This item was submitted to Loughborough's Research Repository by the author.
Items in Figshare are protected by copyright, with all rights reserved, unless otherwise indicated.

Constant factor delta modulation - a new instantaneously adaptive delta modulation system

PLEASE CITE THE PUBLISHED VERSION

PUBLISHER

© U Aye Thein Kyaw

LICENCE

CC BY-NC-ND 4.0

REPOSITORY RECORD

Kyaw, U Aye T.. 2019. “Constant Factor Delta Modulation - a New Instantaneously Adaptive Delta Modulation System”. figshare. https://hdl.handle.net/2134/7941.

Loughborough University

This item is held in Loughborough University's Institutional Repository (https://dspace.lboro.ac.uk/) and was harvested from the British Library's EThOS service (http://www.ethos.bl.uk/). It is made available under the following Creative Commons Licence conditions.

creative commons

COMMONS DEED

Attribution-NonCommercial-NoDerivs 2.5

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

Attribution. You must attribute the work in the manner specified by the author or licensor.

Noncommercial. You may not use this work for commercial purposes.

No Derivative Works. You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the Legal Code (the full license).

Disclaimer

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

"CONSTANT FACTOR DELTA MODULATION -

A NEW INSTANTANEOUSLY ADAPTIVE DELTA MODULATION SYSTEM"

by

U AYE THEIN KYAW, M.Sc.(Loughborough)

A Doctoral Thesis submitted in partial fulfilment

of the requirements for the award of

Doctor of Philosophy of the Loughborough University of Technology.

September 1973

Supervisor: R. Steele, B.Sc., C.Eng.,M.I.E.E.
Department of Electronic and Electrical Engineering.

BEST COPY

AVAILABLE

Poor text in the original thesis.
Some text bound close to the spine.
Some images distorted

ACKNOWLEDGEMENTS

The author expresses his sincere thanks to Mr. R. Steele, his Supervisor, for guidance and assistance during this work. The author also thanks Professor J.W.R. Griffiths, Head of the Department of Electronic and Electrical Engineering, for providing research facilities.

I would like to thank my parents for their guidance, support and encouragement. I wish also to thank the British Council and the Government of the Union of Burma for allowing me to carry out this work.

Finally I wish to thank my Fiancée and Mrs. S. Peach for their secretarial assistance.

CONTENTS

		PAGE
CHAPTER 1	INTRODUCTION	1
CHAPTER 2	CONSTANT-FACTOR DELTA MODULATOR	5
2.1	INTRODUCTION	5
2.2	DESCRIPTION OF C.F.D.M.	7
2.2.1	POSSIBLE GROUPS OF BINARY PATTERN	7
2.2.1a.	GROUP 1	10
2.2.1b.	GROUP 2	10
2.2.1c.	GROUP 3	10
2.2.1d	GROUP 4	10
2.2.2	PRINCIPLES OF C.F.D.M.	10
2.3	C.F.D.M. DECODER	11
2.4	THE RESTRICTIONS ON 'A' PARAMETERS	13
2.5	EVALUATION OF 'A' PARAMETERS	13
2.6	SELECTION RULES	18
2.7	ADAPTATION CONSTANTS	19
2.8	GENERALITY OF THE PRINCIPLES OF C.F.D.M.	19
2.8.1	ADAPTATION ALGORITHM FOR JAYANT'S CODER	20
2.8.2	ADAPTATION ALGORITHM FOR WINKLER'S H.I.D.M.	20
2.8.3	ADAPTATION ALGORITHM FOR L.D.M.	21
2.9	C.F.D.M. CHARACTERISTICS	22
2.10	MINIMUM STEP SIZE	24
2.11	COMPANDING LAW	25
2.12	THRESHOLD OF CODING	28
2.13	OVERLOAD CONDITION	28
2.14	QUANTIZATION NOISE	29
2.15	AMPLITUDE RANGE OF C.F.D.M.	30
2.16	SIGNAL-TO-NOISE RATIO	31

CONTENTS (Continued) PAGE

2.17	STABILITY	31
2.17.1	RESPONSE OF C.F.D.M. TO AN IMPULSE	32
2.18	EFFECT OF CHANNEL ERRORS	32
2.19	COMPARISON WITH LOGRITHMIC P.C.M.	40
2.20	EFFECT OF MISMATCH OF 'A' PARAMETERS ON S.N.R.	40
2.21	C.F.D.M. WITH LARGE VALUES OF MEMORY LENGTH 'm'	42
CHAPTER 3	COMPUTER SIMULATIONS	47
3.1	INTRODUCTION	47
3.2	SIMULATION OF C.F.D.M.	47
3.3	RESPONSE TO BANDLIMITED GAUSSIAN SIGNALS	48
3.3.1	DYNAMIC RANGE OF C.F.D.M.	49
3.4	CALCULATIONS OF SIGNAL-TO-NOISE RATIO (SNR)	53
3.4.1	DEFINITION OF S.N.R.	53
3.4.2	DEFINITION OF D.F.T.	53
3.4.3	THE F.F.T.	54
3.4.4	THE ALGORITHM FOR THE CALCULATION OF SNR	54
3.4.4.1	ALGORITHM FOR INPUT SIGNAL POWER CALCULATION	55
3.4.4.2	ALGORITHM FOR NOISE POWER	56
3.4.5	SIGNAL-TO-NOISE RATIO EQUATION	56
3.5	STEP RESPONSE OF C.F.D.M.	56
3.6.	STEP SIZE DISTRIBUTION OF C.F.D.M.	56
3.7	ESTIMATION OF POWER SPECTRUM	59
3.8	METHOD OF ESTIMATION OF POWER SPECTRUM	63
3.9	INPUT AND NOISE SPECTRA OF C.F.D.M.	64
3.10	SIMULATION OF JAYANT'S A.D.M.	64
3.11	STEP RESPONSE OF JAYANT'S A.D.M.	68
3.12	RESPONSE OF J.A.D.M. TO BANDLIMITED GAUSSIAN INPUT	68
3.13	IMPULSE RESPONSE OF J.A.D.M.	68
3.14	NOISE SPECTRUM OF J.A.D.M.	73
3.15	STEP SIZE DISTRIBUTION OF J.A.D.M.	73

CONTENTS (Continued) PAGE

3.16	SIMULATION OF WINKLER'S H.I.D.M.	73
3.17	STEP RESPONSE OF H.I.D.M.	73
3.18	IMPULSE RESPONSE OF H.I.D.M.	77
3.19	SIMULATION OF L.D.M.	77
3.20	STEP RESPONSE OF LINEAR DELTA MODULATOR	79
3.21	RESPONSE OF L.D.M. TO BANDLIMITED GAUSSIAN INPUT	79
3.22	IMPULSE RESPONSE OF L.D.M.	79
3.23	STEP SIZE DISTRIBUTION OF L.D.M.	79
3.24	NOISE SPECTRUM OF L.D.M.	85
CHAPTER 4	DISCUSSION	86
REFERENCES		87a
APPENDIX		
A.1	CALCULATION OF PROBABILITY OF ERROR IN DETECTED SIGNAL	88
A.2	COMPUTER PROGRAMMES	93
A.2.a	MAIN PROGRAMME FOR THE SIMULATION AND PLOTTING OF DYNAMIC RANGE AND COMPRESSION CHARACTERISTICS OF C.F.D.M.	93
A.2.b	MAIN PROGRAMME FOR THE SIMULATION AND PLOTTING OF STEP SIZE DISTRIBUTION AND THE ESTIMATION OF POWER SPECTRUM OF C.F.D.M.	93
A.2.1	SUBROUTINE FIR	99
A.2.2	SUBROUTINE INVALUES	101
A.2.3	SUBROUTINE FILTER	103
A.2.4	SUBROUTINE NLOGN	105
A.2.5	SUBROUTINE ENCODER	107
A.2.6	SUBROUTINE HISTOGM	112
A.2.7	SUBROUTINE SHARP	114
A.2.8	SUBROUTINE SMOOTH	116
A.2.9	SUBROUTINE WINDOW	118
A.2.10	SUBROUTINE GRAFI	120
A.2.11	SUBROUTINE COMPRESS LAW	122

CONTENTS (Continued) PAGE

A.3 DESIGN OF NON-RECURSIVE DIGITAL LOW-PASS FILTER 124

A.3.1 FREQUENCY SAMPLING TECHNIQUE 124

A.3.2 REQUIREMENT OF A REALIZABLE FILTER 126

A.3.3 METHOD OF ACHIEVING LINEAR PHASE FILTER 126

A.3.4 PROCEDURE FOR DESIGNING DIGITAL LOW-PASS FILTER 128

A.3.4.1 SELECTION OF FREQUENCY SAMPLES 128

A.3.4.2 ALGORITHM FOR CALCULATING THE PARAMETERS 129

A.3.4.3 PARAMETERS USED 130

A.3.4.4 IMPULSE RESPONSE OF THE FILTER 130

A.3.5 FILTERING 132

LIST OF FIGURES

		PAGE
FIGURE 1	LINEAR DELTA MODULATOR	2
FIGURE 2	C.F.D.M. CODER	6
FIGURE 3	C.F.D.M. DECODER	12
FIGURE 4	SIGNAL TO NOISE RATIO AS A FUNCTION OF A_1	14
FIGURE 5	SIGNAL TO NOISE RATIO AS A FUNCTION OF A_2	15
FIGURE 6	SIGNAL TO NOISE RATIO AS A FUNCTION OF A_3	16
FIGURE 7	C.F.D.M. CHARACTERISTIC	23
FIGURE 8	MEASUREMENT OF COMPRESSION LAW	26
FIGURE 8a	COMPRESSION CHARACTERISTIC OF C.F.D.M.	27
FIGURE 9	IMPULSE RESPONSE OF C.F.D.M.	33
FIGURE 10	EFFECT OF CHANNEL NOISE ON RECEIVER SIGNAL	34
FIGURE 11.	C.F.D.M. RESPONSE TO A SINUSOID INPUT WITH NO CHANNEL ERRORS.	38
FIGURE 12.	C.F.D.M. RESPONSE TO A SINUSOID INPUT WITH A SINGLE CHANNEL ERROR AT 4TH POSITION.	39
FIGURE 13.	C.F.D.M. RESPONSE TO A SINUSOID INPUT WITH A SINGLE CHANNEL ERROR AT 5TH POSITION.	40
FIGURE 14	EFFECT OF MISMATCH OF 'A' PARAMETERS ON SNR	43
FIGURE 15	DIGITAL LOW-PASS FILTER GRAIN CHARACTERISTICS	50
FIGURE 16	DYNAMIC RANGE CURVES OF C.F.D. AND L.D.M.	51
FIGURE 17	COMPUTER GENERATED WAVEFORMS OF INPUT AND RECONSTRUCTED SIGNAL OF C.F.D.M.	52
FIGURE 18	STEP RESPONSE OF C.F.D.M.	57
FIGURE 19	STEP-SIZE DISTRIBUTION OF C.F.D.M.	58
FIGURE 20	PROPERTY OF FINITE LENGTH DATA	61
FIGURE 21a	SPECTRAL WINDOW	65
FIGURE 21b	LAG WINDOW	65
FIGURE 22	POWER SPECTRUM OF BANDLIMITED GAUSSIAN INPUT	66
FIGURE 23	NOISE POWER SPECTRUM OF C.F.D.M.	67
FIGURE 24	STEP RESPONSE OF J.A.D.M.	69
FIGURE 25	DYNAMIC RANGE OF J.A.D.M.	70
FIGURE 26	COMPUTER GENERATED WAVEFORMS OF INPUT AND RECONSTRUCTED SIGNALS	71

LIST OF FIGURES (Continued) PAGE

FIGURE 27	IMPULSE RESPONSE OF J.A.D.M.	72
FIGURE 28	NOISE POWER SPECTRUM OF J.A.D.M.	74
FIGURE 29	STEP SIZE DISTRIBUTION OF J.A.D.M.	75
FIGURE 30	STEP RESPONSE OF H.I.D.M.	76
FIGURE 31	IMPULSE RESPONSE OF H.I.D.M.	78
FIGURE 32	STEP RESPONSE OF L.D.M.	80
FIGURE 33	COMPUTER GENERATED WAVEFORMS OF INPUT AND RECONSTRUCTED SIGNALS OF L.D.M.	81
FIGURE 34	IMPULSE RESPONSE OF L.D.M.	82
FIGURE 35	STEP SIZE DISTRIBUTION OF L.D.M.	83
FIGURE 36	NOISE POWER SPECTRUM OF L.D.M.	84
FIGURE 37	C.F.D.M. DECODER IN THE PRESENCE OF CHANNEL ERRORS	89
FIGURE 38	PROBABILITY DENSITY FUNCTION OF q_1 AND q_o	90
FIGURE 39.1	FLOW CHART OF THE MAIN PROGRAMME FOR THE SIMULATION AND PLOTTING OF DYNAMIC RANGE AND COMPRESSION CHARACTERISTICS OF C.F.D.M.	94
FIGURE 39.1a	MAIN PROGRAMME FOR THE SIMULATION AND PLOTTING OF DYNAMIC RANGE AND COMPRESSION CHARACTERISTICS OF C.F.D.M.	95
FIGURE 39.2	FLOW CHART OF THE MAIN PROGRAMME OF C.F.D.M.	96
FIGURE 39.2a	MAIN PROGRAMME OF C.F.D.M.	98
FIGURE 39.3	PROGRAMME FOR COMPUTING IMPULSE RESPONSE OF LOW-PASS FILTER	100
FIGURE 40	PROGRAMME FOR COMPUTING BANDLIMITED GAUSSIAN SIGNAL AND INPUT POWER IN DBM.	102
FIGURE 41	PROGRAMME FOR LOW-PASS FILTERING SIGNALS	104
FIGURE 42	PROGRAMME FOR THE FAST FOURIER TRANSFORM	106
FIGURE 43	PROGRAMME FOR C.F.D.M. ENCODER AND DECODER	109
FIGURE 43a	PROGRAMME FOR PERFORMING ENCODING AND DECODING OF C.F.D.M. SYSTEM AND FOR THE CALCULATION OF SNR	111
FIGURE 44	PROGRAMME FOR STEP-SIZE DISTRIBUTION OF C.F.D.M.	113
FIGURE 45	PROGRAMME FOR CALCULATING SIGNAL-TO-NOISE RATIO	115
FIGURE 46	PROGRAMME FOR ESTIMATION OF POWER SPECTRUM	117

LIST OF FIGURES (Continued)

		PAGE
FIGURE 47	PROGRAMME FOR WINDOWING THE AUTOCORRELATION FUNCTION	119
FIGURE 48	PROGRAMME FOR PLOTTING POWER SPECTRUM	121
FIGURE 48a	PROGRAMME FOR COMPUTING COMPRESSION CHARACTERISTICS OF C.F.D.M.	123
FIGURE 49	ROTATION OF IMPULSE RESPONSE OF LOW-PASS FILTER	127
FIGURE 50	SHIFTED IMPULSE RESPONSE OF THE FILTER	131

LIST OF TABLES

PAGE

TABLE 1.	POSSIBLE BINARY PATTERNS FORMED	8
TABLE 2	POSSIBLE BINARY GROUPS FORMED	9
TABLE 3	POSSIBLE BINARY PATTERNS OF 3rd ORDER C.F.D.M.	44
TABLE 4	POSSIBLE BINARY GROUPS OF 3rd ORDER C.F.D.M.	45

SYNOPSIS

A new model of an instantaneously adaptive delta modulator called here a "CONSTANT FACTOR DELTA MODULATOR", (abbreviated C.F.D.M.) has been developed and the selection of suitable constant factors (adaptation constants) with its adaptation logic has been described.

The basic delta modulator has been adapted to give an improved performance by introducing a small memory and prediction method in the feedback loop thereby enabling the coder to adapt to the instantaneous variations in the analogue input signal. This C.F.D.M. model of adaptive system adapts its step size, at every sampling instant r, as a result of the detection of the four possible binary groups formed from the last three binary values transmitted. The adaptation constant which is the ratio of the present step size m_r, to the previous step size m_{r-1}, can have, at any sampling instant, one of four values with a magnitude of A_1, A_2, A_3 and A_4, corresponding to the four different possible groups formed. The polarity of the present step size is the same as the present binary value L_r.

The effect of this C.F.D.M. system is that for a given decoded signal to noise ratio, the necessary bandwidth of the transmission channel is reduced. The C.F.D.M. described here gives an improved overall coding characteristic and removes an objectionable hunting characteristic compared to the one-bit memory adaptive DM by JAYANT[13]. It offers wider dynamic range for the bandlimited Gaussian input. The results are compared with other similar schemes on adaptive delta modulators and computer plotted graphs are presented whenever necessary. From these results and responses, the C.F.D.M. seems to be promising for encoding video signals.

Several computer simulations have been made for the design of the Constant Factor delta modulator, JAYANT's CODER, WINKLER's H.I.D.M. coder and linear delta modulator. The performances of these coders have been compared. A considerable number of computer simulation results are presented which relate to digital low-pass filter and the estimation of power spectra.

'A NOTE ON PUBLICATION'

A model on an Instantaneously Adaptive Delta-Modulator, described in Chapter2, has been published in Electronics Letters, vol. 9, No. 4, pp 96-97, 22nd Feb.,1973.

CHAPTER I

INTRODUCTION

The basic principles of delta modulator are well documented in the literature[1-3]. Therefore it will be described here, briefly, as one of the simplest and cheapest ways of encoding analogue information into a digital form. It is a closed-loop sampled data system producing a "staircase" approximation to the analogue input signal, at the error point, if the integrator used on the feedback loop is an ideal integrator. Delta modulator can also be regarded as a one-bit differential P.C.M. because it essentially transmits either of the two codes, a positive pulse or a negative pulse, at every sampling instant, corresponding to a positive or a negative step of fixed amplitude, the sign of which is the sign of the pulse transmitted. A schematic diagram of a linear delta modulator is shown in Figure (1).

Despite the attractive simplicity of a linear delta modulation system, it has its drawbacks also. One of the major disadvantages of linear delta modulator is its limited dynamic range. These limitations are due to the two inherent types of distortion noise introduced by the system. Small or insufficient values of step sizes introduce "slope overload noise", that occurs during the large signal slope when the system cannot follow the input signal, by transmitting along a sequence of 'ones' or 'zeros'. The finite step size of the system introduces another type of noise called "granular noise", when the system is tracking the signal, during the small signal slope, by producing an alternate pattern of one and zero, at the output. Hence linear delta modulator has only one peak signal to noise ratio point, offering a very narrow useful dynamic range. Although the dynamic range can be improved by increasing the clock rate of the system, the limitation on the channel band width in every communication system has limited the maximum clock frequency which consequently limits the dynamic range.

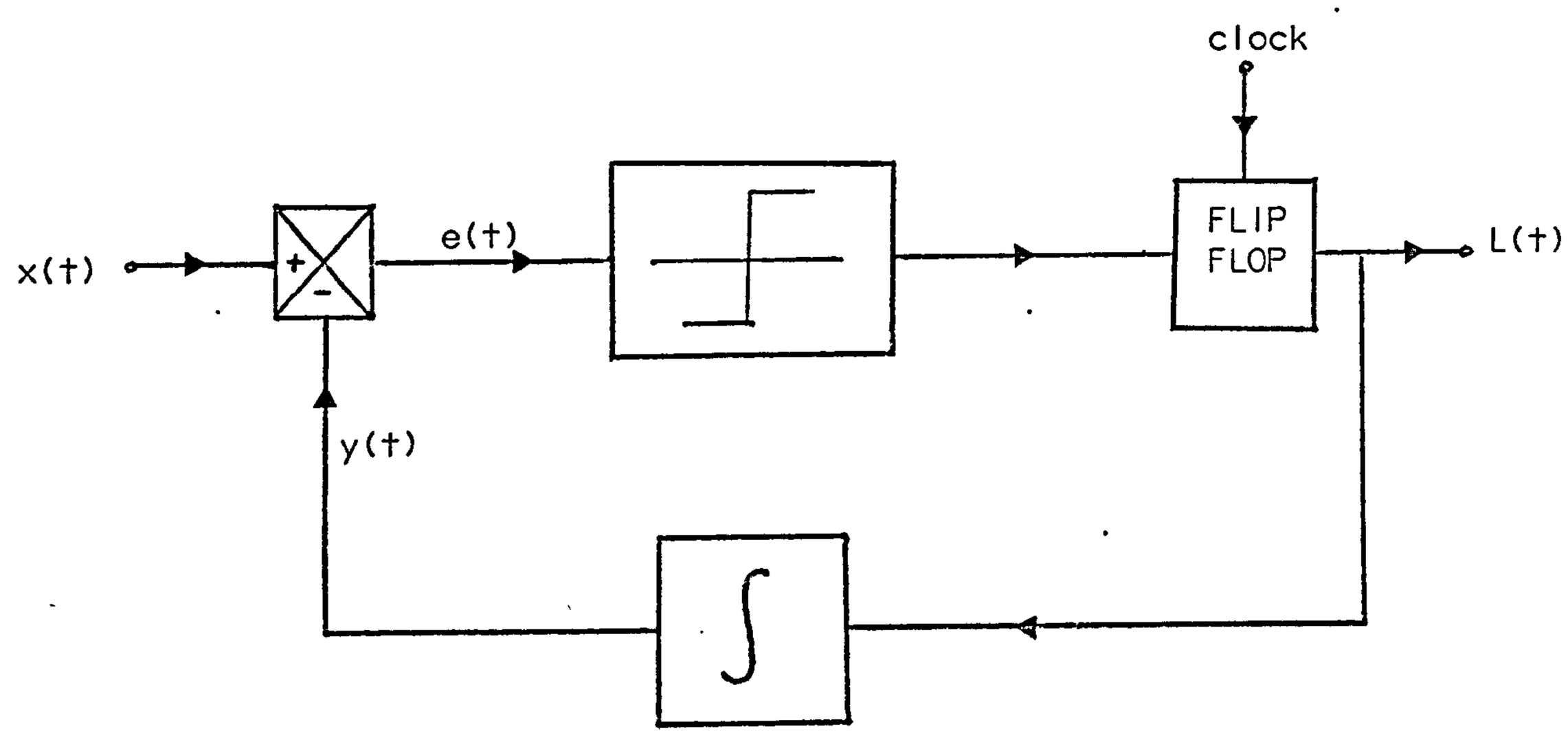

Figure (1) Linear Delta Modulator.

Overload characteristic curves[1-3] for the linear delta modulator have the matching characteristics as that of the speech signals. Due to these drawbacks, and the overload characteristic of linear delta modulator, it has been confined to coding speech only.

Therefore increasing the dynamic range, for a given bit rate, becomes highly essential, so that the limitations imposed by the linear delta modulator have been removed and to open wider and useful application of the system. To meet these requirements, and to operate delta modulator at relatively low bit rate, several types of adaptive delta modulators[4-14] have been proposed. Such adaptation or "companding" can be achieved either at a syllabic rate or instantaneous. The word 'companding' is the combined word for signal dynamic range 'compressing' at the encoder and 'expanding' at the decoder.

Syllabic compandors are those which are characterized by 'continuous' adaptation of the step size, where the gain of the adaptation circuitry varies in accordance with the level of the input signal but is substantially constant over a number of cycles of the input signal. The step size is controlled by the envelope of the analogue signal extracted from the output binary signal. Such types of syllabic compandors are reported in the literature[4-7].

Instantaneously adaptive delta modulators incorporate discrete adaptation of step size at every sampling instant. Most of the adaptation algorithms are based on the principle of doubling, halving or otherwise changing the step size when a string of consecutive 'pulses' or 'no pulses' are detected in the binary output signals. These types of adaptive coders are widely described in references (8-14).

Instantaneously adaptive coders are developed for coding video signals. The reason isthat it is desirable to adapt the step size of linear delta modulator according to instantaneous signal value rather than an average value, as far as coding television signals is concerned.

Another interesting feature of instantaneously adaptive coders is their resistance to mathematical analysis[8,11,12], which means that they are intuitively conceived rather than designed. None of the papers in the literature can achieve meaningful analysis except for those concerned with most simple input signals.[20] Moreover the analysis is further complicated by the variety of performance criteria. These criteria depend upon the type of signals to be encoded, the properties of the transmission channel, etc.

In this piece of work, the main aim is to overcome the limitations of linear delta modulator and to design an instantaneously adaptive delta modulator, that would be able to encode television signals. As a result we have presented here a new type of instantaneously adaptive delta modulator called "Constant Factor Delta Modulator", which is named after the behaviour of the system.

It differs from the other schemes on adaptive delta modulator in that the adaptation of the new step size is made when certain groups of binary patterns are detected. Method of grouping of binary patterns according to their characteristics enables us to indicate the possible intermediate state of the system, rather than the overload and the idling states.

The analyses have been made of the characteristics of C.F.D.M., dynamic range, compression law, quantization noise, and overall transmission characteristics, etc.

The following chapters will describe in detail the development of our C.F.D.M. system and the presentation of the results from the computer simulations of the C.F.D.M.

CHAPTER 2

CONSTANT FACTOR DELTA MODULATOR

2.1 INTRODUCTION

Many types of instantaneously adaptive delta modulators[8-14] have been designed. In these systems, the step size is changed significantly at every sampling instant in accordance with the present and the previous binary levels in L(t) waveform. Most of them try to adapt the new step size when a consecutive 'ones' or 'zeros' are detected, or when a change of the polarity of the binary level occurs, at every sampling instant. Though all of these adaptive delta modulators offer some improvements to a certain extent, over the linear delta modulator, they do not seem to have used all the information available in the L(t) waveform to be used as a control function for the adaptation of new step size.

Therefore, one should realise that a better adaptive system is to accommodate a sensing device which would be able to detect not only the occurence of the string of consecutive 'ones' or 'zeros' for the indication of an overload, but alsoprovide a special care, such that the overloading is not encouraged or removed as quickly as possible if the overloading has occurred. It should sense also the immediate state after the overload condition by the detection of a sign reversal just after the sequence of 'ones' or 'zeros' has been detected and should adapt the step size when the system is in an idling condition, so that the step size converges to a minimum allowable value.

In an attempt to achieve the above requirements and to overcome the limitations of the linear delta modulator which we have discussed earlier, we have developed a new instantaneously adaptive delta modulator called here by the author a "Constant Factor Delta Modulator".[24] The following sections will precisely describe its development and the principles of C.F.D.M.

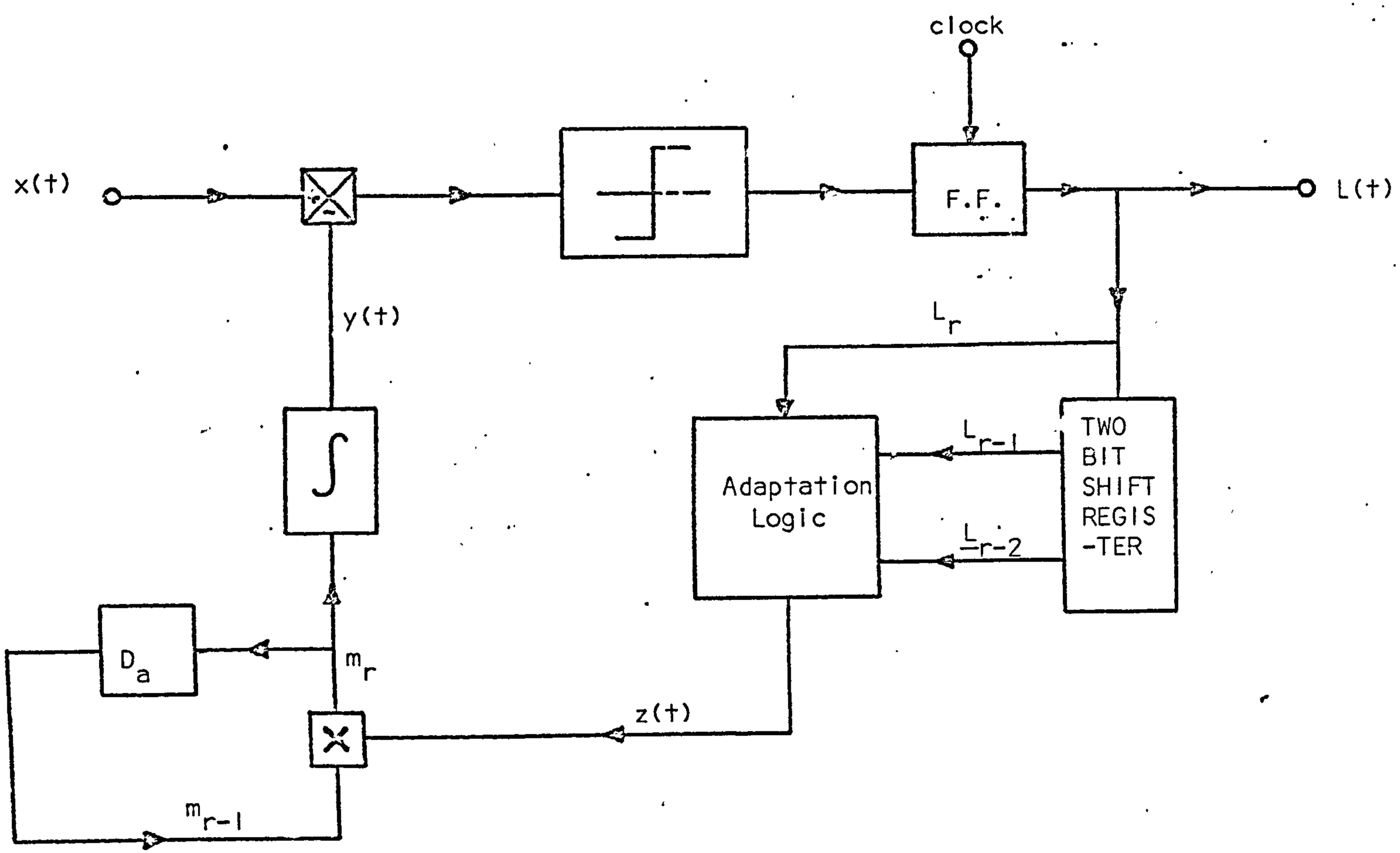

Figure (2) C.F.D.M. CODER.

Prior to the description of the development of C.F.D.M. system we have here a few "assumptions" to be made. It is assumed that there is no error in the transmission channel. This means that the encoded binary signals which are transmitted through a noisy channel are correctly received before the decoding, in the receiving end. It has been assumed that an ideal integrator is used in both the local decoders in the transmitter and the receiver.

2.2 DESCRIPTION OF C.F.D.M.

The C.F.D.M. coder is shown in figure (2), $x(t)$ is the analogue signal and $L(t)$ is the binary output of the coder. The $L(t)$ signal is passed through a two-bit shift register and the outputs of this register, together with the present output from the coder, are applied to the adaptation logic. Thus the coder adapts itself according to the present binary level Lr and the previous two binary levels Lr-1 and Lr-2 in the $L(t)$ waveform. The subscript r denotes the rth sampling instant, and r-1 the previous sampling instant, etc. The system provides a memory length of two for the ease of design feature and also to take into account the correlation existing in the subsequent input signal samples.

Thus, the logic system is confronted with $2^3 = 8$, possible binary patterns as shown in the table - 1. Since each binary level has its own complementary level, these can be conveniently reduced to four groups which indicate the possible state of the system as shown in table - 2. Therefore, in general for a memory length of m bits long, the number of possible groups that could be formed is

$$N = 2^{n-1} \qquad \ldots\ldots\ldots\ldots \qquad (1)$$

where $n = m + 1$

2.2.1 POSSIBLE GROUPS OF BINARY PATTERN

As shown in table - 2, the groups can be classified as follows.

PATTERN NUMBER	L_r	L_{r-1}	L_{r-2}
1	0	0	0
2	0	0	1
3	0	1	1
4	0	1	0
5	1	0	0
6	1	0	1
7	1	1	1
8	1	1	0

TABLE I. POSSIBLE BINARY PATTERNS FORMED.

GROUP NUMBER	NAME OF GROUP	ADAPT-ATION CONSTANT	PATTERN NUMBER	L_r	L_{r-1}	L_{r-2}
1	IDLING	A_1	4	0	1	0
			6	1	0	1
2	SIGN REVERSAL	A_2	3	0	1	1
			5	1	0	0
3	SEMI OVERLOAD	A_3	2	0	0	1
			8	1	1	0
4	OVER LOAD	A_4	1	0	0	0
			1	1	1	1

Table 2. POSSIBLE BINARY GROUPS FORMED.

2.2.1a GROUP 1

It is called the alternating polarity group or idling group and often occurs when the y(t) signal is trying to hunt a slowly varying or a steady slope input signal x(t). Pattern No. 4 and 6 are grouped together since one forms the complementary to the other.

2.2.1b GROUP 2

The sign reversal group is characterized by the previous two bits having the same binary levels and different from the present binary level. As shown in pattern numbers (3) and (5), the presence of group 2 pattern indicates that the coder is coming out of the overload condition and is beginning to reduce its error voltage.

2.2.1c GROUP 3

It is called the semi-overload group as the coder appears to be starting to form an overload pattern of all ones or all zeros. Pattern numbers (2) and (8) are regrouped together.

2.2.1d GROUP 4

It is classified as an overload group. This is characterized by the inspection of pattern numbers (1) and (7). In these patterns all the three binary levels are "zeros" or "ones" indicating that the system is overloaded by the input signal having a negative or positive slope respectively.

2.2.2 PRINCIPLES OF C.F.D.M.

Instantaneous companding is achieved in our C.F.D.M. system. The basic principle of companding is the compression of the signal while encoding and expanding it when the decoding is done to reconstitute the original signal. Thus if an expanding function is applied in the local decoder of linear delta modulator, the coding will be performed with compression characteristic. The companding is performed entirely in the

local decoder by changing the step size at the input to the integrator, and the expanding function of the local decoder can be explained as follows.

In this C.F.D.M. coder the selection of adaptation constants will be made on the basis of the characteristic feature of the groups of binary pattern formed in the logic system provided. It therefore follows that four different adaptation constants will be provided for four groups of binary pattern formed. The wave form at the output of the logic is z(t) and has at any sampling instant one of four possible values of adaptation constants. The evaluation of the actual magnitude of these will be mentioned later.

If the binary pattern representing the group 1 is present, then the z(t) produces a voltage having a magnitude of A_1. Similarly for the binary patterns representing groups 2, 3 and 4, results in having the z(t) with a magnitude of A_2, A_3 and A_4 volt respectively. The polarity is the polarity of L_r.

z(t) is multiplied by m_{r-1} to give m_r which is the voltage fed to the integrator. m_{r-1} is obtained from m_r by passing it through a one bit analogue delay D_a. This m_r differs from m_{r-1} by one of the four constant factors, and hence the name given to this coder. Integration of m_r gives the feed back signal y(t) which when subtracted from x(t) produces an error signal which is then quantised. The output of the quantiser is connected to a sample and hold circuit which is implemented with a D Flip-Flop, to give L(t) waveform.

2.3 C.F.D.M. Decoder

The decoder consists of the complete system in the feed back loop of figure (1), such that the received L(t) waveform can reproduce y(t), and followed by a low-pass filter network. The job of the low-pass filter is to remove the high frequency components in y(t) due to quantization and pass the frequency components contained in the message band to reproduce the x(t) waveform. The C.F.D.M. decoder is shown in figure (3).

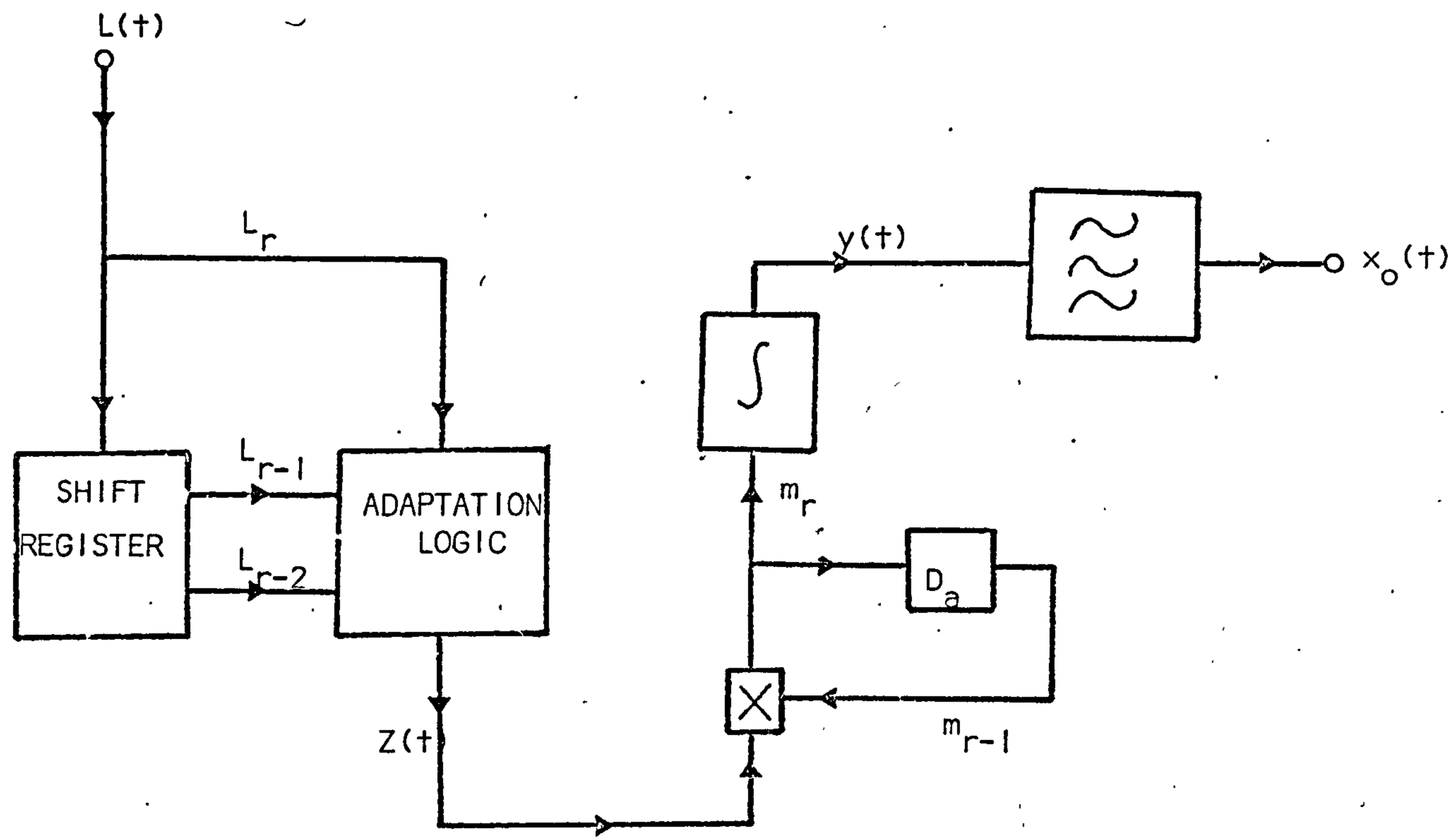

Figure 3. C.F.D.M. DECODER.

2.4 THE RESTRICTIONS ON 'A' PARAMETERS

Before we describe the evaluation of the A parameters, we introduce some basic bounds on these adaptation constants as follows

(i) In order to hunt a signal with converging step sizes, the A_1 and A_2 parameters have magnitude less than unity.

i.e. $|A_1| < 1.0$ ..... (2)

$|A_2| < 1.0$ ..... (3)

(ii) In order to adapt the signal at the semi-overload condition and to prevent getting into the state of overloading, it is of necessity that A_3 parameter should have the magnitude greater than unity, excessively

i.e. $|A_3| > 1.0$ ..... (4)

(iii) Adaptation of A_4 parameter is reached always after the adaptation of A_3. This simply implies that there is an insufficiency with the A_3 parameter and therefore, in order to prevent further overloading of the system, it is essential that A_4 parameter must be made greater than A_3.

i.e. $|A_4| > |A_3|$ ..... (5)

2.5 EVALUATION OF 'A' PARAMETERS

The c.f.d.m. system is required to accommodate a variety of input signals. In this thesis subjective evaluation of the system is not done, and it has been explored using well defined test signals such as sinusoids, steps, impulse functions and band-limited Gaussian signals. It can be intuitively anticipated that the adaptation constants to produce optimum encoding performance will be different for each type of input signal. It might be possible for the adaptation parameters to be themselves adaptive to changing signal conditions, but the solution decided here is to select a set of adaptation parameters which give satisfactory results for a wide range of types of input signals. Thus at the outset it is acknowledged that these parameters are generally sub-optimum. The values ascribed to these parameters were arrived at by a combination of physical reasoning

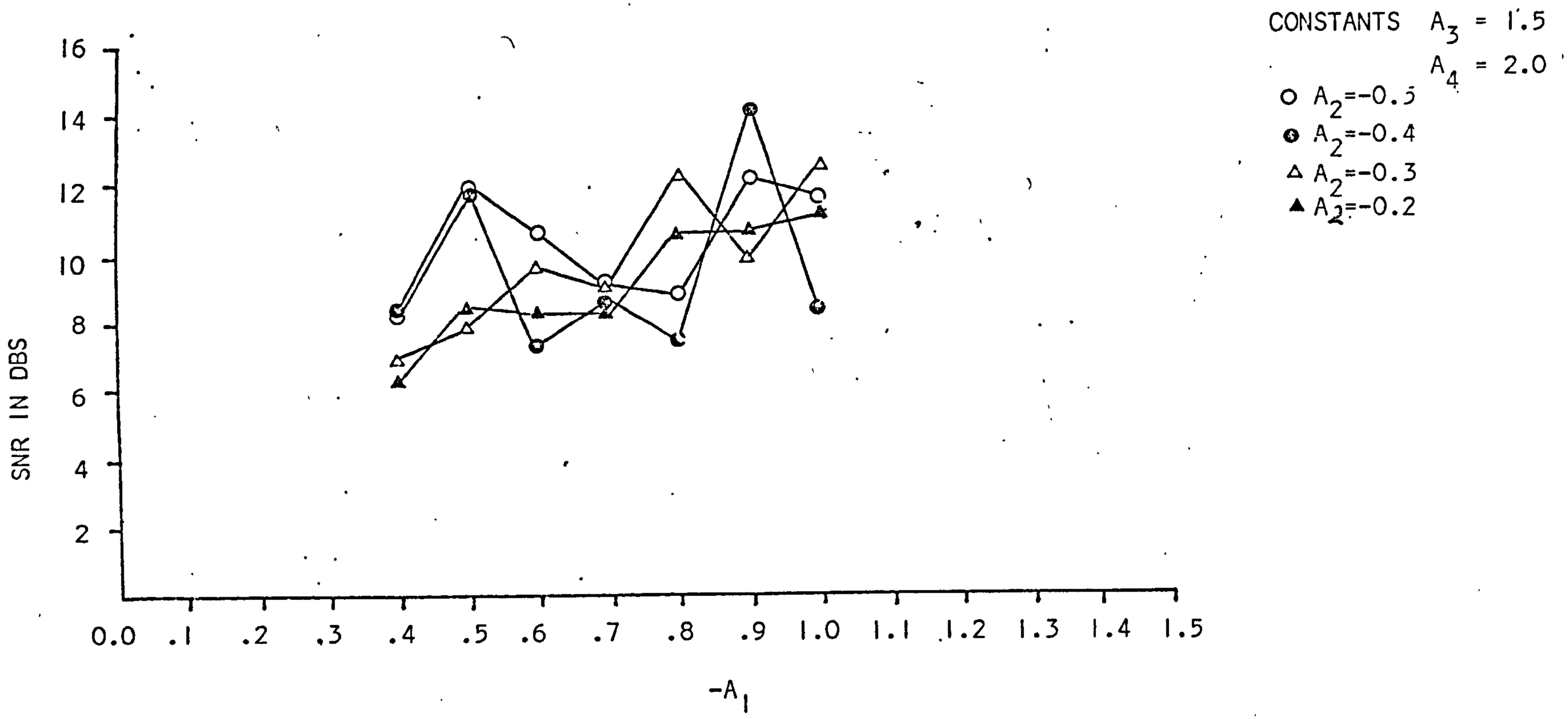

Figure (4) SIGNAL TO NOISE RATIO AS A FUNCTION OF A_1.

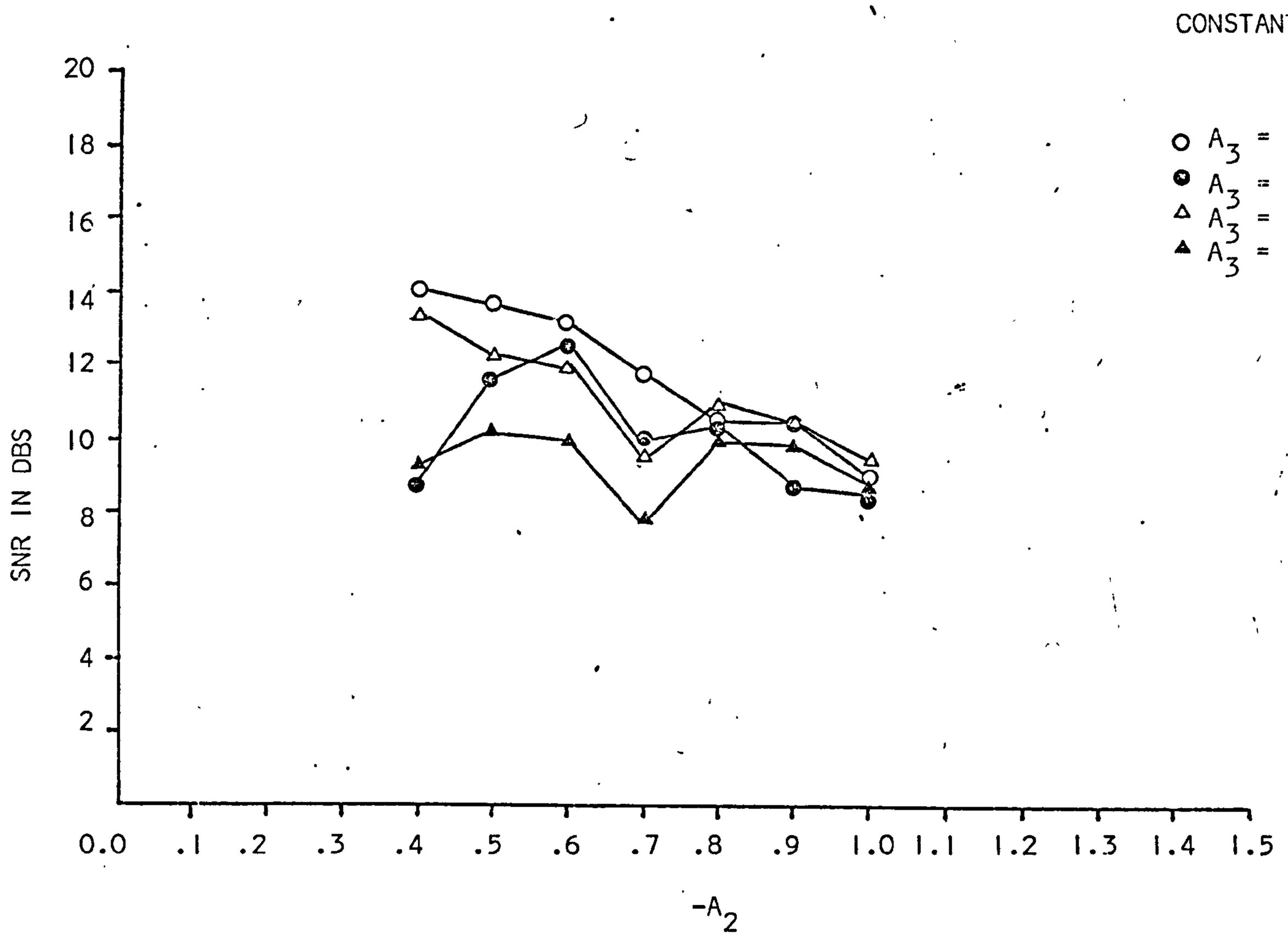

Figure (5) SIGNAL TO NOISE RATIO AS A FUNCTION OF A_2.

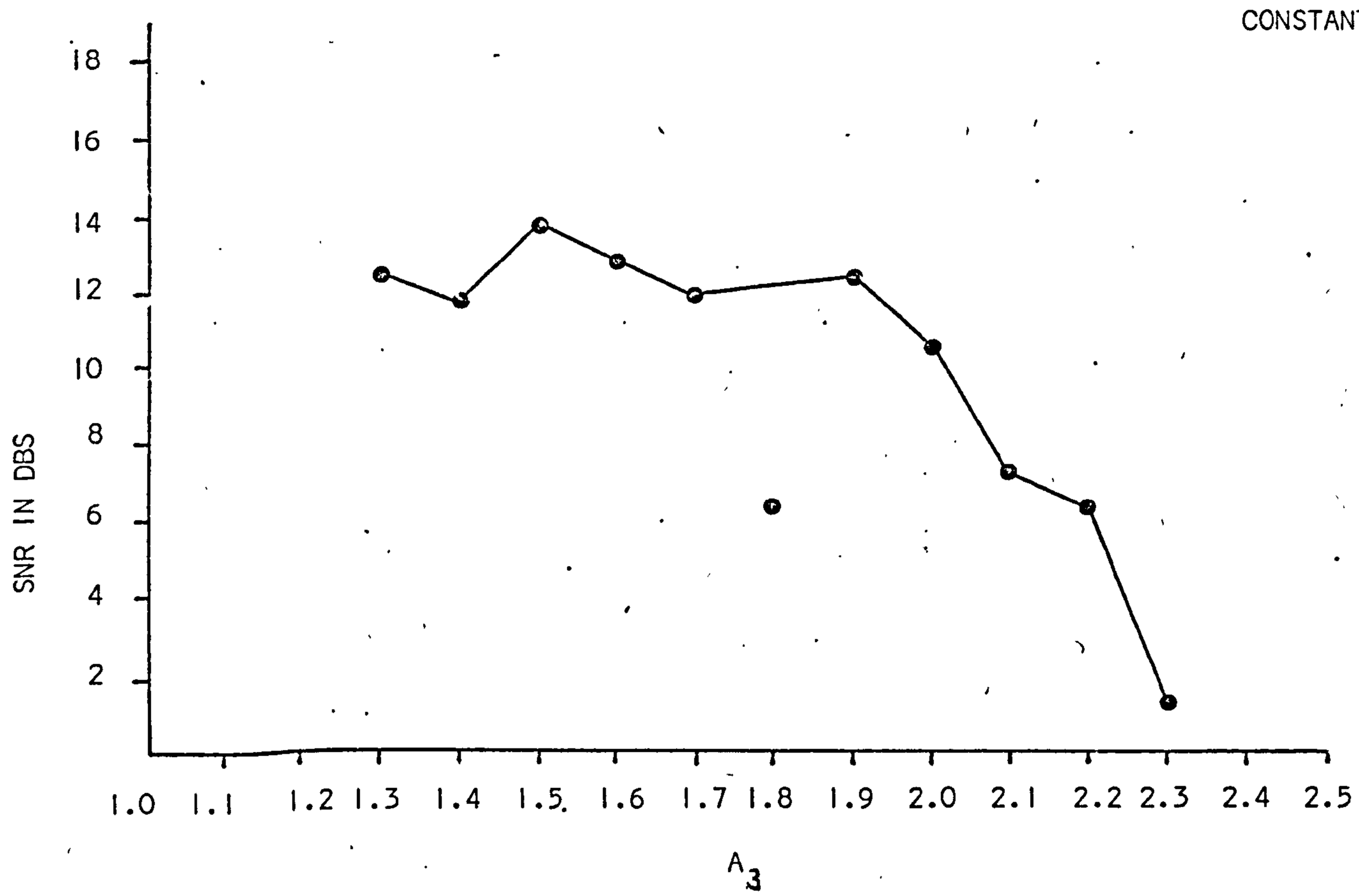

Figure (6) SIGNAL TO NOISE RATIO AS A FUNCTION OF A_3

based on conditions of the encoder's stability, convergence, overshoots etc. and iterative procedures.

The results of the step response are discussed in Section 3.5 and displayed in figure 18. When a sequence of identical polarity output pulses occur, i.e. the encoder is severely overloaded, it is clearly required that the feedback signal y(t) should increase at a rapid rate. The A-parameter in this situation is A_4 and it is given a value of 2 because this ensures that y(t) increases in binary fashion, as it does in the Winkler's[9] coder. The coder described by Jayant[16] having only two parameters, rather than four, has an adaptation parameter in an overload condition of 1.5. The choice of A_4 equal to 2 results in a step response which rises as fast as that of Winkler, and faster than that of Jayant. If A_4 is made in excess of 2 it results in larger overshoots when tracking random signals.

From table 2, it can be seen that A_3 is generated when the encoder is entering an overload condition. A_3 must therefore be in excess of unity, and less than the severe overload parameter A_4. The physical bounds on A_2 mu-t be less than one because the encoder is coming out of an overload condition, and is less than A_1, for the latter parameter is produced when the encoder does not know whether it is about to be semi-overloaded or the prevailing condition will be maintained. The above remarks can be summarised by

$$A_2 < A_1 < A_3 < A_4$$

and the polarity of these parameters is equal to the sign of the current L(t) pulse.

In order to attach some actual values to these parameters the following iterative procedure was adopted for a sinusoidal input signal. The decoded signal to noise ratios were computed as a function of a particular A parameter while the other A parameters were held constant. The results are displayed in figures 4, 5, and 6. When $A_1 > 1$ or $A_2 > 1$, which is in conflict with the above inequality, the snr became negative.

This situation is not shown in these figures. It can be seen that these results are not always consistant and that the curves display irregularities. By themselves they do not indicate the obvious choice of parameters, but when used as a support to the physical arguments presented above they enabled the following set of A parameters to be selected:- $A_1 = 0.9$, $A_2 = 0.4$, $A_3 = 1.5$ and $A_2 = 2.0$. These parameters which were established for a sinusoidal input of 1 KHz and an encoder clock rate of 40 KHz were used for the step response, impulse response and tracking of Gaussian signals as described in sections 3.5, 2.17.1 and 3.3. They were found to give satisfactory results, indeed for step input and pulse inputs, the encoder behaved better than other existing systems. Changes in the parameters were observed to have small effects. A possible explanation of the insensitivity of the performance of the system to differences in these A parameters may be due to the relatively low ratio of clock rate to the highest frequency in the input signal. This is because the input signal makes changes at too fast a rate for the full potential of the adaptation algorithm to be realised. It is anticipated that at higher clock rates the choice of the A parameters would be a crucial factor on the performance of the system. This low (f_p/F_{c2}) ratio also results in a failure to exploit the CFDM.

2.6 SELECTION RULES:

The selection rules which by incirporating some kind of adaptation logic would select an adaptation constant at every sampling instant. Considering the facts we have just described, the following rules for the selection of adaptation constants can be developed.

$$Z(t) = A_1 \quad \text{if} \quad L_r \neq L_{r-1} \text{ and } L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (6)$$

$$Z(t) = A_2 \quad \text{if} \quad L_r \neq L_{r-1} \text{ and } L_{r-1} = L_{r-2} \quad \ldots\ldots\ldots\ldots\ldots \quad (7)$$

$$Z(t) = A_3 \quad \text{if} \quad L_r = L_{r-1} \quad \text{and}$$

$$L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (8)$$

$$Z(t) = A_4 \quad \text{if} \quad L_r = L_{r-1} \quad \text{and}$$

$$L_{r-1} = L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (9)$$

These can also be expressed in Boolean functions from table 2 as -

$$Z_1 = \bar{L}_r L_{r-1} \bar{L}_{r-2} + L_r \bar{L}_{r-1} L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (10)$$

$$Z_2 = \bar{L}_r L_{r-1} L_{r-2} + L_r \bar{L}_{r-1} \bar{L}_{r-2} \quad \ldots\ldots\ldots\ldots \quad (11)$$

$$Z_3 = \bar{L}_r \bar{L}_{r-1} L_{r-2} + L_r L_{r-1} \bar{L}_{r-2} \quad \ldots\ldots\ldots\ldots \quad (12)$$

$$Z_4 = \bar{L}_r \bar{L}_{r-1} \bar{L}_{r-2} + L_r L_{r-1} L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (13)$$

2.7 ADAPTATION CONSTANTS

It is desirable to use the different set of suitable optimised A constants for different input signal, by doing the three SNR tests again. But for the simulation of C.F.D.M. the A constants used are $A_1 = -0.9$, $A_2 = -0.4$, $A_3 = 1.5$ and $A_4 = 2.0$.

2.8 GENERALITY OF THE PRINCIPLES OF C.F.D.M.

The principles of C.F.D.M. can be considered as the generalised principles for several types of instantaneous adaptive delta modulators. This can be verified by deducting different adaptation algorithms for these different adaptive schemes from that of C.F.D.M. For the later simulations of all these coders, the following adaptive algorithm were derived in terms of the adaptation algorithm of our C.F.D.M. coder, having four different adaptation constants for four different binary groups formed from the last three binary levels transmitted.

The same length of memory store will be used as it was in C.F.D.M. coder. The only difference was the use of adaptation constants, different from that of C.F.D.M. coder, for each of these adaptive coders. This shows the sufficiency and the efficiency of the length of memory stores used in our C.F.D.M.

2.8.1 ADAPTATION ALGORITHMS FOR JAYANT'S CODER[13]

In Jayant's coder, when a string of two consecutive binary levels of the same polarity were detected, adaptation constant P was selected. Whenever two unequal binary levels were detected adaptation constant of $Q = -\frac{1}{P}$ was selected where P = 1.5. The detection process was made for only two possible groups formed from the present and last binary levels transmitted.

In terms of C.F.D.M. the two unequal binary levels were detected in group 1 and 2, and the occurence of two consecutive binary levels of the same polarity were detected in group 3 and 4. Therefore the adaptation algorithm for JAYANT's Coder in terms of C.F.D.M. algorithm can be written as

$$Z(t) = A_1 = -Q \quad \text{if } L_r \neq L_{r-1} \text{ and } L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots\ldots \quad (14)$$

$$Z(t) = A_2 = -Q \quad \text{if } L_r \neq L_{r-1} \text{ and } L_{r-1} = L_{r-2} \quad \ldots\ldots\ldots\ldots\ldots \quad (15)$$

$$Z(t) = A_3 = P \quad \text{if } L_r = L_{r-1} \text{ and } L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots\ldots \quad (16)$$

$$Z(t) = A_4 = P \quad \text{if } L_r = L_{r-1} \text{ and } L_{r-1} = L_{r-2} \quad \ldots\ldots\ldots\ldots\ldots \quad (17)$$

where P = 1.5 and Q = -0.66.

2.8.2 ADAPTATION ALGORITHMS FOR WINKLER'S H.I.D.M.[8,9]

The High Information Delta Modulator (H.I.D.M.) does not have any adaptation of the step-size when a string of two like binary levels are detected. However, it doubles the previous step-size when the three binary levels of the same polarity are detected and keeps halving the step-size whenever a reversal of binary levels occurs at the output of the coder. The requirements of this type of coder can be easily achieved by using

the algorithm for C.F.D.M. and by just changing the values of the 'A' parameters to match the H.I.D.M. characteristic. Thus the algorithm for H.I.D.M. in terms of C.F.D.M. principle becomes

$$Z(t) = -0.5 \quad \text{if} \quad L_r \neq L_{r-1} \quad \text{and} \quad L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (18)$$

$$Z(t) = -0.5 \quad \text{if} \quad L_r \neq L_{r-2} \quad \text{and} \quad L_{r-1} = L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (19)$$

$$Z(t) = 1.0 \quad \text{if} \quad L_r = L_{r-1} \quad \text{and} \quad L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (20)$$

$$Z(t) = 2.0 \quad \text{if} \quad L_r = L_{r-1} \quad \text{and} \quad L_{r-1} = L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (21)$$

Therefore 'A' parameters of C.F.D.M. coder now have the values of $A_1 = A_2 = -0.5$, $A_3 = 1.0$ and $A_4 = 2.0$.

2.8.3 ADAPTATION ALGORITHM FOR L.D.M.[1-3]

The linear delta modulator increases or decreases its step-size by one unit depending on whether the last binary bit transmitted is a one or a zero respectively. We can think of linear delta modulator as an adaptive delta modulator system which adapts its step size with the adaption constants having the magnitude of one. Therefore the algorithm for linear delta modulation may be deducted from the adaption algorithm of C.F.D.M. as,

$$Z(t) = -1.0 \quad \text{if} \quad L_r \neq L_{r-1} \quad \text{and} \quad L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (22)$$

$$Z(t) = -1.0 \quad \text{if} \quad L_r \neq L_{r-1} \quad \text{and} \quad L_{r-1} = L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (23)$$

$$Z(t) = 1.0 \quad \text{if} \quad L_r = L_{r-1} \quad \text{and} \quad L_{r-1} \neq L_{r-2} \quad \ldots\ldots\ldots\ldots \quad (24)$$

$$Z(t) = 1.0 \quad \text{if} \quad L_r = L_{r-1} \quad \text{and}$$

$$L_{r-1} = L_{r-2} \qquad \ldots\ldots\ldots\ldots\ldots \qquad (25)$$

Hence the 'A' parameters used in C.F.D.M. now have the value of $A_1 = A_2 = -1.0$ amd $A_3 = A_4 = 1.0$.

2.9 C.F.D.M. CHARACTERISTICS

Constant factor delta modulator has a special feature in the way it adapts the new step size. A adaptation of a new step magnitude is in a constant factor to the previous step magnitude, according to the selection rules described in section (2.6). Since the C.F.D.M. coder needs at least three binary levels to function as a C.F.D.M. coder, the adaptation of step sizes follows the sequence A_1, A_3, A_4 .....A_4^{n-2} and increases exponentially in response to the sequence of n "ones" in L(t). The characteristic of the coder can be evaluated by studying the step response of the coder.

When a step input x(t) is applied to the coder, the C.F.D.M. system transmits a sequence of binary "ones" and the adaptation of step size follows the sequence described earlier and therefore for the nth adaptation the y(t) signal can be evaluated as

$$y_n = -A_1\left[1+A_3 \sum_{r=2}^{r=n} A_4^{r-2}\right] \qquad \ldots\ldots\ldots\ldots \qquad (26)$$

Minimum value of r is 2 because we have assumed an initial step size of unity and two binary bits are required before the adaptation logic functions

After the nth adaptation, let the y(t) overshoot the x(t) waveform. Therefore at t = n + 1 instant, y_t waveform will have the value

$$y_{n+1} = y_n - A_4^{n-2} \; A_1 \; A_2 \; A_3 \qquad \ldots\ldots\ldots\ldots \qquad (27)$$

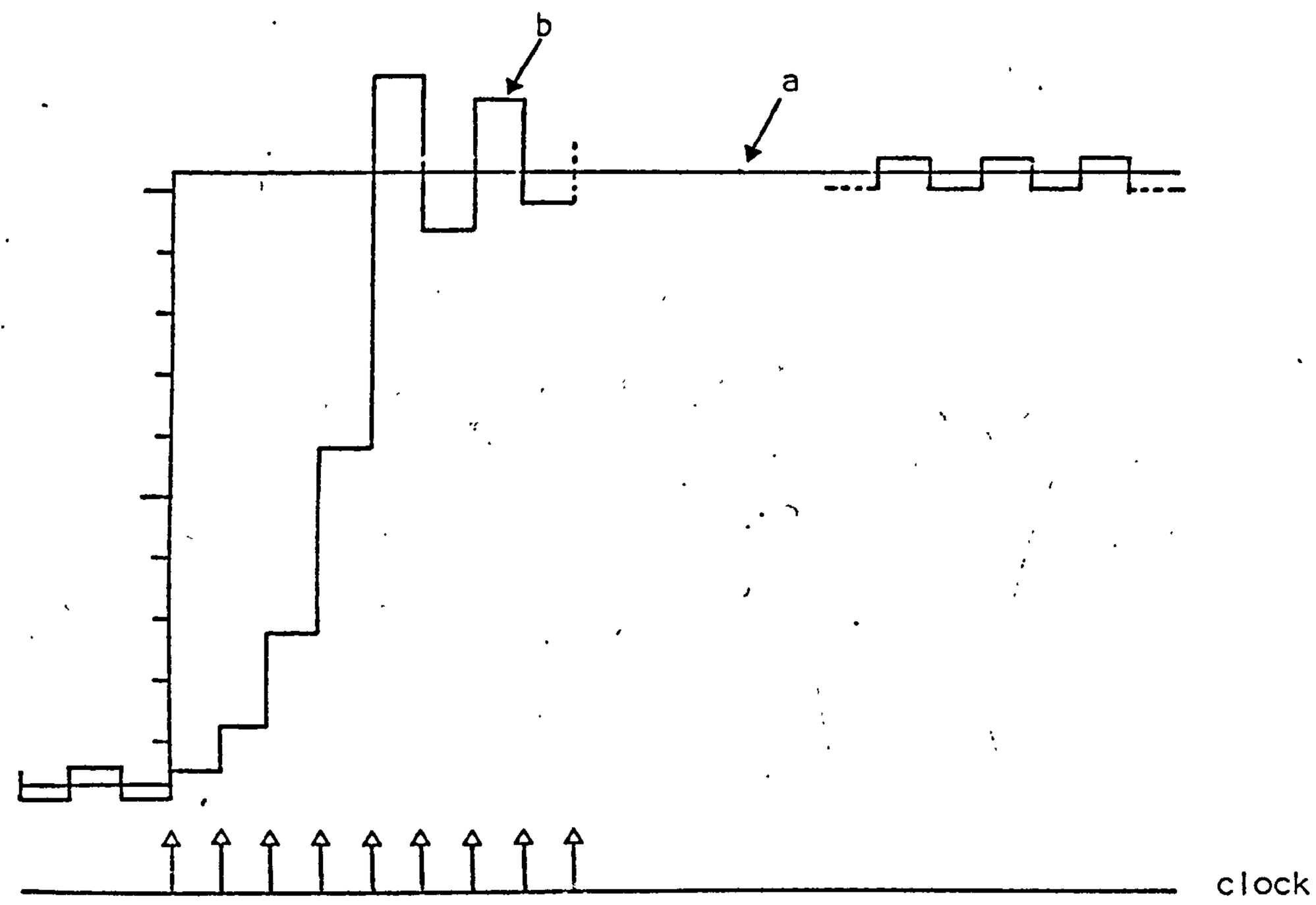

Figure (7) C.F.D.M. CHARACTERISTIC.

a - x(t) step function
b - y(t) signal

Suppose at $t = n + 2$, if $y(t)$ is still $> x(t)$, $y(t)$ waveform becomes,

$$y_{n+2} = y_{n+1} - A_4^{n-2} \; A_1 \; A_2 \; A_3^2 \quad \ldots\ldots\ldots\ldots \tag{28}$$

On the other hand if $y(t) < x(t)$ at $t = n+2$

then,

$$y_{n+2} = y_{n+1} + A_4^{n-2} \; A_2 \; A_1^2 \; A_3 \quad \ldots\ldots\ldots\ldots \tag{29}$$

etc. The essential characteristic of this coder is that it offers different weighting factor for the adaptation of new step size, in accordance with the different situation of the tracking signal $y(t)$. Following the step response just described, the coder has the facility to accommodate the tracking ability to the step function and finally hunts the step with the basic idling pattern of 1 0 1 0 ..... with the smallest allowable step size, limited for the idling condition. The tracking of the step is shown in figure (7).

2.10 MINIMUM STEP SIZE

In order for a .... 1 0 1 0 1 0 ..... pattern to be formed a lower limit to the step size say, δ min., is put. The need of this can be easily verified by studying the step response of C.F.D.M. After a certain length of clock periods Nt, a condition will be reached when the $y(t)$ hunts the steady step input voltage $x(t)$, in the mode of A_1 adaptation, at every sampling instant. The step size will always decrease and the point will be reached when the step size magnitude is less than δ min. At this instant, however, the coder makes this step size equal to δ min. Therefore from that sampling instant onwards, the $y(t)$ will hunt the $x(t)$, with a minimal step size δ min, and the traditional ..... 1 0 1 0 1 0 ..... pattern will be established.

2.11 COMPANDING LAW

This law is obtained by using the experimental arrangement shown in Figure 8. It can be seen that a sinusoidal input $E_s \sin \omega_s t$ is applied to the CFDM encoder and the resulting binary waveform L(t) is decoded by a linear D.M. decoder to give the compressed sinusoid $E_{sr} \sin \omega_s t$. If the CFDM decoder shown in Figure 3 is used the output would of course be approximately equal to $E_s \sin \omega_{st}$, but the absence of the 'expander' in Figure 8 results in a sinusoid having a smaller amplitude, i.e. $E_{sr} < E_s$. The value of E_s is varied over a wide range and for each value the decoded output signal E_{sr} is noted. Figure 8a is the compression law for a sinusoid having a frequency of 800 Hz sampled at 40 KHz.

This companding law should not be confused with those laws used in p.c.m. systems which are independent of the input signal. The companding law shown in figure 8a may not be the same for other input signals like speech, say. Its value is that it gives an indication of the amount of compression obtainable at a particular frequency, and this test can be used for other coders to give a comparison of their ability to compress the signal.

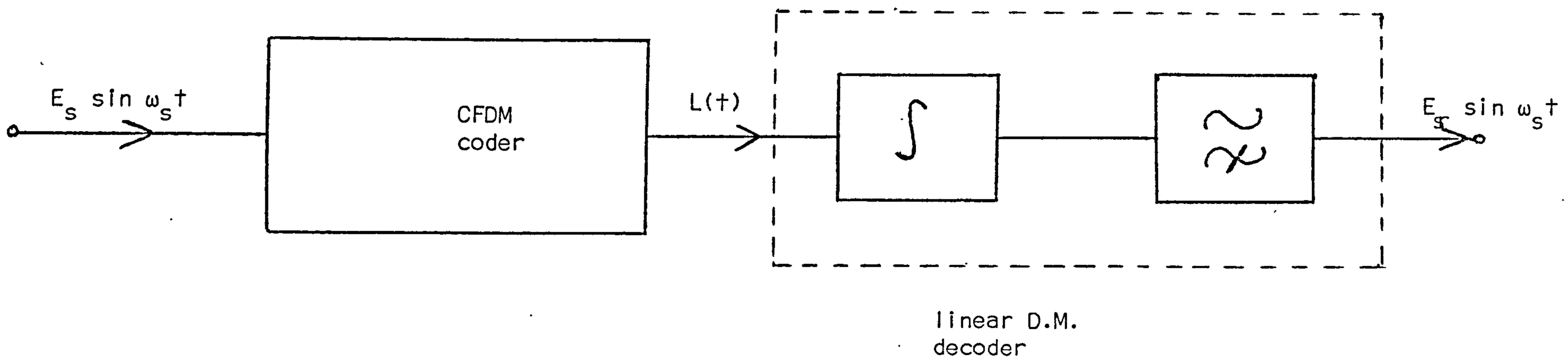

Figure 8. Measurement of compression law

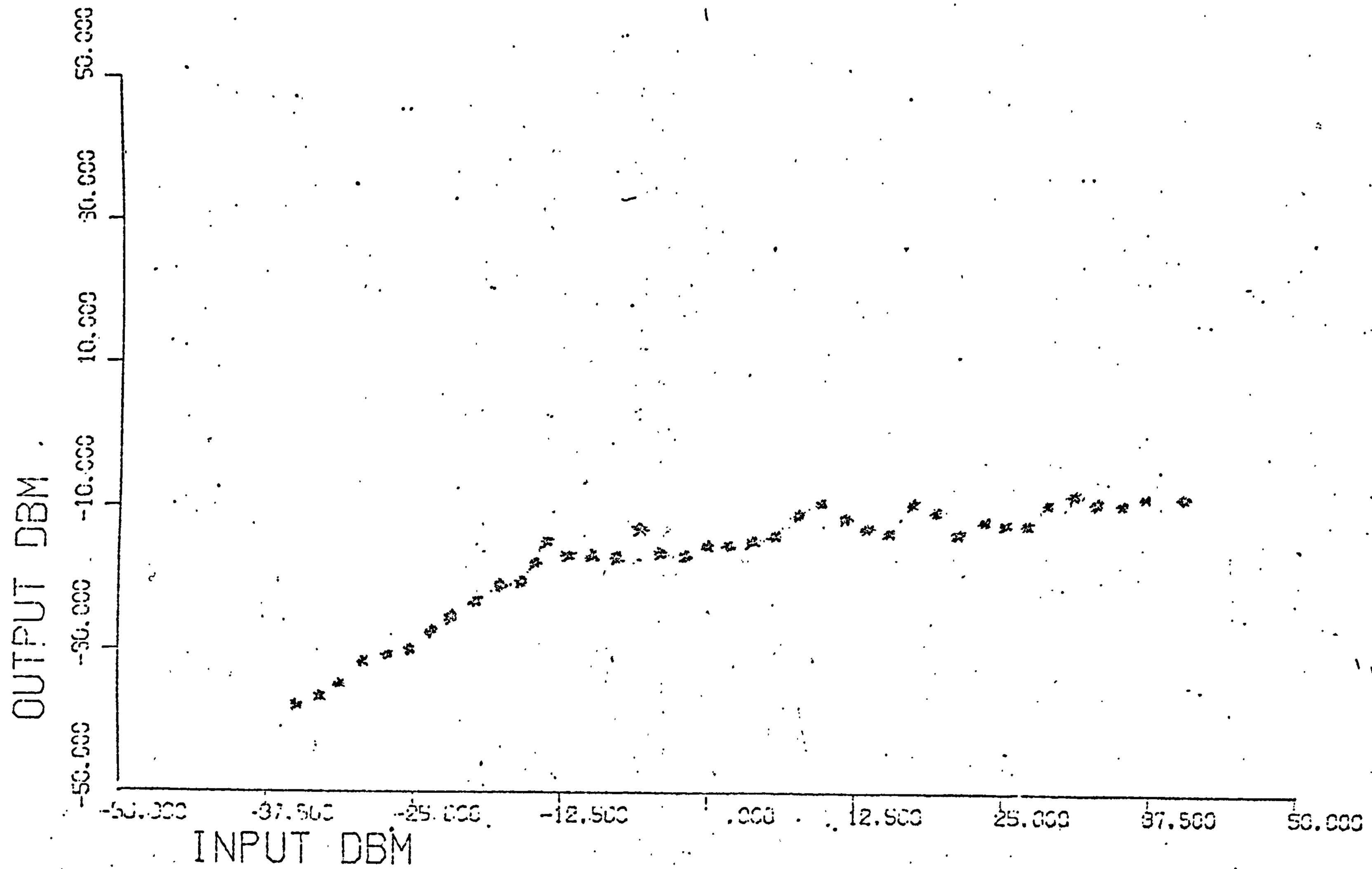

Figure (8)a Compression Characteristic of C.F.D.M.

2.12 THRESHOLD OF CODING

In the absence of input signal, the C.F.D.M. system will operate in the mode of adaptation by selecting the adaptation constant A_1, with reference to the principle of C.F.D.M. producing an alternate pattern of 1 0 1 0 1 0 .... in the L(t) waveform. If the input signal has a peak-to-peak amplitude less than the minimum allowable step size of the system δ min, the alternating pattern of 1 0 1 0 1 0 will not be disturbed and the output of the decoder will remain at zero. This minimum allowable step size δ min below which no information will be transmitted, is called the threshold of coding and can be represented as,

$$X_{min} = \delta \min \qquad \text{.....} \qquad (30)$$

2.13 OVERLOAD CONDITION

Theoretically, there is no limitation in dynamic range since all ranges of step size can be generated by C.F.D.M. But, since every equipment has voltage limitations which affect maximum and minimum step sizes in the feedback signal, it is desirable to consider an overload point of the C.F.D.M. coder. The overload point is governed by the limitation of the maximum allowable step size. The maximum size of the step has been limited to some hundred to 200 times the minimum allowable step size represented by δ_{max}.

Therefore, the overload condition has the relation given by,

$$x'(t) \geq \delta_{max} f_s \qquad \text{.....} \qquad (31)$$

where $x'(t)$ is the slope of the input signal and δ_{max} is the limited maximum step size and f_s is the sampling frequency. Thus for a sine wave input of $x(t) = X_{max} \operatorname{Sin}(\omega_m t)$, overload condition becomes,

$$X_{max}\, \omega_m = \delta_{max}\, f_s$$

$$\text{i.e.} \quad X_{max} = \frac{\delta_{max}\, f_s}{2\pi f_{c2}} \qquad \text{.....} \qquad (32)$$

For a Gaussian input signal rms slope of the input is given by,[1] $\delta_{max} 2\pi f_e$, where f_e is the effective band width of the signal given by,

$$f_e = \left[\frac{\int_{o}^{+fc_2} f^2 \, |X(f)|^2 \, df}{\int_{o}^{+fc_2} |X(f)|^2 \, df} \right]^{\frac{1}{2}} \quad \ldots\ldots\ldots\ldots \quad (33)$$

where fc_2 corresponds to the highest frequency to be transmitted and therefore the overload condition for a band-limited Gaussian signal input can be expressed as,

$$\sigma_{max} 2\pi \, f_e = \delta_{max} \, f_s \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (34)$$

i.e.

$$\sigma_{max} = \frac{\delta_{max} \cdot f_s}{2\pi f_e} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (35)$$

Therefore the maximum allowable input signal before the overloading occurs is,

$$X_{max} = \sqrt{2} \, \sigma_{max} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (36)$$

2.14 QUANTIZATION NOISE

In delta modulator type of systems, there are two types of quantization noise, granular noise and slope overload noise. Granular noise is determined by the instantaneous amplitude of the input signal and it occurs when a sufficient step size was used in the system. All of the quantization noise can be granular if very large step size is used. Slope overload noise is characterised by the slope of the input. This type of noise occurs when the slope of the input signal $x'(t)$ is greater than the maximum slope capability of the C.F.D.M. system. On the other hand, overload noise can occur when the maximum step size that C.F.D.M. could produce is limited to a certain value δ_{max}, so that

$$x'(t) > \delta_{max} \, f_s \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (37)$$

Referring to the figure (2), the quantization noise may be defined as,

$$n_q(t) = x(t) - x_o(t) \quad \ldots\ldots\ldots\ldots \quad (38)$$

where $x(t)$ is the input signal and $x_o(t)$ is the output from the decoder.

The noise power can be calculated as,

$$N_q^2 = \int_o^T |n_q(t)|^2 \, dt = 2\int_o^{+fc_2} |N(f)|^2 \, df \quad \ldots\ldots\ldots\ldots \quad (39)$$

where T and N(f) are the length of the noise signal and noise spectrum respectively.

2.15 AMPLITUDE RANGE OF C.F.D.M.

The amplitude range of the C.F.D.M. system is defined as the ratio of the maximum amplitude of the input signal consistent with the overload condition of the system as described previously, to the minimum peak amplitude, below which the input signal fails to excite the coder occurs when the peak-to-peak amplitude of the signal is smaller than the minimum allowable step size of the C.F.D.M. coder, described in equation.(30).

Hence, using equation (30) and equation (36) the amplitude range DR of the system is,

$$DR = 20.0 \log_{10} \left(\frac{X_{max}}{X_{min.}} \right) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (40)$$

where X_{max} and X_{min} may be calculated from the requirements referred to in Section (2.12) and (2.13).

2.16 SIGNAL-TO-NOISE RATIO

For a sine wave input, the mean square power that the C.F.D.M. handle is,

$$S^2 = X^2_{max} / 2 \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (41)$$

and the signal to noise ratio may be calculated using equation (39) as,

$$SNR = 10 \log_{10} \frac{S^2}{N_q^2} = 10 \log_{10} \left\{ \frac{X^2_{max}}{2\int_0^T |n_q(t)|^2 dt} \right\} = 10 \log_{10} \frac{X^2_{max}}{4\int_0^{fc2} |N(f)|^2 df}$$

$$\ldots\ldots\ldots\ldots\ldots\ldots \quad (42)$$

For the gaussian signal input SNR may be calculated as,

$$SNR = 10 \log_{10} \frac{S^2}{N_q^2} = 10 \log_{10} \left\{ \frac{\sigma^2_{max}}{\int_0^T |n_q(t)|^2 dt} \right\} = 10 \log_{10} \frac{\sigma^2_{max}}{2\int_0^{fc2} |N(f)|^2 dt}$$

$$\ldots\ldots\ldots\ldots\ldots\ldots \quad (43)$$

where σ is restricted by equation (36).

2.17 STABILITY

Due to the presence of a feedback loop, the possibility that the instability in some sense could arise, exists. Since the instantaneous output y(t) of the feedback group always tries to approximate the input signal x(t), the desired condition is for the feedback signal y(t) to make it wandersbetween the adjacent levels in such a way that the base-band components of the noise in the equation (38) are not objectionable. This requirements clearly states the necessity to make the error signal x(t) - y(t) as small as possible. Therefore, the stability of the coder entirely

depends on the design of the "local decoder" in the feedback loop, that produces pulses of various magnitudes to approximate the input x(t). Consequently, unnecessary large variations in the y(t) signal should be avoided.

Taking into account the above requirements, the stability of C.F.D.M. is studied as described in the following section.

2.17.1 RESPONSE OF C.F.D.M. TO AN IMPULSE

Figure (9) shows impulse response of C.F.D.M. plotting x(t) and the y(t) approximation to it. Actual input is pulse of width, '8' clock period and magnitude of 2.5 volts, since the pulse in normal use will be broadened due to band limitation of an impulse. There is a delay of '7' clock periods following the maximum amplitude in x(t). The feedback signal y(t) overshoots the input x(t) to a small degree but it oscillates continuously and finally takes up the smallest step size magnitude, after the fall of the impulse. This demonstrates the stability of the C.F.D.M. coder and the optimality of the selection of its adaptation constants.

2.18 EFFECT OF CHANNEL ERRORS

The calculation[9] of channel errors in linear delta modulation systems has been achieved for random errors. The success of this calculation results from the decoder being linear and time invariant, and enables the effect of transmission errors to be calculated by ignoring the presence of the signal. Figure 10 shows a detected binary signal, and an error signal, the latter being the difference between the transmitted and detected binary signals. The error waveform is therefore composed of pulses having amplitudes ± 2 and a duration of one clock period. The power due to the transmission errors at the output of the decoder is due to the decoded error waveform. The assumption made is that the true waveform and the error waveform are statistically independent

However, this approach is inapplicable for instantaneously adaptive delta modulation systems due to the non-linearity in the local decoders. The effect of an error depends on the history of the L(t) pulses sequence,

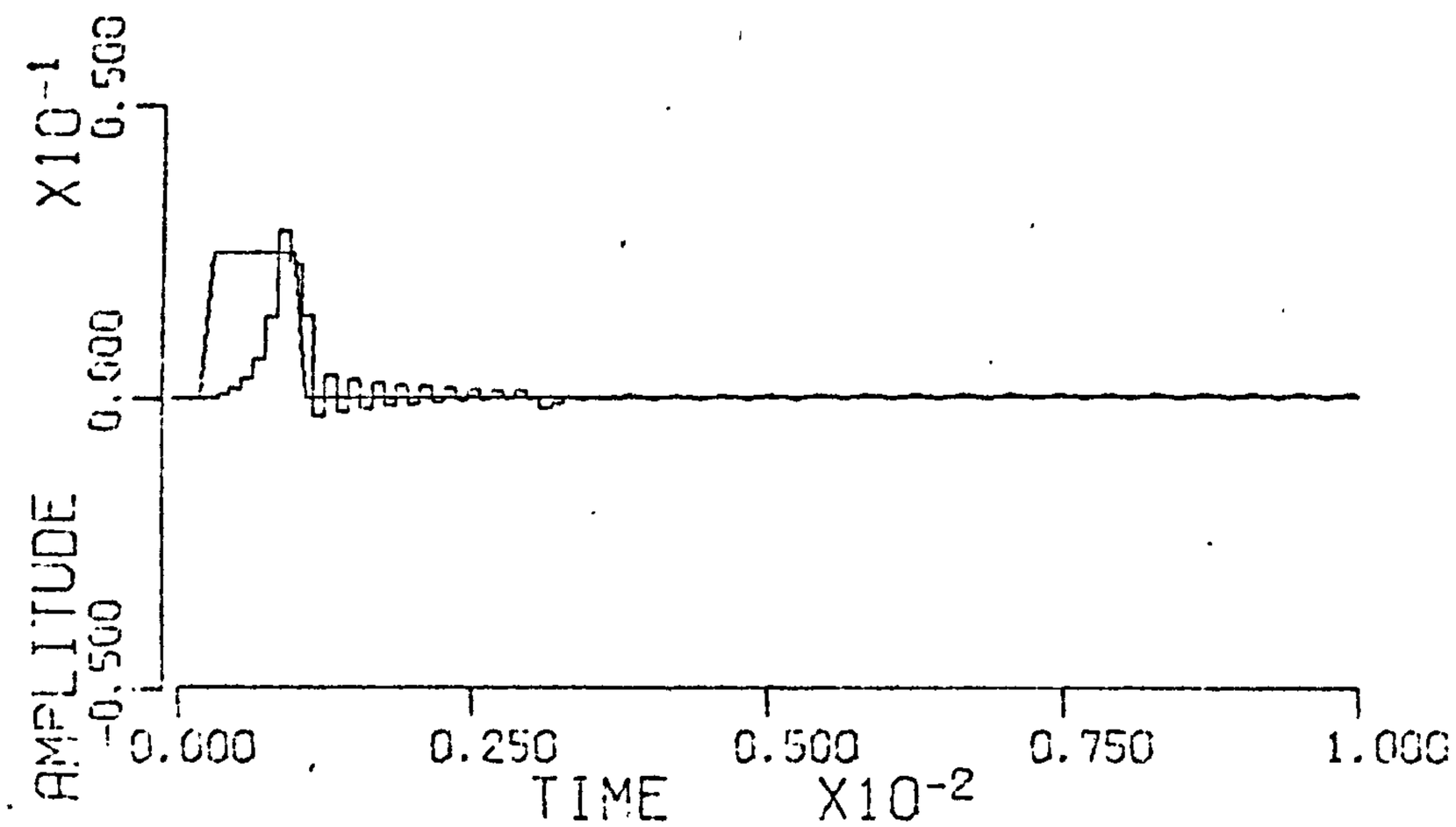

Figure (9) IMPULSE RESPONSE OF C.F.D.M.

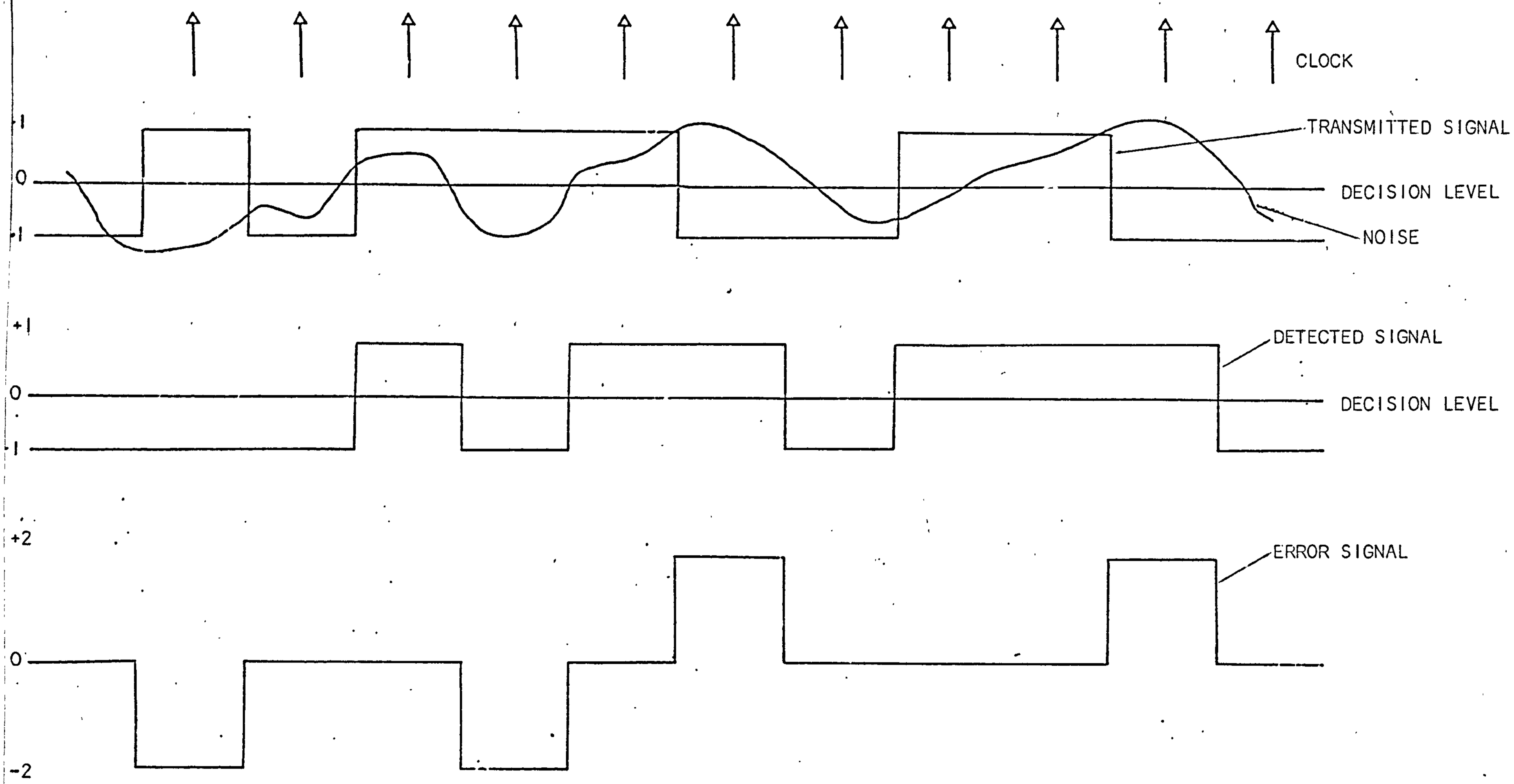

Figure (10) EFFECT OF CHANNEL NOISE ON RECEIVED SIGNAL

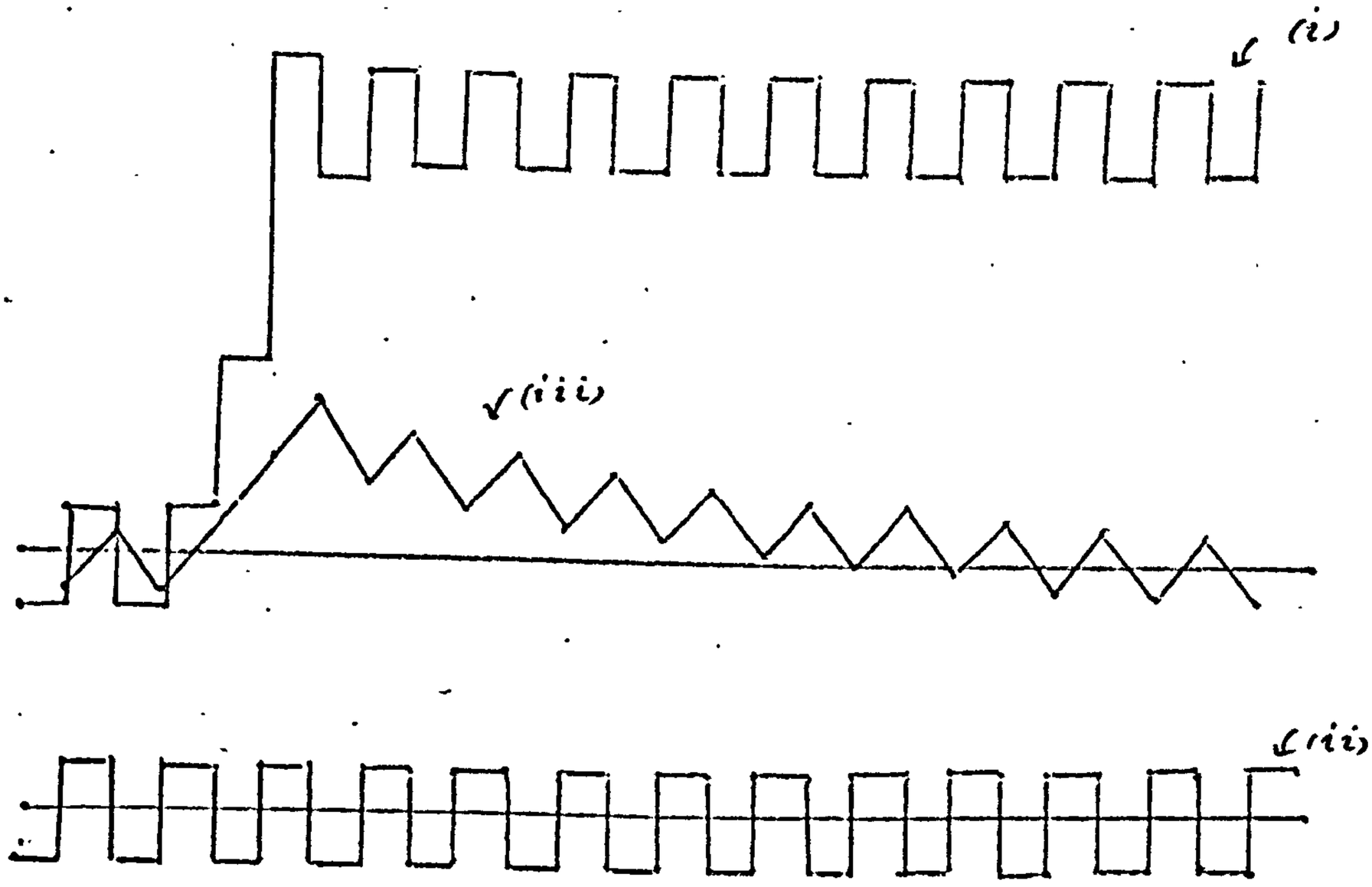

Fig. (10)a. C.F.D.M. RESPONSE TO A ZERO INPUT WITH A SINGLE CHANNEL ERROR AT 4th POSITION.

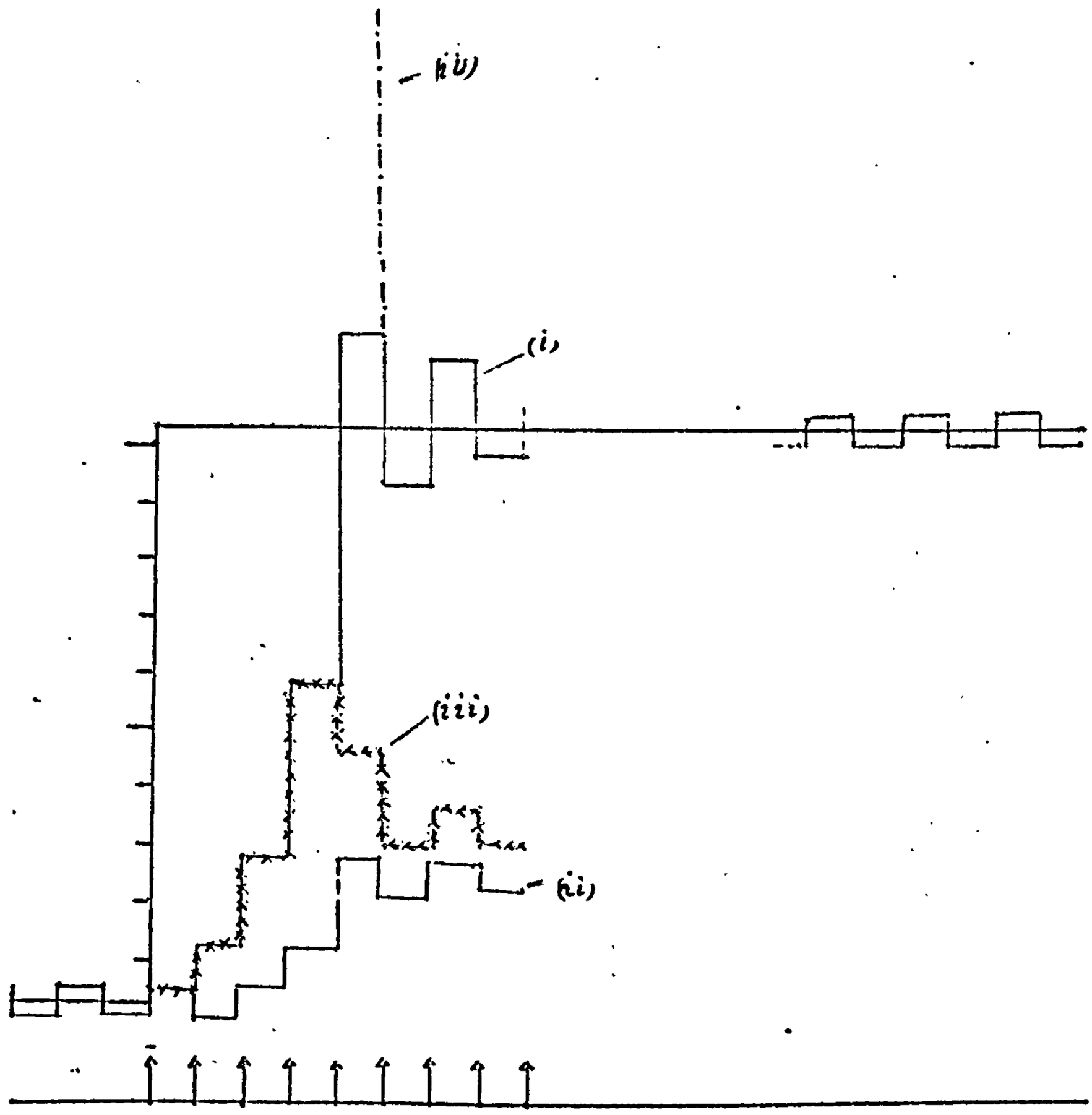

Fig. (10)b C.F.D.M. RESPONSE TO A STEP INPUT WITH A SINGLE CHANNEL ERROR AT 2nd, 5th and 6th POSITION.

and this phenomenon will now be demonstrated for the C.F.D.M. in response to an isolated channel error.

Figure (10)a(i) illustrates the persistant effects of a channel error which results at the output of the integrator in the decoder when a zero is wrongly interpreted by the decision circuit at the receiver as a logical one. Waveform (ii) applies to the errorless idling waveform which oscillates about zero, whereas the error idling waveform oscillates about 4.5 units. Although the computer simulations were done with perfect integrators, it is apparent that in a practical coder integrators must have some leakage in order to combat the effect of channel errors. Waveform (iii) shows that a leaky integrator enables the decoder to overcome the effect of this single channel error.

The effect of a single channel error when signals are being encoded is highly dependent on the location of the error pulse as previously mentioned. This is emphasised by reference to the step response shown in Figure (10)b when the single error occurs in different time positions. The input step is 21-0 units and response (i) is for no errors. Responses (ii), (iii) and (iv) at the output of the decoder are steps having values of 5.0, 6.6 and 80.9 respectively, and demonstrate the profound difference in decoded step sizes due to the location of the error pulse.

Figure 11 shows the CFDM tracking a sine wave with no errors, while figures 12 and 13 show the effect of one error occurring in different positions. It can be seen that the isolated transmission error has a level shift, phase change and distortion on the decoded waveform.

In the examples given above the persistent effect of a single error can be devastating at worse and significant at best. They have however been drawn for perfect integrators which means that these errors persist for all time. By making the integrators in both the encoder and decoder leaky the accommulative effects of these errors are overcome. Nevertheless, the considerable hierarchy in the transmitted binary signal makes C.F.D.M. vulnerable to channel errors. It is expected that in the presence of

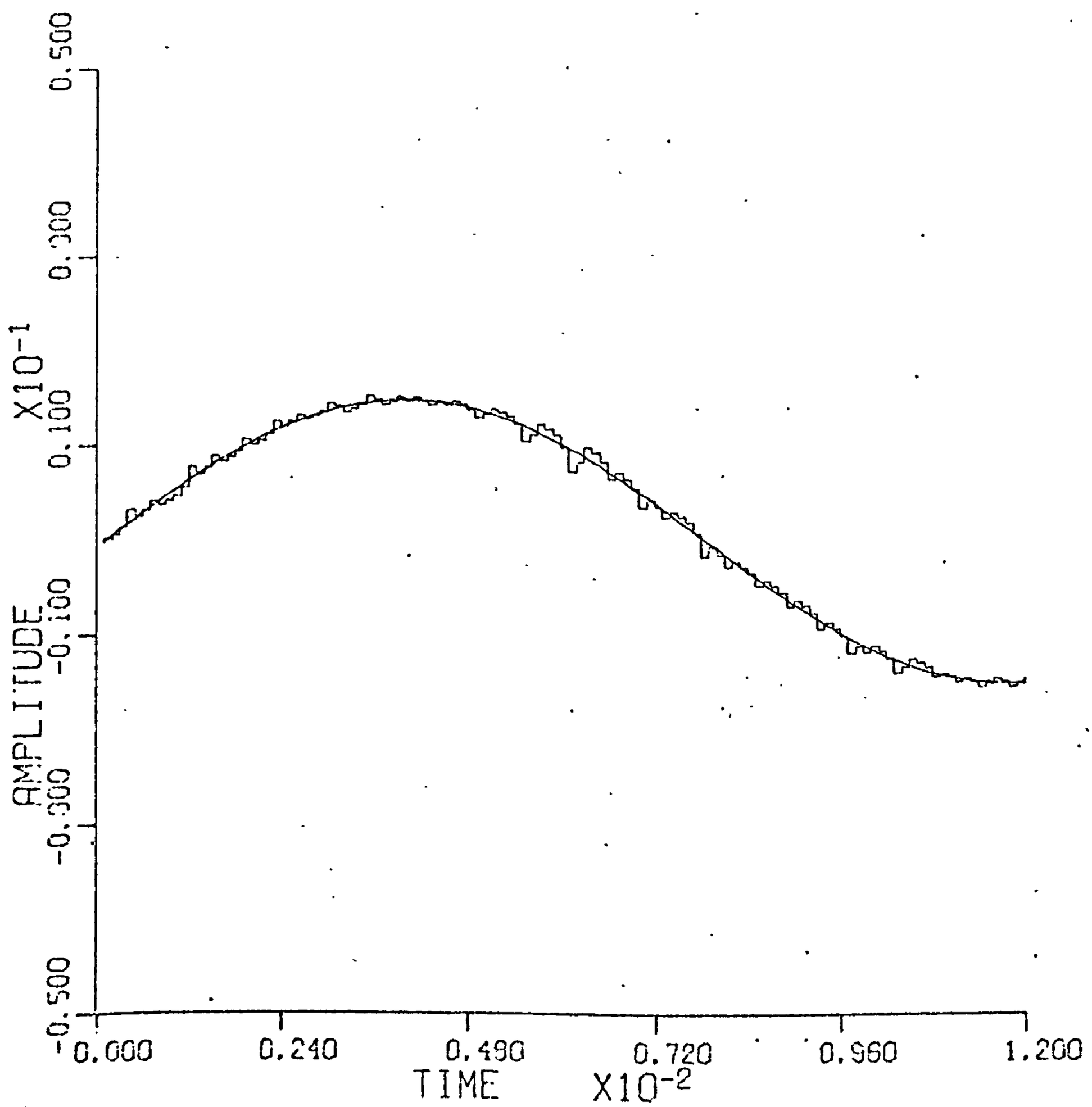

Fig. 11. C.F.D.M. RESPONSE TO A SINUSOID INPUT WITH NO CHANNEL ERRORS.

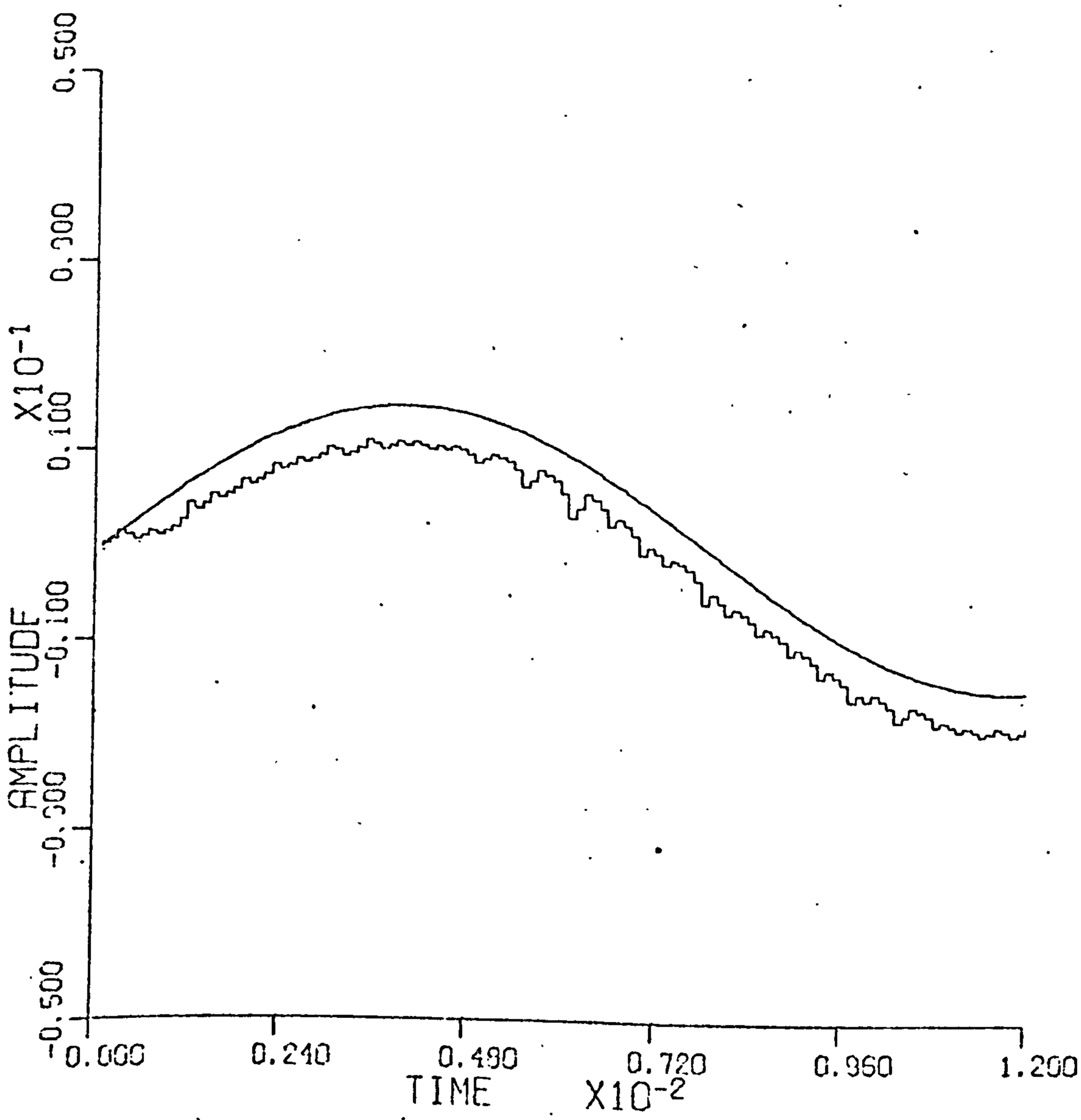

Fig. 12. C.F.D.M. RESPONSE TO A SINUSOID INPUT WITH A SINGLE CHANNEL ERROR AT 4th POSITION.

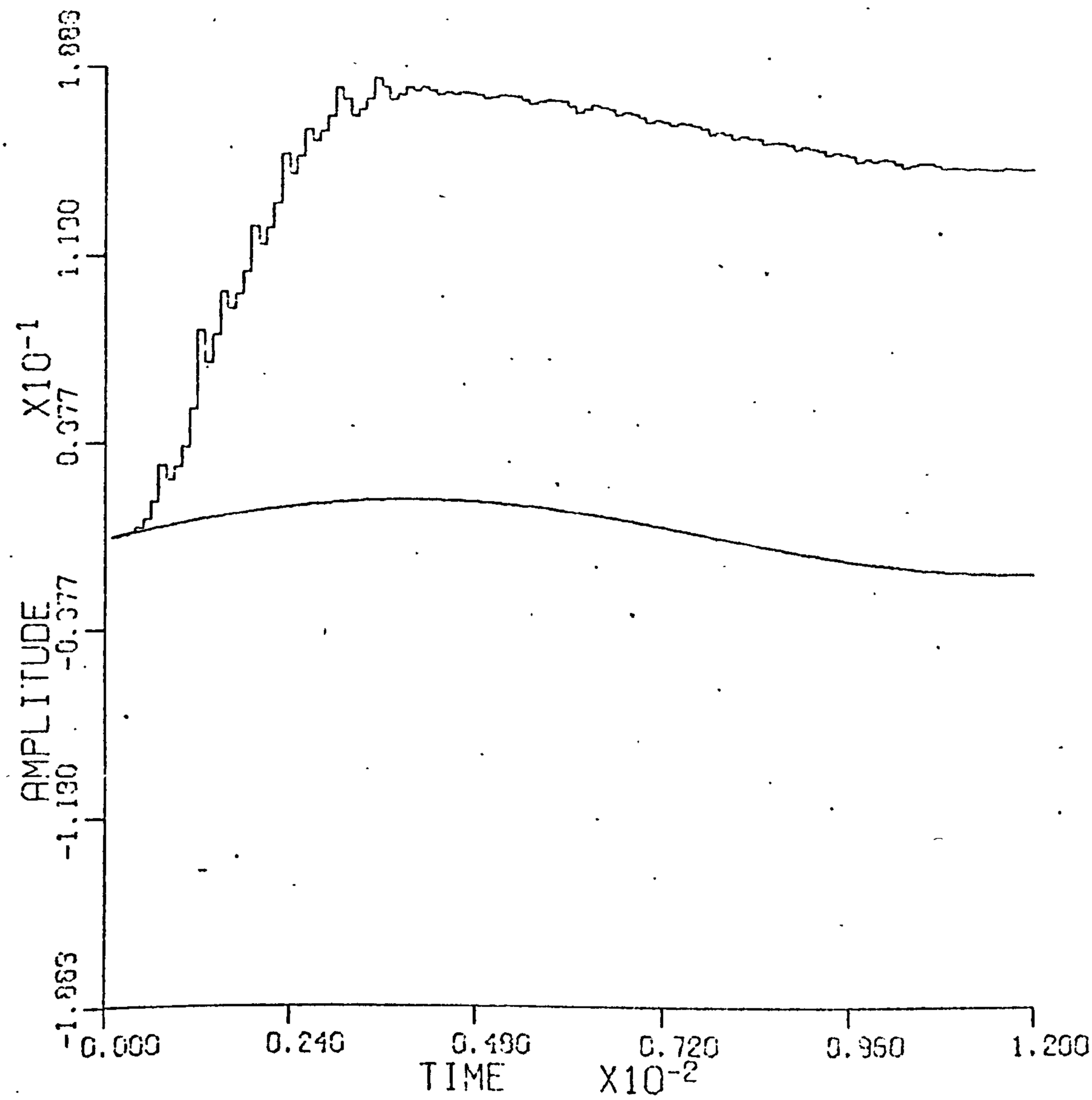

Fig. 13. C.F.D.M. RESPONSE TO A SINUSOID INPUT WITH A SINGLE CHANNEL ERROR AT 5th POSITION.

transmission errors this system would behave better than the h.c.d.m. system with its binary weighting, and worse than the J.A.D.M. system which has only two adaptation constants.

2.19 COMPARISON WITH LOGARITHMIC P.C.M.

The signal to noise ratios for the CFDM and the JADM systems are displayed in figures [16] and [25] for Gaussian inputs bandlimited to 3.1 KHz and a clock rate of 40 KHz. The peak signal-to-noise ratios of these systems are the same having a magnitude 21 dBs. It is expected that the peak signal to noise ratio of the system would diverge at higher ratios of (f_p/f_{c2}) because the full potential of the encoding algorithm of the CFDM would be realised. For example there would be less noise produced due to overshooting when tracking.

An A-law pcm system which samples the input signal at 8 KHz, i.e. above the Nyquist rate of 6.2 KHz, has a peak signal-to-noise ratio of 27.5 dB when A is equal to 87.6. The C.F.D.M. system operating in these conditions therefore has a performance approximately equivalent to that of a 4 bit A-law pcm system with A = 87.6.

2.20 EFFECT OF MISMATCH OF 'A' PARAMETERS ON SNR

We have assumed so far that the local decoders employed in the encoder and decoder have identically the same characteristics. However, it is not so in actual practice, due to the imperfections in the design of multipliers and mismatching of the values of the 'A' parameters used in the encoder and decoder.

The receiver may track the transmitter in polarities of the step: but not in the magnitudes of the step size. This type of effect is called sometimes the "mistracking" of the transmitter and receiver.

The result of this type of mistracking is referred to as the distortion of the scene in busy areas, rather than the introduction of noise when encoding television pictures. The combined effects of incorrect step size adaptation and the channel error, instability of some kind could arise.

Figure (14) displays the study of the effects of mismatch of 'A' adaptation parameters on signal to noise ratio, with Bandlimited Gaussian signal at (3) dBm input level. For the adaptation constants of $A_1 = -0.9$, $A_2 = -0.4$, $A_3 = 1.5$ and $A_4 = 2.0$ in the transmitter, the adaptation constants of $A_1 = -0.9 + \varepsilon$, $A_2 = -0.4 + \varepsilon$, $A_3 = 1.5 + \varepsilon$, $A_4 = 2.0 + \varepsilon$ are used in the receiver. The study has been made for the range of $\varepsilon = -0.2$ to $\varepsilon = +0.2$. These results indicate that the 'A' adaptation constants can have tolerances of the order of 10%, which results in a degradation of the signal to noise ratio of the order of 2 dBs.

2.21 C.F.D.M. WITH LARGE VALUES OF MEMORY LENGTH 'm'

The extension of the coder's memory length 'm' is not expected to cause any difficulty and in doing this it will increase the cost only marginally.

It has been described in section (2.2) that, for a memory length of 'm', the number of possible binary groups formed in the logic circuit is given by $N = 2^n/2$, where $n = m+1$. The number N also gives the number of adaptation constants needed for the system and the value of 'm' gives the order of C.F.D.M.

Thus, for example, C.F.D.M. with a memory length of $m = 3$, the extension could be achieved by again re-defining the pattern of each of the groups formed from table (2) in the C.F.D.M. with $m = 2$. Re-defining the pattern of each group could be made on the basis of how each of the patterns of the group could have been derived from if one more extra memory length was added. Table (3) illustrates how

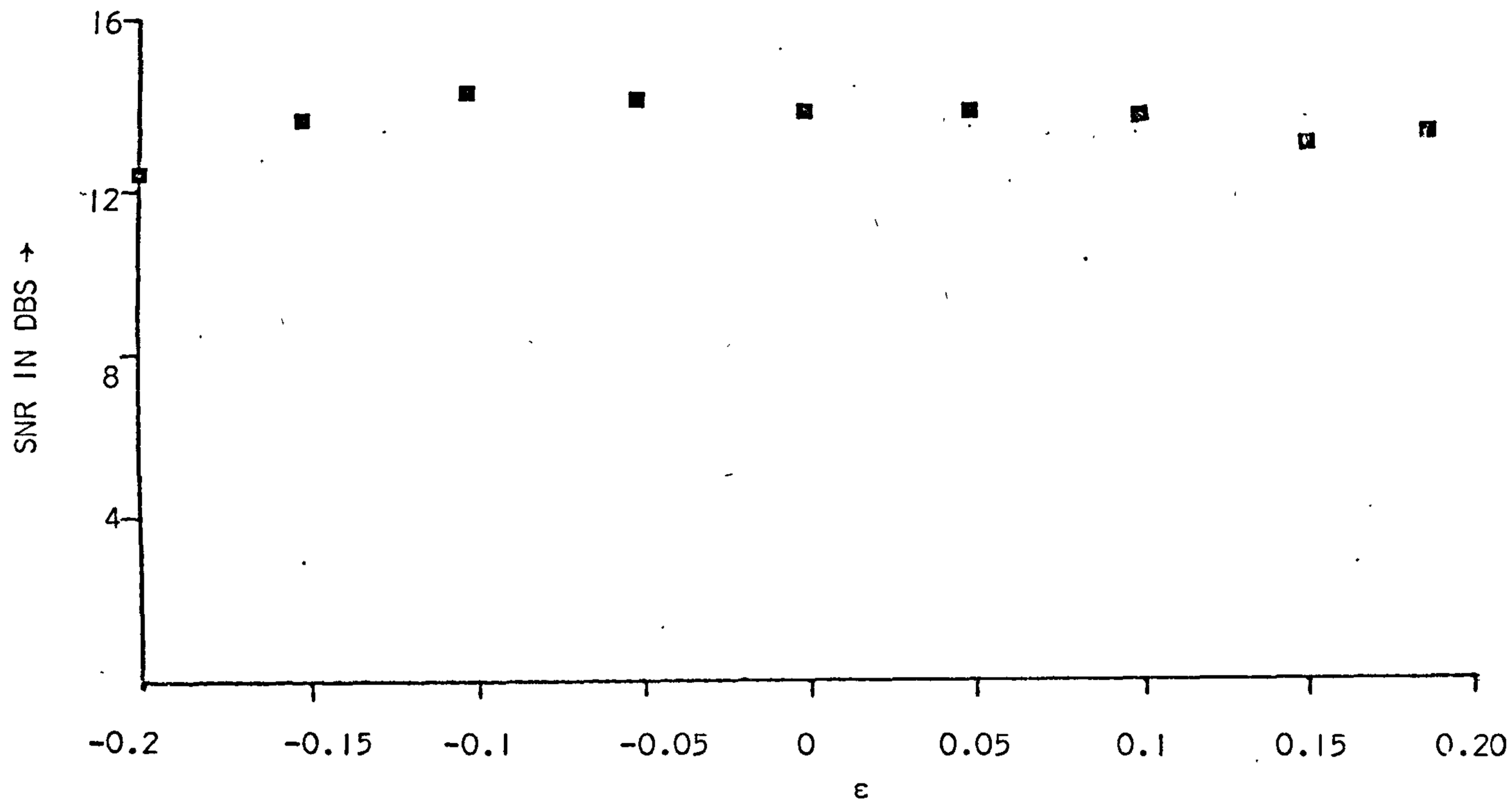

Figure (14) EFFECTS OF MISMATCH OF 'A' PARAMETERS ON SNR

GROUP NO. OF 2nd ORDER CFDM	L_r	L_{r-1}	L_{r-2}	L_r	L_{r-1}	L_{r-2}	L_{r-3}	BIN.PAT.NO OF 3rd ORD C.F.D.M.
1	0	1	0	0	1	0	0	1
				0	1	0	1	2
	1	0	1	1	0	1	0	3
				1	0	1	1	4
2	0	1	1	0	1	1	0	5
				0	1	1	1	6
	1	0	0	1	0	0	0	7
				1	0	0	1	8
3	0	0	1	0	0	1	0	9
				0	0	1	1	10
	1	1	0	1	1	0	0	11
				1	1	0	1	12
4	0	0	0	0	0	0	0	13
				0	0	0	1	14
	1	1	1	1	1	1	0	15
				1	1	1	1	16

Table (3) POSSIBLE BINARY PATTERNS OF 3rd ORDER C.F.D.M.

GRP.NO. OF 2nd ORDER C.F.D.M.	L_r	L_{r-1}	L_{r-2}	L_r	L_{r-1}	L_{r-2}	L_{r-3}	GRP.NO. OF 3rd ORDER C.F.D.M.
1	0	1	0	0	1	0	1	1
				1	0	1	0	
	1	0	1	0	1	0	0	2
				1	0	1	1	
2	0	1	1	0	1	1	0	3
				1	0	0	1	
	1	0	0	0	1	1	1	4
				1	0	0	0	
3	0	0	1	0	0	1	0	5
				1	1	0	1	
	1	1	0	0	0	1	1	6
				1	1	0	0	
4	0	0	0	0	0	0	1	7
				1	1	1	0	
	1	1	1	0	0	0	0	8
				1	1	1	1	

Table (4) POSSIBLE BINARY GROUPS OF 3rd ORDER C.F.D.M.

each pattern of group 1, 2, 3 and 4 of 2nd order C.F.D.M. could be formed from the memory length of three. In this type of 3rd order C.F.D.M., the coder adapts itself according to the presented binary level L_r and the three previous binary levels L_{r-1}, L_{r-2} and L_{r-3}. Thus the logic circuit is confronted with $2^{m+1} = 2^4 = 16$ possible binary patterns as shown in Table (3). These can be conveniently reduced to eight groups with each group having the binary patterns of the same characteristic, as shown in Table (4).

In this type of C.F.D.M. system, eight adaptation constants will be needed, namely A_1 to A_8 corresponding to groups 1 to 8 respectively. The evaluation of these adaptation constants may be made iteratively as has been done for the 2nd order C.F.D.M.

CHAPTER 3

COMPUTER SIMULATIONS

3.1 INTRODUCTION

All the simulations were performed on I.C.L.1904A digital computer at the University of Loughborough. Basically, the simulation of a delta modulator system of Fig.(2) comprises the simulation of input signal, the encoding and decoding functions of the system.

The simulation of C.F.D.M. system was first made. Simulation of similar types of adaptive delta modulators were performed by introducing different adaptation algorithms for selecting appropriate 'A' parameters, which satisfy the corresponding system's requirements. Specifically three types of instantaneously adaptive delta modulators and one linear delta modulator were being simulated, namely,

(i) C.F.D.M.

(ii) JAYANT's Adaptive delta modulator (J.A.D.M.)

(iii) WINKLER's H.I.D.M., and,

(iv) L.D.M.

A description of the simulation of each of these different adaptive delta modulators will not be given in detail on its own but instead they will be described in terms of the principles of our C.F.D.M. coder, since each of them can be considered as the special case of the C.F.D.M. coder, having different sets of 'A' parameters. This generalised principle of C.F.D.M. coder will be mentioned in Section (2.8).

3.2. SIMULATION OF C.F.D.M.

First of all, the description of the simulation of C.F.D.M. coder will be made. The C.F.D.M. system was simulated by making use of the adaptation algorithm described in Section (2.6). The assumptions made in Chapter 2 have been taken into consideration. By doing so the matching of the encoder and decoder characteristics were achieved. Throughout the simulation the minimum and maximum allowable step sizes were limited to

± 0.03 and ± 5.0 volts respectively unless otherwise stated for some simulation purposes. The 'A' parameters used for the simulation of C.F.D.M. were the same as described in Section (2.7) having the magnitudes of $A = -0.9$, $A_2 = -0.4$, $A_3 = 1.5$ and $A_4 = 2.0$. The flow chart of the main programme organised for the simulation of Figure (2) is shown in Figure (39)a of the Appendix. The various sub-programmes that will fulfill the purposes of the main programme are shown in Appendix (A.2) and the purposes of these sub-programmes are described clearly.

3.3 RESPONSE TO BANDLIMITED GAUSSIAN SIGNALS

The Gaussian amplitude distributed signals were generated internally by the University computer. It used the random number generator in the form of function UTRI, which employs the linear feedback shift register technique. Approximately 8 million numbers were generated before the sequence repeats. By setting the arguments (J,K,L) of the function UTRI, accordingly, the random numbers generated can have either a uniform probability distribution in the range 0.0 to 1.0 or a Gaussian probability distribution in the range -6.0 to +6.0 with unit variance.

The algorithm for the random number generator could be written in the form,

X = UTRI (J,K,L). A.

where X = a real variable which will contain the random number generated.

J = An integer which defines the stream number. There are 4 possible sequences of random numbers, these are obtained by setting J = 1, 2, 3 and 4. J = 1 to the time inverse of J = 4 and J = 2 is the time inverse of J = 3.

K = an integer variable which controls the distribution of the random numbers. If K = 0, the random numbers have a uniform probability

distribution in the range 0 to 1.0.

If $K \neq 0$ the random numbers have a Gaussian probability density function in the range -6.0 to +6.0 with unit variance.

L = an integer variable which contains the random number generator.

Band limitation of the Gaussian signals was easily achieved by passing these random numbers through a digital low pass filter having a relatively sharp cut-off frequency. The design of this type of digital filter is described later separately. The filter used has the gain characteristic shown in Figure (15). The filter characteristic is down 3 dB at 3.1 kHz, and that is why we call the bandwidth of the filter as 3.1 kHz. The filtering was accomplished by convolving the input signal samples which we wanted to be filtered with the impulse response samples of the filter.

3.3.1 DYNAMIC RANGE OF C.F.D.M.

The input was a flat band limited Gaussian signal, bandlimited to 3.1 kHz, sampled for the simulation at 40 kHz. A wide range of about 75.0 dBm input was applied to the C.F.D.M. coder and the decoder signal to noise ratio in dBs, against input signal power in dBm were plotted as shown in figure (16).

It offers a maximum signal to noise ratio of 21.0 dBs and the dynamic range of 50.4 dBm for 16.5 dBs SNR.

Waveforms for the input x(t) and y(t) signals are plotted in Figure (17). This can be compared with the response of J.A.D.M. shown in Figure (25). The tracking ability of C.F.D.M. is better than J.A.D.M.

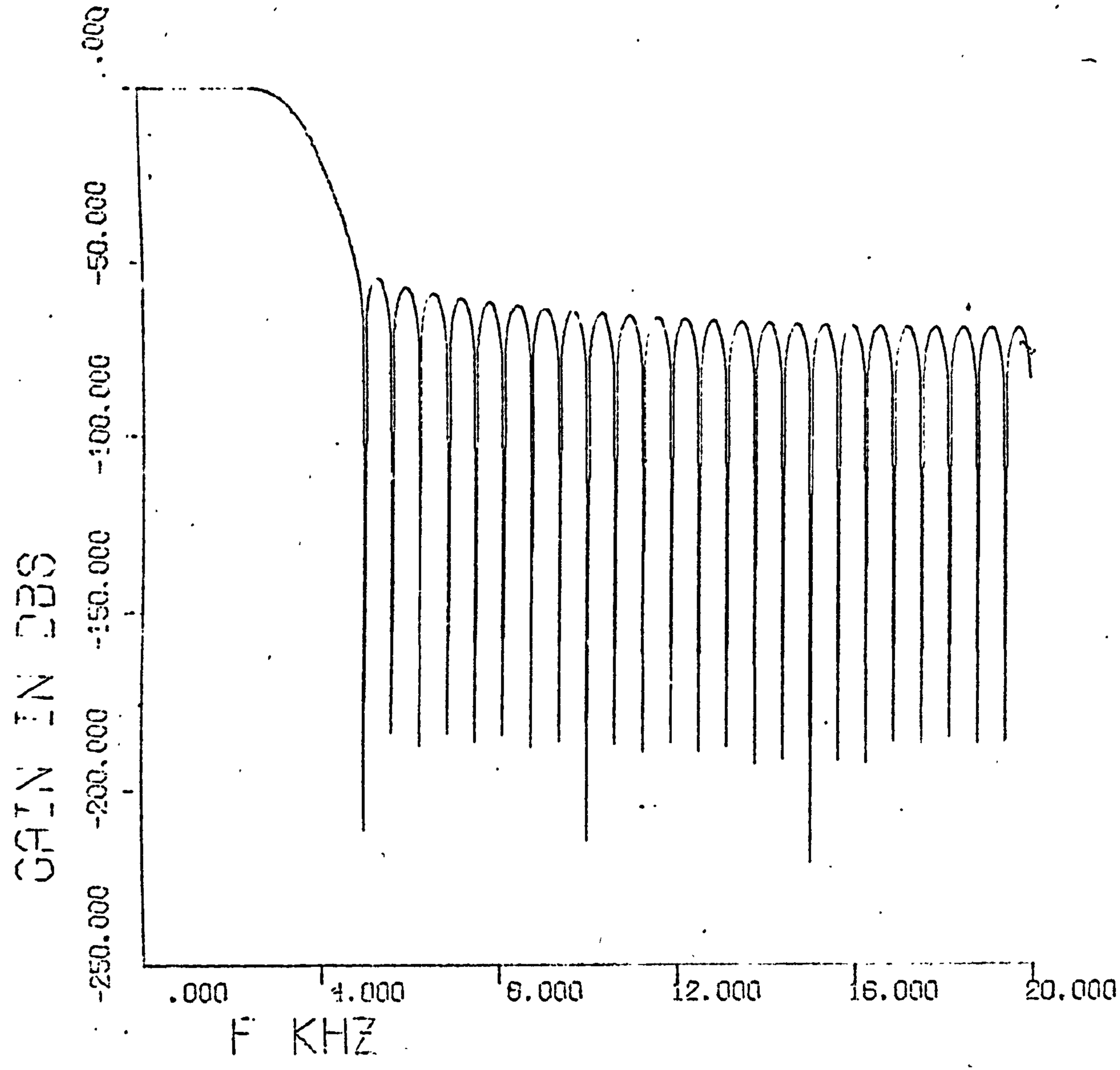

Figure (15) DIGITAL LOW-PASS FILTER GAIN CHARACTERISTICS.

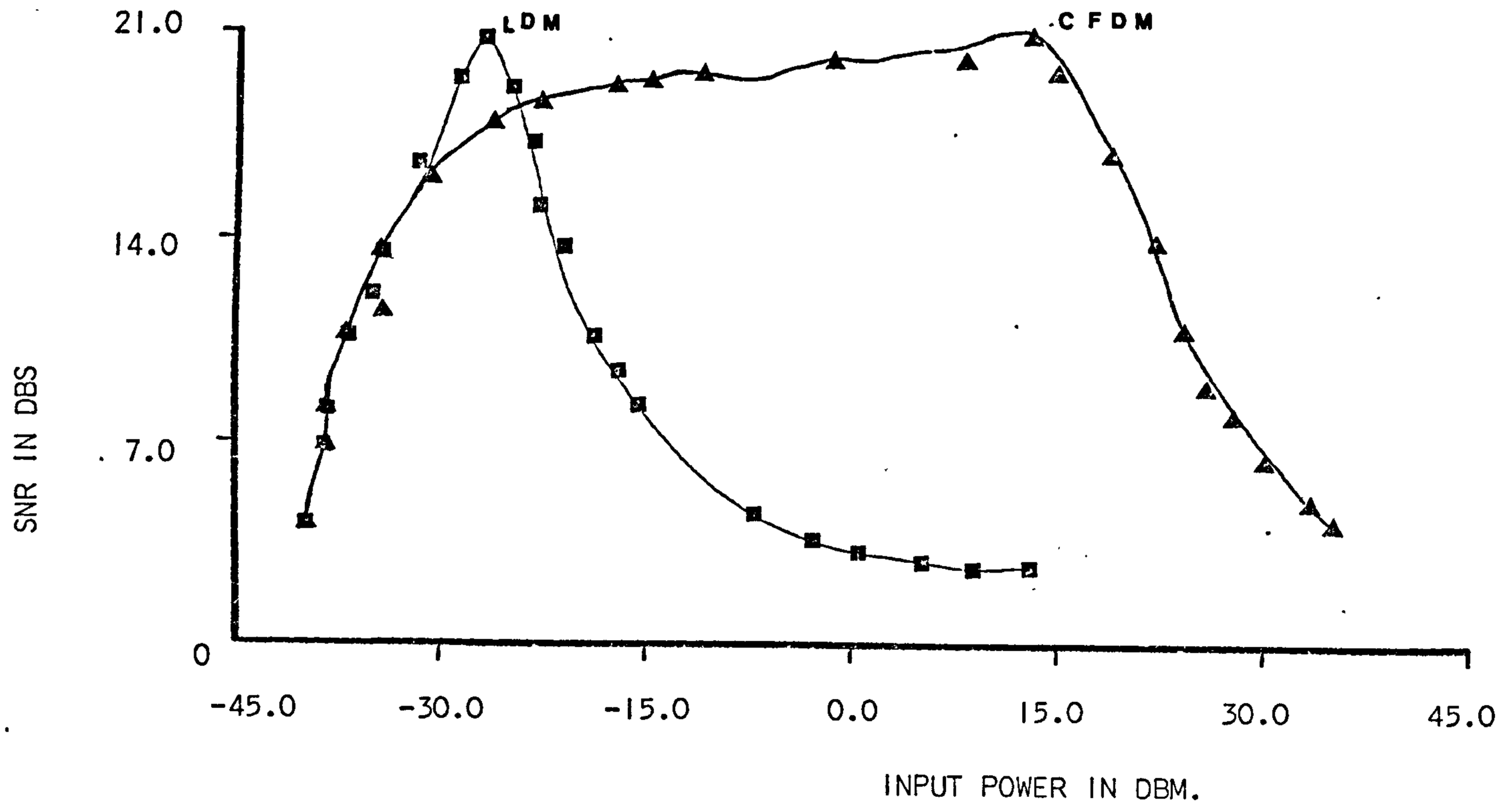

Figure (16) DYNAMIC RANGE CURVES OF C.F.D. AND L.D.M.

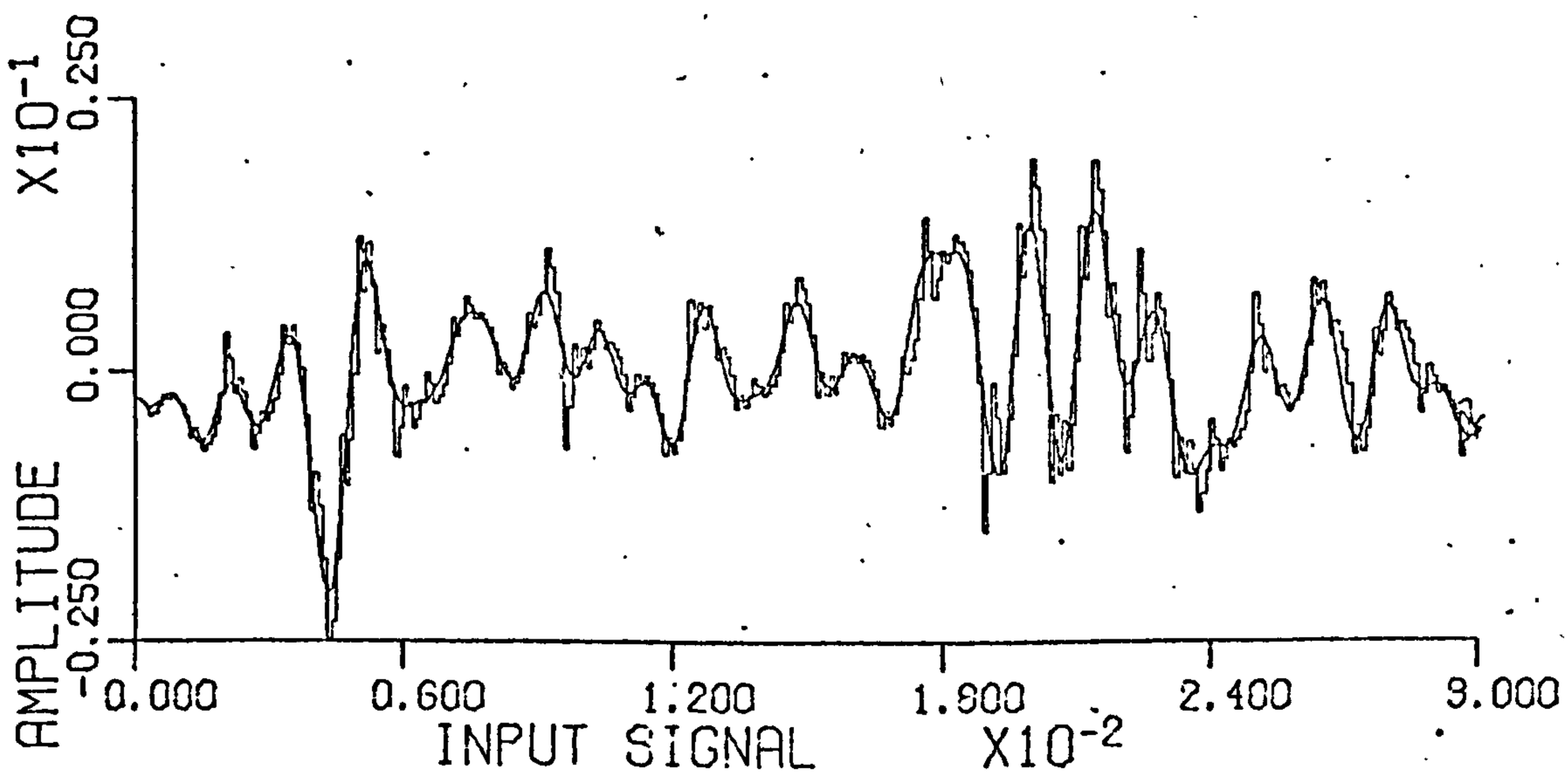

Figure (17) COMPUTER GENERATED WAVEFORMS OF INPUT AND RECONSTRUCTED SIGNAL OF C.F.D.M.

3.4 CALCULATIONS OF SIGNAL-TO-NOISE RATIO (SNR)

There are several ways of calculating signal to noise ratio (SNR) achieved by the system. Some research workers in this field have calculated SNR before the final filtering in the decoder, some after the final filtering. Some have performed the calculation of SNR by filtering the error waveform in the encoding side. In our analysis the calculation was made in frequency domain, following the definition of SNR described later. F.F.T. algorithm rigorously used calculating the D.F.T. (Discrete Fourier Transform) of the time signals. The programmes written for the calculation of SNR is shown in Appendix (A.2) and its functions were explained. This method of calculating SNR will be used throughout the work.

3.4.1 DEFINITION OF SNR

The signal to noise ratio (SNR) was defined as the ratio of the input signal power to the power in the noise signal calculated over the message frequency band. The noise signal was defined as the difference between the input signal and the decoded output signal such that

$$e(t) = x(t) - x_o(t) \qquad \ldots\ldots\ldots\ldots \qquad (52)$$

3.4.2 DEFINITION OF D.F.T.(Discrete Fourier Transform)

Since the manipulation of SNR was carried out in frequency domain and made use of the F.F.T.(Fast Fourier Transform), it is worth while here to recall briefly the D.F.T.

The D.F.T. of a set of N numbers g_K, K = 0, 1, 2 .... N - 1 is a set of N Fourier coefficients G_ℓ, ℓ = 0, 1, 2 ..... N - 1 defined by the expression

$$G_\ell = \sum_{K=0}^{N-1} g_K \; e^{-j2\pi K\ell/N}, \quad \ell = 0, 1, = \ldots\ldots N-1 \qquad \ldots\ldots\ldots \qquad (53)$$

and its inverse transform is

$$g_K = \frac{1}{N} \sum_{\ell=0}^{N-1} G_\ell \; e^{j2\pi K\ell/N} \quad , \quad K = 0, 1, 2 \; ,\ldots N-1 \quad \ldots. \tag{54}$$

where $j = \sqrt{-1}$

3.4.3 THE F.F.T.

The fast fourier transform[18] is a class of algorithm for efficiently computing the D.F.T. of a sequence of data samples. It greatly reduces the number of computations required to calculate the fourier transform of a set of numbers on a digital computer.

3.4.4 THE ALGORITHM FOR THE CALCULATION OF SNR

There are two sets of signal samples that we are dealing with in our calculation of SNR. These are the input signal x(t) and noise signal e(t), which are defined earlier in analysis.

Suppose the spectrum of x(t) and e(t) are attainable by taking the F.F.T. and denoted by

$$x(t) \xrightarrow{F.F.T.} X(f)$$

$$e(t) \xrightarrow{F.F.T.} E(f)$$

Thus,

$$X_\ell = \sum_{K=0}^{N-1} x_K \; e^{-j2\pi K\ell/N}, \quad \ell=0, 1, 2 \ldots\ldots N-1 \quad \ldots\ldots \tag{55}$$

and

$$E_\ell = \sum_{K=0}^{N-1} e_K \; e^{-j2\pi K\ell/N}, \quad \ell=0, 1, 2 \ldots\ldots N-1 \quad \ldots\ldots \tag{56}$$

where N is the number of signal samples under consideration.

(NOTE: N is made equal to 2^n where n is any integer, so that the F.F.T. algorithm can operate on the signals)

3.4.4-1 ALGORITHM FOR INPUT SIGNAL POWER CALCULATION

Though the low pass filter used in the simulations had a relative sharp cut-off frequency at 3.1 kHz, the use of F.F.T. in the design of the filter, allowed the lowpass filter characteristic to extend up to half the sampling frequency as shown in Figure (15). To pretend the low-pass filter had a rectangular sharp cut-off at 3.1 kHz, the input signal power is calculated up to 3.1 kHz band. Since the D.F.T. of a real sequence of time signals gives a two sided spectrum, symmetric about (N/2) + 1 sample, the calculation of the total signal power can be made by using the one side of the spectrum and multiplying the result by two except for the first sample of the spectrum.

Suppose the sampling frequency is f_s and there are N frequency samples, then the elementary frequency spacing between each frequency sample is

$$\Delta f = \frac{fs}{N-1} \quad \text{Hz} \qquad \ldots\ldots\ldots\ldots\ldots \qquad (57)$$

If the frequency band of the message signal is from 0 - f_{c2}, then this frequency interval must contain a certain number of elementary frequency band, in the sense of our treatment of variability.

Therefore, the number of frequency band can be calculated as,

$$M = \frac{\text{message band}}{\text{elementary freq. band}} + 1$$

$$= \frac{f_{c2}}{\Delta f} + 1 \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \qquad (58)$$

Thus the algorithm for calculating signal power using single sided spectrum is,

$$S^2 = 2.0 \sum_{\ell=1}^{M} |X(\ell)|^2 + |X(0)|^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots \qquad (59)$$

3.4.4-2 ALGORITHM FOR NOISE POWER

Noise power is calculated from the error spectrum E_ℓ, by taking only the frequency components that reside in the message frequency band as previcusly described. Therefore, following the same reasoning the algorithm for the calculation of noise power can be written as,

$$N_q^2 = 2.0 \sum_{\ell=1}^{M} |E(\ell)|^2 + |E(0)|^2 \quad \ldots\ldots\ldots\ldots \quad (60)$$

3.4.5. SIGNAL TO NOISE RATIO EQUATION

From the definition, SNR may be calculated from the relation

$$SNR = 10.0 \log_{10} \left(\frac{S^2}{N_q^2}\right) \quad \ldots\ldots\ldots\ldots \quad (61)$$

3.5 STEP RESPONSE OF C.F.D.M.

A step input of 39.5 volts was applied to the coder. The input stayed at 0.5 volt for the first 3 clock periods and the input stepped up to 39.5 volts. The magnitude of this step is the same as that used by Winkler in his article[9]. Figure (18) shows the step response of C.F.D.M. (6) clock periods are taken to catch the input step demonstrating its fast response. It has a faster response and the overshooting is less than J.A.D.M. and H.I.D.M. After 8 clock periods, it hunts the step input with a minimum step provided for the system. It retains the basic ... 1 0 1 0 1 0 ... pattern while tracking the steady input of the step.

3.6 STEP SIZE DISTRIBUTION OF C.F.D.M.

Figure (19) shows the step size distribution of the C.F.D.M. system. Input signal is bandlimited to 3.1 kHz, at the sampling rate of 40 kHz, having the gaussian amplitude distribution with zero mean. Input signal level is about 3 dBm. The step size distribution has a symmetry about the zero and follows the statistical behaviour of the input signal which is very essential. It demonstrates the multi-

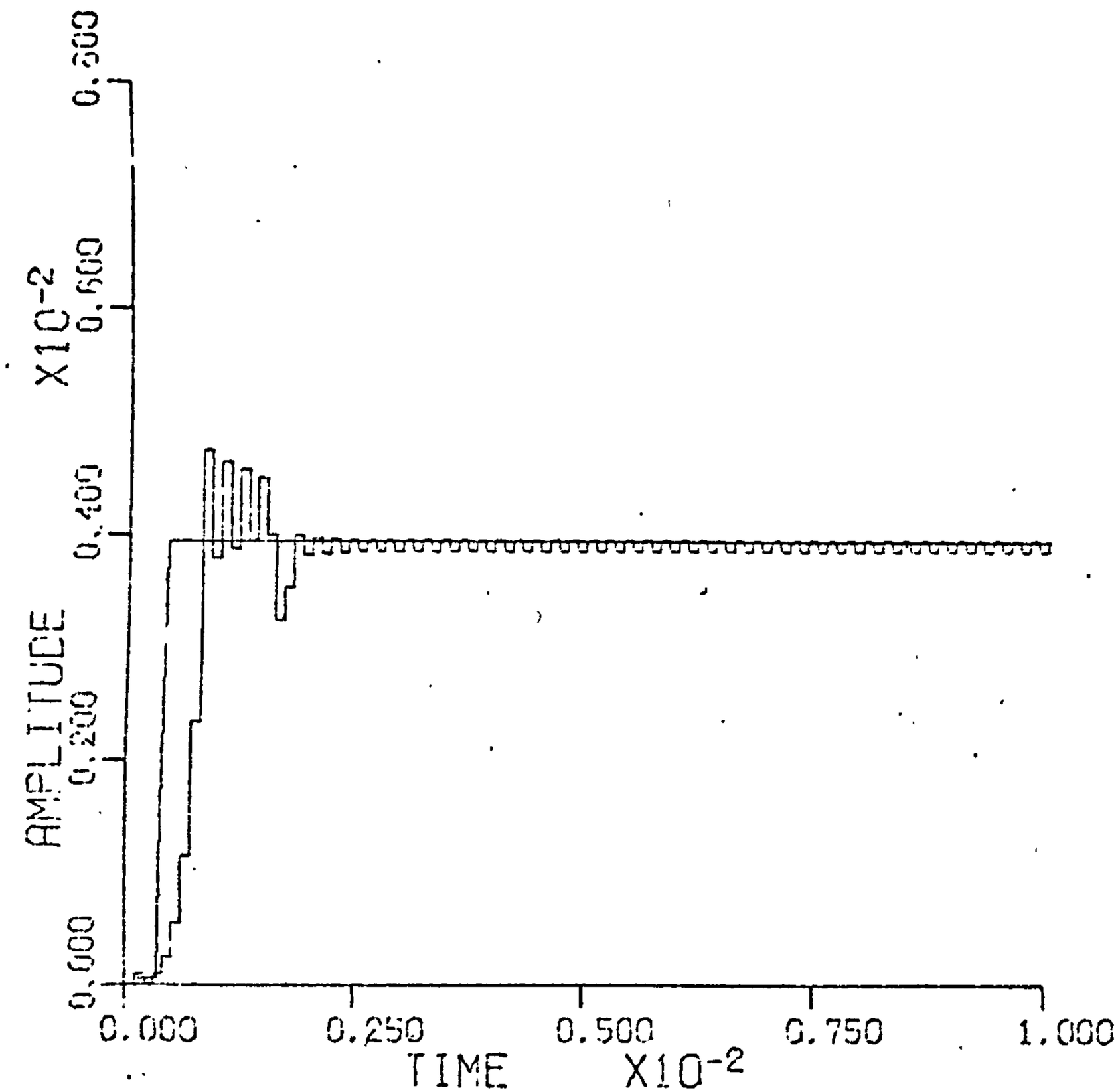

Figure (18) STEP RESPONSE OF C.F.D.M.

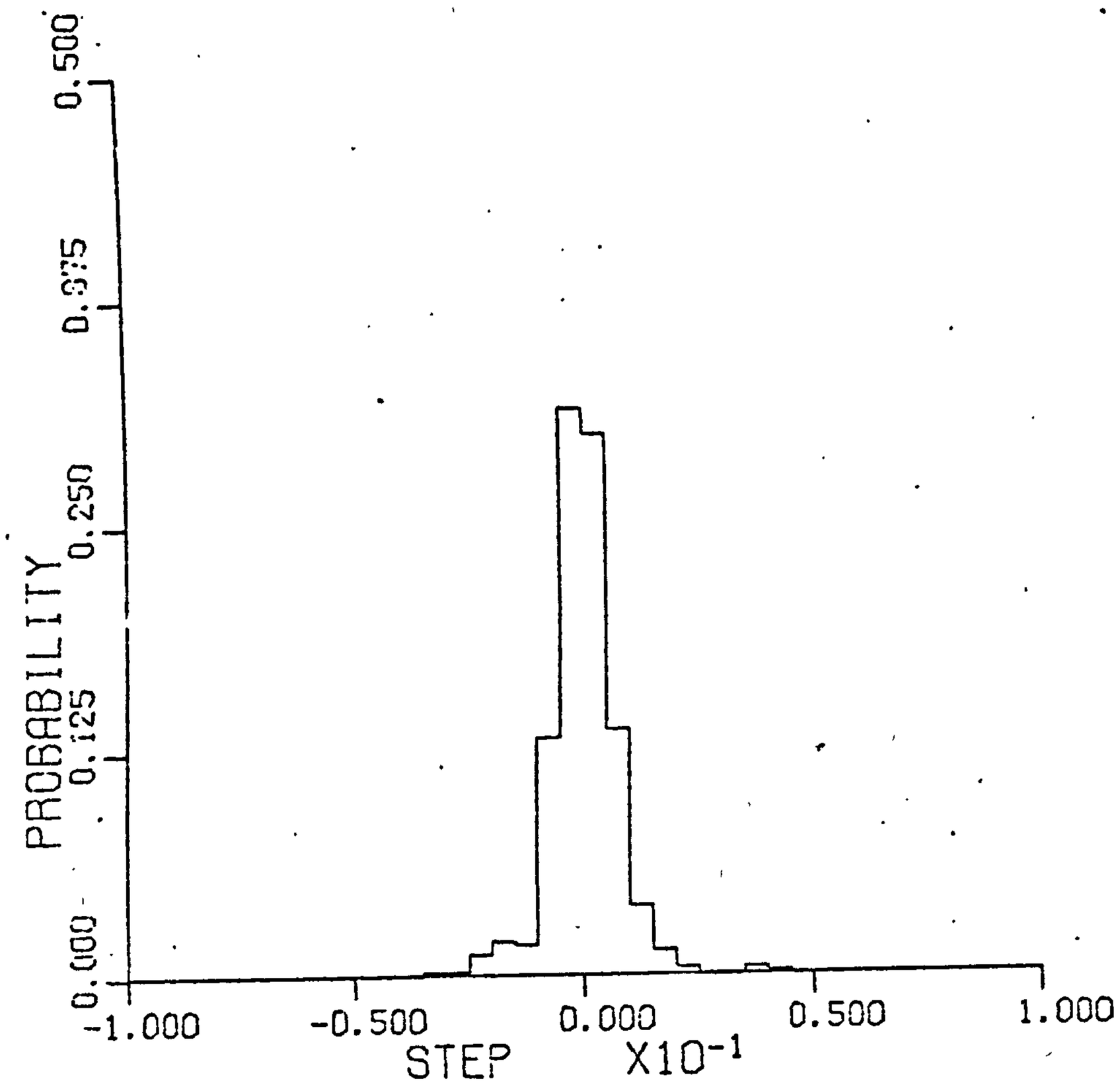

Figure (19) STEP-SIZE DISTRIBUTION OF C.F.D.M.

levelled nature of the step size dictionary achieved with just two bit memory in our C.F.D.M.

3.7 ESTIMATION OF POWER SPECTRUM

This work has been done to analyse the distortion power spectrum of the C.F.D.M. coder. Since we are dealing with finite data length, in our analysis, the spectrum we are getting by taking the fourier transform of this set of data will not be a true spectrum of the actual signal under investigation. Why is this so? This can be easily explained by realising the behaviour of the finite length data.

To investigate into this, it is necessary to discuss the convolution properties of fourier transforms. Supposing we have a signal which is the product of two other signals,

$$x(t) = y(t) \cdot c(t) \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (62)$$

The fourier transform of x(t) will give

$$X(f) = \int_{-\infty}^{+\infty} x(t)\, e^{-j2\pi ft}\, dt \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (63)$$

$$= \int_{-\infty}^{+\infty} y(t) \cdot c(t)\, e^{-j2\pi ft}\, dt \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (64)$$

c(t) can be substituted in terms of its fourier transform $C(f_o)$ in equation (64).

$$\therefore \quad X(f) = \iint_{-\infty}^{+\infty\,+\infty} y(t)\, C(f_o)\, e^{-j2\pi(f-f_o)t}\, df_o\, dt \qquad \ldots\ldots\ldots \qquad (65)$$

Interchanging the integral we have,

$$X(f) = \int_{-\infty}^{+\infty} Y(f-f_o)\, C(f_o)\, df_o \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (66)$$

This relation in which Y(f) and C(f) are interchangeable, is commonly expressed symbolically as,

$$X(f) = Y(f) * C(f) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (67)$$

The implied operation (*) on Y(f) and C(f) is called a convolution.

Similarly it can be proved also that if

$$X(f) = Y(f) \,.\, C(f) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (68)$$

Then,

$$x(t) = \int_{-\infty}^{+\infty} y(t-\tau) \; c(\tau) \; d\tau \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (69)$$

$$x(t) = y(t) * c(t) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (70)$$

Thus, if signals are multiplied in time domain, their respective spectra are convolved with one another in frequency domain. Similarly, multiplication of frequency spectra implies convolution of the time signals.

Now let us use this property of fourier transform to explain our problem. We are now provided with a finite length of data x(t). This data can be considered as the product of the actual infinite length of data y(t) with a rectangular window function c(t) of finite length T. See Figure (20). Hence according to the property of fourier transform the computed spectrum X(f) will be the convolution of the actual spectrum Y(f) with the spectrum of the rectangular window function C(f).

Since

$$c(t) = \begin{cases} 1 & |\tau| < T \\ 0 & |\tau| > T \end{cases} \quad \ldots\ldots\ldots \quad (71)$$

Where T is the length of data.

C(f) can be readily written as,

$$C(f) = \frac{\sin \pi f T}{\pi f} \quad \ldots\ldots\ldots \quad (72)$$

where 1st zero crossing is at $f = \frac{1}{T}$, the bandwidth of the spectrum of rectangular window C(f).

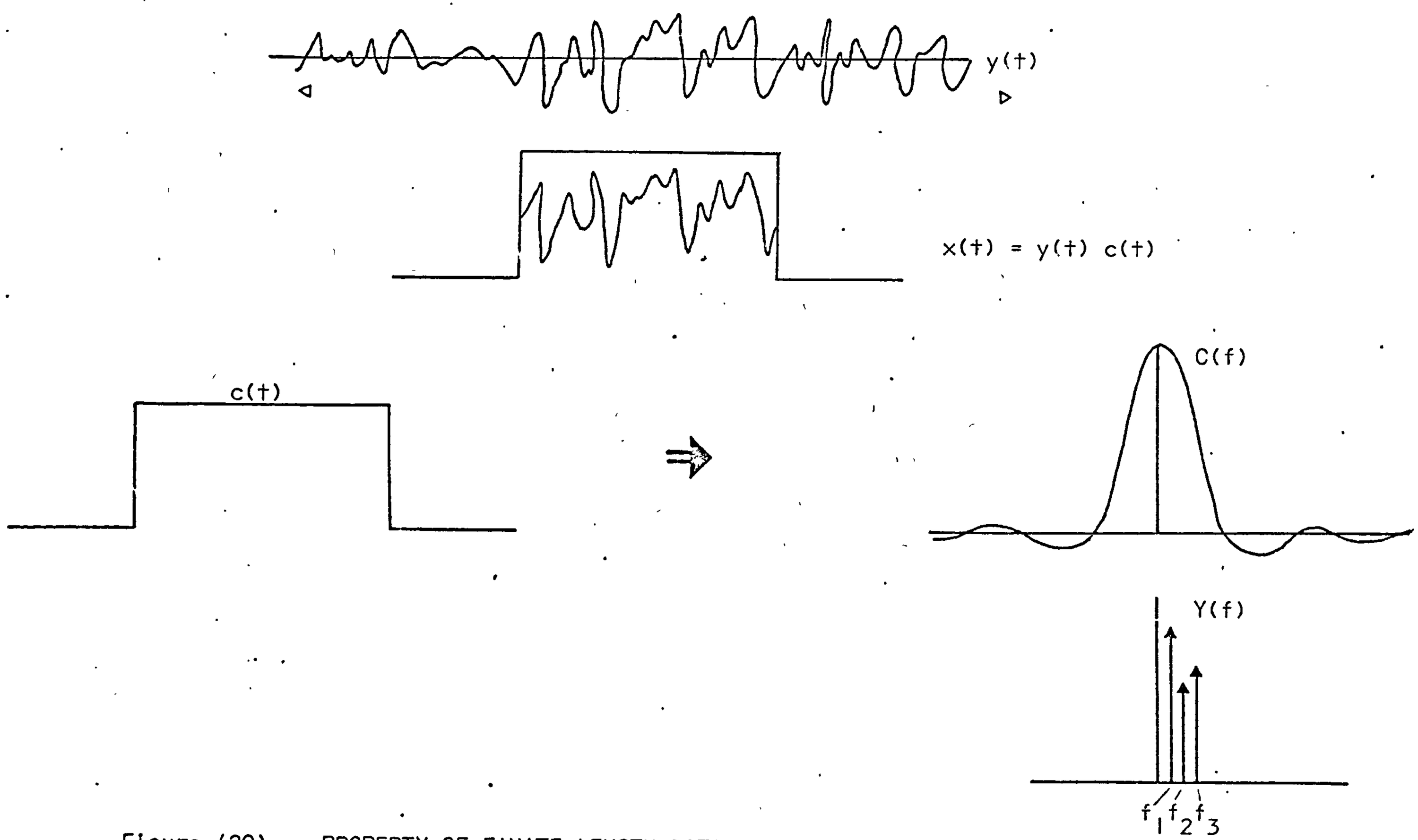

Figure (20) PROPERTY OF FINITE LENGTH DATA.

Due to this convolution effect of the actual spectrum with the spectrum of rectangular window, the spectrum samples in Y(f), whose frequency spacing is smaller than the bandwidth of C(f), will not be resolved. This can be greatly improved by choosing a longer length of data T, since the bandwidth of the spectrum C(f) is inversely proportional to the data length. But still there is some disadvantage of the rectangular window. The spectrum of it has relatively large sidelobes. Though the resolution of the spectrum may be finer, the details of the true spectrum of Y(f) will not be restored again due to the effect of convolution of the spectrum Y(f), with the spectrum having high sidelobes. However, in analysing the signals, it is necessary that the details of the true spectrum of the signal should be brought back,somehow, otherwise mis-interpretation is likely to be made on the signal under analysis. Therefore, the estimation of power spectrum becomes a very important task.

In our case we are engaged with making estimations of power spectrum of random signals such as distortion noise signals and the random Gaussian input signals. From the definition, it is understood that the power spectrum of a random signal is the average of the power per unit frequency band. Clearly from the facts we have just described, obscurring of the spectrum is expected, having some errors in the estimated mean power (variance) of the signal and therefore it is necessary to take an average of the variance over a number of spectral estimates, to reduce the error.

There are many ways of taking the average of the variance over a large number of spectral estimates. The averaging of the spectral estimates may be done either in time or in frequency domain. One of these methods will be adopted in our analysis. A combination of time and frequency averaging method was used for our analysis. This was particularly achieved through the use of lag window.

3.8 METHOD OF ESTIMATION OF POWER SPECTRUM

For the estimation of power spectrum of the signals X(t) and $n_q(t)$ defined in Section (2.14) the following method is adopted and all along the process, F.F.T. algorithm will be used.

Firstly, the F.F.T. of the signal sample was taken and the power spectrum was calculated. This power spectrum differs from actual spectrum and therefore the need for the averaging of the variance over a number of spectral estimates were realised. This had been achieved by making use of the convolution property of the fourier transform. The autocorrelation function for the same data length was calculated via the inverse F.F.T. of the power spectrum we had manipulated. Then the autocorrelation function was multiplied by a time window function, specially chosen for the purpose. There are several types of lag windows to choose. Two types of lag windows[19] are shown in Figure (21) together with their spectral window functions. The nature of the spectral window in these two pairs is the same, having sidelobes 1% to 2% of the height of the main lobe. The major differences are that the highest height of sidelobe of the "HAMMING" spectral window is 1/3 of the highest height of side lobe of "HANNING" window in frequency, and that the sidelobes of "HANNING" window fall off more rapidly than those for the "HAMMING" window. With an idea to reduce the effect of obscuring the spectral estimates by the sidelobes of the spectral window, a HAMMING LAG WINDOW having the same length as the data was selected for our case. The equation for this type of lag window is,

$$\begin{aligned} \omega(\tau) &= 0.54 + 0.46 \cos\left(\frac{\pi\tau}{T}\right) \qquad |\tau| < T \\ &= 0 \qquad |\tau| > T \end{aligned} \qquad \ldots\ldots\ldots\ldots \quad (73)$$

where T is the length of data samples.

It has the spectrum of

$$W(f) = 0.54\ W_o(f) + 0.23\ W_o\left(f + \frac{1}{2T}\right) + W_o\left(f - \frac{1}{2T}\right) \qquad \ldots\ldots\ldots\ldots \quad (74)$$

The multiplied auto-correlation function was then fourier transformed by using F.F.T. and the resulting output gave us the finally averaged power spectrum estimates. Averaging or the smoothing action was achieved by replacing each spectrum estimate by a linear sum of the estimate and the two adjacent estimates, with weight 0.23, 0.54, 0.23 of the HAMMING spectral window.

3.9 INPUT AND NOISE SPECTRA OF C.F.D.M.

Using the estimation of power spectrum technique just described, we have analysed the input and noise power spectrum of the C.F.D.M. and these are illustrated in Figure (22) and Figure (23). Noise signal is defined in equation (18) and the input signal power and noise power spectra are calculated up to 10 kHz, since the actual highest frequency of interest is 3.1 kHz. Noise in C.F.D.M. is flat inside the input frequency band and since the noise is measured after the final filtering in the decoder, the noise spectrum is tapered off after the message band. All the spectra apply for the input power level of 3 dBm.

3.10 SIMULATIONS OF JAYANT'S A.D.M.

One bit memory adaptive delta modulator invented by JAYANT in March 1970, was also simulated. In this type of coder, the adaptation of step size is based on the comparison between the two latest binary levels L_r and L_{r-1}. For the simulation of this coder, adaptation algorithm described in section (2.8.2) has been used. The simulation is a special type of C.F.D.M. coder, but with 'A' parameters, different from that of C.F.D.M. The A parameters used are $A_1 = A_2 = -\frac{1}{1.5}$ and $A_3 = A_4 = 1.5$ A minimum step size of ± 0.03 and the maximum step size of ± 5 are used.

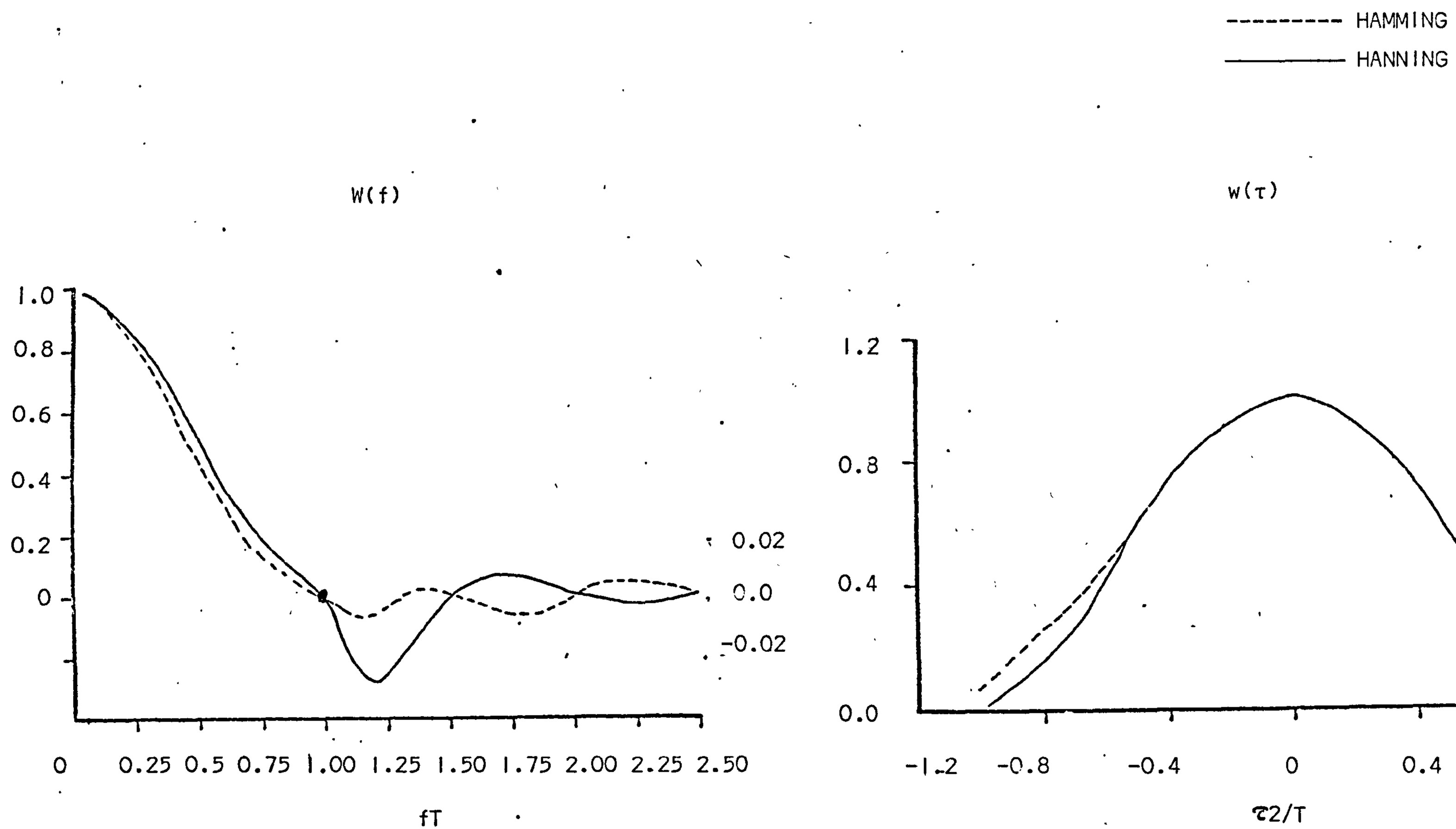

Figure (21)a SPECTRAL WINDOW

Figure (21)b LAG WINDOW

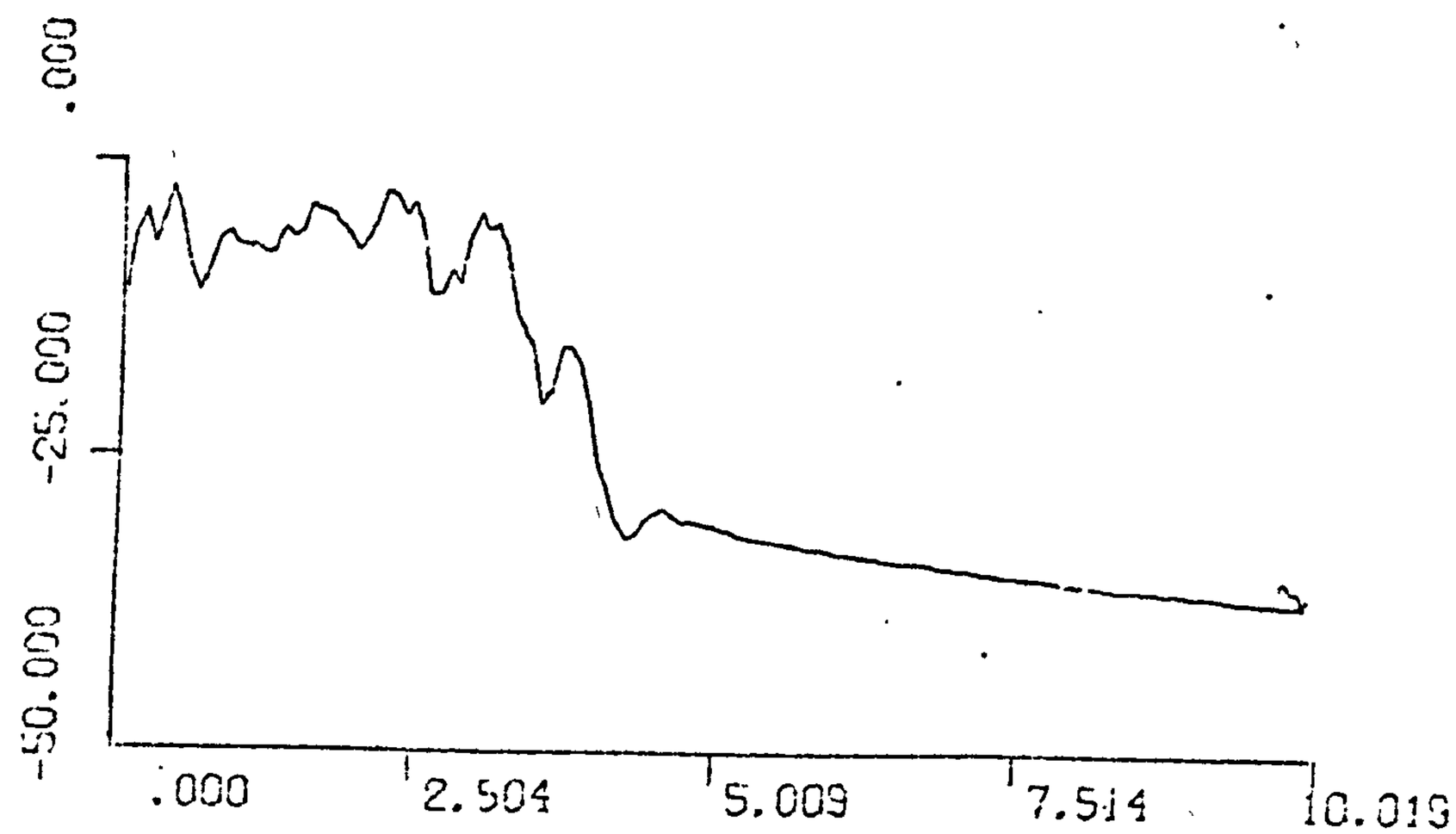

Figure (22) POWER SPECTRUM OF BANDLIMITED GAUSSIAN INPUT.

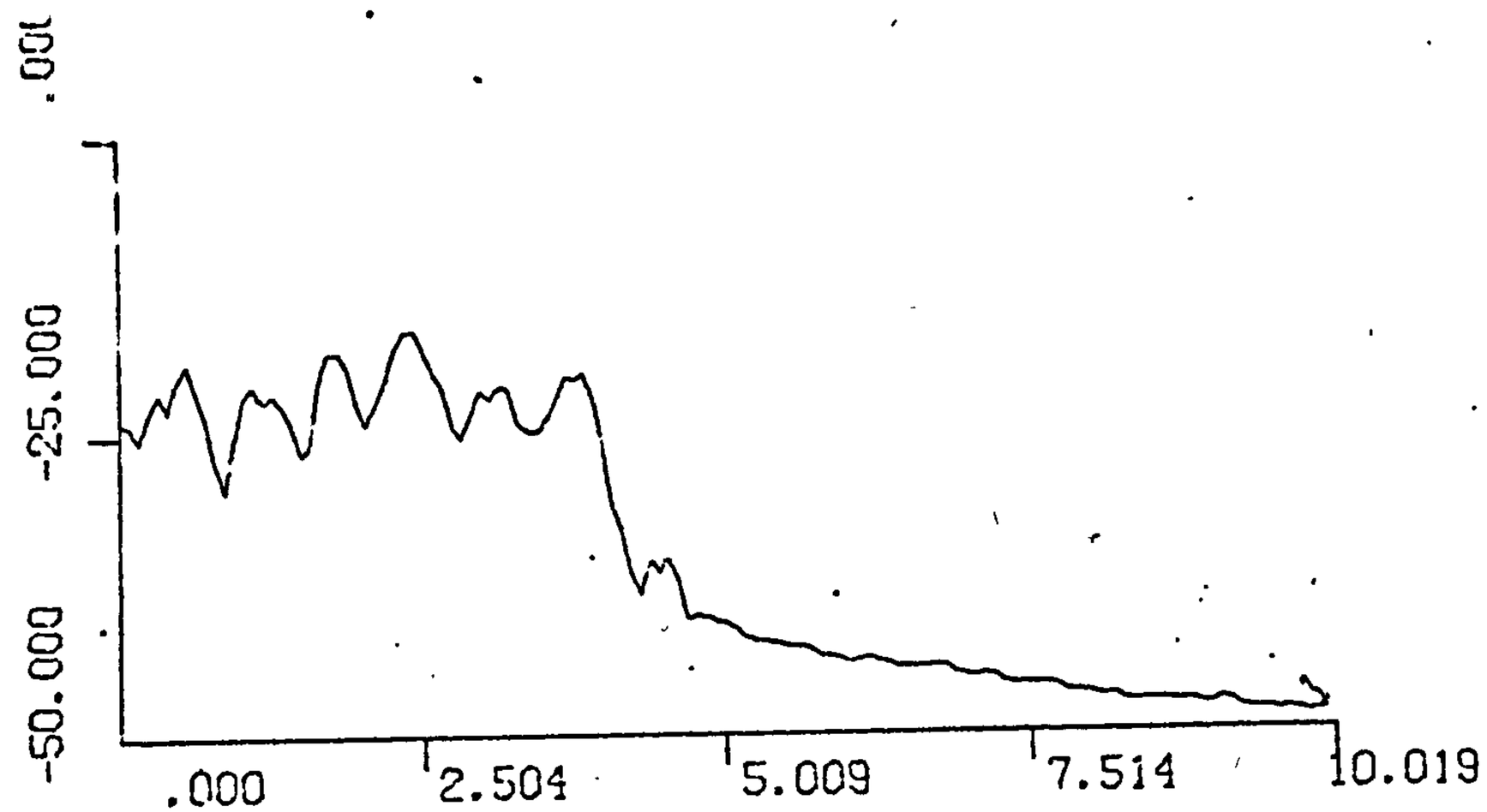

Figure (23) NOISE POWER SPECTRUM OF C.F.D.M.

3.11 STEP RESPONSE OF JAYANT'S A.D.M.

A step input was applied to the coder. The input was 0.5 volt for the first three clock periods, and stepped up to 39.5 volts, as used in Winkler's H.I.D.M. coder to be compared. The graph of the step response of JAYANT's A.D.M. is shown in Figure (24). Minimum step size of 1.0 unit is used. It takes about 8 clock periods to cope with the step input. It has a relatively faster response than L.D.M. However, the step response shows that while tracking the constant level of the step input, (hunting periods), the step size does not always assume the smallest possible value. This is an inherent feature of this type of coder. The stability of the coder seems to be very poor.

3.12 RESPONSE OF J.A.D.M. TO BAND-LIMITED GAUSSIAN INPUT

The input is a Gaussian signal band-limited to 3.1 kHz at the sampling rate of 40 kHz. Input range of 75 dBm is applied to the coder and the graph of decoded signal to noise against input signal power is plotted as shown in Figure (25). Maximum signal to noise ratio of 21 dBs is achieved having the dynamic range of 50 dBm at 16 dB SNR. Graph for input signal x(t) and integrated output signal y(t) is illustrated in Figure (26), at the input power level of 3 dBm.

3.13 IMPULSE RESPONSE OF J.A.D.M.

Impulse of 2.5V having the width of eight clock periods is applied to the input to the J.A.D.M. Figure (27) shows the computer generated waveforms of the impulse input and the reconstructed signals. The representation of an impulse is inferior to our C.F.D.M. system. It took almost the same amount of delays to represent the impulse, but the hunting characteristic of J.A.D.M., after the impulse, is very poor. Due to this fact, the stability of the J.A.D.M. system seems to be poor compared to our C.F.D.M. It is this inability of J.A.D.M.[13] to follow sudden changes of input signal level, and would introduce the 'twinkling' effect when coding television pictures.

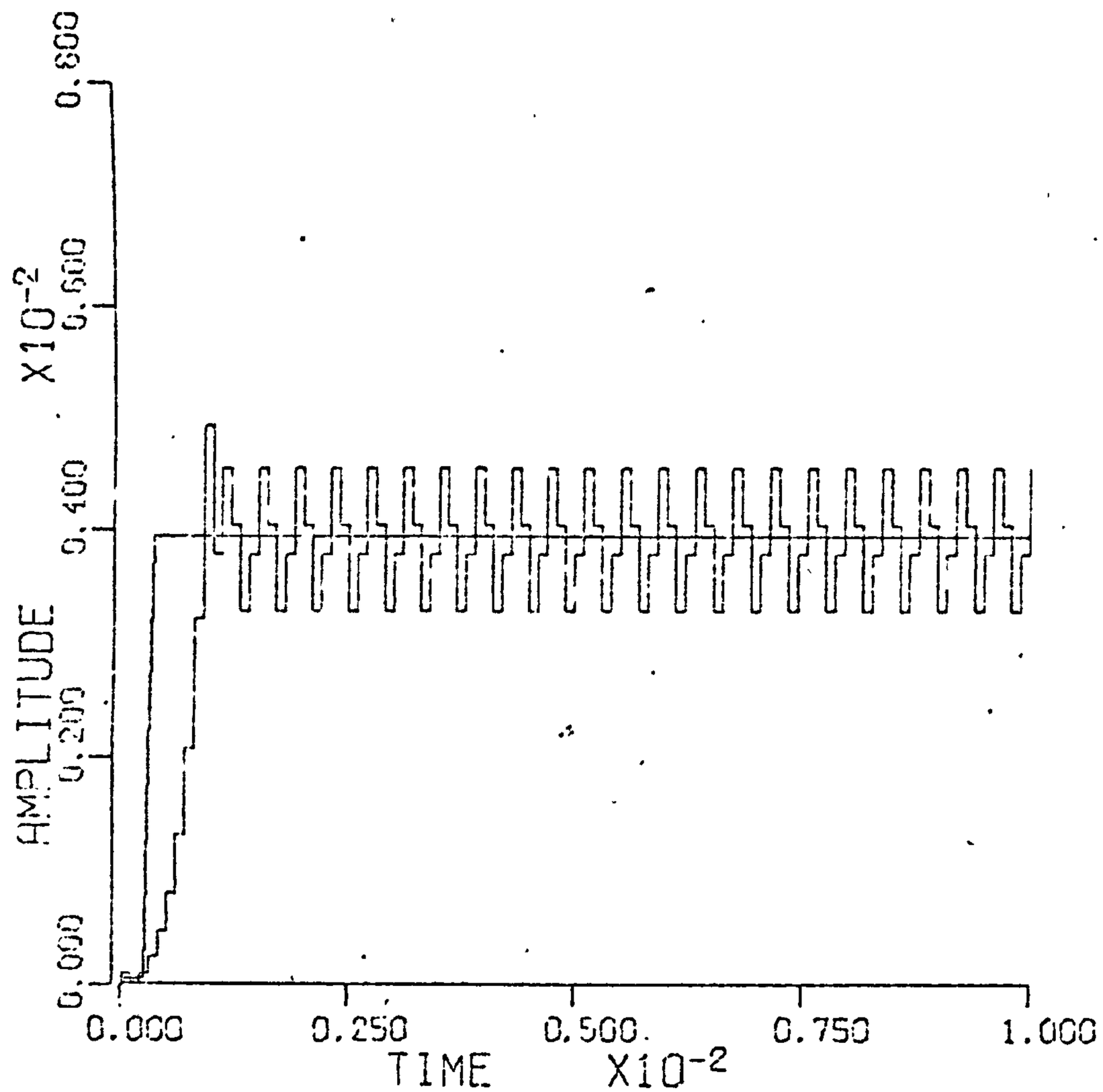

Figure (24) STEP RESPONSE OF J.A.D.M.

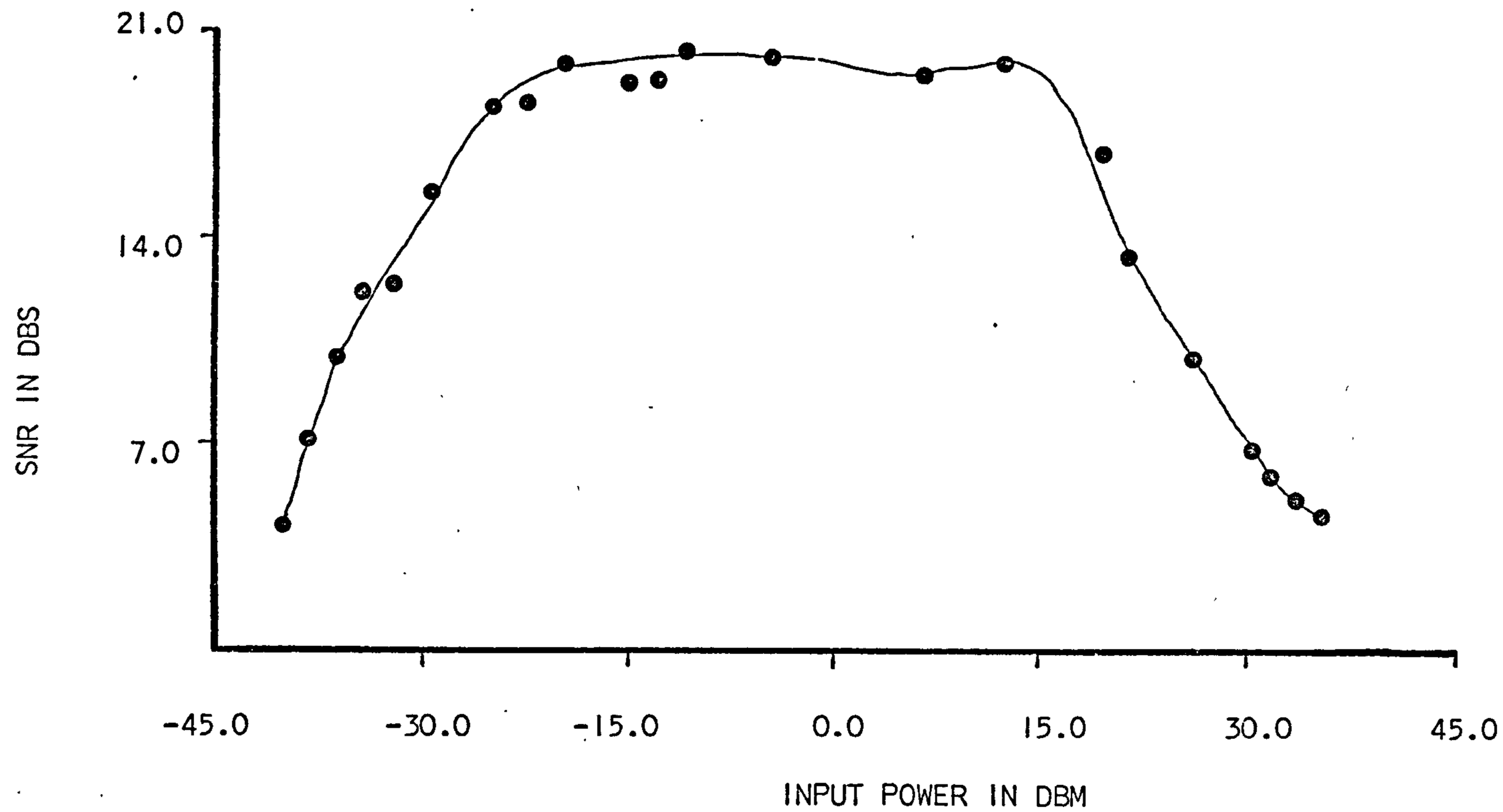

Figure (25) DYNAMIC RANGE OF J.A.D.M.

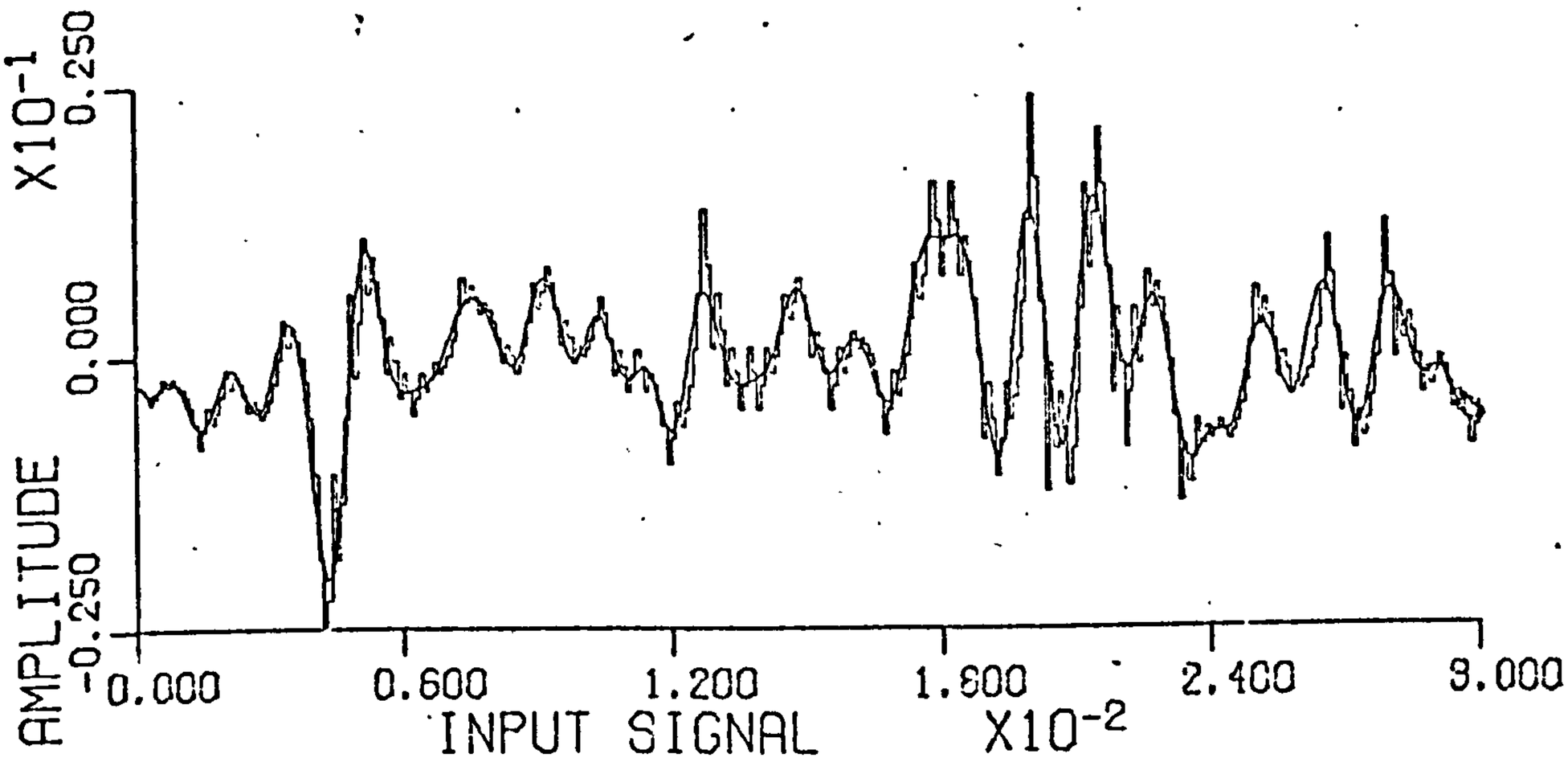

Figure (26) COMPUTER GENERATED WAVEFORMS OF INPUT AND RECONSTRUCTED SIGNALS.

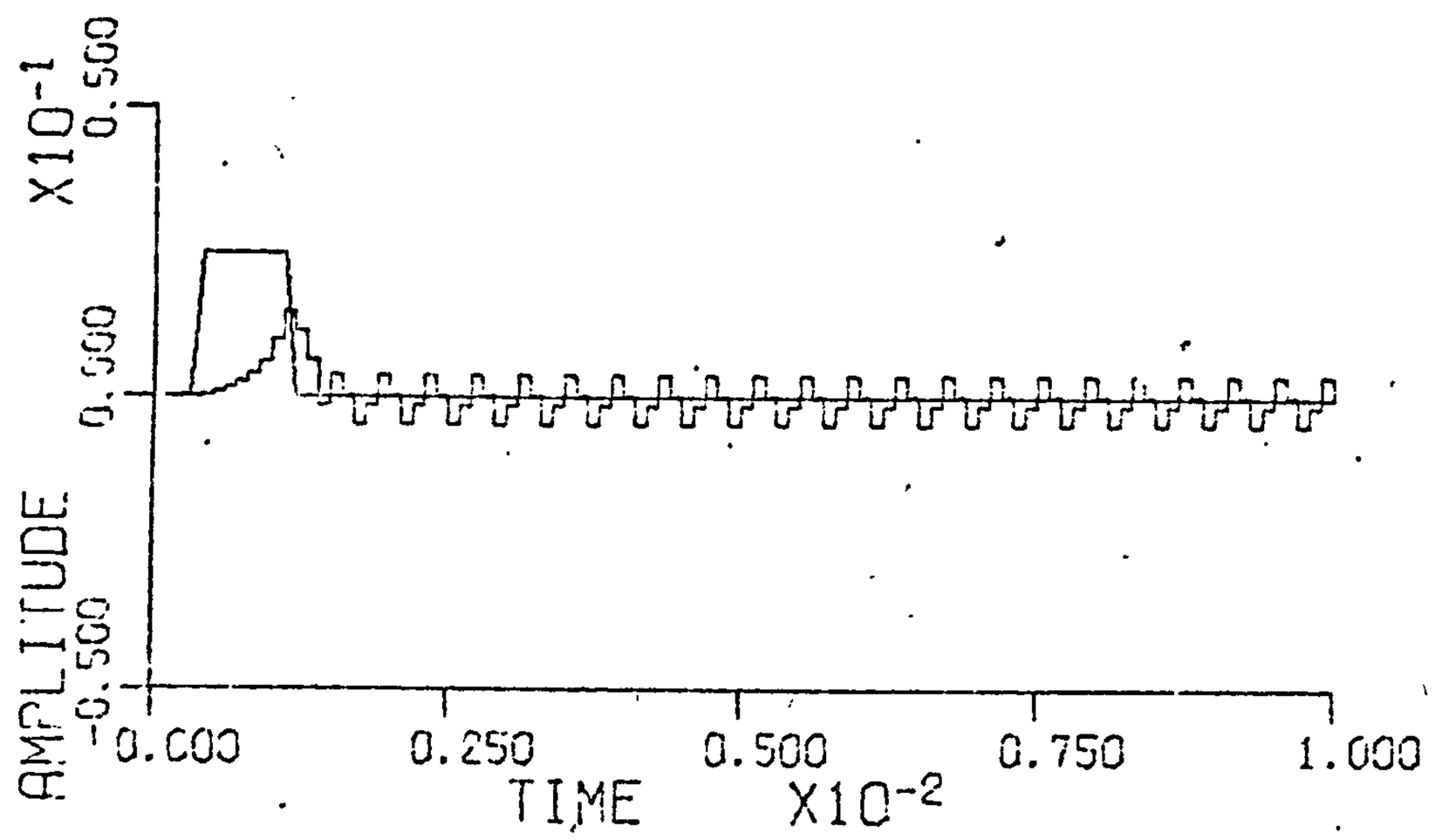

Figure (27) IMPULSE RESPONSE OF J.A.D.M.

3.14 NOISE SPECTRUM OF J.A.D.M.

Figure (28) displays the noise power spectrum of the J.A.D.M. system. Input signal power spectrum is shown in Figure (22). Noise reduction of J.A.D.M. can be realised when compared to LDM of Figure (36). The noise power is nearly proportional to the input power, unlike linear d.m., and results in the large dynamic range displayed in figure 25.

3.15 STEP SIZE DISTRIBUTION OF J.A.D.M.

The distribution of step sizes utilised in the simulation of J.A.D.M. is shown in figure (29). It is symmetric about the mean of zero, and has the similar statistical property as the input signal. It offers a wide range of step size excursion though only one bit memory length is used for the adaptation control.

3.16 SIMULATION OF WINKLER'S H.I.D.M.

High information delta modulator[8] proposed by Winkler, M.K., in 1963 is also simulated. It requires three binary levels to control impulse steering circuit of H.I.D.M. Due to the generality of principles of C.F.D.M. coder, H.I.D.M. has been simulated employing algorithm developed for our C.F.D.M. coder. For this simulation a new set of A parameters are used with the magnitudes of $A_1 = A_2 = -0.5$, $A_3 = 1.0$ and $A_4 = 2.0$. The minimum and the maximum step sizes are limited to ± 0.03 and ± 5.0 volts respectively. Throughout the simulation ideal integration has been used.

3.17 STEP RESPONSE OF H.I.D.M.

The same value of step is applied at the input to the coder. Step height of 39.5 volts has been followed after the constant input of 0.5 volt is applied for three clock instants. The minimum step size of 1.0 unit volt is used and the graph of the response of H.I.D.M. to this input is shown in Figure (30). It increases the magnitude of step size exponentially and it takes 6 clock periods to catch the input, but it overshoots quite high and the step size oscillates around the constant

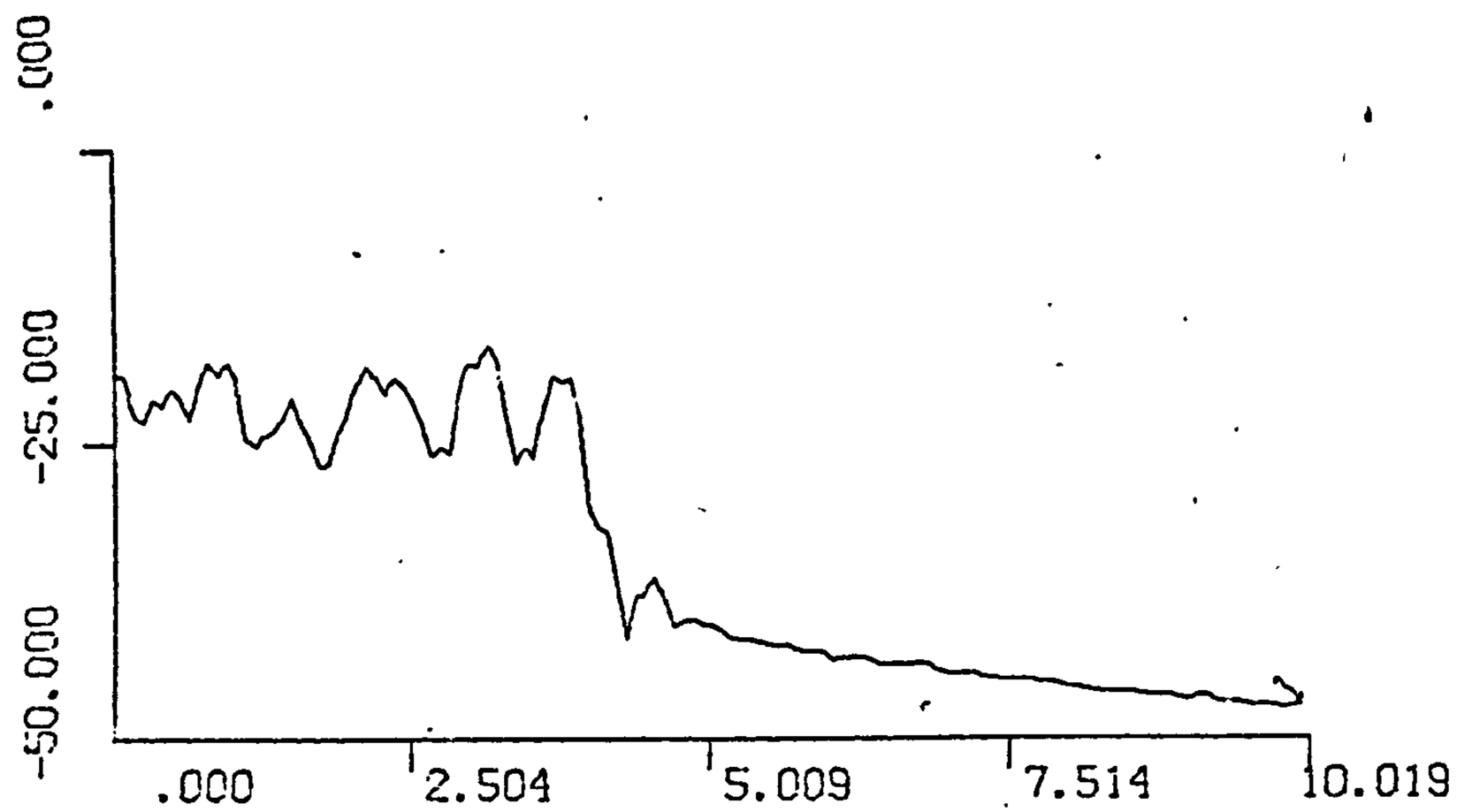

Figure (28) NOISE POWER SPECTRUM OF J.A.D.M.

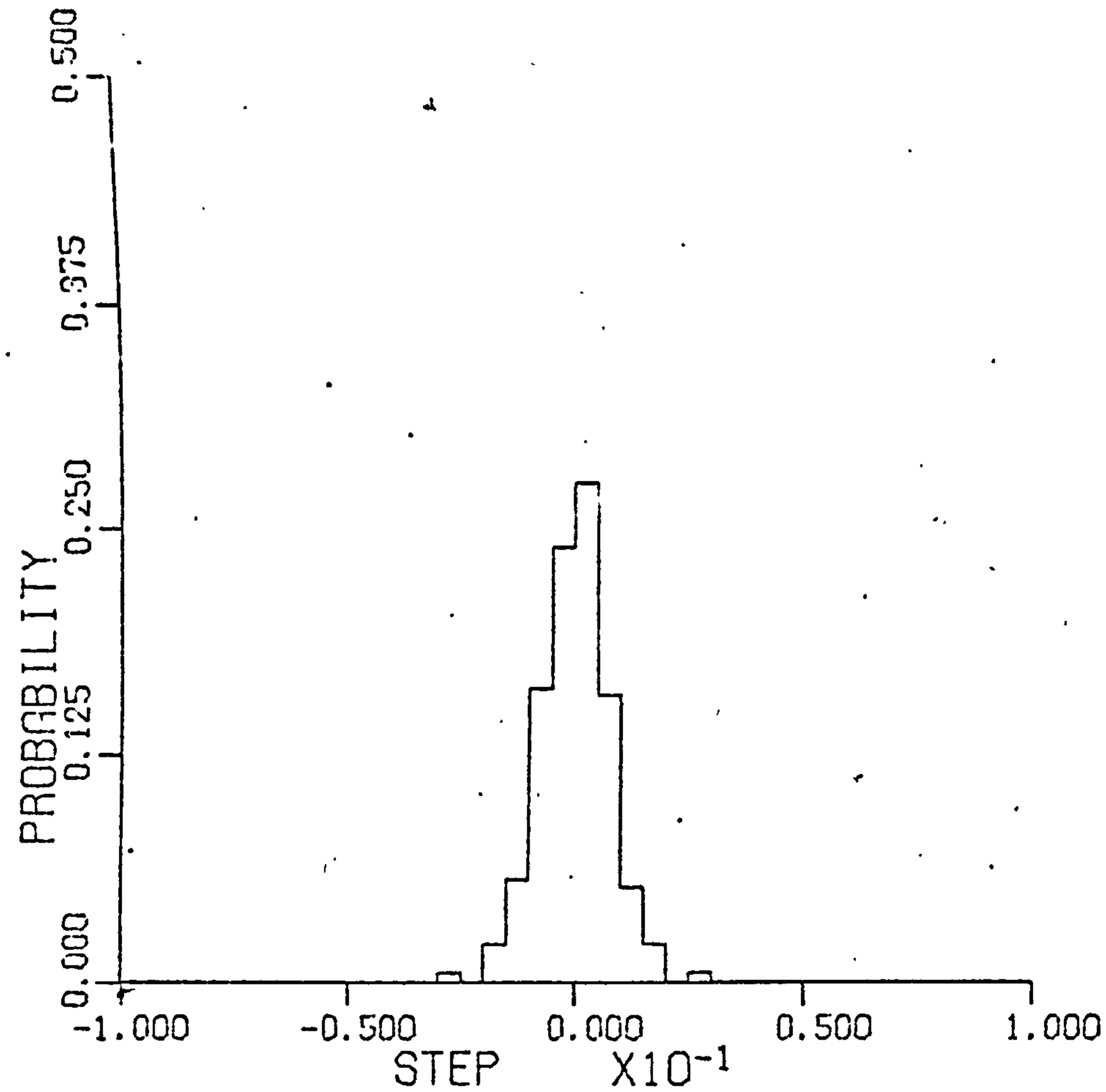

Figure (29) STEP-SIZE DISTRIBUTION OF J.A.D.M.

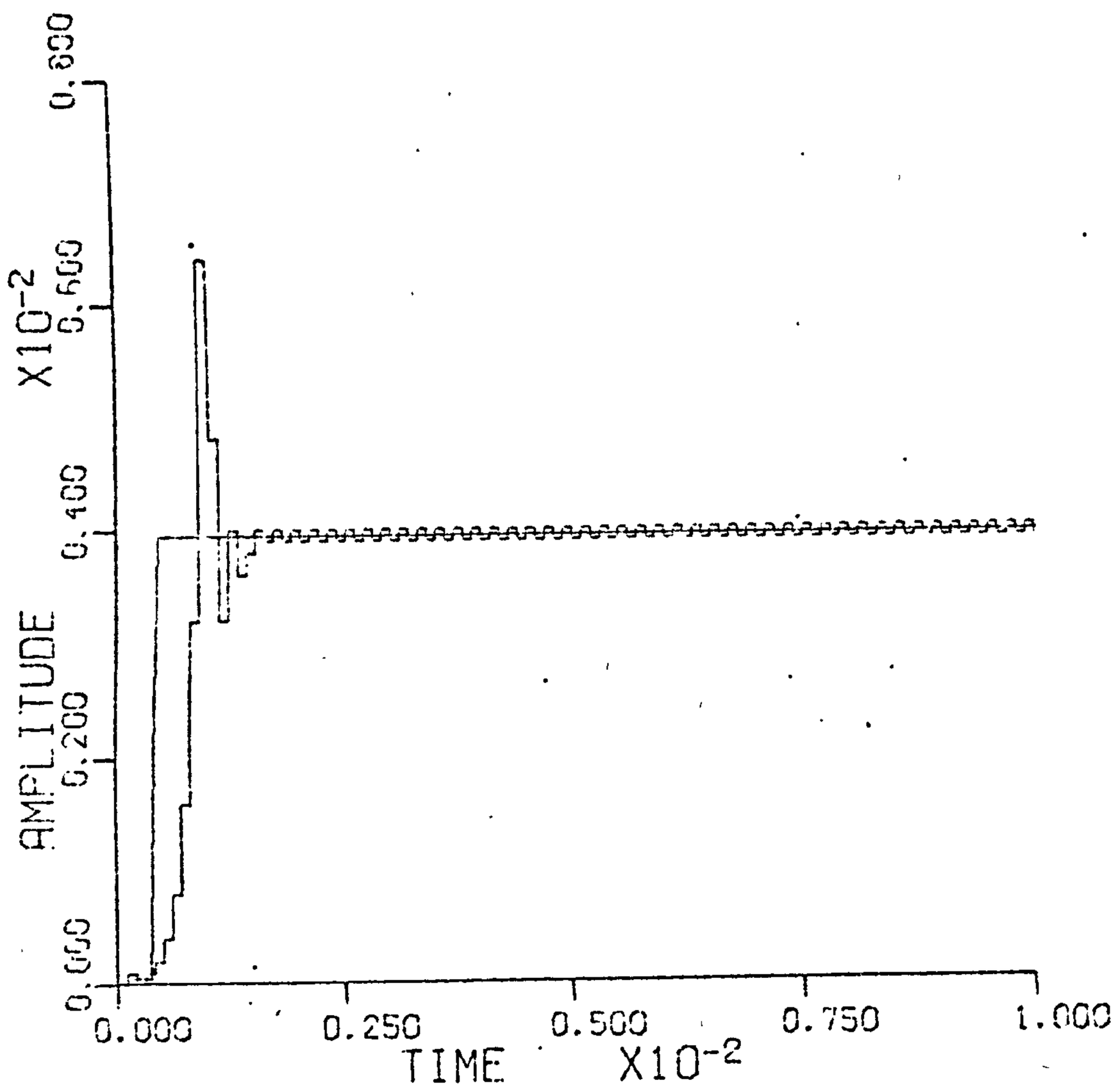

Figure (30) STEP RESPONSE OF H.I.D.M.

input level of the step voltage. It finally hunts the input with a basis 1 0 1 0 1 0 pattern. Its response is quite fast compared to LDM or even faster than JAYANT's ADM.

3.18 IMPULSE RESPONSE OF H.I.D.M.

Figure (31) represents the impulse response of H.I.D.M. Input is a pulse of amplitude 2.5V and width 8 clock periods. It takes about (3) clock periods to track the input pulse. The representation of the impulse is quite good with a bit of overshoot. It has a large undershoot after the impulse, but it later oscillates with a 1 0 1 0 1 0 pattern and becomes stable. The stability is better than J.A.D.M.

3.19 SIMULATION OF L.D.M.

To compare the performance of linear delta modulator with other adaptive delta modulators, a basic delta modulator, with a fixed step size whose sign depended on the sign of the last binary level transmitted, was simulated. Fortunately, the simulation of the linear delta modulator becomes easier, since the simulation of linear delta modulator is the special case of the C.F.D.M. coder and the programme written for the simulating C.F.D.M. has been used with the A parameters having the magnitude of $A_1 = A_2 = -1.0$ and $A_3 = A_4 = 1.0$. The simulation used an ideal integrator in the feed back loop.

The decoding was done in the encoding side of the system, by passing the output of the integrator in the feed back loop through a low pass filter, whose cut-off frequency is the highest frequency component in the message band.

The flow chart of the main programme for the simulation of linear delta modulator is as shown in Figure (39)a. A minimum step size of ± 0.03 volt was used in the simulation.

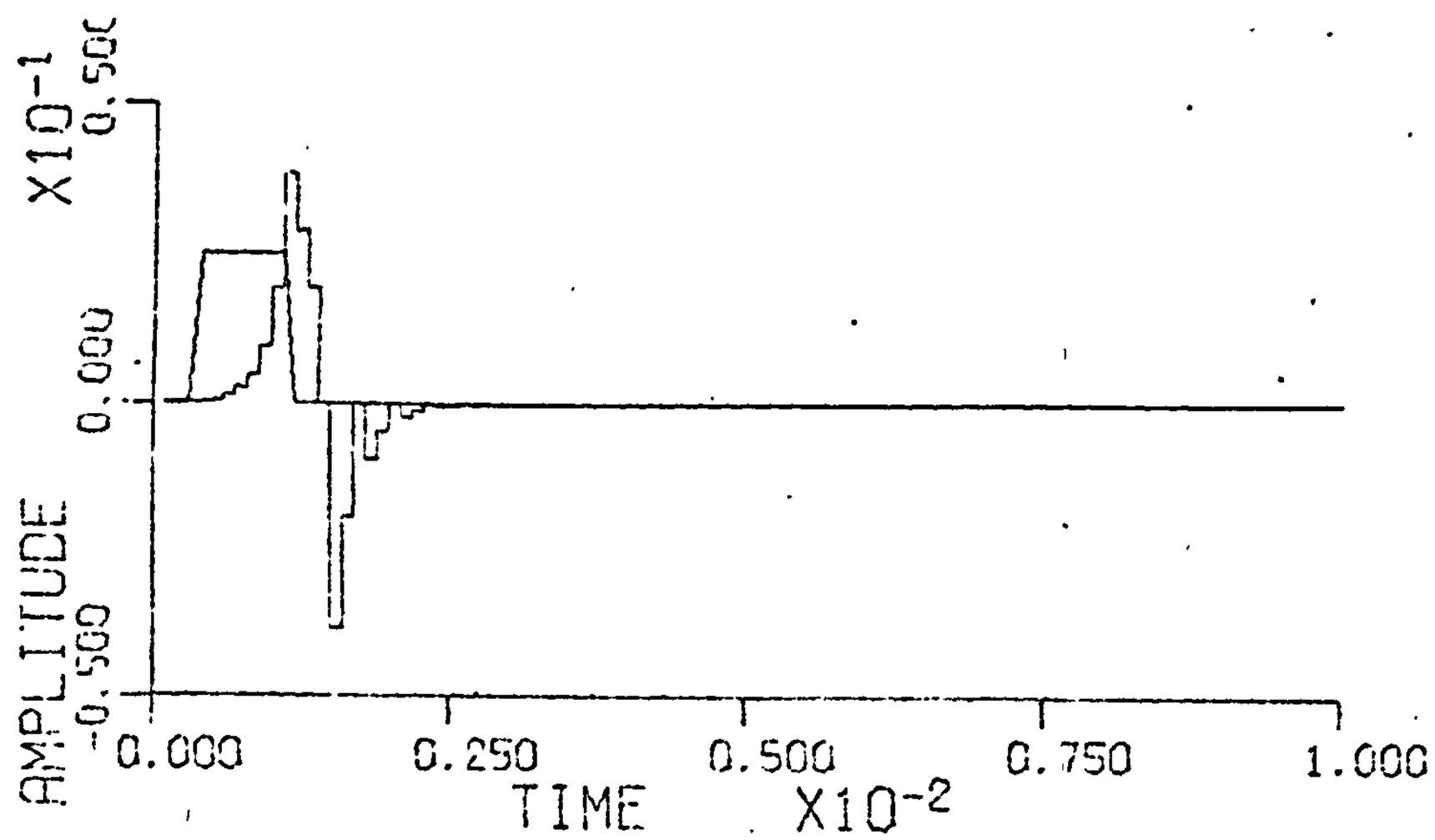

Figure (31) IMPULSE RESPONSE OF H.I.D.M.

3.20 STEP RESPONSE OF LINEAR DELTA MODULATOR

The step response of the L.D.M. has been studied. The input is 0.5 volt for the first 3 sampling periods, and it then steps up to 39.5 volts. This value of the step is selected to compare with the response of Winkler's H.I.D.M. which uses the same value of the step function. The algorithm for generating the step function is shown in Figure (32). The computer drawn graph for the step response of 1.0 unit was chosen for the simulation.

Linear delta modulator increases or decreases its step size by a unit factor to the initial step size, at the reception of one or zero level binary output. It takes L.D.M. 40 clock periods to catch up the step input, verifying the slow response of L.D.M. to rapid change of input level. The hunting characteristic of L.D.M. is a 1 0 1 0 pattern, with a minimum step size of 1.0 unit.

3.21 RESPONSE OF L.D.M. TO BANDLIMITED GAUSSIAN INPUT

Bandlimited Gaussian is applied to the input of linear delta modulator. Input is bandlimited to 3.1 kHz at the sampling frequency of 40 kHz. Minimum step size of 0.03 volt is used for the simulation. The maximum signal to noise achieved by L.D.M. is 21 dBs. It offers a very narrow dynamic range of 7.5 dBm measured at 16 dBs SNR. This is illustrated in Figure (16). Waveforms of x(t) and y(t) signals for linear delta modulator is shown in Figure (33).

3.22 IMPULSE RESPONSE OF L.D.M.

Since only plus or minus one unit of step size can change at every clock instant in linear delta modulator, the representation of an impulse is very very poor. Figure (34) shows the impulse response of L.D.M. The slow response of L.D.M. is verified.

3.23 STEP SIZE DISTRIBUTION OF L.D.M.

Figure (35) displays the step size distribution of L.D.M. Only two levels of steps are being used throughout the simulation. This demonstrates the non-adaptability of L.D.M.

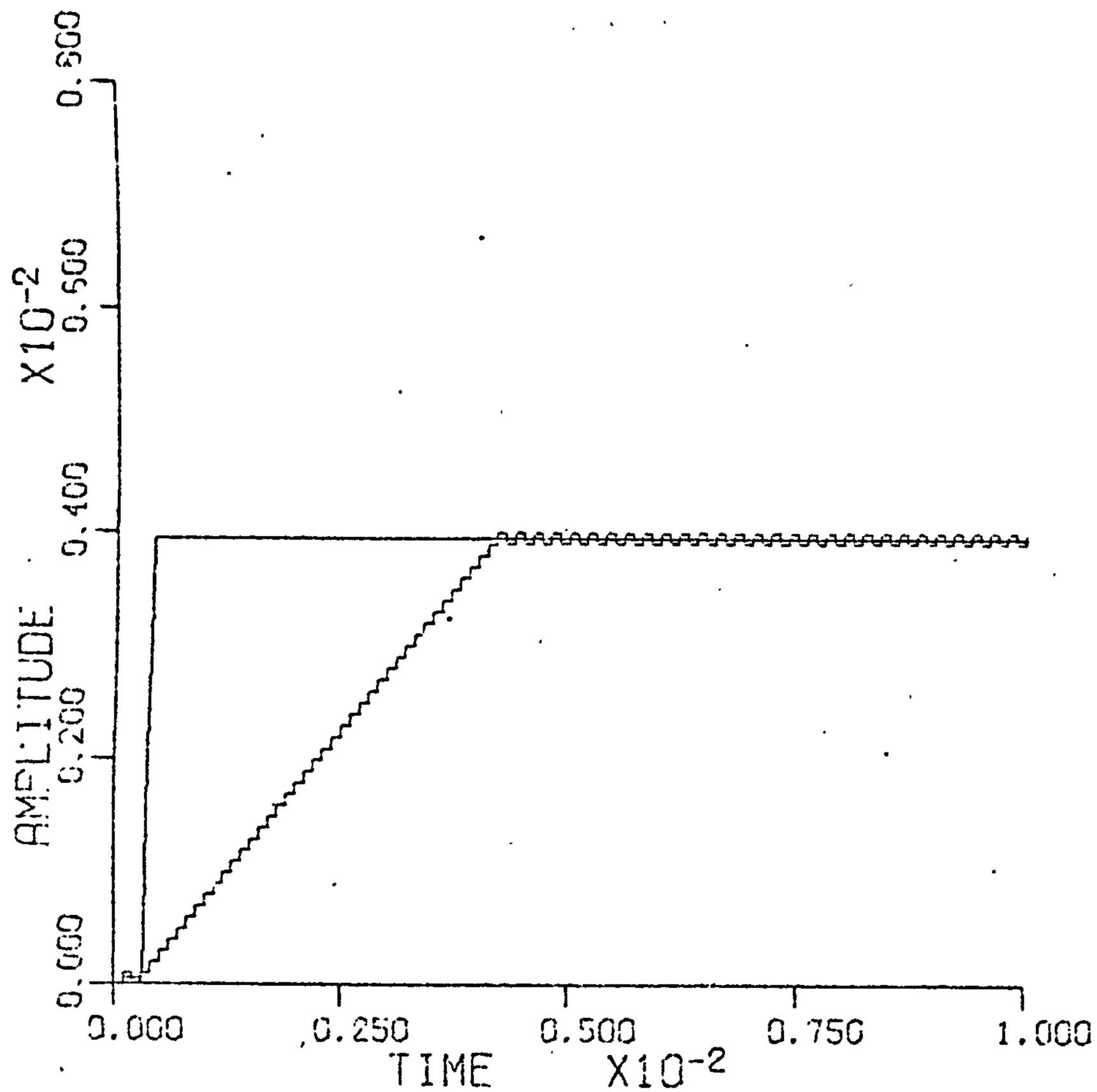

Figure (32) STEP RESPONSE OF L.D.M.

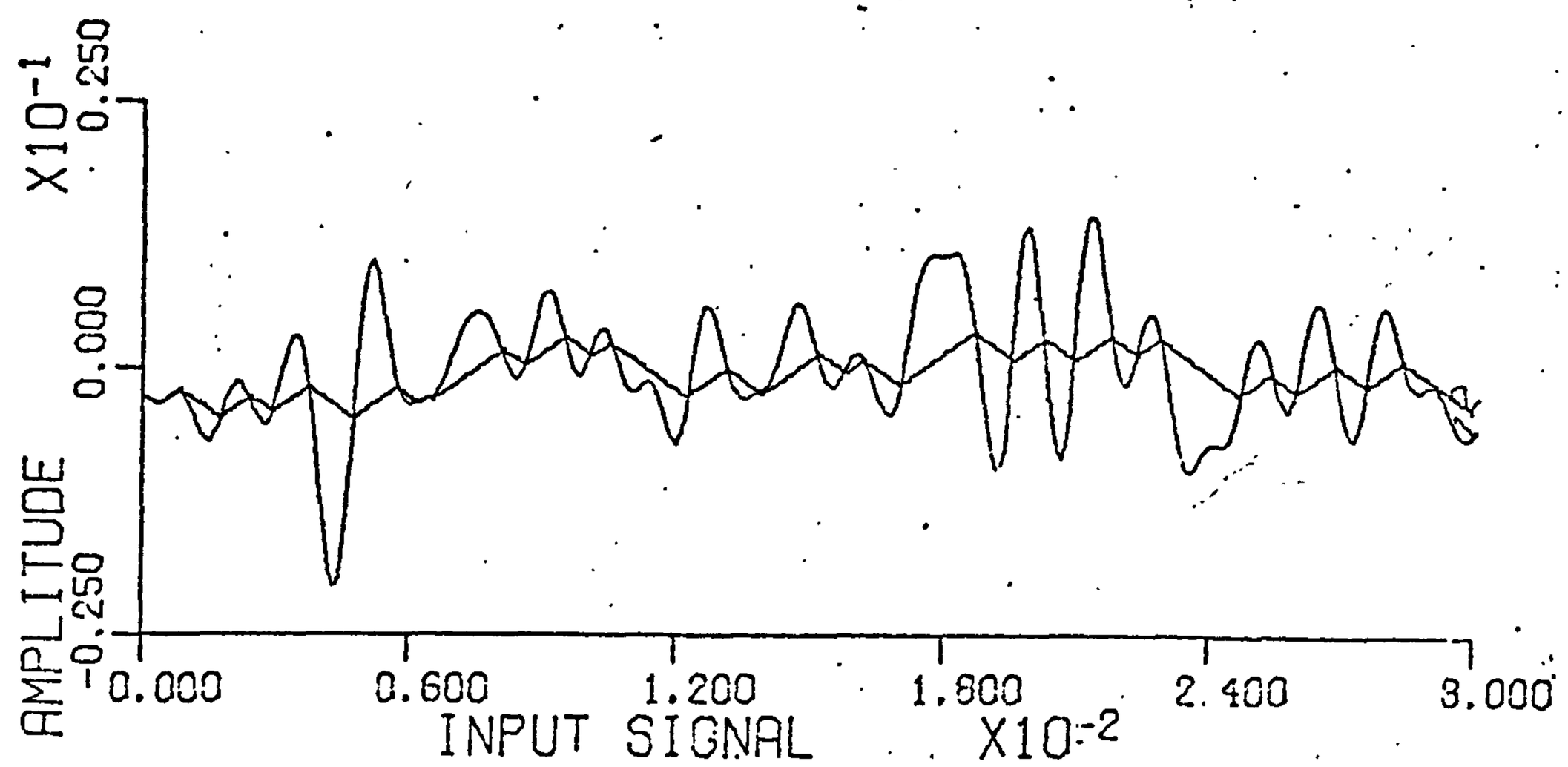

Figure (33) COMPUTER GENERATED WAVEFORMS OF INPUT AND RECONSTRUCTED SIGNAL OF L.D.M.

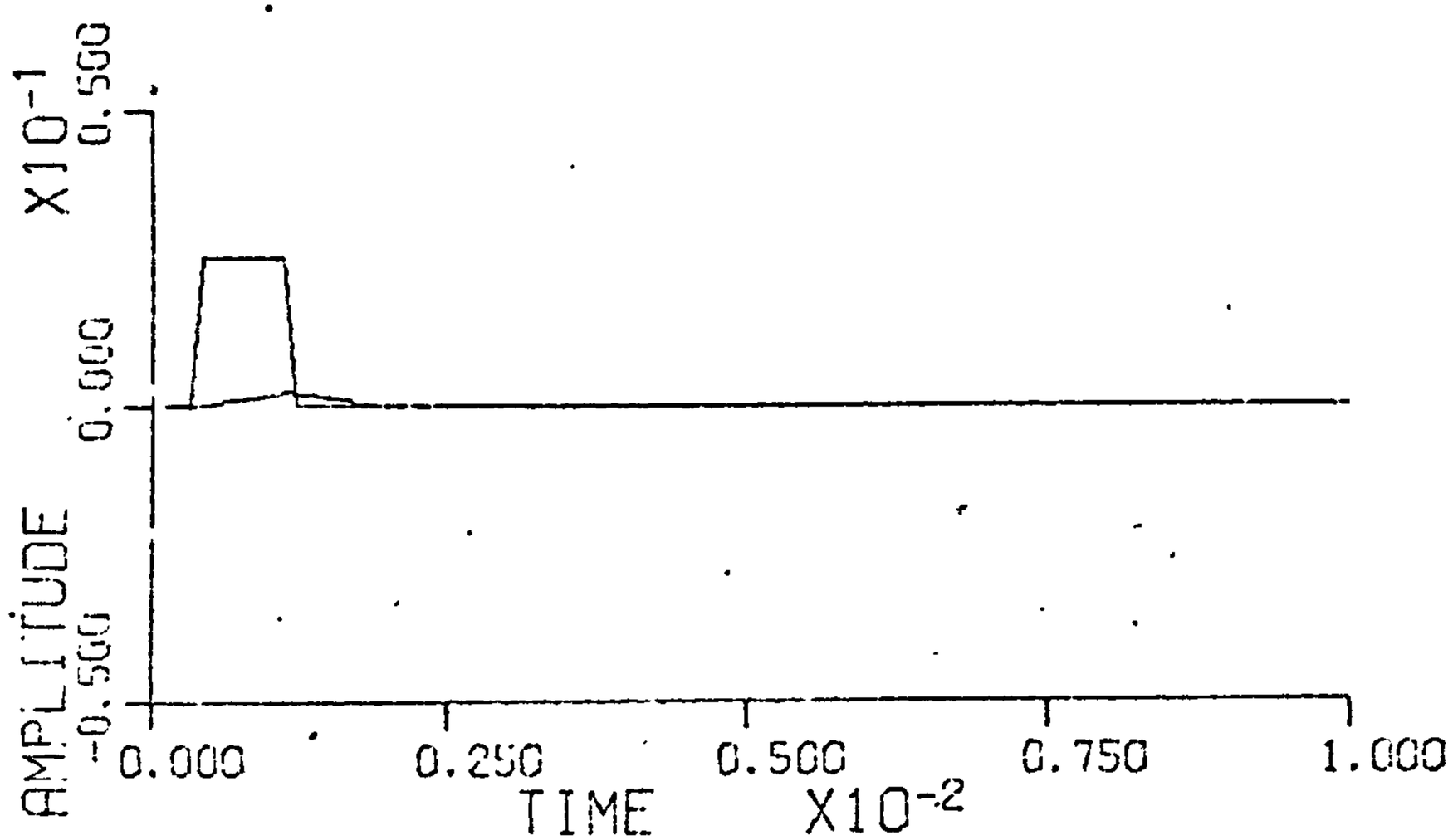

Figure (34) IMPULSE RESPONSE OF L.D.M.

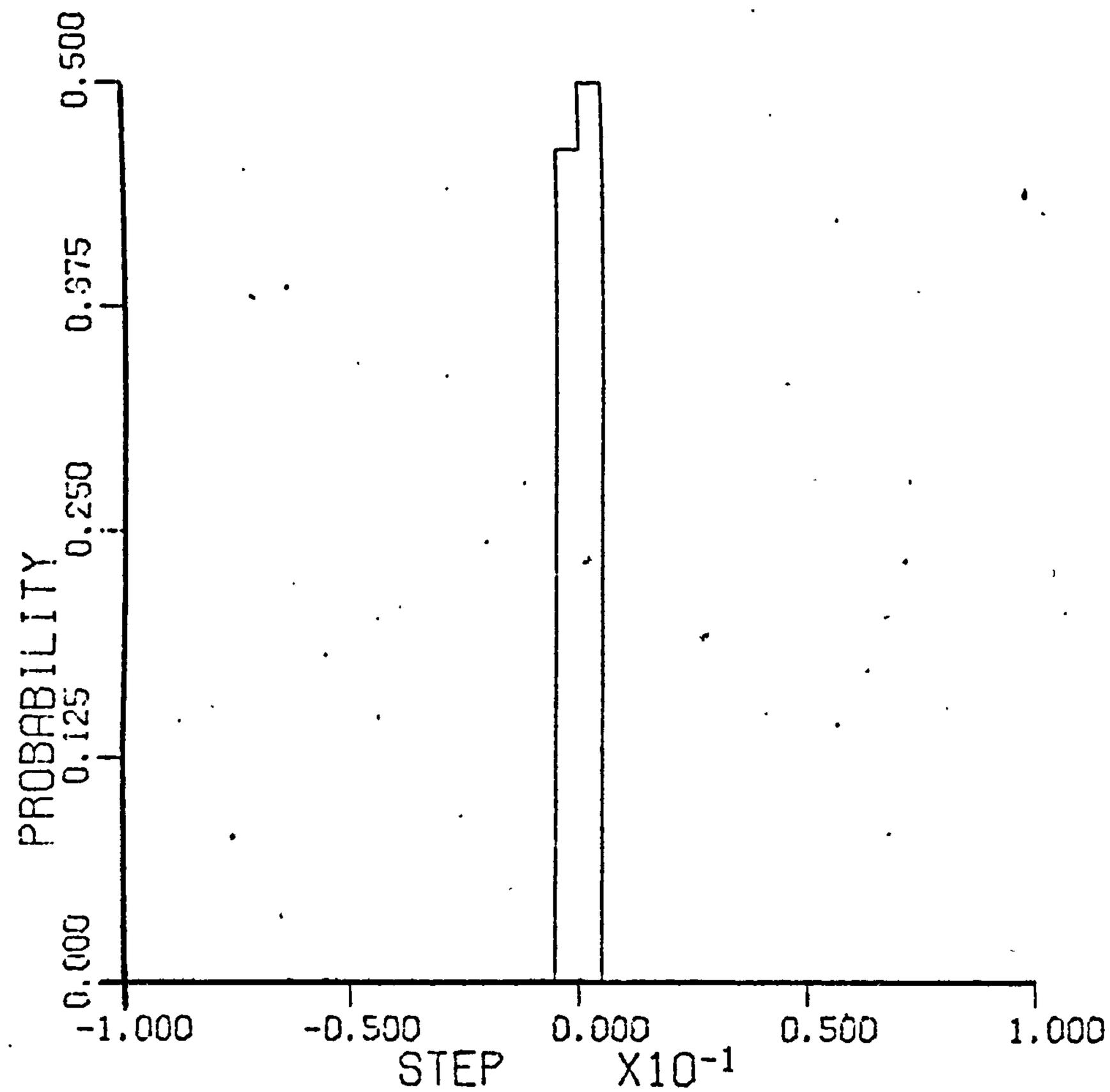

Figure (35) STEP-SIZE DISTRIBUTION OF L.D.M.

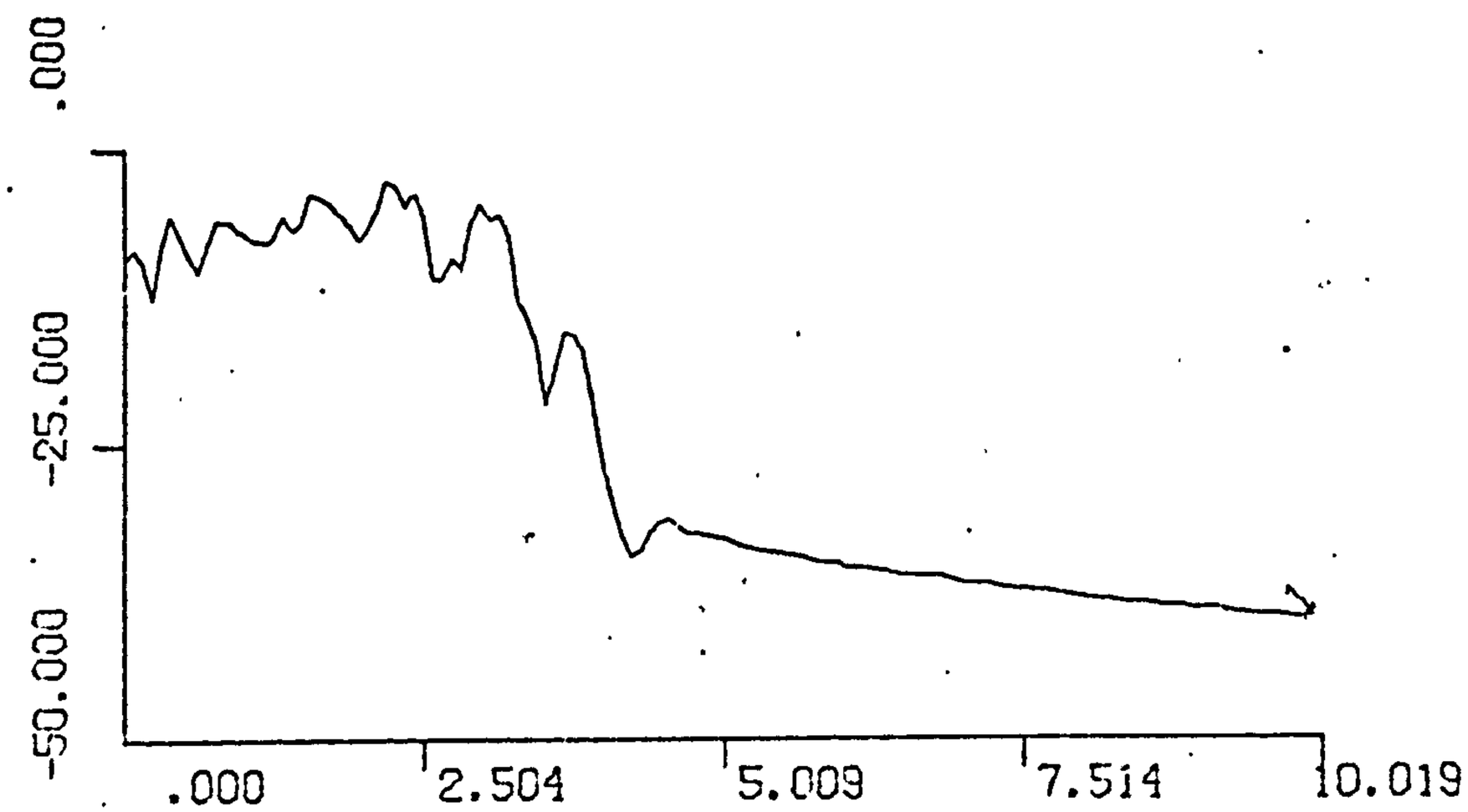

Figure (36) NOISE POWER SPECTRUM OF L.D.M.

3.24 NOISE SPECTRUM OF L.D.M.

Linear delta modulator is grossly overloaded at the input signal level of 3 dBm. Simulation of L.D.M. uses step size of 0.03 volt. Noise power in L.D.M. is as high as input signal power. This explains why we have very low signal to noise ratio at this input level in Figure (16). Noise power spectrum of L.D.M. system is plotted in Figure (36). It is plotted up to 10 kHz, since the highest frequency of interest is 3.1 kHz for a bandlimited gaussian input signal.

CHAPTER 4

DISCUSSION

A new type of instantaneously adaptive delta modulator, named "Constant Factor Delta Modulator", abbreviated C.F.D.M., has been presented. The principles of C.F.D.M. characterised by the use of a memory length of two and its special kind of adaptation logic, which forms the basis of the development of the C.F.D.M. system, has been described. C.F.D.M. coder offers finer interpolation for a given bit rate by producing more feedback step sizes. Analysis is made of the dynamic range, companding law, and stability of the system. Comparison of C.F.D.M. with logarithmic P.C.M. has been made. It has an equivalent to 4 bit A-law PCM. Performance of C.F.D.M. has been compared with J.A.D.M. and H.I.D.M.

Compared to J.A.D.M. it offers a marginal improvement in dynamic range and has approximately the same signal to noise ratio - peak signal to noise ratio of 21.0 dBs is achieved by C.F.D.M. For signal to noise ratio, 6 dBs below the peak value, the improvements in the dynamic range of C.F.D.M. are about 43 dBm, having the same peak signal to noise ratio when compared to linear delta modulator.

Companding is achieved by producing more feedback step sizes as mentioned previously. Although it has only three bits to use for the control function, it offers a very wide dynamic range of step sizes, due to the adaptation constants used and the multipler circuit in the feedback loop. The stability of C.F.D.M. coder, is confirmed by having the best ever seen impulse response. Unlike J.A.D.M. it removes the undersirable oscillations usually occurring after the representation of the impulse. Large oscillations of the step sizes between the largest values, are not desirable because they increase one form of granular noise in the flat areas of the picture when coding video signals.

The step response of C.F.D.M. is very good. Its response is as fast as any instantaneous adaptive delta modulator[9] designed for encoding T.V. signals. It removes the objectionable hunting characteristics, the effect of which appears as a twinkle in the picture after the TV signal has changed rapidly from black to white and vice versa. Although no subjective tests have been made with our C.F.D.M. system, this objectionable twinkling effect should be removed. When the television signal makes a rapid change, the error signal increases. This would not cause any problem to the viewer, since the eye can tolerate a large amount of errors in the picture areas, containing sharp detailed edges. The effect of the response of the C.F.D.M. overshooting the video signal may not be serious because the overshooting is only for a short while, well within two or three clock periods and the response tracks the signal after such a length of time has elapsed. Further transients will be averaged by the eye and their effect should only mitigate the dimension of the spot size.

Analysis made on the tolerances of 'A' parameters of the C.F.D.M. coders indicates that the 'A' constants can have tolerances of the order of 10% which results in a degradation of signal to noise ratio of the order of 2 dBs. The persistent effect of a single channel error on the decoded signal has been studied in section 2.18. The severity of the effect of this error was found to be crucially dependent on the recent adaptations prior to the event of the error. It is recommended that a leaky integrator should be employed to mitigate the accummulative effects of these errors. Spectrum analyses have been made for the input and noise spectra of the system. The noise reduction of C.F.D.M. has been demonstrated.

From the results obtained and the analyses made, though there is the lack of subjective evaluation of the coder for video signal encoding, the C.F.D.M. seems to be promising for coding television pictures.

REFERENCES

1. de JAGER, F., "Delta Modulation, a method of P.C.M. transmission using a 1-unit code", Philips Res.Rept., No.7, pp 442-466, 1952.
2. CUTLER, C.C., "Differential quantization of communication signals", U.S. patent No. 2605361, July 29, 1952.
3. STEELE, R., and PASSOT, M., "Slope limiter model of a delta modulator in slope overload with Gaussian input signals", Joint Conf. on Digital Processing of Signals in Communications, Loughborough University, April 1972.
4. GRAFKES, J.A., and de JAGER, F., "Continuous delta modulation" Philips Res.Rept., No. 23, pp 232-290, 1968.
5. TOMOZAWA, A., and KANEKO, H., "Companded delta modulation for telephone transmission", I.E.E.E. Trans. Communication technology, vol. com. 16, No. 1, pp 109-157, Feb., 1961.
6. BROLIN, S.J., and BROWN, J.H., "Companded delta modulation for telephony", I.E.E.E. Trans. Communication Technology, vol. con.16, No.1, pp 157-162, Feb. 1968.
7. CARTMALE, A.A., and STEELE, R., "Calculating the performance of syllabically companded delta-sigma modulators", Proc. I.E.E., vol. 117, No. 10, pp 1915-1921, Oct. 1970.
8. WINKLER, M.R., "High information delta modulation", I.E.E.E. Internat'l Conv. Record, pt.8, pp 260-265, 1963.
9. WINKLER, M.R., "Pictorial transmission with H.I.D.M.", I.E.E. Internat'l Conv. Record, pt. 1, pp 260-265, 1965.
10. BOSWORTH, R.H., and CANDY, J.C., "A companded one-bit coder for picture phone transmission", Bell Sys. Tech. J, vol. 48, pp 1459-80, July 1969.

11. CANDY, J.C., "Refinement of a delta modulator", Bandwidth Reduction Symposium, Massachusetts Institute of Technology, U.S.A., 1970.

12. ABATE, J.E., "Linear and adaptive delta modulation", Proc. I.E.E.E. vol. 55, pp 298-308, Mar. 1967.

13. JAYANT, N.S., "Adaptive delta modulation with a one-bit memory"; Bell Sys. Tech.J., vol. 49, pp 321-346, Mar. 1970.

14. JAYANT, N.S., and ROSENBERG, A.E., "The preference of slope overload to granularity in the delta modulation of speech", Bell Sys. Tech.J., vol. 50, pp 3117-3125, Dec., 1971.

15. LIMB, J.O., and MOUNTS, F.W., "Digital differential quantizer for Television", Bell Sys. Tech.J, pp 2583-2599, Jan. 1969.

16. JOHNSON, F.B., "Calculating delta modulator performance", I.E.E.E. Trans. Audio and Electroacoustics, vol. AV16, No. 1, Mar.1968.

17. RABINER, R.L., GOLD, B., and McGONEGAL, C.A., "An approach to the approximation problem for non-recursive digital filters", I.E.E.E. Trans. Audio and Electroacoustics, vol. AV.18, No. 2, pp 82-105, June 1970.

18. COCHRAN, W.T., et al, "What is the fast fourier transform?", I.E.E.E. Trans. Audio and Electroacoustics, vol. Av.15, No. 2, p45, June 1967.

19. BLACKMAN, R.B., and TUKEY, J.W., "The measurement of power spectra from the point of view of communication engineering", Dover publi: Inc. New York, 1958.

20. STEELE, R., Book on Delta Modulation (to be published).

21. O'NEAL, J.B., "Predictive quantizing systems (D.P.C.M.) for the transmission of television signals", Bell Sys. Tech. J. pp 689-721, May, June, 1966.

22. CARLSON, A.B., "Communication Systems", McGraw-Hill Inc., Pub. New York, 1968.

23. JOHNSON, F.B., "Calculating delta modulator performance", I.E.E.E. Trans., Audio and Electroacoustics, vol. AU.16, pp 121-129, Mar. 1968.

24. KYAW, A.T., and STEELE, R., "Constant-Factor Delta Modulator", Electronics Letters, vol. 9, pp 96-97, 22nd Feb. 1973.

25. SMITH, B., "Instantaneous companding of quantized signals", Bell Sys. Tech.J., 36, No. 3, pp 653-709, 1st May, 1957.

APPENDIX

A.1 CALCULATION OF PROBABILITY OF ERROR IN DETECTED SIGNAL

Prior to the decoding process, is a decision circuit shown in Figure (37), which decides on the presence or absence of a pulse when noise is present. The threshold level of the decision circuit is set at zero level, if the polar binary signals are transmitted with the pulse amplitudes of +V and -V volts for one level and zero level respectively. As long as the noise amplitude is less than the pulse magnitude, there will be no errors in the detected signal. However, due to the random nature of noise, some binary signals will be detected with errors. When a +V level is transmitted, an error will occur if the signal $v+n(t) < 0$ where $n(t)$ is the noise voltage. Similarly, $-v + n(t) > 0$ causes an error when a -v level is sent. Since most of the electrical noise is Gaussian, we assume here that the noise waveform $n(t)$ has a Gaussian amplitude distribution with zero mean and variance σ^2.

Assuming the presence or absence of a pulse is equally likely, such that the probability of the transmitted code using +v is equal to that of -v,

i.e. $P_1 = P_o$, we can calculate the probability of errors in the detected signal due to the corruption of channel noise.

$$\text{Let } q_1 = v + n(t) \quad \text{and} \qquad \ldots\ldots\ldots\ldots \qquad (75)$$

$$q_o = -v + n(t) \qquad \ldots\ldots\ldots\ldots \qquad (76)$$

Since $n(t)$ is assumed to have a Gaussian density function,

$$p(n) = \frac{1}{\sigma\sqrt{2\pi}} \; e^{-\frac{n^2}{2\sigma^2}} \qquad \ldots\ldots\ldots\ldots \qquad (77)$$

It follows that q_1 and q_o are also gaussian with variance σ^2 but with mean values of +v and -v respectively. Figure (38) illustrates the density functions of $P(q_1)$ and $P(q_o)$. Since $q_1 < 0$ or $q_o > 0$ will make an error when a +v or -v level is transmitted, the probability of errors can be calculated as,

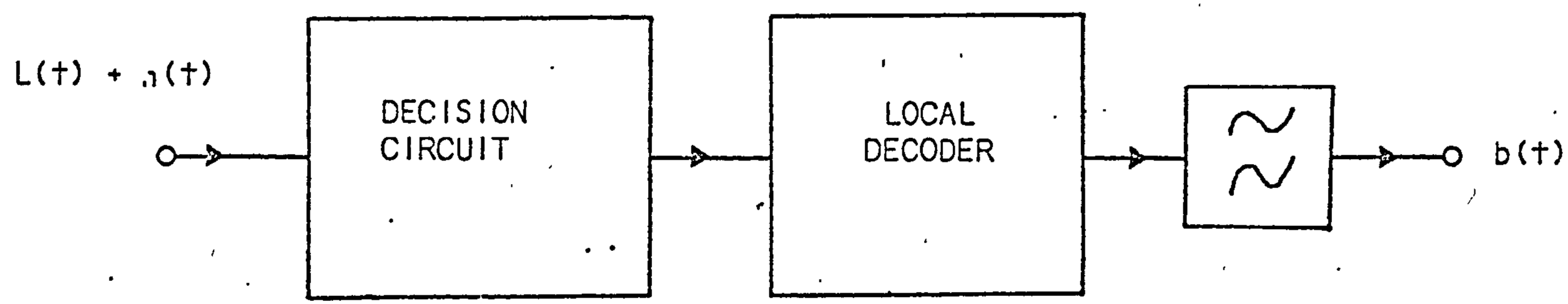

Figure (37) C.F.D.M. DECODER IN THE PRESENCE OF CHANNEL ERRORS.

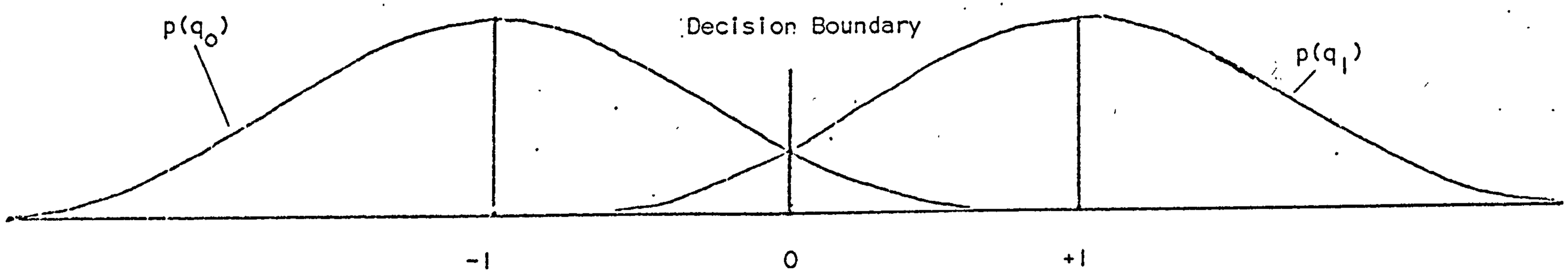

Figure (38) PROBABILITY DENSITY FUNCTION OF q_1 and q_o'.

If 'one' is transmitted, the probability of error is

$$Pe_1 = \int_{-\infty}^{0} p(q_1)\ dq_1 \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (78)$$

$$Pe_1 = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{0} e^{\frac{-(q_1-v)^2}{2\sigma^2}}\ dq \quad \ldots\ldots\ldots\ldots \quad (79)$$

Similarly, if 'zero' is transmitted, the probability of error is,

$$Pe_o = \frac{1}{\sigma\sqrt{2\pi}} \int_{0}^{\infty} e^{\frac{-(q_o+v)}{2\sigma^2}}\ dq_o \quad \ldots\ldots\ldots\ldots \quad (80)$$

clearly,

$$Pe_1 = Pe_o = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{\frac{v}{\sigma}}^{\infty} e^{-\frac{\mu^2\sigma}{2}}\ d\mu \quad \ldots\ldots\ldots\ldots \quad (81)$$

$$= \frac{1}{\sqrt{2\pi}} \int_{+\frac{v}{\sigma}}^{\infty} e^{\frac{-(\mu^2)}{2}}\ d\mu \quad \ldots\ldots\ldots\ldots \quad (82)$$

$$Pe_1 = Pe_o = Q\left(\frac{v}{\sigma}\right)$$

If P_o is the probability of an element '-v' is transmitted and P_1 the probability of an element '+v', then the probability of an error is,

$$Pe = (P_o + P_1)\ Q\left(\frac{v}{\sigma}\right) = Q\left(\frac{v}{\sigma}\right) \quad \ldots\ldots\ldots\ldots \quad (83)$$

Thus, knowing the pulse magnitudes and the variance of noise in the channel, the probability of error can be easily manipulated.

In our case the magnitude of the binary levels become +1 or -1. Therefore Pe becomes

$$Pe = Q\left(\frac{1}{\sigma}\right) \quad \ldots\ldots\ldots\ldots \quad (84)$$

This can be readily calculated from the Q functions tables or equivalently from the Φ functions as

$$Pe = \frac{1}{2} - \Phi\left(\frac{1}{\sigma}\right) \qquad \ldots\ldots\ldots\ldots\ldots \qquad (85)$$

where $\Phi\left(\frac{1}{\sigma}\right) = \frac{1}{\sqrt{2\pi}} \int_{0}^{\frac{1}{\sigma}} e^{-\frac{\mu^2}{2}} \, d\mu$

For the expected error rate of 1 in 10^3, the standard deviation of noise can be calculated as

$$\Phi\left(\frac{1}{\sigma}\right) = 0.5 - .001$$

Looking up from the $\Phi\left(\frac{K}{\sigma}\right)$ from the table[21], σ has been calculated as $\sigma = 0.32$ in which K is taken as 1.

This value of σ is used for our simulation for the analysis of the effect of channel errors.

A.2. COMPUTER PROGRAMMES

A.2a. Main programme for the simulation and plotting of dynamic range and compression characteristics of C.F.D.M.

The flow chart for this purpose is listed in Figure (39)1.

It required two other subroutines INVALUE and ENCODER.

These subroutines are subsequently described in detail later.

The main programme for this is listed in Figure (39)1a.

A.2b. Main programmes for the simulation and plotting of step size distribution and the estimation of power spectrum of C.F.D.M.

The programme for this purpose is shown in figure (39)2a.

It requires five other subroutines namely FIR, ENCODER, SHARP, SMOOTH, and GRAF1. The details of these subroutines are also described in the following section. The flow chart of this main programme is listed in Figure (39)2.

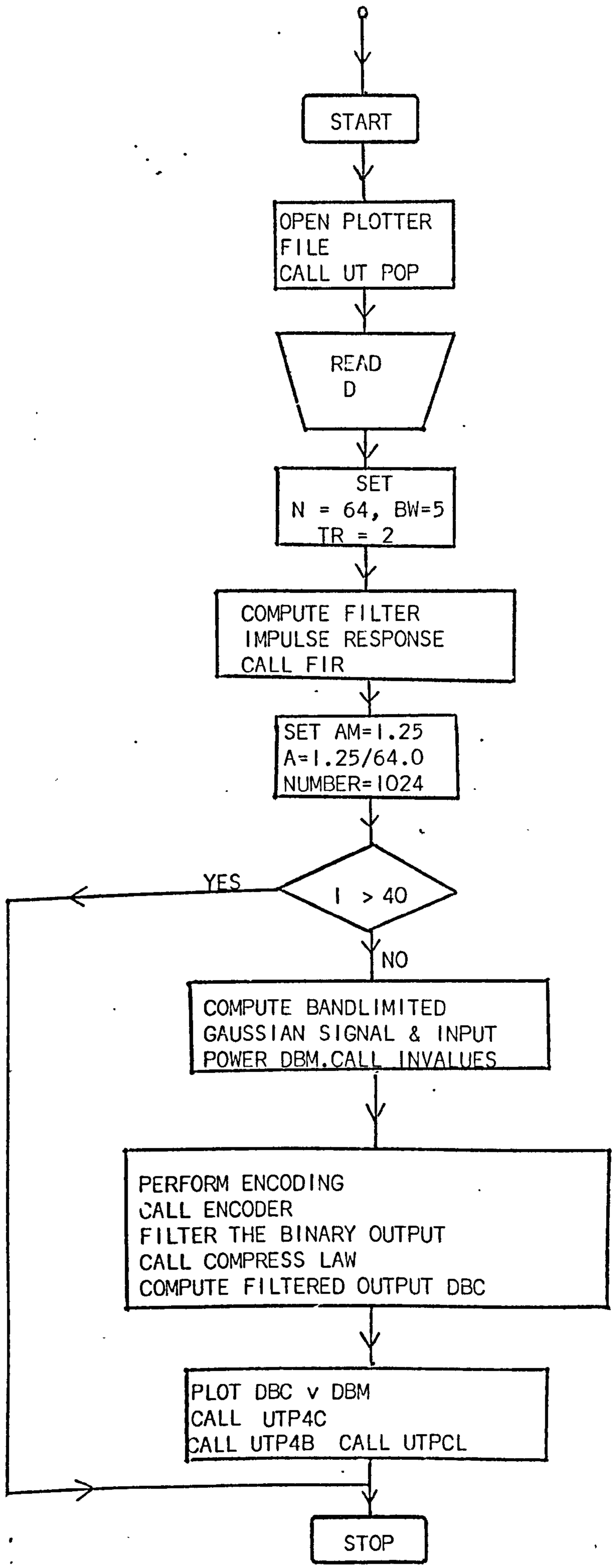

Figure (39)1. FLOW CHART OF THE MAIN PROGRAMME FOR THE SIMULATION AND PLOTTING OF DYNAMIC RANGE AND COMPRESSION CHARACTERISTICS OF C.F.D.M.

```
      MASTER C F D M
      DIMENSION AX(40),AY(40),AG(40),AGG(40),AZ(40),AF(40),AP(40)
      DIMENSION B(64)
      DIMENSION AL(40)
      DIMENSION AXX(40),AYY(40)
C     ***********************************************************
C     THREE BIT MEMORY ADAPTIVE DM
C     THIS INCORPORATES A DETECTION PROCESS FOR CONCECUTIVE LIKE BITS
C     AND  ALSO A NEW ADAPTATION CONSTANT IS SELECTED AT
C     EVERY SAMPLING INSTANT
C     ***********************************************************
      CALL UTPOP
      READ(1,20)D
   20 FORMAT(F0.0)
      N=64
      BU=5
      TR=3
      CALL FIR(N,BU,TR,B)
      AM=1.125
      A=1.25/64.0
      NUMBER=1024
      DO 28  I=1,40
   12 D=0.03
      CALL INVALUES(NUMBER,B,A,DBM,KR)
      CALL ENCODER(D,NUMBER,B,SNR,SSNR,STN,DBMO,DBC,SNRAF)
   15 CONTINUE
      A=A*AM
      AX(I)=DBM
      AF(I)=DBC
      AL(I)=SNRAF
   28 CONTINUE
      CALL UTP4C (-45.0,45.0,0.0,21.0,6.0,3.0,9HINPUT DBM,2,7HSNR DBS,-
      CALL UTP4B(AX,AL,40,2)
      CALL UTP4C(-40.0,40.0,-30.0,10.0,8.0,4.0,
     19HINPUT DBM,2,10HOUTPUT DBM,2)
      CALL UTP4B(AX,AF,40,2)
      CALL UTPCL
      STOP
      END

MENT, LENGTH   119, NAME  CFDM
```

Figure (39) 1a. MAIN PROGRAMME FOR THE SIMULATION AND PLOTTING OF DYNAMIC RANGE AND COMPRESSION CHARACTERISTICS OF C.F.D.M.

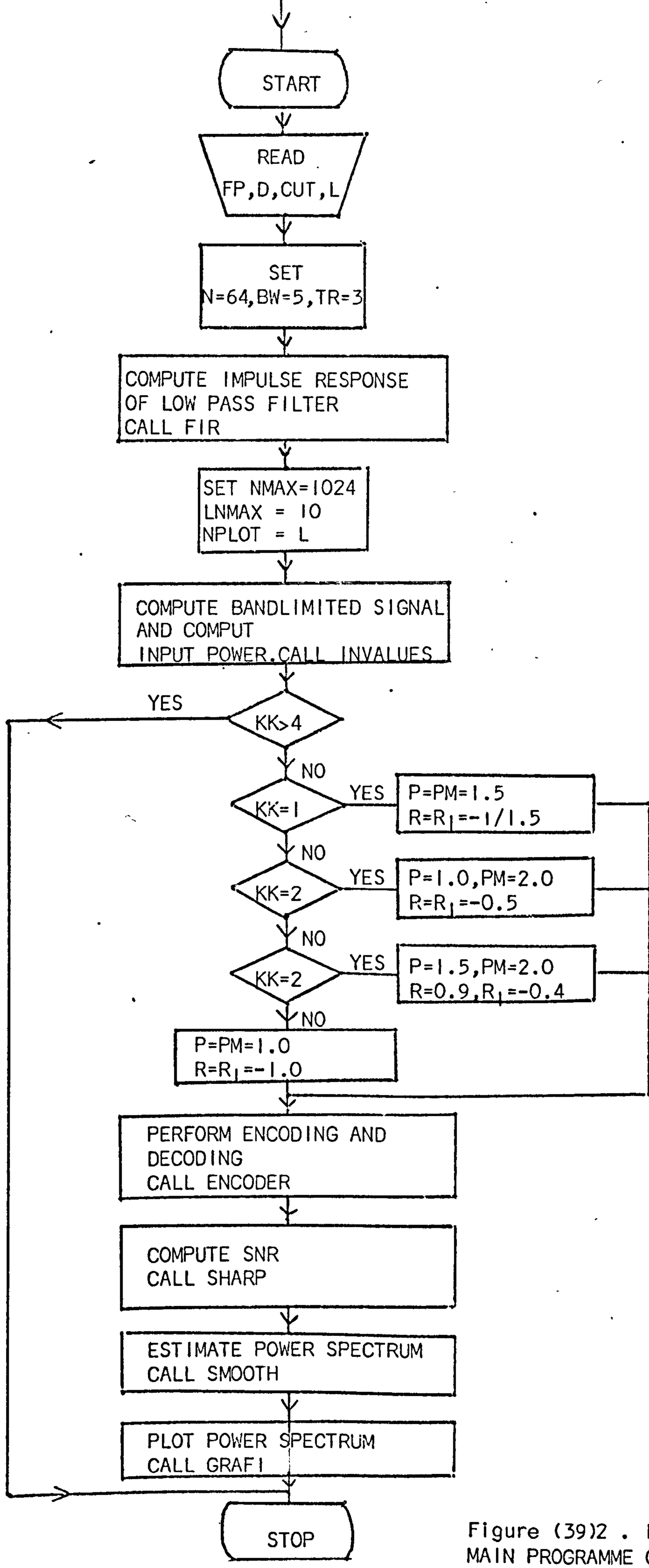

Figure (39)2 . FLOW CHART OF THE MAIN PROGRAMME OF C.F.D.M.

```
      COMPLEX CY
      COMMON S(1024),FKP(1024),CV(1024),GMSQ(1087),A(1024),AM(1087),
     1CY(1024),B(64),FIP(64)
      CALL DTIOP
      READ(1,20) FP,D,CUT
   20 FORMAT(3F10.0)
      READ(1,40) L
   40 FORMAT(I2)
C     FP=6=CLOCK FREQ
C     D=MINIMUM STEP SIZE
C     CUT=LOW PASS CUT OFF FREQ
C     L=NUMBER OF SAMPLES
      N=64
      BW=5
      TR=3
      CALL FIP(N,FW,TP,IN)
      NMAX=1024
      LMAX=10
      NPLOT=L
      CALL INVALUES(FP,L,RMS,PWER)
      DO 800 KK=1,4
      IF(KK.EQ.1) GO TO 443
      IF(KK.EQ.2) GO TO 441
      IF(KK.EQ.3) GO TO 445
      P=1.0
      PM=1.0
      R=-1.0
      RL=-1.0
      GC=-1.0
      GG=1.0
      GO TO 889
  443 P=1.5
      PM=1.5
      R=-1/P
      RL=R
      GC=R
      GG=1.5
      GO TO 889
  441 P=1.0
      PM=2.0
      R=-0.5
      RL=R
      GC=P
      GG=1.5
      GO TO 889
  445 P=1.5
      PM=2.0
      R=-0.9
      RL=-0.4
      GC=-0.4
      GG=1.5
  889 CONTINUE
      WRITE(2,89)
   89 FORMAT(3X,'FP                D                CUT                NUMBER')
      WRITE(2,88) FP,D,CUT,L
   88 FORMAT(3X,3F10.5,4X,I3)
      CALL ENCODER(D,L,P,PM,R,RL,GG,GC)
      G=FP
      CALL  SEARC(NMAX,LMAX,G,CUT,L,FOND,NBPASS,
     2VNOISE,SOR,VHOLD,PWER)
      CALL SHOOT(L)
      CALL GRAF1(D,VHOLD,L)
```

```
 70 FORMAT(1 0 ,2X,11HNOISE POWER,10X,12HSIGNAL/NOISE,/
   123(,15HTIME INTERVAL,11X,9HRATIO DBS)
    WRITE(2,75) WPS,WNOISE,SND
 75 FORMAT(10X,9HINPUT AND,F6.3,F10.6,15X,F6.1) .
 76 CONTINUE
800 CONTINUE
    CALL DTICL
    STOP
    END
```

Figure (39)2a MAIN PROGRAMME OF C.F.D.M.

A.2.1 SUBROUTINE FIR

It is listed in Figure (39).3 It demonstrates how the Finite Duration Impulse Response of digital low pass filter may be designed. It is written in FORTRAN 4 and is called by this statement.

```
CALL FIR (N, BW, TR, B)
```

Where,

N = number of samples representing frequency characteristic of the filter.

BW = number of samples representing the pass band of the filter.

TR = number of samples representing the transition bands of the filter.

B = a real array of dimension N.

This subroutine requires another subroutine NLOGN, listed in Figure (42). NLOGN subroutine is used to take the discrete fourier transform of the frequency samples. The call statement is,

```
CALL NLOGN (LN, X, N, DIR)
```

The output of NLOGN gives the impulse response of the filter. To have the actual realizable filter, all the impulse response samples are shifted back by half the number of samples.

```
      SUBROUTINE FIR(N,BW,TR,B)
      COMPLEX X(64),H(64)
      DIMENSION B(64)
      READ(1,25) (H(N),N=1,64)
   25 FORMAT(8(F0.0,F0.0))
      DO 90 I=1,64
   90 X(I)=H(I)
      CALL NLOGN(6,X,64,1.0)
      I=33
      DO 56 J=1,64
      IF(J.LE.32) GO TO 14
      IF(J.EQ.33) GO TO 18
      IF(J.GT.33) GO TO 17
   14 B(J)=REAL(X(I))
      I=I-1
      GO TO 56
   18 M=1
   17 B(J)=REAL(X(M))
      M=M+1
   56 CONTINUE
      RETURN
      END

SEGMENT. LENGTH   139, NAME   FIR
```

Figure (39) 3. PROGRAMME FOR COMPUTING IMPULSE RESPONSE OF LOW-PASS FILTER.

A.2.2 SUBROUTINE INVALUES

It computes the samples of bandlimited Gaussian signals. The programme is listed in Figure (40). The call statement for this is,

CALL INVALUES (NUMBER, B, A, DBM, KR)

The arguments of the subroutine are,

NUMBER = number of samples of input

B = real array for impulse response of lowpass filter.

A = standard deviation of Gaussian input.

DBM = Input signal power in dBm

KR = Initial setting of the random number generator.

The input to the subroutine is the random number generated from the function UTRI. It requires another subroutine FILTER.

This subroutine is used to bandlimit the gaussian input signal. The call statement is,

CALL FILTER (N, B, NUMBER, VIN, NM, Q)

where N = length of impulse response

VIN = real array for gaussian input

NM = length of the filtered output

Q = real array for the bandlimited gaussian signals.

```
      SUBROUTINE INVALUES(NUMBER,B,A,DBM,KR)
      DIMENSION B(64)
      COMMON VIN(1024),ERR(1087),CV(1024),Q(1087)
      KR=0
      DO 70 I=1,NUMBER
      VIN(I)=UTR1(1,1,KR)*A
   70 CONTINUE
      N=64
      NM=NUMBER+N-1
      CALL FILTER(N,B,NUMBER,VIN,NM,Q)
       J=34
      DO 80 I=1,1024
      VIN(I)=Q(J-I)
      J=I+2
   80 CONTINUE
      AMEAN=0.0
      AK=0.0
      DO 10 I=1,NUMBER
      AMEAN=AMEAN+VIN(I)
   10 AK=AK+VIN(I)**2
      AMEAN=AMEAN/NUMBER
      BINP=AK/NUMBER
      CRMS=SQRT(BINP)
      AMSQ=BINP
      DBM=10.0*ALOG10(AMSQ/0.6)
      WRITE(2,22) AMEAN,CRMS,ARMS,VIN(1)
   22 FORMAT(4F16.5)
      WRITE(2,87)
   87 FORMAT(3X,'INPUT POWER IN DBM')
      WRITE(2,88) DBM
   88 FORMAT(F16.5)
      RETURN
      END

SEGMENT, LENGTH   165. NAME  INVALUES
```

Figure (40) PROGRAMME FOR COMPUTING BANDLIMITED GAUSSIAN SIGNAL AND INPUT POWER IN DBM.

A.2.3 SUBROUTINE FILTER

Subroutine FILTER performs the filtering. Filtering is achieved by performing polynomial multiplication or equivalently performing the complete transient convolution of two signals. The call statement is,

```
CALL FILTER (LA, A, LB, LC, C)
```

where the subroutine inputs are,

LA = length of array A

A = real array of impulse response

LB = length of array B

B = real array of input samples to the filter

and the subroutine outputs are,

LC = length of array C.

C = real array of output samples of the filter

The programmes for this subroutine are listed overleaf in Figure (41).

```
      SUBROUTINE FILTER(LA,A,LB,B,LC,C)
      DIMENSION A(LA),B(LB),C(LC)
      LC=LA+LB-1
      DO 2 I=1,LC
    2 C(I)=0.0
      DO 1 I=1,LA
      DO 1 J=1,LB
      K=I+J-1
    1 C(K)=C(K)+A(I)*B(J)
      RETURN
      END

SEGMENT, LENGTH    94, NAME  FILTER
```

Figure (41) PROGRAMME FOR LOW-PASS FILTERING

A.2.4 SUBROUTINE NLOGN

It computes the fourier transforms and inverse fourier transforms of the real array x of dimension LX. LX is made an integral power of two. The transformed data is left in the array x. The real variable DIR specifies whether a direct or inverse transform is to be computed.

DIR = -1.0 for a direct transform or

= 1.0 for an inverse transform

The call statement for this subroutine is,

CALL NLOGN (N, x, LX, DIR)

Figure (42) shows the programmes for this subroutine.

```
      SUBROUTINE NLOGN(N,X,LX,DIR)
      COMPLEX X,WK,HOLD,Q
      DIMENSION M(25)
      DIMENSION X(LX)
      DO 1I=1,N
    1 M(I)=2**(N-I)
      DO 4 L=1,N
      NBLOCK=2**(L-1)
      LBLOCK=LX/NBLOCK
      LBHALF=LBLOCK/2
      K=0
      DO 4 IBLOCK=1,NBLOCK
      FK=K
      FLX=LX
      V=DIR*6.28318531*FK/FLX
      WK=CMPLX(COS(V),SIN(V))
      ISTART=LBLOCK*(IBLOCK-1)
      DO 2 I=1,LBHALF
      J=ISTART+I
      JH=J+LBHALF
      Q=X(JH)*WK
      X(JH)=X(J)-Q
      X(J)=X(J)+Q
    2 CONTINUE
      DO 3 I=2,N
      II=I
      IF(K.LT.M(I)) GO TO 4
    3 K=K-M(I)
    4 K=K+M(II)
      K=0
      DO 7 J=1,LX
      IF(K.LT.J) GO TO 5
      HOLD=X(J)
      X(J)=X(K+1)
      X(K+1)=HOLD
    5 DO 6 I=1,N
      II=I
      IF(K.LT.M(I)) GO TO 7
    6 K=K-M(I)
    7 K=K+M(II)
      IF(DIR.LT.0.0) RETURN
      DO 8 I=1,LX
    8 X(I)=X(I)/FLX
      RETURN
      END
```

Figure (42) PROGRAMME FOR THE FAST FOURIER TRANSFORM.

A.2.5 SUBROUTINE ENCODER

It performs all the encoding and decoding functions of C.F.D.M. system of Figure (2). Input is stored in the real array S of dimension L. It requires subroutine HISTOGM and subroutine FILTER, to draw the step size utilized in the simulation and to filter the integrated signal, to reconstitute the baseband signal at the decoder output. The call statement for this subroutine is,

```
CALL ENCODER (D, L, P, PM, R, RL, GG, GC)
```

where,

D = initial step size

L = number of samples

P = adaptation constant for group 3

PM = adaptation constant for group 4

R = adaptation constant for group 1

RL = adaptation constant for group 2

GG = P

GC = R

It uses two other subroutines HISTOGM and FILTER.

The programme for ENCODER is shown in Figure (43) and (43)a.

```
      SUBROUTINE FGROUPP(D,L,P,PM,R,RL,GG,GC)
      COMPLEX CY
      COMMON S(1024),ERR(1024),CV(1024),GASQ(1087),A(1024),AG(1087),
     1CY(1024),F(64),FIN(64)
      INTEGER Q
      DIMENSION Q(1024)
      VINT=S(1)+0.015
      STEP=0.03
      DM=5.0
      AP1=R
      AP2=RL


      AP3=D
      AP4=PM
      WRITE(2,14) Ap1,Ap2,Ap3,Ap4
   14 FORMAT(4F5.2)
      DO 70 I=1,L
      IF(I.EQ.1) GO TO 33
      E=S(I)-VINT
      ERR(I)=E
      GO TO 35
   33 E=S(I)-CV(I-1)
      ERR(I)=E
      IF(E.GE.0.0) GO TO 40
      VOUT=-1.0
      GO TO 34
   40 VOUT=1.0
   34 Q(I)=VOUT
      GO TO 50
   35 IF(E.GE.0.0) GO TO 36
      VOUT=-1.0
      GO TO 37
   36 VOUT=1.0
   37 Q(I)=VOUT
      IF(Q(I).EQ.1.0) GO TO 38
      CV(I)=VINT-STEP
      GO TO 61
   38 CV(I)=VINT+STEP
      GO TO 61
   50 IF(I.GE.3) GO TO 51
      IF(Q(I).EQ.Q(I-1)) GO TO 53
      CV(I)=CV(I-1)+(CV(I-1)-VINT)*GC
      GO TO 61
   53 CV(I)=CV(I-1)+(CV(I-1)-VINT)*GG
      GO TO 61
C     ADAPTATION ALGORITHM  THE DETECTION OF FOUR POSSIBLE GROUPS
   51 IF(Q(I-2).NE.Q(I-1)) GO TO 8
      IF(Q(I-1).EQ.Q(I)) GO TO 7
      CV(I)=CV(I-1)+(CV(I-1)-CV(I-2))*R
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)    CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
    7 CV(I)=CV(I-1)+(CV(I-1)-CV(I-2)) *P
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)    CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
    8 IF(Q(I-1).EQ.Q(I)) GO TO 6
      CV(I)=CV(I-1)+(CV(I-1)-CV(I-2)) *RL
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)    CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
    6 CV(I)=CV(I-1)+(CV(I-1)-CV(I-2))*PM
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)    CV(I)=CV(I-1)+DM*Q(I)
   61 CONTINUE
```

```
70 CONTINUE
   WRITE(2,85)
85 FORMAT(3X,'PROBABILITY')
   AMAX=10.0
   AMIN=-10.0
   CALL HISTOGM(AMIN,AMAX,L)
   DO 80 K=1,L
80 GUSO(K)=0.0
   N=64


   NM=N+L-1
C  DECODED OUTPUT  BY PASSING THE INTEGRATED OUTPUT THROUGH THE LOW PASS
C  FILTER , TO GIVE THE APPROXIMATION TO THE INPUT SIGNAL
   CALL FILTER(N,K,L,CV,NM,AN)
   J=34
   DO 600 I=1,L
   A(I)=AN(J-1)
   J=J+2
600 CONTINUE
   RETURN
   END
```

Figure (43) PROGRAMME FOR C.F.D.M. ENCODER AND DECODER.

```
      SUBROUTINE ENCODER(D,NUMBER,B,SNR,SSNR,STN,DBMO,DBC,SNRAF
      COMMON VIN(1024),ERR(1087),CV(1024),Q(1087)
      DIMENSION B(64)
      NP=NUMBER
      DU=5.0
      STEP=0.03
      VINT=VIN(1)+0.015
      DO 97 II=3,3
      IF(II.EQ.1) GO TO 22
      IF(II.EQ.2) GO TO 23
      IF(II.EQ.3) GO TO 26
      P=1.0
      PH=1.0
      R=-1.0
      RL=-1.0
  104 GG=1.0
      GC=-1.0
      GO TO 66
   22 P=1.5
      PH=1.5
      R=-1/P
      RL=R
  106 GG=1.5
      GC=-1/1.5
      GO TO 66
   23 P=1.5
      PH=P
      R=-1/1.6
      RL=R
  105 GG=1.5
      GC=-1/1.6
      GO TO 66
   26 P=1.5
      PH=2.0
      R=-0.9
      RL=-0.4
      GG=1.5
      GC=-0.4
   66 COUNT=0.0
      WRITE(2,55) P,R,RL,PH
   55 FORMAT(1H ,4F5.2)
      DO 70 I=1,NP
      IF(I.NE.1) GO TO 33
      E=VIN(I)-VINT
      ERR(I)=E
      GO TO 35
   33 E=VIN(I)-CV(I-1)
      ERR(I)=E
      IF(E.GE.0.0) GO TO 40
      VOUT=-1.0
      GO TO 34
   40 VOUT=1.0
   34 Q(I)=VOUT
      GO TO 50
   35 IF(E.GE.0.0) GO TO 36
      VOUT=-1.0
      GO TO 37
   36 VOUT=1.0
   37 Q(I)=VOUT
      IF(Q(I).EQ.1.0) GO TO 38
      CV(I)=VINT-STEP
```

```
      GO TO 61
   38 CV(I)=VINT+STEP
      GO TO 61
   50 IF(I.GE.3) GO TO 51
      IF(Q(I).EQ.Q(I-1)) GO TO 53
      CV(I)=CV(I-1)+(CV(I-1)-VINT)*GC
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)   CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
   53 CV(I)=CV(I-1)+(CV(I-1)-VINT)*GG
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)   CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
   51 IF(Q(I-2).EQ.Q(I-1)) GO TO 8
      IF(Q(I).EQ.Q(I-1)) GO TO 7
      CV(I)=CV(I-1)+(CV(I-1)-CV(I-2))*R
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)   CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
    7 CV(I)=CV(I-1)+(CV(I-1)-CV(I-2))*P
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)   CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
    8 IF(Q(I-1).EQ.Q(I)) GO TO 6
      CV(I)=CV(I-1)+(CV(I-1)-CV(I-2))*RL
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)   CV(I)=CV(I-1)+DM*Q(I)
      GO TO 61
    6 CV(I)=CV(I-1)+(CV(I-1)-CV(I-2))*PM
      IF(ABS(CV(I)-CV(I-1)).LT.D) CV(I)=CV(I-1)+D*Q(I)
      IF(ABS(CV(I)-CV(I-1)).GE.DM)   CV(I)=CV(I-1)+DM*Q(I)
   61 CONTINUE
   70 CONTINUE
      CALL COMPRESS LAW(NUMBER,DBC,B)
      N=64
      NM=N+NUMBER-1
      CALL FILTER(N,B,NUMBER,CV,NM,Q)
      J=74
      DO 41 I=1,NP
      Q(I)=Q(J-I)
      J=J+2
   41 CONTINUE
      CALL SHARP FILTER(NUMBER,AM,SNR,SSNR,STN,SNRAF)
   97 CONTINUE
  160 RETURN
      END

GMENT, LENGTH 1034, NAME ENCODER
```

Figure (43)a.—PROGRAMME FOR PERFORMING ENCODING AND DECODING OF C.F.D.M. SYSTEM AND FOR THE CALCULATION OF SNR.

A.2.6 SUBROUTINE HISTOGM

Subroutine HISTOGM computes the step sizes utilized, stored in the real array CV of dimension 1024, and divides the step sizes into various slots of step sizes which are stored in the real array C of dimension 100. Each slot has the range of 0.25 volt. It also computes the probability of occurrence of each slot of step size and stores in the real array PROB of dimension 100. This subroutine also plots the histogram of step sizes utilized in the simulation by plotting PROB against C. The call statement is,

CALL HISTOGM (AMIN, AMAX, L)

AMIN = minimum step size for the plotting of Histogram

AMAX = maximum step for the plotting of Histogram

L = number of samples

The programme of this subroutine is listed in Figure (44) overleaf.

```
      SUBROUTINE HISTOGR(AMIN,AMAX,L)
      COMPLEX CY
      COMMON S(1024),FRE(1024),CV(1024),GNSU(1087),A(1024),AX(1087),
     1CY(1024),P(64),FIP(64)
      DIMENSION ICOUNT(100),C(100),PR(100),PROB(200),CLOCK(200)
      I=1
      A2=AMIN
   10 A1=A2
      A2=A1+0.25
      C(I)=A1
      C(I+1)=A2
      IF(A2.GE.AMAX) GO TO 335
      I=I+1
      GO TO 10
  335 MM=I
      DO 225 J=1,MM
  225 ICOUNT(J)=0
      DO 300 II=1,L
      DO 200 JJ=1,MM
      IF(II.EQ.1) GO TO 616
      IF((CV(II)-CV(II-1))-C(JJ).LE.0.0) GO TO 200
      IF((CV(II)-CV(II-1))-C(JJ+1).GT.0.0) GO TO 206
      ICOUNT(JJ)=ICOUNT(JJ)+1
      GO TO 360
  616 IF((CV(II)-VINT)-C(JJ).LE.0.0) GO TO 200
      IF((CV(II)-VINT)-C(JJ+1).GT.0.0) GO TO 200
      ICOUNT(JJ)=ICOUNT(JJ)+1
      GO TO 360
  200 CONTINUE
  300 CONTINUE
      DO 460 I=1,MM
  400 PR(I)=(FLOAT(ICOUNT(I))/FLOAT(L))
      DO 414 I=1,MM
  414 WRITE(2,444) PR(I)
  444 FORMAT(3X,F5.2)
      XMAX=10.0
      XMIN=-10.0
      YMIN=0.0
      YMAX=0.5
      XINS=4.0
      YINS=4.0
      NA=2*MM
      NX=1
      NY=2
      CALL UTP4A(XMIN,XMAX,YMIN,YMAX,XINS,YINS,SH STEP,NX,
     111HPROBABILITY,NY)
      DO 189 I=1,MM
      CLOCK(2*I-1)=C(I)
  189 CLOCK(2*I)=C(I)
      DO 88 I=1,MM
      IF(I.EQ.1) GO TO 13
      GO TO 12
   13 PROB(1)=PR(1)
      GO TO 88
   12 PROB(2*I-1)=PR(I-1)
   88 PROB(2*I)=PR(I)


      DO 560 J=1,MM
  560 C(J)=J
      CALL UTP4B(CLOCK,PROB,NA,3)
      RETURN
      END
```

A.2.7 SUBROUTINE SHARP

It is used for the computation of signal power, quantization noise power and the signal to noise ratio. The subroutine sharp is listed in Figure (45). This subroutine may be called by the statement,

CALL SHARP (NMAX, LNMAX, G, CUT, L, FUND, NHPASS, VNOISE, SNR, VHOLD, PWER)

The arguments of this subroutine are,

NMAX = number of samples which are integral power of two

LNMAX = logorithm of M_{max} to the base two

G = clock frequency

CUT = cut off frequency of low pass filter

L = number of samples used in the simulation

FUND = elementary frequency band

NHPASS = number of element frequency bands required to represent the cut off frequency

VNOISE = RMS noise voltage

SNR = signal to noise ratio

YHOLD = sample of maximum power in the input power spectrum

PWER = input signal power.

It requires another subroutine NLOGN to compute the frequency of input and noise signals. DIR = 1.0 is used to compute the direct fourier transforms.

```
      SUBROUTINE   SHARP(NMAX,LNMAX,G,CUT,L,FUND,NHPASS,VNOISE,SNR,
     2YHOLD,PUFF)
      COMPLEX CY
      COMMON S(1024),EFF(1024),CY(1024),GNSQ(1087),A(1024),AN(1087),
     1CY(1024),B(64),FIN(64)
      DIR=-1.0
      DO 80 K=1,L
   80 GNSQ(K)=0.0
      DO 90 I=1,L
   90 CY(I)=CMPLX(S(I),0.0)
      CALL NLOGN(LNMAX,CY,NMAX,DIR)
      DO 70 I=1,L
   70 GNSQ(I)=CABS(CY(I))**2
      YHOLD=GNSQ(1)
      DO 31 J=1,L
      IF(YHOLD.GT.GNSQ(J)) GO TO 31
      YHOLD=GNSQ(J)
   31 CONTINUE
      WRITE(2,32) YHOLD
   82 FORMAT(3X,F14.4)
      FP=G
      FUND=FF/L
      NHPASS=CUT/FUND
      SM=0.0
      DO 59 I=2,NHPASS+1
   59 SM=SM+GNSQ(I)
      SM=2.0*SM+GNSQ(1)
      DO 10 N=1,NMAX
   10 CY(N)=CMPLX((S(N)-A(N)),0.0)
      CALL NLOGN(LNMAX,CY,NMAX,DIR)
      DO 20 N=1,NMAX
      GNSQ(N)=CABS(CY(N))**2
   20 CONTINUE
      SMGNSQ=0.0
      DO 30 N=2,NHPASS+1
   30 SMGNSQ=SMGNSQ+GNSQ(N)
      SMGNSQ=SMGNSQ*2.0+GNSQ(1)
      VNOISE=SQRT(SMGNSQ/NMAX)
      SSNR=10.0*ALOG10(SM/SMGNSQ)
      SMGNSQ=SMGNSQ/NMAX
      SNR=10.0*ALOG10(PWER/SMGNSQ)
      WRITE(2,88) SNR
   88 FORMAT(3X,F10.4)
      PSIG=0.0
      PNOISE=0.0
      DO 130 I=1,L
      PSIG=PSIG+S(I)*S(I)
      PNOISE=PNOISE+(S(I)-A(I))**2
  130 CONTINUE
      DSNR=10.0*ALOG10(PSIG/PNOISE)
      WRITE(2,888) DSNR,SSNR
  888 FORMAT(3X,2F10.4)
      RETURN
      END

EGMENT, LENGTH    362, NAME   SHARP
```

Figure (45) PROGRAMME FOR CALCULATING SIGNAL-TO-NOISE RATIO.

A.2.8 SUBROUTINE SMOOTH

It performs the estimation of power spectrum of random signal datas. It computes the autocorrelation function of the power spectrum provided in the real array GNSQ of dimension L. It requires the subroutine NLOGN and subroutine WINDOW. DIR = +1.0 is needed for inverse fourier transform, and DIR = -1.0 is needed for direct fourier transform. The call statement is,

```
CALL SMOOTH (L)
```

The programme is listed in figure (46)

```
      SUBROUTINE SMOOTH(L)
      COMPLEX CY
      COMMON S(1024),ERR(1024),CV(1024),GNSO(1087),A(1024),AN(1087),
     1CY(1024),B(64),FIR(64)
      DO 90 I=1,L
   90 CY(I)=CMPLX(GNSO(I),0.0)
      NW=10
      DIR=1.0
      CALL FLOGN(NW,CY,L,DIR)
      I=256
      DO 69 J=1,L
      IF(J.LE.256) GO TO 84
      IF(J.EQ.257) GO TO 108
      IF(J.GT.257) GO TO 87
   84 GNSO(J)=REAL(CY(I))
      I=I-1
      GO TO 69
  108 I=2
   87 GNSO(J)=REAL(CY(I))
      I=I+1
   69 CONTINUE
      CALL WINDOW (L)
      DO 55 I=1,L
   55 CY(I)=CMPLX(GNSO(I),0.0)
      DIR=-1.0
      CALL FLOGN(NW,CY,L,DIR)
      DO 57 I=1,L
   57 GNSO(I)=CABS(CY(I))
      RETURN
       END
```

Figure (46) PROGRAMME FOR ESTIMATION OF POWER SPECTRUM.

A.2. 9 SUBROUTINE WINDOW

It is listed in Figure (47). It performs the multiplication of the autocorrelation function stored in the real array GNSQ of dimension L, and the lag window function stored in the real array AN of dimension L.

The output of this subroutine is the weighted autocorrelation function stored in the real array GNSQ of dimension L. The call statement for this subroutine is,

```
CALL WINDOW (L)
```

```
      SUBROUTINE WINDOW(L)
      COMPLEX CY
      COMMON S(1024),FPF(1024),CV(1024),GESO(1037),A(1024),AN(1037),
     1CV(1024),H(64),FIN(64)
      PI=3.14159
      DO 20 I=1,L
      K=I-1
   20 AN(I)=0.54-0.46*COS(2.0*PI*K/L)
      DO 30 I=1,L
   30 GESO(I)=GESO(I)*AN(I)
      RETURN
      END
```

Figure (47) PROGRAMME FOR WINDOWING THE AUTOCORRELATION FUNCTION.

A.2.10 SUBROUTINE GRAFI

It plots the input signal waveform, decoded signal and the staircase approximated signal stored in the real arrays of VIN, BB and CV of dimensions 1087, 1024, 1024, respectively. The programme is listed in figure (48), and the call statement for this subroutine is,

CALL GRAFI (D, NUMBER)

where,

D = step size of the coder, and,

NUMBER = number of samples

```
      SUBROUTINE GRAF1(D,YHOLD,L)
      COMPLEX CY
      COMMON S(1024),FRP(1024),CY(1024),GNSQ(1067),A(1024),AN(1087),
     1CY(1024),P(64),PIN(64)
      DIMENSION CLOCK(1024)
      NPLOT=1
      G=40.0
      FOLD=G/L-1
      CLOCK(1)=0.0
      DO 20 I=1,NPLOT
      GNSQ(I)=GNSQ(I)/YHOLD
      GNSQ(I)=10.0*ALOG10(GNSQ(I))
   80 CLOCK(I)=FOLD*(I-1)
      XMAX=FOLD*NPLOT/4
      NPOINT=L/4
      CALL UTP4C(6.0,XMAX,-50.0,0.0,4.0,2.0,5HF KHZ,NX,
     19HPOWER AMP,NY)
      CALL UTP4B(CLOCK,GNSQ,NPOINT,2)
      YHOLD=GNSQ(1)
      DO 31 J=2,L
      IF(YHOLD.GT.GNSQ(J)) GO TO 31
      YHOLD=GNSQ(J)
   31 CONTINUE
      WRITE(2,88) YHOLD
   88 FORMAT(3X,E14.4)
      RETURN
      END
```

Figure (48) PROGRAMME FOR PLOTTING POWER SPECTRUM.

A.2.11 SUBROUTINE COMPRESSION LAW

It performs the computation of y(t) signal, when the L(t) signal is passed through the integrator, of the linear delta modulator, and also computes the baseband content from the y(t) waveform as shown in the experimental arrangement in figure 8. Input to this subroutine is the L(t) signals stored in the real array Q of dimension 1155, and the output filtered waveform E_{sr} sin $\omega_s t$, in figure 8, stored in the real array ERR of dimension 1155. It also computes the filtered output signal power expressed in dBm. Filtering is achieved by convolving the impulse response stored in the real array B of dimension 64 of the filter with the y(t) signal, stored in the real array cv of dimension 1024. The call statement is

CALL COMPRESS LAW (NUMBER, DBC, B).

where NUMBER = number of signal samples

DBC = filtered output signal power in dBm.

B = Impulse response of the lowpass filter

```
      SUBROUTINE COMPRESS LAW(NUMBER,DBC,B)
      DIMENSION Q(64)
      COMMON WIN(1024),ERR(1155),CV(1024),Q(1155)
      NUMBER=1024
      STEP=0.05
      WINT=-0.015
      DO 122 I=1,NUMBER
      IF(I.EQ.1) GO TO 87
      IF(Q(I).EQ.1) GO TO 86
      CV(I)=WINT-STEP
      GO TO 122
   86 CV(I)=WINT+STEP
      GO TO 122
   87 CV(I)=CV(I-1)+STEP*Q(I)
  122 CONTINUE
      N=64
      NN=NUMBER+N-1
      CALL FILTER(N,B,NUMBER,CV,NN,ERR)
      J=34
      AKEA=0.0
      DO 41 I=1,NUMBER
      ERR(I)=ERR(J-I)
      J=J+2
      AKEAK=AKEAK+ERR(I)*ERR(I)
   41 CONTINUE
      AKEAK/NUMBER
      DBC=10.0*ALOG10(AK/0.5)
      WRITE(2,55) DBC
   55 FORMAT(3X,F10.4)
      RETURN
      END
```

GMENT, LENGTH 160, NAME COMPRESSLAW

Figure(48)a. PROGRAMME FOR COMPUTING COMPRESSION CHARACTERISTICS OF C.F.D.M.

A.3. DESIGN OF NON-RECURSIVE DIGITAL LOW-PASS FILTER

For the band limitation of the input signal and filtering the step approximated signal in the decoder to reconstitute the baseband signal, a low-pass filter is required. For these purposes, a digital low-pass filter, having a relatively sharp cut-off frequency at the highest frequency component in the baseband signal, is designed.

Non-recursive filter can be easily designed to approximate any desired continuous frequency response, and have a linear phase characteristic so that the filter contributes minimum signal distortion. Since the non-recursive filter has finite duration impulse response, its frequency response is specified at equispaced frequencies.

The efficiency, in terms of the number of computational operations required per sampling period, for the non-recursive filter design, is usually much greater than that of the recursive filter of comparable characteristic. However, if the design methods based on F.F.T.(Fast Fourier Transform) are used, the efficiency of a non-recursive filter design could be much improved. Throughout this part of the work implementation methods based on F.F.T. algorithm will be employed.

Several techniques for designing finite duration impulse response digital filter (F.I.R. filters) have been reported[17]. Our interest is in the "Frequency Sampling Technique". Non-recursive realization of filters include, basically, direct convolution and fast convolution. In our design feature, frequency sampling technique and direct convolution method realization of the filter will be used.

A.3.1 FREQUENCY SAMPLING TECHNIQUE

Frequency sampling technique is widely explained in the literature[17], but briefly here it can be described as follows;

Given N samples of impulse response, frequency response can be evaluated at N discrete frequencies by means of D.F.T. (Discrete Fourier Transform).

Thus,

$$H\ell = \sum_{K=0}^{N-1} h_K \, e^{\frac{-j2\pi K\ell}{N}} \quad \ldots\ldots\ldots\ldots \quad (86)$$

$$\ell = 0, 1, 2, \ldots. N-1$$

where $H\ell$ is the frequency response at the frequencies $f = \frac{\ell}{NT}$, with a frequency spacing $\Delta f = \frac{1}{NT}$ between each sample, where T is the clock period. This implies that if N discrete frequency response samples are known, we can always find the impulse response $(h_o, h_1, h_2 \ldots h_{N-1})$ by taking the inverse D.F.T. of the frequency response samples $(H_o, H_1 \ldots\ldots H_{N-1})$

i.e.

$$h_K = \frac{1}{N} \sum_{\ell=0}^{N-1} H_\ell \, e^{\frac{j2\pi K\ell}{N}} \quad \ldots\ldots\ldots\ldots \quad (87)$$

$$K = 0, 1, 2 \ldots. N-1$$

where h_K is the impulse response.

From the above relations, it can be realized that we can always approximate the impulse response from the frequency response, sampled at equispaced frequencies, which becomes the basis of the frequency sampling technique.

Several types of frequency sampling design can be used. Basically there are two types of filter design, depending on whether N, the length of the frequency response samples, is odd or even and the frequency of the first sample of the frequency response. Type I design is for those (N even or odd) whose initial frequency is at zero frequency of the filter, whereas type 2 design is for those whose initial frequency is offset by half a sampling interval.

A.3.2 REQUIREMENT OF A REALIZABLE FILTER

In choosing the desired frequency response samples, care must be taken that the resulting filter is physically realizable and that its impulse response samples are purely real. For the impulse to be purely real, there are several constraints which should be met in selecting the frequency response samples. $(H_o, H_1, H_2 \ldots\ldots H_{N-1})$.

For a physically realizable linear phase filter the required phase condition is,

$$A_{rg}\, H(e^{j2\pi ft}) = -\pi fNT \text{ radians} \qquad \ldots\ldots\ldots\ldots\ldots \qquad (88)$$

Above relation implies that there must be a delay of NT/2 corresponding to half the duration of the impulse response samples. In fact for the design of linear phase filter the phase of the desired frequency response samples can be put to zero, by selecting the frequency response samples purely real. The required delay of NT/2 clock periods could be put later in the impulse response. The method is given in the following section.

A.3.3 METHOD OF ACHIEVING LINEAR PHASE FILTER

In fact for the design of a linear phase filter the phase of the desired frequency response samples can be left unspecified, i.e.(the frequency response samples are all real by setting the imaginary part of it zero, for the sake of design sake.).

As the D.F.T. treats the sampled waveform periodic in N, the length of the samples, it is therefore convenient to represent the negative frequencies below the sampling frequency when describing the frequency response of the filter.

$$\text{i.e.} \quad f = -\frac{Kfs}{N} \quad \text{is equivalent to } f = \frac{(N-K)fs}{N} \qquad \ldots\ldots\ldots \qquad (89)$$

$$\text{where } K = 1, 2 \ldots\ldots (N/2 - 1)$$

Thus we only need to describe the spectrum in the positive frequency range $0 \leq f \leq fs$

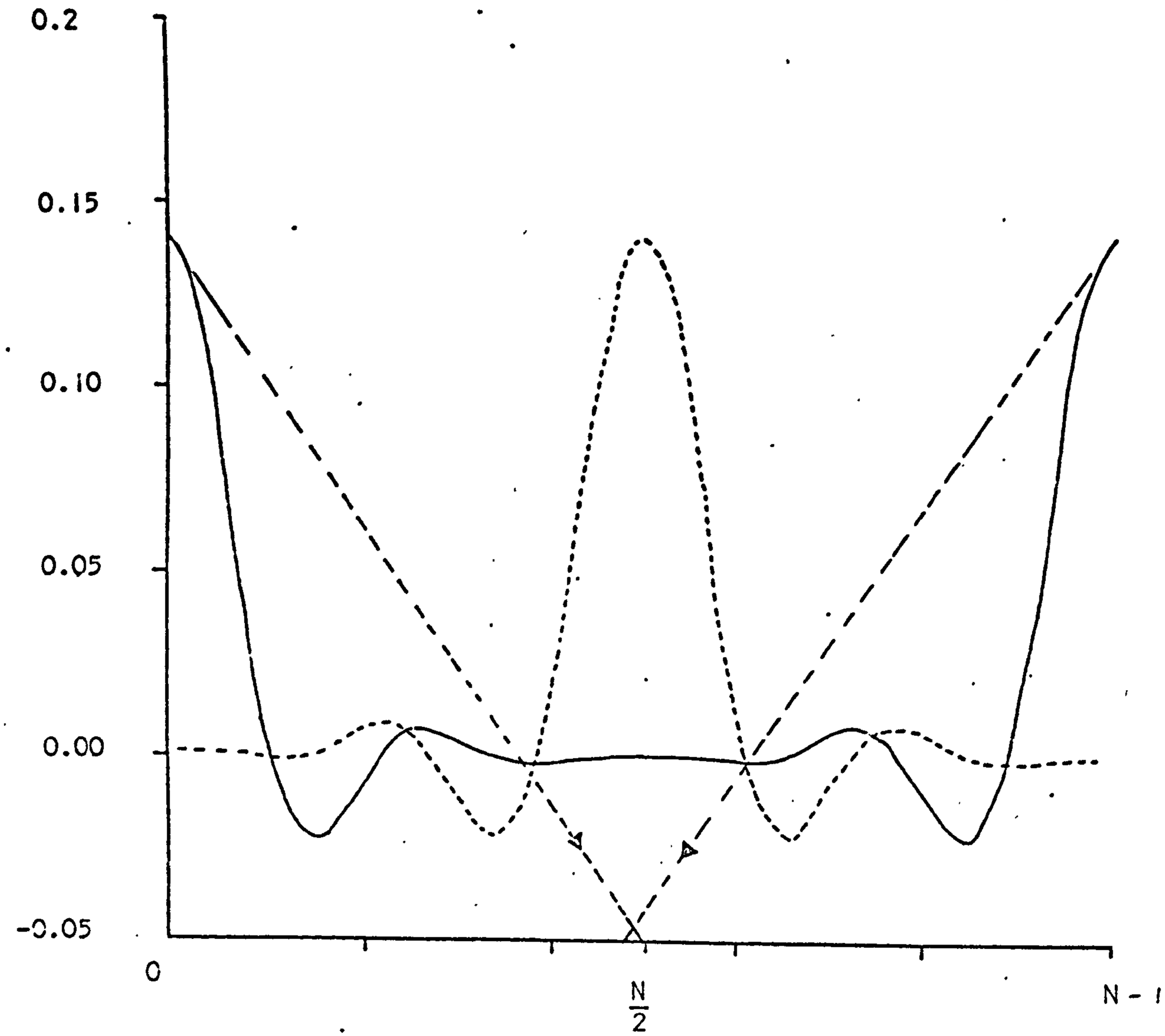

Figure (49) ROTATION OF IMPULSE RESPONSE OF LOW-PASS FILTER

If we take the inverse D.F.T. of this type of frequency of response, the output will be a sequence of impulse response samples existing for positive and negative time, which is not physically realizable. Physically realizable impulse response samples are those whose elements, prior to time t = 0, are all zero. This could be achieved just by rotating the samples by N/2 as shown in Figure (49). By doing this, we are reintroducing the required linear phase characteristic which is equivalent to a delay of N/2 samples or half the duration of impulse response.

A.3.4 PROCEDURE FOR DESIGNING DIGITAL LOW PASS FILTER

A.3.4.1 SELECTION OF FREQUENCY SAMPLES

There are three parameters to be considered for the design problem, namely BW, M and N, where BW is the number of frequencies samples which occur in the passband of the filter, M is the number of frequency samples to describe the frequency characteristic function of the filter in the transition band and N is the number of frequency samples for the filter. Samples which occur in the passband are assigned to have the value of 1.0 and those in the stop band to zero. The transition values are chosen from the optimized values documented[17].

For the frequency sampling technique we used here, the choice of a set of frequencies is merely the choise of the value of N, the length of the frequency response and the initial frequency. In our case the initial frequency is chosen to be at 0.0 Hz.

A.3.4.2 ALGORITHM FOR CALCULATING THE PARAMETERS

For a given length of frequency samples, N, the required number of BW and M that would specify the ideal frequency characteristic, could be calculated with some good approximation. Once N has been chosen, the frequency spacing between samples can be verified as,

$$\Delta f = \frac{fs}{N-1} \qquad \text{.....} \qquad (90)$$

where fs is the clock (or) sampling frequency.

Suppose an approximation is to design a low-pass filter, to have a passband from 0 to fc_2, then the values for BW and M could be calculated as follows.

Let m be the number of frequency spacings that occur in the passband, then,

$$m = \frac{fc_2}{\Delta f} \qquad \text{.....} \qquad (91)$$

Therefore number of samples that occur in the passband can be written as,

$$BW = m + 1 = \frac{fc_2}{\Delta f} + 1 \qquad \text{.....} \qquad (92)$$

The number of M, the transition values, can be chosen as required. From the rough estimation, adding one more transition sample M, for a given N and BW, would reduce the sidelobes about 20 dB.

A.3.4.3 PARAMETERS USED

We are concerned with type I filter design procedure, since D.F.T. using F.F.T. algorithm can only operate for a length of sequence $N = 2^n$, where n is any integer and we are interested in designing a lowpass filter having a passband from zero to fc_2Hz, where fc is the cut off frequency.

For the required filter having a cut off at 3 kHz, the following calculations were made from equations (90) and (91).

Let fs = 40 kHz

N = 64

From equation (90)

$$\text{FREQ. SPACING} = \Delta f = \frac{40.0}{63.0} \text{ kHz}$$

If fc_2 = 3.1 kHz

From equation

$$m = \frac{3.1}{\Delta f} = 4$$

∴ From equation (92)

BW = m + 1 = 5

The choice of N is made relatively low, i.e. N = 64 to use direct convolution method, for the filtering in time domain. Therefore the parameters are N = 64, BW = 5, and M = 3. Selection of values of M is just for increasing ratio of main lobe to sidelobe ratio of the filter characteristic.

A.3.4.4 IMPULSE RESPONSE OF THE FILTER

The frequency samples for all the values of BW were set to 1.0, the transitions values were set according to the table[17], and the values for the stop band were set at 0.0. To form an even frequency characteristic function, an odd number of BW is taken by making the sample value of f_K the complex conjugate of the value at f_{N-K} except for the sample at f = 0.

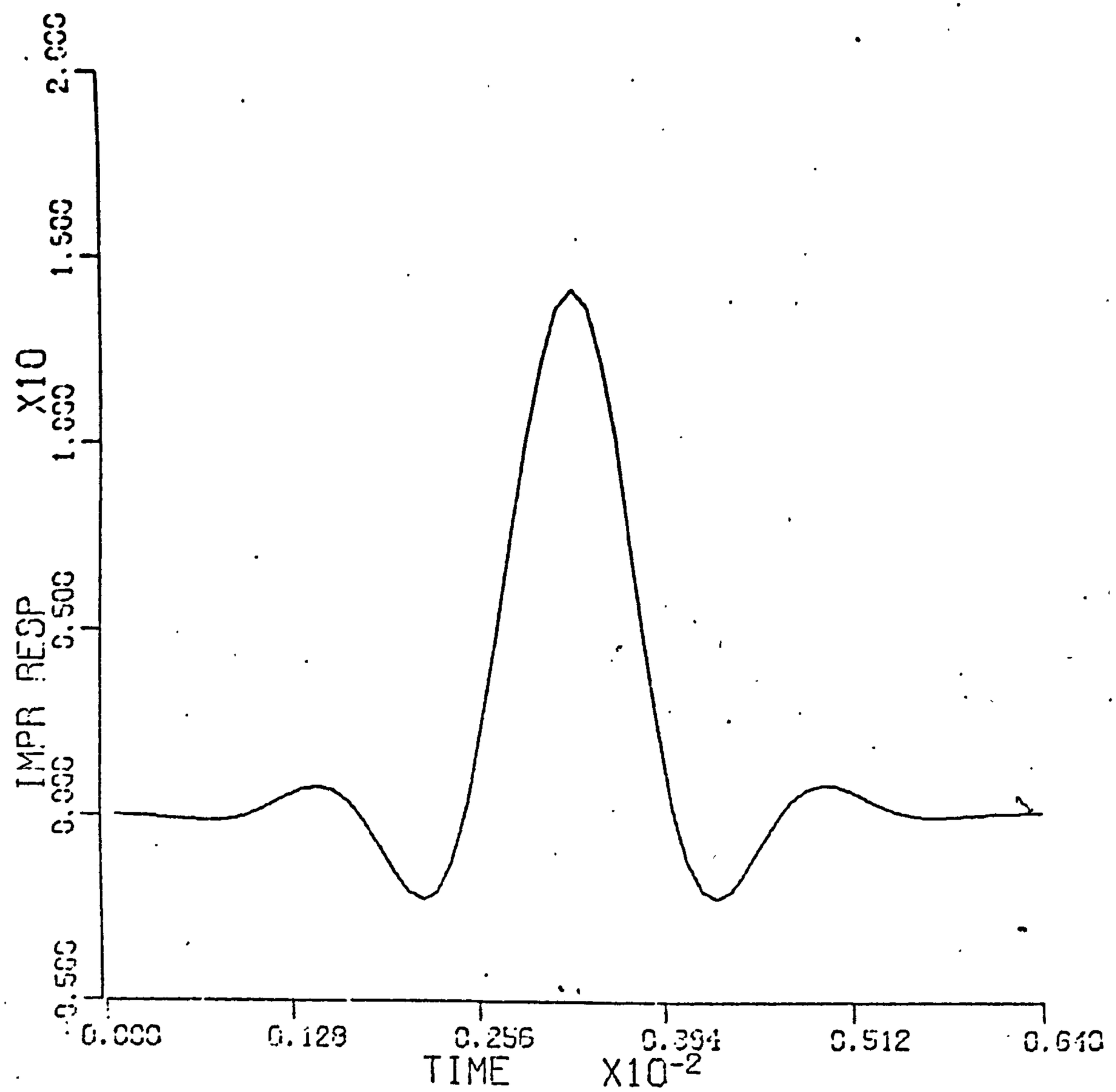

Figure (50) SHIFTED IMPULSE RESPONSE OF THE FILTER.

Having the sequence of frequency samples, they were transformed to complex numbers having their complex terms zero, to operate with D.F.T. Then the D.F.T. of this sequence was taken and the real part of the D.F.T. output was shifted back by N/2 samples, as shown in Figure (49), so that the signal impulse response achieved has a delay of NT/2 with its centre at N/2. (Figure (5)).

A.3.5 FILTERING.

Then the input signal samples are convolved with this shifted impulse response sample in time domain to yield the filtered output. The impulse response approximated from the real discrete, frequency response samples and the 16:1 interpolated frequency response of the desired filter is shown in Figure (15).