

LOUGHBOROUGH  
UNIVERSITY OF TECHNOLOGY  
LIBRARY

AUTHOR/FILING TITLE

NWESSRY, M

ACCESSION/COPY NO. 057854/01

~~057684/01~~

VOL. NO.

CLASS MARK

ARCHIVES  
COPY

FOR REFERENCE ONLY

AN INVESTIGATION INTO THE SCALE CODEC SYSTEM

AND

THE DESIGN OF A-LAW PCM - TO - SCALE AND  
SCALE - TO - A-LAW PCM CONVERTERS

by

MUSTAFA M. NWESSRY, M.Sc.

DOCTORAL THESIS

OCTOBER 1976

Submitted in partial fulfilment of the requirements  
for the award of Doctor of Philosophy of  
The Loughborough University of Technology (England)

Supervisor: R. STEELE, B.Sc., Ph.D.  
Department of Electrical and  
Electronic Engineering

© by M. M. Nwessry 1976

Loughborough University of Technology Library	
Date	Mar. 77
Class	
Acc. No.	<del>057854</del> /01

057854/01

To Sayda, my wife

VOLUME 1

MAIN THESIS

## SYNOPSIS

The type of digital coding which has been internationally approved for telephone interexchange transmission is logarithmically-companded Pulse Code Modulation, known as A-law PCM. Delta and Delta-sigma modulation with digitally controlled syllabic companding are preferred in military applications, and are increasingly used in the local subscriber networks.

At the moment, the users of these systems are unable to communicate directly with each other. The prime objective of this study is to design digital converters to translate the codes from a Syllabically Companded And Logically Encoded Delta-sigma modulation system, known as SCALE, to A-law PCM codes and vice versa.

SCALE has been in service for some years, but a theoretical description of it has been lacking. This investigation begins with a detailed experimental, theoretical, and computer simulation study of SCALE, leading to a better understanding of the system's behaviour under various operating conditions. Formulae for its performance in terms of sampling frequency, dynamic range, memory length, syllabic time constant and signal-to-noise ratio are presented.

A computer aided design technique for replacing the analogue feedback loop by a digital one is proposed. The significance of this proposal is that a completely digital SCALE system can be constructed.

The technique is then used to design two converters one of which accepts A-law PCM signals and converts them to SCALE binary data, while the other converter accepts SCALE binary stream and produces A-law PCM words.

*The entire conversion processes are performed completely digitally, and the design technique is general for converting from any PCM format to any Delta modulation system and vice versa.*

*Finally, the potential applications of the SCALE - to - A-law PCM and A-law PCM - to - SCALE converters are discussed.*

### ACKNOWLEDGEMENTS

The author wishes to acknowledge the technical direction of Dr. R. Steele who suggested the investigation on which this thesis is based, and to thank him for his encouragement and useful discussions.

The assistance of Professor M. N. Faruqi with whom the author had useful discussions is likewise acknowledged.

The author is also indebted to The University of Technology, Loughborough, for providing the research facilities, and to The Arabian Gulf Company (Libya) for the financial grant supporting him during the period of this research.

Finally, thanks are also due to Mrs. M. Connor who typed this thesis.

## CONTENTS

	<u>Page</u>
Synopsis	i
Acknowledgements	iii
Contents	iv
List of Principal Symbols	vii
List of Figures	xiii
List of Tables	xix
1. <u>INTRODUCTION</u>	
1.1.1 Background	1
1.2.1 Digital coding of speech and V.F. data	2
1.2.2 Developments in DM systems	3
1.2.3 Matching the input source to the coder overload characteristic	5
1.2.4 Delta-sigma modulation system	5
1.2.5 Advantages of DM systems	6
1.2.6 Selection of the adaptation algorithm for DM systems used in the encoding of speech	8
1.3 Statement of the problem	10
2. <u>THEORY OF DELTA-SIGMA MODULATION WITH DIGITALLY CONTROLLED SYLLABIC COMPANDING</u>	
2.1.1 Introduction	14
2.1.2 The basic SCALE system	14
2.2.1 The generalized model of SCALE	17
2.2.2 Basic equations and operation of the modified SCALE system	18
2.2.3 Calculation of the system performance	24
3. <u>COMPUTER SIMULATION OF THE SCALE SYSTEM</u>	
3.1 Introduction	31
3.2.1 Statement of the objective	33
3.2.2 Partitioning of the problem	33
3.2.3 Program analysis and coding	34

3.2.4	Debugging and verification of the simulation subprograms	45
3.2.5	Using the model to study the SCALE system	45
3.3	Summary of the simulation results	48
3.4	Evaluation of the constant " $\delta$ "	50
4.	<u>HARDWARE MODEL, PERFORMANCE MEASUREMENTS</u> <u>AND PERFORMANCE CALCULATIONS</u>	
4.1	Practical implementation of the SCALE system	52
4.1.1	Circuit synthesis and description	52
4.1.2	Construction of the CODEC	55
4.1.3	Testing and adjustment of the CODEC system	56
4.2	Investigation into the behaviour of the CODEC under sinusoidal inputs	57
4.2.1	Observations at the coder end	57
4.2.2	Observations at the decoder end	58
4.3	Performance measurements	59
4.3.1	SNR as a function of the input power, $f_p$ , $T_2$ and $N_e$	63
4.3.2	Analysis of the results	64
4.4	Computer program for analysing the results	66
5.	<u>DIGITIZATION OF THE SCALE SYSTEM</u>	
5.1	Introduction	69
5.2	Description of the modified system	70
5.3	Theory of the adaptive accumulator	71
5.4	General design procedure	74
5.5	Design example	77
5.5.1	Segmentation and evaluation of the adaptation algorithm	77
5.6	Computer simulation (software approach)	79
5.7	Hardware approach to the implementation of DSCALE	91
6.	<u>SCALE - TO - A-LAW PCM CONVERSION</u>	
6.1	The DSCALE word generator	98
6.2	The frequency divider and digital interpolator	99
6.3	The digital compressor	103
6.3.1	Logarithmic companding laws - background	103
6.3.2	Theory of segment laws	104
6.3.3	The compression algorithm and synthesis of the compressor	108
6.3.4	Operation of the digital A-law PCM compressor	110

6.4	Computer simulation of the SCALE-to-PCM converter	112
6.4.1	The SCALE-to-linear PCM converter	112
6.4.2	The linear PCM - to - A-law PCM compressor	116
6.5	System performance	120
6.6	Implementation of the SCALE - to - A-law PCM converter	122
6.6.1	Circuit description of the digital SCALE decoder	123
6.6.2	Circuit description of the digital interpolator	123
6.6.3	Circuit description of the A-law PCM digital compressor	125
7.	<u>A-LAW PCM - TO - SCALE CONVERSION</u>	
7.1	Introduction	127
7.2	Description of the all-digital converter	128
7.2.1	The digital expander (theory)	128
7.2.2	The linear PCM-to-SCALE converter	130
7.3	Implementation of the A-law PCM - to - SCALE converter	134
7.3.1	The digital expander (implementation)	134
7.3.2	The full subtractor	136
7.3.3	The digital de-emphaiser	138
7.3.4	The adaptive digital accumulator	140
7.4	Computer simulation of the A-law PCM - to - SCALE converter	140
7.5	Summary of results	142
8.	<u>GENERAL DISCUSSION AND CONCLUSIONS</u>	
8.1	General discussion	145
8.2	Conclusions	147
	<u>REFERENCES</u>	151

## LIST OF PRINCIPAL SYMBOLS

- A, B, C = The states (contents) of the 3-stage shift register
- A, B, C, = The logical values of the contents of the 6-bit shift  
D, E, F  
register stages
- BW = The number of frequency samples of  $|H(j\omega)|$  which are equated to unity
- $C_1R_1 = T_1$  = The time constant of the SCALE forward integrator
- $C_2R_2 = T_2$  = The time constant of the syllabic integrator
- DR = Dynamic Range
- DFT = Discrete Fourier Transform
- DSCALE = Digital SCALE system
- DINT = Digital interpolator
- d = The peak-to-peak amplitude of the integrator error signal during the idling condition and  $V_c = V_{co}$
- $\Delta f$  = Separation between frequency samples of  $|H(j\omega)|$
- $\Delta_c(J_s)$  = The value of the constant increment in segment  $J_s$
- $\Delta'_c$  = The value of the constant increment at a segment edge
- $\delta$  = A constant relating the input power to the feedback power
- $E_D$  = The difference between the input and the feedback signals in the DSCALE system
- $E_I$  = The integral of  $E_D$
- $\epsilon_1(t)$  = The difference between the input  $x_2(t)$  and the feedback signal  $H(t)$
- $\epsilon_2(t)$  = The integrated error seen at the input of the basic SCALE quantizer
- $\eta_c$  = The factor by which SNR is modified as a result of the presence of channel errors at the input of the SCALE decoder

$\eta_a$	= A parameter associated with the segmented A-law companding law. It equals 0 in the first segment and equals 1 otherwise
$F_1$	= The SCALE encoder input filter
FIR	= Finite Impulse Response (in digital filters)
FFT	= Fast Fourier Transform
$F_2$	= SCALE decoder output filter
$f_{b1}$	= The lower cut-off frequency of $F_1$
$f_{b2}$	= The upper cut-off frequency of $F_1$
$f_p$	= Sampling frequency of the SCALE system
$f_b$	= Lower cut-off frequency of $F_2$
$f_a$	= Upper cut-off frequency of $F_2$
$f_1$	= The characteristic frequency of $C_1R_1$
$f_s$	= Sinusoid frequency in $H_z$
$\phi(z)$	= The transfer function of the digital interpolator
$\phi(f)$	= The frequency response of the digital interpolator
G	= The gain of the PHM
$H(t)$	= The feedback step height of the basic SCALE system
$H_{min}$	= The minimum value of the magnitude of the feedback step height $H(t)$
$H_{max}$	= The maximum value of the feedback step height magnitude
$H_p$	= The r.m.s. value of the feedback step height calculated over approximately one pitch period
$H_i$	= The value of $H(t)$ at the $i^{th}$ clock instance
$ H(j\omega) $	= Frequency response of the low pass digital filter
$H_D$	= DSCALE code word
$H_{Di}$	= DSCALE code word seen at the $i^{th}$ clock instance
$\{H_{Dk}\}$	= The set of $H_D$ values corresponding to the values of k
$\overline{H^2}$	= The mean square value of H (the SCALE feedback signal)
$\hat{H}_i$	= The instantaneous value of the output signal of the DSCALE decoder

- $\hat{H}_i$  = The instantaneous value of feedback signal of the PCM-to-SCALE converter in analogue form
- $h(\ell)$  = Impulse response of the low pass digital filter
- IDFT = Inverse DFT
- I/P = Input
- $J_s$  = The segment number in the context of compressed codes
- $\hat{J}_s$  = The compressed code segment number found by searching the corresponding LPCM code word
- $k$  = Constant of proportionality relating the uncompanded Delta-sigma modulator noise to the step height, sampling frequency, etc.
- $k_c$  = Constant relating SCALE quantization noise power to  $f_p$ , B and  $H(t)$
- $k_n$  = A constant relating the noise r.m.s. voltage to the mean square value of the feedback signal in a DSCALE system
- $L(t)$  = The SCALE output binary sequence transmitted to the channel and terminal equipment
- $L_i$  = The logical level of  $L(t)$  at the  $i^{\text{th}}$  clock instance
- LPCM = Linear PCM code
- LSD = Least Significant Digit
- $L_g$  = The leakage of the digital integrator used in the A-law PCM - to SCALE converter
- $\hat{L}(t)$  = The SCALE binary data generated by the A-law PCM - to - SCALE converter
- $\ln$  = The natural logarithm
- $\lambda_{12}$  = The polarity of the integrated error of the A-law PCM - to - SCALE converter
- $\lambda_{12}(i)$  = The value of  $\hat{L}(t)$  at the  $i^{\text{th}}$  clock instance
- $M$  = Number of frequency samples in  $|H(j\omega)|$

MU	=	The mean of the Gaussian noise waveform
MSD	=	Most Significant Digit
$m_1$	=	The characteristic number of a compressed code word
$m_2$	=	The mantissa number of a compressed code word
mv	=	Millivolts
$N_e$	=	The effective SR length, i.e. the number of SR stages to be used
$N_q^2$	=	Quantization noise power residing in the output signal band
$N_{eu}^2$	=	The noise power resulting from channel errors in an un-companded Delta-sigma modulation system
$N_{ec}^2$	=	The noise power due to channel errors in the SCALE system (companded)
$N_T^2$	=	Total noise power taking the channel errors into account
$N_r$	=	The number of clock periods during which an $N_r$ DSCALE samples are averaged to produce one LPCM sample
$N_t$	=	The total number of steps in a segment (= 16 for A-law PCM)
O/P	=	Output
PHM	=	Pulse Height Modulator
P	=	Polarity
$P_{10}$	=	The probability that a digit is transmitted as a logical one and received as a zero or vice versa
$Q_i$	=	The logical value of $V_q$ at the $i^{\text{th}}$ clock instance
$\rho$	=	The minimum acceptable SNR
$R_c$	=	The maximum available signal-to-noise ratio in the hybrid SCALE system
SR	=	Shift Register
$s_0, s_1 \dots s_7$	=	States of the contents of the SR
SNR	=	Signal-to-Noise Ratio
$SNR_e$	=	SNR modified by the channel errors

- $S_q$  = The step number within a segment of the compressed code  
 $\hat{S}_q$  = The number of the step in segment  $\hat{J}_s$  found from the LPCM code word  
 $|\psi|^2$  = The quantization noise power density  
 $\sigma_x$  = The input r.m.s. voltage calculated over approximately one pitch period  
 $\sigma_m$  = The input signal power which just produces  $\rho$   
 $\sigma^2$  = The variance of the Gaussian noise waveform  
 $T_p = \frac{1}{f_p}$  = Clock period  
 $T'_p$  =  $T_p/8$   
 $t$  = Time  
 $\tau_p$  =  $256T_p$  (approximately one speech pitch period)  
 $\omega$  = Angular frequency  
 $X_L$  = Linear PCM code word  
 $X_c$  = A-law PCM code word  
 $x_1(t)$  = The analogue input signal to the encoder input filter  
 $x_2(t)$  = The band limited analogue signal at the output port of  $F_1$   
 $\hat{x}_2(t)$  = The filtered SCALE decoder output  
 $(x_1)_i$  = The value of  $x_1(t)$  at the  $i^{\text{th}}$  clock instance  
 $(x_2)_i$  = The value of  $x_2(t)$  at the  $i^{\text{th}}$  clock instance  
 $x'_2(t)$  = The low pass analogue sample corresponding to  $X_c$   
 $\hat{\hat{x}}_2(t)$  = The filtered output of the DSCALE decoder whose input is  $L(t)$   
 $\tilde{x}_2(t)$  = The filtered output of a SCALE decoder whose input is  $\hat{L}(t)$   
 $V_q$  = The logical level of the quantizer output  
 $V_{cM}$  = The value of  $V_c(t)$  at which the PHM saturates  
 $V_c(t)$  = The analogue voltage output of the syllabic integrator  
 $V_{co}$  = The initial value of the control voltage  $V_c(t)$  (the output of the syllabic integrator when the encoder is first switched on)

- $V_{cp}$  = The r.m.s. voltage of the syllabic integrator output calculated over a time period which is approximately one pitch period
- $V_{nl}$  = The noise voltage contribution of the SCALE - to - A-law PCM converter
- $Y_A$  = The analogue equivalent of  $X_c$  after linearization and D/A conversion
- $Z$  = The output voltage level of the exclusive OR gate
- $Z_h$  =  $Z$  when  $N_e$  consecutive like digits are stored in the SCALE shift register
- $Z_k$  =  $Z$  when the shift register contents are not like digits
- $z$  = The z-transform ( $= e^{-sT_p}$ )
- $H_Z$  = Hertz (basic unit of frequency)
- dB = 10 log (power ratio)
- dBm = A unit for expressing power level in decibel relative to one milliwatt
- dBm<sub>o</sub> = dBm when the impedance in which the power being measured is 600Ω
- $P_{in}$  = Input power in dB at the input port of the SCALE encoder amplifier
- $P_0$  = Power at the input port of the subtractor circuit
- $P_1$  = The input power in dB which is just large enough to provide adequate SNR
- $P_2$  = The input power (in dB) beyond which overload noise reduces SNR to an unacceptable level
- QDM = Quantization distortion meter

$f_q$  = Nyquist rate (8 KHz)

PIPO = Parallel In Parallel out (shift register)

## LIST OF FIGURES

		<u>Page</u>
Figure 1.1	Block diagram of the basic linear Delta-modulation system	3a
Figure 1.2	Block diagram of the basic Delta-sigma system	6a
Figure 1.3	Block diagrams showing the principles of instantaneous and syllabic companding in Delta-modulation systems	9a
Figure 2.1	Block diagram of the basic SCALE system	14a
Figure 2.2	The pulse height modulator characteristic	16a
Figure 2.3	Block diagram of the generalized SCALE system	18a
Figure 2.4	The error, and the integrated error waveforms during the idling condition	19a
Figure 2.5	State tables and state diagrams for the effective shift register lengths $N_e = 2$ and $N_e = 3$	21a/b
Figure 2.6	Syllabic integrator output voltage $V_c$ as a function of the group length	22a
Figure 3.1	Block diagram of the partitioned SCALE system model for computer simulation	33a
Figure 3.2	Examples of the signal waveforms generated by "SUBROUTINE SIGNAL"	34a
Figure 3.3	Speech signal waveform at the output of the SCALE encoder filter	35a
Figure 3.4	The FIR digital filter characteristics	37a
Figure 3.5	Characteristics of the simulated low pass digital filter	41a
Figure 3.6	Simulated Gaussian noise waveforms	41b
Figure 3.7	Flow diagram of computer runs for the purpose of waveform observation	46a
Figure 3.8	Control voltage as a function of time, feedback step height as a function of control voltage, integrated error as a function of time and SCALE input and out speech waveforms	46b-f

Figure 3.9	Sections of the observed input and re-constructed signals of the HSCALE system when the input is a sinusoid	47a-c
Figure 3.10	The control voltage $V_{cp}$ and feedback signal $H_p$ as functions of the r.m.s. voltage $\sigma_x$ of the input signal	48b
Figure 3.11	Signal-to-noise ratio SNR as a function of the input power in $\text{dB}_{\text{mo}}$ (by simulation)	50a
Figure 3.12	Evaluation of the design parameter $\delta$ employing interative techniques	51a/b
Figure 4.1	The SCALE system input filter	53a/b
Figure 4.2	Circuit diagram of the analogue section of the SCALE system	53c
Figure 4.3	The SCALE output filter	54a/b
Figure 4.4	The digital part of the SCALE system	54c, 55a
Figure 4.5	Photographs of the SCALE system hardware model	55b, 55c
Figure 4.6	Testing and adjustment of the pulse height modulator and the comparator	56a/b
Figure 4.7	The waveform observation experiments (encoder end)	57a/b
Figure 4.8	The waveform observation experiment (decoder end)	58a/b
Figure 4.9	Set-up for the measurements of SNR using Marconi TF 2343A	62a
Figure 4.10	Signal-to-noise ratio, the dynamic range, $k_c$ and $\delta$ as functions of the input power, sampling frequency, shift register length and syllabic time constant	63a-d, 65b
Figure 4.11	SNR curves obtained by measurement, calculation and computer simulation (a comparison)	66a
Figure 5.1	Block diagram of the DSCALE	70a
Figure 5.2	Block diagram of the adaptive accumulator	72a
Figure 5.3	The growth and decay of $H$ with the group length $t$ (and time in clock periods)	75a

Figure 5.4	The step height of the feedback signals generated by the HSCALE and the DSCALE decoders as functions of the group length $l$	76a
Figure 5.5	Segmentation of the SCALE feedback stepheight (for the design example)	77d
Figure 5.6	Flow chart for the generation of digital signals for the testing of the SCALE digital decoder	79a
Figure 5.7	Block diagram of the adaptive digital accumulator using the counting techniques	80a
Figure 5.8	Block diagram of the digital-to-analogue converter D/A	86a
Figure 5.9	Flow chart of the digital SCALE decoder simulation program	86b
Figure 5.10	Flow chart of the program used for testing the digital SCALE decoder	87a
Figure 5.11	Comparison between the hybrid and the digital SCALE systems	87b/c
Figure 5.12	Flow diagram of the computer procedure used for calculating $\sigma_x$ and $H_p$	89a/b
Figure 5.13	Computation of $SNR_d$ as a function of the input power (computer program flow chart)	89c
Figure 5.14	Modification to the simulated input signal to enable the comparison between practical and the simulated models	90a
Figure 5.15	Graphs of SNR versus input power obtained by simulation of the hybrid and the digital SCALE systems	90b
Figure 5.16	Generation of the Z control signal (circuit diagram)	92a
Figure 5.17	Block diagram of the adder-subtractor	93a
Figure 5.18	Circuit diagram of the adder-subtractor	94a
Figure 5.19	The parallel-in-parallel-out shift register	94b/c

Figure 5.20	The segment logic	95a/b
Figure 5.21	Generation of the $\Delta$ address signals	95c-d,96a
Figure 5.22	The generation of $\Delta_d(Z,J)$	96a-c
Figure 5.23	Selection of the final output from the accumulator	96d/e
Figure 6.1	Methods of SCALE - to - A-law PCM conversion	97a
Figure 6.2	An all-digital SCALE - to - PCM converter	98a
Figure 6.3	Realization of equation (6.1) using a digital accumulator and a divider (block diagram)	100a/b
Figure 6.4	Equivalent diagrams of the resettable accumulator and divider	101a
Figure 6.5	The transfer function of the digital interpolator in the frequency domain	102a
Figure 6.6	Segmented A-law (standard form)	104a
Figure 6.7	Quantization and coding used in CEPT 30+2 A-law PCM system	104b
Figure 6.8	The digital compressor	109a
Figure 6.9	Block diagram of the SCALE-to-PCM converter (arrangement for computer simulation)	112a
Figure 6.10	Flow chart of the program employed for simulating and testing of the SCALE-to-LPCM converter	114a/b
Figure 6.11	Results of the testing of the SCALE-to-LPCM converter	115a-c, 116a-d
Figure 6.12	Flow chart of the computer program employed for testing the SCALE-to-LPCM converter	117a-c
Figure 6.13	The compressor and the exapnder characteristics for the A-law PCM system	119a,120a
Figure 6.14	Comparison between the outputs from DSCALE decoder, SCALE-to-LPCM converter followed by LPCM decoder, and SCALE - to - A-law PCM converter followed by an A-law decoder - when the input in all cases is a SCALE binary data pattern	120b

Figure 6.15	Equivalent of the DSCALE and the A-law PCM decoders, and a comparison between the performances of the SCALE system and the A-law PCM receivers when a SCALE - A-law PCM converter is employed	121a,122a
Figure 6.16	Block diagram of the SCALE-to-LPCM converter (for hardware implementation)	123a
Figure 6.17	Implementation of the digital interpolator	123b/c, 124a-e
Figure 6.18	Implementation of the A-law PCM compressor	125a, 126a-c
Figure 6.19	Implementation of the timing circuit	126d-g
Figure 6.20	Implementation of the $\hat{J}_s$ search (segment number determination) circuit and frequency divider	126h-j
Figure 7.1	Two possible approaches to the problem of A-law PCM - to - SCALE conversions	127a
Figure 7.2	Block diagram of the all-digital A-law PCM - to - SCALE converter	128a
Figure 7.3	Block diagram of the parallel digital A-law PCM expander (A-law - to - linear PCM converter)	130a
Figure 7.4	The forward paths of the DSCALE and the HASCALE, and a block diagram of the digital de-emphasiser	132a/b
Figure 7.5	Circuit diagram of the digital expander	134a
Figure 7.6	Implementation of binary full subtractors	137a
Figure 7.7	Implementation of the 8-bit by 11-bit binary multiplier	139a,140a
Figure 7.8	Flow diagram of the computer program used for the simulation of the A-law - to - SCALE converter	141a/b
Figure 7.9	Computer simulation results (sinusoidal input experiment)	143a/b

Figure 7.10	Computer simulation results (arbitrary waveform input)	143c/d
Figure 7.11	Simulation results (step response experiment)	144a
Figure 7.12	Simulation results (comparison between $\hat{H}(t)$ and $\hat{H}(t)$ )	144b

LIST OF TABLES

		<u>Page</u>
Table 2.1	The design parameters of the basic SCALE system employed in Wilkinson	17a
Table 3.1	Results of 28 computer runs for the calculation of the r.m.s. voltages of the input signal $\sigma_x$ , the control voltage $V_{cp}$ and the feedback step voltage $H_p$	48a
Table 4.1	Calculation of $\delta$ , $k_c$ and DR from the measured values of the maximum SNR ( $R_c$ ), $f_p$ and $\sigma_x$	65a
Table 5.1	The increment magnitudes $\Delta_d$ , per clock period, as a function of the segment number J and the control logic output Z	75b
Table 5.2a	The binary data pattern L(t) generated by the hybrid SCALE during a $512 T_p$ second period for the cases where the input is (i) Constant step of 5 volts (ii) Zero volts (iii) Sinusoid of frequency $1\text{KH}_z$ and amplitude of 1.5 volts (iv) Bandlimited Gaussian signal	77a-c
Table 5.2b	Design parameters of the hybrid SCALE used in the design of the DSCALE	77d
Table 5.3	Segmentation of the SCALE feedback step height magnitude for the DSCALE design example	77d
Table 5.4	The values of $\Delta_d(Z,J)$ for the DSCALE design example. The values found by computer simulation	78a
Table 5.5	The partitioned value of the step height increment $\Delta_d(Z,J)$ into $\Delta'(Z,J) + \Delta''(Z,J)$	81a

## CHAPTER I

### 1. INTRODUCTION

#### 1.1.1 Background

Analogue circuits still dominate the telecommunications network of today, but the large number of digital facilities which have been introduced into the system, and the ever increasing interest in digital processing, transmission and switching indicate that in the future digital systems are likely to become predominant<sup>(6)</sup>.

In recent years the telephone industries in Europe, America and Japan, for example, have made large investments in digital transmission systems for interexchange communications<sup>(7)</sup>; and more investments are expected to be made to digitize the subscriber loop plant in the near future<sup>(44)</sup>.

In the fields of law enforcement and military communication systems, digital techniques are already in use and in the future many systems which are presently analogue are expected to be phased out by digital ones.

The development of digital techniques in communications, and the advantages of digital signals over analogue ones have been widely discussed in the literature<sup>(1-5)</sup>. For completeness, however, the main factors in favour of digital systems are summarized:

- (i) In digital transmission systems the signal quality can be independent of the distance between the sending and the receiving terminals. This is because digital signals can be regenerated (i.e. retimed and reshaped) at intermediate points along the transmission path, without suffering any degradation.
- (ii) The digital system lends itself to more efficient storage and various digital processings such as encryption (digital

- scrambling), code conversion and digital filtering.
- (iii) If multiplexed, digital signals enable the transmission channel capacity in many circumstances to be increased (e.g. cable pairs originally intended for single telephone channels can carry 30 telephone conversations in PCM coded form).
  - (iv) The required transmitter power can be much less than that required in analogue systems, and the reliability of transmission is much higher. These factors make digital techniques more suitable for satellite and computer-controlled communications.
  - (v) In specialized applications where noise can exceed the signal level, digital systems can still extract the information by introducing redundancy into the transmitted codes. The main disadvantage of the digital system is that when used without multiplexing the bandwidth required is larger than the baseband bandwidth.

#### 1.2.1 Digital coding of speech and V.F. data

In the fields of speech and voice frequency (V.F.) data communications, the most widely used digital formats are Logarithmic Pulse Code Modulation (Log. PCM)<sup>(1,6)</sup>, and Adaptive Delta Modulation (ADM)<sup>(2,8,45)</sup>. The digital coding used on telephone interexchange junction circuits in America and Japan is a logarithmically-companded PCM known as the  $\mu$ -law PCM<sup>(9-11,96)</sup>, while in Europe a similar PCM format known as the A-law PCM is employed<sup>(1,12)</sup>. Recently, however, for reasons relating to international network compatibility, The International Telecommunication Union (I.T.U.) regulations have standardized the coding parameters (sampling rate, bit rate, companding law, etc.), as a result of which the A-law PCM has been recommended for all interexchange transmission and switching applications<sup>(7,12)</sup>.

To provide satisfactory telephone quality in the presence of several Analogue-to-Digital-to-Analogue (ADA) conversions, which may be encountered

in the network, the standardized bit rate will be 64 Kb/s.

In the case of the subscriber loop and switching plant there are very few, or no, ADA conversions and the above ITU standardization is not imposed on the users. For these reasons many suggestions have been made to employ other forms of digital coding such as Differential PCM (DPCM) and ADM in the local network<sup>(13,14)</sup>. Because of their superior performance under channel error conditions, the trend is likely to be towards the use of adaptive Delta and Delta-sigma modulation systems.

ADM can offer some advantages over A-law PCM in many other applications. Consequently various military and civil communication authorities have decided to use it in preference to PCM. Examples of such authorities are the British Army and EUROCOM (a European military communication organisation which performs a similar task to that of the C.C.I.T.T.). The first employ a syllabically-companded Delta-sigma modulation system known as SCALE<sup>(15-19)</sup> (Syllabically Companded And Logically Encoded) while the second employ Delta Modulation system with digitally controlled syllabic companding<sup>(20-21)</sup>. The latter system has been also employed in the airborne equipment of the European Communication Satellite EROSAT<sup>(22)</sup> which was recently (in 1974) launched by the European Space Research Organization (ESRO). A similar system is used in the Canadian communication satellites, known as ANIK I and ANIK II, for thin route (low capacity) speech communication<sup>(23)</sup>.

### 1.2.2 Developments in DM systems

Since its discovery, over twenty-five years ago<sup>(8)</sup>, DM has been the focus of attention of many research workers. The basic linear Delta-modulator is an analogue-to-digital converter, the principle of which is shown in Fig. 1.1. The encoder uses a feedback path comprising an integrator network which produces a feedback signal  $y_0(t)$  to track the input signal  $x(t)$ . The input to the integrator is  $L(t)$  - which is also

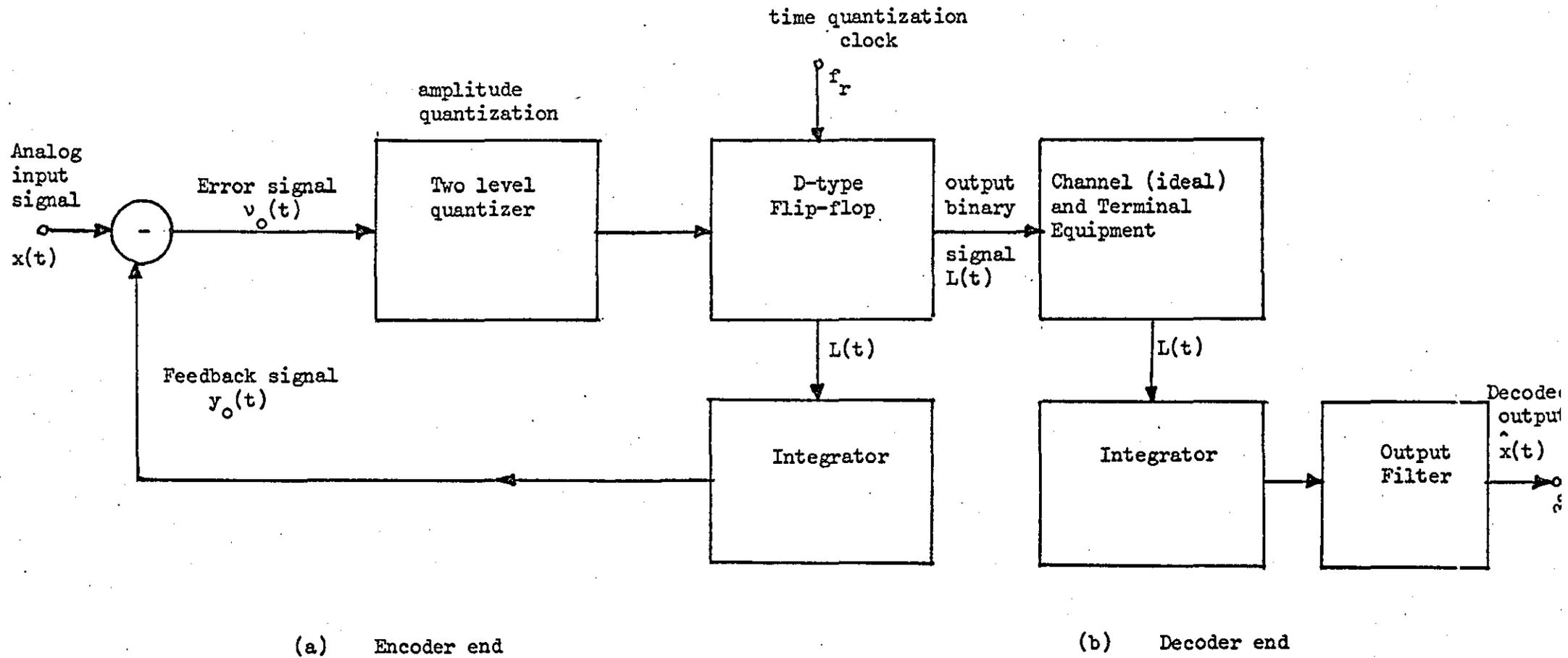


Figure 1.1 Block diagram of the Basic Linear Delta Modulation System

transmitted to the channel - and consists of blocks of like or alternating pulses the length and direction of which are functions of the slope of the input voltage waveform. The two level quantizer output is a logical one if the difference signal  $v_o(t)$  is zero or positive and it is a logical zero if  $v_o(t)$  is negative. Time quantization is achieved by clocking the flipflop at the required bit rate to produce  $L(t)$ .

At the remote receiver - assuming a perfect channel -  $L(t)$  is integrated by an integrator which is identical to the one employed by the encoder to produce  $y_o(t)$ . A low pass filter removes the out-of-band noise present in  $y_o(t)$  to construct  $\hat{x}(t)$  which is similar to the original signal  $x(t)$ .

The process of linear Delta modulation, employing an ideal integrator<sup>(2)</sup>, can be expressed mathematically as

$$\begin{aligned} y_o(t) &= \int L(t) dt \\ &= x(t) + v_o(t) \end{aligned} \quad (1.1)$$

where  $v_o(t)$  is known as the quantization noise.

When  $y_o(t)$  is filtered  $\hat{x}(t)$  is produced. Therefore

$$\hat{x}(t) = x(t) + v'_o(t) \quad (1.2)$$

where  $v'_o(t)$  is the baseband component of the quantization noise  $v_o(t)$ .

The linear DM parameters, such as the overload characteristics<sup>(2,24-26)</sup>, quantization noise<sup>(2,24,29)</sup>, quantizer hysteresis<sup>(2,30)</sup> and idle channel noise have been the subject of many discussions. The linear single-integration Deltamodulator (DM) did not find widespread application. The main reason for this was its limited dynamic range - DR (DR can be defined as the range of the input signal within which the signal-to-noise ratio is not less than a subjectively acceptable value and depends on the field of application). This lack of DR in linear DM is due to its fixed step size which causes its quantization noise spectrum, at the decoder output, to be approximately flat and consequently the signal-to-noise ratio

(SNR) of the decoded output decreases with falling input levels.

Many suggestions have been made for a refinement of the simple DM process in order to achieve acceptable dynamic range. The Constant factor DM, developed at Loughborough University<sup>(32,52)</sup>, and High information DM<sup>(31)</sup> are examples of the refined Deltamodulation systems designed for television encoding. Similarly, Continuous DM<sup>(33)</sup>, Digitally controlled adaptive DM<sup>(20)</sup>, Syllabically Companded Delta-sigma Modulator<sup>(15)</sup>, SCALE<sup>(78)</sup>, and Two-loop DM<sup>(90)</sup> are a few examples of DM systems which have been suggested for the encoding of speech (see also references 34,35) 36 and 37). Other examples of improved DM systems are those suggested by Betts<sup>(91)</sup> for improving HF transmission, and the FX209 A/D converter being developed by Consumer Microcuits Ltd.<sup>(92)</sup>, for domestic applications.

### 1.2.3 Matching the input source to the coder overload characteristic

It is known that the linear Deltamodulator overload characteristic falls with frequency while telephone microphones, and microphones used in military applications, are designed to pre-emphasise these higher frequencies to improve intelegibility. This inherent pre-emphasis in the microphones - which are normally connected to the input of the DM encoder - causes the input spectrum to be flat over the whole baseband. Because of this mis-match between the input spectrum to the simple Delta-modulator and its overload characteristic a matching circuit must be inserted between the microphone and the encoder. In practice the required matching (de-emphasis) circuit is a simple CR network which transforms the simple Deltamodulator to what is known in the literature as Delta-sigma<sup>(46)</sup> modulator. The principle of operation of the Delta-sigma modulation system is reviewed below.

### 1.2.4 Delta-sigma modulation system

The linear Deltamodulator described above has a binary output signal  $L(t)$  which carries information about the differential of the input signal. Its dynamic range and Signal-to-Noise Ratio SNR are inversely proportional to the signal frequency<sup>(8)</sup>. The inevitable integration at the remote

decoder causes the effect of the channel error to be accumulative when the system is subjected to transmission disturbances. Inose, Yasuda and Murkami<sup>(46,47)</sup> have therefore proposed a modification to the simple Delta Modulator to enable it to cope with signals the base-band spectrum of which is flat and which has to be transmitted through adverse channel conditions. The system is known as Delta-sigma modulation and its principle of operation is demonstrated in Fig. 1.2.

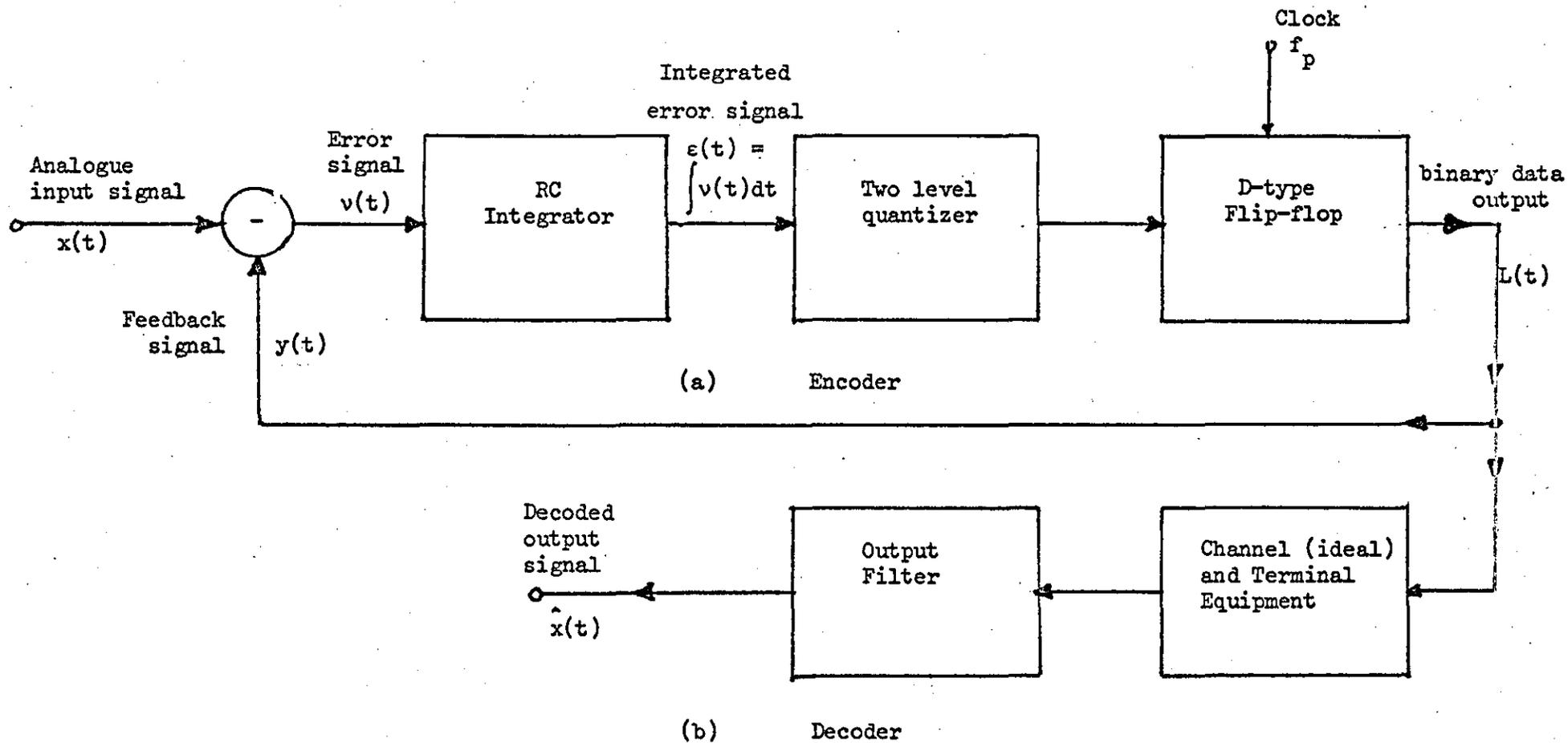
In this figure, the binary output signal  $L(t)$  is subtracted from the input signal  $x(t)$  to produce a difference signal  $v(t)$ .  $v(t)$  is integrated by the RC integrator, which is now in the forward path, to produce  $\epsilon(t)$ . The comparator compares  $\epsilon(t)$  with a reference voltage  $V_r$ , and produces a logical one when  $\epsilon(t) \geq V_r$ , and a logical zero when  $\epsilon(t) < V_r$ . The time quantization is achieved as before by clocking the bistable at the required bit rate. Through this negative feedback process, the integrated difference is kept in the vicinity of the reference  $V_r$ .

The frequency of occurrence of the blocks of consecutive like digits in  $L(t)$  is a function of the magnitude of the input signal which means that  $L(t)$  in this system carries information about the input signal amplitude. The digital data received by the decoder at the end of an ideal channel is reshaped and passed directly through a low-pass filter to reconstruct  $\hat{x}(t)$  which is an estimate of  $x(t)$ . This system avoids the accumulation of channel errors because there is no integration circuit in the decoder and its overload characteristic does not fall with frequency.

### 1.2.5 Advantages of DM systems

Comparison between PCM and DM has been the subject of many discussions and investigations<sup>(38,39,42)</sup>. For completeness, however, the main advantages of DM systems over PCM are reviewed:

- (i) Better performance than PCM under equal channel error and bit slip



6a

Figure 1.2 Block diagram of the Basic Delta-sigma System  
 (a) Encoder, (b) Decoder

conditions (channel error is due to noise in the channel causing logical ones to be received as logical zeros and vice versa, while bit-slip is due to loss of synchronization between the coder and the decoder terminal). In PCM a single bit error gives an absolute amplitude error which may have any value between minimum and maximum step height. In DM systems used for speech (single or double integration Delta and Delta-sigma with syllabic companding), a bit error causes very small amplitude variation<sup>(40)</sup> leading to a better performance under equal sampling and channel error conditions as those of the PCM.

- (ii) Being one bit code, no word synchronization is required.
- (iii) Flexibility. In DM the bit rate can be adjusted between wide limits (e.g. 10 to 100 Kb/s) to obtain the required compromise between acceptable signal quality and channel bandwidth. This adjustment can be made without the need to modify the system in any way except the sampling frequency. Hence an externally variable clock enables signal quality to be traded for bandwidth and vice versa.
- (iv) A comparison with A-law PCM shows that Adaptive DM systems can achieve comparable SNR when the two systems operate at equal bit rates in the range 40-60 Kb/s. At lower bit rates (below 40 Kb/s), however, Adaptive DM performs better. Subjective tests indicate that adaptive DM is generally preferred over A-law PCM if both systems operate at equal speeds and within the range given above. This comparison experiment, results of which have been recently published by Schindler<sup>(41)</sup>, agree with Jayant's results<sup>(42)</sup>, and confirm the superior performance of adaptive DM over log-PCM for speech encoding at low bit rates.
- (v) Although adaptive DM systems are more complicated than the linear Deltamodulator, it can be argued that they can still be classified

as 'simple' when compared with other digital coding systems such as PCM, at least from the point of view of filtering and decoding.

#### 1.2.6 Selection of the adaptation algorithm for DM systems used in the encoding of speech

Speech belongs to the class of non-stationary signal sources. Its spectral properties as well as power level vary with time. The effect of speaker variations (loud talkers, quiet talkers, etc.), different line losses, and variations in the distance between the speaker's mouth and the telephone microphone, must be taken into account when speech communication systems are considered. Purton<sup>(43)</sup> shows that if all the above factors are allowed for a speech communication system can be required to handle a total dynamic range of the order of 62 dB. To accommodate this dynamic range the encoder must be able to continuously adjust its feedback step height in such a manner that the error signal at the summing junction is kept to a minimum, and the encoder operation is consequently kept near the overload point where the SNR is near its maximum value. This situation is equivalent to compressing the variations in the input signal to an encoder whose feedback step height is constant. The two techniques lead to the density of the strings of consecutive like digits in the channel being reduced. To restore the decoded signal to its original form, a reciprocal operation, or its equivalent, must be performed at the receiver.

In practice the dynamic range of the encoder is considered to be adequate if it can handle signal variations in the range of 30-40 dB. In these circumstances a median talker (not extremely loud nor extremely quiet) is assumed and the variations due to different line losses and speaker-microphone distances are assumed to be not very severe. The practical dynamic range can be achieved by making the ratio of the maximum step to the minimum step heights, of the feedback signal, of the order of 100. (Minimum and maximum step sizes here mean the minimum and maximum

values of  $y(t)$  respectively.)

Now at the onset of voiced speech the power level may rise rapidly; but due to the nature of the vocal tract the decay of this power level is relatively slower. For a DM encoder, designed for speech, the adaptation algorithm should be matched to these dynamic properties of the input signal.

Delta modulation systems have used two forms of adaptation. These are known as quasi-instantaneous companding, and syllabic companding.<sup>(83)</sup> In the former the digital code is monitored over a window of a few clock periods (usually 2-8 clock periods) and the step height of the feedback voltage is then instantaneously incremented upward or downward within a finite family of discrete step sizes. The sizes of these increments and their direction is decided by the lengths of consecutive-like-digit strings within the monitoring window and their polarities.

A general arrangement for implementing quasi-instantaneous companding in DM systems is shown in Fig. 1.3a. Jayant<sup>(51)</sup>, for example, has suggested that a suitable adaptation algorithm for single integration Deltamodulator step size can be achieved by using one-bit delay element (shift register) to store the last bit transmitted to the channel. The previous step value is stored in the adaptation logic and its magnitude is respectively doubled or halved depending on whether the content of delay element and the present output digit are two like or two different consecutive digits generated by the encoder. In the case of syllabic companding the digital data is viewed through a window of  $N$ -clock periods, say, and when  $N$  consecutive like digits are detected inside the window, a pulse of voltage is generated. This pulse is applied to a leaky integrator (i.e. a CR integrator) with a syllabic time constant. The output of this integrator is an analogue measure of the average density of the  $N$ -like-digit strings over the syllabic interval, and is used to continu-

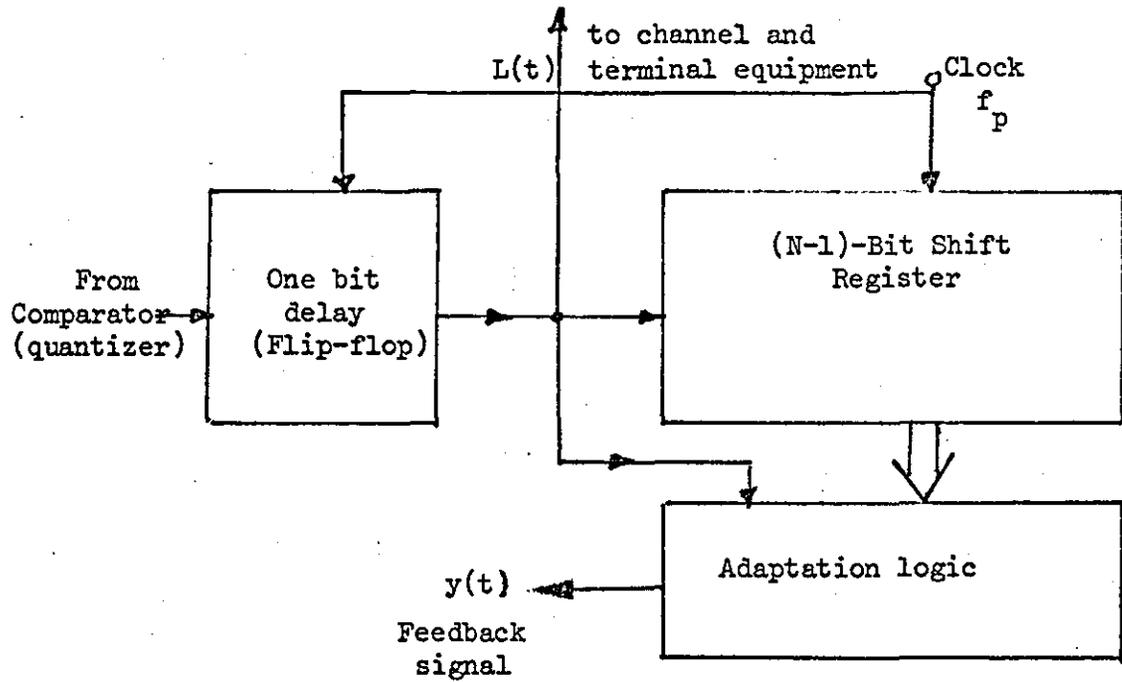


Figure 1.3a Block diagram showing the principle of instantaneous companding in Deltamodulation systems

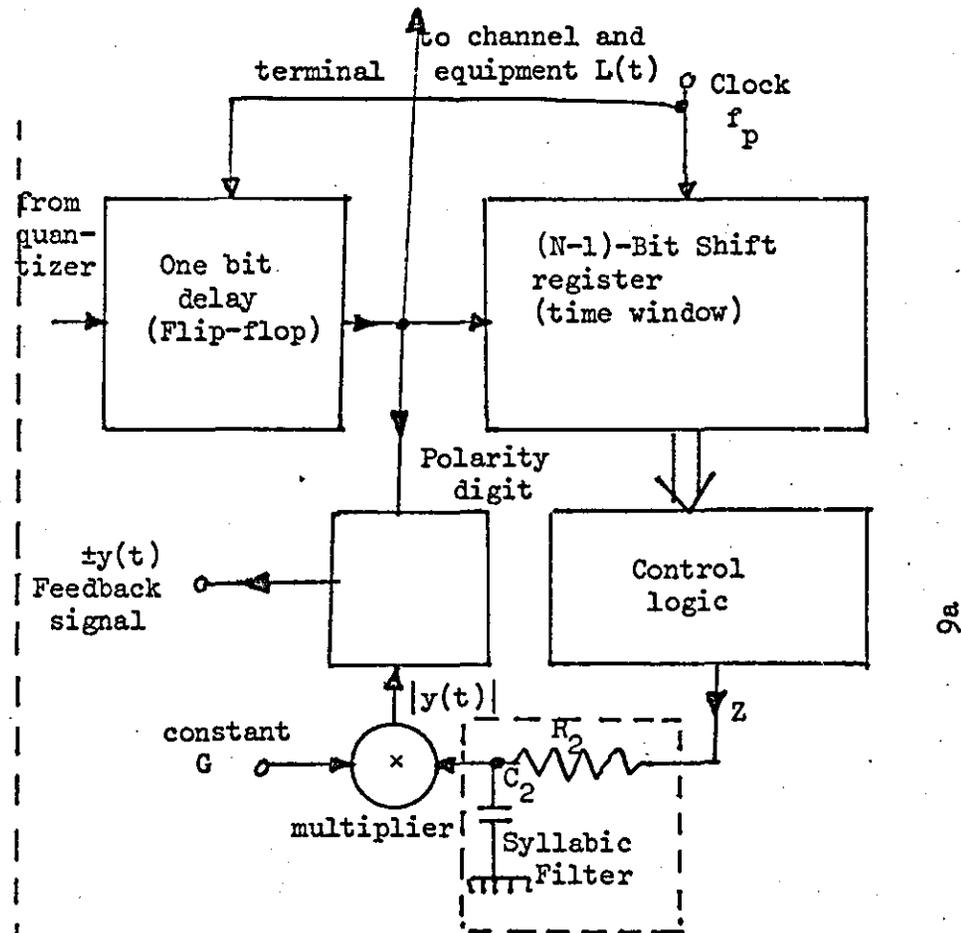


Figure 1.3b Block diagram showing the principle of syllabic companding in Deltamodulation systems

ously control the magnitude of the feedback step heights between two preset extreme limits. In the absence of the N-like-digit strings in the channel, the step size of the feedback signal automatically decays to a minimum (see Fig. 1.3b).

Other attempts to match the companding process to the dynamics of the speech signal have extracted the envelope of the decoded signal, fed this to a power-law element and used the resultant dc voltage to control the magnitude of the feedback step height. Because of its bulky analogue feedback loop (consisting of a bandpass filter, envelope detector and a power-law element) it has not been able to compete with the digitally-controlled versions of syllabically companded systems.

Instantaneous companding achieves rapid adjustment of the step size because it responds to short bit sequences; but this makes it vulnerable to digital transmission errors<sup>(44)</sup>. Channel errors in this case cause the transmitting and the receiving terminals to execute step size changes differently, the result of which is a mis-match between the locally reconstructed signal and that decoded by the remote receiver. This mis-match can persist for a considerable time unless step size information is transmitted to the remote terminal to update compandor tracking. Syllabic companding avoids this complication because it responds more slowly to the average density of the N-like-digit strings in the channel. The average density changes very little even at relatively high error rates. Syllabic time-constants in the range of 5-20 m sec. have been used. In SCALE it is 10 m sec, which is long compared to the fine details of the speech waveform, but short compared to the mean syllabic length (syllabic length varies between 100 and 150 m seconds<sup>(77,78)</sup>).

### 1.3 Statement of the problem

(a) The evolution of the telephone network is directed towards an integrated all-digital transmission and switching system. This ultimately

eliminates the multiple conversions (Analogue-to-digital-to-analogue) within the present network. In such an event, many communication planners - particularly in the Third World countries where PCM is not already in use - argue that there will be no continuing requirement for the 64 Kb/s log-PCM, currently used in America, Europe and Japan; and an alternative simpler and more efficient conversion technique, operating at a reduced bit rate, has to be considered. This argument is further carried on in Chapter VIII.

The efficient companding law, large dynamic range, overload characteristic which is matched to the spectrum from commercial and military microphones and its superiority to A-law PCM at low bit rates (in addition to all the other advantages of one-bit codes) qualify SCALE to be considered as one of the alternative conversion systems in both the present and the expected all-digital communication networks (see Chapter VIII).

Another factor in favour of SCALE is that the other versions of syllabically-companded DM systems - which have wide current and potential use - are closely related to SCALE and with minor modifications they can easily be made compatible with it.

In spite of the potential and current applications of SCALE, a mathematical description of it has been lacking and its performance under various operating conditions has not been fully understood.

(b) At the moment both A-law PCM and syllabically-companded Delta and Delta-sigma modulation systems are used. The coexistence of these two systems has posed a genuine engineering problem, because a conversion technique from one system to the other has not been available, and consequently the users of one system have not been able to communicate directly with those employing the other system.

In Chapters II, III and IV a detailed experimental, theoretical and computer simulation study of the SCALE system is carried out. This leads to the presentation of mathematical formulae for system performance in terms of the bit rate, dynamic range, length of the monitoring window, syllabic time constant and signal to noise ratio.

Chapter V is concerned with replacing the analogue feedback loop in the SCALE encoder with a digital one. A computer-aided design to achieve this is developed and the performance of the new all-digital decoder based on this design is analysed. It is shown by computer simulation that the degradation in the decoded signal quality due to the digitization is negligible. The proposed design procedure is then used to design a practical digital hardware model which employs only commercially available medium scale and small scale integrated circuits.

The aim of Chapter VI is to design a digital converter which can accept the SCALE digital stream and produce A-law PCM words. The chapter begins with a review of the available digital techniques used in the production and the processing of A-law PCM, followed by a review of the present state of the art of digital code conversions. A SCALE-to-A-law PCM converter is then proposed and its performance evaluated by computer simulation. Using integrated circuits which are economically available, a hardware model of the converter is then designed.

Chapter VII contains a computer model and a hardware design of a digital converter which can reverse the process discussed in Chapter VI, i.e. a converter which can accept A-law PCM words and produces SCALE binary data.

The design of these converters aims for a signal which is compatible with that of the standard A-law PCM system. The bit rate in the simulation is therefore made 64 Kb/s.

In Chapter VIII the generality of these proposed conversion techniques and the conclusions reached as a result of this study are discussed and the chapter is concluded with a discussion on the potential applications of converters which can translate PCM codes to syllabically companded Delta and Delta-sigma modulation data, and converters which can reverse the process.

## CHAPTER II

### THEORY OF DELTA-SIGMA MODULATION WITH DIGITALLY-CONTROLLED SYLLABIC COMPANDING

#### 2.1.1 Introduction

In Chapter I it is pointed out that the syllabically companded Delta-sigma Modulation system has many inherent advantages when it is used to digitize pre-emphasized speech waveforms (such as those seen at the output of a telephone or military-type microphones) employing line rates in the order of 20-60 Kb/s. In this chapter, the Syllabically Companded And Logically Encoded Delta-sigma system, known as SCALE, is investigated in detail and its performance is calculated in terms of its circuit parameters and sampling frequency.

The investigation begins with a review of the basic SCALE system proposed by Clarke<sup>(15)</sup> et al. A generalized model of SCALE is then described. This model is first computer simulated and then realized in hardware. The computer program and the hardware experiments will be described in detail and, finally, a mathematical description of the system performance - based on the experimental and simulation results - is presented.

#### 2.1.2 The basic SCALE system

The basic SCALE system which has been first presented by Petford and Clarke<sup>(15)</sup> is shown in Fig. 2.1. The system can be regarded as a single-integration linear Delta Modulation CODEC (Coder-Decoder) with the following modifications:

- (i) The feedback integrator associated with the linear Deltamodulator has been removed from its original position in the feedback path

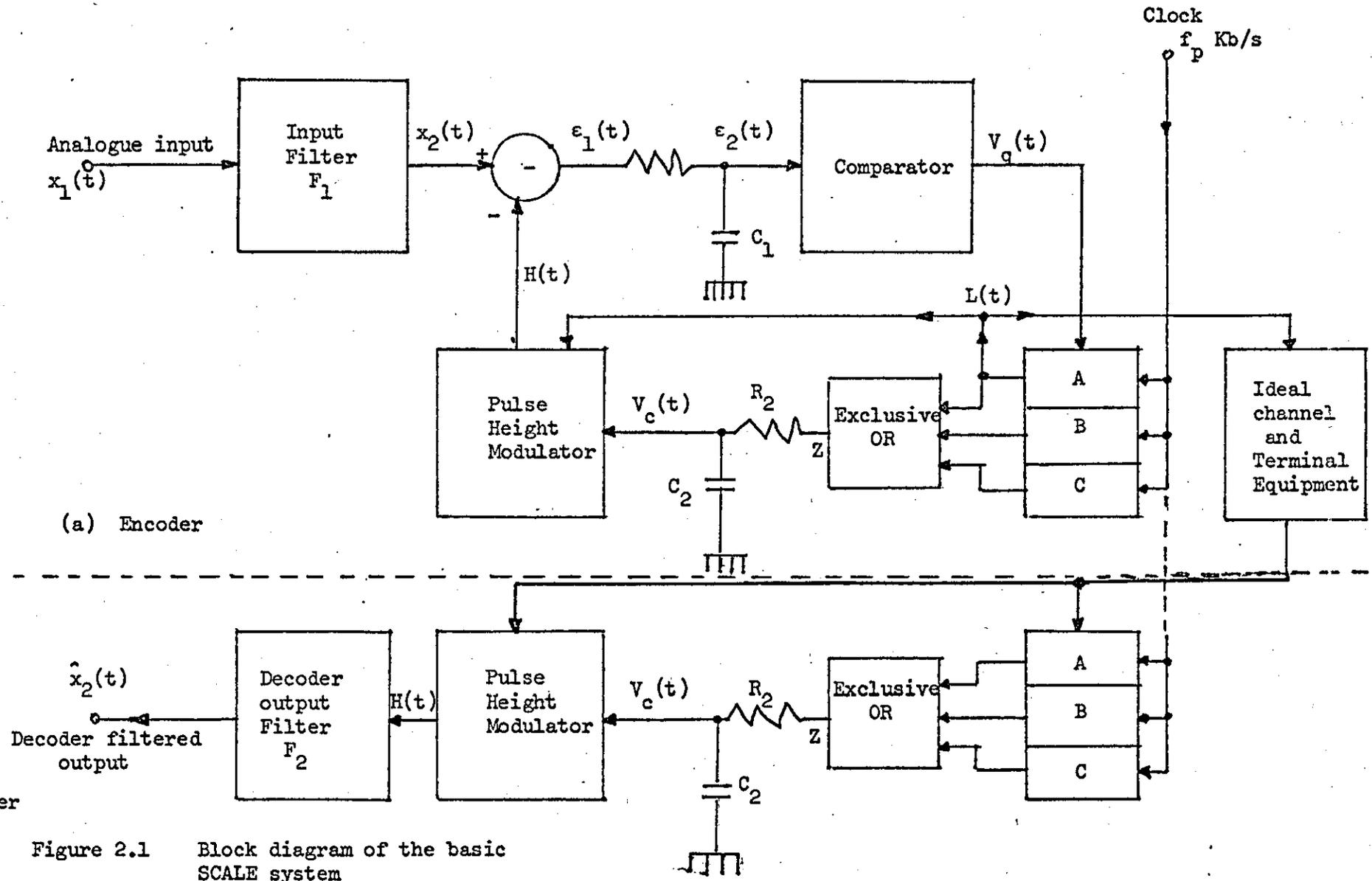


Figure 2.1 Block diagram of the basic SCALE system  
 (a) Encoder (b) Decoder

and reinserted in the forward path, to be shared by both the input and the feedback signals. This has transformed the Delta-modulator into a Delta-sigma system which is more suitable for connection to a pre-emphasised speech source.

- (ii) The feedback loop has been modified in such a manner that it now enables the feedback step size to be varied between two widely separated limits, in response to the input signal variations. This enables the dynamic range of the linear Deltamodulator to be extended to accommodate the expected variations of commercial and military speech signals.

Referring to Fig. 2.1, the analogue input signal  $x_1(t)$  is passed through a low-pass filter  $F_1$  whose lower and upper cut-off frequencies are  $f_{b_1}$  and  $f_{a_1}$  respectively. The output  $x_2(t)$  of  $F_1$  is a band-limited baseband signal which is to be digitized. The feedback signal  $H(t)$  is subtracted from  $x_2(t)$  and the difference signal  $\epsilon_1(t)$  is passed through the forward-path integrator (consisting of  $C_1 R_1$ ) to produce the integrated error signal  $\epsilon_2(t)$ . The output level of the quantizer  $V_q$  is high (5 volts, say) when  $\epsilon_2(t)$  is positive or zero and  $V_q$  is low (0 volt, say) where  $\epsilon_1(t)$  is negative. The output  $L(t)$  of the first stage of the shift register is determined by  $V_q$  every clock period ( $L(t)$  is a logical one if the level of  $V_q$  is high and  $L(t)$  is a logical zero when the level of  $V_q$  is low) and transmitted to channel. The serial shift register SR is clocked at a rate  $f_p$  and it contains the decision levels of the quantizer output during the present and the previous two clock periods. The exclusive OR gate monitors the content (A, B, C) of the SR and produces a voltage pulse  $Z$  which is equal to  $Z_h$  whenever this content is a block of 3 consecutive like digits (i.e. 3 logical ones or 3 logical zeros), and equals to  $Z_l$  otherwise.  $Z$  is passed through an RC integrator, whose time constant  $C_2 R_2$  approximates an average pitch period ( $\approx 10$  m sec.),

to produce a slowly varying signal  $V_c(t)$ . A Pulse Height Modulator (PHM) produces feedback pulses  $H(t)$  the sign and magnitude of which are controlled by  $L(t)$  and  $V_c(t)$  respectively. (The magnitude of  $H(t)$  is controlled by  $V_c(t)$  while the logical level of  $L(t)$  provides the polarity information.) This step height variation of the feedback signal enables it to closely track the input signal  $x_2(t)$  and attempts to minimise the error signals  $\epsilon_1(t)$  and  $\epsilon_2(t)$ . When the input signal is zero the  $L(t)$  sequence is composed of alternate logical ones and zeros viz 010101... (idling pattern) and  $|H(t)|$  decays exponentially towards a predetermined minimum value  $H_{\min}$ . When a large input is applied  $|H(t)|$  rises exponentially towards a predetermined maximum value  $H_{\max}$ . For this condition  $L(t)$  consists of a train of like digits (logical ones if  $x_2(t)$  is positive and logical zeros otherwise). The  $L(t)$  pattern changes only when a change of the polarity of  $\epsilon_2(t)$  occurs. The encoder is said to be tracking whenever 3 or more consecutive digits are observed in the  $L(t)$  pattern and it is said to have reached its absolute overload whenever  $|x_2(t)| \geq H_{\max}$ .

The output of the PHM as a function of the control voltage  $V_c(t)$  and the polarity of the transmitted binary signal  $L(t)$  is depicted in Fig. 2.2. It will be shown later in this chapter that for large dynamic range, the designer should seek to maximize  $H_{\max}$  and minimize  $H_{\min}$ .  $H_{\min}$  is limited by the important requirement that a stable idling pattern must be ensured. To satisfy this requirement  $H_{\min}$  must be made larger than the expected circuit noise voltage and hence avoiding false triggering of the comparator by noise. Practical measurements made on a hardware model of SCALE - which was constructed by Wilkinson<sup>(16,17,18)</sup> using TTL integrated circuits - indicate that the practical limit of  $H_{\min}$  is in the range of 25 to 50 mV.

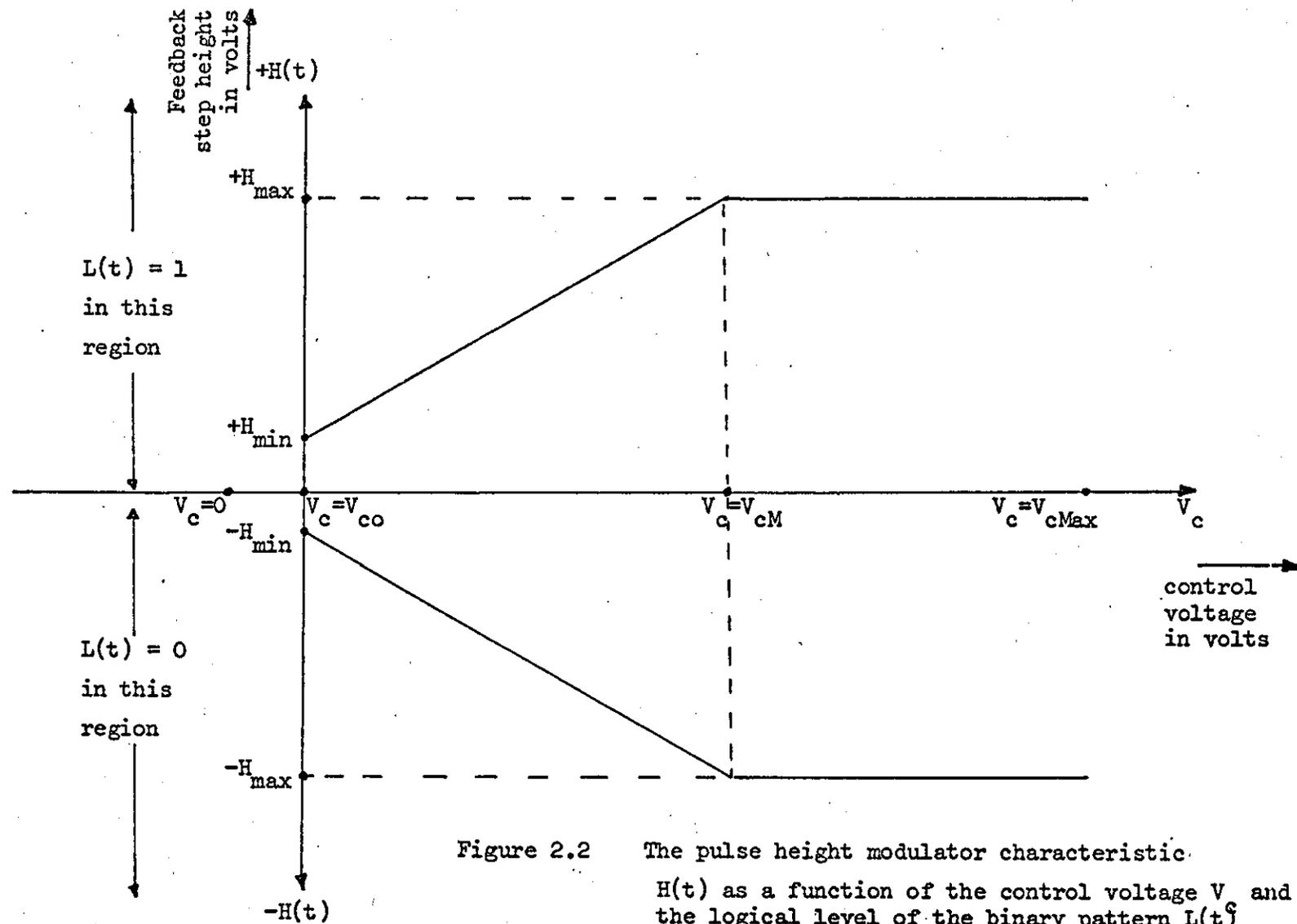


Figure 2.2 The pulse height modulator characteristic.  
 $H(t)$  as a function of the control voltage  $V_c$  and the logical level of the binary pattern  $L(t)$

Since in practice  $H_{\min}$  is a non-zero function (see Fig. 2.2),  $V_{co}$  must also be a non-zero value. The value of  $H_{\max}$  is governed by the maximum expected amplitude of the input signal and its statistical distribution; or by the gain  $G$  of the available operational amplifier employed as a pulse height modulator. Wilkinson has published several reports (16,17,18) on an experimental SCALE system intended for military applications. He suggested that for speech signals  $H_{\max}$  should be equated to the maximum magnitude of the expected waveform ( $|x_2(t)|_{\max}$ ). The system parameters employed by Wilkinson in his experiments are shown in Table 2.1.

At the remote receiver (see Fig. 2.2),  $H(t)$  is reconstructed and passed through a low-pass filter  $F_2$  (with upper and lower cut-off frequencies  $f_a$  and  $f_b$  respectively) which removes the high frequency noise from  $H(t)$  and produces  $\hat{x}_2(t)$  which is a reasonable estimate of  $x_2(t)$ .

### 2.2.1 The generalized model of SCALE

The basic SCALE system described above has employed a fixed shift register length, a fixed syllabic time constant and a fixed sampling frequency. To enable the performance of the SCALE system under various operating conditions to be studied a more general model of it had to be sought. The general model employed in this study was obtained by introducing the following modifications into the basic system:

- (i) The 3-stage shift register SR in the basic system is replaced by another one whose length can be varied between 2 and 6 stages. This arrangement enables the effect of the SR length on the system performance to be examined.
- (ii) The syllabic integrator time constant ( $T_2 = C_2R_2$ ) is made variable (10-20 m sec.) to enable the best  $T_2$ /SR-length combina-

Parameter Designation	Parameter Value	Parameter Description
$T_1 = C_1 R_1$	$2.0 \times 10^{-4}$ second	The SCALE encoder forward path integrator time constant
$T_2 = C_2 R_2$	$10.0 \times 10^{-3}$ second	The syllabic integrator time constant
G	1.4	The gain of the pulse height modulator
$H_{min}$	0.025 volts	The minimum value of the feedback step height
$H_{max}$	2.40 volts	The maximum value of the feedback step height
$Z_l$	0.40 volts	The voltage level of Z - the input to the syllabic integrator - when the contents A, B, C of the shift register are not identical
$Z_h$	3.80 volts	The voltage level of Z when A, B, C are identical
$f_a$	2.4 KHz	The upper cut-off frequency of the input and output filters
$f_b$	250 Hz	The lower cut-off frequency of the input and output filters
$f_p$	19.2 Kb/s	The sampling frequency

Table 2.1 The design parameters of the basic SCALE system employed by Wilkinson

tion to be selected and the effect of  $T_2$  on the system performance to be studied.

(iii) The clock is made external and variable (19-100  $\text{KH}_z$ ). This enables the flexibility of the system (how performance can easily be traded for channel bandwidth and vice versa) to be demonstrated. The modified experimental model of SCALE is shown in Fig. 2.3 and its description follows in the next section.

### 2.2.2 Basic equations and operation of the modified SCALE system

Referring to Fig. 2.3, the input signal  $x_1(t)$  is band-limited by a bandpass filter  $F_1$ , the upper and lower cut-off frequencies of which are respectively  $f_a$  and  $f_b$  and its output is  $x_2(t)$ . The bandwidth of  $x_2(t)$  can, therefore, be written as

$$B = f_a - f_b \quad (2.1)$$

The feedback signal  $H(t)$  is subtracted from  $x_2(t)$  to produce an error signal  $\epsilon_1(t)$ .

$$\epsilon_1(t) = x_2(t) - H(t) \quad (2.2)$$

This is passed through an RC integrator comprising  $C_1R_1$  to produce the integrated error signal  $x_2(t)$ . The time constant of this integrator  $T_1 = C_1R_1$  is selected in such a manner that ensures the matching between the input signal spectrum and the encoder overload characteristic; and at the same time the increase of quantization noise at the decoder output (as a result of the presence of  $C_1R_1$  in the encoder forward path and its absence at the decoder) must be kept to minimum. Details for selecting  $T_1$  can be found in Appendix A, where it is shown that  $T_1$  can be calculated from the following relation:

$$T_1 = \frac{1}{2\pi f_1} = \frac{1}{2\pi \sqrt{\frac{f_a^3 - f_b^3}{30B}}} \quad (2.3)$$

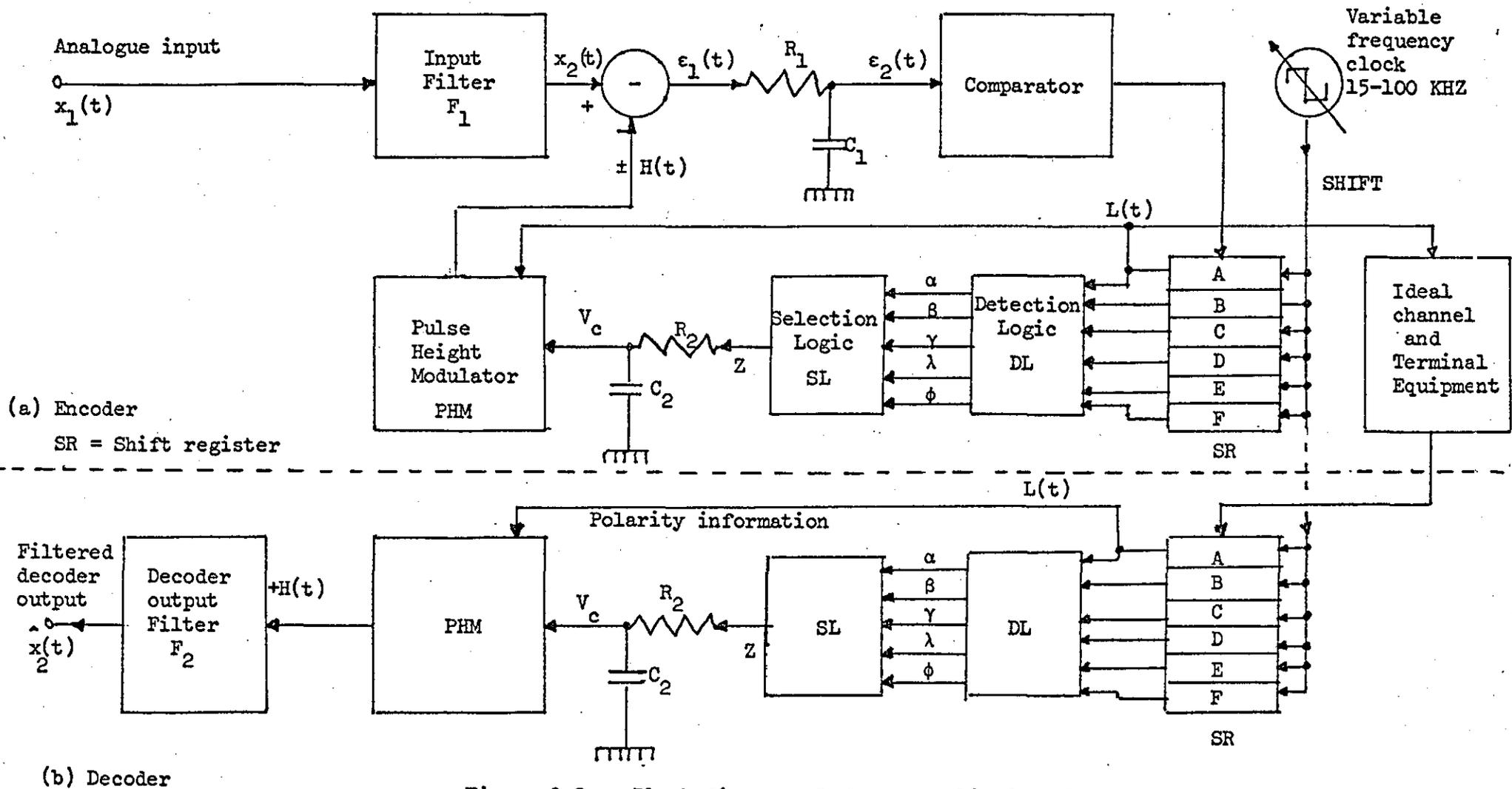


Figure 2.3 Block diagram of the generalized SCALE system

where  $f_1 = 1/2\pi C_1 R_1$ , the cut-off frequency of the  $C_1 R_1$  network.

When the system is first switched on and the input is zero, the channel pattern produced is the idling pattern (01010101) and  $H(t)$  consists of bipolar pulses of fixed amplitude  $\pm H_{\min}$ . In this case the integrated error  $\epsilon_2(t)$  at the end of the  $i^{\text{th}}$  clock pulse can be calculated using the recursive relation:

$$\left. \begin{aligned} (\epsilon_2)_i &= (\epsilon_1)_i(1 - e^{-T_p/T_1}) + (\epsilon_2)_{i-1}e^{-T_p/T_1} \\ &= H_{\min}(1 - e^{-T_p/T_1}) + (\epsilon_2)_{i-1}e^{-T_p/T_1} \end{aligned} \right\} \quad (2.4)$$

where  $(\epsilon_1)_i$  is the value of  $\epsilon_1(t)$  during the  $i^{\text{th}}$  clock period and  $(\epsilon_2)_{i-1}$  is the value of  $\epsilon_2(t)$  at the end of the previous clock period. Because the input to  $C_1 R_1$  is a train of bipolar pulses  $(\epsilon_2)_i = -(\epsilon_2)_{i-1}$  (after transients have died away). Equation (2.4) can therefore be rewritten as:

$$\begin{aligned} (\epsilon_2)_i &= H_{\min} \frac{(1 - e^{-T_p/T_1})}{(1 + e^{-T_p/T_1})} = H_{\min} \text{Tanh}(T_p/2T_1) \\ &= (\pi f_1/f_p) H_{\min} \end{aligned} \quad (2.5)$$

( $T_p/2T_1$  is assumed to be small.)

This is the minimum value of the magnitude of the integrated error signal seen at the input of the quantizer.

The waveforms of  $\epsilon_1(t)$  and  $\epsilon_2(t)$  during the idling condition are shown in Fig. 2.4. It can be seen that the minimum integrated error waveform is triangular in shape with peak-to-peak amplitude  $d$  given by

$$\begin{aligned} d &= 2H_{\min} \text{Tanh}(T_p/2T_1) \\ &\approx (2\pi f_1/f_p) H_{\min} \end{aligned} \quad (2.6)$$

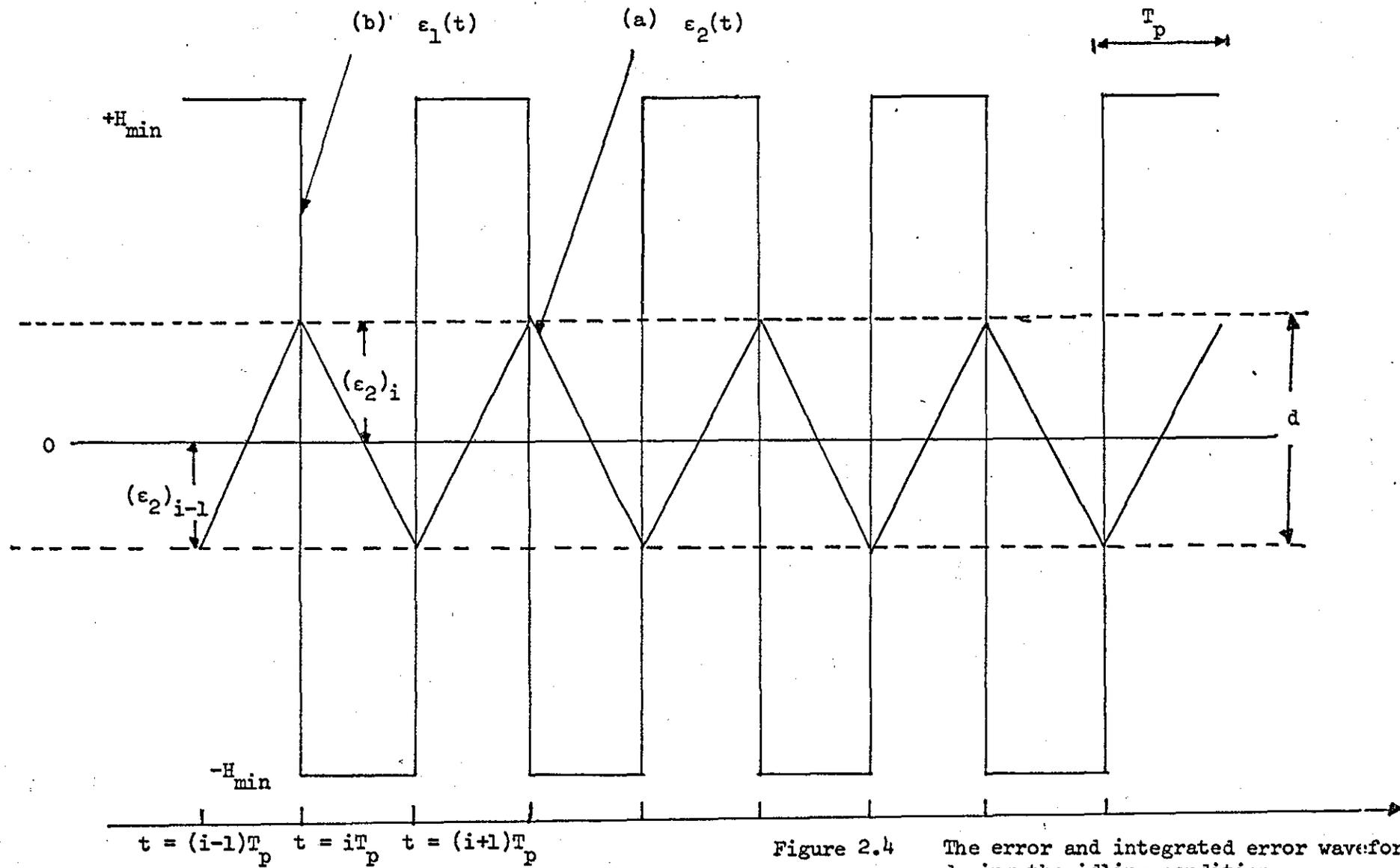


Figure 2.4 The error and integrated error waveforms during the idling condition

Now if we let  $Q_i$  represent the output of the quantizer  $V_q$ , during the  $i^{\text{th}}$  clock period; then  $Q_i$  will be a logical one or a logical zero depending on whether the polarity of  $\epsilon_2(t)$  during the same clock period is positive or negative, respectively.

$$Q_i = \begin{cases} 1 & (\epsilon_2)_i \geq 0 \\ 0 & (\epsilon_2)_i < 0 \end{cases} \quad (2.7)$$

This feeds the 6-stage serial shift register which is clocked at the required bit rate  $f_p$ . The content of SR is continuously monitored by the Detection logic DL, the function of which is to detect the presence of 2,3,4,...6 consecutive like digits stored in the SR. The outputs of the DL are

$$\left. \begin{aligned} \alpha &= A.B + \bar{A}.\bar{B} \\ \beta &= A.B.C + \bar{A}.\bar{B}.\bar{C} \\ \gamma &= A.B.C.D + \bar{A}.\bar{B}.\bar{C}.\bar{D} \\ \lambda &= A.B.C.D.E + \bar{A}.\bar{B}.\bar{C}.\bar{D}.\bar{E} \\ \phi &= A.B.C.D.E.F + \bar{A}.\bar{B}.\bar{C}.\bar{D}.\bar{E}.\bar{F} \end{aligned} \right\} \quad (2.8)$$

The selection logic SL is preset to select any one of the Boolean functions ( $\alpha, \beta, \dots$  or  $\phi$ ) and produces a voltage pulse  $Z$  which equals  $Z_h$  when the function (e.g.  $\beta$ ) is TRUE and equals to  $Z_l$  when the function is FALSE.

e.g. if SL is preset to select  $\beta$  then

$$Z = \begin{cases} Z_h, & \beta \text{ TRUE} \\ Z_l, & \beta \text{ FALSE} \end{cases} \quad (2.9)$$

The ability to select any one of the functions ( $\alpha, \beta, \dots, \phi$ ) to control the magnitude of  $Z$  means that we have a variable length SR, the effective

length  $N_e$  of which is set by SL.

The relationship between Z and the output of the quantizer  $Q_i$  can be visualized by constructing a state table or/and state diagram.

Fig. 2.5 demonstrates these state tables and state diagrams for the cases when SL is preset (i) to select  $\alpha$  and (ii) to select  $\beta$ .

The state of the output of the comparator is transmitted to the channel via the first stage of the SR. The transmitted signal  $L(t)$  is composed of binary pulses carrying information about the magnitude of the input signal. During the  $i^{\text{th}}$  clock period the value of  $L(t)$  is

$$L_i = Q_{i-1} = \{0,1\} \quad (2.10)$$

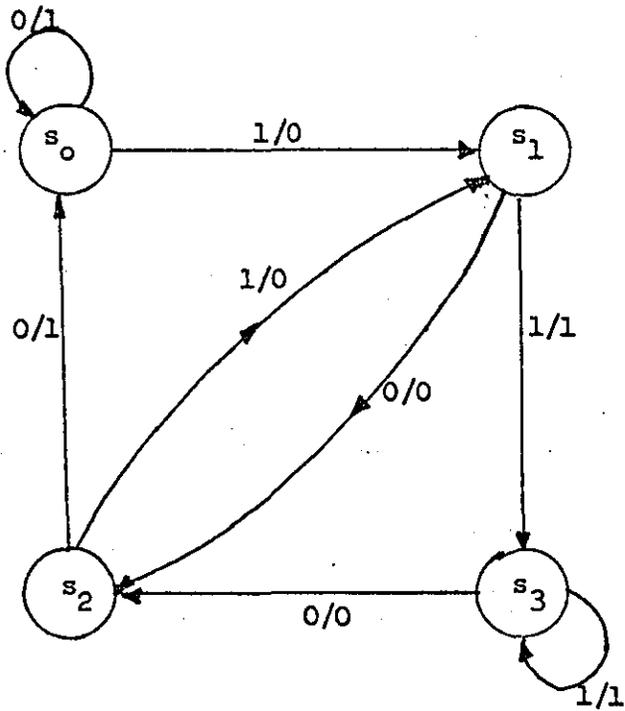
When TTL integrated circuits are used the higher voltage levels of  $L(t)$ ,  $V_q(t)$  and  $Z_h$  are in the range 3.5 - 5 and the lower voltage levels are in the range 0.2 - 0.5 volts.

Now if a constant dc voltage of magnitude  $|x_2(t)|$ ; ( $H_{\min} \leq |x_2(t)| < H_{\max}$ , say) is applied to the input of the encoder, at the moment of switching on, the digital pattern observed in the channel will constitute a block of consecutive like digits, the length of which is increasing until the feedback  $|H(t)|$  is equal to  $|x_2(t)|$ , a point at which a change of polarity takes place. If the time between applying the input and observing a change in polarity is  $mT_p$ , say, then the time during which  $Z = Z_h$  is  $nT_p$  ( $n$  is an integer constant). During this time the control voltage  $V_c(t)$  at the output of the syllabic integrator (the input of which is Z) rises to

$$V_c(mT_p) = Z_h(1 - e^{-nT_p/T_2}) + V_{co} \quad (2.11)$$

$$= V_{CM} \quad (\text{see Fig. 2.6})$$

where  $n = (m - N_e + 1)$ ,  $T_2 = C_2R_2$  and  $V_{co}$  is the minimum value of  $V_c(t)$ .



(a) State diagram

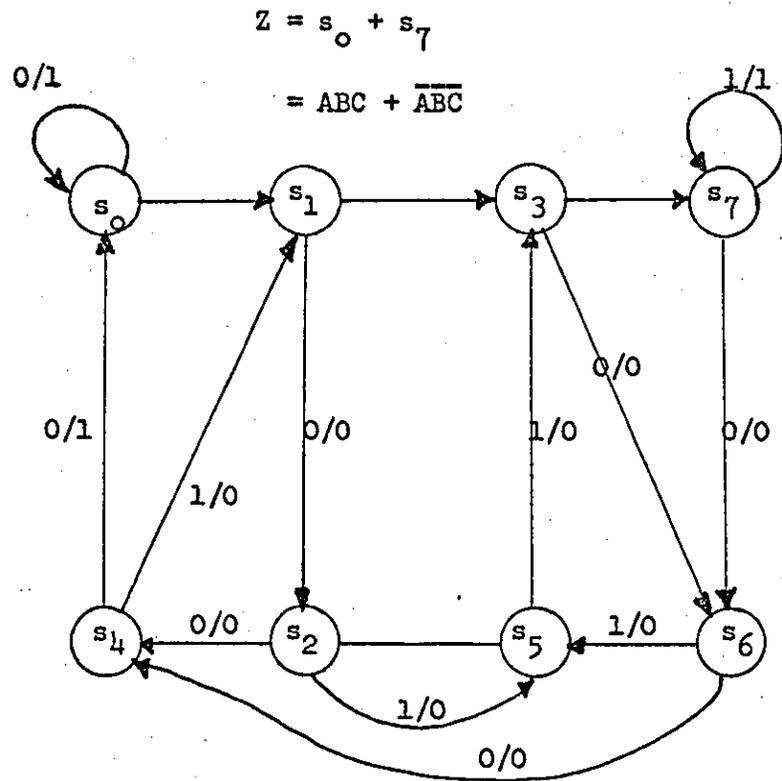
The notation  $Q_i/Z$  is used

$Z_h = \text{logical } 1$ ,  $Z_l = \text{logical } 0$

Present state		Next state		Z	
		$Q_i = 0$	$Q_i = 1$	$Q_i = 0$	$Q_i = 1$
A	B				
0	0	$s_0$	$s_1$	$Z_h$	$Z_l$
1	0	$s_1$	$s_3$	$Z_l$	$Z_l$
0	1	$s_2$	$s_1$	$Z_h$	$Z_l$
1	1	$s_3$	$s_3$	$Z_l$	$Z_h$

(b) State table

Figure 2.5(i) State table and state diagram for the effective shift register length  $N_e = 2$



(a) State diagram

The notation  $Q_i/Z$  is used

$Z_l$  - logical 0,  $Z_h$  = logical 1

Present state		Next state		Z	
		$Q_i = 0$	$Q_i = 1$	$Q_i = 0$	$Q_i = 1$
A	B C				
0	0 0	$s_0$	$s_1$	$Z_h$	$Z_l$
1	0 0	$s_1$	$s_2$	$Z_l$	$Z_l$
0	1 0	$s_2$	$s_4$	$Z_l$	$Z_l$
1	1 0	$s_3$	$s_6$	$Z_l$	$Z_l$
0	0 1	$s_4$	$s_0$	$Z_h$	$Z_l$
1	0 1	$s_5$	$s_2$	$Z_l$	$Z_l$
0	1 1	$s_6$	$s_4$	$Z_l$	$Z_l$
1	1 1	$s_7$	$s_6$	$Z_l$	$Z_h$

(b) State table

Figure 2.5(ii) State diagram and state table for  $N_e$  (the effective shift register length) = 3

Equation (2.11) is graphed in Fig. 2.6.

In practice the input signal is most likely to be a speech waveform, the magnitude, sign and phase of which vary with time. In order to match the feedback step height to the level of the input signal, the average of the latter must be continuously evaluated and  $|H(t)|$  is adjusted in such a manner that  $|\epsilon_2(t)|$  is minimised. The control voltage  $V_c(t)$ , in this case, is still an exponential function of the group length (group length is the length of the group of consecutive like digits generated in the time period during which  $Z = Z_h$ ). In other words,  $V_c(t)$  is obtained by transforming the periods of time during which  $Z$  is high, to a voltage at the output of the syllabic integrator  $C_2R_2$ . When the input is not a constant voltage, however, the growth of  $V_c(t)$  is disrupted for  $(N_e - 1)T_p$  seconds after every change of polarity in the  $L(t)$  pattern. During this period the growth of  $V_c(t)$  is halted and it decays exponentially until  $Z$  becomes  $Z_l$  (when  $V_c(t)$  starts to rise again) or the minimum value of  $V_c(t) = V_{co}$  is reached.

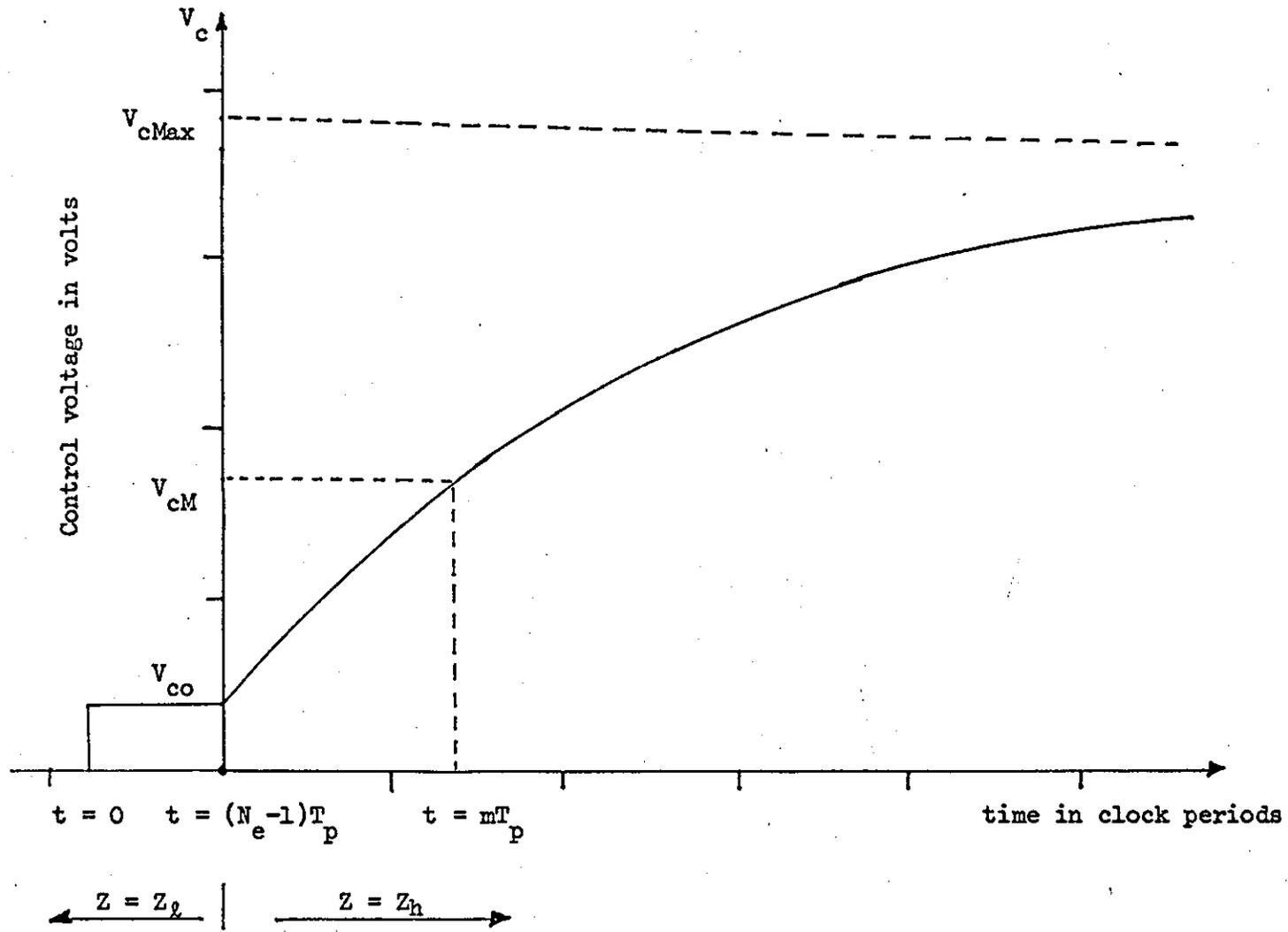
Now since the time during which  $Z = Z_h$  is a measure of the power residing in the input signal, then a relationship between the r.m.s. (root mean square) value  $\sigma_x$ , say, of the input signal and  $V_c(t)$  must exist. An attempt to define this relationship was made as follows:

- (i) The r.m.s. value of the input signal  $\sigma_x$  over a period of time  $T_2$ , which is approximately an average pitch period, is calculated.
- (ii) With the aid of an unknown constant  $\delta$ , say, and the property of the circuit  $C_2R_2$ , a relationship between the control voltage  $V_{cp}$  (the value of  $V_c(t)$  at the end of the calculation period  $T_2$ ) and  $\sigma_x$  is formulated:

$$V_{cp} = Z_h(1 - e^{-\delta\sigma_x}) + V_{co} \quad (2.12)$$

$\delta$  is a constant which has been introduced here <sup>(19)</sup> to enable

Figure 2.6 Syllabic integrator output voltage  $V_c$  as a function of the group length (time during which  $Z = Z_h$ )



equation (2.12) to be obtained. Its value can be found experimentally or by computer simulation. It will be shown later in this chapter that equation (2.12) is valuable in calculating the performance of the SCALE system in terms of its circuit parameters.

Another approach for defining  $V_c(t)$  makes use of the recursive relationship (50):-

$$(V_c)_i = Z(1 - e^{-T_p/T_2}) + (V_c)_{(i-1)}e^{-T_p/T_2} \quad (2.13)$$

where  $(V_c)_i$  is the value of  $V_c(t)$  at the end of the present clock period and  $(V_c)_{i-1}$  is the value of  $V_c(t)$  at the end of the previous clock period.  $Z$  is one of two values  $Z_h$  or  $Z_l$ .

This approach is valuable in the design of a digital feedback loop and a digital decoder for the SCALE system to replace the analogue ones (see Chapter V).

The pulse height modulator PHM used in this system has a transfer function as shown in Fig. 2.2. Its output  $H(t)$  is  $H_{\max}$  (absolute overload) if  $V_c(t) \geq V_{cm}$ , and is equal to  $H_{\min}$  if  $V_c(t) = V_{co}$ . In the  $V_c(t)$  range,  $V_{co} < V_c(t) \leq V_{cm}$ , the PHM acts as a multiplier, the output of which is  $GV_c(t)$ , where  $G$  is a constant given by

$$G = \frac{H_{\max} - H_{\min}}{V_{cm} - V_{co}} \quad (2.14)$$

The magnitude of the feedback step height can be calculated using (2.14), and either equation (2.12) or equation (2.13). The first produces the magnitude  $|H(t)|$  constructed during approximately one pitch period and signified by  $H_p$ , while equation (2.13) leads to the determination of  $|H(t)|$  in terms of the present and previous ( $T_p$  ago) values of  $V_c(t)$ .

$$H_p = GV_{cp} = G Z_h (1 - e^{-\delta\sigma x}) + H_{\min} \quad (2.15)$$

$$H_i = G(V_c)_i = G \left\{ Z(1 - e^{-T_p/T_2}) + (V_c)_{i-1} e^{-T_p/T_2} \right\} \quad (2.16)$$

The value of  $H(t)$  which is subtracted from  $x_2(t)$  to produce  $\epsilon_1(t)$  is

$$H(t) = PH(t) \quad (2.17)$$

where  $P$  is given by

$$P = \text{Sgn}\{\epsilon_2(t - T_p)\} \quad (2.18)$$

and  $T_p = 1/f_p$

At the remote decoder (assuming an ideal channel)  $H(t)$  is reconstructed as in the local decoder (feedback loop). The noise associated with  $H(t)$  is then removed by the decoder output filter  $F_2$ . The output of this filter is  $\hat{x}_2(t)$  and is an estimate of the input signal  $x_2(t)$ .

### 2.2.3 Calculation of the system performance

The judgement of the performance of digital systems can be based on the results of subjective tests<sup>(18)</sup> or on experimentally measurable figures of merit such as signal-to-noise ratio SNR, dynamic range DR and error performance. Here the objective approach is adopted.

#### (a) Signal-to-noise ratio

Quantization noise in the linear Deltamodulation system was first studied by de Jager<sup>(8)</sup>, whose theory was later advanced by Johnson<sup>(49)</sup> to cover Delta-sigma as well as the linear basic Deltamodulation system. The theory is based on the following assumptions (see Appendix A):-

- (i) The noise waveform at the input of the quantizer is triangular with peak-to-peak amplitude =  $d$  and a power spectral distribution

of the form  $(\sin \theta/\theta)^2$ , (see Appendix A). The first null of such distribution is at  $f_p$ .

- (ii) The inherent pre-emphasis present at the transmitter end and the absence of a de-emphasis circuit in the linear (and compressed) Delta-sigma systems causes the noise in these systems to be frequency dependent (rises with increase in frequency).
- (iii) The minimum noise power per  $H_z$  is proportional to the central delta step  $d$ .

These assumptions have led Johnson to calculate the noise power per  $H_z$  for Delta-sigma systems as

$$|\psi|^2 = K \frac{d^2(f_1^2 + f^2)}{f_p f_1^2} \quad (2.19)$$

where  $f$  is any frequency and  $K$  is a constant of proportionality.

Now in the absence of an input signal, which is large and slow enough to produce  $N_e$  consecutive like digits in the  $L(t)$  pattern, SCALE acts as a simple Delta-sigma encoder with a feedback step size  $H_{\min}$  and equation (2.19) is applicable. Substituting for  $d$  from equation (2.6) we can write

$$\begin{aligned} |\psi|^2 &= K \frac{4H_{\min}^2}{f_p} \left\{ \tanh^2(T_p/2T_1) \right\} \left( \frac{f_1^2 + f^2}{f_1^2} \right) \\ &= \frac{4\pi^2 K H_{\min}^2}{f_p^3} (f_1^2 + f^2) \end{aligned} \quad (2.20)$$

The total noise power passed by the decoder output filter  $F_2$  is given by:

$$N_q^2 = 2 \int_{f_b}^a |\psi|^2 df = \frac{8\pi^2 K H_{\min}^2}{f_p^3} \int_{f_b}^a (f_1^2 + f^2) df$$

$$= \frac{8\pi^2 K H_{\min}^2}{f_p^3} \left( f_{1B}^2 + \frac{f_a^3 - f_b^3}{3} \right) \quad (2.21)$$

where B is defined by equation (2.1).

The signal to noise power ratio can be obtained by dividing the total signal power, calculated over a period of time which is approximately one pitch period, by the total noise power calculated over the same period of time.

Now the total signal power in this case is  $\sigma_x^2$  and the total noise power can be obtained from equation (2.21) by replacing K and  $H_{\min}$  respectively by  $K_c$  and  $H_p$ . This leads to:

$$\text{SNR} = \frac{\left( \frac{\sigma_x^2}{K_c H_p^2} \right)}{\left( \frac{3f_p^3}{8\pi^2 (3f_{1B}^2 + f_a^3 - f_b^3)} \right)} \quad (2.22)$$

A formula similar to (2.22) has been presented by Cartmale and Steele<sup>(35)</sup> for GEC MKI companded Delta-sigma systems with sinusoidal input. The difference between the two formulae lies in the meaning of  $H_p$ ,  $\sigma_x$  and the value of  $K_c$ . This difference is a direct result of using an input signal which has a Gaussian probability distribution, while Cartmale and Steele have employed a sinusoidal input signal.

Overload occurs either when  $|H(t)| = H_{\max}$ , i.e. when the PHM becomes absolutely saturated, or when the input signal rate of change is so fast that  $V_c(t)$  (because of the nature of the  $C_2R_2$  circuit) is not capable of tracking correctly. In this case equation (2.22) still holds. Substituting for  $H_p$  from equation (2.15) equation (2.22)

becomes:

$$\text{SNR} = R_c \left( \frac{\sigma_x}{GZ_h (1 - e^{-\delta\sigma_x}) + H_{\min}} \right)^2 \quad \sigma_x \leq H_p \quad (2.23a)$$

$$= R_c (\sigma_x / H_p)^2 \quad \sigma_x \leq H_p \quad (2.23b)$$

where

$$R_c = \frac{3f_p^3}{8\pi^2 K_c (3f_1^2 B + f_a^3 - f_b^3)} \quad (2.24)$$

In the area adjacent to  $\sigma_x = H_p$  the rate of growth of  $H_p$  becomes slower than that of  $\sigma_x$  and consequently the SNR starts to degrade. This continues until absolute overload is reached.

Above absolute overload  $H_p$  becomes a constant even if  $\sigma_x$  continues to increase. In this case it is clear that the larger the value of  $H_p$  the smaller the error signal and vice versa for  $\sigma_x$  (i.e. the larger  $\sigma_x$  the larger the error signal). This suggests that when  $\sigma_x > H_p$  equation (2.23) should be inverted.

$$\text{SNR} = R_c \left( \frac{G(1 - e^{-\delta\sigma_x}) + H_{\min}}{\sigma_x} \right)^2 \quad \sigma_x > H_p \quad (2.25a)$$

$$= R_c \left( \frac{H_p}{\sigma_x} \right)^2 \quad (2.25b)$$

Although this approach might seem somewhat approximate, it will be shown, by practical measurements and computer simulation, later in this chapter, that it is accurate enough for most practical purposes.

(b) Dynamic range

The minimum acceptable signal-to-noise ratio depends on the field of application (military, commercial, etc.) and is to a large extent subjective. In some military applications, for example, signals which are just intelligible to a trained operator (7-10 dB) are considered

to be adequate.

Let the minimum acceptable SNR be  $\rho$  dB and the corresponding required r.m.s. input voltage be  $\sigma_m$ . Then from (2.23)

$$\sigma_m (R/\rho)^{\frac{1}{2}} = GZ_h (1 - e^{-\delta \sigma_m}) + H_{\min} \quad (2.26)$$

$$= GZ_h \left( \delta \sigma_m - \frac{\delta^2 \sigma_m^2}{2!} + \dots \right) + H_{\min} \quad (2.27)$$

It will be shown in Chapter III that  $\delta$  is a small fraction, so if  $\delta \sigma_m$  is such that the 2<sup>nd</sup> and higher terms in (2.27) can be neglected then:

$$\sigma_m = \frac{H_{\min}}{(R_c/\rho)^{\frac{1}{2}} - \delta GZ_h} \quad (2.28)$$

Now equation (2.23) reduces to  $\text{SNR} = R_c$  when  $H_p^2 = \sigma_x^2$ . It has been suggested<sup>(35)</sup> that the system should be adjusted in such a manner that this condition is true when  $\sigma_x = \sigma_M = H_p = H_{\max}$ , where  $\sigma_M$  denotes the value of the input r.m.s. voltage which is just sufficient to initiate absolute overload.

Multiplying both sides of equation (2.28) by  $1/H_{\max}$  ( $= 1/\sigma_M$ ) and performing the necessary manipulations the dynamic range - with companding - can be written as

$$\text{DR}_c = 20 \log_{10} \frac{H_{\max}}{H_{\min}} \left\{ (R_c/\rho)^{\frac{1}{2}} - \delta GZ_h \right\} \quad (2.29)$$

### (c) Error performance

The effect of transmission errors in Delta and Delta-sigma modulation systems has been studied by Johnson<sup>(49)</sup> who shows that for

uncompanded Delta-sigma systems the noise power due to transmission errors is

$$N_{eu}^2 = \frac{8P_{10}V_L^2B}{f_p} \quad (2.30)$$

where  $P_{10}$  is the probability that a digit is transmitted as a logical one and received as a logical zero and vice versa (i.e. the probability of making a wrong decision at the receiver).  $V_L$  is the voltage level associated with  $L(t)$ .

Cartmale and Steele<sup>(35)</sup> have suggested that in companded Delta-modulation systems  $V_L$  in equation (2.30) should be replaced by the variable feedback signal. Following this suggestion, the transmission noise power in SCALE can be written as

$$N_{ec}^2 = \frac{8H_p^2P_{10}B}{f_p} \quad (2.31)$$

From equations (2.23b) and (2.31) the total of the quantization and transmission noise powers (assuming that  $N_{ec}$  and  $N_q$  are statistically independent) is

$$\begin{aligned} N_T^2 &= N_{ec}^2 + N_q^2 \\ &= H_p^2 \left\{ \frac{1}{R_c} + \frac{8P_{10}B}{f_p} \right\} \end{aligned} \quad (2.32)$$

The resultant signal-to-noise ratio in this case is

$$\begin{aligned} \text{SNR}_e &= \frac{\sigma_x^2}{N_T^2} \\ &= \eta_c(\text{SNR}) \end{aligned} \quad (2.33)$$

where 
$$\eta_c = 1/1 + (8R_c P_{10} B/f_p) \quad (2.34)$$

and SNR is that of the SCALE system in the absence of channel errors.

This last equation indicates that the designer of SCALE systems should aim for as large  $f_p$  and as small  $B$  as possible. It however must be remembered that  $R_c$  increases when  $f_p$  is increased or  $B$  is decreased (see equation (2.24)). This means that  $(R_c B/f_p)$  must be chosen in such a manner that its overall value is minimized.

To further illustrate the meaning of equation (2.34) let the probability of making a wrong decision be  $10^{-3}$ . Then a SCALE system in which  $f_p = 19.2$  Kb/s,  $B = 2.4$  KHz and  $R_c = 31$  (i.e. maximum SNR, in absence of channel errors, of 15 dB) the value of  $\eta_c$  is 0.97. This means that even at a sampling frequency as low as 19.2 Kb/s an error rate of 1 in a 1000 will not degrade the SNR by more than 3%. Hence it can be concluded that the SCALE system can tolerate a greater number of errors than other digital systems such as companded PCM and adaptive DPCM.

COMPUTER SIMULATION OF THE S C A L E SYSTEM3.1 INTRODUCTION

Simulation (or modeling) is a combination of techniques employed to solve a problem via the computer; or use the computer to act as, or behave like, the real system or circuit.

There are three basic levels of simulation - system, functional and circuit. Simulation at circuit level produces the best result but needs a long computer time (therefore not economical). The analysis task in this case is also complex and time consuming.

Simulation at the functional level means that only the terminal characteristics of subunits of a system are simulated. A coarser representation of a system where two or more subunits are combined into one larger subunit before modeling is known as system level simulation.

The functional and system simulation techniques are based on the assumption that the internal nodes are of no particular significance as long as their effects are taken implicitly into account at the terminals. This is in practice not true because some of the properties - especially the less important ones - tend to be neglected by the programmer who models at the functional and system levels. For this reason these two levels of simulation techniques, in the strict sense, can only produce an approximate model of the real hardware, even when they are not intentionally designed that way.

An entirely circuit level simulation of the SCALE system was ruled out from the early stages of the modeling. The representation adopted is in terms of subunits such as the input filter, the output filter, the encoder, the channel, the decoder, etc. This partitioning of a system is important only insofar as one of these subunits can be

analysed (by partitioning it to lower functional subunits - quantizer, pulse height modulator, etc.) at the functional or circuit level independently of other parts of the system.

After the initial partition of the SCALE system using the above technique into quasi-independent subunits, functional level simulation is used exclusively except when it is absolutely necessary to employ circuit level modeling (such as in the calculation of the control and integrated error voltages  $V_c(t)$  and  $\epsilon_2(t)$ ).

After deciding on the level of simulation the four major phases of the modeling procedure begin. These are

- (i) The program analysis .
- (ii) Coding
- (iii) Verification and debugging (finding the errors)
- (iv) Using the model for what it is intended for

The program analysis is the procedure whereby the problem is reduced conceptually to a form in which a computer program can be written to solve it. The first step in the analysis is an original statement of the problem. The next step is conceiving a set of rules that the computer can follow. These rules are called the algorithm and must utilize the basic computational capabilities of the digital computer. Once the algorithm has been conceived, it is usually formally stated either in algebraic equations, or logically by means of symbols and a flow chart. The purpose of this formalization is to present the algorithm in step-by-step manner in order to simplify the details fo the coding. The basic equations for SCALE which form the algorithm for this simulation have already been stated in Chapter II.

Coding in this context is the process of instructing the computer to read in and execute the program and produce the results. This can be written either in a low level language, such as machine or assembly

language; or in a high level language such as BASIC, ALGOL, FORTRAN<sup>(64,65,66)</sup> or any other special purpose high level language. In this simulation low level languages were avoided because of their machine dependence, and difficulty of debugging. FORTRAN was chosen because it is relatively free from the above limitations and in addition it uses ordinary algebraic notation and English words. Fortran is not as efficient as low level languages but its simplicity and generality made it the most popular high level language.

Debugging is the process of removing syntactic and logical faults from the written program and the varification stage is that during which the program is being tested to ensure that there are no logical fallacies which might have been introduced at the conception of the algorithm, its formalization, or its coding.

Finally, when the model is ready for experimentation, various observations and measurements under various input and operating conditions are carried out to determine the performance of the CODEC and test the validity of the theoretical analysis.

### 3.2.1 Statement of the objective

The objective is to use ICL 1904 digital computer to implement and study the general model of the SCALE system shown in Fig. 2.2 and its associated filters. Extended Fortran IV language is to be employed.

### 3.2.2 Partitioning of the problem

The system is partitioned into 8 quasi-independent subunits as shown in Fig. 3.1. An algorithm for each subunit is then presented in the form of algebraic equations or flow chart and then translated into FORTRAN IV under the heading of "SUBROUTINE *name (argument list)*".

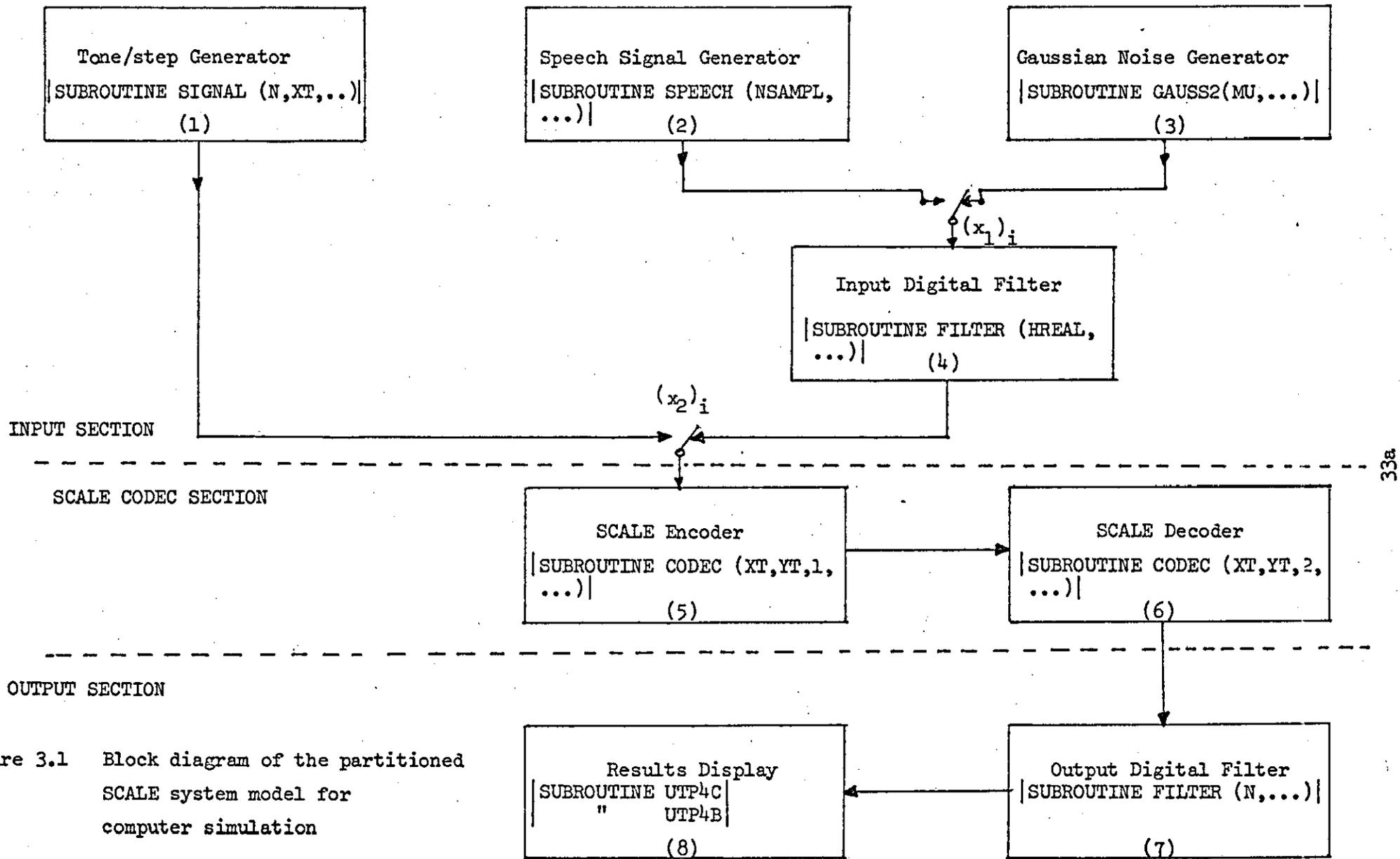


Figure 3.1 Block diagram of the partitioned SCALE system model for computer simulation

### 3.2.3 Program analysis and coding

#### (i) Simulation of the input signal

Referring to Fig. 3.1, the function of the top three subunits is to generate signals to simulate the input waveforms. An input sample  $(x_1)_i$  is supplied by one of these subunits during the  $i^{\text{th}}$  clock period. The waveform generated depends on the subunit selected by the external switches (which call a particular subroutine) and on the internal switches (conditional jumps within the subroutine).

(a) Subunit 1: This subunit can generate steps of assignable magnitude of the form

$$(x_1)_i = A_i(j) \quad \begin{array}{l} i = 1, 2, 3, \dots, 512 \\ j = 1, 2, \dots, 6 \end{array} \quad (3.1)$$

or sinusoides of assignable frequency  $f_s$  and magnitude  $A_i(j)$  of the form

$$(x_1)_i = A_i(j) \sin(2\pi i f_s(r)/f_p) \quad (3.2)$$
$$\begin{array}{l} i = 1, 2, \dots, 512 \\ j = 1, 2, \dots, 6 \\ r = 1, 2, \dots, 6 \end{array}$$

$i$  defines the position of the sample in time,  $j$  defines the samples amplitude and  $r$  defines the frequency of the input tone. The selection of the required input signal is achieved through the setting of the subroutine arguments in the CALL statement.

A flow chart, together with the FORTRAN subroutine to implement this subunit (SUBROUTINE SIGNAL (N, XT, JF, JA, JB)) are given and explained in Appendix B and some of the signals generated inside the computer by this subroutine are shown in Figs. 3.2(a) and (b).

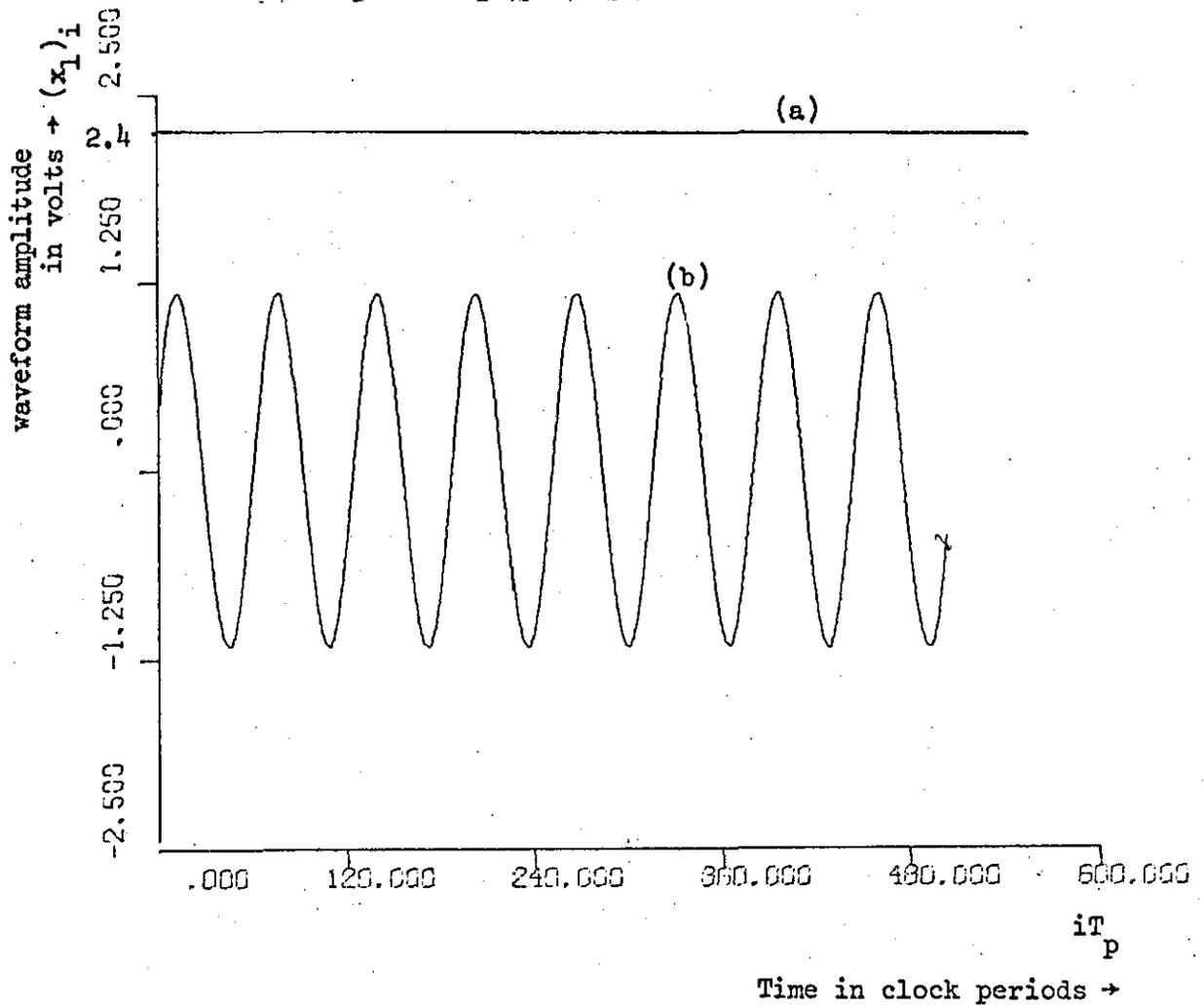


Figure 3.2 Examples of the signal waveforms generated by "SUBROUTINE SIGNAL" to serve as input to the simulated model of the SCALE system

- curve (a) step function of amplitude 2.4 volts
- curve (b) sinusoid of amplitude  $A(j)=1.2$  volts  
and frequency  $f_s = 1000$  HZ

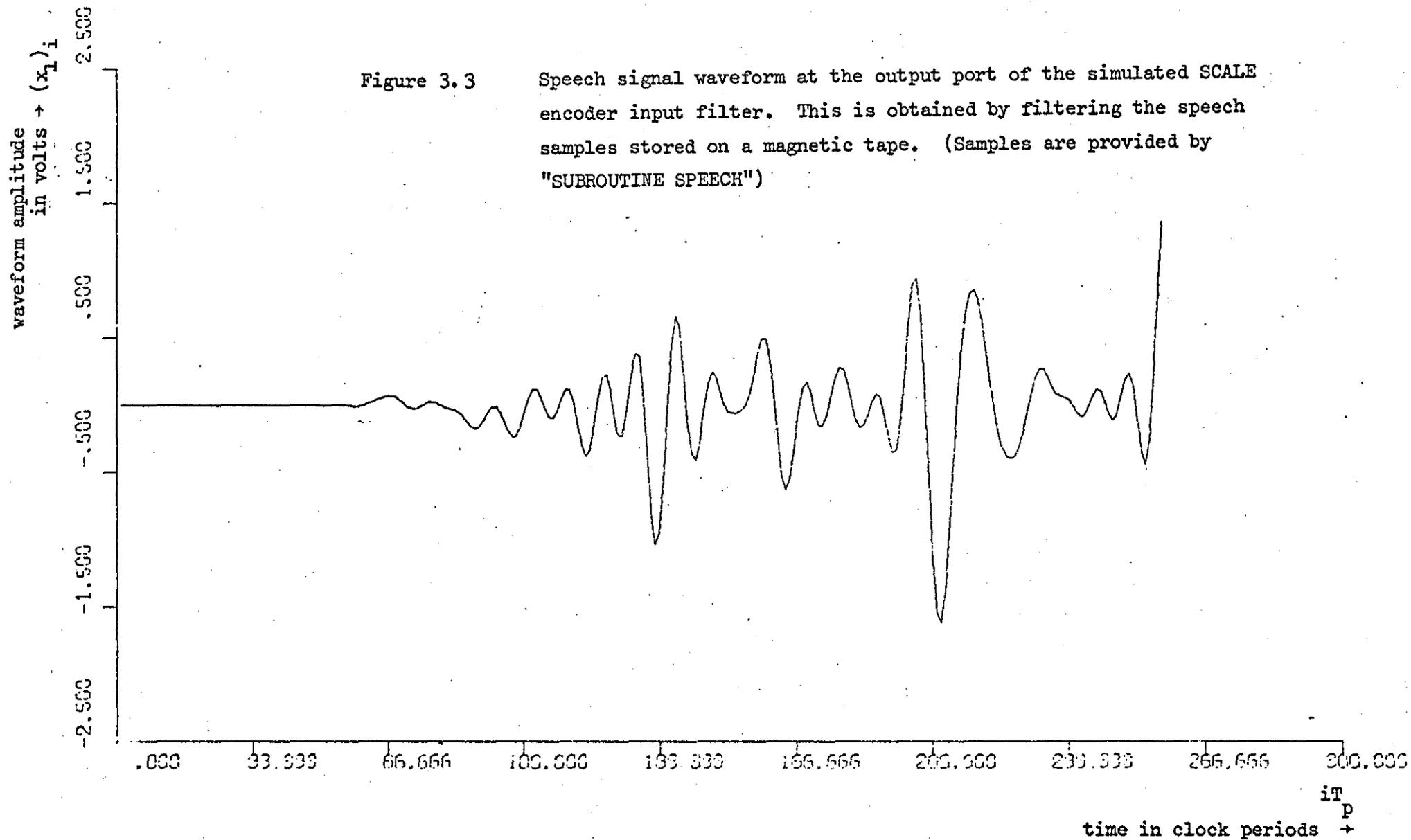
(b) Subunit 2 is a speech waveform generator. It consists of a magnetic tape on which samples of digitized speech are recorded. The speech waveform is sampled at a rate of 9.6 KHz and then analogue-to-digital converted prior to being stored on magnetic tape. The tape was made available for the author by Baskaran<sup>(60)</sup>. To obtain speech samples at a rate of 19.2 Kb/s for this simulation, the 9.6 KHz samples are recovered from the tape and the number of samples are doubled by interpolation. A flow chart of this subunit together with speech sample values,  $(x_1)_i$ ; and the details of the FORTRAN subroutine (SUBROUTINE SPEECH (INPUT, INPT 1,.....)) are given in Appendix B. A section of the waveform generated by this subroutine is shown in Fig. 3.3.

(c) Subunit 3 is a Gaussian signal generator with assignable mean  $\mu$  and variance  $\sigma$ . This type of signal is considered because of its similarity to speech signals and because it is employed in the standard British Post Office (BPO) noise measuring equipment (Marconi TF2343A<sup>(73)</sup>), see Chapter IV).

The flow chart of this subunit together with the detailed FORTRAN subroutine (SUBROUTINE GAUSS2 (SIGM, MU,...)) are given in Appendix B. The subroutine makes use of a documented computer library function (G05AEF) to generate the required signal with Gaussian amplitude probability distribution. G05AEF has been written by Nottingham Algorithm Group (NAG)<sup>(71)</sup> and is available in the University computer library. It employs linear feedback shift register techniques to generate trains of pseudo-random sequences in which millions of numbers are generated before a sequence repeats. The mean and variance of the distribution, SIGMA and MU respectively, are set externally by the programmer. Zero mean has been assumed throughout the simulation. A section of the generated Gaussian signals is shown in Fig. 3.6a.

Figure 3.3

Speech signal waveform at the output port of the simulated SCALE encoder input filter. This is obtained by filtering the speech samples stored on a magnetic tape. (Samples are provided by "SUBROUTINE SPEECH")



Bandlimiting of the speech and Gaussian signals is achieved by passing them through a low-pass digital filter the simulation of which is described below.

(ii) Simulation of the input and output digital filters

In the SCALE system the function of the input filter is to limit the bandwidth of the input signal and remove the out-of-band interfering signals. The output filter employed in the decoder is to eliminate the high frequency (above  $2.4 \text{ KHz}$ ) components of the quantization and overload noise residing in the reconstructed feedback signal  $H(t)$ . Due to the discrete nature of the signals inside the digital computer, filtering is essentially a digital processing operation in which a given sequence of numbers at the input produces another sequence at the output. The essentials of the theory and design of digital filtering are reviewed in Appendix C. The simplest method of designing a realisable filter to approximate to a given ideal shape in the frequency domain is the method presented by Rabiner<sup>(53)</sup>, and known as "the frequency domain sampling method" (see Appendix C). The specifications of the filters required for the SCALE system can be summarized as:

- (1) In the passband, the filters must have minimum amplitude distortion and linear phase/frequency characteristics.
- (2) The filters must have maximum attenuation in the attenuation band (above  $2.4 \text{ KHz}$ ).
- (3) The transition band must be minimum; but the transition from the passband to the attenuation band must be smooth to reduce the side lobes of the filter impulse response.

These specifications can be met by simulating a finite impulse response digital filter (FIR low-pass digital filter) using the

frequency domain sampling method. Through this method the filter can be synthesised by direct discrete convolution (see Appendix C). The output of the filter  $Y(\ell)$  is given by

$$Y(\ell) = \sum_{\ell=0}^{m-1} h(\ell)x(\ell - m) \quad (3.3)$$

where  $h(\ell)$  is the impulse response of the filter,  
 $x(\ell)$  is the input sequence of the signal to be filtered  
and  $h(m) = 0$  for  $m \geq \ell$ . (Finite Impulse Response)

The frequency domain method of modeling is based on choosing the frequency response of the filter  $H(j\omega)$  to satisfy the given specifications, sampling  $H(j\omega)$  and then computing the filter impulse response  $h(\ell)$  by means of the Discrete Fast Fourier transform DFFT (see Appendix C).

Therefore

$$h(\ell) = \frac{1}{M} \sum_{k=0}^{M-1} H_k e^{j2 k\ell/M} \quad (3.4)$$

where  $j = \sqrt{-1}$ ,  $k = 0, 1, 2, \dots, M-1$ ,  $\ell = 0, 1, \dots, M-1$

$M$  = the total number of frequency samples in  $H(j\omega)$

and  $H_k$  is one particular sample of  $H(j\omega)$ .

The shape of the frequency characteristic of the low-pass filter which is used for bandlimiting the input signals to the SCALE encoder and for eliminating the out-of-band noise at the decoder output is shown in Fig. 3.4.

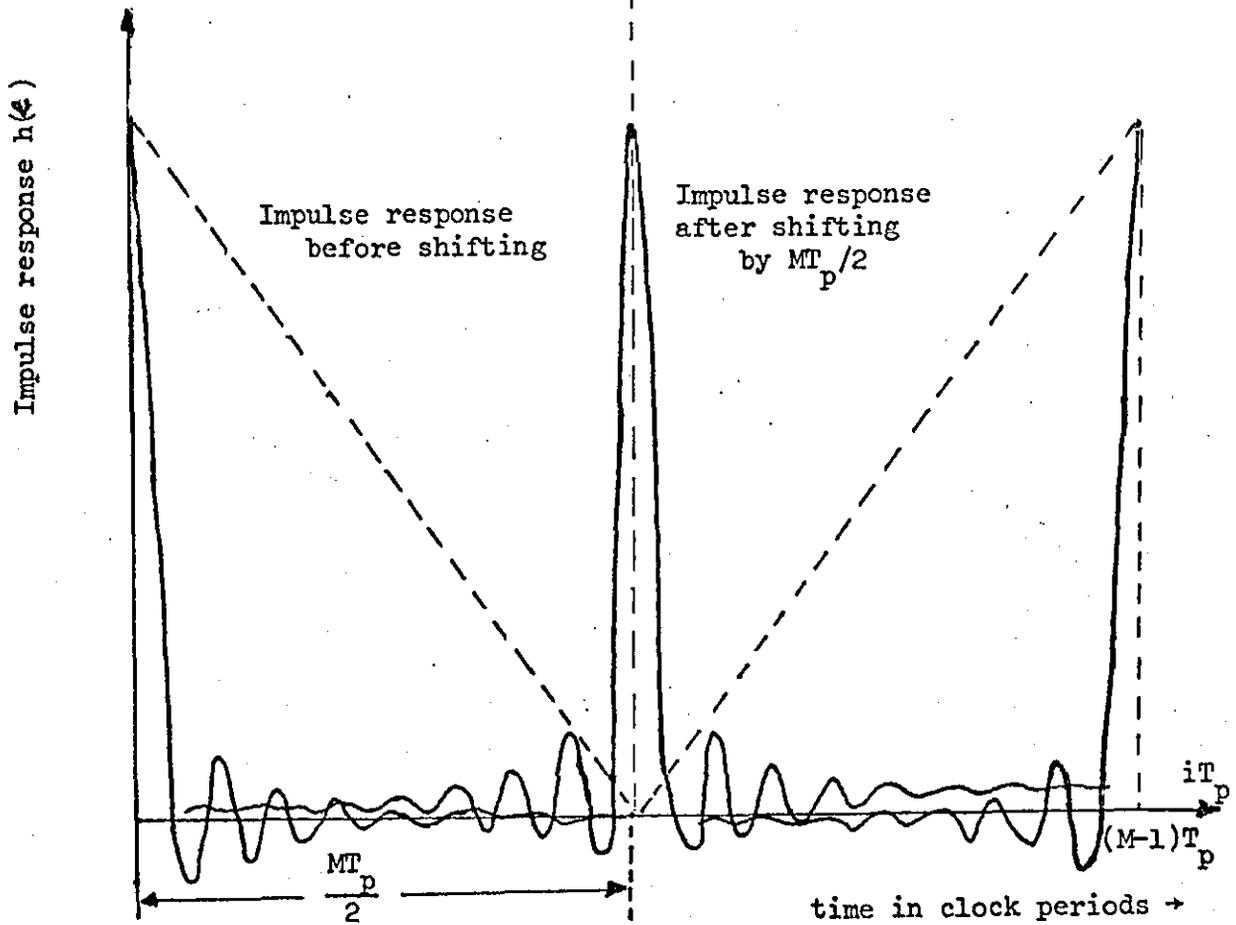
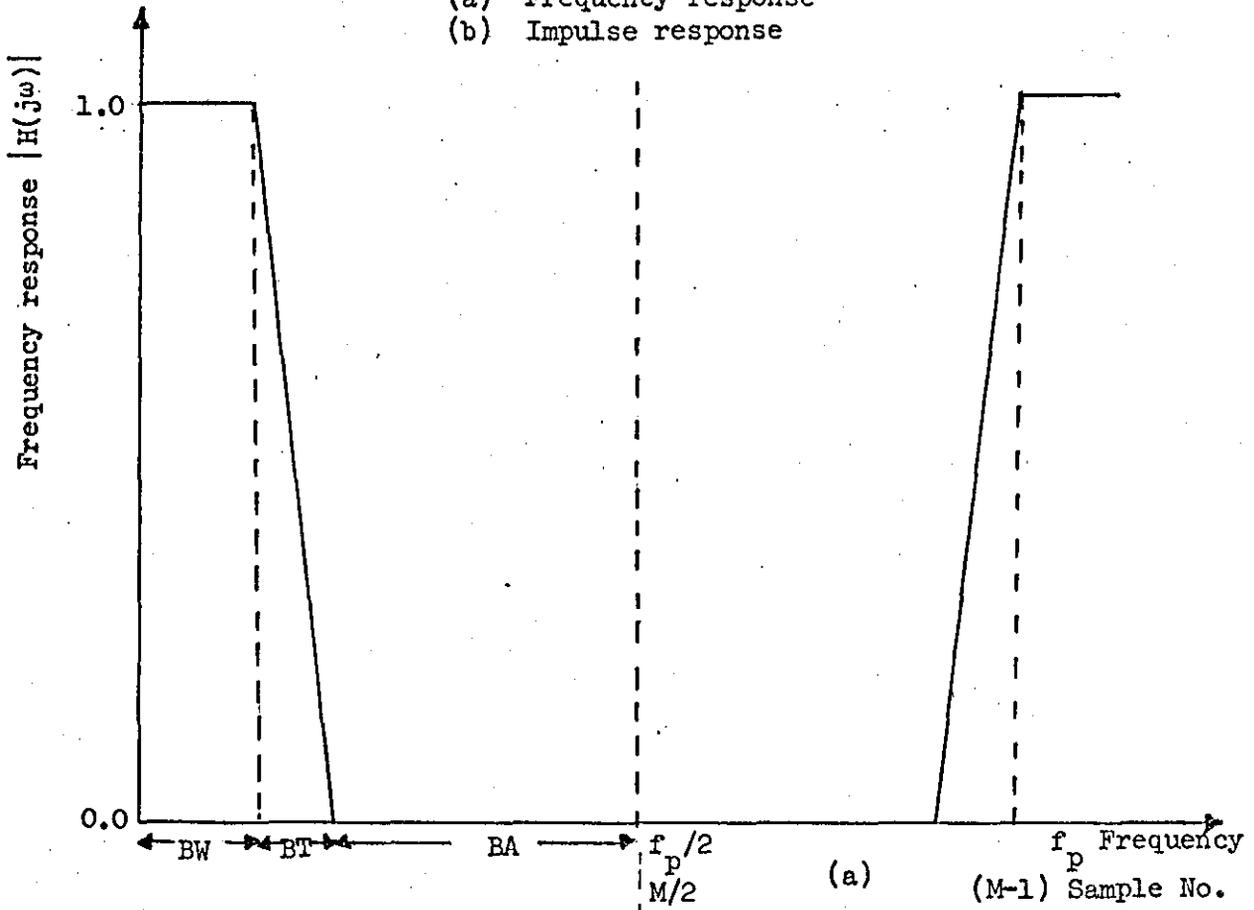
The frequency spacing between any two samples is given by:

$$\Delta f = f_p/M \quad (3.5)$$

Referring to Fig. 3.4a, the pass-band of the filter is taken as the frequency range from zero to the point where the transition band begins. In this range the filter frequency response samples are

Figure 3.4 The FIR digital filter characteristics

- (a) Frequency response
- (b) Impulse response



(b)

BA - Attenuation band      BT - Transition band  
 BW = Pass band            M = Number of frequency samples,  
 $f_p$  = clock rate

unity. In the attenuation band the values of the samples are zeros. In the transition band the values of the samples are fractions which are selected in such a manner that the value of the maximum side lobe in the impulse response is minimized (Minimax optimization technique). For a given  $M$  and base-band bandwidth  $BW$  in samples the values of the optimized transition band samples have been tabulated by Rabiner and are available to the designer.

$$BW = B/\Delta f \quad (3.6)$$

where  $B$  is the bandwidth of the filter in  $H_z$  and  $BW$  is the number of frequency samples within this bandwidth.

The design of the filter can be summarised as follows:

- (1) Select the value of  $M$  from Rabiner's et al tables. The selection must satisfy equation (3.6).
- (2) Calculate the number of frequency samples in the passband using equation (3.6).
- (3) Select the appropriate number of samples  $BT$  in the transition band and their corresponding values from Rabiner's tables.
- (4) Equate the  $BW$  samples in the passband to unity, the  $BT$  transition samples to their optimized values (fraction) and the following  $(M/2) - (BW + BT)$  stopband samples to zero. ( $M$  is even and symmetrical about  $M/2$ ).
- (5) Due to the symmetry of the filter response the full response can be obtained by repeating (4) in the reverse direction (i.e. starting at  $M/2$  and proceeding towards zero frequency).
- (6) Perform the inverse Discrete Fourier Transform operation IDFT, specified by equation (3.4) on the  $H_k$  samples. This operation is not applicable if the  $H_k$  samples are real numbers. For this reason an imaginary part is introduced into every  $H_k$  sample value.

$$\begin{aligned} H_k &= a_k + jb_k \\ &= a_k + j0 \end{aligned} \quad (3.7)$$

A FORTRAN subroutine "SUBROUTINE NLOGN (....)" was used in this simulation to accept the  $H_k$  samples of the filter in the frequency domain, and produce  $W(\ell)$  samples, which are the samples of the filter impulse response in the time domain. This subroutine is basically an implementation of equation (3.4) in FORTRAN. It was first presented by Robinson<sup>(56)</sup> and later slightly modified by Ackroyd<sup>(55)</sup>. The latter version is available in the Electrical Engineering disc File at the University computer centre. The impulse response obtained is not physically realizable because it has components in the negative time domain. To make the filter physically realizable the impulse response is shifted to the right along the time axis by  $M/2$  clock periods as shown in Fig. 3.3b. This leads to a realizable FIR digital filter which has a linear phase/frequency characteristic (i.e. a delay of  $MT_p/2$  seconds for all frequencies).

Analysis of the theory of the IDFT is given in Appendix C and the subroutine "SUBROUTINE NLOGN (....)" is given in Appendix B.

- (7) The filtering process is carried out by performing the Discrete convolution operation - specified by equation (3.3) - on the real part of the filter impulse response samples and the samples of the signal to be filtered. Further details of the theory of convolution and the implementation of equation (3.3) in FORTRAN are given in Appendices C and B. The FORTRAN subroutine "SUBROUTINE CONVOL (....)" accepts the filter impulse samples  $(h(0), h(1), \dots, h(M-1))$ , and the samples of the signal to be filtered  $(x_2(0), x_2(1), \dots, x_2(N-1))$  or  $(H(0), H(1), \dots, H(N-1))$ , and produces a filtered signal  $(x_2(0), x_2(1), \dots, x_2(N-1))$  or  $(\hat{x}_2(0), \hat{x}_2(1), \dots, \hat{x}_2(N-1))$ , depending on whether the filter is used as an input or output filter.

It must be remembered that the filter has a delay of  $(MT_p/2)$  seconds. This can be equalized by shifting the output of "SUB-ROUTINE CONVOL (....)" to the right by  $M/2$  clock periods.

In military applications the sampling frequency  $f_p = 19.2$  Kb/s and the bandwidth  $B$  of the filter is  $2.4$  KH<sub>z</sub>. For this case the value for  $M$  is 128 and  $BT = 3$  (transition band samples).

From (3.5) the separation between frequency samples is

$$\Delta f = f_p/M = 19200/128 = 150 \text{ H}_z$$

and from (3.6) the total number of frequency samples in the pass-band is:

$$BW = B/\Delta f = 2400/150 = 16 \text{ samples.}$$

Following the design procedure discussed above the BW frequency samples  $(H_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_9, H_{10}, H_{11}, H_{12}, H_{13}, H_{14}, H_{15})$  and  $(H_{126}, H_{125}, \dots, H_{111})$  are to be equated to unity. i.e.

$$\left. \begin{aligned} H_0 &= H_1 = H_2 \dots\dots\dots = H_{15} \\ &= H_{126} = H_{125} \dots\dots\dots = H_{111} \end{aligned} \right\} = 1.0$$

The optimized values of the attenuation band are obtained from Rabiner's tables as:

$$\begin{aligned} H_{16} = H_{110} &= 0.72843530, & H_{17} = H_{109} &= 0.258167440, \\ H_{18} = H_{108} &= 0.02633057 \end{aligned}$$

The rest of the samples  $(128-37 = 91)$  are equated to zero.

The impulse response is obtained by introducing an imaginary part into the frequency samples, taking the inverse DFT and shifting the resultant (unrealizable) impulse response by  $M/2$  time samples along the time axis to produce the FIR linear phase digital filter impulse response

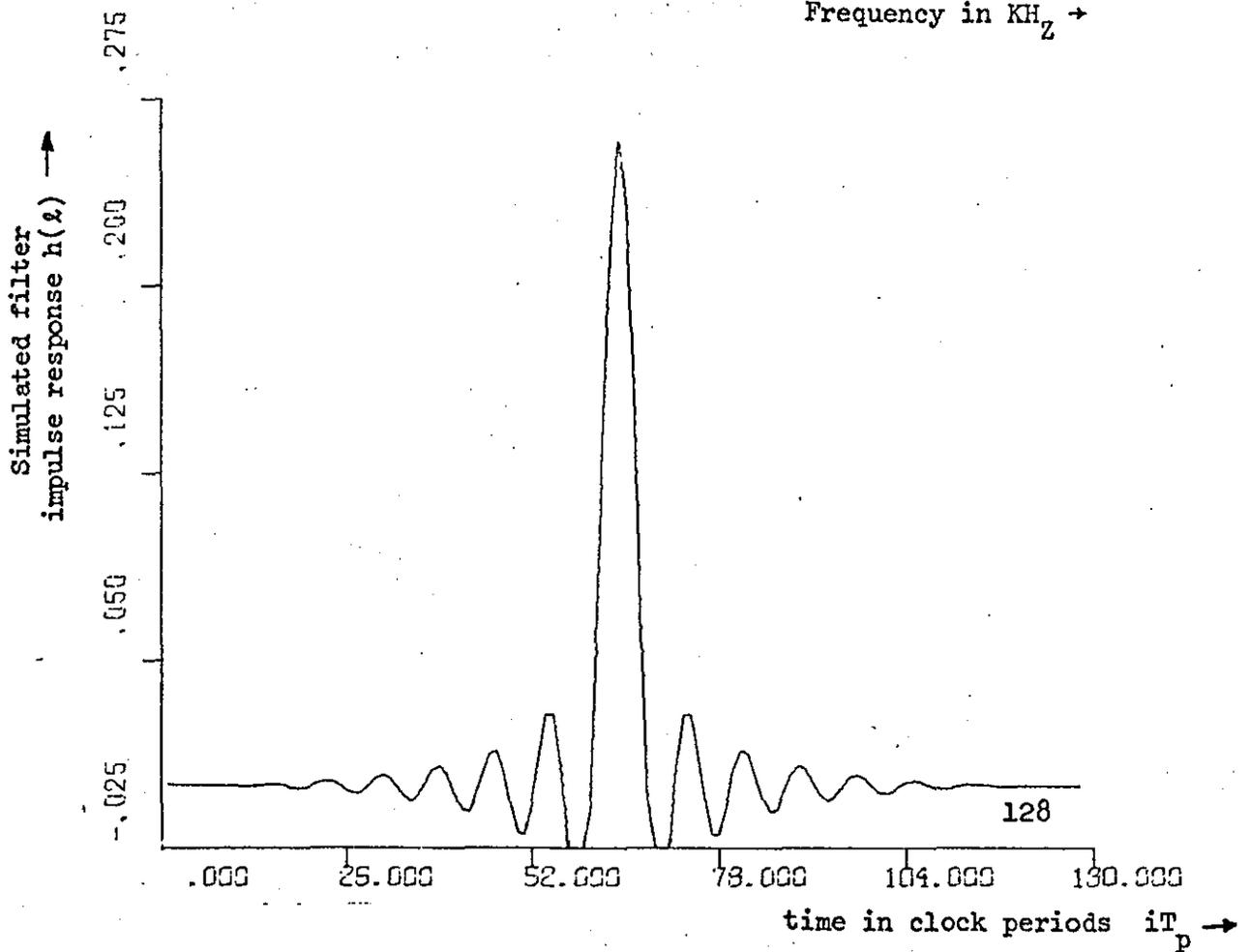
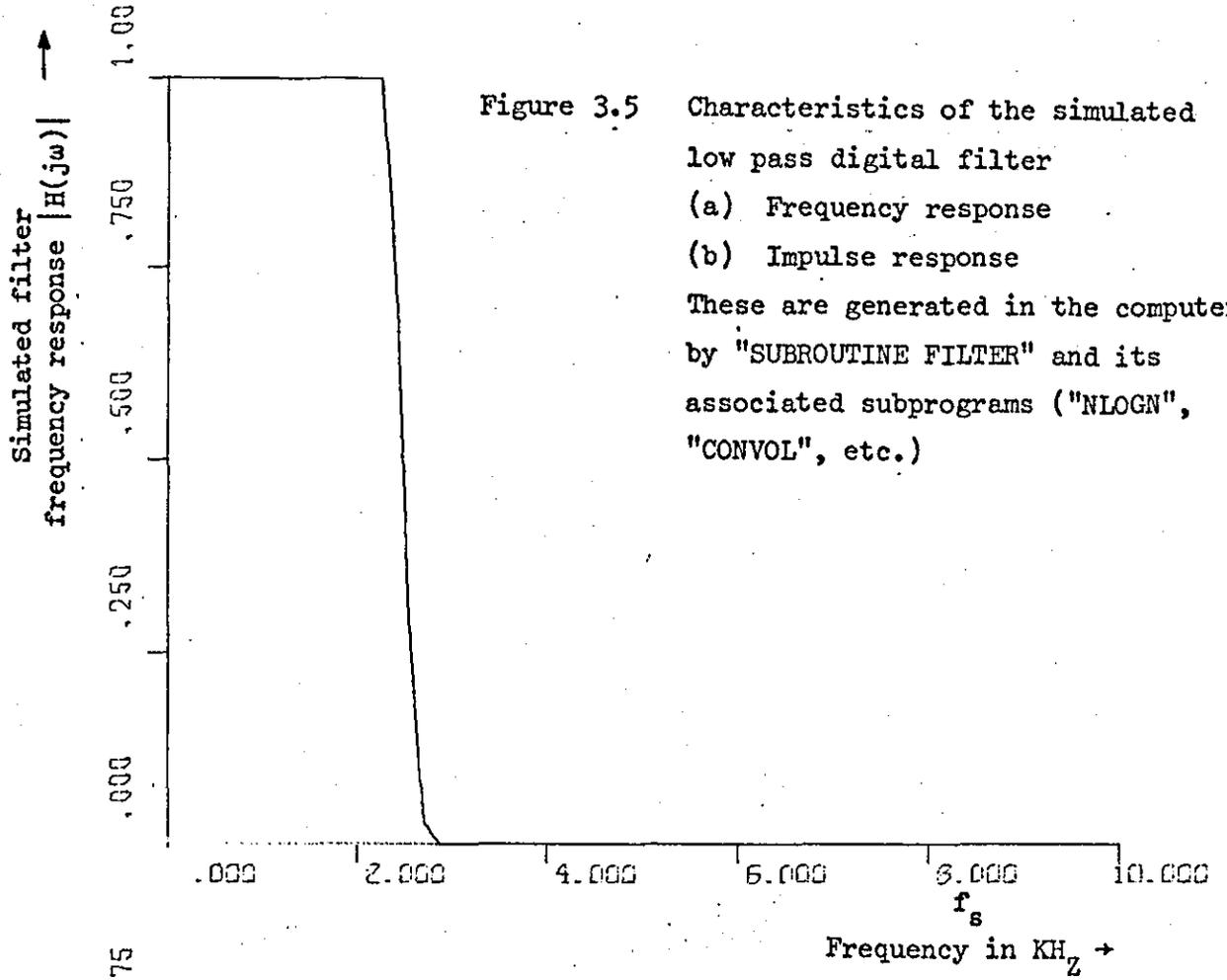
which is shown in Fig. 3.5. The flow chart of this process and the FORTRAN subroutine "SUBROUTINE FILTER (NSAMPL, ORDER, HORDER, BW, HREAL, HLOMLX)" are given in Appendix B.

To demonstrate the filtering action a wide band Gaussian signal was applied to the input of the filter. The input and the output waveforms are shown in Fig. 3.6.

(iii) Simulation of the Encoder and the Decoder of the SCALE system

The SCALE CODEC which was modeled in this study is that shown in Fig. 2.3. It consists of an encoder (Fig. 2.3a) and a decoder (Fig. 2.3b). The simulation of the input and output filters have already been discussed. It can be seen from Fig. 2.3 that the encoder feedback loop is a local decoder which is exactly identical to the remote decoder (excluding the output filter). For this reason it is only necessary to simulate the encoder part. The decoder output can be obtained by filtering the local decoder output steps  $H(t)$ .

A flow chart of the operation of the CODEC based on the block diagram of Fig. 2.3a and its basic equations is given in Appendix B together with the FORTRAN implementation of the flow chart. The FORTRAN subroutine "SUBROUTINE CODEC(NPULSE, KAA, XT, OUTPUT, NS, NNN, FP, IJ)" accepts either NPULSE (Number of samples) input signal samples - when NNN = 1 - in the array "XT (NPULSE)" and produces NPULSE  $L(t)$  samples stored in array "KAA(NPULSE)". In this case the subroutine acts as an encoder. Alternatively, if NNN is equated to 2 when the subroutine is called, the subroutine accepts NPULSE samples of the digital output signal  $L(t)$  stored in the array "KAA (NPULSE)" and produces NPULSE samples of the reconstructed feedback signal  $H(t)$ , stored in the array "OUT (NPULSE)". Hence the subroutine can be used as an encoder or as a decoder by equating NNN to the appropriate number. FP and NS are switching para-



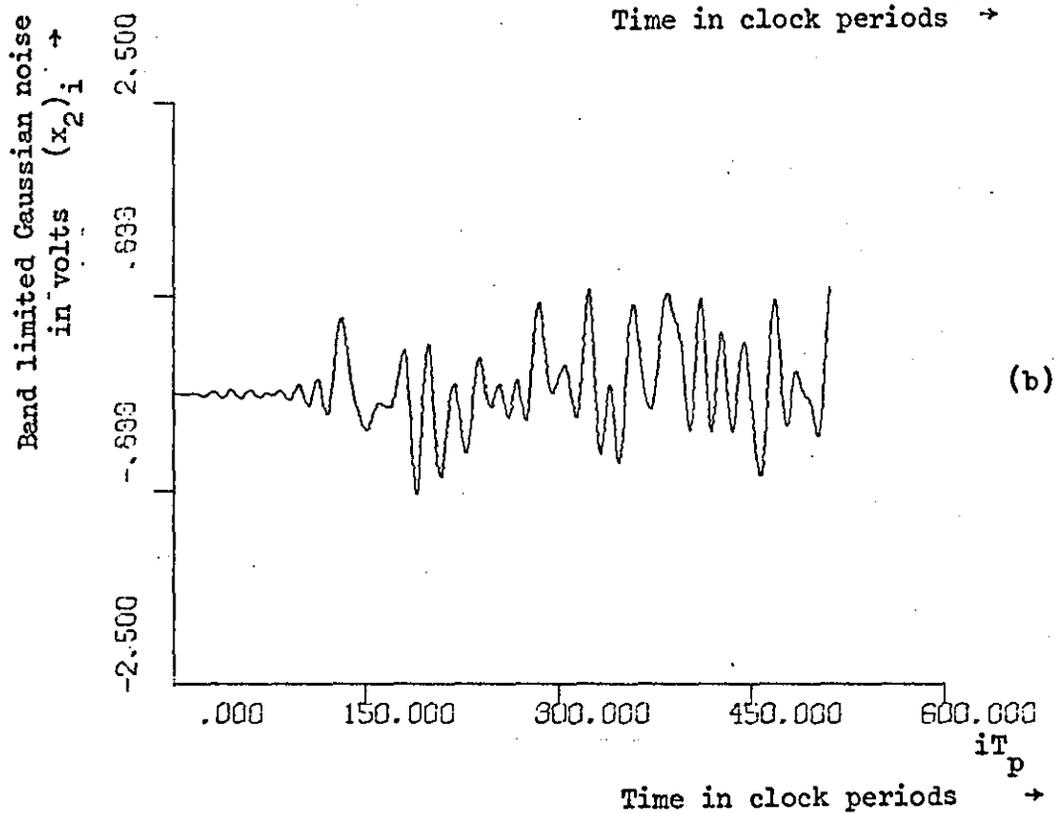
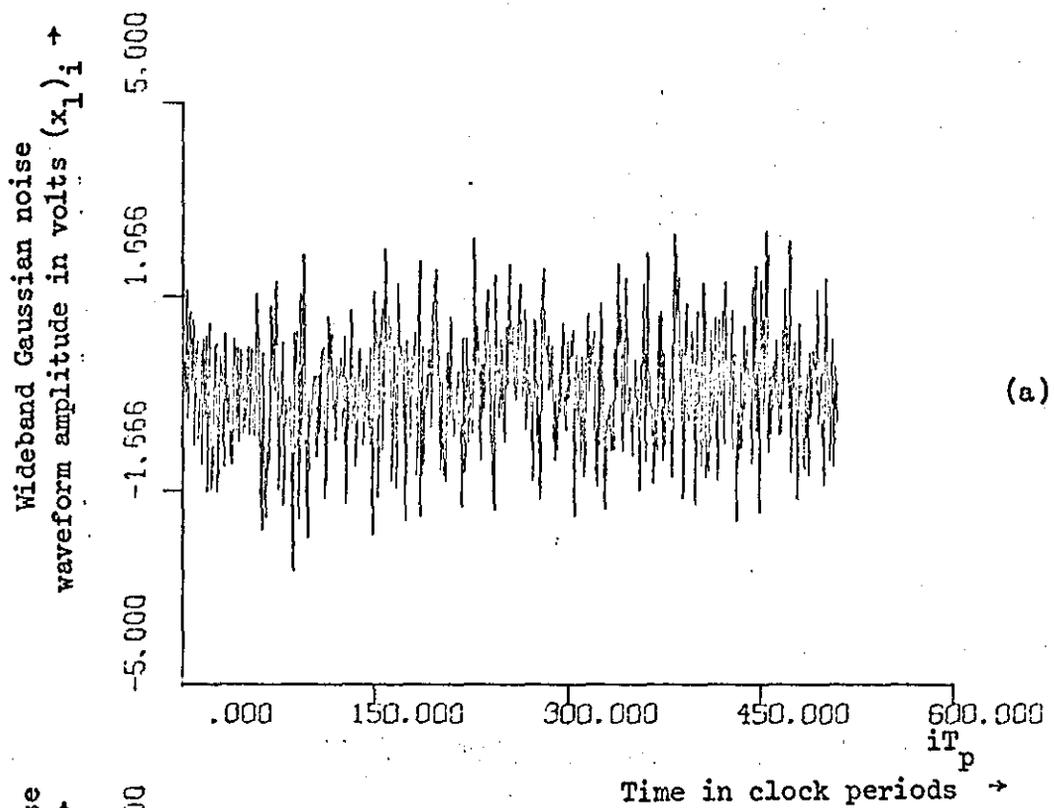


Figure 3.6 Simulated Gaussian noise waveforms

- (a) Wideband Gaussian noise waveform generated in the computer by "SUBROUTINE GAUSS2". (Variance  $\sigma^2 = 1.44$ , mean  $\mu = 0.0$ )
- (b) Band limited Gaussian noise waveform limited to 2.4 KHz by the simulated digital filter (in the form of "SUBROUTINE FILTER")



This simulates the action of the selection logic SL.

- (6) The action of the detection logic DL is simulated by translating equation (2.8) into FORTRAN producing statements 12, 10, 8 and 14 corresponding to 6, 5, 4, 3 stage SR lengths respectively.
- (7) The PHM is simulated by translating its equation (2.16) into FORTRAN giving the statement

$$H = \text{GRAD}*(V_c - EZ) + \text{HMINM}$$

Overload causes this statement to be by-passed and statement 51 to be executed. The sign of H is decided by the execution of statement number 52.

- (8) The difference circuit is described by the FORTRAN translation of equation (2.2) leading to statement 55.
- (9) The action of the error integrator is estimated by multiplying the present input by a growth function  $(1 - e^{-T_p/T_i})$  and the previous integrator output by a leakage function  $(e^{-T_p/T_i})$  and then summing the two resultants to produce the effective error voltage at the input of the quantizer.

This is written in FORTRAN as

$$V1 = A*\text{DIFFCE} + BV1$$

where A in this context is a FORTRAN variable.

- (10) The "logical IF" statement simulates the action of the comparator:

$$\text{IF}(V1.\text{GE}.0.0)\text{G}\phi \text{ T}\phi 53, \text{ELSE KA} = 0.$$

- (iv) Displaying the various waveforms

In ordinary laboratory electronics experiments the various waveforms and tests results are usually displayed on the screen of a CRT (cathode ray tube). In off-line simulation experiments it is not possible to use a CRT but instead some form of plotting routines are often used to control

a graph plotter which displays the experimental results in graphical form.

Efficiently written graph plotting routines are available in the University computer library. They are written in the ICL 1904A computer assembly language - PLAN - but can be called from FORTRAN Programs. These routines control the "X-Y plotter" under the control of the data to be plotted.

It is an advantage of computer simulation over practical experimentation that waveforms at any point in the circuit can be displayed and observed without affecting the operation of the system.

The two graph plotting routines which were used most in this simulation are: "UTP4C" and "UTP4B". The first scales the data and plots the "X-Y axes" while the second controls the actual plotting of the X and Y data points which are stored in two real arrays "X" and "Y". The call for "UTP4C" is

```
CALL UTP4C (XMIN,XMAX,YMIN,YMAX,XINS,YINS,'XTITLE',IYTITLE,J)
```

where XMIN and YMIN define the origin of the graph and must not be less than 0.001 in magnitude. XMAX and YMAX define the maximum values of the largest X and Y element respectively and must not exceed 1000.0. XINS, YINS define the lengths of the X and Y axis in inches. YINS must not exceed 24 inches. I and J are the numbers of the 8-character words constituting the X and Y axes labels. I and J must not exceed 4 words.

The call for "UTP4B" is

```
CALL UTP4B(X,Y,N,I)
```

where X and Y are the arrays holding the data of the independent and the dependent variables respectively, N is the number of coordinates to be plotted and I defines the method by which the coordinates are joined and the number of graphs to be plotted on the same axis. e.g. I = 3: one curve per axis is to be plotted and the coordinates are joined by a straight line. I = 0: 9 curves can be superimposed on one another and the X-Y coordinates in every graph are joined by fitted sections of a

cubic equation.

### 3.2.4 Debugging and verification of the simulation subprograms

There are three different types of errors that might be present in a computer program:

- (1) Source-program-language errors
- (2) Execution errors (e.g. division by zero, address outside the program area, etc.)
- (3) Errors in the logic and formulation of the program

The first two are simpler because the program cannot run until they are removed. They can easily be corrected with the aid of compilation and execution error internal error scanning programs under the control of which the user programs are run. The third type of errors are the most insidious because the program can be compiled, run and produce numerical results except that the results produced are different from the correct ones. This type of error is very serious because it can lead the experimenter to believe in totally wrong answers and consequently to designing absolutely useless systems. For this reason the simulated model of the SCALE CODEC and its associated subunits were subjected to an extensive testing trial in which the waveforms at every relevant test point were displayed and checked sample by sample with the aid of printed results. The results were finally compared with those obtained by measurements on a practical SCALE model.

### 3.2.5 Using the model to study the SCALE system

Having established that the simulated system does represent the SCALE CODEC, it was used to conduct the following experiments:

#### (i) Observation of waveforms

Many published results of speech research experiments (54,67,68,69,70)

indicate that the preservation of the speech waveform by the CODEC is not strictly an important criterion. In fact many speech signal processing techniques, such as the use of Vocoders<sup>(57,58)</sup> and formant coding<sup>(98-100)</sup> are based on this principle. In telecommunication the most important criteria are intelligibility and SNR. Speech waveforms can suffer a great deal of waveform distortion while intelligibility is preserved<sup>(54,67)</sup>.

In experimental simulation, however, comparison between the input and the reconstructed waveforms does give some indication of how the hardware model might perceptually perform<sup>(68)</sup>. In the meantime, observing the waveforms at the various points on the CODEC system serves as a valuable means of debugging and evaluation of the computer program. The procedure for waveform observation on the simulated SCALE system is depicted in the flow chart shown in Fig. 3.7.

The program was compiled and run employing a sampling frequency  $FP (= f_p) = 19.2 \text{ Kb/s}$ , step input of 2.4V and an effective SR length  $NS (= N_e) = 3 \text{ bits}$ . The observed control voltage  $V_c$ , the feedback step height  $H(t)$ , the error voltage  $\epsilon_1(t)$  and the integrated error voltage  $\epsilon_2$  as functions of time (in clock periods), are displayed in Fig. 3.8 parts (a), (b), (c) and (d). With the same values of NS and FP the program was run again with the speech generator "SUBROUTINE SPEECH (...)" providing the input signal to the encoder. The resultant input to the encoder and the reconstructed signals at the decoder output are displayed in Fig. 3.8(e).

It must be noted that in the case of the step input the waveforms can be regarded as functions of time in clock periods or as functions of "block length", where "block length" means the number of consecutive like digits in the channel. The latter approach has been introduced by Flood<sup>(59)</sup> and can be useful in calculating the feedback loop characteristics (see Chapter V).

Further runs of the program were executed with the value of FP as

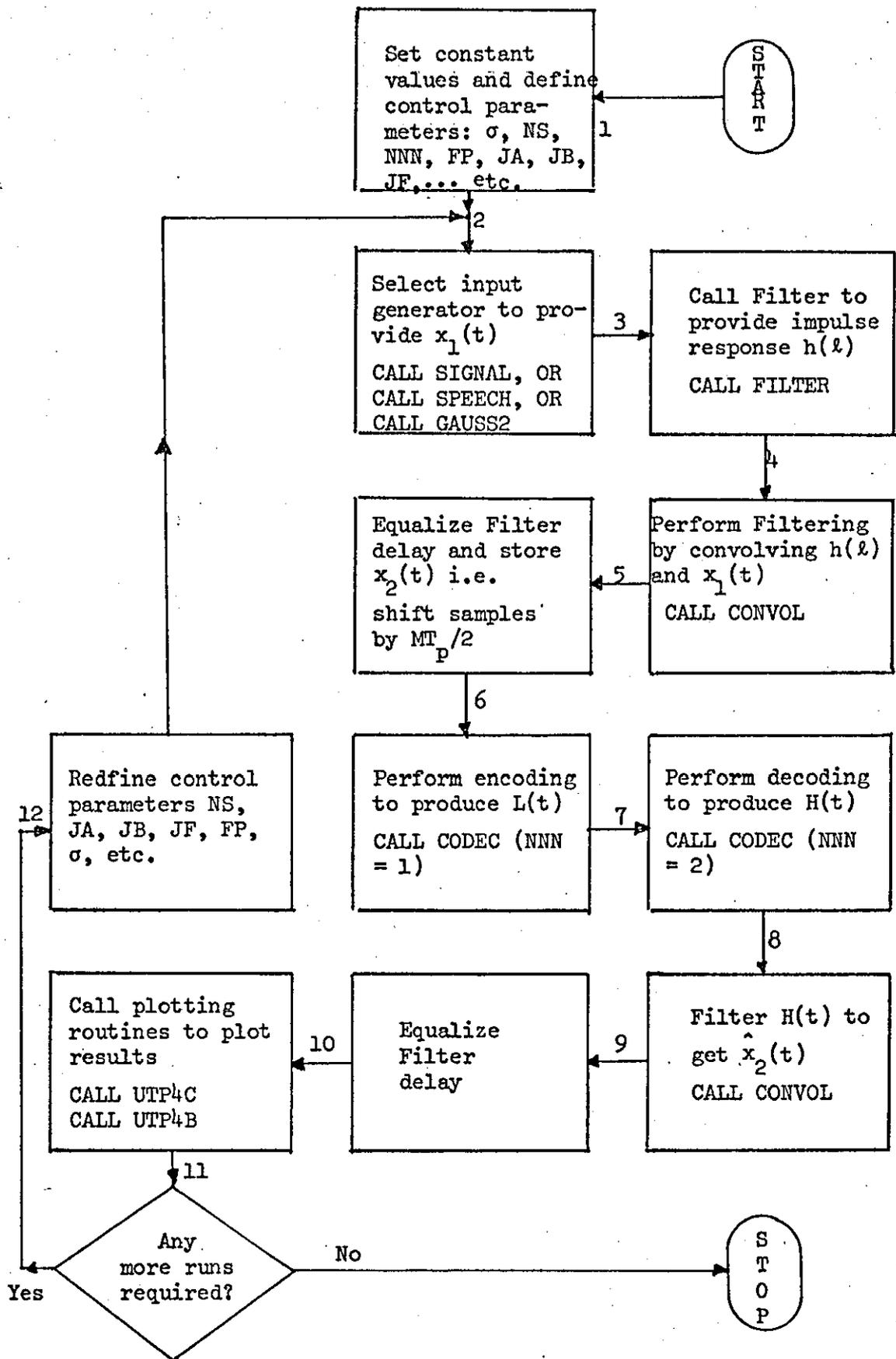


Figure 3.7 Flow diagram of computer runs for the purpose of waveform observation

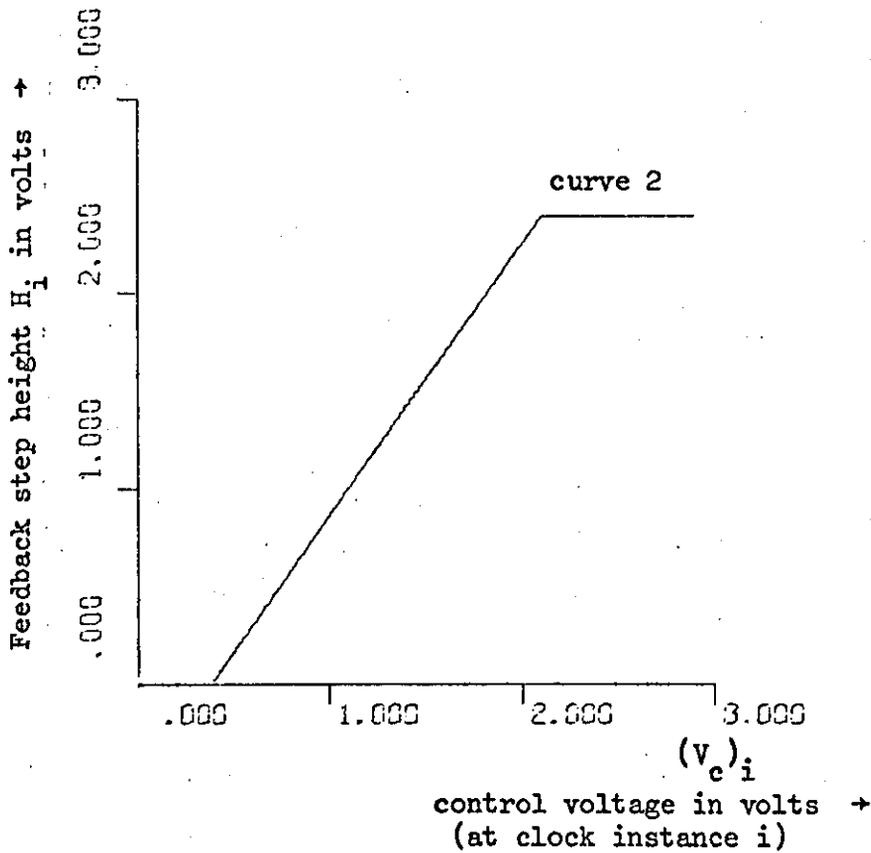
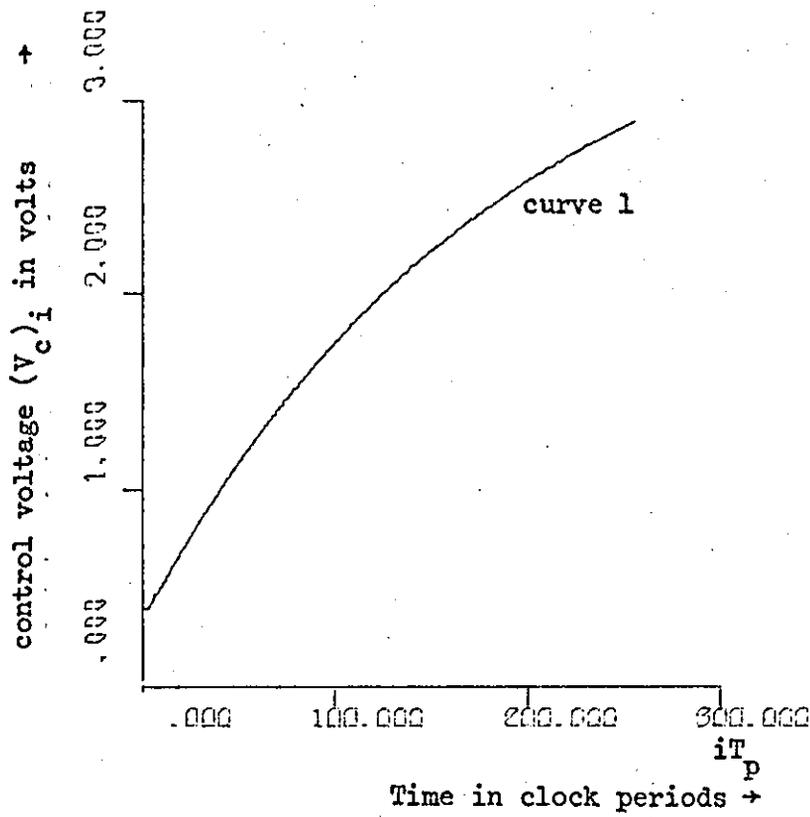


Figure 3.8(a) curve 1: control voltage  $V_c$  as a function of time (in clock periods), shift register length  $N_e (= NS) = 3$ , input signal is a step of amplitude 2.4 volts

curve 2: feedback step height (at the  $i^{\text{th}}$  clock instance) as a function of  $(V_c)_i$

$$f_p = 19.2\text{KHZ}$$

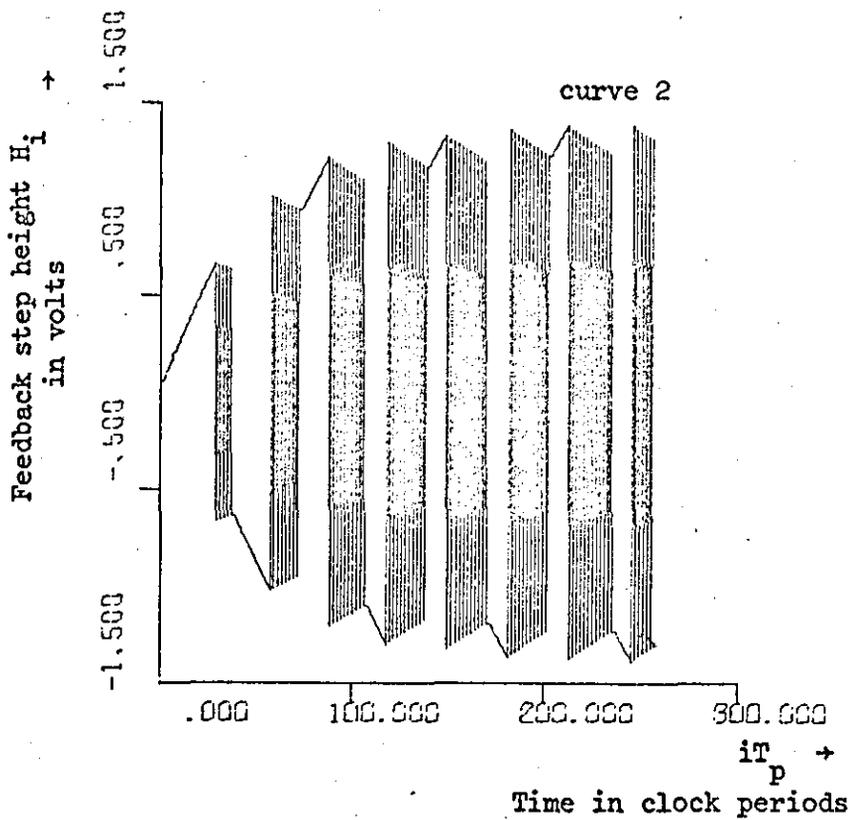
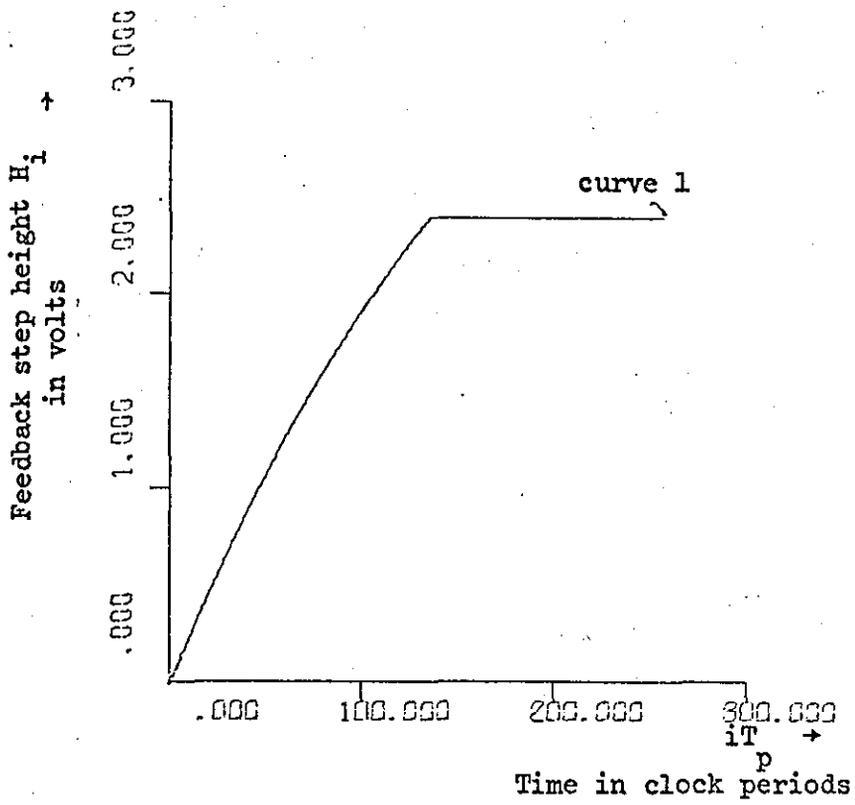


Figure 3.8(b). Observed feedback signal  $H_i$  as a function of time (in clock periods). Sampling frequency  $f_p = 19.2 \text{ Kb/s}$ , shift register length  $N_e = 3$

Curve 1: Input is a step of amplitude 2.4 volts

Curve 2: " " " sinusoid of amplitude 1.5 volts and frequency  $f_s = 1000 \text{ Hz}$ .

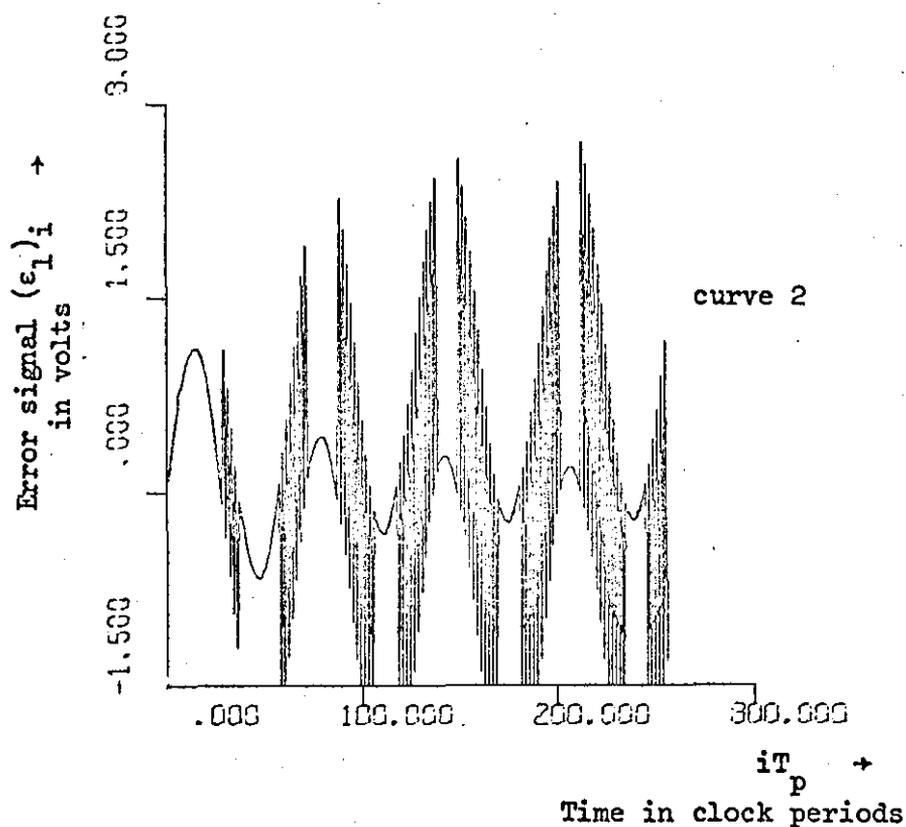
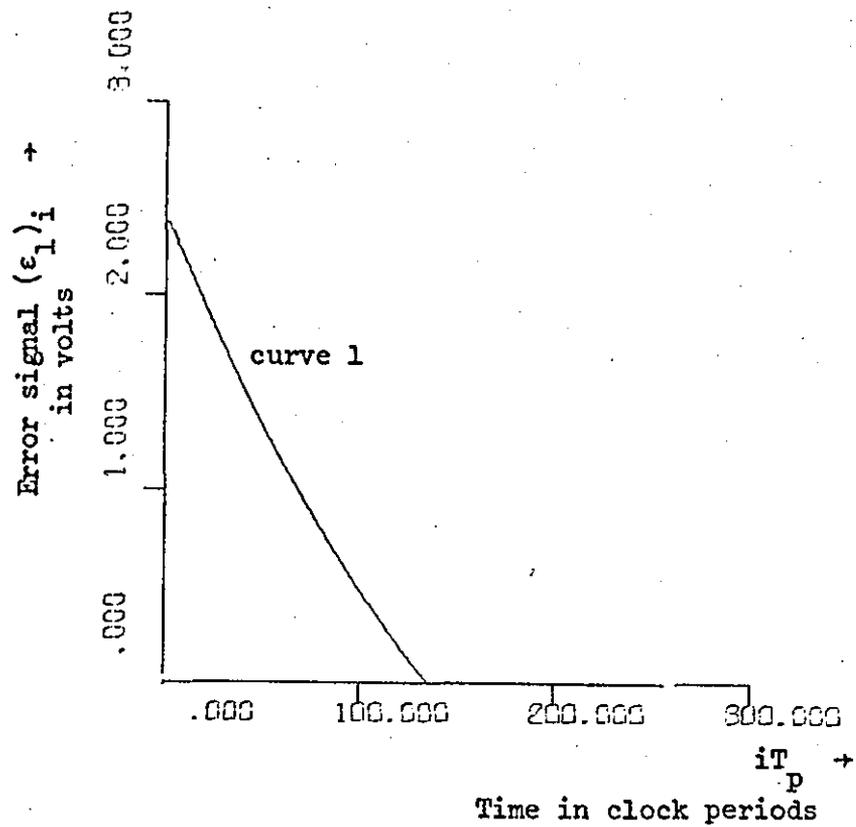


Figure 3.8(c) Observed error signals  $(\epsilon_1)_i$  as functions of time (in clock periods)

Curve 1: The input is a step of amplitude 2.4 volts

Curve 2: " " " a sinusoid of frequency  $f_s = 1000$  HZ

Curves 1 and 2 correspond to the feedback signals shown in Figure 3.8(b)

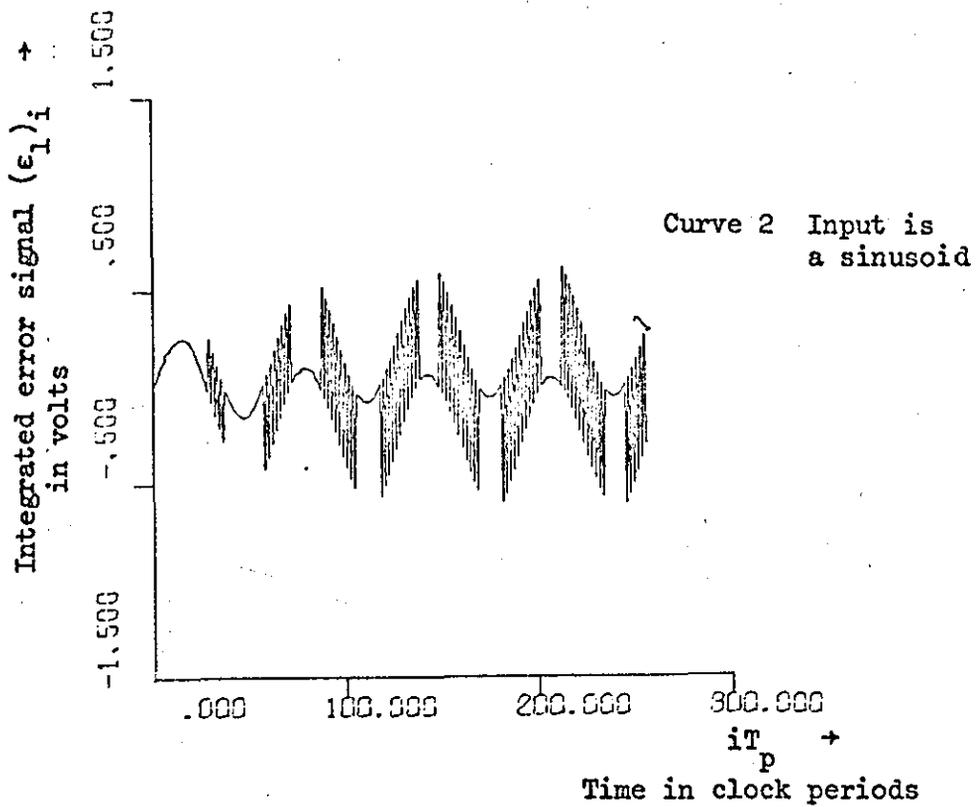
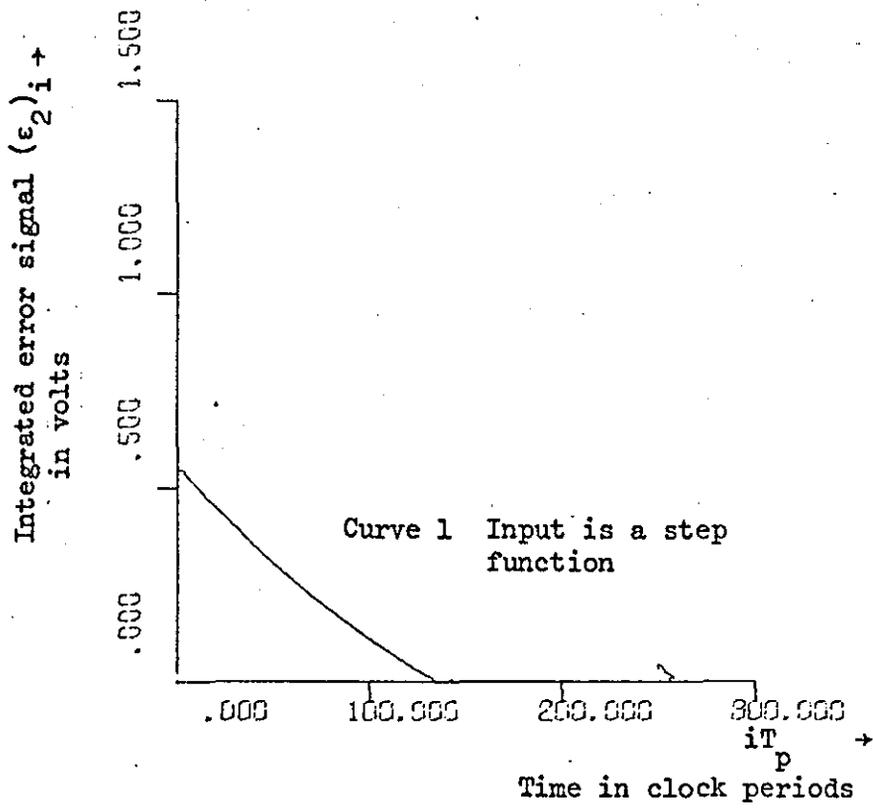


Figure 3.8(d) Integrated error voltage  $(\epsilon_2)_i$  corresponding to the conditions defined in Figures 3.8(b) and 3.8(c)

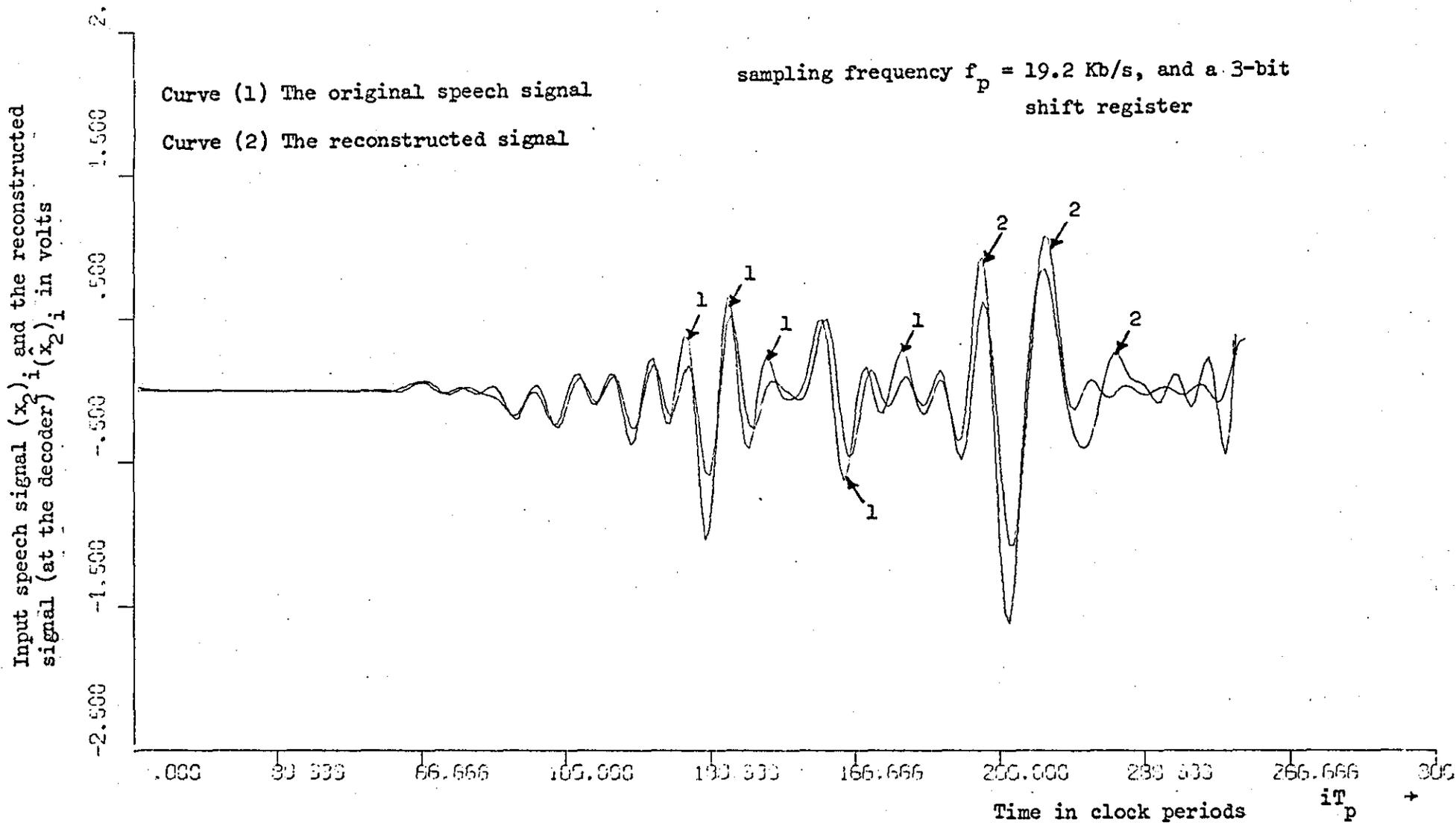


Figure 3.8(e) Input speech signal  $(x_2)_i$  as a function of time (in clock periods) and the decoder reconstructed signal  $(\hat{x}_2)_i$ .

before but with sinusoidal input of assignable frequencies. The process was repeated for SR length NS = 6,5,4,3. The results are displayed in Fig. 3.9.

The lengths of the blocks of the input and output samples dealt with in this experiment are 256 samples occupying a time period which is closely comparable with a pitch period.

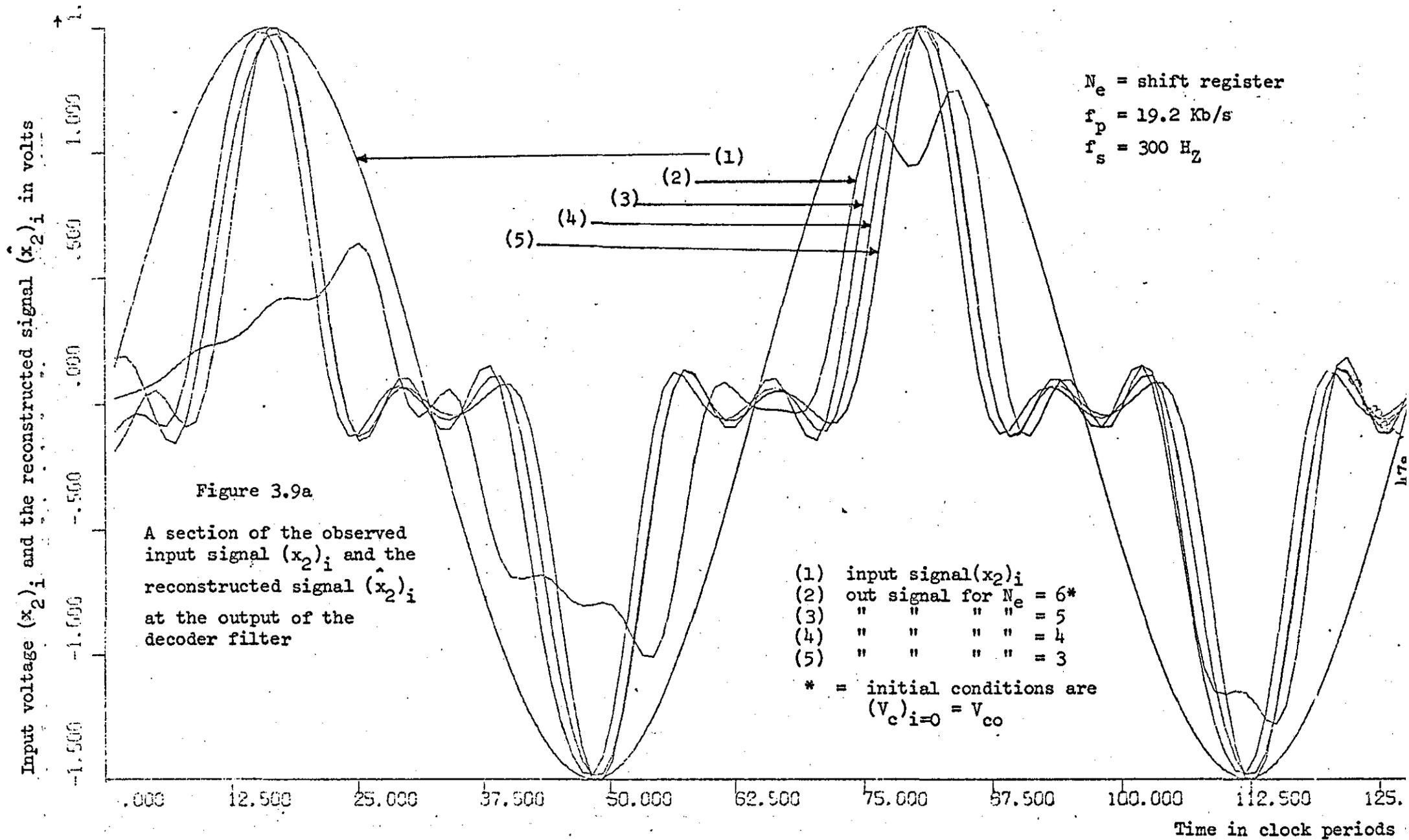
(ii) Measurements of  $V_c(t)$  and  $H(t)$  in the simulated model

The objective of this experiment is to measure the variations in the feedback step height magnitude  $H(t)$  and the control voltage  $V_c(t)$  as functions of the r.m.s. value of the input signal  $x_2(t)$ .

To achieve this the procedure described by Fig. 3.7 is again followed, but this time the input is a wide band Gaussian signal with zero mean and variance  $\sigma^2$ . 28 values of  $\sigma$  are selected randomly but in increasing order, and stored in the computer memory. With FP = 19.2 KHz and NS = 3 bits, the loop 2,3,4,5....12,2 (see Fig. 3.7) is executed 28 times. During each run the following steps are taken:

- (1)  $\sigma$  is incremented (read next  $\sigma$  from memory).
- (2) Call "GAUSS2" to generate a wideband Gaussian signal with variance  $\sigma^2$ .
- (3) Band limit the Gaussian signal to a bandwidth  $B = 2.4 \text{ KHz}$  by filtering.
- (4) Calculate the r.m.s. value of the band limited signal samples over a period  $\tau_p = 256T_p$  (approx. one pitch period), designate this r.m.s. voltage by  $\sigma_x$  and store it in memory.
- (5) Calculate the r.m.s. value - over the same period  $\tau_p$  - of the control voltage  $V_c(t)$  and the feedback step height  $H(t)$ , designate the resultants by  $V_{cp}$  and  $H_p$ , respectively, and store these in the computer memory.

The values of  $\sigma$  used for the first and the 27 subsequent runs, and



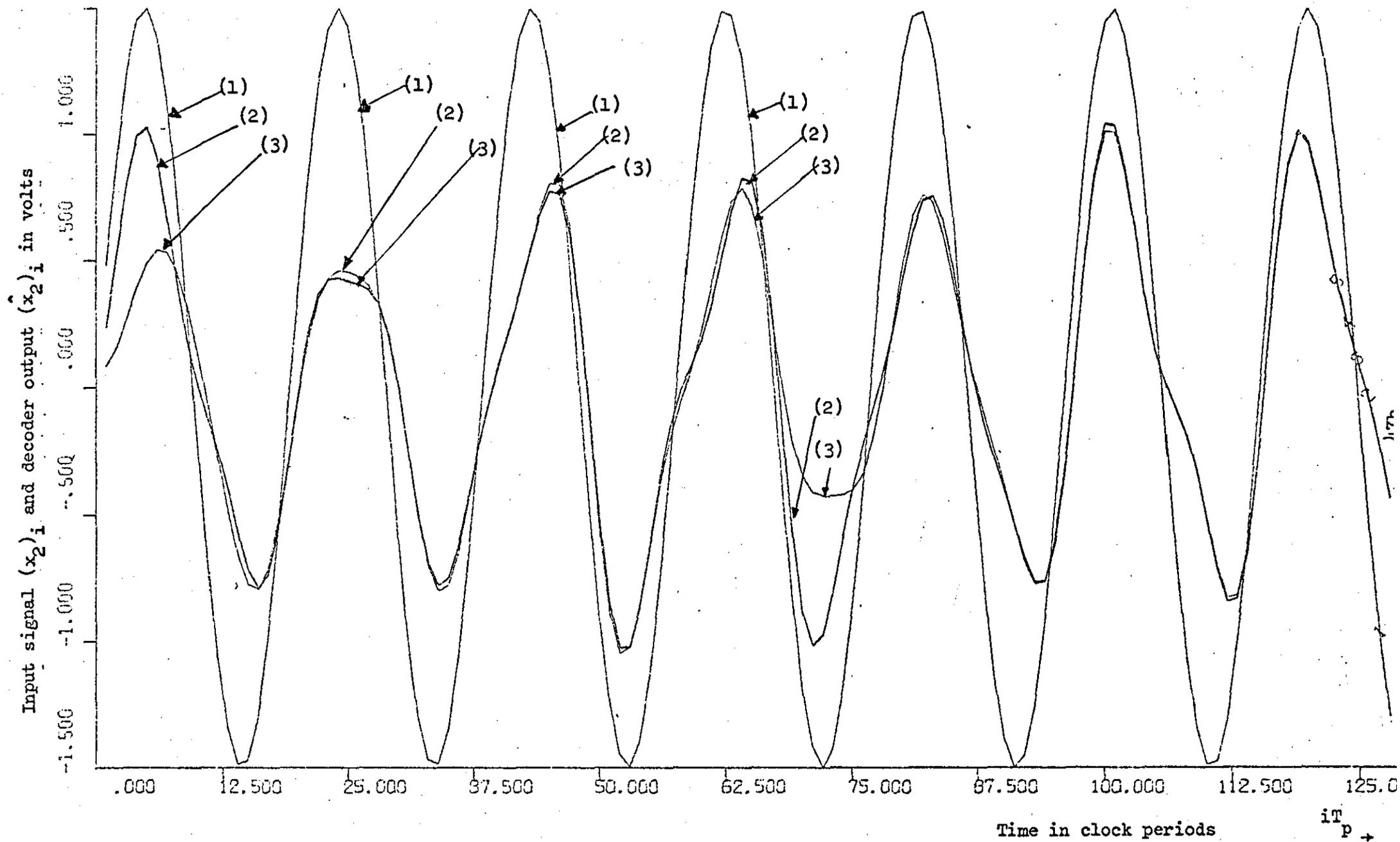


Figure 3.9(b) Input and decoded output signals.  $f_s = 1000 \text{ Hz}$   
 Curve (3) for  $N_e = 5$  and 6, curve (2) for  $N_e = 3$  and 4. All other conditions as for Fig. 3.9(a)

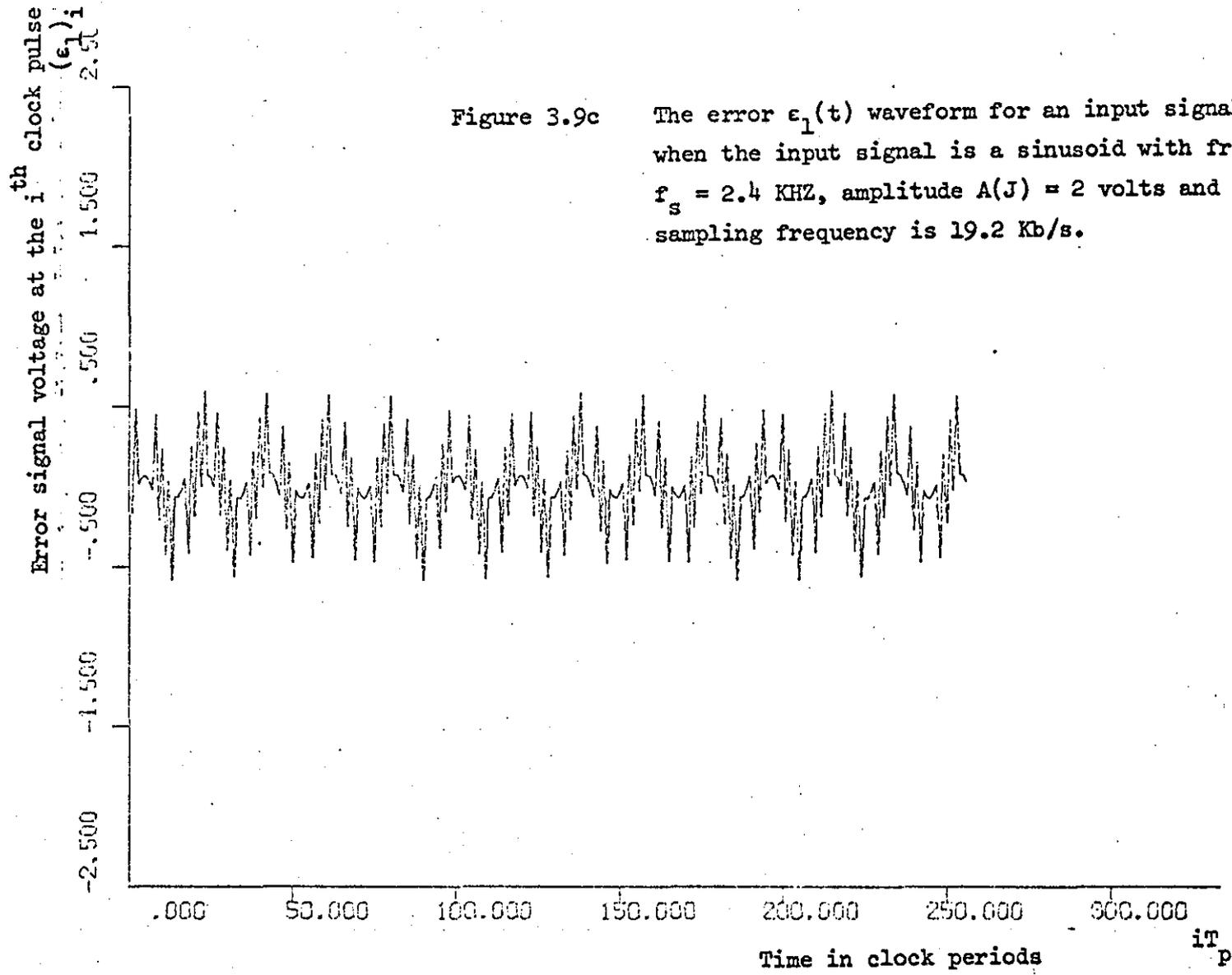


Figure 3.9c The error  $\epsilon_1(t)$  waveform for an input signal  $x_2(t)$  when the input signal is a sinusoid with frequency  $f_s = 2.4$  KHZ, amplitude  $A(J) = 2$  volts and the sampling frequency is 19.2 Kb/s.

the corresponding resultant values of  $\sigma_x$ ,  $V_{cp}$  and  $H_p$  are given in Table 3.1. The recorded variations of  $H_p$  and  $V_{cp}$  are plotted versus the corresponding values of  $\sigma_x$  in Fig. 3.10.

A graph of the Gaussian noise signal generated by "GAUSS2 (...)" before and after bandlimiting - for  $\sigma = 1.2$  - is shown in Fig. 3.6

### 3.3 SUMMARY OF THE SIMULATION RESULTS

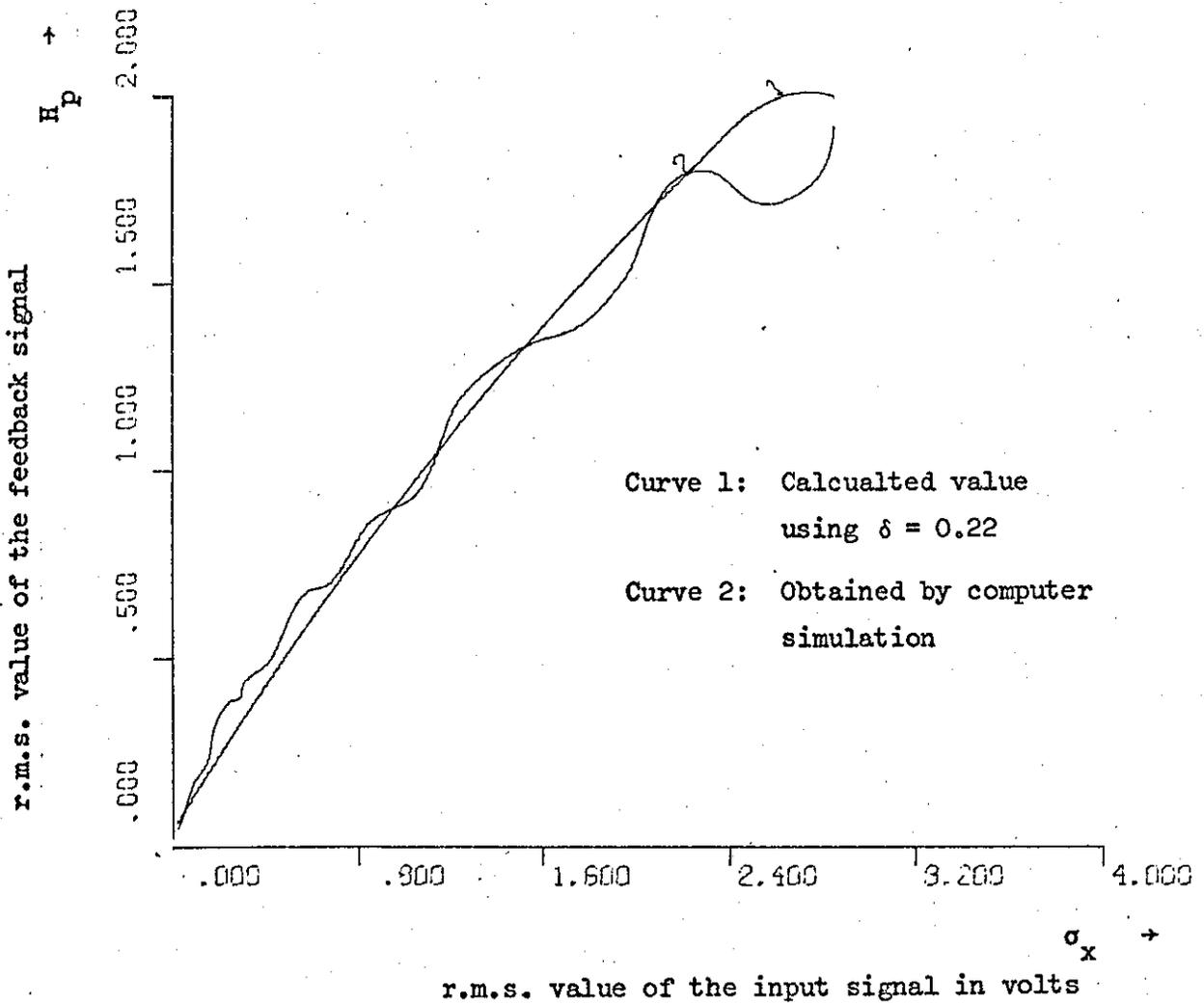
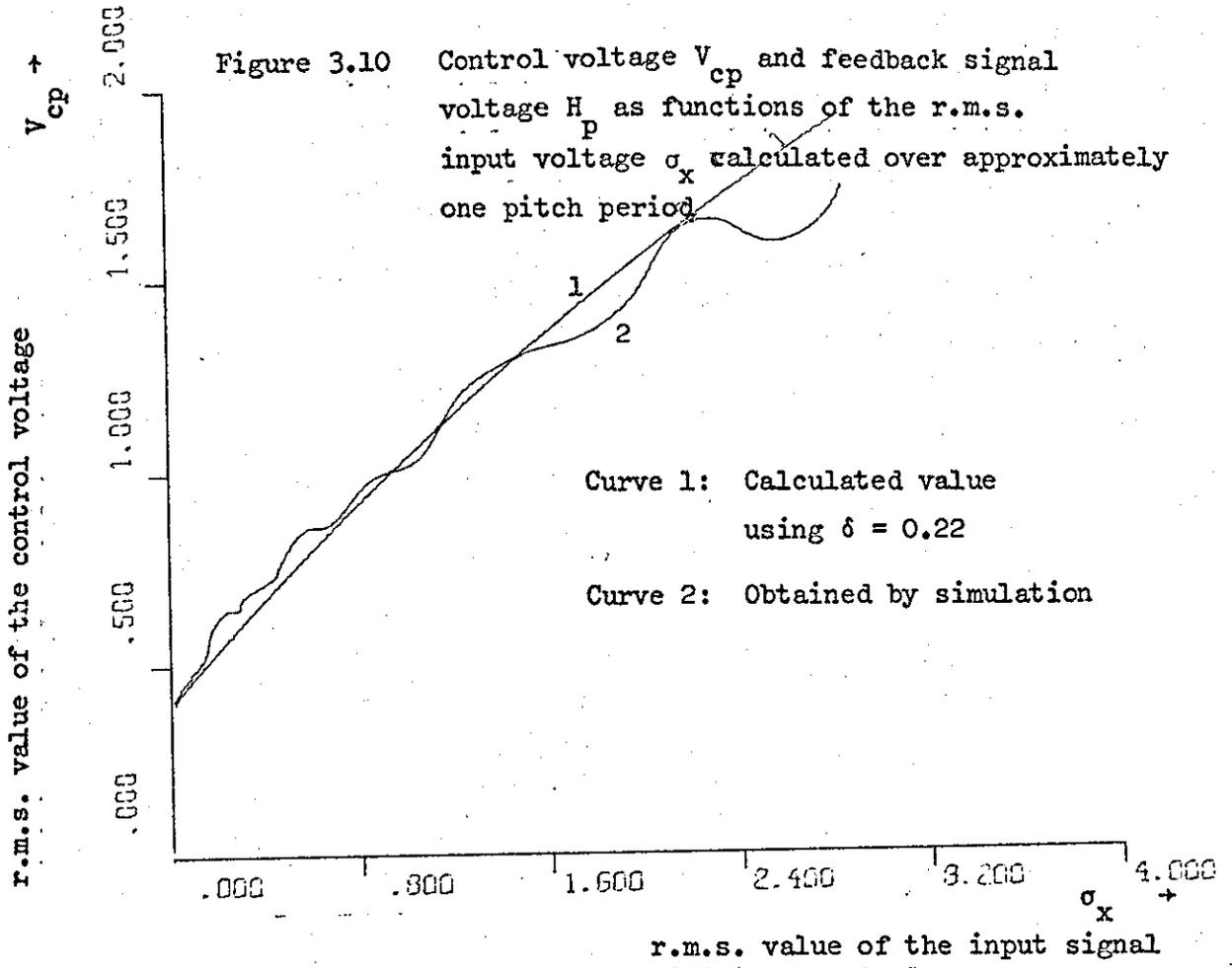
In this chapter a computer simulated model of the SCALE system - shown in Fig. 2.3 - has been compiled, debugged and various observations and measurements have been carried out. The conclusions which can be drawn from experiment (i) are:

- (a) Although the SCALE system does introduce amplitude distortion the general shape of the input waveform (and hence intelligibility)<sup>(70)</sup> is preserved. This can be seen from Fig. 3.8(e) and was later confirmed by conducting a simple listening test on a hardware model constructed for this study.
- (b) For a given sampling rate  $f_p$  the shift register introduces a constant delay equal to  $(N_e - 1)T_p$  seconds before which, in the event of a change of polarity of the input signal, delays the decision to update the feedback step height by this amount ( $(N_e - 1)T_p$  seconds). This also takes place when the integrated error changes polarity (but the input signal does not). In the latter case the effect on the decoded signal waveform is not very severe because the input and the feedback signals are almost equal and the error signal is very small. In the first case (when the input signal changes polarity first) the error signal can be large and the distortion suffered by the decoded signal can be very severe (see Fig. 3.9a). For a given sampling frequency the error decreases exponentially (slowly) for  $(N_e - 1)T_p$  seconds every time a change in polarity

Run No.	$\sigma$ The standard deviation of the Gaussian d. <sup>n</sup>	$\sigma_x$	$V_{cp}$	$H_p$
1	0.034	0.017	0.410	0.064
2	0.093	0.040	0.422	0.081
3	0.187	0.090	0.450	0.119
4	0.281	0.150	0.483	0.165
5	0.375	0.180	0.499	0.188
6	0.468	0.250	0.537	0.240
7	0.562	0.290	0.559	0.270
8	0.656	0.310	0.570	0.284
9	0.750	0.410	0.622	0.357
10	0.843	0.450	0.643	0.386
11	1.030	0.470	0.654	0.401
12	1.120	0.580	0.710	0.479
13	1.310	0.670	0.756	0.542
14	1.590	0.860	0.850	0.672
15	1.870	1.050	0.941	0.798
16	2.250	1.250	1.034	0.926
17	3.650	1.530	1.159	1.099
18	3.840	1.750	1.253	1.229
19	3.940	1.940	1.331	1.337
20	4.120	2.210	1.439	1.486
21	5.250	2.600	1.586	1.690
22	6.000	2.850	1.676	1.813
23	6.370	3.370	1.850	2.055
24	7.78	4.030	2.051	2.333
25	8.81	4.860	2.275	2.400
26	9.75	5.120	2.339	2.400
27	10.86	5.440	2.414	2.400
28	10.87	6.000	2.500	2.400

Table 3.1 Results of 28 computer runs to calculate the r.m.s. voltages of the input signal  $\sigma_x$ , the control voltage  $V_{cp}$  and the feedback step voltage  $H_p$

Figure 3.10 Control voltage  $V_{cp}$  and feedback signal voltage  $H_p$  as functions of the r.m.s. input voltage  $\sigma_x$  calculated over approximately one pitch period



of the input signal or the integrated error is experienced by the encoder and the degradation introduced into the decoded waveform, therefore, increases with the shift register length. For  $f_p = 19.2$  Kb/s  $N_e = 3$  gives the best result and any extension of the shift register above this length increases the signal degradation.

The effect of extending the shift register length above optimum is less pronounced when dealing with the medium and top frequencies of the speech baseband (800 Hz and above). This is because the error signal frequencies fall outside the output filter pass band which causes the contribution by noise to the signal power residing in the output baseband to be reduced. This can be seen by comparing Figs. 3.9a and 3.9b (see also Figure 3.9c).

- (c) The results shown in Figs. 3.8a, 3.8b, 3.8c, 3.8d and 3.6 agree with the theory of the SCALE system discussed in Chapter II. This indicates that the computer model has been simulated correctly.
- (d) Experiment (ii) has enabled the feedback step height  $H_p$ , the control voltage  $V_{cp}$  and the input voltage  $\sigma_x$  - calculated over a period  $\tau_p$  ( $= 256T_p$  which is approximately one speech pitch period) - to be related leading to the introduction of a mathematical relation between these design parameters. The significance of this result is that - because the SNR of the system before overloading is proportional to the ratio of the r.m.s. value of the input signal to the height of the feedback step; and vice versa when the overload point is exceeded (see equations (2.24b) and (2.2b)) - the system performance in terms of SNR and DR can now be determined. This can be achieved by substituting the values of  $\sigma_x$  and  $H_p$  from Table 3.1 in equations (2.24b) and (2.26b). Two FORTRAN subroutines were written specifically for this purpose and will be discussed in detail towards the end of the next chapter. One of these subroutines was named "COMPARE" and its function is to find the value of  $R_c$  from the parameters

given in equation (2.25) and the other subroutine called "SNRACL" accepts the values of  $\sigma_x$  and  $H_p$  - shown in Table 3.1 - as inputs and produces the corresponding SNR values based on performing the calculation specified by equations (2.24b) and (2.26b).

The SNR curve obtained by this method is shown in Fig. 3.11 together with the DR for a minimum acceptable SNR = 10 dB.

### 3.4 Evaluation of the constant $\delta$

~~The value of the constant  $\delta$  introduced in equation (2.12) can now be evaluated using one of the following approaches:~~

#### (a) Using equation (2.15) and Figure 3.10

Examination of Fig. 3.10 reveals that  $H_p = 1$  when  $\sigma_x$  is approximately equal to 1. Substituting these values for  $\sigma_x$  and  $H_p$  in the equation (2.15) and performing the necessary manipulations,  $\delta$  can be calculated if  $G$ ,  $Z_h$  and  $H_{\min}$  are known. So:

$$\begin{aligned} \delta &= \frac{1}{\sigma_x} \ln \left\{ \frac{GZ_h}{GZ_h + H_{\min} - H_p} \right\} \\ &= \ln \left\{ \frac{1.4 \times 3.8}{1.4 \times 3.8 + 0.05 - 1} \right\} \quad 0.203. \end{aligned}$$

#### (b) Using equations (2.23), (2.24), (2.25) and Figure 3.11

Recalling equations (2.24) and (2.26) the SNR in dB is:

$$\begin{aligned} \text{SNR} &= 10(\log R_c) + 20 \log \left( \frac{\sigma_x}{GZ_h (1 - e^{-\delta \sigma_x}) + H_{\min}} \right) & \sigma_x \leq H_p \\ &= 10(\log R_c) + 20 \log \left( \frac{GZ_h (1 - e^{-\delta \sigma_x}) + H_{\min}}{\sigma_x} \right) & \sigma_x > H_p \end{aligned}$$

where  $R_c$  is defined by equation (2.24).

The Dynamic Range DR (for minimum acceptable SNR of 10 dB) is  $\approx 32$  dB,  
 $f_p = 19$  Kb/s,  $N_e = 3$ . The curve was obtained by computer simulation.

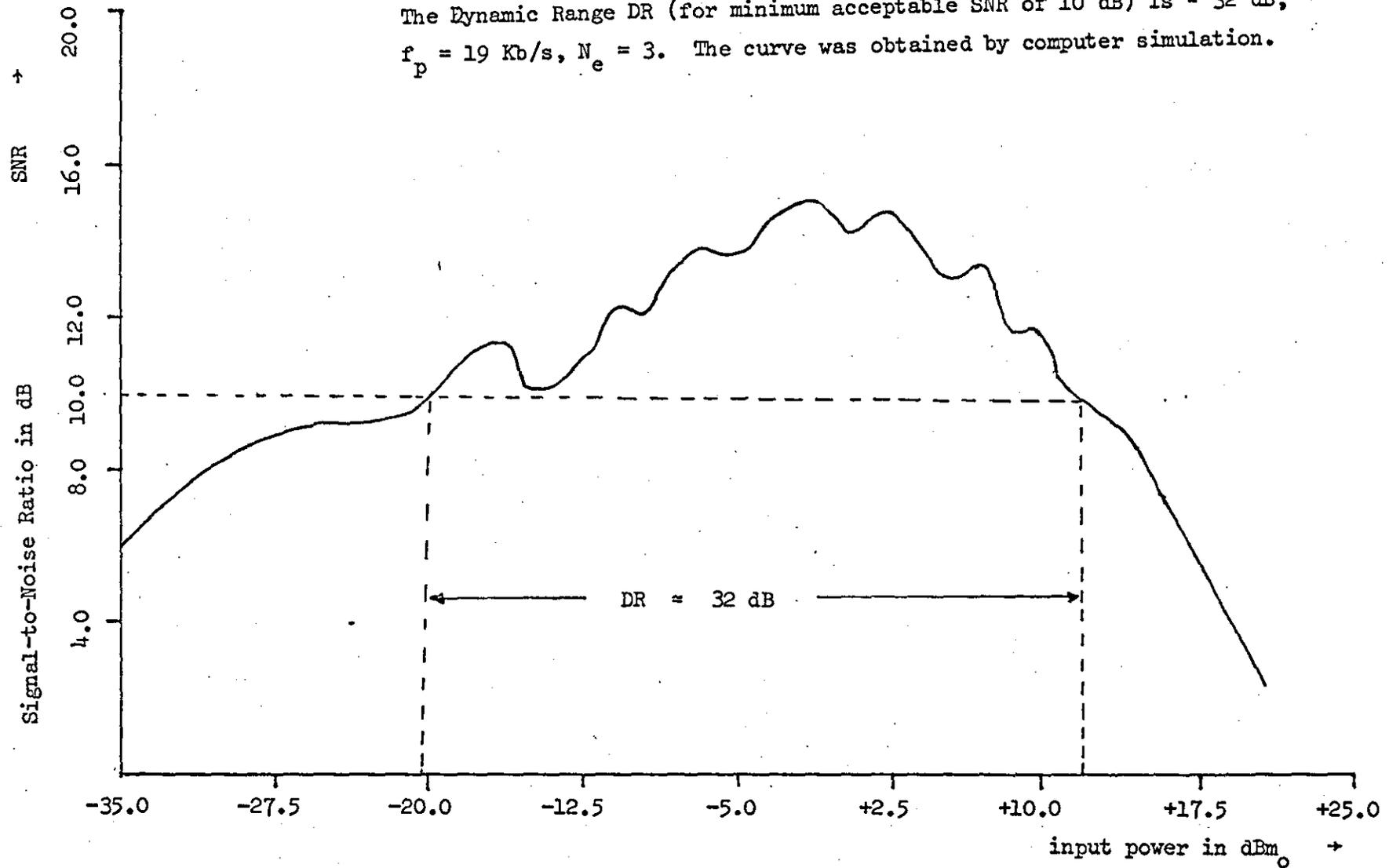


Figure 3.11 Signal-to-Noise Ratio SNR as a function of the input power in dBm.

Now using these equations and the procedure shown in Fig. 3.12a the value of  $\delta$  can be estimated. The iteration procedure continues until the calculated SNR curve and that of Fig. 3.11 are close enough to be considered coincident with one another. This method can be tedious if it is to be performed manually. For this reason a computer subroutine (SUBROUTINE SNRACL) was compiled and used to do the calculation and plotting of the SNR curve while another subroutine (SUBROUTINE FITTER) supplies it with  $GZ_h(1 - e^{-\delta\sigma}x)$ . Details of these subroutines are given in Chapter IV and Appendix B. Fig. 3.12b shows that the value of  $\delta$  obtained using this method is approximately 0.21.

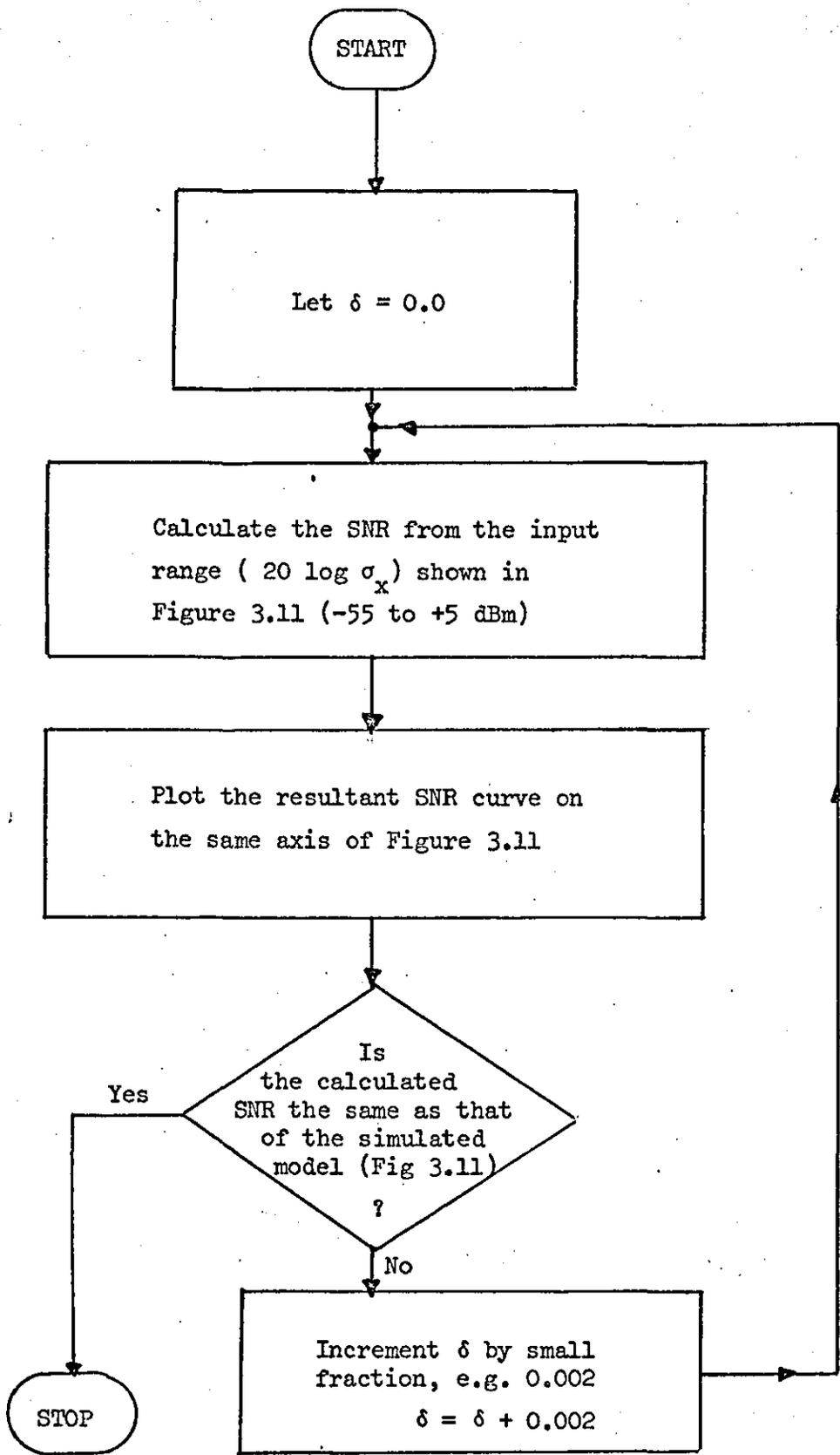


Figure 3.12a Evaluation of the parameter  $\delta$  employing iterative techniques

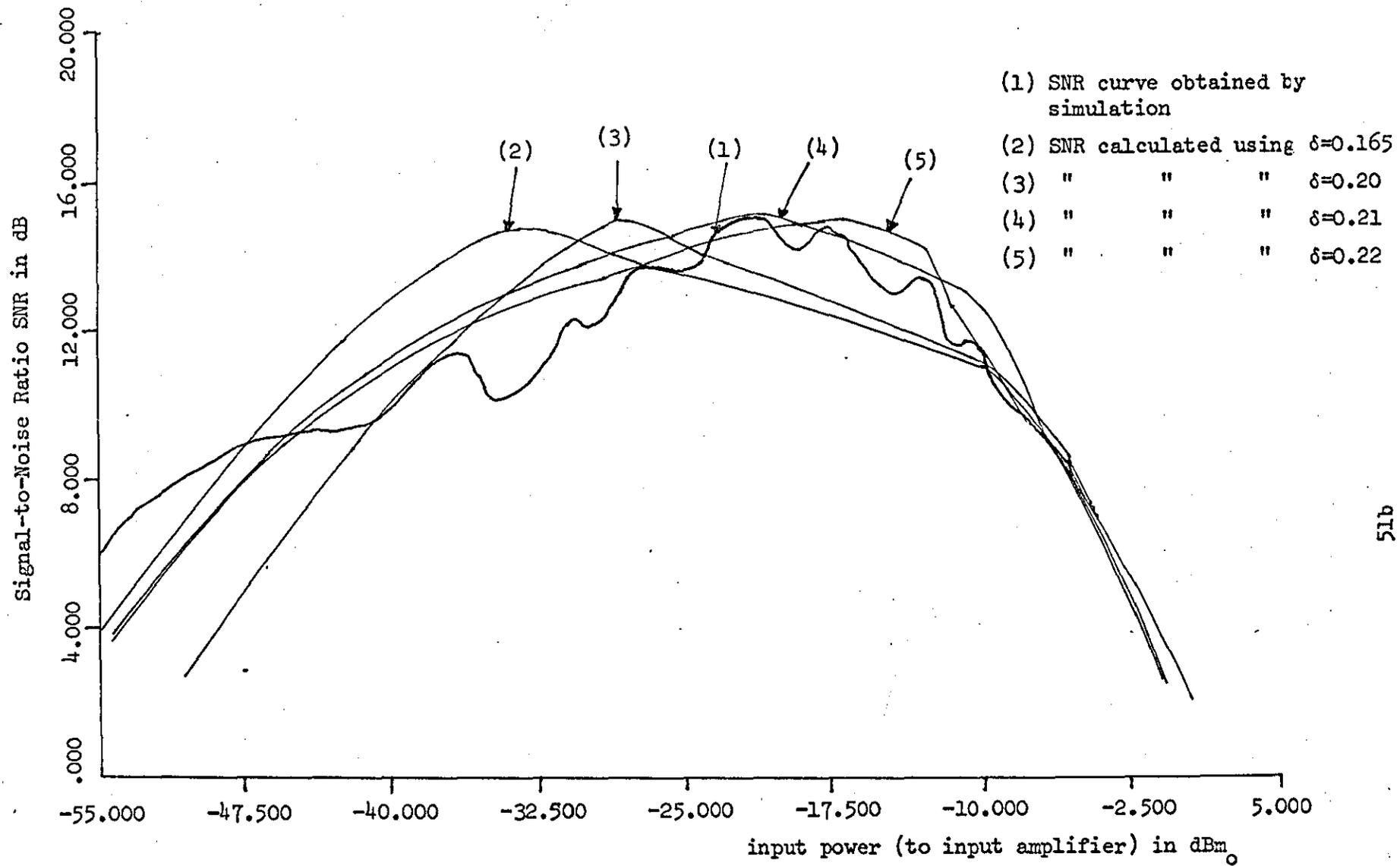


Figure 3.12b Evaluation of  $\delta$  using iteration techniques

## CHAPTER IV

### HARDWARE MODEL, PERFORMANCE MEASUREMENTS AND PERFORMANCE CALCULATIONS

In this chapter, the hardware implementation of the SCALE system shown in Fig. 2.3 is discussed. Performance observations and measurements are made and comparison of the results with those obtained by computer simulation is carried out. Finally, comparison between the theory, experimental and computer simulation results of the SCALE system is made. Procedures for evaluating, experimentally, the unknown constants  $K_c$  and  $\delta$  introduced in Chapter II are also presented.

#### 4.1 PRACTICAL IMPLEMENTATION OF THE SCALE SYSTEM

The realization procedure of the SCALE system shown in Fig. 2.3 can be divided into three basic phases: circuit synthesis, construction and testing.

##### 4.1.1 Circuit synthesis and description

The objective of this phase is to translate the basic equations and logical specifications - discussed in Chapter II - into circuit diagrams with determined circuit parameters' values. For the basic SCALE this has already been carried out by Wilkinson<sup>(17)</sup> who has synthesised and constructed the system shown in Fig. 2.1. Wilkinson circuits for the realization of the input and output filters, comparator, pulse high modulator and the subtractor are all applicable to the system under investigation (Fig. 2.3). However, the SR, control logic and syllabic integrator employed by Wilkinson are not adequate for the present experiment; because in the latter case the SR length  $N_e$  must be variable from 2 to 6 bits (c.f. Wilkinson's SR which is fixed to 3 bits). The

control logic must be capable of selecting any one of the effective SR lengths. The syllabic integrator time constant must also be variable. These items will be discussed in detail later in this section; but first the circuit diagrams presented by Wilkinson are reviewed.

(a) The input filter

The input filter consists of cascaded high and low pass Butterworth filters which are of the active type and both are of the third order. The theory of active filters has been widely discussed in the literature (61, 62) and is reviewed in Appendix D. Unlike passive filters, active filters can be built more easily, requiring no special construction techniques. These advantages in favour of active filters stem from the fact that no inductors are used, which means that they are free from the undesirable parasitic coupling and non-linear inductor effects common to most passive networks. The absence of inductors also makes the tuning simpler and the cost of construction lower. (The filter used in this experiment was constructed at a total cost of = £1.50, for the components).

The design employs the fixed-gain amplifier approach (see Appendix D) in the form of emitter follower which has the desirable high input and low output impedances.

The filter is preceded and followed by amplifiers with a total gain of = 22.0 dB to compensate for various circuit losses. The circuit diagram and the frequency and phase characteristics of the filter are given in Fig. 4.1.

(b) The comparator, pulse-height modulator and subtracting circuits

The comparator, PHM, subtractor and the forward integrator  $C_1R_1$  have been adequately described in reference (17) and their circuit diagrams are given in Fig. 4.2. Components which are different from those used in reference (17) are indicated by \* . These components perform

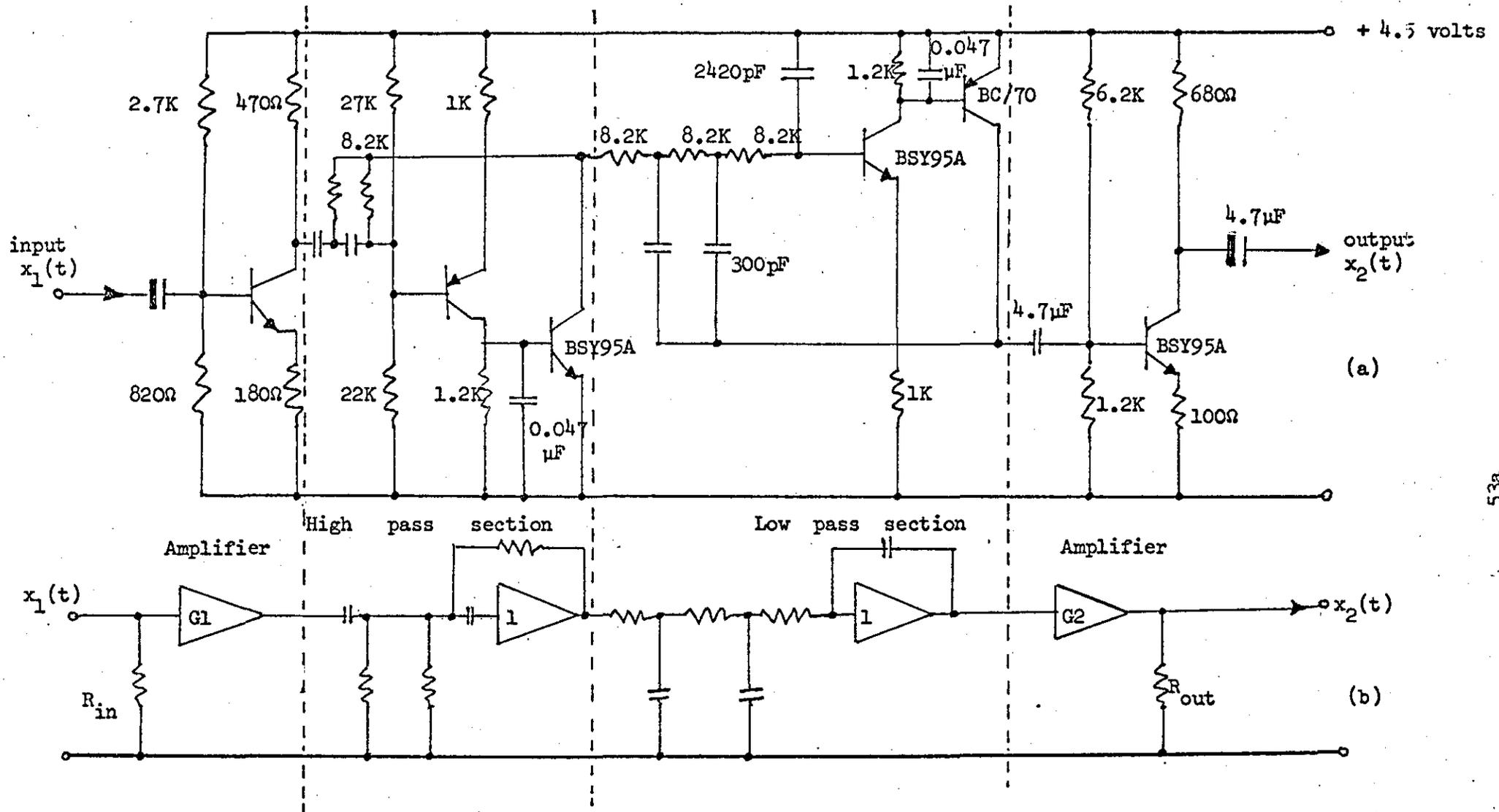


Figure 4.1 The SCALE system input filter

(a) Circuit diagram using transistors and passive components, (b) Equivalent circuit

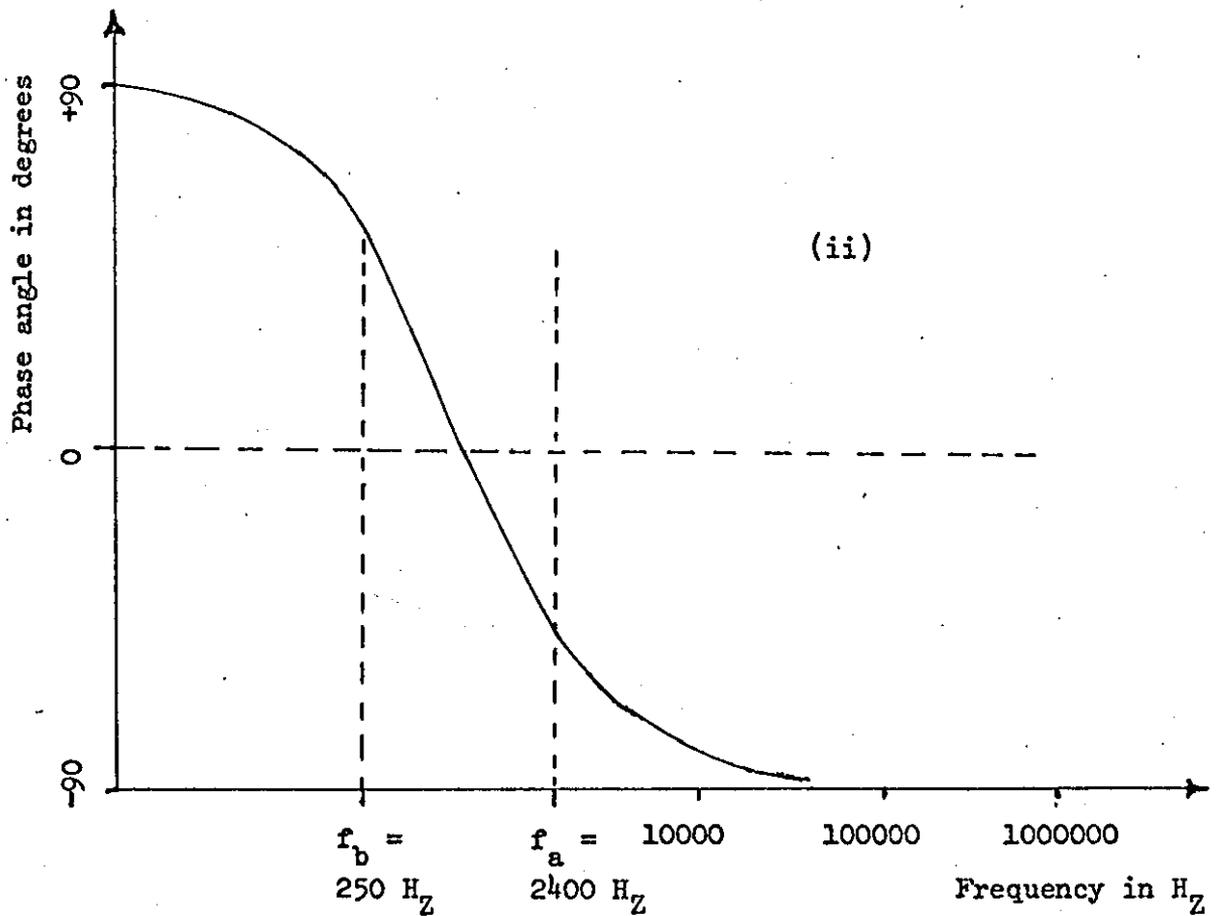
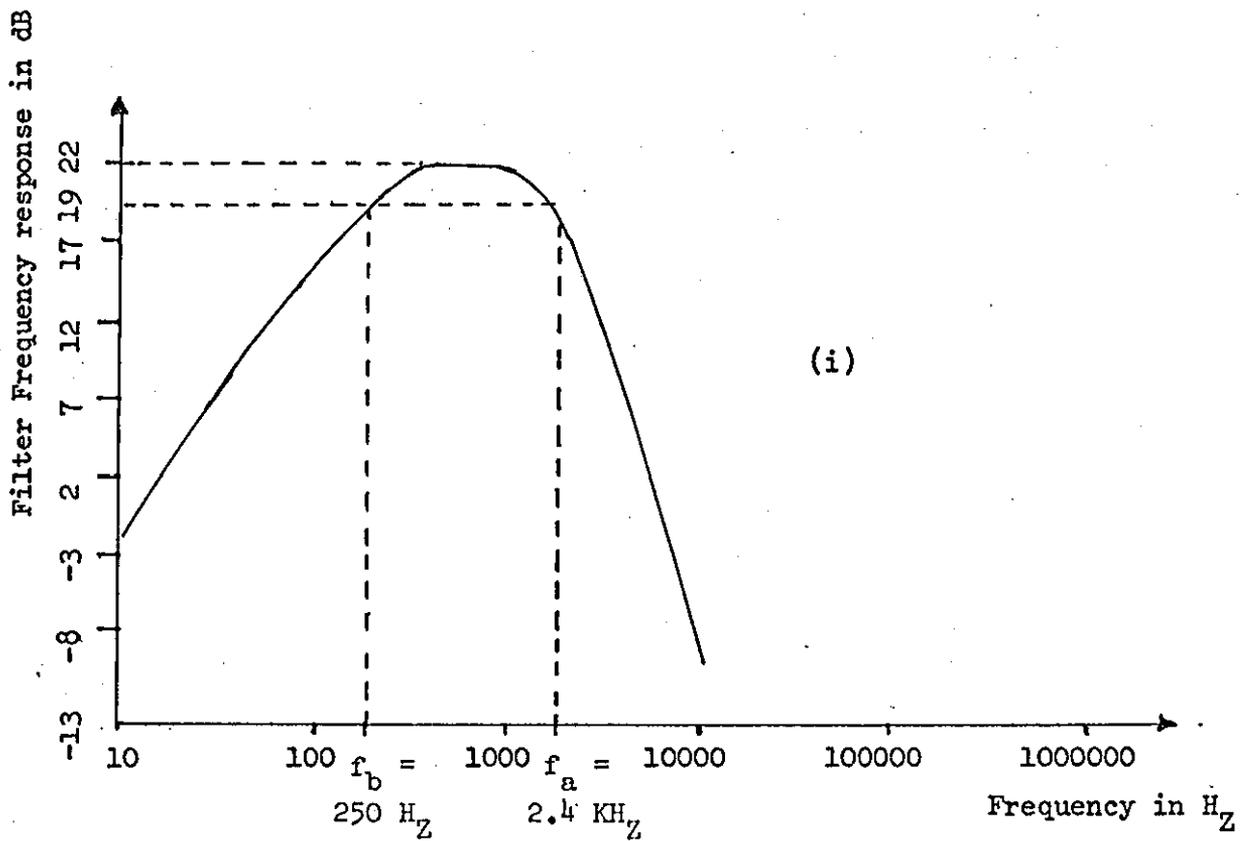


Figure 4.1(c) The input filter characteristics

(i) Frequency response

(ii) Phase response

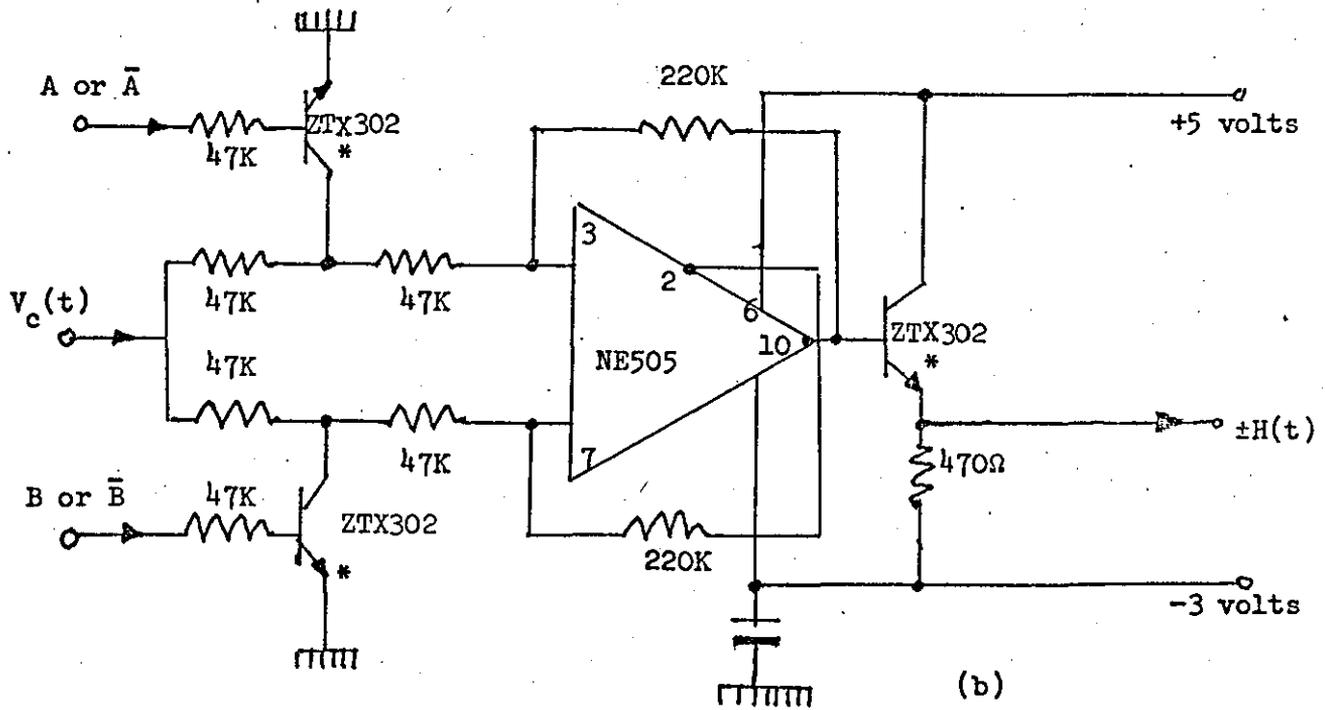
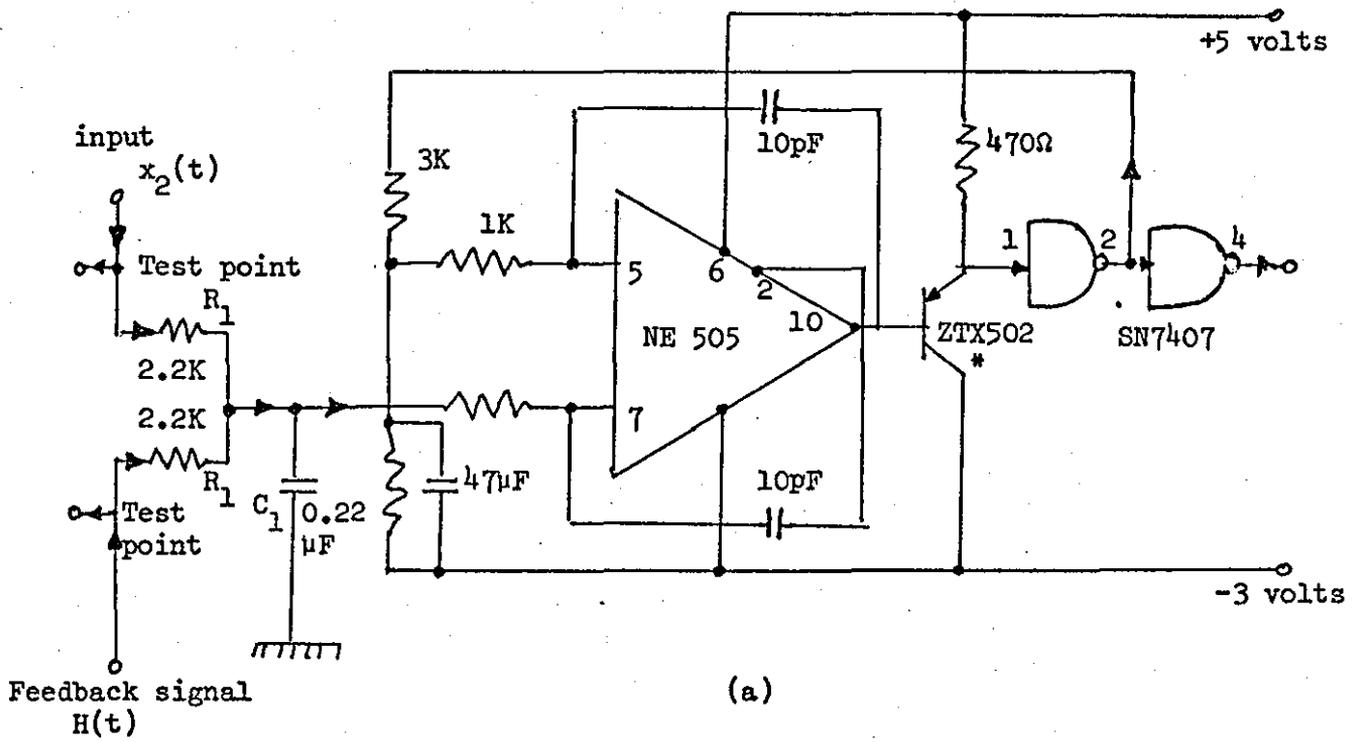


Figure 4.2 Circuit diagram of the analogue section of the SCALE system  
 (a) Voltage comparator  
 (b) Pulse height modulator

exactly the same functions as those used by Wilkinson but are made by a different manufacturer.

(c) Decoder output filter

The decoder output filter comprises a twin-T network, which has a null at  $\approx 4.8 \text{ KH}_z$ , and a third order low-pass Butterworth filter. This combination produces an overall characteristic shown in Fig. 4.3(b)<sup>(17)</sup>. The circuit diagram is given in Fig. 4.3(a).

(d) The decision memory and control logic

The function of this part of the CODEC is to enable the present and previous 5  $L(t)$  logical values to be continuously monitored and when at least 3 of these  $L(t)$  digits are identical a voltage pulse is generated to identify the channel pattern being observed. The process to be implemented is defined by equation (2.8). This operation was realized in 3 steps.

- (i) The present and previous 5  $L(t)$  values are stored in a 6-bit serial SR consisting of 6 D-Flip-Flops in which the first stage is fed from the output of the comparator. The SR is clocked at the required bit rate  $f_p$ . The circuit diagram of this arrangement is given in Fig. 4.4(a). It consists of 3 blocks of Texas TTL74 series<sup>(63)</sup> (SN7474 Flip-flops) connected as shown in the diagram.
- (ii) The output of the SR feeds the Detection logic DL which realizes equation (2.8). Its output  $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $\lambda$  and  $\phi$  depend respectively on the outputs from 2, 3, 4, 5 or 6 flip-flops (e.g. if  $ABCD + \overline{ABCD}$  is true then  $\lambda = \text{logical one}$  irrespective of the output of the last two flip-flops in the SR).

The circuit was realized using Texas TTL74 series NAND/NOR logic gates and invertors as shown in Fig. 4.4(b).

- (iii) The effective length of the SR  $N_e$  (which is also the number of  $L(t)$  digits on which  $Z$  depends) can be selected by masking the



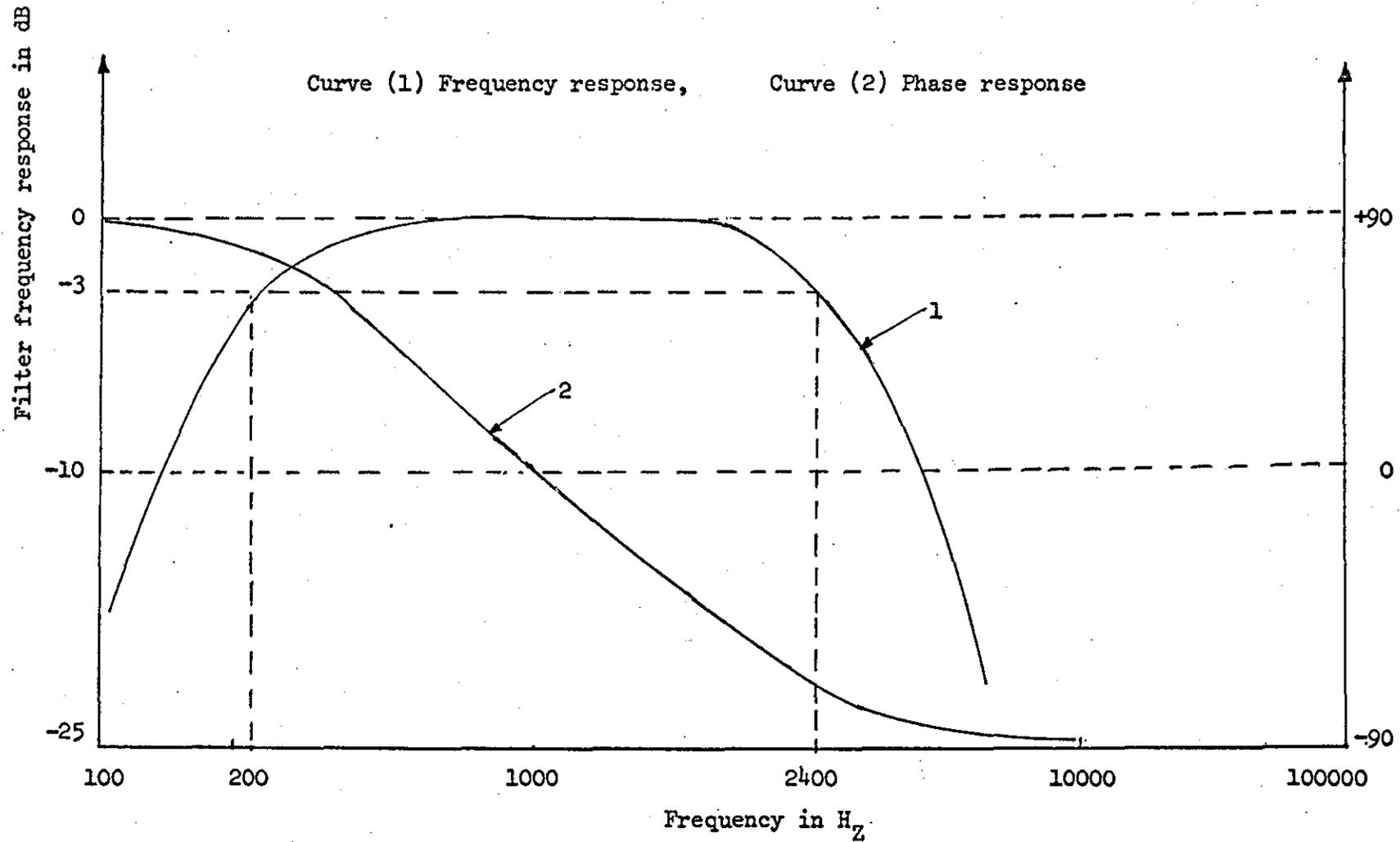


Figure 4.3(c) The SCALE decoder output filter characteristic

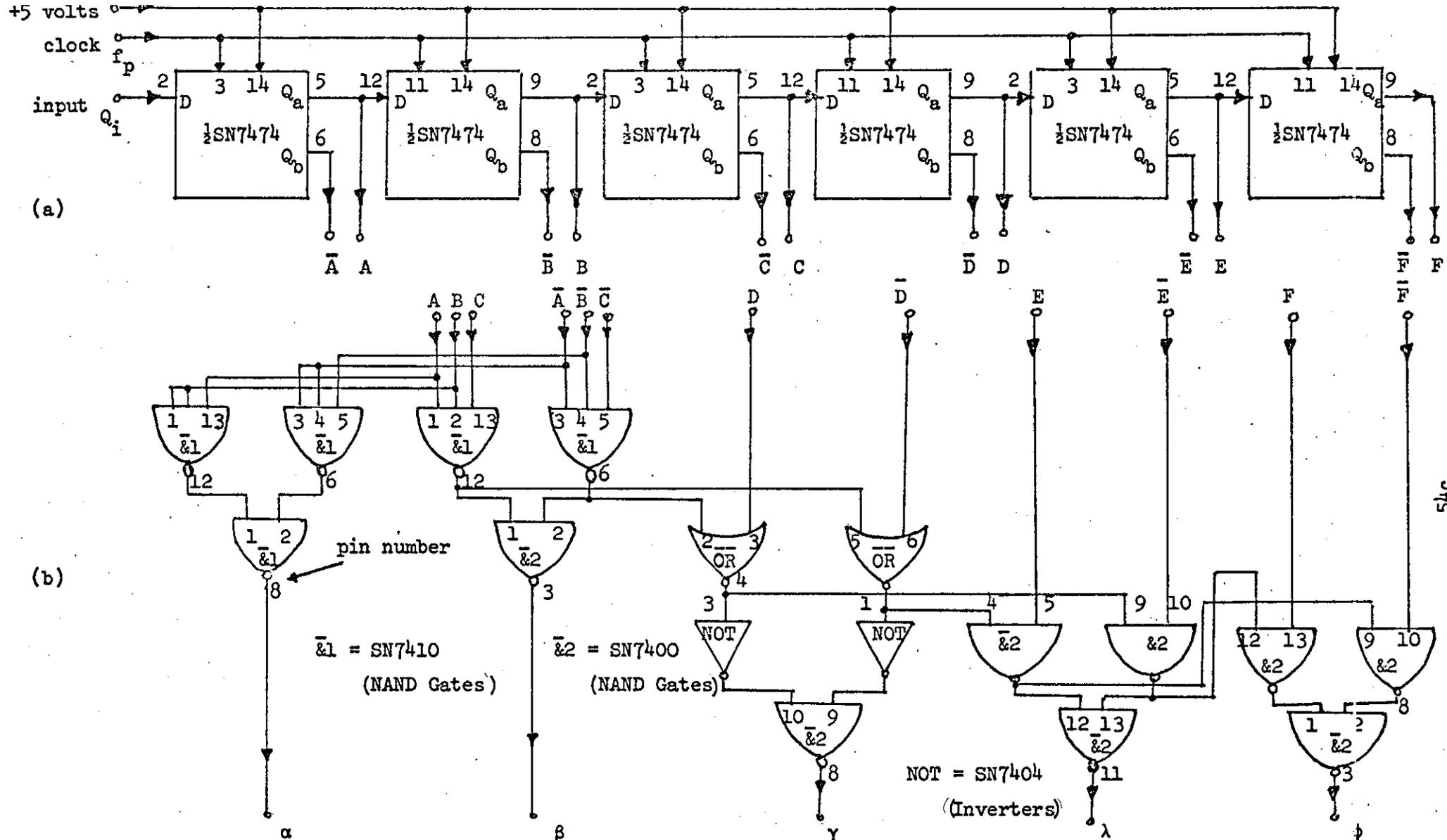


Figure 4.4 The digital part of the SCALE system (a) The 6-bit serial shift register (b) The detection logic

effect the patterns resulting from SR length less than  $N_e$ . (For example, if it is required to use 4 bit SR then  $\alpha$  and  $\beta$  must be ignored and  $\gamma$  is used to switch Z between  $Z_h$  and  $Z_e$ .) This was achieved by a combination of electronic gates and mechanical switches as shown in Fig. 4.4(c).

(e) The syllabic integrator

To enable the syllabic integrator time constant to be varied between  $\approx 10$  and  $20$  m.s.  $C_2R_2$  was realized by a fixed value of  $R_2$   $50$  K $\Omega$  and a bank of three capacitors in parallel  $T_2 = C_2R_2$  can be selected by switching in one of the three capacitor combinations using the switches  $S_{15}$  and  $S_{20}$  as shown in Fig. 4.4(d). For example, when  $S_{15}$  is closed and  $S_{20}$  is open,  $C_2 = 15\mu\text{F}$  and  $T_2 \approx 15$  m.s.

4.1.2 Construction of the CODEC

The encoder and the decoder were constructed on two separate 17 inch plain vero boards. The encoder board consists of the input filter, subtractor, forward integrator, comparator, SR, DL, SL gates, syllabic integrator and PHM. The decoder board consists of a SR, DL, SL,  $C_2R_2$ , PHM and output filter. To facilitate replacement of faulty integrated circuits, 14-pin sockets were fixed to the boards (using general purpose adhesive) and used as bases for the IC chips.

The encoder and decoder boards were enclosed in boxes which were assembled from commercially available 18 S.W.G. aluminium panels and 20 S.W.G. steel covers, side plates and chassis. The vero boards were separated from the chassis by rubber spacers to avoid short circuiting and the input, output, clocks, and various test and power feed points were made accessible from the front panels through suitable sockets and switches. Miniturized two-pole switches were employed to realize the switches  $N_{e_2}$ ,  $N_{e_3}$ ,  $N_{e_4}$  and  $N_{e_5}$  of Fig. 4.4(c). Miniturized lamps were

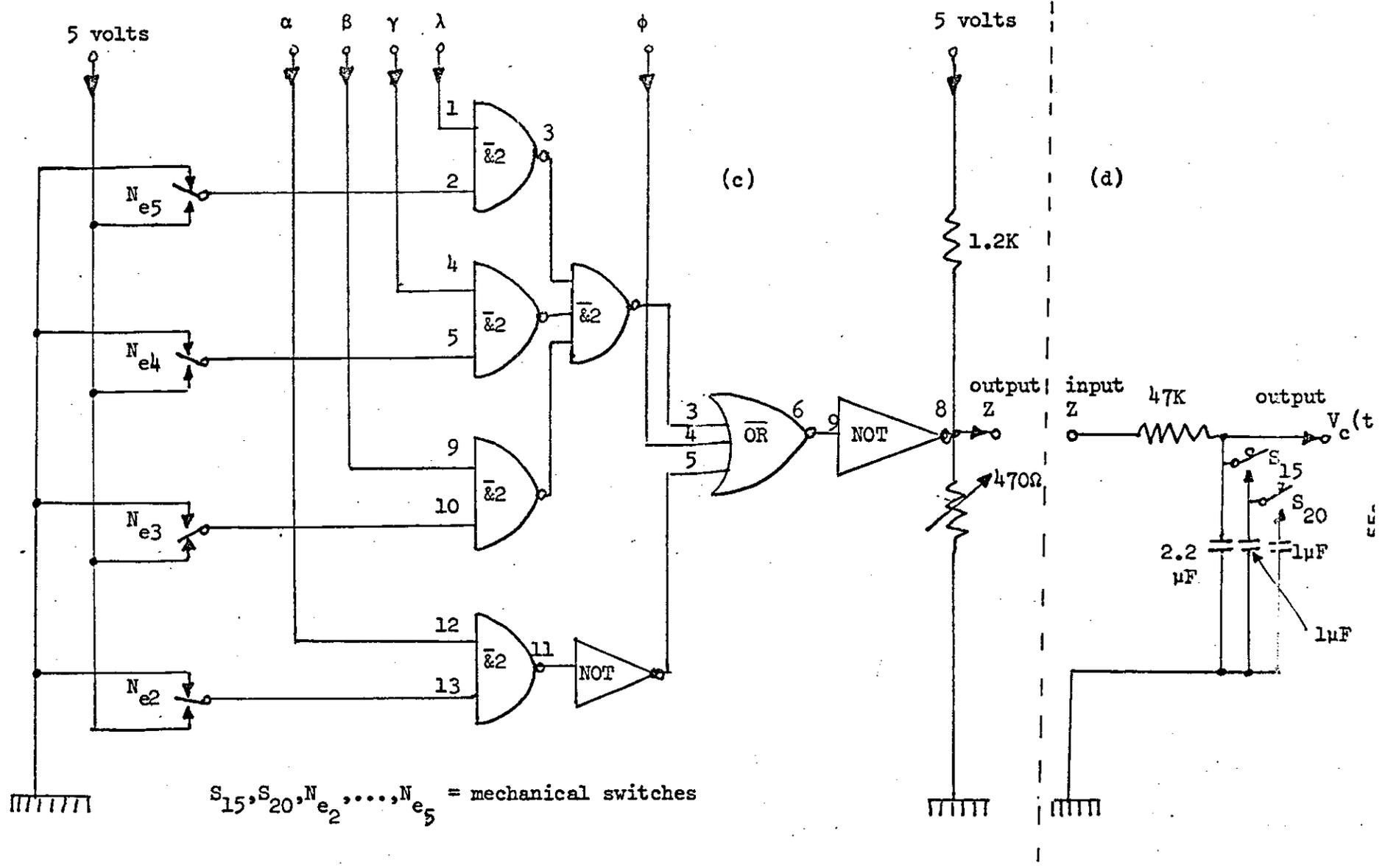
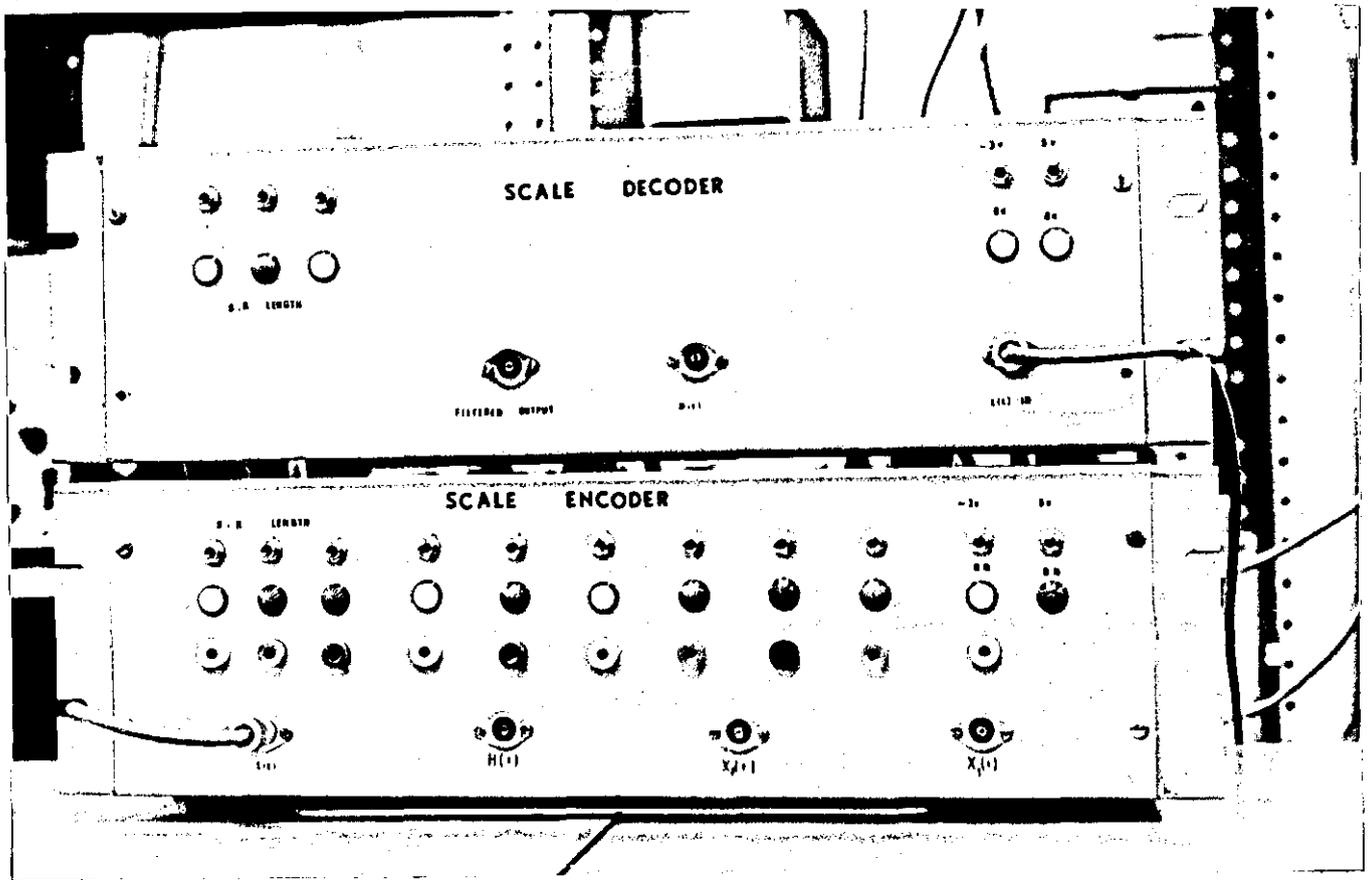


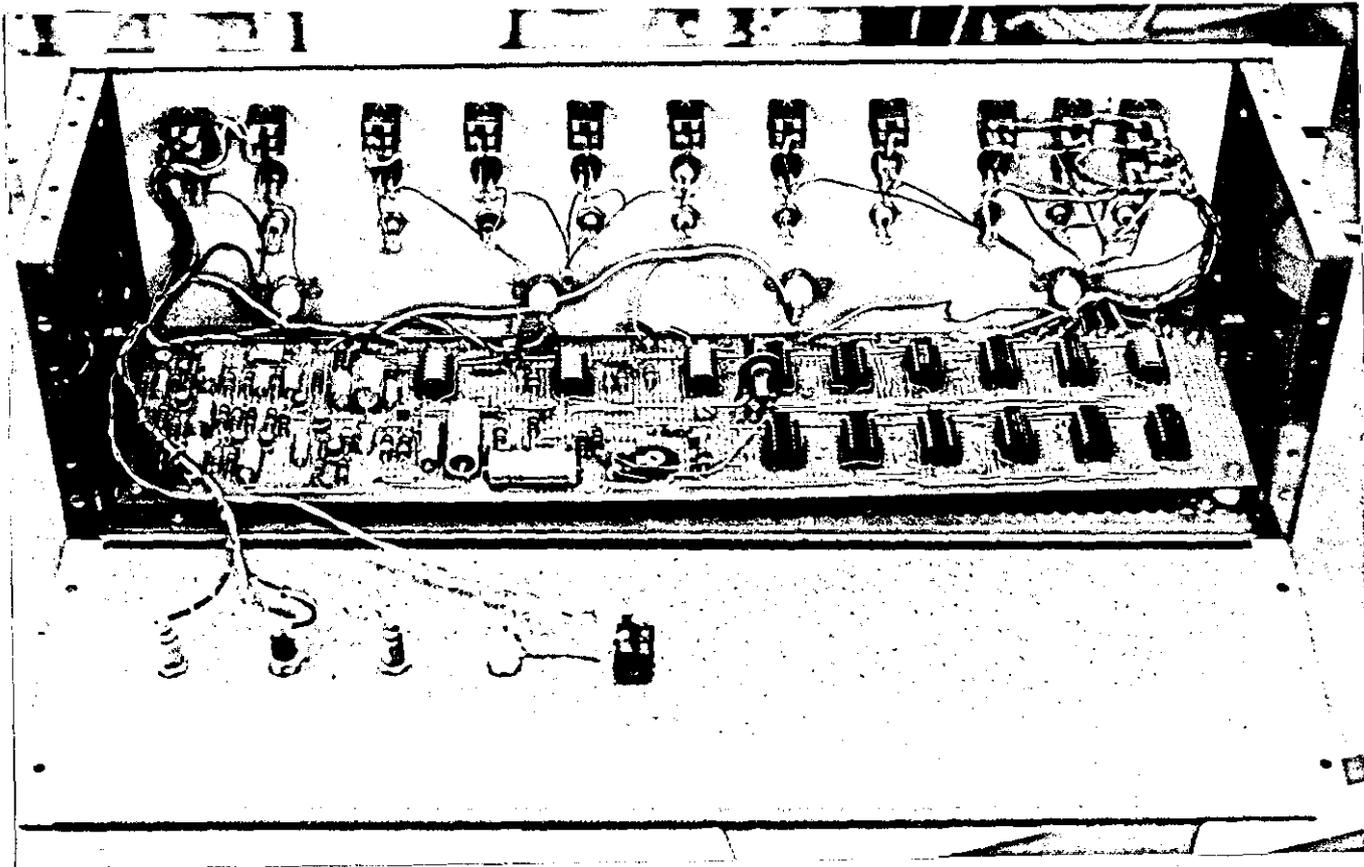
Figure 4.4 The interface of the SCALE system (c) Circuit diagram of the selection logic (d) The syllabic integrator



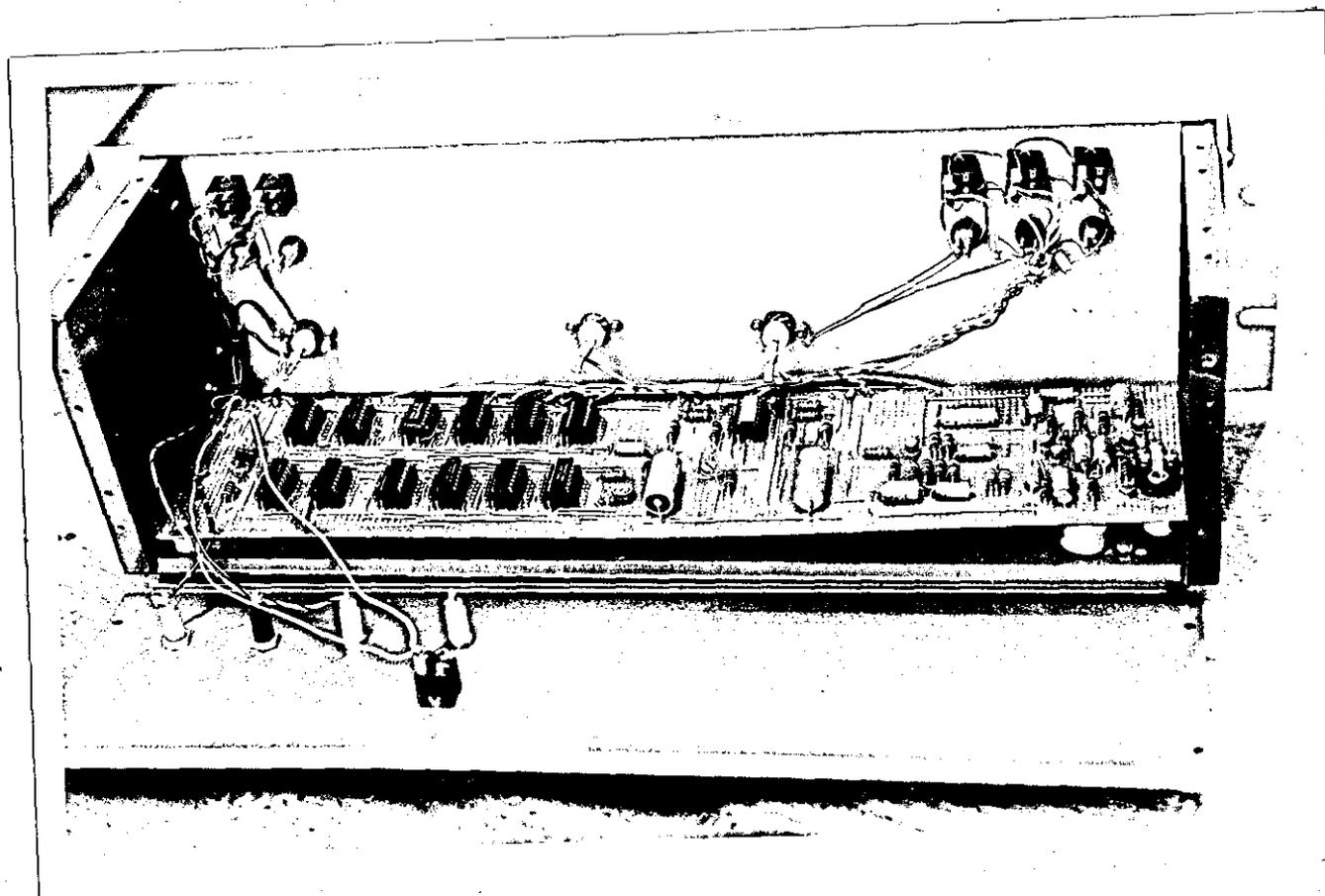
(a)

Figure 4.5 Photographs of the SCALE system hardware model

- (a) Photograph of the front panel
- (b) Encoder circuit layout (see next page)
- (c) Decoder circuit layout (see next page)



(b)



(c)

also employed as indicators to indicate the condition of the various switches.

Side brackets and rubber feet were fitted to the encoder and decoder boxes to enable them to either be fitted into a standard 19 inch rack or to sit on the bench and connected to test equipment. Connections between the encoder, decoder and the test equipment were made by short lengths of screened cables. Photographs of the chassis boards and the front panels are shown in Fig. 4.5.

#### 4.1.3 Testing and adjustment of the CODEC system

The objective of this phase in the experiment is to ensure that the system which has been constructed does satisfy the requirements outlined in Chapter II (see section 2.2.1, Fig. 2.3 and Table 2.1). To achieve this the following tests and adjustments were performed:

##### (i) Testing of the comparator, PHM and the control logic

To ensure that the characteristics of the comparator and the PHM are symmetrical about their reference voltages, a low frequency ( $\approx 300\text{Hz}$ ) sinusoidal signal - of amplitude 5 volts peak - was applied simultaneously to the X-input of the cathode ray oscilloscope, CRT, and to the input of the circuit under test (comparator or PHM). The output circuit was then connected to the Y-input of CRT and the resultant input-versus-output curves were displayed on the screen. The negative supply voltage was then adjusted carefully for minimum asymmetry and hysteresis. The arrangement and the optimized characteristics are shown in Fig. 4.6.

The correct functioning of the SR and the control logic were ensured by feeding a  $500\text{Hz}$  5 volts peak square wave signal into the input of the SR and observing that the correct shifting action was performed and that the  $Z_h$  pulses became systematically narrower as the SR length  $N_e$  was increased.

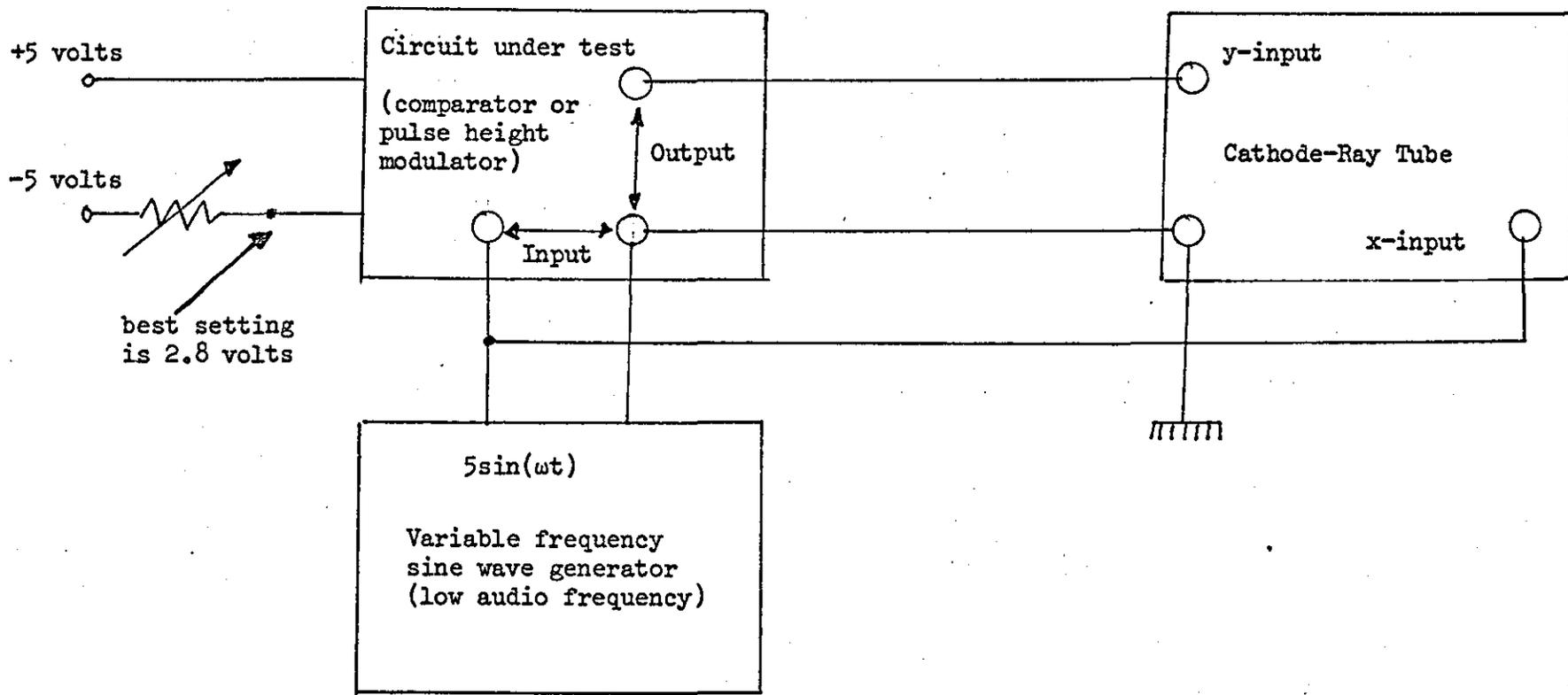


Figure 4.6a Testing and adjustment of the pulse-height modulator and comparator

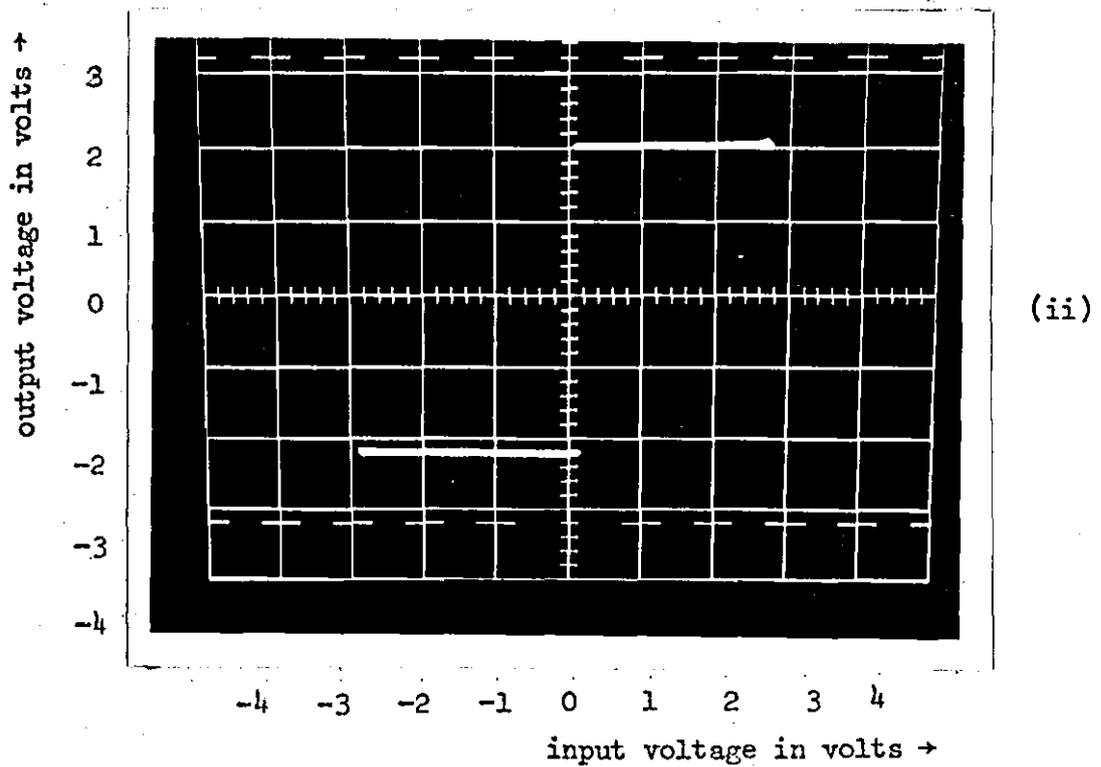
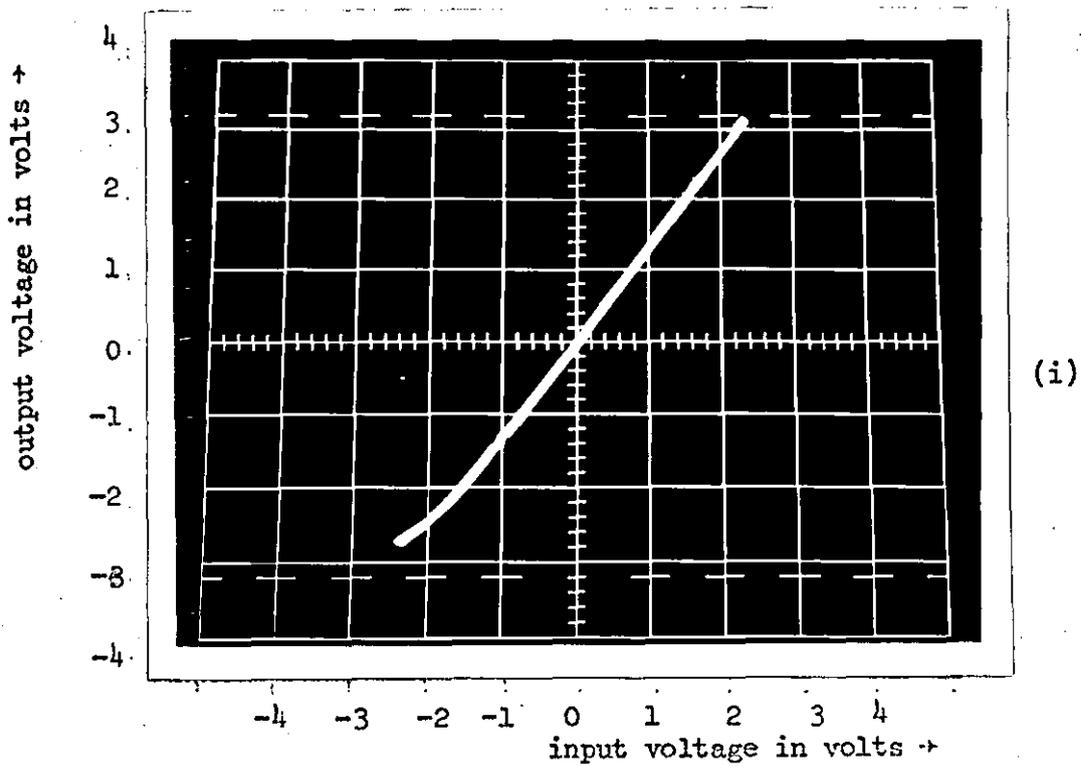


Figure 4.6b Testing and adjustment of the PHM and the comparator

- (i) Photograph of the PHM characteristic taken from the CRT screen for the best setting of the negative supply voltage.
- (ii) Photograph of the optimized characteristic of the comparator.

A low frequency ( $100\text{H}_z$ ) square of 5V amplitude was applied to the input of  $C_2R_2$  and a triangular waveform of the correct amplitude and frequency was produced at the input of the PHM.

#### 4.2 INVESTIGATION INTO THE BEHAVIOUR OF THE CODEC UNDER SINUSOIDAL INPUTS

The aim here is to report the observations of waveforms in the hardware model when sinusoidal input signals of various frequencies and amplitudes are applied. An attempt is then made to relate these observations to the theory of the SCALE system discussed in Chapter II and to the performance of the simulated model described in Chapter III.

##### 4.2.1 Observations at the encoder end

A sinusoidal source of variable frequency and amplitude was connected to the input of the encoder (which was clocked at  $f_p = 19.2 \text{ Kb/s}$ ) and the various waveforms were displayed on a CRT (of  $10\text{MH}_z$  bandwidth). It can be seen from Fig. 4.7(a) that for  $400\text{H}_z$  sinusoidal input of amplitude 1 volt peak the feedback signal amplitude varies smoothly with that of the input waveform. Furthermore, the error signal has large high frequency content. This is reflected in the much smaller value of its integrated value (i.e. the low frequency energy content passed by  $C_1R_1$  - which is essentially a low-pass filter - is small).

It can also be seen from Fig. 4.7(b) that as the feedback signal is increased (by increasing the input amplitude to 1.5 volts), the noise falling in the baseband is also increased as indicated by the error waveform, the integrated error waveform and the corresponding feedback signal.

It was also observed that as the input amplitude approaches  $H_{\text{max}}$  (2.4 volts) the variations in the feedback step height diminish. Further increase of the input amplitude does not have any effect on the feedback step height. This was reflected in the large error signal

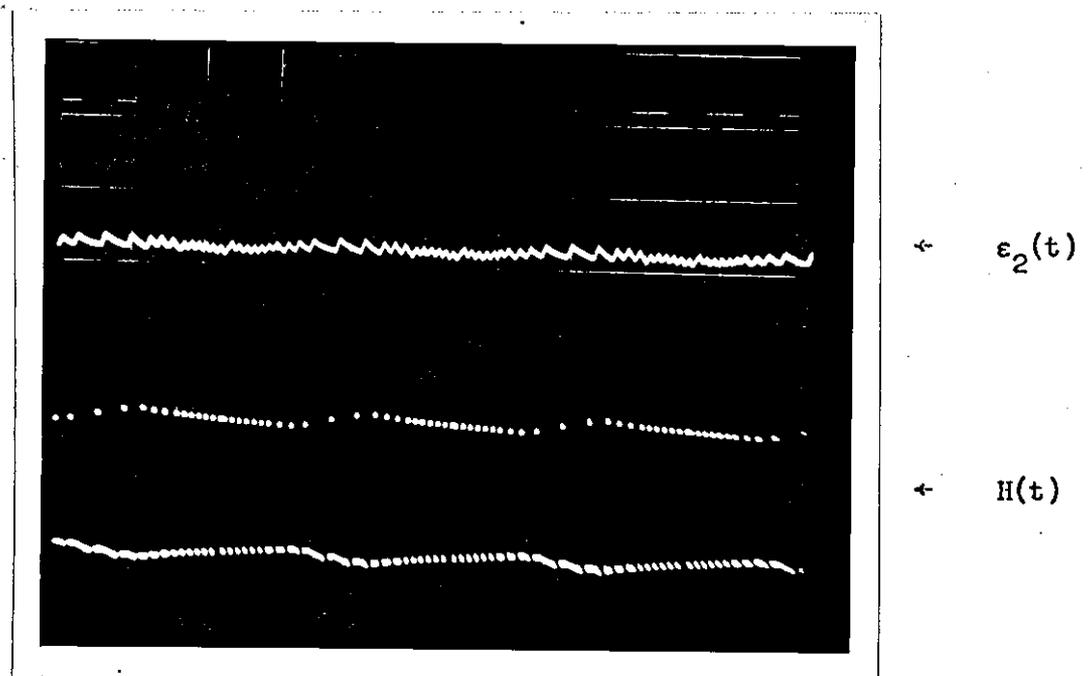
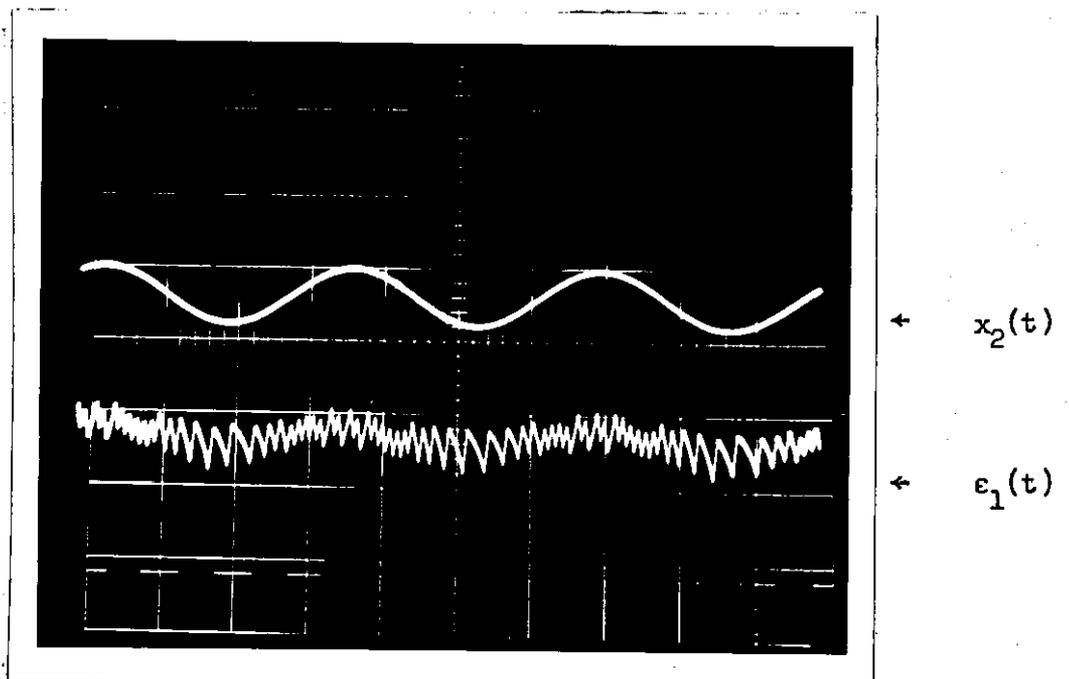


Figure 4.7a The waveform observation experiments at the encoder end

$x_2(t)$  = The input signal to the hybrid SCALE encoder:  
a sinusoid of frequency  $f_s = 400\text{Hz}$  and  
amplitude  $A(j) = 1.0$  volt.

$\epsilon_1(t)$  = The error signal; i.e. the difference between  
 $x_2(t)$  and the feedback signal  $H(t)$ .

$\epsilon_2(t)$  = The integrated error observed at the output of  
the forward integrator

$H(t)$  = The feedback signal corresponding to  $x_2(t)$ .  
 $f_p = 19.2$  Kb/s.

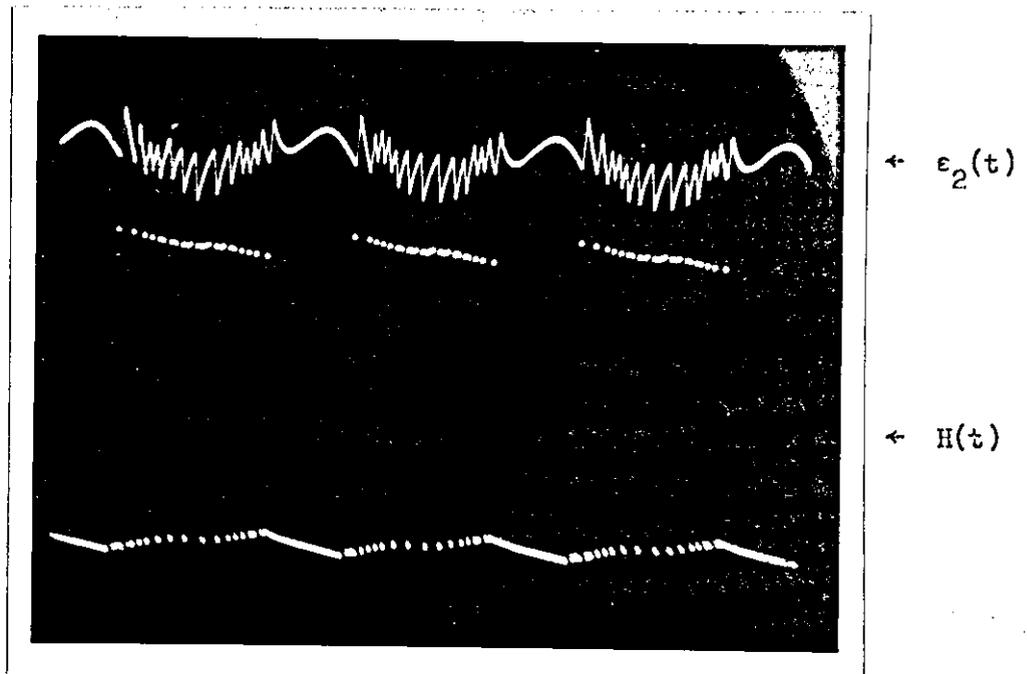
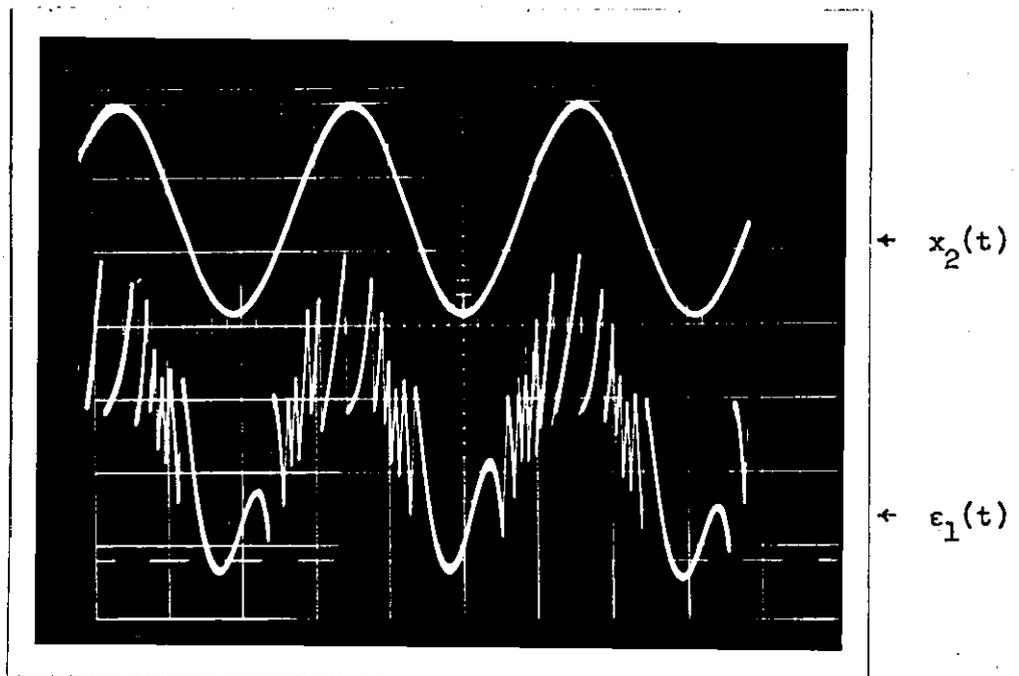


Figure 4.7b The waveform observation experiment (at the encoder end)  
 The symbols ( $x_2(t)$ ,  $\epsilon_1(t)$ , etc.) have the same meaning as in Figure 4.7a except that the amplitude of the input signal  $x_2(t)$  has been increased from 1.0 volt to 1.5 volt.  $\epsilon_1(t)$ ,  $\epsilon_2(t)$  and  $H(t)$  are respectively the corresponding error, the integrated error and the feedback signal.  $f_p = 19.2$  Kb/s.

and its spectral energy distribution in which the low frequency components were predominant.

#### 4.2.2 Observations made at the decoder end

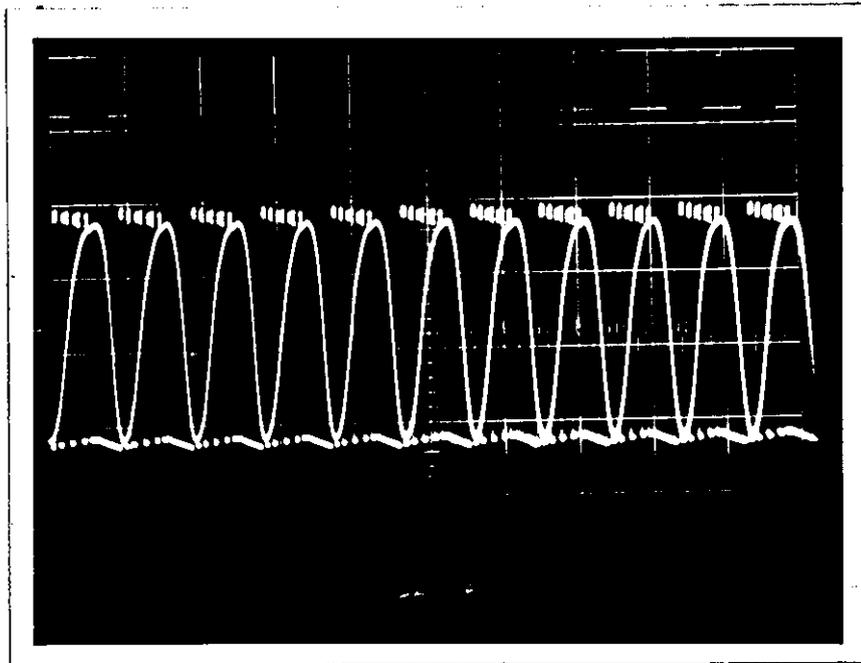
Figure 4.8(a) shows the step height  $H$  and its low-pass content  $\hat{X}_2(t)$  which were reconstructed by the decoder at a sampling rate of 19.2 Kb/s. The input signal to the encoder is a sinusoid with frequency  $f_s = 400\text{Hz}$  and amplitude of 2 volts. For the improvement shown in Fig. 4.8(b) the sampling frequency  $f_p$  had to be increased to 56 Kb/s.

When the encoder input signal frequency was increased to  $1000\text{Hz}$  reasonable reproduction was obtained at  $f_p = 38.4$  Kb/s and further improvement was achieved when  $f_p$  was increased to 76.8 Kb/s. The waveforms for the latter two cases are respectively shown in Figs. 4.8(c) and 4.8(d).

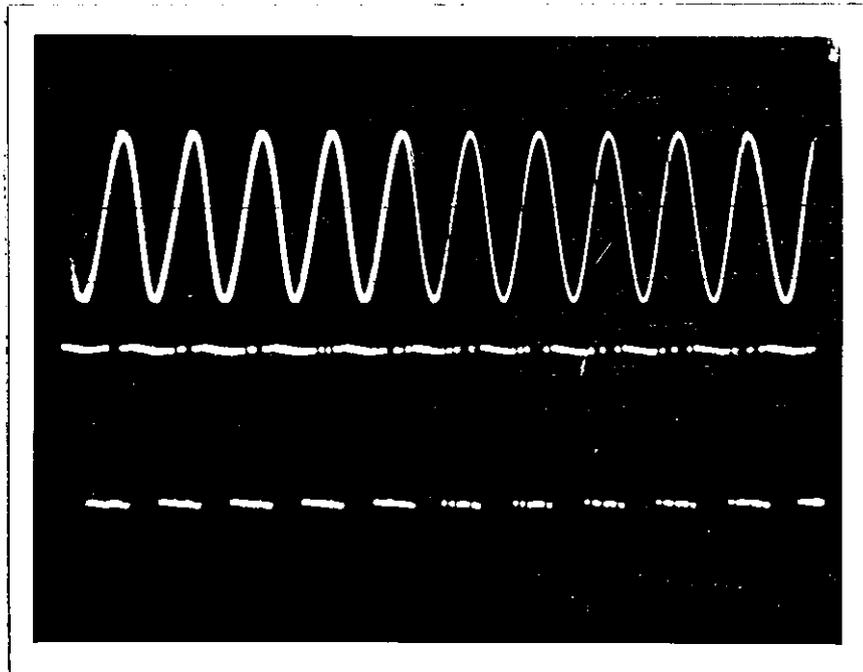
Comparison of the waveforms observed at the decoder lead to two main conclusions:

- (a) The lower frequencies of the input signal suffer more distortion than the mid-range ones ( $\approx 1000\text{Hz}$ ). This is due to the presence of long strings of like digits in the channel - a condition which has a similar effect to that of overloading.
- (b) The estimation of the input signal by the decoder improves with increasing the sampling frequency. This is in agreement with equations (2.24) and (2.26).

The conclusions (a) and (b) above are in agreement with theory suggested by Steele<sup>(2)</sup>. This stems from the fact that for a given sampling frequency  $f_p$ , the error signals  $\epsilon_1(t)$  varies more slowly when the input signal is of low frequency than when the input signal frequency is high (compare Fig. 4.1 with Fig. 3.8c). This means that the noise energy flowing in the output filter passband is maximum when the input signal is a tone of frequency in the range of  $300\text{Hz}$ .

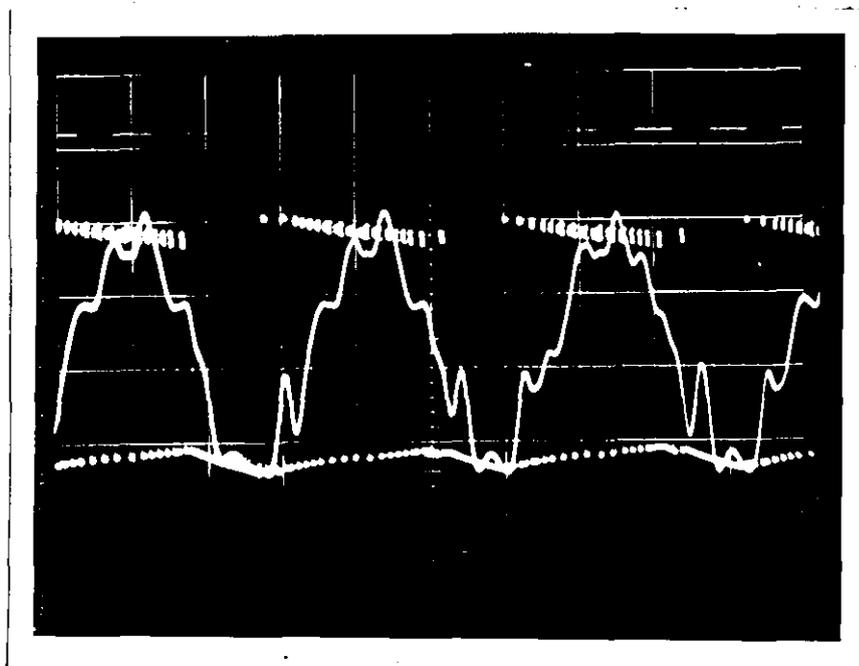


(c)

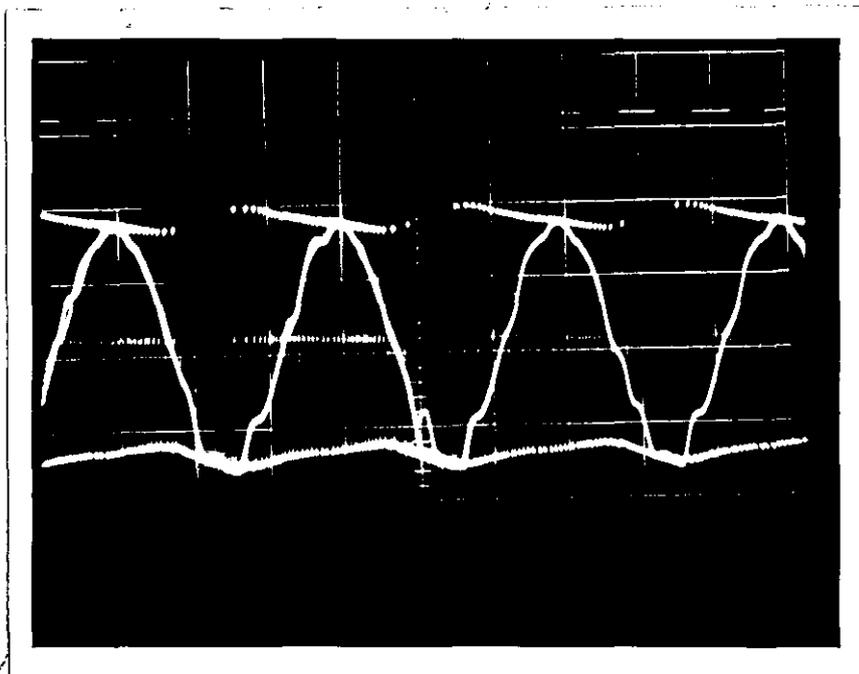


(d)

- Figure 4.8 Waveform observations at the decoder end
- (c) The feedback signal  $H(t)$  and its low frequency content  $\hat{x}_2(t)$  when the input to the encoder was  $2\sin(2\pi \cdot 1000t)$  and  $f_p = 38.4$  Kb/s.
  - (d) The same situation as in (c) above but with  $f_p$  raised to 76.8 Kb/s.



(a)



(b)

Figure 4.8 The waveform observation at the decoder end

- (a) The pulse height modulator output  $H(t)$  and its low pass contents  $\hat{x}_2(t)$  at the output of the decoder filter.  $f_s = 400 \text{ Hz}$ ,  $f_p = 19.2 \text{ Kb/s}$  and the amplitude  $A(j)$  of the input signal is 2 volts.
- (b) The same condition as in (a) above but with  $f_p$  raised to 56 Kb/s.

### 4.3 PERFORMANCE MEASUREMENTS

There are three main approaches to the assessment of the performance of communication systems designed for the transmission of speech<sup>(76)</sup>.

These are:

- (1) Measurement of the articulation index
- (2) Listening method using sentence material
- (3) Measurement of signal-to-quantization noise powers ratio.

#### (1) Articulation measurements

Articulation measurements provide an example of a method intended to determine the acceptability of a speech communication system. In this measurement the ensemble of symbols is defined by the set of phonemes used in the language concerned. These must be built into syllables that can be pronounced and this may be done according to various statistical and linguistic principles. For example, taking only single phonemes, initial and final consonant and the six short vowels, it is possible to generate nonsense syllables (termed "logatoms"), pronounceable by English speaking persons. Examples of these "logatoms" are VUM, FOT, NOTH, etc. (see reference (76) for more details). This type of logatom is unlike syllables found in any real language; but it has the advantage that analysis of the scores for individual sounds is greatly facilitated and confusion matrices<sup>(76)</sup> can readily be constructed from the resulting errors. Articulation tests therefore consist, in principle, of determining the percentage of information bearing elements, emitted by the talker, that are recognised correctly by the listener.

Articulation tests are usually conducted with a trained testing team. The talking conditions are carefully controlled by providing the talker with means of monitoring the level of his utterances, for example a speech voltmeter connected to a high quality microphone located at a controlled distance from his lips. Control of talking is facilitated by

emitting each logatom in a carrier phrase (e.g. the British Post Office uses the carrier phrase "*The next word will be*"). The listening conditions are also controlled including provision of suitable listening cabins and furnishing them with definite levels of room noise. The testing team usually consists of five trained persons of which at any time four listen simultaneously on parallel connected receiving ends, while the fifth does the talking.

The scores obtained from an articulation test have very little significance except when these scores are used to obtain ratings relative to a defined reference system such as that recommended by the CCITT (see reference (76)).

This method of performance testing for the SCALE system was ruled out because of the difficulties relating to the formation of a trained team of listeners and the associated costs of providing the correct testing environment.

## (2) Listening methods using sentence material

One of the simplest listening methods for the assessment of communication systems designed for speech is that known as "the immediate appreciation" method. This was devised by Grinstead in 1937. This method retains the idea of scoring in terms of the amount of material "satisfactorily" received but avoids excessive writing down of the received material by the listener as some other methods specify.

In Grinstead's method, a fixed one way speech path is used, with talking and listening conditions held constant, as for the articulation measurement method. The scoring is done by the listener noting whether or not he understood the meaning of each sentence without too much mental effort. This method can be modified by defining on a category scale the amount of effort needed to understand a sentence. Such a scale can be

designed to assess the threshold of detectability or objectionableness.

The sentence material used by the British Post Office is obtained by selecting, say, one sentence from each page of a number of novels or light literature. The sentences chosen are then placed in random sequence arranged as lists of ten groups of five sentences.

When making recordings, ten groups are recorded by one talker while the next ten are recorded by another so that a variety of voices is used.

When sentence material is used, the subject taking part in the listening tests should preferably be naive, except in the case of immediate appreciation tests on rather poor speech communication systems when an experienced crew might be a necessity.

When the immediate appreciation method is used a score of 95% is given for a speech communication system with approximately 33 dB reference equivalent.

In the SCALE experiment under consideration a subjective performance test similar to the immediate appreciation test was conducted. In this test the SCALE encoder was used to encode a recorded long paragraph from a lecture note on spectrum analysis. The SCALE decoder was used to decode and filter the binary pattern received from the encoder. The subject taking part in the listening were a group of approximately 100 electrical engineers who were gathered at Loughborough University lecture theatre<sup>†</sup>. The sampling frequency was decreased in steps from 100 Kb/s to 8 Kb/s

---

<sup>†</sup>The subjects who took part in the listening were gathered to attend an IEE (Institute of Electrical Engineers) lecture on digital encoding. The title of the lecture was "See it work" by Dr. R. Steele, given on the 4th May 1976.

while the speech paragraph was being fed to the SCALE encoder. The listeners indicated that the sentences they received were completely understood. They have also indicated that at low bit rates (8 Kb/s to 32 Kb/s) they found listening to the SCALE system less effort demanding than listening to companded PCM and DPCM<sup>†</sup> operating at the same bit rates.

Although this test was not carried out in the formal way, it nevertheless gave some indication of how the SCALE system performs subjectively.

### (3) Measurement of signal-to-noise ratio (SNR)

The main disadvantage of this method stems from the fact that the test signal used is a band limited Gaussian noise waveform. The statistical properties of speech differ from those of Gaussian test signals; in particular speech cannot be treated as though it were a statistically stationary process. However, the SNR method of testing is widely used (e.g. used by the British Post Office as a standard test). This is because the test signal and the test procedure are fixed (provided the same type of test equipment is always used) and the results obtained are independent of the human factors affecting other testing methods (e.g. articulation test results might vary when the testing group is changed, etc.).

To test the performance of the SCALE system experimentally through the measurement of SNR the system was connected as shown in Fig. 4.9. The quantization distortion meter QDM (Marconi TF2343A) is a standard British Post Office measuring equipment. It supplies a Gaussian signal (with uniform power spectrum) which is bandlimited by an internal bandpass filter, the lower and upper frequencies of which are 450 and 550 Hz respectively. The output power  $P_{in}$  of the QDM can be varied between -50 dBm and +2 dBm in 1 dBm steps. The output of the QDM is applied through

---

<sup>†</sup>The PCM and DPCM were supplied by Bell Laboratories and demonstrated by Dr. R. Steele of Loughborough University.

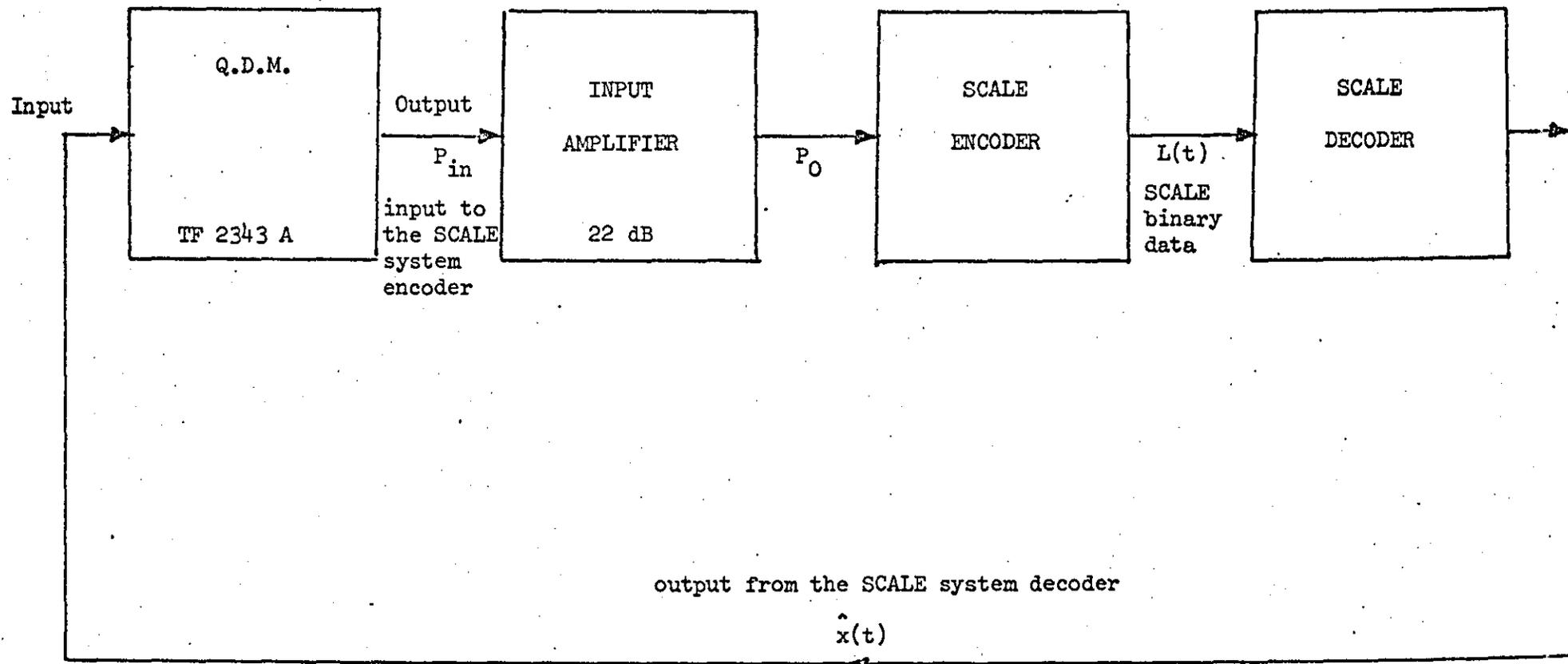


Fig. 4.9 Set-up for measurement of SNR using Marconi TF 2343 A

the input amplifier (which is included in the encoder input filter) to the SCALE encoder. The digital out of the encoder is fed to the SCALE decoder to reconstruct the baseband signal (which has been applied to the input of the QDM). Comparison between the QDM input and out powers enables the noise and hence the SNR to be determined. Using this arrangement of Fig. 4.9 several sets of measurements were performed to relate the SNR to the input signal power, sampling frequency  $f_p$ , syllabic time constant  $T_2$  and the effective SR length  $N_e$ .

#### 4.3.1 SNR as a function of the input power, $f_p$ , $T_2$ and $N_e$

(i) For a sampling frequency  $f_p = 19.2 \text{ KHz}$ ,  $T_2$  was set to 10 m.s. and  $N_e$  was set to 2 in both the encoder and the decoder units. The input power to the encoder input amplifier was varied between  $-50 \text{ dBm}$  and  $+1 \text{ dBm}$  and the corresponding SNR values, displayed on the QDM panel, were noted. The measurement was repeated for  $T_2 = 15$  and  $T_2 = 20 \text{ m.s.}$  The results of this experiment are plotted in Fig. 4.10a.

(ii) With  $f_p$  still at 19.2 Kb/s but with  $N_e$  switched to 3, the input power was again increased from  $-50 \text{ dBm}$  to  $+1 \text{ dBm}$  and the SNR noted. The procedure was then repeated for  $N_e = 4$  and  $N_e = 5$ . The results are graphed in Fig. 4.10d.

(iii) The same procedure as (ii) was repeated but with  $f_p = 38.4 \text{ Kb/s.}$  The results are given in Fig. 4.10c.

(iv) Measurement of  $R_c$  - the maximum available SNR - was achieved by first setting  $N_e = 2$  and  $T_2 = 10 \text{ m.s.}$  For  $f_p = 19.2 \text{ Kb/s,}$  the input power was increased from  $-50 \text{ dBm}$  until maximum SNR " $R_c$ " was indicated on the QDM meter. This was repeated for  $f_p = 28.0, 38.4$  and  $56 \text{ Kb/s.}$  The whole procedure was then repeated for  $N_e = 3, 4$  and  $5$ . Graphs of  $R_c$  as a function of  $N_e$  and  $f_p$  are plotted in Fig. 4.10b.

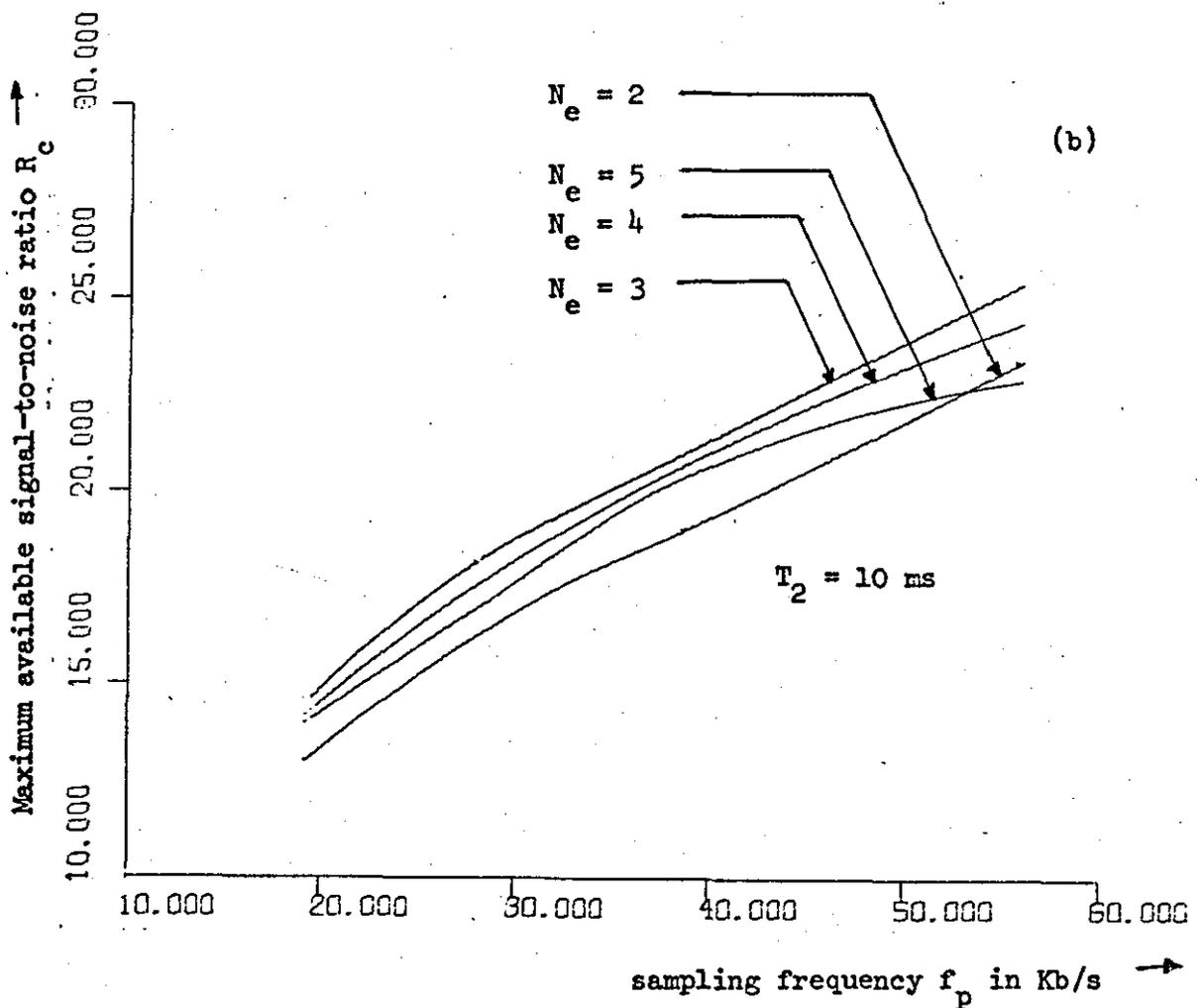
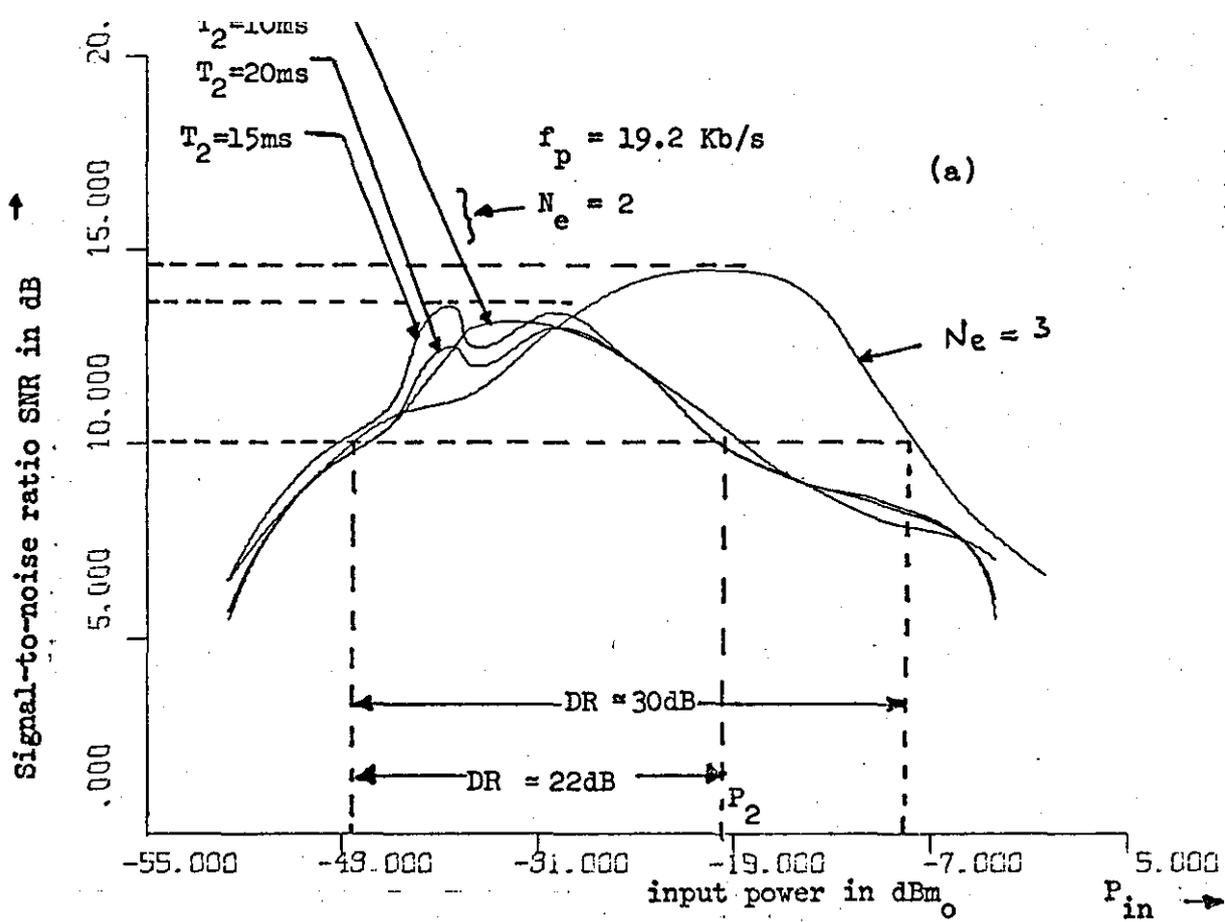


Figure 4.10 Signal-to-noise ratio SNR as a function of the input power  $P_{in}$ , sampling frequency  $f_p$ , syllabic time constant  $T_2$  and shift register length  $N_e$ .

(a)  $T_2$  and  $P_{in}$  are variables. (b)  $N_e$  and  $f_p$  variables.

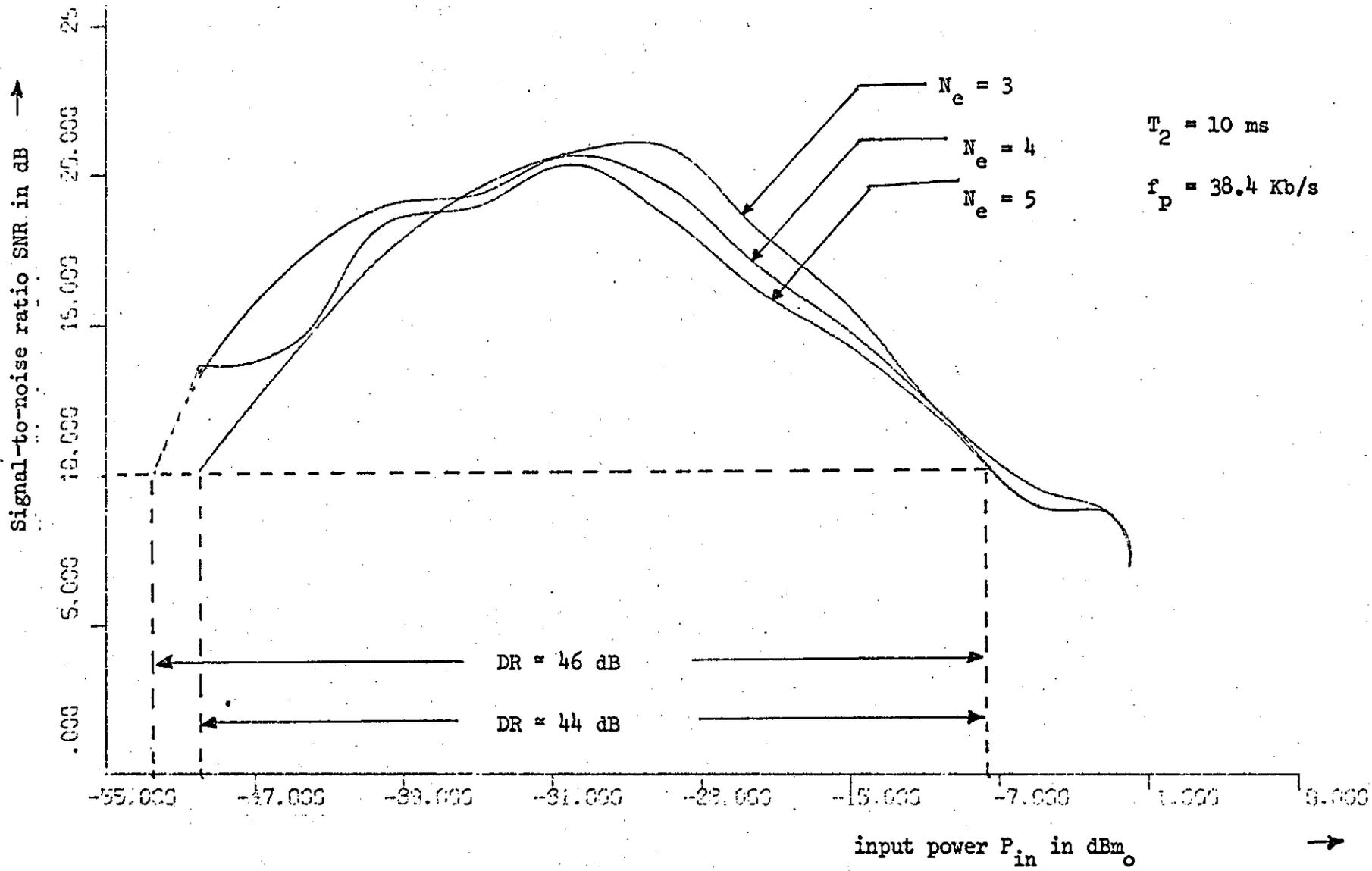


Figure 4.10(c) Variation of the SNR with the input power  $P_{in}$  when  $f_p$ ,  $T_2$  are constants and  $N_e$  is a parameter

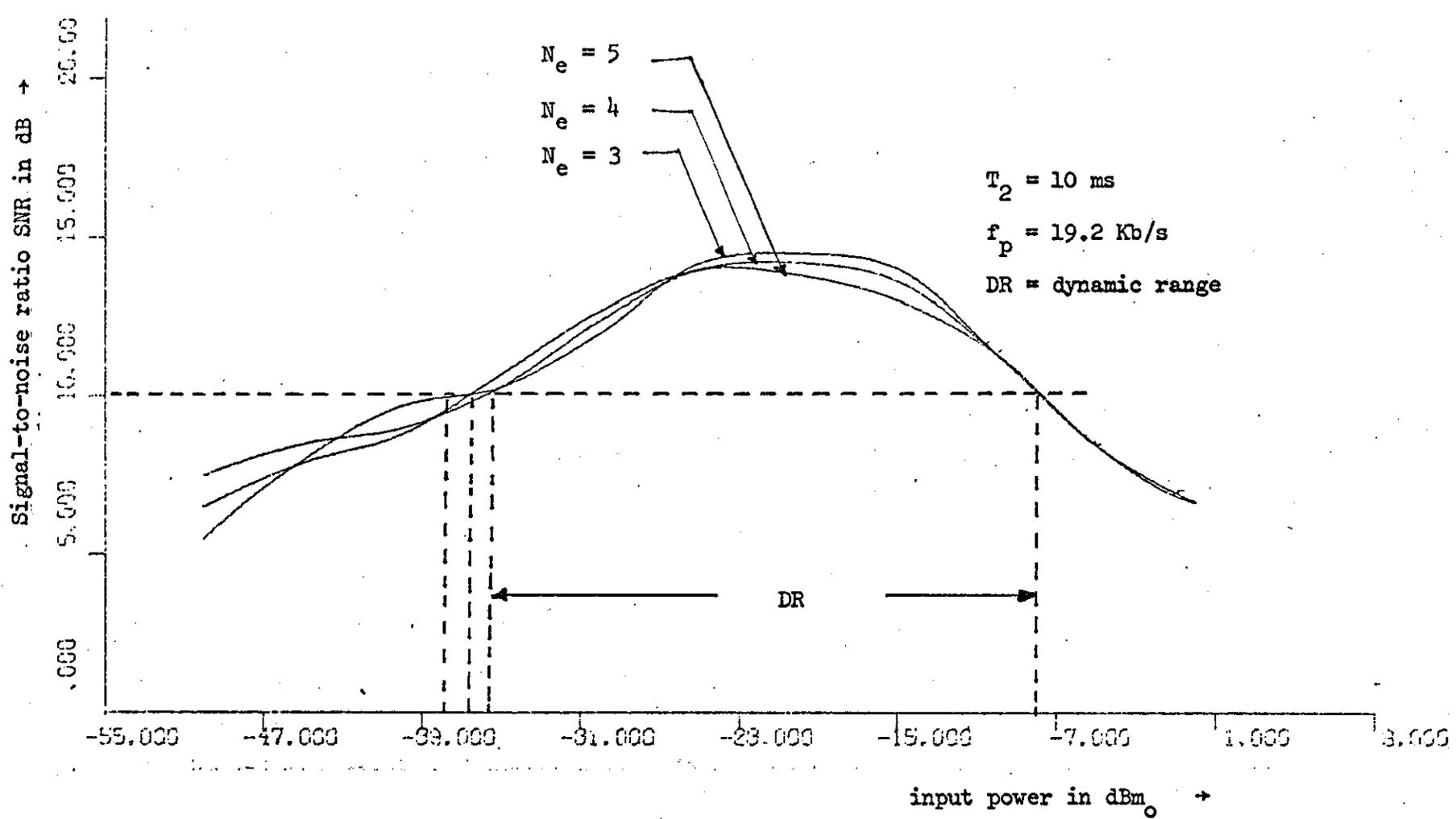


Figure 4.10(d) Variation of the SNR with input power  $P_{in}$  and the shift register length  $N_e$  when  $T_2$  and  $f_p$  are fixed.

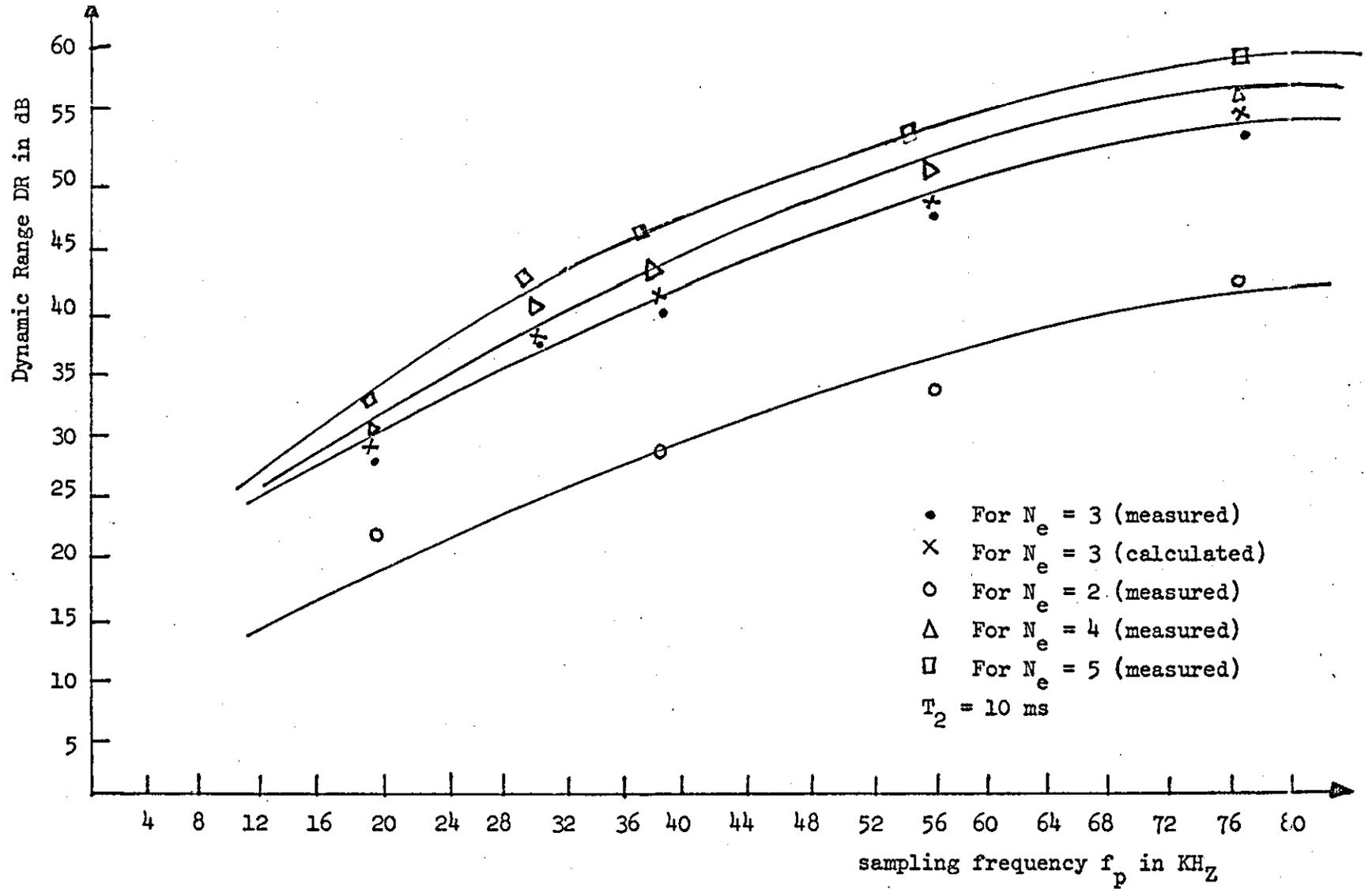


Figure 4.10(e) Variations of the dynamic range DR with the sampling frequency  $f_p$  and shift register length  $N_e$  when  $T_2$  is fixed to 10 ms.

(v) For minimum acceptable SNR " $\rho$ " of 10 dB the dynamic range DR, (iv) was repeated but the input power this time was increased from -50 dBm to  $P_1$  which corresponds to an SNR of  $\rho$ . The process of incrementing the input power was continued to  $P_2$  when the SNR =  $\rho$  again. The DR is by definition "the input range in dB within which the SNR is not less than the acceptable minimum" and can be obtained by subtracting  $P_2$  from  $P_1$  (see Fig. 4.10a: DR =  $P_1 - P_2 = 22$  dB). The results of these tests are plotted in Fig. 4.10e.

#### 4.3.2 Analysis of the results

The experimental results obtained above are now examined carefully and conclusions based on them are drawn.

(i) The results given in Table 4.1 and Fig. 4.10a indicate that the variation of  $T_2$  in the range 10-20 m.s. has negligible effect on the maximum available SNR and dynamic range.

(ii) Comparison of the results shown in Figs. 4.10a, 4.10c and 4.10d (see also Fig. 4.10e) shows that at  $f_p = 19.2$  Kb/s a considerable improvement in the DR (in addition to about 1 dB improvement in the maximum available SNR) when the effective SR length  $N_e$  is increased from 2 to 3. Further extension of the SR does result in a relatively small improvement in the DR but this improvement is only available at the expense of reduced value of the maximum available SNR (see Fig. 4.10b).

(iii) Closer examination of Figs. 4.10b and 4.10e reveals that the rate of increase of both the maximum available SNR and the DR with  $f_p$  decreases with rising values of  $f_p$ . This leads to the conclusion that the multipliers  $K_c$  and  $\delta$  used in equations (2.25) and (2.29) for calculating the SNR and the DC cannot be constants but variables which are functions of the sampling frequency and the effective SR length.

(iv) Calculation of  $\delta$  and  $K_c$

Recalling equation (2.25)

$$K_c = \frac{3f_p}{8R_c \pi^2 (3f_l^2 B + f_a^3 - f_b^3)}$$

The constants  $f_l$ ,  $B$ ,  $f_a$  and  $f_b$  are given in Table 4.6. So the measurement of  $R_c$  for various values of  $f_p$  enables  $K_c$  variation with the sampling frequency to be calculated.

$\delta$  can be calculated as a function of frequency by measuring  $R_c$  and the corresponding input power  $P_{in}$  (see Fig. 4.9). Now remembering that  $P_{in}$  is in dBm across  $600\Omega$  and that the input amplifier has a gain of  $\approx 22$  dB, the input power to the SCALE encoder  $P_o$  (see Fig. 4.9) is given by

$$\begin{aligned} P_o &= P_{in} + 22 + 2.3 \text{ dB and} \\ &= 20 \log \sigma_x \end{aligned}$$

Recalling equation (2.24)

$$SNR = R_c \left\{ \frac{\sigma_x}{GZ_h (1 - e^{-\delta \sigma_x}) + H_{min}} \right\}^2$$

The value of  $\sigma_x$  which produces maximum SNR is that which reduces the above equation to

$$\frac{SNR}{R_c} = \left[ \frac{\sigma}{GZ_h (1 - e^{-\delta \sigma_x}) + H_{min}} \right]^2 = 1$$

Measurement of  $R_c$  and the corresponding  $\sigma_x$  for different values of  $f_p$  enables  $\delta$  to be calculated. Therefore

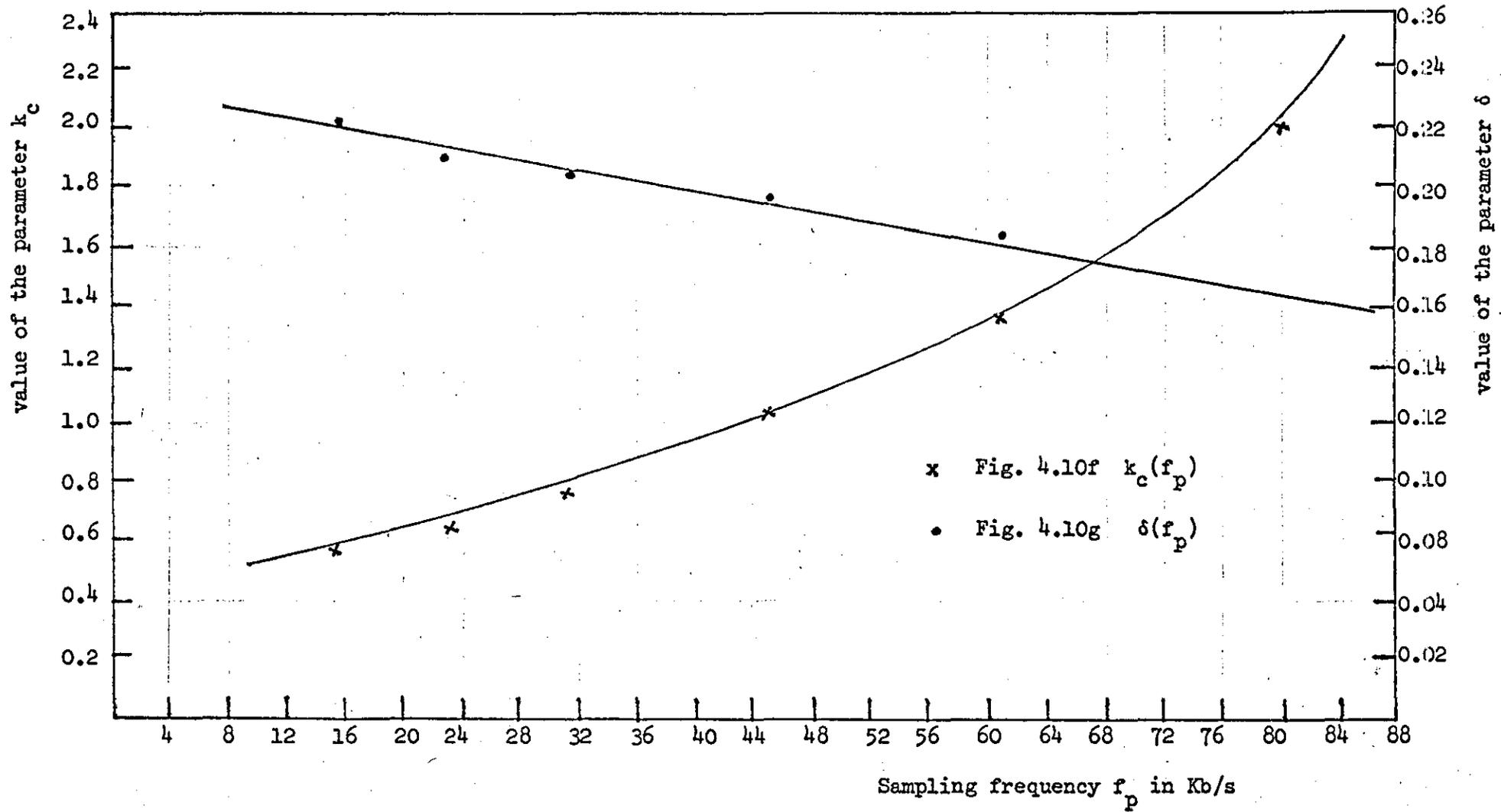
$$\delta = \frac{1}{\sigma_x} \ln \left( \frac{GZ_h}{GZ_h + H_{min} - \sigma_x} \right)$$

The variations of  $K_c$  and  $\delta$  with  $f_p$ , for  $N_e = 3$ , are defined in Table 4.1 and plotted in Figs. 4.10f and 4.10g.

Maximum measured SNR in dB	Measured* input power to the input amplifier $P_{in}$ in dBm	Measured* input power to the SCALE encoder $P_o$ in dB	Input voltage corresponding to $P_o$ $\sigma_x$ in volts	$\delta = \frac{1}{\sigma_x} \cdot \ln  GX_h / (GZ_h + H_{min} - \sigma_x) $ †	$k_c = 3f_p^3 /  8\pi^2 R_c (3f_1^2 B + f_a^3 - f_b^3) $ †	Dynamic range DR in dB	Sampling frequency $f_p$ in Kb/s
14.5	-20.0	0.0	1.000	0.2024	0.55	29.6	19.2
18.2	-25.0	-5.0	0.560	0.1892	0.64	37.8	28.0
21.0	-27.0	-7.0	0.446	0.1848	0.76	41.9	38.4
52.5	-30.0	-10.0	0.316	0.1780	1.05	47.4	56.0
28.5	-36.0	-16.0	0.158	0.1603	1.36	57.0	76.8

Table 4.1 Calculation of  $\delta$ ,  $k_c$  and DR from the measured values of maximum SNR( $R_c$ ),  $f_p$  and  $\sigma_x$ .

\* See Fig. 4.9. † See equations (2.23), (2.24) and (2.25).



Figures 4.10(f) and (g) Variations of the parameters  $k_c$  and  $\delta$  with the sampling frequency  $f_p$  when  $T_2 = 10$  ms,  $N_e = 3$

Comparison of the values of  $\delta$  obtained by experiment and computer simulation (see Chapter III) show close agreement between the hardware and simulated model results ( $\delta_{\text{computer}} \approx 0.21$ ,  $\delta_{\text{hardware}} \approx 0.20$  for  $f_p = 19.2$  Kb/s,  $N_e = 3$ ). The close agreement between the theory of SCALE discussed in Chapter II and the experimental and computer simulation results is further demonstrated by plotting the SNR curves obtained by the three methods on the same axes as shown in Fig. 4.11.

#### 4.4 Computer program for analysing and displaying the results

Numerical calculations and graph plottings normally associated with practical experimental results can be dealt with more accurately, neatly and efficiently by computer than by pencil and paper. For these reasons a computer program was written to accept the experimentally measured data (SNR, input power, etc.), perform the necessary calculations and - using the X-Y plotter - display the results in graphical form.

The program consists of a Master Segment "SCALE DESIGN" and five subroutine segments: "COMPARE", "TWOBIT", "FITTER", "RESULT" and "SNRACL".

##### (a) Subroutines TWOBIT and RESULT

These are reading and writing subroutines. "TWOBIT" reads into the computer memory all the results obtained from measurements performed on the SCALE hardware model when  $N_e = 2$ . Subroutine "RESULT" performs the same function for  $N_e = 3, 4$  and  $5$ .

##### (b) Subroutine COMPARE

The function of this subroutine is to accept the various parameters involved in equation (2.25) ( $f_p, f_1, f_a, f_b$ , etc.) and calculate  $R_c$  - the maximum expected SNR. The value of  $R_c$  and  $10 \log R_c$  are returned to the master segment for comparison with the values of  $R_c$  obtained by measurement and computer simulation.

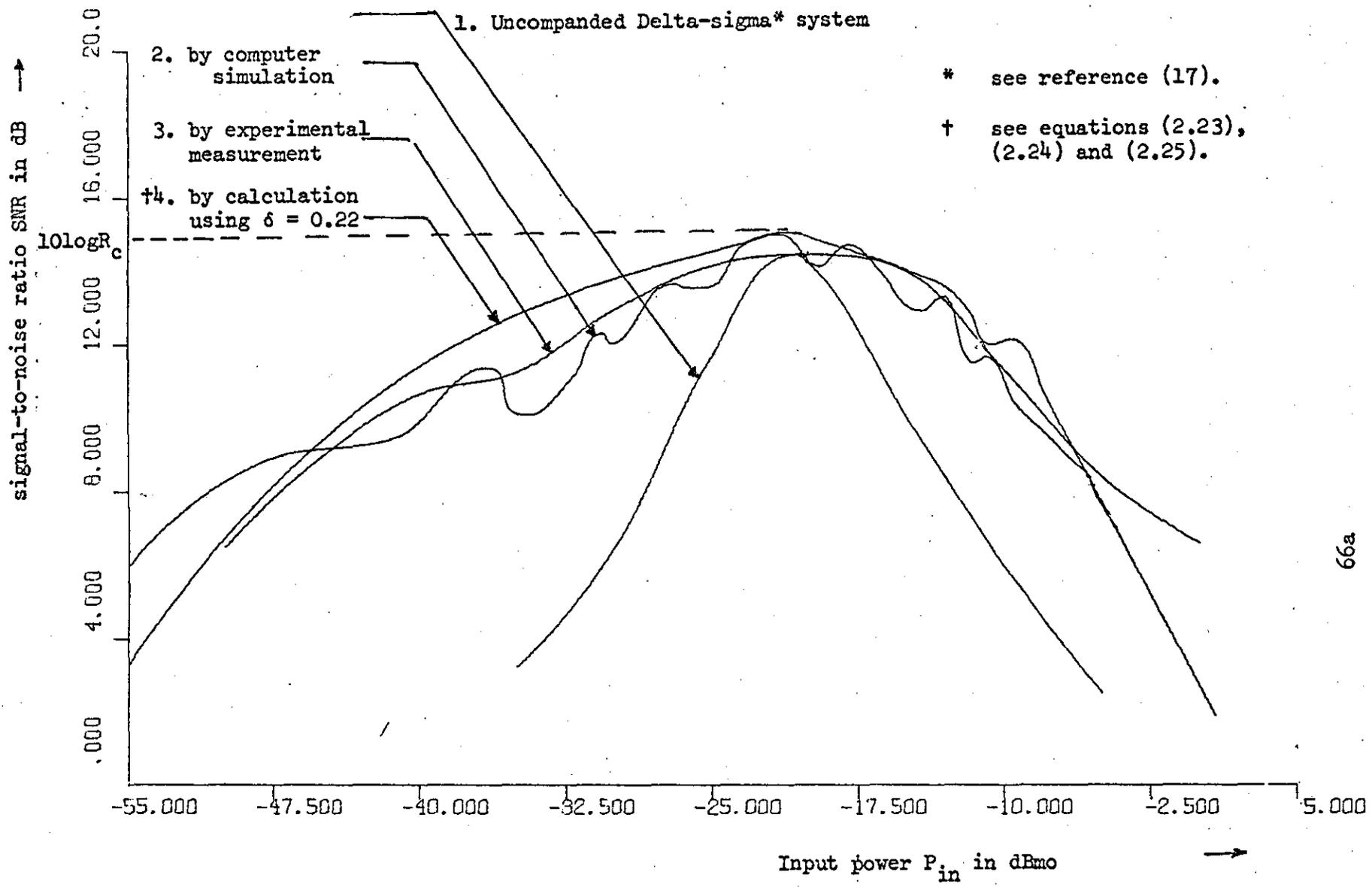


Figure 4.11 SNR curves obtained by measurement (3), computer simulation (2), calculation (4) and for comparison, the curve presented by Wilkinson<sup>(17)</sup> for uncompanded Delta-sigma system.

(c) Subroutine FITTER

This subroutine reads in  $\sigma_x$ ,  $\delta$ ,  $G$ ,  $Z_h$  and  $H_{min}$  and performs the calculations specified by equations (2.12) and (2.15) (calculates  $V_{cp}$  and  $H_p$ ). When  $\delta$  is unknown this subroutine can be used in conjunction with "SNRACL" (as discussed in Chapter III) to find the value of  $\delta$  which causes the calculated SNR curve to coincide with that obtained experimentally or by computer simulation.

(d) Subroutine SNRACL

The inputs to this subroutine are:

- (1) The measured SNR and input power provided by "RESULT".
- (2) The values of  $V_{cp}$  and  $H_p$  measured on the simulated model as functions of the input r.m.s. voltage.
- (3) The values of  $V_{cp}$  and  $H_p$  calculated by "FITTER" as functions of the r.m.s. voltage ( $\delta$  taken from Fig. 4.10g).
- (4) SNR values and the corresponding input power for a Delta-sigma system. (These values were extracted from a SNR curve given in reference (17))

"SNRACL" accepts the above inputs and performs the following operations:

- (i) It calculates the SNR as a function of the input power for the simulated model. This is done by substituting the appropriate values of the simulated feedback and the corresponding simulated input signal voltages  $H_p$  and  $\sigma_x$ , respectively, into equations (2.23) and (2.25).
- (ii) Repeats (i) above but this time the value of  $H_p$  to be substituted in equations (2.23) and (2.25) is obtained by calculation. The calculations of  $H_p$  in this case is carried out by "SUBROUTINE FITTER" which performs the operation specified by equation (2.15).

In (i) and (ii) above the value of  $R_c$  in equations (2.23) and (2.25) is calculated from equation (2.24) using "SUBROUTINE COMPARE" which can be

called by "SNRACL".

- (iii) It converts  $\sigma_x$  into dBm across  $600\Omega$  at the input port of the input amplifier (it attenuates the values of  $\sigma_x$  by 22 dB to compensate for the gain of the amplifier and amplifies the resultant by 2.3 dB converting the result into dBm across  $600\Omega$ ).
- (iv) It plots on the same axis the SNR curves as functions of the input power in dBm for the hardware, software and the theoretical models of the SCALE system and the uncompanded Delta-sigma system.

The program flow chart (Fig. E1), program listing and explanation of terms are given in Appendix E.

## CHAPTER V

### DIGITIZATION OF THE SCALE SYSTEM

#### 5.1 INTRODUCTION

The design and performance of the SCALE system have been investigated in detail in the last two chapters. Both the objective tests (carried out through measurements of SNR and DR) and subjective tests (carried out through actual listening to speech and tones signals processed by the SCALE CODEC hardware model) showed that the system, when carefully implemented, can provide a commercially viable source-encoding system for speech signals. This viability is based on its flexibility (SNR can be traded for lower bit rate and vice versa), tolerance to channel errors and acceptable subjective performance (the degree of annoyance caused by overload and quantization noise is less than that associated with other forms of digital encoding systems). Observations made on the performance of the hardware model do, however, confirm the inherent disadvantages in the system. These are:

- (i) Dependence of the system performance on the tolerance and reliability of the analogue components which constitute the bulk of the building blocks. It was observed, for example, that the pulse height modulator characteristics can drift with time and rail voltage variations causing severe degradation in the system performance. The discrete analogue components such as transistors, resistors and capacitors were replaced more frequently during the experiment than the integrated digital ones reflecting the superior reliability of digital integrated circuits.
- (ii) The dependence of the reconstructed feedback signal on the long term history of the channel pattern through the slowly varying control voltage  $V_c$ . This is a serious disadvantage because it

makes the SCALE binary data unsuitable for digital signal processings such as digital filtering and code conversion.

This chapter is concerned with the modification of the SCALE system shown in Fig. 2.3 so that the difficulties listed above can be overcome. This objective can be achieved by replacing the hybrid local and remote decoders by digital ones. The modification procedure will now be developed. It will also be shown by computer simulation that the degradation in the system performance due to digitization is negligible. Finally, hardware implementation of the modified system - using commercially available integrated circuits - is discussed.

## 5.2 DESCRIPTION OF THE MODIFIED SYSTEM

The new system is shown in Fig. 5.1. The de-emphasis circuit and the comparator of Fig. 2.3 are retained because they are common to the input signal which is in analogue form. The shift register SR and the control logic (DL and SL) are also retained because they are already digital. The syllabic integrator, pulse-height modulator and polarity circuits of Fig. 2.3 have now been eliminated. They are replaced by an adaptive digital accumulator, D/A converter and a digital polarity switch P.

The operation of the system up to and including the control logic is exactly the same as described in Chapter II. At the instance of the  $i^{\text{th}}$  clock pulse the digital accumulator produces  $H_{Di}$  which constitutes a 12 parallel digit word and represents the magnitude of the feedback step height in digital form. The D/A converter accepts  $H_{Di}$  and produces an analogue signal  $\hat{H}_i$  which is a very close estimate of  $H_i$  defined by equation (2.16). The closeness of  $\hat{H}_i$  to  $H_i$  depends on the adaptation algorithm of the digital accumulator which will be discussed in section 5.4. The polarity of the feedback signal is decided as before by the logical level of the present digit of  $L(t)$ .

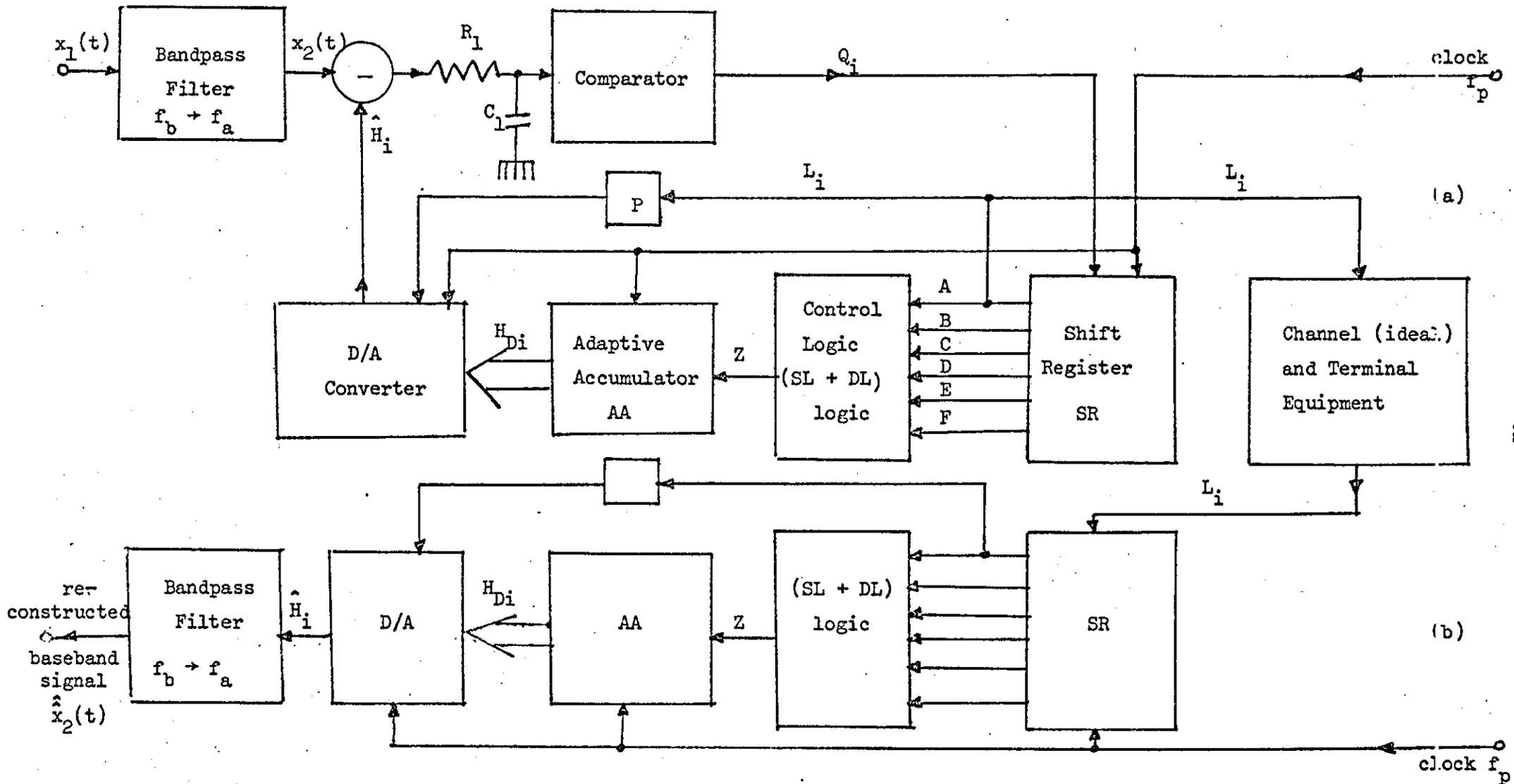


Figure 5.1 Block diagram of the Digital SCALE. (a) Encoder (b) Decoder

At the decoder the reconstructed baseband signal  $\hat{x}_2(t)$  is obtained by passing  $\hat{H}_i$  through a bandpass filter.

The significance of the arrangement of Fig. 5.1 is that - in addition to the elimination of the undesirable analogue components associated with Fig. 2.3 - the output  $H_{Di}$  of the decoder is now in digital form. This means that code conversion between the SCALE binary data and PCM formats - which is one of the main objectives of this study - is now possible. Code conversion between the SCALE system and A-law PCM is discussed in detail in Chapters VI and VII.

(102)

### 5.3 THEORY OF THE ADAPTIVE ACCUMULATOR

The magnitude of the instantaneous value of the feedback step height  $H_i$  in the case of the hybrid SCALE system (Fig. 2.3) depends on two factors:

- (i) The most recent  $N_e$  logical values of  $L(t)$ , which determine the logical level of the selection logic output pulses  $Z$ .
- (ii) All previous logical values of  $L(t)$  (history of the channel pattern), which determine the value of the control voltage  $V_{c_{i-1}}$  (see equation (2.16)).

The sign of  $H_i$  depends only on the present logical value of  $L(t)$ .

The difference between the present and previous values of  $H_i$  (in the case of the hybrid SCALE) can be written as:

$$\begin{aligned} \Delta_i &= H_i - H_{i-1} \\ &= \Delta(Z, H) \end{aligned}$$

$$\text{i.e.} \quad H_i = H_{i-1} + \Delta(Z, H) \quad (5.1)$$

where  $Z$  is as defined by equations (2.8) and (2.9), and  $H$  is defined in Fig. 2.2 -  $H_{\min} \leq H \leq H_{\max}$ . Because  $H$  is an analogue signal and thus has an infinite number of values - between the limits specified above - it is not possible to realize equation (5.1) by digital hardware. It is, however, possible to divide  $H$  into  $M_0$  regions - or segments - within each of which  $\Delta_i$  magnitude is  $\approx$  constant. Using this piece-wise linear

approximation technique a step height  $\hat{H}_i$  (at the  $i^{\text{th}}$  clock instance) which is a good approximation to  $H_i$  can be digitally constructed from  $L(t)$ . In this case equation (5.1) becomes:

$$\hat{H}_i = \hat{H}_{i-1} + \Delta_i(Z, J) \quad (5.2)$$

where  $J = 0, 1, \dots, M_0$  and each of its values defines a particular region of  $H$ .

The relationship between  $H$  and  $J$  must satisfy the following conditions:

- (a)  $M_0$  - the maximum number of segments in one polarity of  $H$  - must be as small as possible to minimize the amount of the required hardware; but large enough to minimize the difference between  $H_i$  and  $\hat{H}_i$ .
- (b) To simplify the implementation, the ratio of the maximum feedback step height to the minimum within each segment must be a number divisible by 2 without a remainder.

The binary equivalents of  $\hat{H}_i$  and  $\Delta$  are  $H_{Di}$  and  $\Delta_{di}$  respectively (see Fig. 5.1). So equation (5.2) can be rewritten as:

$$H_{Di} = H_{D(i-1)} + \Delta_d(Z, J) \quad (5.3a)$$

It is known that in the case of the hybrid SCALE system (see Fig. 2.3) the feedback step-height magnitude increases when  $Z$  is a logical one ( $Z = Z_h$ ) and decreases when  $Z$  is a logical zero ( $Z = Z_l$ ). This means that at the  $i^{\text{th}}$  clock instance we must increment  $H_{D(i-1)}$  if  $Z$  is a logical one and decrement it when it is a logical zero.

One way of realizing the circuit described by equation (5.3a) is in the form shown in Fig. 5.2.

In Fig. 5.2 the main function of the  $\Delta$ -selection logic ASL is to constantly monitor  $Z$  and  $H_{D(i-1)}$  in order to evaluate  $J$ , and to produce the appropriate value of  $\Delta_d(Z, J)$ . When this is added to  $H_{D(i-1)}$ , the sum  $S_p$  - where  $S_p$  is a 12-bit word - is produced.  $\Delta_d(Z, J)$  is transferred

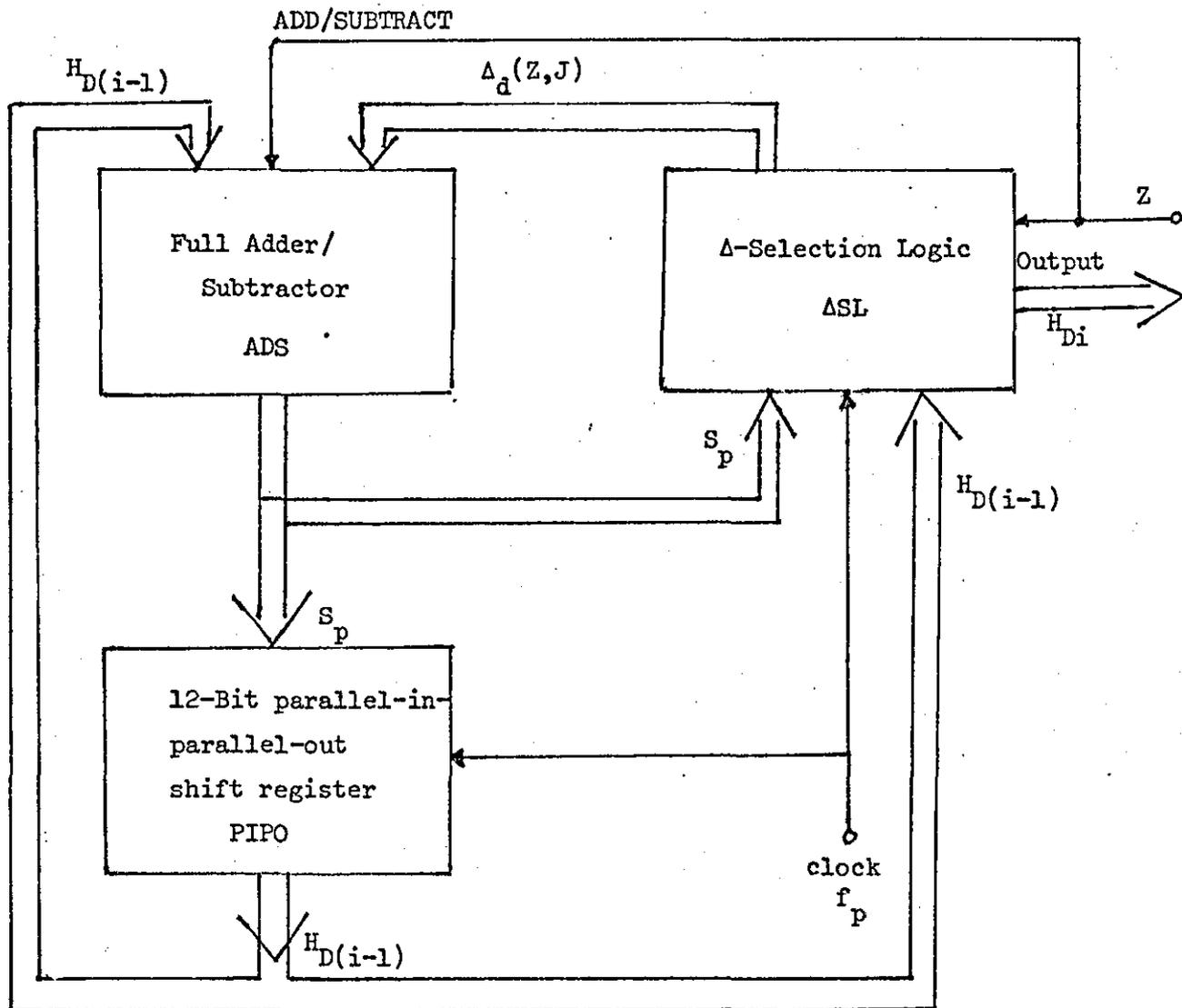


Figure 5.2 Block diagram of the adaptive accumulator

Note:  $H_{Di} = S_p$  except when  $H_{D(i-1)} = H_{Dmin}$  or  $H_{D(i-1)} = H_{Dmax}$ . In such cases  $H_{Di}$  is equated to  $H_{Dmin}$  or  $H_{Dmax}$  as appropriate and  $S_p$  is not passed to the output port.

to the input of the Adder/Subtractor (ADS) where it is added to  $H_{D(i-1)}$  or subtracted from it depending on whether  $Z$  is a logical one or zero respectively. When the  $i^{\text{th}}$  clock pulse comes along the output of the ADS, which is  $S_{p_i}$ , is inspected by the  $\Delta$ -selection logic  $\Delta$ SL and is also loaded into the parallel-in parallel-out shift register PIPO to be used by the  $\Delta$ SL for the evaluation of  $J$  for the next clock period operation.

Another function which must be performed by the  $\Delta$ SL is to detect when  $H_{D(i-1)}$  is a minimum (i.e. idling condition and  $H_{D(i-1)}$  is therefore equal to  $H_{\min}$ ) and when it is a maximum (overload condition and  $H_{D(i-1)}$  is therefore equal to  $H_{\max}$ ). On detection of one of these conditions  $\Delta$ SL sets  $\Delta_d(Z, J)$  to zero and  $H_{Di}$  to  $H_{\min}$  or  $H_{\max}$  as appropriate.

If neither of these two conditions is true the SL sets the output  $H_{Di}$  to  $S_p$ . The equation which completely describes the Adaptive Accumulator, abbreviated AA, can therefore be obtained by modifying equation (5.3a) to

$$\left. \begin{aligned}
 H_{Di} &= H_{D(i-1)} + \Delta_d(Z, J) & H_{\min} < H_{D(i-1)} < H_{\max} \\
 &= H_{\min} & H_{D(i-1)} \leq H_{\min} \\
 &= H_{\max} & H_{D(i-1)} \geq H_{\max}
 \end{aligned} \right\} (5.3b)$$

An alternative approach to the implementation of this equation is to replace the ADS of Fig. 5.2 by an Adaptive Up-Down Counter, AUDC, which increments its content by  $\Delta(Z, J)$  when  $Z$  is a logical one and decrements it by  $\Delta(Z, J)$  when  $Z$  is a logical zero. The  $\Delta$ SL in this case must be replaced by a control logic circuit which inspects the previous content of the AUDC (stored in PIPO) and presents to the input of the AUDC the corresponding values of  $Z$  and  $J$ . When the value of the PIPO output exceeds  $H_{\max}$  or becomes less than  $H_{\min}$  the control logic circuit

inhibits the counting and generates the appropriate value of  $H_{Di}$ .

Consideration in detail of these two approaches has shown that the resultant circuits are of equal complexity and would cost approximately the same price in the case of hardware implementation.

In the following sections of this chapter a general procedure for designing a digital SCALE decoder - from the knowledge of the parameters governing the operation of any hybrid SCALE system - is described. This is followed by a design example in which equation (5.3b) is used to synthesise a digital SCALE decoder for the decoding of the binary data originating from a hybrid SCALE. The first aim of the synthesis is to produce a digital SCALE using the counting approach described above which is then simulated on a digital computer. The second aim of the synthesis is to produce the digital SCALE in the form of Fig. 5.2 and employing commercially available TTL integrated circuits for hardware implementation.

#### 5.4 GENERAL DESIGN PROCEDURE

In order to design a digital SCALE system the following parameters must be defined:

- (a) The initial value of the feedback signal  $H_{Dmin}$  ( $H_{min}$  in binary).
- (b) The number of regions (segments) of  $H$  within each of which  $\Delta(Z)$  can be assumed to be constant.
- (c) The values of  $\Delta(Z_j)$  and  $\Delta(Z_h)$  for all values of  $J$ .
- (d) The maximum value of the feedback signal  $H_{Dmax}$  ( $H_{max}$  in binary).

Because the digital SCALE and the hybrid SCALE systems must be compatible (either one must be able to decode the transmission from the other), the design parameters ( $f_p, G, Z, T_2, H_{max}, H_{min}$ ) of the hybrid system must be available.

The design procedure can be carried out as follows:

- (i)  $H_{Dmin}$  and  $H_{Dmax}$  are equated to the binary equivalents of  $H_{min}$  and  $H_{max}$  respectively.
- (ii) Using the computer subprogram (SUBROUTINE CODEC) developed in Chapter II, simulate the hybrid SCALE decoder on a digital computer.
- (iii) Generate a group of consecutive like digits of length  $\ell = N_e$ , feed it to the simulated decoder and measure  $H_\ell$  ( $H_\ell$  is the value of  $H(t)$  when  $t = \ell T_p$ ).
- (iv) Increment  $\ell$  by one and repeat (iii).
- (v) Repeat (iv) until absolute overloading ( $H_\ell = H_{Dmax}$ ) and denote the value of  $\ell$  corresponding to  $H_{Dmax}$  by  $\ell_m$ .
- (vi) Repeat (iii) but with groups of  $\ell$  alternate digits (010101....)
- (vii) Repeat (iv) until  $\ell = \ell_m$ .

The variation of the feedback signal  $H_\ell$  as a function of the group length  $\ell$  - for the cases when the  $L(t)$  digits are like and when they are alternate - is of the form shown in Fig. 5.3. It can be seen from this figure that

- (a) The rate of growth of  $H_\ell$  ( $H_\ell - H_{\ell-1}$  in curve 1) *decreases with increasing values of  $\ell$  and  $H_\ell$ .*
- (b) The rate of decay of  $H_\ell$  ( $H_\ell - H_{\ell-1}$  in curve 2) *decreases with increasing values of  $\ell$  and decreasing values of  $H_\ell$ .*
- (c)  $H_\ell$  is incremented when the group consists of like digits and  $\ell \geq N_e$ , i.e. when  $Z$  is a logical one. For all other patterns  $Z$  is a logical zero and  $H_\ell$  is decremented.
- (viii) With the aid of the observed values of  $H_\ell$  above, divide the range of  $H_\ell$  between  $H_{min}$  and  $H_{max}$  into  $(M_0 + 1)$  segments. The segmentation process must satisfy the following conditions:
  - (a) The magnitude of  $H_\ell$  at the segment edges must be an even multiple of  $\Delta_{min}$  - the minimum change in  $H_\ell$  during one clock period.
  - (b) Within a segment,  $H_\ell$  must vary approximately linearly with  $\ell$ .

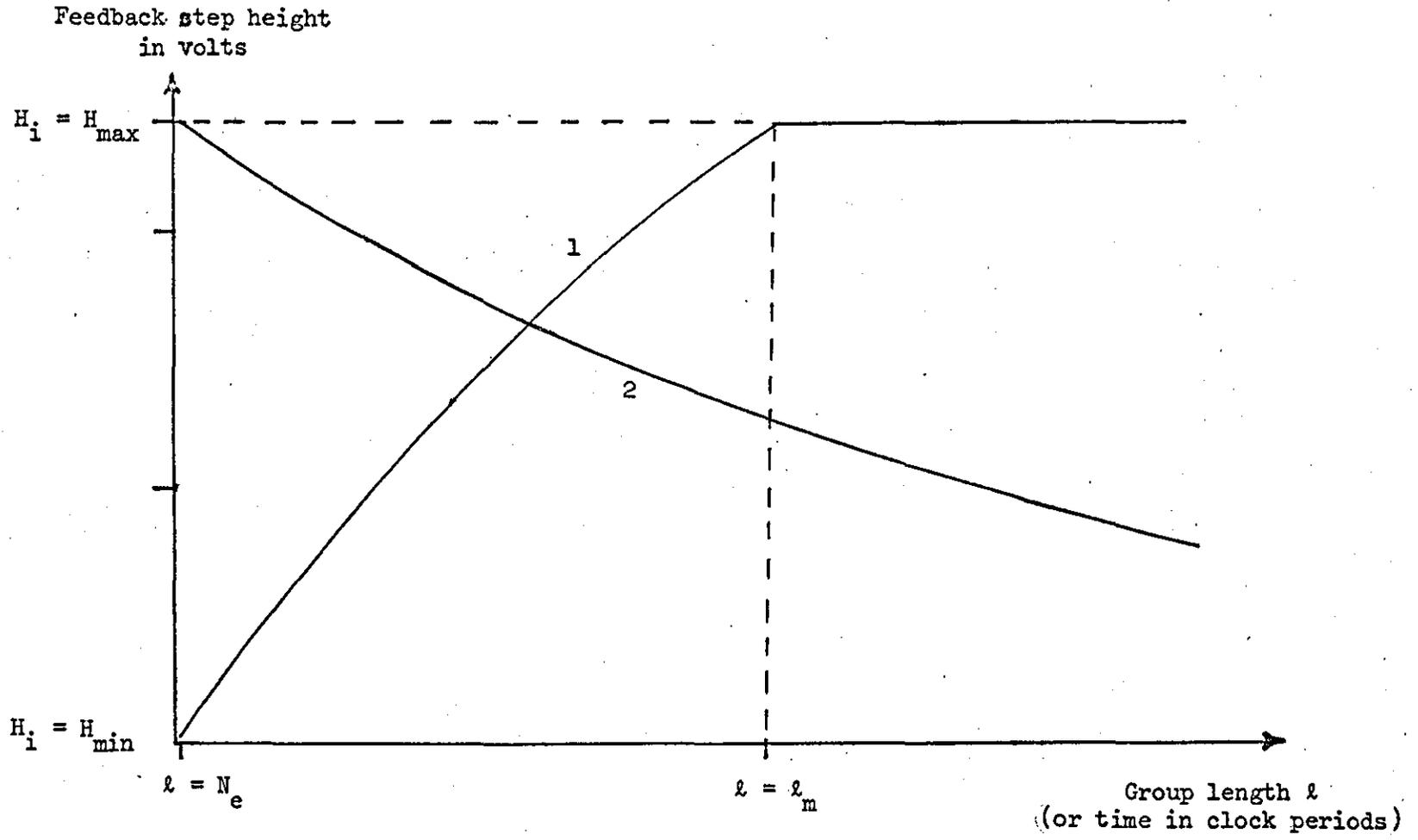


Figure 5.3 (1) The growth of  $H$  with  $l$  (the group length of the  $L(t)$  consecutive like-digits).  
 (2) The decay of  $H$  with the length of the group  $l$  (of the  $L(t)$  consecutive different digits).

Control logic Output Z	Segment No. J	Magnitude of Increment $\Delta_d(Z,J)$
$Z_l$	0	$\Delta_d(Z_l,0)$
	1	$\Delta_d(Z_l,1)$
	2	$\Delta_d(Z_l,2)$
$Z_h$	0	$\Delta_d(Z_h,0)$
	1	$\Delta_d(Z_h,1)$
	2	$\Delta_d(Z_h,2)$

(a)

Control logic Output Z	Segment No. J	Magnitude of Increment $\Delta_d(Z,J)$
0	00	$\Delta_d(0,00)$
	01	$\Delta_d(0,01)$
	10	$\Delta_d(0,10)$
1	00	$\Delta_d(1,00)$
	01	$\Delta_d(1,01)$
	10	$\Delta_d(1,10)$

(b)

Table 5.1 The increment magnitude per clock period  $\Delta_d$  as a function of the segment no. J and the control logic output Z.

(a) Z and J in decimal notation

(b) Z and J in binary notation

The principle can be seen from Fig. 5.4 for  $M_0 = 2$ .

- (ix) Connect the adjacent segment edges by straight lines and denote the segment numbers by  $J$  (where  $J \in \{0, 1, \dots, M_0\}$ ). The required constant increment (or decrement) of  $H_D$  per clock period can now be calculated using the linear relation:

$$\Delta(Z, J) = \frac{H_M(Z, J) - H_0(Z, J)}{\tau(J)} \quad (5.4)$$

where  $H_M$  and  $H_0$  are respectively the maximum and minimum values of  $H$  lying in segment  $J$ ,  $Z$  is a logical one or logical zero throughout segment  $J$  and  $\tau(J)$  is the number of clock periods needed for the change from  $H_0(Z, J)$  to  $H_M(Z, J)$  (or vice versa) in segment  $J$  to be completed. Note that the length of the group of digits received by the decoder during this time is also  $\tau$  digits.

- (x) Convert  $\Delta(Z, J)$  to binary form to obtain  $\Delta_d(Z, J)$  and tabulate the results as in Table 5.1.
- (xi) The system can now be synthesised using equation (5.3).

The difference between the hybrid SCALE and the segmented digital SCALE characteristics (see Fig. 5.4) can be reduced by increasing the number of segments. The number of segments ( $M_0 + 1$ ) that produce adequate fitting between the hybrid and the digital SCALE characteristics can be found by computer simulation - starting with a small number of segments (2 segments, say) and increasing it until satisfactory results are obtained (see the design example in section 5.5).

Investigation of Fig. 5.4 reveals that when  $H_{DM}$  is not limited to the digital equivalent of  $H_{max}$ , the digital system does not become overloaded even after the hybrid system has reached its saturation point. This characteristic of the digital system enables it to have an improved dynamic range and better performance under the condition of large input

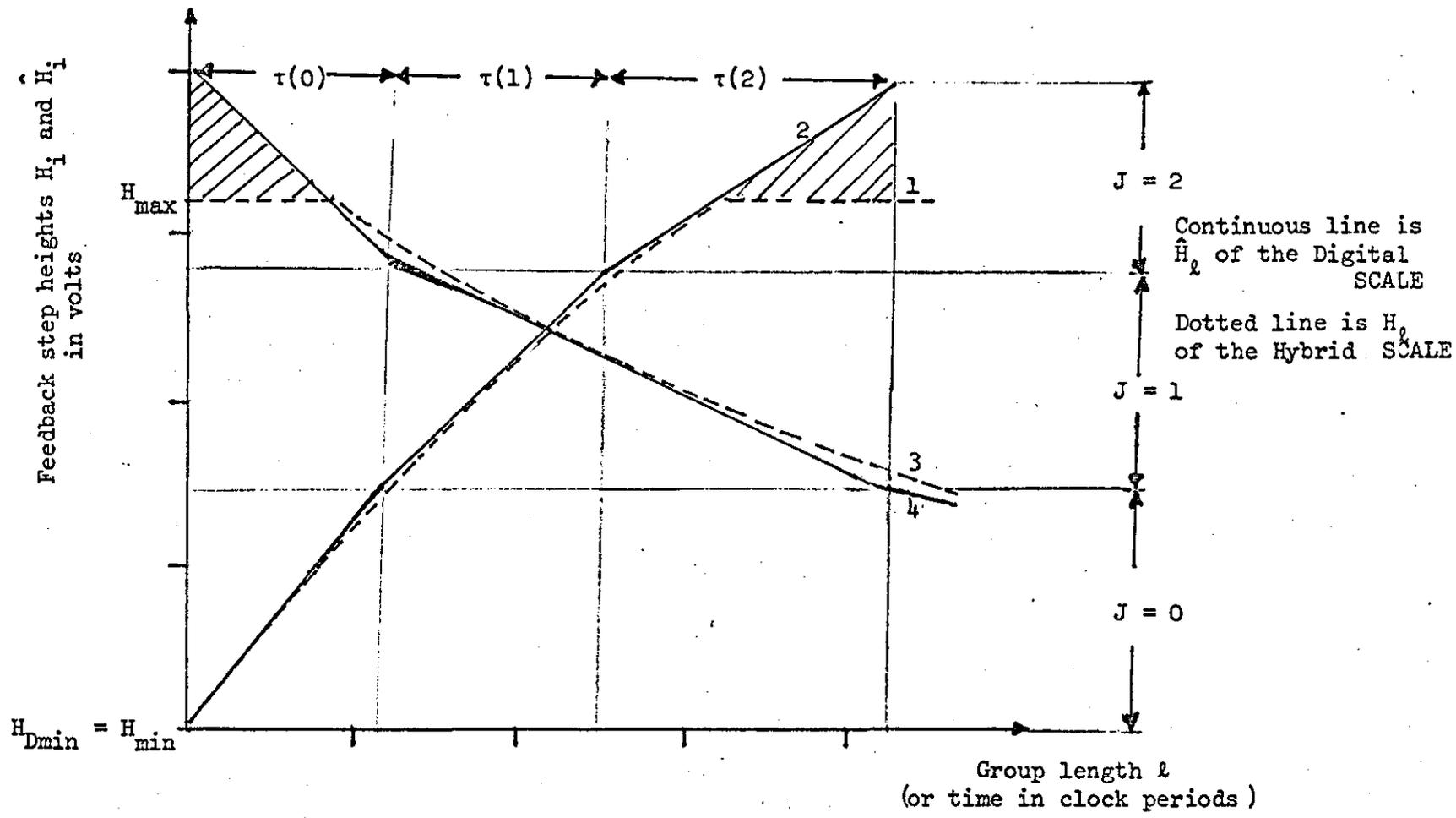


Figure 5.4 Step heights generated by the Hybrid SCALE and the Digital SCALE decoders as functions of the group length  $l$  (or time in clock periods).

- (1) Hybrid SCALE for  $Z = Z_h$
- (2) Digital SCALE for  $Z = Z_h$
- (3) " " " " =  $Z_l$
- (4) " " " " =  $Z_l$

signals.

## 5.5 DESIGN EXAMPLE

The design procedure outlined above is a general one (i.e. given any hybrid SCALE system the procedure can be used to design a digital SCALE equivalent).

A specific case is the digitization of the hybrid SCALE system discussed in Chapters II, III and IV with the circuit parameters given in Table 5.2b.

### 5.5.1 Segmentation and evaluation of the adaptation algorithm

Following the design procedure of Section 5.4, the hybrid SCALE decoder was simulated on a digital computer using the sub-program "SUB-ROUTINE CODEC" discussed in Chapter II. A block of 512 consecutive like digits (see Table 5.2a) (logical ones) followed by a second block of 512 alternate digits (0101...) were fed to the input of the decoder which was being clocked at a bit rate  $f_p$  of 64 Kb/s. This particular value of  $f_p$  was chosen for its compatibility with the line rate recommended by the CCIR for A-law PCM transmission. The magnitude and sign of the PHM output  $H_i$  at the instance of the  $i^{\text{th}}$  clock pulse was recorded for  $i = 1$  to 512, as shown in Fig. 5.4, curves 1 and 3. Investigation of these values of  $H_i$  revealed that:

- (i) As expected, when the input pattern consists of logical ones  $Z = Z_h$  (except during the first two clock periods),  $H$  increases continuously until the overload point is reached, and the polarity of  $H$  is positive. The opposite is true for the case when the input pattern does not consist of consecutive like-digits.
- (ii) The difference between the present and previous values of the magnitude of  $H$  (below absolute overload) varies between  $\approx 1\text{mV}$  and  $\approx 10\text{mV}$  depending on the history of the channel pattern. The







Parameter Designation	Parameter Value
$f_p$	64 Kb/s
$Z_l$	0.4 volts
$Z_h$	2.4 volts
$N_e$	3 stages
$T_2$	10 m. secs.
$T_1 = 1/2\pi.f_1$	0.2 m. sec.
$G$	1.4
$H_{min}$	0.025 volts
$H_{max}$	2.4 volts
$f_a - f_b = B$	4.000 - 0.0 = 4 KHz

Table 5.2b Design parameters of the hybrid SCALE used in the design of the DSCALE

Segment No. J	Magnitude of $\hat{H}_i(Z,J)$ in millivolts
0	$25 < \hat{H}_i < 1024$
1	$1024 < \hat{H}_i < 2304$
2	$2304 \leq \hat{H}_i$

Table 5.3 (Figure 5.5) Segmentation of the SCALE feedback step height magnitude for the design example  
Note: the segmentation is independent of Z in this example

effect of this history can be measured in terms of the group length  $\ell$  or the previous value of  $H$ .

- (iii) A convenient segmentation is to divide the magnitude of  $H$  into three segments (i.e.  $M_0 = 2$  and  $J = 0,1,2$ ) as shown in Table 5.3 (Fig. 5.5). Note that the segment edges  $102^4$  and  $230^4$  are powers of 2 and can be easily detected by digital circuitry. This facilitates the implementation of the segment logic (see the section on implementation).

Note that only the magnitude of  $H$  is shown in Fig. 5.5 for convenience.

Using equation (5.4) and the segment edges defined in Table 5.3 (e.g.  $H_0(Z,J=0) = 25$ ,  $H_M(Z,J=0) = 102^4$ )  $\Delta(Z,J)$  was calculated for all values of  $Z$  and  $J$  ( $Z = 0$  and  $1$ ,  $J = 0, 1$  and  $2$ ). The calculation was performed with the aid of the digital computer and "SUBROUTINE DECODERD" which is listed and discussed in Appendix B. The resultant values of  $\Delta(Z,J)$  define the adaptation algorithm depicted in Table 5.4. "AT EVERY CLOCK INSTANCE  $J$  AND  $Z$  ARE INSPECTED AND THE PREVIOUS VALUE OF  $H$  IS INCREMENTED BY A DISCRETE STEP  $\Delta(Z,J)$ ". The magnitude of the PHM output  $H$  (generated by the hybrid SCALE decoder) as a function of time in clock periods (or group length  $\ell$ ) is plotted in Fig. 5.5, together with its piece-wise linear-approximated curve.

Note that the upper limit of  $H(J=2)$  can be set arbitrarily anywhere above  $H = H_{\max} = 2400\text{mv}$ . In this experiment the absolute overload point of the digital SCALE was allowed to be higher than that of the hybrid one, as indicated by the shaded area of Fig. 5.5. This, as discussed above, improves the dynamic range and performance of the system when subjected to large input signal conditions.

Control Logic Z	Segment Number J	Step height Increment per clock period $\Delta_d(Z,J)$
$Z_l$	0	-1
	1	-3
	2	-8
$Z_h$	0	+8
	1	+6
	2	+4

Table 5.4 The values of  $\Delta_d(Z,J)$  for the design example. The values were found by computer simulation.

## 5.6 COMPUTER SIMULATION (SOFTWARE APPROACH)

In order to test the validity of the proposed digital SCALE system and to evaluate its performance the hybrid SCALE encoder and decoder, and the digital SCALE decoder, were simulated on ICL 1904A digital computer using FORTRAN IV.

The testing and evaluation were performed by encoding sections of various analogue signals (d.c., sinusoidal and Gaussian waveforms) into binary data using the hybrid SCALE encoder. The sampling frequency  $f_p$  was 64 Kb/s, the input signal waveform duration 512 clock periods and the circuit parameters simulated are those shown in Table 5.2(b). The encoder was switched off at the end of each input section (except in the case of Fig. 5.11a) and switched on at the beginning of the next. The blocks of data of  $L(t)$  corresponding to each input signal (d.c., sine waves and Gaussian) shown in Table 5.2(a) were stored on punched cards. These  $L(t)$  blocks of data were then decoded by the hybrid SCALE decoder and the digital equivalent; and the reconstructed analogue signals at the output of the two systems were compared. Further testing of the digital decoder was carried out through measurements of SNR performance compared to that of the optimized hybrid system.

The simulation program was divided into the following steps:

(i) Generation of the  $L(t)$  binary data

The necessary input data for testing the digital SCALE decoder was generated by the digital computer following the procedure described by the flow chart shown in Fig. 5.6.

For each value of "LOOP" (LOOP = 1,2,3,4) an analogue waveform, occupying a time interval of 512 clock periods, is generated. The first three waveforms are generated by "SUBROUTINE SIGNAL" and are respectively: 5 volts dc, zero volts and 1  $KH_2$  sinusoids of amplitude 1.5 volts. The fourth waveform is a band limited Gaussian signal of bandwidth 4  $KH_2$  and standard deviation of unity.

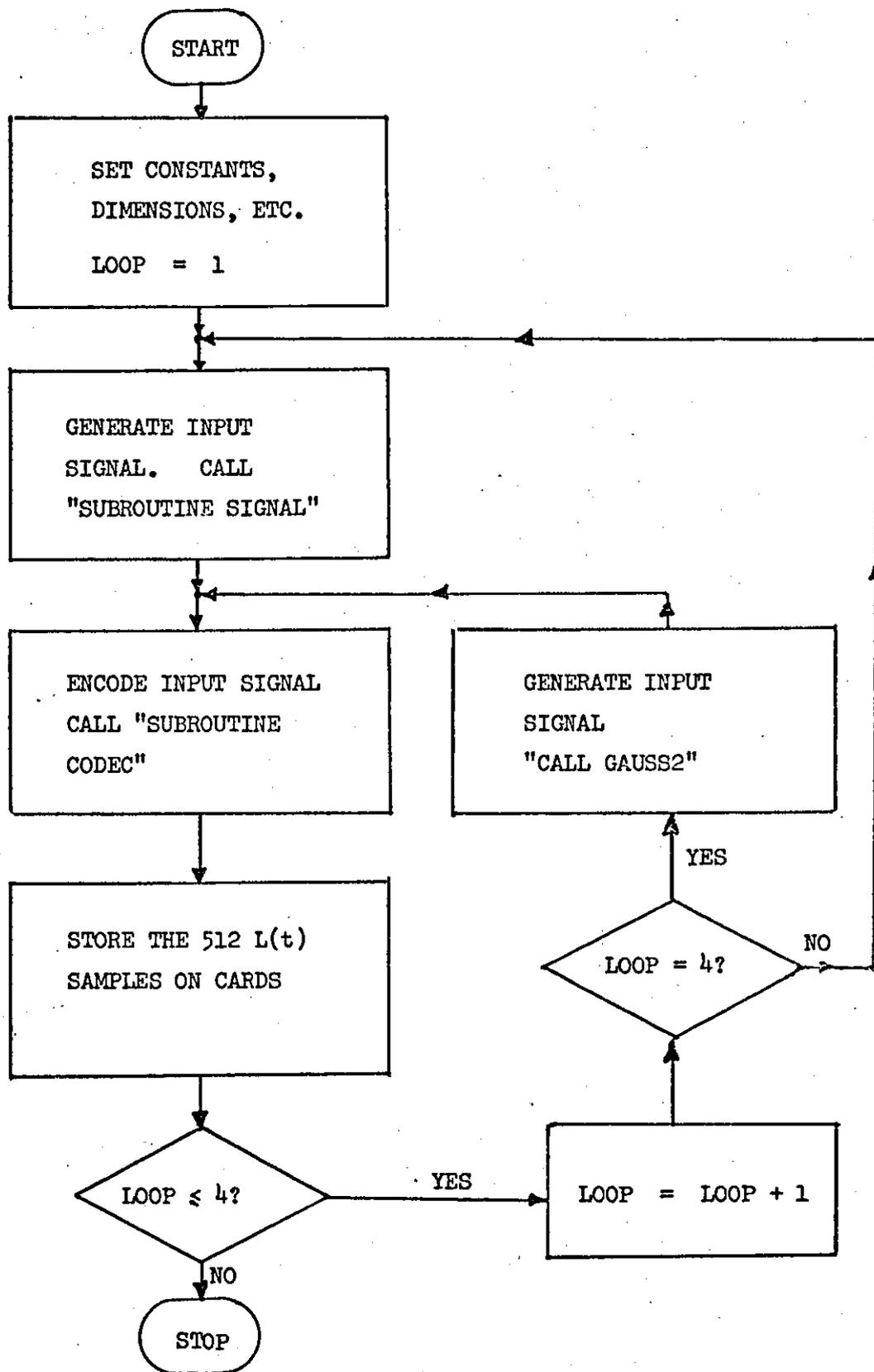


Figure 5.6 Flow chart for the generation of digital signals for testing the SCALE digital decoder

LOOP = 1 The encoded signal is dc of +5V amplitude

LOOP = 2 The input signal is 0 volts

LOOP = 3 The encoded signal is sinusoidal of frequency  $f_s = 1 \text{ KHz}$  and amplitude = 1.5V.

LOOP = 4 The input signal is a bandlimited Gaussian signal of bandwidth 4 KHz and variance = 1.

The waveforms are encoded into binary data by the hybrid SCALE encoder and then punched on cards for storage. The FORTRAN IV subroutines "SIGNAL", GAUSS2" and "CODEC" have already been discussed in Chapter III and are listed in Appendix B.

(ii) Simulation of the digital decoder

The block diagram of the decoder shown in Fig. 5.1(b) was simulated. The control logic in this case monitored the present logical value of the incoming signal  $L_i$  (the value of  $L(t)$  at the instance of the  $i^{\text{th}}$  clock pulse) and the previous two values  $L_{i-1}$  and  $L_{i-2}$  - which are stored in the SR - and produce  $Z$ .  $Z$  is a logical one when  $L_i$ ,  $L_{i-1}$ ,  $L_{i-2}$  are all identical, and it is a logical zero otherwise. Note that the actual voltage value of  $Z$  in this case is not important; only its logical value is required.

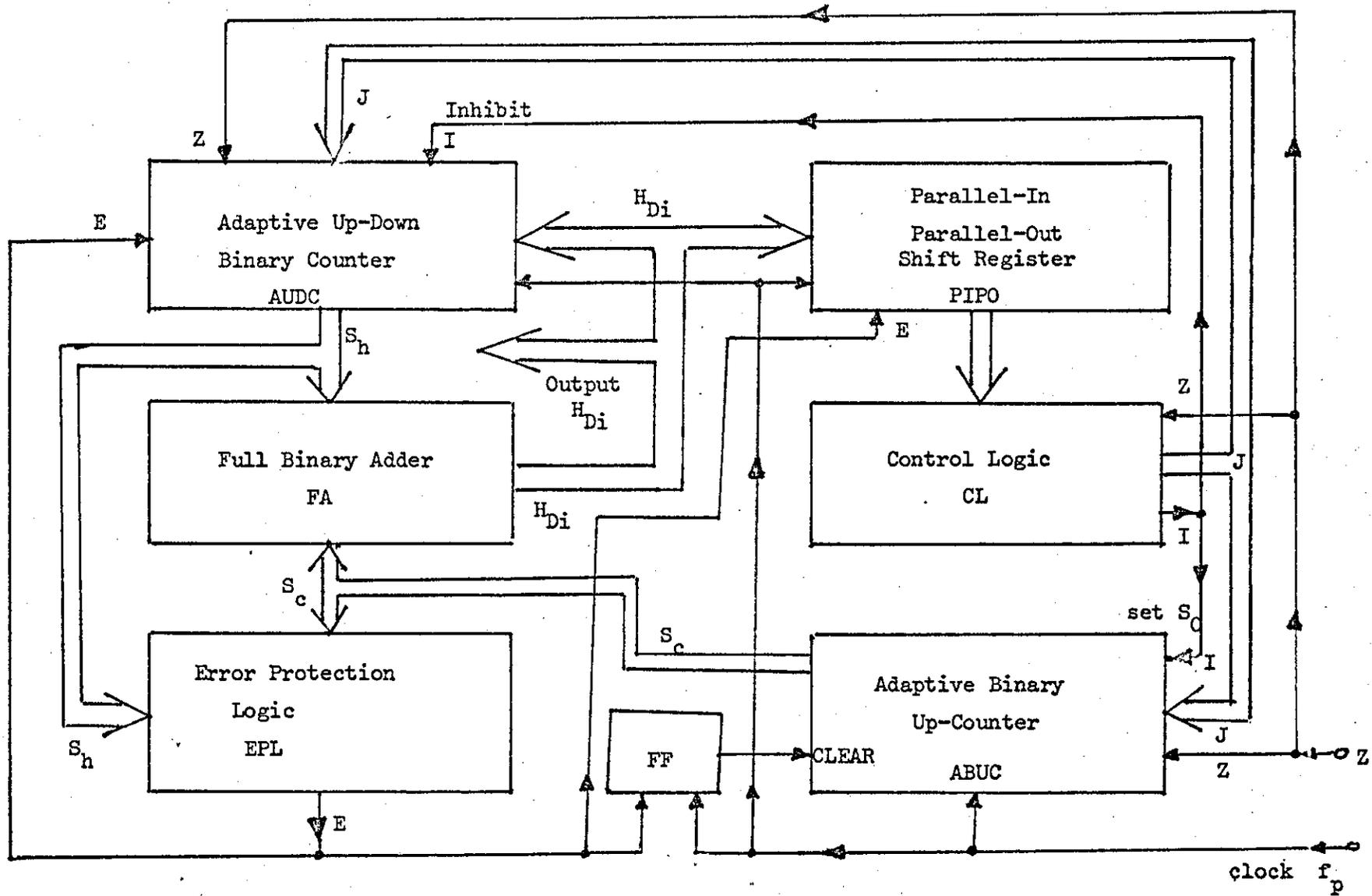
The SR and control logic are represented in the computer program by "SUBROUTINE DIRECTION" which is listed in Appendix B. The polarity digit  $a_p$  is identical to  $L_i$  and indicates that the polarity of the amplitude - represented by the accumulator digital output - is positive when it ( $a_p$ ) is a logical one and the amplitude is negative when it is a logical zero. The output filter is simulated by "SUBROUTINE FILTER" which has already been described in Chapter III.

Simulation of the remaining two blocks of the decoder - the adaptive accumulator AA and the digital-to-analogue converter D/A - is discussed below:

(1) The Adaptive Accumulator

The AA which was simulated employed the counting approach. This approach satisfied equation (5.3b) and results in a system block diagram which is shown in Fig. 5.7.

Below overload equation (5.3b) can be written as:



80a

Figure 5.7. Block diagram of the adaptive digital accumulator using the counting techniques

$$\begin{aligned}
H_{Di} &= H_{D(i-1)} + \Delta_d(Z,J) \dots H_{D(i-1)} > H_{\min} & (5.5) \\
&= H_{D\min} \dots H_{D(i-1)} \leq H_{\min}
\end{aligned}$$

where J is defined by Table 5.3 in terms of the decimal equivalent of  $H_{D(i-1)}$ . The decimal equivalent of  $\Delta_d(Z,J)$  is defined by Table 5.4. To simplify the implementation of equation (5.5) in the form described by the block diagram of Fig. 5.7, equation (5.5) and Table 5.4 were modified as shown in equation (5.6) and Table 5.5 below.

$$\begin{aligned}
H_{Di} &= H_{D(i-1)} + \Delta_d(Z,J) \\
&= H_{D(i-1)} + \Delta'_d(Z,J) + \Delta''_d(Z,J) \dots H_{D(i-1)} > H_{D\min} \\
&= H_{D\min} \dots H_{D(i-1)} \leq H_{D\min}
\end{aligned} \quad (5.6)$$

where 
$$\Delta_d(Z,J) = \Delta'_d(Z,J) + \Delta''_d(Z,J) \quad (5.7)$$

The equivalent decimal values of  $\Delta'_d(Z,J)$  and  $\Delta''_d(Z,J)$  are respectively defined by Tables 5.5a and 5.5b.

Now in Fig. 5.7 the functions  $H_{Di}$ ,  $H_{D(i-1)}$ ,  $S_o$ ,  $S_h$  and  $S_c$  are all 12-digit parallel binary words having the meanings given below:

$H_{Di}$  is the present DSCALE word representing  $\hat{H}_i$  (the estimate of  $H_i$  - the hybrid SCALE present step height) in binary notation. In binary  $H_{Di}$  consists of the sequence  $(a_{12}, a_{11}, \dots, a_1)$  where  $a_1$  is the least significant digit LSD.

Therefore:

$$H_{Di} = \hat{H}_i = \sum_{j=1}^{12} a_j 2^{j-1} \quad (5.8)$$

and  $a_j = \{0,1\}$ .

$H_{D(i-1)}$  is the previous word of the DSCALE and represents  $\hat{H}_{i-1}$  (the estimate of  $H_{i-1}$  - the hybrid SCALE previous step height)

Control logic output Z	Segment No. J	Partial increment $\Delta'_d(Z,J)$
$Z_l$	0	0
	1	+1
	2	0
$Z_h$	0	0
	1	2
	2	0

(b)

Control logic output Z	Segment No. J	Partial increment $\Delta''_d(Z,J)$
$Z_l$	0	-1
	1	-4
	2	-8
$Z_h$	0	+8
	1	+4
	2	+4

(a)

Table 5.5 Partitioned value of the step height increment  $\Delta_d(Z,J)$  into  $\Delta'_d(Z,J) + \Delta''_d(Z,J)$ . The sum of these two parts for any of the combinations of Z and J produces the corresponding value of  $\Delta_d(Z,J)$  given in Table 5.4.

in binary notation.  $H_{D(i-1)}$  consists of the binary sequence  $(b_{12}, b_{11}, \dots, b_1)$  where  $b_1$  is the LSD.

Therefore:

$$H_{D(i-1)} = \hat{H}_{i-1} = \sum_{j=1}^{12} b_j 2^{j-1} \quad (5.9)$$

and  $b_j = \{0,1\}$ .

$S_0$  is the minimum value of  $H_{Di}$  (i.e.  $H_{Dmin}$ ) and represents the initial idling condition step height in the hybrid SCALE except that  $S_0$  is in binary notation.

Therefore:

$$S_0 = H_{Dmin} = \sum_{j=1}^{12} d_j 2^{j-1} \quad (5.10)$$

and  $d_j = \{0,1\}$ .

Now in the hybrid SCALE system  $H_{min}$  was 25 m. volts. In the DSCALE, however,  $H_{Dmin}$  was made 31 m. volts. This small change simplifies the logic circuit which is to be designed to detect the condition when  $H_{Di} = H_{Dmin}$ ; but its effect on the output signal from the DSCALE decoder was shown experimentally to be negligible.

$S_0$  is therefore a constant binary word in which the five least significant digits,  $d_1, d_2, d_3, d_4$  and  $d_5$ , are set to logical ones while the rest of the word digits ( $d_6, d_7, \dots, d_{12}$ ) are set to logical zeros. i.e.

$$(d_{12}, d_{11}, d_{10}, d_9, d_8, d_7, d_6, d_5, d_4, d_3, d_2, d_1) \equiv (0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1)$$

which has the decimal value of 31.

$S_c$  is the contribution to  $H_{Di}$  from  $\Delta_d''(Z, J)$ , see equation (5.6). It is a 12-digit binary word consisting of the sequence  $(e_{12}, e_{11}, \dots, e_1)$  and represents the sum of all the increments  $\Delta_d''(Z, J)$  since the

counter ABUC was cleared. In decimal notation  $S_c$  can be written as:

$$S_c = \sum_{j=1}^{12} e_j 2^{j-1} \quad (5.11)$$

where  $e_1$  is the LSD and  $e_j = \{0,1\}$ .

The contents of the binary counter ABUC represented by the sequence  $(e_{12}, e_{11}, \dots, e_1)$  are added to the contents of the adaptive up/down counter AUDC whenever an overload condition (i.e.  $H_{D(i-1)} = H_{Dmax}$ ) occurs or when the AUDC itself becomes overloaded or when its contents are equal to or less than  $H_{Dmin}$ . After this addition is completed the digits of the sequence  $(e_{12}, e_{11}, \dots, e_1)$  are all set to logical zeros.

$S_h$  is the sum of all the increments  $\Delta_d^i(Z, J)$  (i.e. all the history of  $\Delta_d(Z, J)$  plus the contribution to  $H_{Di}$  from  $\Delta_d^i(Z, J)$  which might have been transferred from the adaptive up-counter ABUC to the adaptive up-down counter AUDC during the history of the system operation.

$S_h$  consists of the binary sequency  $(c_{12}, c_{11}, c_{10}, \dots, c_1)$  and can be written in decimal notation as:

$$S_h = \sum_{j=1}^{12} c_j 2^{j-1}$$

where  $c_1$  is the LSD and  $c_j = \{0,1\}$ .

The operation of the system in Fig. 5.7 is as follows:

Whenever  $H_{D(i-1)}$  is equal to or less than  $H_{Dmin}$  and  $Z$  is a logical zero, the control logic CL produces a voltage pulse  $I$ . This pulse sets the contents of the adaptive up-counter (ABUC) to  $S_0$  and inhibits the adaptive up-down counter (AUDC) from any further counting. The "set" signal shown at the ABUC input is generated by a logic circuit inside

the control logic block (CL). This logic circuit simply monitors the  $H_{D(i-1)}$  sequence ( $b_{12}, b_{11}, \dots, b_1$ ) and Z, and generates a pulse I when  $b_{12}, b_{11}, \dots, b_7$  and  $b_6$  are all logical zeros and Z is also a logical zero.

When Z becomes a logical one the "set" pulse I disappears and the adaptive counters AUDC and ABUC start counting according to the rules given in Tables 5.5a and 5.5b, respectively. This counting is performed under the control of the clock (which is running at  $f_p$  Kb/s).

Every clock period ( $T_p$ ) the contents of the adaptive binary counters AUDC and ABUC ( $S_h$  and  $S_c$ ) are added together by the binary full adder (FA) to produce the parallel binary word  $H_{Di}$  which is the output of the adaptive accumulator. i.e.

$$H_{Di} = S_c + S_h \quad (5.13)$$

$H_{Di}$  is transmitted to the next processing stage (e.g. D/A conversion, SCALE-to-PCM conversion, etc.) and also loaded into the 12-bit parallel-in-parallel out shift register (PIPO). The PIPO acts as one word digital delay element which provides  $H_{D(i-1)}$  at its output every clock period. The output  $H_{D(i-1)}$  of the PIPO is inspected by the control logic CL every clock period and the segment number J is generated according to Table 5.3.

In this example  $J = 0, 1$  or  $2$  and can, therefore, be represented in binary notation by 2 parallel binary digits which form the binary sequence ( $g_1, g_2$ ).

Therefore

$$J = \sum_{j=1}^2 g_j 2^{j-1} \quad (5.14)$$

where  $g_j = \{0, 1\}$  and  $g_1$  is the LSD.

Now because the ABUC is an Up-counter overflow will sooner or later take place. Also because the AUDC is an Up-down counter its

content can go on decreasing even below zero. These overflow and underflow conditions can cause serious errors and degrade the performance of the system. To avoid these errors, the error protection logic (EPL) was included to detect the overflow in the ABUC and the underflow in the AUDC.

The EPL inspects the contents  $S_h$  and  $S_c$  of the AUDC and the ABUC, respectively, and generates a pulse E when:

(a) The 9 most significant digits of  $S_c$ , i.e.  $(c_{12}, c_{11} \dots c_4)$ , are all logical ones

and/or

(b) The 8 most significant digits of  $S_h$ , i.e.  $(c_{12}, c_{11} \dots c_4)$  are all logical zeros.

Note that the AUDC maximum decrement ( $\Delta_d(Z=0, J=2)$ ) is -8 m. volts and the maximum ABUC increment ( $\Delta_d(Z=1, J=1)$ ) is +2 m. volts. (See Tables 5.5 a, and b.)

Therefore

$$E = \begin{cases} 1 & \text{when } (e_{12}, e_{11} \dots e_4) \text{ are all logical} \\ & \text{ones, and/or } (c_{12}, c_{11} \dots c_4) \text{ are all} \\ & \text{logical zeros.} \\ 0 & \text{otherwise} \end{cases} \quad (5.15)$$

When E is a logical one the sum  $H_{Di} = S_h + S_c$  is formed in the normal way, passed to the PIPO and thence loaded into the AUDC. Now, with the aid of the D-type flip-flop (FF) the ABUC is cleared at the next clock instance.

Using the properties and the mathematical equations of the adaptive accumulator (AA) shown in Fig. 5.7, the AA was simulated on the digital computer using three subprograms which were written specially for this purpose.

The subprograms are: "SUBROUTINE HISTORY", "SUBROUTINE AUCOUNTER" and "SUBROUTINE ADDER". These are listed and described in Appendix B.

(2) The Digital-to-Analogue Converter

The function of the D/A converter appearing in Fig. 5.1(b) is to accept the 12-parallel binary digits ( $a_{12}, a_{11} \dots a_1$ ) - representing the input signal at the input of the remote SCALE encoder - and reconstruct an analogue signal  $\hat{H}_i$  which is delivered to the input port of the output filter at the  $i^{\text{th}}$  clock pulse instance. The polarity of  $\hat{H}_i$  is positive when " $a_p$ " is a logical one and it is negative when " $a_p$ " is a logical zero. The symbol  $V_R$  in Fig. 5.8 represents the reference voltage for the D/A and was made (in this example) equal to 4.096 volts.  $V_R$  is the maximum voltage step that can appear at the output of the D/A converter.

Referring to Fig. 5.8, the output signal of the D/A at the  $i^{\text{th}}$  clock instance can be written as

$$\hat{H}_i = \pm \sum_{j=1}^{12} a_j 2^{j-1} \quad (5.16)$$

where  $a_j = \{0,1\}$  and  $a_1$  is the LSD.

$\hat{H}_i$  is in mv. and approximately equal to  $H_i$  generated by the PHM in the hybrid encoder.

Using (5.16) the D/A was simulated in FORTRAN by a subprogram which was named "DAC". "SUBROUTINE DAC" is listed in Appendix B.

Summary of the simulation program of the digital SCALE decoder is shown in the form of a flow chart in Fig. 5.9.

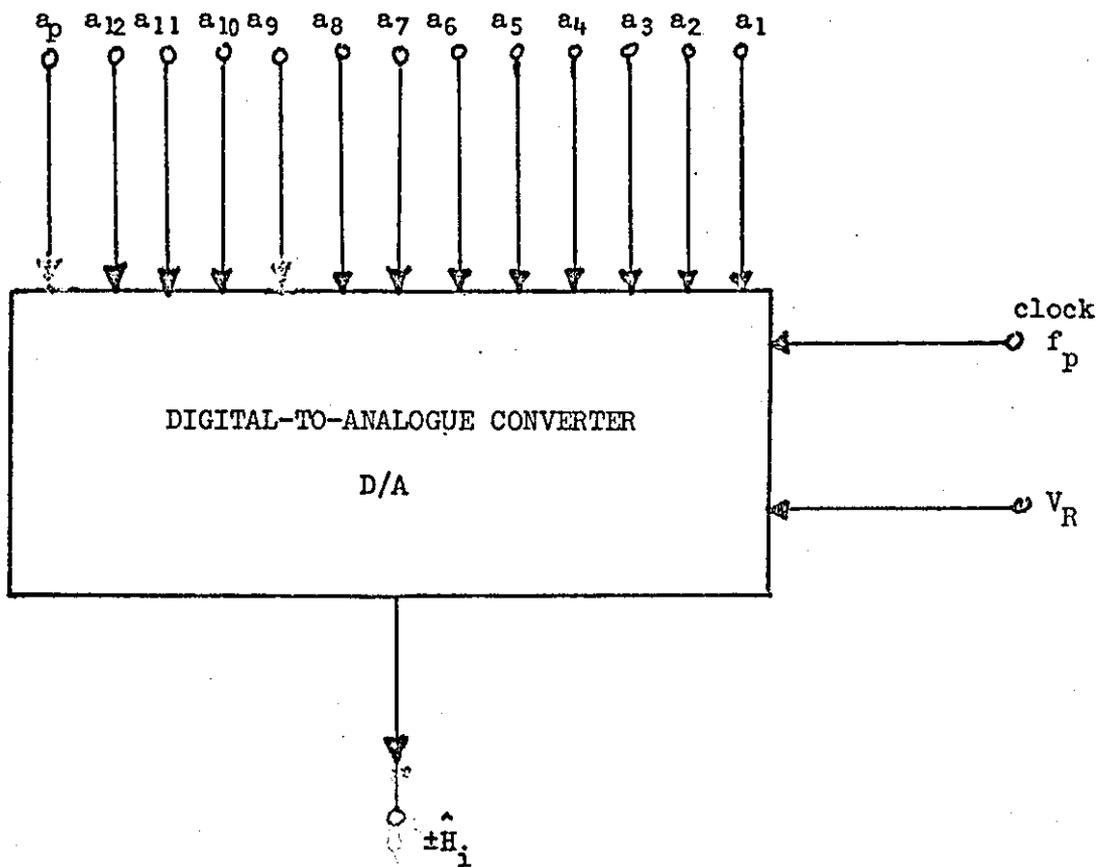


Figure 5.8 Block diagram of the Digital-to-analogue converter D/A.  
 $a_1, a_2, \dots, a_{12}$  are the input binary sequence representing the DSCALE digital word  $H_{Di}$ ,  $a_p$  is the sign bit and  $\hat{H}_i$  is the analogue equivalent of  $H_{Di}$ .

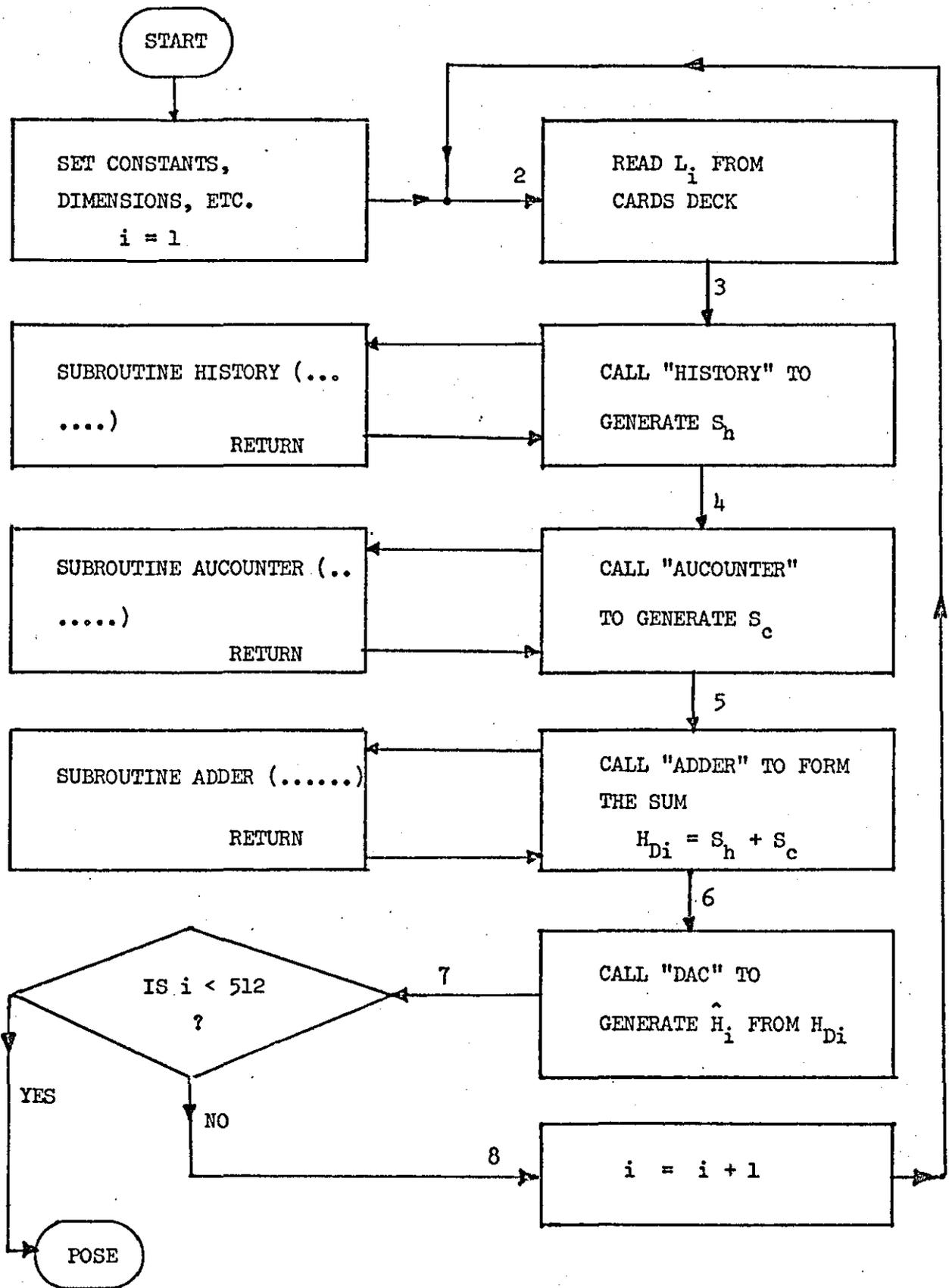


Figure 5.9 Flow chart of the digital SCALE decoder simulation program

### (3) Testing and calculating the system performance

#### (i) Waveform reproduction

The FORTRAN IV program which was compiled for testing the effect of digitizing the pulse-high modulated signal  $\hat{H}(t)$  on the final output filtered waveform  $\hat{x}_2(t)$  is summarized by the flow chart shown in Fig. 5.10. The program was run 4 times and in each run a block of 512 data bits were read, decoded simultaneously by the hybrid and the digital SCALE decoders, and the corresponding filtered output waveforms from the two different decoders were plotted on the same axis so that they could be easily compared.

The binary data blocks represent the waveforms which were encoded according to the procedure defined by Fig. 5.6.

The comparison graphs are shown in Figs 5.11a, 5.11b and 5.11c for d.c., sinusoidal and bandlimited Gaussian noise signals respectively.

It can be seen from these graphs that the waveforms obtained by analogue and those obtained by digital processing of the  $L(t)$  signals are almost identical.

#### (ii) Signal-to-Noise Ratio Performance

The signal-to-noise ratio SNR for the digital SCALE before overload can be calculated using equation (2.24); but with  $H_p$  replaced by  $\hat{H}_p$  the average signal at the output of the D/A converter). This leads to the relation

$$\text{SNR}_d = R_c \left( \frac{\sigma_x}{\hat{H}_p} \right)^2 \quad \sigma_x \leq \hat{H}_p \quad (5.17)$$

where  $\sigma_x$  is the RMS value of the input signal voltage waveform and

$$R_c = \frac{3f_p^3}{8\pi k_c (3f_1^2 B + f_a^3 - f_b^3)}$$

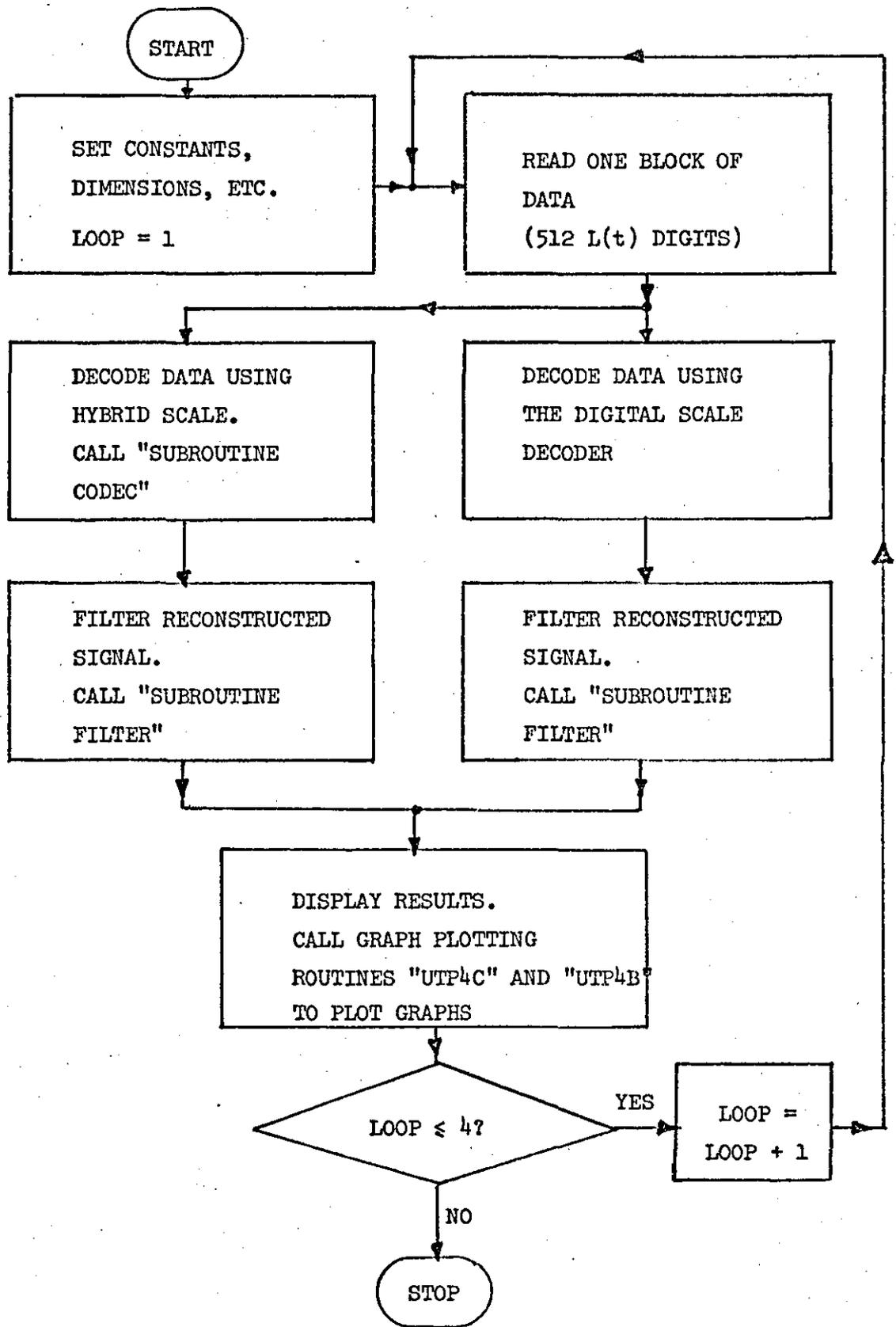


Figure 5.10 Flow chart of the program used for testing the digital SCALE decoder (comparison with the hybrid decoder).

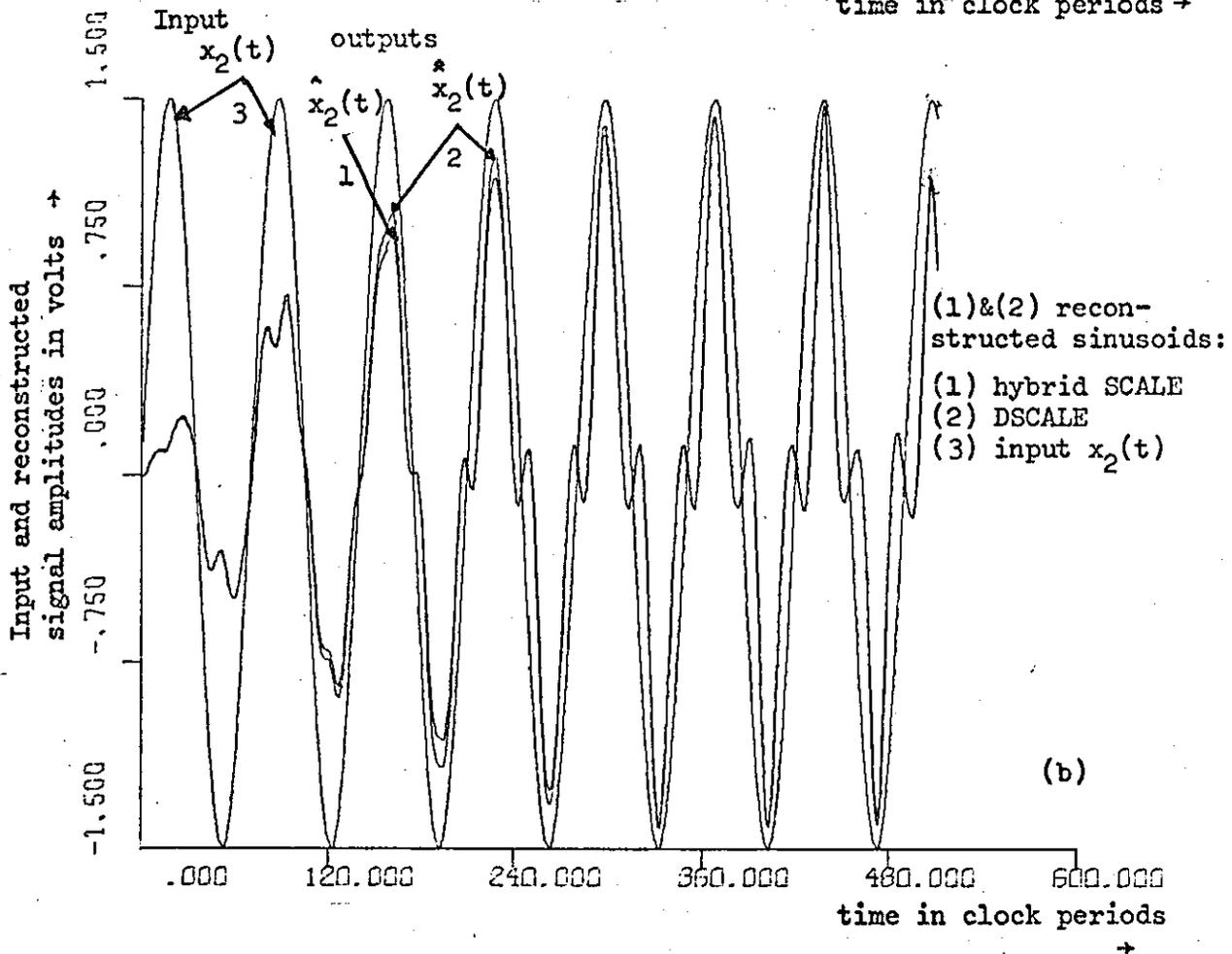
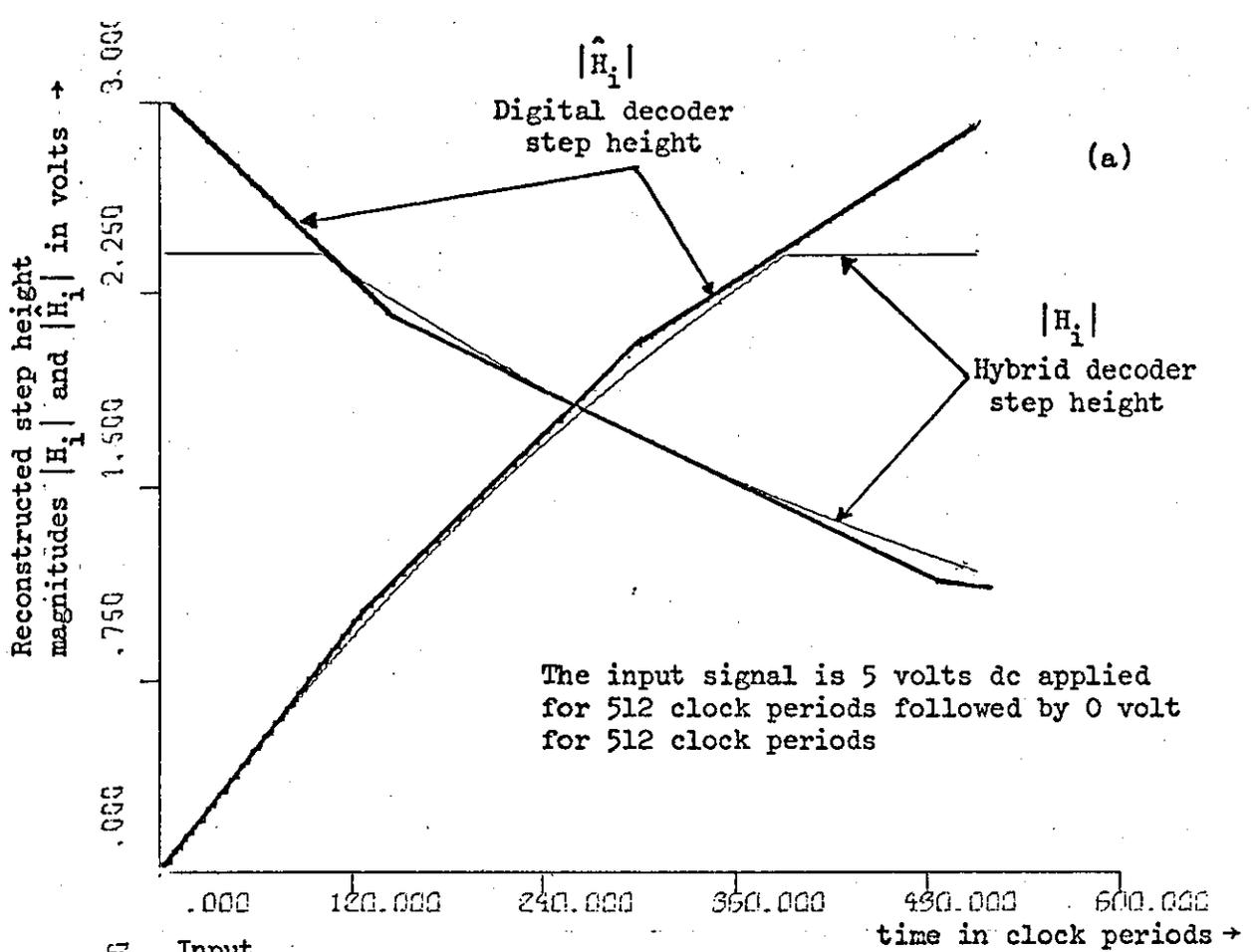


Figure 5.11 Comparison between the hybrid and the digital SCALE systems using dc input signal in (a) and sinusoid of frequency  $f_s = 1 \text{ KHz}$ , 1.5 volts amplitude in (b)

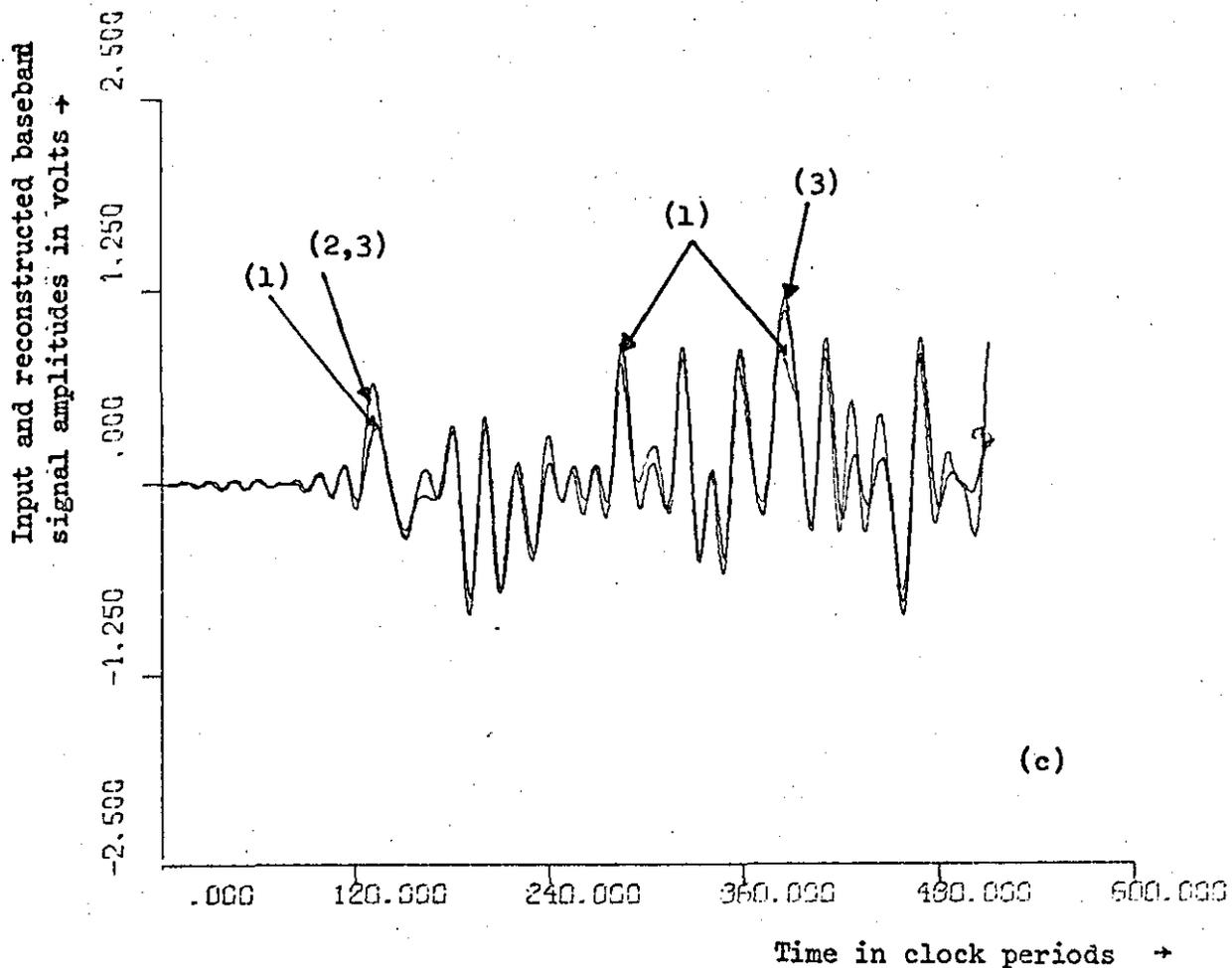


Figure 5.11 Comparison between the hybrid and the digital SCALE systems

(c) The input and the reconstructed baseband signal waveforms comparison when the input (curve (1)) is a band limited Gaussian noise (bandwidth 0 to 4 KHZ)

Curve (1) Input to the HSCALE encoder

Curve (2) Output of the HSCALE decoder

Curve (3) Output of the DSCALE decoder

HSCALE = hybrid SCALE

DSCALE = digital SCALE

Above overload the  $SNR_d$  can be calculated as in equation (2.26),

i.e.

$$SNR_d = R_c \left( \frac{\hat{H}_p}{\sigma_x} \right)^2 \quad \sigma_x > \hat{H}_p \quad (5.18)$$

Now in this experiment  $f_1$ ,  $f_a$ ,  $f_b$ ,  $f_p$ , and  $B$  are defined by Table 5.2.  $k_c$  can be found from Fig. 4.10f (the graph of  $k_c$  versus  $f_p$ . This gives  $k_c(f_p = 64 \text{ Kb/s})$  as approximately 1.1).

Like the hybrid SCALE - the digital SCALE was designed in such a manner that the feedback step-height  $\hat{H}_p$  varies in sympathy with the variations in the power residing in the input waveform. This means that for every value of  $\sigma_x$  (the RMS value of the input signal calculated over a period of time which is  $\approx 1$  pitch period), there exists a corresponding value of  $\hat{H}_p$  (which is the average value of  $\hat{H}_i$  over all values of  $i$ ).

To find the relationship between  $\sigma_x$  and  $\hat{H}_p$  - and hence calculate the  $SNR_d$  as a function of the input power - the digital computer was employed as follows.

- (i) The subprogram "SUBROUTINE GAUSS2" was employed to generate a Gaussian signal  $x_1(t)$  of variance  $\sigma^2$  and occupying a time period  $\tau'_p = 512 T_p$ . The Gaussian signal was then bandlimited by the simulated digital filter - represented by "SUBROUTINE FILTER" - and the RMS value of the input signal  $\sigma_x$  was computed.

$$\sigma_x = \sqrt{\frac{\sum_{i=1}^{\ell} x_i^2}{\ell}} \quad (5.19)$$

where  $\ell$  is the input duration in clock periods (i.e.  $\ell = 512$ ) and  $x_i$  is the signal value seen at the instance of the  $i^{\text{th}}$  clock pulse, at the output port of the input filter.

- (ii) The 512 samples  $x_i$  were applied to the simulated hybrid SCALE encoder represented by the subprogram "SUBROUTINE CODEC" to generate 512 binary digits  $L_i$ .
- (iii) The block of data generated was applied to the simulated digital SCALE decoder - represented by "SUBROUTINE DIRECTION", "SUBROUTINE HISTORY", "SUBROUTINE AUCOUNTER", "SUBROUTINE ADDER" and "SUBROUTINE DAC" - and the average value of the reconstructed signal  $\hat{H}_p$  - before filtering - was computed.

$$\hat{H}_p = \left[ \sum_{i=1}^{\ell} |H_i| \right] / \ell \quad (5.20)$$

where  $\ell = 512$  and  $\hat{H}_i$  is the output of the D/A converter at the  $i^{\text{th}}$  clock instance.

- (iv) The steps (i), (ii) and (iii) above were repeated for 24 values of  $\sigma$  between 0 and 10 and the corresponding computed values of  $\sigma_x$  and  $\hat{H}_p$  were noted.

This procedure for calculating  $\hat{H}_p$  as a function of  $\sigma_x$  is summarized by the flow chart of the computer program, shown in Fig. 5.12, which was used for this purpose.

The computation of the  $\text{SNR}_d$  as a function of the input power was performed by substituting the values of  $\sigma_x$  and the corresponding values of  $\hat{H}_p$  (noted from the computation carried out by the above procedure) inside the computer. This was achieved by reading in the values of  $\sigma_x$  and the corresponding values of  $\hat{H}_p$  which were already punched on cards (see Fig. 5.12, statement No. 19) and then calling the subprograms "SUBROUTINE COMPARE" and "SUBROUTINE SNRACL" to implement equations (5.17) and (5.18) into FORTRAN.

The procedure flow chart for this operation is shown in Fig. 5.13.

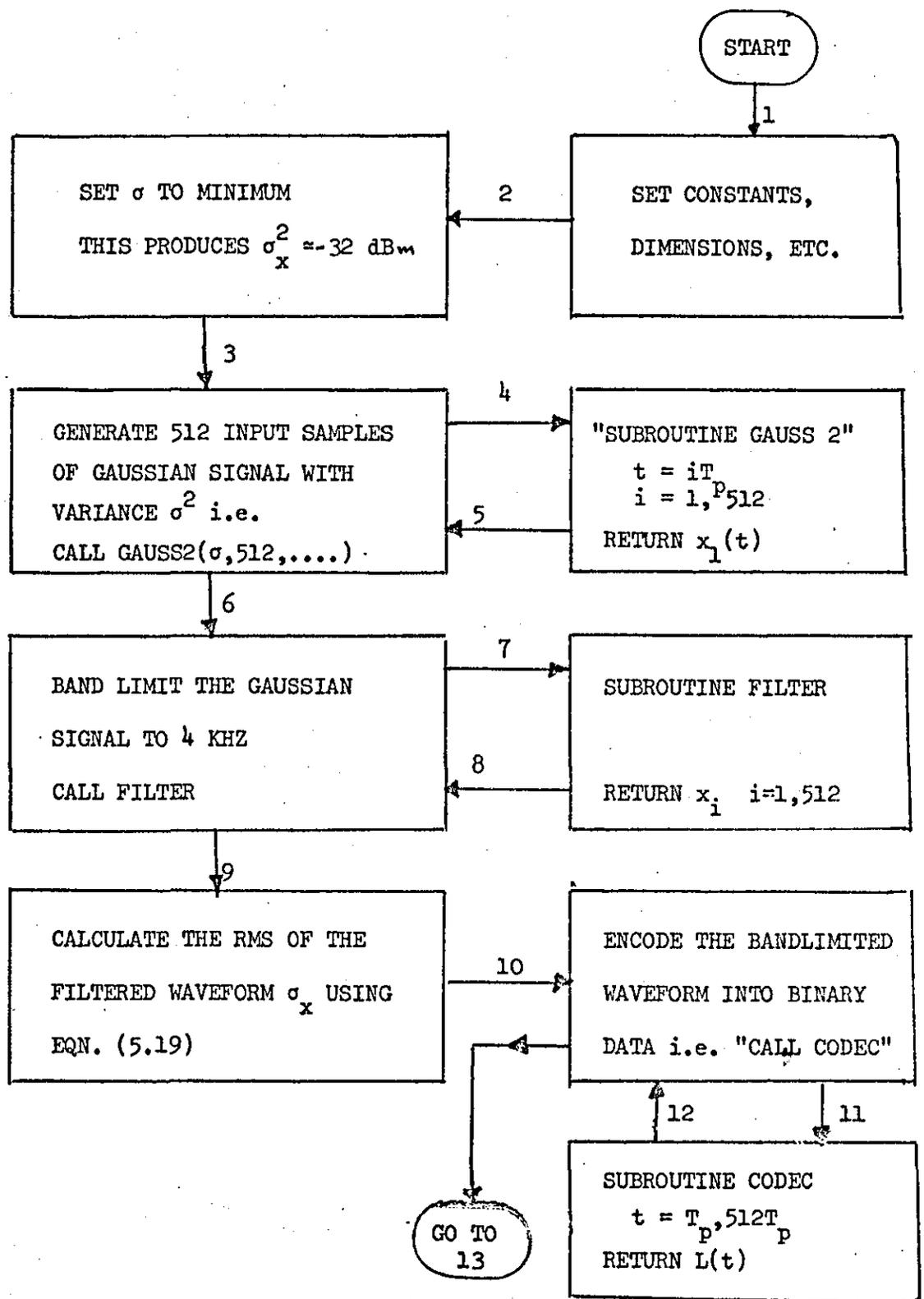
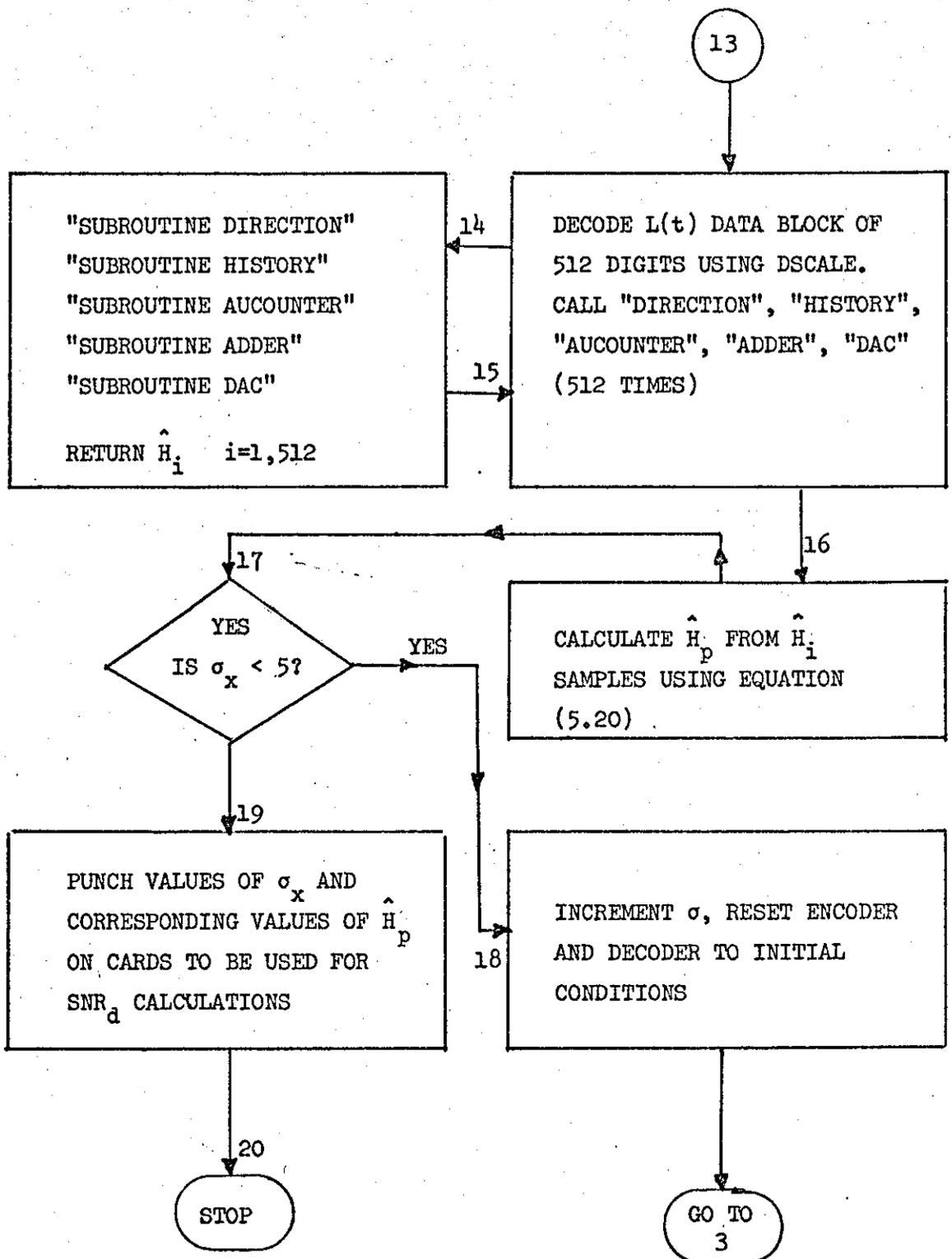


Figure 5.12 Flow diagram of the computer procedure used for calculating  $\sigma_x$  (the input rms voltage) and  $\hat{H}_p$  (the average of the magnitude of  $\hat{H}_i$ ) over 512 clock periods



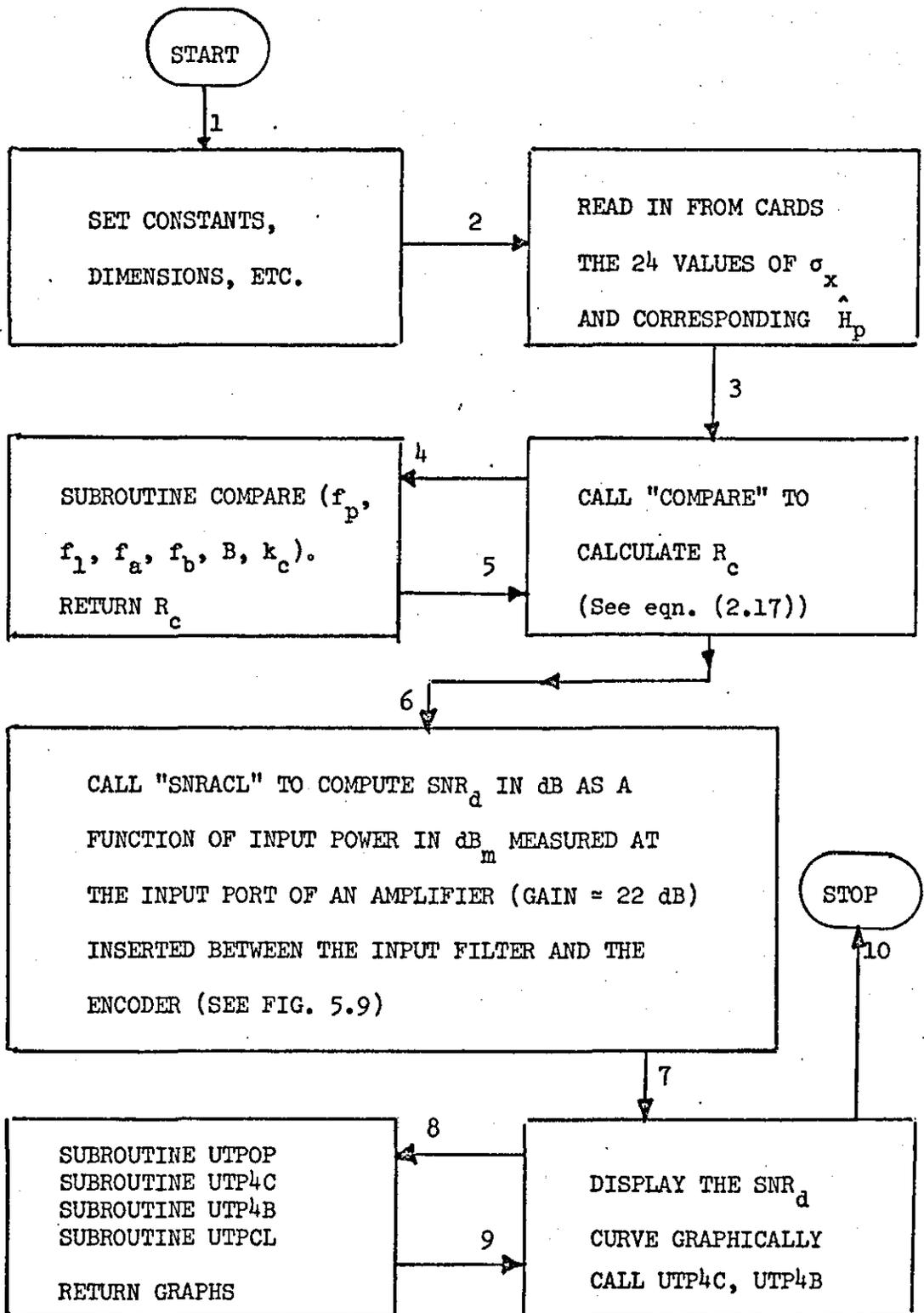


Figure 5.13 Computation of  $SNR_d$  as a function of the input power.  
(Computer program flow chart)

Now the hardware model of the hybrid SCALE, discussed in Chapter IV, employs an input amplifier which accepts the filtered signal  $\sigma'_x$  and produces amplified signal  $\sigma_x$ . The amplifier gain is =22 dB.

To enable the comparison between the practical and simulation results, "SUBROUTINE SNRACL" calculates the signal-to-noise ratio as a function of the equivalent input signal  $\sigma'_x$  which when amplified by 22 dB produces  $\sigma_x$  (see Figs 4.9 and 5.14). The subprogram "SUBROUTINE SNRACL" also converts  $\sigma'_x$  from dB to dBm in 600 ohms.

The signal-to-noise ratio curves  $SNR_d$  and SNR for the digital and hybrid SCALE systems, computed as functions of input power in dBm and the input filter bandwidth in  $KHz$ , are displayed in Fig. 5.15. It can be seen from Fig. 5.15 that, compared with the hybrid SCALE, the digital SCALE system has clearly improved signal-to-noise ratio performance when the input signal is small ( $\sigma'_x \leq 1$ ). For larger input signals, however, the hybrid SCALE system performs better. (Note that in this calculation the DSCALE was assumed to reach the absolute overload point at the same input which causes the hybrid SCALE PHM to limit.) These effects are mainly due to:

- (1) The digital system employs smaller feedback step heights when the input signal is small ( $\hat{H}_i < H_i$  for  $\hat{H}_i \leq 500$  mv.).
- (2) The digital system employs larger feedback step heights when the input signal is large. (See Fig. 5.6a.)

Further investigation of Fig. 5.10 reveals that on average the two systems are very close to one another and it can be concluded that the proposed digital SCALE is at least as good as the hybrid one. In addition, in the digital SCALE the number of discrete and analogue components has been reduced and the reconstructed signal, which represents the input waveform, is available in digital form. This latter characteristic of the digital SCALE is the most important advantage of

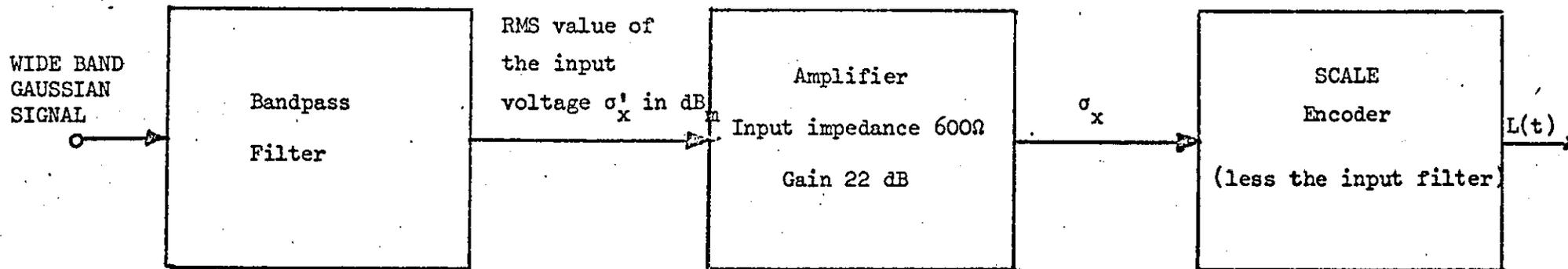


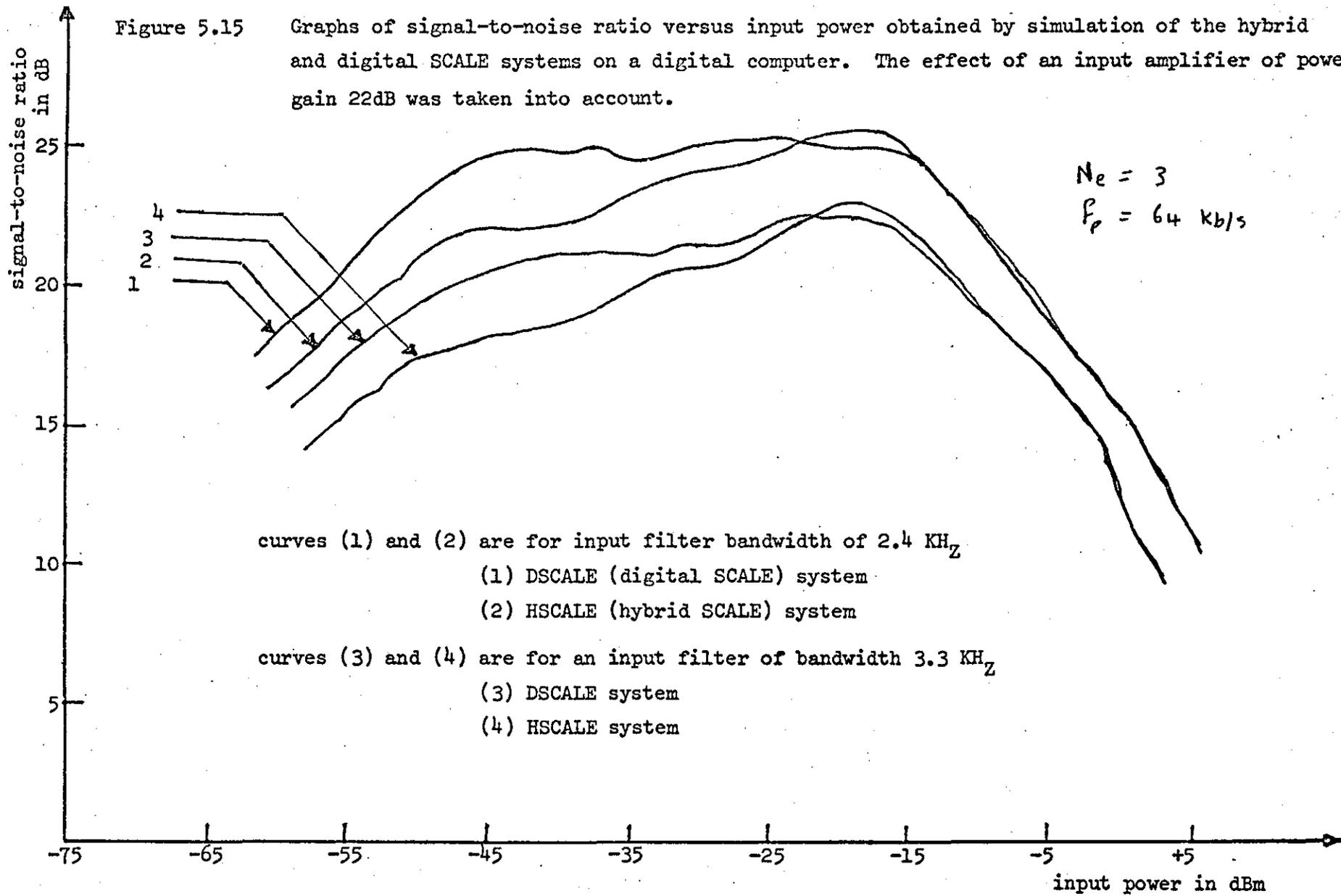
Figure 5.14 Modification to the simulated input signal to enable comparison between the practical and simulated models

$\sigma_x$  = The RMS value of the input voltage calculated over a period  $\tau_p = 512T_p$

$\sigma'_x$  = The input RMS voltage which when amplified by 22 dB produces  $\sigma_x$

In the computer simulation  $\sigma_x$  was calculated first and was then attenuated by 22 dB to obtain  $\sigma'_x$ . This was then used in the calculation of the  $SNR_d$  as a function of the input power ( $\sigma'_x$  in  $dB_m$ ).

Figure 5.15 Graphs of signal-to-noise ratio versus input power obtained by simulation of the hybrid and digital SCALE systems on a digital computer. The effect of an input amplifier of power gain 22dB was taken into account.



the system because it makes digital processing of the SCALE system binary data possible. In the next chapters the results concluded above will be employed to the design of all-digital converters which accept the SCALE binary data and produce Linear and A-law PCM and vice versa.

#### 5.7 HARDWARE APPROACH TO THE IMPLEMENTATION OF DSCALE

The aim of this section is to take the DSCALE design, discussed above, one step further by describing a hardware model of the system. The model must satisfy the design tables and equations derived in Sections 5.3 and 5.4; and it must employ off-the-shelf, economically available integrated circuits. The system block diagram is shown in Fig. 5.1. The forward path need not be discussed because it has already been investigated in Chapter II and IV. The part which are to be discussed in detail are the local and the remote decoders shown in Fig. 5.1b. The hybrid SCALE, the equivalent of which was designed, employs a 3-bit long shift register. The implementation design procedure can be partitioned as follows.

##### (1) Generation of the control signal Z

The Z control signal can be generated from the binary data  $L_i$  using the circuit shown in Fig. 5.1b. This circuit is similar to that employed in the experimental hybrid SCALE system discussed in Chapter II. The main differences between the two circuits are:

- (i) The circuit in Fig. 5.1b was designed for fixed-length shift register of 3 bits. Therefore, the selection logic SL of Fig. 2.3 is not necessary.
- (ii) The value of Z is 5 or 0 volts as compared with that of Fig. 2.3 of Chapter II where Z varies between 0.4 and 3.8 volts. This is because in the case of the DSCALE the actual value of Z is of no

significance and therefore only its logical value is required.

The circuit shown in Fig. 5.16 accepts a binary digit  $L_i$  at the  $i^{\text{th}}$  clock instance and produces a control signal  $Z$  which, as in the case of the hybrid SCALE, is a function of the present and two previous logical values of  $L(t)$ . i.e.

$$Z = (L_i \cdot L_{i-1} \cdot L_{i-2}) + (\bar{L}_i \cdot \bar{L}_{i-1} \cdot \bar{L}_{i-2})$$

The circuit employs two Texas 74 series TTL integrated circuit chips. The first chip SN7474 is a dual D-type flip-flop to store  $L_{i-1}$  and  $L_{i-2}$ . The second chip SN7410 is a triple 3-input NAND gate to implement the exclusive OR function described by the equation given above.

## (2) The Adaptive Accumulator AA

The AA can be implemented either in the form of Fig. 5.2 or in the form of Fig. 5.7 (using the counting approach). The two forms give exactly the same result, i.e. given  $Z$  they produce  $H_{Di}$ . The circuit synthesis for the hardware implementation was chosen to be based on the form of Fig. 5.2.

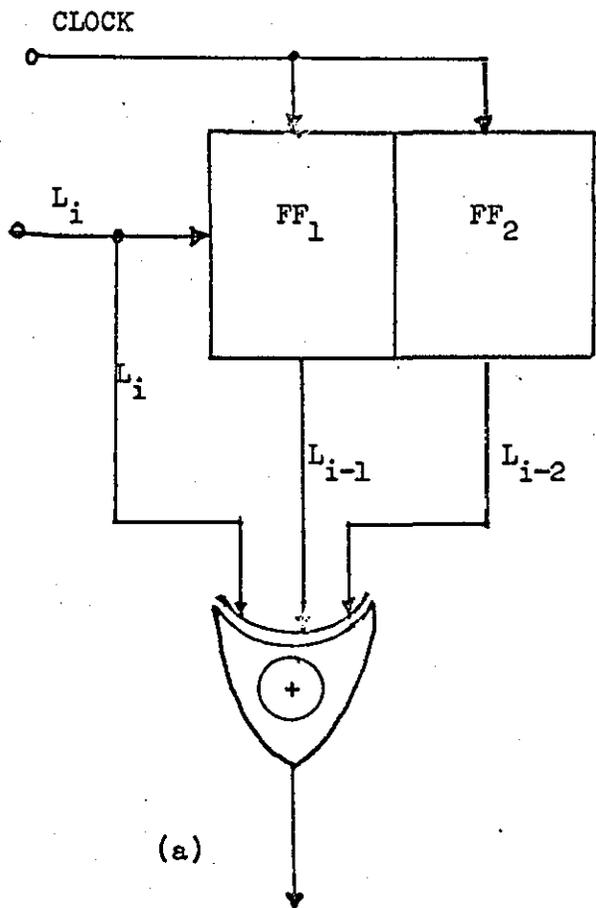
The AA in this case can be divided into three parts: the adder-subtractor ADS, the parallel-in parallel-out shift register PIPO and the selection logic ASL.

### (a) The adder-subtractor ADS

The function of this block is to perform the operation

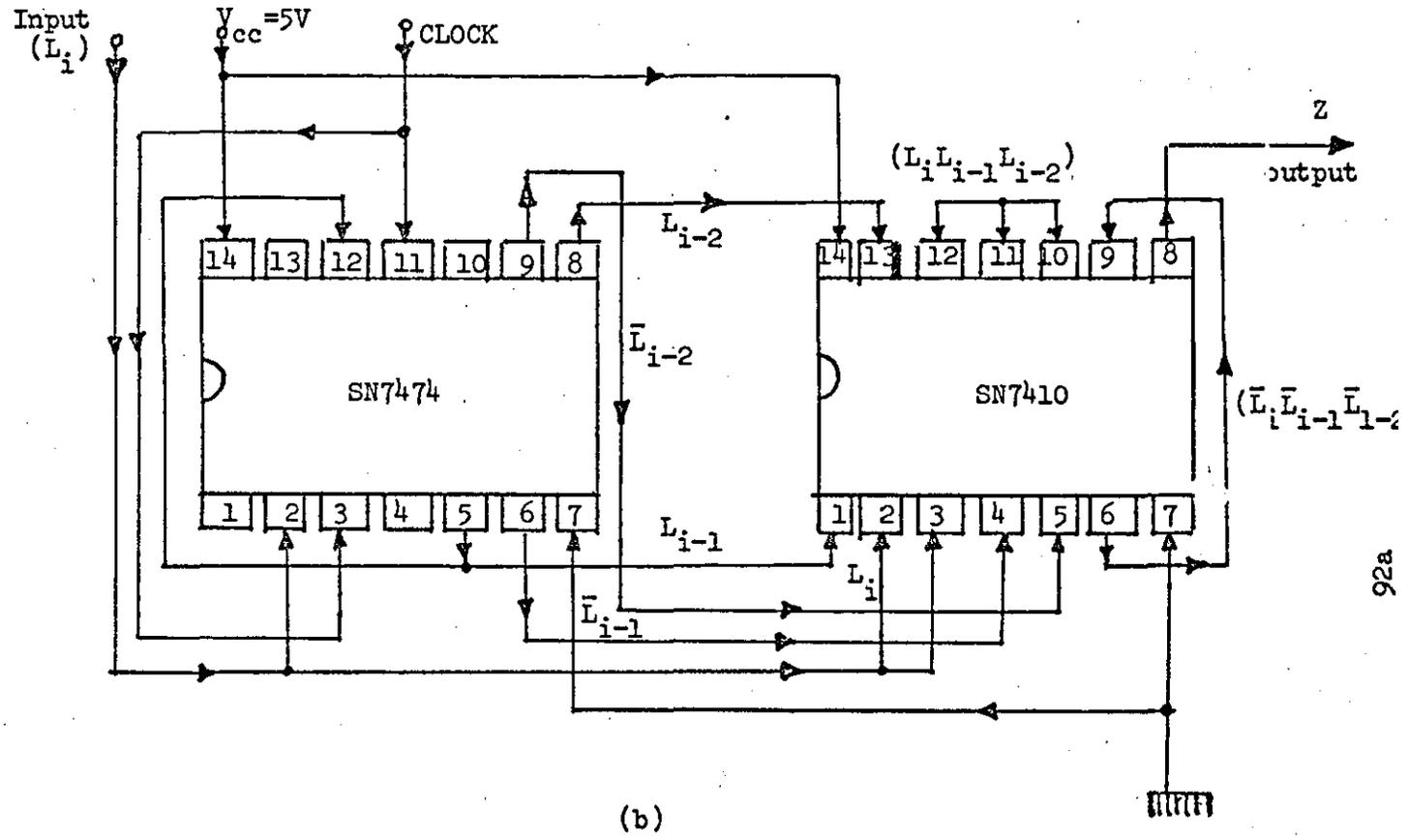
$$\begin{aligned} H_{Di} &= H_{D(i-1)} + \Delta_d(Z, J) & Z = 1 \\ &= H_{D(i-1)} - \Delta_d(Z, J) & Z = 0 \end{aligned}$$

where  $H_{Di}$  and  $H_{D(i-1)}$  are 12-digit wide parallel binary signals defined by equation (5.8).



(a)

$$Z = (L_i \cdot L_{i-1} \cdot L_{i-2}) + (\bar{L}_i \cdot \bar{L}_{i-1} \cdot \bar{L}_{i-2})$$



(b)

Figure 5.16 Generation of the Z control signal

(a) Logical diagram. FF are D flip-flops and  $\oplus$  is an exclusive OR.

(b) Circuit diagram.

SN7474 = Texas Instruments 74 TTL Dual D type Flip-Flop

SN7410 = Texas Instruments 74 TTL Triple 3-Input NAND Gates

$\Delta_d(Z,J)$  is a parallel 4-digit binary word consisting of the binary sequence  $(\Delta_4, \Delta_3, \Delta_2, \Delta_1)$  representing the increment of the step-height in binary notation. In decimal notation  $\Delta_d(Z,J)$  can be written as

$$\Delta_d(Z,J) = \sum_{j=1}^4 \Delta_j 2^{j-1} \quad (5.21)$$

where  $\Delta_1$  is the LSD

$$\Delta_j = \{0,1\} \quad (5.22)$$

The adder-subtractor can be realized by 3 blocks of 4-bit binary adders in conjunction with 3 blocks of 4-bit true-zero complement elements. The arrangement is shown in Fig. 5.17. Subtraction is carried out by the addition of 2's complements. Negative values of the number  $\Delta_d$  are represented by the 2's complement of  $\Delta_d$ , i.e.  $(2^{13} - \Delta_d)$ ; this when added to  $H_D(i-1)$  gives the result  $(H_D(i-1) - \Delta_d)$ , i.e.

$$(2^{12+1} - \Delta_d(Z,J)) + H_D(i-1) = (H_D(i-1) - \Delta_d(Z,J) + 2^{13}).$$

The 2's complement of  $\Delta_d(Z,J)$  is generated by interchanging the ones and zeros in  $\Delta_d(Z,J)$  and adding one to the result. The interchange is carried out by the true-zero complement elements COM1, COM2, COM3 (see Fig. 5.12) and the one is added at the carry input of the adder (comprising ADD1, ADD2, ADD3). The  $2^{13}$  term appears as a carry  $C_4'$  from the last adder stage and indicates that the result of the subtraction is positive. When  $C_4'$  is a logical zero the output of the adder represents a negative quantity and appears in 2's complement form. This cannot happen in our case because  $\Delta_d(Z,J)$  is always less than  $H_D(i-1)$ .

When  $Z = 1$  the carry input to the first stage of the adder is a logical zero and the action of the COM1, COM2, COM3 is inhibited. In this case  $\Delta_d(Z,J)$  is simply passed to the input of the adder where it

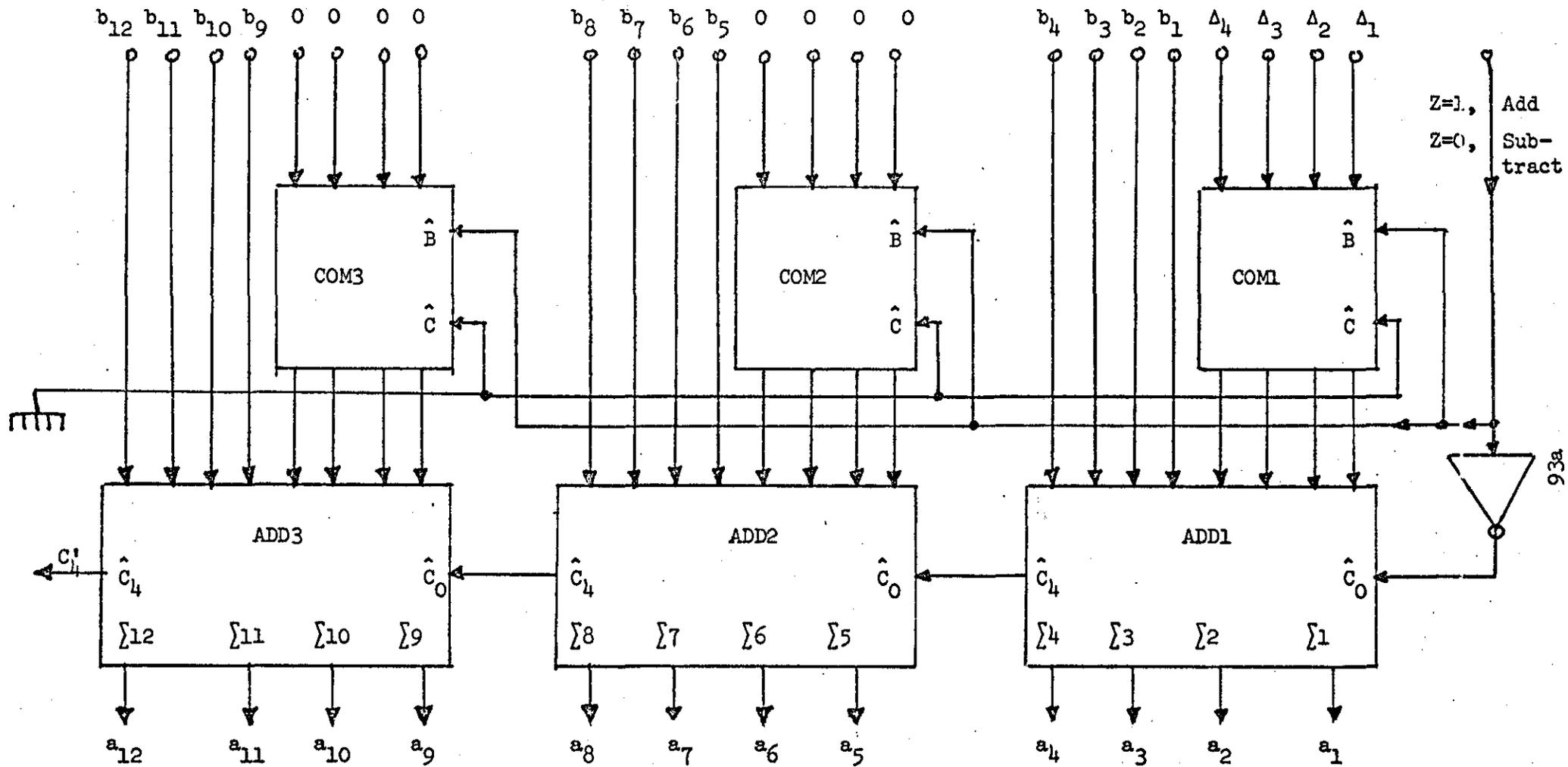


Figure 5.17 Block diagram of the Adder-subtractor

$C_4'$  = 1 when result is Positive  
 = 0 when result is Negative

COM1, COM2, COM3 = True-zero complement elements

ADD1, ADD2, ADD3 = 4-Bit binary adders

$\hat{C}_0$  = carry in,  $\hat{C}_4$  = carry out

is added to  $H_{D(i-1)}$ .

Fig. 5.18 shows how the circuit of the adder/subtractor can be implemented using Texas Instruments integrated circuits, chips SN54283 and SN74H87. The SN54283 are 4-bit binary adders. The sum outputs ( $a_j$ ) provided for each bit and the resultant carry ( $\hat{C}_4$ ) is obtained from the 4<sup>th</sup> bit. These adders feature full internal look-ahead across all four bits generating the carry term in 10 nanosecond. This provides partial look-ahead performance at the economy and reduced package count of the ripple-carry implementation shown in Fig. 5.13. The complete addition operation of two 12-bit binary numbers is carried out in approximately 35 nanosecond.

The SN74H87 are 4-bit true/complement elements. When the input to the control pin (pin 8) is low (logical zero) the 4-bits applied to the input (pins 2, 5, 10, 13) are transferred to the output pins (pins 3, 6, 9, 12) in either complement or true form depending on whether the second control pin (pin 1) is at logical zero or logical one, respectively.

Detailed description of SN7400, SN74H87 and SN54283 is available in the manufacturer's literature<sup>(77,88)</sup>.

(b) The PIPO and segment logic

The function of the PIPO is to act as a temporary memory in which the output of the adder/subtractor at the instance of the previous clock pulse is stored. The output of the PIPO at the  $i^{\text{th}}$  clock pulse is  $H_{D(i-1)}$ . This can be achieved by the arrangement depicted in Fig. 5.19. The circuit diagram shown in Fig. 5.14c employs two Texas chips - SN74198, which is an 8-bit, and SN74194, which is a 4-bit wide parallel-in parallel-out shift register - to form a 12-bit wide PIPO. The clear pins 1 (SN74194) and 13 (SN74198) are taken to the 5 volts

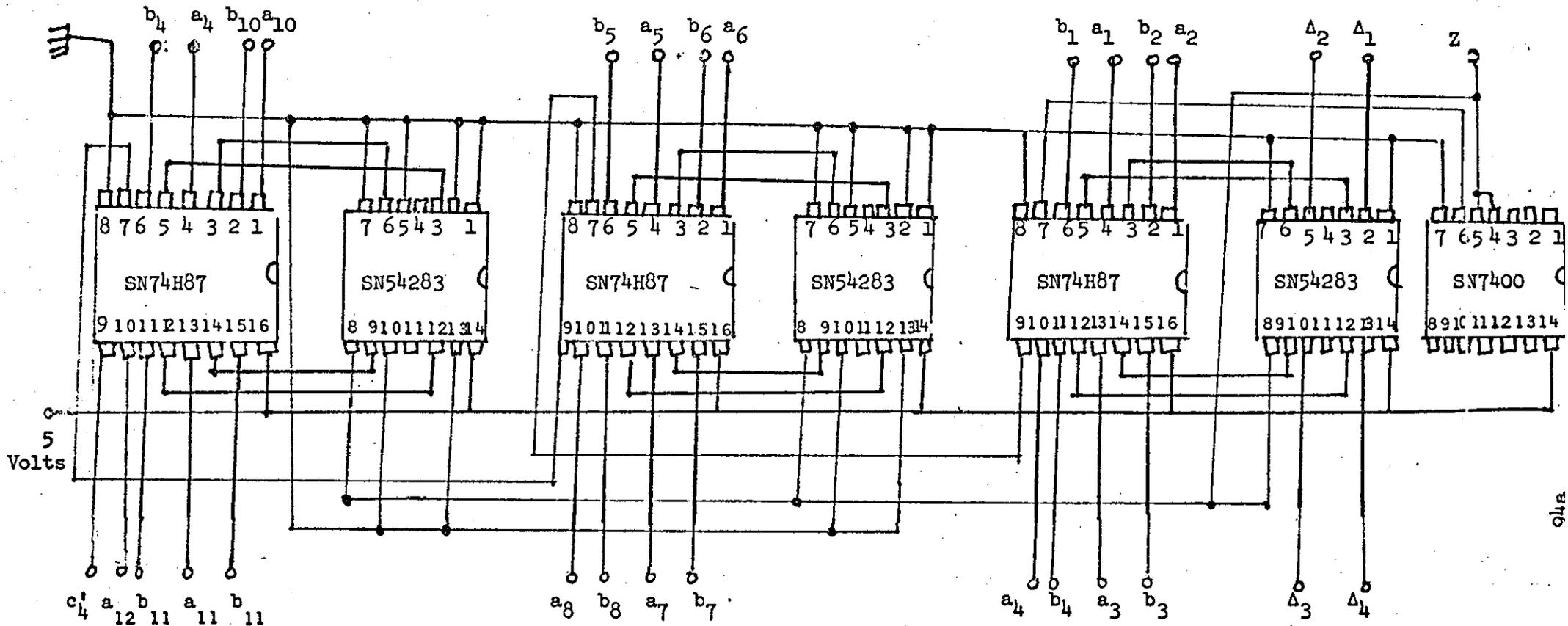


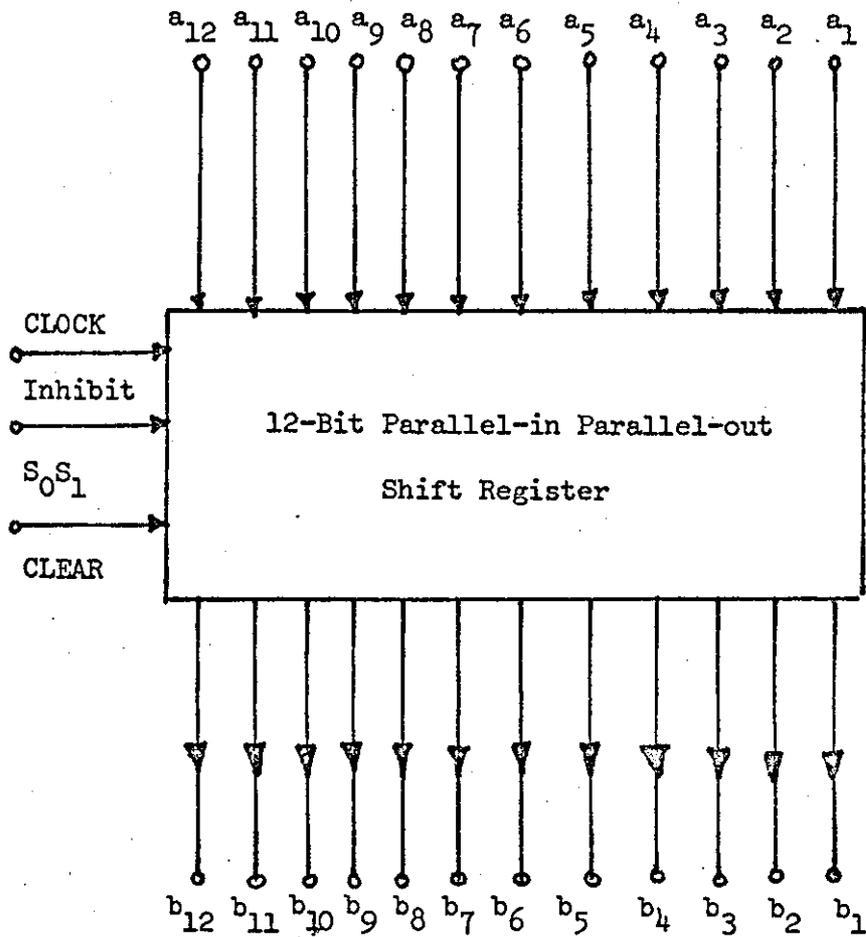
Figure 5.18 Circuit diagram of the Adder/subtractor

SN74487 = Texas 4-bit wide true-zero complement chip

SN54283 = Texas 4-bit binary full adder with carry look-ahead

SN7400 = Texas quad. 2-input Nand gates

The circuit acts as an adder when  $Z = 1$  and acts as a subtractor when  $Z = 0$ .



(a) Functional diagram

Inputs				Outputs
CLEAR	S <sub>0</sub> S <sub>1</sub>	CLOCK	a <sub>12</sub> a <sub>11</sub> ...a <sub>1</sub>	b <sub>12</sub> b <sub>11</sub> ...b <sub>1</sub>
0	X X	X	X X...X	0 0...0
1	1 1	↑	a <sub>12</sub> ...a <sub>1</sub>	a <sub>12</sub> ...a <sub>1</sub>
1	0 0	X	X X...X	(b <sub>12</sub> ...b <sub>1</sub> )

(b) Functional table

X ≡ don't care condition

↑ ≡ Transition from low to high level

S<sub>0</sub>S<sub>1</sub> ≡ Loading control signals

Figure 5.19 The Parallel-in Parallel-out Shift Register



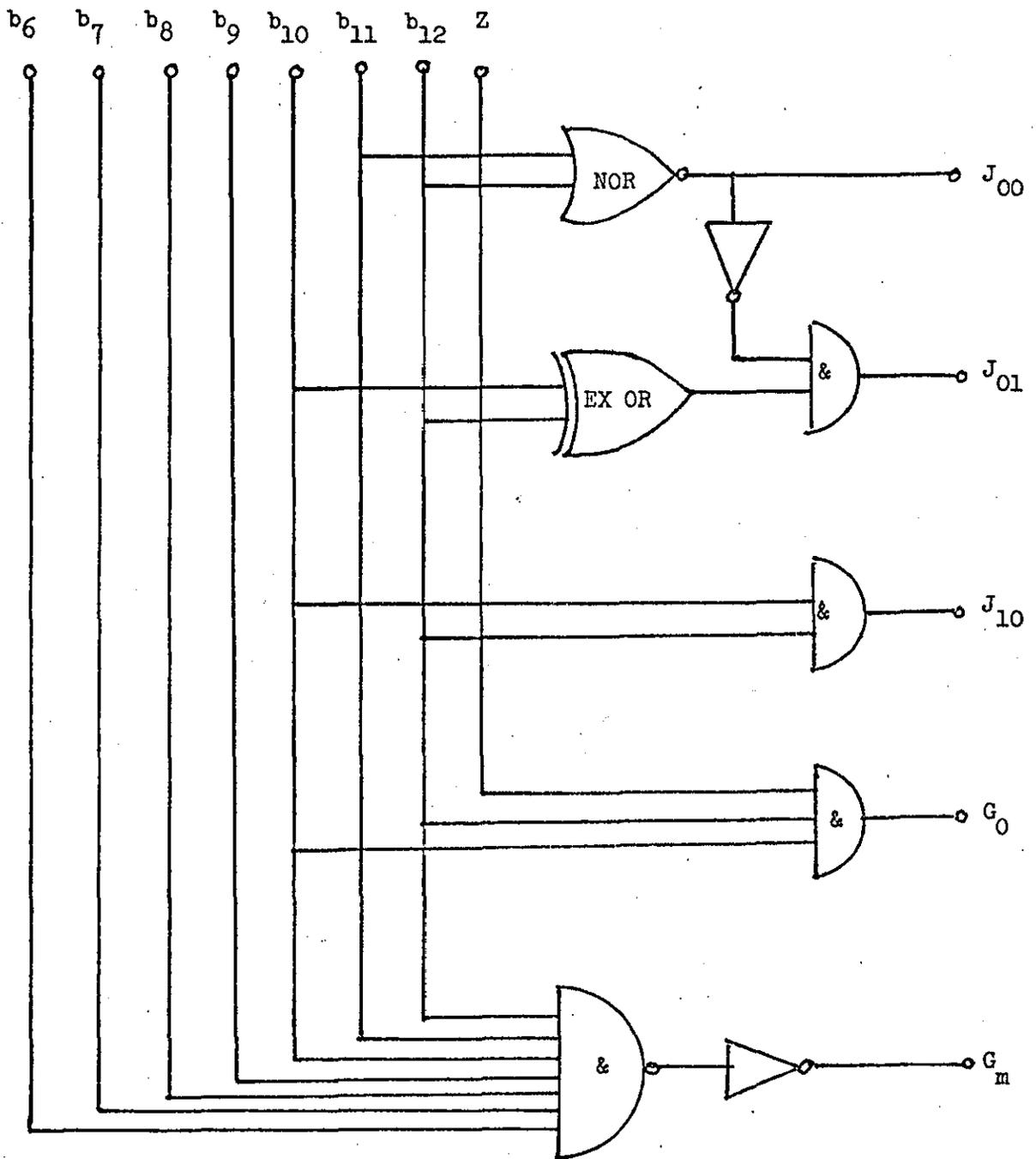
rail and the control pins  $S_0$ ,  $S_1$  (pins 9, 10 for SN74194, pins 1, 23 for SN74198) to ground. This enables the input digits to the PIPO before the positive edge of the clock pulse arrives to be transferred to the PIPO output synchronously with the leading edge of the clock pulse (see Fig. 5.19b).

The segment logic shown in Fig. 5.20 scans the output  $H_{D(i-1)}$  of the PIPO according to Table 5.3 and produces  $J_{00}$  when  $J = 0$ ,  $J_{01}$  when  $J = 1$  and  $J_{10}$  when  $J = 2$ . The segment logic also detects the overload (when  $H_{D(i-1)} \geq 2.56$  volts) and the initial idling (when  $H_{D(i-1)} \leq 32$  mv) condition and generates the indicator signals  $G_0$  and  $G_m$ , respectively.

The circuit diagram of the board which performs the above operations is shown in Fig. 5.20b. It employs Texas Instruments integrated circuits which are economically and commercially available as listed in the figure.

### (c) Selection logic

The function of the selection logic is to select the appropriate value of  $\Delta_d(Z, J)$  according to Table 5.4. This task can be performed by first generating the address signals  $A_R$ ,  $B_R$  and  $J_3^1$  from the outputs  $J_{00}$ ,  $J_{01}$  and  $J_{10}$  of the segment logic. The functional and circuit diagrams of the arrangement which performs this intermediate operation is shown in Fig. 5.21a and 5.21c. The generated address signals  $A_R$ ,  $B_R$  and  $J_3^1$  as functions of  $J_{00}$ ,  $J_{01}$ ,  $J_{10}$  and  $Z$  are defined by the functional Table of Fig. 5.21b. Having generated the address signals  $A_R$ ,  $B_R$ ,  $J_3^1$  and  $G_0$ , the appropriate binary values of  $\Delta_d(Z, J)$  defined by Table 5.4 are accordingly selected using the arrangement shown in Fig. 5.22b. The realization of this block diagram using Texas integrated circuit chips (SN7400 quadruple 2-input NAND gates, SN74153 dual 4-line to 1-line selectors and SN74125 quadruple bus buffer gates) is



(a)  
Functional diagram

$$J_{00} = (b_{12} + b_{11})$$

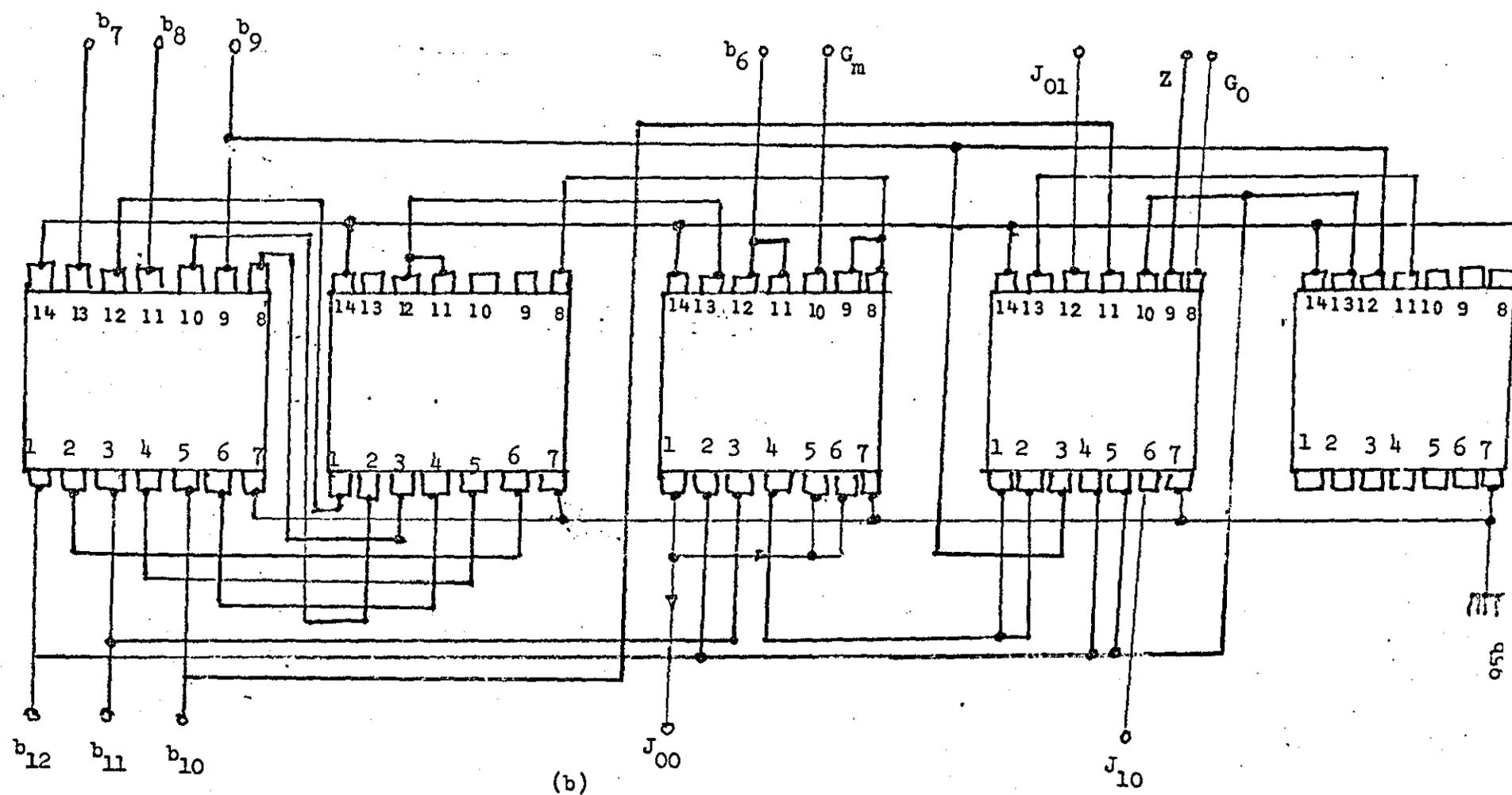
$$J_{01} = (b_{12} + b_{11}) \cdot (b_{12}\bar{b}_4 + \bar{b}_{12}b_4)$$

$$J_{10} = b_4 b_{12}$$

$$G_0 = (b_{12} b_{10}) Z$$

$$G_m = \bar{b}_{12} \bar{b}_{11} \bar{b}_{10} \bar{b}_4 \bar{b}_5 \bar{b}_7 \bar{b}_6$$

Figure 5.20 The segment logic



(b) Circuit diagram using Texas TTL integrated circuits

- |   |                                   |
|---|-----------------------------------|
| SN7404 = Hexinverter                      | SN7430 = 8-input NAND gate        |
| SN7402 = Quad. 2-input positive NOR gates | SN7411 = Triple 3-input AND gates |
|   | SN7436 = Exclusive OR gate        |

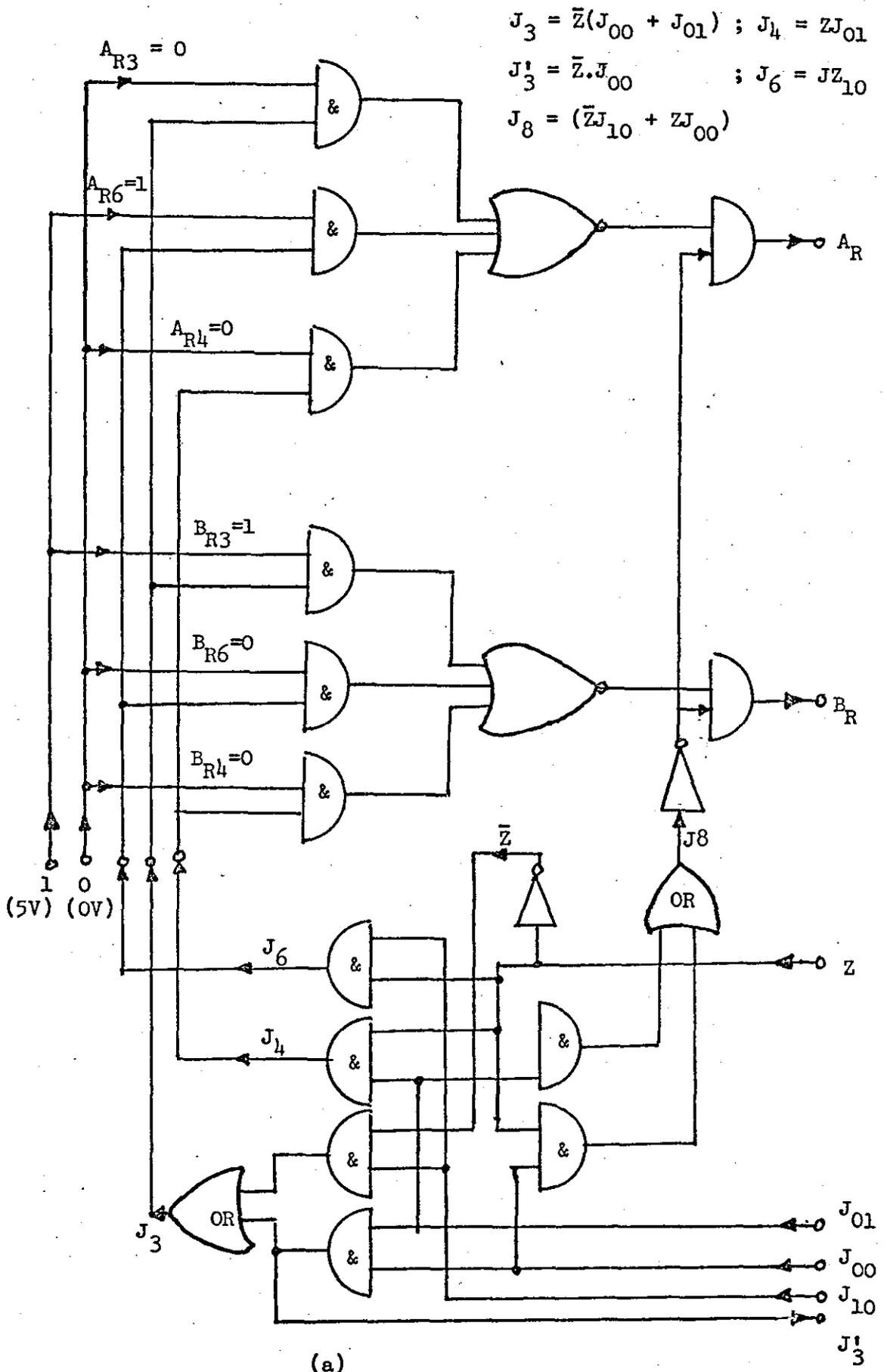


Figure 5.21(a) Generation of the  $\Delta$ -address signals  
Functional diagram

Note: Functional table Figure 5.21(b) is on page

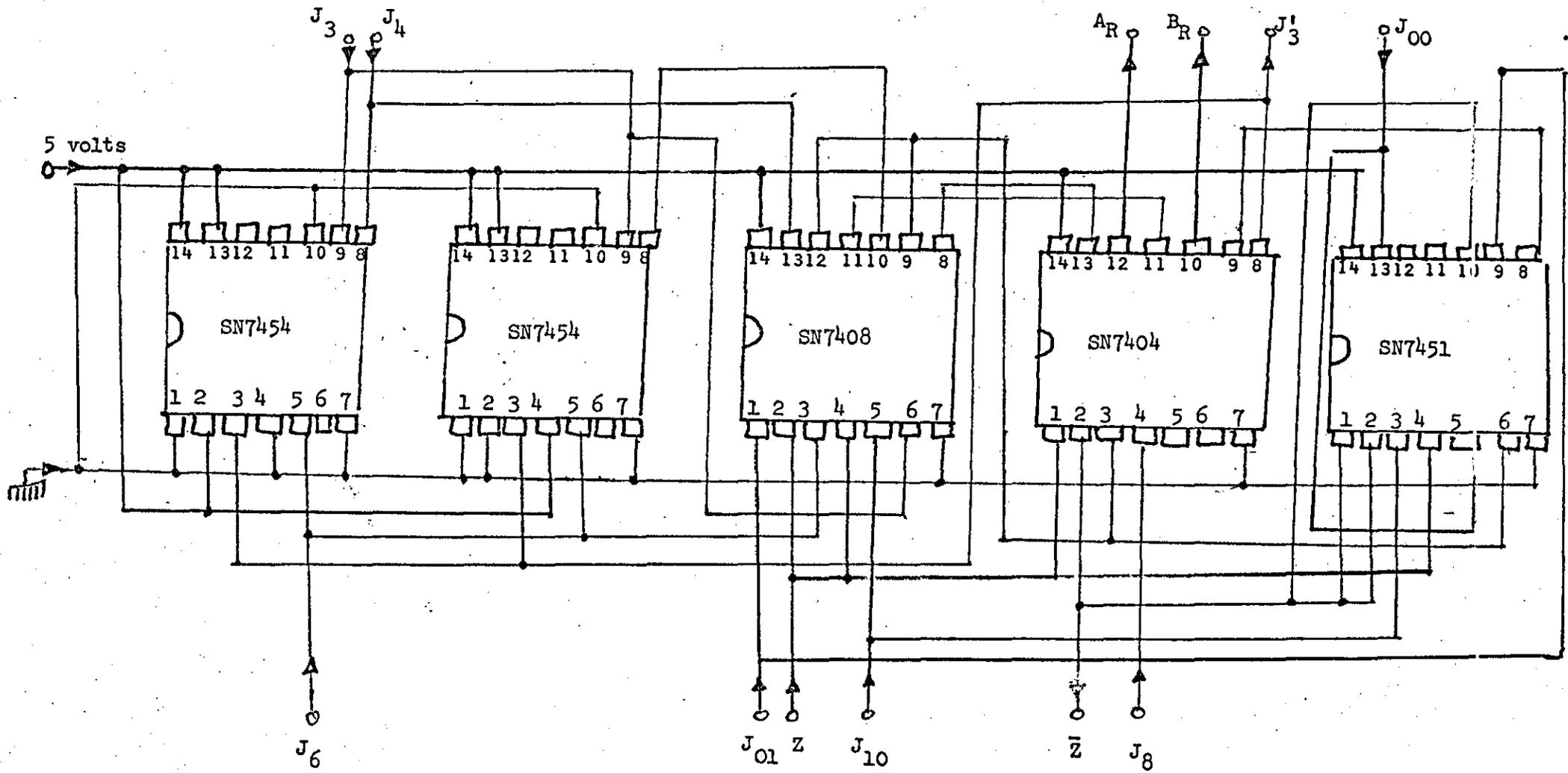


Figure 5.21(c) Circuit diagram of the A-address signal generator

SN7454 = 4-wide AND-OR-INVERT gates

SN8408 = Quad. 2-input positive AND gates

SN7451 = Dual 2-wide 2-input AND-OR-INVERT gates

SN7404 = Hex inverters

shown in Fig. 5.22c. The relationship between the address/control signals ( $A_R$ ,  $B_R$ ,  $J'_3$  and  $G_0$ ) and  $\Delta_d(Z, J)$  in binary and decimal notation is defined by Table 5.22a. It can be noticed from this table that when  $G_0 = 1$  (indicating that overloading has been detected) the generated increment  $\Delta_d(G_0 = 1)$  is zero and the reconstructed signal remains constant.

The final output of the digital adaptive accumulator which is to be applied to the D/A converter is passed in parallel from the ADS to another set of 4-bit to 1-line selectors, as shown in Fig. 5.23. These selectors are controlled by  $G_m$  (the initial idling control signal). As long as  $G_m = 0$ ,  $H_{Di}$  at the output of the ADS is transferred unaltered to the input of the D/A converter. When  $G_m = 1$  the output of the selectors is a constant binary signal  $S_0$  defined by equation 5.10 (i.e. the minimum step-height of 31 mv). The actual circuit diagram is given in Fig. 5.23b. It employs 3 blocks of quadruple 2-line-to-1-line selectors which accept the output of the ADS and a constant signal  $S_0$  and passes one of these signals to the output according to the functional table of Fig. 5.23a. In other words, the selectors pass  $a_1 \dots a_{12}$  to the out without alteration when  $G_m = 0$  but replace  $a_j$  by  $d_j$  ( $j = 1, 12$ ) when  $G_m = 1$ .

The output  $H_{Di}$  of the selectors is then passed to the D/A converter for analogue processing or to a digital processing circuit, as required.

Control signals				Increment $\Delta_d(Z,J)$				
$G_0$	$J'_3$	$B_R$	$A_R$	$\Delta_3$	$\Delta_2$	$\Delta_1$	$\Delta_0$	decimal value of $\Delta_d$
0	0	0	0	0	1	0	0	4
0	0	0	1	0	1	1	0	6
0	0	1	0	0	0	1	1	3
0	0	1	1	1	0	0	0	8
0	1	1	0	0	0	0	1	1
1	X	X	X	0	0	0	0	0

Figure 5.22(a) Truth table of the selection of  $\Delta_d(Z,J)$

Z	$J_{mn}$ m n	$J'_3$	$B_R$	$A_R$
0	0 0	1	1	0
0	0 1	0	1	0
0	1 0	0	1	1
1	0 0	0	1	1
1	0 1	0	0	1
1	1 0	0	0	0

Figure 5.21(b) Functional table of the generation of the  $\Delta$ -address signals

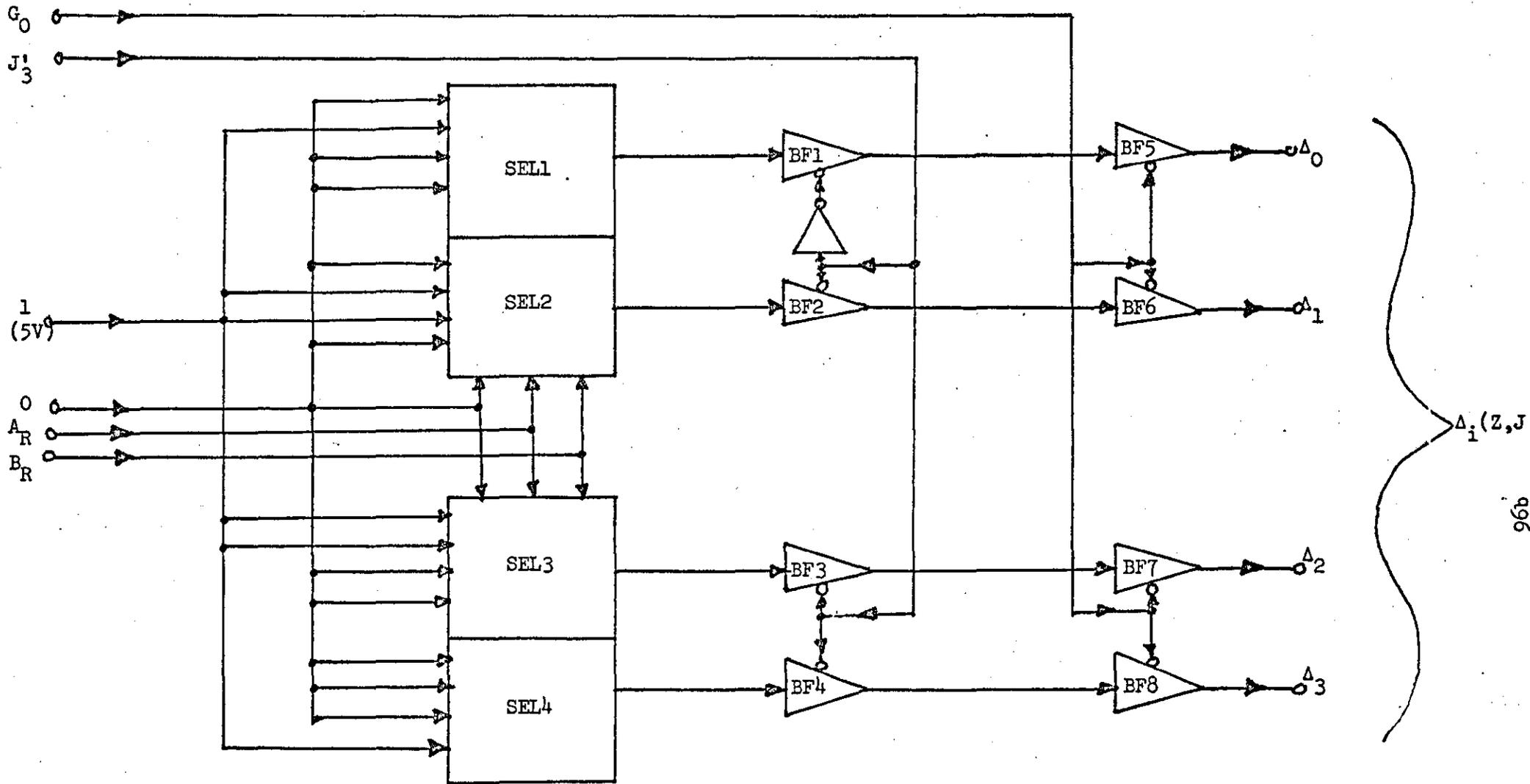


Figure 5.22(b) Functional diagram of the selection of  $\Delta_i(Z, J)$

SEL1,....SEL4 = 4-line to one line data selectors

BF1,.....BF8 = Bus Buffer gates with 3-state outputs. Output is disabled when  $J_3^1$  is high.

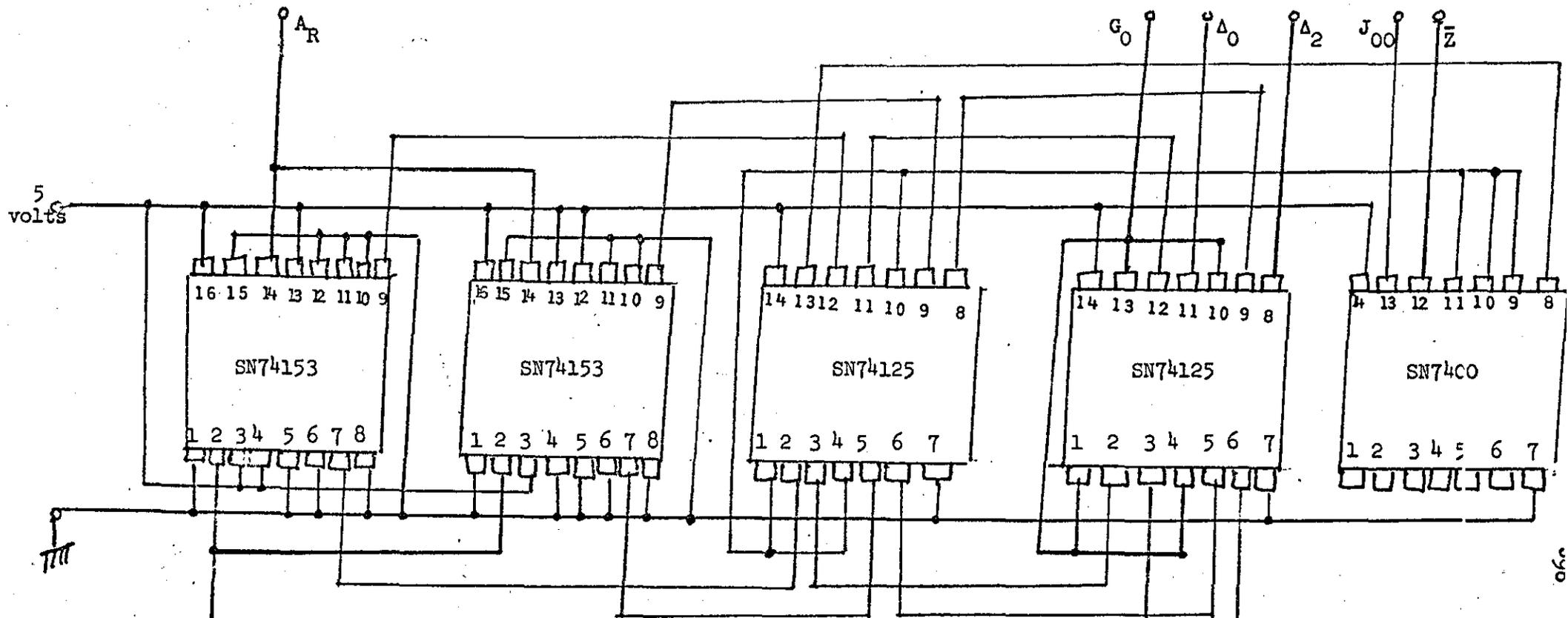


Figure 5.22(c) Circuit diagram of the selection of  $\Delta_d(Z, J)$

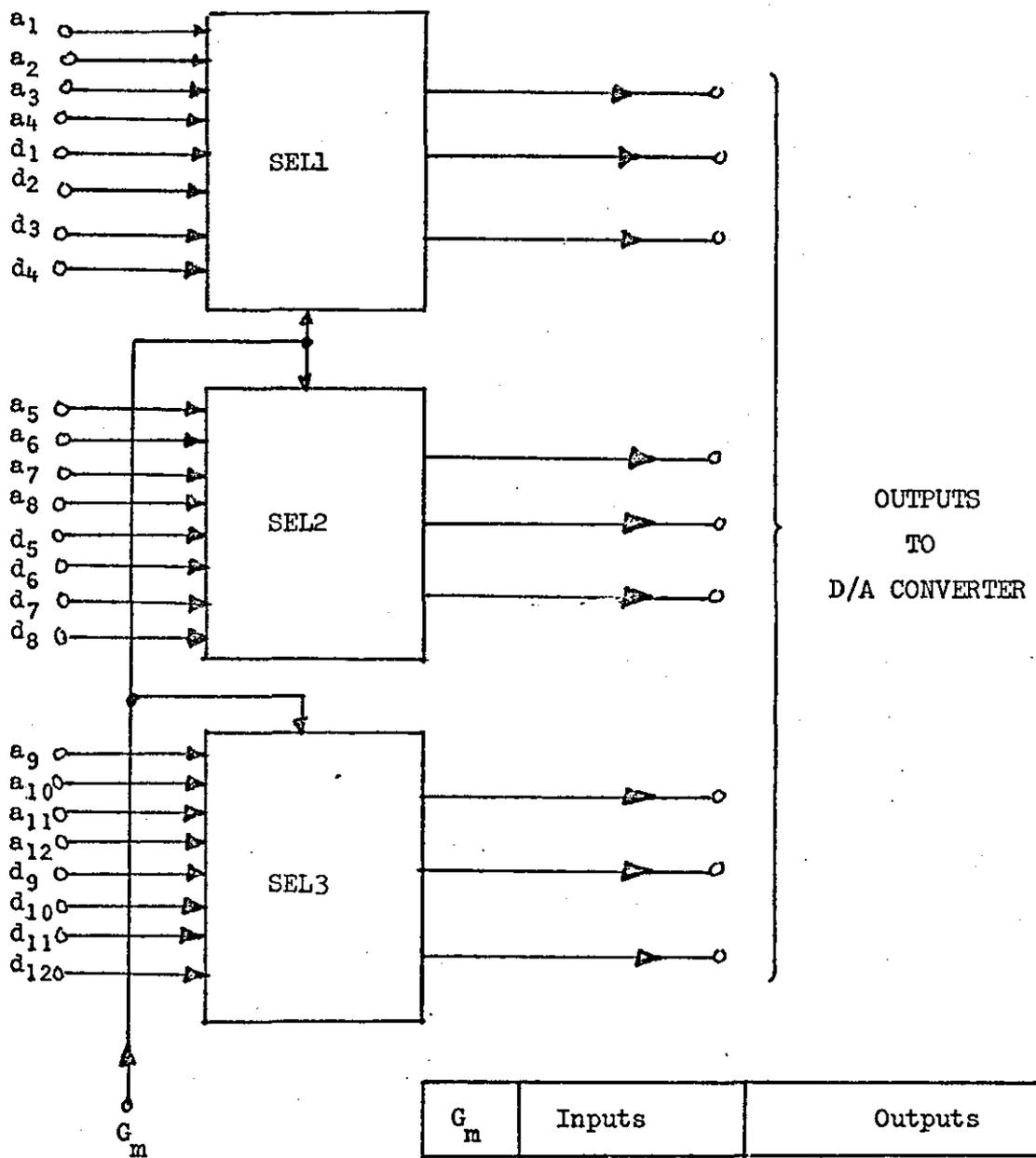
SN74153 = Texas Dual 4-line to 1-line data selector/multiplexer

SN74125 = Texas Quadruple Bus Buffer gates

SN7400 = Quadruple 2-input NAND gates (to generate  $J_3'$  from  $\bar{Z}$  and  $J_{00}$ )

$A_R$ ,  $B_R$ ,  $\bar{Z}$ ,  $J_{00}$  and  $G_0$  are inputs

$\Delta_0$ ,  $\Delta_1$ ,  $\Delta_2$ ,  $\Delta_3$  are outputs



(a)

$G_m$	Inputs	Outputs
0	$S_0$ $H_{Di}$	$H_{Di} = a_{12} \dots a_1$
1	$S_0$ $H_{Di}$	$S_0 = d_{12} \dots d_1$

Functional table

Figure 5.23 Selection of the final output from the Accumulator

(a)  $a_1, \dots, a_{12}$  are defined by equation (5.8)

$d_1, \dots, d_{12}$  " " " " (5.10)

SEL1, ..., SEL3 = Quadruple 2-line to 1-line data selectors

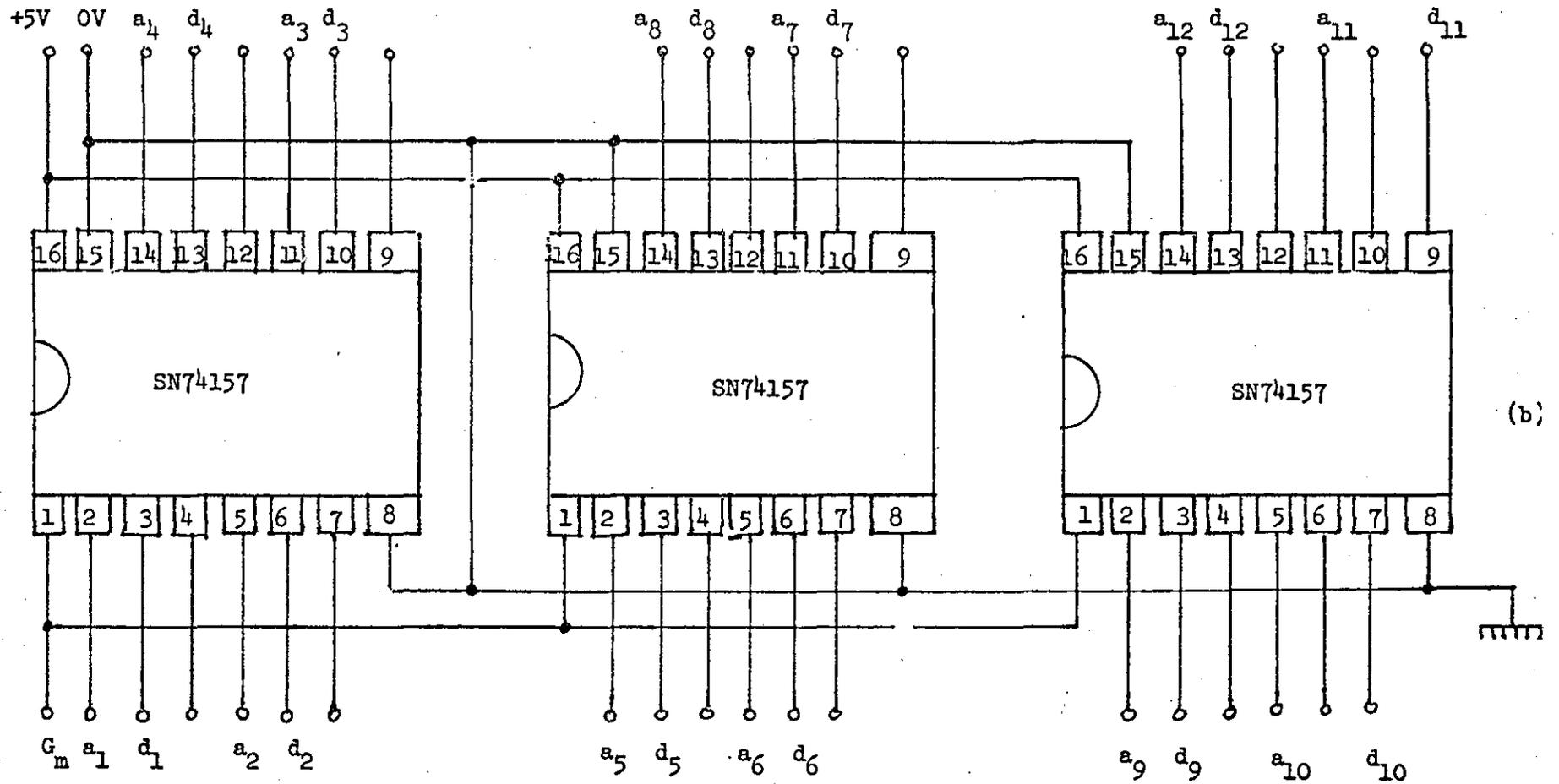


Figure 5.23 Final selection of  $H_{Di}$

(b) Circuit diagram using Texas SN74157

SN74157 = 2-line to 1-line data selectors/multiplexers

## CHAPTER VI

### SCALE - TO - A-LAW PCM CONVERSION

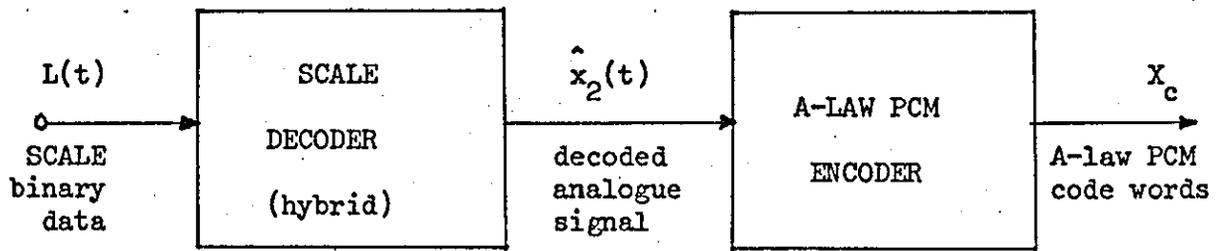
There are many situations when it is desirable to convert the binary data originating from a SCALE encoder into PCM words and vice versa. Such situations have been mentioned in Chapter I and will be discussed in more detail in Chapter VIII.

The purpose of this chapter is the design of a converter which accepts the SCALE binary data stream  $L(t)$  and produces A-law PCM words. When such an A-law PCM code word  $X_c$  is applied to an ordinary A-law PCM decoder, the filtered output signal  $x_2'(t)$  must be a close estimate of  $x_2(t)$  - the input signal to the SCALE encoder.

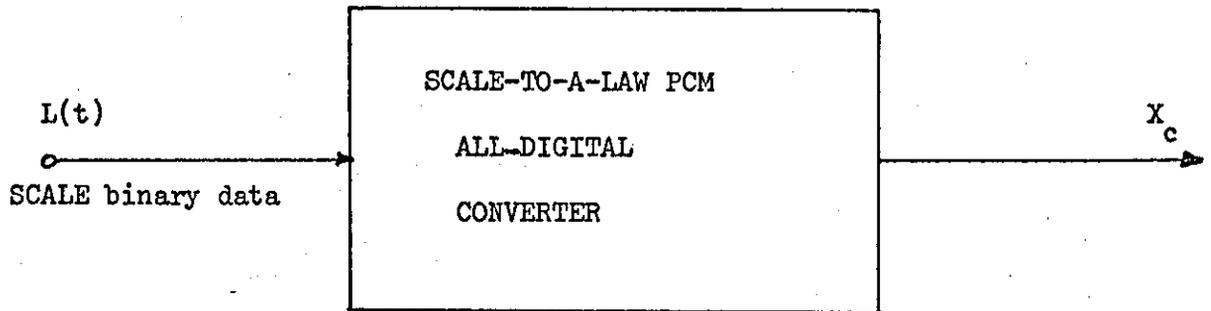
The most obvious approach to this conversion problem is that shown in Fig. 6.1a. The SCALE binary signal  $L(t)$  is first converted to analogue waveform  $\hat{x}_2(t)$  by a SCALE decoder and subsequently encoded into A-law PCM by an A-law PCM encoder.

This approach does not only lead to a costly hardware realization but extra noise is added to the final decoded signal due to the unavoidable noise associated with the analogue circuit; and due to the digital-analogue-digital operations themselves. To avoid these complications it is necessary to perform the conversion completely in the digital domain, as shown in Fig. 6.1b.

During the last seven years several attempts have been made to use the linear Deltamodulation system as a part of linear PCM<sup>(78)</sup>, A-law PCM<sup>(79)</sup> and DPCM<sup>(80)</sup> encoders and decoders. The practical applications of the results of these experimental and theoretical studies are, to date, rare. This is a direct consequence of the constant step-height nature of the linear Deltamodulator, which resulted in converters with



(a)



(b)

Figure 6.1 Methods of SCALE-to-A-law PCM conversion

- (a) Digital-Analogue-Digital approach
- (b) All digital approach

prohibitively high sampling rates and/or complex realization hardware (80,81).

No attempt has been made before to use SCALE as a part of any other digital system or to convert its binary signal to any other digital form. This is in spite of SCALE superiority to the linear Deltamodulation, from the point of view of intelligibility at reduced bit rate. The reasons for this were its slowly varying feedback signal (at syllabic rates) and the dependence of this signal on the remote history of the channel patterns. These difficulties can no longer impede the conversion of SCALE binary signal to any other digital format. This is a direct consequence of the results of the previous chapter which enabled the analogue signal  $|x_2(t)|$  at the input of the SCALE encoder to be represented by binary digital words  $H_{Di}$  which is derived from the SCALE binary data stream.

In the light of these results the all-digital SCALE-to-A-law-PCM converter of Fig. 6.1b can be synthesised by partitioning it in the manner depicted in Fig. 6.2. The details of the subunits of the converter are given below.

### 6.1 THE DSCALE WORD GENERATOR

This subunit accepts the SCALE binary data stream  $L_i$  and produces 12-digit binary words representing the estimate of the analogue signal  $|x_2(t)|$  presented to SCALE encoder at the  $i^{\text{th}}$  clock pulse instance. The digital word  $H_{Di}$  is delivered to the Digital Interpolator DINT at a rate of  $f_p$  words/sec.

The design of a digital circuit which can perform the task of the DSCALE word generator has already been dealt with in detail in Chapter V. It is simply a Digital SCALE decoder. It is worth mentioning again that the actual hardware structure of the DWG is a function of the clock

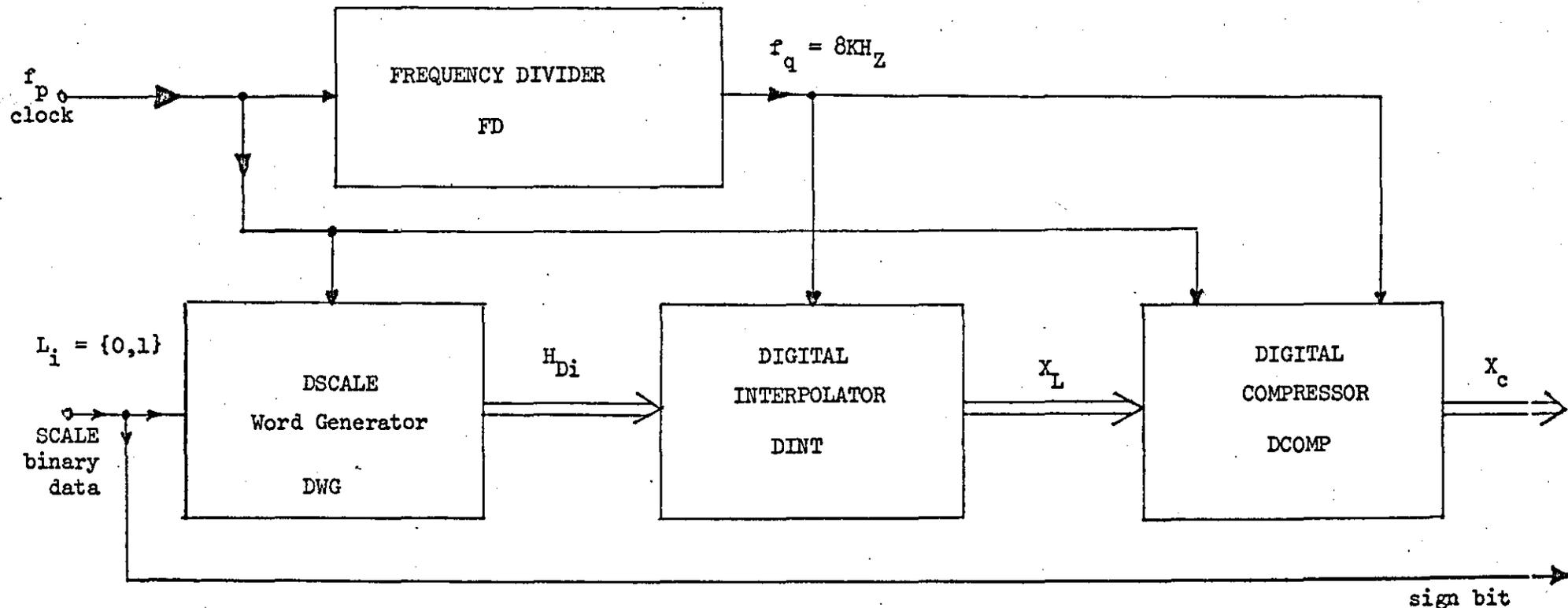


Figure 6.2 An All-Digital SCALE-to-PCM Converter

- $L_i$  = SCALE binary data element seen at the  $i^{\text{th}}$  clock pulse instance
- $H_{Di}$  = 12 parallel binary word representing the analogue input sample to the SCALE encoder (not shown)
- $X_L$  = 12 parallel digits binary word representing the equivalent linear PCM PAM sample produced at Nyquist rate
- $X_c$  = 8-bit A-law PCM word representing  $x_2(t)$  and produced at Nyquist rate
- $f_p$  = SCALE and A-law signal bit rate
- $f_q$  = 1 times Nyquist rate ( $2 \times f_s = 2 \times 4000\text{Hz}$ )

frequency  $f_p$  and the physical design parameters of the hybrid SCALE encoder from which  $L_i$  is originating. The design example considered in Chapter V used a clock frequency  $f_p = 64$  Kb/s. This frequency is compatible with that used in A-law PCM and therefore convenient for the conversion processor.

## 6.2 THE FREQUENCY DIVIDER AND DIGITAL INTERPOLATOR

The sampling theorem states that if a bandlimited signal is sampled at regular intervals and at a rate  $f_q$  which is at least twice the highest frequency in the band, then the samples contain all the information of the original signal. According to this theorem it is not necessary to consider every sample of  $|x_2(t)|$  represented by  $H_{Di}$ . This is because  $x_2(t)$  has an upper frequency limit of 2.4 - 4.0 KHz while the PAM samples represented by  $H_{Di}$  are being read at the output of the DWG at an over-sampled rate of 64 Kb/s.

The function of the digital interpolator DINT is to observe  $N_r$  ( $= f_p/f_q$ ) adjacent SCALE words  $H_{Di}$  and from these words generate linear PCM words  $X_c$  which are delivered to the Digital compressor DCOMP at a rate  $f_q = f_p/N_r$ . For  $f_q = 8$  KHz and  $f_p = 64$  Kb/s the number  $N_r$  of SCALE words to be observed is 8 words.

The process of generating one linear PCM word from  $N_r$  adjacent SCALE words can be considered as  <sup>$a_n$</sup>  estimation process whereby the PCM word is obtained by digitally interpolating between  $N_r$  SCALE samples. Alternatively the process can be considered as a digital filtering operation whereby undesirable SCALE noise is removed from  $H_{Di}$  leaving the filtered signal to be read into the DCOMP at Nyquist rate.

One way of estimating the value of the linear PCM sample is to average the  $N_r$  adjacent  $H_{Di}$  samples and equate the PCM word to this

average. This operation can be mathematically represented by:

$$X_{L(i+N_r-1)} = \frac{1}{N_r} \cdot \sum_{k=0}^{N_r-1} H_D(i+k) \quad (6.1)$$

This equation can be realized by a resettable accumulator followed by a digital divider, as shown in Fig. 6.3a. The intermediate sum  $\hat{y}_i$ , which is a 16-bit binary word, is loaded into the output buffer every  $N_r T_p$  seconds by the timing signal  $f_q$  which also clears the delay store, and hence sets the previous intermediate sum  $\hat{y}_{(i-1)}$  to zero every  $N_r T_p$  seconds. The division by  $N_r$  can simply be achieved by shifting the content of the output buffer by  $2^{(\log_2 N_r)}$  places to the right every  $N_r T_p$  seconds. Because of the slowly varying nature of the value of the SCALE word,  $X_{L(i+N_r-1)}$  is a good estimate of  $\hat{x}_2(t)$  sampled at  $f_q$  KHz.

The effect of the resettable accumulator and divider of Figure 6.3a. in the frequency domain can be evaluated by redrawing the equivalent circuit of the accumulator-divider as shown in Fig. 6.3b. Again the present and the previous  $(N_r-1)$  SCALE words are summed and then loaded into the output buffer before being averaged over the interpolation (estimation) time period  $N_r T_p$  seconds. The delay elements are also reset to zero every  $N_r T_p$  seconds. The PCM words are read out at a rate determined by the appropriate Nyquist rate (8 KHz). Examination of Fig. 6.3b does not only reveal that it performs exactly the same function as that of Fig. 6.3a - and hence satisfies equation (6.1) - but also confirms the fact that the SCALE-to-Linear-PCM conversion process can be viewed as a digital filtering operation. The system depicted in Fig. 6.3b is clearly a transversal digital filter with uniform coefficients (all coefficients are unity) followed by a d.c. gain adjustment circuit (the divider  $1/N_r$ ).

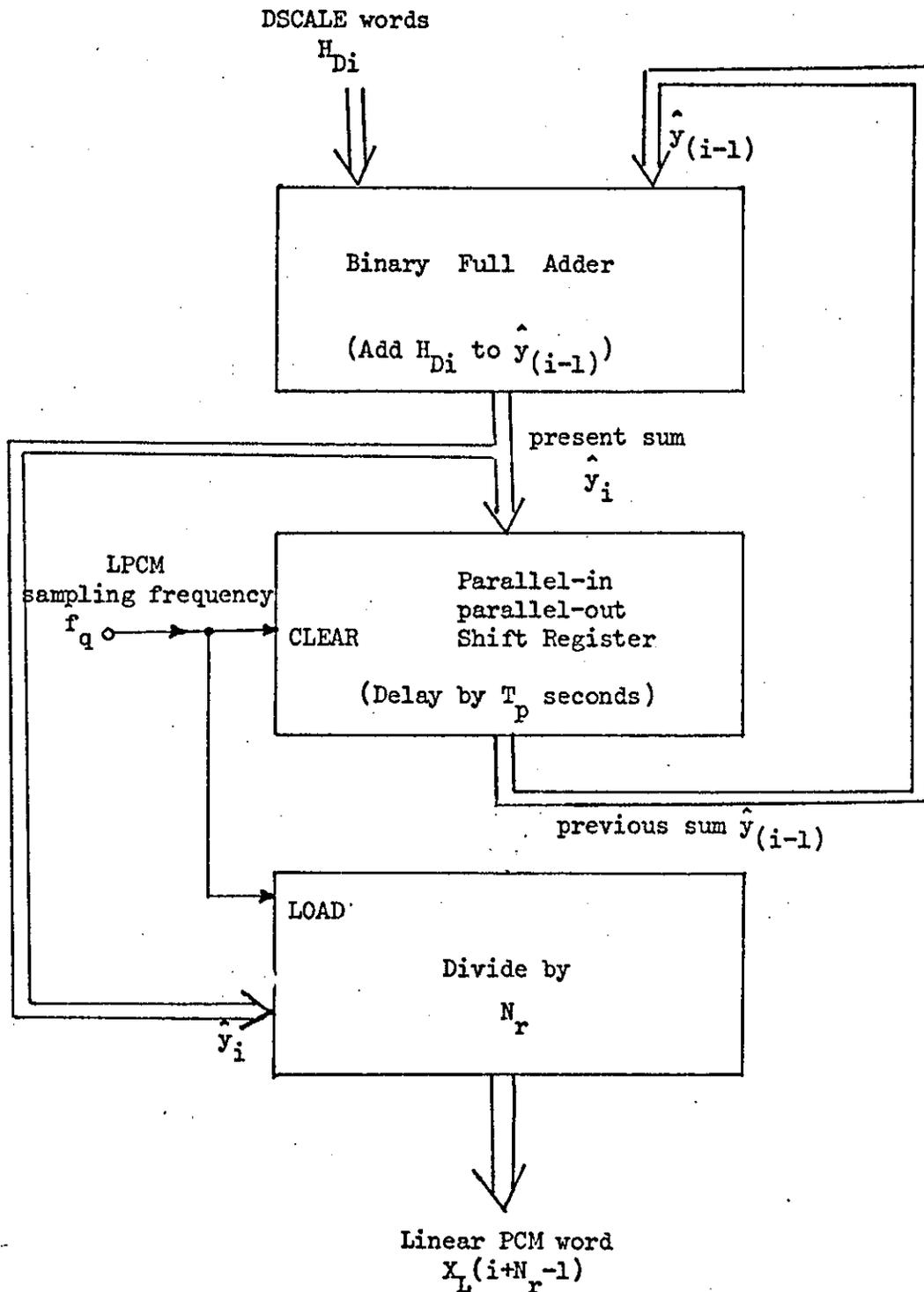


Figure 6.3a Realization block diagram of equation (6.1) using a digital accumulator and a divider

$$T_p = 1/f_p \quad (f_p = \text{SCALE bit rate})$$

$$N_r = \text{Averaging period} = f_p/f_q$$

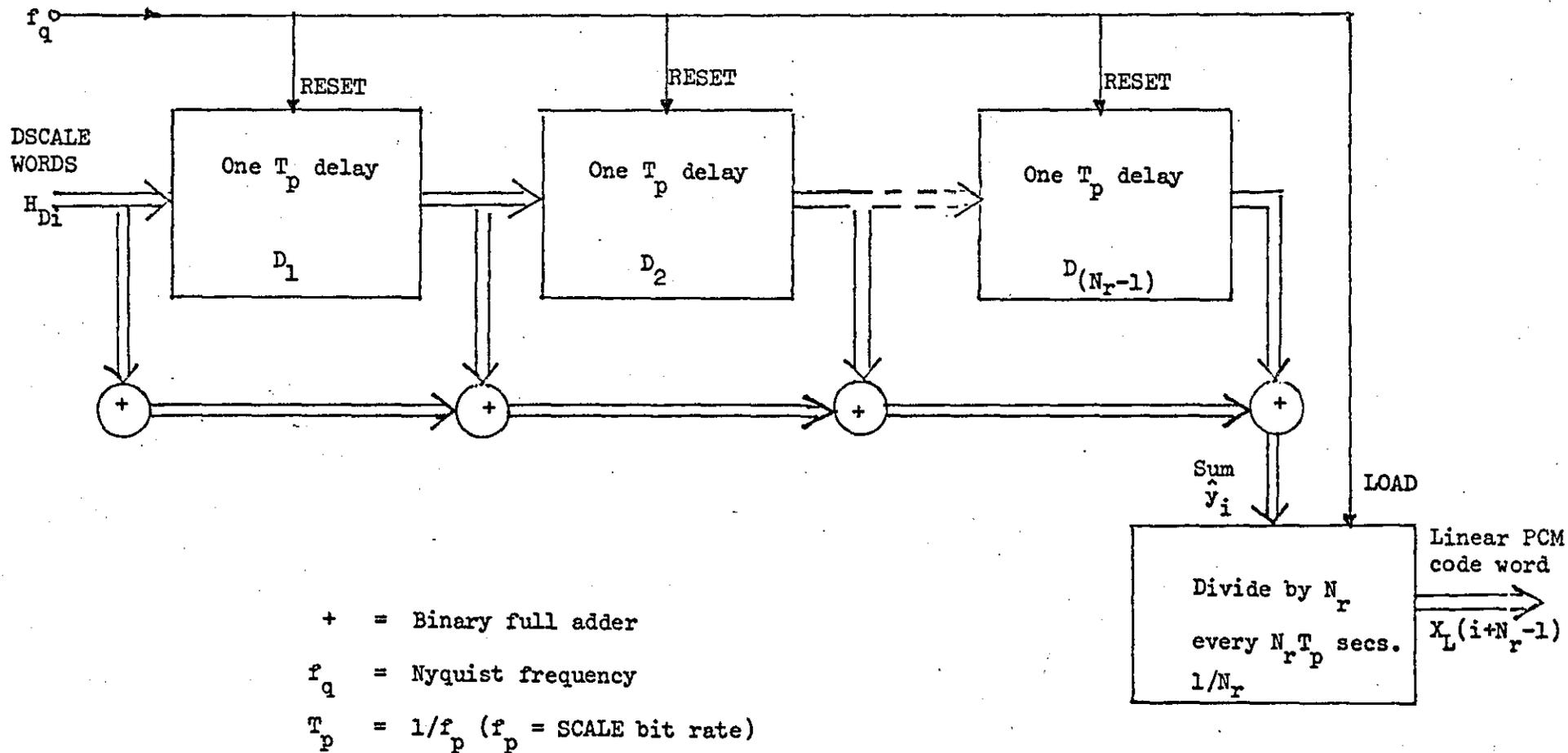


Figure 6.3b An alternative block diagram of the realization of equation 6.1 (SCALE-to-LPCM converter)

To write down the  $z$ -transform of the resettable accumulator and averaging circuit of Figure 6.3 the system block diagram is reduced to that shown in Figure 6.4a.

Another way of looking at the representation of Fig. 6.3a is to remove the resetting switch in Fig. 6.4a and subtract from the output sum of the accumulator, every  $N_r T_p$  seconds, the value of the sum  $N_r T_p$  seconds ago; then divide the result by  $N_r$  as shown in Fig. 6.4b.

Hence from Fig. 6.4b the  $z$ -transform of the resettable accumulator and averaging circuit of Fig. 6.3a can be written as:

$$\phi(z) = \frac{1}{N_r} \left( \frac{1}{1 - z^{-1}} - \frac{z^{-N_r}}{1 - z^{-1}} \right) \quad (6.2)$$

$$= \frac{1}{N_r} \left( \frac{1 - z^{-N_r}}{1 - z^{-1}} \right) \quad (6.3)$$

But  $z^{-1} = e^{-sT_p}$  and the Laplacian variable  $s = j\omega$ ; hence

$$\phi(e^{j\omega T_p}) = \frac{1}{N_r} \left( \frac{1 - e^{-j\omega N_r T_p}}{1 - e^{-j\omega T_p}} \right) \quad (6.4)$$

Remembering that

$$e^{j\omega} = \cos\omega + j\sin\omega \quad ,$$

$$1 = \cos^2\omega + \sin^2\omega \quad \text{and}$$

$$\sin^2\omega = \frac{1}{2}(1 - \cos 2\omega)$$

it follows that

$$\phi(j\omega) = \frac{1}{N_r} \left( \frac{1 - \cos(\omega N_r T_p) + j\sin(\omega N_r T_p)}{1 - \cos(\omega T_p) + j\sin(\omega T_p)} \right)$$

Conjugating both sides:

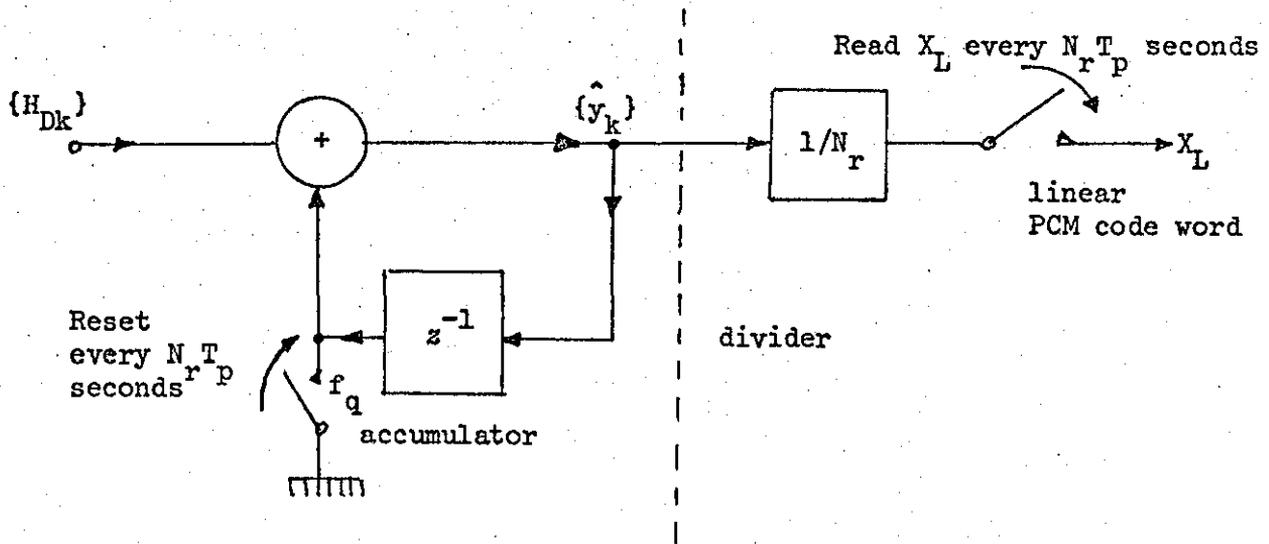


Figure 6.4a Equivalent of the resettable accumulator and divider of Figure 6.3a.

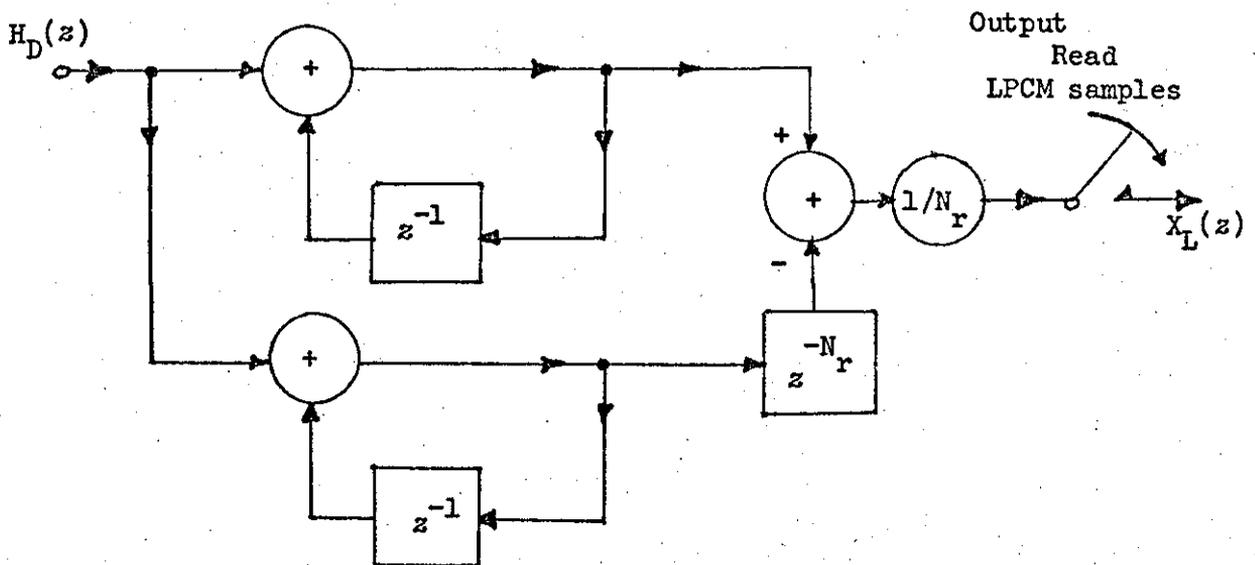


Figure 6.4b Equivalent of Figure 6.3a which enables simpler mathematical representation of the transfer function of the system to be calculated.

$$z = e^{-sT_p} \text{ where } s \text{ is the Laplacian variable.}$$

$$s = j\omega \text{ for the frequency response calculations.}$$

$$\begin{aligned}
\phi(j\omega) \cdot \phi^*(j\omega) &= |\phi(\omega)|^2 \\
&= (1/N_r)^2 \left\{ \frac{1 - \cos(N_r T_p \omega)}{1 - \cos(T_p \omega)} \right\} \\
&= (1/N_r)^2 \left\{ \frac{\sin(\frac{1}{2} N_r \omega T_p)}{\sin(\frac{1}{2} \omega T_p)} \right\}^2
\end{aligned} \tag{6.5}$$

Replacing  $T_p$  by  $1/f_p$  and  $\omega$  by  $2\pi f_s$  the filter frequency response can be written as

$$|\phi(f)| = 1/N_r \left( \frac{\sin(N_r \pi f_s / f_p)}{\sin(\pi f_s / f_p)} \right) \tag{6.6}$$

where  $f_p$  and  $f_s$  are respectively the SCALE clock and the SCALE signal frequencies.

Examination of equation (6.6) reveals that:

as  $f_s$  tends to 0,  $|\phi(f)|$  tends to unity, and

as  $f_s$  increases  $|\phi(f)|$  falls in the manner depicted in Fig. 6.5.

The digital interpolator acts as a filter whose rejection to SCALE noise, for a given  $f_p$  and baseband bandwidth, increases with increasing value of  $N_r$ . In the same time, however, the distortion suffered by the input signal also increases with  $N_r$  (see Fig. 6.5). For  $f_p = 64$  Kb/s (i.e.  $N_r = 8$ ) is a reasonable compromise between noise rejection and baseband distortion. In this case the SCALE baseband signal - which is usually bandlimited by the SCALE input filter to  $\approx 2.4$   $KH_z$  - will suffer a distortion which is less than 1 dB, while the SCALE noise is effectively attenuated. A further advantage of this arrangement is the fact that it has zeros at all multiples of the linear PCM Nyquist frequency (8, 16, 24, 32..... $KH_z$ ).

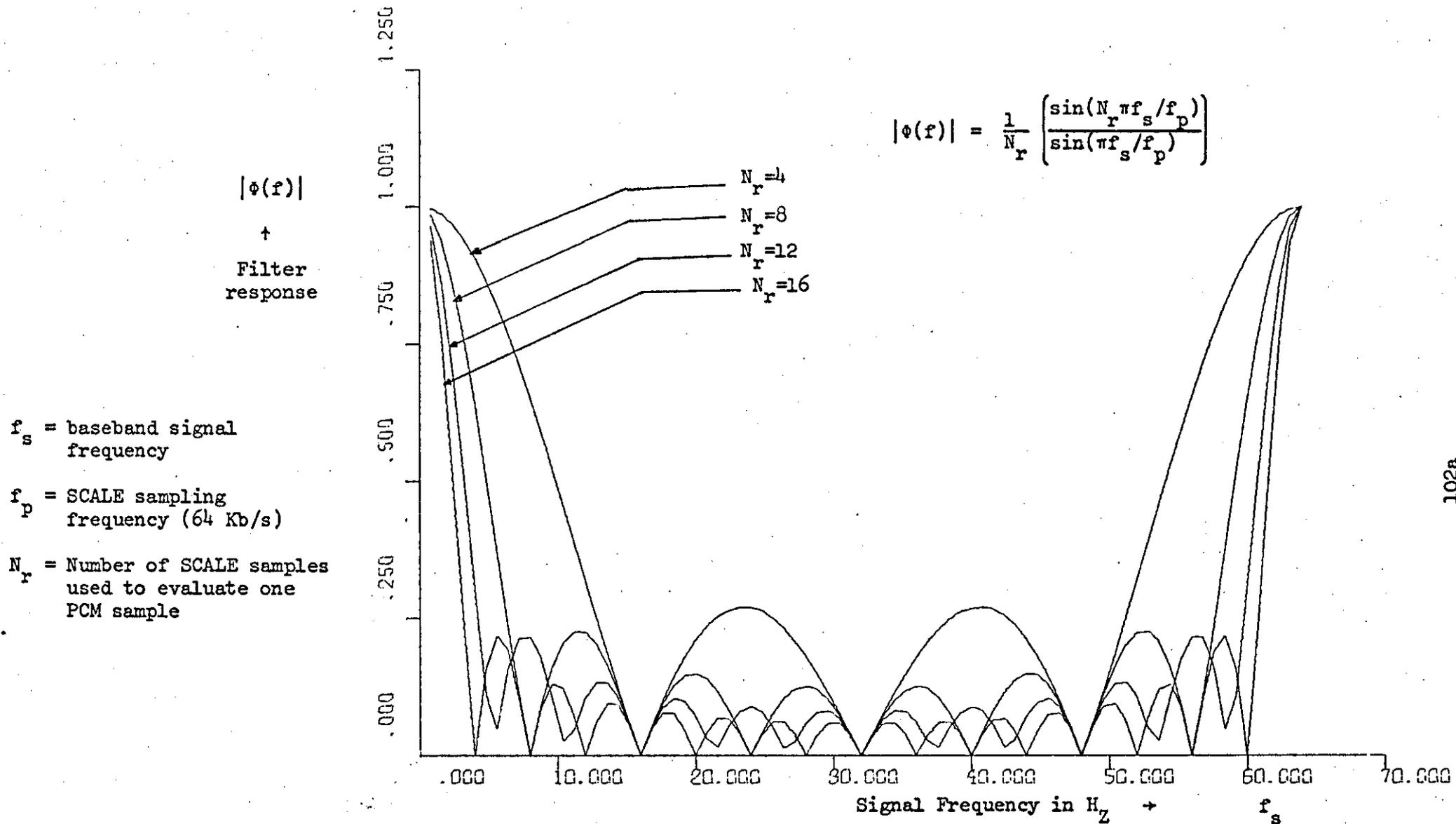


Figure 6.5 The transfer function of the Digital interpolator in the frequency domain

When dealing with linear Deltamodulation signals the uniform digital filter (accumulator + divider) approach is not a suitable one because of the fact that, in addition to the prohibitive clock rates required, it is not possible to reduce the filter bandwidth by increasing  $N_r$  without suffering large baseband distortion. The results presented by Goodman and Greenstein<sup>(81)</sup> confirm this remark. In the case of SCALE the spectrum of the signal at output of the SCALE decoder has been shaped by the syllabic companding effect in such a manner that the lower frequencies of the baseband signal predominate. That is, the spectrum of the output of the SCALE decoder is matched to the uniform digital filter characteristic. This enables Analogue-to-PCM converters to combine low clock rates with simple interpolator circuit.

### 6.3 THE DIGITAL COMPRESSOR

The task of the digital compressor DCOMP is to accept the linear PCM code words  $X_L$  generated by the digital interpolator (accumulator + divider) and it produces A-law PCM compressed code words  $X_c$ . The digital code words are supplied to the DCOMP at a rate of 8  $\text{KH}_z$ .

#### 6.3.1 Logarithmic companding laws - background

The main reason for logarithmic companding is to make use of the statistical properties of speech to obtain a wide dynamic range and better signal-to-noise ratio performance. The two well known logarithmic companding laws, the  $\mu$ -law and the A-law, are smooth input/output curves describing the transfer characteristics of the PCM encoder and decoder. The theory of logarithmic companding and the choice of the companding law for speech signals is widely discussed in the literature<sup>(9,43,82)</sup>. It is, however, worth recalling that the A-law is represented by the relation:

$$\begin{aligned}
F(x) &= \operatorname{sgn}(x) \left\{ \frac{1 + \ln A \cdot |x|}{1 + \ln A} \right\} & 1/A \leq |x| \leq 1 & \quad (6.7) \\
&= \operatorname{sgn} \left\{ \frac{A \cdot |x|}{1 + \ln A} \right\} & 0 \leq |x| \leq 1/A &
\end{aligned}$$

where  $|x|$  is the magnitude of the input signal to the compressor,  $F(x)$  is the compressed output signal and  $A (= 87.6)$  is a parameter suggested by Cattermale<sup>(1)</sup>; it determines the degree of companding.

At one time logarithmic laws were approximated by nonlinear devices such as diodes. Now these laws are implemented by piece-wise linear approximation techniques using several segments. As opposed to the smooth companding laws, segment companding laws permit more efficient signal processing on digital basis. They are capable of digital linearization<sup>(84)</sup>, and permit digital operations such as filtering and digital attenuation<sup>(85)</sup>, companding law conversion<sup>(86,87)</sup>, etc. A unified formulation of segmented companding laws was first presented by H. Kaneko in 1970. A similar study was published by W. L. Montgomery about the same time. The analysis presented by H. Kaneko started from conventional concepts of  $\mu$ -law and A-law and derived expressions for the decoder output level. The result of Kaneko's study was later employed as a useful vehicle for the synthesis of A-law-to- $\mu$ -law converters<sup>(87)</sup> and digital attenuators<sup>(85)</sup>. The analysis also leads to systematic synthesis of digital compressors and digital expanders.

The theory of segment laws will now be reviewed and the A-law digital compressor is then synthesised.

### 6.3.2 Theory of Segment Laws

Referring to figs. 6.6 and 6.7, suppose that a digital compressed code  $X_c$  is composed of  $m_1$  binary digits called "characteristic bits"

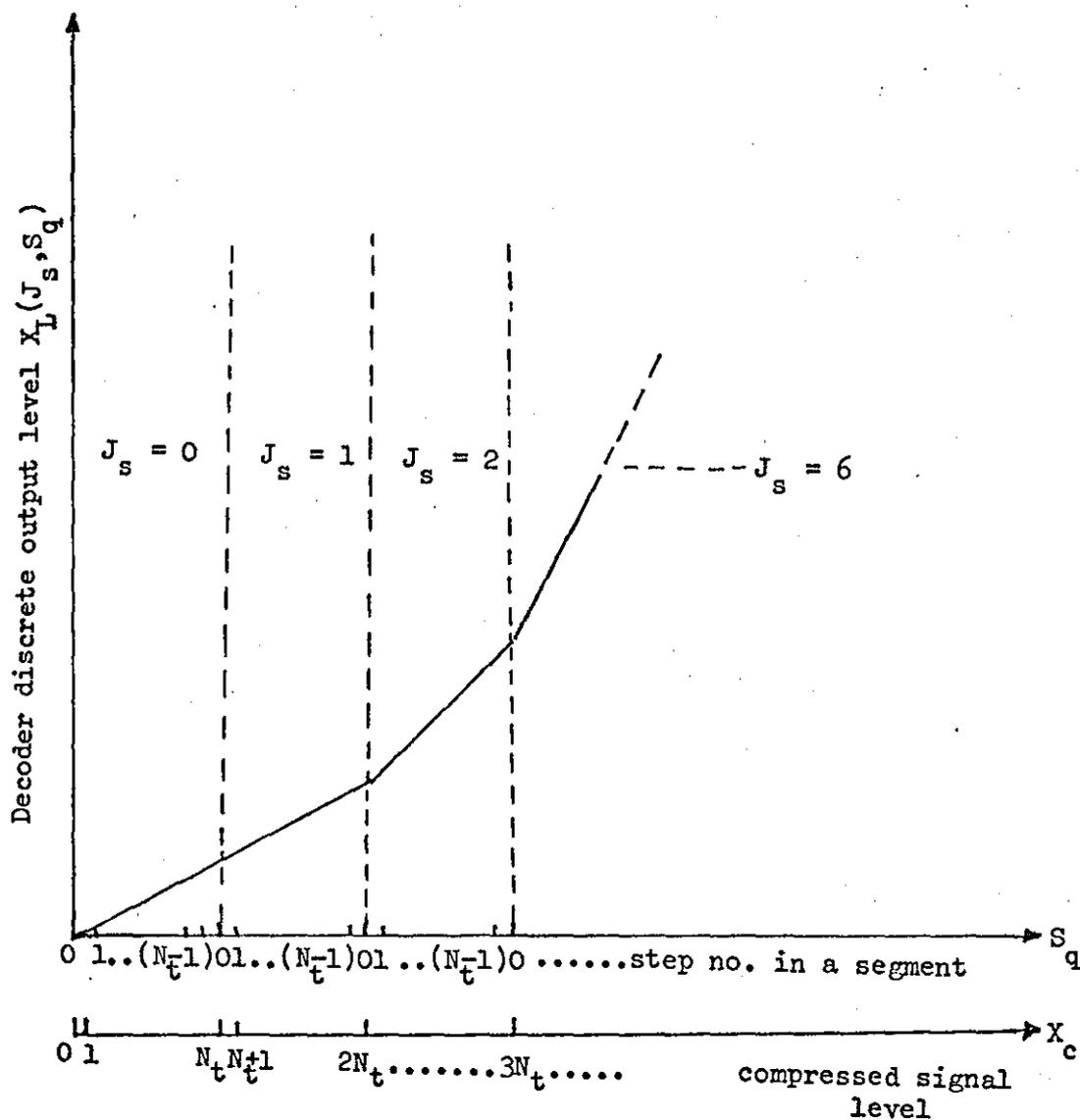


Figure 6.6 Segmented A-law - standard form

$J_s$  = segment number =  $\{0, 1, \dots, 7\}$  in each polarity

$S_q$  = step number within a segment =  $\{0, 1, \dots, 15\}$

$N_t$  = total number of steps in a segment  
= 16

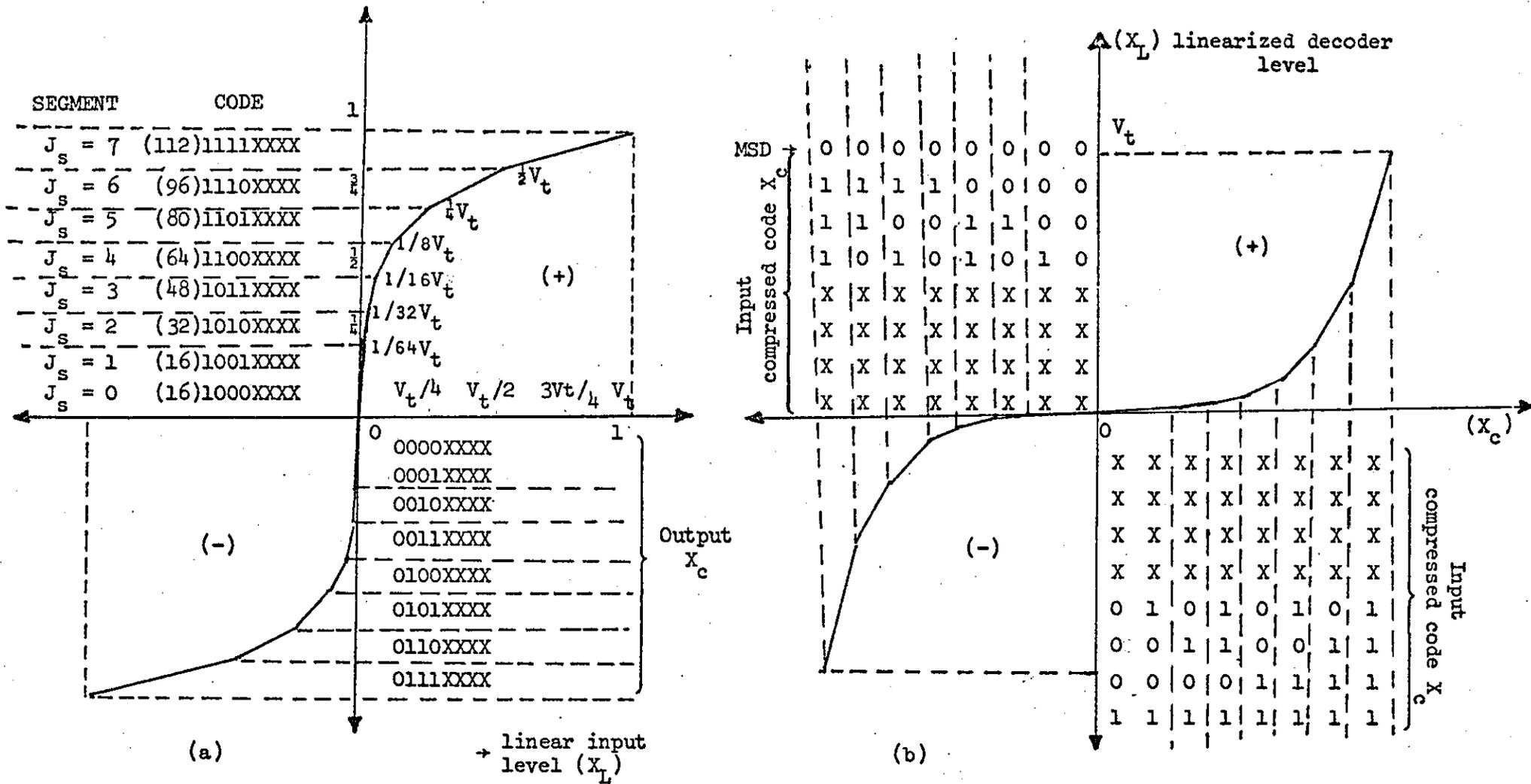


Figure 6.7 Quantization and coding used in CEPT 30+2 A-law PCM system

- (a) Encoder compression characteristic
  - (b) Decoder expander characteristic
- }  $X = \{0, 1\}$   
 $J_s = \text{segment number}$

representing the segment number  $J_s$  and  $m_2$  binary digits called "mantissa bits" representing the quantizing step  $S_q$  in a segment. The total number  $M_t$  of segments in one polarity is equal to  $2^{m_1}$  and the total number  $N_t$  of quantizing steps within a segment is  $2^{m_2}$ . Then the digital compressed code word  $X_c$  may be expressed in terms of  $J_s$  and  $S_q$  as:

$$X_c(J_s, S_q) = S_q + N_t \cdot J_s \quad (6.8)$$

where  $X_c = \{0, 1, 2, \dots, N_t \cdot M_t\}$  = the digital compressed code  
 $J_s = \{0, 1, 2, \dots, M_t - 1\}$  = the segment number  
 $S_q = \{0, 1, 2, \dots, N_t - 1\}$  = the step number within a segment

The digitally linearized (expanded) digital word  $X_L$  is a function of the compressed code  $X_c$ , i.e.

$$X_L = X_L(X_c) = X_L(J_s, S_q) \quad (6.9)$$

This is the output of the PCM decoder - before filtering - in binary format (see Fig. 6.6). The quantizing step size is then given by the difference of two adjacent output levels and is given by

$$\Delta_c(J_s) = X_L(J_s, S_q + 1) - X_L(J_s, S_q) \quad S_q \neq (N_t - 1)$$

$$\Delta'_c(J_s) = X_L(J_s + 1, 0) - X_L(J_s, N_t - 1) \quad S_q = (N_t - 1) \quad (6.10)$$

#### Definition of the segment A-law

The so-called segment A-law should satisfy the following conditions:-

(a) Except for the segment edges, the ratio of the step size of the adjacent segments is

$$\frac{\Delta_c(J_s + 1)}{\Delta_c(J_s)} = \begin{cases} 1 & , & J_s = 0 \\ 2 & , & J_s \in \{1, 2, \dots, M_t - 1\} \end{cases} \quad (6.11)$$

This implies that the 0<sup>th</sup> and 1<sup>st</sup> segments constitute a long co-linear

segment and the step size ratio of the adjacent segments elsewhere is equal to 2.

$$(b) \quad X_L(0,0) = P_c \quad (\text{centering parameter}) \quad (6.12)$$

$$(c) \quad \Delta_c(0) = 1 \quad (\text{normalization}) \quad (6.13)$$

A solution of equation (6.11) that satisfies (6.10), (6.12) and (6.13) is

$$X_L(J_s, S_q) = \begin{cases} S_q + P_c - P_a & J_s = 0 \\ 2^{(J_s-1)} \cdot (S_q + N_t) + P_c - P_a & J_s \neq 0 \end{cases} \quad (6.14)$$

where  $P_a$  is the segment edge parameter the value of which is between 0 and 1. The centering parameter  $P_c$  determines the origin (midread or midriser), and the edge parameter  $P_a$  describes the discontinuity of the step size at the segment edges. It has been shown by H. Kaneko<sup>(86)</sup> that the effect of  $P_a$  on the system performance is negligible and  $P_c$  for A-law is 0.5.

For A-law equation (6.14) can be rewritten in the general form

$$X_L(J_s, S_q) = \Delta_c(J_s) \cdot (S_q + P_t) - k_a \quad (6.15)$$

$$\text{where} \quad \left. \begin{aligned} \Delta_c(J_s) &= 2^{J_s - \eta_a} \\ P_t &= N_t \cdot \eta_a + P_a \\ k_a &= P_a - P_c \\ &= 0 \quad \text{for } P_a = 0.5 \\ \eta_a &= \begin{cases} 0, & J_s = 0 \\ 1, & J_s \neq 0 \end{cases} \end{aligned} \right\} \quad (6.16)$$

The A-law PCM decoder operation can therefore be explained in the following manner:

Every Nyquist interval the decoder accepts an  $(m_1+m_2+1)$  binary digits sequence representing the magnitude and sign of the signal represented by the compressed code word  $X_c$ . On reception of this sequence the decoder partitions it into 3 groups as shown below:

$$\begin{array}{cccccccc}
 & & & & X_c & & & \\
 \hline
 \varepsilon_8 & \varepsilon_7 & \varepsilon_6 & \varepsilon_5 & \varepsilon_4 & \varepsilon_3 & \varepsilon_2 & \varepsilon_1 \\
 \hline
 \text{sign} & \text{segment no.} & \text{step number} & & & & & \\
 & J_s & S_q & & & & & 
 \end{array} \quad (6.17)$$

It then determines the sign from the most significant digit (MSD  $\varepsilon_8$ ),  $J_s$  the number of the segment - in which the level lies - from the next 3 most significant digits ( $\varepsilon_7, \varepsilon_6, \varepsilon_5$ ) and the step number  $S_q$  within the segment from the 4 least significant digits ( $\varepsilon_4, \varepsilon_3, \varepsilon_2, \varepsilon_1$ ).

Therefore

$$\begin{aligned}
 J_s &= J_s(\varepsilon_{m_1+m_2}, \varepsilon_{m_1+m_2-1}, \dots, \varepsilon_{m_2+1}) \\
 &= \sum_{k=1}^{m_1} 2^{(m_1-k)} \cdot \varepsilon_{(m_2+m_1-k+1)} \quad (6.18)
 \end{aligned}$$

$$\begin{aligned}
 S_q &= S_q(\varepsilon_{m_2}, \varepsilon_{m_2-1}, \dots, \varepsilon_1) \\
 &= \sum_{k=1}^{m_2} 2^{(m_2-k)} \cdot \varepsilon_{(m_2-k+1)} \quad (6.19)
 \end{aligned}$$

where for 8 digit word A-law PCM  $m_1 = 3, m_2 = 4$ .

Having done these calculations the decoder forms  $\Delta_c(J_s)$  according to equation (6.16) and subsequently  $X_L(J_s, S_q)$  according to equation (6.15). This latter operation can be interpreted as:

- (i) adding  $P_t$  to  $S_q$
- (ii) multiplying the sum by  $\Delta_c(J_s) = 2^{J_s - \eta_a}$  (i.e. shifting the sum of  $(P_t + S_q)$  by  $(J_s - \eta_a)$  to the left.
- (iii) subtract  $k_a$  (this is not necessary in the case of the A-law with  $P_a = 0.5$  since  $k_a = P_a - P_c = 0$ ).

Further details of the theory of the smooth and segmented A-laws can be found in references referred to earlier in this chapter.

### 6.3.3 The compression algorithm and synthesis of the compressor

We consider the conversion from a linear PCM code  $X_L$  to a non-linear code  $X_c$ . The linear code  $X_L$  is a positive real number representing a signal amplitude normalized by the minimum step of the compressed code ( $\Delta_q(Z=0, J=0) = 1$  mv).

The linear code is represented by equation (6.15). To form the compressed code word  $X_c$  we require to know  $J_s$  and  $S_q$  (see equation (6.17)). These can be determined from the  $X_L$  binary sequence (beginning with the LSD). Now for A-law, equation (6.15) reduces to

$$X_L(J_s, S_q) = 2^{J_s - \eta_a} \{S_q + \eta_a N_t + 0.5\} \quad (6.20)$$

Investigation of this equation reveals that if the linear code word  $X_L(J_s, S_q)$  is shifted  $(J_s - \eta_a)$  times to the right (division) the resultant binary sequence represented by  $\{S_q + \eta_a N_t - 0.5\}$  must lie in the  $0^{\text{th}}$  segment, i.e.  $(S_q + \eta_a N_t - 0.5)$  lies between 0 and  $N_t$  ( $\eta_a = 0$  when  $J_s = 0$ ). Conversely, if the linear code  $X_L(J_s, S_q)$  is shifted towards the LSD until the result lies between the limits 0 and  $N_t$  then the number of times by which the linear code has been shifted is  $J_s$  ( $\eta_a = 0$  in the  $0^{\text{th}}$  segment). This means that the search for  $J_s$  can be accomplished by shifting the linear code  $X_L(J_s, S_q)$  to the right and

counting the number of times this shifting has been performed, until the resultant code value lies between 0 and  $N_t$ . The result of the counting during the search gives  $J_s$ . Shifting  $X_L(J_s, S_q)$  by  $J_s$  to the direction of LSD therefore results in a binary sequence representing  $(S_q - 0.5)$ . So rounding of this sequence (e.g. by truncation of fractions) gives  $S_q$  in binary representation. Signifying these values of  $J_s$  and  $S_q$  - obtained above by search and determination - by  $\hat{J}_s$  and  $\hat{S}_q$  respectively the compression operation can now be summarized:

- (1) search the linear code  $X_L$  to find  $\hat{J}_s$
- (2) shift  $X_L$  sequence to the right by  $(\hat{J}_s - 1)$  places and read  $\hat{S}_q$  from the least significant four digits of the shifted  $X_L$  sequence.

Now, the linear PCM code word  $X_L(J_s, S_q)$  which is supplied to the compressor by the Digital Interpolator is composed of 11 parallel binary digits  $(u_1, u_2, \dots, u_{11})$  representing the magnitude of the base-band signal  $|x_2(t)|$  at the input of the SCALE encoder. Therefore

$$X_L(J_s, S_q) = \sum_{k=1}^{11} 2^{k-1} u_k \quad (6.21)$$

An alternative approach to the search for  $J_s$ , given the linear PCM binary sequence, is to employ a 3-bit binary counter which inspects  $u_5, u_6, \dots$  to  $u_{11}$  in turn. The counter content is reset to "all ones" whenever  $u_k$  ( $k = 5, 6, \dots, 11$ ) is a logical one. The counter counts down as long as  $u_k$  is a zero. At the end of the inspection, the content of the binary counter is a binary representation of  $\hat{J}_s$ .  $\hat{S}_q$  can then be determined from the linear PCM sequence  $X_L(J_s, S_q)$  by shifting the sequence by  $(J_s - 1)$  position to the direction of LSD.

Based on this approach the required digital compressor for the SCALE-to-A-law PCM converter was designed in the form shown in Fig. 6.8.

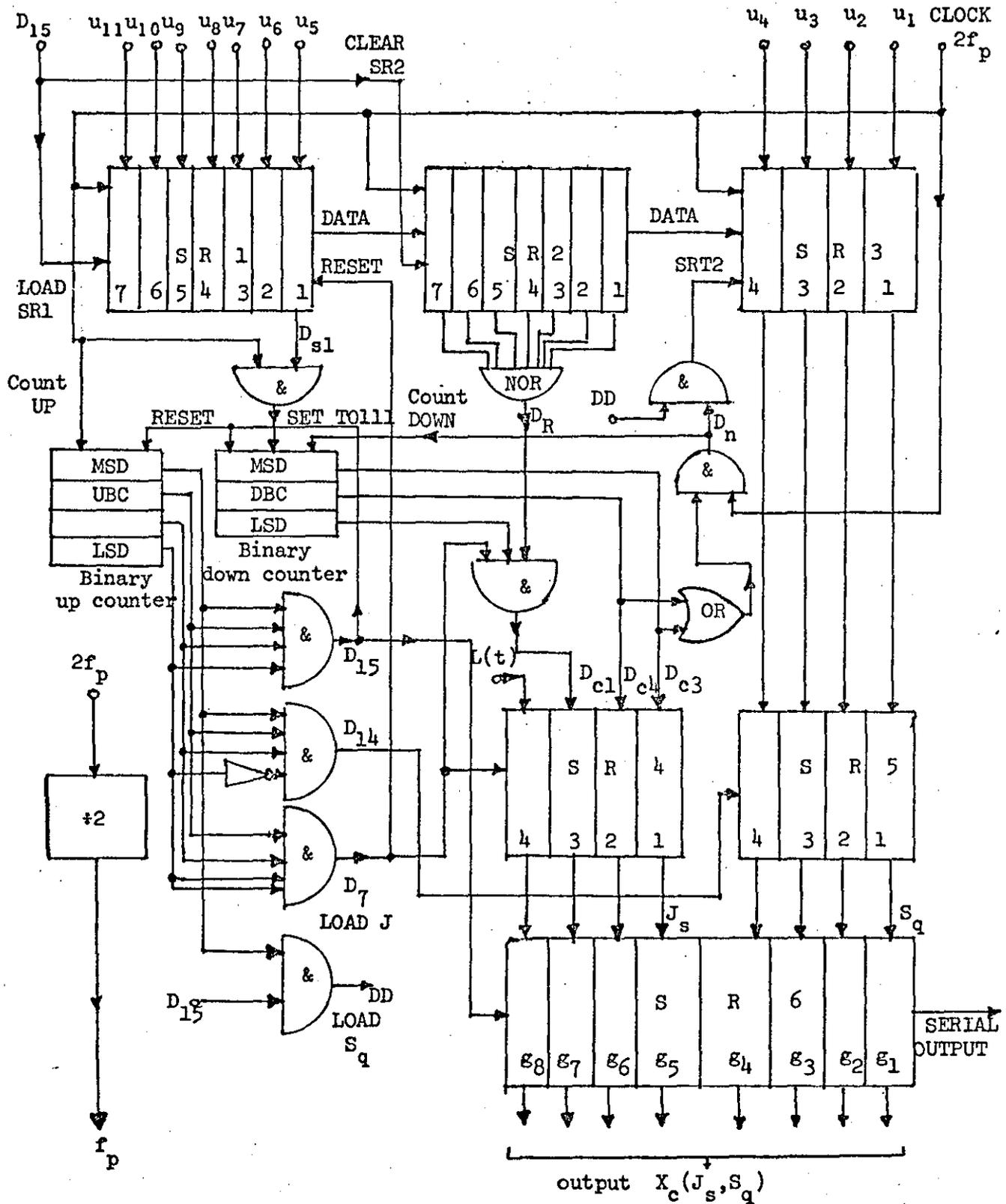


Figure 6.8 The digital compressor

SR1, SR3, SR4, SR5, SR6 = PIPO shift registers  
 SR2 = FIFO " "  
 $u_k$  ( $k=1,2,\dots,15$ ) = The binary sequency representing  $X_L$   
 $\epsilon_k$  ( $k=1,2,\dots,7$ ) = " " " "  
 $\epsilon_8$  = sign digit ( $\epsilon_8=1$  means signal positive)  
 $f_p$  = SCALE sampling frequency ( $=6^{1/2}$  Kb/c)

Its operation is explained below.

#### 6.3.4 Operation of the Digital A-law PCM Compressor

At the beginning of every Nyquist period ( $1/8000$ ) sec. the linear PCM word  $X_L = \{u_1, u_2, \dots, u_{11}\}$  is presented to the compressor. At this instance several operations are carried out instantaneously. These operations are:

- (i) The 7-bit First-in-First-out (FIFO) shift register  $SR_2$  is cleared.
- (ii) The 4 least significant digits of  $X_L$  are loaded in parallel into the 4-bit parallel-in-parallel-out (PIPO) shift register  $SR_3$ .
- (iii) The remaining 7 most significant digits of  $X_L$  are loaded in parallel into the PIPO shift register  $SR_1$ .
- (iv) The compressed code word comprising  $\hat{S}_q$  and  $\hat{J}_s$  corresponding to the previous  $X_L$  word, together with its polarity digit, are loaded in parallel into the output register  $SR_6$ .  $SR_6$  is an 8-bit PIPO shift register. The function of this register is to hold one compressed A-law PCM word while the next word is being processed by the other parts of the DCOMP.

At this time the contents of the binary counter DBC and UBC, which are respectively 3-bit down counter and 4-bit up counter, are all zeros. A master clock (clock 2) runs at twice  $f_p$  and its frequency is therefore  $16f_q$ . This means that there are 16 pulses of clock 2 in each Nyquist period. Let the period of clock 2 be  $T'_p = 1/2f_p = T_p/2$ .

During the first  $7T'_p$  seconds of each Nyquist period the following operations are all taking place simultaneously:

- (i) The clock pulses of clock 2 are counted by the binary Up-counter UBC.
- (ii) The contents of  $SR_1$  are shifted from left to right into  $SR_2$  under the control of clock 2.
- (iii) In the same time the  $2^{\text{nd}}$  binary counter DBC inspects the digit in

position 1 of  $SR_1$  and it resets its contents to all ones whenever this digit is a logical one. When no logical one is encountered the DBC counts down under the control of its content, clock 2 and UBC. The content of the DBC together with clock 2 control the counting down during the first  $6T_p'$  seconds of each compression period, while the seventh consecutive count down is controlled by clock 2 and  $D_7$  which indicates that seven clock 2 pulses have been counted. When the content of UBC is exactly 7 (0111) the content of the DBC is a binary representation of  $\hat{J}_s$  the segment number. At this instance  $D_7$  is a logical one.

- (iv) When  $D_7$  becomes a logical one  $\hat{J}_s$  is loaded in parallel into the 3 LSD<sup>s</sup> of the 4-bit PIPO shift register  $SR_4$  together with the sign digit  $L_i$  (see Fig. 6.2).

At the same time  $SR_1$  is cleared. The clearing of  $SR_1$  inhibits any further resetting of DBC to "all ones".

During these first 7 clock pulses - counted by UBC - the contents of  $SR_3$  remain constantly unchanged. At the end of this period the binary sequence  $(u_5, \dots, u_1)$  representing the 7 most significant digits of  $X_L$  will have all been moved to  $SR_2$  ( $u_5$  in the position adjacent to  $u_4$ ) and in the same time the search for  $\hat{J}_s$  has been performed.

When the contents of the UBC become  $\geq 8$  the shifting of the contents of  $SR_3$  to the right, under the control of clock 2 and the contents of the DBC, is enabled and the following operations are simultaneously executed:

- (i) The binary counter DBC counts down under the control of clock 2 as long as its content  $> 1$ .
- (ii) As long as the inequality above is true the contents of  $SR_3$  are shifted to the right while the contents of  $SR_2$  are also shifted one position to the right at a time into  $SR_3$ .

- (iii) Shifting from  $SR_2$  to  $SR_3$  is inhibited when the content of the DBC  $\leq 1$ . At this time the content of  $SR_3$  is a binary representation of  $\hat{S}_q$  (the step number in segment  $\hat{J}_s$ ). The maximum time necessary for reducing the content of the DBC to 1 is  $6T'_p$  seconds. So  $\hat{S}_q$  can be loaded-in-parallel into the 4-bit PIPO shift register  $SR_5$  when the content of the UBC is exactly 14.
- (iv) The pulse  $D_{14}$  is also used to reset the contents of DBC and UBC to all zeros preparing the circuit for the next cycle of compression.

It must be noted that the shift register  $SR_6$  is not part of the compressor. It is included only for convenience. However, if the A-law digital signal is to be transmitted in compressed form  $SR_6$  can be utilised as a serializer by clocking it at a rate of  $f_p$  b/s.

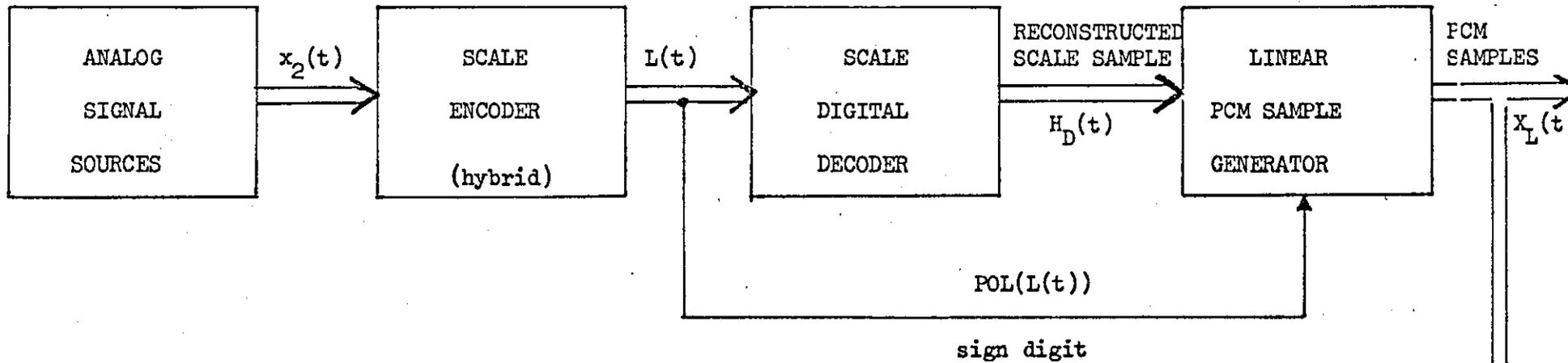
The timing signals ( $f_p$ ,  $f_q$ , etc.) can be easily derived from clock 2 using frequency dividers.

#### 6.4 COMPUTER SIMULATION OF THE SCALE-TO-PCM CONVERTER

To verify the theory of the SCALE-to-A-law PCM converter discussed above, the converter was implemented on a 1904A ICL digital computer, using FORTRAN IV. The simulation was carried out in two phases. The first phase was devoted to the modeling of the SCALE-to-linear PCM converter part of the system. The second phase was concerned with simulation of the complete converter which accepts SCALE binary data and produces A-law PCM words. These two phases of the simulation will now be described.

##### 6.4.1 The SCALE-to-linear PCM Converter (Phase 1)

The simulation was based on the block diagram shown in Fig. 6.9. The arrangement is based on the theory of conversion discussed earlier



$x_2(t)$  = an analog signal (d.c., sinusoidal bandlimited Gaussian noise) waveform of duration  $512 T_p$ .

$L(t)$  = a block of 512 data samples (bits) representing  $x_2(t)$  in SCALE digital format.

$H_D(t)$  = 512 PAM samples reconstructed by the SCALE decoder from  $L(t)$  samples.

$X_L(t)$  = 64 linear PCM PAM samples generated from the 512  $H_D(t)$  samples and the 512 polarity samples.

$x_2^1(t)$  = reconstructed baseband signal.

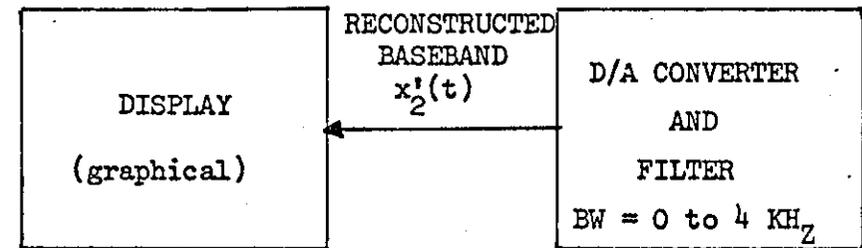


Figure 6.9 Block diagram of the SCALE-to-PCM converter arrangement for computer simulation

in this chapter. The FORTRAN IV implementation was achieved in the following manner:

(a) The analogue sources were implemented by the subprograms "SUBROUTINE SIGNAL" and "SUBROUTINE GAUSS2". These two subroutines have already been discussed in Chapter III and are listed in Appendix B. A "CALL" statement - with the appropriate arguments (number of samples, type of signal, etc.) - to "SUBROUTINE SIGNAL" results in its execution and the "RETURN" of 512 d.c. or sinusoidal waveform of duration  $512T_p$  seconds. Alternatively, a "CALL" to "GAUSS2" results in a Gaussian distributed noise waveform being "returned" to the master segment.

(b) The analogue SCALE accepts the selected analogue waveform and encodes it into 512 binary digits representing the sign of the difference between the input samples of  $x_2(t)$  and their locally generated estimate. This operation was implemented on the computer by the subprogram "SUBROUTINE CODEC" described in Chapter III. A listing of this subroutine is given in Appendix B.

(c) The digital SCALE decoder (including the D/A converter which converts the digital SCALE words to SCALE reconstructed samples) is exactly the same as that discussed in Chapter V with the exception of the overload point which has now been reduced from 2.5 volts to 2.048 volts for compatibility with the linearized A-law PCM maximum step voltage ( $2048 \times \text{min. step}$ ). Because here we were only interested in the SCALE and the linear PCM sample values (PAM samples) the simulated SCALE digital decoder was implemented in FORTRAN IV according to equation (5.1), i.e.

$$\begin{aligned}
 H_i &= H_{i-1} + \Delta(Z, H_{i-1}) \\
 &= H_{\min} && H_{i-1} \leq H_{\min} \\
 &= H_{\max} && H_{i-1} > H_{\max}
 \end{aligned}$$

$\Delta(Z, H_{i-1})$  satisfies Table 5.3 and 5.4, and the polarity of  $H_i$  is, as before, governed by the logical value of the input digit  $L_i$  to the SCALE decoder.

The digital SCALE decoder was implemented on the computer by a specially compiled subprogram named "SUBROUTINE DECODERD", which is listed in Appendix B.

In this experiment the digital SCALE accepts a block of 512 SCALE binary digits and produces 512 SCALE PAM samples  $H(t)$  every time the program is run.

(d) For every 8 SCALE PAM samples the linear PCM sample generator forms one linear PCM sample according to the relationship defined in equation (6.1), i.e.

$$X_L(i + 7T_p) = \frac{1}{8} \sum_{k=0}^7 H_{i+k} .$$

Note that  $X_L$  here is represented in base 10 arithmetic. So from every 512 SCALE PAM samples the linear PCM sample generator produces 64 linear PCM, PAM samples. The above equation was translated into FORTRAN IV and given the name "SUBROUTINE ESTIMATE". The polarity of each  $X_L(t)$  sample is positive when the SCALE binary digit  $L_{i+k}$  is a logical one and negative otherwise. "SUBROUTINE ESTIMATE" is listed in Appendix B.

The linear PCM PAM samples, produced at a rate of 8  $\text{KH}_2$ , are convolved with the impulse response of the digital Filter (designated LPF) to produce the reconstructed baseband signal  $x_2'(t)$  which is an estimate of  $x_2(t)$ . The low pass digital filter was implemented on the digital computer by the subprogram "SUBROUTINE FILTER" and the convolution operation by "SUBROUTINE CONVOL". These two subprograms have both been discussed in Chapter III and are listed in Appendix B. The various waveforms were displayed in graphical form using the Loughborough University computer library graph-plotting routines "UTP4C" and "UTP4B".

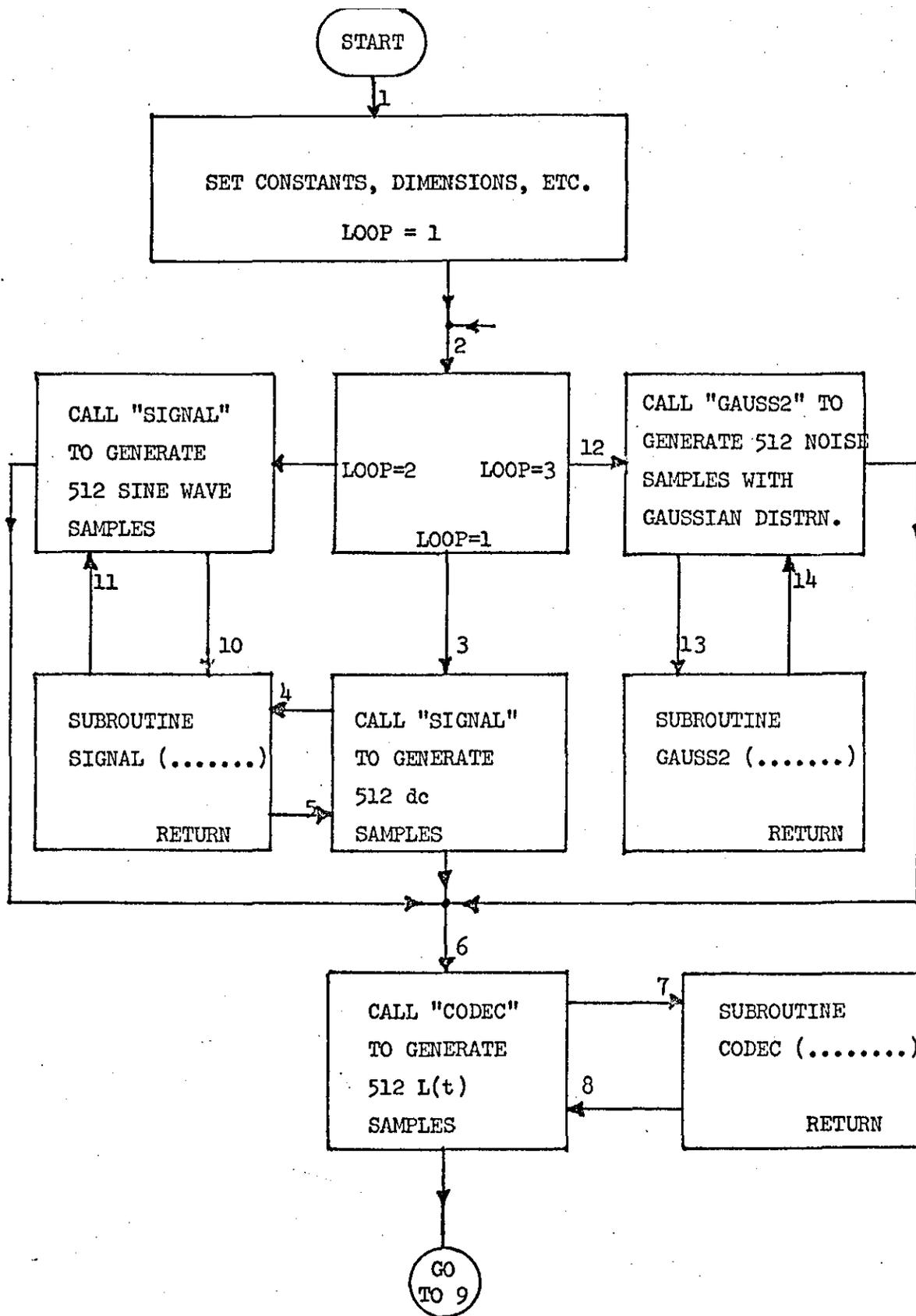
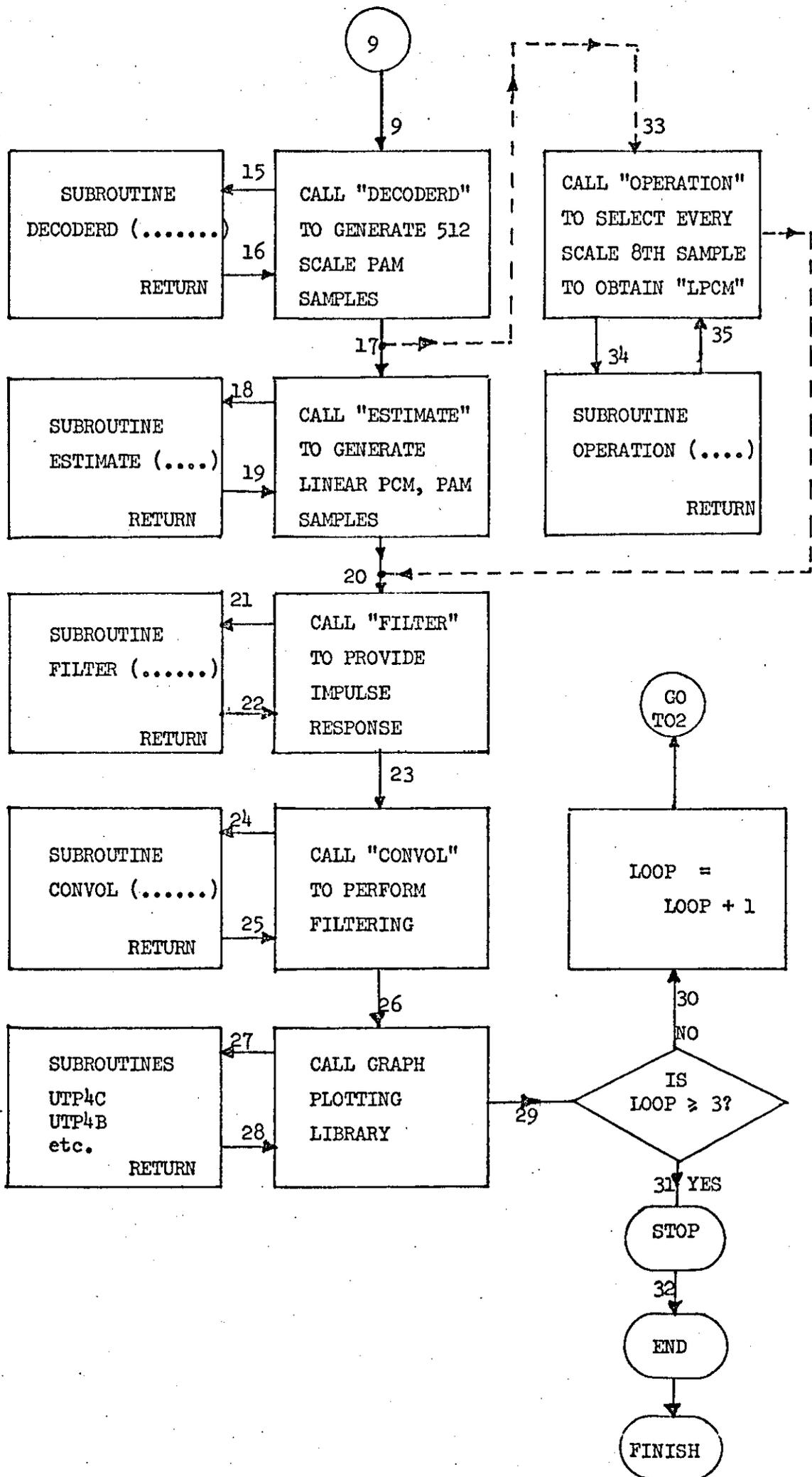


Figure 6.10 Flow chart of the program employed for simulating and testing of the SCALE-to-LPCM converter



To demonstrate the effect of the digital interpolator the simulation was repeated as follows:

A subprogram "SUBROUTINE OPERATION" was written and compiled. This subprogram can be called from the master segment to by-pass "SUBROUTINE ESTIMATE" as indicated by the dotted line in flow chart given in Fig. 6.10 (flow chart of the complete FORTRAN computer program). "SUBROUTINE OPERATION" accepts the 512 SCALE PAM samples and, instead of averaging every block 8 SCALE samples to obtain one linear PCM, PAM sample, it ignores 7 SCALE samples and selects every 8<sup>th</sup>. The selected 8<sup>th</sup> sample is then taken as the effective linear PCM PAM sample which is presented to the filter for removal of the out-of-band noise.

The input waveform to the SCALE encoder, the final reconstructed estimate of this waveform, and the other intermediate resultant time signals are displayed in Fig. 6.11. Figs. 6.11a and 6.11b show the reconstructed waveform at the output of SCALE-to-linear PCM converter, before and after filtering. These waveforms are "supposed" to represent a sinusoid with frequency  $f_s$  and amplitude  $A_s$  which are respectively 1000 Hz and 1.5 volts. The reconstruction of these waveforms from the SCALE binary data by first reconstructing the SCALE PAM samples and then resampling these samples at 8 KHz, was done using "SUBROUTINE OPERATION" (see the dotted line path in Fig. 6.10). Because large amounts of information have been omitted (7 SCALE PAM samples are ignored while only the 8<sup>th</sup> sample is taken to represent the baseband signal), the performance of this converter is so poor that the output of the converter is no longer a representative of the original baseband signal.

The improvement gained by performing the digital interpolation (digital filtering) operation, using "SUBROUTINE ESTIMATE" can be clearly seen by comparing Figs. 6.11a, 6.11b and Figs. 6.11d and 6.11e.

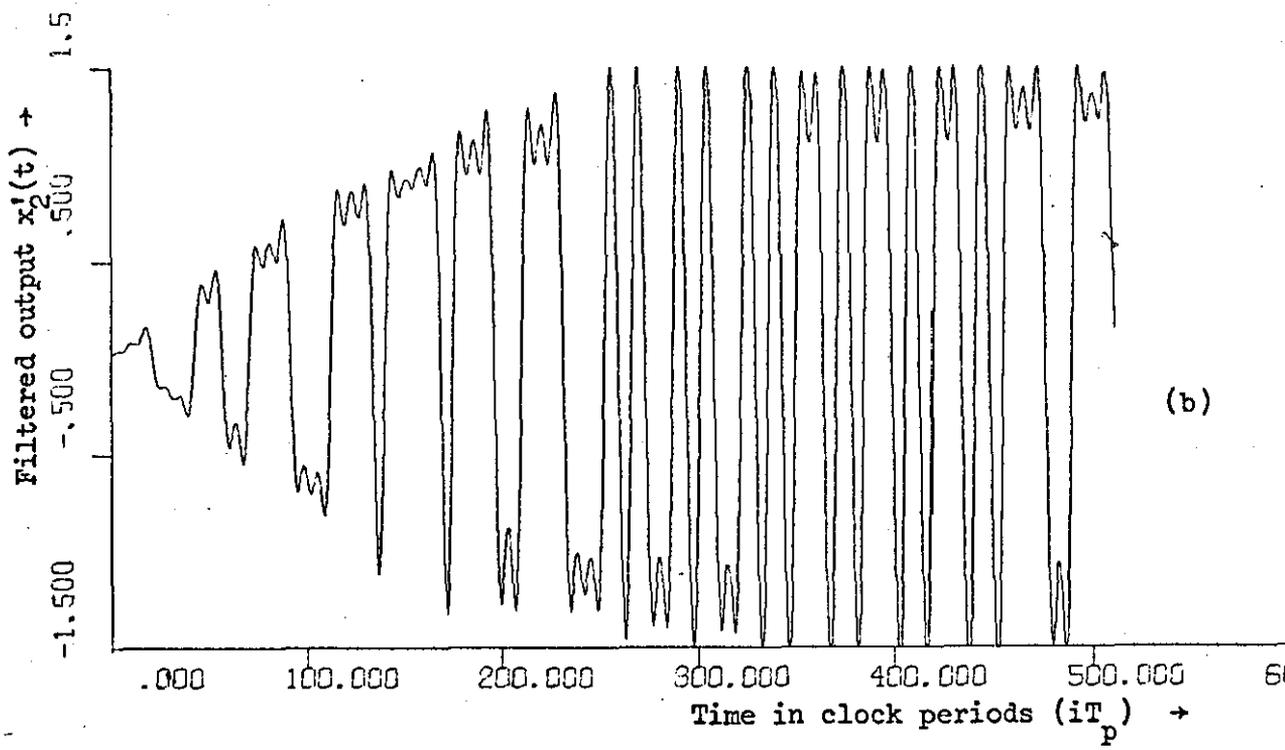
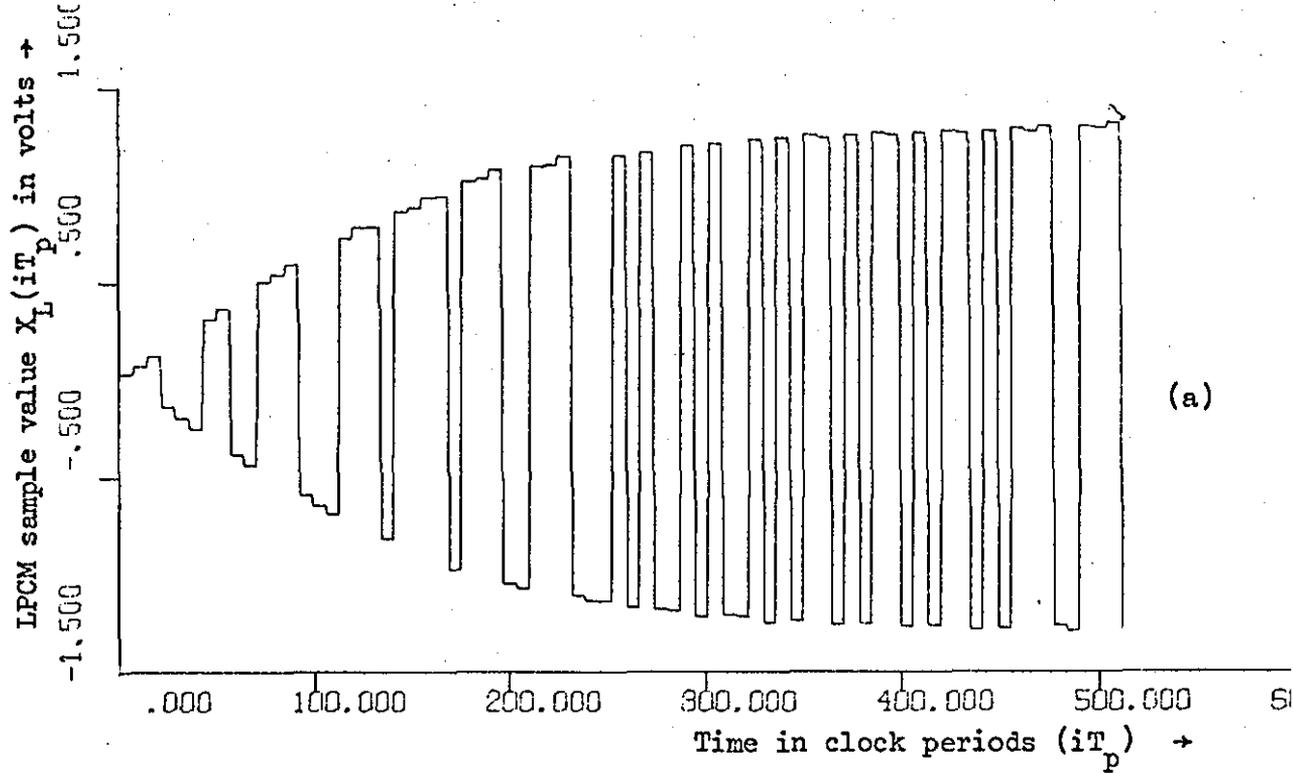
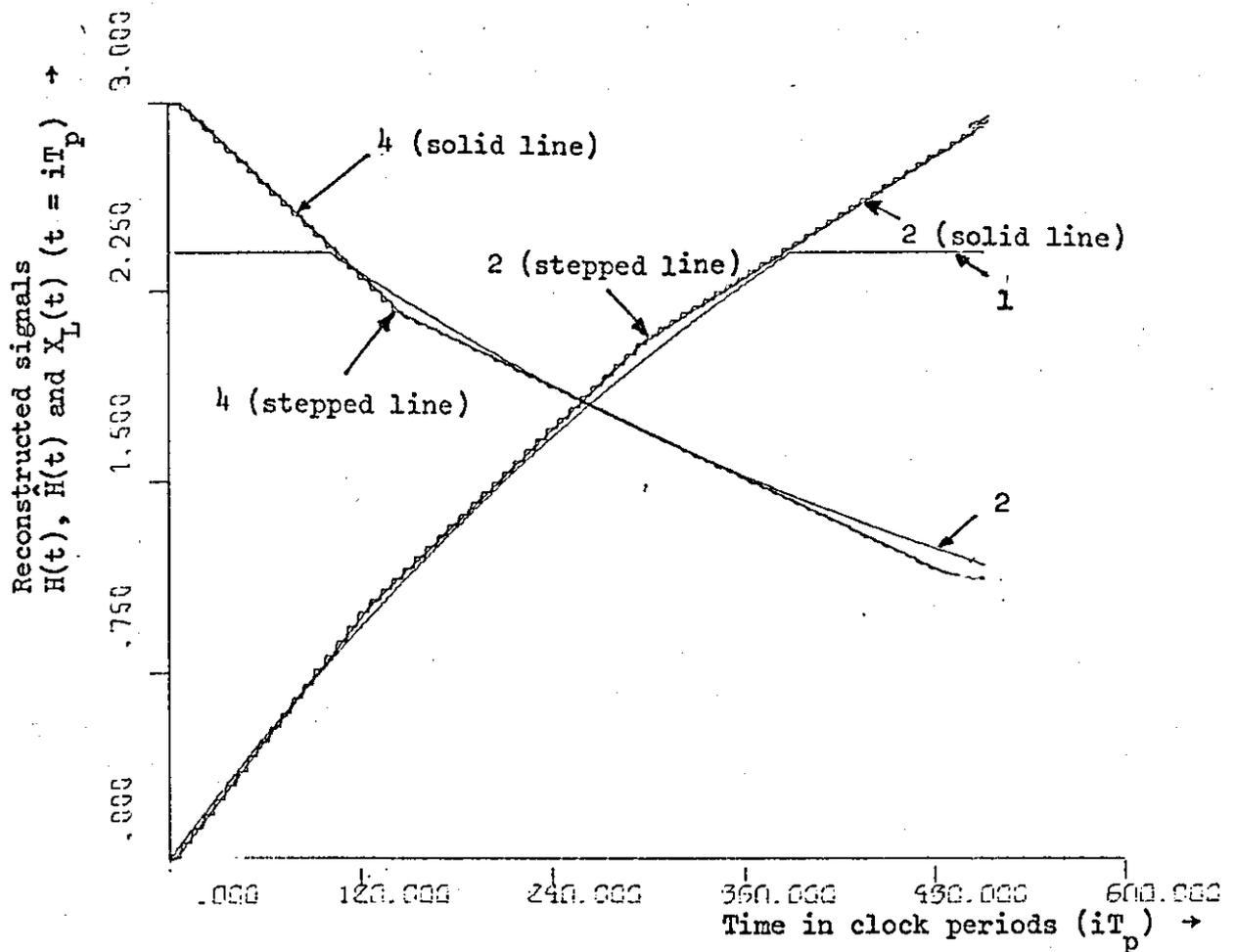


Figure 6.11 Results of the testing of the SCALE-to-LPCM converter

(a) LPCM PAM waveform constructed using "SUBROUTINE OPERATION". The SCALE encoder input was a sinusoid of frequency  $f_s = 1 \text{ KHz}$  and amplitude = 1.5 volts. "SUBROUTINE OPERATION" examines the set of 8 SCALE samples and selects the 8<sup>th</sup> sample to represent the LPCM level.

(b) Signal shown in (a) after passing through 4 KHz low pass filter. A poor representation of  $x_2(t)$ .



(c) Hybrid SCALE and Digital SCALE PAM levels and the corresponding generated linear PCM PAM levels as functions of time in clock periods when the input is a dc voltage; value 5 volts (curves 1, 2) followed by zero input (curves 3, 4).

Curve 1: Hybrid SCALE decoder output (HSCALE PAM)

when the input to the Hybrid SCALE encoder = 5 volts.

Curve 3: Hybrid SCALE decoder output (HSCALE PAM) when the

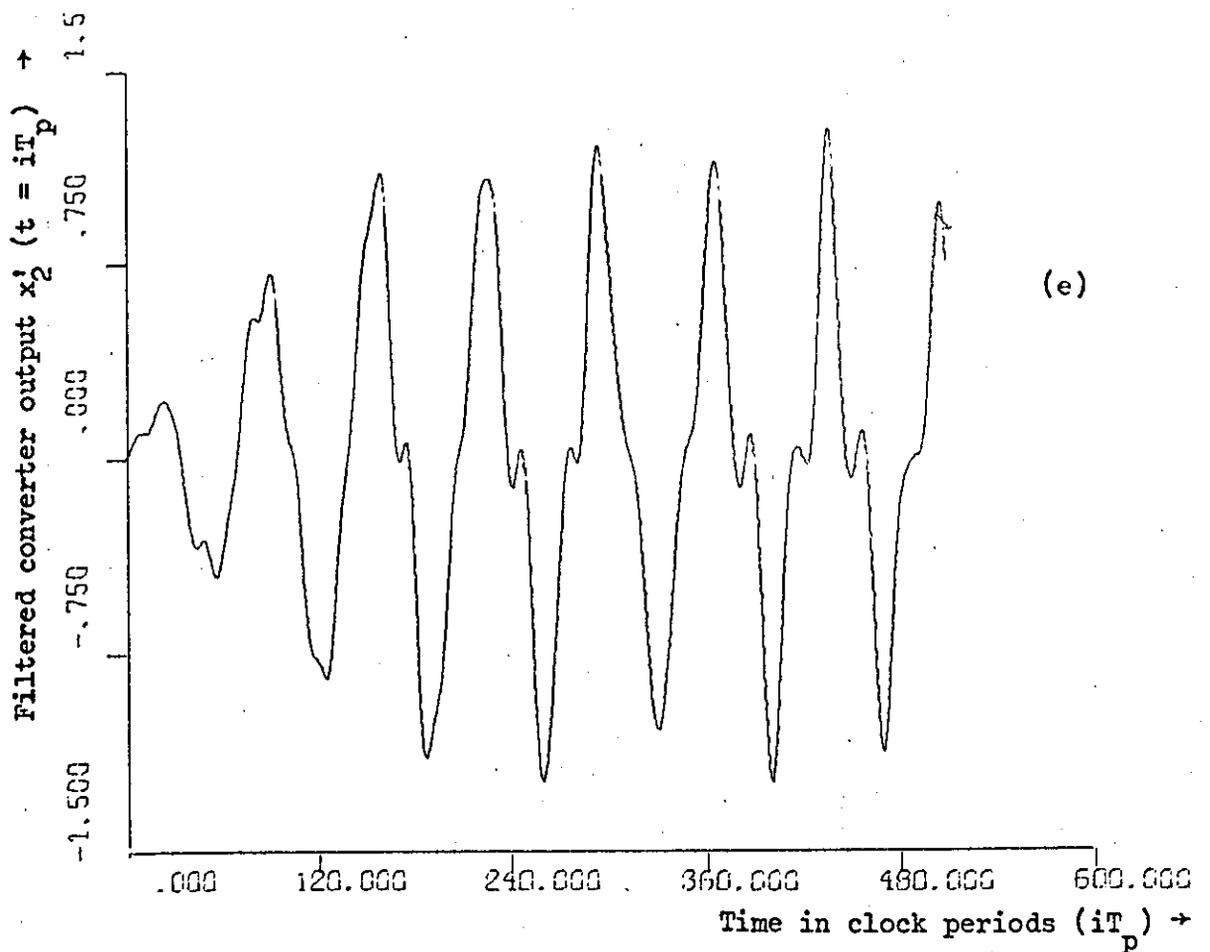
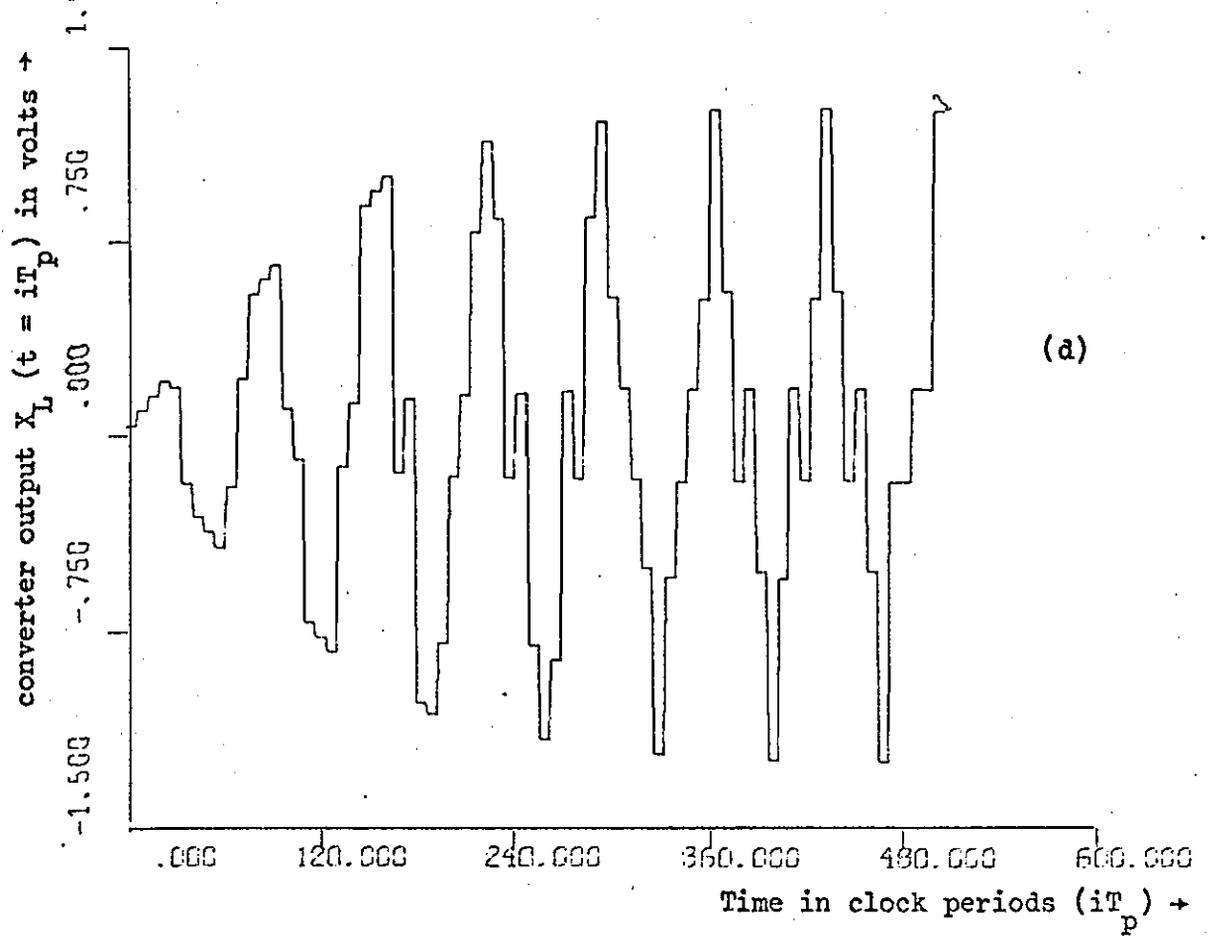
input to the hybrid SCALE encoder = 0.0 volts

Curve 2 (solid line): DSCALE decoder output (DSCALE PAM) I/P = 5V

Curve 4 (solid line): DSCALE decoder output (DSCALE PAM) I/P = 0V

Curve 2 (stepped line): Digital interpolator output (PCM PAM) I/P = 5V

Curve 4 (stepped line): Digital interpolator output (PCM PAM) I/P = 0V



- (d) Output of the SCALE-to-LPCM converter when "SUBROUTINE ESTIMATE" replaced "SUBROUTINE OPERATION" i.e. obtaining the LPCM by averaging every 8 consecutive DSCALE samples. Input signal as in (a) above.
- (e) Signal in (d) after passing through 4 KHz low pass filter.

"SUBROUTINE ESTIMATE" defines in FORTRAN IV the digital interpolation operation carried out on the SCALE PAM discrete samples. The digital interpolation algorithm which was adopted is as follows: "ACCEPT THE DIGITAL SCALE DISCRETE PAM SAMPLES, SUM EVERY 8 CONSECUTIVE SAMPLES, AVERAGE THE SUM OVER 8 SCALE CLOCK PERIODS (i.e. divide by 8); AND THEN READ THE AVERAGE AT PCM NYQUIST RATE ( $8\text{KH}_2$ ). THE SIGN OF THE PCM PAM SAMPLES IS GIVEN BY THE POLARITY OF THE LAST SCALE PAM SAMPLE USED IN THE AVERAGING OPERATION". The algorithm is defined by equation (6.1).

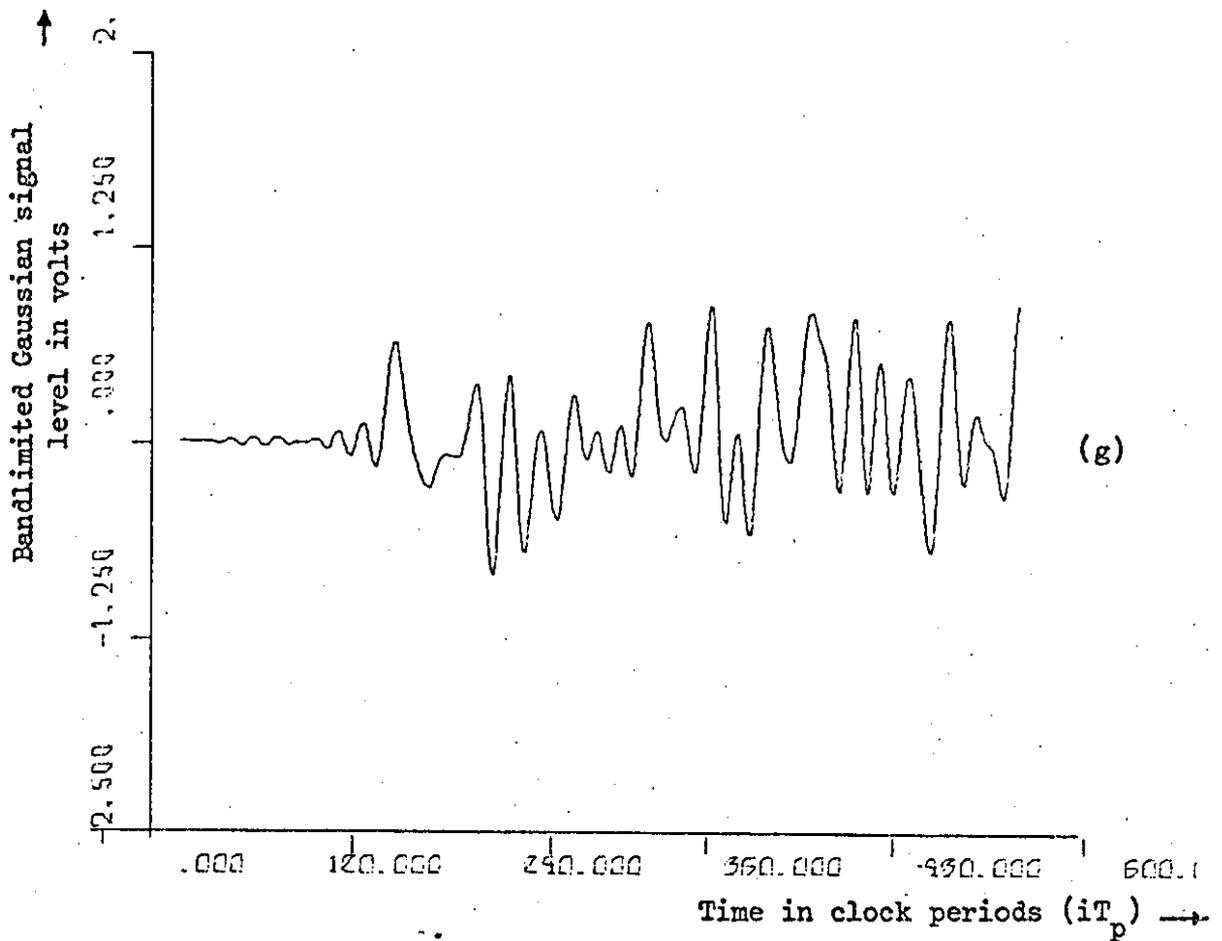
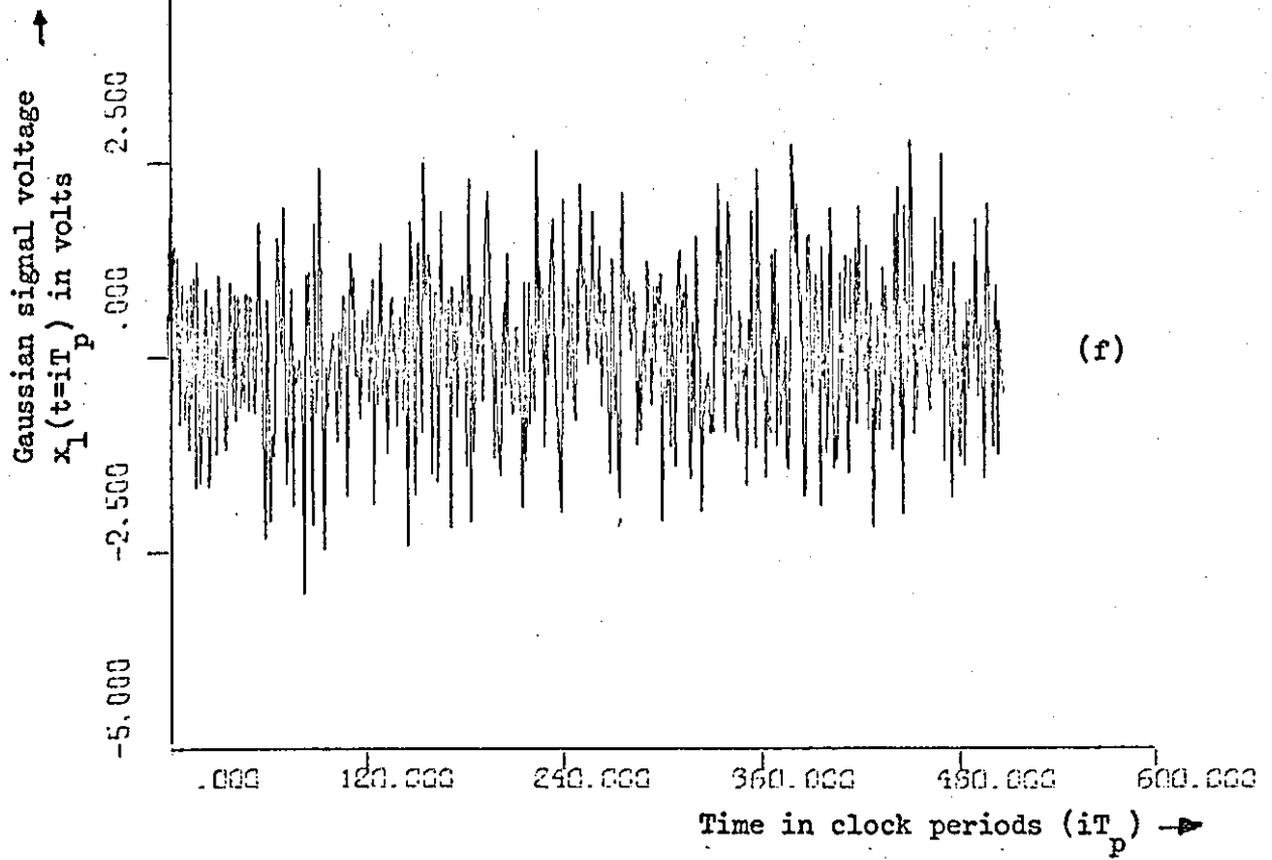
The generated linear PCM PAM samples, from the binary data originating from a Hybrid SCALE encoder (encoding d.c. and white Gaussian noise), are shown in Figs. 6.11c and 6.11h respectively. The extent of the agreement between the waveforms - reconstructed from the SCALE binary data - using Hybrid SCALE decoder, Digital SCALE decoder and SCALE-to-LPCM converter is demonstrated by the graphs shown in Figs. 6.11j, 6.11k and 6.11c. From these figures it is clear that the waveforms obtained by the 3 different decoding methods are very close. In fact for sinusoids and Gaussian signals (see Figs. 6.11j and 6.11k) the waveforms reconstructed by Hybrid SCALE decoder and the SCALE-to-LPCM converter are co-incident, indicating that the noise added by the converter is so small that it can be neglected.

#### 6.4.2 The LPCM-to-A-law PCM Compressor (Phase 2)

In Section 6.4.1 above, it has been established that:

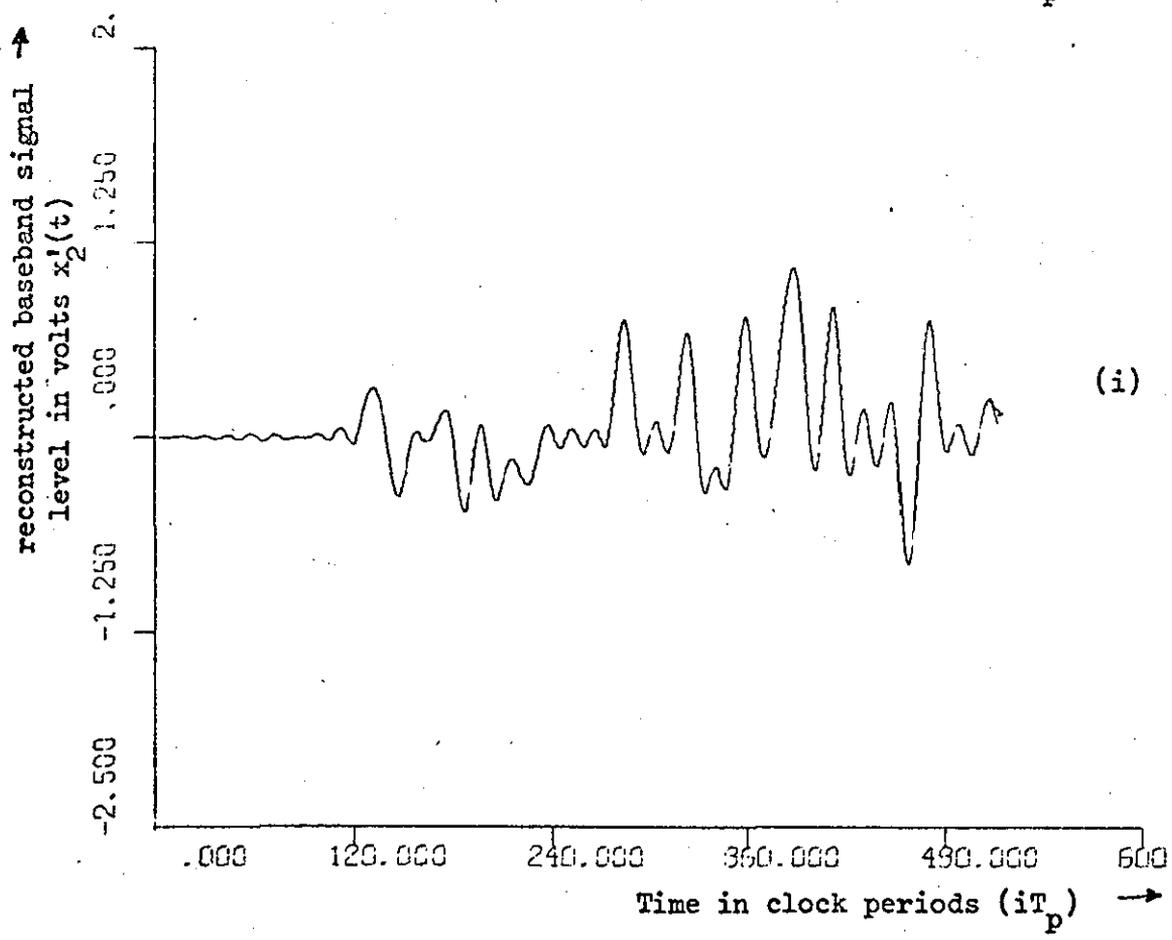
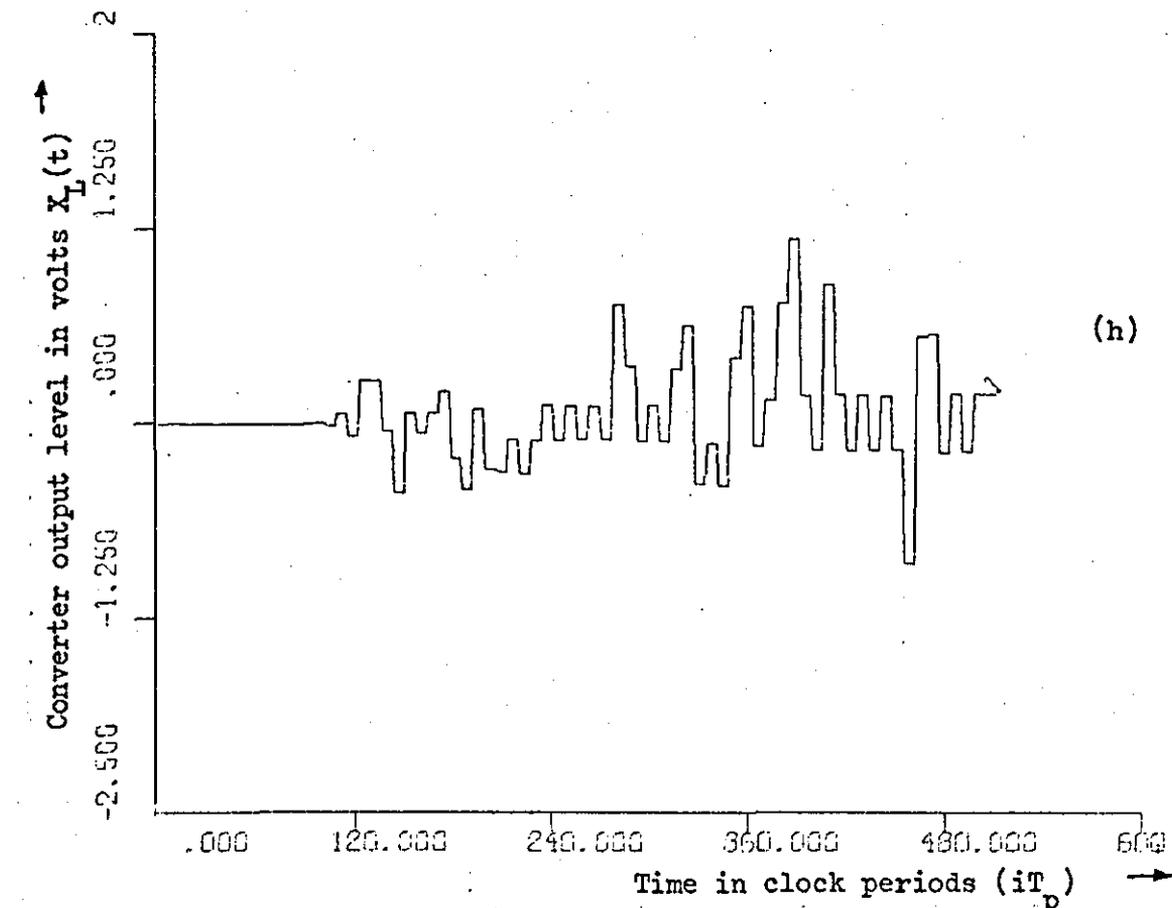
- (a) SCALE-to-linear PCM - using an all-digital converter - is possible
- (b) when the resultant LPCM words are decoded (digital-to-analogue converted) and low-pass filtered, the baseband signal obtained is very close to that obtained from an ordinary hybrid SCALE decoder followed by a filter.

What is still to be shown is that the A-law digital compressor,

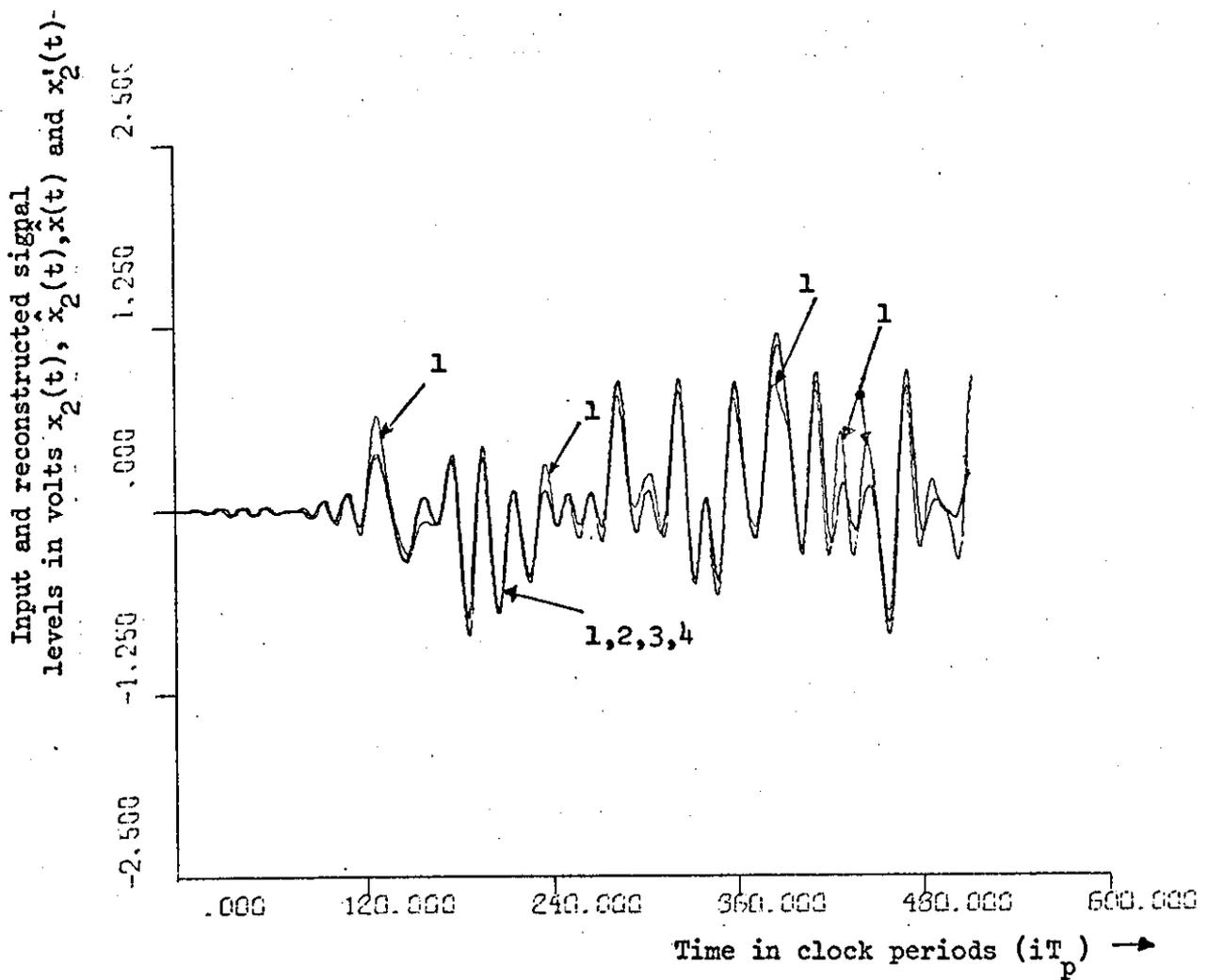


(f) Gaussian distributed noise voltage waveform generated by "GAUSS2".

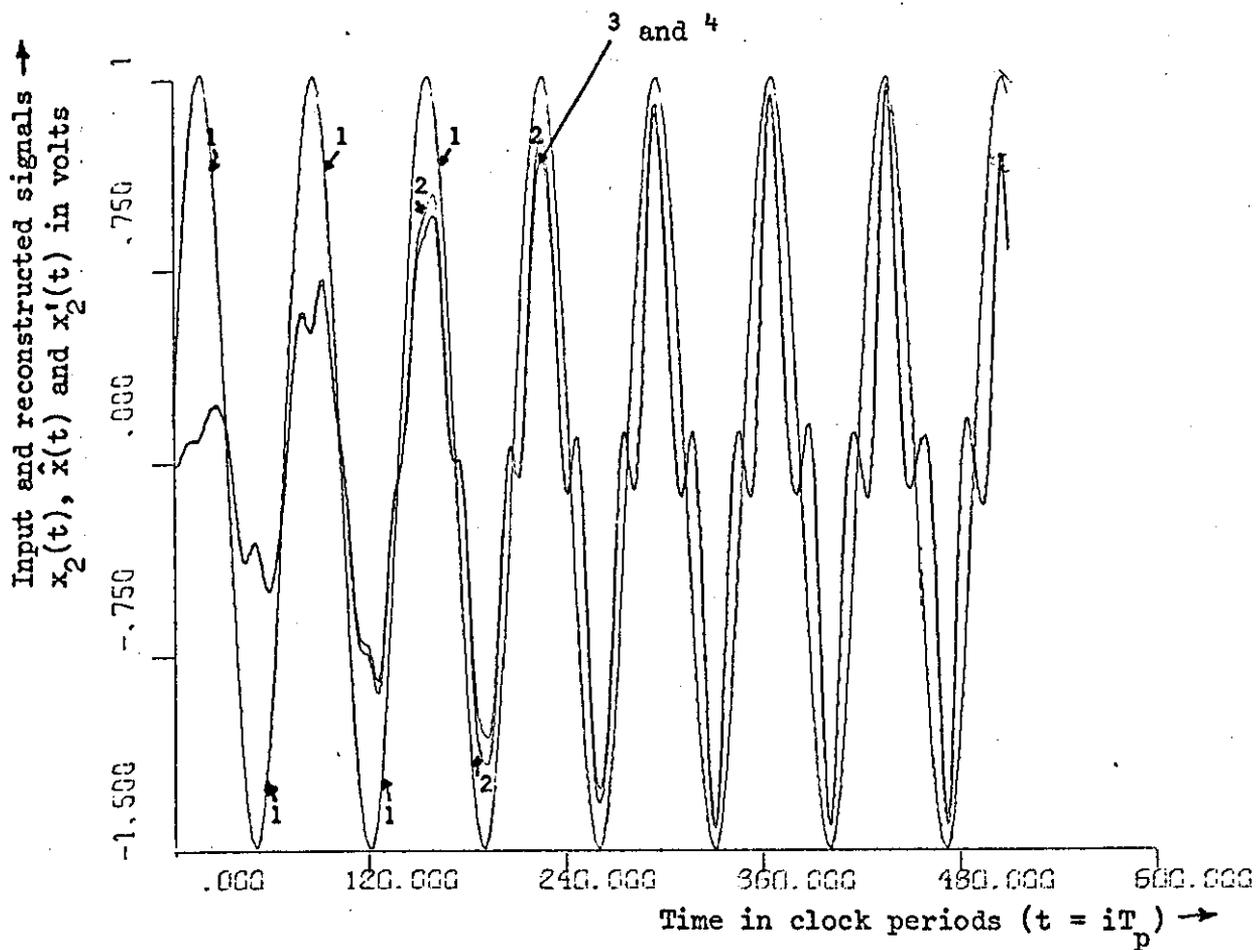
(g) Gaussian noise voltage waveform after passing through a 4 KHz low pass filter.



- (h) SCALE-to-LPCM converter output when the SCALE encoder input signal is the waveform shown in (g) above.
- (i) The waveform of (h) above after filtering.



- (j) Comparison between the baseband band-limited Gaussian waveform seen at the input of the hybrid SCALE encoder (curve 1) and the reconstructed baseband after: (i) one Hybrid SCALE encoding-decoding operation, (ii) one Hybrid SCALE encoding operation followed by one Digital SCALE operation, (iii) one Hybrid SCALE encoding, one DSCALE decoding, and one digital interpolation operations. The curves corresponding to the last 3 cases (curves 2, 3, 4) are so close that they are indistinguishable from one another.



- (k) curve (1) input signal to SCALE encoder  
 curve (2) reconstructed base band signal after one Hybrid SCALE encoding and one Digital SCALE decoding operations followed by one filtering operation.  
 curves (3),(4) Reconstructed base band signals after one Hybrid SCALE encoding-decoding operation and filtering curve (3); and one Hybrid SCALE encoding, one Digital SCALE decoding and one Digital interpolation (linear PCM forming) operations followed by a filtering process, curve (4).

described earlier in this chapter, does have a segmented A-law input-output characteristic, and furthermore that the overall degradation suffered by the baseband signal in passing through the SCALE - to - A-law PCM converter is not severe. To enable these points to be investigated another computer program simulating the system was compiled. The flow chart of this program is shown in Fig. 6.12. The program consists of the following segments (see Fig. 6.12):

(a) Master Segment

After "dimensioning," etc. the master segment reads in from cards a SCALE binary digit, calls the necessary subroutines to perform the required operations on the SCALE channel pattern to obtain the SCALE word  $H_D$ . The linear PCM word  $X_L$  and the compressed A-law PCM word  $X_C$ , and the final PAM sample value  $y$  (at the output of the A-law PCM receiver) are then generated. Another SCALE binary digit is then read in and the cycle repeats until all the SCALE data stored on the card deck are processed. At this point a graph of the digital compressor output as a function of its input level is plotted. This is followed by a second graph showing the voltage levels constructed from the SCALE binary data by:

- (1) The DSCALE decoder
- (2) The SCALE-to-LPCM converter followed by a digital-to-analogue converter and a low pass filter.
- (3) The SCALE-to-A-law PCM converter followed by an A-law PCM receiver.

The SCALE binary data was recorded on punched cards in advance. This was achieved by first simulating a hybrid SCALE encoder whose input was a constant signal of magnitude 2.4 volts for 512 clock periods. The generated binary data at the SCALE encoder output were then punched on cards to be used as described above.

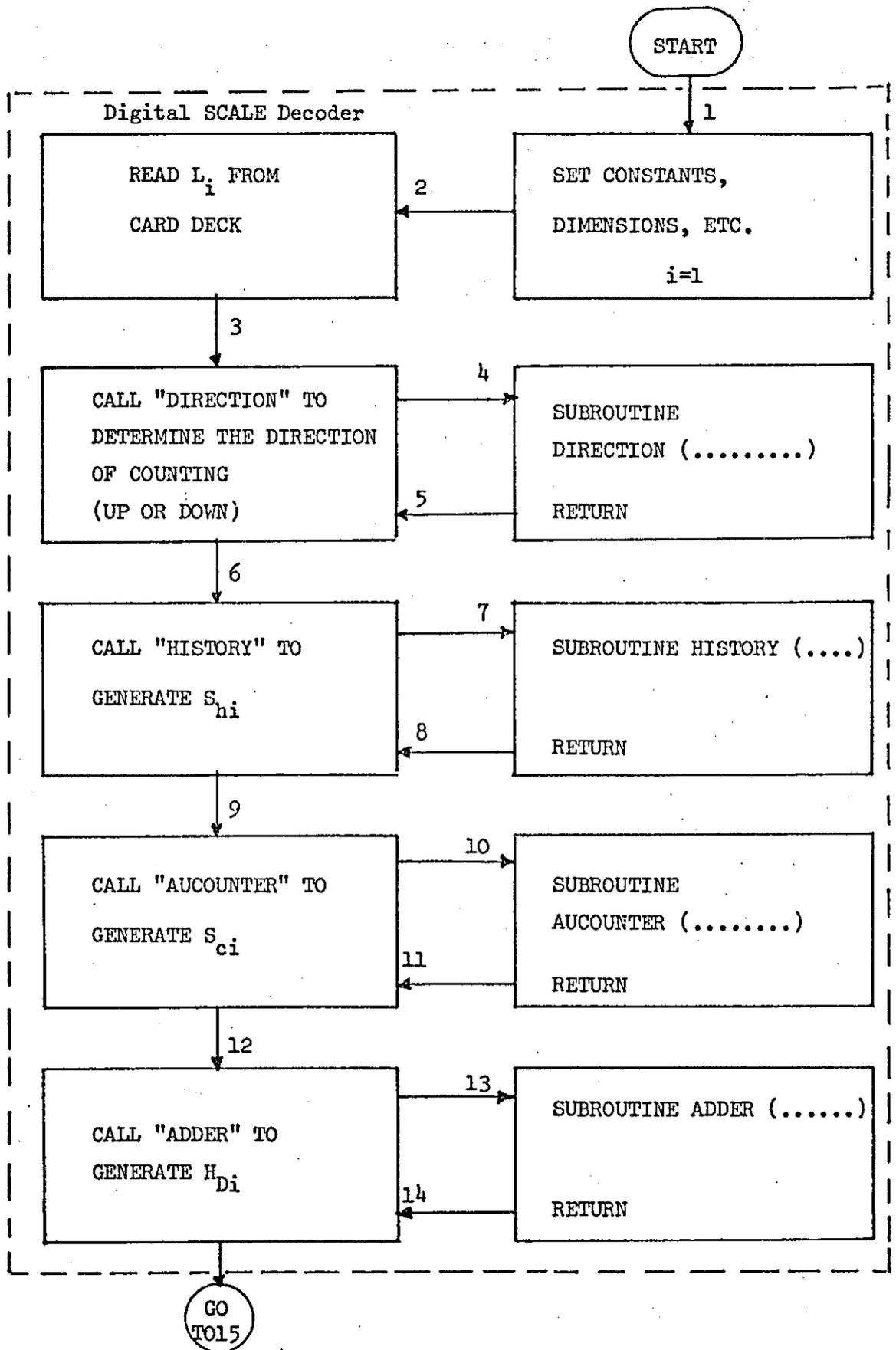
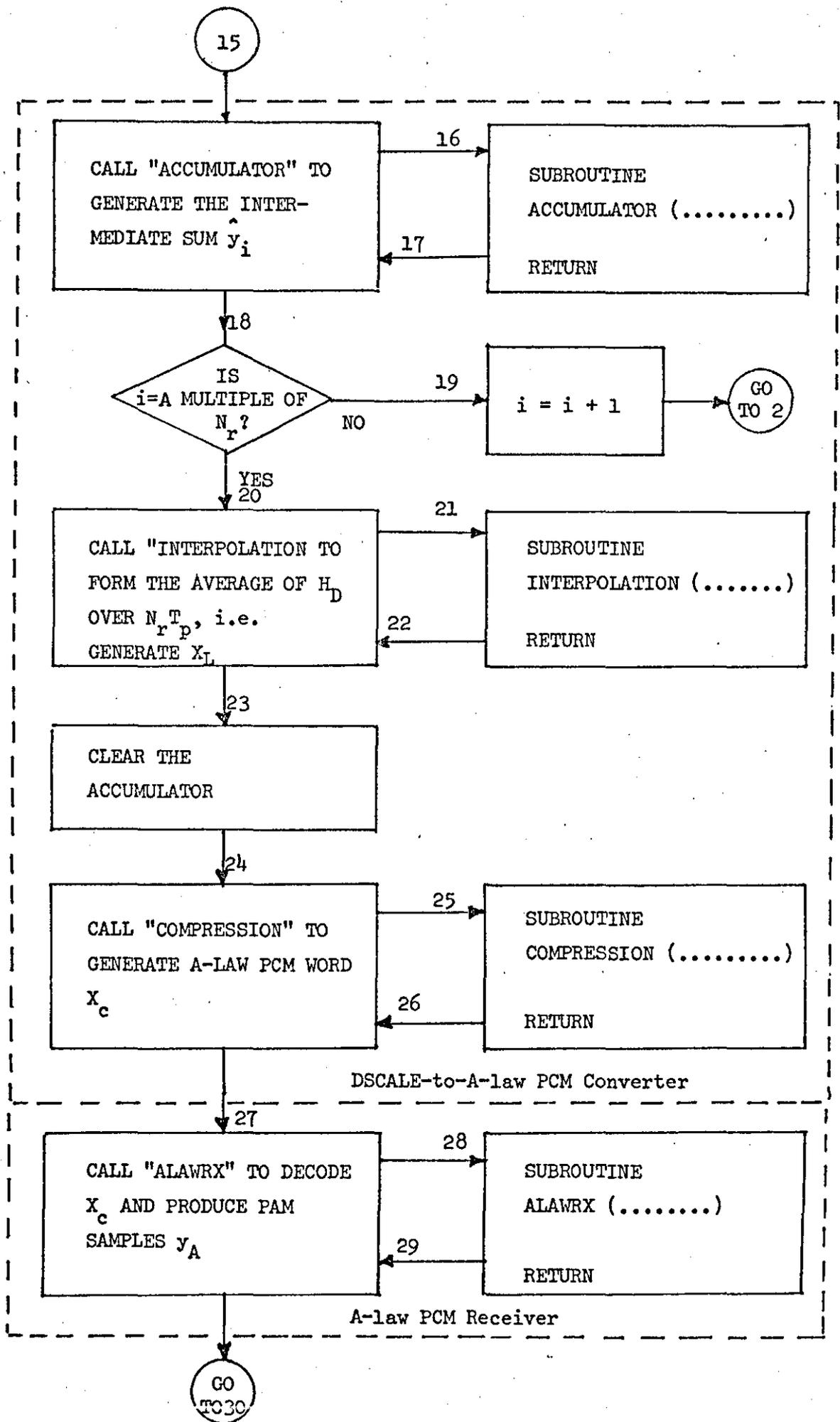
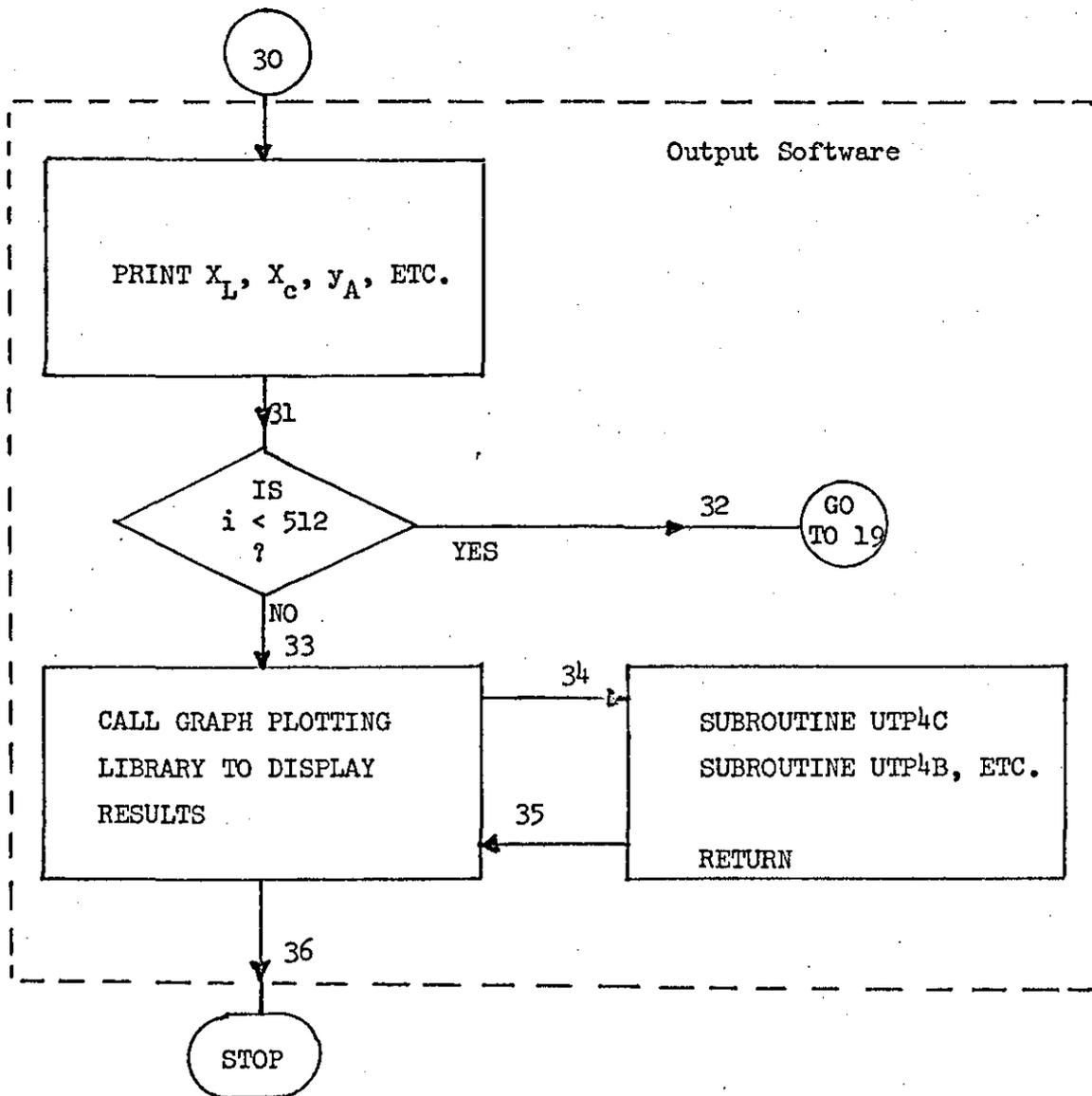


Figure 6.12 Flow chart of the computer program employed for testing the SCALE to LPCM converter





(b) The Digital SCALE decoder

After reading a SCALE digit from the punched cards the master segment calls "SUBROUTINE DIRECTION", "SUBROUTINE HISTORY", "SUBROUTINE AUCOUNTER" and "SUBROUTINE ADDER". These subroutines constitute the digital SCALE decoder discussed in Chapter V. Every  $T_p$  second this produces a DSCALE code word  $H_{D_i}$  which is delivered to the SCALE-to-LPCM converter (see point 15 on Fig. 6.12). Reconstruction of the SCALE feedback signal can be achieved by calling "SUBROUTINE DAC" (see Chapter V) to convert from digital to analogue form.

All the above subroutines have been discussed in detail in Chapter V and are listed in Appendix B.

(c) The SCALE-to-LPCM converter

In the program this is represented by "SUBROUTINE ACCUMULATOR" and "SUBROUTINE INTERPOLATION". The first sums every 8 consecutive SCALE code words while the second averages this sum over 8 clock periods. Between them these two subroutines implement in FORTRAN IV the operation defined by the relation:

$$X_L(j + 7) = \frac{1}{8} \sum_{i=0}^7 H_D(j+i)$$

where  $j = 0, 7, 15, 23, \dots$ ,  $H_D(j+i)$  is the SCALE digital word at the  $(j + i)^{th}$  clock instance and  $X_L(j + 7)$  is the interpolated linear PCM word. When operated upon by "SUBROUTINE DAC",  $X_L$  provides the linear PCM PAM sample constructed from the present and the 7 previous SCALE binary digits. The accumulator is cleared after the completion of calculating every LPCM sample.

"SUBROUTINE ACCUMULATOR" and "SUBROUTINE INTERPOLATION" are listed in Appendix B.

(d) The compressor and the A-law PCM receiver

"SUBROUTINE COMPRESSION" and "SUBROUTINE ALAWRX" were compiled to

represent the A-law PCM compressor and the A-law PCM receiver respectively. The first operates on the LPCM 11-digit words  $X_L(j + 7)$  to produce an A-law compressed word  $X_c(j + 7)$ . The relationship between the input and output of the compressor is given by equation (6.20) and the simulation was performed according to the theory discussed in Section 6.3.4 and Fig. 6.8.

The A-law PCM receiver was simulated to allow the compressed A-law PCM words to be decoded to produce an estimate of the LPCM PAM samples seen at the input of the compressor. The simulated receiver is represented in the program by "SUBROUTINE ALAWRX" and operates as follows:

- (i) It determines the sign of the signal sample from the MSD ( $g_8$ ) of the compressed code word  $\{g_8 g_7 \dots g_1\}$
- (ii) It determines the segment in which the level lies from the next 3 MSD's ( $g_7, g_6, g_5$ ) by performing the operation

$$\hat{J}_s = \sum_{i=1}^3 2^{i-1} g_{i+4}$$

where  $\hat{J}_s$  = the determined segment number

- (iii) It determines the step number within  $\hat{J}_s$  by performing the operation

$$\hat{S}_q = \sum_{i=1}^4 2^{i-1} g_i$$

where  $\hat{S}_q$  is the determined step number

- (iv) It forms the estimated PAM sample by performing the operation

$$\hat{X}_L(\hat{J}_s, \hat{S}_q) = 2^{\hat{J}_s - n_a} \cdot \{ \hat{S}_q + n_a N_t + 0.5 \}$$

as defined by equation (6.20).

"SUBROUTINE COMPRESSION" and "SUBROUTINE ALAWRX" are listed in Appendix B.

The resultant input-out characteristic of the compressor (companding law), plotted from the computer results, is shown in Fig. 6.13a

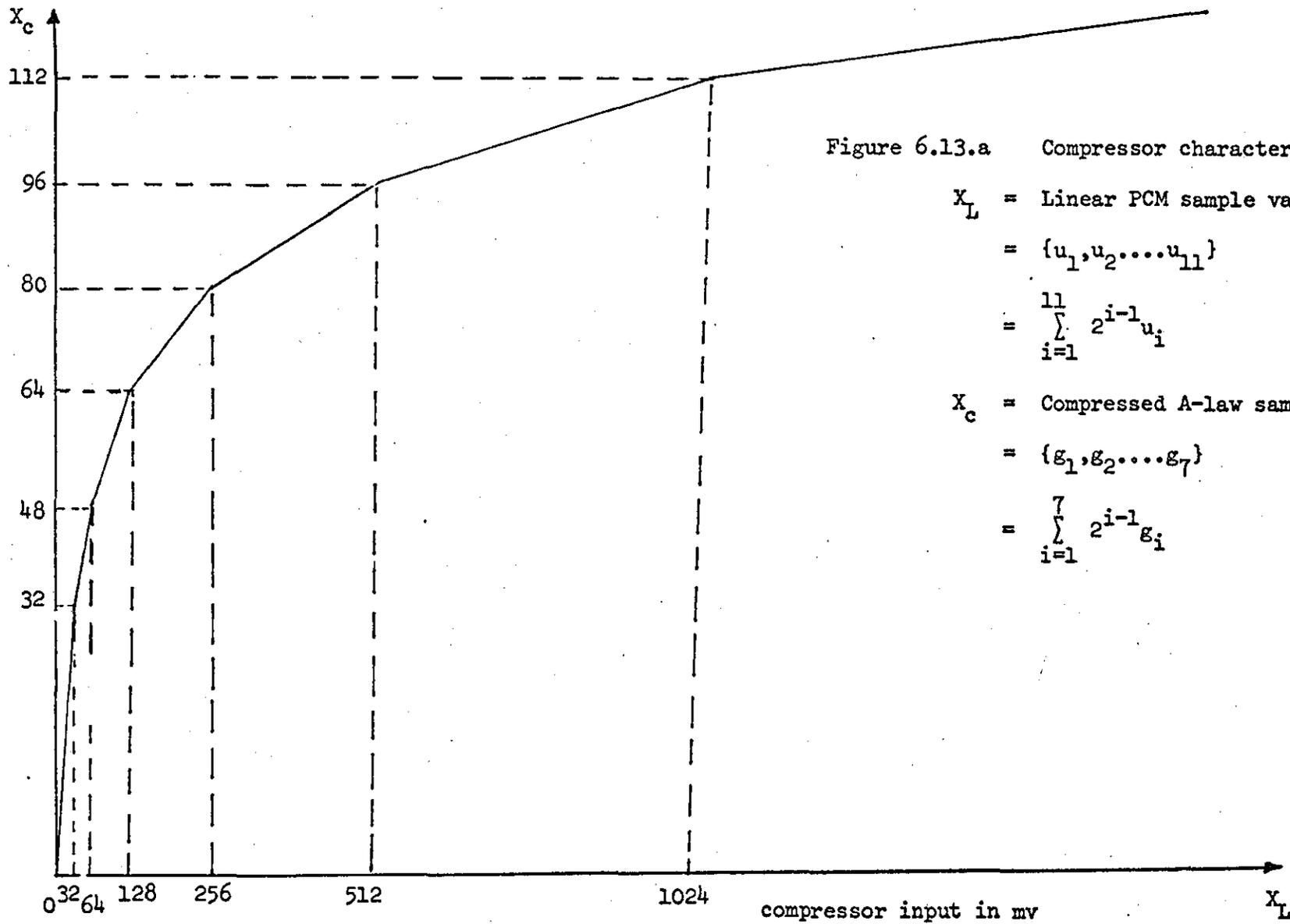


Figure 6.13.a Compressor characteristic

$X_L$  = Linear PCM sample value

=  $\{u_1, u_2 \dots u_{11}\}$

$$= \sum_{i=1}^{11} 2^{i-1} u_i$$

$X_c$  = Compressed A-law sample value

=  $\{g_1, g_2 \dots g_7\}$

$$= \sum_{i=1}^7 2^{i-1} g_i$$

while the inverse (the expander characteristic) is shown in Fig. 6.13b. The simulated compressor and expander (receiver) satisfy the 13-segment A-law recommended by the CCIR ( $A = 86.7$ ).

Figures 6.14a and 6.14b show how the estimation of the SCALE feedback step height at the output of an A-law PCM receiver (after the SCALE to - A-law PCM conversion) compares with that at the output of a linear PCM receiver (after SCALE-to-LPCM conversion), and the reconstructed signal at the output of a digital SCALE decoder.

It is clear from these curves that the signal estimates seen at the LPCM and A-law PCM receivers are in close agreement with that at the SCALE decoder output port. It is however worth noticing that most of the noise is introduced by the compressor/expander circuit and not by the SCALE-to-LPCM converter. The degradation in the system performance as a result of the presence of the SCALE - to - A-law PCM converter is discussed below.

## 6.5 SYSTEM PERFORMANCE

For a given SCALE system the mean square noise voltage  $v_n^2$  is related to the feedback step height  $H^2$  by a constant  $k_n$ , say (see Chapter II). i.e.

$$v_n^2 = k_n \cdot \overline{H^2} \quad H \leq H_{\max} \quad (6.22)$$

(see equation (6.21))

This means that before overload the mean square noise voltage is directly proportional to the mean square of the feedback step height (e.g. if  $\overline{H^2}$  is doubled or halved  $v_n^2$  is also doubled or halved, etc.).

Also, in Chapter V it has been found by experiment (see Fig. 5.10) that for the given digital SCALE decoder operating at 64 Kb/s and employing a 3.3 KHz filter, the maximum signal-to-noise ratio is = 22dB. (= 25dB for 2.4 KHz filter).

But the signal-to-noise ratio is by definition

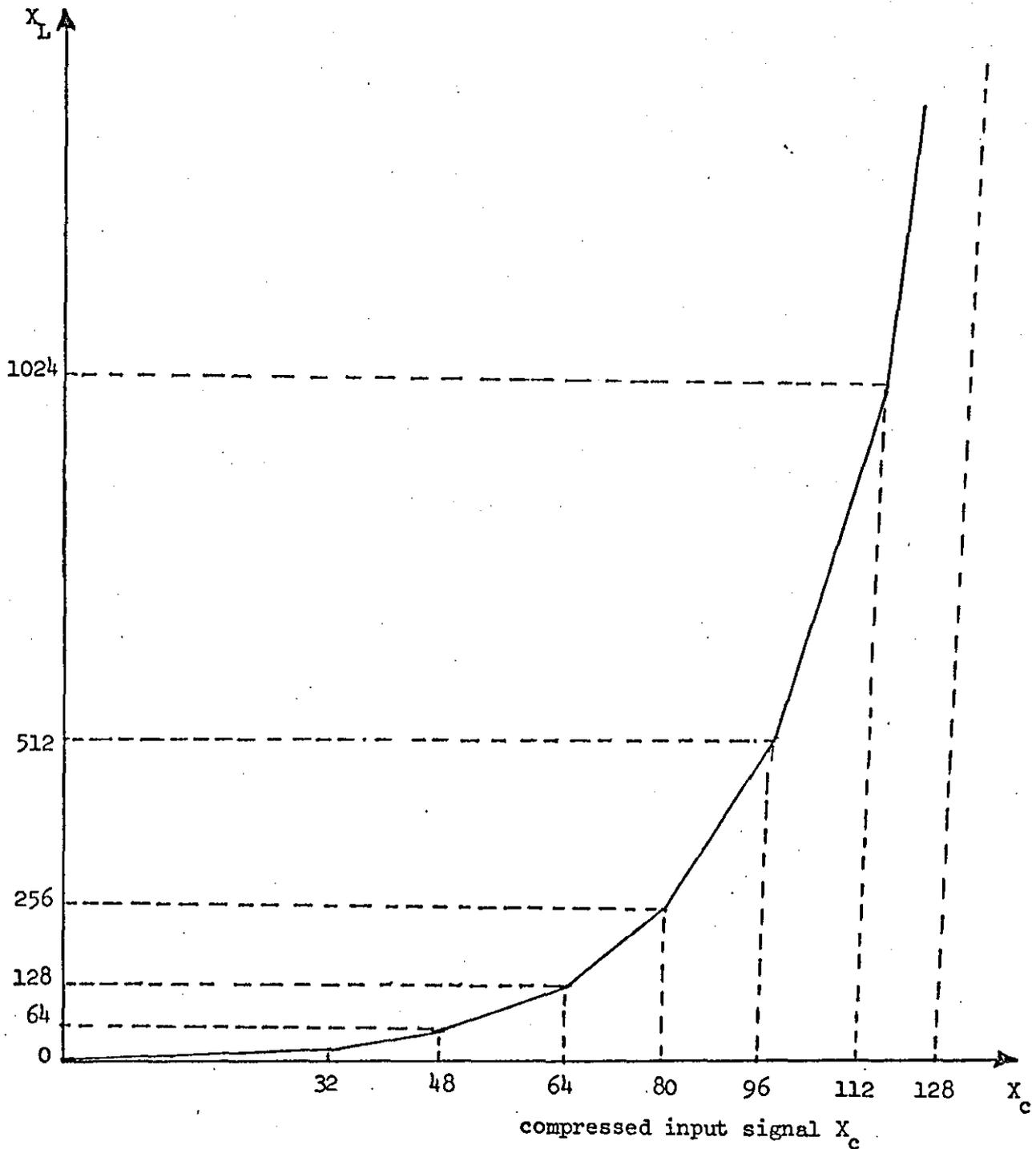


Figure 6.13.b Expander characteristic

$X_c$  = Compressed signal

$$= \{g_1, g_2, \dots, g_7\}$$

$$= \sum_{i=1}^8 2^{i-1} g_i$$

$X_L$  = Expanded signal

$$= \{u_1, u_2, \dots, u_{11}\}$$

$$= \sum_{i=1}^{11} 2^{i-1} u_i$$

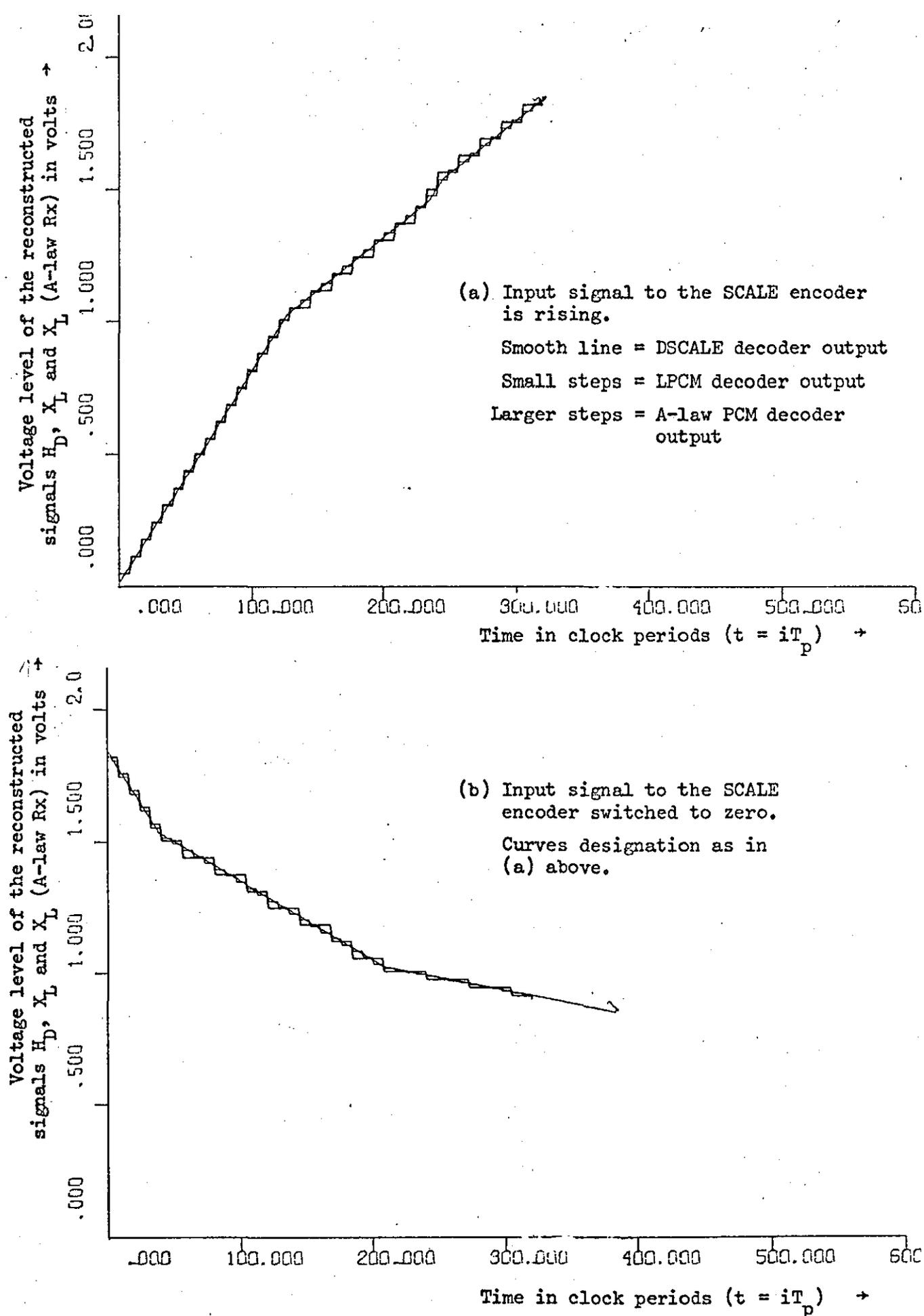


Figure 6.14 Comparison between the outputs from

- (i) DSCALE decoder
- (ii) SCALE-to-PCM converter followed by a LPCM decoder
- (iii) SCALE - to - A-law PCM converter followed by an A-law PCM decoder representing a constant d.c. signal of 2 volts.

$$\text{SNR} = 10 \log \frac{\sigma_x^2}{v_n^2} \quad (6.23)$$

where  $\sigma_x$  is the input mean square voltage.

Now the maximum signal-to-noise ratio occurs when  $\overline{H^2} = \sigma_x^2 \approx 1$  (see Chapter III Section 4). Therefore it follows that

$$(\text{SNR})_{\max} = 10 \log_{10} \frac{1}{v_n^2} = 22\text{dB} \quad (6.24)$$

from which

$$v_n = k_n = (0.00631)(\text{volt})^2 \quad (6.25)$$

This is the maximum value of the mean square noise voltage and it exists when  $\sigma_x^2 = \overline{H^2} \approx 1$ .

Now in the case of the hybrid SCALE, for any values of  $\sigma_x$  and  $H$ , equation (2.2) is true, i.e. the feedback power is the sum of the input power and the noise power generated by the system. This is also true for the case of the DSCALE, i.e.

$$\sigma_x^2 = \overline{H^2} - v_n^2 \quad (6.26)$$

This means that the SCALE feedback signal is the sum of the outputs of two generators superimposed on one another as shown in Fig. 6.15a. Similarly, the Digital interpolator, the A-law PCM digital compressor and the A-law PCM receiver can be represented by two generators: one generator regenerates the exact SCALE PAM sample  $H_D$  with no additive noise, while the other generator generates  $v_{nl}$  - the difference between  $\overline{y_A^2}$ , the A-law PCM receiver output (before filtering), and  $H_D$ , the input to the Digital interpolator in  $(\text{volt})^2$ . See Fig. 6.15b. i.e.

$$v_{nl}^2 = \overline{y_A^2} - \overline{H_D^2} \quad (6.27)$$

This argument leads to the conclusion that the overall signal-to-

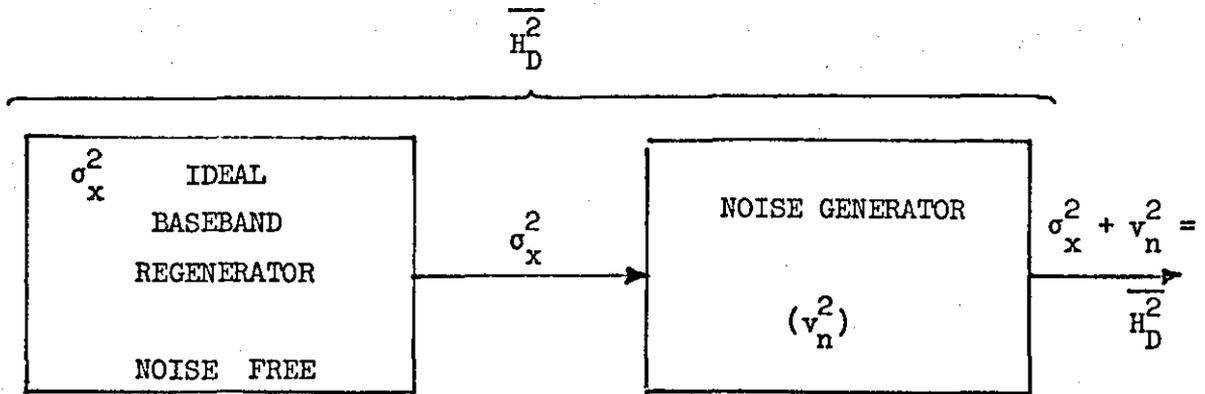


Figure 6.15.a Equivalent circuit of the DSCALE decoder

$H_D$   $\equiv$  DSCALE PAM sample

$\sigma_x$   $\equiv$  Baseband r.m.s. voltage

$v_n^2$   $\equiv$  Quantization noise mean square voltage

= a constant times  $\overline{H_D^2}$

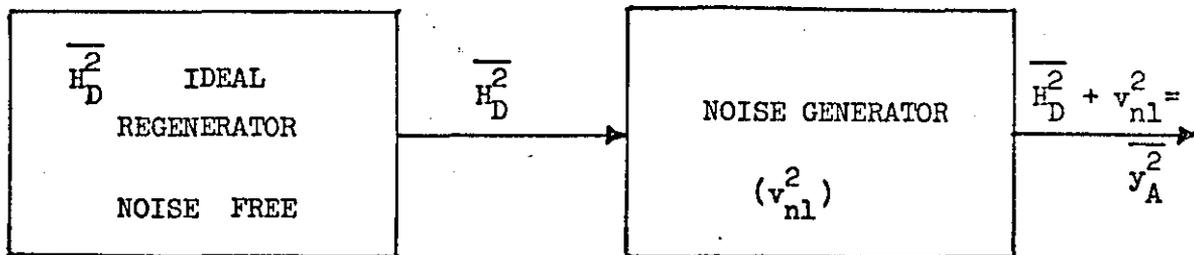


Figure 6.15.b Equivalent of the Digital SCALE to A-law PCM word generator and A-law PCM receiver.

(Word  $\equiv$  PAM sample)

noise ratio  $(SNR)_0$  can be calculated according to the relation

$$(SNR)_0 = 10 \log_{10} \left[ \overline{y_A^2} / (v_n^2 + v_{nl}^2) \right] \quad (6.28)$$

where

$$\overline{y_A^2} = \sum_{i=1}^{M_n} y_A^2(i) \quad (6.29)$$

$$\begin{aligned} v_{nl}^2 &= \overline{y_A^2} - \overline{H_D^2} \\ &= \frac{1}{M_n} \left[ \sum_{i=1}^{M_n} y_A^2(i) - \sum_{i=1}^{M_n} H_D^2(i) \right] \end{aligned} \quad (6.30)$$

and

$$v_n^2 = k_n \cdot \overline{H_D^2},$$

$M_n$  being the number of the A-law receiver PAM samples used in the calculation

With the aid of the computer simulation program the values of  $\overline{H_D^2}$  and  $\overline{y_A^2}$  were computed when the input to the SCALE encoder was a sinusoidal signal and  $M_n = 64$ . From these,  $v_n^2$  and  $v_{nl}^2$  were then evaluated using equations (6.22) and (6.30) respectively. The signal-to-noise ratio at the output of the A-law PCM receiver was calculated using equation (6.28). The resultant curve is shown in Fig. 6.15c. As expected, in the lower region of the output power the overall signal-to-noise ratio  $(SNR)_0$  closely corresponds to that of the DSCALE (without the converter). Above overload both curves of Fig. 6.15c degrade sharply with increasing signal power.

## 6.6 IMPLEMENTATION OF THE SCALE-TO-A-LAW PCM CONVERTER

The implementation of the SCALE-to-A-law PCM converter can be easily achieved using off-the-shelf medium and small integration TTL integrated circuits. A hardware model using Texas TTL 74 series was

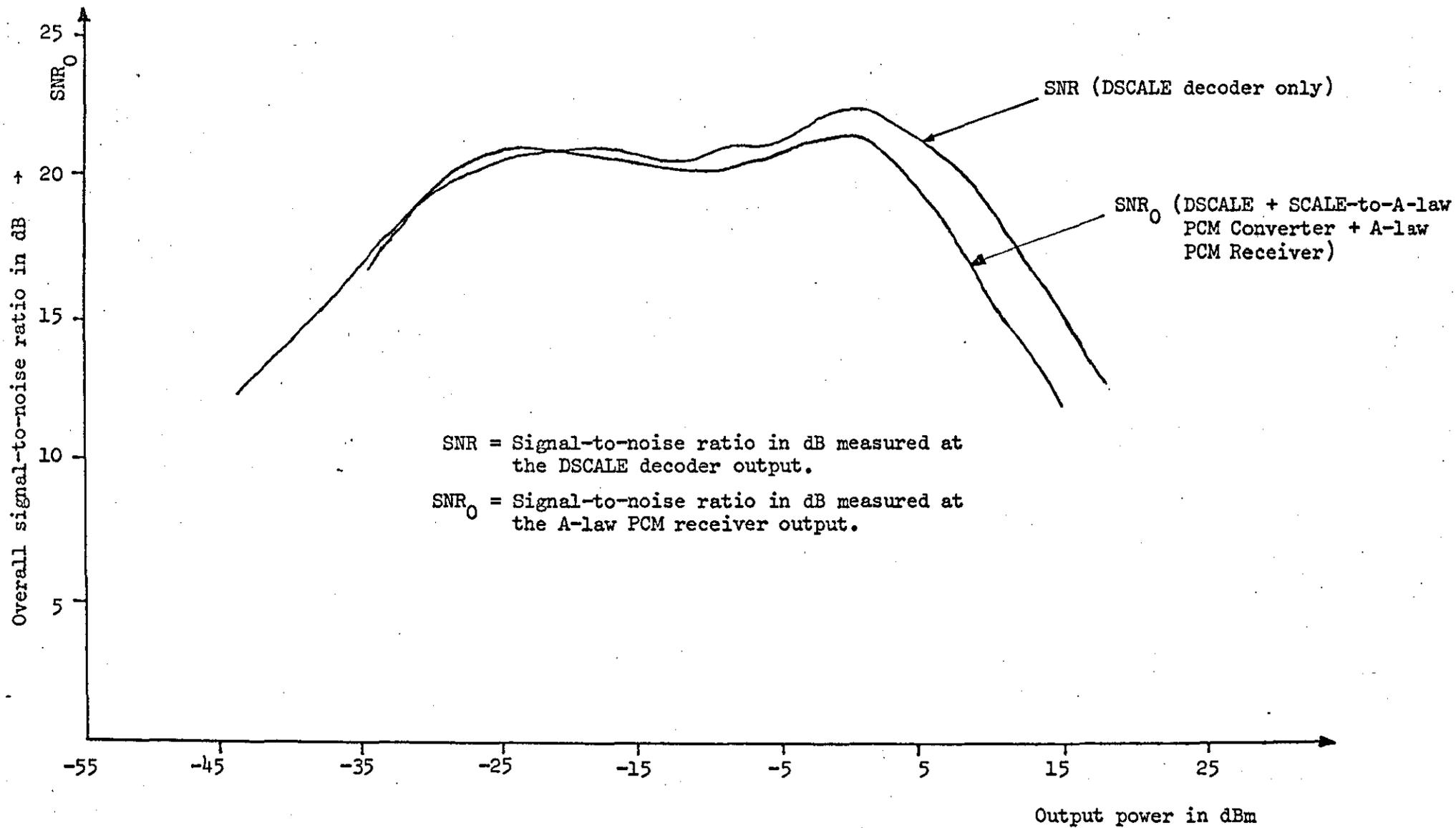


Figure 6.15.c Output signal-to-noise ratio as a function of the baseband input power

designed and will be discussed in detail in the next sections.

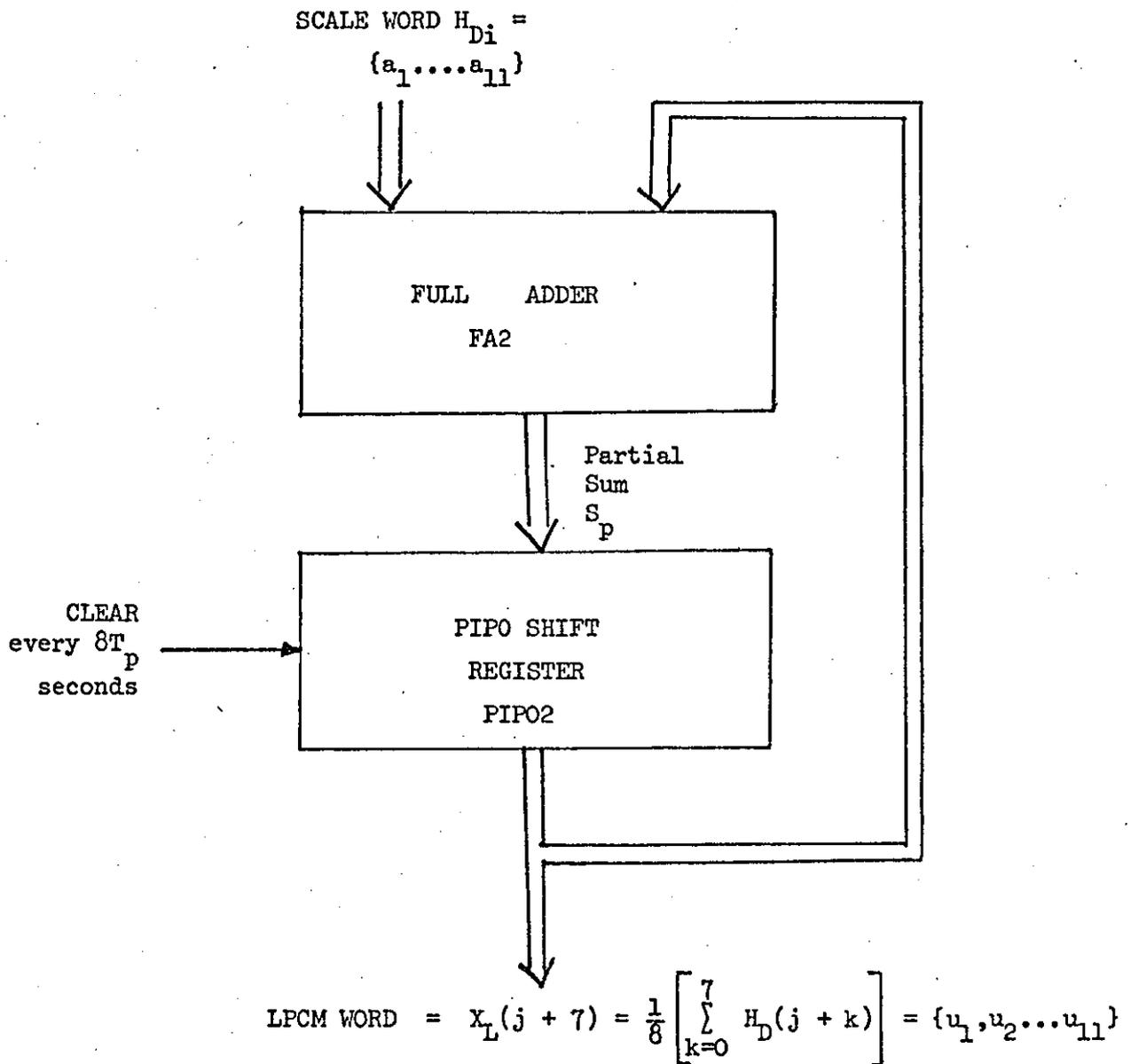
#### 6.6.1 Circuit description of the Digital SCALE decoder

A circuit diagram and the implementation procedure of the Digital SCALE decoder have already been discussed in detail in Chapter V. The circuit employs Texas TTL economically available small and medium integrated circuits (see Section 5.7). The only condition for using this circuit in the SCALE-to-A-law PCM converter is that the input analogue signal to be encoded by the SCALE encoder (and subsequently converted to PCM) shall not exceed 2.048 volts. However if this latter condition cannot be met the circuit of Section 5.7 can easily be modified to cope with any level of the input signal.

#### 6.6.2 Circuit description of the Digital Interpolator

Referring to Fig. 6.2 and 6.1b the digital interpolator (which converts SCALE data to linear PCM words) can be implemented using Texas TTL 74 series small and medium integration integrated circuits which are economically available. A circuit which was designed to achieve this is shown in Figs 6.17. This circuit was designed under the condition that the SCALE encoder overloading does not persist for more than 5 clock periods. However, should this condition be violated by the SCALE encoder input signals the circuit of Fig. 6.17.1a must be modified by the extension of the input and output digital word lengths from 16 to 20 digits. This can be simply achieved by adding one SN7481, one SN74182 and one SN74194 ( $\cong \frac{1}{2}$  SN74198) blocks to the present circuit of 6.17.1a.

The Digital SCALE decoder provides an 11-digit SCALE word  $H_{Di} = \{a_1, a_2, \dots, a_{11}\}$  every  $T_p$  seconds. The full adder FA2 (see Fig.16) forms the sum of the present value of  $H_D$  and the partial sum is stored in the register PIP02. The cycle continues for  $8T_p$  seconds, at the end of which the interpolated linear PCM word is read from the 11 least



$$H_{Di} = \sum_{l=0}^{10} a_l \cdot 2^l = \{a_1, a_2 \dots a_{11}\}$$

$$j = 0, 8, 16, \dots$$

$$u_l = \{0, 1\}$$

$$a_l = \{0, 1\}$$

PIPO  $\equiv$  Parallel-in-Parallel-out

Figure 6.16 Block diagram of the SCALE-to-LPCM converter

Figure 6.17.1a Circuit diagram of the SCALE-to-LPCM converter  
 (See next page)

SCALE sample at the  $i^{\text{th}}$  clock instance is:

$$H_{Di} = \sum_{j=1}^{10} 2^j a_j$$

LPCM sample at the  $m^{\text{th}}$  reading instance is:

$$\begin{aligned} X_L(mN_r) &= \frac{1}{8} \sum_{m=0}^7 H_{Dm} \\ &= \sum_{n=0}^{10} 2^n \cdot u_n \end{aligned}$$

$$u_n = \{0,1\}$$

$$a_j = \{0,1\}$$

$$N_r = 8$$

$$m = 0, 8, 16, 24 \dots$$

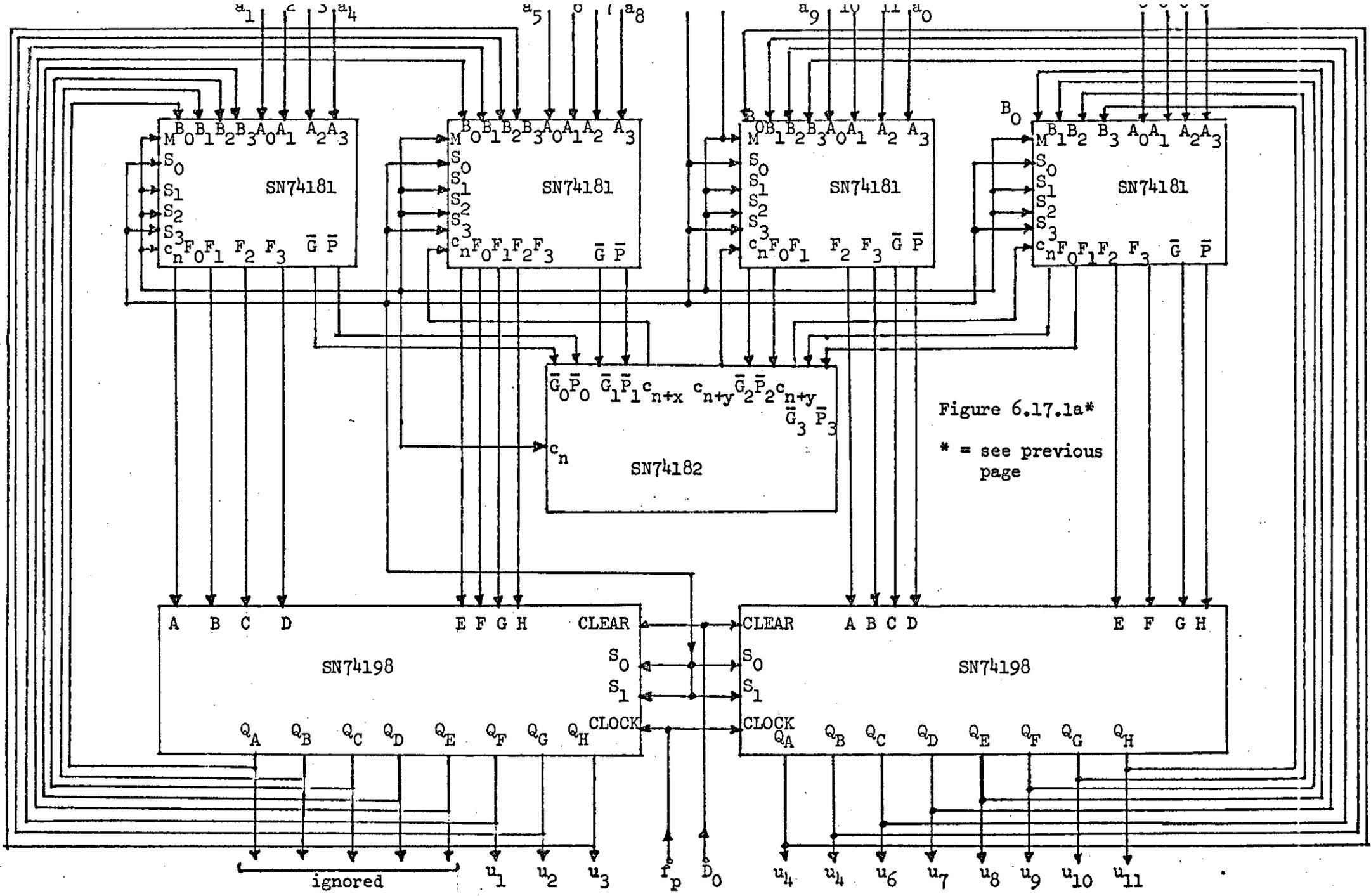


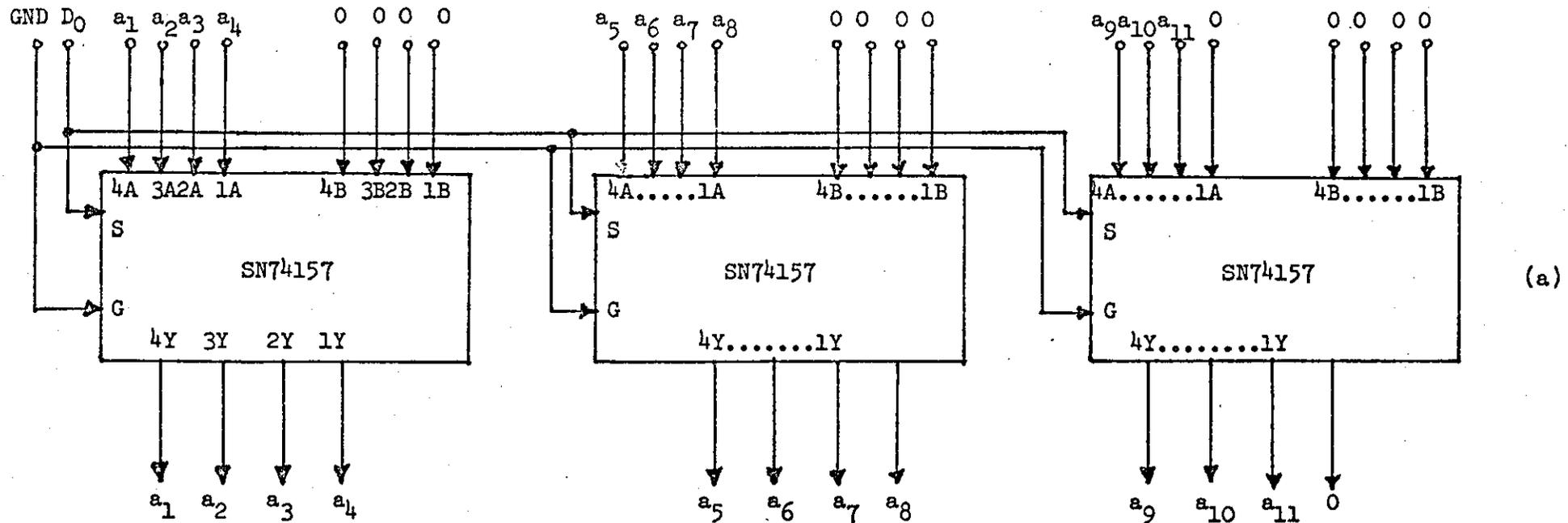
Figure 6.17.1a\*  
 \* = see previous page

significant digit positions at the output of the 16-digit wide shift register PIPO2. The PIPO2 is then cleared and the cycle repeats.

To enable the clearance of PIPO2 at the beginning of every interpolation cycle without affecting the operation of the adder the circuit shown in Fig. 6.17.1b was needed. This circuit accepts SCALE digital words  $H_D$  at a rate  $f_p$  (i.e. every  $T_p$  seconds) and transfers it without delay to the adder FA2. An exception to this is the time during the first  $T_p/2$  seconds of the initial SCALE clock pulse of every interpolation cycle. During this time the  $(a_1, a_2, \dots, a_{11})$  sequence is replaced by zeros and in the same time PIPO2 is cleared. The circuit employs Texas integrated circuit SN74157 quadruple 2-line to 1-line selectors/multiplexers, the details of which are given in Figures 6.17.1b.

The 16-bit adder FA2 employs 4 Texas SN74181 as 4-bit adders in conjunction with the carry look-ahead circuit SN74182. Compared with conventional 4-bit adders SN74181 provides 3 additional outputs. These are: an equality output indicating when the two input words are identical, together with a carry propagate "P" and carry generate "G" (see Figure 6.17.1a). The latter two outputs are important because they allow a very fast carry to be produced. When the G and P outputs are fed to SN74182 a carry look-ahead over the 16-bit is achieved, enabling the addition operation time to be reduced to about 30 n. seconds. Details of SN74181 and SN74182 are given in Figs. 6.17.2 and 6.17.4 respectively.

The output of the adder is loaded into PIPO2 every  $T_p$  seconds and the linear PCM word  $X_L(j + 7)$  is read into the digital compressor at the end of each interpolation cycle (see Fig. 6.17.1a). PIPO2 employs 2 Texas SN74198 8-bit parallel-in-parallel-out shift registers, the details of which are given in Fig. 6.14.3.

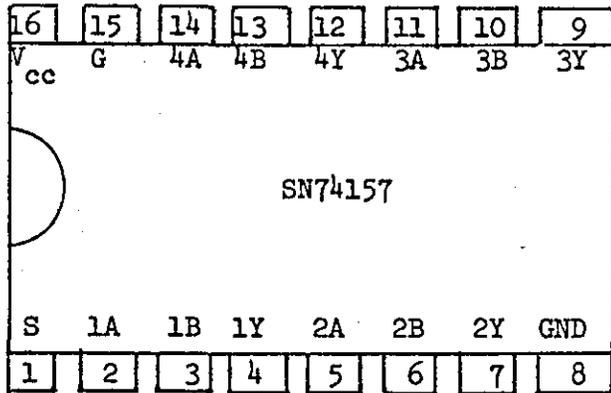


(a)

Figure 6.17.1b Implementation of the SCALE-to-A-law PCM converter - the Digital interpolator (part a)

- (a) circuit diagram of the selection logic
- (b) Pin layout of SN74157\*
- (c) SN74157 - Functional Table\*
- (d) " - Pin Designation\*

\* See next page



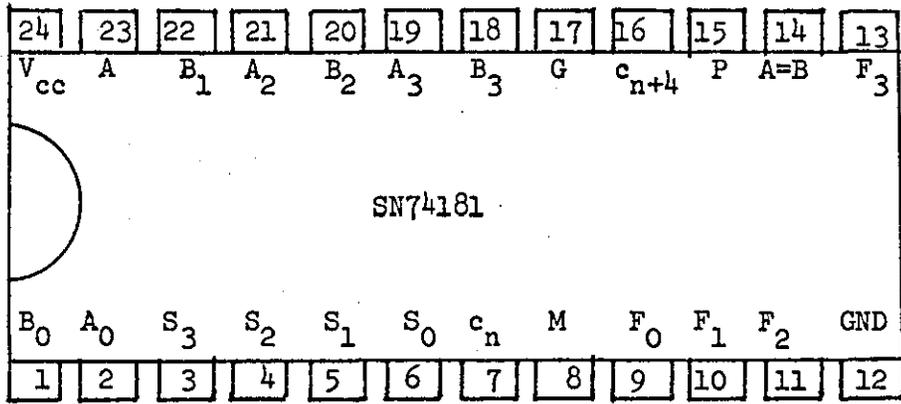
(b)

(c)

I N P U T S				OUTPUTS
G STROBE	S SELECT	A	B	Y
1	X	X	X	0
0	0	0	X	0
0	0	1	X	1
0	1	X	0	0
0	1	X	1	1

Pin Designation	Pin Number	Pin Function
V <sub>cc</sub> , GND	16, 8	Supply volts, Ground
S	1	Select inputs A or B
G	15	Enable selection
1A, 2A, 3A, 4A	2, 5, 11, 14	) Data inputs
1B, 2B, 3B, 4B	3, 6, 10, 13	
1Y, 2Y, 3Y, 4Y	4, 7, 9, 12	Data outputs

(d)



(a)

CONTROL FUNCTIONS					ARITHMETIC OPERATION	
S <sub>3</sub>	S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>	M	c <sub>n</sub> = 1 (carry)	c <sub>n</sub> = 0 (no carry)
1	0	0	1	0	A + B + 1	A + B

(c)

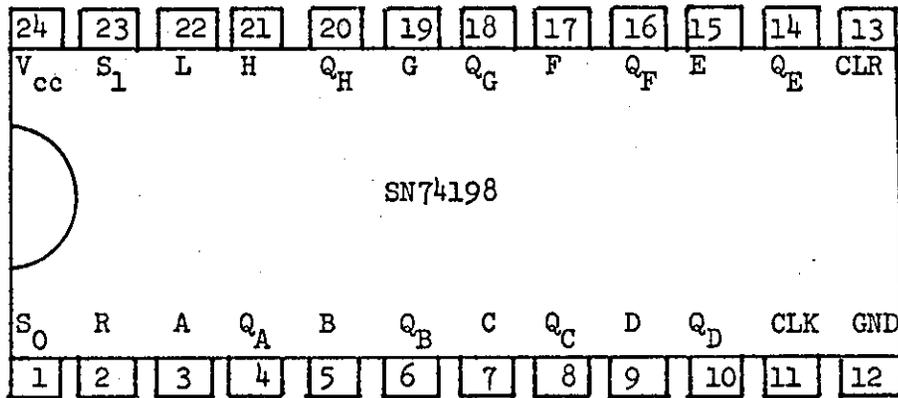
- (a) Pin numbers (top view)
- (b) Pin Designations
- (c) Operation and control signals

Pin Designation	Pin Number	Pin Function
A <sub>0</sub> , A <sub>1</sub> , A <sub>2</sub> , A <sub>3</sub>	2, 23, 21, 19	Word A Inputs
B <sub>0</sub> , B <sub>1</sub> , B <sub>2</sub> , B <sub>3</sub>	1, 22, 20, 18	Word B Inputs
S <sub>0</sub> , S <sub>1</sub> , S <sub>2</sub> , S <sub>3</sub>	6, 5, 4, 3	Control Inputs
c <sub>n</sub>	7	Inverse Carry Input
M	8	Mode Control Input
F <sub>0</sub> , F <sub>1</sub> , F <sub>2</sub> , F <sub>3</sub>	9, 10, 11, 12	Sum Outputs (A + B) sequence
P	15	Carry Propagate Output
c <sub>n+4</sub>	16	Inverse Carry Output
G	17	Carry Generate Output
V <sub>cc</sub> , GND	24, 12	Supply, Ground

(b)

Figure 6.17.2 Description of the integrated circuit Texas SN74181 which can be used as a 4-bit word adder to implement the SCALE-to-LPCM converter

(a)



Pin designation	Pin Number	Pin Function
S <sub>0</sub> , S <sub>1</sub>	1, 23	Mode control
L, R	22, 2	Shift Right, Shift Left (Serial Data)
CLR, CLK	13, 11	Clear, Clock
V <sub>cc</sub> , GND	24, 12	Supply, Ground
A, B, C, D, E, F, G, H	3, 5, 7, 9, 15 ) 17, 19, 21 )	Parallel input Data digits
Q <sub>A</sub> , Q <sub>B</sub> , Q <sub>C</sub> , Q <sub>D</sub>	4, 6, 8, 10 )	Parallel output Data digits
Q <sub>E</sub> , Q <sub>F</sub> , Q <sub>G</sub> , Q <sub>H</sub>	14, 16, 18, 20 )	

(b)

I N P U T S						O U T P U T S	
C L E A R	C L O C K	MODE (CONTROL)		SERIAL OPERATION		PARALLEL OPERATION  A ..... H	Q <sub>A</sub> ..... Q <sub>H</sub>
		S <sub>1</sub>	S <sub>0</sub>	LEFT	RI- GHT		
0	X	X	X	X	X	X	0 ..... 0
1	0	X	X	X	X	X	Q <sub>AO</sub> ..... Q <sub>HO</sub>
1	↑	1	1	X	X	a ..... h	a ..... h PIPO
1	↑	0	1	X	1	X	1 Q <sub>An</sub> ... Q <sub>Gn</sub> } R
1	↑	0	1	X	0	X	0 " ... " } L
1	↑	1	0	1	X	X	Q <sub>Bn</sub> ... Q <sub>Fn</sub> 1 } L
1	↑	1	0	0	X	X	" ... " 0 } L

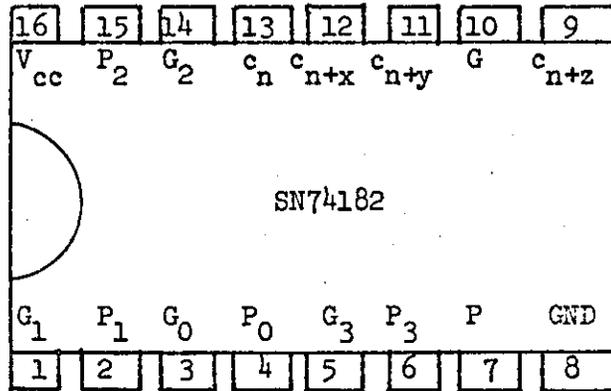
(c)

(a) Pin numbers and designations

(b) Pin allocation Table

(d) Functional (Truth) Table

Figure 6.17.3 Description of Texas integrated circuit SN74198 shift register



(a)

$$c_{n+x} = \bar{G}_0 + \bar{P}_0 c_n$$

$$c_{n+y} = \bar{G}_1 + \bar{P}_1 \bar{G}_0 + \bar{P}_1 \bar{P}_0 c_n$$

$$c_{n+z} = \bar{G}_2 + \bar{P}_2 \bar{G}_1 + \bar{P}_2 \bar{P}_1 \bar{G}_0 + \bar{P}_2 \bar{P}_1 \bar{P}_0 c_n$$

$$\bar{G} = \bar{G}_3 (\bar{P}_3 + \bar{G}_2) \cdot (\bar{P}_3 + \bar{P}_2 + \bar{G}_1) (\bar{P}_3 + \bar{P}_2 + \bar{P}_1 + \bar{G}_0)$$

$$\bar{P} = \bar{P}_3 \bar{P}_2 \bar{P}_1 \bar{P}_0$$

(c)

Pin Designation	Pin Number	Pin Function
$G_0, G_1, G_2, G_3$	3, 1, 14, 5	Active-low carry generate inputs
$P_0, P_1, P_2, P_3$	4, 2, 15, 6	Active-low carry propagate inputs
$c_n$	13	Carry input
$c_{n+x}, c_{n+y}, c_{n+z}$	12, 11, 9	Carry outputs
$G$	10	Active-low carry generate output
$P$	7	Active-low carry propagate output
$V_{cc}, GND$	16, 8	Supply, Ground

(b)

Figure 6.17.4 Description of Texas integrated circuit SN74182 Look-ahead carry generator

- (a) Pin numbers layout
- (b) Pin designation table
- (c) Positive logic Boolean equations

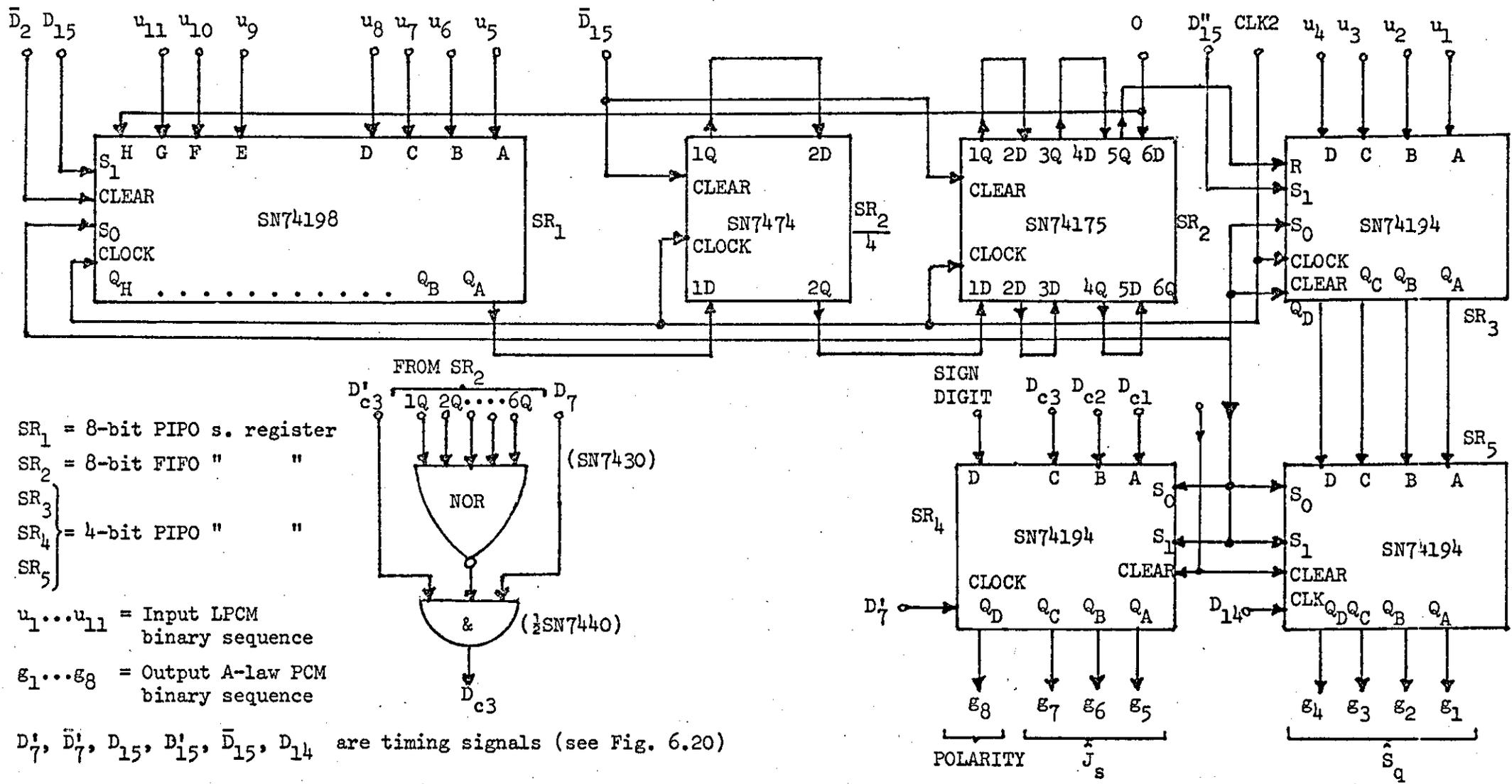
### 6.6.3 Circuit description of the A-law PCM Digital compressor

The A-law PCM digital compressor accepts the linear PCM words  $X_L(j + 7) = \{u_1, u_2, \dots, u_{11}\}$  every  $8T_p$  seconds and produces compressed A-law PCM words,  $X_c(j + 7) = X_c(J_s, S_q)$ , at the same rate. A block diagram of the compressor is shown in Fig. 6.8. The circuit which was designed to perform this function is given in Fig. 6.18.1.

The linear PCM word  $(u_1, u_2, \dots, u_{11})$  is loaded into a 12-bit wide shift register comprising  $SR_1$  and  $SR_3$ , as shown in Fig. 6.18.1. At the same time the previous compressed code word  $X_c(j + 7)$  is read out. When this has been completed a binary counter UBC (see Figs. 6.8 and 6.19.1) starts counting up under the control of clock 2. This clock runs at a frequency which is  $2f_p$ . In the mean time the content of  $SR_3$  is kept unchanged while the content of  $SR_1$  is shifted to the right into  $SR_2$ , under the control of clock 2. When this has been done a second binary counter DBC (see Figs. 6.8 and 6.19.1) monitors the binary digit in position 1 of  $SR_1$ . Whenever this digit is a logical one DBC resets itself to all ones (1111); otherwise it counts down under the control of clock 2 until its content reaches (0001). However, if the content of  $SR_2$  is zero when the content of UBC is 7, the compressed signal must be in the 0<sup>th</sup> segment and consequently the content of DBC is set to (0000). Furthermore, the contents of DBC, irrespective of its value at this time, is loaded into  $SR_4$  and represents the segment number  $J_s$ .

The polarity of the PAM sample represented by  $X_c(j + 7)$  is  $L(j + 7)$ . This can be loaded with  $(u_1, u_2, \dots, u_n)$  in position 8 of  $SR_1$  and shifted with  $(u_1, u_2, \dots, u_8)$ . In this case the polarity digit appears in position 1 of  $SR_1$  when the content of UBC is 7 and therefore can be loaded with the  $J_s$  sequence  $(D_{c1}, \dots, D_{c3})$  into  $SR_4$  just before  $SR_1$  is cleared by  $\bar{D}_7$ .

As the content of UBC gets larger than 7 the DBC can no longer be set to (1111), since  $SR_1$  has been cleared by  $D_7$ , and consequently it



SR<sub>1</sub> = 8-bit PIPO s. register  
 SR<sub>2</sub> = 8-bit FIFO " "  
 SR<sub>3</sub> }  
 SR<sub>4</sub> } = 4-bit PIPO " "  
 SR<sub>5</sub> }

$u_1 \dots u_{11}$  = Input LPCM binary sequence  
 $\epsilon_1 \dots \epsilon_8$  = Output A-law PCM binary sequence

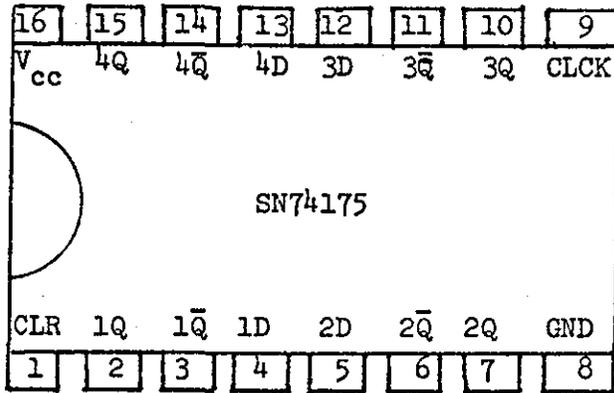
$D'_7$ ,  $\bar{D}'_7$ ,  $D_{15}$ ,  $D'_{15}$ ,  $\bar{D}_{15}$ ,  $D_{14}$  are timing signals (see Fig. 6.20)

Figure 6.18.1 Implementation of the A-law PCM compressor

(the DBC) counts down until its content is reduced to (0001). While DBC is counting down the contents of  $SR_2$  and  $SR_3$  are shifted from Right to Left under the control of clock 2.

When the content of UBC reaches 14 (1110) the contents of  $SR_3$  represent the step number  $\hat{S}_q$  within a segment  $\hat{J}_s$ . This is now loaded into  $SR_5$ . When the count reaches 15 (1111)  $SR_2$  is cleared,  $SR_1$  is loaded and  $X_c(j + 7) = \{g_1, g_2, \dots, g_7\}$  together with the sign digit  $g_8$  are read out into the serializer  $SR_6$  (see Fig. 6.8) for transmission to the remote A-law PCM decoder or for further PCM digital processing.

The integrated circuits employed in the compressor are Texas SN74198 and SN74194 8-bit and 4-bit parallel-in-parallel-out shift registers, SN74175 6-bit serial-in-serial-out shift register, SN7474 2-bit D-flip-flops, SN74193 synchronous up-down 4-bit binary counters, SN74140 Dual 4-input NAND gates, SN7404 Hexinverters and SN7430 8-input NAND gates. The details of these integrated circuits and their operation are given in Figures 6.17.2, 6.17.3, 6.17.4, 6.18.2, 6.18.3, 6.18.4, 6.19.2, 6.20.2 and 6.20.3. Further details are obtainable from the manufacturer's literature<sup>(88,89)</sup>.



(a)

Inputs			Outputs	
Clear	Clock	D(Input)	Q	$\bar{Q}$
0	X	X	0	1
1	↑	1	1	0
1	↑	0	0	1
1	0	X	$Q_0$	$\bar{Q}_0$

Pin Designation	Pin Number	Function
$V_{cc}$	16	Supply voltage
GND	8	Ground
CLR	1	Clear
CLCK	9	Clock input
1D, 2D, 3D, 4D	4, 5, 12, 13	Inputs to individual register stages
1Q, 2Q, 3Q, 4Q	1, 7, 10, 15	Outputs
$1\bar{Q}$ , $2\bar{Q}$ , $3\bar{Q}$ , $4\bar{Q}$	3, 6, 11, 14	Complemented outputs

(b)

(a) Pin layout

(b) Pin designation and allocation

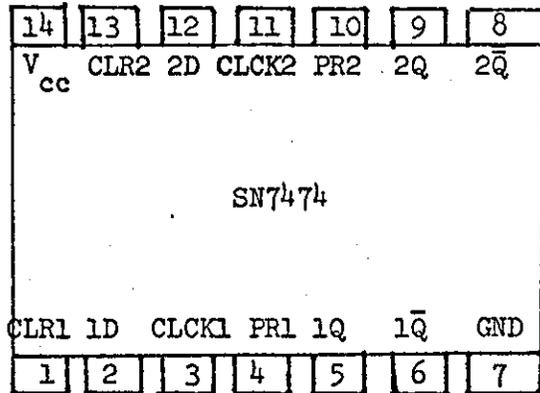
(c) Truth Table (for each stage of the register)

X = don't care

↑ = transition from low to high level (0 to 1)

 $Q_0$  = the level of Q before the indicated steady state input conditions were established

Figure 6.18.2 Description of Texas integrated circuit SN74175 serial-in-serial-out 6-bit shift register



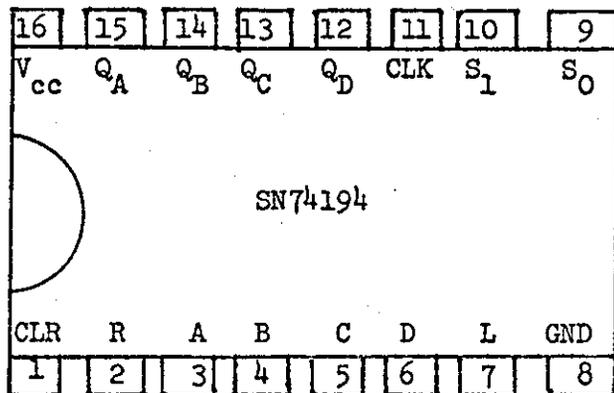
(a)

Pin Designation	Pin Number	Function
V <sub>cc</sub>	14	Supply voltage
GND	7	Ground
CLCK1, CLCK2	3, 11	Clock inputs
CLR1, CLR2	1, 13	Clear inputs
PR1, PR2	4, 10	Preset inputs
1D, 2D	2, 12	D-inputs to individual stages
1Q, 2Q	5, 9	Outputs
1 $\bar{Q}$ , 2 $\bar{Q}$	6, 8	

(b)

Figure 6.18.3 Description of Texas integrated circuit SN7474 2-bit Shift Register (D-Flip-Flops)

- (a) Pin layout
- (b) Pin designation



(a)

(c) FUNCTION TABLE IDENTICAL TO THAT FOR SN74198 (see Fig. 6.16)

Pin Designation	Pin Number	Pin Function
V <sub>cc</sub> , GND	16, 8	Supply, Ground
S <sub>0</sub> , S <sub>1</sub>	9, 10	Control (mode)
CLK, CLR	11, 1	Clock, Clear
L, R	7, 2	Shift Left, Shift Right, Serial data inputs
A, B, C, D	3, 4, 5, 6	Parallel Data inputs
Q <sub>A</sub> , Q <sub>B</sub> , Q <sub>C</sub> , Q <sub>D</sub>	15, 14, 13, 12	Parallel Data outputs

(b)

Figure 6.18.4 Description of Texas integrated circuit SN74194 Shift Register (4-bit wide)

- (a) Pin layout
- (b) Pin designation and allocation
- (c) Functional (Truth) Table.

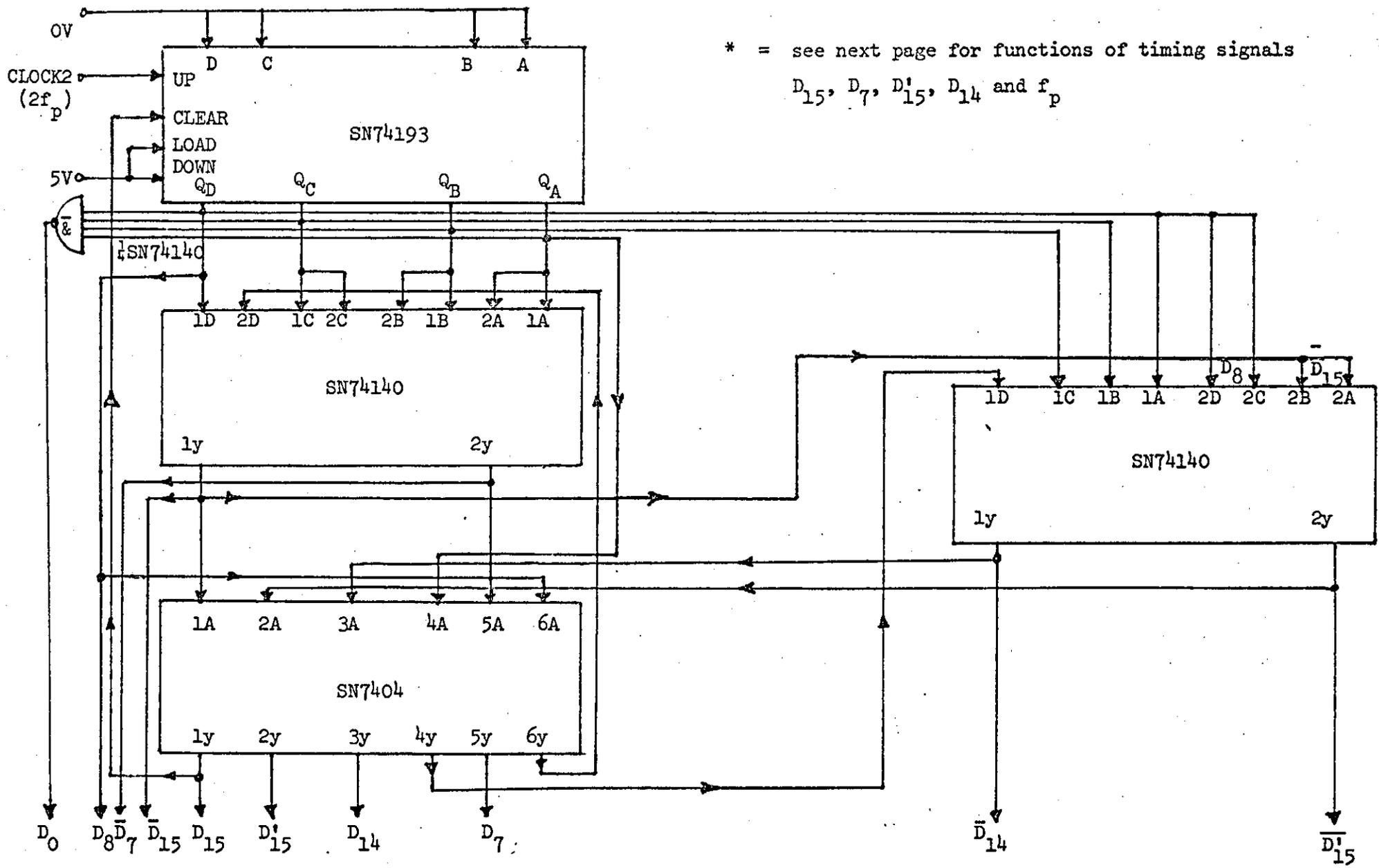


Figure 6.19.1 Implementation of the SCALE-to-A-law PCM converter (part 2) - the timing circuit.\*

$D_{15}$  = 1 LOAD INPUT DATA ( $u_1 \dots u_{11}$ ) IN  $SR_1$  AND  $SR_3$  AND  
READ OUTPUT DATA ( $g_1 \dots g_8$ ) FROM  $SR_4$  AND  $SR_5$  AND  
CLEAR  $SR_2$ .

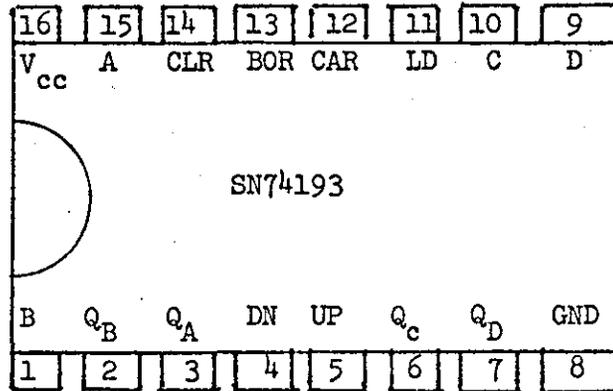
$D_7$  = 1 LOAD CHARACTERISTIC SIGN DIGITS INTO  $SR_4$  AND  
CLEAR  $SR_1$ .

$D'_{15}$  = 1 SHIFT CONTENTS OF  $SR_2$  AND  $SR_3$  TO THE RIGHT UNDER THE  
CONTROL OF THE CLOCK.

$D_{14}$  = 1 LOAD MANTISSA DIGITS FROM  $SR_3$  INTO  $SR_5$ .

$f_p$  = SCALE clock frequency (64 Kb/s)

(b)



(a)

Pin Designation	Pin Number	Pin Function
V <sub>cc</sub>	16	Supply voltage
GND	8	Ground
UP	5	Count up
DN	4	Count down
LD	11	Load data
CAR	12	Carry output
BOR	13	Borrow output
A, B, C, D	15, 1, 9, 10	Data inputs
Q <sub>A</sub> , Q <sub>B</sub> , Q <sub>C</sub> , Q <sub>D</sub>	3, 2, 6, 7	Data outputs (counter contents)

(a) Pin layout

(b) Pin designation and functions

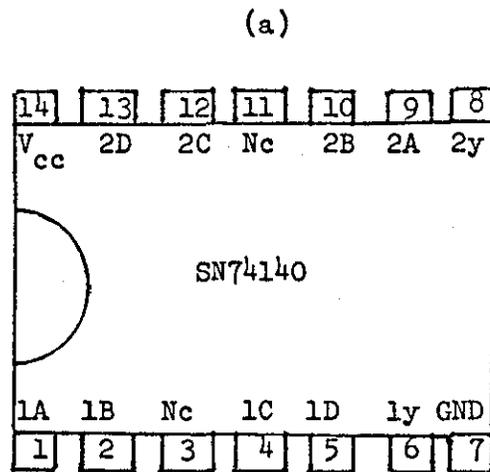
(c) Functional Table

X = don't care

INPUTS				OUTPUTS			
CLEAR	LOAD	UP	DOWN	D	C	B	A
1	X	X	X	X	X	X	X
0	0	X	X	d	c	b	a
0	1	PULSED	1	d	c	b	a
0	1	1	PULSED	d	c	b	a

(c)

Figure 6.19.2 Implementation of the timing circuit using Texas integrated circuit SN74193 binary synchronous 4-bit Up-down counter.



(c)  $y = \overline{ABCD}$

(b)

Pin Designation	Pin Number	Pin Functions
V <sub>cc</sub>	14	Supply voltage
GND	7	Ground
Nc	3, 11	Not connected
1A, 1B, 1C, 1D	1, 2, 4, 5	Data inputs
2A, 2B, 2C, 2D	9, 10, 12, 13	Data inputs
1y, 2y	6, 8	Data outputs

Figure 6.19.3 Implementation of the timing circuit using Texas integrated circuit SN74140 Dual 4-input positive NAND gates.

- (a) Pin layout
- (b) Pin designation
- (c) Logical equation

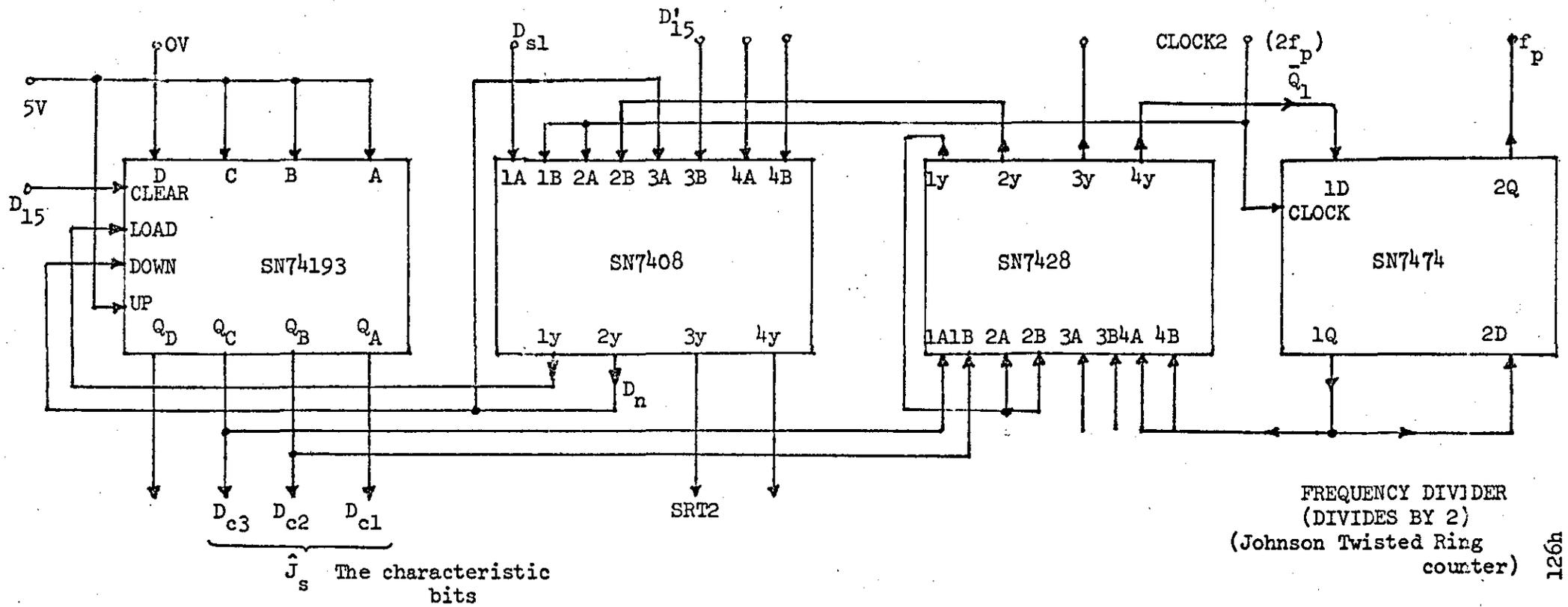


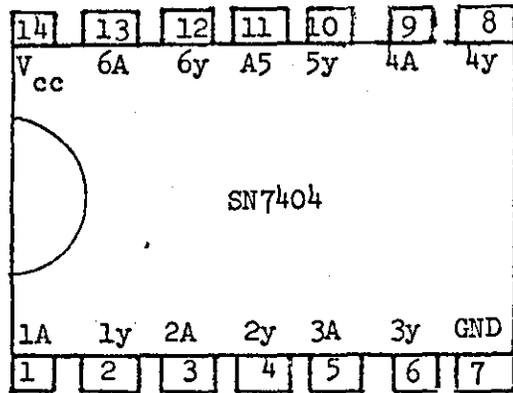
Figure 6.20.1 Implementation of the SCALE-to-A-law PCM converter (port 3) - the  $J_s$ -search and clock frequency divider (divides by 2).

$D_{15}$ ,  $D'_{15}$ ,  $D_{s1}$ , SRT2 are control (timing signals)

$f_p$  is the SCALE clock frequency

$2f_p = \text{clock 2} = \text{twice } f_p \text{ Hz}$

$D_{c1} \dots D_{c3} = \text{Binary sequence representing A-law segment number.}$



(a)

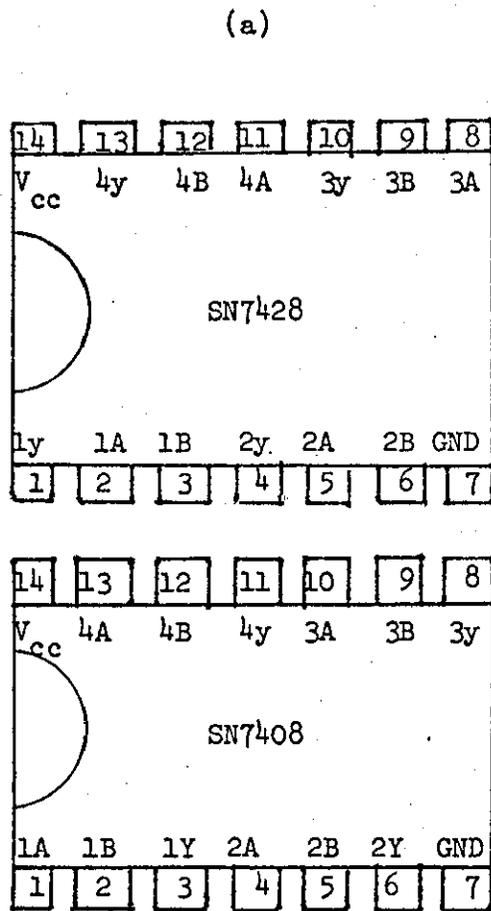
(c)  $y = \bar{A}$

Pin Designation	Pin Number	Function
V <sub>cc</sub>	14	Supply voltage
GND	7	Ground
1A, 2A, 3A, 4A 5A, 6A	1, 3, 5, 9 11, 13	Data inputs
1y, 2y, 3y, 4y, 5y, 6y	2, 4, 6, 8 10, 12	Data outputs

(b)

Figure 6.20.2 Implementation of the timing circuit using Texas integrated circuit SN7404 Hex inverter

- (a) Pin layout
- (b) Pin designation and functions
- (c) Logical equation



(c)

$$y = \overline{A + B}$$

(d)

$$y = A.B$$

(b)

Pin Designation	Pin Number	Pin Functions
V <sub>cc</sub>	14	Supply voltage
GND	7	Ground
1A, 2A, 3A, 4A	2, 5, 8, 11	Data inputs
1B, 2B, 3B, 4B	3, 6, 9, 12	" "
1y, 2y, 3y, 4y	1, 4, 10, 13	Data outputs

Figure 6.20.3 Implementation of the  $\hat{J}_s$  search circuit using Texas integrated circuits SN7408 (quadruple 2-input positive NAND gates) and SN7428 (quadruple 2-input NOR gates).

- (a) Pin layouts
- (b) Pin designations and functions
- (c) Boolean equations of SN7428
- (d) " " " SN7408

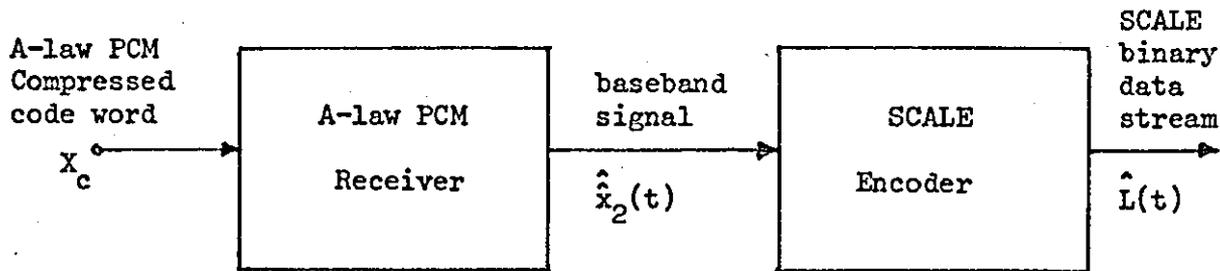
## CHAPTER VII

### A-LAW PCM - TO - SCALE CONVERSION

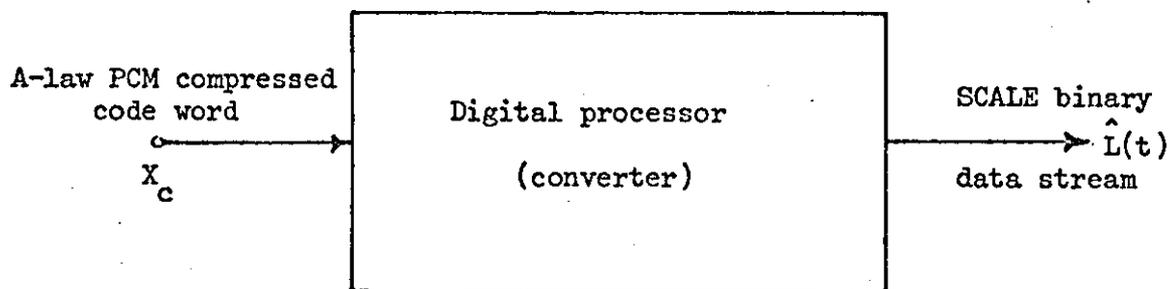
#### 7.1 INTRODUCTION

As discussed in Chapter I, the A-law PCM compressed code has been internationally agreed for interexchange speech transmission. Nevertheless, situations where the need for A-law PCM - to - SCALE format conversion might arise exist (see Chapter VIII). The purpose of this chapter is to discuss the theory and design of an all-digital converter which has been conceived for this purpose. This converter accepts 8-bit word A-law PCM compressed code words at Nyquist rate ( $8 \text{ KH}_z$ ) and produces SCALE binary data at a line rate which is the same as that of the A-law PCM (64 Kb/s). An attempt was made to ensure that when the resultant SCALE data stream  $\hat{L}(t)$  is decoded by an ordinary hybrid SCALE decoder, the filtered output is a close replica of the base band analogue signal which has been presented to the A-law PCM encoder.

It is possible, of course, to convert from A-law PCM to SCALE by first decoding the A-law code words to obtain the analogue signal they represent and then encoding the resultant analogue signal into SCALE binary data stream using a SCALE encoder. This approach, which is depicted in Fig. 7.1a, will not only lead to an expensive and bulky converter (as a result of the large amount of hardware required for its implementation), but also the resultant converter will have poor performance (as a result of the noise introduced during the second - i.e.  $\hat{x}_2(t)$  to  $\hat{L}(t)$  - analogue to digital conversions). For these reasons the all-digital approach depicted in Fig. 7.1b is preferred.



(a)



(b)

Figure 7.1 Two possible approaches to the problem of A-law - to - SCALE conversion.

(a) Digital-Analogue-Digital method

(b) All-digital method

## 7.2 DESCRIPTION OF THE ALL-DIGITAL CONVERTER

The A-law PCM generation can be viewed as a single process which converts the base band analogue signal  $x_2(t)$  into a compressed signal  $X_c$  in the digital domain. Alternatively, the compressed code  $X_c$  can be viewed as generated from  $x_2(t)$  by a linear encoder (Analogue-to-digital converter) followed by a digital compressor. The latter case is more attractive because it allows  $X_c$  to be first linearised (A-law to Linear PCM converted) and then the resultant linear PCM words are converted to SCALE binary data stream. A strategy for the design of the all-digital A-law - to - SCALE conversion, based on the latter approach, is shown in Fig. 7.2. The operation of this converter is described in the following paragraphs:-

### 7.2.2 The Digital expander

The compressed A-law PCM code word  $X_c$  available at the input of the digital expander DE consists of 8 bit binary sequence  $g_8 \dots g_1$  (see equation (6.17)). The most significant digit  $g_8$  represents the polarity of the sample,  $g_7, g_6, g_5$  represent the segment number  $J_s$  in which the step lies and  $g_4 \dots g_1$  represent the step number  $S_q$  within  $J_s$ .

$$\begin{aligned}
 J_s &= \sum_{k=1}^3 2^{3-k} g_{(4+3+1-k)} \\
 \text{and } S_q &= \sum_{k=1}^4 2^{4-k} g_{(4+1-k)}
 \end{aligned}
 \left. \vphantom{\begin{aligned} J_s \\ S_q \end{aligned}} \right\} \begin{array}{l} (7.1) \\ \text{See equations (6.18)} \\ \text{and (6.19)} \end{array}$$

When the compressed code  $X_c(J_s, S_q)$  is converted by the digital expander to the linear PCM code  $X_L(J_s, S_q)$ , the latter should correspond to the A-law PCM decoder output level. The relationship between  $X_L(J_s, S_q)$  and  $X_c(J_s, S_q)$  is defined by equation (6.15). For convenience the latter equation is recalled below:

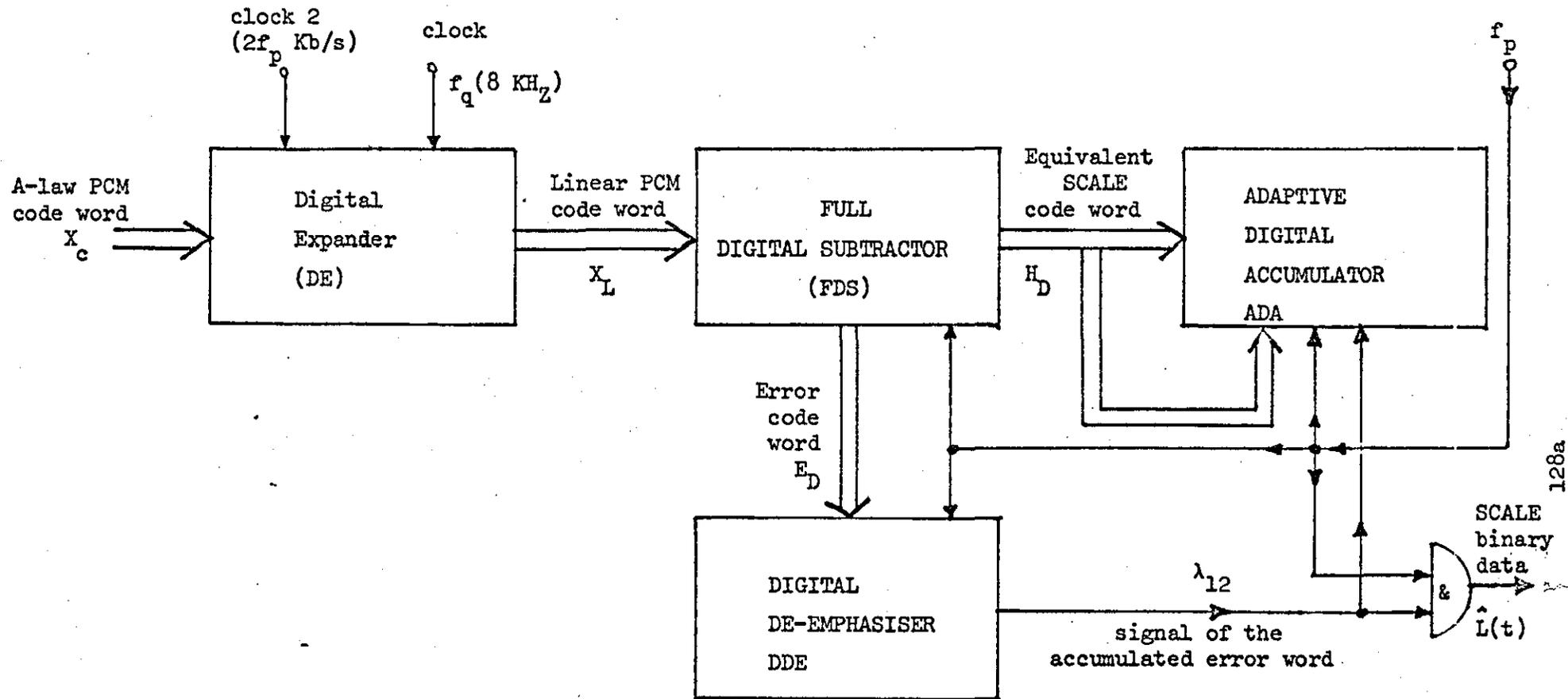


Figure 7.2 Block diagram of the all-digital A-law PCM - to - SCALE converter

$$\left. \begin{aligned}
 X_L(J_s, S_q) &= \Delta_c(J_s) \cdot (S_q + P_t) - k_a \\
 &= 2^{J_s - \eta_a} (S_q + \eta_a \cdot N_t + 0.5)
 \end{aligned} \right\} \quad (7.2)$$

where

$$k_a = 0, \quad \eta_a = \begin{cases} 0 & J_s = 0 \\ 1 & J_s \neq 0 \end{cases} \quad (\text{see also equation } (6.16))$$

$N_t$  = the number of steps in a segment (= 16).

This value of  $X_L(J_s, S_q)$  above excludes the sign (the companding law is symmetric for positive and negative signals). Now

$$\Delta_c = 2^{J_s - \eta_a}$$

is a shifting operator, such that the binary sequence  $(S_q + \eta_a \cdot N_t + 0.5)$  is shifted to the left by  $(J_s - \eta_a)$ . Therefore the algorithm for code conversion from A-law to linear PCM is:-

- (i) Add  $(\eta_a N_t + 0.5)$  to  $S_q$
- (ii) Shift the sum  $(S_q + \eta_a \cdot N_t + 0.5)$  by  $(J_s - \eta_a)$  positions to the left (i.e. division by  $(2^{J_s - \eta_a})^{-1}$ )

where all the parameters involved are in binary representation.

A parallel digital expander based on the above argument is shown in Fig. 7.3. The function of the selector LS is to select  $(N_t + 0.5)$  or  $(0.5)$  for the addition to  $S_q$ . The selection depends on the segment number  $J_s$ . When  $J_s = 0$  (i.e.  $g_7 = g_6 = g_5 = 0$ ) the linearised output signal falls in the 0<sup>th</sup> segment and  $\eta_a$  therefore equals 0. In this case the signal  $S_\eta = 0$  and input B is selected. If, however,  $J_s \neq 0$ , then  $\eta_a = 1$  and input A is selected. The full adder forms the sum of  $S_q$  and  $(\eta_a \cdot N_t + 0.5)$  and the sum is then loaded in parallel into the first 6 LSD position of the 12-bit shift register. The full adder is of the parallel type so that the selection and addition operations are rapidly executed (in approximately 50 n. seconds when TTL integrated circuits

are used).  $J_s$  is loaded into a binary counter which then starts counting down and continues doing so until the two most significant digits of its contents are zeros (i.e. 000 or 100). This process takes  $(J_s - \eta_a)$  clock2 periods (clock2 runs at a rate of  $2f_p$  Kb/s where  $f_p = 64$  Kb/s). The contents of the shift register are shifted to the left by the same clock while the vacant positions of the shifted data are being filled by zeros which are shifted serially into the DL terminal (see Fig. 7.3). When the  $(J_s - \eta_a)$  shifts are performed the contents of the 11 most significant positions yield the linearised A-law PCM code which is the linear PCM sequence  $u_1 u_2 \dots u_{11}$ , and its polarity  $\epsilon_8$ .

### 7.2.2 The Linear PCM - to - SCALE converter

This consists of the full digital subtractor FDS, the adaptive digital accumulator ADA, the digital de-emphasiser DDE and the associated timing circuitry (not shown) - see Fig. 7.2. In this subunit of the A-law PCM - to - SCALE converter an estimate of the linear PCM word  $X_L(J_s, S_q)$  is formed by the ADA. This estimate is then subtracted from  $X_L(J_s, S_q)$  by the FDS and the error is modified according to the history of the previous errors. The modified error is then used to improve the estimation and also to generate the SCALE binary data representation of  $X_L(J_s, S_q)$  - and hence of the A-law compressed code  $X_c(J_s, S_q)$ . To reduce the converter complexity to a minimum it is assumed that the minimum LPCM step size is equal to the minimum SCALE step increment. The estimation is carried out in the digital domain and the estimated value of  $X_L(J_s, S_q) - H_D$ , say - is represented by a 12-bit binary word, one of which represents the polarity, while the remaining 11 digits ( $a_{11} a_{10} \dots a_1$ ) represent  $|H_D|$ , the estimated magnitude of  $X_L(J_s, S_q)$ . The estimation process of each code word  $X_L(J_s, S_q)$  must be

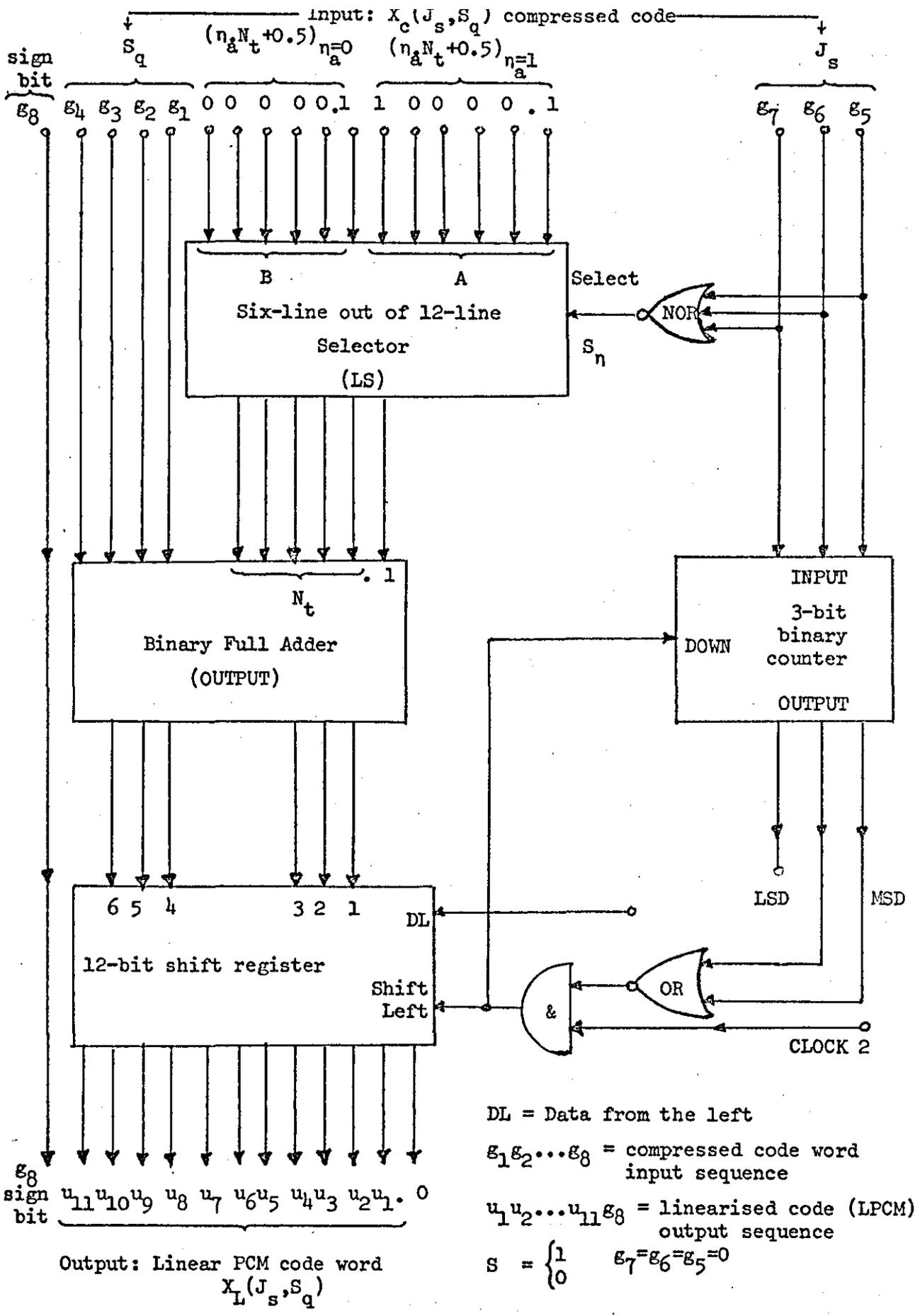


Figure 7.3 Block diagram of the parallel digital A-law PCM expander (A-law to Linear PCM converter)

completed within the Nyquist interval. For this reason  $H_D = \{a_{12}a_{11}\dots a_1\}$  must be a good estimate of  $X_L(J_s, S_q)$  at least  $f_q = (8000)$  times per second. A suitable estimation algorithm is that of the digital SCALE decoder discussed in Chapter V. In this case the Adaptive digital accumulator is simply a digital SCALE decoder which is clocked at a rate  $f_p$  Kb/s, where  $f_p = 8f_q$ . Every  $T_p (= 1/f_p = 1/8f_q)$  the content of the accumulator  $H_{Di}$  (subscript means at the  $i^{\text{th}}$  clock instance) is subtracted from  $X_L(J_s, S_q)$  to produce an error signal  $E_{Di}$  which is an 11-bit word representing the error magnitude and one bit representing the error polarity. Therefore

$$E_{Di} = X_L(J_s, S_q) - H_{Di} \quad (7.3)$$

where

$$\begin{aligned} X_L(J_s, S_q) &= \{u_{12}u_{11}\dots u_1\} \\ &= \pm \sum_{k=1}^{11} 2^{k-1} u_k \quad (+ \text{ when } u_{12} = 1, - \text{ when } u_{12} = 0) \end{aligned}$$

$$\begin{aligned} H_{Di} &= \{a_{12}a_{11}\dots a_1\} \\ &= \pm \sum_{k=1}^{11} 2^{k-1} a_k \quad (+ \text{ when } a_{12} = 1, - \text{ when } a_{12} = 0) \end{aligned}$$

$$\begin{aligned} E_{Di} &= \{v_{12}v_{11}\dots v_1\} \\ &= \pm \sum_{k=1}^{11} 2^{k-1} v_k \quad (+ \text{ when } v_{12} = 1, - \text{ when } v_{12} = 0) \quad (7.4) \end{aligned}$$

The DDE is a leaky digital integrator whose leakage is determined by the forward integrator  $C_1R_1$  of the hybrid SCALE encoder shown in Fig. 7.4 (see also Fig. 2.3 of Chapter II). In the present situation  $x_2(t)$  is

represented by a discrete sample word  $X_L(J_s, S_q)$ . This means that the DDE input/output relation can be written as

$$\begin{aligned} E_{Ii} &= E_{Di} \left[ 1 - e^{-T_p/C_1 R_1} \right] + E_{I(i-1)} \cdot e^{-T_p/C_1 R_1} \\ &= \left[ E_{I(i-1)} - E_{Di} \right] \cdot e^{-T_p/C_1 R_1} + E_{Di} \end{aligned} \quad (7.5a)$$

$$= E_{Di} + L_g \cdot \left[ E_{I(i-1)} - E_{Di} \right] \quad (7.5b)$$

where  $L_g \equiv e^{-T_p/C_1 R_1}$  and all parameters are in binary representation.

The Digital spectrum shaping and comparison (between  $X_L$  and  $H_D$ ) algorithm, therefore, is:

- (i) Subtract the present error word  $E_{Di}$  from the previous modified error word  $E_{I(i-1)}$ .
- (ii) Multiply the result by the leakage word  $L_g$ .
- (iii) Add this product to the present error word  $E_{Di}$ .

All the parameters are represented in binary form and include their polarities.

The equation for  $E_{Ii}$  is:-

$$\begin{aligned} E_{Ii} &= \{\lambda_{12} \lambda_{11} \dots \lambda_1\} \\ &= \pm \sum_{k=1}^{11} 2^{k-1} \lambda_k \quad (+ \text{ when } \lambda_{12} = 1, - \text{ when } \lambda_{12} = 0) \end{aligned} \quad (7.6)$$

The rate at which the sign digit  $\lambda_{12}$  is produced is  $f_p$  Kb/s. It is analogous to  $V_q$  in the hybrid SCALE system (see Chapter II Fig. 2.3 and Fig. 7.4a). i.e.

$$\lambda_{12} = \text{Sgn}(E_{Ii}) \quad (7.7)$$

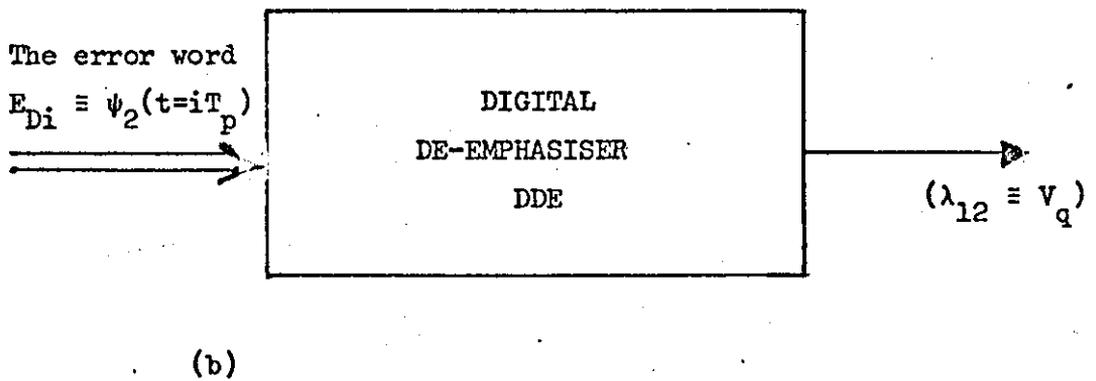
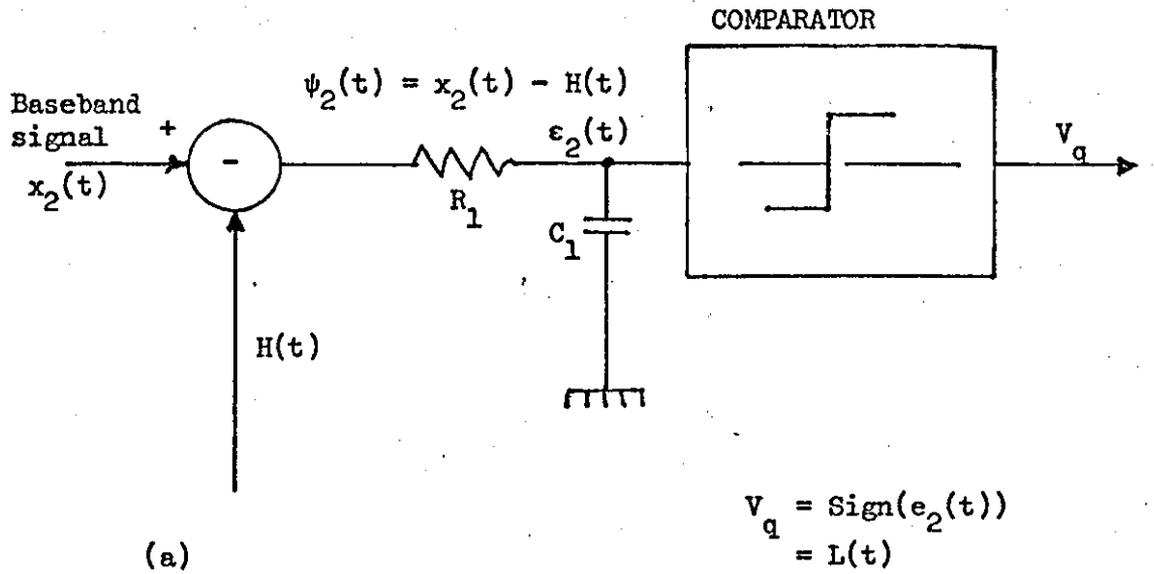


Figure 7.4 (a) The forward path of the hybrid SCALE encoder  
 (b) The forward path of the digital equivalent of (a)  
 (excluding the subtractor)

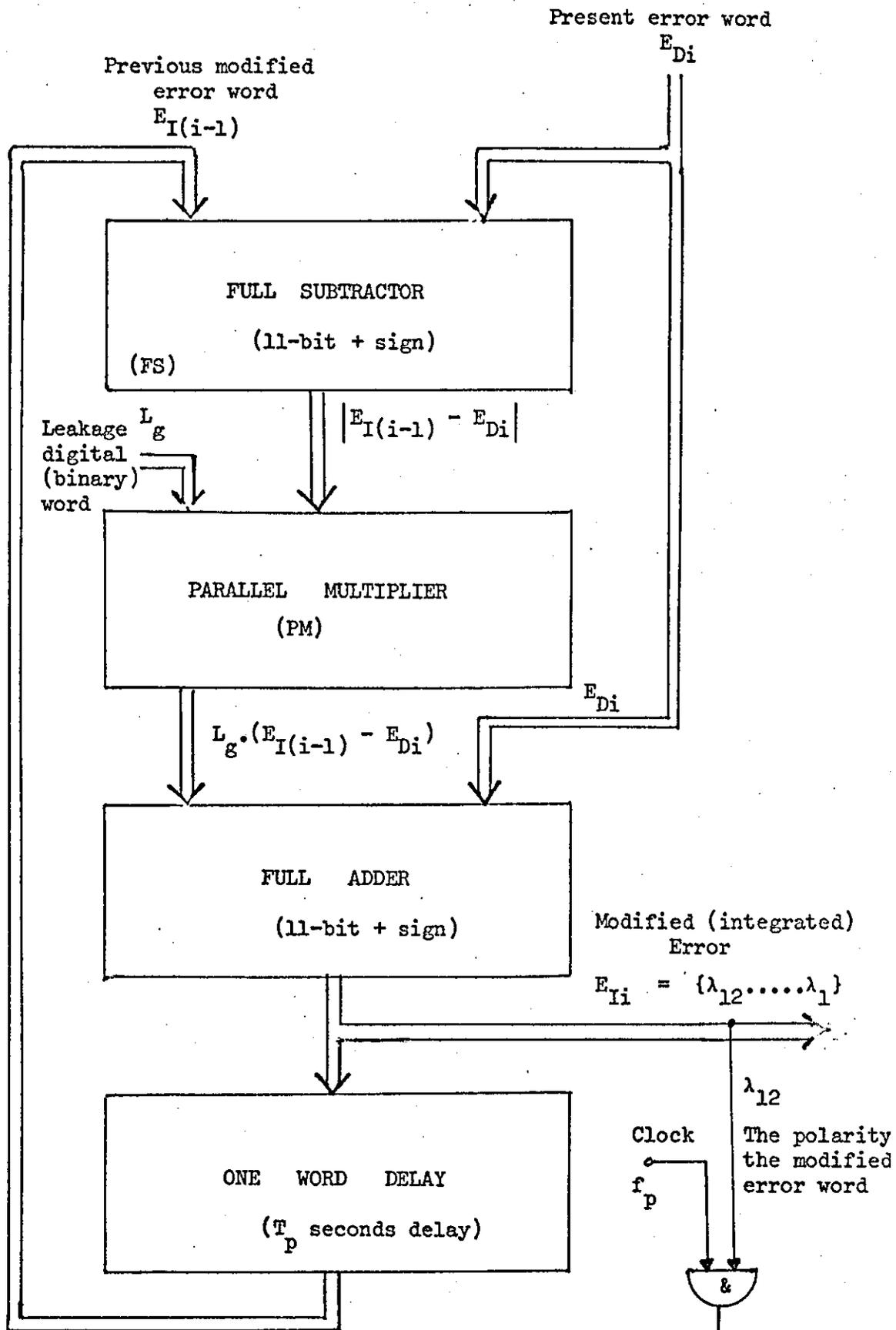


Figure 7.4.c Block diagram of the digital De-emphasiser (DDE) based on the algorithm described by equation (7.5). Both  $E_{Di}$  and  $E_{Ii}$  (and hence  $\lambda_{12}$ ) are produced at a rate of 64 KH<sub>z</sub>.

$$E_{Ii} = E_{Di} + L_g \cdot [E_{I(i-1)} - E_{Di}]$$

where  $V_q$  in the hybrid SCALE system is given by:

$$V_q = \text{Sign}(\varepsilon_2(t)).$$

$\lambda_{12}$  is fed back to the ADA to update its contents and also gated by the clock  $f_p$  to produce the SCALE binary data stream  $\hat{L}(t)$  which represents the digital signal  $X_c$  (and hence represents  $x_2(t)$  at the input of the A-law PCM encoder).

Because  $X_L$  is generated from  $X_c$  at the Nyquist rate (8 KHz), while  $\lambda_{12}$  is generated by the converter at a rate of 64 Kb/s;  $H_D$  is up-dated 8 times during each Nyquist interval. This enables  $H_{Di}$ , the locally constructed estimate  $X_L$ , and  $X_L$  itself to be very close to each other in magnitude. The magnitude of  $H_{Di}$  is incremented as long as the present and the two previous logical values of  $\lambda_{12}$  are identical; and is decremented otherwise. The polarity of  $H_{Di}$  is positive when the present logical value of  $\lambda_{12}$  is a logical one, otherwise it is negative.

The remote SCALE decoder receiving  $\hat{L}(t)$  reconstructs a signal  $H(t)$  or  $\hat{H}(t)$  (depending on whether it is a hybrid or digital SCALE decoder) which is very close to the one which would be constructed from the binary data stream  $L(t)$  originating from a SCALE encoder whose input signal is  $x_2(t)$ .

A digital circuit which realizes equation (7.5) is given in Fig. 7.4c.

The Adaptive digital accumulator (which is a digital SCALE decoder with minimum step size variation of 1 m. volt) up-dates its contents according to the digital SCALE algorithm described in Chapter V. This means that : *"THE ACCUMULATOR INCREMENTS ITS CONTENTS BY (00000001000), (0000000110) OR (00000000100) WHEN  $\lambda_{12}(iT_p) \equiv \lambda_{12}(iT_p - T_p) \equiv \lambda_{12}(iT_p - 2T_p) = 1$  AND ITS PRESENT CONTENT IS IN THE RANGES (000000011000) TO (10000000000), (10000000000) TO (11100000000) OR (10000000000) TO (11111111111) RESPECTIVELY. WHEN  $\lambda_{12}(iT_p)$ ,  $\lambda_{12}(iT_p - T_p)$ , AND  $\lambda_{12}(iT_p - 2T_p)$  ARE NOT IDENTICAL AND THE CONTENTS OF THE ACCUMULATOR FALL WITHIN THE ABOVE RANGES THE ACCUMULATOR DECREMENTS ITS CONTENT RESPECTIVELY BY (00000000001),*

(00000000100), (00000001000)."

The SCALE binary data  $\hat{L}(t = iT_p)$  generated by the converter can be decoded by both the digital and the hybrid versions of the SCALE system.

### 7.3 IMPLEMENTATION OF THE A-LAW PCM - TO - SCALE CONVERTER

The A-law PCM - to - SCALE converter discussed above (see Fig. 7.2) can be implemented in hardware using economically available off-the-shelf small and medium integration integrated circuits. In this section a circuit diagram for the converter using Texas digital integrated circuits will be presented and discussed. The implementation of the individual sub-units (DE, FDS, etc.) are considered separately. The sub-units are connected as shown in Fig. 7.2 to form the complete converter.

#### 7.3.1 The Digital expander

The digital expander circuit diagram which was designed to realize the block diagram of Fig. 7.3 is shown in Fig. 7.5. In the circuit diagram the two six-line to one-line selectors have been replaced by one two-input AND gate. This simplification became possible when it was realized that the output  $(n_a \cdot N_t + \frac{1}{2})$  of the selector is either (00000.1) or (10000.1) - i.e. only one digit changes. The change of the selector output from (00000.1) to (10000.1) occurs when the signal level is outside the 0<sup>th</sup> segment ( $J_s \neq 0$ ). In Fig. 7.5 the NOR gate (SN7427) accepts,  $J_s \{g_7g_6g_5\}$ , and produces a logical one ( $ly = 1$ ). This output of the NOR gate is combined with 5 volts level by the NAND gate (SN74000) to produce a logical zero when ( $J_s = 0$ ) and a logical one when ( $J_s \neq 0$ ). The digit produced is then placed in the 5<sup>th</sup> least significant position of the 8-bit parallel Full adder (SN7483A). The rest of the adder A-inputs are connected to earth. The 4 least significant digits of the adder B-inputs are fed by  $S_q$  - the step number in segment  $J_s$  - while

From timing circuit Fig.  
6.19.1 (SN74193)

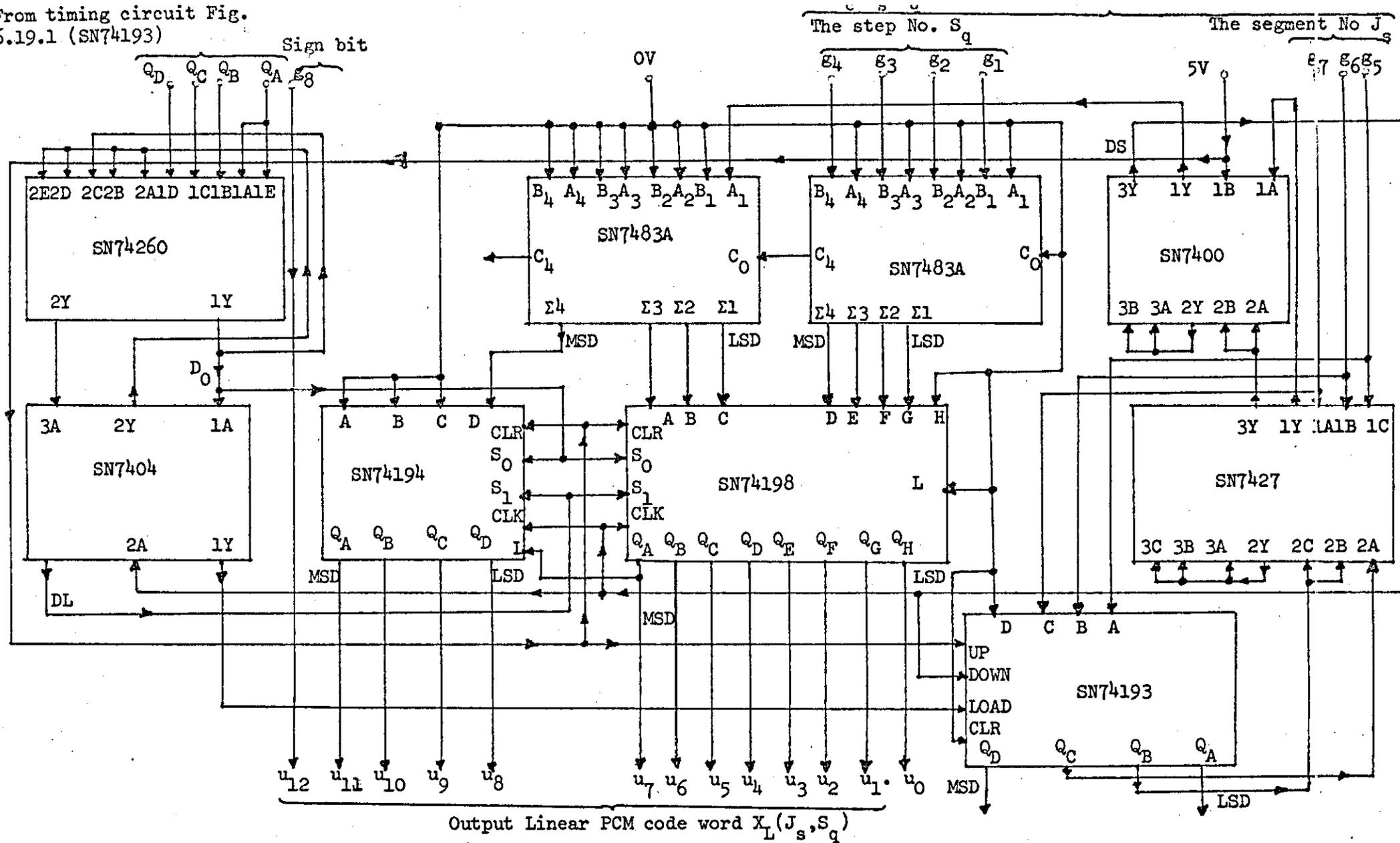


Figure 7.5 Circuit diagram of the Digital expander using Texas integrated circuits

the rest of the adder B-inputs are connected to ground.

The 4-bit binary adder blocks (SN7483A  $\equiv$  SN74283) are full adders which perform the addition of  $2 \times 4$ -bit binary words. Two blocks were cascaded to accommodate the width of the input signal (5 digits wide). The input, output and the carry logic levels are in their true form so no inversion is required. The adders feature full look-ahead across the 4-bit words and generate the carry term in approximately 10 n. sec. The typical add-time for two 8-bit words is 23 n. sec.<sup>(88)</sup> so the sum of  $(\eta_a \cdot N_t + S_q)$  is produced about 25 n. sec. or less after the compressed code  $X_c(J_s, S_q)$  has been received by the converter. This sum is then loaded in parallel into the second and the next seven least significant positions of the 12-bit shift register (SN74198 and SN74194). The first position in the shift register is loaded with a logical one and is a fraction to complete the formation of the value  $(\eta_a \cdot N_t + S_q + 0.5)$ .  $J_s = \{g_7 \dots g_5\}$  is also now loaded into the binary counter SN74193. The timing signal needed for the loading of the shift register is generated from  $\{Q_A \dots Q_D\}$  by the logic circuit comprising the dual 5-input NAND gate SN74260 and the Hex inverter SN7404. The sequence  $\{Q_A \dots Q_D\}$  is the content of the binary up-counter SN74193 shown in the timing circuit of Fig. 6.19.1. We recall that this counter is used as a frequency divider which divides its clock frequency ( $2f_p = 128 \text{ KH}_z$ ) by 16 enabling Nyquist interval to be divided by 8. In Fig. 7.5:

$$\left. \begin{aligned} \bar{D}_0 &= \bar{Q}_D + \bar{Q}_C + \bar{Q}_B + \bar{Q}_A \\ D_L &= \bar{D}_S + D_0 \end{aligned} \right\} \quad (7.8)$$

This now means that in the first  $(T_p/8)$  of each Nyquist interval  $\bar{D}_0$  is true and  $D_L$  is true when  $D_S$  is FALSE. The first two logical conditions are needed to load the shift register, while the third is

required to load  $J_s$  into the 4-bit binary counter SN74193. When this first interval is over,  $D_L$  remains TRUE while  $\bar{D}_0$  becomes FALSE. This is the correct condition for enabling the contents of the shift register to be shifted from right to left (multiplication by 2). The counter (SN74193) now counts down until its contents are at most (0001). Every time the counter contents are decremented by one the contents of the shift register are shifted one position from right to left and zeros are entered from the left. When the counter stops counting down the shift register contents would have been shifted to the left by  $(J_s - n_a)$  times. This completes the operation defined by the relation

$$X_L(J_s, S_q) = 2^{(J_s - n_a)} \{S_q + N_t n_a + 0.5\}$$

discussed earlier. The linearized code  $X_L(J_s, S_q)$  which is the Linear PCM representation of the A-law code  $X_c(J_s, S_q)$  is now read out in the form of the sequence  $(u_{11} u_{10} \dots u_1)$ , plus the polarity digit  $u_{12}$ .

The pin layout and pin designation of the integrated circuits used for the synthesis of the converter have already been described in Chapters V and VI (e.g. see Chapter 6 Section 3). Further details of these integrated circuits characteristics can be found in the manufacturer's data sheets, etc. (88,89)

### 7.3.2 The Full subtractor

The function of the FAS (see Fig. 7.2) is to form the difference  $D_{Ei} = X_L(J_s, S_q) - H_{Di}$  where  $i$  is the  $i^{\text{th}}$  clock pulse instance. The clock frequency is  $f_p$  Kb/s. Eight subtraction operations are performed every Nyquist interval (125  $\mu$ . sec.). This calls for a fast parallel subtractor to prevent errors due to subtraction time delays.

The difference between the magnitudes of  $X_L(J_s, S_q)$  and  $H_{Di}$  can be formed in approximately 50 n. secs. employing the method of subtract-

ion by addition of the 2's complements. The magnitudes of  $X_L(J_s, S_q)$  and  $H_{Di}$  consist of 11-digit words. If the negative of  $|H_{Di}|$  is represented by the 2's complement of  $|H_{Di}|$ , i.e.  $\{2^{11+1} - |H_{Di}|\}$ , and added to  $|X_L(J_s, S_q)|$  the result will be

$$\{2^{12} - |H_{Di}|\} + |X_L(J_s, S_q)| = \{X_L(J_s, S_q) - |H_{Di}|\} + 2^{12} \quad (7.9)$$

The two's complement of  $|H_{Di}|$  can be generated by interchanging the logical ones and logical zeros in  $|H_{Di}|$  and adding one to the result. The interchange of the 11 digits of  $|H_{Di}|$  can be carried out using 3 cascaded true-zero complement elements such as Texas SN74H87. This operation can then be performed in approximately 40 n. secs. The sum of  $|X_L(J_s, S_q)|$  and the complement of  $|H_{Di}|$  can be formed in approximately 35 n. sec. using 3 Texas SN7483A 4-bit adder blocks cascaded as shown in Fig. 7.6. The  $2^{12}$  term appears in the most significant position of the last stage of the adder (position 12) and indicates that the result of the subtraction is positive (sign bit). When this bit is a zero the output is a negative quantity and appears in the 2's complement form. To ensure that the output is always in its true form the outputs of the adder are passed through another set of true-zero complement elements which are controlled by the sign bit residing in the most significant position of the adder. When the sign bit is a logical one the outputs of the adders are passed to the output unaltered. When the sign bit is a logical zero the adders' outputs are complemented. It must be emphasised that the output obtained so far represents the difference between the magnitudes of  $X_L(J_s, S_q)$  and  $H_{Di}$ . The polarity digit  $v_{12}$  of the error word can be determined from  $P_D$ , the polarity of the magnitudes' difference (see Fig. 7.6a) and the polarities  $u_{12}$  and  $a_{12}$  of the input words to the subtractor. This can be achieved by first constructing the truth table shown in Fig. 7.6b.

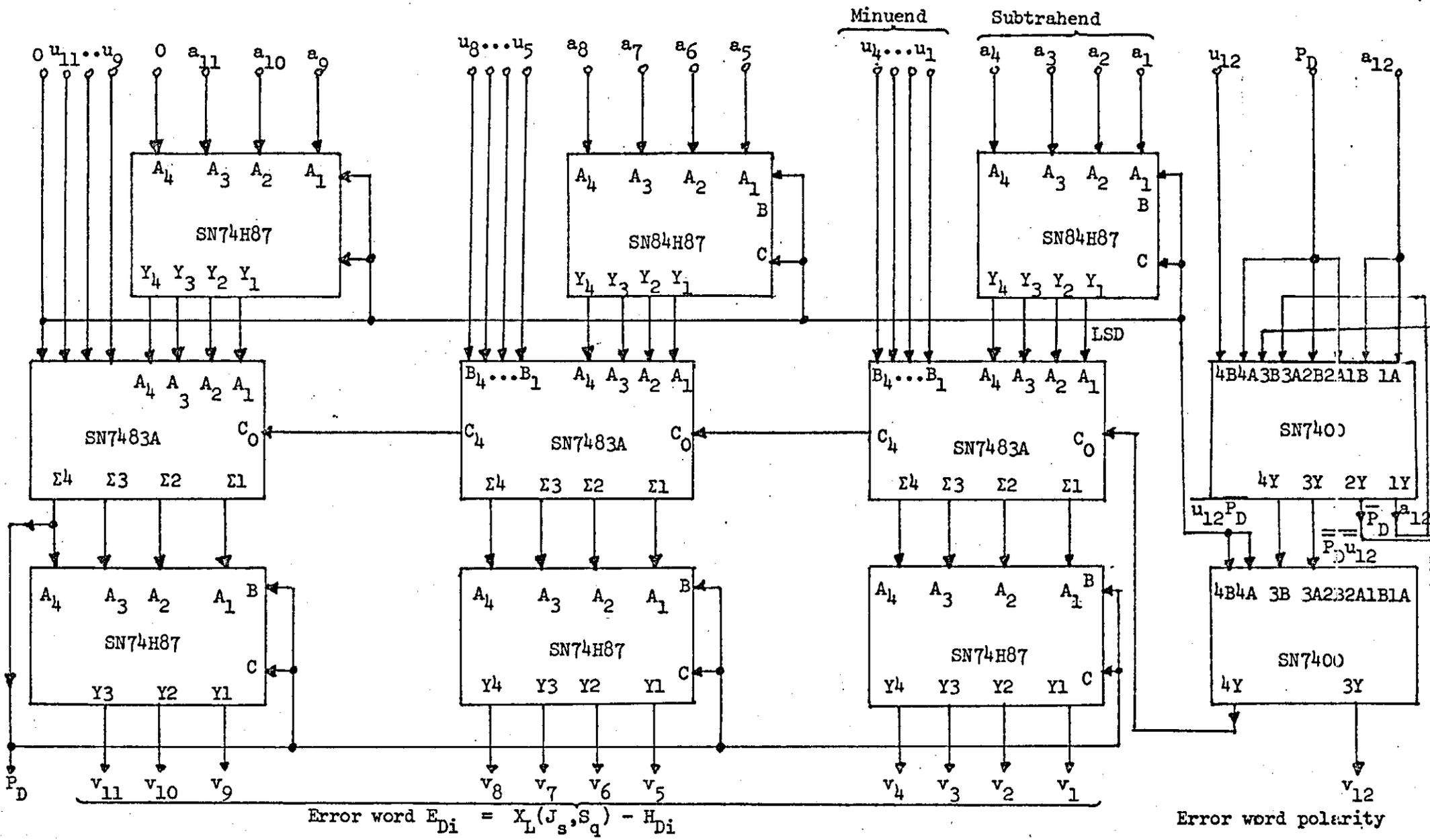


Figure 7.6a Implementation of the Digital subtractor using Texas integrated circuits

Polarity bit of $X_L(J_s, S_q)$ $u_{12}$	Polarity digit of $H_{Di} : a_{12}$ $a_{12}$	Polarity digit of $ X_L - H_D $ $P_D$	(Meaning)	Polarity digit of $E_{Di} : v_{12}$
0	0	0	} $X_L$ and $H_D$ are negative and $ H_D  >  X_L $	1
0	0	1		} $X_L$ and $H_D$ are negative and $ H_D  <  X_L $
0	1	0	} $X_L$ is negative, $H_D$ positive and $ H_D  >  X_L $	0
0	1	1		} $X_L$ is negative, $H_D$ positive and $ H_D  <  X_L $
1	0	0	} $X_L$ is positive, $H_D$ negative and $ H_D  >  X_L $	1
1	0	1		} $X_L$ is positive, $H_D$ negative and $ H_D  <  X_L $
1	1	0	} $X_L$ and $H_D$ are positive and $ H_D  >  X_L $	0
1	0	1		} $X_L$ and $H_D$ are positive and $ H_D  <  X_L $

Figure 7.6b Truth table of the relationships between the polarity digits of  $X_L(J_s, S_q)$  (i.e.  $u_{12}$ ),  $H_{Di}$  (i.e.  $a_{12}$ )  $\{|X_L| - |H_D|\}$  (i.e.  $P_D$ ) and  $E_{Di}$  (i.e.  $v_{12}$ )

$$v_{12} = u_{12} \cdot P_D + \bar{a}_{12} \cdot \bar{P}_D$$

The logical equation relating  $u_{12}^{P_D, a_{12}}$  and  $v_{12}$  can then be written from the truth table.

$$v_{12} = u_{12}^{P_D} + \bar{a}_{12}^{P_D} \quad (7.10)$$

The polarity digit  $v_{12}$  of the error word can therefore be generated from the other available sign bits as shown in Fig. 7.6. The circuit in this figure is a complete 12-bit Full subtractor.

### 7.3.3 The Digital De-emphasiser

The algorithm for realizing this subunit of the converter has been stated in Section 7.7.2. A block diagram of the DDE based on this algorithm is also given in Fig. 7.4.c.

Now, referring to Fig. 7.4.c, the subtractor is identical to that described in the previous section of this chapter (see Fig. 7.6). The addition of two 12-bit binary words and the delay of a binary word by one clock period have also been discussed in detail in Chapters V and VI (e.g. see Section 5.2 of Chapter VI and Fig. 6.17.1b). It is therefore unnecessary to describe these subunits again. The implementation of the parallel multiplier using Texas integrated circuits is discussed below.

### The Parallel Multiplier

The parallel multiplier function is to form the product of the leakage coefficient  $L_g$  and the difference between the previous value of the modified error  $E_{I(i-1)}$  and the present error word  $E_{Ii}$ .

$$\begin{aligned} \text{Let } \hat{E}_{Di} &= \{E_{I(i-1)} - E_{Di}\} & (7.11) \\ &= \{\theta_{12} \theta_{11} \dots \theta_1\} \end{aligned}$$

where  $\theta = \{0,1\}$  and  $\theta_{12}$  represents the polarity of the 11-digit magnitude word.

Now the leakage  $L_g \equiv e^{-T/P/C_1R_1}$  (see equation (7.5a)), where  $C_1R_1$  is the time constant of the forward integrator of the hybrid SCALE encoder which was implemented in hardware and described in Chapter III;  $C_1R_1 = 0.0002$  secs and  $L_g$  in decimal notation is 0.924848813. To represent this number in binary notation a very large sequence is required; and when this is multiplied by  $\hat{E}_{Di}$  a prohibitively long product sequence results. The multiplier implementation in this case is very complex and expensive. To avoid these complications  $L_g$  can be approximated by an 8-bit binary word  $\hat{L}_g$  consisting of the binary sequence  $(\ell_1 \ell_2 \dots \ell_8)$  where  $\ell_1$  is the MSD. So,

$$\hat{L}_g = \left. \begin{aligned} & \sum_{k=1}^8 \ell_k \cdot 2^{-k} \\ & 2^{-1} + 2^{-2} + 2^{-5} + 2^{-16} \end{aligned} \right\} \quad (7.12)$$

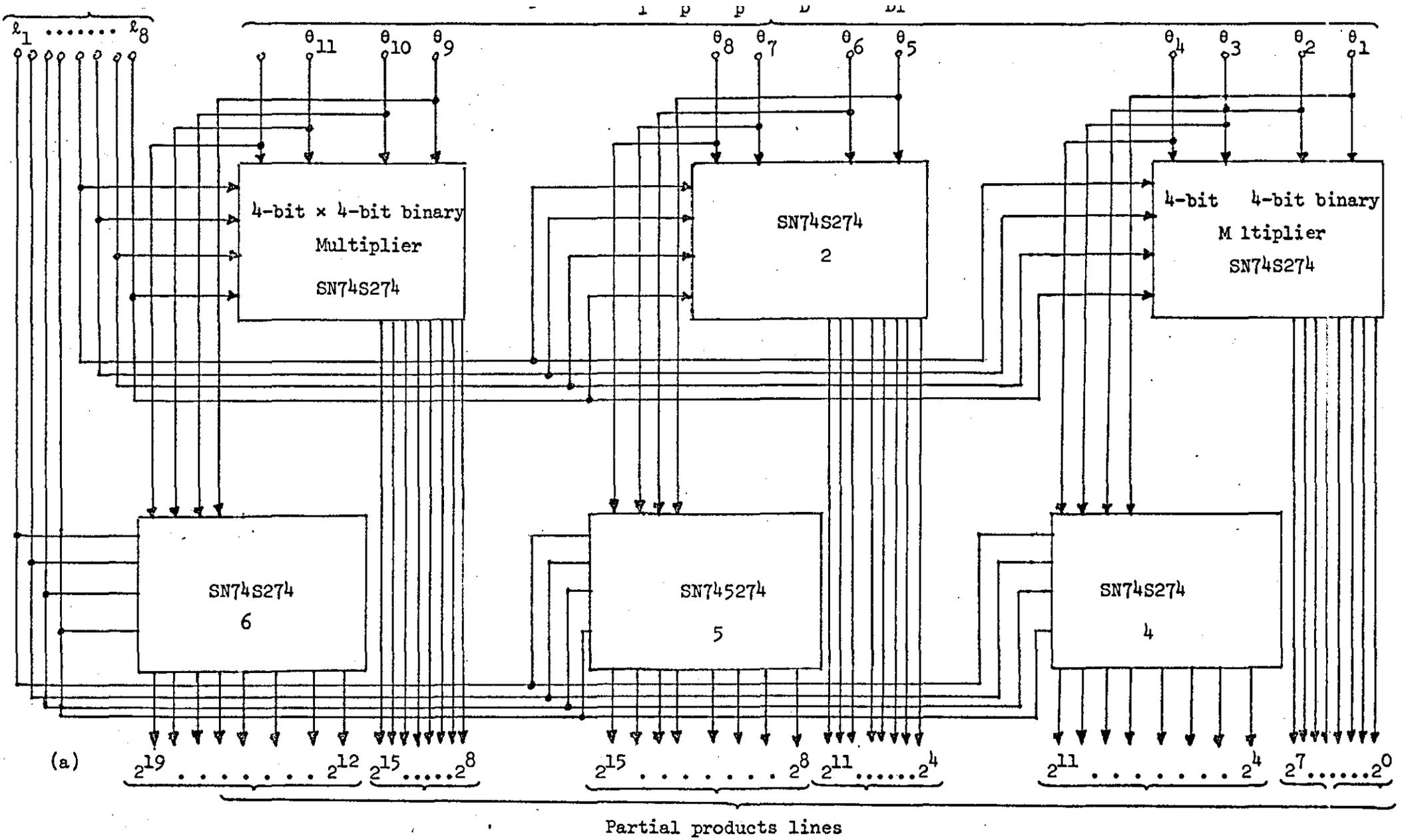
$$= 0.922$$

(i.e.  $\ell_1 = \ell_2 = \ell_3 = \ell_5 = \ell_6 = 1$  and  $\ell_4 = \ell_7 = \ell_8 = 0$ )

This approximation enables a reasonably manageable multiplier and the penalty suffered in terms of error is very small. The approximation has the effect of replacing the time constant ( $C_1R_1 = 0.0002$  seconds) of the hybrid SCALE by  $\hat{C}_1\hat{R}_1 = 0.0002025$  seconds.

The product of  $\hat{L}_g$  and  $\hat{E}_{Di}$  (see equations (7.11) and (7.12)) can be produced in  $\approx 100$  n. secs. using parallel multiplication strategy. The complete multiplier based on the above argument is shown in Fig. 7.7.

In Fig. 7.7a Texas SN74S274 4-bit  $\times$  4-bit multipliers are employed to produce the partial products of the two input words  $\hat{L}_g$  and  $\hat{E}_{Di}$ . Now to form the final product it is necessary to use a fast addition scheme to sum the partial products produced by the multiplier chips.



139a

Figure 7.7 Implementation of the 8-bit x 11-bit multiplier using Texas integrated circuit  
 (a) 6 Cascaded 4-bit x 4 bit multiplier to form 6 partial products

Texas SN74S275 and SN74H183 are "Carry Save" adders which generate the carry term at the same time as the sum term. When these adders are arranged in the form of Wallace trees as shown in Fig. 7.7.b the final product of  $\hat{L}_g$  and  $\hat{E}_{Di}$  is formed and appears at the output terminals of the adders. Because  $\hat{L}_g$  is a positive constant number the polarity of the product word is the same as that of  $\hat{E}_{Di}$ . This means that the polarity bit =  $\theta_{12}$ .

Other multiplication schemes such as those employing Read Only Memory (ROM) to implement the  $4 \times 4$  bit multipliers; or performing the multiplication serially with the aid of a clock, serial multipliers and an 11-bit shift register, are also possible. For a prototype implementation however the difference between these methods and the parallel scheme adopted above, from the economic point of view, is not significant while the gain in speed is quite attractive.

#### 7.3.4 The Adaptive digital Accumulator

This, as discussed earlier, is simply a digital SCALE decoder discussed in Chapter V. For this reason it is not necessary to go into the details of its implementation again.

### 7.4 COMPUTER SIMULATION OF THE A-LAW - TO - SCALE CONVERTER

To test the validity of the theory of the A-law - to - SCALE converter discussed earlier in this chapter, a computer program was compiled to simulate it on the digital computer. A flow diagram summarizing the main simulation procedure is given in Fig. 7.8. The A-law PCM code words  $X_c$  are read from a buffer store at the Nyquist rate, i.e. every 125  $\mu$  seconds. These words are used as an input to "SUBROUTINE EXPANDER" which operates on the compressed A-law code to produce the equivalent linear PCM code words  $X_L$ . "SUBROUTINE EXPANDER" is basically a FORTRAN IV translation of equation (7.2) and is listed

from multipliers (partial products)

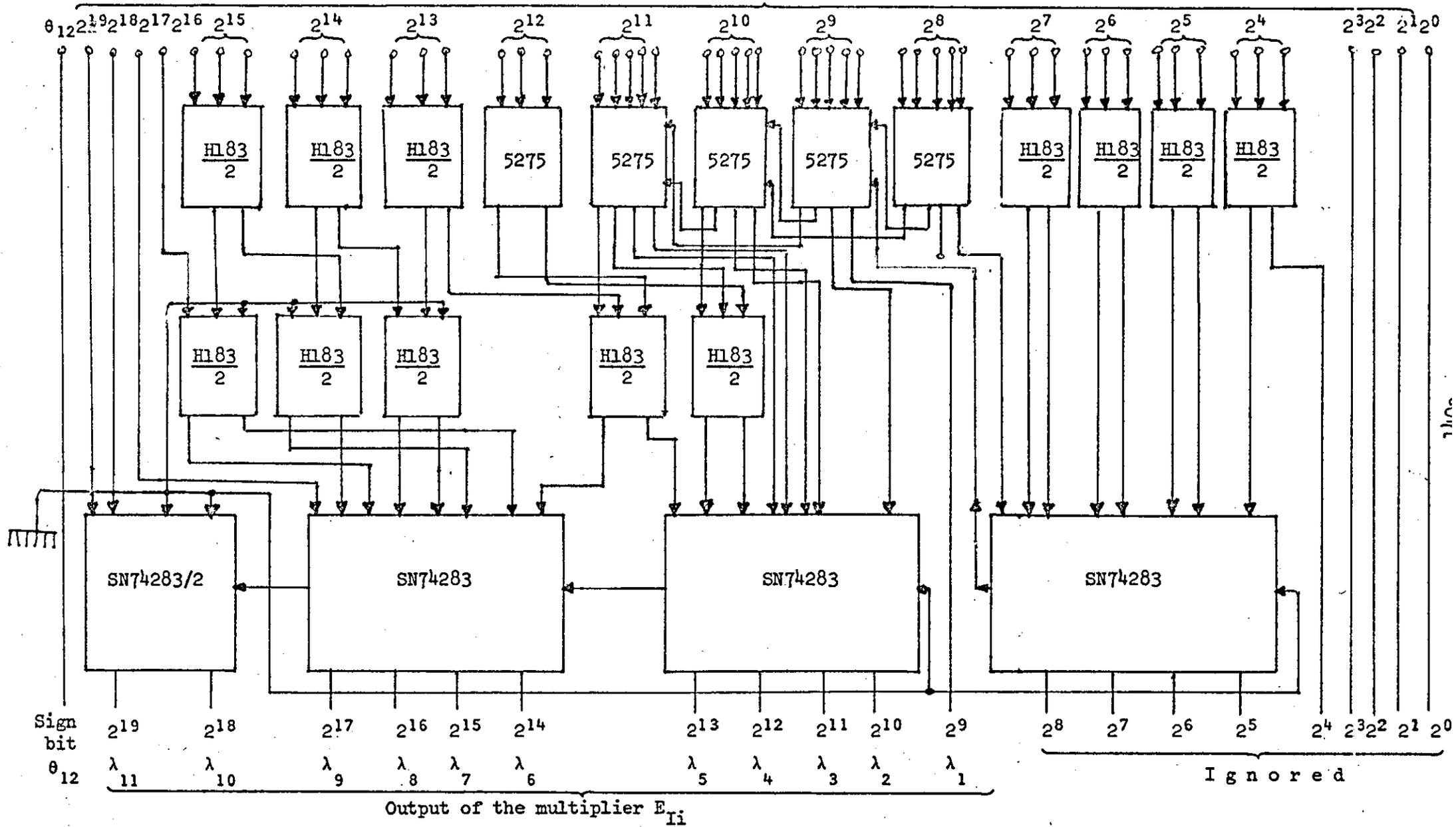


Figure 7.7(b) Texas 4 and 2 bit adders connected in Wallace trees arrangement to sum the partial product produced in Fig7.7a

in Appendix B.

"SUBROUTINE COMPARITOR" inspects  $X_L$ , the linear PCM equivalent of  $X_c$ , every  $T_p$  seconds ( $T_p = 1/f_p = 1/64000$  seconds). It also inspects the locally generated DSCALE step height  $H_{Di}$ , and forms the instantaneous difference  $E_{Di}$  between these two quantities, i.e.

$$E_{Di} = X_L - H_{D(i-1)}$$

Furthermore this instantaneous difference is operated upon to form the integrated error  $E_{Ii}$  which is defined as

$$E_{Ii} = E_{Di}(1 - e^{-T_p/T_1}) + E_{I(i-1)} \cdot e^{-T_p/T_1}.$$

The output of "SUBROUTINE COMPARITOR" is the polarity of the integrated error  $\lambda_{12}$ . It ( $\lambda_{12}$ ) is "returned" to the master segment every  $T_p$  seconds to generate the binary output  $\hat{L}_i$  of the converter ( $\hat{L}_i = \{0,1\}$ ). In the meantime  $\hat{L}_i$  is passed to a local DSCALE decoder which uses it to improve the estimation of the linear PCM sample. The content of the DSCALE  $H_{D(i-1)}$  at the previous clock instance is incremented or decremented, at the  $i^{\text{th}}$  clock instance, according to the DSCALE algorithm discussed in Chapter V (i.e.  $H_{D(i-1)}$  is incremented when  $\hat{L}_{i-2} = \hat{L}_{i-1} = \hat{L}_i$  and decremented otherwise). The local DSCALE decoder was implemented on the computer using "SUBROUTINE DIRECTION", "SUBROUTINE HISTORY", "SUBROUTINE AUCOUNTER" and "SUBROUTINE ADDER". These subprograms have already been discussed in earlier chapters and they can be found listed in Appendix B. "SUBROUTINE COMPARITOR" is also listed in Appendix B.

During each Nyquist interval, therefore,  $H_D$  is up-dated 8 times and 8 binary data digits are transmitted to the remote SCALE decoder to reconstruct the linear PCM sample. At the end of each Nyquist interval a new A-law PCM sample  $X_c$  is read in and the cycle repeats.

An alternative test program, which bypasses the expander, accepts the baseband signal in analogue form, applies it to an Analogue-to-

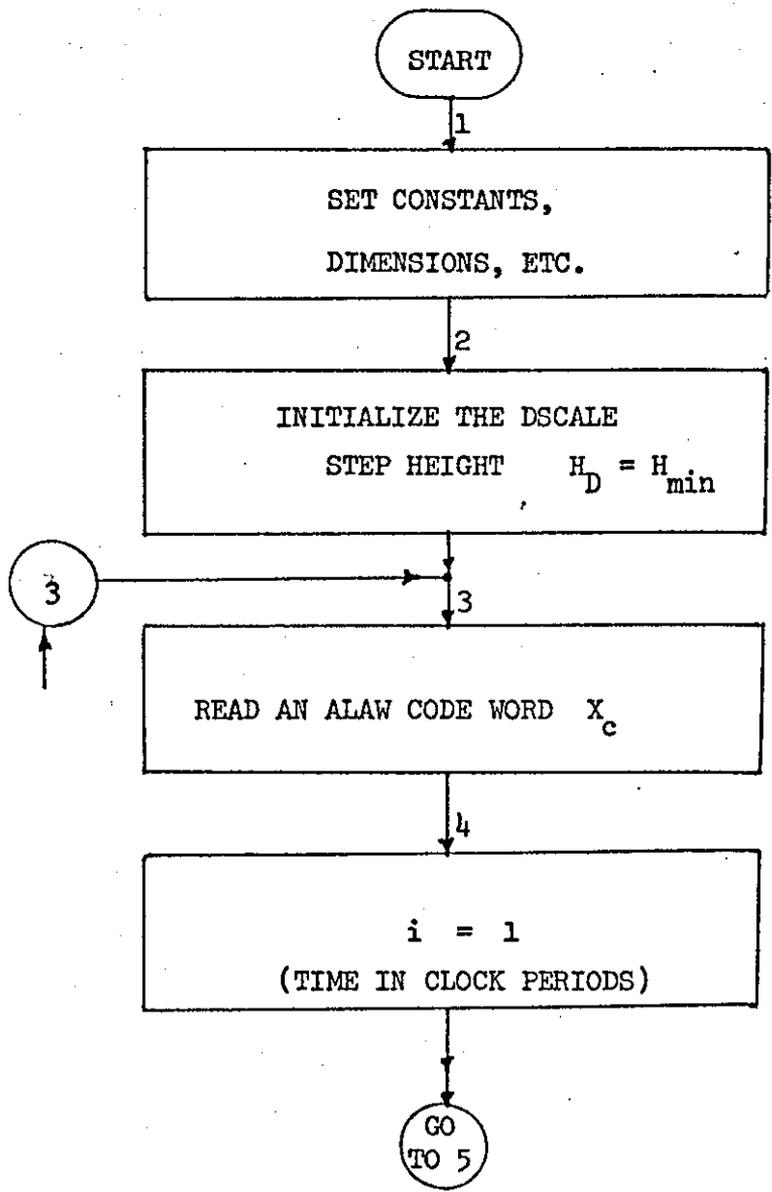


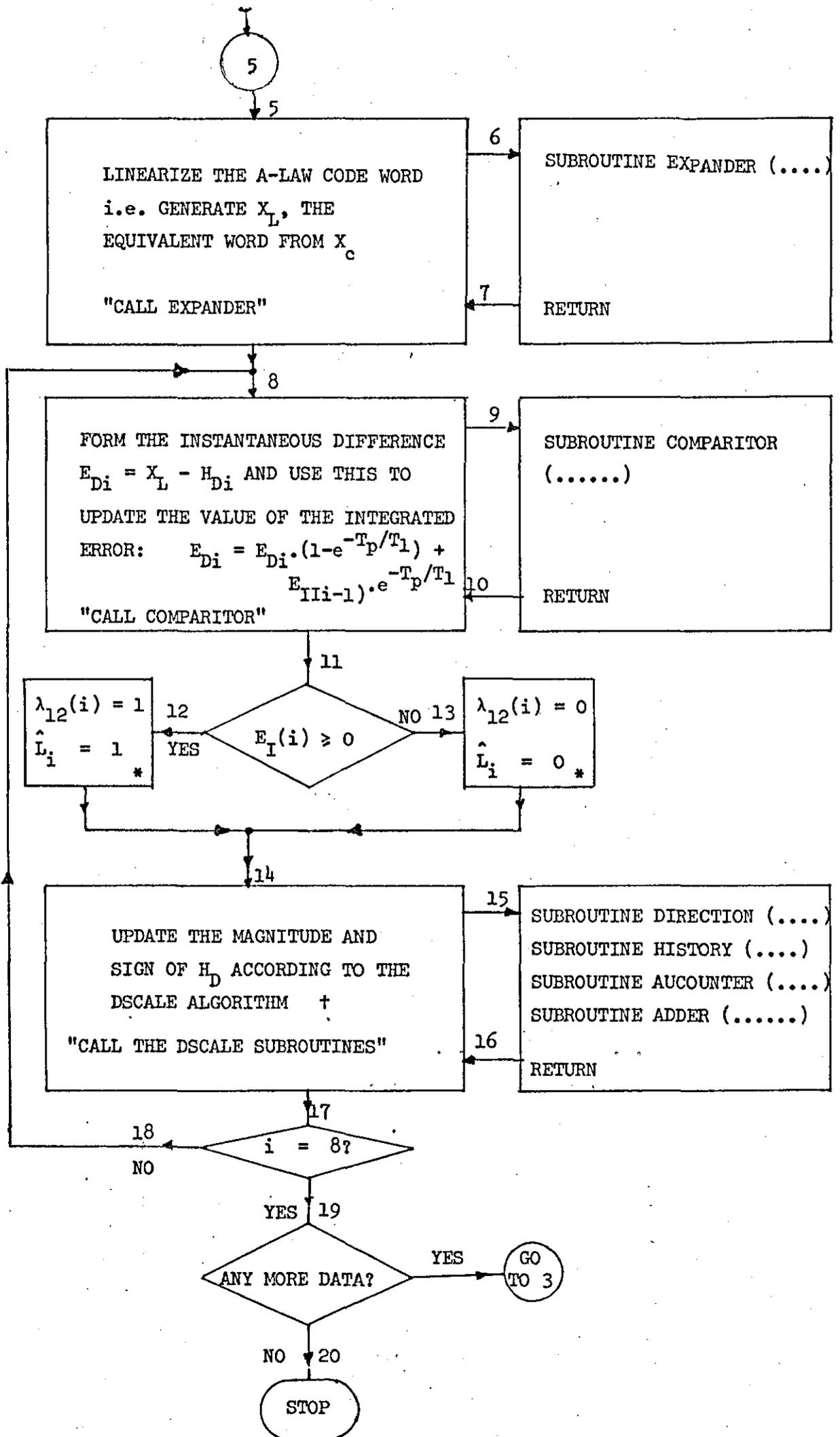
Figure 7.8 Flow diagram of the computer program for the simulation of the A-law PCM to SCALE converter.

\* = See Figure 7.2

$\lambda_{12}$  = The polarity of the integrated error  
 $E_{Ii} = X_L - H_D$  at the  $i^{\text{th}}$  DSCALE clock instance

$\hat{L}_i$  = The output digit of the A-law to SCALE converter

† = See Chapter V



Digital Converter ADC and applies the resultant linear PCM words to the "SUBROUTINE COMPARITOR". A subprogram was written to simulate the ADC and was named "SUBROUTINE ATODCONV". It can be found listed in Appendix B. The significance of this approach is that the system can be used as a completely digital SCALE encoder or a LPCM-to-SCALE converter.

## 7.5 SUMMARY AND RESULTS

When an analogue signal  $x_2(t)$  is applied to an A-law PCM encoder the compressed code  $X_c$  is produced. At the converter end,  $X_c$  is linearized by the expander to produce the linear PCM code  $X_L$ . The local digital SCALE decoder (which forms a part of the converter) generates the digital signal  $H_{Di}$  which is an estimate of  $X_L$  and has an analogue equivalent  $\hat{H}(t)$ . Both the local and the remote DSCALE decoders outputs  $H_{Di}$  (or  $\hat{H}(t)$ ) depend on the present and all previous values of the difference between  $X_L$  and  $H_D$ . When  $\hat{H}(t)$  is passed through a low pass filter  $\tilde{x}_2(t)$ , which is an estimate of  $x_2(t)$ , is obtained.

The performance of the system can be investigated qualitatively by comparing the input  $x_2(t)$  to the A-law encoder and the final output  $\tilde{x}_2(t)$  at the remote DSCALE decoder filter. This was achieved by simulating the system using the procedure discussed above in Section 7.4, and using the computer graph plotting routines to display the results in graphical form.

A sinusoidal signal was first used as the original waveform  $x_2(t)$ . In the second and third runs of the program  $x_2(t)$  was respectively, an arbitrary and step waveform.

The results of these computer runs corresponding to the cases when  $x_2(t)$  was a sinusoidal and arbitrary waveform are displayed in Figures 7.9 and 7.10, respectively. Part (a) of each of these figures

shows the original waveform  $x_2(t)$ ; while parts (b), (c) and (d) are respectively devoted to the display of the corresponding signals  $X_L$  (after D/A conversion),  $\hat{H}(t)$  and  $\tilde{x}_2(t)$  (superimposed on  $x_2(t)$  for comparison).

The results for the cases when  $x_2(t)$  were steps are shown in Figure 7.11.

From Figures 7.9(d) and 7.10(d) it can be seen that when an input is initially applied to the remote DSCALE decoder, the reconstructed baseband signal  $\tilde{x}_2(t)$  is distorted. This is an inherent characteristic of the SCALE system and is due to the delay introduced by the syllabic time constant. After this initial transient period (20 to  $100T_p$  seconds), however, it can be seen that  $\tilde{x}_2(t)$  is a good approximation of the original signal  $x_2(t)$ .

If the decoded signal  $\tilde{x}_2(t)$  in Fig. 7.9(d) is compared with  $\hat{x}_2(t)$  (which was constructed by the same DSCALE decoder from  $L(t)$  originating from a hybrid SCALE encoder operating directly on the analogue signal  $x_2(t)$  - see Chapter V) it can be seen that  $\tilde{x}_2(t)$  is a better representation of  $x_2(t)$  than  $\hat{x}_2(t)$ . This is mainly due to the fact that the local DSCALE decoder employed in the converter was tracking a signal  $X_L$  which is composed of discrete levels. Because the changes in such a signal take place instantaneously, the density of consecutive like digits in the  $\hat{L}(t)$  pattern is less than that of  $L(t)$ . This means that the total time during which the SCALE decoder is overloaded when it is decoding  $\hat{L}(t)$  is less than that when  $L(t)$  is the signal being decoded (see Fig. 7.12).

From this argument it can be concluded that the SCALE system performance can be improved if the analogue input signal  $x_2(t)$  is sampled and held at Nyquist's rate prior to its application to the encoder; instead of encoding  $x_2(t)$  directly.

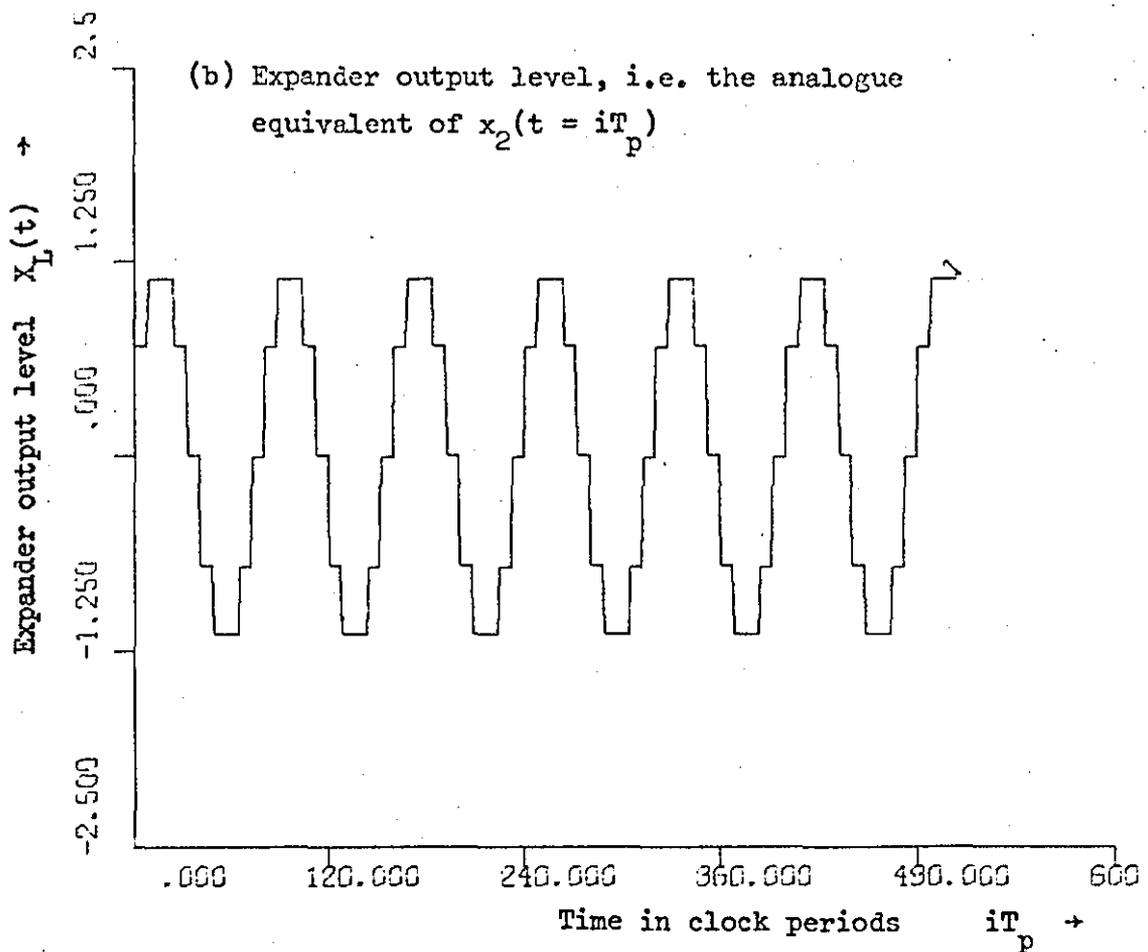
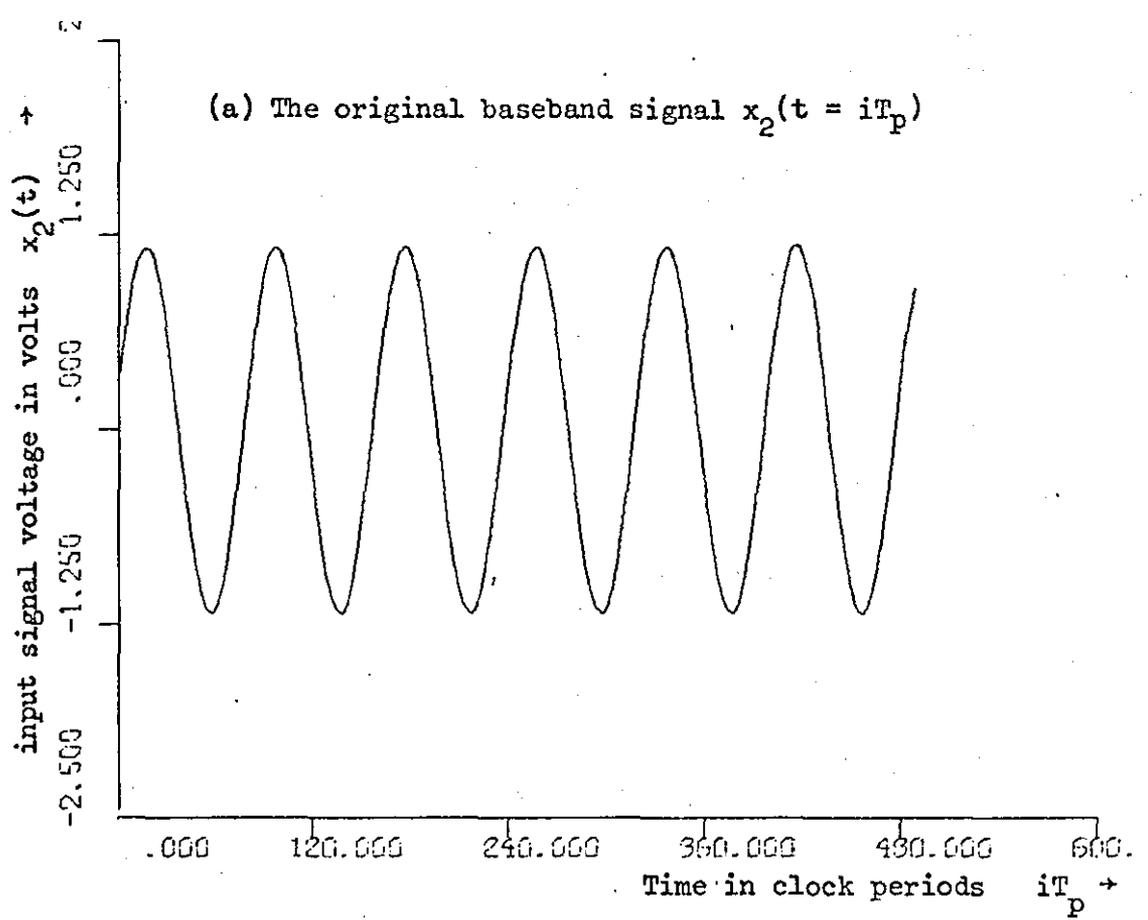


Figure 7.9 Waveforms of the original baseband signal  $x_2(t)$  and the corresponding LPCM PAM signal at the output of the expander

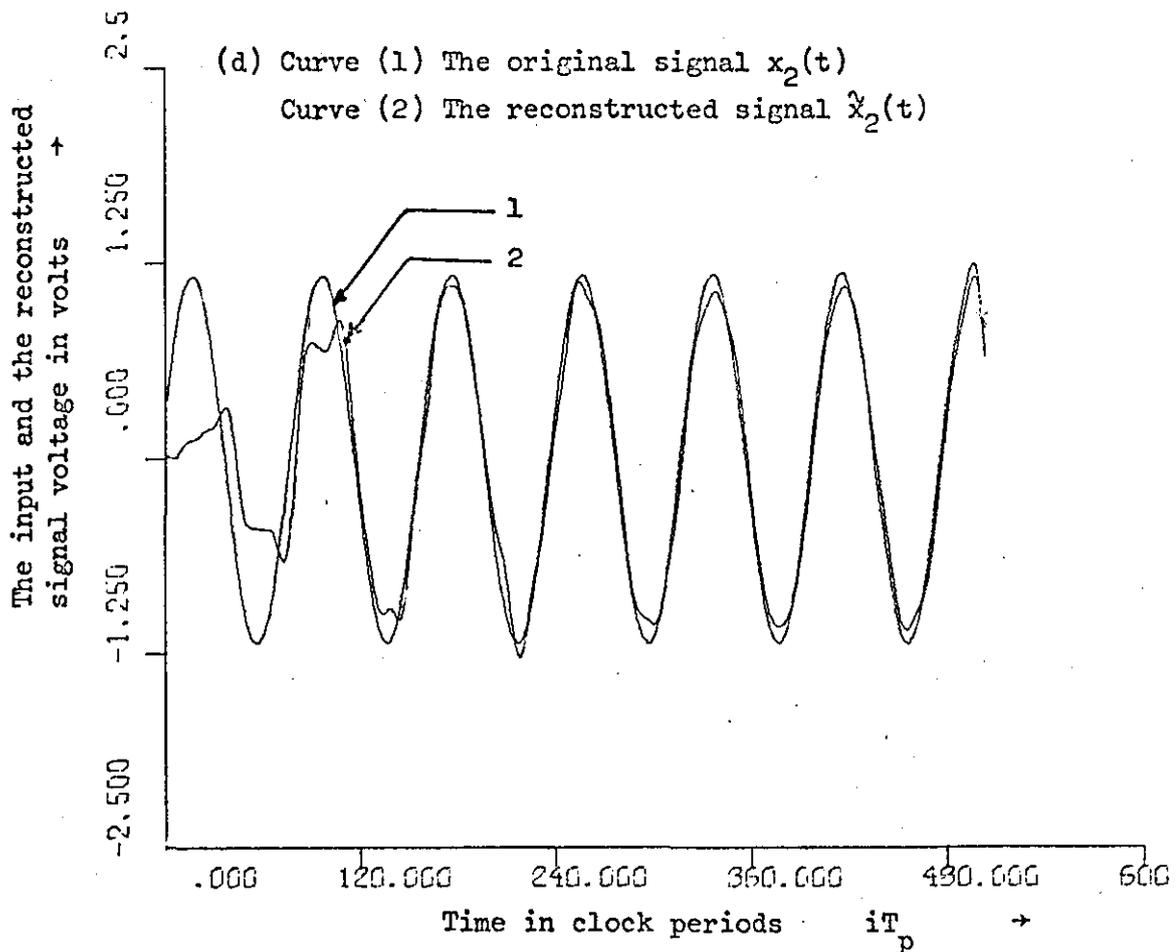
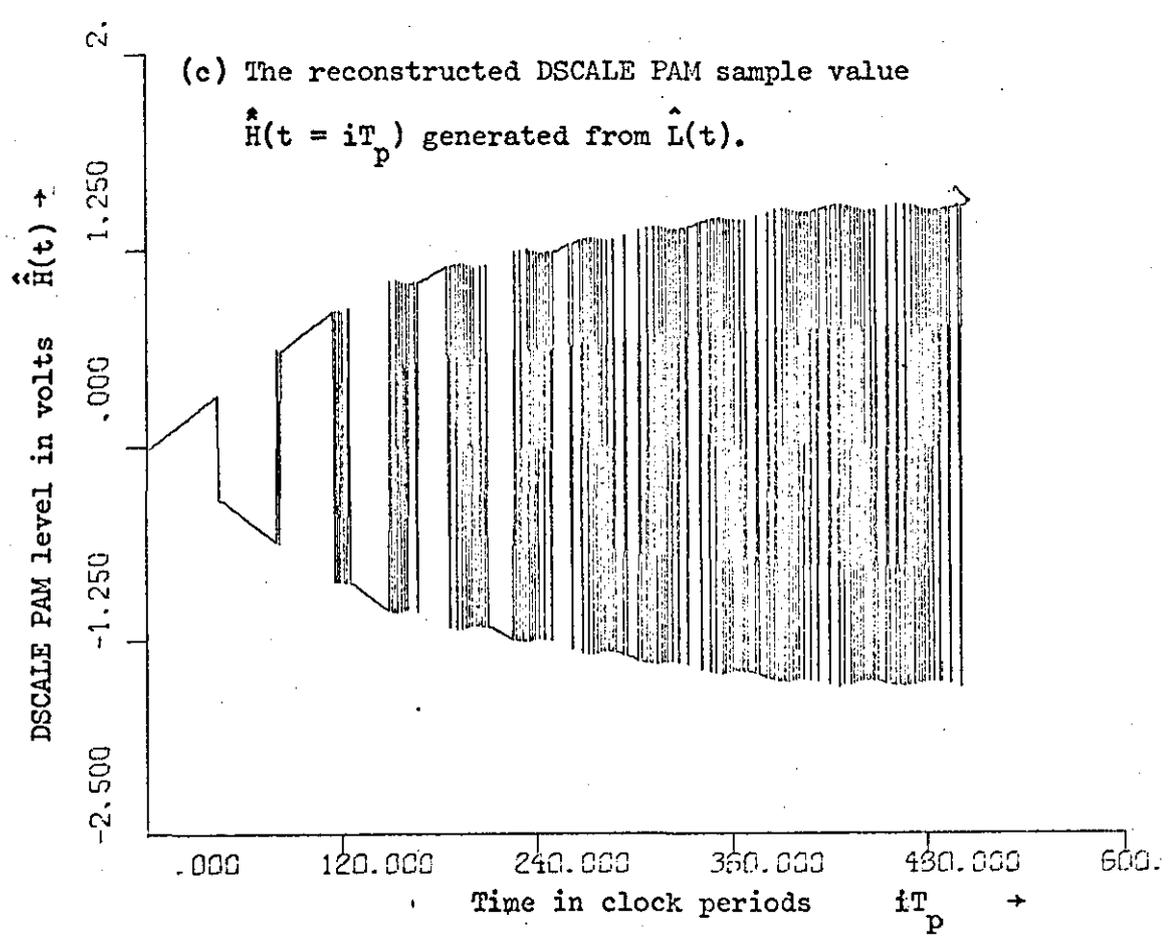


Figure 7.9 The reconstructed SCALE PAM signal  $H(t)$  and its corresponding low pass contents. The original signal is shown in part (d) curve (2).

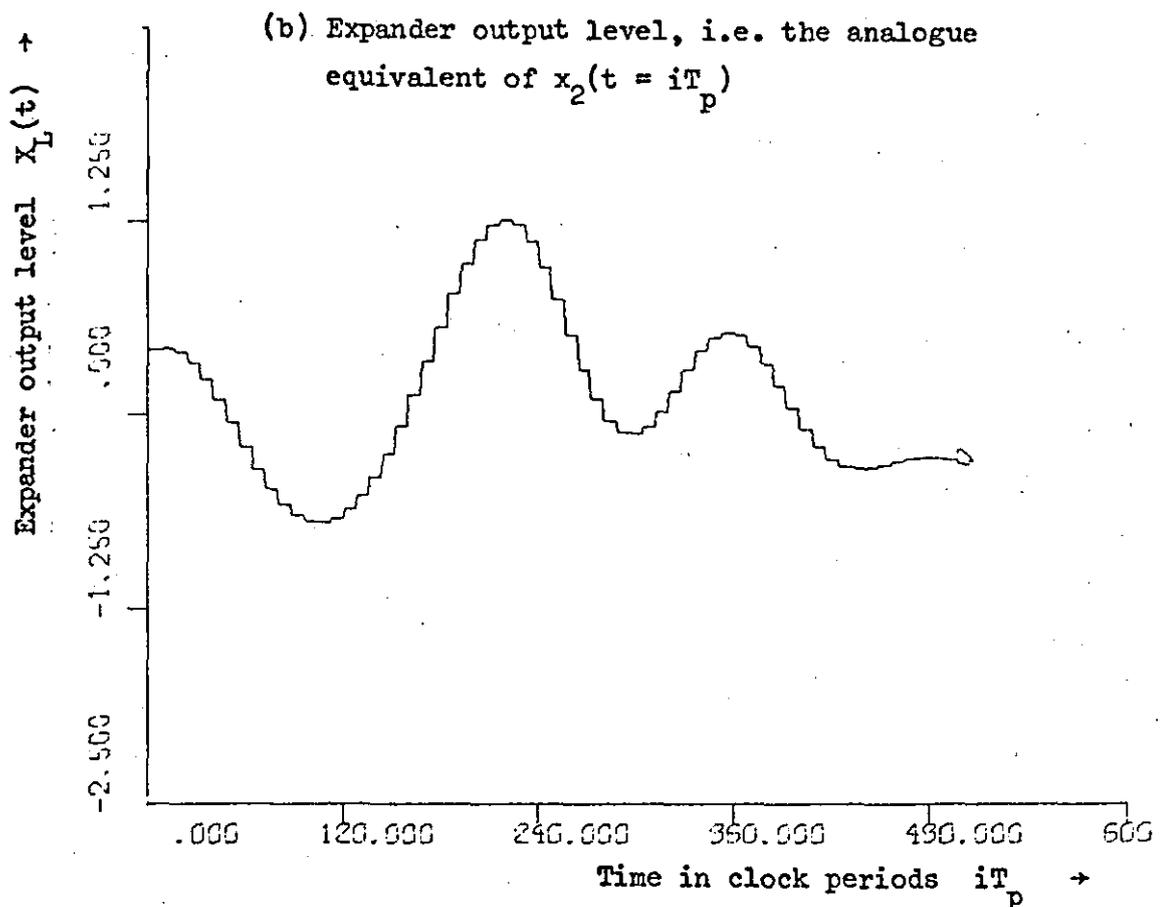
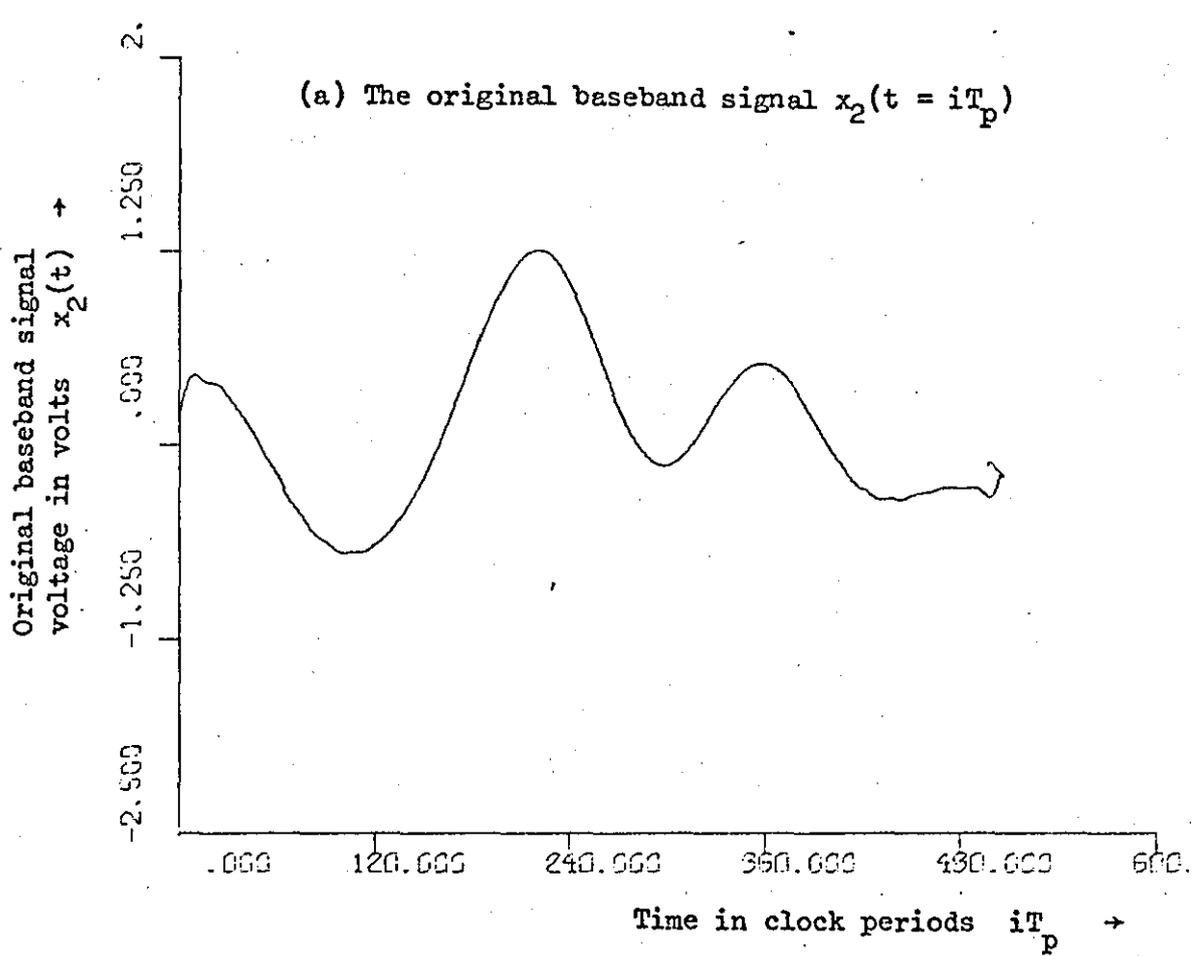


Figure 7.10 Waveforms of the original baseband signal  $x_2(t)$  and the corresponding LPCM PAM signal at the output of the expander

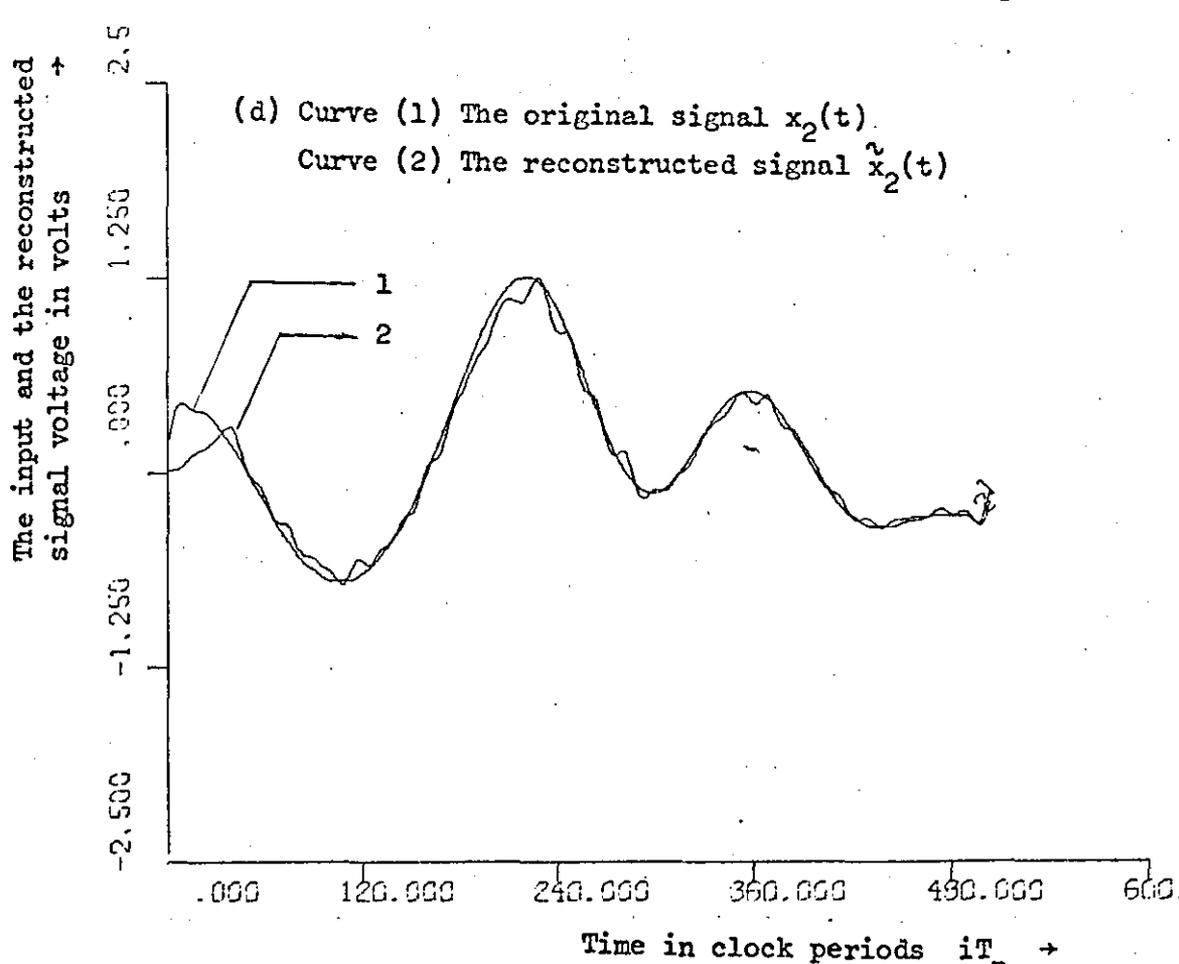
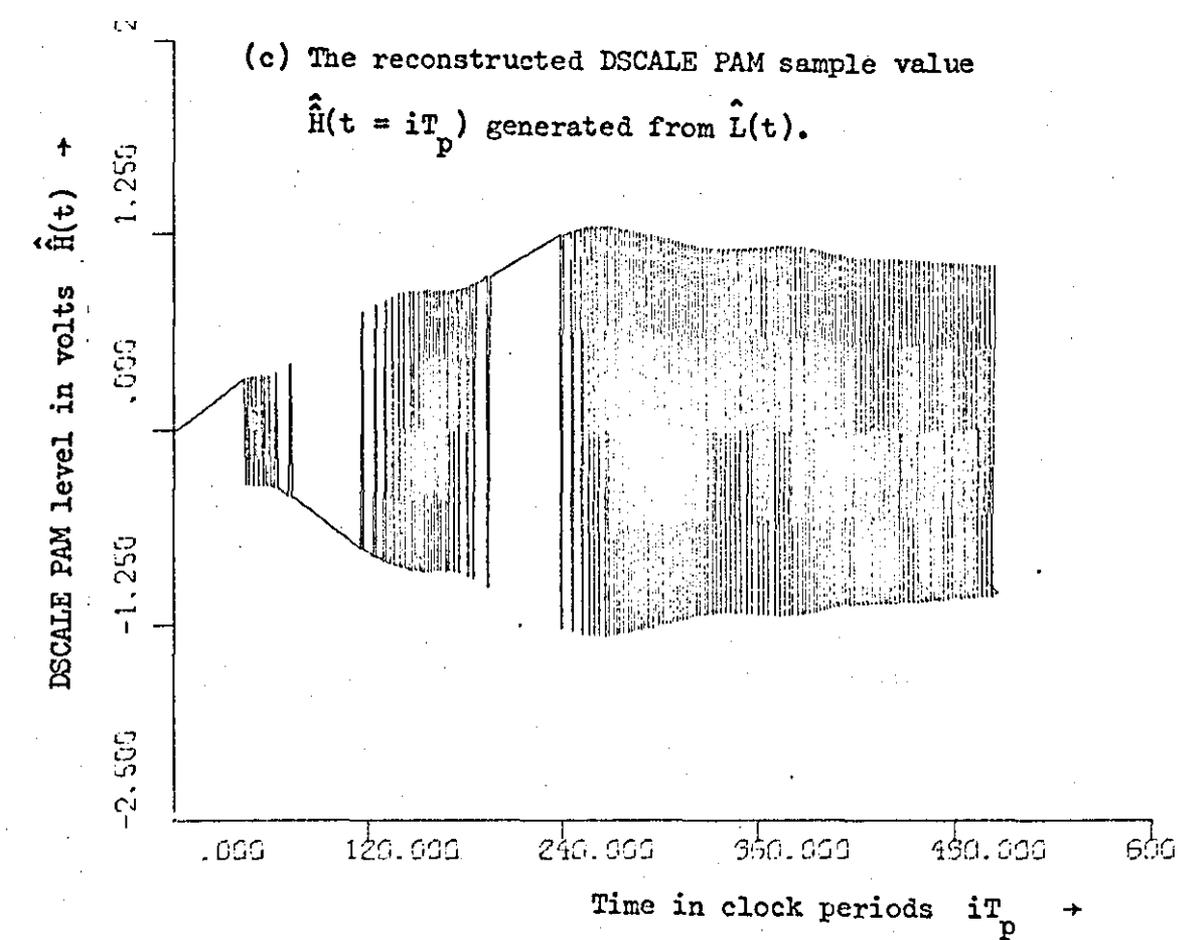


Figure 7.10 The reconstructed SCALE PAM signal  $H(t)$  and its corresponding low pass contents. The original signal is shown in part (d) curve (2).

The graphs displayed in Figures 7.11(a) and 7.11(b) show how the system responds to step inputs when  $x_2(t)$  is encoded into A-law PCM form prior to the generation of the SCALE binary data  $\hat{L}(t)$  by the A-law - to - SCALE converter. Comparison of Figures 7.11 with Figure 5.11a reveals the close agreement between  $\hat{H}(t)$  and  $\hat{\hat{H}}(t)$  when the original signal  $x_2(t)$  is a step function.

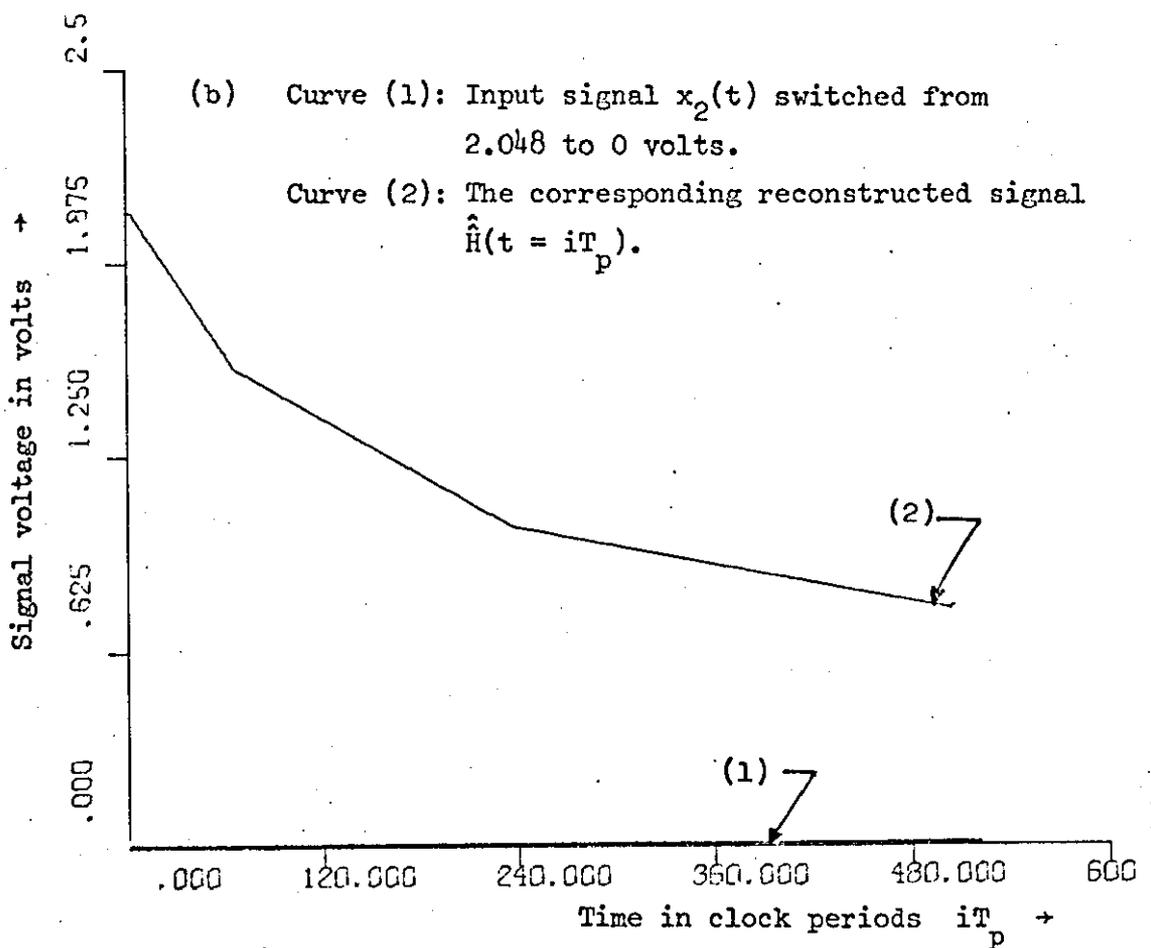
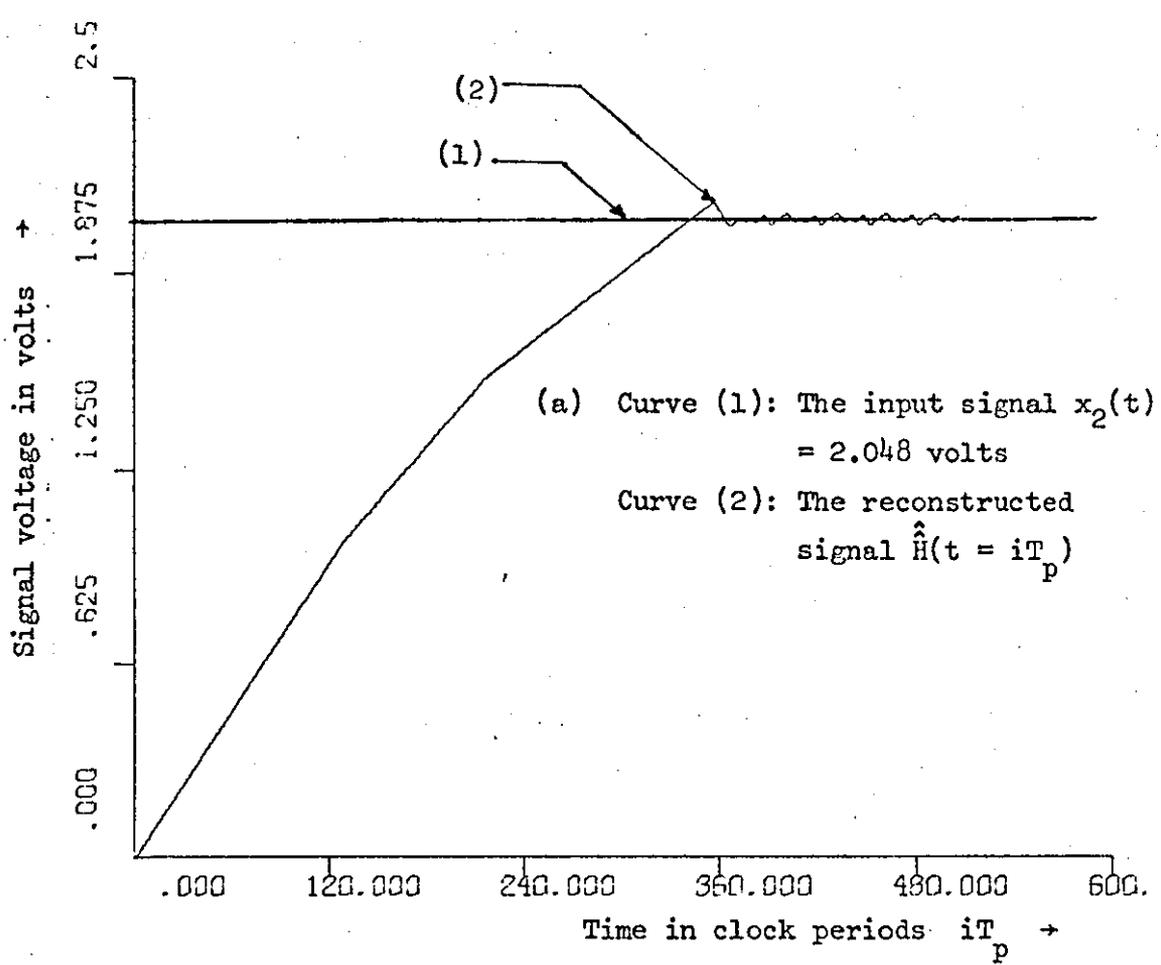


Figure 7.11 Step response of the converter followed by a SCALE decoder.

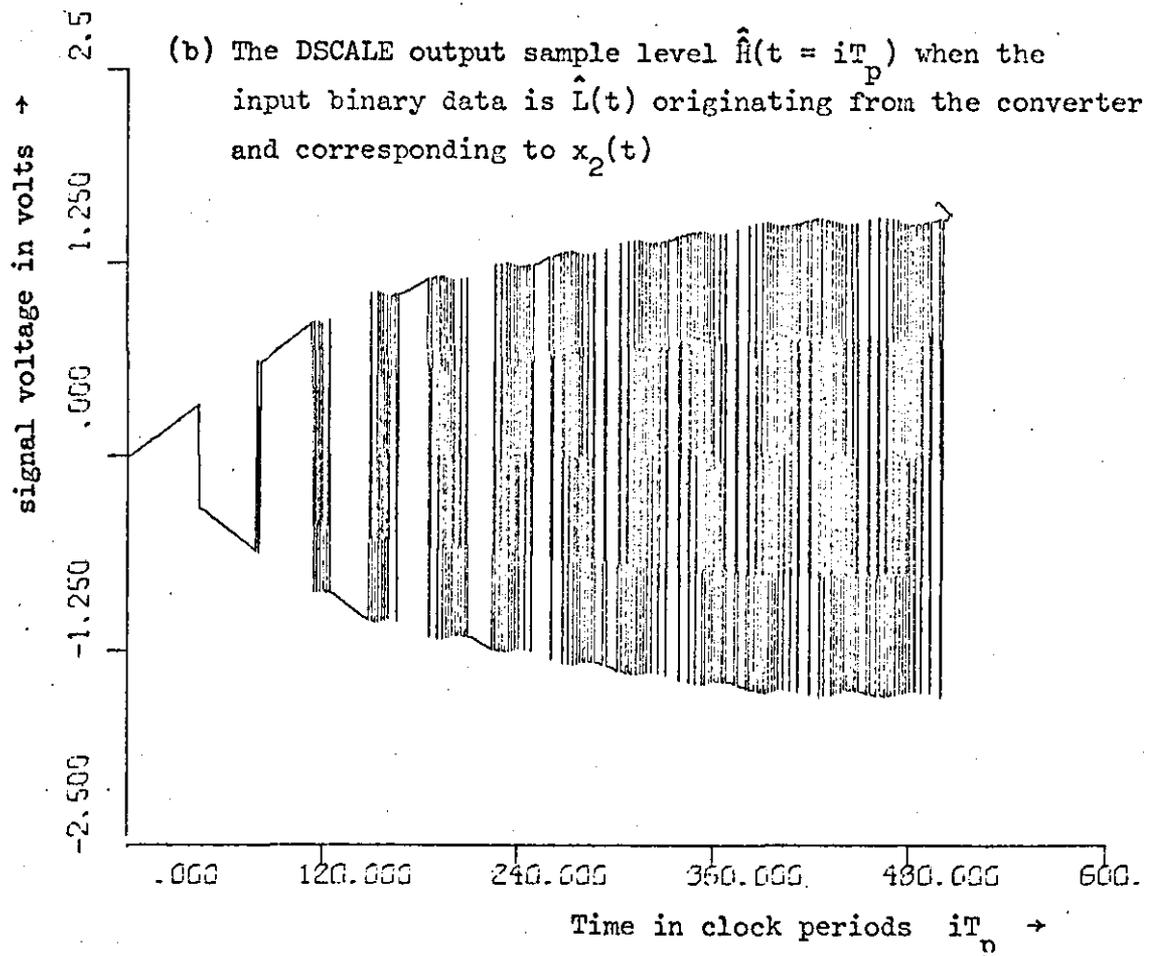
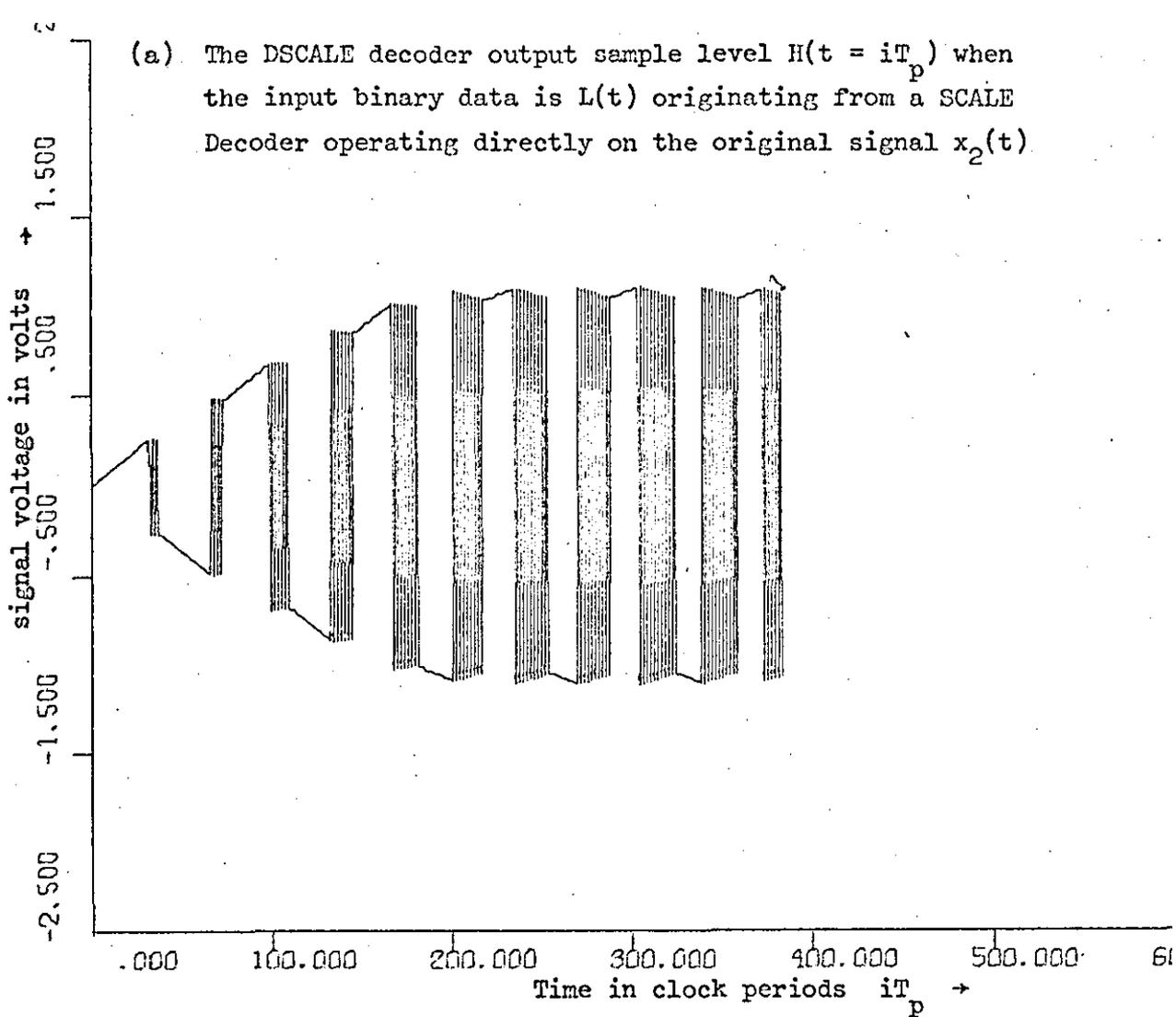


Figure 7.12 Comparison between  $\hat{H}(t)$  and  $\hat{H}(t)$

## CHAPTER VIII

### GENERAL DISCUSSION AND CONCLUSIONS

#### 8.1 GENERAL DISCUSSION

When this work was initiated, the A-law Pulse Code Modulation system had already been agreed internationally as the digital method of interexchange speech transmission. For military and local network applications, however, there was a considerable support for using some form of Delta Modulation as an alternative to PCM. This support was reflected in the investments made by both the military and civilian organizations into research and development of various Delta Modulation systems.

The main factor in favour of Delta Modulation systems then was the simplicity of their implementation; and the main argument against them was their high sampling rates.

The fact that the Delta Modulation system has an inherent immunity to channel noise did not carry much weight because all the telephone traffic was then carried by transmission lines or line of sight high grade microwave channels.

As the work in this project progressed other considerations came to light. For example, during the last two years many new radio channels have been integrated into the telephone network. These radio channels have been, in most cases, installed to establish communication links across difficult terrain<sup>(93)</sup> and are characterized by fading and relatively poor channel error performance. Examples of such channels, which employ a Troposcatter transmission mode, are those established across the Mediterranean and those employed between the United Kingdom and the North Sea oil fields. Tests carried out on these radio channels have indicated that error probabilities of the order of  $10^{-3}$  are not uncommon<sup>(94)</sup>.

This means that in such cases A-law PCM (which, for satisfactory performance, requires a maximum error probability of  $10^{-5}$ ) cannot be used and an alternative has to be found.

To telecommunications engineers in the Third World countries, such as the Middle East and South American countries, the points discussed above are of importance; because:-

(a) The communities in those countries tend, in most cases, to be separated by large distances of difficult terrains. Consequently - for economical reasons - most of the trunk traffic is carried by radio channels of the type discussed above. Hence the channel error problem.

(b) Even if a solution to the geographical problems is found, the large capital investments which have been made in radio communication systems prohibit their phasing out before at least 20 to 30 years.

(c) Since no capital investment has so far been made by the majority of those countries in any form of digital communication systems, then if they have to change over from analogue to digital techniques, the digital format to be adopted can be chosen exclusively on the basis of its suitability to the available communication channels.

Comparisons between the performance of the SCALE system (or its Delta modulation versions) with other digital formats (A-law PCM, DPCM, etc.), have indicated that the SCALE system has an immunity to channel noise which is superior to any of the other systems.

In addition, because it is relatively simpler to implement, the SCALE system is more suitable for per channel bases applications; a characteristic which is desirable for private and mobile communications, and in isolated areas of low density of population.

In the light of all the above it seemed probable that A-law PCM and the SCALE system are going to co-exist for a considerable time in the future. As a result, in order to provide direct compatibility, it is desirable that SCALE - to - A-law PCM and A-law PCM - to - SCALE conversion techniques are established.

## 8.2 CONCLUSIONS

The need for a conversion technique to enable direct compatibility between A-law PCM and the SCALE system was the motivation which originated this work.

SCALE has been in service for some years, but a theoretical description of it has been lacking. This study began with an investigation in which detailed experimental, theoretical and computer simulation of SCALE have been carried out. This has enabled a general model of the hybrid SCALE system to be developed, and formulae for its performance, in terms of sampling frequency, dynamic range, memory length, syllabic time constant and signal-to-noise ratio to be presented.

The results of the investigation into the hybrid SCALE system have shed a new light on the exact system behaviour, under various operating conditions.

With the aid of digital computer and the knowledge gained from the investigation into the hybrid SCALE system, an algorithm for the design of a new all-digital Delta-sigma decoder with digital syllabic companding has been established. It has also been shown by computer simulation that the new digital SCALE decoder can decode the binary data, originating from an ordinary hybrid SCALE encoder, and the quality of the reconstructed baseband signal can be at least as good as that obtainable from a hybrid SCALE decoder.

The main significance of the introduction of the new all-digital SCALE decoder, is that because an equivalent of the hybrid SCALE encoder feedback voltage is now available in digital form, it can be digitally filtered to produce the linear digital equivalent of the original analogue input signal. This means that the difficult problem of converting the SCALE binary data to other compressed digital formats has now been reduced to finding an algorithm which enables the linear digital representation of the original analogue signal to be digitally compressed according to the specified requirements.

These results have led directly to the design of an all-digital SCALE - to - A-law PCM converter which accepts the SCALE binary data and produces A-law PCM words. The converter has been computer simulated; and it has been shown that the performance of the combination of the SCALE encoder and the A-law PCM receiver, including the converter, can produce a signal-to-noise ratio performance which is very similar to that obtainable from the SCALE encoder-decoder combination on their own.

A design procedure for the realization of an A-law PCM - to - SCALE converter has also been proposed. This converter accepts A-law PCM words and produces SCALE binary data. This has concluded the search for the complete compatibility between the A-law PCM and the SCALE system; a compatibility which has been lacking.

It has also been demonstrated by computer simulation that the proposed all-digital A-law PCM - to - SCALE converter can generate from an A-law PCM coded signal a SCALE binary data stream which when decoded by a SCALE decoder is a close replica of the original analogue signal.

To demonstrate the fact that the conceived converters can be realized by small and medium scale integrated circuits, all the necessary circuit diagrams and data necessary for their hardware implementation have been presented. Because of the time limit imposed on the period

which was available for this research, it was not possible to carry out the actual construction of these converters. However, since only digital circuits are going to be employed in the construction, it is expected that the hardware versions of the converters will produce results which are identical to those obtained from the simulated models.

The converters developed in this work may have applications in both the military and civilian communication systems. For example in areas where the channel noise is severe the vulnerable A-law PCM codes can be converted to SCALE binary data format for transmission and reconverted back to A-law PCM at the remote end of the channel. Another situation where the converters may prove to be useful is when the military SCALE and commercial A-law PCM users have to communicate directly with one another. A third application which in the long run may become the most important, is in the local subscriber network where the SCALE type system can be used at the handset and the converters at the exchange.

It must be emphasised that in the absence of transmission errors, the performance - in terms of SNR - of the present 8-digit A-law PCM is better than the SCALE systems operating at the same bit rate. This means that systems employing the converters discussed in this thesis will have a SNR which is less than that of the conventional A-law PCM system. However the availability of these converters have the advantages of enabling the users of SCALE and the A-law PCM to communicate directly with each other, and the conversion of the A-law signal to SCALE for transmission over noisy channels.

It may also be worth emphasizing that the conversion techniques employed in this work are applicable to converting between the A-law PCM and other versions of Deltamodulation. For example the double integration SCALE system which is in the development stage<sup>(101)</sup> has a SNR which is comparable with that of A-law PCM when the two systems operate at 64 Kb/s.

With the aid of the conversion techniques discussed in this thesis, it is possible to design digital converters which can provide direct compatibility between A-law PCM and the new double integration SCALE system. Hence the problem of SNR incompatibility between the two systems can be solved.

Another point which is worth mentioning is that in the future the switching centres are likely to become completely digital. This means that the number of the digital-analogue-digital conversion operations carried out along the transmission path will be reduced to a minimum. Consequently the high SNR performance presently specified for digital speech transmission will no longer be necessary and a lower performance is likely to become acceptable. In this case the SCALE system operating at 64 Kb/s (or even at lower bit rates) will no longer be at a disadvantage.

## REFERENCES

- (1) CATTERMOLE, K. W. Principles of Pulse Code Modulation, Iliffe Books, 1969
- (2) STEELE, R. Delta Modulation Systems, Prentice-Hall Book Co., 1974
- (3) SCHWARTZ, M. Information Transmission, Modulation and Noise - Chapter 3, McGraw-Hill Book Co., 1959
- (4) PANTER, P. F. Modulation, Noise and Spectral Analysis - Chapter 18, McGraw-Hill Book Co., 1965
- (5) LATHI, B. P. Random Signals and Communication Theory - Chapters 5 & 6, Intertext Books, 1968
- (6) LENKURT, G. T. E. PCM - Concepts, Developments and Potential, Telecommunications, September, 1975
- (7) BUCCI, W. PCM: A Global Scramble for Systems Compatibility, Electronics, June 23, 1969
- (8) DE JAGER, F. Delta Modulation, A Method of PCM Transmission using the One-Unit Code, Philips Research Report No. 7, 1952
- (9) BELL TELEPHONE LABS STAFF Bell Telephone Lab. Staff. Transmission Systems for Communication (Book)
- (10) MAN, H., STRAUBE, H.M., and VILLARS, C.P. A Companded Coder for an Experimental PCM Terminal, B.S.T.J. Vol. 41, 1962
- (11) SHENNUM, R. H. and GREAY, J. R. Performance Limitations of Practical PCM Terminals, B.S.T.J. Vol. 41, 1962
- (12) C.C.I.T.T. Draft Recommendation G.702 for 30/32 Channels PCM
- (13) BOXALL, F. S. A Digital Carrier-Concentrator System with Elastic Traffic Capacity, IEEE Conf. Rec. Int. Conf. on Communications, June, 1974
- (14) MCNAIR, I. M. Subscriber Loop Multiplexer - A High Pair Gain System for Up-Grading and Growth in Rural Areas, IEEE Trans. Com. Vol. 19 No. 4, 1971
- (15) PETFORD, B and CLARKE, C. M. A Companded Delta-Sigma Speech Digitizer, IEE Conference on Signal Processing Methods for Radio Telephony, No. 64, May 1, 1970

- (16) WILKINSON, R. M.           Speech Coding in Digitally Switched Networks, IEE Conference Publication No. 64, May, 1970
- (17) WILKINSON, R. M.           A Companded Delta-Sigma Modulation System, S.R.D.E. (Ministry of Technology) Report No. 68012, July, 1968
- (18) WILKINSON, R. M.           Delta Modulation Techniques for Analogue to Digital Conversion of Speech Signals, S.R.D.E. Report No. 69022, April, 1969
- (19) NWESSRY, M. M. and        Performance of the SCALE CODEC, IEE Colloquium on Computer Simulation of Communication Systems, May, 1975
- STEELE, R.
- (20) GREEFKES, J. A. and        Code Modulation with Digitally Controlled Companding for Speech Transmission, Philips Tech Rev. Vol. 31, 1970
- RIEMENS, K.
- (21) ZIEKMAN, C. and            Deltamax: A Design Element for Military Communication Networks, Philips Telecomms. Rev. Vol. 32, July, 1974
- ZWAAL, P.
- (22) DAVID, G.                   Digital Airborne Equipment for AEROSAT, Philips Telecomms. Rev. Vol. 32, Dec. 1974
- (23) KWAN, R. K.                 Television and Radio Program Distribution via Satellite in Canada, International Conf. on Satellite Communication Systems Technology, IEE Conf. Publication No. 126, 1975
- (24) GREENSTEIN, L. J.         Slope Overload Noise in Linear Delta Modulators with Gaussian Inputs, B.S.T.J., March, 1973
- (25) STEELE, R. and             Slope Limiter Model of a Delta Modulator in Slope Overload with Gaussian Input Signals, IEE/IRE Joint Conf. on Digital Processing of Signals in Communication, University of Technology, Loughborough, April, 1972
- PASSOT, M.
- (26) PROTONOTARIOS, E. N.      Slope Overload Noise in Differential Pulse Code Modulation Systems, B.S.T.J., Nov. 1967
- (27) O'NEAL, J. B.             Delta Modulation Quantization Noise - Analytical and Computer Simulation Results for Gaussian and Television Signals, B.S.T.J., Jan. 1966

- (28) IWERSEN, J. E. Calculated Quantization Noise of Single Integration Delta-Modulation CODERS, B.S.T.J., September, 1969
- (29) PASSOT, M. and STEELE, R. Application of Normal Spectrum Technique to the Calculation of Distortion Noise in Delta Modulators, The Radio and Electronic Engineer, Vol. 44 No. 10, October, 1974
- (30) GOTZ, B., CHING, Y.C. and BALDWIN, G. L. Idle Channel Noise Characteristic of a Delta Modulator with Step Imbalance and Quantizer Hysteresis, IEEE Trans. on Communications, Vol. Comm. 23, No. 9, 1975
- (31) WINKLER, M. R. High Information Delta Modulation, IEEE Conversion Record Part 8, pp 260-265, 1963
- (32) KYAW, A. T. and STEELE, R. Constant Factor Delta Modulator, Electronic Letters, 22nd February, 1973
- (33) GREEFKES, J. A. and DE JAGER, F. Continuous Delta Modulation, IEEE Int. Conv. Rec. 11 No. 8, 1963
- (34) CLUTIER, M. and DECHENES, P. Modulateurs Delta: Compressions Analogique et Numerique, Genie Electrique, Université de Sherbrook, Canada
- (35) CARTMALE, A. A. and STEELE, R. Calculating the Performance of Syllabically Companded Delta-Sigma Modulators, Proc. IEE Vol. 17, 1970
- (36) BOSWORTH, R. H. and CANDY, J. C. A Companded One-Bit Coder for Television Transmission, B.S.T.J., May, 1969
- (37) BRALIN, S. J. and BROWN, J. M. Companded Delta Modulation for Telephony, IEEE Trans. on Communication Technology, February, 1968
- (38) ZETTERBURG, L. H. A Comparison between Delta and Pulse Code Modulation, Ericson Technics, Vol. 11, 1955
- (39) HAUSER, K. J. and ZARDU, S. J. The Design of Digitally Controlled Delta Modulation Coders, Proc. IREE, Australia, July, 1971
- (40) NOLL, D. P. Colloquium on Differential Quantizing Systems, Loughborough University, 3rd October, 1975

- (41) SCHINDLER, H. R. Linear, Nonlinear and Adaptive Delta Modulation, IEEE Trans. on Comm., Nov. 1971
- (42) JAYANT, N. S. Digital Coding of Speech Waveforms: PCM, DPCM and DM Quantizers, Proc. IEEE, May, 1974
- (43) PURTON, R. F. A Survey of Telephone Speech Signal Statistics and their Significance in the Choise of a PCM Companding Law, Proc. IEEE, 109, Part B, 1962
- (44) BOXALL, F. S. Time Shared Codecs for Multichannel Delta-Modulation and DPCM, IEEE Trans. on Communications, Vol. Comm. 23, March, 1975
- (45) SCHINDLER, H. R. Delta Modulation, IEEE Spectrum, Oct. 1970
- (46) INOSE, H. YASUDA, Y. and MURAKAMI, H. A Telemetry System by Code Modulation Delta-Sigma Modulation, IRE Trans. on Space Electronics and Telemetry, September, 1962
- (47) INOSE, H. and YASUDA, Y. A Unity Bit Coding Method by Negative Feedback, Proc. IEEE, Vol. 51, 1963
- (48) NNESSRY, M. Theory, Design and Performance of SCALE CODEC System, Departmental Memorandum No. 105, Department of Electrical Engineering, Loughborough University of Technology, England
- (49) JOHNSON, F. B. Calculating Delta Modulation Performance, IEEE Transactions on Audio and Electroacoustics, Vol. AU16, No. 1, March, 1968
- (50) MILLMAN, J. and TAUB, H. Pulse, Digital and Switching Waveforms, McGraw-Hill Book Co., 1965
- (51) JAYANT, N. S. Design and Implementation of an Adaptive Delta Modulator, Proc. IEEE, Int. Conf. on Comm., 1972
- (52) KYAW, A. T. Constant Factor Delta Modulation - A New Instantaneously Adaptive Delta Modulation System, Ph.D. Thesis, Loughborough University, 1973.
- (53) RABINER, L.R., GOLD, R. and MCGONEGAL, C. A. An Approach to the Approximation Problem for Non-Recursive Digital Filters, IEEE Trans. on Electroacoustics, Vol. AU18 No. 2, June, 1970

- (54) HOLM, J. N. Colloquium on Speech Analysis and Synthesis, Loughborough University, February, 1976
- (55) ACKROYD, M. H. Digital Filters (Book), Butterworth, London, 1973
- (56) ROBINSON Time Series Analysis, IEEE Trans. on Audio and Electronics, June, 1967
- (57) JUDD, M. and HOLM, J. Colloquium on Speech Analysis and Synthesis, Loughborough University, February, 1976
- (58) FLANAGAN, J. L. Voices of Men and Machines, The Journal of The Acoustical Society of America, 1972
- (59) FLOOD, J. E. and HAWKSFORD, M. J. Adaptive Delta-Sigma Modulation using Pulse Grouping Techniques, IEE Conference on Digital Signal Processing, May, 1972
- (60) BASKARAN The Computer Library, Loughborough University (The Computer Centre)
- (61) AARONSON, G. Active Filters: Part 11, Electronics, July 7, 1969
- (62) SHEPARD, R. R. Active Filters: Shortcut to Network Design, Electronics, August 18, 1969
- (63) TEXAS INSTRUMENTS, LTD. System 74, Digital Integrated Circuits
- (64) ORGANICK, E. I. A Fortran IV Primer, Addison-Wesley Publishing Co., 1967
- (65) MURRILL, P. W. and SMITH, C. L. Fortran IV Programming, International Textbook Co., 1968
- (66) MOON, B. A. M. Computer Programming for Science and Engineering, Butterworth, 1966
- (67) HOLM, J. N. Speech Synthesis (Book), Mills and Boon, 1972
- (68) CHEN, M. Low Bit Rate Coding of Speech Signals, Ph.D. Thesis, The Imperial College (London)
- (69) MITRA, S. K. Active Inductorless Filters, IEEE Press, 1971
- (70) SAVESKIE, P. N. Intelligibility, Consuletter International, Vol. 5 No. 1, 1976
- (71) NOTTINGHAM ALGORITHM GROUP Computer Library, The Computer Centre, Loughborough University, England

- (72) RABINER, L. R. and  
RADER, C. M. Digital Signal Processing, IEEE Press, 1972
- (73) MARCONI CO. LTD.  
(Chelmsford, England) Quantization Noise Meter, No. TF2343A
- (74) RUSTON, H. Electronic Networks Functions, Filter  
Analysis, McGraw-Hill, 1966
- (75) WEINBURGE, L. Network Analysis and Synthesis, McGraw-  
Hill, 1962
- (76) RICHARDS, D. L. Telecommunication by Speech, The Trans  
Performance of Telephone Networks (Book),  
Butterworth Book Co., 1973
- (77) MUNDAY, S. The Use of Companders in the International  
Telephone Network, IEE Conference Publication  
No. 64, May, 1970
- (78) GOODMAN, D. J. The Application of Delta Modulation to  
Analogue-to-PCM Encoding, Bell System Tech.  
Journal, February, 1969
- (79) PENNINGTON, H. and  
STEELE, R. A-Law Pulse Code Modulation by a Delta-Modula-  
tion Technique, Electronic Letters, May, 1973
- (80) ISHIGURO, T.,  
OSHIMA, T. and  
KANEKO, H. Digital DPCM CODEC for TV based on AM/DPCM  
Digital Conversions, IEEE Trans. on Comm.  
Vol. COM-22 No. 7, July, 1974
- (81) GOODMAN, D. J. and  
GREENSTEIN, L. J. Quantization Noise of AM/DPCM Encoders  
B.S.T.J., Vol. 52 No. 2, February, 1973
- (82) RICHARDS, D. L. Transmission Performance of Telephone  
Networks Containing PCM Links, Proc. IEE,  
Vol. 115 No. 9, September, 1968
- (83) CARTER, R. O. Theory of Syllabic Companders, IEE Conf.  
Publication, May, 1968
- (84) MONTGOMERY, W. L. Digitally Linearizable Companders with  
Comments on "Project for Digital Telephone  
Network", IEEE Trans. on Comm. Vol. COMM-18,  
No. 1, February, 1970
- (85) AARON, M. R. and  
KANEKO, H. Synthesis of Digital Attenuators for Segment-  
Companded PCM Codes, IEEE Trans. on Comm.  
Tech., Vol. COM-19 No. 6, December, 1971

- (86) KANEKO, H. A Unified Formulation of Segment Companding Laws and Synthesis of Coders and Digital Companders, B.S.T.J., Vol. 47 No. 7, Sept. 1970
- (87) KANEKO, H. and AARON, M. R. Digital Conversion between Segment Companded PCM Codes, IEE/IEEE Conference on Digital Signal Processing, May, 1972, Loughborough University of Technology, England
- (88) TEXAS INSTRUMENTS INC. The TTL Data Book for Design Engineers
- (89) TEXAS INSTRUMENTS INC. Supplement to TTL Data Book
- (90) CHAKRAVARTHY, C. V. and FARQUI, M. N. Two-Loop Adaptive Delta Modulation Systems, IEEE Trans. on Comm., Vol. COM-22, Oct. 1974
- (91) BETTS, J. A. Adaptive Delta Modulator for Telephony and its Application to the Adaptifon System - An Alternative Implementation of the Lincopex Concept, IEEE Trans. on Comm. Tech., Aug. 1976
- (92) CONSUMER MICROCIRCUIT LTD. The FX209 Analogue-to-Digital Converter, 1976
- (93) WASE, A. E. N. Post Office Services for North Sea Oil Production, IEE/IERE Colloquium on Tropospheric Scatter Communications, 3rd March, 1976
- (94) POWER, R. Operational Experience with Tropospheric Scatter Systems, IEE/IERE Colloquium on Troposcatter Communications, 3rd March, 1976
- (95) GRIFFITHS, J. W. R. Digital Filters, Department of Electronics and Electrical Engineering, Loughborough University of Technology, England
- (96) CANDY, J. C. A Per-Channel A/D Converter having 15-segment  $\mu$ -255 Companding, IEEE Trans. on Comm., Vol. COMM-24, May, 1976
- (97) GOLD, B. and RADER, C. M. Digital Processing of Signals (Book), McGraw-Hill, 1969
- (98) RABINER, L. R. Digital-Formant Synthesizer for Speech-Synthesis Studies, Journal of The Acoustical Society of America, Vol. 43 No. 4, 1968

- (99) SCHAFFER, R. W. A Survey of Digital Speech Processing Techniques, IEEE Trans. on Audio and Electronics, Vol. AU-20, March, 1972
- (100) SCHAFFER, R. W. System for Automatic Formant Analysis of Voiced Speech, Journal of The Acoustical Society of America, Vol. 47 No. 2, 1970
- (101) STEELE, R. FX-209 Adaptive Delta Modulation Encoder and Decoder (to be published)
- (102) NWESSRY, M. AND STEELE, R. The DSCALE, A new All-digital Syllabically Companded Delta Sigma Modulation Decoder (to be published).

