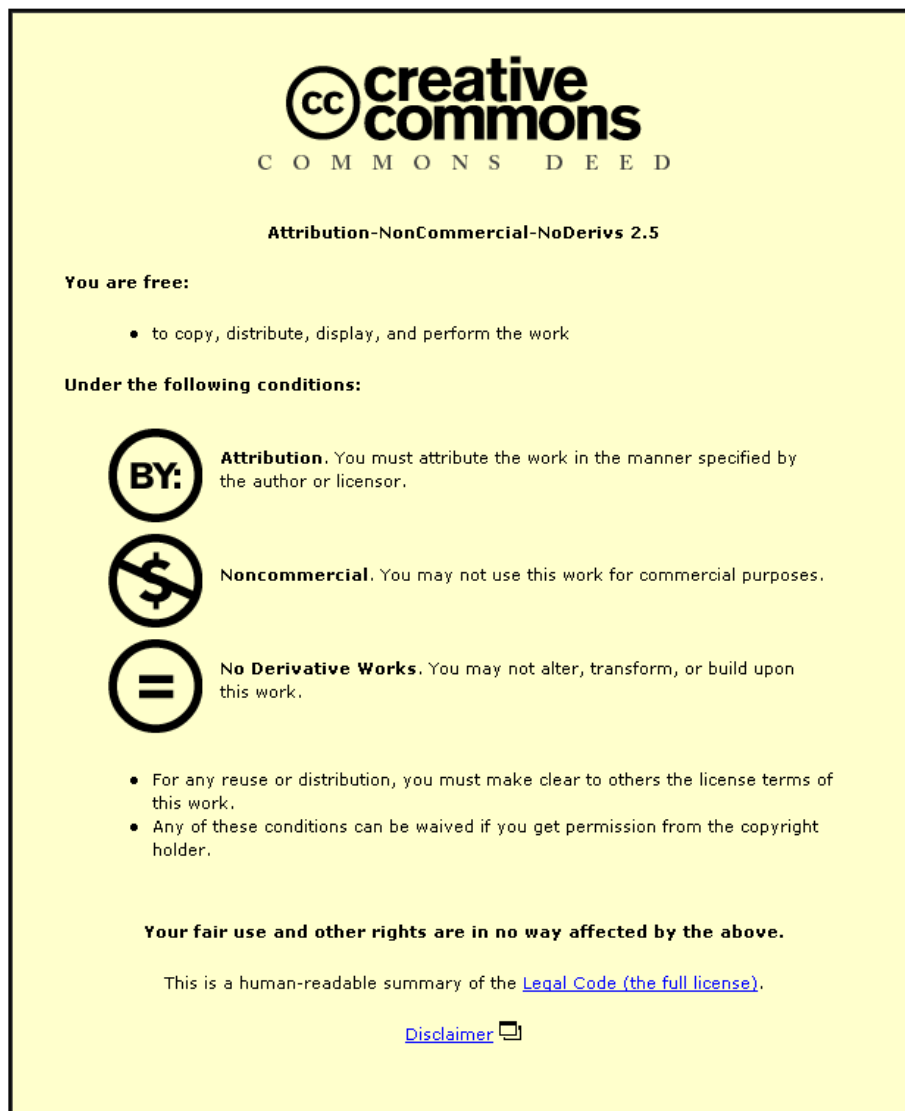


This item was submitted to Loughborough University as a PhD thesis by the author and is made available in the Institutional Repository (<https://dspace.lboro.ac.uk/>) under the following Creative Commons Licence conditions.



For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

LOUGHBOROUGH
UNIVERSITY OF TECHNOLOGY
LIBRARY

AUTHOR/FILING TITLE

BATEMAN, S C

ACCESSION/COPY NO.

008141/01

VOL. NO.

CLASS MARK

T

- 5 JUL 1991

- 3 JUL 1992

- 2 JUL 1993

6 JUL 1990

000 8141 01



This book was bound by

Badminton Press

18 Half Croft, Syston, Leicester, LE7 8LD

Telephone: Leicester (0533) 602918.

LOUGHBOROUGH
UNIVERSITY OF TECHNOLOGY
LIBRARY

AUTHOR/FILING TITLE

BATEMAN, S C

ACCESSION/COPY NO.

008141/01

VOL. 2.

CLASS MARK

T

240 25

01. JUL

01. OCT 83

~~20. FEB~~

~~03. OCT 87~~

30 JUN 1989

~~16. MAR 87~~

~~25. APR 87~~
~~1 JUL 1988~~

~~04. JUN 87~~

30 JUN 1989
- 6 JUL 1990

DATA TRANSMISSION AT 19,200 BIT/S
OVER TELEPHONE CHANNELS.

by

S C Bateman, B.Sc.

A Doctoral Thesis

*Submitted in partial fulfilment of the requirements for the
award of Doctor of Philosophy of the Loughborough University
of Technology.*

October 1985

Supervisor: Prof. A P Clark
Department of Electronic and Electrical Engineering

© by S C Bateman, 1985

Loughborough University of Technology Library	
Date	June 86
Class	T
Acc. No	008141/01

List of Contents

	<u>Page</u>
ABSTRACT	(vi)
ACKNOWLEDGEMENTS	(vii)
GLOSSARY OF SYMBOLS AND ABBREVIATIONS	(viii)
1. INTRODUCTION	
1.1 Background	1
1.2 Outline of the Investigation	2
2. GENERAL CONSIDERATIONS FOR DIGITAL DATA TRANSMISSION SYSTEMS	
2.1 Introduction	7
2.2 Model of a Data Transmission System	7
2.3 19,200 bit/s over Telephone Channels	25
2.4 References	32
3. THE TELEPHONE CHANNEL; CHARACTERISTICS AND IMPAIRMENTS	
3.1 Introduction	46
3.2 Types of Telephone Circuits	48
3.3 Characteristics of Telephone Circuits	50
3.3.1 Characteristics of Switched Connections	60
3.3.2 Characteristics of Private Lines	63
3.4 Other Factors Influencing Data Transmission over Telephone Circuits	65
3.4.1 Frequency Modulation Effects	66
3.4.2 Amplitude Modulation Effects	70
3.4.3 Additive Noise	72
3.5 Representation of Line Characteristics and Impairments for Theoretical and Practical Development of Modems	77
3.6 References	80

4.	BANDPASS, BASEBAND AND MODULATION SYSTEMS	
4.1	Introduction	99
4.2	Bandpass Signals and Systems and the Equivalent Baseband Model	100
4.3	Selection of Modulation Method	103
4.4	QAM Signal Sets	118
4.4.1	Coding Scheme 1 ; 64-Point Structure	122
4.4.2	Coding Scheme 2 ; 64-Point Structure	126
4.4.3	Coding Scheme 3 ; 64-Point Structure	130
4.4.4	Coding Scheme 4 ; 256-Point Structure	133
4.5	References	135
5.	EQUIPMENT FILTERS AND OVERALL BASEBAND CHARACTERISTICS	
5.1	Introduction	166
5.2	General Requirements of the Equipment Filters	168
5.3	Equipment Filter Design	174
5.3.1	The Digital Low-Pass Filters	175
5.3.2	The Analogue Low-Pass and Bandpass Filters	176
5.4	Characteristics of Combined Transmitter and Receiver Filters	177
5.5	Characteristics of the Three Test Telephone Circuits	178
5.6	Sampled Impulse Responses of the Overall Linear Baseband Channels	179
5.7	References	192
6.	DETECTION PROCESSES FOR 19,200 BIT/S MODEMS	
6.1	Introduction	226
6.2	General Model and Basic Assumptions	227
6.3	Equalisation Techniques	234
6.3.1	Linear Equalisation	234
6.3.2	Non-Linear Equalisation	238

	<u>Page</u>
6.4 Maximum-Likelihood Detection Processes	242
6.4.1 The Viterbi Algorithm Detector	243
6.5 The Selected Near-Maximum-Likelihood Detector	246
6.6 Computer Simulation Results and System Comparisons	257
6.7 Effect of the Detection Parameters α , n and γ on Detector Complexity	264
6.8 References	269
7. ADAPTIVE ADJUSTMENT OF THE DETECTOR AND THE CHANNEL ESTIMATION PROBLEM	
7.1 Introduction	286
7.2 Channel Estimation using the Mean-Square-Error Criterion	286
7.3 Adaptive Decision-Feedback Equalisation	294
7.4 Combined Adjustment of the Linear Pre-Filter, Channel Estimator and Cancellation Equaliser	297
7.5 Performance of the Adaptive Adjustment System	302
7.6 Alternative Adaptive Adjustment Schemes	307
7.6.1 Alternative Scheme 1	310
7.6.2 Alternative Scheme 2	311
7.6.3 Alternative Scheme 3	320
7.6.4 Determination of the Complex Roots of Complex Polynomials	327
7.6.5 Alternative Scheme 4	329
7.7 References	339
8. MODEM SYNCHRONISATION	
8.1 Introduction	363
8.2 Carrier Phase and Frequency Offset Tracking	364
8.3 Carrier Phase Jitter	374
8.4 Timing Recovery	381
8.5 Automatic Gain Control	390
8.6 References	395

9. MODEM DESIGN AND IMPLEMENTATION AND SUGGESTIONS FOR
FURTHER INVESTIGATIONS

9.1	Introduction	426
9.2	Transmitter Design	427
9.3	Receiver Design	436
9.3.1	The Multiplier-Accumulator (MAC)	438
9.3.2	Processor Bus Structure	440
9.3.3	Data Transfer Instructions	441
9.3.4	Arithmetic Instructions	443
9.3.5	Extended MAC Architecture	444
9.3.6	Alternative Methods for Receiver Implementation	446
9.4	Recommendations for Further Investigations	448
9.5	References	450
10.	CONCLUSIONS	474

APPENDICES

APPENDIX 1 :	Bandpass Signals and Systems and the Equivalent Baseband Model	476
APPENDIX 2 :	Signal and Noise Powers in the Equivalent Baseband Model	481
APPENDIX 3 :	Signal-to-Noise Ratio Definitions and Calculations	484
APPENDIX 4 :	Results of Tests on Coding Scheme 3	490
APPENDIX 5 :	Error Probabilities for 64-Point and 256-Point QAM Signal Structures	498
APPENDIX 6 :	Digital Filter Design	518
APPENDIX 7 :	Selection of Switched-Capacitor Filters	529
APPENDIX 8 :	Relationship between Continuous and Discrete Convolution	536
APPENDIX 9 :	The Extended Newton-Raphson Root-Finding Algorithm	541
APPENDIX 10:	Listings of Computer Programs	549

ABSTRACT

The thesis is concerned with the transmission and reception of digital data at 19,200 bit/s over voice-frequency telephone channels. Following a feasibility study based on both practical and theoretical constraints, the telephone network itself is investigated to determine methods of circuit characterisation and the causes and effects of distortion and other signal impairments. Quadrature amplitude modulation (QAM) is then presented as the most suitable modulation method for this application; rectangular signal constellations, carefully designed to include differential and Gray coding properties, are developed for both 64-point and 256-point QAM structures. The use of equivalent baseband models is also investigated in depth to derive a suitable model for the simulation of QAM systems using digital computers.

The characteristics of three test channels introducing varying amounts of distortion are given along with the development and design details of two sets of practically realisable equipment filters. The results of extensive computer simulation studies for various combinations of signal constellations, equipment filters, test channels and detection processes are then presented. In particular, an adaptive near-maximum-likelihood detector is developed and its performance compared with a more conventional detector, the comparisons being based on the tolerance to additive Gaussian noise and signal distortion.

Following a discussion on the complexities of the detectors, several new techniques for the adjustment of adaptive detectors are introduced and analysed; the new methods involve the theories of complex polynomials and the determination of the complex roots of such polynomials, which are studied in depth.

Finally, the practical problems of receiver synchronisation and the technological problems of system implementation are considered for a synchronous serial 19,200 bit/s modem.

ACKNOWLEDGEMENTS

The Author would like to express sincere thanks to the project supervisor, Professor A P Clark, for his advice, assistance and unbounded enthusiasm.

The financial support of the Rutherford and Appleton Laboratories is gratefully acknowledged.

Thanks also to Ann and Marilyn, for their excellent typing, and to Louise for her patience.

GLOSSARY OF SYMBOLS AND ABBREVIATIONS

$a(t), A(f)$:	Impulse response and transfer function of combined transmitter filter
A_i, B_i	:	Sample values of recovered timing waveform
$b(t), B(f)$:	Impulse response and transfer function of combined receiver filter
$b_{in}(t)$:	Source binary data signal
$b_{out}(t)$:	Output binary data signal
B	:	Bandwidth, in Hertz
$c(t), C(f)$:	Impulse response and transfer function of channel output filter
C	:	Channel Capacity, bit/s
$ d_k ^2$:	Incremental cost function
d_i	:	Components of sampled impulse response of the pre-filter
d_i^r	:	Components of reversed-order sequence corresponding to d_i
$D(z)$:	z -Transform of sampled impulse response of the pre-filter and channel
$D^R(z)$:	z -Transform of reversed-order sequence corresponding to $D(z)$
e_i	:	Error signal at time $t = iT$
$e_{i,j}$:	Update parameter for root-finding algorithm
E_a	:	Energy of the sequence $\{a_i\}$
$f(x)$:	Function of z , usually a polynomial in x
f_c	:	Carrier frequency in Hertz
f_s	:	Sampling frequency
f_r	:	Frequency of symmetry in sinusoidal roll-off filters
$F^{-1}(\cdot)$:	Inverse Fourier Transform of (\cdot)
F	:	Vector containing components of the sampled impulse response of the pre-filter and channel

$F(z)$:	z -Transform of F
f_i^r	:	Components of reversed-order sequence corresponding to f_i
$F^R(z)$:	z -Transform of reversed-order sequence corresponding to $F(z)$
$g(t), G(f)$:	Impulse response and transfer function of the pre-modulation transmitter filter
g	:	The sampled impulse response, Y , is assumed to have $(g+1)$ components
Gain_i	:	AGC amplifier gain control signal
$h(t), H(f)$:	Bandpass impulse response and transfer function of the transmission path
$\hat{h}(t), \hat{H}(f)$:	Baseband versions of $h(t), H(f)$
j	:	$j = \sqrt{-1}$ when not used as a subscript
m	:	Signal alphabet
n	:	Number of consecutive samples used in the detection process. Also, delay introduced by a detector
$p(t), P(f)$:	Impulse response and transfer function of the post-demodulation low-pass filters
p_i	:	Input sample to pre-filter at time $t = iT$
Prime, $'$:	Estimate of ; for example, Y' is the estimate of Y . Also, s_i' is the detected data symbol at time $t = iT$.
$P_u(f)$:	Noise power spectral density of the random process $u(t)$
$q(t), Q(f)$:	Impulse response and transfer function of the channel input filter
$r(t)$:	Received signal after demodulation and filtering
r_i	:	Sample of $r(t)$ at time $t = iT$.
$R(\tau)$:	Autocorrelation function
s_i	:	The i^{th} transmitted symbol value

(x)

$s(t)$:	Continuous-time input baseband signal
T	:	Symbol period. Symbol rate is $1/T$ baud
$U_i(z)$:	Deflated form of the polynomial $Y(z)$ after removal of i roots
$u(t)$:	Gaussian random processes with zero mean values and stated variance
$v(t)$:	
$w(t)$:	
$n(t)$:	
u_i	:	Sample values of $u(t)$, $v(t)$, $w(t)$ and $n(t)$ at times $t = iT$
v_i	:	
w_i	:	
n_i	:	
$\{x_i\}$:	The sequence obtained by sampling $x(t)$ at times $t = iT$
$y(t)$:	Impulse response of the baseband channel
Y_i	:	Row vector containing the sampled version of $y(t)$ at time $t = iT$
Y_i	:	$i^{\text{th}} + 1$ sample of Y
$Y(z)$:	z -Transform of Y
$Y^R(z)$:	z -Transform of reversed sequence corresponding to $\{y_i\}$
$Z^{-1}(\cdot)$:	Inverse z -Transform of (\cdot)
$\frac{1}{2}N_0$:	Two-sided noise power spectral density of additive white Gaussian noise
α	:	Number of stored vectors in NMLH detector
α_i	:	Root (zero) of a complex polynomial
β_i	:	Root (zero) of a complex polynomial
γ	:	Number of expansions of a stored vector in a NMLH detector
γ_i	:	Root (zero) of a complex polynomial
Δf	:	Frequency offset, in Hertz

$\delta(t)$:	The Dirac impulse function
ϵ_i	:	Update parameter for root-finding algorithm
λ_n	:	Approximation to root β_i after n iterations of root-finding algorithm
σ^2	:	Variance of a random process
$\phi(t)$:	Time varying phase
ϕ_i	:	Phase at time $t = iT$
ϕ_0	:	Constant phase shift or error
$\phi_j(t)$:	Time varying phase caused by phase jitter
$\psi_u(\tau)$:	Autocorrelation function of $u(t)$
$*$:	Convolution operator
\star	:	Discrete convolution operator
∇	:	Gradient operator

AGC	:	Automatic Gain Control
DTI	:	Data Transfer Instruction
FDM	:	Frequency Division Multiplexing
HF	:	High Frequency
ISB	:	Independent Side-Band
Im	:	Imaginary part
MAC	:	Multiplier-Accumulator
NLE	:	Non-Linear Equaliser
NMLH	:	Near-Maximum-Likelihood (Detector)
PSTN	:	Public Switched Telephone Network
QAM	:	Quadrature Amplitude Modulation
Re	:	Real part
SIR	:	Sampled Impulse Response
SNR	:	Signal-to-Noise Ratio
SSB	:	Single Side-Band
VSF	:	Vestigial Side-Band

1 Introduction

1.1 Background

The telephone network has for many years provided subscribers with a relatively cheap medium for the communication of low-rate data between remote sites, whether using the flexibility of the public switched telephone network (PSTN) or the more expensive dedicated private circuits. For example, a subscriber may purchase a simple 600-1200 baud frequency-shift-keyed (FSK) modulator-demodulator (modem) unit for around £150 whilst incurring only the standard rental and call charges of any other customer.

The field of digital data communications has, like all other areas in the Communications Industry, benefited greatly from the rapid advances in microelectronic and computer technologies. The past few years have witnessed many advancements in the design, implementation and use of modems, from the early days of simple ON-OFF keyed systems, to modern devices operating reliably at rates of up to 9600 bits per second (bit/s) which use very sophisticated modulation and detector structures. As well as providing the impetus for actual design advancements, the new technologies have also provided extra markets for the final product. An obvious example of this is the "personal-computer" market; from the very humble beginnings of early home computers with limited memory and single connections to monochrome television receivers, the very latest machines feature full colour graphics, disc storage facilities and so on. More importantly in the context of the investigation, is the ability and the desire to join computers together using some form of network, for example, the PSTN. The "information explosion" and the general awareness of managers to their companies' data processing and telecommunication requirements have also played their part in producing a very lucrative, but

highly competitive market for suppliers offering faster, cheaper modems for use over the existing telephone network.

At the time of this investigation, state-of-the-art modems for use over telephone circuits operate at 14,400 bit/s in America and around 16,000 bit/s in the U.K. Although little information is available regarding the performance of these modems in terms of their tolerance to typical levels of noise and distortion introduced by the telephone network, it is apparent that they both operate using well established modulation and detection methods but fail to achieve reliable communications over anything but the better quality telephone circuits.

The aim of the investigation presented in this thesis is to develop suitable modulation and detection methods to enable the reliable transmission and reception of digital data at 19,200 bit/s over telephone circuits typically encountered in practice and to compare the performance of the developed systems with more conventional solutions. Where possible, the development of the modem is discussed in a practical manner with the intention of providing the basic design strategy for a commercially competitive high-speed modem.

1.2 Outline of the Investigation

Chapter 2 provides an introduction to the terminology and general requirements of digital data communication systems. A simple system model is presented and described along with an intuitive introduction to equivalent models of bandpass processes. System parameters such as bandwidth, information and signalling rate, signal alphabet and signal-to-noise ratio are then introduced and their relationships explained. The feasibility of data transmission at 19,200 bit/s is investigated with regard to theoretical limits and the compromises necessary for practical system development.

Chapter 3 discusses the nature of telephone circuits and the types of signal distortion and other impairments introduced by these circuits. The possible types of connection are considered (i.e. switched connections and private circuits) followed by methods of circuit characterisation. The contribution to signal distortion and levels of signal impairments of individual links comprising a connection are discussed as is the problem of mismatch between the links. The influence of frequency and amplitude modulated impairments on data transmission systems is then considered, along with the types and effects of additive noise. Finally, "typical" characteristics for both private and switched circuits are presented along with the internationally agreed power and bandwidth restrictions for non-speech apparatus connected to telephone networks.

Chapter 4 describes the relationships between bandpass processes and signals with their equivalent baseband representation in a more rigorous manner and supports the suggestion of using an equivalent baseband model. The suitability of quadrature amplitude modulation (QAM) is discussed and the system model is refined to include the case of complex signals operating over complex baseband channels. Two-dimensional signal sets or constellations are then considered in detail for both 64-point, 3200 bauds and 256-point, 2400 bauds systems capable of transmitting data at 19,200 bit/s. The desirability of differentially and Gray coded structures is discussed and four coding schemes are derived; coding schemes 3 and 4 (64-point and 256-point structures, respectively) being the most suitable for the current application.

Chapter 5 considers the requirements of the system in terms of the responses of the various filtering processes in both the transmitter and receiver units. After discussing the advantages of using digital filtering techniques, the characteristics of two sets of equipment filters, known as

filter sets 1 and 2, are presented which provide a compromise between the ideal filters and those which satisfy the spectral restrictions imposed by channel operators. The various filters comprising filter sets 1 and 2 are considered separately to ensure conformation with the above restrictions and conjointly to determine the characteristics of the overall filter sets. The characteristics of three test telephone circuits, circuits 1-3, are then presented which cover the range from "typical-good" to "typical worst-case" characteristics likely to be encountered on both switched and private connections. Finally, the overall characteristics of the combinations of filter sets 1 and 2 and the three telephone circuits are derived and the sampled impulse responses of these combinations are determined. The overall sampled responses are presented at both 3200 and 2400 samples per second in preparation for computer simulation studies of 64-point and 256-point QAM systems respectively, using equivalent baseband models.

Chapter 6 uses the results of the previous two chapters to form a suitable baseband model for the simulation of the system when operating with different combinations of filter sets, telephone circuits and detection methods. The chapter considers two main types of data detector, namely equalisation and maximum likelihood detection and discusses two subsets of these types, non-linear equalisation and near-maximum likelihood detection, in detail. The application of the Viterbi algorithm to the latter detector is then introduced, leading to the derivation of a hybrid scheme using both decision-feedback equalisation and a reduced-state Viterbi detector. The results of extensive computer simulation studies are then given which compare the performance of 64-point and 256-point systems operating over various baseband channels with the more conventional optimum non-linear equaliser. Finally, the complexity of

the selected detection scheme is investigated in terms of computational effort and amount of storage required to implement the detector.

Chapter 7 investigates the adaptive adjustment of the data detector and the channel estimation problem. To be capable of operation over different telephone channels, the detector must be adaptive in the sense that it takes full account of distortion introduced by the channel. The detector must therefore be able to estimate the sampled impulse response of the channel and to use this estimate to adjust adaptive filters internal to the detection scheme.

The chapter first discusses the adaptive estimation of the sampled impulse response of a channel and then extends the discussion to include the adaptive adjustment of the equalisers used in the selected detection scheme. A combined adjustment system is then presented for use in a near-maximum likelihood detector. Finally, four new methods, involving the manipulation of complex polynomials and the determination of the complex roots of such polynomials, are introduced and compared with a more conventional approach in terms of computational efficiency and speed of operation.

In Chapter 8, the problems of synchronising a synchronous serial modem are considered. The discussion includes the need for synchronisation, the effects of synchronisation imperfections and possible solutions for carrier-phase tracking, symbol timing recovery/correction and automatic gain control (AGC). The performance of the various systems are then studied using computer simulation techniques.

Finally, Chapter 9 considers the actual design and implementation of a 19,200 bit/s modem. The transmitter unit is first considered and a fairly simple hardware-based solution is presented. Due to the complexity of the receiver unit, the use of a programmable digital signal processor

is recommended; several possible methods of realisation are discussed although detailed system designs are not included.

2. General Considerations for Digital Data Transmission Systems.

2.1 Introduction

This chapter aims to provide an introduction to the general requirements and potential shortcomings of digital data communication systems operating over telephone circuits. A simplified system block diagram is first presented and described followed by a more detailed model which illustrates the various component parts of the system including a generalised introduction to the concept of the complex communication channel. The various system parameters, such as bandwidth, information and signalling rates, signal alphabet and their relationships are then introduced and discussed along with an explanation of the causes of signal distortion and additive noise and how these impairments force the use of quite sophisticated receiver and detector structures. Finally, the feasibility of transmitting data at 19,200 bits per second is investigated with particular reference to theoretical limits and the more practical compromises that a system designer is forced to make between performance in the presence of signal distortion and additive noise, signal element rate, system bandwidth, signal alphabet and signal-to-noise ratio.

2.2 Model of a Data Transmission System

A simplified model of a general data transmission system operating over an analogue channel is shown in Figure 2.2.1. The system is assumed to be operating with a serial transmission format and with the transmitter and receiver in element synchronism⁽¹⁾.

The source binary data signal, $b_{in}(t)$ is assumed to be supplied to the converter block at a rate of 19,200 bits per second, (bit/s). Throughout this investigation the primary input to the data communication system will be considered as a continuous stream of binary digits with an information

rate of 19,200 bits per second, the system itself being unconcerned with the actual meaning of the data nor with any pre-coding, scrambling or pre-framing that might be conditioned onto the raw source data by the user. Similarly, the receiver will deliver a continuous stream of binary data at a rate of 19,200 bit/s to the destination, the end-user being responsible for all inverse coding, framing, etc. The overall aim of the system is, therefore, to provide reliable data communications in a fully transparent manner.

The baseband binary data signal, $b_{in}(t)$ is applied to a converter whose principle task is to match the spectral characteristics of the communication channel with those of a signal suitable for transmission. However, due to the position of the converter within the system topology, the converter block could also include any source encoding, encryption, channel coding and multiplexing required by the system as a whole⁽²⁾.

The converted (or perhaps more generally, encoded) signal $s(t)$ is then applied to the transmission channel which for the purpose of this investigation, will be considered a bandpass channel with an approximate bandwidth of 300 Hz to 3400 Hz. The channel, as well as exhibiting bandpass characteristics and hence possibly introducing attenuation distortion, can also introduce delay distortion, noise of various types and levels, frequency offsets due to the use of carrier-based systems within the transmission path and both random phase perturbations (phase noise) and the more regular fluctuations of signal phase caused by phase jitter⁽¹⁾; Chapter 3 considers line characteristics and impairments in more detail.

The output signal from the channel, that is, the input signal to the receiver, $r(t)$, will in general be a distorted and noise corrupted version of the channel input signal, $s(t)$. This noisy, distorted signal is applied to the detector block which operates on the received signal in some non-

linear fashion to yield the detected signal, $\hat{s}(t)$, which is then decoded in the receiver converter to give the output data signal, $b_{out}(t)$. Provided the detector makes the correct decision when determining $\hat{s}(t)$ from $r(t)$ and assuming the two converters act as an inverse-pair, the output data should be identical to the input data, albeit with some constant time shift due to encoding and propagation times.

A slightly more detailed system block diagram is given in Figure 2.2.2, where, as before, the input to the communication system is the sequence of binary digits, $\{b_{in}\}$, where,

$$b_k = b(k\tau) = 0 \text{ or } 1 \quad (2.2.1)$$

τ being the digit duration,

k being a positive integer.

Since the binary data arrives from the source at a rate of 19,200 bits per second, then;

$$\tau = \frac{1}{19,200} \approx 52.1 \mu s \quad (2.2.2)$$

Further, the members of the sequence $\{b_{in}\}$ are statistically independent and equally likely to have either binary value. This input sequence is applied to the encoder (Figure 2.2.2) which operates on it to produce a corresponding sequence of data symbols, $\{s_i\}$, where;

$$s_i = \text{symbol value at time } t = iT \quad (2.2.3)$$

T being the symbol duration

i being an integer

The basic shape and form of each s_i is called a signal element and so each signal element will convey one data symbol, the actual relationship between the $\{b_{in}\}$ and $\{s_i\}$ being dependent on;

- a) the rate of transmission of symbols along the channel,
- b) extra coding imposed by the modulation scheme, for example, differential and Gray coding discussed further in Chapter 4.

The $\{s_i\}$ will in general be complex valued quantities and will be able to take on any one of a given set of values for an element period, the set being known as the signal alphabet, the number of members of which determine the number of "levels" attainable, ("level" here meaning not just simply amplitude, but the number of different values that the $\{s_i\}$ may take).

In general,

$$s_i = a_i + jb_i \quad (2.2.4)$$

where $j = \sqrt{-1}$

a_i = real part of the two dimensional symbol

b_i = imaginary part of the two dimensional symbol

the data symbol occurring at a time $t = iT$, where T is the symbol interval and so,

$$\text{signalling rate} = \frac{1}{T} \text{ bauds} \quad (2.2.5)$$

It is also assumed that the $s_i = 0$, for all $i \leq 0$, so that the data symbol at time $t = iT$, s_i , will also be the i^{th} symbol transmitted.

Since the required information rate (and therefore the input/output binary data rate) is 19,200 bit/s and since the systems signalling rate is

given in equation (2.2.5), the numerical relationship, but not the actual coding relationship, can be readily established;

$$\frac{\text{Information rate}}{\text{Line signalling rate}} = \text{Number of bits of information per symbol} \quad (2.2.6)$$

$$\text{Hence, } 19,200 \times T = \text{Number of bits per symbol} \quad (2.2.7)$$

The result of expressions (2.2.6) and (2.2.7) is often called the coding efficiency of the bit-to-symbol encoding process. For example, if the line signalling rate selected was 2400 bauds, then each data symbol would correspond to 8 bits of the input binary data. It is important to note therefore, that extra coding may be required to prevent all 8 bits of information being in error if a single data symbol is erroneously detected. For this reason, it is usual to measure the performance of digital data systems in terms of bit error rate (B.E.R.) rather than the symbol error rate, giving an indication of the system's tolerance to noise, interference and distortion. Table 2.1 lists some other possible bit-to-symbol coding arrangements for the case of 19,200 bit/s transmission, given that equations (2.2.6) and (2.2.7) result in integer values.

The output from the encoder block, Figure 2.2.2, will consist of a sequential stream of data symbols, $\{s_i\}$, which will be statistically independent and equally likely to have any of the possible values within the set defined by the signal alphabet. This serial symbol stream is then applied to the input of the baseband channel which comprises the totality of all transmitter pulse-shaping and noise rejection filters, one or more linear modulators to permit the efficient transmission of the baseband signal over the bandpass telephone circuit, one or more linear demodulators and the totality of all filters present in the receiver unit, where receiver unit is here defined as all equipment between the telephone

circuit output and the point of emergence of the received baseband signal. Figure 2.2.3 shows in block diagram form the constituents of the bandpass channel described above. The modulation/demodulation process is assumed to occur in an ideal manner and so is considered a linear process introducing no distortion to the transmitted signal.

For the system under consideration here, the transmission circuit will be bandpass in form, the total circuit being made up of several possible communication sub-systems in series; for example, the circuit might well comprise links using carrier-telephony and/or microwave systems. Consequently, as well as the bandpass channel itself not being suited to the direct transmission of the baseband data symbols, the transmission circuit over which the data is to be communicated can also introduce frequency translation effects which could prevent the successful transmission of baseband signals unless the amount of frequency shift can be accurately determined and corrected at the receiver. The use of a suitably positioned (in frequency) analogue carrier waveform which is modulated by the baseband data symbols will alleviate these problems by;

- (a) Matching the spectral properties of the baseband signal to the transfer function of the baseband channel, and
- (b) Correcting for any internal frequency translations that occur within the transmission circuit, the correction being achieved by adjusting the frequency of the demodulator's carrier signal. If a carrier component is present in the received modulated signal, a phase-locked-loop can be employed to track the variations of frequency introduced by system offsets but for suppressed-carrier signals, the carrier will have to be recovered from the received signal itself. Chapter 8 addresses the problem of carrier recovery and synchronisation in some detail.

The choice of carrier frequency depends to a large extent on actual system design and implementation as discussed in Chapter 9, but for the bandpass characteristics of typical telephone circuits, the selected frequency must be around 1800 Hz such that the mid-point of the (symmetrical) spectrum of the modulated signal lies approximately half-way between the limits of the channels passband. Figure 2.2.4 illustrates this requirement in relation to a hypothetical baseband signal spectrum and the amplitude characteristic of a typical telephone channel.

The output from the baseband channel, $r(t)$ in Figures 2.2.2 and 2.2.3, will be a noise corrupted and possibly distorted version of the channel's input signal, the distortion being a consequence of the channel's bandpass characteristics and so the received symbols could suffer from intersymbol interference where adjacent symbols spread-out in time and mutually interfere with each other. It should also be realised that for the case of complex-valued signal elements, two modulators, demodulators and associated modulator/demodulator filters will be required. If the two modulators operate with the same carrier frequencies but in phase quadrature, the baseband channel itself may be considered complex, the real part of the channel carrying the real components of the $\{s_i\}$ and the quadrature channel carrying the imaginary components of the $\{s_i\}$. Consequently, channel distortion can not only result in intersymbol interference between the elements in each individual channel, but can also cause interchannel interference in the form of crosstalk between the real and imaginary data streams. The quadrature channel model and a rigorous proof of this model are further discussed in Chapter 4 and Appendix 1, whereas the major causes and effects of intersymbol interference and interchannel interference are considered in more detail in Chapter 3. For the purpose of illustration, Figure 2.2.3 shows the

addition of a second modulator/demodulator configuration which can be assumed to be operating in phase quadrature with the upper part of the diagram over the complex transmission circuit.

Since the data symbols are to be transmitted over the baseband channel comprising the actual transmission path and equipment filters in series, the amount of distortion introduced by the channel can be estimated from a knowledge of the overall amplitude and phase characteristics or from the complex transfer function which provides both amplitude and phase information. Given that a system's transfer function and impulse response are related via the Fourier Transform pair, then if;

$H_1(f)$ = Complex transfer function of all transmitter filters,

$h_1(t)$ = Impulse response of all transmitter filters,

$H_2(f)$ = Complex transfer function of all receiver filters,

$h_2(t)$ = Impulse response of all receiver filters,

$C(f)$ = Complex transfer function of the transmission circuit,

$c(t)$ = Impulse response of the transmission circuit

the overall baseband channel transfer function will be given by;

$$Y(f) = H_1(f) \cdot H_2(f) \cdot C(f) \quad (2.2.8)$$

where $Y(f)$ = Complex transfer function of the overall baseband channel.

By using the convolution theorem and the inverse Fourier Transform, the impulse response of the overall baseband channel may be written as;

$$y(t) = F^{-1}[Y(f)] = h_1(t) * h_2(t) * c(t) \quad (2.2.9)$$

$$F^{-1}[Y(f)] = \text{Inverse Fourier Transform of } Y(f)$$

and * denotes the convolution operator.

Furthermore, if the input data symbols can be represented by;

$$s_i \delta(t-iT) \quad (2.2.10)$$

where equation (2.2.10) implies a stream of impulses occurring at times $t = iT$ and with strengths equal to the $\{s_i\}$, then assuming the necessary change to the transmitter filter response has been made in order to permit the representation of the data symbols as impulses, the i^{th} received signal element at the output of the baseband channel will be:-

$$s_i y(t-iT) \quad (2.2.11)$$

It follows that the output signal, $r(t)$ in Figure 2.2.2, in the absence of noise, will be given by;

$$r(t) \underset{\text{(no noise)}}{=} \sum_{i=0}^{\ell} s_i y(t-iT) \quad (2.2.12)$$

where ℓ is some large positive integer.

The resultant signal $r(t)$ is then sampled once per symbol at times $t = iT$, producing the sequence of received samples, $\{r_i\}$. The convolution of the data symbols with the channel impulse response and the sampling operation described above can be combined by considering the sampled impulse response of the channel; the use of the sampled impulse response not only makes the model easier to interpret but more importantly, it allows the direct simulation of the model by a digital computer since only discrete-time (sampled) quantities are required.

The sampled impulse response of the system under consideration may be theoretically obtained by applying a single impulse at time $t = 0$ to the input of the baseband channel. From equation (2.2.10), the output of the channel, after sampling, will be:-

$$\sum_{i=0}^{\ell} y(t) \delta(t-iT) = y_0 \delta(t) + y_1 \delta(t-T) + y_2 \delta(t-2T) + \dots y_{\ell} \delta(t-\ell T) \quad (2.2.13)$$

which is a sequence of impulses whose strengths are given by the sample values of $y(t)$ at the times $t = iT$. Provided that instantaneous sampling is assumed to occur, that is, a train of impulses is used for the sampling function or alternatively if the rate of change of the waveform being sampled is very small during the actual sampling operation, the sequence of impulses given in equation (2.2.13) may be more simply written as the sequence of sample values;

$$Y_0 \ Y_1 \ Y_2 \ \dots \ Y_{\ell} \quad (2.2.14)$$

For convenience of notation, any constant delay in transmission over the baseband channel is neglected. Furthermore, if it is assumed that $y_i = 0$ for $i < 0$ and $i > g$, equation (2.2.14) may be considered as the $(g+1)$ -component row vector given by;

$$Y = [Y_0 \ Y_1 \ Y_2 \ \dots \ Y_g] \quad (2.2.15)$$

where Y is the vector whose components define the sampled impulse response of the baseband channel when sampled at times $t = iT$ and the y_i $i = 0$ to g , are those sample values, which in general will be complex. The value of g , where $g \leq \ell$, determines the overall length of the sampled impulse response and so is dependent on the characteristics of the particular channel being investigated. However, in general the value of g will provide a good indication to the amount of distortion introduced by the channel in that a channel which has a sampled impulse response exhibiting

non-zero elements either before or after the component with the highest sample value will introduce varying degrees of intersymbol interference, the severity of the interference depending on the number and relative values of the non-zero components.

Equation (2.2.12) may now be written in terms of sampled signals;

$$r_i = \sum_{h=0}^g s_{i-h} y_h \quad (2.2.16)$$

where $r_i = i^{\text{th}}$ received sample at the output of the channel

$s_i = i^{\text{th}}$ data symbol transmitted

y_h = sample values of the channel's impulse response..

In an ideal system, the output from the baseband channel will be identical to the input and so $\{r_i\} = \{s_i\}$ for all i . However, this situation will only exist if, from equation (2.2.16)

$$y = [1 \quad 0 \quad 0 \quad \dots \quad 0] \quad (2.2.17)$$

A sampled impulse response of this form is called ideal and describes the special case of an ideal channel. It should be noted that a sampled impulse response of the form;

$$y = [0 \quad 0 \quad \dots \quad 1 \quad 0 \quad 0 \quad \dots] \quad (2.2.18)$$

will also define an ideal channel, albeit one which introduces a constant delay in transmission of n sample intervals, where n is the number of zero-valued components in the sampled response occurring before the main component. It is for this reason that the assumptions made earlier, that is, $y_i = 0$ for $i < 0$ and $i > g$, do not affect the performance of the system apart from ignoring a constant transmission delay. Figure 2.2.5 illustrates the above by considering two adjacent signal elements and a

channel with an ideal sampled impulse response both with and without zero-valued pre-cursors.

It follows that if the channel can be considered ideal, the task of designing data modems would be considerably simplified since the designer would not be concerned with signal distortion. In practice however, the designer is not often confronted with an ideal channel. Clearly, this is not the case for telephone channels whose amplitude characteristics are severely bandlimited although some wideband satellite channels do approach the ideal in as much as they may possess flat characteristics over a bandwidth greater than that required by the transmitted signal's spectrum. Unfortunately, such satellite systems may also suffer from non-linear distortion introduced by the saturation of high-power amplifiers, which further complicates the designer's task.

A typical telephone channel will not possess an ideal sampled impulse response for two reasons; the transmission path itself will have a bandpass characteristic and also the transmitter and receiver equipment filters, including all shaping and noise-rejection filters, will further modify the characteristics of the overall baseband channel^(1,2,4).

If the amplitude response of the overall channel is not constant over the bandwidth occupied by the signal spectrum and/or the phase response of the channel differs either from a constant value or from a characteristic which has a constant slope across the required bandwidth, then intersymbol interference will occur due to the spreading and merging of adjacent signal elements^(1,5). The condition $\{r_i\} = \{s_i\}$ for all i will no longer apply since each received sampled, r_i , will now contain not only the wanted symbol, s_i , but also fractions of previously transmitted symbols. For example, if the sampled impulse response of a baseband channel is given by;

$$y = \begin{bmatrix} 1 & 0.2 & 0.1 \end{bmatrix} \quad (2.2.19)$$

then, from equation (2.2.16);

$$r_i = s_i + 0.2s_{i-1} + 0.1s_{i-2} \quad (2.2.20)$$

As the length of the sampled impulse response increases, the amount of intersymbol interference will also increase so distorting the received samples even further. The data detector which operates on the received samples to yield the detected data symbols, now has the more difficult task of detecting a given symbol, say s_i , from the severely distorted received sample r_i . It is for this reason that when attempting to transmit large amounts of data over bandlimited communication channels of a given bandwidth, extremely sophisticated detector stages must be included to extract the required data symbol from the received samples.

As well as containing distortion, the received signal $r(t)$ and hence the received samples, $\{r_i\}$ will also be corrupted by noise (see Figure 2.2.2). Whereas the effects of interference and distortion can, at least in theory, be reduced or eliminated by careful design, noise of one form or another will always be present in any communication system not operating at a temperature of absolute zero^(6,7). The noise may be due to many sources and be of many types but in general, additive noise will be present over both switched and private telephone channels and will be impulsive in nature^(1,2). It has been shown^(1,9), that the tolerance of digital data systems to white Gaussian noise gives a good indication to their tolerances to impulsive noise, even though white Gaussian noise does not necessarily model the actual noise present in a given system. Furthermore, since Gaussian noise may be readily generated and analysed, the tolerance of data systems to impulsive noise can be deduced by modelling and simulating the system on a digital computer; the noise used in the simulations being white Gaussian noise with a zero mean and known 2-sided power

spectral density which is added to the received signal at the output of the transmission path before the receiver filter.

With reference to Figures 2.2.2 and 2.2.3, the output from the receiver filter will therefore be;

$$r(t) = \sum_{i=1}^{\ell} s_i y(t-iT) + w(t) \quad (2.2.21)$$

where $w(t)$ is white Gaussian noise with a zero mean and two-sided power spectral density of $\frac{1}{2} N_0$ W/Hz.

The sampled version of equation (2.2.21) will therefore be;

$$r_i = \sum_{h=0}^g s_{i-h} y_h + w_i \quad (2.2.22)$$

where the $\{w_i\}$ are sample values of a Gaussian random variable with zero mean and fixed variance. Due to the passage of the noise through the receiver filter, adjacent noise samples may be correlated to some degree^(9,10). The resultant noisy and distorted received samples, $\{r_i\}$, are then fed to the signal processing and detector block in Figure 2.2.2, where a decision has to be made as to the values of the $\{s_i\}$. Depending upon the complexity of the detector, it may either operate directly on a single received sample or on several received samples before producing the detected data symbols, $\{s'_i\}$, as its output^(1, 8-15). Again depending upon system complexity, there may well be a delay in the detection process such that at time $t = iT$, the detector output will be s'_{i-n} , where n is the detection delay in multiples of T seconds and s'_{i-n} corresponds to the input sample r_{i-n} . Clearly, a detection process which involves several received samples will inevitably introduce some delay in detection although this is not of great consequence since channel set-up time and initial synchronisation will in general be orders of magnitude higher⁽⁹⁾.

The principle function of the signal processor and detection block is therefore to output a symbol value, s_i' which, as far as the particular detector is concerned, most resembles the transmitted data symbol, s_i . However, the detector must base its decision only on the information contained within the received samples $\{r_i\}$ which have been shown to be both distorted and corrupted by additive noise. Clearly, a "good" detector is one which can supply the correct output even when the received samples are severely distorted and at the same time, be very tolerant to the levels of additive noise experienced on the given channel. The three main methods of detecting data symbols from noisy and distorted digital signals are ^(5,8);

- (a) Linear equalisation
- (b) Non-linear or decision feedback equalisation
- (c) Maximum likelihood detection.

Although these methods will be re-visited in some detail in Chapter 6, there follows a brief description of each method to illustrate the complexities of detector design and to help emphasise the importance of the detector in a digital data system.

The linear equaliser attempts to remove the effects of distortion by further filtering the received data signal, the filter or equaliser being adjusted such that the overall sampled impulse response of the baseband channel and equaliser approaches the ideal response shown in equation (2.2.17) or (2.2.18).

So far, the discussion of the baseband channel has assumed a non-time varying response. However, it should be realised that most communication channels do in fact have characteristics which vary with time. Some systems, for example H.F. radio links and land mobile radio systems, suffer from severe fades due to multipath propagation ^(5,16) whereas other

systems, including most telephone circuits, vary only very slowly with time. It is also important to make clear the distinction between switched telephone circuits and non-switched (or private) circuits. Digital data systems which operate over the public switched telephone network (PSTN) will in general communicate over different physical connections; the actual routing through the network depending upon the loading of the system at the time of the "call". Consequently, the channel characteristics and therefore the sampled impulse response of the channel will vary from connection to connection. Private circuits (also called leased lines and private wires) do not pass through any switching stages and so can be thought of as fixed, hardwired connections between the source and destination. However, whether due to maintenance, development of fault conditions or mutual coupling between circuits as they enter and leave terminating stations, the characteristics of private lines and of switched lines once the connection has been established, do vary with time. Clearly, if the detector stage of a data system employs an equaliser or any other device which requires a knowledge of the sampled impulse response, the detector must be made adaptive to enable it to cope with any time variations of the channel characteristics.

A non-linear equaliser usually comprises a linear pre-filter followed by a non-linear equaliser which together attempt to cancel the effects of distortion by subtracting an estimate of the intersymbol interference components present in a given received sample. For the example given in equation (2.2.20), a non-linear equaliser will subtract $(0.2s_{i-1} + 0.1s_{i-2})$ from the received sample r_i to leave $r_i = s_i$. Figure 2.2.6 illustrates the basic principles of linear equalisation and non-linear equalisation. It should be noted that the latter method is non-linear by virtue of the detector (a non-linear device) being included in a feedback loop.

Provided the equaliser, whether linear or non-linear in form, has cancelled the effects of distortion sufficiently, its output will comprise sample values which are related to single transmitted symbols plus, of course, noise. That is, the output from the equaliser will be of the general form⁽⁸⁾,

$$x_{i+h} = s_i + u_{i+h} \quad (2.2.23)$$

where x_{i+h} is the equaliser output at time $t = (i+h)T$

s_i is the i^{th} transmitted symbol

u_{i+h} is the Gaussian noise component at time $t = (i+h)T$

and h is the equaliser delay in multiples of T seconds.

Due to the noise component in equation (2.2.23), a detector will be required in order to determine which allowable symbol value is most likely being conveyed by the equalised sample, x_{i+h} . Figure 2.2.7 shows the basic operation of a simple threshold detector which, from a knowledge of the expected levels of the received samples in the absence of noise and a knowledge of the signal alphabet, bases its decision on a set of fixed threshold values, the figures giving an example of a 4-level signal where the detector operates on the amplitude of the received sample.

The third detection method, maximum likelihood detection or sometimes called maximum likelihood sequence estimation operates as follows. The receiver waits until the whole message has been received and so it holds in store a set of distorted and noise corrupted received samples. The detector also holds in store all possible combinations of the message (assuming a fixed message length) and convolves each combination with the sampled impulse response of the channel to yield noiseless versions of all possible sequences. It then compares the received sequence with

all of the stored sequences and determines which noiseless version most closely resembles the received sequence, in a minimum mean-square sense. The sequence so selected, in the presence of additive white Gaussian noise, is called the maximum likelihood sequence and is taken to be the best approximation to the data sequence transmitted; the maximum likelihood sequence is therefore the detected data sequence. The basic operation is shown in Figure 2.2.8 where a message length of 4 symbols is assumed to be transmitted over a channel with a real sampled impulse response having only two components. The $2^4 = 16$ possible transmitted sequences which are identical to the 16 sequences held in the detectors store, are given in Table 2.2 along with the 16 possible results from convolving the stored sequences with the channel sampled impulse response. In conditions of no noise, the convolved sequences and the 16 possible received sequences will be identical. The detector now compares each stored sequence with the received sequence, the comparison being based on the calculation of a cost function, $\sum_{i=1}^{n+p-1} (x_i - r_i')^2$ for each of the 16 convolved sequences, where n is the number of symbols in the message, p is the number of components in the sampled impulse response, $\{r_i\}$ is the received sequence and the $\{r_i'\}$ are the convolved sequences. The stored sequence associated with the smallest cost function is the maximum likelihood sequence. Clearly, the cost function associated with the received sequence $[1 \quad 3.5 \quad 2.5 \quad 1.5 \quad 0.5]$ when calculated with the identical stored and convolved sequence will be zero, whereas all other cost functions will yield a non-zero result. Consequently, the stored sequence $[1 \quad 3 \quad 1 \quad 1]$ is taken to be the detected data sequence.

It should be clear that whilst the maximum likelihood detector presented above is a very good detector, it is totally impractical as it stands for use in digital data systems which do not have a block message

structure. It is normal for systems to transmit very long streams of serial data, for example, several hundred-thousand symbols and so the storage requirements and computational effort required at the detector to locate the maximum likelihood sequence would be enormous. However, methods are available to simplify the detector structure and to reduce both the memory requirement and computational power required at the detector^(5,11-17), these methods being studied in detail in Chapter 6.

Before leaving the subjects of equalisation and detection, it should be noted that all three methods presented require a knowledge of the channel's sampled impulse response. Unless the transmission path is high quality, fixed and time invariant, which is not often the case, the receiver unit must also provide a method of estimating the sampled impulse response based only on the received samples and the detected data symbols. Furthermore, if the channel characteristics are expected to vary with time, the estimation process must be highly adaptive in order to track the channel's variations. This further complication to the overall system design is considered in Chapter 7, the whole chapter being devoted to this extremely important facet of modern digital data communication systems.

2.3 19,200 bit/s over Telephone Channels

As explained previously, the information rate (I bit/s) and the symbol rate ($1/T$ bauds) are related by some encoding rule where every symbol will represent $I \times T$ bits of information, (see equations (2.2.6) and (2.2.7)). Also, assuming a signal alphabet of size m , each symbol, where the symbols are statistically independent and equally likely to have any one of m possible values, will contain $\log_2 m$ bits of information^(1,3,6,7). Hence,

$$I.T = \log_2 m \quad (2.3.1)$$

$$\text{giving} \quad T = \frac{1}{I} \log_2 m \quad (2.3.2)$$

$$\text{or} \quad \frac{1}{T} = \frac{I}{\log_2 m} \quad (2.3.3)$$

so the symbol rate is clearly a function of both the required transmission rate and the size of the signal alphabet. Another constraint which is placed upon the symbol rate is due to the Nyquist criterion^(1,3,6,7,10), which states that under given conditions, the maximum symbol rate over a channel having a bandwidth B Hz such that no intersymbol interference occurs, is $2B$ bauds. Such a statement is rather oversimplified in that it assumes an ideal bandlimited channel with a constant group delay response over the channel bandwidth and also that effects such as frequency offsets, phase jitter and any other factors which produce additional signal distortion are negligible. In general, because of the non-ideal characteristics of practical communication channels, operating a system with a symbol rate below the Nyquist rate will result in lower levels of signal distortion compared with systems operating close to the Nyquist rate (see Chapters 3 and 4). However, due to the relationship between symbol rate, information rate and signal alphabet, equation (2.3.3), and the fact that in practical power-limited systems the alphabet cannot be increased without bound⁽¹⁾, some form of compromise must be made between the reduction in signal distortion obtained by reducing the symbol rate and the reduction in tolerance to noise caused by increasing the signal alphabet, whilst maintaining a given information rate. Since the more sophisticated data detectors described in Chapter 6 are designed to operate with severely distorted data signals, the choice of compromise is very difficult to make but the current trend, due to both system performance and complexity, is to design systems with symbol rates close to the Nyquist rate and with the lowest signal alphabet consistent with the required rate of transmission^(11,16,18).

To illustrate the point further, consider as an example a channel with a nominal bandwidth of 3200 Hz, which could be a good private circuit or a very good switched line, over which it is required to transmit data at 19200 bit/s. The (double-sideband) baseband equivalent bandwidth will therefore be 1600 Hz and so the Nyquist rate will be 3200 bauds.

System a:

Symbols are transmitted at 3200 bauds and so the signal alphabet, from equations (2.3.1)-(2.3.3) will be;

$$m = 2^{1T} = 2^{19200/3200} = 64$$

$$\text{and, coding efficiency} = \frac{19200}{3200} = 6 \text{ bit/symbol} \quad (2.3.4)$$

System b:

The next closest symbol rate to the Nyquist rate which,

- (i) Is below the Nyquist rate
- (ii) Results in an integer value for the signal alphabet
- (iii) Results in an integer value for the coding efficiency
- (iv) Produces a signal alphabet with an even number of signal points such that symmetry can occur around zero and the signal can be considered multilevel polar in form,

occurs when;

$$m = 256$$

$$1/T = 2400 \text{ bauds}$$

$$\begin{array}{l} \text{Coding} \\ \text{efficiency} \end{array} = 8 \text{ bit/symbol} \quad (2.3.5)$$

A comparison of systems a) and b) above reveals that to transmit 19200 bit/s over the given channel will require either a baud-rate at (or even slightly above) the Nyquist rate, so increasing the possibility of signal distortion but with a tolerable signal alphabet or alternatively, a baud-rate considerably less than the Nyquist rate but with a greatly increased signal alphabet. What is not clear from this simple comparison however, is the trade-off that exists between actual system performance in terms of tolerance to distortion at the higher symbol rates and the predicted reduction in the tolerance to noise occurring at the higher signal alphabets; this prediction being based on the assumption that the average power of the transmitted signal is limited to some maximum value and so any increase in the signal alphabet will necessarily reduce the "distance" between adjacent symbol values, so increasing the probability of error in the presence of additive noise. This trade-off between symbol rate and signal alphabet is still further complicated by the coding efficiencies of the two systems. If one symbol is incorrectly detected when system b) is used, up to 8 data bits may be in error compared with a maximum of 6 erroneous bits when using system a). Consequently, system b) may or may not give a better overall performance when judged against system a).

In order to resolve the above uncertainties, both systems are investigated in detail in the course of this study, the results of which are given in Chapter 6, but before leaving this discussion of parameter compromises, it is worth mentioning a further factor which can also influence the system designer; that of realisable implementation. If, as has been suggested above, a fairly sophisticated detection process is to be employed within the system, the designer must ensure that the technology actually exists which enables the detector to carry-out the quite involved

signal processing and large amounts of computations that must be achieved within a single symbol period. In the case of a maximum likelihood detector, the amount of processing and the number of calculations required increase alarmingly as the number of allowed symbol values increase, but as the required symbol rate would decrease (assuming that a constant information rate is to be maintained), the time available to the processor to actually detect each symbol would be greater. This problem is considered further in Chapter 6, where a comparison is made between the two systems in terms of computational effort required by the detector.

Finally in this section dealing with some of the practicalities of system design and the very feasibility of transmitting data at 19,200 bit/s over telephone channels, mention must be made of the theoretical upper bound of the rate of information transfer over noisy, bandlimited channels and to ensure that 19,200 bit/s does in fact lie below this bound for sensible and practically obtainable transmission conditions. The upper theoretical limit, called the Shannon limit after Claude Shannon⁽¹⁹⁾ relates the maximum theoretical error-free rate of information transmission through a linear channel which has a bandwidth of B Hz and signal-to-noise power ratio of S/N , where S/N here refers to the ratio of average signal and noise powers at the output of the communication channel. Appendices 2 and 3 define the signal-to-noise ratio, or SNR, as used in the remainder of this work. Shannon showed that for an ideal distortionless bandlimited linear system operating at its Nyquist rate of $2B$ bauds, the symbols being shaped such that the symbol stream resembles a Gaussian source, the maximum error-free rate of transmission of information is given by:-

$$C = B \log_2 (1 + S/N) \text{ bit/s} \quad (2.3.6)$$

where C is the capacity of a noisy transmission system. Figure 2.3.1

illustrates this relationship graphically for the two bandwidth values 2800 Hz and 3200 Hz, where it can be seen that in order to transmit 19,200 bit/s through a channel satisfying Shannon's criteria, signal-to-noise ratios of approximately 21 dB and 18 dB respectively will be required.

Over practical telephone channels, Shannon's results do not fully apply due to the considerable levels of distortion present on the signal. It is reasonable to expect therefore that the capacity of such channels will be less than the corresponding Shannon channel with the same bandwidth and signal-to-noise ratio. As an example, Lucky et.al.⁽³⁾ show that over a typical telephone channel with a signal-to-noise ratio of around 30 dB, the capacity reduces to approximately 23500 bit/s compared with 31898 bit/s and 27911 bit/s for the two Shannon channels with bandwidths of 3200 Hz and 2800 Hz respectively.

An interesting analysis by Harvey et.al. in 1977⁽¹⁰⁾, takes the investigation slightly further by including effects caused by actual equipment imperfections as well as channel distortion. It was shown that for a given detection process, several system parameters may well be incorrectly adjusted or inaccurately estimated at the receiver and that the combined effect of these imperfections could be considered as a form of noise waveform which could be added to the signal, plus any additive noise (impulsive) introduced within the system itself. The investigation suggests that the more sophisticated detection processes (implying better detector performance) will be able to tolerate greater amounts of equipment imperfections and so another compromise will have to be made between the detectors complexity and performance and the achievable accuracy of the various parameters required by the detector. The investigation concludes that data rates up to 20,000 bit/s should be obtainable over channel bandwidths of around 3000 Hz when due allowance is made for the types of noise found on typical telephone circuits.

The transmission of 19200 bit/s over telephone channels therefore, at least in theory, appears to be a feasible proposition. The remainder of this work investigates in a practical manner the validity of this proposition by using various modelling and computer simulation techniques and also assess the feasibility of implementation using the latest technology, with a view to producing a commercially competitive modem.

2.4 References

1. Clark, A.P., "Principles of Digital Data Transmission", Pentech Press Ltd., 1976.
2. Sklar, B., "A Structured Overview of Digital Communications - A Tutorial Review, Parts 1 and 2", IEEE Communications Magazine, Vol.21, No.5, August 1983 and Vol.21, No.7, October 1983.
3. Lucky, R.W., Salz, J. and Weldon, E.J., "Principles of Data Communications", McGraw-Hill Book Company, 1968.
4. Monk, C.L., "Transmission Characteristics of Connections in the Switched Telephone Network Measured in the 1967 Field Study", Post Office Telecommunications Research Department Report 598, 1977.
5. McVerry, F., "High Speed Data Transmission over HF Radio Links", Ph.D. Thesis, 1982, Loughborough University of Technology.
6. Carlson, A.B., "Communication Systems", McGraw-Hill Inc., Second Edition, 1975.
7. Schwartz, M., "Information Transmission, Modulation and Noise, McGraw-Hill Inc., Third Edition, 1980.
8. Clark, A.P., "Advanced Data-Transmission Systems", Pentech Press Ltd., 1977.

9. Harvey, J.D., "Synchronisation of a Synchronous Modem", S.E.R.C. Report No. GR/A/1200.7, 1980.
10. Harvey, J.D., Clark, A.P. and Driscoll, J.P., "High Speed Line Modems", Final Report, Contract No. K/M2 11a/616, Loughborough University of Technology.
11. Clark, A.P. and Fairfield, M.J., "Detection Processes for a 9600 bit/s Modem", The Radio and Electronic Engineer, Vol.51, No.9, pp.455-465, September 1981.
12. Clark, A.P., Harvey, J.D. and Driscoll, J.P., "Near-Maximum Likelihood Detection Processes for Distorted Digital Signals", The Radio and Electronic Engineer, Vol.48, No.6, pp.301-309, June 1978.
13. Clark, A.P. and Harvey, J.D., "Detection Processes for Distorted Binary Signals", The Radio and Electronic Engineer, Vol.46, No.11, pp.533-542, November 1976.
14. Clark, A.P., Kwong, C.P. and Harvey, J.D., "Detection Processes for Severely Distorted Digital Signals", Electronic Circuits and Systems, Vol.3, No.1, pp.27-37, January 1979.
15. Clark, A.P. and Harvey, J.D., "Detection of Distorted Q.A.M. Signals", Electronic Circuits and Systems, Vol.1, No.3, pp.103-109, April 1977.

16. Najdi, H.Y., "Digital Data Transmission Over Voice Channels",
Ph.D. Thesis, 1982, Loughborough University of Technology.
17. Forney, G.D.Jr., "The Viterbi Algorithm", Proc. of the IEEE,
Vol.61, No.3, pp.268-278, March 1973.
18. Shimbo, O. and Celebiler, M.I., "The Probability of Error due to
Intersymbol Interference and Gaussian Noise in Digital
Communication Systems", IEEE Trans. on Communication Technology,
Vol.Comm-19, No.2, pp.113-119, April 1971.
19. Shannon, C.E., "A Mathematical Theory of Communication", Bell.
Syst.Tech.J., Vol.27, pp.379-423 and pp.623-656, July and October 1948.

Line Signalling Rate (Baud)	Coding Efficiency Bits per symbol
1920	10
2400	8
3200	6
3840	5
4800	4
6400	3
9600	2

TABLE 2.1

Possible line signalling rates and
resultant coding efficiencies for the
transmission of 19200 bit/s

Possible transmitted sequences and sequences held in detectors store			
1	1	1	1
1	1	1	3
1	1	3	1
1	1	3	3
1	3	1	1
1	3	1	3
1	3	3	1
1	3	3	3
3	1	1	1
3	1	1	3
3	1	3	1
3	1	3	3
3	3	1	1
3	3	1	3
3	3	3	1
3	3	3	3

Stored Sequences Convolved with Sampled Impulse Response					Value of Cost Function *
1	1.5	1.5	1.5	0.5	5.0
1	1.5	1.5	3.5	1.5	10.0
1	1.5	3.5	2.5	0.5	6.0
1	1.5	4.5	4.5	0.5	17.0
1	3.5	2.5	1.5	0.5	0.0
1	3.5	2.5	3.5	1.5	5.0
1	3.5	4.5	2.5	0.5	5.0
1	3.5	4.5	4.5	1.5	14.0
3	2.5	1.5	1.5	0.5	6.0
3	2.5	1.5	3.5	1.5	11.0
3	2.5	3.5	2.5	0.5	7.0
3	2.5	3.5	4.5	1.5	16.0
3	4.5	2.5	1.5	0.5	5.0
3	4.5	2.5	3.5	1.5	10.0
3	4.5	4.5	2.5	0.5	10.0
3	4.5	4.5	4.5	1.5	19.0

* Cost function calculated from $\sum_{i=1}^5 (r_i - r_i')^2$
 r_i = Received sequence [1 3.5 2.5 1.5 0.5] in this example
 r_i' = Convolved sequences

TABLE 2.2 Example of maximum likelihood detection

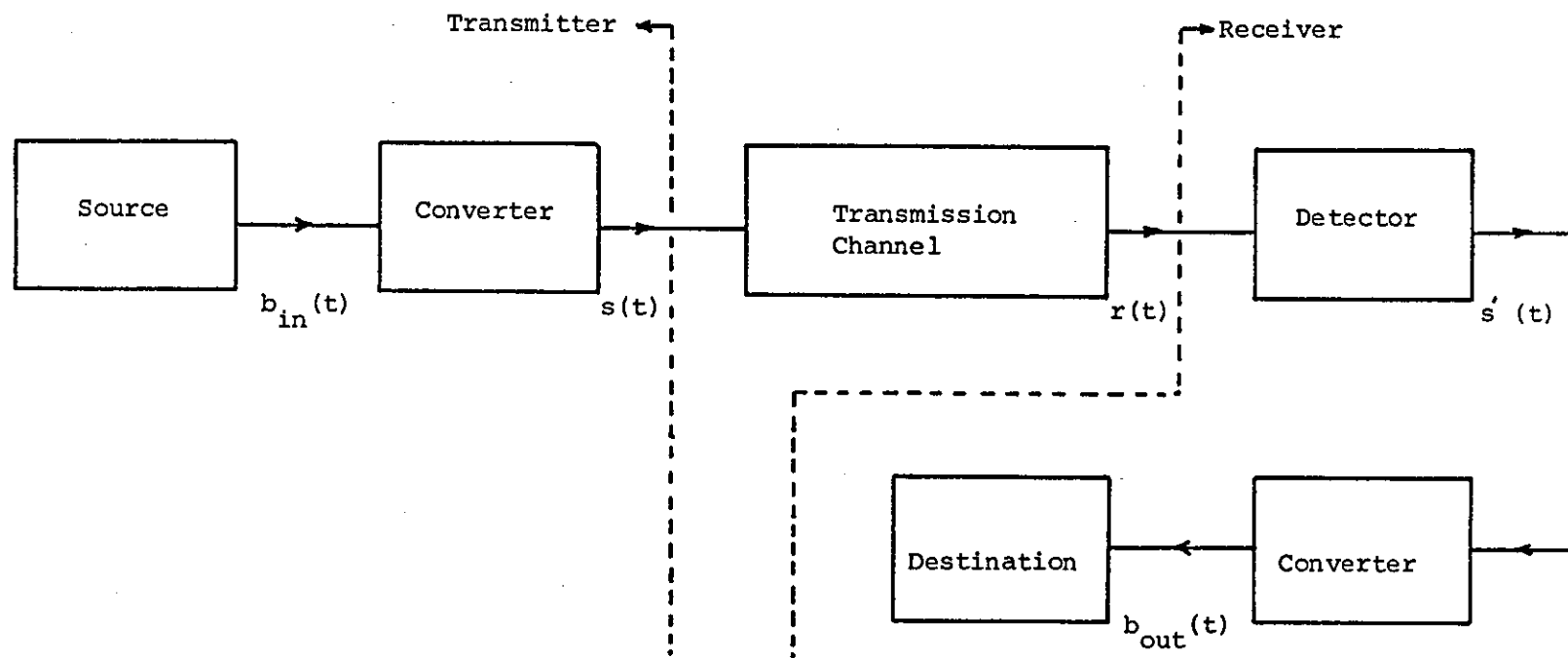


Figure 2.2.1: Simplified model of a general digital data transmission system

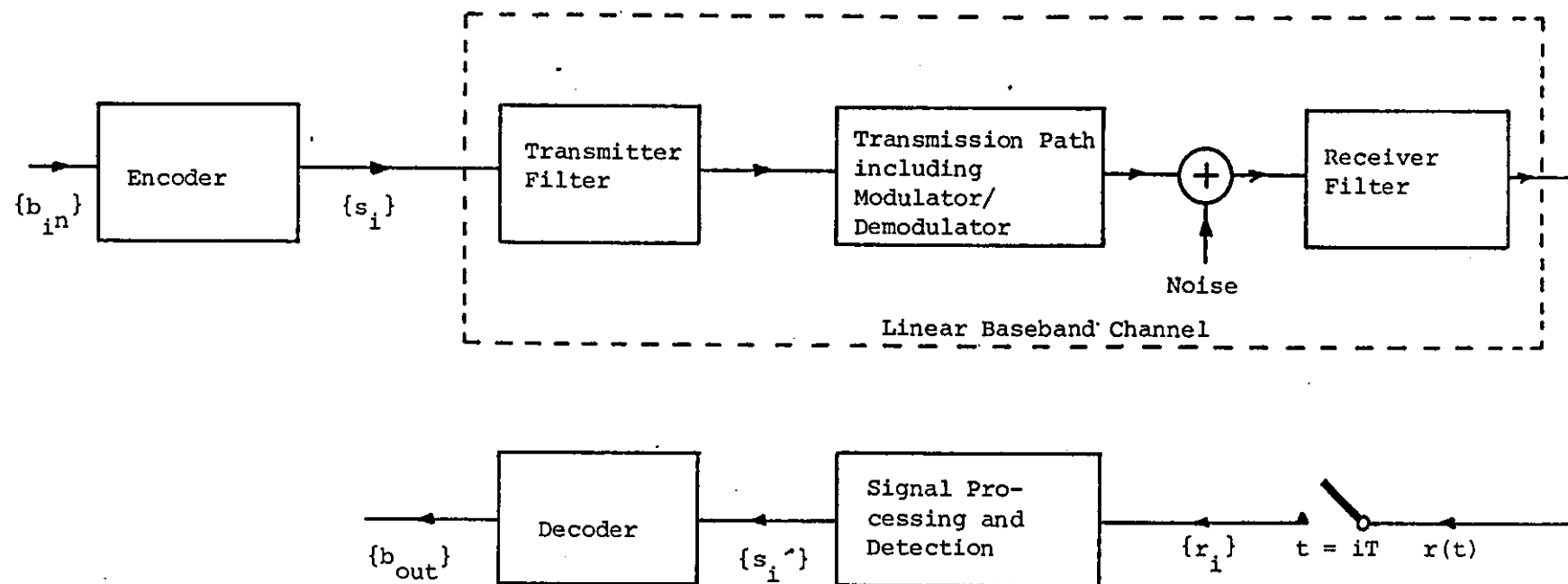


Figure 2.2.2: Model of data transmission system

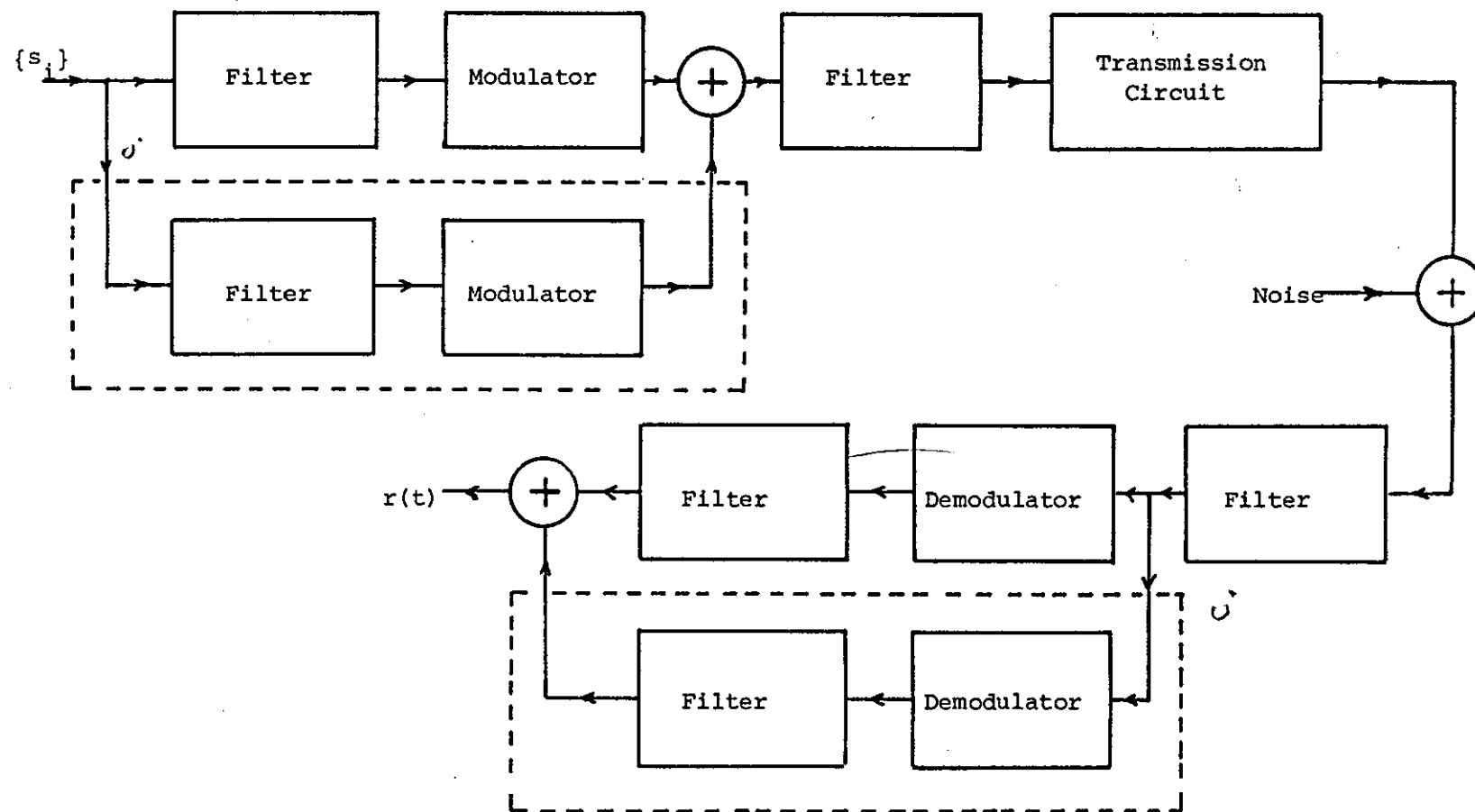


Figure 2.2.3: Constituents of linear (complex) baseband channel

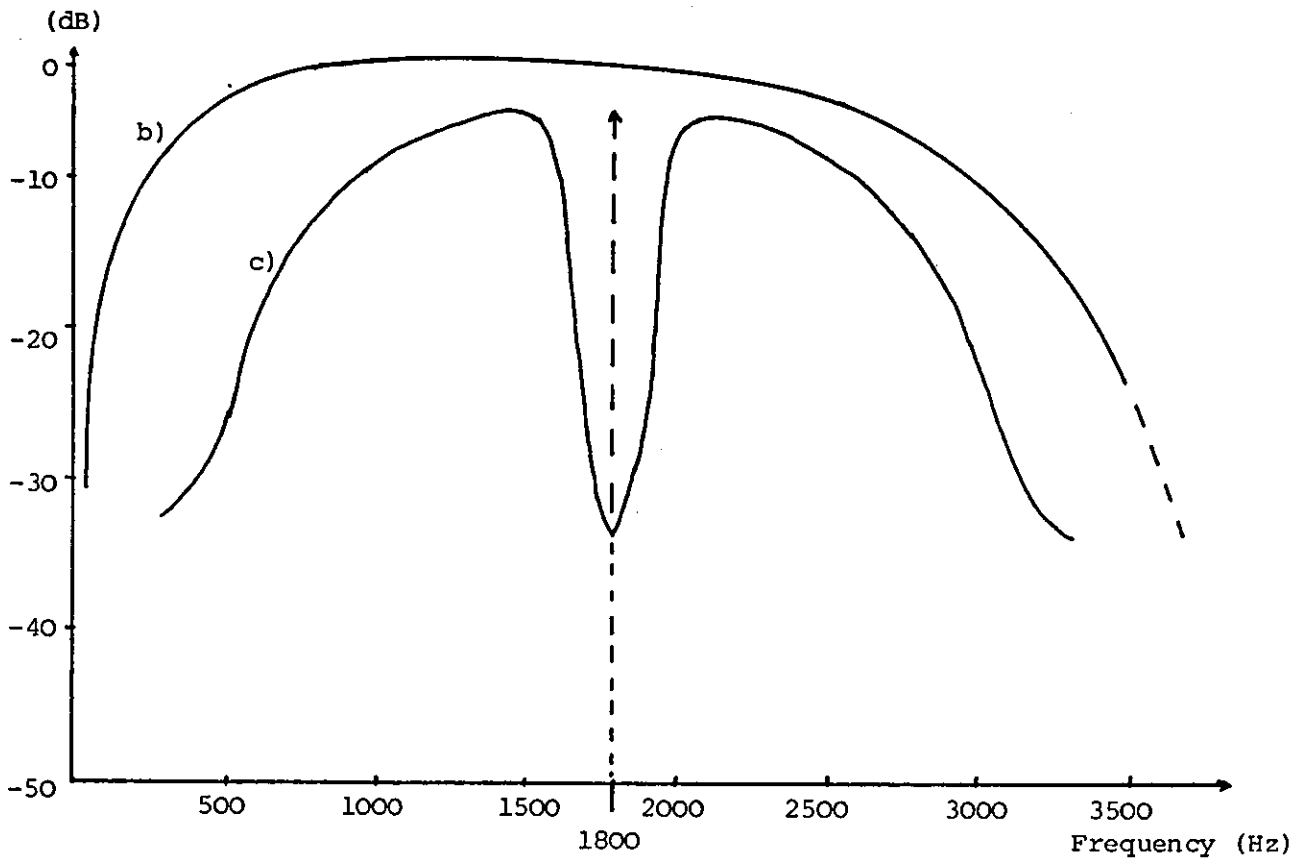
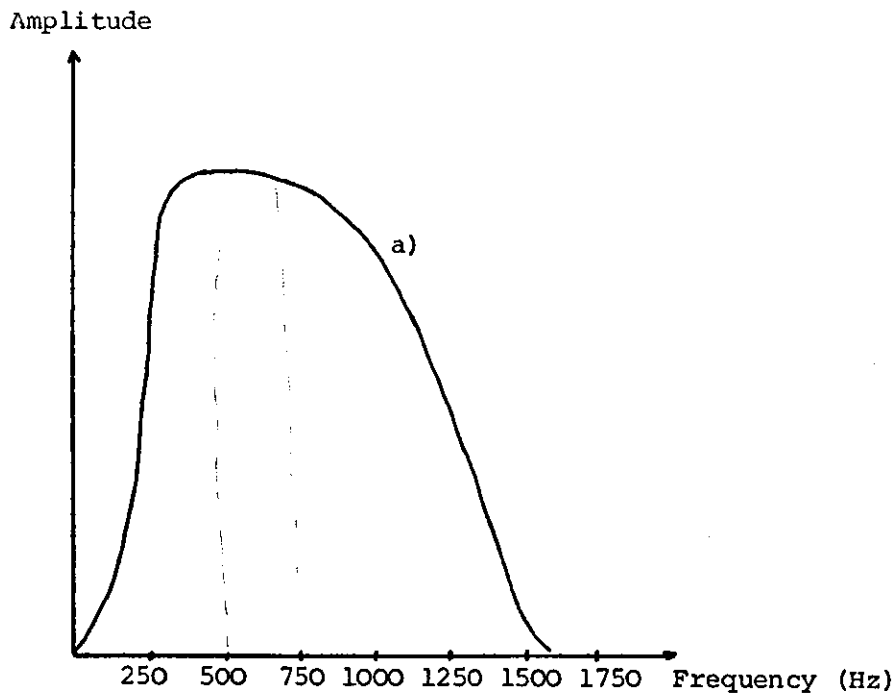


Figure 2.2.4: a) Amplitude spectrum of hypothetical baseband signal
 b) Amplitude characteristic of bandpass channel
 c) Amplitude spectrum of modulated signal with a carrier frequency of 1800 Hz

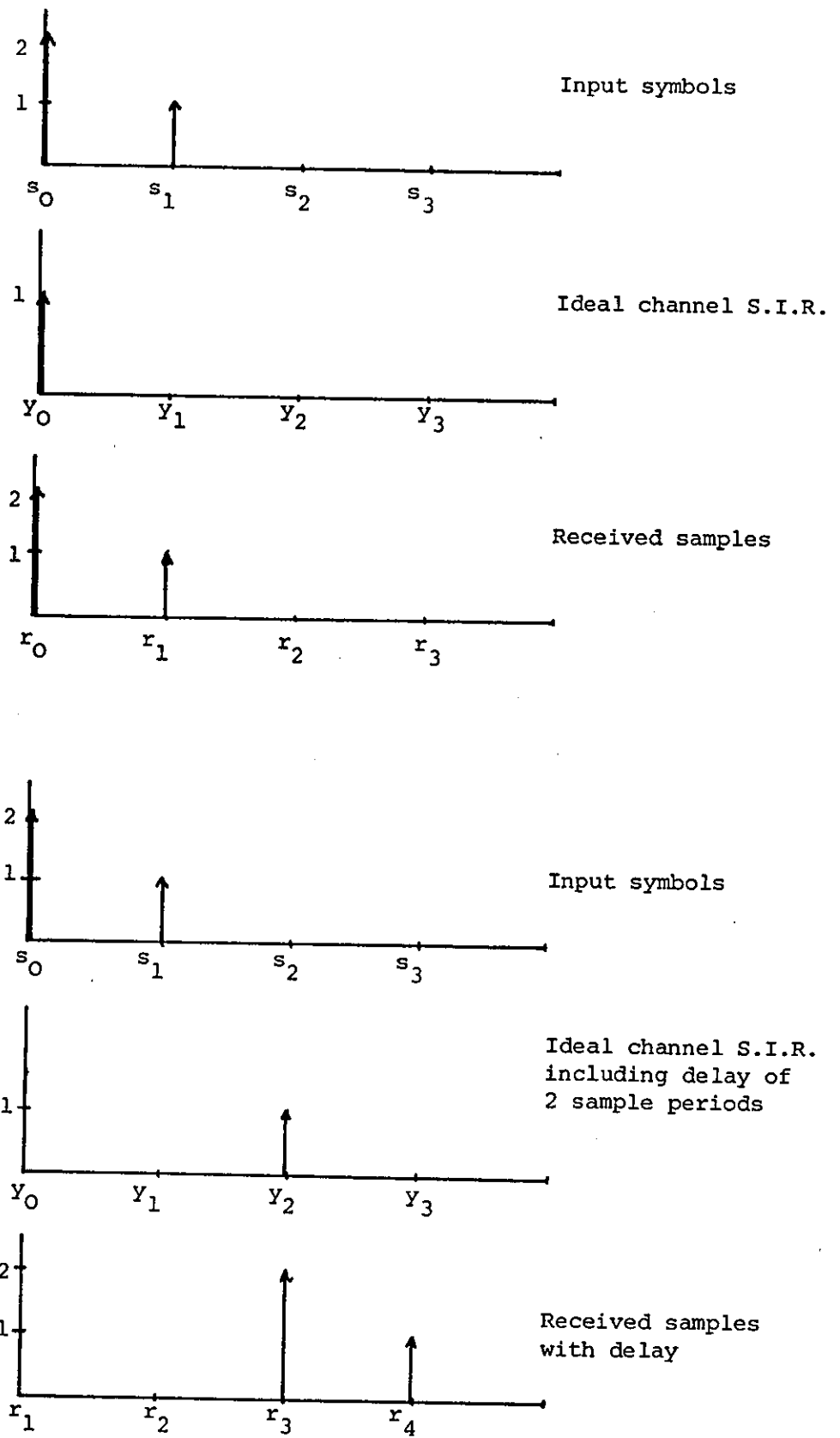
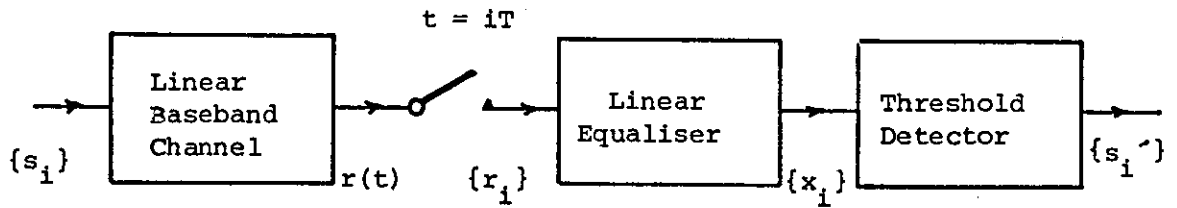


Figure 2.2.5: Effect of zero-valued precursors in the channels sampled impulse response (S.I.R.)

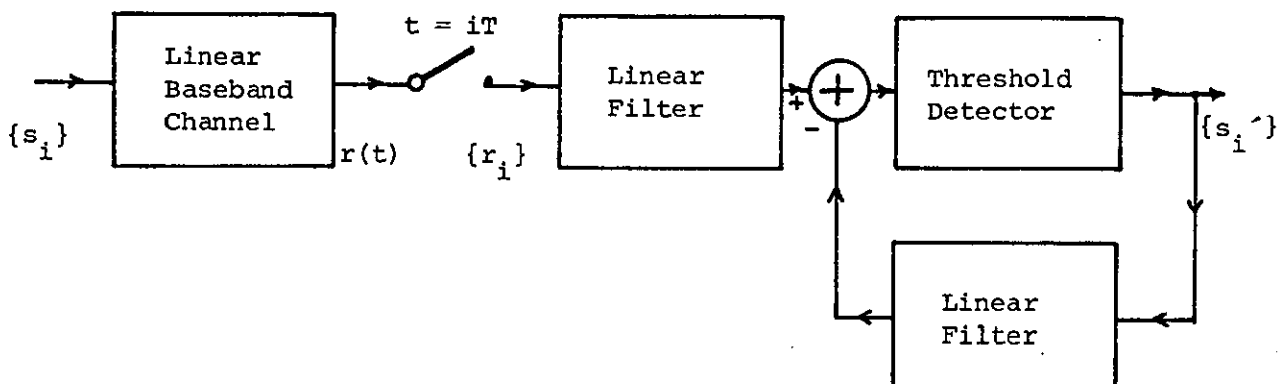


$$\text{S.I.R. of channel} = [y_0 \ y_1 \ \dots \ y_g] = Y$$

$$\text{S.I.R. of equaliser} = [c_0 \ c_1 \ \dots \ c_m] = C$$

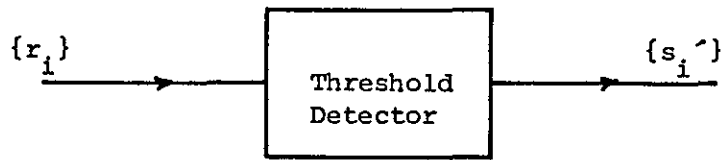
$$\text{S.I.R. of equalised channel} = [0 \ 0 \ \dots \ 1 \ 0 \ \dots \ 0] = E$$

(a) Linear Equalisation

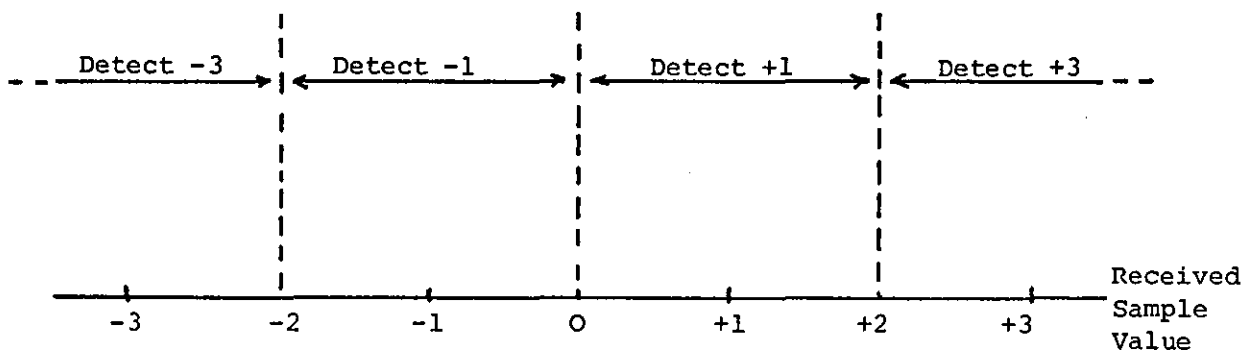


(b) Non-linear Equalisation

Figure 2.2.6: Linear and non-linear equalisation



Signal Alphabet $\pm 1, \pm 3$



Thresholds set at $0, \pm 2$

Figure 2.2.7: Principles of threshold detection in the presence of Gaussian noise with zero-mean value

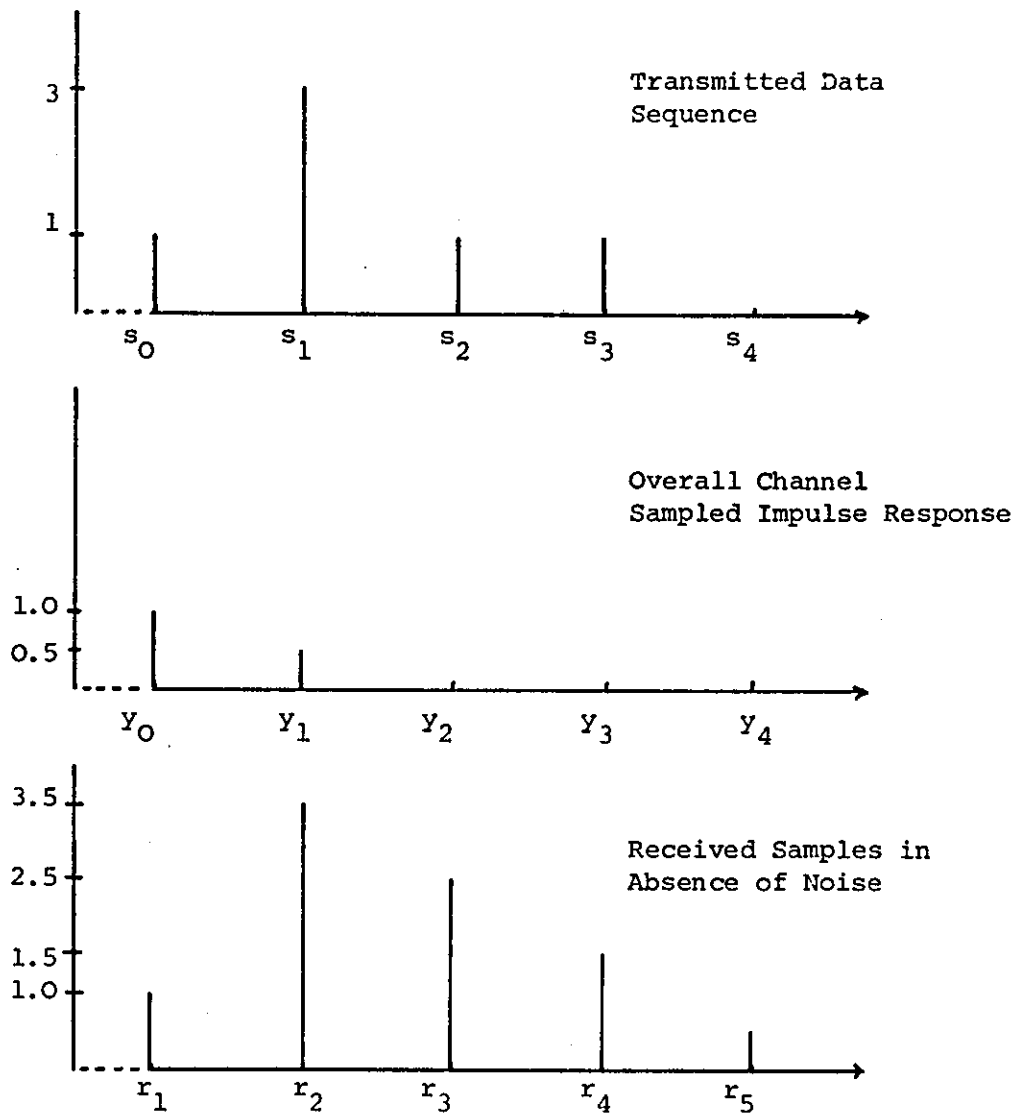


Figure 2.2.8: Maximum likelihood detection.
 Example of distortion present in
 received samples due to non-ideal
 sampled impulse response (see also Table 2.2)

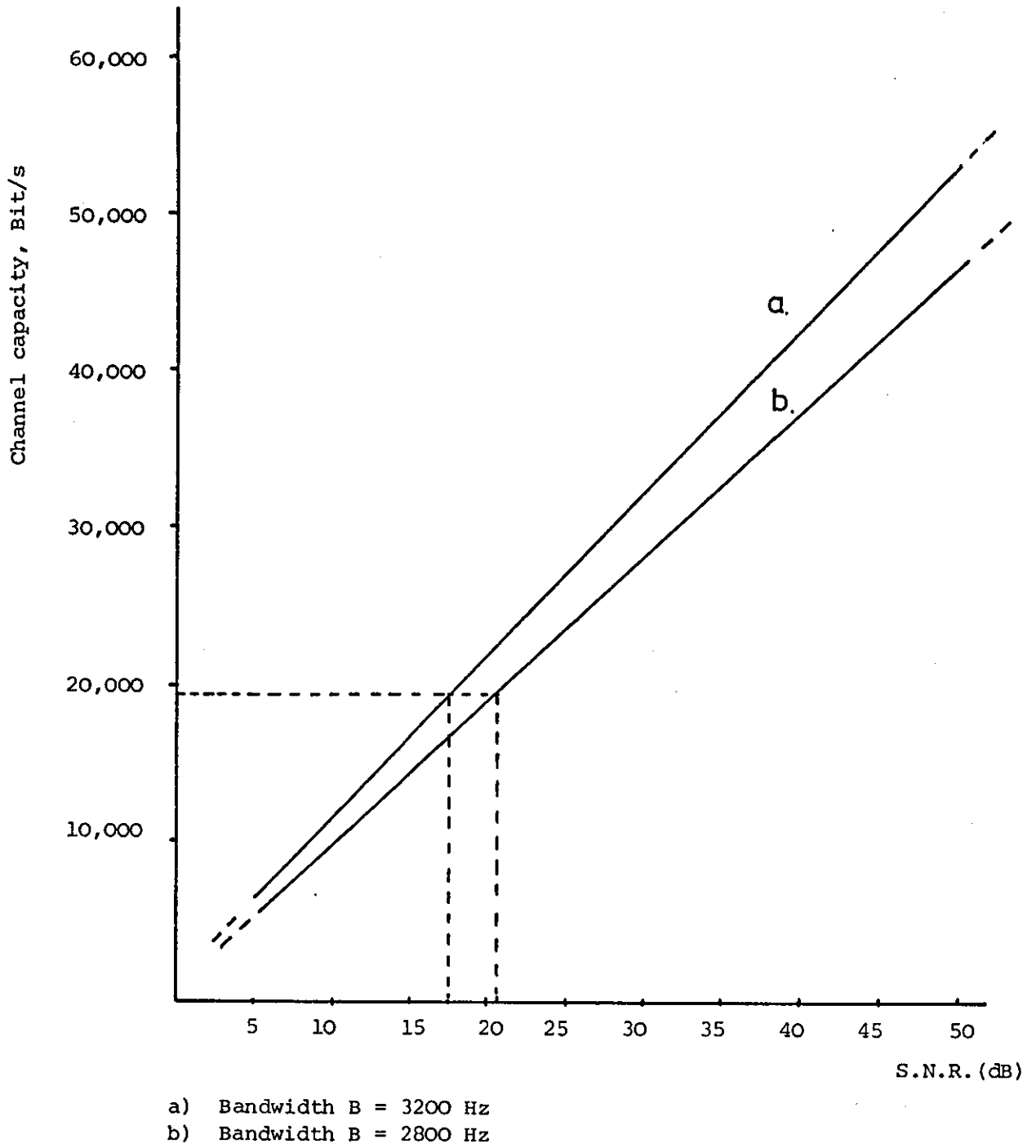


Figure 2.3.1: Shannon capacity of a noisy channel

$$C = B \log_2 \left(1 + \frac{S}{N} \right)$$

3 The Telephone Channel; Characteristics and Impairments

3.1 Introduction

Although this work is concerned with the transmission of high-speed digital data over telephone lines, it must be remembered that the telephone network from its conception in the late 19th century to the present automated systems of today, has been designed and finely-honed for the reliable and economic transmission of (analogue) speech between human users. As well as providing reliable communications, the network must also provide a grade of service acceptable to its users (subscribers) in terms of the quality of transmission, the time necessary to establish a connection between the calling and called parties, the ability to communicate with parties in other countries (requiring international collaboration) etc., and so it is important to be able to measure the quality of service provided and set standards of acceptability for the majority of the network's customers. It is interesting to note that in the very early days of telephony when telephones and telegraphy shared the same basic line plant, the very ability to talk directly to a person remotely situated would have been service enough. However, in later years, the increase in customer's expectations of system performance forced network operators to adopt more stringent methods of measurement and assessment. This has led to detailed subjective studies being performed in order to determine the grade of service (in particular, the quality of speech reproduction) tolerable to the majority of users. Based upon the results of these subjective studies, network operators were then able to set standards for the physical components of actual network equipment, standards relating to, for example, levels of signal distortion, amount of noise introduced, limits for tolerable cross-talk, line bandwidths, etc., an obvious improvement on

the standards set in 1891 when Mr A R Bennett, addressing the British Association with his lecture "On the Telephoning of Great Cities", suggested that for an effective and popular telephone service, transmission should occur in the "absence of all disturbing sounds"⁽¹⁾.

In terms of the transmission of digital data, the telephone network is important due to its very existence. The national (and international) Public Switched Telephone Network (PSTN) provides the medium for communication between all subscribers of the network, the mode of communication being in the main, speech over voice-frequency channels. Provided methods exist which enable digital data to be transmitted and received successfully over these voice-frequency channels, then users wishing to communicate data rather than speech will benefit from all of the facilities the network has to offer, for example, the ability to exchange data with any other subscriber with suitable equipment. The major problem associated with the transmission of data over the telephone network stems from the quality of service tolerable to the main-stream (speech) users. As mentioned earlier, the quality of service requirement was initially based on subjective studies, where varying degrees of noise, interference and distortion were introduced under experimental conditions to assess human tolerance to these impairments⁽¹⁾. However, the human brain is a very sophisticated processing device and can successfully extract messages from extremely noisy and distorted signals whereas digital data systems are in general, far less tolerant to such impairments. To illustrate this point further, a trained person can successfully isolate and comprehend a given speech signal in the presence of many other simultaneous signals, as in the case of military aircraft intercommunication systems. However, if many digital data signals, each with similar signal spectra, were to be transmitted over a single channel, then an extremely sophisticated receiver and decoder structure would be required to isolate the individual signals.

Although digital data has been transmitted successfully over the telephone network for many years, the modern need to communicate larger amounts of data in shorter times has resulted in the design of higher-speed systems. These systems still have to operate over the existing voice-frequency channels provided by the network and so it has become increasingly important for design engineers to fully understand the limitations of the networks and the effect of impairments on the transmission of high-speed digital signals. (It is worth noting that whilst the increased demand for high-speed data transmission has been due to the rapid advances in Information Technology related subjects, it is one of these subjects, microelectronics, which has supplied the technology to actually implement modern designs.) The remainder of this chapter studies the various types of connection that may be established over the telephone network and investigates the causes of signal distortion, the effect on the transmission of digital data and methods of characterising telephone channels. The characteristics of switched connections and private lines are then discussed along with typical characteristics likely to be encountered in practice. Finally, other impairments found on telephone circuits are described. Where possible, each impairment is treated separately and an appropriate model presented but it should be appreciated that in practice, some or all of the impairments will occur at the same time.

3.2 Types of Telephone Circuits

In general, telephone channels are made up of circuits connected in cascade to form a continuous physical medium over which electrical signals representing either speech or data may be transmitted. Telephone networks are normally constructed using a star topology⁽²⁾ where each subscriber is directly connected to a local exchange or switching centre. All control and

supervisory signals and the information carrying signal itself are operated on or routed through the local switching centre. Consequently, any switched connection via the PSTN will involve at least two separate circuits; one extending from the calling subscriber to the exchange, the other extending from the exchange to the called subscriber via any equipment and switches necessary to connect the two parties. Clearly, there exists many different possible physical connections between subscribers since any user can initiate a connection between any other user attached to the system and also because the internal organisation of telephone exchanges is such to share common equipment wherever possible. That is, apart from the subscribers dedicated input /output access unit connecting the line-plant to the exchange, equipment such as step-by-step switches, cross-connection switches, exchange batteries, etc., are shared between many (if not all) subscribers connected to that exchange. As the complexity of the network hierarchy increases to include trunk exchanges, group switching centres and international exchanges, so the number of possible physical connections increases and consequently the electrical characteristics of connections via the PSTN will vary over quite wide limits. More importantly, the characteristics of a particular connection will not be known in advance nor indeed can it be guaranteed to comply with the levels of noise, distortion and interference introduced on published "typical" telephone circuits⁽¹⁻⁷⁾.

The second type of telephone circuit to be considered is the private circuit, also known as a private line, leased line and private wire^(2-6,24). Private lines differ from switched lines in that although they may pass through terminating stations such as telephone exchanges and repeater stations, they are not connected through any of the switches in an exchange nor to the exchange or repeater station's battery supplies. Private lines

can therefore be considered as point-to-point connections which are permanently rented by one or more subscribers. Since private lines are of a more permanent nature than switched lines, their characteristics can be checked individually and where necessary, additional equalisation equipment included to give a channel response suitable for the particular customer's requirements. Because of this, private lines tend to introduce lower amounts of certain impairments compared with switched connections and are in general, better suited for higher-speed transmission of digital data. However, private lines are still basically voice-frequency channels and will therefore pass through the internal telephone network comprising local, junction and trunk circuits.

3.3 Characteristics of Telephone Circuits

Regardless of the type of telephone line, the physical medium which joins two customers may be made up of several different cascaded links. Generally speaking, three distinct types of circuit exist, namely, unloaded audio, loaded audio and carrier circuits. Unloaded audio circuits are usually limited to fairly short connections, of the order of 5 km and comprise a pair of balanced twisted wires with a nominal impedance of 600Ω . A pair of wires is used to reduce the effects of capacitively coupled hum from power lines and power systems and the wires within each pair are twisted together such that they lie in close proximity to one another. This ensures that any interference will be coupled to both wires by equal amounts and so there will be no interfering signals between the two conductors. To further reduce cross-talk between pairs of wires, cables comprising many pairs are designed such that adjacent pairs are given different twist lengths. As well as

capacitive coupling, inductive coupling can also occur but again, the twisting of the pairs reduces the amounts of both inductively picked-up mains hum and inductively coupled crosstalk^(2,25).

All electrical (or electromagnetic) transmission lines exhibit characteristics whereby a signal passing through the line will suffer a loss of signal strength. This loss or attenuation of the signal is caused by the dissipation of energy either as heat or as a consequence of coupling into adjacent circuits. The lines may therefore be represented in terms of the distributed primary line coefficients; R , the resistance of the line; L , the inductance of the wires; G , the leakage between the wires and C , the capacitance between the wires as illustrated for a short length of line in Figure 3.3.1⁽⁹⁾. Clearly, the output of such a line when fed from a constant amplitude, variable frequency source will be a function of the applied frequency due to the presence of the reactive elements L and C . The phase of the output signal, when normalised with respect to the phase of the input signal, will also be a function of the input signal's frequency. As a result of this, all transmission lines may be diagrammatically or tabularly characterised in terms of;

- a) the loss-verses-frequency characteristic
 - b) the (relative) phase-verses-frequency characteristic
- or by the derivatives;
- c) attenuation response
 - d) group-delay response.

When attempting to interpret line characteristics, it is extremely important to realise that although a) and c) above are basically the same, the definitions of the characteristics and therefore the units with which they are expressed are quite different;

- a) Loss or more accurately, insertion loss, is defined as;

$$\text{Insertion loss at frequency } f = 10 \log_{10} \left(\frac{\text{Power in at frequency } f}{\text{Power out at frequency } f} \right) \quad (3.3.1)$$

- c) Amplitude response (also called frequency response) is defined as;

$$\text{Amplitude response at frequency } f = 20 \log_{10} \left(\frac{\text{Voltage in at frequency } f}{\text{Voltage out at frequency } f} \right) \quad (3.3.2)$$

Clearly, if the input impedance of the given line and the load impedance at the output of the line are identical, then a) and c) above give identical characteristics. However, if a mismatch of impedances is present on a given line, the two characteristics will not be identical. Moreover, the actual shapes of the characteristics will differ from those of a correctly terminated line due to the presence of reflected signals and standing waves⁽⁹⁾.

The phase-verses-frequency characteristic and group-delay response of a given line are also related;

- b) Phase characteristic is defined as;-

$$\text{Phase response at frequency } f = (\text{Output phase} - \text{Input phase}) \quad (3.3.3)$$

- d) Group delay characteristic is the rate of change of the phase characteristic with respect to frequency.

The amplitude and phase characteristics of any linear system, including transmission lines, may also be presented in the form of the complex transfer function. If $H(f)$ is the complex transfer function,

$$H(f) = \text{Re}(f) + j\text{Im}(f) \quad (3.3.4)$$

where $\text{Re}(f)$ is the real part of $H(f)$

$\text{Im}(f)$ is the imaginary part of $H(f)$

and $j = \sqrt{-1}$

then if $A(f)$ is the amplitude response and $\phi(f)$ is the phase response of the system,

$$A(f) = 20\log_{10} |H(f)| \quad (3.3.5)$$

$$A(f) = 20\log_{10} (\text{Re}^2(f) + \text{Im}^2(f))^{\frac{1}{2}} \quad (3.3.6)$$

$$A(f) = 10\log_{10} (\text{Re}^2(f) + \text{Im}^2(f)) \quad (3.3.7)$$

and $\phi(f) = \text{Arg}(H(f)) \quad (3.3.8)$

$$\phi(f) = \tan^{-1} \left(\frac{\text{Im}(f)}{\text{Re}(f)} \right) \quad (3.3.9)$$

Alternatively, if $A_t(f)$ is the attenuation characteristic and $\theta(f)$ the group delay characteristic of the system;

$$A_t(f) = -10\log_{10} (|H(f)|^2) \quad (3.3.10)$$

$$A_t(f) = -10\log_{10} (\text{Re}^2(f) + \text{Im}^2(f)) \quad (3.3.11)$$

and $\theta(f) = \frac{-1}{2\pi} \frac{d(\phi(f))}{df} \quad (3.3.12)$

where $\phi(f)$ is expressed in radians.

The main advantage of using the transfer function to describe a telephone circuit is that the time response of the line may be obtained by taking the inverse Fourier transform of $H(f)$. This time response, called the impulse response of the system, gives an alternative method of characterising telephone circuits and of estimating the amount of distortion a signal will suffer as it passes through the system, as described later in this section. Also, the sampled version of the impulse response (the sampled impulse response or SIR) is of direct importance to the transmission of digital data over analogue channels and is considered in detail in Chapter 5.

As the length of an audio link increases, so too will the amount of attenuation experienced by a signal passing over that link. Cables used for audio circuits are usually classified in terms of the weight of conductors used (lb per mile) or more recently by the diameter of the wires. Clearly, different classifications of cables will have different values for the primary line coefficients and will therefore exhibit different attenuation and phase characteristics, the amount of attenuation and phase being dependent on the length of the conductors. Figure 3.3.2 shows the attenuation characteristic (dB per km verses frequency) for a 0.9 mm cable⁽²⁾. It can be seen that the combined effects of the series inductance and parallel capacitance shown in Figure 3.3.1 results in increasing attenuation per kilometer as frequency increases. Over very long audio circuits, the high-frequency components of a signal passing through it can be severely attenuated, so changing the shape of the signal at the output of the circuit. To counter this effect, inductance is added which gives

rise to loaded audio circuits, the term loading implying the addition of the inductance. In practice, loading is achieved by placing induction coils or "loading-pots" of 88 mH at about 1.8 km intervals on junction and trunk circuits. Figure 3.3.2 also shows the attenuation response of a loaded section of 0.9 mm cable, from which it can be realised that transmission may be achieved over greater distances when using loaded audio circuits. However, the price paid for a flatter response in the passband of commercial quality speech is an increase in the rate at which both attenuation and phase change with frequency at the high frequency end of the characteristics.

For loaded or unloaded circuits to operate over relatively long distances, the effect of increasing attenuation with distances must be offset by the inclusion of amplifiers. The amplifiers or repeaters are spaced at regular intervals along the line and are adjusted to provide sufficient gain to counter the effects of attenuation but without exceeding the limits of average power input; too much gain could cause a later amplifier unit to saturate, resulting in non-linear distortion and a possible increase in cross-talk between adjacent circuits. It should also be realised that the addition of amplifiers alone does not improve the signal-to-noise ratio on a line since both the noise and signal will be amplified by similar amounts. However, if a filter precedes an amplifier, the bandwidth of the noise and therefore the noise power may be reduced which would cause an increase in signal-to-noise ratio provided the amplifier itself does not contribute any significant additional noise to the system. The filters themselves will of course modify the characteristics of the circuit and can be a major source of delay distortion^(2-5,12). The presence of noise on the line also dictates the use of regularly spaced amplifiers rather than one high gain amplifier at the receiving end of a circuit since the attenuated signal

could be completely lost in noise before reaching the amplifier. A single amplifier is not used at the sending end of the circuit as the gain required to counter the line losses would be impractically high.

The third type of link likely to be encountered is the carrier link. Carrier links can operate over greater distances than audio links, whether loaded or unloaded and utilise the spectrum translating property of amplitude modulation, where the spectrum of the signal to be transmitted is translated to a higher frequency range by the process of single-sideband suppressed-carrier amplitude modulation^(2,3). Clearly, if a transmission circuit exists which exhibits low attenuation over a wide range of frequencies then many signals may be transmitted simultaneously by arranging for each signal spectrum to be shifted to a different frequency range. This combination of modulation and frequency-division multiplexing (FDM) is the basis of multi-channel carrier telephony systems which operate over carrier pairs, coaxial cables, land microwave radio relay systems, satellite systems and H.F. radio links⁽²⁾. At the receiver-end of a carrier link, demodulation with suitable local oscillator frequencies and bandpass filters restore the spectra of the individual signals to their baseband locations. The majority of present day carrier systems operate over wideband coaxial cables capable of supporting over 10,000 multiplexed voice frequency circuits simultaneously, the structure of the system being clearly defined in strict hierarchical levels ranging from the basic group level (12 channels) to supergroups and hypergroups. Each level requires additional modulator/demodulator units and appropriate filters to limit the spectra of the input signals to the system and to separate the components of the multiplexed received signal.

One of the advantages of coaxial cable systems (and microwave systems) over carrier pairs is the lower rate of change of attenuation per kilometer with increasing frequency. For coaxial systems operating at 12 MHz and 60 MHz, the variation of attenuation and group delay over the bandwidth occupied by a single circuit can be considered negligible and so the characteristics of such a single circuit will be determined almost entirely by the filters used in the frequency translation equipment at both the transmit and receive ends of the link. Also, because of the hierarchical structure of carrier systems, the circuits positioned towards the ends of the 12 channel groups may exhibit poorer characteristics than circuits located centrally due to the extra filtering required to separate the groups from each other⁽¹²⁾.

As with the case of loaded audio circuits, carrier systems designed to operate over long distances must include regularly spaced amplifiers to offset the increasing attenuation with distance, although the actual amplifiers used will necessarily have wider bandwidths than their counterparts on loaded audio circuits. Since amplifiers are uni-directional devices, a single 2-wire audio circuit or a single coaxial cable can carry signals in one direction only. Clearly, for a real-time speech conversation or for a real-time exchange of digital data, two 2-wire circuits or two coaxial cables with included amplifiers must be provided; one for the "go" path, the other for the "return" path. As well as increasing the amount of line plant and associated equipment, 4-wire working also presents a problem at the input/output stages of the network since the majority of subscribers are connected to their local exchanges by only a single pair of wires, this pair carrying signals in both directions. Arrangements must therefore be made to couple 2-wire lines to 4-wire systems, which is achieved by the use of hybrid transformers^(2,3). When correctly

balanced, a hybrid transformer should feed signals from the 2-wire line to the transmit half of the 4-wire line and feed signals from the receive half of the 4-wire line to the 2-wire line without coupling signals between the two halves of the 4-wire line. In practice, hybrid transformers are often not perfectly balanced and coupling between the transmit and receive sections of the 4-wire line exists to some extent. Furthermore, if the hybrid transformers at both ends of a 4-wire transmission system are both unbalanced (mismatched), a signal can pass repeatedly around the transmit-receive loop although with each repetition the coupled signal level will reduce due to attenuation across the mismatched hybrid. At the receive end of the system, the main signal will therefore be followed by a series of echoes, the echo delay being equal to the time taken for the signal to travel the 4-wire loop and where the level of each echo will steadily reduce due to extra attenuation of the signal as it passes through the loop. The presence of such echoes will obviously change the time response of the channel (and therefore the channel characteristics) and can introduce quite serious distortion on a digital data signal as well as causing undesired "listener-echo" on speech systems. From the above discussion, it should be clear that for high quality transmission of high-speed data over the telephone network, a customer should use 4-wire private circuits as end-to-end 4-wire connections, including the connections to the local exchange. However, such a system would not only be expensive to lease, but would also exclude the customer from the flexibility offered by the PSTN.

The two most widely studied characteristics of telephone connections are the attenuation and group delay variations with frequency, from which definitions of attenuation and group delay distortion may be based, viz:

Attenuation distortion is the variation of attenuation over a stated band of frequencies and group delay distortion is the variation of group delay over a stated band of frequencies.

For voice-frequency channels, the band of frequencies of interest is 300 Hz to 3400 Hz and so from the above definitions, the amount of signal distortion introduced by a telephone channel may be estimated from the attenuation and group delay characteristics of that channel, where of course the channel itself may be composed of several types of link. Consequently, the amount and types of distortion that a signal will suffer will depend on the distortions introduced by the individual links^(3,5-7,12). Unloaded audio links, due to their short lengths, introduce only a small amount of attenuation and delay distortion over the voice frequency band, typically about 2 dB and 20 μ s respectively. Loaded circuits, due to the presence of loading inductors, exhibit very low attenuation distortion over the centre of the band but show greatly increased attenuation and delay distortion at the higher frequencies, the amount of distortion increasing with distance as more loading is introduced. The characteristics of carrier circuits are determined almost entirely by the filters involved in the modulation, demodulation and multiplexing processes. A carrier circuit therefore exhibits bandpass characteristics with appreciable delay distortion towards the edges of the band. Unlike distortion introduced by loaded audio links, distortion on carrier systems originates at the terminal stations and so is independent of distance. Clearly, the presence of multiplexing and modulation equipment, amplifiers etc., will affect a signal as it passes through a telephone channel, particularly if the equipment is mis-aligned or unbalanced. This results in signal impairments rather than pure distortion and is investigated further in Section 3.4.

Line terminating equipment, transmission bridges and switches used in the PSTN will all affect the attenuation and group delay characteristics to some extent, the general effect being an increase in attenuation over the band and a degradation in the low-frequency characteristics of the circuit. Furthermore, even if the individual links are checked for suitable characteristics, the combination of the links forming a complete circuit will not necessarily result in a predictable characteristic due to mismatches that can occur between the links. The effect of a mismatch is similar to that produced by an unbalanced hybrid transformer where multipath propagation modes exist and the signal may arrive at the destination over more than one path. This gives rise to echo signals and the time dispersion of received data signal elements, resulting in intersymbol interference. Echoes can occur on both switched and private circuits although because the possibility of line mismatch is greater for connections over the switched network, severe echoes are more likely to be encountered over the PSTN. However, regardless of both the source of echoes and the type of line, their presence indicates a combination of attenuation and delay distortions where the signal distortion is dependent upon the relative magnitudes of the echo and direct signal components.

3.3.1 Characteristics of Switched Connections

It is obviously impractical to attempt to fully characterise all possible connections over the PSTN since equipment and line plant used for each connection is selected on a purely random basis. It is also impractical to attempt to design a digital data system which would be capable of reliable operation over all PSTN circuits and so a realistic assessment of the characteristics likely to be encountered for the majority of connections must be undertaken. Because of the increased use of the telephone network

for non-speech purposes over the past 10-20 years, detailed surveys of the network have been performed in order to provide both customers and system designers with a reasonable knowledge of likely circuit characteristics^(2,4-7,10,24,35).

Based on the definitions of attenuation and group delay distortion given earlier, the characteristics of an ideal voice-frequency channel may be shown as in Figure 3.3.3. In general, both the attenuation characteristic and group delay characteristic will be "U" shaped, rising quite rapidly at frequencies below about 1 KHz and rising again, but less rapidly, beyond about 3 KHz. Channels introducing severe distortion may well have characteristics with a sharper "V" shaped response. Figure 3.3.4 illustrates "typical" characteristics likely to be encountered on the switched network whereas Figure 3.3.5 shows the attenuation and group delay responses of poor-quality switched connections^(2,3). References 4 to 7 give detailed results of transmission characteristic field studies carried out on the British PSTN whereas reference 10 gives results of a comprehensive study performed recently (1983) in America.

The major effect of attenuation and delay distortion on the transmission of digital data is the setting of an upper limit on the rate at which data symbols may be transmitted for acceptable levels of intersymbol interference^(3,12). Since the amount of time dispersion that a signal suffers usually increases with increasing levels of attenuation and delay distortion, then for similar levels of intersymbol interference, a channel with characteristics similar to those in Figure 3.3.4 will be able to operate at a higher symbol rate compared with the poorer-quality channel characterised in Figure 3.3.5. In terms of impulse response, the combination of attenuation and delay distortion causes the length of the impulse response to increase, so spreading the energy associated with a particular

symbol over a longer time period. This again results in intersymbol interference when symbols are transmitted in a serial stream. Obviously, the ideal impulse response of a channel is a single impulse occurring at time $t = 0$ which implies an infinitely wide channel bandwidth and so this cannot be achieved even over an ideal bandpass channel. In practice, a "good" channel will have an impulse response which has a rapid rise to its peak value followed by a rapid decay, with increasingly smaller ripples of short duration. The effect of distortion and the presence of echo paths combine to lengthen the impulse response of a channel; an interesting and rather surprising result of the 1968 British field survey revealed that around 75% of all switched connections, including both 2-wire and 4-wire circuits, may exhibit echoes with delay times of upto 4 ms with a signal-to-echo ratio of less than 30 dB. Also, echo delay times of upto 15 ms were recorded on extreme connections⁽⁵⁾. Other surveys however, have shown that the time dispersal of digital signals operating over the telephone network does not often exceed 6-8 ms, with only occasional higher values^(3,8). Methods of cancelling the echoes produced on telephone circuits for both voice and data have been available for some years in the form of adaptive filtering techniques, although it is only recently that microelectronic technology has reached a sufficient level of complexity and advancement to make such cancelling devices economically feasible. A recent article by Messerschmitt⁽¹¹⁾ reviews the techniques of echo cancellation applied to both voice and data transmission and includes an informative section on canceller implementation.

3.3.2 Characteristics of Private Lines

Private lines are used for many applications where direct, point-to-point communication is required between two permanent locations; for example, "hot-lines" between senior Government Departments, direct security lines from Banks to local police stations and dedicated lines for high-speed data transmission.

Theoretically, the characteristics of all private lines can be determined by making end-to-end measurements and so a particular customer will have a knowledge of exactly what is being paid for in terms of the circuit's quality. Whilst this can be provided in practice for short private circuits, say between locations served by the same local exchange, the identity of individual private circuits may become lost when working over long distances involving carrier-based systems. The loss of identity occurs when equipment renewals, maintenance schedules, automatic changeover of carrier station equipment, etc., change the circuit configuration which in turn changes the characteristics of the circuit and increases the likelihood of echo signals from mismatched equipment. However, because more control over the actual routing of private circuits is available compared with random circuit selection of the PSTN, several different grades or schedules of private circuits are available for a number of different applications. At present, British Telecom. provide four basic schedules of private circuit, schedules A to D, where each schedule refers to specific limits for overall characteristics and other circuit parameters⁽²⁾. Circuits which do not meet the specifications of the published scheduled circuits are usually modified by including loss-equalisers and delay-equalisers to force compliance with the required specification, although on some circuits, particularly those with long 2-wire links, equalisation may not be economically feasible due to the very poor unequalised response. Loss-equalisers

operate by increasing the attenuation in various frequency bands in an attempt to flatten the overall loss characteristics; clearly, to offset equaliser loss, extra amplification may also be necessary. Delay-equalisers operate in a similar manner but insert extra delay over various narrow bands such that all signal components are delayed by similar amounts. Consequently, delay equalisers may increase signal propagation times⁽¹²⁾. Table 3.1 lists the attenuation and group delay limits for schedule A-D circuits. Figures 3.3.6 and 3.3.7 show graphically the characteristics of schedule D circuits. With reference to the table and figures, it can be seen that schedule A represents lower quality circuits compared with the other three, with the quality, or tightness of specification increasing to a maximum for schedule D circuits.

Although the characteristics of circuits complying with the specifications of schedules A-C may well be satisfactory for data transmission of 300, 600 and 1200 bit/s using simple binary F.S.K. techniques (for example, Datel 200^(3,4)), transmission of data at higher rates involving multi-level phase or amplitude-phase modulation will require the tighter delay specifications defined under schedule D. In order to minimise the amount of overall delay distortion and to minimise the amount of delay equalisation needed, schedule D circuits (which are identical to the British Post Office Tariff "T" circuits and virtually the same as the C.C.I.T.T. specification M1020 for high-quality circuits suitable for data transmission), must be carefully routed over the available line plant. In particular, since filters used in carrier multiplexing equipment and the inductors used for the loading of long junction routes both increase delay distortion, the number of individual carrier systems used and the length of loaded links should be restricted by careful planning and design.

For completeness, Figures 3.3.8 and 3.3.9 show four examples of attenuation and group delay characteristics complying with schedule D specifications. Curves a) - c) show typical worst-case weighted average characteristics for various telephone regions obtained through actual field studies whilst curve d) illustrates a typical weighted average "good" circuit for the U.K. as a whole⁽⁴⁻⁷⁾.

3.4 Other Factors Influencing Data Transmission Over Telephone Circuits

As well as introducing attenuation distortion and delay distortion, telephone circuits also introduce other impairments which can be generally referred to as "noise". Although the distortions and forms of noise can be present on both voice and data carrying circuits, it is generally true to say that data signals will be less tolerant to all forms of signal impairment when compared with analogue voice signals. For example, delay distortions of say 15 ms would not be distinguishable by a human ear, whereas delay distortions of fractions of a millisecond may well cause some high-speed data systems to fail completely. For this reason it is important for a system designer to understand the causes and effects of all forms of signal impairments introduced by telephone circuits and to include suitable methods of countering these effects in the overall system design. In practice, it will not be possible to design a system capable of countering all levels of every possible impairment simultaneously. Indeed, the presence of one type may well prevent the elimination of another and so, as in the case of the frequency characteristics described earlier, a designer is faced with the task of attempting

to discover typical limits for the various impairments likely to be encountered over the majority of telephone circuits. The following subsections describe the forms of signal impairment likely to cause erroneous detection of digital data signals transmitted over both switched and private telephone lines. These impairments are introduced by the network itself and so are not dependent on the method of data transmission being used. Clearly, different methods of transmission will have different tolerances to particular impairments and will also introduce additional operational limits and constraints by virtue of the tolerance to parameters internal to the actual modem design.

3.4.1 Frequency Modulation Effects

Of the three main types of frequency modulated induced noise, frequency offsets and sudden phase jumps or phase hits can only occur over telephone circuits containing frequency translation equipment, i.e. over carrier systems. Phase jitter may occur over any system which can suffer interference from 50 Hz power frequencies. Frequency offsets are caused when the oscillators used in the transmitter and receiver equipment of carrier links operate at slightly different frequencies, the difference in frequency usually being less than ± 0.5 Hz but occasionally reaching upto ± 5 Hz for the U.K. network⁽⁵⁾. A slightly better performance has been recently cited for the American A.T. & T. network⁽¹⁰⁾. Private circuits specified under schedule D are required to have a maximum offset of ± 2 Hz⁽²⁾.

Sudden, quite large signal phase changes can also occur over carrier links when for example, an oscillator receives a correction signal to restore its output frequency or an automatic changeover is initiated at either end of the link. (Equipment at carrier terminal stations are connected to a synchronising network which endeavours to contain frequency drift of

oscillators within high specifications. However, aging of equipment, malfunction of crystal ovens and faulty units all contribute to small amounts of drift which over time may become significant.) The effect of frequency offsets and phase errors can be illustrated by simple example. Figure 3.4.1 shows the basic format of a carrier system where the unmodulated carrier wave is assumed to be $\cos(\omega_c t)$. The demodulator's carrier wave is represented by $\cos((\omega_c + \Delta\omega)t + \phi)$, where $\Delta\omega t$ implies the frequency difference between the modulator and demodulator carrier frequencies and ϕ represents a constant phase error. If a single tone, $A\cos(\omega_m t)$ is to be transmitted as a single sideband suppressed carrier signal, then assuming that the upper sideband is selected, the signal at the input to the demodulator is:-

$$x(t) = K\cos(\omega_c + \omega_m)t \quad (3.4.1)$$

The demodulator output is, therefore:-

$$y(t) = K\cos(\omega_c + \omega_m)t \cdot \cos((\omega_c + \Delta\omega)t + \phi) \quad (3.4.2)$$

$$= \frac{K}{2} \{ \cos((2\omega_c + \omega_m + \Delta\omega)t + \phi) + \cos((\omega_m + \Delta\omega)t - \phi) \} \quad (3.4.3)$$

Low-pass filtering results in;

$$z(t) = \frac{K}{2} \cos((\omega_m - \Delta\omega)t - \phi) \quad (3.4.4)$$

From equation (3.4.4) it can be seen that a frequency offset of Δf Hz will shift the spectrum of the signal at the output of the carrier system by an amount equal to the difference between the two carrier frequencies. Similarly, the phase error existing between the two carriers will also be transferred to the output spectrum. For voice transmission, subjective tests have shown that frequency offsets of upto ± 10 Hz are

tolerable to the majority of users⁽¹³⁾. For data transmission however, the effect of relatively small offsets can be catastrophic, particularly if a modulation method based upon phase deviations is employed within the modem; Chapter 8 presents a more detailed study of the effects of frequency offsets and phase errors and introduces various methods of carrier-phase correction necessary for the transmission of high-speed data over telephone circuits.

Phase hits are defined as sudden phase changes which can occur almost instantaneously. From equation (3.4.4) phase hits will also be passed directly to the output of a carrier system and can again cause complete failure of data transmission systems, at least until the system can re-synchronise to the new phase of the receiver's input signal. Phase hits of up to 30° have apparently been recorded over older, valve carrier systems although on more modern systems, 15° seems to be a realistic figure^(10,14) and is also used as a criterion for phase hit measurements. (Data circuits specified under C.C.I.T.T. recommendation M.1060 are required to have less than 10 phase hits that exceed 15° within any 15 minute period.) Phase hits can also be considered as an impulsive form of phase noise.

Phase jitter is the term used to describe continuously fluctuating phase of the received signal at the output of a telephone line. It can be distinguished from phase noise since jitter tends to be a sinusoidal process occurring at power frequencies and harmonics, whereas phase noise is a more random process. For example, phase jitter would occur if the phase of either oscillator in a carrier system was modulated by the 50 Hz (or 60 Hz) power-line frequencies or its harmonics. The superposition of a deterministic interfering signal can also cause phase jitter since a regular phase beat can be produced so deviating the zero-crossings of a signal in a regular fashion.

Although the presence of phase jitter is now widely acknowledged, as is its detrimental effect on the performance of high-speed data modems^(10,14,15-18), little information is available regarding the main causes of jitter in telephone networks. However, a compilation of jitter measurements by Harvey⁽¹⁴⁾ and a recent American study by Carey et. al.⁽¹⁰⁾ indicate that jitter in the form of sinusoidal phase fluctuations occurring at the fundamental, third and fifth harmonics of power frequencies may be present on the majority of telephone circuits, with peak-to-peak values between 5° and 15° , although jitter components up to 30° peak-to-peak have apparently been recorded. Furthermore, phase jitter components at the power frequency harmonics may be present simultaneously with different peak-to-peak values. Unless the data receiver is able to track the phase jitter components and remove them from the received signal by some means, the phase of each received sample will vary in a sinusoidal manner so seriously reducing the tolerance of the system to additive noise. Most high-speed data modems use carrier phase tracking and correction loops in order to ensure a phase-coherent frequency reference at the receiver modem. However, for reasons of noise performance and the fact that other non-jitter forms of phase variations tend to vary slowly with time, the tracking loop bandwidth is usually set to a low value making it impossible for the loop to track the relatively high frequency components caused by phase jitter. Consequently, either a compromise between noise performance and jitter tracking ability must be made or separate jitter-suppression loops must be included in the receiver design. The latter approach will in general lower the performance of the modem in terms of tolerance to noise when only small amounts of jitter are present⁽¹⁴⁾.

Phase jitter therefore is a considerable problem when attempting to transmit high data-rates over telephone circuits. Chapter 8 investigates phase jitter in more detail and presents some conventional and new methods of jitter estimation and elimination. A simple model of phase jitter may be formulated by considering the overall phase error of the output signal after demodulation by an oscillator running at the modem's carrier frequency;

$$\phi(t) = \phi_o + 2\pi\Delta ft + \phi_j(t) \quad (3.4.5)$$

where ϕ_o is some constant phase error, Δf is a constant frequency offset introduced by the telephone channel and $\phi_j(t)$ represents the phase jitter components.

If the phase jitter is assumed to be interference from power frequencies and harmonics, then $\phi_j(t)$ may be further characterised by;

$$\phi_j(t) = \sum_{k=1}^m a_k \sin(2\pi \cdot 50 \cdot k \cdot t + p_k) \quad (3.4.6)$$

where a_k and p_k represent the peak value and positional offset of the k^{th} harmonic of 50 Hz present in the combined jitter term $\phi_j(t)$. The aim of a carrier-phase correction loop is therefore to reduce $\phi(t)$ in equation (3.4.5) to zero.

3.4.2 Amplitude Modulation Effects

Examples of amplitude modulation induced impairments are modulation noise, transient interruptions, amplitude jitter and sudden level changes.

Modulation noise appears as amplitude modulation of the signal being carried over carrier links by bandlimited white noise and occurs in 1-2 second bursts. On older equipment, microphony was a major cause of

modulation noise but with modern solid-state equipment, intermodulation distortion is probably the major contributor⁽³⁾. Compared with other forms of noise, modulation noise, where it exists, is relatively unimportant except perhaps on the rare occasion that in-line amplifier units are severely overloaded⁽¹⁰⁾.

Transient interruptions or dropouts appear as breaks in transmission lasting from between 1 ms to 100 ms. Impairments of this type can be caused by power-supply failures, cable faults, etc., and will clearly have a considerable affect on the error rate of high-speed data systems, although due to the transient response of typical telephone channels, interruptions of less than 0.1 ms will not cause a significant drop in received signal level. A dropout is assumed to have occurred if a reduction in received signal strength of more than 10 dB is recorded over a period of 3 ms or more; dropout rate is taken as the number of dropouts observed within any 15 minute interval. In general, because of the extra amounts of equipment involved, dropouts are more likely to occur over longer telephone circuits although tests performed on over 5000 connections in America have indicated that around 90% of telephone circuits will on average experience only one measurable dropout in any 15 minute interval⁽¹⁰⁾.

Amplitude jitter is defined as the deviation of the peak value of a single tests tone from its nominal value. Although measurements of amplitude jitter have apparently been made⁽¹⁰⁾, it has not been made clear how the measuring equipment distinguished between the effects of impulsive noise and "pure" amplitude jitter. A better definition would perhaps include the proviso that the jitter should be sinusoidal in form, as in the case of phase jitter, but no information has been found to support this hypothesis.

As with dropouts, sudden signal level changes, also called gain hits, are more likely to occur over longer and more complex circuits and tend to be less of a problem over shorter circuits. Gain hits are usually restricted to around 1-2 dB, but can occasionally exceed 6 dB^(3,10). Clearly, the effect of particular levels of gain hits on the detection of digital data will depend on the type of system (modulation) being used in the modem, although in general, a level change which reduces the received signal power will obviously reduce the systems tolerance to noise by some degree.

3.4.3 Additive Noise

Additive noise may be further divided into three main types viz., impulsive noise, crosstalk and uniform spectrum or white noise.

Of all the types of signal impairments introduced by telephone circuits, impulsive noise is probably the biggest single factor affecting the error rate of digital data transmission systems⁽²⁻⁴⁾ but since it generally arises from line-plant deficiencies, it is also the least predictable form of noise. Impulsive noise comprises short pulses of noise with random amplitudes and durations and is measured by counting the number of noise pulses exceeding a given threshold level in a 15 minute observation period within a given bandwidth. For example, private circuits operating under schedule D specifications must have less than 18 impulsive noise counts exceeding -21 dBmO in any period of 15 minutes across the frequency range 300-3000 Hz, where dBmO refers to the ratio of the noise power to the test level at a particular point on the circuit called the relative zero level point and where the test level is 1 mw into 600Ω. Impulsive noise can be caused from a variety of different sources including common impedance coupling between circuits sharing the same exchange batteries, overloading of amplifiers used in multiplexed carrier systems, electro-

magnetic coupling of switching transients and contact noise due to the small movements of switch contacts used in mechanical selectors and relays. Since private lines do not pass through any exchange switches and are not connected to exchange battery supplies, impulsive noise is generally less of a problem over these lines when compared with switched lines. Over switched lines, impulsive noise is the predominant type of noise and can completely swamp uniform spectrum and other forms of noise. Due to the predominance of impulsive noise, it can be quite easily measured by arranging for the threshold used in the measuring equipment to be set higher (usually around 13 dB higher) than the mean power of any background random noise, so ensuring that indications are most likely to be due to noise with a different source than the random noise. Figures 3.4.2 and 3.4.3 show the occurrence of impulsive noise over typical switched connections established over circuits containing electro-mechanical switches. As modern electronic switches replace the old electromechanical Strowger type of selectors, the level and frequency of occurrence of impulsive noise due to switch movement and high resistance contacts would be expected to reduce significantly, although no comparisons could be found in the literature.

Crosstalk due to capacitive and inductive coupling can exist to some degree on both switched and private circuits but is usually at a low enough level to be insignificant compared with other sources of noise. Four wire circuits can also suffer from transmit-to-receive crosstalk, the limit being 45 dB for schedule D circuits although 60 dB or more is typical for most modern equipment^(2,4).

Strictly, white noise is any random noise which has a uniform power distribution over a very wide range of frequencies. Figure 3.4.4a shows the power spectrum of such a noise source with a uniform 2-sided

power spectral density of $\frac{N_0}{2}$ W/Hz. From the Wiener-Khintchine theorem^(3,19) the autocorrelation function can be found to be a single impulse of strength $\frac{N_0}{2}$ occurring at a time displacement $\tau=0$, as shown in Figure 3.4.4b. Two important observations may be made from Figure 3.4.4. Firstly, there will be no correlation between any two samples of white noise separated by a time interval $\tau>0$ and secondly, the average power of a white noise waveform will be infinite. Obviously, a noise source with infinite average power is practically unrealisable and the associated concept of noise with a uniform power spectral density over all frequency is equally impractical for the majority of communications systems. Clearly, if for no other reason than the presence of transmitter and receiver filters, practical systems will be bandlimited to some degree. A more realistic definition of white noise is therefore one which allows the noise power spectral density to be uniform over a given bandwidth, where the bandwidth is much greater than the range of frequencies being used in the particular communication system. Figure 3.4.5a shows the power spectrum of a bandlimited noise source with a uniform 2-sided power spectral density of $\frac{N_0}{2}$ over the range of frequencies $-B$ to $+B$ Hz. If $G(f)$ is the power spectrum of the noise waveform and $R(\tau)$ the autocorrelation function, then;

$$R(\tau) = \int_{-B}^B G(f) e^{j2\pi f\tau} df \quad (3.4.7)$$

$$= \frac{N_0}{2} \int_{-B}^B e^{j2\pi f\tau} df \quad (3.4.8)$$

$$= \frac{N_0}{2} \left[\frac{e^{j2\pi f\tau}}{j2\pi\tau} \right]_{-B}^B \quad (3.4.9)$$

$$R(\tau) = N_0 B \frac{\sin(2\pi B\tau)}{2\pi B\tau} \quad (3.4.10)$$

which is shown in Figure 3.4.5b.

Clearly, as $B \rightarrow \infty$, $R(\tau) \rightarrow \frac{N_0}{2} \delta(\tau)$, as expected from the non-bandlimited white noise source considered earlier. Making the same observations (as made earlier) to the case of bandlimited "white" noise, it can be seen that noise samples over intervals of about $\frac{1}{2B}$ seconds will be correlated and the average power of the noise will be $N_0 B$ watts. It is also important to note that if pure white noise is passed through an ideal lowpass filter with a bandwidth B Hz, the output from the filter is bandlimited (or coloured) noise with a finite average power and has correlated output samples. Furthermore, since the average noise power at the output of such a filter is directly proportional to the filter bandwidth, the analysis above also supports the statement made in Section 3.3, which advised the use of noise-rejection filters prior to line amplifiers on long audio and carrier-based telephone circuits.

White noise on telephone lines is mainly due to thermal (or Johnson) noise, shot noise and partition noise⁽¹⁹⁾. The noise is measured by using a psophometer⁽²⁰⁾ with a weighting filter centred on 1700 Hz with the 3 dB points set at 1100 Hz and 2300 Hz. The sending end of the connection is terminated in 600Ω. Published results of field surveys suggest that good long-distance connections are expected to have noise powers lower than -60 dBmOp whereas for longer international circuits, -40 dBmOp is typical⁽⁴⁻⁷⁾. The unit dBmOp refers to the ratio of the psophometrically measured noise power to a test level of 1 mW at the relative zero level point. For schedule D private circuits, the specified limit for random circuit noise is quoted at -45 dBmOp, implying a maximum permissible noise power of -45 dB relative to 1 mW.

Although the measurement of absolute levels of noise power is useful for the comparison of different types of connection, a much more important factor as far as the transmission of signals is concerned is the ratio of signal power to noise power, the signal-to-noise ratio, or SNR. Clearly, the SNR is dependent upon the actual power of the signal at the particular test point and it would therefore be beneficial, from an SNR point of view, to increase the signal power to some high value. It should also be noted that the SNR of a system can take different values depending upon the point of measurement. This follows since the signal will suffer attenuation as it progresses through the line plant and can be amplified at various locations. Similarly, the noise, if viewed as resulting from a single point source, will also suffer attenuation but the level of noise power will change as it passes through filters preceding line amplifiers. However, as far as modem design is concerned, the only important SNR is that which occurs at the output of the telephone connection.

As mentioned above, the power of a data signal transmitted over a telephone circuit should be as high as possible to maximise the SNR measured at the line output. However, too high a signal power could cause overloading of amplifiers and repeaters used in multiplexed carrier systems. With the above points in mind, internationally agreed standards have been set by the C.C.I.T.T. which limit the maximum signal power emitted from apparatus connected to both switched and private lines.

In brief, these limits require that the absolute maximum emitted power from subscribers apparatus should not exceed 1 mW and that the mean power of data systems operating over carrier circuits should not exceed the standard channel loading specification of -15 dBm. Assuming that only a small proportion of circuits carry data signals at any one time, the recommended value is -10 dBm for both switched and private connections.

(The figure -10 dBm assumes simplex operation; duplex working would necessitate the lowering of signal power in both directions by a further 3 dBm to ensure that the total power remains within the -10 dBm limit.) Figures 3.4.6 and 3.4.7 show the maximum allowable one-minute-mean power levels of individual spectral components of signals emitted from apparatus connected to the PSTN and private circuits respectively, whereas Figure 3.4.8 shows the maximum permissible power levels for spectral components above 3400 Hz⁽⁴⁾. Based upon the "typical" levels of noise power given earlier, -60 dBmOp and -40 dBmOp and assuming that the signal power at the output of a line system is identical to the input signal power, the end-of-line SNR for a transmitted power of -10 dBm can be calculated to be 50 dB and 30 dB respectively, although due to attenuation, 30-40 dB is probably a more realistic range for the majority of telephone connections. However, a study by Ridout and Rolfe⁽⁵⁾ suggests an even better performance based on measurements made over telephone channels in four telephone regions of the U.K., only 1% of the connections tested having an SNR of less than 45 dB. (In their tests, Ridout and Rolfe used a 1700 Hz test tone at a level of -6 dBm rather than the recommended -10 dBm; applying a -4 dB adjustment to their quoted results yields an SNR of greater than 41 dB for majority of the connections tested. It should be noted however, that the quoted SNR values are based upon uniform spectrum noise only and do not include contributions from impulsive noise.)

3.5 Representation of Line Characteristics and Impairments for

Theoretical and Practical Development of Modems

Sections 3.3 and 3.4 above have given an insight to the variations of line characteristics and the type and values of the most common impairments likely to be encountered on telephone circuits. Due to the very wide range of possible channel conditions, it is clearly unrealistic

to test digital data systems over a single circuit and equally impractical to actually implement every conceivable design and to assess performance by obtaining statistical measures over a large number of randomly selected line plant. In the initial theoretical development stage, the most useful method of performance evaluation is to compare the performance of different systems in terms of tolerances to characteristic variations and signal impairments. In the course of this study, three test channels, each introducing various amounts of attenuation and delay distortion were modelled to compare different system designs using a digital computer to simulate the complete system, including the telephone circuit. In addition, the simulations included the facilities of introducing and varying the amounts of frequency offsets, constant phase error, phase jitter, amplitude variations and signal-to-noise ratio. Although the actual types of noise introduced over a telephone circuit are not necessarily Gaussian in form, it has been shown both experimentally and theoretically that the relative tolerances of different systems to additive Gaussian noise gives a good indication to their relative tolerances to actual line noise, including impulsive noise. Moreover, it has also been shown that if one system has a better tolerance to additive Gaussian noise, then in general it will have a better tolerance to other types of additive noise found on telephone circuits^(3,13,14,16,17,21-23). Also, Gaussian noise may be readily produced in the laboratory and may be easily generated using digital computers. For these reasons, line noise is usually modelled by a random process with a Gaussian probability density function with a zero mean and two-sided power spectral density of $\frac{N_0}{2}$ W/Hz. If the process is considered white then all sample values are uncorrelated and as the process is Gaussian, they will also be statistically independent. Also, if white Gaussian noise is applied to a

time invariant linear system, for example a filter, the output will also be Gaussian in form but with sample values being correlated to some degree^(13,14,21). Chapters 4 and 5 explain the system model and illustrate the characteristics of the test channels in more detail whereas Chapter 6 presents the results of computer simulations for different systems using the model.

Finally, after a particular design has been theoretically studied, successfully simulated and eventually physically implemented, its actual performance may be evaluated by using a test-bed comprising a data signal generator, a telephone line simulator⁽⁶⁾ and a bit error-rate analyser. References 26 and 27 give model numbers and manufacturers of test-bed equipment available at Loughborough University.

3.6 References

- 1 Richards, D.L., "Telecommunication by Speech, the Transmission Performance of Telephone Networks", Butterworth, London, 1973.
- 2 "Handbook of Data Communications", Editor, G.B.Bleazard, National Computing Centre Ltd, 1982.
- 3 Clark, A.P., "Principles of Digital Data Transmission", Pentech Press, 1976.
- 4 Williams, M.B., "The Characteristics of Telephone Circuits in Relation to Data Transmission", Post Office Elec. Eng. J., Vol. 59, Part 3, pp. 151-162, October 1966.
- 5 Ridout, P.N. and Rolfe, P., "Transmission Measurements of Connections in the Switched Telephone Network", Post Office Elect. Eng. J., Vol. 63, pp. 97-104, 1970.
- 6 Groves, K. and Mackrill, P., "A Line-Transmission Simulator for Testing Data Transmission Systems", Post Office Elect. Eng. J., Vol. 63, pp. 117-122, 1970.
- 7 Monk, C., "Transmission Characteristics of Connections in the Switched Telephone Network Measured in the 1967 Field Study", Post Office Research Centre Report No. 31403000, Post Office Research Centre, Martlesham Heath, Ipswich.

- 8 Harvey, J.D., Clark, A.P. and Driscoll, J.P., "High-speed Line Modems", Final Report, Contract No. K/MMZ11a/616, Loughborough University Technology.
- 9 Sinnema, W., "Electronic Transmission Technology", Prentice-Hall Inc., 1979.
- 10 Carey, M.B. et al, "1982/83 End Office Connection Study: Analog Voice and Voiceband Data Transmission Performance Characterization of the Public Switched Network", AT & T Bell Labs. Tech. J., Vol. 63, No.9, pp. 2059-2119, November 1984.
- 11 Messerschmitt, D.G., "Echo Cancellation in Speech and Data Transmission", IEEE J. on Selected Areas in Communications, Vol. SAC-2, No. 2, pp. 283-297, March 1984.
- 12 Jones, I.O. and Adcock, R.C., "Group Delay in the Audio Data Network", Post Office Elect. J., Vol. 64, pp. 9-15, 1971-72.
- 13 Carlson, A.B., "Communication Systems", McGraw-Hill Inc., second edition, 1975.
- 14 Harvey, J.D., "Synchronisation of a Synchronous Modem", S.E.R.C. Report No. GR/A/1200.7, 1980.
- 15 Matyas, R and McLane, P.J., "Decision-Aided Tracking Loops for Channels with Phase Jitter and Intersymbol Interference", IEEE Trans. on Comms., Vol. COM-22, No. 8, pp. 1014-1023, August 1971.

- 16 Ho, E.Y. and Spaulding, D.A., "Data Transmission Performance in the Presence of Carrier Phase Jitter and Gaussian Noise", Bell Sys. Tech. J., Vol. 51, pp. 1927-1931, October 1972.
- 17 Foschini, G.J., Gitlin, R.D. and Weinstein, S.B., "On the Selection of a Two-Dimensional Signal Constellation in the Presence of Phase Jitter and Gaussian Noise", Bell Sys. Tech. J., Vol. 52, No. 6, pp. 927-965, July-August 1973.
- 18 Kearsley, N. and McLintock, R.W., "Jitter in Digital Telecommunication Networks", B.T. Engineering, Vol. 3, pp. 108-116, July 1984.
- 19 Connor, F.R., "Noise", Edward Arnold (Publishers) Ltd, 1973.
- 20 Richards, D.L., "Telecommunication by Speech", Butterworth and Co. (Publishers) Ltd., 1973.
- 21 Schwartz, M., "Information Transmission, Modulation and Noise", McGraw-Hill Inc., Third edition, 1980.
- 22 Ho, E.Y. and Yeh, Y.S., "A New Approach for evaluating the Error Probability in the Presence of Intersymbol Interference and Additive Gaussian Noise", Bell Sys. Tech. J., pp. 2249-2256, November 1970.
- 23 Shimbo, O and Celebiler, M.I., "The Probability of Error Due to Intersymbol Interference and Gaussian Noise in Digital Communication Systems", IEEE Trans. on Comm. Tech., Vol. COM-19, No. 2, pp. 113-119, April 1971.

- 24 Mertz, P. and Mitchell, D., "Transmission Aspects of Data Transmission Service Using Private Line Voice Telephone Channels", Bell Sys. Tech. J., pp. 1451-1486, November 1957.
- 25 Hall, R.D., "Crosstalk Limitations of 2.048 mbit/s Digital Systems on Junction Cables", B.T. Tech. J., Vol. No. 2, pp. 68-77, October 1983.
- 26 AEA Electronic Ltd., "S3 Telephone Channel Simulator", Operating Handbook, Ontario, Canada, Sept. 1980
- 27 Telecommunications Techniques Corp., "Fireberd 1500 Data Error Analyzer", Operating Manual, Gaithersburg, Maryland.

	SCHEDULE			
	A	B	C	D
ATTENUATION RESPONSE dB Re. 800 Hz.				
300-500 Hz.	-7 to +12	-3 to + 10	-2 to +7	-2 to +6
500-2000 Hz.	-7 to +8	-3 to +6	-1 to +4	-1 to +3
2000-2600 Hz.	NS	-3 to +6	-1 to +4	-1 to +3
2600-2800 Hz.	NS	-3 to +10	-1 to +4	-1 to +3
2800-3000 Hz.	NS	-3 to +10	-2 to +7	-2 to +6
GROUP DELAY RESPONSE ms Re. Min.				
500-600 Hz.	NS	NS	NS	3.0
600-1000 Hz.	NS	NS	NS	1.5
1000-2600 Hz.	1.25	1.0	1.0	0.5
2600-2800 Hz.	NS	NS	NS	3.0

Table 3.1: Specified characteristics for schedule A-D circuits

NS = not specified

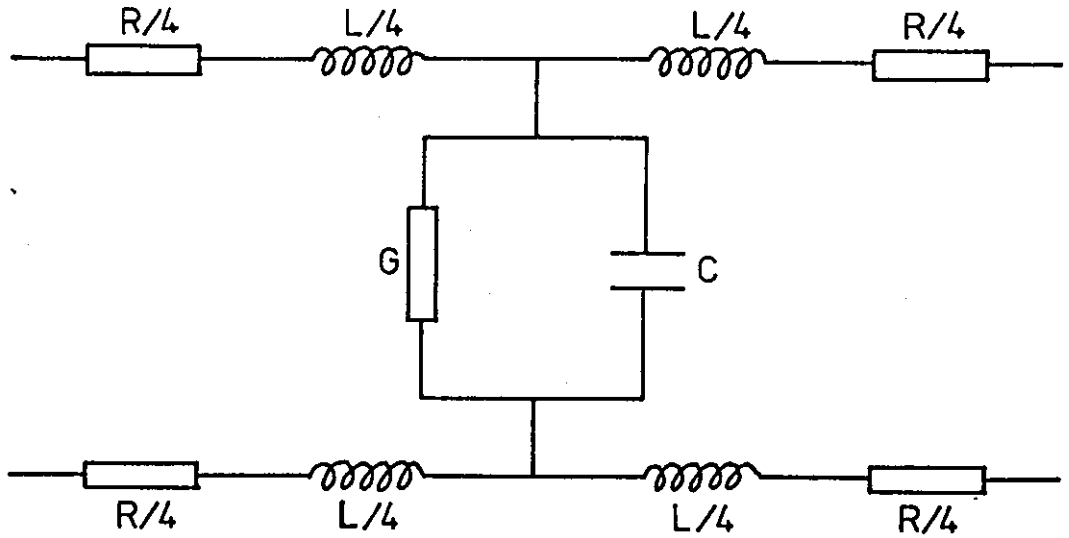


Figure 3.3.1: Representation of elemental length of transmission line by the primary line coefficients

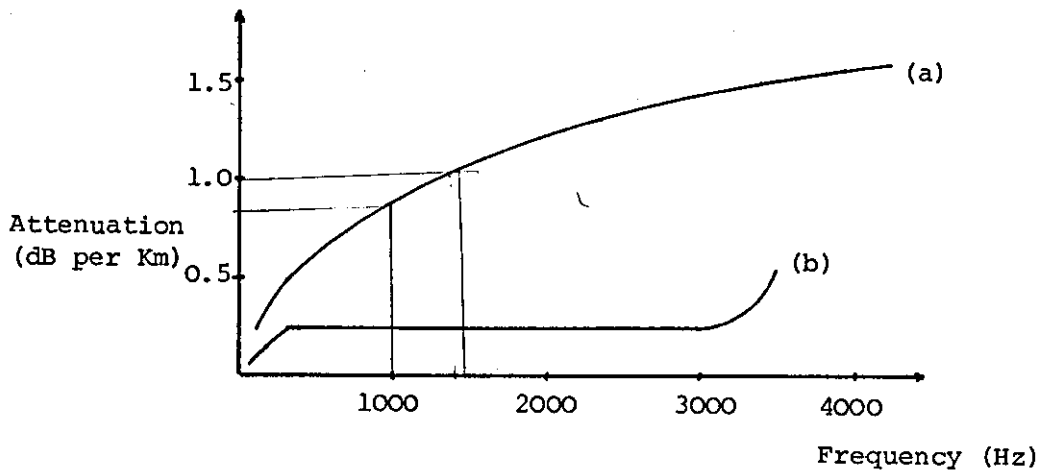
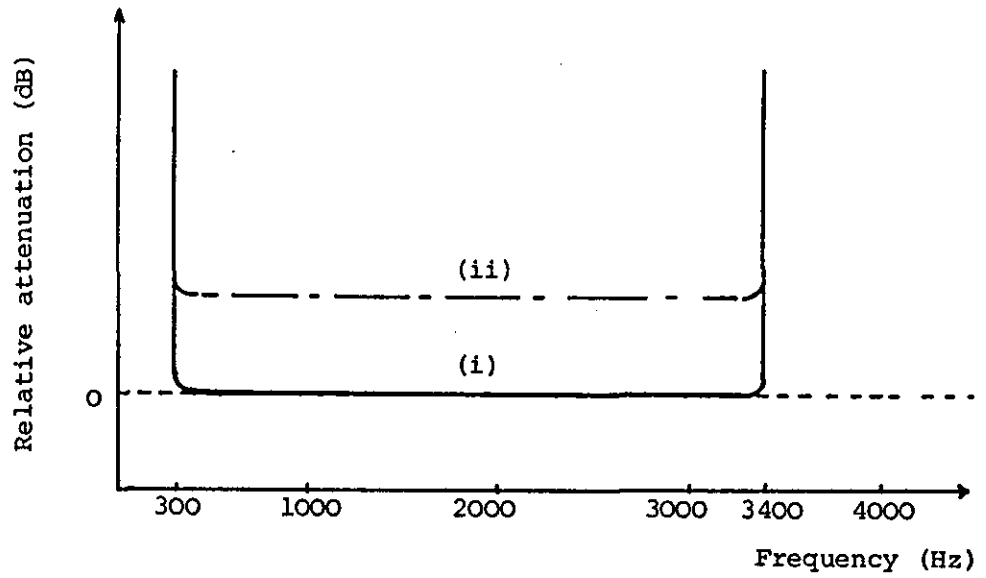


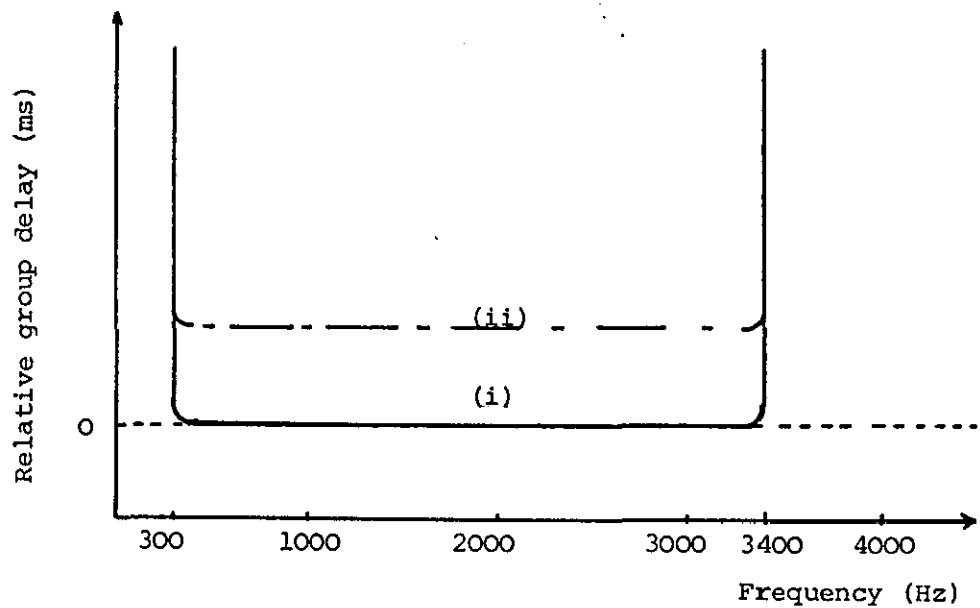
Figure 3.3.2: Frequency response of 0.9 mm cable

(a) Unloaded

(b) Loaded

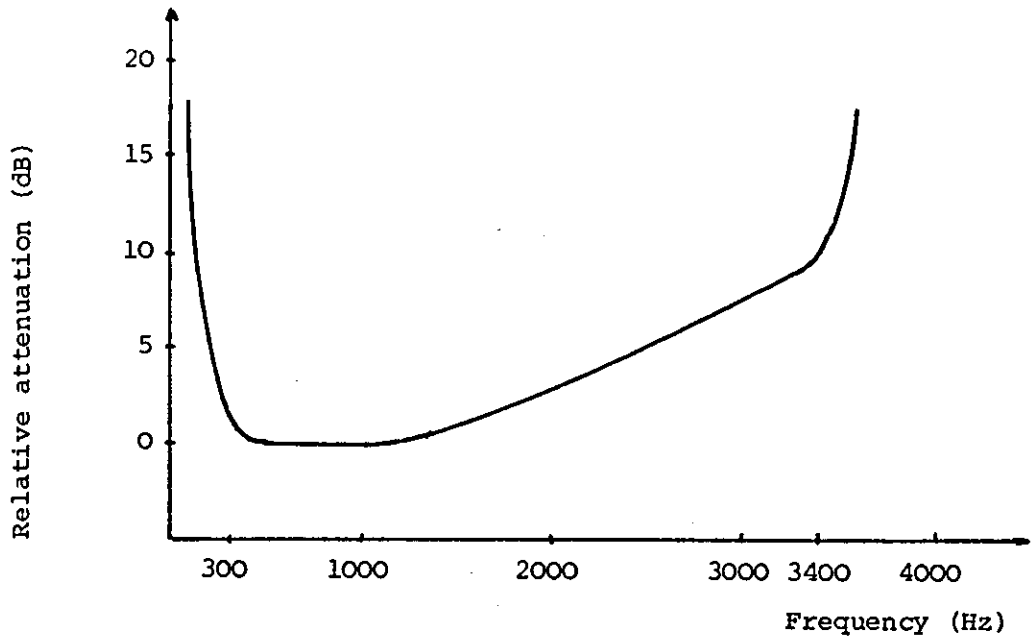


- a) (i) Zero attenuation, zero attenuation distortion
 (ii) Constant attenuation, zero attenuation distortion over frequency range 300-3400 Hz

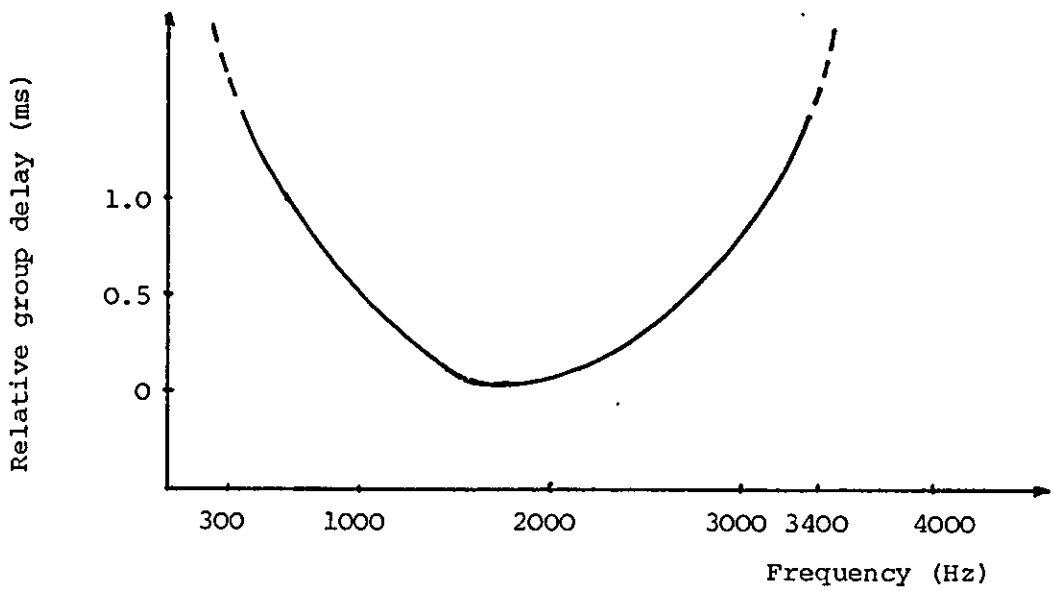


- b) (i) Zero delay, zero delay distortion
 (ii) Constant delay, zero delay distortion over frequency range 300-3400 Hz

Figure 3.3.3: Attenuation and group delay characteristics for an ideal telephone channel



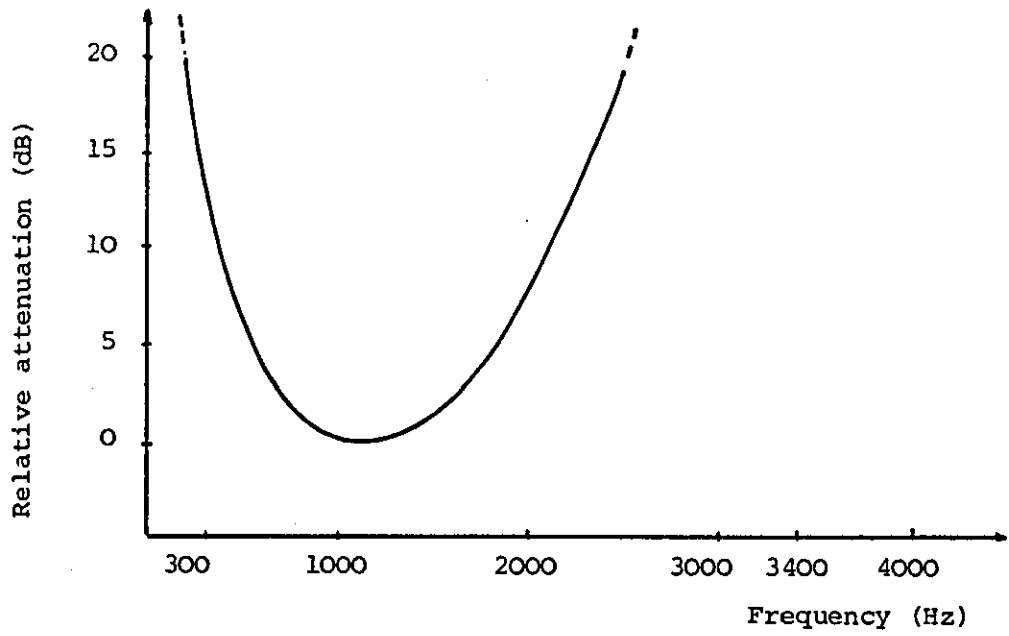
a) Attenuation characteristic



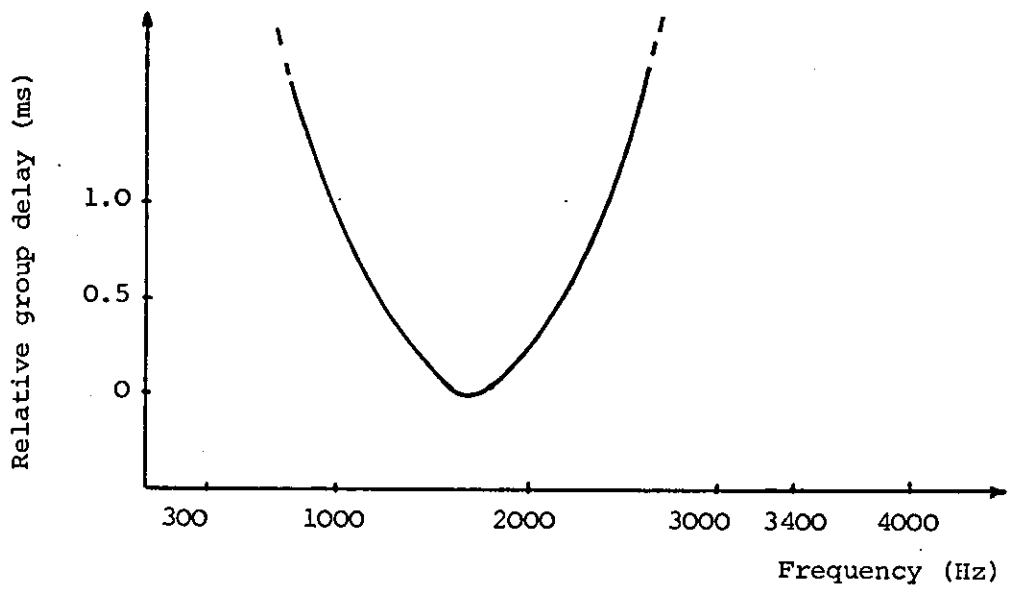
b) Group delay characteristic

Figure 3.3.4: Characteristics of a typical telephone channel

(unequalised) ref. 2-7,10



a) Attenuation characteristic



b) Group delay characteristic

Figure 3.3.5: Characteristics of a poor telephone channel

(unequalised) ref. 2-7,10

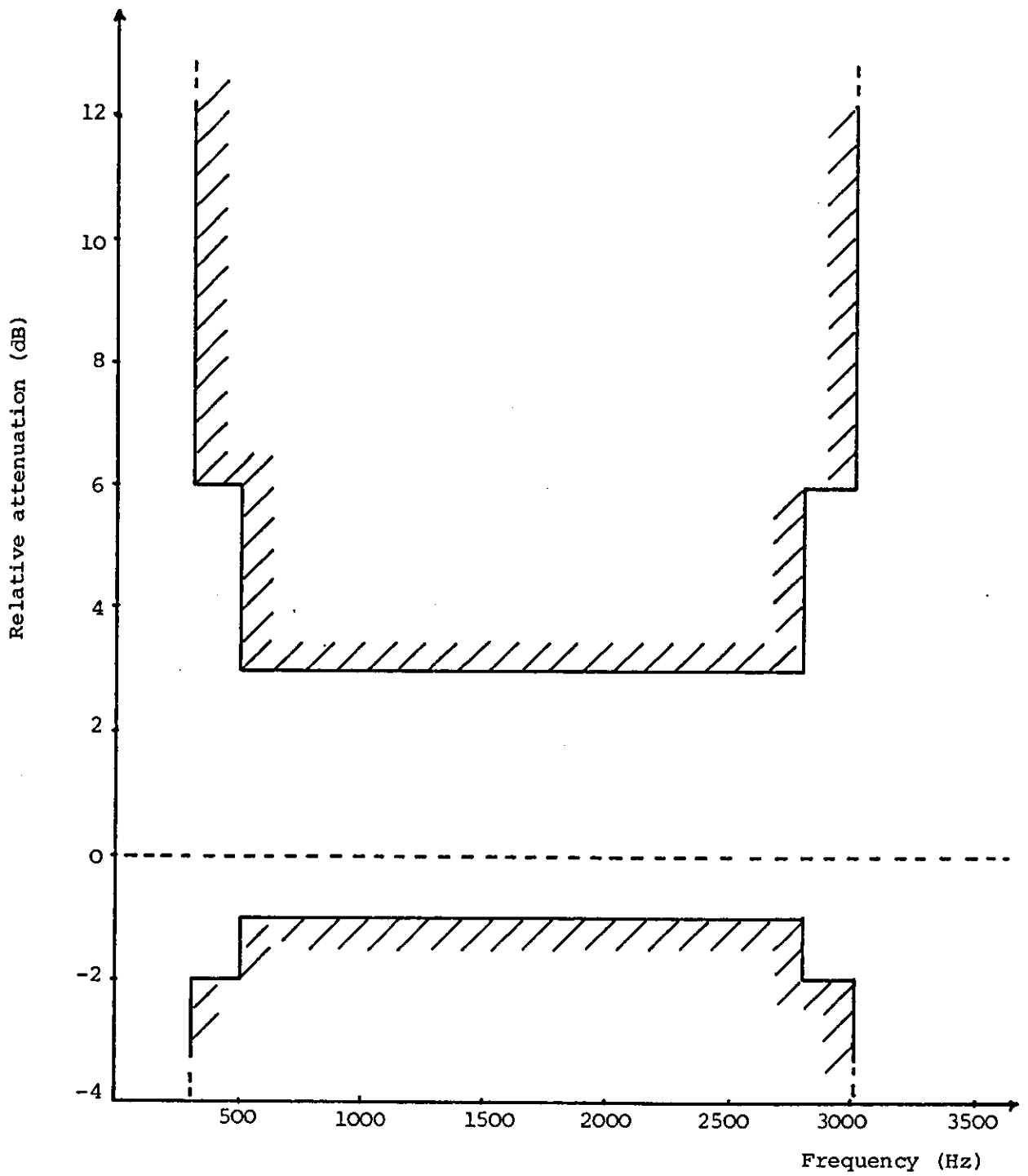


Figure 3.3.6: Limits for attenuation characteristic for
schedule D private circuits

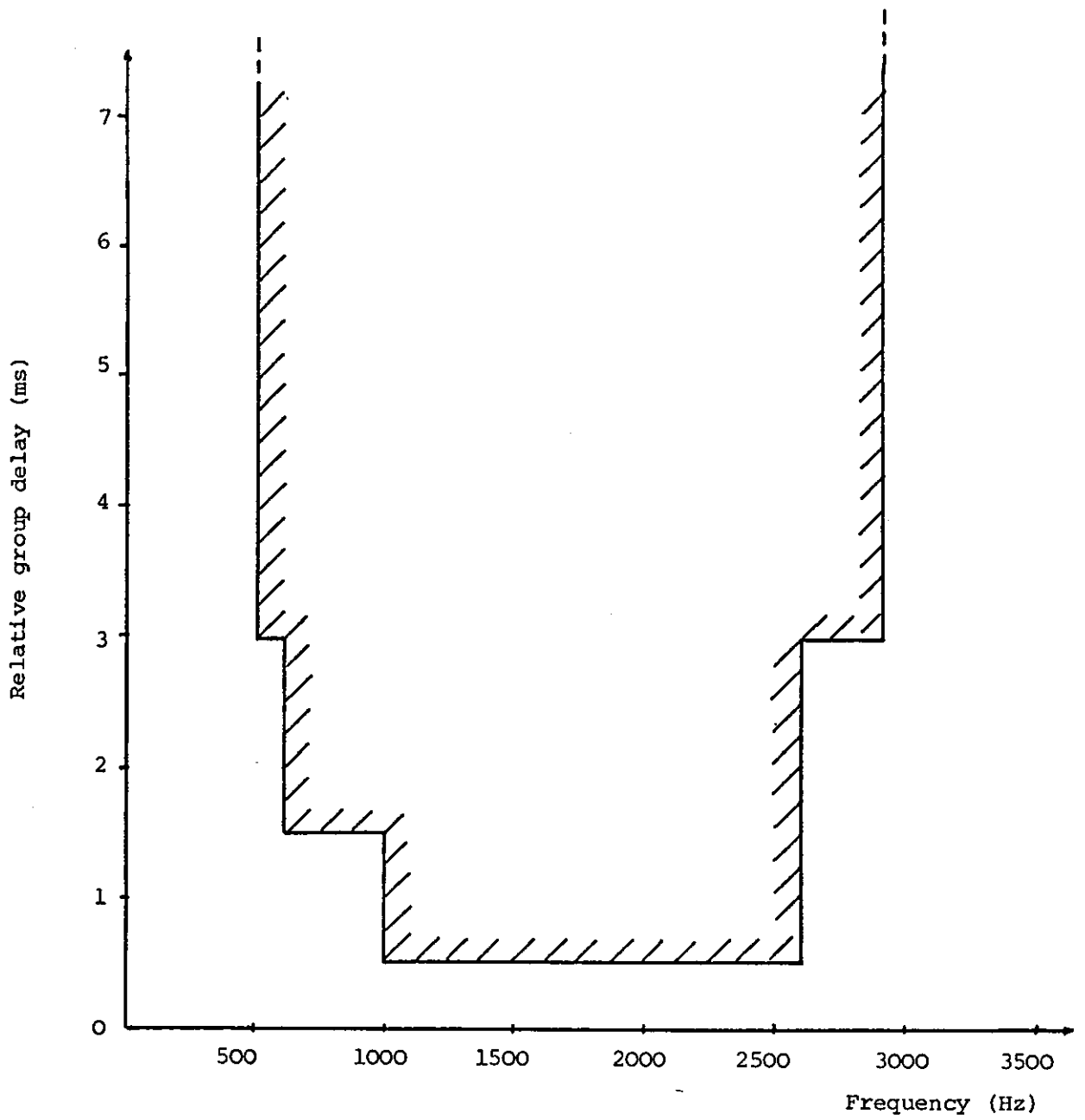


Figure 3.3.7: Limits for group delay characteristic for
schedule D private circuits

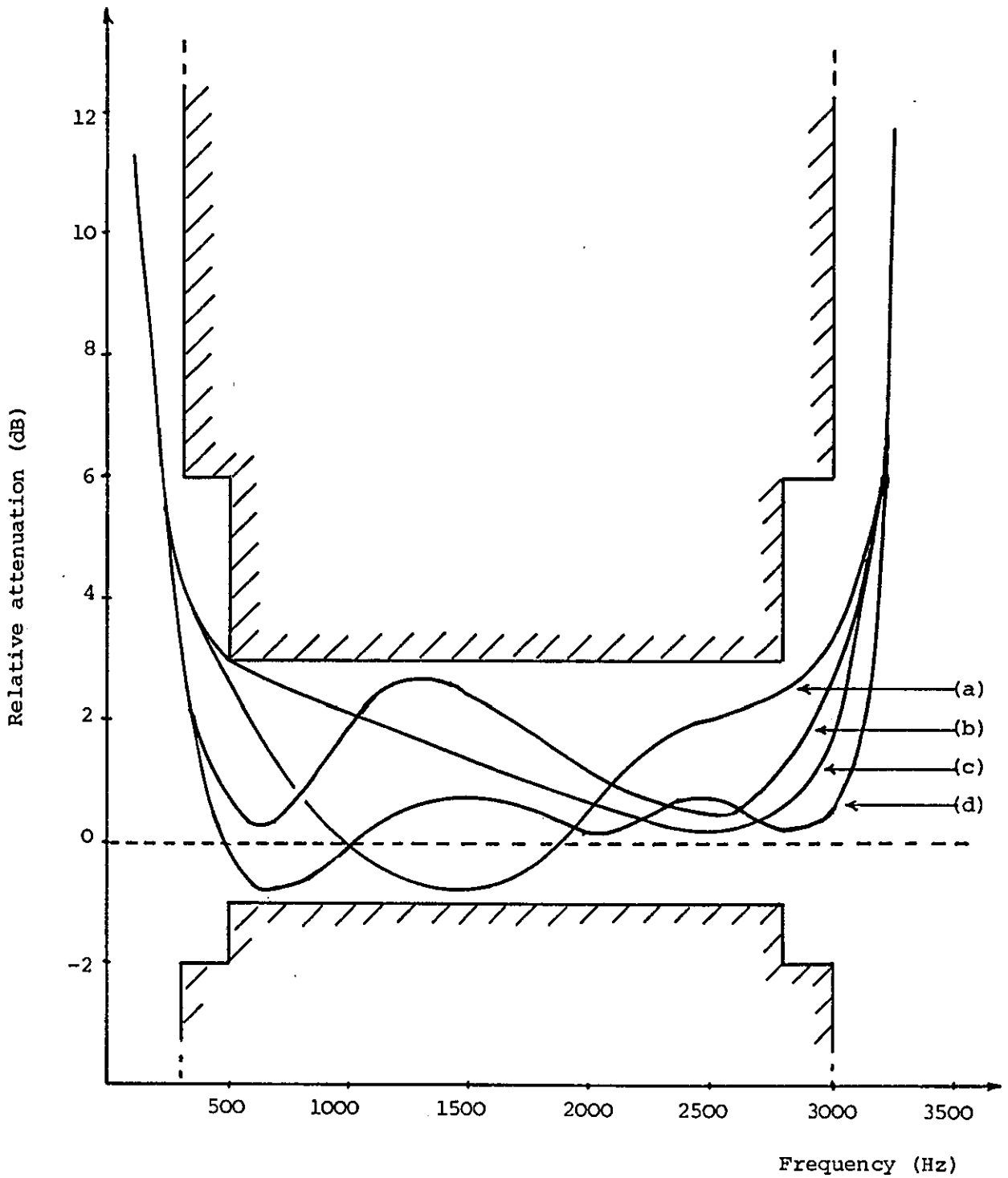



Figure 3.3.8: Typical worst-case attenuation characteristics, ref 4-7

Curves a)-c); regional variations

Curve d) ; Averaged "good" circuit

 ; schedule D limits

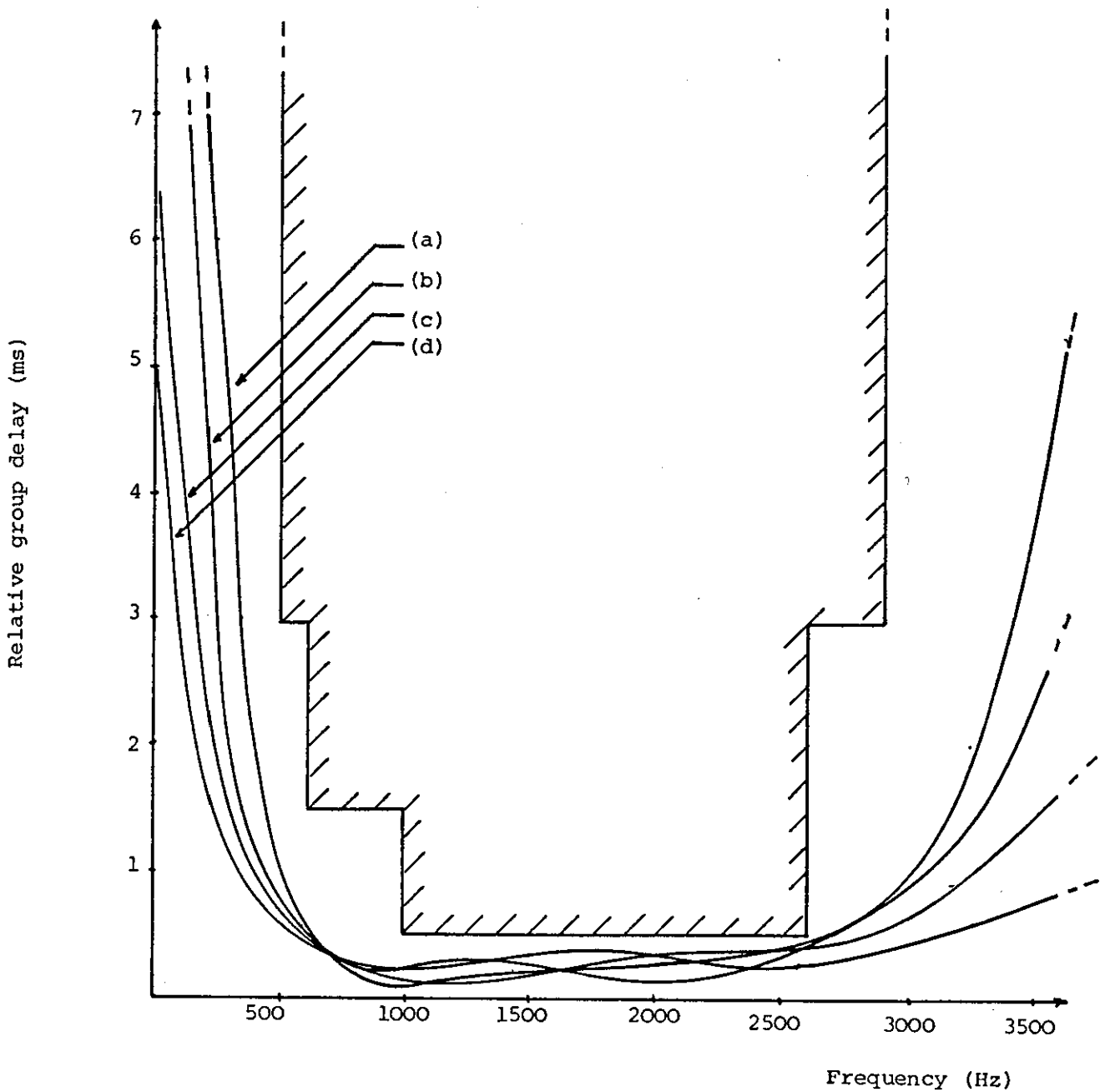



Figure 3.3.9: Typical worst-case group delay characteristics
ref 4-7

Curves a)-c); Regional variations

Curve d) ; Averaged "good" circuit

 ; Schedule D limits

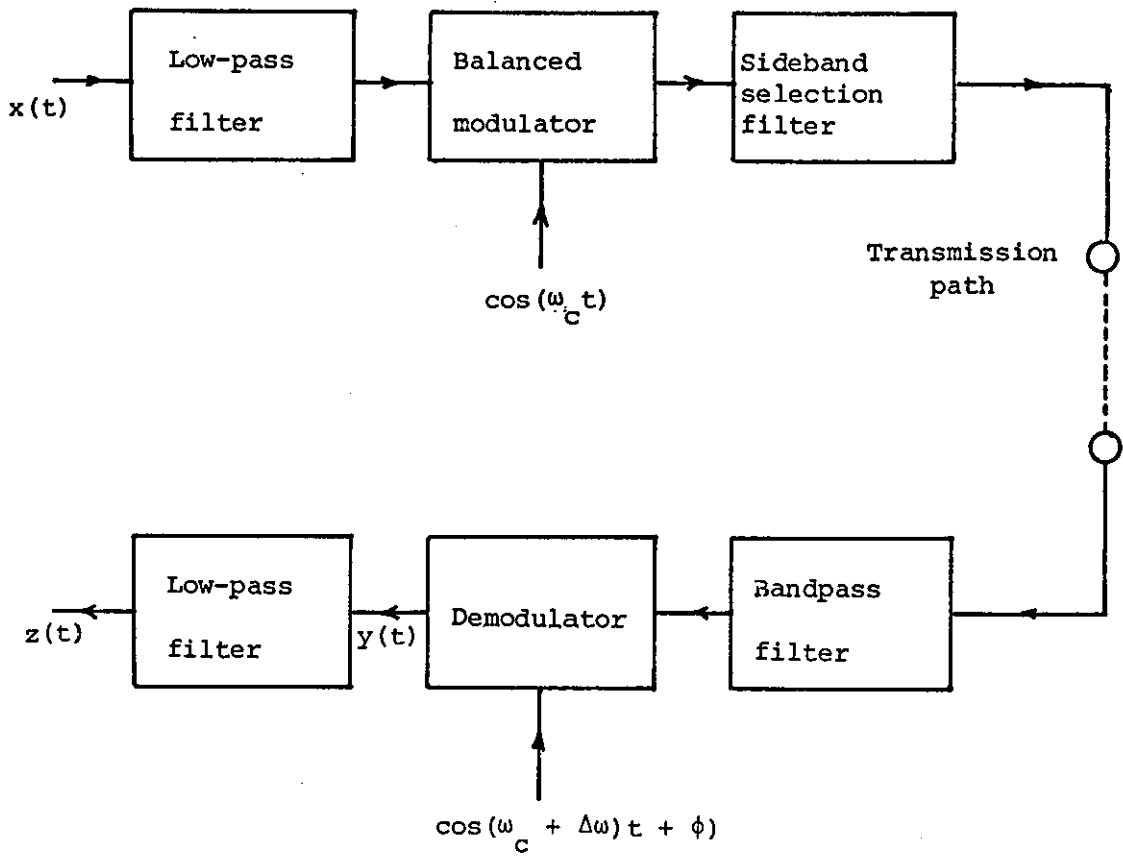


Figure 3.4.1: Modulated carrier system with frequency and phase errors between the local oscillators

If $x(t) = A\cos(\omega_m t)$

$$z(t) = \frac{K}{2}\cos((\omega_m - \Delta\omega)t - \phi), \text{ for U.S.B. system}$$

$$z(t) = \frac{K}{2}\cos((\omega_m + \Delta\omega)t + \phi), \text{ for L.S.B. system}$$

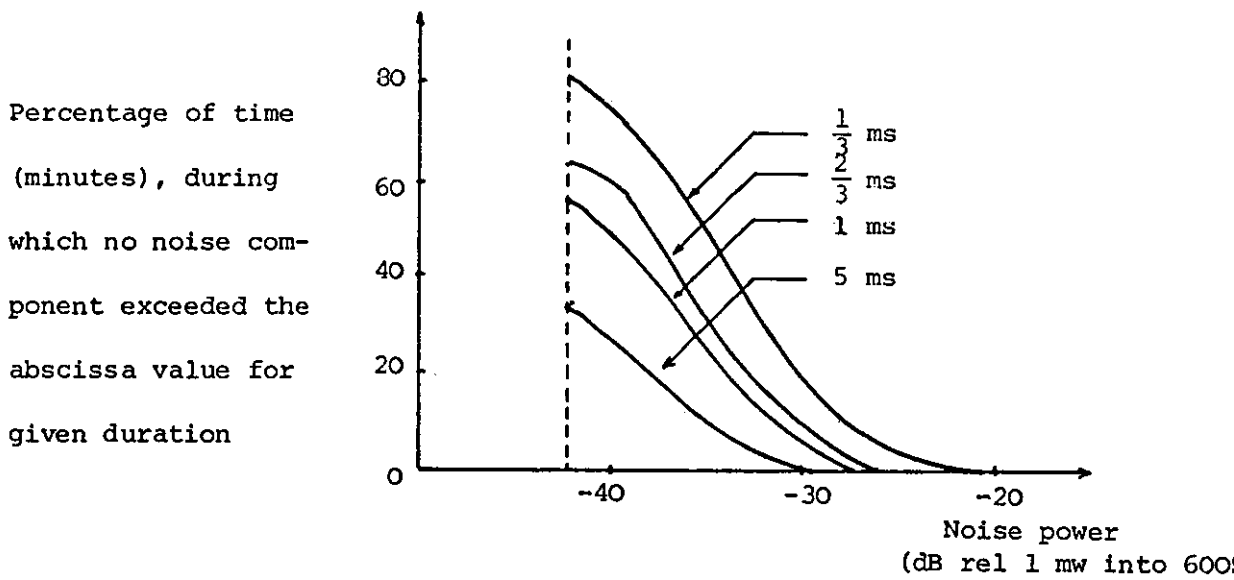


Figure 3.4.2: Measured impulsive noise power on connections
using electromechanical switches (ref. 4)

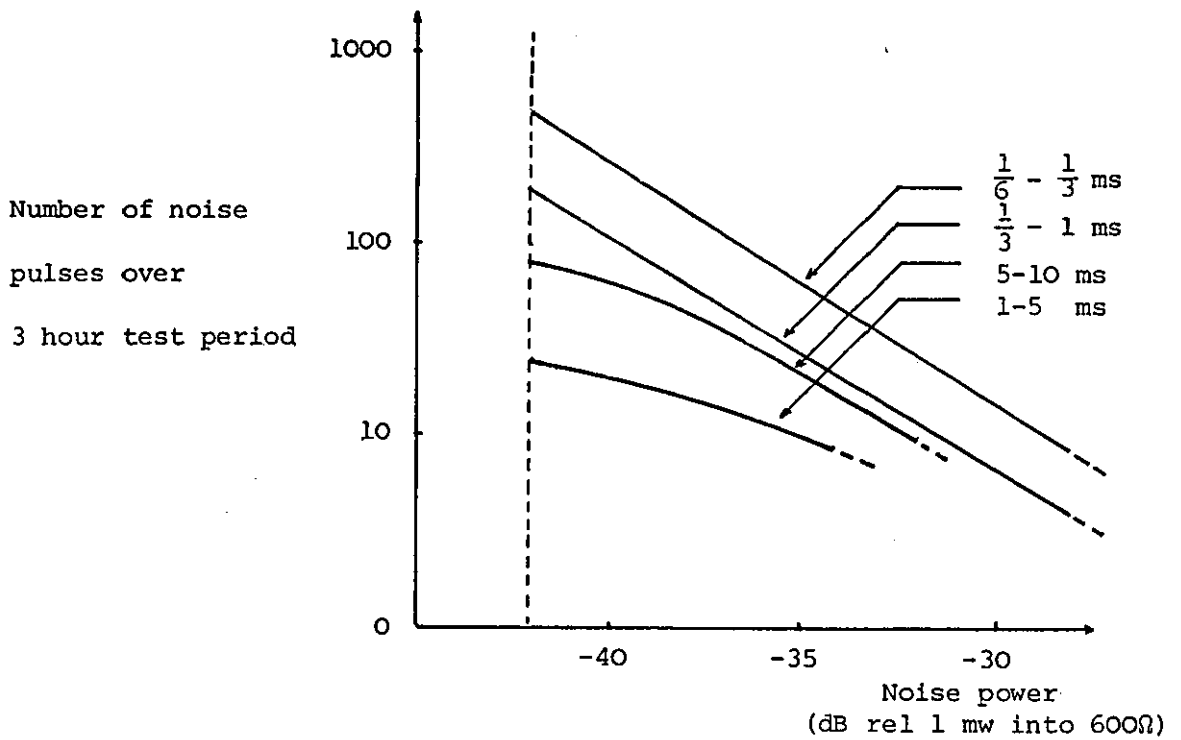


Figure 3.4.3: Total number of noise pulses during which given
noise power was exceeded for given duration (ref.4)

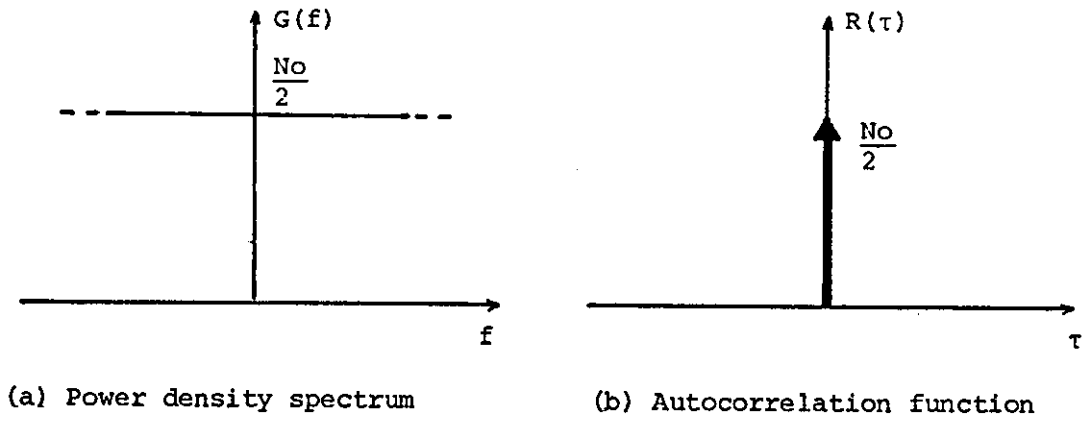


Figure 3.4.4: Properties of white noise

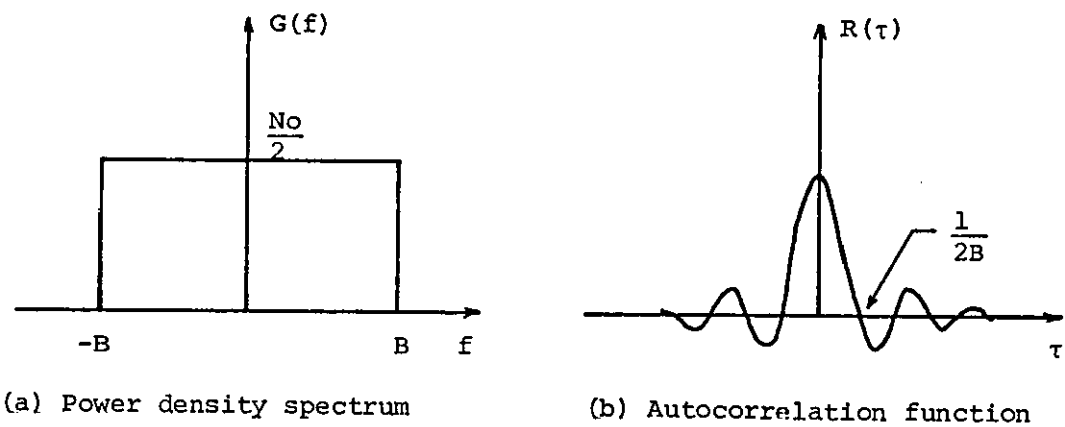
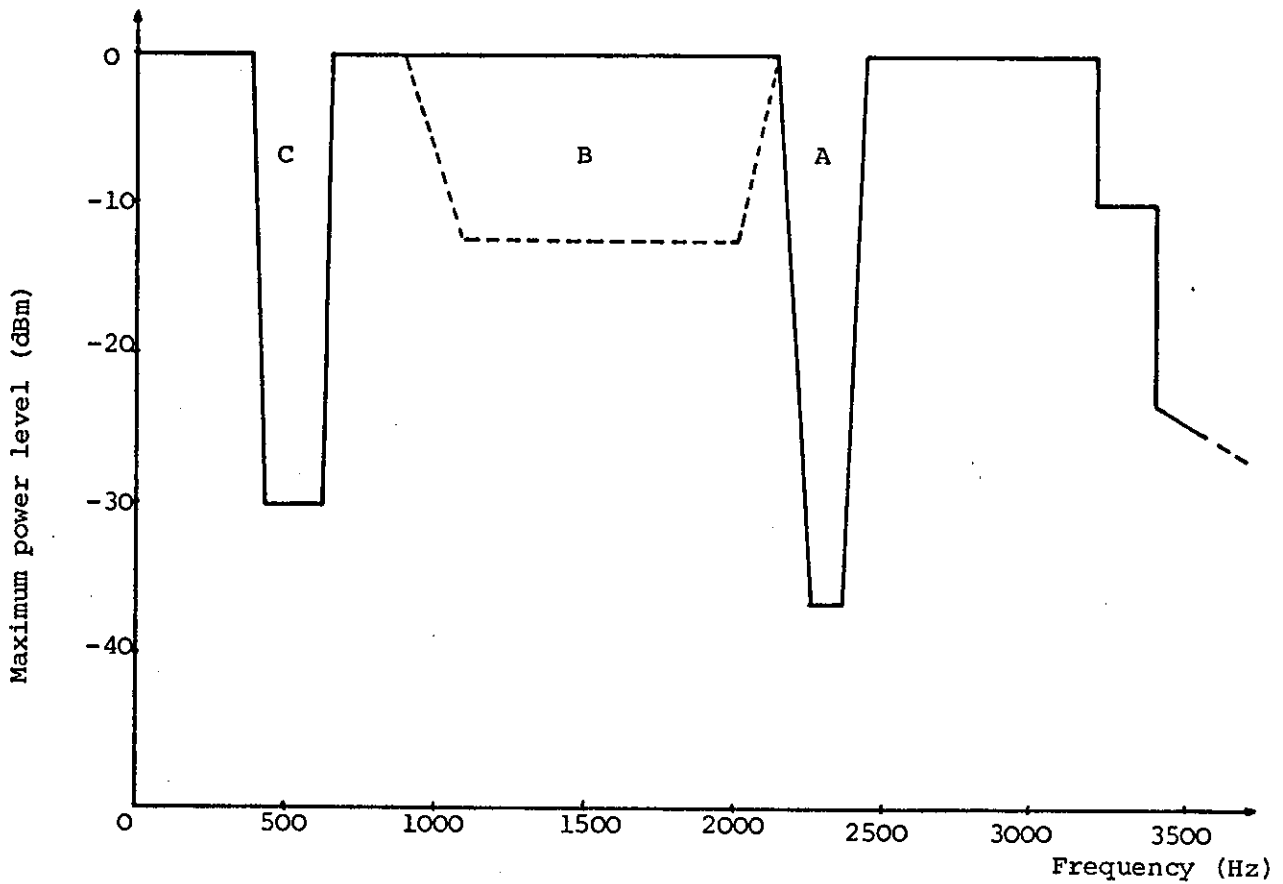
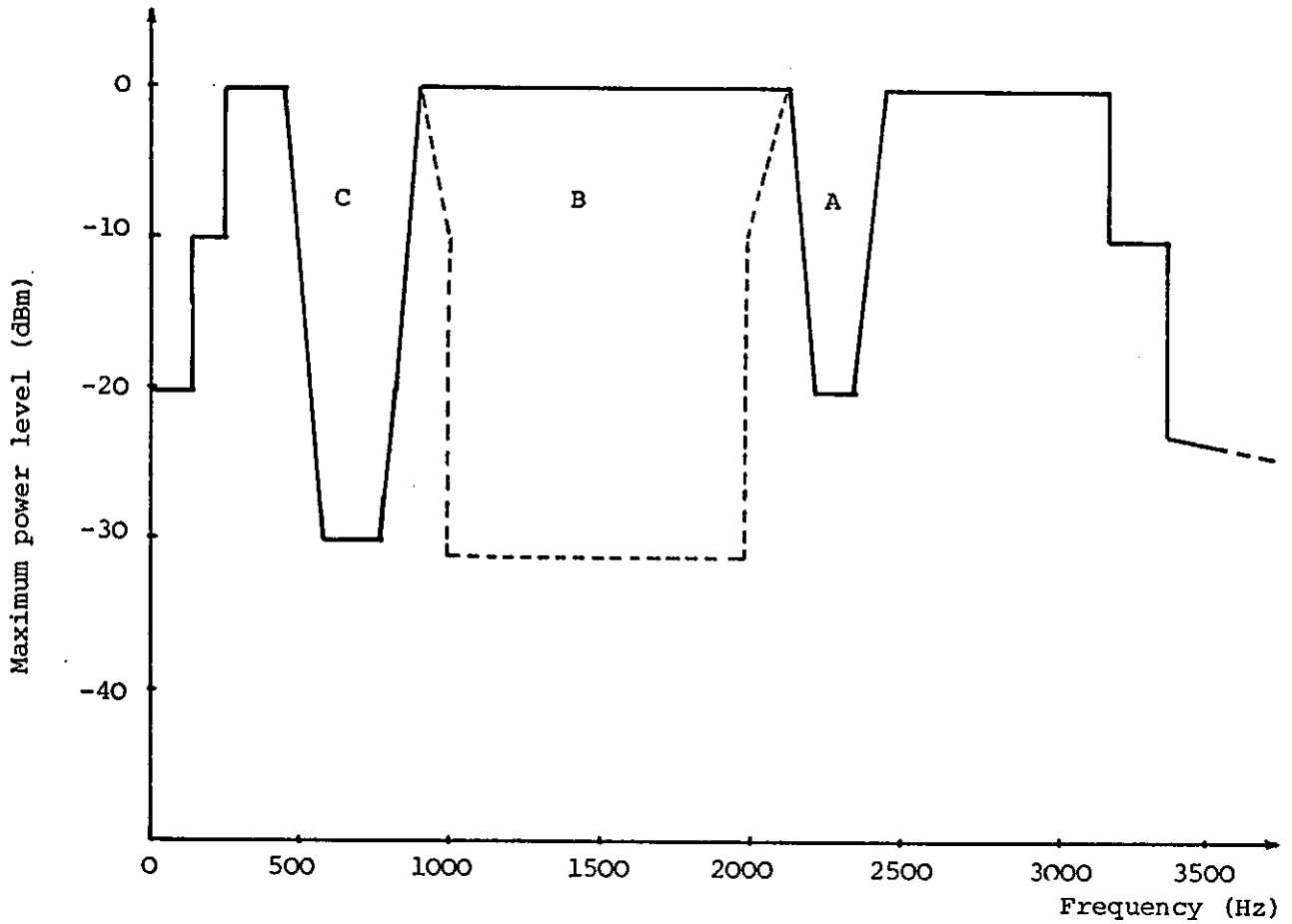


Figure 3.4.5: Properties of bandlimited white noise



- Notes: 1. Signals are permitted in area A only if accompanied by signals in area B at a power level not lower than 12 dB below the power level of the signal in area A.
2. Signals are permitted in area C provided there is no false operation of trunk signalling equipment.

Figure 3.4.6: Power level restrictions for apparatus connected to the PSTN



- Notes: 1. Area of sensitivity of in-band signalling equipment. Signal components upto 0 dBm may be permitted within area A if always accompanied by signals in area B.
2. Area of sensitivity of 500/20 Hz signalling receivers. Signals in area C may be permitted provided false operation does not occur.

Figure 3.4.7: Power level restrictions for apparatus connected to private circuits

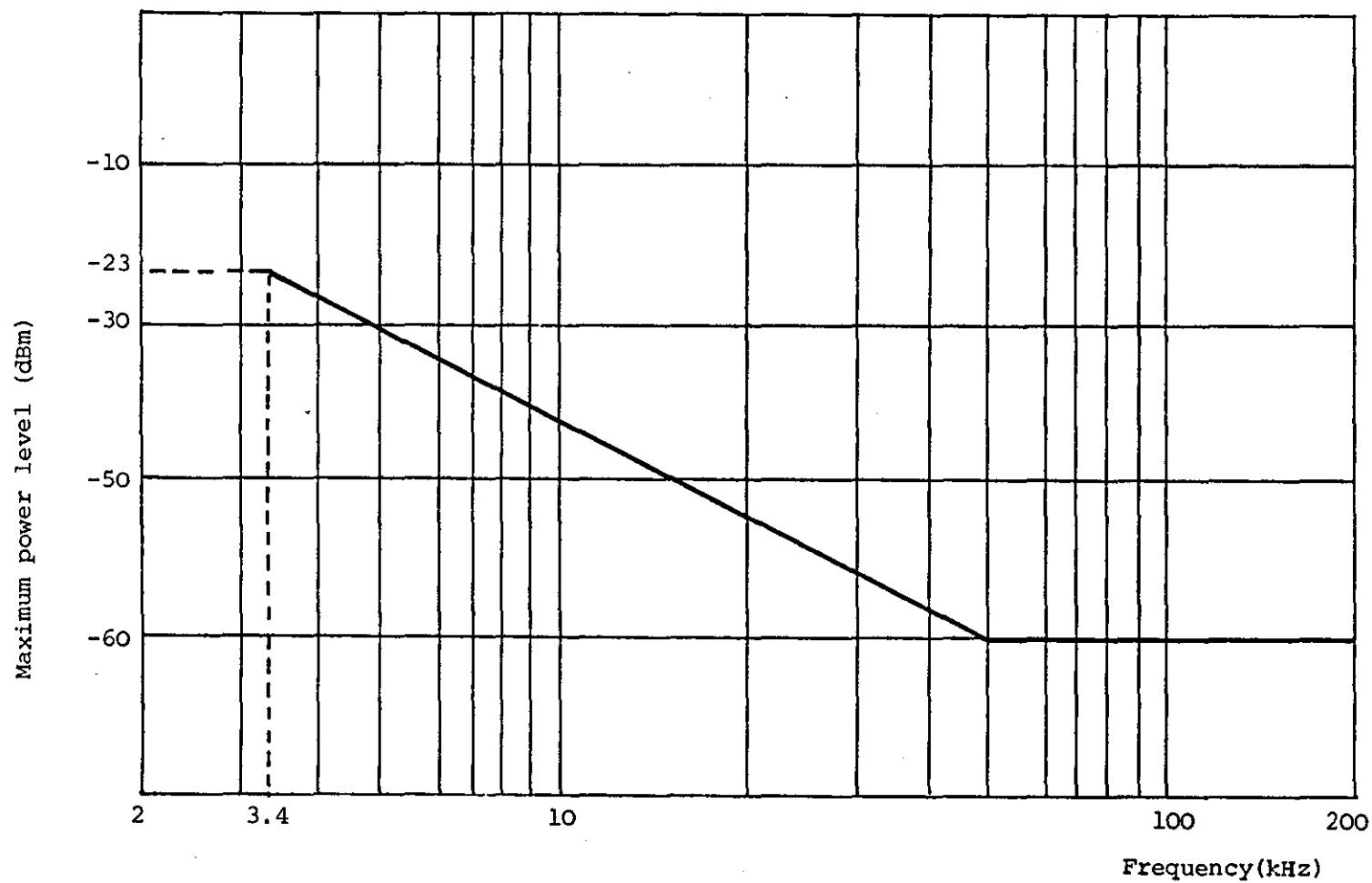


Figure 3.4.8: Power level restrictions for signal components above 3400 Hz for apparatus connected to voice-frequency circuits

4. Bandpass, Baseband and Modulation Systems

4.1 Introduction

In the brief overview of digital data transmission systems given in Chapter 2, it was assumed that a modulated synchronous serial transmission system operating over a bandpass channel could be represented by an equivalent baseband model with no loss in generality and no change in system performance. Before considering the various types of modulation processes frequently used for the transportation of high-speed digital data over bandlimited systems such as telephone lines, this chapter examines in more detail the relationship between bandpass and baseband systems/signals and illustrates in a more rigorous manner the representation of bandpass systems/signals using baseband equivalents. Both one and two dimensional baseband equivalent models are then presented and discussed in terms of their validity for the present study. After briefly reviewing general modulation methods, quadrature amplitude modulation (QAM) is introduced as an effective modulation method for the transmission of digital data at 19,200 bit/s and the system model is further refined to include the baseband representation of QAM. Finally, specific QAM signal structures (or constellations), carefully designed to exhibit combined differential and Gray coding properties are developed along with methods of implementing the encoder/decoder units. Two constellations, one being a 64-point structure, the other having 256-points, are then selected for further investigations. A mathematical analysis of the performance of these structures in the presence of additive white Gaussian noise is given in Appendix 5.

4.2 Bandpass Signals and Systems and the Equivalent Baseband Model

As already explained in Chapters 2 and 3, telephone channels, as well as many other communication channels, exhibit bandpass characteristics; the ideal telephone channel being bandlimited to the frequency band 300-3400 Hz.

The presence of frequency translation effects on the majority of bandpass channels has also been discussed in some detail in Chapter 3, where it was shown that the spectrum of the transmitted signal can be shifted by several Hz. Such a shift in the spectrum of a baseband signal will prevent the correct detection of the data signal unless the amount of spectral shift can be accurately determined at the receiver and suitably corrected. The majority of systems operating over bandpass channels use some form of modulation to shift the spectrum of the actual data signal (not necessarily the raw binary data signal) such that it lies within the passband of the channel. The received signal is then demodulated to recover the baseband signal, which is then detected. By tracking the carrier component present in the received signal or by recovering the carrier in a suppressed carrier system, the receiver may automatically correct for any internal frequency translations and constant or very slowly varying phase fluctuations⁽¹⁻³⁾.

A general model of the bandpass system therefore appears as in Figure 4.2.1. In this model, $s(t)$ is the input baseband signal; $q(t)$ is the modulated version of $s(t)$; $u(t)$ is a sample function of a white Gaussian random process; $p(t)$ is the received and filtered version of $q(t)$ which, in the absence of distortion, constant attenuation and constant delay is identical to $q(t)$; $w(t)$ is the noise component of the received and filtered signal which, due to the presence of the noise-rejection bandpass filter, may be correlated over time shifts $\tau > 0$; and

$r(t)$, $v(t)$ are the components of the baseband signal and noise, respectively, after demodulation and filtering. The two low-pass filters are included to limit the spectrum of the baseband signal before modulation and after demodulation. In practice, a further bandpass filter would be included between the modulator output and baseband channel input to ensure that the spectrum of the transmitted signal conforms with the channel's specification. Assuming that the modulation/demodulation process is linear, the output baseband signal, $r(t)$, will be identical in form to the input baseband signal, $s(t)$, in the absence of any signal distortion. Again assuming linear modulation and demodulation, it is intuitively clear that the distortion will have a linearly related counterpart in the received baseband signal, $r(t)$. It should also be clear that the presence of additive noise does not affect this intuitive hypothesis since, because of the system's linearity, the principle of superposition will apply. With the above in mind, it is logical to suppose that the complete bandpass system given in Figure 4.2.1 has an equivalent model which is concerned only with baseband signals and the equivalent baseband representation of the modulation/demodulation process, linear filtering and the linear bandpass transmission channel, as shown in Figure 4.2.2. Appendix 1 presents a more rigorous proof of the validity of the equivalent baseband model. For reasons of mathematical convenience, the majority of this study deals with baseband signals and the equivalent baseband model.

The equivalent models of Figures 4.2.1 and 4.2.2 are assumed to be one-dimensional in that the input data symbols, $s(t)$, is represented by real-valued quantities. For particular modulation formats, however, the complete system may conveniently be modelled in two dimensions, where

the input data symbols are considered complex quantities such that $s(t) = s_a(t) + js_b(t)$. Figure 4.2.3 shows the bandpass model of such a two-dimensional system where $s_a(t)$ and $s_b(t)$ represent two parallel data signals. The data signals are then modulated by carrier waveforms with identical frequencies, f_c Hz but in phase-quadrature to each other, before being applied to the bandpass transmission path as the sum of the two modulated signals. As before, the noise introduced by the system is represented by an additive white Gaussian noise source with a zero-mean value and two-sided power spectral density of $\frac{1}{2} N_0$ W/Hz. After suitable filtering, the received signal is then demodulated by the two quadrature carrier waveforms and filtered by the two low-pass filters to give the two baseband signals $r_a(t)$ and $r_b(t)$. These signals are then operated upon to give the detected symbol values. Clearly, the receiver of such a system must be capable of separating the two symbol waveforms to retain the identity of each of the two parallel data channels. This ability of the receiver is discussed in more detail in Section 4.3 where the two-dimensional model is adapted to the particular case of Quadrature Amplitude Modulation, (QAM). However, it is important to recall from Chapter 2 that a practical transmission path introducing signal distortion may not only cause intersymbol interference but also interchannel interference whereby the two parallel channels mutually interfere with each other.

The bandpass model presented above may be reduced to an equivalent baseband model by applying the results of Appendix 1, as derived for the one-dimensional model and by considering the input (and therefore the output) baseband signals as the real and imaginary components of a complex baseband signal. The real part of the complex signal represents $s_a(t)$ and is therefore associated with the "cosine" modulator/demodulator and the

imaginary part represents $s_b(t)$ and is associated with the "sine" modulator/demodulator. The complex baseband signal is then applied to the equivalent baseband transmission path which is assumed to carry complex signals. A more detailed description of the two models for the particular case of QAM is given in the following section. Appendix 2 considers the relative values of signal and noise levels required to maintain a given signal-to-noise ratio across the two models. It is shown in the following section that the baseband model given in Figure 4.2.2 also applies to the case of two-dimensional systems where the $s(t)$ and $r(t)$ are complex-valued and most importantly, where the 2-sided power spectral density of the additive Gaussian noise now has the value N_0 W/Hz, twice that of the noise added to the original bandpass system⁽³⁻⁵⁾.

4.3 Selection of Modulation Method

The previous section and Appendices 1 and 2 show that a bandpass communication channel such as a telephone circuit can be represented by an equivalent baseband model which aids the mathematical description and simulation of such systems. However, the designer of practical modems must bear in mind that in real systems, a modulated baseband signal will be carried over a bandpass channel which may introduce some or all of the impairments discussed in Chapter 3. It is imperative therefore for the designer to select the most appropriate modulation method for the given operational requirements and conditions. It should also be recalled from Chapter 2 that in order to transmit digital data at very high speeds over bandlimited channels, some form of multilevel signalling must be used.

Yet another important criterion for a system designer is overall complexity and indeed the very feasibility of modem implementation. It is interesting to note that whilst research into methods of high-speed data transmission has been intense over a period of 25 years or more, it is only in the past 5 years that suitable detection methods have been studied, (see Chapter 6). Moreover, it is only the past 3-4 years that have witnessed the staggering advances in microelectronic technology necessary to physically realise these designs, (see Chapter 9).

Based on the results of a wealth of literature on the subjects of modulation methods, their performance over the type of channel considered here and the implementation of resulting systems, the most cost-effective modulation format for the transmission of 19,200 bit/s appears to be Quadrature Amplitude Modulation with carefully designed signal constellations operating with adaptive detection processes^(1-3, 5-33). A QAM signal comprises the sum of two parallel, suppressed carrier, double sideband amplitude modulated signals utilising carriers of the same frequency but in phase quadrature to each other, the two parallel signals being in element synchronism. Since the input binary data is encoded into two parallel data signals, the signal element rate (symbol rate) will be half that of a corresponding single or vestigial sideband system having the same information rate⁽³⁻⁶⁾. The price paid for this, however, is an apparent increase in complexity since a QAM system requires two modulator/demodulator units and associated filters. In practice, this does not increase system complexity to any great extent due to the use of digital modulation techniques and the fact that the implementation of suitable adaptive detectors require the use of extremely sophisticated and powerful signal

processing devices (Chapter 9). The advantages of QAM systems over other popular modulation formats for the transmission of high-speed data over telephone lines, in particular vestigial sideband (VSB), can be summarised as follows^(3, 5, 6);

a) Bandwidth efficiency and simple filter design.

Since QAM is a double-sideband process, the carrier frequency can be positioned at the centre of the available bandwidth. Consequently, the baseband signal can be permitted to have significant energy at low frequencies without the need for the strict shaping filters required by both VSB and independent sideband (ISB) systems.

b) As explained later, an adaptive detection scheme must be used to enable the system to operate over telephone circuits with different characteristics, which introduce the various types of signal distortion and impairments. Provided that the receiver can track any phase errors between the reference carriers used by the demodulators and the suppressed carriers of the QAM signal, no particular phase relationship need be maintained between these carriers. The two reference carriers must, of course, remain at 90° to each other. Consequently, it is not necessary to transmit a pilot carrier along with the QAM signal, which avoids the complications of pilot carrier isolation at the receiver.

c) Good immunity to additive noise.

The combination of less severe filtering and the elimination of a pilot carrier increases the tolerance of QAM systems to additive noise when compared with VSB systems.

The bandpass model for a QAM system is shown in Figure 4.3.1. The binary input data to be transmitted is initially fed to an encoder unit which structures the incoming bits of information into blocks of suitable

size. The actual length of each block is dependent on the required information rate and the system's line transmission rate. The encoder output comprises two parallel baseband signals, $s_a(t)$ and $s_b(t)$, which contain the statistically independent data symbols. The symbols are in element synchronism and both have signal alphabets of size \sqrt{m} . The combination of any two data symbols may therefore have one of m possible values. The two baseband signals, $s_a(t)$ and $s_b(t)$, are then fed to two low-pass filters with real-valued impulse responses given by $g(t)$. The shaped signals are then modulated by the two quadrature carriers at frequency f_c Hz, which positions the modulated and combined (QAM) spectrum into the bandwidth of the transmission channel with no loss of signal energy. The resulting QAM signal can therefore be written as;

$$\begin{aligned} \text{QAM}(t) = & \sqrt{2} \int_{-\infty}^{\infty} s_a(\tau) g(t-\tau) d\tau \cos 2\pi f_c t - \\ & - \sqrt{2} \int_{-\infty}^{\infty} s_b(\tau) g(t-\tau) d\tau \sin 2\pi f_c t \end{aligned} \quad (4.3.1)$$

Before application to the transmission channel itself, the resultant QAM signal is first bandpass filtered to ensure that all signal components lie within the limits of the channel's specifications, as described in Chapter 3. In practice, the transfer function of this filter will be flat across the range of frequencies defined by the low-pass shaping filters and so it will have only a small effect on the transmitted and received QAM spectrum.

Additive Gaussian noise with a zero mean and two-sided power spectral density of $\frac{1}{2} N_0$ W/Hz is then added to the output of the transmission path, which is assumed to represent the actual noise introduced over the path.

The combined signal-plus-noise waveform is then filtered to remove as much out of band noise as possible without however, introducing significant distortion to the signal itself. In practice, the filter will be bandpass in shape such that it rejects high frequency noise components and also removes frequency components below 200-300 Hz to avoid the intrusion of power frequencies and immediate harmonics. Following filtering, the received signal is coherently demodulated by two carriers in phase quadrature with the same frequency as the modulator carriers, f_c . The two low-pass filters with impulse responses given by $p(t)$ remove the high frequency components produced by the demodulation process. The two outputs from the filters, $r_a(t)$ and $r_b(t)$, are shaped, distorted and noisy equivalents of $s_a(t)$ and $s_b(t)$, respectively. The received baseband signals are then simultaneously sampled once per signal element to yield the sampled signals $r_a(iT)$ and $r_b(iT)$ which are operated upon by the detector to give the detected symbols $s'_a(iT)$, $s'_b(iT)$ corresponding to $s_a(iT)$, $s_b(iT)$. Finally, the decoder restructures the two parallel detected symbol streams into binary data at the required information rate.

Comparison between Figures 4.2.3 and 4.3.1 shows that the bandpass model of the QAM system is identical in form to the model of the general two-dimension communication system when the modulation method and associated post-modulation filters use double sideband, suppressed carrier amplitude modulation. This suggests that the QAM system can be modelled by an equivalent baseband model of a generalised two-dimensional model, as shown in Figure 4.2.2. This equivalence is now proved for a QAM system assumed to be operating at the Nyquist rate of $\frac{1}{T}$ bauds^(1-4,6,7).

Apart from the pre-modulation shaping filters, all other lowpass and bandpass filters are, for purposes of illustration, assumed to exhibit

ideal transfer characteristics and are therefore clearly unrealisable; Chapter 5 studies the requirement of all equipment filters in detail and presents designs for practical analogue and digital filters.

The signals $s_a(t)$ and $s_b(t)$ in Figure 4.3.1 are assumed to be rectangular multilevel polar baseband signals in element synchronism.

The analysis of the QAM system can be simplified mathematically by;

- a) Considering the baseband inputs to be represented by a stream of impulses, where the impulses occur at times $t = iT$ and have strengths $s_a(iT)$ and $s_b(iT)$, respectively. If the appropriate changes are made to the low-pass pre-modulation shaping filters, the system as a whole will not be affected.
- b) Equation 4.3.1 may be written more succinctly by expressing the two baseband signals, at time $t = iT$, as the real and imaginary components of a complex baseband signal;

$$s(iT) = s_a(iT) + js_b(iT) \quad (4.3.2)$$

No change in the system will be required due to the quadrature relationships between both the symbol pairs and the carriers used in the modulation/demodulation process.

Due to the assumed impulsive nature of the real and imaginary data symbols, the transfer functions of the pre-modulation filters, $G(f)$, now define the spectra of the signal elements prior to modulation and so $G(f)$ must be bandlimited to $\frac{1}{2T}$ Hz. Figure 4.3.2 shows a suitable amplitude response for $G(f)$ and Figure 4.3.3 shows the two-sided amplitude spectrum of the QAM signal, where it is assumed that $f_c < \frac{1}{2T}$ to ensure a fully bandlimited QAM spectrum. In practice, $f_c - \frac{1}{2T}$, the lowest signal frequency to be passed through the channel, will be around 200-300 Hz to ensure that the receiver bandpass filter does not add significant distortion to the low frequency components of the received QAM signal.

For the case of 3200 baud transmission at the Nyquist rate, $\frac{1}{2T} = 1600$ Hz and so a carrier frequency of 1800 Hz would give $f_c - \frac{1}{2T} = 200$ Hz. The receiver filter is ideally a "brick-wall" filter with a passband between $\left[f_c - \frac{1}{2T}\right]$ Hz and $\left[f_c + \frac{1}{2T}\right]$ Hz, as shown in Figure 4.3.4. The post-demodulation filters ideally pass the baseband signal spectra without, however, passing the higher-order demodulation products. The ideal shape of these filters is therefore "brickwall" with cut-off frequencies at $\pm \frac{1}{2T}$ Hz, as illustrated in Figure 4.3.5.

From equation (4.3.1) and using the simplifications introduced earlier,

$$\begin{aligned} s(t) &= s_a(t) + j s_b(t) \\ &= \sum_i \left[s_{ai} \delta(t-iT) + j s_{bi} \delta(t-iT) \right] \end{aligned} \quad (4.3.3)$$

where $\delta(t)$ is the Dirac impulse function and where s_{ai} and s_{bi} can each have one of \sqrt{m} possible values. Consequently,

$$QAM(t) = \sqrt{2} \sum_i (s_{ai} g(t-iT) \cos 2\pi f_c t - s_{bi} g(t-iT) \sin 2\pi f_c t) \quad (4.3.4)$$

Alternatively, using complex exponential notation,

$$QAM(t) = \sqrt{2} \operatorname{Re} \left\{ \sum_i s_i g(t-iT) e^{j2\pi f_c t} \right\} \quad (4.3.5)$$

where $s_i = s_{ai} + j s_{bi}$, the complex symbol at time $t = iT$,

$\operatorname{Re}(\cdot)$ denotes the real part of (\cdot)

and $j = \sqrt{-1}$

With reference to Figure 4.3.1, where;

$g(t)$ = Impulse response of the pre-modulation filters,

$c(t)$ = Impulse response of the receiver input filters,

$p(t)$ = Impulse response of the post-demodulation filters,

$h(t)$ = Impulse response of the bandpass channel,

$q(t)$ = Impulse response of the channel input filter,

the input signal to the demodulators is

$$d(t) = QAM(t) * q(t) * h(t) * c(t) + n(t) * c(t) \quad (4.3.6)$$

where $*$ denotes the convolution operation,

$n(t)$ = A sample function of a white Gaussian random variable with a zero mean value and two-sided power spectral density of $\frac{1}{2}$ No W/Hz

The output from the two post-demodulation filters is therefore,

$$r_a(t) = \sqrt{2} (d(t) \cos 2\pi f_c t) * p(t) \quad (4.3.7)$$

$$r_b(t) = -\sqrt{2} (d(t) \sin 2\pi f_c t) * p(t) \quad (4.3.8)$$

Furthermore, if

$$r(t) = r_a(t) + jr_b(t) \quad (4.3.9)$$

then,

$$r(t) = \sqrt{2} (d(t) e^{-j2\pi f_c t}) * p(t) \quad (4.3.10)$$

To allow for a relative phase error between the reference (demodulator) carriers and the actual QAM suppressed carriers, Figure 4.3.1 shows the reference carriers with a phase shift of ϕ rads. Consequently, equation (4.3.10) becomes,

$$r(t) = \sqrt{2} (d(t) e^{-j(2\pi f_c t + \phi)}) * p(t) \quad (4.3.11)$$

Recognising that equation (4.3.5) may be re-written as;

$$QAM(t) = \frac{\sqrt{2}}{2} \left[\sum_i (s_i e^{j2\pi f_c t} + s_i^* e^{-j2\pi f_c t}) g(t - iT) \right] \quad (4.3.12)$$

where s_i^* is the complex conjugate of s_i ,

then from equations (4.3.6), (4.3.11) and (4.3.12),

$$r(t) = \sqrt{2} (QAM(t) * q(t) * h(t) * c(t) + n(t) * c(t)) e^{-j(2\pi f_c t + \phi)} * p(t) \quad (4.3.13)$$

$$r(t) = \sqrt{2} \left[\left(\frac{1}{\sqrt{2}} \left[\sum_i s_i e^{j2\pi f_c t} + s_i^* e^{-j2\pi f_c t} \right] g(t-iT) \right) * q(t) * h(t) \right. \\ \left. * c(t) + n(t) * c(t) \right] e^{-j(2\pi f_c t + \phi)} * p(t) \quad (4.3.14)$$

hence,

$$r(t) = \sum_i s_i \left[g(t-iT) * \left[\left(q(t) * h(t) * c(t) \right) e^{-j\phi} \right] \right] * p(t) \\ + \sum_i s_i^* \left[e^{-j4\pi f_c t} g(t-iT) * \left[\left(q(t) * h(t) * c(t) \right) e^{-j2\pi f_c t} \right] \right] e^{-j\phi} * p(t), \\ + \sqrt{2} \left[\left(n(t) * c(t) \right) e^{-j(2\pi f_c t + \phi)} \right] * p(t) \quad (4.3.15)$$

The expression within the second summation contains the term $g(t-iT)e^{-j4\pi f_c t}$, which represents the demodulation product corresponding to a signal at twice the frequencies defined by $G(f)$. The convolution of this term with $p(t)$ will reduce its contribution to zero by virtue of the bandwidth of the post-demodulation filters shown in Figure 4.3.5.

Equation (4.3.15) can now be expressed as a single signal term,

$$\sum_i s_i y(t-iT) = \sum_i s_i \left[\left(g(t-iT) * \left[\left(q(t) * h(t) * c(t) \right) e^{-j2\pi f_c t} \right] \right) e^{-j\phi} \right] * p(t), \quad (4.3.16)$$

and an additive noise term,

$$u(t) = \sqrt{2} \left[\left(n(t) * c(t) \right) e^{-j(2\pi f_c t + \phi)} \right] * p(t) \quad (4.3.17)$$

and so $r(t)$ can be written as;

$$r(t) = \sum_i s_i y(t-iT) + u(t) \quad (4.3.18)$$

where $y(t)$ is the overall impulse response of the system and $u(t)$ is the (complex) noise component in the received, demodulated and filtered signal.

Using the results of Appendix 1, it can be seen that $y(t)$ represents the equivalent baseband impulse response of the system since from equation (4.3.16), the bandpass responses $h(t)$, $c(t)$ and $q(t)$ have been translated to baseband by the multiplying term $e^{-j2\pi f_c t}$. Similarly, the expression for $u(t)$ in equation (4.3.17) can be interpreted as a baseband version of the noise at the output of the post-demodulation filters. The transfer functions associated with $n(t)$ and $c(t)$ are broadband and bandpass respectively and are translated down in frequency by f_c Hz.

It is interesting to note from equation (4.3.16) that the effect of any phase difference between the signal and reference carriers is to alter the allocation of the QAM signal to the two baseband signals, $r_a(t)$ and $r_b(t)$. This can be illustrated as follows. If the received and demodulated signals in the absence of phase error are represented as;

$$r(t) = r_a(t) + jr_b(t) \quad (4.3.19)$$

then with a phase error of ϕ rads;

$$r(t) = [r_a(t) + jr_b(t)]e^{-j\phi} \quad (4.3.20)$$

$$r(t) = |r|e^{j\theta_r}e^{-j\phi} \quad (4.3.21)$$

$$r(t) = |r|e^{j(\theta_r - \phi)} \quad (4.3.22)$$

where $|r| = |r_a(t) + jr_b(t)|$

Equation (4.3.22) and Figure 4.3.6 show that the allocation of the real and imaginary baseband signals will have changed due to the phase rotation by $-\phi$ rads, but with no loss in baseband signal power. This, in fact, illustrates another advantage of QAM since a phase error in a

VSB or SSB signal would introduce severe delay distortion into the signal; yet another reason for having a pilot carrier to phase lock the receiver's reference carrier. The phase error, ϕ , in the above analysis has so far been considered constant or varying only very slowly with time. However, if a frequency offset exists between the signal and reference carriers, the received baseband signal vector will rotate around a circle of diameter $|r|$ in equation (4.3.22), where the rotation will be determined by the phasor $e^{-j2\pi\Delta f t}$; Δf represents the amount of frequency offset in Hertz. In practical modems, the receiver will track these offsets by using some form of phase-locked loop technique⁽¹⁻⁷⁾. From equation (4.3.16), the phase error and any offset present may alternatively be considered to vary the overall baseband impulse response of the system; as $2\pi\Delta f t$ changes (and if ϕ changes) with time, the impulse response will also change since the contribution to the real and imaginary parts of $y(t)$ will vary. Clearly, for a data detector to take full account of the distortion present in the transmission system, it must have prior knowledge of $y(t)$ and therefore of ϕ and Δf . As mentioned above, this is achieved by adaptive estimation processes which are discussed in more detail in Chapters 7 and 8.

Based upon equations (4.3.16), (4.3.17) and (4.3.18), the baseband model of the QAM system is shown in Figure 4.3.7. In this model, $y(t)$ is generally complex valued and represents the impulse response of the overall linear system comprising the baseband pre-modulation shaping filter pair, the characteristics of the transmission channel referred to baseband, the response of the receiver filter referred to baseband, the post-demodulation baseband filter pair and the modulation/demodulation

process itself. The input data is a stream of complex valued symbols, $\{s_i\}$ where each s_i is given by equation (4.3.2) and may have one of m possible values. Gaussian noise with a zero mean and two-sided power spectral density of N_0 W/Hz is added to the system before the receiver filter in Figure 4.3.7, this filter representing the combined processes of the baseband referred receiver input filter and the post-demodulation low-pass filters. This can be seen more easily by expanding equation (4.3.17) as;

$$u(t) = \sqrt{2} \left[\left[n(t) e^{-j2\pi f_c t} \right] * \left[c(t) e^{-j2\pi f_c t} \right] e^{-j\theta} \right] * p(t) \quad (4.3.23)$$

From Figure 4.3.1, $n(t)$ has a two-sided power spectral density of $\frac{1}{2} N_0$ W/Hz and so the term $n(t) e^{-j2\pi f_c t}$ will be simply a spectrally shifted version of $n(t)$ with the same power spectral density. Since equation (4.3.23) states that the baseband equivalent noise, $u(t)$, is given by the passage of $n(t)$ through the baseband equivalent of the receiver input filter and the low-pass filters, then using the results of Appendix 1 and Section 3.4.3, the power spectral density of $u(t)$ is,

$$|U(f)|^2 = N_0 |C(f+f_c)|^2 \cdot |P(f)|^2 \quad (4.3.24)$$

and the auto-correlation function of $u(t)$ is,

$$R_u(\tau) = N_0 \int_{-\infty}^{\infty} |C(f+f_c)|^2 \cdot |P(f)|^2 e^{j2\pi f \tau} df \quad (4.3.25)$$

so sample values of the (coloured) Gaussian noise used in the baseband model of Figure 4.3.7 may be correlated to some degree, depending upon the transfer function of the combined receiver filter. When using computer simulation methods to assess the performance of digital data systems, Gaussian noise generators with zero mean values, fixed

variances and flat power spectra over a very wide range of frequencies are usually available as add-on subroutines. Under the conditions discussed in Appendix 3, the power spectral density of such noise generators can be made equal to the variance of the distribution, which greatly simplifies the analysis of system signal-to-noise ratio. Appendix 3 also defines the method of calculation for signal-to-noise ratio as used throughout the remainder of this investigation. When using digital computers to simulate the system, it is more convenient to calculate $y(t)$, the overall impulse response, before running the actual simulation program. This method saves computer run time by eliminating the multiplications required in the convolution of the various baseband stages. Noise is then added at the output of the baseband channel prior to detection. Because of the possibility of noise correlation, as given by equation (4.3.25), it is most important to pass the noise from the white noise generator subroutine through a filter whose transfer function is identical to that of the combined receiver filter in the baseband model. Figure 4.3.8 shows the modified model for use in computer simulation studies. An extremely rigorous description of noise correlation and methods of generating correlated Gaussian noise for use in computer simulations can be found in reference 31.

The received baseband signal, $r(t) = r_a(t) + jr_b(t)$ in Figures 4.3.1 and 4.3.3 is then sampled once per signal element at times $t=iT$ to give the complex sequence $\{r_i\}$, where each $r_i = r_{ai} + jr_{bi}$. The detector then operates on r_i to give the detected complex symbol value, $s'_i = s'_{a,i} + js'_{b,i}$. Clearly, the detector must be capable of operating on complex-valued signals. From equation (4.3.18), the received sample

at time $t = kT$ will be;

$$r_k = \sum_i s_i y((k-i)T) + u(kT) \quad (4.3.26)$$

If $y(t)$ is now sampled at times $t=kT$ to give the sampled impulse response Y , where Y is the (time invariant) row vector given by;

$$Y = [y_0 \ y_1 \ y_2 \ \dots \ y_g]$$

and where it is assumed that $y_j = 0$ for $j < 0$ and $j > g$ and where the y_j are in general complex, then equation (4.3.26) may be written as;

$$r_i = \sum_{h=0}^g s_{i-h} y_h + u_i \quad (4.3.27)$$

where $r_i = r(iT)$, $u_i = u(iT)$ and r_i , s_i , y_i and u_i are all complex valued. Equation (4.3.27) shows the discrete time form of the received samples as used in the model of Figure 4.3.8.

As mentioned in Section 2.2, the sampled impulse response of an ideal channel is given by;

$$Y = [1 \ 0 \ 0 \ \dots \ 0] \quad (4.3.28)$$

From equation (4.3.27), the received samples from the QAM system will therefore be given by;

$$r_i = s_i + u_i \quad (4.3.29)$$

$$r_{ai} + jr_{bi} = (s_{ai} + js_{bi}) + (u_{ai} + ju_{bi}) \quad (4.3.30)$$

where u_{ai} , u_{bi} are the noise samples added to the real and imaginary channels respectively.

Hence;

$$\begin{aligned} r_{ai} &= s_{ai} + u_{ai} \\ r_{bi} &= s_{bi} + u_{bi} \end{aligned} \quad (4.3.31)$$

and so the received sample for each of the parallel signals uniquely contains the symbols transmitted along a single channel of the parallel pair.

If the sampled impulse response is non-ideal but purely real,

$$r_{ai} + jr_{bi} = \sum_h s_{a(i-h)} y_h + u_{ai} + j \sum_h s_{b(i-h)} y_h + ju_{bi} \quad (4.3.32)$$

and again, one of the parallel channels conveys only one part of the complex signal. Figure 4.3.9 shows the relationship between the $\{s_i\}$ and the $\{r_i\}$ when the overall sampled impulse response is purely real.

In practice, the communication channel will be bandpass in form with a non-symmetrical amplitude response and will introduce varying degrees of attenuation and delay distortion. This not only increases the length of the sampled impulse response, which results each r_i containing several transmitted symbols, but also the sampled impulse response will be complex, which causes a loss of orthogonality between the two parallel data channels. Consequently, from equation (4.3.27),

$$r_{ai} + jr_{bi} = \sum_h \left[s_{a(i-h)} + js_{b(i-h)} \right] (y_{ah} + jy_{bh}) + (u_{ai} + ju_{bi}) \quad (4.3.33)$$

where y_{aj} and y_{bj} are the real and imaginary values of the j^{th} component in the sampled impulse response.

Hence,

$$r_{ai} = \sum_h (s_{a(i-h)} y_{ah}) - (s_{b(i-h)} y_{bh}) + u_{ai} \quad (4.3.34)$$

$$\text{and } r_{bi} = \sum_h (s_{a(i-h)} y_{bh}) + (s_{b(i-h)} y_{ah}) + u_{bi} \quad (4.3.35)$$

Equations (4.3.34) and (4.3.35) and Figure 4.3.10 clearly show that mutual interference between the two parallel channels will occur. In

general, the amount of interchannel interference will increase as the distortion introduced by the channel becomes more severe. Clearly, a channel introducing a lower amount of distortion will reduce both intersymbol and interchannel interference and for a given detector, the performance of the system will improve. This can be achieved by lowering the signal element rate of the system which will result in a shorter sampled impulse response with fewer high-order ripples, as briefly mentioned in section 2.2 and proved more rigorously in Chapter 5. However, for a given information rate, a reduction in signal element rate will necessitate an increase in the signal alphabet of both s_{ai} and s_{bi} . For these reasons, the following section on signal constellations introduces two structures, one of which requires the lower signal element rate of 2400 bauds compared with the other system operating at 3200 bauds.

4.4 Q.A.M. Signal Sets

The signal set or "constellation" of a QAM signal is the graphical, 2-dimensional representation of the possible values of the input and detected baseband data symbols. It has been mentioned that for the transmission of 19,200 bit/s over telephone circuits carrying QAM signals, the most suitable values of symbol rate and signal alphabet are,

$$\left. \begin{array}{l} \text{System A; } 3200 \text{ bauds, complex alphabet} = 64 \text{ level} \\ \text{System B; } 2400 \text{ bauds, complex alphabet} = 256 \text{ level} \end{array} \right\} \quad (4.4.1)$$

Since each transmitted symbol, s_i , and of course, each detected symbol, s'_i is complex, where;

$$\begin{aligned} s_i &= s_{ai} + js_{bi} \\ s'_i &= s'_{ai} + js'_{bi} \end{aligned} \quad (4.4.2)$$

and since the symbols are all multilevel with an alphabet of m , it follows that every complex symbol, sometimes called a "two-tuple" (14-15), can take one of m possible values. The \sqrt{m} possible values for both the

real and imaginary symbols are assumed to be such that the element waveforms are multilevel polar, with equal distances between every symbol in the signal space. The particular shape of signal constellation to be considered here is the rectangular signal set, although many other constellation shapes are possible (7-19, 22-30). The works of Foschini, et. al. (14,15) give particularly interesting discussions on the selection of signal constellations in the presence of Gaussian noise and phase jitter. Their conclusions, as are those of Harvey⁽³⁾ and Schwartz⁽¹⁾, suggest that although the rectangular constellation is not optimum, it does provide a good tolerance to phase jitter and Gaussian noise at the levels expected over telephone channels, and is particularly easy to implement; it is also popular for many commercial designs⁽²⁴⁻³⁰⁾.

The signal alphabets for the real and imaginary symbols for systems A and B are given by;

$$\text{System A; } \sqrt{m} = \sqrt{64} = 8$$

$$\text{System B; } \sqrt{m} = \sqrt{256} = 16 \quad (4.4.3)$$

and so the constellation for system A can be defined as the set whose signal points occur at;

$$\begin{array}{ll} \pm 1 & \pm j1 \\ \pm 3 & \pm j3 \\ \pm 5 & \pm j5 \\ \pm 7 & \pm j7 \end{array} \quad \begin{array}{l} \\ \text{with any of} \\ \\ \end{array} \quad (4.4.4)$$

For system B, each component of the complex symbols may have one of 16 possible values giving a constellation defined on;

± 1		$\pm j1$	
± 3		$\pm j3$	
± 5		$\pm j5$	
± 7	with any of	$\pm j7$	
± 9		$\pm j9$	
± 11		$\pm j11$	
± 13		$\pm j13$	
± 15		$\pm j15$	(4.4.5)

Using the terminology of Appendix 3, each symbol within the constellation carries B/S bits per symbol. Consequently, for a transmission rate of 19,200 bit/s,

$$\text{System A; } B/S = \frac{19200}{3200} = 6$$

$$\text{System B; } B/S = \frac{19200}{2400} = 8 \quad (4.4.6)$$

The performance of systems investigated in the study are based on the probability of bit errors (in a relative frequency sense) rather than the probability of symbol errors. Clearly, as each symbol can carry 6 or 8 bits of information, it will be advantageous to code the signal constellation such that a given symbol error at the receiver will cause as few bit errors as possible. This suggests the use of a Gray coding scheme where every set of 6 or 8 bits defining a particular symbol point differs by only 1 bit when compared with its immediate neighbours. Gray coding will be considered further after a much more important coding requirement, differential coding, has been discussed.

For the variety of reasons discussed in Chapter 3, sudden and quite large phase changes can occur over telephone channels which could cause the reference carriers of the demodulators to become unsynchronised with the signal carriers. Due to the quadrature relationship between the

reference carriers and the type of phase tracking loops used for phase synchronisation, the phase of the reference carriers is usually equally likely to be set to any multiple of 90° ; the two carriers, of course, remaining in phase quadrature. The result of these large phase errors is to change the position of the detected symbol values within the signal constellation. For example, a phase error of 180° changes the sign of both the real and imaginary components of the complex symbol and so the detected symbol (in the absence of all noise and distortion) is the value diagonally opposite the transmitted symbol in the signal constellation, as illustrated in Figure 4.4.1. To prevent long sequences of errors in the detected symbols due to phase jumps, the transmitted data symbols must be differentially coded, with the corresponding decoding operation at the receiver. Figure 4.4.2 shows the general arrangement of the encoder. The input binary data, $b_{in}(t)$ is first structured into blocks of length 6 or 8 bits depending on the system under consideration. Each block of bits is then mapped via the signal constellation into the real and imaginary parts of the complex symbol, as explained later. The corresponding symbol is not, however, transmitted directly through the system. Instead, the difference between the actual symbol to be transmitted and the previously transmitted symbol is computed and transmitted as a valid symbol value. At the receiver, the decoder compares the detected encoded data with the previously detected encoded data symbol and computes the symbol value being conveyed. This method ensures that phase ambiguity in the reference carriers will not cause long sequences of symbol errors since adjacent symbols suffer identical rotations and so the differential phase error is zero.

Several signal constellations have been designed and are discussed below. In the design of the signal sets, the objective was to create a constellation which exhibited full differential coding for protection against phase ambiguities of 90° , 180° and 270° . To further reduce bit errors due to noise causing the detector to select a symbol adjacent to the correct symbol, full Gray coding between all adjacent symbols within the constellations was also required. However, it was discovered that the exact realisation of full Gray coding with simultaneous differential coding is not obtainable for 64 point and 256 point structures, although Gray coding within each quadrant of the constellation, but not across the quadrant boundaries, can be achieved.

4.4.1 Coding Scheme 1; 64-point Structure

This is an extension of a combined differential and Gray coding method derived for 16 point QAM structures by Harvey⁽³⁾ and Fairfield⁽³³⁾.

Figure 4.4.3 shows the two dimensional space defining the signal constellation. The real and imaginary axes of the constellation represent the real part, s_{ai} , and the imaginary part, s_{bi} , of the complex symbol s_i , respectively. The $\{s_{ai}\}$ and $\{s_{bi}\}$ can each take-on one of the 8 possible values, $\pm 1, \pm 2, \pm 3, \pm 5, \pm 7$, giving 64 possibilities for the complex symbol, s_i . These 64 signal points are shown in Figure 4.4.3 as line crosspoints in the two-dimensional matrix. Each signal point is characterised in three ways;

- a) By the 6-bit block of input data which is assumed to map to the appropriate values of s_{ai} and s_{bi} .
- b) By the decimal equivalent of the 6-bit binary code plus 1; this is called the symbol number and is used in the generation of random

symbols in computer simulation studies. In these studies, a random number between 1 and 64 is generated to represent the i^{th} transmitted symbol, s_i . This number is then used as an address to locate the corresponding values of s_{ai} and s_{bi} held in a look-up table. At the receiver, the detected symbols, s'_{ai} and s'_{bi} are converted back into a symbol number from which the 6-bit binary pattern is decoded, again using a look-up table. The 6-bit patterns representing the transmitted symbol and detected symbol are then compared and the number of bit errors calculated.

- c) By the real and imaginary numbers representing the component parts of the complex symbols.

All vertically and horizontally adjacent signal points can be seen to be Gray coded, that is, the adjacent 6-bit blocks differ by only one binary digit. A detection error causing a single vertical or horizontal movement in the constellation would therefore only result in a single bit error. Signal points separated by a vertical or horizontal distance of 2 and points diagonally opposite each other differ by two binary digits. The structure shown in Figure 4.4.3 is called fully Gray coded since coding is maintained across the real and imaginary axes as well as within the four quadrants.

Based upon previous work⁽³³⁾, it was anticipated that Coding Scheme 1 would give protection against phase rotations of $\frac{n\pi}{2}$, $n = 1, 2, 3$, by using the coding and decoding tables shown in Figures 4.4.4 and 4.4.5. The first two binary digits of each 6-bit block represent the quadrant identifiers, or phase bits, whilst the remaining four bits, taken as two groups of two bits, give the position of a symbol within that quadrant, that is, they represent the imaginary and real amplitudes respectively.

Each 6-bit block is divided into three 2-bit sub-blocks for purposes of encoding. If $B_5 B_4 B_3 B_2 B_1 B_0$ are the binary digits of the symbol for transmission and if $A_5 A_4 A_3 A_2 A_1 A_0$ are the 6-bits representing the previous coded and transmitted data symbol, then the coder first compares $B_5 B_4$ with $A_5 A_4$ using Figure 4.4.4 to give $D_5 D_4$, the phase bits of the encoded symbol. $B_3 B_2$ are then compared with $A_3 A_2$, $B_1 B_0$ with $A_1 A_0$ again using Figure 4.4.4, to give $D_3 D_2$ and $D_1 D_0$. The 6-bit block $D_5 D_4 D_3 D_2 D_1 D_0$ is the coded symbol which is transmitted along the channel by the corresponding values of s_a, s_b obtained from the constellation, Figure 4.4.3.

The operation of the coding and decoding scheme will now be illustrated by example. Consider the transmission of three uncoded data symbols, s_{i-2}, s_{i-1}, s_i . It is assumed that the previously transmitted encoded symbol, $s_{d,i-3}$, has the binary representation given by 000000 (arbitrarily selected) and that the symbol has been correctly detected at the receiver. Further, let $s_{i-2} = 001110$, $s_{i-1} = 010001$ and $s_i = 010100$. The coder selects the phase bits of $s_{d,i-3}$, (00) and the phase bits of the next data symbol, s_{i-2} , (00) and supplies the phase bits of the encoded symbol, $s_{d,i-2}$, (00). The coder next takes the first pair of amplitude bits of $s_{d,i-3}$, (00) and the first pair of amplitude bits of s_{i-2} , (11), giving the encoded pair as (11). A similar operation is performed on the final pair of amplitude bits of $s_{d,i-3}$, (00) and of s_{i-2} , (10) to give (10). The three sets of bit-pairs are then regrouped to give the encoded symbol, $s_{d,i-2}$, (001110), which is transmitted as $s_{d,i-2} = 5 + j7$. At the receiver, the decoder works in a similar manner but compares bit-pairs of the previously detected encoded symbol with the corresponding bit-pairs of the most recently detected encoded symbol to yield the true data symbol. The full sequence of operations at the encoder and decoder is

given in Figure 4.4.6.

Having shown that the encoder and decoder act as an inverse pair, it is now necessary to determine whether or not the coded structure given in Figure 4.4.3 is tolerant to phase rotations of 90° , 180° and 270° in the transmitted encoded symbols. This is illustrated in Figure 4.4.7 for the case of a 90° phase rotation in the received, detected and encoded symbols for the same symbol values considered in Figure 4.4.6. The binary quantities shown in the first two columns of Figure 4.4.7 are obtained by locating the signal points representing the encoded data symbols as transmitted, and rotating these points by 90° within the signal constellation. For example, if $s_{d,k} = 3 + j5$, (symbol 8 in Figure 4.4.3), the rotated and detected symbol will be,

$$\begin{aligned} s''_{d,k} &= (3 + j5)j \\ &= -5 + j3 \\ &= \text{symbol 30 in Figure 4.4.3} \end{aligned} \tag{4.4.7}$$

the same result being obtained by rotating the signal point by 90° in the signal constellation.

Comparison between column 3 in Figure 4.4.6 and column 3 in Figure 4.4.7 shows that coding structure 1 does not protect the transmitted symbols from a 90° phase rotation. Further investigation reveals that neither 90° , 180° nor 270° phase rotations can be tolerated by the system and so coding structure 1 does not satisfy the requirements of a differentially coded system. The reasons for the failure of structure 1 can be seen with reference to Figures 4.4.3, 4.4.6 and 4.4.7 and by observing that the first two bits, the phase bits, are in fact protected against phase rotations. Rotation of a signal point by 90° , 180° or 270° necessarily causes the phase bits to change as the rotated signal points lie in a different quadrant. Figure 4.4.8 shows the change in the phase bits as

a signal point is rotated through 90° , 180° and 270° . Any rotation of phase due to errors between the signal carriers and reference carriers will therefore be differentially cancelled by the encoding and decoding operations of Figures 4.4.6 and 4.4.7. However, the amplitude bits in Figure 4.4.3, when taken in pairs, do not possess the same rotational consistency. For example, the amplitude bits 01|11 should ideally map to 11|10 after a 90° rotation, by following the rule given in Figure 4.4.8; in general, the amplitude bits in structure 1 do not meet this requirement nor in fact, are they consistent with any other mapping rule, and so the structure does not allow the differential coding of all 6 bits.

Several modifications were made to the basic structure of Coding Scheme 1 in an attempt to provide simultaneous rotational consistency of the 6 bits of all signal points and full Gray coding. It was discovered however, that the requirements of full Gray coding and differential coding are contradictory for rectangular constellations with more than 4 points.

4.4.2 Coding Scheme 2, 64-Point Structure

Based upon the conclusions for Coding Scheme 1 and that the major requirement of a coded constellation is to provide differential protection against phase rotations of 90° , 180° and 270° , a family of signal constellations were constructed which, as an overriding condition, exhibited differential coding. The resulting family of constellations were then subjected to a ranking scheme which classified them in terms of the closest approximation to the ideal, fully Gray coded constellation. The scheme shown in Figure 4.4.9, known here as Coding Scheme 2, shows the coded constellation with optimum Gray coding in the presence of full differential coding across all four quadrants. It can be seen from the figure that all signal

points within the separate quadrants are fully Gray coded but adjacent points across the real and imaginary axes differ by 3 binary digits rather than a single binary digit in the ideal fully coded case.

As with Coding Scheme 1, the first two bits of a 6-bit block identify the quadrant of the corresponding data symbol and are called the phase bits of that symbol. The remaining four bits are again called the amplitude bits but it should be noticed that these bits no longer uniquely define the position of a signal point within all four quadrants; they do, however, define the position of a signal point within a single quadrant, as determined by the phase bits of that quadrant. The operation of the coder and decoder will now be described and illustrated by example using the same nomenclature, but different symbol values from those used in the example of Section 4.4.1.

A 2-bit store in the encoder holds the last two bits of the previously transmitted encoded symbol, $A_5 A_4$. In practice, as illustrated by the constellation in Figure 4.4.9, these two bits will be the phase bits of the transmitted symbol as this will give increased protection against error extension effects. This follows since the phase information of the detected symbol is likely to be correct even if the symbol as a whole is detected erroneously, except of course, if the transmitted symbol lies close to the real or imaginary axis of the constellation. The coder then compares these stored phase bits with the phase bits of the new data symbol, $B_5 B_4$, to give $D_5 D_4$ which are the phase bits of the encoded symbol. The two pairs of amplitude bits of the previously coded symbol, $A_3 A_2$ and $A_1 A_0$, are then compared with the corresponding bit pairs, $B_3 B_2$ and $B_1 B_0$ of the new data symbol to yield $D_3 D_2$ and $D_1 D_0$. The coded 6-bit

block is then converted to the appropriate values of s_{ai} and s_{bi} obtained from the constellation and the symbol is transmitted over the transmission path as $s_i = s_{ai} + js_{bi}$. The coder and decoder therefore operate on the symbols two bits at a time and so the encoder unit can be implemented easily by using a shift register arrangement and a 32-bit memory configured as a 16 x 2-bit look-up table, as shown in Figure 4.4.10. A somewhat simpler implementation of the coder unit is shown in Figure 4.4.11 which uses a 4K x 6 look-up table. Although a larger memory device is required compared with Figure 4.3.10, the availability of cheap memory and the overall lower complexity of this method make it a more practical proposition; the corresponding decoder implementation is shown in Figure 4.4.12.

The coding and decoding protocols can be conveniently shown by using the coding diagram of Figure 4.4.13. This diagram shows the method of coding the 2-bit sub-blocks of the symbol binary values and can also be used to determine the contents of the memory units in Figures 4.4.11 and 4.4.12. Coding is achieved by moving to the circle (in Figure 4.3.13) containing the phase bits of the previously transmitted coded symbol and moving along the line labelled with the phase bits of the latest data symbol. The circle at the end of this line contains the phase bits of the coded data symbol. This process is then repeated for the two pairs of amplitude bits. At the decoder, move to the circle containing the phase bits of the previously detected encoded symbol. Then, from the circle, move to the circle containing the phase bits of the most recently detected encoded symbol; the line between these circles gives the phase bits of the decoded data symbol. Again, the process is repeated for the remaining two pairs of bits. The full sequence of operations at the

encoder and decoder is shown in Figure 4.4.14, where, as in earlier examples, three data symbols are to be coded and decoded, these symbols now being $s_{i-2} = 000111$, $s_{i-1} = 100100$, $s_i = 110101$. The previously transmitted coded symbol is assumed to be $s_{d,i-3} = 111111$.

Using the phase rotation table given in Figure 4.4.8, which also applies to Coding Scheme 2 as can be seen from Figure 4.4.9, the operation of the decoder in the presence of 90° , 180° and 270° phase rotations can be determined and is given in Figure 4.3.15. Comparison between the third columns of Figures 4.4.14, 4.4.15 a), b) and c) shows that Coding Scheme 2 indeed satisfies the differential coding requirement since the decoded data has been protected against phase ambiguities of $n\frac{\pi}{2}$, $n=1, 2, 3$.

An alternative method of using the constellation shown in Figure 4.4.9 and the coding chart shown in Figure 4.4.13, is to use the previously coded bit pair as the reference bits for the encoding process. That is, instead of comparing a bit pair of the new data symbol with the corresponding bit pair of the previously encoded symbol, each pair of bits in the uncoded data symbol is mapped to the coded pair by comparison with the immediately previous pair of coded bits. The coder unit may now be implemented as shown in Figure 4.4.16, where the contents of the memory is identical to that of the memory shown in Figure 4.4.10. A listing of the memory content is given in Figure 4.4.17 and is based on the coding chart of Figure 4.4.13.

Using the same symbol values as in the previous example, Figure 4.4.18 shows the operation of the modified coder/decoder, where it is assumed that the previously encoded bit pair (i.e. the phase bits of the previously encoded symbol) is 1 1. Since both of the coders/decoders described in this section operate on one pair of bits at a time, and since the first

arrangement has been shown to exhibit tolerance to phase rotations of $n\frac{\pi}{2}$, $n=1, 2, 3$, it follows that the second coding/decoding method will also protect the decoded data symbols from these phase rotations. For completeness, Figure 4.4.19 illustrates the operation of the decoder when the transmitted symbols have suffered a 270° phase rotation; comparison of column 3 in Figure 4.4.18 and column 3 in Figure 4.4.19 shows that the decoded data symbols are indeed protected.

4.4.3 Coding Scheme 3, 64 point structure

The signal constellation for Coding Scheme 3 is shown in Figure 4.4.20. As with Coding Scheme 2, coding is performed two bits at a time and obeys the rules of the coding chart in Figure 4.4.13. Also, the first two bits of the 6-bit block again define the quadrant of the signal point and the remaining four bits define the amplitude (real and imaginary) within that quadrant. The advantage of this scheme over Coding Scheme 2 is that only the phase bits of the binary representation of a data symbol need be coded since the positional bits within each quadrant have been carefully designed to exhibit rotational consistency. This is illustrated in Figure 4.4.21 where the symbol 001101 (symbol number 14) is rotated by 90° , 180° and 270° to give respectively, 011101, 111101 and 101101. Clearly, the quadrant identifiers have changed as expected due to the rotations and as predicted by Figure 4.4.8, but the four bits defining the position of the symbols within the quadrants have remained at 1101. This may be shown mathematically as follows. The four positional bits define the positions of the (16) signal points within the all-positive quadrant, that is, the first two bits define the position of the point along the imaginary axis whereas the last pair of bits define position in terms of

the real axis according to;

$$00 \rightarrow +1$$

$$01 \rightarrow +3$$

$$11 \rightarrow +5$$

$$10 \rightarrow +7 \quad (4.4.8)$$

The positional bits 1101 therefore locate the signal point at $+3+j5$, shown as point A in Figure 4.4.21. Rotation of this point by 90° gives;

$$(+3+j5)j = -5+j3 \quad (4.4.9)$$

which is located at point B in Figure 4.4.21 and corresponds to binary

011101. Rotation by a further 90° gives;

$$(-5+j3)j = -3-j5 \quad (4.4.10)$$

which is point C in the figure and corresponds to symbol number 62 (111101).

Rotation by a total of 270° gives

$$(-3-j5)j = 5-j3 \quad (4.4.11)$$

which is shown as point D in Figure 4.4.21 and is the point 101101.

Clearly, the first two bits of a 6-bit block determine the quadrant of the corresponding signal point whereas the last 4 bits uniquely determine the position of the signal point in the all-positive quadrant and will also define its position in the quadrant of interest if the all-positive quadrant is rotated by an angle determined by the first two bits, where;

$$00 \rightarrow \text{no rotation}$$

$$01 \rightarrow \text{rotate by } 90^\circ$$

$$11 \rightarrow \text{rotate by } 180^\circ$$

$$10 \rightarrow \text{rotate by } 270^\circ \quad (4.4.12)$$

Consequently, the complete signal constellation of Figure 4.4.20 is uniquely defined by the phase bits and the all-positive quadrant only.

The implementation of the coder and decoder now reduces to that shown in Figure 4.4.22 where the contents of the memories used as the mapping tables are determined from the coding diagram in Figure 4.4.13. Using the nomenclature;

$A_5 A_4 A_3 A_2 A_1 A_0$ = Binary-coded representation of the previously coded symbol,

$B_5 B_4 B_3 B_2 B_1 B_0$ = Binary-coded representation of the data symbol to be encoded,

then the binary representation of the coded data symbol will be

$D_5 D_4 B_3 B_2 B_1 B_0$ where $D_5 D_4$ are the coded phase bits as determined from either the coding chart of Figure 4.4.13 or from the memory contents of Figure 4.4.8, and are obtained by comparing $A_5 A_4$ with $B_5 B_4$.

As in the case of Coding Scheme 2, Gray coding is achieved in each of the four quadrants in Figure 4.4.20, but the price paid for the rather elegant and simple differential coder/decoder is an increase in the number of binary digits changed between adjacent symbols across the real and imaginary axes. Since the data symbols are assumed to be statistically independent and equally likely to have any of their 64 possible values, the increase in bit error rate due to this less-than-optimum Gray coding can be considered negligible when compared with the major causes of erroneous symbol detection, that is, distortion and other signal impairments introduced by the transmission channel.

The performance of Coding Scheme 3 in terms of its tolerance to phase rotations of $n\frac{\pi}{2}$ rads, $n=1,2,3$, can be predicted from the discussion in Section 4.4.2 and its application to the constellation of Figure 4.4.20. However, for completeness, the signal constellation and the coder/decoder

operations were modelled and simulated on a digital computer. The simulation program appears as Program 1 in Appendix 10 whereas the simulation results are given in Appendix 4.

4.4.4 Coding Scheme 4, 256-point structure

Coding scheme 4 and the signal constellation of Figure 4.4.23 is an extended version of Coding Scheme 3 and applies to a 256-point signal set. Each signal point is now defined in terms of 8-bits of the incoming data. Consequently, the symbol rate is now 2400 bauds and each of the real and imaginary components in a complex data symbol have 16 possible values. The first two bits of the 8-bit binary representations of the data symbols again define the phase or quadrant of the signal point in the constellation, the remaining 6-bits defining the position within the quadrant; as for Coding Scheme 3, the 6 positional bits are rotationally consistent. For this reason and for clarity of the diagram, Figure 4.4.23 shows only the all-positive quadrant of the 256-point structure, the remaining three quadrants being obtained by changing the phase bits and rotating the quadrant as described in Section 4.4.3. As for Coding Scheme 3, it is only necessary to code the two phase bits and so the coder/decoder for Coding Scheme 4 may be implemented in a similar manner to Figure 4.4.22, the number of bits, of course, being increased to eight.

For the rest of this study, Coding Schemes 3 and 4 with the associated signal constellation of Figures 4.4.20 and 4.4.23 will be used for all computer simulations and designs for the 64-point, 3200 baud and 256-point, 2400 baud QAM systems, respectively. Appendix 5 derives expressions for

the error probabilities of the 64-point and 256-point QAM structures when operating over an ideal telephone circuit which introduces no distortion nor signal impairments except additive white Gaussian noise.

4.5 References

1. Schwartz M., "Information Transmission, Modulation and Noise", McGraw-Hill Inc., Third Edition, 1980
2. Clark, A.P., "Principles of Digital Data Transmission", Pentech Press, 1976.
3. Harvey, J.D., "Synchronisation of a Synchronous Modem"
SERC Report No. GR/A/1200.7, December 1980
4. Clark, A.P., "Advanced Data-Transmission Systems", Pentech Press, 1977
5. Clark, A.P. and Harvey, J.D., "Detection of Distorted Q.A.M. Signals", Electronic Circuits and Systems, Vol 1, No.3, pp 103-109, April 1977
6. Harvey, J.D., Clark, A.P. and Driscoll, J.P., "High Speed Line Modems"
Final Report No. K/M2 11a/616, Loughborough University of Technology
7. Proakis, J.G., "Digital Communications", McGraw-Hill Inc., 1983
8. Cahn, C.R., "Combined Digital Phase and Amplitude Modulated Communication Systems", IRE Trans. Comms. Systems, Vol. CS-8, pp 150-155, Sept. 1960

9. Hancock, J.C. and Lucky, R.W., "Performance of Combined Amplitude and Phase Modulated Communication Systems", IRE Trans. Comms. Systems, Vol. CS-8, pp 232-237, Dec. 1960
10. Lucky, R.W. and Hancock, J.C., "On the Optimum Performance of N-ary Systems having Two Degrees of Freedom", IRE Trans. Comms. Systems, Vol CS-10 pp 185-192, June 1962
11. Salz, J., Sheehan J.R. and Paris, D.J., "Data Transmission by Combined AM and PM", Bell System Tech. J., Vol.50, pp 2399-2419, Sept. 1971
12. Simon, M.K. and Smith J.G., "Hexagonal Multiple Phase-and-Amplitude Shift Keyed Signal Sets", IEEE Trans. Comms., Vol. COM-21, pp 1108-1115, Oct. 1973
13. Thomas, C.M., Weidner, M.Y. and Durrani, S.H., "Digital Amplitude-Phase Keying with M-ary Alphabets", IEEE Trans. Comms. Vol. COM-22, pp 168-180, Feb. 1974
14. Foschini, G.J., Gitlin, R.D. and Weinstein, S.B., "On the Selection of a Two-Dimensional Signal Constellation in the Presence of Phase Jitter and Gaussian Noise", Bell System Tech.J., Vol.52, No.6, pp 927-965, July-August 1973

15. Foschini, G.J., Gitlin, R.D. and Weinstein, S.B., "Optimisation of 2-Dimensional Signal Constellations in the Presence of Gaussian Noise", IEEE Trans. Comms., Vol COM-22, pp 28-37, 1974
16. Falconer, D.D. and Magee, F.R., "Evaluation of Decision Feedback Equalisation and Viterbi Algorithm Detection for Voice-Band Data Transmission, Parts 1 and 2", IEEE Trans. Comms., Vol. COM-24, pp 1130-1139 and 1238-1245, 1976
17. Prabhu, V.K., "The Detection Efficiency of 16-QAM", Bell System Tech. J., Vol.59, pp 639-656, 1980
18. Thomas, C.M., Mag, C.L. and Welty, G.R., "Hybrid Amplitude and Phase Modulation for Analog Data Transmission", IEEE Trans. Comms., Vol COM-23, pp 634-645, June 1975
19. Steber, J.M., "Understanding P.S.K. Demodulation Techniques", Microwaves and R.F., pp 137-145, March 1984 and pp 113-118, April 1984
20. Clark, A.P. and Fairfield, M.J., "Detection Processes for a 9600 bit/s Modem", Radio and Electronic Engineer, Vol.51, No.9, pp 455-465, Sept. 1981
21. Clark, A.P., Najdi, H.Y. and Fairfield, M.J. "Data Transmission at 19.2 Kbit/s over Telephone Circuits", Radio and Electronic Engineer, Vol.53, No.4, pp 157-166, April 1983

22. Sklar, B., "A structured Overview of Digital Communications - A Tutorial Review, Parts 1 and 2", IEEE Comm. Mag., Vol.21, No.5, pp 4-17, Aug. 1983 and Vol.21, No.7, pp 6-21, Oct. 1983
23. Rockwell, D., "AT and T Introduce 64-QAM Digital Microwave Radio", Microwaves and RF, pp 43, 46, 48, 61; August 1984.
24. Logan, H.L. Jnr., and Forney, G.D. Jnr., "A MOS/LSI Multiple-Configuration 9600 BPS Data Modem", Codex Corp. Report 48, pp 7-12
25. Murano, K. et. al., "Multiprocessor Architecture for Voiceband Data Processing, (Application to 9600 BPS Modem)", Fujitsu Labs. Report 37.3, pp 1-5, 1979
26. Watanabe, K. et. al., "A 4800 BPS Microprocessor Data Modem", N.E.C. Co.Ltd. Report 47.6, pp 252-256
27. Akashi, F. et. al., "A High Performance Digital QAM 9600 bit/s Modem", N.E.C. Res. and Dev. No.45, pp 38-48, April 1977
28. Jones, J. and Tearher, V., "Modem at 2400 Bits per Second for Data Transmission over Telephone Lines", Electrical Communication, Vol.44, No.1, pp 66-71, 1969
29. Murano K. et. al., "LSI Processor for Digital Signal Processing and its Application to 4800 Bit/s Modem", IEEE Trans. Comms. Vol. COM-26, No.5, pp 499-506, May 1978

30. Brownlie, J.D. et. al., "Custom-Designed Integrated Circuits for Data Modems", B.T. Tech. J., Vol.3, No.1, pp 14-19, Jan. 1985
31. Najdi, H.Y., "Digital Data Transmission Over Voice Channels", Ph.D. Thesis, Loughborough University of Technology, Loughborough, Leics. 1982
32. McVerry, F., "High Speed Data Transmission Over HF Radio Links", Ph.D. Thesis, Loughborough University of Technology, 1982
33. Clark, A.P. and Fairfield, M.J., "Detection Processes for a 9600 bit/s Modem", The Radio and Electronic Engineer, Vol. 51, No.9, pp 455-465, Sept. 1981.

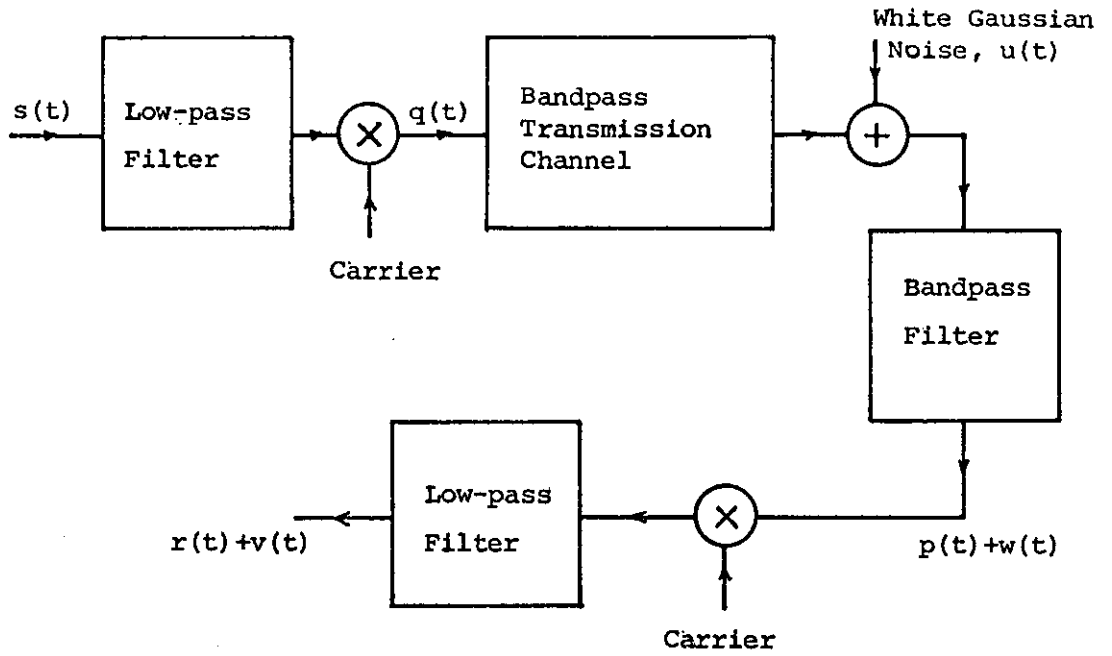


Figure 4.2.1: Model of bandpass communication system

— 3.1

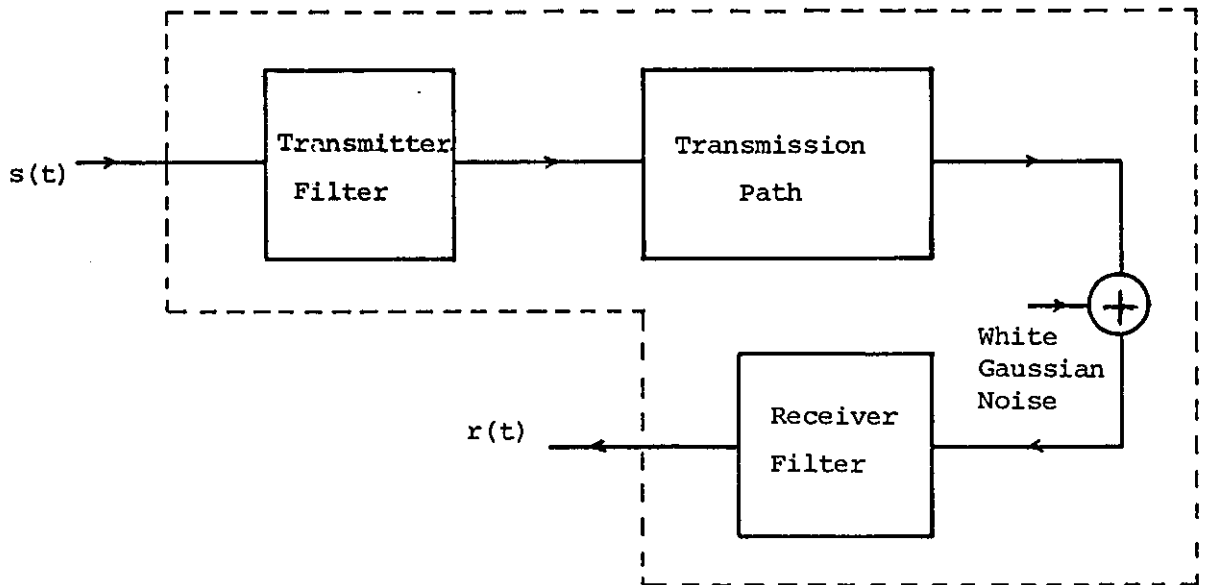


Figure 4.2.2: Equivalent Baseband Model

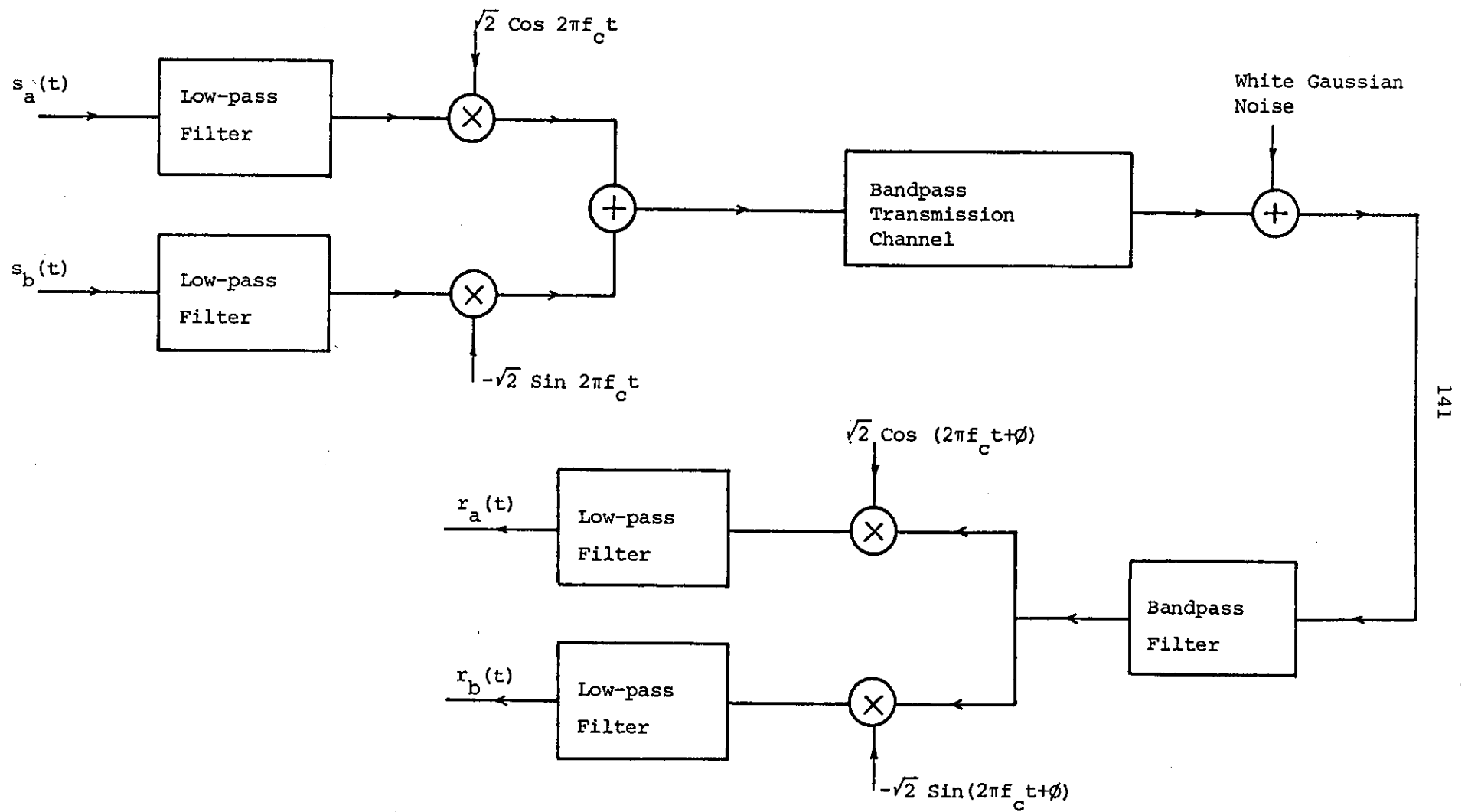


Figure 4.2.3: Model of 2-dimensional bandpass communication system.

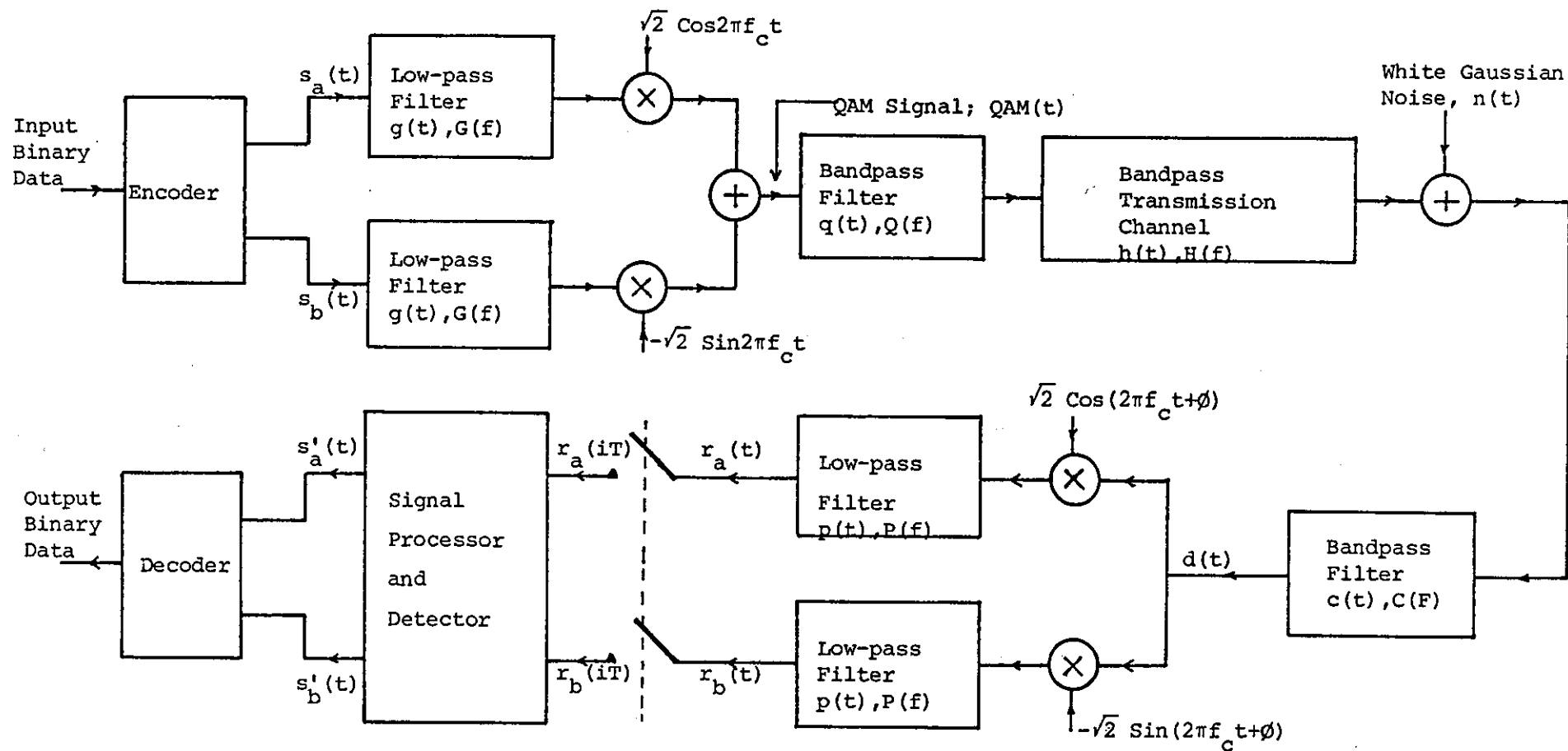


Figure 4.3.1: Model of QAM data transmission system

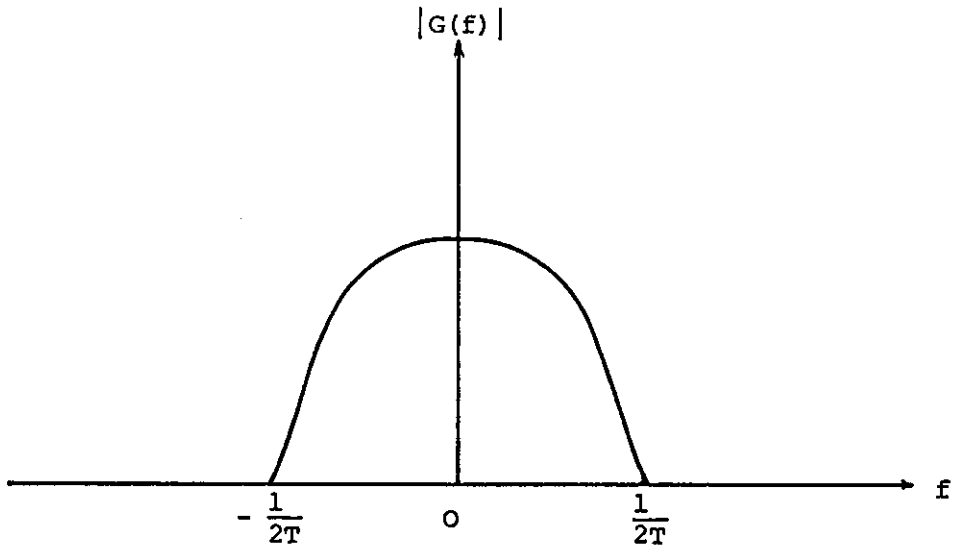


Figure 4.3.2: Amplitude response of transmitter low-pass filters

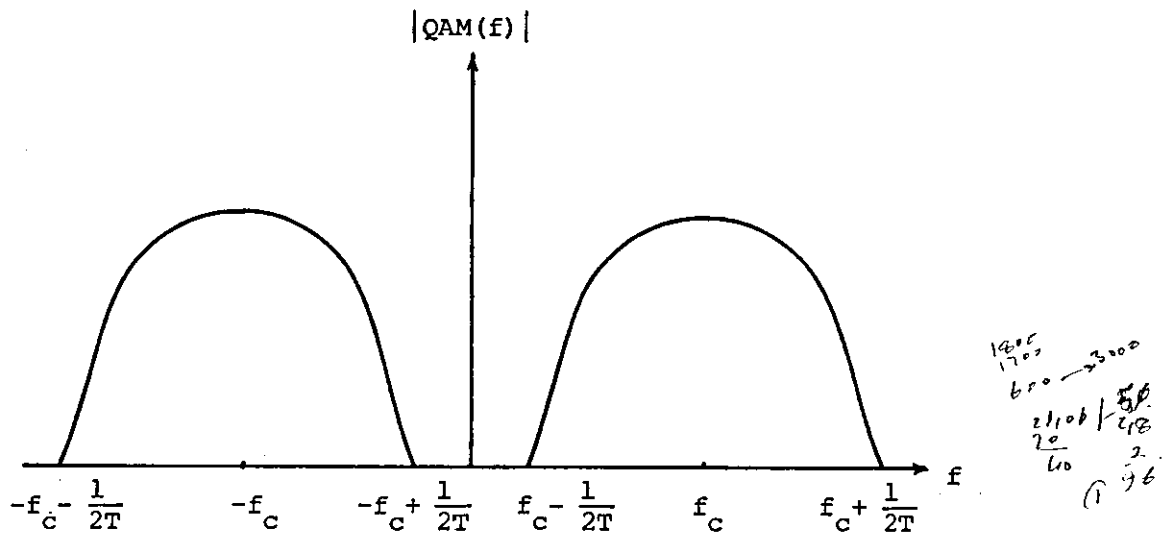


Figure 4.3.3: Amplitude spectrum of the QAM signal at the input to the transmission channel

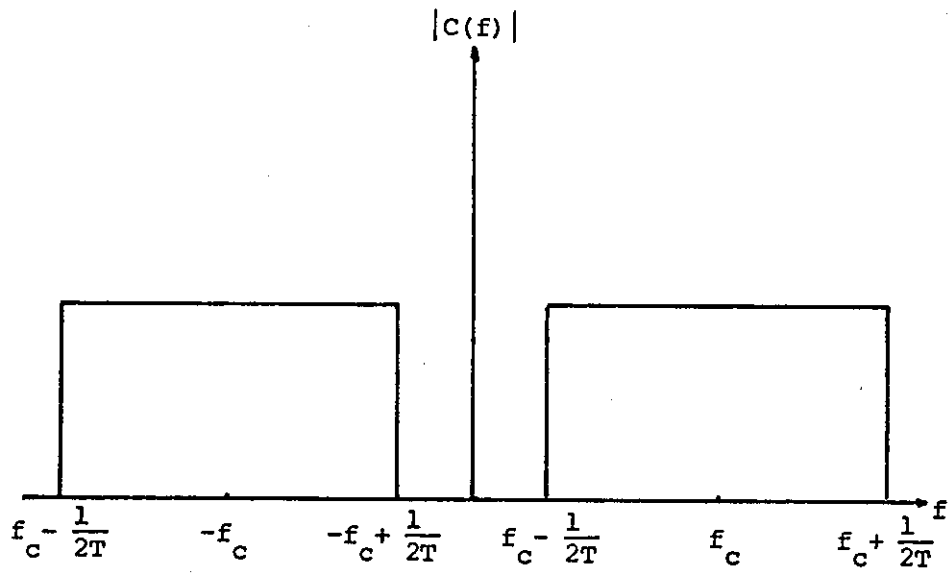


Figure 4.3.4: Ideal Amplitude Response of Receiver Bandpass Filter

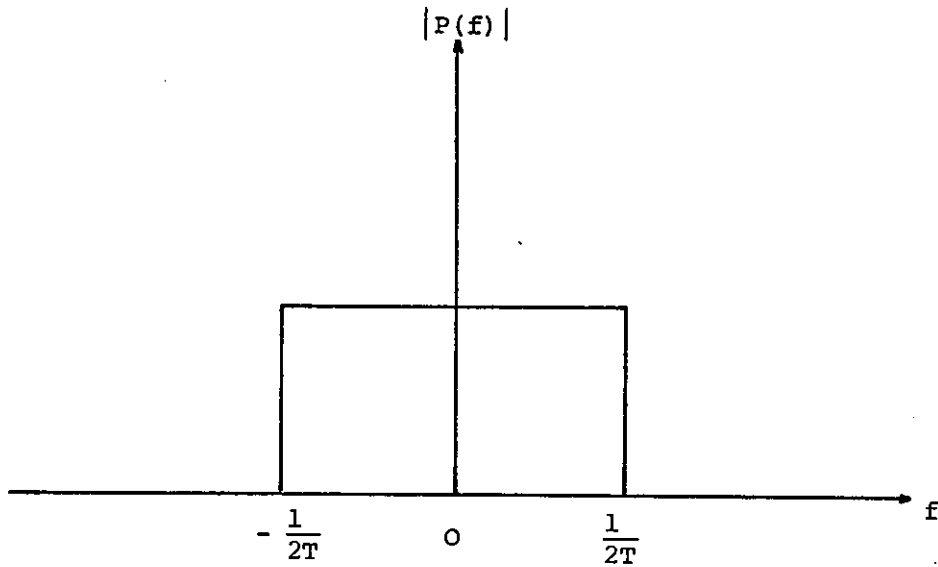


Figure 4.3.5: Ideal Amplitude Response of Post-Demodulation Low-Pass Filters

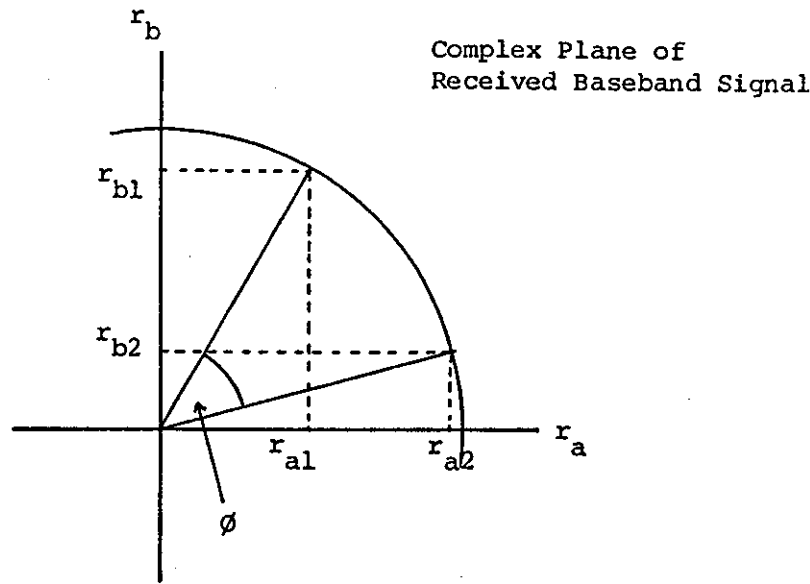


Figure 4.3.6: Effect of Phase Error on the Allocation of the Quadrature Components of the Received Baseband Signal

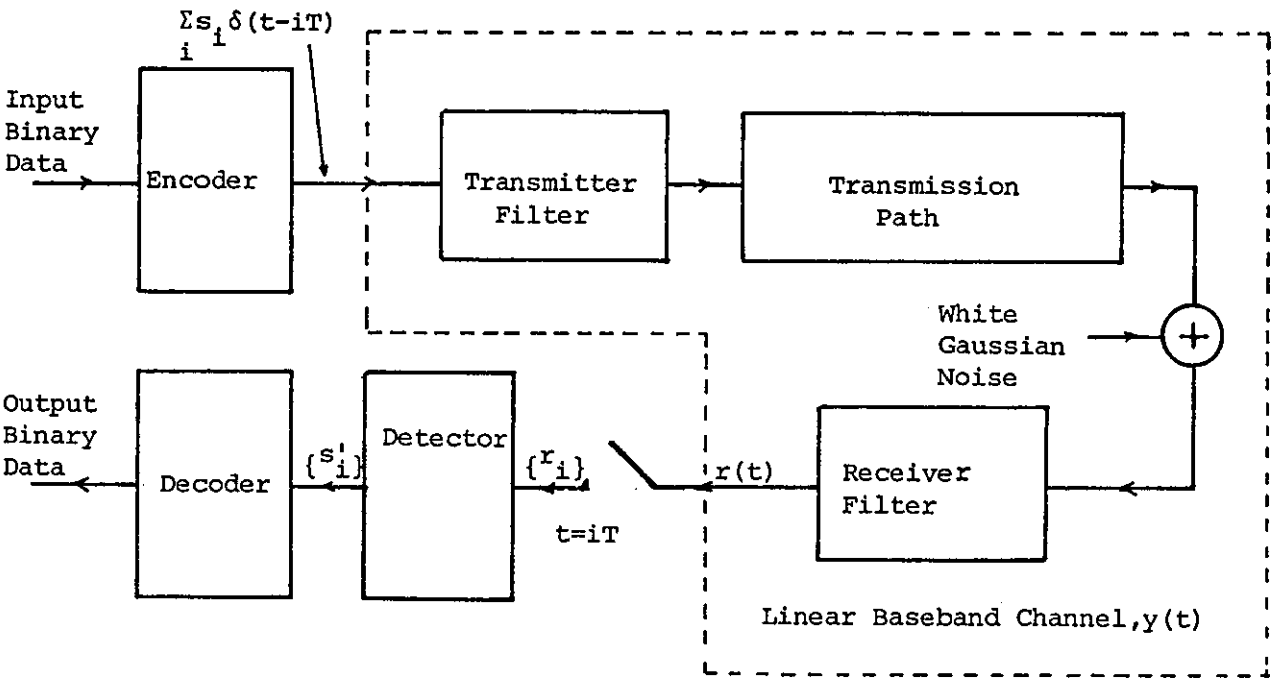


Figure 4.3.7: Equivalent Baseband Model of the QAM System Shown in Fig. 4.3.1

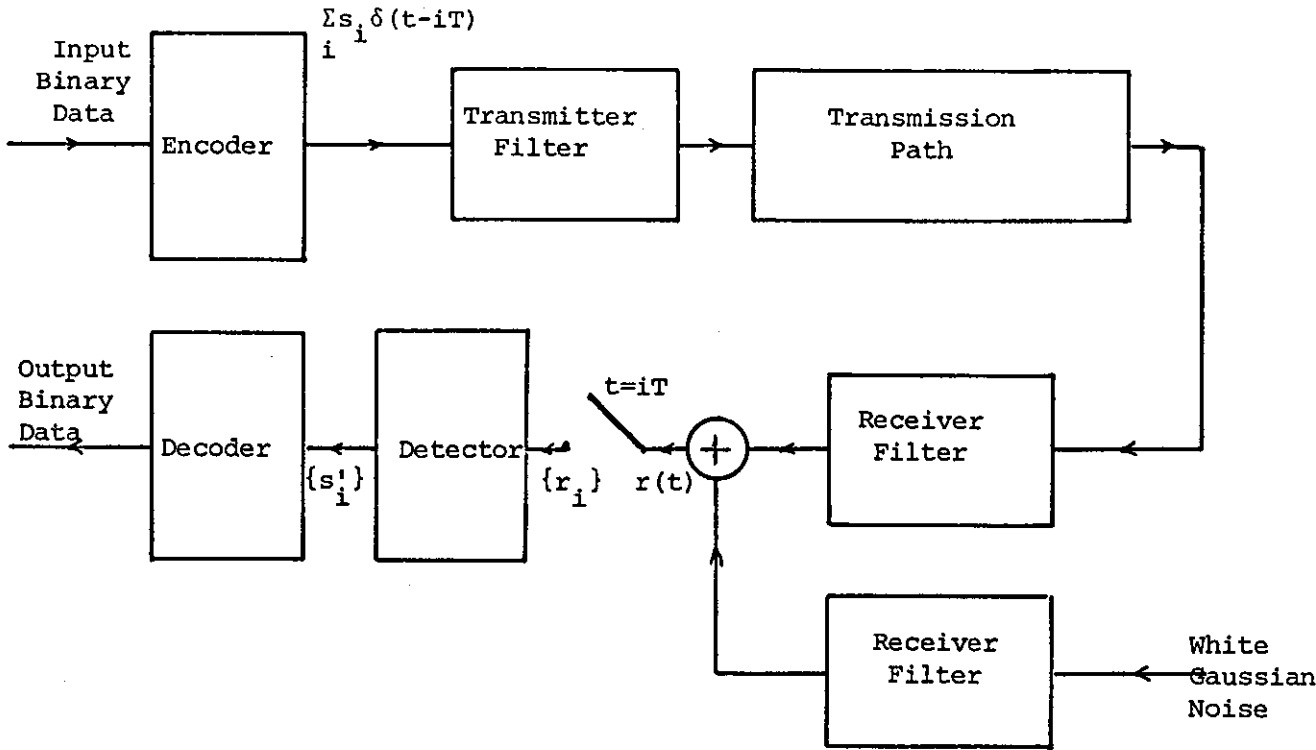


Figure 4.3.8: Modified Baseband Model for Use in Computer Simulations

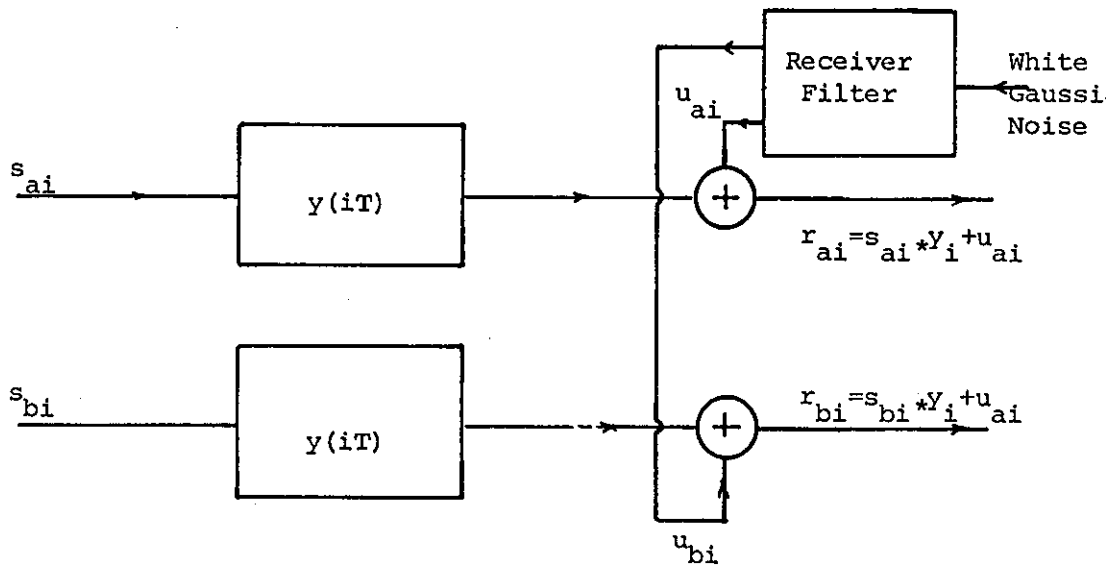


Figure 4.3.9: Equivalent Model for a Purely Real Baseband Channel

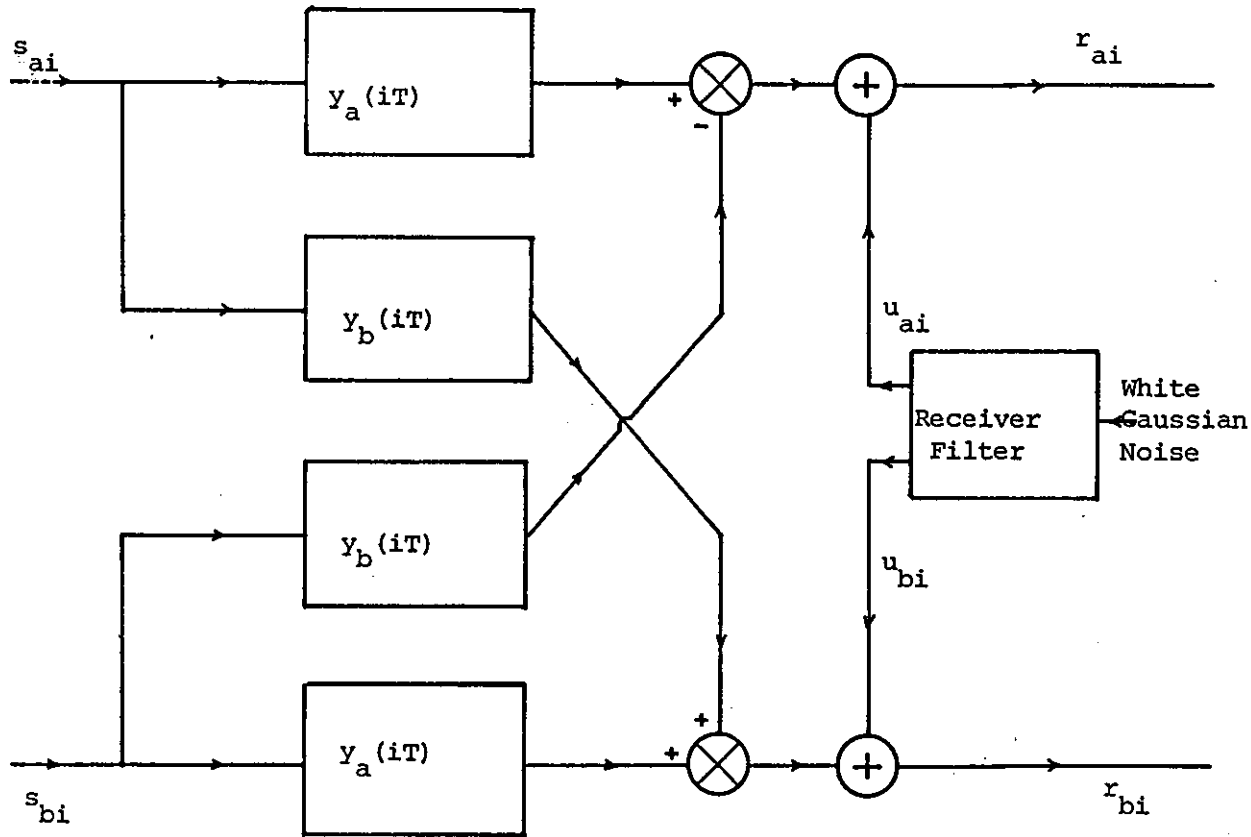


Figure 4.3.10: Equivalent Model for a Complex Baseband Channel

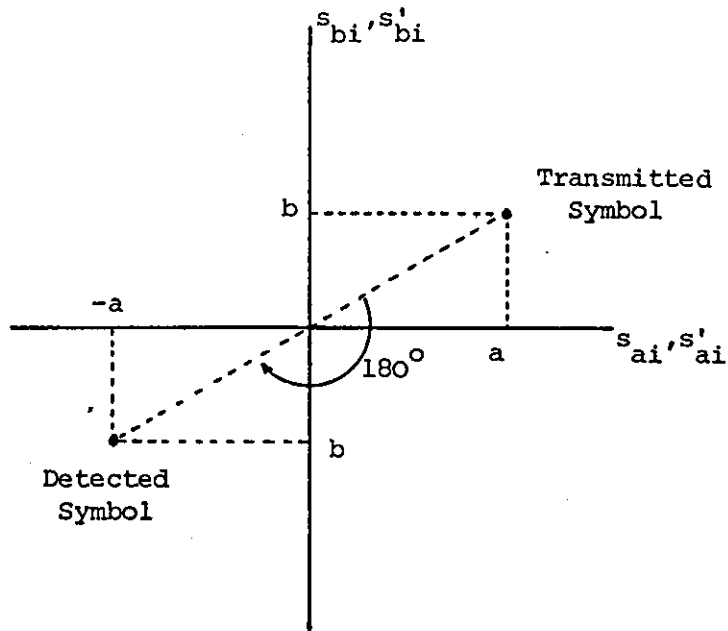


Figure 4.4.1: Erroneous Symbol Detection Due to 180° Phase Ambiguity at the Receiver

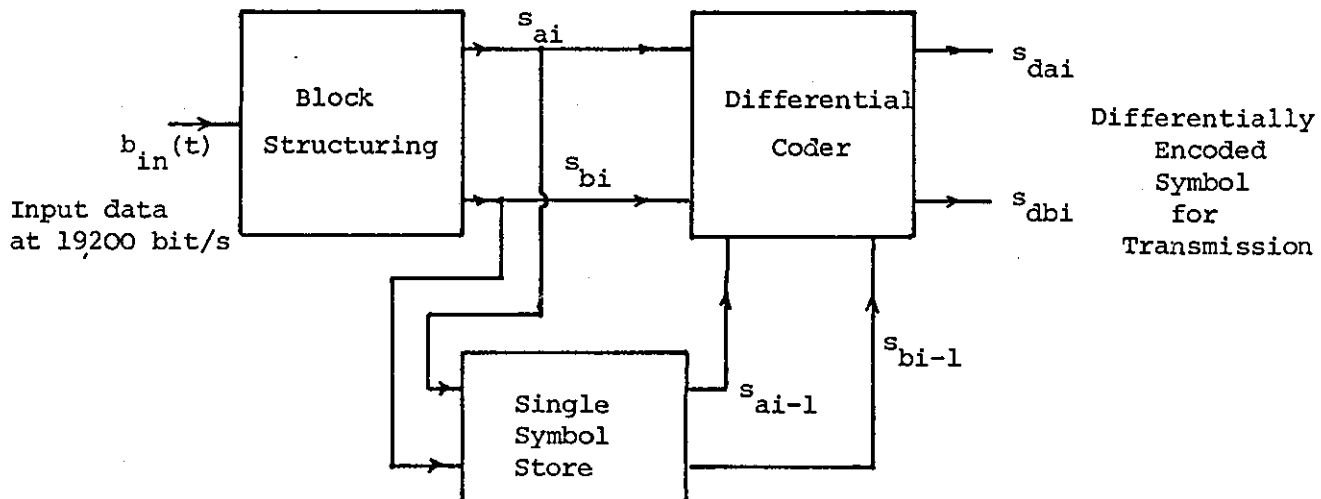


Figure 4.4.2: General Arrangement for Differentially Encoding the Data Symbols

				s_{bi}					
011010	011110	010110	010010		000010	000110	001110	001010	
27	31	23	19	7	3	7	15	11	
011011	011111	010111	010011		000011	000111	001111	001011	
28	32	24	20	5	4	8	16	12	
011001	011101	010101	010001		000001	000101	001101	001001	
26	30	22	18	3	2	6	14	10	
011000	011100	010100	010000		000000	000100	001100	001000	
25	29	21	17	1	1	5	13	9	
-7	-5	-3	-1		1	3	5	7	s_{ai}
111000	111100	110100	110000		100000	100100	101100	101000	
67	61	53	49	-1	33	37	45	41	
111001	111101	110101	110001		100001	100101	101101	101001	
58	62	54	50	-3	34	38	46	42	
111011	111111	110111	110011		100011	100111	101111	101011	
60	64	56	52	-5	36	40	48	44	
111010	111110	110110	110010		100010	100110	101110	101010	
59	63	55	51	-7	35	39	47	43	

Figure 4.4.3: Signal Constellation for Coding Scheme 1;
64-Point Structure

New Symbol (Bit-Pairs)	Previously Coded Symbol (Corresponding Bit-Pair)	Differen- tially Coded Symbol (Bit-Pairs)
00	00	00
00	01	01
00	11	11
00	10	10
01	00	01
01	01	11
01	11	10
01	10	00
11	00	11
11	01	10
11	11	00
11	10	01
10	00	10
10	01	00
10	11	01
10	10	11

Figure 4.4.4: Coding Table for Coding Scheme 1

New Detected Symbol (Bit-Pairs)	Previously Detected Symbol (Corresponding Bit-Pairs)	Decoded Symbol (Bit-Pairs)
00	00	00
00	01	10
00	11	11
00	10	01
01	00	01
01	01	00
01	11	10
01	10	11
11	00	11
11	01	01
11	11	00
11	10	10
10	00	10
10	01	11
10	11	01
10	10	00

Figure 4.4.5: Decoding Table for Coding Scheme 1

New Uncoded Symbol	Previously Transmitted Encoded Symbol	New Coded Symbol For Transmission
		$s_{d,i-3}'$ 00 00 00
s_{i-2} 00 11 10	$s_{d,i-3}$ 00 00 00	$s_{d,i-2}$ 00 11 10
s_{i-1} 01 00 01	$s_{d,i-2}$ 00 11 10	$s_{d,i-1}$ 01 11 00
s_i 01 01 00	$s_{d,i-1}$ 01 11 00	$s_{d,i}$ 11 10 00

a) Coder Operation

New Detected Encoded Symbol	Previously Detected Encoded Symbol	Resultant Data Symbol
$s_{d,i-2}'$ 00 11 10	$s_{d,i-3}'$ 00 00 00	s_{i-2}' 00 11 10
$s_{d,i-1}'$ 01 11 00	$s_{d,i-2}'$ 00 11 10	s_{i-1}' 01 00 01
$s_{d,i}'$ 11 10 00	$s_{d,i-1}'$ 01 11 00	s_i' 01 01 00

b) Decoder Operation

Figure 4.4.6: Coder and Decoder Operations for Coding Scheme 1

New Detected. Encoded Symbol (Rotated by 90°)	Previously Detected Encoded Symbol (Rotated by 90°)	Resultant Data Symbol
$s''_{d,i-2}, 01 10 11$	$s''_{d,i-3}, 01 00 00$	$s''_{i-2}, 00 10 11$
$s''_{d,i-1}, 11 00 11$	$s''_{d,i-2}, 01 10 11$	$s''_{i-1}, 01 01 00$
$s''_{d,i}, 10 00 10$	$s''_{d,i-1}, 11 00 11$	$s''_i, 01 00 01$

Figure 4.4.7: Decoder Operation for Coding Scheme 1 in the Presence of a 90° Phase Ambiguity Between the Signal and Reference Carriers

Original Phase Bits	Phase Bits After Rotation by		
	90°	180°	270°
00	01	11	10
01	11	10	00
11	10	00	01
10	00	01	11

Figure 4.4.8: Change of Phase Bits in Coding Scheme 1 Due to Phase Rotations of 90° , 180° and 270°

				s_{bi}					
010000	011000	011100	010100		001000	001001	001011	001010	
17	25	29	21	7	9	10	12	11	
010001	011010	011110	010110		001100	001101	001111	001110	
19	27	31	23	5	13	14	16	15	
010011	011011	011111	010111		000100	000101	000111	000110	
20	28	32	24	3	5	6	8	7	
010010	011001	011101	010101		000000	000001	000011	000010	
18	26	30	22	1	1	2	4	3	
-7	-5	-3	-1		1	3	5	7	s_{ai}
111101	111100	111110	111111		101010	100010	100110	101110	
62	61	63	64	-1	43	35	39	47	
111001	111000	111010	111011		101000	100000	100100	101100	
58	57	59	60	-3	41	33	37	45	
110001	110000	110010	110011		101001	100001	100101	101101	
50	49	51	52	-5	42	34	38	46	
110101	110100	110110	110111		101011	100011	100111	101111	
54	53	55	56	-7	44	36	40	48	

Figure 4.4.9: Signal Constellation for Coding Scheme 2:
64-Point Structure

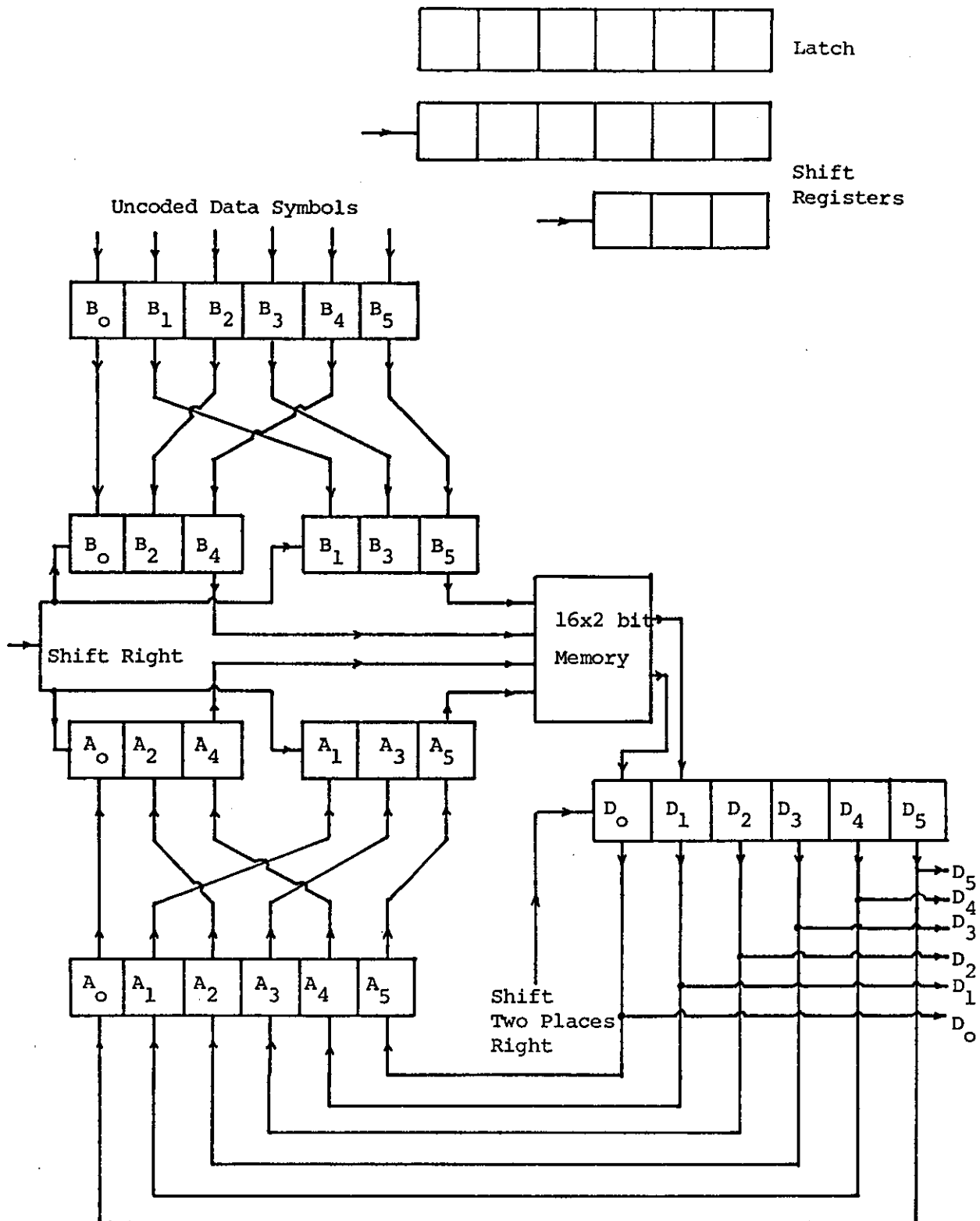


Figure 4.4.10: Possible Hardware Implementation of Coding Scheme 2

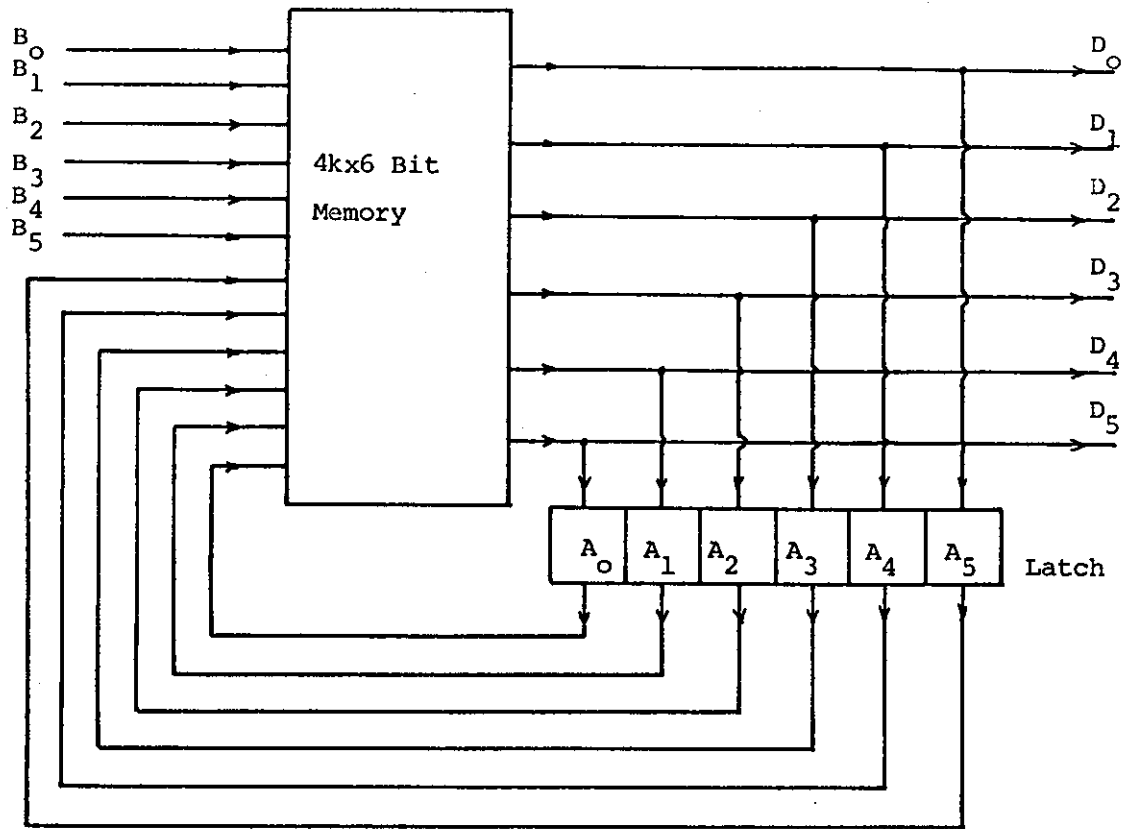


Figure 4.4.11: Simplified Coder Implementation for Coding Scheme 2

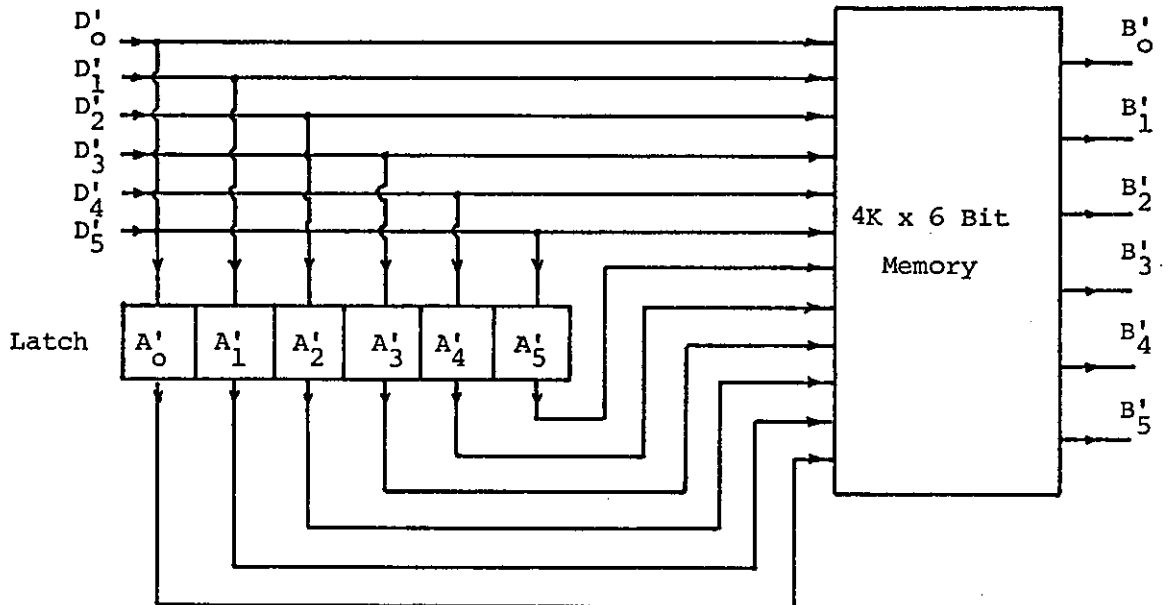


Figure 4.4.12: Decoder Implementation for Coding Scheme 2

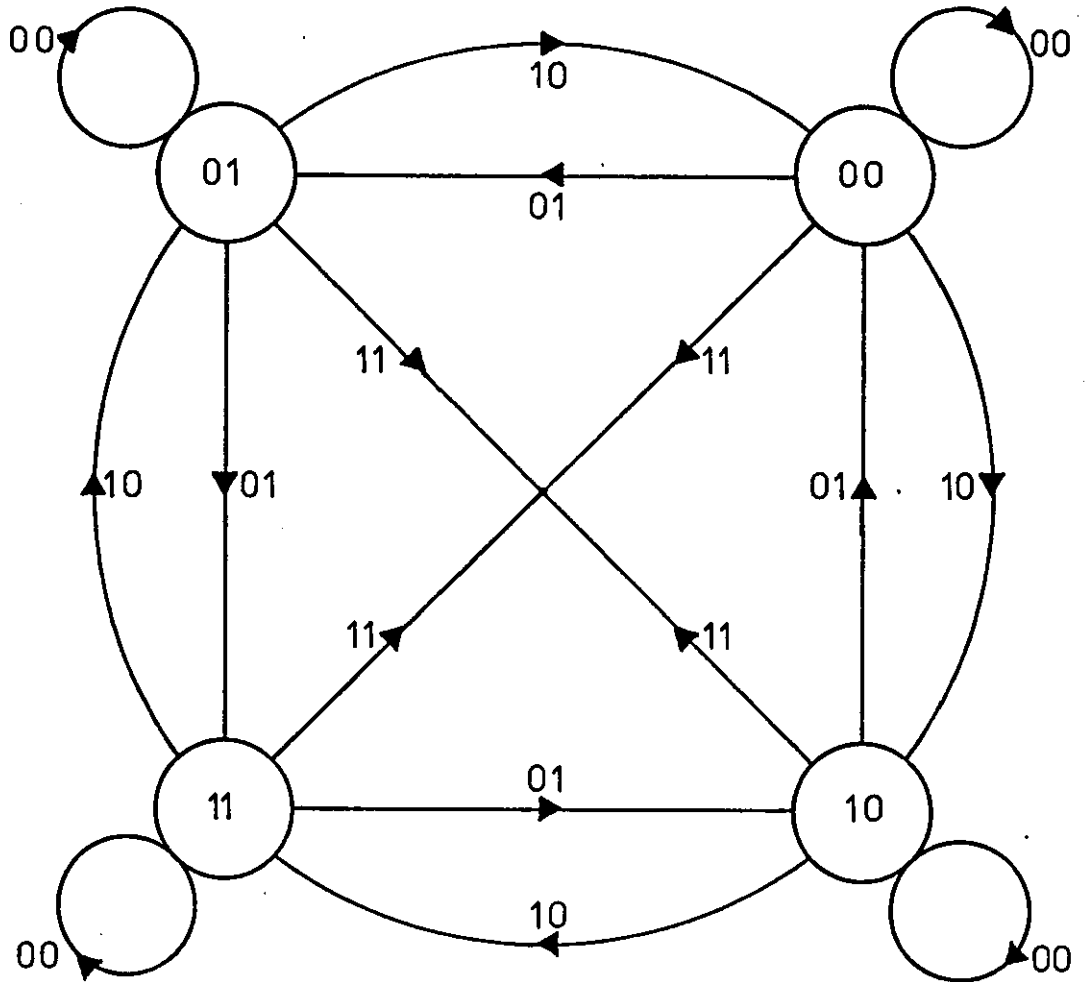


Figure 4.4.13: Coding and Decoding Diagram Applicable to Coding Schemes 1-4

New Uncoded Symbol	Previously Transmitted Encoded Symbol	New Coded Symbol For Transmission
		$s_{d,i-3}', 11 11 11$
$s_{i-2}', 00 01 11$	$s_{d,i-3}', 11 11 11$	$s_{d,i-2}', 11 10 00$
$s_{i-1}', 10 01 00$	$s_{d,i-2}', 11 10 00$	$s_{d,i-1}', 01 00 00$
$s_i', 11 01 01$	$s_{d,i-1}', 01 00 00$	$s_{d,i}', 10 01 01$

a) Coder Operation

New Detected Encoded Symbol	Previously Detected Encoded Symbol	Resultant Data Symbol
$s_{d,i-2}', 11 10 00$	$s_{d,i-3}', 11 11 11$	$s_{i-2}', 00 01 11$
$s_{d,i-1}', 01 00 00$	$s_{d,i-2}', 11 10 00$	$s_{i-1}', 10 01 00$
$s_{d,i}', 10 01 01$	$s_{d,i-1}', 01 00 00$	$s_i', 11 01 01$

b) Decoder Operation

Figure 4.4.14: Coder and Decoder Operations for Coding Scheme 2,
Using the Coding Chart of Fig. 4.3.13

New Detected Encoded Symbol Rotated by 90°	Previously Detected Encoded Symbol Rotated by 90°	Resultant Decoded Data Symbol
$s''_{d,i-2}, 10 00 01$	$s''_{d,i-3}, 10 10 10$	$s''_{i-2}, 00 01 11$
$s''_{d,i-1}, 11 01 01$	$s''_{d,i-2}, 10 00 01$	$s''_{i-1}, 10 01 00$
$s''_{d,i}, 00 11 11$	$s''_{d,i-1}, 11 01 01$	$s''_i, 11 01 01$

a) Rotation by 90°

New Detected Encoded Symbol Rotated by 180°	Previously Detected Encoded Symbol Rotated by 180°	Resultant Decoded Data Symbol
$s''_{d,i-2}, 00 01 11$	$s''_{d,i-3}, 00 00 00$	$s''_{i-2}, 00 01 11$
$s''_{d,i-1}, 10 11 11$	$s''_{d,i-2}, 00 01 11$	$s''_{i-1}, 10 01 00$
$s''_{d,i}, 01 10 10$	$s''_{d,i-1}, 10 11 11$	$s''_i, 11 01 01$

b) Rotation by 180°

New Detected Encoded Symbol Rotated by 270°	Previously Detected Encoded Symbol Rotated by 270°	Resultant Decoded Data Symbol
$s''_{d,b-2}, 01 11 10$	$s''_{d,i-3}, 01 01 01$	$s''_{i-2}, 00 01 11$
$s''_{d,b-1}, 00 10 10$	$s''_{d,i-2}, 01 11 10$	$s''_{i-1}, 10 01 00$
$s''_{d,i}, 11 00 00$	$s''_{d,i-1}, 00 10 10$	$s''_i, 11 01 01$

c) Rotation by 270°

Figure 4.4.15: Operation of Decoder for Coding Scheme 2 in the Presence of a) 90° , b) 180° , c) 270° Phase Rotations

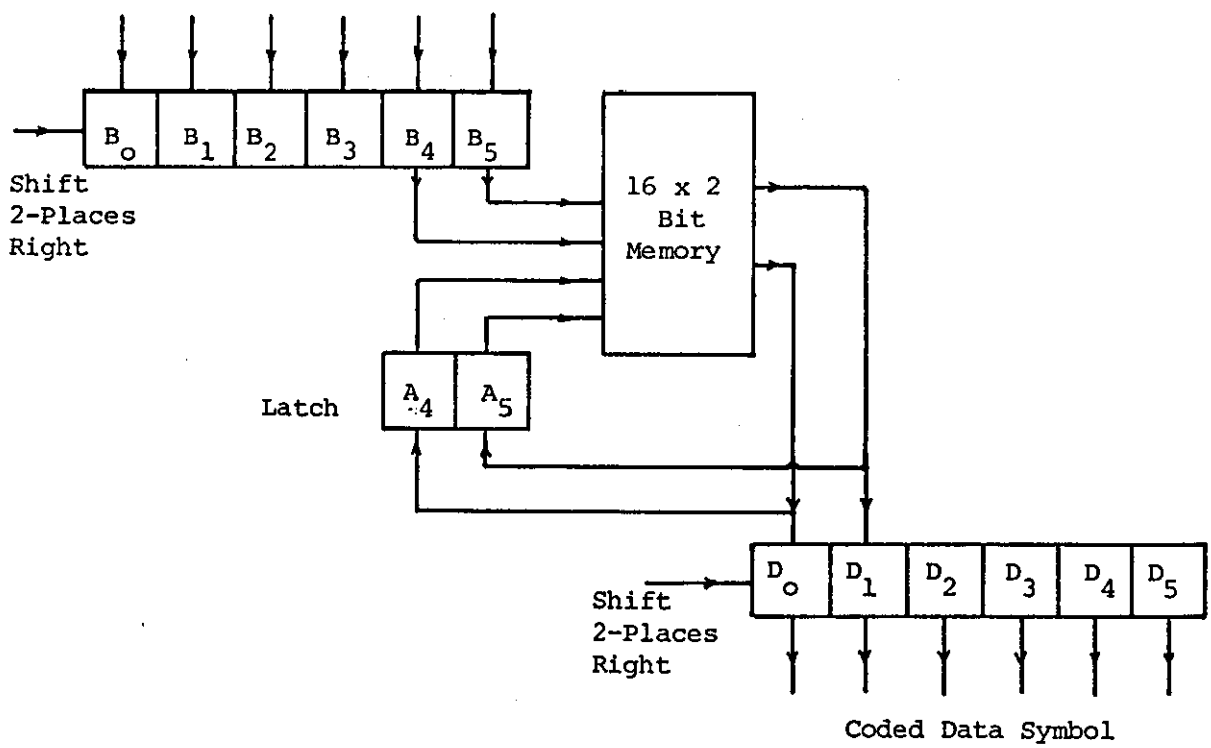


Figure 4.4.16: Coder Implementation for Alternative Coding Method for Coding Scheme 2.

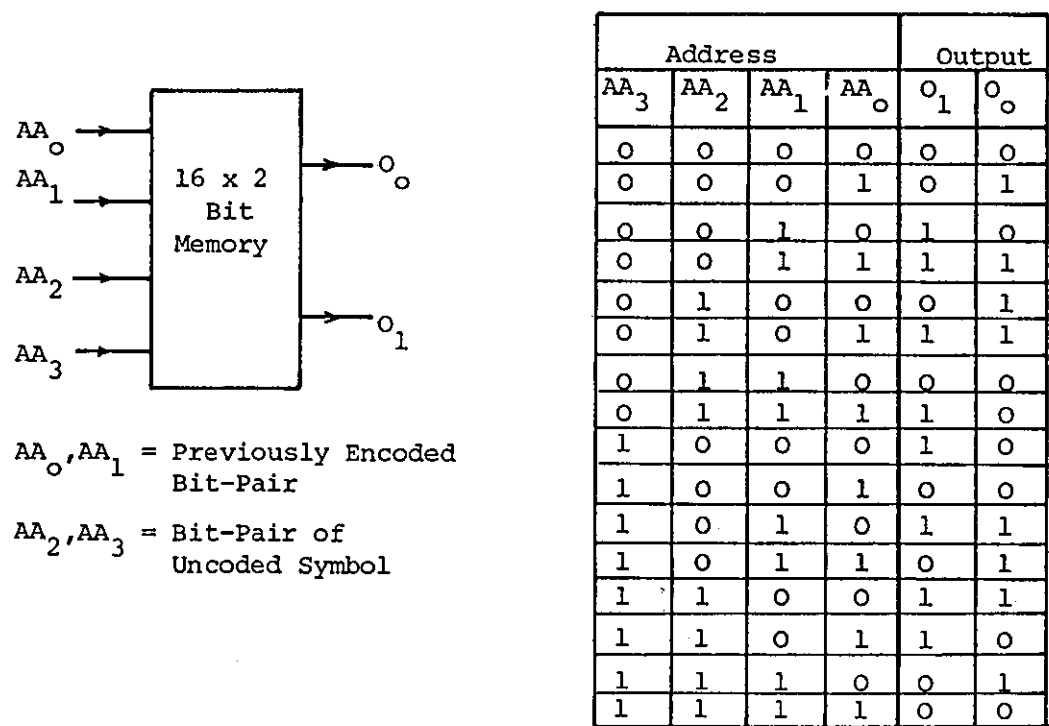


Figure 4.4.17: Contents of 16 x 2 Bit Memories Used in Coder Implementation

New uncoded symbol	Previously transmitted coded bit pair	Resultant coded bit-pair
$s_{i-2}, 00 01 11$		
11	11	00
01	00	01
00	01	01
		$s_{d,i-2}, 01 01 00$
$s_{i-1}, 10 01 00$		
00	01	01
01	01	11
10	11	01
		$s_{d,i-1}, 01 11 01$
$s_i, 11 01 01$		
01	01	11
01	11	10
11	10	01
		$s_{d,i}, 01 10 11$

a) Coder Operation

New detected coded symbol	Previously detected coded bit pair	Resultant decoded bit pair and data symbols
$s'_{d,i-2}, 01 01 00$		
00	11	11
01	00	01
01	01	00
		$s'_{i-2}, 00 01 11$
$s'_{d,i-1}, 01 11 01$		
01	01	00
11	01	01
01	11	10
		$s'_{i-1}, 10 01 00$
$s'_{d,i}, 01 10 11$		
11	01	01
10	11	01
01	10	11
		$s'_i, 11 01 01$

b) Decoder Operation

Figure 4.4.18: Modified coder and decoder operation for
Coding Scheme 2 using the Coding Chart of Figure 4.4.13

New detected coded symbol (Rotated by 270°)	Previously detected coded bit pair (Rotated by 270°)	Resultant decoded bit pair and data symbols
$s''_{d,i-2}$ 00 00 10		
10	01	11
00	10	01
00	00	00
		$s''_{i-2,00 01 11}$
$s''_{d,i-1}$ 00 01 00		
00	00	00
01	00	01
00	01	10
		$s''_{i-1,10 01 00}$
$s''_{d,i}$ 00 11 01		
01	00	01
11	01	01
00	11	11
		$s''_i, 11 01 01$

Figure 4.4.19: Modified decoder operation for Coding Scheme 2
in presence of 270° phase rotation

				s_{bi}					
011010	011110	010110	010010		001000	001001	001011	001010	
27	31	23	19	7	9	10	12	11	
011011	011111	010111	010011		001100	001101	001111	001110	
28	32	24	20	5	13	14	16	15	
011001	011101	010101	010001		000100	000101	000111	000110	
26	30	22	18	3	5	6	8	7	
011000	011100	010100	010000		000000	000001	000011	000010	
25	29	21	17	1	1	2	4	3	
-7	-5	-3	-1		1	3	5	7	s_{ai}
110010	110011	110001	110000		100000	100100	101100	101000	
51	52	50	49	-1	33	37	45	41	
110110	110111	110101	110100		100001	100101	101101	101001	
55	56	54	53	-3	34	38	46	42	
111110	111111	111101	111100		100011	100111	101111	101011	
63	64	62	61	-5	36	40	48	44	
111010	111011	111001	111000		100010	100110	101110	101010	
59	60	58	57	-7	35	39	47	43	

Figure 4.4.20: Signal constellation for Coding Scheme 3;
64-point structure

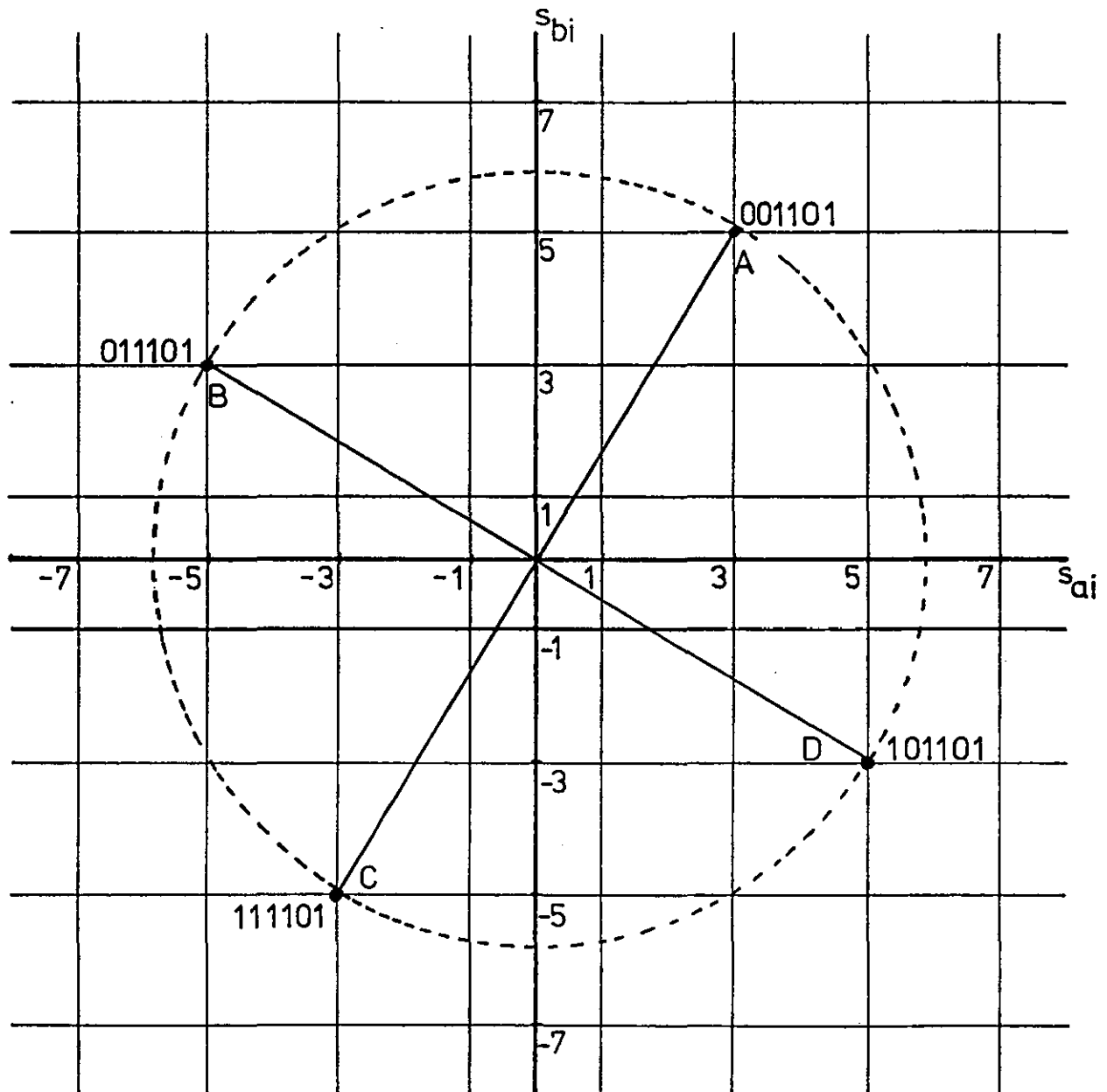
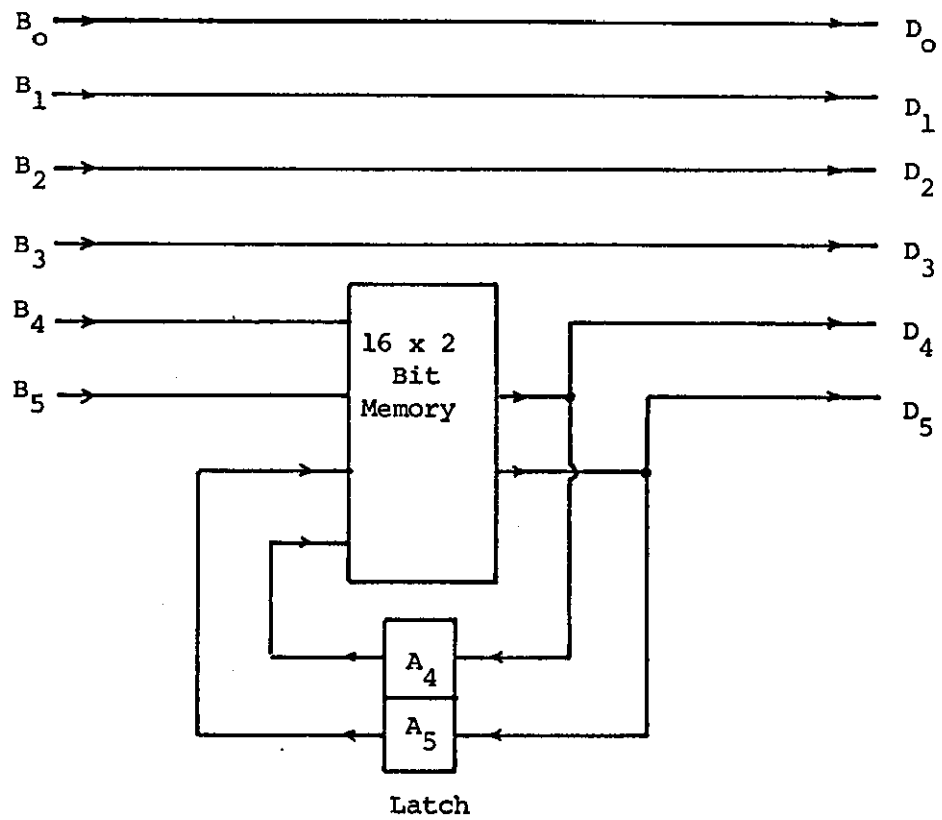
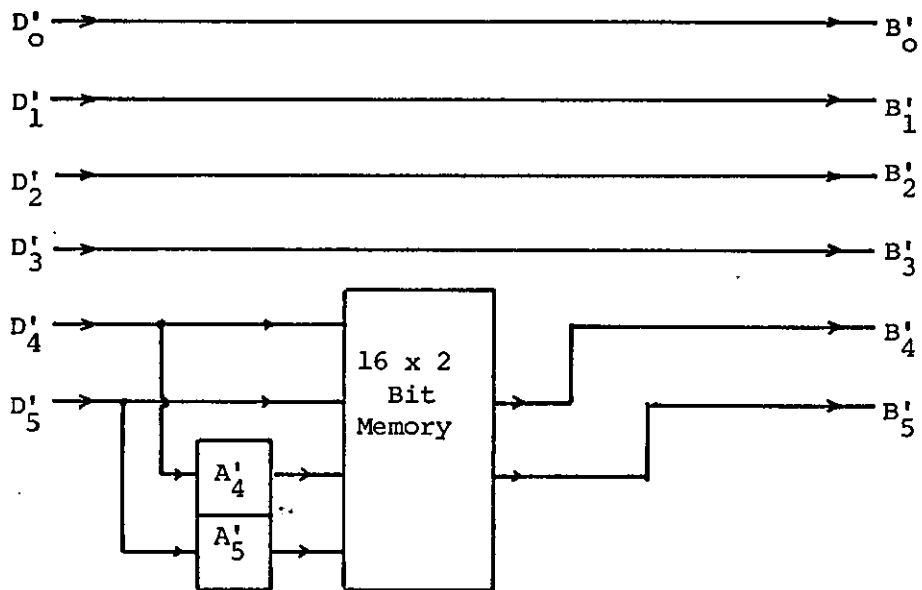


Figure 4.4.21: Illustration of rotational consistency of the four positional bits in Coding Scheme 3



a) Coder



b) Decoder

Figure 4.4.22: Coder and decoder implementation for Coding Scheme 3

5 Equipment Filters and Overall Baseband Characteristics

5.1 Introduction

It has been shown in Chapters 2-4 that the transmission of 19,200 bit/s over bandlimited telephone channels may be modelled by an equivalent baseband system. Furthermore, it has been shown that the baseband model reduces to a discrete-time model which simplifies the mathematical description of the system and also aids in translating the basic model into suitable program code for simulating the system on a digital computer. In order to simulate the operation of what is in reality, a bandpass analogue system, the characteristics of all bandpass components within the system, must be referred to baseband as explained in Appendix 1. The combination of the original baseband processes and the referred bandpass processes result in the overall equivalent baseband characteristics of the digital data transmission system. The impulse response of the baseband characteristic may then be sampled to determine the sampled impulse response of the overall linear baseband channel, which is used to generate the distorted and noisy received samples according to;

$$r_i = \sum_h \sum_s s_i y_{i-h} + u_i \quad (5.1.1)$$

where r_i = The i^{th} received sample
 s_i = The i^{th} transmitted symbol
 y_i = The $(i+1)^{\text{th}}$ sample of the sampled impulse response of the baseband channel
 u_i = The i^{th} sample value of a Gaussian random variable with a zero mean value and variance $2\sigma^2$.

In general, the r_i , s_i , y_i and u_i will all be complex-valued. It is also assumed that the $(g+1)$ -component vector,

$$Y = [y_0 \quad y_1 \quad \dots \quad y_g] \quad (5.1.2)$$

corresponds to a finite length impulse response such that the sampled response has $(g+1)$ components; any constant delay in transmission is neglected and so $y_j \neq 0$ for $(g+1) > j > 0$. Furthermore, the impulse response of a particular baseband channel is assumed to be time invariant or at worst, varies very slowly over a large number of symbol intervals.

This chapter is concerned with two important areas in modem design. The first is the design of practical "equipment filters" which is a term used to describe all of the filters required in the system. The second area is the selection of suitable "typical" telephone line characteristics and the determination of the overall baseband equivalent sampled impulse responses, which include the practically realisable equipment filters; this enables the transmission elements of the system to be modelled and simulated using equation (5.1.1). The performance of the data transmission system when using different detection techniques may then be determined by allowing the modelled detectors to operate on the received samples, $\{r_i\}$ and comparing the detected symbols (and the information bits contained therein) with the transmitted symbols when operating over the selected channels with various signal-to-noise ratios, as described in Chapter 6.

5.2 General Requirements of the Equipment Filters

The model of the QAM system introduced in Chapter 4 is reproduced here as Figure 5.2.1 and the equivalent baseband model as Figure 5.2.2. The transmitter filter in Figure 5.2.2 is the baseband equivalent of the combination of the pre-modulation filter $(g(t), G(f))$ and the channel input filter $(q(t), Q(f))$ shown in Figure 5.2.1 and is assumed to be described by $(a(t), A(f))$. Similarly, the baseband equivalent of the combination of the post-demodulation filters $(p(t), P(f))$ and the channel output filter $(c(t), C(f))$ is designated the receiver filter $(b(t), B(f))$ in Figure 5.2.2. The baseband equivalent of the bandpass transmission path $(h(t), H(f))$ is represented by $(\hat{h}(t), \hat{H}(f))$. The pre-modulation filters and the channel input filters must jointly ensure that the transmitted signal satisfies any transmission restrictions imposed by the channel operator; Chapter 3 gives details of the restrictions for U.K. British Telecom. circuits. In practice, the channel input filter will usually have a wider bandwidth than that of the pre-modulation filters and so these latter filters will be the major contributors to the shaping of the transmitted signal spectrum. The spectrum of the received signal at the output of the transmission path will, of course, be a modified version of the transmitted spectrum due to the non-ideal characteristics of the channel itself and the introduction of other signal impairments. Ideally, the bandwidth of the pre-modulation lowpass filters should be wide enough to allow the system to operate at the Nyquist rate, as determined by the symbol transmission rate, which will ensure that no signal distortion and hence no intersymbol interference occurs due to the filtering process alone. This requirement, coupled with the need to bandlimit the spectrum of the transmitted QAM signal, implies the use of an ideal Nyquist shaping lowpass filter characteristic as suggested in Section 4.3. In practice, the need to conform

with spectral restrictions and the practical problems of designing realisable filters will result in the bandwidth of the combined transmitted filter being slightly less than that required to satisfy Nyquist's criterion^(1,2).

The channel output filter and the post-demodulation filters must remove any unwanted received spectral components and any frequencies generated by the demodulation process itself, without however, distorting the signal component of the received waveform. These filters must also remove any mains induced components which may be present on the received signal. In practice, the channel output filter alone must reject frequency components greater than the highest signal frequency and components lower than say 250 Hz, the fifth harmonic of the U.K. power frequency. The post-demodulation filters are usually designed to have a narrower bandwidth than that of the channel output filter and so the former filters will contribute most to the spectral shaping of the received baseband signal. These filters are also normally chosen to be approximately equal to the transmitter pre-modulation filters so that for the case of filters with purely real, symmetrical impulse responses, the receiver filter will be closely matched to the signal at the output of the transmitter filter. Hence, when the transmission path introduces no signal distortion, the signal-to-noise ratio at the output of the receiver filter will be maximised⁽¹⁻⁴⁾. Apart from these requirements of the individual filters, it is most important for the impulse response of the combination of all filters in the signal path to have a rapid rise time to its peak value and to be of limited duration. These latter conditions imply a sampled impulse response with relatively few non-zero components and whose largest component appears as close to the start of the sequence as possible. This requirement can be achieved approximately in practice by designing the pre-modulation and post-demodulation filters to be minimum phase processes^(2,5-8).

Before presenting a general strategy for the design of practical filters for use in a 19,200 bit/s modem, it is first necessary to decide on the actual method of filter implementation, that is, whether to use analogue or digital techniques. Clearly, the digital data signal has to interface with a bandlimited analogue transmission path which is governed by spectral restrictions as mentioned earlier, so the channel input filter must be analogue in form regardless of the format of the signal prior to this point in the system. This follows since digital filters exhibit repetitive responses centred at multiples of the digital sampling rate. (It is important to realise that the digital sampling rate here is not the same as the sample rate or symbol rate of the data transmission system. The former term refers to the sample rate of the digital processing circuitry used within the system and will necessarily be much higher than the rate of transmitting data symbols to avoid the introduction of additional distortion caused through aliasing^(3,6).)

To avoid the undesirable effects of drift and imbalance which can occur when using analogue signal processing devices such as multipliers, it is advisable to perform the actual modulation and demodulation operations digitally. Since the input data itself is also in digital form, it is obviously convenient and in fact very cost effective when considering the modem as a commercial product, to implement the whole of the transmitter unit in digital form up to a digital-to-analogue converter located just before the analogue bandpass channel input filter. By a similar argument, the channel output filter will be analogue and will now have the further function of ensuring that the input signal spectrum to an analogue-to-digital converter located just after this filter is suitably limited to avoid aliasing. As the majority of the signal processing performed at the transmitter will now be digital, the channel input filter must also ensure

that frequency components at the digital sampling rate are not passed to the transmission line. In practice, this does not necessitate stricter filtering as the digital sampling frequency will be much higher than the bandwidth of the bandpass filter itself.

The general strategy for the design of the equipment filters is as follows. The lowpass pre-modulation filters and post-demodulation filters are designed to exhibit the desired responses in both the time and frequency domains, as discussed earlier. Having selected the lowpass filters, the analogue bandpass filters can be designed to ensure that the resultant of all transmitter filtering restricts the spectrum of the transmitted signal to that permitted by the circuit's operational limits. At the receiver, the bandwidth of the bandpass filter must prevent all out-of-band frequencies from entering an analogue-to-digital converter, whilst allowing the signal itself, in the absence of channel distortion, to pass without the introduction of additional distortion. In practice, the channel input bandpass filter shown in Figure 5.2.1 is replaced by a lowpass analogue filter which suppresses any high order frequency components; the highpass section of the overall bandpass characteristic required for transmission being obtained from the attenuation characteristic of the pre-modulation lowpass filter which will be translated after modulation and will therefore be symmetrical about the carrier frequency. Clearly, the frequency domain characteristics of the digital lowpass filter must now also ensure that the spectrum of the transmitted signal complies with the channel's low-frequency restrictions given in Figures 3.4.6 and 3.4.7. Figure 5.2.3 shows the spectra of signals at various points in the system and illustrates the actions of the filtering processes.

Before considering the actual design of the digital and analogue filters, it is necessary here to anticipate a requirement of some of the

detection processes considered in detail in Chapter 6. It was mentioned in Section 2.2 that three main types of detector are generally used in high-speed digital data transmission systems, namely linear equalisation, non-linear equalisation and maximum-likelihood detection. It will be shown in Chapter 6 that the inclusion of an adaptive linear transversal filter acting as a whitened matched filter between the sampler and detector will, when suitably adjusted, remove all phase distortion introduced by the overall channel including the equipment filters and then adjusts the sampled impulse response of the overall channel plus the linear filter to a form which is ideally suited to near-maximum-likelihood detectors. It will also be shown that the adaptive linear filter is identical to the first part of the optimum non-linear detector. As far as this section is concerned, the adaptive linear filter performs a process of pure phase transformation and so the overall sampled impulse response of the channel, equipment filters and the adaptive filter itself will have a fast rise time and be of limited duration^(2,5,9-12). The adaptive filter, also called the pre-detection filter or simply the pre-filter, replaces the roots (zeroes) of the z-transform of the sampled impulse response of the overall linear baseband channel which lie outside the unit circle in the z-plane, with the complex conjugates of their reciprocals. Since all of the zeroes will now lie within the unit circle, the resulting sampled impulse response will therefore be representative of a minimum phase system⁽⁶⁾. Theoretically, this process is achieved without introducing any gain or attenuation and does not equalise any amplitude distortion. Consequently, the statistics of the noise samples at the output of the pre-filter remain the same as at its input so the signal-to-noise ratio, defined in Appendix 3, remains unchanged. Furthermore, some of the detection processes considered in Chapter 6 may be

greatly simplified by arranging for the first non-zero component of the minimum phase sampled impulse response to be adjusted to the value $1+j0$. Clearly, this adjustment will require all components of the sampled impulse response to be scaled by the reciprocal of the first component which in general will introduce a constant shift of all of the components of the sampled impulse responses discrete energy density spectrum. When simulating digital data transmission systems using detection processes which include a pre-filter, it is convenient to assume that the pre-filter has been suitably adjusted and that the overall sampled impulse response is therefore minimum phase. The simulation program therefore operates with the minimum phase versions of the overall sampled impulse responses which, for reasons of run-time economy, are calculated separately from the main program. The model shown in Figure 5.2.2 will now once more apply but with the proviso that the pre-filter is also included in the linear baseband channel. However, if the models shown in Figures 4.3.8, 4.3.9 or 4.3.10 are used as a basis for the simulations, the noise, after passing through a filter identical to the combined baseband equivalent receiver filter, is added after the signal has passed through the linear baseband channel which now includes the pre-filter. Clearly, to maintain the signal-to-noise ratio, the noise must also be scaled by a value identical to that introduced into the sampled impulse response when adjusting the first component to be $1+j0$. Alternatively, the tolerance of the system to additive noise must be degraded by $10\log_{10}(\sum_i |q_i|^2) - 10\log_{10}(\sum_i |y_i|^2)$ dB where the $\{q_i\}$ are the sample values of the minimum phase sampled impulse response and the $\{y_i\}$ are the samples of the unequalised or "raw" sampled impulse response.

5.3 Equipment Filter Design

To enable the performance of the final modem design to be studied when operating over a variety of linear baseband channels, two complete sets of equipment filters were investigated. The first set, known here as filter set 1, were designed by M J Fairfield and originally featured in the design of a 9600 bit/s modem which used a 16-point QAM structure^(7,8,11). Other research workers at Loughborough University have successfully used this filter set for a variety of different applications, including feasibility studies for the transmission of 19,200 bit/s data over telephone lines and the serial transmission of data at 2400 bit/s over fading HF radio links^(2,4,9,10). The design of individual filters constituting filter set 1 has been considered in detail elsewhere^(7,8) so only the characteristics and responses are given in subsequent sections.

The second set of filters, filter set 2, have been designed specifically for application to a 19,200 bit/s modem when operating with the 64-point structure at a symbol rate of 3200 bauds. Details of the procedure used in the design of the digital low-pass filters in filter set 2 are given in Appendix 6.

In both filter sets, the pre-modulation and post-demodulation low-pass filters are both digital whereas the channel input and output filters are both analogue in form, although these latter filters in filter set 2 are actually a hybrid of both analogue and digital techniques and are a form of switched-capacitor filter⁽¹⁵⁻¹⁷⁾. References 18-22 give details of currently available switched-capacitor filter integrated circuits, which, under the conditions given in Appendix 7, may be suitable for use in high-speed modems.

5.3.1 The Digital Low-Pass Filters

Figures 5.3.1 and 5.3.2 show the attenuation and group delay characteristics of the digital low-pass filters at the transmitter and receiver respectively, for filter set 1. These filters will henceforth be designated filters 1b and 1d and are identical to the filters $(g(t), G(f))$ and $(p(t), P(f))$ respectively, in Figure 5.2.1.

Figures 5.3.3 and 5.3.4 give the filter characteristics after frequency translation by 1800 Hz. Comparison between the attenuation characteristics of filters 1b and 1d shows that the receiver low-pass filter (1d), has a slightly wider bandwidth than the transmitter filter (1b). This design feature was included to minimise noise correlation at the receiver when using the original 9600 bit/s, 2400 bauds system⁽⁸⁾.

Whilst filter set 1 has been used quite successfully for the transmission of data symbols at rates in excess of 2400 bauds, the low-pass filters 1b and 1d will clearly introduce greater amounts of signal distortion as the symbol rate is increased, due to the resulting sampled impulse response having a larger number of non-zero components. It has been suggested by Najdi that an increase of around 2 dB in system performance may be possible when operating over telephone circuits conforming to schedule D specifications when the bandwidth of the equipment filters is somewhat wider than that of filter set 1^(9,10). To test this hypothesis, a new set of filters, filter set 2, has been designed which aim to minimise the amount of distortion introduced by the filtering processes themselves whilst being consistent with channel input restrictions. Appendix 6 outlines the procedure followed in the design of the digital low-pass filters used in filter set 2, these filters being designated filter 2b and 2d and correspond to the pre-modulation and post-demodulation filters, respectively. Figure 5.3.5 shows diagrammatically the attenuation and group delay

characteristics of these filters where in fact the filters exhibit identical characteristics; the translated characteristics are given in Figure 5.3.6. Comparison between Figures 5.3.1 and 5.3.5 shows that filter 2b has a -3 dB bandwidth approximately 200 Hz wider than filter 1b and so the combination of filters 2b and 2d after modulation will exhibit a characteristic which is approximately 200 Hz wider (at the -6 dB frequencies) than that of the translated combination of filters 1b and 1d at both the low and high frequency limits of the characteristic.

5.3.2 The Analogue Low-Pass and Bandpass Filters

Since both filters 1b and 2b restrict the signal bandwidth to that permitted by British Telecom, the channel input analogue filter need only exhibit a low-pass characteristic with a cut-off frequency lying outside the signal bandwidth. Also, because the pre-modulation filters are digital, the channel input filter must provide sufficient attenuation to frequencies within the region of the digital sampling frequency.

The channel input filter for filter set 1, designated filter 1a, is a 5th order Butterworth filter having a cut-off frequency at 3000 Hz⁽⁸⁾. The attenuation and group delay characteristics of this filter are given in Figure 5.3.7. The channel output filter, designated filter 1c for filter set 1, uses an identical low-pass section as filter 1a, but with an additional 3rd order Butterworth high-pass filter with a cut-off frequency of 600 Hz to provide attenuation at mains frequencies and closely related harmonics. Figure 5.3.8 shows the attenuation and group delay characteristics of the resultant bandpass filter.

Due to the increased bandwidth of digital filter 2b compared with filter 1b, the analogue filters characterised in Figures 5.3.7 and 5.3.8 will not be suitable for inclusion in filter set 2. As an alternative to

redesigning the Butterworth filters, which for practical realisation would require component tolerances of around 2-3% if using discrete technology⁽¹⁵⁾ or the use of some form of hybrid technology, which in general will increase the cost of the filters⁽¹⁶⁾, single-chip filters operating on the switched-capacitor principle were considered. These filters offer several advantages over conventional filters, the most important being their low cost, compactness and the easy repeatability of filter design. Also, several commercially available integrated circuits feature two independent filters within a single package allowing both analogue filters to be implemented very simply and economically⁽¹⁵⁻²²⁾. Appendix 7 gives the results of a comparative study between two such filter circuits, the Mitel MT8912 and Intel I2912 devices, the results of which suggest the use of the Mitel device for the current application^(18,19,23).

Figures 5.3.9 and 5.3.10 give the attenuation and group delay characteristics of the two filters contained in the Mitel MT8912 integrated circuit which were obtained experimentally, as explained in Appendix 7. Henceforth these filters will be designated filter 2a and filter 2b, the designations applying to the transmitter low-pass and receiver bandpass filters respectively.

5.4 Characteristics of Combined Transmitter and Receiver Filters

The combination of filters $(g(t), G(f))$ and $(q(t), Q(f))$ in Figure 5.2.1 when suitably translated in frequency to allow for the modulation process, give either the bandpass or low-pass equivalents of the filtering operations at the transmitter; the equivalent low-pass filter is designated $(a(t), A(f))$ in Figure 5.2.2. Similarly, the combination of $(c(t), C(f))$ and $(p(t), P(f))$ in Figure 5.2.1, again with suitable frequency translation applied, yield the combined bandpass or low-pass equivalent filter,

this filter being designated $(b(t), B(f))$ in Figure 5.2.2. Since the combined transmitter and receiver filters are themselves combined with the bandpass characteristics of three different telephone circuits in the following section, it is more convenient here to show the overall transmitter and receiver filters as bandpass processes. Figure 5.4.1 gives the attenuation characteristics of the combination of filters 1a and 1b, this combination being designated filter 1e. Figure 5.4.2 shows the corresponding characteristic for filter 1f, which represents the bandpass equivalent process of filters 1c and 1d combined together. Similar combinations of filter 2a with 2b and filter 2c with 2d are shown in Figures 5.4.3 and 5.4.4, these combinations being designated filter 2e and 2f respectively. Finally, the totality of all filtering in filter sets 1 and 2, known here as filters 1g and 2g respectively, are shown in Figures 5.4.5 and 5.4.6; the attenuation and group delay characteristics when sampled at frequency intervals of 50 Hz are given in Table 5.4.1.

5.5 Characteristics of the Three Test Telephone Circuits

In order to evaluate the performance of the 64-point and 256-point QAM systems when operating with various combinations of equipment filters and detection schemes, three different test telephone circuits were selected. The attenuation and group delay characteristics for these channels, known here as telephone circuits 1 to 3, are shown in Figures 5.5.1, 5.5.3 and 5.5.5 respectively. Telephone channels 1 and 2 introduce typical levels of attenuation and delay distortion whereas telephone channel 3 is close to the typical worst case circuit normally considered for the transmission of data at 600-1200 bit/s over the public switched telephone network⁽⁹⁻¹¹⁾; circuit 3 introduces severe group delay distortion as well as considerable attenuation distortion^(1,5,11). Figures 5.5.2, 5.5.4 and

and 5.5.6 show the impulse responses of the three test circuits from which an idea of the distortion introduced by each channel can be gained. For completeness, Table 5.5.1 lists the attenuation and group delay values for telephone channels 1 to 3 when sampled at frequency intervals of 50 Hz. To permit direct comparison between data transmission systems operating over different test circuits, the minimum attenuation and minimum group delay for each of circuits 1 to 3 have been adjusted to occur at 0 dB and 0 ms respectively; the characteristics shown in Figures 5.5.1 to 5.5.6 are therefore relative rather than absolute characteristics.

5.6 Sampled Impulse Responses of the Overall Linear Baseband Channels

The sampled impulse response of the complete linear baseband channel shown in Figure 5.2.2 may be obtained by convolving together the individual impulse responses of the transmitter filter, transmission path and receiver filter to give the continuous impulse response, $y(t)$, followed by a sampling operation to determine the discrete sequence $\{y_i\}$. Alternatively, the sampled impulses of the individual baseband referred processes shown in Figure 5.2.2 may be obtained at the required sample rate which must be greater than or equal to the Nyquist rate and then convolved and suitably scaled to give the overall sampled impulse response of the overall baseband channel as shown in Appendix 8. The latter approach is adopted here to evaluate the sampled impulse responses of the equivalent baseband channels using combinations of filter sets 1 and 2 with telephone channels 1 to 3.

The calculation of the sampled impulse responses of the filters and channels has been simplified by the use of two computer programs, listed as Programs 3 and 4 in Appendix 10. Program 3 is a development of a general channel analysis program originally written by members of the Communication

Research Group at Loughborough University. The program calculates the sampled impulse response, at any desired sampling rate and sample phase, of the combination of two bandpass processes defined at the program's input level in terms of their attenuation and group delay characteristics sampled at 50 Hz intervals. In the application considered here, only a single bandpass process (a filter or telephone circuit) is applied to the program for reasons which will become clear later.

Since the calculated sampled impulse responses will be used in simulation programs to assess the relative performances of different systems, it is clearly essential to have absolute confidence in their values and therefore in the operation of Program 3, which is summarised below.

Let $A_{tt}(f)$ and $GD(f)$ be the attenuation and group delay respectively of a bandpass process at a frequency, f . $A_{tt}(f)$ and $GD(f)$ are related to the complex transfer function of the process by;

$$\begin{aligned} A_{tt}(f) &= -20 \log_{10} |A(f)| \\ GD(f) &= -\frac{1}{2\pi} \frac{d\phi(f)}{df} \end{aligned} \quad (5.6.1)$$

where $A(f)$ is the complex transfer function and

$$A(f) = |A(f)| e^{j\phi(f)} \quad (5.6.2)$$

where $|A(f)|$ is the absolute value of $A(f)$ and

$$\phi(f) = \arg(A(f)) = \tan^{-1} \left\{ \frac{\text{Im}(A(f))}{\text{Re}(A(f))} \right\} \quad (5.6.3)$$

$\text{Im}(\cdot)$ and $\text{Re}(\cdot)$ being the imaginary and real values of (\cdot) respectively.

The input to the program consists of $A_{tt}(f)$ and $GD(f)$ for values of f between 50 and 3750 Hz, at intervals of 50 Hz. The filter characteristics are then referred to baseband by translating $A_{tt}(f)$ and $GD(f)$ down in frequency by an amount equal to the systems carrier frequency, f_c , to give,

$$A_{\text{ttref}}(f) = A_{\text{tt}}(f + f_c)$$

$$GD_{\text{ref}}(f) = GD(f + f_c) \quad (5.6.4)$$

where f_c is the carrier frequency.

From equations (5.6.1) to (5.6.3), the complex transfer function of the referred process will be:

$$A(f) = 10^{-A_{\text{ttref}}(f)/20} e^{j(-2\pi \int GD_{\text{ref}}(f) df)}, \quad |f| \leq f_c$$

$$= 0, \quad |f| > f_c \quad (5.6.5)$$

The real and imaginary components of $A(f)$ are then calculated according to equation (5.6.5) where the integration required for the phase calculation is performed numerically; $A(f)$ itself will of course occur at 50 Hz intervals. An inverse discrete Fourier Transform routine then operates on $A(f)$ to give the sampled impulse response of the filter or telephone channel. If Δf represents the sampling interval in the frequency domain and T is the sampling interval in the time domain, then;

$$\Delta f = \frac{1}{NT} \quad (5.6.6)$$

where N is the number of samples in the inverse discrete Fourier Transform. To ensure that time domain aliasing does not occur for processes exhibiting relatively long impulse responses, Δf must be a sufficiently small value. In practice, for the filters and channels considered here a value of $\Delta f = 25$ Hz has proved sufficient. Consequently, prior to the calculation of $A(f)$ in equation (5.6.5) both $A_{\text{ttref}}(f)$ and $GD_{\text{ref}}(f)$ are operated upon to give samples at 25 Hz spacing, the sample values at frequencies between integer multiples of 50 Hz being obtained by interpolation whilst the samples at the band edges are calculated using linear prediction. Furthermore, to ensure that the components of the

sampled impulse response are separated by a sufficiently small interval, the frequency range of the frequency domain signal is increased to 25600 Hz by assuming zero-valued samples outside the actual known frequency range. At 25 Hz spacing, this results in 1024 samples and so the inverse discrete Fourier Transform routine returns 1024 samples covering 40 ms of impulse response⁽²⁴⁾. Finally, the sampled impulse response of the process at the required rates (3200 and 2400 samples per second) are obtained by suitable selection from the finely sampled output of the inverse transform routine.

X The above description has assumed that the sample values of the sampled impulse response are representative samples of the continuous impulse response. In practice, different inverse discrete Fourier Transform routines introduce different scaling factors into the sample values. The routine used in Program 3 introduces a scaling factor equal to $\frac{1}{\sqrt{N}\Delta f}$ where N is the number of samples and Δf is the frequency spacing. Consequently, the sample values at the output from the routine are multiplied by $\sqrt{N}\Delta f = \sqrt{1024} \times 25 = 800$ to give samples representative of the continuous impulse response. Finally, the operation of the complete program was checked using the well known responses of "brickwall" and $\sin x/x$ characteristics.

X Program 3 was used to derive the sampled impulse responses at 2400 and 3200 samples per second for the baseband equivalents of filters 1e, 1f, 2e and 2f and telephone channels 1 to 3. During the calculation of the filter's sampled impulse responses, extra program code was included in the program to normalise the energy of the finely sampled sequences to unity, as follows.

If $\{a_j\}$ is the sequence representing sample values of the sampled impulse response of the transmitter filter shown as $(a(t), A(f))$ in

Figure 5.2.2, then the energy of this sequence is defined as,

$$E_a = \sum_{j=1}^{1024} \{(\operatorname{Re}(a_j))^2 + (\operatorname{Im}(a_j))^2\} \Delta t \quad (5.6.7)$$

where $\operatorname{Re}(a_j)$, $\operatorname{Im}(a_j)$ are the real and imaginary values of the finely spaced sampled impulse response and Δt is the time separation between samples.

Each sample a_j , $j = 1$ to 1024, is then scaled by $\frac{1}{\sqrt{E_a}}$ to yield a new sequence, $\{a_n\}$, which is normalised such that;

$$E_{na} = \sum_{j=1}^{1024} \{(\operatorname{Re}[a_n])^2 + (\operatorname{Im}[a_n])^2\} \Delta t = 1 \quad (5.6.8)$$

Since Δt is sufficiently small, then;

$$E_{na} \approx \int_{-\infty}^{\infty} a^2(t) dt \quad (5.6.9)$$

and so

$$\int_{-\infty}^{\infty} |a(t)|^2 dt = \int_{-\infty}^{\infty} |A(f)|^2 df \approx 1 \quad (5.6.10)$$

which is the condition required of the transmitter filter to simplify signal-to-noise ratio calculations, as discussed in Appendix 3. Under these conditions, the average energy per symbol at the input to the transmission path is $\overline{|s_i|^2}$, where the $\{s_i\}$ are the possible symbol values and where it is assumed that the $\{s_i\}$ are statistically independent. It should be clear from equation (5.6.10) that the shape of neither the attenuation characteristic nor the impulse response will be affected by the normalising process. However, as E_a in equation (5.6.7) will in general be less than unity, normalising will introduce some constant gain into the system. This does not, of course, change the signal-to-noise ratio as signal energy is defined in Appendix 3 as the average energy per bit at the input to the transmission path. A similar normalising operation is

performed on the receiver filter such that;

$$\int_{-\infty}^{\infty} |b(t)|^2 dt = \int_{-\infty}^{\infty} |B(f)|^2 df \approx 1 \quad (5.6.11)$$

where $(b(t), B(f))$ defines the receiver filter in Figure 5.2.2. Under these conditions, as shown in Appendix 3, the average noise power at the output of the receiver filter will be equal to the variance of the noise samples at its input, which again simplifies signal-to-noise ratio calculations.

It has been mentioned in Section 5.2 that the inclusion of a linear pre-filter just prior to the actual detector can give beneficial results when using certain types of detector. The pre-filter is adjusted to produce a process of pure phase transformation and ideally should not introduce any gain or attenuation on signals passing through it. When located between the sampler and detector in Figure 5.2.2, the pre-filter adjusts the sampled impulse response of the overall linear baseband channel to be minimum phase^(1,5,6). Let Y be a $(g+1)$ -component row vector whose components are the (complex) values of the sampled impulse response of the overall baseband channel,

$$Y = [y_0 \quad y_1 \quad y_2 \quad \dots \quad y_g] \quad (5.6.12)$$

where, from Figure 5.2.2,

$$y(t) = a(t) * \hat{h}(t) * b(t) \quad (5.6.13)$$

$$\text{and} \quad \{y_i\} = \{a_i\} * \{\hat{h}_i\} * \{b_i\} \quad (5.6.14)$$

where, $a(t)$ is the impulse response of the transmitter filter

$\hat{h}(t)$ is the impulse response of the transmission path and

$b(t)$ is the impulse response of the receiver filter.

Also, the $\{y_i\}$, $\{a_i\}$, $\{\hat{h}_i\}$ and $\{b_i\}$ are sample values of $y(t)$, $a(t)$, $\hat{h}(t)$ and $b(t)$, respectively, when sampled at times $t = iT$. The symbol $*$ denotes the joint operation of discrete-time convolution and scaling such that equation (5.6.14) is representative of equation (5.6.13).

The actual relationship between discrete-time and continuous-time convolution and details of the operator $*$ are given in Appendix 8.

The z -transform of Y is therefore,

$$Y(z) = y_0 + y_1 z^{-1} + y_2 z^{-2} + \dots + y_g z^{-g} \quad (5.6.15)$$

which is a polynomial in z^{-1} of order g . The roots (zeroes) of $Y(z)$ are defined as those values of z for which $Y(z) = 0$ and so equation (5.6.15) may be written as,

$$Y(z) = y_0 (1 - \alpha_1 z^{-1}) (1 - \alpha_2 z^{-1}) \dots (1 - \alpha_g z^{-1}) \quad (5.6.16)$$

where α_i , $i = 1, 2 \dots g$ are the g roots of $Y(z)$ and are in general complex-valued.

By definition, $z = e^{sT}$, where $s = \sigma + j\omega$ is the Laplace operator (not to be confused with $\{s_i\}$, the symbol values) and T is the time interval between adjacent samples^(5,6). Consequently,

$$z = e^{(\sigma + j\omega)T} \quad (5.6.17)$$

$$z = e^{\sigma T} e^{j\omega T} \quad (5.6.18)$$

If, for any of the α_i , $i = 1, 2 \dots g$, $|\alpha_i| > 1$, then the corresponding position of root α_i in the s -plane can be found from,

$$e^{\sigma T} |e^{j\omega T}| > 1 \quad (5.6.19)$$

which reduces to,

$$e^{\sigma T} > 1 \quad (5.6.20)$$

hence $\sigma T > 0$

(5.6.21)

Since T is a real non-negative number, equation (5.6.21) implies that $\sigma > 0$ and so the root must lie within the right-hand half-plane of the s -plane. This violates the definition of a minimum phase process⁽⁶⁾ and so any sequence which has roots of its z -transform with absolute values greater than unity (i.e. has roots which lie outside the unit circle in the z -plane) is not a minimum phase sequence.

A sequence $\{y_i\}$ with a z -transform $Y(z)$ can be forced to be minimum phase by including poles at the locations $z = \alpha_i$, where $|\alpha_i| > 1$ which cancel the effect of these zeroes, and adding new zeroes at values of z given by the complex conjugates of the reciprocals of the α_i , which is a phase transformation process on the sequence $\{y_i\}$ ⁽⁵⁾. The linear pre-filter is therefore an adaptive phase equaliser which attempts to remove any phase distortion present in the sampled impulse response of the baseband channel. Several methods of actually implementing the pre-filter are given in a later chapter but the general effect of the filtering process on an input sequence $\{y_i\}$ is now considered by use of a simple example.

Let $\{y_i\}$ be a sequence having 4 components, which is applied to the input of the pre-filter,

$$\begin{aligned} Y &= [y_0 \ y_1 \ y_2 \ y_3] \\ Y(z) &= y_0 + y_1 z^{-1} + y_2 z^{-2} + y_3 z^{-3} \\ Y(z) &= y_0 (1 - \alpha_1 z^{-1}) (1 - \alpha_2 z^{-1}) (1 - \alpha_3 z^{-1}) \end{aligned} \quad (5.6.22)$$

where $|\alpha_1| > 1$ and $|\alpha_2| = |\alpha_3| > 1$. α_1 is assumed to be a real-valued root whereas α_2 and α_3 are a complex conjugate pair. The input sequence is therefore not minimum phase since α_1 , α_2 and α_3 lie outside the unit circle in the z -plane.

The pre-filter now replaces the roots α_1 , α_2 and α_3 with new roots located at $z = \frac{1}{\alpha_1}$, $z = \frac{1}{\alpha_2^*}$ and $z = \frac{1}{\alpha_3^*}$, where α_i^* is the complex conjugate of α_i . The z-transform, $Q(z)$, of the output sequence, $\{q_i\}$, is,

$$Q(z) = y_0 \left(1 - \frac{1}{\alpha_1} z^{-1}\right) \left(1 - \frac{1}{\alpha_2^*} z^{-1}\right) \left(1 - \frac{1}{\alpha_3^*} z^{-1}\right) \quad (5.6.23)$$

Clearly, $\left|\frac{1}{\alpha_2^*}\right| = \left|\frac{1}{\alpha_2}\right|$, $\left|\frac{1}{\alpha_3^*}\right| = \left|\frac{1}{\alpha_3}\right|$ and for this example, $|\alpha_2| = |\alpha_3|$.

The locations of the new and original roots are illustrated in Figure 5.6.1.

Equation (5.6.23) may be re-written as,

$$Q(z) = \frac{-y_0}{\alpha_1 \alpha_2^* \alpha_3^*} (-\alpha_1 + z^{-1}) (-\alpha_2^* + z^{-1}) (-\alpha_3^* + z^{-1}) \quad (5.6.24)$$

which can be expanded to give,

$$\begin{aligned} Q(z) = \frac{y_0}{\alpha_1 \alpha_2^* \alpha_3^*} & (\alpha_1 \alpha_2^* \alpha_3^* - (\alpha_1 \alpha_2^* + \alpha_3^* (\alpha_1 + \alpha_2^*)) z^{-1} \\ & + (\alpha_1 + \alpha_2^* + \alpha_3^*) z^{-2} - z^{-3}) \end{aligned} \quad (5.6.25)$$

A similar expansion of equation (5.6.22) gives,

$$\begin{aligned} Y(z) = y_0 & (1 - (\alpha_1 + \alpha_2 + \alpha_3) z^{-1} + (\alpha_1 \alpha_2 + \alpha_3 (\alpha_1 + \alpha_2)) z^{-2} \\ & - (\alpha_1 \alpha_2 \alpha_3) z^{-3}) \end{aligned} \quad (5.6.26)$$

Comparisons between equations (5.6.25) and (5.6.26) reveals that the pre-filter introduces a constant shift in signal level across the filter, which, for the above example, is $-20 \log_{10}(|\alpha_1 \cdot \alpha_2 \cdot \alpha_3|)$ dB. When allowance is made for this change in signal level, the noise components at the output of the filter will have the same statistical properties as the noise components at its input and so as both the signal and noise components pass

through the filter in a practical system, the signal-to-noise ratio across the filter will not be changed⁽¹⁰⁾. However, when simulating the system using the models shown in Figures 4.3.8 to 4.3.10, it is convenient to include the pre-filter as part of the overall linear baseband channel. It is therefore necessary to make suitable adjustments to the signal-to-noise ratio, as defined in Appendix 3, when comparing systems operating over different telephone channels.

Tables 5.6.1 to 5.6.3 list the sampled impulse responses of the following combinations of normalised equipment filters and telephone circuits 1 to 3,

Table 5.6.1; Telephone circuits 1 to 3 with filter set 1, sampled at 3200 samples per second

Table 5.6.2; Telephone circuits 1 to 3 with filter set 1, sampled at 2400 samples per second

Table 5.6.3; Telephone circuits 1 to 3 with filter set 2, sampled at 3200 samples per second.

The combination of telephone circuits 1 to 3 with filter set 2, sampled at 2400 samples, has not been included here. Filter set 1 was originally designed for 2400 baud transmission and the use of filter set 2 at this rate does not significantly reduce the number of components in the overall channel's sampled impulse response. However, the increased bandwidth of the receiver filter in filter set 2 will cause an increase in noise power at its output which will degrade performance in the presence of additive noise.

In all cases, Tables 5.6.1 to 5.6.3 list the minimum-phase sampled impulses with the first component set to $1 + j0$ which, as explained in Chapter 6, simplifies detector design. The results listed in the tables were calculated using Program 4 in Appendix 10 which operates on the "raw"

sampled impulse responses of the baseband channels in the manner described by equations (5.6.22) to (5.6.26). The adjustment of the first component of the minimum phase sampled impulse responses to be $1 + j0$ introduces a further change of level across the pre-filter, as can be seen from equation (5.6.25). The total amount of signal level change due to both equalisation and scaling can be calculated by comparing the energies of the sampled impulse responses at the input and output of the pre-filter, or more strictly, the energies of the input and output sequences of Program 4. Table 5.6.4 lists the values by which all components in Tables 5.6.1 to 5.6.3 must be multiplied to ensure that the simulated linear pre-filter introduces no change in signal level; alternatively the numbers in brackets in Table 5.6.4 give the overall gain in signal level (in dB) introduced by the pre-filter for the various telephone circuit and filter combinations. The sampled impulses given in Tables 5.6.1 to 5.6.3 have been used in computer simulation studies to assess the relative performances of systems operating over different telephone channels. The equivalent baseband model of Figure 5.2.2 shows that in practical systems, the received baseband signal at the output of the receiver filter is sampled once per symbol (that is, at the baud rate). Clearly, the sampler must be synchronised to the symbol element rate of the actual data signal to prevent symbol "slippage". Another parameter of interest here is the phase of the timing signal; in the model of Figure 5.2.2, the sampler could theoretically operate at any sample phase whilst maintaining accurate timing frequency. However, in the computer simulations, the phase of the receiver sampler has effectively been fixed by the choice of sample phase used when determining the overall channel impulse response. Clearly, any degradation of system performance due to non-optimum timing phase must be taken into account when comparing systems operating over different

circuits or alternatively, the phase of the overall sampled impulses used in the simulations must be optimised for each of the combinations of telephone circuits and equipment filters.

The well-known Nyquist sampling theorem states that in order to avoid aliasing effects, a baseband signal with a maximum frequency of f_m Hz should be sampled at $\geq 2f_m$ ^(1,3,5,6). For the telephone circuits and filters considered here, the minimum expected baseband bandwidth will occur when filter set 2 is used with telephone channel 3 (about 400 Hz), whilst the widest bandwidth will result from the combination of telephone channel 1 with filter set 2 (about 1200 Hz), where it is assumed that the input data symbol is in the form of weighted impulses. The minimum sample rates at the receiver to avoid aliasing are therefore around 800 and 2400 samples per second, respectively. The sample rates used here, 2400 and 3200 samples per second, are obviously greater than the Nyquist rate for the worst case channel and so the performance of a system using this channel will not be affected by the phase of the timing signal ⁽⁴⁾. However, for the case of the largest bandwidth baseband signal, 2400 samples per second will just equal the Nyquist rate. In fact, for reasons mentioned earlier, the combination of filter set 2 with a baud rate of 2400 Hz is not considered in this work. The next highest bandwidth occurs with the combination of telephone circuit 1 with filter set 1, which has a bandwidth of approximately 1000 Hz. Sampling at 2400 samples per second (as in the case for the 256-point QAM system) again satisfies the Nyquist criterion and so performance will be independent of sample phase. The above results were in fact confirmed by obtaining the sampled impulse responses of telephone circuits 1 to 3 combined with filter set 1, at various sample rates and sample phases; Program 3 was again used to perform these tests. However, as the sample phase of the sampled impulse responses is easily changed by

simply selecting different sets of finely-sampled values from the output of Program 3, the results in Tables 5.6.1 to 5.6.3 have been selected to be as near optimum as possible. (It has been shown elsewhere⁽¹⁰⁾ that at sample rates very slightly less than the Nyquist rate, the phase of receiver sampling has a negligible effect on the performance of near maximum-likelihood detectors but has a larger effect for systems using optimum non-linear equalisers; to protect against the latter case, the phases of the sampled impulse responses shown in Tables 5.6.1 to 5.6.3 have been adjusted to ensure that the non-linear equaliser performs within about 0.5 dB of optimum.)

Finally, Table 5.6.5 gives a comparison between the sampled impulse responses of telephone circuits 1-3 with filter sets 1 and 2 in terms of the position of the peak component and the length of the responses relative to the value of the peak component. As expected from Figures 5.5.5 and 5.5.6, the use of the wider bandwidth filters will have less of an effect on the overall sampled impulse of telephone circuit 3 due to the more severe bandlimiting introduced by this circuit compared with telephone circuits 1 and 2. However, the reduction in the length of the sampled impulse responses when using filter set 2, which implies lower levels of distortion, can clearly be seen from Table 5.6.5.

C

5.7 References

1. Clark, A.P., "Principles of Digital Data Transmission, Pentech Press, 1976
2. Harvey, J.D., "Synchronisation of a Synchronous Modem", SERC Report No. GR/A/1200.7, December 1980
3. Schwartz, M., "Information Transmission, Modulation and Noise", McGraw-Hill Inc., Third Edition, 1980
4. McVerry, F., "High Speed Data Transmission over H.F. Radio Links", Ph.D. Thesis, Loughborough University of Technology, 1982.
5. Clark, A.P., "Advanced Data-Transmission Systems", Pentech Press, 1977
6. Lynn, P.A., "The Analysis and Processing of Signals", Macmillan Press Ltd., 1979
7. Fairfield, M.J., "Equipment Filter Design", Internal Communication, Loughborough University of Technology, 1978
8. Fairfield, M.J., "The Chosen Equipment Filters (410)", Internal Communication, Loughborough University of Technology, 1978
9. Najdi, H.Y., "Digital Data Transmission over Voice Channels", Ph.D. Thesis, Loughborough University of Technology, 1982

10. Clark, A.P., Najdi, H.Y. and Fairfield, M.J., "Data Transmission at 19.2 kbit/s over Telephone Circuits", Radio and Electronic Engineer, Vol.53, No.4, pp 157-166, April 1983
11. Clark, A.P. and Fairfield, M.J., "Detection Processes for a 9600 bit/s Modem", Radio and Electronic Engineer, Vol.51, No.9, pp 455-465, Sept 1981
12. Clark, A.P., Harvey, J.D. and Driscoll, J.P., "Near-Maximum-Likelihood Detection Processes for Distorted Digital Signals", Radio and Electronic Engineer, Vol.48, No.6, pp 301-309, June 1978
13. Clark, A.P., Kwong, C.P. and Harvey, J.D., "Detection Processes for Severely Distorted Digital Signals", Electronic Circuits and Systems, Vol.3, No.1, pp 27-37, Jan 1979
14. Clark, A.P. and Harvey, J.D., "Detection of Distorted QAM Signals", Electronic Circuits and Systems, Vol.1, No.3, pp 103-109, April 1977
15. Orr, T., "Designers Notebook; Switched Capacitor Filters", Electronics Today International, pp 45-47, November 1982
16. "Switched Capacitor Filters for 1200 Baud Modems", Technology Trends, Electronic Product Design, p9, July 1982
17. "The Holt Inc. HI-3700 Semicustom Capacitor Filter Chip", Design News, Electronic Product Design, p17, Sept 1982

18. Mitel Semiconductot Corp., "The MT8912 ISO²-CMOS PCM Filter", Specification and Data Sheets, 1981
19. Intel Corporation, "The 2912A PCM Transmit/Receive Filter", Specification and Data Sheets, 1980
20. Motorola Inc., "The MC145414 Dual Tunable Low-Pass Sampled Data Filters", Specification and Data Sheets, 1982
21. GTE Microcircuits, "The G8912B ISO-CMOS PCM Transmit/Receive Filters", Specification and Data Sheets, Dec 1981
22. EG and G Reticcon Inc., "The R5620 Universal Active Filter", Preliminary Data Sheet, 1982.
23. Bateman, S.C., "An Investigation of Group Delay Characteristics of the Mitel MT8912 and Intel I2912 PCM Line Filters and Their Possible Application in High Speed Data Modems", Internal Communication, Loughborough University of Technology, Jan 1982
24. Fairfield, M.J., "Analysis of Telephone Connections", Internal Communication, Loughborough University of Technology, 1978

FREQUENCY (Hz)	FILTER SET 1		FILTER SET 2	
	ATTENUATION (dB)	GROUP DELAY (mS)	ATTENUATION (dB)	GROUP DELAY (mS)
50	99.99	1.51	93.87	-
100	93.79	1.83	54.94	1.42
150	77.62	2.12	41.67	2.00
200	64.73	2.37	31.66	2.02
250	53.94	2.58	25.25	2.01
300	44.70	2.76	20.00	1.46
350	36.70	2.91	15.64	1.41
400	30.40	3.03	12.00	1.34
450	24.40	3.15	8.96	1.35
500	19.50	3.26	6.52	1.36
550	15.65	3.37	4.55	1.31
600	12.30	3.47	2.98	1.25
650	9.55	3.48	1.97	1.19
700	7.30	3.48	0.94	1.11
750	5.50	3.47	0.39	0.98
800	4.10	3.43	0.09	0.93
850	3.10	3.41	0.00	0.84
900	2.20	3.37	0.00	0.77
950	1.65	3.32	0.00	0.71
1000	1.25	3.26	0.00	0.64
1050	0.75	3.19	0.00	0.59
1100	0.35	3.14	0.00	0.55
1150	0.02	3.09	0.00	0.51
1200	0.00	3.04	0.00	0.46
1250	0.00	3.01	0.00	0.41

Table 5.4.1: Attenuation and Group Delay Characteristics
of Filter Set 1 and 2

1300	0.00	2.98	0.00	0.38
1350	0.00	2.95	0.00	0.37
1400	0.00	2.93	0.00	0.36
1450	0.00	2.90	0.00	0.37
1500	0.00	2.88	0.00	0.37
1550	0.00	2.87	0.00	0.37
1600	0.00	2.87	0.00	0.36
1650	0.00	2.86	0.00	0.36
1700	0.00	2.86	0.00	0.36
1750	0.00	2.85	0.00	0.36
1800	0.00	2.85	0.00	0.36
1850	0.00	2.85	0.00	0.35
1900	0.00	2.88	0.00	0.34
1950	0.00	2.88	0.00	0.35
2000	0.01	2.89	0.00	0.35
2050	0.05	2.90	0.00	0.39
2100	0.13	2.92	0.00	0.43
2150	0.23	2.95	0.00	0.40
2200	0.35	2.97	0.00	0.48
2250	0.40	3.00	0.00	0.49
2300	0.45	3.03	0.00	0.52
2350	0.57	3.05	0.00	0.57
2400	0.75	3.10	0.00	0.62
2450	0.93	3.15	0.00	0.71
2500	1.45	3.19	0.00	0.78
2550	1.97	3.25	0.00	0.75
2600	2.64	3.30	0.00	0.92
2650	3.25	3.35	0.00	0.97

Table 5.4.1 continued...

2700	4.05	3.39	0.00	1.10
2750	5.20	3.42	0.00	1.61
2800	6.72	3.44	0.10	1.29
2850	8.20	3.47	0.39	1.43
2900	10.25	3.49	0.94	1.55
2950	12.45	3.50	1.79	1.72
3000	14.95	3.51	3.48	1.91
3050	17.70	3.51	6.04	2.12
3100	21.10	3.49	9.03	2.36
3150	24.80	3.45	11.96	2.31
3200	28.60	3.41	19.00	2.28
3250	32.83	3.33	24.04	2.15
3300	37.63	3.22	31.12	2.05
3350	43.10	3.08	39.24	1.91
3400	49.15	2.89	47.66	1.76
3450	55.55	2.65	59.67	1.52
3500	62.30	2.36	74.54	1.30
3550	69.55	2.05	89.87	0.98
3600	76.75	1.69	98.39	0.67
3650	84.05	1.33	101.56	0.60
3700	90.85	1.03	105.62	0.54
3750	96.80	0.79	107.44	-

Table 5.4.1 continued....

FREQUENCY (Hz)	TELEPHONE CIRCUIT 1		TELEPHONE CIRCUIT 2		TELEPHONE CIRCUIT 3	
	ATT. (dB)	G.D. (mS)	ATT. (dB)	G.D. (mS)	ATT. (dB)	G.D. (mS)
50	87.10	11.00	30.00	15.00	84.50	24.00
100	42.10	9.30	20.00	8.00	44.50	19.00
150	17.10	8.30	15.00	3.70	19.50	15.00
200	11.60	7.35	11.80	2.80	9.30	12.00
250	9.10	6.40	8.50	2.00	7.00	10.00
300	7.40	5.81	6.90	1.43	5.30	8.33
350	5.60	4.90	5.50	1.15	3.60	6.80
400	4.40	4.40	4.50	0.95	2.60	5.85
450	3.60	3.80	3.70	0.73	1.80	4.80
500	2.70	3.30	3.00	0.52	1.25	4.10
550	2.30	2.85	2.50	0.39	0.60	3.50
600	2.00	2.48	1.95	0.28	0.20	3.12
650	1.70	2.10	1.70	0.18	0.12	2.70
700	1.30	1.80	1.50	0.12	0.06	2.25
750	1.00	1.55	1.25	0.08	0.02	1.90
800	0.80	1.39	0.75	0.05	0.00	1.67
850	0.60	1.18	0.62	0.03	0.01	1.40
900	0.40	1.02	0.45	0.02	0.05	1.22
950	0.22	0.88	0.26	0.01	0.10	1.04
1000	0.20	0.75	0.30	0.00	0.20	0.90
1050	0.18	0.67	0.26	0.00	0.40	0.80
1100	0.16	0.61	0.22	0.00	0.60	0.70
1150	0.14	0.55	0.18	0.00	0.85	0.62
1200	0.12	0.50	0.15	0.00	1.20	0.59
1250	0.10	0.45	0.12	0.01	1.40	0.48
1300	0.08	0.40	0.09	0.02	1.75	0.40
1350	0.06	0.35	0.03	0.06	2.20	0.36
1400	0.04	0.29	0.00	0.10	2.60	0.33
1450	0.02	0.24	0.02	0.12	3.00	0.28
1500	0.00	0.20	0.10	0.13	3.50	0.22
1550	0.00	0.15	0.30	0.14	4.00	0.20
1600	0.02	0.11	0.60	0.15	4.50	0.18
1650	0.06	0.07	0.80	0.16	5.00	0.17
1700	0.10	0.05	1.10	0.18	5.75	0.16
1750	0.20	0.03	1.40	0.20	6.40	0.14
1800	0.32	0.02	1.60	0.22	7.00	0.12
1850	0.46	0.01	1.80	0.25	7.60	0.09
1900	0.60	0.00	2.00	0.28	8.30	0.06
1950	0.75	0.00	2.15	0.31	9.25	0.02
2000	0.90	0.00	2.30	0.35	9.80	0.00
2050	1.05	0.00	2.45	0.39	10.50	0.01
2100	1.20	0.00	2.60	0.44	11.30	0.02
2150	1.30	0.00	2.80	0.48	12.20	0.05
2200	1.40	0.00	3.10	0.51	12.90	0.08
2250	1.55	0.01	3.40	0.55	13.80	0.11

Table 5.5.1: Attenuation and Group Delay Characteristics of
Telephone Circuits 1 to 3, sampled at Intervals
of 50 Hz.

2300	1.70	0.02	3.70	0.58	14.60	0.18
2350	1.85	0.03	4.00	0.61	15.50	0.27
2400	1.95	0.04	4.30	0.65	16.30	0.33
2450	2.10	0.05	4.50	0.70	17.10	0.39
2500	2.30	0.06	4.80	0.75	18.10	0.48
2550	2.45	0.07	5.00	0.81	19.00	0.57
2600	2.60	0.09	5.20	0.86	19.50	0.66
2650	2.75	0.11	5.50	0.93	20.10	0.73
2700	2.90	0.12	5.70	1.00	21.00	0.88
2750	3.20	0.13	5.95	1.10	21.90	1.00
2800	3.60	0.15	6.20	1.25	22.50	1.04
2850	3.85	0.18	6.45	1.39	23.10	1.12
2900	4.20	0.20	6.70	1.53	23.80	1.30
2950	4.50	0.25	6.95	1.67	24.50	1.60
3000	4.80	0.30	7.20	1.81	25.30	1.92
3050	5.50	0.34	7.45	1.96	26.10	2.20
3100	6.35	0.44	7.80	2.12	26.85	2.50
3150	7.60	0.52	8.20	2.30	27.60	2.85
3200	8.80	0.58	8.60	2.50	28.45	3.37
3250	9.80	0.70	8.80	2.70	29.15	3.70
3300	11.30	0.85	9.10	2.90	29.50	4.50
3350	13.10	1.05	9.50	3.10	30.10	5.40
3400	14.80	1.20	9.90	3.25	30.70	6.00
3450	16.60	1.40	10.05	3.50	31.20	7.00
3500	18.60	1.75	10.25	3.75	31.70	8.00
3550	21.10	2.05	10.40	4.00	32.10	9.00
3600	24.10	2.40	10.55	4.25	32.50	10.00
3650	27.10	2.75	10.70	4.50	32.90	11.05
3700	31.10	3.15	10.90	4.75	33.30	12.10
3750	34.10	3.60	11.10	5.00	33.50	13.20

Table 5.5.1 continued...

TELEPHONE CIRCUIT 1		TELEPHONE CIRCUIT 2		TELEPHONE CIRCUIT 3	
REAL	IMAG.	REAL	IMAG.	REAL	IMAG.
1.0000	0.0000	1.0000	0.0000	1.0000	0.0000
2.0391	0.2530	2.0738	0.3987	1.8203	1.3280
0.8687	0.4136	0.8787	0.7217	-0.0214	2.0083
-0.4242	0.0415	-0.4814	0.1716	-1.2337	0.1005
-0.0944	-0.1284	-0.1541	-0.2023	0.0324	-0.7086
0.1541	0.0491	0.1289	0.0262	0.3202	0.1528
-0.0305	0.0295	-0.0662	0.0538	-0.1583	0.1402
-0.0180	-0.0319	-0.0380	-0.0428	0.0085	-0.0914
0.0171	0.0058	0.0253	0.0022	0.0385	0.0254
-0.0078	0.0046	-0.0108	0.0051	-0.0402	0.0121
-0.0058	-0.0050	0.0072	-0.0111	0.0085	-0.0212
0.0052	0.0018	0.0031	0.0109	0.0091	0.0043
0.0021	0.0024	0.0014	-0.0045	0.0017	0.0098
-0.0017	-0.0011	0.0006	0.0064	-0.0072	-0.0017
-0.0033	0.0005	-0.0088	-0.0035	0.0032	-0.0105
0.0009	0.0008	0.0062	-0.0050	0.0068	0.0042
0.0012	-0.0007	-0.0042	0.0056	-0.0064	-0.0049
-0.0002	0.0024	-0.0005	-0.0048	0.0122	-0.0031
-0.0011	-0.0019	0.0041	0.0029	-0.0027	0.0028
-0.0012	-0.0017	-0.0056	-0.0003	0.0040	-0.0053

Table 5.6.1: Minimum Phase Impulse Responses for Telephone Circuits 1 to 3 with Filter Set 1, Sampled at 3200 Samples/Second.

TELEPHONE CIRCUIT 1		TELEPHONE CIRCUIT 2		TELEPHONE CIRCUIT 3	
REAL	IMAG.	REAL	IMAG.	REAL	IMAG.
1.0000	0.0000	1.0000	0.0000	1.000	0.0000
0.4903	-0.0961	0.5151	0.3385	0.4861	1.0988
-0.1443	0.0112	-0.1780	0.0406	-0.5980	0.0703
0.0349	0.0132	0.0158	-0.0455	0.1702	-0.1938
-0.0081	0.0090	-0.0109	-0.0194	0.0100	-0.0258
0.0029	-0.0089	0.0011	0.0072	-0.0134	0.0110
-0.0052	0.0086	-0.0047	-0.0072	0.0056	-0.0042
0.0047	-0.0062	0.0104	0.0040	0.0003	0.0003
-0.0017	0.0043	-0.0040	0.0024	-0.0008	0.0041
-0.0018	-0.0025	-0.00-6	-0.0022	0.0000	-0.0061
0.0027	0.0013	0.0020	-0.0007	0.0007	-0.0007
-0.0010	-0.0004	-0.0005	0.0001	0.0037	0.0002
-0.0006	-0.0002	-0.0021	0.0003	-0.0019	-0.0025
0.0002	0.0008	0.0008	-0.0020	0.0020	0.0008
0.0001	-0.0006	0.0016	0.0009	0.0005	-0.0002
-0.0000	-0.0001	0.0000	-0.0007	-0.0022	0.0002
0.0000	0.0000	0.0000	0.0000	0.0007	-0.0005
0.0000	0.0000	0.0000	0.0000	-0.0008	0.0002
0.0000	0.0000	0.0000	0.0000	0.0005	0.0005

Table 5.6.2: Minimum Phase Impulse Responses for Telephone

Circuits 1 to 3 with Filter Set 1, Sampled at 2400
Samples/Second.

TELEPHONE CIRCUIT 1		TELEPHONE CIRCUIT 2		TELEPHONE CIRCUIT 3	
REAL	IMAG.	REAL	IMAG.	REAL	IMAG.
1.0000	0.0000	1.0000	0.0000	1.0000	0.0000
1.2012	0.1967	1.1147	0.3411	1.3409	1.3469
-0.0825	0.1576	-0.1176	0.3031	-0.7359	1.2836
-0.1528	-0.0706	-0.1499	-0.0926	-0.6184	-0.0579
0.1311	0.0119	0.0916	-0.00-8	0.0457	-0.0106
-0.0365	0.0236	-0.0551	0.0272	-0.1298	0.2201
-0.0021	-0.0040	-0.0164	-0.0180	-0.0272	-0.1301
-0.0031	0.0035	-0.0068	-0.0006	0.0351	0.0534
0.0078	-0.0046	0.0082	-0.0071	-0.0288	-0.0274
0.0037	0.0034	0.0060	0.0055	0.0356	0.0136
-0.0093	-0.0004	-0.0068	-0.0014	-0.0291	0.0125
0.0027	-0.0008	0.0040	-0.0013	0.0036	-0.0189
0.0026	-0.0017	0.0013	0.0014	0.0113	0.0043
-0.0020	0.0022	-0.0029	-0.0004	-0.0035	0.0055
-0.0098	-0.0001	-0.0005	-0.0021	-0.0035	-0.0080
0.0015	-0.00-5	0.0000	-0.0006	0.0081	-0.0011
0.0007	0.0021	-0.0001	0.0003	-0.0007	0.0008
-0.0024	0.0003	-0.0003	0.0007	0.0033	-0.0017
0.0017	-0.0008	-0.0009	0.0007	-0.0016	0.0001
-0.0006	0.0000	0.0006	-0.0005	0.0032	-0.0027

Table 5.6.3: Minimum Phase Impulse Responses for Telephone Circuits

1-3 with Filter Set 2, at 3200 Samples/Second.

TELEPHONE CIRCUIT	FILTER SET AND SAMPLE RATE		
	FILTER SET 1	FILTER SET 1	FILTER SET 2
	3200 s/s	2400 s/s	3200 s/s
1	0.4261 (7.41)	0.8296 (1.62)	0.5816 (4.71)
2	0.3692 (8.65)	0.7182 (2.88)	0.4123 (7.70)
3	0.2023(13.88)	0.3630 (8.80)	0.2471(12.14)

Table 5.6.4: Scale Factors Introduced by the Phase Equalisation Process. Figures in Brackets give the Gain (in dB) Between "Raw" and Equalised Sequences.

	Telephone Circuit 1		Telephone Circuit 2		Telephone Circuit 3	
	Filter Set 1	Filter Set 2	Filter Set 1	Filter Set 2	Filter Set 1	Filter Set 2
a)	2	2	2	2	2	2
b)	4	4	5	5	6	6
c)	6	6	7	5	8	8
d)	8	6	9	7	11	10

a) Position of peak component

b) Number of components for which $(\text{Real}^2 + \text{Imag}^2) > 0.1 \times \text{peak}$

c) Number of components for which $(\text{Real}^2 + \text{Imag}^2) > 0.01 \times \text{peak}$

d) Number of components for which $(\text{Real}^2 + \text{Imag}^2) > 0.001 \times \text{peak}$

Table 5.6.5: Comparison of Minimum Phase Sampled Impulse Responses

for Telephone Circuits 1-3 when using Filter Sets 1 and 2

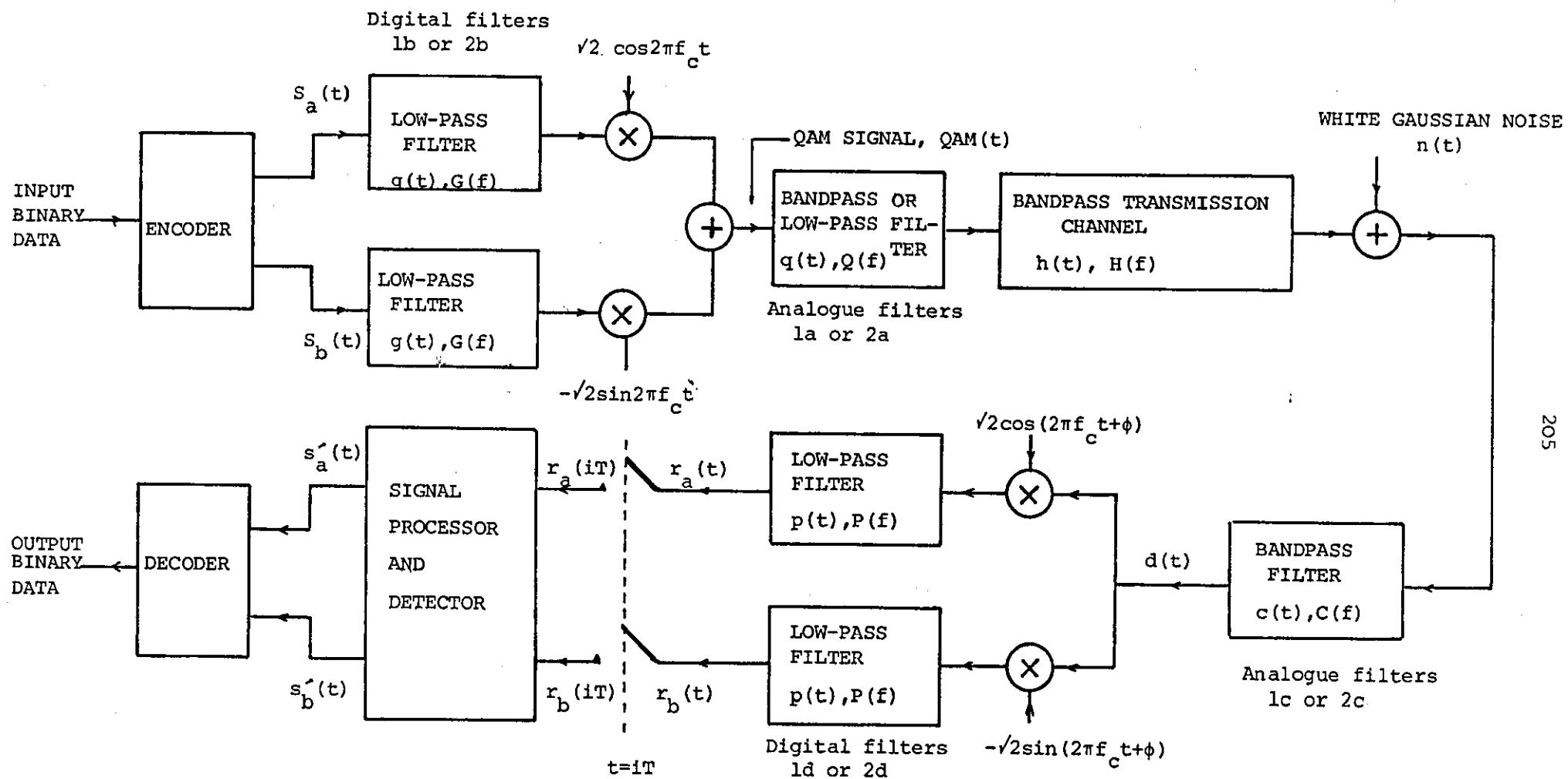


Figure 5.2.1: Model of QAM Data Transmission System

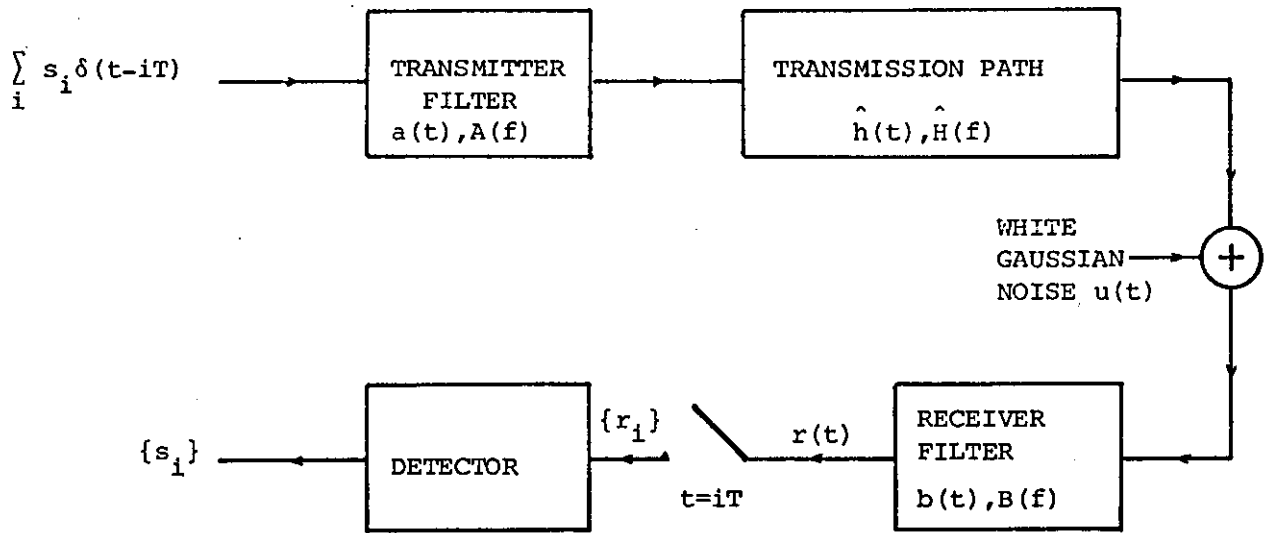


Figure 5.2.2: Baseband Model of the QAM System

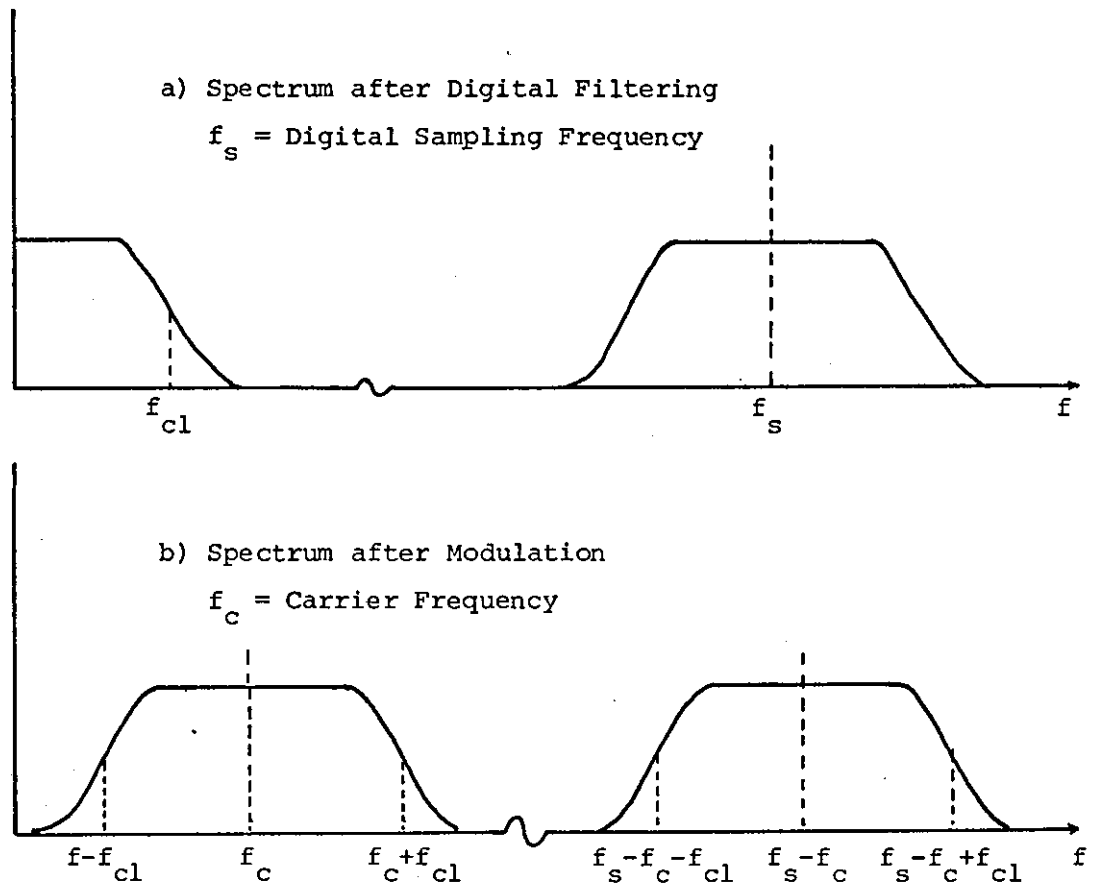
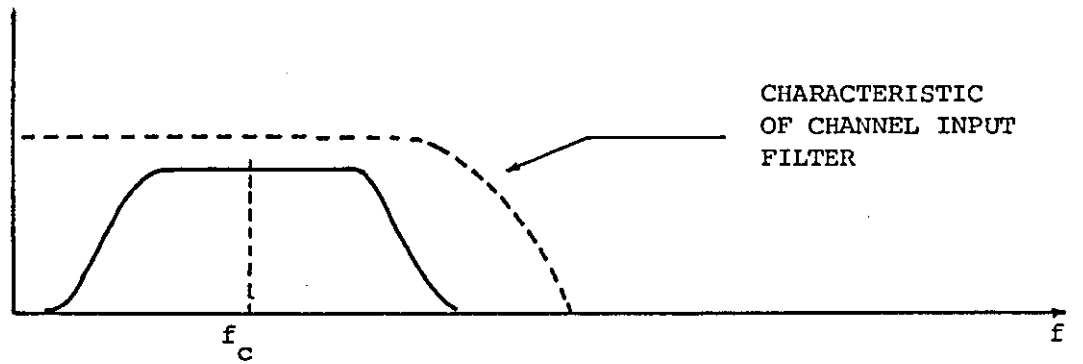
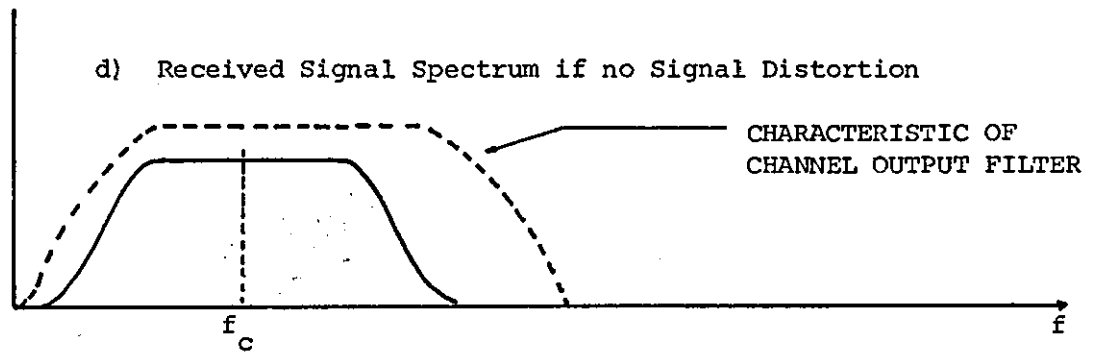


Figure 5.2.3: Signal Spectra at Specific Points in the QAM System

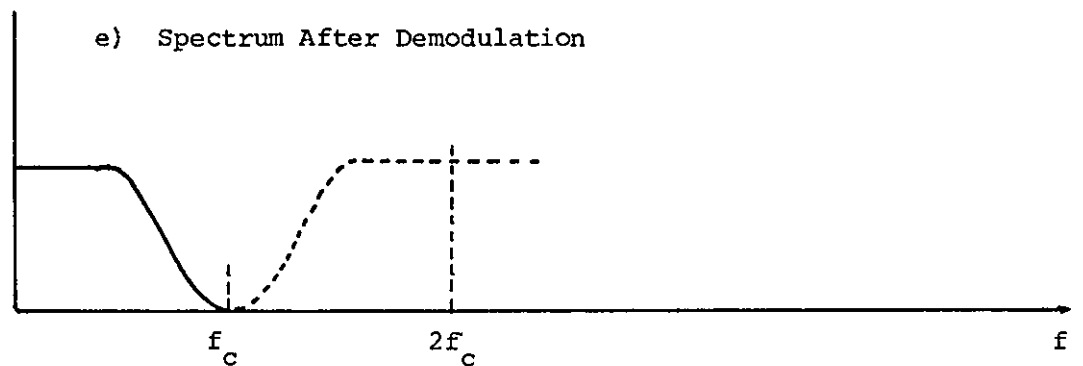
c) Spectrum of Transmitted Signal



d) Received Signal Spectrum if no Signal Distortion



e) Spectrum After Demodulation



f) Received Baseband Spectrum after Low-Pass Filter

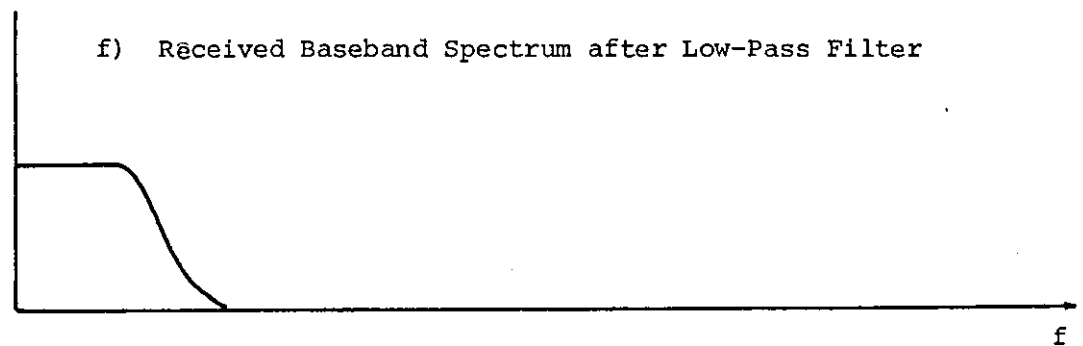


Figure 5.2.3: Contd. Signal Spectra at Specific Points in the QAM System

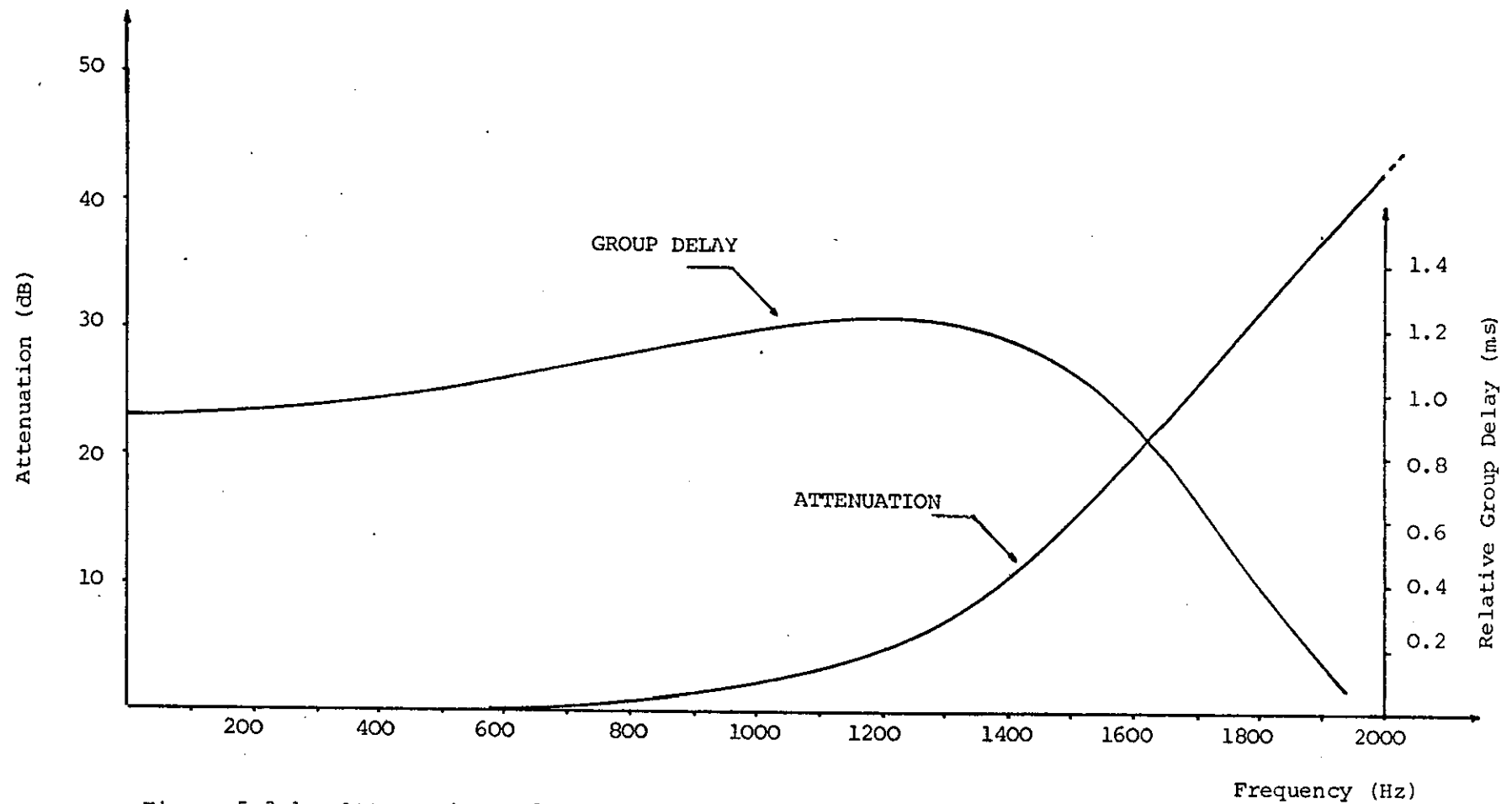


Figure 5.3.1: Attenuation and Group Delay Characteristics for Low-Pass Filter 1b

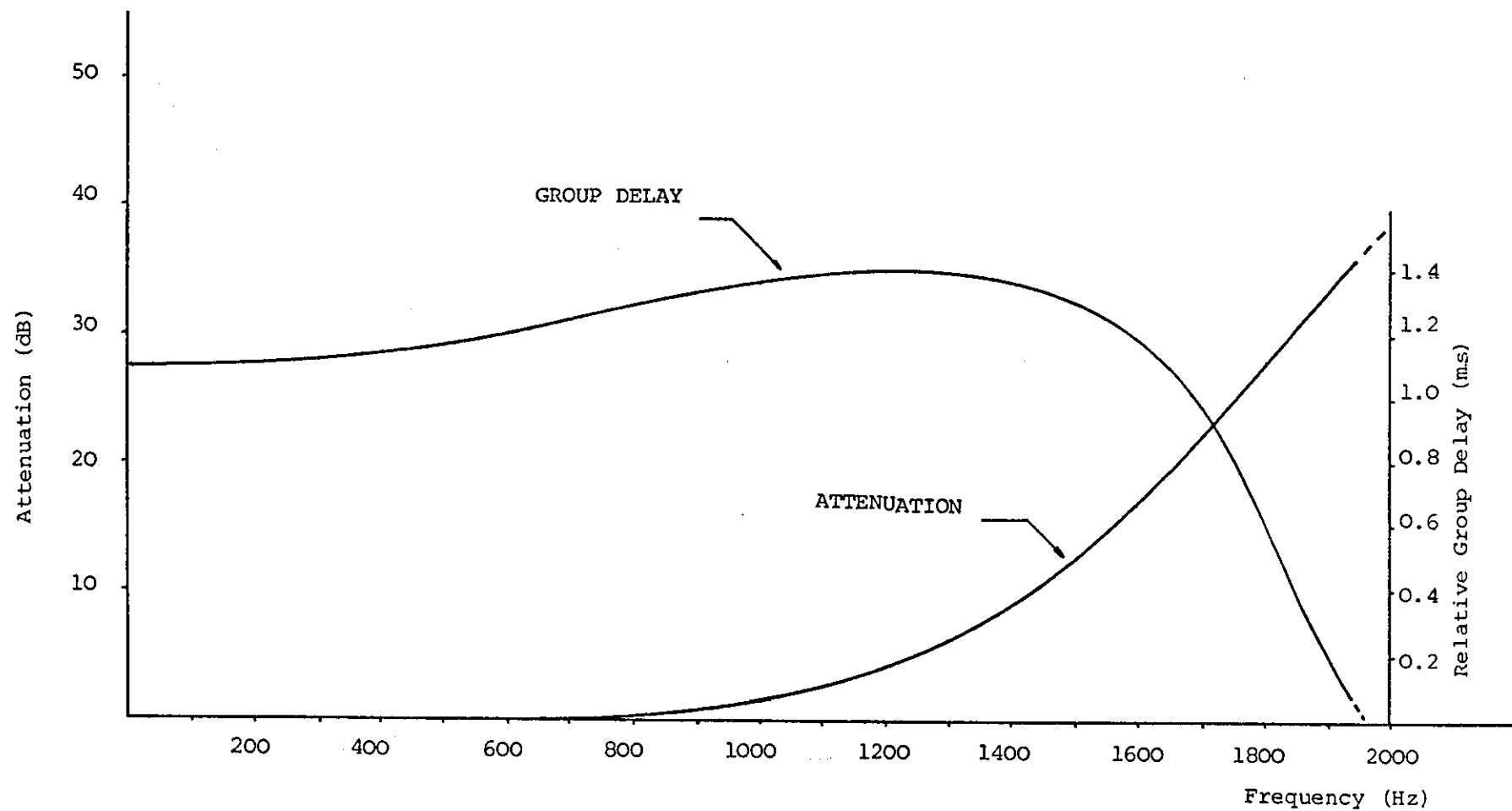
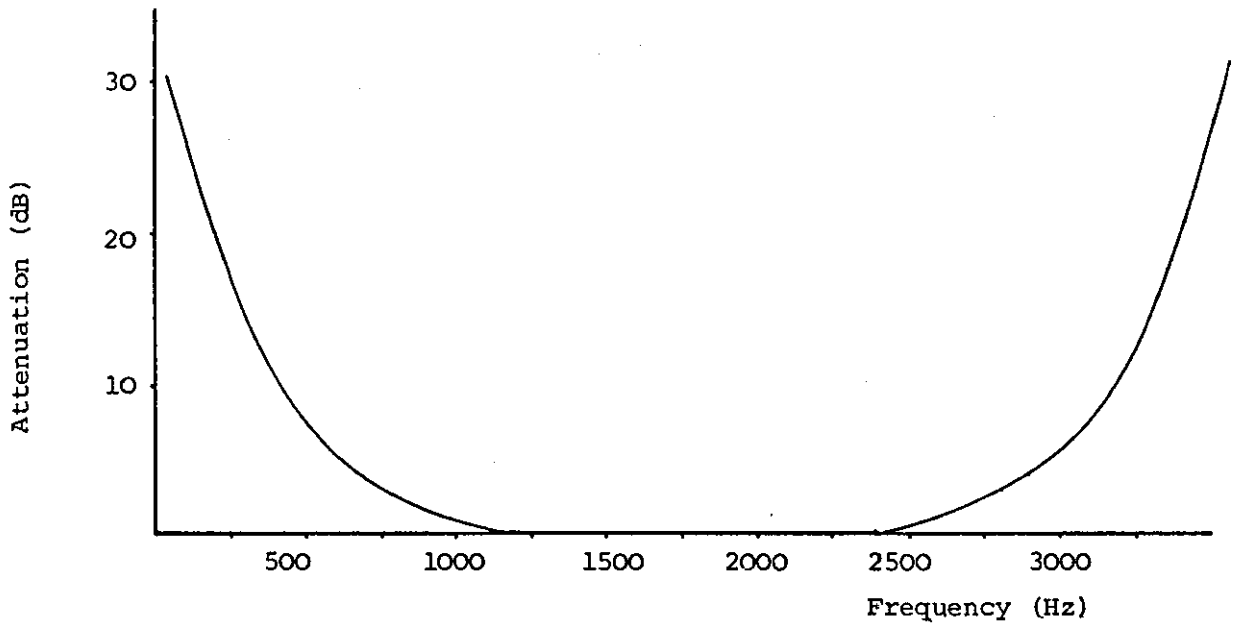
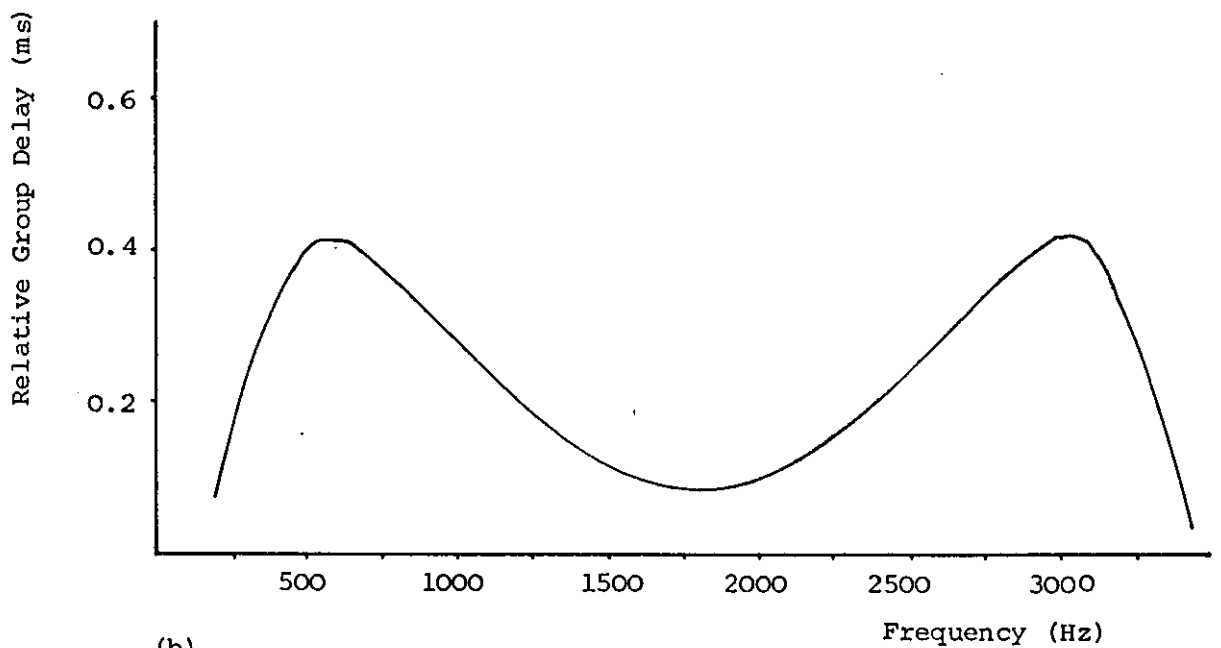


Figure 5.3.2: Attenuation and Group Delay Characteristics for Low-Pass Filter 1d



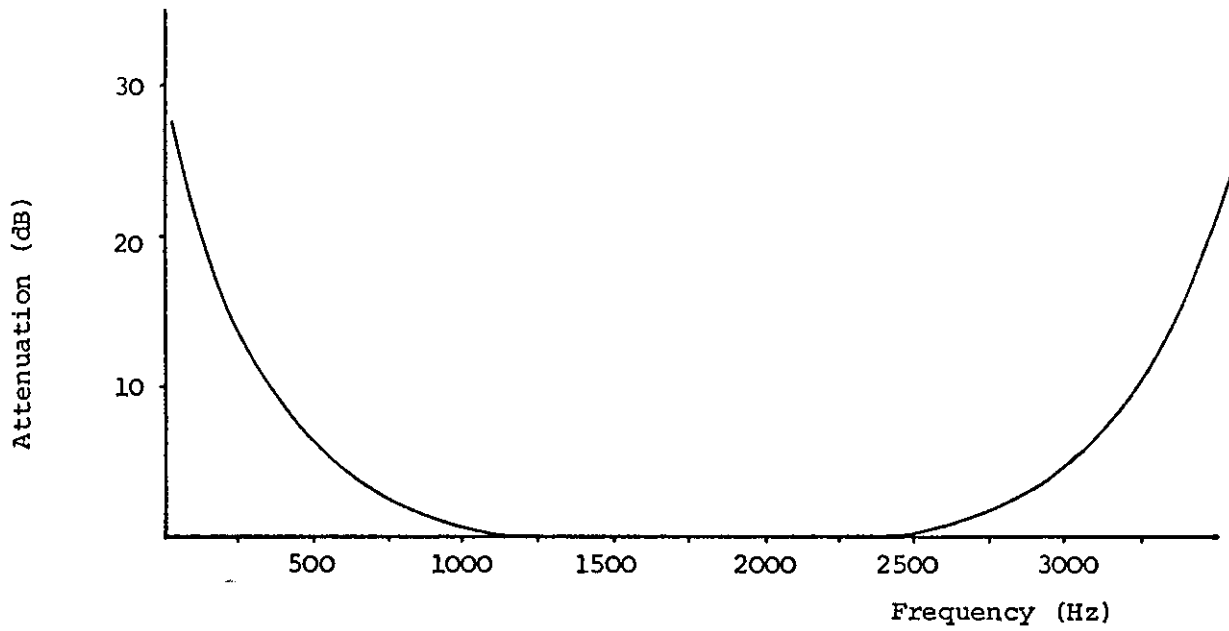
(a)



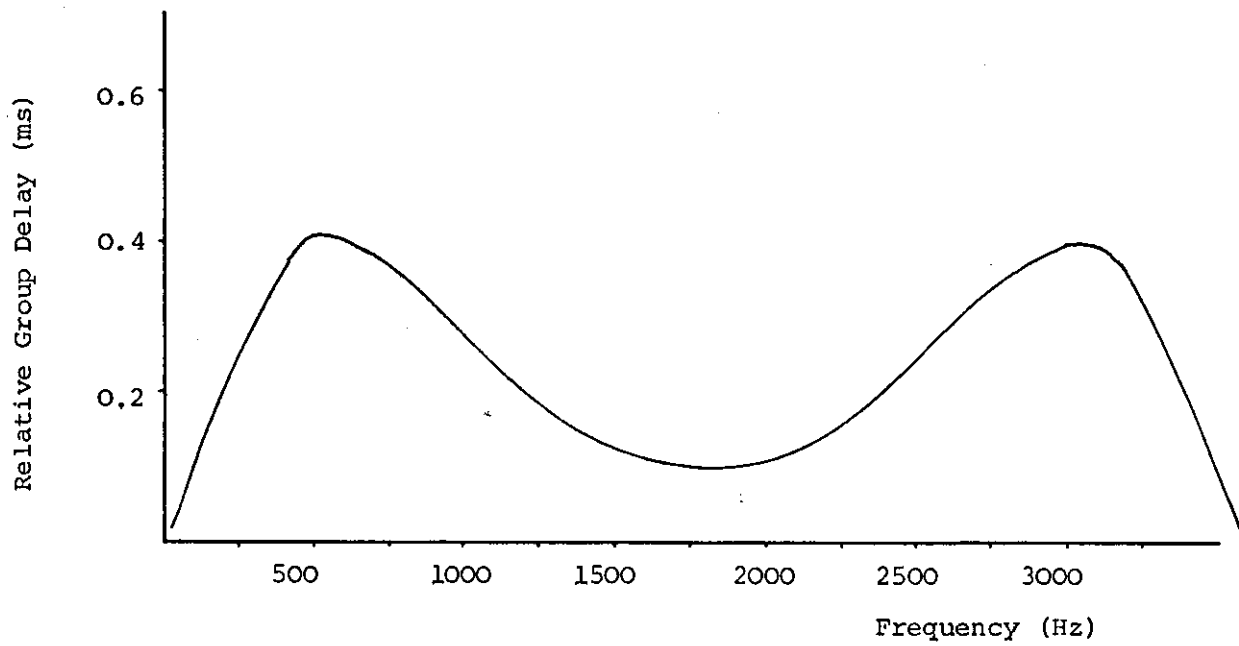
(b)

Figure 5.3.3: a) Attenuation Characteristic

b) Group Delay Characteristic
of Filter 1b after Translation



(a)



(b)

Figure 5.3.4: a) Attenuation Characteristic

b) Group Delay Characteristic
of Filter 1d after Translation

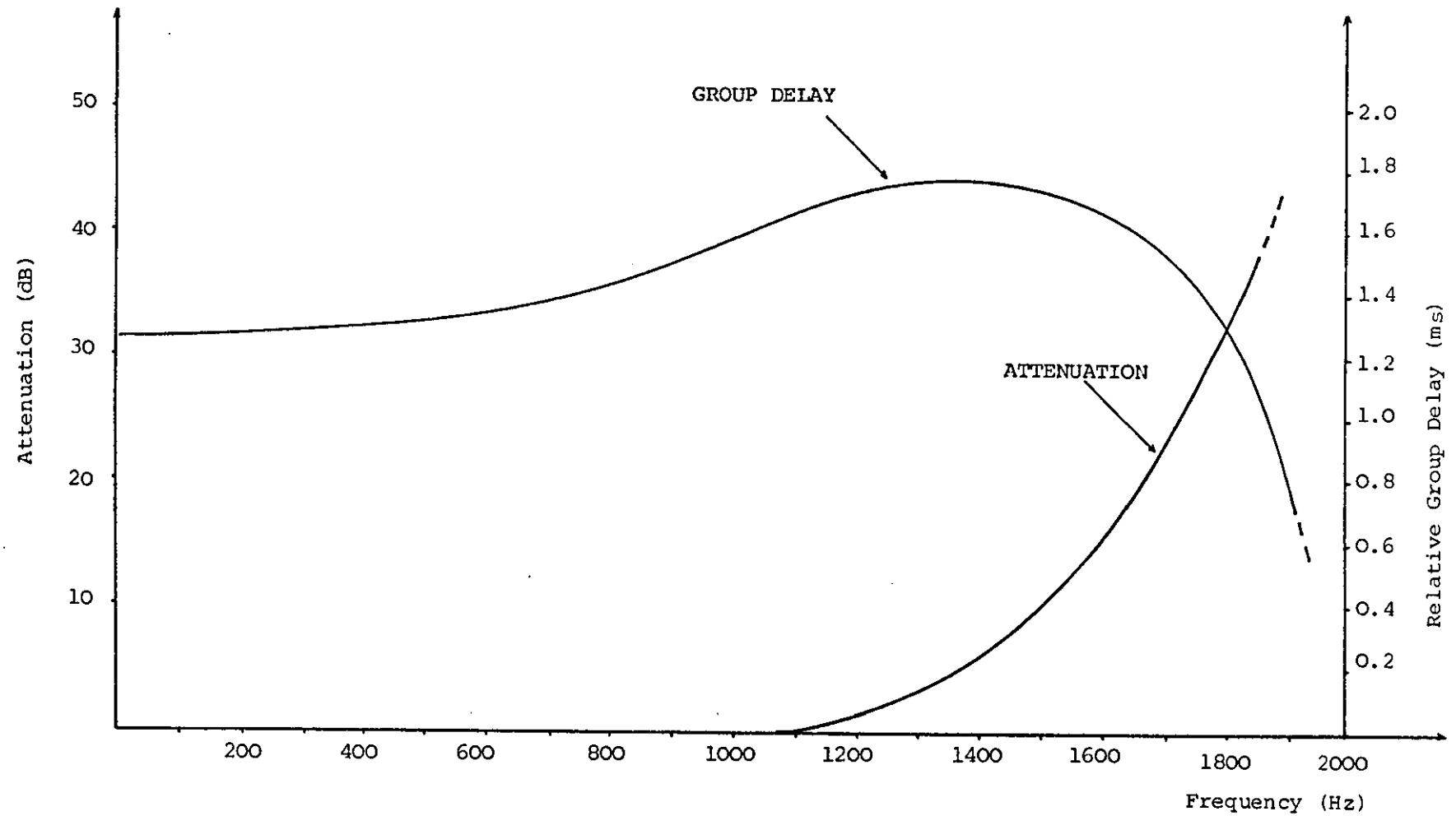


Figure 5.3.5: Attenuation and Group Delay Characteristics for Low-Pass Filters 2b and 2c

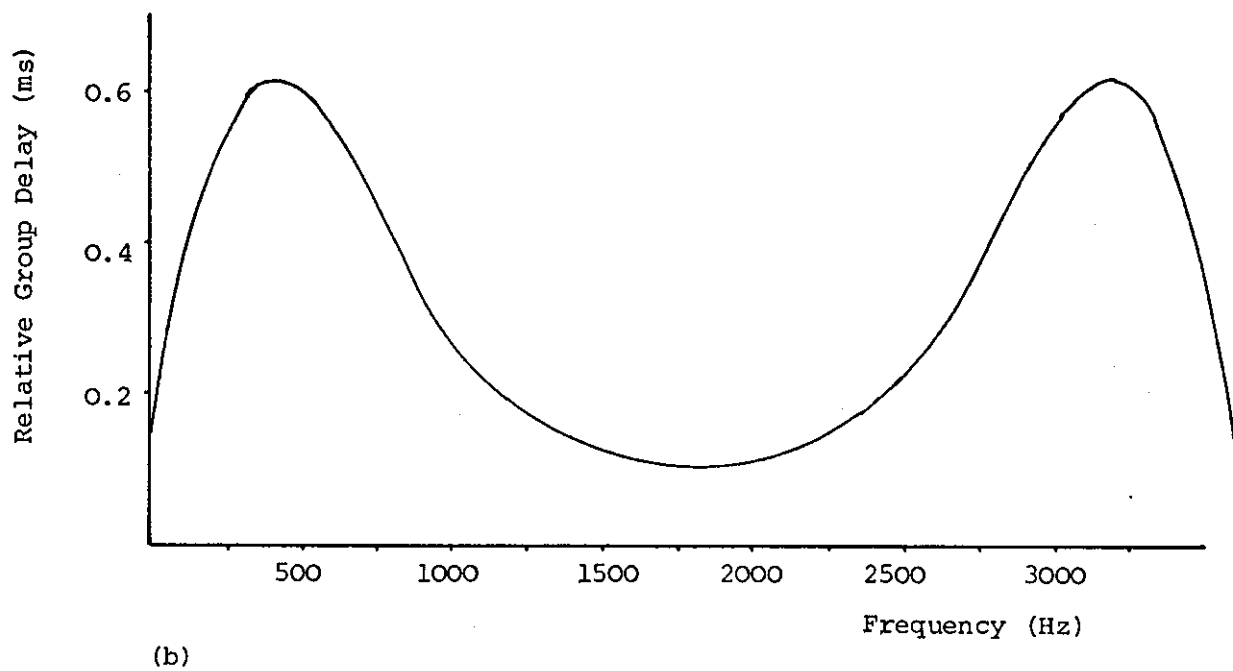
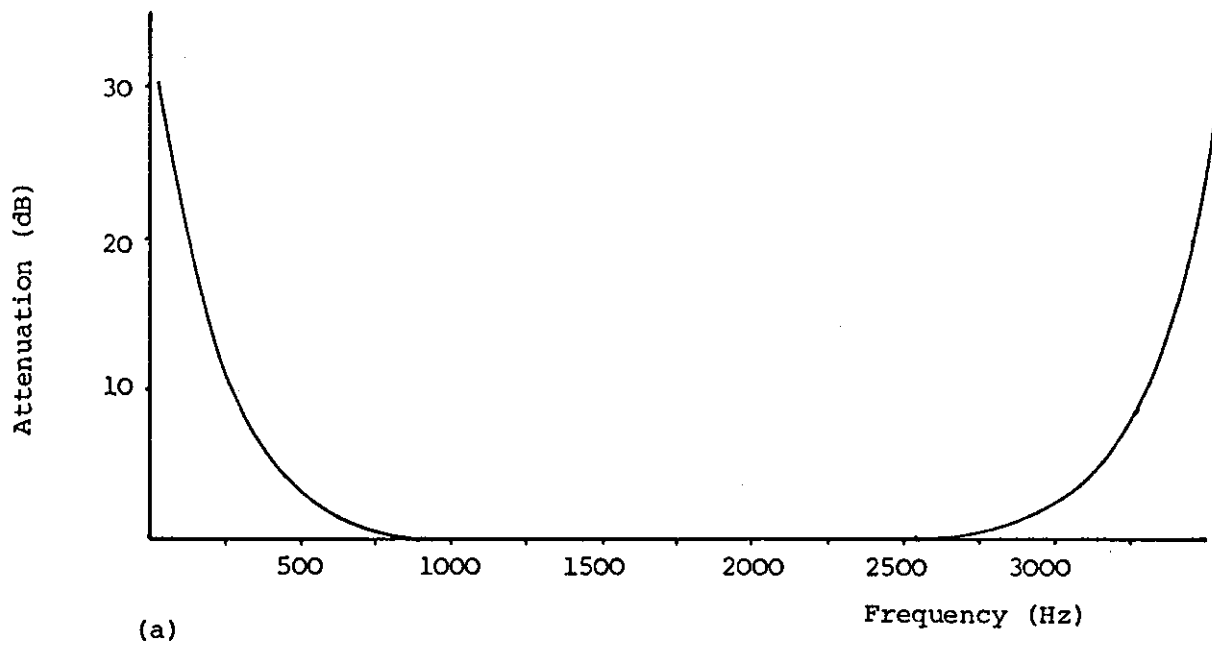


Figure 5.3.6: a) Attenuation Characteristic

b) Group Delay Characteristic
of Filters 2b and 2d After Translation

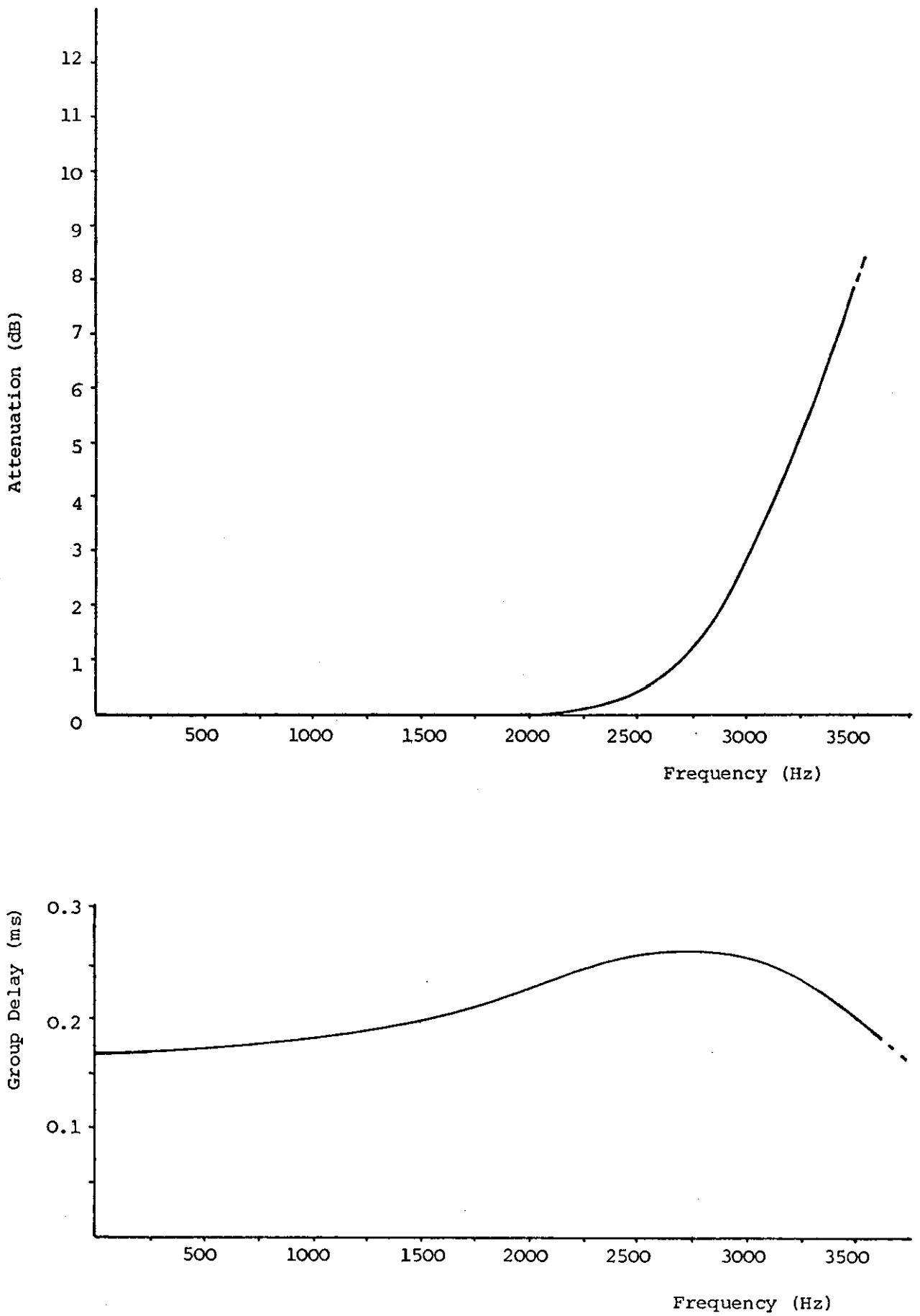


Figure 5.3.7: Attenuation and Group Delay Characteristics for Filter 1a

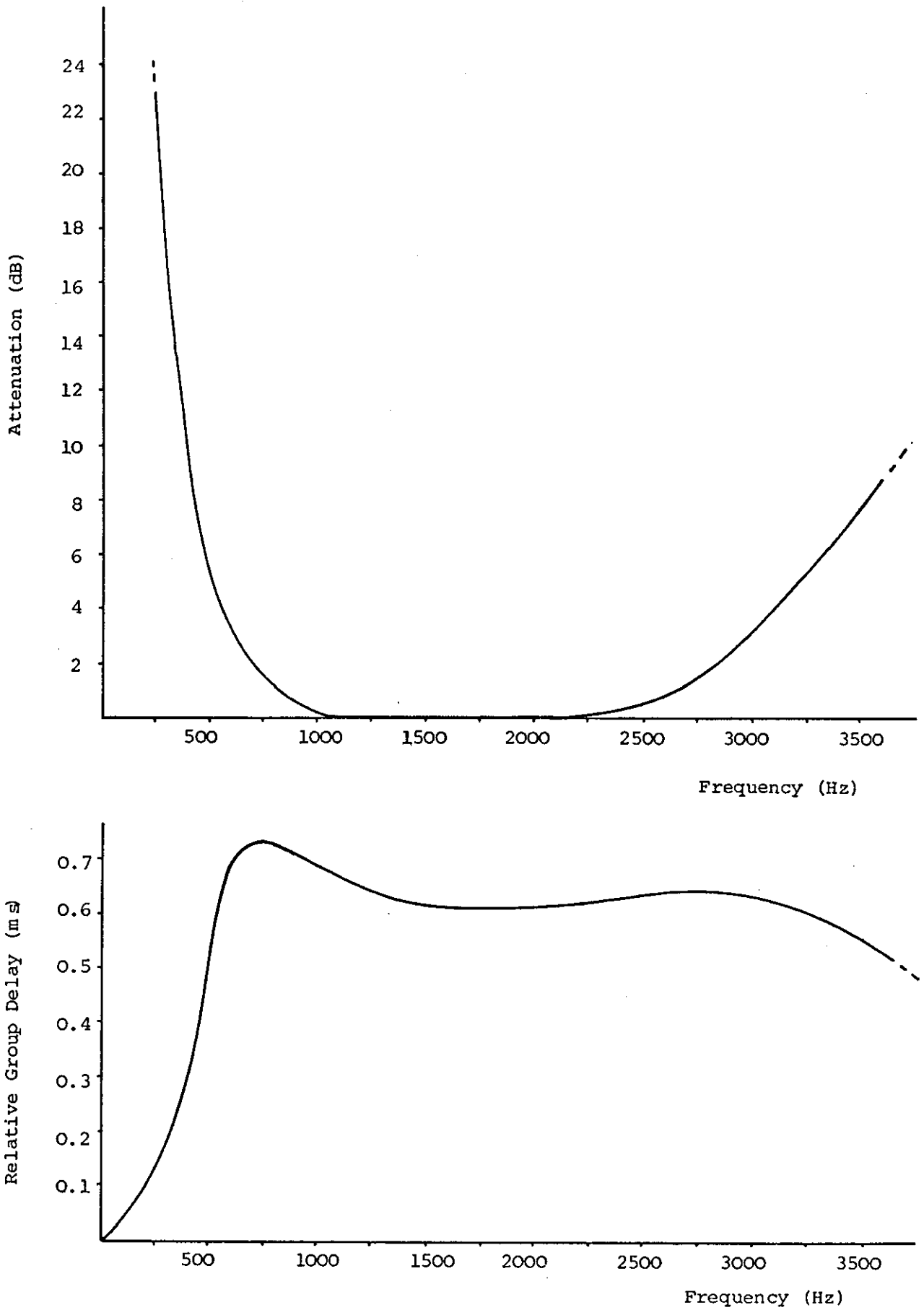


Figure 5.3.8: Attenuation and Group Delay Characteristics for Filter 1c

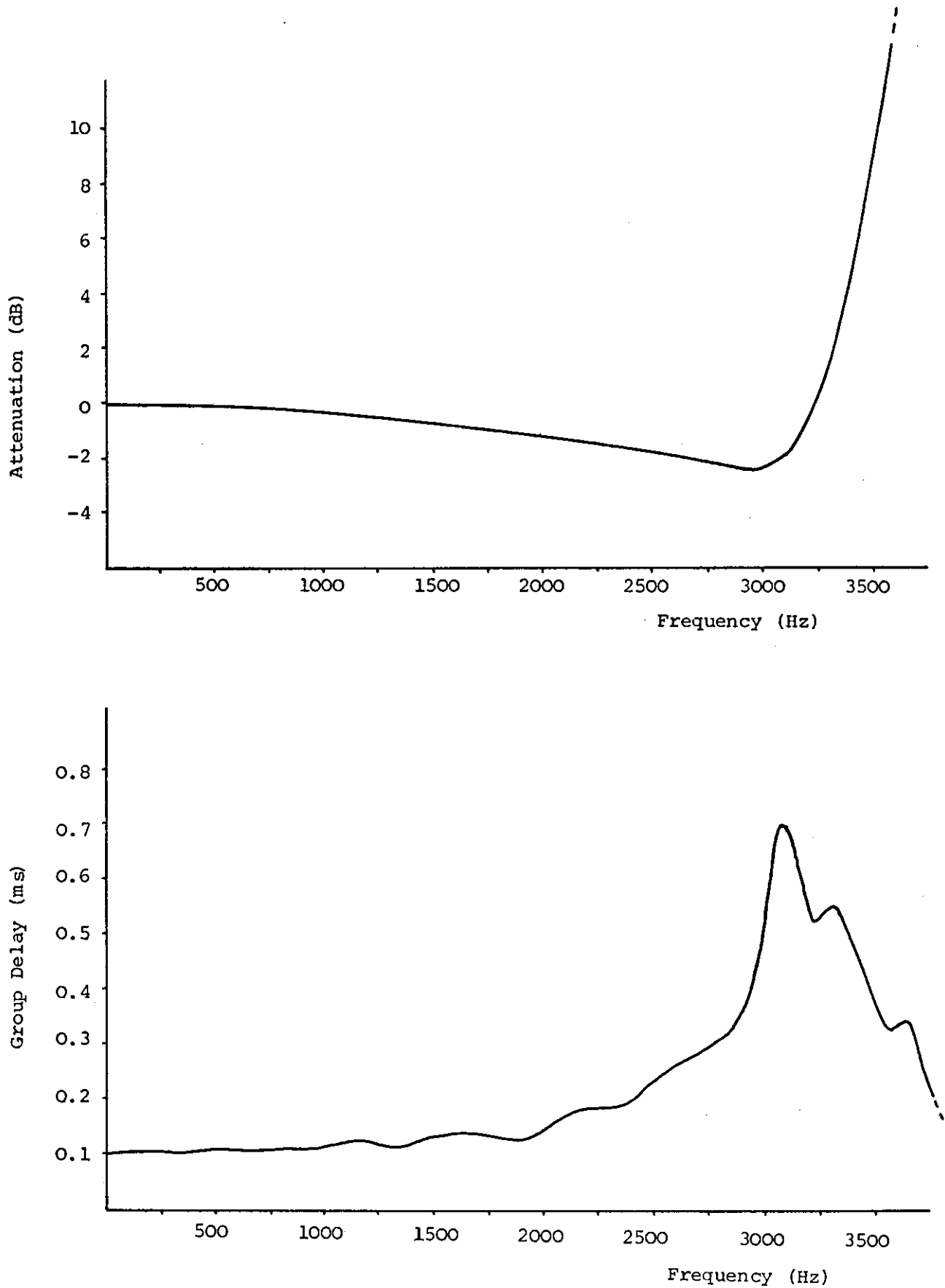


Figure 5.3.9: Attenuation and Group Delay Characteristic for Filter 2a

Clock Range = 2.048 MHz
Clock Frequency = 1.8432 MHz
Digital Sampling Rate = 7.2 kHz

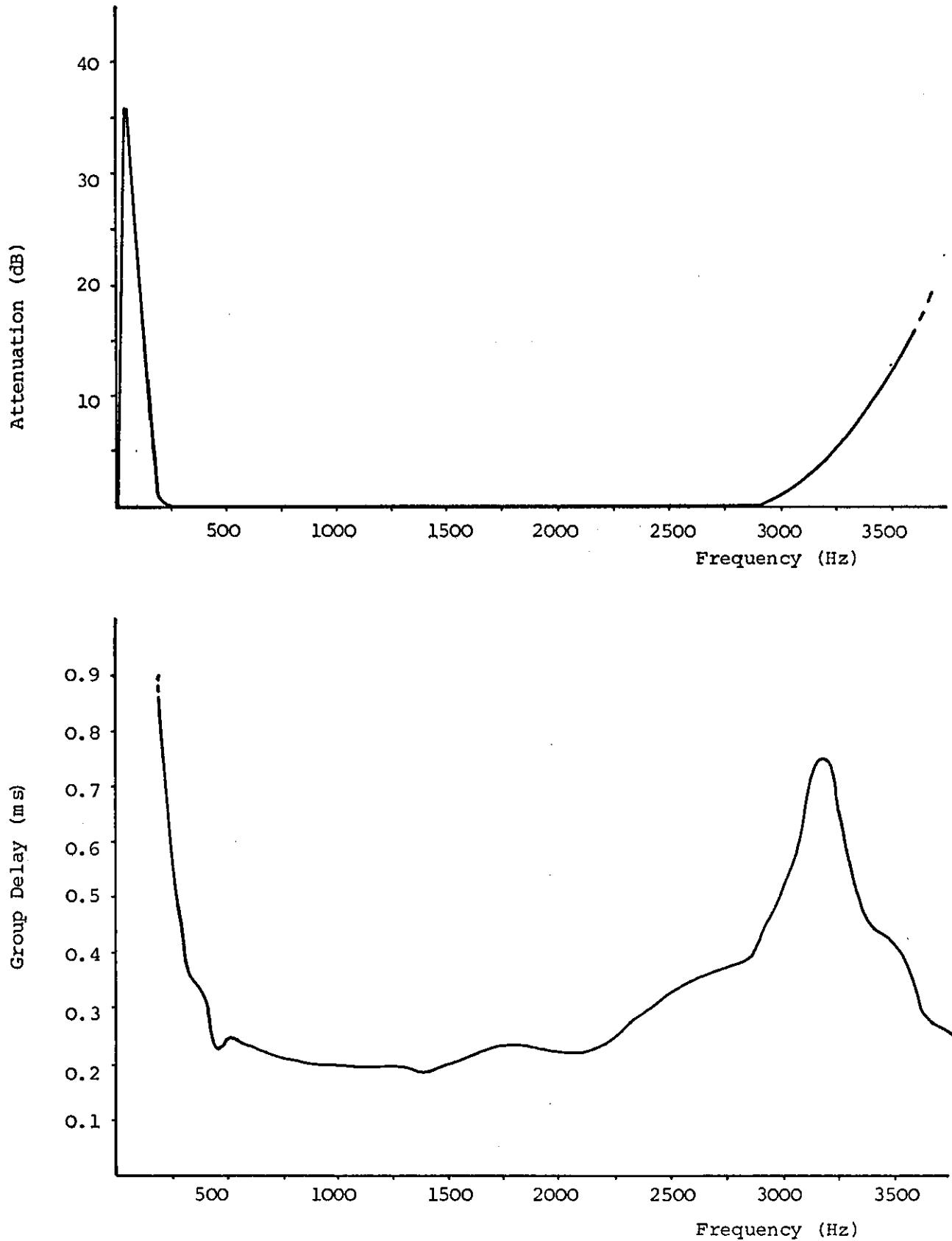


Figure 5.3.10: Attenuation and Group Delay Characteristics for Filter 2c

Clock Range = 2.048 MHz

Clock Frequency = 1.8432 MHz

Digital Sampling Rate = 7.2 kHz

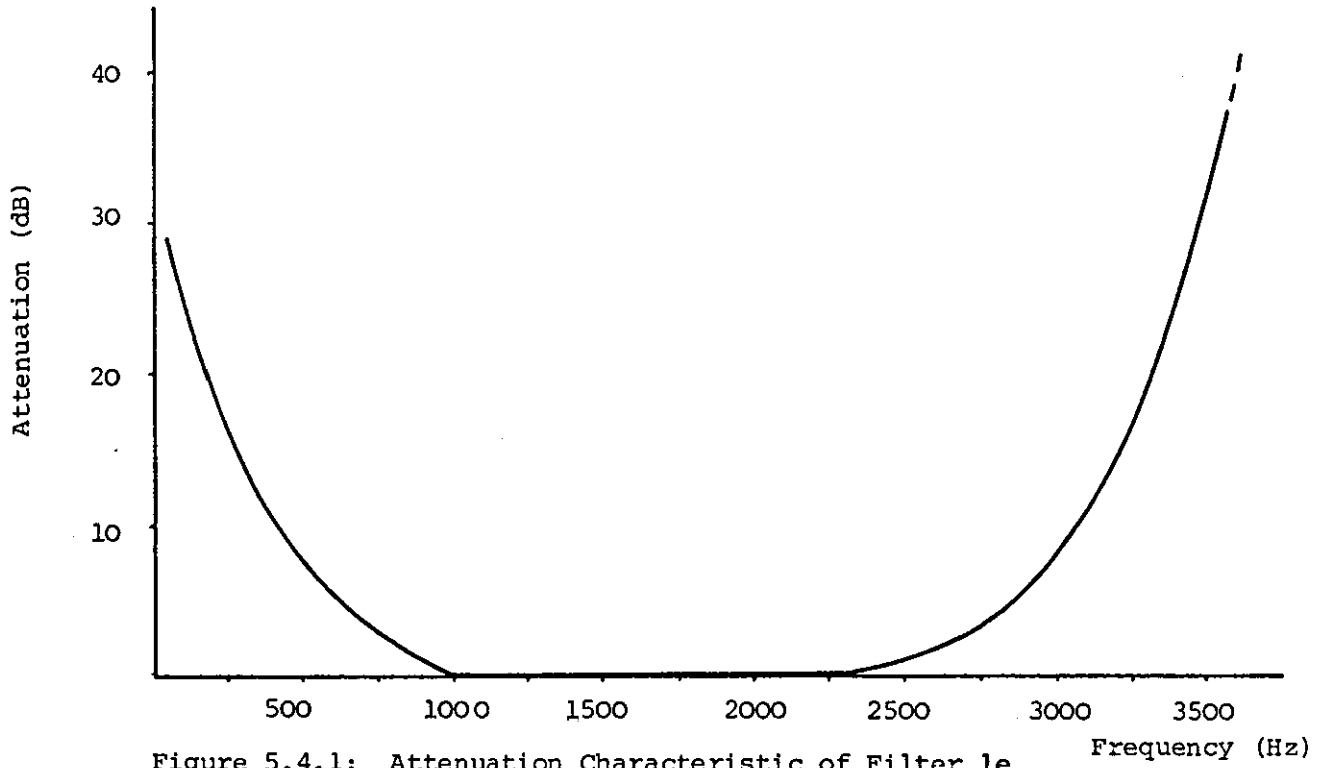


Figure 5.4.1: Attenuation Characteristic of Filter 1e

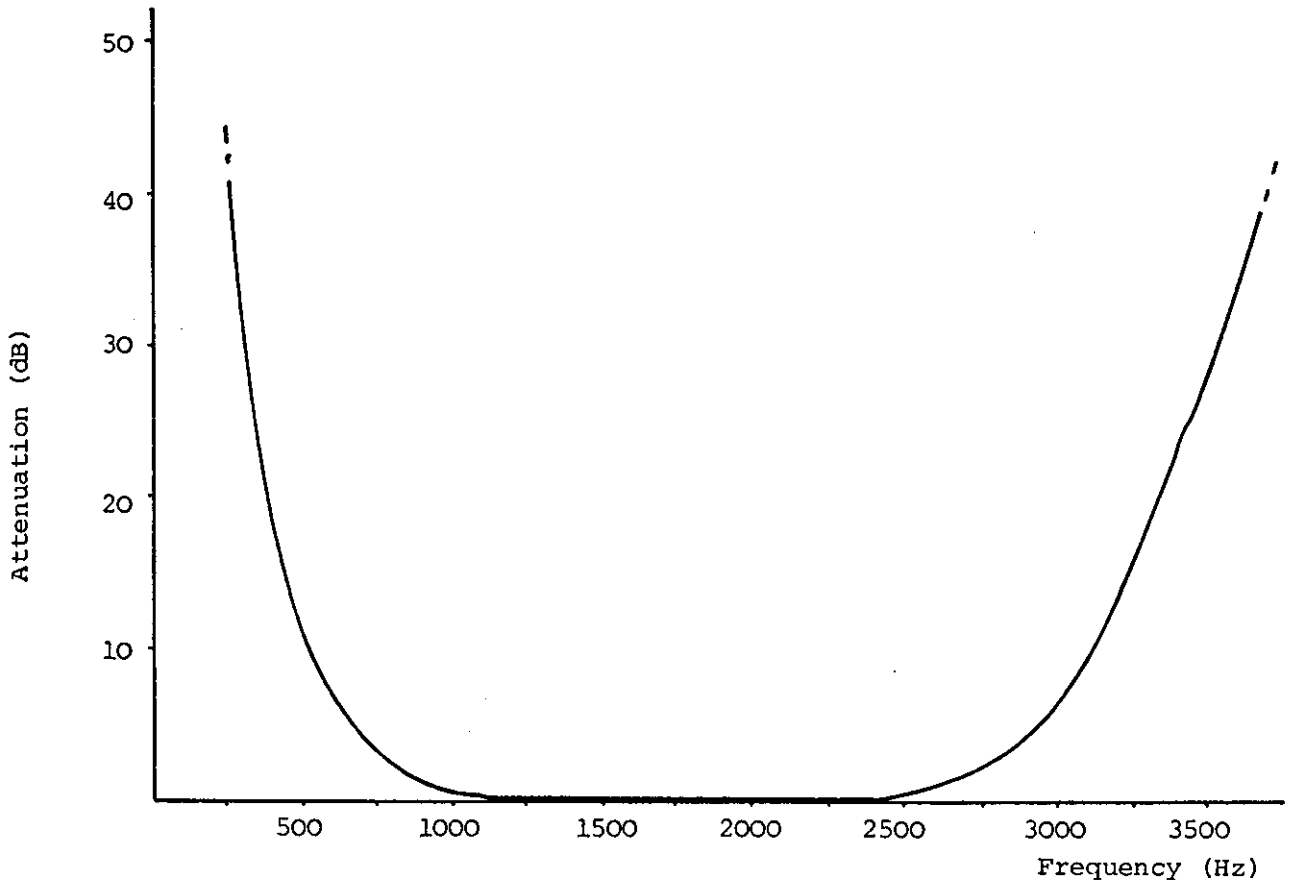


Figure 5.4.2: Attenuation Characteristic of Filter 1f

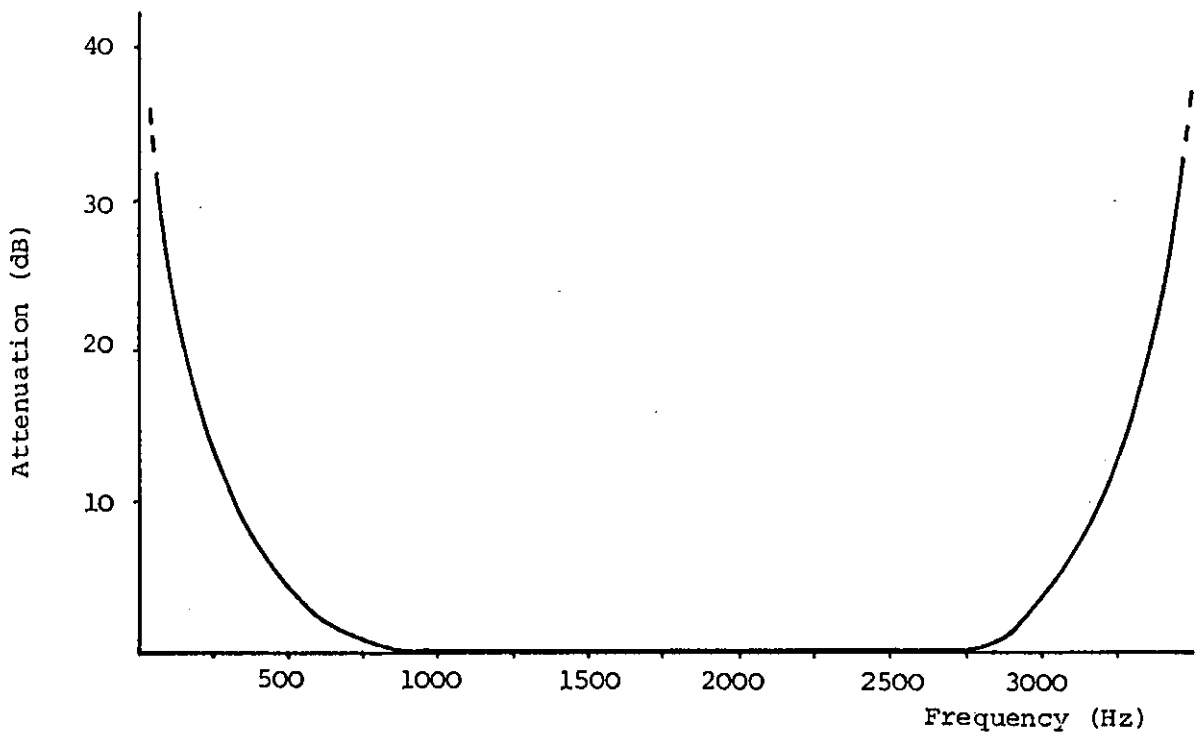


Figure 5.4.3: Attenuation Characteristic of Filter 2e

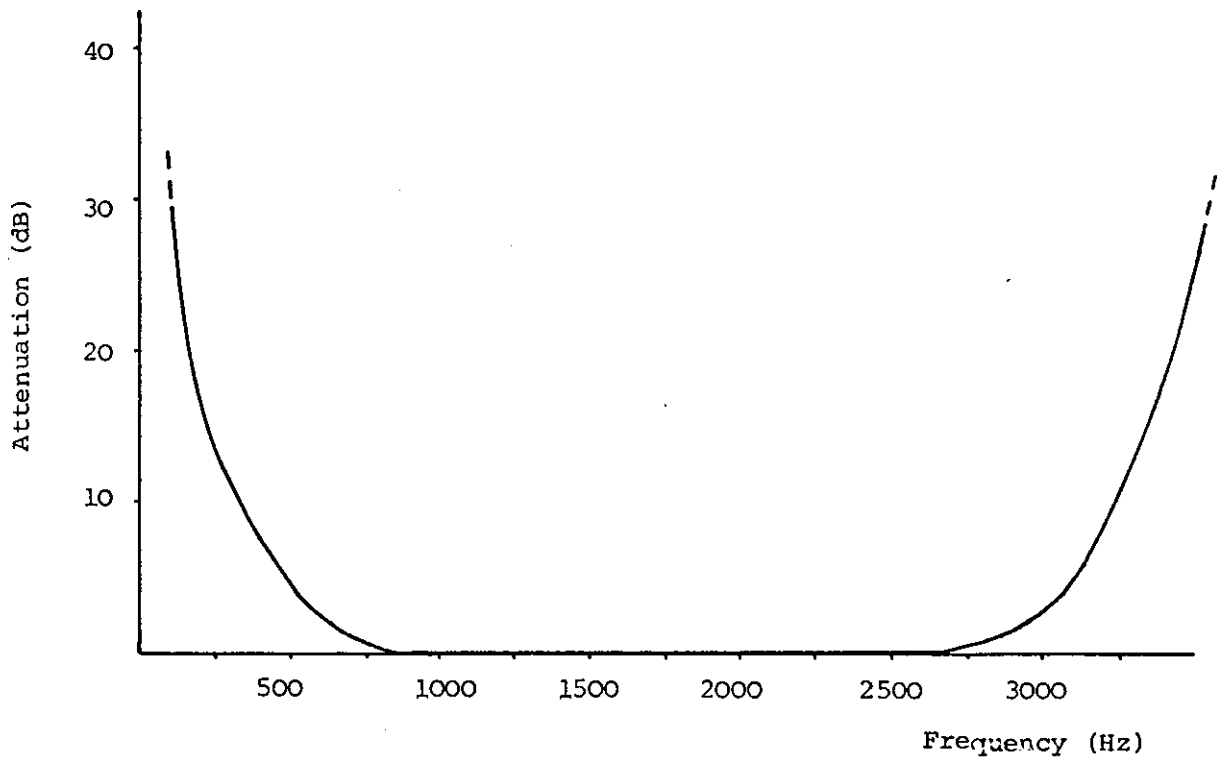


Figure 5.4.4: Attenuation Characteristic of Filter 2f

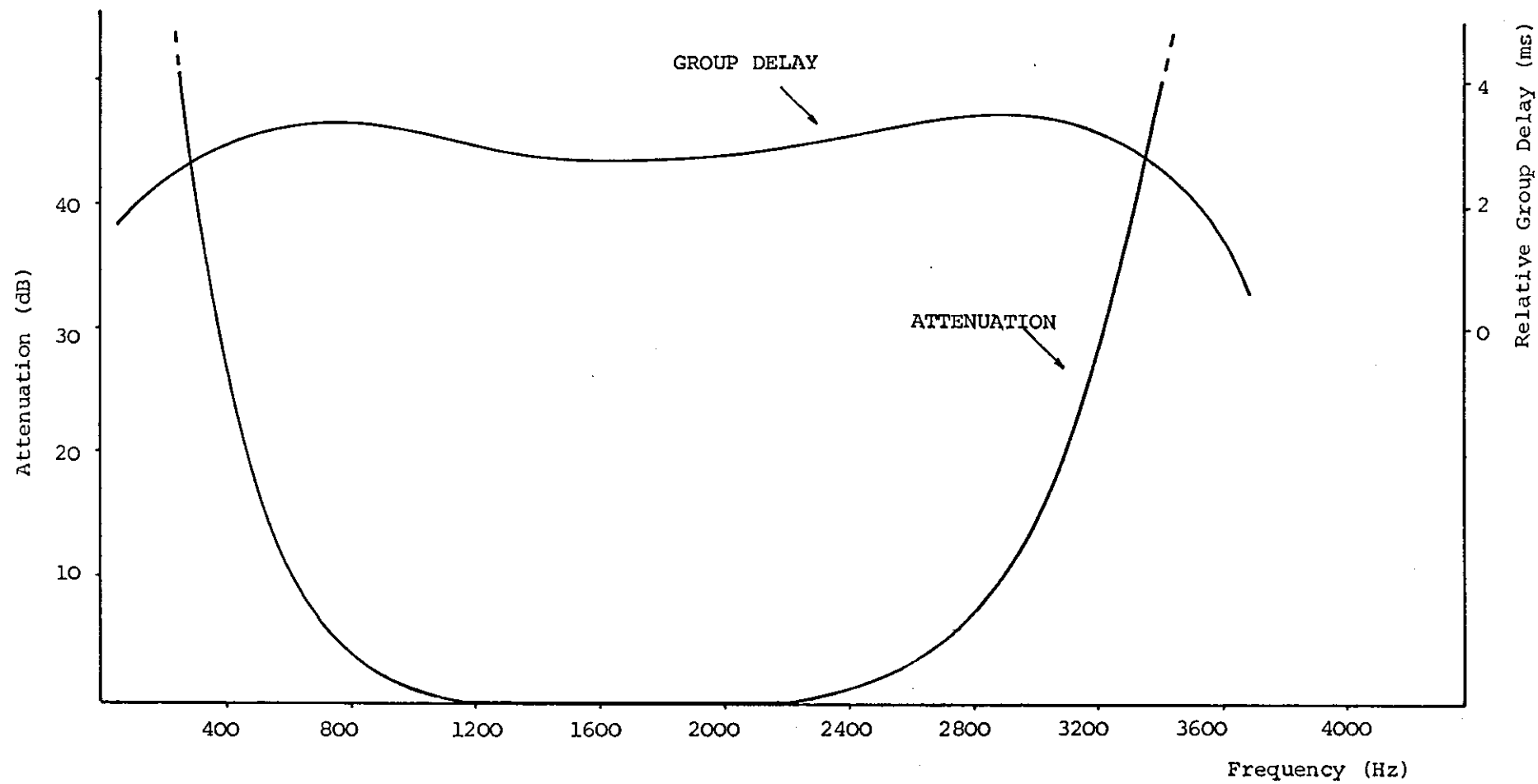


Figure 5.4.5: Attenuation and Group Delay Characteristics for Filter 1g; Combined Filter Set 1

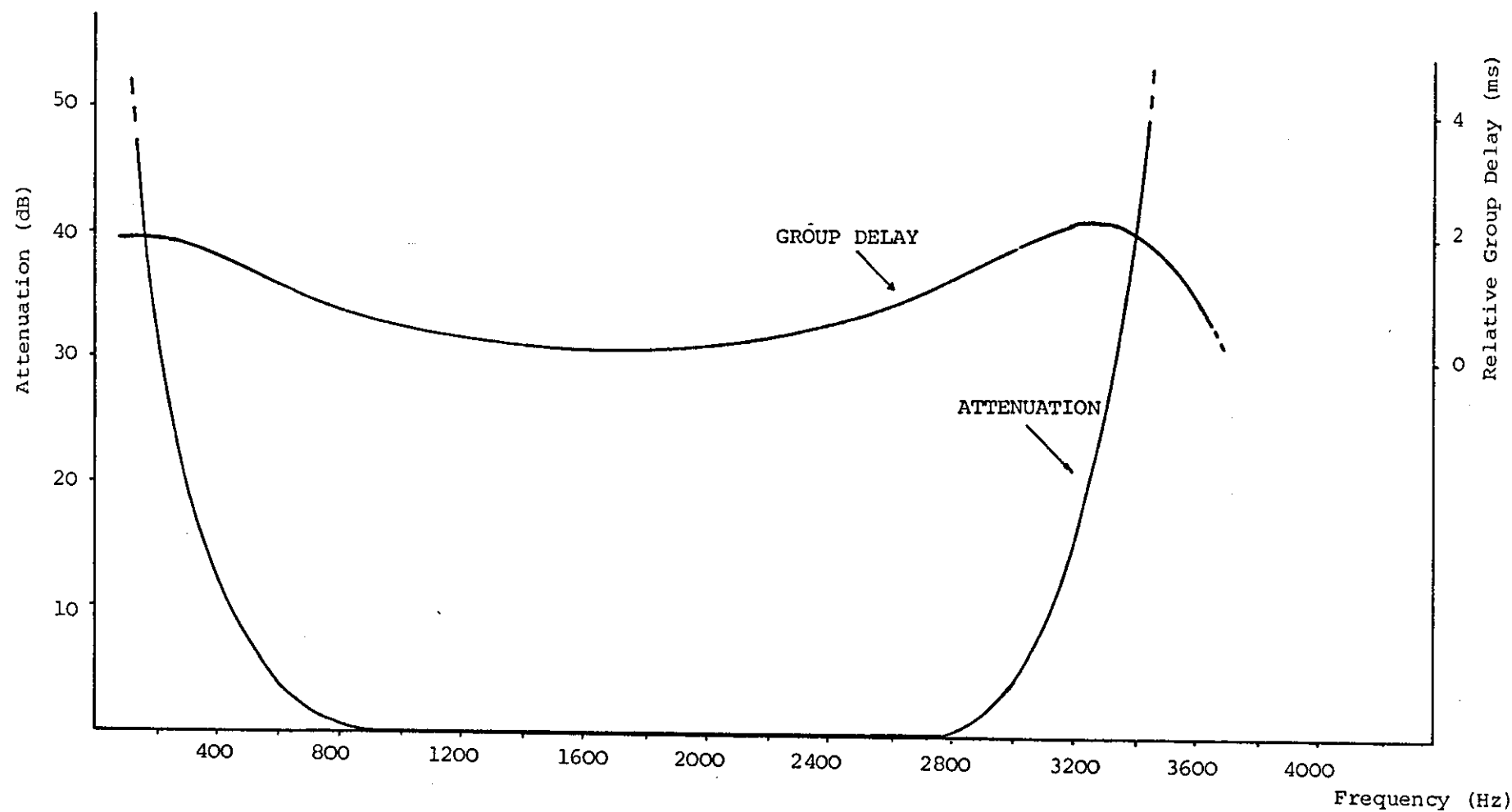


Figure 5.4.6: Attenuation and Group Delay Characteristics for Filter 2g; Combined Filter Set 2

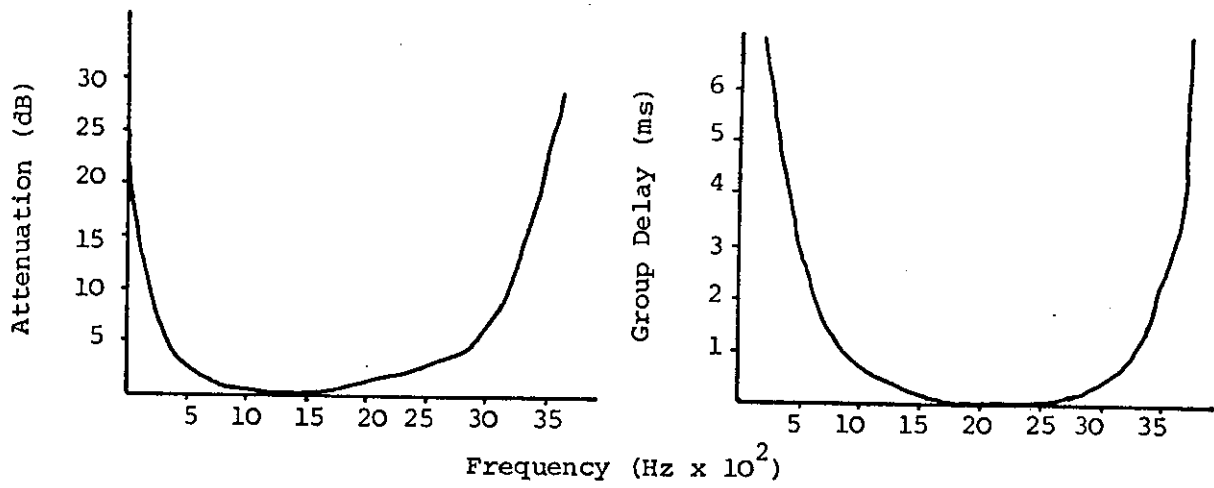


Figure 5.5.1: Characteristics of Telephone Circuit 1

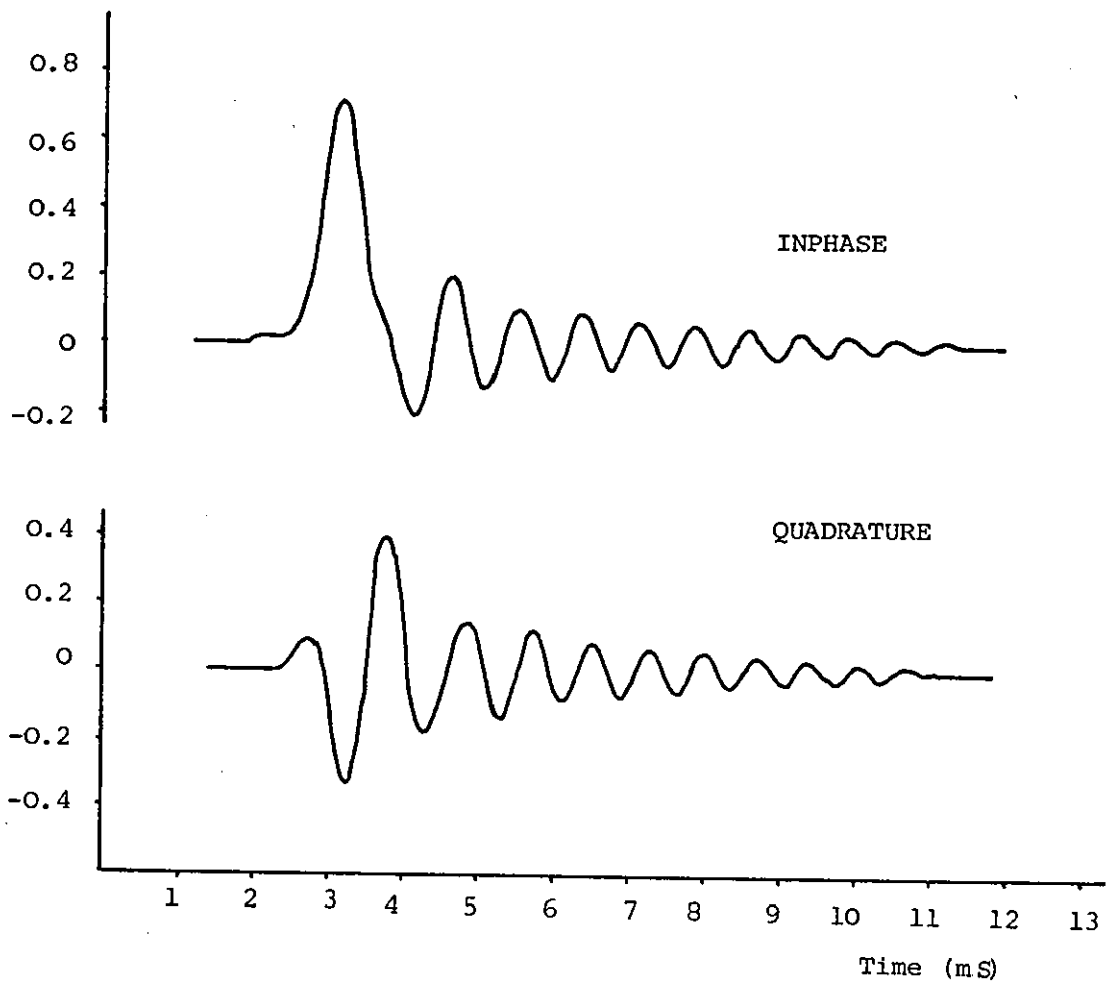


Figure 5.5.2: Impulse Response of Telephone Circuit 1

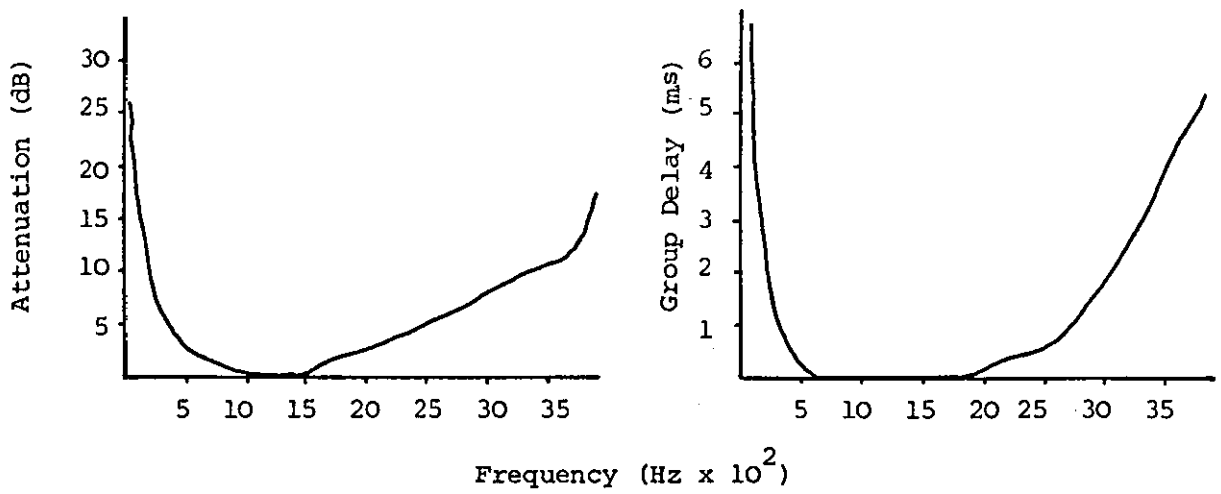


Figure 5.5.3: Characteristics of Telephone Circuit 2

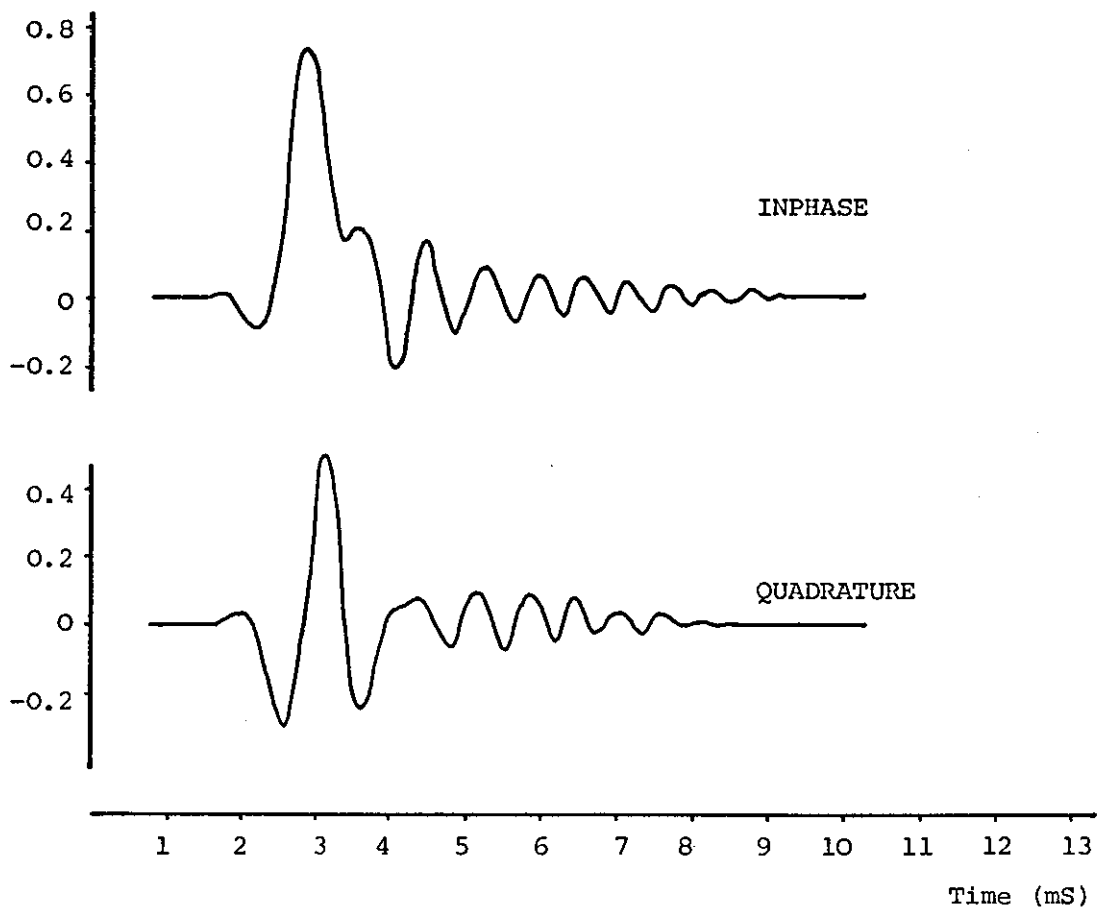


Figure 5.5.4: Impulse Response of Telephone Circuit 2

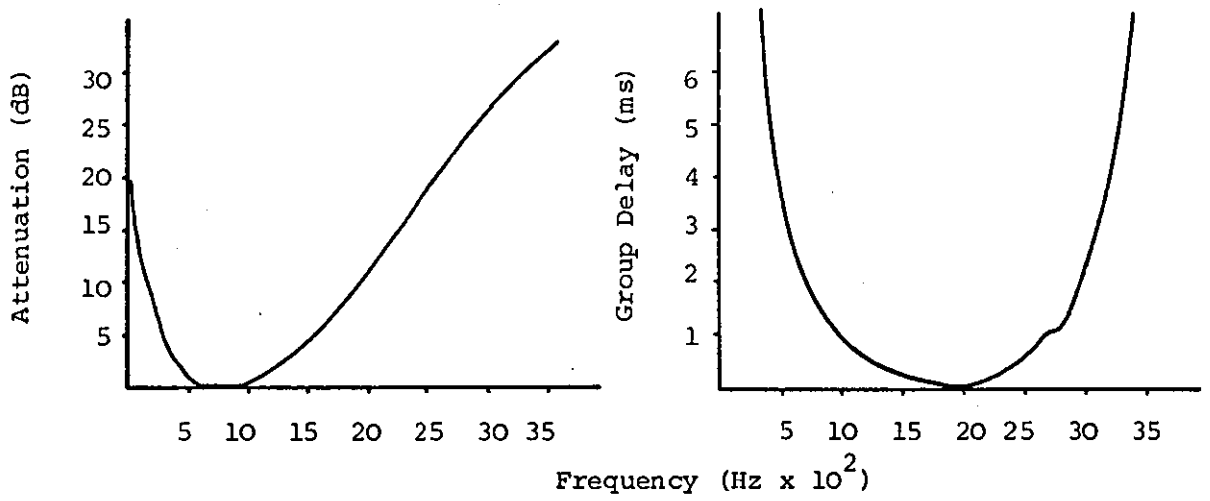


Figure 5.5.5: Characteristics of Telephone Channel 3

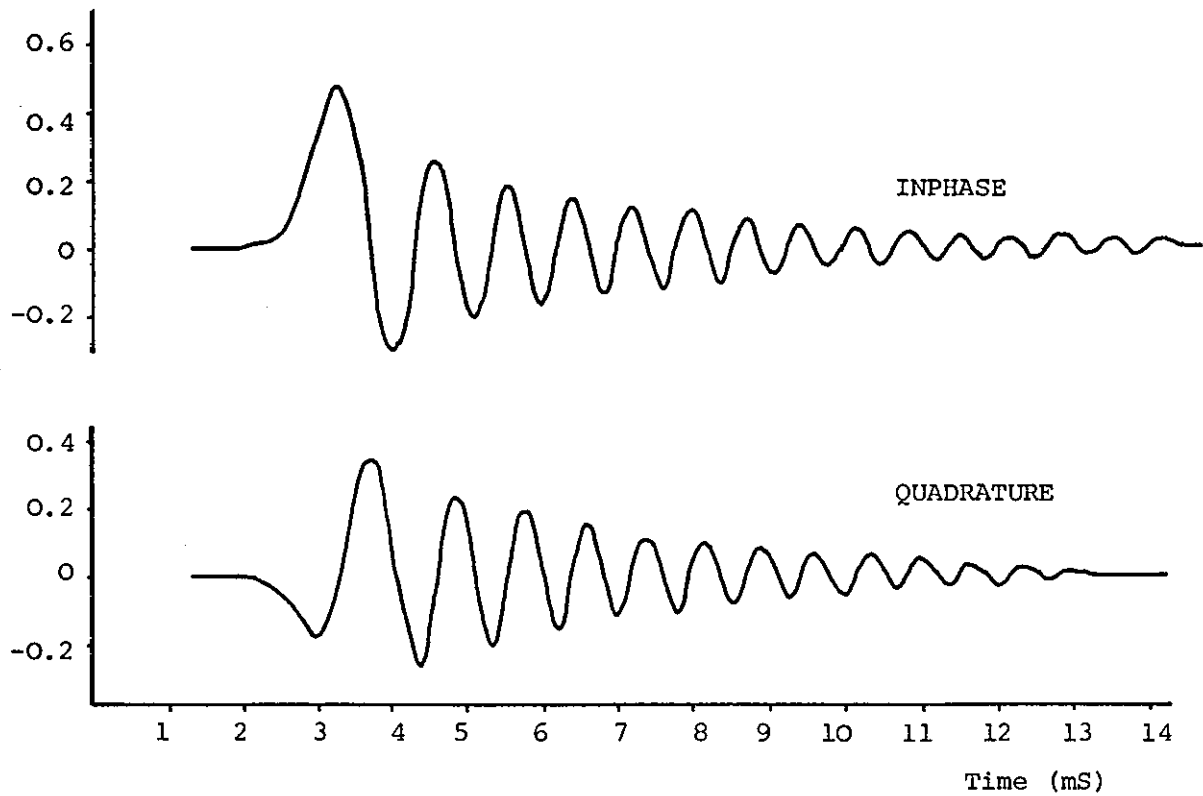
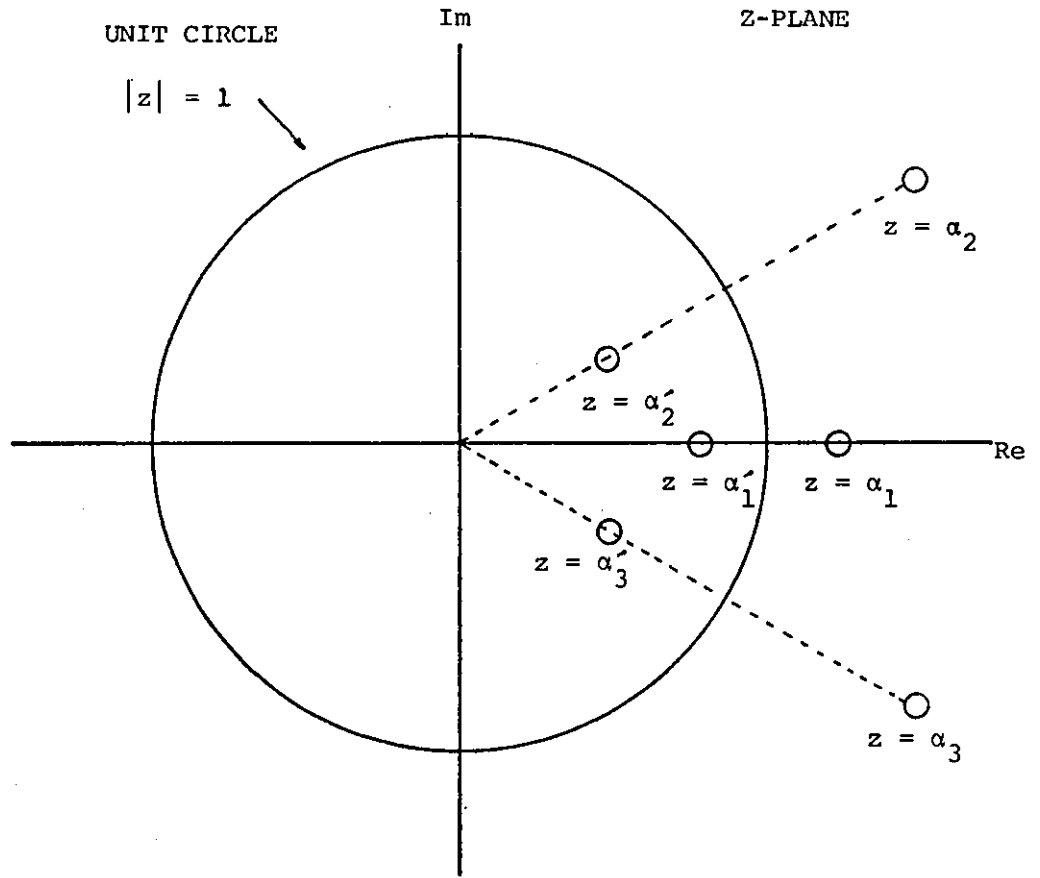


Figure 5.5.6: Impulse Response of Telephone Channel 3



$$\alpha'_1 = \frac{1}{\alpha_1^*} = \frac{1}{\alpha_1}$$

$$\alpha'_2 = \frac{1}{\alpha_2^*}$$

$$\alpha'_3 = \frac{1}{\alpha_3^*}$$

$$|\alpha'_1| = \frac{\alpha_1^*}{|\alpha_1|^2} = \frac{\alpha_1}{|\alpha_1|^2}$$

$$|\alpha'_2| = \frac{\alpha_2}{|\alpha_2|^2}$$

$$|\alpha'_3| = \frac{\alpha_3}{|\alpha_3|^2}$$

α'_1 replaces α_1

α'_2 replaces α_2

α'_3 replaces α_3

$\alpha_2^* = \alpha_3$ in this example

Figure 5.6.1: Illustrating Root Replacement for Minimum Phase Response

6. Detection Processes for 19200 bit/s Modems

6.1 Introduction

In the overview of digital data transmission systems given in Chapter 2, it has been mentioned that the task of the detector or signal processor/detector is to operate on the received signal in some manner and to produce as its output, the detected data symbol s_i' at time $t = iT$ which, according to the detector's knowledge, most closely resembles the transmitted data symbol, s_i . It has also been shown that samples of the received signal taken at times $t=iT$ will, for the case of the practical bandlimited circuits and practically realisable equipment filters discussed in earlier chapters, contain information relating to many transmitted symbols rather than one symbol (that transmitted at time $t=iT$) which is the ideal case. The presence of this intersymbol interference presents two problems to the detector; firstly, it must make a decision as to the value of the transmitted data symbol in the presence of remnants of other symbols and secondly, the energy of the symbol transmitted at time $t=iT$ will be spread over many samples at the input to the detector itself. As the rate of symbol transmission increases, then for a given channel with a particular time invariant impulse response, the number of significant components in the corresponding sampled impulse response will also increase and hence a sample of the received signal will contain more interfering components, as shown in Chapter 5. It is evident that for the successful transmission and reception of data at 19,200 bit/s even over the better private telephone circuits, the receiver must be adaptive in the sense that it takes full account of the distortion present in the received data signal.

In broad terms, two main methods are used in the detection of high-speed data. The first method is one of equalisation, where the detector attempts to remove the effects of distortion from every received sample prior to the decision-making process whilst the second method, maximum-likelihood detection, ideally waits for the arrival of all received samples representing a given transmitted message and detects all symbols comprising the message simultaneously. This detector effectively bases its decision on the whole, rather than part of the symbols' energies. The Chapter first presents and describes a general system model and related basic assumptions and includes comparisons between the ideal and practical realisations of the model for the three systems considered here; system a) and system b) are both 64-point QAM arrangements operating at 3200 bauds but utilizing filter set 1 and 2 respectively, whilst system c) is a 256-point QAM arrangement operating at 2400 bauds using filter set 1.

Following an introduction to the basic concepts of non-linear equalisation and maximum-likelihood detection, the practical implementation of near-maximum-likelihood detectors using the reduced-state Viterbi-algorithm and the requirements of such detectors are considered in detail for application to the above three systems. Finally, the results of computer simulation studies are presented, comparing the tolerances of the three systems to additive noise when operating over telephone circuits 1 to 3 (Chapter 5), when using both the non-linear equaliser and near-maximum likelihood detection schemes.

6.2 General Model and Basic Assumptions

The general model for the 19200 bit/s data transmission systems investigated here is the linear baseband model, shown in Figure 6.2.1.

The input data signal, $s(t)$, where $s(t) = s_a(t) + js_b(t)$, is first filtered by the combined baseband referred transmitter filter with an impulse response and transfer function given by $a(t)$ and $A(f)$, respectively. The shaped data signal is then passed through the transmission path $(\hat{h}(t), \hat{H}(f))$ at whose output, additive white Gaussian noise, $n(t)$, with zero mean and two-sided noise power spectral density of N_0 W/Hz is added to the data signal. After passing through the receiver filter, $(b(t), B(f))$, the received signal is sampled at the system baud rate to obtain received samples which are operated upon to yield the detected data symbol values, $\{s'_i\}$. If $y(t)$ is the impulse response of the linear baseband channel, then

$$y(t) = a(t) * \hat{h}(t) * b(t) \quad (6.2.1)$$

where $*$ denotes the convolution operator, and so;

$$r(t) = \int_{-\infty}^{+\infty} s(\tau) y(t-\tau) d\tau + u(t) \quad (6.2.2)$$

As mentioned in earlier chapters, the analysis of the system can be simplified somewhat by using the discrete-time representation of equation (6.2.2), where the input signal is assumed to be impulses with weights (areas) given by the symbol values at times $t=iT$, hence;

$$s(t) = \sum_i s_i \delta(t-iT) \quad (6.2.3)$$

Since the baseband data signal, $s(t)$, will in practice usually be a rectangular or rounded waveform, the appropriate change must be made to the transmitter filter to ensure that the same signal in both cases is applied to the transmission path. If it is assumed that the baseband signal, $s(t)$, at the transmitter input is a rectangular waveform given by $s(t) = \sum_i s_i v(t-iT)$ where the function $v(t)$ is shown in Figure 6.2.2 and that the required shape of the signal at the output of the transmitter

filter is defined as $q(t)$, then let the impulse response of the filter be $x(t)$. As shown in Figure 6.2.3a), the signal at the output of the filter will be;

$$q(t) = \sum_i s_i v(t-iT) * x(t) \quad (6.2.4)$$

If the input signal is now defined as $s(t) = \sum_i s_i \delta(t-iT)$, then a further filter with an impulse response $p(t)$ will have to be added before the original filter, as shown in Figure 6.2.3 b), such that;

$$q(t) = \sum_i s_i \delta(t-iT) * p(t) * x(t) \quad (6.2.5)$$

Clearly, in this example $p(t) = v(t)$ and the two filters may be combined into a single filter with an impulse response given by $a(t)$, where;

$$a(t) = b(t) * x(t) \quad (6.2.6)$$

as shown in Figure 6.2.3 b). From equations (6.2.5) and (6.2.6), the output from the transmitter filter is, therefore,

$$\sum_i s_i \delta(t-iT) * a(t) = \sum_i s_i a(t-iT) \quad (6.2.7)$$

where the $\{s_i\}$ are assumed to have one of m possible values from the set $(2\ell-1-\sqrt{m}) + j(2k-1-\sqrt{m})$ where ℓ and k are independent variables with possible values $\ell, k = 1, 2, 3, \dots, \sqrt{m}$. \sqrt{m} here is the square root of the signal alphabet and has the value 8 for systems a) and b) and 16 for system c).

In the ideal situation, the transmitter filter will have a flat amplitude response over a frequency range exceeding $\pm \frac{1}{2T}$ Hz, where $1/T$ is the system baud rate. Under these conditions, the data signal may be transmitted, over an ideal transmission path, at T bauds with no intersymbol interference. However, as explained in Chapter 5, the combined effects of channel restrictions and the constraints imposed by

practical filter design necessitate the use of sub-optimum equipment filters. It should be clear from the discussion in Chapter 5 and from Tables 5.6.1 to 5.6.3 that intersymbol interference will occur for all combinations of telephone channels and filter sets considered in this work. Moreover, because of the complex nature of the overall minimum phase sampled impulses given in Tables 5.6.1 to 5.6.3, inter-channel interference will also occur. For example, from Table 5.6.1, if a system is operating over telephone circuit 1 with filter set 1, the received sample at time $t=iT$, where $T = \frac{1}{3200}$ sec, is;

$$r_i = s_i + s_{i-1}(2.0391 + j0.253) + s_{i-2}(0.8687 + j0.4136) + \dots \quad (6.3.8)$$

Expanding r_i and s_i as,

$$r_i = r_{ai} + jr_{bi}, \quad s_i = s_{ai} + js_{bi}, \quad (6.3.9)$$

gives,

$$r_{ai} = s_{ai} + s_{ai-1}(2.0391) - s_{bi-1}(0.253) + s_{ai-2}(0.8687) - s_{bi-2}(0.4136) \quad (6.3.10)$$

and

$$r_{bi} = s_{bi} + s_{ai-1}(0.253) + s_{bi-1}(2.0391) + s_{ai-2}(0.4136) + s_{bi-2}(0.8687) \quad (6.3.11)$$

which clearly shows the occurrence of both intersymbol and interchannel interference. It is informative to note that if telephone circuit 1 operates with filter set 1 but at the lower rate of 2400 bauds (Table 5.6.2), or if telephone circuit 1 operates with filter set 2 at 3200 bauds (Table 5.6.3), the amount of both intersymbol and interchannel interference is reduced, as expected from Chapter 5.

In the case of the example above, the receiver may remove the interfering components present in the received samples $\{r_i\}$, either by linear

equalisation, which essentially attempts to force the sampled impulse response of the baseband channel plus equaliser to be as near ideal as possible or alternatively, by cancelling the terms such as $s_{i-1}(2.0391 + j0.253) + s_{i-2}(0.8687 + j0.4136) + \dots$ from each of the $\{r_i\}$. Both methods obviously involve a knowledge of the sampled impulse response of the baseband channel and in addition, the second method must have a knowledge of the previously detected symbols, s'_{i-1} , s'_{i-2} , ... These equalisation methods, known as linear and non-linear equalisation respectively, will be considered in more detail later but it should be realised that when operating over severely bandlimited telephone circuits such as telephone circuit 3, the length of the sampled impulse response will further increase and so introduce more severe intersymbol interference. This will obviously place extra demands on the equalisers which must also be adaptive to cope with the different transmission paths encountered and any slowly time varying effects. It will be shown later that the maximum-likelihood method of detection also requires a knowledge of the sampled impulse response of the baseband channel and will also involve previously detected symbols in the detection process.

From equations (6.2.1), (6.2.2) and (6.2.7),

$$r(t) = \sum_i s_i y(t-iT) + u(t) \quad (6.2.12)$$

where $u(t)$ is the noise component present in the received signal, $r(t)$, and is given by;

$$u(t) = n(t) * b(t) \quad (6.2.13)$$

Ideally, the receiver filter given by $(b(t), B(f))$ should be matched to the transmitter filter such that when operating over an ideal transmission path, the signal-to-noise ratio at the output of the receiver filter is

maximised. When operating over practical transmission paths which introduce further signal shaping, the receiver filter should ideally be matched to the complete baseband channel which, due to the wide spread of channel characteristics encountered, implies an adaptive receiver filter; this is considered in more detail in a later section.

The noise power spectral density at the output of the receiver filter is,

$$P_u(f) = N_0 |B(f)|^2 \quad (6.2.14)$$

and so the autocorrelation function of $u(t)$ is;

$$\psi_u(\tau) = N_0 \int_{-\infty}^{\infty} |B(f)|^2 e^{j2\pi f\tau} df \quad (6.2.15)$$

Ideally, when data is transmitted over the channel at the Nyquist rate, the receiver filter will have a flat amplitude characteristic and linear phase characteristic over the passband of the signal and a cut-off frequency at half the symbol rate. The output signal from this filter is sampled once per data symbol (therefore at the Nyquist rate). These samples then contain all of the information in the continuous data signal and the corresponding noise samples will be sample values of a Gaussian random variable with zero mean and fixed variance. The noise samples are therefore uncorrelated over time intervals $\tau = nT$ seconds since, from equation (6.2.15);

$$\psi_n(\tau) = \frac{N_0}{T} \frac{\sin \pi \tau / T}{\pi \tau / T} \quad (6.2.16)$$

For convenience, assume that the combined receiver and transmitter filters have a sinusoidal roll-off amplitude characteristic, $|C(f)|$ and that the filtering is shared equally between the two filters. Although these assumptions are not strictly valid here due to the presence of the channel input and output filters and also because the characteristics have been adjusted to be minimum phase, they do simplify the following

analysis. Using the above assumptions,

$$B(f) = C^{\frac{1}{2}}(f) \quad (6.2.17)$$

where $B(f)$ is the transfer function of the combined receiver filter, referred to baseband and $C(f)$ is the transfer function of the combined baseband equivalent transmitter and receiver filters.

$$\text{So, } |B(f)|^2 = |C(f)| \quad (6.2.18)$$

where,

$$|C(f)| = 1, \quad \left\{ \begin{array}{l} 0 < |f| < f_1 - f_r \\ \frac{1}{2} \left[1 - \sin \frac{\pi}{2} \left\{ \frac{|f| - f_1}{f_r} \right\} \right], f_1 - f_r < |f| < f_1 + f_r \end{array} \right. \quad (6.2.19)$$

From equation (6.2.15),

$$\psi_u(\tau) = N_0 \int_{-\infty}^{\infty} |C(f)| e^{j2\pi f\tau} df \quad (6.2.20)$$

Since $|C(f)|$ is a real and even function of frequency,

$$\psi_u(\tau) = 2N_0 \int_0^{\infty} |C(f)| \cos 2\pi f\tau df \quad (6.2.21)$$

so, substituting equation (6.2.19) in equation (6.2.21),

$$\psi_u(\tau) = 2N_0 f_1 \frac{\sin(2\pi f_1 \tau) \cos(2\pi f_r \tau)}{2\pi f_1 \tau} \cdot \frac{1}{\left[1 - (4f_r \tau)^2 \right]} \quad (6.2.22)$$

Clearly, if f_1 is set to half the Nyquist rate and f_r set to zero, the filter reduces to the ideal case considered earlier and equation (6.2.22) is identical to equation (6.2.16). However, for the practical receiver filters considered here, f_1 is less than half of the lowest symbol rate, 2400 bauds, and so the noise samples at the output of the receiver filter will be correlated over time intervals of $\tau = T$ secs, where $T = \frac{1}{3200}$ or $\frac{1}{2400}$ sec.

The received signal given in equation (6.2.12) is then sampled once per symbol period at times $t = iT$, $i = 0, 1, 2, \dots$ so the received sample at

time = kT is,

$$r(kT) = \sum_i s_i y(k-i)T + u(kT)$$

$$r_k = \sum_{h=0}^g s_{k-h} y_h + u_k \quad (6.2.23)$$

where r_k is the k^{th} received sample and y_h is the $(h+1)^{\text{th}}$ component of the sampled impulse response of the linear baseband channel which, for the detection processes discussed here, must be known at the receiver and where Y is the $(g+1)$ component row vector;

$$Y = [y_0 \quad y_1 \quad y_2 \quad \dots \quad y_g] \quad (6.2.24)$$

Equation (6.2.23) is the discrete-time representation of the signal at the input to the detector in Figure 6.2.1; the discrete-time version of the modelled system can therefore be shown diagrammatically as a linear feedforward transversal filter with tap coefficients given by the components of Y and with noise samples, u_k , added at its output, as shown in Figure 6.2.4.

6.3 Equalisation Techniques

6.3.1 Linear Equalisation

Figure 6.3.1 shows the basic form of a detection system employing a linear equaliser and threshold detector. The equaliser studied here is assumed to be a linear feedforward transversal filter which is probably the most widely studied form of linear equalisation; it has been shown elsewhere that such an arrangement is optimum when only linear processing of the signal is introduced prior to detection⁽¹⁻⁴⁾.

The linear feedforward transversal equaliser shown in Figure 6.3.2 has $(l+1)$ taps, $c_0, c_1 \dots c_l$, and operates on the sequence of received samples, $\{r_i\}$. The sample values shown in Figure 6.3.2 are those present at the time instant $t=iT$. Each received sample, $r_i, r_{i-1}, r_{i-2}, \dots$

is held in a store marked "T" and so at the arrival of sample r_{i+1} , each sample is shifted one place to the right. The output from each store is multiplied by the appropriate tap value and then summed to give an output of time $t=iT$ as;

$$x_i = \sum_{j=0}^{\ell} r_{i-j} c_j \quad (6.3.1)$$

The transversal filter has a sampled impulse response given by the $(\ell+1)$ component row vector;

$$C = [c_0 \quad c_1 \quad \dots \quad c_{\ell}] \quad (6.3.2)$$

which has the corresponding z-transform,

$$C(z) = c_0 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_{\ell} z^{-\ell} \quad (6.3.3)$$

If the sampled impulse response of the linear baseband channel preceding the equaliser is the $(g+1)$ component vector,

$$Y = [y_0 \quad y_1 \quad y_2 \quad \dots \quad y_g] \quad (6.3.4)$$

with a z-transform,

$$Y(z) = y_0 + y_1 z^{-1} + \dots + y_g z^{-g} \quad (6.3.5)$$

then the sampled impulse response of the equalised channel will be the $(\ell+g+1)$ component vector F , which has the z-transform,

$$F(z) = \sum_{i=0}^{\ell+g} f_i z^{-i}$$

$$\text{where } F(z) = Y(z)C(z) \quad (6.3.6)$$

In the absence of noise, the z-transform of the received sequence $\{r_i\}$ may be written^(2,3),

$$R(z) = S(z)Y(z) \quad (6.3.7)$$

where
$$R(z) = \sum_{i=1}^p r_i z^{-i}$$

$$Y(z) = \sum_{i=0}^g y_i z^{-i}$$

and
$$S(z) = \sum_{i=1}^q s_i z^{-i}$$

From equation (6.3.1) and (6.3.7), the z-transform of the sequence at the output of the equaliser will be,

$$X(z) = R(z)C(z) \quad (6.3.8)$$

and from equations (6.3.6) and (6.3.7),

$$X(z) = S(z)Y(z)C(z)$$

$$X(z) = S(z)F(z) \quad (6.3.9)$$

For the exact equalisation of the channel, the vector F must have only a single non-zero component whose value must be unity,

$$F = \underbrace{[0 \ 0 \ 0 \ \dots \ 0]}_h \ 1 \ 0 \ 0 \ 0 \ \dots \ 0] \quad (6.3.10)$$

where h is some non-negative integer in the range 0 to $(\ell+g+1)^{(2)}$ and represents a delay of h sample intervals in the equalised sampled impulse response. In this case, $F(z)$ reduces to

$$F(z) = z^{-h} \quad (6.3.11)$$

and so from equation (6.3.6),

$$C(z) = z^{-h} Y^{-1}(z) \quad (6.3.12)$$

Since $Y(z)$ is a polynomial in z^{-1} of order g , then $Y^{-1}(z)$ will be a polynomial in z^{-1} with an infinite number of coefficients. Consequently, exact equalisation cannot be achieved in practice using a non-recursive equaliser structure. However, as the coefficients of $Y^{-1}(z)$ decrease as the order of the polynomial increases, approximate equalisation can be achieved, the accuracy of equalisation being dependent on the number of taps used in the equaliser. Equalisers with fewer taps in general

result in incomplete equalisation and larger residual interference terms are present at the output of the equaliser.

The actual determination of the equaliser's tap gains, $\{c_i\}$, $i = 0, 1, 2, \dots, \ell$, can be achieved in several ways depending on different criteria used to minimise the amount of intersymbol interference present in the sample at the output of the equaliser and are described in detail in references 1-4. Clearly, if the sampled impulse response of the channel changes over time, the equalisation process must be adaptive.

If additive noise is now added to the received samples, the output from the equaliser at time $t=iT$, is, from equation (6.3.1),

$$x_i = \sum_{j=0}^{\ell} r_{i-j} c_j + w_i \quad (6.3.13)$$

$$\text{where } w_i = \sum_{j=0}^{\ell} u_{i-j} c_j \quad (6.3.14)$$

and where u_i is the i^{th} noise sample at the equaliser input. Consequently, from equations (6.3.9), (6.3.12) and (6.3.13), the sample at the input to the detector at time $t=(i+h)t$ is,

$$x_{i+h} = s_i + w_{i+h} \quad (6.3.15)$$

If the $\{u_i\}$ are statistically independent Gaussian random variables with zero mean and variance σ^2 , then the $\{w_i\}$ are Gaussian random variables with zero-mean and variance $^{(3,4)}$,

$$\eta^2 = \sigma^2 \sum_{j=0}^{\ell} c_j^2 = \sigma^2 |C|^2 \quad (6.3.16)$$

where $|C|$ is the length of the vector C and where, for the systems considered here, the components of C are complex-valued. It is interesting to note that if the channel introduces any amplitude distortion, $|C|^2 > 1$ and

so the variance of the noise passing through the equaliser is enhanced^(4,5). However, if the channel introduces only pure phase distortion, the linear equaliser performs pure phase equalisation and the linear equaliser followed by a suitable threshold detector (two dimensional, multi-level for systems a), b) and c)), achieves the best tolerance to additive white Gaussian noise. Under these conditions, the equaliser is also matched to the channel⁽⁴⁾. If it is assumed that the equaliser is accurately adjusted, then the average probability of error for systems a), b) and c) can be found from the analysis given in Appendix 5, where the noise variance is replaced by $\sigma^2 |C|^2$.

6.3.2 Non-Linear Equalisation

A considerable amount of research has been carried out over the past few years on non-linear equalisation methods applied to high-speed data transmission systems; references 3 and 4 list a large number of recent publications in this area. Due to this research effort, the non-linear equaliser has become the conventional method of high-speed data detection and so is studied here to set comparative standards for the more sophisticated detection processes developed later in this work.

The basic form of non-linear equalisation shown in Figure 6.3.3 a) uses previously detected data symbols and a knowledge of the channel's sampled impulse response to form estimates of the intersymbol interference components and then cancels these components from the received signal. The receiver therefore operates by quantised feedback correction, removing the intersymbol interference from the detector's input signal^(3,4).

From equation (6.2.23),

$$r_i = \sum_{j=0}^g s_{i-j} y_j + u_j$$

$$r_i = s_i y_0 + \sum_{j=1}^g s_{i-j} y_j + u_j \quad (6.3.17)$$

where $\sum_{j=1}^g s_{i-j} y_j + u_j$ represents the intersymbol interference term present in r_i . If the receiver is assumed to have a knowledge of the sampled impulse response of the channel and has correctly detected the data symbols $\{s_{i-j}\}$ such that $\{s'_{i-j}\} = \{s_{i-j}\}$, $j = 1, 2, \dots, g$, then the receiver can calculate the intersymbol interference term given above by passing the $\{s'_{i-j}\}$ through a linear transversal filter with tap gains given by y . From Figure 6.3.3 b), the input sample to the detector at time $t=iT$ is,

$$x_i = \frac{1}{y_0} \left(r_i - \sum_{j=1}^g s'_{i-j} y_j \right) \quad (6.3.18)$$

$$= \frac{1}{y_0} \left(s_i y_0 + \sum_{j=1}^g s_{i-j} y_j - \sum_{j=1}^g s'_{i-j} y_j + u_i \right) \quad (6.3.19)$$

which, provided that $s'_i = s_i$, for each i here, reduces to

$$x_i = s_i + \frac{u_i}{y_0} \quad (6.3.20)$$

The detector then operates on x_i to determine the value of s_i which is closest to the value of x_i , the process being repeated for each x_i . The variance of the noise component in x_i is given by $\sigma^2/|y_0|^2$ where σ^2 is the variance of the $\{u_i\}$ which are assumed to be statistically independent Gaussian random variables with zero mean values. The average probability of error in a detected symbol now reduces to that given in Appendix 5, but where σ^2 is replaced with $\sigma^2/|y_0|^2$. It can be seen from equations (6.3.18) and (6.3.20) that both r_i/y_0 and x_i have the same wanted signal component, s_i , and the same noise component u_i/y_0 , so the non-linear equaliser removes intersymbol interference without changing the signal-to-noise ratio. Clearly, if the detector makes an erroneous decision for any of the $\{s_i\}$, then its interference in the following elements will not be

eliminated. Errors therefore tend to occur in bursts and the method suffers from error-extension effects. However, at high signal-to-noise ratios and if y_0 is one of the larger components in the sampled impulse response, the error extension effects are not normally serious and do not reduce noise tolerance by more than about 1dB^(4,5).

It is shown in references 4 and 5 that a linear equaliser is the optimum detection process when the channel introduces only pure phase distortion whereas a non-linear equaliser of the type considered above performs better than a linear equaliser in the presence of noise if all the roots of the z-transform of $Y(z)$ lie on or within the unit-circle of the z-plane. However, practical channels can have many roots outside the unit circle, so the non-linear equaliser can often give a lower tolerance to additive noise than a linear equaliser. It appears therefore, that for the channels considered here, the combination of a linear equaliser to equalise phase distortion followed by a non-linear equaliser which performs decision-feedback equalisation, will in general give a better tolerance to noise than either of the two equalisers used individually.

The non-linear equaliser now appears as a linear feedforward transversal filter followed by a pure decision-feedback equaliser, as shown in Figures 6.3.4 and 6.3.5. It has been shown elsewhere that the optimum non-linear equaliser is one which shares the equalisation between the two filters such that the linear filter replaces all roots of $Y(z)$ which lie outside the unit circle by the complex conjugates of their reciprocals and therefore performs a pure phase transformation; it also forces the sampled impulse response of the channel and linear filter itself to be

\propto minimum phase^(3,4,5). As discussed in Chapter 5, the minimum phase version of a sampled impulse response will have a rapid rise time and a reduced number of components, so the majority of the energy of the sequence will appear within the first few components. Clearly, this reduces the length of the transversal filter required in the cancellation of intersymbol interference process. Also, it can be shown⁽⁴⁾ that the signal-to-noise ratio at the output of the linear filter is maximised. As the signal-to-noise ratio is unchanged across the pure decision-feedback equaliser, the signal-to-noise ratio at the input to the detector will also be maximised and so, subject to accurate equalisation, this combination is optimum in terms of its tolerance to additive white Gaussian noise. Clearly, for accurate equalisation, the detector must have full knowledge of the sampled impulse response of the channel and be able to adjust the linear filter accordingly. For a system to operate over the switched telephone network, the equalisers must be adaptive to cope with the varying conditions and more importantly, with the wide spread of characteristics encountered when operating over different connections. The detailed description of the adaptive adjustment of the equalisers and the estimation of the channel sampled impulse response is deferred until Chapter 7. As far as this Chapter is concerned, it is sufficient to know that these conditions can be met in practice⁽¹⁻⁵⁾. Indeed, the linear filter described above performs the very function mentioned in Chapter 5, where the sampled impulse responses of the various telephone circuit and filter set combinations were presented in their minimum phase form. The performance results presented in Section 6.6 use these sampled impulse responses and therefore assume that the linear filter (or "pre-filter") is perfectly adjusted.

6.4 Maximum-Likelihood Detection Processes

A brief description and simple example of maximum-likelihood detection has been presented in Section 2.1 to illustrate the basic principles of operation. To recap, the maximum-likelihood method waits for all of the received samples, $\{r_i\}$ of a given message to be received before commencing the detection process which detects all symbols within the message simultaneously. The detector is assumed to have a knowledge of the m possible values of the $\{s_i\}$, the length of the message, l , and the sampled impulse response of the channel, Y , where Y is taken to have $(g+1)$ components. The receiver holds in store the m^l possible sequences of data symbols that could have been transmitted and forms the m^l sequences comprising the convolution of the possible data sequences with the sampled impulse response of the channel. These latter sequences therefore represent possible received sequences for the given signal distortion, but in the absence of noise. The detector then compares each of these sequences with the actual sequence of received samples and selects as the detected sequence, the sequence of data symbols which results in the minimum mean-square difference. When the transmitted signal elements are statistically independent and equally likely to have any of their possible values and the noise samples are statistically independent Gaussian random variables, then the detected sequence is also the maximum-likelihood sequence which minimises the probability of error in the received message. Furthermore, at high signal-to-noise ratios, maximum-likelihood detection can be taken to minimise the average probability of error in the detection of an individual data symbol^(3-5,8,14-15). In the example given in Section 2.1, the message length was 4 symbols and the sampled impulse response had only 2 components; this resulted in 16

possible symbol sequences to be stored at the receiver along with the 16 sequences, each having 5 components, comprising the convolution of the possible symbol sequences with the sampled impulse response. The computational effort and storage requirements for this example are therefore relatively manageable. However, as the length of message, signal alphabet and length of sampled impulse response increase, both computation effort and memory size increase enormously, making pure maximum-likelihood detection totally impractical for real systems. Moreover, the use of complex symbols and the complex nature of the sampled impulse responses considered in this work further increases the complexity of the detector. Before considering methods of reducing the complexity of such detection processes, maximum-likelihood detection using the Viterbi algorithm is introduced to illustrate the practical difficulties associated with these processes^(3,4,5,8-15).

6.4.1 The Viterbi Algorithm Detector

The optimum performance maximum-likelihood detector uses the Viterbi algorithm for its implementation⁽¹¹⁾. In the context of data detection in the presence of intersymbol interference and uncorrelated additive Gaussian noise samples with zero mean values, the Viterbi algorithm detector operates as follows:

From equation (6.2.23), the j^{th} received signal sample is,

$$r_j = \sum_{h=0}^g s_j - h y_h + w_j \quad (6.4.1)$$

where the $\{r_j\}$, $\{s_j\}$, $\{y_j\}$ and $\{w_j\}$ are in general complex-valued. In equation (6.4.1), w_i is used to represent the noise samples rather than u_i in equation (6.2.23) to illustrate that the noise samples here are

considered to be statistically independent.

Let S_j , R_j and W_j be the j -component row vectors representing the sequences s_i , r_i and w_i , respectively, for $i=1,2, \dots j$. Also, let X_j and N_j be the j -component row vectors whose i^{th} components are x_i and n_i respectively, for $i=1,2, \dots j$, where x_i has one of the m possible values of s_i and n_i is the value of w_i satisfying;

$$x_j = \sum_{h=0}^g x_{j-h} y_h + n_j \quad (6.4.2)$$

S_j and X_j have m^j possible combinations of $\{s_i\}$ and $\{x_i\}$, $i=1,2, \dots j$, respectively.

The maximum-likelihood vector, X_j is the possible vector X_j for which

$$|N_j|^2 = |n_1|^2 + |n_2|^2 + \dots |n_j|^2 \quad (6.4.3)$$

is minimised.

Just prior to the receipt of sample r_{j+1} , the detector holds in store the m^g possible maximum-likelihood vectors, $\{X_j\}$, corresponding to the m^g possible combinations of allowed data symbol values, x_{j-g+1} , x_{j-g+2} , $\dots x_j$. A maximum-likelihood vector here is the possible value of X_j that minimises $|N_j|^2$, called the "cost" of X_j , subject to the constraint that the x_{j-g+1} , x_{j-g+2} , $\dots x_j$ have allowable values of $\{s_i\}$. Associated with each of the m^g stored vectors, $\{X_j\}$, is the corresponding cost term $|N_j|^2$, where each n_i satisfies equation (6.4.2). The true maximum-likelihood vector is the vector X_j whose cost function, $|N_j|^2$, is minimum.

On receipt of sample r_{j+1} , each of the stored vectors $\{X_j\}$ form the common part of m vectors $\{X_{j+1}\}$, each of which has the associated value of

$$|N_{j+1}|^2 = |N_j|^2 + |r_{j+1} - \sum_{h=0}^g x_{j-h+1} y_h|^2 \quad (6.4.4)$$

where,

$$r_{j+1} - \sum_{h=0}^g x_{j-h+1} y_h = n_{j+1} \quad (6.4.5)$$

Clearly, the $\{|N_{j+1}|^2\}$ can be determined from the appropriate stored values $\{|N_j|^2\}$. For each of the m^g possible combinations of x_{j-g+2} , x_{j-g+3} , ..., x_{j+1} , the detector now selects the vector X_{j+1} associated with the smallest cost, $|N_{j+1}|^2$. The resultant m^g vectors $\{X_{j+1}\}$ are then stored together with the associated costs, $\{|N_{j+1}|^2\}$. One of these vectors is the true maximum-likelihood vector for the message with $(j+1)$ symbols. The process continues in this way. Ideally, the detector waits for the complete message to be received before selecting the final value of the true maximum-likelihood vector. For practical systems, where the message may consist of many thousand data symbols, the maximum likelihood vector is selected after some suitable delay. For the true maximum-likelihood vector X_j , the component x_{j-k+1} is taken as the detected value of s_{j-k+1} , giving a detection delay of $(k-1)$ sample intervals. If possible, $K \gg 3(g+1)$ ⁽³⁾. In determining s_{j-k+1} , the detector need not consider the values of x_{j-k} , x_{j-k-1} , ... and so instead of storing m^g j -component vectors $\{X_j\}$, it need only store the corresponding m^g k - component vectors $\{Q_j\}$ where;

$$Q_j = [x_{j-k+1}, x_{j-k+2}, \dots, x_j] \quad (6.4.6)$$

so that Q_j is formed from the last k components of the corresponding vector X_j .

Even in this reduced form, the Viterbi algorithm detector needs to store m^g k - component vectors together with the associated m^g cost terms. Also, for the detection of every data symbol, m^{g+1} squaring operations will have to be performed along with the selection of the

minimum cost functions. Clearly, when $g \gg 1$, both the computational effort (within a single symbol interval) and the amount of storage required becomes excessive. For practical implementation, it follows that the detector must operate with a smaller number of stored vectors.

Several methods have been widely studied in the literature which aim to achieve maximum-likelihood detection but with a reduced number of operations per iterative cycle and with lower storage requirements. In general, these methods give inferior performances when compared with the pure Viterbi-algorithm detector^(3-5, 8-17). For example, one such method uses a linear equaliser before the detector. This equaliser is adjusted to produce the "desired" sampled impulse response of the channel plus equaliser; that is, an impulse response which has a small number of components. The Viterbi algorithm may now be implemented very easily but since the equaliser may well equalise some of the amplitude distortion introduced by the channel, noise enhancement, together with correlation of the noise samples can result with the associated reduction in tolerance to noise^(3,5,16). Other methods also involve some pre-processing of the received signal before detection. The detection scheme developed in this work is an extended version of the method successfully used by Clark and Fairfield for the adaptive detection of data at 9600 bit/s when transmitted over telephone circuits⁽¹⁶⁾.

6.5 The Selected Near-Maximum-Likelihood Detector

Before embarking on a full description of the selected detection process, it is important to note from Sections 6.3 and 6.4 that optimum performance of non-linear equalisers and maximum-likelihood detectors

can only be achieved in the presence of statistically independent noise samples. However, from the discussion in Section 6.2, the noise samples at the output of the receiver filter in practical systems will be correlated to some degree. Ideally, the best tolerance to additive white Gaussian noise is achieved when a "whitened matched filter" is connected immediately ahead of a pure Viterbi algorithm detector. The whitened matched filter is the combination of a linear filter that is matched to the channel (and therefore to the individual received signal elements under the present assumptions), a sampler that samples the signal at the output of the filter at the symbol rate and a linear noise-whitening network implemented as a feedforward transversal filter, ^(3,9,11,13,16). Furthermore, when the data signal is transmitted over the channel close to the Nyquist rate, the whitened matched filter reduces to a low-pass filter with a flat amplitude response and linear phase characteristic over the passband with a cut-off frequency at half the rate of symbol transmission. Following this filter is a sampler and a linear transversal filter which removes all zeroes of the z -transform of the channel and filter that lie outside the unit circle in the z -plane and replaces them with the complex conjugates of their reciprocals, all remaining zeroes being left unchanged ^(3-5,16,17). This transversal filter therefore performs a pure phase transformation on the received signal without, however, changing any amplitude distortion introduced by the channel. From Section 6.2, this filter is identical to the linear feedforward transversal filter that forms the first part of a conventional decision-feedback non-linear equaliser which is adjusted to maximise the signal-to-noise ratio at its output. This filter forces the sampled impulse response of the channel plus filter combination to

be minimum phase and will therefore have both a rapid rise time and reduced duration. Since, after allowing for any constant level change across the filter, the statistics of the noise at the output of the transversal filter are identical to that at its input; a Viterbi algorithm detector operating at the output of the filter has the same tolerance to Gaussian noise as an appropriate Viterbi algorithm detector operating at its input. However, as the length of the sampled impulse response will be shorter at the output of the filter, the detector can be implemented with a lower number of computations and a lower storage requirement. Unfortunately, as the transversal filter does not equalise any amplitude distortion, it in general will not reduce the length of the sampled impulse response sufficiently to permit the practical use of a pure Viterbi algorithm detector. It follows that a practically realisable detector, both in terms of cost and technology, must make use of a detector that achieves near-maximum likelihood detection but with very much less storage and far fewer computations per received data symbol, for a given length of channel sampled impulse response.

The detection process developed here uses an adaptive linear transversal filter, the pre-filter, to remove the phase distortion introduced by the channel and adjusts the sampled impulse response of the channel plus filter to be minimum phase. An arrangement of decision directed cancellation of intersymbol interference is used to remove some of the intersymbol interference components present on the signal at the output of the linear pre-filter and so effectively adjusts the length of the sampled impulse response as seen by the detector to a known value. A modified Viterbi-algorithm detector operates on the received samples at the output of the signal

canceller, with a delay in detection of the data symbols equal to the length of the (equalised) sampled impulse response. The complexity of the detector is therefore proportional to the amount of intersymbol interference cancelled by the non-linear equaliser. It should be clear that if the detection delay is reduced to zero, the system degenerates into the conventional optimum non-linear equaliser of Section 6.3.1. Alternatively, if the delay in detection is increased to cover the whole message length, the system will approximate closely to the pure Viterbi-algorithm detector.

In practice, the ideal receiver filter in the reduced whitened matched filter cannot, of course, be physically realised. Also, due to both channel restrictions and noise bandwidth considerations, the receiver filters presented in Chapter 5 will be close approximations to the best possible practical filters. Consequently, noise samples at the input to the detector will be correlated to some degree and so ideal maximum-likelihood detection cannot be achieved in practice, although the work of Kobayashi^(10,24) indicates that correlated noise does not cause significant loss in performance in the Viterbi algorithm detector.

Figure 6.5.1 shows the general arrangement of the detection scheme, from which the similarity between the non-linear equaliser method can be seen. The linear pre-filter performs the same function as the first part of the conventional non-linear equaliser but now, the feedback filter removes only part of the intersymbol interference present on the samples at the output of the linear pre-filter. The detection process itself is a near-maximum likelihood detector which holds in store a pre-determined number of vectors ($\leq m^g$), where the length of these vectors is also fixed.

At time $t=iT$, the output of the linear pre-filter is

$$r'_i = \sum_{h=0}^g s_{i-h} y_h + u_i \quad (6.5.1)$$

where,

$y = [y_0 \ y_1 \ y_2 \ \dots \ y_g]$ here is the sampled impulse response of the channel and pre-filter. The $\{y_i\}$ will in general be complex-valued, as will the $\{r'_i\}$, $\{s_i\}$ and $\{u_i\}$. It is assumed here that the pre-filter is perfectly adjusted such that the z-transform $Y(z)$ has all of its zeroes inside the unit circle in the z-plane. The $\{u_i\}$ are Gaussian random variables with zero-mean values, the samples being correlated to some degree. The delay in transmission, other than that caused by time-dispersion of the received signal is neglected here, so that $y_0 \neq 0$ and $y_i = 0$ for $g < i < 0$. Furthermore, in the adjustment of the pre-filter, y_0 has been arranged to equal $1+j0$. The actual method of adjustment of the pre-filter and the channel estimator required by the detector itself and the intersymbol interference canceller is deferred until Chapter 7. It should also be mentioned that the $\{r'_i\}$ in equation (6.5.1) have been phase-corrected to allow for any phase error in the demodulation process; again, this is considered in more detail in a later Chapter.

The intersymbol interference canceller removes from the $\{r'_i\}$ estimates of all components involving data symbols whose final values have already been determined by the detector. The input to the detector itself is therefore;

$$r_i = r'_i - \sum_{h=n+1}^g s'_{i-h} y'_h \quad (6.5.2)$$

where the $\{s'_i\}$ are the detected values of the $\{s_i\}$ and $Y' = [y'_0 \ y'_1 \ \dots \ y'_g]$ are the estimates of the sampled impulse response of the channel and pre-filter, Y ; n here is the delay in detection and so s'_i is determined after the receipt of r_{i+n} , where $n < g$. Assuming that $s'_i = s_i$ and $y'_i = y_i$, then;

$$r_i = \sum_{h=0}^n s_{i-h} y_h + u_i \quad (6.5.3)$$

so that the sampled impulse response of the channel, pre-filter and canceller has been reduced from $(g+1)$ to $(n+1)$ components, thus simplifying the detection process when $n < g$.

Let the following k -component row vectors be defined;

$$\begin{aligned} S_k &= [s_1 \ s_2 \ s_3 \ \dots \ s_k] \\ R_k &= [r_1 \ r_2 \ r_3 \ \dots \ r_k] \\ U_k &= [u_1 \ u_2 \ u_3 \ \dots \ u_k] \\ X_k &= [x_1 \ x_2 \ x_3 \ \dots \ x_k] \\ Z_k &= [z_1 \ z_2 \ z_3 \ \dots \ z_k] \\ W_k &= [w_1 \ w_2 \ w_3 \ \dots \ w_k] \end{aligned} \quad (6.5.4)$$

x_i has one of the 64 or 256 possible values of s_i for systems a) and b) or system c), respectively, and

$$z_i = \sum_{h=0}^n x_{i-h} y_h \quad (6.5.5)$$

Also, w_i is the possible value of u_i satisfying;

$$r_i = z_i + w_i$$

In the k -dimensional complex vector space containing the vectors R_k , Z_k and W_k , the square of the unitary distance between the vectors R_k and Z_k is $(4, 14)$,

$$|w_k|^2 = |w_1|^2 + |w_2|^2 + \dots + |w_k|^2 \quad (6.5.6)$$

When the real and imaginary parts of the $\{w_i\}$ are statistically independent Gaussian random variables with zero means and fixed variance, the maximum-likelihood vector X_k corresponding to the received signal vector R_k is the vector X_k for which $|w_k|^2$, called the cost function of that vector, is minimum. Clearly, from the preceding work this is not the case here since the $\{u_i\}$ are correlated. However, if the correlation is not too high, it can be assumed for practical purposes that the smaller the value of $|w_k|^2$ associated with a vector X_k , the better will be the estimate, X_k , of S_k . From equation (6.2.22) it should be clear that widely separated samples, $\{u_i\}$, are essentially uncorrelated.

Just prior to the receipt of the sample r_k , the detector holds in store α different n -component row vectors, $\{Q_{k-1}\}$, where;

$$Q_{k-1} = [x_{k-n} \quad x_{k-n+1} \quad \dots \quad x_{k-1}] \quad (6.5.7)$$

and where, as before, x_i can take any of the 64 or 256 possible values of s_i . Each of the α vectors Q_{k-1} is formed by taking the last n components of the corresponding vector X_{k-1} . The cost term associated with each vector Q_{k-1} is the cost of the corresponding vector X_{k-1} from which Q_{k-1} was formed. On the receipt of sample r_k , the detector operates on the stored vectors $\{Q_{k-1}\}$ to give the detected data symbol s'_{k-n} . Finally, the detector decides which n -component vectors, $\{Q_k\}$, and their associated cost terms $\{|w_k|^2\}$, will be stored in place of the vectors $\{Q_{k-1}\}$ and costs $\{|w_{k-1}|^2\}$. Many detectors operating with the basic structure described above have been reported in the literature, the detectors being variants of Detection Scheme 1 in reference 8^(4,6-8,14-17). The scheme described below is a modification of the detection systems presented in reference 16.

On the receipt of sample r_k , each of the α stored vectors $\{Q_{k-1}\}$ are expanded into two groups of 8 or 16 $(n+1)$ -component vectors for systems a) and b) and system c), respectively. One group of vectors is designated $\{P_{k, \text{re}}\}$, the other $\{P_{k, \text{im}}\}$ where any one vector $P_{k, \text{re}}$ and $P_{k, \text{im}}$ is given by;

$$P_{k, \text{re}} = [x_{k-n} \quad x_{k-n+1} \quad \dots \quad \text{Re}(x_k)] \quad (6.5.8)$$

$$P_{k, \text{im}} = [x_{k-n} \quad x_{k-n+1} \quad \dots \quad j\text{Im}(x_k)] \quad (6.5.9)$$

where $\text{Re}(x_k)$ is the real part of x_k and can therefore be one of $\pm 1, \pm 3, \pm 5, \pm 7$ for systems a) and b), or $\pm 1, \pm 3, \pm 5, \pm 7, \pm 9, \pm 11, \pm 13, \pm 15$ for system c).

Similarly, $\text{Im}(x_k)$ is the imaginary part of x_k and can take one of the 8 values $(2l-1-\sqrt{m})$, $l=1, 2, \dots, \sqrt{m}$, with $m=64$ for systems a) and b) and the 16 values, $(2l-1-\sqrt{m})$, $l=1, 2, \dots, \sqrt{m}$, with $m=256$, for system c).

The cost associated with each of the vectors $P_{k, \text{re}}$ and $P_{k, \text{im}}$ is calculated from;

$$c_k = |w_{k-1}|^2 + \left| r_k - \sum_{h=0}^n x_{k-h} y_h \right|^2 \quad (6.5.10)$$

Now, for each of the 8 or 16 vectors $P_{k, \text{re}}$ and each of the 8 or 16 vectors $P_{k, \text{im}}$ derived from the root vector Q_{k-1} , the term $|w_{k-1}|^2$ is common and so the term;

$$|d_k|^2 = \left| r_k - \sum_{h=0}^n x_{k-h} y_h \right|^2 \quad (6.5.11)$$

can be seen to be the contribution to the total cost due to the extra component, x_k . The term $|d_k|^2$ is called the incremental contribution to c_k by x_k . From equation (6.5.11),

$$d_k = \mu_k - x_k \quad (6.5.12)$$

where

$$\mu_k = r_k - \sum_{h=1}^n x_{k-h} y_h \quad (6.5.13)$$

and where by design, $y_0 = 1+j0$.

The detector first selects the γ vectors from the vectors $\{P_{k, \text{re}}\}$, for which $\text{Re}(|d_k|^2)$ have the smallest γ values. Clearly, from equation (6.5.12), this may be achieved very simply by determining the γ values of $\text{Re}(x_k)$ which are closest to $\text{Re}(\mu_k)$ and so involves only a comparison procedure rather than the actual evaluation of any cost terms. The detector then operates in an identical fashion on the vectors $\{P_{k, \text{im}}\}$ but determines here the γ vectors for which $\text{Im}(x_k)$ is closest to $\text{Im}(\mu_k)$, for the possible values of $\text{Im}(x_k)$. This process is repeated for the α stored vectors $\{Q_{k-1}\}$ after which the detector will have a knowledge of $2\gamma\alpha$ low-cost vectors. From each of the α sets of 2γ vectors, the detector now forms the γ^2 $(n+1)$ -component vectors $\{P_k\}$. The first n components of the vectors $\{P_k\}$ are the n components of the corresponding α vectors $\{Q_{k-1}\}$ whilst the last components, x_k , are the γ^2 possible complex combinations of the γ values of $\text{Re}(x_k)$ and the γ values of $j\text{Im}(x_k)$ selected above. For each of the γ^2 vectors, $\{P_k\}$, the detector now evaluates the cost term;

$$|w_k|^2 = |w_{k-1}|^2 + |d_k|^2 \quad (6.5.14)$$

from the real and imaginary values of the $\{d_k\}$ already evaluated.

The detector now has a knowledge of the γ^2 vectors $\{P_k\}$ and their associated costs for each of the α stored vectors, $\{Q_{k-1}\}$, giving a total of $\alpha\gamma^2$ vectors and cost terms. The vector P_k with the overall smallest cost is selected from the set of vectors $\{P_k\}$ as the vector most likely to contain s_{k-n} as its first component and so s'_{k-n} is taken as x_{k-n} in the lowest cost vector. All vectors $\{P_k\}$ for which $x_{k-n} \neq s'_{k-n}$ are discarded and the first component of the remaining vectors is omitted, including that from the lowest cost vector, to give a set of n -component vectors, $\{Q_k\}$. The detector then selects α vectors from this set as follows. The first selected vector is the maximum-likelihood

vector itself; the second vector selected is, if consistent with the above, the lowest cost vector originating from the previous maximum-likelihood vector; the remaining $(\alpha-2)$ vectors are selected in order of smallest cost. These α n -component vectors are stored along with their cost terms for use after the receipt of sample r_{k+1} .

In practice, it is not, in fact, necessary for the detector to expand each of the α vectors $\{Q_{k-1}\}$ into the vectors $\{P_{k, \text{re}}\}$ and $\{P_{k, \text{im}}\}$ since the selection of the γ vectors from the $\{P_{k, \text{re}}\}$ and $\{P_{k, \text{im}}\}$ is based only on the incremental cost terms, $|d_k|^2$. The detector need only calculate the term μ_k once for each of the root vectors $\{Q_{k-1}\}$, and then determine the γ possible values of $\text{Re}(x_k)$ and the γ possible values of $\text{Im}(x_k)$ for which $|\mu_k - x_k|$ is closest to zero; the expanded vectors can then be reconstructed from a knowledge of the particular base vector and the particular value of x_k giving the minimum value of $|d_k|^2$. Indeed, by extending the above argument, the detector only has to form the vectors $\{Q_k\}$ after the detection and selection cycle is complete. Clearly, the processor implementing the detection algorithm in the above manner would require a very organised memory management scheme for the determination of the new stored vectors $\{Q_k\}$ and associated cost terms from the $\{Q_{k-1}\}$, the associated cost terms $|w_{k-1}|^2$ and the values of x_k resulting in the γ^2 minimum values of $|d_k|^2$.

Assuming that each of the α vectors $\{Q_{k-1}\}$ are not actually expanded into the 8 or 16 vectors $\{P_{k, \text{re}}\}$ and the 8 or 16 vectors $\{P_{k, \text{im}}\}$, the detector may determine the γ values of the $\text{Re}(x_k)$ and the γ values of $\text{Im}(x_k)$ as follows, where for illustration, it is assumed that $\gamma=2$ and the system uses a 64-point structure; Figure 6.5.2 a) shows the expansion scheme diagrammatically. For each stored vector Q_{k-1} , the detector

calculates $\text{Re}(u_k)$ and $\text{Im}(u_k)$ from equation (6.5.13). If $-\text{Re}(u_k) < 0$, then $\text{Re}(x_k) = 3$ is added to $-\text{Re}(u_k)$; if $-\text{Re}(u_k) > 0$, then $\text{Re}(x_k) = -3$ is added to $-\text{Re}(u_k)$. (Clearly, if $-\text{Re}(u_k) = 0$, then the two values of $\text{Re}(x_k)$ resulting in minimum $|d_k|^2$ will be $\text{Re}(x_k) = \pm 1$). From equation (6.5.12), $-\text{Re}(u_k) + \text{Re}(x_k) = -\text{Re}(d_k)$. The detector now determines the sign of $-\text{Re}(d_k)$; if $-\text{Re}(d_k) < 0$, the value of $\text{Re}(x_k)$ used originally was too small and so $+1$ is added to $-\text{Re}(d_k)$. Conversely, if $-\text{Re}(d_k) > 0$, the original value of $\text{Re}(x_k)$ was too large and so -1 is added to $-\text{Re}(d_k)$. At this point, the detector will have determined one of the two values of $\text{Re}(x_k)$ giving one of the two minimum values of $\text{Re}(d_k)$. After the addition of ± 1 as appropriate, the sign of $-\text{Re}(d_k)$ is again investigated; if $-\text{Re}(d_k) < 0$, $+1$ is added, if $-\text{Re}(d_k) > 0$, -1 is added. At this point, the detector will have determined the second value of $\text{Re}(x_k)$ for the second minimum value of $|\text{Re}(d_k)|$. The detector then stores the two values of $\text{Re}(x_k)$ and the two values of $-\text{Re}(d_k)$, these values having been determined after only 3 operations rather than 15 operations had a simple two-pass threshold comparison method been used. The detector then operates on $-\text{Im}(u_k)$ in a similar manner to determine the two values of $\text{Im}(x_k)$ and the associated two minimum values of $\text{Im}(d_k)$. The $\gamma^2 = 4$ cost terms associated with the $\gamma^2 = 4$ combinations of the $\text{Re}(x_k)$ and $\text{Im}(x_k)$ for each root vector Q_{k-1} are then calculated from;

$$|w_k|^2 = |w_{k-1}|^2 + (-\text{Re}(d_{k,1,2}))^2 + (-\text{Im}(d_{k,1,2}))^2 \quad (6.5.15)$$

where $-\text{Re}(d_{k,1,2})$ and $-\text{Im}(d_{k,1,2})$ are the two values of $-\text{Re}(d_k)$ and the two values of $-\text{Im}(d_k)$ closest to zero.

Figure 6.5.2 b) shows the corresponding expansion scheme for a 256-point signal structure, from which it can be seen that 8 operations

are required to determine the two values of $\text{Re}(x_k)$ and the two values of $\text{Im}(x_k)$ giving the minimum values of $\text{Re}(d_k)$ and $\text{Im}(d_k)$, respectively. Clearly, other values of γ may be accommodated by making suitable modifications to the lattice structures of Figures 6.5.2.

6.6 Computer Simulation Results and System Comparisons

In order to determine the relative performances of systems a), b) and c) to additive Gaussian noise when operating over telephone circuits 1-3 with filter sets 1 and 2, the operation of the data transmission system given in Figure 6.2.1 has been simulated on a digital computer. The general computer program used in these simulations is given as Program 5 in Appendix 10, which is a FORTRAN program written for the University of Manchester Region Computer Centre CDC7600 machine; access to the CDC7600 was either via an ICL 7020 system, or via the MIDNET link^(18,19,20). Program 5 in Appendix 10 actually shows an example program used in the simulation of 64-point QAM signal sets using coding scheme 3 presented in Chapter 4. However, the main program code is quite general and is easily modified to simulate 256-point structures by making appropriate changes to input parameters and including the extended constellation, new coding/decoding tables and increasing the number of bits per symbol in the error analysis section.

In all cases, the linear pre-filter prior to the detector itself is assumed to be perfectly adjusted and so the program operates with the sampled impulse responses given in Tables 5.6.1 - 5.6.3. Furthermore, the estimate of the sampled impulse response of the linear baseband channel, as required by the intersymbol interference cancellation process and the detector itself, is also assumed to be perfect and time invariant.

Again, prior to the actual simulation of transmission, an initialisation procedure is used to ensure that the various vectors and cost terms used by the detector are correctly set. This can be achieved simply by setting one of the stored vectors, X_j , in the detector to the correct data sequence and setting its associated cost function, $|w_j|^2$, to zero. The remaining vectors can be filled with any arbitrary sequences with their cost functions set to some large value. During the actual simulation, the cost functions given in equation (6.5.15) are prevented from increasing at each symbol detection by subtracting the value of the smallest cost function, (i.e. that associated with the maximum-likelihood vector) from all other cost functions stored by the detector.

Figures 6.6.1 - 6.6.4 show the performance of the detection scheme described earlier when operating on data transmitted over telephone circuits 1-3 with filter sets 1 and 2. As before, systems a) and b) refer to 64-point QAM signal structures with a signal element rate of 3200 bauds including filter set 1 and 2 respectively, whilst system c) is the 256-point QAM system operating at 2400 bauds with filter set 1. Systems a) and b) both use coding scheme 3 (Figure 4.4.20) whilst system c) uses coding scheme 4, (Figure 4.4.23). Associated with the performance curves in Figures 6.6.1 - 6.6.4 are the identifiers α , n and γ which are the number of stored vectors, the delay in detection and the number of real and imaginary expansions used by the detector, respectively; the curves show variation of bit-error rate with signal-to-noise ratio for the given values of α , n and γ , with 95% confidence limits of around 0.6 dB for all cases. As before, Appendix 3 defines signal-to-noise ratio as used throughout this work.

For purposes of comparison, Figures 6.6.1 - 6.6.4 also show the performance of conventional non-linear equalisers when operating over the given baseband channel, where "non-linear equaliser a, b or c" refers to systems a), b) or c) respectively, operating with the optimum non-linear equaliser described in Section 6.3.2. The results presented for the near-maximum likelihood detector cover a range of α , n and γ values in an attempt to illustrate the detectors performance over channels introducing various levels of distortion. When comparing the relative performances of these systems, it should be borne in mind that the complexity of the detector increases very rapidly with increasing values of the three parameters α , n and γ , which is studied in more detail in the following section. Hence, whilst enhanced performance is to be expected for larger values of these parameters, the number of operations required per symbol period and the amount of storage required may well make the detector unrealisable for both economic and technological reasons.

Figures 6.6.1 - 6.6.3 show the performances of systems a) and c) when operating over telephone circuits 1-3. Circuits 1 and 2 represent fairly mild telephone channels although circuit 2 introduces slightly more attenuation distortion compared with circuit 1; the effect of the extra distortion can be seen in Figures 6.6.1 and 6.6.2 where in general, the performance over circuit 1 is slightly superior to that obtained over circuit 2 for similar detection schemes. However, as illustrated by the figures, the near-maximum-likelihood detectors (NMLH) achieve more of an advantage over circuit 2 when compared with the non-linear equaliser (NLE). For example, system a) with $\alpha=8$, $n=20$, $\gamma=4$ achieves an advantage of approximately 5dB over NLE a) for circuit 1 whilst with $\alpha=8$, $n=16$, $\gamma=4$,

system a), over circuit 2, gives an improvement of about 5.2 dB when compared at a bit error rate of 1 in 10^4 . Since system c) operates at a lower signal element rate than system a), (or system b)), the amount of distortion suffered by the received signal will be less. However, when operating at the same average energy per bit, system c) will necessarily give a poorer performance than system a) when operating over a perfect channel. Consequently, when operating over more realistic telephone circuits, system c) may well gain an advantage over system a) provided the advantage due to the lower distortion offsets the inherent disadvantage due to the reduced distance between symbols in the larger signal constellation. The NLE a) and NLE c) curves in Figures 6.6.1 - 6.6.3 clearly show that system c) does in fact gain an overall advantage over system a) when both systems employ non-linear equalisers, but the relative advantages of using NMLH detectors with system c) are lower than the corresponding advantages with system a). The figures also show that an improved performance can be achieved with system a) or c) at larger values of α , n and γ , which is to be expected since detection processes with these larger values approximate more closely to the full maximum-likelihood detector.

Figure 6.6.5 shows the performance curves for the 64-point and 256-point signal structures when operating over an ideal channel; these results were obtained using the same simulation programs as used for the production of Figures 6.6.1 - 6.6.4, but with the sampled impulse response of the channel having only one non-zero component (the first component), which was set to $1+j0$. Comparison between Figures 6.6.5 and 6.6.1 shows that the better detectors with system a) lose about 3.5-4.5 dB in tolerance to additive Gaussian noise over telephone circuit 1 relative to the performance

over an ideal channel, this loss being attributed to the bandlimiting introduced by filter set 1. It follows that if filter set 1 is replaced by the wider bandwidth filters constituting filter set 2, system b), when operating over telephone circuit 1, will lose less than 3.5 - 4.5dB relative to the ideal channel. Since, from Figure-6.6.5, system c) must be inferior to a 64-point system (operating over an ideal channel) by about 5dB, it follows that system b) must have a better performance over telephone circuit 1 than system c) when working over the ideal channel, the improvement being of the order 1-2 dB.

As can be seen from Figures 6.6.1 - 6.6.4, the performance of NLE a) is inferior to all other detection processes over telephone circuits 1-3. Even over the better circuit (circuit 1), NLE a) loses around 9.6 dB in tolerance to noise relative to operation over the ideal channel; over the worst channel considered here (telephone circuit 3 with filter set 1), NLE a) is inferior to the ideal by about 17.6 dB. The non-linear equaliser with system c) achieves a better performance than NLE a) over telephone circuits 1-3 but is inferior to all other detection processes considered except for system a) with $\alpha=8$, $n=8$, $\gamma=2$, where it gains approximately 1dB at an error rate of 10^{-3} . However, as argued above, system b) will achieve approximately 1-2 dB improvement over system c) when working over telephone circuit 1. Consequently, the performance of NLE c) will be inferior to that of system b) when using NMLH detection schemes with values of α , n and γ greater than, or equal to 8, 8 and 2 respectively. A similar argument applies to Figure 6.6.2 which shows simulation results for systems a) and c) when operating over telephone circuit 2.

As previously mentioned, telephone circuit 3 represents a typical worst-case circuit likely to be considered as suitable for the transmission of data at 9600 bit/s over the switched telephone network^(16,17). As the characteristics of Schedule D private circuits are guaranteed to be superior to those of telephone circuit 3⁽²¹⁾, successful transmission of data at 19,200 bit/s over the latter circuit should also prove that successful operation may be achieved over private circuits conforming to lower standards than Schedule D circuits. For this reason, the following Chapters concerned with the adaptive adjustment of the modem receiver and receiver synchronisation assume operation over the worst-case characteristics of telephone circuit 3.

Figures 6.6.3 and 6.6.4 show the results of computer simulations of systems a), b) and c) when operating over telephone circuit 3. Predictably, the absolute performances of the various detection schemes used over this circuit are all inferior to similar schemes operating over the less severe telephone circuits. However, the relative advantages of using NMLH detectors over the conventional non-linear equalisers are very encouraging; for example, system a) with $\alpha=8$, $n=8$, $\gamma=2$ achieves an advantage of approximately 5dB relative to NLE a), this giving a performance which is only 1-2 dB inferior to NLE a) when operating over telephone circuit 2. The most important results of the simulation studies performed here are those given in Figures 6.6.3 and 6.6.4 which show that system b) achieves an advantage of around 1.6 dB in tolerance to Gaussian noise relative to system a) with similar values for α , n and γ , when operating over telephone circuit 3; this advantage giving system b) an extra tolerance of some 7dB compared with NLE a). Based upon the above

discussion, it appears that a near-maximum-likelihood detector will give an overall increase in tolerance to additive Gaussian noise when the data is transmitted over telephone circuits such as circuits 1, 2 and 3. Moreover, from observations made during the computer simulations, the near-maximum-likelihood detectors recover from a burst of errors much more rapidly than the non-linear equalisers tested; indeed, under severe conditions, the non-linear equalisers had not recovered from an error burst at the end of a program iteration which comprised some 50,000 detections. It has also been shown that this advantage over the non-linear equalisers becomes larger when operating over channels which introduce larger amounts of distortion, such as telephone circuit 3. Coupled with the advantages in terms of tolerance to the timing phase at the receiver and the fact that the non-linear equaliser has no inherent advantage in its tolerance to phase jitter^(5,17), then provided the adaptive detector can be accurately adjusted when operating over circuit 3 and provided phase-jitter can be controlled at the receiver, a near-maximum likelihood detection scheme will provide a superior alternative to the conventional non-linear equaliser.

Finally, Figures 6.6.6 and 6.6.7 show the dependence of bit-error rate on the accuracy of the receiver's estimate of received signal level and carrier phase respectively, for systems b) and c) when operating over telephone circuit 3. Clearly, system b) is to be preferred to system c) since the former system can tolerate slightly larger parameter inaccuracies than the latter. However, for the correct operation of a detector working with system b), the receiver will have to acquire very accurate estimates for both received signal level and signal phase.

In conclusion, near-maximum-likelihood detection will give an improved system when compared with a conventional non-linear equaliser provided the signal level, carrier phase, sampled impulse response of the channel and adjustment of the pre-filter can all be accurately adjusted or estimated at the detector. Furthermore, the detection scheme selected, and hence the modem receiver, should not be so complex that it becomes technologically unfeasible nor economically unrealistic.

6.7 Effect of the Detection Parameters α , n and γ on Detector Complexity

From equation (6.5.7), each of the α n -component vectors stored in the receiver just prior to the receipt of the k^{th} received sample will be of the general form;

$$\mathbf{Q}_{k-1} = [x_{k-n} \quad x_{k-n+1} \quad \dots \quad x_{k-1}] \quad (6.7.1)$$

where the $\{x_i\}$ are complex quantities and can take any of the 64 or 256 possible values of s_i for systems a) and b) and system c) respectively. For a given detection delay, n , increasing the number of vectors from α to $\alpha + \Delta\alpha$ will increase the amount of memory required by $(2\Delta\alpha n B_1)$ bits where B_1 is the number of bits used to define each $\text{Re}(x_i)$ and $\text{Im}(x_i)$. Since each stored vector has an associated stored cost function $|w_{k-1}|^2$, which is, of course, a real valued quantity, a further $(\Delta\alpha B_2)$ bits of memory will be required to store the extra cost terms, where B_2 is the number of bits used to represent a single cost value.

For example, if α is increased from 8 to 16 stored vectors and n remains constant at say, 8 symbol periods, the amount of storage required at the detector will increase by 270 bytes where it is assumed that the $\text{Re}(x_i)$, $\text{Im}(x_i)$ and $|w_{k-1}|^2$ values are stored as 2-byte words. Clearly,

with relatively cheap semiconductor memory readily available, this increase in memory requirement will not add significant complexity (nor cost) to the modem receiver.

The major impact of increasing α , n or γ will be on the amount of computational effort required at the detector. As explained in Section 6.5, the selection of vectors during the first stage of the double-expansion scheme⁽²²⁾ is based on the values of the associated cost functions and so it is only necessary to determine (using Figure 6.5.2) the γ values of $\text{Re}(x_i)$ and the γ values of $\text{Im}(x_i)$ which result in the smallest values of d_k for every stored vector, Q_{k-1} . These 2γ values are then stored along with the associated values of $\text{Re}^2(d_k)$ and $\text{Im}^2(d_k)$; clearly, increasing the value of γ for given values of α and n will also increase the amount of storage required at the detector. On the receipt of sample r_k , the detector must first evaluate μ_k for each of the α stored vectors $\{Q_{k-1}\}$. From equation (6.5.13), this operation will involve $4n$ real multiplications and $4(n+1)$ real additions or subtractions for each of the α stored vectors. Depending upon the internal architecture of the particular processing device used, it may be beneficial in terms of data transfer operations, to calculate all of the μ_k values prior to the selection process; the alternative method would be to operate on each stored vector, Q_{k-1} , individually by first calculating the associated μ_k followed by the selection of the γ^2 values of x_k giving the γ^2 minimum values of $|d_k|^2$. Regardless of the method actually used, the initial selection processes will involve a minimum of 8 real additions or subtractions and 4 real squaring operations to determine the 4 minimum values of $|d_k|^2$ for each stored vector, where $\gamma=2$ and system a) or b)

is used. The maximum number of operations will occur when system c) is used with $\gamma=16$, when all of the 16 possible values of $\text{Re}(x_i)$ and $\text{Im}(x_i)$ will be used to determine the 256 combinations of the complex-valued x_i . In this case, 32 squaring operations will be required for each stored vector.

Following the determination of the γ^2 minimum values of $|d_k|^2$ for each of the α stored vectors, the detector then selects as the maximum-likelihood vector, the vector Q_{k-1} with the overall lowest cost function, c_k , as given in equation (6.5.10). This process only involves real additions or subtractions and some comparison technique; the actual method used being dependent on the processing device employed. The detector must then select and store the α vectors $\{Q_k\}$ and associated cost functions $\{|w_k|^2\}$ prior to the receipt of sample r_{k+1} ; again, this process will not involve any multiplications. In general, the majority of the computational effort at the detector will be concerned with the complex multiplications involved when calculating the α values of the $\{\mu_k\}$ and the squaring operations required to evaluate the $\{|d_k|^2\}$. Assuming that multiplications will be the most time consuming operation required of the detector's processor, an idea of system complexity when using different values of α , n and γ can be obtained by considering the number of multiplications. It should be clear from the above discussion that increasing α or n will rapidly increase the number of complex multiplications required in the evaluation of the $\{\mu_k\}$ whilst the number of real squaring operations will increase in proportion to 2γ . For example, if system a) or b) is used with $\alpha=8$, $n=8$, $\gamma=2$, the minimum number

of real multiplications required during every detection period of $1/3200$ sec. will be 260 which implies a single multiply time of less than $1.2\mu\text{s}$. If system c) is used with $\alpha=16$, $n=16$, $\gamma=16$, the number of real multiplications increases to 1056 which would require a multiplication cycle time of less than $0.4\mu\text{s}$. Due to the extra loading on the detector's processor when including the various additions, subtractions, comparisons and data transfer operations it must perform to complete a single detection cycle, it should be clear that the multiplier times given above must be reduced significantly to ensure successful operation. Moreover, as will be shown in Chapter 9, high-speed multiplier-accumulator devices such as the TRW 10010J⁽²³⁾ can only perform a single 16×16 bit multiply operation in about $0.2\mu\text{s}$ and so although the results of the computer simulations point towards the use of detectors with high values of α , n and γ , the practical constraints imposed by present-day technology may force the system designer to select detection systems with fairly low values of the 3 detection parameters. For the above mentioned technological reasons, it is strongly recommended that any system which is developed as a result of this work should initially use system b) with $\alpha=8$, $n=8$, $\gamma=2$ in order to assess the amount of processing and therefore the processing time taken in the physical implementation of the detector, whilst maintaining a good advantage in performance over that of a non-linear equaliser.

It should be clear from the discussion above that a major part of a detection cycle will be concerned with the calculation of the $\{\mu_k\}$. As can be seen from equation (6.5.13), the detector must perform n complex multiply-accumulate cycles in the evaluation of the term

$$\sum_{h=1}^n x_{k-h} y_h$$

for each stored vector.

Since the values of the $\{x_i\}$ are fixed and known at the detector, a considerable reduction in the number of operations required per detection cycle could be achieved if the sampled impulse response, Y , can be considered time invariant or only very slowly varying with time. Under these conditions, the detector may form and store the partial-products $x_{k-h} y_h$ which would then reduce the evaluation of the $|\mu_k|$ to simple accumulation and subtraction operations. Since the modem receiver will require the use of an adaptive channel estimator in order to adjust the tap-gains of the pre-filter, a rather simple method of implementing this operation-saving technique would be for the estimator to supply the detector's memory with the partial products during modem initialisation and to update the memory when the sampled impulse response of the channel changes by some pre-determined amount. It is shown in the following chapters that this may be achieved in practice by having a dedicated processor operating in parallel with the detector, which supplies the tap-gains of the pre-filter, the estimate of the overall sampled impulse response of the channel plus filter and the partial products as required by the detector.

6.8 References

1. Hawksford, M.J. and Rezaee, N., "Adaptive Mean-Square-Error Transversal Equaliser", IEE Proc. Pt. F., Vol.128, No.5, pp 296-304, Oct.1981
2. Clark, A.P. and Tint, U.S., "Linear and Non-Linear Transversal Equalizers for Baseband Channels", The Radio and Electronic Engineer, Vol.45, No.6, pp 271-283, June 1975
3. Harvey, J.D., "Synchronisation of Synchronous Modems", SERC Report No.GR/A/1200-7, December 1980
4. Clark, A.P., "Advanced Data Transmission Systems", Pentech Press Ltd., 1977
5. Najdi, H.J., "Digital Data Transmission Over Voice Channels", Ph.D.Thesis, Loughborough University of Technology, 1982
6. Clark, A.P. and Harvey, J.D., "Detection Processes for Distorted Binary Signals", The Radio and Electronic Engineer, Vol.46, No.11, pp533-542, Nov.1976
7. Clark, A.P. and Harvey, J.D., "Detection of Distorted Q.A.M. Signals", Electronic Circuits and Systems, Vol.1, No.3, pp103-109, April 1977
8. Clark, A.P., Harvey, J.D. and Driscoll, J.P., "Improved Detection Processes for Distorted Digital Signals", IERE Conf.Proc.37, pp125-130, 1977
9. Farney, G.D., "Maximum Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference", IEEE Trans. Inform. Theory, Vol.IT-18, pp363-378, May 1972
10. Magee, F.R. and Proakis, J.G., "Adaptive Maximum-Likelihood Sequence Estimation for Digital Signalling in the Presence of Intersymbol Interference", IEEE Trans. Inform. Theory, Vol.IT-19, pp120-124, Jan.1973

11. Farney, G.D., "The Viterbi Algorithm", Proc.IEEE, Vol.61, No.3, pp268-278, March 1973
12. Magee, F.R., "A Comparison of Compromise Viterbi Algorithm and Standard Equalisation Techniques over Band-Limited Channels", IEEE Trans. Commun., Vol.COM-23, pp361-367, March 1975
13. Falconer, D.D. and Magee, F.R., "Adaptive Channel Memory Truncation for Maximum-Likelihood Sequence Estimation", Bell System Tech. J., Vol.52, pp1541-1562, Nov.1973
14. Clark, A.P., Kwong, C.P. and Harvey, J.D., "Detection Processes for Severely Distorted Digital Signals", Electronic Circuits and Systems, Vol.3, No.1, pp27-37, Jan.1979
15. Clark, A.P., Harvey, J.D. and Driscoll, J.P., "Near-Maximum-Likelihood Detection Processes for Distorted Digital Signals", The Radio and Electronic Engineer, Vol.48, No.6, pp301-309, June 1978
16. Clark, A.P. and Fairfield, M.J., "Detection Processes for a 9600 Bit/s Modem", The Radio and Electronic Engineer, Vol.51, No.9, pp455-465, Sept.1981
17. Clark, A.P., Najdi, H.Y. and Fairfield, M.J., "Data Transmission at 19.2 kbit/s over Telephone Circuits", The Radio and Electronic Engineer, Vol.53, No.4, pp157-166, April 1983
18. Gerrard, G.P., "Using the ICL19045/CDC7600 Joint System at Manchester", Loughborough University Computer Centre Document No.AD/IN/317, May 1980
19. Bateman, S.C., "Sending a Job Via MIDNET", Internal Comm. 1982.
20. Bateman, S.C., "Using the Manchester Joint System Via the Loughborough Prime", Internal Communication, Nov.1982
21. "Handbook of Data Communications", Editor, G.B. Bleazard, National Computing Centre Ltd., 1982

22. Clark, A.P. and Fairfield, M.J., "U.K. Patent Application GB 2 087 198A", The Patent Office, Application No.8034682, May 1982
23. TRW LSI Products, "TRW Multiplier-Accumulator TDC1010J", Data Sheet, 1979
24. Kobayashi, H., "Application of Probabilistic Decoding to Digital Magnetic Recording Systems", IBM J.Res.Develop., Vol.14, pp64-74, Jan.1971

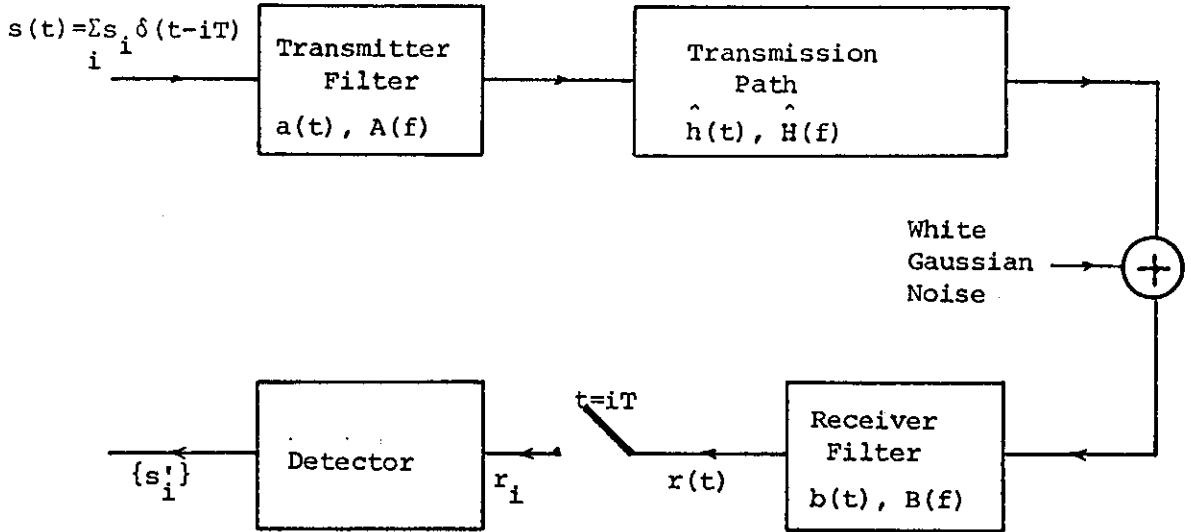


Figure 6.2.1: Basic system model

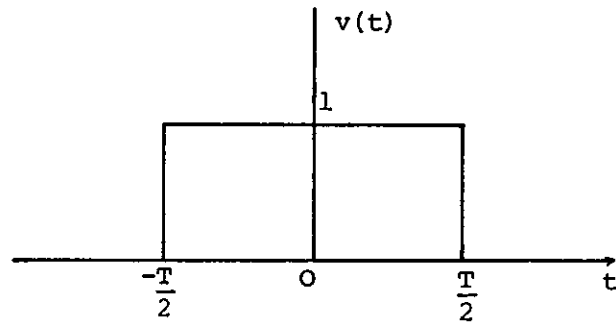
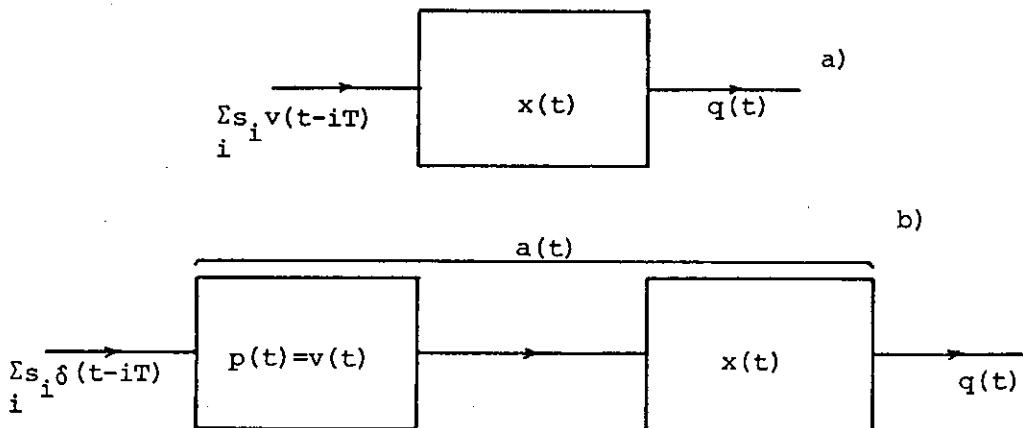
Figure 6.2.2: The function $v(t)$ 

Figure 6.2.3: Changes to transmitter filter

- a) Rectangular input waveform
- b) Changes to filter for same output but with impulsive input

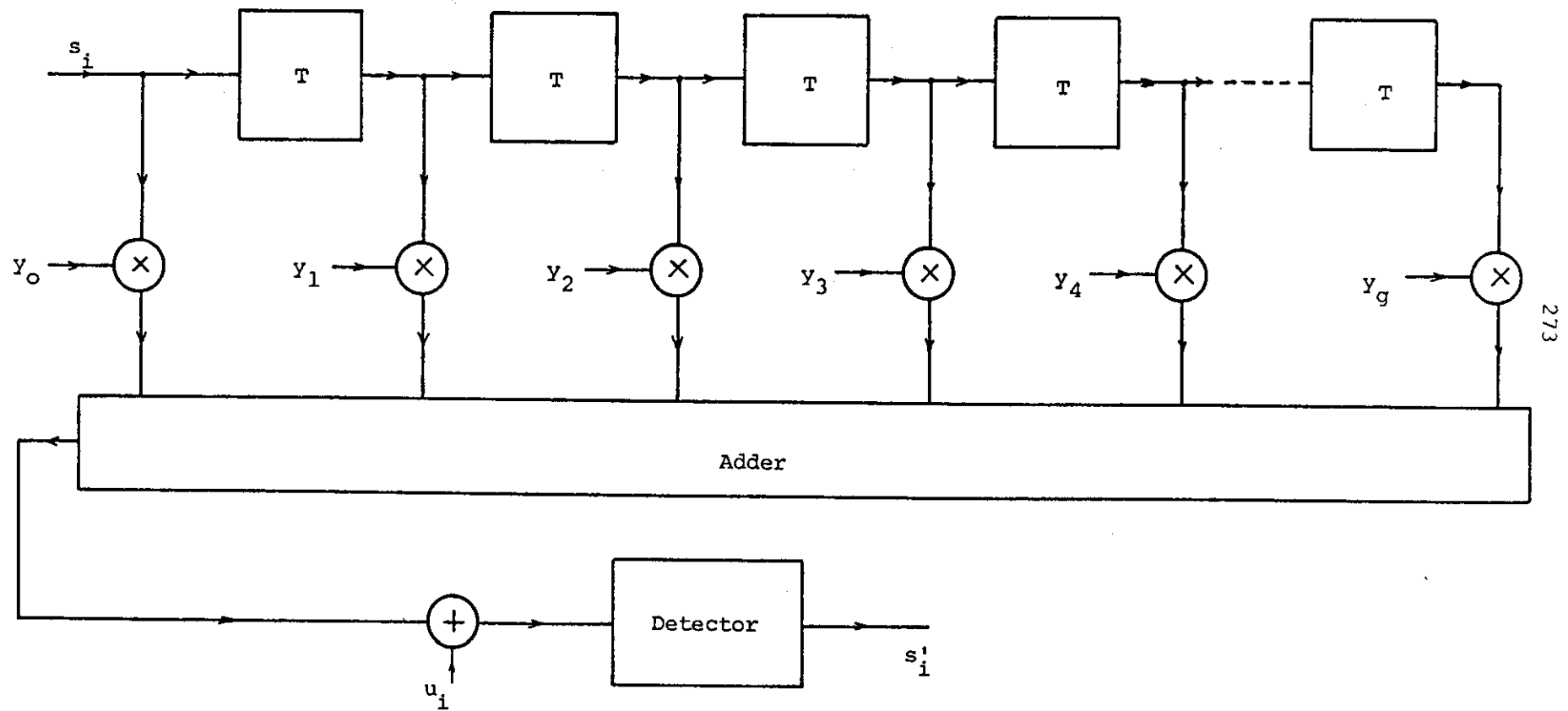


Figure 6.2.4: Discrete-time version of the baseband model

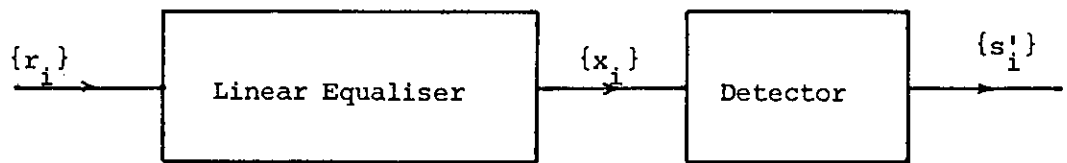


Figure 6.3.1: General arrangement of linear equalisation

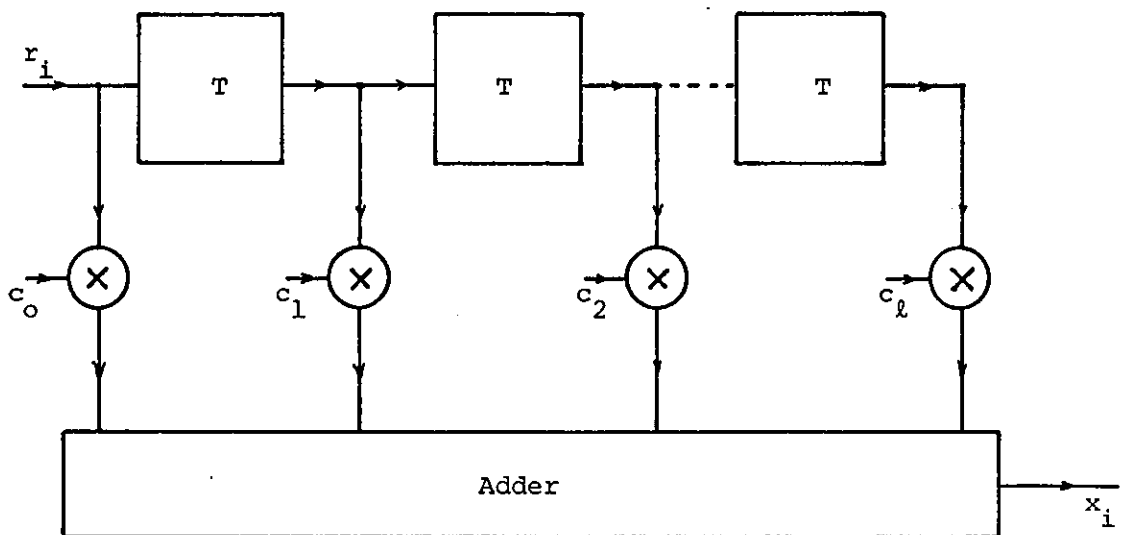


Figure 6.3.2: Linear feedforward transversal equaliser

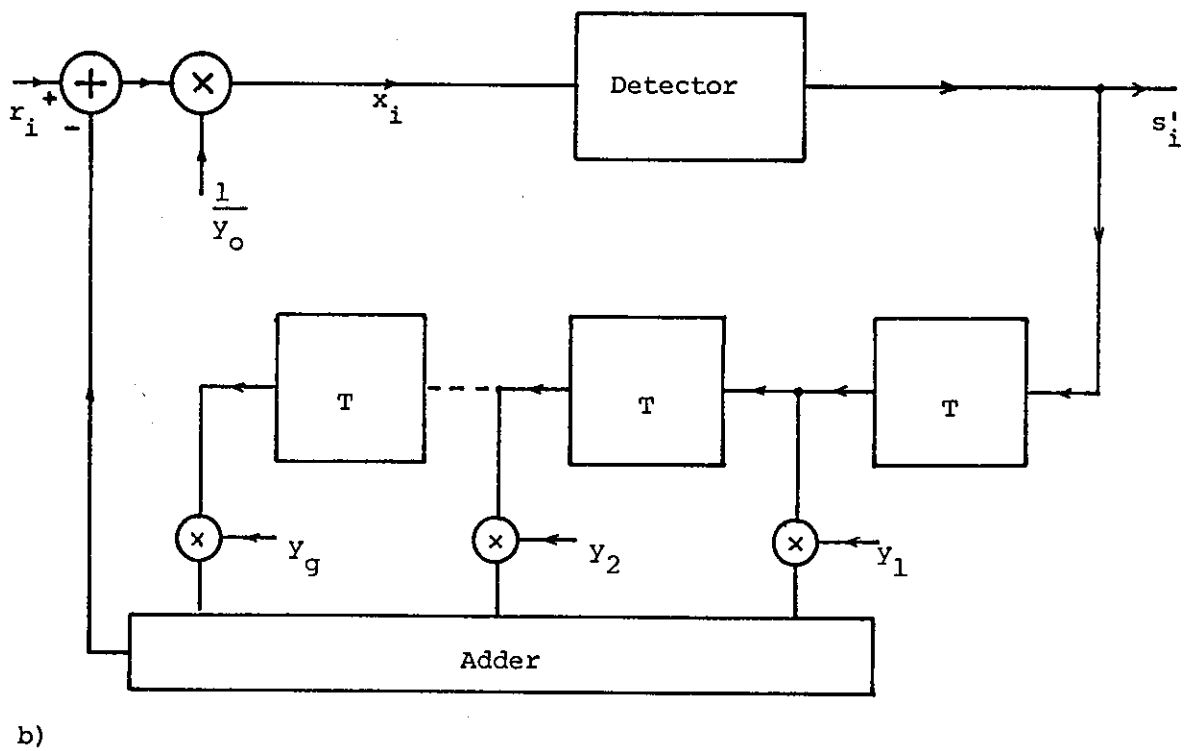
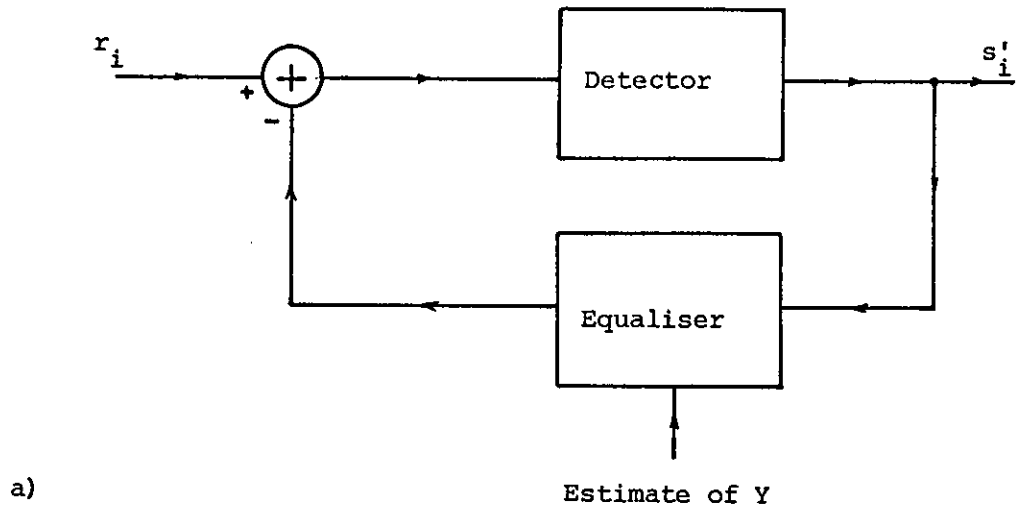


Figure 6.3.3: Non-linear equalisation

a) Basic arrangement

b) Decision-directed cancellation of intersymbol interference

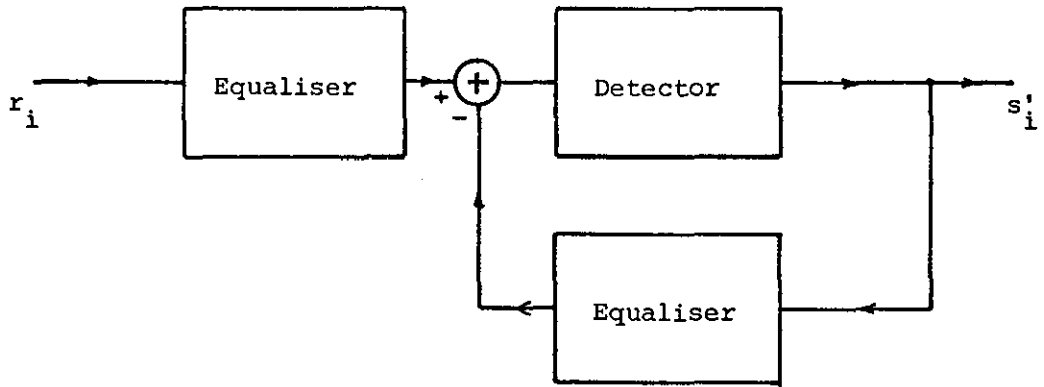


Figure 6.3.4: General arrangement of combined linear and non-linear equalisation

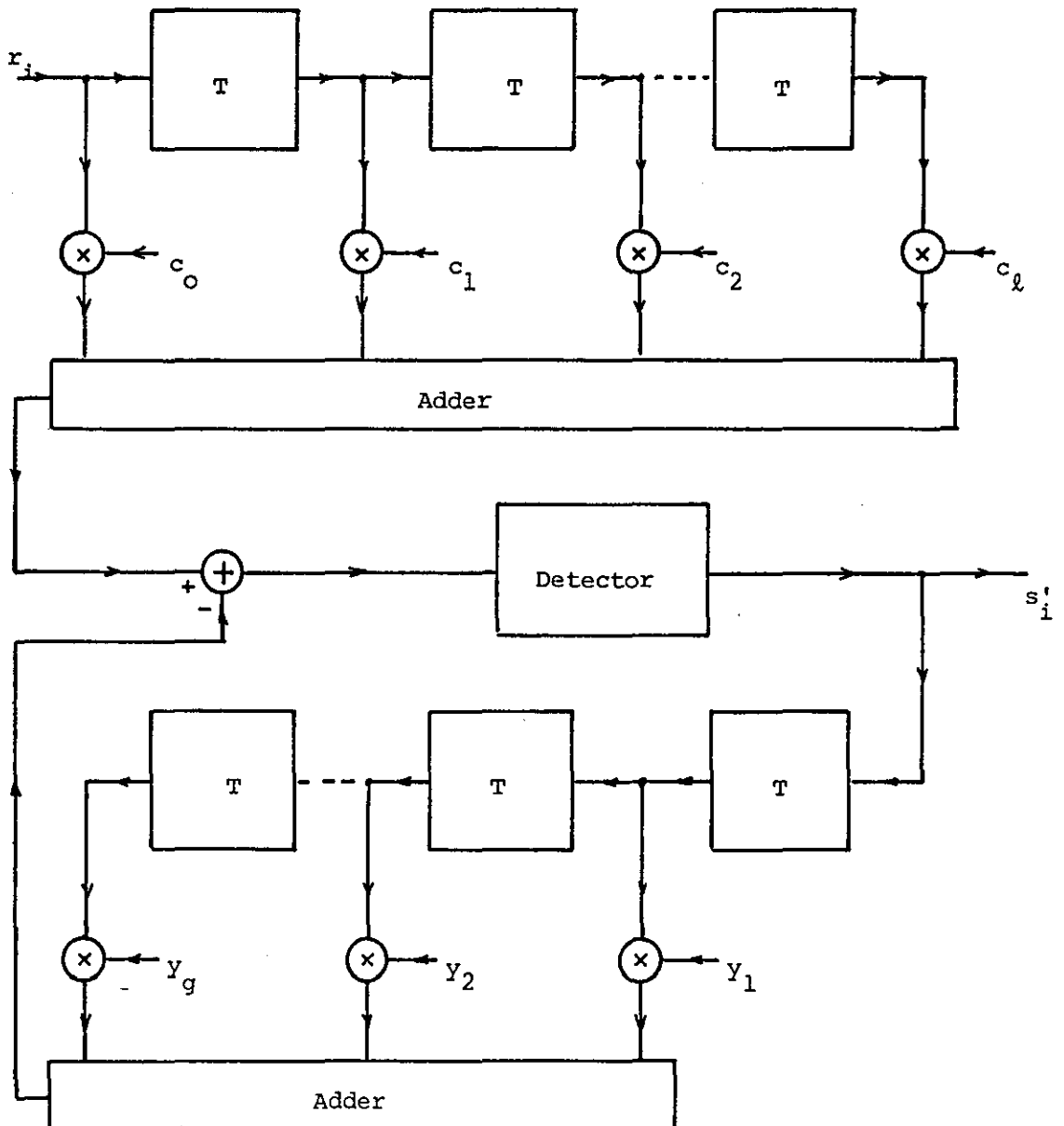


Figure 6.3.5: The combined non-linear equaliser

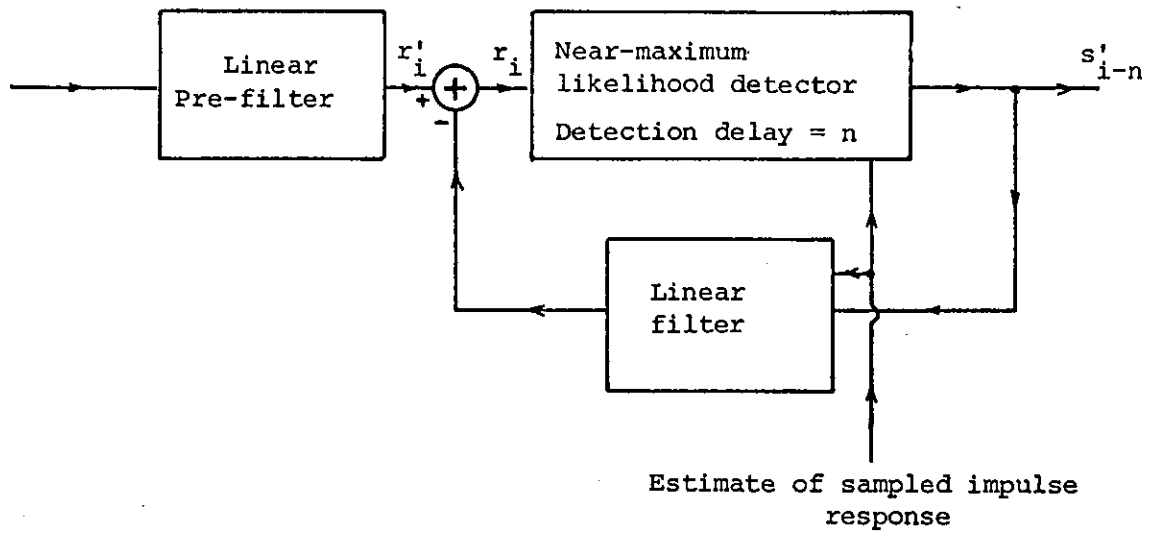


Figure 6.5.1: General arrangement of the near-maximum-likelihood detection scheme

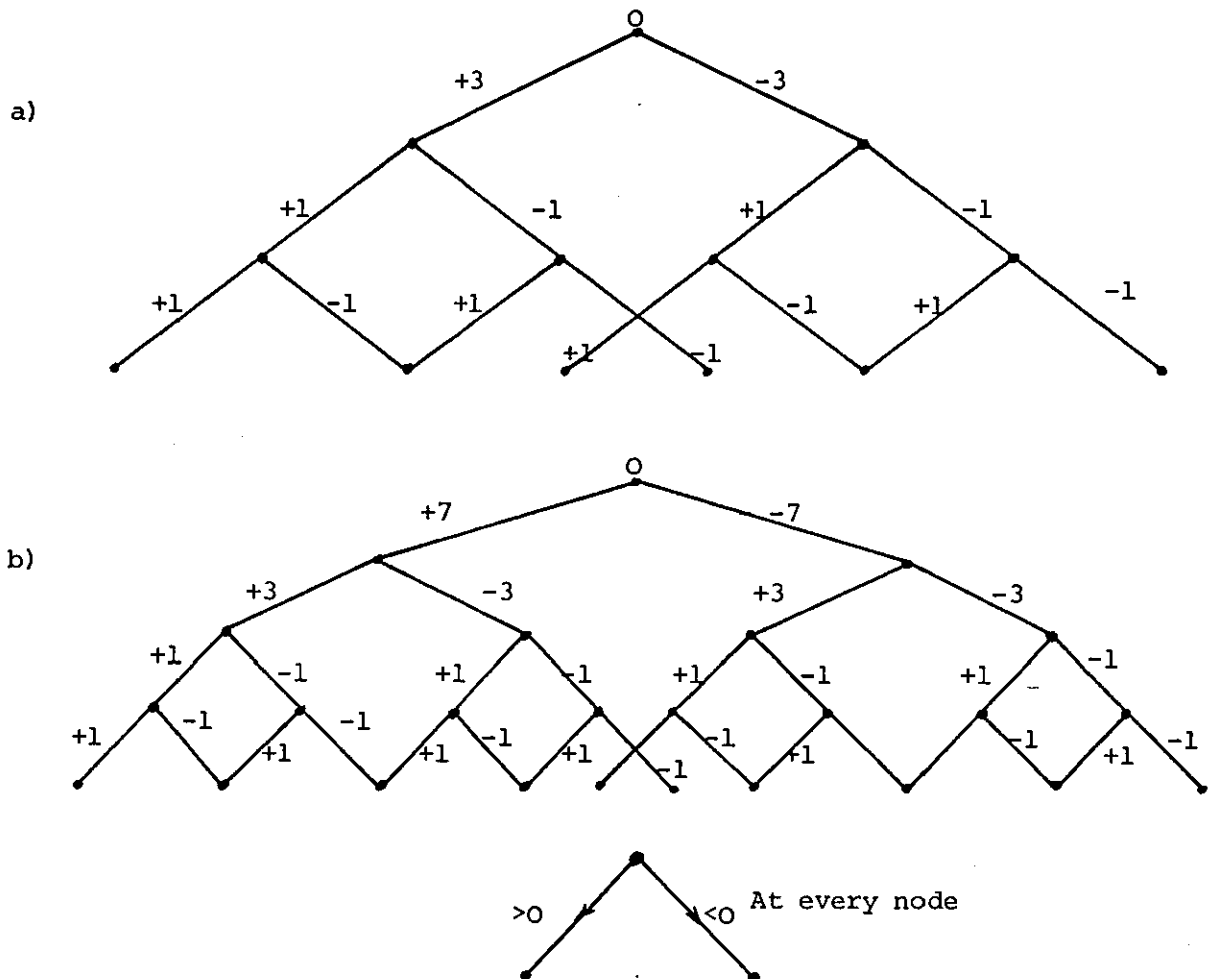


Figure 6.5.2: Expansion schemes for:
 a) 64-point structure
 b) 256-point structure

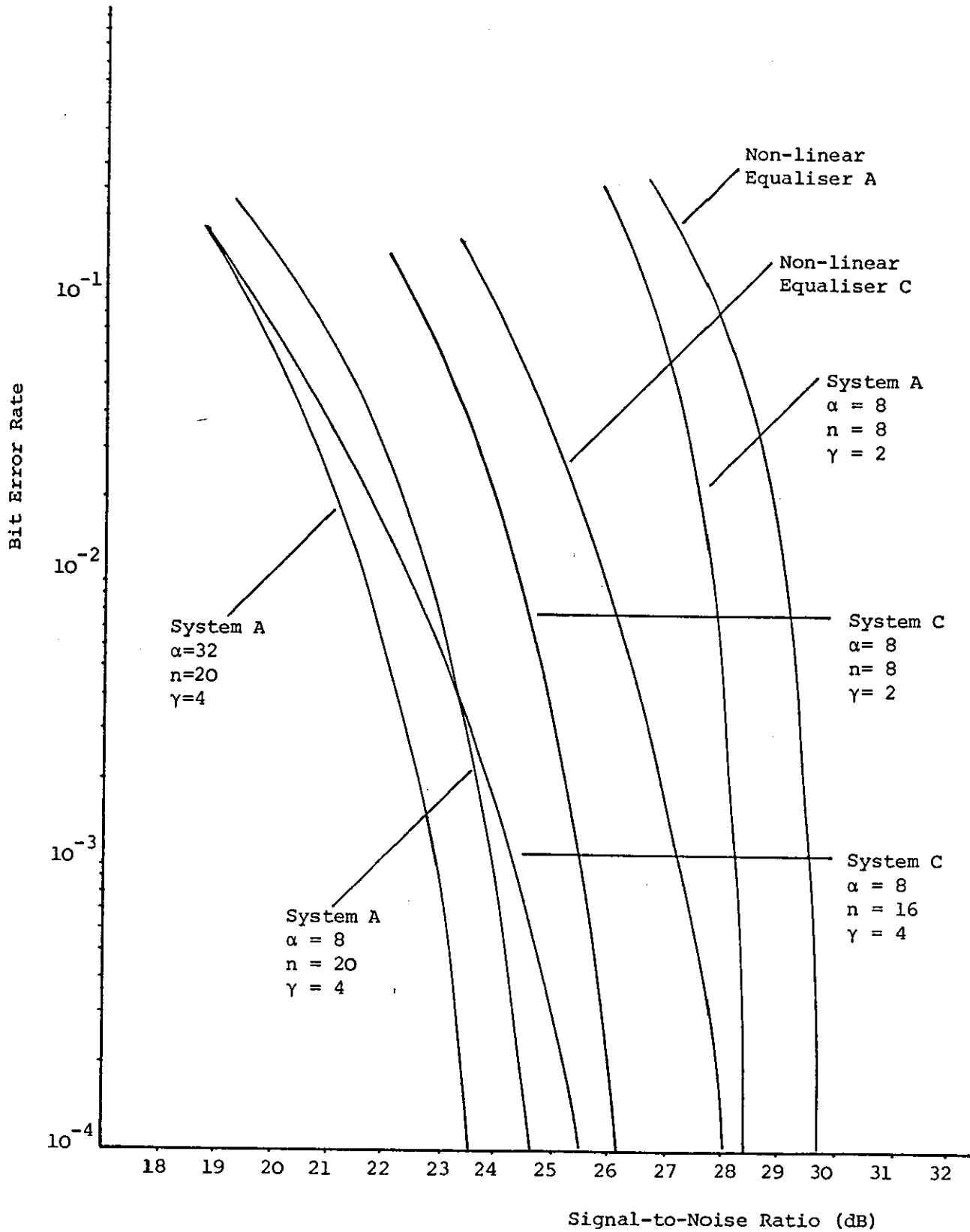


Figure 6.6.1: Variation of bit-error rate for systems A and C, over telephone circuit 1.

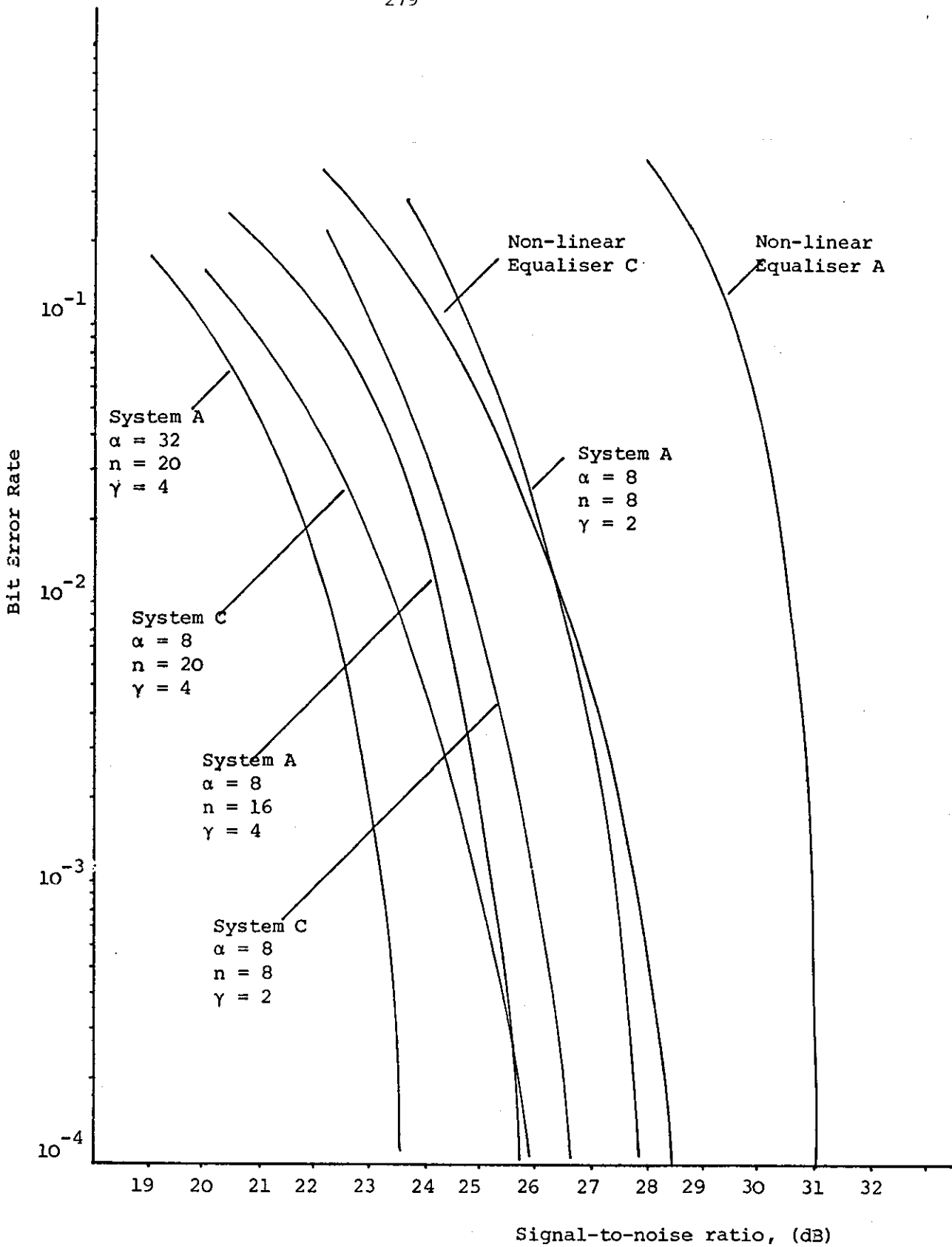


Figure 6.6.2: Variation of bit-error rate for systems A and C over telephone circuit 2

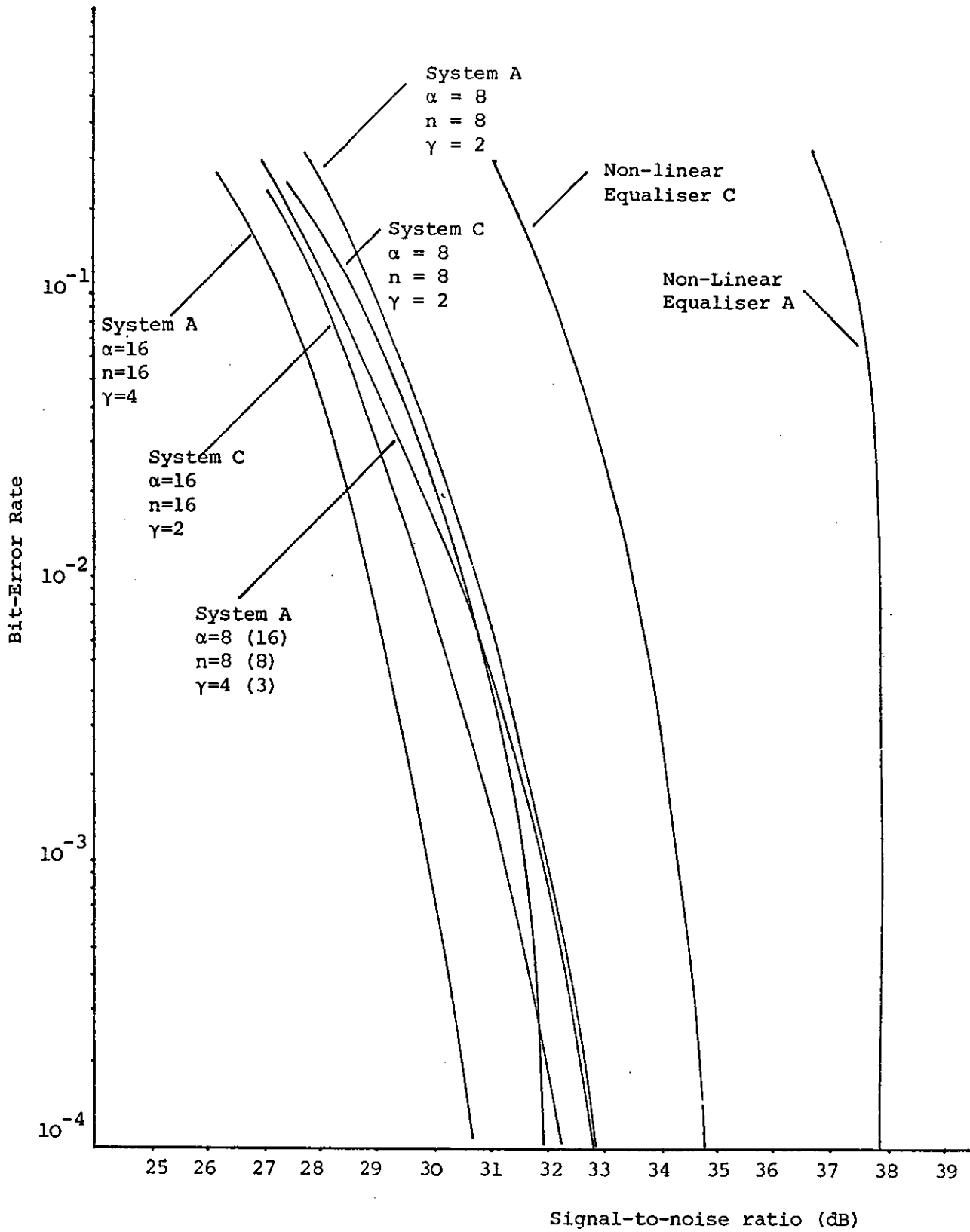


Figure 6.6.3: Variation of bit-error rate for systems A and C,
 over telephone circuit 3

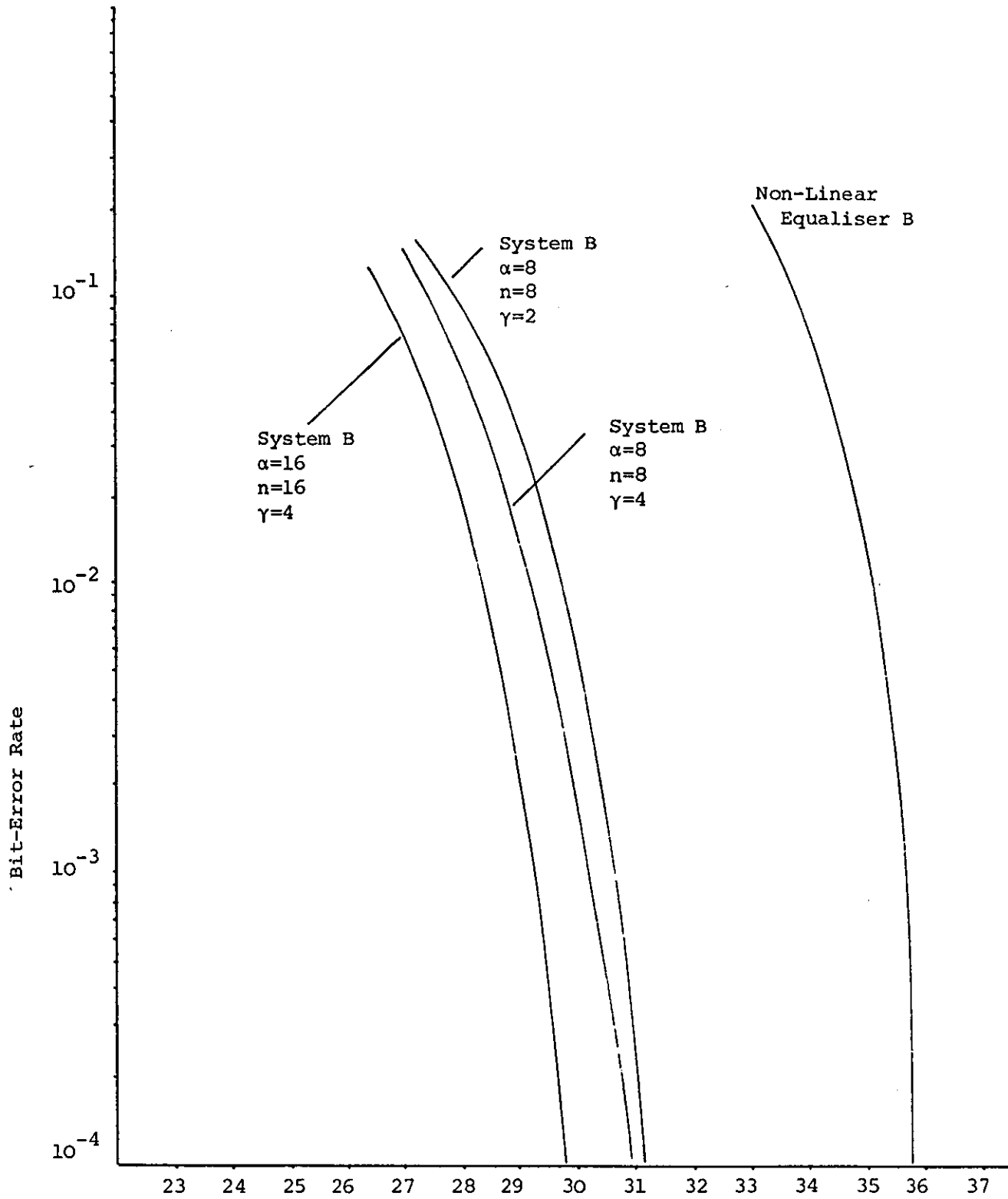


Figure 6.6.4: Variation of bit-error rate for system B,
over telephone circuit 3

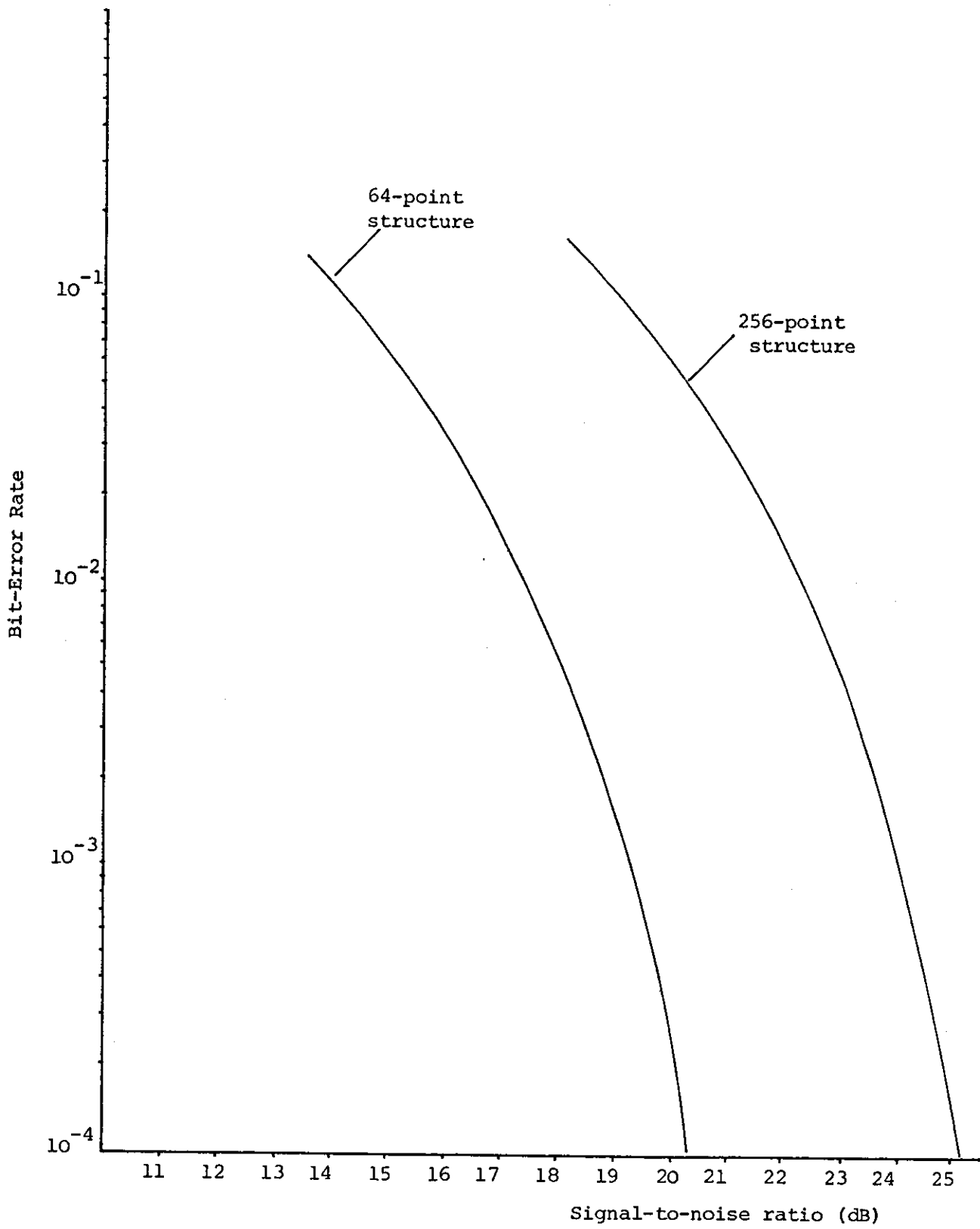


Figure 6.6.5: Variation of bit-error rate for the two QAM systems when operating over an ideal channel

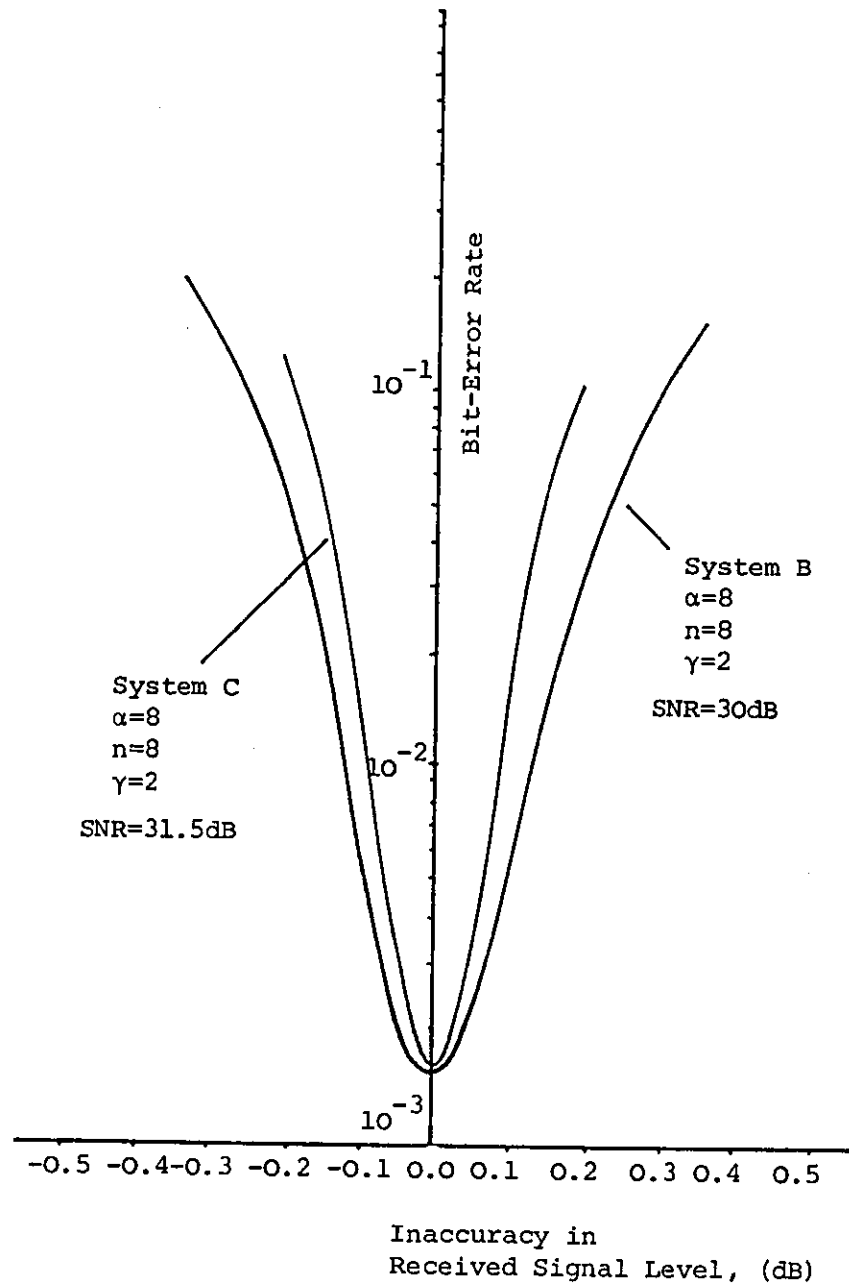


Figure 6.6.6: Variation of error-rate with an inaccuracy in the estimate of the level of the received signal, over telephone circuit 3

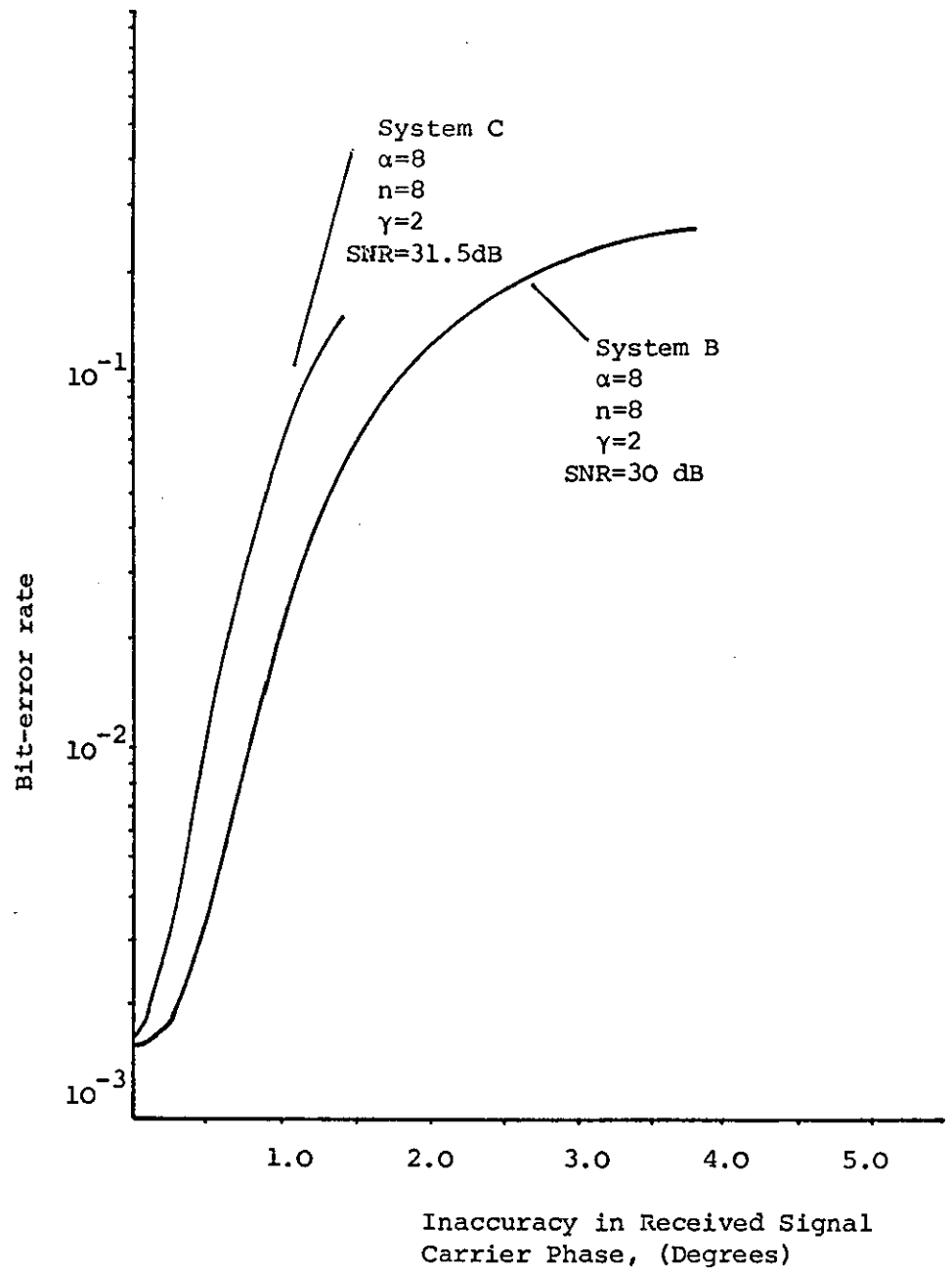


Figure 6.6.7: Variation of error-rate with an inaccuracy in the estimate of the carrier phase of the received signal, over telephone circuit 3

7. Adaptive Adjustment of the Detector and the Channel Estimation Problem

7.1 Introduction

The detection processes discussed in Chapter 6 and the results derived therein suggest that a useful advantage in tolerance to Gaussian noise may be achieved by using a near-maximum-likelihood detection process in place of a conventional non-linear equaliser. Both detection schemes require the setting of the tap gains associated with the linear pre-filter and the decision feedback equaliser. The near-maximum-likelihood detector also needs an accurate knowledge of the sampled impulse response of the system at the input to the detector itself.

It has been shown in Chapter 3 that the characteristics of telephone circuits cannot be assumed ideal nor constant for different connections via the PSTN. Similarly, the characteristics of high-quality private circuits, although complying with published limits, cannot be considered perfect nor time invariant. Consequently, if a data detector is to take full account of signal distortion introduced by different telephone circuits, the receiver must be adaptive in the sense that it must adjust the settings of its equalisers and the estimate of the overall channel sampled impulse response to the conditions which prevail just prior to, and during the actual transmission of data. This implies that the receiver must be "trained" prior to the transmission of valid data to enable it to derive the required equaliser settings and that it must also track any variations in the channel response during the transmission period. Intuitively, the training period, which usually consists of the transmission of a data sequence known to the receiver, should be as short as possible but consistent

with the adjustment time required by the receiver. In the steady-state mode, the adjustment system should be able to follow any long-term variations in the channel response without however, attempting to track very short-term variations as for example, may be simulated by impulsive noise.

Figure 7.1 shows the general arrangement of the adaptive detection scheme. When configured as a non-linear equaliser, the actual detector uses a simple threshold technique and so does not require knowledge of the sampled impulse response of the overall channel. However, when a near-maximum-likelihood detector is used, an estimate of the sampled impulse response is required by the detector (equation (6.5.10)) and the delay in detection must also be taken into account, as indicated in Figure 7.1.

This chapter first discusses the adaptive estimation of the sampled impulse response of the channel as viewed at the input to the detector itself, and then extends the discussion to include the adaptive adjustment of the decision feedback equaliser. A combined adjustment system is then presented for use in receivers employing a near-maximum-likelihood detector. Finally, four new methods are introduced and compared in terms of computational efficiency and speed of operation.

7.2 Channel Estimation Using the Mean-Square-Error Criterion

Figure 7.2.1 shows in block diagram form the basic method used in the estimation of the sampled impulse response of a channel. To ease the understanding of the method, no particular type of detector is assumed in the figure and so any delay in detection is neglected; it is assumed, however, that the detected data symbols, $\{s'_i\}$, are equal to the transmitted symbols, $\{s_i\}$.

The channel estimator is a linear transversal filter whose tap gains, when perfectly adjusted, are equal to the components of the sampled impulse response at a time $t=iT$, Y_i , where;

$$Y_i = [y_{i,0} \quad y_{i,1} \quad \dots \quad y_{i,g}] \quad (7.2.1)$$

The subscript i is included here to allow the sampled impulse response to be slowly time varying. When perfectly adjusted, the tap gains of the linear filter are therefore identical to the tap gains of the linear filter used in the discrete time model of the equivalent baseband system, Figure 6.2.4

The channel estimator described here is a development of the basic form given by Magee and Proakis^(1,2) which has been modified to allow for the complex-valued sampled impulse responses considered in this work. From the detected data symbols $s'_i, s'_{i-1} \dots s'_{i-g}$, the detector first forms a noise-free estimate of the received sample, r_i denoted here as r'_i . The estimator then adjusts the tap gains of the linear filter such that the mean-square-error between the actual received samples and their estimates is minimised.

At time $t=iT$, the mean square error is,

$$\epsilon_i = E \left[\left| r_i - \sum_{h=0}^g s'_{i-h} y'_{i,h} \right|^2 \right] \quad (7.2.2)$$

where $E(\cdot)$ denotes the expectation of (\cdot) .

The iterative method of tap adjustment used here is the so called steepest descent or gradient algorithm, where the $(k+1)^{th}$ tap coefficient is changed in a direction opposite to the gradient $\frac{\partial \epsilon_i}{\partial y'_{i,k}}$

and where the amount of change is directly proportional to the gradient.

The update equation used is,

$$y'_{i+1,k} = y'_{i,k} - \frac{a \partial \epsilon_i}{2 \partial y'_{i,k}} \quad (7.2.3)$$

where "a" is a small positive constant.

If $\nabla(\epsilon_i, y'_i)$ defines the $(g+1)$ -component row vector given by,

$$\nabla(\epsilon_i, y'_i) = \left[\frac{\partial \epsilon_i}{\partial y'_{i,0}} \quad \frac{\partial \epsilon_i}{\partial y'_{i,1}} \quad \dots \quad \frac{\partial \epsilon_i}{\partial y'_{i,g}} \right] \quad (7.2.4)$$

then equation (7.2.3) may be written in vector form as,

$$y'_{i+1} = y'_i - a \nabla(\epsilon_i, y'_i) \quad (7.2.5)$$

where y'_i is defined in equation (7.2.1).

The following analysis gives the full derivation of the steepest descent algorithm for the channel estimator used here. Consider initially the update equation for the $(k+1)^{\text{th}}$ tap coefficient, given in equation (7.2.3). Since the $\{y_{i,k}\}$ are in general complex, then the gradient must be expanded as,

$$\frac{\partial \epsilon_i}{\partial y_{i,k}} = \frac{\partial \epsilon_i}{\partial \text{Re}(y_{i,k})} + j \frac{\partial \epsilon_i}{\partial \text{Im}(y_{i,k})} \quad (7.2.6)$$

which is consistent with the theories of vector calculus when a complex variable, Z , is defined as a point in a two-dimension number space.

$\text{Re}(\cdot)$, $\text{Im}(\cdot)$ denote the real and imaginary parts of (\cdot) , respectively, and $j = \sqrt{-1}$.

From equation (7.2.2),

$$\epsilon_i = E(|e_i|^2) = E(e_i e_i^*) \quad (7.2.7)$$

$$\text{where } e_i = r_i - \sum_{h=0}^g s_{i-h} y'_{i,h} \quad (7.2.8)$$

$$\epsilon_i =$$

and where it is assumed that the $\{s'_i\} = \{s_i\}$, which is quite reasonable when the system is operating at a high signal-to-noise ratio. Applying equations (7.2.6) and (7.2.8) to equation (7.2.7) gives,

$$\frac{\partial(e_i e_i^*)}{\partial y'_{i,k}} = -(s_{i-k})^* e_i - (s_{i-k}) e_i^* + j(j(s_{i-k})^* e_i - j(s_{i-k}) e_i^*)$$

from which,

$$\frac{\partial(e_i e_i^*)}{\partial y'_{i,k}} = -2e_i s_{i-k}^* \quad , \quad k=0,1, \dots g \quad (7.2.9)$$

Hence, from equation (7.2.7),

$$\frac{\partial \epsilon_i}{\partial y'_{i,k}} = -2E(e_i s_{i-k}^*) \quad , \quad k=0,1, \dots g \quad (7.2.10)$$

and substitution in equation (7.2.3) gives,

$$y'_{i+1,k} = y'_{i,k} + aE(e_i s_{i-k}^*) \quad (7.2.11)$$

It should be clear from equations (7.2.2), (7.2.10) and (7.2.11) that the mean square error between the actual and estimated received samples, r_i and r'_i , respectively, will be minimum when $\frac{\partial \epsilon_i}{\partial y'_{i,k}} = 0$, which implies that $E(e_i s_{i-k}^*) = 0$. This follows since the second derivative of ϵ_i with respect to $y'_{i,h}$ is $\frac{\partial^2 \epsilon_i}{\partial y'^2_{i,h}} = 2E(|s_{i-k}|^2)$ which is clearly greater than zero for all $y'_{i,h}$. Due to the noise term present in r_i , the error signal, e_i , will not necessarily be zero when $y'_i = y_i$ and so $E(e_i s_{i-k}^*) = 0$ implies that e_i and s_{i-k}^* must be orthogonal. Under this condition, from equation (7.2.11), $y'_{i+1,k} = y'_{i,k}$ and so provided that y'_{i+1} is time invariant, no further change is made to the estimator's tap gains and the algorithm is said to have converged. It now remains to show that when $E(e_i s_{i-k}^*) = 0$, $y'_{i,k} = y_{i,k}$, for $k=0,1, \dots g$, such that

the estimator's tap gains are optimally adjusted.

From equation (7.2.8),

$$E(e_i s_{i-k}^*) = E\left((r_i - \sum_h s_{i-h} y'_{i,h}) s_{i-k}^*\right) \quad (7.2.12)$$

$$= E\left(r_i s_{i-k}^*\right) - E\left(\sum_h s_{i-h} s_{i-k}^* y'_{i,h}\right) \quad (7.2.13)$$

$$= E\left(\sum_h s_{i-h} s_{i-k}^* y_{i,h} + w_i s_{i-k}^*\right) - E\left(\sum_h s_{i-h} s_{i-k}^* y'_{i,h}\right) \quad (7.2.14)$$

when $E(e_i s_{i-k}^*) = 0$,

$$\sum_h y'_{i,h} E(s_{i-h} s_{i-k}^*) = \sum_h y_{i,h} E(s_{i-h} s_{i-k}^*) + E(w_i s_{i-k}^*) \quad (7.2.15)$$

If the sample values of the zero-mean additive Gaussian noise is uncorrelated with the detected sequence, then $E(w_i s_{i-k}^*) = 0$. Furthermore, if the data symbols themselves are uncorrelated, then,

$$E(s_{i-h} s_{i-k}^*) = b \delta_{h,k} \quad (7.2.16)$$

where $\delta_{h,k}$ is the Kronecker delta and is defined as,

$$\delta_{h,k} \left\{ \begin{array}{l} = 1, \quad h=k \\ = 0, \quad h \neq k \end{array} \right. \quad (7.2.17)$$

and where $b = E(|s_{i-k}|^2)$

Equation (7.2.15) then becomes,

$$\sum_h y'_{i,h} b \delta_{h,k} = \sum_h y_{i,h} b \delta_{h,k}, \quad k=0,1, \dots, g \quad (7.2.18)$$

and so the tap gains of the equaliser are identical to respective components of the channel sampled impulse response. Under these conditions, the

mean square error, ϵ_i , will be a minimum and from equation (7.2.2) will equal $E(|w_i|^2)$; since the noise process has a zero mean value, then $E(|w_i|^2)$ is simply the noise variance.

If F is the $(g+1)$ -component vector defined as;

$$F = [E(e_i s_i^*) \ E(e_i s_{i-1}^*) \ \dots \ E(e_i s_{i-g}^*)] \quad (7.2.19)$$

then equation (7.2.11) may be written in vector form as

$$y'_{i+1} = y'_i + aF \quad (7.2.20)$$

Equations (7.2.11) and (7.2.20) therefore give the steepest descent algorithm for the adjustment of the channel estimator's tap gains. In the latter equation, F can be thought of as the gradient vector (strictly, $G = -\frac{1}{2}F$ is the true gradient vector, as can be seen from equation (7.2.10)) and so an adaptive channel estimator operating under this algorithm would need to calculate the expectations given in equation (7.2.19). To avoid this complication, it is usual to involve an estimate of the gradient vector, rather than the vector itself. The estimate used here is,

$$\hat{F} = [e_i s_i^* \ e_i s_{i-1}^* \ \dots \ e_i s_{i-g}^*] \quad (7.2.21)$$

and so the update equations become,

$$y'_{i+1,k} = y'_{i,k} + a e_i s_{i-k}^* \quad (7.2.22)$$

$$y'_{i+1} = y'_i + a\hat{F} \quad (7.2.23)$$

Figure 7.2.2 shows the adaptive channel estimator described above.

The parameter "a" controls the speed of convergence and the effect of additive noise on the estimates; the smaller the value of "a", the slower is the rate of convergence but the more accurate is the eventual steady-state estimate. Clearly, "a" need not be a fixed value, so a large value of

"a" during the training period could give an increased rate of convergence whilst a small value during steady-state operation will give improved accuracy in the estimate^(1,4).

In the form presented in Figure 7.2.2, the estimator has to perform $2(g+1)$ complex multiplications for the actual tap gain adjustment plus one complex multiplication in the derivation of the correction signal, ae_1 , for every received sample. When operating at low symbol error rates, the major cause of error signal variation with time is due to the noise components present in the $\{r_1\}$. The estimator parameter "a", often called the averaging or integration parameter, is usually a fairly small number (of the order 1×10^{-3} here) and determines the amount of adjustment made to the tap accumulators. Clearly, at high signal-to-noise-ratios and in the absence of any other signal impairments, the rate of change of the tap gains will be small, assuming steady-state operation. Consequently, after initial convergence of the estimator has been achieved, the processor which performs the estimation process may be allowed to idle, the actual adjustment of the tap gains only being performed after the error signal exceeds some pre-determined limit or after a given number of symbols have been processed and detected. Depending upon the actual symbol error rate, signal-to-noise ratio and levels of other signal impairments introduced by the telephone circuit, this technique may save a considerable number of operations. For practical systems, it is unlikely that a single processor will be powerful enough (or fast enough) to perform both detection and estimation and so a separate processor operating in parallel to the detector algorithm will usually be necessary. However, as the parallel processor may well be able to perform

other functions, such as the control of the carrier phase tracking loop described in the following chapter, any operation saving technique will be worth consideration when attempting to implement a practical modem.

The estimation process described above is probably the least complex of all the possible methods and involves relatively few operations per iteration. Many other channel estimation techniques have been reported in the literature⁽²⁻¹²⁾ for operation over, for example, rapidly fading H.F. radio links^(4,6,8). Several sophisticated and very effective techniques have also been developed by Harvey⁽³⁾ which have been shown to possess faster convergence times than the gradient estimator but for greatly increased complexity. Due to this increased complexity and the observable fact⁽³⁾ that when operating in the steady-state mode, these faster algorithms do not give a significant improvement in the accuracy of the final estimate nor of the tolerance to signal impairments experienced over telephone circuits, the simple gradient algorithm estimator can be considered as a suitable method for the initial development of a 19,200 bit/s modem. It will be shown later that this type of estimator, when initially operating with no prior knowledge of the sampled impulse response of telephone circuit 3, will converge to an accurate estimate after less than 500 symbol periods. For system b), this represents a convergence time of less than 150 ms which is considered sufficiently small for the current development. However, if it is found that the modem training period is excessive for particular applications or if extra processing power is available at the receiver, it is recommended that the works of Harvey,⁽³⁾ or Clark, et. al.⁽²⁶⁾ are consulted for alternative schemes.

7.3 Adaptive Decision - Feedback Equalisation

As will be recalled from Section 6.3.2, a pure non-linear equaliser operates by attempting to remove intersymbol interference components from the received samples, $\{r_i\}$. From equation (6.3.17),

$$r_i = s_i y_{i,0} + \sum_{h=1}^g s_{i-h} y_{i,h} + u_i \quad (7.3.1)$$

where

$$\sum_{h=1}^g s_{i-h} y_{i,h}$$

represents the intersymbol interference present on the i^{th} received sample. Equation (7.3.1) and Figure 6.3.3 show that the non-linear equaliser requires an estimate of the sampled impulse response of the baseband channel just prior to the detector itself^(2,3,13). Figure 7.3.1 shows the adaptive form of the decision-feedback equaliser where the estimate of y_i is derived from the steepest descent channel estimator considered in the previous section. Figure 7.3.1 actually shows the case of pure decision feedback equalisation, where all of the interfering components are removed from the $\{r_i\}$ and where the detected data symbols are obtained by a process of simple threshold detection. The figure also assumes that the first component of y_i is $y_{i,0} = 1 + j0$, which occurs naturally when the linear feedforward transversal filter forming the first part of the equaliser is adjusted to minimise the mean-square error in the equalised signal⁽¹³⁾. As can be seen from Figure 7.3.1, only an extra adder (accumulator) plus the canceller/subtractor need be included in the basic estimator structure to give the non-linear equaliser; the number of operations, if approximated by the number of multiplications, therefore remains at $(2g + 3)$.

Figure 7.3.2 shows the general arrangement of the estimation system when including a near-maximum-likelihood detector and intersymbol interference cancellation. It is assumed here that the delay in detection is equal to n symbol periods and that the sampled impulse response of the system prior to cancellation, y_i , has $(g+1)$ components, where $y_{i,0}$ is constrained to the value $1+j0$.

The received sample at time $t=iT$ is,

$$r_i = \sum_{h=0}^g s_{i-h} y_{i,h} + w_i \quad (7.3.2)$$

Due to the detection delay, the detected symbol at time $t=iT$, is s'_{i-n} . The error signal at this time, e_i , is therefore,

$$e_i = (x_{i-n} - q'_i) \quad (7.3.3)$$

as shown in Figure 7.3.2, where

$$x_{i-n} = r_{i-n} - \sum_{h=n+1}^g s'_{i-n-h} y'_{i,h} \quad (7.3.4)$$

$$q'_i = \sum_{h=0}^n s'_{i-n-h} y'_{i,h} \quad (7.3.5)$$

and so,

$$e_i = (r_{i-n} - r'_{i-n}) \quad (7.3.6)$$

where, from equations (7.3.3) to (7.3.5)

$$r'_{i-n} = \sum_{h=0}^g s'_{i-h-n} y'_{i,h} \quad (7.3.7)$$

As before, $y_{i,h}$ and $y'_{i,h}$ are the actual and estimated values of the $(h+1)^{\text{th}}$ component of the sampled impulse response of the baseband channel and x_i is the input sample to the detector, at time $t=iT$. It is important to note from equation (7.3.6) that the error signal at

time $t=iT$ is actually derived from delayed versions of r_i and r'_i , due to the delay in detection of n sample intervals. Also, if $y'_{i,0}$ is constrained to equal $y_{i,0} = 1+j0$, then the error signal at time $t=iT$ becomes,

$$e_i = \left\{ r_{i-n} - s'_{i-n} - \sum_{h=1}^g s'_{i-n-h} y'_{i,h} \right\} \quad (7.3.8)$$

Following the analysis given in Section 7.2, the steepest descent tap update algorithm for the system is,

$$y'_{i+1,k} = y'_{i,k} + a e_i s'^*_{i-n-k}, \quad k=1,2, \dots, g \quad (7.3.9)$$

Figure 7.3.3 shows the adaptive system described by equation (7.3.9), where it can be seen that the $(k+1)^{th}$ tap of the estimator is updated by using the detected symbol s'_{i-n-k} . The first $(n+1)$ components of the estimated sampled impulse response are fed to the detector in readiness for the detection of s'_{i-n+1} , whilst the remaining $(g-n)$ components are used by the intersymbol interference canceller to remove from the received samples, $\{r_i\}$, estimates of all interfering components whose symbol values have already been determined by the detector. The sampled impulse response at the input to the detector is now reduced from $(g+1)$ to $(n+1)$ components, as can be seen from equation (7.3.4).

It should be clear from equations (7.3.8) and (7.3.9) that the estimate of the sampled impulse response at time $t=iT$, $\{y'_{i,h}\}$, is actually an estimate of $\{y_{i-n,h}\}$, due to the delay in detection of n sample periods. For time invariant channels, or for channels which vary only slowly with time, as is assumed here, the estimate is usually a very close approximation to the $\{y_{i,h}\}$. Clearly, if the sampled impulse response varies considerably with time, $\{y'_{i,h}\}$ can be very different from $\{y_{i,h}\}$. In such cases, either prediction must be included in the estimator structure or a method of early detection must

be used in the evaluation of the estimator's error signal^(4,6,8).

7.4 Combined Adjustment of the Linear Pre-filter, Channel Estimator and Cancellation Equaliser.

Figure 7.4.1 shows the general arrangement for the combined adjustment of an adaptive near-maximum likelihood detector which includes a linear pre-filter and cancellation of intersymbol interference. The adjustment method for the channel estimator and decision feedback equaliser is identical to that considered earlier except that the estimator now determines an estimate of the sampled impulse response of the baseband channel plus the linear pre-filter. When correctly adjusted, the sampled impulse response of the channel plus filter will, at high signal-to-noise ratios, have a z-transform whose roots (zeros) all lie within the unit circle in the z-plane^(3,13).

Let the following row vectors be defined,

$$\left. \begin{aligned} Y_i &= [y_{i,0} \quad y_{i,1} \quad y_{i,2} \cdots y_{i,g}] \\ C_i &= [c_{i,0} \quad c_{i,1} \quad c_{i,2} \cdots c_{i,\mu-1}] \\ B_i &= [b_{i,0} \quad b_{i,1} \quad b_{i,2} \cdots b_{i,g}] \\ P &= [p_i \quad p_{i-1} \quad p_{i-2} \cdots p_{i-\mu-n}] \end{aligned} \right\} \quad (7.4.1)$$

Where the $\{y_{i,j}\}$ are the components of the sampled impulse response of the linear baseband channel, the $\{c_{i,j}\}$ are the tap coefficients of the linear pre-filter and the $\{b_{i,j}\}$ are the tap coefficients of the channel estimator, at a time $t=iT$. As will be seen later, the number of non-zero components present in the sampled impulse response at the output of the pre-filter is identical to that at its input, which is $(g+1)$ components here. The pre-filter does, however, introduce a delay. If p_i is the input sample to the pre-filter at time $t=iT$, the

output sample will be r_{i-h} , where the delay through the filter is assumed to be h sample periods. For convenience, this delay is neglected here so that,

$$r_i = \sum_{j=0}^{\mu-1} p_{i-j} c_{i,j} + u_i \quad (7.4.2)$$

Also, the sampled impulse response at the output of the pre-filter is assumed to be $b_0 \neq 0$ and $b_i = 0$, for $g < i < 0$. Consequently, the channel estimator, which now estimates the sampled impulse response of the baseband channel and pre-filter, is also assumed to have $(g+1)$ tap coefficients.

The adaptive pre-filter itself is assumed to have μ tap coefficients, where for accurate equalisation, $\mu \rightarrow \infty$. It follows from earlier work that p_i in equation (7.4.2) is,

$$p_i = \sum_{k=0}^g s_{i-k} y_{i,k} + w_i \quad (7.4.3)$$

and so u_i in equation (7.4.2) is the noise component at time $t=iT$ at the output of the pre-filter.

The output samples from the pre-filter, the $\{r_i\}$, are then applied to the non-linear part of the detector and are operated upon as described in Section 7.3. Figure 7.3.3 still applies here, but the estimator tap coefficients should be replaced by $\{b_{i,j}\}$. If the delay in detection is again assumed to be n sample periods, then the error signal generated by the adaptive process is now,

$$e_i = (x_{i-n} - q'_i) \quad (7.4.4)$$

where

$$x_{i-n} = \left(r_{i-n} - \sum_{j=n+1}^g s'_{i-n-j} b_{i,j} \right) \quad (7.4.5)$$

and

$$q'_i = \sum_{j=0}^n s'_{i-n-j} b_{i,j} \quad (7.4.6)$$

The mean square error is, therefore,

$$\epsilon_i = E(|x_{i-n} - q'_i|^2) \quad (7.4.7)$$

which, from equation (7.4.2), can be written,

$$\epsilon_i = E\left(\left|\sum_{j=0}^{\mu-1} p_{i-n-j} c_{i,j} + u_i - \sum_{j=0}^g s'_{i-n-j} b_{i,j}\right|^2\right) \quad (7.4.8)$$

where it should be realised that $\epsilon_i = E[|e_i|^2]$ is the mean square error calculated between r_{i-n} and an estimate of r_{i-n} , r'_{i-n} , at time $t=iT$, due to the delay in detection.

The pre-filter studied here is adjusted using a gradient algorithm to minimise the mean square error between r_{i-n} and its estimate, r'_{i-n} , where, from equations (7.4.4) to (7.4.6) it can be seen that,

$$r'_{i-n} = \sum_{j=n+1}^g s'_{i-n-j} b_{i,j} - q'_i \quad (7.4.9)$$

$$r'_{i-n} = \sum_{j=0}^g s_{i-n-j} b_{i,j} \quad (7.4.10)$$

Consequently,

$$c_{i+1,k} = c_{i,k} - \frac{a_2}{2} \frac{\partial \epsilon_i}{\partial c_{i,k}}, \text{ for } k=0,1, \dots, \mu-1 \quad (7.4.11)$$

$$\text{or } C_{i+1} = C_i - \frac{a_2}{2} \nabla(\epsilon_i, C_i) \quad (7.4.12)$$

where C_i is defined in equation (7.4.1),

$$\nabla(\epsilon_i, C_i) = \left[\frac{\partial \epsilon_i}{\partial c_{i,0}} \quad \frac{\partial \epsilon_i}{\partial c_{i,1}} \quad \dots \quad \frac{\partial \epsilon_i}{\partial c_{i,\mu-1}} \right] \quad (7.4.13)$$

and a_2 is a small positive constant.

Following the analysis given in Section 7.2 and noting that,

$$\frac{\partial \epsilon_i}{\partial c_{i,k}} = \frac{\partial (e_i e_i^*)}{\partial \text{Re}(c_{i,k})} + j \frac{\partial (e_i e_i^*)}{\partial \text{Im}(c_{i,k})} \quad (7.4.14)$$

then

$$\begin{aligned} \frac{\partial \epsilon_i}{\partial c_{i,k}} &= E \left[e_i(p_{i-n-k})^* + e_i^*(p_{i-n-k}) + j \left(e_i(-jp_{i-n-k})^* + e_i^*(jp_{i-n-k}) \right) \right] \\ &= E(2e_i p_{i-n-k}^*) \end{aligned} \quad (7.4.15)$$

The update equation for the pre-filter tap coefficients is, therefore, from equations (7.4.11) and (7.4.12),

$$c_{i+1,k} = c_{i,k} - a_2 E(e_i p_{i-n-k}^*), \quad k=0,1, \dots, \mu-1 \quad (7.4.16)$$

or
$$c_{i+1} = c_i - a_2 G_i$$

where
$$G_i = \begin{bmatrix} E(e_i p_{i-n}^*) & E(e_i p_{i-n-1}^*) & \dots & E(e_i p_{i-n-\mu+1}^*) \end{bmatrix} \quad (7.4.17)$$

For similar reasons as given in Section 7.2, $e_i p_{i-n-k}^*$ is used as an estimate of $E(e_i p_{i-n-k}^*)$ and so,

$$c_{i+1,k} = c_{i,k} - a_2 e_i p_{i-n-k}^*, \quad k=0,1, \dots, \mu-1 \quad (7.4.18)$$

or
$$c_{i+1} = c_i - a_2 \hat{G} \quad (7.4.19)$$

where

$$\hat{G} = \begin{bmatrix} e_i p_{i-n}^* & e_i p_{i-n-1}^* & \dots & e_i p_{i-n-\mu+1}^* \end{bmatrix} \quad (7.4.20)$$

When adjusted in accordance with equation (7.4.18), the linear pre-filter, at convergence, will be the filter which replaces roots of the z-transform of the sampled impulse response of the baseband channel, $Y_i(z)$, which lie outside the unit circle in the z-plane, by the complex conjugates of their reciprocals, which is a process of pure phase transformation⁽¹³⁾.

Figure 7.4.2 shows the adjustment system for a pre-filter which obeys the algorithm given by equation (7.4.18), where it should be observed that

the $(j+1)^{\text{th}}$ tap coefficient at time $t=iT$, $c_{i,j}$, is updated by the $(i-j-n)^{\text{th}}$ signal sample; that is, the signal samples are delayed by n sample intervals to allow for the delay in detection. It should also be clear from Section 7.3 and equation (7.4.8) that the algorithm for the adjustment of the channel estimator/cancellation filter is,

$$b_{i+1,k} = b_{i,k} + a e_i s_{i-n-k}^*, \quad k=1,2, \dots, g \quad (7.4.20)$$

which is identical to equation (7.3.9) when the $\{y_{i,k}\}$ are replaced with the $\{b_{i,k}\}$.

The complete adjustment system involves both Figures 7.3.3 and 7.4.2, where the $\{y_{i,k}\}$ should be replaced by the $\{b_{i,k}\}$ in the former figure to allow for the presence of the adaptive pre-filter. In most practical applications, the two averaging parameters, a and a_2 , are replaced by a single parameter which controls the rate of convergence and the accuracy of the final estimates. However, due to the larger number of tap coefficients in the pre-filter compared with the estimator, the latter will take a larger time to converge and so it may be beneficial to use the two averaging values independently. For tap coefficient initialisation, a training sequence is usually transmitted prior to the transmission of data. The receiver is assumed to have full knowledge of the training sequence and can therefore perform the initial adjustment of the pre-filter and equaliser^(2,3). A very comprehensive study of start-up techniques can be found in reference 3.

As mentioned earlier, various techniques have been published in the literature which can give a considerable reduction in convergence time and therefore require shorter training sequences^(2,3,26). The fast Kalman algorithm, for example, achieves very rapid convergence when operation over typical telephone circuits and would therefore be worth further consideration for applications such as polled modems^(2,3).

However, the reduction in convergence time is greatly offset by the increase in complexity. Harvey⁽³⁾ suggests that such rapid convergence algorithms could be used during initial modem training to shorten the training period whilst a simple gradient algorithm could be used to maintain receiver adjustment when operating in the steady-state mode. Clearly, this is a technique worthy of consideration for a prototype modem based around a software implementation.

7.5 Performance of the Adaptive Adjustment System

In order to evaluate the performance of the adaptive adjustment system described in Section 7.4, computer simulations have been carried out to determine both convergence rate and accuracy of the adaptive processes. The program structure used in these tests is shown as Program 6 in Appendix 10. Based upon the results and discussion of Chapter 6, system b) was used for the computer simulation tests where the data is transmitted over telephone circuit 3; this circuit being selected as the typical worst-case circuit under consideration. It is shown later that telephone circuit 3 can present problems to the pre-filtering process since many of the roots of the z-transform of its sampled impulse response lie outside, but close to the unit circle in the z-plane. In all cases, a good start-up procedure was used prior to the transmission of random data, the start-up sequence being known at the receiver such that all vectors could be filled with relevant information prior to operation in the steady-state mode. The measure of performance of the system when under different operating conditions was based on the mean-square-error between the known sampled impulse response of the time invariant channel, Y_i , and the estimated sampled impulse response, Y'_i , or alternatively, between the calculated sampled

impulse response of the channel and pre-filter, B_i , and its estimate, B_i' , where mean square error is defined here as,

$$MSE = \frac{1}{(g+1)} \sum_{h=0}^g (|Y_i - Y_i'|^2)$$

or

$$MSE = \frac{1}{(g+1)} \sum_{h=0}^g (|B_i - B_i'|^2) \quad (7.5.1)$$

Figures 7.5.1 to 7.5.3 show the performance of the adjustment system when estimating the sampled impulse of telephone circuit 3 before the linear filter, (that is, the "raw" sampled impulse response), at signal-to-noise ratios of 20, 31 and 80 dB respectively, where signal-to-noise ratio is defined in Appendix 3. Clearly, the operation of the system at a signal-to-noise ratio as low as 20 dB is unrealistic for the detection systems of Chapter 6 and so for these tests, the detector is assumed to be operating ideally such that the effect of the noise on the actual estimation process can be determined. Also, for the tests depicted in Figures 7.5.1 to 7.5.8, the pre-filter is assumed to be perfectly adjusted.

The first set of results, Figures 7.5.1 to 7.5.3, illustrate the performance of the system when the tap-gains of the channel estimator are initially set to zero, whilst Figures 7.5.4 to 7.5.5 assume that the estimator has some prior knowledge of the channel but at a different sampling phase. This might be the case when operating over a permanently rented private circuit. Comparisons between appropriate figures show that very little advantage, if any, is gained by using this "pre-loading" technique.

Comparison between Figures 7.5.1 to 7.5.5 reveal some interesting points. Firstly, the initial rate of convergence, up to about 150 transmitted symbols, is apparently independent of signal-to-noise ratio. This in fact, is due to the initial start-up procedure which assumes that the detector has full knowledge of the first 100 data symbols. Also, at signal-to-noise ratios above 31 dB, the detector will be operating at a symbol error rate lower than 1×10^{-4} , (Figure 6.6.4), assuming of course, that accurate estimation of the sampled impulse response has been achieved. Secondly, when operating at a signal-to-noise ratio of 20 dB, the system converges with a relatively high error value, as would be expected from the discussion in Section 7.2. The third point of interest is the variation of convergence rate and steady-state accuracy with changes in the averaging or step-size parameter, a . In all cases, " a " has been maintained at a constant value for the duration of each test. It is clear from the figures that the general effect of increasing " a " is to increase the rate of convergence and to decrease the accuracy of the final estimate, at least over the time periods investigated. Figure 7.5.3 gives a slightly distorted view of the latter effect since the very high signal-to-noise ratio results in a final error of approximately zero; however, the different rates of convergence are clearly illustrated. Curves f) in Figures 7.5.1 to 7.5.8 show that increasing the averaging value 'ad infinitum' does not result in improved convergence rates. Beyond a certain value of step size and especially at low signal-to-noise ratios, the loop becomes over-active and responds too quickly to noise-induced variations in the error signal. When averaged over time, this results in a poorer convergence rate, reduced accuracy of the estimate and the undesirable tracking of transient effects such as a burst of impulsive

noise. An interesting analysis of the stability of decision feedback channel estimators is given in reference 3.

Figures 7.5.6 - 7.5.8 show the performance of the channel estimator when attempting to estimate the sampled impulse response of the baseband channel and linear pre-filter, where again, the pre-filter is assumed to be perfectly adjusted. The observations made above from Figures 7.5.1 to 7.5.5 apply equally to Figures 7.5.6 to 7.5.8, although it should be noted that both the steady-state error and the variation of this error with different values of "a" are lower than those of the earlier figures. This is due to the reduced length of the equalised sampled impulse response and the constraining of the first component of the estimate to $1+j0$.

Figure 7.5.9 shows the performance of the complete adjustment system illustrated in Figures 7.3.3 and 7.4.2. As before, the performance is measured by the error existing between the known sampled impulse response of the bandpass channel and perfectly adjusted linear pre-filter and the estimate of this sampled impulse response provided by the adaptive estimator. Both curves in Figure 7.5.9 use the same absolute value of averaging constant although from equation (7.4.11), the tap gain update signal fed to the adaptive pre-filter will be the negative of that used by the estimator itself. The value 0.0005 was chosen from the results of Figures 7.5.1 - 7.5.8 and gives a good compromise between rate of convergence and accuracy of the steady-state estimate. The signal-to-noise ratios used in Figures 7.5.1 to 7.5.9 are, as throughout this work, based on the average energy per transmitted bit of information and so curve a. in Figure 7.5.9 represents the receiver working at an error rate of around 1×10^{-4} . Comparison between Figures 7.5.7 and 7.5.9 shows that the inclusion of

the adaptive pre-filter in the adjustment system causes both a lengthening of the convergence period and a reduction in the final accuracy of the channel estimate. The former is due to the fact that a single error signal now has to adjust two inter-related adaptive processes whilst the latter is probably due to the truncation of the number of taps used in the pre-filter⁽³⁾. It is clear from Figure 7.5.9 that at least 3000 data symbol intervals are required to allow the adaptive process to converge, which represents slightly less than 1 second for a system operating at 3200 bauds. This is unlikely to cause problems for modems operating over the PSTN where the call set-up time will be an order of magnitude higher, nor over a private circuit which is dedicated to the transfer of data on a point-to-point basis. However, if complete loss of receiver synchronism occurs, due perhaps to a fault condition or equipment change-over, then the convergence rate of the adaptive detector could place a severe constraint on overall system performance. As already mentioned, the use of a fast-convergence algorithm could be employed during start-up or re-training phases but at the expense of a considerable increase in complexity. Whether or not this increase in complexity can be absorbed by the adjustment system is clearly dependent upon the actual processing power available at the receiver, which in turn depends not only on the intended market, but also on the pricing policy of the manufacturer. A further factor which must be considered here is the steady lowering in price of integrated circuits designed for signal processing applications and indeed the increase in processing power available in the latest generation of such devices. Although the availability of such cheap, powerful devices may well solve many problems in modem design over the next few years it is of course vital that design engineers use the increased processing

power to advantage, rather than simply implementing conventional methods with a "brute-force" strategy.

Based on the results of this section, it appears that an adaptive detection system can be implemented for a 19,200 bit/s modem using fairly conventional and simple techniques, although the presence of the adaptive linear filter results in an increased convergence time and reduced accuracy in the estimated sampled impulse response. Clearly, as the detector operates directly on the output of the pre-filter and uses the estimate of the overall sampled impulse response, the accuracy of the adaptive system is a most important factor. The following section reviews the basic requirements of the detection scheme and presents alternative techniques for the derivation of the pre-filter tap-gains and the estimation of the overall sampled impulse response.

7.6 Alternative Adaptive Adjustment Schemes

It will be recalled from Chapters 5 and 6 and earlier sections of this chapter, that the successful operation of the near-maximum likelihood detector of the type considered here depends upon the accurate adjustment of the linear pre-filter and the accurate determination of the sampled impulse response of the baseband channel and linear pre-filter. Ideally, the pre-filter will be an all-pass network with an infinite number of taps and will adjust the sampled impulse of the channel and filter to be minimum phase without, however, changing the amplitude distortion introduced by the channel and without changing the signal-to-noise ratio^(3,13). It has also been shown earlier that when correctly adjusted, the ideal filter achieves an overall minimum phase sampled impulse response by a process of pure phase transformation which locates all roots of the z-transform of the

sampled impulse response of the channel which lie outside the unit circle in the z -plane and replaces these roots by the complex conjugates of their reciprocals. The resulting overall impulse response, as seen by the detector, has a rapid rise time and short duration and so has a concentration of energy towards the first few components. In the case of a non-linear equaliser followed by a simple symbol-by-symbol threshold detector, it is also necessary for the first component of the overall sampled impulse response to have a relatively large value since detection is based entirely on the first component.

The approach adopted so far in this chapter for the adjustment of the linear pre-filter has been to use a fairly simple gradient algorithm which, at the same time, can estimate the sampled impulse response of the channel and filter. It is clear from Figures 7.5.6 - 7.5.9 and in fact can be proved theoretically^(2,13) that as the signal-to-noise ratio increases, the adaptively adjusted linear filter approximates more closely to the ideal filter described above provided the adaptive filter has a sufficiently large number of taps to ensure accurate equalisation. It has also been shown elsewhere⁽³⁾ that the replacement of an adaptively adjusted linear filter by the corresponding ideal filter produces only a small advantage in tolerance to noise when using near-maximum likelihood detection and when operating over good and typical telephone channels. However, when operating over poorer quality telephone circuits, which in general will introduce more severe phase distortion, the increased number of taps required by the filter for accurate equalisation can not only reduce the accuracy of the overall adjustment system, but will also increase the time required for the process to converge. In the limit, when operation is required over very severe circuits and at relatively low signal-to-noise ratios, the

combination of tap requirements, inaccuracies of the final estimate and excessive convergence time may well force the use of more sophisticated adjustment schemes.

The remainder of this chapter is concerned with alternative methods for the joint determination of the pre-filter tap gains and an estimate of the sampled impulse response of the channel and filter. In all cases, these methods make full use of the prior knowledge that the ideal linear filter is an all-pass network which adjusts the overall sampled impulse response to be minimum phase and that, from Figures 7.5.2 to 7.5.8, an accurate estimate of the sampled response of the channel may be obtained in a relatively small time by the use of simple steepest-descent algorithms.

Figure 7.6.1 shows the basic baseband model for the receiver assumed here. As before, the sampled impulse response of the baseband channel is given by the $(g+1)$ -component row vector;

$$Y = [y_0 \quad y_1 \quad y_2 \quad \dots \quad y_g] \quad (7.6.1)$$

whose z -transform is;

$$Y(z) = y_0 + y_1 z^{-1} + y_2 z^{-2} + \dots + y_g z^{-g} \quad (7.6.2)$$

and is therefore a polynomial in z^{-1} of order g , with complex coefficients, $\{y_i\}$, $i = 0, 1, \dots, g$. For practical purposes, Y is considered to be a finite length sampled impulse response and, as before, any delay in transmission introduced by the channel is neglected for convenience.

The channel estimator uses the received samples, $\{p_i\}$ and the detected data symbols, $\{s'_i\}$ to form an estimate of the sampled impulse response of the channel Y' . This estimate is then used to determine the tap gains of the linear pre-filter. When these tap gains have been determined, an estimate of the sampled impulse response of the channel and filter is

derived and used in the partial cancellation of intersymbol interference process along with the $\{s'_i\}$ and by the detector itself. Clearly, the detector must also have knowledge of the possible values of the $\{s_i\}$. It has been shown in Section 7.3 that the performance of the channel estimator is not affected by a delay in detection when the channel characteristics vary only very slowly with time, provided that the delay is taken into account in the determination of the error signal. For this reason, Figure 7.6.1 and the following discussions assume no delay in the detection of the $\{s'_i\}$ from the $\{x_i\}$.

The fundamental difference between the alternative schemes and the conventional method of Section 7.5, is that instead of using a search algorithm to determine the tap gains of the pre-filter which will eventually result in an overall minimum phase sampled impulse response, the former schemes use the fact that the pre-filter must replace all roots of the sampled impulse of the channel which lie outside the unit circle by the complex conjugates of their reciprocals. The problem therefore reduces to locating these roots, equalising them and determining the filter tap gains to force the condition, subject to the limitation of a finite number of taps.

7.6.1 Alternative Scheme 1

This scheme represents initial ideas for alternative adjustment systems and is loosely based on the "desired-impulse-response" approach used elsewhere to reduce the complexity of Viterbi algorithm detectors^(3,13-17).

Figure 7.6.2 shows the general arrangement. When correctly adjusted, the linear filter, which is assumed to have $(l+1)$ taps, will introduce a delay of l sample intervals⁽¹³⁾, which for convenience is neglected here. The tap gains of the pre-filter are given by the $(l+1)$ -component row vector D , where;

$$D = [d_0 \quad d_1 \quad d_2 \quad \dots \quad d_\ell] \quad (7.6.3)$$

Estimator 1 determines an estimate of the sampled impulse response of the channel and from this estimate, calculates the desired sampled impulse response of the channel plus filter by using some root-finding technique. Estimator 2 then determines the sampled impulse response at the output of the pre-filter. An error signal is then formed based on the mean-square-error between the desired and estimated impulse responses. This error signal is then used to update the tap gains of the filter. Clearly, the convergence time of the system depends on the time taken for Estimator 1 to produce an accurate estimate of the channel response, the time required to locate and equalise roots of $Y(z)$ which lie outside the unit circle in the z -plane and the time necessary to adjust the pre-filter taps to sufficient accuracy. If it is assumed that Y does not itself change during the actual convergence period and a suitable root-finding algorithm is available, the system should intuitively converge to more accurate estimates of the pre-filter tap gains. and overall sampled impulse response. However, apart from requiring two estimators, the major problem with this scheme is the inter-dependence of the two error signals used in Estimator 2 and in the adjustment of the pre-filter taps which, as in the case of the earlier adjustment system, results in rather long convergence times.

7.6.2 Alternative Scheme 2

Let $Y(z)$, $D(z)$ and $F(z)$ be the z -transforms of the sampled impulse response of the channel, ideal linear filter and channel plus ideal filter, respectively. A channel estimator first determines an estimate, Y' , of Y , which is then operated upon to produce a sequence whose z -transform has

all roots inside the unit circle in the z -plane. It is assumed here that $Y'=Y$, so the output from the root-finding and equalisation algorithm will be a good approximation to $F(z)$. From Figure 7.6.3,

$$F(z) = Y(z) D(z) \quad (7.6.4)$$

and so,

$$D(z) = F(z) Y^{-1}(z) \quad (7.6.5)$$

Now, from equation (7.6.2),

$$Y(z) = y_0 + y_1 z^{-1} + y_2 z^{-2} + \dots y_g z^{-g}$$

$$Y(z) = y_0 (1 + \alpha_1 z^{-1}) (1 + \alpha_2 z^{-1}) \dots (1 + \alpha_g z^{-1}) \quad (7.6.6)$$

where the $\{-\alpha_i\}$ are the complex roots of the complex polynomial $Y(z)$. The $\{|\alpha_i|\}$ are not constrained.

Also,

$$F(z) = f_0 + f_1 z^{-1} + f_2 z^{-2} + \dots f_g z^{-g} \quad (7.6.7)$$

$$F(z) = f_0 (1 + \gamma_1 z^{-1}) (1 + \gamma_2 z^{-1}) \dots (1 + \gamma_g z^{-1}) \quad (7.6.8)$$

where the $\{-\gamma_i\}$ are the complex roots of $F(z)$ and by definition, are constrained such that $|\gamma_i| < 1$, $i=1, 2, \dots g$.

From equation (7.6.5),

$$D(z) = \frac{f_0 + f_1 z^{-1} + f_2 z^{-2} + \dots f_g z^{-g}}{y_0 + y_1 z^{-1} + y_2 z^{-2} + \dots y_g z^{-g}} \quad (7.6.9)$$

so, from equation (7.6.6),

$$D(z) = \frac{f_0 + f_1 z^{-1} + f_2 z^{-2} + \dots f_g z^{-g}}{y_0 (1 + \alpha_1 z^{-1}) (1 + \alpha_2 z^{-1}) \dots (1 + \alpha_g z^{-1})} \quad (7.6.10)$$

Equation (7.6.5) shows that the required tap gains for the linear pre-filter can be obtained by passing the sequence given by the vector F through

a linear filter whose tap gains are given by the coefficients of $Y^{-1}(z)$. The problem has therefore reduced to the evaluation of Y^{-1} , which, as will be shown, is one of the major weaknesses of the system.

The factors $(1+\alpha_i z^{-1})$ in the denominator of equation (7.6.10) can be split into two groups such that $|\alpha_i| > 1$ or $|\alpha_i| < 1$. It is assumed for convenience that the exact equality $|\alpha_i| = 1$ does not occur here. When $|\alpha_i| > 1$, the corresponding roots $\{\alpha_i\}$ of $Y(z)$ lie outside the unit circle in the z -plane whilst for $|\alpha_i| < 1$, the roots $\{\alpha_i\}$ lie inside the unit circle. From equations (7.6.6) and (7.6.10),

$$Y^{-1} = \frac{1}{Y_0 (1+\alpha_1 z^{-1}) (1+\alpha_2 z^{-1}) \dots (1+\alpha_g z^{-1})} \quad (7.6.11)$$

and so

$$Y^{-1} = \frac{1}{Y_0} \left(1+\alpha_1 z^{-1}\right)^{-1} \cdot \left(1+\alpha_1 z^{-1}\right)^{-1} \dots \left(1+\alpha_g z^{-1}\right)^{-1} \quad (7.6.12)$$

When $|\alpha_i| < 1$, the factors $(1+\alpha_i z^{-1})^{-1}$ can either be expressed as infinite polynomials in z^{-1} by application of the Binomial expansion or as a cascade of simple one-tap feedback filters of the type shown in Figure 7.6.4. However, when $|\alpha_i| > 1$, the Binomial expansion will of course result in a divergent series whilst the feedback filter will be unstable⁽¹³⁾.

If now the roots $\{\alpha_i\}$ which satisfy $|\alpha_i| < 1$ are designated $\{\alpha_{1,i}\}$ and those which satisfy $|\alpha_i| > 1$ designated $\{\alpha_{2,i}\}$, then from equations (7.6.8) and (7.6.10),

$$D(z) = \frac{(\alpha_{2,1}^* + z^{-1})(\alpha_{2,2}^* + z^{-1}) \dots (\alpha_{2,k}^* + z^{-1})(1+\alpha_{1,1} z^{-1})(1+\alpha_{1,2} z^{-1}) \dots (1+\alpha_{1,g-k} z^{-1})}{Y_0 (1+\alpha_{2,1} z^{-1})(1+\alpha_{2,2} z^{-1}) \dots (1+\alpha_{2,k} z^{-1})(1+\alpha_{1,1} z^{-1})(1+\alpha_{1,2} z^{-1}) \dots (1+\alpha_{1,g-k} z^{-1})} \quad (7.6.13)$$

where it is assumed that k of the g roots of $Y(z)$ lie outside the unit circle in the z -plane. Also, each of the linear factors of the form $(\alpha_{2,i}^* + z^{-1})$ has a root at $z = -\frac{1}{\alpha_{2,i}^*}$, that is, at a position in the z -plane which is the complex conjugate $\alpha_{2,i}$ of the reciprocal of the original root, $(-\alpha_{2,i})$, where $|\alpha_{2,i}| > 1$. This form of linear factor is an alternative to that used in Section 5.6 and is used here since the sum of the squares of the components of $(\alpha_{2,i}^* + z^{-1})$ and that of its associated factor in $Y(z)$, $(1 + \alpha_{2,i} z^{-1})$, are identical. The replacement of a factor such as $(1 + \alpha_{2,i} z^{-1})$ by $(\alpha_{2,i}^* + z^{-1})$ therefore replaces the root $(-\alpha_{2,i})$ with $(-\frac{1}{\alpha_{2,i}^*})$, but does not cause a change in signal level, which is contrary to the case considered in Section 5.6.

Equation (7.6.13) clearly shows the action of the linear filter whose z -transform is $D(z)$; the filter removes all roots of $Y(z)$ which lie outside the unit circle by inserting poles at the locations $z = -\alpha_{2,i}$, $i=1, 2, \dots, k$ and includes extra roots (zeros) at the positions $z = (-\frac{1}{\alpha_{2,i}^*})$, $i=1, 2, \dots, k$ without, however, changing the roots $\{-\alpha_{1,j}\}$, $j=1, 2, \dots, g-k$, which lie inside the unit circle. Moreover, the filter does not cause a change of signal level, nor does it change the number of non-zero components in the sampled impulse response at its output, compared with that of its input. Since the $(g-k)$ linear factors of $Y(z)$ for which $|\alpha_i| < 1$ appear in both the numerator and denominator of equation (7.6.13), it may be simplified to,

$$D(z) = F_2(z) Y_2^{-1}(z) \quad (7.6.14)$$

where,

$$F_2(z) = (\alpha_{2,1}^* + z^{-1})(\alpha_{2,2}^* + z^{-1}) \dots (\alpha_{2,k}^* + z^{-1}) \quad (7.6.15)$$

$$\text{and } Y_2(z) = (1 + \alpha_{2,1} z^{-1})(1 + \alpha_{2,2} z^{-1}) \dots (1 + \alpha_{2,k} z^{-1}) \quad (7.6.16)$$

If $Y(z)$ is now passed through the filter with a z -transform $D(z)$, the output will be,

$$\begin{aligned} F(z) &= Y(z)D(z) \\ &= Y_2(z)Y_1(z)F_2(z)Y_2^{-1}(z) \\ &= Y_1(z)F_2(z) \end{aligned} \quad (7.6.17)$$

where $Y_1(z)$ is taken to represent the factors of $Y(z)$ for which $|\alpha_i| < 1$ and also includes the term y_0 .

If it is assumed that the root-finding algorithm has successfully determined the g roots of $Y(z)$, both $Y_2(z)$ and $F_2(z)$ can be formed and so the problem has now reduced to the evaluation of $Y_2^{-1}(z)$, such that;

$$Y_2(z)Y_2^{-1}(z) = 1 \quad (7.6.18)$$

Since all of the factors of the form $(1+\alpha_{2,i}z^{-1})$ in equation (7.6.16) have roots satisfying $|\alpha_{2,i}| > 1$, the corresponding terms $(1+\alpha_{2,i}z^{-1})^{-1}$ in $Y_2^{-1}(z)$ cannot be determined by simple application of the Binomial Theorem since the resultant infinite series in z^{-1} will be divergent; it follows that simple long division of 1 by $Y_2(z)$ will also result in a divergent series. Consequently, either the individual factors $(1+\alpha_{2,i}z^{-1})^{-1}$ or the complete k^{th} order polynomial $Y_2^{-1}(z)$ must be evaluated by some other method.

From equation (7.6.16),

$$Y_2(z) = \prod_{i=1}^k Y_{2,i}(z) \quad (7.6.19)$$

$$\text{where } Y_{2,i}(z) = (1+\alpha_{2,i}z^{-1}) \quad (7.6.20)$$

and so,

$$Y_2^{-1}(z) = \prod_{i=1}^k Y_{2,i}^{-1}(z) \quad (7.6.21)$$

For each of the factors $Y_{2,i}(z)$, $Y_{2,i}^{-1}(z)$ can be determined as follows.

Let $N_i(z)$ be defined as,

$$N_i(z) = (\alpha_{2,i} + z^{-1}) \quad (7.6.22)$$

where $N_i(z)$ is obtained from $Y_{2,i}(z)$ by reversing the order of its coefficients. The root of $N_i(z)$ is therefore $-\frac{1}{\alpha_{2,i}}$ and will lie inside the unit circle. It follows that,

$$\begin{aligned} N_i^{-1}(z) &= (\alpha_{2,i} + z^{-1})^{-1} \\ &= \alpha_i^{-1} (1 + \alpha_i^{-1} z^{-1})^{-1} \end{aligned} \quad (7.6.23)$$

where for convenience, $\alpha_{2,i} = \alpha_i$ since for all α_i , $|\alpha_i| > 1$ here.

Since $|\alpha_i^{-1}| < 1$, equation (7.6.23) may be expanded to give,

$$\begin{aligned} N_i^{-1}(z) &= \alpha_i^{-1} (1 - \alpha_i^{-1} z^{-1} + \alpha_i^{-2} z^{-2} - \dots) \\ &= \alpha_i^{-1} - \alpha_i^{-2} z^{-1} + \alpha_i^{-3} z^{-2} - \dots \end{aligned} \quad (7.6.24)$$

which is a convergent series in z^{-1} . $N_i^{-1}(z)$ may also be represented by the z -transform of the sampled impulse response of the single tap feedback filter section shown in Figure 7.6.5. Clearly, the filter has a single pole at $z = -\frac{1}{\alpha_i}$ and is therefore unconditionally stable.

The infinite series given in equation (7.6.24) can be approximated by taking the first j terms, viz.,

$$N_i^{-1}(z) \approx \alpha_i^{-1} - \alpha_i^{-2} z^{-1} + \alpha_i^{-3} z^{-2} - \dots - (-\alpha_i)^{-j} z^{-j+1} \quad (7.6.25)$$

Consider the z -transform $C_i(z)$ which is obtained from the approximation of $N_i^{-1}(z)$ by reversing the order of its coefficients,

$$C_i(z) = -(-\alpha_i)^{-j} - (-\alpha_i)^{-j+1} z^{-1} - \dots + \alpha_i^{-1} z^{-j+1} \quad (7.6.26)$$

From equations (7.6.20) and (7.6.26),

$$Y_{2,i}(z) C_i(z) = -(\alpha_i)^{-j} + z^{-j} \quad (7.6.27)$$

Consider now the full term $Y_2(z)$ in equation (7.6.16). From the above analysis, $N(z)$ can be written as,

$$N(z) = Y_{2,k} + Y_{2,k-1}z^{-1} + Y_{2,k-2}z^{-2} + \dots z^{-k} \quad (7.6.28)$$

or alternatively,

$$N(z) = (\alpha_1 + z^{-1})(\alpha_2 + z^{-1}) \dots (\alpha_k + z^{-1}) \quad (7.6.29)$$

which follows from the fact that if the order of the coefficients of two polynomials is reversed, then so too is the order of the coefficients of their product. $N^{-1}(z)$ can now be calculated from the long division of 1 by $N(z)$, or from

$$N^{-1}(z) = N_1^{-1}(z) N_2^{-1}(z) \dots N_k^{-1}(z) \quad (7.6.30)$$

where the expansion of each $N_i^{-1}(z)$ is given by equation (7.6.24). It follows that $N^{-1}(z)$ may be obtained by cascading k filter sections of the form shown in Figure 7.6.5. The infinite power series in z^{-1} representing $N^{-1}(z)$ is then truncated to the first J terms, such that,

$$N^{-1}(z) \approx n_0 + n_1 z^{-1} \dots n_J z^{-J+1} \quad (7.6.31)$$

where the n_i are derived from equations (7.6.25) and (7.6.30). $C(z)$ is then obtained from $N^{-1}(z)$ by reversing the order of its coefficients,

$$C(z) = n_J + n_{J-1}z^{-1} + \dots n_0 z^{-J+1} \quad (7.6.32)$$

It follows from equations (7.6.26) and (7.6.27) that,

$$Y_2(z)C(z) \approx z^{-J} \quad (7.6.33)$$

$$\text{and so, } C(z) \approx Y_2^{-1}(z) z^{-J} \quad (7.6.34)$$

If now $Y_2^{-1}(z)$ in equation (7.6.14), is replaced by $C(z)$,

$$D(z) \approx F_2(z)Y_2^{-1}(z) z^{-J} \quad (7.6.35)$$

so $D(z)$ performs the required filtering process but also introduces a delay of J sample periods.

It should be clear from the preceding analysis that equation (7.6.14) can only be exactly satisfied if $J \rightarrow \infty$, such that the residual terms of the form $-(\alpha_i)^{-j}$ given in equation (7.6.27) reduce to zero. For the practical implementation of the pre-filter, J must obviously be reduced to some finite number; the larger the value of J , the more accurate is the filtering process but the filter will introduce a larger delay. In practice, when the pre-filter whose z -transform is $D(z)$ is implemented as a linear transversal filter with $(\ell+1)$ taps, the value of J will, in fact, be limited to a maximum of ℓ . This follows since, if the sampled impulse response at the input to the filter has $(g+1)$ components, the sampled impulse response at its output must have $(g+\ell+1)$ components. Since it is known that the required output sampled impulse response also has $(g+1)$ components, (from equation 7.6.17), then the maximum delay that can be introduced by the filter is ℓ sample periods. Consequently, a filter having $(\ell+1)$ tap coefficients most closely approximates the ideal filter when,

$$D(z) = F_2(z)Y_2^{-1}(z)z^{-\ell} \quad (7.6.36)$$

$$\text{that is, } D(z) = F_2(z)C(z) \quad (7.6.37)$$

where $C(z)$ is a polynomial in z^{-1} of order ℓ .

It is also interesting to note here that a suitable value for ℓ , and therefore the length of the pre-filter, depends not only on the number of roots, k , of the z -transform of the particular baseband channel which lie outside the unit circle, but also on the distance of those roots from the unit circle in the z -plane, that is, the $\{|\alpha_i|\}$, where $|\alpha_i| > 1$, $i=1,2,\dots,k$. This can be illustrated as follows. If $(-\alpha_i)$ is a root of $Y(z)$ which lies outside the unit circle and if $|\alpha_i|$ is just greater than unity, then

$|\alpha_i^{-1}|$ will be just less than unity. It follows from equations (7.6.22) - (7.6.27), that the accurate equalisation of this root will necessitate a very large number of coefficients such that the residual term, $-(\alpha_i)^{-j}$, in equation (7.6.27), approaches zero. On the other hand, if $|\alpha_i| \gg 1$, then $|\alpha_i^{-1}| \ll 1$ and a relatively small number of terms in $C_i(z)$, equation (7.6.26), will reduce $-(\alpha_i)^{-j}$ to an insignificant value. Consequently, a filter with a large number of taps is required to equalise roots of $Y(z)$ which lie close to the unit circle. However, it can be shown by a simple example that the equalisation of a root of $Y(z)$ which lies just outside the unit circle does not alter the sampled impulse response to any great extent.

$$\text{Let } Y(z) = (1 + \alpha_1 z^{-1})(1 + \alpha_2 z^{-1}) \quad (7.6.38)$$

If, for convenience, purely real roots are considered, where

$$\alpha_1 = 1.05, \alpha_2 = 0.5$$

$$\text{then, } Y(z) = 1 + 1.55z^{-1} + 0.525z^{-2} \quad (7.6.39)$$

The equalised sampled impulse response is,

$$\begin{aligned} F(z) &= (1.05 + z^{-1})(1 + 0.5z^{-1}) \\ F(z) &= 1.05 + 1.525z^{-1} + 0.5z^{-2} \end{aligned} \quad (7.6.40)$$

Comparison of equations (7.6.39) and (7.6.40) shows quite clearly the validity of the above statement.

Based on the above result, several refinements can be made to the basic system. Since, from equation (7.6.37), the tap coefficients of the pre-filter are calculated from a knowledge of the roots of $Y(z)$ which lie outside the unit circle in the z -plane, the root-finding algorithm need only supply these k roots, rather than the full g roots of $Y(z)$. Furthermore, since roots close to but outside the unit circle have little influence on

the system, the root-finding algorithm may be instructed to ignore all roots for which, say, $|\alpha_1| < 1.1$. Based upon a knowledge of the number and magnitudes of the roots ($|\alpha_1| > 1.1$), the process may then allocate a value to $(l+1)$, the number of taps in the filter, such that equalisation may be achieved to sufficient accuracy. Finally, the sampled impulse response of the baseband channel and linear pre-filter can be obtained from $F(z)$, where,

$$F(z) = F_1(z)F_2(z) = Y_1(z)F_2(z) \quad (7.6.41)$$

which is required by the near-maximum likelihood detector and the arrangement of intersymbol interference cancellation.

Clearly, this method involves fairly complex signal-processing techniques, the most time consuming being the determination of the roots of $Y(z)$ and the inverse of $Y_2(z)$. In practice, the system will be based around a software-controlled processing machine, utilising a set of standardised algorithms to calculate both $D(z)$ and $F(z)$.

7.6.3 Alternative Scheme 3

From Figure 7.6.3,

$$F(z) = Y(z)D(z) \quad (7.6.42)$$

where,

$$\begin{aligned} F(z) &= f_0 + f_1 z^{-1} + f_2 z^{-2} + \dots + f_g z^{-g} \\ Y(z) &= y_0 + y_1 z^{-1} + y_2 z^{-2} + \dots + y_g z^{-g} \\ D(z) &= d_0 + d_1 z^{-1} + d_2 z^{-2} + \dots + d_l z^{-l} \end{aligned} \quad (7.6.43)$$

As before, $Y(z)$ is estimated using a conventional channel estimator and, by a suitable root-finding algorithm and root equalisation method,

the required functions, $F(z)$ and $D(z)$ can be determined. In terms of discrete time sequences, equation (7.6.42) can be written as,

$$\{f_i\} = \{y_i\} * \{d_i\} \quad (7.6.44)$$

where $*$ denotes convolution.

It follows that,

$$f_i = \sum_{h=0}^g d_{i-h} y_h, \quad i = 0, 1, \dots, g \quad (7.6.45)$$

and so,

$$f_0 = y_0 d_0, \quad \therefore d_0 = \frac{f_0}{y_0}$$

$$f_1 = y_0 d_1 + y_1 d_0, \quad \therefore d_1 = \frac{1}{y_0} (f_1 - y_1 d_0)$$

⋮

$$f_i = \sum_{h=0}^g d_{i-h} y_h, \quad \therefore d_i = \frac{1}{y_0} \left(f_i - \sum_{h=1}^i d_{i-h} y_h \right) \quad (7.6.46)$$

As it stands, the above method for the determination of the $\{d_i\}$ is not in fact realisable since equations (7.6.43) and (7.6.46) imply that the sequence at the output of the filter with $(\ell+1)$ taps has $(g+1)$ components when the input sequence also has $(g+1)$ components. This, of course, is not possible since the discrete convolution of a sequence having $(g+1)$ components with a sequence having $(\ell+1)$ components must result in a sequence with $(g+\ell+1)$ components. However, this can easily be resolved by forcing the first ℓ components of the output sequence to be zero-valued. Equation (7.6.43) then becomes,

$$\begin{aligned}
 F(z) &= f_0 + f_1 z^{-1} + f_2 z^{-2} + \dots + f_{g+l} z^{-g-l} \\
 Y(z) &= y_0 + y_1 z^{-1} + y_2 z^{-2} + \dots + y_g z^{-g} \\
 D(z) &= d_0 + d_1 z^{-1} + d_2 z^{-2} + \dots + d_l z^{-l}
 \end{aligned} \tag{7.6.47}$$

where the pre-filter has $(l+1)$ tap coefficients and where ideally,

$$f_i = 0, \quad i=0,1, \dots, l-1 \tag{7.6.48}$$

The method to be described can be considered as the time-domain version of Alternative Scheme 2. The function of the pre-filter is still, of course, to achieve a minimum phase version of $Y(z)$ at its output and so $D(z)$ still satisfies equations (7.6.13) and (7.6.14), although this method attempts to determine the taps $\{d_i\}$ in the time domain using a form of equation (7.6.46) rather than explicitly calculating $Y_2^{-1}(z)$ in equation (7.6.14).

From the discussion in Section 7.6.2 and as shown by equation (7.6.14), the tap coefficients of the pre-filter only depend on the roots of $Y(z)$ which lie outside the unit circle in the z -plane. After determining these k roots, the root-finding and root equalisation algorithm forms the polynomials,

$$\begin{aligned}
 Y_2(z) &= (1+\alpha_1 z^{-1})(1+\alpha_2 z^{-1}) \dots (1+\alpha_k z^{-1}), |\alpha_i| > 1, \\
 &= y_{2,0} + y_{2,1} z^{-1} + y_{2,2} z^{-2} + \dots + y_{2,k} z^{-k}
 \end{aligned} \tag{7.6.49}$$

$$\begin{aligned}
 F_2(z) &= (\alpha_1^* + z^{-1})(\alpha_2^* + z^{-1}) \dots (\alpha_k^* + z^{-1}) \\
 &= f_{2,0} + f_{2,1} z^{-1} + f_{2,2} z^{-2} + \dots + f_{2,k} z^{-k}
 \end{aligned} \tag{7.6.50}$$

$$\text{and } F_3(z) = f_{3,0} + f_{3,1} z^{-1} + f_{3,2} z^{-2} + \dots + f_{3,l+k} z^{-l-k} \tag{7.6.51}$$

where,

$$f_{3,i} = \begin{cases} 0, & i=0,1, \dots, l-1 \\ f_{2,i-l}, & i=l, l+1, \dots, l+k \end{cases}$$

Since $Y_2(z)$ has all of its roots outside the unit circle, the derivation of $D(z)$ from $Y_2(z)$ must actually involve $Y_2^R(z)$, which is obtained from $Y_2(z)$ by reversing the order of its coefficients; this is necessary to ensure that $D(z)$ is a convergent series in z^{-1} . Also, since the product of two polynomials whose coefficients have been reversed in order is equal to the polynomial resulting from the product of the two original polynomials but with the order of its coefficients reversed, it follows from equation (7.6.42) that,

$$F_2^R(z) = Y_2^R(z) D^R(z) \quad (7.6.52)$$

where $F_2^R(z)$, $Y_2^R(z)$ and $D^R(z)$ are obtained from $F_2(z)$, $Y_2(z)$ and $D(z)$, respectively, by reversing the order of their coefficients. To satisfy the conditions given in equation (7.6.47), $F_3^R(z)$ is used in place of $F_2^R(z)$ to give,

$$F_3^R(z) = Y_2^R(z) D^R(z) \quad (7.6.53)$$

Consequently, from equation (7.6.46), the coefficients of $D^R(z)$, the $\{d_i^R\}$, are found from,

$$d_i^R = \frac{1}{y_{2,0}^R} \left[f_{3,i}^R - \sum_{h=1}^i y_{2,h}^R d_{i-h}^R \right] \quad (7.6.54)$$

for $i=0,1, \dots, l$

where the $\{d_i^R\}$, $\{y_{2,i}^R\}$ and $\{f_{3,i}^R\}$ are the components of the corresponding reversed sequences.

As an example of the method, consider a baseband channel whose sampled impulse response has the z -transform;

$$\begin{aligned} Y(z) &= (1 + \alpha_1 z^{-1})(1 + \alpha_2 z^{-1}) \\ &= 1 + (\alpha_1 + \alpha_2) z^{-1} + \alpha_1 \alpha_2 z^{-2} \end{aligned} \quad (7.6.55)$$

where, $|\alpha_1| < 1$, $|\alpha_2| > 1$.

So,

$$\left. \begin{aligned} Y_2(z) &= (1 + \alpha_2 z^{-1}) \\ Y_2^R(z) &= (\alpha_2 + z^{-1}) \\ F_2(z) &= (\alpha_2^* + z^{-1}) \\ F_2^R(z) &= (1 + \alpha_2^* z^{-1}) \end{aligned} \right\} \quad (7.6.56)$$

If the filter is assumed to have 10 taps, $(l+1) = 10$ and from equation (7.6.51), $F_3(z)$ will have $(l+k+1) = 11$ coefficients, where,

$$F_3(z) = 0 + 0z^{-1} + 0z^{-2} + \dots + 0z^{-8} + \alpha_2^* z^{-9} + z^{-10} \quad (7.6.57)$$

$$\text{and } F_3^R(z) = 1 + \alpha_2^* z^{-1} + 0z^{-2} + \dots + 0z^{-10} \quad (7.6.58)$$

From equations (7.6.54), (7.6.56) and (7.6.58),

$$\begin{aligned} d_0^r &= \frac{1}{y_{2,0}^r} (f_{3,0}^r - 0) = \frac{1}{\alpha_2} \\ d_1^r &= \frac{1}{y_{2,0}^r} (f_{3,1}^r - y_{2,1}^r d_0^r) = \frac{1}{\alpha_2} \left(\alpha_2^* - \frac{1}{\alpha_2} \right) \\ d_i^r &= (-1)^{i+1} \frac{1}{(\alpha_2)^i} \left(\alpha_2^* - \frac{1}{\alpha_2} \right), \quad i=1, 2, \dots, 9 \\ &\quad i \neq 0 \end{aligned} \quad (7.6.59)$$

Since $|\alpha_2| > 1$, it is clear that the sequence $\{d_i^r\}$ is convergent, as required.

$D(z)$ can now be formed from $D^R(z)$ by reversing its coefficients, hence,

$$D(z) = d_9^r + d_8^r z^{-1} + d_7^r z^{-2} + \dots + d_0^r z^{-9} \quad (7.6.60)$$

The operation of the filter can now be checked by forming the product $Y(z)D(z)$ which should equal,

$$Y(z)D(z) = 0 + 0z^{-1} + 0z^{-2} + \dots + 0z^{-8} + \alpha_2^* z^{-9} + (1 + \alpha_1 \alpha_2^*) z^{-10} + \alpha_1 z^{-11} \quad (7.6.61)$$

Using equations (7.6.55) and (7.6.60)

$$Y(z)D(z) = \frac{1}{(\alpha_2)^9} \left(\alpha_2^* - \frac{1}{\alpha_2} \right) + \frac{\alpha_1}{(\alpha_2)^9} \left(\alpha_2^* - \frac{1}{\alpha_2} \right) z^{-1} + 0z^{-2} + \dots + 0z^{-8} + \alpha_2^* z^{-9} + (1 + \alpha_1 \alpha_2^*) z^{-10} + \alpha_1 z^{-11} \quad (7.6.62)$$

Clearly, as $|\alpha_2| > 1$, $(\alpha_2)^{-9} \ll 1+j$, and so the output from the filter adjusted by the above method approximates very closely to $F(z)$. The filter also introduces a delay of ℓ sample intervals, which is in complete agreement with equation (7.6.36). Again, as ℓ is increased, the filter will approximate more closely with the ideal filter, but the delay will also increase in direct proportion. For telephone circuit 3 introduced in Chapter 5, a 40-tap pre-filter has proved to be a reasonable compromise between the delay introduced and the ability of the filter to reduce its early outputs to suitably low values.

Compared with Alternative Scheme 2, the method developed above offers a substantial reduction in the signal processing requirement due to the repeated application of equation (7.6.54), which itself is in a form suitable for application to a software-controlled processor. Assuming that a root-finding algorithm has calculated the g roots of $Y(z)$, then the following number of operations will be required to calculate the tap coefficients of the filter and the coefficients of the sampled impulse response of the channel and filter, where it is assumed that the sampled impulse response of the channel has $(g+1)$ components, the filter has $(\ell+1)$ taps and k of the g roots of $Y(z)$ lie outside the unit circle;

a) Determination of $Y_2^R(z)$; $\frac{1}{2}(k^2+k)$ complex multiplications

$\frac{1}{2}(k^2+k)$ complex additions

- b) Determination of $F_2^R(z)$; k sign changes. This follows since the coefficients of $F_2^R(z)$ are the conjugate reverses of the coefficients of $Y_2^R(z)$.
- c) Calculation of the $\{d_1\}$; $\frac{1}{2}(k^2+3k+2)+(l-k-1)(k+1)$ complex multiplications
 $\frac{1}{2}(k^2+k) + (l-k-1)k$ complex subtractions
 1 complex division
- d) Calculation of overall $\frac{1}{2}(g^2+g)$ complex multiplications
 sampled impulse response ; $\frac{1}{2}(g^2+g)$ complex additions

The processor may also calculate the partial products, $\{s_i f_j\}$, $i=0,1,\dots,m$, $j=0,1,\dots,g$, where the $\{s_i\}$ are the m possible values of the data symbols and the $\{f_j\}$ are the components of the sampled impulse response of the channel and pre-filter. These partial products can then be stored and accessed by the detector as required.

A further advantage of this method over the conventional gradient adjustment system is that once the estimator which estimates the channel sampled impulse response has converged to an accurate estimate of Y , the time taken to determine the tap coefficients of the pre-filter is dependent only on the speed of the processing device. Also, the calculation of the tap coefficients can effectively be performed "off-line", or in parallel with the main receiver processor. For a time varying channel, the variations in the channel sampled impulse response can be monitored and new values of filter tap coefficients and the channel plus filter sampled impulse response calculated when the variations exceed pre-determined limits. Further, if Y changes only slowly with time, the speed requirement of the processor performing the adjustment system calculations can be reduced by spreading the calculation period over more than one sample interval.

7.6.4 Determination of the Complex Roots of Complex Polynomials

Any method which makes full use of the prior knowledge that the linear pre-filter should ideally replace all roots of $Y(z)$ which lie outside the unit circle in the z -plane by the complex conjugates of their reciprocals, must necessarily involve some root-finding algorithm. In preparation for use in an alternative adjustment system, several root-finding algorithms⁽¹⁸⁻²⁴⁾ were investigated and tested over telephone channels 1-3; not surprisingly, telephone channel 3 proved to be the most difficult channel on which to operate due to its relatively long impulse response and the amount of phase distortion introduced.

Although several published algorithms appear, at least initially, to be worthy of consideration for the current application, their very high levels of complexity would unfortunately require the use of quite sophisticated and powerful processing devices. However, with the cost of such devices falling, the algorithms developed by Grant and Hitchens⁽²²⁾ or the Lehmer-Schur method⁽¹⁸⁾ may well become cost-effective in the near future.

In terms of computational effort, an iterative system based on Newton's method^(23,24) is probably the most cost-effective although convergence to the root positions is relatively slow. Appendix 9 gives details of an extended version of Newton's method and of a procedure, PROCCOMPLEXROOTS, used to evaluate its performance when operating on the sampled impulse response of telephone channel 3; Program 7 in Appendix 10 is a listing of the procedure.

Figure 7.6.6 shows the positions of the roots for the combination of telephone channel 3 and filter set 1 which represents the worst-case baseband channel considered here; for clarity, only the 9 roots which lie outside the unit circle are illustrated. In fact, the omission of the

roots of $Y(z)$ which lie inside the unit circle points to a fairly straightforward modification to the root-finding algorithm. Since the adjustment systems described earlier only require details of the roots which lie outside the unit circle, the root-finding algorithm can be instructed to ignore all roots with an absolute value of less than unity. For a channel having relatively few roots outside the unit circle compared with the length of its sampled impulse response, a considerable saving in computational time and effort can be achieved; Appendix 9 considers the modification in more detail.

On average, 16 iterations of the root-finding algorithm were required for each of the 9 roots of the telephone circuit/filter combination, the sampled impulse of which was represented by polynomial in z^{-1} with 27 coefficients. From the analysis given in Appendix 9, approximately 288 real divisions, 40,212 real multiplications and 33,300 additions/subtractions will therefore be required. If the computational time and effort is considered to be proportional to the number of real multiplications and divisions performed by the adjustment system, then the combination of,

- a) The root-finding algorithm
- b) The determination of the equalised roots
- c) Forming the polynomials $Y_2^R(z)$, $F_2^R(z)$
- d) The determination of the pre-filter tap-gains using Alternative Scheme 3, with a 40-tap filter

will require approximately 41,000 operations. Allowing $1\mu s$ per operation, which should be easily achieved with the present generation of high-speed processing devices, the adaptive adjustment system will take approximately 41 ms to produce the required information. From the results of Section 7.5, this represents a useful advantage when compared with the steepest-descent

adjustment system which required about 1 second to converge.

7.6.5 Alternative Scheme 4

This method, proposed by Clark and Hau⁽²⁵⁾, is similar to Alternative Schemes 2 and 3 in that it only requires an estimate of the sampled impulse response of the channel. As before, the scheme reduces to the evaluation of the k roots of $Y(z)$ which lie outside the unit circle in the z -plane and the determination of the pre-filter tap gains such that the overall sampled impulse response is minimum phase.

As in equation (7.6.17), let,

$$Y(z) = Y_1(z)Y_2(z) \quad (7.6.63)$$

where now,

$$Y_1(z) = \eta(1+\alpha_1 z^{-1})(1+\alpha_2 z^{-1}) \dots (1+\alpha_{g-k} z^{-1}) \quad (7.6.64)$$

and
$$Y_2(z) = (1+\gamma_1 z^{-1})(1+\gamma_2 z^{-1}) \dots (1+\gamma_k z^{-1})$$

$$= z^{-k}(1+\beta_1 z)(1+\beta_2 z) \dots (1+\beta_k z) \quad (7.6.65)$$

It is assumed here that the vector Y has $(g+1)$ components and that no root of $Y(z)$ lies exactly on the unit circle. Also, the α_i , $i=1,2, \dots g-k$ in equation (7.6.64) all satisfy $|\alpha_i| < 1$, whilst the γ_i , $i = 1,2, \dots k$ satisfy $|\gamma_i| > 1$. In equation (7.6.65), $Y_2(z)$ has been re-written as a polynomial in z by including k poles at the origin of the z -plane. The β_i , $i=1,2,\dots,k$ in equation (7.6.65) all satisfy $|\beta_i| < 1$ and so all roots of $Y_2(z)$ lie outside the unit circle in the z -plane.

When the $(\ell+1)$ -tap pre-filter is correctly adjusted, the z -transform of the sampled impulse response of the filter is, by similarity with equation (7.6.14),

$$D(z) = z^{-\ell} Y_2^{-1} Y_3(z) \quad (7.6.66)$$

$$\text{where } Y_3(z) = (1+\beta_1^* z^{-1})(1+\beta_2^* z^{-1}) \dots (1+\beta_k^* z^{-1}) \quad (7.6.67)$$

$$\text{and so } F(z) \approx Y(z)D(z) = z^{-\ell} Y_1(z) Y_3(z) \quad (7.6.68)$$

where the approximation gets better as $\ell \rightarrow \infty$. As before, all the roots of $F(z)$ will now lie inside the unit circle and so the overall sampled impulse response will be minimum phase.

The basic operation of the system parallels closely that of Adjustment Scheme 3, but with a built-in root-finding technique to find the roots of $Y(z)$ which lie outside the unit circle. The system first forms a simple one-tap feedback filter with the z-transform,

$$A_i(z) = (1+\lambda_i z)^{-1} \quad (7.6.69)$$

for $i = 0, 1, \dots, n$, using an iterative process to update the value of λ_i such that as i increases, $\lambda_i \rightarrow \beta_i$. When $i=n$, the z-transform of the filter is,

$$A_n(z) \approx (1+\beta_1 z)^{-1} \quad (7.6.70)$$

and so the first root of $Y(z)$ which lies outside the unit circle has been found. The system then forms a filter $C_1(z)$, where,

$$\begin{aligned} C_1(z) &= (1+\lambda_n z)^{-1} (1+\lambda_n^* z^{-1}) \\ &\approx (1+\beta_1 z)^{-1} (1+\beta_1^* z^{-1}) \end{aligned} \quad (7.6.71)$$

In practice, equation (7.6.69) can be implemented since the system will operate on stored values of $Y(z)$ rather than in real time. The whole process is then repeated for all $\beta_h, h=1, 2, \dots, k$ to give a total of k filters of the form $C_h(z)$. These k filters are then cascaded, along with a delay of $(\ell-k)$ sample intervals to make the system realisable, to produce the filter whose z-transform approximates to $D(z)$ in equation (7.6.66). The problem has therefore reduced to the determination of an iterative process to adjust the $\{\lambda_n\}$ towards the $\{\beta_i\}$.

If β_1 is assumed to be known, then

$$U_1(z) = \frac{Y(z)}{(1+\beta_1 z)} \quad (7.6.72)$$

$$\text{where } U_1(z) = u_0 z^{-1} + u_1 z^{-2} + \dots + u_{g-1} z^{-g} \quad (7.6.73)$$

and is a deflated form of $Y(z)$. From equation (7.6.69),

$$Y(z)A_i(z) = (1+\beta_1 z)(1+\lambda_i z)^{-1}U_1(z) \quad (7.6.74)$$

If the output from the filter $A_i(z)$, when fed with $Y(z)$, is given as,

$$Y(z)A_i(z) = e_{i,0} + e_{i,1}z^{-1} + e_{i,2}z^{-2} + \dots + e_{i,g}z^{-g} \quad (7.6.75)$$

then,

$$e_{i,j} = u_{j-1} + (\beta_1 - \lambda_i) \sum_{p=j}^{g-1} u_p (-\lambda_i)^{p-j} \quad (7.6.76)$$

for $j=0,1,2 \dots g$, where it is assumed that $u_j=0$, for all $j < 0$.

When λ_i approaches the root β_1 ,

$$e_{i,j} \approx u_{j-1}, \quad j=1,2, \dots g \quad (7.6.77)$$

and so, from equation (7.6.76),

$$e_{i,0} \approx (\beta_i - \lambda_i) \varepsilon_i \approx 0 \quad (7.6.78)$$

where,

$$\varepsilon_i = \sum_{j=1}^g e_{i,j} (-\lambda_i)^{j-1} \quad (7.6.79)$$

From equation (7.6.78)

$$\beta_i \approx \lambda_i + \frac{e_{i,0}}{\varepsilon_i} \quad (7.6.80)$$

which implies that λ_{i+1} is a better estimate of β_1 than is λ_i , where

$$\lambda_{i+1} = \lambda_i + \frac{e_{i,0}}{\varepsilon_i} \quad (7.6.81)$$

although $e_{i,0}$ cannot be calculated directly from equation (7.6.76) since β_1 is not, of course known. However, $e_{i,0}$ and ε_i can be determined from equation (7.6.75) by passing $Y(z)$ through the filter $A_i(z)$. In fact, to make the system realisable, $Y(z)$ is applied in reverse order through

a filter of the form shown in Figure 7.6.4 where α_i is replaced with λ_i ; this process introduces the delay of g sample intervals required to make the filter $A_i(z)$ realisable. The output sequence from the modified filter will therefore be $\{e_{i,j}\}$, $j=g, g-1, g-2 \dots 0$. It is also important to note that the filter will be unconditionally stable since $|\beta_i| < 1$, $i=1, 2, \dots k$.

The system now determines $e_{i,0}$, ε_i and hence λ_{i+1} from equation (7.6.81) and the process is repeated until $|\lambda_{i+1} - \lambda_i| < \text{TOL}$, where TOL is some small positive real number. To ensure that the algorithm does converge to a root of $Y(z)$, safety checks can be included within the structure. For example, if the number of iterations of the root-finding algorithm exceeds a predetermined value, or if $|\lambda_i| > 1$, the process is terminated and restarted with a different value of λ_0 .

After n iterations, and assuming convergence has occurred, $\lambda_n \approx \beta_1$, so $(\beta_1 - \lambda_n) \approx 0$. From equations (7.6.73) and (7.6.74), the output sequence from the filter now has the z -transform,

$$\begin{aligned} Y(z)A_n(z) &\approx e_{n,0} + e_{n,1}z^{-1} + \dots e_{n,g}z^{-g} \\ &\approx u_0z^{-1} + u_1z^{-2} + \dots u_{g-1}z^{-g} \end{aligned} \quad (7.6.82)$$

since $e_{n,0} \approx 0$.

Equation (7.6.82) shows that the factor $(1+\beta_1z)$ has been effectively removed from $Y(z)$ to give $U_1(z)$. The sequence $U_1(z)$ is now passed through the filter shown in Figure 7.6.7, whose z -transform is given by,

$$B_n(z) = (1 + \lambda_n^* z^{-1}) \quad (7.6.83)$$

The output sequence from this filter is,

$$U_1(z)B_n(z) = f_{1,0} + f_{1,1}z^{-1} + \dots f_{1,g+1}z^{-g-1} \quad (7.6.84)$$

$$= Y(z)A_n(z)B_n(z) \quad (7.6.85)$$

where $f_{1,n} \approx 0$.

The filter $B_n(z)$ therefore introduces the factor $(1+\lambda_n^* z^{-1}) \approx (1+\beta_1^* z^{-1})$ into $U_1(z)$ to give the sequence $U_1(z)B_n(z)$ where,

$$zU_1(z)B_n(z) \approx Y(z) \frac{(1+\beta_1^* z^{-1})}{(1+\beta_1 z)} \quad (7.6.86)$$

The overall effect of the two filters, $A_n(z)$ and $B_n(z)$ is, therefore,

$$\begin{aligned} C_1(z) &= A_n(z)B_n(z) \\ &\approx (1+\beta_1 z)^{-1}(1+\beta_1^* z^{-1}) \end{aligned} \quad (7.6.87)$$

From equations (7.6.84), (7.6.85) and (7.6.87)

$$zY(z)C_1(z) \approx f_{1,1} + f_{1,2}z^{-1} + \dots f_{1,g+1}z^{-g} \quad (7.6.88)$$

where $F_1(z) = zY(z)C_1(z)$ is the estimate of the sampled impulse response of the filter and channel when the filter has equalised the root $-1/\beta_1$ of $Y(z)$.

The system now proceeds to adjust the tap gains of the $(\ell+1)$ -tap linear filter as follows. Initially, the tap gains of the filter, $\{d_{0,i}\}$, $i=0,1,2 \dots \ell$ are all set to zero except the last tap, which is set to unity, so,

$$D_0(z) = z^{-\ell} \quad (7.6.89)$$

When the root-finding algorithm has converged such that $\lambda_n \approx \beta_1$, the sequence $\{d_{0,i}\}$, $i=0,1, \dots \ell$ is fed through the filter $B_n(z)$ to give an output sequence whose z -transform is $D_0(z)B_n(z)$ which will have $(\ell+2)$ coefficients. The sequence is then fed, in reverse order, through the filter $A_n(z)$, to give the sequence with a z -transform,

$$D_0(z)A_n(z)B_n(z) = D_0(z)C_1(z) \quad (7.6.90)$$

where the first $(l+1)$ components from the output of $A_n(z)$ are the coefficients of $z^{-l-1}, z^{-l}, \dots, z^{-1}$ in equation (7.6.90). When reversed in order, these coefficients are the tap gains of the linear filter which will replace the root $-1/\beta_1$ of $Y(z)$ with $-1/\beta_1^*$, when $Y(z)$ is passed through it.

The whole process, including the adjustment of λ_1 is now repeated in order to locate β_2 , but using $F_1(z)$ in place of $D(z)$ and $D_1(z)$ in place of $D_0(z)$. When all k roots of $Y(z)$ have been located, which is assumed to happen when λ_1 does not converge to a root β_j for any of the (set-number) of initial values, λ_0 , the z -transform of the adaptive filter is,

$$D_k(z) \approx D(z)$$

$$\text{and } F_k(z) \approx Y(z)D_k(z) \quad (7.6.91)$$

where the delay introduced by the filter is, for convenience, ignored.

In an interesting discussion on this method, Clark and Hau⁽²⁵⁾ note that when one or more roots of $Y(z)$ lie just outside the unit circle, a very large value of $(l+1)$ will be required to achieve sufficient accuracy in the adjustment system. Furthermore, they observe that, for the case of a near maximum likelihood detector, very little advantage in tolerance to additive noise is achieved by the processing of such roots. It follows that instead of processing all roots $\{\alpha_i\}$ where $|\alpha_i| > 1$, a reduction in both the processing time and the length of the filter will be possible if roots for which, say, $|\alpha_i| < 1.1$ are ignored. In fact, these same conclusions were reached in the development of Alternative Scheme 2, Section 7.6.2 and apply equally to Alternative Scheme 3, Section 7.6.3, and so useful savings in computational time and effort can be achieved when using this refinement with the more severe telephone circuits.

It should be clear from the above discussion that the Clark-Hau method for the determination of the pre-filter tap gains is very similar in principle to Alternative Scheme 3, although the former method gives a more automated solution compared with equation (7.6.54). Nevertheless, although not explored further here, it is intuitively felt that equation (7.6.54) could be manipulated to give the iterative solution discussed above. The major advantage of the Clark-Hau method is the built-in root-finding algorithm and the inclusion of this process in the overall adaptive scheme. Clearly, yet another alternative system could use the root-finder discussed in reference 25 to determine $Y_2^R(z)$ and $F_3^R(z)$ in equation (7.6.53), followed by the use of equation (7.6.54) to determine the pre-filter tap gains. However, it is apparent that the resulting system will be more expensive in terms of computational effort compared with the pure Clark-Hau method. This follows since the latter method, by using an iterative approach, effectively reduces the overall number of operations, in particular, complex multiplications.

Each iteration of the process involves approximately $(\ell+g+1)+(k-1)(2\ell+g+2)+2gkn$ complex multiplications where,

$(\ell+1)$ = number of taps in the pre-filter

$(g+1)$ = length of sampled impulse response of the channel

k = number of roots of $Y(z)$ outside the unit circle

n = average number of iterations of the root-finding algorithm,
per root.

Also, nk complex divisions will be required. If the divisions are included with the number of multiplications, then for $(g+1)=27$, $(\ell+1)=40$, $k=9$, $n=16$, the total number of (real) operations required to implement the system will

be approximately 34,200 which compares favourably with the 41,000 operations for Alternative Scheme 3. Furthermore, tests have shown that for telephone channels 1-3, the root finding algorithm will generally require fewer than 16 iterations to converge, which will reduce the number of operations even further.

In practice, the filtering functions and up-dating of the λ_i will most certainly be performed within a processor programmed to execute the necessary algorithms rather than in hardware filters. Although Chapter 9 considers implementation in more detail, it is informative to note here that most processes performed in discrete-time can be reduced to combinations of standard methods such as multiply, multiply-accumulate, addition/subtraction and comparisons. For example, the $\{e_{i,j}\}$ can be obtained very easily using a multiplier, subtractor and a single word memory. A further example is the evaluation of ϵ_i in equation (7.6.79) or indeed any polynomial of the form;

$$f(x) = a_0 \pm a_1 x^{\pm 1} \pm a_2 x^{\pm 2} \pm \dots \pm a_n x^{\pm n} \quad (7.6.92)$$

One method to evaluate $f(x)$ is to first calculate and store the $\{x^{\pm i}\}$, $i=0,1, \dots, n$ followed by the evaluation of the partial products $\{\pm a_i x^{\pm i}\}$ and finally sum all such partial products. Apart from requiring storage space, this method can introduce large time overheads due to the isolated nature of the various processing steps and because of the time taken to transfer data to and from the memory. Figure 7.6.8 shows an alternative method of implementation which eliminates the memory requirement and speeds-up the process by careful "pipelining" of operations; it does however involve two multiply-accumulate devices. A second alternative is

to split the polynomial into basic signal processing blocks,

$$\begin{aligned}
 f(x) &= a_0 \pm (a_1 x^{\pm 1} \pm a_2 x^{\pm 2} \pm \dots a_n x^{\pm n}) \\
 &= a_0 \pm x^{\pm 1} (a_1 \pm a_2 x^{\pm 1} \pm \dots a_n x^{\pm n-1}) \\
 &= a_0 \pm x^{\pm 1} (a_1 \pm x^{\pm 1} (a_2 \pm \dots a_n \pm x^{\pm n-2})) \\
 &= a_0 \pm x^{\pm 1} (a_1 \pm x^{\pm 1} (a_2 \pm x^{\pm 1} (\dots x^{\pm 1} (a_{n-1} \pm x^{\pm 1} (a_n))))))
 \end{aligned}
 \tag{7.6.93}$$

This can now be implemented using the simple structure as shown in Figure 7.6.9. Clearly, as the $\{e_{i,j}\}$ are actually produced in reverse order, a second processing device may start the calculation of ϵ_i as soon as $e_{i,g}$ is available, so further reducing both storage requirements and the time spent transferring data to and from the memory.

The main drawback of both the Clark-Hau and the extended Newton-Raphson methods of root location is the requirement of (complex) divisions at every iteration of the algorithms. In practice, the divisions can either be performed by a successive approximation method or by addressing a suitably programmed memory configured as a look-up table. The first method reduces the memory requirement of the system in the following way, where it is required to calculate $\frac{e_{i,0}}{\epsilon_i}$.

$$\frac{e_{i,0}}{\epsilon_i} = y, \therefore \epsilon_i \times y = e_{i,0}
 \tag{7.6.94}$$

if $\epsilon_i \times y > e_{i,0}$, y is reduced in value, if $\epsilon_i \times y < e_{i,0}$, y is increased. This process continues until $\epsilon_i \times y = e_{i,0}$ to the accuracy of the processor being used. The process therefore reduces to repeated multiplications, addition/subtractions and comparisons.

In conclusion, the Clark-Hau method or Alternative Scheme 3 and some suitable root-finding algorithm both represent extremely useful alternatives to the conventional jointly-adaptive system considered in Section 7.4. The sensible selection of a high-speed signal processing device and careful programming which reduces data transfer instructions to a minimum, whilst permitting the use of "pipelining" either by using multiple processors or a processor/multiplier-accumulator structure, will result in a fast and fairly flexible adjustment system. From the discussion in Chapter 6, it follows that these new methods may also be used to adaptively adjust the linear filter and the feedback equaliser in a conventional non-linear equaliser.

7.7 References

1. Magee, F.R. and Proakis, J.G., "Adaptive Maximum-Likelihood Sequence Estimation for Digital Signalling in the Presence of Intersymbol Interference", IEEE Trans. on Info. Theory, Vol. IT-19, pp 120-124, Jan.1973.
2. Proakis, J.G., "Digital Communications", McGraw-Hill Inc., 1983.
3. Harvey, J.D., "Synchronisation of Synchronous Modems", SERC Report No. GR/A/1200.7, Dec.1980.
4. McVerry, F., "High Speed Data Transmission over H.F. Radio Links", Ph.D.Thesis, Loughborough University of Technology, 1982.
5. MacKechnie, L.K., "Receivers for Channels with Intersymbol Interference", IEEE Int. Symp. Info. Theory, California, Jan/Feb. 1972.
6. Clark, A.P., Kwong, C.P. and McVerry, F., "Estimation of the Sampled Impulse Response of a Channel", Signal Processing, pp 39-53, Vol.2, No.1, Jan. 1980.
7. Justesen, J., "Fast Calculation of Impulse Responses Without Multiplications", Electronics Letters, Vol.17, No.19, pp 700-701, Sept. 1981.
8. Clark, A.P. and McVerry, F., "Channel Estimation for an HF Radio Link", IEE Proc. pt.F, Vol.128, No.1, pp 33-42, Feb. 1981.

9. Speidel, J., "An Automatic Decision Feedback Equaliser with Variable Optimum Step Size", IEEE 1981, pp 653-657, Source Unknown, IEEE Ref. No; CH1635-2/81/0000-0653 500.75.
10. Ingram, D.G.W. and Takawira, F., "Estimation of Minimum-Phase Impulse Response", Electronics Letters, Vol.19, No.22, pp 946-947, Oct. 1983.
11. Bucklew, J.A., "A Maximum Likelihood Estimator of Channel Impulse Response", IEEE Trans. on Comms., Vol. COM-31, No.2, pp 277-280, Feb. 1983.
12. Lee, T.S. and Cunningham, D.R., "Kalman Filter Equalisation for QPSK Communications", IEEE Trans. on Comms., pp 361-364, March 1976.
13. Clark, A.P., "Advanced Data Transmission Systems", Pentech Press, 1977.
14. Falconer, D.D. and Magee, F.R., "Adaptive Channel Memory Truncation For Maximum-Likelihood Sequence Estimation", Bell Syst. Tech. J., Vol.52, pp 1541-1562, 1973.
15. Messerschmitt, D.G., "Design of a Finite Impulse Response for the Viterbi Algorithm and Decision-Feedback Equaliser", IEEE Int. Conf. Comms., Minneapolis, p 370-1, June 1974.
16. Desblache, A.E., "Optimum Short Desired Impulse Response for Maximum-Likelihood Sequence Estimation", IEEE Trans. Comms. , Vol. COM-25, pp 735-738, 1977.

17. Beare, C.T., "The Choice of the Desired Impulse Response in Combined Linear-Viterbi Algorithm Equalisers", IEEE Trans. Comms., Vol. COM-26, pp 1301-1307, 1978.
18. Lehmer, D.H., "A Machine Method for Solving Polynomial Equations", Ass.Comp. Mach. J., Vol.8, pp 151-162, 1961.
19. Adams, D.A., "A Stopping Criterion for Polynomial Root Finding", Comm. of Ass. Comp. Mach., Vol.10, No.10, pp 655-658, Oct. 1967.
20. Peters, G. and Wilkinson, J.H., "Practical Problems Arising in the Solution of Polynomial Equations", J. Inst. Maths. Applics., Vol.8, pp 16-35, 1971.
21. Grant, J.A. and Hitchins, G.D., "An Always Convergent Minimisation Technique for the Solution of Polynomial Equations", J. Inst. Maths. Applics., Vol.8, pp 122-129, 1971.
22. Grant, J.A. and Hitchins, G.D., "Two Algorithms for the Solution of Polynomial Equations to Limiting Machine Precision", The Computer Journal, Vol. pp 258-264.
23. Ralston, A., "A First Course in Numerical Analysis," McGraw-Hill Inc., 1965.
24. Wilkinson, J.H., "The Algebraic Eigenvalue Problem", Oxford University Press, 1965.

25. Clark, A.P. and Hau, S.F., "Adaptive Adjustment of Receiver for Distorted Digital Signals", IEE Proc. F., Vol.131, No.5, pp 526-536, August 1984.
26. Clark, A.P., Zhu, Z.C. and Joshi, J.K., "Fast Start-up Channel Estimation", IEE Proc. F., Vol. 131, No.4, pp 375-382, July 1984.

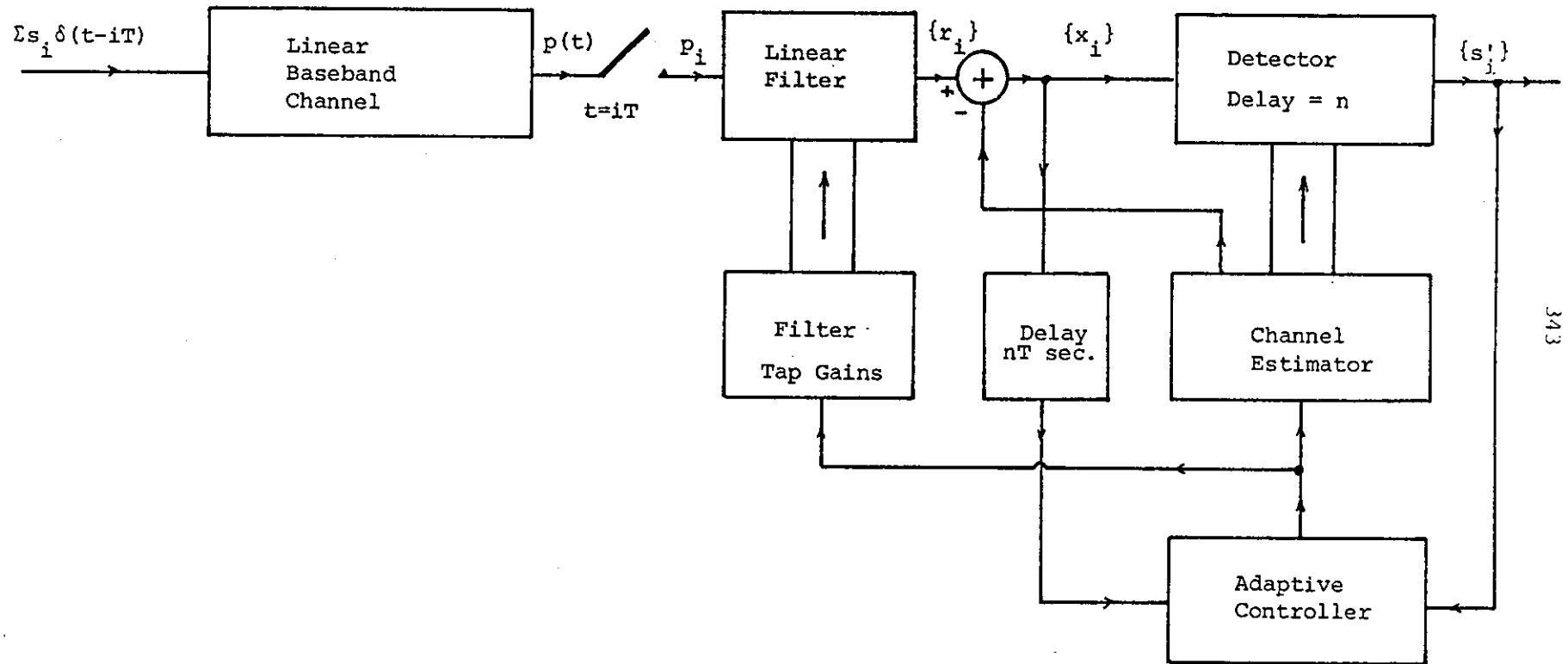
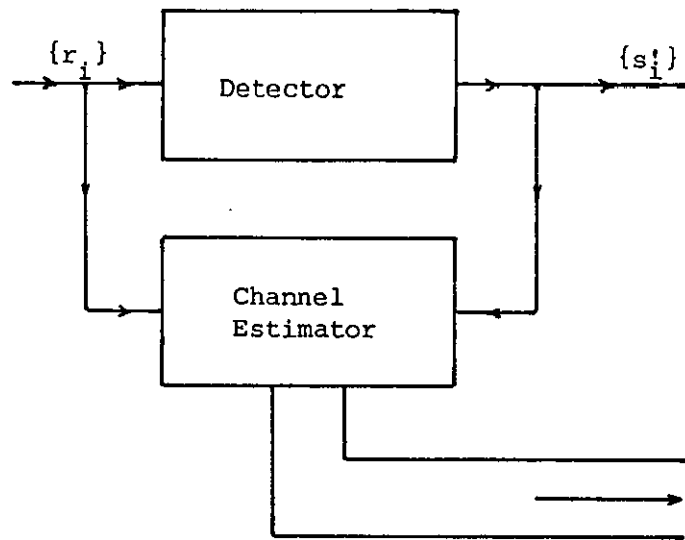


Figure 7.1: General arrangement of an adaptive detection system.



Estimate of
Sampled Impulse Response

Figure 7.2.1: General arrangement for channel estimation

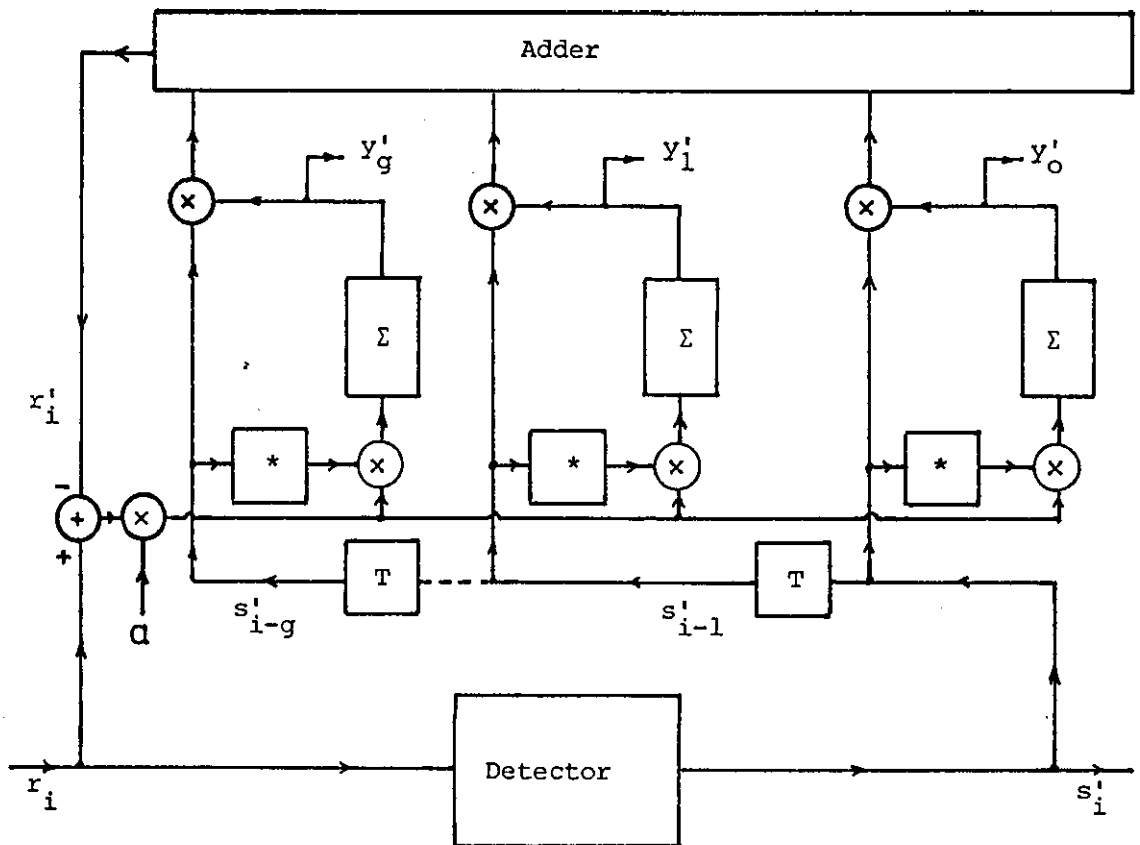


Figure 7.2.2: Complex linear feedforward channel estimator

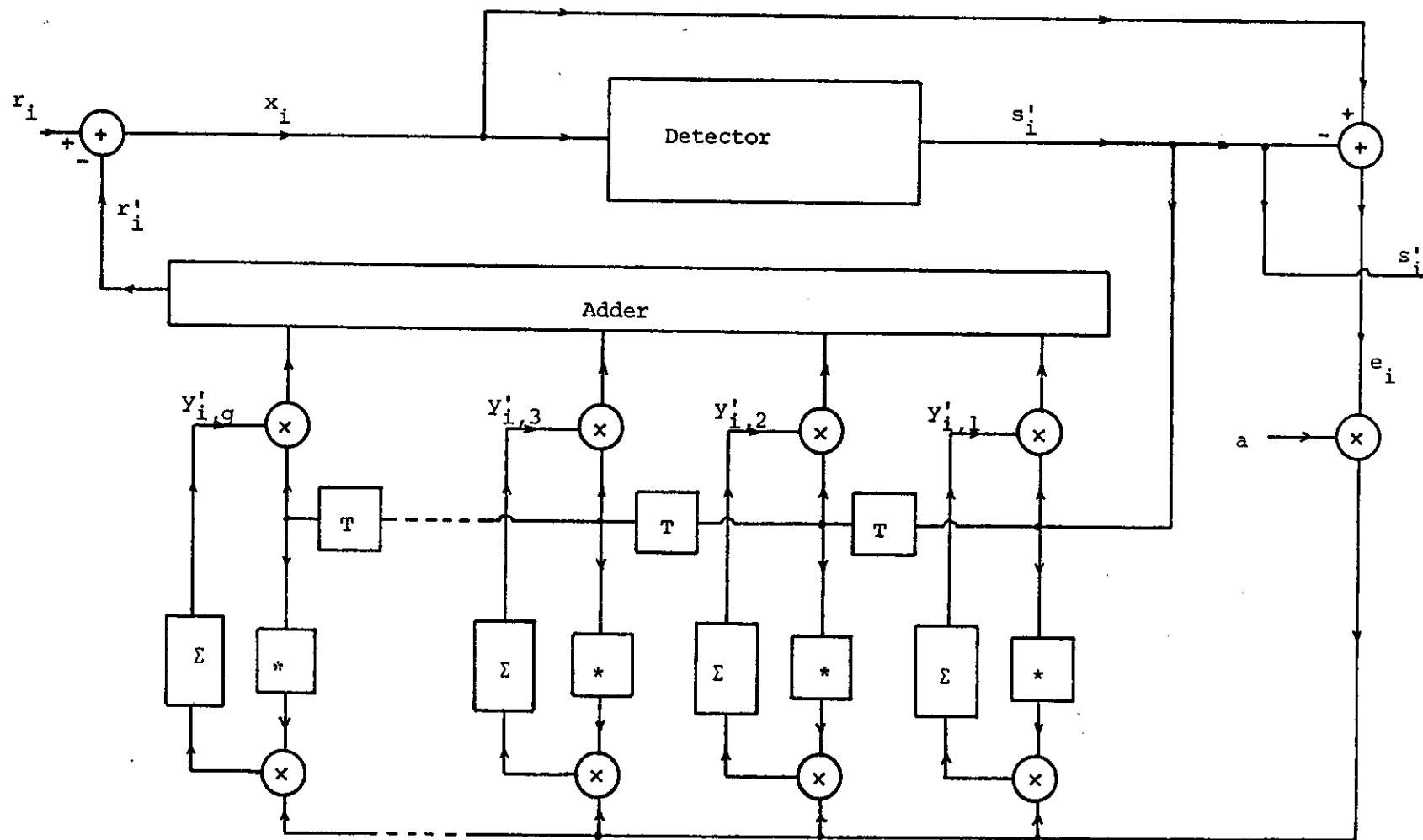


Figure 7.3.1: Adaptive decision-feedback equaliser

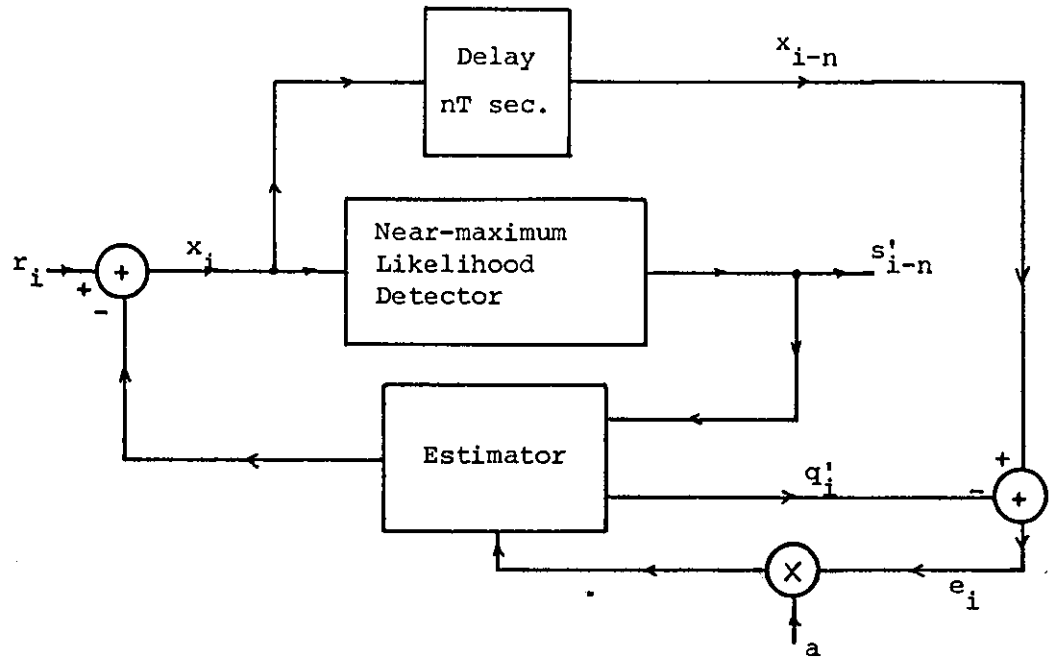


Figure 7.3.2: General arrangement of estimation and cancellation with near-maximum likelihood detection.

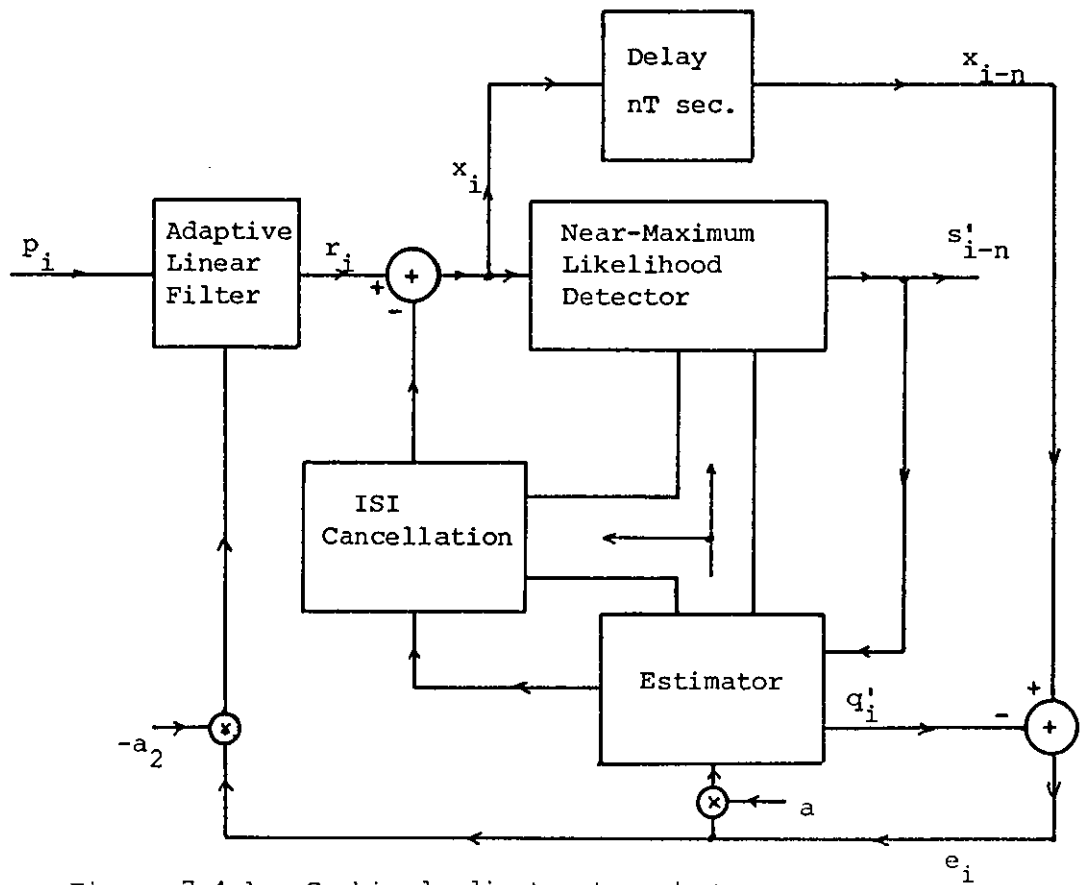


Figure 7.4.1: Combined adjustment system

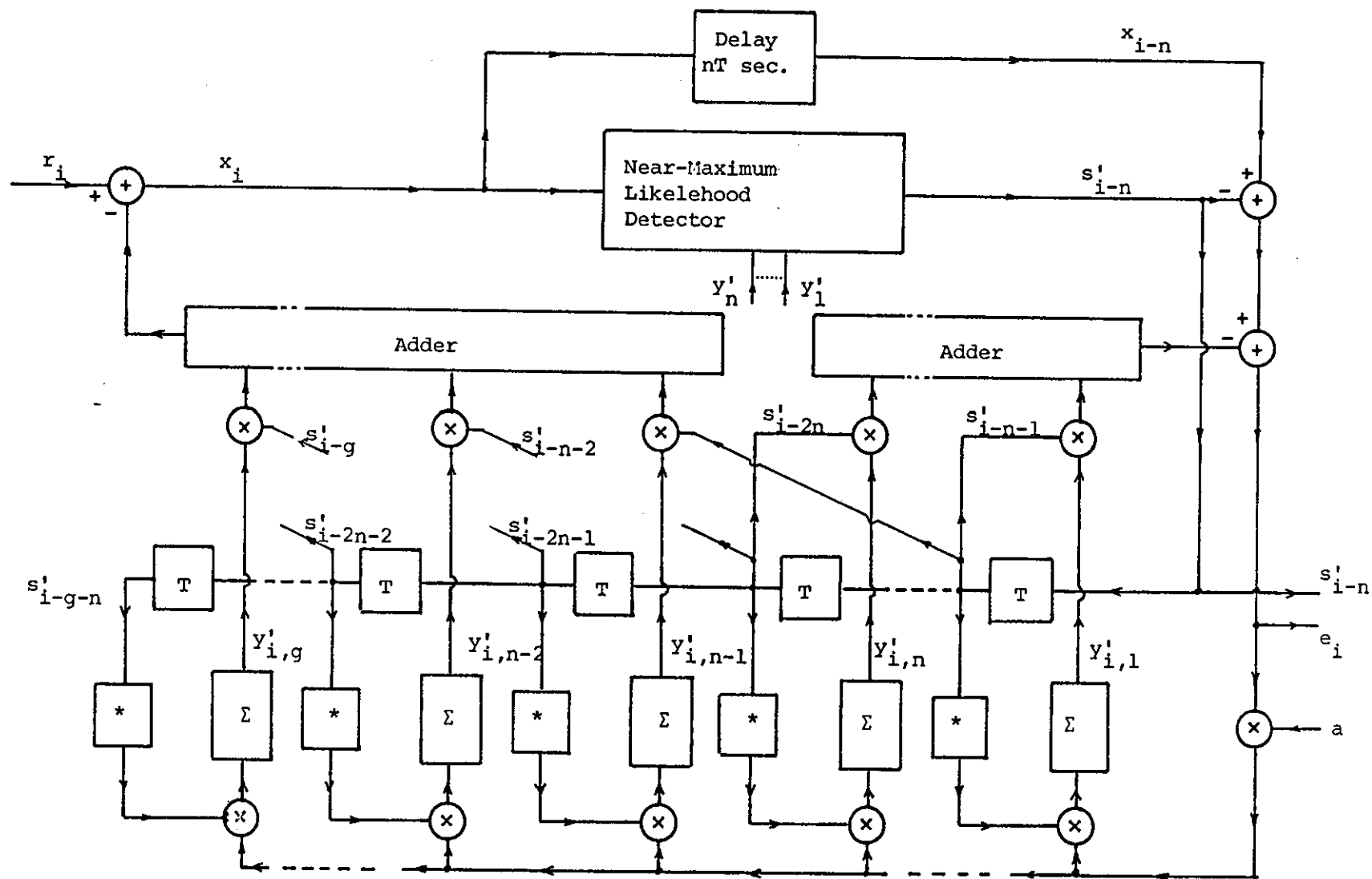


Figure 7.3.3: Adaptive near-maximum-likelihood detector

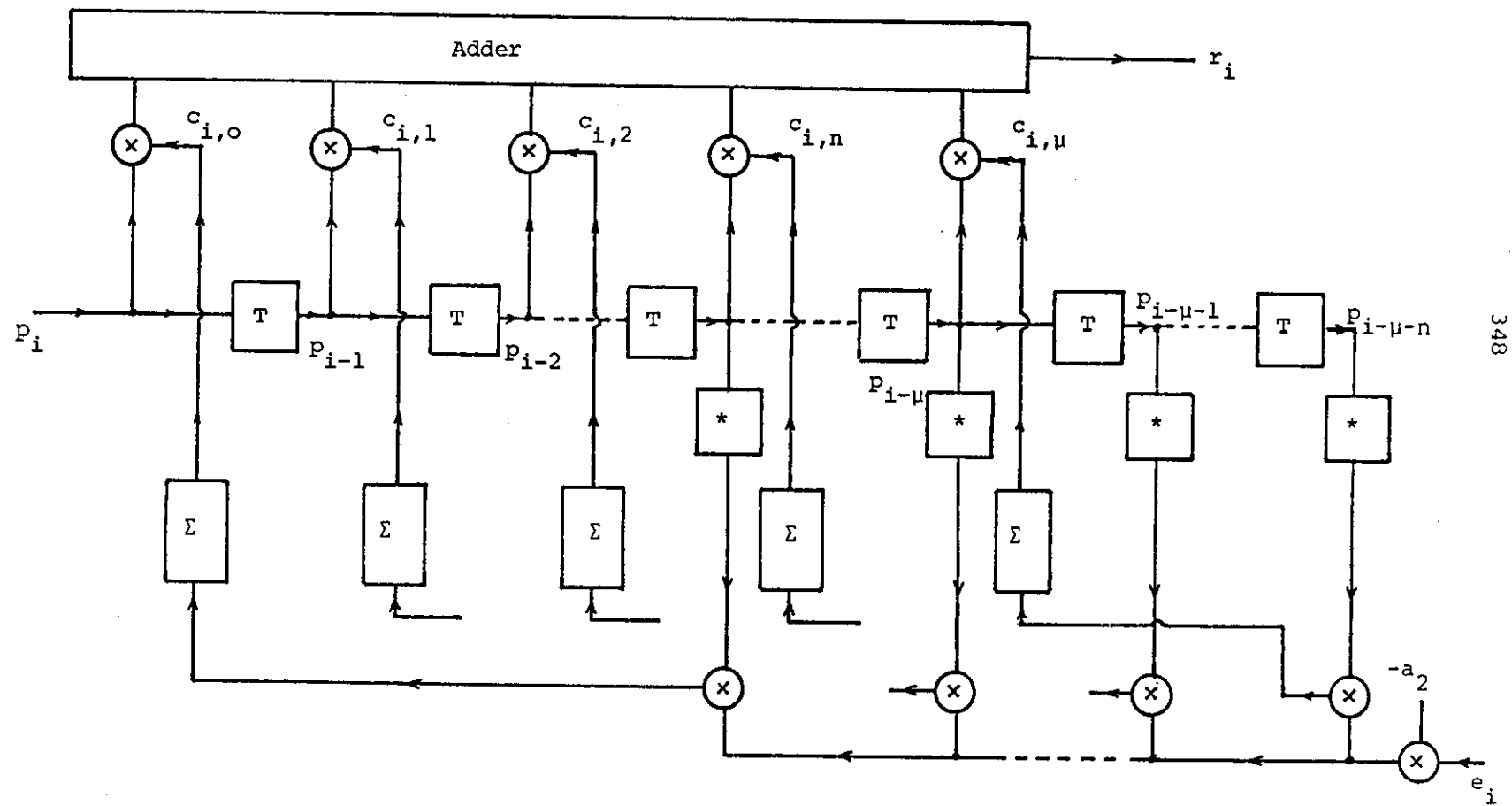
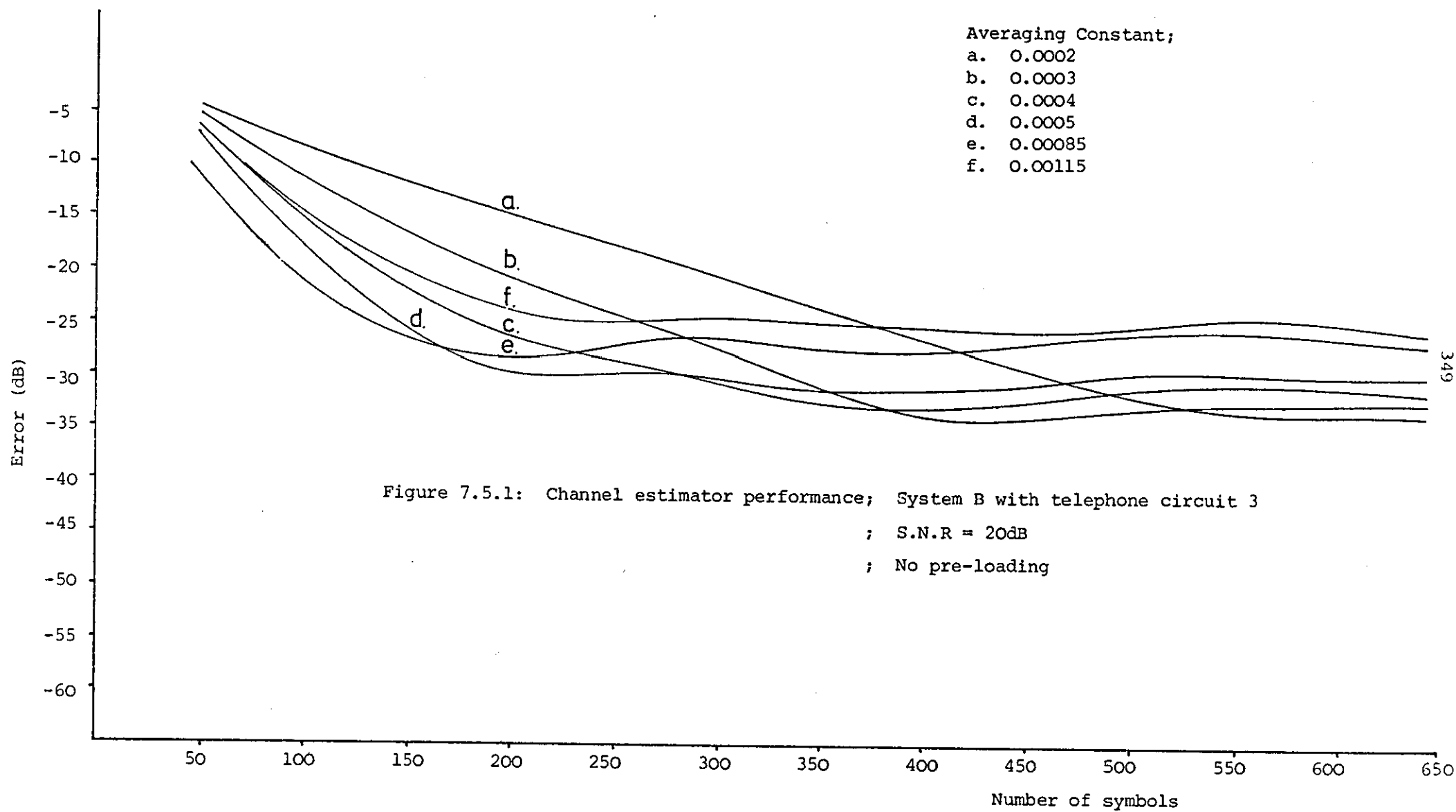
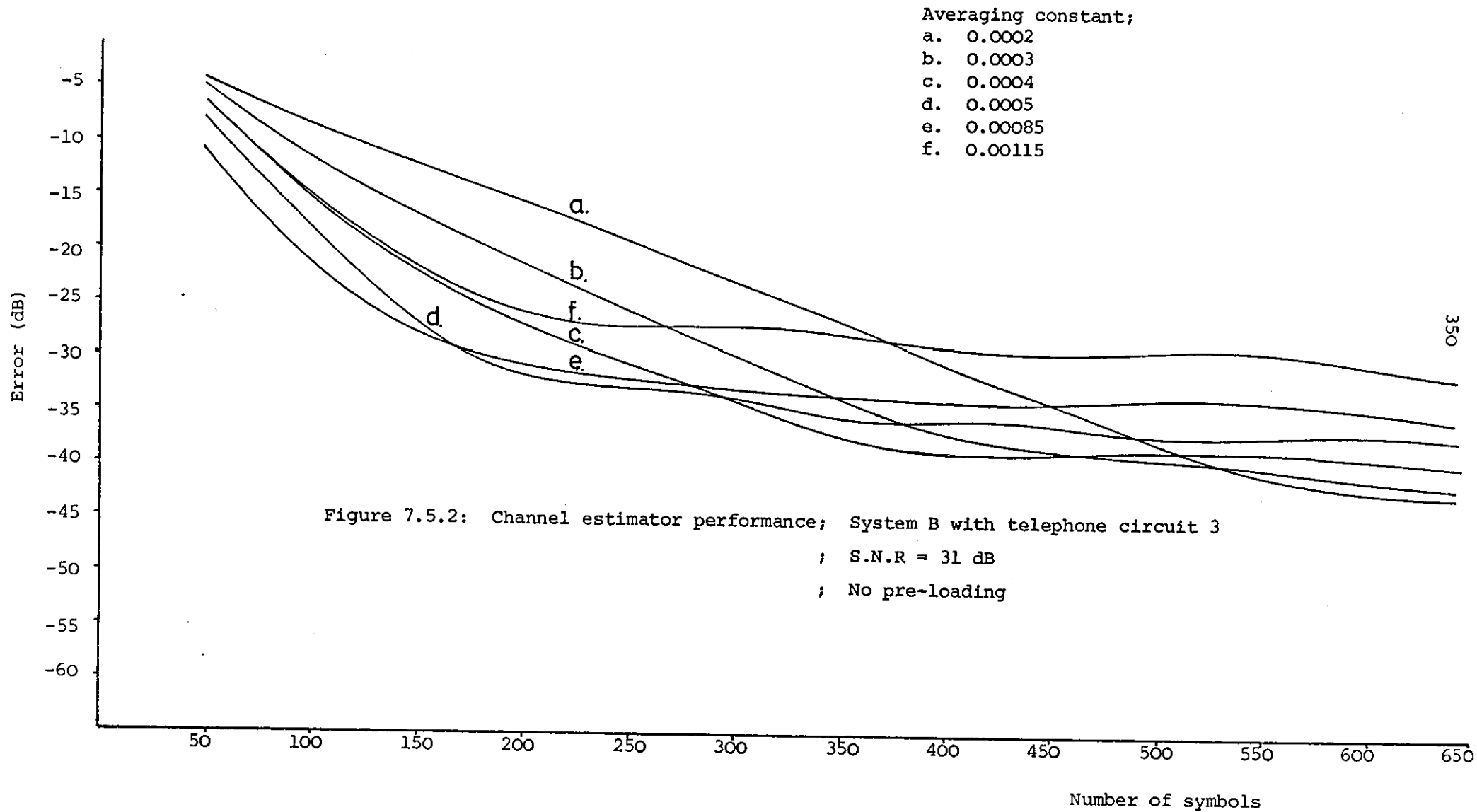
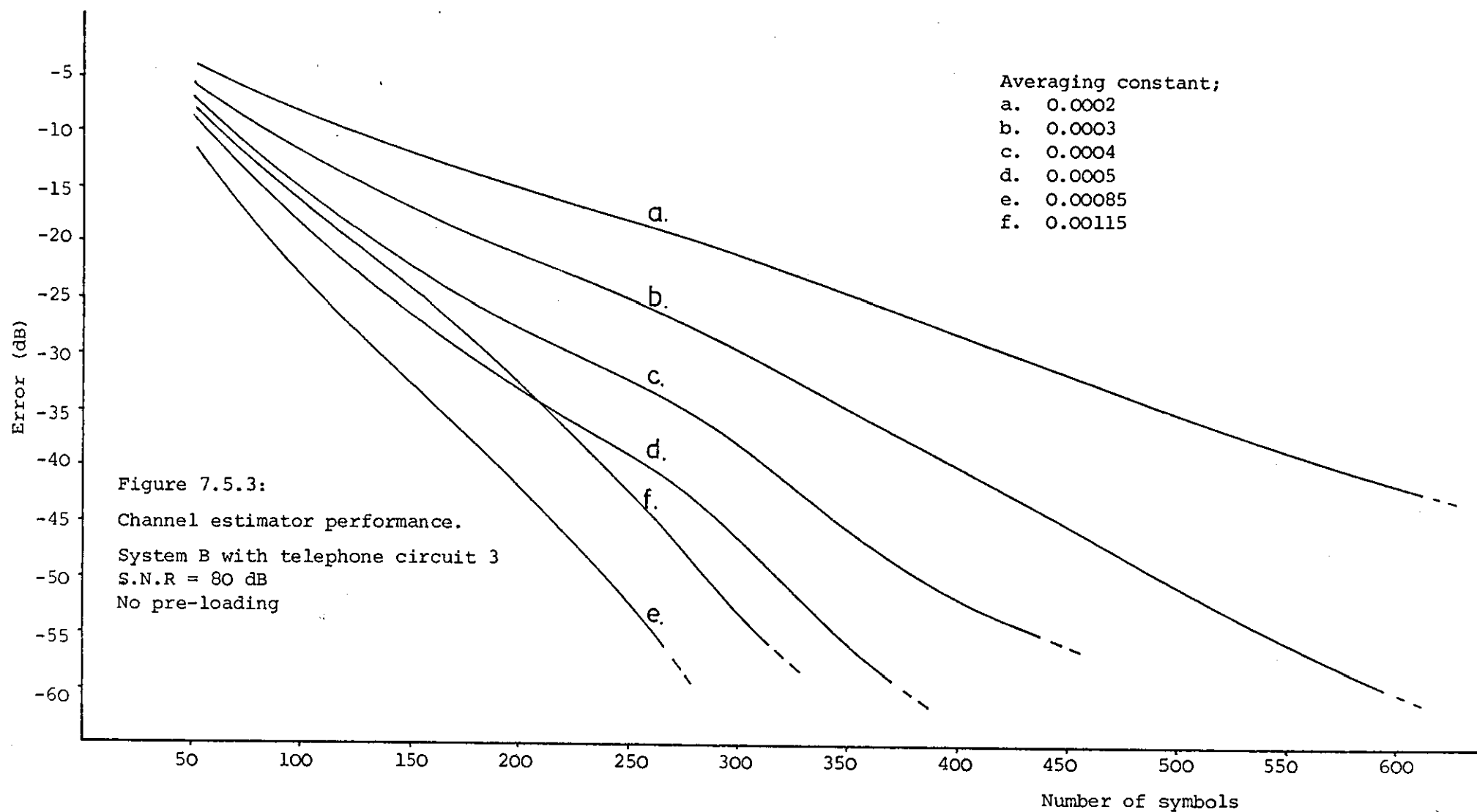
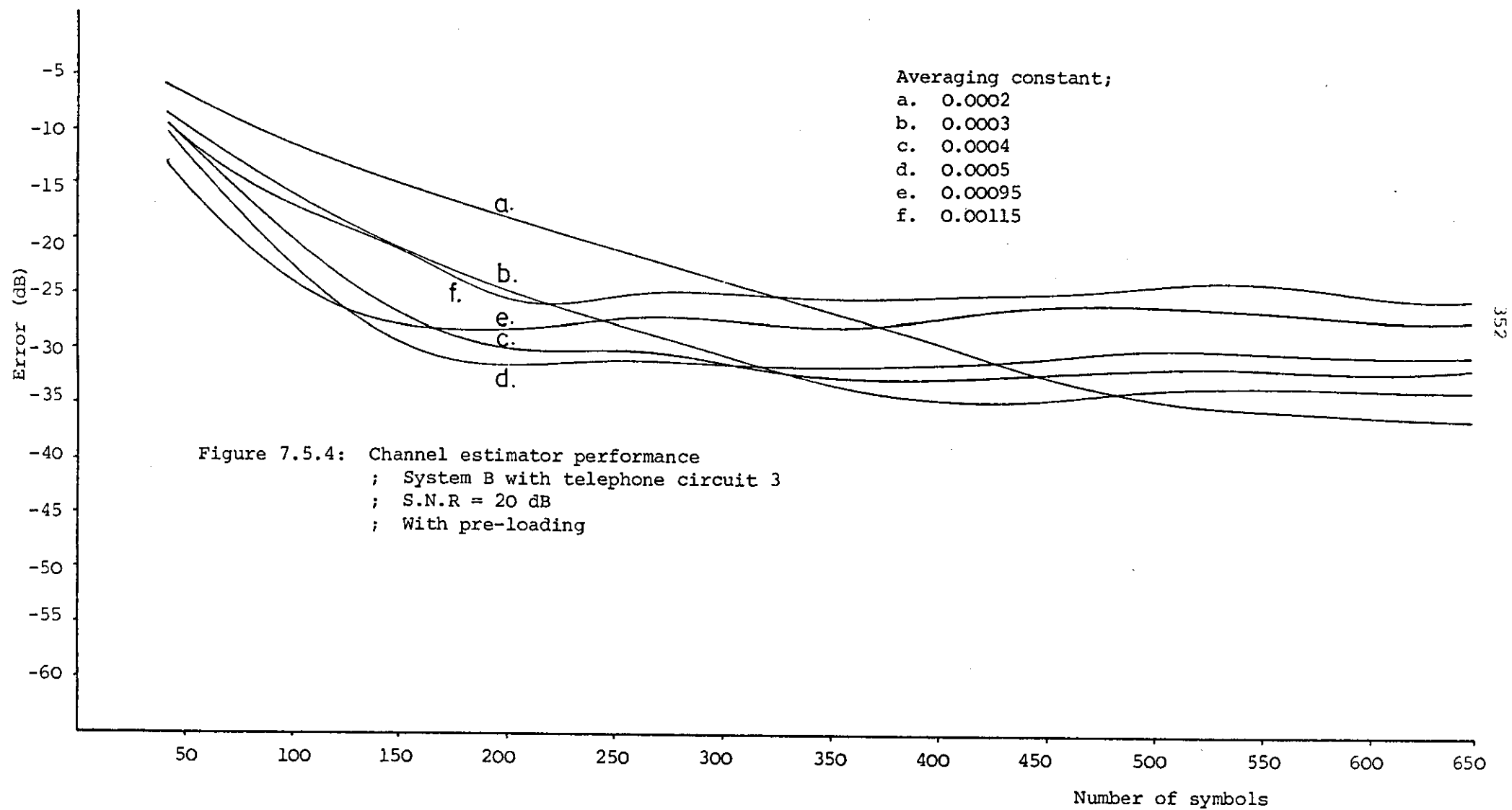


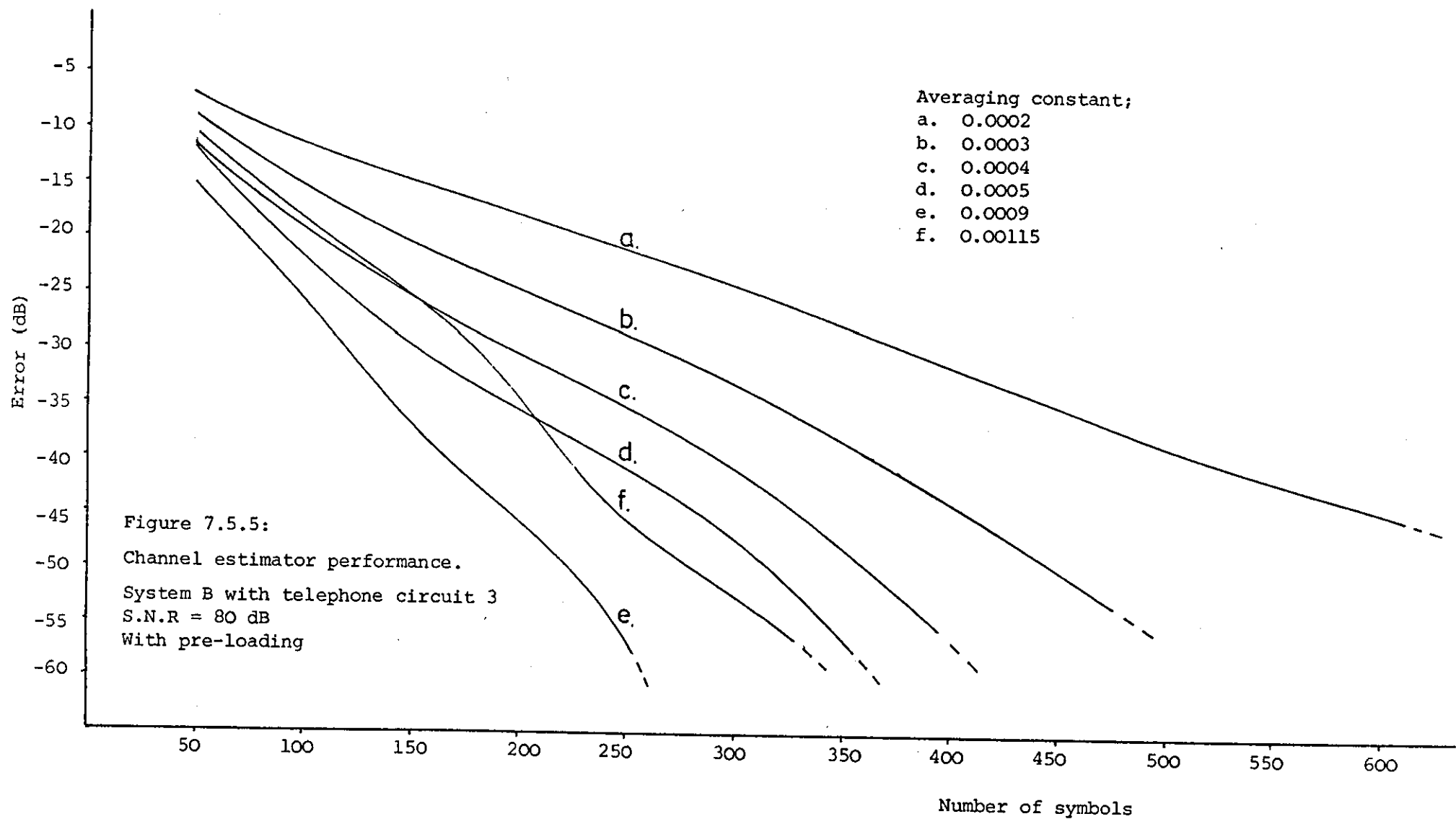
Figure 7.4.2: Adaptive adjustment of the linear pre-filter

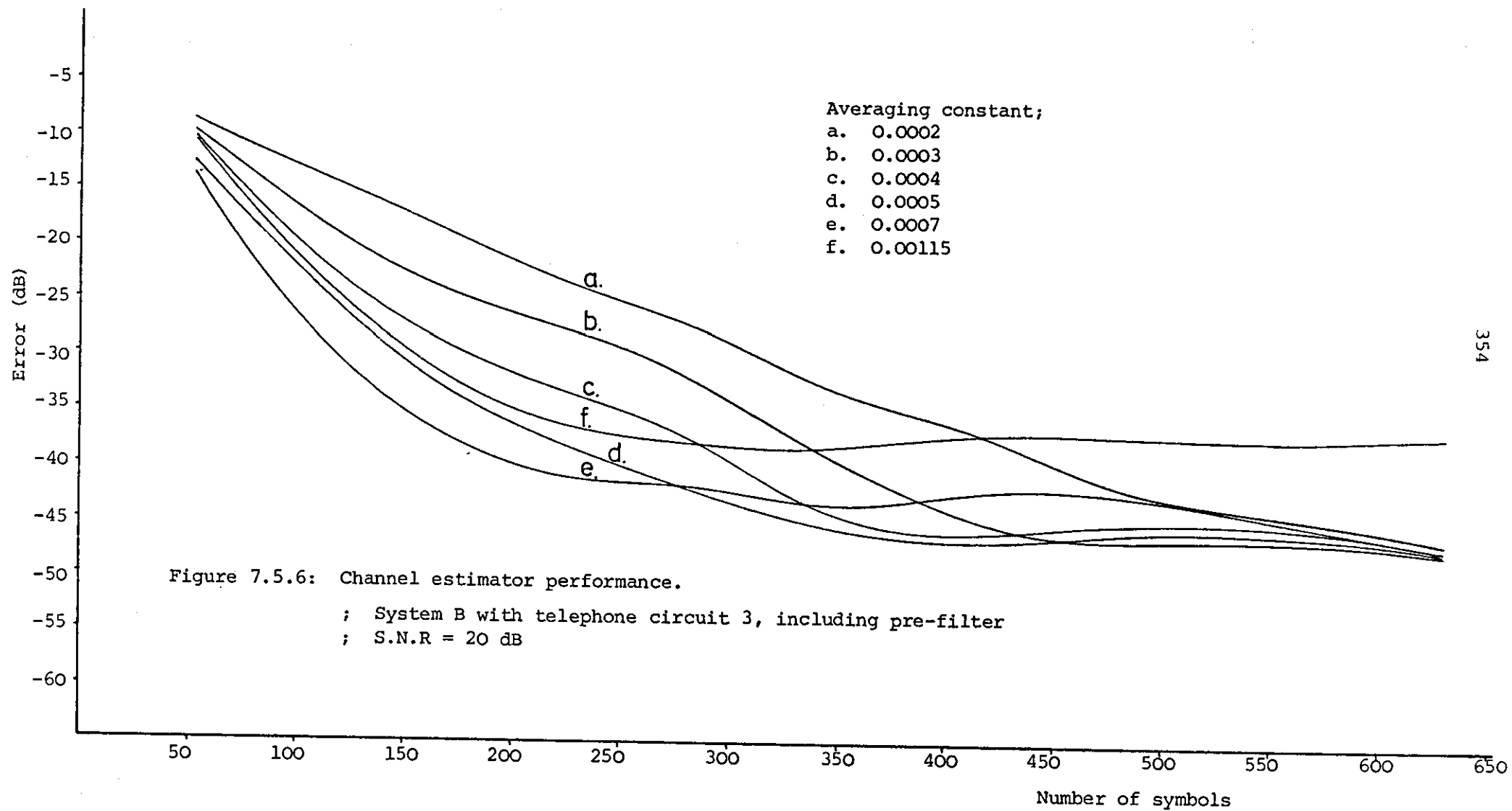


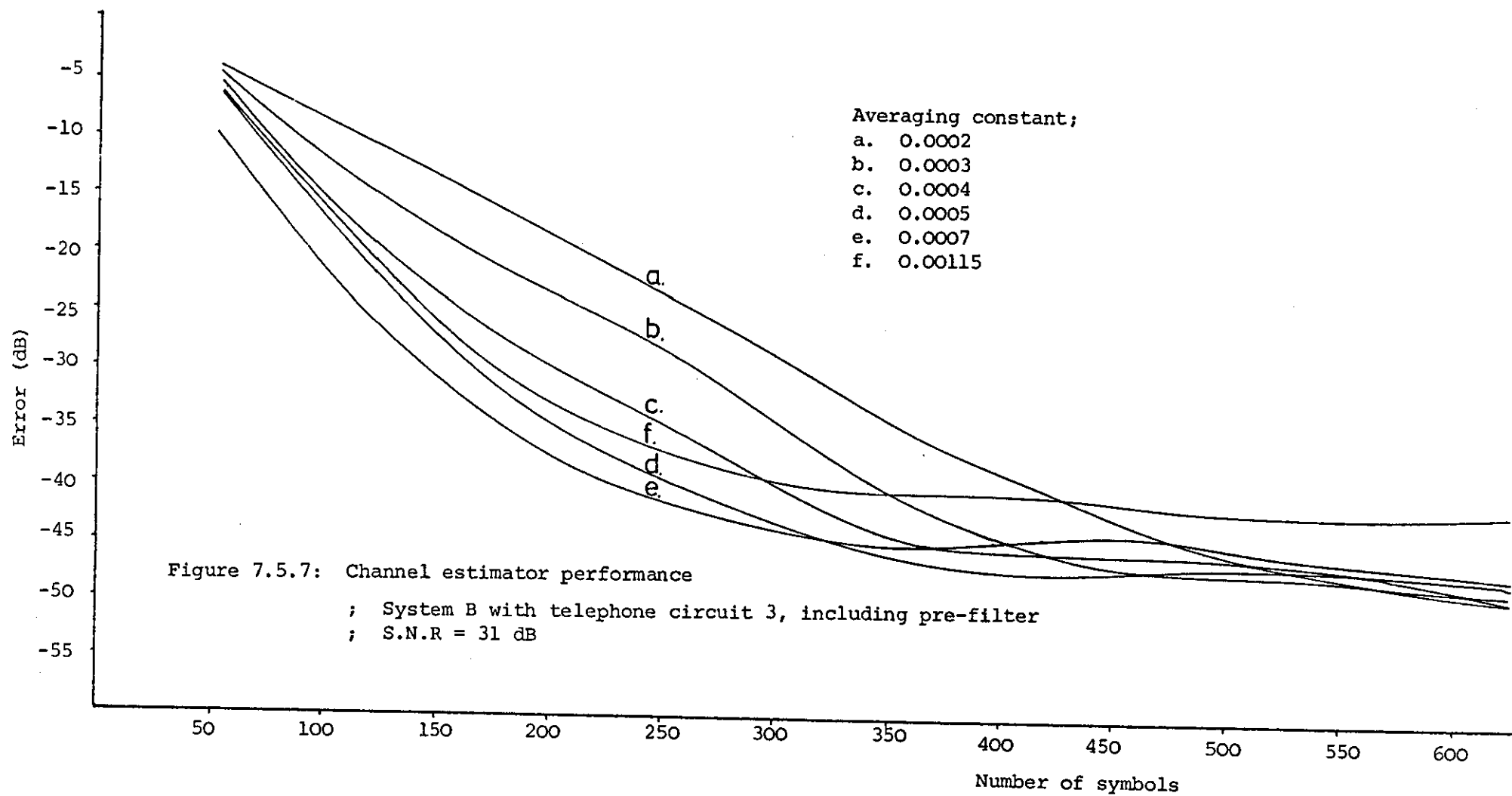


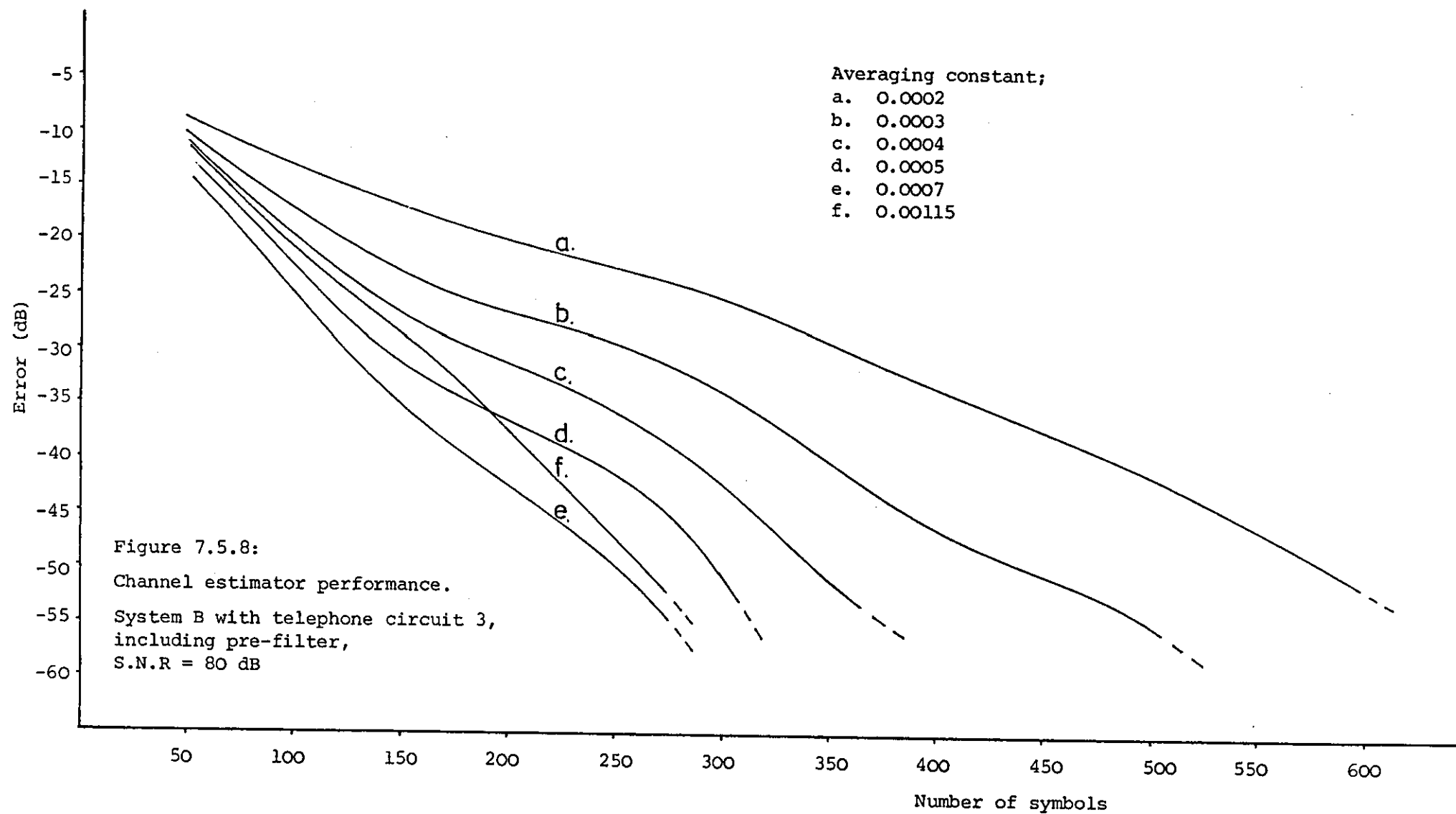












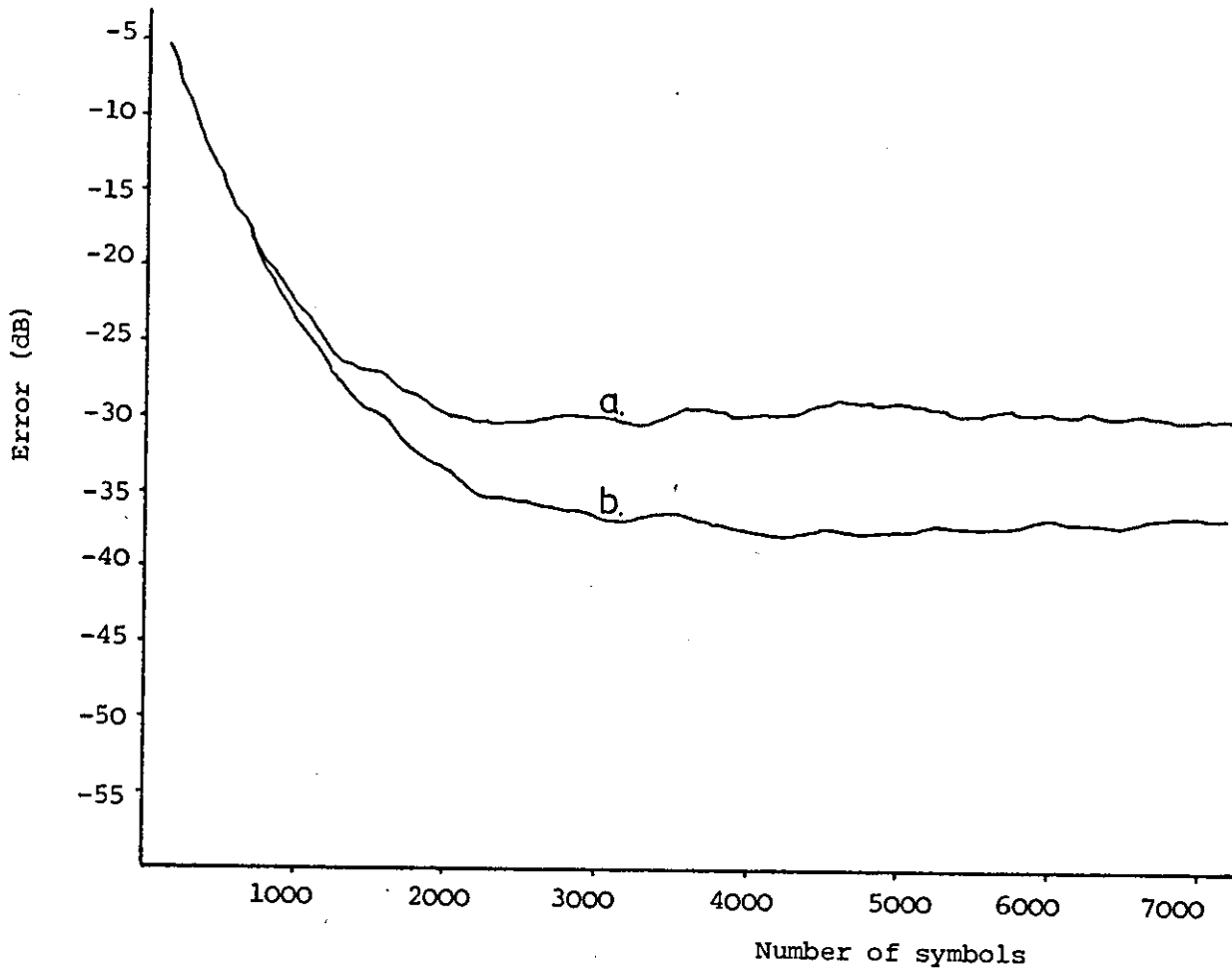


Figure 7.5.9: Performance of combined adjustment system.

System B over telephone circuit 3

$\mu = 40$

Signal-to-noise ratio; $a = 31$ dB

$b = 40$ dB

Averaging constant = 0.0005

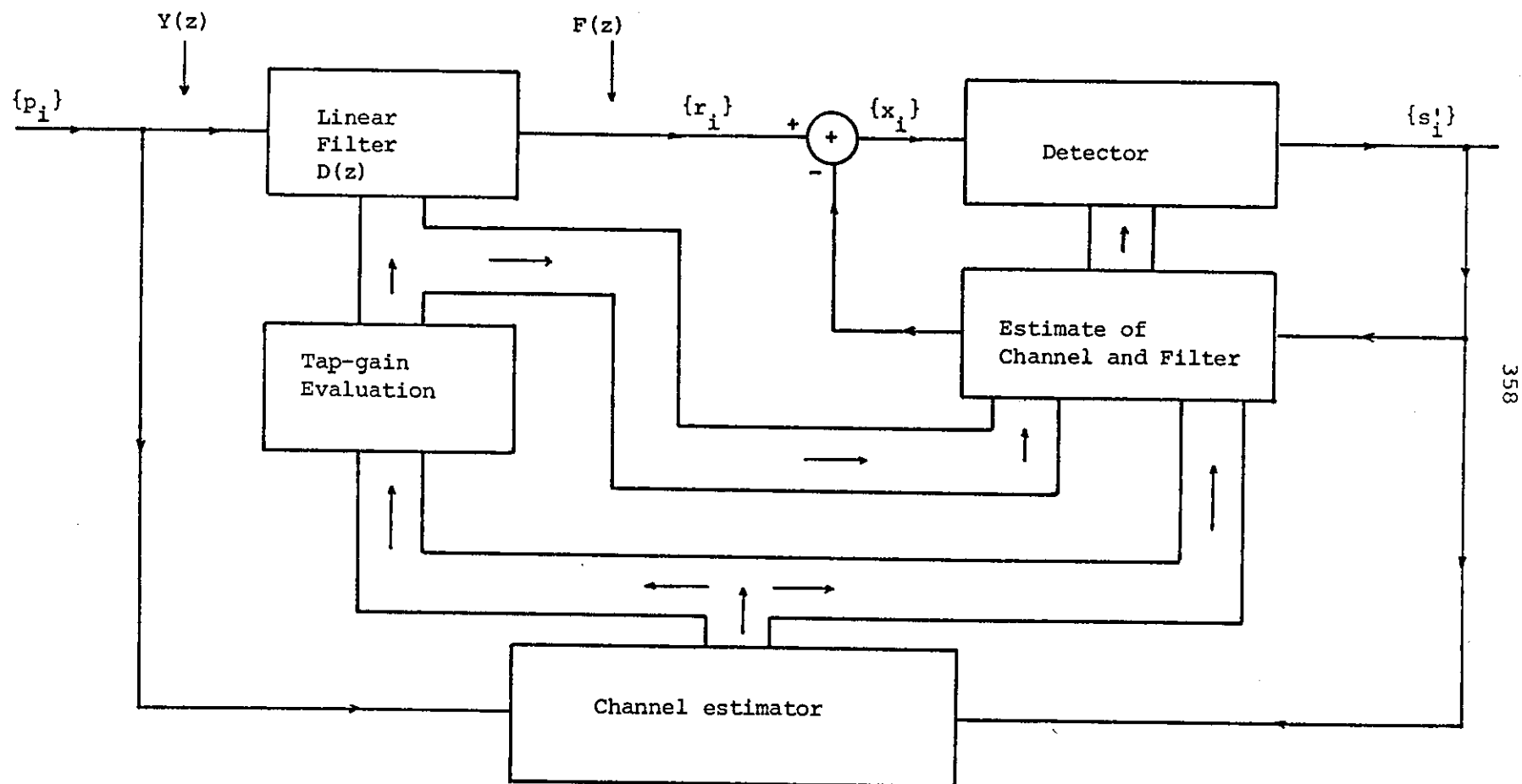


Figure 7.6.1: Basic structure of an adaptive detector

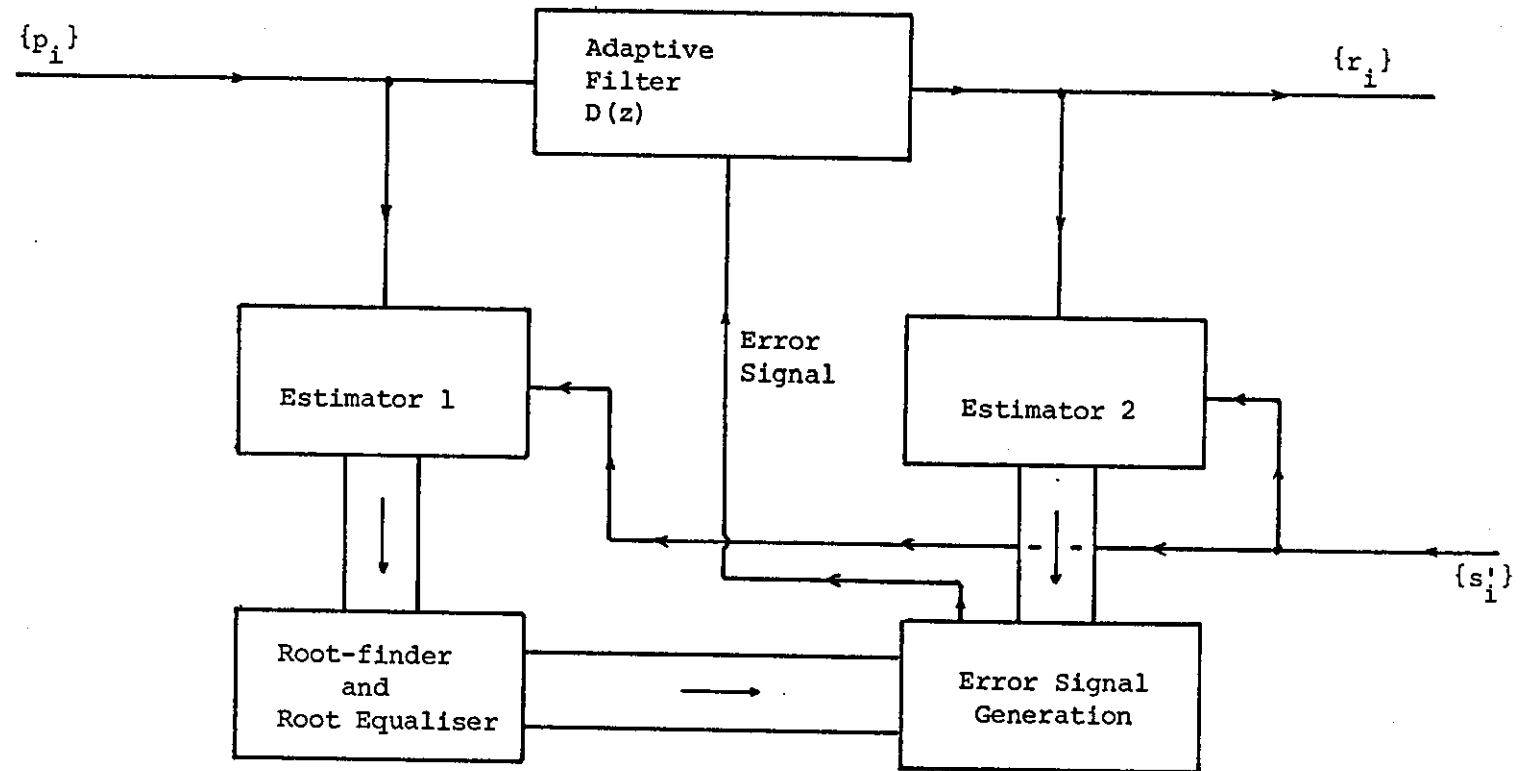


Figure 7.6.2: Alternative adjustment scheme 1

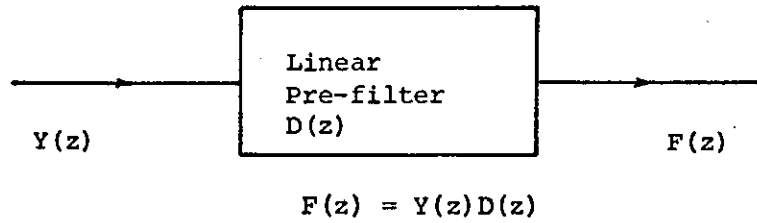


Figure 7.6.3: Relationship between the input and output z-transforms and the z-transform of the pre-filter

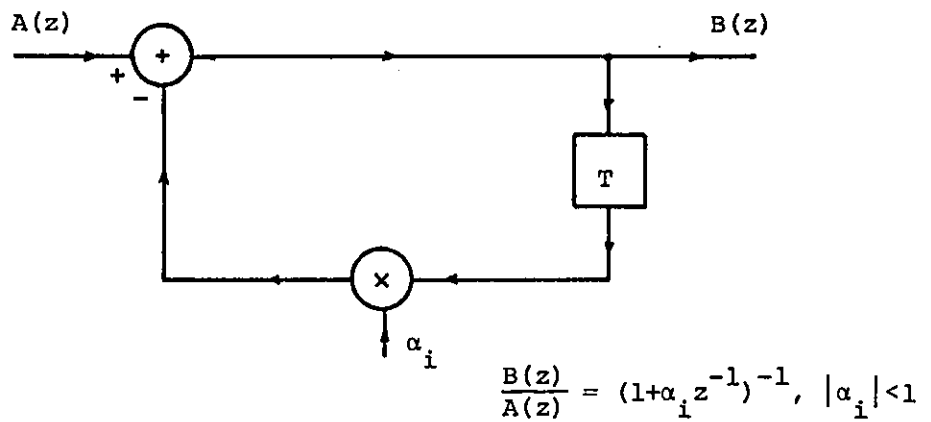


Figure 7.6.4: Single tap feedback filter section.

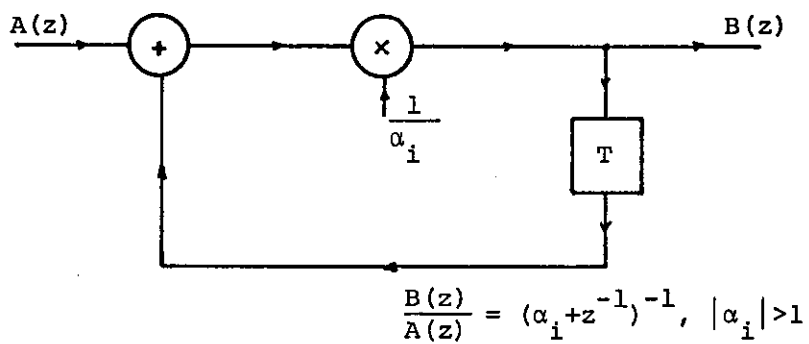


Figure 7.6.5: Modified feedback filter section.

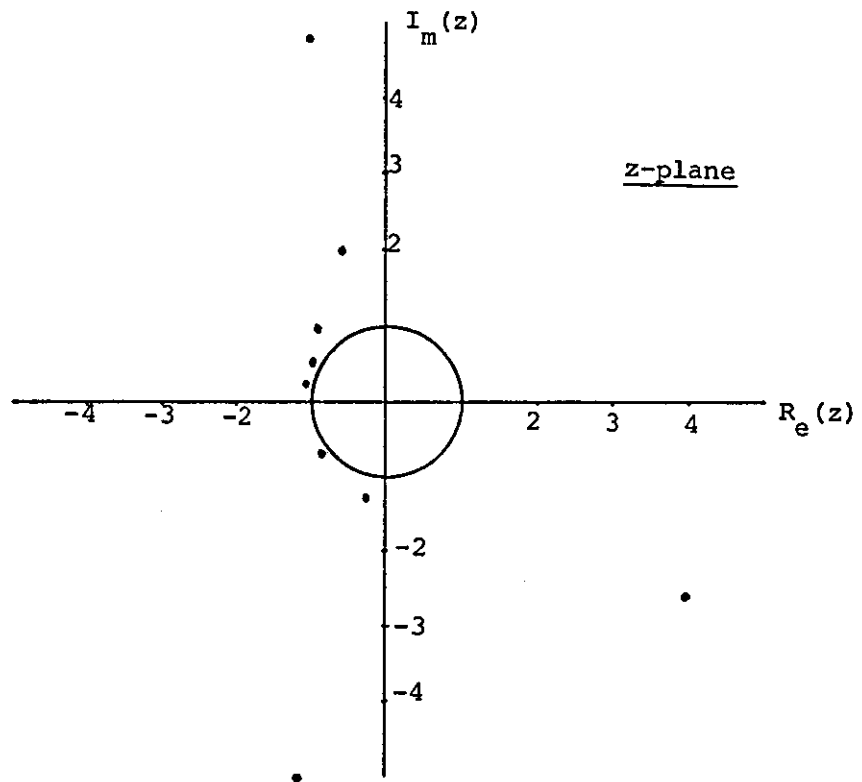


Figure 7.6.6: Locations of zeros outside the unit circle for telephone circuit 3 with filter set 1.

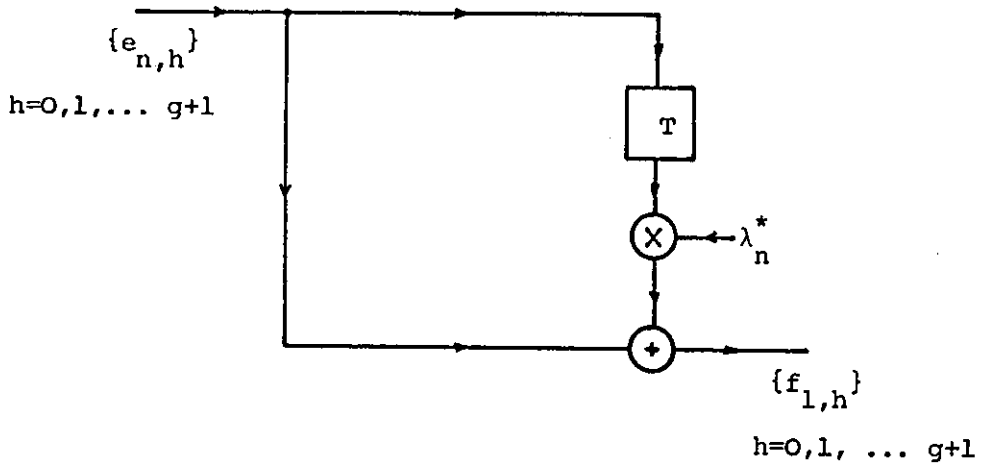


Figure 7.6.7: The two-tap feedforward filter, $B_n(z)$

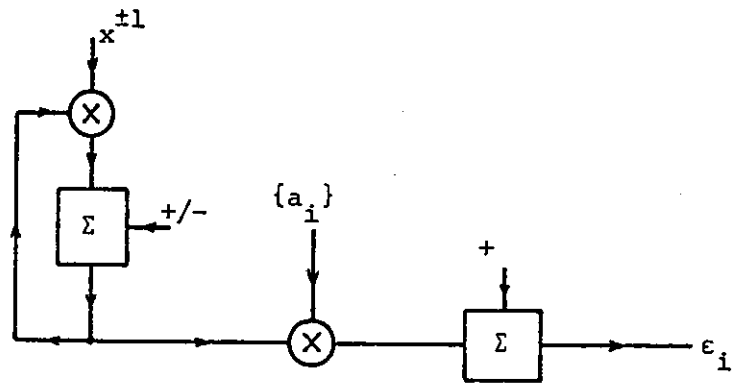


Figure 7.6.8: Possible method to implement
 $f(x) = a_0 \pm a_1 x^{\pm 1} \pm a_2 x^{\pm 2} \pm \dots$

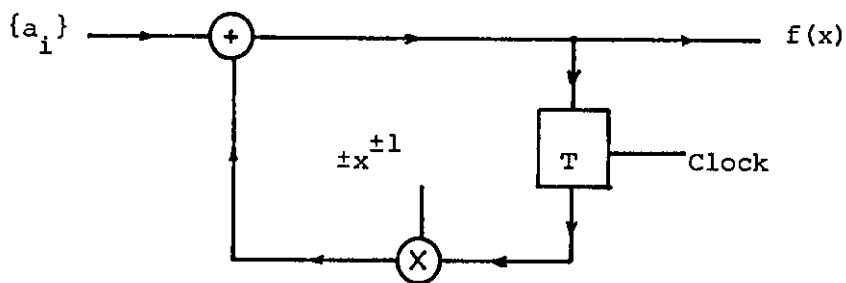


Figure 7.6.9: Simplified structure to implement
 $f(x) = a_0 \pm a_1 x^{\pm 1} \pm a_2 x^{\pm 2} \pm \dots$

8 Modem Synchronisation

8.1 Introduction

Previous chapters have made clear that the system under investigation here is a synchronous serial system using coherent demodulation. As such, the receiver must be kept in element synchronism with the transmitter, which will require the extraction of a timing waveform to sample the received signal at times $t = iT$. It is assumed here that a separate timing signal is not transmitted for reasons of efficiency and bandwidth utilisation. Also, in the discussion concerning signal impairments introduced by typical circuits, Chapter 3 has shown that frequency modulation and intermodulation effects can cause the frequency and phase of the transmitted signal to vary due to the presence of frequency offsets, constant phase errors and phase jitter. Consequently, for coherent demodulation, the receiver must be able to track such variations fairly accurately. Since the modulation method consists of two suppressed carrier amplitude modulated systems in phase quadrature and since a pilot carrier is not to be transmitted, the receiver must be able to assess and correct for any frequency and phase errors introduced by the transmission path.

A further matter for consideration is the level of the received signal. As the majority of the receiver is to be implemented using digital signal techniques, the analogue/digital interface will consist of an analogue-to-digital converter (ADC) of suitable accuracy. To ensure that the ADC operates in an efficient manner, some form of automatic gain control (AGC) must be included prior to the conversion process to counter the different amounts of attenuation introduced by different telephone circuits.

The chapter first considers the problem of carrier-phase tracking and presents a method for the estimation of frequency and phase errors.

The discussion is then extended to include the particularly troublesome effect of phase-jitter. After the introduction of a fairly simple symbol timing and start-up method, the problem of estimating the level of the received signal is addressed in order to determine a suitable AGC system.

8.2 Carrier Phase and Frequency Offset Tracking

Figure 8.2.1 shows geometrically the effects of constant phase errors and frequency offsets. Each vector in the figure represents the baseband equivalent of a received sample in the complex signal plane. If there exists a phase error of ϕ^0 between the suppressed carriers of the QAM signal and the reference carriers used for demodulation, the signal vector will be rotated by a constant angle, ϕ^0 . If the phase error is assumed to be constant in value over all time, then an equivalent effect can be produced by rotating all of the components of the complex-valued sampled impulse response by ϕ^0 in the appropriate direction. The adaptive channel estimator used in the system will now produce an estimate of the rotated sampled impulse response which will be used by one of the adjustment algorithms given in Chapter 7 and by the detector itself. It follows that a constant phase error will have little or no effect on the performance of the overall system^(9,10).

Figure 8.2.1 b) shows that a frequency difference between the suppressed carrier and the reference carrier, a frequency offset of Δf Hz, will rotate the received signal samples at a rate determined by Δf . If it is assumed that the transmitted symbol value is fixed at say, $(c+jd)$ as in Figure 8.2.1 b), the angular difference between each (demodulated) received sample will be $2\pi\Delta fT$ rads. If the sampled impulse response of the channel itself is assumed to be time invariant, then by the argument used above for a constant phase error, a frequency offset will give the impression of a time varying channel characteristic. For low

values of Δf , the channel estimator may well be able to follow the phase increments occurring on every received sample. However, as the estimator is designed to have a relatively long update period, it will not respond to the time varying phase errors due to frequency offsets of more than a few Hertz. As frequency offsets of around 5-10 Hz can be expected to occur over some telephone circuits, it is clear that some form of frequency tracking and correction system must be employed within the modem receiver.

The obvious solution to the problem of frequency offsets is to design the oscillator that provides the reference carrier waveform to be part of a phase-locked loop, as shown in Figure 8.2.2. For the application considered here, the figure is somewhat oversimplified in that it assumes the phase-locked loop can derive a suitable error signal from the modulated input signal. For QAM signals, the carrier is, of course, suppressed and so some other method of loop control is required. Many methods exist such as N^{th} -order loops, variants of the Costas loop, decision-directed or data-aided loops, etc. References 1-22 give details of some interesting solutions as well as providing an insight to the main effects of carrier-phase errors on the performance of digital data systems. Figure 8.2.3 shows the general arrangement of a data-aided carrier tracking loop which uses the received samples themselves and the detected data symbols to form an estimate of the phase error. The phase error signal, after suitable filtering, is then used to control a voltage-controlled oscillator (VCO) which acts as the reference carrier required by the demodulator. It will be recalled from Chapter 5 that the 19,200 bit/s modem developed here will use digital modulation and demodulation and so Figure 8.2.3 is not directly applicable. Figure 8.2.4 shows a modified version of the system where the received analogue signal is first digitised in an ADC. Demodulation is then achieved within some digital signal processing device by

multiplying the received samples by suitably phased samples of the two quadrature carriers. For reasons discussed in Chapter 9, the processes of demodulation and low-pass filtering have to occur at a digital sampling rate very much higher than either the carrier frequency or the signal element rate and so many samples will be necessary to define the digital equivalents of the quadrature carriers. The phase-locked loop will now itself be a digital process and so in order for it to correct any time varying phase errors, it must calculate the set of numbers which describe the sampled carrier signals at the required frequency and phase.

In terms of the amount of signal processing required to implement the demodulator, a much simpler solution is to arrange for the samples representing the quadrature carriers to be taken from a fixed set of numbers held in memory. As shown in Chapter 9, this can be achieved in practice by satisfying the condition $\frac{f_c}{f_d} = n$, where f_c is the nominal carrier frequency, f_d is the digital sampling frequency, and n is an integer. The use of a fixed set of samples implies that demodulation occurs with reference carriers whose frequency is fixed at f_c Hz and with some arbitrary but time invariant phase. The samples at the output of the post-demodulation low-pass filter will, in the presence of a constant phase error or a frequency offset, be shifted or rotated in phase as shown in Figure 8.2.1. However, given a knowledge of the phase error, each of these samples may be phase-corrected by applying a suitable phase rotation. Figure 8.2.5 illustrates the basic principles of this method.

Two carrier phase tracking loops were investigated in the course of this study, known here as carrier loops 1 and 2. Consider initially carrier loop 1 shown in Figure 8.2.6^(22,23). The loop error signal, e_1 , is first filtered by the process $F(z)$ before being applied to a digital voltage controlled oscillator (VCO), which adjusts its output so as to minimise the

phase difference between the received (and pre-filtered) samples, $\{r'_i\}$, and an estimate of these samples. The estimate is derived by passing the detected data symbols, $\{s'_{i-n}\}$, through a filter whose sampled impulse response is an estimate of the sampled impulse response of the baseband channel and pre-filter, the filter's tap coefficients being determined by any one of the adjustment schemes discussed in Chapter 7.

As can be seen from Figure 8.2.6, the detection process forms part of the phase tracking system and so any delay in detection, as produced in a near-maximum likelihood detector, must be taken into account. Consequently, at time $t = iT$, the loop error signal is,

$$e_i = \text{Im}(r'_{i-n} q_i^*) \quad (8.2.1)$$

where
$$r'_{i-n} = r_{i-n} e^{-j\hat{\phi}_{i-n}} \quad (8.2.2)$$

$$q_i = \sum_{h=0}^g s'_{i-n-h} y'_h \quad (8.2.3)$$

and where,

$\text{Im}(\cdot)$ denotes the imaginary part of (\cdot) ,

r_{i-n} is the received sample at time $t = (i-n)T$ which includes a phase rotation of ϕ_{i-n} rads,

q_i is a noiseless and ideally non-rotated estimate of r_{i-n} ,

which is formed from the detected symbols, $\{s'_{i-n}\}$ and the estimate of the sampled impulse response of the baseband channel, which is designated $\{y'_h\}$ here.

The phase update equation is given by,

$$\hat{\phi}_{i+1} = \hat{\phi}_i + F(z) \text{Im}(r'_{i-n} q_i^*) \quad (8.2.4)$$

where $\hat{\phi}_i$ is the loop's estimate of ϕ_i and $F(z)$ is the transfer function of

the loop filter^(22,31). The phase update equation can in fact, be derived from first principles by considering the tracking system as an adaptive process that attempts to minimise the mean square error between phase corrected samples, $\{r_i^c\}$ and the estimates of the $\{r_i\}$, given here by $\{q_i\}$ ^(22,32). Clearly, in the absence of noise, phase errors, errors in the $\{s_i^c\}$ and $\{y_i^c\}$, then $q_i = r_i$ when the detector introduces zero detection delay.

$$\text{If } \epsilon_i = r_i^c - q_i \quad (8.2.5)$$

$$\text{then } \epsilon_i = r_i e^{-j\phi_i} - q_i \quad (8.2.6)$$

The mean square error is,

$$E(|\epsilon_i|^2) = E((r_i e^{-j\phi_i} - q_i)(r_i e^{-j\phi_i} - q_i)^*) \quad (8.2.7)$$

and the gradient of $E(|\epsilon_i|^2)$ with respect to ϕ_i , is,

$$\frac{\partial E(|\epsilon_i|^2)}{\partial \phi_i} = E((r_i e^{-j\phi_i} - q_i)(jr_i^* e^{j\phi_i}) + (r_i e^{-j\phi_i} - q_i)^*(-jr_i e^{-j\phi_i})) \quad (8.2.8)$$

$$= E(j((r_i e^{-j\phi_i} - q_i)r_i^* e^{j\phi_i} - (r_i e^{-j\phi_i} - q_i)^* r_i e^{-j\phi_i})) \quad (8.2.9)$$

$$\frac{\partial E(|\epsilon_i|^2)}{\partial \phi_i} = E(2\text{Im}((r_i e^{-j\phi_i} - q_i)^* r_i e^{-j\phi_i})) \quad (8.2.10)$$

$$= E(-2\text{Im}(r_i e^{-j\phi_i} q_i^*)) \quad (8.2.11)$$

$$\text{hence, } \frac{\partial E(|\epsilon_i|^2)}{\partial \phi_i} = E(-2\text{Im}(r_i^c q_i^*)) \quad (8.2.12)$$

The gradient algorithm for minimising the mean square error therefore becomes, from the analysis given in Chapter 7,

$$\hat{\phi}_{i+1} = \hat{\phi}_i + \Delta \text{Im}(r_i' q_i^*) \quad (8.2.13)$$

where Δ is the averaging parameter. Comparison between equations (8.2.4) and (8.2.13) show that if Δ is replaced by the loop filter, $F(z)$, then the phase tracking loop shown in Figure 8.2.6 attempts to minimise an approximation to the mean square error between the phase corrected samples and the locally generated estimates of the $\{r_i\}$.

It should be clear from Figure 8.2.6 that due to delay in detection, the estimate of the phase error present on sample r_i is actually derived from r_{i-n}' and an estimate of r_{i-n}' . Clearly, for the case of a constant phase error, the loop should, after convergence has occurred, correct each sample regardless of any detection delay. However, for the case of a time varying phase error as produced by a frequency offset, the loop will not be able to reduce the error to a low value in the presence of a detection delay unless the loop bandwidth, controlled by the filter parameters α and β , is reduced⁽²²⁾. A reduction in the loop bandwidth will necessarily increase the transient response time of the loop to any phase perturbations, which is obviously undesirable both for initial convergence and the ability of the loop to correct for any sudden phase jumps.

Carrier loop 2^(21,23) shown in Figure 8.2.7, attempts to reduce the effects of any delay in detection by including an inner loop which tracks the phase difference between the delayed received samples and the locally generated estimates. In addition, a simple prediction process is included which forms an estimate of the phase error existing at time $t = iT$ from a knowledge of both the phase error at time $t = (i-n)T$ and the average rate of change of phase per symbol period⁽²¹⁾. For a delay in detection of n

symbol periods, the predicted phase error, at time $t = iT$, is then given by,

$$\hat{\phi}_{i/i-n} = \hat{\phi}_{i-n} + n\bar{\phi} \quad (8.2.14)$$

where $\hat{\phi}_{i/i-n}$ is the predicted phase at time $t = iT$, based on the phase at time $t = (i-n)T$

and $\bar{\phi}$ is the average rate of change of the phase per symbol period.

Since the loop filters shown in Figures 8.2.6 and 8.2.7 contain digital integrators, the output of these integrators is an estimate of $\bar{\phi}$ (21).

The phase correction signal can therefore be formed from equation (8.2.14).

Before presenting the results of computer simulation studies, the basic operation of carrier loop 2, when in the presence of constant phase errors and frequency offsets, can be illustrated by example. Consider carrier loop 2 operating at a high signal-to-noise ratio and assume that the estimates of the $\{r_i\}$ are perfect. If the received samples, $\{r_i\}$, do not contain any phase errors, then $r_{i-n} = r'_{i-n} = q_i$. The loop error signal, $\text{Im}(r_{i-n}q_i^*) = 0$, since $r_{i-n}q_i^*$ is real valued, and so no phase correction is applied to the $\{r_{i-n}\}$. When a constant phase rotation exists between the $\{r_{i-n}\}$ and the $\{q_i\}$, then,

$$(r_{i-n}q_i^*) = |r_{i-n}||q_i|e^{j(\phi_r - \phi_q)} \quad (8.2.15)$$

where $r_{i-n} = |r_{i-n}|e^{j\phi_r}$ and $q_i^* = |q_i|e^{-j\phi_q}$

$$\text{hence, } e_i = |r_{i-n}||q_i|\sin(\phi_{i-n} - \hat{\phi}_{i-n/i}) \quad (8.2.16)$$

where ϕ_{i-n} is the phase error between r_{i-n} and q_i and $\hat{\phi}_{i-n/i}$ is the estimate of ϕ_{i-n} evaluated at time $t = iT$ by the inner loop. As more samples arrive, the loop adjusts the estimated phase error towards zero. At this point in time, the error signal is zero, as is the output from the loop

filter. The output from the predictor therefore remains constant.

It should be realised from equation (8.2.16) that the phase detector used in the loop has a sinusoidal characteristic; the use of differential coding will, however, remove any phase ambiguities due to false-lock⁽³¹⁾. It is also interesting to note that the error signal is proportional to $|r_{i-n}| |q_i|$. For fairly small QAM structures, the presence of this term in the error signal has been shown to have only a small effect on the loop's performance^(22,23). However, for large constellations, such as the 64-point structure considered here, tests have proved it necessary to either remove the sensitivity by normalisation or to limit the range of the error signal by using a soft limiter. The presence of a frequency offset of Δf Hz will rotate the received samples by an amount $2\pi\Delta fT$ radians per symbol period relative to the reconstructed samples. After a time determined by the loop parameters, α and β , the output from the loop filter's digital integrator will be $2\pi\Delta fT$. The loop error signal is then zero and the output to the phase corrector is a stepwise ramp function, where each step equals $2\pi\Delta fT$ rads.

To test the performance of the two carrier phase tracking loops, computer simulations were performed which subjected the loops to various amounts of constant phase errors and frequency offsets; the simulation program appears as Program 8 in Appendix 10. Figures 8.2.8-8.2.10 show the transient response of carrier loop 1 when operating at a high signal-to-noise ratio over an ideal baseband channel. The figures clearly show the effect of the loop parameters, α and β , on the convergence rate. In general, α controls the averaging properties of the loop and hence its ability to track frequency offsets, whereas β determines the response of the loop to sudden phase changes. Throughout the simulations, $\beta = 2\alpha$ which has been shown to be the optimum relationship between these parameters⁽²²⁾.

Clearly, increasing α and β will reduce the response time of the loop but, as will be seen later, also increases its sensitivity to noise. Figure 8.2.10 shows that the loop will converge to a small steady-state phase error in just over 300 symbol periods when initially disturbed with a 90° constant phase error and a 10 Hz frequency offset; the delay in detection here is set to zero. Figure 8.2.11 shows the loop operating under similar conditions, with the same values of α and β but with a delay in detection equal to 3 symbol periods. It is clear that convergence has not occurred within 300 symbol intervals. In fact, to enable the loop to converge, α and β must be reduced which will, of course, lengthen the convergence time. Figure 8.2.12 shows an extreme case of a detection delay of 8 symbols where α and β have been reduced to 0.0015 and 0.003 respectively. Again, convergence has not occurred within the 2500 symbol periods tested. The performance of carrier loop 2 under similar operating conditions is shown in Figures 8.2.13 and 8.2.14, from which it is clear that the addition of the prediction process greatly enhances loop performance in the presence of any detection delay. Clearly, as the detection process discussed in Chapter 6 includes a delay in detection of 8 symbol periods, carrier loop 2 is to be preferred to loop 1 for the application considered here.

Figures 8.2.15 to 8.2.21 detail the performance of carrier loop 2 when operating upon samples which have been transmitted over the combination of telephone circuit 3 and filter set 2 at signal-to-noise ratios of 20 and 30 dB, where signal-to-noise ratio is defined in Appendix 3. In all cases, the detected data symbols and the estimate of the sampled impulse response of the system are assumed to be ideal to illustrate the performance of the phase tracking system in isolation. Also, the phase error shown in the figures is that at the input to the detector and so includes the random phase perturbations due to the additive noise. It is clear from

Figures 8.2.15 and 8.2.16 that the presence of fairly high levels of additive noise increases the convergence time of the loop when operating with detection delays greater than zero. The effects of noise can, however, be reduced by reducing α and β , but this will further increase the convergence time. An extreme case of the effects of noise can be seen in Figure 8.2.17, which shows the loop operating with a detection delay of 8 symbol periods at a signal-to-noise ratio of 20 dB; the effects of high noise, fairly large loop bandwidth and a large detection delay combine to give the system a very poor performance.

One method of reducing the effects of noise whilst at the same time maintaining a good rate of initial convergence is to use a loop bandwidth which reduces with time. Figures 8.2.18 to 8.2.21 show the results of this modification, where α_i obeys the function shown in Figure 8.2.22 and given by

$$\alpha_i = \begin{cases} \alpha_0 & i < 500 \text{ symbols} \\ \alpha_{i-1} - \Delta(\alpha_{i-1} - \alpha_f) & i \geq 500 \text{ symbols} \end{cases}$$

$$\text{and } \beta_i = 2\alpha_i, \quad \text{all } i \quad (8.2.17)$$

and where α_0 is the initial value of α ,

Δ controls the rate of decay

and α_f is the final value of α .

It can be seen from Figures 8.2.18 to 8.2.21 that the convergence rate of the loop depends now on the values of α_0 and Δ ; too high a value of α_0 will cause instability, as shown in Figure 8.2.19, whilst too small a value of Δ results in an increased convergence period, as shown in Figure 8.2.20. A good compromise therefore appears to be $\alpha_0 = 0.015$, $\Delta = 0.005$ and $\alpha_f = 0.0015$, Figure 8.2.21, which gives convergence within approximately 1500 symbols when the loop is tracking a 90° phase offset and 10 Hz frequency

offset at 30 dB signal-to-noise ratio. The major problem with the fading bandwidth modification is that once the loop has converged and the loop parameters have been reduced, any change in the phase or frequency offset introduced by the channel will not be tracked rapidly. This is clearly an undesirable situation when operating over telephone channels employing carrier links. A possible solution to this problem is to monitor the absolute value of the error signal before any limiting device and to re-set α and β to their initial values when the error signal exceeds a pre-determined level. However, the choice of triggering level and the values of α_0 , Δ and α_f all depend on the expected levels of phase and frequency offset and so some compromise will be necessary in a practical system design. Finally, Figure 8.2.23 shows the relationship between the RMS phase error and signal-to-noise ratio when the loop is operating under the conditions given in Figure 8.2.21, where RMS phase error is defined as,

$$\phi_{e \text{ RMS}} = \left(\frac{1}{3000} \sum_{i=3000}^{6000} (\phi_i - \hat{\phi}_i)^2 \right)^{1/2} \quad (8.2.18)$$

where, ϕ_i is the phase rotation on r_i
and $\hat{\phi}_i$ is the loop's estimate of ϕ_i .

It will be seen from equation (8.2.18) that the RMS phase error is calculated after convergence and so gives an indication of the steady-state phase error of the system.

8.3 Carrier Phase Jitter

Carrier phase jitter is essentially any perturbation in the phase of the (suppressed) signal carrier although the term is generally used to describe sinusoidal variations in phase at mains frequencies and harmonics. If $\phi(t)$ represents the total time varying phase error after demodulation by

a fixed carrier frequency, then,

$$\phi(t) = \phi_0 + 2\pi\Delta f t + \phi_j(t) \quad (8.3.1)$$

where ϕ_0 is a constant phase offset

Δf is a frequency offset

and $\phi_j(t)$ is the jitter component

Assuming that the jitter term is harmonically related to 50 Hz, then,

$$\phi_j(t) = \sum_{i=1}^k a_i \sin(2\pi \cdot 50 \cdot i t + p_i) \quad (8.3.2)$$

where a_i is the amplitude of the phase variation associated with the i^{th} harmonic of 50 Hz and p_i is the corresponding positional offset.

Phase jitter, as with any phase error present in the system, will result in the received signal being rotated in phase. It is clear from Figure 8.2.1 that the combination of a constant phase error, frequency offsets and phase jitter can have a catastrophic effect on the performance of a detector, even in the absence of additive noise; it should also be clear from Figure A5.9 that the problem of phase errors becomes worse as the size of the signal constellation increases.

As the carrier phase tracking system shown in Figure 8.2.7 has been shown to track frequency offsets and constant phase errors, it is obviously worth examining its effectiveness for reducing phase jitter. As mentioned in Chapter 3, phase jitter over telephone lines tends to consist of sinusoidal phase variations at 50, 150 and 250 Hz. A 10° peak-to-peak jitter component at 250 Hz implies the presence of a frequency offset varying with time with a peak value of approximately 22 Hz. Consequently, the loop bandwidth must be increased in order to enable the loop to track the fairly rapid rate of change of phase with time. Moreover, as the simple prediction process operates from a knowledge of the average rate of change of phase

with time, carrier loop 2 is unlikely to give any advantage over loop 1 when attempting to track phase jitter.

Figures 8.3.1 to 8.3.6 show the performance of carrier loop 2 when tracking phase jitter under ideal conditions. Figure 8.3.1 shows the response of the loop to a 50 Hz, 10° peak-to-peak phase jitter component with $\alpha = 0.015$, $\beta = 0.03$. Clearly, the phase error contains the jitter function, albeit with a slightly reduced amplitude, approximately 9.75° peak-to-peak. The combination of a 90° phase offset and a 20° peak-to-peak 50 Hz jitter component is shown in Figure 8.3.2, where, as expected, the loop is able to remove the constant component, but not the jitter. In this case, the RMS phase error is equal to the RMS value of the jitter component, which is approximately 7° . Increasing the loop parameters, α and β , has the expected result of reducing the peak-to-peak phase jitter present on the corrected samples, as shown in Figures 8.3.3 and 8.3.4, where, in the latter figure, the values $\alpha = 0.4$, $\beta = 0.8$ reduce a 10° peak-to-peak 50 Hz jitter component to approximately 1° peak-to-peak. If the jitter frequency is now increased to 250 Hz whilst maintaining the values $\alpha = 0.4$, $\beta = 0.8$, the loop is less successful in that it reduces a 10° peak-to-peak component to approximately 3.5° peak-to-peak, Figure 8.3.5. However, reference to Figure 8.2.19 shows that the loop will be unstable when operating with such high values of α and β at signal-to-noise ratios somewhat greater than 30 dB. Finally, Figure 8.3.6. illustrates the composite phase error remaining after phase correction when 10° peak-to-peak jitter components at 50 and 150 Hz are included on the signal samples. As expected from the above results, the 150 Hz jitter component has the largest influence on the phase error.

It should be clear from the results of the computer simulation studies presented above that whilst carrier loop 2 can successfully track and remove any constant phase and frequency offsets, it cannot reduce rapidly

time-varying phase components to a sufficiently low level to ensure the correct operation of the detector, even in the absence of noise. Consequently, the receiver will fail completely if the data is transmitted over telephone channels introducing more than a few degrees peak-to-peak phase jitter at 50 Hz and smaller amplitudes at the higher jitter frequencies.

One solution to the problem of phase jitter suppression is to include three extra tracking loops in the carrier phase correction system. By careful filter design, the loops can be adjusted to respond to the components of the error signal at 50, 150 and 250 Hz, respectively, whilst exhibiting fairly narrow band characteristics to reduce the combined noise bandwidth. The outputs from the three jitter loops can then be added along with the output of the main frequency offset tracking loop, to produce the required phase correction. In a recent article, Bingham⁽²⁵⁾ has proposed a similar scheme which uses a single loop whose filter contains a low-pass section and two bandpass sections. The low-pass section permits the tracking of small frequency offsets and constant phase errors, whilst the two bandpass sections, centred on 60 and 120 Hz, allow the tracking of jitter components at these frequencies. It is interesting to note that this American author has found it necessary to include only a second harmonic response rather than the third and fifth harmonics considered relevant here.

It has been shown elsewhere⁽²²⁾ that the multi-loop approach, after optimisation of the various loop parameters, can substantially reduce the deterministic components of phase jitter at the expense, however, of a higher mean-square phase error due to the increased noise bandwidth of the system. One method to reduce this degradation in performance is to add selectivity to the system such that particular jitter loops are only included when required. Clearly, this requires some means of sensing the presence of the jitter components at the three frequencies of interest. This can be

achieved in practice by forming an error signal which is not part of the closed loop system and applying this signal to three narrowband filters. The fact that the spectrum of the error signal will contain components at the jitter frequencies can be seen from Figures 8.3.1 to 8.3.6, and from Figure 8.3.7, which shows the spectrum of the error signal after phase jitter at 50 Hz and all harmonics of 50 Hz up to 250 Hz is added to received signal samples. The three outputs from the narrowband filters are then monitored and used to select the jitter loops as required.

Although the system proposed above provides a suitable solution to the problem of phase jitter correction, the convergence of the system as a whole is fairly slow. The major reason for this is that the individual jitter loops must strive to estimate both the amplitude (degrees peak) and the temporal position (i.e. the "phase") of the jitter components. In an interesting series of experiments performed by Cheung and Bateman⁽²⁴⁾, the main loop error signal was applied to a bank of narrow bandpass filters where each filter was centred on one of the odd harmonics of 50 Hz, up to 250 Hz. If the amplitude and phase responses of the filters are known and given by $A(\omega)$ and $\phi(\omega)$, respectively, then the output from each filter $x_i(t)$, at time $t = kT$, assuming a high signal-to-noise ratio, is

$$x_{i,k} = \theta_{i,k} A(\omega_{i,k}) \sin(\omega_{i,k} kT + p_{i,k} + \phi(\omega_{i,k})) \quad (8.3.3)$$

where $i = 1, 3, 5$ denotes the harmonics of 50 Hz, $\theta_{i,k}$ is the peak amplitude of the jitter component whose frequency at time $t = kT$ is $\omega_{i,k}$ rads/s, and $p_{i,k}$ is the positional offset (or "phase") of that jitter component. Each $x_{i,k}$ is then multiplied by

$$\frac{2}{A(\omega_{i,k-1})} e^{j(\omega_{i,k-1} kT + \phi(\omega_{i,k-1}) + p_{i,k-1})} \quad (8.3.4)$$

to give,

$$\frac{\theta_{i,k} A(\omega_{i,k})}{A(\omega_{i,k-1})} \{c_{i,k} + jd_{i,k}\} \quad (8.3.5)$$

where

$$\begin{aligned} c_{i,k} = & \sin((\omega_{i,k-1} + \omega_{i,k})kT + p_{i,k-1} + p_{i,k} + \phi(\omega_{i,k-1}) + \phi(\omega_{i,k})) \\ & + \sin((\omega_{i,k-1} - \omega_{i,k})kT + (p_{i,k-1} - p_{i,k}) + (\phi(\omega_{i,k-1}) - \phi(\omega_{i,k}))) \end{aligned} \quad (8.3.6)$$

and

$$\begin{aligned} d_{i,k} = & -\cos((\omega_{i,k-1} + \omega_{i,k})kT + p_{i,k-1} + p_{i,k} + \phi(\omega_{i,k-1}) + \phi(\omega_{i,k})) \\ & + \cos((\omega_{i,k-1} - \omega_{i,k})kT + (p_{i,k-1} - p_{i,k}) + (\phi(\omega_{i,k-1}) - \phi(\omega_{i,k}))) \end{aligned} \quad (8.3.7)$$

If $\omega_{i,k-1} \approx \omega_{i,k}$, then $(\omega_{i,k-1} + \omega_{i,k}) \approx 2\omega_{i,k}$ and $(\omega_{i,k-1} - \omega_{i,k}) = \Delta\omega_{i,k}$.

Also, if $\phi(\omega)$ and $A(\omega)$ both change only slowly around $\omega = \omega_i \pm \Delta\omega_i$, then equations (8.3.6) and (8.3.7) may be written as,

$$\begin{aligned} c_{i,k} \approx & \sin(2\omega_{i,k}kT + p_{i,k-1} + p_{i,k} + 2\phi(\omega_{i,k})) \\ & + \sin(\Delta\omega_{i,k}kT + (p_{i,k-1} - p_{i,k})) \end{aligned} \quad (8.3.8)$$

and

$$\begin{aligned} d_{i,k} \approx & -\cos(2\omega_{i,k}kT + p_{i,k-1} + p_{i,k} + 2\phi(\omega_{i,k})) \\ & + \cos(\Delta\omega_{i,k}kT + (p_{i,k-1} - p_{i,k})) \end{aligned} \quad (8.3.9)$$

The signals $c_{i,k}$ and $d_{i,k}$ are then fed through identical low pass filters to remove the components at approximately $2\omega_{i,k}$ to give,

$$g_{i,k} \approx \theta_{i,k} \sin(\Delta\omega_{i,k}kT + (p_{i,k-1} - p_{i,k})) \quad (8.3.10)$$

$$h_{i,k} \approx \theta_{i,k} \cos(\Delta\omega_{i,k}kT + (p_{i,k-1} - p_{i,k})) \quad (8.3.11)$$

The signals $\tan^{-1}(\frac{g_{i,k}}{h_{i,k}}) \approx \Delta\omega_{i,k}kT + (p_{i,k-1} - p_{i,k})$ are then used as correction signals to control three digital phase-locked loops whose outputs at time $t = kT$ are $e^{j(\theta_{i,k} \sin(\omega_{i,k}kT + p_{i,k}))}$, $i = 1, 3, 5$, where

$$\theta_{i,k} = \sqrt{g_{i,k}^2 + h_{i,k}^2}. \quad \text{If a delay in detection is used in the main}$$

carrier phase tracking loop, then the above method will estimate the phase jitter present on the sample r_{k-n} rather than r_k . Since the jitter

processes are assumed to be sinusoidal, and if it is assumed that θ_i and p_i change fairly slowly with time, then a good approximation to the jitter present on the sample r_k is given by $e^{j(\theta_{i,k-n} \sin(\omega_{i,k-n}kT + p_{i,k-n}))}$,

where n is the detection delay. The three outputs are then multiplied together, along with the estimate of any phase or frequency offset produced by the main carrier phase loop, and used to correct the received samples $\{r_i\}$. Alternatively, the estimates of the angles from the four processes may be added and the correction signal determined from $\cos(\theta_{T,i}) + j\sin(\theta_{T,i})$ where $\theta_{T,i}$ is the total phase correction required on the i^{th} received sample. Finally, to reduce the effects of performance degradation when no phase jitter is present, the jitter estimators are instructed to ignore the bandpass filter outputs until pre-determined levels are exceeded.

The system described above has been modelled and simulated on a digital computer by S.W.Cheung⁽²⁴⁾, during a study financed by CASE plc of Watford, Herts. Initial results have been extremely encouraging, as shown in Figure 8.3.8, which illustrates the performance of the system under

conditions of high signal-to-noise ratio, a frequency offset of 10 Hz phase offsets of 45° and 90° and phase jitter at 50 Hz, 150 Hz and 250 Hz, each with a peak value of 17° . Comparison with Figures 8.3.1 to 8.3.6 shows a marked improvement in performance of the complete phase tracking system, where the jitter components have been reduced rapidly leaving the main loop to track the phase and frequency offsets.

8.4 Timing Recovery

Timing recovery or more precisely, element timing synchronisation refers to the need for the receiver to remain in synchronism with the transmitted data symbols⁽²²⁾. As in the case of carrier synchronisation, timing synchronisation must take into account both the frequency and phase of the timing signal used to sample the received data. Due to the proposed digital nature of the receiver, it follows that the receiver's timing signal not only determines the rate and phase of the signal used to control the ADC (at the required digital sampling rate) but also determines the samples, at the symbol rate, used in all of the subsequent receiver processes and the control of all store-and-shift operations used in these processes.

For the modem considered here, the transmitter sends data symbols at 3200 bauds. As the input binary data arrives at (nominally) 19,200 bit/s, the transmitter's baud rate clock will operate at one-sixth of the incoming bit-rate. To allow for small deviations in the input bit-rate, the transmitter's master oscillator should in practice be phase-locked to either the binary data itself or preferably, to a 19,200 Hz clock signal generated by the source equipment. The receiver unit will also use a master oscillator from which all digital sampling processes and timing signals are derived.

Clearly, any difference in frequency between the transmitter and receiver baud rate clocks will result in the two units becoming out of synchronism. One effect of such a timing frequency offset is that the estimate of the sampled impulse response of the channel experiences a drift through the taps of the adaptive channel estimator, the amount of drift being dependent on the actual value of the timing offset. If no attempt is made to correct the offset, the sampled impulse response can effectively be "shifted-out" of the estimator which will obviously lead to erroneous detection. This movement of the estimated sampled impulse response can in fact be used to derive a correction signal to synchronise the receiver clock with the symbol rate of the received data signal, as will be shown later.

Timing phase refers to the actual instant in time that the received signal is sampled when it is assumed that the sampling frequency remains constant and no timing offset exists between the transmitted data rate and the receiver's timing waveform. As mentioned in Chapter 5, because the sample rate (symbol rate) here is slightly above the Nyquist rate of the baseband channel, the linear pre-filter used ahead of the detector effectively corrects for any departure of the sampling phase from the ideal value. Tests performed on telephone circuits 1-3 with filter sets 1 and 2 have confirmed this by using Program 3 in Appendix 10 to sample the baseband channels at different sample phases. After phase equalisation using Program 4, the resulting sampled impulse responses were found to be identical. It follows that for the system under consideration here, the phase of the sampling instants will not critically affect the tolerance of the system to additive Gaussian noise, so long as the sampling phase remains constant. The problem here is therefore to control timing frequency rather than frequency and phase. As it is reasonable to assume that the receiver has prior knowledge of the nominal data rate used at the transmitter, it follows that in practice,

when stable oscillators are used at both ends of the system, only fairly small timing frequency corrections will be necessary.

In general, two main methods of timing synchronisation are possible^(22,28). The first method relies on the transmitter sending a separate timing signal which can then be extracted by the receiver. Although conceptually simple, this method has major disadvantages for use in a 19,200 bit/s modem. Firstly, because an additional signal must be transmitted, extra complexity will be involved at both transmitter and receiver. Secondly, when operating under bandwidth and average power constraints, the presence of the timing signal will necessarily mean a reduction in tolerance to additive noise. Thirdly, and most importantly for the case of high-speed systems, the timing signal will have to be positioned towards the edge of the available channel bandwidth where group-delay distortion can be severe. When used with detectors which are highly sensitive to timing phase, the variation of group delay at the band edges of different telephone channels will make optimum operation impossible.

The second method avoids the above problems by deriving the timing signal from the received data signal itself. The additional complexity is now limited to the receiver unit, where the majority of the processing power resides. The following methods of timing recovery operate on this basic principle.

If it can be assumed that the sampled impulse response of the channel itself varies very slowly with time compared with the symbol rate, then Figure 8.4.1 illustrates a fairly simple method of obtaining a suitable correction signal to drive a phase-locked timing loop^(22,23,27). The figure shows the coefficients of the sampled impulse response as held in the taps of the channel estimator. If a timing offset exists between the transmitter and receiver baud rate timing signals, the coefficient

values will appear to drift towards the last tap when the receiver clock frequency is greater than the symbol rate of the received data signal or towards the first tap if the receiver's timing signal has a lower value than that at the transmitter. If the absolute value of a particular tap coefficient is held in store and compared with the sample value held in the same tap some time later, then a difference in these values will indicate the presence of a timing frequency offset. The absolute values are used here to remove any sensitivity to carrier phase errors.

Unfortunately, due to the shape of the sampled impulse response, this difference signal will not be able to determine the direction of the shift. However, by comparing the absolute values of two taps equally spaced either side of the main tap, directional information can be obtained. Extending this further, the magnitude of the timing frequency offset can be determined by monitoring many tap values and observing the rate of movement through the estimator. This has the additional advantage of not confusing a timing frequency offset with changes in the sampled impulse response due to other factors. Also, due to the fairly small timing offsets likely to be met in practice, the timing loop may be operated with a much longer averaging period than the channel estimator's integration time, which further reduces the possibility of confusion. This simple system may also be used to recover timing information when operating over channels which vary fairly rapidly with time by including the modifications of Clark and McVerry^(23,27).

The operation of the timing loop described above has assumed correct initialisation of the modem receiver and so it will be working in its steady-state tracking mode. Mention has already been made of the need for modem training in order to set-up the stored vectors and cost functions within the detector as well as to ensure convergence of the adaptive channel

estimator. Although the modem receiver can be instructed to expect the arrival of a training sequence (followed by actual data) after a pre-determined time has elapsed following receipt of some control tone, the receiver may actually start sampling the received data just before or just after the first symbol reaches the receiver. The result of this is that the tap values of the estimated channel impulse response, which are adjusted by assuming that the received samples correspond to the known symbols used in the training sequence, may become pushed to one end of the estimator. To enable the use of an estimator with relatively few taps, it is therefore necessary to establish the correct correspondance between the received samples and the symbols actually transmitted in the training sequence. This problem, known as frame synchronisation⁽²²⁾, can be solved by initially transmitting a periodic training sequence, where of course the period length, the content of the sequence and the number of repetitions used are all known by the receiver. If the received sample at the output of the post-demodulation filter is given as r_i , then under normal operating conditions, the detected data symbol will be $s'_{i-\ell-n}$, where ℓ is the delay in the linear pre-filter and n is the detection delay. During training, $s'_{i-\ell-n}$ will be replaced by a known symbol value which in general, will not necessarily be the correct value due to uncertainty in the receiver's timing. In fact, during frame synchronisation and the initial setting of the estimator's taps, the delay terms need not be included since the system is assumed to know the training sequence data. Consequently, at time $t = iT$, the receiver should select the symbol s_i but will probably select the symbol $s_{i-\mu}$, where μ is a small positive or negative integer. Provided the length of the periodic training sequence is the same as the number of taps used in the channel estimator, the effect of μ will be to force a cyclic shift of the tap values within the estimator itself, the shift being equal

to μ . Frame synchronisation can be achieved by observing the estimator's tap values and rotating the taps within the estimator until the main lobe appears in a pre-determined range of tap positions. A non-cyclic sequence can then be used to finely-tune the estimator and to set-up the detector's vector stores. In practice, frame synchronisation is achieved by observing the two end taps as well as the main tap position such that the direction of movement involving the minimum number of shifts may be determined. Once initiated, the estimator and timing adjustment systems operate in the decision-feedback mode.

An alternative technique for the recovery of timing information is shown in Figure 8.4.2. The method uses a digital version of the squaring-loop technique^(22,28,29,33) and has the advantage over the method proposed earlier in that it operates independently of the channel estimator and hence of the detected data sequence $\{s_i^d\}$. It is assumed in Figure 8.4.2 that the ADC operates at a digital sampling frequency of 16 kHz which is derived from a stable, but controllable master oscillator. The choice of digital sampling frequency is considered in more detail in Chapter 9.

The spectrum of the samples at the output of the squaring function will contain a line component at the symbol rate of the received and filtered signal which can be recovered by a narrow bandpass filter centred approximately around 3200 Hz⁽³³⁾. This filter can be realised as a digital filter operating at the 16 kHz digital sampling frequency. Consequently, the output from the filter is as shown in Figure 8.4.3, where 5 samples are spread over the duration of one signal element. Essentially, the squaring circuit is acting as a (square-law) approximation to a rectifying non-linear device and, as it acts on the modulated signal, the recovered timing signal after filtering may contain some amplitude modulation ripple⁽³³⁾. During training and timing acquisition, a strong spectral component at the symbol

rate can be achieved by ensuring that different symbols are transmitted, rather than the same point in the signal constellation. Clearly, in the latter case, the transmitted signal becomes a continuous carrier containing no timing information.

From Figure 8.4.3, the receiver timing waveform will be locked in frequency to the transmitter's baud rate clock if the sample values always occur at the same points on the recovered timing waveform. If the receiver clock is slightly low in frequency, then a given point, for example, point A_i in Figure 8.4.3, will slowly move down the sine wave. Similarly, if the receiver's timing signal is slightly high in frequency, then point A_i will move towards the left.

Consider the points A_i and B_i in Figure 8.4.3. These points represent two sample values of the timing waveform for the i^{th} timing waveform cycle and have been selected to lie either side of the zero mean value. When the receiver timing is correctly set, then A_i/B_i will be a negative constant for all values of i . However, if the frequency of the receiver's timing waveform is too high, then $|A_i/B_i|$ will gradually increase with increasing values of i as the sample values move towards the left of the sinusoidal waveform. If, at any time, samples A_i and B_i appear in the same half cycle, that is, they are either both positive or both negative values, then A_i/B_i becomes positive. Under this condition, the next adjacent pair of samples that differ in sign are chosen for the process. By comparing the ratio A_{i-1}/B_{i-1} with A_i/B_i , the amount of timing frequency error can be determined and hence the frequency of the receiver's timing oscillator can be adjusted accordingly. The following algorithm, based on a suggestion by Dr. F. McVerry (Rockwell International, USA) appears to give a simple solution to the receiver timing problem, although due to its late development has yet to be proved as a practical method for this application.

The difference between two adjacent ratios is given by,

$$D_i = \frac{A_i}{B_i} - \frac{A_{i-1}}{B_{i-1}}$$

$$= \frac{B_{i-1}A_i - B_iA_{i-1}}{B_iB_{i-1}} \quad (8.4.1)$$

If $D_i > 1$, then a control signal must be issued to the receiver oscillator so as to reduce the frequency of the timing signal. Due to the small values of offset likely to be encountered, even during the start-up period, the denominator of equation (8.4.1) acts effectively as a normalising term and can be ignored to simplify the equation, hence,

$$D_i = B_{i-1}A_i - B_iA_{i-1} \quad (8.4.2)$$

Finally, to prevent excessive timing frequency jitter, the actual signal used to control (or "pull") the VCO is an integrated version of equation (8.4.2) such as

$$P_i = P_{i-1} + kD_i \quad (8.4.3)$$

where P_i is the pulling signal and k is a small positive constant. From the discussion of carrier tracking loops earlier, it should be clear that k controls the loop noise immunity and convergence rate.

When the timing loop is closed, as shown in Figure 8.4.2, the system attempts to adjust the (nominal) 16 kHz digital sampling frequency to be 5 times the symbol rate of the incoming received signal. Figure 8.4.4 shows a possible realisation for the digital bandpass loop filter⁽³⁰⁾. The complex conjugate pole-pair on the circle of radius r give the basic bandpass amplitude characteristic whilst the zeros at $z = \pm 1$ provide low and high frequency rejection. The bandpass characteristic is centred

around $f_0 = 3200$ Hz and so from the Figure 8.4.4,

$$\begin{aligned}\theta &= \omega_0 \tau \\ &= 2\pi f_0 \tau \\ &= 2\pi \frac{f_0}{f_s} \text{ rads}\end{aligned}\tag{8.4.4}$$

where f_s is the digital sampling rate and τ is the sample period. The distance of the poles from the origin of the z -plane, r , controls the bandwidth of the filter; for this application $r \approx 0.95$.

From Figure 8.4.4, the transfer function of the filter can be written as,

$$H(z) = \frac{(z+1)(z-1)}{(z-re^{j\omega_0\tau})(z-re^{-j\omega_0\tau})}\tag{8.4.5}$$

$$H(z) = \frac{1-z^2}{1-2r\cos\omega_0\tau z^{-1}+r^2 z^{-2}}\tag{8.4.6}$$

from which the recurrence equation may be derived,

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1-z^2}{1-2r\cos\omega_0\tau z^{-1}+r^2 z^{-2}}\tag{8.4.7}$$

$$Y(z) = X(z) - X(z)z^{-2} + 2r\cos\omega_0\tau Y(z)z^{-1} - r^2 Y(z)z^{-2}\tag{8.4.8}$$

hence,

$$y_i = x_i - x_{i-2} + 2r\cos\omega_0\tau y_{i-1} - r^2 y_{i-2}\tag{8.4.9}$$

Figure 8.4.5 shows the filter given by equation (8.4.9) in block diagram form where $c_1 = -r^2$ and $c_2 = 2r\cos\omega_0\tau$.

At $f = f_0$,

$$|H(z)| = \frac{2(1 + \cos^2 \theta)^{\frac{1}{2}}}{(1-r) [(1+r)^2 \sin^2 \theta + (1-r)^2 \cos^2 \theta]^{\frac{1}{2}}} \quad (8.4.10)$$

and c_3 in Figure 8.4.5 is equal to

$$\frac{1}{|H(z = e^{j\omega_0 \tau})|}$$

so that $|H(z = e^{j\omega_0 \tau})| = 1$

8.5 Automatic Gain Control

Automatic Gain Control (AGC) is required to counter the attenuation introduced by the transmission path. More precisely, AGC attempts to maintain a given received signal level at the input to the ADC when operating over different telephone circuits (where the actual attenuation over the pass-band is unknown prior to transmission and may cover a range of about 40 dB⁽²⁸⁾) and when the attenuation introduced by a channel varies with time (Chapter 3). Clearly, the AGC system must view the received signal over a fairly long time period and hence counteract any long term variations in the received signal level.

The use of digital signal processing techniques in the receiver obviously requires an ADC within the system. Since in this design, the only analogue component in the receiver is the channel output bandpass filter, it follows that conversion must be performed prior to demodulation. In order to reduce the quantisation noise introduced by the ADC (typically of 8-12 bits accuracy), the level of the analogue input must be adjusted to make full use of the available input signal range.

Although several AGC systems are available which operate directly on the received samples, $\{r_i\}$, to determine a control signal which then adjusts the range control of the ADC or alternatively controls the gain of an AGC amplifier, these systems, known in general as "power" AGC systems, can give a poor performance when operating on signals derived from large QAM structures⁽²²⁾.

An alternative method of AGC and the one adopted here, utilises information available in the form of the estimated sampled impulse response of the baseband channel before the pre-filter. Figure 8.5.1 shows the general arrangement of the AGC system where the signal level to the input of the ADC is controlled by a variable gain amplifier. It is interesting to note that the switched capacitor filters considered in Chapter 5 contain amplifiers which can provide around 20 dB of gain.

Figure 8.5.2 shows the AGC system in more detail. At time $t = iT$, the received and demodulated sample, r_i , is used to generate an estimate of the sampled impulse response of the channel, y'_{i-h} , along with the detected data symbols s'_{i-h} , s'_{i-h-1} , ..., s'_{i-h-g} , where h is the delay introduced by the pre-filter and the detection process itself. Assuming that the channel estimator has $(g+1)$ taps, then the length of vector y'_{i-h} is given by

$$|y'_{i-h}| = \left(\sum_{j=0}^g |y'_{i-h,j}|^2 \right)^{1/2} \quad (8.5.1)$$

then since the $\{y'_{i-h,j}\}$ are derived from the received samples, $\{r_i\}$, a knowledge of $|y'_{i-h}|$ gives an indication to the level of the received signal at time $t = iT$. If the level of the received signal changes, then so too will the level of the $\{r_i\}$, causing a corresponding change in $|y'_{i-h}|$. Consequently, if the value of $|y'_{i-h}|$, such that the input signal to the ADC occupies its full range, is known in advance and is given as ϵ ,

then the difference between ϵ and $|y'_{i-h}|$ can be used to alter the gain of the AGC amplifier. In general, however, ϵ will not be known due to the wide spread of attenuation experienced over different telephone circuits and a more appropriate control signal is given by $(1 - |y'_{i-h}|)$. This signal is then integrated over a period controlled by k in Figure 8.5.2 to give the gain update equation as

$$\text{Gain}_i = \text{Gain}_{i-1} + k(1 - |y'_{i-h}|) \quad (8.5.2)$$

where the relationship between the amplifier's control signal, Gain_i and the actual gain produced by the amplifier depends on the input range of the ADC, and where the value of k determines the convergence time and the noise immunity of the AGC system.

Results of computer simulation studies performed on the above AGC system are shown in Figures 8.5.3 to 8.5.8; the simulation program is included here as Program 9 in Appendix 10. In all of the tests, the channel estimator was allowed to converge to its steady-state condition before application of a known change of level to the received samples $\{r_i\}$. The difference between $|y'_i|$ and the known length of the sampled impulse response vector, $|y_i|$ was then computed and used in the evaluation of the gain parameter according to,

$$\text{Gain}_i = \text{Gain}_{i-1} + k(|y_i| - |y'_i|) \quad (8.5.3)$$

The AGC loop was then closed by using Gain_i as a multiplying coefficient for the $\{r_i\}$. In the adjustment of the adaptive estimator, the detected symbols $\{s'_i\}$, were taken to be equal to the transmitted symbols $\{s_i\}$, and a delay of zero assumed. Also for all tests, the integration parameter, "a" used in the adaptive estimator was set at 0.0005, the value recommended in Chapter 7.

Figure 8.5.3 shows the transient response of the AGC system for various values of k after the application of a 6 dB change in signal level. The signal-to-noise ratio used here is defined in Appendix 3 and was fixed at 31 dB for this test. The sampled impulse response of telephone circuit 3 with filter set 2 was used in all tests considered in the section. It is clear from the figure that a value of $k = 0.1$ will give rapid convergence of the AGC system, with very little overshoot, in about 250 symbol periods. The effect of noise on the system can be seen in Figure 8.5.4 which shows the variation of level error with time, where,

$$\text{level error, dB} = 20\log_{10}(x/z) \quad (8.5.4)$$

where, x is the amount of signal level change introduced

and z is the estimate of this change from the AGC system.

The insensitivity to noise illustrated in the figure can be attributed to the tolerance to noise of the adaptive channel estimator (Figures 7.5.1-7.5.3), but it should be recalled that the present series of tests assume ideal detection regardless of signal-to-noise ratio.

The variation of RMS level error with k and with signal-to-noise ratio is shown in Figures 8.5.5 and 8.5.6, respectively, where RMS level error is defined as,

$$\text{RMS level error, dB} = 20\log_{10}\left(\frac{1}{(n_2 - n_1)} \sum_{i=n_1}^{n_2} (x - z_i)^2\right)^{1/2} \quad (8.5.5)$$

where n_1, n_2 are the number of symbols concerned in the test $n_2 > n_1$, and z_i is the estimate of x produced by the AGC system at the i^{th} symbol period. For both figures, $n_2 = 4000$ $n_1 = 3000$, which, for the case of $k = 0.001$, is insufficient to ensure convergence, as implied by curve a, in Figure 8.5.3. For this reason, the part of Figure 8.5.5 around

$k = 0.001$ is shown with a broken line to indicate the uncertainty in the results. The results of Figures 8.5.3 and 8.5.5 suggest that a good compromise between the convergence time of the transient response and the steady-state level error exists when $k \approx 0.1$, this giving an RMS level error of around -37 dB at a signal-to-noise ratio of 36 dB.

Clearly, the convergence time can be reduced somewhat by increasing k but at the expense of an increased RMS level error. Figure 8.5.7 shows the effect of a time varying value for k , where,

$$k_{i+1} = k_i - \Delta(k_i - k_f) \quad (8.5.6)$$

where Δ controls the rate of decay, k_f is the final value and k_0 is the initial value of k . As should be expected from the above analysis, a reduction of k , over about the first 60 symbol periods, increases the convergence period very slightly but is accompanied by a reduction in the steady-state error.

Finally, Figure 8.5.8 shows the transient response of the system to an initial level error of 20 dB, as might be expected to occur during call initiation over switched telephone circuits. Clearly, the convergence rate of the system is much slower than the case of a 6 dB level change when used with $k_0 = 0.1$, $k_f = 0.05$, $\Delta = 0.15$, as suggested by Figure 8.5.7.

However, an increase in convergence rate can be achieved by reducing the rate of decay to $\Delta = 0.01$, as shown by curve b in Figure 8.5.8. To enable the system to converge fairly rapidly during initialisation, it is recommended that Δ is set to 0.01 during the training cycle of the receiver, whilst any small level changes should be tracked with $\Delta = 0.15$.

8.6 References

1. Lindsey, W.C., "Phase-Shift Keyed Signal Detection with Noisy Reference Signals", IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-2, No.4, pp. 393-400, July 1966.
2. Natali, F.D. and Walbesser, W.J., "Phase-Locked-Loop Detection of Binary PSK Signals Utilising Decision Feedback", IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-5, No.1, pp. 83-90, Jan 1969.
3. Lindsey, W.C., "Hybrid Carrier and Modulation Tracking Loops", IEEE Trans. on Comms; Vol. COM-20, No.1, pp. 53-55, Feb 1972.
4. Cahn, C.R., "Phase Tracking and Demodulation with Delay", IEEE Trans. on Info. Theory, Vol. IT-20, No.1, pp. 50-58, Jan 1974.
5. D'Andrea, A.N. and Russa, F., "Noise Analysis of a PSK Carrier Recovery DPLL", IEEE Trans. on Comms, Vol. COM-31, No.2, pp. 190-199, Feb 1983.
6. Chiu, H.C. and Simpson, R.S., "Effect of Quadrature and Demodulation Phase Errors in a Quadrature PSK System", IEEE Trans. on Comms, pp. 945-948, August 1973.
7. Taylor, D.P. and Cheung, D., "The Effect of Carrier Phase Error on the Performance of a Duobinary Shaped QPSK Signal", IEEE Trans. on Comms, pp. 738-744, July 1977.

8. Matyas, R. and McLane, P.J., "Decision-Aided Tracking Loops for Channels with Phase Jitter and Intersymbol Interference", IEEE Trans. on Comms. Vol. COM-22, No.8, pp. 1014-1023, August 1971.
9. Falconer, D.D., "Analysis of a Gradient Algorithm for Simultaneous Passband Equalisation and Carrier Phase Recovery", Bell Sys. Tech.J., Vol. 55, No.4, pp. 409-429, April 1976.
10. Falconer, D.D., "Jointly Adaptive Equalisation and Carrier Recovery in Two-Dimensional Digital Communication Systems", Bell Sys. Tech. J., Vol. 55, No.3, pp. 317-334, March 1976.
11. Macchi, O. and Scharf, L.L., "A Dynamic Programming Algorithm for Phase Estimation and Data Decoding on Random Phase Channels", IEEE Trans. on Info. Theory, Vol. IT-27, No.5, pp. 581-595, Sept 1981.
12. Magee, F.R. Jnr., "Simultaneous Phase Tracking and Detection in Data Transmission over Noisy Dispersive Channels", IEEE Trans. on Comms. pp. 712-715, July 1977.
13. Simon, M.K. and Smith, J.G., "Carrier Synchronisation and Detection of QASK Signal Sets", IEEE Trans. on Comms, Vol. COM-22, No.2, pp. 98-106, Feb 1974.
14. Lindsey, W.C. and Simon, M.K., "Carrier Synchronisation and Detection of Polyphase Signals", IEEE Trans. on Comms, pp. 441-454, June 1972.

15. Lindsey, W.C. and Simon, M.K., "Data-Aided Carrier Tracking Loops", IEEE Trans. on Comm. Tech, Vol. COM-19, No. 2, pp. 157-167, April 1971.
16. Foschini, G.J. et.al., "On the Selection of a Two-Dimensional Signal Constellation in the Presence of Phase Jitter and Gaussian Noise", Bell Sys. Tech.J., Vol. 52, No.6, pp. 927-965, July-August 1973.
17. Ho, E.Y. and Spalding, D.A., "Data Transmission Performance in the Presence of Carrier Phase Jitter and Gaussian Noise", Bell Sys. Tech.J., pp. 1927-1931, Oct 1972.
18. Falconer, D.D., "Optimal Reception of Digital Data over the Gaussian Channel with Unknown Delay and Phase Jitter", IEEE Trans. on Info. Theory, Vol. IT-23, No.1, pp. 117-126, Jan 1977.
19. Cheung, R. and McLane, P.J., "Carrier Reference Error Sensitivity of a Viterbi Detector for PAM Data Transmission", IEEE Trans. on Comms. Vol. COM-30, No.2, pp. 410-414, Feb 1982.
20. Matsuo, Y.M. and Namiki, J., "Carrier Recovery Systems for Arbitrarily Mapped APK Signals", IEEE Trans. on Comms., Vol.COM-30, No.10, pp. 2385-2390, Oct 1982.
21. Mohammed, H.D., "Carrier Phase Predictors for High-Speed Data Transmission Systems", IEE Proc. F., Vol.130, No.2, pp. 173-184, March 1983.

22. Harvey, J.D., "Synchronisation of Synchronous Modems", SERC Report No. GR/A/1200.7, Dec 1980.
23. McVerry, F., "High Speed Data Transmission Over H.F. Radio Links", Ph.D. Thesis, Loughborough University of Technology, 1982.
24. Cheung, S.W. and Bateman, S.C., "Phase Jitter Tracking Loops in High-Speed Data Communication Systems", Final Report, Loughborough University of Technology (to be published).
25. Bingham, J.A., "Design of Carrier Recovery Loop Filters to Track Phase Jitter", IEEE Document No. CH1809-3/83/0000/048S, 1983.
26. Ungerboeck, G., "New Application for the Viterbi Algorithm: Carrier Phase Tracking in Synchronous Data-Transmission Systems", Rec.IEEE Nat. Telecom. Conf., San Diego, California, Dec 1974.
27. Clark, A.P. and McVerry, F., "Time Synchronisation of an H.F. Radio Modem", IEE Proc. F., Vol.129, No.6, pp. 403-410, Dec 1982.
28. Clark, A.P., "Principles of Digital Data Transmission", Pentech Press, 1976.
29. Lyon, D.L., "Timing Recovery in Synchronous Equalised Data Communication", IEEE Trans. Comms. Vol. 23, pp. 269-274, Feb 1975.
30. Lynn, P.A., "An Introduction to the Analysis and Processing of Signals", MacMillan Press Ltd., 1973.

31. Gardener, F.M., "Phaselock Techniques", Wiley & Sons, 2nd Edition 1979.
32. Kobayashi, H., "Simultaneous Adaptive Estimation and Decision Algorithm for Carrier Modulated Data Transmission Systems", IEEE Trans. on Comms. Tech., Vol. COM-19, No.3, pp. 268-280, June 1971.
33. O'Reilly, J.J., "Timing Extraction for Baseband Digital Transmission", Mathematical Topics in Telecommunications, Vol.2, Ed. K.W.Cattermole, J.J.O'Reilly, Ch.10, Pentech Press, 1984.

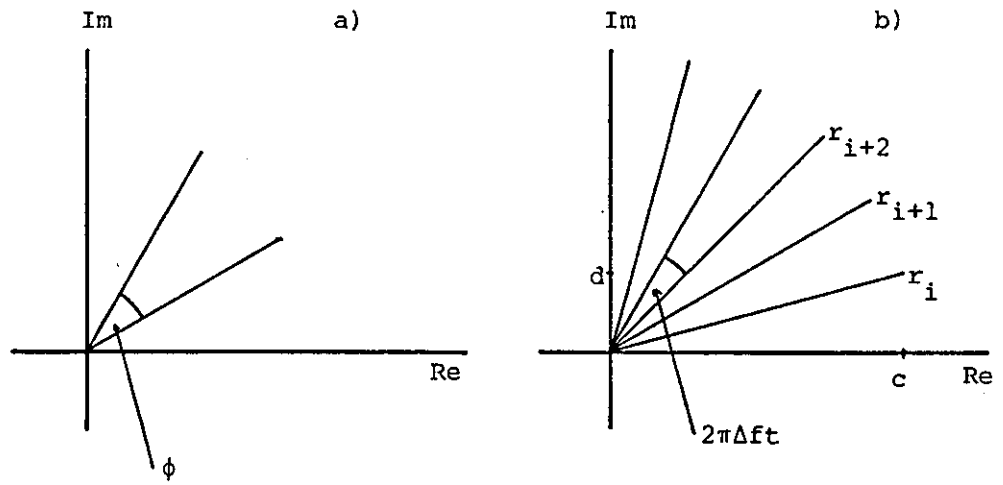


Figure 8.2.1: Rotation of Signal Vectors due to
 a) Constant Phase Error
 b) Frequency Offset of Δf Hz

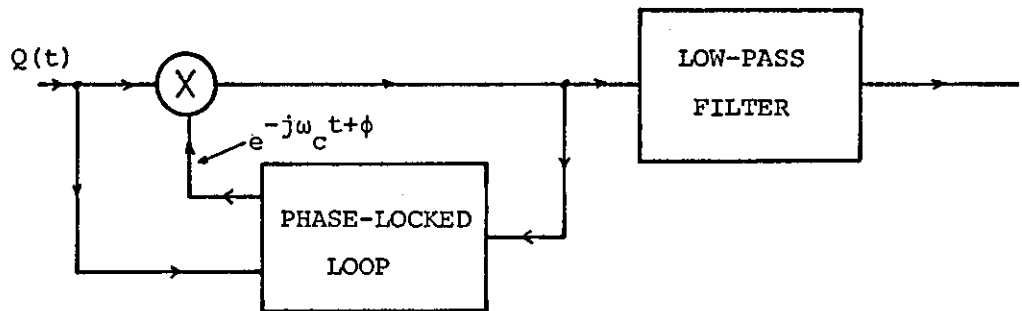


Figure 8.2.2: Phase-Locked Coherent Demodulation

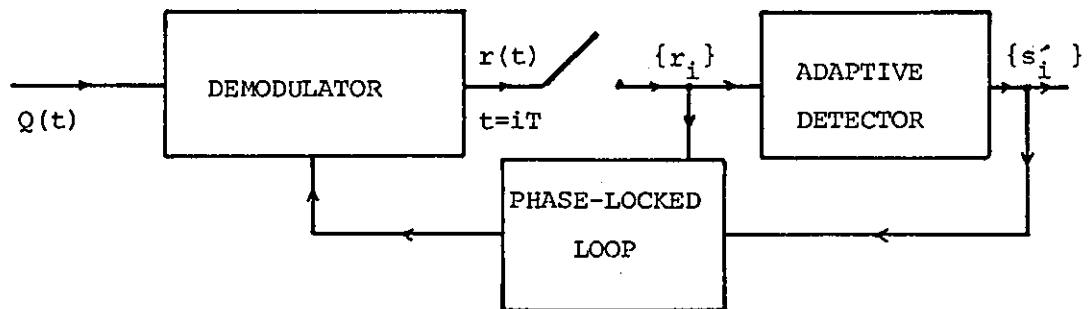


Figure 8.2.3: Data-Aided Carrier Tracking Loop

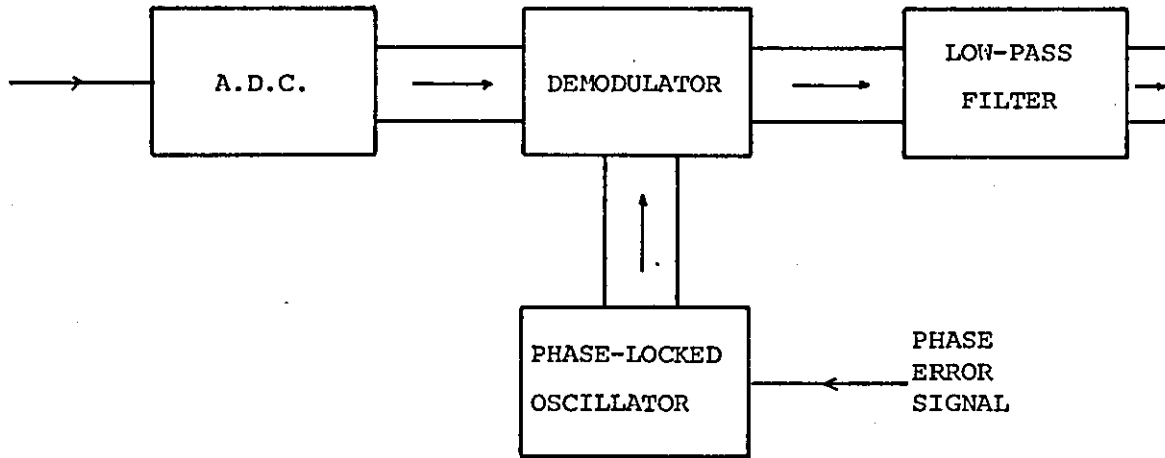


Figure 8.2.4: Digital Equivalent of Phase-Locked Coherent Demodulation

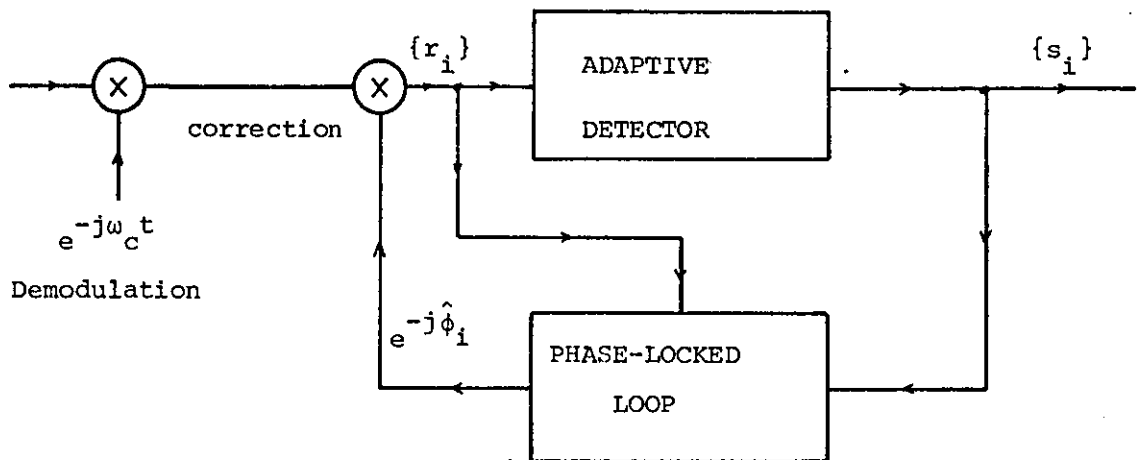


Figure 8.2.5: Demodulation and Phase Correction System

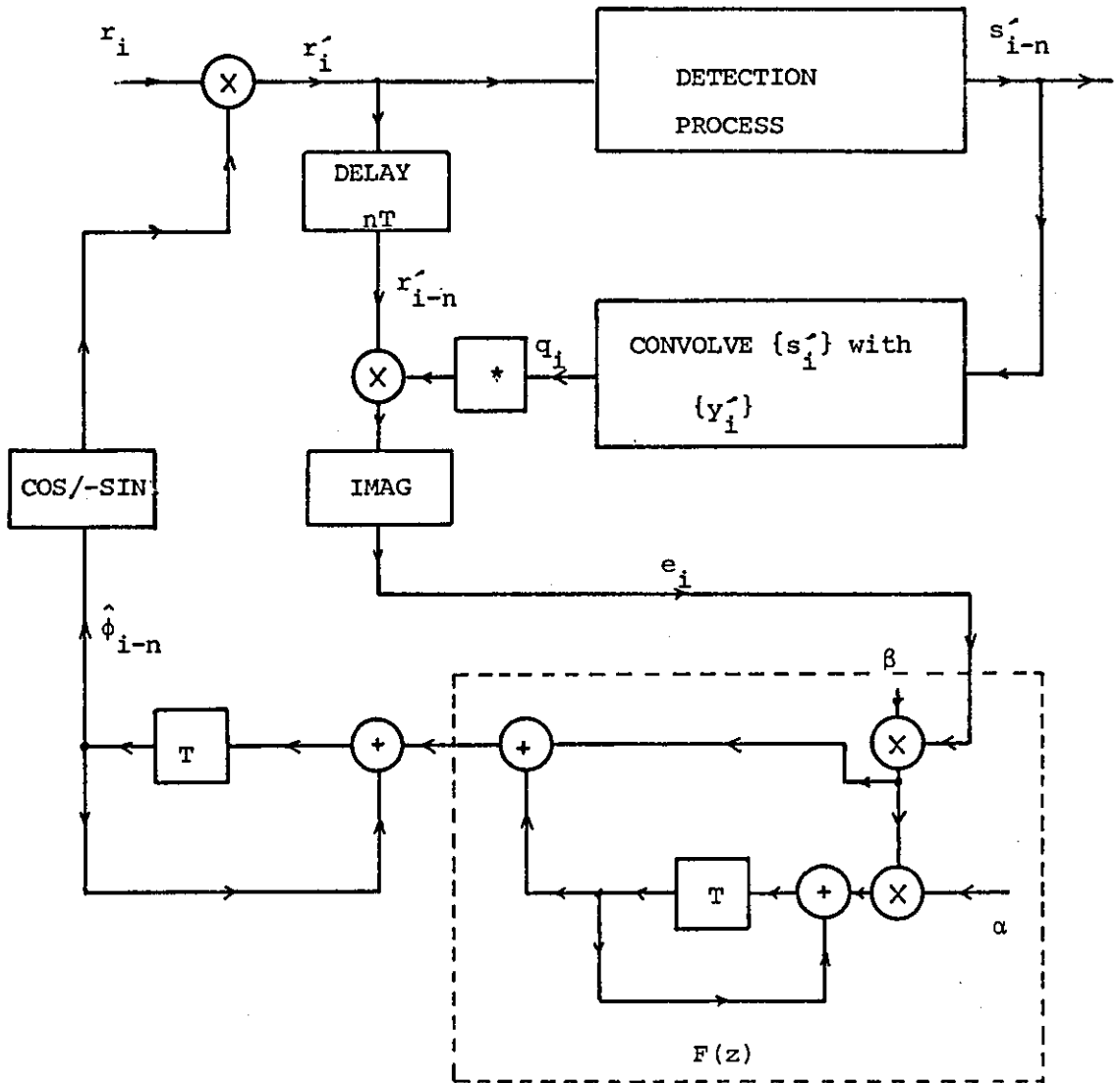


Figure 8.2.6: Carrier Loop 1

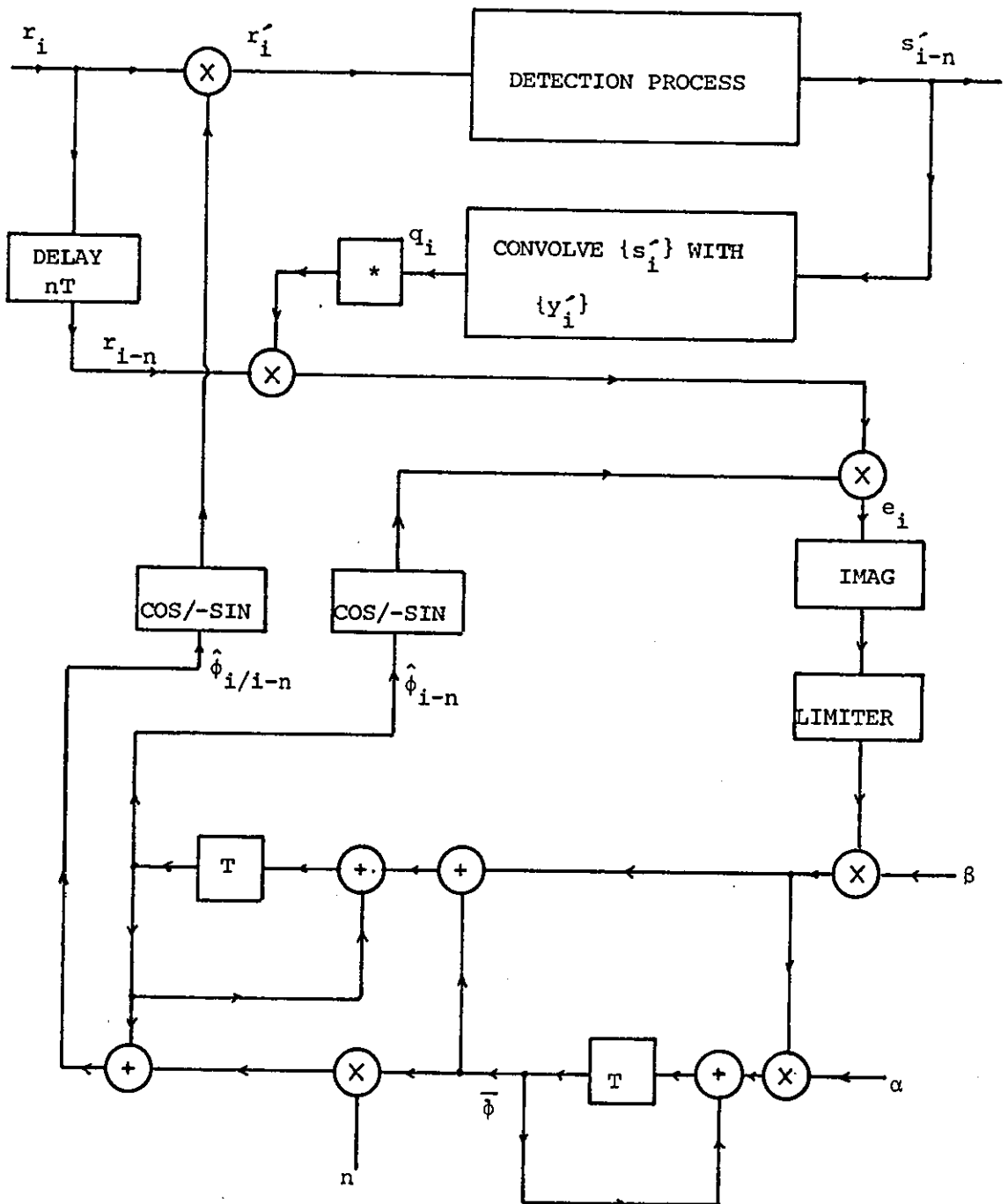


Figure 8.2.7: Carrier Loop 2

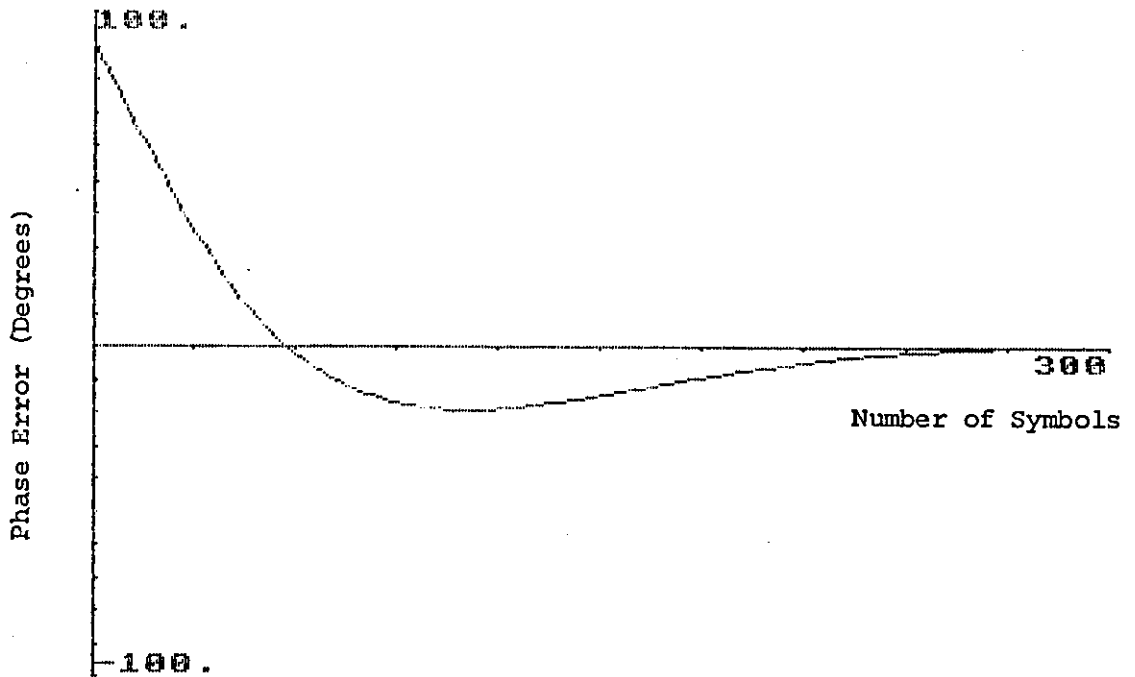


Figure 8.2.8: Carrier Loop 1

Baud rate = 3200
 Channel = Ideal
 $\alpha = 0.015$
 $\beta = 0.03$
 Phase Offset = 90°
 SNR = 80 dB
 Delay = 0

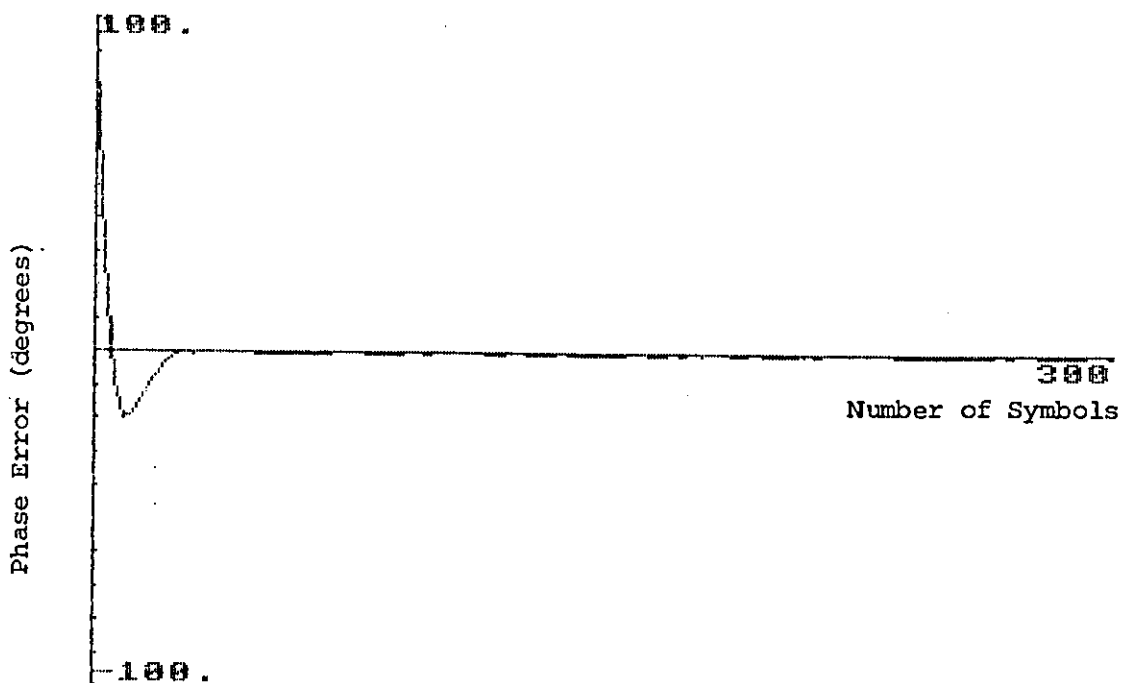


Figure 8.2.9: Carrier Loop 1

Baud rate = 3200
 Channel = Ideal
 $\alpha = 0.15$
 $\beta = 0.3$
 Phase Offset = 90°
 SNR = 80 dB
 Delay = 0

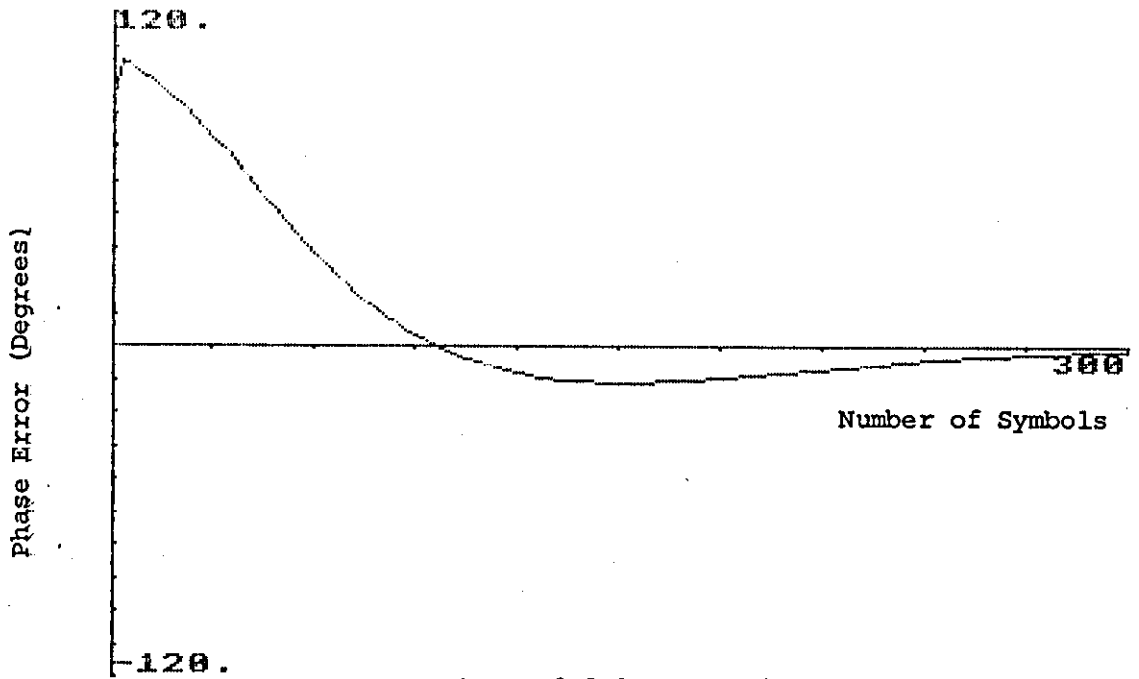


Figure 8.2.10: Carrier Loop 1

Baud Rate = 3200
 Channel = Ideal
 $\alpha = 0.015$
 $\beta = 0.03$

Phase Offset = 90°
 Freq. Offset = 10 Hz
 SNR = 80 dB
 Delay = 0

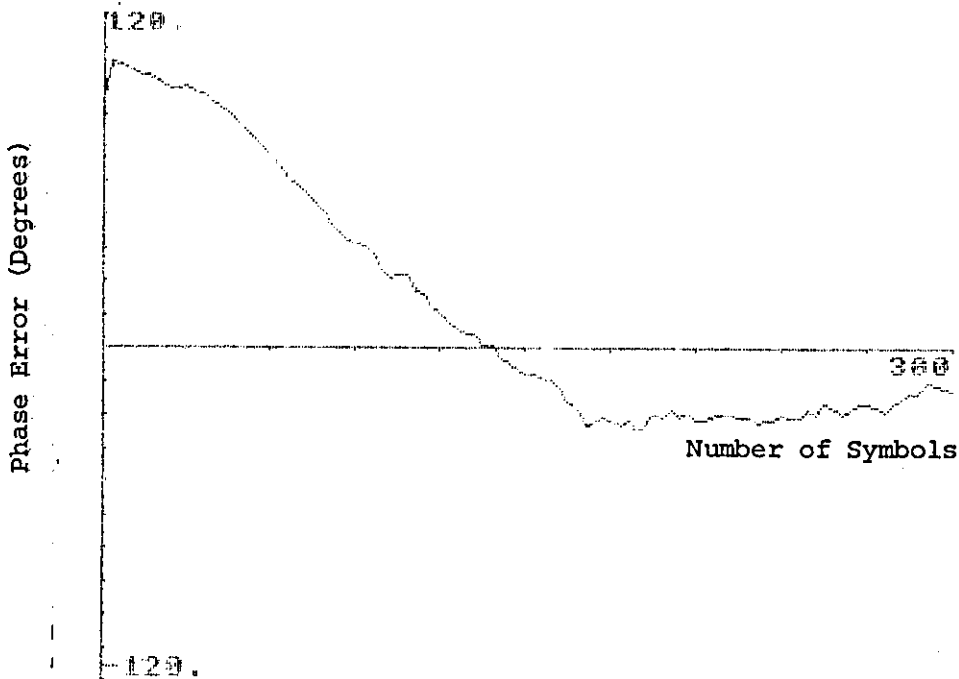


Figure 8.2.11: Carrier Loop 1

Baud Rate = 3200
 Channel = Ideal
 $\alpha = 0.015$
 $\beta = 0.03$

Phase Offset = 90°
 Freq. Offset = 10 Hz
 SNR = 80 dB
 Delay = 3T

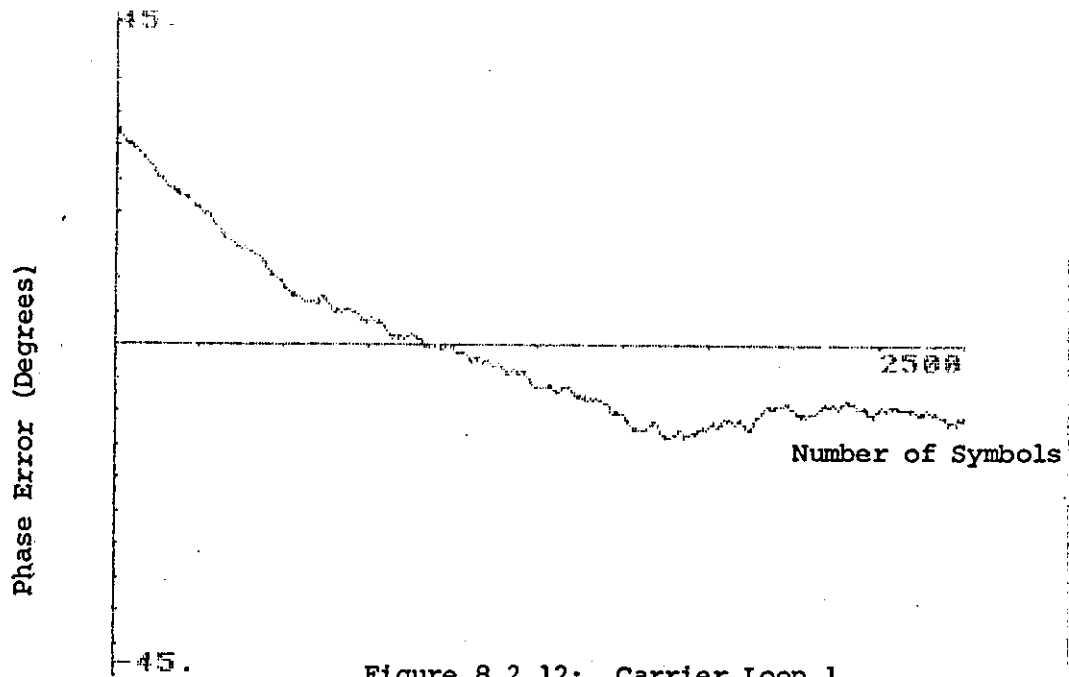


Figure 8.2.12: Carrier Loop 1

Baud Rate = 3200 Phase Offset = 30°
 Channel = Ideal SNR = 80 dB
 $\alpha = 0.0015$ Delay = $8T$
 $\beta = 0.003$

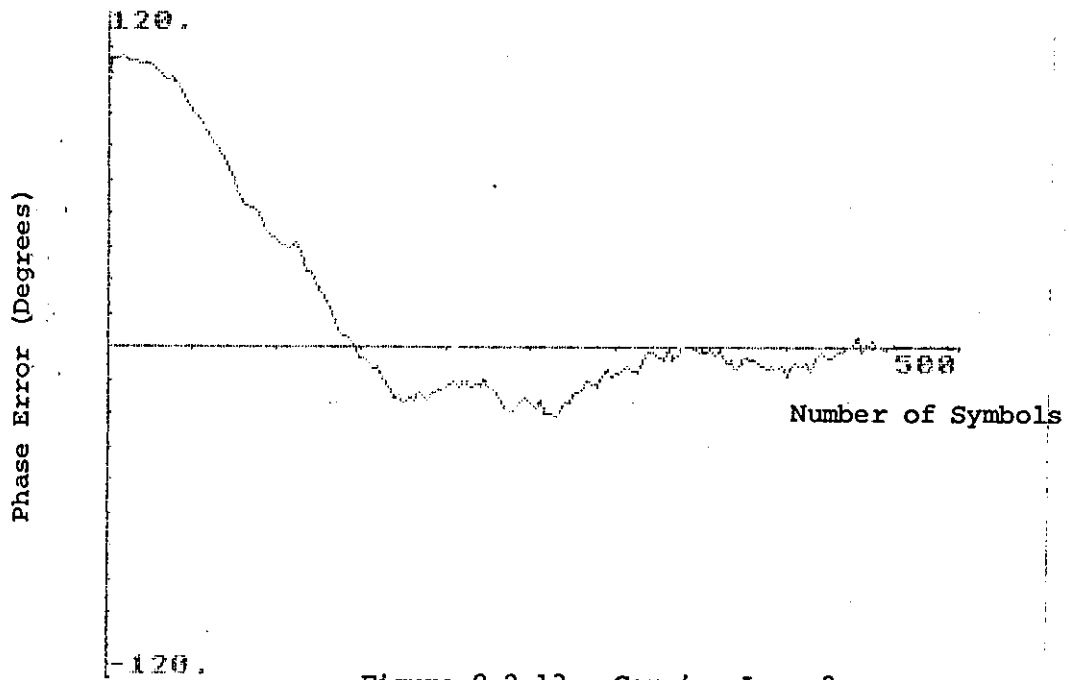


Figure 8.2.13: Carrier Loop 2

Baud Rate = 3200 Phase Offset = 90°
 Channel = Ideal Freq. Offset = 10 Hz
 $\alpha = 0.015$ SNR = 80 dB
 $\beta = 0.03$ Delay = $4T$

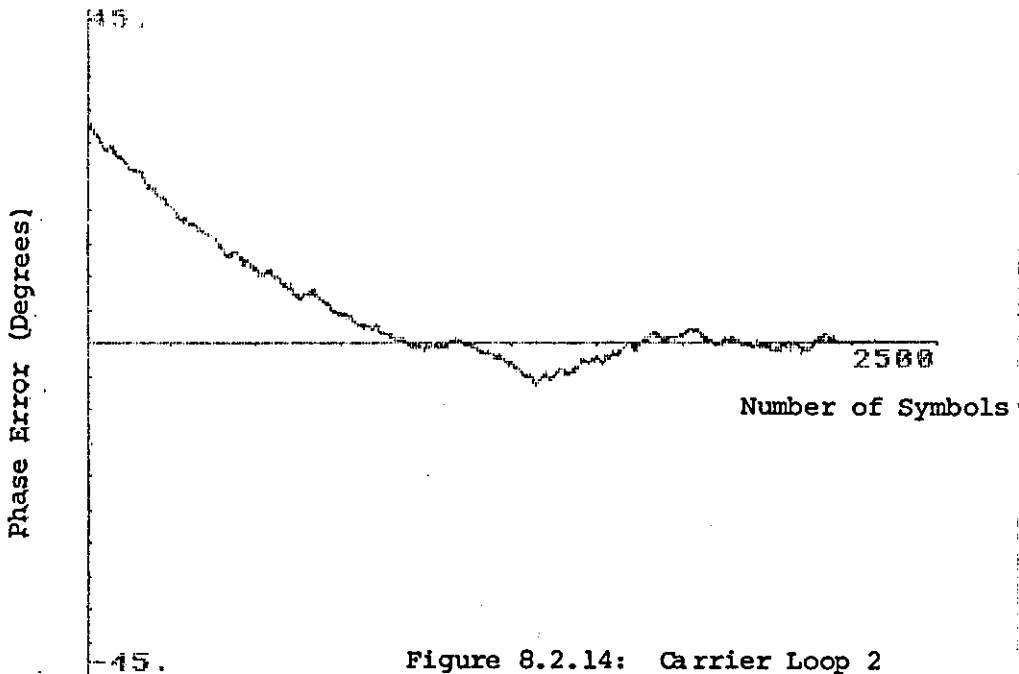


Figure 8.2.14: Carrier Loop 2

Baud Rate = 3200 Phase Offset = 30°
 Channel = Ideal SNR = 80 dB
 $\alpha = 0.0015$ Delay = $8T$
 $\beta = 0.003$

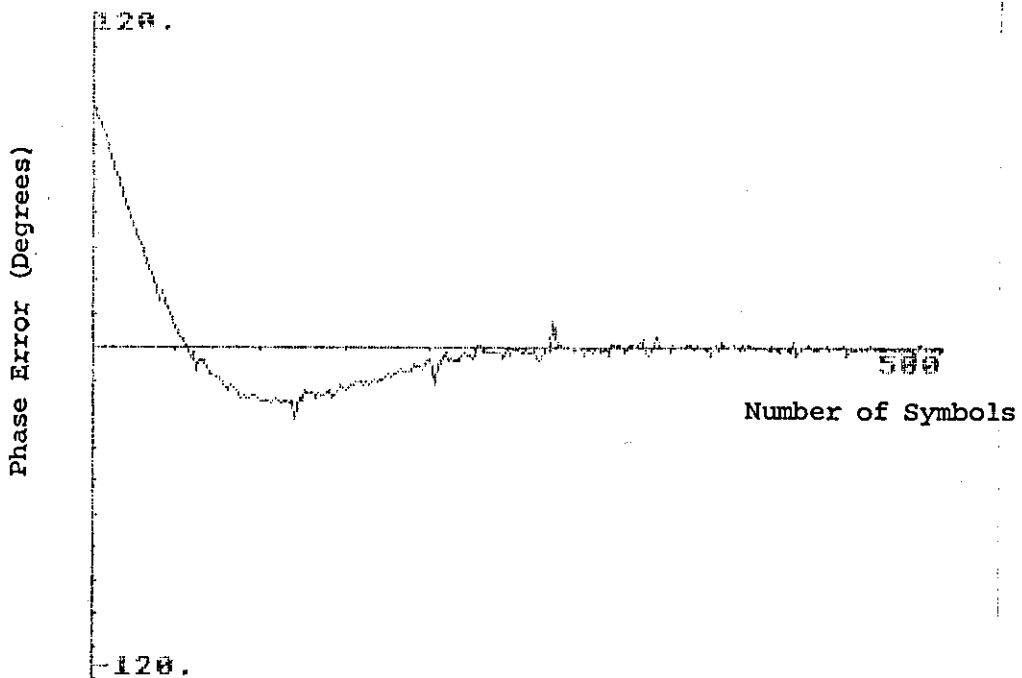
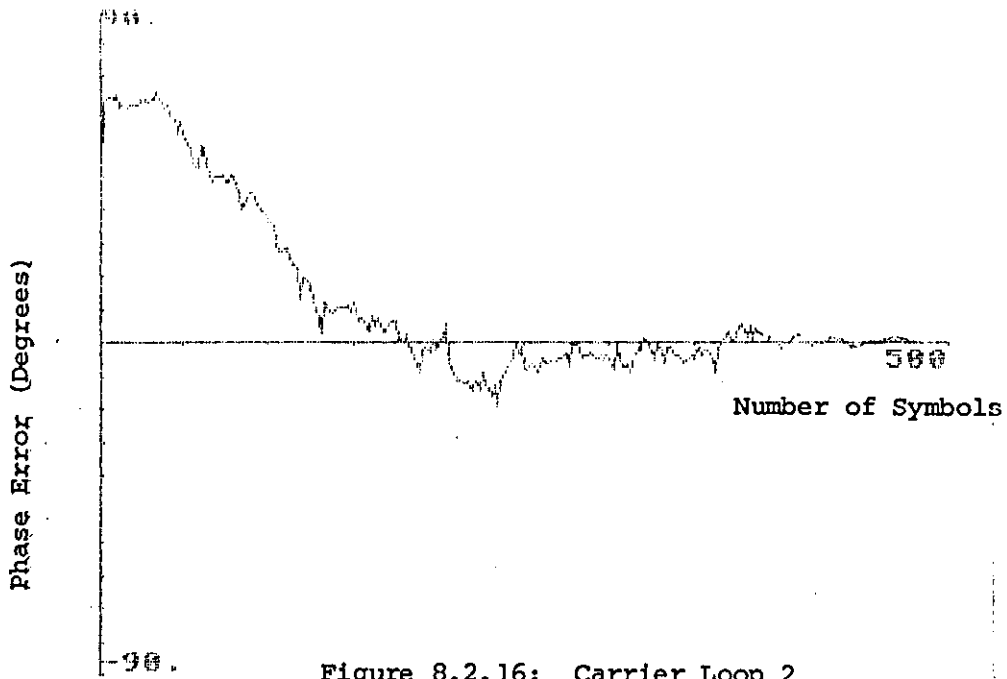


Figure 8.2.15: Carrier Loop 2

Baud Rate = 3200
 Channel = Circuit 3 + Filter 2
 $\alpha = 0.015$ $\beta = 0.03$
 Phase Offset = 90° SNR = 20 dB
 Delay = 0



```

Baud Rate = 3200          Phase Offset = 45°
Channel = Circuit 3 + Filter 2  Freq. Offset = 10 Hz
    α = 0.015              SNR = 20 dB
    β = 0.03               Delay = 4T

```

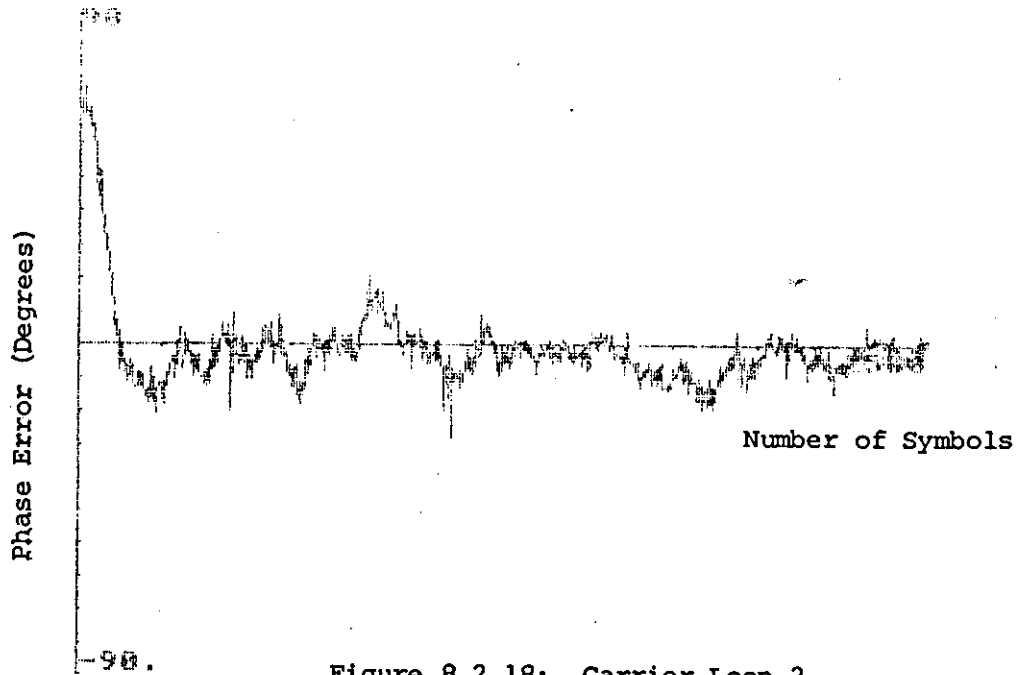


Figure 8.2.18: Carrier Loop 2

Baud Rate = 3200 Phase Offset = 45°
 Channel = Circuit 3 + Filter 2 Freq. Offset = 10 Hz
 $\alpha = 0.015$ SNR = 20 dB
 $\alpha_o = 0.0015$ Delay = 8T
 $\Delta = 0.0005$
 $\beta = 2\alpha$

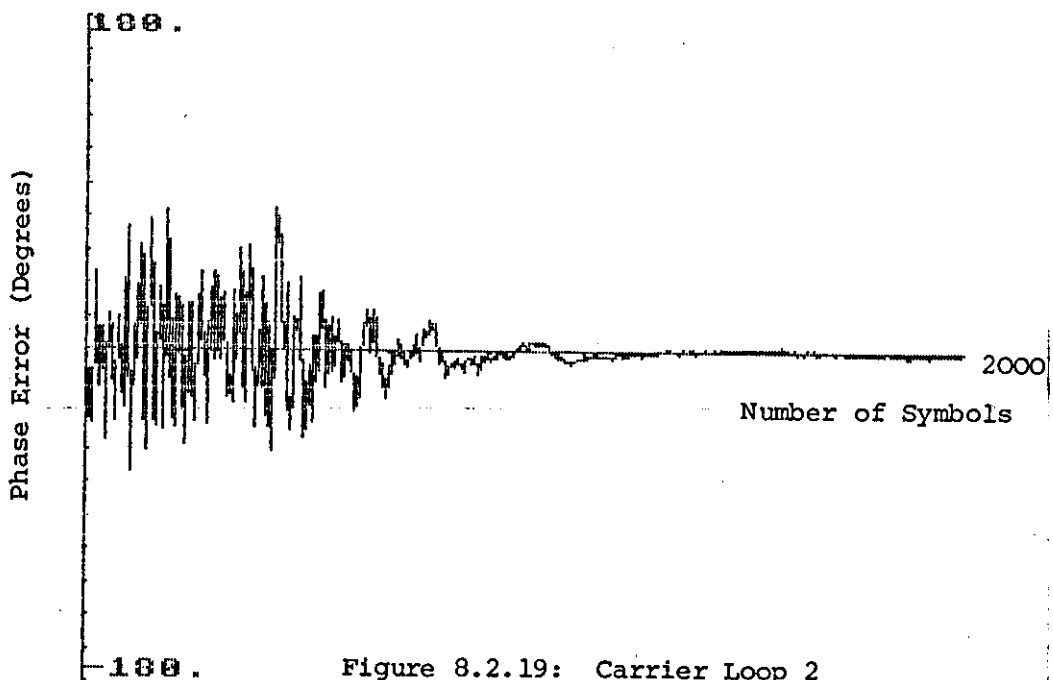


Figure 8.2.19: Carrier Loop 2

Baud Rate = 3200 Phase Offset = 90°
 Channel = Circuit 3 + Filter 2 Freq. Offset = 10 Hz
 $\alpha = 0.15$ SNR = 30 dB
 $\alpha_o = 0.0015$ Delay = 8T
 $\Delta = 0.005$
 $\beta = 2\alpha$

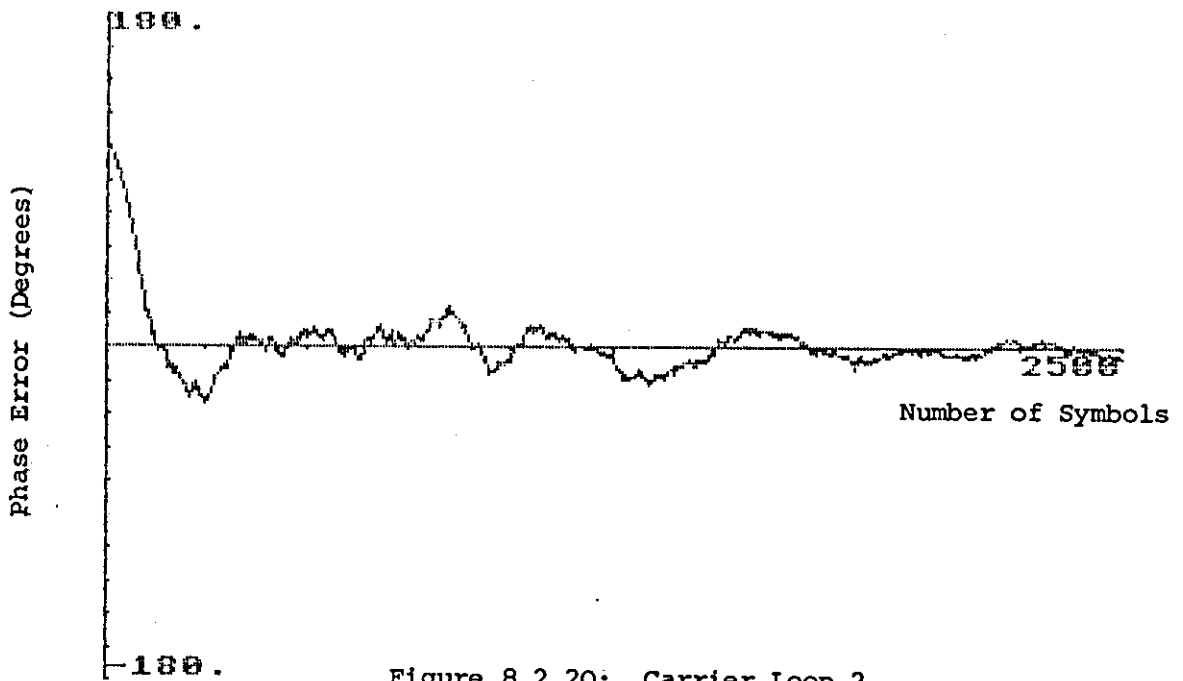


Figure 8.2.20: Carrier Loop 2

Baud Rate = 3200	Phase Offset = 90°
Channel = Circuit 3 + Filter 2	Freq. Offset = 10°
$\alpha_o = 0.015$	SNR = 30 dB
$\alpha_f = 0.00015$	Delay = $8T$
$\Delta = 0.0005$	
$\beta = 2\alpha$	

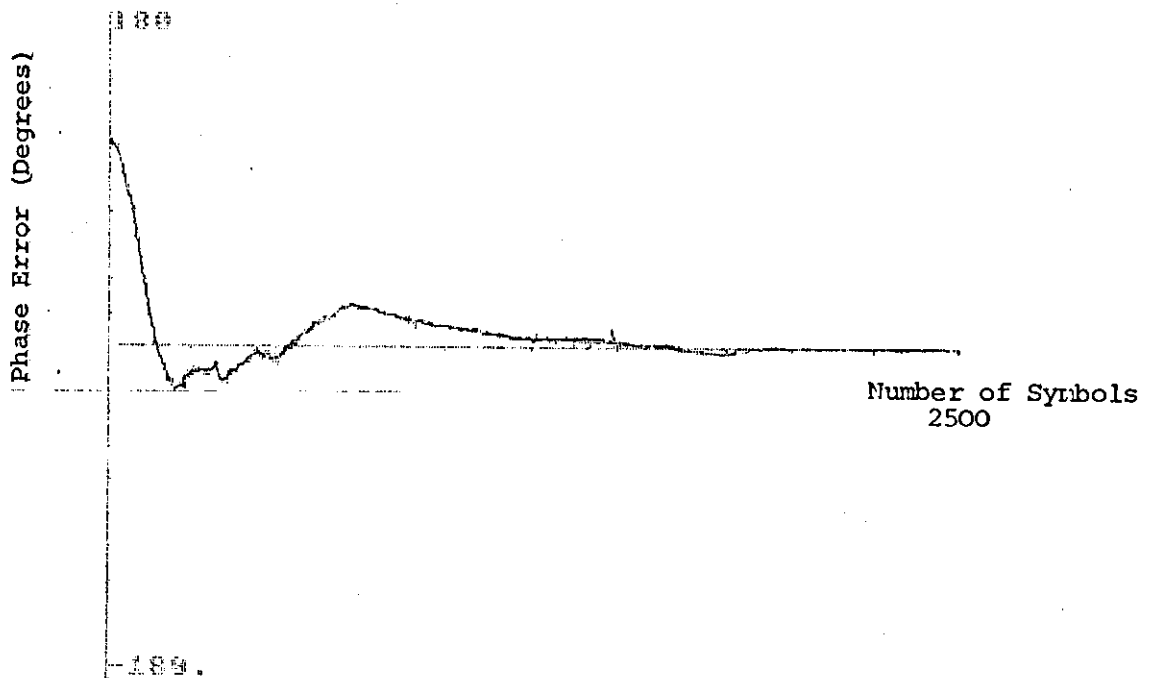


Figure 8.2.21: Carrier Loop 2

Baud Rate = 3200	Phase Offset = 90°
Channel = Circuit 3 + Filter 2	Freq. Offset = 10 Hz
$\alpha_o = 0.015$	SNR = 30 dB
$\alpha_f = 0.00015$	Delay = $8T$
$\Delta = 0.0005$	
$\beta = 2\alpha$	

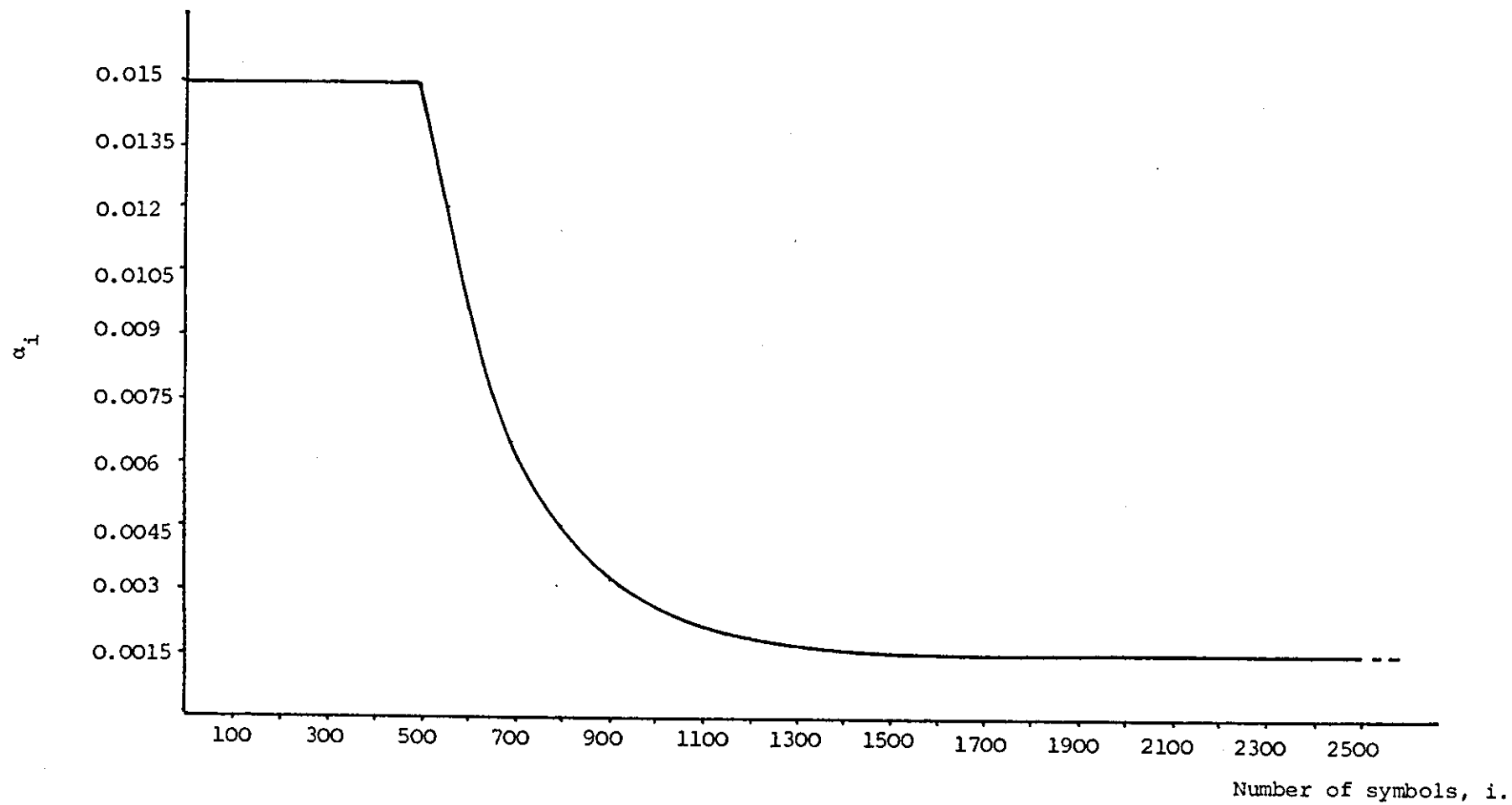


Figure 8.2.22: Variation of α_i with Number of Symbols,

$$\begin{aligned}
 \alpha_i &= \alpha_0, \quad i < 500 \\
 \alpha_i &= \alpha_{i-1} - \Delta(\alpha_{i-1} - \alpha_f), \quad i < 500, \\
 \beta_i &= 2\alpha_i, \quad \text{all } i
 \end{aligned}
 \qquad
 \begin{aligned}
 \alpha_0 &= 0.015 \\
 \alpha_f &= 0.0015 \\
 \Delta &= 0.005
 \end{aligned}$$

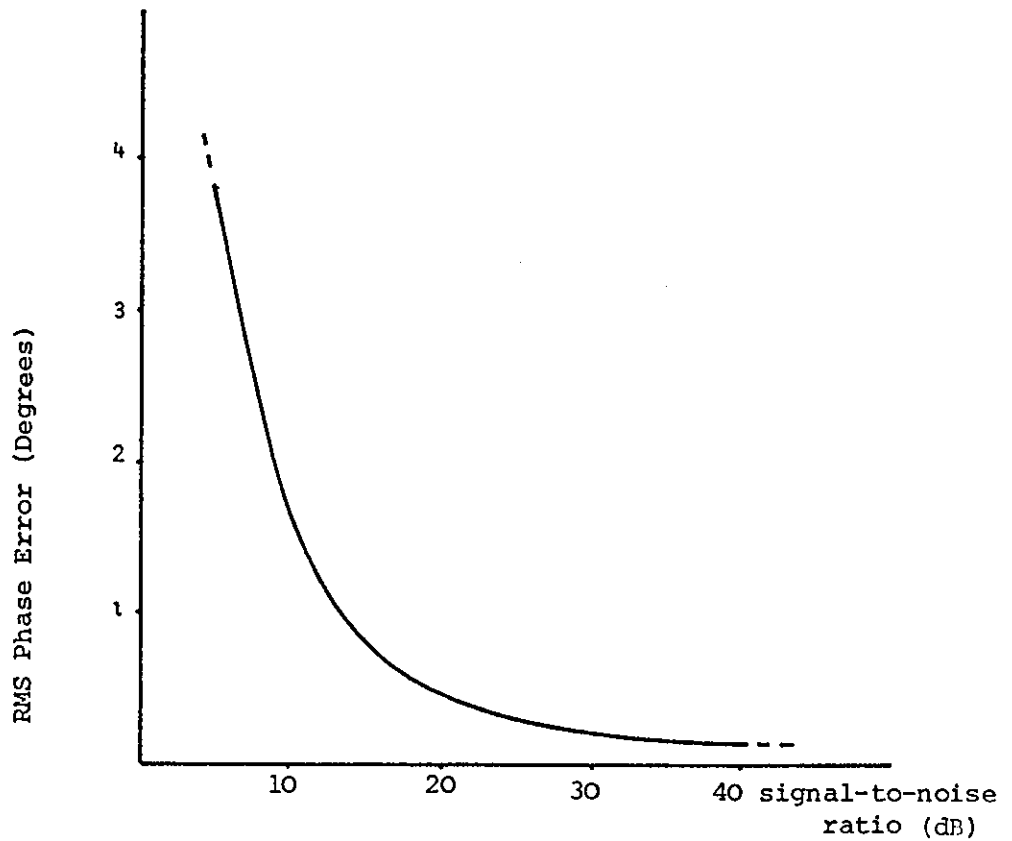


Figure 8.2.23: Variation of RMS Phase Error with Signal-to-Noise Ratio for Carrier Loop 2

Channel = Circuit 3 + Filter 2

$\alpha = 0.015$

$\alpha_F^0 = 0.0015$

$\Delta_F = 0.005$

$\Delta_B = 2\alpha$

Phase Offset = 90°

Freq. Offset = 10 Hz

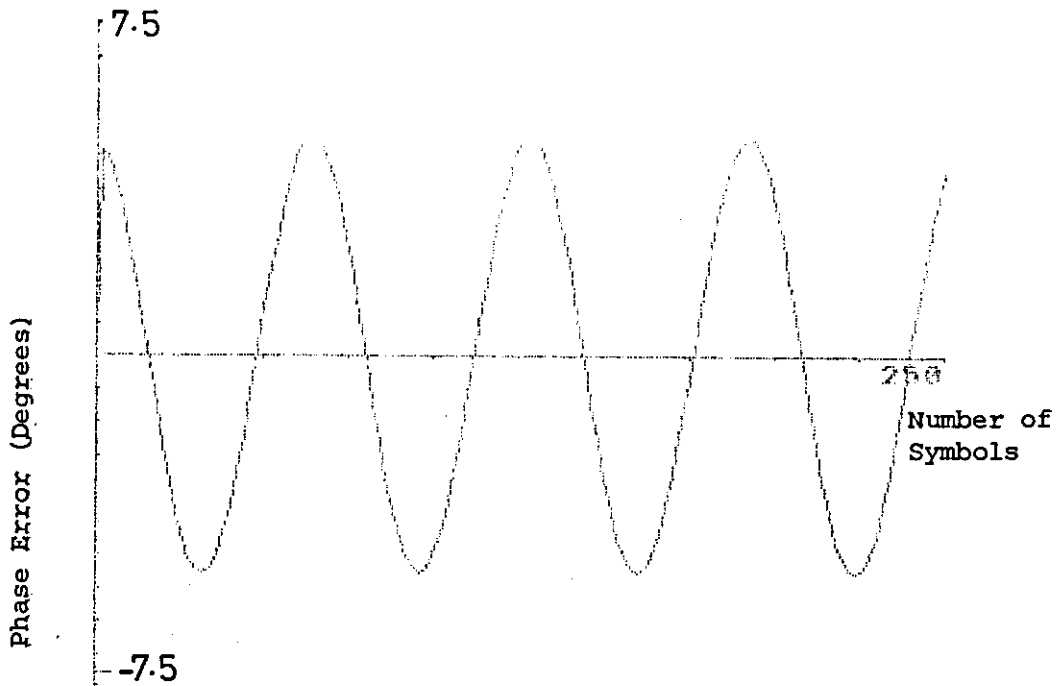


Figure 8.3.1: Carrier Loop 2

Baud Rate = 3200	Phase Jitter = 5° pk @ 50 Hz
Channel = Ideal	SNR = 80 dB
$\alpha = 0.015$	Delay = 0
$\beta = 0.03$	

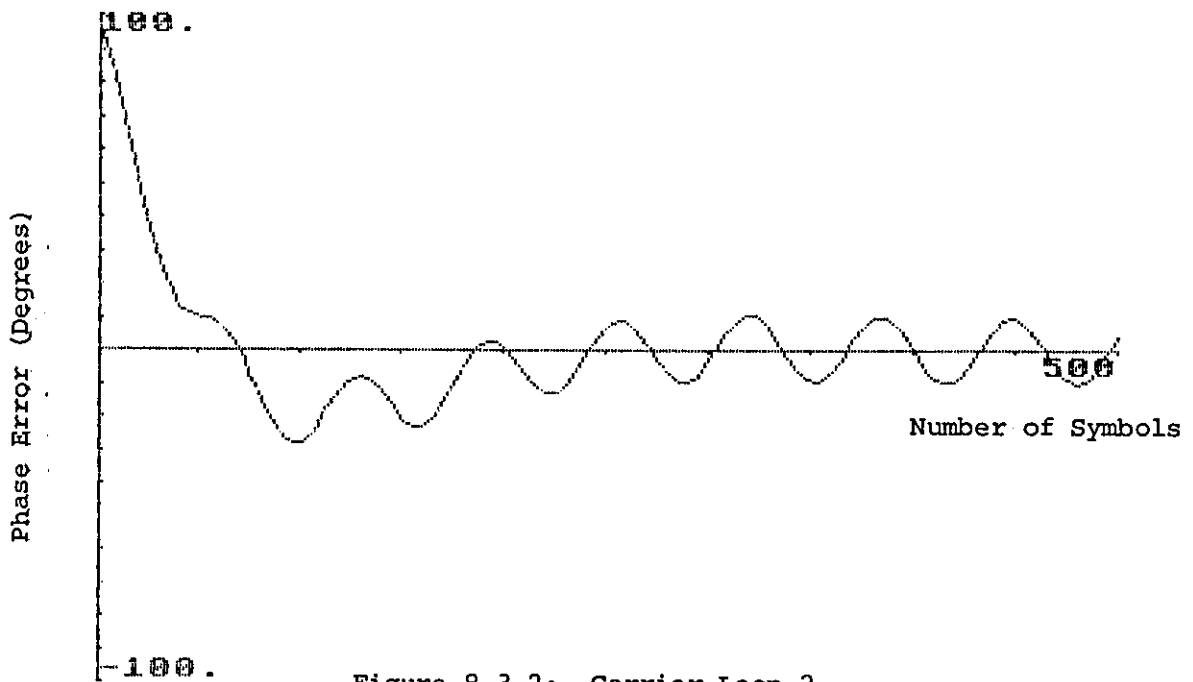
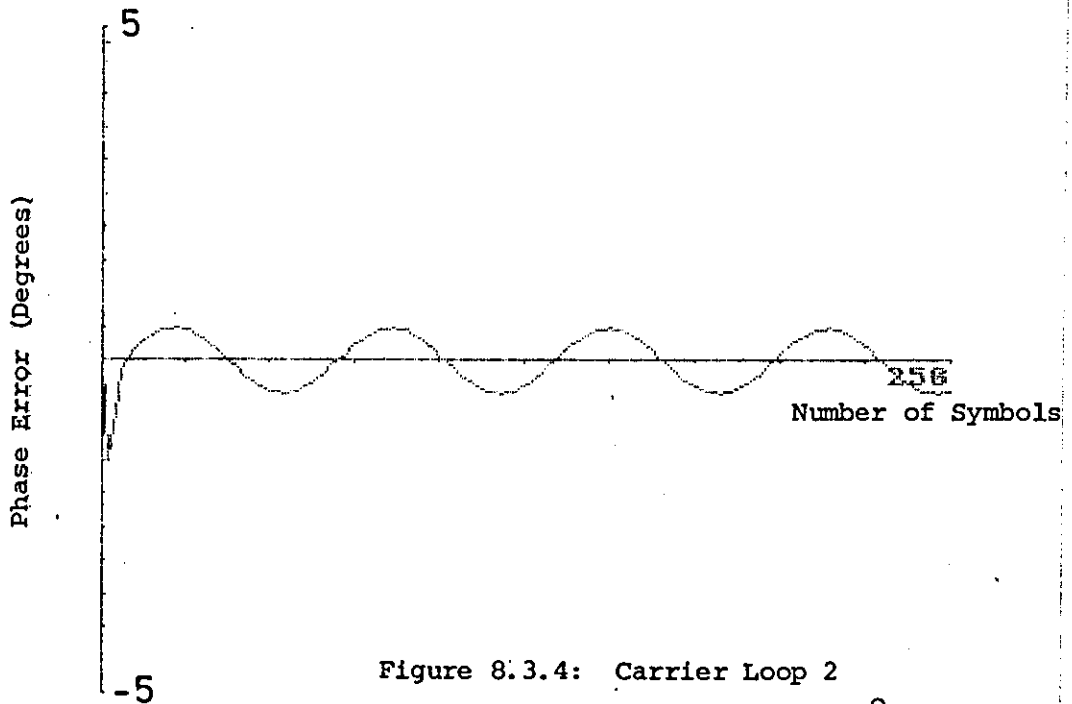
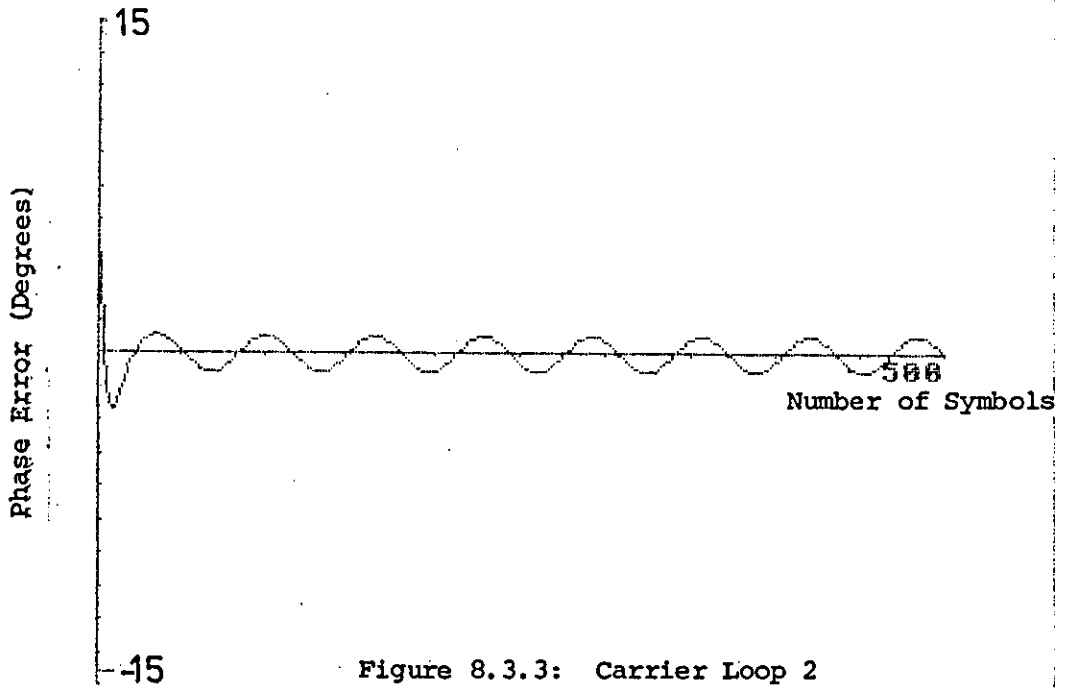


Figure 8.3.2: Carrier Loop 2

Baud Rate = 3200	Phase Offset = 90°
Channel = Ideal	Phase jitter = 10° pk @ 50 Hz
$\alpha = 0.015$	SNR = 80 dB
$\beta = 0.03$	Delay = 0



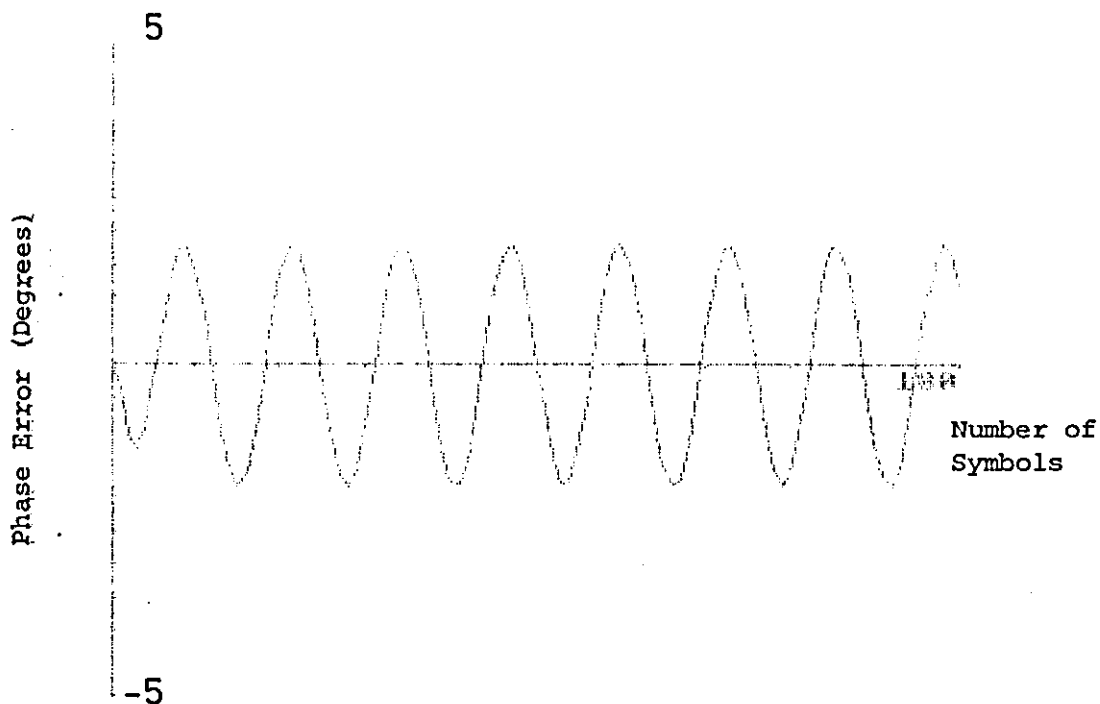


Figure 8.3.5: Carrier Loop 2

```

Baud Rate = 3200   Phase Jitter = 5° pk @ 250 Hz
Channel = Ideal      SNR = 80 dB
α = 0.4              Delay = 0
β = 0.8

```

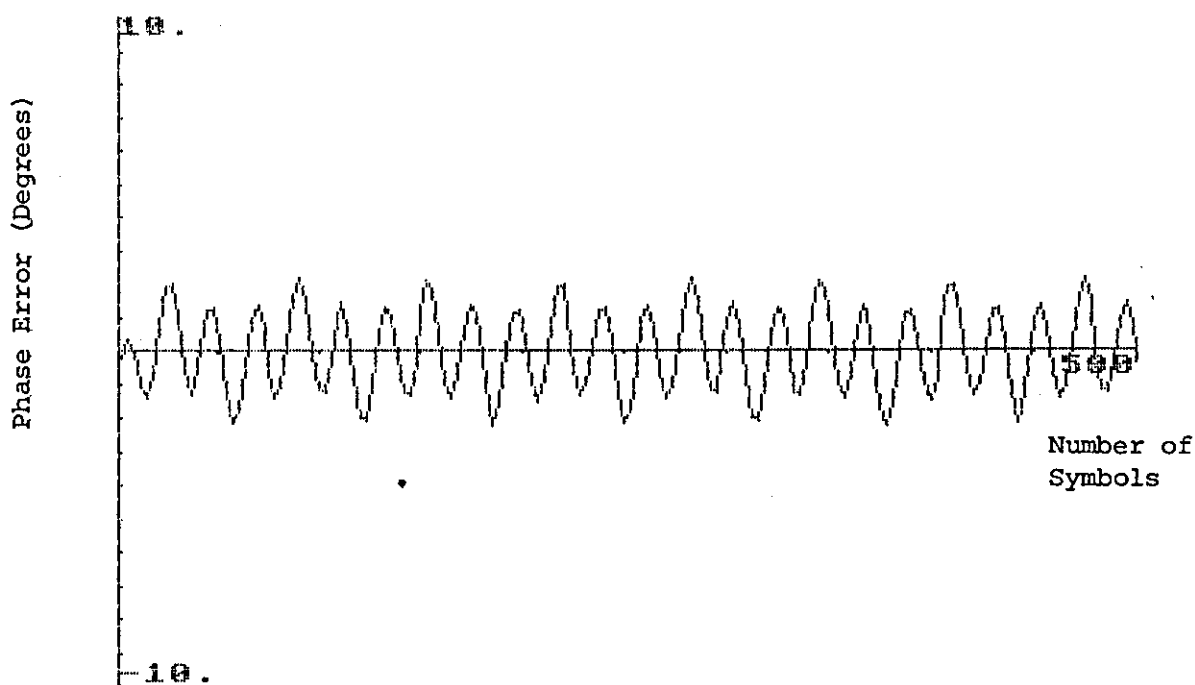


Figure 8.3.6: Carrier Loop 2

```

Baud Rate = 3200      Phase Jitter = 5opk @ 50 Hz
Channel = Ideal        5opk @ 150 Hz
α = 0.25              SNR = 80 dB
β = 0.5                Delay = 0

```

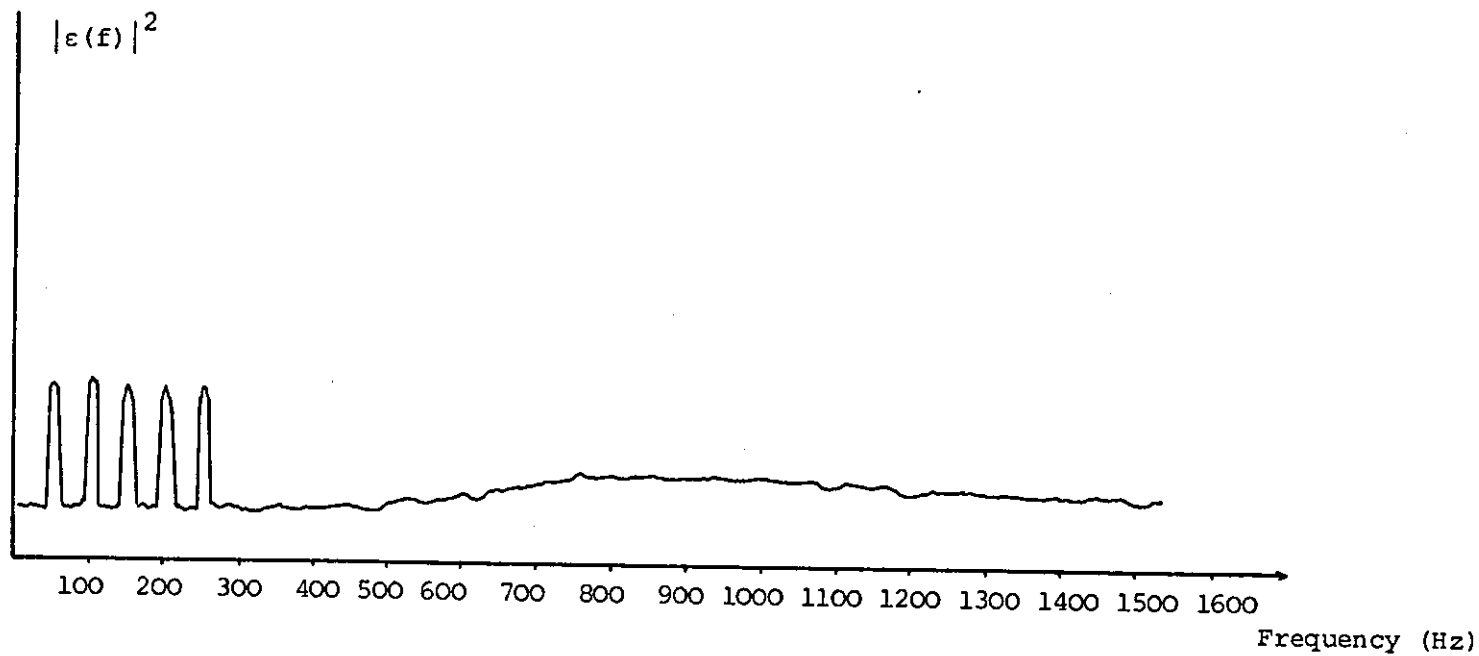
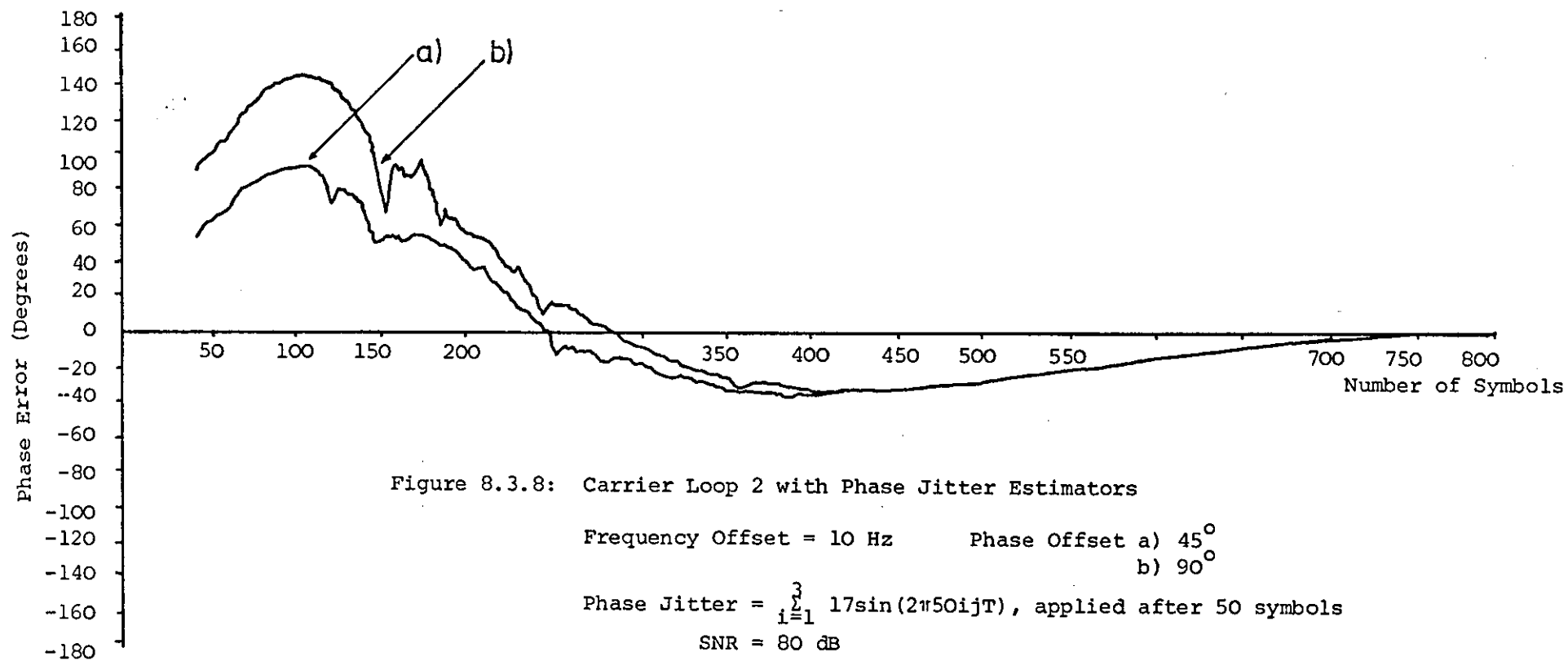


Figure 8.3.7: Amplitude Spectrum of the Error Signal in Carrier Loop 2

$$\text{Applied Phase Jitter} = \sum_{i=1}^5 17\sin(2\pi 50i jT) \text{ Degrees.}$$



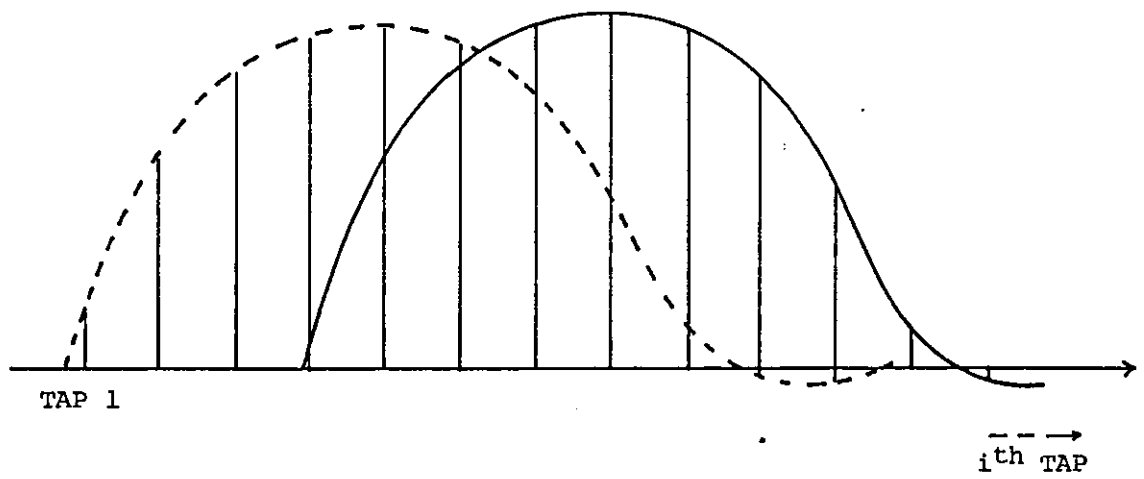


Figure 8.4.1: Movement of Tap Coefficients in Presence of Timing Frequency Offset

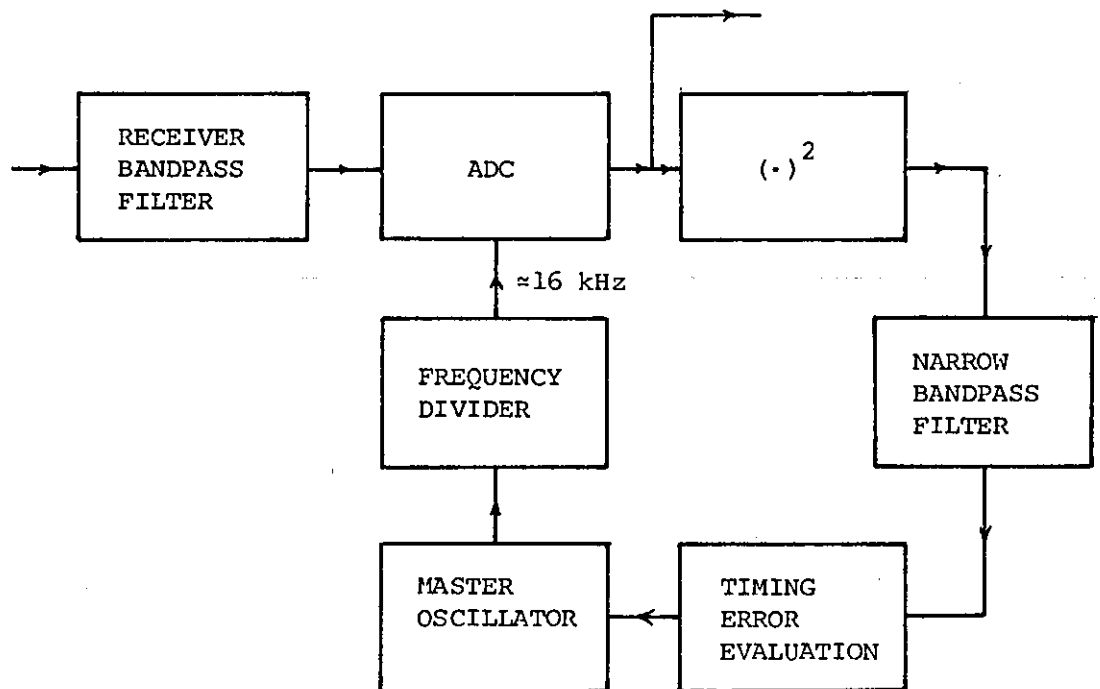


Figure 8.4.2: Timing Recovery Using a Digital Squaring Loop

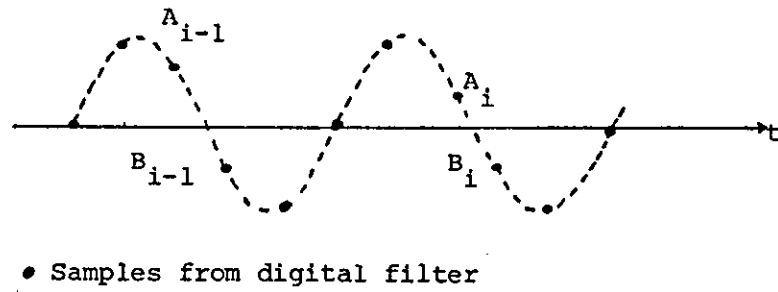


Figure 8.4.3: Output from the Narrow Bandpass Digital Filter

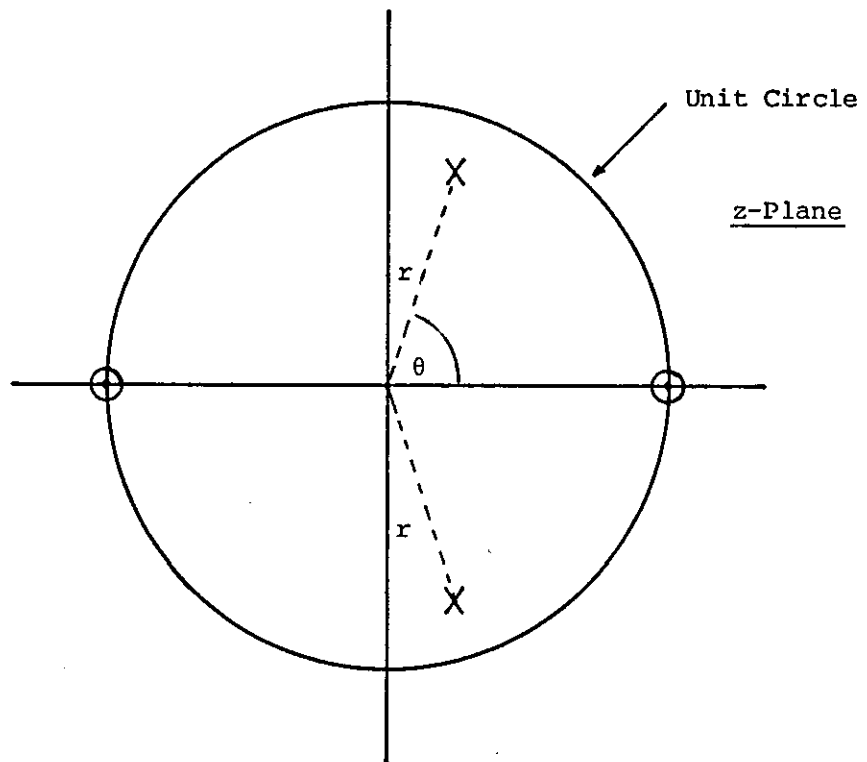


Figure 8.4.4: Pole-Zero Diagram of a Digital Bandpass Filter

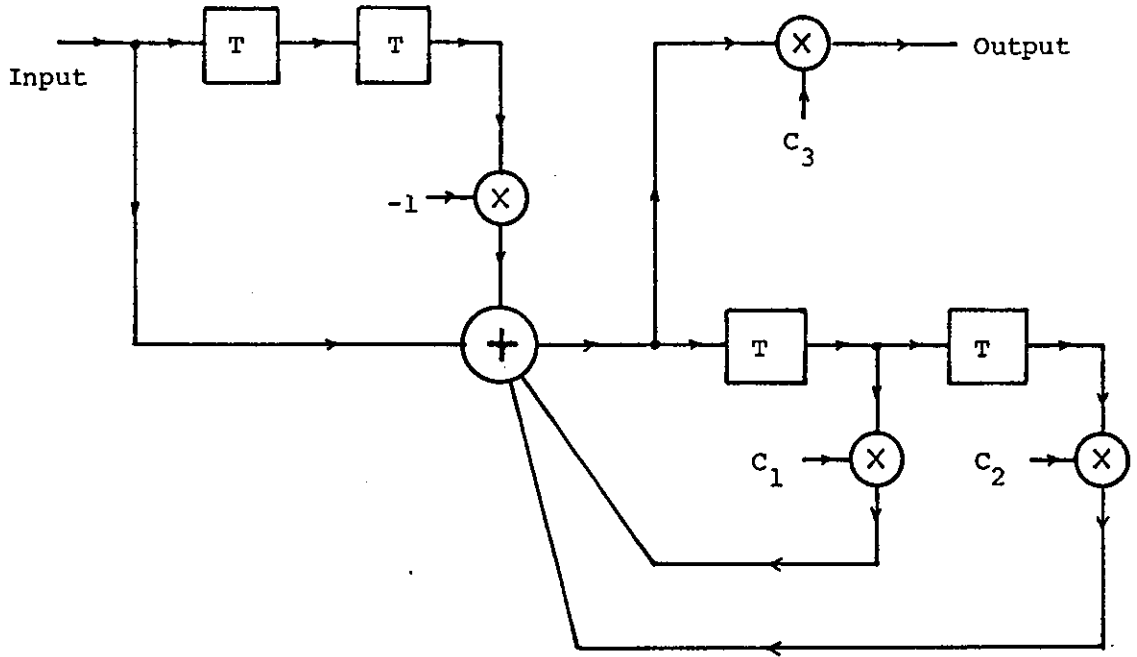


Figure 8.4.5: Digital Bandpass Filter

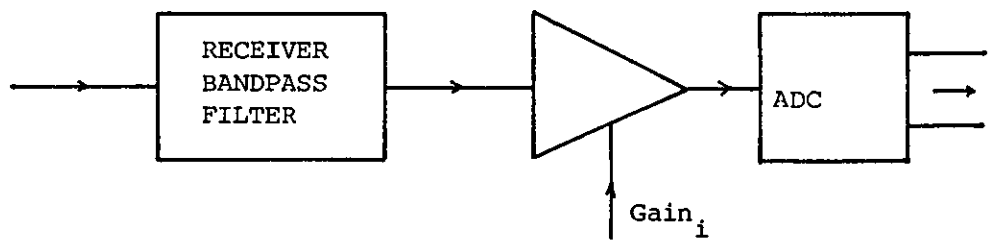


Figure 8.5.1: Receiver Automatic Gain Control

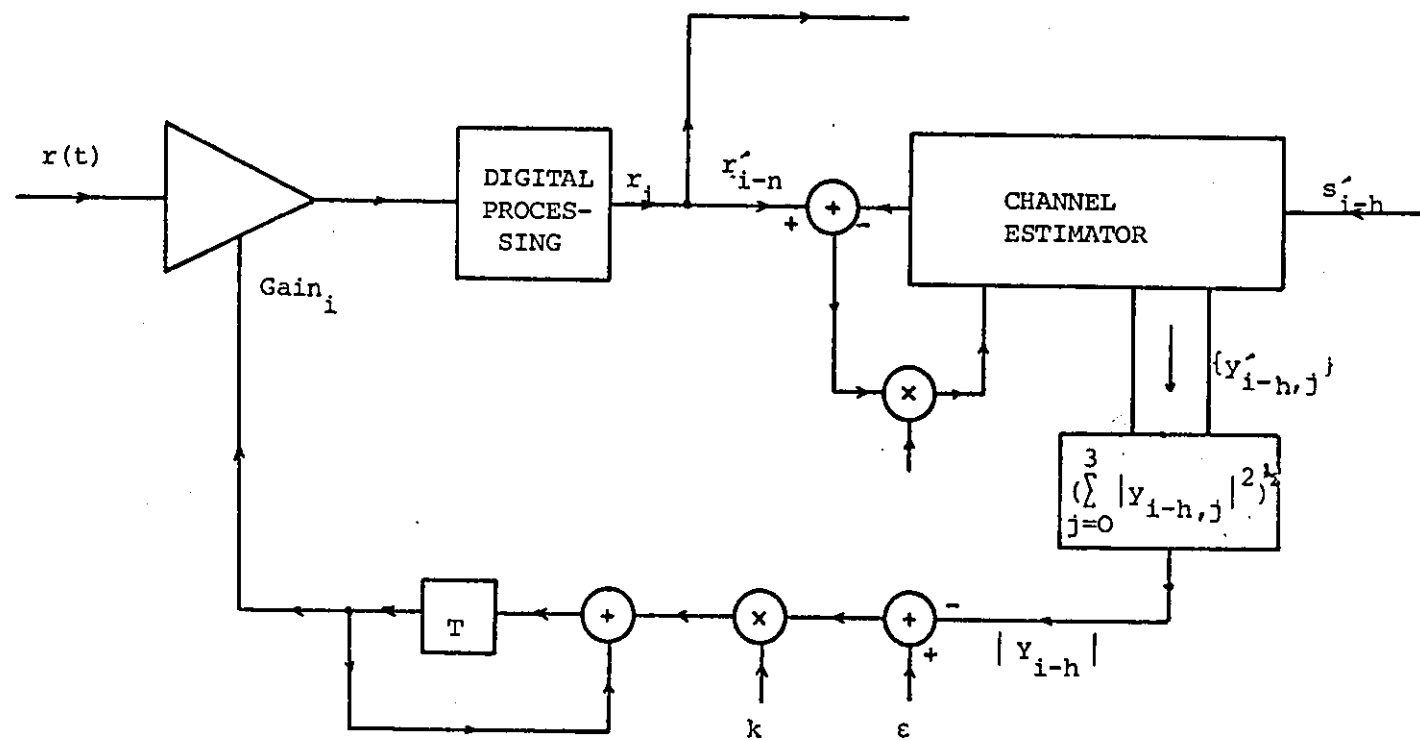


Figure 8.5.2: AGC System

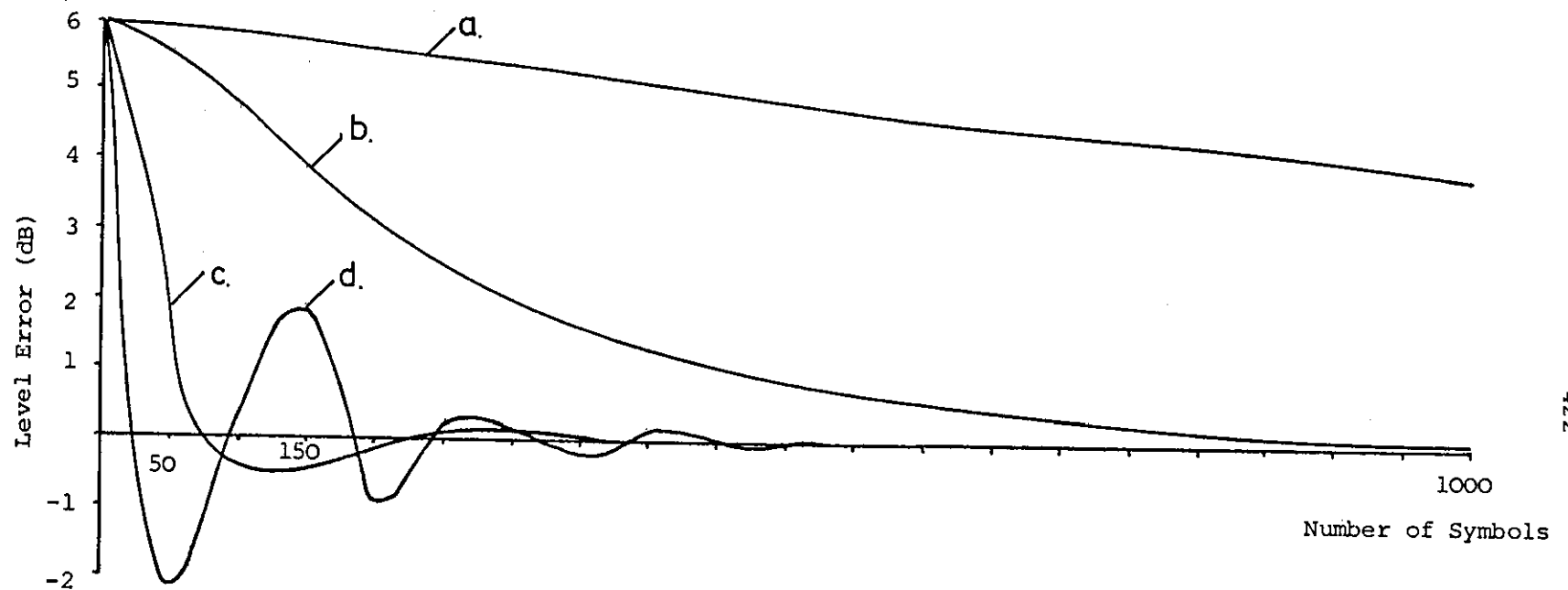


Figure 8.5.3: AGC Circuit

Variation of Level Error with Number of Symbols after
Application of a 6 dB Change in Signal Level, at 31 dB
Signal-to-Noise Ratio.

Integration Parameter; a = 0.001
b = 0.01
c = 0.1
d = 1.0

Channel = Circuit 3 + Filter 2

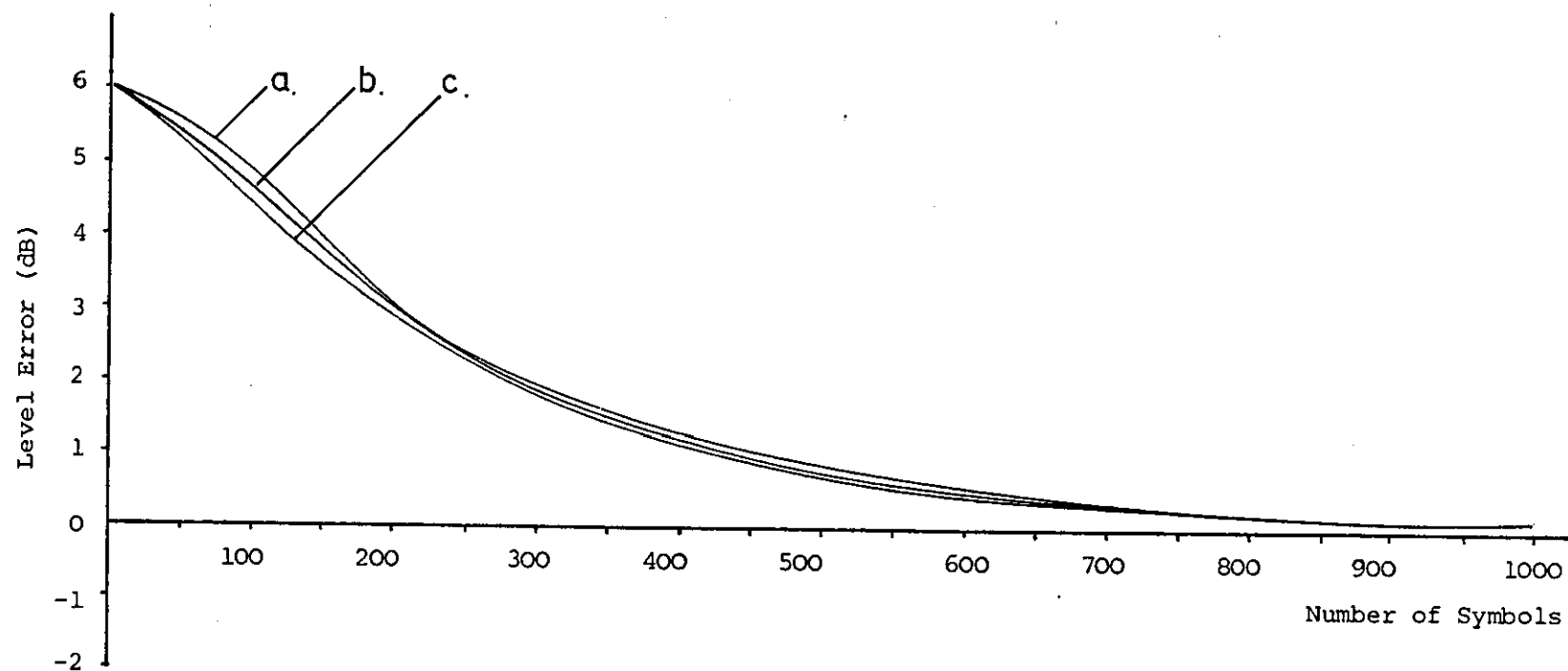


Figure 8.5.4: AGC Circuit

Variation of Level Error with Number of Symbols
after Application of a 6 dB Change in Signal Level

Integration Parameter = 0.01

Signal-to-Noise Ratio a = 15 dB

b = 20 dB

c = 35 dB

Channel = Circuit 3 + Filter 2

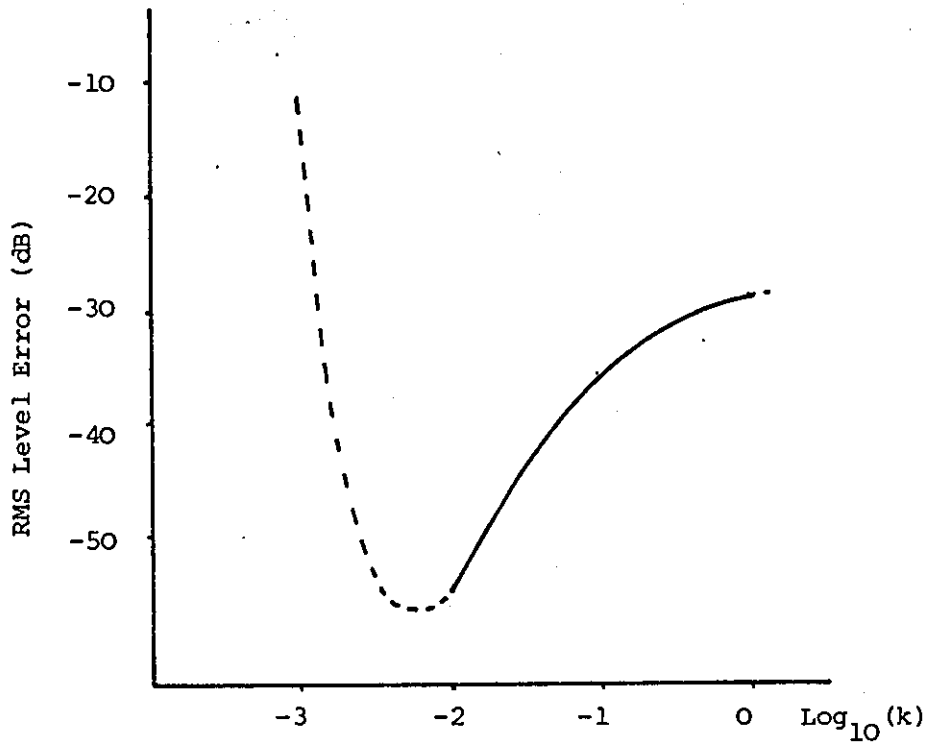


Figure 8.5.5: AGC Circuit

Variation of RMS Level Error with Integration

Parameter after Convergence to a 6 dB Level Change

SNR = 36 dB

Channel = Circuit 3 + Filter 2

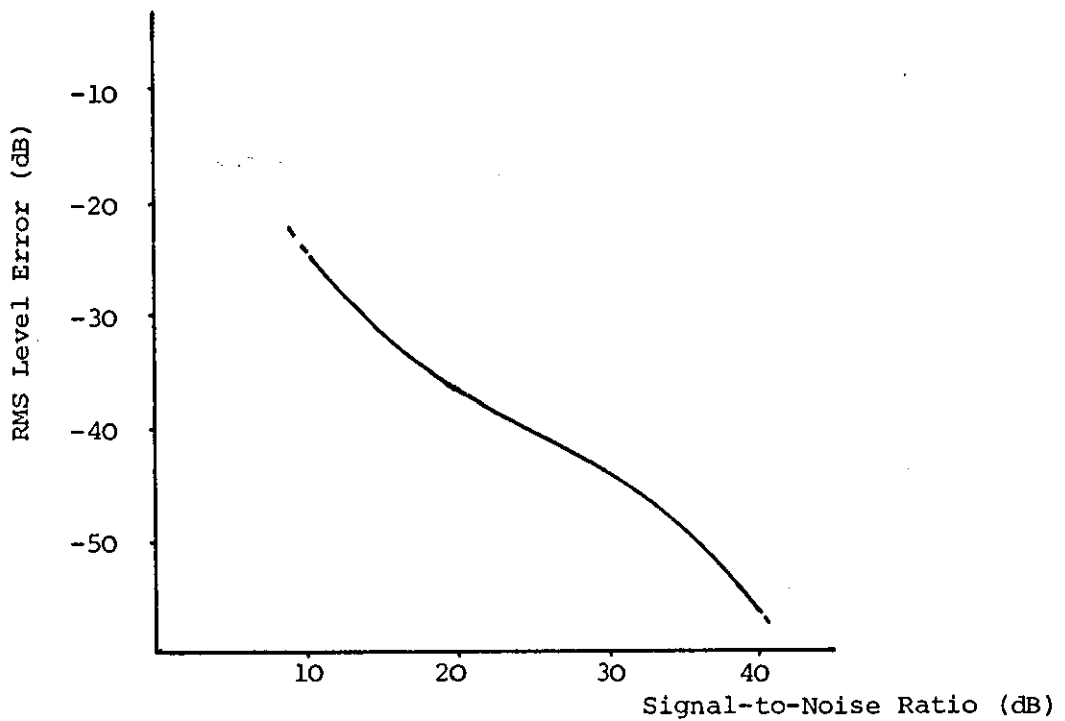


Figure 8.5.6: AGC Circuit

Variation of RMS Level Error with SNR, after

Convergence to a 6 dB Level Change

$k = 0.01$

Channel = Circuit 3 + Filter 2

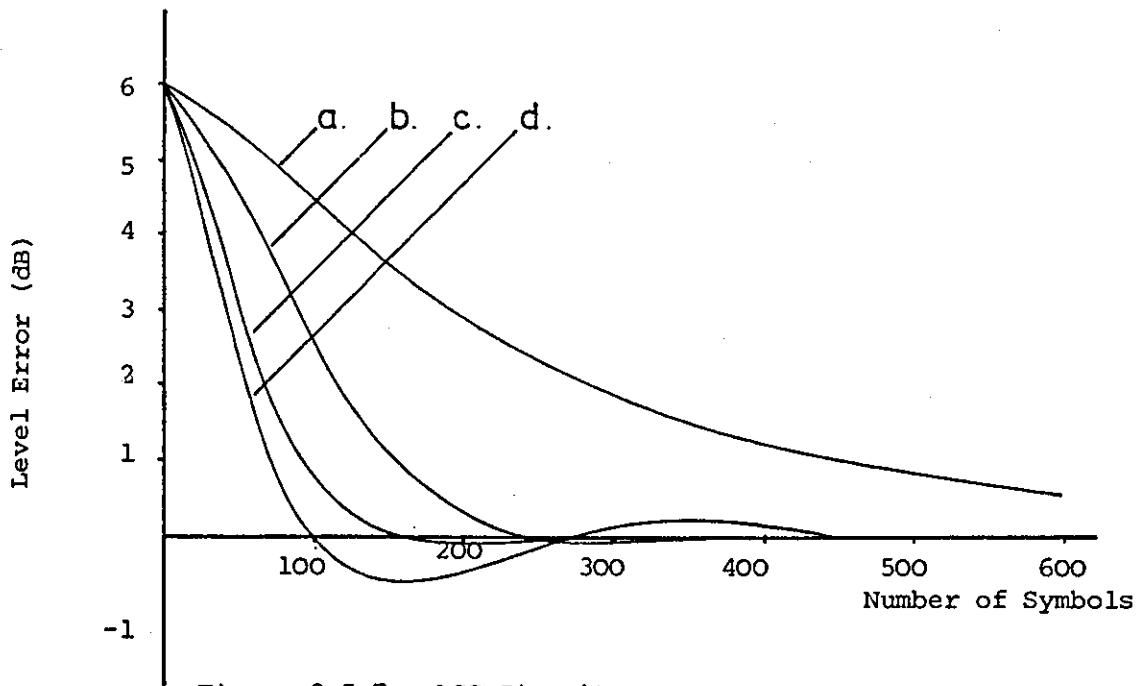


Figure 8.5.7: AGC Circuit
 Variation of Level Error with Number of
 Symbols after Application of a 6 dB
 Level Change
 SNR = 30 dB
 a; $k_o = 0.1$, $k_f = 0.01$, $\Delta = 0.15$
 b; $k_o = 0.1$, $k_f = 0.03$, $\Delta = 0.15$
 c; $k_o = 0.1$, $k_f = 0.05$, $\Delta = 0.15$
 d; $k_o = 0.1$, $k_f = 0.075$, $\Delta = 0.175$

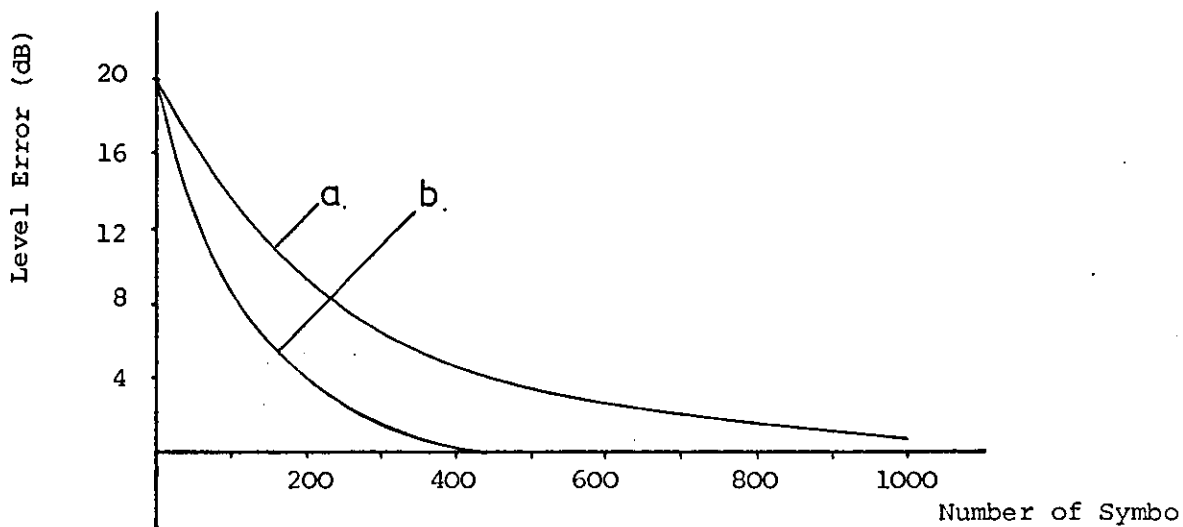


Figure 8.5.8: AGC Circuit
 Variation of Level Error with Number of
 Symbols after Application of a 20 dB
 Level Change.
 SNR = 35 dB, a; $k_o = 0.1$, $k_f = 0.05$, $\Delta = 0.15$
 b; $k_o = 0.1$, $k_f = 0.05$, $\Delta = 0.01$

9 Modem Design and Implementation, and Suggestions for Further Investigations

9.1 Introduction

The results and discussions of previous chapters have shown that transmission of digital data at 19,200 bit/s over telephone circuits is indeed a feasible proposition. However, due to the severe levels of signal distortion introduced by these bandlimited channels, the modem receiver unit is considerably more complex than systems operating at lower information rates. Moreover, as the proposed system will operate at 3200 baud, the reduced time available to actually perform the fairly involved receiver functions will place extra demands on the system designer.

The aim of this chapter is to introduce a general design strategy for a 19,200 bit/s modem rather than the presentation of very detailed system design. The reason for this approach is twofold. Firstly, about two-thirds of the way through the research project, a leading data communication company, CASE of Watford, UK, expressed an interest in the modem design and agreed to devote considerable resources for the development of a prototype system under the direction of Prof. A. P. Clark, Loughborough University. Although some very encouraging results were forthcoming from this collaboration and indeed a basic working prototype was apparently produced, an unfortunate reorganisation within the company involved the relocation of the Research and Development Group to America. Apart from making close collaboration impossible, the move also resulted in the loss of key personnel at an extremely critical part of the work. More importantly, contact with the remaining member of the original team was not re-established. It is indeed disappointing to speculate that the system may now be under development in America rather than in this country.

The second reason for adopting a general approach to the system design leads on from the above mentioned industrial collaboration. Due to the complexity of the design, it soon became clear that state-of-the-art signal processing devices such as the Texas Instruments TMS 32010 or custom-designed signal processors would have to be used in the implementation of the modem receiver. Unfortunately, the combination of inadequate supplies of developmental hardware and the projected man-months (or even man-years) necessary for the development of the modem receiver placed further work well out of the research contract period. The task of actually implementing the algorithms and various time-saving techniques given in earlier chapters has, therefore, still to be completed.

This chapter first discusses the requirements of the transmitter stage when implemented using fairly conventional digital techniques and then presents a fairly simple transmitter design. Following a review of possible methods for the receiver implementation, two arrangements are then considered in more detail. The first arrangement involves a custom-designed parallel multiplier-accumulator structure whilst the second method briefly assesses the suitability of single-chip digital signal processing devices for modem implementation.

9.2 Transmitter Design

Figure 9.2.1 shows the basic block-diagram of the modem transmitter. Since the majority of the transmitter is to be implemented using digital technology, it can be seen that it is a relatively simple system although as will be shown later, the sensible selection of digital sampling rate and carrier frequency will reduce complexity, and therefore the chip-count, considerably. The various functions outlined in Figure 9.2.1 will now be explained in more detail.

The serial input stage is shown to comprise a data multiplexer, data scrambler and serial-to-parallel converter. Strictly, the input to the modem itself is considered here to be suitably scrambled data at 19,200 bit/s, but the multiplexer and scrambler are included to illustrate the possible origins of the source data; it follows that an unscrambler and demultiplexer would be required at the destination. The serial-to-parallel converter also forms part of the baseband signal generator and operates on the 19,200 bit/s data to form the complex data symbols, where each symbol is contained in a 6-bit word. Clearly, the block structuring of the data produces the required symbol rate of 3200 symbols per second and the 6-bit symbols represent the 64 possible points in the complex signal constellation shown in Figure 4.4.20.

The 6-bit symbols are then fed to the differential and Gray encoder which operates on the symbols in the manner of Coding Scheme 3 in Section 4.4.3. This coding stage is included to increase the tolerance of the system to phase ambiguities of $n \frac{\pi}{2}$ rads, $n = 1, 2, 3$, and to reduce the number of bit errors if a symbol is erroneously detected. The differentially encoded 6-bit word is then fed to the symbol encoder which divides the word into two 3-bit subwords, where the subwords represent the real and imaginary components of the complex data symbol. The subwords also represent the digital equivalent of one of the eight possible levels $\pm 1, \pm 3, \pm 5, \pm 7$, as defined by the signal constellation. In practice, the differential/Gray and symbol encoding may be performed simultaneously.

Although the data symbols at the output of the encoder are represented by some digital numbering scheme, the spectrum of the underlying time signal will be a $\sin x/x$ function with zeroes located at multiples of 3200 Hz. To shape the spectrum of the signal prior to modulation,

identical digital low-pass filters are included in the signal path of both

the real and quadrature symbol components, as shown in Figure 9.2.1.

Assuming that the carrier frequency to be used in the (quadrature) modulator is set at 1800 Hz and that the transmitted signal is to be constrained to a bandwidth of ± 1600 Hz around the carrier, it will be necessary to change the sample rate of the signals prior to filtering and modulation in order to prevent severe aliasing problems. This can be illustrated as follows. Consider a digital low-pass filter operating at 3200 samples per second, (s/s). If the filter is assumed to have a cut-off frequency of around 1600 Hz, its amplitude response will be approximately as shown in Figure 9.2.2. The spectrum of the signal at the output of the filter will therefore be similar to that given in Figure 9.2.3. After modulation by an 1800 Hz carrier, the output signal spectrum will contain severe aliasing components, as shown in Figure 9.2.4. To eliminate the aliasing, it will be necessary to increase the sample rate of the signal and perform the filtering operation at the higher sample rate prior to modulation.

Consider now a digital sampling rate of 7200 s/s. At this rate, four samples per cycle of an 1800 Hz carrier will be required for the modulation process. Figure 9.2.5 shows that the sampling phase can be arranged such that the modulation process reduces to multiplication by the sequences (1, 0, -1, 0, 1, 0, -1, 0, 1, ...) for the in-phase channel and (0, 1, 0, -1, 0, 1, 0, -1, 0, ...) for the quadrature channel. Also, since the zero-valued samples of the sine and cosine carriers alternate, then complete modulation can be performed by using the sequence (1, 1, -1, -1, 1, 1, ...) "switched" between the in-phase and quadrature data channels. The process of modulation will therefore be very simple to perform. The spectrum of the signal at the output of one of the modulators now appears as in Figure 9.2.6 where a 400 Hz guard band will exist between the required spectrum and the (modulated) reflection. To enable the specification of the analogue channel filter to be relaxed,

the digital sampling rate can be further increased which will enlarge the separation between the wanted and unwanted spectral components. Moreover, further simplification can be achieved if the ratio (digital sample rate)/(symbol rate) is arranged to an integer value, whilst maintaining an integer relationship between digital sampling rate and the carrier frequency. The latter condition is required to enable the samples of the cosine and sine carrier waveforms to consist of (known) repetitive sequences. A good compromise is to use a digital sampling frequency of 16 kHz and a carrier frequency of 1777.777 Hz. Each cycle of the cos/sin carriers will now be defined by 9 sample values, as shown in Figure 9.2.7, whilst the sample rate of the data symbols will be increased 5-times.

Several different approaches can be adopted for the actual realisation of the transmitter unit and will depend upon the experience of the design team as well as the type of implementation preferred by the manufacturer. For example, the various stages shown in Figure 9.2.1 may be implemented separately in hardware, including the multiplications required by the filtering and modulation processes. Alternatively, it would be possible to use a custom-designed integrated circuit or one of the commercially available digital signal processing devices described in the following section. The following design uses a hardware approach and attempts to reduce the number of calculations required by using a pre-programmed memory device as a simple look-up table.

A possible input configuration is shown in Figure 9.2.8. The 19,200 bit/s serial data is first converted to 6-bit parallel form by a single serial-in-parallel-out shift register, such as the 74LS164⁽¹⁾. The shift rate of the register is controlled by a 19,200 Hz clock signal synchronised to the incoming bit stream. The parallel data is then fed to an 8-bit latch (74LS377⁽¹⁾) at 3200 words per second. The data loaded into the

latch is therefore representative of a new 6-bit symbol value. The data symbol is then fed to a look-up table which performs simultaneously the differential/Gray coding (as described in Section 4.4.3) and the symbol to subsymbol coding process. The output of the look-up table comprises two 3-bit words which represent the position of the coded symbol within the signal constellation and a further two bits which define the quadrant of the coded symbol. These latter two bits are fed-back to the latch on the input side of the memory for purposes of differential coding. The actual contents of the memory may be easily obtained from the signal constellation, Figure 4.4.20, and the coding diagram, Figure 4.4.13. As the look-up table operates with 8 address lines, the minimum memory size will be 256×8 . At this point in the transmitter, the coded symbols are represented by two 3-bit words occurring at a rate of 3200 per second. That is, two 3-bit words effectively define a pair of multilevel, rectangular baseband signals⁽²⁾. The change of digital sample rate from 3200 s/s to 16,000 s/s may be achieved in one of two ways. The first method simply takes the samples defined by each of the 3-bit words and repeats them for four other sample points at the increased digital sampling rate. This has the same effect as sampling the underlying rectangular baseband signals at 16,000 s/s. An alternative to this is to apply the samples (occurring at 3200 Hz), to a 5-tap digital filter operating at 16,000 s/s. The output from this filter, assuming all tap gains are set to unity, will be the same finely-sampled sequence as obtained from the first method. The higher-rate samples are then applied to the digital low-pass filters prior to modulation. The advantage of the second method is that the 5-tap filter and the pre-modulation filter may be combined in a single process, which is effectively the same as considering the symbols as a stream of suitably weighted impulses feeding into a modified filter, where the modifications

convert the impulses into a rectangular baseband signal. By using the modified filter, the super-sampling of the data symbols can be performed by following each sample (at 3200 s/s) with four equally spaced, zero-valued samples.

Consider now a filter with $(l+1)$ taps whose tap coefficients are the $\{c_i\}$, $i = 0, 1, \dots, l$. If s_i represents a sample occurring at both the original and higher sample rates, then $\{s_{i-j}\} = 0$, for $j = 1, 2, 3, 4, 6, 7, 8, 9, 11, \dots$. The output from the filter at time $t = i\tau$ will be,

$$f_i = s_i c_0 + s_{i-5} c_5 + s_{i-10} c_{10} + \dots \quad (9.2.1)$$

and at time $t = (i+1)\tau$, where $\tau = 1/16,000$,

$$f_{i+1} = s_{i+1} c_1 + s_{i-4} c_6 + s_{i-9} c_{11} + \dots \quad (9.2.2)$$

It follows that the samples at the output of the filter may be obtained by passing the original symbol samples through a filter with $(g+1)/5$ tap coefficients, where each tap can have one of five different values. If the original filter is assumed to have, say 80 tap coefficients, then the following five sets of new filter taps can be defined;

$$\begin{aligned} \text{Set 1 , } & (c_0 \ c_5 \ c_{10} \ c_{15} \ c_{20} \ \dots \ c_{75}) \\ \text{Set 2 , } & (c_1 \ c_6 \ c_{11} \ c_{16} \ c_{21} \ \dots \ c_{76}) \\ \text{Set 3 , } & (c_2 \ c_7 \ c_{12} \ c_{17} \ c_{22} \ \dots \ c_{77}) \\ \text{Set 4 , } & (c_3 \ c_8 \ c_{13} \ c_{18} \ c_{23} \ \dots \ c_{78}) \\ \text{Set 5 , } & (c_4 \ c_9 \ c_{14} \ c_{19} \ c_{24} \ \dots \ c_{79}) \end{aligned} \quad (9.2.3)$$

In order for the filter's output samples to appear at the required rate, each tap coefficient must be changed to its corresponding element in the next set in the sequence at a rate of 16,000 per second. By using this

method, the super-sampling and filtering operations have been reduced to the fairly simple task of operating on the original symbol samples with selectable sets of filter tap values.

Every output sample from the filters will now have to be multiplied by the appropriate sample of the cosine or sine carrier. The two samples will then be added to produce a sample of the QAM signal. The multiplications can either be performed on the final outputs from the filters or alternatively, by multiplying all tap coefficients in the selected set by the samples of the carrier waveform. Using the latter method, the modulation and filtering operations can be combining by arranging for all tap coefficients in the selected set to be replaced by the product of the original tap value and the sine or cosine sample value. Consequently, the complete digital processing required between the symbol encoder and the digital-to-analogue converter (DAC) can be performed by using a look-up table, an accumulator and an address sequencer.

The look-up table will contain the partial products of all combinations of the sampled sine and cosine waveforms (sampled at 16,000 s/s), with the 8 possible sub-symbol values and the 80 tap coefficients. A sequencer then selects the appropriate partial products for accumulation before application to the DAC. The accumulator essentially performs the same function as the adder in a conventional digital filter. Since the filters used to shape the in-phase and quadrature sub-symbols will be identical and since the output from the two modulators must be added together to produce the QAM signal sample, the final addition can also be performed very easily by accumulating the partial products of the in-phase components and the quadrature components. Figure 9.2.9 shows the basic format of the proposed filtering and modulation scheme.

The sequencer used to control the selection of the partial products is obviously an important part of the system and will now be considered in more detail. As well as providing information relating to the actual cosine or sine value and filter tap coefficient required, the sequencer must also select either the in-phase or quadrature sub-symbol from a set of 16 symbols held in a temporary store. This suggests that the current symbol and the previous 15 symbols must be held in a shift register from which the sequencer can select a symbol as required. Clearly, a new symbol, or more precisely, the 6-bits representing the new symbol, must be entered in the register every $1/3200$ sec.

As the symbol samples are represented by 6-bit words, the shift and select functions can be implemented using 6 x 16-bit shift registers, 6 x 1-of-16 data selectors and a single quadruple 2-to-1 line data selector. Four control bits will be required to select one symbol from the 16 held in the shift register, these bits being designated "symbol identifiers". In addition, one bit designated H, will be required to select the in-phase or quadrature sub-symbols (3-bits) from the selected symbol.

The sequencer will also control the memory address lines concerned with tap value and sine or cosine sample selection. In fact, bit H will be one of the bits in the latter address field. In total, 14 address lines to the memory will be required; 4 for the sine or cosine samples, 7 for the filter tap selection and 3 for the sub-symbol identifier, as shown in Figure 9.2.9. Hardware solutions for the sequencer, including shift and select operations, are shown in Figures 9.2.10 and 9.2.11. The operation of the complete arrangement will now be explained, with reference to Figures 9.2.9, 9.2.10 and 9.2.11.

As already stated, the output samples from the two modulation processes have to be added to produce the QAM sample. It is therefore

convenient to accumulate the partial products produced by a single tap value address for both the sine and cosine waveform samples. The in-phase sub-symbol and cosine sample are used to generate the "in-phase" partial product and the quadrature sub-symbol and sine sample generate the "quadrature" partial product for a given tap selection. For example, a particular tap coefficient, say c_0 is selected by the appropriate address on the "tap select" lines. The required symbol, say s_n , is selected from the 16 held in store and the sine/cosine control bits are set to give the required cosine sample and the in-phase symbol component. The partial product obtained from the memory is then stored in the accumulator. Maintaining the same tap and symbol selection, the sine sample and quadrature symbol component are then selected to give the corresponding "quadrature" partial product which is added to the contents of the accumulator. The next symbol is then selected along with the next tap value. Again, the cosine sample and in-phase symbol component provide the "in-phase" partial product, whilst the sine sample and quadrature symbol component provide the "quadrature" partial product. These are accumulated as before. This process is repeated 16 times after which the value held in the accumulator will be the required QAM sample and is applied to the DAC.

After the 16 pairs of basic cycles, where each pair uses the same tap coefficient and symbol selection, the symbol select lines are reset to re-select symbol s_n whereas the tap select lines locate the first tap value, c_1 , in the second set of tap coefficients. Again, 32 basic cycles will be performed but now using the next sine/cosine sample. It can be seen that each set of 32 basic cycles use a different set of filter tap coefficients and so after 5 sets of cycles, i.e. 160 basic cycles, the tap select lines must be reset to address tap c_0 . The latest coded symbol will also be available at this point in the process and is shifted into the input registers, all other symbols moving one place to the right.

From the above description, it should be clear that the shift registers will require a 3200 Hz clock signal, the symbol select logic should operate at 256 kHz and the sub-symbol selector should change at 512 kHz. Similarly, the cosine/sine select lines should change at 512 kHz. In practice, the location of the partial-products within the memory will be such to enable the sequence controller to be constructed using simple counter circuits. From Figure 9.2.11, the symbol select lines ABCD will change at 256 kHz and can be realised as a $0 \rightarrow 15$ counter. After 16 changes or counts, the select lines automatically reset to zero. The cosine/sine select mechanism first selects the required sample position, from sample 0 to sample 8 and then selects either the sine sample or cosine sample using bit H. Bit H also controls the decoder which selects either the in-phase or quadrature sub-symbol.

9.3 Receiver Design

Figures 9.3.1 and 9.3.2 show the general arrangement of the functions required to be performed by the receiver unit. All of these functions have been described in some detail in previous chapters along with suitable algorithms, possible time-saving techniques and the approximate number of operations required by each function. However, it should be noted that for the actual implementation of the receiver unit, the ADC, demodulation and filtering processes will have to be performed at a higher digital sampling rate. For the same reasons as discussed in Section 9.2, 16 kHz will be used as the digital sampling frequency when the system operates with 1777.777 Hz carriers.

As the majority of the modem receiver (and in fact the transmitter) operate with digital signals, all of the processes required within both units can be reduced to multiplication, addition/subtraction, accumulation and comparisons along with adequate control and routing of the data obtained

during intermediate processing steps. During the "early" years of high-speed modem design, from about 1975-1980, the majority of systems were implemented using a mixture of analogue and discrete digital technologies⁽³⁾. Although the necessary speed of operation could be achieved with this approach, the requirement of higher-speed systems and the development of more sophisticated algorithms limited the use of discrete technology for reasons of sheer size.

The late 1970's and early 1980's witnessed the development of large scale integration which brought with it the attractive possibility of designing state-of-the-art modems which were software, rather than hardware intensive. During this period, many modems were successfully designed and made commercially available but due to the comparatively slow cycle-times of these early processors, the designs had to be kept at a fairly low levels of sophistication, unless multi-processor, highly parallel structures were used in the implementation. Perhaps the most important development in microelectronics as far as digital signal processing is concerned, was the introduction of microcomputers whose internal architectures were tailored for the needs of discrete-time signal processing. These devices, such as NEC 7720⁽²⁹⁾ and Texas TMS 32010⁽³¹⁻³³⁾, allow multiple operation cycles by providing several data transfer paths between their various sections, hence encouraging "pipelined" operation. Many such devices are now available, each boasting extra levels of sophistication for a multitude of applications. Indeed, although these devices have solved many problems for design engineers, a secondary problem now exists in the choice of device to use in a specific application.

References 4-14 give some examples of the various methods used to implement data modems over the past 15 years. At present, the general trend in design is to implement a prototype using available high-speed processing devices followed by the production of VLSI custom-designed

integrated circuits. It is also interesting to note that modems operating at up to 1200 bit/s are now available as single integrated circuits⁽¹⁵⁻¹⁷⁾. Also, major companies such as Texas Instruments are now producing a whole range of integrated circuits for the telecommunication industry⁽¹⁷⁾.

As mentioned earlier, the majority of digital signal processing functions and therefore the realisation of the algorithms given in earlier chapters, may be reduced to a series of basic operations such as,

$$A.B + C \rightarrow D \quad (9.3.1)$$

where the contents of data registers A and B are first multiplied together before being added to the contents of register C. The result is then stored in register D. An alternative form of equation (9.3.1) is

$$X.Y \pm P_n \rightarrow P_{n+1} \quad (9.3.2)$$

Here, the register P acts as an accumulator of the products X.Y. If the values of X and Y change every cycle, then equation (9.3.2) defines the basic multiply-accumulate function. Clearly, if P_n is set to zero, register P (the product register) will simply contain the product X.Y. Alternatively, if X or Y is set to unity, simple addition/subtraction can be performed. The following sections describe the development of an arithmetic unit based around an LSI multiplier-accumulator device. Although intended for application in a data modem, the architecture presented is quite general and can be easily adapted for other real-time digital signal processing applications such as speech coding and signal analysis.

9.3.1 The Multiplier-Accumulator (MAC)

Multiplier-accumulators are available as single integrated circuits from a variety of manufacturers covering a range of speed, power consumption and number of bits used. To allow for a sufficient accuracy, a 16-bit MAC was chosen for this application. At the time of the study, the only

available device was the TRW TDC1010J MAC which executes a 16 x 16 bit multiplication and 35-bit extended accumulation in a cycle time of approximately 115 ns^(18,19). CMOS versions of this device (for example, TRW TMC2010⁽²⁰⁾) are now available which operate with a considerable reduction in power dissipation. Also, higher speed 16 x 16 bit MACs have become available over the past two years, such as the Weitek WTL1010A, which has a typical cycle time of 80 ns^(20,27).

Figure 9.3.3 shows the contents of the TRW TDC1010J MAC in block diagram form⁽¹⁸⁾. The input data is loaded into the X and Y registers on the rising edges of CLK X and CLK Y signals respectively. Typical data set-up time is 12 ns. To limit the number of pins used on the integrated circuit package, the Y data inputs are timeshared with the LSP outputs. The product, or other result, is divided into a least significant product (LSP), a most significant product (MSP) and an extended product (XTP). The LSP and MSP are both 16-bits wide whilst the XTP is 3-bits wide. The product is loaded into these output registers (the P registers) on the rising edge of CLK P. The XTP register allows 35-bit accumulation and accumulation without overflow.

Arithmetic control instructions are loaded into a 4-bit control register on the rising edge of (CLK X. OR. CLK Y). The TC control selects either two's complement (TC = 1) or integer magnitude (TC = 0) data format. RND = 1 adds a '1' to the MSB of the LSP to round-up the product held in the MSP register. If ACC is low, the output from the multiplier array is clocked into the P register. When ACC = 1 and SUB = 0, the contents of the P register are added to the product X.Y before application to the product register. If ACC = SUB = 1, subtraction is performed.

Input/output control is achieved by the PREL, TSX, TSM and TSL bits which are direct, non-latched control signals. If PREL is low, then TSX, TSM and TSL control the three-state output buffers on the XTP, MSP and

and LSP registers respectively. A logic '0' on these controls enables the output buffers. If PREL is high, then TSX, TSM and TSL enable CLK P to operate on the three output registers, respectively. If a register is enabled by a logic '1' on TSX, TSM or TSL, data present on the input/output lines of the register is loaded into that register on the rising edge of CLK P to allow pre-loading of the product registers.

9.3.2 Processor Bus Structure

The task of the MAC is to relieve the controlling microcomputer of all "number crunching" of data, leaving the microcomputer to oversee general system control, memory management and data transfers to and from the MAC. As all of the registers internal to the MAC (with the exception of the XTP register) are 16-bits wide, an external bi-directional 16-bit bus is required to load the X, Y, LSP and MSP registers and to read the contents of the LSP and MSP registers. To prevent a bottleneck occurring when data transfer is required to both the X and Y registers, an additional register, AUX, is included in series with the X input. The memory can now be accessed twice per MAC cycle with the data bus multiplexed between the two input registers. Figure 9.3.4 shows this modification. During the first half of a MAC cycle, the AUX register is loaded with the X input data from the memory. On the next memory cycle, the Y input data is loaded into the Y register, whilst at the same time, the data in the AUX register is applied to the X input. If 120 ns is allowed for each MAC cycle, the memory will operate with a 60 ns cycle time. Also shown in Figure 9.3.4 are 3 three-state buffers used to sign-extend the XTP register; the sign of the product is obtained from the MSB of the extended product. The data memory connected to the 16-bit bus is controlled by the address and control lines from the microcontroller. Depending upon the

degree of sophistication, the controller may be the overseeing microcomputer or an address sequencer under the control of the microcomputer.

However, regardless of the actual source of the control, the MAC and associated memory will be essentially under the control of some device programmed to execute the required functions in the correct sequence using data generated from the MAC, obtained from the data memory, or from external sources. For this reason, part of the memory is reserved for memory-mapped input/output. For a digital modem, these input/output channels could be used for communication between a DAC or ADC and data transfers between the controlling processor. As the memory must be capable of two accesses per MAC cycle, an access time of 60 ns will be required. For example, the Intel 2149H RAM has an access time of 45-55 ns. Control over the memory mapped input/output is achieved by the I/O Enable, the address lines and the \overline{RW} memory control. Device selection is achieved via the address decoder.

9.3.3 Data Transfer Instructions

A high degree of flexibility is available in the signal processor shown in Figure 9.3.4 due to the various ways of transferring data to and from the MAC, memory and memory-mapped devices. Four data transfer instructions (DTI) will now be described. Each instruction involves two transfer operations occupying a single MAC cycle.

DTI 1 MEM \rightarrow AUX ; MEM \rightarrow Y , AUX \rightarrow X

memory cycle 1 ; Data in memory loaded into AUX register

memory cycle 2 ; Data in memory loaded into Y register.

Contents of AUX register loaded simultaneously
into the X register.

This instruction allows pairs of data values to be loaded into the MAC and is useful for the efficient calculation of sums of products.

DTI 2 MEM \rightarrow LSP ; MEM \rightarrow MSP

memory cycle 1 ; Pre-load LSP with memory content.

memory cycle 2 ; Pre-load MSP with memory content.

This instruction, when used with DTI 1, allows the calculation of $X.Y \pm C \rightarrow P$, where C is a constant held in memory and P is the product register. If X or Y is set to zero, the product register can be used as a storage register for the relocation of data within the memory.

DTI 3 LSP \rightarrow MEM ; MSP \rightarrow MEM

memory cycle 1 ; Load memory with contents of LSP.

memory cycle 2 ; Load memory with contents of MSP.

If only the MSP is required, the same memory address can be accessed so overwriting the LSP. As well as the storage of results or intermediate results, DTI 3 may also be used to transfer data within the memory.

DTI 4 MSP \rightarrow MEM ; MEM \rightarrow Y

memory cycle 1 ; Load memory with contents of MSP.

memory cycle 2 ; Transfer memory content to Y register.

This instruction allows a series of data values loaded into the Y register to be scaled by a common factor held in the X register. At the end of every MAC cycle, the scaled data is written back to the memory and the next value to be scaled is loaded into the Y register. Powers of X may also be generated.

To enable the four data transfer instructions to be carried-out, the processor must be supplied with the necessary address and control information twice per MAC cycle. This information will be held in the

the address and DTI fields of the microcode instruction word. Figure 9.3.5 lists the various control line conditions necessary for the processor to perform the required instructions. Logic levels are represented by '1' and '0' whilst positive clock edges are shown as \uparrow . All three-state buffers are synchronised to avoid bus contention; an enable is represented by "e". In practice, the four instructions will be selected by two control bits which will be decoded to supply the necessary control conditions to the MAC. These bits are designated B_0 , B_1 in Figure 9.3.5. When the four instructions described above refer to memory, it should be realised that they apply equally to the input/output devices via the memory-mapped selection.

9.3.4 Arithmetic Instructions

The three basic arithmetic operations that the MAC can perform are;-

- a) $X.Y \rightarrow P$
- b) $X.Y + P \rightarrow P$
- c) $X.Y - P \rightarrow P$

A fourth instruction, which would be extremely useful for the implementation of a maximum likelihood detector, is a compare-type function, which can select the minimum value of a set of numbers held in memory. This instruction operates by using the sign output, SGN, shown in Figure 9.3.4. If the product $X.Y$ is less than P , the operation $X.Y - P \rightarrow P$ will result in $SGN = 1$ and $X.Y$ is loaded into P as the latest minimum value. However, if $X.Y$ is greater than P , $SGN = 0$, but the product register contains $X.Y - P$ rather than the original value of the product register. To rectify this, the operation $X.Y - P \rightarrow P$ is repeated such that,

$$X.Y - (X.Y - P) \rightarrow P = P \rightarrow P \quad (9.3.3)$$

Since the input and output registers can be clocked independently, the actual flow of operations through the MAC can be pipelined. That is, when the CLK P latches the result of a calculation into the product register, the new input values may be simultaneously loaded into the X and Y inputs, along with the new instruction. As the sign of the latest product only appears after activation of the product register with CLK P, the control register must be re-clocked (by CLK X. OR. CLK Y) half-way through the MAC cycle. In fact, the control may be achieved by setting $ACC = \overline{SGN}$ and entering this value via CLK X. Since the content of the AUX register remains unchanged, activation of CLK X will not alter the contents of the X register. Figure 9.3.6 shows the control inputs required to implement the four arithmetic instructions. In practice, two inputs, A_0 and A_1 will be used with an instruction decoder to control the state of the MAC. In fact, the four control bits A_0 , A_1 , B_0 and B_1 may be entered simultaneously to a combined data transfer and arithmetic decoder which controls both the memory and MAC. To complete the processor, two address fields can be included with the 4-bit instruction field to give an n-bit word format. Full details of a digital signal processor utilising a MAC in the manner described above, including interface circuitry, address sequencing and example uses may be found in reference 21.

9.3.5 Extended MAC Architecture

An increasing number of digital signal processes require the use of complex arithmetic; for example, the filters used in adaptive channel estimators and equalisers. Consider the multiplication of two complex numbers, $(a + jb)(c + jd)$;

$$(a + jb)(c + jd) = (ac - bd) + j(ad + bc) = x + jy \quad (9.3.4)$$

Clearly, such a multiplication involves four real multiplications, one addition and one subtraction and may be represented diagrammatically as in Figure 9.3.7. From the figure, a MAC state table may be constructed and a timing diagram drawn, as in Figure 9.3.8. It can be seen that a single complex multiplication will require $4\frac{1}{2}$ MAC cycles (9 memory cycles) which reduces to 4 MAC cycles when pipelined.

A possible method to increase the throughput rate of the processor is to use two MACs in parallel. For example, one MAC may calculate the real part ($ac - bd$) of the product whilst the second device can simultaneously calculate the imaginary part ($ad + bc$). However, since both MACs will require data to be entered at the same time, bus contention can be a serious problem with such systems, although full duplication of the data memory can reduce contention to some degree. An alternative solution is to stagger the parallel operation of the two processors by one MAC cycle. Although this method eliminates the need for duplicate memories, the pipelined throughput will be less than that achieved when operating in full-parallel mode. Also, due to the staggered-parallel operation, the times required for data transfers to and from memory, I/O registers, etc., can be lengthy.

After several stages of evolution, the extended MAC architecture shown in Figure 9.3.9 was derived. This maximises the throughput for complex arithmetic by,

- a) Using full parallel operation,
- b) Using bus cross-coupling for full data transfer facilities,
- c) Dividing the data memory into two blocks sharing common address lines,
- d) Providing a complex multiply-accumulate time of 2 MAC cycles.

Essentially, each MAC unit in Figure 9.3.9 is identical to the single MAC processor considered in earlier sections but with cross-coupling between two bi-directional data buses and data memories. The memories themselves can be considered as containing the real and imaginary parts of the data values, with the additional data transfer facility of data exchange between the two memories. Figure 9.3.10 shows the processor state table and Figure 9.3.11 the timing diagram for the realisation of complex multiplications.

By suitable control decoding and memory addressing, the extended MAC processor can be designed to have 8 arithmetic instructions defined by the 3 control bits A_0 , A_1 and A_2 . Four of these instructions will be identical to the 4 instructions described for the single MAC processor, which can be considered as operating on real data. When used in the real mode, the extended MAC processor will double the number of operations performed per MAC cycle. Three of the remaining 4 instructions will provide similar functions to the first 3 real instructions, but will operate on complex-valued data. By using the $X.Y + P \rightarrow P$ instruction, the final arithmetic instruction can calculate the squares of the absolute values of 2 complex numbers. It is interesting to note that the two values produced may then be compared using the fourth instruction in the real set, to determine the smaller of the two values. Figure 9.3.12 summarises the arithmetic instruction set, where it should be noted that control bit A_0 determines the selection of either the four real or four complex instructions.

9.3.6 Alternative Methods for Receiver Implementation

Since the start of this investigation in late 1980, the advances in microelectronic technology and the impact of this technology on the telecommunication industry has been phenomenal^(22,35). Initial ideas in

terms of modem implementation revolved around the use of high-speed arithmetic processors such as the MAC systems discussed in previous sections, under the control of one of the many single-chip microcomputers. However, over the past 2-3 years, major companies such as NEC, Texas Instruments, STC, Hitachi, AMD, etc., have introduced either single-chips or chip families dedicated to digital signal processing applications⁽²³⁻³⁴⁾. Of all these devices, the AMD 29500 family^(23,24,27) most closely resembles the original architecture in that the implementation of the processing machine is built-up using a combination of discrete blocks, including multi-port pipelined processors, multipliers and address sequencers. Moreover, the use of ECL internal circuitry and TTL input/output interfaces enable these devices to operate 3-10 times faster than older solutions based on bit-slice technology^(23,26). Another advantage of using the "building-block" approach compared with the single-chip digital signal processors (DSPs) is that the system may be implemented with minimum complexity for the particular application. This is obviously important to a manufacturer when considering the cost-effectiveness of the final product. However, the more recent additions to the DSP market, such as the Hitachi HD61810⁽²⁷⁾, the STC "CRISP"⁽³⁰⁾ and the Texas TMS32020⁽²⁷⁾, all boast increased memory size, larger instruction sets and more efficient use of internal bus structures for fast data transfer operations.

Although the devices mentioned above are apparently now available, two major problems face the development section of a product manufacturer. Firstly, the choice of device depends not only on the availability of the part(s), but also on the support facilities offered by the device manufacturer. It is an unfortunate fact that the majority of first-issue development and support systems for new devices are expensive and notoriously unreliable. Secondly, although the devices themselves are

being marketed as solutions for many applications, the take-up of the industry and therefore the supply of suitably experienced design engineers has been fairly slow⁽²⁷⁾. It follows that a company entering the highly competitive field of high-speed data communications may face many man-years of development time before the product is realised. A further point for consideration is the eventual selling price of the modem. Due to the high levels of competition, a manufacturer may well have to write-off development costs in order to price the system at a level meeting customer's expectations and demands.

In conclusion, and bearing in mind that none of the present generation of DSPs contain enough processing power to implement the proposed modem design using a single device, it is suggested that development of a prototype modem proceeds by using the minimum "building-block" approach, followed by an assessment of the cost-effectiveness of the solution in relation to market forces and customer's expectations.

9.4 Recommendations for Further Investigations

1. The thesis has shown that the transmission of data at 19,200 bit/s over telephone channels is indeed feasible, albeit with considerable overheads in terms of system complexity. It follows that development costs, company pricing policy and the existence of a suitable market must be carefully considered before embarking on a prototyping development programme.
2. Several operation-saving techniques have been suggested in Chapters 6, 7 and 8. The effects of these techniques on system performance should be investigated further. Also, several of the suggestions were based on the assumption of a very slowly time varying channel characteristic. It would be useful to determine the expected variations of channel characteristics for private and switched connections established through modern

line plant and solid-state switches and to compare these variations (and the levels of impairments introduced) with the results of field studies performed over older equipment.

3. Further work is required to determine the interaction between the synchronising loops and alternative adjustment schemes 3 and 4.

Furthermore, a detailed study of modern root-finding algorithms and their digital implementation could possibly reduce the complexity of an adaptive detector.

4. An in-depth investigation into the use of single-chip DSP devices should be undertaken to determine the number of devices required and possible methods of parallel operation. With the price of these devices falling and their processing power increasing, these devices should be expected to provide a cost-effective solution at least for prototype development in the very near future, although the comments and observations made in Section 9.3.6 should be borne in mind.

5. Considerable interest has been shown of late in the areas of trellis coding and Trellis Coded Modulation⁽³⁶⁻⁴⁰⁾. Increased tolerances to additive noise of up to 4 dB have been reported for trellis-coded QAM systems operating at 14,400 bit/s, 2400 bauds over telephone circuits⁽³⁹⁾. It is strongly recommended that trellis-coded techniques be investigated for the 19,200 bit/s modem by coding the 64-point structure to a 128-point constellation via a convolutional encoder prior to the modulation process.

9.5 References

1. Texas Instruments Ltd., "The TTL Data Book, Vols 1 and 2", Texas Instruments, sixth edition, 1983.
2. Clark, A.P., "Principles of Digital Data Transmission", Pentech Press, 1976.
3. Harvey, J.D., Clark, A.P. and Driscoll, J.P., "High-Speed Line Modems" Final Report, SERC Contract No K/112 11a/616, June 1977.
4. Jones, B.J. and Teacher, V., "Modem at 2400 Bits per Second for Data Transmission over Telephone Lines", Electrical Communication, Vol. 44, No.1, pp. 66-71, 1969.
5. Akashi, F. et al., "A High Performance Digital QAM 9600 bit/s Modem", NEC Res. and Devel., No. 45, pp. 38-49, April 1977.
6. Murano, K. et al., "LSI Processor for Digital Signal Processing and its Application to 4800 Bit/s Modem", IEEE Trans. on Comms., Vol.COM-26, No. 5, pp. 499-506, May 1978.
7. Murano, K. et al., "Multiprocessor Architecture for Voiceband Processing", IEEE Doc. No. CH1435-7/79/0000-0197, pp. 37.3.1-37.3.5, 1979.
8. Kawakami, Y. et al., "A Single-Chip Digital Signal Processor for Voice-band Applications", IEEE Int. Solid-State Circuits Conf., pp. 40-41, 1980.

9. Koya, M. et al., "High-Speed Data Modem Using Digital Signal Processor", IEEE Doc. No. 0536-1486181/0000-0078, pp. 14.7.1-14.7.5, 1981.
10. Watanabe, K. et al., "A 4800 BPS Microprocessor Data Modem", NEC Publication, pp. 47.6.252-47.6.256.
11. Logan, H.L. and Forney, G.D., "A MOS/LSI Multiple-Configuration 9600 BPS Data Modem", source unknown.
12. Stein, P.J. and Gibson, R.W., "A 1200 B/S Single-Chip Microcomputer Data Modem", Phillips Research Lab Publication.
13. Hagiwora, Y., "A High Performance Signal Processor for Speech Synthesis and Analysis", Hitachi Ltd. Publication.
14. Brownlie, J.D. et al., "Custom-Designed Integrated Circuits for Data Modems", IEE 2nd Int. Conf. on the Impact of High Speed and VLSI Technology on Communication Systems, pp. 31-35, 1983.
15. AMD Inc., "The AM7910 FSK Modem", AMD Data Sheet Jan. 1983.
16. EXAR Integrated Systems, Inc., "The XR-14412 FSK Modem System", EXAR Data Sheet.
17. Texas Instruments Ltd., "Telecom. Circuits Data Book", Texas Instruments, 1984.

18. TRW LSI Products., "The TDC1010J Multiplier-Accumulator", TRW Data Sheet 109A-1/79, 1979.
19. Schirm, L., "Multiplier-Accumulator Application Notes", TRW LSI Products, Jan. 1980.
20. TRW Products Division", "The VLSI Data Book", TRW Inc., 1984.
21. Shah, M.B., "Digital Signal Processor" Final Year Undergraduate Project Report, Loughborough University of Technology, 1983.
22. IEE Conference, "The Impact of High-Speed and VLSI Technology on Communication Systems", Conference Publication No. 230, London, 1983.
23. AMD Inc., "AMD29116 and AM29500, High-Performance VLSI for DSP, Array Processing and Intelligent Peripheral Control", AMD Seminar, London, 1983.
24. AMD Inc., "AM29500 Family" AMD Data and Specification, August 1982.
25. Mithani, D., "A Microprogrammed C.P.U. Using AM29116", AMD Inc., Application Note, August 1982.
26. New, B. and Pittroff, L., "Record Signal-Processing Rates Spring from Chip Refinements", Electronics, 28 July, 1982.
27. Product Focus, "Digital Signal Processing", Electronic Eng., pp. 101-124, Feb. 1985.

28. Pickvance, R., "A Single Chip Digital Signal Processor", Electronic Eng., pp. 53-63, Feb. 1985, and pp. 55-62, March 1985.
29. NEC Inc., "The μ PD7720 Digital Signal Processor", Specification and Data Sheet, 1981.
30. STL, "The DSP 128 (CRISP) Digital Signal Processor", Preliminary Specification Sheet, Jan. 1983.
31. Texas Instruments Inc., "The TMS32010 and TMS320M10 High-Performance 16/32-Bit Microcomputers", Texas Preliminary Data Manual, Oct. 1982.
32. Texas Instruments Prod. Div., "Digital Signal Processor, the TMS320 Product Description".
33. Texas Instruments Prod. Div., "Signal Processing Products and Technology, Press Coverage in Leading Electronic Publications", Texas Instruments, 1983.
34. Caudel, E.R. and Hester, R.K., "A Chip Set for Audio Frequency Digital Signal Processing", IEEE Publication CH 1746-7/82/0000.1065, 1982.
35. "Digital Processing of Signals in Communications", IERE Conference, April 1981, Loughborough, Leics.

36. Ungerboeck, G., "Channel Coding with Multilevel/phase Signals", IEEE Trans. Inform. Theory, Vol. IT-28, pp. 55-69, Jan. 1982.
37. Calderbank, A.R. et al., "Upper Bounds on the Minimum Distance of Trellis Codes", Bell Syst. Tech. J., Vol. 62, pp. 2617-2646, Oct. 1983.
38. Motorola Information Systems Ltd., "Introduction to Trellis Coded Modulation", Motorola Publication.
39. CCITT "Proposal for a 14,400 bit/s Modem for use on 4-Wire Telephone Circuits", CCITT Document COM XVII-No. 100-E, 1983.
40. Calderbank, A.R. et al., "Asymptotic Upper Bounds on the Minimum Distance of Trellis Codes", IEEE Trans. on Comms., Vol. COM-33, No.4. pp. 305-309, April, 1985.

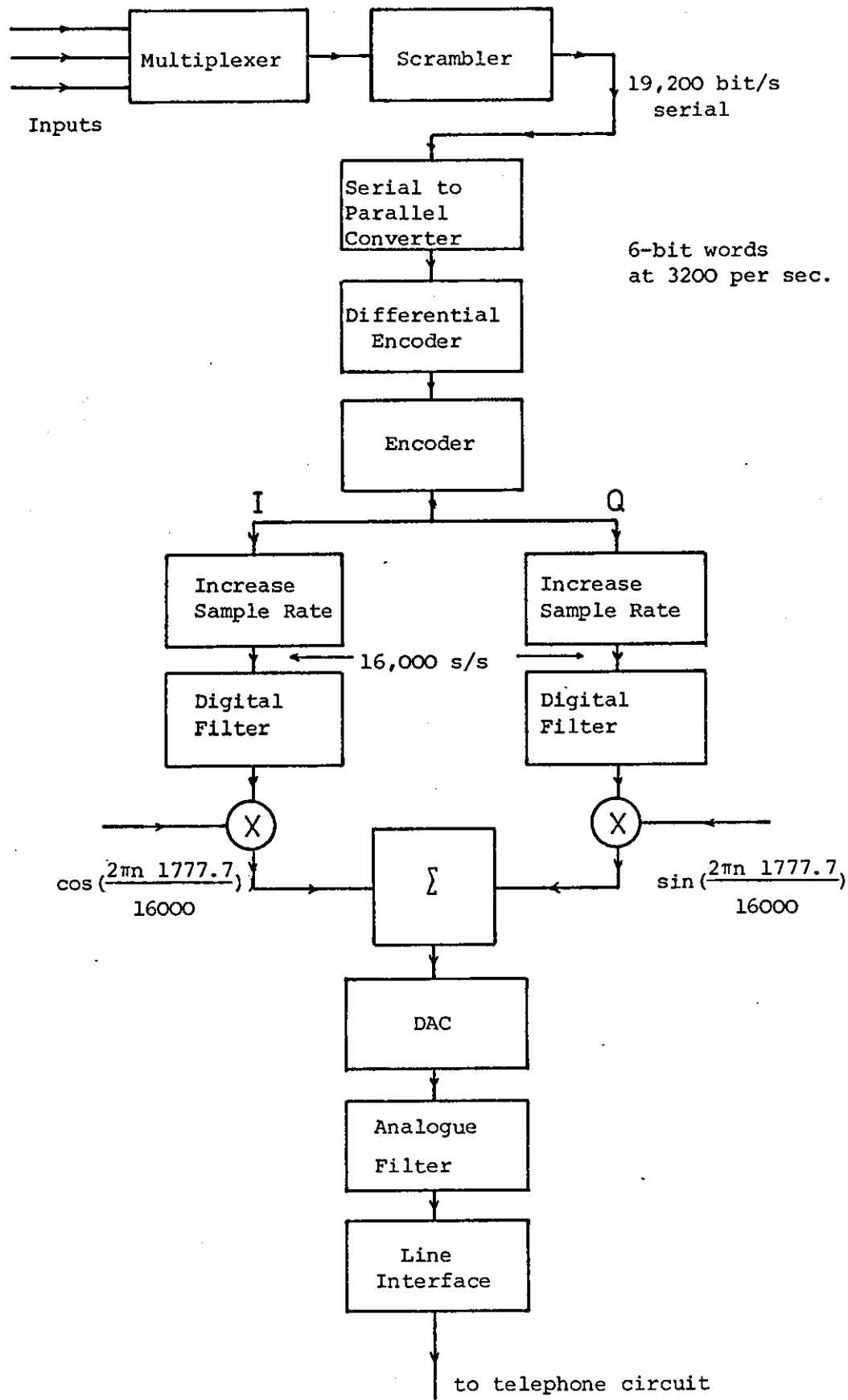


Figure 9.2.1: Block Diagram of Modem Transmitter

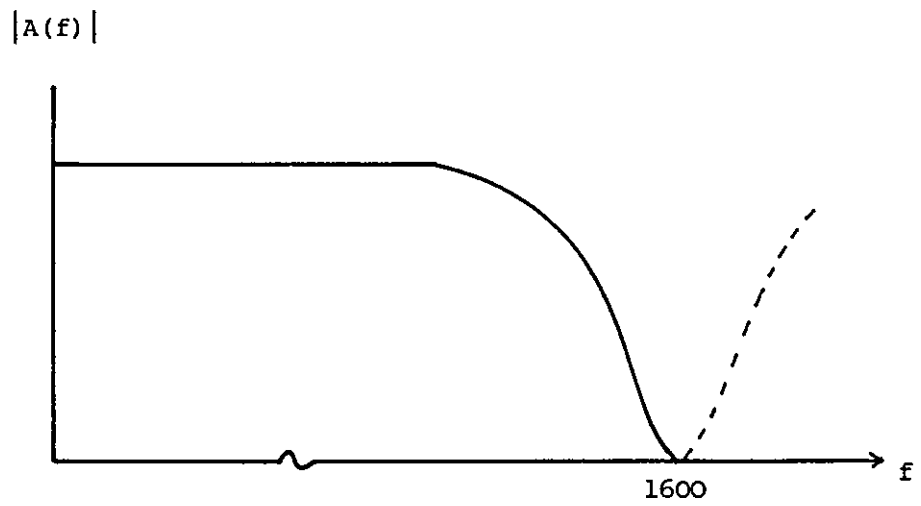


Figure 9.2.2: Amplitude Response of Filter

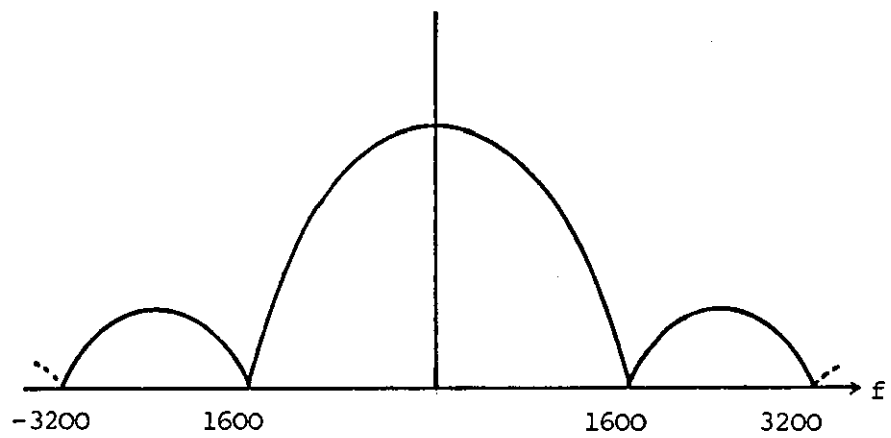


Figure 9.2.3: Amplitude Spectrum at Filter Output

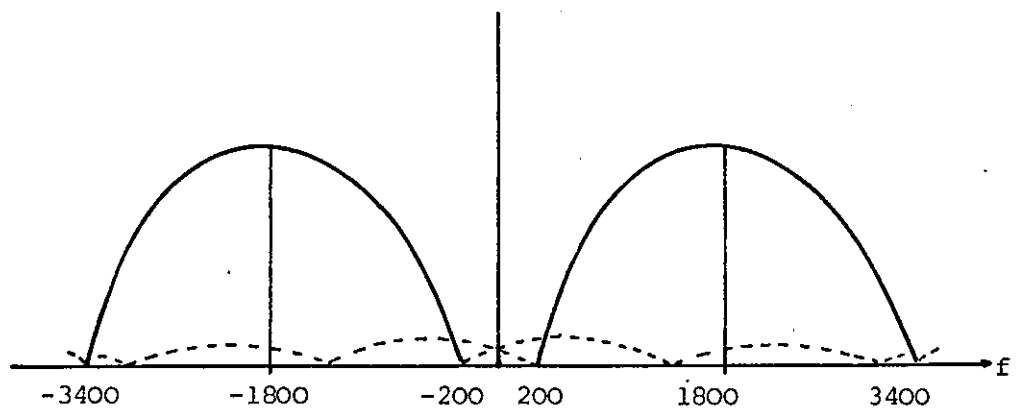


Figure 9.2.4: Spectrum After Modulation

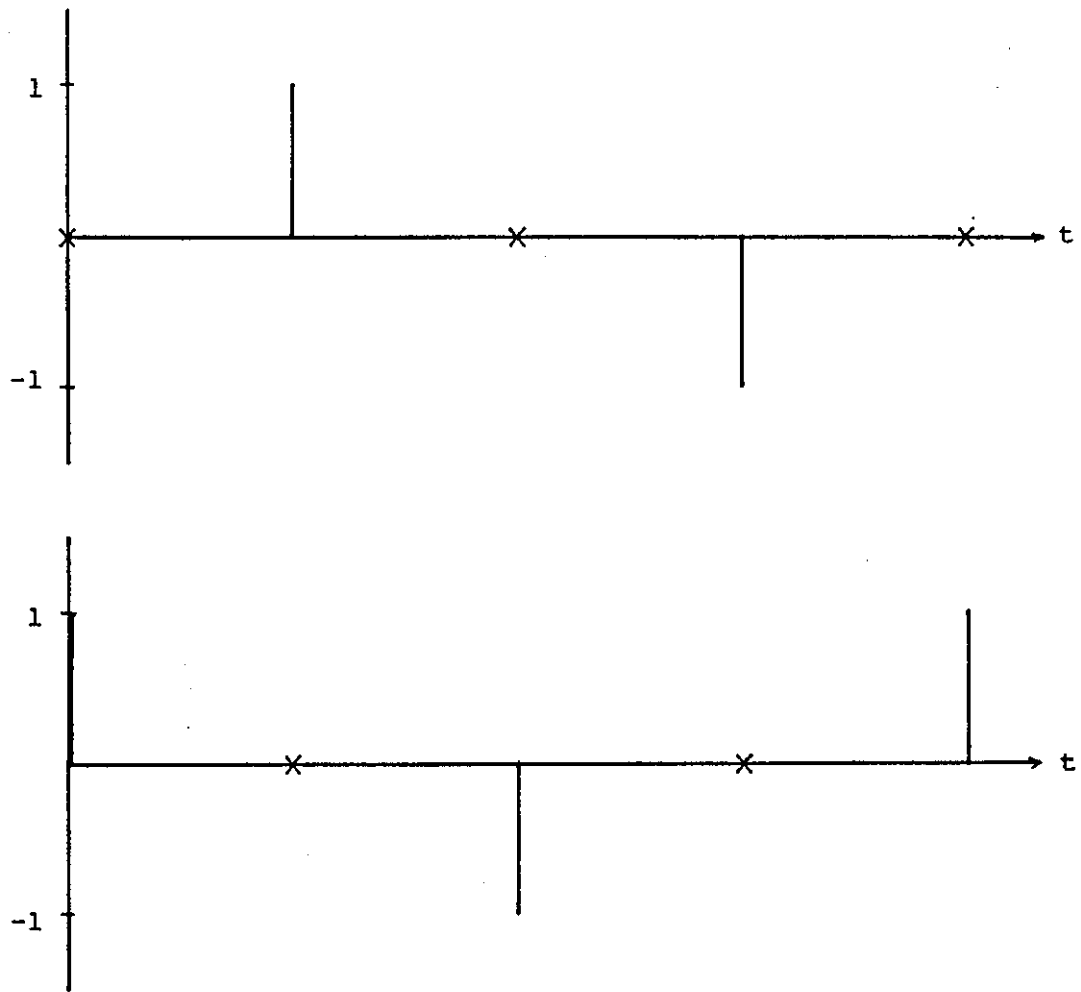


Figure 9.2.5: Samples of a) $\sin(2\pi 1800t)$ at 7200 s/s
b) $\cos(2\pi 1800t)$

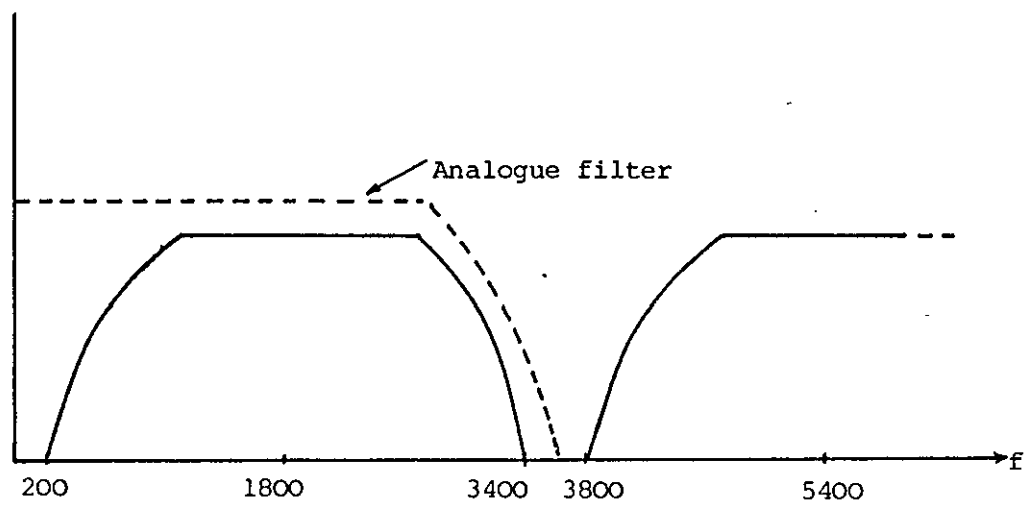
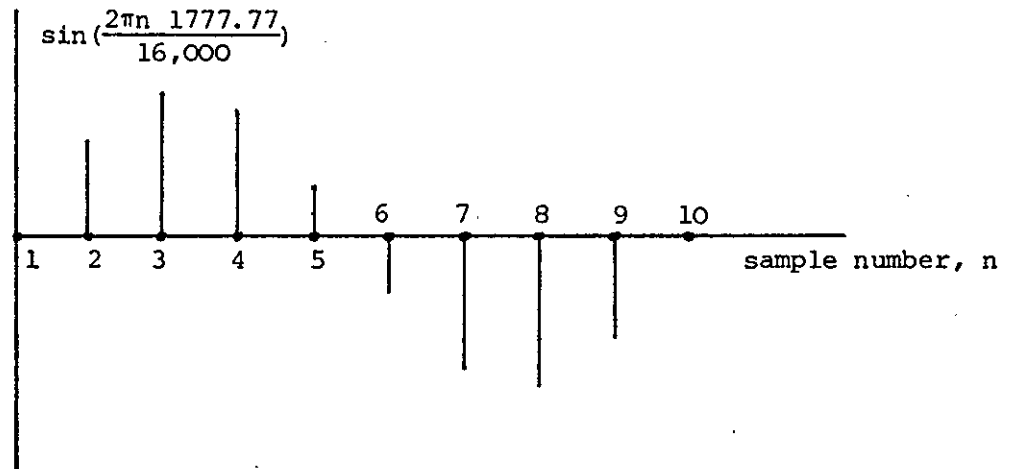
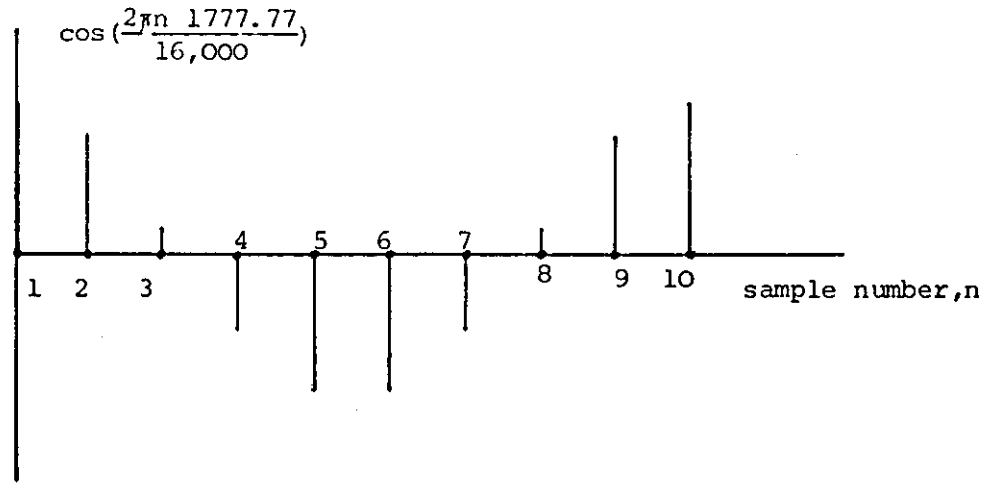


Figure 9.2.6: Modulator Output, Sample rate = 7200 s/s



Sample Number n	$\cos\left(\frac{2\pi n 1777.7}{16,000}\right)$	$\sin\left(\frac{2\pi n 1777.7}{16,000}\right)$
1	1	0
2	0.7660	0.6428
3	0.1736	0.9848
4	-0.5	0.8660
5	-0.9397	0.3420
6	-0.9397	-0.3420
7	-0.5	-0.8660
8	0.1736	-0.9848
9	0.7660	-0.6428

Figure 9.2.7: Samples of 1777.77 Hz Carriers at 16,000 s/s

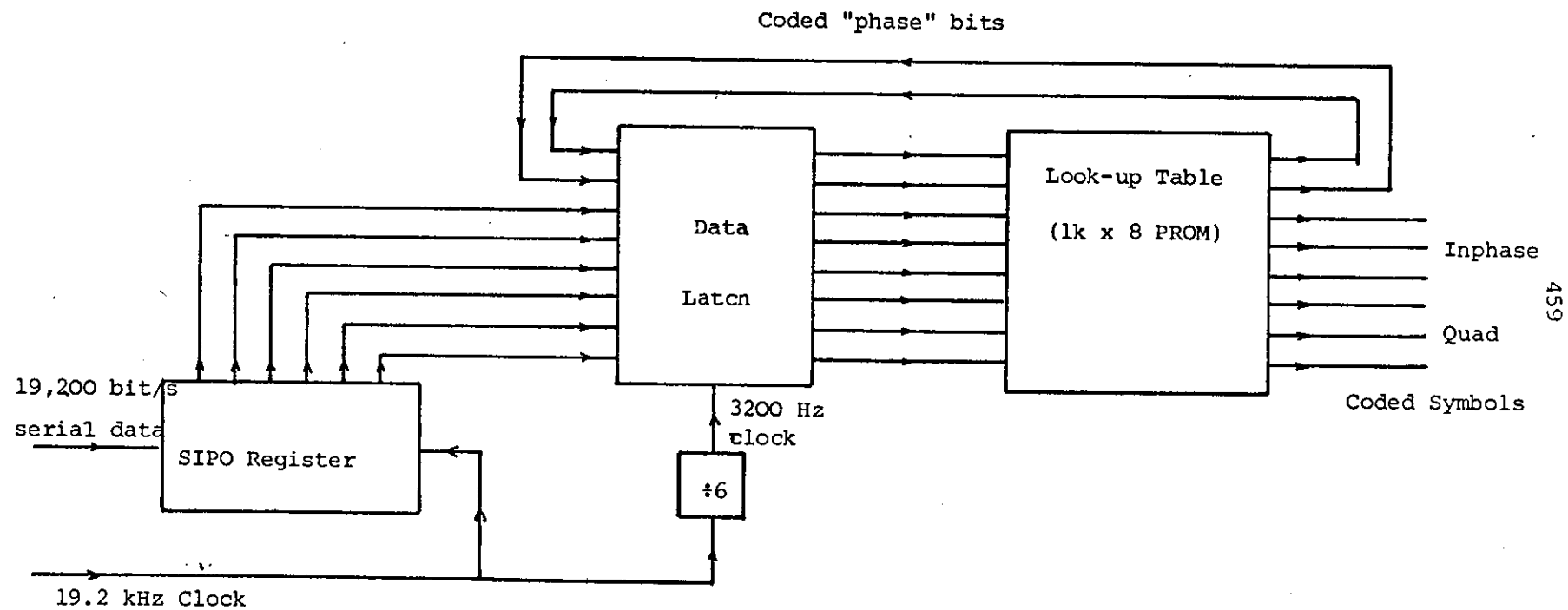


Figure 9.2.8: Transmitter Input Detail

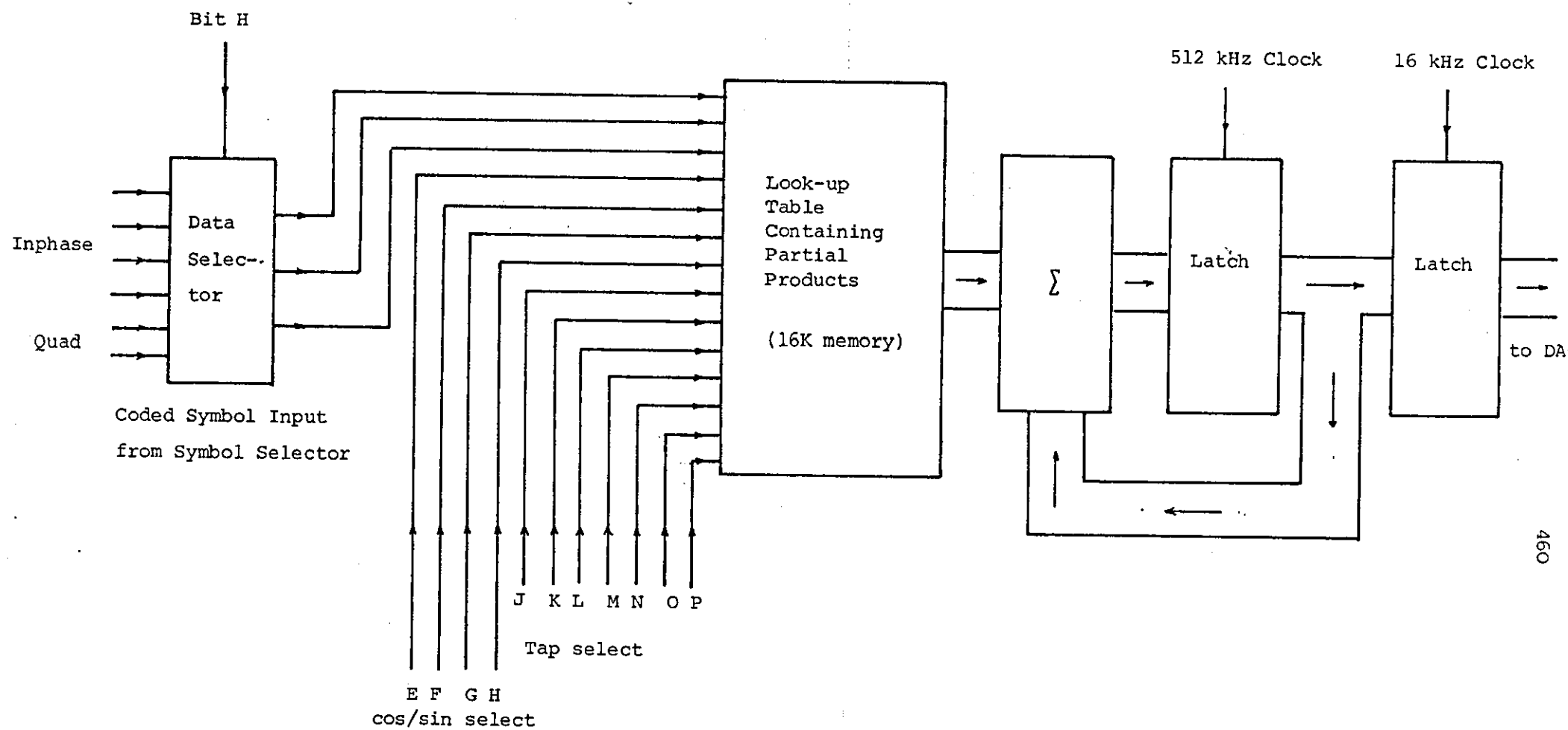
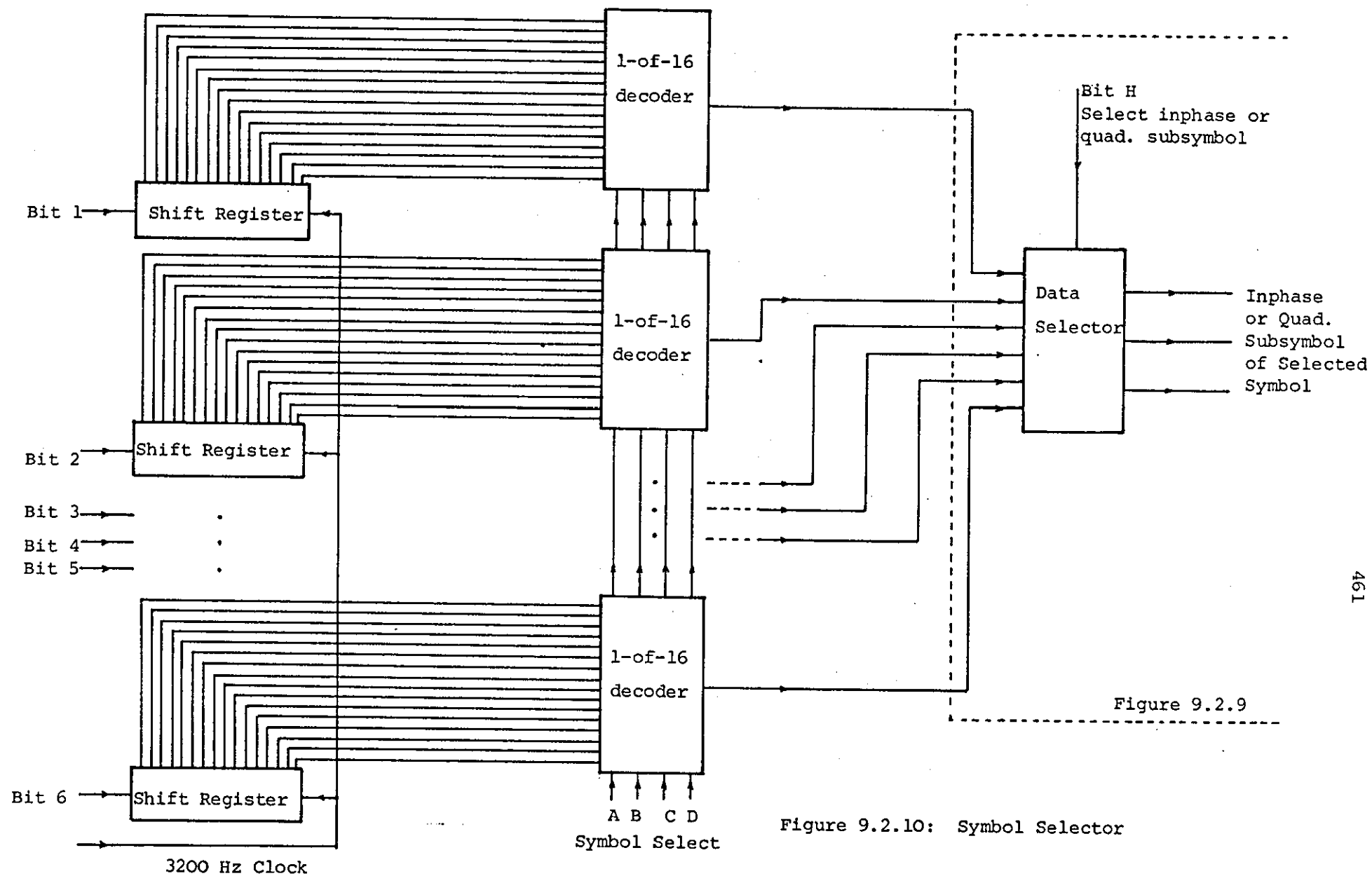


Figure 9.2.9: Filtering, Modulation and Summation



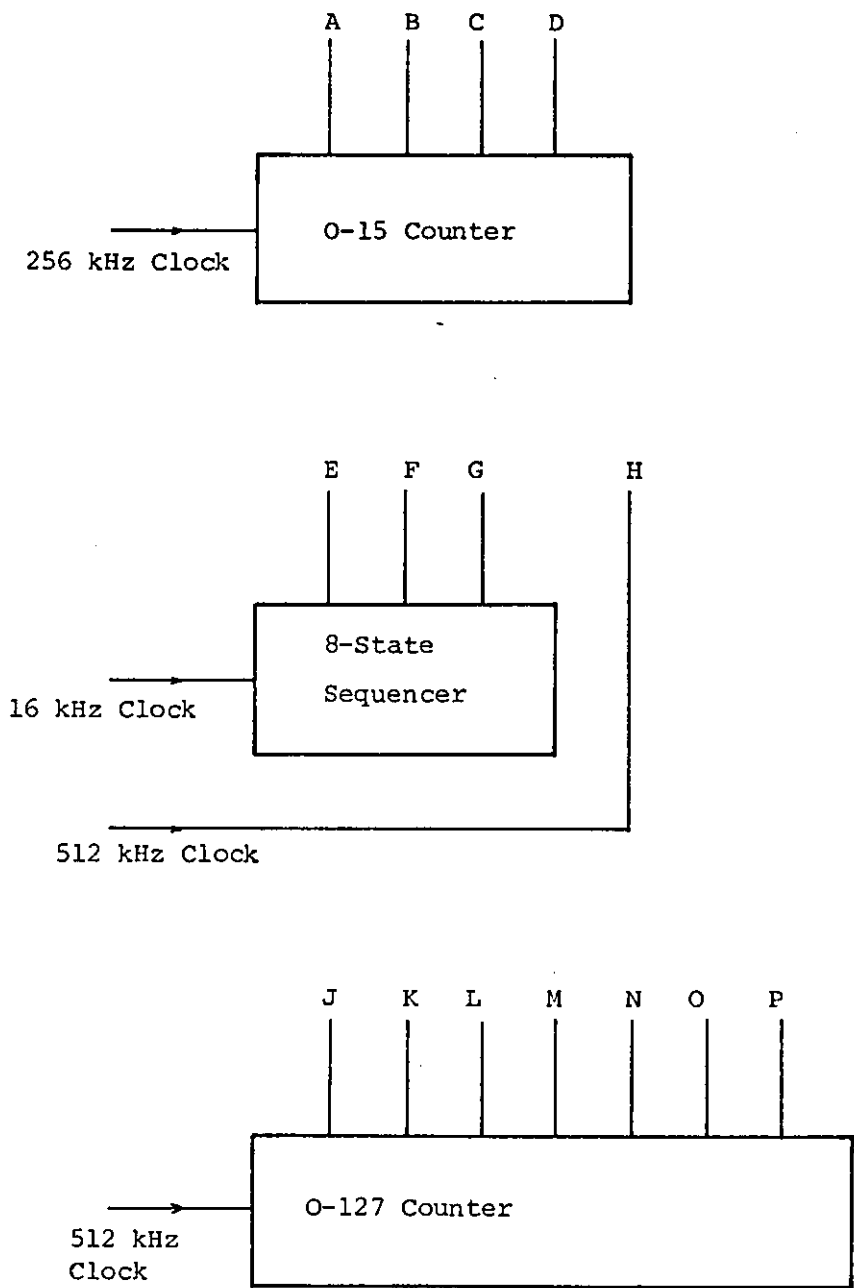


Figure 9.2.11: Sequencer Control Logic

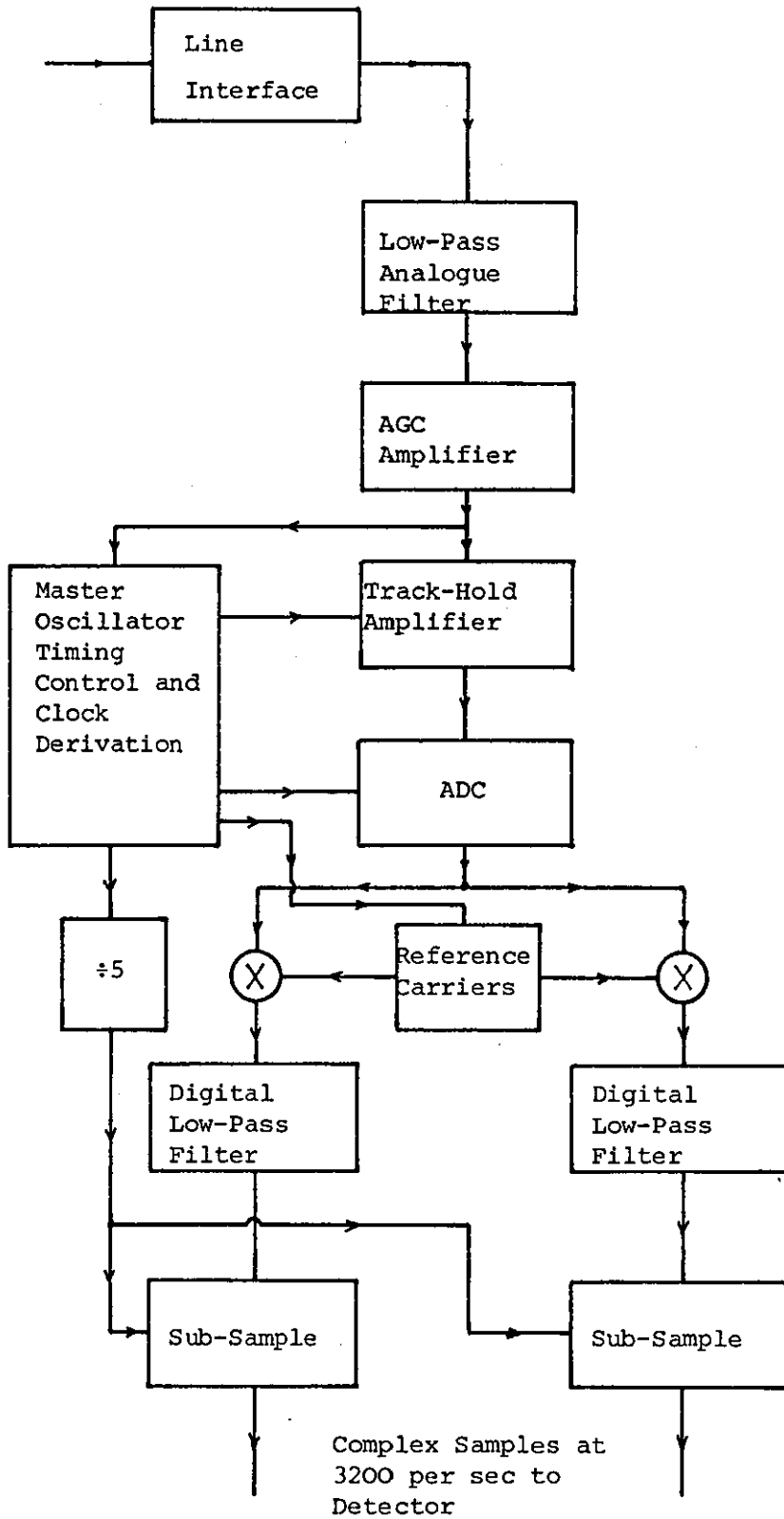


Figure 9.3.1: Modem Receiver, Excluding Detection Stage

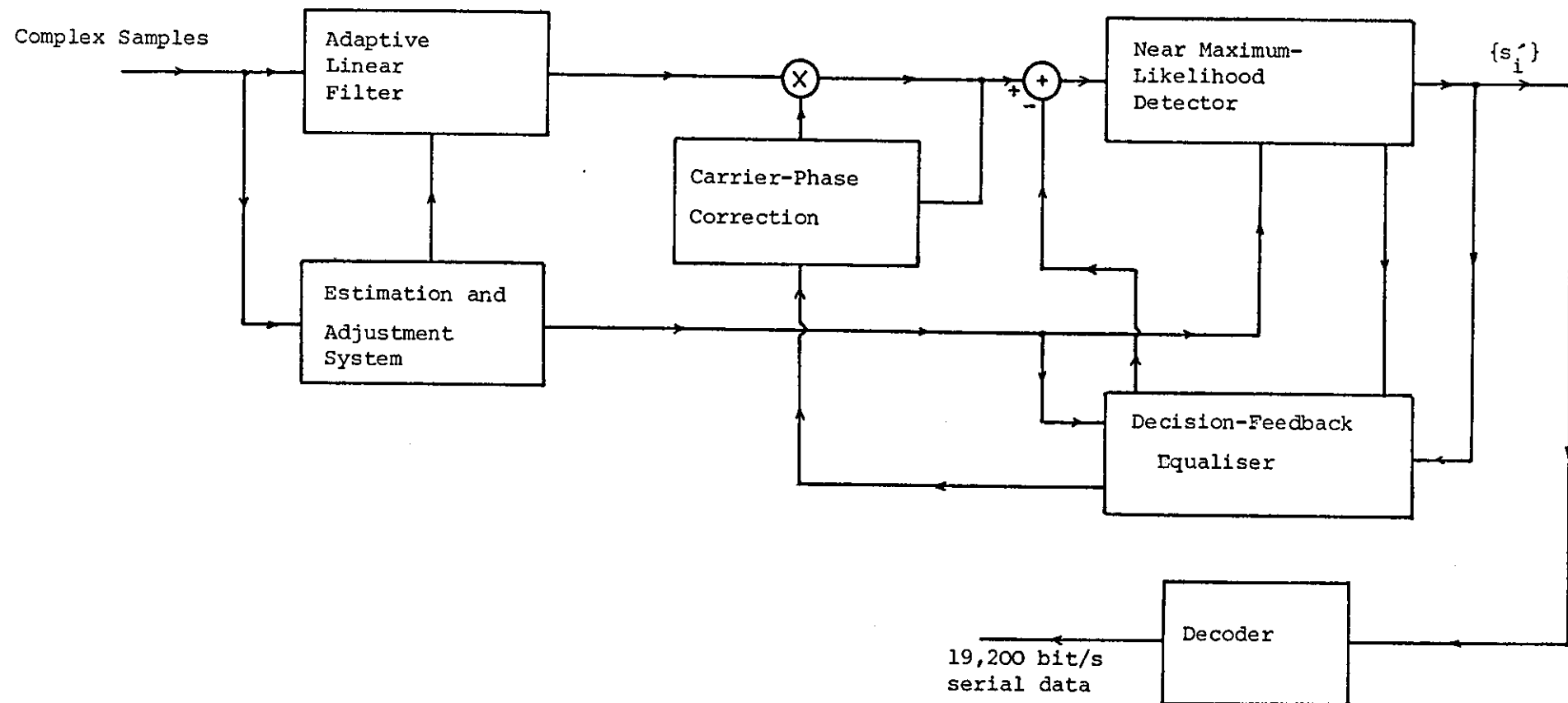


Figure 9.3.2: Modem Receiver; Adaptive Detector

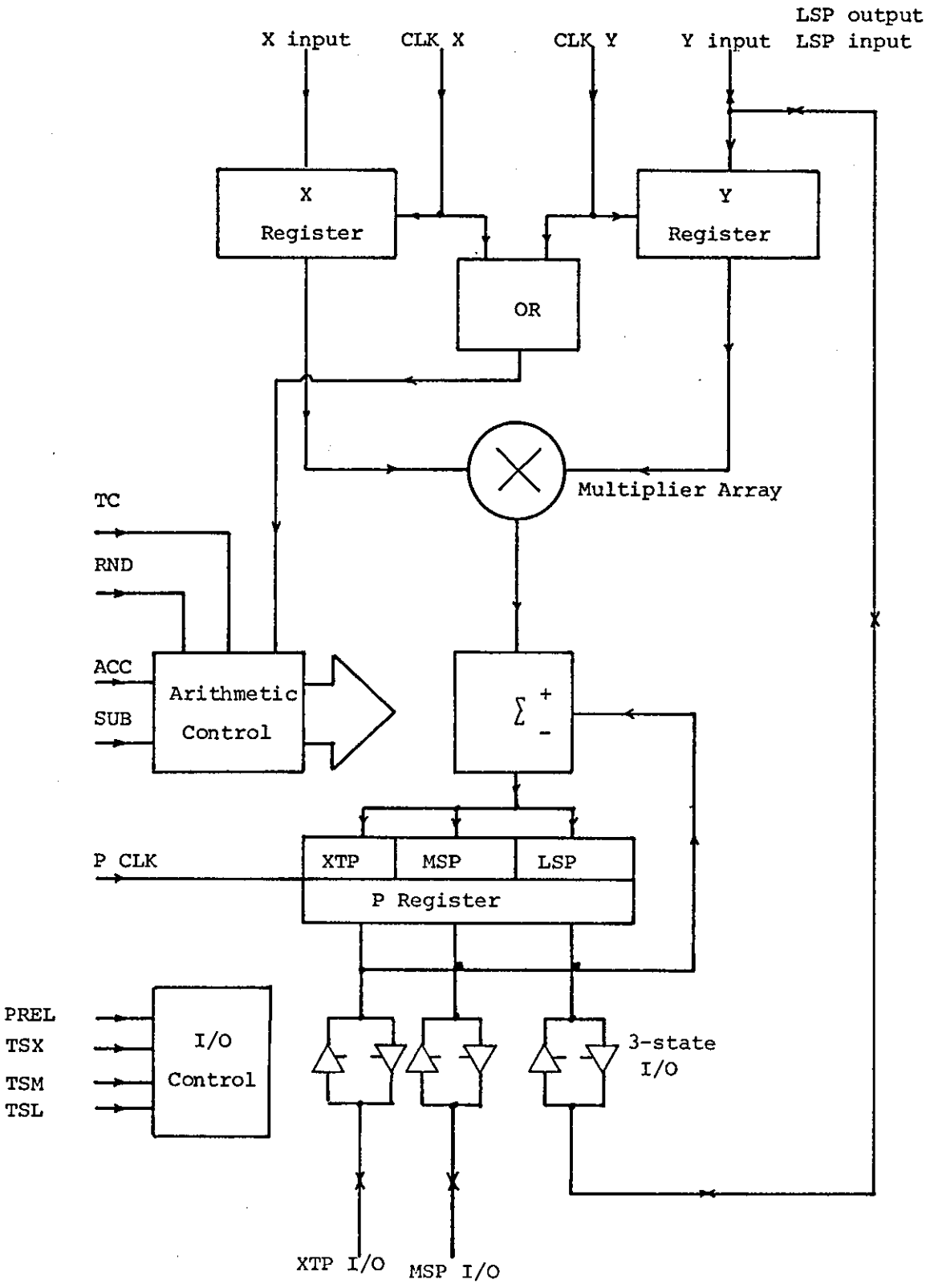


Figure 9.3.3: The TRW TDC1010J MAC

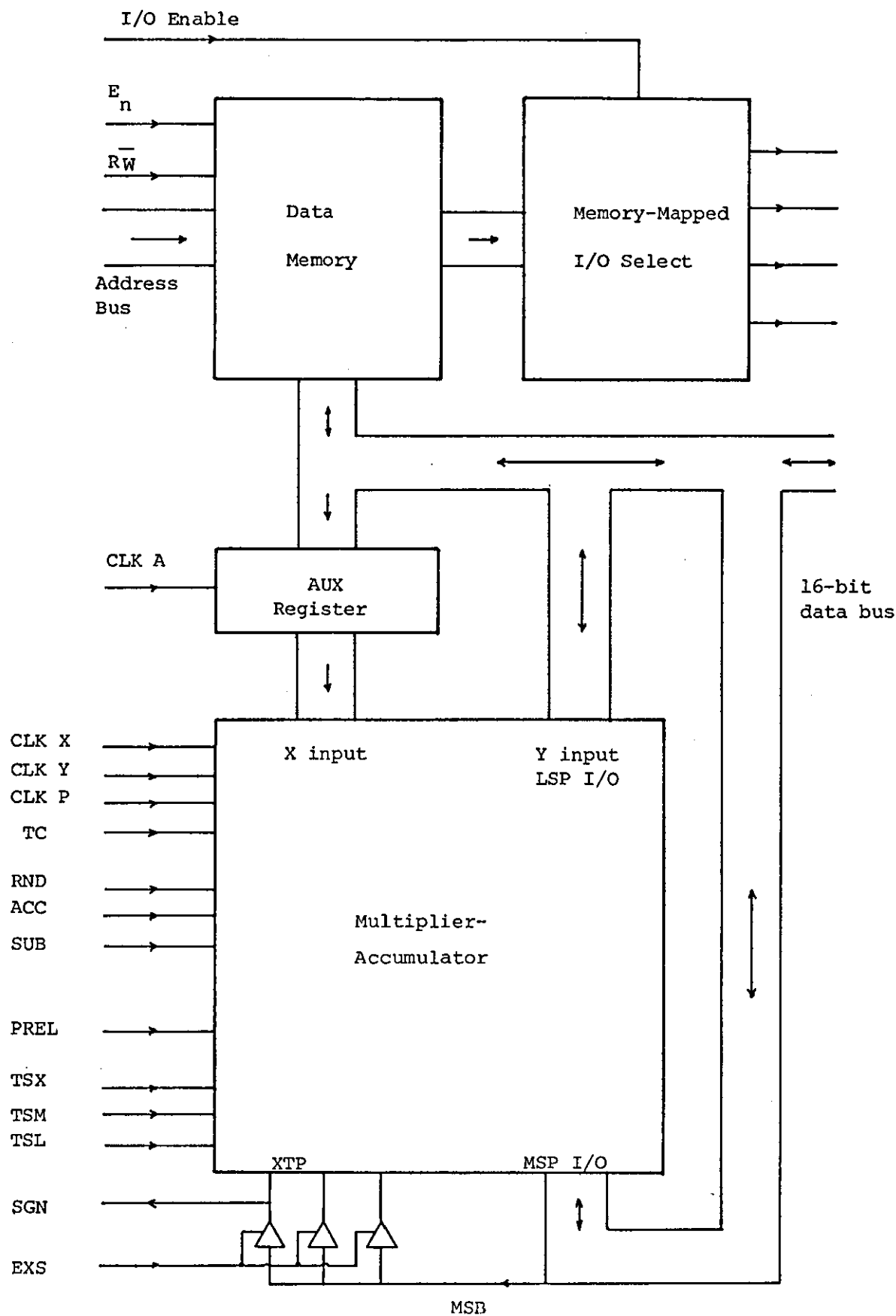


Figure 9.3.4 Architecture of MAC Processor

Instruction Code		Data Transfer on each half cycle	Clocks				P Register Control					Memory	
B ₀	B ₁		A	X	Y	P	PREL	EXS	TSX	TSM	TSL	E _n	$\overline{\text{RW}}$
0	0	MEM → AUX	↑	1	1	1	0	0	0	1	1	e	1
		MEM → Y AUX → X	1	↑	↑	↑	0	0	0	1	1	e	1
0	1	MEM → LSP	1	1	1	↑	1	1	1	e	1	e	1
		MEM → MSP	1	↑	1	↑	1	1	1	1	e	e	1
1	1	LSP → MEM	1	1	1	1	0	0	0	1	e	e	0
		MSP → MEM	1	↑	1	↑	0	0	0	e	1	e	0
1	0	MSP → MEM	1	1	1	1	0	0	0	e	1	e	0
		MEM → Y	1	↑	↑	↑	0	0	0	1	1	e	1

Figure 9.3.5: Control Signals for Data Transfer Instructions

Instruction Code		Arithmetic Operation	MAC Control			
A ₀	A ₁		ACC	SUB	RND	TC
0	0	X.Y → P	0	-	1	1
0	1	X.Y + P → P	1	0	0	1
1	0	X.Y - P → P	1	1	0	1
1	1	X.Y < P; X.Y → P; X.Y > P; X.Y - P → P	$\overline{\text{SGN}}$	1	0	1

Figure 9.3.6: Control Signals for Arithmetic Instructions

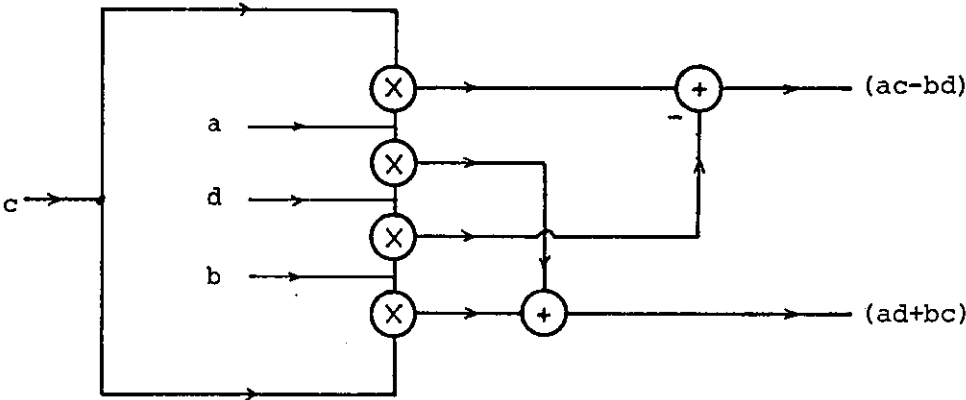


Figure 9.3.7: Diagrammatic Representation of Complex Multiplication, $(a+jb)(c+jd)$

MAC Inputs			MAC Outputs
Data	SUB	ACC	
b,d	0	0	
a,c	1	1	b.d
a,d	0	0	a.c.-b.d→ store
b,c	1	0	a.d
			b.c.-a.d. → store

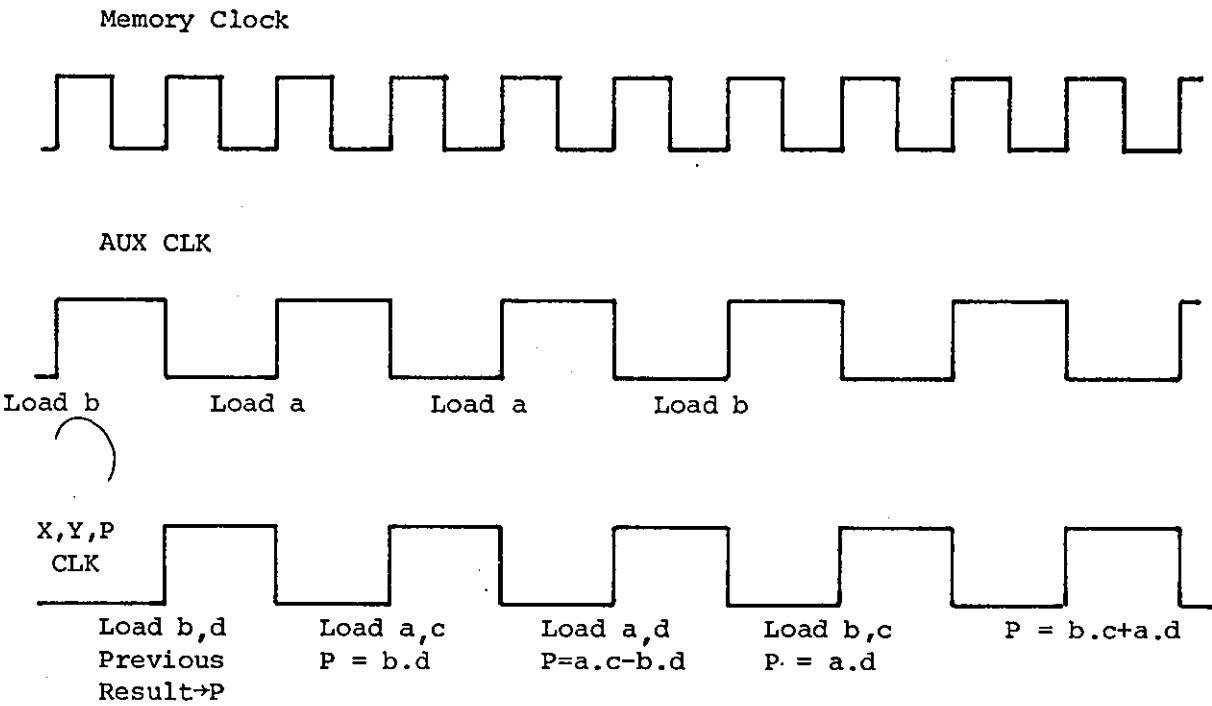


Figure 9.3.8: MAC State Table and Timing Diagram for Complex Multiplication

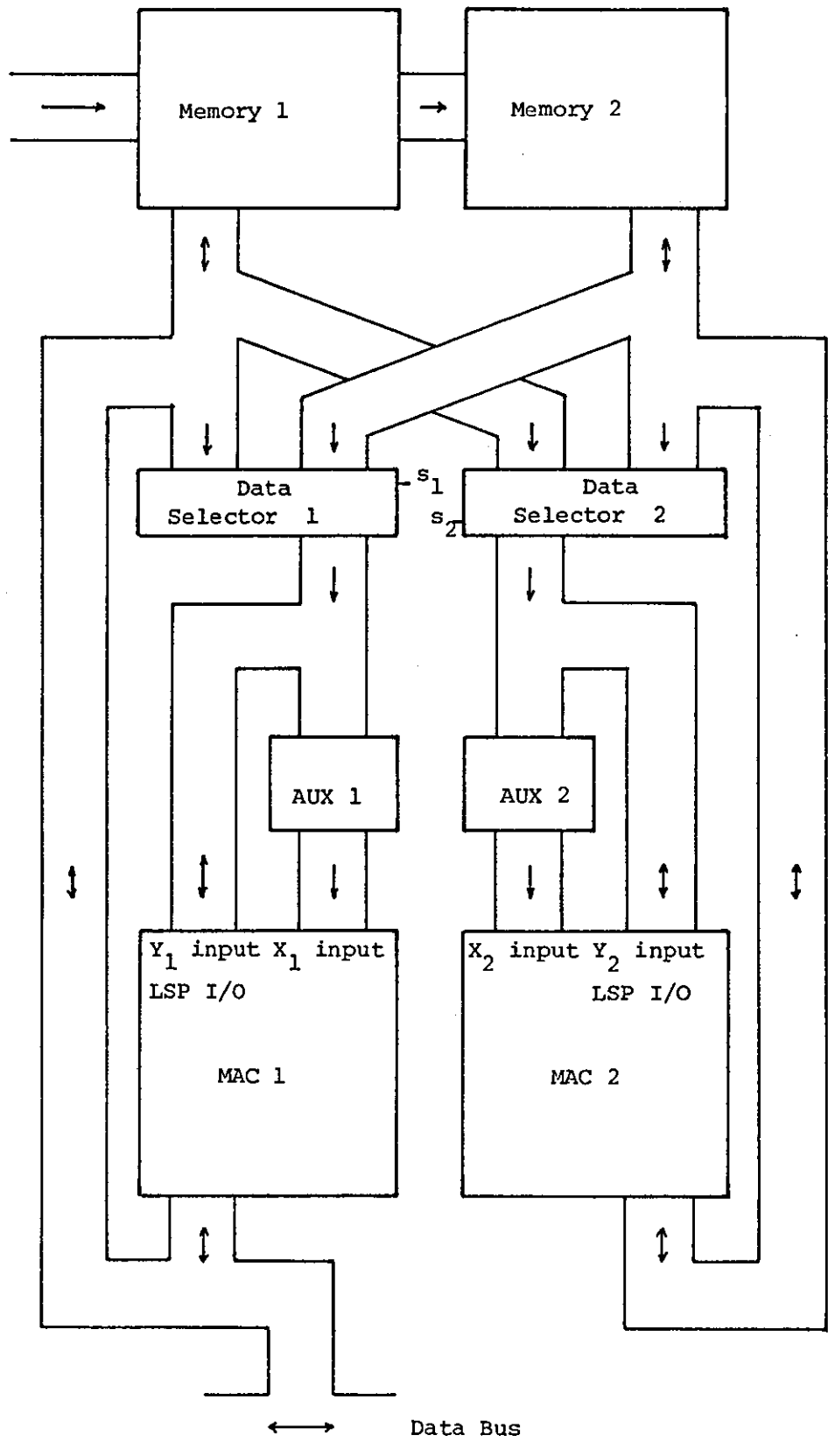


Figure 9.3.9: Extended MAC Processor Architecture

MAC 1 Inputs						MAC 2 Inputs						Outputs	
AUX1	X ₁	Y ₁	ACC	SUB	S ₁	AUX2	X ₂	Y ₂	ACC	SUB	S ₂	MAC 1	MAC 2
b					1	a					1		
	b	d	0	-	1		a	d	0	-	1	Previous result	
a					0	b					0		
	a	c	1	1	0		b	c	1	0	1	b.d	a.c
b ₂					1								
	b ₂	d ₂	0	-	1		a ₂	d ₂	0	-	1	a.c-b.d	b.c+a.d

Figure 9.3.10: State Table for the Extended MAC Processor Implementing Complex Multiplications $(a+jb)(c+jd)$

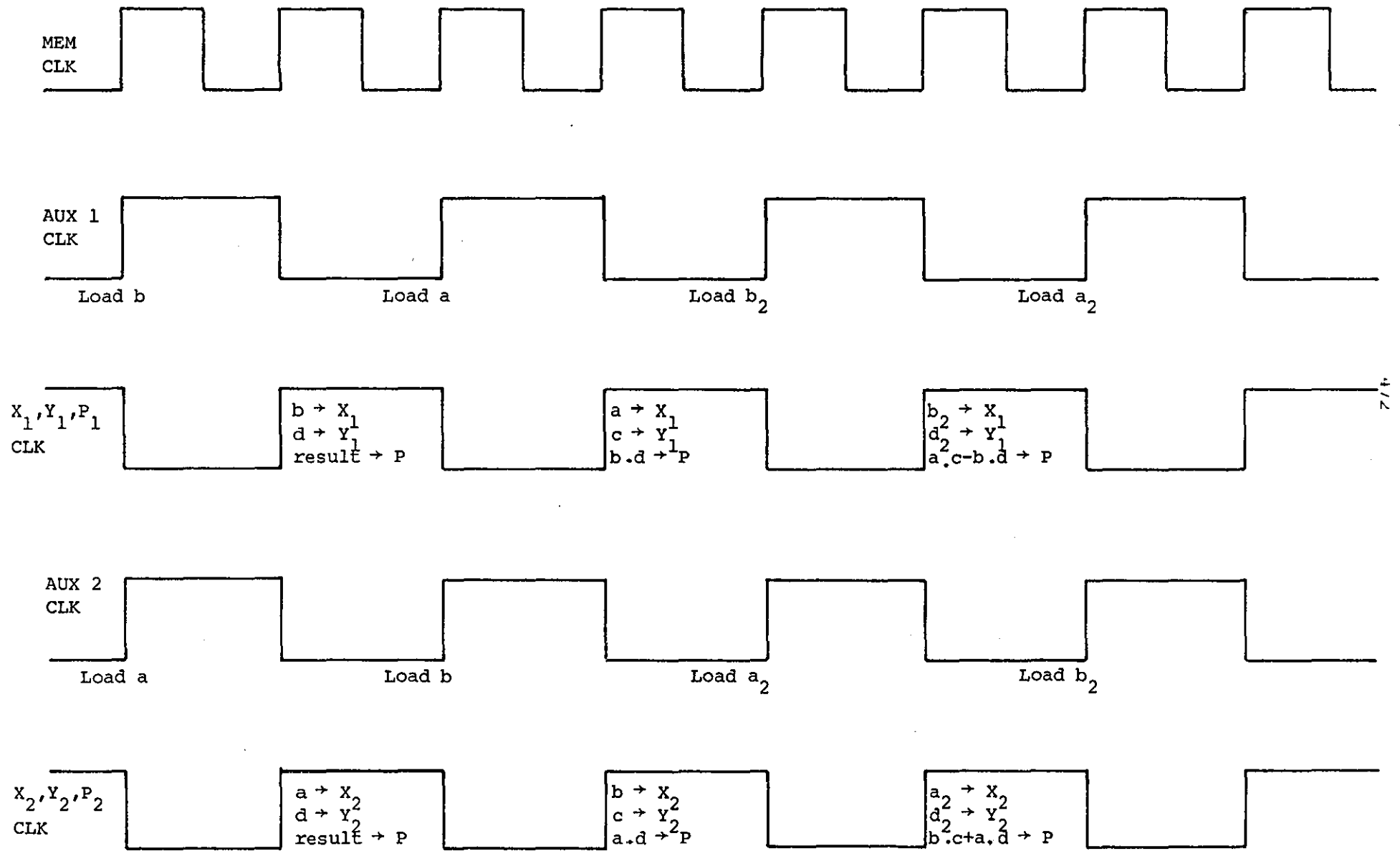


Figure 9.3.11: Processor Timing Diagram for Complex Multiplications

Instruction Code			Arithmetic Operation	Control States							
				MAC 1				MAC 2			
				ACC	SUB	RND	TC	ACC	SUB	RND	TC
0	0	0	$X.Y \rightarrow P$	0	-	1	1	0	-	1	1
0	0	1	$X.Y + P \rightarrow P$	1	0	0	1	1	0	0	1
0	1	0	$X.Y - P \rightarrow P$	1	1	0	1	1	1	0	1
0	1	1	If $X.Y < P$; $X.Y \rightarrow P$ If $X.Y > P$; $X.Y - P \rightarrow P$	\overline{SGN}_1	1	0	1	\overline{SGN}_2	1	0	1
1	0	0	$X.Y \rightarrow P$	0 1	- 1	1 0	1	0 1	- 0	1 0	1
1	0	1	$X.Y + P \rightarrow P$	1 1	1 1	0 0	1	1 1	0 0	0 0	1
1	1	0	$X.Y - P \rightarrow P$	1 1	0 1	0 0	1	1 1	1 0	0 1	1
1	1	1	$X^2 + Y^2 \rightarrow P$	0 1	- 0	1 0	1	0 1	- 0	1 0	1

Figure 9.3.12: Arithmetic Instructions for Extended MAC Processor

10 Conclusions

Consideration of the characteristics of typical switched and private telephone circuits and the types and levels of distortion and other signal impairments introduced by these circuits (Chapter 3), led to the selection of QAM as a suitable modulation method and the derivation of a baseband equivalent system model (Chapters 4 and 5). Results of extensive computer simulation studies based around the model have shown that a 64-point QAM system utilising coding scheme 3 (Chapter 4) and filter set 2 (Chapter 5) will have a superior performance, in terms of tolerance to additive white Gaussian noise, than a 256-point system using filter set 1 or a non linear equaliser (Chapter 6). The 64-point system was designated as system b) and employed a modified adaptive near-maximum likelihood detector (Chapter 6).

Several alternative methods for the adaptive adjustment of the detector have been introduced and studied along with more conventional methods (Chapter 7). The results of the studies suggested the use of alternative schemes 3 and 4 in conjunction with a gradient estimation algorithm. Alternative scheme 3 will require the inclusion of an algorithm for the determination of the complex roots of complex polynomials. Both schemes have been shown to produce the parameters required by an adaptive near-maximum likelihood detector in a considerably shorter time than the conventional methods, but involve extra complexity and therefore require more processing power at the receiver (Chapter 7).

The practical problems of modem synchronisation and implementation have then been considered. Several methods of carrier phase, modem timing and AGC have been presented along the results of computer simulation studies (Chapter 8). Finally, a hardware realisation for the transmitter unit has been presented along with suggestions for a receiver design.

It is recommended that the receiver unit should be implemented using digital signal processing techniques such as the extended MAC system or one (or more) single-chip DSP devices under the management of a programmable controller (Chapter 9).

Appendix 1

The use of an equivalent baseband model to describe the operation of a bandpass transmission system has been used extensively throughout this study. Chapter 2 introduced the concept whereas Chapters 3 and 4 explained the bandpass nature of telephone circuits and presented an intuitive description of the baseband model. Here, the validity of an equivalent baseband description of a bandpass system is explored in a more rigorous manner⁽¹⁾.

Consider first the representation of a bandpass signal by an equivalent baseband signal. A general expression for a bandpass signal can be written as;

$$q(t) = a(t)\cos(2\pi f_c t + \theta(t)) \quad (\text{A1.1})$$

where $a(t)$ is the amplitude and $\theta(t)$ the phase of the modulating signal. For the case of amplitude modulation, $\theta(t)$ will be zero or some constant value and for angle modulation, $a(t)$ will be constant. The frequency f_c is the carrier frequency and it is assumed that the spectrum of $q(t)$ is centred around f_c Hz.

Expansion of equation (A1.1) results in;

$$q(t) = a(t)\cos\theta(t).\cos 2\pi f_c t - a(t)\sin\theta(t)\sin 2\pi f_c t \quad (\text{A1.2})$$

$$\text{If } x(t) \triangleq a(t)\cos\theta(t)$$

$$\text{and } y(t) \triangleq a(t)\sin\theta(t) \quad (\text{A1.3})$$

$$\text{then } q(t) = x(t)\cos 2\pi f_c t - y(t)\sin 2\pi f_c t \quad (\text{A1.4})$$

Because of their relative phase relationships, the signals $x(t)$ and $y(t)$ are called the quadrature components of $q(t)$. Whatever the actual form of modulation used, the two signals $x(t)$ and $y(t)$ represent low frequency components of $q(t)$. For example, if amplitude modulation is employed;

$$\begin{aligned} x(t) &= K a(t) \\ y(t) &= L a(t) \end{aligned} \quad (A1.5)$$

where K and L are constants and $a(t)$ is the baseband signal to be carried by $q(t)$.

From equation (A1.4), $q(t)$ may also be expressed as;

$$q(t) = \operatorname{Re} \left[s(t) e^{j2\pi f_c t} \right] \quad (A1.6)$$

where $s(t) \triangleq x(t) + jy(t)$

and $\operatorname{Re}[\cdot]$ denotes the real part of the bracketed term.

The spectrum of $q(t)$ is obtained by taking its Fourier Transform, viz;

$$Q(f) = \int_{-\infty}^{\infty} q(t) e^{-j2\pi f t} dt \quad (A1.7)$$

$$Q(f) = \int_{-\infty}^{\infty} \left[\operatorname{Re} \left[s(t) e^{j2\pi f_c t} \right] e^{-j2\pi f t} dt \right] \quad (A1.8)$$

$$\text{Since } \operatorname{Re}[s(t)] = \frac{1}{2} [s(t) + s^*(t)]$$

$$\text{and } \operatorname{Re}[s(t) e^{j2\pi f t}] = \frac{1}{2} [s(t) e^{j2\pi f t} + s^*(t) e^{-j2\pi f t}]$$

then equation (A1.8) may be written as,

$$Q(f) = \frac{1}{2} \left[\int_{-\infty}^{\infty} s(t) e^{j2\pi f t} e^{-j2\pi f t} dt + \int_{-\infty}^{\infty} s^*(t) e^{-j2\pi f t} e^{-j2\pi f t} dt \right] \quad (A1.9)$$

that is,

$$Q(f) = \frac{1}{2} [S(f - f_c) + S^*(-f - f_c)] \quad (A1.10)$$

where $S(f)$ is the Fourier Transform of $s(t)$ and $*$ denotes the complex conjugate.

Equation (A1.10) further illustrates the low frequency nature of $s(t)$ and hence $x(t)$ and $y(t)$ since the spectrum of $s(t)$ is clearly in the vicinity of $f=0$. For this reason, the signal $s(t)$ is called the

(complex) baseband signal. It is worth noting that if $s(t)$ is a purely real signal modulating a cosinusoidal carrier waveform, then equation A(1.10) reduces to the well-known Modulation or Frequency Shift Theorem of Fourier Transform theory⁽²⁾.

Consider next the representation of a linear baseband channel with a real impulse response $h(t)$ and an associated transfer function $H(f)$. Defining $C(f-f_c)$ as;

$$C(f-f_c) = \begin{cases} H(f), & f > 0 \\ 0, & f < 0 \end{cases} \quad (\text{A1.11})$$

$$\text{then } C^*(-f-f_c) = \begin{cases} 0 & f > 0 \\ H^*(-f) & f < 0 \end{cases} \quad (\text{A1.12})$$

then, since for a real impulse response,

$$H^*(-f) = H(f) \quad (\text{A1.13})$$

$$H(f) = C(f-f_c) + C^*(-f-f_c) \quad (\text{A1.14})$$

$$\text{and so, } h(t) = \int_{-\infty}^{\infty} H(f) e^{j2\pi f_c t} df$$

$$= c(t) e^{j2\pi f_c t} + c^*(t) e^{-j2\pi f_c t}$$

$$h(t) = 2\text{Re}[c(t) e^{j2\pi f_c t}] \quad (\text{A1.15})$$

where $c(t)$ is the equivalent (complex) impulse response referred to baseband.

Given that both bandpass signals and systems may be represented by baseband equivalents, it now remains to show that the response of a bandpass system to a bandpass signal may be obtained from knowledge of the baseband input signal and the equivalent baseband impulse response. If $p(t)$ is the output of the bandpass system, then;

$$p(t) = \text{Re}[x(t) e^{j2\pi f_c t}] \quad (\text{A1.16})$$

where $r(t)$ is the baseband equivalent of the output bandpass signal.

$p(t)$ is also related to the input signal, $q(t)$ and to the impulse response of the system, $h(t)$, by the convolution integral;

$$p(t) = \int_{-\infty}^{\infty} q(\tau)h(t-\tau)d\tau \quad (\text{A1.17})$$

In the frequency domain;

$$P(f) = Q(f)H(f) \quad (\text{A1.18})$$

Substituting for $Q(f)$ from equation (A1.10) and for $H(f)$ from equation (A1.14) gives;

$$P(f) = \frac{1}{2}[S(f-f_c) + S^*(-f-f_c)][C(f-f_c) + C^*(-f-f_c)] \quad (\text{A1.19})$$

For the signals and systems considered here, the bandwidth of the baseband equivalent processes are in all cases less than the carrier frequency and so $S(f-f_c) = C(f-f_c) = 0$, for $f < 0$. Equation (A1.19) therefore simplifies to;

$$P(f) = \frac{1}{2}[S(f-f_c).C(f-f_c) + S^*(-f-f_c)C^*(-f-f_c)] \quad (\text{A1.20})$$

$$= \frac{1}{2}[R(f-f_c) + R^*(-f-f_c)] \quad (\text{A1.21})$$

$$\text{where } R(f) = S(f).C(f) \quad (\text{A1.22})$$

By comparison with equations (A1.17 - A1.19) the time domain representation of the output baseband signal, $r(t)$, is therefore;

$$r(t) = \int_{-\infty}^{\infty} s(\tau)c(t-\tau)d\tau \quad (\text{A1.23})$$

which shows that the equivalent baseband output signal from the bandpass channel is given by the convolution of the baseband input signal and the equivalent baseband impulse response of the bandpass channel.

References

1. Proakis, J.G., "Digital Communications", McGraw-Hill Inc. 1983.
2. Schwartz, M., "Information Transmission, Modulation and Noise", McGraw-Hill Inc., Third Edition, 1980.

Appendix 2

Chapter 4 and Appendix 1 have shown that a bandpass system may be modelled by an equivalent baseband system for reasons of mathematical convenience. However, due to the two-sided nature of bandpass signals about the carrier frequency and the one-sided nature of baseband signals, it may be necessary to alter signal power levels and/or noise power levels to maintain a given signal-to-noise ratio across the two models. The following analysis assumes a QAM system operating over a bandpass channel which is modelled by the appropriate 2-dimensional baseband system as described in Chapter 4.

From Figure 4.3.1 and equations (4.3.17) and (4.3.23), the noise component present on the received baseband signal is;

$$u(t) = \sqrt{2} \left[(n(t)e^{-j2\pi f_c t}) * (c(t)e^{-j2\pi f_c t})e^{-j\phi} \right] * p(t) \quad (A2.1)$$

where,

$n(t)$ is a white Gaussian process with zero mean and 2-sided power spectral density of $\frac{1}{2} N_0$ W/Hz,

$c(t)$ represents the impulse response of the receiver input bandpass filter,

$p(t)$ represents the impulse response of the low-pass post-demodulation filters,

f_c is the carrier frequency,

ϕ is some phase error between the signal's carriers and the receiver's reference carriers.

From equation (4.3.24), the power spectral density of $u(t)$ is;

$$|U(f)|^2 = N_0 |C(f+f_c)|^2 |P(f)|^2 \quad (A2.2)$$

Consequently, $u(t)$ is a low-pass process and the effect of the demodulator alone has been to multiply the noise power spectral density by a factor of 2. It is shown elsewhere that the noise power spectral density of the individual real and imaginary parts of $u(t)$ will be $\frac{1}{2}N_0$ W/Hz⁽¹⁻⁶⁾. As far as the model used in computer simulations are concerned, the noise, $n(t)$, used in the bandpass model and $u(t)$, used in the baseband model, will have an equivalent effect on the system provided that the two quadrature components of $u(t)$ are generated from independent white Gaussian noise generators with zero mean values and 2-sided power spectral densities of $\frac{1}{2}N_0$ W/Hz and are both passed through a filter identical to the baseband equivalent of the combined receiver filters, before being added to the received signal at the output of the baseband model.

It has also been shown⁽⁵⁾, that the coherent demodulator does not change the energy of the signal component existing in the two models. Consequently, the same signal-to-noise ratio for the two models may be achieved by applying the symbols as detailed in Chapter 4, but with the necessary conditions mentioned above applied to the complex noise samples.

References

1. Schwartz, M., "Information Transmission, Modulation and Noise", McGraw Hill Inc., Third Edition, 1980.
2. Clark, A.P., "Principles of Digital Data Transmission", Pentech Press, 1976.

3. Clark, A.P., "Advanced Data-Transmission Systems", Pentech Press, 1977.
4. Harvey, J.D., "Synchronisation of a Synchronous Modem", SERC Report No. GR/A/1200.7, Dec. 1980.
5. Najdi, H.Y., "Digital Data Transmission Over Voice Channels", Ph.D. Thesis, Loughborough University of Technology, 1982.
6. McVerry, F., "High Speed Data Transmission over HF Radio Links", Ph.D. Thesis, Loughborough University of Technology, 1982.

Appendix 3

Signal-to-noise ratio definition

The signal-to-noise ratio is defined in this study as

$$\text{SNR} = 10 \log_{10} \left[\frac{E_B}{N_0} \right] \quad (\text{A3.1})$$

where E_B is the average energy per bit at the output of the transmitter and N_0 is the 2-sided noise power spectral density at the input to the receiver in the equivalent baseband model.

Since the properties of white Gaussian noise sources available as computer subroutines are usually presented in terms of the mean value and the variance of the distribution, it is necessary to interpret noise power spectral density in terms of the noise variance.

Figure A3.1 shows the baseband equivalent model of the data transmission system where it is assumed that;

- a) The input signal is represented by a stream of weighted impulses, the appropriate change having been made to the transmitter filter to accommodate this representation,
- b) $A(f)$, $a(t)$ represent the transfer function and impulse response, respectively, of the combined baseband referred transmitter filter,
- c) $B(f)$, $b(t)$ represent similar expressions for the combined baseband referred receiver filter,
- d) The filtering present in the baseband model is equally shared between the transmitter and receiver filters such that

$$A(f) = B(f) = H^{\frac{1}{2}}(f)$$

where $H(f)$ is the transfer function of the combined transmitter and receiver filters. Furthermore, $H^{\frac{1}{2}}(f)$ is assumed to satisfy Nyquist's vestigial symmetry theorem⁽¹⁾.

Under these conditions, the receiver filter is matched to the output signal from the transmitter filter and the signal-to-noise ratio at the output of the receiver filter is maximised^(1,2).

The waveform of the i^{th} transmitted signal element is therefore,

$$s_i a(t-iT) \quad (\text{A3.2})$$

and the energy of the signal element at the input to the transmission path is,

$$\epsilon_i = \int_{-\infty}^{\infty} |s_i a(t-iT)|^2 dt \quad (\text{A3.3})$$

$$\epsilon_i = |s_i|^2 \int_{-\infty}^{\infty} |a(t-iT)|^2 dt \quad (\text{A3.4})$$

$$\text{If } \int_{-\infty}^{\infty} |a(t-iT)|^2 dt = 1$$

which implies,

$$\int_{-\infty}^{\infty} |a(t)|^2 dt = \int_{-\infty}^{\infty} |A(f)|^2 df = 1 \quad (\text{A3.5})$$

$$\text{then } \epsilon_i = |s_i|^2 \quad (\text{A3.6})$$

The energy of any sequence of n signal elements, $\sum_{i=1}^n s_i a(t-iT)$ at the input to the transmission path is,

$$\epsilon_n = \int_{-\infty}^{\infty} \left| \sum_{i=1}^n s_i a(t-iT) \right|^2 dt \quad (\text{A3.7})$$

Now, if $h(t)$ is the impulse response of the combined filters, then due to the above assumptions,

$$h(iT) = 0, \quad i \neq 0 \quad (\text{A3.8})$$

hence, since $a(t)$ is a real and even function of time,

$$h(iT) = \int_{-\infty}^{\infty} a(t-kT) a(t-(k+i)T) dt = 0 \quad (\text{A3.9})$$

which implies that the signal elements themselves are orthogonal waveforms. Consequently, from equation (A3.8),

$$\left| \sum_{i=1}^n s_i a(t-iT) \right|^2 = \sum_{i=1}^n |s_i a(t-iT)|^2 \quad (\text{A3.10})$$

and so equation (A3.7) becomes

$$\epsilon_n = \sum_{i=1}^n |s_i|^2 \int_{-\infty}^{\infty} |a(t-iT)|^2 dt \quad (\text{A3.11})$$

which, from equation (A3.5),

$$\epsilon_n = \sum_{i=1}^n |s_i|^2 \quad (\text{A3.12})$$

The average energy per symbol is,

$$\epsilon = \frac{1}{n} \sum_{i=1}^n |s_i|^2 = \overline{|s_i|^2} \quad (\text{A3.13})$$

and if B/S represents the number of bits per symbol, then the average energy per bit is,

$$E_B = \frac{\overline{|s_i|^2}}{B/S} \quad (\text{A3.14})$$

The additive white Gaussian noise introduced at the output of the transmission path has zero mean and 2-sided noise power density of N_0 W/Hz. The average noise power at the output of the receiver filter is equal to the variance of the noise samples and is;

$$\sigma^2 = \int_{-\infty}^{\infty} N_0 |B(f)|^2 df \quad (\text{A3.15})$$

If the receiver filter is normalised such that

$$\int_{-\infty}^{\infty} |B(f)|^2 df = 1 \quad (\text{A3.16})$$

then from equation (A3.15),

$$\sigma^2 = N_0 \quad (\text{A3.17})$$

Using the results of equations (A3.14) and (A3.17), equation (A3.1) may now be written as;

$$\text{SNR} = 10 \log_{10} \left[\frac{|s_i|^2}{B/S} \cdot \frac{1}{\sigma^2} \right] \quad (\text{A3.18})$$

The above equation gives the expression used to calculate signal-to-noise ratio throughout this study. In all simulation tests the transmitter and receiver filters shown in fig. Q3.1 are normalised such that;

$$\begin{aligned} \int_{-\infty}^{\infty} |a(t)|^2 dt &= \int_{-\infty}^{\infty} |A(f)|^2 df = 1 \\ \int_{-\infty}^{\infty} |b(t)|^2 dt &= \int_{-\infty}^{\infty} |B(f)|^2 df = 1 \end{aligned} \quad (\text{A3.19})$$

Clearly, for practical filters, the conditions of equation (A3.19) will not be met directly and so the characteristics of the filters will have to be changed accordingly. As shown in Chapter 5, the normalising process does not change the shape of the filter characteristics but does introduce gain into the system. For the case of the receiver filter, the gain does not alter the signal-to-noise ratio since both the signal and noise components pass through the filter. Also, since signal-to-noise ratio is defined in terms of signal energy per bit at the output of the transmitter filter, no adjustment to the signal-to-noise level is necessary due to the filter satisfying equation (A3.19).

One other important point to make about signal-to-noise ratio calculations concerns the form of baseband model used in actual computer simulations. In general, the model of Figure (A3.2) is used in preference to the model of Figure (A3.1). The overall sampled impulse response of the baseband system can be calculated prior to run time which reduces the number of multiplications required within the program's main iterative loop. However, noise is now added to the received baseband signal at

the output of the receiver filter. Clearly, to correctly model noise correlation and allow for scaling factors caused by filter normalisation, the white noise samples must be fed through a filter identical to the receiver filter, as shown in Figure (A3.2). Also, in all simulations performed in this study, the real and imaginary parts of all signals, impulse responses and noise samples are treated separately at all points in the system model and so convolution, for example, is performed using real arithmetic and distinct variables rather than using complex variables. Consequently, the real and imaginary values of the complex noise samples are generated separately and added to the real and imaginary parts of the signal at the output of the receiver filter. To allow for this, the variance of each of the real and imaginary noise components must be equal to half the required variance of the complex noise source to ensure that each component is a sample value of a Gaussian random process with zero mean and 2-sided noise spectral density of $\frac{1}{2} N_0$ W/Hz, as discussed in Appendix 2.

References

1. Clark, A.P., "Principles of Digital Data Transmission Systems", Pentech Press, 1976.
2. McVerry, F., "High Speed Data Transmission over H.F. Radio Links", Ph.D. Thesis, Loughborough University of Technology, 1982.

Appendix 4

Results of computer simulations of Coding Scheme 3 discussed in Section 4.4.3.

The computer program listed as Program 1 in Appendix 10 is written in FORTRAN and models the coding scheme by generating random symbols complying with the signal constellation shown in Figure 4.4.20. The symbols are then converted to their 6-bit binary-coded representation and differentially coded according to Section 4.4.3. After every 100 symbols, the encoded symbols are rotated by $\frac{SN \cdot \pi}{200}$ rads. where SN is the number of symbols generated at the time of rotation. The decoder then differentially decodes the (rotated) symbols and converts the binary representation back to the symbol number for error analysis. The results presented in Table A4.1 show the correct operation of the coder/decoder when used with the signal constellation of Figure 4.4.20. The program assumes perfect detection with the symbols being transmitted over an ideal channel with no signal impairments; the S.N.R is assumed infinite. The errors recorded in the final column of Table A4.1 occur at the moment of phase rotation, when the reference pair of bits used at the decoder differ by 90° from those necessary to give true differential decoding.

OK. SEG #CODE1

DIFFERENTIAL CODE CHECK

ROTATION BY 90 DEGREES EVERY 100 SYMBOLS

Z=	DATA	CODE	DECODE	ROT	TX	RX	ERROR	
1	-7	-1	-7	-1	0	51	51	0
2	7	5	-7	-5	7	5	0	0
3	-3	5	-3	5	0	15	15	0
4	7	5	-5	7	7	5	0	0
5	-1	-7	7	-1	-1	-7	0	0
6	5	1	1	-5	5	1	0	0
7	5	7	7	-5	5	7	0	0
8	-7	5	-5	-7	-7	5	0	0
9	7	1	-7	-1	7	1	0	0
10	3	-1	3	-1	3	-1	0	0
11	-3	-5	-5	3	-3	-5	0	0
12	3	-5	-5	3	3	-5	0	0
13	7	-1	7	-1	7	-1	0	0
14	5	-1	1	5	5	-1	0	0
15	1	5	1	5	1	5	0	0
16	-1	-7	-1	-7	-1	-7	0	0
17	-3	7	-3	7	-3	7	0	0
18	-5	5	5	-5	5	5	0	0
19	-1	5	1	-5	-1	5	0	0
20	7	3	3	-7	7	3	0	0
21	-7	-1	-1	7	-7	-1	0	0
22	-5	-3	3	-5	-5	-3	0	0
23	3	3	3	-3	3	3	0	0
24	5	1	1	-5	5	1	0	0
25	7	1	1	-7	7	1	0	0
26	3	1	1	-3	3	1	0	0
27	-7	3	-3	-7	-7	3	0	0
28	3	-5	3	-5	3	-5	0	0
29	-3	-7	-7	3	-3	-7	0	0
30	-7	5	5	-7	-7	5	0	0
31	-5	-3	-5	3	-5	-3	0	0
32	-3	-5	3	-5	-3	-5	0	0
33	5	5	5	5	5	5	0	0
34	3	-5	-3	5	3	-5	0	0
35	7	-1	-1	-7	7	-1	0	0
36	-5	1	-5	1	-5	1	0	0
37	-5	-3	3	-5	-5	-3	0	0
38	1	7	7	-1	1	7	0	0
39	-5	7	-7	-5	-5	7	0	0
40	-5	-3	3	-5	-3	-5	0	0
41	3	-7	-3	7	3	-7	0	0
42	-5	3	3	-5	-5	3	0	0
43	-5	5	5	-5	-5	5	0	0
44	-7	5	-5	-7	-7	5	0	0
45	1	7	-7	1	7	7	0	0
46	5	3	-5	-3	5	3	0	0
47	3	-7	3	-7	3	-7	0	0
48	1	1	1	-1	1	1	0	0
49	3	1	1	-3	3	1	0	0
50	5	-3	3	5	-5	3	0	0
51	-1	-3	-1	-3	-1	-3	0	0
52	3	-5	3	-5	3	-5	0	0
53	-5	-7	-7	5	-5	-7	0	0
54	-5	1	1	5	-5	1	0	0
55	-5	5	5	-5	-5	5	0	0
56	1	-7	7	1	1	-7	0	0
57	7	1	7	1	7	1	0	0
58	-3	5	3	-5	-3	5	0	0
59	3	-1	1	3	3	-1	0	0
60	-7	1	7	-1	-7	1	0	0
					25	25		0

Table A4.1: Results of tests on Coding Scheme 3

61	3	-5	5	3	3	-5	0	40	40	0
62	-7	7	7	-7	-7	7	0	27	27	0
63	-3	-5	-5	3	-3	-5	0	62	62	0
64	1	1	-1	1	1	1	0	1	1	0
65	5	-3	-3	-5	5	-3	0	46	46	0
66	-5	1	-5	1	-5	1	0	29	29	0
67	-7	7	7	7	-7	7	0	27	27	0
68	-3	5	3	-5	-3	5	0	24	24	0
69	1	-3	3	1	1	-3	0	34	34	0
70	7	7	7	7	7	7	0	11	11	0
71	-5	-5	-5	-5	-5	-5	0	64	64	0
72	7	-3	7	-3	7	-3	0	42	42	0
73	1	-1	1	1	1	-1	0	33	33	0
74	-1	7	1	-7	-1	7	0	19	19	0
75	3	-3	3	3	3	-3	0	38	38	0
76	-7	-3	-7	-3	-7	-3	0	55	55	0
77	7	1	-7	-1	7	1	0	3	3	0
78	7	1	-7	-1	7	1	0	3	3	0
79	1	5	-1	-5	1	5	0	13	13	0
80	1	5	-1	-5	1	5	0	13	13	0
81	-5	-3	5	3	-5	-3	0	56	56	0
82	7	1	7	1	7	1	0	3	3	0
83	7	-7	-7	7	7	-7	0	43	43	0
84	7	1	-1	7	7	1	0	3	3	0
85	-1	5	5	1	-1	5	0	20	20	0
86	7	-1	-7	1	7	-1	0	41	41	0
87	5	7	-7	5	5	7	0	12	12	0
88	-1	-7	7	-1	-1	-7	0	57	57	0
89	-1	3	-3	-1	-1	3	0	18	18	0
90	1	3	-1	-3	1	3	0	5	5	0
91	7	1	-7	-1	7	1	0	3	3	0
92	-5	-3	5	3	-5	-3	0	56	56	0
93	-1	-1	-1	-1	-1	-1	0	49	49	0
94	1	-1	1	-1	1	-1	0	33	33	0
95	-3	-1	-1	3	-3	-1	0	50	50	0
96	7	1	-1	7	7	1	0	3	3	0
97	1	-7	-7	-1	1	-7	0	35	35	0
98	-7	3	-7	3	-7	3	0	26	26	0
99	3	5	-5	3	3	5	0	14	14	0
100	1	3	-3	1	3	-1	90	5	37	1
101	1	3	-3	1	1	3	90	5	5	1
102	-7	-5	5	-7	-7	-5	90	63	63	1
103	1	7	7	-1	1	7	90	9	9	1
104	3	-1	1	3	3	-1	90	37	37	1
105	-5	5	5	-5	-5	5	90	32	32	1
106	-7	-7	-7	7	-7	-7	90	59	59	1
107	3	-1	-1	-3	3	-1	90	37	37	1
108	1	1	-1	-1	1	1	90	1	1	1
109	3	5	-3	-5	3	5	90	14	14	1
110	5	-5	5	-5	5	-5	90	48	48	1
111	-5	7	-7	-5	-5	7	90	31	31	1
112	3	1	-3	-1	3	1	90	2	2	1
113	3	1	-3	-1	3	1	90	2	2	1
114	-5	-5	5	5	-5	-5	90	64	64	1
115	1	-7	-1	7	1	-7	90	35	35	1
116	3	-5	-5	-3	3	-5	90	40	40	1
117	7	-1	7	-1	7	-1	90	41	41	1
118	3	-5	5	3	3	-5	90	40	40	1
119	5	-1	-5	1	5	-1	90	45	45	1
120	-7	7	7	7	-7	7	90	27	27	1
121	-3	-3	-3	-3	-3	-3	90	54	54	1
122	-1	-1	1	1	-1	-1	90	49	49	1
123	1	-5	-1	5	1	-5	90	36	36	1
124	-1	1	1	1	-1	1	90	17	17	1
125	3	-5	-3	5	3	-5	90	40	40	1

Table A4.1 Contd.

126	7	-7	-7	-7	7	-7	90	43	43	1
127	-7	5	-7	5	-7	5	90	28	28	1
128	-3	5	5	3	-3	5	90	24	24	1
129	-5	-3	-5	-3	-5	-3	90	56	56	1
130	3	5	-3	-5	3	5	90	14	14	1
131	3	5	-3	-5	3	5	90	14	14	1
132	-7	3	-7	3	-7	3	90	26	26	1
133	3	-3	-3	-3	3	-3	90	38	38	1
134	-1	-5	1	5	-1	-5	90	61	61	1
135	-1	-3	-1	-3	-1	-3	90	53	53	1
136	3	-3	3	-3	3	-3	90	38	38	1
137	-7	-3	-3	7	-7	-3	90	55	55	1
138	7	1	-1	7	7	1	90	3	3	1
139	-5	-5	5	-5	-5	-5	90	64	64	1
140	5	-5	5	5	5	-5	90	48	48	1
141	7	3	7	3	7	3	90	7	7	1
142	-1	5	1	-5	-1	5	90	20	20	1
143	7	-5	5	7	7	-5	90	44	44	1
144	7	-1	-7	1	7	-1	90	41	41	1
145	-5	-7	7	-5	-5	-7	90	60	60	1
146	-1	5	-5	-1	-1	5	90	20	20	1
147	-3	7	-3	7	-3	7	90	23	23	1
148	5	-3	-3	-5	5	-3	90	46	46	1
149	7	-1	7	-1	7	-1	90	41	41	1
150	3	-5	5	3	3	-5	90	40	40	1
151	-7	5	7	-5	-7	5	90	28	28	1
152	-5	-1	-1	5	-5	-1	90	52	52	1
153	7	5	-5	7	7	5	90	15	15	1
154	7	3	-3	7	7	3	90	7	7	1
155	5	-5	-5	-5	5	-5	90	48	48	1
156	5	-7	5	-7	5	-7	90	47	47	1
157	5	3	3	-5	5	3	90	8	8	1
158	7	-3	3	7	7	-3	90	42	42	1
159	-3	-3	-3	-3	-3	-3	90	54	54	1
160	3	1	-3	-1	3	1	90	2	2	1
161	5	-7	5	-7	5	-7	90	47	47	1
162	-7	-1	-1	7	-7	-1	90	51	51	1
163	3	-7	-7	-3	3	-7	90	39	39	1
164	3	3	-3	-3	3	3	90	6	6	1
165	-5	-1	5	1	-5	-1	90	52	52	1
166	1	1	1	1	1	1	90	1	1	1
167	7	3	7	3	7	3	90	7	7	1
168	7	-5	-7	5	7	-5	90	44	44	1
169	-1	-1	1	-1	-1	-1	90	49	49	1
170	5	5	5	-5	5	5	90	16	16	1
171	-7	7	-7	-7	-7	7	90	27	27	1
172	-1	5	-1	5	-1	5	90	20	20	1
173	5	1	-1	5	5	1	90	4	4	1
174	-5	-7	7	-5	-5	-7	90	60	60	1
175	-3	-5	-5	3	-3	-5	90	62	62	1
176	-7	3	3	7	-7	3	90	26	26	1
177	-3	-5	-3	-5	-3	-5	90	62	62	1
178	-1	1	-1	1	-1	1	90	17	17	1
179	7	-7	-7	-7	7	-7	90	43	43	1
180	3	3	-3	-3	3	3	90	6	6	1
181	5	5	-5	-5	5	5	90	16	16	1
182	7	7	-7	-7	7	7	90	11	11	1
183	7	-3	7	-3	7	-3	90	42	42	1
184	5	-5	5	5	5	-5	90	48	48	1
185	-1	-5	-1	-5	-1	-5	90	61	61	1
186	-3	3	-3	3	-3	3	90	22	22	1
187	-3	-1	1	-3	-3	-1	90	50	50	1
188	-7	-3	-3	7	-7	-3	90	55	55	1
189	5	-1	-1	-5	5	-1	90	45	45	1
190	5	5	-5	-5	5	5	90	16	16	1

Table A4.1 Contd.

191	3	-3	3	-3	3	-3	90	38	38	1
192	7	7	7	-7	7	7	90	11	11	1
193	3	3	3	-3	3	3	90	6	6	1
194	-3	1	-1	-3	-3	1	90	21	21	1
195	5	7	-5	-7	5	7	90	12	12	1
196	3	3	-3	-3	3	3	90	6	6	1
197	-1	1	-1	1	-1	1	90	17	17	1
198	-3	1	1	3	-3	1	90	21	21	1
199	-5	-7	-5	-7	-5	-7	90	60	60	1
200	-7	3	-7	3	3	7	180	26	10	2
201	-3	-5	5	-3	-3	-5	180	62	62	2
202	-5	7	-7	-5	-5	7	180	31	31	2
203	-1	3	-1	3	-1	3	180	10	10	2
204	-1	-3	3	-1	-1	-3	180	53	53	2
205	1	3	3	-1	1	3	180	5	5	2
206	-1	7	-7	-1	-1	7	180	19	19	2
207	-5	7	-5	7	-5	7	180	31	31	2
208	-3	-3	3	-3	-3	-3	180	54	54	2
209	7	-7	7	7	7	-7	180	43	43	2
210	-7	-1	-7	-1	-7	-1	180	51	51	2
211	5	-1	5	-1	5	-1	180	45	45	2
212	5	5	5	-5	5	5	180	16	16	2
213	7	-1	1	7	-7	-1	180	41	41	2
214	-7	3	7	-3	-7	3	180	26	26	2
215	7	-3	3	7	7	-3	180	42	42	2
216	3	-1	-3	1	3	-1	180	37	37	2
217	5	-5	-5	-5	5	-5	180	48	48	2
218	-1	7	-1	7	-1	7	180	19	19	2
219	-5	-7	7	-5	-5	-7	180	60	60	2
220	-1	1	-1	-1	-1	1	180	17	17	2
221	-5	-7	5	7	-5	-7	180	60	60	2
222	3	-1	-3	1	3	-1	180	37	37	2
223	-3	3	3	3	-3	3	180	22	22	2
224	-1	-3	-1	-3	-1	-3	180	53	53	2
225	-7	3	-7	3	-7	3	180	26	26	2
226	-7	-3	3	-7	-7	-3	180	55	55	2
227	-1	1	-1	-1	-1	1	180	17	17	2
228	5	5	-5	-5	5	5	180	16	16	2
229	5	-7	5	-7	5	-7	180	47	47	2
230	3	-5	5	3	3	-5	180	40	40	2
231	1	1	1	1	1	1	180	1	1	2
232	7	7	7	7	7	7	180	11	11	2
233	-5	7	5	-7	-5	7	180	31	31	2
234	3	-1	1	3	3	-1	180	37	37	2
235	-1	7	1	-7	-1	7	180	19	19	2
236	1	1	1	-1	1	1	180	1	1	2
237	-1	7	-7	-1	-1	7	180	19	19	2
238	5	-7	5	-7	5	-7	180	47	47	2
239	5	-1	1	5	5	-1	180	45	45	2
240	-5	1	5	-1	-5	1	180	29	29	2
241	-5	-5	-5	5	-5	-5	180	64	64	2
242	3	-7	-7	-3	3	-7	180	39	39	2
243	-1	-7	1	7	-1	-7	180	57	57	2
244	5	5	5	5	5	5	180	16	16	2
245	-1	-1	-1	-1	-1	-1	180	49	49	2
246	5	-1	5	-1	5	-1	180	45	45	2
247	-3	7	-7	-3	-3	7	180	23	23	2
248	3	-7	3	-7	3	-7	180	39	39	2
249	1	-1	1	1	1	-1	180	33	33	2
250	3	5	3	5	3	5	180	14	14	2
251	1	-7	-1	7	1	-7	180	35	35	2
252	-7	-1	1	-7	-7	-1	180	51	51	2
253	-7	-1	-1	7	-7	-1	180	51	51	2
254	-7	5	5	7	-7	5	180	28	28	2

Table A4.1 Contd.

319	-1	3	-3	-1	-1	3	270	18	18
320	-5	3	-5	3	-5	3	270	30	30
321	7	-3	-3	-7	7	-3	270	42	42
322	3	-3	3	-3	3	-3	270	38	38
323	-3	3	-3	-3	-3	3	270	22	22
324	3	7	-3	-7	3	7	270	10	10
325	-7	-1	7	1	-7	-1	270	51	51
326	1	3	1	3	1	3	270	5	5
327	-3	3	3	-3	-3	3	270	22	22
328	1	1	1	-1	1	1	270	1	1
329	-1	-7	-7	1	-1	-7	270	57	57
330	-3	-7	7	-3	-3	-7	270	58	58
331	7	-7	7	7	7	-7	270	43	43
332	-7	3	7	-3	-7	3	270	26	26
333	7	-5	5	7	7	-5	270	44	44
334	3	-1	-3	1	3	-1	270	37	37
335	-5	1	1	5	-5	1	270	29	29
336	7	-7	-7	7	7	-7	270	43	43
337	-7	-1	1	-7	-7	-1	270	51	51
338	3	-1	1	3	3	-1	270	37	37
339	-3	3	3	3	3	3	270	6	6
340	-5	1	5	-1	-5	1	270	29	29
341	7	5	5	-7	7	5	270	15	15
342	7	-7	7	7	7	-7	270	43	43
343	5	-3	-5	3	5	-3	270	46	46
344	5	-5	-5	-5	5	-5	270	48	48
345	7	7	-7	-7	7	7	270	11	11
346	1	-5	1	-5	1	-5	270	36	36
347	3	-1	1	3	3	-1	270	37	37
348	3	7	3	7	3	7	270	10	10
349	3	3	3	3	3	3	270	6	6
350	-5	-1	-5	-1	-5	-1	270	52	52
351	-1	5	-1	5	-1	5	270	20	20
352	-1	-5	5	-1	-1	-5	270	61	61
353	-7	3	-3	-7	-7	3	270	26	26
354	3	5	-3	-5	3	5	270	14	14
355	7	3	-7	-3	7	3	270	7	7
356	1	3	-1	-3	1	3	270	5	5
357	7	-3	7	-3	7	-3	270	42	42
358	3	5	-3	3	5	-3	270	14	14
359	1	5	5	-1	1	5	270	13	13
360	3	-5	5	3	3	-5	270	40	40
361	-1	1	1	-1	-1	1	270	17	17
362	1	1	1	-1	1	1	270	1	1
363	5	-3	3	5	5	-3	270	46	46
364	-5	-3	-5	-3	-5	-3	270	56	56
365	-1	-7	1	7	-1	-7	270	57	57
366	1	5	1	5	1	5	270	13	13
367	5	3	5	3	5	3	270	8	8
368	-1	-1	-1	-1	-1	-1	270	49	49
369	5	3	-5	-3	5	3	270	8	8
370	-1	7	-1	7	-1	7	270	19	19
371	-3	-3	3	-3	-3	-3	270	54	54
372	-7	-3	-3	7	-7	-3	270	55	55
373	-3	-3	3	-3	-3	-3	270	54	54
374	-3	-7	-7	3	-3	-7	270	58	58
375	7	-5	-5	-7	7	-5	270	44	44
376	5	-1	5	-1	5	-1	270	45	45
377	7	-7	7	7	7	-7	270	43	43
378	5	-3	-5	3	5	-3	270	46	46
379	7	-7	-7	-7	7	-7	270	43	43
380	-1	-1	1	1	-1	-1	270	49	49
381	1	1	1	1	1	1	270	1	1
382	-5	-5	-5	-5	-5	-5	270	64	64
383	1	7	-1	-7	1	7	270	9	9
384	-1	3	-1	3	-1	3	270	18	18

Table A4.1 Contd.

Appendix 5

Error Probabilities for 64-Point and 256-Point QAM Signal Structures.

In this appendix, the signal constellations shown in Figures A5.1 and A5.2 are used to determine the probability of symbol errors for both 64-point and 256-point QAM signals operating over an ideal telephone channel which introduces no distortion, no interchannel interference and no signal impairments except additive white Gaussian noise. The results of the analysis therefore determines the performance of modems employing the two QAM systems over such channels and also demonstrates the necessary increase in tolerance of the 256-point, 2400 baud system to signal distortion for similar bit error rates at given signal-to-noise ratios. Throughout the following analysis, the output from the QAM demodulator is assumed to be;

$$r_i = s_i + u_i \quad (A5.1)$$

where $r_i = r_{ai} + jr_{bi}$

$$s_i = s_{ai} + js_{bi}$$

and $u_i = u_{ai} + ju_{bi}$

The $\{s_i\}$ are also assumed to be statistically independent and equally likely to have any of their 64 or 256 possible values. The $\{u_i\}$ are considered statistically independent Gaussian random variables with zero mean and variance $2\sigma^2$; the $\{u_{ai}\}$ and $\{u_{bi}\}$ are also sample values of Gaussian random processes with zero means but each with a variance of σ^2 . The outputs from the two quadrature demodulators, $r_{ai} = s_{ai} + ju_{ai}$ and $r_{bi} = s_{bi} + ju_{bi}$, are then operated on by two threshold detectors to give the detected data symbol, $s'_i = s'_{ai} + js'_{bi}$.

In both the 64-point and 256-point structures, the signal points are separated by a distance, $d=2$. Also since both structures are symmetrical about the real and imaginary axes, the constellations are known as minimum energy constellations⁽¹⁾. At the receiver, the received sample r_i is positioned within the signal constellation at a point determined by its real and imaginary components r_{ai} and r_{bi} , respectively. The detector then selects as the detected symbol, s'_i , the symbol which contains the r_i within its two-dimensional decision threshold region. Due to the rectangular shape of the constellations, three distinct shapes of decision region are possible. Figure A5.3 shows examples of these three regions occurring in the 64-point constellation and Figure A5.4 shows the general shape of the regions in isolation. Table A5.1 lists the symbol numbers (from Figure A5.1) which are surrounded by each of the three decision regions for the 64-point constellation whilst Table A5.2 gives a similar listing for the 256-point constellation of Figure A5.2.

Consider first the probability of error of a symbol surrounded by decision region 1 shown in Figure A5.4 a). For a given transmitted data symbol, s_i , the received sample, r_i , is itself a Gaussian random variable with a mean value given by s_i and variance $2\sigma^2$. The noise component present in the received sample may cause the r_i to cross the decision region surrounding the required symbol, s'_i . If the r_i crosses any of the decision boundaries associated with the transmitted symbol, the detector will output an incorrect symbol, giving a single symbol error. From Figure A5.4 a), an error will occur if the real noise sample, u_{ai} , forces r_i to cross either the left or right-hand boundaries AND/OR if the imaginary noise sample u_{bi} causes r_i to be positioned above the top boundary or below the bottom boundary. The analysis may be simplified

somewhat by initially calculating the probability of correct detection, P_c , for the entire signal set and then calculating the probability of error P_e , from $P_e = 1 - P_c$. For decision region 1, no error will occur provided that r_i remains within the decision boundary surrounding the transmitted signal element. That is, both u_{ai} and u_{bi} must be less than $\frac{d}{2}$. Since the $\{u_{ai}\}$ and $\{u_{bi}\}$ are statistically independent, then the probability of correct detection conditional on the transmitted symbol lying within a decision region as in Figure A5.4 a) will be,

$$P(\text{correct}|s_i)_1 = P(s_{ai} - \frac{d}{2} < r_{ai} < s_{ai} + \frac{d}{2}) \cdot P(s_{bi} - \frac{d}{2} < r_{bi} < s_{bi} + \frac{d}{2}) \quad (\text{A5.2})$$

$$\begin{aligned} P(\text{correct}|s_i)_1 &= \int_{s_{ai} - \frac{d}{2}}^{s_{ai} + \frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(r_a - s_{ai})^2}{2\sigma^2}\right) dr_a \\ &\quad \times \int_{s_{bi} - \frac{d}{2}}^{s_{bi} + \frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(r_b - s_{bi})^2}{2\sigma^2}\right) dr_b \end{aligned} \quad (\text{A5.3})$$

$$P(\text{correct}|s_i)_1 = \left[\int_{-\frac{d}{2}}^{\frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{u^2}{2\sigma^2}\right) du \right]^2 \quad (\text{A5.4})$$

where u is a Gaussian random variable with zero mean and variance σ^2 .

As illustrated in Figure A5.5,

$$\begin{aligned} \int_{-\frac{d}{2}}^{\frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{u^2}{2\sigma^2}\right) du &= 2 \int_0^{\frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{u^2}{2\sigma^2}\right) du \\ &= 1 - 2 \int_{\frac{d}{2}}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{u^2}{2\sigma^2}\right) du \end{aligned} \quad (\text{A5.5})$$

Now, if x is a Gaussian random variable with zero mean and variance σ^2 ,

then;

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right) dx = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) dz \quad (\text{A5.6})$$

where z is a Gaussian random variable with zero mean and unit variance.

Applying equation (A5.6) to equations (A5.5) and (A5.4) gives;

$$P(\text{correct}|s_i)_1 = \left[1 - 2 \int_{\frac{d}{2\sigma}}^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) dz \right]^2 \quad (\text{A5.7})$$

Therefore;

$$P(\text{correct}|s_i)_1 = \left[1 - 2Q\left(\frac{d}{2\sigma}\right) \right]^2 \quad (\text{A5.8})$$

where the Q -function, $Q\{y\} = \int_y^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) dz$

$Q\{y\}$ therefore gives the area of the Gaussian tail from $z=y$ to $z=+\infty$ and so is the probability $P(z > y)$.

Consider next the probability of correct detection for a symbol surrounded by decision region 2, as in Figure A5.4 b). Here, only one side of both r_a and r_b are constrained by the decision boundaries, the remaining two sides being totally unconstrained. Consequently, the probability of correct detection conditional on the transmitted symbol, s_i , lying within such a decision region is;

$$P(\text{correct}|s_i)_2 = P(s_{ai} - \frac{d}{2} < r_{ai} < \infty) \cdot P(s_{bi} - \frac{d}{2} < r_{bi} < \infty) \quad (\text{A5.9})$$

as illustrated in Figure A5.6.

$$P(\text{correct}|s_i)_2 = \int_{s_{ai} - \frac{d}{2}}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(r_a - s_{ai})^2}{2\sigma^2}\right) dr_a \\ \times \int_{s_{bi} - \frac{d}{2}}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(r_b - s_{bi})^2}{2\sigma^2}\right) dr_b \quad (\text{A5.10})$$

$$P(\text{correct}|s_i)_2 = \left[\int_{-\frac{d}{2}}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{u^2}{2\sigma^2}\right) du \right]^2 \quad (\text{A5.11})$$

Applying equations (A5.6) and (A5.8),

$$P(\text{correct}|s_i)_2 = \left[1 - Q\left(\frac{d}{2\sigma}\right) \right]^2 \quad (\text{A5.12})$$

Finally, the probability of correct symbol detection for symbols located within the third decision region shown in Figure A5.4 c) is,

$$P(\text{correct}|s_i)_3 = P(s_{ai} - \frac{d}{2} < r_{ai} < s_{ai} + \frac{d}{2}) \cdot P(-\infty < r_{bi} < s_{bi} + \frac{d}{2}) \quad (\text{A5.13})$$

as illustrated in Figure A5.7.

$$\begin{aligned} P(\text{correct}|s_i)_3 &= \int_{s_a - \frac{d}{2}}^{s_{ai} + \frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(r_a - s_{ai})^2}{2\sigma^2}\right) dr_a \\ &\times \int_{-\infty}^{s_{bi} + \frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(r_b - s_{bi})^2}{2\sigma^2}\right) dr_b \end{aligned} \quad (\text{A5.14})$$

$$\begin{aligned} P(\text{correct}|s_i)_3 &= \int_{-\frac{d}{2}}^{\frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{u^2}{2\sigma^2}\right) du \\ &\times \int_{-\infty}^{\frac{d}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{u^2}{2\sigma^2}\right) du \end{aligned} \quad (\text{A5.15})$$

From equations (A5.6) and (A5.8),

$$P(\text{correct}|s_i)_3 = \left[1 - 2Q\left(\frac{d}{2\sigma}\right) \right] \cdot \left[1 - Q\left(\frac{d}{2\sigma}\right) \right] \quad (\text{A5.16})$$

The probability of correct detection for the complete signal constellations can now be found from;

$$P(\text{correct}) = \sum_{i=1}^m P(s_i) P(\text{correct}|s_i) \quad (\text{A5.17})$$

where $P(s_i)$ is the a priori probability of transmitting a given symbol, s_i , and $P(\text{correct}|s_i)$ is the associated probability of correct detection

conditional on the position of that symbol within the signal space.

From Table A5.1, 36 symbols use decision region 1, 4 symbols use region 2 and 24 symbols use region 3, so for the 64-point structure, the probability of correct detection is given by;

$$\begin{aligned}
 P(\text{correct})_{64} = \frac{1}{64} & \left[36 \times \left[1 - 2Q\left(\frac{d}{2\sigma}\right) \right]^2 \right. \\
 & + 4 \times \left[1 - Q\left(\frac{d}{2\sigma}\right) \right]^2 \\
 & \left. + 24 \times \left[1 - 2Q\left(\frac{d}{2\sigma}\right) \right] \cdot \left[1 - Q\left(\frac{d}{2\sigma}\right) \right] \right] \quad (\text{A5.18})
 \end{aligned}$$

Expanding equation (A5.18) gives

$$P(\text{correct})_{64} = \left[1 - \frac{224}{64} Q\left(\frac{d}{2\sigma}\right) + \frac{196}{64} Q^2\left(\frac{d}{2\sigma}\right) \right] \quad (\text{A5.19})$$

The probability of error in the 64-point constellation is therefore given by;

$$\begin{aligned}
 P(\text{error})_{64} &= 1 - P(\text{correct})_{64} \\
 P(\text{error})_{64} &= \frac{224}{64} Q\left(\frac{d}{2\sigma}\right) - \frac{196}{64} Q^2\left(\frac{d}{2\sigma}\right) \quad (\text{A5.20})
 \end{aligned}$$

Equation (A5.20) may be further simplified by noting that $Q\left(\frac{d}{2\sigma}\right)$ represents the area under the tail of the Gaussian distribution and will ideally be a small number such that the probability of error will also be small. Consequently, the term $\frac{196}{64} Q^2\left(\frac{d}{2\sigma}\right)$ in equation (A5.20) will be extremely small and can be neglected. It follows that the approximation,

$$P(\text{error})_{64} \approx \frac{224}{64} Q\left(\frac{d}{2\sigma}\right) \quad (\text{A5.21})$$

represents the best possible performance of the 64-point system.

From Table A5.2, which refers to the 256-point structure, it can be seen that 196 signal points use decision region 1, 4 symbols use decision region 2 and 56 symbols use region 3 so, from equation (A5.17) the probability of correct detection for the 256-point structure will be;

$$\begin{aligned}
 P(\text{correct})_{256} = & \frac{1}{256} \left[196 \times \left[1 - 2Q\left(\frac{d}{2\sigma}\right) \right]^2 \right. \\
 & + 4 \times \left[1 - Q\left(\frac{d}{2\sigma}\right) \right]^2 \\
 & \left. + 56 \times \left[1 - 2Q\left(\frac{d}{2\sigma}\right) \right] \cdot \left[1 - Q\left(\frac{d}{2\sigma}\right) \right] \right] \quad (\text{A5.22})
 \end{aligned}$$

Expanding equation (A5.22) gives;

$$P(\text{correct})_{256} = \left[1 - \frac{960}{256} Q\left(\frac{d}{2\sigma}\right) + \frac{900}{256} Q^2\left(\frac{d}{2\sigma}\right) \right] \quad (\text{A5.23})$$

Applying the same approximation to equation (A5.23) as was made earlier to equation (A5.20) gives the approximate probability of error for the 256-point system as;

$$P(\text{error})_{256} \approx \frac{960}{256} Q\left(\frac{d}{2\sigma}\right) \quad (\text{A5.24})$$

The m symbols in the two constellations of Figures A5.1 and A5.2 are statistically independent and equally likely to have any of their m possible values. From Appendix 3, the average transmitted energy per symbol is,

$$\overline{|s_i|^2} = \frac{1}{m} \sum_{i=1}^m |s_i|^2 \quad (\text{A5.25})$$

Inserting the possible values of the $|s_i|^2 = s_{ai}^2 + s_{bi}^2$ into equation (A5.25) gives;

$$\overline{|s_i|_{64}^2} = 42 \quad (\text{A5.26})$$

$$\overline{|s_i|_{256}^2} = 170 \quad (\text{A5.27})$$

and since the 64-point structure block-codes 6 input binary digits to one symbol and the 256-point structure groups 8 binary digits to a single symbol, then the average energy per bit is given by

$$\lambda_{64} = \frac{\overline{|s_i|_{64}^2}}{6} = 7 \quad (\text{A5.28})$$

$$\lambda_{256} = \frac{\overline{|s_i|_{256}^2}}{8} = 21.25 \quad (\text{A5.29})$$

From Appendix 3, the signal-to-noise ratio is defined as;

$$\text{SNR} = 10 \log_{10} \left(\frac{\text{Average energy per bit}}{\text{2-sided noise spectral density}} \right) \quad (\text{A5.30})$$

Furthermore, the conditions stated in Appendix 3 are assumed to apply here such that the variance of the noise process at the output of the receiver's bandpass matched filter is equal to the 2-sided noise power spectral density at the input to the filter, hence,

$$\text{SNR} = 10 \log_{10} \left(\frac{\lambda}{\sigma^2} \right) \quad (\text{A5.31})$$

The variance of the noise process can therefore be written in terms of the average energy per bit of the particular system and the required SNR, viz;

$$\sigma^2 = \frac{\lambda}{10^{\left(\frac{\text{SNR}}{10}\right)}} \quad (\text{A5.32})$$

Using the fact that d , the distance between signal points, is equal to 2 for both the 64-point and 256-point constellations, then

equations (A5.21) and (A5.24) may be rewritten using the results of equations (A5.28), (A5.29) and (A5.32) as;

$$P(\text{error})_{64} \approx \frac{224}{64} Q \left[\frac{1}{\sqrt{\frac{7}{10} \left(\frac{\text{SNR}}{10} \right)}} \right]$$

$$P(\text{error})_{64} \approx 3.5 Q \left[\frac{10 \left(\frac{\text{SNR}}{20} \right)}{\sqrt{7}} \right] \quad (\text{A5.33})$$

and;

$$P(\text{error})_{256} \approx \frac{960}{256} Q \left[\frac{1}{\sqrt{\frac{21.25}{10} \left(\frac{\text{SNR}}{10} \right)}} \right]$$

$$P(\text{error})_{256} \approx 3.75 Q \left[\frac{10 \left(\frac{\text{SNR}}{20} \right)}{\sqrt{21.25}} \right] \quad (\text{A5.34})$$

Equations (A5.33) and (A5.34) therefore give the expressions for the probability of symbol errors against signal-to-noise ratio and are shown in graphical form in Figure A5.8. The figure depicts the performance of the 64-point and 256-point QAM systems when operating over a transmission channel which introduces only additive white Gaussian noise and no other form of signal impairment or signal distortion. When comparing the performance of the two systems, it should be realised from equations (A5.28) and (A5.29), that the average energy per bit of the 256-point system is $10 \log_{10} \left(\frac{21.25}{7} \right) = 4.82$ dB higher than that of the 64-point system. However, in the derivation of equations (A5.33) and (A5.34), equation (A5.32)

ensures that the value of σ^2 is adjusted accordingly to maintain a similar signal-to-noise ratio for the two systems. Inspection of Figure A5.8 shows that the signal-to-noise ratio for the 256-point system (which, for a given noise variance implies the average energy per bit) must be increased by about 4.8 dB to give a performance comparable with the 64-point system. Put another way, the 256-point system, which it will be recalled operates at a lower baud rate compared with the 64-point system, must gain at least 4.8 dB in terms of performance in the presence of signal distortion and other impairments likely to be introduced over a practical telephone channel if it is to offer an overall advantage over the 64-point system. Chapter 6 investigates the performance of the two systems operating over a range of telephone circuits with practical detectors by means of computer simulations, but it is worth noting here that if the two systems are operating with identical average energy per bit, the minimum distance between adjacent signal points in the 256-point structure will be less than the corresponding distance in the 64-point constellation by a factor of 1.74. This not only results in a reduced noise tolerance, as shown in Figure A5.8, but also reduces the tolerance to small phase errors between the signal and reference carriers, as illustrated in Figure A5.9. The accurate estimation of signal carrier phase therefore becomes increasingly important as the size of the signal constellation increases.

References

1. Proakis, J.G. : "Digital Communications", McGraw-Hill Inc., 1983.

DECISION REGION	SYMBOL NUMBERS	TOTAL SYMBOLS
	1,2,4,5,6,8,13,14,16,17,18,20,21,22,24,25,26,28,	
REGION 1	29,30,32,33,34,36,37,38,40,41,42,44,45,46,48,49, 50,52,53,54,56,57,58,60,61,62,64	36
REGION 2	11,27,43,59	4
REGION 3	3,7,9,10,12,15,19,23,25,26,28,31,35,39,41,42, 44,47,51,55,57,58,60,63	24

Table A5.1: Listing of symbols contained within the three decision regions; 64-point structure.

DECISION REGION	SYMBOL NUMBERS	TOTAL SYMBOLS
REGION 1	1,2,3,4,6,7,8,17,18,19,20,22,23,24,25,26,27, 28,30,31,32,33,34,35,36,38,39,40,41,42,43,44, 46,47,48,49,50,51,52,54,55,56,57,58,59,60,62, 63,64,65,66,67,68,70,71,72,81,82,83,84,86,87, 88,89,90,91,92,94,95,96,97,98,99,100,102,103, 104,105,106,107,108,110,111,112,113,114,115, 116,118,119,120,121,122,123,124,126,127,128, 129,130,131,132,134,135,136,145,146,147,148, 150,151,152,153,154,155,156,158,159,160,161, 162,163,164,166,167,168,169,170,171,172,174, 175,176,177,178,179,180,182,183,184,185,186, 187,188,190,191,192,193,194,195,196,198,199,200, 209,210,211,212,214,215,216,217,218,219,220,222, 223,224,225,226,227,228,230,231,232,233,234,235, 236,238,239,240,241,242,243,244,246,247,248,249, 250,251,252,254,255,256	196
REGION 2	13,77,141,205	4
	5,9,10,11,12,14,15,16,21,29,37,45,53,61,69,73, 74,75,76,78,79,80,85,93,101,109,117,125,133,	
REGION 3	137,138,139,140,142,143,144,149,157,165,173,181, 189,197,201,202,203,204,206,207,208,213,221,229, 237,245,253	56

Table A5.2: Listing of symbols contained within the three decision regions; 256-point structure.

				s_{bi}					
011010	011110	010110	010010		001000	001001	001011	001010	
27	31	23	19	7	9	10	12	11	
011011	011111	010111	010011		001100	001101	001111	001110	
28	32	24	20	5	13	14	16	15	
011001	011101	010101	010001		000100	000101	000111	000110	
26	30	22	18	3	5	6	8	7	
011000	011100	010100	010000		000000	000001	000011	000010	
25	29	21	17	1	1	2	4	3	
-7	-5	-3	-1		1	3	5	7	s_{ai}
110010	110011	110001	110000		100000	100100	101100	101000	
51	52	50	49	-1	33	37	45	41	
110110	110111	110101	110100		100001	100101	101101	101001	
55	56	54	53	-3	34	38	46	42	
111110	111111	111101	111100		100011	100111	101111	101011	
63	64	62	61	-5	36	40	48	44	
111010	111011	111001	111000		100010	100110	101110	101010	
59	60	58	57	-7	35	39	47	43	

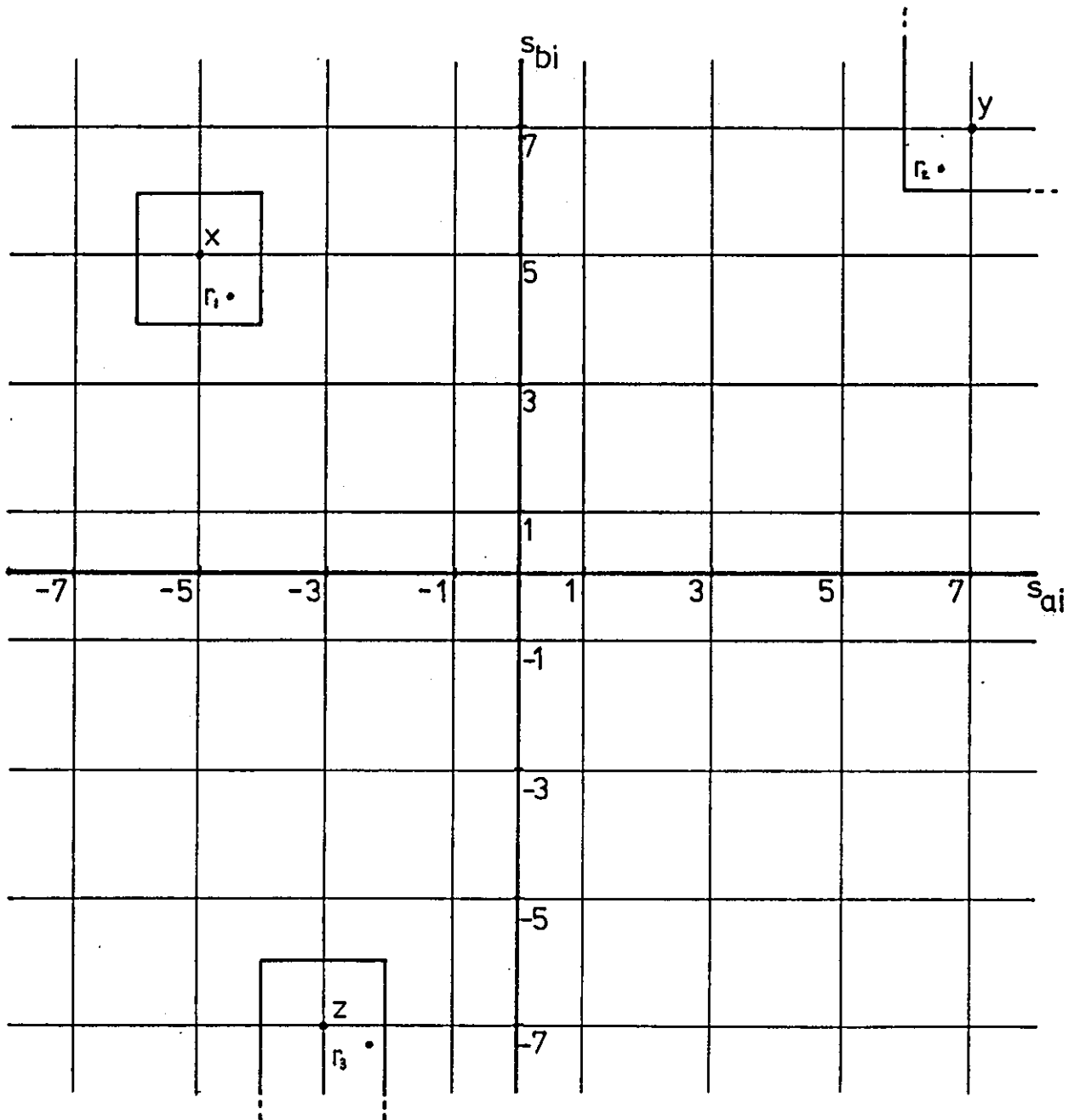
Figure A5.1 : Signal constellation for Coding Scheme 3;
64-point structure

Quadrant identifiers; 00

Symbol value; 00XXXXXX

s_{bi}								
15	001000	001001	001011	001010	001110	001111	001101	001100
	9	10	12	11	15	16	15	13
13	011000	011001	011011	011010	011110	011111	011101	011100
	25	26	28	27	31	32	30	29
11	111000	111001	111011	111010	111110	111111	111101	111100
	57	58	60	59	63	64	62	61
9	101000	101001	101011	101010	101110	101111	101101	101100
	41	42	44	43	47	48	46	45
7	100000	100001	100011	100010	100110	100111	100101	100100
	33	34	36	35	39	40	38	37
5	110000	110001	110011	110010	110110	110111	110101	110100
	49	50	52	51	55	56	54	53
3	010000	010001	010011	010010	010110	010111	010101	010100
	17	18	20	19	23	24	22	21
1	000000	000001	000011	000010	000110	000111	000101	000100
	1	2	4	3	7	8	6	5
	1	3	5	7	9	11	13	15 s_{ai}

Figure A5.2 : Signal constellation for Coding Scheme 4;
256-point structure



$$r_1 = -4.5 + j4.8; \text{ Detected Symbol, } s'_1 = x$$

$$r_2 = 6.3 + j6.5; \text{ Detected Symbol, } s'_2 = y$$

$$r_3 = -2.3 - j7.3; \text{ Detected Symbol, } s'_3 = z$$

Figure A5.3: The three different decision regions in rectangular constellations.

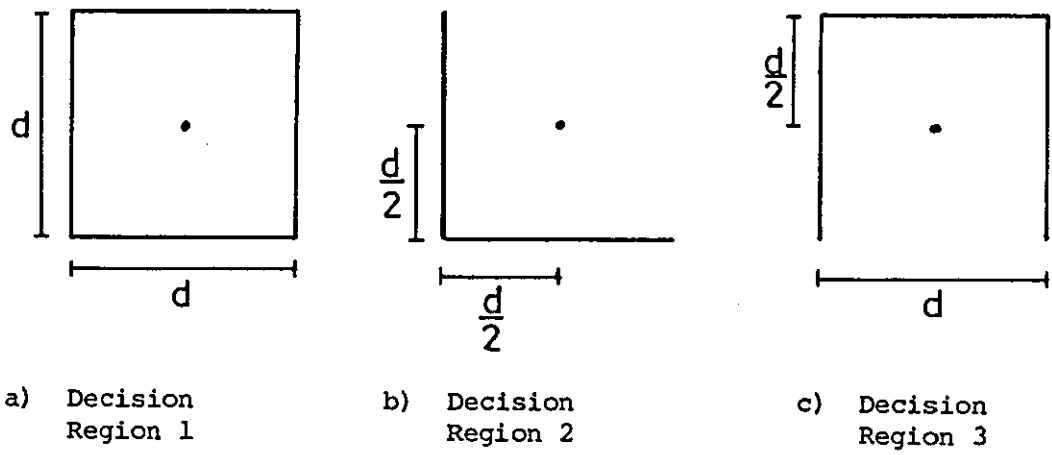


Figure A5.4: The three decision regions and distances to decision boundaries.

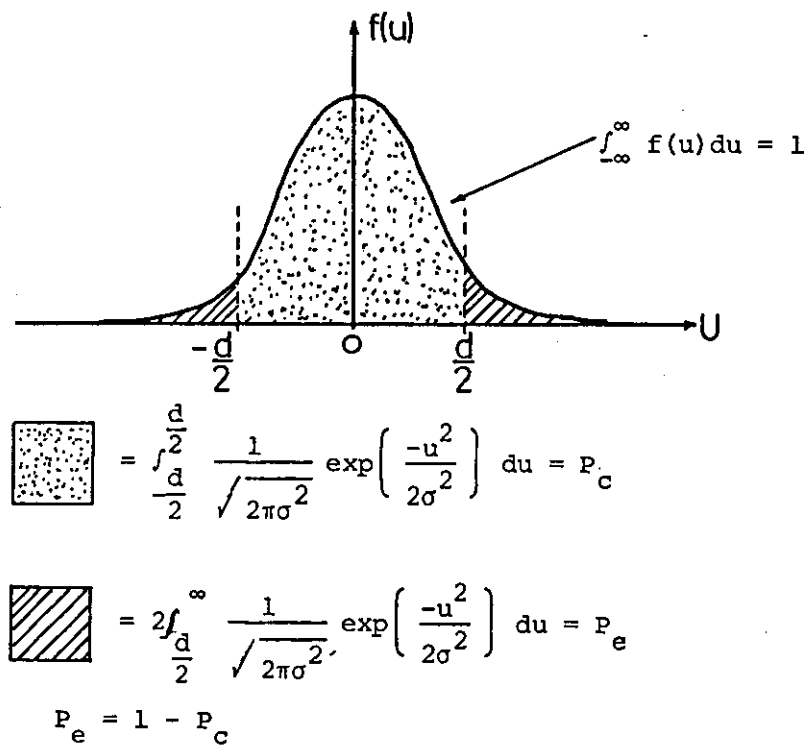


Figure A5.5: Gaussian distribution for decision region 1

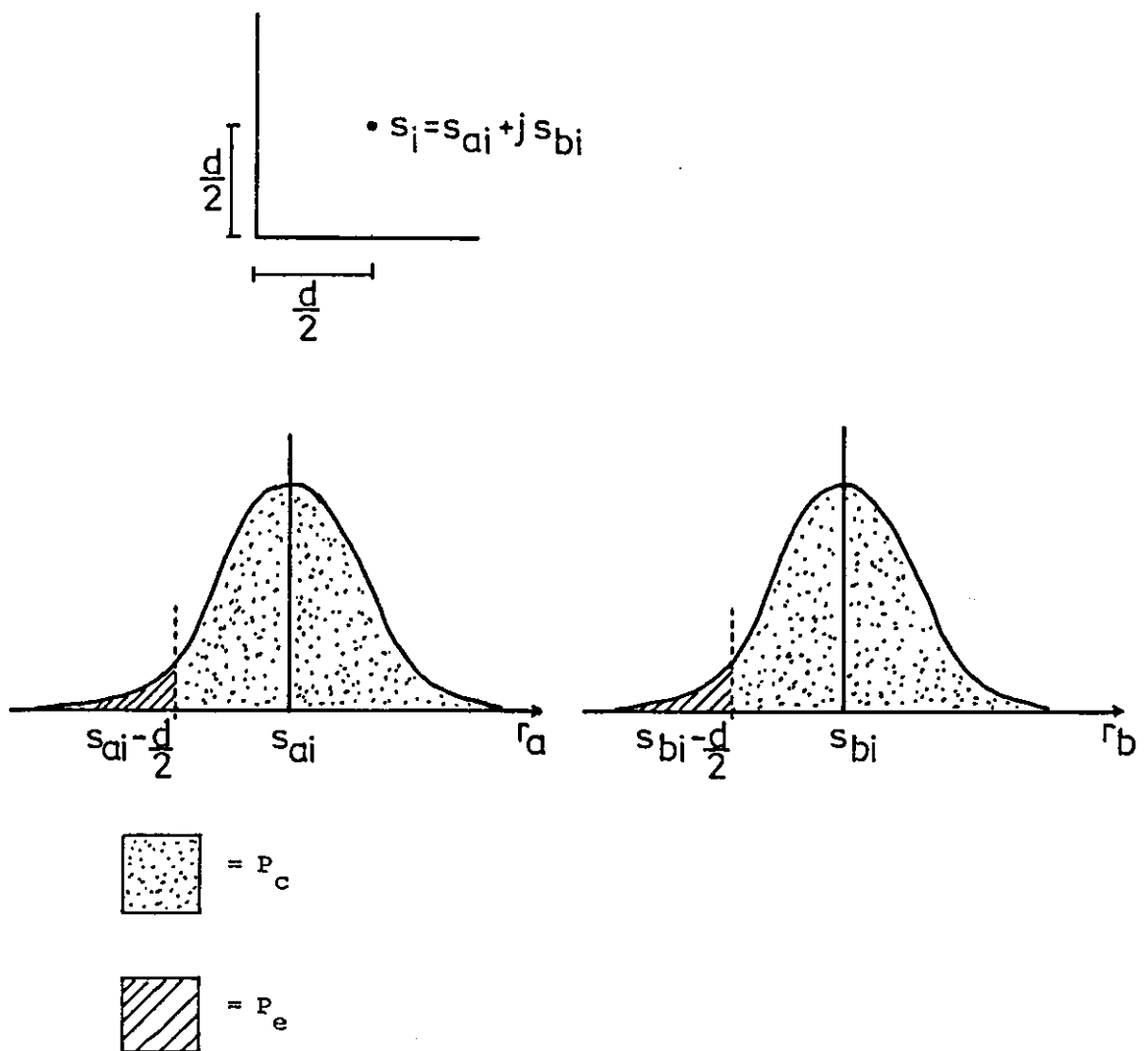


Figure A5.6: Gaussian distributions for decision region 2

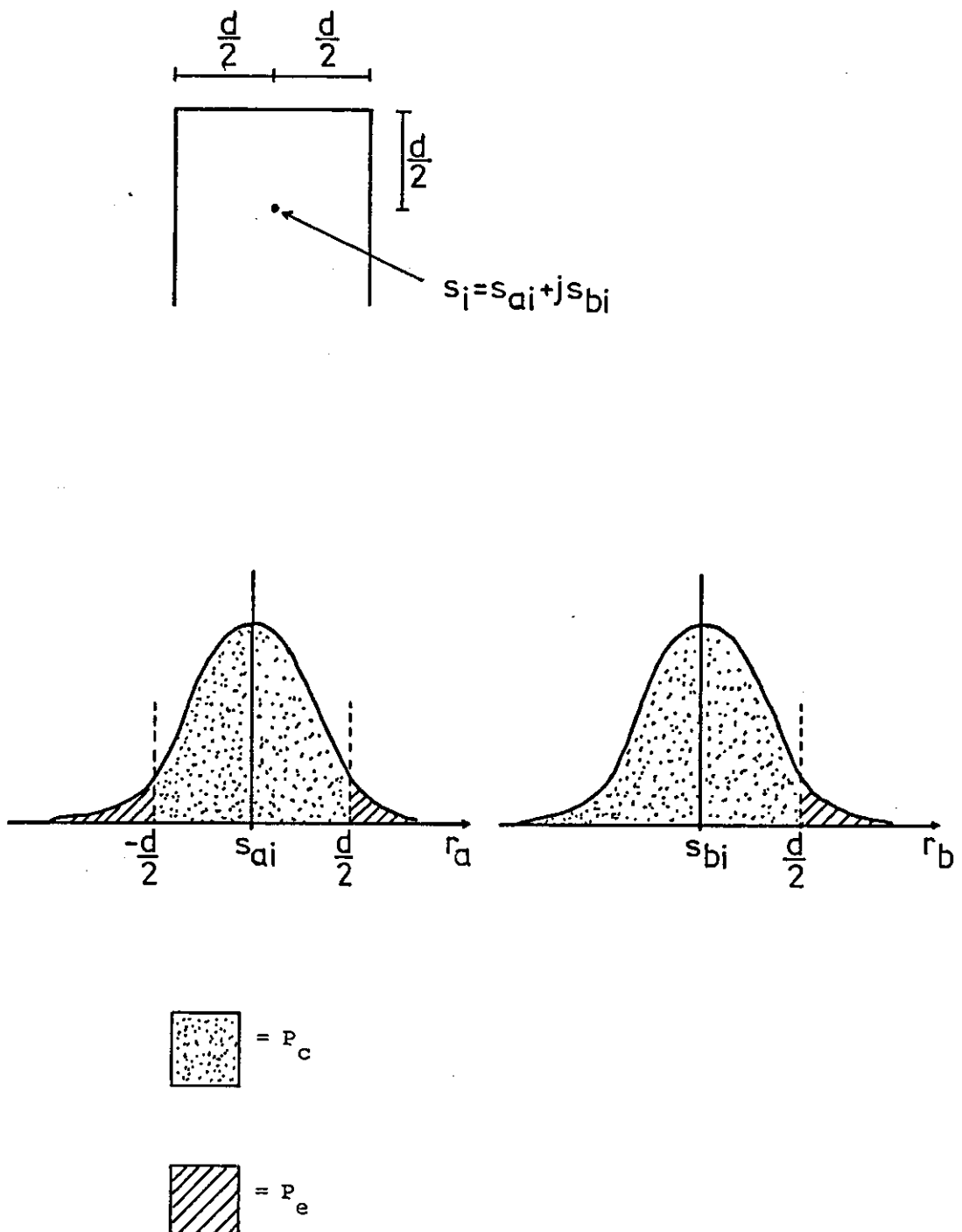


Figure A5.7 Gaussian distributions for decision region 3

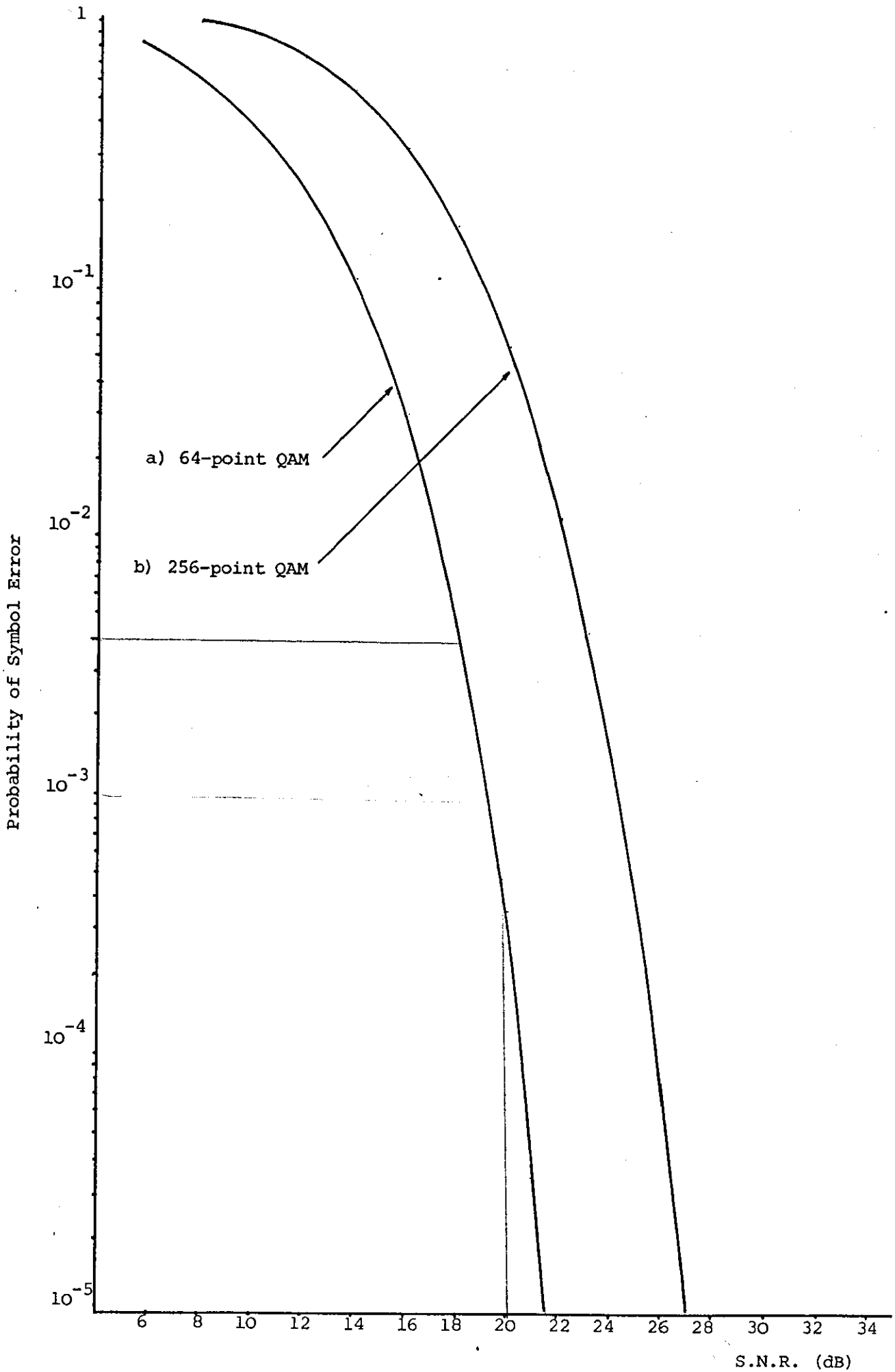


Figure A5.8: Symbol error probabilities for the 64-point and 256-point QAM structures.

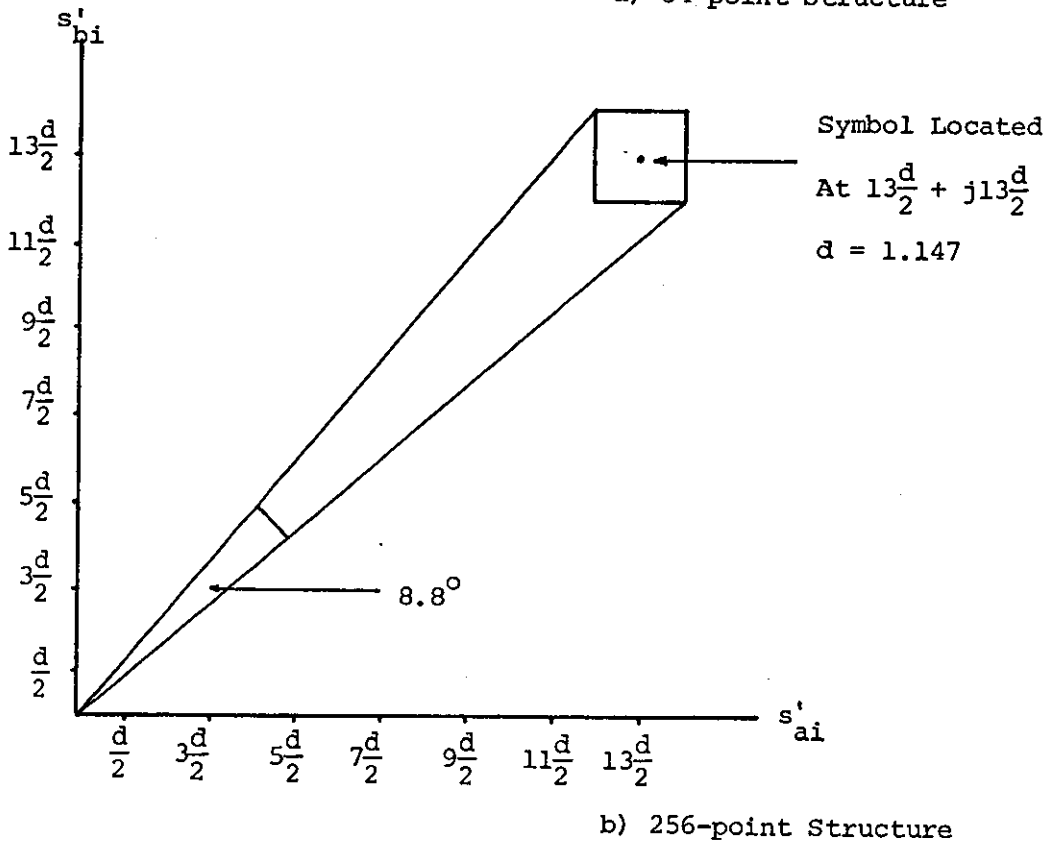
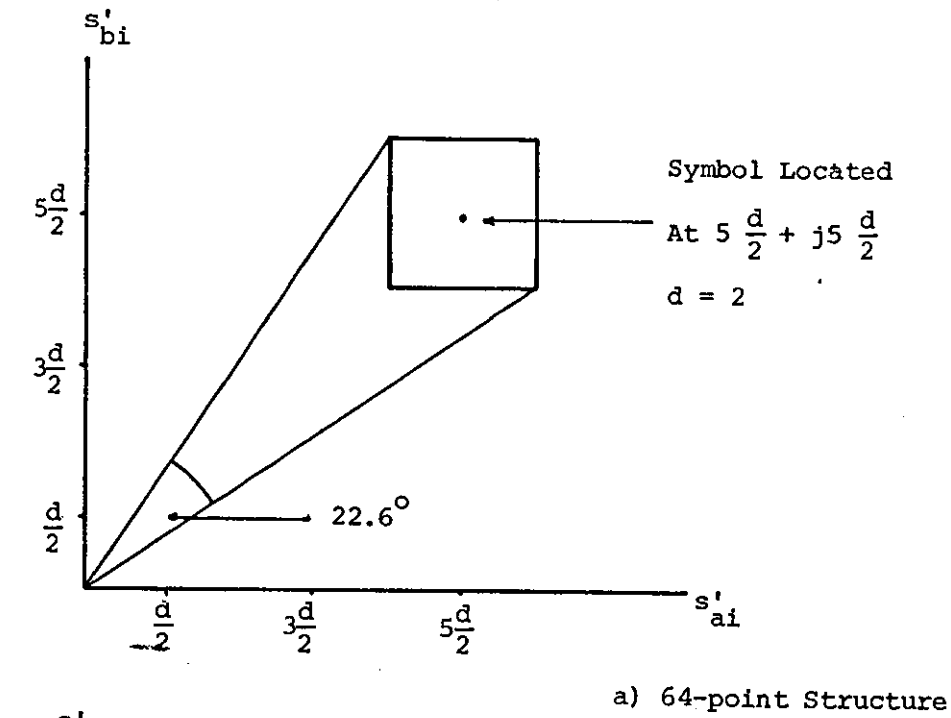


Figure A5.9: Best-case tolerance to phase errors for most vulnerable symbols in 64-point and 256-point constellations having identical average energy per bit.

Appendix 6

Digital Filter Design

This appendix describes the procedure followed for the design of non-recursive low-pass digital filters exhibiting minimum phase equivalents of sinusoidal roll-off characteristics centred around the cut-off frequency, f_{cut} Hz⁽¹⁻⁵⁾.

Figure A6.1 shows the attenuation characteristic of the combination of all filters known here as filter set 1^(6,7). From the figure, it can be seen that the -6dB points occur at about 775 Hz and 2800 Hz. As stated in Chapter 5, the major contributors to the overall filter characteristic are the pre-modulation and post-demodulation low-pass filters. The requirement here is to design a digital filter which will be used for both the transmitter and receiver low-pass filters, which has a bandwidth wider than the corresponding filter in filter set 1 and which has a minimum phase response.

For transmission of digital data at 3200 bauds over bandlimited channels, the shape of the transmitted signal spectrum should be such to ensure that the channel does not introduce intersymbol interference. The Nyquist vestigial-symmetry theorem states that if the spectrum of the transmitted signal is purely real and has odd symmetry about the nominal cut-off frequency $\frac{1}{2T}$ Hz, then the corresponding impulse response of the signal is an even time function and has zeros at all non-zero integer values of iT . Under these conditions, it is possible to transmit data at $\frac{1}{T}$ baud without intersymbol interference^(8,9). The most commonly used spectrum which satisfies these conditions is one which has a sinusoidal roll-off centred at $\pm \frac{1}{2T}$ Hz, where the roll-off factor is adjusted to give the required spectral characteristics for a given system. Figure A6.2

shows the case of a filter exhibiting a sinusoidal roll-off, with a roll-off factor of about 40%. It is assumed that the symbol rate is 3200 baud and so the frequency of odd-symmetry is 1600 Hz. After passing through a modulator which uses a carrier frequency of 1800 Hz, the spectrum of the signal applied to the telephone channel will be as shown in Figure A6.3, where it is assumed that the channel input filter does not modify the signal spectrum to any great extent. It is also assumed that the input signal to the filter is in the form of impulses. Figure A6.3 also shows the spectral restrictions which are imposed on signals connected to telephone circuits in the U.K. It should be obvious from the figure that the baseband signal spectrum shown in Figure A6.2 is not suitable since the transmitted signal spectrum violates the channel input restrictions. Moreover, because of the restriction occurring at frequencies above 3150 Hz, even a signal spectrum with a zero roll-off factor and a cut-off frequency of 1600 Hz will violate these restrictions. Consequently, it is necessary to lower the filter's cut-off frequency to a value below 1600 Hz which will mean that the filter will be sub-optimum with regards to the Nyquist criterion and will therefore introduce some degree of intersymbol interference.

The starting point in the filter design is to specify the bandwidth of the combined filters to be approximately 200 Hz wider at the -6dB points than the characteristic shown in Figure A6.1 at both the high and low frequency ends; this increase being achieved in practice by increasing the cut-off frequency of both low-pass filters (filters 1b and 1d in Chapter 5) by 200 Hz at the -3dB points. It is assumed here that the filtering is shared equally between the two low-pass filters. Figure A6.4

shows an approximation to the attenuation characteristic of the required low-pass filters. Approximations for the transition frequency and the frequency about which odd symmetry occurs can now be made by modelling the actual sinusoidal roll-off part of the amplitude characteristic around the cut-off region as a linear increase of attenuation which just satisfies the channel restrictions when referred to baseband. Using these approximate values, initial values for cut-off frequency and roll-off factor are calculated and entered into a computer program which performs the required design steps. The program used is listed as Program 2 in Appendix 10 and is a modified version of the program developed by Fairfield for the design of filters 1b and 1d presented in Chapter 5^(6,7). The filter design can be summarised as follows and with reference to Figure A6.5.

- a) The program operates with a digital sampling frequency of 14,400 samples per second and stores 256 samples; the separation between samples in the frequency domain is therefore 56.25 Hz. Other values can be accommodated by changing the appropriate input variables.
- b) From input data relating to the cut-off frequency and the width (in Hertz) of the roll-off region, the program forms an attenuation characteristic exhibiting sinusoidal roll-off centred about the cut-off frequency; a zero group delay response is assumed (Figure A6.5a).
- c) The impulse response of the transfer function is obtained from an inverse FFT routine and is sampled in the time domain at the digital sampling rate. Due to the assumed group delay response, the sampled impulse response will be symmetrical about zero time and will be of infinite duration. (Figure A6.5b).

- d) The sampled impulse response is then truncated by a Hamming window function. (Figure A6.5c)
- e) The resulting sampled impulse response is still symmetrical about zero time and has far from minimum rise time. It is mentioned in Chapter 5 that the desired time domain response of the low-pass filters has a rapid rise time and is of short duration. This can be achieved by making the response minimum-phase. The program evaluates the roots (zeros) of the z-transform of the sampled impulse response and replaces all zeros which lie outside the unit circle in the z-plane by the conjugate of their reciprocals, as explained in detail in Chapter 5. This operation produces a filter with an impulse response that is constrained to have minimum rise time and duration. (Figure A6.5d).
- f) The minimum-phase samples of the filter's impulse response are then quantised to allow the filter to be implemented to the required accuracy.
- g) Finally, the attenuation and group delay characteristics are calculated using an FFT routine to give the final filter characteristics.

Several low-pass filters were designed using the above procedure with different combinations of cut-off frequency and roll-off factor. The filter finally selected for use in filter set 2 represented the best compromise between increased bandwidth and length of impulse response. The filter also satisfied the channel spectral restrictions illustrated earlier so enabling the analogue channel input filter to be designed with less strict constraints on its cut-off frequency. The amplitude and group

delay characteristics of the selected low-pass filter, designated filter 2b and 2d for the pre-modulation and post-demodulation filters respectively, are shown in Figure A6.6. It can be seen from the figure that the -3dB bandwidth of the individual filters occurs at approximately 1300 Hz compared with approximately 1100 Hz for the corresponding filters in filter set 1, which gives the required bandwidth increase of approximately 200 Hz.

References

1. Cain, G.D., "Digital Filter Wizardry - Up for Grabs", IERE Conference on Digital Processing of Signals in Communications, Loughborough University of Technology, April, 1981.
2. Forsythe, W., "A New Method for the Computation of Digital Filter Coefficients", Departmental Memorandum No.142, Dept. Electronic and Electrical Engineering, Loughborough University of Technology, June 1983.
3. Bradley, J.T., "Digital Filter Design Procedure", Wireless World, pp 42-43, July 1983.
4. Strizecki, P., "Digital Filtering", Systems International, pp 116-117, November 1983.
5. Bin Nun, M.A. and Woodward, M.E., "A Modular Approach to the Hardware Implementation of Digital Filters", The Radio and Electronic Engineer, Vol.46, No.8/9, pp 393-400, August/September 1976.
6. Fairfield, M.J., "Equipment Filter Design", Internal Communication, Loughborough University of Technology, 1978.
7. Fairfield, M.J., "The Chosen Equipment Filters", Internal Communication, Loughborough University of Technology, 1978.

8. Clark, A.P., "Principles of Digital Data Transmission",
Pentech Press, 1976.
9. Carlson, A.B., "Communication Systems", McGraw Hill Inc.,
Second Edition, 1975.

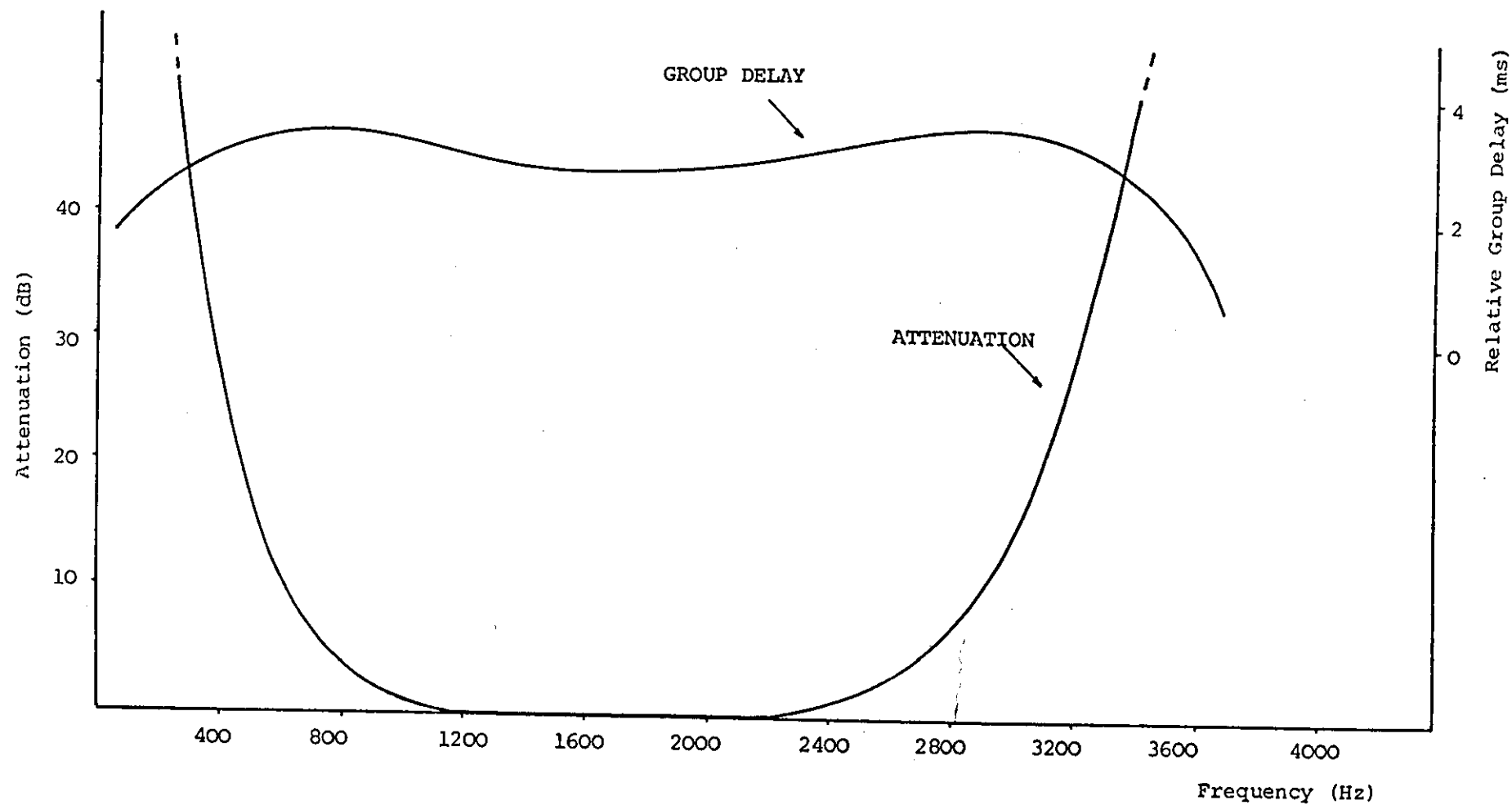


Figure A6.1 : Attenuation and Group Delay Characteristics for Filter 1g; Combined Filter Set 1

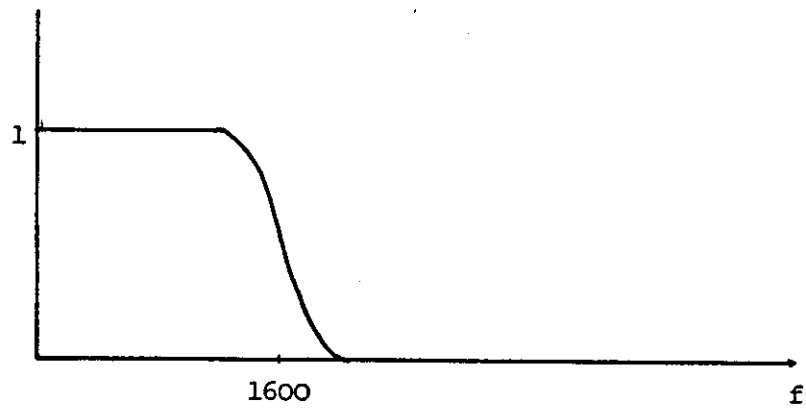


Figure A6.2: Amplitude characteristic of low-pass filter with sinusoidal roll-off around 1600 Hz

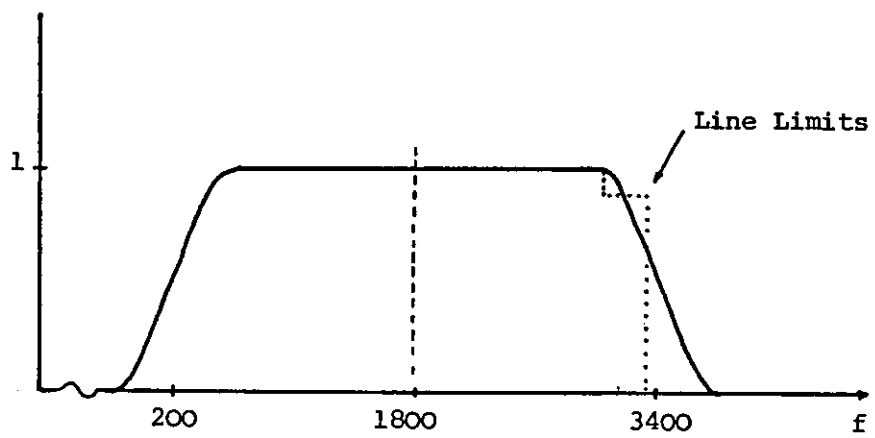


Figure A6.3: Translated amplitude characteristic after modulation.

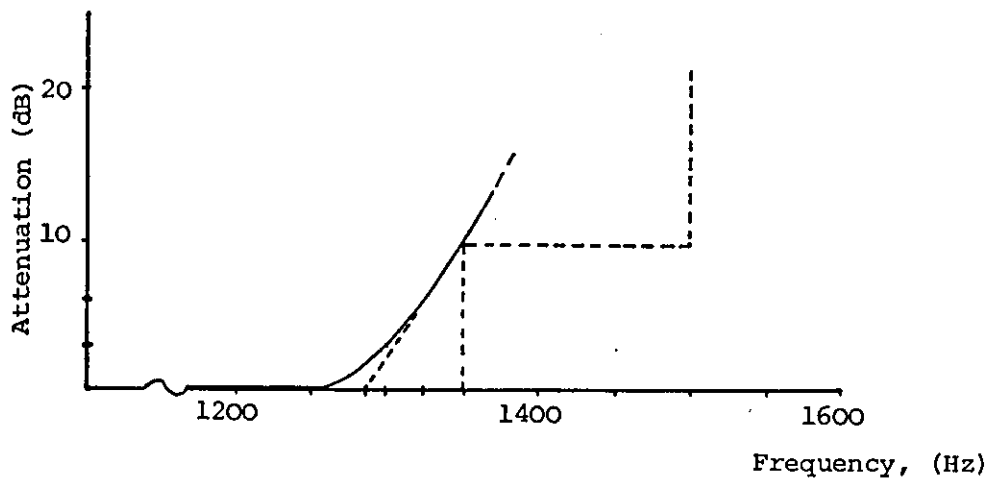
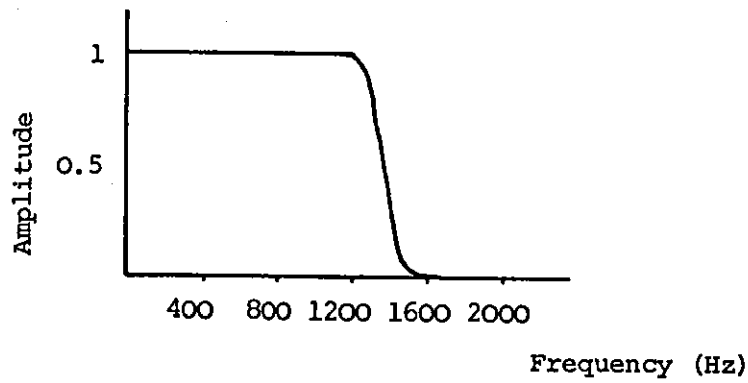
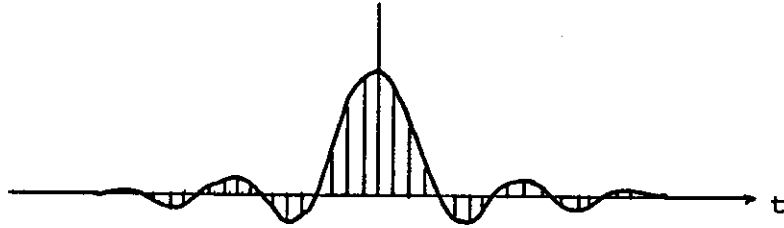


Figure A6.4: Approximations of transition and cut-off frequencies.

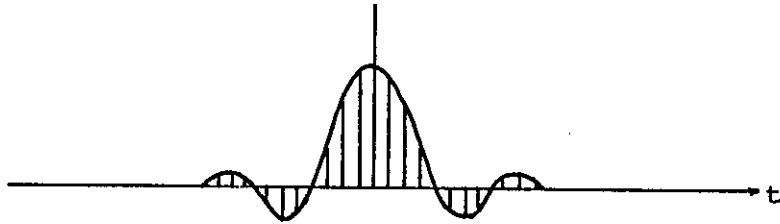
$$f_{\text{tran.}} \approx 1285 \text{ Hz}, f_{\text{cut.}} \approx 1325 \text{ Hz.}$$



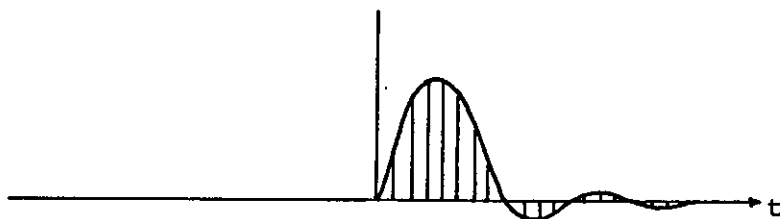
- a) Define amplitude response.



- b) Derive sampled impulse response at digital sampling rate.



- c) Apply window function to truncate impulse response



- d) Phase-equalise the sampled impulse response.

Figure A6.5: Digital filter design procedure.

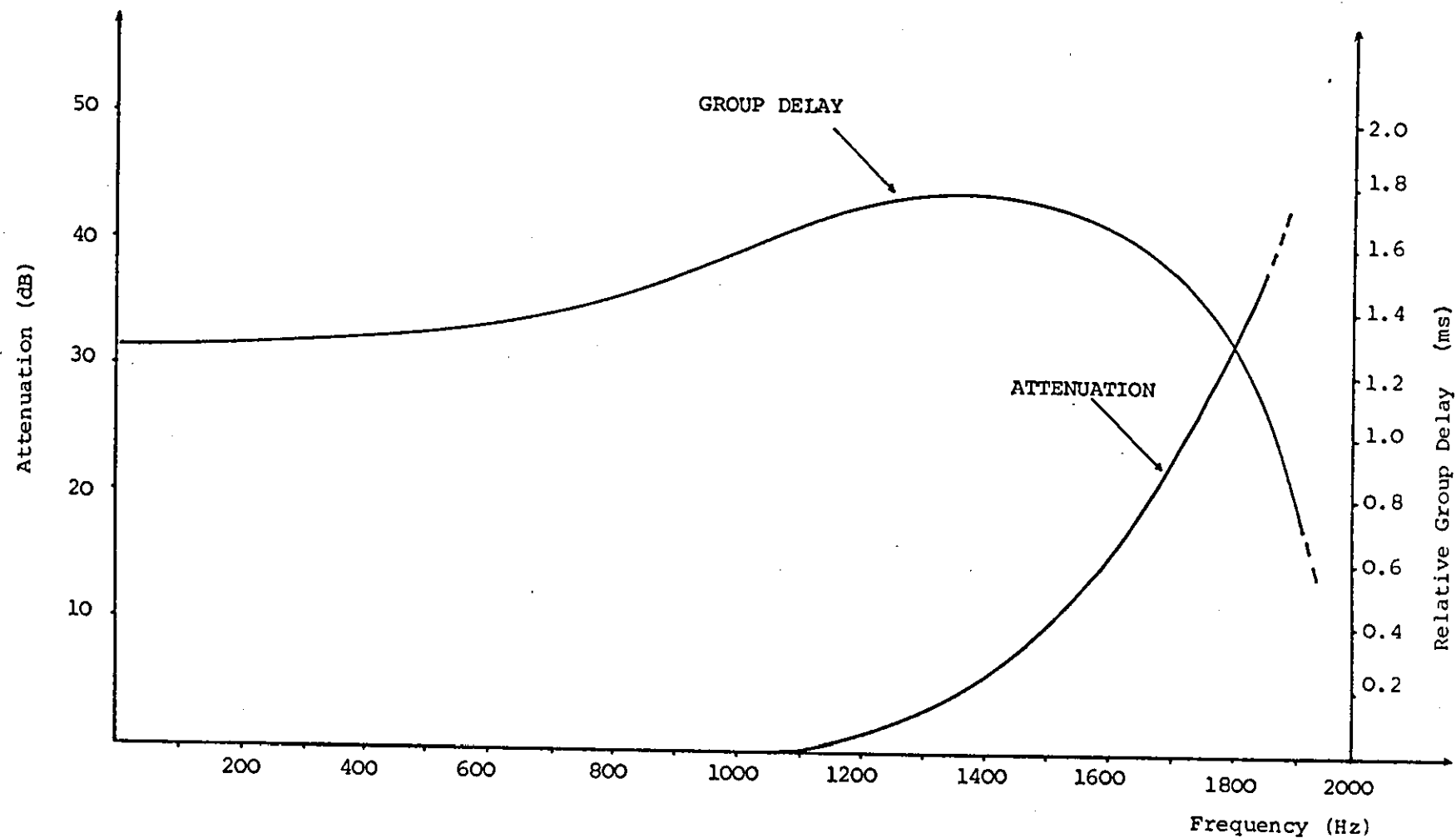


Figure A6.6 : Attenuation and Group Delay Characteristics for Low-Pass Filters 2b and 2c

Appendix 7Selection of Suitable Switched-Capacitor Filters for Use
in High-Speed Data Modems

After initially considering several commercially available switched-capacitor integrated circuits⁽¹⁻⁶⁾, the Mitel MT8912 and Intel I2912 devices were selected as being potentially suitable to perform the low-pass and bandpass filtering functions necessary at the input and output respectively of a telephone circuit. At the time of study, both devices were relatively recent additions to the integrated circuit market and only preliminary specifications were available. In both cases, the specification and data sheets gave insufficient data relating to attenuation and phase or group delay characteristics and so a comparative study between the two devices was performed in the laboratory⁽⁷⁾. Figures A7.1 to A7.4 show the results of this study. In all cases, the attenuation characteristics for both filters were identical when operating under similar conditions. It is important to realise that the filters designated by both manufacturers as "receive" and "transmit" must be transposed when using the devices in digital modems. The conclusions of this study recommend the MT8912 filter in preference to the I2912 device due to the lower supply current requirement of the former device and its slightly flatter group delay characteristic over the frequency range 0 to 3200 Hz. The choice of clock frequency depends to a large extent on the frequency of the master clock generators used in the overall modem design. Initially 1.8432 MHz was used for the filters as this enabled the digital sampling frequency 14.4 kHz required by the digital low-pass filters to be derived by simple division

by 128. In the actual modem design given in Chapter 9, the clock frequency of the MT8912 must be increased to 2.048 MHz to accommodate simple derivation of the updated digital sampling frequency, 16kHz.

References

1. "Switched Capacitor Filters for 1200 baud Modems", Technology Trends, Electronic Product Design, p.9, July 1982.
2. Orr, T., "Designers Notebook; Switched Capacitor Filters", Electronics Today International, pp 45-47, November 1982.
3. Mitel Semiconductors Corp., "The MT8912 ISO²-CMOS PCM Filter", Specification and Data Sheets, 1981.
4. Intel Corporation, "The 2912A PCM Transmit/Receive Filter", Specification and Data Sheets, 1980.
5. Motorola Inc., "The MC145414 Dual Tunable Low-Pass Sampled Data Filters", Specification and Data Sheets, 1982.
6. EG and G Reticon Inc., "The R5620 Universal Active Filter", Preliminary Data Sheet, 1982.
7. Bateman, S.C., "An Investigation of Group Delay Characteristics of the Mitel MT8912 and Intel I2912 PCM Line Filters and their Possible Application in High-Speed Data Modems", Internal Communication, Loughborough University of Technology, 1982.

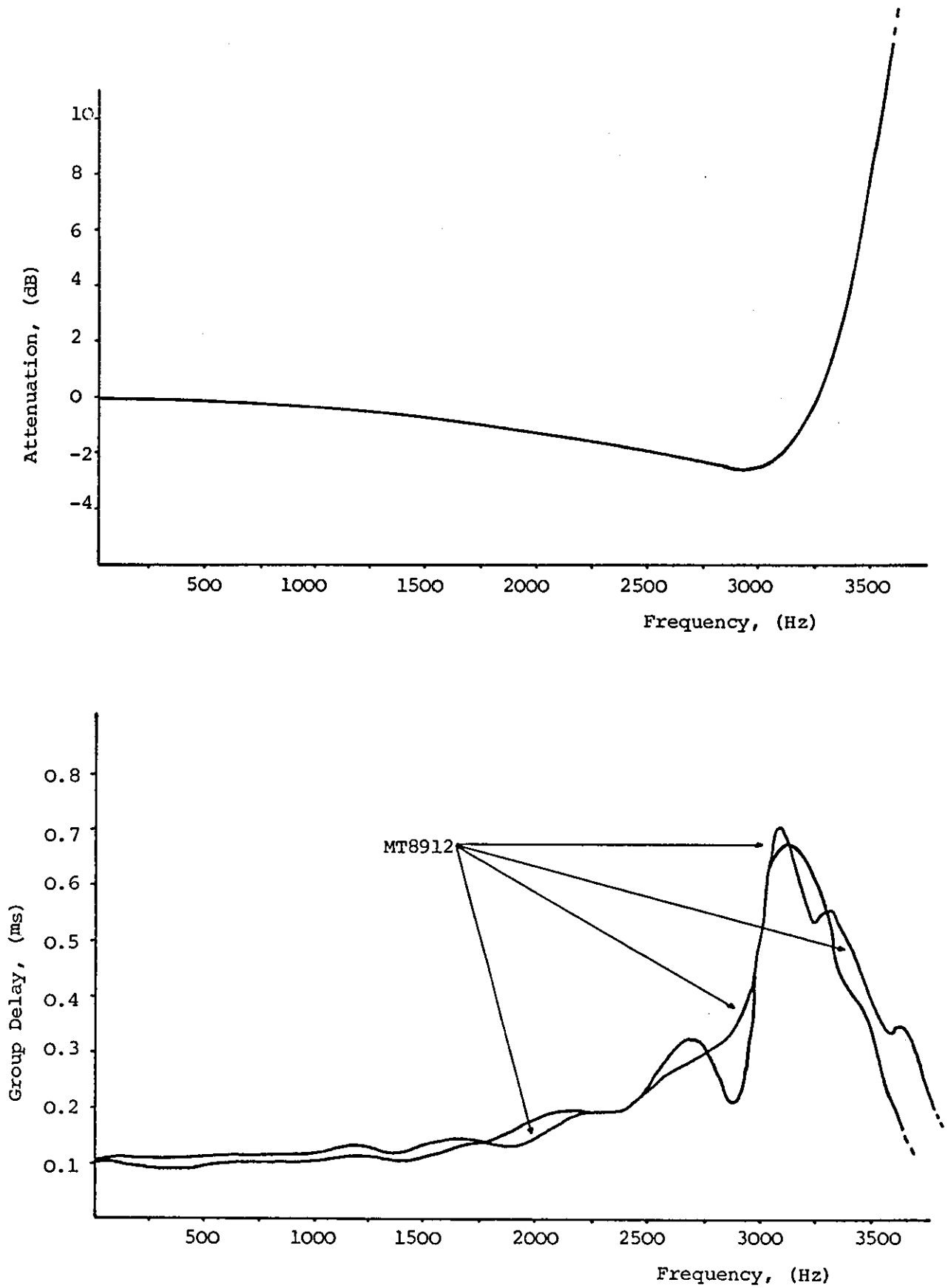


Figure A7.1: Attenuation and group delay characteristics for the MT8912 and I2912 "Receive" filters.

Clock range = 2.048 MHz

Clock frequency = 1.8432 MHz

Digital sampling frequency = 7.2 kHz

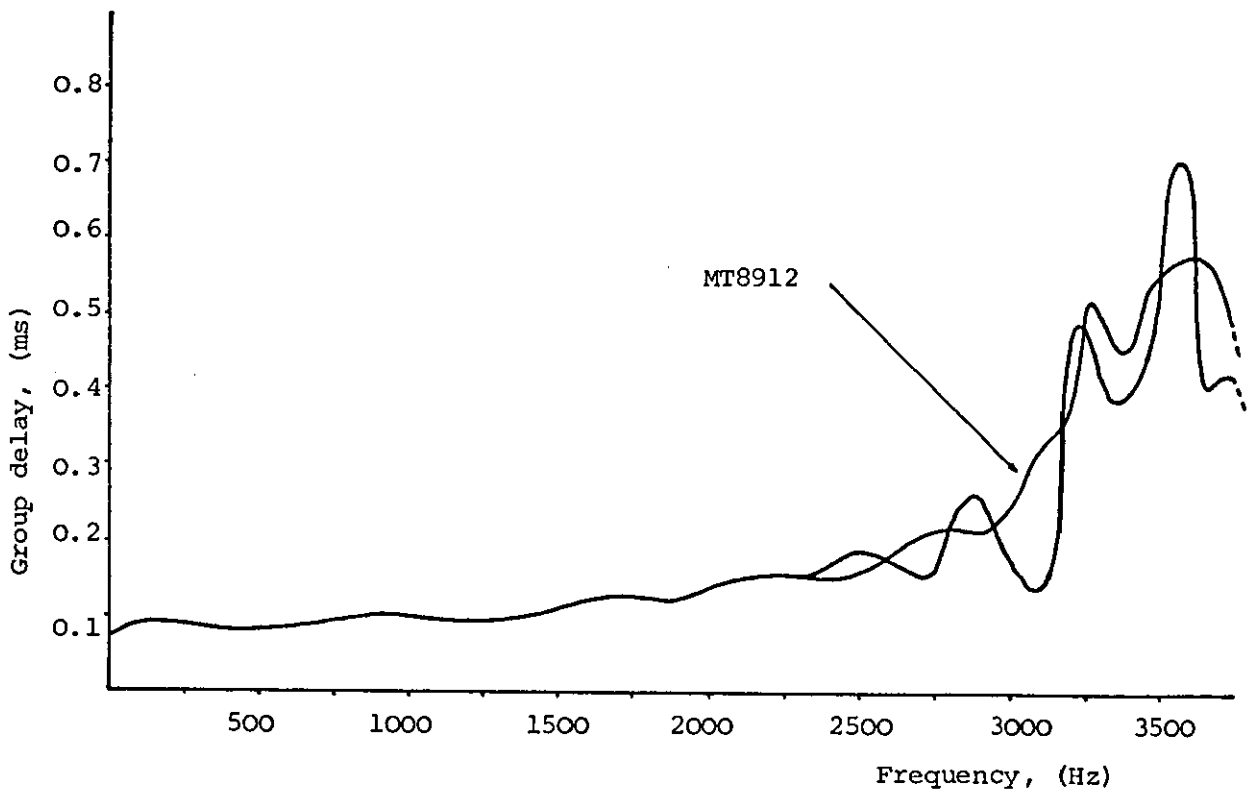
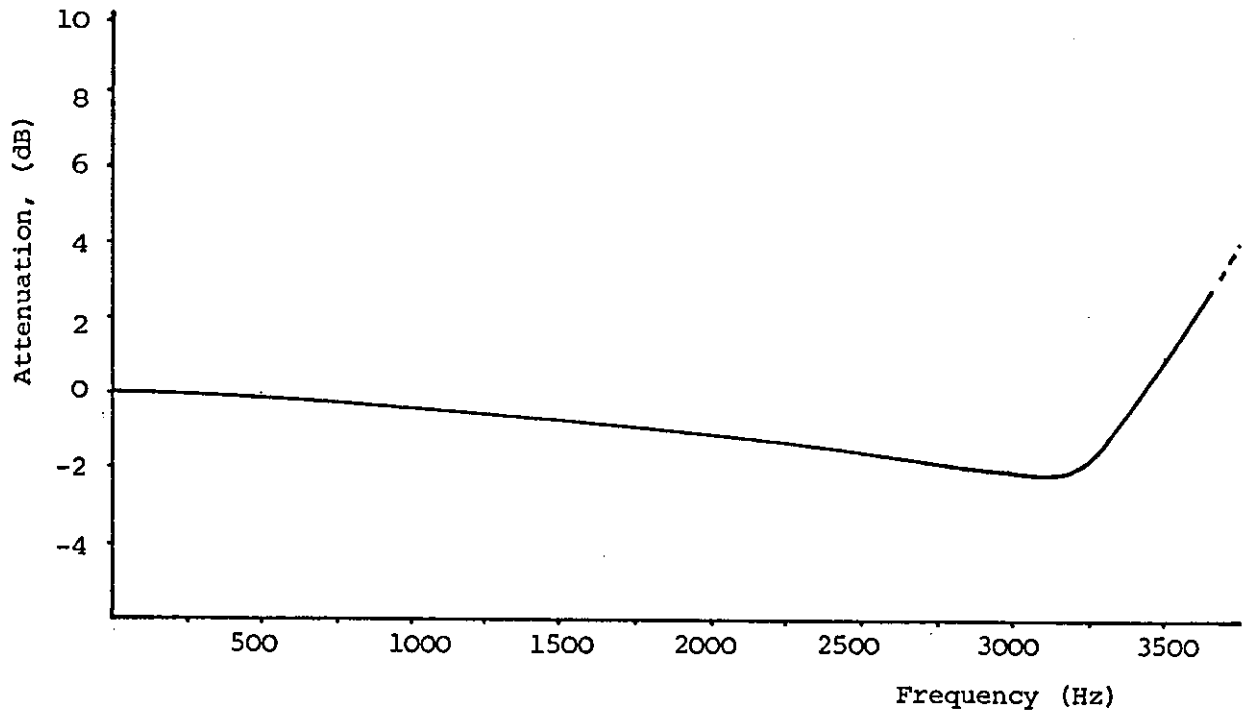


Figure A7.2: Attenuation and group delay characteristics for the MT8912 and I2912 "Receive" filters.

Clock frequency = 2.048Hz
Digital sampling frequency = 8kHz

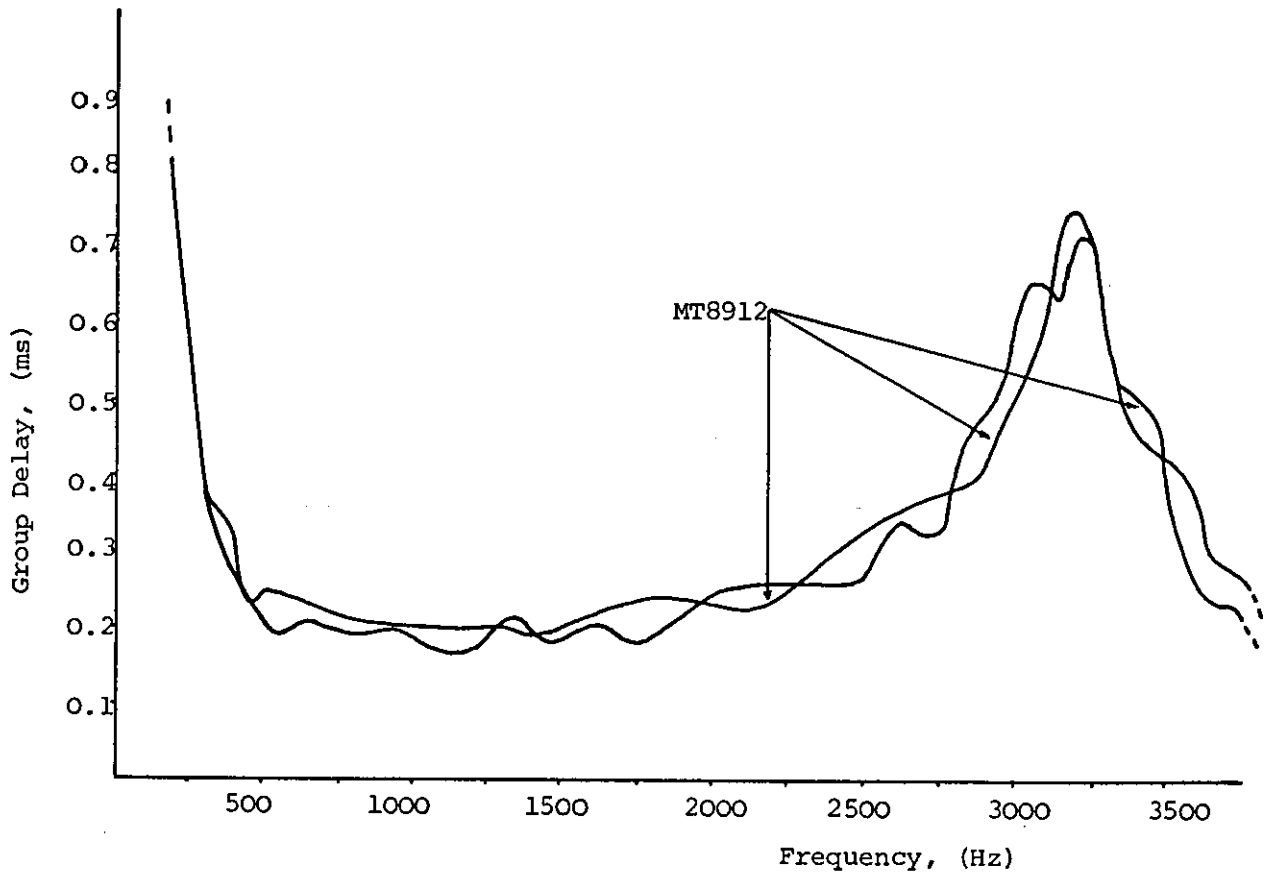
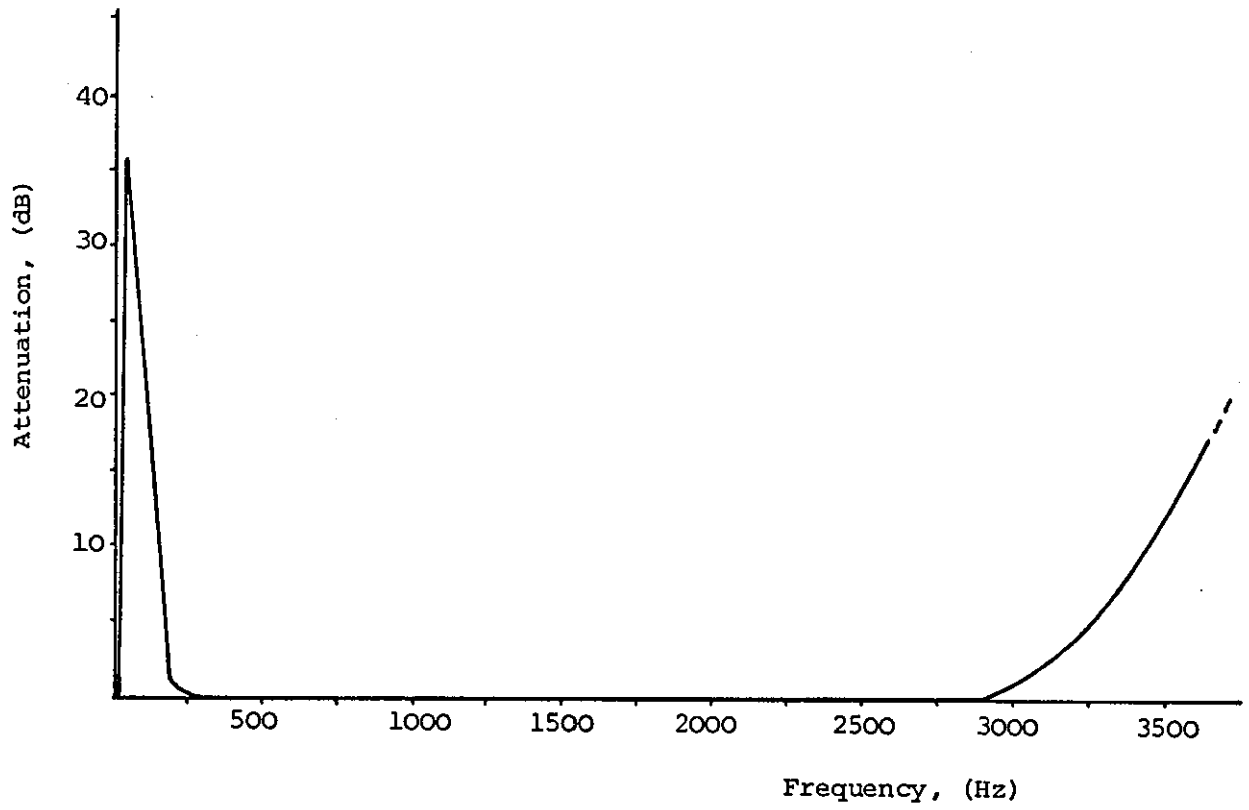


Figure A7.3: Attenuation and group delay characteristics for the MT8912 and I2912 "Transmit" filters.

Clock range = 2.048 MHz

Clock frequency = 1.8432 MHz

Digital sampling frequency = 7.2 kHz

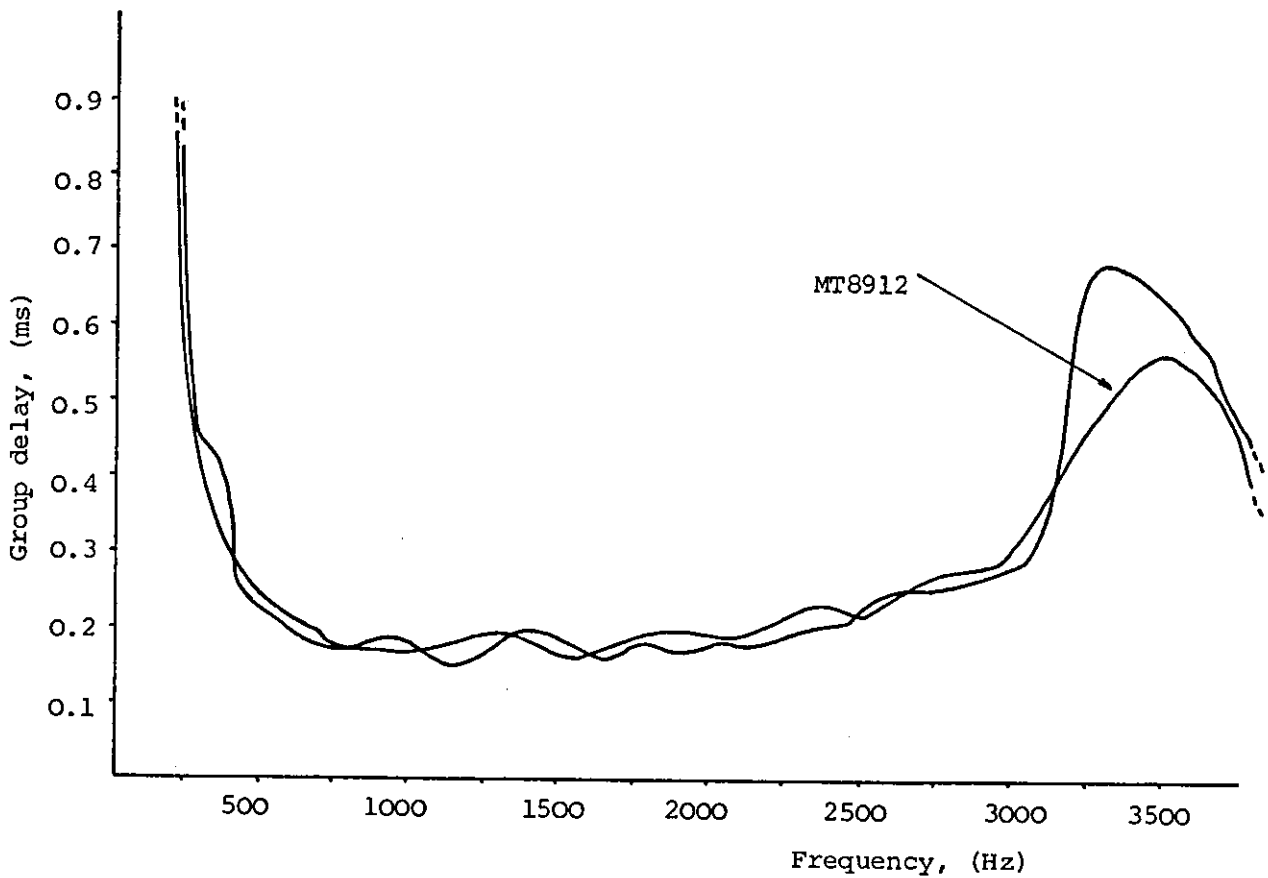
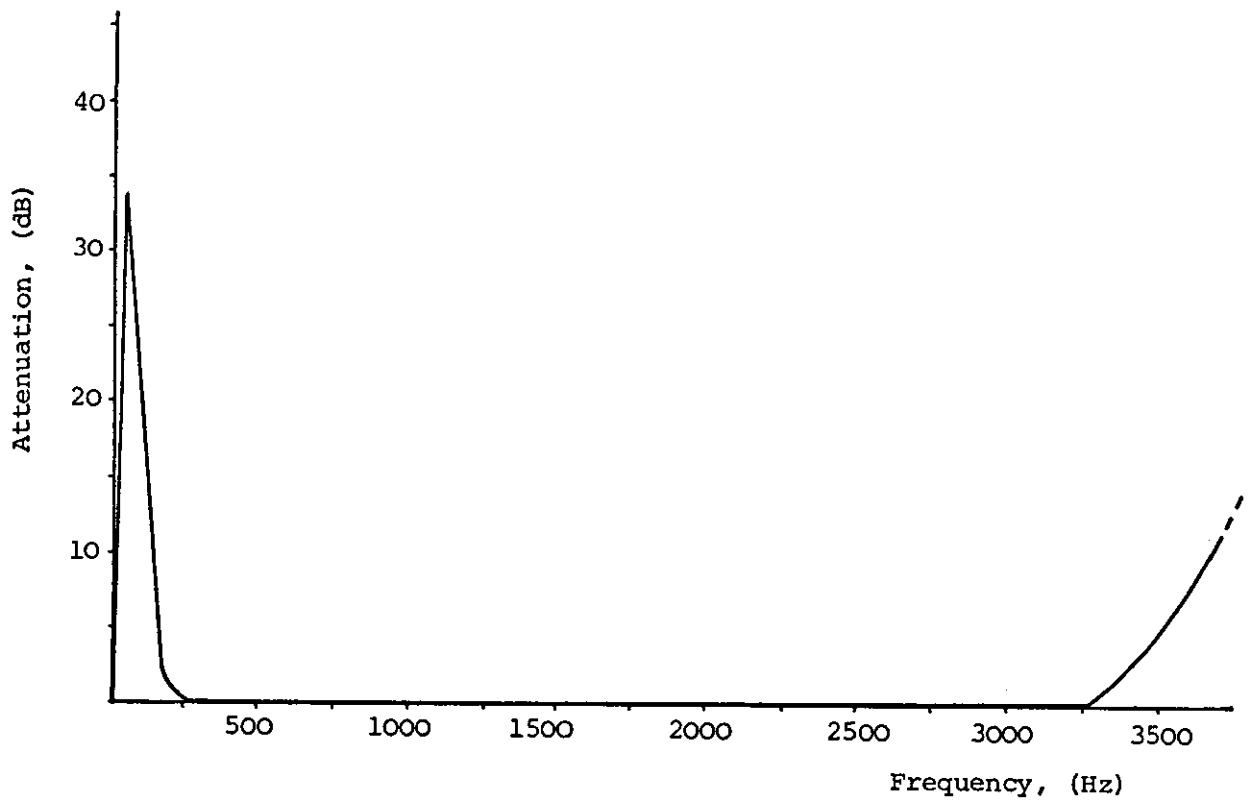


Figure A7.4: Attenuation and group delay characteristics for the MT8912 and I2912 "Transmit" filters.

Clock frequency = 2.048 MHz

Digital sampling frequency = 8kHz

Appendix 8

Relationship between Continuous and Discrete Convolution Operations.

Consider a general waveform, $h(t)$, which is sampled by a sampling waveform, $ss(t)$, where,

$$ss(t) = \sum_{n=-\infty}^{+\infty} \delta(t-nT) \quad (A8.1)$$

$ss(t)$ is therefore a signal comprising a stream of equally spaced unit impulses, separated in time by T sec. The sampled waveform, $h_s(t)$, is therefore,

$$\begin{aligned} h_s(t) &= h(t) \cdot ss(t) \\ h_s(t) &= \sum_{n=-\infty}^{+\infty} h(t) \delta(t-nT) \\ h_s(t) &= \sum_{n=-\infty}^{+\infty} h(nT) \delta(t-nT) \end{aligned} \quad (A8.2)$$

where $h(nT) = h(t)$, at times $t = nT$, for all integer values of n .

From the Convolution Theorem⁽¹⁾,

$$F(h(t) \delta(t-nT)) = H(f) * F(\delta(t-nT)) \quad (A8.3)$$

where $F(\cdot)$ is the Fourier Transform of (\cdot)

$H(f)$ is the Fourier Transform of $h(t)$

and $*$ is the (continuous) convolution operator

From equations (A8.2) and (A8.3), the Fourier Transform of the sampled signal, $H_s(f)$, is,

$$H_s(f) = \sum_{n=-\infty}^{+\infty} H(f) * F(\delta(t-nT)) \quad (A8.4)$$

$$H_s(f) = H(f) * \sum_{n=-\infty}^{+\infty} F(\delta(t-nT)) \quad (A8.5)$$

Since ⁽¹⁾,

$$\sum_{n=-\infty}^{+\infty} F(\delta(t-nT)) = \frac{1}{T} \sum_{m=-\infty}^{+\infty} \delta(f - \frac{m}{T}) \quad (\text{A8.6})$$

then equation (A8.5) becomes,

$$H_s(f) = H(f) * \frac{1}{T} \sum_{m=-\infty}^{+\infty} \delta(f - \frac{m}{T}) \quad (\text{A8.7})$$

If $H(f)$ is a bandlimited function of f such that $H(f) = 0$, $|f| > B$, and if $\frac{1}{T} \geq 2B$, which is the well known Nyquist Sampling Theorem ⁽¹⁾, then the spectrum of $h_s(t)$ will comprise the original signal spectrum, $H(f)$ scaled by the constant $\frac{1}{T}$, plus periodic repetitions of $\frac{1}{T}H(f)$ centred around $f = \frac{m}{T}$ Hz, for all integer values of m . Under these conditions, $H(f)$ can be recovered from $H_s(f)$ and so $h(t)$ can be reconstructed from the sampled signal $h_s(t)$, by passing $h_s(t)$ through an ideal low-pass filter, $Q(f)$, where,

$$\begin{aligned} Q(f) &= T, & -\frac{1}{2T} < f < \frac{1}{2T}, \\ &= 0, & |f| > \frac{1}{2T} \end{aligned} \quad (\text{A8.8})$$

Consequently,

$$H(f) = H_s(f) \cdot Q(f) \quad (\text{A8.9})$$

$$\text{and so } h(t) = h_s(t) * q(t) \quad (\text{A8.10})$$

From equation (A8.2),

$$h(t) = \sum_{n=-\infty}^{+\infty} (h(nT)\delta(t-nT)) * q(t) \quad (\text{A8.11})$$

$$h(t) = \sum_{n=-\infty}^{+\infty} h(nT)q(t-nT) \quad (\text{A8.12})$$

where $q(t)$ is the inverse Fourier Transform of $Q(f)$ and from equation (A8.8)

$$q(t) = \int_{-\infty}^{+\infty} Q(f) e^{j2\pi ft} df$$

$$q(t) = \frac{\sin(\frac{\pi t}{T})}{\frac{\pi t}{T}} \quad (\text{A8.13})$$

Using equation (A8.13), $h(t)$, from equation (A8.12) becomes,

$$h(t) = \sum_{n=-\infty}^{+\infty} h(nT) \frac{\sin\pi(\frac{t-nT}{T})}{\frac{\pi(t-nT)}{T}} \quad (\text{A8.14})$$

which states that a general waveform, $h(t)$, can be represented by the sum of suitably weighted $\frac{\sin x}{x}$ pulses. It should be noted that here sampling is assumed to occur at a rate equal to, or greater than the Nyquist rate and that,

$$\begin{aligned} \frac{\sin\pi(\frac{t-nT}{T})}{\pi(\frac{t-nT}{T})} &= 1, \quad \text{for } t = nT, \\ &= 0, \quad \text{for } t = mT, m \neq n \end{aligned} \quad (\text{A8.15})$$

Consider now two general bandlimited waveforms, $x(t)$ and $y(t)$ and let,

$$w(t) = x(t) * y(t) \quad (\text{A8.16})$$

$$x_i = x(iT) \quad (\text{A8.17})$$

$$y_k = y(kT) \quad (\text{A8.18})$$

where the $\{x_i\}$, $\{y_k\}$ are sample values of $x(t)$ and $y(t)$ at times $t=iT$ and $t=kT$, respectively, where sampling occurs at or greater than the Nyquist rate of either $x(t)$ or $y(t)$, whichever occupies the largest bandwidth.

From equation (A8.14),

$$x(t) = \sum_i x_i \frac{\sin\pi(\frac{t-iT}{T})}{\frac{\pi(t-iT)}{T}} \quad (\text{A8.19})$$

$$\text{and } y(t) = \sum_k y_k \frac{\sin\pi(\frac{t-kT}{T})}{\pi(\frac{t-kT}{T})} \quad (\text{A8.20})$$

so from equations (A8.16), (A8.19) and (A8.20),

$$w(t) = \int_{-\infty}^{\infty} \sum_i x_i \frac{\sin\pi(\frac{\tau-iT}{T})}{\pi(\frac{\tau-iT}{T})} \cdot \sum_k y_k \frac{\sin\pi(\frac{t-kT-\tau}{T})}{\pi(\frac{t-kT-\tau}{T})} d\tau \quad (\text{A8.21})$$

$$= \sum_i \sum_k x_i y_k \int_{-\infty}^{\infty} \frac{\sin\pi(\frac{\tau-iT}{T})}{\pi(\frac{\tau-iT}{T})} \frac{\sin\pi(\frac{t-kT-\tau}{T})}{\pi(\frac{t-kT-\tau}{T})} d\tau \quad (\text{A8.22})$$

Now, since $\frac{\sin x}{x}$ is an even function of x ,

then,

$$\frac{\sin\pi(\frac{t-kT-\tau}{T})}{\pi(\frac{t-kT-\tau}{T})} = \frac{\sin\pi(\frac{\tau-(t-kT)}{T})}{\pi(\frac{\tau-(t-kT)}{T})} \quad (\text{A8.23})$$

Also, if w_ℓ is the sample value of $w(t)$ at a time $t=\ell T$, then equation (A8.22) becomes,

$$w_\ell = \sum_i \sum_k x_i y_k \int_{-\infty}^{\infty} \frac{\sin\pi(\frac{\tau-iT}{T})}{\pi(\frac{\tau-iT}{T})} \cdot \frac{\sin\pi(\frac{\tau-(\ell-k)T}{T})}{\pi(\frac{\tau-(\ell-k)T}{T})} d\tau \quad (\text{A8.24})$$

which, using equation (A8.15), simplifies to,

$$w_\ell = \sum_k x_{\ell-k} y_k \int_{-\infty}^{\infty} \left[\frac{\sin\pi(\frac{\tau-(\ell-k)T}{T})}{\pi(\frac{\tau-(\ell-k)T}{T})} \right]^2 d\tau \quad (\text{A8.25})$$

The integral in equation (A8.25) can be evaluated by making the substitution,

$$x = \pi(\frac{\tau-(\ell-k)T}{T}) \quad (\text{A8.26})$$

such that,

$$\int_{-\infty}^{\infty} \left[\frac{\sin \pi \left(\frac{\tau - (\ell - k)T}{T} \right)}{\pi \left(\frac{\tau - (\ell - k)T}{T} \right)} \right]^2 d\tau = \frac{2T}{\pi} \int_0^{\infty} \left(\frac{\sin x}{x} \right)^2 dx \quad (\text{A8.27})$$

From a standard result ⁽²⁾,

$$\int_0^{\infty} \left(\frac{\sin x}{x} \right)^2 dx = \frac{\pi}{2} \quad (\text{A8.28})$$

and so, from equations (A8.25) - (A8.28),

$$w_{\ell} = T \sum_k x_{\ell-k} y_k \quad (\text{A8.29})$$

Equation (A8.29) shows that in order for discrete-time and continuous-time convolution operations to be equivalent, the sequence obtained by the former operation must be scaled by the constant T . All computer programs involving the convolution of sequences included this necessary scale factor and were also checked for correct operation by verifying results obtained when using well-known functions.

For ease of notation, the convolution of sequences throughout this thesis is distinguished from continuous-time convolution by using the symbol $*$ '. Hence, the convolution of $\{x_i\}$ and $\{y_i\}$ is denoted,

$$\{w_{\ell}\} = \{x_k\} *' \{y_i\} = T \sum_k x_k y_{\ell-k} \quad (\text{A8.30})$$

and so the operator $*$ ' includes the scale factor.

References

1. Brigham, E.O., "The Fast Fourier Transform", Prentice-Hall Inc., 1974.
2. Carlson, A.B., "Communication Systems", McGraw-Hill Inc.,
2nd Edition, 1975.

Appendix 9

The Extended Newton-Raphson Root-Finding Algorithm

The alternative schemes for the determination of the linear pre-filter tap-gains presented in Chapter 7 require the use of a root-finding algorithm which operates on the z-transform of the channel's sampled impulse response. References 1-7 give details of several algorithms which, being very sophisticated, unfortunately involve a relatively large number of computations to achieve the root locations. However, in general, these algorithms achieve convergence to the roots values with relatively few iterations when compared with more conventional methods. As far as this work is concerned, the important factor is the total number of operations, particularly complex multiplications, required to implement the process rather than the mathematical elegance of the solution. For this reason and the fact that the alternative algorithms all appear to suffer equally with deflation inaccuracies and the problem of saddle-point traps⁽³⁾ the relatively simple Newton-Raphson method has been extended to the case of polynomials with complex coefficients. Also, simple modifications have been included to reduce errors due to polynomial deflation and to provide checks and escape routes from saddle-points.

Consider a polynomial $f(z)$, of order g ,

$$f(z) = a_0 + a_1 z + a_2 z^2 + \dots + a_g z^g \quad (\text{A9.1})$$

where in general, the $\{a_i\}$ and $\{z^i\}$ are complex-valued.

The roots of $f(z)$ are those values of z which satisfy,

$$f(z) = 0 \quad (\text{A9.2})$$

If z_0 is a value of z close to one of the g roots of $f(z)$, then

$(z_0 + h)$, for a suitable value of h , will satisfy,

$$f(z_0 + h) = 0 \quad (\text{A9.3})$$

Expanding equation (A9.3) by Taylor's method gives,

$$f(z_0 + h) = f(z_0) + hf'(z_0) + \frac{h^2 f''(z_0)}{2!} + \dots \quad (\text{A9.4})$$

where $f'(z_0)$ denotes the first derivative of $f(z)$ evaluated at $z=z_0$.

If z_0 is close to a root of $f(z)$, h will be a small value so from equations (A9.3) and (A9.4),

$$0 \approx f(z_0) + hf'(z_0)$$

$$\text{and so } h \approx \frac{-f(z_0)}{f'(z_0)} \quad (\text{A9.5})$$

It follows that,

$$z = z_0 + h = z_0 - \frac{f(z_0)}{f'(z_0)} \quad (\text{A9.6})$$

In terms of an iterative algorithm,

$$z_{n+1} = z_n - \frac{f(z_n)}{f'(z_n)} \quad (\text{A9.7})$$

where z_{n+1} is a better approximation of a value of z which satisfies equation (A9.2). Equation (A9.7) gives the Newton-Raphson algorithm for the evaluation of the complex roots of a complex polynomial. Figure A9.1 illustrates the algorithm for the simple case of a real root of a real polynomial, $f(x)$.

For a particular initial "guess" at a value of z , equation (A9.7) is repeated until $(z_{n+1} - z_n) < \text{TOL}$ where TOL is some small complex number; at this point, the process is assumed to have converged to a root of $f(z)$ within the accuracy set by TOL. To ensure the iterative process does not get trapped by a saddle point, $f'(z_n)$ and $f(z_n)$ are checked

after evaluation. At a saddle point, $f'(z_n) = 0$ but, as $f'(z_n)$ will also be zero at a multiple root, the test for a saddle point must also involve $f(z_n)$, which will equal zero at a root but not at a saddle point⁽⁵⁾. If a possible saddle point is located, the search is terminated and restarted at a different value of z_0 . In general, the process will determine the roots of $f(z)$ in no particular order although some element of control is achieved by the selection of z_0 . If z_0 is set to a low value, the roots will be located in order of approximately increasing magnitude.

Once located, the first root of $f(z)$ must be removed from the polynomial to enable the search algorithm to find the next root. The conventional method is to remove the root by dividing $f(z)$ by $(z-\gamma)$, where γ is most recently located root. This process, called polynomial deflation (3), results in a reduced order (deflated) polynomial,

$$f_1(z) = \frac{f(z)}{(z-\gamma)} \quad (\text{A9.8})$$

Provided γ has been accurately located and the division is also accurately achieved, $f_1(z)$ will contain the remaining roots of $f(z)$. The iterative process is then repeated using $f_1(z)$ in place of $f(z)$. In practice, the roots of $f(z)$ can only at best be obtained to the accuracy of the particular processing machine used and so will not be exact. It follows that the deflated polynomials and therefore the roots derived from them will also contain errors. Although the use of composite deflation techniques can be used to reduce the magnitude of the errors⁽³⁾ a much simpler approach is to use the deflated polynomial to find a good approximation to a root which is then used as the starting value in the iterative algorithm

operating on the original polynomial, $f(z)$. Although it requires a few more iterations, this method does produce accurate root values. Problems can arise, however, if roots occur in very tight bunches or in multiples. The former case is only likely over very poor telephone circuits where the performance of the data detector itself will be poor whilst the case of multiple roots is unlikely to be met in practice.

Consider now the z -transform of the sampled impulse response of a baseband channel, $Y(z)$, where,

$$Y(z) = y_0 + y_1 z^{-1} + y_2 z^{-2} + \dots + y_g z^{-g} \quad (\text{A9.9})$$

$Y(z)$ is a polynomial, of order g , in z^{-1} and can be expressed as the function of z^{-1} , $f(z^{-1})$. For the application considered here, it is necessary to locate the values of z for which $f(z^{-1}) = 0$; this can be achieved in two ways.

In the first method, the root-finding algorithm given in equation (A9.7) is applied directly to $f(z^{-1})$ which results in the g values of z^{-1} which satisfy $f(z^{-1}) = 0$. The k roots of $f(z^{-1})$ whose magnitudes are less than unity are therefore the k roots which lie outside the unit circle in the z -plane. Having identified these roots, the processor may then form $Y_2(z)$ and $F_2(z)$ as required in equation (7.6.14) or (7.6.52). The process does not need to calculate the reciprocals of the roots of $f(z^{-1})$. The second method determines directly the values of z for which $f(z^{-1}) = 0$ by using the fact that if the coefficients of a polynomial, $f(x)$, are reversed in order, the roots of the reversed polynomial will be the reciprocals of the roots of $f(x)$. The root-finding algorithm is therefore applied to,

$$f^r(z^{-1}) = y_g + y_{g-1} z^{-1} + y_{g-2} z^{-2} + \dots + y_0 z^{-g} \quad (\text{A9.10})$$

The k roots of $f^r(z^{-1})$ whose magnitudes are greater than unity will lie outside the unit circle; $Y_2(z)$ and $F_2(z)$ can then be calculated as before. Since the components $\{y_i\}$ will all be known to the adjustment system prior to any processing, both methods involve the same number of operations so the choice between them is arbitrary. However, it is interesting to note that the method of Lehmer⁽¹⁾, can apparently locate roots selectively by limiting the search algorithm to values within a circle of specified radius in the complex z -plane. If Lehmer's system can be implemented, it would be sensible to use the first method to locate the k roots of $f(z^{-1})$ which lie outside the unit circle.

To test the operation of the extended Newton-Raphson algorithm when operating on the polynomials $Y(z)$ for various combinations of telephone circuits and equipment filters, a procedure, PROCCOMPLEXROOTS was written using pseudo-structured BASIC for a BBC microcomputer. Details of the procedure appear in reference 9, whilst a specimen program is included as Program 7 in Appendix 10. In terms of computational efficiency, it is important to note that the long division process required at each polynomial deflation actually reduces to shift, multiply and accumulate operations, the number of operations required falling in proportion to the reduced order of the deflated polynomials. (See PROCLDIV, Program 7, Appendix 10). The major drawback of the system is that it requires the evaluation of $f(z_n)/f'(z_n)$ at every iteration. For practical implementation, a further procedure may well be required to calculate the reciprocal of $f'(z_n)$. A further refinement that can be added to the process which will considerably reduce both computational time and effort revolves

around the fact that only the k roots lying outside the unit circle are required here. If it is clear from early iterations of the root-finding algorithm that convergence is towards a root of $f(z)$ whose magnitude is less than unity, then the process may be terminated. However, extreme care must be exercised with this refinement if roots are likely to occur just outside the unit circle as would be the case for a telephone circuit which introduced some phase distortion and severe attenuation distortion⁽⁸⁾.

The total number of operations required by the process to give k roots of a polynomial of order g will obviously depend upon the number of iterations required per root and so cannot be generalised. However, if the average number of iterations per root is given by n , an approximation to the total number of operations is;

a) Real multiplications;

$$n(12kg - 6k^2 + 8k) + 2k(k+1)$$

b) Real Divisions; $2nk$

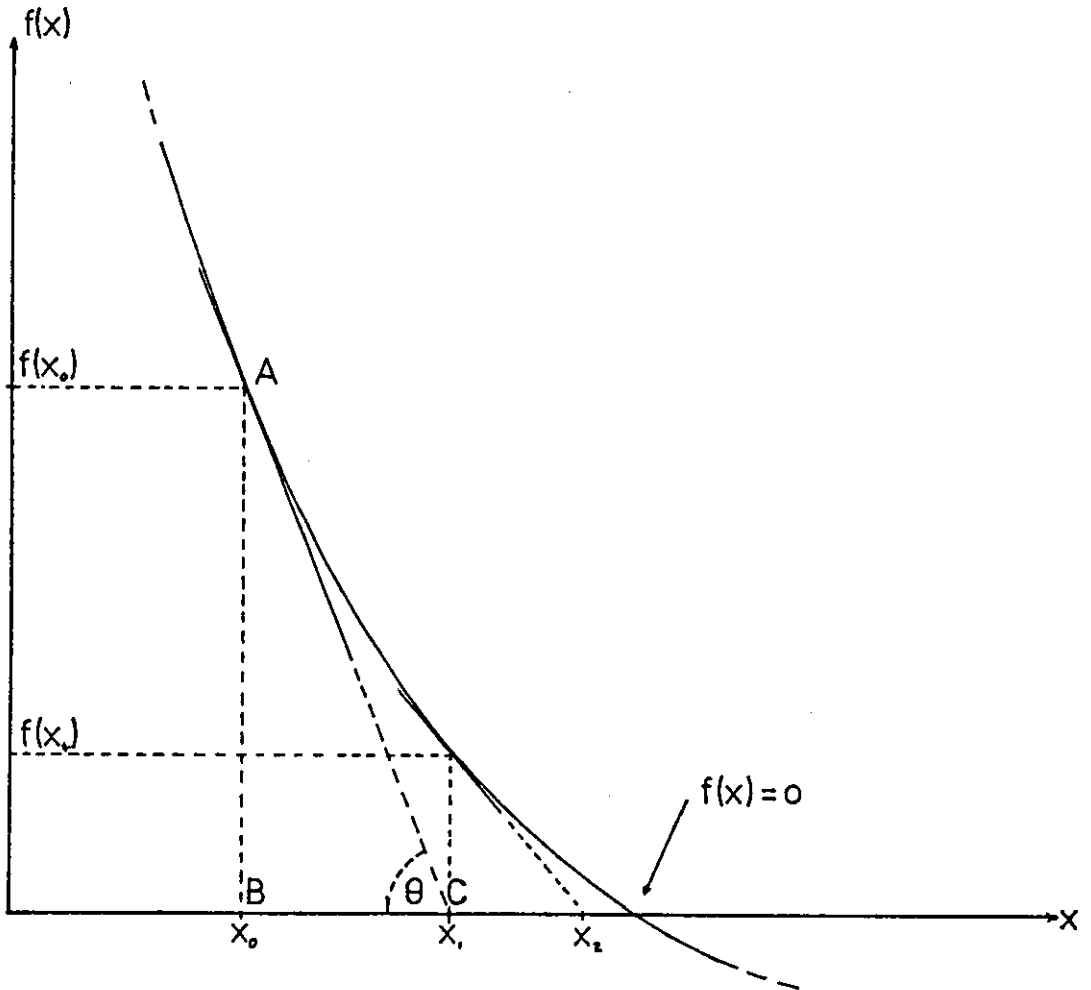
c) Real additions/subtractions;

$$n(10kg - 5k^2 + 5k) + 2k(k+1)$$

References

1. Lehmer, D.H., "A Machine Method for Solving Polynomial Equations", Ass.Comp. Mach.J., Vol.8, pp 151-162, 1961.
2. Adams, D.A., "A Stopping Criterion for Polynomial Root Finding", Comm. of Ass. Comp. Mach., Vol.10, No.10, pp 655-658, Oct.1967.

3. Peters, G. and Wilkinson, J.H., "Practical Problems Arising in the Solution of Polynomial Equations", J. Inst. Maths Applics., Vol.8, pp 16-35, 1971.
4. Grant, J.A. and Hitchins, G.D., "An Always Convergent Minimisation Technique for the Solution of Polynomial Equations", J. Inst. Maths Applics., Vol.8, pp 122-129, 1971.
5. Grant, J.A. and Hitchens, G.D., "Two Algorithms for the Solution of Polynomial Equations to Limiting Machine Precision", The Computer Journal, Vol. 18, No.3, pp 258-264.
6. Ralston, A., "A First Course in Numerical Analysis", McGraw-Hill Inc., 1965.
7. Wilkinson, J.H., "The Algebraic Eigenvalue Problem", Oxford University Press, 1965.
8. Clark, A.P., "Advanced Data Transmission Systems", Pentech Press, 1977.
9. Bateman, S.C., "BBC Microcomputer Software Library, Procedure PROCCOMPLEXROOTS", Internal Communication, Loughborough University of Technology, Sept. 1982.



$$\tan \theta = \frac{AB}{BC} = \frac{f(x_0)}{x_1 - x_0} = -f'(x_0)$$

$$\therefore x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

$$\text{In general, } x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Figure A9.1: Newton-Raphson method applied to a real root of a real polynomial, $f(x)$

Appendix 10Listings of Computer Programs

- Program 1; Simulation of Coding Scheme 3.
- Program 2; Non-recursive digital filter design.
- Program 3; Channel and filter analysis and derivation of sampled impulse responses.
- Program 4; Determines the minimum-phase sampled impulse response of filters and overall channel characteristics. Also a root-finding program.
- Program 5; System simulation for a) NMLH detector
b) NLE
- Program 6; Channel Estimator performance analysis.
- Program 7; Extended Newton-Raphson root-finding routine.
- Program 8; Carrier phase tracking loop 2 simulation.
- Program 9; AGC system simulation.

. NULL

[illegible]

C*****TRANSMITTER CODING*****

ITX=IA(M)*2+IB(M)+1
 NNTX=KCODE(NOTX, ITX)
 NOTX=NNTX
 ISUM=IC(M)*8+ID(M)*4+IE(M)*2+IG(M)
 INSYM=(NNTX-1)*16+ISUM+1
 NS=INSYM
 IRL=R(NS)
 IRS=R(NS)
 IQU=IM(NS)
 IQS=IM(NS)

C INSERT ROTATIONS HERE****

IROT=0
 IF(Z, LT, 100)GOTO 35
 IROT=90
 IRL=-IQS
 IQU=IRS
 IF(Z, LT, 200)GOTO 35
 IROT=180
 IRL=-IRS
 IQU=-IQS
 IF(Z, LT, 300)GOTO 35
 IROT=270
 IRL=IQS
 IQU=-IRS

35 IF(IRL, EQ, -1)IRL=2
 IF(IRL, EQ, -3)IRL=4
 IF(IRL, EQ, -5)IRL=6
 IF(IRL, EQ, -7)IRL=8
 IF(IQU, EQ, -1)IQU=2
 IF(IQU, EQ, -3)IQU=4
 IF(IQU, EQ, -5)IQU=6
 IF(IQU, EQ, -7)IQU=8
 NNS=KSY(IRL, IQU)

C *****AT THE RECEIVER*****

IRX=IA(NNS)*2+IB(NNS)+1
 NNRX=KCODE(NORX, IRX)
 NORX=IRX
 ISUM=IC(NNS)*8+ID(NNS)*4+IE(NNS)*2+IG(NNS)
 IDSYM=(NNRX-1)*16+ISUM+1
 INS=IDSYM
 IF(M, NE, INS) IER=IER+1
 WRITE(1, 110)Z, R(M), IM(M), IRS, IQS, R(INS), IM(INS), IROT, M, INS,
 1 IER

110 FORMAT(7I4, I5, 3X, 2I5, 9X, I5)

60 Z=Z+1

IF(400-Z)80, 80, 1

80 STOP

END

BOTTOM

PROGRAM 2

```

SEND TO (ED,SEMIGROUPUSER:AXXX)
DUMP ON (ED,PROGRAM USER)
WORK (ED,WORKFILEUSER)
RUN

LIBRARY(ED,SUBGROUPNAGF)
LIBRARY(ED,SUBGROUPGINO)
PROGRAM(DFILT)
COMPRESS INTEGER AND LOGICAL
INPUT 1=CR0
OUTPUT 2=LPO
OUTPUT 7=CP0
TRACE 2
END

```

```

MASTER NON RECURSIVE DIGITAL FILTER
COMPLEX XVECT,XVAL,XSUM,XROOT
DIMENSION AA(90),GD(90),GDN(90),X(80),Y(80)
DIMENSION XVECT(80),AR(80),AC(80),C(10)
DIMENSION AMP(256),BMP(256),ARR(70),ACC(70),DIST(70)
DIMENSION IAMP(80),IBMP(80)
LOGICAL TRAN
DATA PI/3.14159/,N1/256/,M1/9/,FMAX/14400/
DF=FMAX/N1
READ(1,180) NFDS
180 FORMAT(10)
CALL C1051N
80 READ(1,182) ITRAN,ICUT,IWL,ITK
182 FORMAT(410)
WRITE(2,184) ITRAN,ICUT,IWL,ITK
184 FORMAT(1H1,10X,'ITRAN=',14,' ICUT=',14,' IWL=',14,' ITK=',14)
DO 70 I=1,50
70 AR(I),AC(I),ARR(I),ACC(I)=0.0
C GENERATE AMPLITUDE RESPONSE
THETA=PI/(2*ITRAN)
IK=ICUT-ITRAN
IKK=ICUT+ITRAN
DO 1 I=1,IK
1 AMP(I)=1.0
DO 2 I=IK+1,IKK-1
2 AMP(I)=(1+COS((I-1K)*THETA))/2

```

```

DO 4 I=1,N1
3 AMP(I)=0.0
I=N1
DO 4 K=2,126
AMP(I)=AMP(K)
I=I-1
4 CONTINUE
DO 5 I=1,N1
5 AMP(I)=0.0
TRAN=.TRUE.
CALL C06ABF(AMP,BMP,N1,TRAN,M1,C)
C APPLY WINDOW FUNCTION
ARG=PI/IWL
DO 6 I=1,IWL-1
BMP(I)=0.0
6 AMP(I)=AMP(I)*(0.54+0.46*COS(I*ARG))
DO 7 I=IWL,N1
BMP(I)=0.0
7 AMP(I)=0.0
C END OF WINDOWING
NPTS=2*IWL-3
J=IWL
DO 8 I=1,IWL-1
J=J-1
AR(I)=AMP(J)
ACC(I)=BMP(J)
8 CONTINUE
J=1
DO 9 I=IWL,NPTS
J=J+1
AR(I)=AMP(J)
9 ACC(I)=BMP(J)
DO 11 I=1,NPTS
11 ARR(I),ACC(I)=0.0
TOL=X02AAF(0)
IFAIL=0
MM=NPTS
CALL C02ADF(AR,AC,MM,ARR,ACC,TOL,IFAIL)
WRITE(2,201) MM,IFAIL
201 FORMAT(1H0,5X,' N= ',113,' IFAIL= ',113)
DO 83 I=1,NPTS
DIS=ARR(I)**2+ACC(I)**2
IF(1-DIS)81,82,82
81 AR(I)=ARR(I)/DIS
ACC(I)=ACC(I)/DIS
GO TO 83
82 AR(I)=ARR(I)
ACC(I)=ACC(I)
83 DIST(I)=SQRT(DIS)

XVECT(1)=CMPLX(1.0,0.0)
DO 85 I=2,NPTS
85 XVECT(I)=CMPLX(0.0,0.0)
DO 87 J=1,NPTS
XSUM=CMPLX(0.0,0.0)
XROOT=CMPLX(-AR(J),-AC(J))
J=1
86 XVAL=XVECT(I)
XVECT(I)=XVECT(I)+XSUM
I=I+1
XSUM=XVAL*XROOT

```

```

      IF (1-NPTS) 86, 86, 87
87  PEAK=0.0
      DO 89 I=1, NPTS
        AR(I)=RFAL(XVECT(I))
        AC(I)=AIMAG(YVECT(I))
        IF (AR(I)-PEAK) 89, 89, 88
88  PEAK=AR(I)
        IPK=I
89  CONTINUE
      DO 90 I=1, NPTS
        AR(I)=AR(I)/PEAK
90  AC(I)=AC(I)/PEAK
        AMAX=0.0
        DO 20 I=1, ITK
          20  AMAX=AMAX+7*ABS(AR(I))
          DO 21 I=ITK+1, NPTS
            21  AR(I), AC(I)=0.0
          SCALE=32767/AMAX
          DO 22 I=1, ITK
            AR(I)=ANINT(AR(I)*SCALE)
          22  WRITE(2,203) I, AR(I)
          203  FORMAT(1H, 10X, 14, 12X, F10.2)
          DO 50 I=1, 256
            AMP(I)=0.0
            RMP(I)=0.0
          50  CONTINUE
            PK=0.0
            DO 60 I=1, NPTS
              IF (AR(I) LE PK) GOT060
              PK=AR(I)
              IPPK=I
            60  CONTINUE
              J=1
              I=IPPK
            61  AMP(J)=AR(I)
              INI=1
              J=J+1
              IF (I LT ITK+1) GOT061
              J=256
              IF IPPK=1
            71  AMP(J)=AR(I)
              JEJ=1
              I=I-1
              IF (I GT 0) GOT071
              TRAN=.FALSE.
              CALL C06ABF(AMP, RMP, N1, TRAN, M1, C)
              DO 17 I=1, 81.1
                FREQ=(I-1)*DF
                SIZE=AMP(I)
                GAIN=SQRT(SIZE**2+BMP(I)**2)
                AMP(I)=20*AI0G10(GAIN)
                IF (I LE 1) DATT=AMP(I)
                AMP(I)=AMP(I)-DATT
            17  BMP(I)=ATAN(RMP(I)/SIZE)
                DF1=6.25
                DF=56.25
                DF2=DF-DF1/2
                K=1
                J=1
                I=1
            300  SLOPF=(AMP(I)-AMP(J+1))/DF
                DFF=DF-(K*DF1)

```

```

AA(I)=A(I)-(SLOPE*DEF)
J=J+1
I=I+1
K=K+1
IF(DEF)301,301,300
301 J=J-1
K=1
IF(79=(J+1)) 302,300,300
302 J=1
305 RBMP=BMP(J+1)
IF(BMP(J).LT.0.0.AND.BMP(J+1).LT.0.0.OR.BMP(J).GE.0.0.AND.
1 BMP(J+1).GE.0.0) GOTO 303
IF(ABS(BMP(J))+ABS(BMP(J+1)).LE.PI/2) GOTO 303
RBMP=BBMP+P
303 GD(J)=(RBMP-BMP(J))/(DF*2*PI)
J=J+1
IF(79=(J+1)) 304,305,305
304 J=1
I=1
K=1
307 SLOPE2=(GD(J+1)-GD(J))/DF
DFF1=DF-(K*DEF1)
GDN(I)=GD(J)+(SLOPE2*DFF1)
J=J+1
I=I+1
K=K+1
IF(DFF1)308,308,307
308 J=J-1
K=1
IF(79=(J+1)) 309,307,307
309 GDNN=0.0
ZAMP=0.0
DO 310 I=1,79
IF(AA(I).LT.ZAMP) GOTO 311
ZAMP=AA(I)
311 IF(GDN(I).GT.GDNN) GOTO 310
GDNN=GDN(I)
310 CONTINUE
DO 320 I=1,79
AA(I)=AA(I)-ZAMP
GDN(I)=GDN(I)-GDNN
FR=50*I
320 WRITE(2,204)FR,AA(I),GDN(I)
204 FORMAT(1H,10X,F9.4,5X,F9.4,5X,F11.6)
C*****GRAPHICS PLOT*****
CALL DEVPAP(210.0,297.0,1)
CALL WINDOW(2)
CALL MOVTO2(0.0,0.0)
CALL LINTO2(0.0,297.0)
CALL LINTO2(210.0,297.0)
CALL LINTO2(210.0,0.0)
CALL LINTO2(0.0,0.0)
CALL MOVTO2(28.0,3.0)
CALL CHAHOL(3,0)
CALL CHAHOL(1,ITRAN,1)
CALL CHAINT(ITRAN,3)
CALL CHAHOL(1,ICUT,1)
CALL CHAINT(ICUT,3)
CALL CHAHOL(1,IWL,1)
CALL CHAINT(IWL,3)
CALL CHAHOL(1,ITK,1)
CALL CHAINT(ITK,3)

```

```

CALL MOVTO2(130.0,7.0)
CALL CHASIZ(2.0,2.0)
CALL CHAHOL('F*LFREQUENCY IN *UH*LZ*')
CALL MOVTO2(130.0,157.0)
CALL CHAHOL('F*LFREQUENCY IN *UH*LZ*')
CALL CHAANG(00.0)
CALL MOVTO2(20.0,60.0)
CALL CHAHOL('A*LTENUATION IN D*UR*')
CALL MOVTO2(20.0,200.0)
CALL CHAHOL('G*LRUP *UH*DELAY IN M*US*LECS*')
V(1)=0.0

```

```
DO 700 I=2,79
```

```
V(I)=GDN(I-1)*1000
```

```
IF(V(I).GT.4.0) V(I)=4.0
```

```
700 IF(V(I).LT.-2.0) V(I)=-2.0
```

```
DO 710 I=1,79
```

```
710 X(I)=(I-1)*50
```

```
CALL CHAANG(0.0)
```

```
CALL WINDO2(10.0,200.0,153.5,290.0)
```

```
CALL GRAF(X,Y,79,0)
```

```
V(1)=0.0
```

```
DO 720 I=2,79
```

```
V(I)=AA(I-1)
```

```
IF(V(I).LT.-50.0) V(I)=-50.0
```

```
720 IF(V(I).GT.1.0) V(I)=1.0
```

```
CALL WINDO2(10.0,200.0,6.0,146.0)
```

```
CALL GRAF(X,Y,79,0)
```

```
C*****FORMAT FOR CARD O/P SUITABLE FOR IMPS90*****
```

```
AMP(36)=0.0
```

```
RMP(36)=GDN(1)
```

```
DO 600 I=1,36
```

```
AMP(36+I)=-AA(I)*100
```

```
600 RMP(36+I)=GDN(I)*100000
```

```
DO 610 I=1,35
```

```
AMP(I)=AMP(72-I)
```

```
610 RMP(I)=RMP(72-I)
```

```
DO 620 I=73,75
```

```
AMP(I)=-AA(I-36)*100
```

```
620 RMP(I)=GDN(I-36)*100000
```

```
DO 660 I=1,75
```

```
IAMP(I)=INT(AMP(I))
```

```
660 IBMP(I)=INT(RMP(I))
```

```
WRITE(7,641)ITRAN,ICUT,IWL,ITK
```

```
WRITE(7,640)(IAMP(I);I=1,75)
```

```
WRITE(7,640)(IBMP(I);I=1,75)
```

```
640 FORMAT(15I4)
```

```
641 FORMAT(4I8)
```

```
NFDS=NFD5-1
```

```
IF(NFDS)500,500,490
```

```
490 CALL PICCLE
```

```
GOTO 80
```

```
500 CALL DEVEND
```

```
STOP
```

```
END
```

```
* LENGTH 2269, NAME NONRECURSIVEDIGITALFILTER
```

```
FINISH
```

PROGRAM 3

```

SEND TO (ED,SEMICOMPUUSER,AXXX)
DUMP ON (ED,PROGRAM USER)
WORK (ED,WORKFILEUSER)
RUN

```

```

LIBRARY(FD,SUBGROUPNAGF)
LIBRARY(FD,SUBGROUPGINO)
PROGRAM(IMPS)
COMPRESS INTEGER AND LOGICAL
INPUT 1=CR0
OUTPUT 2=LP0
OUTPUT 7=CP0
TRACE 2
END

```

MASTER CHANNEL ANALYSIS

LOGICAL TRAN

```

DIMENSION A(1024),B(1024),ATT(151),DLV(151),AMP(151),ANG(151)
DIMENSION IAT(75),IDY(75),IFAT(75),IFDY(75),C(12),ST(400)
DIMENSION X(1024),Y(1024),Z(1024),RSAM(60),QSAM(60),PCT(60)
DATA NECHO/0/,EKDH/-80.0/,EKMS/2.0/
DATA NSAM/75/,CFREQ/1800.0/,NCH/5/
DATA BAUD/3200.0/,NDS/40/,TPHA/0.0/,THETA/0.0/,SCALE/1.0/
CALL C1051N
READY(1,100) IFILT
PEAD(1,101)(IFAT(I),I=1,NSAM)
READ(1,101)(IFDY(I),I=1,NSAM)
PI=3.141592654
K1=INT(CFREQ/50.0)
WRITE(2,209)(IFAT(I),I=1,NSAM)
WRITE(2,209)(IFDY(I),I=1,NSAM)
K2=2*K1
DF=PI/40.0
DT=1000.0/BAUD
T=40.0/1024.0
ASR=SCALE*SIN(THETA*PI/180.0)
ACR=SCALE*COS(THETA*PI/180.0)
1 READ(1,100) ICHAN
READ(1,101)(IAT(I),I=1,NSAM)
READ(1,101)(IDY(I),I=1,NSAM)
IAT=IAT(K1)+IFAT(K1)
IDY=IDY(K1)+IFDY(K1)
I=1

```

```

DO 2 J=2,150.2
ATT(J)=(NAT-IAT(I)-IFAT(I))/200.0
DLY(J)=(IDY(I)+IFDY(I)-NDY)/200.0

```

```

2 I=I+1
ATT(1)=3*ATT(2)-ATT(4)
DLY(1)=3*DLY(2)-DLY(4)
DO 3 J=3,149.2
ATT(J)=ATT(J-1)+ATT(J+1)
DLY(J)=DLY(J-1)+DLY(J+1)
3 ATT(151)=3*ATT(150)-ATT(148)
DLY(151)=3*DLY(150)-DLY(148)
DO 4 J=2,150.2
ATT(J)=ATT(J)*2
4 DLY(J)=DLY(J)*2

```

C. CALCULATE PHASE AND AMPLITUDE FROM GROUP DELAY AND ATTENUATION

```

ANG(K2)=0.0
DO 5 J=K2,150
5 ANG(J+1)=ANG(J)+(DLY(J)+DLY(J+1))*DF
J=K2
6 ANG(J-1)=ANG(J)-(DLY(J)+DLY(J-1))*DF
J=J-1
IF(J-1)7,7,6
7 DO 8 J=1,151
8 AMP(J)=EXP10(ATT(J)/20.0)

```

C. FORMAT AMPLITUDE AND PHASE DATA FOR F.F.T

```

I=1
DO 10 J=K2,151
A(I)=AMP(J)*COS(ANG(J))
R(I)=AMP(J)*SIN(ANG(J))
10 I=I+1
II=1026-K2
DO 11 I=1,II
11 A(I),B(I)=0.0
DO 12 J=1,K2-1
A(II)=AMP(J)*COS(ANG(J))
B(II)=AMP(J)*SIN(ANG(J))
12 II=II+1

```

C. CALCULATE F.F.T. AND ORDER TIME DOMAIN DATA

```

TRAN=.TRUE.
CALL COGABE(A,B,1024,TRAN,11,C)
I=1
DO 14 J=769,1024
X(I)=A(J)
Y(I)=B(J)
14 I=I+1
DO 15 J=1,768
X(I)=A(J)
Y(I)=B(J)
15 I=I+1
DO 16 J=1,1024
16 A(J),B(J)=0.0
ST(1)=3.0
ST(2)=0.0
DO 17 J=3,400
17 ST(J)=0.2

```

C. ADD ECHO IF REQUIRED

```

IF(NECHO)24,24,21
21 DO 22 K=1,NECHO
NDLY=NINT(EKMS*K/T)
PHA=NDLY*T*ECAR*PI/500.0
ZEK=COS(PHA)*EXP10(EKDB+K/20.0)
OEK=SIN(PHA)*EXP10(EKDB+K/20.0)

```

```

      I=1
      DO 22 J=NDIV,1024
        A(J)=A(I)+ZFK*X(I)-QFK*Y(I)
        B(J)=B(I)+ZFK*Y(I)+QFK*X(I)
      22 I=I+1
C   END OF ECHO ADDITION SECTION
      24 TENG,EPK=0.0
      DO 25 J=1,1024
        A(J)=A(I)+X(J)
        B(J)=B(I)+Y(J)
        ENEG=A(I)**2+B(J)**2
        IF(ENEG.GT.EPK) EPK=ENEG
      25 TENG=TENG+ENEG
C   RECALCULATE ATTENUATION CHARACTERISTIC FOR ECHO
      IF(NECHO.LE.10) GO TO 30
      I=1
      DO 26 J=769,1024
        X(J)=A(I)
        Y(J)=B(I)
      26 I=I+1
      I=257
      DO 27 J=1,768
        X(J)=A(I)
        Y(J)=B(I)
      27 I=I+1
      TRAN=.FAISE
      CALL C06ABF(X,Y,1024,TRAN,11,C)
      IF1
      DO 28 J=K2,151
        ATT(J)=10.0*A1OG10(X(I)**2+Y(I)**2)
      28 I=I+1
      I=1026-K2
      DO 29 J=1,K2-1
        ATT(J)=10.0*A1OG10(X(I)**2+Y(I)**2)
      29 I=I+1
      30 CONTINUE
C   SAMPLE IMPULSE RESPONSE AT REQUIRED BAUD RATE
      I=180
      PMIN=0.0001*EPK
      31 I=I+1
      IF((A(I)**2+B(I)**2).LT.PMIN) GO TO 31
      I=I-12
      TIM=12*T
      TM=TPHA+DT
      32 I=I+1
      TIM=TIM+T
      IF(TIM.I.T.TM) GO TO 32
      ENORM=1.0
      FPK=FPK+FNORM*ENORM
      DO 34 K=1,NDS
        FRA=(TM-TIM)/T
        RSAM(K)=(A(I)+FRA*(A(I)-A(I-1)))*ENORM
        OSAM(K)=(B(I)+FRA*(B(I)-B(I-1)))*ENORM
        PCT(K)=100*SQR((RSAM(K)**2+OSAM(K)**2)/EPK)
        IF(FRA.LE.-0.5) ST(I-181)=0.25+PCT(K)*0.0025
        IF(FRA.GT.-0.5) ST(I-180)=0.25+PCT(K)*0.0025
        TM=TM+DT
      33 I=I+1
      TIM=TIM+T
      IF(TIM.I.T.TM) GO TO 33
      34 CONTINUE
      ACR=SGA,F+CDS(THETA)

```

```

      ASI=SCALE*SIN(THETA)
C  WRITE OUT CHANNEL SAMPLES DATA
      WRITE(2,200) ICHAN,IFILT
      IF(ECHO.GT.0) WRITE(2,201) NECHO,EKMS,EKDB
      WRITE(2,202) NDS,BAUD
      WRITE(2,205) TPHA
      WRITE(2,206) THETA,SCALE
      WRITE(2,203)
      DO 40 I=1,NDS
      RSN=RSAM(I)
      RSAM(I)=RSN*ACR-QSAM(I)*ASR
      QSAM(I)=RSN*ASR+QSAM(I)*ACR
40  WRITE(2,204) RSAM(I),QSAM(I),PCT(I)
C  WRITE DATA OUT ON CARDS
      WRITE(7,207) ICHAN,IFILT,BAUD,NDS,TPHA,THETA,SCALE
      WRITE(7,208) (RSAM(I),I=1,NDS)
      WRITE(7,208) (QSAM(I),I=1,NDS)
C  FORMAT AND PLOT GRAPHICS PAGE
      DO 60 I=1,161
60  X(I)=(I-1)*25.0
      CALL DEVPAP(210,0,297.0,1)
      CALL WINDOW(2)
      CALL MOVTO2(0.0,0.0)
      CALL LINTO2(0.0,297.0)
      CALL LINTO2(210,0,297.0)
      CALL LINTO2(210,0,0.0)
      CALL LINTO2(0.0,0.0)
      CALL MOVTO2(30.0,20.0)
      CALL CHAHOL(3,0)
      CALL CHAHOL('FIGURE CHARACTERISTICS OF CHANNEL*.')
      CALL CHAINT(ICHAN,3)
      CALL CHAHOL('WITH FILTER*.')
      CALL CHAINT(IFILT,4)
      CALL CHAINT(2,0,2.0)
      CALL MOVTO2(90.0,30.0)
      CALL CHAHOL('TIME IN H*US*LECS*.')
      CALL MOVTO2(150.0,45.0)
      CALL CHAHOL('SAMPLING INSTANTS*.')
      CALL MOVTO2(150.0,91.0)
      CALL CHAHOL('QUADRATURE RESPONSE*.')
      CALL MOVTO2(153.0,135.0)
      CALL CHAHOL('INPHASE RESPONSE*.')
      CALL MOVTO2(50.0,181.0)
      CALL CHAHOL('FREQUENCY IN *UH*LZ*.')
      CALL CHAANG(90.0)
      CALL MOVTO2(30.0,205.0)
      CALL CHAHOL('ATTENUATION IN D*UB*.')
      IF(NECHO.LE.0) GO TO 61
      CALL CHAANG(0.0)
      CALL MOVTO2(130.0,250.0)
      CALL CHAHOL('CHANNEL WITH ECHO*.')
      CALL MOVTO2(130.0,240.0)
      CALL CHAHOL('ECHO DELAY = M*US*LECS*.')
      CALL MOVTO2(154.0,240.0)
      IMS=IFIX(EKMS)
      CALL CHAINT(IMB,4)
      CALL MOVTO2(130.0,230.0)
      CALL CHAHOL('ATTENUATION = D*UB*.')
      CALL MOVTO2(156.0,230.0)
      IDB=IFIX(EKDB)
      CALL CHAINT(IDB,5)
      CALL MOVTO2(130.0,220.0)

```

```

CALL CHAHOL('R*LEPEATED TIMES*')
CALL MOVTO2(148,0,220,0)
CALL CHAINT(NFCHO,3)
GO TO 64
61 CALL MOVTO2(120,0,205,0)
CALL CHAHOL('G*LRDUP *UD*LELAY IN *H*US*LEGS*')
CALL CHAANG(0,0)
CALL MOVTO2(140,0,181,0)
CALL CHAHOL('F*LRFEQUENCY IN *UH*LZ*')
V(1)=-3.0
DO 62 I=2,152
V(I)=D1*V(I-1)
IF(V(I).GT.7.0) V(I)=7.0
62 IF(V(I).LT.-3.0) V(I)=-3.0
DO 63 I=153,161
63 V(I)=7.0
CALL WINDO2(110,0,197,0,180,0,270,0)
CALL GRAF(X,V,161,0)
C ATTENUATION CHARACTERISTIC GRAPH
64 V(1)=-14.90
DO 67 I=2,152
V(I)=-ATT(I,1)
IF(V(I).GT.35.0) V(I)=35.0
67 IF(V(I).LT.-15.0) V(I)=-15.0
DO 68 I=153,161
68 V(I)=35.0
CALL WINDO2(20,0,107,0,180,0,270,0)
CALL GRAF(X,V,161,0)
C IMPULSE RESPONSE GRAPHS
DO 70 I=1,385
70 X(I)=(I-1)*T
ENORM=1/N/SQRT(TENG*T*2.4)
DO 71 I=2,384
V(I)=A(180+I)*ENORM*2.0
71 Z(I)=B(180+I)*ENORM*1.0
V(1)=3.0
Z(1)=0.0
V(385)=0.0
Z(385)=3.0
CALL WINDO2(20,0,197,0,30,0,180,0)
CALL GRAF(X,V,385,0)
CALL GRAF(X,Z,385,0)
CALL GRAF(X,ST,385,0)
NCH=NCH+1
IF(NCH)90,90,80
80 CALL PICCLE
GO TO 1
90 CALL DEVEND
100 FORMAT(10)
101 FORMAT(1514)
200 FORMAT(1H1,10X,'CHANNEL ',I3,' PLUS FILTER ',I3)
201 FORMAT(1H0,10X,'F4.1,1 MSECS',F6.1,1'DB')
202 FORMAT(1H0,10X,'14.1 SAMPLES AT ',F7.1,1' BAUD')
203 FORMAT(1H0,10X,'INPHASE QUADRATURE PERCENTAGE OF PEAK')
204 FORMAT(1H,10X,F7.4,5X,F7.4,5X,F7.2)
205 FORMAT(1H0,10X,'TIMING PHASE =',F4.2,1' SYMBOL PERIOD')
206 FORMAT(1H0,10X,'PHASE ROTATION =',F6.1,1' DEGREES SCALE =',F7.4)
207 FORMAT(1H0,110,F10.1,110,F10.2,F10.2,F10.4)
208 FORMAT(10F7.4)
209 FORMAT(1H,1514)
STOP
END

```

PROGRAM 4

```

SEND TO (ED,SEMICOMPUSER,AXXX)
DUMP ON (ED,PROGRAM USER)
WORK (ED,WORKFILEUSER)
RUN

```

```

LIBRARY(ED,SUBGROUPNAGE)
PROGRAM(ROOT)
COMPRESS INTEGER AND LOGICAL
INPUT 1=CR0
OUTPUT 2=LR0
OUTPUT 7=CP0
TRACE 2
END

```

```

MASTER ROOTS OF CHANNEL IMPULSE RESPONSES
COMPLEX XVECT,XVAL,XSUM,XROOT
DIMENSION XVECT(70),AR(70),AC(70),RIMP(70),QIMP(70)
DIMENSION ARR(70),ACC(70),VLN(70)
READ(1,100)NCH
100 FORMAT(110)
1 READ(1,101)ICHAN,IFILT,DR,NPTS
101 FORMAT(210,1F0,0.110)
N=NPTS
READ(1,102)(AR(I),I=1,N)
READ(1,102)(AC(I),I=1,N)
102 FORMAT(10F7.16)
TOL=X02AAF(0)
IFAIL=0
TENG=0.0
DO 2 I=1,N
2 TENG=AR(I)**2+AC(I)**2+TENG
CALL C02ADF(AR,AC,N,ARR,ACC,TOL,IFAIL)
DO 5 I=1,NPTS
DIS=ARR(I)**2+ACC(I)**2
IF(1-DIS)3,4,4
3 AR(I)=ARR(I)/DIS
AC(I)=ACC(I)/DIS
GO TO 5
4 AR(I)=ARR(I)
AC(I)=ACC(I)
5 VLN(I)=SQRT(DIS)
XVECT(1)=CMPLX(1.0,0.0)
DO 6 I=2,NPTS

```

nrp = no of roots

roots in z plane

if roots in z plane > 1

```

6 XVECT(I)=CMPLX(0.0,0.0)
DO 8 J=1,NPTS
XSUM=CMPLX(0.0,0.0)
XROOT=CMPLX(-AR(J),-AC(J))
IM1
7 XVAL=XVECT(I)
XVECT(I)=XVECT(I)+XSUM
IM1+1
XSUM=XVAL*XROOT
IF(I=NPTS)7,7,8
8 CONTINUE
FNEG=0.0
DO 9 IM1=NPTS
RIMP(I)=REAL(XVECT(I))
OIMP(I)=AIMAG(XVECT(I))
9 FNEG=FNEG+RIMP(I)**2+OIMP(I)**2
C WRITE OUT RESULTS
WRITE(2,200) ICHAN,DR
200 FORMAT(1H,10X,'LINEAR EQUALIZATION OF CHANNEL',14,
11 'SAMPLED AT',1F6.0,' RAUD')
WRITE(2,201)
201 FORMAT(1H,12X,'ROOTS IN Z PLANE DIST. EQUALIZED ROOTS',2X7
11 'IMPULSE RESPONSE I/P ENG. O/P ENG')
WRITE(2,202)
202 FORMAT(1H,12X,'R AXIS I AXIS 1,10X,
11R AXIS I AXIS REAL IMAG')
DO 12 I=1,NPTS
12 WRITE(2,203)ARR(I),ACC(I),VLN(I),AR(I),AC(I),RIMP(I),OIMP(I),
11ENG,FNEG
203 FORMAT(1H,11X,1F7.3,2X,1F7.3,2X,1F5.2,3X,1F7.3,2X,1F7.3,3X,
11F7.4,2X,1F7.4,3X,1F9.4,2X,1F9.4)
C PUNCH RESULTS ONTO CARDS
WRITE(7,210) ICHAN,IFILT,DR,NPTS
210 FORMAT(6X,14,4X,114,4X,1F6.1,4X,113,6X,9HEQUALIZED)
WRITE(7,211)(RIMP(I),I=1,NPTS)
WRITE(7,211)(OIMP(I),I=1,NPTS)
211 FORMAT(10F7.4)
C END OF DATA CARD SECTION
NCH=NCH+1
IF(NCH)14,14,1
14 STOP
END

```

PROGRAM 5a)

```

PROGRAM MAIN(INPUT,OUTPUT,TAPE1=INPUT,TAPE2=OUTPUT)
  REAL MX,CR(50,64),CQ(50,64),C(35),ICMIN,IRS,IQS,IRSM,IQSM,IREAL,
1  IQUAD,KST,IRK1,IRK2,IRK3,IRK4,IQK1,IQK2,IQK3,IQK4,D(520),X(4)
2  ,Y(4),RXR(17),RXI(17),RNR(17),RNI(17),RGR,RGI,RSM,QSM,SAR,SAQ
  INTEGER R(64),IH(64),IDV(50),IDEV(50),SV(35,35),SVE(35,35),
1  IR1,IR2,IR3,IR4,IQ1,IQ2,IQ3,IQ4,IEX(520),IPT(8,8),ISNRC,IA(64),
2  IB(64),IC(64),ID(64),IE(64),IG(64),ISNR
  INTEGER KCODE(4,4),P,Q,Z,KT(6),KSY(8,8),KDECODE(4,4),H(3),KB(6),
1  NDD,LD,NDSC,LISC,H,IN8,MARR(50),MAR(50),IV(35)
  DIMENSION NDT(10),SNR(10),RSAM(50),QSAM(50)
  DATA IPT/54,53,55,56,44,36,40,48,50,49,51,52,42,34,38,46,
1  58,57,59,60,41,33,37,45,62,61,63,64,43,35,39,47,
2  18,26,30,22,1,2,4,3,20,28,32,24,5,6,8,7,
3  19,27,31,23,13,14,16,15,17,25,29,21,9,10,12,11/
  DATA R/1,3,7,5,1,3,7,5,1,3,7,5,1,3,7,5,-7,-7,-7,-7,-1,-1,-1,-1,
1  -5,-5,-5,-5,-3,-3,-3,-3,3,3,3,3,5,5,5,5,1,1,1,1,7,7,7,7,
2  -5,-7,-3,-1,-5,-7,-3,-1,-5,-7,-3,-1,-5,-7,-3,-1/
  DATA IH/1,1,1,1,3,3,3,3,7,7,7,7,5,5,5,5,7,1,5,3,7,1,5,3,
1  17,1,5,3,-3,-5,-1,-7,-3,-5,-1,-7,-3,-5,-1,-7,-3,-5,-1,-7,-5,-5,-5,
2  -5,-7,-7,-7,-7,-3,-3,-3,-3,-1,-1,-1,-1/
  DATA KCODE/1,2,3,4,2,3,4,1,3,4,1,2,4,1,2,3/
  DATA KDECODE/1,2,3,4,4,1,2,3,3,4,1,2,2,3,4,1/
  DATA KSY/1,22,2,30,4,26,3,18,43,64,35,63,39,61,47,62,
1  15,24,6,32,8,28,7,20,41,60,33,59,37,57,45,58,
2  313,23,14,31,16,27,15,19,42,52,34,51,38,49,46,50,
3  49,21,10,29,12,25,11,17,44,56,36,55,40,53,48,54/
  DATA IA/0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,
1  10,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,
2  20,1,0,1/
  DATA IB/0,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,
1  10,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,
2  2,0,0,1,1/
  DATA IC/0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,
1  11,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,
2  21,1,1,1/
  DATA ID/0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,
1  11,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,
2  21,1,1,1/
  DATA IE/0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,
1  11,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
2  21,1,1,1/
  DATA IG/0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
1  10,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
2  21,1,1,1/
  DATA IRNG/1/,NBK/500/,NVECT/8/,NCOMP/8/,IEXN/4/,ISNR/10/
  DATA NDT/500,50000,500,50000,500,50000,500,50000,500,50000/
  DATA SNR/80,0,24,0,80,0,23,0,80,0,22,0,80,0,21,0,80,0,20,0/
  DATA INFC/17/
  DATA RXR/-0.2839,-0.4235,-1.2537,31.8210,45.6184,-2.7041,-8.4167,
1  4.2957,-0.3406,-0.4289,0.3150,-0.1859,-0.0037,0.0063,0.0217,
2  -0.0092,-0.0541/
  DATA RXI/-0.4682,0.4754,-0.1114,-0.2488,-0.9704,1.7794,-0.1499,
1  -1.1946,1.0291,-0.5497,0.1146,0.0659,-0.1124,0.0656,0.0206,
2  -0.0065,0.0055/
  READ(1,300) ICHAN,DTR,ND5
  READ(1,301) (RSAM(I),I=1,ND5)
  READ(1,301) (QSAM(I),I=1,ND5)

```

```

      MX=1000000000.0
      SDTR=SQRT(DTR)
      DO 8 J=1,NFC
      RXR(I)=RXR(I)/SDTR
      RXI(I)=RXI(I)/SDTR
      RNR(I)=0.0
      8 RNI(I)=0.0
      DO 10 J=1,NDS
      DO 9 K=1,64
      CR(J,K)=(RSAH(J)*R(K)-QSAM(J)*IM(K))
      9 CQ(J,K)=(QSAH(J)*R(K)+RSAM(J)*IM(K))
      10 IDV(J)=IDEV(J)=MARR(J)=1
C      THIS SETS-UP TABLE FOR REAL AND QUAD. MULTIPLICATIONS OF S,I,R.
C      AND THE DATA SYMBOLS
      WRITE(2,302)
      WRITE(2,303) ICHAN,NDS,DTR
      WRITE(2,304) (RSAH(J),J=1,NDS)
      WRITE(2,304) (QSAM(J),J=1,NDS)
      WRITE(2,305) IRNG,NVECT,NCOMP,IEXN
      K1=IEXN
      K2=IEXN*NVECT
      K3=NCOMP+1
      K4=NCOMP+2
      NCOMPS=NCOMP+5
      CALL G05CAF(IRNG)
C      SET-UP VECTORS AND COSTS
      DO 12 K=1,NVECT
      DO 11 J=1,NCOMPS
      11 SV(J,K)=1
      12 C(K)=10000000.0
      C(2)=100000.0
      ISNRC=0
      ICHIN=IRS=IQS=C(1)=0.0
      KTN=KRN=1
      19 ISNRC=ISNRC+1
      IF (ISNRC.GT.ISNR) GO TO 100
      SIGMA=SQRT(7.07/(10.0*(SNR(ISNRC)/10.0)))
      NBT=H8K
      NERBOR=NERA=NERB=NERC=NERD=NERE=NERG=NSYH=0
      WRITE(2,308) ISNRC,SNR(ISNRC)
      WRITE(2,311)
      20 H=1+IFIX(63.99999999*G05CAF(00))
      DO 22 I=1,NDS
      22 HAR(I)=HARR(I)
      DO 23 I=1,NDS
      ISH=I+1
      23 HARR(ISH)=HARR(I)
      HARR(1)=H
      ITEL=HARR(K3)
C      DIFFERENTIALLY CODE DATA SYNROL
      I=3
      LTSC=RTN
      P=IB(M)
      Q=IA(M)
      IF (P.EQ.0.AND.Q.EQ.0) NDSC=1
      IF (P.EQ.0.AND.Q.EQ.1) NDSC=2
      IF (P.EQ.1.AND.Q.EQ.1) NDSC=3

```

```

IF(P.EQ.1.AND.Q.EQ.0) NDSC=4
H(I)=KCODE(NDSC,LTSC)
LTSC=H(I)
I=2
P=ID(M)
Q=IC(M)
IF(P.EQ.0.AND.Q.EQ.0) NDSC=1
IF(P.EQ.0.AND.Q.EQ.1) NDSC=2
IF(P.EQ.1.AND.Q.EQ.1) NDSC=3
IF(P.EQ.1.AND.Q.EQ.0) NDSC=4
H(I)=KCODE(NDSC,LTSC)
LTSC=H(I)
I=1
P=IG(M)
Q=IF(M)
IF(P.EQ.0.AND.Q.EQ.0) NDSC=1
IF(P.EQ.0.AND.Q.EQ.1) NDSC=2
IF(P.EQ.1.AND.Q.EQ.1) NDSC=3
IF(P.EQ.1.AND.Q.EQ.0) NDSC=4
H(I)=KCODE(NDSC,LTSC)
KTN=H(I)
C DIFFERENTIAL SYM IS H1,H2,H3. KTN SET-UP FOR NEXT SYM
C CONVERT FROM H1 TO BITS
DO 70 I=1,3
IF(H(I).NE.1) GOTO71
KT(2*I-1)=0
KT(2*I)=0
GOTO 70
71 IF(H(I).NE.2) GOTO72
KT(2*I-1)=0
KT(2*I)=1
GOTO 70
72 IF(H(I).NE.3) GOTO73
KT(2*I-1)=1
KT(2*I)=1
GOTO 70
73 KT(2*I-1)=1
KT(2*I)=0
70 CONTINUE
C CONVERT TO SYM NO
NS=0
DO 74 I=1,6
74 NS=NS+KT(I)*(2**((6-I)))
NS=NS+1
C NS=SYM NO OF DIFF CODED SYM
IRSH=IQSH=0,0
C CHANNEL/SYMBOL CONVOLUTIONS
DO 21 J=1,NDS
IRSH=IRSH+CR(J,NS)
IQSH=IQSH+CQ(J,NS)
IDATA=NS
NS=IDV(J)
21 IDV(J)=IDATA
C CORRELATE THE NOISE
RGR=G05DDF(0,0,SIGMA)
RGI=G05DDF(0,0,SIGMA)
RSH=0,0

```

```

QSM=0.0
DO 99 J=1,INFC
RSH=RSH+RXH(J)*RGR
QSM=QSM+RXI(J)*RGI
SAR=RNR(J)
SAQ=RNI(J)
RNR(J)=RGR
RNI(J)=RGI
RGR=SAR
99 RGI=SAQ
C RECEIVED SYMBOLS ARE, WITH ECHO CANCELLING.
IREAL=IRSM+RSM-IRS
IQAD=IQSM+QSM-IQS
C *****DETECTOR STAGE*****
N=1
DO 50 K=1,NVECT
IRS=-IREAL
IQS=-IQAD
DO 24 J=2,K3
H=SVE(J,K)=SV(J,K)
IRS=IRS+CR(J,M)
24 IQS=IQS+CO(J,M)
C *****PUT IN REQ. EXP. CARDS FOR 4,9,OR16 EXPANSIONS.*****
C CARDS FOR TWO-WAY EXPANSION
IF(IRS.GT.-5)GOTO25
IR1=7
IR2=8
IRK1=(IRS+5)**2
IRK2=(IRS+7)**2
GO TO 31
25 IF(IRS.GT.-3)GOTO26
IR1=7
IR2=6
IRK1=(IRS+5)**2
IRK2=(IRS+3)**2
GO TO 31
26 IF(IRS.GT.-1)GOTO27
IR1=6
IR2=5
IRK1=(IRS+3)**2
IRK2=(IRS+1)**2
GO TO 31
27 IF(IRS.GT.1)GOTO28
IR1=5
IR2=4
IRK1=(IRS+1)**2
IRK2=(IRS-1)**2
GO TO 31
28 IF(IRS.GT.3)GOTO29
IR1=4
IR2=3
IRK1=(IRS-1)**2
IRK2=(IRS-3)**2
GO TO 31
29 IF(IRS.GT.5)GOTO30
IR1=3
IR2=2

```

```

      IRK1=(IRS-3)**2
      IRK2=(IRS-5)**2
      GO TO 31
30  IR1=2
      IR2=1
      IRK1=(IRS-5)**2
      IRK2=(IRS-7)**2
31  IF(IQS.GT.-5)GOTO32
      IQ1=7
      IQ2=8
      IQK1=(IQS+5)**2
      IQK2=(IQS+7)**2
      GO TO 38
32  IF(IQS.GT.-3)GOTO33
      IQ1=7
      IQ2=6
      IQK1=(IQS+5)**2
      IQK2=(IQS+3)**2
      GO TO 38
33  IF(IQS.GT.-1)GOTO34
      IQ1=6
      IQ2=5
      IQK1=(IQS+3)**2
      IQK2=(IQS+1)**2
      GO TO 38
34  IF(IQS.GT.1)GOTO35
      IQ1=5
      IQ2=4
      IQK1=(IQS+1)**2
      IQK2=(IQS-1)**2
      GO TO 38
35  IF(IQS.GT.3)GOTO36
      IQ1=4
      IQ2=3
      IQK1=(IQS-1)**2
      IQK2=(IQS-3)**2
      GO TO 38
36  IF(IQS.GT.5)GOTO37
      IQ1=3
      IQ2=2
      IQK1=(IQS-3)**2
      IQK2=(IQS-5)**2
      GO TO 38
37  IQ1=2
      IQ2=1
      IQK1=(IQS-5)**2
      IQK2=(IQS-7)**2
38  D(N)=C(K)+IRK1+IQK1
      TEX(N)=IPT(IR1,IQ1)
      N=N+1
      D(N)=C(K)+IRK1+IQK2
      TEX(N)=IPT(IR1,IQ2)
      N=N+1
      D(N)=C(K)+IRK2+IQK1
      TEX(N)=IPT(IR2,IQ1)
      N=N+1
      D(N)=C(K)+IRK2+IQK2

```

```

      IEX(H)=IPT(IP2,I02)
50 N=N+1
C   FIND LOWEST COST
      ICHIN=MX
      DO 52 N=1,K2
      IF(ICHIN.LE.D(N))GO TO 52
      ICHIN=D(N)
      ISV=N
      IV(1)=N
52 CONTINUE
C   FIND LOWEST COST OF FIRST VECTOR EXPANSIONS
      ICH=IX
      DO 61 N=1,IEXN
      IF(ICH.LE.D(N))GOTO61
      ICH=D(N)
      ILF=N
61 CONTINUE
C   SET LOWEST COST TO HI. VAL.
      D(ISV)=MX
C   FIND VECTOR NO. FROM LOWEST COST N NO.
      IVAD=1+((ISV-1)/K1)
      IVEX=IEX(ISV)
      IDEL=SVE(K3,IVAD)
C   SELECT VECTORS WITH SAME LAST COMPS.
      DO 55 K=1,NVECT
      IF(SVE(K3,K).EQ.IDEL)GO TO 55
      N1=1+(K-1)*K1
      N2=K*K1
      DO 54 N=N1,N2
54 D(N)=MX
55 CONTINUE
C   SHIFT ALONG FIRST SELECTED VECTOR
      DO 56 J=2,K3
      SV(J,1)=IVEX
56 IVEX=SVE(J,IVAD)
C   WORK OUT REMAINING STORED VECTORS
      DO 60 KK=2,NVECT
      KST=IX
      DO 58 N=1,K2
      IF(KST.LE.D(N))GO TO 58
      KST=D(N)
      ISV=N
58 CONTINUE
      IV(KK)=ISV
      IVAD=1+((ISV-1)/K1)
      IVEX=IEX(ISV)
      C(KK)=KST-ICHIN
      DO 59 J=2,K3
      SV(J,KK)=IVEX
59 IVEX=SVE(J,IVAD)
60 CONTINUE
      DO 75 KK=1,NVECT
      IF(ILF.EQ.IV(KK))GOTO 63
75 CONTINUE
C   IF NOT, CHANGE THE LAST SELECTED VECTOR
      IVAD=1+((ILF-1)/K1)

```

```

IVEX=IEX(ILF)
C(NVECT)=ICH-ICMIN
DO 62 J=2,K3
SV(J,NVECT)=IVEX
62 IVEX=SVE(J,IVAD)
C *****END OF DET. NEXT ECHO CALC.

```

```

63 H=IDEL
IRS=IGS=0.0
DO 65 J=K4,NDS
IRS=IRS+CR(J,M)
IGS=IGS+CO(J,M)
HM=IDEV(J)
IDEV(J)=H
65 H=HM

```

```

C DECODE DETECTED SYMBOL

```

```

NNS=IDEL
I=3
LDD=KRN
P=IB(NNS)
Q=IA(NNS)
IF(P.EQ.0.AND.Q.EQ.0) NDD=1
IF(P.EQ.0.AND.Q.EQ.1) NDD=2
IF(P.EQ.1.AND.Q.EQ.1) NDD=3
IF(P.EQ.1.AND.Q.EQ.0) NDD=4
H(I)=KDECODE(NDD,LDD)
LDD=NDD
I=2
P=ID(NNS)
Q=IC(NNS)
IF(P.EQ.0.AND.Q.EQ.0) NDD=1
IF(P.EQ.0.AND.Q.EQ.1) NDD=2
IF(P.EQ.1.AND.Q.EQ.1) NDD=3
IF(P.EQ.1.AND.Q.EQ.0) NDD=4
H(I)=KDECODE(NDD,LDD)
LDD=NDD
I=1

```

```

P=IG(NNS)
Q=IE(NNS)
IF(P.EQ.0.AND.Q.EQ.0) NDD=1
IF(P.EQ.0.AND.Q.EQ.1) NDD=2
IF(P.EQ.1.AND.Q.EQ.1) NDD=3
IF(P.EQ.1.AND.Q.EQ.0) NDD=4
H(I)=KDECODE(NDD,LDD)
KRN=NDD

```

```

C CONVERT TO BITS

```

```

DO 80 I=1,3
IF(H(I).NE.1) GOTO81
KB(2*I-1)=0
KB(2*I)=0
GOTO 80
81 IF(H(I).NE.2) GOTO82
KB(2*I-1)=0
KB(2*I)=1
GOTO 80

```

```

82 IF(H(I).NE.3) GOTO83

```

```

KB(2*I-1)=1
KB(2*I)=1

```

```

      GOTO 80
83  KB(2*I-1)=1
      KB(2*I)=0
80  CONTINUE
C   CONVERT TO SYMBOL NOS.
      INS=0
      DO 90 I=1,6
90  INS=INS+KB(I)*(2**(6-I))
      INS=INS+1
C   INS=DECODED DATA SYH. NO.
      IDEL=INS
C   *****ERROR ANAL.*****
      NSYH=NSYH+1
      IF(ITEL.EQ.IDEL)GO TO 67
      NERROR=NERROR+1
      IF(IA(ITEL).NE.IA(IDEL))NERA=NERA+1
      IF(IB(ITEL).NE.IB(IDEL))NERB=NERB+1
      IF(IC(ITEL).NE.IC(IDEL))NERC=NERC+1
      IF(ID(ITEL).NE.ID(IDEL))NERD=NERD+1
      IF(IE(ITEL).NE.IE(IDEL))NERE=NERE+1
      IF(IG(ITEL).NE.IG(IDEL))NERG=NERG+1
67  IF(NSYM.LT.NBT)GO TO 20
      NBT=NBT+NBK
      WRITE(2,312)NSYM,NERROR,NERA,NERB,NERC,NERD,NERE,NERG,ICMIN
      IF(NSYM=NDT(CSNRC))20,19,19
C   *****FORMATS*****
300  EDHAT(I10,F10,1,I10)
301  FORMAT(10F7.4)
302  FORMAT(1H1,10X,16,'POINT Q.A.M. VITERBI DETECTOR SIMULATION')
303  FORMAT(1H0,'CHANNEL',14,10X,16,'SAMPLES AT BAUD RATE',F9.4)
304  FORMAT(1H,10X,10F7.4)
305  FORMAT(1H0,5X,'R.N.G. START =',14,10X,14,'STORED VECTORS',5X,14,
     1'COMPONENTS',10X,'NO. OF EXPANSIONS=',14)
308  FORMAT(1H0,'XXXXXXISNRC=',13,' SNR =',F7.2,' DB XXXXXX')
311  FORMAT(1H,9X,'SYMBOLS = SYM,ERR, BIT ERR, A B C D F
     1 F',10X,'ICMIN')
312  FORMAT(1H,10X,16,7X,16,11X,6I5,8X,F11.4)
100  STOP
      END

```

PROGRAM 5b)

```

PROGRAM MAIN(INPUT,OUTPUT,TAPE1=INPUT,TAPE2=OUTPUT)
DIMENSION NDT(40),SNR(40),RSAM(50),QSAM(50)
INTEGER ISNRC,ISNR,IDV(50),IDEV(50),ILDS,IR,IQ,ITX,R(64),
2 IM(64),IA(64),IB(64),IC(64),ID(64),IE(64),IG(64),IPT(8,8)
REAL CR(50,64),CQ(50,64),IRS,IQS,IQSM,IRSM,IREAL,IQUAD,X(6),Y(6)
1,IRR,IQQ
DATA IPT/59,63,55,51,35,39,47,43,60,64,56,52,36,40,48,44,
158,62,54,50,34,38,46,42,57,61,53,49,33,37,45,41,25,29,21,17,1,5,13
2,9,26,30,22,18,2,6,14,10,28,32,24,20,4,8,16,12,27,31,23,19,3,
37,15,11/
DATA R/1,1,1,1,3,3,3,3,7,7,7,7,5,5,5,5,-1,-1,-1,-1,-3,-3,-3,-3,
1-7,-7,-7,-7,-5,-5,-5,-5,1,1,1,1,3,3,3,3,7,7,7,7,5,5,5,5,
2-1,-1,-1,-1,-3,-3,-3,-3,-7,-7,-7,-7,-5,-5,-5,-5/
DATA IM/1,3,7,5,1,3,7,5,1,3,7,5,1,3,7,5,1,3,7,5,1,3,7,5,
11,3,7,5,1,3,7,5,-1,-3,-7,-5,-1,-3,-7,-5,-1,-3,-7,-5,-1,-3,-7,-5,
2-1,-3,-7,-5,-1,-3,-7,-5,-1,-3,-7,-5,-1,-3,-7,-5/
DATA IA/0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,
10,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,
20,1,0,1/
DATA IB/0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,
10,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,
2,0,0,1,1/
DATA IC/0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,
11,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,
21,1,1,1/
DATA ID/0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,
11,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,
21,1,1,1/
DATA IE/0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,
11,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,
21,1,1,1/
DATA IG/0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
10,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
21,1,1,1/
DATA IRNG/1,NBK/500,ISNR/40/
DATA NDT/500,50000,500,50000,500,50000,500,50000,500,50000,500,
1,50000,500,50000,500,50000,500,50000,500,50000,500,50000,500,
250000,500,50000,500,50000,500,50000,500,50000,500,50000,500,
350000,500,50000,500,50000/
DATA SNR/80,0,29,0,80,0,28,0,80,0,27,0,80,0,26,0,80,0,25,0,
1 80,0,24,0,80,0,23,0,80,0,22,0,80,0,21,0,80,0,20,0,80,0,19,0,
2 80,0,18,0,80,0,17,0,80,0,16,0,80,0,15,0,80,0,14,0,80,0,13,0,
3 80,0,12,0,80,0,11,0,80,0,10,0/
READ(1,300) ICHAN,DTR,NDS
READ(1,301)(RSAM(I),I=1,NDS)
READ(1,301)(QSAM(I),I=1,NDS)
C
SET-UP RE AND IMAG, MULTIPLICATION ARRAYS OF SIR AND SYMS.
DO 10 J=1,NDS
DO 9 K=1,64
CR(J,K)=(RSAM(J)*R(K)-QSAM(J)*IM(K))
9 CQ(J,K)=(QSAM(J)*R(K)+RSAM(J)*IM(K))
10 IDV(J)=IDEV(J)=1
WRITE(2,302)
WRITE(2,303) ICHAN,NDS,DTR
WRITE(2,304) (RSAM(J),J=1,NDS)
WRITE(2,304) (QSAM(J),J=1,NDS)

```

CALL G05CBF(IRNG)

ISNRC=0

IRS=IQS=0.0

ILDS=1

19 ISNRC=ISNRC+1

IF(ISNRC.GT.ISNR)GO TO 100

NBT=NRK

NERROR=NERA=NERB=NERC=NERD=NERE=NERG=NSYM=0

WRITE(2,308) ISNRC,SNR(ISNRC)

WRITE(2,311)

SIGMA=SQRT(7.0/(10.0+*(SNR(ISNRC)/10.0)))

C SELECT RANDOM SYMBOL NO. FROM 1 TO 64

20 M=1+IFIX(63.99999999*G05CAF(QQ))

IRSM=IQSM=0.0

ITX=ITEL=M

C CREATE CHANNEL CONVOLUTIONS

DO 21 J=1,NDS

IRSM=IRSM+CR(J,M)

IQSM=IQSM+CQ(J,M)

IDATA=M

M=IDV(J)

21 IDV(J)=IDATA

C ADD NOISE

IREAL=IRSM+G05DDF(0.0,SIGMA)

IQUAD=IQSM+G05DDF(0.0,SIGMA)

C *****DETECTOR SECTION*****

C WORK-OUT CANCELLATION TERMS

IRS=IQS=0.0

M=ILDS

C ILDS IS THE LAST DETECTED SYMBOL

DO 22 J=2,NDS

IRS=IRS+CR(J,M)

IQS=IQS+CQ(J,M)

MM=IDV(J)

IDV(J)=M

22 M=MM

C SUBTRACT THE ESTIMATE OF I,S,I, FROM THE RX. SAMPLES

IRR=IREAL-IRS

IQQ=IQUAD-IQS

C NEXT THRESHOLD DETECT ON IRR AND IQQ

DO 23 K=2,6,2

X(K)=K*RSAM(1)

23 Y(K)=X(K)

IF(IRR.GT.Y(6))GO TO 24

IR=1

GOTO 31

24 IF(IRR.GT.Y(4)) GO TO 25

IR=2

GOTO 31

25 IF(IRR.GT.Y(2))GO TO 26

IR=3

GO TO 31

26 IF(IRR.GT.0.0) GO TO 27

IR=4

GO TO 31

27 IF(IRR.GT.X(2))GO TO 28

IR=5

```

      GOTO 31
28 IF(IRR.GT.X(4)) GO TO 29
   IR=6
   GO TO 31
29 IF(IRR.GT.X(6)) GO TO 30
   IR=7
   GO TO 31
30 IR=8
31 IF(IQQ.GT.Y(6)) GO TO 32
   IQ=1
   GO TO 39
32 IF(IQQ.GT.Y(4)) GO TO 33
   IQ=2
   GO TO 39
33 IF(IQQ.GT.Y(2)) GO TO 34
   IQ=3
   GO TO 39
34 IF(IQQ.GT.0,0) GO TO 35
   IQ=4
   GO TO 39
35 IF(IQQ.GT.X(2)) GO TO 36
   IQ=5
   GO TO 39
36 IF(IQQ.GT.X(4)) GO TO 37
   IQ=6
   GO TO 39
37 IF(IQQ.GT.X(6)) GO TO 38
   IQ=7
   GO TO 39
38 IQ=8
39 ILDS=IPT(IR,IQ)
C *****ERROR ANAL.*****
  NSYM=NSYM+1
  IF(ITX.EQ.ILDS) GO TO 67
  NERROR=NERROR+1
  IF(IA(ITEI).NE.IA(ILDS)) NERA=NERA+1
  IF(IB(ITEI).NE.IB(ILDS)) NERB=NERB+1
  IF(IC(ITEI).NE.IC(ILDS)) NERC=NERC+1
  IF(ID(ITEI).NE.ID(ILDS)) NERD=NERD+1
  IF(IE(ITEI).NE.IE(ILDS)) NERE=NERE+1
  IF(IG(ITEI).NE.IG(ILDS)) NERG=NERG+1
67 IF(NSYM.LT.NBT) GO TO 20
  NBT=NBT+NBK
  WRITE(2,312) NSYM,NERROR,NERA,NERB,NERC,NERD,NERE,NERG
  IF(NSYM.NDT(ISNRC)) 20,19,19
C *****FORMATS*****
300 FORMAT(I10,F10,1,1,10)
301 FORMAT(10F7,4)
302 FORMAT(1H0,10X,'CONVENTIONAL NON-LINEAR DETECTOR SIMULATION')
303 FORMAT(1H0,'CHANNEL',14,10X,16,' SAMPLES AT BAUDRATE',F9,4)
304 FORMAT(1H,10X,10F7,4)
308 FORMAT(1H0,'XXXXXXXXX PHASE=',13,' SNR=',F7,4,' DB XXXXXX')
305 FORMAT(1H0,5X,'R.G. START AT',14)
311 FORMAT(1H,9X,'SYMBOLS SYMB.ERR. BIT ERR. A B C D
      1 E F')
312 FORMAT(1H,10X,16,7X,16,11X,6I5)
100 STOP

```

PROGRAM 6

```

SEND TO (ED,SEMICOMPOSEK,AXXX)
DUMP ON (ED,PROGRAM USER)
WORK (ED,WORKFILEUSER)
RUN

```

```

LIBRARY(ED,SUBGROUPNAGE)
PROGRAM(ESTIMATE)
COMPRESS INTEGER AND LOGICAL
INPUT 1=CR0
OUTPUT 2=LPO
TRACE 0
END

```

```

MASTER ESTIMATOR
REAL MSE
DIMENSION AR(50),AI(50),CR(50),CI(50),ESDR(50),ESDI(50),SI(50)
1 SR(50),XR(50),XI(50),DR(50),DI(50)
READ(1,290) A,ISNR
READ(1,300) ICHAN1,N
READ(1,301)(AR(I),I=1,N)
READ(1,301)(AI(I),I=1,N)
READ(1,300) ICHAN2,N
READ(1,301)(DR(I),I=1,N)
READ(1,301)(DI(I),I=1,N)
C*****FORMATS FOR READS*****
290 FORMAT(F0.0,I0)
300 FORMAT(2I0)
301 FORMAT(10F7.4)
STP=0.0
DO 60 J=1,N
60 STP=STP+(AR(J)**2+AI(J)**2)
NCTC=0
2 NCTC=NCTC+1
WRITE(2,400)
WRITE(2,401) A
WRITE(2,402) ICHAN1
WRITE(2,403) ICHAN2
WRITE(2,408) ISNR
WRITE(2,409)
C SET UP THE INITIAL VECTORS
DO 10 J=1,N
CR(J)=DR(J)

```

```

      CI(J)=DI(J)
10  ESDR(J),ESDI(J),SR(J),SI(J)=1.0
C    VARIANCE OF THE NOISE
      CALL G05CBF(1)
      SIGMA=SQRT(7.07/(10.0*(TSNR/10.0)))
      NCT=0
      NCR=0
      NNCT=0
      NNN=0
5   NCT=NCT+1
      NNN=NNN+1
C    RANDOM SYMBOLS
      SRR=2*IFIX(9.99999999*G05CAF(QQ))-7
      SII=2*IFIX(9.99999999*G05CAF(QQ))-7
C    SHIFT SIGNAL VECTORS
      DO 20 J=1,N
        XR(J)=SR(J)
20  XI(J)=SI(J)
      DO 21 J=1,(N-1)
        SR(J+1)=XR(J)
21  SI(J+1)=XI(J)
      RR(1)=SRR
      SI(1)=SII
C    CONVOLVE CHANNEL AND SYMBOLS
      XPR=0.0
      XPI=0.0
      DO 22 J=1,N
        XPR=XPR+SR(J)*AR(J)-SI(J)*AI(J)
22  XPI=XPI+SI(J)*AR(J)+SR(J)*AI(J)
C    ADD NOISE
      XPR=XPR+G05DDF(0.0,SIGMA)
      XPI=XPI+G05DDF(0.0,SIGMA)
C    SET UP ESTIMATOR DELAY STORES
      DO 23 J=1,N
        XR(J)=ESDR(J)
23  XI(J)=ESDI(J)
      DO 24 J=1,(N-1)
        ESDR(J+1)=XR(J)
24  ESDI(J+1)=XI(J)
      ESDR(1)=SR(1)
      ESDI(1)=SI(1)
      IF(NCT=35) 5,5,30
C    EST. O/P TO ECHO CANCELLER....
30  ESOR,ESOI=0.0
      DO 25 J=2,N
        ESOR=ESOR+CR(J)*ESDR(J)-CI(J)*ESDI(J)
25  ESOI=ESOI+CR(J)*ESDI(J)+CI(J)*ESDR(J)
      NCR=NCR+1
      NNCT=NNCT+1
      ER=XPR-ESOR=(ESDR(1)*CR(1)-CI(1)*ESDI(1))
      FI=XPI-ESOI=(ESDI(1)*CR(1)+CI(1)*ESDR(1))
      UAR=ER*A
      UAI=FI*A
C    UPDATE TAPS
      DO 31 J=1,N
        CR(J)=CR(J)+(UAR*ESDR(J)+UAI*ESDI(J))
31  CI(J)=CI(J)+(UAI*ESDR(J)-UAR*ESDI(J))
      IF(NCR=50) 5,40,40
40  NCR=0
      MSE=0.0
      DO 45 J=1,N
45  MSE=MSE+(AR(J)*CR(J))**2+(AI(J)-CI(J))**2

```

```

MSE=MSE/STP
WRITE(2,410) NNCT,MSE
IF(NNN-500)5,50,50
50 WRITE(2,406)
NNN=0
DO 500 J=1,N
WRITE(2,407) CR(J),CI(J),AR(J),AI(J)
500 CONTINUE
IF(NNCT-1000)5,550,550
550 A=A+0.00005
IF(NCTC-20)2,2,600
C  $$$$$$$$$$$$ FORMATS FOR WRITES$$$$$$$$$$$$$$$
400 FORMAT(1H, '*****'/)
401 FORMAT(1H, 'INTEGRATION CONSTANT IS...',F11.5)
402 FORMAT(1H, 'CHANNEL TO BE ESTIMATED IS...',I2//)
403 FORMAT(1H, 'EST. TAPS PRE SET TO...',I2//)
408 FORMAT(1H, 'SNR IS ...',I3//)
409 FORMAT(1H, 'NO OF SYMBOLS',I10X, 'M.S. ERROR')
410 FORMAT(1H, '4X,14,20X,F11.5)
406 FORMAT(1H, '4X, EST. TAPS',I20X, 'ACTUAL CHN. SIR'//)
407 FORMAT(1H, 'F7.4,3X,F7.4,10X,F7.4,3X,F7.4)
600 STOP
END

```

STOR-A-FILE IMAGING LTD

DOCUMENTS OF POOR ORIGINAL HARD COPY

PROGRAM 7

LIST

```

10CLS
20PRINT" THIS PROG. LOCATES THE COMPLEX ROOTS OF A COMPLEX POLYNOMIAL. THE PO
LY. IS INPUTED FROM A DISC FILE."
30 PRINT"INPUT THE FILE NAME CONTAINING THE COEFFICIENTS"
40 INPUT Q$
50 QQ=OPENUP(Q$)
60 INPUT#QQ,NN%
70 NN%=NN%-1:N%=NN%
80 DIM AR(NN%+1),AC(NN%+1),REZ(NN%),IMZ(NN%)
90 FOR I%=NN% TO 0 STEP -1
100 INPUT#QQ,AR(I%),AC(I%)
110 NEXT
120 CLOSE#QQ
130 FOR I%=0 TO NN%
140 AR(I%)=AR(I%):AC(I%)=AC(I%):NEXT
150 PRINT"INPUT THE COEFF. POLY IS AS:- (AR(NN%)+jAC(NN%)Z^NN%+...AR(0)+jAC(0
). ENTER AR(0),AC(0) FIRST."
160 IND=0
170 PROCCOMPLEXROOTS
180 VDU2
181 CLS
182 PRINT"ROOTFINDING PROGRAMME"
183PRINT
184 PRINT"ROOTS OF POLYNOMIAL FROM FILE ";Q$
190PRINT"ERROR INDICATOR=",IND
200 @%=&02060D
210 FOR I%=1 TO N%
220 PRINT REZ(I%),IMZ(I%)
230 NEXT
240 VDU3
250 @%=0
260 PRINT"SAVE THESE ROOTS??"
270 INPUT SS$
280 IF SS$="N" GOTO 350
290 PRINT"INPUT THE FILE NAME"
300 INPUT FM$
305 Y=OPENOUT(FM$)
310 PRINT#Y,NN%
320 FOR I%= 1 TO NN%
330 PRINT#Y,REZ(I%),IMZ(I%)
340 NEXT
345 CLOSE#Y
350 END
360 DEF PROCCOMPLEXROOTS
370 DIM ZPR(NN%),ZPC(NN%),BR(NN%+1),BC(NN%+1)
380 LOCAL ZPR,ZPC,L%,I%,K%,X,Y,ZZPR,ZZPC,ZDR,ZDC,ZZDR,ZZDC,DEM,CVR,CVC,X1,Y1,Z
%,BR,BC,J%,DEN
390 FP%=-31:TLL=2^FP%
400 TLL2=SQR(TLL)
410 FOR K%=NN% TO 2 STEP -1
420 L%=1:X=0.0001:Y=0.001:Z%=1
430 PROCFUNC
440 IF (ABS(ZDR)>TLL AND ABS(ZDC)>TLL) OR (ABS(ZDDR)<TLL AND ABS(ZDDC)<TLL) GO
TO 480
450 L%=L%+1
460 IF L%>100 GOTO 590
470 X=X+0.001:Y=Y+0.001:GOTO 430
480 PROCNROOT
490 IF Z%>200 GOTO 580

```

```

500 IF ABS(X1-X)>TLL AND ABS(Y1-Y)>TLL THEN X=X1:Y=Y1:GOTO430
510 REZ(K%)=X1:IMZ(K%)=Y1
520 PROCLDIV
530 NEXT K%
540 DEN=(AR(1)^2+AC(1)^2)
550 REZ(1)=- (AR(0)*AR(1)+AC(0)*AC(1))/DEN
560 IMZ(1)=- (-AR(0)*AC(1)+AC(0)*AR(1))/DEN
570 GOTO 600
580 IND=1:GOTO 600
590 IND=2
600 ENDPROC
610 DEF PROCFUNC
620 ZPR(1)=X:ZPR(0)=1.0
630 ZPC(1)=Y:ZPC(0)=1.0
640 FOR I%=2 TO K%
650 ZPR(I%)=ZPR(I%-1)*X-ZPC(I%-1)*Y
660 ZPC(I%)=ZPR(I%-1)*Y+ZPC(I%-1)*X
670 NEXT I%
680 ZZPR=AR(0):ZZPC=AC(0)
690 FOR I%=1 TO K%
700 ZZPR=ZZPR+(ZPR(I%)*AR(I%)-ZPC(I%)*AC(I%))
710 ZZPC=ZZPC+(ZPR(I%)*AC(I%)+ZPC(I%)*AR(I%))
720 NEXT I%
730 REM ZZPR+jZZPC IS f(Z) AT Z=X+jY
740 ZDR=AR(1):ZDC=AC(1)
750 FOR I%=2 TO K%
760 ZDR=ZDR+I%*(ZPR(I%-1)*AR(I%)-ZPC(I%-1)*AC(I%))
770 ZDC=ZDC+I%*(ZPR(I%-1)*AC(I%)+ZPC(I%-1)*AR(I%))
780 NEXT I%
790 REM ZDR+jZDC IS f'(Z) AT X+jY
800 ZDDR=2*AR(2):ZDDC=2*AC(2)
810 IF K%=2 GOTO 860
820 FOR I%=3 TO K%
830 ZDDR=ZDDR+I%*(I%-1)*(ZPR(I%-2)*AR(I%)-ZPC(I%-2)*AC(I%))
840 ZDDC=ZDDC+I%*(I%-1)*(ZPR(I%-2)*AC(I%)+ZPC(I%-2)*AR(I%))
850 NEXT I%
860 REM ZDDR+jZDDC IS f''(Z) AT X+jY
870 ENDPROC
880 DEF PROCNROOT
890 Z%=Z%+1
900 DEM=ZDR^2+ZDC^2
910 CVR=(ZZPR*ZDR+ZZPC*ZDC)/DEM
920 CVC=(ZDR*ZZPC-ZDC*ZZPR)/DEM
930 X1=X-CVR:Y1=Y-CVC
940 ENDPROC
950 DEF PROCLDIV
960 J%=K%-1
970 BR(J%)=AR(J%+1):BC(J%)=AC(J%+1)
980 FOR I%=(J%-1) TO 0 STEP -1
990 BR(I%)=AR(I%+1)+(BR(I%+1)*X1-BC(I%+1)*Y1)
1000 BC(I%)=AC(I%+1)+(BR(I%+1)*Y1+BC(I%+1)*X1)
1010 NEXT I%
1020 FOR I%=0 TO J%
1030 AR(I%)=BR(I%)
1040 AC(I%)=BC(I%)
1050 NEXT I%
1060 ENDPROC

```

580
PROGRAM 8

```

LIST
 10 CLS
 20 PRINT"PPREDICTING CARRIER-TRACKING LOOP"
 25 REM ALL REMS REMOVED, SEE "CARRIER" FOR PROG. DETAILS
 30 PRINT
 40 PRINT
 50 DIM ER(50),EQ(50),SR(50),SQ(50),XR(50),XQ(50),RSR(50),RSQ(50),ZR(50),ZQ(50
),RSRN(50),RSON(50)
 60 INPUT"INPUT INTEGRATION CONSTANTS, ALPHA AND BETA",ALPHA,BETA
 61 INPUT"INPUT FADING PARA.",DAB
 62 INPUT"INPUT FINAL ALPHA VAL.",FAL
 70 INPUT"INPUT THE SNR",SNR
 80 INPUT"INPUT GAIN FACTOR IF EQUALISED SIR IS TO BE USED, IN dB",GF
 90 SNR=SNR-GF
100 INPUT"INPUT THE BAUD RATE",B
110 INPUT"INSERT DISC CONTAINING THE REQUIRED CHANNEL SIR,PRESS Y TO CONT
INUE",ZZ$
120 IF ZZ$<>"Y" GOTO 110
130 PRINT"WHAT IS THE SIR FILENAME?"
140 INPUT X1$
150 X=OPENUP(X1$)
160 INPUT I,X,NP%
170 FOR I%=1 TO NP%
171 SR(I%)=0.0:SQ(I%)=0.0
172 XR(I%)=0.0:XQ(I%)=0.0
173 RSR(I%)=0.0:RSQ(I%)=0.0
174 ZR(I%)=0.0:ZQ(I%)=0.0
175 RSRN(I%)=0.0:RSON(I%)=0.0
180 INPUT I,X,ER(I%),EQ(I%):NEXT
190 CLOSE I,X
200 INPUT" NUMBER OF SAMPLES TO TEST OVER",NS%
205 INPUT"INPUT NO. OF SAMPLES PER PLOT",PN%
210 INPUT"INPUT DETECTION DELAY",N%
220 INPUT"INPUT INITIAL PHASE ERROR IN DEGREES",PED
225 INPUT"INPUT PHASE SCALE ON PLOT,IN DEGREES",PMA%
230 INPUT"INPUT FREQ. OFFSET IN HERTZ ",FOD
233 INPUT"INPUT JITTER AMPLITUDE IN DEGREES, AT 50, 150, 250 HERTZ",JA1,JA2,JA
3
235 PER=PED*PI/180:FSS=2*PI*FOD/B
236 PJ1=2*PI*50/B:PJ2=PJ1*3:PJ3=PJ1*5
237 AJ1=JA1*PI/180:AJ2=JA2*PI/180
238 AJ3=JA3*PI/180
250 SIGMA=SQR(7.0/(10.0^(SNR/10)))
260 NCR%=0:NNCT%=0
270 TS=0.0:TS2=0.0:PCR=1.0:PCQ=0.0
271 KK%=0
272 PCIR=1.0:PCIQ=0.0
280 CLS
290 @%=802050C
300 VDU2
310 PRINT"PPREDICTING CARRIER PHASE TRACKING LOOP"
320 PRINT"BAUD RATE IS",B
330 PRINT"CHANNEL USED IS",X1$
340 PRINT"ALPHA=",ALPHA
350 PRINT"BETA=",BETA
351 PRINT"ALPHA,BETA FADING VA. ";DAB
352 PRINT"FINAL ALPHA VA. ";FAL
360 PRINT"SNR IS ";SNR;" dB"
370 PRINT"DETECTION DELAY IS",N%
380 PRINT"INITIAL PHASE ERROR IS ";PED;"DEGREES"
390 PRINT"FREQ. OFFSET IS ";FOD;"HERTZ"
391 PRINT"PHASE JITTER AT 50,150 250 HERTZ IS ";JA1;" ";JA2;" ";JA3;" DEGREES
PEAK"
400 PRINT"*****"
401 PRINT"*****"
402 PRINT

```

```

403 PRINT
405 @Z=8020003
410 VDU3
415 MODE 4
420 MOVE0,0:DRAW0,1000
430 MOVE0,500:DRAW1279,500
440 FOR IZ=1TO10
450 SS=IZ*127.9
460 MOVESS,500:DRAWSS,496:NEXT
461 MOVE 0,0
462 FOR IZ=0TO20
463 SS=IZ*1000/20
464 MOVE 0,SS:DRAWSS,SS:NEXT
465 VDU5:MOVE0,32:PRINT-PMAX
466 MOVE0,1000:PRINTPMAX
467 MOVE(1279-INT(LOG(NS%)+1)*32),488
468 PRINT NS%:VDU4
469 MOVE 0,(500+500*PED/PMAX)
470 REM SET-UP RANDOM SYMBOLS
480 SRR=2*INT(7.99999999*RND(1))-7
490 SQQ=2*INT(7.99999999*RND(1))-7
491 @Z=802050C
510 FOR IZ=1 TO NP%
520 XR(IZ)=SR(IZ):XQ(IZ)=SQ(IZ):NEXT
530 FOR IZ=1TO NP%-1
540 SR(IZ+1)=XR(IZ):SQ(IZ+1)=XQ(IZ)
550 NEXT
560 SR(1)=SRR:SQ(1)=SQQ
580 RR=0.0:RQ=0.0
590 FOR IZ=1 TO NP%
600 RR=RR+SR(IZ)*ER(IZ)-SQ(IZ)*EQ(IZ)
610 RQ=RQ+SQ(IZ)*ER(IZ)+SR(IZ)*EQ(IZ)
620 NEXT
630 ESR=RR:ESQ=RQ
640 REM ADD NOISE
650 NR=(RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+
RND(1)+RND(1)-6)*SIGMA
660 RRN=RR+NR
670 NQ=(RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+
RND(1)+RND(1)-6)*SIGMA
680 RQN=RQ+NQ
720 FS=PER+FSS*KK%+AJ1*SIN(PJ1*KK%)+AJ2*SIN(PJ2*KK%)+AJ3*SIN(PJ3*KK%)
730 RR=COS(FS)*RRN-SIN(FS)*RQN
735 RQ=COS(FS)*RQN+SIN(FS)*RRN
880 FOR IZ=1 TO NP%
890 XR(IZ)=RSR(IZ):XQ(IZ)=RSQ(IZ)
900 ZR(IZ)=RSRN(IZ):ZQ(IZ)=RSQN(IZ)
910 NEXT
920 FOR IZ=1 TO (NP%-1)
930 RSR(IZ+1)=XR(IZ):RSQ(IZ+1)=XQ(IZ)
940 RSRN(IZ+1)=ZR(IZ):RSQN(IZ+1)=ZQ(IZ)
950 NEXT
960 RSR(1)=ESR:RSQ(1)=ESQ
965 REM RSR,RSQ IS UNCORRUPTED SAMPLE VECTOR
967 REM RSRN,RSQN IS CORRUPTED SAMPLE VECTOR
970 RSRN(1)=RR:RSQN(1)=RQ
980 KK%=KK%+1
981 IF KK%<NP% GOTO 470
990 NNCT%=NNCT%+1
1000 NCR%=NCR%+1
1010 REM THIS IS PREDICTIVE LOOP
1020 JZ=NZ+1
1030 MR=RSRN(JZ)*RSR(JZ)+RSQN(JZ)*RSQ(JZ)
1032 MQ=RSQN(JZ)*RSR(JZ)-RSRN(JZ)*RSQ(JZ)
1034 MRM=PCIR*MR+PCIQ*MQ
1036 MQM=PCIR*MQ-PCIQ*MR

```

```

1038 REM NORM. IMAG BY MOD. OF PRODUCT
1039 MAG=SQR(MR^2+MQ^2)
1040 MOM=MOM/MAG
1041 IF KK%<500 THEN GOTO 1050
1042 ALPHA=ALPHA-DAB*(ALPHA-FAL)
1043 BETA=ALPHA*2
1050 A=BETA*MOM
1060 TS=TS+A*ALPHA
1070 B=TS+A
1080 TS2=TS2+B
1090 TS3=TS*(N%)+TS2
1100 PCIR=COS(TS2):PCIQ=SIN(TS2)
1110 PCR=COS(TS3):PCQ=SIN(TS3)
1120 IF NCR%<PN% GOTO 470
1130 NCR%=0
1131 CRR=RSRN(1)*PCR+RSQN(1)*PCQ
1132 CRQ=RSQN(1)*PCR-RSRN(1)*PCQ
1133 CPT=DEG(ATN(ABS(CRQ/CRR)))
1134 CPR=DEG(ATN(ABS(RSQ(1)/RSR(1))))
1135 IF CRR>0 AND CRQ<0 THEN CPT=360-CPT
1136 IF CRR<0 AND CRQ<0 THEN CPT=180+CPT
1137 IF CRR<0 AND CRQ>0 THEN CPT=180-CPT
1138 IF RSR(1)>0 AND RSQ(1)<0 THEN CPR=360-CPR
1139 IF RSR(1)<0 AND RSQ(1)<0 THEN CPR=180+CPR
1140 IF RSR(1)<0 AND RSQ(1)>0 THEN CPR=180-CPR
1145 PE=ABS(CPT-CPR)
1150 IF PE>180 THEN PE=PE-360
1160 IF CPT<CPR THEN PE=-PE
1225 XX=XX+1279*PN%/NS%
1229 DRAW XX,500+500*PE/PMAX
1230 IF NNCT%<NS% GOTO 470
1240 VDU3
1250 @%=80
1260 SOUND 1,-15,100,40
1265 PROCSDUMP(1)
1266 SOUND 1,-15,100,40
1267 PRINT"PHASE ERROR AFTER";NNCT%;"SYMBOLS IS";PE;"DEGREES"
1270 END
10000 REM*****
10010 REM SCREEN DUMP TO PRINTER PROC *
10020 REM F% IS FORGROUND COLOUR (1) *
10030 REM*****
10040 DEF PROCSDUMP(F%)
10050 VDU 2,1,27,1,51,1,24
10060 FOR Y%=1023 TO 0 STEP -32
10070 VDU 1,27,1,75,1,64,1,1
10080 FOR X%=1 TO 1279 STEP 4
10090 S%=1:P%=0
10100 FOR Z%=Y%-32 TO Y% STEP 4
10110 IF POINT(X%,Z%)=F% THEN P%=P%+S%
10120 S%=S%+S%:NEXT
10130 VDU 1:PRINT CHR$(P%);:NEXT
10140 VDU 1,13:NEXT
10150 VDU 1,10,1,13,1,27,1,50
10160 VDU 3:ENDPROC

```

```

>LIST
10 CLS
15 @%=402050C
20 DIM XR(50),XQ(50),SR(50),SQ(50),ER(50),EQ(50),YR(50),YQ(50)
40 INPUT"INPUT AGC INTEG VAL ",AGCI
50 INPUT"INPUT ESTIM INTEG VAL ",ESTIMI
60 INPUT"INPUT SIG LEVEL CHANGE IN dB",DB
70 INPUT"INPUT SNR",SNR
80 INPUT"ENTER SIR FILEMANE",X1$
90 X=OPENUP(X1$)
95 TENG=0.0
100 INPUT LX,NP%
105 NP%=10
110 FOR I%=1 TO NP%
120 INPUT LX,ER(I%),EQ(I%)
130 TENG=TENG+ER(I%)^2+EQ(I%)^2
140 NEXT
150 CLOSE LX
160 TENG=SQR(TENG)
200 Y=1.0:KK%=0:Z=1.0:NCT%=0
250 SIGMA=SQR(7.0/(10.0^(SNR/10)))
255 VDU2
260 PRINT"    AGC TEST"
270 PRINT"AGC INTEG VAL= ";AGCI
280 PRINT"ESTIM INTEG VAL= ";ESTIMI
290 PRINT"SIG LEVEL CHANGE= ";DB;" dB"
300 PRINT"SNR= ";SNR
310 PRINT"CHANNEL IS ";X1$
320 PRINT"CHANNEL ENG IS ";TENG
330 VDU3
400 SRR=2*INT(7.99999999*RND(1))-7
410 SQQ=2*INT(7.99999999*RND(1))-7
420 FOR I%=1 TO NP%
430 XR(I%)=SR(I%):XQ(I%)=SQ(I%)
435 NEXT
440 FOR I%=1 TO NP%-1
450 SR(I%+1)=XR(I%):SQ(I%+1)=XQ(I%)
460 NEXT
470 SR(1)=SRR:SQ(1)=SQQ
480 RR=0.0:RQ=0.0
500 FOR I%=1 TO NP%
510 RR=RR+SR(I%)*ER(I%)-SQ(I%)*EQ(I%)
520 RQ=RQ+SQ(I%)*ER(I%)+SR(I%)*EQ(I%)
530 NEXT
600 NR=(RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+
RND(1)+RND(1)-6)*SIGMA
610 NQ=(RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+RND(1)+
RND(1)+RND(1)-6)*SIGMA
620 RRN=RR+NR
630 RQN=RQ+NQ
650 RRN=RRN*Z/Y
660 RQN=RQN*Z/Y
730 KK%=KK%+1
740 IF KK%<NP% GOTO 400
755 ESR=0.0:ESQ=0.0
760 FOR I%=1 TO NP%
765 ESR=ESR+SR(I%)*YR(I%)-SQ(I%)*YQ(I%)
770 ESQ=ESQ+SR(I%)*YQ(I%)+SQ(I%)*YR(I%)
780 NEXT
785 ERI=(RRN-ESR)*ESTIMI
790 ERQ=(RQN-ESQ)*ESTIMI
795 ENG=0.0
800 FOR I%=1 TO NP%
810 YR(I%)=YR(I%)+(SR(I%)*ERI+SQ(I%)*ERQ)
820 YQ(I%)=YQ(I%)+(SR(I%)*ERQ-SQ(I%)*ERI)
830 ENG=ENG+YR(I%)^2+YQ(I%)^2

```

```
840 NEXT
850 ENG=SQR(ENG)
900 IF KK%<500 GOTO 400
950 NCT%=NCT%+1
1000 Y=10*(DB/20)
1010 Z=Z+AGCI*(TENG-ENG)
1011 AA=AGCI
1012 AGCI=AA-(AA-0.03)*0.15
1020 IF NCT%<50 GOTO 400
1025 VDU2
1030 PRINT KK%-500-NP%,Y,Z,TENG,ENG
1035 IF ABS(Z-Y)<0.0001 GOTO 2000
1040 NCT%=0
1050 IF KK%>5000 THEN GOTO 2000 ELSE GOTO 400
2000 END
```

