n = (a_n, a_{n-1}, \dots, a_{n-N+1})^T \quad (2.2.3.2.2)$$

is used as the address input to the memory and the output is the echo replica for that particular transmitted data pattern.

The adaptation occurs in such a way that only the location of the memory whose content is currently being used as the echo replica gets updated. In other words

$$f_{\vec{a}}^{(n+1)} = \begin{cases} f_{\vec{a}}^n + \beta y(n) & \vec{a} = \vec{a}^n \\ f_{\vec{a}_n} & \vec{a} \neq \vec{a}_n \end{cases} \quad (2.2.3.2.3)$$

The differences between such a general nonlinear echo canceller and a linear transversal-type canceller are:

- (1) The nonlinear echo canceller can cancel any nonlinear echo.

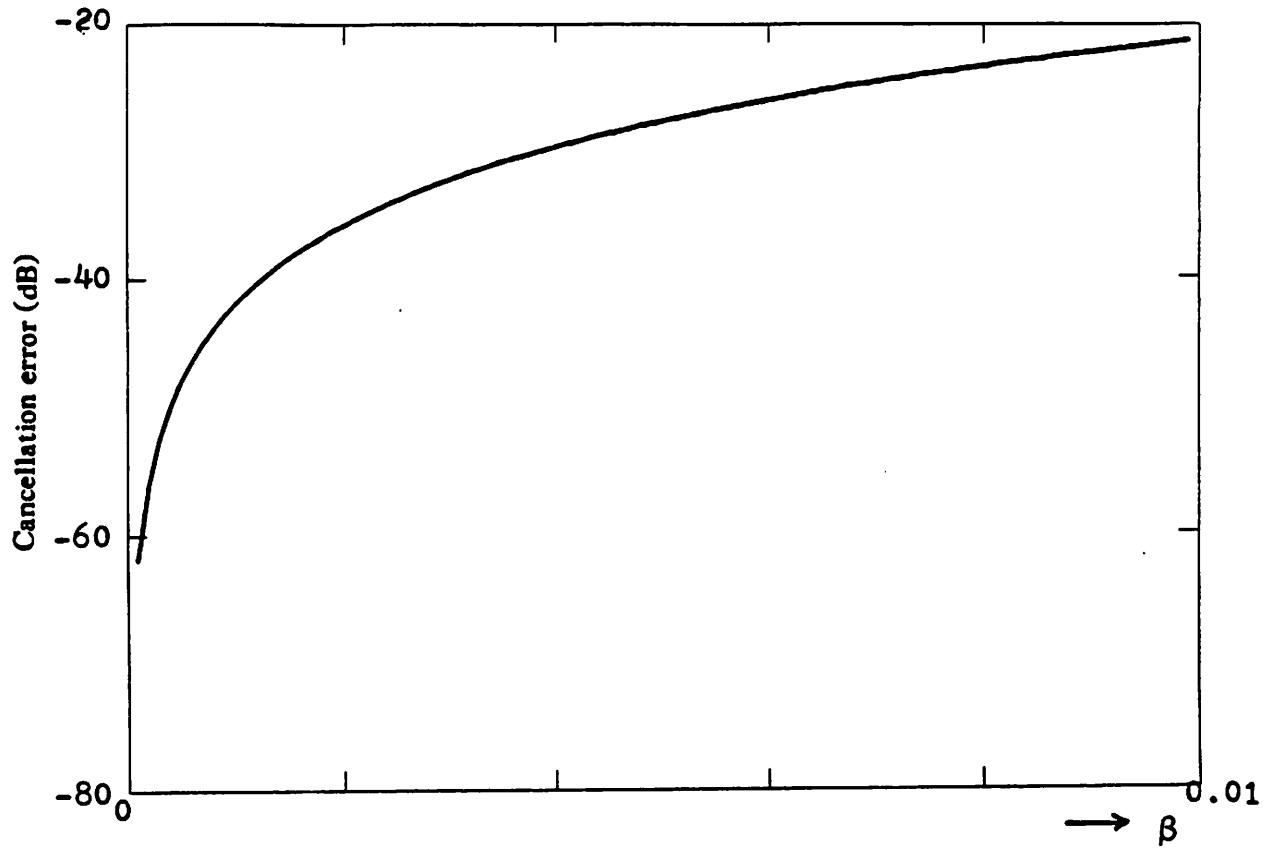


Figure 2.2.3.1.2. Cancellation error versus step size.

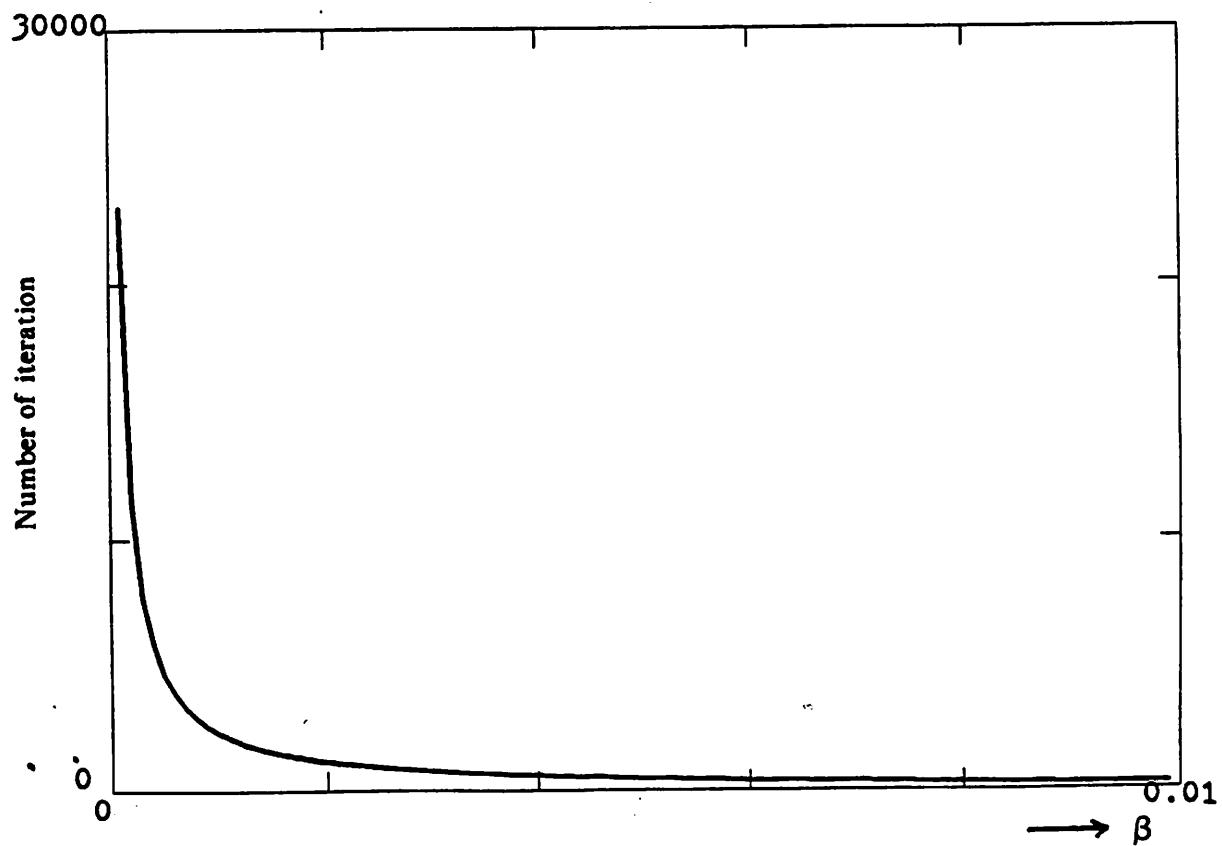


Figure 2.2.3.1.3. Number of iterations versus step size.

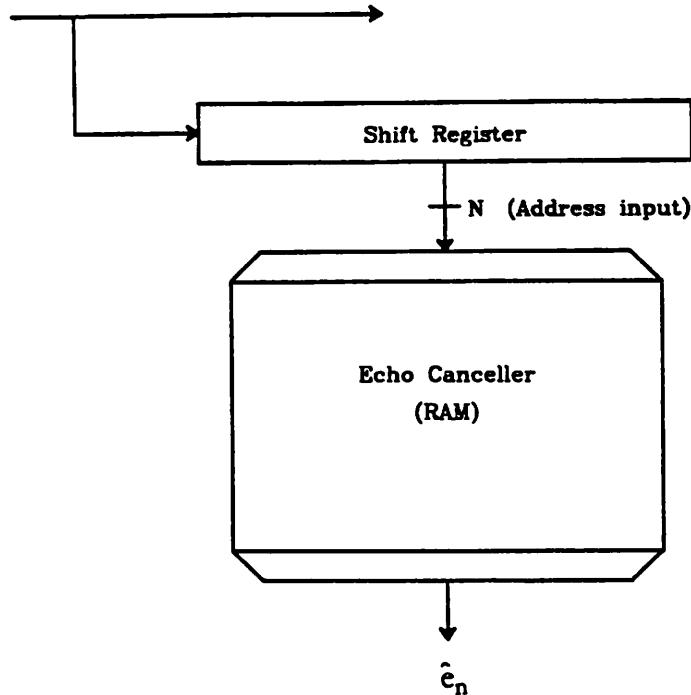


Figure 2.2.3.2.1. Memory-based echo canceller.

- (2) The nonlinear echo canceller does not require any calculation to get the echo replica as opposed to the linear echo canceller where a convolution has to be performed.
- (3) However, a nonlinear canceller requires much larger size of memory than a linear canceller. The size goes up exponentially as  $N$  increases. For example, for  $N = 16$ ,  $2^{16}$ , i.e., 64k words are needed instead of only 16 words in the linear echo canceller.
- (4) The rate of convergence also slows down exponentially as  $N$  increases.

One way to get most of the advantages of nonlinear cancellation without the disadvantages of such a large required memory and such a slow adaptation speed is to break up the input data vector into blocks and perform nonlinear echo cancellation on

each of the blocks. This is shown in Fig. 2.2.3.2.2. In such a case, we are assuming that the echo can be expressed as

$$\begin{aligned} e_n = & g_1(a_n, a_{n-1}, \dots, a_{n-p+1}) + g_2(a_{n-p}, \dots, a_{n-2p+1}) \\ & + \dots + g_m(a_{n-(q-1)}, \dots, a_{n-N+1}) \end{aligned} \quad (2.2.3.2.4)$$

where  $p$  is the number of data per block,  $m$  is the number of blocks and  $pm = N$ .

And the echo replica is

$$\begin{aligned} \hat{e}_n = & f_1(a_n, a_{n-1}, \dots, a_{n-p+1}) + f_2(a_{n-p}, \dots, a_{n-2p+1}) \\ & + \dots + f_m(a_{n-(q-1)}, \dots, a_{n-N+1}) \end{aligned} \quad (2.2.3.2.5)$$

where  $f_k(\cdot)$ ,  $1 \leq k \leq m$  is implemented using a memory of size  $2^p$  words.

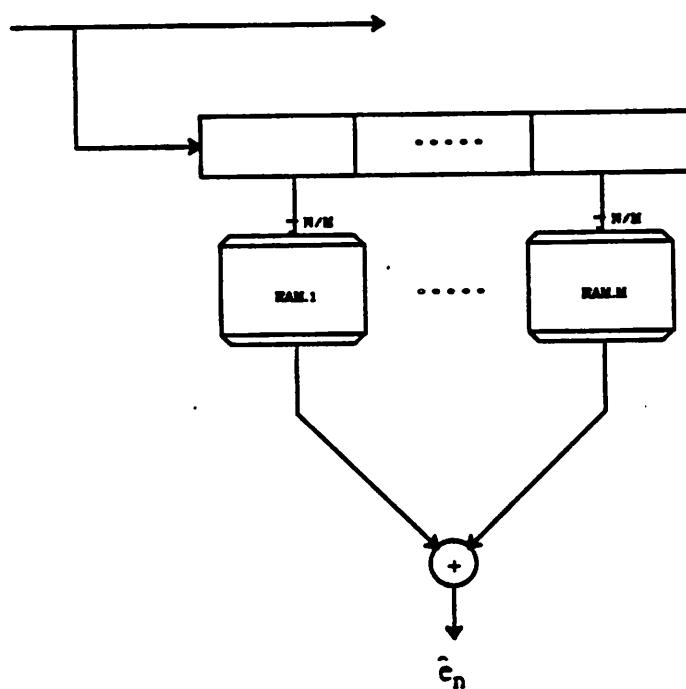


Figure 2.2.3.2.2.

The assumption embedded in (2.2.3.2.4) is that the nonlinearities occur within each block of input sequences and not between different blocks. This does not impose severe restriction on the generality of the nonlinearity in the echo path since the significant nonlinearities usually occur among the most recent bits, which are entirely contained within the first block. Thus, it is generally true that only the first block needs to be a full RAM and others can be linear; therefore, further reduction in memory size is possible. However, this might destroy the regularity of the echo canceller structure and complicate the cancellation operation. Notice that at one extreme,  $q = 1$ , this is just the single-memory echo canceller (2.2.3.2.1). At the other extreme where  $q = N$ , this becomes

$$\hat{e}_n = f_1(a_n) + f_2(a_{n-1}) + \cdots + f_N(a_{n-N+1})$$

and the memory size is  $N 2^1 = 2N$  words.

One consequence of the break-up is that the size of the required memory is reduced, from  $2^N$  words to  $m \cdot 2^p$  words. For example, if  $N = 12$ , and if we simply break the cancellation into two parts, i.e.,

$$\hat{e}_n = f_1(a_n, \dots, a_{n-7}) + f_2(a_{n-8}, \dots, a_{15})$$

The size of the memory reduces from 64k words for the one stage echo cancellation case to  $2 \cdot 2^8 = 512$  words for the two stage case.

In addition to the size reduction, the rate of convergence also improves exponentially as  $m$ , the number of blocks, increases. It has been shown that the mean-square cancellation error for a  $m$ -stage memory-based canceller with each sub-memory having size  $P \triangleq 2^p$  can be described by the following difference equation

$$\epsilon_{n+1} = (1 - \frac{1}{P}(2\beta - m\beta^2))\epsilon_n + \frac{m\beta^2 E[u_n^2]}{P} \quad (2.2.3.2.6)$$

where  $\epsilon_n$  is the mean-square cancellation error at time  $n$ ,  $\beta$ , the step size of adaptation, and  $u_n$  the uncancelable error. The solution of (2.2.3.2.6) is given by

$$\frac{\epsilon_n}{E[u_n^2]} = \left(1 - \frac{1}{P}(2\beta - m\beta^2)\right)^n \left(\frac{\epsilon_0}{E[u_0^2]} - \frac{m\beta}{2-m\beta}\right) + \frac{m\beta}{2-m\beta} \quad (2.2.3.2.7)$$

As in the linear echo canceller case, we can have the following observations:

- (1) Convergence occurs only if

$$\left|1 - \frac{1}{P}(2\beta - m\beta^2)\right| < 1, \text{ or, } 0 < \beta < \frac{2}{m} \quad (2.2.3.2.8)$$

- (2) The ratio of the residual cancellation error to the uncancelable signal is

$$\frac{\epsilon_\infty}{E[u_n^2]} = \frac{m\beta}{2-m\beta} \quad (2.2.3.2.9)$$

- (3) The rate of convergence is governed by the convergence rate of the sequence

$$\left(1 - \frac{1}{P}(2\beta - m\beta^2)\right)^k$$

Since  $P (= 2^P)$  is a large number, the rate of convergence is much slower than that of a linear echo canceller. The convergence improves as  $P$  decreases, and for a fixed  $N$ ,  $P$  decreases exponentially as  $m$  increases. Fig. 2.2.3.2.3. plots the number of iterations required to achieve 20 dB cancellation for different step size and different values of  $m$ .

### 2.2.3.3. Adaptation speed and residual error considerations

From Equations (2.2.3.1.16) and (2.2.3.1.17), we can see that the adaptation step size  $\beta$  can be used to trade off between the adaptation speed and the minimum residual cancellation error. The smaller the step size is, the smaller the final cancellation error will be. However, the adaptation speed decreases as the step size decreases. Therefore, to achieve smaller cancellation error, the rate of convergence will have to be sacrificed. One simple solution is to use a variable step size. At the beginning of the adaptation when the system is still in its transient state, a larger step size can be used to speed up the adaptation. When the system is approaching steady state, the step size can be reduced to achieve small residual cancellation error. The variable step size gives the

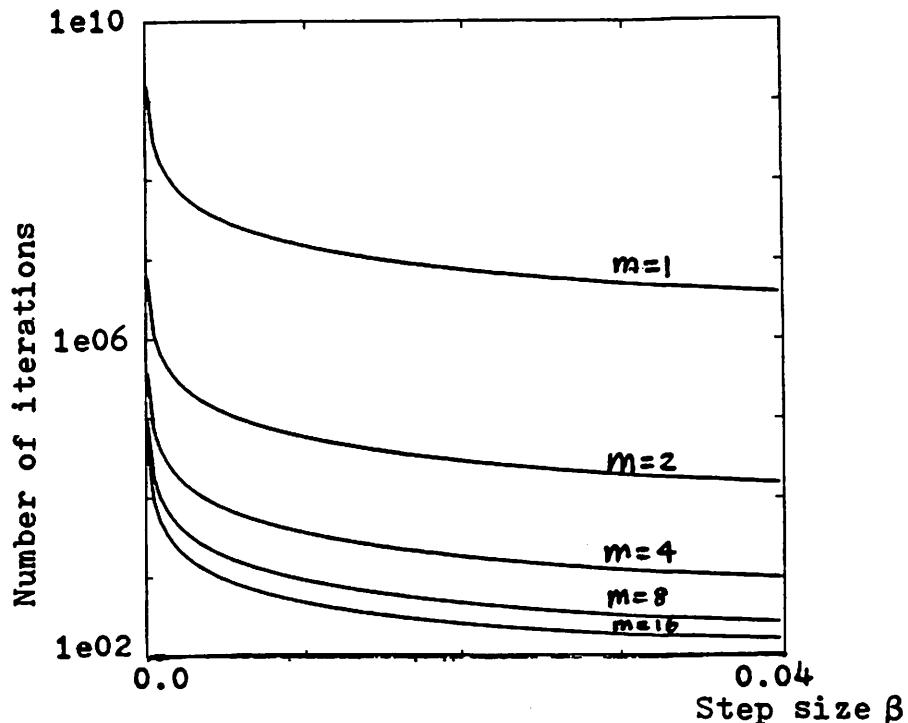


Figure 2.2.3.2.3. Number of iterations versus step size.

advantages of both the fast convergence associated with large step size and minimum cancellation error associated with small step size. The price paid is the increase in control complexity since either the residual error has to be monitored or some start-up strategies must be provided.

From Equation (2.2.3.1.14) we also see that the cancellation error ( $e_n - \hat{e}_n$ ) can be reduced if  $E[u_n^2]$ , the variance of the uncancelable error, is reduced. This fact is clear if we recall that we use the residual signal

$$y_n = (e_n - \hat{e}_n) + s_n + n_n$$

to adapt the echo canceller coefficients to minimize the cancellation error. Due to the presence of  $s_n$  and  $n_n$ , the coefficients will be updated even if the cancellation is perfect, i.e.,  $e_n = \hat{e}_n$ . Although statistically the coefficients are adjusted to minimize the

mean-square cancellation error, the presence of the uncancelable error does increase the variance of the adaptation. The uncancelable error is composed of several elements: the additive noise, the portion of the echo that is not covered by the echo canceller, and the far-end signal. The far-end signal is the largest component among these elements. We can reduce the uncancelable error by estimating the far-end signal and subtracting it from the residual signal used to update the echo canceller coefficients. This technique is called *adaptive reference echo cancellation* [13]. This is shown in Fig. 2.2.3.3.1.b, in which a decision feedback equalizer is used in conjunction with the echo canceller and the signal used to adapt the echo canceller coefficients is  $y_n$ :

$$y_n = (e_n - \hat{e}_n) + n_n + (s_n - \hat{s}_n)$$

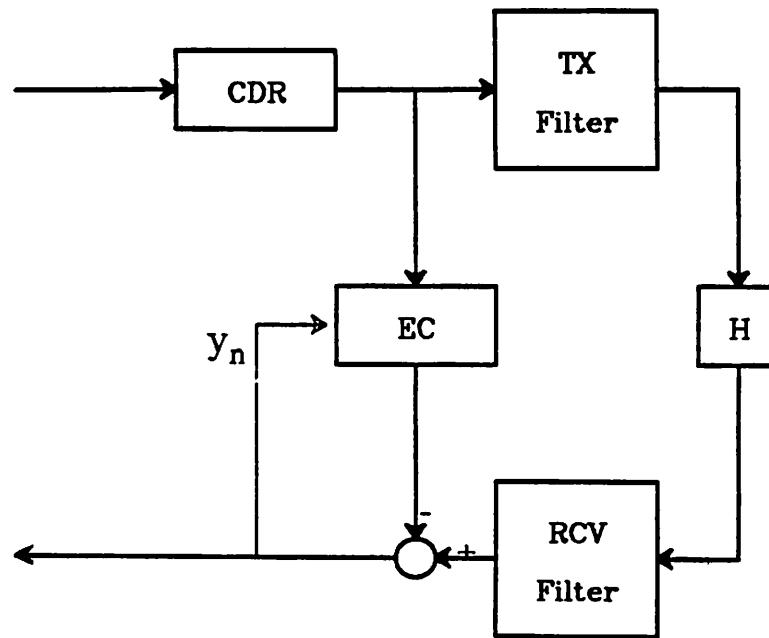
where

$$\hat{s}_n = \sum_{k=1}^M d_k \hat{b}_{n-k} + \delta_n$$

$d_k$  are the DFE coefficients, and  $\hat{b}_k$  are the estimated far-end data. The far-end signal is assumed to be in the form of

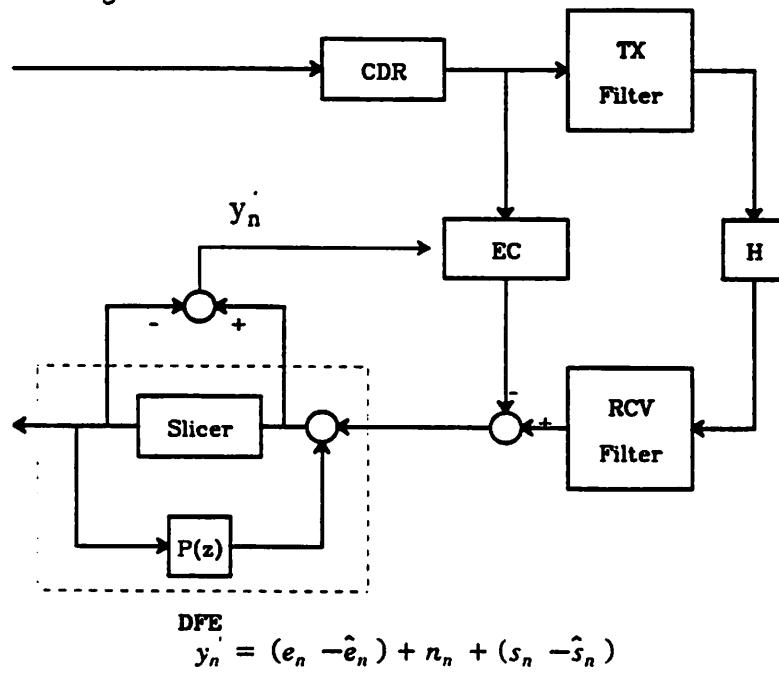
$$s_n = \sum_{k=-\infty}^{\infty} h_k b_{n-k}$$

and  $h_0 = 1$ . Notice that the same error signal is used to adjust the decision feedback equalizer coefficients. This works well if the far-end signal to echo noise ratio is high enough so that most decisions made about the far-end data are correct. However, before the echo canceller achieves a certain degree of cancellation, decision errors will be made since the signal to noise ratio is not large enough. Due to the error propagation, the DFE not only will be unable to function correctly and reduce the uncancelable signal power but will also contribute additional noise. This inhibits echo canceller convergence. Therefore, the DFE should be disabled and  $y_n$  be used to adapt the echo canceller coefficients at the beginning and the scheme of Fig. 2.2.3.3.1b should be used only if a certain degree of cancellation has been achieved.



$$y_n = (e_n - \hat{e}_n) + n_n + s_n$$

Figure 2.2.3.3.1.a. Conventional Echo cancellation.

**DFE**

$$y_n = (e_n - \hat{e}_n) + n_n + (s_n - \hat{s}_n)$$

$$s_n - \hat{s}_n = \left( \sum_{k=0}^M h_k b_{n-k} + b_n \right) - \left( \sum_{k=1}^M d_k \hat{b}_{n-k} + \hat{b}_n \right)$$

Figure 2.2.3.3.1.b Adaptive reference echo cancellation.

Another issue is the variation of the adaptation algorithms. The updating equation for the stochastic gradient algorithm is given as

$$\bar{c}^{n+1} = \bar{c}^n + \beta y_n \bar{a}_n$$

Equation (2.2.3.1.16) and (2.2.3.1.17) give the residual cancellation error and rate of convergence of this updating equation, respectively.

A modification to this algorithm is called sign adaptation where the updating equation is given as

$$\bar{c}^{n+1} = \bar{c}^n + \beta sgn(y_n) \bar{a}_n \quad (2.2.3.3.1)$$

and

$$sgn(y_n) = \begin{cases} 1 & y_n > 0 \\ -1 & y_n < 0 \end{cases}$$

This is easier to implement than the previous algorithm since only the sign of the error signal is needed in the adaptation. The dynamics of the convergence of the sign adaptation algorithm are different from the straightforward implementation of the stochastic gradient algorithm. Reference [14] gives a technique to derive the expression governing the dynamic of the convergence process. Using this technique, we have the difference equation of the cancellation error for the sign adaptation:

$$\epsilon_{n+1} = \left( 1 - \beta \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{\epsilon_n + \sigma_u^2}} \right)^2 \epsilon_n + N \beta^2 \quad (2.2.3.3.1)$$

where  $\beta$  is the step size,  $N$ , the number of echo canceller taps, and  $\sigma_u$ , the standard deviation of the uncancelable noise. Notice that in order to achieve the same cancellation error as in the linear stochastic gradient adaptation case, a smaller step size has to be used. The requirement in reducing the step size can be explained intuitively by noticing that even in the absence of the noise, the adaptation will never stop even when the cancellation is perfect. This is due to the sign adaptation, which always adds or subtracts the coefficients corresponding to those non-zero data taps with the magnitude of the step size  $\beta$ . To reduce the wandering of the coefficients, which limits the

minimum achievable cancellation error, the step size has to be kept small. Small step size corresponds not only to the slower convergence, which occurs only during the transient or the start-up of the system and therefore may not be of great concern, but also to an increase in the word length required for the register or memory to store the echo canceller coefficients. Notice that the word length has to be large enough to give the accuracy as well as the long time constant required. To overcome this drawback, a modified sign adaptation scheme can be used. As we recall that the reason for the coefficient wandering is due to the sign adaptation. If we use the adaptation

$$\bar{c}^{n+1} = \bar{c}^n + \beta g_{e_{th}}(y_n) \bar{a}_n \quad (2.2.3.3.2)$$

where

$$g_{e_{th}}(x) = \begin{cases} 1 & x \geq e_{th} \\ -1 & x \leq -e_{th} \\ 0 & |x| < e_{th} \end{cases} \quad (2.2.3.3.3)$$

the coefficients will not be updated once the residual signal  $y_n$  falls within some threshold  $e_{th}$ . By doing that, we can choose a larger step size. This does not mean, however, that the coefficients will cease moving toward the optimum point once the residual signal becomes smaller than the threshold. The adaptation of the coefficients to the optimum point continues due to the existence of the noise. Using a technique similar to the one presented in [14], we can get the following difference equation governing the cancellation error for this modified sign adaptation scheme:

$$\epsilon_{n+1} = \left[ 1 - \beta \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{\epsilon_n + \sigma_u^2}} e^{-\frac{e_{th}^2}{2(\epsilon_n + \sigma_u^2)}} \right]^2 \epsilon_n + N \beta^2 G(\epsilon_n + \sigma_u^2) \quad (2.2.3.3.4)$$

where

$$G(x^2) = \frac{1}{\sqrt{2\pi x^2}} \int_{-\infty}^{\infty} g^2(t) e^{-\frac{t^2}{2x^2}} dt$$

$$\begin{aligned}
 &= \sqrt{\frac{2}{\pi x^2}} \int_{e_{th}}^{\infty} e^{-\frac{t^2}{2x^2}} dt \\
 &= 2 \frac{1}{\sqrt{2\pi}} \int_{\frac{e_{th}}{x}}^{\infty} e^{-\frac{t^2}{2}} dt
 \end{aligned} \tag{2.2.3.3.5}$$

To achieve the same cancellation error, we have to choose an appropriate threshold  $e_{th}$  for a given step size  $\beta$ . This threshold is a function of the variance of the uncancelable noise. This modification allows us to use a larger step size for the same cancellation error at the sacrifice of the adaptation speed. For example, to achieve 60dB echo cancellation for -40dB uncancelable error (i.e., we assume the far-end signal has been attenuated by 40 dB and recall the the far-end signal is the dominant component of the uncancelable error), the step size needed for linear adaptation can be found from Equation (2.2.3.1.16):

$$\frac{\epsilon_\infty}{E[u_n^2]} = \frac{\beta N}{2 - \beta N}$$

We find that  $\beta = 1.25 * 10^{-3}$ . To achieve the same echo cancellation, the step size for a sign adaptation echo canceller can be found by substituting  $\epsilon_n$  and  $\epsilon_{n+1}$  by  $\epsilon_\infty$  in Equation (2.2.3.3.1):

$$\epsilon_\infty = (1 - \beta \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{\epsilon_\infty + \sigma_u^2}})^2 \epsilon_\infty + N \beta^2$$

and we have

$$\beta = \frac{\frac{1}{\sqrt{2\pi}} \frac{\epsilon_\infty}{\sqrt{\epsilon_\infty + \sigma_u^2}}}{N + \epsilon_\infty \frac{2}{\pi} \frac{1}{\epsilon_\infty + \sigma_u^2}} \tag{2.2.3.3.6}$$

The step size will be approximately equal to  $10^{-4}$ .

## CHAPTER 3

### Partial Response Coding

One of the requirements for ideal data transmission is to have zero intersymbol interference. Under this constraint, the overall signal spectrum must satisfy the Nyquist criterion, where for bandlimited signals, the spectrum must be vestigial symmetric about the Nyquist frequency, and the bandwidth be greater than or equal to the Nyquist frequency. This requirement greatly limits the signal spectrum design flexibility. On the other hand, if we allow a controlled amount of ISI in the transmission, we could gain an extra degree of freedom in the design of the overall signal spectrum. This is important because the shape of the signal spectrum has a profound impact on the overall system performance; furthermore, ISI can still be removed eventually. The idea of allowing a controlled amount of ISI is the basis of partial response coding.

Partial response coding (PRC), also called correlative level coding, combines successive bits together in a known fashion. It permits overall spectrum shaping as well as individual pulse shaping. Much information regarding PRC can be found in the literature[15, 16, 17, 18].

In section 3.1, we discuss the PRC in general. In section 3.2, we focus on a particular partial response code, namely, modified duobinary code, which we consider to be well suited to the DSL systems.

#### 3.1. Partial Response Coding

A data transmission system (Fig. 3.1.1) employing a partial response code can be modeled as in Fig. 3.1.2, where  $D$  represents one baud-period delay,  $F(D)$  is the partial response system polynomial, and  $G(\omega)$  is a filter satisfying the Nyquist criteria.

i.e.,

$$\frac{1}{T} \sum_{k=-\infty}^{\infty} G(\omega - \frac{2\pi k}{T}) = \text{constant}$$

For an ordinary zero-ISI transmission system, the system polynomial  $F(D) = \sum_{k=0}^{N-1} f_k D^k$  becomes unity:

$$f_k = \begin{cases} 1 & k = 0 \\ 0 & k \neq 0 \end{cases}$$

In general, the coefficients of the system polynomial  $f_k$  can be chosen to shape the signal spectrum. Let  $\{A_n\}$  be the input data sequence, then the sequence at the output of the system polynomial  $\{B_n\}$  is given by:

$$B_n = \sum_{k=0}^{N-1} f_k A_{n-k}$$

Thus,  $\{B_n\}$  is not random and has the autocorrelation function:

$$\begin{aligned} \phi(m) &= E(B_n B_{n+m}) \\ &= \sum_{k=0}^{N-1} \sum_{i=0}^{N-1} f_k f_i E(A_{n-k} A_{n+m-i}) \end{aligned}$$

If the sequence  $\{A_n\}$  is zero mean and white, i.e.,

$$E(A_{n-k} A_{n+m-i}) = \begin{cases} 1 & k = m - i \\ 0 & k \neq m - i \end{cases}$$

then,

$$\phi(m) = \begin{cases} \sum_{k=0}^{N-1-m} f_k f_{k+m} & m = -(N-1), \dots, 0, \dots, (N-1) \\ 0 & \text{elsewhere} \end{cases}$$

Taking the z-transform of  $\phi(m)$  and setting  $z = e^{j2\pi f T}$  we have:

$$\begin{aligned} \Phi(e^{j2\pi f T}) &= \sum_{m=-(N-1)}^{N-1} \phi(m) e^{-j2\pi f m T} \\ &= \left| \sum_{m=0}^{N-1} f_m e^{-j2\pi f m T} \right|^2 \end{aligned}$$

Thus, the power spectrum at the output of the filter  $G(f)$  is:

$$S(f) = \Phi(e^{j2\pi f T}) |G(f)|^2$$

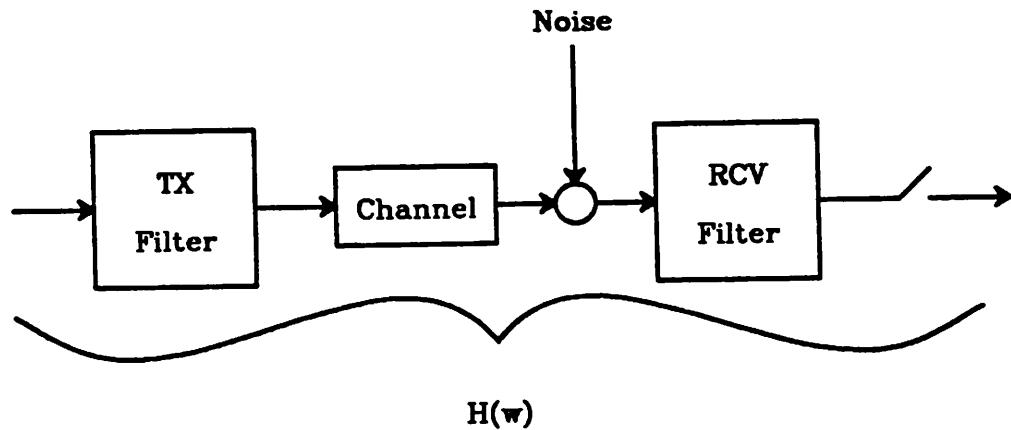
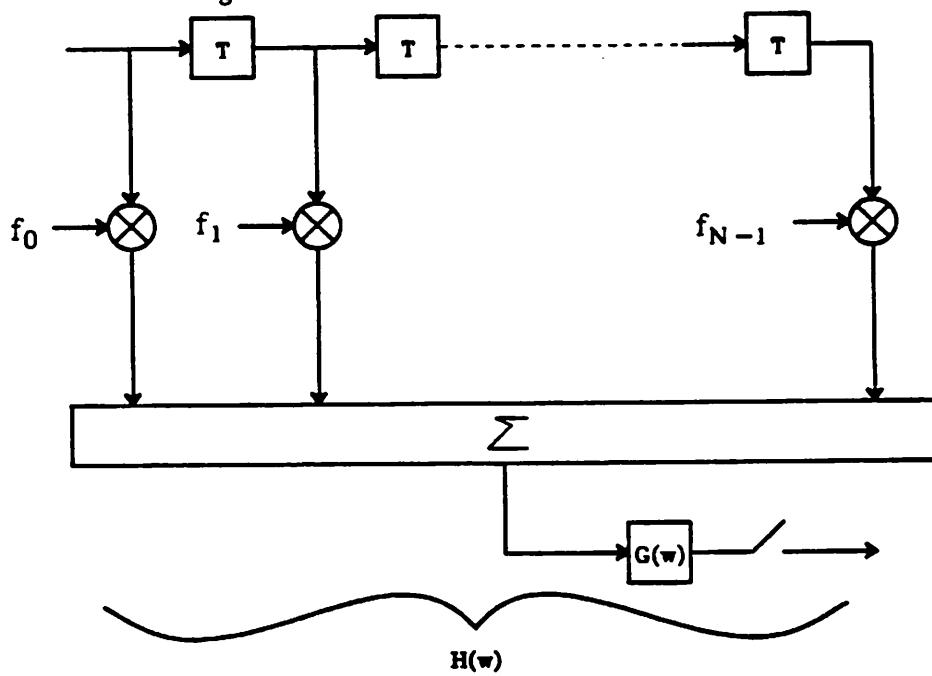


Figure 3.1.1. Data transmission system.



$G(\omega)$  satisfies Nyquist first criterion.  
Figure 3.1.2. Partial response system.

$$= \left| \sum_{m=0}^{N-1} f_m e^{-j2\pi fmT} \right|^2 |G(f)|^2$$

The same result can also be derived if we treat the system polynomial as a digital filter with coefficients  $f_k$ . The frequency response of the filter is:

$$F(e^{j2\pi\omega T}) = \sum_{k=0}^{N-1} f_k e^{-j2\pi\omega kT}$$

Thus, the power spectral density of the sequence  $\{B_n\}$  is given by:

$$\Phi_B(e^{j2\pi\omega T}) = \left| \sum_{k=0}^{N-1} f_k e^{-j2\pi\omega kT} \right|^2 \Phi_A(e^{j2\pi\omega T})$$

and Equation (3.1.1) can be obtained.

It is clear that by appropriately choosing  $f_k$ , the signal spectrum can be changed accordingly. Notice that because the system polynomial is used in the PRC system, the sampled impulse response has more than one nonzero component as opposed to the Nyquist system where only the main sample is allowed to be nonzero. Thus, there is ISI in the partial response system. However, the ISI is known (equal to  $f_k$ ,  $k \neq 0$ ) and therefore can be removed easily.

According to the superposition rule, Kretzmer classified most of the important partial response codes into five classes[16]. They are (Fig. 3.1.3):

(1). Equal Amplitude Superpositions - class I.

In this class,  $f_k = 1$ ,  $0 \leq k \leq N-1$ , i.e.,

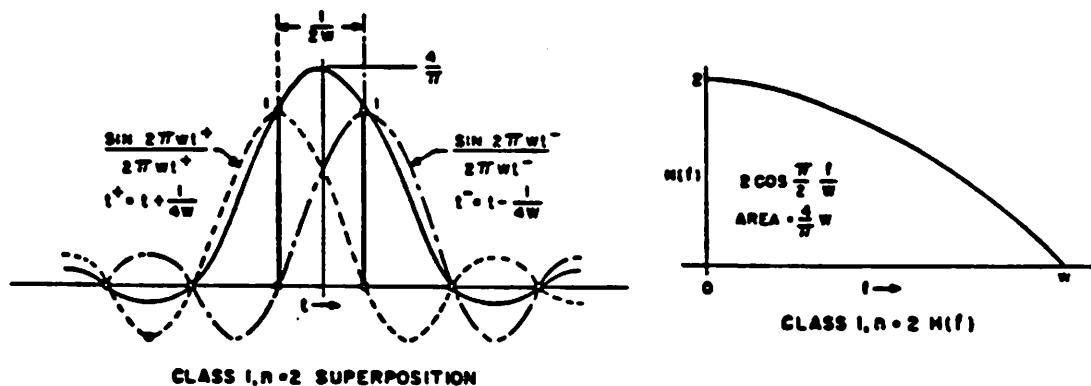
$$= \sum_{k=0}^{N-1} D^k$$

One example of this class is the *duobinary* code, where  $N = 2$  and the frequency response  $F(e^{j2\pi fT})$  is:

$$F(e^{j2\pi fT}) = 1 + e^{-2\pi fT}$$

# Partial Response Coding

## Class I : Equal Amplitude Superposition.



## Class II : Triangular Envelope Superposition.

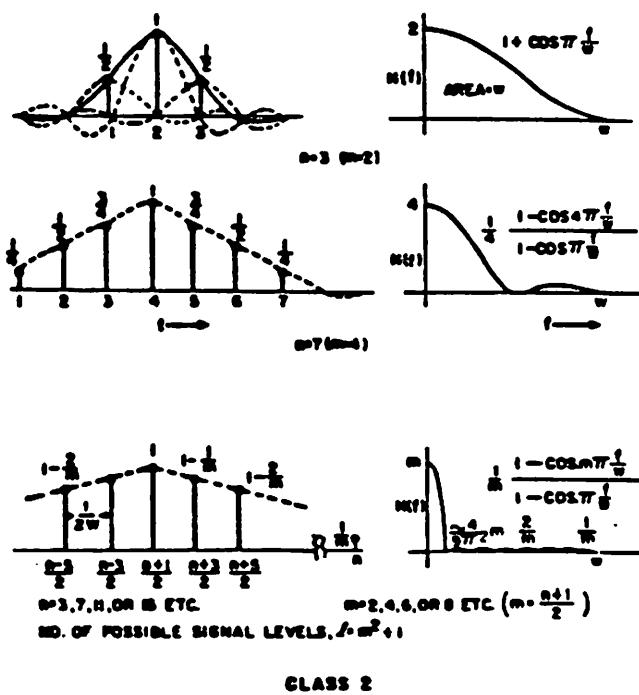
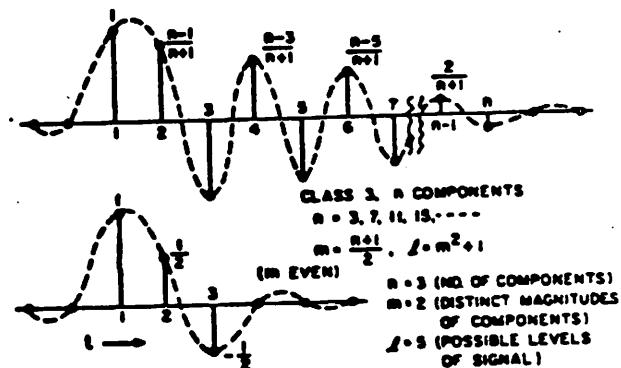
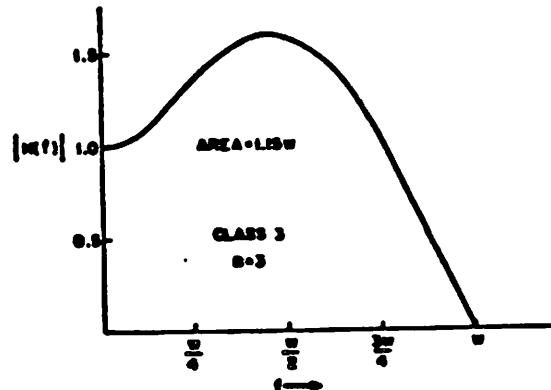


Figure 3.1.3. Partial Response Codes (From Reference [17])

## Class III : Superpositions Including Negative Components.



$$= 2 e^{-\pi f T} \cos(\pi f T)$$

The general expression of the frequency response for this class of PRC is:

$$\begin{aligned} F(e^{j2\pi f T}) &= \sum_{k=0}^{N-1} e^{j2\pi k f T} \\ &= \frac{1 - e^{j2\pi N f T}}{1 - e^{j2\pi f T}} \end{aligned}$$

Thus, the magnitude of the frequency response is:

$$|F(e^{j2\pi f T})| = \left| \frac{\sin(N \pi f T)}{\sin(\pi f T)} \right|$$

Notice that there is an important difference between the even- $N$  case and the odd- $N$  case. For an even  $N$ , the spectrum has a zero at the Nyquist frequency and the impulse response diminishes as  $\frac{1}{t^2}$ . The reason will be discussed later. Odd  $N$ , on the other hand, does not have a zero at the Nyquist frequency and the impulse response decays as  $\frac{1}{t}$ . Several important properties of this class of PRC are:

- (a). The power is concentrated at low frequencies.
  - (b). The number of possible levels of each sample is  $N + 1$ .
  - (c). In one sampling interval, the signal value can change by only one level. Any change in level greater than one implies that a transmission error has occurred.
- (2). Triangular Envelope Superposition - class II

This class of PRC is characterized by its pyramidal envelope of the system polynomial. For example, if  $N=3$ , the coefficients are  $f_0=\frac{1}{2}$ ,  $f_1=1$  and  $f_2=\frac{1}{2}$ . Thus, the polynomial becomes  $F(D) = \frac{1}{2} + D + \frac{1}{2}D^2$ . Its frequency response is given by:

$$F(e^{j2\pi f T}) = 2 e^{j2\pi f T} \cos^2(\pi f T)$$

This polynomial has the same magnitude response as that of a raised-cosine filter with 100% excess bandwidth. In general, the frequency response of the polynomial of this class is given by the expression:

$$F(e^{j2\pi fT}) = \left[ \sum_{k=1}^{m-1} \frac{m-k}{m} e^{j2\pi kfT} + 1 + \sum_{k=1}^{m-1} \frac{m-k}{m} e^{-j2\pi kfT} \right] e^{j2\pi(m-1)fT}$$

$$= [1 + 2 \sum_{k=1}^{m-1} \frac{m-k}{m} \cos(2\pi kfT)] e^{j2\pi(m-1)fT}$$

where  $m = \frac{(N+1)}{2}$  and  $N$  has to be an odd number. The magnitude of the response is:

$$\left| \frac{1 - \cos(2\pi f mT)}{1 - \cos(2\pi fT)} \right|$$

Several important properties of this class PRC are :

- (a). The bulk of the power is confined to  $f < \frac{1}{m} f_b$ .
  - (b). Both the spectrum and its first derivative are continuous. This means faster decaying in the time response ( $\frac{1}{t^3}$ ). Consequently, smaller ISI is expected. However, this advantage is offset by an increase in the number of output levels relative to class I PRC.
  - (c). The number of possible levels is  $m^2 + 1$ .
- (3). Superpositions Including Negative Components - class III

In this class, the polynomial coefficients ( $f_k$ ) are unequal, differing by integral multiples of the smallest component, and the magnitude of the coefficients falls off in a linear fashion.

For N=3,  $F(D) = 1 + \frac{1}{2}D - \frac{1}{2}D^2$ . Its frequency response is:

$$F(e^{j2\pi fT}) = 1 + \frac{1}{2}e^{-j2\pi fT} - \frac{1}{2}e^{-j4\pi fT}$$

Notice that unlike class II PRC, the first derivative of the spectrum is not continuous. Furthermore, its phase response is not linear.

(4). Superpositions Free of DC Transmission - class IV and V

These two classes are of quasi band-pass in the sense that their spectra reach zero at both DC and the Nyquist frequency. The spectrum shapings for class IV and V are symmetrical versions of those for class I and II. In class IV, all the nonzero  $f_k$ 's have the same magnitude, while in class V, the magnitude of the nonzero  $f_k$ 's decrease linearly as the distance from the center sample increases. One example of class IV is the modified duobinary code where

$$F(D) = 1 - D^2$$

and its frequency response is

$$F(e^{-j\omega T}) = 1 - e^{-2j\omega T}$$

The example for class V is

$$\begin{aligned} F(D) &= -\frac{1}{2} + D^2 - \frac{1}{2}D^4 \\ &= -\frac{1}{2}(1 - D^2)^2 \end{aligned}$$

In the following sections, we will discuss some important elements of the partial response system. These elements are: the system polynomial  $F(D)$ , the decoding, and the precoding procedures.

### 3.1.1. Spectrum shaping by the system polynomial $F(D)$

There is a direct relationship between the continuity of the signal spectrum and the decaying speed of the system impulse response. This is given in [17] and is quoted as the following:

If  $H(f)$  ( $= F(e^{j2\pi fT})G(f)$ ) and its first  $k-1$  derivatives are continuous and  $k$ -th derivative is discontinuous,  $|h(t)|$  decays asymptotically as  $\frac{1}{|t|^{k+1}}$ .

In order that the system has an open horizontal eye, the decaying of the impulse response must be faster than  $\frac{1}{t}$ . Thus,  $H(f)$  must be continuous. For a minimum

bandwidth system,  $G(f)$  is a brick-wall filter:

$$G(f) = \begin{cases} T & |f| \leq \frac{1}{T} \\ 0 & \text{elsewhere} \end{cases}$$

Thus,  $G(f)$  is discontinuous at Nyquist frequency. To give a continuous  $H(f)$ ,  $F(e^{j2\pi fT})$  must have a zero at the Nyquist frequency. This implies that  $F(D)$  should have  $(1 + D)$  as a factor to guarantee an open eye for a minimum bandwidth system. To generalize the above observation, we have:

**The first  $k - 1$  derivatives of  $H(w)$  of a minimum bandwidth system are continuous if and only if  $F(D)$  has  $(1 + D)^k$  as a factor.**

In addition to the continuity of the signal spectrum, the  $(1 + D)$  factor also contributes to the redistribution of the signal power to low frequencies. This is beneficial in reducing crosstalk interference as well as RFI, as discussed in chapter 2. The  $(1 + D)$  factor also gives a graceful roll-off near the band edge. This will ease the design of practical filters.

Although the  $(1 + D)$  factor is beneficial, a large multiplicity of  $(1 + D)$  factors may not be desirable since it results in the increase in the number of output levels, which degrades the error performance and complicates the hardware implementation.

Another requirement on the signal spectrum in the digital subscriber loop systems is a zero at DC. This is due to the fact that no DC transmission is possible through the hybrid transformer. This requirement can be fulfilled either by the pulse shaping technique or through the choice of the system polynomial  $F(D)$ . If the latter technique is used,  $F(D)$  must have  $(1 - D)$  as a factor.

Similar to the  $(1 + D)$  case, a large multiple of  $(1 - D)$  factors may not be desirable due to the increase in the number of output levels, although this factor will give gradual roll-off near DC and therefore ease the filter design.

Consequently, an appropriate choice of  $F(D)$  for DSL is to include both  $(1 + D)$  and  $(1 - D)$  factors. One simple choice is the modified duobinary code whose system polynomial is  $(1 - D^2)$ .

### 3.1.2. Decoding

In order to eliminate this controlled amount of ISI, a reverse process has to be taken. This process can be done at the receiver during the decoding. This is shown in Fig. 3.1.2.1. The structure is the same as a standard decision feedback equalizer. The received signal  $y_n$  is

$$y_n = f_0 A_n + \sum_{i=1}^{N-1} f_i A_{n-i} + n_n$$

where  $n_n$  is the noise at the sampling instant. The receiver estimates the transmitted data and use these to cancel the ISI. Thus, the signal at the input of the slicer is

$$\frac{1}{f_0} (y_n - \sum_{i=1}^{N-1} f_i \hat{A}_{n-i}) = \frac{1}{f_0} (f_0 A_n + \sum_{i=1}^{N-1} f_i (A_{n-i} - \hat{A}_{n-i}) + n_n)$$

If past decisions  $\hat{A}_k$  are correct, the ISI  $\sum_{i=1}^{N-1} f_i A_{n-i}$  can be completely eliminated and

the input to the slicer becomes  $(A_n + \frac{n_n}{f_0})$ .

However, there is one problem associated with the decoding process. Since the decision is fed into the feedback loop to generate the estimate of the ISI, a decision error will result in incorrect estimates of ISI; therefore, subsequent decisions will have increased error probability. This phenomenon is called *error propagation*.

### 3.1.3. Precoding

The problem of error propagation arises from that  $A_k$  are not available at the receiver and  $\hat{A}_k$ , the estimates of  $A_k$ , have to be used. One solution to this problem is to eliminate the ISI at the transmitter instead of subtracting the ISI at the receiver. This technique is called *precoding*. It eliminates the effect of previous symbols at the

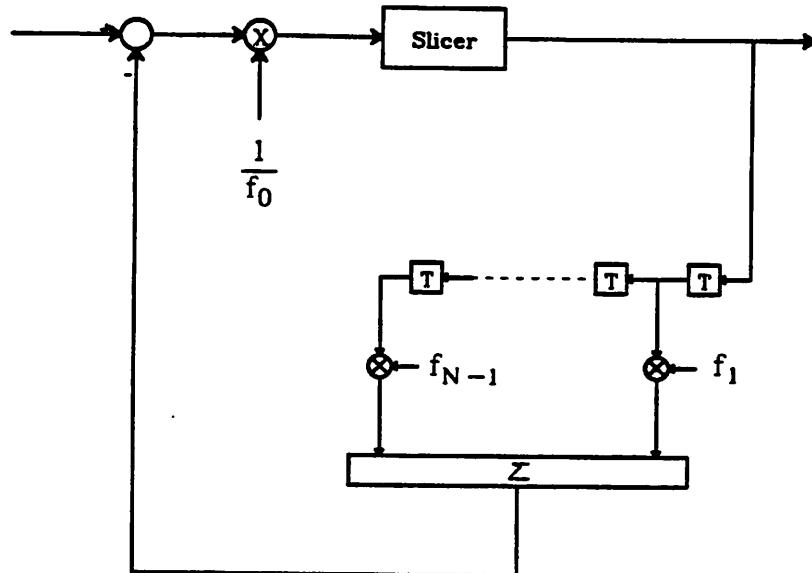


Figure 3.1.2.1. Partial Response Decoder.

transmitter where these symbols are known.

Let  $\{A_n\}$  be the sequence to be transmitted, then the sequence  $\{B_n\}$  at the output of the digital filter  $F(D)$  is:

$$B_n = \sum_{k=0}^{N-1} f_k A_{n-k}$$

If a reverse process is taken before feeding the data into the digital filter  $F(D)$  (Fig. 3.1.3.1), i.e., if we transmit a new sequence  $C_n$  such that:

$$C'_n = \frac{1}{f_0} [A_n - \sum_{k=1}^{N-1} f_k C_{n-k}]$$

and  $C_n = C'_n \bmod m$ , assuming integer-valued  $f_k$ ; then, the sequence  $\{D_n\}$  at the output of the digital filter will be:

$$D_n = \sum_{k=0}^{N-1} C_k f_{n-k}$$

and

$$\begin{aligned} D_n &= D_n \bmod m \\ &= A_n \end{aligned}$$

Therefore, in the absence of noise, the original data can be recovered by interpreting the received signal modulo  $m$ , which is a memoryless operation.

Fig. 3.1.3.2 shows the precoder and the coder of the duobinary signaling. An example is given in table 3.1.3.1, where  $\{A_n\}$  is the data sequence to be transmitted. Based on  $\{A_n\}$ , a precoded sequence  $\{C_n\}$  is generated:

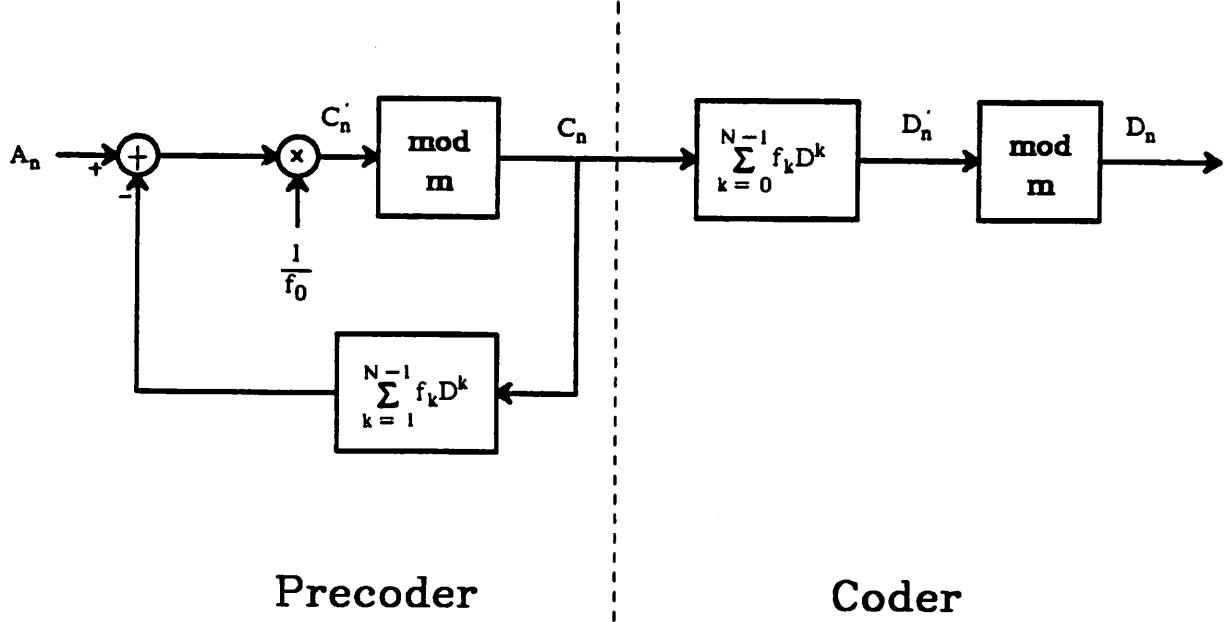


Figure 3.1.3.1. Precoder and Coder structure.

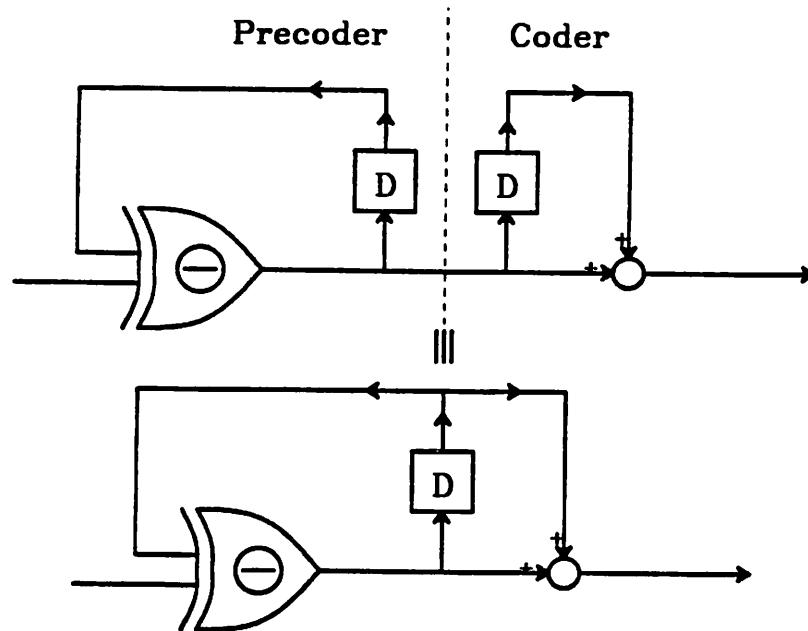


Figure 3.1.3.2. Duobinary Precoder and Coder.

$$C_n = A_n \ominus C_{n-1} \quad n = 1, 2, \dots$$

where  $\ominus$  denotes modulo-2 subtraction. Notice that modulo-2 subtraction is the same as modulo-2 addition. The coded sequence  $\{ D_n' \}$  is:

$$D_n' = C_n + C_{n-1}$$

And the decoded sequence  $\{ D_n \}$  is modulo 2 of  $\{ D_n' \}$ .

Duobinary (1 + D)												
Data $A_n$	1	0	1	1	1	0	1	0	0	0	1	1
Precoded $C_n$	1	1	0	1	0	0	1	1	1	1	0	1
Coded $D_n'$	1	2	1	1	1	0	1	2	2	2	1	1
Decoded $D_n$	1	0	1	1	1	0	1	0	0	0	1	1

Table 3.1.3.1 Duobinary signaling

One important fact is that the precoding process doesn't change the statistics of the data sequence. Therefore, if the input symbols to the precoder  $A_k$  are equally likely and statistically independent, the precoded sequence will also be equally likely and statistically independent.

With the precoding procedure, we can show that the precoded  $(1 - D)$  partial response code is exactly the same as AMI code. This is shown in table 3.1.3.2 where the coded sequence  $\{D_n\}$  can be interpreted as transmitting the 1 bits alternately as +1 and -1. The coder and precoder of AMI code is shown in Fig. 3.1.3.3.

### 3.2. Modified Duobinary PRC

In this section, we focus on the modified duobinary PRC. It is a three-level code with a system polynomial  $(1 - D^2)$ . The frequency response of the MDB coder is:

$$(1 - D^2)|_{D = e^{-j\omega T}} = 1 - e^{-2j\omega T} \\ = 2je^{-j\omega T} \sin(\omega T)$$

As discussed previously, the  $(1 + D)$  factor gives a continuous system spectrum, and the  $(1 - D)$  factor places a zero at DC. Due to the particular choice of the system polynomial, several advantages are realized for the MDB DSL systems. The first benefit is that the MDB system requires less excess bandwidth to achieve the same maximum horizontal eye opening than other ordinary transmission systems. In particular, a zero-excess-bandwidth modified duobinary system still has 36% horizontal eye

AMI ( $1 - D$ )												
Data $A_n$	1	0	1	1	1	0	1	0	0	0	1	1
Precoded $C_n$	1	1	0	1	0	0	1	1	1	1	0	1
Coded $D_n$	1	0	-1	1	-1	0	1	0	0	0	-1	1
Decoded $D_n$	1	0	1	1	1	0	1	0	0	0	1	1

Table 3.1.3.2 AMI signaling

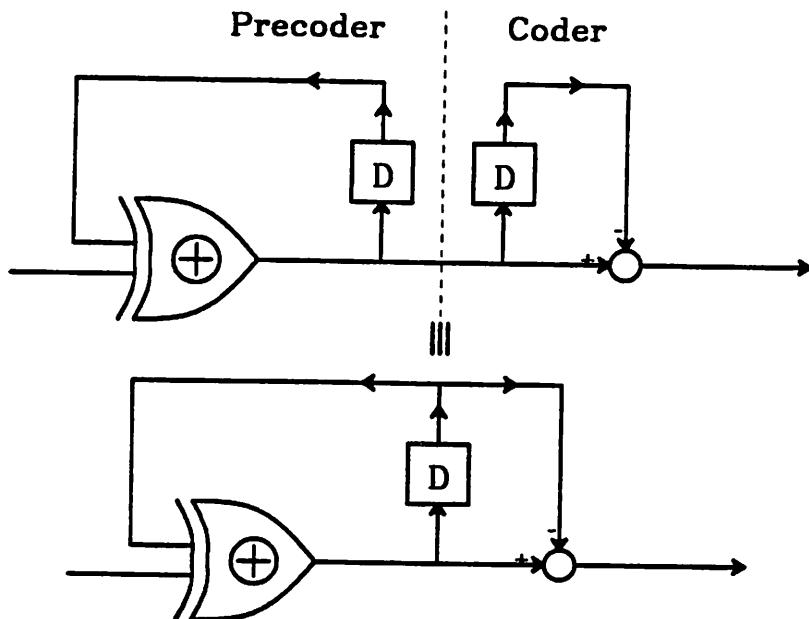


Figure 3.1.3.3. AMI Precoder and Coder.

opening. In contrast, in order to have same amount of horizontal eye opening, an AMI system will require 27% excess bandwidth. This wider eye opening, or equivalently, smaller ISI, of the MDB system can be explained from Fig.3.2.1, where a MDB pulse is considered as one zero-excess bandwidth, raised-cosine pulse superimposed with a same but negative pulse delayed by two baud periods. As can be seen in the figure, their tails tend to cancel with each other and therefore reduce the ISI. While in AMI (Fig. 3.2.2), the tails interfere constructively, and consequently, increase the ISI. Notice that besides MDB code, the duobinary code,  $(1 + D)$ , also behaves in such a way that the tails of two pulses superimpose destructively.

The open eye under zero excess bandwidth condition can also be predicted by knowing that the spectrum of a zero excess-bandwidth MDB system is continuous.

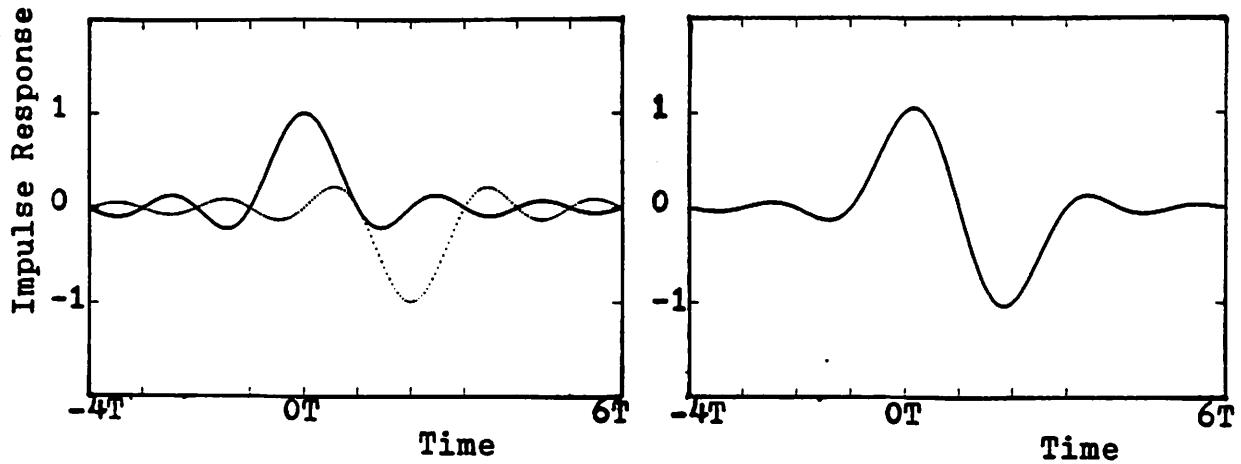


Figure 3.2.1. MDB impulse response as a superposition of two raised-cosine pulses

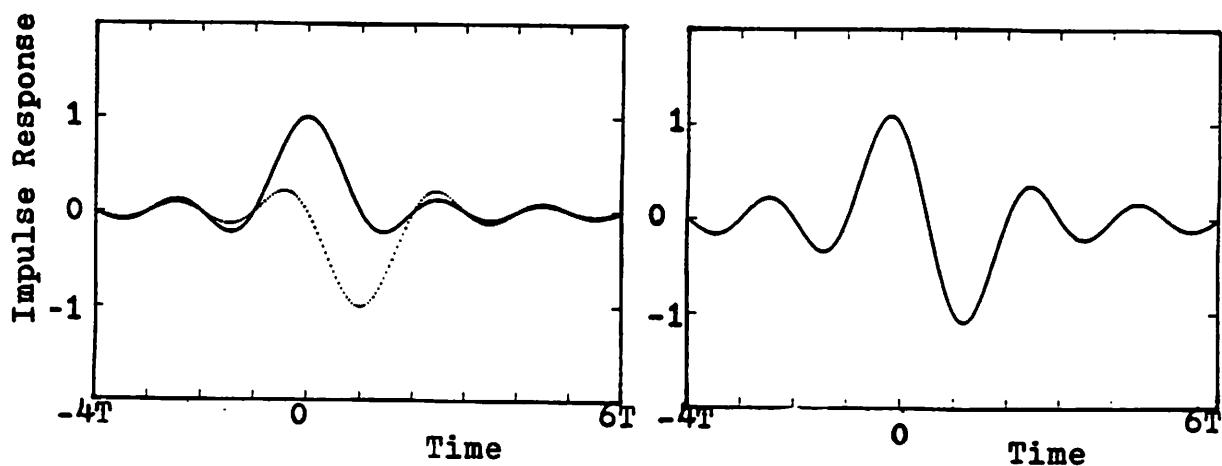


Figure 3.2.2. AMI impulse response as a superposition of two raised-cosine pulses

which ensures that the impulse response would die off asymptotically as  $\frac{1}{t^2}$ . To be more specific, we notice that the equalized raised-cosine pulse of the modified duobinary coding system with zero excess bandwidth is:

$$\begin{aligned} & \frac{T}{\pi} \frac{\sin(\frac{\pi t}{T})}{t} - \frac{T}{\pi} \frac{\sin(\frac{\pi t}{T} - 2\pi)}{t - 2T} \\ &= \frac{T}{\pi t(t - 2T)} \end{aligned} \quad (3.2.1)$$

while the pulse of the AMI system is

$$\frac{T}{\pi} \sin(\frac{\pi t}{T}) \frac{2t - T}{t(t - T)} \quad (3.2.2)$$

It is obvious that the pulse depicted by Equation (3.2.1) dies off asymptotically as  $\frac{1}{t^2}$

while Equation (3.2.2) gives a pulse that dies off as  $\frac{1}{t}$ . Fig. 3.2.3 shows the pulses of the MDB and AMI with various excess bandwidths.

A second important benefit of the MDB system polynomial is the redistribution of the signal power to low frequencies. This results in the improvement in NEXT and white noise signal to noise ratios. A detailed discussion of the system performance in NEXT and white noise environments will be given in Chapter 4.

The precoding and coding procedures of the MDB code is shown in Fig. 3.2.6. The precoded sequence  $\{C_n\}$  is given by:

$$C_n = A_n \oplus C_{n-2} \quad (3.2.3)$$

and the coded sequence  $\{D'_n\}$  is

$$D'_n = C_n - C_{n-2} \quad (3.2.4)$$

Table 3.2.1 is an example of the precoding, coding and decoding of a MDB sequence.

Let us now evaluate the probability of error of the symbol-by-symbol detection for a precoded MDB system over a white Gaussian noise channel. If the input data

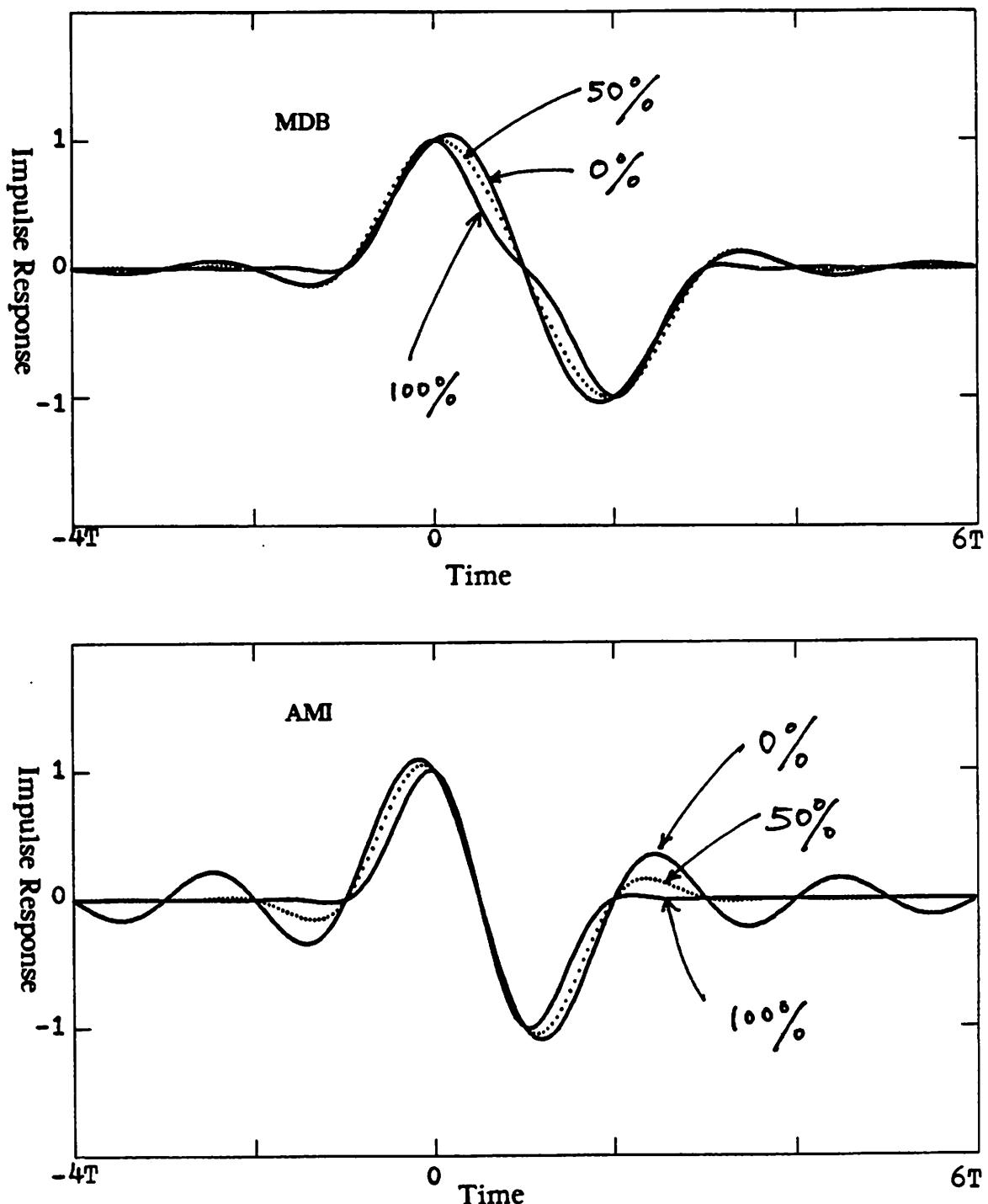


Figure 3.2.3. Impulse responses of MDB and AMI codes.

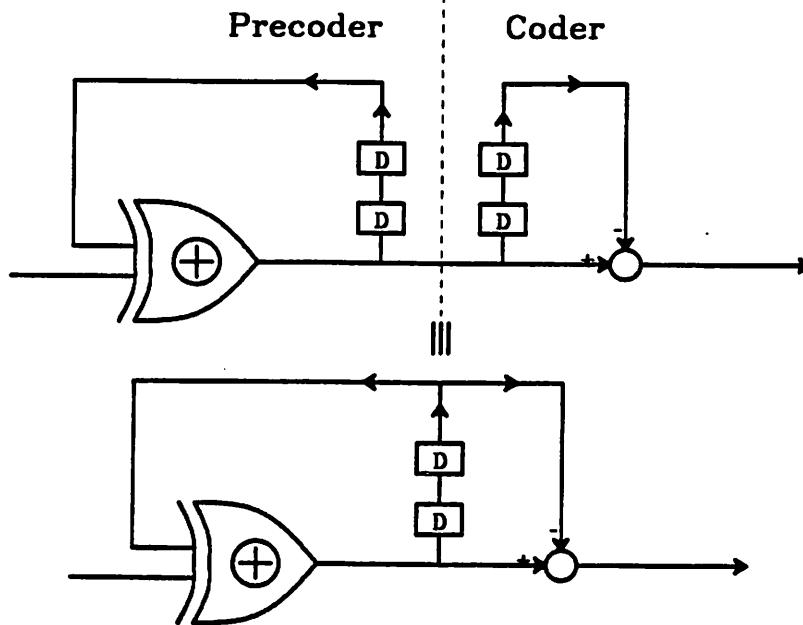


Figure 3.2.6. MDB Precoder and Coder.

Modified Duobinary ( $1 - D^2$ )												
Data $A_n$	1	0	1	1	1	0	1	0	0	0	1	1
Precoded $C_n$	1	0	0	1	1	1	0	1	0	1	1	0
Coded $D_n$	1	0	-1	1	1	0	-1	0	0	0	1	-1
Decoded $D_n$	1	0	1	1	1	0	1	0	0	0	1	1

Table 3.2.1 Modified Duobinary signaling

symbols are independent and the probabilities of the symbols being 1 and 0 are equal, then the probability distribution of the coded symbol sequence is:

$$Pr(D_n = l) = \begin{cases} \frac{1}{4} & l = 1 \\ \frac{1}{2} & l = 0 \\ \frac{1}{4} & l = -1 \end{cases} \quad (3.2.5)$$

Under the assumption that the additive noise is white Gaussian, the received signal  $y_n$ .

given  $D'_n = l$ , will be a Gaussian distributed random variable with a mean  $E(y_n) = 2ld$ , where  $2d$  is the distance between two adjacent received levels, and a variance  $\sigma^2$ .

$$\sigma^2 = \frac{N_0}{2} \int_{-\frac{1}{2T}}^{\frac{1}{2T}} |RCV(f)|^2 df \quad (3.2.6)$$

where  $RCV(f)$  is the frequency response of the receive filter, and  $\frac{N_0}{2}$  is the power spectral density of the additive white noise. Therefore, the probability of error  $P_e$  is:

$$\begin{aligned} P_e &= \sum_{\text{all } l} Pr[\text{error} \mid D'_n = l] Pr[D'_n = l] \\ &= Pr(y_n > -d \mid D'_n = -1) Pr(D'_n = -1) + Pr(y_n < d \mid D'_n = 1) Pr(D'_n = 1) \\ &\quad + Pr(|y_n| > d \mid D'_n = 0) \end{aligned} \quad (3.2.7)$$

However, since both "1" and "-1" are decoded as the symbol "1", the decision of "1" based on the received signal  $y_n$  given  $D'_n = -1$  is not considered as an error. Similarly, the decision of "-1" given  $D'_n = 1$  still gives a correct symbol detection. Thus, equation (3.2.7) becomes a bound for the probability of error. Using equation (3.2.5), we have:

$$\begin{aligned} P_e &\leqslant Pr(y_n > -d \mid D'_n = -1) Pr(D'_n = -1) + Pr(y_n < d \mid D'_n = 1) Pr(D'_n = 1) \\ &\quad + Pr(|y_n| > d \mid D'_n = 0) \\ &= \frac{3}{4} Pr(|y_n| > d \mid B_n = 0) \end{aligned} \quad (3.2.8)$$

Since

$$Pr(|y_n| > d \mid B_n = 0) = \frac{2}{\sqrt{2\pi}\sigma_d} \int_d^\infty e^{-\frac{x^2}{2\sigma_d^2}} dx \quad (3.2.9)$$

equation (3.2.8) becomes

$$P_e \leqslant \frac{3}{4} erfc(\sqrt{\frac{d^2}{2\sigma_d^2}}) \quad (3.2.10)$$

where

$$\operatorname{erfc}(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\frac{t^2}{2}} dt$$

Equation (3.2.10) gives the upper bound of the probability of error for a binary-signaling MDB system. For an M-level system, the probability of error is

$$P_e \leq (1 - \frac{1}{M^2}) \operatorname{erfc}(\sqrt{\frac{d^2}{2\sigma^2}}) \quad (3.2.11)$$

We can compare the probability of error of an MDB system with an ordinary system. In order to make the comparison, certain assumptions have to be made regarding the partitioning of the PRC coder between the transmitter and the receiver. In addition, the comparison is associated with a specific power constraint. Let's assume that the average power constraint is used and the PRC coder is equally partitioned between the transmitter and the receiver. Thus,

$$\begin{aligned} |TX(f)|^2 &= |RCV(f)|^2 \\ &= |H(f)| \end{aligned}$$

where  $H(f)$  is the Fourier transform of the PRC pulse. Under such assumptions, the variance  $\sigma^2$  is:

$$\begin{aligned} \sigma^2 &= \frac{N_0}{2} \int_{-\frac{1}{2T}}^{\frac{1}{2T}} |RCV(f)|^2 df \\ &= \frac{N_0}{2} \int_{-\frac{1}{2T}}^{\frac{1}{2T}} |2\sin(2\pi Tf)|^2 df \\ &= \frac{2N_0}{\pi} \end{aligned} \quad (3.2.12)$$

The average power  $P_{ave}$  is given by:

$$P_{ave} = \frac{l^2}{T} \int_{-\frac{1}{2T}}^{\frac{1}{2T}} |RCV|^2 df$$

$$= \frac{4l^2}{\pi T}$$

where  $l^2$  is:

$$l^2 = \begin{cases} \frac{1}{M} 2d^2 [1^2 + 3^2 + \dots + (M-1)^2], & \text{for even } M \\ \frac{1}{M} 2d^2 [2^2 + 4^2 + \dots + (M-1)^2], & \text{for odd } M \end{cases}$$

$$= \frac{d^2(M^2 - 1)}{2}$$

Thus,

$$P_{ave} = \frac{4d^2(M^2 - 1)}{3\pi T} \quad (3.2.13)$$

By substituting Equations (3.2.12) and (3.2.13) into (3.2.11), we have:

$$P_e \leq (1 - \frac{1}{M^2}) \operatorname{erfc} \left[ \sqrt{\frac{3}{M^2 - 1}} \left( \frac{\pi}{4} \right)^2 \frac{TP_{ave}}{N_0} \right] \quad (3.2.14)$$

The best achievable probability of error of an M-level system is[10]

$$P_e = (1 - \frac{1}{M}) \operatorname{erfc} \left[ \sqrt{\frac{3}{M-1}} \frac{TP_{ave}}{N_0} \right] \quad (3.2.15)$$

Comparing Equation (3.2.14) with (3.2.15), we see that if  $M$  is large, the MDB system will be  $\left(\frac{\pi}{4}\right)^2$ , or 2.1 dB worse than the optimum M-level system. Note that this result is only valid for this particular partitioning of the transmit and receive filters. A different set of partitioning and power constraint will result in different performance.

## CHAPTER 4

### Noise Performance of the Modified Duobinary PRC

A detailed study of the noise and crosstalk performance of the digital subscriber loop system employing modified duobinary coding is presented in this chapter. Several other line codes are also evaluated and compared by constraining the excess bandwidth of an equalized raised-cosine pulse to yield equal horizontal eye openings. In the study, performance of both the zero-forcing linear equalization and the decision feedback equalization schemes are analyzed.

This study also demonstrates the flexibility and compatibility of modified duobinary coding in terms of hardware implementation. We conclude that one form of modified duobinary coding gives a desirable tradeoff between hardware complexity and performance. This chapter is essentially based on [19] and [20].

#### 4.1. Comparison of Line Codes with zero-forcing LE Design

In this section, we compare the performance of various line codes using the zero-forcing linear equalization scheme. The line codes compared will be described in section 4.1.1. Section 4.1.2 summarizes the assumptions made for the comparisons. The comparative performances are given in section 4.1.3.

##### 4.1.1. Line codes

The line codes compared are: biphase, AMI (dicode partial response), modified duobinary (Class IV partial response), modified modified duobinary (MMDB), MMS43, and DI43.

**4.1.1.1. Biphas** Biphas is a binary antipodal signal, where a zero bit and one bit are transmitted as two-baud interval waveforms, one the negative of the other. One of the waveforms is a 50% duty cycle positive pulse followed immediately by a 50% duty cycle negative pulse. This line code can be considered to be binary phase shift keying (PSK) with modulation frequency equal to the baud rate. It has the advantage that compromise equalization is adequate, although undoubtably not in the presence of bridged taps.

#### 4.1.1.2. AMI

Alternate mark inversion is a ternary code in which a 50% duty cycle pulse is modulated by the levels -1, 0, and +1. Each ternary symbol transmits one bit, where zero bits are transmitted as zero pulses, and one bits are transmitted alternately as positive and negative pulses.

As discussed in chapter 3, AMI can be considered as a precoded partial response with polynomial

$$F(D) = 1 - D.$$

This is known as dicode partial response.

#### 4.1.1.3. Modified Duobinary (MDB)

As described in chapter 3, MDB is a class IV partial response with polynomial

$$F(D) = (1 - D)(1 + D) = 1 - D^2$$

The advantage of MDB over AMI is the additional  $(1 + D)$  factor, which places a zero in the spectrum at half the baud rate. This zero makes the spectrum continuous; thus zero excess bandwidth is practical. This results in the reduction of the impact of near-end crosstalk and the potential reduction of the radio frequency interference (RFI).

Note that the MDB coding can be interpreted as the interleaving of two lower-rate AMI streams. This also explains why MDB has a zero in the spectrum at half the baud rate, as pictured in Fig. 4.1.1.3.1. Since the spectrum of AMI is periodic in the baud rate (which it must be since it is a discrete-time signal), its zero in the spectrum at d.c. also results in a zero at the baud rate. If we apply AMI coding to two half rate streams, then each encoded stream must have a zero at half the baud rate by simple frequency scaling. Superimposing these two signals by interleaving them results similarly in a signal with a zero in the spectrum at half the baud rate.

An extension of the above statement leads to the idea of interleaving three lower rate AMI data streams. This new coding scheme has a polynomial

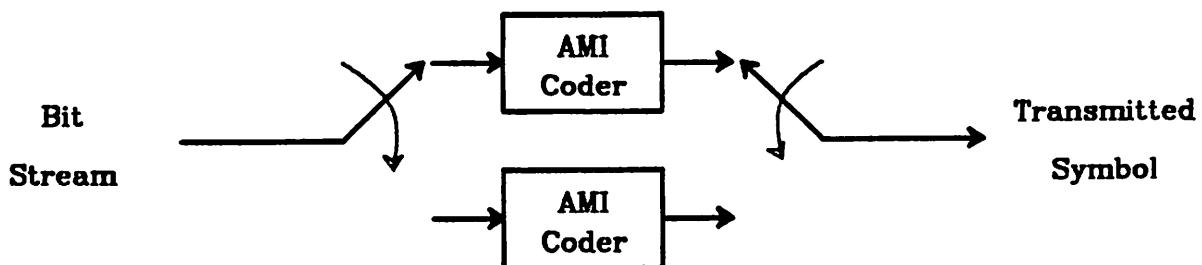


Figure 4.1.1.3.1. MDB as interleaved AMI.

$$F(D) = 1 - D^3$$

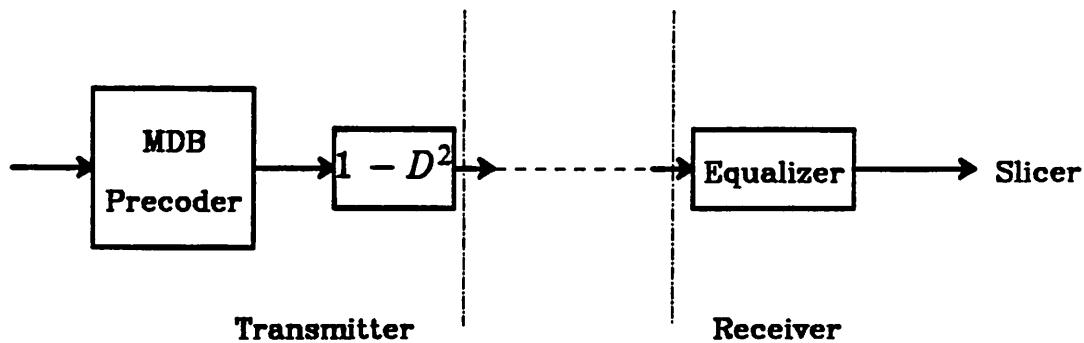
and has three zeros in the spectrum located at DC,  $\frac{1}{3f_b}$ , and  $\frac{2}{3f_b}$  respectively. However, this new coding scheme does not have any obvious advantage over the MDB scheme since the signal spectrum is discontinuous under zero excess bandwidth situation and therefore zero excess bandwidth is not practical.

The implication of Fig. 4.1.1.3.1 is that MDB requires complexity comparable to AMI implementation in the transmitter, but as we will see offers considerable benefit in performance in a NEXT-dominated or white noise dominated environment.

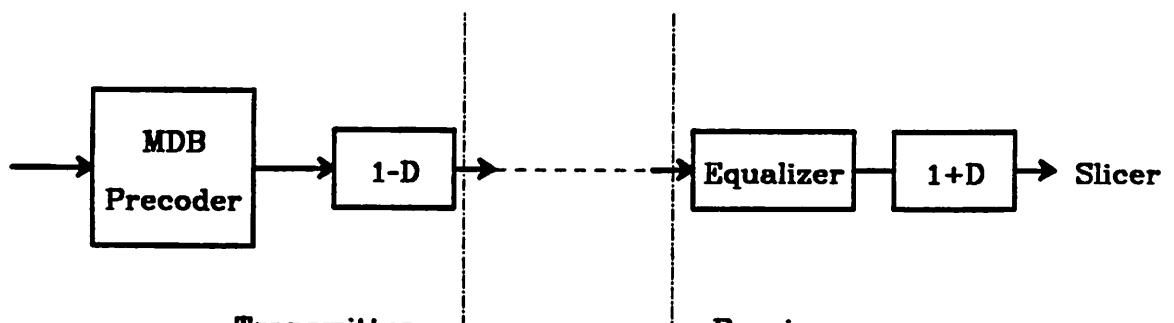
In a NEXT-dominated environment the partitioning of the filtering between the transmitter and receiver has no impact on the signal-to-interference ratio. However, this partitioning will have an impact on the crosstalk into foreign systems and the immunity to other types of interference such as white noise. It also has an impact on the implementation difficulty for the transmit and receive filters.

Fig. 4.1.1.3.2 displays two possible partitionings between transmit and receive filtering for MDB. In Fig. 4.1.1.3.2.a, the factor  $(1 - D^2)$  has been included in the transmitter, which is the assumption made in the comparisons that have been made for foreign crosstalk and white noise. Fig. 4.1.1.3.2.b shows an alternative, in which the factor  $(1 + D)$  has been moved to the receiver. This alternative implies that the receive filtering has a null at half the baud rate, which will simplify the filtering required to achieve low excess bandwidth. It also has superior performance in white noise, since the null at half the baud rate implies a lower gain in the receiver equalization at high frequencies.

In the following we have included both partitioning schemes. The partitioning of Fig. 4.1.1.3.2.a is called MDB1 and the partitioning of Fig. 4.1.1.3.2.b is called MDB2.



(a) MDB1



(b) MDB2

Figure 4.1.1.3.2. Two possibilities for partitioning  
of filtering between transmitter and receiver.

#### 4.1.1.4. Modified Modified Duobinary (MMDB)

Since the factor  $(1 + D)$  is beneficial in modified duobinary, we included a more complicated partial response with an additional zero at half the baud rate, resulting in a polynomial

$$F(D) = (1 - D)(1 + D)^2 = 1 + D - D^2 - D^3$$

This will be more complicated to implement, since the transmitted signal now has five levels rather than three, but will offer an additional improvement in signal-to-interference ratio in a near-end crosstalk environment due to the additional suppression of high frequency components near half the baud rate. On the other hand, it is more susceptible to crosstalk interference at a given signal-to-interference ratio because of the closer spacing of levels. This case will therefore serve to illustrate the tradeoffs between implementation complexity and performance within the class of partial response coders.

In the following we call this code modified-modified duobinary, or MMDB.

**4.1.1.5. MMS43** MMS43 is a particular 4B3T code, in which four binary bits are mapped into three ternary transmitted symbols[21]. This code makes more efficient use of the ternary symbols than AMI, and is able as a result to reduce the baud rate by a factor of 0.75 relative to AMI (ignoring frame format issues). This reduction in baud rate is beneficial in reducing the effects of near-end crosstalk.

**4.1.1.6. DI43** DI43 is a line code derived from a particular 4B3T by the method of Fig. 4.1.1.3.3 [22]. Two lower rate streams are encoded using 4B3T, and the resulting symbols are interleaved. This code therefore has a similar relationship to MMS43 as MDB has to AMI. This additional zero is beneficial in reducing the power spectral energy in the vicinity of the half baud rate. However, unlike in MDB and MMDB, this half-baud zero does not reduce the excess bandwidth required to maintain an adequate

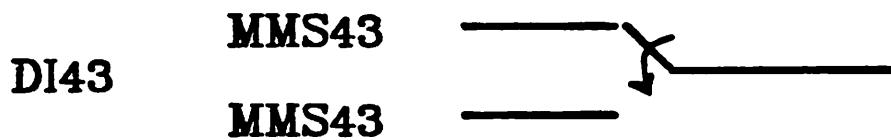


Figure 4.1.1.3.3.

---

horizontal eye opening. The reason for this is somewhat subtle, but relates to the fact that the zero is introduced strictly by the coding rather than by pulse shaping.

#### 4.1.2. Assumptions

Several assumptions have to be made in the relative comparisons. These assumptions are summarized in this section.

A block diagram of the modeled system is shown in Fig. 4.1.2.1. The transmitter takes a stream of bits to be transmitted, and consists of a scrambler, a coder which maps these bits into transmitted data symbols (typically three level), a pulse generator which associates with each transmitted data symbol a pulse with an appropriate amplitude, and a transmit filter. The receiver consists of an equalizer which compen-

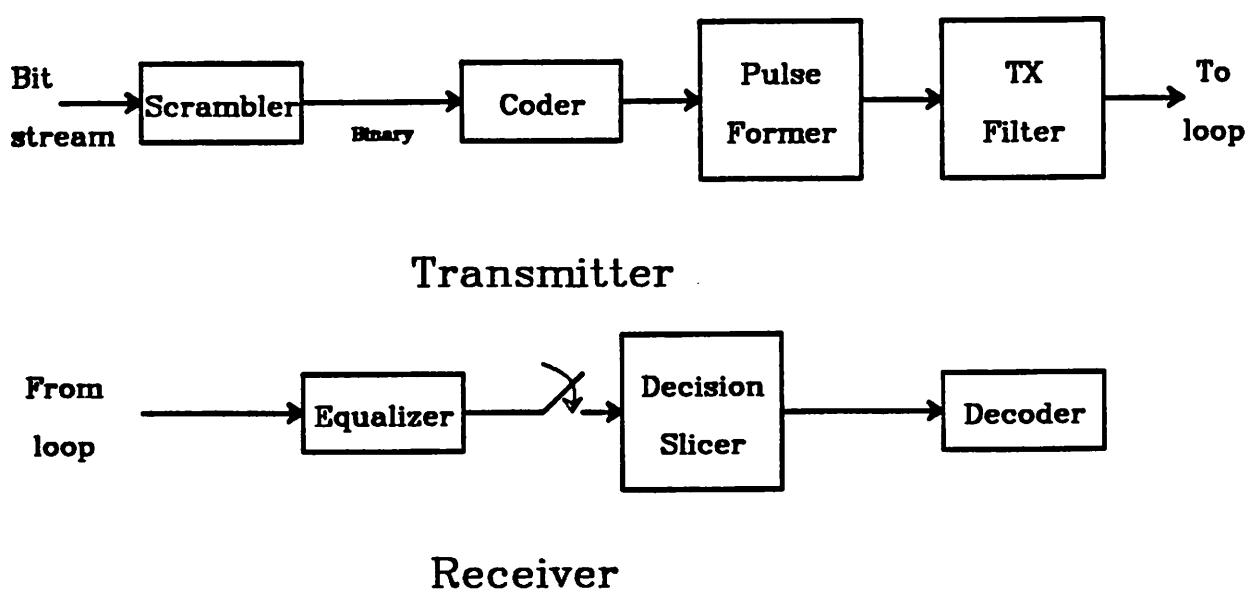


Figure 4.1.2.1. Block diagrams

sates for the channel distortion, a decision slicer, and a decoder.

#### 4.1.2.1. Transmitted Data Stream

The transmitted data bits include the customer 2B+D data, as well as any framing bits and auxiliary channels. The resulting bit stream is assumed to consist of independent equally likely binary symbols. This is a reasonable model as long as scrambling is included to break up likely deterministic sequences such as all marks or spaces.

#### 4.1.2.2. Transmitted Pulse

The pulse generated in the pulse generator is in all cases except BP a 50% duty cycle rectangular pulse. For BP it consists of two such pulses, one positive and the

other negative, spaced by one-half baud interval.

#### 4.1.2.3. Baud Rate

The baud rate is assumed to be 122 kHz for MMS43 and DI43, and 152 kHz for the remaining line codes.

#### 4.1.2.4. Line

In the following comparisons, an 18 kft 24 AWG line is assumed, and no bridged taps are included. Since bridged tap distortion is usually compensated by decision-feedback equalization (DFE), the presence of bridged taps will to first order have little impact on the relative noise immunity of the different line codes.

#### 4.1.2.5. Equalized Pulse

We have considered an idealized receiver model consisting of an equalizer which equalizes an isolated transmitted pulse to an ideal raised cosine response. The resulting performance can be approximated closely by practical receiver adaptive equalization techniques.

The excess bandwidth of the raised cosine response is a critical choice in the comparisons. The excess bandwidth affects the bandwidth of the equalized signal, and hence the amount of crosstalk and noise which gets through to the decision slicer. In addition, it affects the horizontal eye opening, and hence the sensitivity of the error probability to timing jitter, and the ease with which timing can be recovered from the received signal. We have adjusted the excess bandwidth to maintain approximately the same maximum horizontal eye opening for all cases. This implies that the sensitivity of the error probability to timing jitter will be approximately the same in all cases. As an anchor for this comparison, we have assumed zero excess bandwidth for the partial response cases. This results in a very reasonable maximum horizontal eye opening of 36% of T, or in other words the peak jitter which closes the eye has

magnitude equal to 18% of T. The resulting excess bandwidths and maximum horizontal eye openings for the different cases are shown in the table below:

Line Code	EXCESS BANDWIDTH	TOTAL BANDWIDTH (kHz)
BP1 and BP2	160%	198
AMI	27%	97
MDB1 and MDB2	0%	76
MMDB	0%	76
MMS43	33%	81
DI43	33%	81

Table 4.1.2.5.1

MDB and MMDB both have open eyes at zero excess bandwidth. This is a consequence of fact that the zero in the spectrum at half the baud rate is introduced by pulse shaping rather than by coding. The width of the eye at zero excess bandwidth has been computed in [17] and is about 36% of T for both. We estimated for the other signals, by direct computation of the eye diagram, the excess bandwidth required to get an eye opening of 36%.

The block coding (MMS43 and DI43) eye widths are not significantly different from those of ordinary three level PAM. This is perhaps a little surprising for DI43, which like MDB and MMDB has a zero at half the baud, but is due to the fact that adjacent ternary symbols account for the bulk of the eye closure, and are independent of one another. This is also true of MDB and MMDB, but the close relationship of symbols two baud intervals apart largely cancels the tails of the pulse and opens the eye, even for zero excess bandwidth. The eye diagrams of AMI, MDB, MMS43, and DI43 with excess bandwidth given in table 4.1.2.5.1 are shown in Fig. 4.1.2.5.1.

The excess bandwidth for all methods is given with respect to the minimum Nyquist bandwidth required for transmission at the appropriate baud rate, namely half the baud rate. The receiver equalization we assume for BP does not explicitly include a demodulator, so it's bandwidth is quite high compared to the other methods.

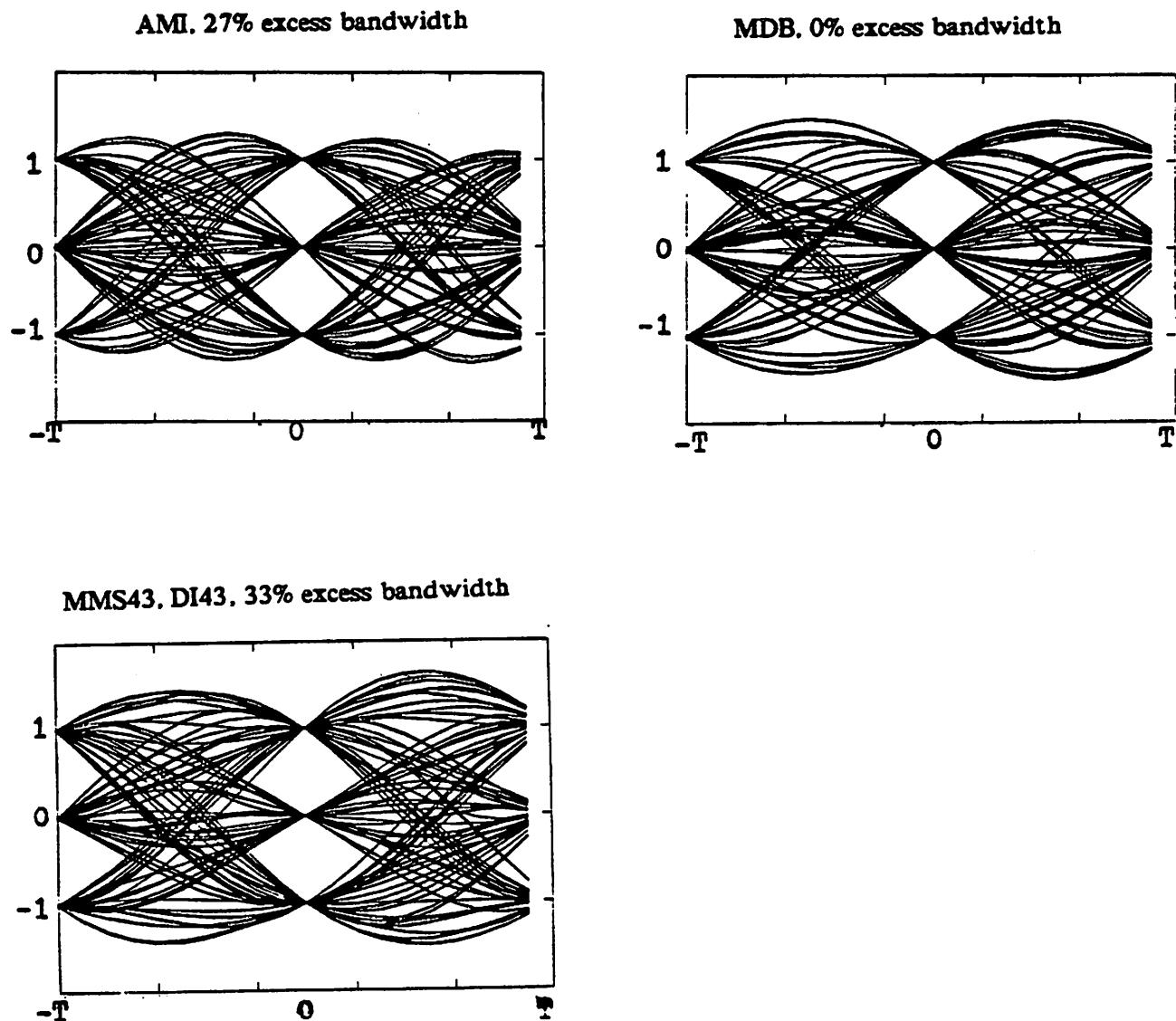


Figure 4.1.2.5.1. Eye diagrams.

Note that the bandwidth required for the block codes is greater than that required for MDB and MMDB, in spite of the fact that the baud rate is lower, due to the requirement for greater excess bandwidth.

As another case of reference, the excess bandwidth requirements to get an eye width of 50% of T are given in the following table:

Line Code	EXCESS BANDWIDTH	TOTAL BANDWIDTH (kHz)
Biphase	400%	380
AMI	40%	106
MDB	55%	118
MMDB	75%	133
MMS43	60%	98
DI43	60%	98

Table 4.1.2.5.2.

Although we expect that a horizontal eye width of 36% is totally adequate, this table illustrates that the conclusions are sensitive to the assumed eye opening. This larger eye opening is more favorable to the block codes. The remaining comparisons in this chapter are made for a 36% horizontal eye opening.

Some additional explanation of the assumptions in the case of BP is in order. A fair comparison in terms of eye opening is not obvious for BP, since there is a zero crossing in the center of the eye. Therefore, we determined the eye opening and excess bandwidths above by putting a transfer function  $1 - D^{0.5}$  in the receiver (in other words subtracting samples at the positive and negative peaks of the received pulse). The result is a more conventional pulse shape, for which the excess bandwidths can be compared meaningfully. However, this factor enhances high frequency signals and therefore degrades the NEXT and white noise SNR. In the following comparisons, BP1 is an equalized pulse without this additional factor, and BP2 equalizes to a pulse shape including this factor.

The power spectrum of the received signal at the decision slicer for equal transmitted average powers is shown in Fig. 4.1.2.5.2.

#### 4.1.2.6. Transmit Filter

The transmit filter is a two-pole Butterworth low-pass filter. The cutoff frequency of this filter does not affect the comparisons for near-end crosstalk, since any loss in this filter is made up in the receiver equalization, and hence is washed out in a NEXT interference path. Choosing as low a bandwidth for the transmit filter as possible is beneficial in reducing foreign crosstalk and RFI. However, choosing the bandwidth too low will result in reduced margin against foreign sources of interference and white noise, since the loss of the transmit filter will have to be compensated in the receive equalization. Therefore, in all cases the transmit filter bandwidth has been chosen to result in a penalty of 1 dB in white noise. This is done as the following: the receive filters for all the line codes other than MDB2 is given by the expression:

$$|Rcv(f)| = \left| \frac{\text{Raisedcosine}(f)}{P(f)Tx(f)C(f)} \right|$$

where  $P(f)$  is the frequency response of the pulse former.  $C(f)$  is the channel response, and  $Tx(f)$  is the frequency response of the transmit filter and is given as:

$$|Tx(f)|^2 = \frac{1}{1 + \left(\frac{f}{f_o}\right)^2}$$

For MDB2, the receive filter is given as:

$$|Rcv(f)| = \left| \frac{\text{Raisedcosine}(f)}{P(f)Tx(f)C(f)} (1 - e^{-j2\pi f T}) \right|$$

The 1 dB penalty is the noise enhancement relative to the systems whose transmit filters are brick-wall filters with cutoff frequencies chosen according to their required excess bandwidths. Thus, the receive filters of the reference systems are

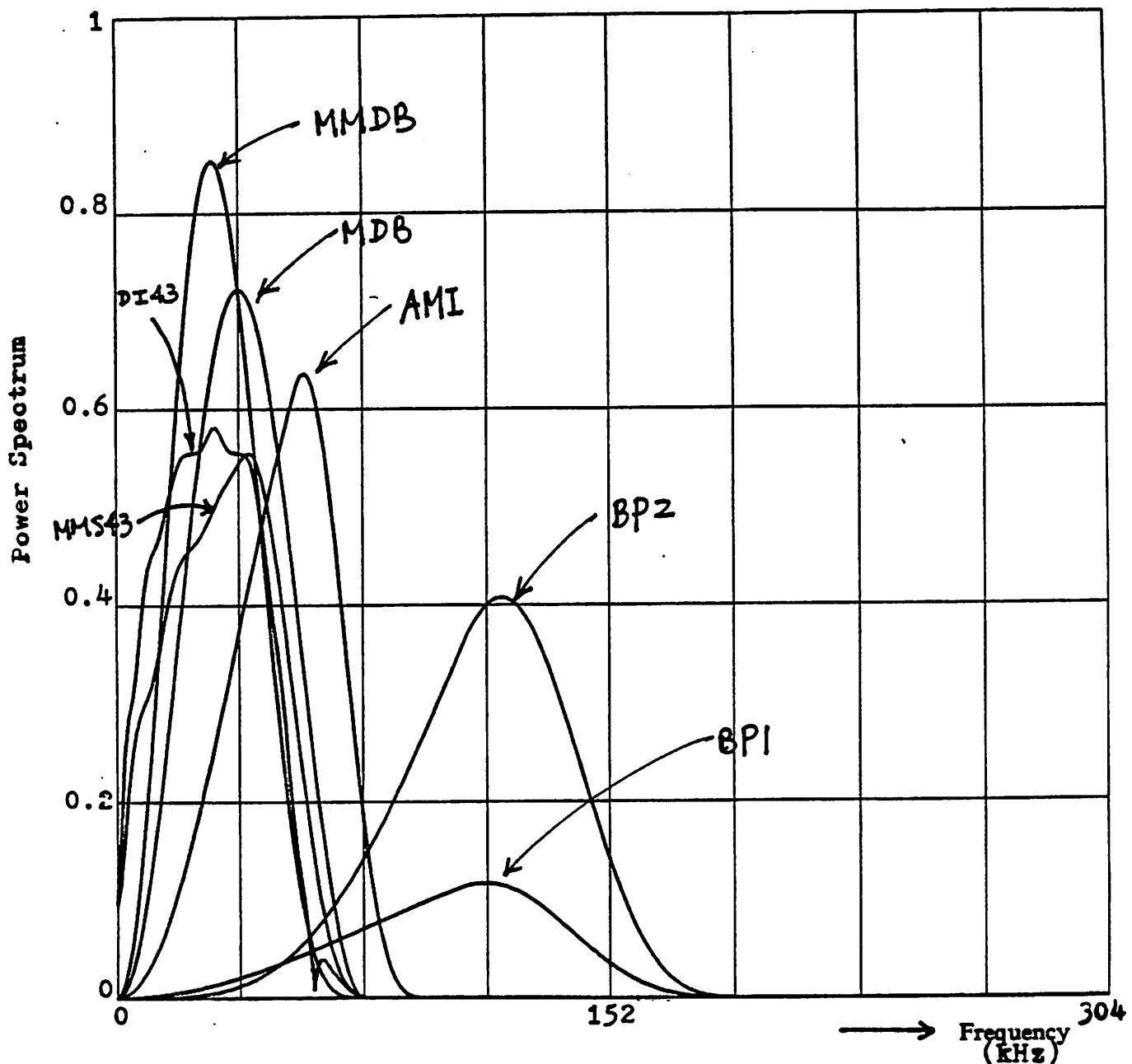


Figure 4.1.2.5.2. Power spectrum of signal at receiver slicer for equal transmitted average power.

$$|Rcv_{ref}(f)| = \left| \frac{\text{Raisedcosine}(f)}{P(f)C(f)} \right|$$

for all codes except MDB2, and

$$|Rcv_{ref}(f)| = \left| \frac{\text{Raisedcosine}(f)}{P(f)C(f)} (1 - e^{-j2\pi f T}) \right|$$

for MDB2. The cutoff frequency of the transmit filter  $f_c$  is chosen such that

$$10 \log \left| \frac{\int |Rcv(f)|^2 df}{\int |Rev_{ref}(f)|^2 df} \right| = 1$$

This 1 dB penalty will not affect the relative performance of the line codes in white noise, since it affects their absolute performance equally. It does of course affect the plots of transmitted signal power spectrum.

The resulting cutoff frequencies for the different line codes are shown in the table below:

Tx Cutoff Frequency	
Line code (excess BW)	Frequency (Hz)
Biphase (30%)	184200
AMI (27%)	86160
MDB1 (0%)	87750
MDB2 (0%)	66620
MMDB (0%)	87750
MMS43 and DI43 (33%)	69800

Table 4.1.2.6.1.

The power spectrum of the transmitted signal for these filter bandwidths and for equal average transmitted power is shown in Fig. 4.1.2.6.1.

The interesting fact to note is that MDB2 allows a lower cutoff frequency for the filter than MDB1. This is because the receiver equalizer requires less gain at high frequencies due to the  $(1 + D)$  factor. This lower cutoff frequency is beneficial in terms of reducing RFI and crosstalk into foreign systems.

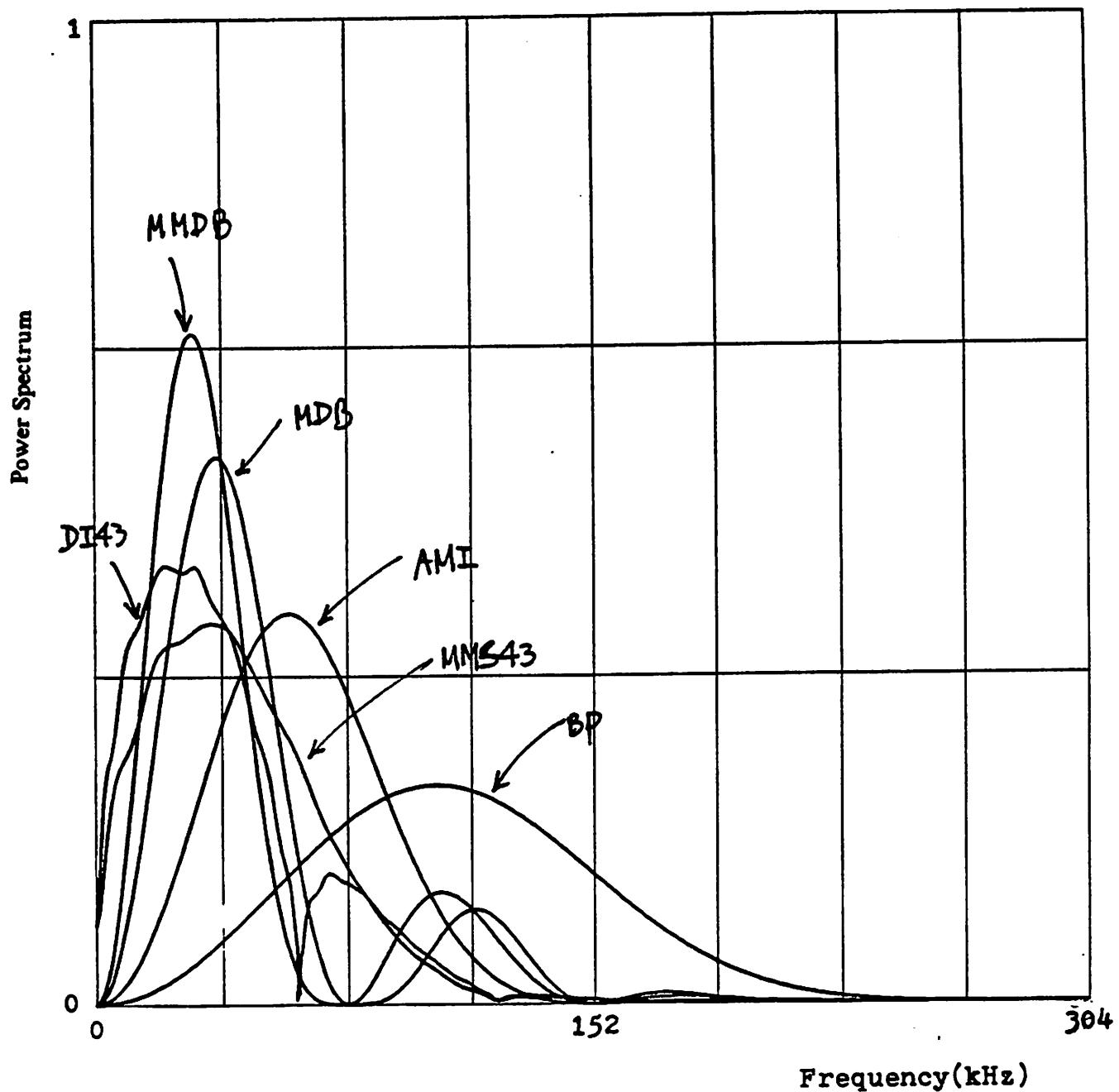


Figure 4.1.2.6.1. Power spectrum of transmitted signal for equal average transmitted power.

### 4.1.3. Comparative Performance

The performance of the different line codes are considered in this section. We first relate the probability of error to the SNR at the decision slicer input for the different line codes. The following three subsections consider the relative SNR performance in a near-end crosstalk dominated environment, in a white noise dominated environment, and in a foreign crosstalk environment. Finally, we compare the ranges of the line codes in a near-end crosstalk environment.

#### 4.1.3.1. Probability of Error

The most important direct performance measure is the probability of error. Our approach is to relate the probability of error to the signal-to-noise ratio in this section making the following assumptions:

1. The total noise and crosstalk interference is Gaussian. The assumption is justified for thermal noise and the accumulated crosstalk from many interferers, but not for impulse noise.
2. The signal level is the average signal power at the decision slicer input.

The probability of error for a given symbol  $s_n$  selected from an alphabet  $A = \{a_1, \dots, a_N\}$  can be written

$$\Pr[E] = \sum_i \Pr[E | s_n = a_i] \Pr[s_n = a_i].$$

We will assume that the probability of a single symbol error is a good measure of the overall probability of error. Since we assume ideal equalization in the receiver, there is no channel-induced intersymbol interference to compromise the independence of successive symbols. However, due to coding, successive symbols are not independent: certain symbol sequences are forbidden. If this property is used only for *detection* of some symbol errors, then it does not affect the overall probability of symbol errors. Assuming additive white Gaussian noise, independent of the data, the conditional pro-

babilities  $\Pr[E | s_i = a_i]$  can be given in terms of the tail of the Gaussian

$$\Pr[\eta_n > x] = \int_x^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-\alpha^2/2\sigma^2} d\alpha$$

where  $x$  is the distance from the symbol  $a_i$  to a decision boundary, and  $\eta_n$  is the noise. For convenience, we define the tail of a zero mean unit variance Gaussian distribution

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\alpha^2/2} d\alpha.$$

This function is related to the well-tabulated *error function* ( $\text{erf}(x)$ ) or *complementary error function* ( $\text{erfc}(x)$ ).

$$Q(x) = \frac{1}{2} \text{erfc}\left(\frac{x}{\sqrt{2}}\right) = \frac{1}{2} \left[1 - \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right].$$

It is easily shown that the tail of a Gaussian with any variance  $\sigma^2$  is given by

$$\int_x^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-\alpha^2/2\sigma^2} d\alpha = Q\left[\frac{x}{\sigma}\right].$$

We now individually consider the various signaling schemes. To make the comparisons meaningful, we assume equal average symbol power at the slicer. This average power can be normalized to unity without loss of generality, and with considerable gain in clarity.

**Biphase.** At the decision device, a BP signal is a binary antipodal signal. This is clear if it is considered a PSK signal. A binary antipodal signal with alphabet  $A = \{-1, 1\}$ , where each of the two symbols is equally likely, has unity average power. The optimum decision threshold is halfway between the two symbols, at zero, so

$$\Pr(E) = \frac{1}{2}Q(1/\sigma) + \frac{1}{2}Q(-1/\sigma) = Q(1/\sigma)$$

**AMI and MDB.** In AMI and MDB signaling, the alphabet is  $A = \{-\alpha, 0, \alpha\}$ , where  $\Pr[s_i = 0] = 1/2$  and  $\Pr[s_i = \alpha] = \Pr[s_i = -\alpha] = 1/4$ . For unity average power, we require that

$$\frac{1}{2}a^2 = 1$$

so  $a = \sqrt{2}$ . The optimum decision thresholds are[23]

$$x_{opt} = \pm \left( \frac{a}{2} - \frac{\sigma^2}{a} \ln\left(\frac{1}{2}\right) \right)$$

where  $\sigma^2$  is the variance of the noise. Using decision thresholds that are a function of the noise variance, however, is inconvenient at best, and we find that the improvement in probability of error compared to using decision thresholds halfway between the symbols is negligible, so we set the decision thresholds at  $x = \sqrt{2}/2$ . The zero symbol has probability 1/2 and is surrounded by two decision boundaries at distance  $x$  so it contributes a  $2 \times \frac{1}{2}Q(x/\sigma)$  to the probability of error. The contributions of the other two symbols are similarly found. The probability of error is therefore

$$\Pr(E) = Q(x/\sigma) + 1/4Q(x/\sigma) + 1/4Q(x/\sigma) = 1.5Q(\sqrt{2}/2\sigma)$$

**MMDB.** In MMDB signaling, the alphabet is  $A = \{-2a, -a, 0, a, 2a\}$ , where

$$\Pr[s_i = 0] = 3/8,$$

$$\Pr[s_i = a] = \Pr[s_i = -a] = 1/4,$$

and

$$\Pr[s_i = 2a] = \Pr[s_i = -2a] = 1/16.$$

For unity average power, we find that  $a^2 = 1$  so  $a = 1$ . Again, we find that use of the optimum decision thresholds[23] does not significantly affect the probability of symbol error, so we simply set the decision thresholds halfway between the symbols. The probability of error is therefore

$$\Pr(E) = \frac{11}{8}Q(1/2\sigma).$$

**MMS43 and DI43 Block Codes.** The choice of coding only affects the probability of error as a function of SNR at the slicer by affecting the probability of occurrence for each level. With both of the block codes we consider, it can be shown that the four states are equally likely, and the three levels from the alphabet  $A = \{-a, 0, a\}$  are

approximately equally likely. For a average power of unity, therefore, the levels are given by

$$2\sigma^2/3 = 1$$

which yields  $\sigma = \sqrt{3/2}$ . The probability of error is found to be

$$\Pr(E) = \frac{4}{3}Q\left(\frac{\sqrt{3/2}}{2\sigma}\right).$$

Subsequent sections will calculate the SNR for the two types of impairment, NEXT and white noise. The results of this section allow us to relate these results to the probability of error.

The probability of error was calculated for the line codes as a function of SNR. Drawing a horizontal line at the error probability equal to  $10^{-6}$ , we developed the following table which tabulates the relative SNRs required to achieve the same probability of error.

Line Code	SNR
BP1 and BP2	13.5db
AMI	16.6db
MDB1 and MDB2	16.6db
MMDB	19.6db
MMS43	18db
DI43	18db

Table 4.1.3.1.1.

Note that, as expected, the more levels at the decision slicer, the higher the SNR required to support the  $10^{-6}$  error rate.

#### 4.1.3.2. Near-End Crosstalk

Under most circumstances, NEXT is expected to be the dominant impairment which limits range for an echo cancellation U-transceiver. We compare in this section the relative SNRs for NEXT interference at the decision slicer. These SNRs are calcu-

lated under the assumption that the interfering systems are identical to the transceiver for which the performance is being calculated. The NEXT SNR is independent of transmitted signal power and any arbitrary gain introduced in the receiver. It is also independent of the partitioning of filtering functions between transmitter and receiver.

The following analysis refers to the transmitter and receiver structures in Fig.

#### 4.1.2.1. First we define the power transfer function $X_{next}$ :

$$|X_{next}(f)|^2 = K |f|^{\frac{3}{2}}$$

where  $K$  is an empirical constant. The power transfer function accounts for the mechanism of the coupling between twisted pairs. The NEXT power spectral density at the output of the receiver equalizer is given by

$$S_{next}(f) = S_{tx}(f) |X_{next}(f)|^2 |H_{rcv}(f)|^2$$

where  $H_{rcv}(f)$  is the receiver filtering (equalizer) and  $S_{tx}(f)$  is the transmitted power spectral density

$$S_{tx}(f) = S_{cdr}(f) |P(f) H_{tx}(f)|^2.$$

$P(f)$  is the transfer function of the pulse former, which we assume in the analysis to have an impulse response equal to a square pulse with width  $T/2$ .  $H_{tx}(f)$  is the transmit filter, assumed to be a two pole Butterworth filter.  $S_{cdr}(f)$  is the power spectral density of the discrete-time data signal after the coder

$$S_{cdr}(f) = S_{sequence}(f) |H_{cdr}(f)|^2$$

where  $S_{sequence}(f)$  is the power spectrum of the bit stream after the scrambler, and  $H_{cdr}(f)$  is the transfer function of the coder.

$$|H_{cdr}(f)| = \begin{cases} 2 |\sin(\pi f T)| & \text{for AMI} \\ 2 |\sin(2\pi f T)| & \text{for MDB} \\ 4 |\sin(2\pi f T) \cos(\pi f T)| & \text{for MMDB} \\ B^{1/2}(f T) & \text{for MMS 43} \\ B^{1/2}(2f T) & \text{for DI 43} \end{cases}$$

where the functions  $B(\cdot)$ , for the block codes are derived in Appendix F. Thus

$$S_{cdr}(f) = \begin{cases} \frac{1}{T} \sin^2(\pi f T) & \text{for AMI} \\ \frac{1}{T} \sin^2(2\pi f T) & \text{for MDB} \\ \frac{4}{T} \sin^2(2\pi f T) \cos^2(\pi f T) & \text{for MMDB} \\ B(fT) & \text{for MMS43} \\ B(2fT) & \text{for DI43} \end{cases}$$

Therefore

$$S_{next}(f) = S_{cdr}(f) |P(f)H_{tx}(f)|^2 |X_{next}(f)|^2 |H_{rcv}(f)|^2$$

The receive filter is designed so that the received equalized pulse is a raised cosine pulse, i.e.,

$$P(f)H_{tx}(f)C(f)H_{rcv}(f) = \text{Raisedcosine}(f)$$

where  $C(f)$  is the transfer function of the channel. Consequently

$$S_{next}(f) = K S_{cdr}(f) |\text{Raisedcosine}(f)|^2 \frac{1}{|C(f)|^2} |f|^{\frac{3}{2}}$$

and the NEXT power is

$$P_{next} = \int_{-\infty}^{\infty} S_{next} df$$

Table 4.1.3.2.1 summarizes these relative SNRs, normalized to AMI. Also shown in this table is a set of SNRs which have been adjusted to account for the probability of error using the results of Section 4.1.3.1. This right column is therefore the one to concentrate on.

NEXT SNR (36% eye opening)		
Line Code	SNR (dB)	SNR(dB) adjusted for $P_e$
BP1	-11.92	-8.82
BP2	-12.96	-9.86
AMI	0.00	0.00
MDB1 and MDB2	4.13	4.13
MMDB	7.04	4.04
MMS43	5.30	3.90
DI43	6.79	5.39

Table 4.1.3.2.1.

This table illustrates that the performance of MDB and MMDB is quite comparable to that of the block codes in a NEXT dominated environment. DI43 offers the best performance, with an SNR which is 1.35 dB better than MDB. Note that MMDB is quite attractive, until the increased probability of error due to its five level eye is accounted for, in which case it is slightly inferior to MDB. Table 4.1.3.2.2 and 4.1.3.2.3 are the NEXT SNR for 37% and 50 % eye openings respectively.

NEXT SNR (37% eye opening)		
Line Code	SNR (dB)	SNR(dB) adjusted for $P_e$
BP1	-11.99	-8.89
BP2	-13.06	-9.96
AMI	0.00	0.00
MDB1 and MDB2	4.16	4.16
MMDB	7.04	4.04
MMS43	5.31	3.91
DI43	6.80	5.40

Table 4.1.3.2.2.

NEXT SNR (50% eye opening)		
Line Code	SNR (dB)	SNR(dB) adjusted for $P_e$
BP1	-12.32	-9.22
BP2	-13.46	-10.36
AMI	0.00	0.00
MDB1 and MDB2	4.32	4.32
MMDB	6.33	3.33
MMS43	5.05	3.65
DI43	6.61	5.21

Table 4.1.3.2.3.

#### 4.1.3.3. White Noise

The SNR at the decision slicer is compared in this section. This SNR is indicative of the immunity of the line code to interference from foreign systems and thermal noise. This relative immunity is important primarily since the transmit level will be lowered to reduce foreign crosstalk (without affecting the NEXT SNR) to a level where other forms of interference and noise become important.

The noise power is given by

$$P_{white noise} = \int_{-\infty}^{\infty} \frac{N_0}{2} |H_{rcv}(f)|^2 df$$

Since

$$P(f)H_{tx}(f)C(f)H_{rcv}(f) = Raisedcosine(f)$$

$$|H_{rcv}(f)|^2 = \left| \frac{Raisedcosine(f)}{C(f)H_{tx}(f)P(f)} \right|^2$$

Therefore the noise power is

$$P_{noise} = \int_{-\infty}^{\infty} \frac{N_0}{2} \left| \frac{Raisedcosine(f)}{C(f)H_{tx}(f)P(f)} \right|^2 df$$

Table 4.1.3.3.1 compares the white noise SNR of the line codes, and again a column is included which normalizes these SNRs to a constant error probability.

Relative white noise SNR (36% eye opening)		
Line Code	SNR (dB)	SNR adjusted for $P_e$
BP1	-10.31	-7.21
BP2	-11.17	-8.07
AMI	0.00	0.00
MDB1	-0.37	-0.37
MDB2	2.77	2.77
MMDB	-0.39	-3.39
MMS43	3.25	1.85
DI43	3.44	2.04

Table 4.1.3.3.1.

These results show that the block codes are less sensitive to white noise interference than AMI by about 2 dB. Interestingly, MDB1 has less immunity to white noise than AMI, but MDB2 has the best white noise immunity of all the line codes. This is due to the lower gain at high frequencies in the receiver equalizer as a result of the zero at half the baud rate.

Fig. 4.1.3.3.1 shows the white noise spectrum at the decision slicer input for the non-block codes. These curves also correspond to the magnitude-squared of the equalizer transfer function. Note the fast cutoff required in MDB1 and MMDB to maintain

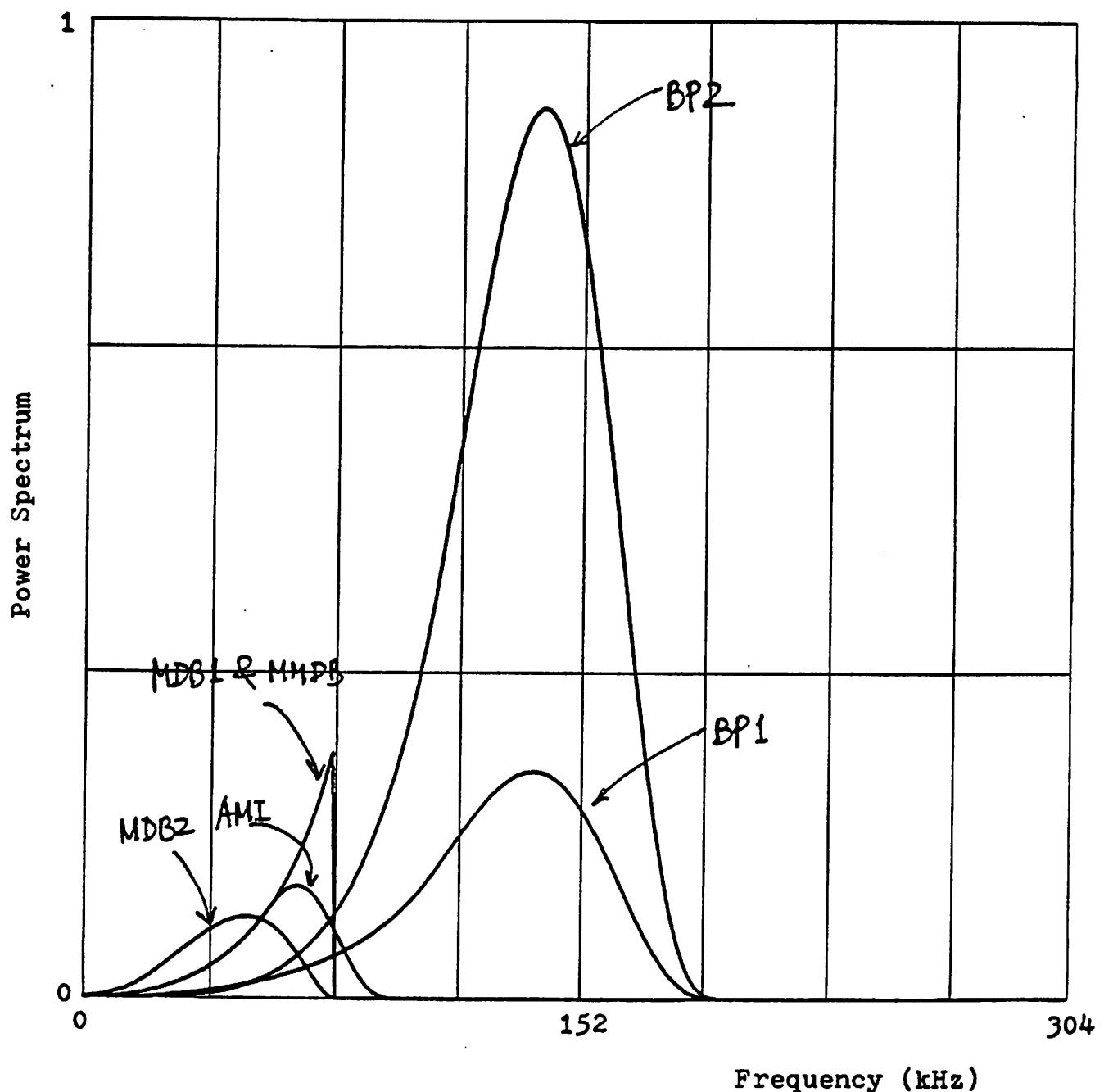


Figure 4.1.3.3.1. Power spectrum of white noise at decision slicer input.

zero excess bandwidth, which is eliminated in MDB2. The improved white noise immunity of MDB2 is also evident in this Figure.

#### 4.1.3.4. Foreign Crosstalk

The relative crosstalk power into another cable pair was also calculated. This quantity gives some indication of the crosstalk into a foreign system, such as T-carrier. However, this quantity should be used with caution, since taking account of the frequency-dependent gain in a repeater in the foreign system might result in a quite different comparison. This foreign crosstalk is also closely related to RFI, and codes with the least foreign crosstalk are likely to also have the least RFI.

The foreign NEXT power spectral density is given by :

$$S_{foreignnext}(f) = S_{tx}(f)|X'_{next}(f)|^2$$

where  $|X'_{next}(f)|$  is the power transfer function for foreign NEXT.

$$|X'_{next}(f)|^2 = K'|f|^{\frac{3}{2}}$$

Therefore the foreign NEXT power is:

$$P_{foreignnext} = \int_{-\infty}^{\infty} K' S_{cdr}(f) |P(f) H_{tx}(f)|^2 |f|^{\frac{3}{2}} df$$

The relative foreign crosstalk powers are tabulated in table 4.1.3.4.1.

Relative foreign NEXT		
Line Code	Interference power (dB)	Adjusted for Tx Level
BP	3.66	10.87
AMI	0.00	0.0
MDB1	-1.17	-0.8
MDB2	-0.65	-3.42
MMDB	-1.89	1.5
MMS43	-2.18	-4.03
DI43	-2.64	-4.68

Table 4.1.3.4.1.

In the third column, we have taken account of the white noise performance of the

various codes quantified in the previous subsection. The average transmit level for MDB2 can be reduced by 2.77 dB relative to AMI to yield the same immunity to white noise interference. This adjustment for transmitted power is incorporated into the third column for all the line codes. Concentrating on this third column, the two block codes and MDB2 have comparable foreign crosstalk performance. With respect to this foreign crosstalk, MDB2 is inferior to MDB1 because it is missing the pulse-shaping zero in the transmitted power spectrum at the half baud rate. This missing zero is compensated to some extent by the lower cutoff frequency of the transmit filter, however. In fact, the transmitted power spectrum of AMI and MDB2 are identical except for the lower cutoff frequency for the transmit filter in MDB2. This lower cutoff frequency results in slightly less foreign crosstalk at the same average transmitted power for MDB2 as compared to AMI.

#### 4.1.3.5. Transceiver Range

Thus far the comparisons have been relative measures of the performance of the different line codes. In this section we attempt to quantify the impact of this performance on transceiver range. This comparison was done only for a NEXT dominated environment, and requires some assumptions about the absolute level of crosstalk interference.

In order to calculate the range of the different schemes, our approach is to determine the required signal to noise ratio for a given probability of error and then to vary the cable length until that value is achieved. Intuitively, the mechanism is as follows. At each receiver, the receive filter/equalizer must compensate for the cable loss and therefore implements a transfer function with a characteristic that increases with frequency. This results in noise enhancement and a reduction in the signal to noise ratio with increasing length.

Using the notation of Section 4.1.3.2, the power spectrum of the NEXT at the receiver input is

$$S_{\text{x talk}}(f) = S_{tx}(f) \frac{1}{X_{f_0}} \left| \frac{f}{f_0} \right|^{\frac{3}{2}}$$

where we have substituted  $\frac{1}{X_{f_0}}$  for the empirical constant  $K$ .  $X_{f_0}$  is then the value of NEXT coupling loss at the reference frequency of  $f_0$ .

The signal to noise ratio at the output of the receive filter/equalizer is then

$$SNR = \frac{\int_{-\infty}^{\infty} S_{cdr}(f) | \text{Raisedcosine}(f) |^2 df}{\int_{-\infty}^{\infty} \frac{1}{X_{f_0}} \left| \frac{|f|}{f_0} \right|^{\frac{3}{2}} S_{cdr}(f) \left| \frac{\text{Raisedcosine}(f)}{C(f)} \right|^2 df}$$

In our simulations, we have considered a range of values for  $X_{f_0}$  of 40 to 75 dB at 80 kHz. This encompasses most of the crosstalk levels encountered in practice. Below are tabulated the results for  $X_{f_0} = 72$  dB for AWG 24 twisted wire pairs. As a reminder, we have listed in the table the SNR *at the slicer* that achieves a probability of error of  $10^{-6}$ , which reflects differences due to the number of levels and the relative frequency of occurrence of each level. Recall that NEXT dominated performance is unaffected by the partitioning of filtering between receiver and transmitter, so only one set of figures is given for MDB.

Coding Scheme	Excess Bandwidth (%)	SNR (dB) for $P(E) = 10^{-6}$	Range (km)
Biphase	30	13.5	6.1
AMI	27	16.6	7.4
MDB1 and MDB2	0	16.6	8.1
MMDB	0	19.6	8.2
MMS43	33	18.0	8.1
DI43	33	18.0	8.3

**Table 4.1.3.5.1. Range versus Coding Scheme for AWG 24  
(NEXT Coupling Loss = 72 dB at 80 kHz)**

In our analysis we have ignored the effect of cyclostationarity on the NEXT. As shown in [24], the effect of this assumption is to underestimate the crosstalk power by less than 1 dB.

#### 4.2. Comparison of Line Codes with Optimum DFE Design

In this section, we compare the performance of different line codes based on optimum DFE design. The following line codes are compared in a near-end crosstalk and white noise environment: AMI (dicode partial response), MDB (modified duobinary or Class IV partial response), and two block codes, MMS43, and DI43. In the case of MDB, a factor  $(1 + D)$  is included in the receiver rather than the transmitter. These codes are compared by constraining the excess bandwidth of an equalized raised cosine pulse to yield equal horizontal eye openings, zero-forcing the precursor intersymbol interference, and optimizing the feedback taps of a DFE to minimize the noise or crosstalk at the slicer input.

##### 4.2.1. Assumptions

Several assumptions have to be made in the relative comparisons. Care has been exercised to insure that the same assumptions are applied to the different line codes. Generally these assumptions are identical to those listed in Section 4.1.2 with the

single exception that we have gone to a more idealized transmit filter (the earlier filter used a Butterworth filter). The assumptions are summarized again in this section.

#### 4.2.1.1. Transmitted Data Stream

The transmitted data bit stream is assumed to consist of independent equally likely binary symbols.

#### 4.2.1.2. Transmitted Pulse

The transmitted pulse has a Fourier transform equal to the square root of a raised cosine response (see Section 4.4).

#### 4.2.1.3. Baud Rate

The baud rate is assumed to be 122 kHz for MMS43 and DI43, and 152 kHz for AMI and MDB.

#### 4.2.1.4. Transmit Filter

The transmit filter has a transfer function equal to the square-root of a raised cosine filter characteristic. This transmit filter is assumed to be driven by an impulse. The excess bandwidth of this raised cosine filter is the same as the assumed excess bandwidth of the equalized pulse in the receiver. For example, if the equalized pulse has an excess bandwidth of 27%, then the excess bandwidth of the transmit filter is also 27%.

The square-root of the equalized pulse response is known to be the optimum transmit filtering for a channel with a flat attenuation characteristic and additive white noise. While this response is not optimum in the present case, where the channel response is definitely a function of frequency, it is nevertheless a common choice for a transmit filter response where the channel is unknown or time-varying. This choice is thus less arbitrary than the Butterworth filter considered in Section 4.1, and is also

more idealized.

#### 4.2.1.5. Line

Four lines are considered in this comparison. The impulse response and the frequency response of these four lines is shown in Fig. 4.2.1.5.

##### 4.2.1.5.1. Line 1

This line is a simple 18 kft 24 AWG line with no bridged taps.

##### 4.2.1.5.2. Line 2

This line was kindly provided by Bell Communications Research as a representative line from their data base which demonstrated marginally poor performance. The configuration of this line follows:

Series resistance 135.000000 ohms  
Transmission line 3.069600 km long with gauge AWG26  
Transmission line 0.027430 km long with gauge AWG24  
Transmission line 1.213100 km long with gauge AWG26  
Transmission line 0.071018 km long with gauge AWG22  
Transmission line 0.118570 km long with gauge AWG24  
Shunt resistance 135.000000 ohms

##### 4.2.1.5.3. Line 3

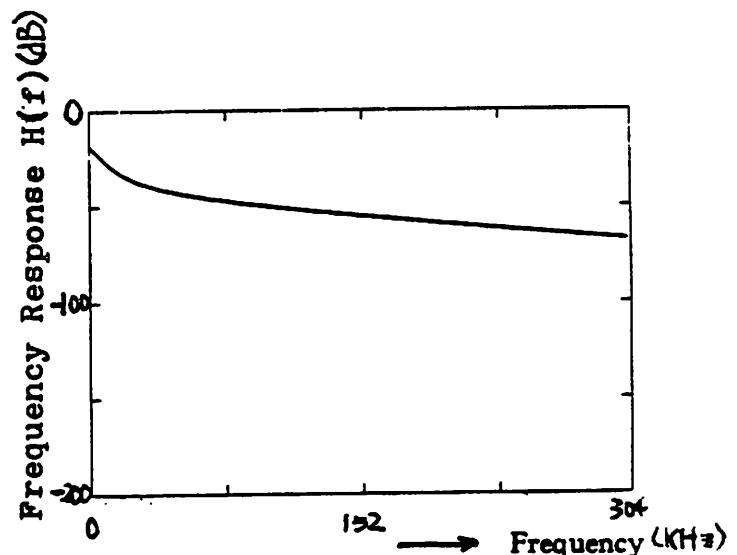
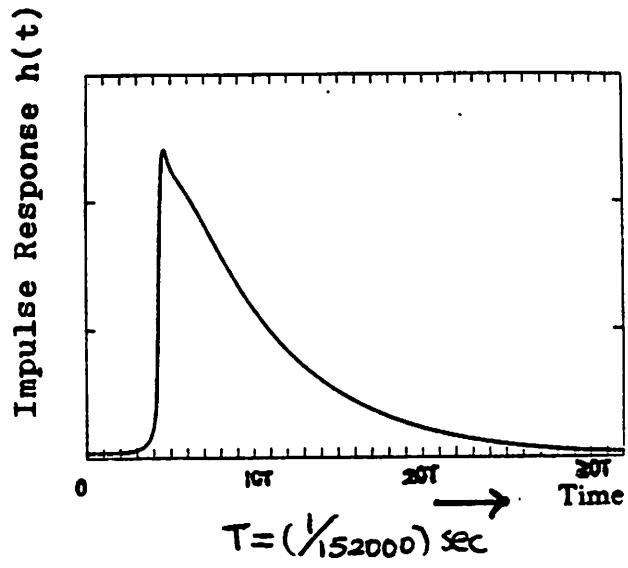
This is another line provided by Bell Communications Research in the same category. The configuration is:

Series resistance 135.000000 ohms  
Transmission line 2.684400 km long with gauge AWG26

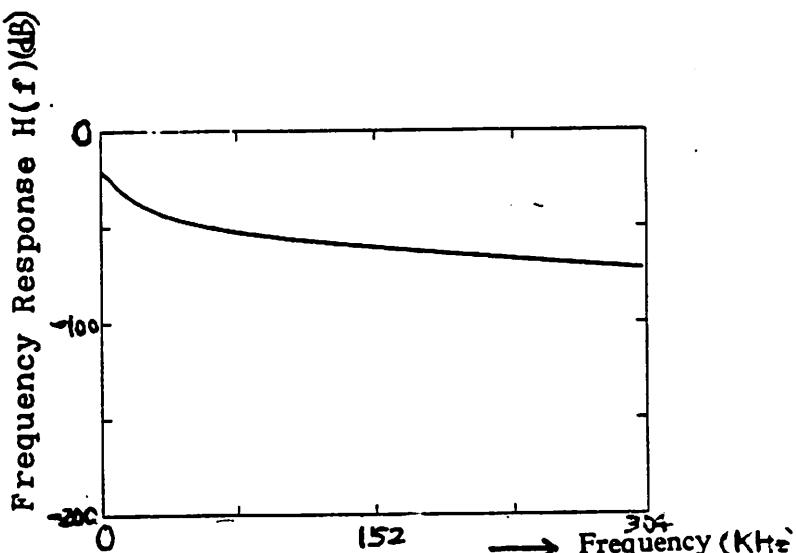
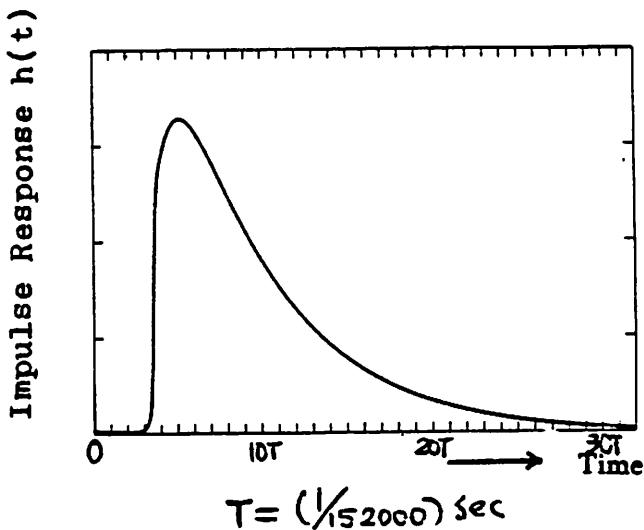
Bridge Tap consisting of:  
Transmission line 0.021336 km long with gauge AWG26

Transmission line 0.748590 km long with gauge AWG26  
Transmission line 0.359970 km long with gauge AWG22  
Transmission line 0.319130 km long with gauge AWG24  
Transmission line 0.491030 km long with gauge AWG26

Bridge Tap consisting of:  
Transmission line 0.060960 km long with gauge AWG26

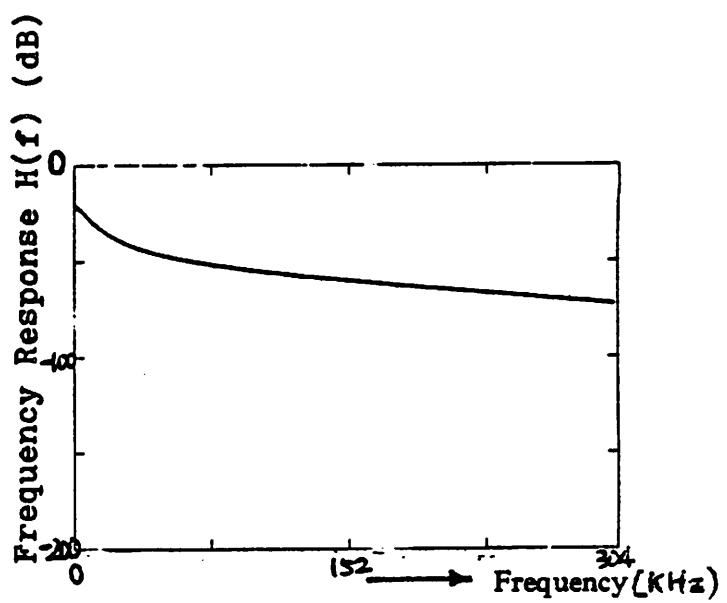
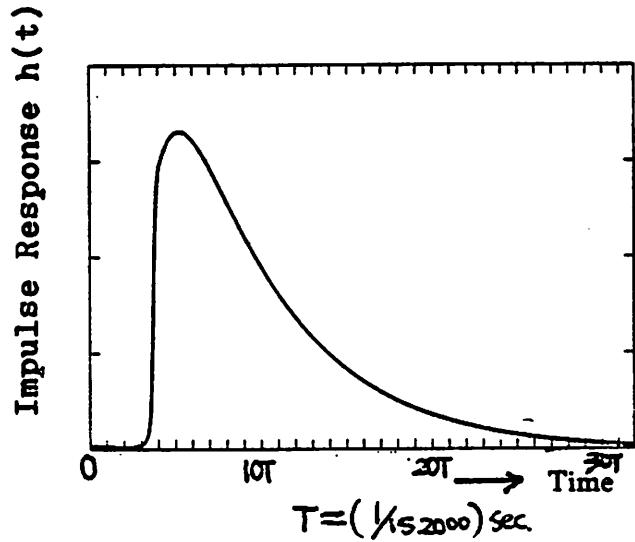


Line 1

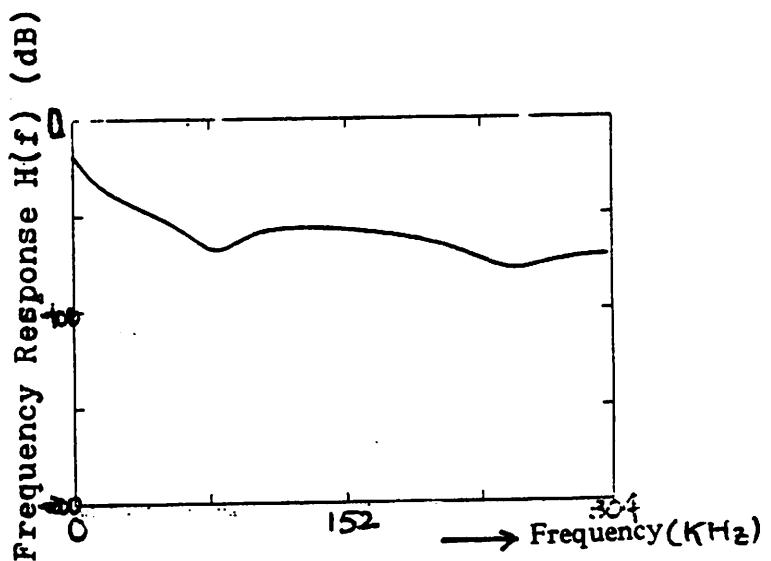
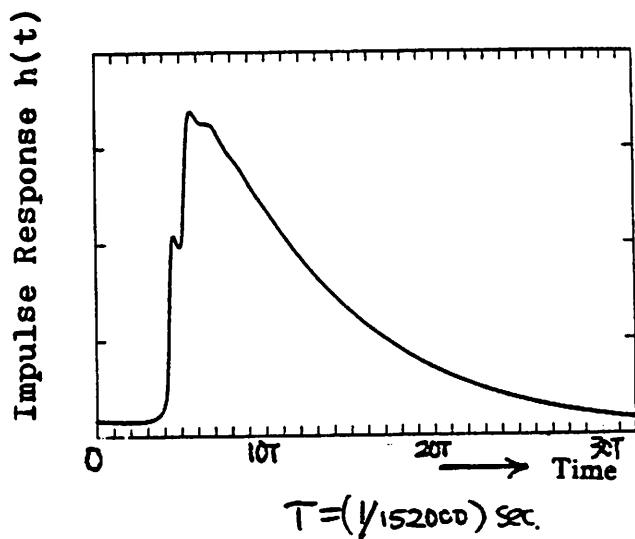


Line 2

Figure 4.2.1.5. Impulse response and frequency response of cables.



Line 3



Line 4

Figure 4.2.1.5.

Shunt resistance 135.000000 ohms

#### 4.2.1.5.4. Line 4

This line was concocted to be pathologically poor. It should demonstrate very poor performance because the bridged taps are chosen to maximize the loss of the line within the bandwidth of interest. This 18 kft line has two bridged taps, both adjusted to give approximately a one baud reflected echo group delay at 152 kHz. The result is to maximize the loss at 76 kHz, which is half the baud rate of the AMI and MDB systems, and is within the bandwidth of the block codes. Fig. 4.2.1.5 confirms that this line displays extreme loss characteristics within the bandwidth of interest. In addition, the loss has a large slope vs. frequency, which should magnify any differences among line codes due to differing bandwidths. The configuration is:

Series resistance 135.000000 ohms

Transmission line 2.000000 km long with gauge AWG24

Bridge Tap consisting of:

Transmission line 0.579000 km long with gauge AWG19

Transmission line 1.480000 km long with gauge AWG24

Bridge Tap consisting of:

Transmission line 0.579000 km long with gauge AWG19

Transmission line 2.000000 km long with gauge AWG24

Shunt resistance 110.000000 ohms

It should be noted that this line is probably too poor to support data transmission at 144 kb/s. The purpose of including this line is merely to magnify any differences among the line codes in the relative comparison. However, these differences will be larger than on any reasonable collection of lines that will support 144 kb/s transmission.

#### 4.2.1.6. Equalized Pulse

Our objective is to allow intersymbol interference at the output of the forward equalizer, and to remove this intersymbol interference using DFE taps. At the same time, we wish to maintain the constraint of equal horizontal eye opening at the slicer input, so that the susceptibility of the error probability to timing jitter is the same. We accomplish these two goals by defining the signal at the output of the forward equalizer to be of the form

$$r(t) = \sum_k x_k g(t - kT) \quad (4.2.1.6.1)$$

where  $g(t)$  is the equalized pulse shape and the sequence  $x_k$  is of the form

$$x_k = B_k + \sum_{j=1}^N d_j B_{k-j} \quad (4.2.1.6.2)$$

and  $B_k$  are the transmitted (three level) data symbols. Hence, the intersymbol interference is limited to the past  $N$  data symbols and can be eliminated by using an  $N$ -tap DFE. The horizontal eye width at the slicer input is controlled by choosing the received pulse shape  $g(t)$  to be an ideal raised-cosine pulse, where the excess bandwidth is chosen appropriately.

With the output of the forward equalizer constrained to be of the given form, for each value of  $N$  the intersymbol interference response

$$D(z) = \sum_{j=1}^N d_j z^{-j}$$

is chosen to minimize the noise variance at the slicer input. This analysis ignores the effect of error propagation. The theory of this optimization is covered in [20].

The excess bandwidths of the equalized pulse is chosen as in section 4.1. This results in a maximum horizontal eye opening of 36% of  $T$ , or in other words the peak jitter which closes the eye has magnitude equal to 18% of  $T$ .

#### 4.2.2. Comparative Performance

The performance of the different line codes are considered in this section. We first relate the probability of error to the SNR at the decision slicer input for the different line codes. The following three subsections consider the relative SNR performance in a near-end crosstalk dominated environment and in a white noise dominated environment.

##### 4.2.2.1. Near-End Crosstalk

The SNR's were calculated as a function of the number of DFE coefficients for a NEXT-dominated environment. As throughout this comparison, the DFE coefficients are optimized for the noise spectrum at the receiver input, which in this case is the transmitted power spectrum modified by the NEXT transfer function. The following tables, for lines 1 through 4 in order, summarize the relative SNRs, normalized to AMI. Also shown in this table is a set of SNRs which have been adjusted to account for the probability of error using the results of Section 4.1.3.1. This right half of the table is therefore the one to concentrate on.

NEXT SNR (Line 1)								
Line Code	SNR (dB)				SNR(dB) adjusted for $P_e$			
	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap	35-tap
AMI	0.0	4.2	6.5	12.9	0.0	4.2	6.5	12.9
MDB	4.1	5.3	7.8	13.1	4.1	5.3	7.8	13.1
MMS43	5.3	8.7	10.4	14.1	3.9	7.3	9.0	12.7
DI43	6.8	9.1	10.9	14.2	5.4	7.7	9.5	12.8

NEXT SNR (Line 2)									
Line	SNR (dB)				SNR(dB) adjusted for $P_e$				
	Code	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap	35-tap
AMI	AMI	0.0	4.6	7.2	14.2	0.0	4.6	7.2	14.2
MDB	MDB	4.6	6.0	8.6	14.4	4.6	6.0	8.6	14.4
MMS43	MMS43	5.8	9.7	11.6	15.7	4.4	8.3	10.2	14.3
DI43	DI43	7.4	10.2	12.2	15.9	6.0	8.8	10.8	14.5

NEXT SNR (Line 3)									
Line	SNR (dB)				SNR(dB) adjusted for $P_e$				
	Code	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap	35-tap
AMI	AMI	0.0	4.5	7.0	13.9	0.0	4.5	7.0	13.9
MDB	MDB	4.4	5.8	8.4	14.1	4.4	5.8	8.4	14.1
MMS43	MMS43	5.7	9.4	11.3	15.4	4.3	8.0	9.9	14.0
DI43	DI43	7.3	9.9	11.9	15.6	5.9	8.5	10.5	14.2

NEXT SNR (Line 4)									
Line	SNR (dB)				SNR(dB) adjusted for $P_e$				
	Code	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap	35-tap
AMI	AMI	0.0	8.9	12.6	21.1	0.0	8.9	12.6	21.1
MDB	MDB	8.9	12.1	14.9	21.4	8.9	12.1	14.9	21.4
MMS43	MMS43	10.7	16.9	19.6	24.2	9.3	15.5	18.2	22.8
DI43	DI43	12.9	17.7	20.2	24.4	11.5	16.3	18.8	23.0

In the absence of a DFE, the performance of MDB is slightly inferior to DI43 by an amount ranging from 1 to 1.5 dB for loops with severe dispersion. With a small number of DFE taps, AMI and the block codes improve their position relative to MDB. This is expected, since one interpretation of the MDB noise advantage is that it results from a single-tap DFE response  $1 + D$ , which is compensated by the precoding without error propagation. As the number of DFE taps grows large, the NEXT SNRs of all the line codes considered are approximately the same, within a small fraction of a dB. The single exception is on a loop designed to have extreme attenuation, for which the block codes have about a 1 dB advantage over AMI and MDB.

#### 4.2.2.2. White Noise

The SNR at the decision slicer for input white noise is compared in this section. This SNR is somewhat indicative of the immunity of the line code to interference from foreign systems, thermal noise, and impulse noise. This relative immunity is important primarily because the transmit level will be lowered to reduce the crosstalk into foreign systems (without affecting the NEXT SNR) to a level where other forms of interference and noise become important. The following tables, for lines 1 through 4 in order, compare the white noise SNR of the line codes, and again a column is included which normalizes these SNRs to a constant error probability.

Relative white noise SNR (Line1)								
Line	SNR (dB)				SNR(dB) adjusted for $P_e$			
	Code	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap
AMI	0.0	2.4	3.0	4.0	0.0	2.4	3.0	4.0
MDB	2.2	2.3	3.5	4.6	2.2	2.3	3.5	4.6
MMS43	3.1	5.5	6.2	7.0	1.7	4.1	4.8	5.6
DI43	3.3	5.7	6.4	7.2	1.9	4.3	5.0	5.8

Relative white noise SNR (Line2)								
Line	SNR (dB)				SNR(dB) adjusted for $P_e$			
	Code	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap
AMI	0.0	3.0	3.9	5.1	0.0	3.0	3.9	5.1
MDB	3.0	3.2	4.5	5.8	3.0	3.2	4.5	5.8
MMS43	3.5	6.6	7.5	8.5	2.1	5.2	6.1	7.1
DI43	3.7	6.8	7.7	8.7	2.3	5.4	6.3	7.3

Relative white noise SNR (Line 3)								
Line	SNR (dB)				SNR(dB) adjusted for $P_e$			
	Code	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap
AMI	0.0	2.8	3.6	4.9	0.0	2.8	3.6	4.9
MDB	2.8	2.9	4.3	5.5	2.8	2.9	4.3	5.5
MMS43	3.4	6.3	7.1	8.2	2.0	4.9	5.7	6.8
DI43	3.6	6.5	7.3	8.4	2.2	5.1	5.9	7.0

Line Code	Relative white noise SNR (Line 4)							
	SNR (dB)				SNR(dB) adjusted for $P_e$			
0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap	35-tap	
AMI	0.0	8.1	10.1	11.9	0.0	8.1	10.1	11.9
MDB	9.0	10.3	11.4	12.6	9.0	10.3	11.4	12.6
MMS43	7.1	12.6	15.0	16.3	5.7	11.2	13.6	14.9
DI43	7.3	12.8	15.2	16.5	5.9	11.4	13.8	15.1

#### 4.2.2.3. Foreign Crosstalk

The foreign crosstalk power given in Section 4.1.3.4 will not be appreciably affected. The absolute level of foreign crosstalk is determined only by the transmitter, and not by the receiver design. However, to the extent that the DFE reduces susceptibility to white noise and impulse noise interference, this may affect the foreign crosstalk if the transmit level is adjusted accordingly. The magnitude of this effect can be predicted from the foreign crosstalk results given in Section 4.1.3.4.

#### 4.2.3. Impact on Echo Cancellation

To simplify analog circuit design, the echo canceler should take account of the possibility of transmitted pulse asymmetry. All the ternary codes are amenable to the echo canceler implementation approach shown in Fig. 4.2.3.a[25]. A circuit, essentially a two-bit D/A converter, takes a two-bit representation  $(C_k, D_k)$  of the ternary transmitted level and generates the analog level  $x_k$ . Then for an arbitrary D/A converter, the transmitted level can be represented as

$$x_k = a + bC_k + cD_k + dC_k D_k$$

Since only three levels are required, one level, say (1,1), is not allowed. Then, the transmitted analog level can be represented as

$$x_k = a + bC_k + cD_k$$

for some constants  $(a, b, c)$ . Therefore, transmitted pulse asymmetry can be compen-

sated by building two binary cancelers with inputs  $C_k$  and  $D_k$ , plus a single d.c. offset coefficient. The three terms are summed to yield the echo replica. This is essentially the same as for a single binary canceler.

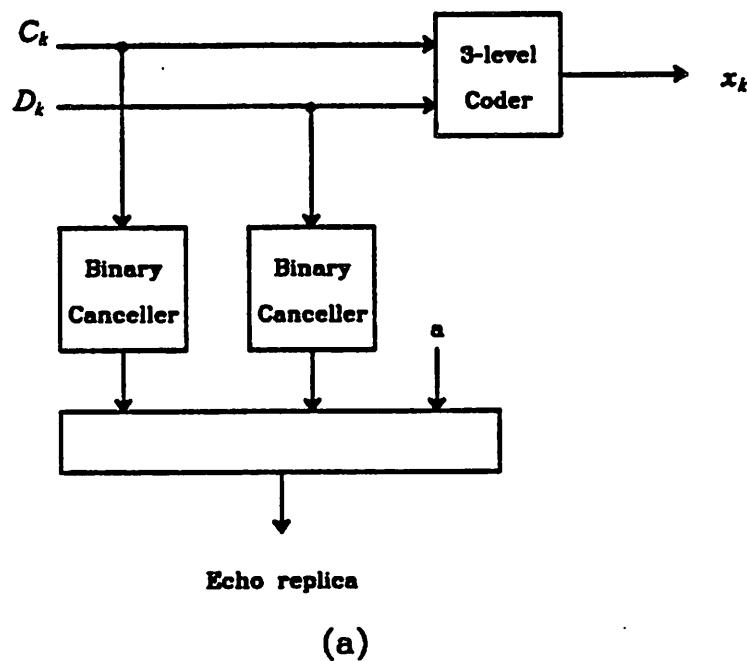
The two binary cancelers require the storage of  $2N+1$  coefficients, which is approximately double the storage requirements for a single binary canceler.

In the case of AMI and MDB another implementation which may be simpler is possible. This is shown in Fig. 4.2.3.b. In this case the binary precoding (modulo two accumulation), which is a nonlinear operation, is performed separately. For MDB, two modulo 2 accumulators are actually required, one for each of two half-rate bit streams. The output is fed into a binary canceler and is also converted to analog. The remainder of the coding, the  $(1 - D)$  or  $(1 - D^2)$  operations are performed in the analog domain, and can therefore be highly linear if not entirely accurate. As long as they are linear, they can be compensated by the echo canceler. For this case the coefficient memory of a single linear canceler is all that is needed, with one or two additional coefficients to account for the additional delay introduced by  $D$  or  $D^2$ .

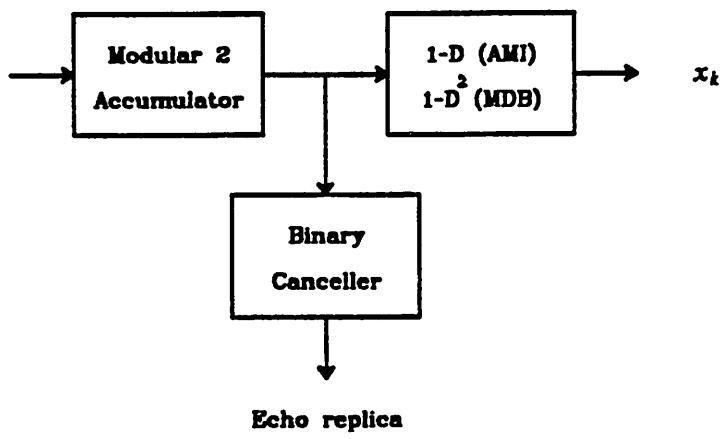
In summary, it appears that an overall reduction in complexity of the echo canceler may be possible in the AMI and MDB cases.

#### 4.2.4. Number of Echo Canceler Taps

The line code will affect the number of echo canceler taps required. In particular, the  $(1 - D)$  response of the AMI code is beneficial in reducing the number of echo canceler taps by approximately cancelling the tails of a long echo response. The nominal  $(1 - D^2)$  response of MDB is not so effective, and thus would result in an increase in the number of echo canceler taps. However, this assumes that  $(1 - D^2)$  is included in the loop from echo canceler input to the cancellation. In fact, as discussed in the following section, the  $(1 + D)$  response in the receiver can be placed *after* the echo can-



(a)



(b)

Figure 4.2.3.a. Two cancellation techniques for transmitted pulse asymmetry.

cellation. For this implementation, the number of echo canceler taps will be the same as for AMI, and less than for the block codes.

#### 4.2.5. MDB Implementation Alternatives

In this section we would like to point out that there are some alternative implementations of an MDB receiver which allow a tradeoff between the performance and implementation complexity.

The preferred implementation in terms of minimizing the noise enhancement in the receiver is shown in Fig. 4.2.5.1.a. In this realization, the factor  $(1 - D)$  is included in the transmitter to eliminate the d.c. content of the signal, and the factor  $(1 + D)$  is grouped with the receive filter to allow that filter to have zero excess bandwidth. Any equalization for channel characteristics (not shown) could be grouped with the receive filter, or in the more conventional configuration could be placed after the echo cancellation to avoid enhancement of the echo by the equalizer.

An alternative implementation is shown in Fig. 4.2.5.1.b, where the  $(1 + D)$  factor in the receiver has been moved after the echo cancellation. This partition will affect the receive filter design, since it will now have to have some excess bandwidth — it is not practical to implement a "brick wall" filter. In this case, if the cable forward equalizer is placed after the echo canceler it can be combined with the  $(1 + D)$  factor, or alternatively the  $(1 + D)$  factor can be implemented in isolation by simply adding adjacent samples. In any case, the  $(1 + D)$  factor is still beneficial in minimizing noise immunity since it places a zero at the half baud rate.

We calculated the effect of the realization of Fig. 4.2.5.1.b on the NEXT and white noise immunity by assuming the same excess bandwidth for the receive filter as in the AMI case — 27%. The results are in the following table. The effect on noise immunity is not appreciable.

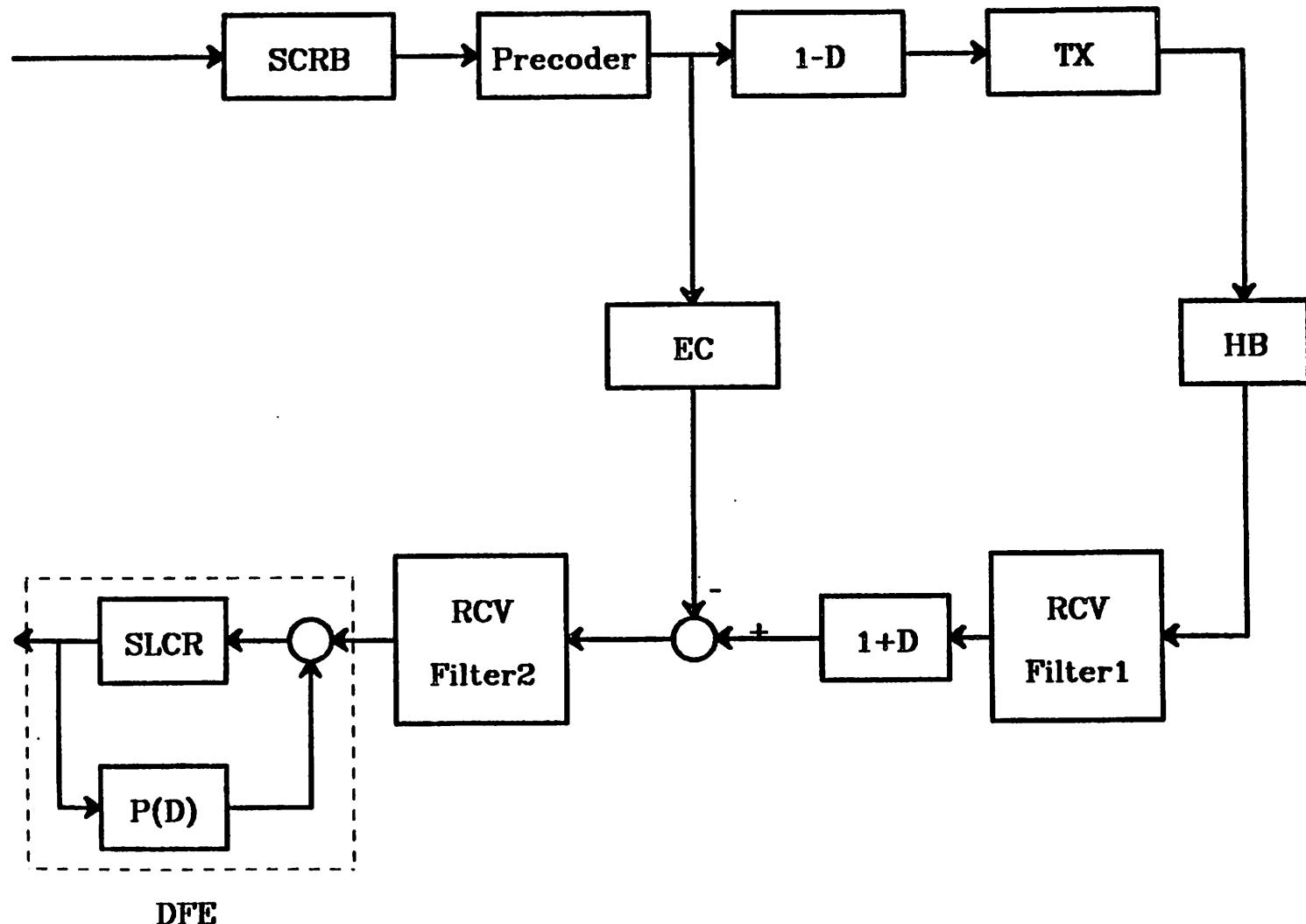


Figure 4.2.5.1.a. MDB Realization for minimum noise enhancement.

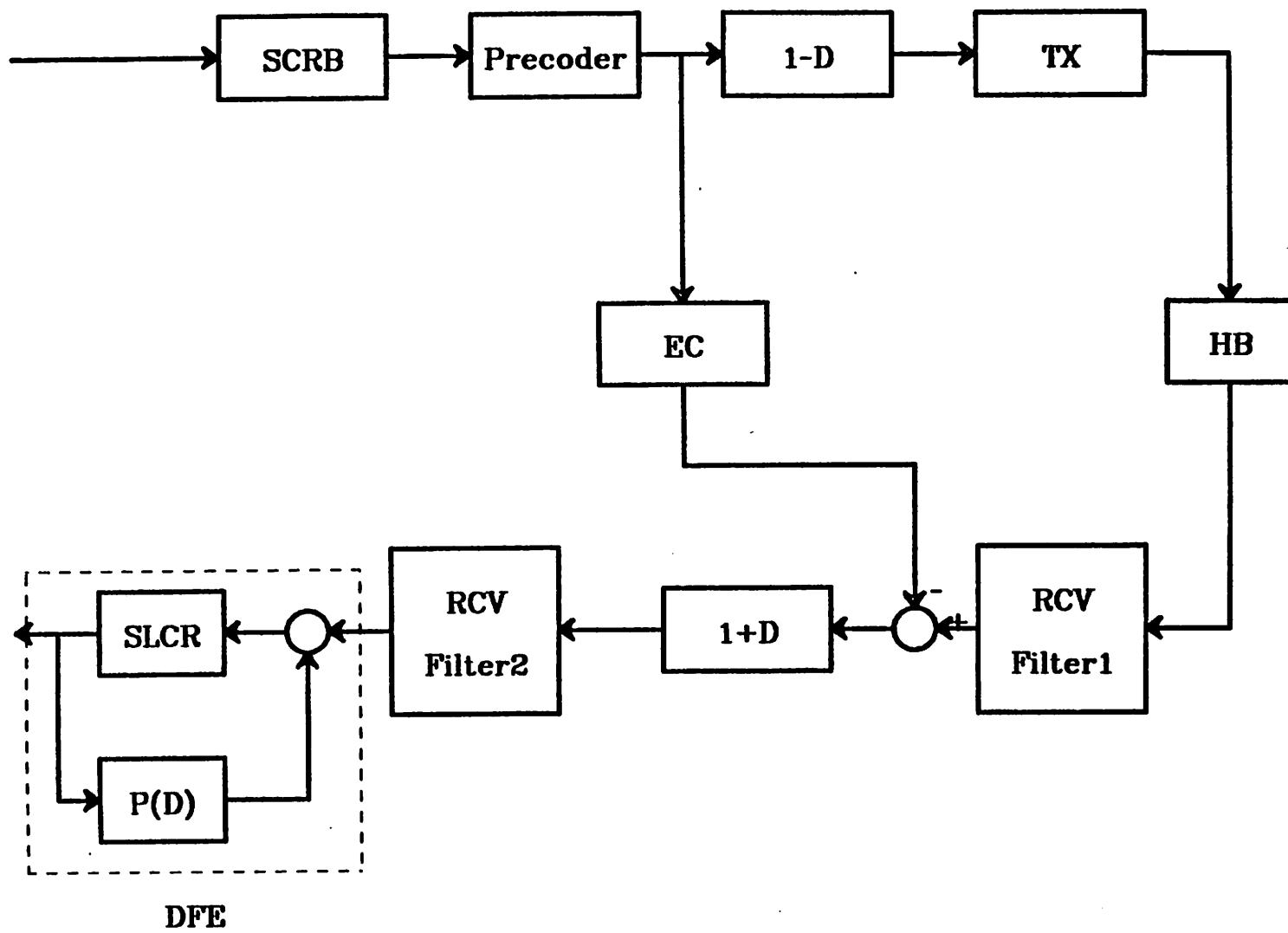


Figure 4.2.5.1.b. MDB realization for minimum number of echo canceller taps.

Line	Relative SNR (dB)							
	NEXT				White Noise			
	0-tap	1-tap	2-tap	35-tap	0-tap	1-tap	2-tap	35-tap
1	4.3	5.4	8.0	13.3	2.1	2.2	3.3	4.4
2	4.7	6.0	8.9	14.7	2.9	3.1	4.4	5.6
3	4.6	5.8	8.6	14.4	2.7	2.8	4.1	5.4
4	9.2	12.2	15.1	21.6	8.4	10.0	11.0	12.4

Table 4.2.5.1.

The main benefit of the realization of Fig. 4.2.5.1.b is that the echo response is identical to that of AMI. Hence, the number of echo canceler taps would be less for this MDB realization than for the MDB realization of Fig. 4.2.5.1.a.

A third realization which is potentially even simpler is shown in Fig. 4.2.5.1.c. This realization places the  $(1 + D)$  factor after the slicer. Since the slicer has no effect on the signal component, it doesn't matter whether the  $(1 + D)$  factor is placed before or after the slicer from the point of view of the data symbols. This realization will have poorer noise immunity, however, since the  $(1 + D)$  factor now has no effect whatsoever on the noise (which is absent at the slicer output), and hence its beneficial effects are lost. However, the realization of  $(1 + D)$  becomes very simple since we are simply adding adjacent three-level signals — a very simple digital circuit. The performance of this approach will be essentially equivalent to AMI, and the implementation is virtually identical. The only modification to AMI is the use of two modulo-two summations (toggle flip-flops) rather than one in the transmitter precoder, and the addition of the simple  $(1 + D)$  factor in the receiver.

The point of all this is that with the proposed MDB transmitter realization, there are a range of possibilities for the implementation of the receiver. At one extreme, we can realize the receiver with a complexity and performance virtually identical to AMI for applications where less performance is required (for example, in the PBX). With perhaps a slight increase in complexity, we can realize it as an MDB receiver, where the  $(1 + D)$  factor is somewhere before the slicer, and obtain improved performance over

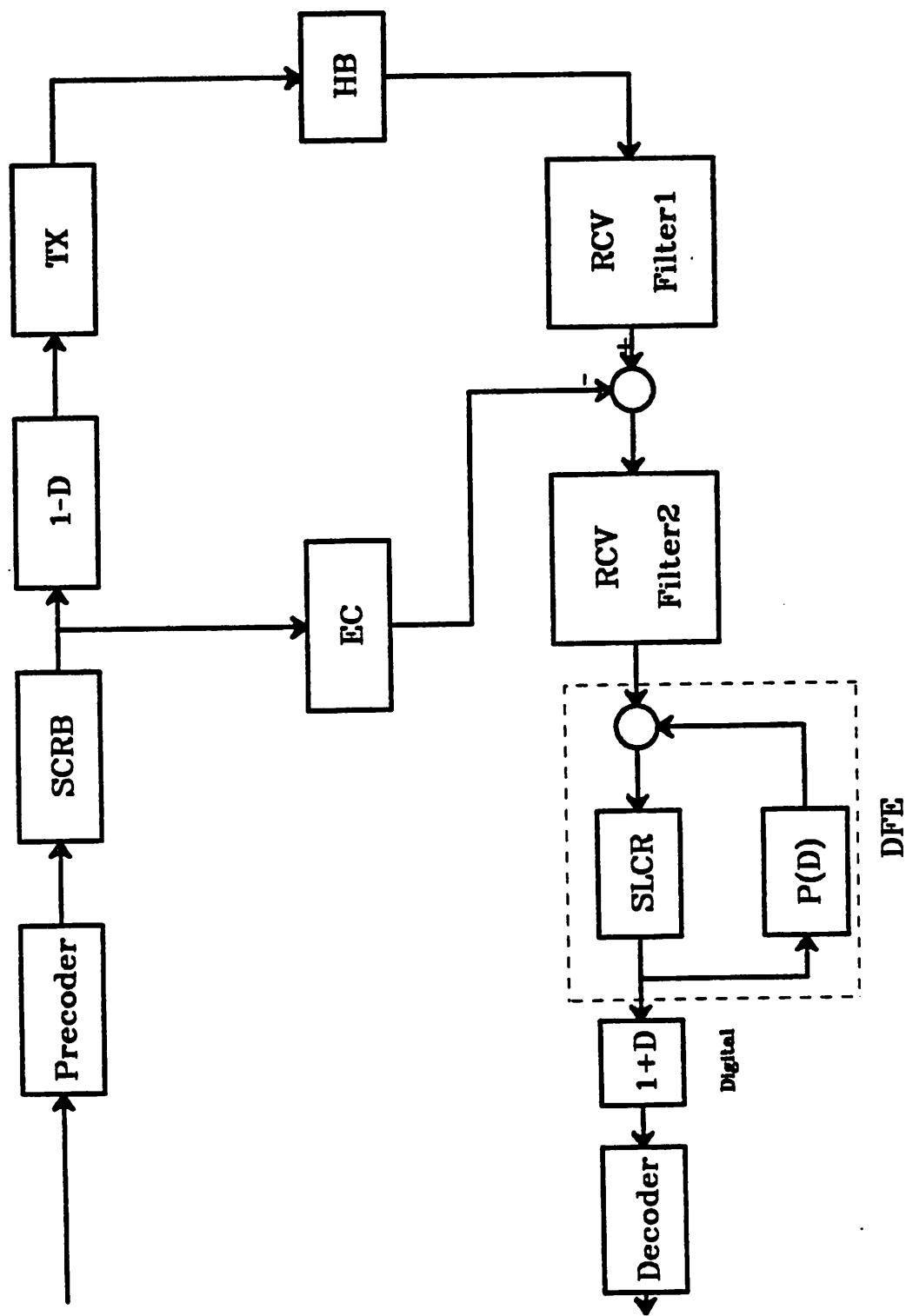


Figure 4.2.5.1.c. MDB realization equivalent to AMII (with loss of noise immunity).

AMI. Finally, as we add adaptive forward and feedback equalizers, in the limit we can obtain a performance equivalent to the block codes (as can AMI).

#### 4.3. Conclusions

In terms of this comparison, there does not appear to be any appreciable difference among the line codes for a large number of DFE taps. This demonstrates that all the line codes have approximately equivalent ultimate performance as the complexity of the receiver implementation gets large. The appropriate line code is therefore the one with the simplest implementation.

MDB offers an advantage over AMI in the range of simpler implementations. In particular, for an MDB transmitted signal, a receiver virtually identical to an AMI receiver (with a very slight modification) can be used, achieving AMI performance. Then with the slight increase in complexity required to implement an MDB receiver, approximately a 5 dB improvement in NEXT performance can be obtained on severe lines. As further complexity is added, in the form of DFE taps chosen for minimum noise enhancement, the difference between AMI and MDB narrows.

## CHAPTER 5

### Timing Recovery

Timing recovery is one of the most important elements in digital subscriber loops. Much work can be found in the literature regarding timing recovery algorithms. For DSL however, due to the economical implementation consideration, only a certain classes of algorithms are considered practical. Thus, in the evaluation of a timing recovery technique for DSL application, not only its effectiveness and robustness have to be considered, but much attention should also be paid to the simplicity of its hardware implementation.

In Section 5.2, we summarize some conventional timing recovery algorithms. These techniques are either operated in the continuous time domain, or in the discrete time domain with high sampling rate. The techniques suitable for DSL application are considered in Section 5.3, where only low sampling-rate, discrete-time techniques are discussed due to the requirement for economic integrated circuit implementation of the DSL.

Two important terminologies used in the discrete-time techniques should first be defined. they are the *timing update* (or driving function,  $\delta(\tau_k, r(\cdot))$ ) and the *timing function* ( $f(\tau)$ ). The timing update is the signal used to correct the timing phase and is a function of the current timing phase and the received signal, i.e.,

$$\tau_{k+1} = \tau_k + \beta \delta(\tau_k, r(\cdot)) \quad (5.0.1)$$

where  $\tau_k$  is the sampling phase at time  $k$ , and  $\beta$  is the step size of the adaptation. The timing function  $f(\tau)$  is the expectation of the timing update, i.e.,

$$f(\tau) = E[\delta(\tau, r(\cdot))] \quad (5.0.2)$$

The performance of a timing recovery technique depends heavily on the characteristic of its timing function. Some of the important characteristics of the timing function are:

(1) Root of the timing function:

Let  $\tau_s$  denote the steady-state sampling phase. From Equation (5.0.1), we have:

$$\begin{aligned} E[\tau_s] &= E[\tau_s] + \beta E[\delta(\tau_s, \{r_k\})] \\ &= E[\tau_s] + f(\tau_s) \end{aligned} \quad (5.0.3)$$

Thus

$$f(\tau_s) = 0$$

Therefore, the steady-state sampling phase  $\tau_s$  is determined by the root of the timing function. The timing function should be chosen such that its root is unique and at the desired location. For example, one may set up a performance measure function (such as ISI) which is to be maximized (or minimized). The timing function can be chosen to be the partial derivative of the performance measure with respect to timing phase. The root of the timing function therefore gives the phase that maximizes (or minimizes) the performance measure function.

(2) Shape of the timing function:

The shape of the timing function, especially its slope near the zero crossing, determines the transfer characteristic of the control loop. It affects the dynamics of the convergence. Therefore, it is important that the timing function be well behaved.

(3) Implementation complexity:

Although the gradient mentioned in (1) gives a good timing phase, it may require very complicated hardware to realize. In the DSL application, it is always important to minimize the hardware complexity while achieving good performance. Therefore, not only should the timing function give a good zero-crossing point

but also it should be easy to compute. If baud-rate sampling is the goal, the timing function should be expressible in terms of the baud-spaced samples of the impulse response.

Since some of the information required in calculating the timing function is not available, it is necessary to construct the timing update, a unbiased estimate of the timing function. The timing update should have small variance such that the timing jitter can be kept small.

### 5.1. Optimum Sampling Phase

The optimum sampling phase for a particular system is the one that minimizes the probability of data-detection error. However, for practical applications, more convenient suboptimal criteria, such as maximum eye opening or minimum mean square error, are used due to the highly nonlinear nature of the error probability function.

Since the error performance of a data transmission system depends jointly on the sampling phase and the equalization scheme used, the sampling phase that gives optimum performance would naturally depend on the equalization scheme. For the simplest case where no equalization is used, the measure of optimality therefore solely depends on the sampling phase itself. Thus, the criteria such as maximum eye opening or minimum mean squared error should be applied directly to the signal at the output of the sampler. For example, the sampling phase can be adjusted to maximize the eye opening  $D(\tau_o)$ :

$$D(\tau_o) = h(\tau_o) - (m - 1) \sum_{k=-\infty, k \neq 0}^{\infty} |h(\tau_o + kT)| \quad (5.1.1)$$

where  $h(t)$  is the system impulse response, and  $m$  is the number of levels. This can be done by adjusting the phase according to the gradient of  $D(\tau_o)$ :

$$\frac{\partial D(\tau_0)}{\partial \tau_0} = \frac{\partial h(\tau_0)}{\partial \tau_0} - (m-1) \sum_{k=-\infty, k \neq 0}^{\infty} \frac{\partial h(\tau_0 + kT)}{\partial \tau_0} sgn[h(\tau_0 + kT)] \quad (5.1.2)$$

In fact, this is the *improved sampled-derivative* method suggested by Burton and Saltzberg[26].

If the equalization scheme followed is a linear equalizer, whose performance has been shown to be very sensitive to the sampling phase, the optimum timing was shown to be the one that maximizes the band-edge component[27]. The reason that the performance of LE depends highly on the sampling phase is the following: the power spectral density of the signal after sampling (i.e., the folded power spectrum) depends on the sampling phase if the signal has excess bandwidth greater than zero. This is shown in Fig 5.1.1. We see that the aliases can add destructively resulting in a null.

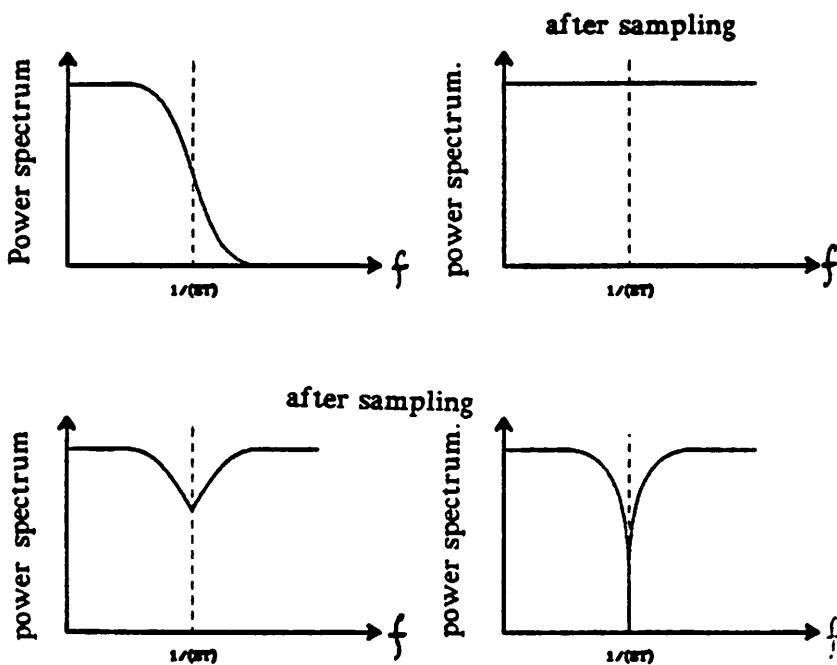


Figure 5.1.1. Power spectrum and sampling phase.

or depression, in the equivalent Nyquist band depending on the sampling phase. The job of the equalizer is to compensate the resultant spectrum such that it becomes flat, or close to flat, to minimize the ISI. However, in the process of equalizing the signal, noise may get enhanced. If the equalization scheme used is a linear equalizer, the amount of noise enhancement will depend heavily on the shape of the unequalized, folded signal spectrum. The further the signal spectrum is from flat, the more noise will be enhanced. Therefore, the performance of a linear equalizer depends heavily on the sampling phase. In fact, there will be severe noise enhancement if a null or a deep depression is present in the folded spectrum. A badly chosen sampling phase could sometimes result in tens of dB degradation in performance due to the noise enhancement[28].

On the other hand, the adverse characteristic is better compensated by a decision-feedback equalizer than a linear equalizer, since the DFE will not result in noise enhancement as the LE does. Therefore, we could expect that the performance of the DFE is much less sensitive to the sampling phase.

Note that for a minimum bandwidth system, class IV PRC for example, the performance of a linear equalizer will be independent of the sampling phase if the number of equalizer taps is infinite. For a finite tap linear equalizer however, the performance still depends on the sampling phase.

## 5.2. Conventional Timing Recovery Techniques

In the section, some of the important conventional timing recovery techniques are summarized. Although most of them are designed for either continuous time operation or discrete time with very high sampling rate, some of the ideas can be extended to the low sample-rate, discrete-time techniques. The techniques discussed are

- (1) Maximum likelihood method.
- (2) Spectral line method.
- (3) Threshold-crossing phase detector.
- (4) Sampled-derivative detector.
- (5) Modified sampled-derivative detector.

### 5.2.1. Maximum-likelihood timing recovery

The maximum-likelihood method is shown in [29]. In this method, the timing phase is adjusted to achieve the maxima of the likelihood function.

Let  $r(t)$  be the received signal:

$$r(t) = \sum_n a_n h(t - nT - \tau^*) + v(t)$$

where  $\{a_n\}$  are the data symbols taking on values  $\pm 1, \pm 3, \dots, \pm(L-1)$  with equal probability.  $h(t)$  is a bandlimited pulse whose peak value occurs at  $\tau^*$ , and  $v(t)$  is the noise. It is shown that the likelihood function for measuring the goodness of the estimation of  $\tau^*$  given the observed signal for  $T_s$  seconds is proportional to

$$L[r] \sim E \left[ \exp \left( -\frac{1}{2N_0} \int_0^{T_s} [r(t) - s(t; \tau)]^2 dt \right) \right]_a \quad (5.2.1.1)$$

where  $s(t; \tau) = \sum_n a_n h(t - nT - \tau)$ .  $E[\cdot]_a$  denotes expectation with respect to the data symbols, and  $N_0$  is the noise power density. If some reasonable assumptions are made, the likelihood function becomes:

$$L(r) \sim \prod_n \left\{ \frac{2}{L} \sum_{k=1, k \text{ odd}}^{L-1} \cosh \left( \frac{kd}{N_0} z_n(\tau) \right) \right\} \quad (5.2.1.2)$$

where

$$z_n(\tau) = \int_0^{T_s} r(t) h(t - nT - \tau) dt$$

is the sampled output (at time  $nT + \tau$ ) of the matched filter, and  $d$  is the spacing

between decision levels. Therefore, the maximum-likelihood timing recovery method needs a matched filter as the prefilter. The maxima of the  $L(r)$  can be obtained by finding the root of the derivative of  $L(r)$  with respect to  $r$ . This is equivalent to finding the root of the derivative of any monotonic function of  $L(r)$ . A convenient monotonic function is the logarithm:

$$\Lambda[r] \triangleq \ln L[r]$$

The derivative of  $\Lambda[r]$  with respect to  $r$  is given as :

$$\frac{\partial \Lambda}{\partial r} \sim \sum_n \frac{dz_n(r)}{d\tau} \tanh \left( \frac{(L+1)d}{N_0} z_n(r) \right) \quad (5.2.1.3)$$

Therefore, in the maximum-likelihood scheme, Equation (5.2.1.3) is used to drive the VCO. Fig. 5.2.1.1 is the diagram of the maximum-likelihood scheme where the received signal is prefiltered with a matched filter and the lower branch gives  $\frac{dz_n(r)}{d\tau}$ .

while the upper branch produces the function  $\tanh \left( \frac{(L+1)d}{N_0} z_n(r) \right)$ .

Notice that since a matched filter is used, the signal at the output of the matched filter is symmetrical around its peak. Therefore the optimum sampling phase is obviously at the center of the response.

As will be seen later, several well-known timing recovery techniques are in fact approximations to the maximum-likelihood method.

### 5.2.2. Spectral line method

In the spectral line method, the timing recovery is accomplished in two steps. First, a spectral line is generated at the baud frequency through a nonlinear operation. Second, a narrow band pass filter centered at the baud frequency, or equivalently, a PLL, is used to extract the timing information. Fig 5.2.2.1.a demonstrates the idea of this method. 5.2.2.1.b is the received signal, and 5.2.2.1.c shows the signal after

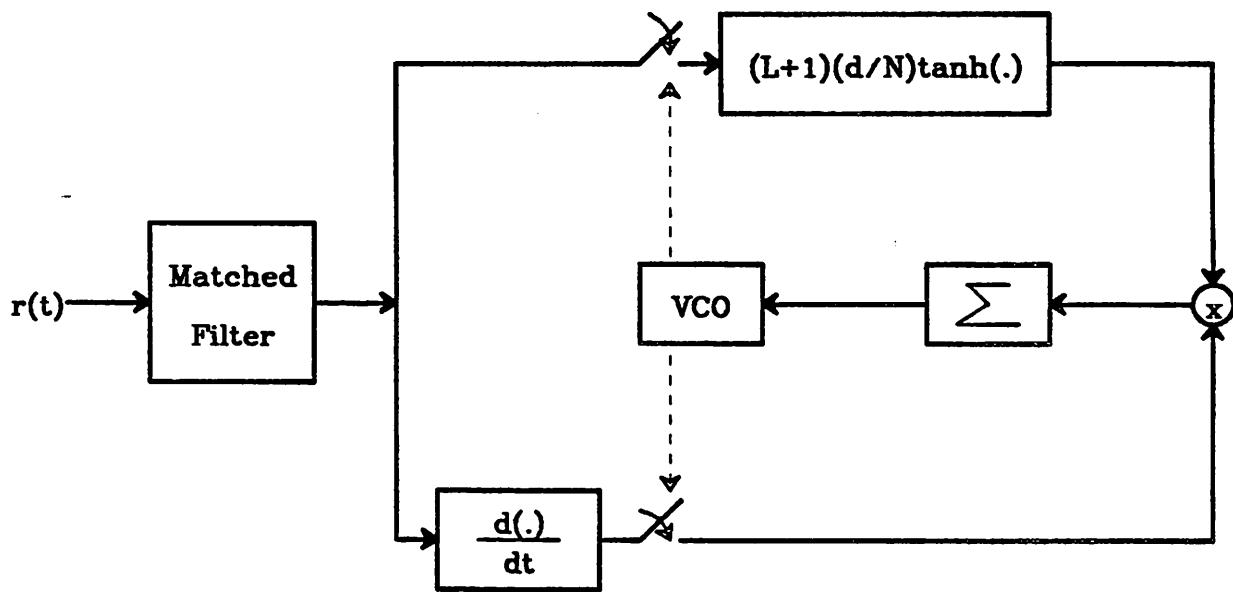


Figure 5.2.1.1. Maximum-likelihood timing recovery scheme.

square-law rectification. The signal shown in Fig. 5.2.1.1.c contains a DC component, a component at the baud rate, and some noise. This can be shown as the following: let  $r(t)$  be the received signal:

$$r(t) = \sum_n a_n h(t - nT)$$

where  $h(t)$  is the effective impulse response. The signal after the square-law rectification  $y(t)$  is given by the expression:

$$\begin{aligned} y(t) &= x^2(t) = \sum_m \sum_n a_m a_n h(t - nT) h(t - mT) \\ &= \sum_n h^2(t - nT) + \sum_n \sum_{m \neq n} a_n a_m \sum_m h(t - nT) h(t - mT) \end{aligned} \quad (5.2.2.1)$$

Define  $z(t)$  to be the expectation of  $y(t)$ , i.e.,

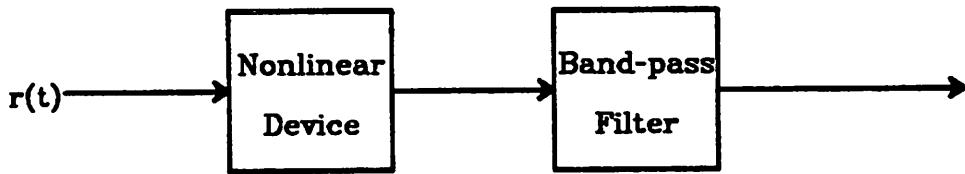


Figure 5.2.2.1. Spectral line method.

$$z(t) \triangleq E[y(t)] = \sum_n h^2(t - nT) \quad (5.2.2.2)$$

Notice that  $z(t)$  is a periodic function and has a Fourier series expansion:

$$z(t) = \sum_k c_k e^{j \frac{2\pi k t}{T}} \quad (5.2.2.3)$$

where

$$\begin{aligned} c_k &= \frac{1}{T} \int_0^T z(t) e^{-j \frac{2\pi k t}{T}} dt \\ &= \frac{1}{T} \int_{-\infty}^{\infty} h^2(u) e^{-j \frac{2\pi k u}{T}} du \end{aligned} \quad (5.2.2.4)$$

Let  $H(f)$  be the Fourier transform of  $h(t)$ , i.e.

$$H(f) = \int_{-\infty}^{\infty} h(t) e^{-j 2\pi f t} dt \quad (5.2.2.5)$$

Then

$$c_k = \int_{-\infty}^{\infty} H(\lambda)H\left(\frac{k}{T} - \lambda\right) d\lambda \quad (5.2.2.6)$$

If the signal is band-limited to  $\frac{1}{T}$ , then  $c_k = 0$  for all  $k \geq 2$ . This implies that the only spectral lines generated are at DC and the baud frequency. Furthermore, if the signal is band-limited to the Nyquist frequency, there will be no timing information available at the baud frequency under square-law rectification. This can be overcome if a higher-order nonlinear device is used.

### 5.2.3. Threshold-crossing phase detector

The threshold-crossing technique detects the zero-crossings of the received signal, and the sampling phase is adjusted to be halfway between two consecutive zero-crossings. Fig. 5.2.3.1 is the block diagram of this scheme. A timing signal is generated each time the received, filtered signal crosses zero. The amplitude of this timing signal is proportional to the difference between the time of occurrence of the zero crossing instant and the nearest sampling phase displaced by half a period. This signal is fed to a LPF, whose output is then used to adjust the VCO.

### 5.2.4. Sampled-derivative detector

The sampled-derivative detector technique is similar to the threshold-crossing technique. During each bit interval, the detector generates an error signal whose magnitude is proportional to the time derivative of the received signal at the sampling instant. The sampling phase is adjusted until the derivative reaches zero. Therefore, the steady-state sampling phase is the peak of the overall system impulse response, regardless whether the impulse response is symmetrical around its peak or not. However, the time derivative alone does not have enough information for adjusting the

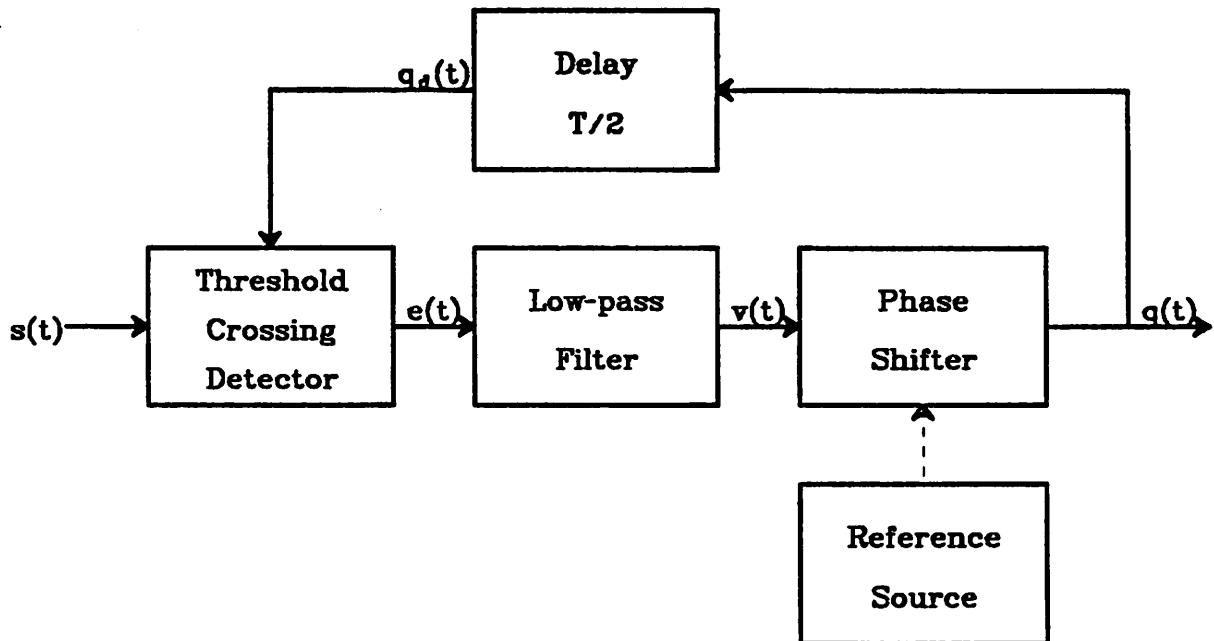


Figure 5.2.3.1. Threshold-crossing timing recovery method.

timing phase since both polarities are possible for the signal at the sampling instant. To include the direction information in the timing signal, the polarity of the signal at the sampling time has to be incorporated. Fig. 5.2.4.1 is the block diagram of this method. As shown in this figure, the error signal is generated by multiplying the derivative of the signal by the decision  $\hat{a}_k$  at the slicer output. Thus, the timing update is given as

$$\Delta\tau_k = \hat{a}_k \left( \frac{\partial r(t)}{\partial t} \right) |_{t=kT + \tau} \quad (5.2.4.1)$$

This error signal is then filtered by a low pass filter and fed to the VCO to adjust the sampling phase. It is clear that this is a decision-directed method. The timing function is:

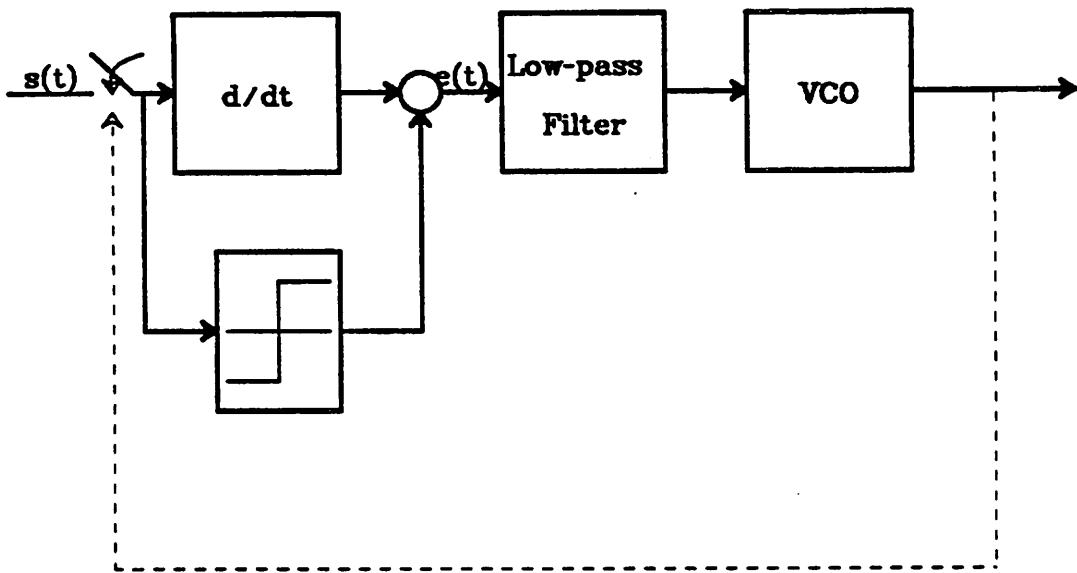


Figure 5.2..4.1. Sampled-derivative timing recovery method.

$$\begin{aligned}
 f(\tau) &= E[a_k \frac{\partial}{\partial \tau} (\sum_m a_m h(kT + \tau - mT))] \\
 &= \frac{\partial h(\tau)}{\partial \tau}
 \end{aligned} \tag{5.2.4.2}$$

It is interesting that, with a matched filter added in the front end, this method becomes an approximation to the maximum-likelihood timing recovery method. Recall that the timing update for the maximum-likelihood method is:

$$\frac{\partial z_n(\tau)}{\partial \tau} \tanh \left[ \frac{(L+1)d}{N_0} z_n(\tau) \right]$$

where  $z(\tau)$  is the output signal of the matched filter. If we approximate  $\tanh \left[ \frac{(L+1)d}{N_0} z_n(\tau) \right]$  by  $\text{sgn}(z_n(\tau)) = \hat{a}_n$ , the estimate of data symbol, the timing update becomes:

$$\frac{\partial z_n(\tau)}{\partial \tau} \hat{a}_n$$

This is exactly the same as the timing update of the sample-derivative method given in Equation (5.2.4.1).

The above approximation is reasonable as long as the SNR is large since

$$\begin{aligned} \tanh(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}} \\ &\approx \begin{cases} 1 & x > 0 \\ -1 & x < 0 \end{cases} \quad \text{if } |x| \gg 1 \\ &= \sin(x) \end{aligned}$$

Another approximation of the maximum-likelihood method will lead us to the commonly known *early-late timing recovery* technique (Fig. 5.2.4.2). This requires the following approximation steps:

Step 1:

$$\begin{aligned} \frac{\partial \Lambda}{\partial \tau} &\rightarrow \frac{\Lambda(\tau + \Delta) - \Lambda(\tau - \Delta)}{2\Delta} \\ &= \sum_k \left\{ \ln \left[ \cosh \left( \left( \frac{kd}{N_0} \right) z_n(\tau + \Delta) \right) \right] - \ln \left[ \cosh \left( \left( \frac{kd}{N_0} \right) z_n(\tau - \Delta) \right) \right] \right\} \quad (5.2.4.3) \end{aligned}$$

This requires that  $\Delta \ll T$ .

Step 2:

$$\cosh(x) \rightarrow \frac{1}{2} e^{|x|}$$

$$\text{where } x = \frac{kd}{N_0} z_n(\tau \pm \Delta).$$

This approximation is reasonable as long as SNR is large since

$$\begin{aligned} \cosh(x) &= \frac{e^x + e^{-x}}{2} \\ &\approx \begin{cases} \frac{1}{2} e^x & x > 0 \\ \frac{1}{2} e^{-x} & x < 0 \end{cases} \quad \text{if } |x| \gg 1 \end{aligned}$$

$$= \frac{1}{2} e^{|x|}$$

After making these two approximations, the timing update becomes the same as for the early-late timing recovery detector:

$$\delta(\tau_n) = \left| \frac{kd}{N_0} z_n(\tau + \Delta) \right| - \left| \frac{kd}{N_0} z_n(\tau - \Delta) \right| \quad (5.2.4.4)$$

### 5.2.5. Modified sampled-derivative detector

The sampling points of threshold-crossing and sampled-derivative methods coincide if the effective channel impulse response is symmetrical around its peak. However, for a general impulse response, neither of them gives the optimum sampling phase. Burton and Saltzberg suggested a modified timing recovery technique which optimizes the performance in the sense that it maximizes the eye opening.

The eye opening  $D(\tau)$  for a two-level signal with impulse response  $h(t)$  is :

$$D(\tau) = h(\tau) - \sum_{k \neq 0} |h(\tau + kT)| \quad (5.2.5.1)$$

For an m-level signal, the eye opening is:

$$D(\tau) = h(\tau) - (m - 1) \sum_{k \neq 0} |h(\tau + kT)| \quad (5.2.5.2)$$

Since the eye opening  $D(\tau)$  is almost always a concave function of  $\tau$ , the maxima of  $D(\tau)$  can be found if the timing phase  $\tau$  is adjusted according to the gradient of  $D(\tau)$ , i.e.:

$$\begin{aligned} f(\tau) &= \frac{\partial D(\tau)}{\partial \tau} \\ &= \frac{\partial h(\tau)}{\partial \tau} - (m - 1) \sum_{k \neq 0} \frac{\partial h(\tau + kT)}{\partial \tau} sgn[h(\tau + kT)] \end{aligned} \quad (5.2.5.3)$$

The first term in the right hand side of Equation (5.2.5.3) is the same as the sample-derivative technique. It drives the sampling point to the peak of the impulse response. The second term forces the sampling phase to give a minimum ISI. The

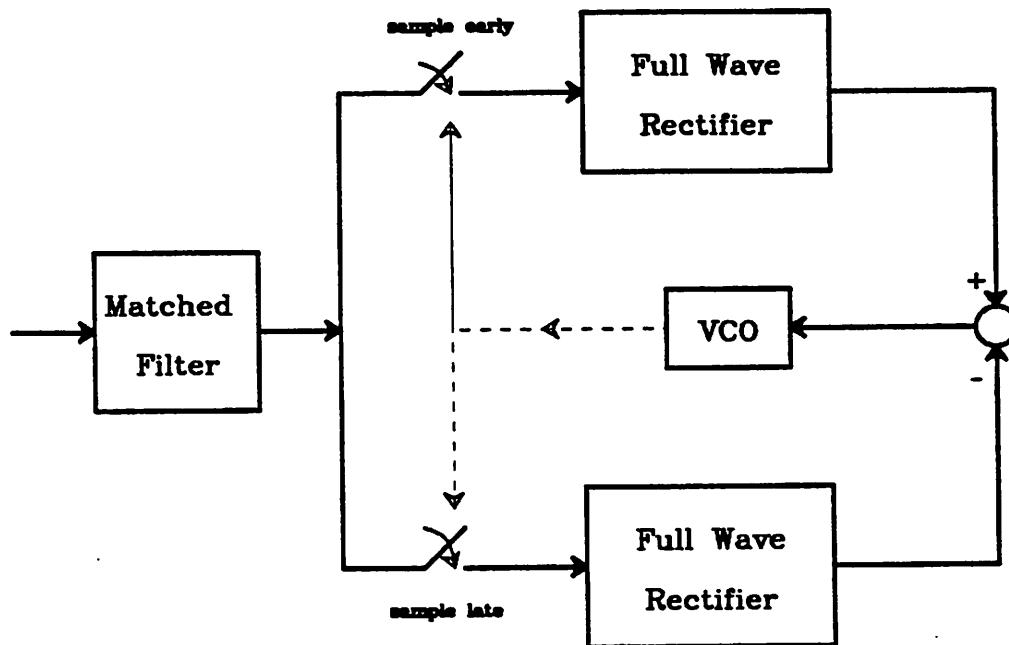


Figure 5.2.4.2. Early-late timing recovery.

resultant steady-state sampling phase is the instant where the eye opening is maximum. The timing update of this modified sampled-derivative method can be found in [26].

### 5.3. Low Sample-rate Timing Recovery Techniques

In this section, we summarize some low sample-rate timing recovery techniques applicable to DSL. The first technique is the wave difference method where at least twice baud-rate sampling is required. Secondly, two techniques that require baud-rate sampling will be discussed.

#### 5.3.1. Wave difference method

The wave difference method (WDM) was first proposed by Suzuki[30]. Agazzi et.al. then generalized this method and gave a detailed analysis [31]. The desired sampling phase for this method is the peak of the received, filtered pulse. At least twice baud-rate sampling is required for this technique. The original form of this method when proposed by Suzuki is a decision-directed technique. It can be described as the following:

Let  $r(t)$  be the received, filtered signal (Fig. 5.3.1.1):

$$r(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT)$$

where  $a_k$  is the data sequence and  $h(t)$  is the overall impulse response. To find the peak of the pulse, the values at  $\frac{1}{4}T$  before and after the main sampling phase  $\tau_k$  are compared. If the pulse is symmetrical with respect to the peak and if the main sampling point is at the peak, the two values  $r(kT + \tau_k - \frac{1}{4}T)$  and  $r(kT + \tau_k + \frac{1}{4}T)$  should be equal. Thus the quantity  $[r(kT + \tau_k + \frac{1}{T}) - r(kT + \tau_k - \frac{1}{T})]$  can be used as a indication of how far the main sampling phase is away from the peak. However,

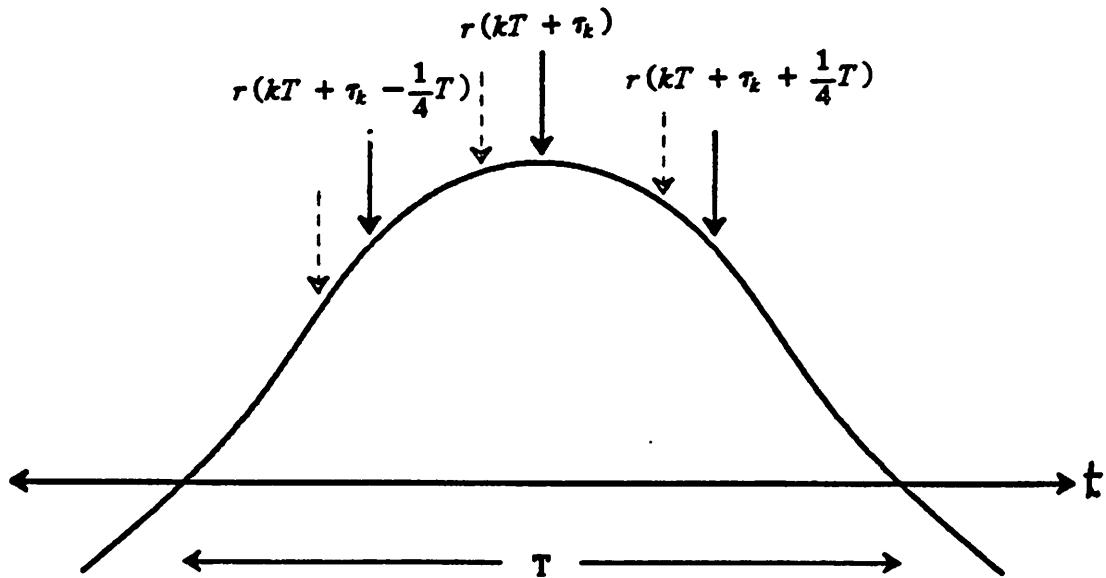


Figure 5.3.1.1. Wave difference method.

this quantity itself cannot be used to adjust the sampling phase since it doesn't have the direction information. To include the direction information, the sign of the pulse must be incorporated. Thus the timing update of this method becomes:

$$\delta(\tau_k) = a_k [r(kT + \tau_k + \frac{1}{4}T) - r(kT + \tau_k - \frac{1}{4}T)] \quad (5.3.1.1)$$

and the timing function is:

$$\begin{aligned}
 f(\tau) &= E[a_k [r(kT + \tau_k + \frac{1}{4}T) - r(kT + \tau_k - \frac{1}{4}T)]] \\
 &= E[a_k (\sum_{l=-\infty}^{\infty} a_l h(kT + \tau + \frac{1}{4}T - lT) - \sum_{m=-\infty}^{\infty} a_m h(kT + \tau - \frac{1}{4}T - mT))] \\
 &= h(\tau + \frac{1}{4}T) - h(\tau - \frac{1}{4}T)
 \end{aligned} \quad (5.3.1.2)$$

It is obvious from Equation (5.3.1.2) that for a channel with a symmetrical impulse

response, the steady-state sampling phase is at the peak.

Very interestingly, if we take the sampled-derivative technique and approximate the time derivative *syntax error file -*, between lines 990 and 990

by  $[r(\tau + \frac{1}{4}T) - r(\tau - \frac{1}{4}T)]$ , we will get exactly (5.3.1.1). Therefore, this version can

be considered as a simplification of the sampled-derivative method. In fact, if the time derivative  $\frac{\partial r(t)}{\partial t}|_{t=\tau}$  is replaced by another approximation  $[r(\tau + T) - r(\tau - T)]$ , this becomes a particular case of the baud-rate sampling timing recovery technique described by Mueller and Muller. This will be discussed in Section 5.3.2.2.

The generalized version of the wave difference method[31] is described as the following:

Define the function  $w(t)$

$$\begin{aligned} w(t) &\triangleq E[g(r(t))] \\ &= E\{g\left[\sum_{k=-\infty}^{\infty} a_k h(t - kT)\right]\} \end{aligned} \quad (5.3.1.3)$$

where  $g(\cdot)$  is a nonlinear function. Since

$$\begin{aligned} w(t + mT) &= E\{g\left[\sum_{k=-\infty}^{\infty} a_k h(t + mT - kT)\right]\} \\ &= E\{g\left[\sum_{l=-\infty}^{\infty} a_k h(t + lT)\right]\} \\ &= w(t) \end{aligned}$$

$w(t)$  is a periodic function with its spectrum consisting of a set of discrete lines at multiples of data rate  $\frac{2\pi}{T}$ . The timing function is chosen to be:

$$f(\tau) = w(\tau - \frac{T}{4}) - w(\tau + \frac{T}{4}) \quad (5.3.1.4)$$

The steady-state sampling phase is the  $\tau$  which gives

$$w(\tau - \frac{T}{4}) = w(\tau + \frac{T}{4})$$

If  $g(x) = x^2$ ,  $w(t)$  becomes:

$$\begin{aligned} w(t) &= E\left\{\left[\sum_{k=-\infty}^{\infty} a_k h(t-kT)\right]^2\right\} \\ &= \sum_{k=-\infty}^{\infty} h^2(t-kT) \\ &= \sum_m c_m e^{j\frac{2\pi m t}{T}} \end{aligned}$$

where  $c_m$  are the coefficients of its Fourier series expansion. If the data signal is band limited to less than  $\frac{2\pi}{T}$ ,  $c_m$  will be equal to zero for any  $m \geq 2$ . The timing function becomes:

$$\begin{aligned} f(\tau) &= w(\tau - \frac{T}{4}) - w(\tau + \frac{T}{4}) \\ &= \sum_m c_m \left[ e^{j\frac{2\pi m (\tau - \frac{T}{4})}{T}} - e^{j\frac{2\pi m (\tau + \frac{T}{4})}{T}} \right] \\ &= \sum_m c_m e^{j\frac{2\pi m \tau}{T}} (-2j \sin(m \frac{\pi}{2})) \\ &= -2j \left[ c_1 e^{j\frac{2\pi \tau}{T}} - c_{-1} e^{-j\frac{2\pi \tau}{T}} \right] \end{aligned}$$

If we decompose the real-valued impulse response  $h(t)$  into an even and an odd functions, i.e.:

$$h(t) = h_e(t) + h_o(t)$$

then, its corresponding frequency response is:

$$H(f) = H_e(f) + jH_o(f)$$

where the even function is transformed into the real part of the frequency response and the odd function mapped into the imaginary part. And

$$H_e(-f) = H_e(f)$$

$$H_o(-f) = -H_o(f)$$

Now, we can express  $c_1$  as:

$$\begin{aligned} c_1 &= \int_{-\infty}^{\infty} \{[H_e(\lambda) + jH_o(\lambda)][H_e(\frac{1}{T}-\lambda) + jH_o(\frac{1}{T}-\lambda)]\} d\lambda \\ &= H^R_1 + jH^I_1 \end{aligned}$$

where

$$\begin{aligned} H^R_1 &= \int_{-\infty}^{\infty} [H_e(\lambda)H_e(\frac{1}{T}-\lambda) - H_o(\lambda)H_o(\frac{1}{T}-\lambda)] d\lambda \\ &= \int_0^{\frac{1}{T}} [H_e(\lambda)H_e(\frac{1}{T}-\lambda) - H_o(\lambda)H_o(\frac{1}{T}-\lambda)] d\lambda \end{aligned}$$

and

$$\begin{aligned} H^I_1 &= \int_{-\infty}^{\infty} [H_o(\lambda)H_e(\frac{1}{T}-\lambda) + H_e(\lambda)H_o(\frac{1}{T}-\lambda)] d\lambda \\ &= \int_0^{\frac{1}{T}} [H_o(\lambda)H_e(\frac{1}{T}-\lambda) + H_e(\lambda)H_o(\frac{1}{T}-\lambda)] d\lambda \end{aligned}$$

The change in the integration limits is due to the fact that the signal is band-limited to  $\frac{1}{T}$  Hz. Similarly,  $c_{-1}$  can be expressed as:

$$\begin{aligned} c_{-1} &= \int_{-\infty}^{\infty} \{[H_e(\lambda) + jH_o(\lambda)][H_e(-\frac{1}{T}-\lambda) + jH_o(-\frac{1}{T}-\lambda)]\} d\lambda \\ &= H^R_{-1} + jH^I_{-1} \end{aligned}$$

where

$$\begin{aligned} H^R_{-1} &= \int_{-\infty}^{\infty} [H_e(\lambda)H_e(-\frac{1}{T}-\lambda) - H_o(\lambda)H_o(-\frac{1}{T}-\lambda)] d\lambda \\ &= \int_{-\frac{1}{T}}^0 [H_e(\lambda)H_e(-\frac{1}{T}-\lambda) - H_o(\lambda)H_o(-\frac{1}{T}-\lambda)] d\lambda \end{aligned}$$

and

$$\begin{aligned}
 H^I_{-1} &= \int_{-\infty}^{\infty} [H_o(\lambda)H_e(-\frac{1}{T}-\lambda) + H_e(\lambda)H_o(-\frac{1}{T}-\lambda)]d\lambda \\
 &= \int_{-\frac{1}{T}}^{0} [H_o(\lambda)H_e(-\frac{1}{T}-\lambda) + H_e(\lambda)H_o(-\frac{1}{T}-\lambda)]d\lambda
 \end{aligned}$$

Notice that

$$\begin{aligned}
 H^R_{-1} &= \int_{-\frac{1}{T}}^{0} [H_e(\lambda)H_e(-\frac{1}{T}-\lambda) - H_o(\lambda)H_o(-\frac{1}{T}-\lambda)]d\lambda \\
 &= \int_0^{\frac{1}{T}} [H_e(f - \frac{1}{T})H_e(-f) - H_o(f - \frac{1}{T})H_o(-f)]df \\
 &= H^R_1
 \end{aligned}$$

$$\triangle H^R$$

and similarly

$$\begin{aligned}
 H^I_1 &= -H^I_{-1} \\
 \triangle H^I
 \end{aligned}$$

Therefore, we have:

$$\begin{aligned}
 f(\tau) &= -2j[(H^R + jH^I)e^{j\frac{2\pi\tau}{T}} - (H^R - jH^I)e^{-j\frac{2\pi\tau}{T}}] \\
 &= 4[H^R \sin(\frac{2\pi\tau}{T}) + H^I \cos(\frac{2\pi\tau}{T})]
 \end{aligned} \tag{5.3.1.5}$$

For a given impulse response, we can solve for the root of Equation (5.3.1.5) to obtain the sampling phase. For a system whose impulse response is symmetrical with respect to its peak,  $H^I$  equals zero, thus, the sampling phase will be at  $\tau = 0$ , i.e., the peak of the impulse response. Notice that this generalized WDM is not decision-directed, as opposed to the original method proposed by Suzuki. And the similarity

between this generalized WDM and the spectrum-line method is obvious[31].

We should be aware that this technique does not work well if the excess bandwidth of the system is small. This is because as the excess bandwidth decreases,  $H^R$  also becomes smaller, thus the gain of the adaptation decreases, and the adaptation becomes ineffective. In fact, as in the Spectral-line method, this technique transforms the power at around half the baud rate to power at the baud rate through the squaring operation. However, this particular problem can be overcome by using a higher-order nonlinear function.

Although only twice baud-rate sampling is required for the WDM, neither of the two samples coincides with the decision sample. Therefore, either higher sampling rate is required, or some interpolation will be needed.

Gardner proposed another timing recovery technique which requires two samples per baud interval [32]. The timing update of this technique is given by the expression:

$$\delta(\tau) = r(\tau + nT - \frac{1}{2}T)[r(\tau + nT) - r(\tau + (n - 1)T)]$$

Fig. 5.3.1.2 is an example of the signal waveform at the receiver assuming that the impulse response is symmetrical with respect to its peak. If there is a transition between two consecutive received signal samples  $r(\tau + nT)$  and  $r(\tau + (n - 1)T)$ , and if the sampling phase is at the peak, the middle point  $r(\tau + nT - \frac{1}{2}T)$  will be zero.

Therefore, the quantity  $r(\tau + nT - \frac{1}{2}T)$  can be considered as a measure of the deviation of the sampling phase from the peak and thus can be used to adjust the sampling phase. The term  $[r(\tau + nT) - r(\tau + (n - 1)T)]$  gives the direction of the phase adjustment. If a transition does not occur,  $r(\tau + nT) = r(\tau + (n - 1)T)$ , no timing adaptation is performed, which is what it should be since  $r(\tau + nT - \frac{1}{2}T)$  now does not correspond to the phase error.

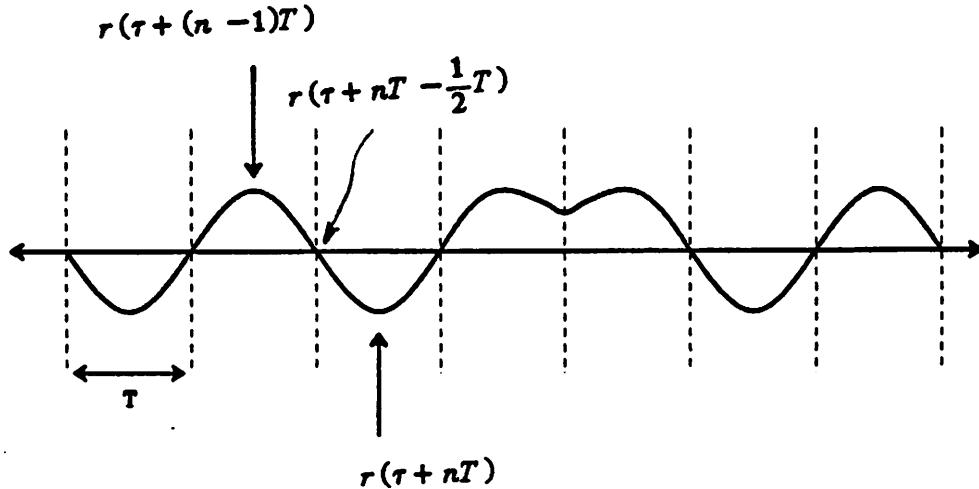


Figure 5.3.1.2.

Notice that the samples  $r(t + nT)$  and  $r(t + (n - 1)T)$  used in the timing recovery coincide with the decision sample. Consequently, only two samples per baud interval are required and no explicit interpolation is needed. Also observe that this method is not decision-directed. However, if we replace  $r(t + nT)$  and  $r(t + (n - 1)T)$  by their signs, i.e., we use

$$r(t + nT - \frac{1}{2}T)[\hat{a}_n - \hat{a}_{n-1}]$$

as the timing update, this method becomes decision directed. The advantage of this modification is that it eliminates the effect of noise in  $r(t + nT)$  and  $r(t + (n - 1)T)$  and no multiplication is required in generating the timing update.

### 5.3.2. Baud-rate sampling

In DSL system, the complexity of the hardware increases almost linearly with the sampling rate. Thus, baud-rate sampling is highly desirable since it is the lower limit of the sampling rate for a data transmission system. However, for a non-minimum bandwidth system, baud-rate sampling will not permit an exact signal reconstruction by any interpolation technique due to the aliasing distortion. Thus, the timing recovery techniques that require a full knowledge of the signal can not be applied. On the other hand, it is theoretically possible to use the conventional techniques for systems with minimum bandwidth since the whole information is still contained in the samples that are baud period apart.

#### 5.3.2.1. Least mean-square timing recovery technique

The least mean-square error timing recovery technique was first proposed by Kobayashi[33]. The essence of this method is to choose a sampling phase which minimizes the mean-square error. This technique is rather complicated in general. Qureshi showed that it can be simplified when applied to partial-response systems with no excess bandwidth[34].

In Fig. 5.3.2.1.1, the received signal  $r(t)$  is :

$$r(t) = \sum_k a_k h(t - kT) + n(t)$$

where  $a_k$  is the data sequence,  $h(t)$  is the effective impulse response, and  $n(t)$  is the additive white noise. Let  $\tau$  be the sampling phase. Define

$$r_k \triangleq r(t)|_{t=kT+\tau}$$

Then the error signal  $e_k$  is

$$e_k = r_k - \sum_{i=0}^{N-1} g_i a_{k-i} \quad (5.3.2.1.1)$$

where  $g_i$  are the coefficients of the desired pulse response. For an ordinary transmis-

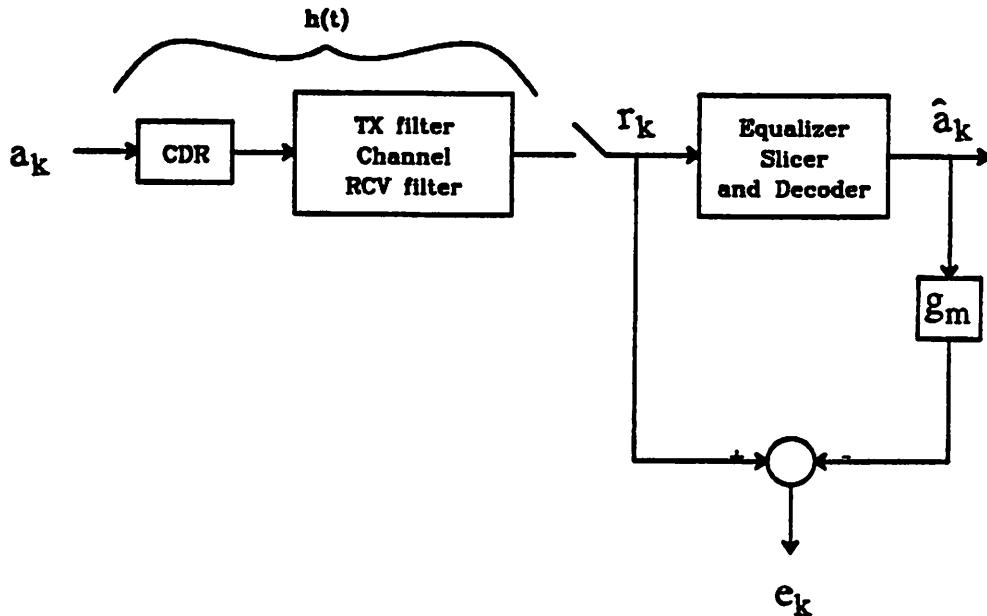


Figure 5.3.2.1.1. Model for deriving least MSE timing recovery method.

sion system.  $g_k = 0$  for any  $k \neq 0$ . For a modified duobinary system.

$$g_k = \begin{cases} 1 & k = 0 \\ -1 & k = 2 \\ 0 & \text{otherwise} \end{cases}$$

In order to minimize the MSE, the sampling phase is adjusted in a direction opposite to the gradient of MSE. The gradient of the MSE is :

$$\frac{\partial E[e_k^2]}{\partial \tau} = 2E \left[ e_k \frac{\partial r}{\partial \tau} \right]$$

Here is the place where the simplification takes place for a minimum-bandwidth partial-response system. If a system is band-limited, the signal  $r(t)$  can be expressed in terms of baud-spaced samples:

$$r(t) = \sum_m r(mT + \tau) \frac{\sin(\frac{\pi}{T}(t - (mT + \tau)))}{\frac{\pi}{T}(t - (mT + \tau))} \quad (5.3.2.1.2)$$

This gives the derivative of  $r(t)$ :

$$\frac{\partial r(t)}{\partial t} = \sum_m r(mT + \tau) \left[ \frac{\cos(\frac{\pi}{T}(t - (mT + \tau)))}{(t - (mT + \tau))} - \frac{\sin(\frac{\pi}{T}(t - (mT + \tau)))}{\frac{\pi}{T}(t - (mT + \tau))^2} \right]$$

Therefore,

$$\frac{\partial r(t)}{\partial t} \Big|_{t=kT+\tau} = \sum_{m, m \neq k} r(mT + \tau) \frac{(-1)^{k-m}}{(k-m)T} \quad (5.3.2.1.3)$$

The message given by the above equation is that for a band-limited system, the derivative at the sampling instants can be obtained by filtering the sampled waveform through an antisymmetrical filter, the coefficients of which decreases as  $\frac{1}{n}$ . Consequently, the driving function is given by:

$$\begin{aligned} \delta(\tau) &= -e_k \frac{\partial r}{\partial \tau} \\ &= -2e_k \left[ \sum_{m, m \neq k} r(mT + \tau) \frac{(-1)^{k-m}}{(k-m)T} \right] \\ &= -2e_k \left[ \sum_{l=1}^M [r_{k-l} - r_{k+l}] \frac{(-1)^l}{lT} \right] \end{aligned}$$

Thus,

$$r_{k+1} = r_k - \beta e_k \left[ \sum_{l=1}^M [r_{k-l} - r_{k+l}] \frac{(-1)^l}{lT} \right]$$

Here, the sum has been truncated to a finite number  $M$ . If we make a first order approximation, i.e., let  $M = 1$ , then

$$\frac{\partial r(t)}{\partial \tau} \sim \frac{1}{T} (r_{k+1} - r_{k-1})$$

and the timing update becomes:

$$\delta(\tau) = \frac{1}{T} e_k (r_{k+1} - r_{k-1})$$

For a modified duobinary system, where

$$e_k = r_k - \sum_{i=0}^{N-1} g_i a_{k-i}$$

$$= r_k - a_k - a_{k-2}$$

the timing function will be:

$$\begin{aligned} f(\tau) &\sim E[e_k (r_{k+1} - r_{k-1})] \\ &= E[(r_k - (a_k - a_{k-2}))(r_{k+1} - r_{k-1})] \\ &= E[r_k r_{k+1}] + E[r_k r_{k-1}] - [(h_1 - h_{-1}) - (h_3 - h_1)] \end{aligned}$$

Since

$$r_k = \sum_m a_m h_{k-m} + n_k$$

$$r_{k+1} = \sum_m a_m h_{k+1-m} + n_{k+1}$$

$$r_{k-1} = \sum_m a_m h_{k-1-m} + n_{k-1}$$

thus

$$E[r_k r_{k+1}] = \frac{1}{2} \sum_m h_m h_{m+1}$$

$$E[r_k r_{k-1}] = \frac{1}{2} \sum_m h_m h_{m-1}$$

Therefore, the timing function would be proportional to:

$$(h_{-1} - h_1) - (h_1 - h_3) \quad (5.3.2.1.4)$$

This timing function is intuitively suitable for MDB. As we can see that for a perfectly equalized MDB pulse, this timing function gives a desired steady-state sampling phase that does not have any ISI. Moreover, in the absence of noise, not only the timing function, but also the timing update is exactly equal to zero when sampling at this steady-state phase. The timing function also has a well-behaved shape that is antisymmetrical to its zero-crossing point. This implies that the timing error will be corrected equally if the timing is shifted to either side of the steady-state sampling phase. Fig. 5.3.2.1.2 is the timing function corresponding to an ideal MDB pulse.

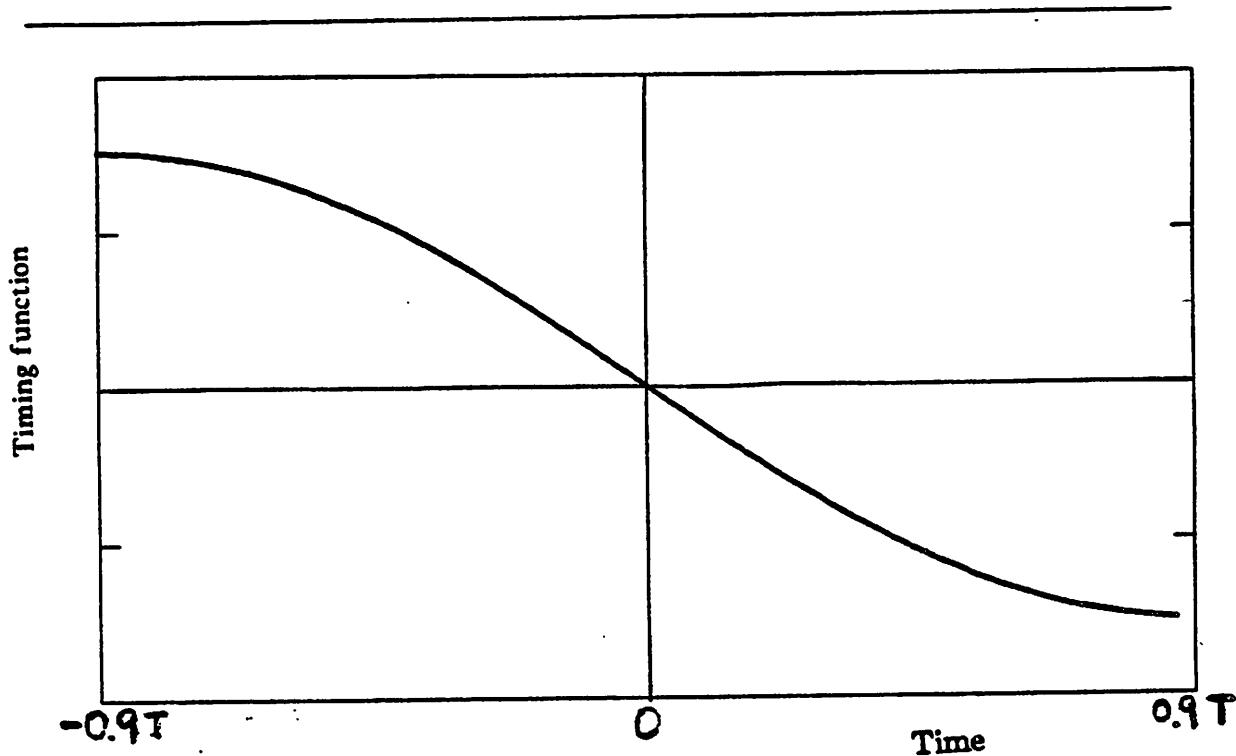


Figure 5.3.2.1.2. Timing function of LMS for an MDB pulse.

However, for some badly distorted pulses, the timing function given by Equation (5.3.2.1.4) may give a bad sampling phase. The steady-state sampling phases and shapes of different timing functions under different line conditions will be compared in the next section.

### 5.3.2.2. Mueller and Muller's timing recovery method

Mueller and Muller proposed a class of baud-rate sampling timing recovery techniques and gave a systematic procedure in constructing the corresponding timing updates[35]. The procedure can be partitioned into two parts: 1) choosing a timing function, and 2) constructing the corresponding timing update.

The timing function is chosen so that:

- (1) It can be expressed as a linear combination of the baud-spaced samples of the impulse response. This requirement is to make possible timing recovery by baud-spaced received-signal samples.
- (2) Its root is close to the optimum of a reasonably chosen performance measure.

In [35], the authors compared two timing functions  $\frac{1}{2}(h_1 - h_{-1})$  and  $h_1$ , where  $h_k \triangleq h(\tau + kT)$ , and concluded that if the impulse response is an even function (a channel with amplitude distortion only), the first scheme gives a better sampling phase, in the sense that it results in smaller ISI. However, for a channel where phase distortion is the main impairment, the second scheme works better.

Besides the two timing functions mentioned in [35], all the functions satisfying criterion (1) and (2) can be used as timing functions. Combining this method with a special line coding and pulse shaping technique, Tzeng proposed a timing recovery method using  $h_{-1} \triangleq h(\tau - T)$  as the timing function[3]. The line code and pulse shape is shown in Fig. 5.3.2..2.1. They are used to ensure a zero crossing at approximately one baud period before the peak of the impulse response. The steady-state sampling phase of this method is at exactly one baud period after the zero-crossing point of the impulse response due to the choice of this particular timing function  $h_{-1}$ . This scheme gives some important advantages. First, since the sampling phase is located at one baud period after the zero-crossing point, the first precursor ISI, which is normally the dominant precursor ISI, is automatically zero. Thus, this sampling phase gives a very small precursor ISI. This makes possible the signal equalization using a decision feedback equalizer only, which is highly desirable from the hardware simplicity point of view. Second, by combining the timing recovery with the decision feedback equalization, the variance of timing jitter is reduced. Third, since the sampling point is determined by the location of the first zero crossing prior to the main lobe of

the impulse response, which is in the leading part of the impulse response, the sampling point is very insensitive to the presence of the bridged taps, whose effects only appear in the trailing part of the impulse response. Consequently, this timing recovery technique gives a stable sampling phase with small jitter variance. However, the particular pulse shaping used results in an increase in the signal power at high frequencies.

Another possible candidate of timing function is:

$$(h_1 - h_{-1}) - 2h_{-1} \quad (5.3.2.2.1)$$

This timing function can be considered as the combination of  $(h_1 - h_{-1})$  and  $h_{-1}$ . The first part is to search for the peak of the impulse response and the second part is to minimize the first precursor ISI. Thus, the combined timing function will give a sampling phase that is a compromise between maximizing  $h(\tau)$ , the amplitude of the impulse response at the sampling instant, and minimizing the first precursor ISI  $h(\tau-T)$ . This timing function has a reasonable antisymmetrical curve around its zero crossing. This implies that the timing errors can be corrected without too much difference if the timing is shifted to either side of the zero-crossing point. In addition, this timing function does not require the use of the particular pulse shaping to ensure a well-behaved zero crossing since the timing phase is not solely determined by the zero-crossing point. Thus, a reasonable sampling point can still be achieved even when the system impulse response has a badly-behaved zero crossing.

The choice of Equation (5.3.2.2.1) as a timing function can also be explained from the point of view of maximizing the eye opening. Recall that one of the criteria for choosing the sampling phase is to maximize the eye opening  $D(\tau)$ :

$$D(\tau) = h(\tau) - 2 \sum_{k=-\infty}^{\infty} |h(\tau + kT)|$$

Since a decision feedback equalizer can be used to eliminate the post-cursor ISI without difficulty, we can define a new eye opening measure  $D'(\tau)$  in which only precursor ISI is considered:

$$D'(\tau) = h(\tau) - 2 \sum_{k < 0} |h(\tau + kT)|$$

We may simplify this eye opening measure further by truncating the above summation to contain only first precursor ISI only, i.e..

$$D'(\tau) = h(\tau) - 2 |h(\tau-T)| \quad (5.3.2.2.2)$$

The sampling phase can be adjusted according to its gradient:

$$\frac{\partial D'(\tau)}{\partial \tau} = \frac{\partial h(\tau)}{\partial \tau} - 2 \frac{\partial |h(\tau-T)|}{\partial \tau} \quad (5.3.2.2.3)$$

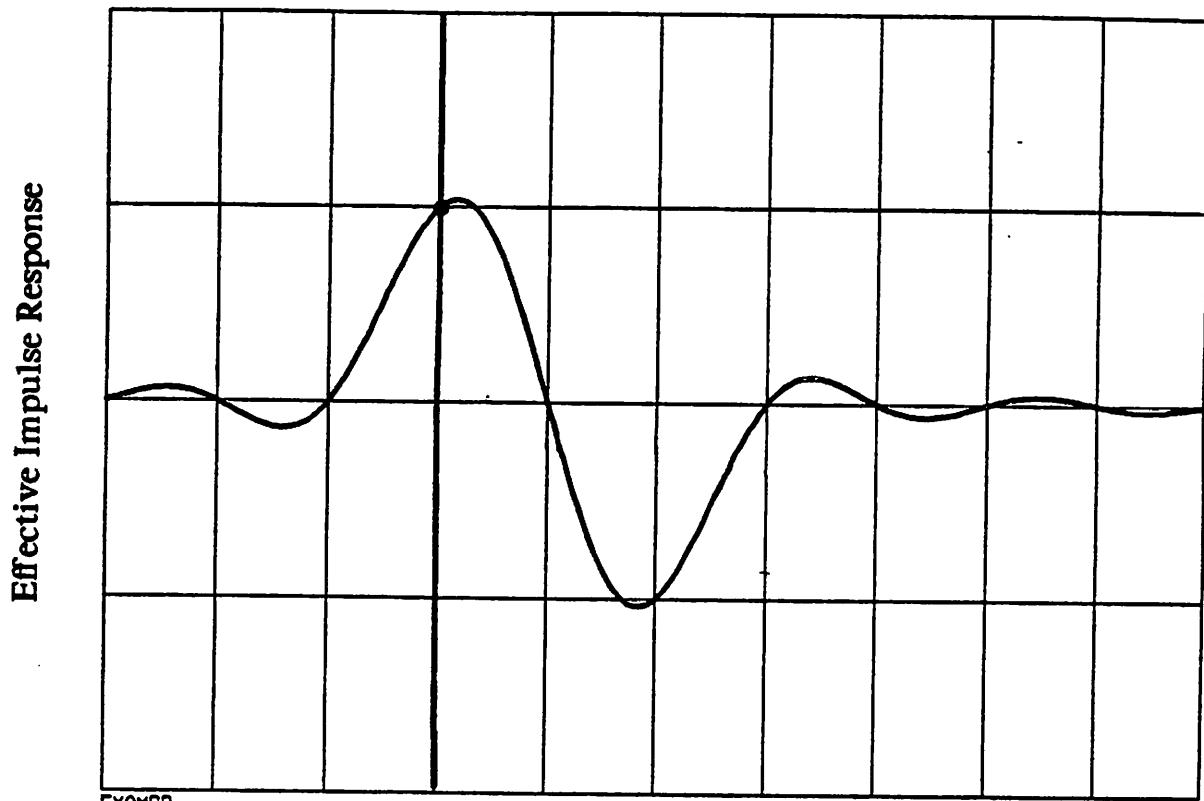
The first order approximation to the first term  $\frac{\partial h(\tau)}{\partial \tau}$  in Equation (5.3.2.2.2) is proportional to  $(h_1 - h_{-1})$ . The second term  $\frac{\partial |h(\tau-T)|}{\partial \tau} = \frac{\partial h(\tau-T)}{\partial \tau} sgn(h(\tau-T))$ , while not being able to be approximated by a simple expression, is to force  $h(\tau-T)$  to zero and therefore can be replaced by  $kh_{-1}$ . Therefore, Equation 5.3.2.2.3 becomes

$$(h_1 - h_{-1}) - k h_{-1} \quad (5.3.2.2.4)$$

where  $k$  should be determined by the impulse response  $h(t)$ . Equation (5.3.2.2.4) is identically the same as (5.3.2.2.1) if  $k = 2$ .

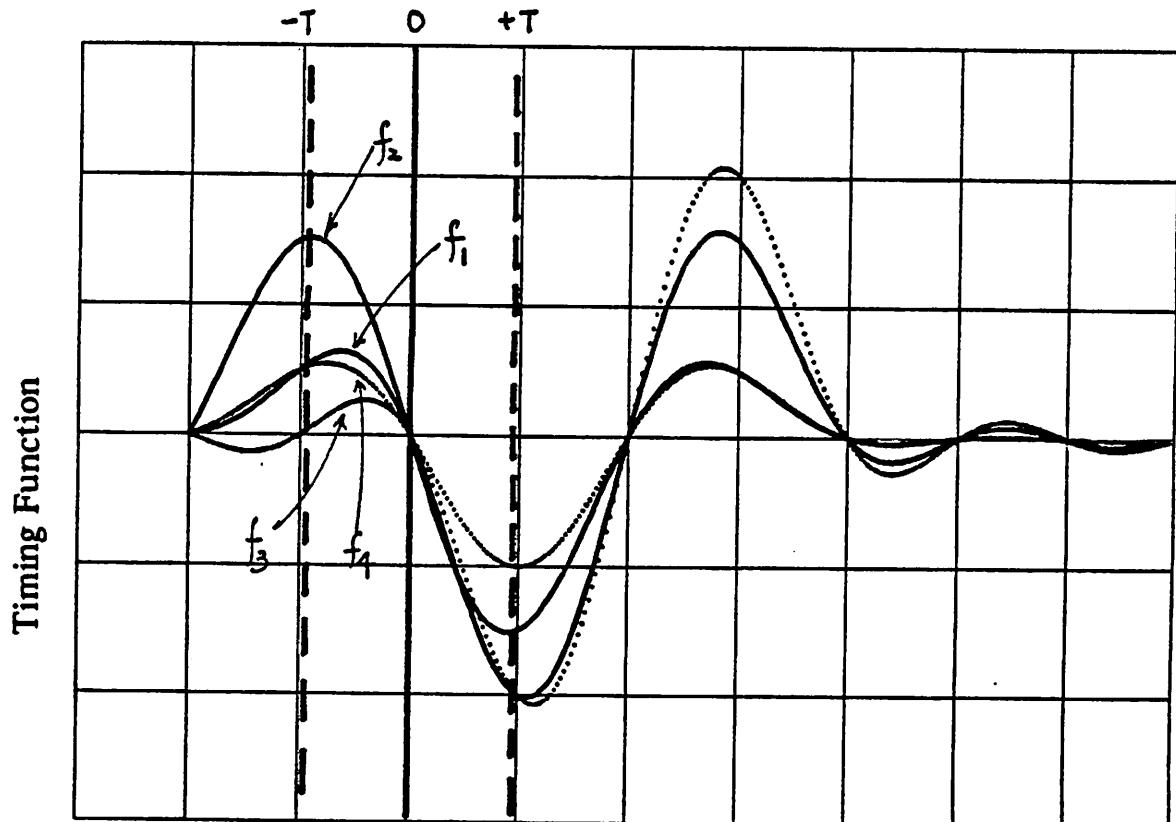
Fig. 5.3.2.2 gives the transfer characteristics of various timing functions under different line and equalization conditions. The four timing functions compared are:  
(1).  $(h_1 - h_{-1}) - 2h_{-1} = h_1 - 3h_{-1}$ . (2).  $(h_1 - h_{-1}) - (h_3 - h_1) = 2h_1 - h_{-1} - h_3$ .  
(3).  $-4h_{-1}$ , and (4)  $h_1 - h_{-1}$ . Condition I is an 18 kf line with an equalizer perfectly matched to the line. Thus, the pulse is a perfect MDB pulse. Under this circumstance, all the above timing functions give the same sampling point that results in zero ISI. In condition II, the same length of cable with two BTs attached is used. This is the same as the worst line considered in Chapter 4. The equalizer is chosen to equalize the 18 kf cable. Due to the existence of the BTs, the equalized pulse is not a perfect raised-cosine pulse and zero ISI is not possible. In Case III, the same line used in Case II is assumed. However, no equalization is used. Thus, the received pulse is a badly distorted pulse. In evaluating the performance of a timing function, one has to examine the characteris-

Condition I



EXAMPLE

$-T \quad 0 \quad +T$  Time



$$f_1 \triangleq h_1 - 3h_{-1}$$

$$f_3 \triangleq -h_{-1}$$

Time

$$f_2 \triangleq 2h_1 - h_{-1} - h_3 \quad f_4 \triangleq h_1 - h_{-1}$$

Figure 5.3.2.2.

Condition II

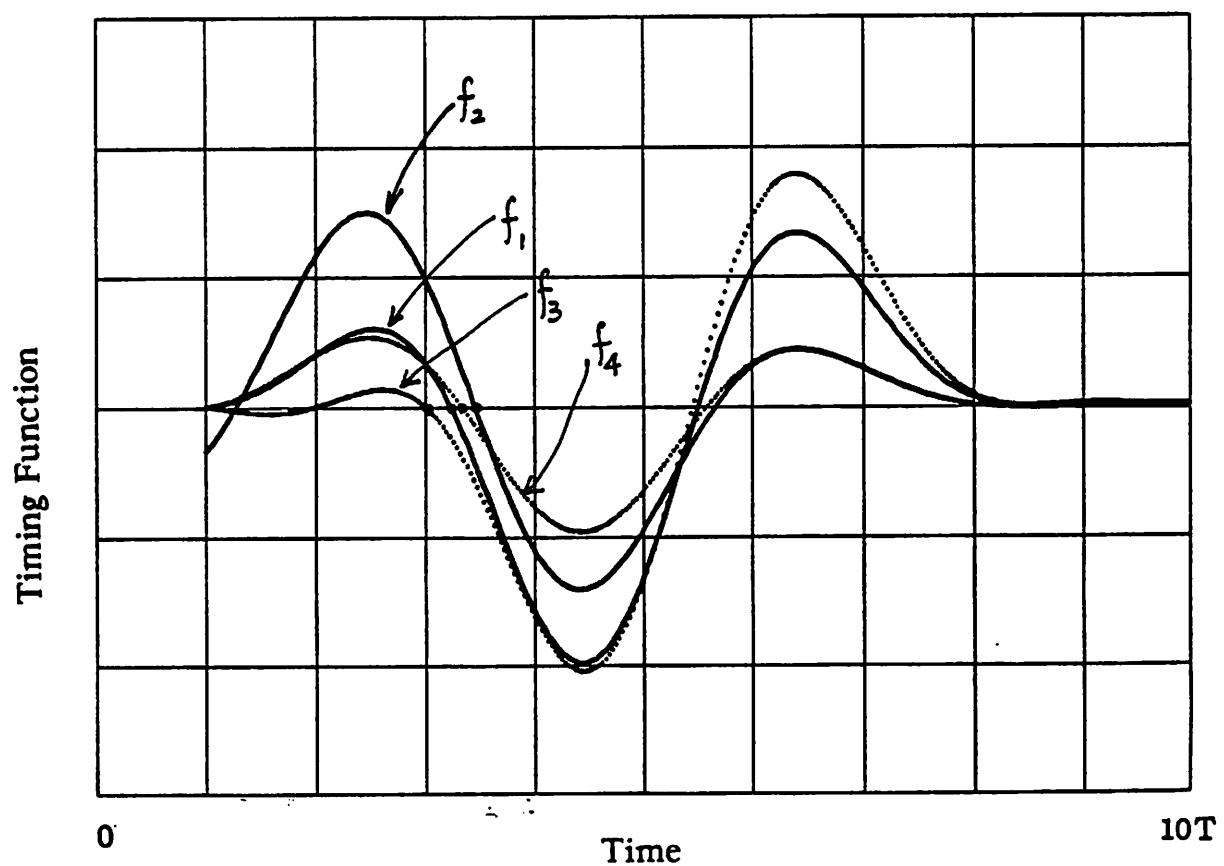
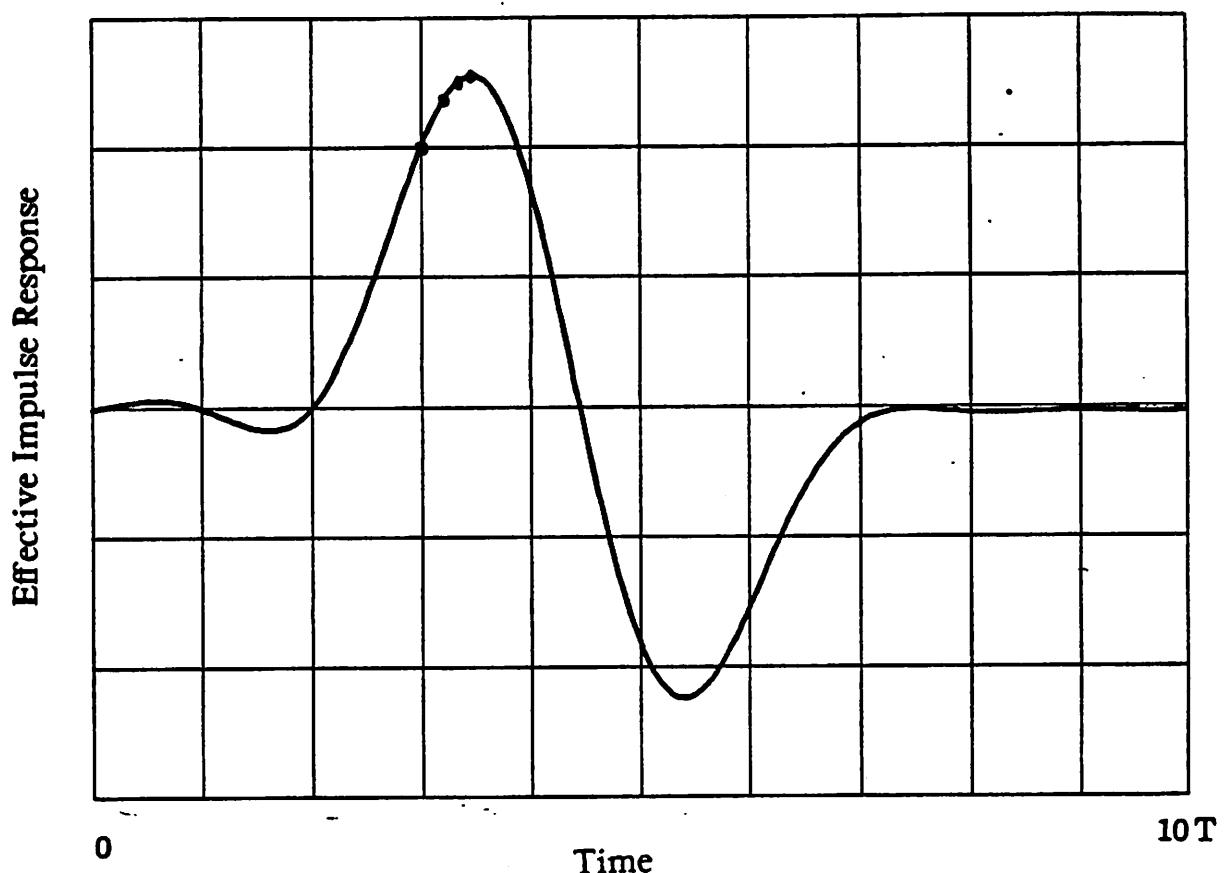


Figure 5.3.2.2.

### Condition III

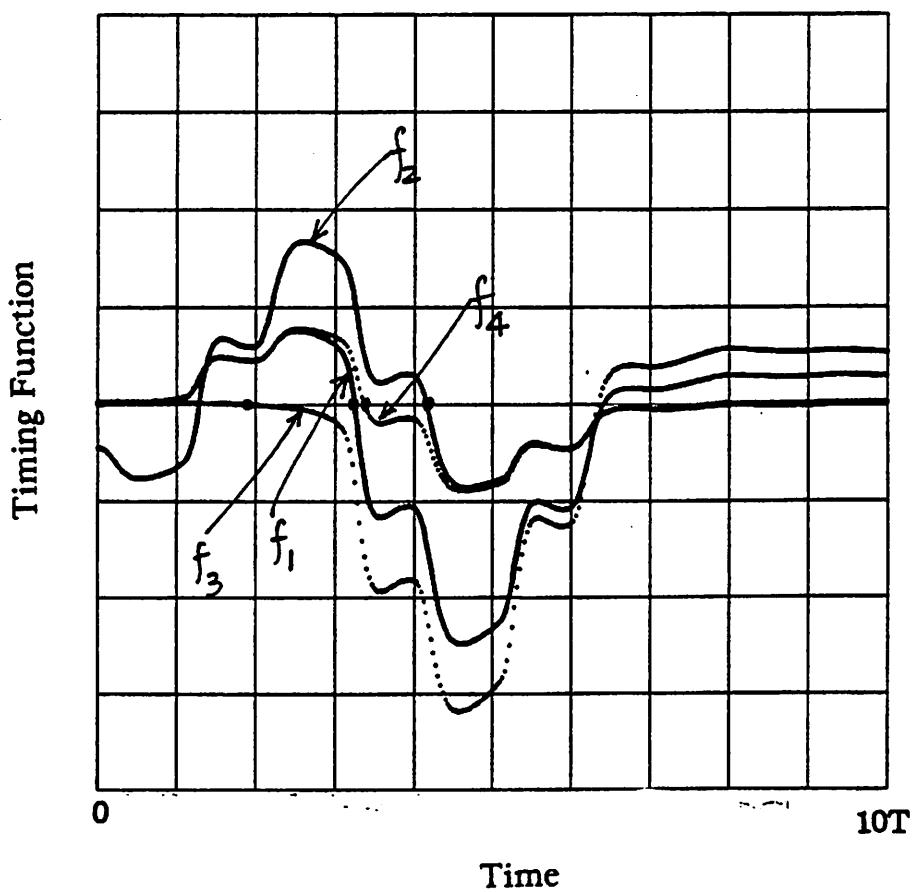
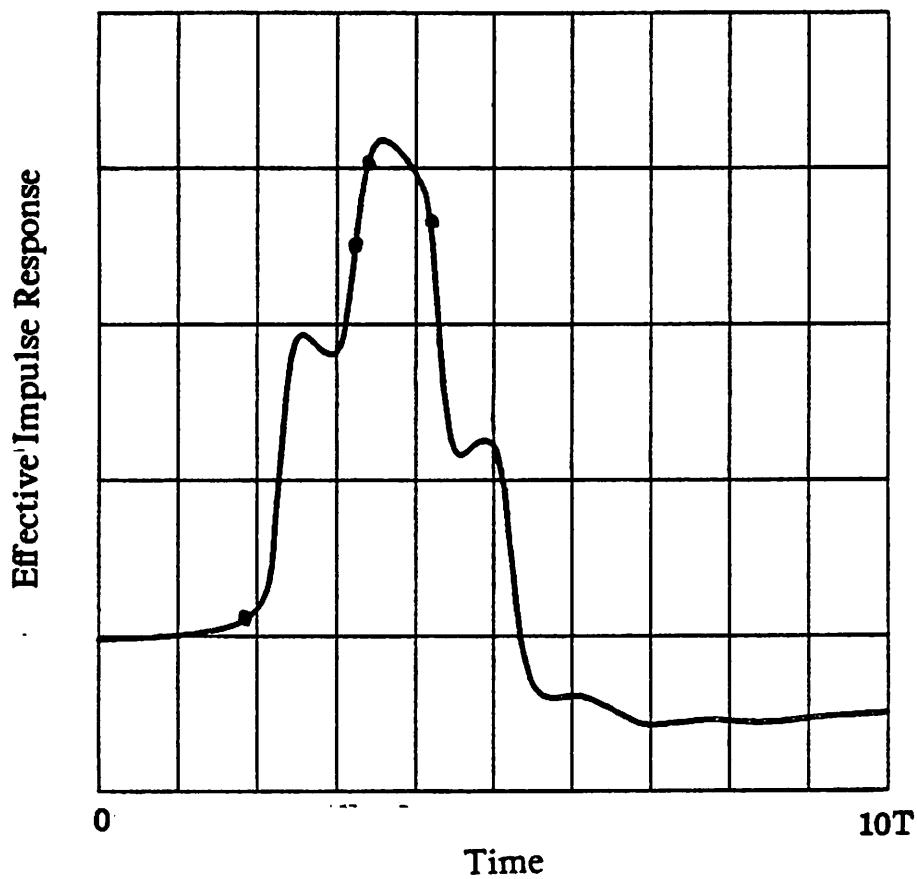


Figure 5.3.2.2.

tic of the timing function at the zero-crossing point as well as the shape of the timing function around the zero crossing point. As can be seen from Fig. 5.3.2.2, both  $h_1 - h_{-1}$  and  $2h_1 - h_{-1} - h_3$  have in general better antisymmetrical shapes around the zero crossing points than  $h_{-1}$  and  $h_1 - 3h_{-1}$  have. The characteristics of the timing functions at the zero-crossing point under these three cases are summarized in Table 5.3.2.2.1, where  $h_0 \triangleq h(\tau)$  is the magnitude of the impulse response at the sampling point, and  $h_0 - 2|h_{-1}|$  is the truncated eye opening measure  $D'(\tau)$  defined in Equation 5.3.2.2 and is considered to be a more important performance measure than  $h_0$ . Under the first condition, all the timing functions have the same  $h(\tau)$  and  $D'(\tau)$ . This is not surprising, since the equalized pulse is a raised-cosine MDB pulse. In the second case, although the timing function  $2h_1 - h_{-1} - h_3$  has the largest  $h(\tau)$ ,  $h_{-1}$  gives the widest eye opening  $D'(\tau)$ , and  $h_1 - 3h_{-1}$  has a slightly smaller  $D'(\tau)$  than  $h_{-1}$  has. In the third case, only  $h_1 - 3h_{-1}$  gives an acceptable  $D'(\tau)$ .

The second part of Mueller and Muller's technique is to find the timing update, an unbiased, minimum-variance estimate of the timing function, as a function of past decisions and past received signal samples. In fact, the timing update is to be expressed as a inner product of the two vectors, the weighting vector  $\tilde{g}_k$  and a vector of past sample values  $\tilde{r}_k$ , i.e.,

	Performance measure	Timing function			
		$h_1 - 3h_{-1}$	$2h_1 - h_{-1} - h_3$	$-h_{-1}$	$h_1 - h_{-1}$
Condition I	$h_0$	1.00	1.00	1.00	1.00
	$h_0 - 2 h_{-1} $	1.00	1.00	1.00	1.00
Condition II	$h_0$	9.41E-01	9.98E-01	7.93E-01	9.82E-01
	$h_0 - 2 h_{-1} $	7.12E-01	4.31E-01	7.93E-01	6.05E-01
Condition III	$h_0$	7.99E-01	8.74E-01	3.80E-02	9.37E-01
	$h_0 - 2 h_{-1} $	3.13E-01	negative	3.80E-02	negative

Table 5.3.2.2.1.

$$\delta = \tilde{g}_k^T \tilde{r}_k$$

The weighting vector is a vector whose entities are functions of past decisions:

$$\tilde{g}_k = \begin{bmatrix} g_1(a_{k-m+1}, \dots, a_k) \\ g_2(a_{k-m+1}, \dots, a_k) \\ \vdots \\ g_m(a_{k-m+1}, \dots, a_k) \end{bmatrix}$$

and the received signal vector is:

$$\tilde{r}_k^T = (r_{k-m}, r_{k-m+1}, \dots, r_k)$$

Reference[ 35] gives a systematic approach of finding the timing update given the timing function.

It is interesting to examine the timing update for the particular timing function  $f(\tau) = \frac{1}{2}(h_1 - h_{-1})$ . Given in [35] , the timing update corresponding to this timing function is

$$a_k r(\tau + (k-1)T) - a_{k-1} r(\tau + kT)$$

Thus,

$$\tau_{k+1} = \tau_k + \beta(a_k r(\tau + (k-1)T) - a_{k-1} r(\tau + kT))$$

or

$$\tau_{k+1} = \tau_0 + \beta \sum_i a_i (r_{i+1} - r_{i-1})$$

This suggests that the timing update can be rewritten as

$$a_k (r_{k+1} - r_{k-1})$$

Notice that if we interpret  $(r_{k+1} - r_{k-1})$  as the first order approximation of  $\frac{\partial r(t)}{\partial t}|_{t=\tau}$  as we did in the LMS technique. this scheme becomes the first order approximation of the sampled-derivative technique. This is similar to the WDM proposed by Suzuki where the derivative in the sampled-derivative method is approximated by the difference of two points each a quarter-period apart from the main

Timing function	Weighting vector
$h_1 - h_{-1}$	$-a_k$ $a_{k-1}$
$h_1$	$-a_{k-1} - 2a_k a_{k-1} a_{k-2}$ $a_k + 2a_{k-2}$ $a_{k-1} - a_k a_{k-1} a_{k-2}$
$h_{-1}$	$a_{k-1} - a_k a_{k-1} a_{k-2}$ $a_{k-2} + 2a_k$ $-a_{k-1} - 2a_k a_{k-1} a_{k-2}$
$h_1 - 3h_{-1}$	$-2a_{k-1} + 4a_k a_{k-1} a_{k-2}$ $-2a_{k-2} - 7a_k$ $7a_{k-1} + 4a_k a_{k-1} a_{k-2}$

Table 5.3.2.2.2.

sampling point.

Listed in table 5.3.2.2.2 are the weighting vectors corresponding to some timing functions.

## CHAPTER 6

### Timing Jitter

Hybrid-mode full-duplex digital data transmission on ordinary twisted-pair cables at data rates up to 160 kb/s is desired for implementation of digital subscriber loops. The performance of the echo canceller in a hybrid-mode system is degraded by jitter in the timing signal recovered from the line. Analysis shows that to achieve 60 dB of echo suppression requires that the magnitude of the jitter be smaller than 0.1%, a difficult requirement for any design. Timing signal recovery employing a digital phase-locked loop (PLL) is desired for simplicity of MOSLSI implementation. For a 160 kb/s system in straightforward implementation, a digital PLL would require a local oscillator running at the impractically high speed of 160 Mb/s to limit the timing jitter to 0.1%.

In addition to the desire of using digital PLL, our design is aimed at the baud-rate sampling timing recovery technique for the same reason of the hardware simplicity consideration. However, the use of the baud-rate sampling results in a more difficult echo estimation, which further complicates the timing jitter problem.

In this chapter, we show that an *echo interpolation technique*, used in conjunction with *modified duobinary partial response coding (PRC)*, can greatly relax the requirement on the magnitude of the maximum jitter and, as a result, facilitate the design of the digital phase-locked loop.

#### 6.1. Digital phase-locked loop

Digital phase-locked loop (PLL) has always been considered impractical for the DSL application since it produces discrete phase jumps that results in the degradation

of the echo cancellation, despite of the fact that digital PLL has the desired property of having much lower implementation complexity than its analog counterpart. However, due to the need of the single chip realization of the DSL system, hardware simplicity becomes critically important. Therefore, in this chapter, we develop a technique which makes possible the use of the digital PLL in DSL.

The basic structure of a digital PLL is very similar to an analog PLL. (Fig. 6.1.1.a and 6.1.1.b give the block diagram of the analog and the digital PLL respectively.) In the steady-state, the analog PLL can be treated as a closed loop feedback system consisting of a phase comparator, a loop filter, and a voltage controlled oscillator (VCO). The phase comparator compares the phase of the incoming signal with that of a reference signal, and generates an output that is proportional to their difference. Due to the presence of noise in the incoming signal, the output of the phase comparator also contains the noise in addition to the phase error signal. The loop filter is a critical part of the PLL system and is used to pass the phase error signal and filter out the noise. The design of the loop filter is a trade-off between the phase tracking ability and the susceptibility of noise penetration. A wider loop filter bandwidth corresponds to a wider tracking bandwidth, which enables the system to track the high frequency phase variations. On the other hand, the system will be less immune to the noise since the noise bandwidth also increases as the bandwidth of the loop filter increased. This increases the variance of the timing jitter. The VCO gives an output whose frequency change is proportional to its input, which is the loop filter output. Thus, the phase change of the VCO output is proportional to the integral of its input. Therefore, the VCO is equivalent to an integrator.

There are several different kinds of digital PLL[ 36]. The digital PLL referred here is the *all digital PLL* (or ADPLL) discussed in [36]. The output of the phase comparator is first quantized into a binary signal representing either phase lead or lag of the

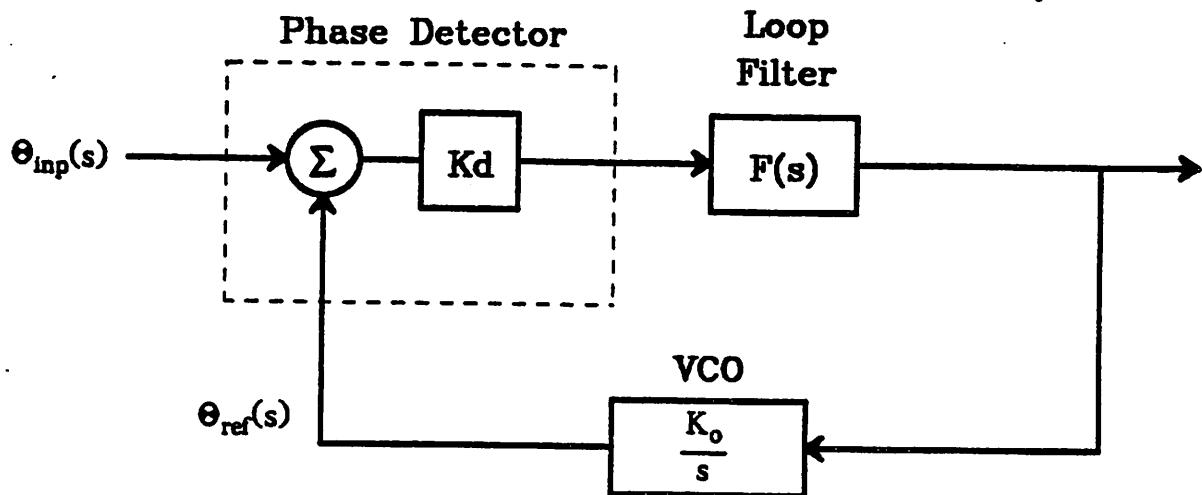


Figure 6.1.1.a. Analog phase-locked loop.

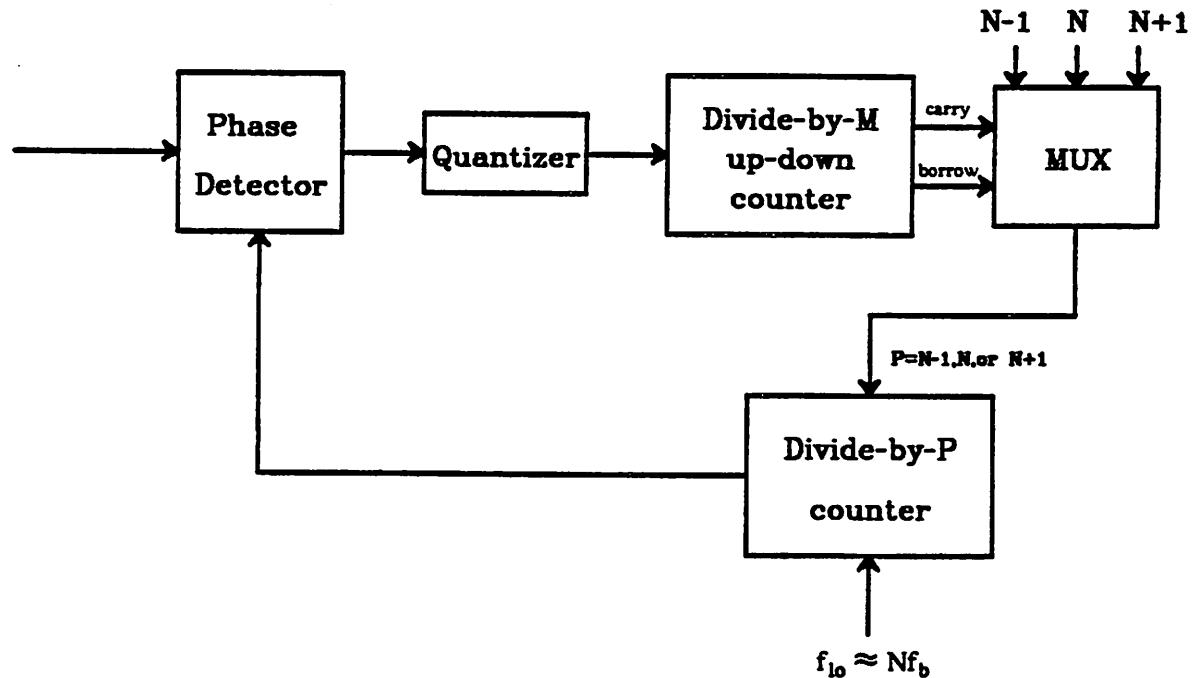


Figure 6.1.1.b. Digital phase-locked loop.

incoming signal relative to the reference signal (e.g., 1 for phase lead and 0 for phase lag). This binary signal is sent to a divide-by- $M$  up-down counter to cause the counter to count up or count down (for example, 1 causes the counter to count up and 0 causes it to count down). Once the counter overflows (or underflows), the carry (or borrow) signal of the counter will become active. The carry and the borrow signals of the up-down counter are used to select one among three numbers,  $N$ ,  $N + 1$ ,  $N - 1$ , to be used in the digital VCO. This divide-by- $M$  up-down counter performs the function of the loop filter. Roughly, it can be considered as an integrator. Thus, this type of digital PLL is equivalent to a second order PLL. The number  $M$  determines the bandwidth of the filter (or the time constant of the integrator). A large  $M$  corresponds to a small noise bandwidth since the noise component will most likely be averaged out before the loop filter generates a carry or a borrow signal. On the other hand, large  $M$  implies a small tracking bandwidth since the loop filter will take more time to generate carry or borrow signals and therefore cannot track high-frequency phase signals. An additional requirement is that  $M$  should be larger than the number of periods that is covered by the echo canceller. This ensures that after a phase jump occurs, the system will come back to the steady state before a second phase jump could happen. This eases the system control sequence. The above requirement does not impose serious limitation on  $M$  since  $M$  is normally chosen to be greater than 20 while the number of periods covered by the echo canceller is normally less than 20.

The digital VCO is a divide-by- $P$  up counter, where  $P$  can be  $N - 1$ ,  $N$ , or  $N + 1$  depending on the carry and the borrow signals of the loop filter. It is driven by a local oscillator running at  $f_{lo} \approx Nf_b$ , where  $f_b$  is the baud frequency and is equal to the frequency of the master clock in the central office. In a normal mode of operation, where neither the carry nor the borrow signal of the loop filter is active, the digital VCO generates one output pulse after receiving  $N$  pulses from the local oscillator. Therefore, the spacing between two consecutive VCO output pulses is  $\frac{N}{f_{lo}}$ . Thus, the

output frequency is normally  $\frac{f_{lo}}{N} \approx f_b$ . If the carry or the borrow signal of the loop filter should go active, the digital VCO becomes a divide-by-( $N + 1$ ) or divide-by-( $N - 1$ ) counter, i.e., it produces one pulse every ( $N + 1$ ) or ( $N - 1$ ) local oscillator pulses. Therefore, the spacing between two consecutive output pulses becomes  $\frac{N+1}{f_{lo}} = \frac{N}{f_{lo}} + \frac{1}{f_{lo}}$  or  $\frac{N-1}{f_{lo}} = \frac{N}{f_{lo}} - \frac{1}{f_{lo}}$ . This particular arrangement results in the constraint that the magnitude of the phase jump is fixed and is equal to  $\frac{1}{f_{lo}}$ . The property of the digital PLL gives an important advantage in solving the timing jitter problem, as will be seen later in this chapter. Notice that the magnitude of the phase jump is inversely proportional to the frequency of the local oscillator. The higher the local oscillator frequency is, the smaller the magnitude of the phase jump will be. Thus, a straightforward way of reducing the magnitude of timing jitter is to increase the local oscillator frequency.

The output frequency of the digital VCO falls within a certain range. The highest frequency is achieved when the phase comparator gives 0 (phase lag) all the time. In such a case, the output frequency is

$$\left\{ \frac{1}{M} [(N - 1) \frac{1}{f_{lo}} + (M - 1)N \frac{1}{f_{lo}}] \right\}^{-1} = \frac{M}{MN - 1} f_{lo} \quad (6.1.1)$$

If the phase comparator gives signal 1 (phase lead) continuously, the output frequency will be minimum and is equal to

$$\left\{ \frac{1}{M} [(N + 1) \frac{1}{f_{lo}} + (M - 1)N \frac{1}{f_{lo}}] \right\}^{-1} = \frac{M}{MN + 1} f_{lo} \quad (6.1.2)$$

These two equations set a limit on the quantity by which the center frequency of the local oscillator can be deviated from  $Nf_b$ . From Equations (6.1.1) and (6.1.2), we have

$$\frac{MN - 1}{M} f_b \leq f_{lo} \leq \frac{MN + 1}{M} f_b \quad (6.1.3)$$

Thus

$$(1 - \frac{1}{MN})Nf_b \leq f_{lo} \leq (1 + \frac{1}{MN})Nf_b \quad (6.1.4)$$

If  $M = 32$  and  $N = 64$ , the center frequency of the local oscillator must be within 0.05% of  $N$  times of the master clock of the central office. Since crystals with accuracy of 50 ppm are readily available at reasonable cost, this does not present a problem.

## 6.2. PROBLEM AND SOLUTION FORMULATION

Only discrete-time techniques, which are considered to be more feasible than continuous-time techniques in integrated-circuit implementation, are considered here. Fig. 6.2.1 is the block diagram of the subscriber end. The echo path can be modeled as

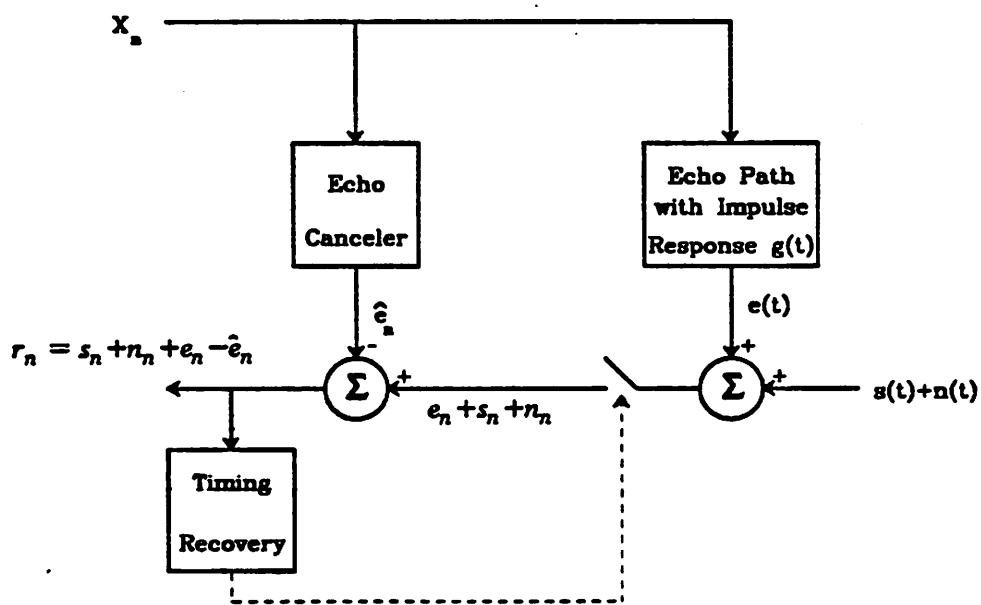


Figure 6.2.1. Block diagram of the subscriber end of DSL.

a linear system with impulse response  $g(t)$ , the resultant sampled echo signal  $e_n$  will be the convolution of the near end input data  $a_n$  with the sampled impulse-response of echo path. i.e.,

$$e_n = \sum_{k=0}^{N-1} g_k a_{n-k}$$

where  $g_k \triangleq g(kT + \tau)$ , and  $\tau$  is the sampling phase.

One very important feature accompanied with the discretization is that only the sampled echo signal is important and it depends on the echo path impulse response at the sampling points only. Therefore, different sampling phases will give different echo signals. The linear echo canceller is an adaptive transversal filter. The output of echo canceller, echo replica, is:

$$\hat{e}_n = \sum_{k=0}^{M-1} c_k a_{n-k}$$

where  $c_k$ 's are the coefficients of echo canceller, and  $M$  is the number of coefficients. The coefficients are adapted by a feedback loop to match the values of the sampling points of echo path impulse response. If the number of coefficients is large enough and if the coefficients  $c_k$ 's have been adapted to  $g_k$ 's, the desired cancellation can be achieved.

### 6.2.1. Problem formulation

When jitter occurs, the output of the digital phase locked loop will have a discrete phase jump. The jump in phase will result in a different echo signal because of the change in  $g_k$ 's. Since the coefficients of echo canceller can not adapt simultaneously to be equal to the new  $g_k$ 's, the predicted echo signal, i.e., echo replica, will be different from the real echo. Therefore, the desired cancellation can not be achieved and the system performance degrades. Fig. 6.2.1.1 shows the jitter performance of an ordinary AMI system.

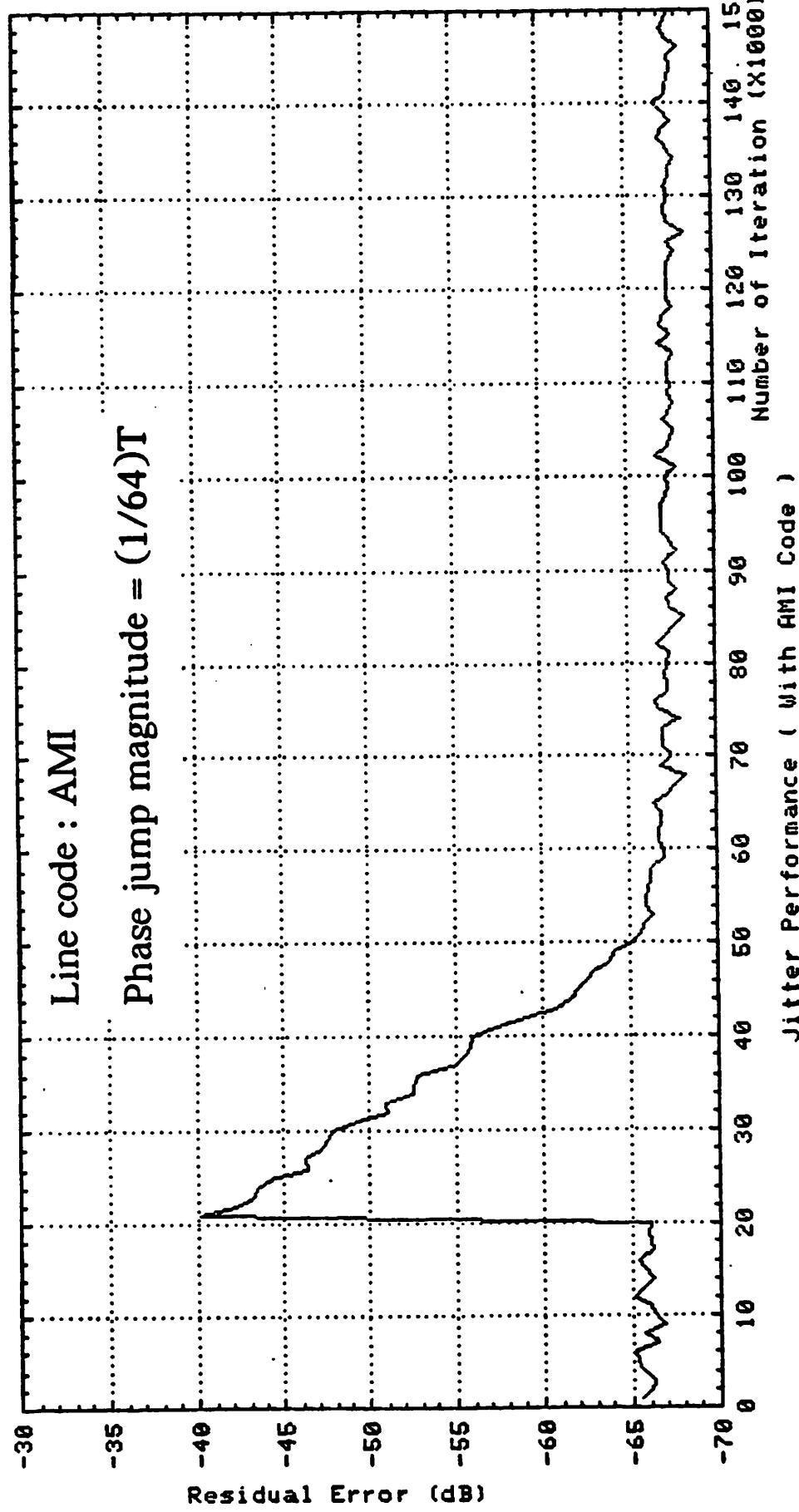


Figure 6.2.1.1. Jitter performance of an AMI system.

In order to reduce the degradation in performance in the presence of jitter (phase jump), the values of the echo path impulse response at new sampling phases have to be obtained. In other words, the new coefficients of echo canceller have to be calculated when jitter occurs. This can be achieved by first reconstructing the continuous waveform of the echo path impulse response and then calculating the new sample values according to the magnitude of the phase jump. The data available for reconstructing the continuous waveform is the echo canceller coefficients, which are equal to the values of the echo path impulse response at old sampling phase spaced from one another by one sampling period.

Two major difficulties have to be overcome before this technique is applicable.

The first difficulty is the aliasing distortion due to *baud-rate sampling*. The echo canceller is the most complex portion of digital subscriber loops. Its hardware complexity increases linearly with the sampling rate due to the discrete-time nature of the system. Therefore, a minimum sampling rate is desirable. Baud rate is the lower limit of the sampling rate for a digital data transmission system and baud-rate sampling[3] has been shown to be a feasible technique. Therefore, the design considerations should be aimed at baud-rate sampling. But because of the baud-rate sampling, the data available for reconstructing the continuous waveform of echo path impulse response is not enough according to Nyquist theorem. There will be aliasing distortion introduced in the process of reconstructing the continuous waveform. In other words, the calculated sample values won't be equal to the real ones, and, as a result, there will be errors in predicting the new echo signal. Fig. 6.2.1.2 is the frequency response of echo signal before sampling. The power at frequencies higher than half baud rate will contribute to aliasing distortion.

The second problem is the distortion introduced by non-ideal low pass filtering. A low-pass filter is needed in order to reconstruct continuous waveform from discrete

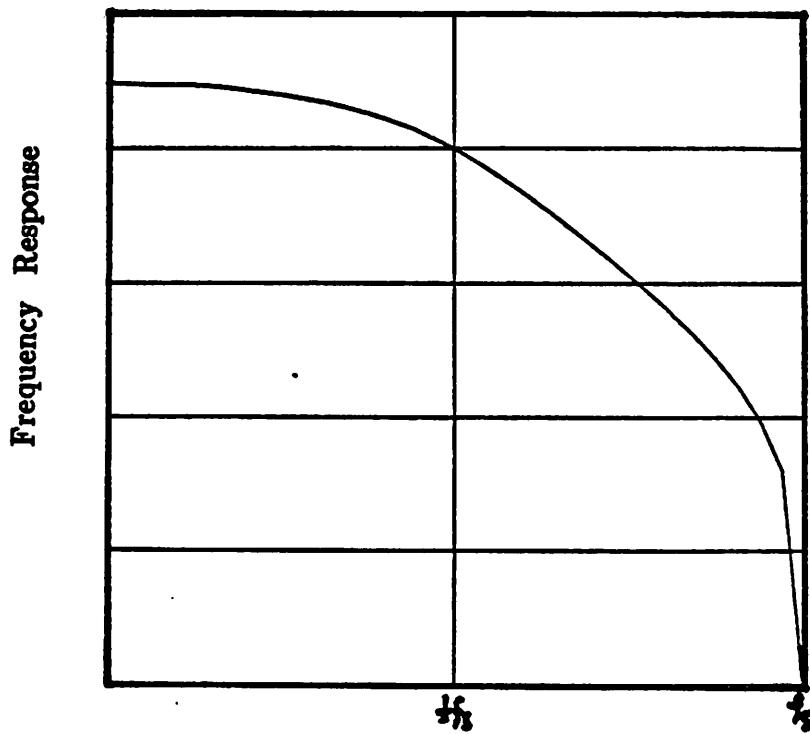


Figure 6.2.1.2. Frequency response of echo path (without coding).

sampled data. It is not practical to have a high-order filter on board to perform the function of low pass filtering. The technique studied in this paper is *interpolation*. This is discussed in detail in Section 3.4.

### 6.2.2. Partial Response Coding (PRC)

The proposed solution for the first problem is to shape the echo signal spectrum by using partial response coding (PRC).

As discussed in Chapter 3, the basic idea of PRC is to combine successive binary bits together by some given rules. It introduces intersymbol interference in a controlled manner. As a result, the autocorrelation of the input sequence and the signal power spectrum are changed accordingly. The requirement of reducing the aliasing dis-

tortion can be met if the echo signal spectrum is shaped in such a way that the power in the range between half the baud rate up to one baud rate is reduced. One partial response code which satisfies this criterion is the *modified duobinary (MDB) code* (class IV, n=3). It has a system polynomial of  $(1 - D)^2$  and has zeros at DC and half baud rate ( $1/2 f_b$ ). The transfer function of this scheme is:

$$H(\omega) = 1 - e^{-2j\omega T} = 2je^{-j\omega T} \sin(\omega T)$$

The magnitude of the transfer function is:

$$|H(\omega)| = 2|\sin(\omega T)|$$

If the MDB PRC is introduced into the echo path, the resultant power spectrum of echo signal will be the original power spectrum multiplied by the square of the magnitude of the transfer function given above. Thus, by introducing the MDB code, the power in the range of frequencies between half and one baud rate is reduced by 3.74 dB compared with that of alternate mark inversion (AMI) code, assuming the same total signal power. As a result, the aliasing distortion is reduced.

One other advantage associated with the use of the MDB PRC is that the power of the error due to jitter is reduced. In other words, even without the use of interpolation, the performance of the MDB system is improved compared with the AMI system. This can be explained by examining the power spectrum of the error signal:

$$S_{error}(\omega) \approx 2\pi E[\alpha_n^2] S_{sequence}(\omega) \omega^2 |G(\omega)|^2 \quad (6.2.2.1)$$

where  $\alpha_n$  is the jitter at time n,  $S_{sequence}(\omega)$  is the power spectrum of input data sequence and  $G(\omega)$  is the transfer function of the filters in the echo path. Because of the  $\omega^2$  factor, the error power increases dramatically at high frequencies. The use of the MDB PRC reduces the high frequency component. This is also the original reason for using the PRC to reduce the aliasing distortion, and consequently, the error caused by jitter is reduced. This phenomenon can also be explained in the time domain. If one system has less power at high frequencies than the other, the former system will

have a smoother and slower-changing time response than the latter system. Thus, for the same amount of displacement of the sampling instant, the error made by the system with less power at high frequencies will be smaller than that of the one with more power at high frequencies.

Equation (6.2.2.1) can be derived as the following: the expected (or calculated) echo  $e_n$  is given as

$$e_n = \sum_k g[(n - k)T]a_k$$

and the echo received in the presence of timing jitter with magnitude  $\alpha_k$  is

$$e'_n = \sum_k g[(n - k)T + \alpha_k]a_k$$

The difference between  $e_n$  and  $e'_n$  is the error produced in the presence of timing jitter

$$\begin{aligned} \epsilon_n &= e'_n - e_n \\ &= \sum_k \{g[(n - k)T + \alpha_k]a_k - g[(n - k)T]a_k\} \\ &= \sum_k \{g[(n - k)T + \alpha_k] - g[(n - k)T]\}a_k \end{aligned}$$

If  $\alpha_k$  is small

$$\epsilon_n \approx \sum_k \alpha_k g'[(n - k)T]a_k \quad (6.2.2.2)$$

where  $g'(\cdot)$  is the time derivative of  $g(\cdot)$ .

This is equivalent to the system shown in Fig. 6.2.2.1. Assuming both  $\{\alpha_n\}$  and  $\{a_n\}$  are independent and wide-sense stationary, we have

$$E_\alpha E_a[\alpha_n a_n \alpha_m a_m] = E_\alpha[\alpha_n \alpha_m] E_a[a_n a_m]$$

where  $E_\alpha$  is the expectation with respect to  $\alpha$ , and  $E_a$  is the expectation with respect to the data symbol  $a$ . Therefore, the autocorrelation  $R_{\alpha\alpha}$  is given by

$$R_{\alpha\alpha}(n - m) = R_\alpha(n - m)R_a(n - m)$$

Taking the Fourier transform, we have the power spectral density  $S_{\alpha\alpha}$  of the sequence  $\{\alpha_k a_k\}$ :

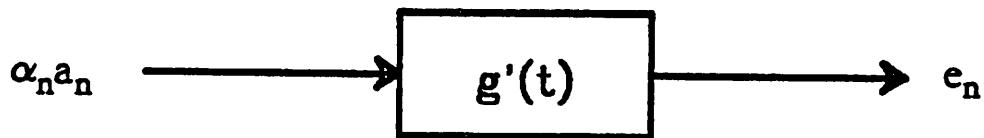


Figure 6.2.2.1.

$$S_{\alpha\alpha}(\omega) = S_\alpha(\omega)^* S_\alpha(\omega)$$

where  $*$  represents the convolution. The bandwidth of the timing jitter spectrum  $S_\alpha$  is normally narrower than that of the signal spectrum  $S_a(\omega)$ . we can approximate the jitter spectrum by a delta function located at DC with its amplitude equal to the total power:

$$S_\alpha(\omega) \approx 2\pi R_\alpha(0)\delta(\omega)$$

Then, it follows that

$$\begin{aligned} S_{\alpha\alpha}(\omega) &= S_\alpha(\omega)^* S_\alpha(\omega) \\ &\approx 2\pi R_\alpha(0) S_\alpha(\omega) \end{aligned} \tag{6.2.2.3}$$

From Equations (6.2.2.2) and (6.2.2.3), we can have the expression for the power spectral density of the prediction error  $S_\epsilon(\omega)$ :

$$\begin{aligned}
 S_\epsilon(\omega) &\approx S_{\alpha\alpha}(\omega) |j\omega G(\omega)|^2 \\
 &\approx 2\pi R_\alpha(0) S_a(\omega) |j\omega G(\omega)|^2 \\
 &= 2\pi E[\alpha_n^2] S_a(\omega) \omega^2 |G(\omega)|^2
 \end{aligned}$$

Here we have used the fact that

$$R_\alpha(0) = E[\alpha_n^2]$$

The amount of improvement due to the power shaping of the MDB coding is 2.1 dB over AMI code.

### 6.2.3. Interpolation

Mathematically, a k-th order interpolation is the fit of k+1 points by a k-th order polynomial. For example, first order interpolation is to use a straight line to fit two given points and second order interpolation is to fit three given points by a parabola. In general, the k-th order polynomial which fits k+1 points is given by:

$$\begin{aligned}
 f(t) &= f_n + (t - t_n) \frac{\nabla f_n}{h} + (t - t_n)(t - t_{n-1}) \frac{\nabla^2 f_n}{2} h^2 \\
 &\quad + \dots + (t - t_n) \dots (t - t_{n-k+1}) \frac{\nabla^k f_n}{k! h^k}
 \end{aligned}$$

where

$$\begin{aligned}
 \nabla f_n &= f_n - f_{n-1} \\
 \nabla^2 f_n &= \nabla(f_n - f_{n-1}) \\
 &= f_n - 2f_{n-1} + f_{n-2}
 \end{aligned}$$

$$h = t_n - t_{n-1}$$

and  $f_n$  is the value of the function at  $t=n$ .

Instead of a low pass filter, an interpolator is used here to perform the function of low pass filtering, i.e., to reconstruct the continuous waveform from discrete sampled data. There is a tradeoff between computation accuracy and hardware complexity. The higher the order of interpolation is, the better the approximation to ideal low

pass filter will be. But in the meantime, the hardware will become more complicated. Therefore, minimum order of interpolation is desired in order to minimize the hardware complexity, yet still adequate to achieve the desired accuracy. Based on computer simulation, the minimum order of interpolation acceptable is found to be two. Some computer simulation results are presented in Section 6.4.

### 6.3. Extension to the memory-based echo canceller

The technique derived in the previous section relies on the knowledge of the baud-spaced samples of the echo path impulse response, which are equal to the values of the coefficients of the linear echo canceller of the linear system. The echo signal is given by

$$e_n = \sum_{k=0}^{N-1} g_k a_{n-k}$$

For a nonlinear system, the echo signal is a general nonlinear function of the data symbols, i.e.,

$$\begin{aligned} e_n &= f(a_n, a_{n-1}, \dots, a_{n-N+1}) \\ &= \sum_{k=0}^{N-1} g_{1,k} a_{n-k} + \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} g_{2,(k,l)} a_{n-k} a_{n-l} + \text{higher order terms} \end{aligned} \quad (6.3.1)$$

When a phase jump occurs, the echo signal becomes:

$$\begin{aligned} e_n &= f'(a_n, a_{n-1}, \dots, a_{n-N+1}) \\ &= \sum_{k=0}^{N-1} g_{1,k} a_{n-k} + \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} g_{2,(k,l)} a_{n-k} a_{n-l} + \text{higher order terms} \end{aligned}$$

The concept of the estimation of the new sampled echo-path impulse response cannot be applied directly. However, since the echo signal is dominated by the first summation on the right hand side if the degree of nonlinearity is small, we can apply the same idea derived for linear system to estimate the new  $g_{1,k}$  when timing jitter occurs. And the effect of the timing jitter on those higher order terms is of second order effect and can be neglected. For example, if we consider a system composed of a linear part

with impulse response  $g_k$  and a nonlinear element whose input output relationship is given by

$$y = x + bx^2, \quad b \ll 1$$

The echo signal is

$$e_n = \sum_k g_k a_{n-k} + b \sum_k \sum_l g_k g_l a_{n-k} a_{n-l}$$

In the presence of jitter, the echo signal becomes:

$$\begin{aligned} e_n &= \sum_k g_k a_{n-k} + b \sum_k \sum_l g_k g_l a_{n-k} a_{n-l} \\ &= \sum_k (g_k + g_k - g_k) a_{n-k} + b \sum_k \sum_l (g_k + g_k - g_k)(g_l + g_l - g_l) a_{n-k} a_{n-l} \end{aligned}$$

The error in the first term is of the order  $(g_k' - g_k)$  and can be eliminated by the technique developed for the linear system. The error in the second term is of the order  $b(g_k' - g_k)g_l$  and therefore can be neglected. Some simulations have been done and confirmed the validity of the extension of the interpolation concept to the nonlinear system.

However, for a memory-based nonlinear echo canceller, the samples of the "echo path impulse response" ( $g_{1,k}$ ) are not known. This is different from the linear echo canceller case, where the tap coefficients are equal the sampled echo path impulse response and the echo replica is produced by performing the convolution

$$\hat{e}_n = \sum_{k=0}^{N-1} c_k a_{n-k}$$

where  $c_k$  is the coefficients of the echo canceller. For the memory-based nonlinear echo canceller, every echo replica corresponding to a specific input data sequence is stored in a particular memory location. The echo replica is obtained by fetching the content of the memory whose address is the current data vector  $\bar{a}_n = (a_n, a_{n-1}, \dots, a_{n-N+1})$ . In other words,

$$\hat{e}_n = \text{Memory}(\bar{a}_n)$$

and no convolution is needed. Thus, unlike the linear echo canceller, the  $c_{1,k}$ , or

equivalently  $g_{1,k}$ , are not explicitly known. However, if we examine Equation (6.3.1) and let the data vector be  $(1,0,0, \dots, 0,0)$ , the echo signal will be equal to.

$$\begin{aligned} e_n &= g_{1,0} + g_{2,(0,0)} + \text{higher order terms} \\ &\approx g_{1,0} \end{aligned}$$

Therefore, the content of the memory at location  $(1,0, \dots, 0,0)$  is approximately equal to  $g_{1,0}$ . Similarly,

$$\begin{aligned} \text{Memory}(0,1,0,0, \dots, 0,0) &\approx g_{1,1} \\ \text{Memory}(0,0,1,0, \dots, 0,0) &\approx g_{1,2} \\ \text{Memory}(0,0,0,1, \dots, 0,0) &\approx g_{1,3} \end{aligned}$$

$$\begin{aligned} \text{Memory}(0,0,0,0, \dots, 1,0) &\approx g_{1,(N-2)} \\ \text{Memory}(0,0,0,0, \dots, 0,1) &\approx g_{1,(N-1)} \end{aligned}$$

Therefore, using the contents of these memory locations, the same technique developed for linear echo canceller can be applied. This is the approach we used in the breadboard experiment described in Section 6.5.

#### 6.4. Simulation Results

The transmit and receive filters used in the simulation are designed to minimize the intersymbol interference. They have a single real pole and two pairs of complex poles respectively [8]. The first part of the simulation assumes that the transmitter does not follow the recovered timing phase. This serves the purpose of illustrating the characteristics of the jitter performance, which includes the degree of degradation and the rate of convergence. Notice that this is also the situation in the central office side, where the transmit timing is controlled by the master clock.

Fig. 6.4.1 shows the jitter performance of the system with AMI code. This is used as the reference for the comparison. The magnitude of jitter is 1/64. This corresponds to a digital phase locked loop with a local oscillator running with 10 MHz. Here, the

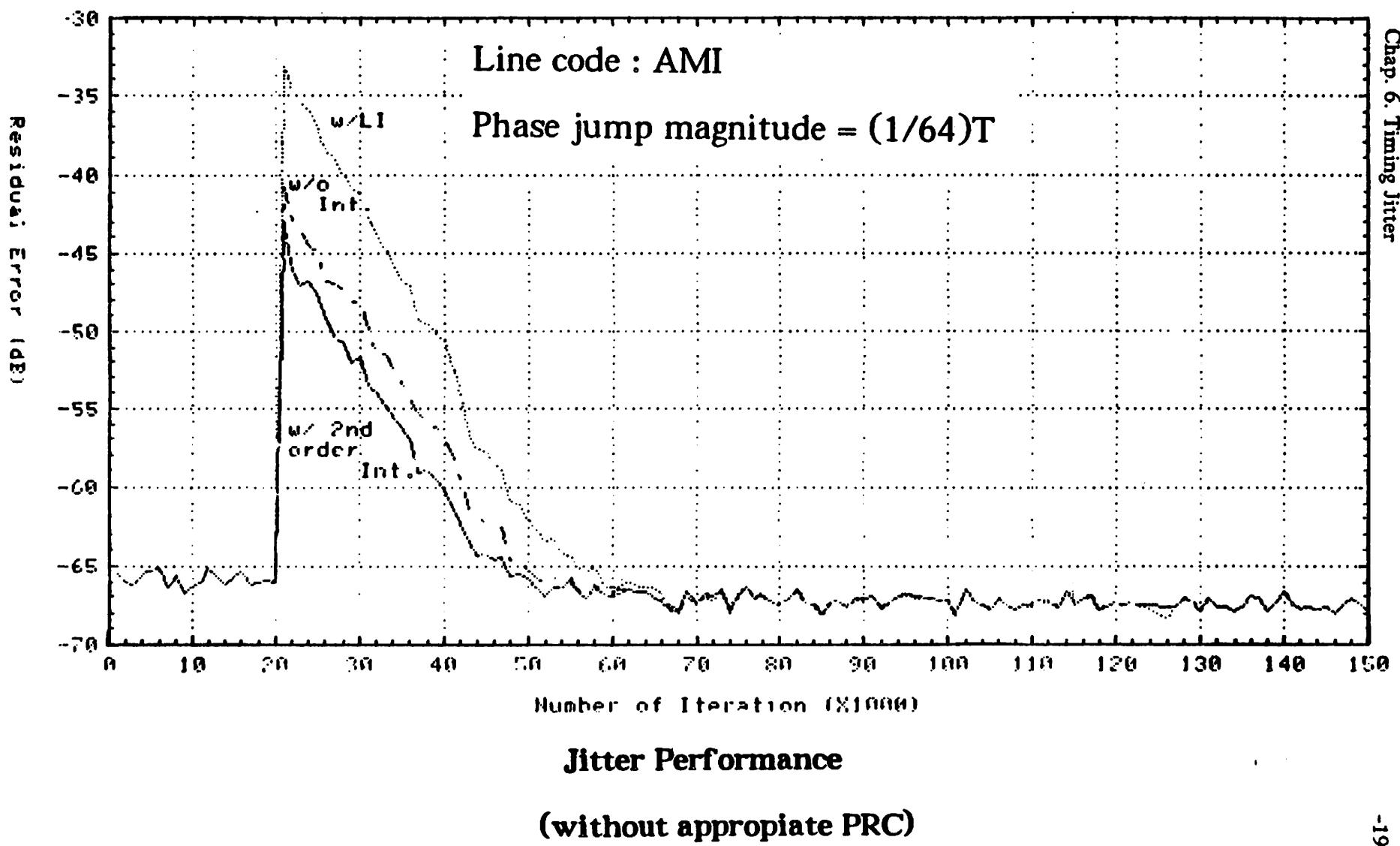


Figure 6.4.1. Jitter performance of the AMI system.

system is assumed to be purely linear, and therefore, the echo canceller is an adaptive transversal filter with linear taps only. Three curves are shown in the Figure. They represent the situation of no interpolation, first-order interpolation, and second order interpolation respectively. In this figure, the result of first-order interpolation is seen to be worse than that of no interpolation. This is because the improvement due to interpolation is less significant than the distortion introduced by non-ideal low pass filtering, which is first order interpolation. The result of using second order interpolation does improve the system performance, but the improvement is less than 3 dB.

Fig. 6.4.2 shows the jitter performance of a linear system, but with modified duobinary PRC instead. Again three curves are shown in this figure. They correspond to those curves shown in Fig. 6.4.1. The result is that, for a system with modified duobinary PRC, the use of second order interpolation improves the performance by about 6 dB. If compared to an AMI system, more than 9 dB improvement is achieved. Fig. 6.4.3 is the jitter performance of periodic jitter. The magnitude of jitter is 1/512 per step.

Fig. 6.4.4 is the jitter performance of a slightly nonlinear system. The nonlinearity is assumed to be in the form of

$$f(x) = x - b|x|$$

where  $b = 0.005$ . Here, the echo canceller is assumed to have four additional nonlinear taps [9]. The result indicates that this technique also works for slightly nonlinear systems.

In the subscriber end of the DSL, the transmitter follows the recovered timing phase, the effect of jitter will only last for a very short period of time. The number of periods it lasts depends on how fast the impulse response dies away. Typically, it is smaller than twenty periods. The second part of the simulation assumes that the phase of transmitter is synchronized to the recovered timing phase. Since every point

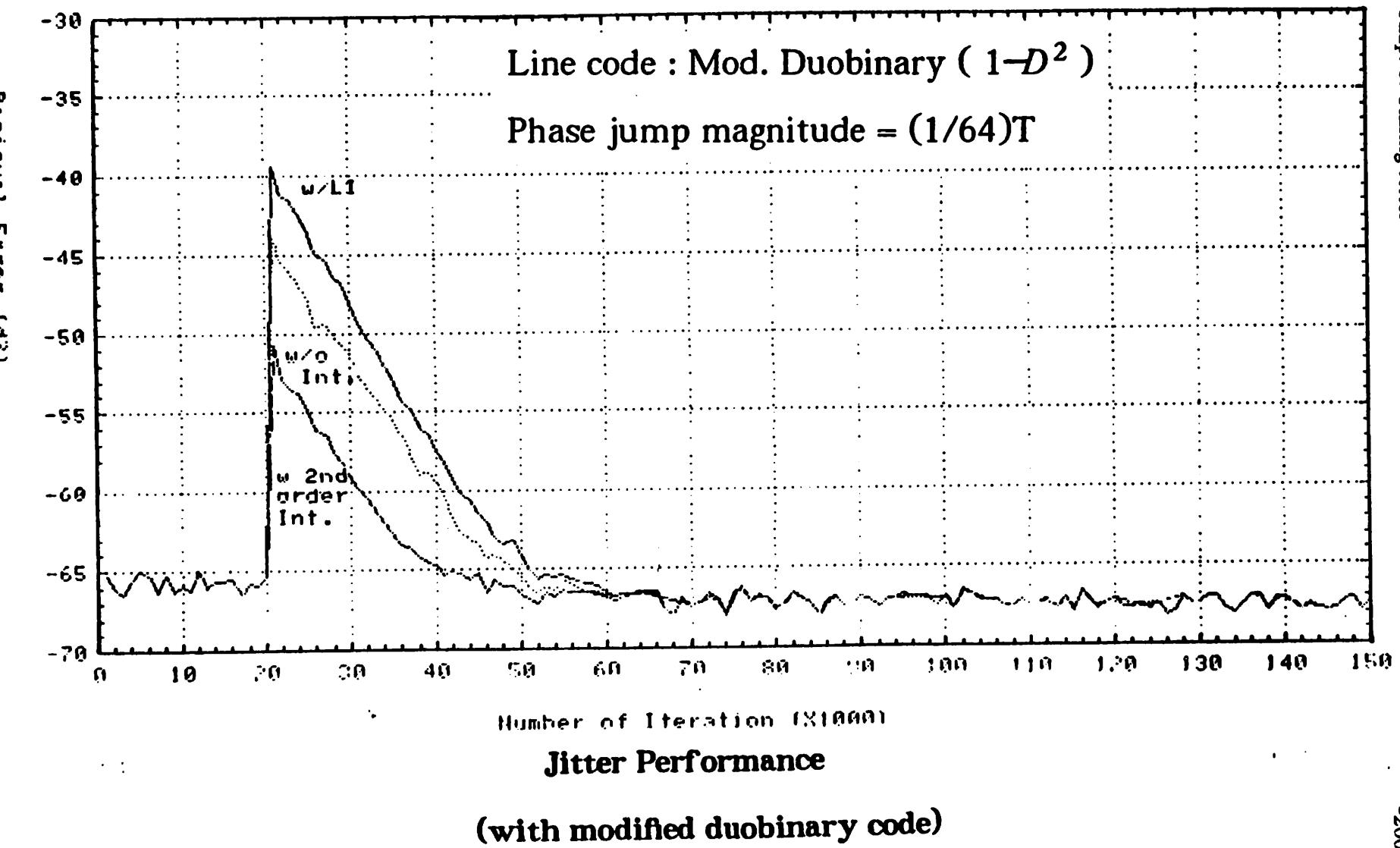


Figure 6.4.2. Jitter performance of the MDB system.

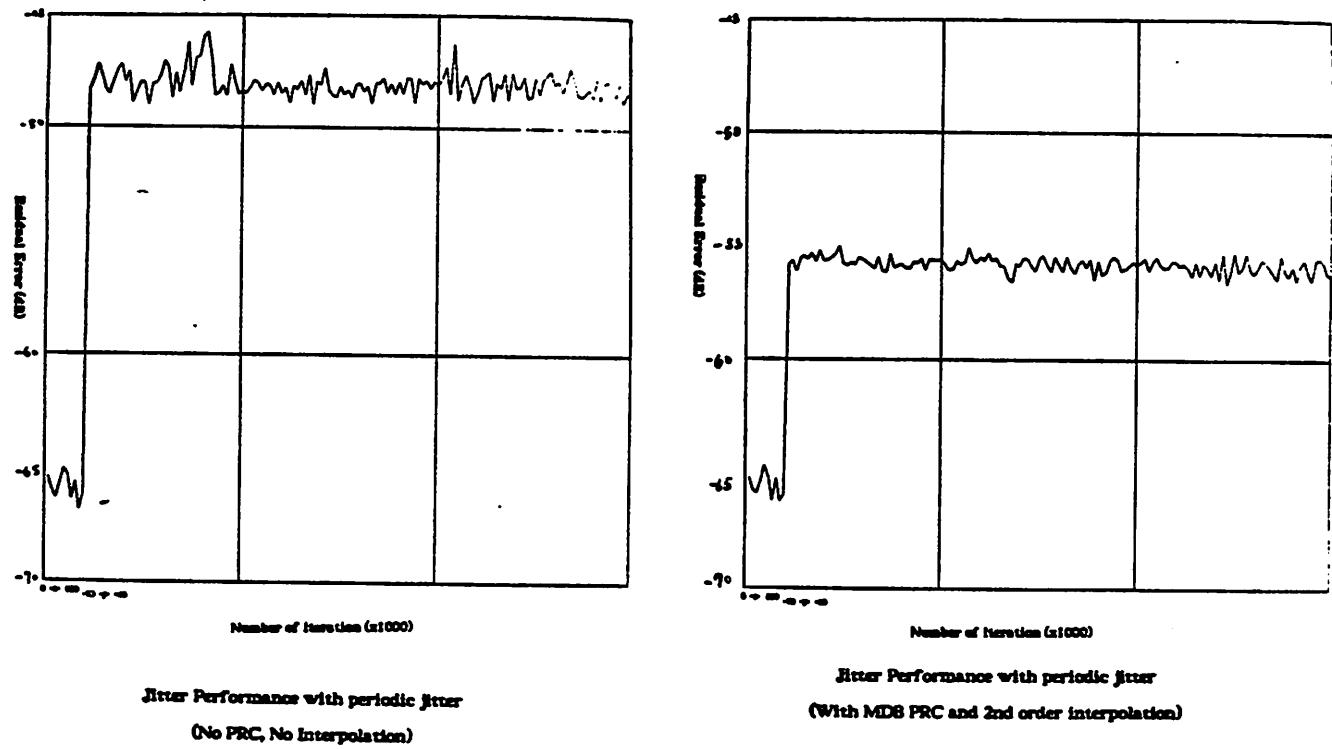
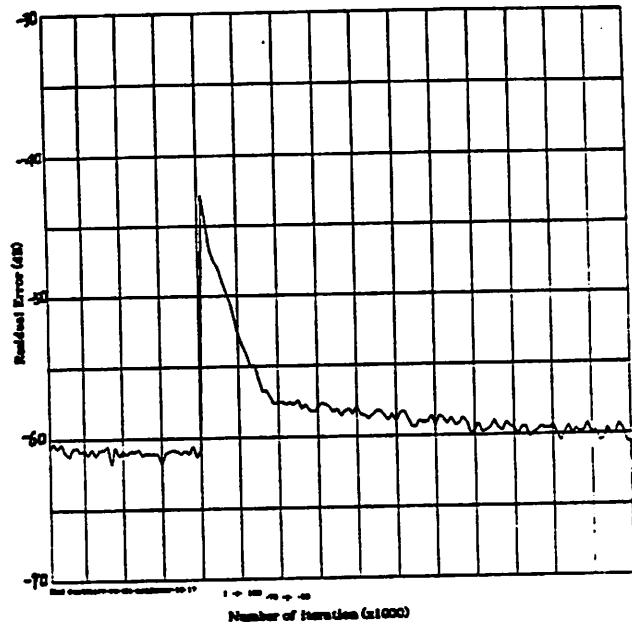
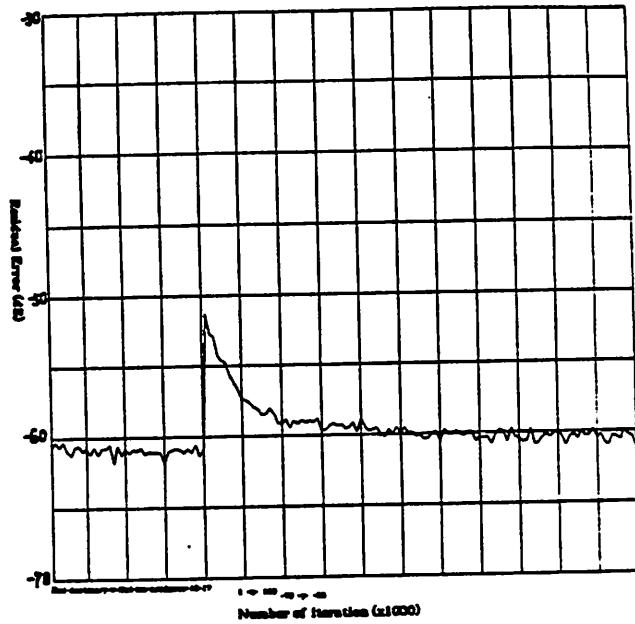


Figure 6.4.3. Jitter performance of periodic jitter.



Jitter Performance in the presence of nonlinearity  
(with modified duobinary, no interpolation)

$$f(x) = x - b|x|, b = 0.005$$



Jitter Performance in the presence of nonlinearity  
(with modified duobinary and 2nd order interpolation)

$$f(x) = -b|x|, b = 0.005$$

$$f(x) = x - b|x|, b = 0.005$$

Figure 6.4.4. Jitter performance of a nonlinear system.

in those figures of the first part simulation has been calculated by averaging over one thousand periods, it is not possible to see the effect of jitter for the scheme of the second part. But still, the transient response should be examined. Since the transient response of jitter performance depends on the input data pattern, the simulations and calculations are done by examining the worst situation and root-mean-square case. The results are shown in Fig. 6.4.5. For an AMI system without interpolation, the eye is completely closed for the worst case and the eye opening is small for the rms case. For MDB system with interpolation, the eye opening is greater than 65% even in the worst case. They once again confirm that the use of modified duobinary PRC and second order interpolation improves the jitter performance.

It is worth mentioning that the same set of transmit and receive filters are used in all the simulations above. This means that both the AMI system and the MDB system in the simulations have the same excess bandwidth. We can improve the performance of the MDB system further by taking the advantage of the very important fact that less excess bandwidth is required for MDB system to achieve the same maximum horizontal eye opening. In fact, the use of MDB coding make it possible to have zero excess bandwidth. By reducing the excess bandwidth, the aliasing distortion can be reduced further and consequently the eye can be widened further.

## 6.5. Hardware Implementation

The equation to be implemented for the second order interpolation is

$$\begin{aligned} C'_k &= C_{k+1} - (C_{k+1} - C_k)(1 - \delta) - 0.5(1 - \delta)\delta(C_{k+1} - 2C_k + C_{k-1}) \\ &= C_k(1 - \delta^2) + (C_{k+1} - C_{k-1})(0.5\delta + 0.5\delta^2) \end{aligned}$$

where  $C'_k$  is the corrected coefficient,  $C_k$ , the original coefficient, and  $\delta$ , the magnitude of the phase jump in baud. For small  $\delta$ ,  $\frac{1}{64}$  for example, the equation can be approximated as

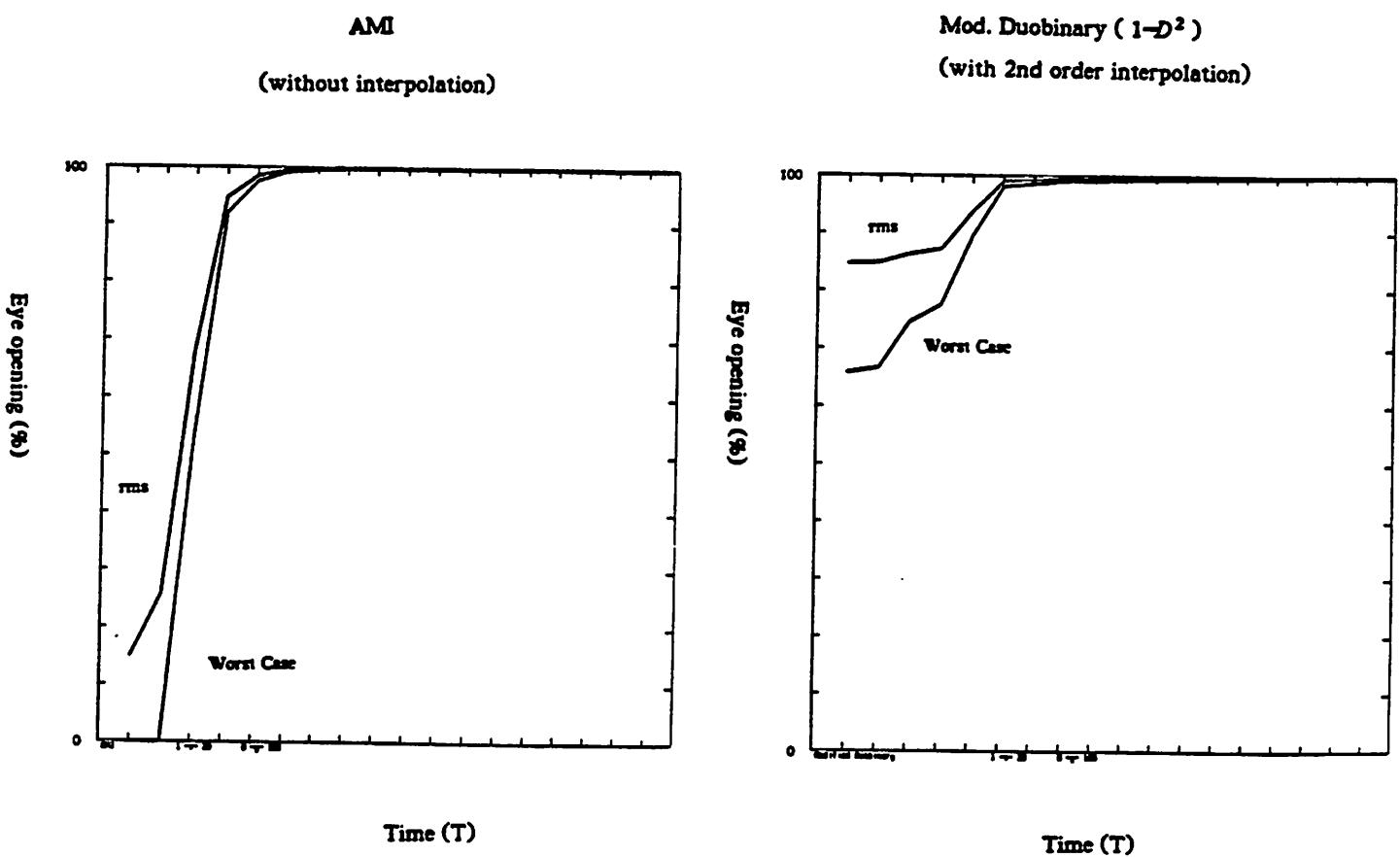


Figure 6.4.5. Eye opening of AMI and MDB systems.

$$C_k' = C_k + 0.5\delta(C_{k+1} - C_{k-1})$$

Since the digital PLL is used,  $\delta$  is fixed and known, thus no multiplier is needed. This equation can be calculated either digitally or by analog approach. To do it digitally, a simple arithmetic unit can do the job. The scaling of  $0.5\delta$  can be accomplished by right shifting the quantity  $(C_{k+1} - C_{k-1})$  if  $\delta$  is chosen to be in the form of  $2^{-n}$ , which is usually the case since this is a natural way in designing the digital PLL. For analog approach, a simple switched-capacitor filter with capacitor ratio equal to  $0.5\delta$  is sufficient. The choice between digital and analog approach depends on the structure of the echo canceller. A digital interpolator is more appropriate if a digital echo canceller is used. The overall correction produced by the interpolator is

$$\sum_{k=0}^N 0.5\delta(C_{k+1} - C_{k-1})a_{n-k}$$

## 6.6. Experimental realizations

In this section, a description of the breadboard realization of the echo cancellation mode DSL system will be given. In this system, MDB PRC is used in conjunction with the interpolation technique to tackle the timing jitter problem.

### 6.6.1. System architecture

Four alternative echo canceller configurations have been studied extensively in [2]. They are:

- (1) Fully analog implementation.
- (2) Fully digital implementation.
- (3) Digital echo canceller with analog cancellation.
- (4) Analog-digital echo canceller with analog cancellation.

Among them, fully analog implementation has been considered to be not feasible

for the present MOS technology. A digital echo canceller with analog cancellation has the advantage over the fully digital implementation in that the former needs a 160 kHz, 13-bit D/A while the latter must have a A/D of the same requirement. The D/A is easier to implement than a A/D of the same requirement. The difference between (3) and (4) is the domain in which the convolution is performed. In (3), the convolution is performed in the digital domain while in (4), it is performed by analog filtering. Since we chose the memory-based nonlinear echo canceller, no convolution is needed. Only additions are required to sum up the contents from different sub-memories (Chapter 2). In such a case, (3) and (4) become of little difference. We chose to do the summation in digital domain. The system block diagram is shown in Fig. 6.6.1.1.

### 6.6.2. State diagram

Fig. 6.6.2.1 shows the state diagram of the system. In the normal mode of operation, the digital processor generates echo replica and updates the content of the RAM according to the sign of the residual error. In addition, it reads from RAM the equivalent linear echo canceller coefficients for the use during the timing jitter cycle. Upon receiving a phase jump command, i.e., when the JIT signal becomes active, the system performs interpolation and generates the correcting term for the echo replica according to the polarity of the phase jump. The system stays in the timing jitter mode for 16 baud periods and finally returns to the normal mode of operation.

### 6.6.3. Echo canceller

The block diagram of the echo canceller is shown in Fig. 6.6.3.1. It is a memory-based nonlinear canceller covering 16 baud period. The word length of the echo canceller is 24 bits/word. It is chosen to fulfill the accuracy and the long adaptation time constant requirements. In order to reduce the memory size and to speed up the

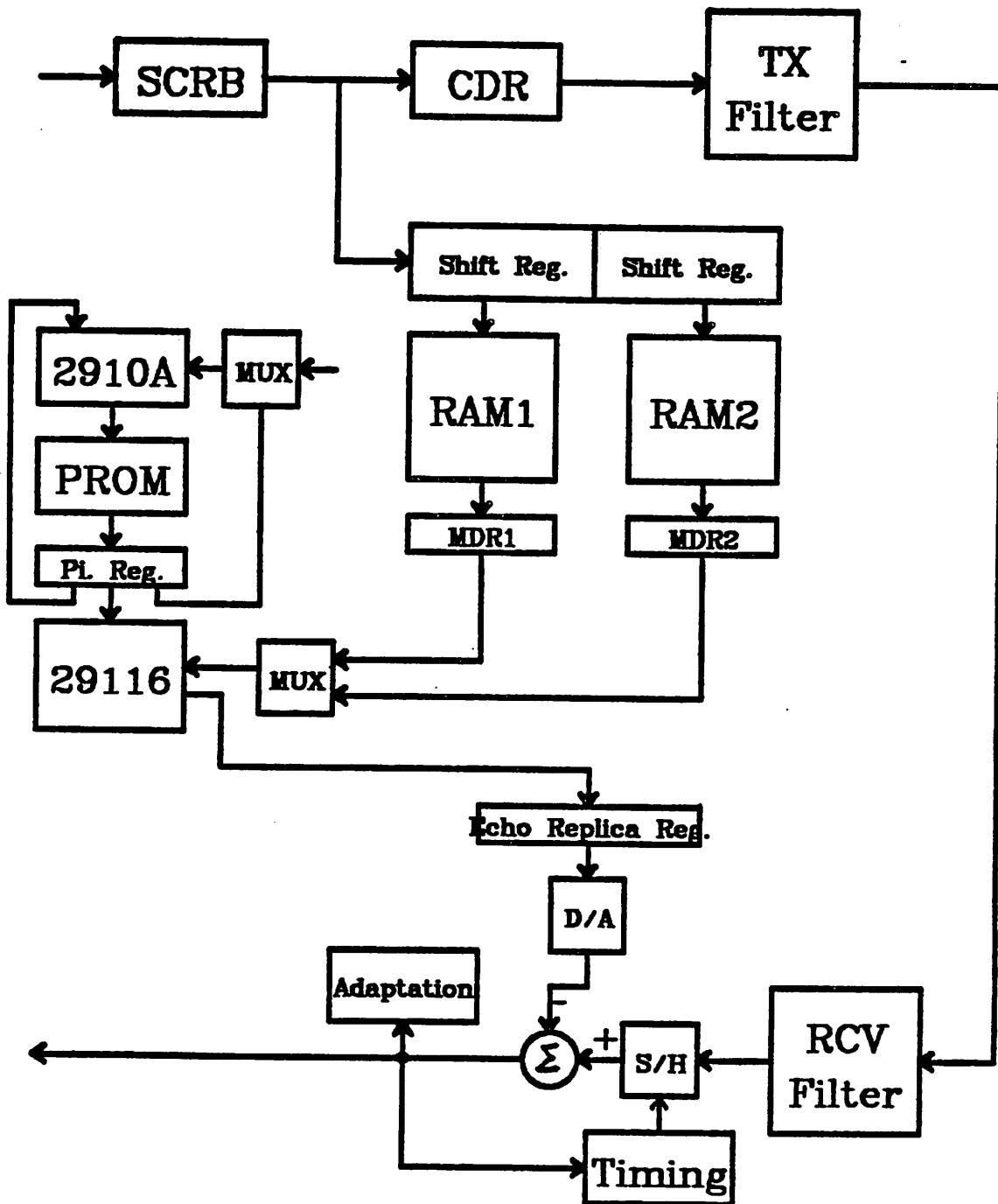
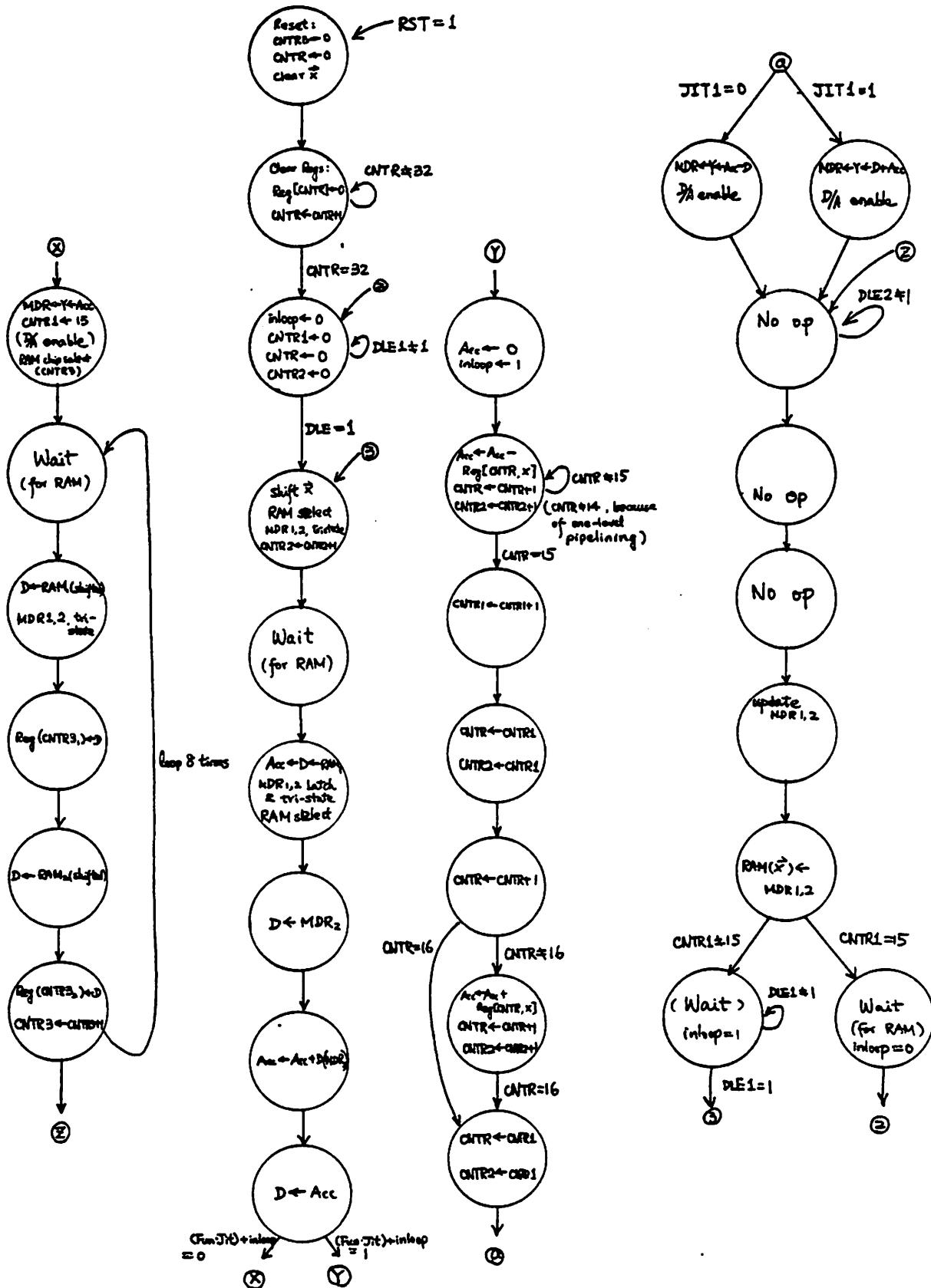


Figure 6.6.1.1.



**Figure 6.6.2.1.** State diagram.

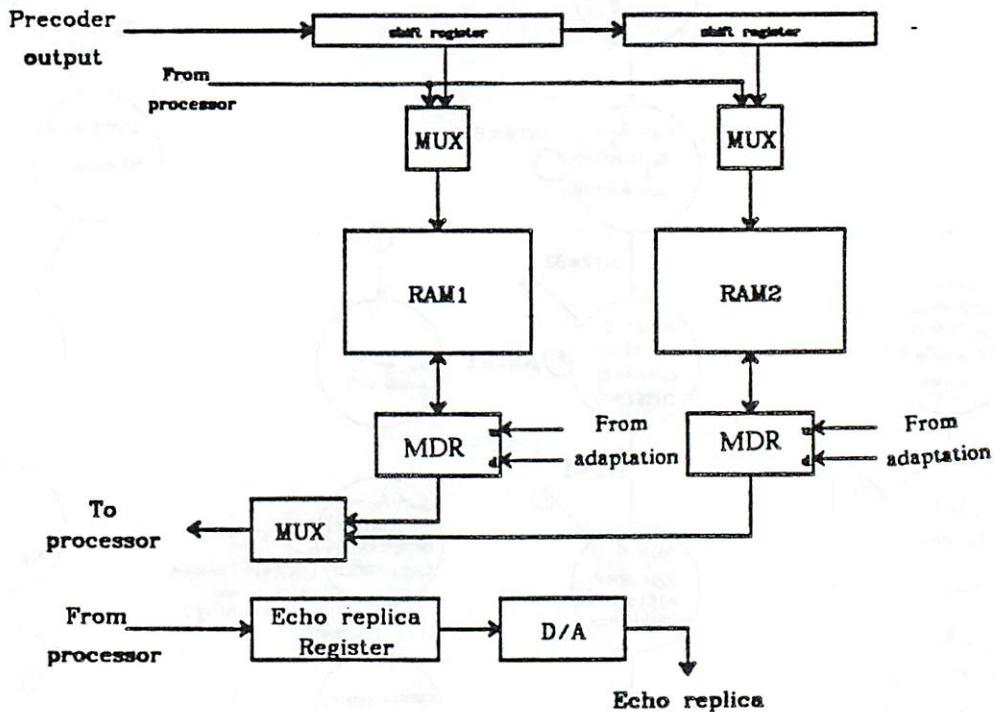


Figure 6.6.3.1.

adaptation, the memory is broken up into two separate sub-memories. Therefore, the total memory size becomes  $2 \cdot 2^8$  words, i.e., 512 words. Despite of this effort, we found that the rate of convergence was still extremely slow. Therefore, We feel that the memory should be partitioned into more separate sub-memories. The address input to the memory comes from either the transmitted data vector, which is stored in a shift register, or the digital processor, whose main function is to calculate the echo replica and to do the interpolation when phase jump occurs. To generate the echo replica, the transmitted data vector stored in the shift register is chosen as the address input of the echo canceller memory. The digital processor needs to read from the memory periodically those values corresponding to the linear coefficients to perform the interpolation in case of phase jump. Thus, the address generated by the digital processor is chosen to be the address input to the echo canceller memory during those

cycles. The outputs of the memory are loaded into the memory data registers as well as the input port of the processor, which, in the normal mode of operation, generates the echo replica and places it in the echo replica register. The echo replica register is connected to a D/A converter to generate an analog echo replica so that the cancellation can be performed in the analog domain. The memory data registers are two up-down counters whose contents are being counted up or down according to the sign of the residual error. In other words, sign adaptation is used in the experiment. Another effort is taken to speed up the adaptation. The step size is  $2^4$  times larger than its steady-state value during the transient and is switched back to the steady-state step size as the residual error becomes small. This is done by updating the 4th least significant bit during the transient.

#### 6.6.4. System controller and Digital processor

The system controller and the digital processor is shown in Fig. 6.6.4.1. They are consisting of an AM2910A, the microprocessor controller, a microprogram memory, a pipeline register, and an AM29116, the microprocessor. Basically, a one-level pipelining structure is used. The pipeline register contains the micro-instruction currently being executed. In the mean time, the address of the next micro-instruction is being fed to the address input of the microprogram memory, and the contents of the corresponding memory word are being fetched and set up at the inputs to the pipeline register. The micro-instruction can be divided into four sub-fields. The first field contains the instruction to the AM2910A. The second field is the next address select field whose content is used to select one among several condition signals as the AM2910A condition input. The third field contains the OP code of AM29116, and the last field is the control code needed for other part of the system. The microprogram memory is a programmable read-only memory (PROM). There are totally 31 micro instructions and the word length of the micro instruction is 52 bits.

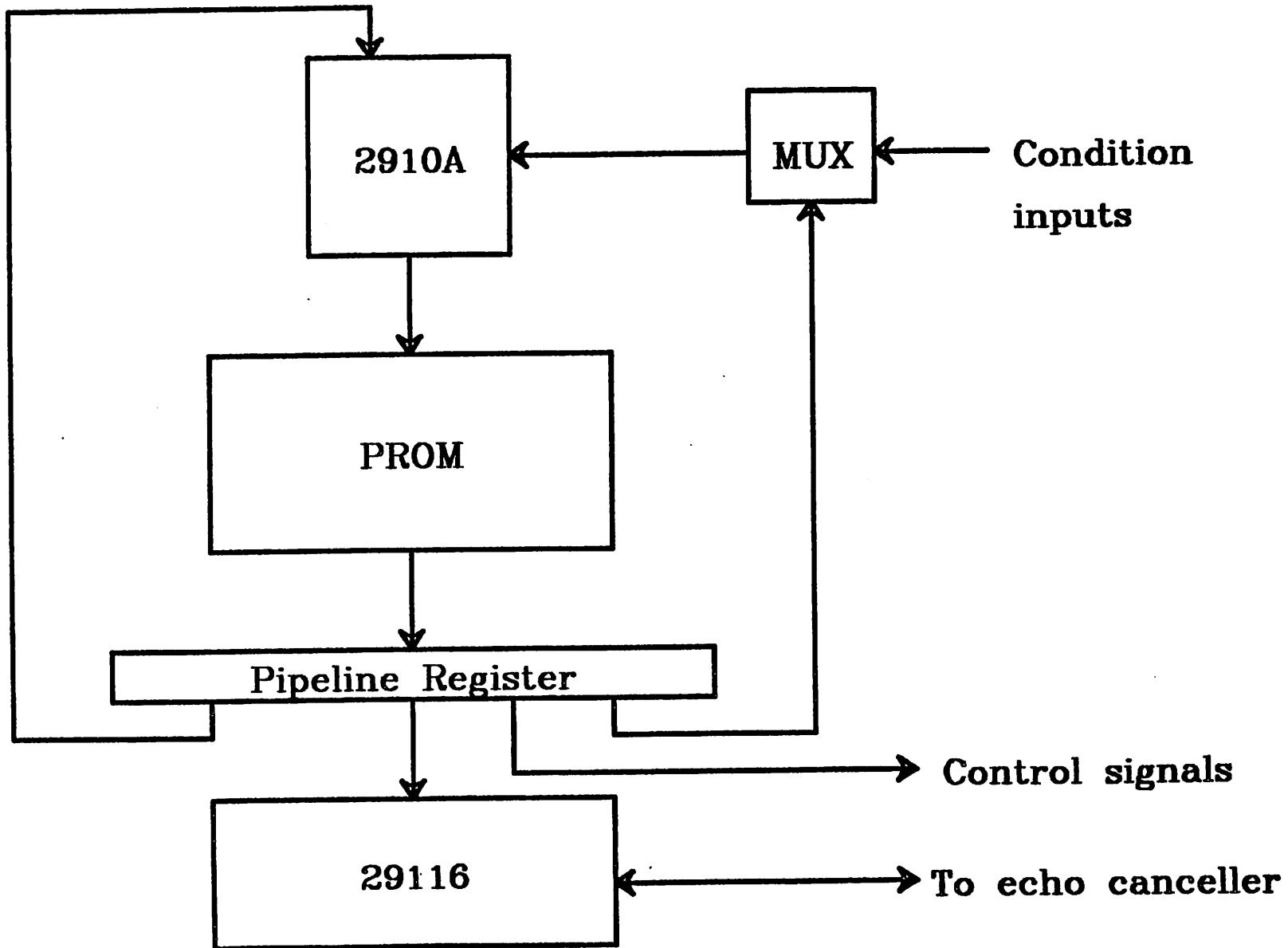


Figure 6.6.4.1. Controller and processor.

The AM29116 is a 16-bit bipolar microprocessor. It sums up the contents of the outputs of two sub-memories to produce the echo replica. Upon receiving the phase jump signal, it performs the interpolation to produce the echo replica correction. Since the word length of the memories in the echo canceller is 24 bits, the lower 8 bits are not used in the calculation of the echo replica.

The AM2910A is a microprogram controller. It generates the next address to the microprogram memory according to the instruction inputs as well as the condition inputs.

#### 6.6.5. Digital PLL and timing generation

The block diagram of the digital PLL is shown in Fig. 6.6.5.1.a. The loop filter is a divide-by-32 counter whose carry and borrow outputs are used to select one among three numbers 63,64,65 to be used in the digital VCO. The digital VCO, driven by a local oscillator running at 10 MHz, is normally a divide-by-64 counter. If the carry or the borrow signal of the loop filter becomes active, the digital VCO will generate an output pulse every 63 or 65 input pulses. The outputs of the digital VCO are fed to the timing generation circuitry which generates timing signals needed in the transmitter and in the sample-and-hold circuit.

A more detailed diagram is shown in Fig. 6.6.5.1.b. The JIT and JIT1 signals are needed in the system controller to control the digital processor to do the appropriate interpolating. Since the divide-by-32 counter is loaded with binary number 32 every time the signal "a" changes from 0 to 1 (meaning that the counter has just accumulated 32 up countings or 32 down countings), two consecutive phase jumps cannot occur within 32 baud periods. This guarantees that the echo cancellation system will return to the normal operation mode from the transient mode caused by timing jitter before the next phase jump occurs. Fig. 6.6.5.1.c is the timing diagram of the control signal used in the digital PLL circuit.

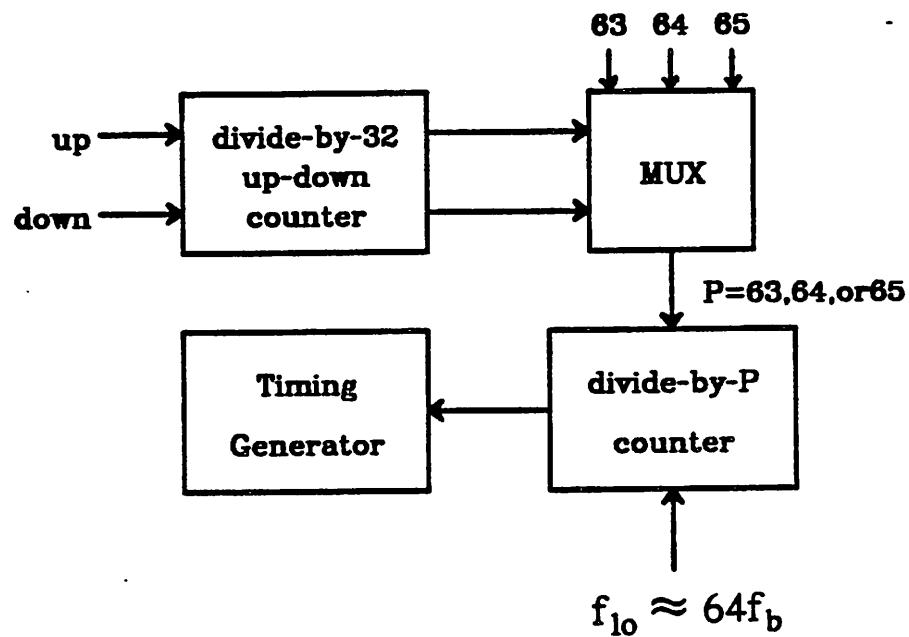


Figure 6.6.5.1.a. Block diagram of digital PLL.

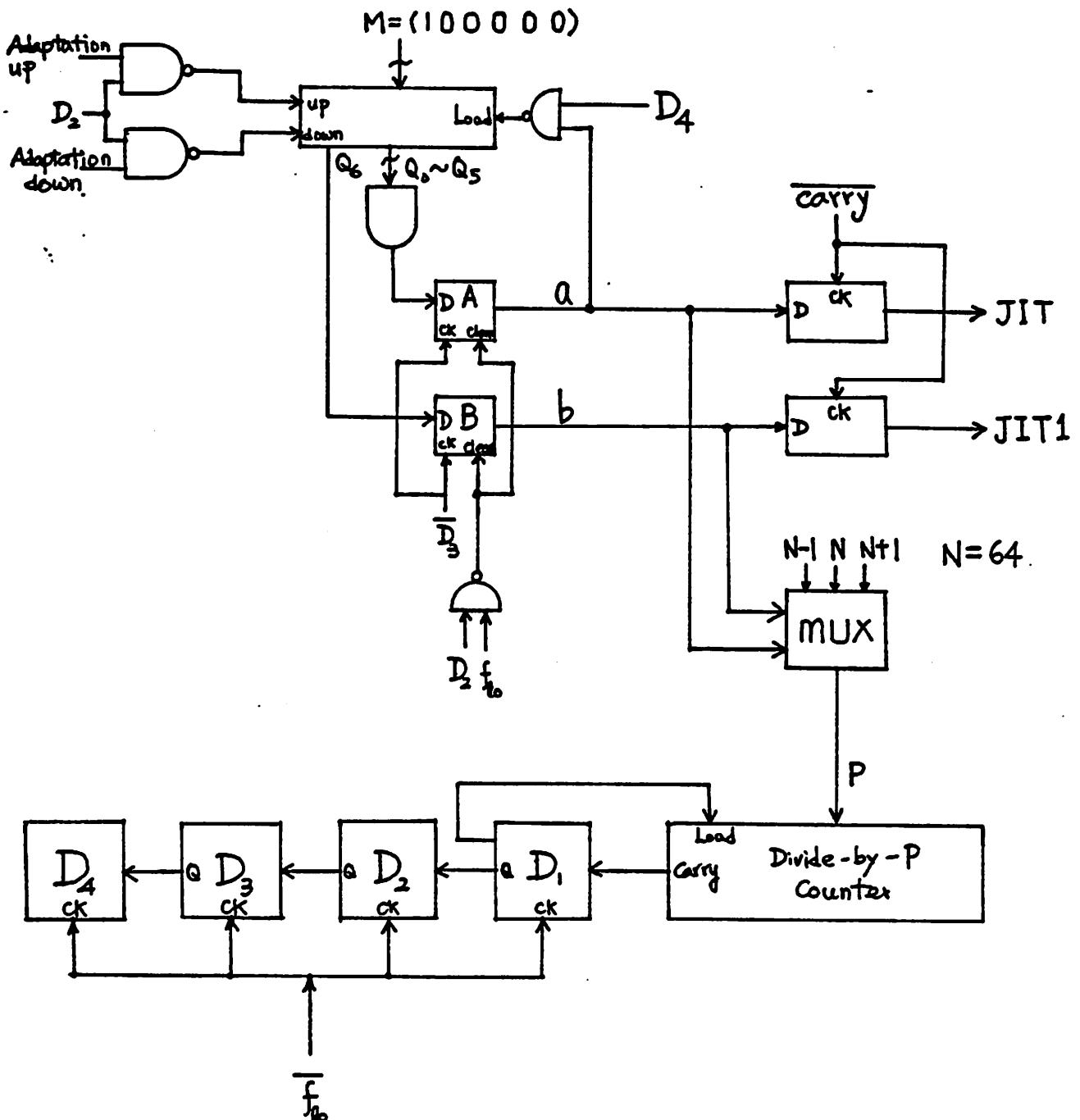


Figure 6.6.5.1.b

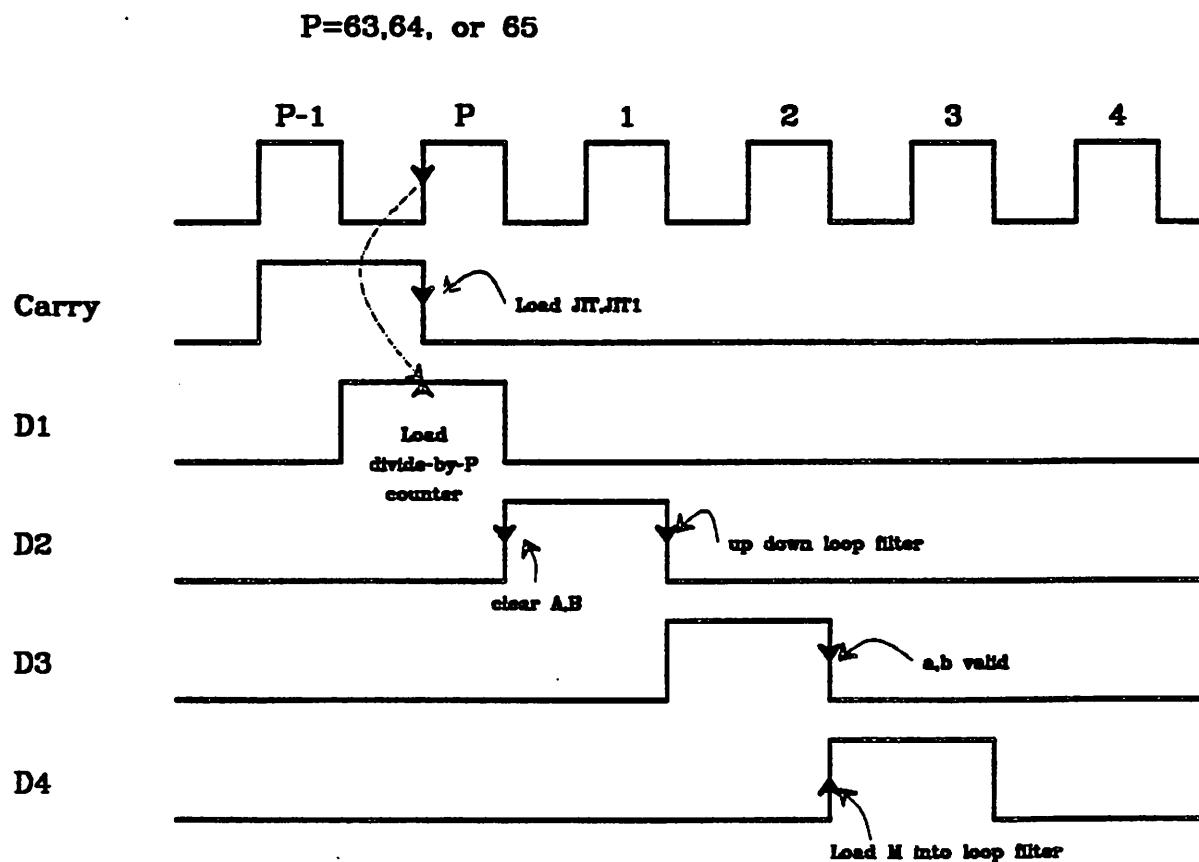


Figure 6.6.5.1.c

#### 6.6.6. Coder, pulse former, and filters

This part of the system is a modification of the existing system built by Winship and O'Riordan[37, 38]. The receive filter has two pairs of complex poles. These poles are (normalized to the baud frequency)  $-1.31 \pm j2.97$ , and  $-2.14 \pm j1.15$ . The transmit filter has a single real pole located at  $-1.3$ . The transmit filter and the receive filter are designed to minimize the intersymbol interference.

The coder is an  $(1 - D^2)$  MDB code. The pulse former gives 100% duty-cycle pulses. They were built using the erasable programmable logic arrays (EPLA's).

#### 6.6.7. Analog cancellation

The block diagram of the analog cancellation is shown in Fig. 6.6.7.1. The digital echo replica is first converted into an analog signal by a high speed D/A converter AD565A which gives a current output. A current-to-voltage converter is used to convert the current output of the AD565A into a voltage signal. An summing operational amplifier is used to generate a voltage whose magnitude is equal to the residual error. the difference between the echo signal and the echo replica. After sample-and-hold operation, the residual error is fed to the comparator to determine the sign of the residual error. This sign is needed in the adaptation of the echo replica.

#### 6.6.8. Experimental results

The experimental results agree with the computer simulations and confirm that the use of the interpolation technique alone can give 6 dB improvement. This is shown in Fig. 6.6.8.1.a where both the residual errors with and without interpolation are included. Fig. 6.6.8.1.b gives the spectrum of the AMI system and the MDB system. Fig. 6.6.8.1.c is the the pulse responses of AMI code and MDB code.

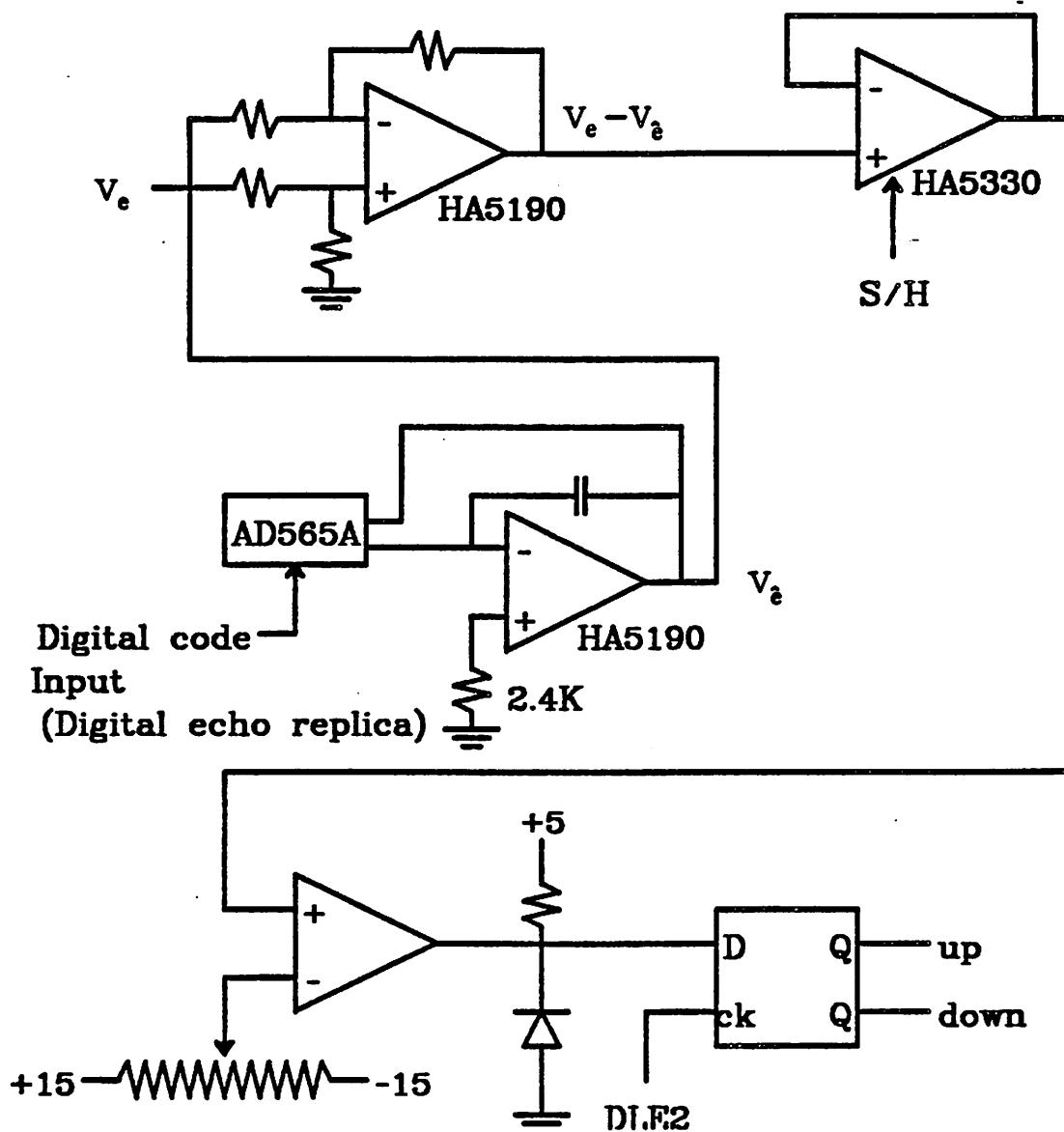


Figure 6.6.7.1. Analog cancellation.

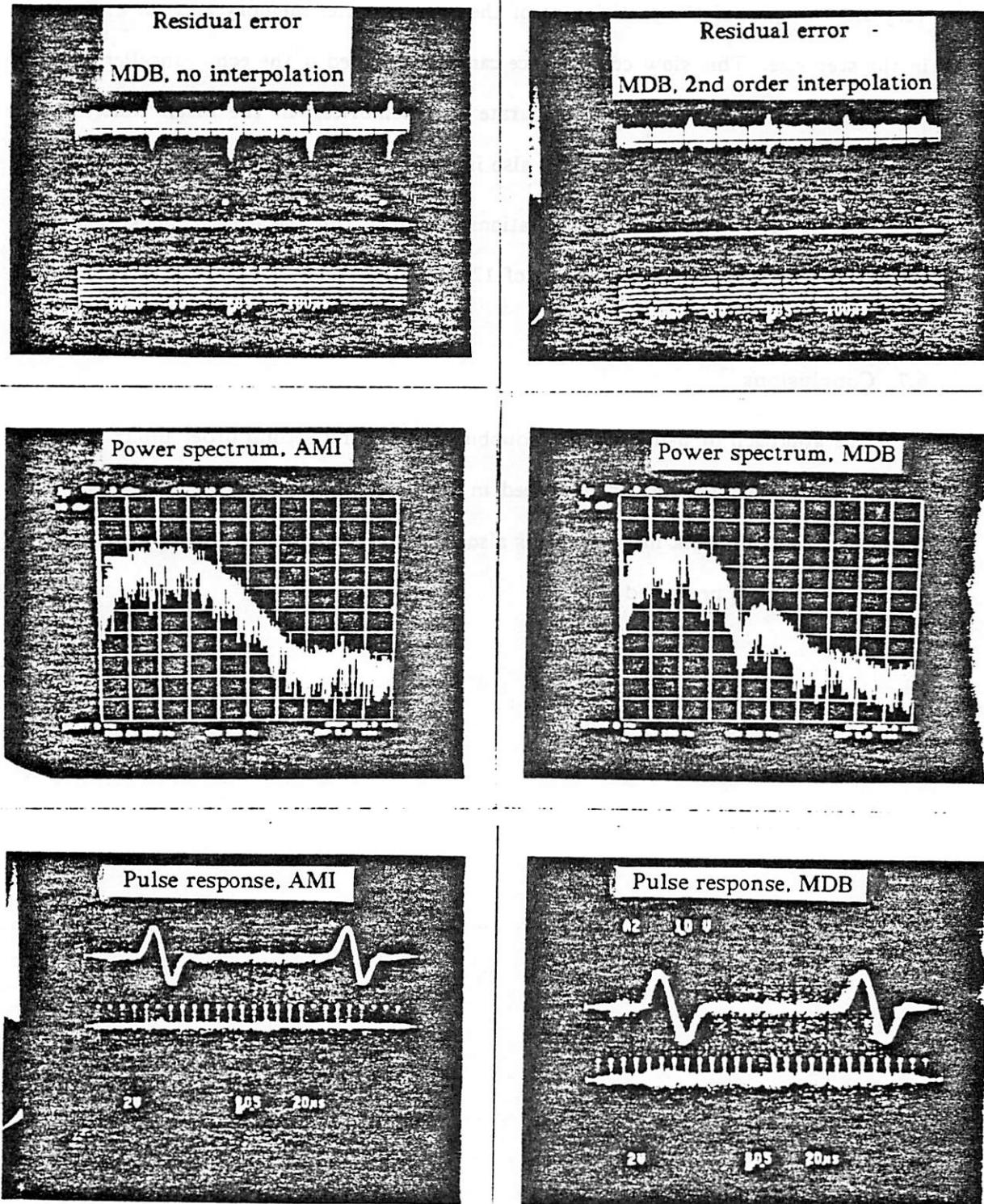


Figure 6.6.8.1

There are two limitations in this experiment. First, the rate of convergence is very slow despite of the partitioning of the echo canceller memory and the adjustment in the step size. This slow convergence can be improved if the echo canceller memory is further partitioned into more separate sub-memories. In the mean time, a more complicated step-size adjustment can also improve the convergence rate further.

Second, due to component limitations, we only run the master clock at 8 MHz. This corresponds to a baud frequency of 128kHz.

### 6.7. Conclusions

The approach of using modified duobinary PRC and second order interpolation to improve jitter performance is described in this chapter. The effect of modified duobinary PRC on crosstalk interference is also studied. Computer simulations and experimental results are presented.

## CHAPTER 7

### Conclusion

In this work, *modified duobinary partial response coding (PRC)* was studied and shown to be a desired line coding technique for digital subscriber loops (DSL). We showed that modified duobinary PRC offers an improved performance over AMI coding with essentially the same implementation complexity. With increased complexity, modified duobinary PRC gives the same ultimate performance as some block codings such as MMS43 and DI43. In addition, we demonstrated some realization alternatives of the modified duobinary code and concluded that this code offers a desirable tradeoff between implementation complexity and system performance.

The timing jitter problem was studied. We showed that an interpolation technique used in conjunction with modified duobinary coding can greatly relax the maximum allowable jitter in a digital subscriber loop system. This readily facilitates the design of a digital phase-locked loop.

A bread-board prototype system employing MDB PRC was built and the performance of the system with timing jitter was measured experimentally. The results were in good agreement with computer simulations.

## REFERENCES

1. William Stallings. *Data and Computer Communications*, Macmillan Publishing Company (1985).
2. Oscar E. Agazzi, D. A. Hodges, and D. G. Messerschmitt. "Large Scale Integration of Hybrid-method Digital Subscriber Loops." *IEEE Trans. on Communications COM-30* pp. 2095-2106 (Sept. 1982).
3. Jeremy C-P Tzeng. "Timing Recovery in Digital Subscriber Loops." *Ph. D. Dissertation*, University of California, Berkeley, (April 1985).
4. B. R. Narayana Murthy. "Crosstalk Loss Requirements for PCM Transmission." *IEEE Transactions on Communications COM-24(1)*(January 1976).
5. David. G. Messerschmitt. *Private Communication*.
6. P. E. Fleischer and L. Wu. "Novel Block Codes for DSL Applications." *IEEE Global Telecommunications Conference Conference record 43.4.1()*.
7. L. F. Lind and S. E. Nader. "Optimal Analogue Filters for Data Transmission." *Proc. IEEE International Symp. on Circuits and Systems*. pp. 334-347 (1977).
8. S. E. Nader and L. F. Lind. "Optimal Data Transmission Filters." *IEEE Trans. on Circuits and Systems CAS-26* pp. 36-45 (January 1979).
9. D. G. Messerschmitt. *U. C. Berkeley 290I lecture notes*. () .
10. J. G. Proakis. *Digital Communications*, McGraw-Hill Book Company (1983).
11. J. Salz. "Optimum Mean-Square Decision Feedback Equalization." *Bell System Tech. J.* 52 pp. 1341-1373 (October 1973).
12. Donald A. George, Robert R. Bowen, and John R. Storey. "An Adaptive Decision Feedback Equalizer," *IEEE Trans. on Communication Technology COM-19(3)* pp. 281-293 (June 1971).

13. David D. Falconer, "Adaptive Reference Echo Cancellation," *IEEE Trans. on Communications* COM-30(9) pp. 2083-2094 (September 1982).
14. Donald D. Duttweiler, "Adaptive filter performance with nonlinearities in the correlation multiplier," *IEEE Trans. Acoust., Speech, Signal Processing ASSP-30* pp. 578-586 (August 1982).
15. Adam Lender, "Correlative level coding for binary-data transmission," *IEEE Spectrum*, pp. 104-115 (February 1966).
16. E. R. Kretzmer, *Binary Data Communication by Partial-Response Transmission*, Conf. Rec IEEE Ann. Communications Conf. 1965.
17. Peter Kabal and Subbarayan Pasupathy, "Partial-Response Signaling," *IEEE Trans. on Communications* COM-23 pp. 921-934 (September 1975).
18. Mischa Schwartz, *Information Transmission, Modulation, and Noise*, McGraw-Hill Book Company (1980).
19. Graham Brand , Edward A. Lee, Nan-Sheng Lin, David A. Hodges, Vijay Madisetti, and David G. Messerschmitt, "Comparison of Line Codes and Proposal for Modified Duobinary," *American National Standards Institute, Telecommunications Committee*, (Nov. 19, 1985). Contribution no. T1D1.3/85-237
20. Graham Brand , Edward A. Lee, Nan-Sheng Lin, David A. Hodges, and David G. Messerschmitt, "Comparison of Line Codes with Optimum DFE Design," *American National Standards Institute, Telecommunications Committee*, (Jan. 24, 1986). Contribution no. T1D1.3/86-018
21. G. Schollmeier, *A Transmission Line Code for the Network Side of the NT - Basic Access*, Contribution no. T1D1.3/85-069, Siemens Communications Systems, Inc. May 6, 1985.

22. P. E. Fleischer and L Wu. *The D143 Line Code and its Spectral Properties*. Contribution no. T1D1.3/85-154, Bell Communications Research August 21, 1985.
23. John M. Wozencraft and Irwin Mark Jacobs. *Principles of Communication Engineering*, John Wiley & Sons, Inc. (1965).
24. John C. Campbell, Alan J. Gibbs, and Bernard M. Smith. "The Cyclostationary Nature of Crosstalk Interference from Digital Signals in Multipair Cable - Part II: Applications and Further Results." *IEEE Transactions on Communications* COM-31(5)(May 1983).
25. Oscar Agazzi, David G. Messerschmitt, and David A. Hodges. "Nonlinear Echo Cancellation of Data Signals." *IEEE Trans. on Comm.* COM-30(11)(November, 1982).
26. Burton R. Saltzberg. "Timing Recovery for Synchronous Binary Data Transmission." *The Bell System Technical Journal*, (March 1967).
27. J. E. Mazo. "Optimum Timing Phase for an Infinite Equalizer." *The Bell System Technical Journal*, (January 1975).
28. Jack Salz. "On Mean-Square Decision Feedback Equalization and Timing Phase." *IEEE Trans. on Communications* COM-25(12)(December 1977).
29. R. D. Gitlin and J. Salz. "Timing Recovery in PAM Systems." *The Bell System Technical Journal* 50(5)(May-June 1971).
30. Toshiro Suzuki, Hiroshi Takatori, Makoto Ogawa, and Keiji Tomooka. "Line Equalizer for a Digital Subscriber Loop Employing Switched Capacitor Technology." *IEEE Trans. on Communications* COM-30(9) pp. 2074-2082 (September 1982).
31. Oscar Agazzi, Chin-Pyng J. Tzeng, David G. Messerschmitt, and David A. Hodges. "Timing Recovery in Digital Subscriber Loops." *IEEE Trans. on Communications*

COM-33(6)(December, 1985).

32. F. M. Gardner, "A BPSK/QPSK Timing-Error Detector for Sampled Receivers." *IEEE Trans. on Communications* COM-34(5)(May 1986).
33. Hishi Kobayashi, "Simultaneous Adaptive Estimation and Decision Algorithm for Carrier Modulated Data Transmission Systems." *IEEE Trans. on Communications* COM-19(3) pp. 268-280 (June 1971).
34. Shahid U. H. Qureshi, "Timing Recovery for Equalized Partial-Response Systems." *IEEE Trans. on Communications*, (December, 1976).
35. Kuth H. Mueller and Markus Muller, "Timing Recovery in Digital Synchronous Data Receivers," *IEEE Trans. on Communications* COM-24(5) pp. 516-531 (May 1976).
36. Roland E. Best. *Phase-Locked Loops*, McGraw Hill Book Company () .
37. Peter O'Riordan, "Implementation and Testing of Timing Recovery Algorithm for Use in Digital Subscriber Loop." *Master's Report*, University of California, Berkeley, (August 1984).
38. Peter Winship, "An Implementation of a Coding, Equalization, and Timing recovery Algorithm for Use in the Digital Subscriber Loop." *Master's Report*, University of California, Berkeley, (December 1984).