LOW BIT RATE SPEECH COMMUNICATION BASED ON CHARGE COUPLED DEVICE FOURIER TRANSFORM PROCESSORS

A thesis submitted to the Faculty of Science of the University of Edinburgh, for the degree of Doctor of Philosophy

by

M. C. DAVIE, B.Sc.

Department of Electrical Engineering Sept 1979

DECLARATION OF ORIGINALITY

This thesis, composed entirely by myself, reports on work conducted by myself in the Department of Electrical Engineering, University of Edinburgh, and the Advanced Development Division, Racal Group Services, Reading.

ACKNOWLEDGEMENTS

The author would like to express his sincere gratitude to Dr.P.M.Grant, Mr.R.J.Preston, Dr.M.A.Jack, Dr.J.M.Hannah and Professor J.H.Collins for their supervision and encouragement given throughout this work. He would like to acknowledge Racal Group Services Ltd., Reading and the Science Research Council for financial assistance.

Sincere thanks are also due to the many friends and members of staff, both in the Department and at Racal for their useful comments relating to this work. The advice and help given by Mr.J.N.Holmes and his colleagues at the Joint Speech Research Unit, Cheltenham are gratefully acknowledged. Finally, thanks are due to Mrs.M.C.Davie for her patience and help.

CONTENTS

GLOSSARY OF ABBREVIATIONS

agc	automatic gain control
ARAM	Analogue Random Access Memory
BBD	Bucket Brigade Device
bps	bits per second
CCD	Charge Coupled Device
CMOS	Complementary Metal Oxide Silicon
CTD	Charge Transfer Device
CZT	Chirp Z-transform
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DSAT	Double Sample Alternate Tap
FFT	Fast Fourier Transform
FILO	First In Last Out
FIR	Finite Impulse Response
FM	Frequency Modulation
FT	Fourier Transform
FTP	Fourier Transform Processor
IDFT	Inverse Discrete Fourier Transform
IF	Intermediate Frequency
IFT	Inverse Fourier Transform
LSB	Least Significant Bit
MDAC	Multiplying Digital to Analogue Convertor
MOS	Metal Oxide Semiconductor
N/S	Noise to Signal Ratio
PCM	Pulse Code Modulation
PE/N	Peak Error to Noise ratio

PT	Prime Transform
RAM	Random Access Memory
rms	root mean square
ROM	Read Only Memory
SCZT	Sliding Chirp Z-transform
SIPO	Serial-In-Parallel-Out
TF	Transversal Filter
TRF	Tuned Radio Frequency
TTL	Transistor-Transistor Logic
VLSI	Very Large Scale Integration
V/UV	Voiced/Unvoiced decision

To Moira

CHAPTER 1

INTRODUCTION

1.1 ADVANCED ANALOGUE SIGNAL PROCESSING

Low bit rate speech communication systems (vocoders) have been available for many years now, but their application areas have always remained extremely specialised. One of the main reasons for this trend has been the large size and expense associated with such equipments, especially when the synthetic speech quality achieved is rather poor. Within the last few years, however, vocoder interest has been rekindled and stimulated by the demand for digital communications. Efficient data compression techniques are necessary since the increased bandwidth inherent in digital coding is contrary to the overriding philosophy that bandwith conservation is requisite [1].

At present, the two established systems for low bit rate (2400bps) speech compression are the channel vocoder [2] and the linear predictive vocoder [3]. The linear predictive vocoder is becoming increasingly popular because of its more elegant digital implementation. However, recent advances in analogue signal processing may yet produce a channel vocoder which is even more attractive in terms of engineering premiums.

One technology which offers high density analogue signal processing is the Charge Coupled Device (CCD) [4]. The CCD is a shift register which stores samples of analogue information directly as charge packets. These charge packets can be transferred along the register under the control of an external clock to form a variable delay line structure. Further, by adding a non-destructive tapping technique, CCD permits fabrication of very compact Transversal Filters (TFs) [5] which form the basic building blocks for many powerful signal processing modules [6].

Since the conception of CCD at the turn of the last decade, industry's acceptance of CCD for analogue signal processing has been laboured. This has been due to several factors which include (a) the increased competition from digital componentry (b) the limited CCD operating characteristics and (c) the slow development of CCD peripheral integration. However, the CCD is now being manufactured as a fully modular 'black box' which frees the consumer from many of the 'setting-up' problems, and advanced components such as Fourier Transform Processors (FTPs) [6], adaptive filters [7] and correlators [8] are finding extensive real-time application.

The intention of this thesis is to examine and demonstrate the potential of analogue CCD in the realisation of complex signal processing systems. In particular, the application of CCD FTPs in a low bit rate channel vocoder is investigated, since the market for a low power, small size and low cost vocoder is potentially vast.

1.2 LAYOUT OF THESIS

Chapter 2 gives an introduction to the subject of speech and vocoders, highlighting the particular areas in which CCD may have application. The channel vocoder is described in detail since this algorithm is examined in later chapters. Some of the basic CCD principles are explained in chapter 3, along with a summary of the most important operational characteristics. The transversal filter, one of the most powerful analogue signal processing blocks, is introduced in this chapter.

Chapter 4 investigates several of the many algorithms which have been proposed for real-time Fourier transformation and, in particular, compares the advantages and disadvantages of the CCD Chirp-Z Transform (CZT) and the CCD Prime Transform (PT). The detailed design and construction of a CCD CZT processor together with the performance limitations are presented in chapter 5. Practical aspects of CCD system design are emphasised.

In chapter 6, an extensive computer simulation of a novel channel vocoder is reported. Both the analyser and the synthesiser are based upon discrete Fourier transform processors. The simulation involved the design and construction of a specialised 'intelligent' computer terminal and this is described briefly. Practical experience in CCD processors and the results of the above simulation are merged in chapter 7 to suggest an optimal hardware configuration for a CCD channel vocoder.

Finally, the most important achievements of this work are summarised in chapter 8, and the conclusions suggest suitable areas for research continuation.

CHAPTER 2

SPEECH AND VOCODERS

The advantages of digital communication are well known. For example, binary waveforms may be regenerated at stages along the transmission path without cumulative addition of noise and distortion. Also, the user is free to "scramble" the message in complex ways for secure or private transmission. The price paid for these important advantages is additional transmission bandwidth. In order to transmit a speech signal with 3kHz bandwidth and 40dB signal to noise ratio using direct Pulse Code Modulation (PCM), a data rate of approximately 64000 bits per second (bps) is required. A normal 3kHz analogue communication channel will handle only 2400 bps without equalisation.

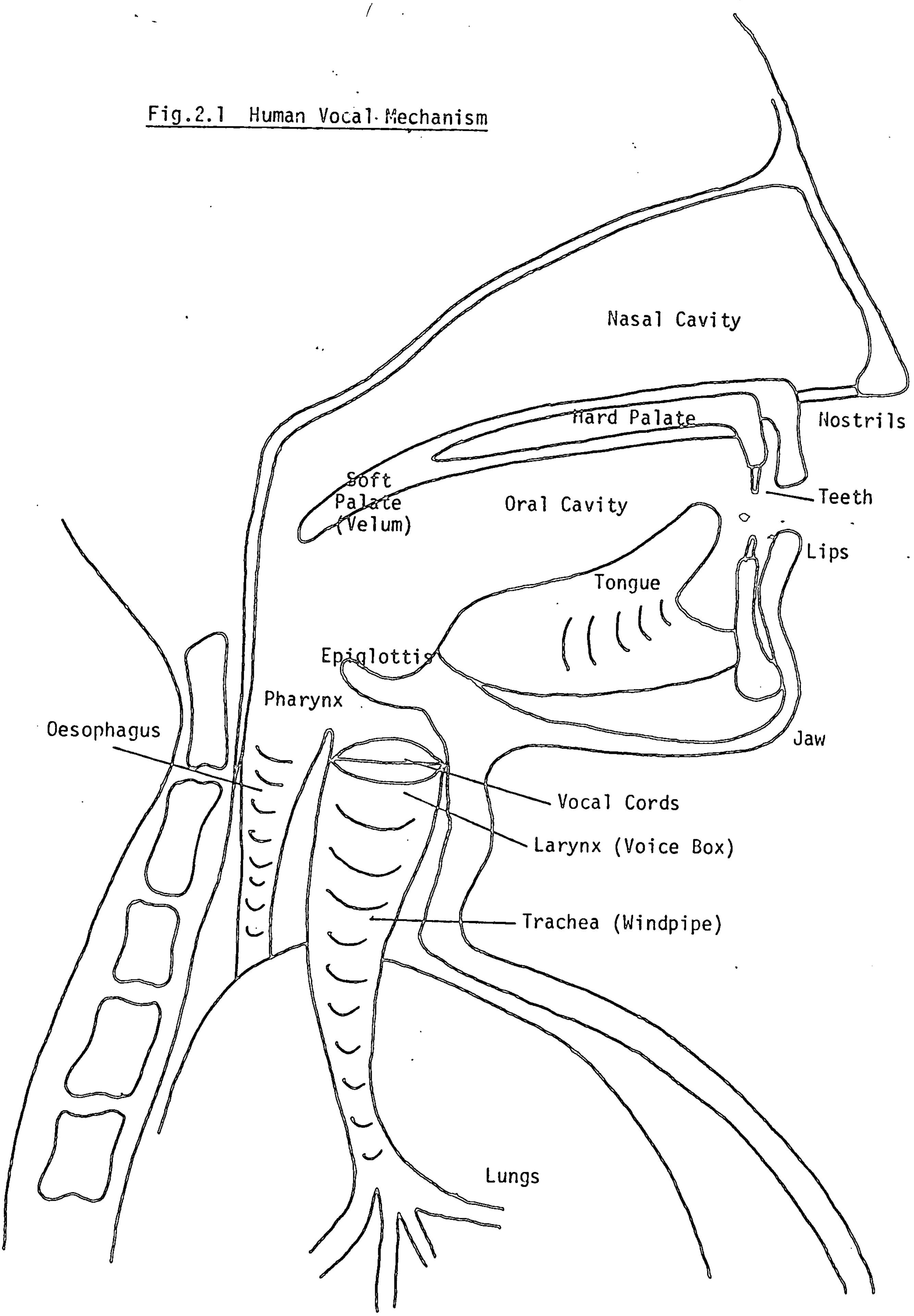
The solution is therefore to find an efficient coding algorithm for speech, which permits more economical use of the spectrum. If one examines the information content of speech, assuming a vocabulary of 2^{14} words and a speaking rate of 10 words per second, only 140 bps are required. Of course, this figure does not include other information such as emotional content and speaker characteristic. Nevertheless, it is clear that the analogue speech signal contains considerable redundancy because the human vocal mechanism generates sounds by relatively slow articulatory movements. A system which attempts to exploit this redundancy is called a vocoder (short for voice coder). In general, a vocoder operates by analysing speech in the transmitter, generating a set of control parameters at a much lower bit rate and synthesising the original speech in the receiver. The amount of information which can be thrown away depends on the application: military systems might require only intelligibility whereas telephone systems demand high quality.

Section 2.1 reviews briefly the most important characteristics of the human speech mechanism. This is necessary since any vocoder algorithm must capitalise on various aspects of speech production; the optimum vocoder will model the human mechanism exactly. Section 2.2 examines pitch detection algorithms which are vital for good quality vocoding, and sections 2.3 through 2.5 summarise some of the most important, established low-bit rate vocoder techniques. Other speech coding algorithms such as adaptive delta modulation [9] and time encoded digital speech [10], which are generally considered to have application in higher bit-rate communication channels (i.e. 4800 - 16000bps), are not considered here.

2.1 HUMAN SPEECH PRODUCTION [11,12]

A cross-section through the human vocal mechanism is shown schematically in Fig.2.1. The main vocal tract is a non-uniform acoustical tube which starts at the pharynx and terminates at the lips. It can be deformed by four

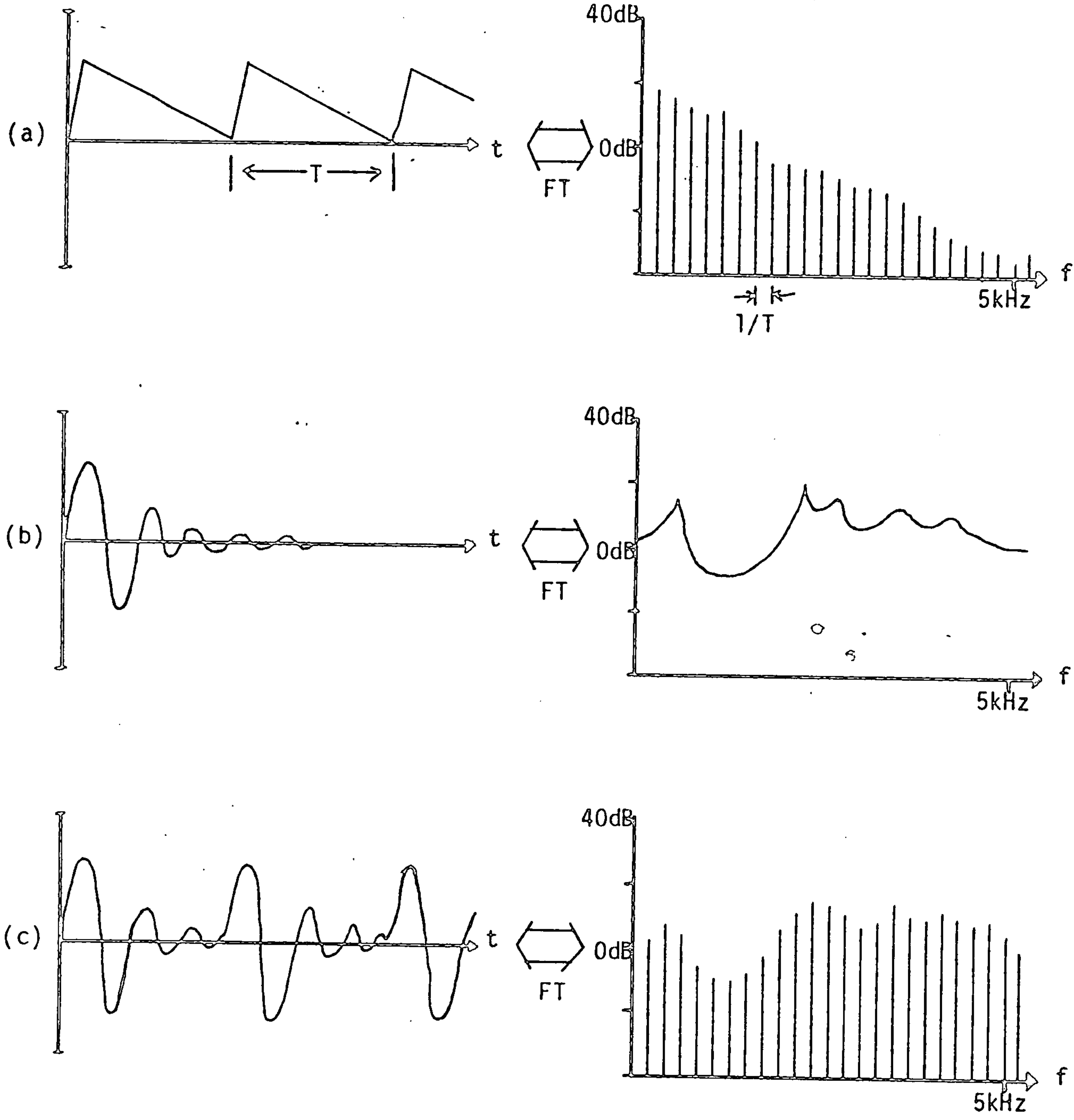
Fig.2.1 Human Vocal Mechanism



articulators, namely the lips, the tongue, the jaw and the velum, to provide the resonances and anti-resonances (poles and zeroes) which modify the energy/frequency distribution of the excitation source. The resonances of the vocal tract are normally known as the "formants" of speech. An ancillary path for sound transmission is formed by the nasal tract, extending from the velum to the nostrils, and has essentially fixed characteristics.

The excitation for the vocal tract is a controlled flow of air from the lungs which first passes through the larynx or voice-box. The larynx has a cartilage frame and houses two lips of ligament and muscle called the vocal cords.

When "voiced" speech is produced, the vocal cords are held tensioned by cartilages and the Bernoulli effect makes the slit between the cords (the glottis) open and close at a rate determined by both the sub-glottal pressure and the cord tension. The quasi-periodic pulses of air have a triangular shape and the repetition rate, which is closely related to the perceived "pitch" of the speech, lies between 50 and 400 times per second. Because of the triangular shape, the vocal cord excitation source has a line spectrum which falls off at approximately -12dB/octave (Fig.2.2a). Voiced speech includes the vowels and several consonants such as /l,r,m,n/ and a typical speech segment is shown in Fig.2.2c.



(a) excitation source

(b) vocal tract impulse response and transfer function

(c) speech waveform (a) * (b)

Fig.2.2 Typical Voiced Speech Segment (vowel /i/)

The second main excitation source is random noise and is used in the production of "unvoiced" speech. In this case the vocal cords are held wide apart and the air passes uninterrupted through the larynx. A subsequent stricture in the tract causes turbulent air flow creating acoustic random noise. An example of unvoiced speech is the fricative consonant /f/ which is produced by a labio-dental stricture (upper teeth on lower lip). Another group of speech sounds, call "voiced fricatives" (e.g. /z,v/), uses both the turbulent and the vocal cord sources simultaneously.

The final type of excitation results from the build- up of pressure that occurs when the vocal tract is completely closed at some point. A sudden release of this pressure causes a transient excitation of the vocal tract. If the vocal cords were vibrating immediately before the closure, the sound is called a "voiced stop consonant" (e.g. /b,d/) and if the closure is preceded by silence, an "unvoiced stop consonant" is produced (e.g. /p,t/).

Speech production represents only half of the human communication system; the acoustic signal has to be received by the ear and decoded into the appropriate neural stimuli. Several aspects of the receiving process give an insight into speech waveform redundancy. The acoustic pressure wave received by the ear is converted into a mechanical vibration by the tympanic membrane (eardrum) in the middle ear. This vibration is amplified and transmitted to the cochlea in the inner ear by a system of levers. The fluid filled cochlea is partitioned by the basilar membrane which tapers in width from base to apex. Because the cochlea is a rigid structure, the input vibrations pump the cochlea fluid back and forward and the basilar membrane vibrates at a position dependent on the frequency. The organ of Corti, which rests along the length of the basilar membrane and contains some 30,000 sensory cells, detects and converts the vibration into electrical pulses to be fed in parallel along the auditory nerve to the brain. How the brain decodes these pulses is still unknown. However, it is clear that the ear effectively performs a spectrum analysis and passes spectral amplitude information to the brain. Both the frequency and amplitude are logarithmically resolved. In addition, it has been shown that the relative phase of the input signal does not affect human hearing[11].

2.2 PITCH DETECTION

In almost all speech synthesisers, use is made of the fact that, to a first approximation, the excitation source and the vocal tract may be treated independently; electrical models of speech production consist of a set of waveform generators feeding a filter bank which represents the vocal tract (Fig.2.3). The excitation source in Fig.2.3 does not cater for voiced fricatives and stops, because in general,

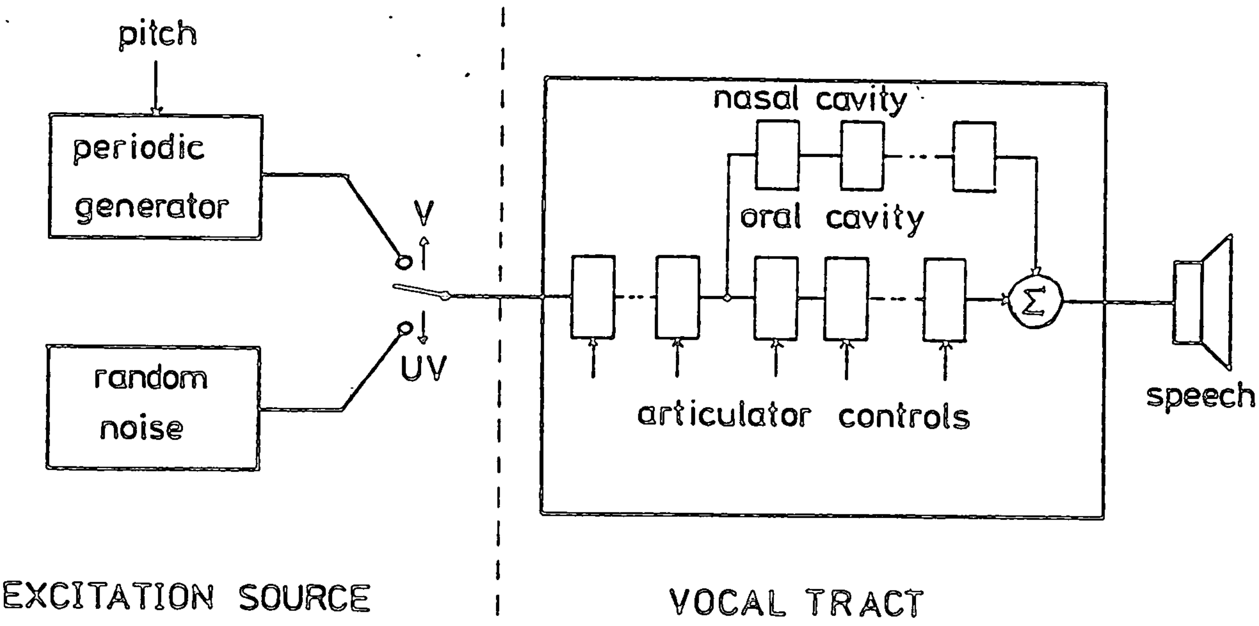


Fig.2.3 Electrical Model for Speech Synthesis

the extra complexity does not provide a significant speech quality improvement. Recognition of these sounds is left to human perception.

Two control parameters are required for the synthesiser excitation source: (a) a voiced/unvoiced (V/UV) input to select the appropriate waveform generator and (b) a number representing the pitch period for the voiced generator. In a vocoder, the analyser has to estimate automatically these parameters. The V/UV decision can either be computed separately or can be a by-product of the pitch detection. The most common V/UV detector compares the speech energy in two frequency bands. For example, a typical system might compare the energy in the band 200-600Hz with that in the range 5000-7000Hz [13]. A high ratio of low frequency to high frequency indicates voiced sound and a low ratio (<<1) indicates unvoiced sound.

At first glance, it would appear that the accurate and reliable measurement of pitch period is relatively straightforward. However, this is not the case for several reasons and, in fact, turns out to be one of the most difficult vocoder operations. In principle, the fundamental frequency may be extracted by a low-pass filter. In practice, the fundamental frequency varies over 3 or 4 octaves so that a fixed low-pass filter often extracts more than the fundamental. Also, in many practical situations, the fundamental frequency is not present or is greatly attenuated (e.g. 300-3000Hz telephone channel). Another reason for pitch detector inaccuracy is that the glottal excitation waveform is not a perfect train of periodic pulses. This results in a speech waveform varying both in period and in the detailed structure within a period.

Over the years, considerable effort has been invested in the search for a reliable pitch detector. The reason for this quest is that the intonation and intelligibility of synthetic speech depends to a large extent on the correct pitch. Pitch detector errors cause very objectionable effects. If, for example, the pitch detector selects the second harmonic instead of the fundamental, the resultant "squeak" not only sounds unnatural but causes the listener to lose concentration temporarily, thereby masking a longer section of the speech.

Pitch extraction techniques may be classed in two main categories: (a) time domain and (b) frequency domain. The most common time domain pitch detector in an analogue implementation is a tracking band-pass filter. This attempts to follow the fundamental frequency by assuming that the fundamental component has the largest amplitude. If the input speech is high quality (wide-band) with a good signal to noise ratio, then reasonable performance can be maintained over a limited frequency range. However, because of the filter response time, fast pitch inflexions may temporarily unlock the filter.

Another pitch detector makes use of parallel processing in the time domain [15] and relies on the philosophy that one simple measurement is unlikely to be satisfactory but, by combining the results of several measurements performed in parallel and taking a majority vote, a reliable answer is obtained. The pitch detector in Ref.[15] low-pass filters the input speech to 900Hz and makes six parallel estimates of the pitch, based on peak and valley measurements.

In recent years, autocorrelation pitch detectors have become popular in digital systems. There are many variations on this basic theme, but perhaps the most successful is autocorrelation of centre clipped speech [16]. The centre clipping tends to remove the formant structure of the speech and effectively flattens the spectrum. It is then much easier to resolve the autocorrelation peak due to pitch (i.e. autocorrelation peaks due to the formants are suppressed). The pitch period is measured as the time lag from a reference to the largest autocorrelation peak and, normally, logic circuits after the correlator check the validity of the measurement (Fig.2.4).

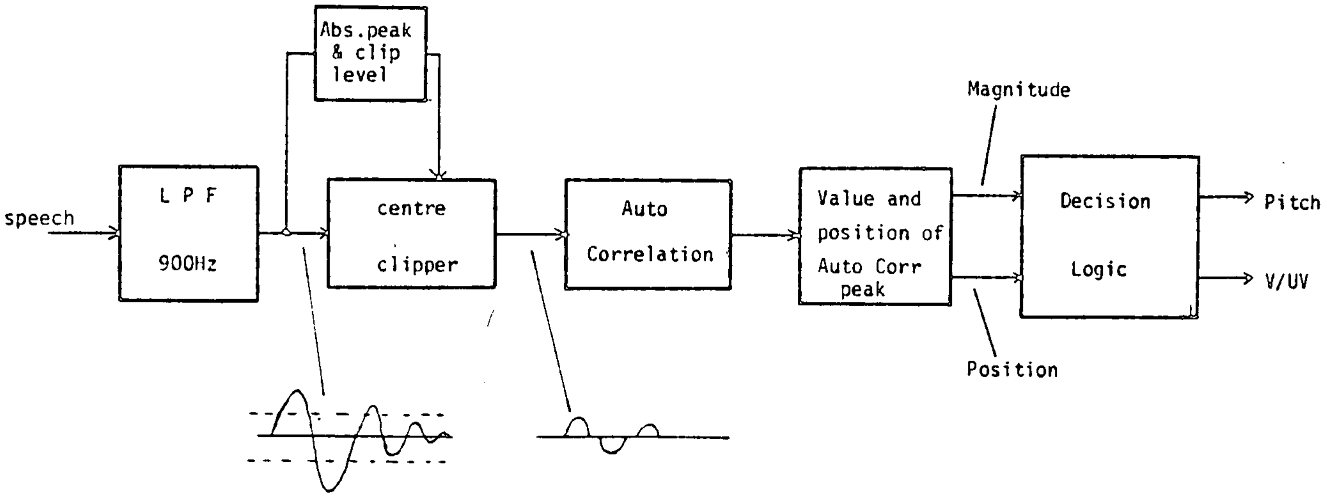


Fig.2.4 Centre Clipped Auto-correlation Pitch Detector

A hardware implementation of the above approach has been reported [17] which uses both centre clipping and infinite clipping to ease computational problems. Another approach computes the average magnitude difference function [18] instead of the autocorrelation function so that multiplications are replaced by additions.

The final time domain technique of significance makes use of adaptive filtering. In one example, the coefficients of a digital filter are updated to minimise the mean square error function and the resulting residue approximates to the glottal pulse train [19]. One advantage of this technique is that in a linear predictive vocoder (section 2.4) the filter structure already exists and the pitch period is therefore a by-product. A slightly different technique based on adaptive principles employs a recursive comb filter which homes in on the speech line spectrum by minimising the mean square output of the filter [20]. Since this method involves only addition, it is faster than the inverse filtering method.

A frequency domain pitch extractor has been described by Noll [21] and is based on the cepstrum of speech, which is defined as the Fourier transform of the logarithm of the power spectrum. The non-linear operation on the spectrum equalises the line harmonic amplitudes and effectively de-emphasises the formant structure. Mathematically, the log operation deconvolves the effect of the vocal tract and the excitation source. The second transform measures periodicity in the frequency domain. Fig.2.5 illustrates the operations involved in the cepstral computation. The time domain signal in Fig.2.5a is transformed to give the line spectrum in Fig.2.5b which is subsequently logged (Fig.2.5c) and Fourier transformed a second time to provide the cepstrum (Fig.2.5d). (The cepstrum variable is called

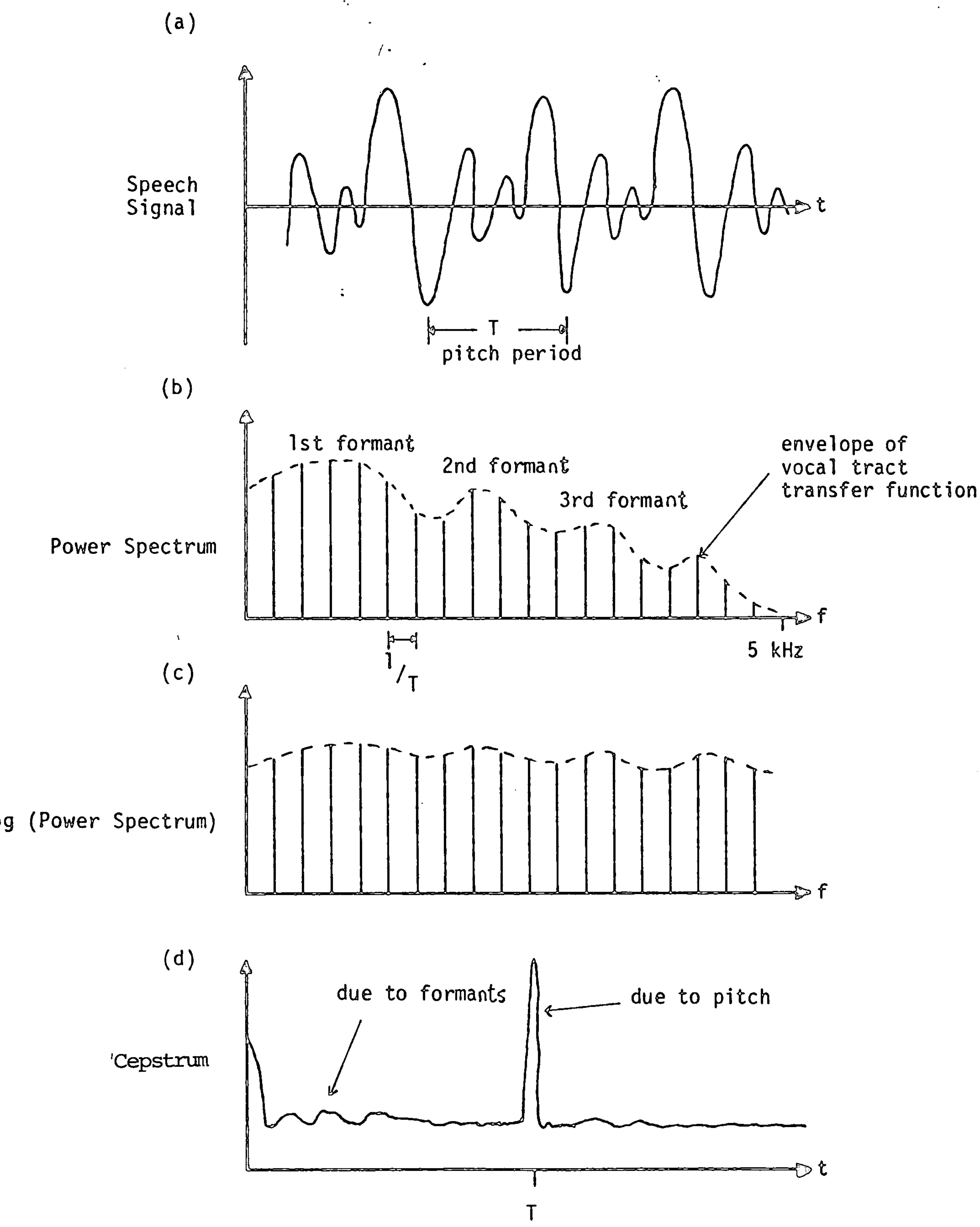


Fig.2.5 Illustration of the Cepstrum

"quefrency" and has units of time). The cepstrum has a main peak due to the speech periodicity and a series of smaller peaks at high quefrencies due to the compressed formants. When the cepstrum is computed for unvoiced speech, there is no line spectrum and therefore no cepstral pitch peak, so that an automatic V/UV decision can be made. As in the autocorrelation technique, logic circuits are usually necessary to decide if the pitch measurement is feasible. It is generally considered that the cepstrum is an extremely powerful technique because the input signal does not require to be high quality.

A comparative performance study of several of the above pitch detectors was carried out by Rabiner et. al.[22]. The conclusion was that each detector has its own strengths and weaknesses and no single detector was top ranked for all cases of input signal. For example, the cepstrum was poor on high pitch speakers whereas time domain techniques were poor on low pitch speakers. Overall, the cepstrum technique proved to be the best all-round performer but was probably the most inefficient in terms of hardware implementation.

2.3 THE CHANNEL VOCODER

Historically, the channel vocoder was invented in 1939 by Homer Dudley [23] of Bell Telephone Laboratories. During the 1940's and 1950's it was realised that vocoders might have a useful role in communication systems and by the late 1950's, practical systems were being developed. The basic principles described by Dudley in 1939 remain today as an established speech compression technique.

A block diagram of the channel vocoder is shown in Fig.2.6. In the analyser, the main processing block is a bank of contiguous band-pass filters arranged to cover continuously the speech bandwidth of interest. The outputs from these filters are rectified and low-pass filtered so that an approximation to the short-time spectral envelope [11] of the speech is available. Normally, the amplitude components of the smoothed spectral envelope are sampled, quantised logarithmically, multiplexed with pitch and voicing information (section 2.2) into frames and transmitted serially to the synthesiser. Data reduction is achieved because:

1. phase information is not transmitted

2. only the smoothed envelope of the voiced speech line spectrum is transmitted (in addition to the line spectrum fundamental)

3. both amplitude and frequency are logarithmically quantised.

In the synthesiser, the received data are inversly decoded and fed in parallel to the appropriate circuit

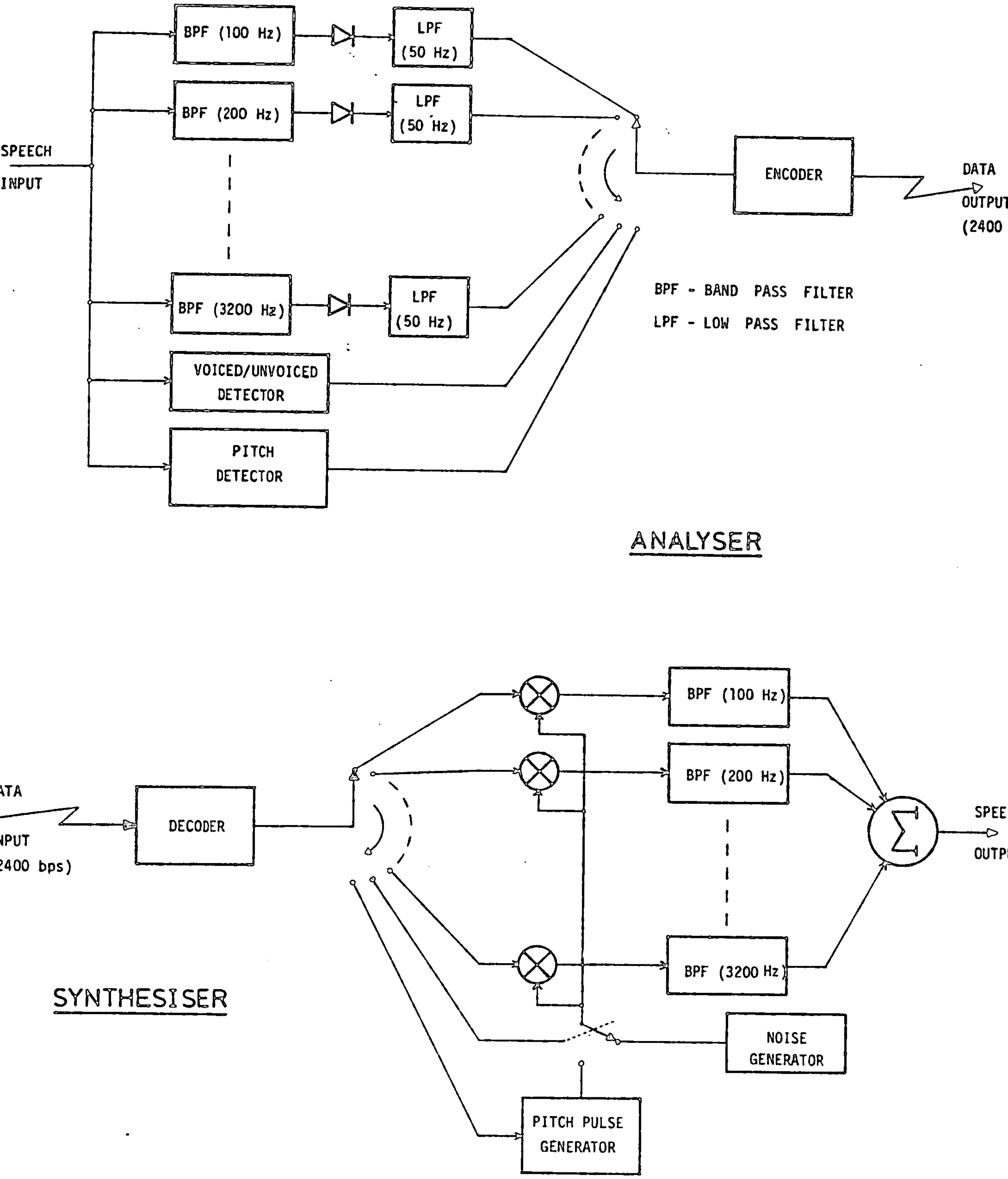


Fig.2.6 The Channel Vocoder

elements. Speech is synthesised by summing the outputs from a contiguous filter bank, similar to that in the analyser, which has been input with weighted versions of an excitation source. The particular source is selected by the voicing control and the period of the periodic source is given by the received pitch information. Finally, the synthetic speech is filtered to remove the analyser pre-equalisation and to compensate for the non-triangular excitation source.

It is generally accepted that the minimum data rate which can be achieved by a channel vocoder (without special coding techniques [24]) is in the order of 2400 bps. At this data rate, the speech has a mechanical quality but still maintains good intelligibility [2].

The channel vocoder fidelity depends on the design of the contiguous filter banks [2]. Ideally, these should consist of steep-sided filters narrow enough for no more than a single harmonic to enter any one filter during voicing. The spectrum envelope generated would thus be a correct measure of the speech spectral energy at that time and synthetic speech could be constructed exactly from this data. The disadvantage of this filter bank, disregarding practical considerations, would be that relatively little bandwidth compression would result. For example, if the lower limit of the pitch frequency is 50Hz then 80 parallel filters are required to analyse a 4kHz bandwidth. As the filter bandwidths are increased so that the total number of filters can be reduced, a spectral distortion becomes evident. This is because more than one harmonic appears in some of the filters at low pitch frequencies. It is exceptionally difficult to quantify this distortion in terms of synthetic speech quality and generally, the number of filters in the bank is chosen through practical experience. Channel vocoder designs typically employ between 16 and 32 logarithmically spaced filters to cover a 4kHz bandwidth. The filter characteristics for a 19-channel vocoder are given in Table 2.1 [25].

The individual filter characteristic is also an important consideration. It can be deduced that sharp cut-off filters will give a more accurate spectral measurement; in practice, sharp cut-off filters have long settling times so that their use would result in a smearing of rapid spectral changes and subsequent reverberation effects in the synthetic speech. Unequal filter time delays, as might be the case if different channels employ different bandwidths, also give rise to a temporal smearing effect. The compromise utilised by most channel vocoders is a 2-pole Butterworth characteristic [2].

The cut-off frequencies of the low-pass filters which smooth or average the rectified band-pass filter outputs have to be chosen to follow the slowly varying spectral content of speech. Practical experience [26] has shown that

Channel Number	Filter Centre Freq. (Hz)	Analysis Filter BW (Hz)	Synthesis Filter BW (Hz)
1	240	120	40
2	360	120	40
3	480	120	40
4	600	120	40
5	720	120	40
6	840	120	40
7	1000	150	40
8	1150	150	40
9	1300	150	40
10	1450	150	40
11	1600	150	40
12	1800	200	60
13	2000	200	60
14	2200	200	60
15	2400	200	60
16	2700	300	60
17	3000	300	60
18	3300	300	60
19	3760	500	(f_o 3600)60
19a	-	-	(f_o 3750)500

Analysis Filters are second order Butterworth

Synthesis Filters are single tuned with alternate outputs summed in antipha

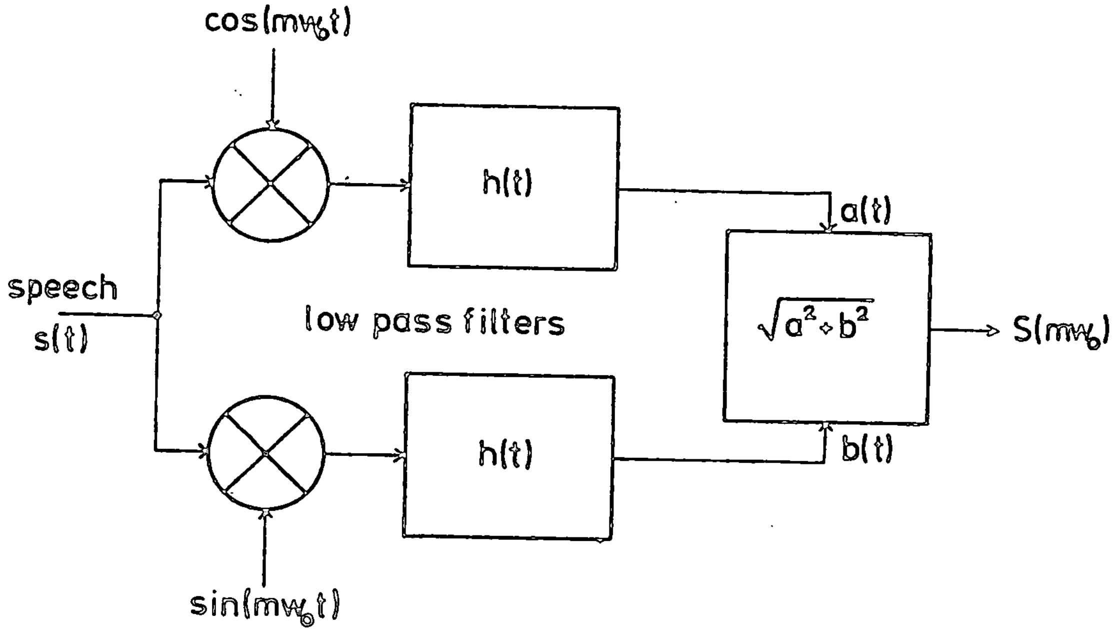
Note: (a) Only 19 analysis channels
(b) Synthesis filter 19 excited during voiced sounds
(c) Synthesis filter 19a excited during unvoiced sounds

Table 2.1 Parameters for a 19 Channel Vocoder

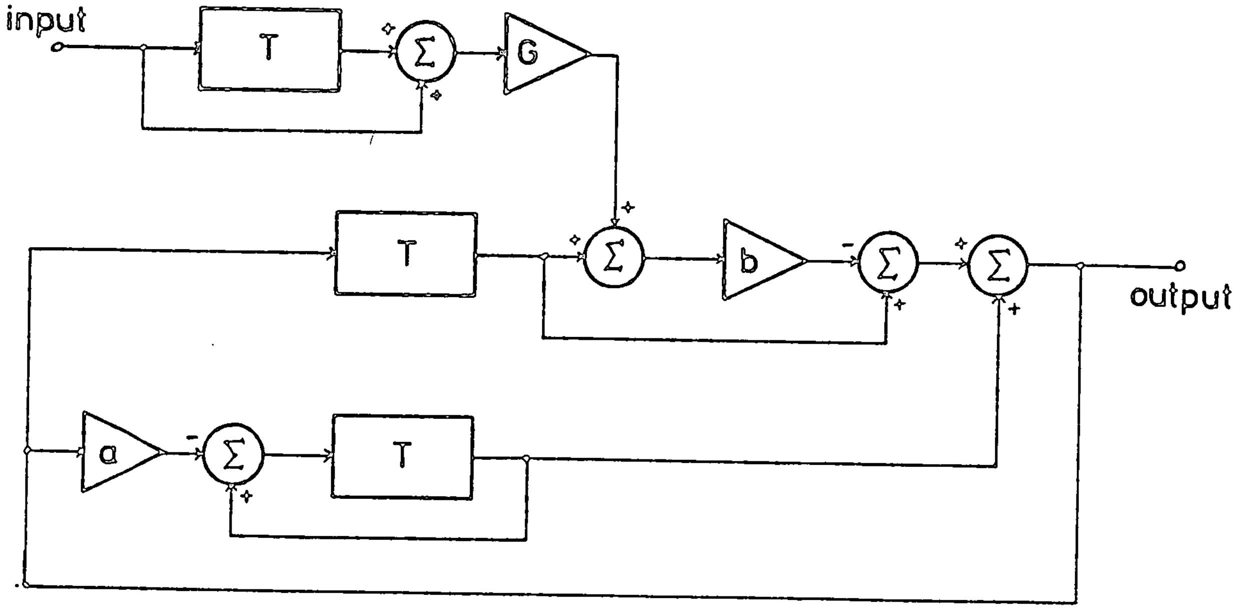
the spectral content of speech is fairly constant for periods of 20mS but has probably changed significantly after 40mS. The smoothing filter is usually chosen to have a 3dB attenuation at 25-35Hz and an 18dB/octave roll-off [2].

As in other speech processing systems, the large dynamic range (> 60dB) associated with speaker variations and conditions causes practical circuit problems in vocoder implementations. Several techniques are used to ease the situation. Pre-equalisation in the analyser is designed to boost the high frequencies so that the spectral energy is spread more evenly between the channel filters. This boost is typically 6dB/octave from 1kHz [25]. (The vocal cord excitation source has a general trend of approximately -12dB/octave which is differentiated when the acoustic pressure wave is launched from the human lips [11] so that the input speech to a vocoder has a general trend of -6dB/octave). Automatic gain control (agc) may be applied to save up to 20dB dynamic range [27], but this practice is not generally desirable since agc distorts the speech and the vocoder is a non-linear system.

More recently, fully digital implementations of the channel vocoder have been reported. In one example [28], use is made of the filtering structure in Fig.2.7a. Speech is simultaneously modulated by two quadrature sine waves of the same frequency and the resultant waveforms are separately low-pass filtered. The modulus of the quadrature



(a) Heterodyne Analyser



(b) 2nd Order Recursive Filter

Fig.2.7 Digital Filtering for the Channel Vocoder

channels gives the spectral amplitude of the speech input evaluated at the frequency of the modulating sinewaves. By sequentially filtering the same segment of speech using different modulating frequencies, an equivalent to the channel vocoder filter bank is achieved. In digital hardware, the modulating frequencies are stored in read only memory and the low-pass filters are accumulate and dump algorithms.

An alternative digital channel vocoder [29] operates by multiplexing a recursive band-pass filter. The filter shown in Fig.2.7b has a Z transfer function given by

$$H(z) = \frac{-G\,b\,(1 - z^{2})}{1 - (2 - a - b)\,z^{-1} + (1 - b)\,z^{-2}} \qquad \ldots (2.1)$$

which has a centre frequency and a Q-factor approximately proportional to $\sqrt{a}$ and $\sqrt{a/b}$ respectively. Using a filter of this type it is relatively straightforward to update the filter coefficients (stored in read only memory) and produce a logarithmically spaced filter bank.

Both of these digital filtering techniques have potential for Very Large Scale Integration (VLSI) and must be considered serious contenders for the single chip vocoder. However, at the present time, typical digital vocoder implementations employ in the region of 200 discrete integrated circuits and consume 15W of power[28].

One other channel vocoder implementation has recently been reported [30]. Here the filter bank is fabricated as a single integrated circuit using 19 parallel finite impulse response (FIR) CCD filters. This approach to the CCD channel vocoder will be discussed in more detail in chapter 7 and compared to the alternative CCD channel vocoder studied by the author.

2.4 THE LINEAR PREDICTIVE VOCODER

An increasingly popular technique for low bit rate speech analysis and synthesis employs the properties of linear prediction [31]. This method is suited to sampled-data implementation and utilises either an all-pole recursive filter [3] or a lattice filter [32] to synthesise the speech. The filter coefficients represent a linear prediction of the analyser input speech.

Briefly, linear prediction consists of predicting or estimating the present value of a signal using a linear weighted sum of delayed signals. The linear weights that minimise, for example, the least mean square predictor error are then information parameters which characterise the properties of the signal under analysis. These linear weights or predictor values can be transmitted directly or can be further processed in a variety of ways for different applications [3].

A useful by-product of the prediction process is that once the weights have converged, the error signal is an approximation to the excitation source. Pitch information and a V/UV decision can therefore be derived by means of a peak picking algorithm.

Speech synthesis from a recursive digital filter using linear prediction is shown in Fig.2.8 [3].

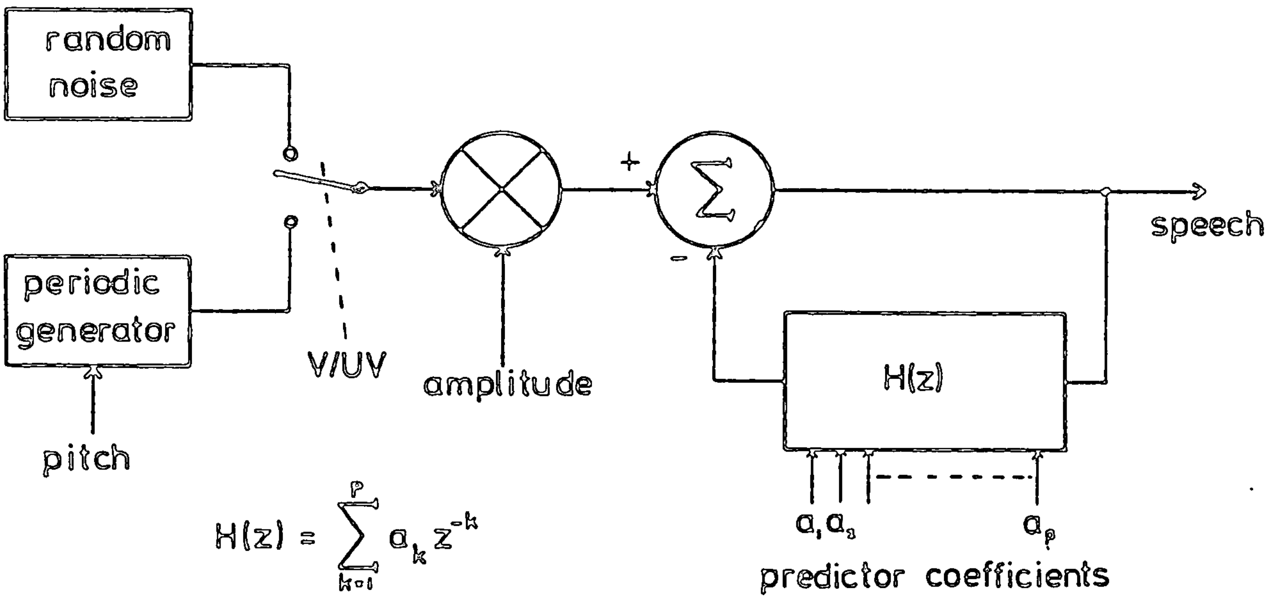


Fig.2.8 Linear Predictive Synthesis

The excitation source is selected in exactly the same manner as in the channel vocoder (section 2.3) and is input to the filter via an amplitude control. The amplitude level is derived from the rms value of the analyser input speech and is necessary since the predictor coefficients contain only information concerning the spectral shape.

Due consideration to the quantisation of control parameters [33] results in vocoders which operate down to 2400 bps with more natural sounding synthetic speech than the equivalent channel vocoder[33]. The predictor typically requires 10-12 coefficients and a sample rate of 10kHz for good quality speech. Charge coupled device programmable transversal filters (chapter 3) are suitable for use in this application, but, because only 10-12 taps are required, it seems likely that digital implementations will almost always be preferred.

2.5 OTHER VOCODER PRINCIPLES

Since the introduction of the digital computer which facilitated simulation studies of complex systems, the interest in speech research has grown enormously and many other vocoding techniques have been reported. Some of these are still too complex for real-time hardware implementation, but others are now realistic.

One such system is the homomorphic vocoder [34] which has the potential for real-time hardware implementation using charge coupled devices. It relies on the deconvolution of the speech excitation source and the vocal tract impulse response by homomorphic filtering [35]; homomorphic filtering is based on the computation of the cepstrum. The vocoder's block diagram is shown in Fig.2.9 where it can be seen that the analyser is identical to the cepstral pitch detector (section 2.2) except for a gating operation which extracts the quefrencies due to the vocal tract (Fig.2.5d). These quefrencies are then coded and transmitted to the synthesiser where the vocal tract impulse response is constructed by an inverse set of operations. Synthetic speech is produced by convolving the vocal tract impulse response with the excitation source. Computer simulations [34] have shown that high quality speech may be reconstructed at 7800 bps and compression to 4000 bps may be obtained by more complex coding [36]. Simplification in synthesiser hardware and a bit rate reduction to less than 2000 bps are possible by using a log magnitude approximation filter [37].

A vocoder technique which promises to give an almost optimum bit rate reduction (600 bps) is the formant vocoder. Research has been continuing for many years but still no practical solution has been found. The principle is to extract the centre frequencies and bandwidths of the main formants (there are three in male speech below 3kHz) and to use this information to control a resonant model of speech production [11]. Formant extraction techniques generally rely on formant tracking algorithms which are based on accumulated and detailed experience of speech waveforms. These algorithms have been designed to operate on the short-time spectra of speech [38], autocorrelation functions [39] and linear prediction spectra [40], but, in general, the complex formant movements create problems. Another

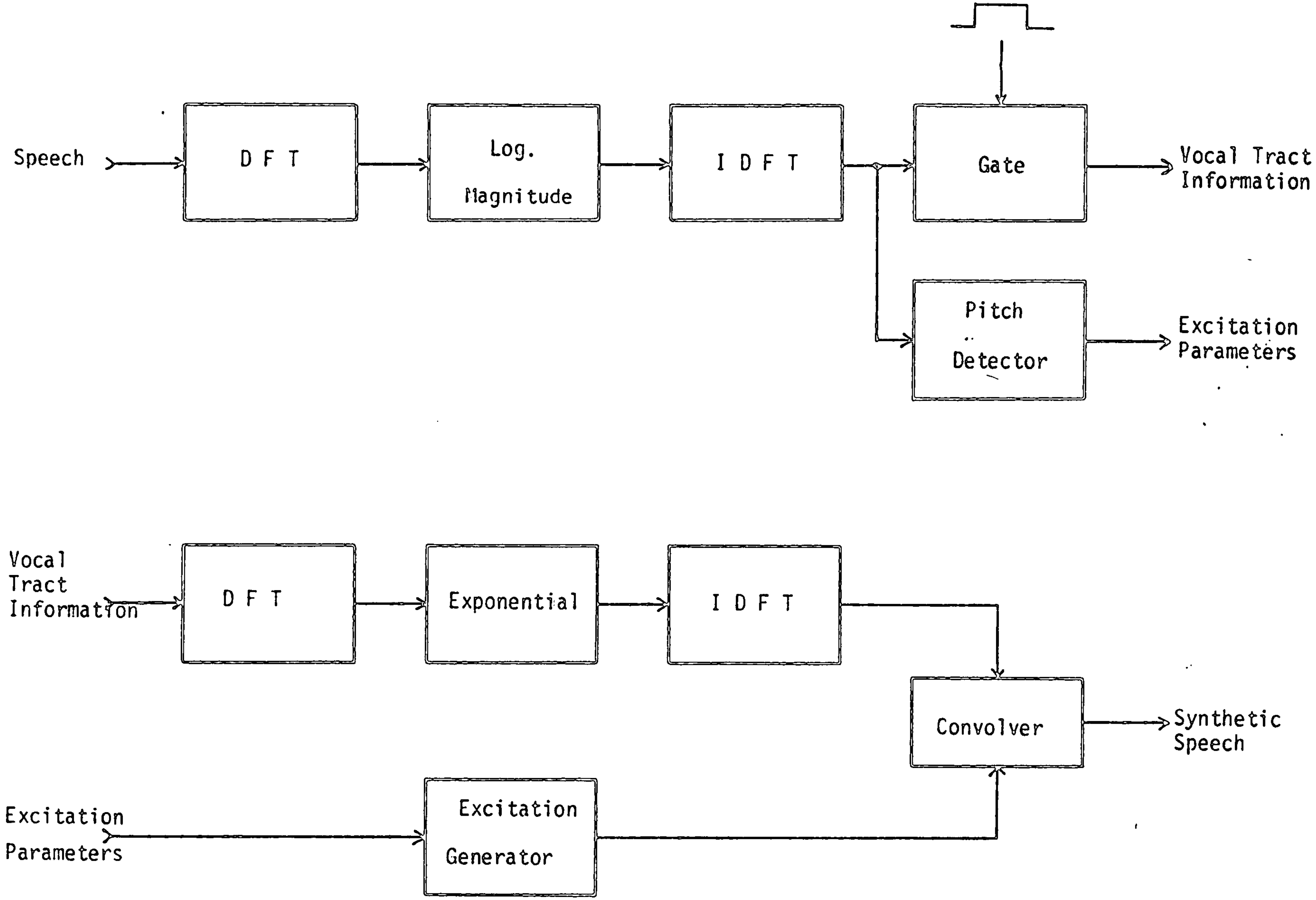


Fig.2.9 The Homomorphic Vocoder

philosophy that applies to formant extraction is analysis-by-synthesis. Here an educated guess is made of the formant parameters and a spectrum is generated which is compared to the actual speech spectrum. The formant parameters are then varied until the difference between the two is minimised according to some criterion [41]. The latter technique has some advantages because the entire spectral shape is considered and not simply the spectral peaks.

CHAPTER 3

THE CHARGE COUPLED DEVICE

The Charge Coupled Device (CCD) is essentially an analogue shift register which can be fabricated as an integrated circuit using Metal Oxide Silicon (MOS) technology. Discrete samples of input signal are stored as charge packets in potential wells and these may be moved along the CCD register by applying a sequence of clock pulses. The CCD therefore provides the flexibility of a time-quantised, clock variable system which does not require analogue to digital conversion.

The CCD was first reported by Boyle [42] in 1970 and is a member of the more general Charge Transfer Device (CTD) family which includes the earlier Bucket Brigade Device (BBD) [43]. In recent years the CCD has become important because:

1. the technology (MOS) is standard and hence is low cost

2. the silicon area required per stage is very small

3. the CCD can be used to process either analogue or digital electrical signals, or optical signals

4. the range of applications is very wide.

Section 3.1 explains the basic principles of CCD charge storage and charge transfer whilst section 3.2 discusses the merits of several methods of charge input and output. The third section summarises the main defects and causes of degradation in CCDs and, finally, section 3.4 describes one of the most powerful analogue signal processing blocks, the transversal filter.

3.1 BASIC PRINCIPLES

The fundamental concepts of CCD operation [4,44] have been developed directly from the established theory of MOS transistors [45] and, as shown in Fig.3.1, the basic CCD delay line structure resembles a rather large multi-gate MOS transistor. The input diffusion converts a sample of the input signal into a charge packet of minority carriers, which is subsequently transferred along the register at a rate controlled by the clock waveforms. After some delay, this charge packet can be sensed at the output diffusion.

To understand the CCD operation it is best to start by examining the basic CCD storage element [45], the MOS capacitor shown in Fig.3.2. With zero potential on the gate there is a uniform distribution of majority carriers (holes in this case) in the "p type" semiconductor. When the gate terminal is pulsed more positive than the substrate, majority carriers from the silicon/silicon dioxide (Si/SiO2) interface immediately below the gate are repelled, thereby

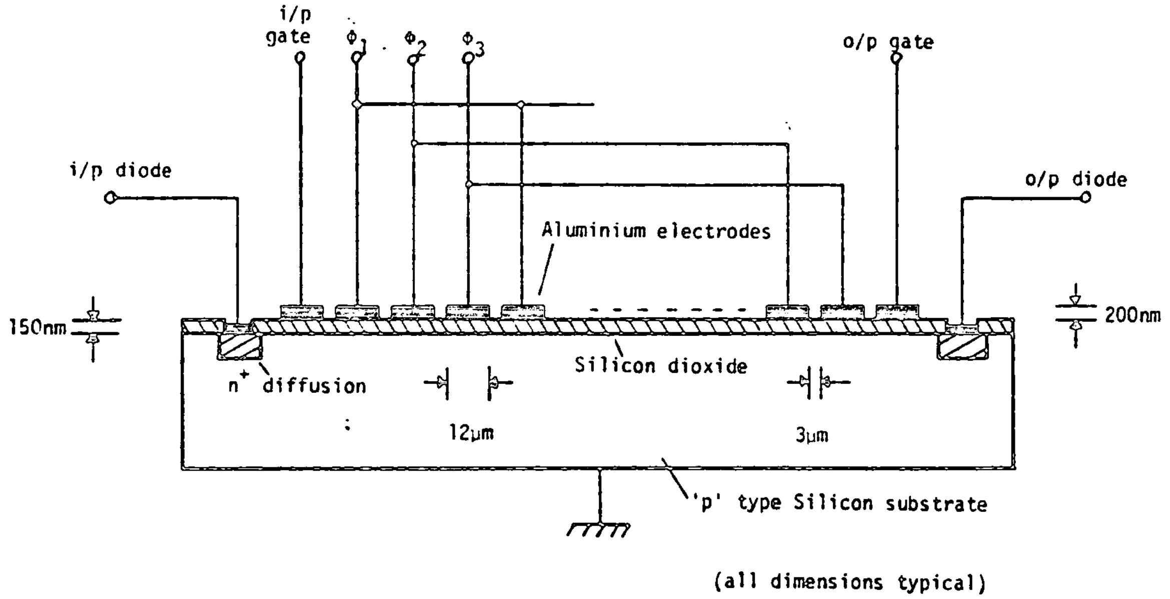


Fig.3.1 Cross Section of CCD Delay Line

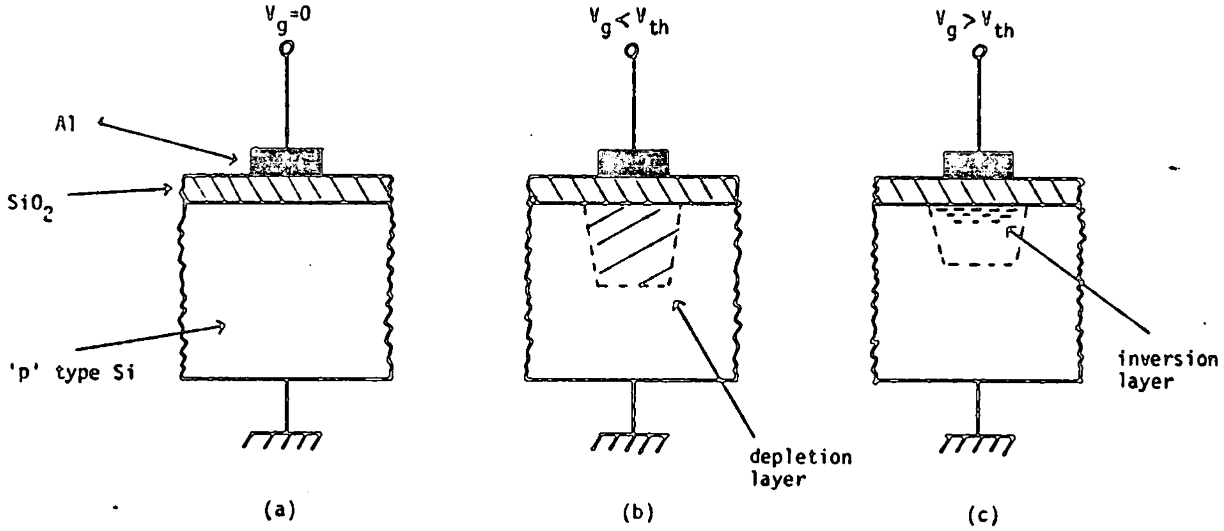


Fig.3.2 The CCD Storage Element

creating a depletion region (Fig.3.2b). If the pulse amplitude exceeds the threshold voltage, Vth, minority carriers (electrons) can be attracted towards the interface to form an extremely thin "inversion layer" (Fig.3.2c). As the amount of charge stored is increased, the extent of the depletion region must decrease to preserve charge neutrality in the system. The creation of this layer corresponds to the formation of a channel in a MOS transistor. These minority carriers can be stored in the inversion layer for typically hundreds of milli-seconds before thermally generated minority carriers from within the depletion region significantly distort the charge packet.

An extremely useful model for visualising the operation of CCD structures results from the "potential well" concept [46]. If a potential well is formed by applying a gate pulse greater than the threshold voltage, then the introduction of minority carriers is analogous to liquid being poured into a well. The maximum quantity of charge which can be stored in the structure (typically 1pC) depends on the volume of the potential well; the depth of the well is related to both the magnitude of the gate pulse and the oxide thickness, and the area of the well is defined by the electrode area.

The next operation is to transfer the stored packet of charge to an adjacent CCD element. Consider the structure

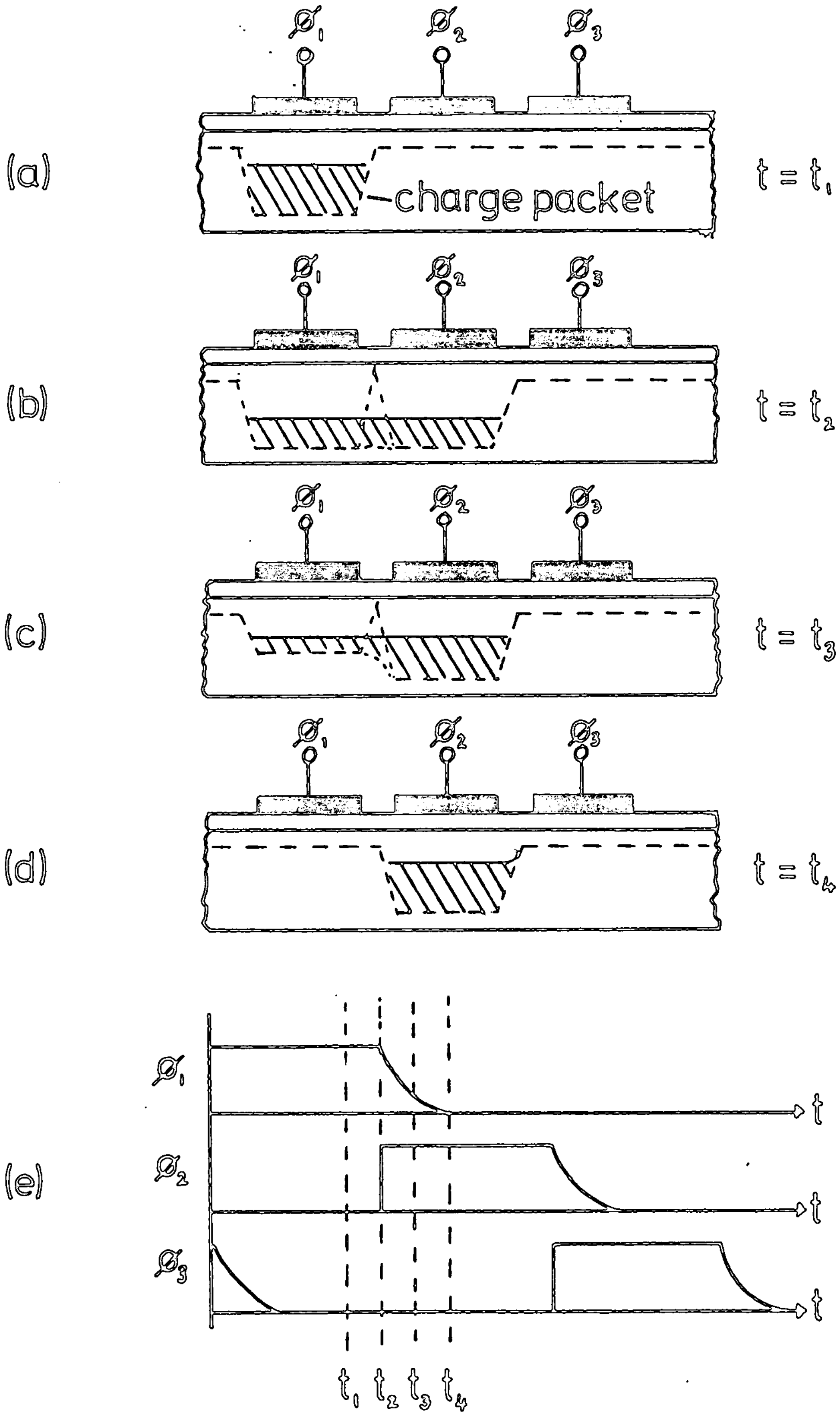


Fig.3.3 Charge Transfer in CCD

illustrated in Fig.3.3a where there is a charge packet stored under the Ø1 electrode. If each electrode is physically very close to its neighbour (<3μm separation), then when Ø2 turns on, the depletion regions under Ø1 and Ø2 will merge and the charge will redistribute itself (Fig.3.3b). By slowly reducing the potential on Ø1, the charge remaining under Ø1 will spill over into the Ø2 well to make the transfer complete at time t4. It is therefore possible to transfer charge packets along an entire register by applying the appropriate time sequence of pulses. In the simple structure shown in Fig.3.3, a three phase clocking system is necessary to propagate the charge unambiguously in one direction. This structure has, however, certain practical limitations (some of which are discussed in section 3.3) and many other more sophisticated electrode arrangements have been developed [43].

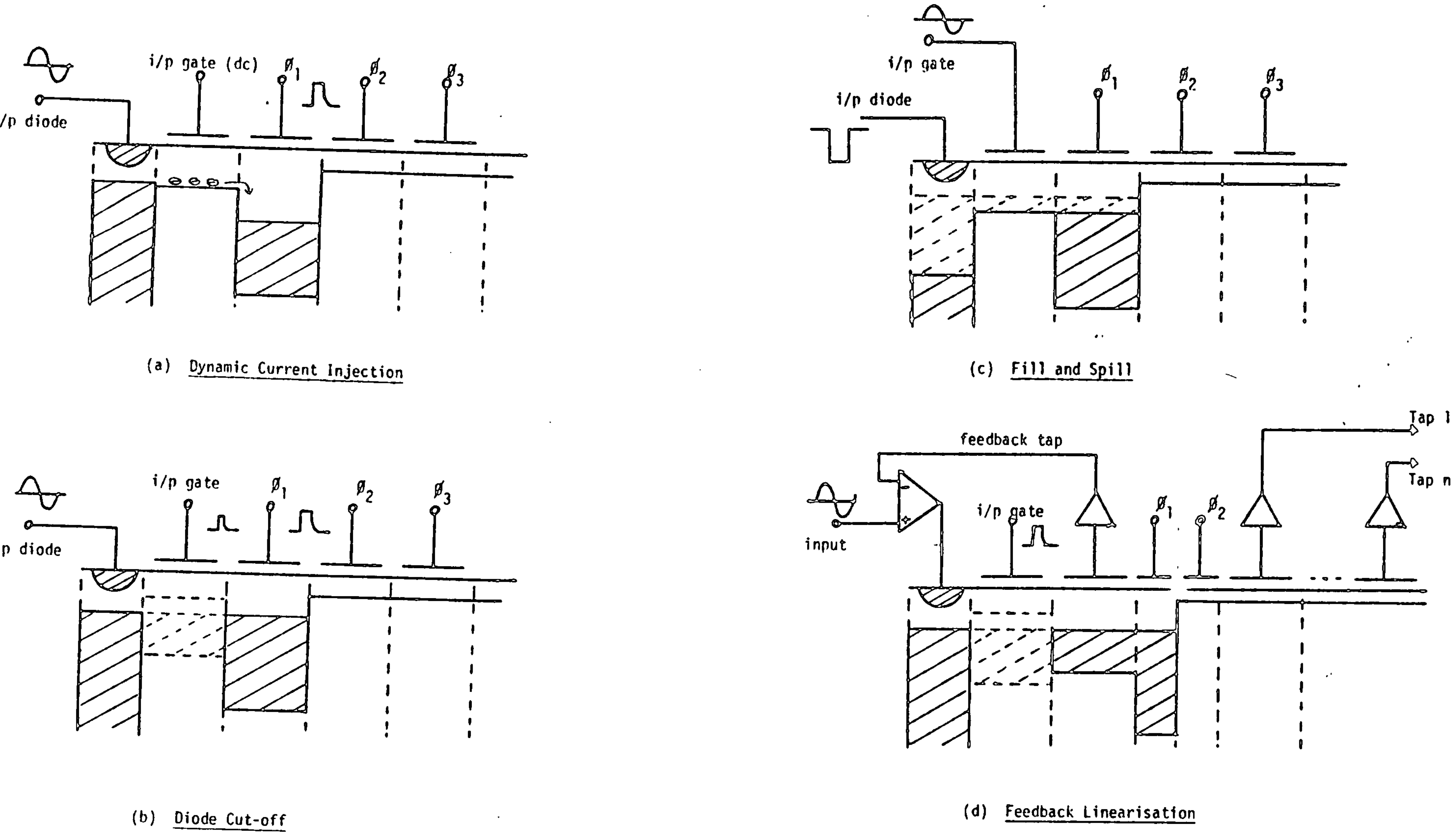
3.2 CHARGE INPUT AND OUTPUT

When the CCD is used in an analogue mode, the linearity of charge input and output is extremely important. In the following section, four serial input schemes of varying complexity and performance will be compared and three output techniques, one serial and two parallel, will be discussed.

3.2.1 Input Techniques

Dynamic current injection [47], which uses the input structure in Fig.3.4a, is one of the simplest input techniques. The input signal is applied to the input diffusion and the input gate is held at a relatively low d.c. potential. When the $\phi 1$ potential well is created, charge flows from the diffusion across the gate into the well. The size of the injected charge packet is determined by the input diode potential, the channel conductance and the available injection time (governed by the CCD clock). This process is inherently non-linear and the resulting distortion is quite severe. For a sinusoidal input giving full well capacity, the 2nd harmonic is typically at -16dB and the third harmonic at -30dB [48].

In a diode cut-off scheme [49] (Fig.3.4b), the signal is normally capacitively coupled to a reverse biassed input diode. During $\phi 1$, the input gate is pulsed on to allow minority carriers from the diffusion to flow into the $\phi 1$ potential well. The surface potential is therefore set directly by the diode potential and the sample is trapped when the input gate is turned off. The trailing edge of the gate pulse has to be designed carefully because: (a) if the channel is cut-off too quickly, some charge from within the channel will be emptied into the signal packet (partition noise) and (b) if the channel is cut-off too slowly, the exact sampling instant will depend on the signal level.



(a) Dynamic Current Injection

(c) Fill and Spill

(b) Diode Cut-off

(d) Feedback Linearisation

Fig.3.4 Charge Input Techniques

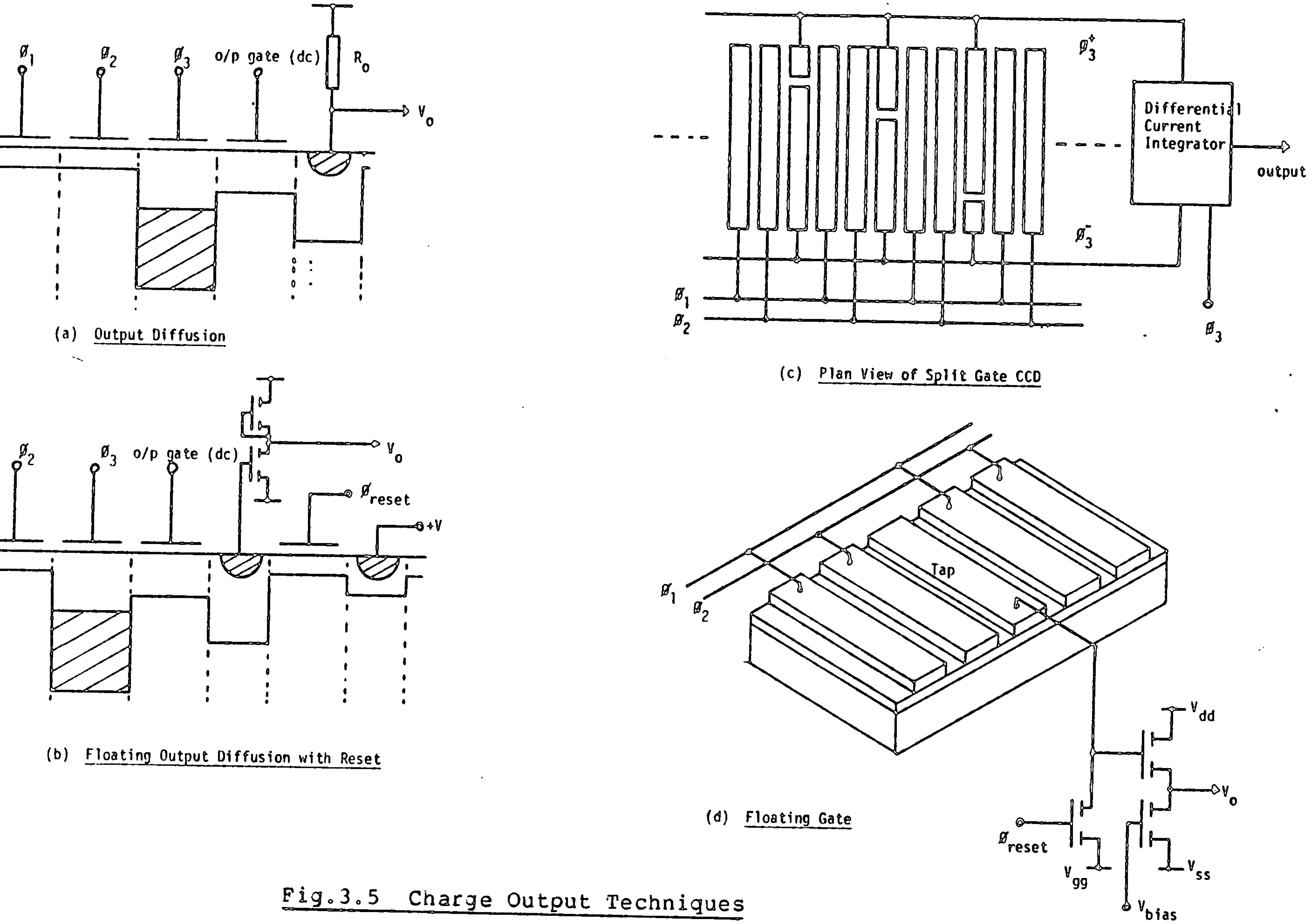
However, even with an ideal gate cut-off, this method still has an inherent non-linearity due to the depletion capacitance changing with the diode potential. These effects give rise to second harmonics in the order of -26dB [48].

A significant improvement in linearity may be obtained by using a fill and spill method [50] of charge input. In the variation shown in Fig.3.4c, the input signal is applied to the control gate and the input diode is pulsed to a low potential to fill the well created under $\phi 1$. When this pulse is returned to a high potential, the excess charge drains back into the diffusion, leaving a charge packet proportional to the gate potential. Two second order distortions are present: (a) spurious noise on the $\phi 1$ driving waveform enters the signal packet directly and (b) the signal dependent fringe field from the input gate alters the effective area of the $\phi 1$ potential well. However, both of these problems may be eliminated by a more complex "pump priming" method of fill and spill [51]. Second harmonic distortion components of less than -40dB have been reported [50].

The technique described above helps to linearise the CCD input structure; feedback linearisation [52], however, attempts to linearise the complete CCD input to output transfer function. An essential feature of the input structure in Fig.3.4d is the inclusion of an extra non-destructive output tap to monitor the input charge. The output is compared with the original signal to generate an error which subsequently corrects the stored charge. If the monitoring tap is electrically identical to all the other output taps, the CCD transfer function will tend to be linearised and the total harmonic distortion will theoretically be reduced by the open loop gain of the system (in practice to less than -40dB [53]). The disadvantage of this technique is the need for a high quality differential amplifier which may be difficult to integrate with the CCD.

3.2.2 Output Techniques

When the CCD is used as a serial delay line, an output diffusion [46] (Fig.3.5a) senses the magnitude of the charge packet. The pn junction is normally held reverse biassed and is positioned so that its depletion region couples with that of the last storage element. An extra gate held at a constant bias is generally included to help minimise capacitive pick-up from the last transfer electrode. When $\phi 3$ is turned off, any charge in the $\phi 3$ potential well will be collected by the output diode to appear as a current change in the output circuitry. A voltage output can be produced simply by incorporating a resistor. However, output changes are very small because of the minute charge packets and, except in wideband application, it is desirable



(a) Output Diffusion

(b) Floating Output Diffusion with Reset

(c) Plan View of Split Gate CCD

(d) Floating Gate

Fig.3.5 Charge Output Techniques

to perform on-chip amplification. This is best accomplished by a MOS transistor with its gate connected directly to the sense diffusion. Since in this case the sense diffusion is floating, an extra diffusion and control gate are required to reset the sense diffusion after the detection of each charge packet (Fig.3.5b).

Split electrode tapping [54] is an extremely elegant method for the implementation of fixed weight transversal filters (section 3.4). The basic split electrode arrangement for use in a 3 phase CCD is shown in Fig.3.5c. Here the third electrode in each cell is divided into two sections and each is connected to either the ∅3+ or the ∅3- clock lines. As charge transfers into the region under a gate, an opposite charge is induced onto the electrode from the clock line. Assuming that the oxide capacitance is very much greater than the depletion capacitance and that the latter may be considered constant, the induced current in a ∅3 clock line due to one section of a split electrode is proportional to the amount of charge in that potential well times the area of the section. The transversal filter is obtained by differencing the total current change in each clock line. Thus, a split in the middle of an electrode corresponds to a weighting factor of zero. This technique does not interfere with the signal charge packet in any way and is therefore non-destructive.

A very powerful and flexible mode of CCD operation is made possible by the floating gate tapping technique [55,56]: serial charge packets may be sensed non-destructively and their magnitudes output in parallel. The applications of this configuration include programmable transversal filtering [57], correlation [8] and adaptive filtering [8].

Fig.3.5d shows the floating gate tap schematic for a three phase CCD with pseudo two phase clocking. The CCD tap electrode is directly connected to the gate of a sense transistor and also to a reset diffusion. Assuming that the reset transistor is off and that the tapped CCD electrode is floating at a potential of Vgg, the transfer of minority carriers into the potential well below this electrode induces a charge redistribution which causes a related change in electrode potential. This voltage change is buffered by a MOS source follower to provide an output signal at a low impedance. After transfer of this charge packet to the next potential well (under Ø1), the reset transistor is pulsed on to reset the tap electrode potential in preparation for the next cycle.

3.3 DEVICE LIMITATIONS AND DEFECTS

3.3.1 Transfer Efficiency

The charge transfer efficiency is an important measure of device performance in analogue signal processing. When a charge packet is transferred from under one electrode to the next, some of the charge is left behind and some lost completely. Two main effects are responsible for this inefficiency, the first of which is due to "interface states" at the Si/SiO2 boundary [58]. As each charge packet is passed along the device, interface states are filled almost instantaneously by minority carriers and then, when the charge packet moves on, the states are emptied much more slowly. Some of the emitted charges return to the correct packet but others empty into trailing packets. The primary effect of these states can be reduced considerably by passing a background charge or "fat zero" continuously along the device. The second source of inefficiency is caused by the transfer mechanism itself [59]. When the transfer process begins, minority carriers move across quickly under the influence of a drift field. As the charge in the new well builds up, the drift field is reduced and the predominant transfer process becomes thermal diffusion, which is a relatively slow process characterised by a time constant defined by the electrode length and the carrier mobility. This time constant therefore gives a trade-off between clock frequency and transfer efficiency. The efficiency can be maximised by careful design of the driving waveforms and by making the interelectrode spacing as small as possible.

The effect of charge transfer efficiency can be visualised by impulsing a CCD delay line. The delayed output will consist of an attenuated version of the original pulse, followed by a time series of smaller residual pulses. In practical CCD delay lines only the first residual is normally significant. This smearing gives rise to a low pass filter characteristic in the frequency domain reducing the device bandwidth. A simple analytical expression relating the transfer efficiency to the frequency response has been developed by Vanstone et. al. [60] using Z-transforms. The amplitude response at the nth output stage is

$$A_n(w) = \left[\frac{\alpha^2}{1 + \varepsilon^2 - 2\,\varepsilon\cos(wT)}\right]^{n/2} \qquad \ldots (3.1)$$

and the phase response is given by

$$\phi_n(w) = \tan^{-1}\left[\frac{n\sin(wT)}{\varepsilon - \cos(wT)}\right] \qquad \ldots (3.2)$$

where α is the transfer efficiency per stage, ε is the transfer inefficiency ($\varepsilon=1-\alpha$), T is the sampling period and w is the angular frequency of the input. Fig.3.6 shows a plot of the normalised amplitude transfer function for various values of the transfer inefficiency product $n\varepsilon$.

In current CCDs, charge transfer inefficiency is in the order of 0.0001 which restricts the number of serial stages to 1000. However, this depends to a great extent on the

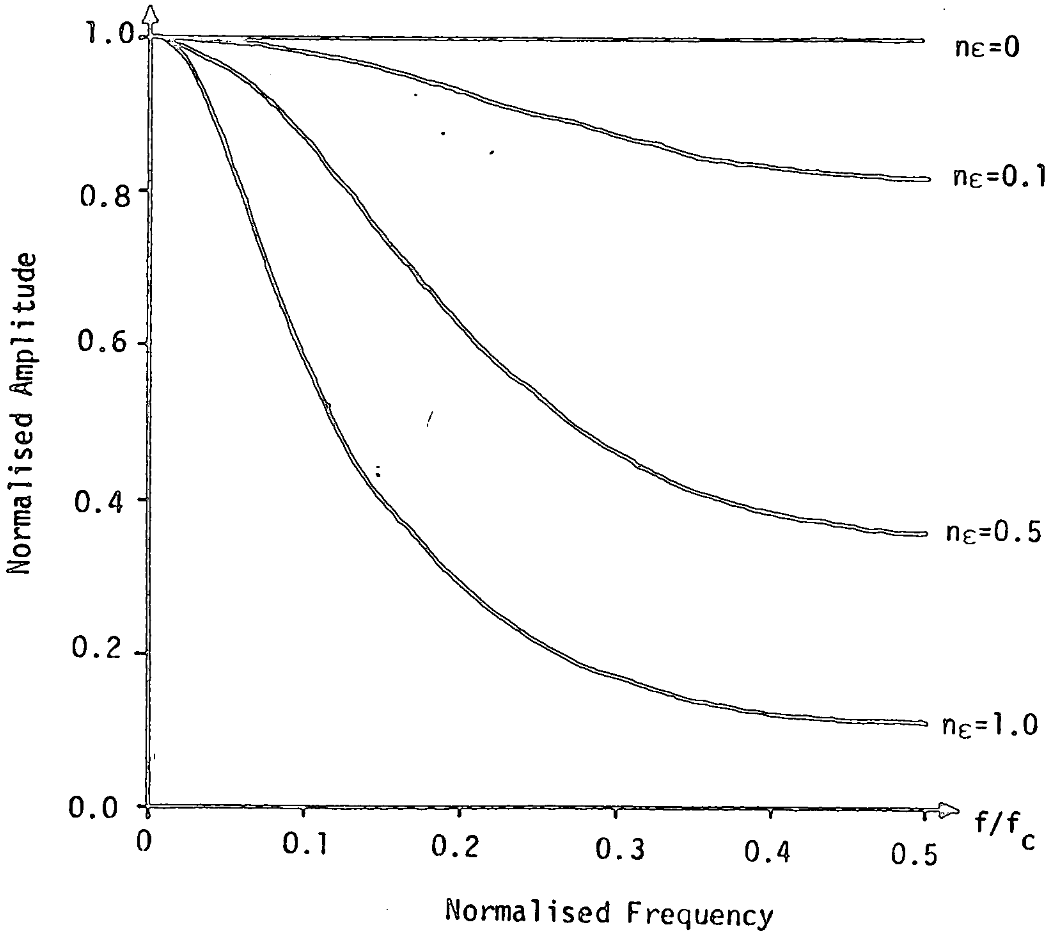


Fig.3.6 CCD Amplitude Frequency Response

application and the amount of permissible signal degradation. Several techniques have been developed to compensate for charge transfer inefficiency [61,62,63], but in general these result in considerable circuit complexity or redundancy.

3.3.2 Noise [64]

Noise sources can be classified into four different categories; input, storage, transfer and output. The combined effect of all noise sources is to limit the dynamic range.

The input and output noise sources depend largely on the particular techniques used. For example, dynamic current injection suffers from random fluctuations in voltage levels and pulse jitter in the clocking waveforms whereas dynamic cut-off has associated partition noise and sampling jitter. In the split electrode tapping technique, the summing amplifier noise tends to dominate all other sources.

Of the inherent noise groups, storage and transfer, the most significant sources are due to the shot noise in dark current (storage) and the fluctuating trapping of fast interface states (transfer). In typical signal processing applications, this last source causes most concern and may be minimised by careful design. Signal to noise ratios in the order of 70-80dB have been achieved in current devices.

3.3.3 Dark Current [44]

Dark current is the equivalent of "leakage current" in MOS transistors and is caused by the thermal generation of minority carriers both in the bulk semiconductor and at the Si/SiO2 interface. This extra charge accumulates in the potential wells, thereby degrading the stored information. At normal temperatures, dark current limits the maximum storage time to several hundred milli-seconds. In continuously clocked delay lines, the dark current effect simply reduces the dynamic range, whereas in transversal filters (SIPO) there is a non-uniform noise distribution. The level of dark current approximately doubles for every ten degrees centigrade increase in substrate temperature, necessitating careful consideration of the total on-chip power dissipation.

3.3.4 Peripheral On-chip Circuitry

To make the CCD appear as a "black box" which may be readily configured for any system, it is highly desirable to integrate along with the CCD many of the necessary peripherals. For example, the operation of a CCD requires clock drivers, timing logic, input and output amplifiers, anti-aliasing filters and sample and hold gates. All of these functions have, of course, to be realisable in a compatible technology.

The integration of the clock drivers and timing logic is relatively easy, but their inclusion normally limits the maximum operating frequency to about 1MHz. This limit is due to the power dissipated when driving capacitive clock lines. MOS amplifiers have been improved considerably in recent years and suitable amplifier designs have been reported [65] which operate at over 1MHz bandwidth with very low d.c. drift. The anti-aliasing filters and sample and hold circuits should be clock variable; switched capacitor techniques [66] provide suitable characteristics at audio frequencies.

3.4 THE TRANSVERSAL FILTER

The transversal filter structure shown in Fig.3.7 is an extremely powerful and flexible building block in analogue (and digital) signal processing [5].

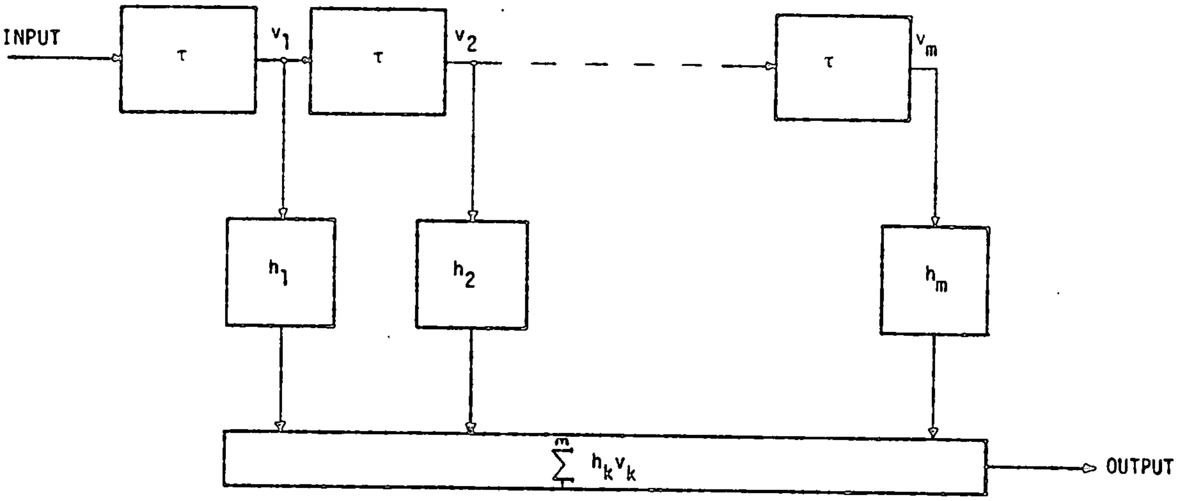


Fig.3.7 Transversal Filter Schematic

It consists of an N-stage shift register with non-destructive taps after each delay, T. Each tap output is multiplied by a weighting coefficient, h (k=1,2,...m) and the results are summed. The filter output is given by

$$V_o(nT) = \sum_{k=1}^{m} V_i\,(nT - kT + T)\,h_k \qquad \ldots (3.3)$$

which is the discrete convolution of the input with an impulse response function (correlation may be obtained by time reversing the impulse response). The frequency response of this filter is given by the discrete Fourier transform of the weighting coefficients. Therefore, by modifying the weights appropriately, any linear Finite Impulse Response (FIR) filter [67] can be constructed.

Split electrode weighting (see section 2.2.2) is an extremely efficient technique for the implementation of fixed weight transversal filters in CCD, and one powerful application of these filters is in the CCD Chirp-Z transform processor (chapters 4 and 5). If, however, the weights are made electrically programmable, then a vast range of sophisticated analogue signal processing applications is possible e.g. time variant adaptive filtering [7] and programmable correlation [8,68].

CHAPTER 4

CCD FOURIER TRANSFORM PROCESSORS

When analysing electrical signals, it is most common to display the waveform in the time domain. This gives the necessary amplitude and timing information. However, in certain cases, it can be more illuminating to picture the same information from an entirely different viewpoint, i.e. from in the frequency domain. In much the same way, it is sometimes much more powerful to perform signal processing in the frequency domain.

This chapter discusses the advantages and disadvantages of several techniques for time to frequency domain conversion (and the inverse) when applied to signal processing. Section 4.1 reviews the conventional filtering spectrum analysers, one of which has recently been implemented in CCD. The fundamental mathematical tools relating the two domains, both theoretically and practically, are summarised in section 4.2 along with an efficient discrete transform algorithm suited to microprocessor implementation. Sections 4.3, 4.4 and 4.5 investigate three algorithms which can be realised efficiently using CCD transversal filters to give real-time operation up to several megahertz. The final section compares their performance.

4.1 CONVENTIONAL SPECTRUM ANALYSERS

One of the first systems used to resolve the frequency components of time domain signals employed a variable centre frequency filter to scan the temporal signal. This type of analyser, known as a Tuned Radio Frequency (TRF) analyser, is simple and inexpensive but suffers from several disadvantages. Firstly, because the TRF analyser has a swept filter, its sweep width is limited (usually one decade); secondly, since the swept filter bandwidth is not normally constant with frequency, the resolution is dependent on frequency.

A significant development in spectrum analysis was initiated by the invention of the heterodyne principle. In contrast to the TRF analyser, the heterodyne spectrum analyser uses a bandpass filter with fixed characteristics. The input signal is mixed with a swept local oscillator before being filtered. An output from the filter will be present only when the difference frequency (or the sum frequency) falls within the passband. The advantages of this technique are considerable. It obtains high sensitivity through the use of IF amplifiers and many decades in frequency can be covered. Also, the resolution can be varied by changing the bandwidth of the IF filter and the sweep rate of the local oscillator.

Both the TRF and heterodyne analysers discussed so far are swept tuned and hence the frequency components of a spectrum are sampled sequentially in time. This is sufficient for "off-line" spectrum analysis, but to transform signals continuously "on-line", the output rate of the processor has to be the same as or greater than the input rate. Equivalently, the processor's bandwidth must be at least that of the input signal. Such processors are normally termed real-time.

One way of achieving this real-time performance is to use a bank of staggered band-pass filters, each with equal bandwidth, and to process the input signal in parallel. The frequency range and resolution of this analyser is normally restricted by the amount of hardware required and typical applications (e.g. the channel vocoder) have fewer than 30 resolution bins across the bandwidth. The contiguous filter bank arrangement obviously lacks flexibility and is always used with fixed parameters. More recently, many of the engineering disadvantages of this approach have been relieved. For example, an integrated circuit has recently been reported [30] which houses 19 parallel CCD FIR filters. Alternatively, switched capacitor filters [66] may be used in audio frequency applications.

Another technique, similar to the analogue filter bank but rather more flexible, is the multiplexed digital filter. The filter coefficients are stored in Read Only Memory (ROM) and accessed when required. For a single integrated circuit realisation, the serial digital processing restricts the real-time operation, at present, to several kilohertz. Hardwired versions may be constructed to operate considerably faster (hundreds of kilohertz in real-time) but the power consumption and physical size increase accordingly.

4.2 THE FOURIER TRANSFORM

The principle mathematical tool for time to frequency domain conversion is the Fourier Transform (FT). The definition given in equation 4.1 transforms the time domain signal, f(t), into its frequency domain counterpart F(w). The inverse process, from the frequency to the time domain, is given by the Inverse Fourier Transform (IFT) in equation 4.2. These equations are used widely in communication theory and are fundamental to spectrum analysis.

$$F(w) = \int_{-\infty}^{\infty} f(t)\, e^{-jwt}\, dt \qquad \ldots (4.1)$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(w)\, e^{jwt}\, dw \qquad \ldots (4.2)$$

In general, F(w) and f(t) are complex quantities. The amplitude and phase components may be extracted by taking the modulus and argument in the usual way, i.e. for a complex number x=a+jb

$$|x| = \sqrt{a^2 + b^2} \qquad \ldots (4.3)$$

and

$$\phi(x) = \tan^{-1} (b / a) \qquad \ldots (4.4)$$

The FT pair are well defined for most signals encountered in practical systems. One sufficient but not necessary condition for the existence of the FT is that the time signal, f(t), should have finite energy. Periodic signals, commonly represented by the Fourier series, have infinite energy and are therefore excluded by this condition. However, by the introduction of the Dirac impulse function, $\delta(t)$, which has an infinite amplitude, an infinitesimal width and unit area, it can be shown [69] that the FT of both periodic and singular functions can be defined.

If the time signal, f(t), is zero for all negative time, then the FT is equivalent to the evaluation of the Laplace transform on the imaginary axis in the "s" plane (complex frequency). The FT is therefore a special case of the more general Laplace transform.

One reason for the popularity of the FT pair is its wide range of useful properties [70]. Possibly the most important of these is given by the convolution theorem. If h(t) is the linear, time-invariant impulse response of a system, and this system is excited by the input signal, x(t), then the output, y(t), is given by the convolution integral

$$y(t) = \int_{-\infty}^{\infty} x(\tau)\, h(t-\tau)\, d\tau \qquad \ldots (4.5)$$

The convolution theorem states that the FT of the output, Y(w), is equal to the product of the system transfer function, H(w), and the transformed input signal, X(w), viz.

$$Y(w) = X(w)\ H(w) \qquad \ldots (4.6)$$

Therefore, convolution in the time domain is the same as multiplication in the frequency domain. In addition, it can be shown that convolution in the frequency domain is equivalent to multiplication in the time domain.

4.2.1 The Discrete Fourier Transform

When there is a need to calculate the FT by computer, the definition given in equation 4.1 must be modified because it requires an infinite amount of processing. Firstly, the input function has to be band-limited and sampled at discrete time instants, and secondly, this sequence has to be time truncated to say N points. The resulting definition is an approximation to the FT and is called the Discrete Fourier Transform (DFT). The DFT pair corresponding to equations 4.1 and 4.2 is given by

$$X_k = \sum_{n=0}^{N-1} x_n\ e^{-j2\pi nk/N} \qquad k=0,1..N-1 \qquad \ldots(4.7)$$

$$x_n = \sum_{n=0}^{N-1} X_k\ e^{j2\pi nk/N} \qquad k=0,1..N-1 \qquad \ldots(4.8)$$

where X_k is the kth Fourier coefficient, x_n is the nth sample of the input data and N is the number of points in the transform.

In discrete sampled data analysis, the Z-transform plays the same role as does the Laplace transform in continuous analysis. The s plane is related to the "z" plane through the expression z=exp(st) and the imaginary axis in the s plane maps to the unit circle in the z plane. The equivalence of the FT and the Laplace transform evaluated on the imaginary axis is therefore analogous with that of the N-point DFT and the Z-transform evaluated at N equidistant points round the unit circle.

To find out how closely the DFT approximates to the continuous FT it is necessary to examine each stage in the development of the DFT. Firstly, consider the effect of sampling in the time domain. If the sampling period is Ts, then the output spectrum will contain not only the correct result, but also an infinite number of aliased replicas each separated in frequency by fs, where fs=1/Ts. As long as the input function is band-limited to fs/2 (Nyquist), the aliased spectra will not overlap and there will be no distortion. In the practical case, the band-limiting filter cannot have an infinitely sharp cut-off and so it is usual to sample several times faster than the Nyquist limit. For a fixed filter, the only way to reduce aliasing errors is to increase the sample rate.

Secondly, the input data are truncated to N points. This is equivalent to multiplying the time signal by a rectangular window of length N.Ts and its effect is to convolve the true FT with a (sin x)/x response (which is the FT of a rectangular window). Fig.4.1 compares the true FT of a sinewave (impulse function) with that of the windowed version. It can be seen that the (sin x)/x main lobe width limits the frequency resolution i.e. the transform's ability to distinguish between adjacent frequencies. The main lobe width is inversely proportional to the time domain window length and to increase the inherent frequency resolution therefore requires an increase in window length. (Note that if the window is increased to infinity then the ideal impulse function results). In addition, the (sin x)/x sidelobes create a "leakage" effect and this limits the amplitude resolution. The most significant sidelobes are the first pair, their amplitudes being -13dB with respect to the main lobe peak. However, weighting functions (section 4.2.3) can be employed to increase the amplitude resolution at the expense of frequency resolution.

The final modification necessary to obtain the DFT is sampling in the frequency domain. This is achieved by assuming that N samples of the input function are one period of a periodic waveform. The output spectrum will then consist of N discrete samples, each spaced by 1/(N.Ts) = fs/N. No information is lost by this sampling but great care has to be exercised in the interpretation of such spectra. For example, consider the DFT of a sinewave. If the sinewave is a basis vector (i.e. it has an integral

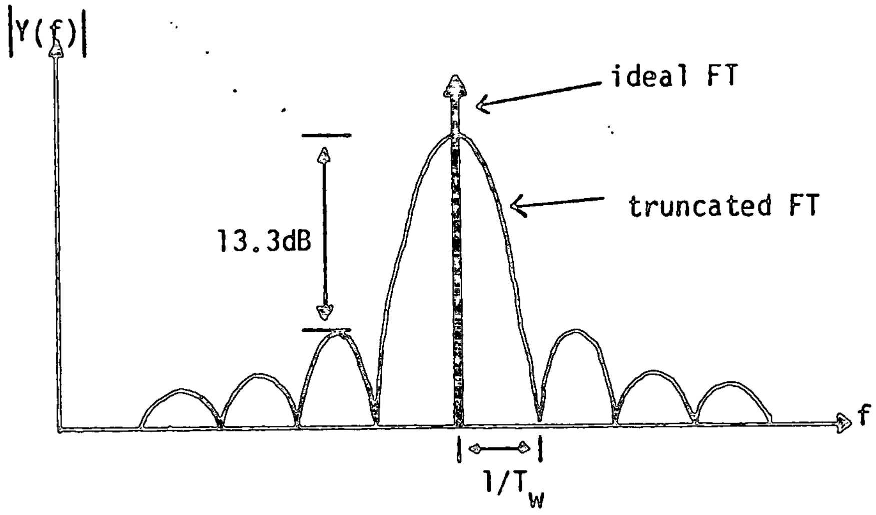
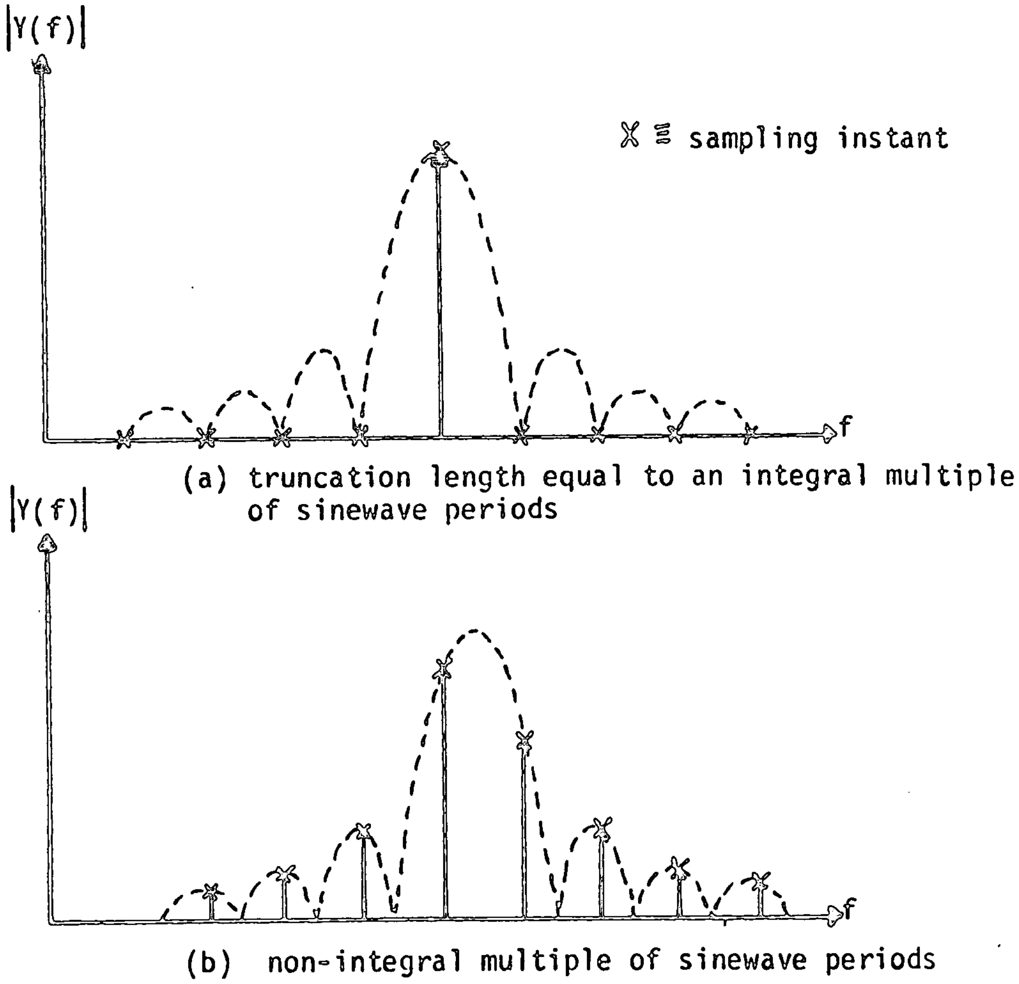


Fig.4.1 The Infinite and the Truncated Fourier Transform



(a) truncation length equal to an integral multiple of sinewave periods

(b) non-integral multiple of sinewave periods

Fig.4.2 DFT of a Sinewave

number of periods within the truncation window) then the resulting frequency domain (sin x)/x will fall exactly on the sampling grid (Fig.4.2a) and the output sequence will be all zeroes except for a "1" at the appropriate frequency sample. If, however, the sinewave is not a basis vector, the (sin x)/x will be offset from the sampling grid to give an output similar to that shown in Fig.4.2b.

4.2.2 The Fast Fourier Transform

It can be seen from equation 4.7 that N^2 complex multiplications and associated additions are required to compute an N-point DFT. Since the processing time and hence the cost are usually proportional to the number of multiplications, the DFT calculation for large N (>64) becomes prohibitive. A Fast Fourier Transform (FFT) is an algorithm which significantly reduces the number of multiplications needed to calculate the exact DFT.

The first FFT algorithm to achieve widespread acclaim was developed by Cooley and Tukey [71] in 1965 and remains today as the foundation for most other FFT algorithms. The mechanics of the Cooley-Tukey FFT algorithm are well documented [72] and it is sufficient to note that the key to its efficiency results from the periodicity of the function W in N, where

$$W = e^{-j2\pi/N} \qquad \ldots (4.9)$$

Using the properties

$$W^{nk} = 1 \quad \text{for all } nk = pN,\ p=0,1..N$$

$$W^{nk+N/2} = -W^{nk}$$

and

$$W^{nk} = W^{nk \text{ modulo}(N)}$$

where nk modulo(N) is the remainder upon division of nk by N, it is possible to structure the DFT to minimise the number of multiplications. Fig.4.3a shows the flow diagram for a decimation in time radix-2 FFT algorithm with N=8. The fundamental operation in this algorithm is the "butterfly" represented by a circle in the flow diagram. Each butterfly takes two complex inputs A and B, and combines them to give P and Q through the operations

$$P = A + W_N^r B \qquad \ldots (4.10)$$

$$Q = A - W_N^r B \qquad \ldots (4.11)$$

where W_N^r are the so called "twiddle factors". To evaluate the complex P and Q using real arithmetic involves four multiplications, three additions and three subtractions (Fig.4.3b).

The complex input sequence $\{x_n\}$, n=0,1...N-1, is initially reordered and the first set of butterflies performs what is essentially a 2-point DFT on pairs of input data. The second set of butterflies combines the 2-point DFTs using twiddle factors to give two 4-point DFTs of the even and odd numbered input data. Finally, these are combined to achieve the 8-point DFT.

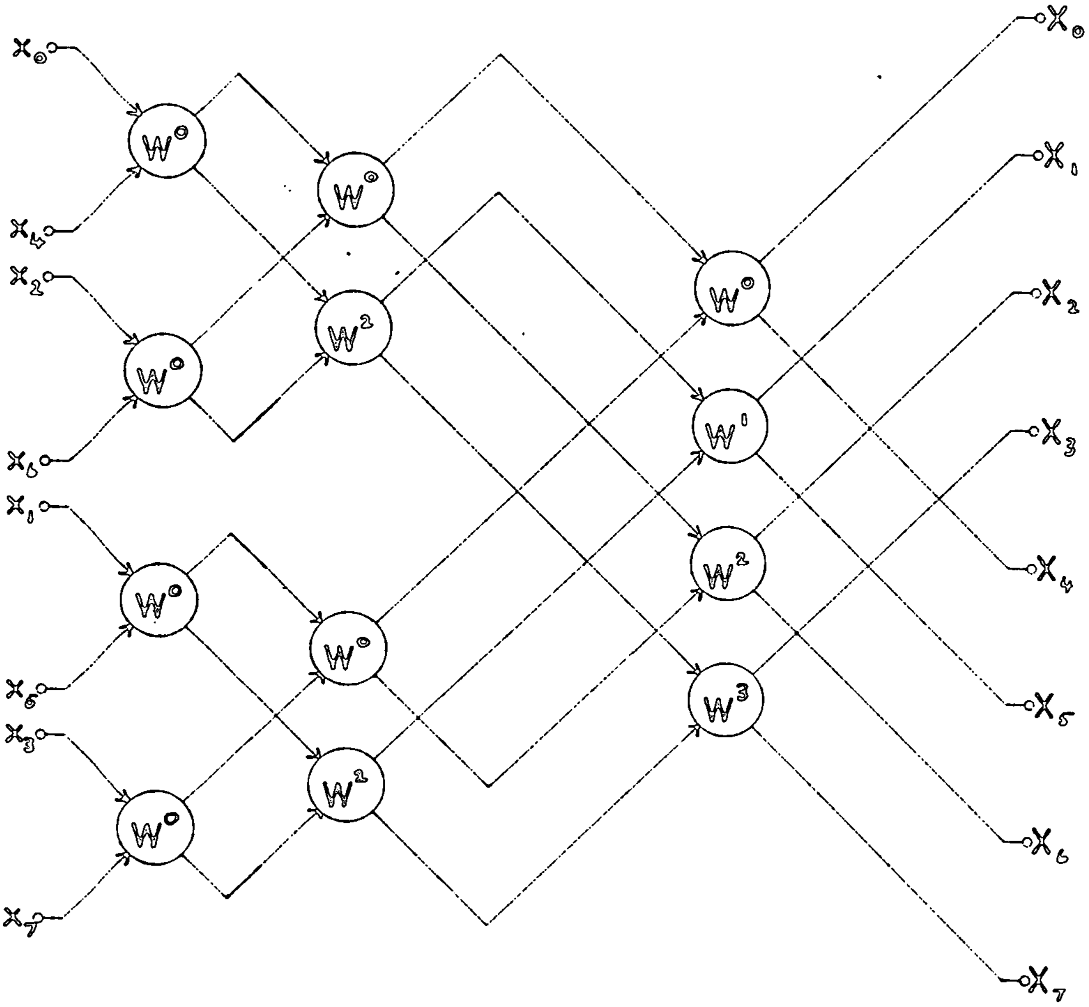


Fig.4.3a 8-point, Radix-2, Decimation in Time FFT

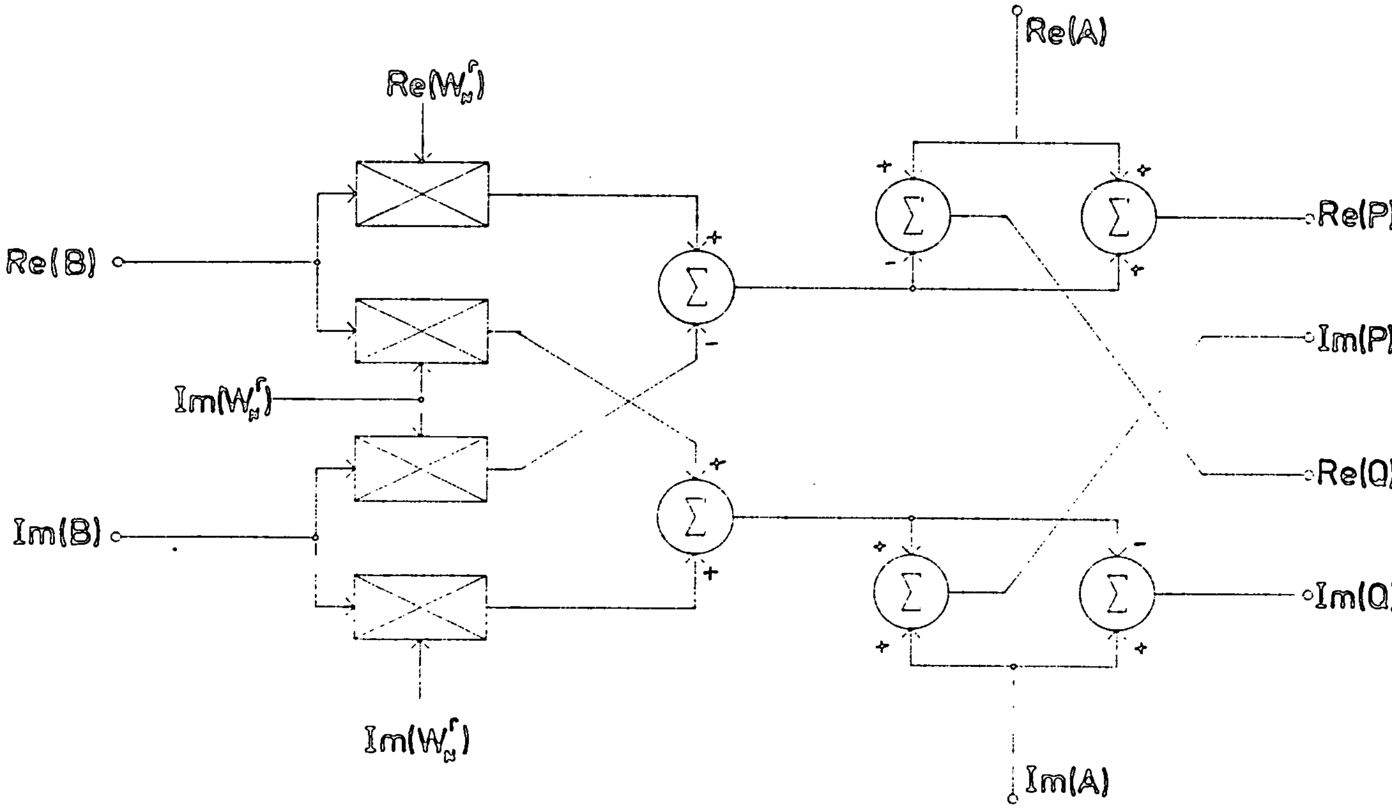


Fig.4.3b Implementation of a Radix-2 Butterfly

From this description, it is clear that N must be restricted to an integral power of 2 for efficient implementation i.e. $N=2^{\gamma}$, where γ is an integer. There are therefore γ or $\log_2 N$ stages each with N/2 complex butterflies so that a total of $(N/2)\log_2 N$ complex butterflies are required to provide the N-point DFT. In terms of real arithmetic, this is a total of $2N\log_2 N$ multiplications, which compares with $4N^2$ for the direct calculation of the DFT. For N=1024, $2N\log_2 N$=20,480 and $4N^2$=4,194,304 ; in this case, a saving of approximately 200:1 in processing time has been achieved.

The FFT accuracy is limited by the finite word lengths used in digital machines [73]. The error sources can be divided into three categories: (1) the analogue input quantisation, (2) the finite word lengths used to represent the twiddle factors and (3) the truncation and round-off within the butterflies. The last error source is the most important because its effects are cumulative and depend on the transform length. Each Fourier coefficient is processed through $\log_2 N$ butterfly operations, which indicates that higher accuracy is necessary for longer transforms.

Many other FFT algorithms have since been developed either to capitalise on particular properties of the input data or to optimise the speed-storage trade-off. For example, if a larger memory is tolerable, a faster transform may be obtained by increasing the FFT radix [67].

An FFT algorithm can be implemented in one of two ways: (a) in software on a general purpose mini- or micro-computer, or (b) as a specialised hardware structure. The software implementations tend to be used for non-real-time applications as the transform rate is limited (typically 600mS for 1024 complex points on microprocessor based systems [74]). Hardware structures commonly employ a single multiplexed high-speed butterfly [75] or make use of pipelining to achieve a transform rate of up to 0.65mS for 1024 complex points [76]. However, the cost, power consumption and size of these array processors greatly restrict the range of possible applications.

4.2.3 On The Use Of Weighting Functions

As dicussed in section 4.2.1, the result of transforming a finite sample of the input data (i.e. a rectangular window) is to convolve the output with a (sin x)/x response. This reduces the frequency resolution and limits the amplitude resolution. The purpose of a weighting function is to reduce the sidelobes without significantly broadening the main lobe.

A weighting function is normally multiplied into the input data before transformation and, in general, brings the data smoothly to zero at the window edges. Many different weighting schemes are available for this and Ref.[77] gives an excellent summary of the most important of these. Alternative figures of merit are given to facilitate the most appropriate choice.

One of the most popular functions is the Hamming window defined by

$$W_n \doteq 0.54 - 0.46\cos(2\pi n/N) \qquad \ldots (4.12)$$

Theoretically, this window gives sidelobes which are approximately -43dB down on the main peak, and broadens the 3dB main lobe width by 1.3 (when compared to a (sin x)/x), which is accepted as a good compromise. Fig.4.4 compares the rectangular window with that of the Hamming window.

Unfortunately, the use of weighting functions inevitably leads to loss of data at the window edges. For on-line signal processing applications, it becomes necessary to overlap successive windows. For instance, if the transform is being used to detect short duration signals, the non-overlapped analysis could miss the event if it occurred near the boundaries. The amount of overlap depends on the weighting function used but is almost always between 50 and 75 percent.

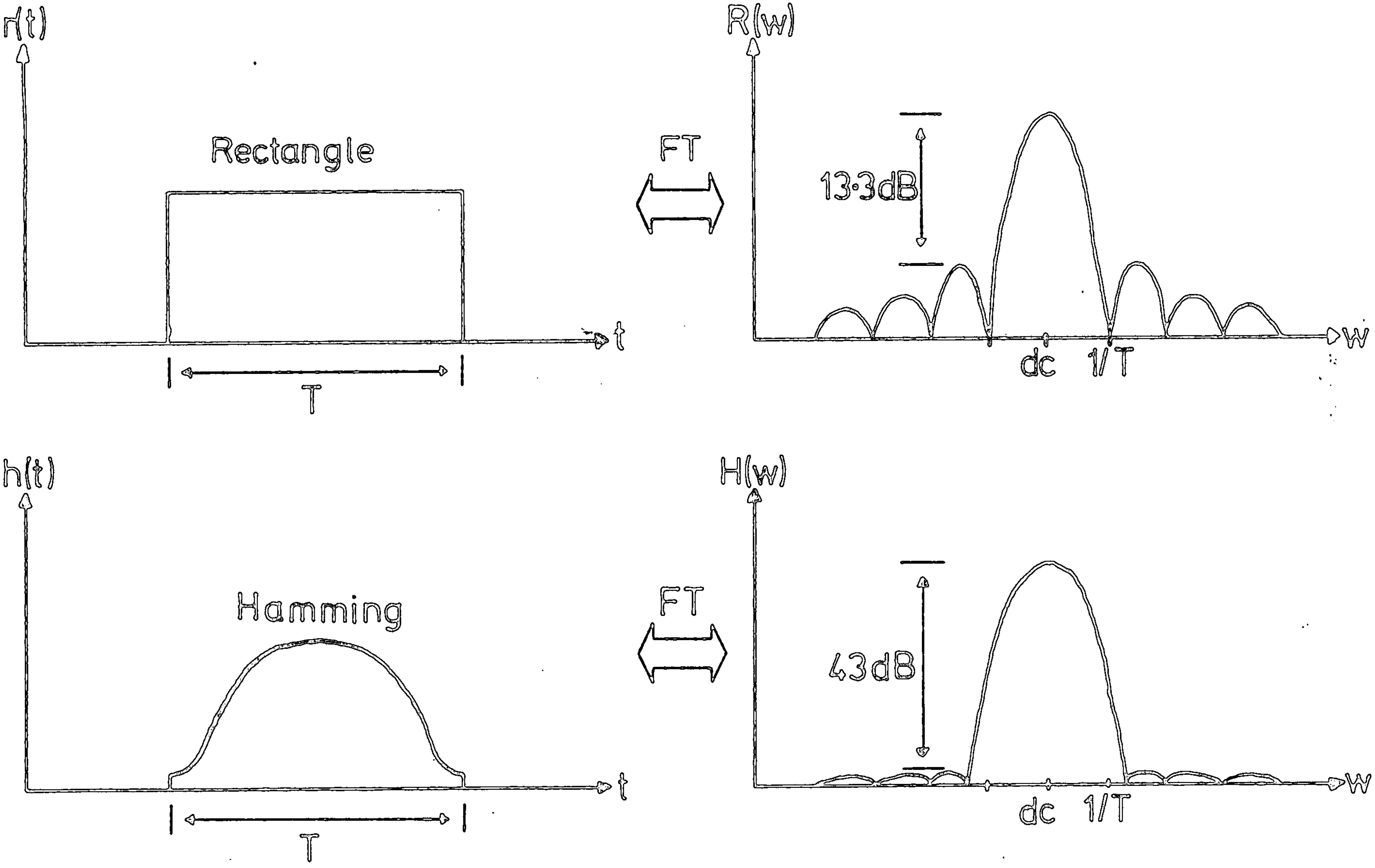


Fig.4.4 Comparison of Weighting Functions

4.3 THE CHIRP Z-TRANSFORM

The Chirp Z-Transform (CZT) [78], as suggested by its name, is a restricted version of the Z-transform. It is however considerably more general than the DFT. The main additional freedoms offered by the CZT are: (a) the number of time samples does not have to equal the number of samples of the Z-transform, and (b) the summation contour in the Z plane need not be a circle, but can spiral in or out with respect to the origin. Historically, the CZT was not considered as useful as the DFT since the special symmetries which are exploited in an FFT derivation are absent. However, because the CZT can be structured to allow efficient real-time implementation of the DFT using analogue CTD transversal filters, the CZT has recently become an extremely important algorithm [79,80,81].

4.3.1 Derivation

The finite Z-transform, $\{X_k\}$, of a sequence, $\{x_n\}$, n=0,1,2...N-1, is defined as

$$X_k = \sum_{n=0}^{N-1} x_n \, z_k^{-n} \qquad \ldots (4.13)$$

The CZT can be derived from equation 4.13 by substituting the restricted contour

$$z_k = A\,W^{-k} \qquad k=0,1..M-1 \qquad \ldots (4.14)$$

where M is an arbitrary integer and both A and W are arbitrary complex numbers of the form

$$A = A_o \exp(j2\pi\phi_o) \qquad \ldots(4.14a)$$

$$W = W_o \exp(j2\pi\phi_o) \qquad \ldots(4.14b)$$

This general Z plane contour, Fig.4.5, begins at the point Z=A and, depending on the value of W, spirals in or out with respect to the origin.

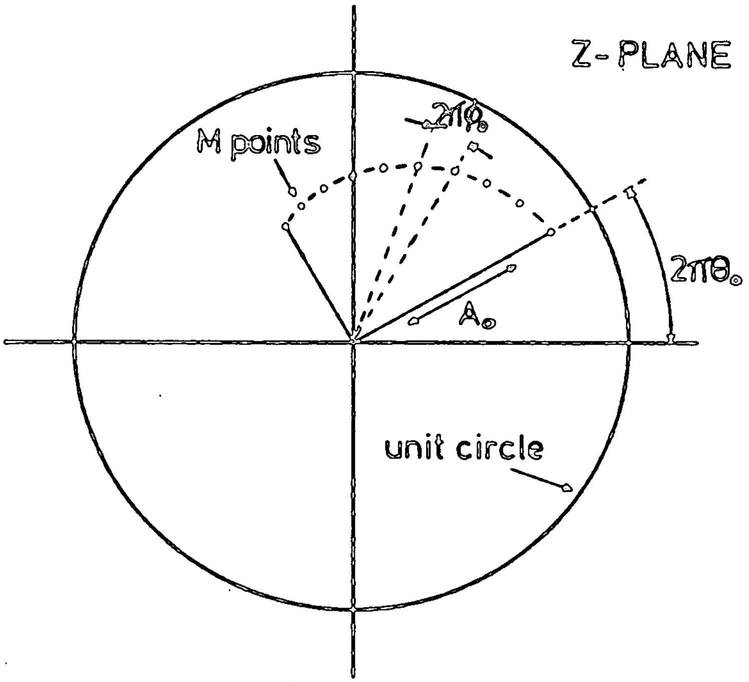


Fig.4.5 The General Z-plane Contour of the CZT

If W =1 then the contour is an arc of a circle. The angular spacing of the samples is $2\pi\phi_o$. The special case of A=1, M=N and W=exp(-j2π/N) corresponds to the DFT.

The key to the usefulness of the CZT is an equality given by Bluestein [82]

$$2nk = n^2 + k^2 - (k - n)^2 \qquad \ldots (4.15)$$

The substitution of this equation into the restricted Z-transform

$$X_k = \sum_{n=0}^{N-1} x_n \, A^{-n} \, W^{nk} \qquad \ldots (4.16)$$

results in an apparently more complicated expression

$$X_k = W^{k^2/2} \sum_{n=0}^{N-1} x_n \, A^{-n} \, W^{n^2/2} \, W^{-(k-n)^2/2} \qquad \ldots (4.17)$$

On close inspection, equation 4.17 can be decomposed into a three step process:

1. pre-multiplication of the input sequence, $\{x_n\}$, by a weighting function to give an intermediate sequence, $\{p_n\}$,

$$p_n = x_n \, A^{-n} \, W^{n^2/2} \qquad \ldots (4.18)$$

2. convolution of the $\{p_n\}$ with a sequence, $\{h_n\}$, where

$$h_n = W^{-n^2/2} \qquad \ldots (4.19)$$

to form the sequence $\{q_k\}$,

$$q_k = \sum_{n=0}^{N-1} p_n \, h_{k-n} \qquad \ldots (4.20)$$

3. post-multiplication of $\{q_k\}$ by $W^{k^2/2}$ to give $\{X_k\}$

$$X_k = q_k \, W^{k^2/2} \qquad \ldots (4.21)$$

This three stage operation is illustrated in Fig.4.6.

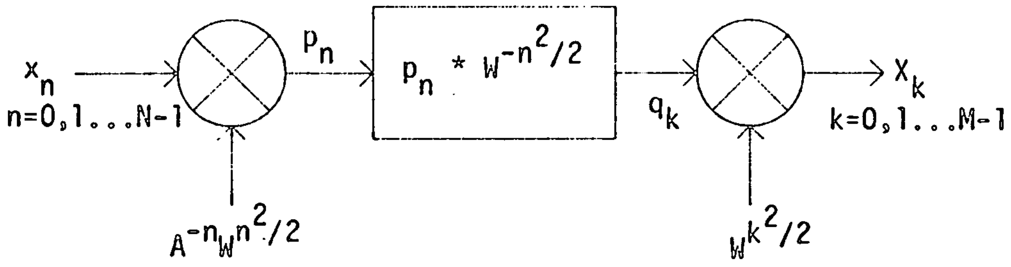


Fig.4.6 The Chirp Z-Transform

The symbol * is used to represent convolution. The advantage of the CZT in a practical implementation is now clear, since a fixed kernal convolution can be performed by a transversal filter in real-time i.e. data can be output as fast as they can be input because the filter multiplies N samples in parallel.

Before proceding to translate the mathematical CZT into a realisable hardware configuration, the application of the CZT's generality is examined. Firstly, is there any advantage in transforming on contours other than the unit circle (DFT)? In linear systems analysis, there is often a need to determine the poles and zeroes. By making the CZT contour pass close to these, the pole and zero positions will be enhanced [83]. This particular advantage is an exception. Most systems are characterised by their response on the unit circle (i.e. the frequency response) and any deviation from this standard would lead to confusion. In signal processing, no advantage would be gained unless there was a requirement to "home in" on features of interest by adaptively shifting the contour. From the practical point of view, major computational inaccuracies can arise when the contour is moved significantly from the unit circle. This is because the function $W_o^{\pm n^2/2}$ is required in the CZT evaluation. (W_o controls the rate of contour spiral). For large N (e.g. 1000), if W_o differs by very much from 1.0, $W_o^{\pm n^2/2}$ can become very large or small when n becomes large [83]. The second CZT freedom is that M, the number of output samples, can be chosen independently from N, the number of input samples. Also, the starting frequency of the contour can be selected. This makes the CZT ideal for high resolution, narrow-band analyses [83]. When using the DFT for such analyses, many of the output points are of no interest and therefore represent wasted processing.

It can be concluded from the above discussion that the CZT's generality is not particularly useful in signal processing, as most applications either require or prefer the now standardised DFT. In the following sections, the CZT will therefore be restricted to the special case corresponding to the DFT.

4.3.2 Implementation

The CZT algorithm for the DFT reduces to

$$X_k = e^{-j\pi k^2/N} \sum_{n=0}^{N-1} x_n \, e^{-j\pi n^2/N} \, e^{j\pi(k-n)^2/N} \quad \ldots (4.22)$$

The pre-multiplying sequence is a constant amplitude complex "chirp" or linear FM function (hence the name Chirp Z-Transform) and the filter needed to perform the convolution has an impulse response which is the time reverse of the pre-multiplying signal. To perform the arithmetic in equation 4.22 using real components demands a parallel or interleaved structure with separate real and imaginary channels. For example, the multiplication of the two complex numbers $x_k = a_k + jb_k$ and $y_m = c_m + jd_m$ (a_k etc. all real) gives

$$x_k \, y_m = (a_k \, c_m - b_k \, d_m) + j\,(a_k \, d_m + b_k \, c_m) \quad \ldots (4.23)$$

Thus the complex CZT processor requires four convolution filters, four separate multipliers for pre-multiplication and four multipliers for post-multiplication. Figure 4.7 shows the expanded block diagram. The pre- and post-multiplying complex chirp functions are now represented by their real and imaginary parts [viz. $\cos(\pi n^2/N)$ and $\sin(\pi n^2/N)$].

A circular convolution is necessary in Equ.4.22 i.e. those values that are shifted from one end of the

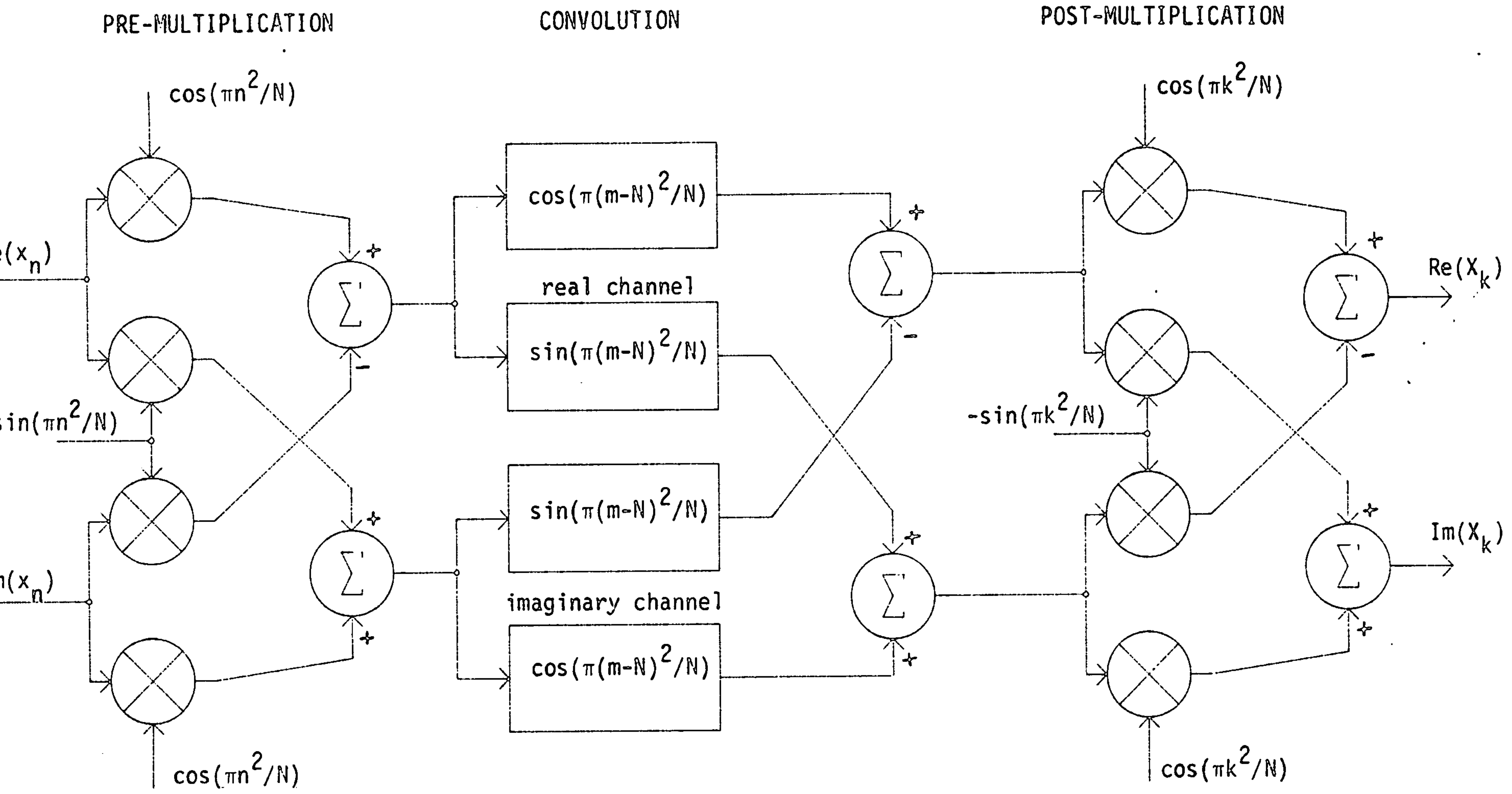


Fig.4.7 Chirp Z-Transform Implementation

summation interval are circulated into the other [84]. Transversal filters can only perform linear convolution. However, a linear convolution can appear as a circular convolution by doubling the length of the filter and padding the input sequence with zeroes [84]. The transversal filters shown in Fig.4.7 therefore need 2N-1 stages and have impulse responses C_n and S_n given by

$$\begin{aligned} C_n &= \cos\left[\pi(m - n)^2 / N\right] \quad m=1,2..2N-1 \\ S_n &= \sin\left[\pi(m - n)^2 / N\right] \quad m=1,2..2N-1 \end{aligned} \qquad \ldots (4.24)$$

where m=k-n+N and m is the mth filter stage. (This can be confirmed by noting that the range of the index k-n is 2N-1).

The operation of the processor is as follows. N sequential samples of the input data are shifted into the processor, pre-multiplied and loaded into the transversal filters. At this point in time these contain N-1 leading zeroes and N data points and after post-multiplication the first output is available. The data are then shifted by one stage and a zero is input to each filter. The second output sample is now available. This operation is repeated until all N output samples have been calculated, at which point the transversal filters contain N leading data points and N-1 trailing zeroes. When a new frame of N data samples is shifted into the processor, the old data are shifted out.

Several undesirable features of this implementation are now apparent. The output must be blanked during the loading of the data and the input must be set to zero during the calculation of the coefficients. This means that the processor has a duty cycle of approximately only 50% (100N/(2N-1)%). In addition, inefficient use is made of the filters since only half of each contains useful information at any point in time.

The use of weighting functions (section 4.2.3) to reduce (sin x)/x sidelobes can be readily incorporated into the pre-multiplying chirps without the addition of extra hardware.

4.3.3 Hardware Reduction

The implementation described in the previous section is configured to process a complex input (real and imaginary parts) and produce a complex output. In many applications, this complexity is not required and savings in hardware are possible.

Table 4.1 summarises several properties of the FT which may be used to reduce the CZT hardware. If the input data are either real or imaginary only (properties 1,2,5,6,7,8 in Table 4.1), then two of the pre-multipliers and both of the input summers are redundant. The reduced pre-multiplication structure is shown in Fig.4.8. For the restricted inputs

	Time Domain h(t)	Frequency Domain H(f)
1	Real	Real part even Imaginary part odd
2	Imaginary	Real part odd Imaginary part even
3	Real even Imaginary odd	Real
4	Real odd Imaginary even	Imaginary
5	Real and even	Real and even
6	Real and odd	Imaginary and odd
7	Imaginary and even	Imaginary and even
8	Imaginary and odd	Real and odd
9	Complex and even	Complex and even
10	Complex and odd	Complex and odd

Table 4.1 Properties of the FT

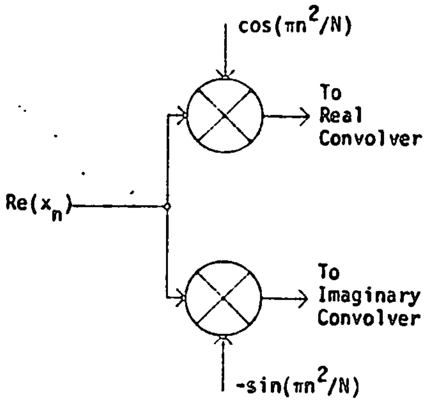


Fig.4.8 Reduced Pre-multiplier for Real Data Only

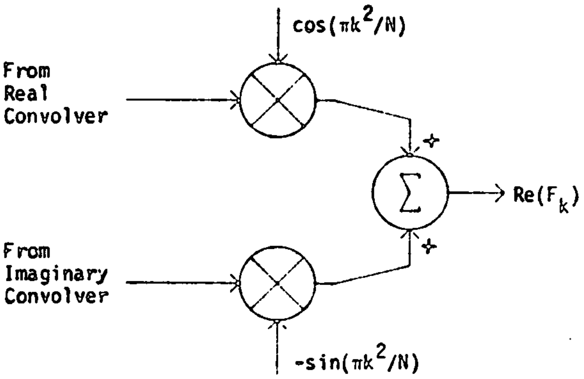


Fig.4.9 Hardware Savings in Post-multiplier

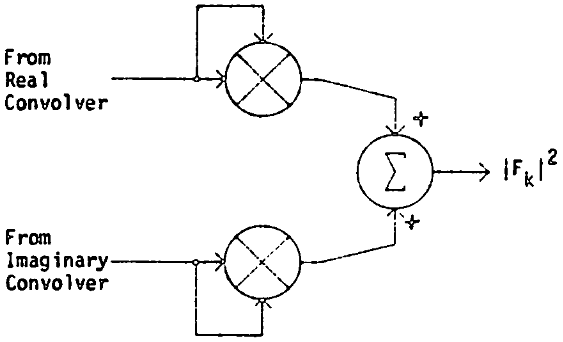


Fig.4.10 Modulus Circuit for Power Spectra

described by cases 3 to 8 in Table 4.1, there is a similar situation in the post-multiplication circuitry and the hardware savings are shown in Fig.4.9.

When only the power spectrum is wanted i.e. when the phase spectrum is irrelevant, the complex post-multiplication may be replaced by a modulus circuit consisting of two squarers and one summer (Fig.4.10). It can be seen from equation 4.22 that the complex post-multiplication function, $\exp(-j\pi k^2/N)$, contributes only to the phase of $\{X_k\}$.

Finally, a technique exists to process simultaneously the DFT of two independent N-point real sequences in a single N-point transformer [69]. A slight increase in peripheral circuitry is required, but by doubling the throughput a substantial increase in hardware efficiency is achieved. If the sequences $\{h_n\}$ and $\{g_n\}$, n=0,1,2...N-1 are purely real, then the complex sequence $\{x_n\}$ can be formed by taking one of the sequences to be imaginary, i.e.

$$x_n = h_n + j\, g_n \qquad \ldots (4.25)$$

The DFT of this sequence, $\{X_k\}$, is

$$X_k = R_k + j\, I_k \qquad k=0,1..N-1 \qquad \ldots (4.26)$$

where R_k and I_k are the real and imaginary parts of X_k, respectively. By making use of the even and odd properties of the FT, in particular 1 and 2 in Table 4.1, it can be shown that

$$2\,H_k = (R_k + R_{N-k}) + j\,(I_k - I_{N-k}) \qquad \ldots\,(4.27)$$

and

$$2\,G_k = (I_k + I_{N-k}) - j\,(R_k - R_{N-k}) \qquad \ldots\,(4.28)$$

where $\{H_k\}$ and $\{G_k\}$ are the DFTs of $\{h_n\}$ and $\{g_n\}$ respectively.

The extra hardware required to sort out the results, as described by equations 4.27 and 4.28, is shown in Fig.4.11.

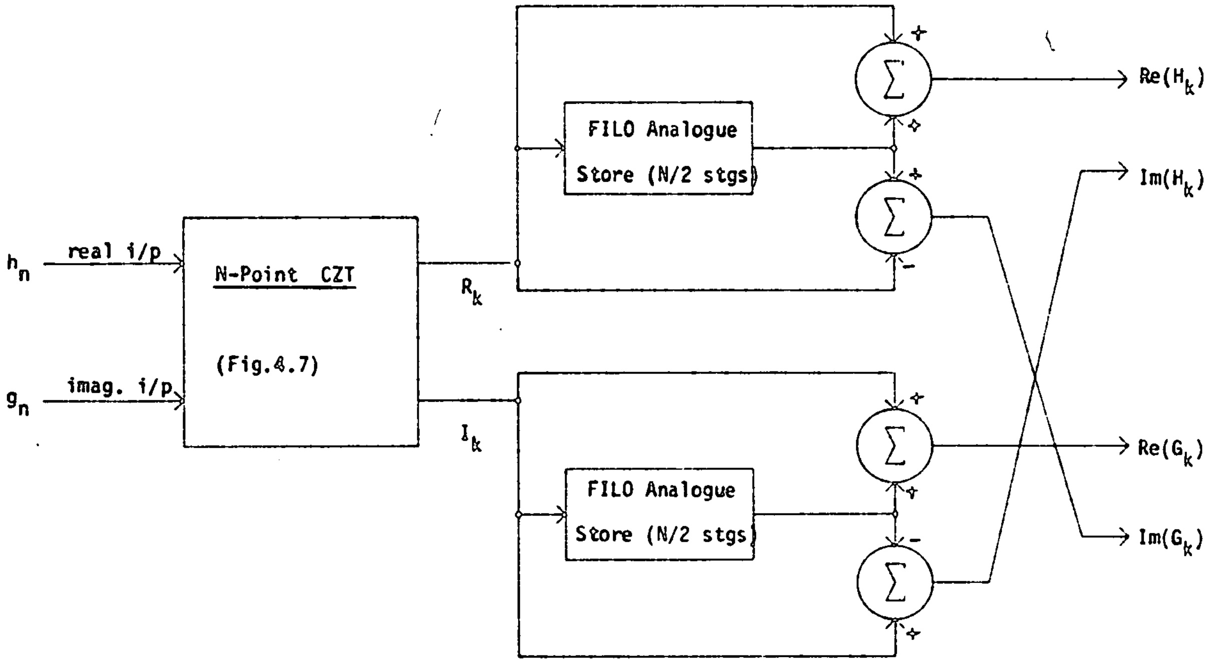


Fig.4.11 Doubling CZT Throughput for Real Inputs

The main components are two N/2 stage first-in last-out analogue delay lines. The first N/2 CZT output points are read into the stores and then read out in reverse order in parallel with the second N/2 CZT output points. After the appropriate summations, the intended data are available. Note that in this implementation, the output data are in reverse order and only the unique N/2 points in each sequence are output. If all N output points are called for the real outputs can be reflected and the imaginary outputs reflected and inverted (property 1 in Table 4.3.1). Thus a doubling in throughput may be obtained by adding N stages of delay and four summers to the complex CZT structure.

4.3.4 Inaccuracies and Limitations

In a CCD implementation of the CZT, the main error sources [85] are due to (1) transversal filter weight accuracy (2) pre- and post-multiplier quantisation (assuming that the sequences are stored in digital ROM) (3) thermal noise and (4) charge transfer efficiency. These errors will be discussed in terms of an r.m.s. noise to signal ratio (N/S) [86] for the purposes of comparison with other transform techniques. In section 5.2, a more practical error definition, the peak error to peak signal ratio, is used in a computer simulation of the CZT.

The weighting coefficient accuracy depends on the particular technique employed. If the tap weights are formed by floating gates (section 3.2.2) and external resistors then a percentage error is appropriate. This is dealt with in section 5.2. The most common tapping technique for fixed weight transversal filters is the split gate (section 3.2.2) where the error depends on the photolithographic resolution. It has been shown [6] that errors for a 500 point CZT are in the order of 0.08% (-62dB). The N/S ratio is approximately independent of N, the number of transform points.

Pre- and post-multiplier sequences are stored typically as 8 bits (including sign) and this quantisation gives a dominant random error in the CZT of about 0.3% (-50dB) [6].

The thermal noise in CCDs gives rise to a signal independent error source analogous to input quantisation in the FFT. N/S ratios of less than -70dB are possible in CCDs, making this error source relatively insignificant.

The final CCD error source is charge transfer efficiency. Since this is a coherent error, its effects can be expressed both as degradation in CZT frequency resolution and as a N/S ratio. The study of a 64 point CZT [86] has shown that for a charge transfer inefficiency, ε , of 0.0001 the N/S ratio is -40dB. In terms of frequency resolution degradation, the sensitivity is found to be three times worse for high frequency than for low frequency inputs. Since longer transforms call for longer transversal filters, the N/S ratio increases with N. For a given ε, the N/S ratio increases by about 5dB for each doubling of N.

Techniques are available for charge transfer compensation (section 3.3) but these tend to be impractical.

The performance limitations of the CCD CZT are set by various aspects of the analogue circuitry. The realistic real-time bandwidth of the processor is limited to 5MHz by peripheral electronics (10MHz clock rate) and significantly less (1MHz) for fully integrated implementations. Charge transfer efficiency and also physical chip size limit the number of transform points to a maximum of 500 (1000 stage convolvers) and thermally generated dark current in the CCDs restricts the total storage time to several hundred milliseconds. This sets the maximum resolution. Due to the processing gain in matched chirp filters (which is defined as the square root of the time-bandwidth product), the CZT's linear dynamic range is limited not by the CCDs but by the output analogue multipliers. In a processor configured for power spectrum output the linear dynamic range is limited to 40dB by the squaring multipliers [87]. The overall accuracy of a 500 point CZT can be likened to an equivalent 13-bit FFT [6].

4.4 THE SLIDING CHIRP Z-TRANSFORM

The sliding variation of the direct CZT permits a reduction in transform hardware but does not give the true DFT for a general input [6]. The Sliding CZT (SCZT) can be defined as

$$X_k^s = e^{-j\pi k^2/N} \sum_{n=k}^{k+N-1} x_n \, e^{-j\pi n^2/N} \, e^{j\pi(k-n)^2/N} \qquad \ldots (4.29)$$

The difference between the direct CZT and equation 4.29 is the summation index which is incremented for each new spectral component so that the current N-point transform depends on data from the immediately following N input points. The only class of input signal for which the SCZT gives the exact DFT is a periodic waveform in N, i.e. $x_n=x_{n+N}$; this is an extremely restricted input. However, if only the power spectral density is required, the range of inputs can be expanded to cover any stationary signal because the indexing only results in a modified phase. (A stationary signal has a constant amplitude spectrum even though each N point time record is different).

Examination of equation 4.29 reveals the main advantage of the SCZT: for an N point transform, the convolution process demands the filters to have only N stages. The filter impulse responses are defined by

$$C^s_m = \cos(\pi(m-N)^2/N) \quad m=1,2..N \qquad \ldots (4.30)$$

$$C^s_m = \sin(\pi(m-N)^2/N) \quad m=1,2..N \qquad \ldots (4.31)$$

where m=k-n+N and m is the mth filter stage. Apart from the obvious hardware savings, the reduced filter lengths mean that the transform degradation due to imperfect charge transfer efficiency is less in the SCZT than in the direct CZT. In addition, since one new data point is input for each spectral coefficient output, the processor has a 100% duty cycle and blanking is not necessary.

4.5 THE PRIME TRANSFORM

The Prime Transform (PT) is an alternative algorithm for computing the DFT. It is suited to CCD implementation because the bulk of the computation is performed in transversal filters. Using a concept from number theory, Rader [88] has demonstrated that the DFT can be calculated from the correlation of two sequences if the number of data samples is prime.

4.5.1 Derivation

The derivation begins by writing the DFT in a more convenient form

$$X_o = \sum_{n=0}^{N-1} x_n \qquad \ldots (4.32)$$

$$X_k = x_o + \sum_{n=0}^{N-1} x_n W^{nk} \qquad \ldots (4.33)$$

where W=exp(-j2π/N). If N is chosen to be prime, there exists at least one integer R, called a primitive root, which will produce a one to one mapping of the integers n' to the integers n according to the relationship

$$n = R^{n'} \text{ modulo}(N) \qquad n,n'=1,2..N-1 \qquad \ldots (4.34)$$

Similarly,

$$k = R^{k'} \text{ modulo}(N) \qquad k,k'=1,2..N-1 \qquad \ldots (4.35)$$

Taking advantage of the cyclic properties of W, it can be shown [88] that equation 4.33 can be rewritten as

$$X_{((R^{k'}))} = x_o + \sum_{n'=1}^{N-1} x_{((R^{n'}))} W^{((R^{(k'+n')}))} \qquad \ldots (4.36)$$

where ((.)) represents modulo(N). Equations 4.32 and 4.36 together form the PT. The term X_o has to be calculated separately because $((R^{k'}))=0$.

Equation 4.36 thus represents a circular correlation of the permuted (reordered) input sequence $\{x_{((R^{n'}))}\}$ with the permuted values of the complex sinusoid $\{W^{((R^{n'}))}\}$. The resulting sequence $\{X_{((R^{k'}))}\}$ is a permuted sequence of the DFT coefficients $\{X_k\}$.

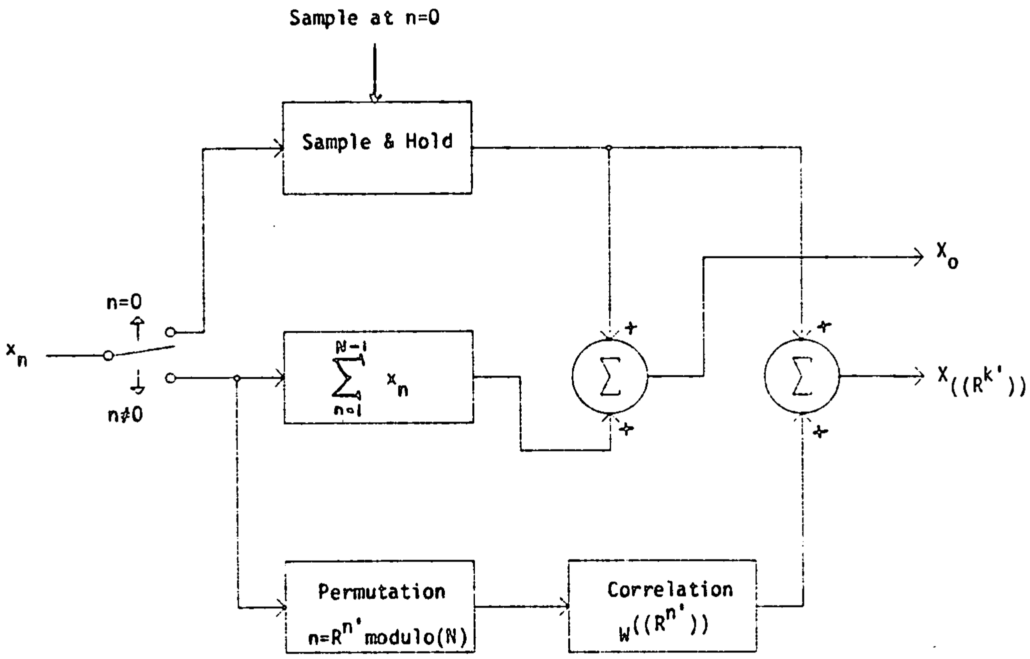


Fig.4.12 The Prime Transform

Fig.4.12 gives the hardware configuration for the complex PT.

4.5.2 Implementation

The PT architecture has to be configured for complex arithmetic using real components (Fig.4.13) in much the same way as the CZT.

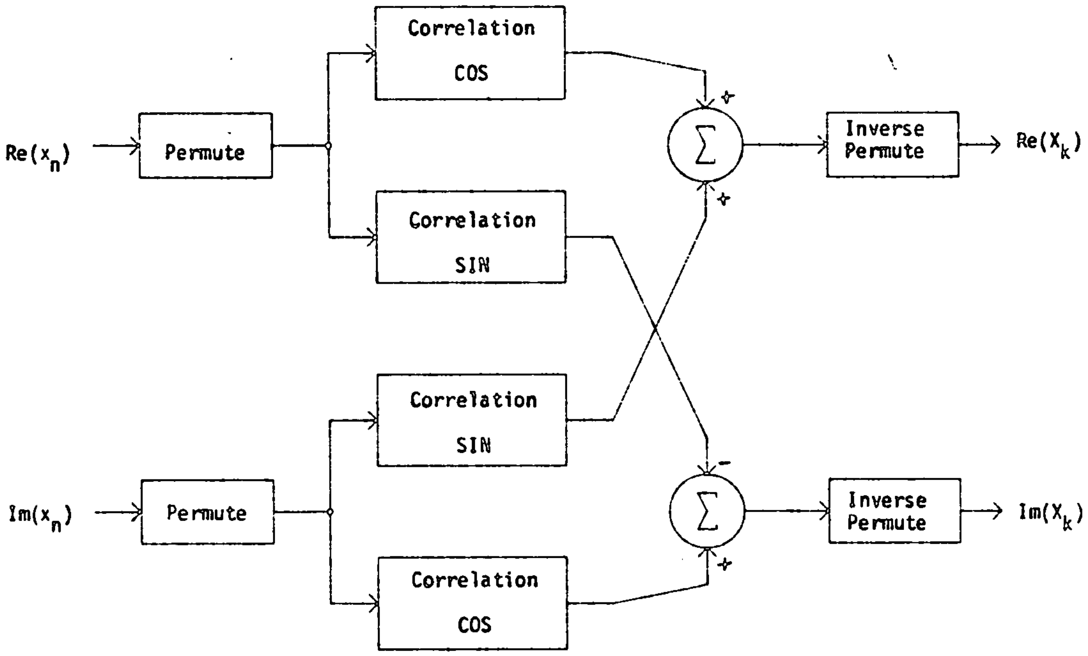


Fig.4.13 Expanded PT Architecture

Note that in Fig.4.13, the calculation of X_o and the addition of the spectral offset, x_o, are not included for clarity. It can be seen that the basic difference between the CZT and the PT is the replacement of the analogue multipliers by analogue permuters. Data permutation, however, does not involve complex arithmetic and hence yields the simpler structure. The permuters can be implemented using Analogue Random Access Memory (ARAM) [89] or a CCD store in conjunction with an analogue demultiplexer

[90].

The circular correlations are performed in transversal filters. Since these filters inherently perform linear convolution, one of the two sequences has to be time reversed. Examination of equation 4.36 reveals that a total of 2N-3 stages are required in each filter. The filter impulse responses are given by

$$C_m^p = \cos\left[2\pi((R^{m+1}))\ /\ N\right] \quad m=1,2..2N-3 \qquad \ldots (4.37)$$

$$S_m^p = \sin\left[2\pi((R^{m+1}))\ /\ N\right] \quad m=1,2..2N-3 \qquad \ldots (4.38)$$

where $m=k'+n'-1$ and m is the mth filter stage. The operation of this convolution is similar to that in the CZT and a duty cycle of approximately 50% results.

The inverse permutations at the output are included to reorder the PT coefficients, thereby giving the conventional DFT. However, if subsequent processing involves inverse Prime transforming, these may be omitted.

4.5.3 Hardware Reduction

In contrast to the CZT, special cases of input signal result in significant hardware savings. For the case of real data only, two permuters, two convolvers and two summers become redundant (Fig.4.14). A further reduction is possible if the data are purely real and also even. In this

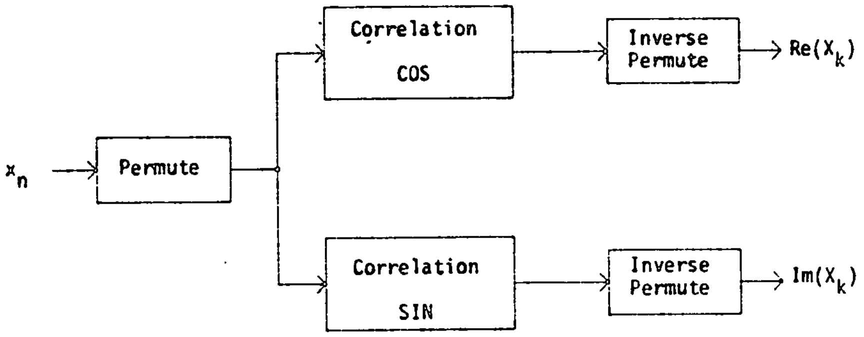


Fig.4.14 PT Hardware Reduction for Real I/P Data Only

case there is no imaginary output and the PT is reduced to the Discrete Cosine Transform (DCT), requiring only one correlator (Fig.4.15).

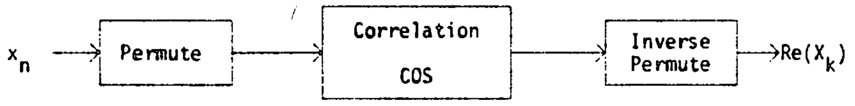


Fig.4.15 PT for Real and Even Input Data (DCT)

4.5.4 Errors and Limitations

The error sources in the PT are similar to those in the CZT with the exception of the multiplier quantisation, which has been replaced by a permuter error.

It has been found [86] that the effect of tap weight error in the PT is the same as that in the CZT. The effect of charge transfer efficiency on the PT is not as staightforward as in the CZT case because the correlation for a particular DFT coefficient depends on the permutation code and not on the linear position in the filter. It is therefore not possible to treat charge transfer efficiency as a simple degradation in frequency resolution. For a 67 point PT with a charge transfer inefficiency of 0.0001, the N/S ratio is approximately -41dB [86], which is marginally better than the equivalent CZT.

The Achilles' heel in the PT is the analogue permuter. State-of-the-art permuters using ARAM have an accuracy of between 5% and 10% [89]. An accuracy of 5% gives a N/S ratio of -30dB [86]. The alternative is to perform the permutation digitally i.e. A to D convert, store in digital RAM and finally D to A convert the reordered words. While this approach may achieve superior performance, many of the engineering advantages associated with a CCD PT implementation are lost.

4.6 COMPARISON OF REAL-TIME SPECTRUM ANALYSERS

Table 4.2 summarises the main operating characteristics for real-time spectrum analysers operating below 5MHz. There are three distinct approaches illustrated here: the digital FFT, the analogue CCD DFT processor and the parallel

PROCESSOR RAMETER	F F T † Microcomputer Stand-Alone	F F T I^2L Bit Slice Custom	F F T Specialised Hardware	C Z T Integrated (1977)	C Z T Discrete	S C Z T Discrete	P T Discrete	Filter Bank CCD IC	Filter Bank Switch. Cap. IC	Filter Bank Digital Discrete
x. Number of ansform Points	1024	512	1024	64	512	1024	512	32	32	32
ansform Speed for 2 Complex Points	275ms	9ms	0.65ms	-----	0.1ms	0.1ms	0.1ms	Parallel Processing	Parallel Processing	-----
al-time Process- g Bandwidth	2 kHz	55 kHz	780 kHz	1 MHz	5 MHz	5 MHz	5 MHz	200 kHz	20 kHz	5 kHz
curacy	0.1 %	0.1 %	0.1 %	1 %	1 %	1 %	1 %	1 %	0.3 %	0.1 %
namic Range	12 bits	12 bits	12 bits	40 dB	50 dB	50 dB	60 dB	70 dB	70 dB	13 bits
. of CCD Stages r N pt cmplx trans	-----	-----	-----	8N	8N	4N	8N	100N	-----	-----
st ††	£4500	£3000 *	£75000	£200 **	£2000	£2000	£2000 *	£200 **	£200 **	£5000 *
sical Size	5"x19"x20"	9 chips	43"x19"x28"	IC	1 Board 4"x9"	1 Board 4"x9"	1 Board 4"x9"	IC	IC	19"x12"x5"
antages	Flexibility	Flexibility Accuracy Low Power	Accuracy and Speed	Small Size Low Power Low Cost	High Speed	100% Duty Cycle	Hardware Reduction for Special Inputs	Small Size Low Power	Small Size Semi-Programmable	Programmable
advantages	Limited Real Time Application	Limited Speed	Power Consumption Size and Cost	50% Duty Cycle	50% Duty Cycle Analogue Multipliers	Power Spectra Only	50% Duty Cycle Analog Permuter	Fixed Characteristic	Audio Bandwidth Only	Large Size

† Plessey Micproc
†† hardware costs only

* - Estimate ** - Estimate for large quantities

Table 4.2 Comparison of Real Time DFT Processors

filter bank. Each of these is complementary in that their application areas are well defined and tend not to overlap.

The FFT is used in cases demanding high accuracy coupled with high resolution and, at present, there is no alternative approach. Major problems arise when the real-time bandwidth is much greater than 50kHz and the only solution involves an exponential increase in power, cost and size.

In applications requiring fewer than 32 frequency points the "brute force" parallel filter bank will often give the optimum engineering result. This is especially true for the analogue filter bank because there is no need for A to D and D to A conversion.

The CCD transform processor fits into application areas requiring low power, low cost and only modest accuracy. For a fully complex processor, there is little to choose between the CZT and the PT. However, in cases where the input signal can be restricted sufficiently, the PT offers distinct hardware advantages. The SCZT provides the best solution when the input signal is stationary and only power spectra are required.

CHAPTER 5

THE DESIGN AND CONSTRUCTION

OF A

CCD CHIRP Z-TRANSFORM PROCESSOR

This chapter describes the practical considerations taken into account during the design and construction of a CCD CZT processor. The main design objectives are summarised in section 5.1. Section 5.2 discusses CZT computer simulation results which allow component tolerances to be specified for the implementation in section 5.3. Finally, the hardware performance is examined in section 5.4.

5.1 DESIGN OBJECTIVES

The main objective was the design and construction of a CZT processor suitable for speech processing. For research purposes, maximum flexibility was desirable together with minimal hardware overcomplication. In addition, the operating speed had to be maximised without the need for special high-bandwidth circuit techniques. The restricted availability of suitable CCD transversal filters (Jan.1977) necessitated consideration of hardware efficiency.

The preliminary design specifications resulting from the above were as follows:

1. CONFIGURATION: a structure permitting either a 32-point direct CZT or a 64-point sliding CZT

2. INPUTS: complex (real and imaginary), used in conjunction with a 90-degree phase difference network (section 5.3.5) to maximise convolver efficiency

3. OUTPUTS: power spectra only

4. WEIGHTING: optional (rectangular or Hamming windows)

5. MAX. PROCESSING BANDWIDTH: >100kHz real-time bandwidth (>200kHz clock frequency)

6. LINEAR DYNAMIC RANGE: >40dB

7. POWER DISSIPATION: <10W

5.2 COMPUTER SIMULATION

The need for computer simulation arises because the mathematical analysis of the CZT using real signal representation is a very laborious task (see Appendix B). Moreover, it is often difficult to obtain a closed mathematical solution.

The computer simulation strategy centres around the tecnnique employed to calculate the chirp filter convolutions. These can be computed either directly (i.e. $4N^2$ multiplications) or by means of the convolution theorem and an FFT routine. The direct method is by far the simplest because the hardware structure can be simulated exactly without FFT errors interfering with the results. However, for large N, the direct method is inefficient. Since in this simulation the transform length is less than or equal to 64, the direct convolution method has been adopted and the CCD transversal filters are modelled in the time domain. The simulation flow diagram is the same as the block diagram in Fig.4.7 with the post-multiplication replaced by the modulus circuit (Fig.4.10).

As already discussed in section 4.3.4, the most significant errors in the CZT are due to pre-multiplier sequence quantisation (when the pre-multiplier is a Multiplying D to A Converter(MDAC)), analogue post-multiplier accuracy, transversal filter weight accuracy and charge transfer inefficiency. One additional error source has been considered here: phase shifter accuracy when generating pseudo complex data (section 5.3.5). In sections 5.2.2 through 5.2.7, these errors are quantified in terms of a peak error to peak signal ratio. This is a more practical error classification than the r.m.s. equivalent because in many CZT applications, spurious peak errors may significantly alter the outcome of automatic decision algorithms.

To help in the understanding of the hardware operation, the simulation has been used to obtain a graphical analysis of the CZT i.e. the computer has been used as a software oscilloscope. This analysis is presented in section 5.2.1.

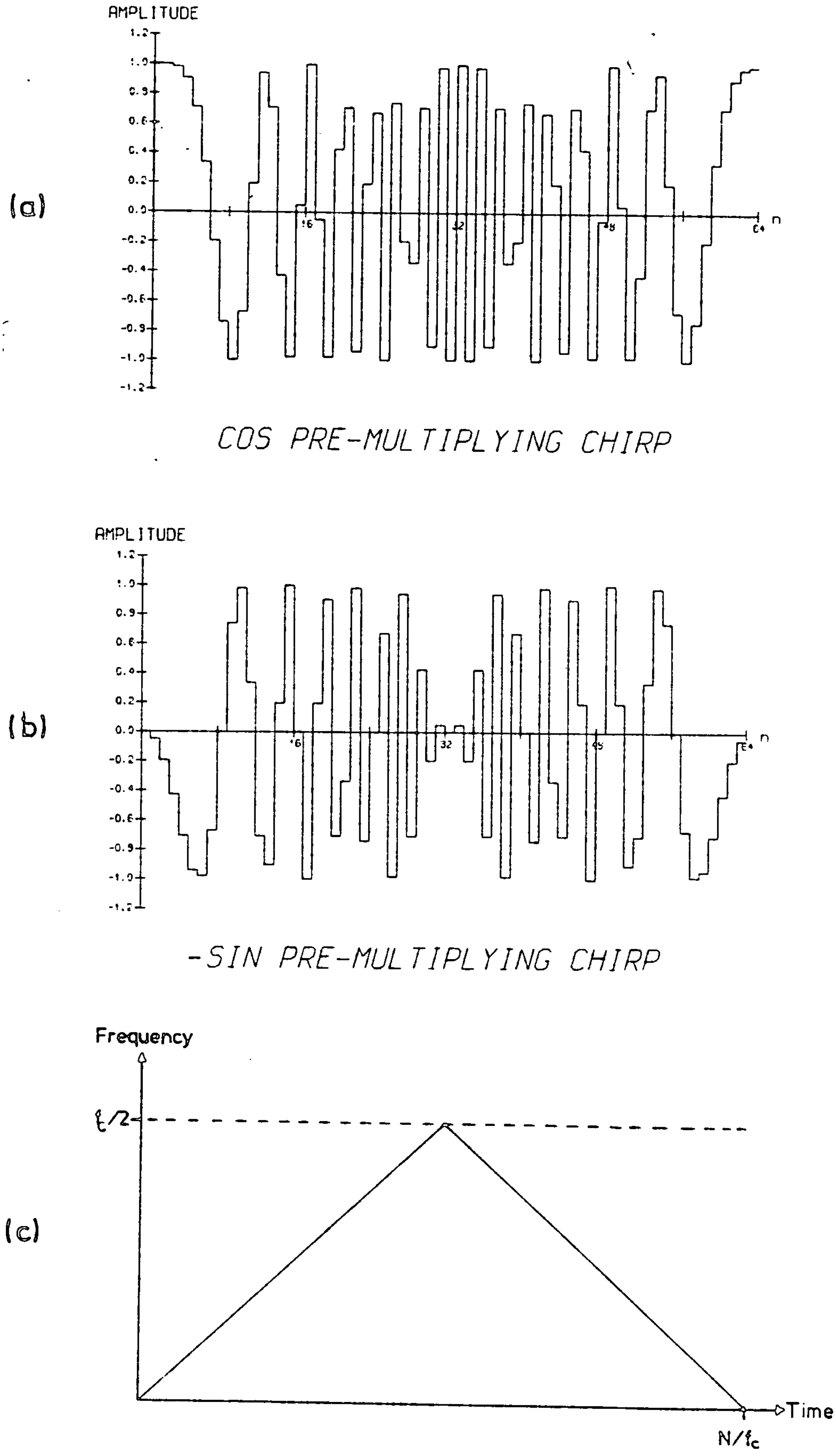
The simulation was written in the Edinburgh IMP language [91] and run on a large time-shared twin ICL 4-75 computer configuration.

5.2.1 Graphical Analysis of the CZT

The graphical analysis is based on a 64-point direct CZT. Reference is made to the block diagram in Fig.4.7.

The pre-multiplying chirps are plotted in Fig.5.1 and appear as "V" chirps because of aliasing at N/2. If the sample frequency is f_c, then the waveforms chirp from dc through $f_c/2$ to dc. When the input signal is a complex basis vector (Figs.5.2a and 5.2b), the pre-multiplication produces the real and imaginary waveforms shown in Figs.5.2c and 5.2d. These waveforms may be considered in terms of sum and difference frequency sidebands. (Note that because the system is at baseband, these sidebands cannot be separated). The upper sideband chirps from f_s, (where f_s is the tone input frequency) -> $f_c/2$ -> dc -> f_s and the lower sideband chirps from f_s -> dc -> $f_c/2$ -> f_s (Fig.5.3).

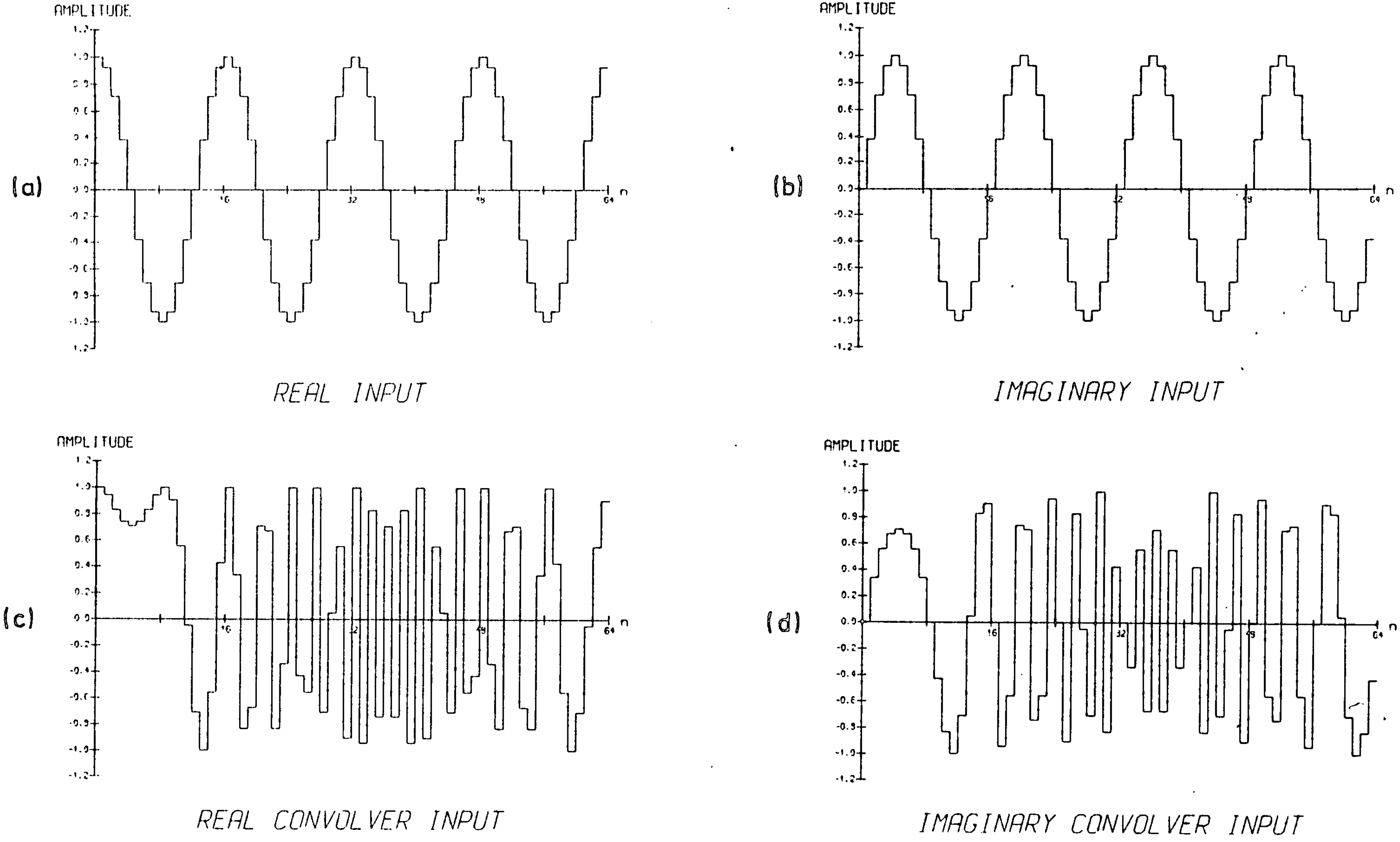
For circular convolution, the convolver filter impulse responses are double "V" chirps lasting for 2N-1 samples.



COS PRE-MULTIPLYING CHIRP

-SIN PRE-MULTIPLYING CHIRP

Fig.5.1 CZT Pre-multiplier Waveforms



REAL INPUT

IMAGINARY INPUT

REAL CONVOLVER INPUT

IMAGINARY CONVOLVER INPUT

Fig.5.2 CZT Pre-multiplication

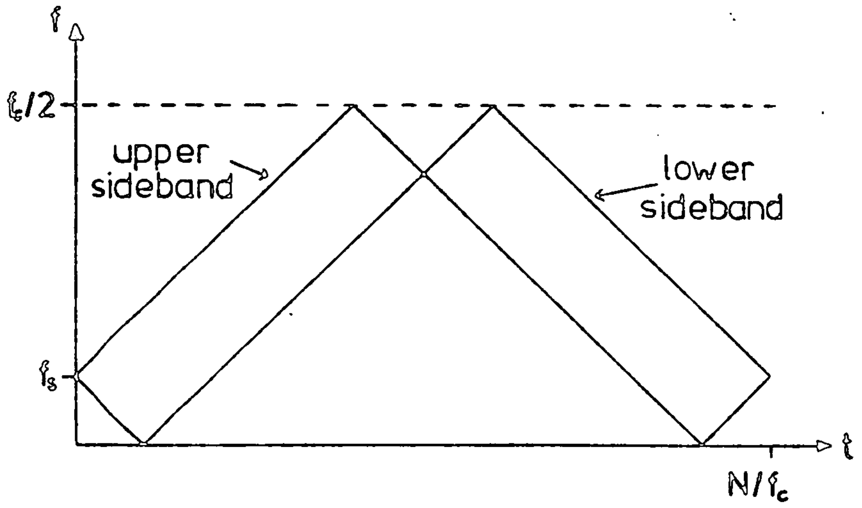


Fig.5.3 Sideband Representation of Pre-multiplication

The complex convolution is illustrated graphically in Fig.5.4. After the pre-multiplied signal has been loaded into the convolver (t=0) , the process of shifting and multiplying begins. At time $t=f_s N/f_c^2$, the frequencies in the upper sidebband of the pre-multiplied input signal match exactly with those in the convolver, and a convolution peak is output. Similarly, at $t=N(1-f_s/f_c)/f_c$, the lower sideband matches. The timing of a convolution peak is thus proportional to input frequency. When the complex convolver is split into four real channels, the individual outputs are shown in Fig.5.5. A mathematical analysis for these outputs

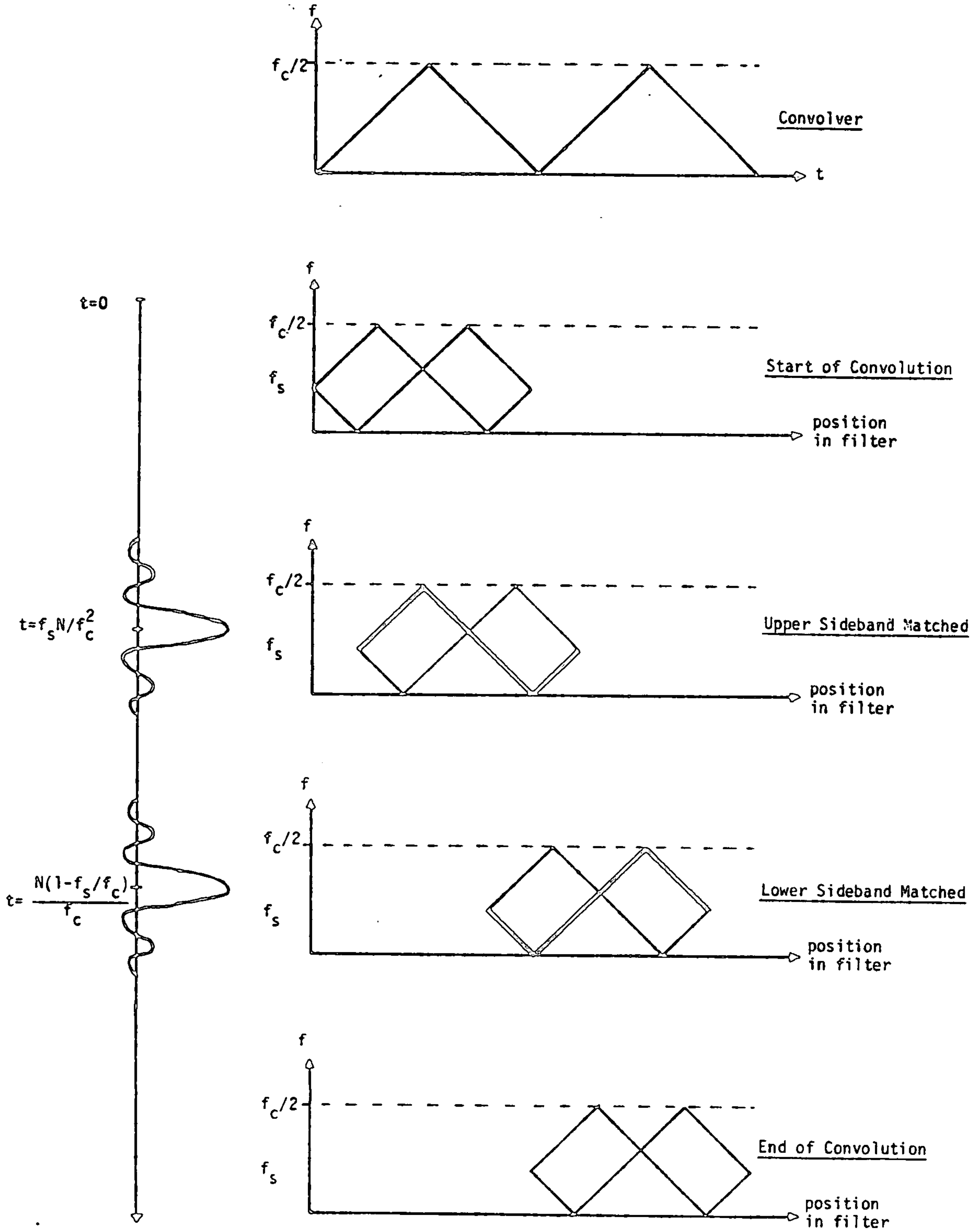


Fig.5.4 Graphical CZT Convolution

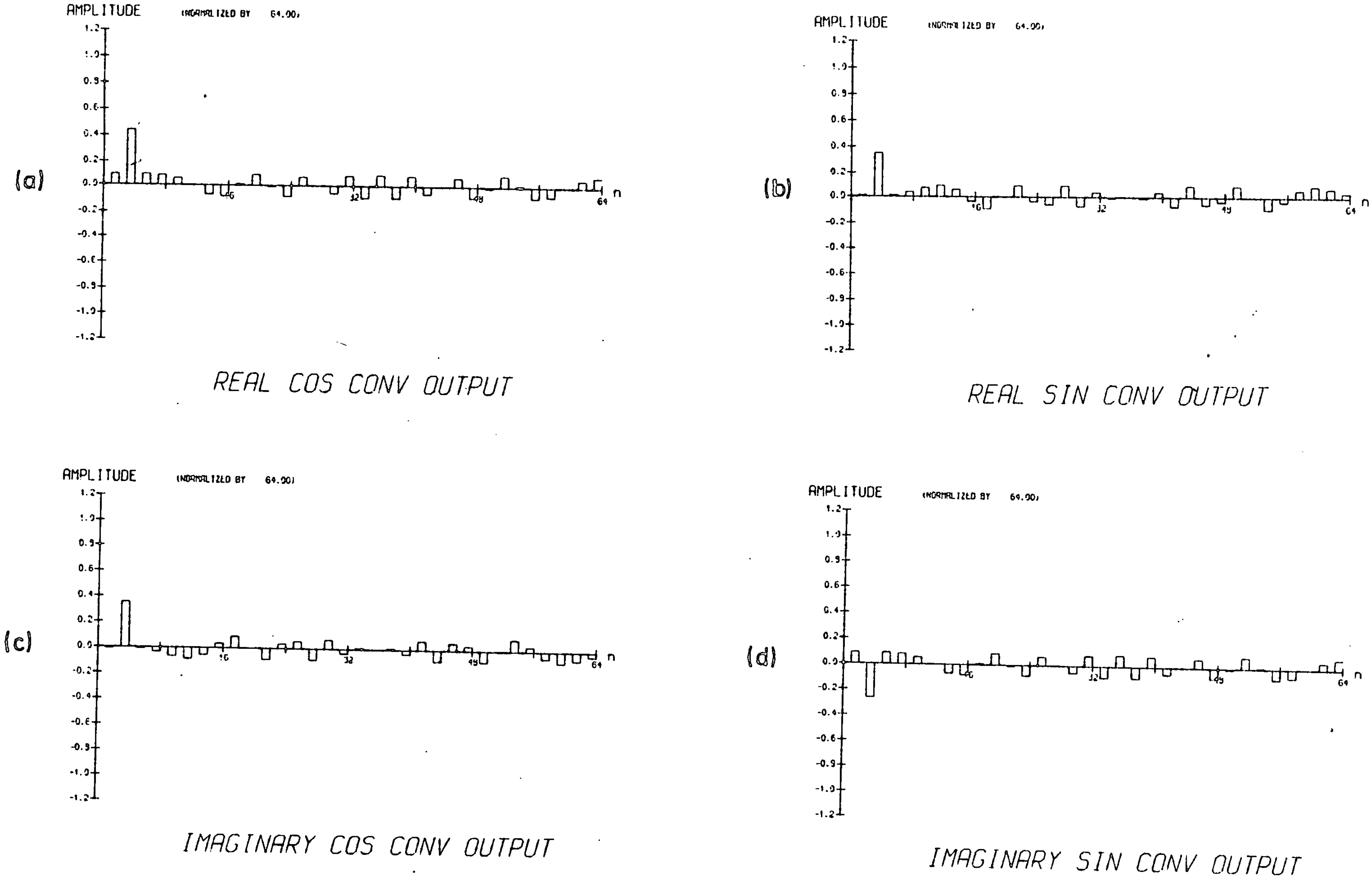


Fig.5.5 Individual Convolution Filter Outputs

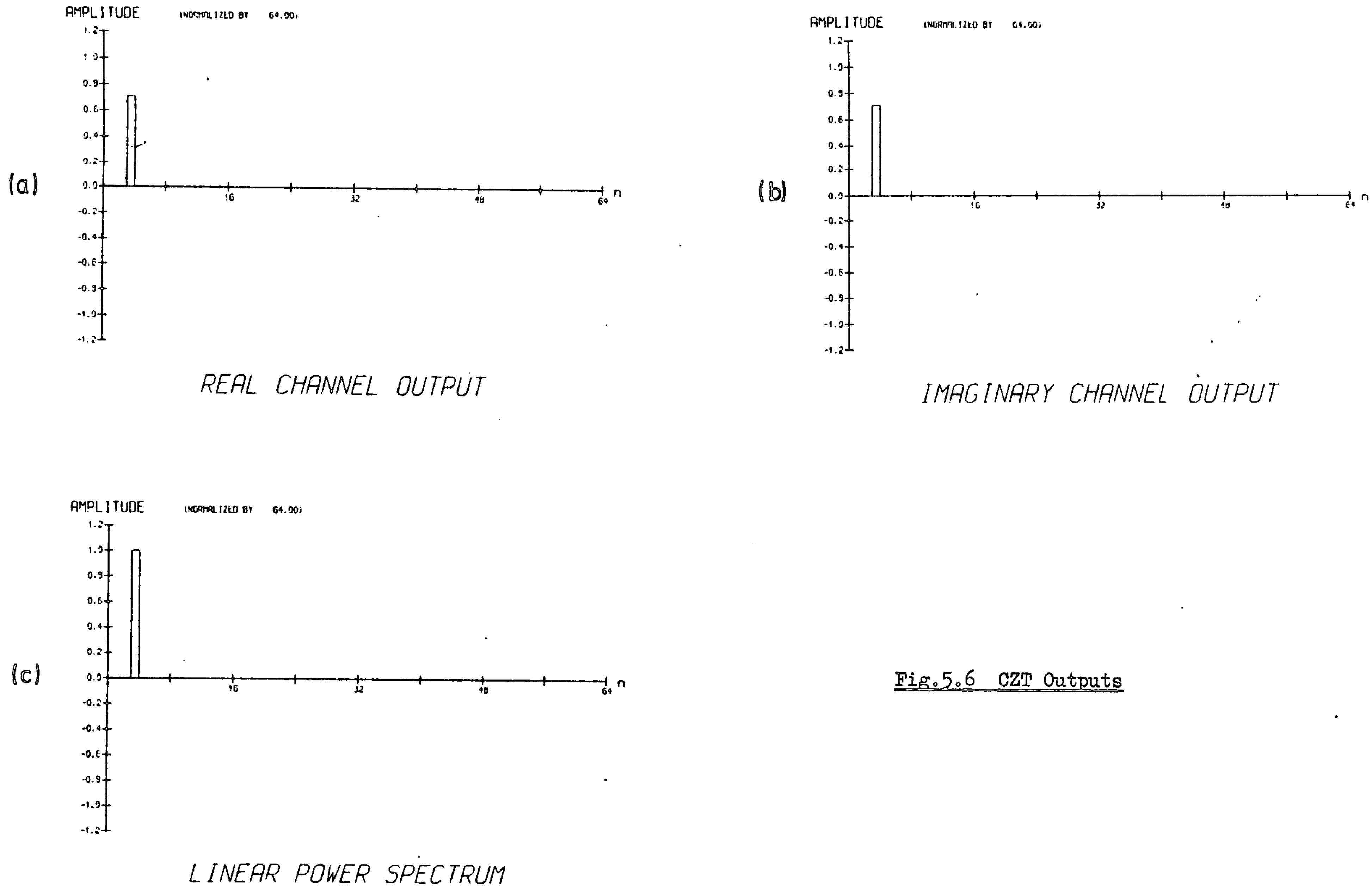
is given in Appendix B. Note that because the input signal is complex, the even and odd properties of COS and SIN combine to cancel one of the convolution peaks. The real and imaginary convolution outputs (after summing the individual filter outputs0 are plotted in Figs.5.6a and 5.6b respectively.

The modulus operation at the CZT's output removes the dependence on input phase giving the power spectrum in Fig.5.6c.

5.2.2 Pre-multiplier Quantisation Errors

In hardware, the pre-multiplying sequences are normally stored in ROM to an accuracy of b bits including sign and the input signal is pre-multiplied in an MDAC. If it is assumed that the MDAC is accurate to within $\frac{1}{2}$ LSB of the pre-multiplying sequence, the pre-multiplication error is due to the level of quantisation.

The computer simulation was set to calculate a 64-point direct CZT and the input chosen as a complex basis vector. The pre-multiplier quantisation was varied from 11 to 6 bits and all other error sources set to zero. A typical output from this simulation is given in Fig.5.7. Here the quantisation is 7 bits and the peak error to signal ratio



REAL CHANNEL OUTPUT

IMAGINARY CHANNEL OUTPUT

LINEAR POWER SPECTRUM

Fig.5.6 CZT Outputs

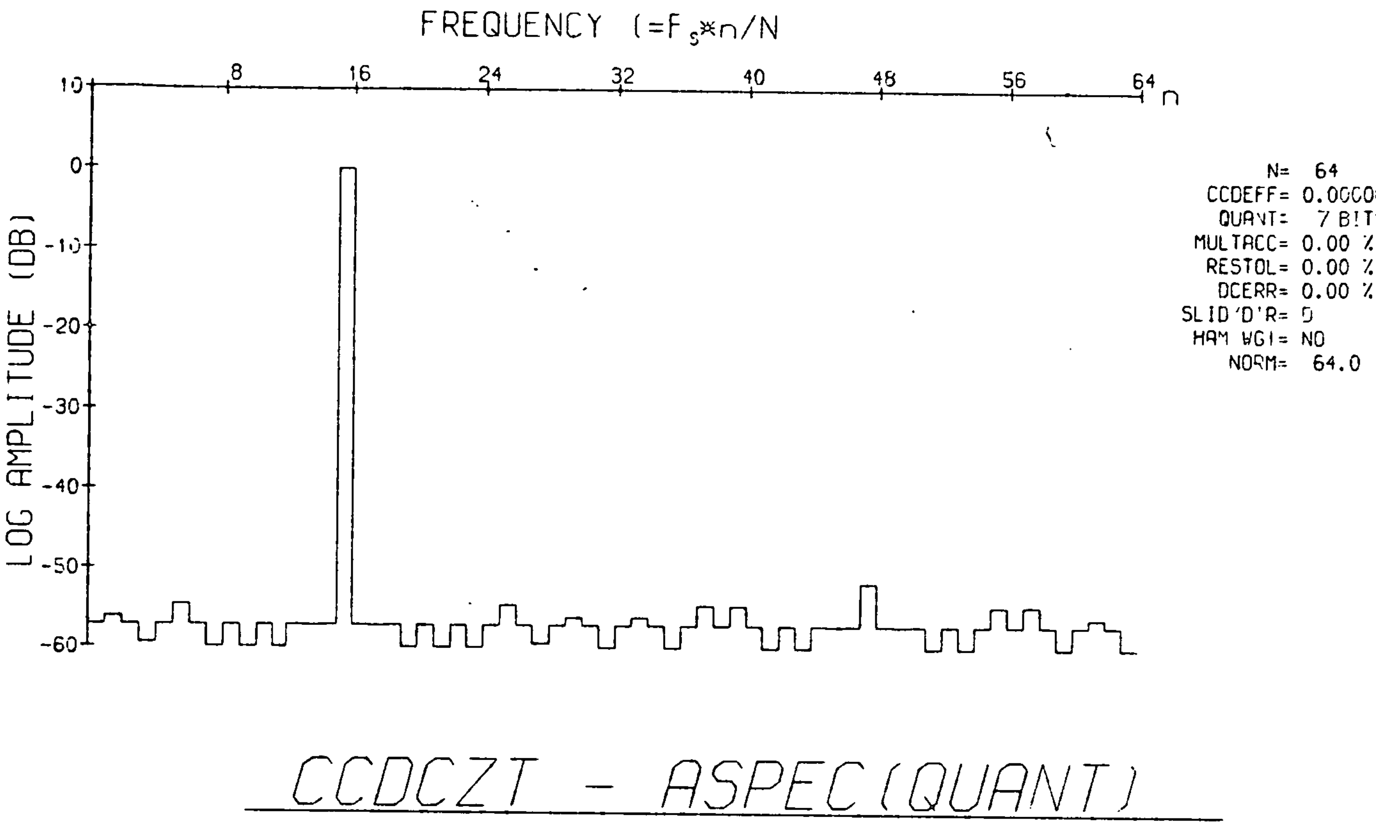


Fig.5.7 Typical Simulation Output - Quantisation Accuracy

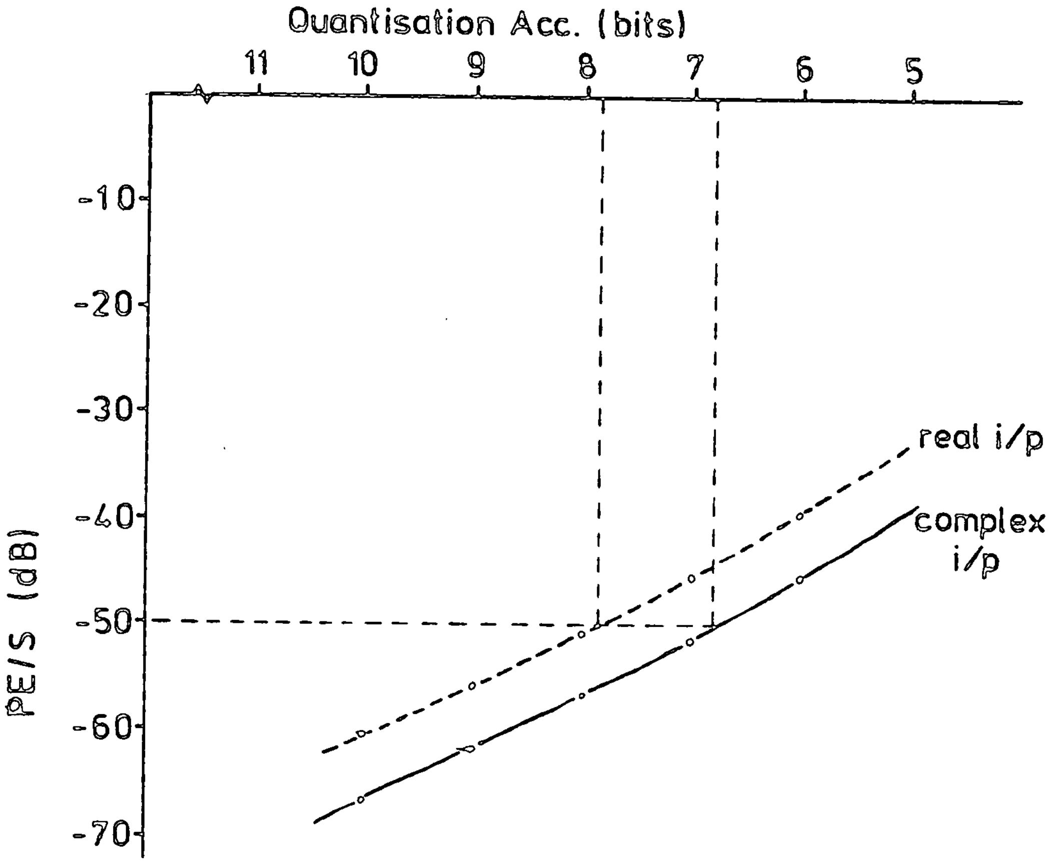


Fig.5.8 Errors Due to Pre-multiplier Quantisation

(PE/S) is -51.7dB. Fig.5.8 shows a graph of quantisation accuracy against the PE/S ratio and, as expected, the PE/S is increased by about 6dB for each quantisation step. Note that for real inputs only, the peak signal is reduced by 6dB (because of the spectral image) and, consequently, the PE/S ratio is degraded similarly. It can be seen that a quantisation accuracy of at least 8 bits (including sign) is required for PE/S ratios of less than -50dB.

5.2.3 CCD Tap Weight Tolerance

The CCDs available for experimental use were floating gate tapped delay lines. This implied the use of external discrete resistors for tap weighting. A percentage tolerance is therefore an appropriate error classification i.e.

$$\text{Tap Weight} = R\ (1 + \frac{\gamma T}{100}) \qquad \ldots\ (5.1)$$

where R is the exact tap weight , γ is a random number (Gaussian distribution) in the range -1 to +1 and T is the percentage tolerance. The tolerance T includes both the CCD tap amplifier gain mismatch and the resistor accuracy. Fig.5.9 shows the simulation results for a 64-point direct CZT with basis vector inputs. Once again, when the input data are real only, there is a 6dB loss in peak signal.

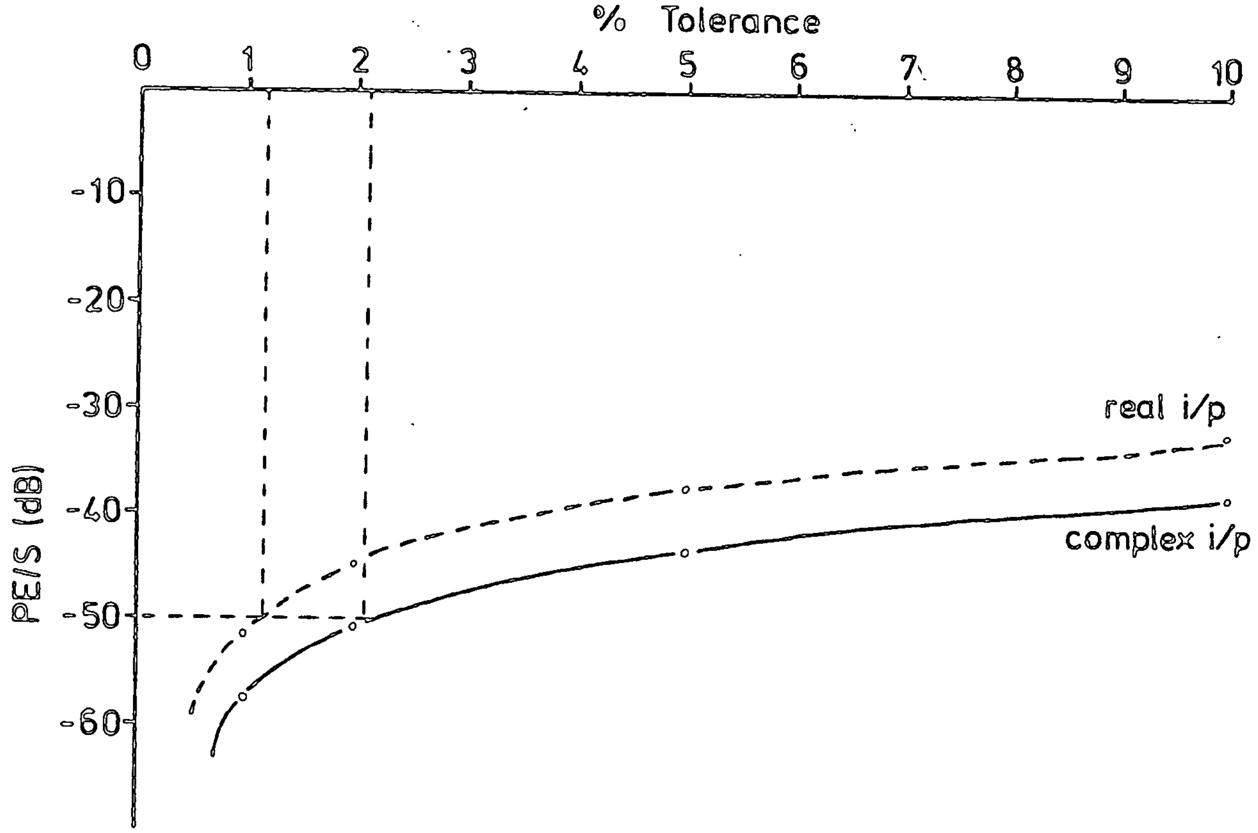


Fig.5.9 CCD Tap Weight Accuracy

These results indicate that tap accuracy should be at least 1% for peak errors of less than -50dB.

5.2.4 Charge Transfer Efficiency

In practical CCD registers, an amount α of the signal packet is transferred, and a fraction ε is left behind. The CZT computer simulation models this mechanism directly in the time domain. Three results are of interest: (a) the variation of output PE/S ratio with ε, (b) the PE/S dependence on input frequency for a fixed ε and (c) the variation of PE/S with N, the number of transform points, for a fixed ε.

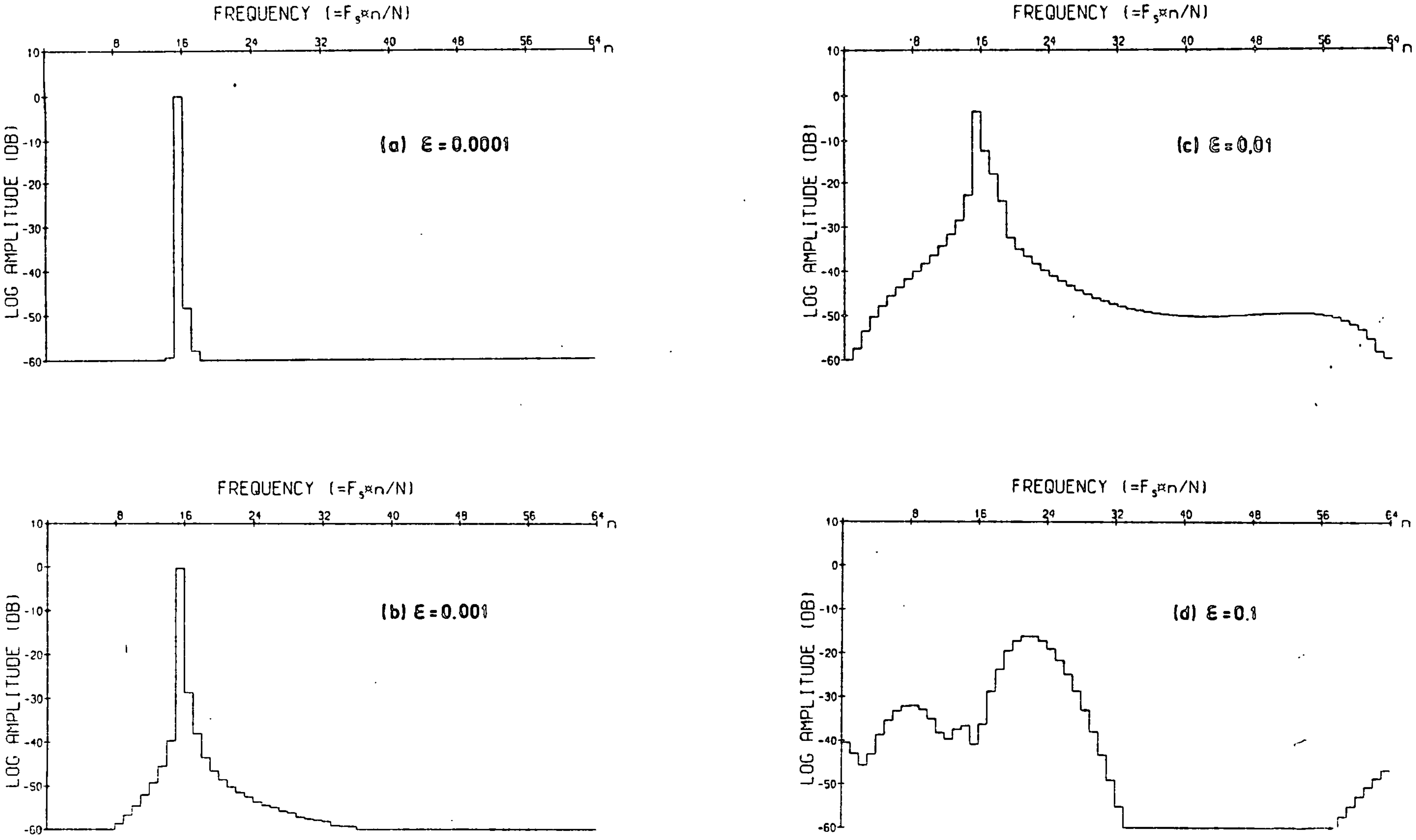


Fig.5.10 Degradation Caused by Charge Transfer Inefficiency

Fig.5.10 shows the progressive degradation in the CZT output as the charge transfer inefficiency, ε, is increased from 0.0001 to 0.1. It can be seen that the effect of ε is to reduce the frequency resolution. The graph in Fig.5.11 gives the relationship between PE/S and ε.

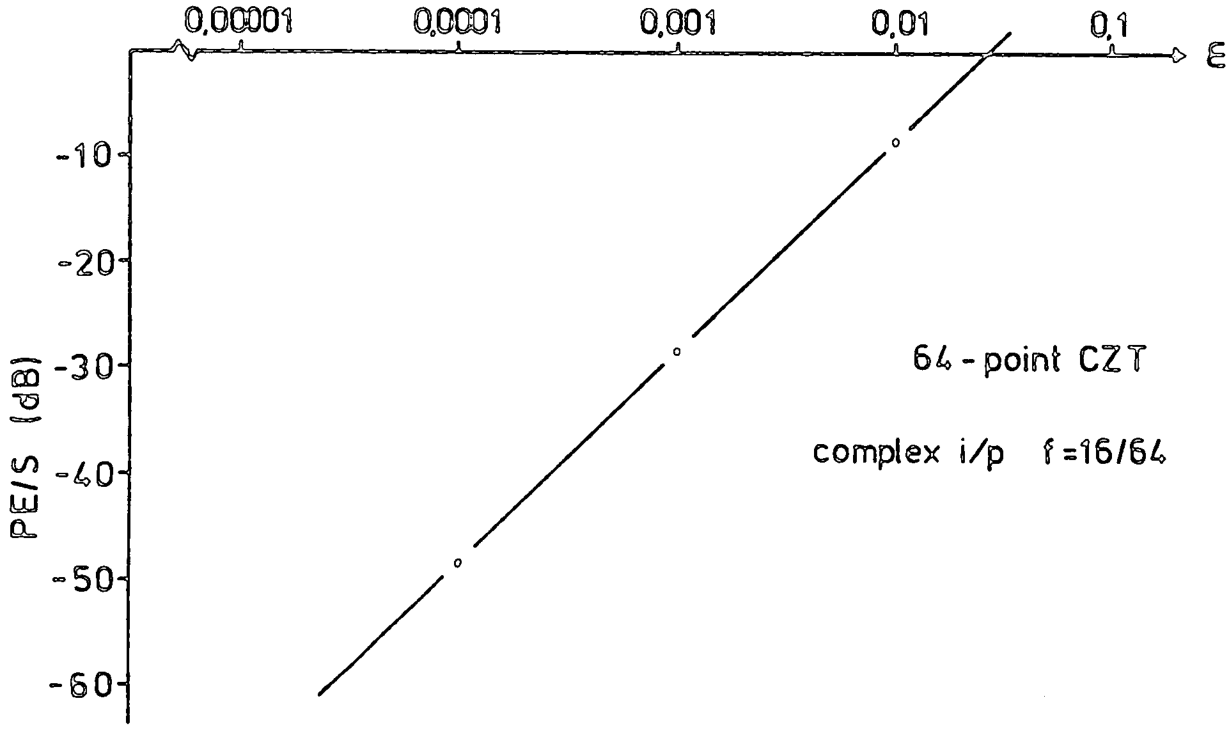


Fig.5.11 The Effect of Charge Transfer Inefficiency

(When the curve crosses the horizontal axis, i.e. PE/S=0dB, the error component becomes larger than the signal component). For the particular CZT parameters in Fig.5.11, a transfer inefficiency of better than 0.0001 is necessary for a PE/S ratio of -50dB.

The amplitude spectrum shown in Fig.5.12 is the result when the CZT is input simultaneously with seven equal amplitude complex basis vectors.

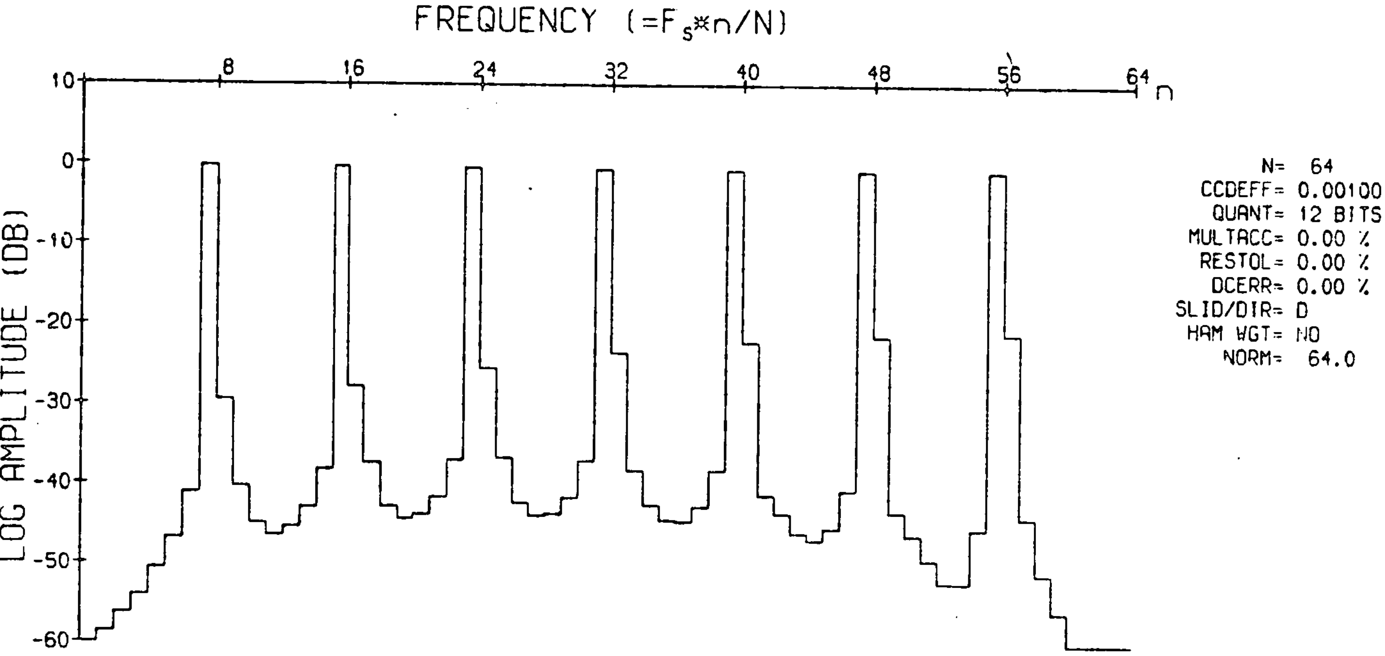


Fig.5.12 Dependence of PE/S on Input Frequency

With increasing input frequency, the pre-multiplied input signal moves further along the CCD register before the appropriate convolution term is produced. The higher frequencies are therefore affected more by ε than the lower frequencies. In the example shown here (N=64 and ε =0.001), the PE/S ratio is 9dBs worse for the higher frequencies.

When the number of transform points is increased, the effect of ε becomes greater because each output point depends on more serial transfers. Fig.5.13 shows the variation of PE/S with N for two different values of ε.

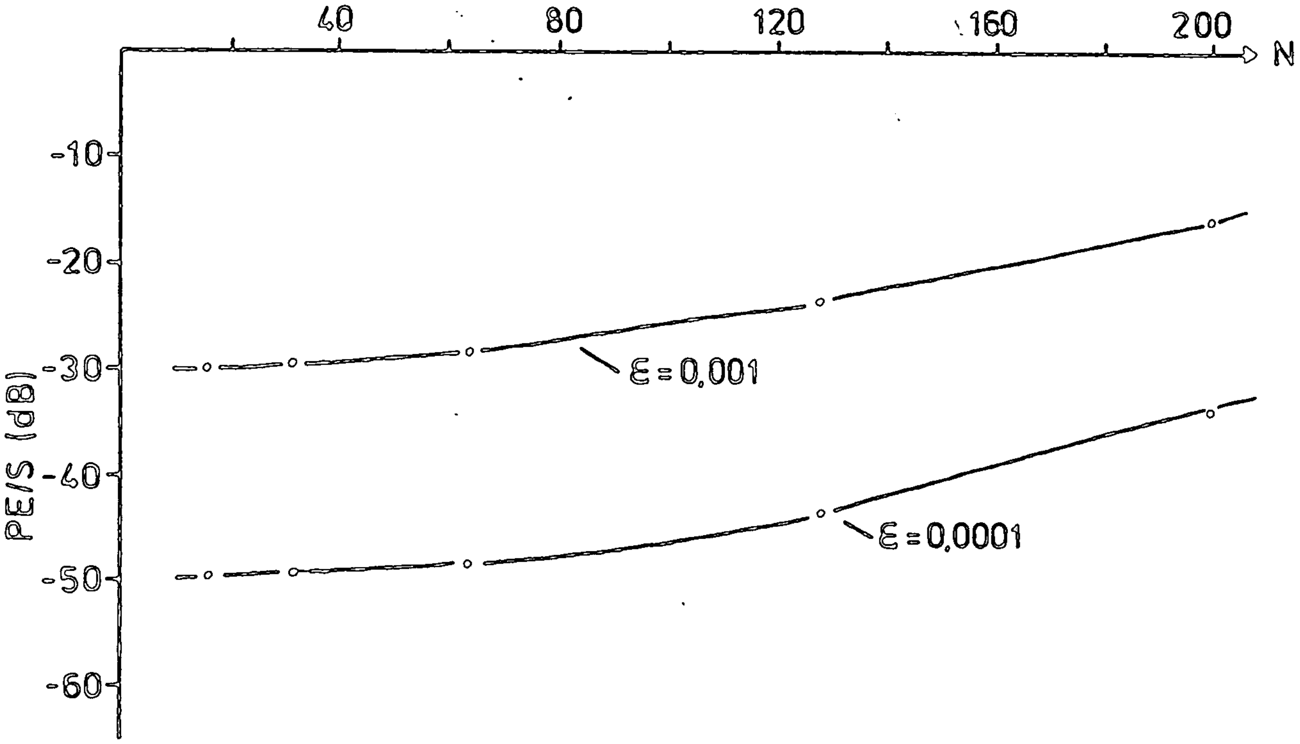


Fig.5.13 Relationship between ε and N

5.2.5 Analogue Multiplier Accuracy

The post-convolver modulus circuitry is normally implemented using analogue transconductance multipliers. Simulation results for random multiplier errors of between 0 and 10% are illustrated in Fig.5.14. As expected, these errors produce results similar to those obtained by the tap weight accuracy simulation and a PE/S ratio of less than -50dB demands 1% multiplier accuracy.

5.2.6 Phase Shifter Errors

Stictly speaking, the 90-degree phase shifter (section 5.3.5) is not part of the CZT processor. Nevertheless, it

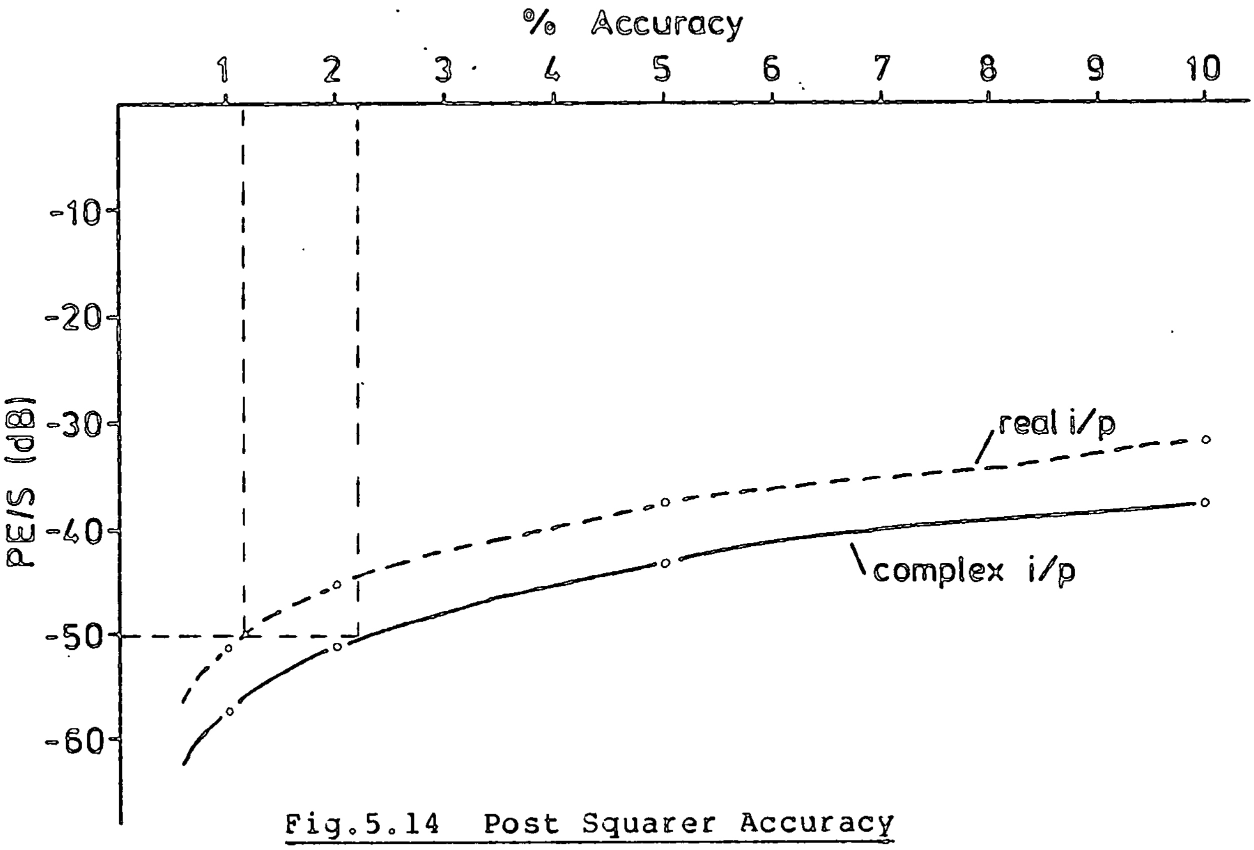


Fig.5.14 Post Squarer Accuracy

is necessary to investigate the accuracy required by this peripheral unit in order to suppress image frequencies to any desired level. In this simulation, the ideal CZT was supplied with real and imaginary inputs of the form

$$x_R = A\cos(2\pi kn/N) \qquad \ldots (5.2)$$

$$x_I = A\sin(2\pi kn/N + \epsilon) \qquad \ldots (5.3)$$

where ϵ is the phase shift error and is in the range 0 to 5°.

The output for a 2° error is shown in Fig.5.15 where the image frequency is clearly visible. The relationship between PE/S and phase shift error, ϵ, is plotted in

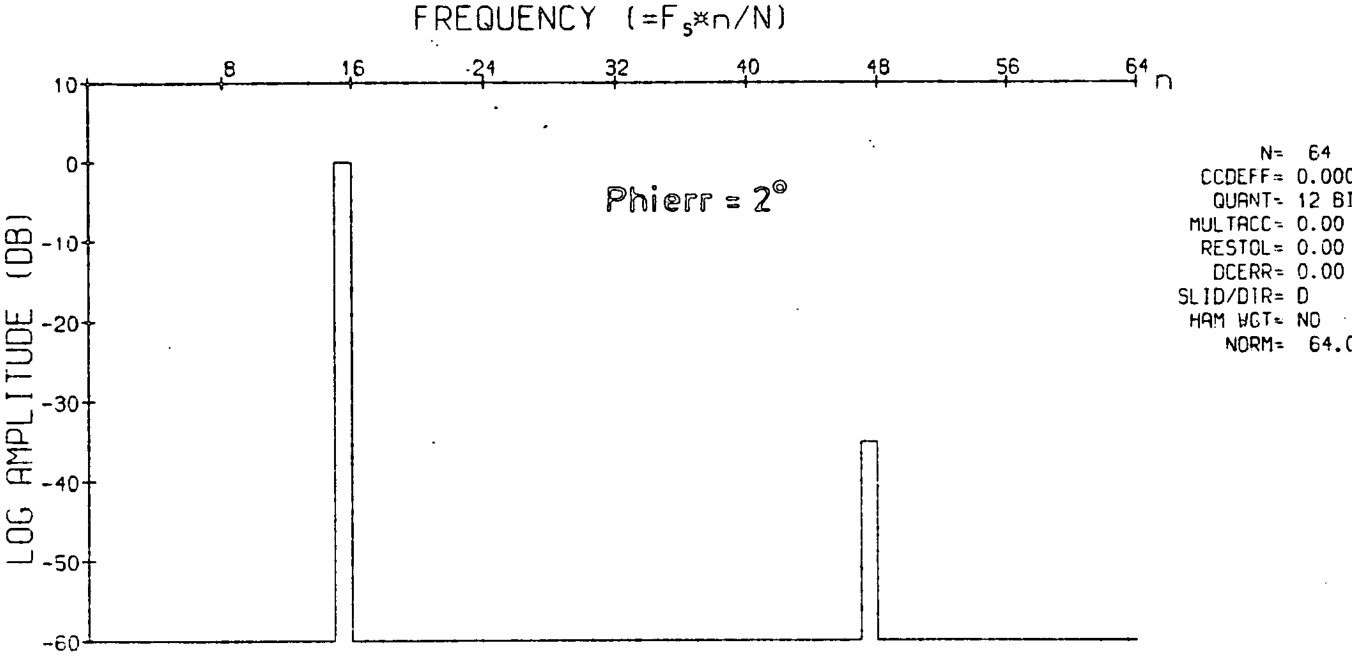


Fig.5.15 Image Frequency Suppression

Fig.5.16 which shows that the phase shifter has to be accurate to about 1° for an image supression of -40dB.

5.2.7 Summary of Simulation Results

The simulations discussed so far consider the effect of each error source independently; in the practical situation, all sources combine to give an increased PE/S ratio. However, to compare the relative significance of each source, it is worthwhile summarising these results (Table 5.1).

To predict the accuracy of a practical CZT processor, tne estimated error tolerances for typical components (see

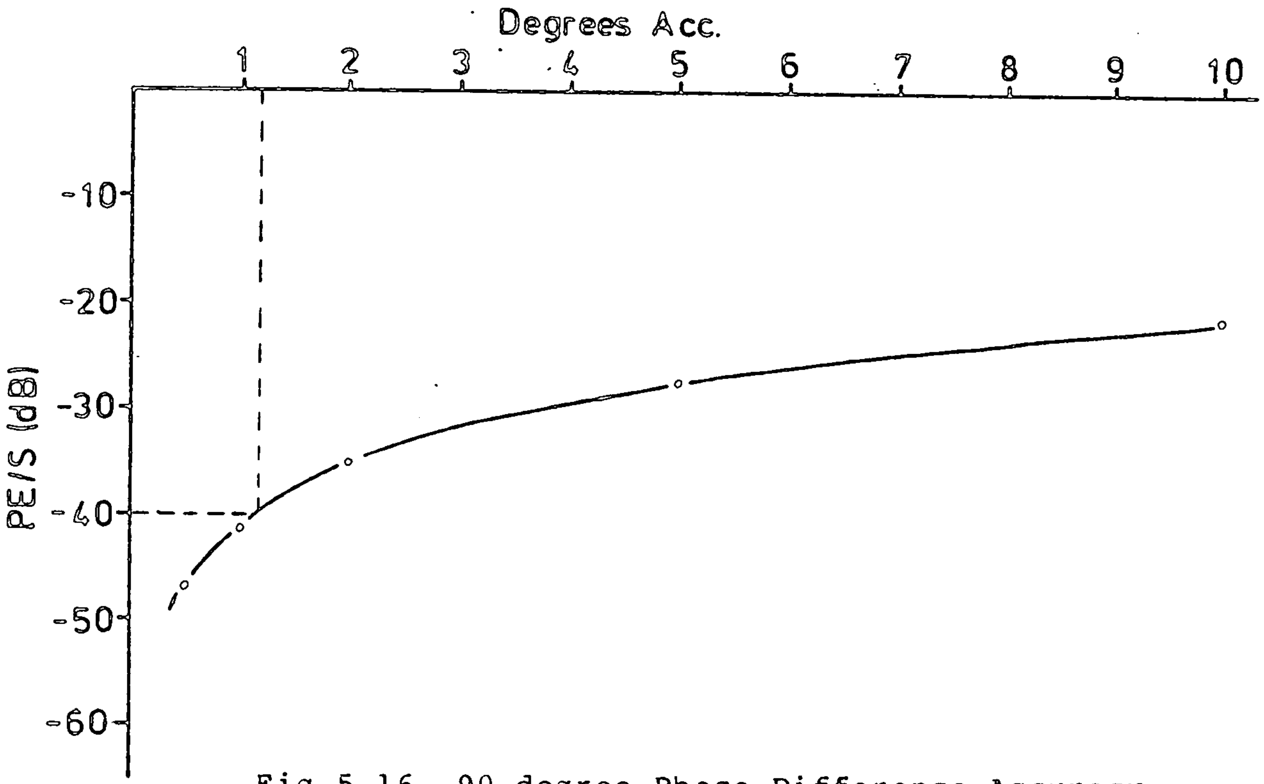


Fig.5.16 90-degree Phase Difference Accuracy

	Accuracy to achieve -50dB PE/S ratio (Simulation)	Typical Practical Component Values
Pre-multiplier Quantisation	8	8
Tap Weight Tolerance	1%	2%
Charge Transfer Inefficiency	0.0001	0.0001
Post Squarer Accuracy	1%	5%
Phase Shifter Error	½°	1°

Table 5.1 Error Analysis Summary for a 64-point direct CZT

Table 5.1) were used in the CZT simulation. For a basis vector input the PE/S ratio is -42.7dB. The amplitude spectrum resulting from a square wave input is shown in Fig.5.17.

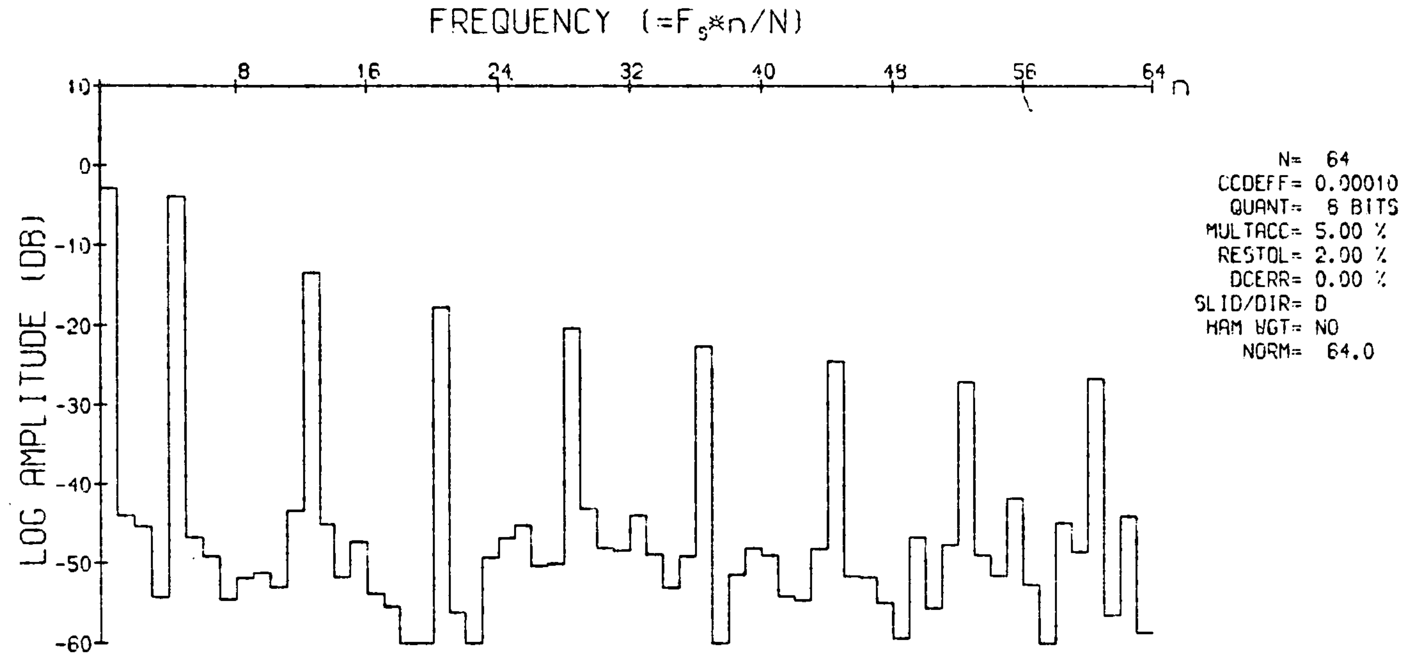


Fig.5.17 Simulation Output for Square Wave Input (practical errors)

The peak spurious error is at -42dB and the largest harmonic amplitude error is 1.3dB in the 13th harmonic.

5.3 IMPLEMENTATION

This section describes in detail the CZT hardware and is split into four subsections dealing with pre-multiplication, convolution, post-multiplication and timing. In addition, two CZT peripherals specifically designed for speech processing are discussed: (a) a 90-degree phase difference network and (b) a low-pass filter.

5.3.1 The Pre-multiplier

Two different pre-multiplication techniques exist. The first employs four-quadrant analogue transconductance multipliers, the pre-multiplying chirps being generated either actively or by impulsing CCD chirp transversal filters. The second uses MDACs and the pre-multiplying sequences are stored in ROM. Although the difference in speed between these two approaches is insignificant, the second technique has been adopted here because digital processing offers increased stability and flexibility over its analogue equivalent.

Fig.5.18 illustrates the complex pre-multiplication schematic. The circuit employs four low cost, monolithic MDACs allowing up to 10 bit accuracy (only 8 bits are used). These devices are fabricated using a combination of Complementary MOS (CMOS) and thin film technologies to give a power dissipation of only 20mW and a current settling time of 500nS. Because the reference (signal) input has bipolar capability, four-quadrant multiplication is achieved by providing offset binary at the digital input. Linearity measurements on both the digital and analogue inputs showed

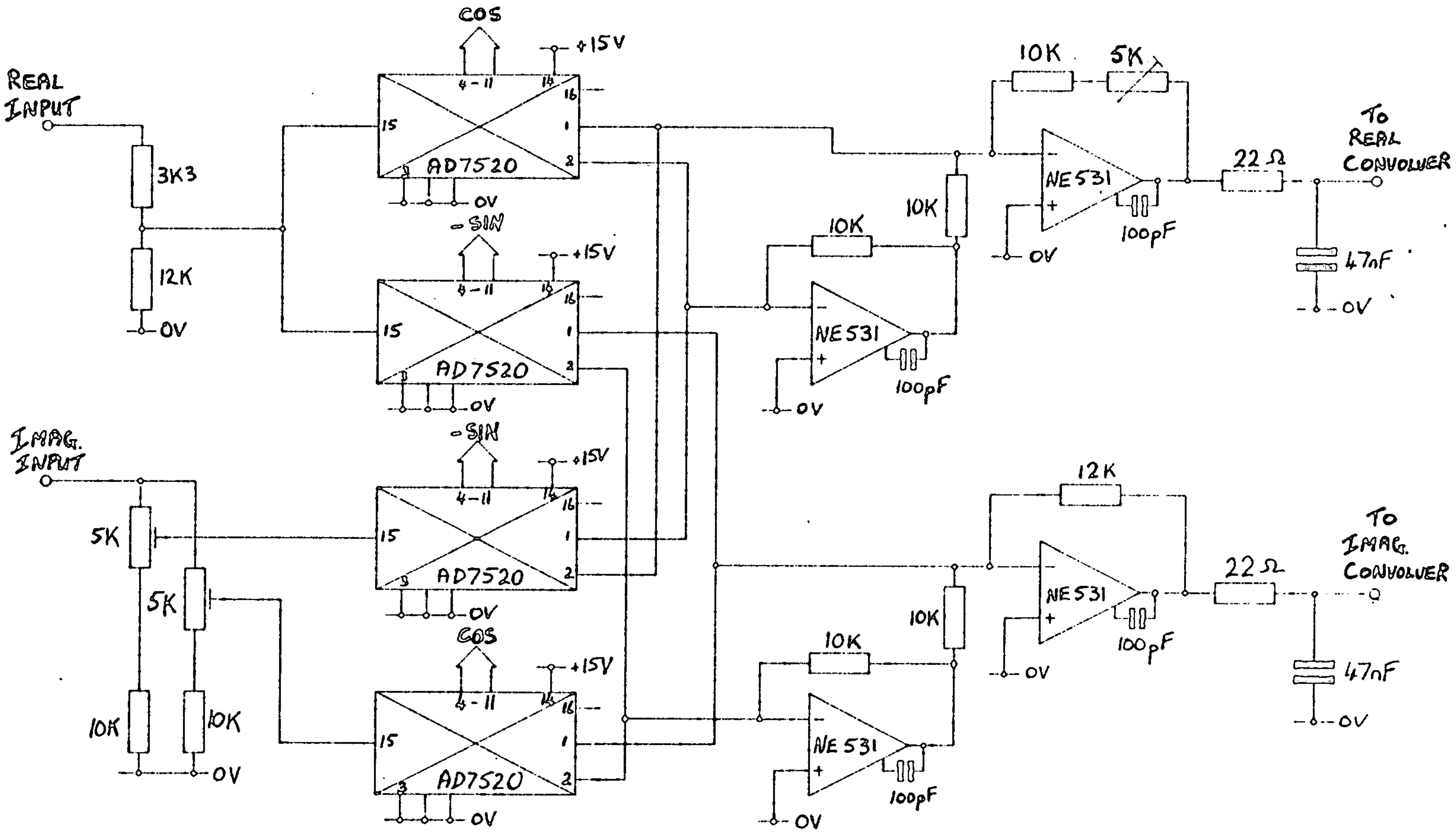


Fig.5.18 Hardware Schematic for Complex Pre-multiplication

the non-linearity to be 0.25% of full scale output (better than -52dB). Normally, on-chip feedback resistors are used in conjunction with external amplifiers to define accurately the gain of each MDAC. However, to minimise the total number of amplifiers required in the quad configuration, the MDAC current summers are combined, thereby necessitating three variable gain controls to equalise the circuit.

In the direct CZT algorithm, the convolvers have to be serially loaded with N pre-multiplied data points followed by N-1 zeroes. This blanking operation may be conveniently incorporated by extending the pre-multiplying sequence to 2N-1 samples, the last N-1 being zeroes. A further simplification in hardware timing results if, instead of N-1 zeroes, N zeroes are loaded. The timing for a 32-point direct CZT then becomes identical to that for a 64-point sliding CZT. In the direct case, the extra input zero simply means that one extra output point has to be discarded.

The pre-multiplying sequences are stored in four 32x8 bit bipolar ROMs which are used in pairs to form two 64x8 bit offset binary chirp sequences. The ROM addresses are supplied by a 6 bit synchronous binary counter (Fig.5.19).

Three different sets of ROMs were programmed to provide alternative CZT configurations. The sequences are defined by

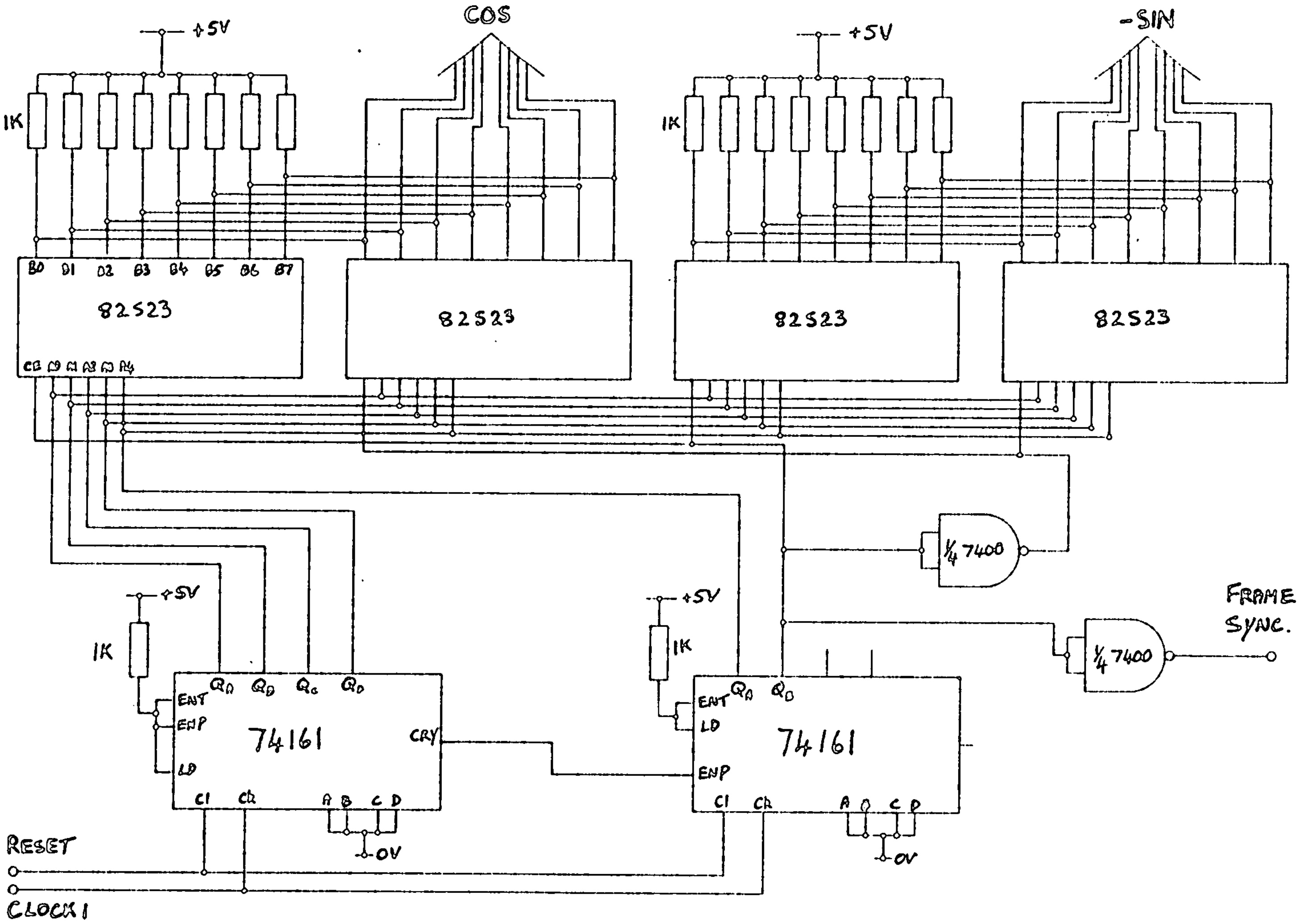


Fig.5.19 Pre-multiplier Sequence Storage

1. 32-Point Direct CZT (rectangular window)

$$C1_n = \cos(\pi n^2/32) \qquad S1_n = \sin(\pi n^2/32) \qquad n=0,1..31$$
$$C1_n = 0 \qquad S1_n = 0 \qquad n=32,33..63 \qquad \ldots (5.4)$$

2. 32-Point Direct CZT (Hamming window)

$$C2_n = (0.54 - 0.46\cos(\pi n/32))\, C1_n \qquad n=0,1..63$$
$$S2_n = (0.54 - 0.46\cos(\pi n/32))\, S1_n \qquad n=0,1..63 \qquad \ldots (5.5)$$

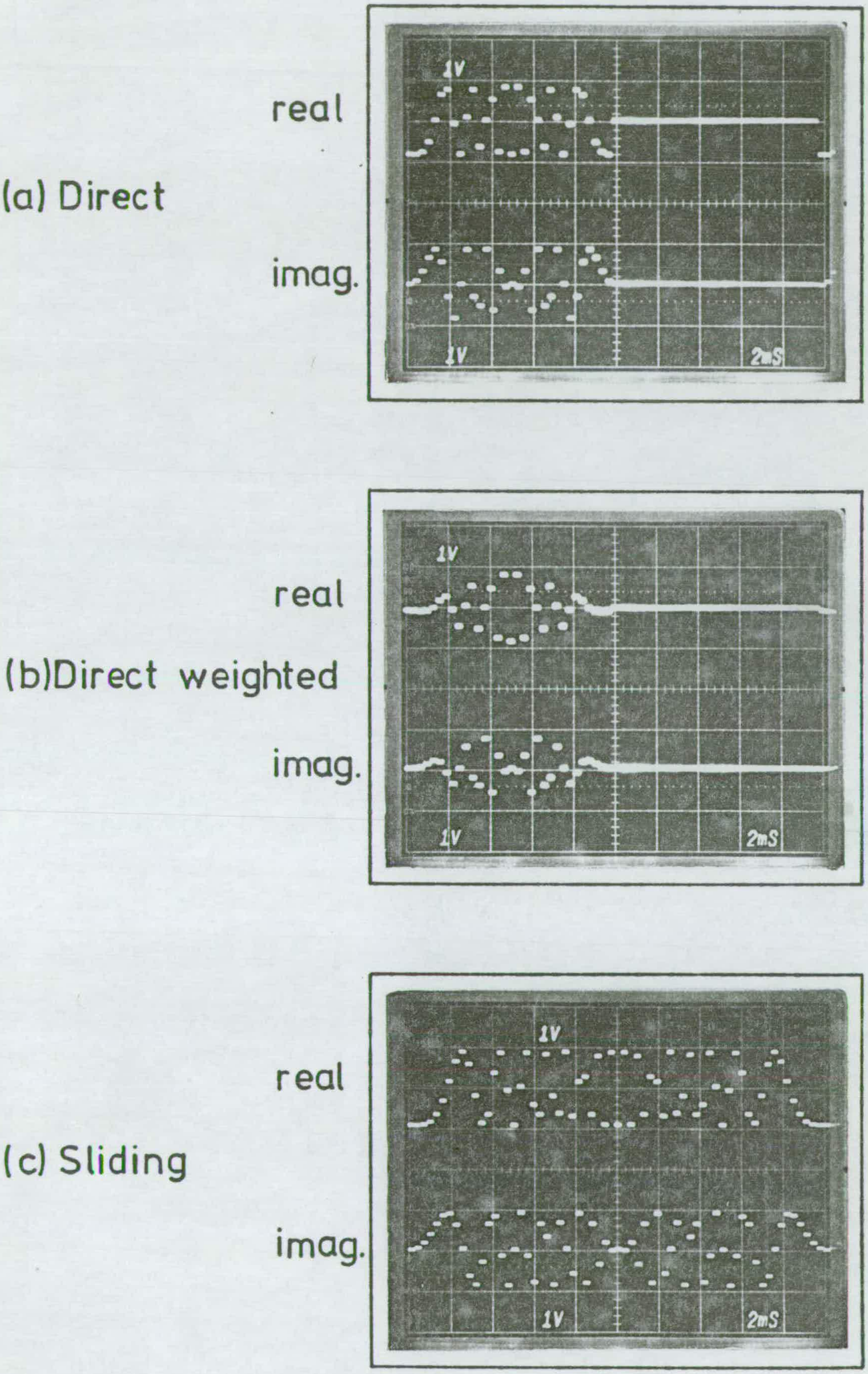
3. 64-Point Sliding CZT

$$C3_n = \cos(\pi n^2/64) \qquad n=0,1..63$$
$$S3_n = \sin(\pi n^2/64) \qquad n=0,1..63 \qquad \ldots (5.6)$$

The pre-multiplier operation for each of the above cases is shown in Fig.5.20. Here, the real input is dc and the imaginary input is grounded.

5.3.2 The Convolver

In-house 32-tap CCD delay lines [56] were available for prototype construction. These devices were fabricated using an "n" channel, aluminium gate process and were designed for three phase operation. The CCD registers have 64 serial stages, tapped every alternate stage [63] to make room for



(a) Direct

(b)Direct weighted

(c) Sliding

Fig.5.20 Pre-multiplier Waveforms

the peripheral floating-gate-reset cicuitry (Fig.3.5d).

The CZT convolver design (Fig.5.21) employs discrete resistors and two CCDs in series to implement two 64 point transversal filters. Since the convolvers in both the real and imaginary channels have the same inputs, only one set of CCDs and two sets of resistor weights are required for each of the channels. The resistor values are calculated from the appropriate impulse responses and normalised to reduce loading effects. The CCDs are operated with a diode cut-off input technique similar to that described in section 3.2.1. In this case, the input gate pulse is derived from the timing logic (section 5.3.4) and is arranged to sample the input signal during ϕ_3. The sample is then temporarily stored under an extra floating gate before being transferred to the ϕ_0 potential well. In the circuit of Fig.5.21, external diodes are paralleled with the CCD input diffusions to protect against the accidental application of negative voltages which could damage the devices.

For optimum charge transfer and charge handling capacity, clock amplitudes in the region of 30V are required. The clocking waveforms are produced by transistor buffers described in section 5.3.4. The floating gate structures (Fig.3.5d) are reset to the Vgg potential during ϕ_0 (section 3.2.2).

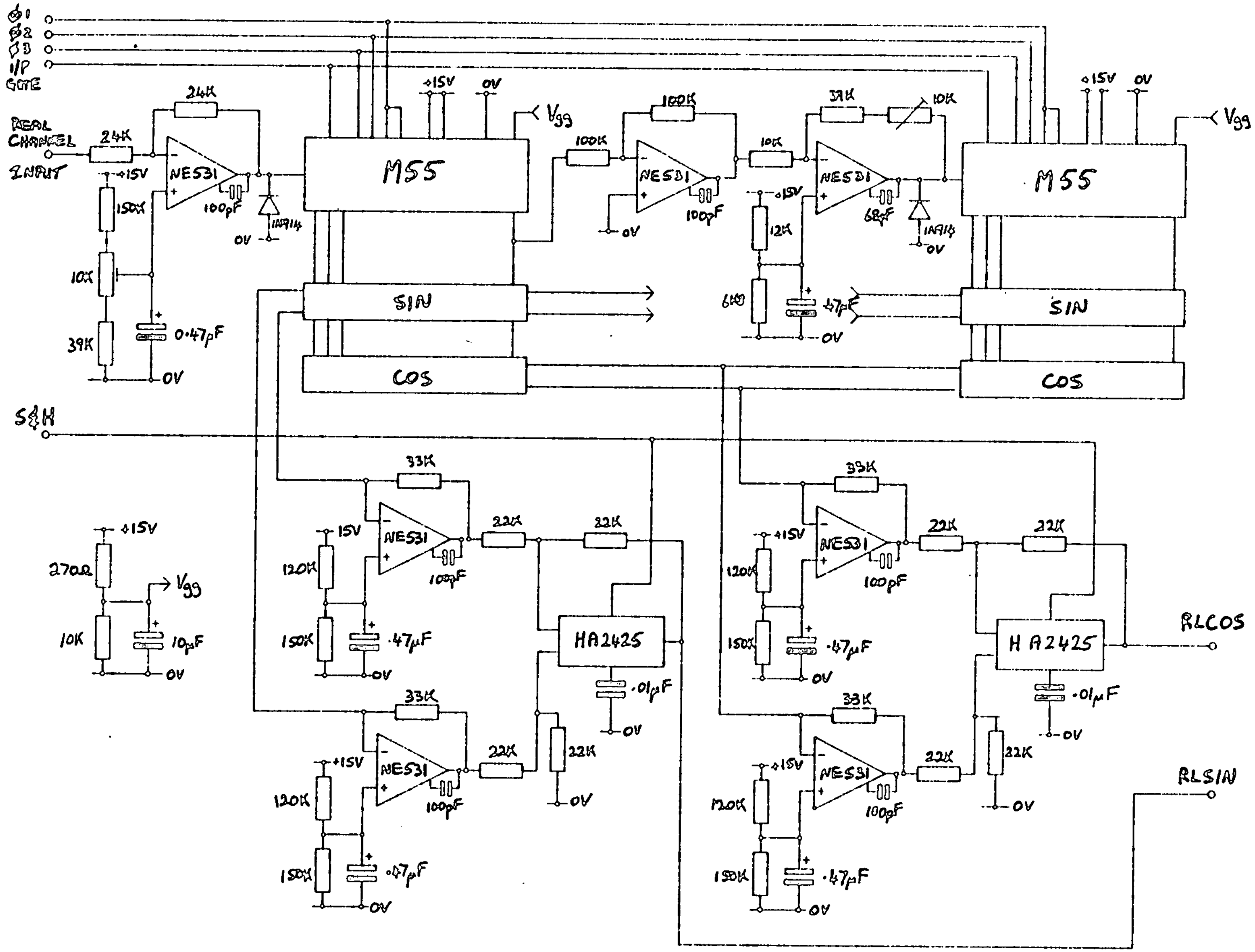


Fig 5.21 Real or Imaginary Channel of Convolver

The currents in each transversal filter busbar are summed into separate virtual earth amplifiers before being differenced. Fast slew rate amplifiers are necessary here in order to cope with the CCD clock breakthrough. The positive and negative busbars cannot be differenced in one operation, thereby eliminating the CCD breakthrough (and hence the slew rate problem), because the parallel impedance of each set of weights is different. Sample and hold circuits subsequently remove the CCD clock breakthrough and provide a stable waveform for post-processing.

Finally, two circuits of Fig.5.21 are combined to provide the full complex convolver in Fig.5.22. Offset controls are provided on the output of each convolver channel so that dc pedestals may be removed before the modulus circuit (section 5.3.3).

The photographs in Fig.5.23 show the outputs from the four convolvers when the processor is configured for a 64-point sliding CZT. These waveforms may be compared to the computer simulation in Fig.5.5 and the mathematical analysis in Appendix B.

5.3.3 Post Circuitry

Since only power spectra are required, the post convolver processing is reduced to a modulus function

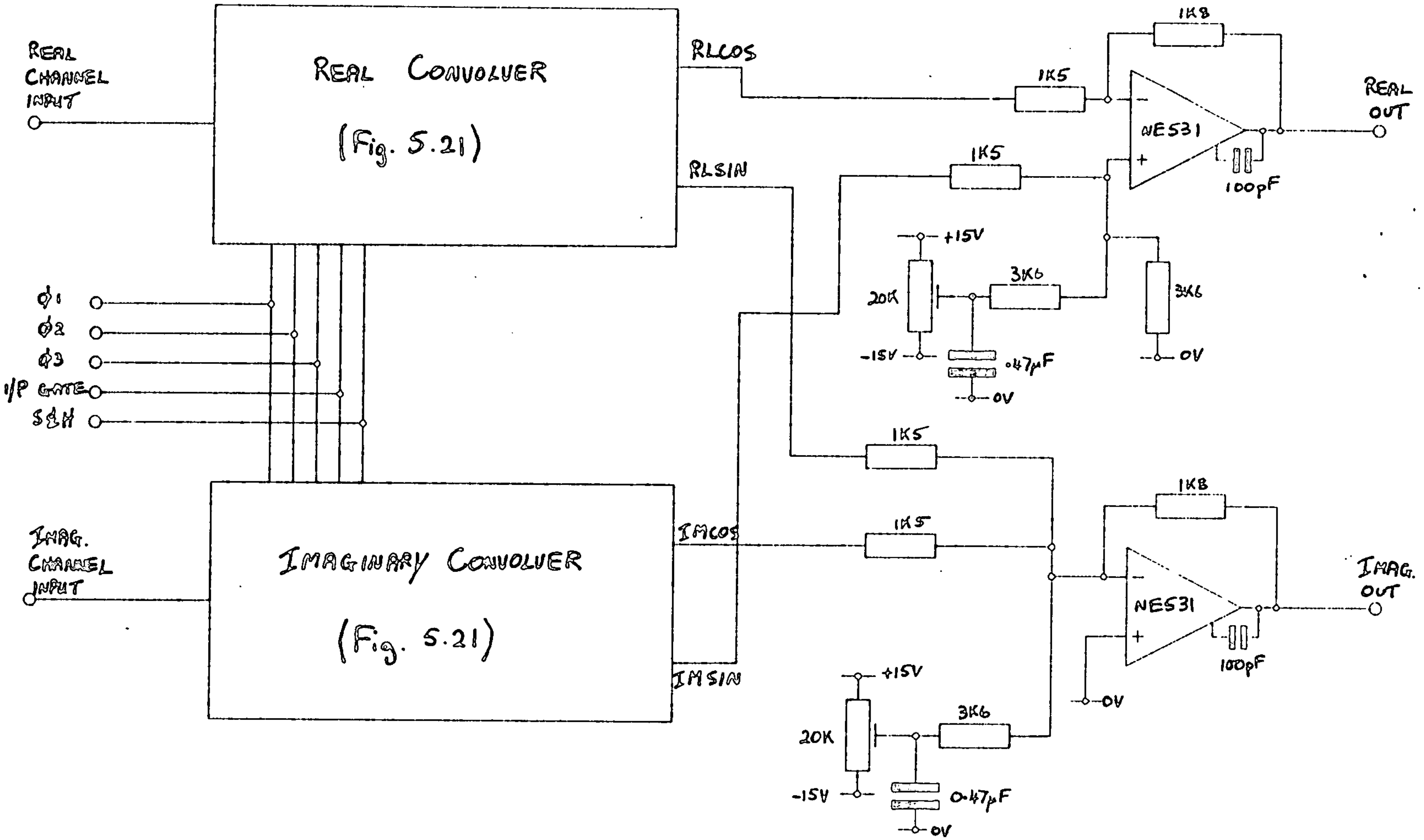
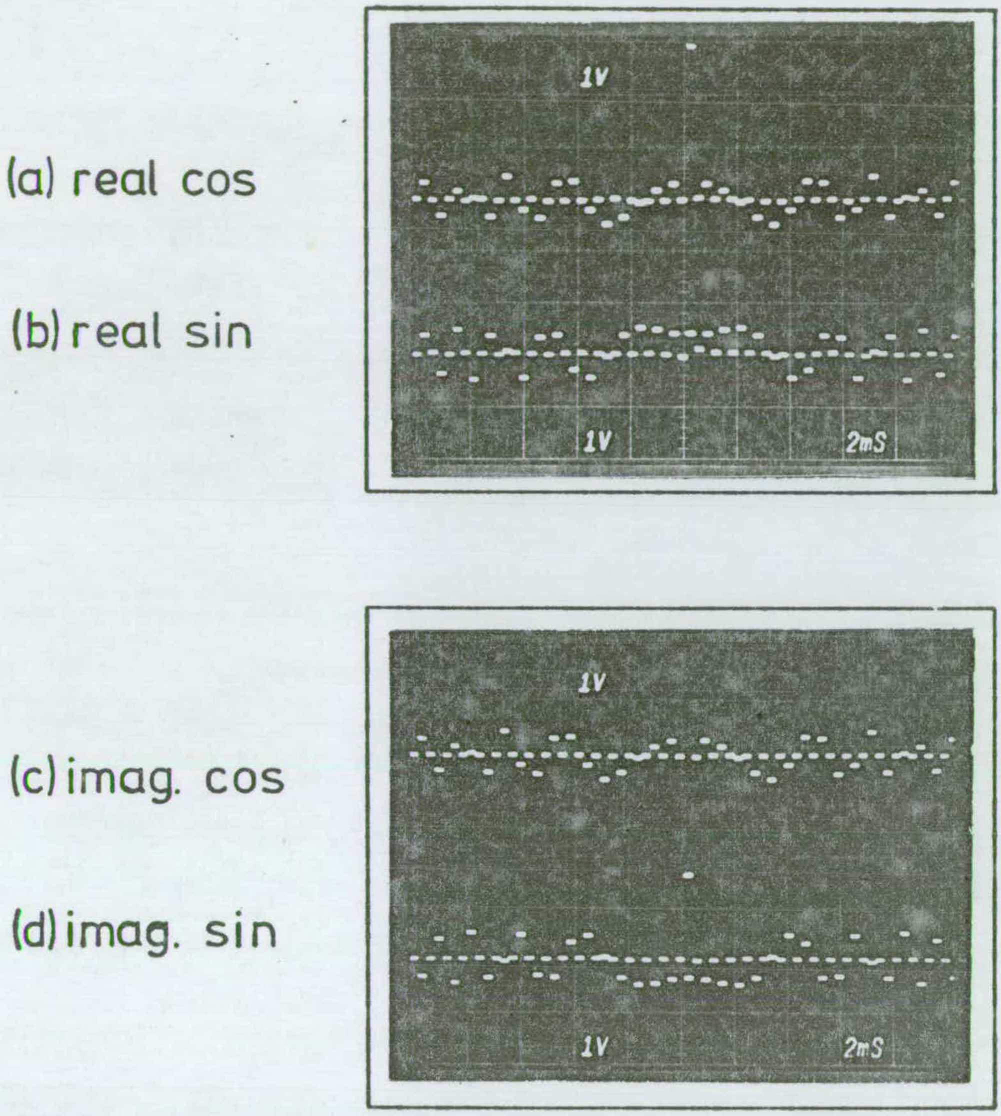


Fig.5.22 Fully Complex Convolver



[CZT inputs -- Re = Im = dc]

Fig.5.23 64-point SCZT Convolver Outputs

(Fig.4.10) i.e.

$$y = \sqrt{R^2 + I^2} \qquad \ldots (5.7)$$

where y is the CZT output, R is the real convolver output and I is the imaginary convolver output. The obvious circuit solution is to employ analogue transconductance multipliers configured as squarers. In the schematic shown in Fig.5.24, two AD533 bipolar multipliers, which have a full power bandwidth of 750kHz, and one summing amplifier perform the square and add operation. A square-rooter has oeen added as an optional extra. The main disadvantage of this implementation is the limited linear dynamic range. Typically, analogue multipliers have a 60dB output dynamic range which implies 30dB at their inputs when the multipliers are used as squarers. This figure does not allow the full potential of the CZT to be exploited.

An alternative approach to the direct implementation of the modulus function is to use a linear approximation. It has been shown [80] that the approximation

$$\begin{aligned} y &= |R| + \alpha|I| \qquad \text{if } |R| > |I| \\ y &= \alpha|R| + |I| \qquad \text{if } |I| \leqslant |R| \end{aligned} \qquad \ldots (5.8)$$

gives an answer to within 0.5dB of the exact value when α=0.409. Such an approximation is generally acceptable.

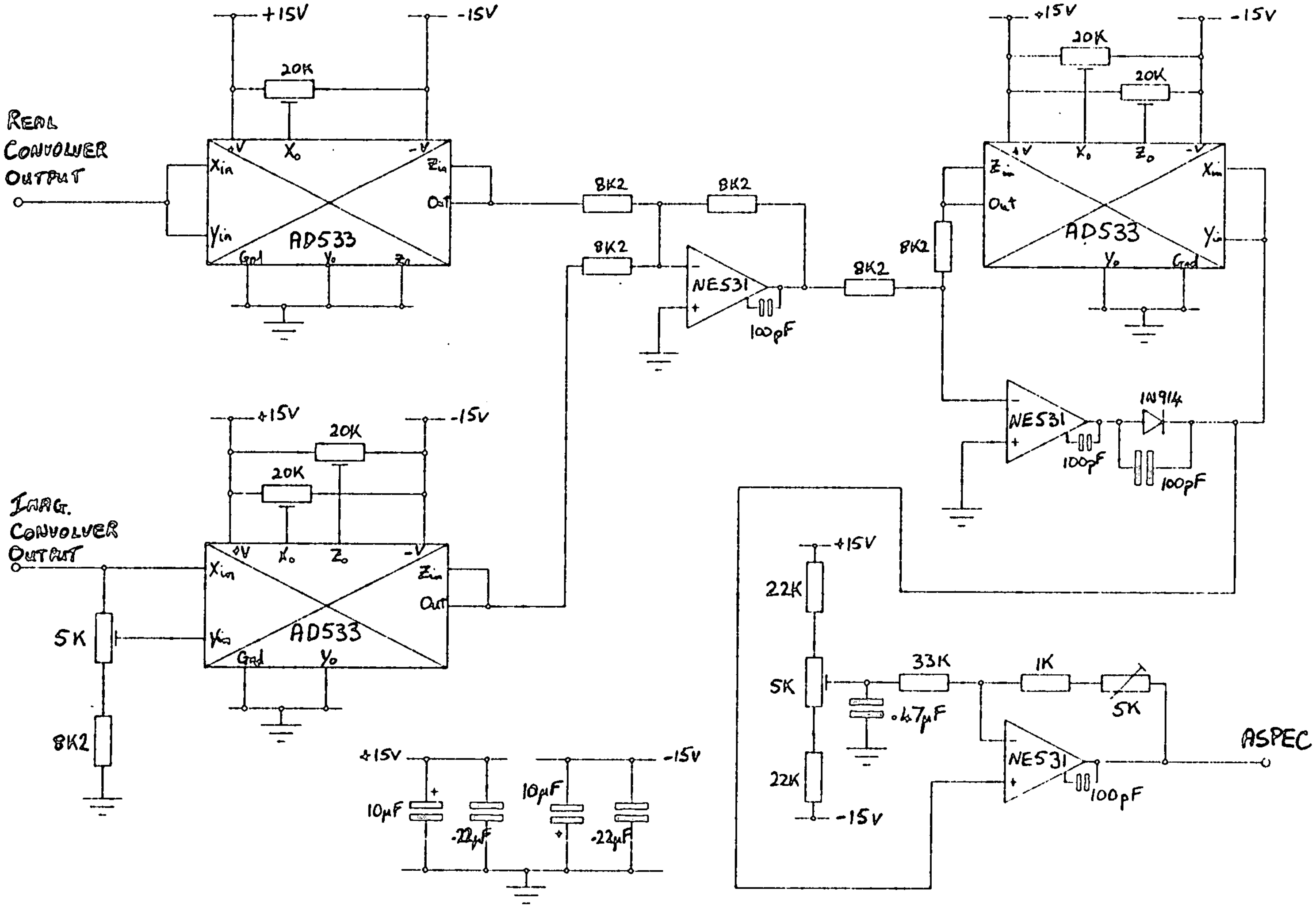


Fig.5.24 Analogue Multiplier Modulus Function

The advantage is that this approximation does not require analogue multipliers and the dynamic range is well in excess of 40dB.

Fig.5.25 gives the circuit diagram. The inputs R and I are full-wave rectified before being combined in the appropriate ratios. A comparator makes the decision as to whether |R| is greater than |I| and the result is used to generate complementary control signals for two analogue switches. A small amount of hysteresis is applied in the decision algorithm to prevent random switching by noise. A third analogue switch is included in the feedback loop of the summing amplifier to equalise the gain defining resistors. The speed of this circuit is limited by the full-wave rectifier to about 200kHz.

5.3.4 Timing

The timing circuitry (Fig.5.26) is designed to accept a master clock and generate various sub-clocks, as well as the appropriate CCD driving waveforms.

The main timing information is derived from two ring-of-three counters connected in series. This allows each phase of the 3-phase CCD clock to be divided into three segments. The CCD clock waveforms, ϕ_1 , ϕ_2 and ϕ_3 are generated by inverting transistor drivers supplied by TTL

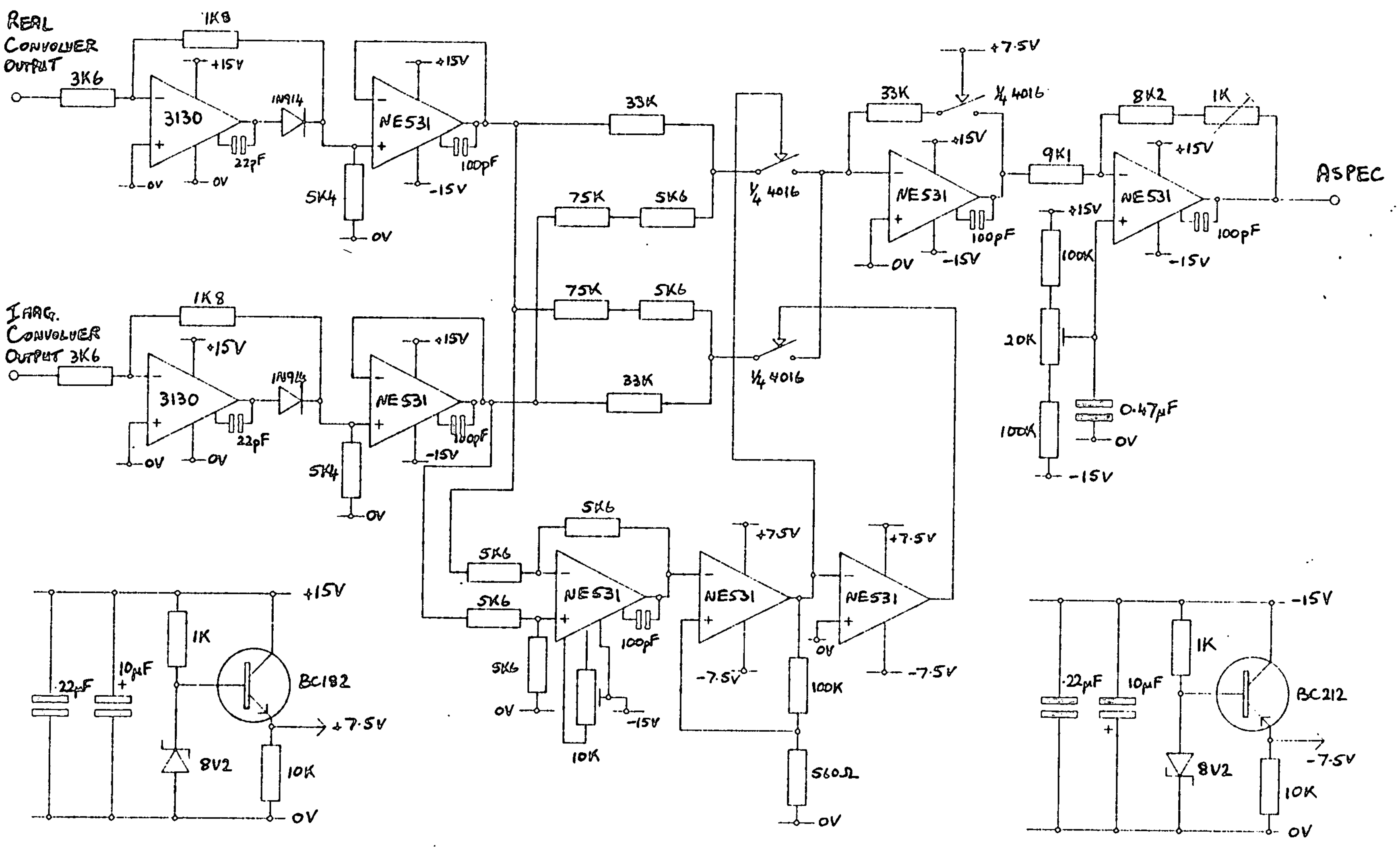


Fig.5.25 Linear Approximation to Modulus Function

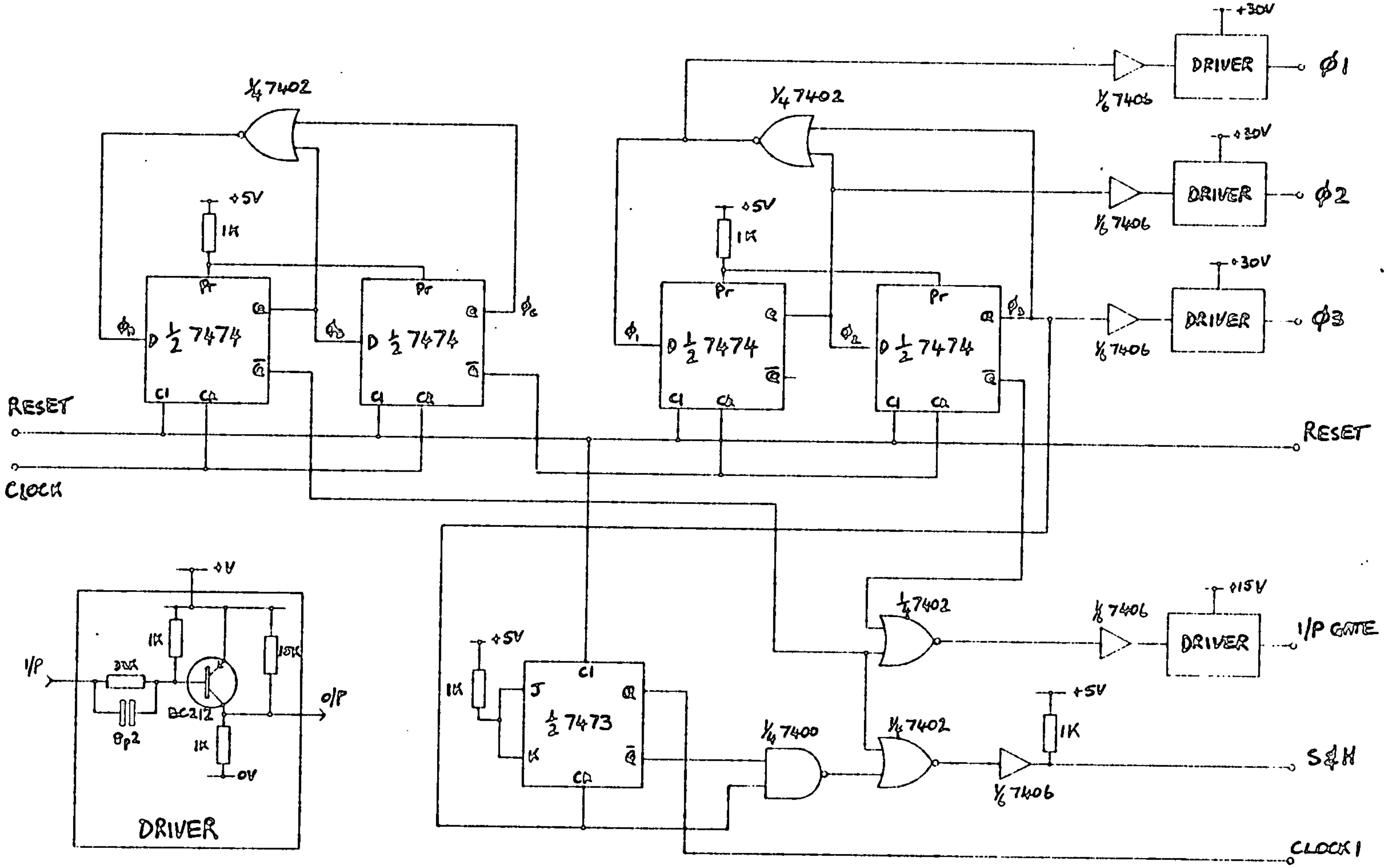


Fig.5.26 CZT Timing Logic

open collector buffers. To improve the CCD charge transfer efficiency, the drivers are designed to give the CCD clocks a fast turn-on and a slow turn-off. A pull-up resistor is included in each driver so that the clocks do not go more negative than 2V which ensures that there is always a thin depletion region at the surface of the CCD (the CCD substrate is at 0V).

The middle segment of ϕ_3 (ϕ_8) is selected to provide a gate pulse for the CCD diode cut-off input technique. The TTL signal is shaped by a circuit similar to the CCD clock driver to give a 15V pulse with a slow trailing edge. In this case, there is no pull-up resistor.

Because the CCDs are tapped every alternate stage, sample and hold pulses are required only after every second CCD transfer. A JK flip-flop connected as a toggle is used to divide the ϕ_3 clock by two and, after appropriate gating, the sample and hold pulse is available (see timing diagram in Fig.5.27). The sample and hold pulse is chosen in the middle of a ϕ_3 cycle to allow the signal time to settle.

Again, because of the CCD's alternate tapping, the clock to the pre-multiplier ROM address counter is at half tne CCD transfer rate. The ϕ_3/2 signal is appropriately timed for this purpose. The CCD is therefore operated in a Double Sample Alternate Tap (DSAT) mode as described in

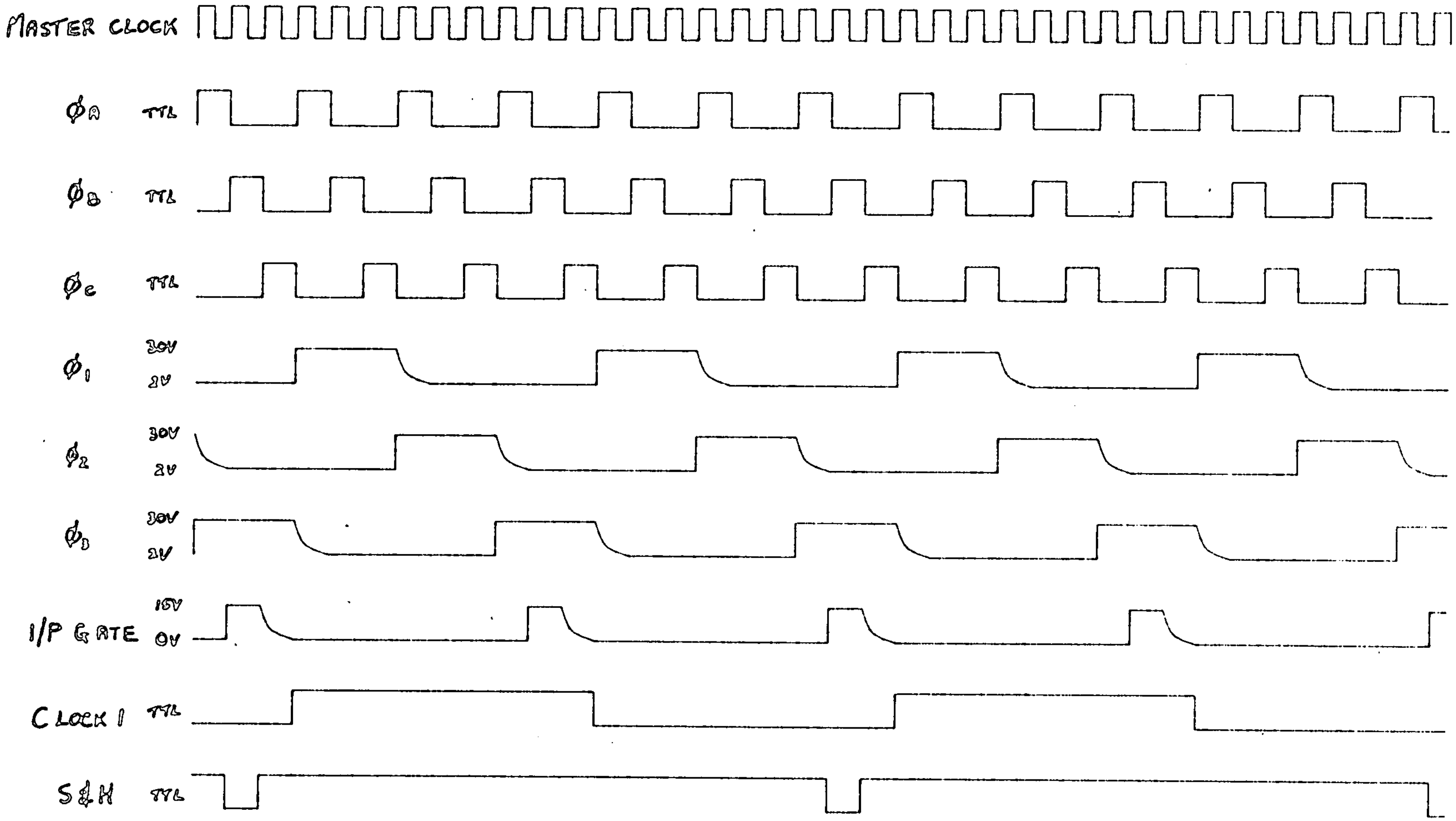


Fig.5.27 CZT Timing Diagram

Ref.[63].

A frame sync. output waveform is derived from the ROM address counter (Fig.5.19) and a reset input is provided to synchronise the CZT frame. Overall, the master clock input rate is 18 times the effective CZT sample rate.

5.3.5 90-Degree Phase Difference Network

In most CZT applications, only real input data are available; real data on their own make inefficient use of the convolvers since the resulting spectrum always contains an image. However, by generating quadrature data (i.e. real and imaginary parts) from the real input, it is possible to utilise the full processing bandwidth. Quadrature inputs may be generated by (a) filtering or (b) modulating the data onto an IF carrier and demodulating in quadrature [92]. Only the filtering method is considered here.

The generation of an output signal with exactly 90° phase shift relative to the input signal is an extremely difficult operation. It is much easier to produce two new outputs with 90° phase shift relative to each other, i.e. for an input signal of the form

$$x(t) = A\cos(wt + \emptyset) \qquad \ldots (5.9)$$

it is relatively straightforward to generate

$$y_1(t) = A\cos(wt + \emptyset + \theta) \qquad \ldots (5.10)$$

and

$$y_2(t) = A \sin(wt + \emptyset + \theta) \qquad \text{....} (5.11)$$

where θ is an arbitrary phase. This method of generating complex data is valid in cases where only the amplitude spectrum is required and where the absolute phase information is unimportant. The 90° phase difference technique is therefore suitable for speech processing.

The synthesis and analysis of a 90° phase difference network designed for speech processing is detailed in Appendix C. This particular network was specified to operate over the band of frequencies from 50Hz to 3200Hz with an absolute phase difference error of 1°. The complete circuit, including input drivers and output buffers, was constructed on a printed circuit board measuring 114mmx75mm.

Measurement of the practical phase difference function was accomplished by an analogue phase meter and the results are plotted in Fig.5.28. The peak phase difference error is 1.7° which exceeds the design tolerance of 1°. This is due to a linear phase difference error produced by phase mismatch in the drive circuitry.

5.3.6 Low-pass Filter

For correct operation with analogue input signals, the sampled-data CZT processor demands an input anti-aliasing

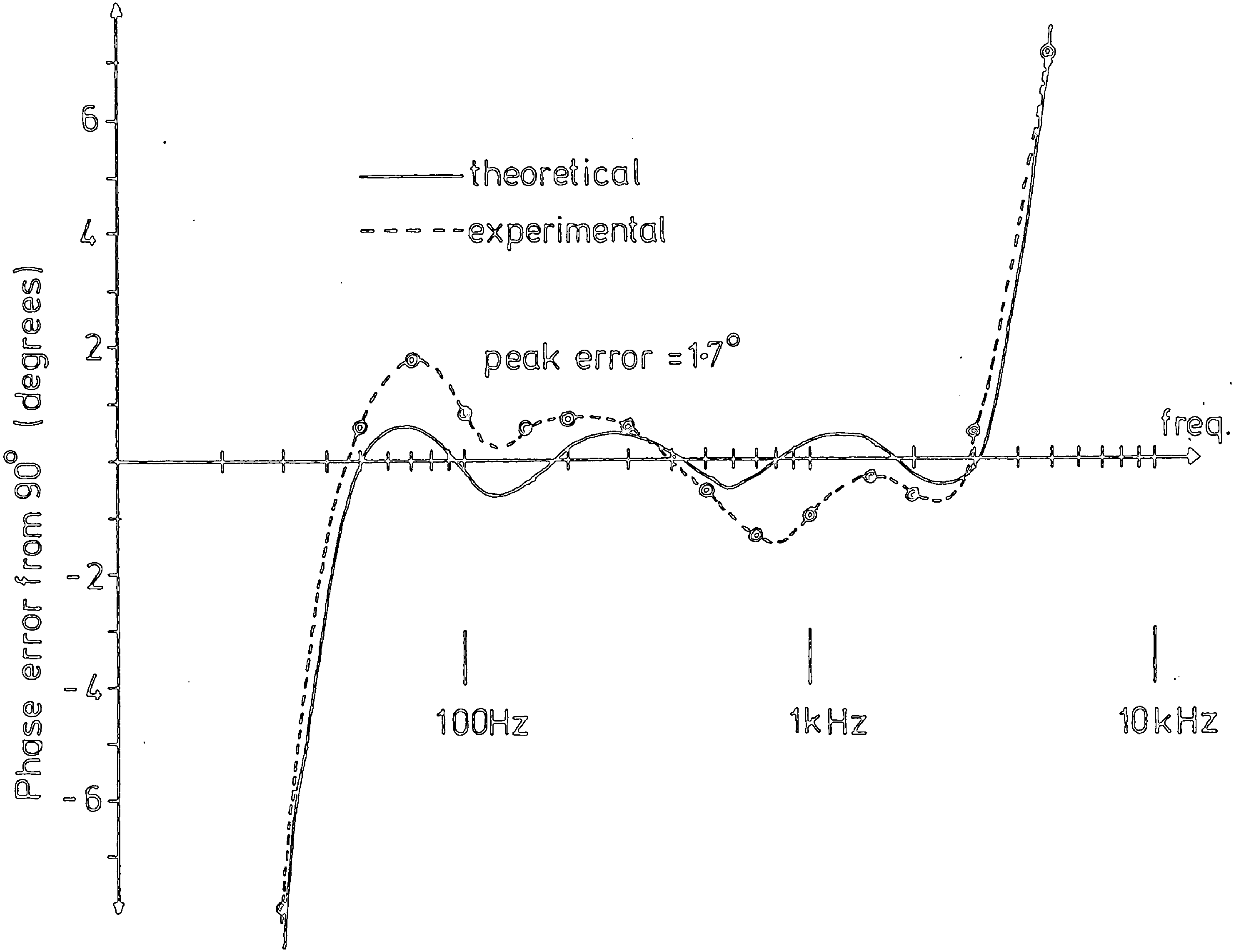


Fig.5.28 Theoretical and Experimental Phase Difference Ripples

low-pass filter. To make efficient use of the CZT's processing bandwidth, this filter must cut-off very close to the Nyquist limit and roll-off very steeply (consistent with the step response) so that aliased components are attenuated sufficiently.

Such a filter has been designed and constructed for use in speech processing with the following prerequisite characteristics:

1. cut-off frequency: 3000Hz

2. attenuation: >20dB at 3200Hz

3. in-band ripple: <1dB

4. phase characteristic: unimportant

A ½dB ripple 10-pole Tschebycheff transfer function was selected to give the best compromise between the roll-off and the step response and was implemented by cascading five buffered two-pole Rauch sections [93]. Fig.5.29 shows the circuit diagram for a single Rauch section and the component values for each of the five sections are given in Table 5.2.

The measured frequency response is illustrated in Fig.5.30a and compared with the theoretical response in Fig.5.31. It can be seen that the higher frequency poles are not exactly matched giving a 2dB ripple at the

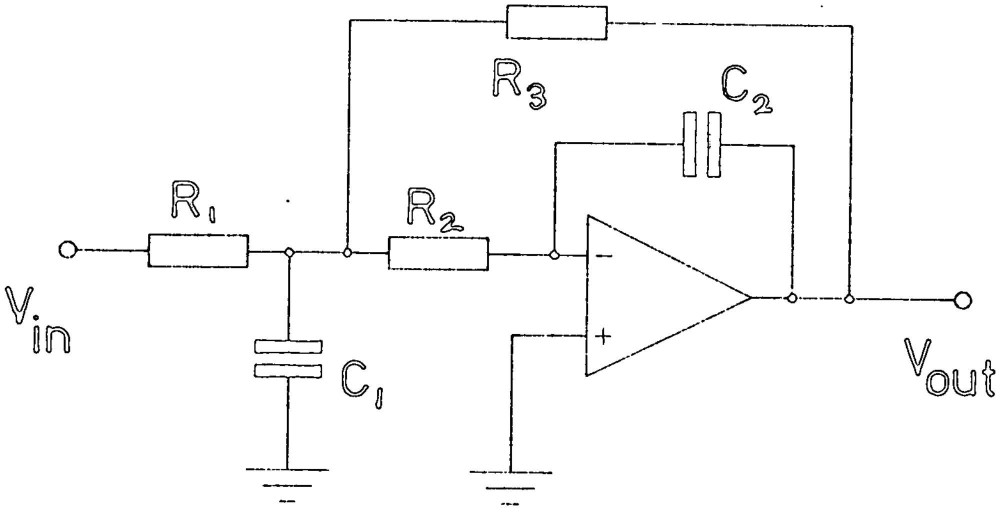
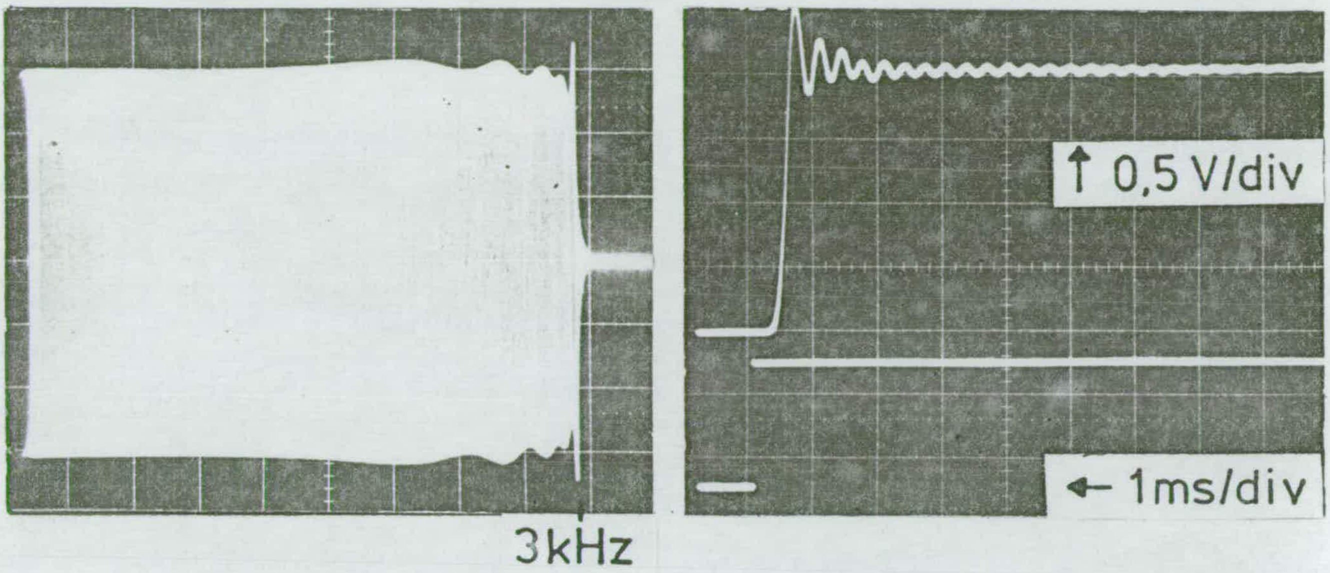


Fig. 5.29 Two Pole "Rauch" Low Pass Filter

Section	C_1	C_2	R_1, R_2, R_3 *
1	0.47μF	160pF	6k2
2	0.1μF	345pF	10k
3	0.047μF	620pF	13k
4	0.01μF	470pF	51k
5	4700pF	1200pF	96k

* All resistors in ohms

Table 5.2 Component Values for 10-pole Tschebyscheff Low-pass Filter



(a) frequency response (b) step response

Fig.5.30 Practical Tschebycheff Filter Measurements

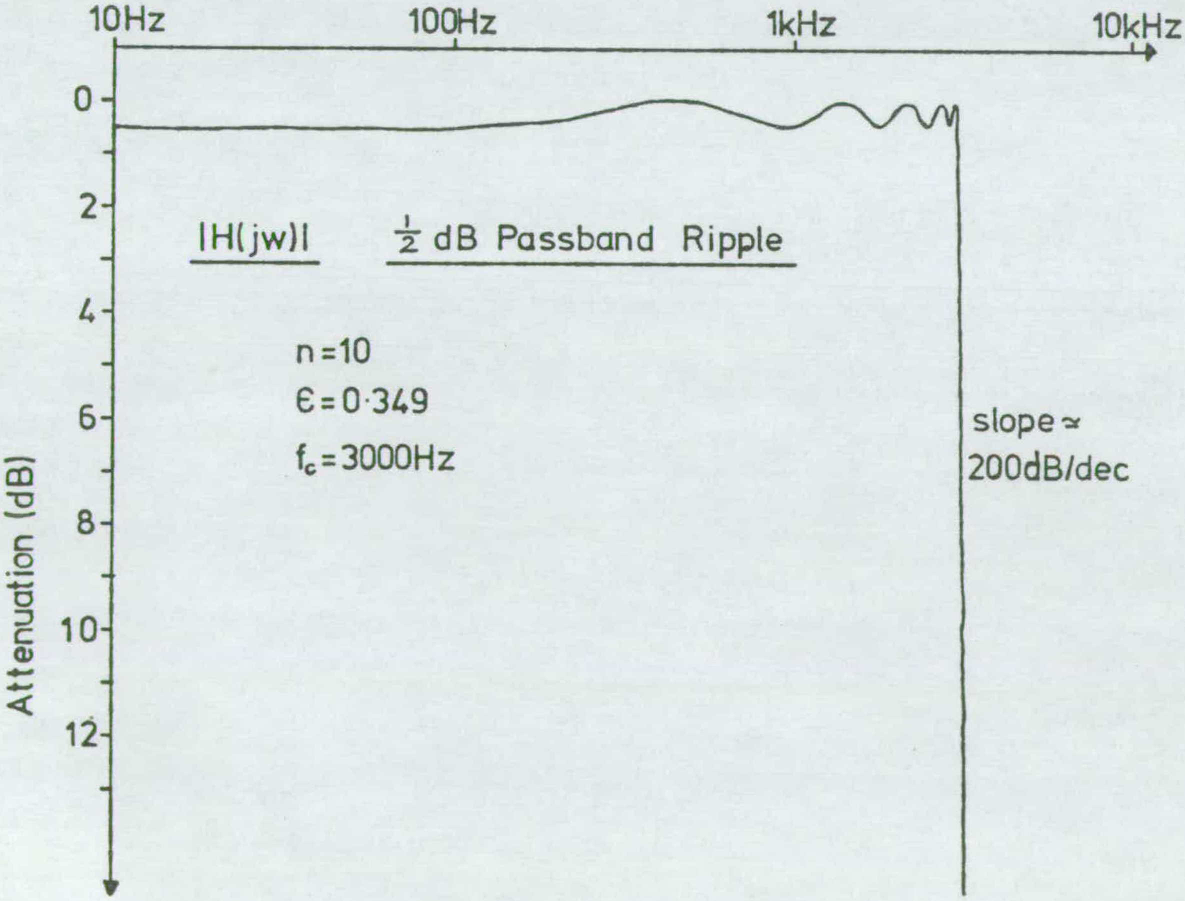


Fig.5.31 Tschebycheff LPF Theoretical Frequency Response

band-edge. This problem is inherent in filters having a large number of poles. The filter step response is reproduced in Fig.5.30b and has settled to 5% within 2.5mS.

5.3.7 Physical Construction

The CZT circuitry in sections 5.3.1 to 5.3.4 is constructed on two printed circuit boards measuring 250x115mm (see photograph in Fig.5.32). One of these boards contains the complex convolver with the filter weighting networks mounted on pluggable subassemblies. The other board houses the timing logic and CCD drivers, the pre-multipliers and both the post-modulus circuits. The overall processor fits into a volume of 30x250x115mm.

5.4 HARDWARE PERFORMANCE

When the processor is configured for a 64-point unweighted sliding CZT, the output in response to d.c. inputs in both the real and imaginary channels is shown in Fig.5.33. The master clock rate is 57.6kHz giving an effective processor clock of 3.2kHz and a resolution of 50Hz. The expanded photograph shows that the PE/S ratio is -42dB. The magnitude output in Fig.5.34 is the response when the real input is a 2V p/p 950Hz tone (integral number

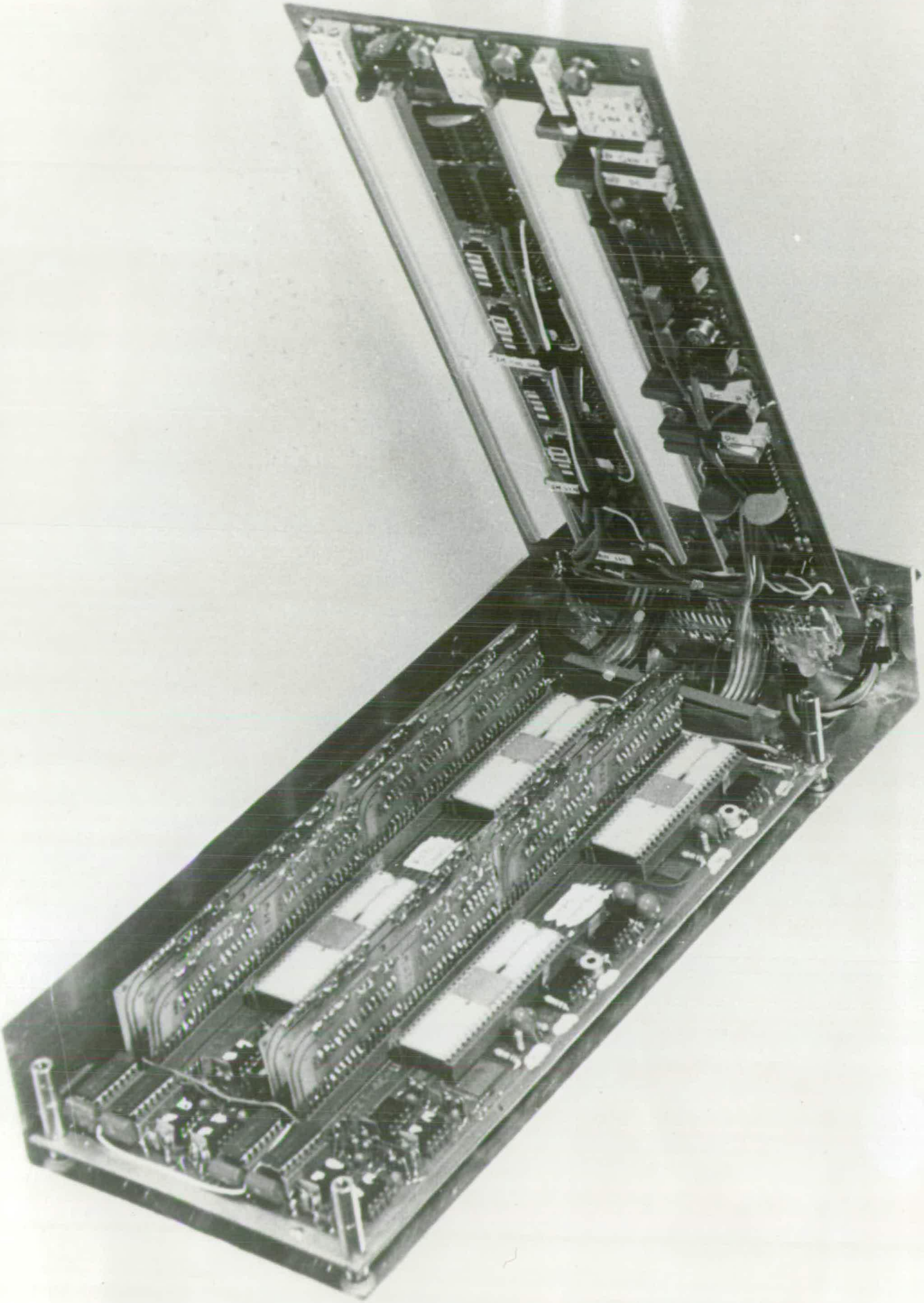


Fig.5.32 Photograph of CZT Hardware

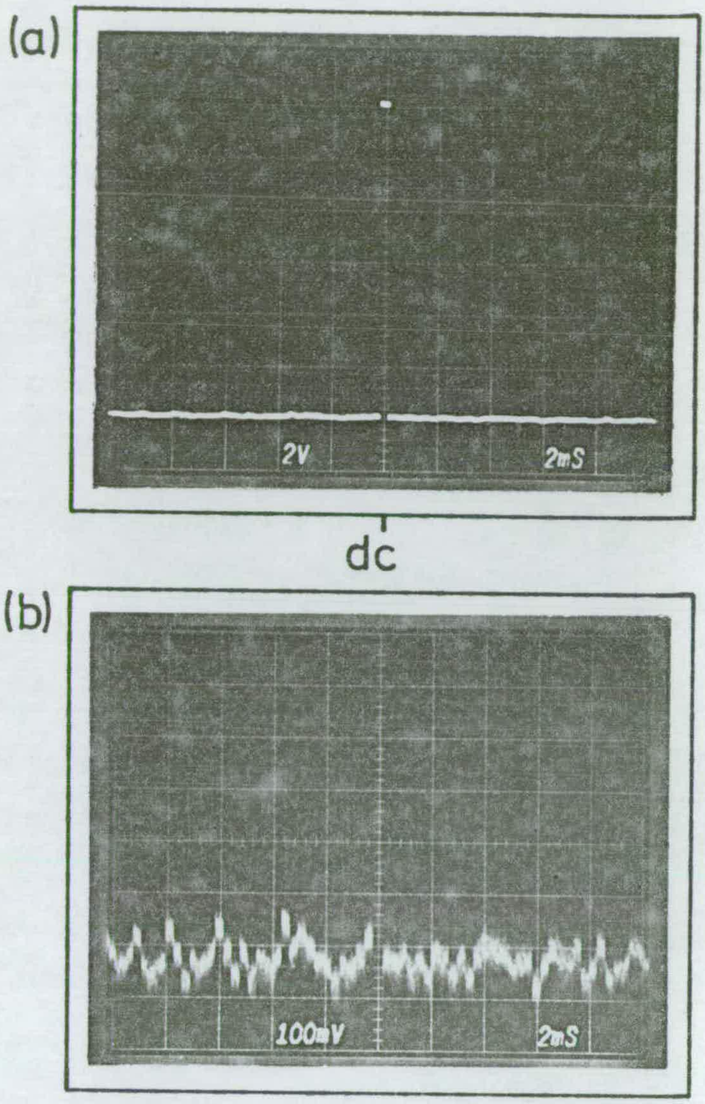


Fig.5.33 SCZT Output for a DC Input

(a) response to dc

(b) expanded (x20)

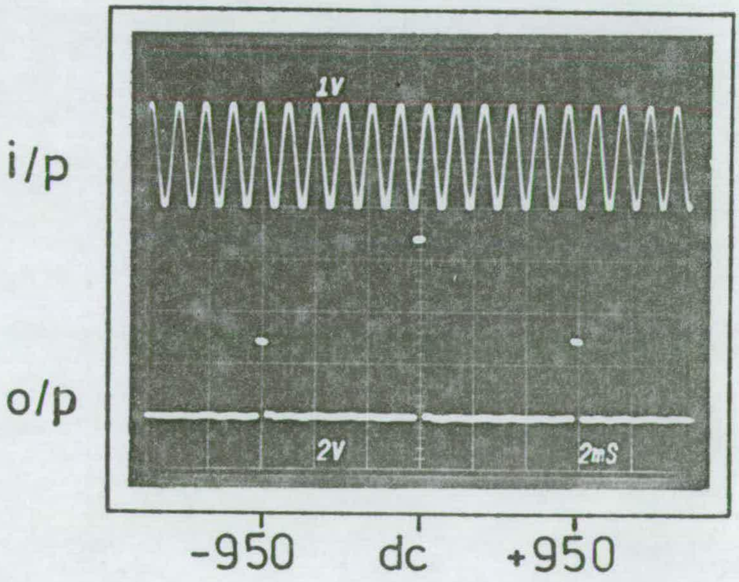
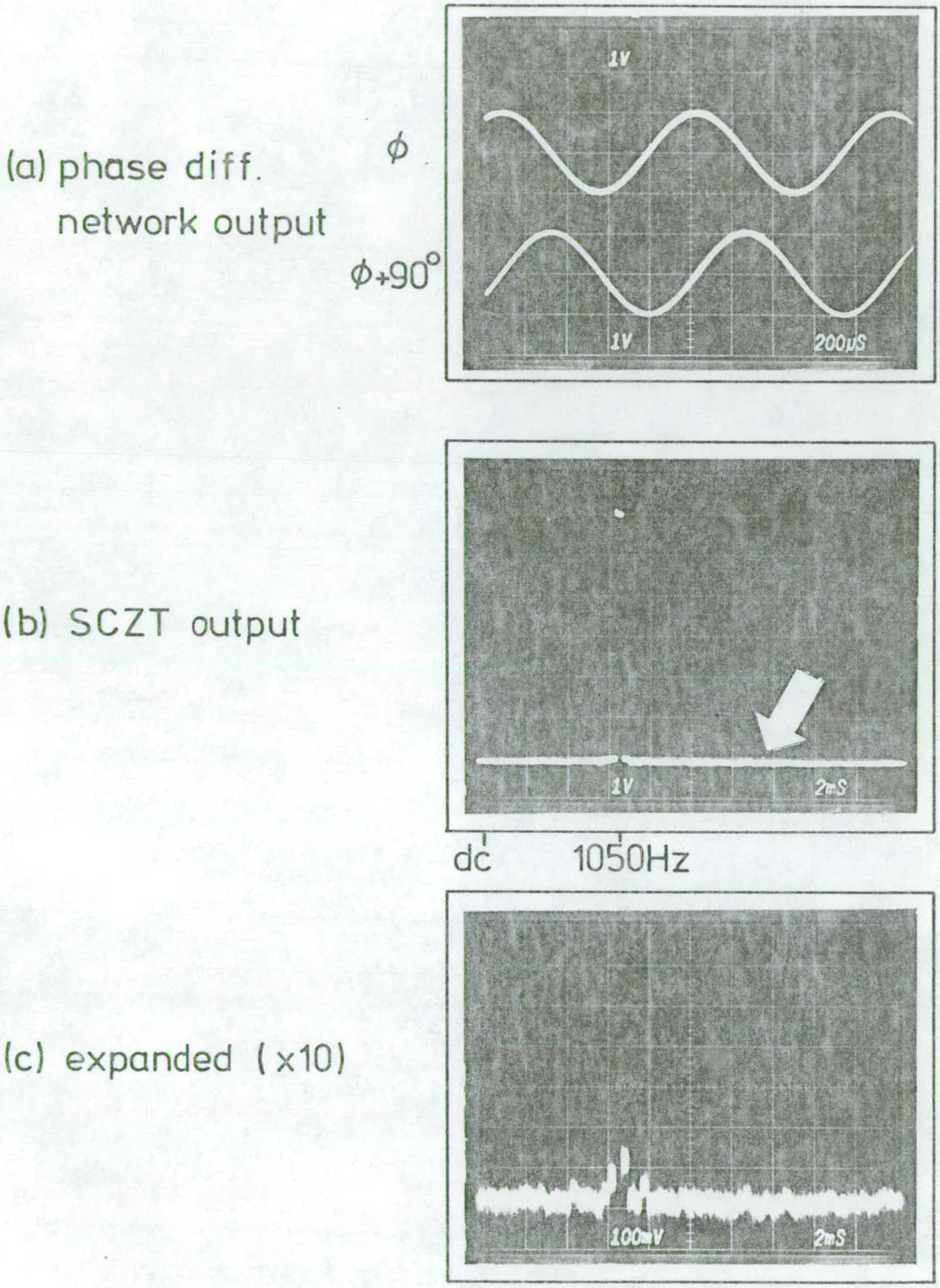


Fig.5.34 Amplitude Spectrum of 950Hz Real Tone Input

of cycles) on a 1V d.c. pedestal and the imaginary input is zero. Because the input is real only, Fig.5.34 may be interpreted in terms of positive and negative frequencies with d.c. in the middle.

The addition of a 90-degree phase difference network cancels the image frequencies and effectively doubles the processing bandwidth. The phase difference network outputs (Fig.5.35a) are input to the real and imaginary CZT inputs to give the output shown in Figs.5.35b and 5.35c. In these oscillograms, the frame has been rotated by N/2 points to make the d.c. response appear at the left-hand side. The peak error is due to transfer inefficiency and the PE/S ratio is approximately -40dB. It can be seen that the image frequency, which should appear at the arrow in Fig5.35c, has been well suppressed by the phase difference network. As explained in section 4.2.1, the peak and nulls of the output (sin x)/x response move off the sampling grid for a non-basis vector input. This is demonstrated in Fig.5.36 where the input frequency is 1175Hz and the output falls exactly between two adjacent resolution cells.

Dynamic range and linearity measurements are plotted in Fig.5.37. The linearity of the output is limited to 30dB by the output transconductance multipliers and noise restricts the overall dynamic range to 48dB. It is thought that the



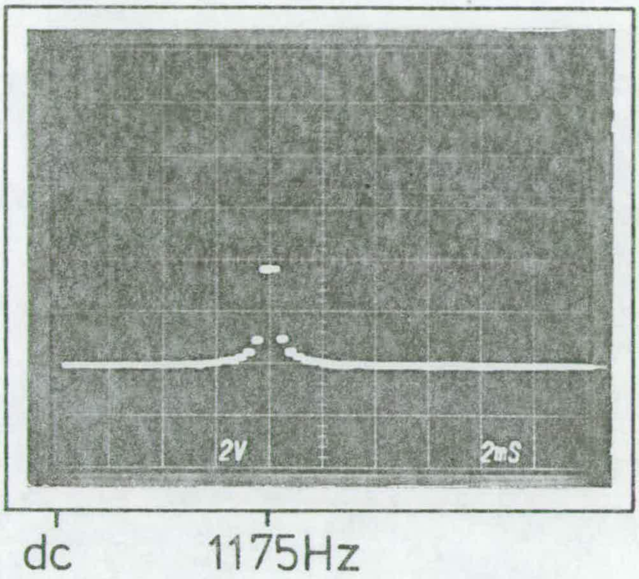
(a) phase diff. network output

(b) SCZT output

(c) expanded (x10)

Fig.5.35 SCZT Operation with Phase Difference Network

Fig.5.36 Output for Non-Basis Vector Input



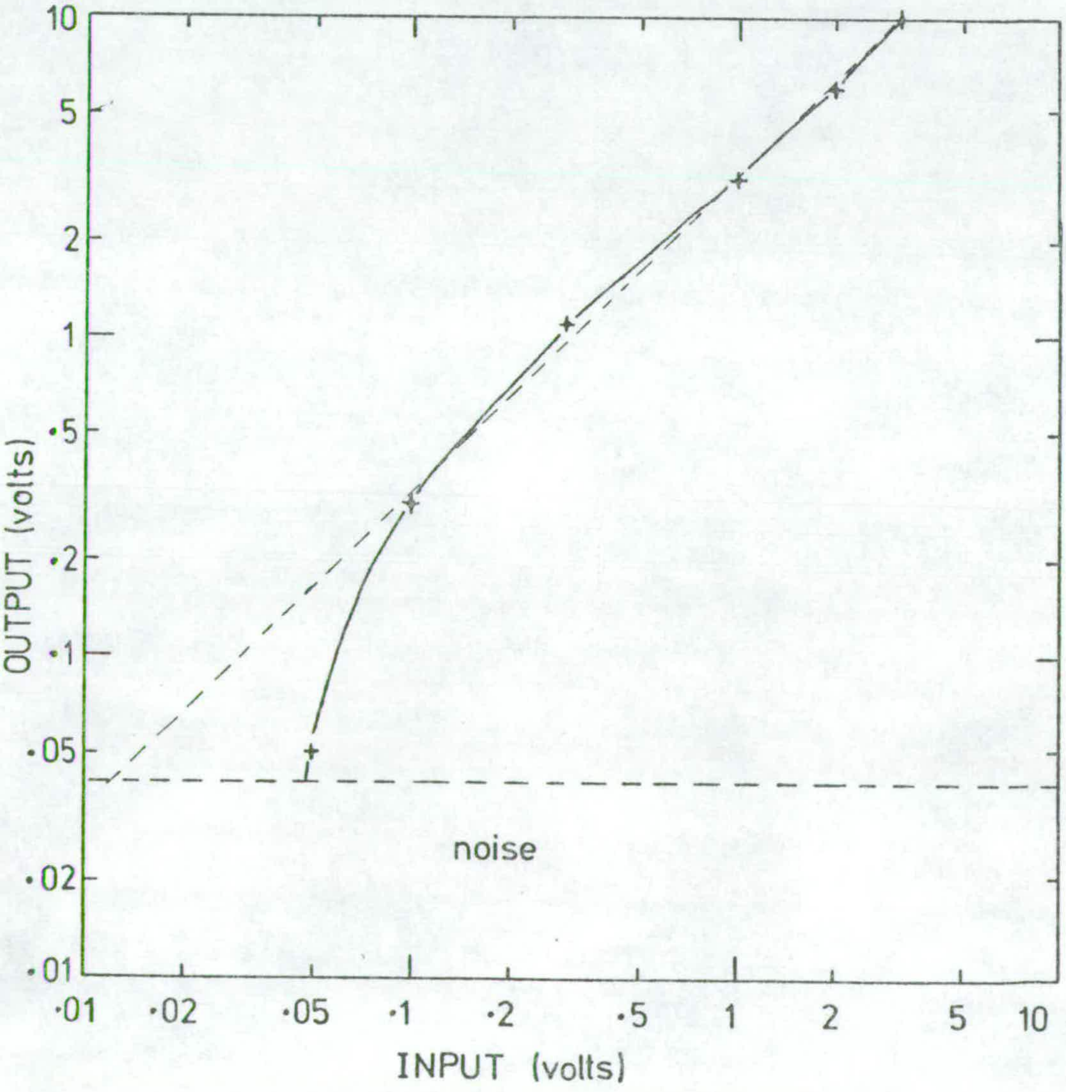


Fig.5.37 Linearity and Dynamic Range Measurements

noise level could be improved by separating the digital and analogue sections of the circuitry on the printed circuit board.

To test the processor's performance with regard to charge transfer efficiency, the input signal was swept from 10Hz to 3kHz in 5 seconds and a time exposure of the output was developed (Fig.5.38). It can be seen that the input low-pass filter characteristic (Fig.5.30a) is superimposed upon a general attenuation trend in the spectral output. The high frequency components are attenuated by 3.5dB more than the low frequency components.

The master clock rate can be varied from below 10kHz to almost 2MHz providing effective clock rates of between 550Hz and 110kHz. At the lower clock rate, the resolution is 8.6Hz and the effect of dark current significantly distorts the output. The upper clock frequency gives a resolution of 1718Hz and is limited by the slew rate of the transversal filter summing amplifiers (NE531) and also by the sample and hold amplifiers (HA2425). A major hardware defect is the variation of transversal filter d.c. output with clock frequency; changes in d.c. offset cause the modulus circuitry to malfunction. This point is discussed more fully later.

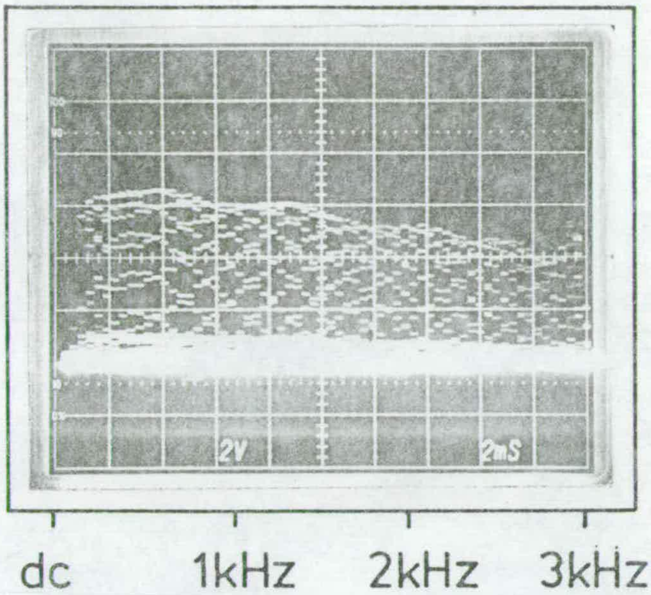


Fig.5.38 Response to Chirp Input (time exposure)

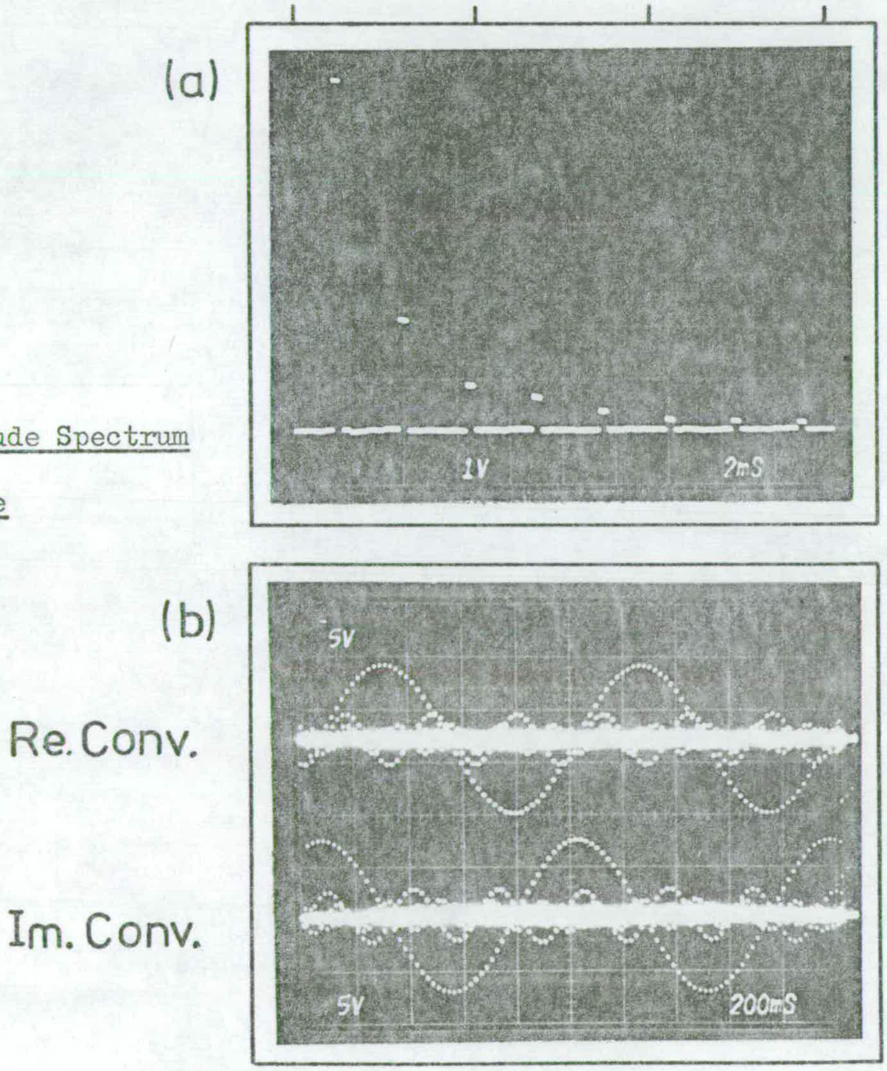


Fig.5.39 Amplitude Spectrum of Square Wave

An example of the processor analysing a square wave is demonstrated in Fig.5.39a. Here the fundamental is 200Hz and the odd harmonics are spaced at 400Hz (the processor's resolution is 50Hz). The photograph in Fig.5.39b displays many sequential frames of real and imaginary convolver outputs, neatly illustrating the function of the modulus circuit. Because the input signal is free running, the phase of the input signal ,θ, relative to the CZT frame is changing. The fundamental convolver outputs are modulated by COS θ and SIN θ, the third harmnics by COS 3θ and SIN 3θ, etc. The modulus function performs the operation

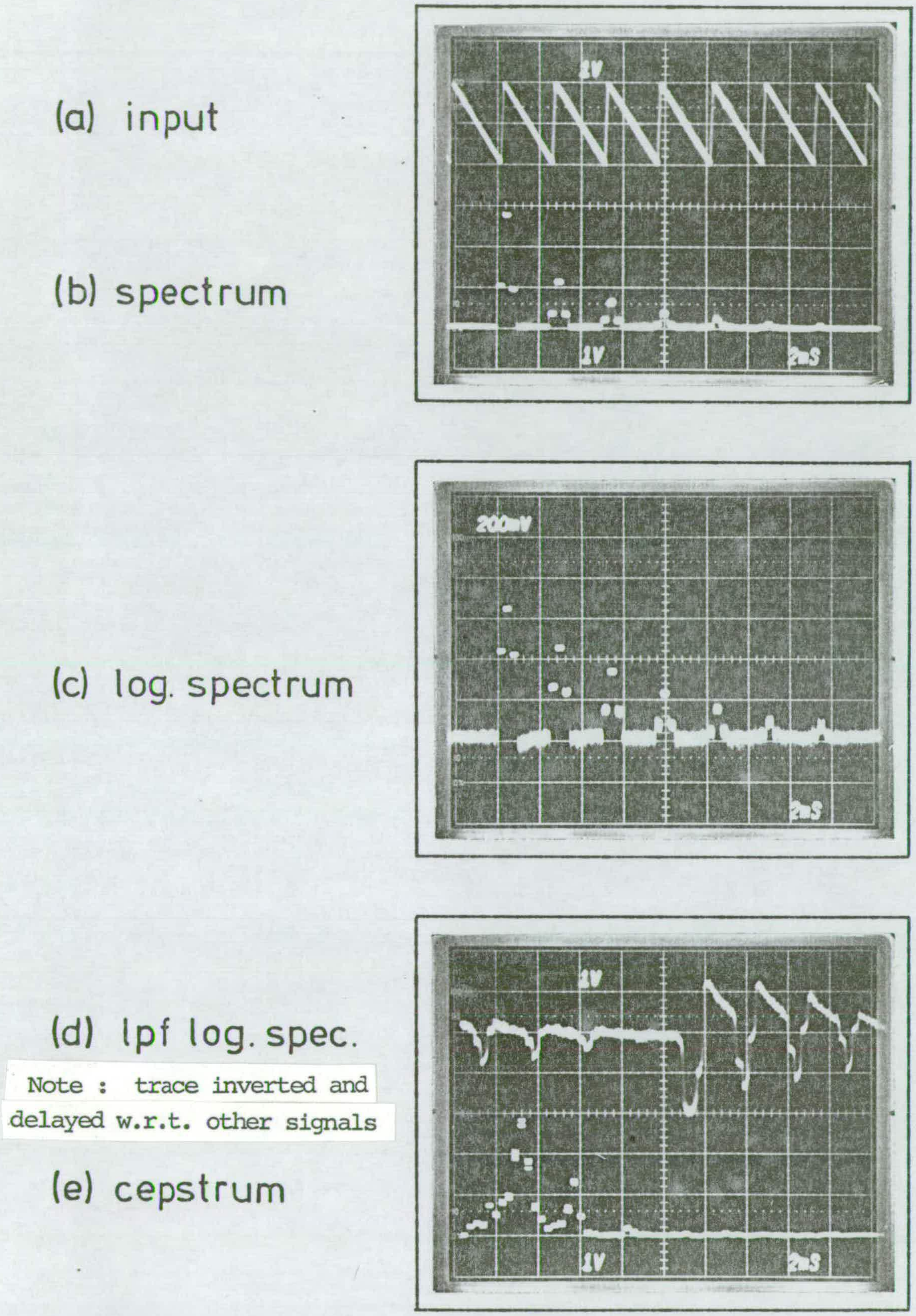
$$\cos^2(\theta) + \sin^2(\theta) = 1 \qquad \ldots (5.12)$$

on each of tne components so that the amplitude spectrum is independent of input phase. When there is a d.c. offset added to either the real or imaginary channel outputs, the COS θ ,SIN θ etc. terms do not completely cancel and a frame to frame ripple is present in the output. This is a serious effect because the d.c. offsets change with both clock frequency and CCD temperature. A practical solution is to employ chopper stabilisation in a feedback loop to the CCD input. However, this implies major hardware modification.

The processor's power dissipation was measured and found to be almost 12W. This figure could be reduced significantly be the use of CMOS circuitry and MOS amplifiers.

A final demonstration of the CZT's operation is in the calculation of the cepstrum (section 2.2) for a triangle waveform. Here the processor is a weighted 64-point sliding CZT and the modulus circuitry is the linear approximation. The sawtooth waveform in Fig.5.40a is processed to provide the amplitude spectrum which is subsequently logged and stored on a tape recorder. (Note the decrease in resolution caused by the weighting function). The log spectrum is replayed through the CZT to provide the cepstrum in Fig.5.40e. The first peak is the fundamental quefrency at 2.5mS and the smaller peaks are the rahmonics.

In this section, only the 64-point SCZT configuration has been demonstrated. It is considered that the operation of the direct CZT has been covered sufficiently in previous sections.



(a) input

(b) spectrum

(c) log. spectrum

(d) lpf log. spec.

Note : trace inverted and delayed w.r.t. other signals

(e) cepstrum

Fig.5.40 Demonstration of Cepstral Processing

CHAPTER 6

THE ON-LINE COMPUTER SIMULATION OF A CCD CHANNEL VOCODER

Armed with both the vocoder design philosophies and the signal processing capabilities of a new technology, it is possible to postulate new vocoder implementations which may provide engineering benefits. In the design of any system as complex as a vocoder, it is generally a wise precaution to simulate fully the effect of system variables before the commitment of hardware. This is especially true in low bit rate speech processing because there are many sources of distortion and ignoring any of these can be dangerous.

Section 6.1 describes the basic computing facilities which were used by the author in this simulation. The computer models for a novel CCD implementation of the now established channel vocoder, together with the simulation detail and conclusions, are summarised in sections 6.2 and 6.3 for the analyser and synthesiser respectively.

6.1 COMPUTING FACILITIES

In the computer simulation of speech processing systems, several special requirements must be considered. Any digital speech facility must have access to both A to D and D to A conversion with associated buffer store or Direct

Memory Access (DMA) and be capable of handling a large amount of data. For example, a one second segment of speech signal (the minimum which is useful) sampled at 10kHz and quantised to a 12 bit accuracy requires 15,000 bytes of storage. In addition, it is necessary to have suitable output facilities for the convenient display of this large amount of data, either in the time domain or in the frequency domain. Although processing in real-time is not vital, hardware additions such as a floating-point processor or an FFT processor combine to make the overall system more efficient and less time-consuming. In the course of the simulations described in this chapter, two alternative computing facilities were used and each will be described briefly.

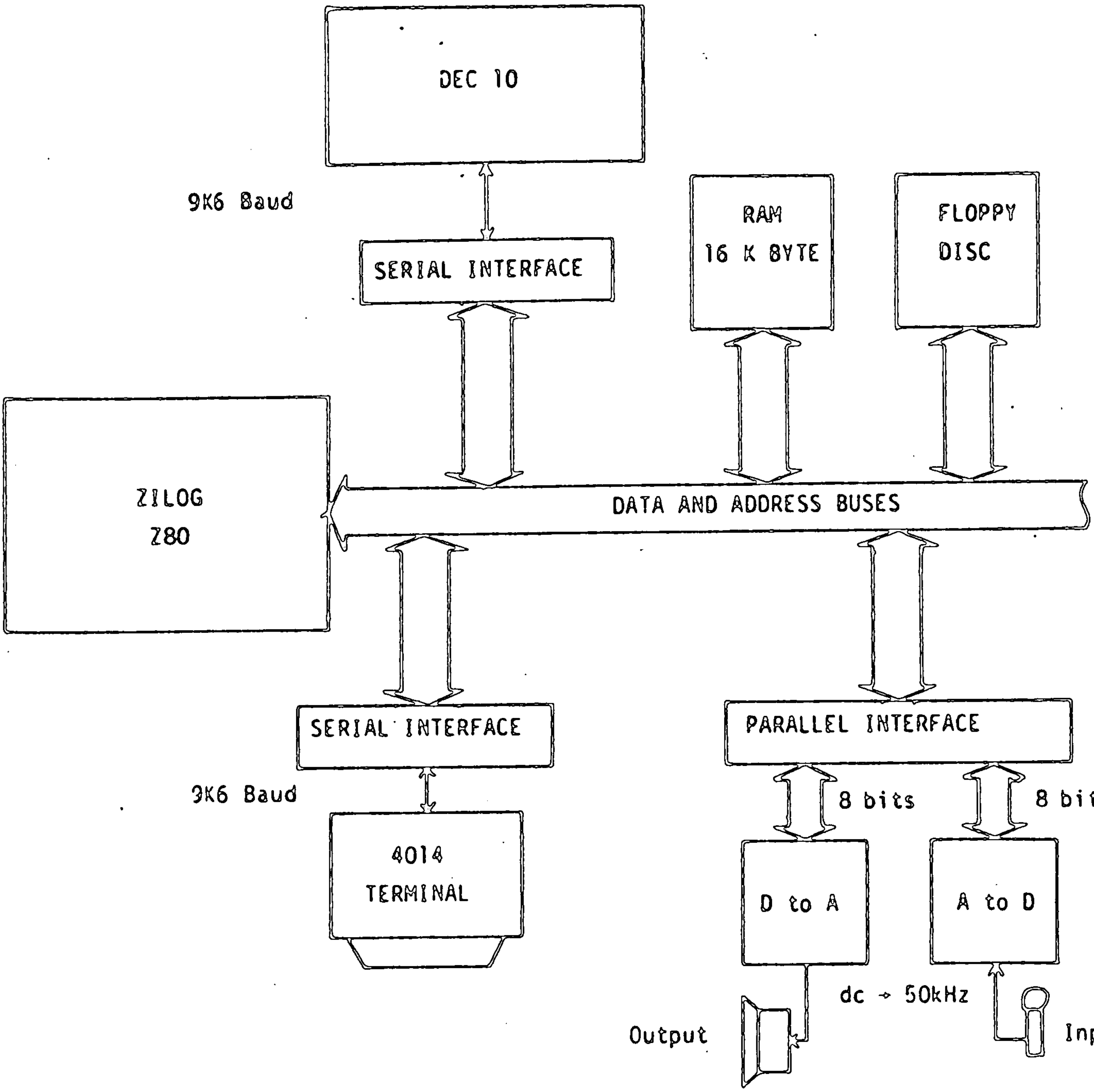
The first was based upon a PDP11/70 mainframe with 128k words (16 bits) in main memory. Real arithmetic was handled by a hardwired floating point processor. Four disc units, three cartridge and one fixed, provided the fast store where all user programmes and an RSX-11M operating system resided. A floppy disc drive catered for individual user programme backup and a magnetic tape unit for longer term and system management backup. The multi-access system communicated with up to four users at any one time through three VDUs and one teletypewriter. Hardcopy output could be obtained from either a lineprinter or a graph plotter controlled by a dual 12-bit D to A converter (x and y channels). Analogue data were input to the computer via a 12-bit A to D converter under the control of an external clock (variable from dc to 100kHz) using DMA. The software included a Fortran IV compiler, a machine language assembler and the usual DEC utility programmes, as well as an extensive user written library.

The second computing facility, more readily available to the author, consisted of a large time-shared DEC-10 system. DMA was not permitted and the only access was via a standard serial terminal input port. This restriction necessitated the development of a specialised microprocessor based buffer to control the input and output of speech data. The design of this "intelligent terminal" is detailed in Reference[99]and a block diagram illustrating the main component parts is given in Fig.6.1. Software in the Z80 microprocessor controls different modes of operation and these may be selected by either the DEC-10 computer or the 4014 graphics terminal. These modes of operation are designed to allow:

1. direct communication between the 4014 terminal and the DEC-10 computer

2. speech input through an 8-bit A to D converter into a 12k byte buffer and subsequent serial transmission to file storage on the DEC-10

3. speech output, by first gathering data from the DEC-10 into a buffer, and then recirculating these



Modes of Operation
(controlled by Z80)

(a) Feedthrough (Full Duplex) [4014 DEC 10]

(b) Analog I/P [A/D → RAM/FLOPPY]

(c) Transmit (Half Duplex) [RAM → DEC 10]

(d) Analog O/P [RAM → D/A]

(e) Receive (Half Duplex) [DEC 10 → RAM/FLOPPY]

Fig 6.1 Microprocessor Based Speech Terminal

data through an 8-bit D to A converter.

Although this system is not as flexible as the first facility described, the processing power as applied in speech vocoder simulation is similar.

6.2 THE CHANNEL ANALYSER

The channel vocoder architecture (Fig.2.6) is ideally suited for CCD implementation. The parallel filter bank, which is the main processing block, may be replaced directly by an equivalent Fourier transform processor. Also, because of the increased processing power and flexibility made available by the use of such a processor, it is possible to replace the conventional time domain pitch detector by a technique which promises superior performance, the cepstral pitch detector (section 2.2).

A computer model for the analyser simulation is shown in Fig.6.2. The input signal is pre-emphasised and Fourier transformed in frames to provide sequential short-time representations of the speech amplitude spectrum. At this stage, the processing divides into two paths. In the upper path, each short-time spectrum is smoothed and compressed to achieve the required data reduction, whereas in the lower path the speech cepstrum is calculated. A decision

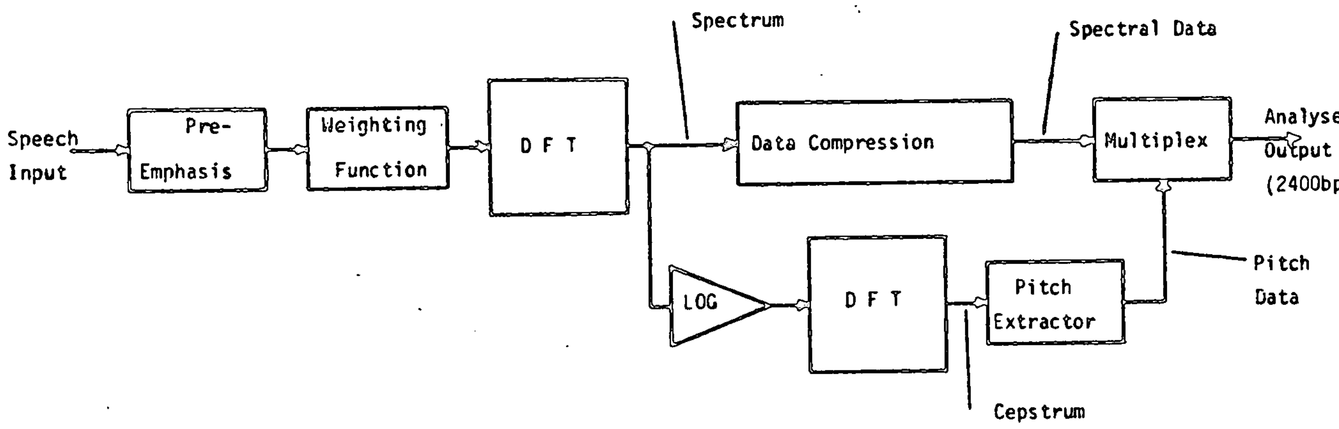


Fig.6.2 Channel Analyser Simulation

algorithm then extracts the appropriate excitation source information and transfers these data to the output for transmission. The main difference between the analogue filter bank implementation and Fig.6.2 is that the processing is performed in serial rather than in parallel.

6.2.1 Speech Input

Speech was input to the computer via a microphone, an anti-aliasing filter, an optional pre-equalisation filter and a 12-bit A to D converter, under the control of a supervisory software programme. This programme accessed an area set aside in main memory consisting of 12k integer words (1 integer word = 2 bytes). The extra four bits in each word were used as control flags for the A to D converter. To conserve accuracy in later processing, the integer values were "floated" to real values before being stored on disc. The speech sampling rate was set at 8kHz from an external clock and it was arranged to input 10240 samples. This allowed 1.28 seconds of speech to be recorded, sufficient for a short sentence.

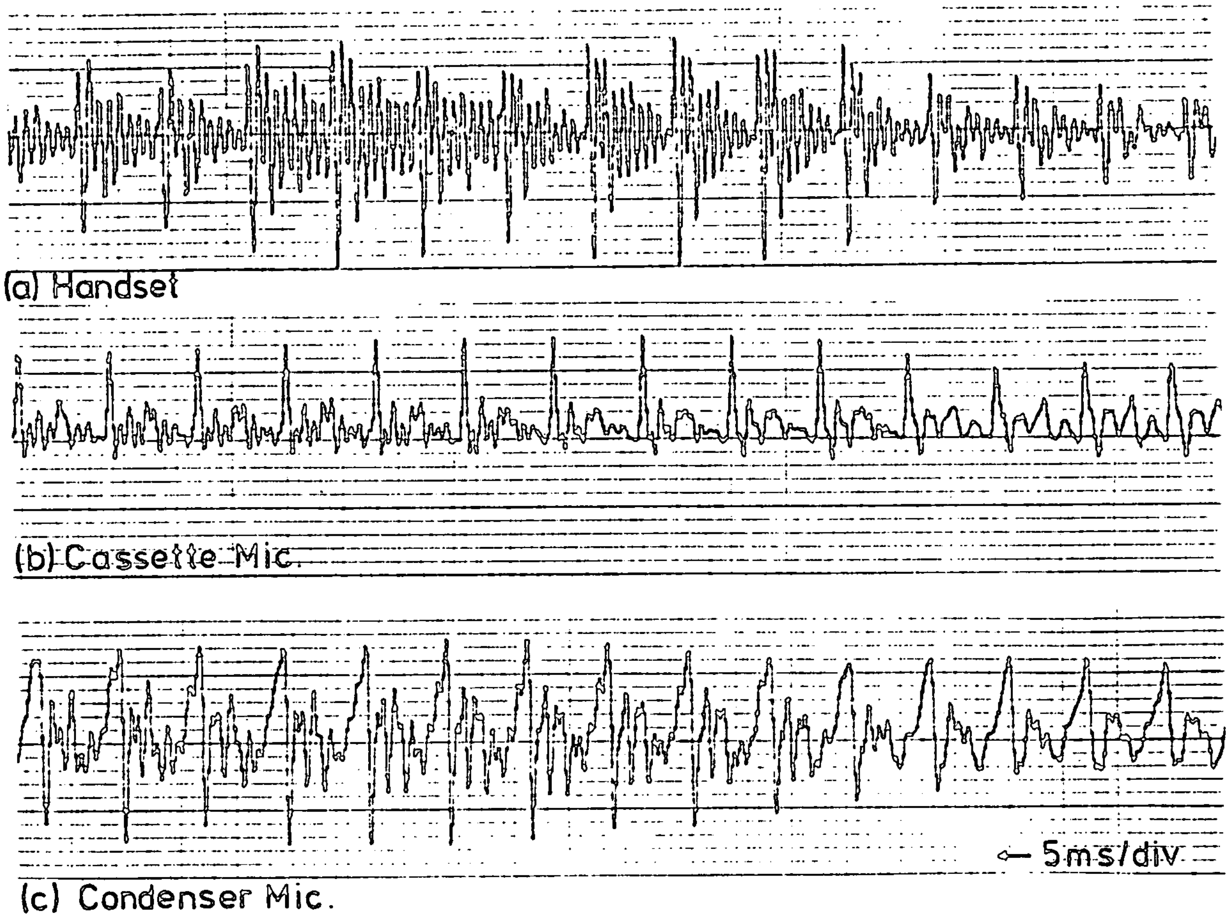
Three different microphones were used to take samples of varying quality. These were:

1. military handset with 300-3000Hz band-pass filter (telephone quality)

2. cheap cassette microphone with 4000Hz low-pass filter

3. standard condenser microphone with 4000Hz low-pass filter.

Segments of time domain speech produced by these three combinations (without pre-equalisation) are shown in Fig.6.3. (Each segment is part of the same phrase spoken by the same speaker). The signal in (a) clearly demonstrates the high frequency emphasis placed on the speech by the telephone handset. The lack of bass frequencies makes time domain pitch period detection extremely difficult. In



(a) Handset

(b) Cassette Mic.

(c) Condenser Mic.

Fig.6.3 Comparing Microphone Characteristics

waveforms (b) and (c), the band-pass filter has been replaced by a low-pass filter which allows the pitch fundamental to pass unattenuated (in this example, f_{pitch} = 125Hz) and the increased pitch period definition in these waveforms is obvious. The condenser microphone used in this experiment had a flat frequency response (1dB) from 20Hz to approximately 40kHz.

In addition, three different male speakers and one female provided samples with low, middle and high pitch. The input sentences recorded were:

1. "I know when my lawyer is due"

2. "We were away a year ago"

3. "Every salt breeze comes from the sea"

4. "I was stunned by the beauty of the view".

Sentences (1) and (2) are all voiced (except for the stop gaps) whereas (3) and (4) contain both voiced and unvoiced speech. The above sentences were used by Rabiner et. al. [22] to investigate several pitch detection algorithms.

Conventionally, pre-emphasis of 6dB/octave is applied to the speech to compensate for the general trend [25]. This can be easily implemented in one of two ways: (a) by time domain differencing of the speech data according to the relationship

$$y_k = x_k - p\, x_{k-1} \qquad \ldots (6.1)$$

where $\{y_k\}$ is the pre-emphasised speech, $\{x_k\}$ is the input speech and p=+1 for 6dB/octave lift or (b) by filtering the speech before it is digitised. To minimise the computer processing time, method (b) was selected and the filter section shown in Fig.6.4 was added before the A to D converter. The component values used give a 6dB/octave boost from 1kHz to 10kHz.

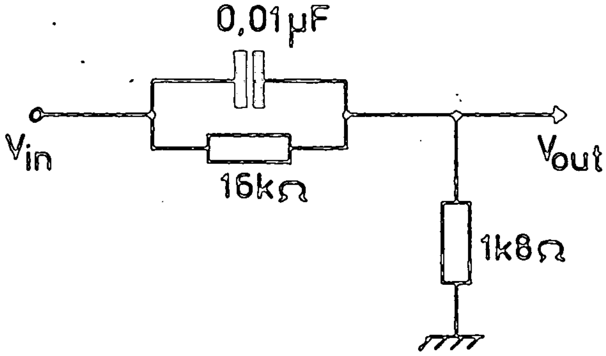


Fig.6.4 Pre-emphasis Filter

Finally, each segment stored on the computer was normalised to its peak value to ease scaling problems in later stages of the processing.

6.2.2 Spectrum and Cepstrum Computation

The main questions to be resolved are:

1. what resolution is required in the spectrum?

2. what resolution is required in the cepstrum?

3. are weighting functions necessary, and if so, what type should be used?

4. can a sliding transform be utilised to give a potential saving in hardware complexity?

In the channel analyser configuration shown in Fig.6.2, both the speech spectrum and cepstrum are used. For an efficient hardware structure, the DFT processors employed in each calculation should, if at all possible, have identical characteristics. However, it is important to examine the needs of each calculation independently and then, if a suitable compromise can be reached, merge the two.

It has been found in the channel vocoder (section 2.3) that filter bandwidths of between 100 and 300Hz provide sufficient resolution for representation of the short-time spectral envelope of speech [2]. Some vocoders have linearly spaced constant bandwidth filters whereas others have logarithmically spaced filters (Table 2.1). In order to provide a good approximation to the vocal tract transfer function, each rectified filter output is averaged (low-pass filtered) over a period of between 20 and 30ms. Normally, if the averaging period is greater than 30ms, the spectral output will not reflect fast changes in spectral content and, if the period is less than 20ms, too few pitch periods will be included in the average (see chapter 2).

There are two alternative strategies for the implementation of a suitable approximation to the channel vocoder filter bank using a sampled-data DFT processor. If the input speech is sampled at 8kHz, the useful signal bandwidth is less than 4kHz, and the application of a 40-point DFT processor transforms this real signal into a

4kHz amplitude spectrum with linear resolution of 200Hz. However, this spectrum is obtained from a 5ms segment of the speech waveform and it is therefore necessary to average four successive spectral frames to achieve a result which represents a 20ms segment. The alternative solution is to employ a 160-point processor integrating over 20ms of speech and then reduce the resolution by grouping and averaging spectral coefficients. This solution is preferred because the higher resolution (50Hz) spectral coefficients may then be grouped to approximate not only a linear filter bank but also a logarithmically spaced filter bank. The obvious disadvantage is increased transform length.

The DFT characteristics for the cepstral computation depend on the desired pitch detector resolution and range. In speech, the maximum range of pitch period likely to be encountered is from 20ms (50Hz) to 2ms (500Hz) [11]. This is rather a wide range and most pitch detectors operate on reduced limits e.g. 14.3ms (70Hz) to 2.5ms (400Hz). Typically, a 6-bit word is used to represent the logarithmically coded pitch data i.e. 64 resolution cells. The resolution at the short period (high frequency) end of the range is in the region of 0.1ms and, for the longest periods (low frequencies), is about 1ms [94].

For a cepstral processor to detect pitch periods of up to 20ms, a 40ms segment of speech must be analysed. Since the cepstrum has linear period resolution, the maximum resolution (0.1ms) has to be provided; a logarithmic scale may then be approximated by grouping together the high resolution bins. These requirements imply the use of a 400-point DFT in the cepstral computation.

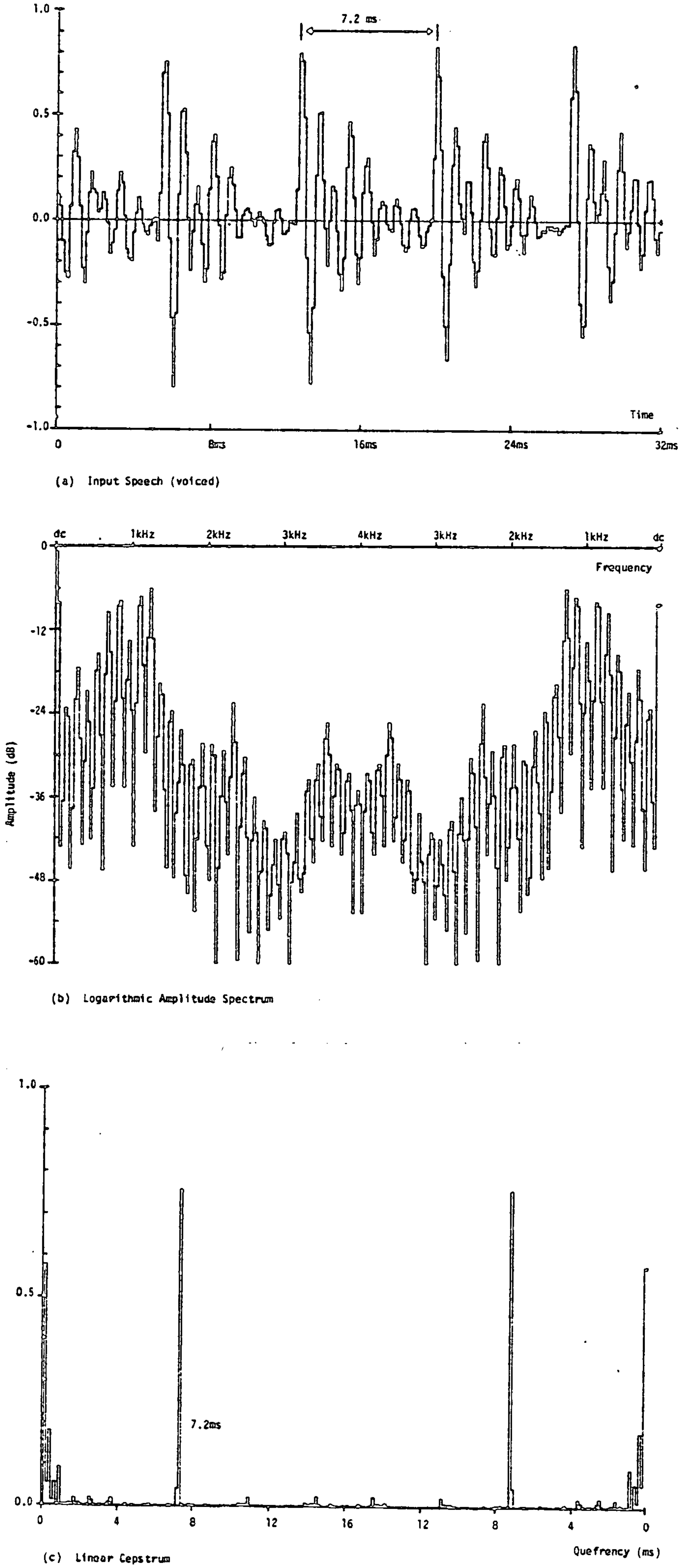
The discussion so far has ignored the effect of (sin x)/x "leakage" in the DFT (section 4.2.1), wnich limits the inherent amplitude resolution to -13dB. In applications where non-linear operations (e.g.modulus) are performed in the frequency domain, it is common practice to employ a weighting function (section 4.2.3) to trade frequency resolution for amplitude resolution. Since non-linear operations are involved in both of the above computations, a weighting function is necessary.

Practical experience in analogue DFT processors (chapter 5) has shown that -40dB is a realistic Peak Error to Signal ratio (PE/S); it would therefore be wasteful in terms of frequency resolution to employ a weighting function giving a much greater amplitude resolution. In addition, a resolution of 40dB is sufficient for the human ear. (Note that amplitude resolution is not the same as dynamic range). The best weighting function for this application is therefore the Hamming window (Equ.4.12) which provides a theoretical amplitude resolution of 43dB and decreases the 3dB frequency resolution by a factor of 1.3. As explained in section 4.2.3, the use of a weighting function leads to loss of data at the window edges and overlapping techniques are necessary.

Bearing in mind the decreased frequency resolution imposed by the weighting function, it is necessary to compromise the DFT characteristics for both the spectrum and cepstrum computations. Consider a 256-point DFT operating on a 32ms segment of Hamming weighted speech (f_{sample} = 8kHz). The nominal frequency resolution is 31.25Hz which is decreased to 40.6Hz by the weighting function. This is (section 2.3) certainly sufficient for spectral envelope representation. Pitch periods in the range 0 to 16ms may be detected from the cepstrum with a resolution of 0.125ms. Although the full pitch range (2 to 20ms) is not covered, the above DFT characteristics permit a very useful analysis of the speech waveform.

In the United Kingdom, a standard frame rate for updating the synthesiser control parameters is once every 20ms [27]. To facilitate independent testing of the analyser and synthesiser, this frame rate is also chosen here. The DFT input speech therefore consists of 32ms segments each overlapped by 12ms, a percentage overlap of 37.5%. The frame to frame correlation for different overlaps is given in Ref.[77].

Fig.6.5 illustrates an example of the spectrum and cepstrum of a voiced segment (part of the vowel "I"). The unweighted speech in (a) has a pitch period of 7.2ms and the corresponding spectrum in (b) has a line spacing of 139 Hz.



(a) Input Speech (voiced)

(b) Logarithmic Amplitude Spectrum

(c) Linear Cepstrum

Fig.6.5 Example of Spectrum and Cepstrum Computation

The cepstrum in (c) has a large peak at quefrency of 7.2ms.

To demonstrate the overall performance of the cepstral processor, Fig.6.6 shows a 3-dimensional cepstrogram (amplitude-quefrency-time) of the phrase "I was stunned". Each plane in the time axis is separated by 20ms. The fundamental peak is clearly visible during the voiced segments and, at the end of "stunned", the rapid pitch inflexion is tracked.

In order to maximise the efficiency of the proposed hardware, initial simulations were performed using the sliding CZT (section 4.4). For the sliding transform to operate correctly, the input data have to appear stationary from frame to frame. The simulation showed that the sliding transform distorted and smeared the speech spectra. Fig.6.7 illustrates several frames of log power spectra obtained via the sliding CZT. (Note that Fig.6.7 was generated by an unweighted 128-point sliding CZT from speech sampled at 6.4kHz. This was part of an earlier simulation). Each spectral frame should be symmetrical about 3.2kHz but, as can be seen, several frames are slightly skewed and distorted. Comparison with a direct transform of the same segment shows that there is a definite smearing effect due to the operation of the sliding CZT. Rapid changes in the speech waveform e.g. from silence to speech, create the worst conditions and the spectral output is reflected after

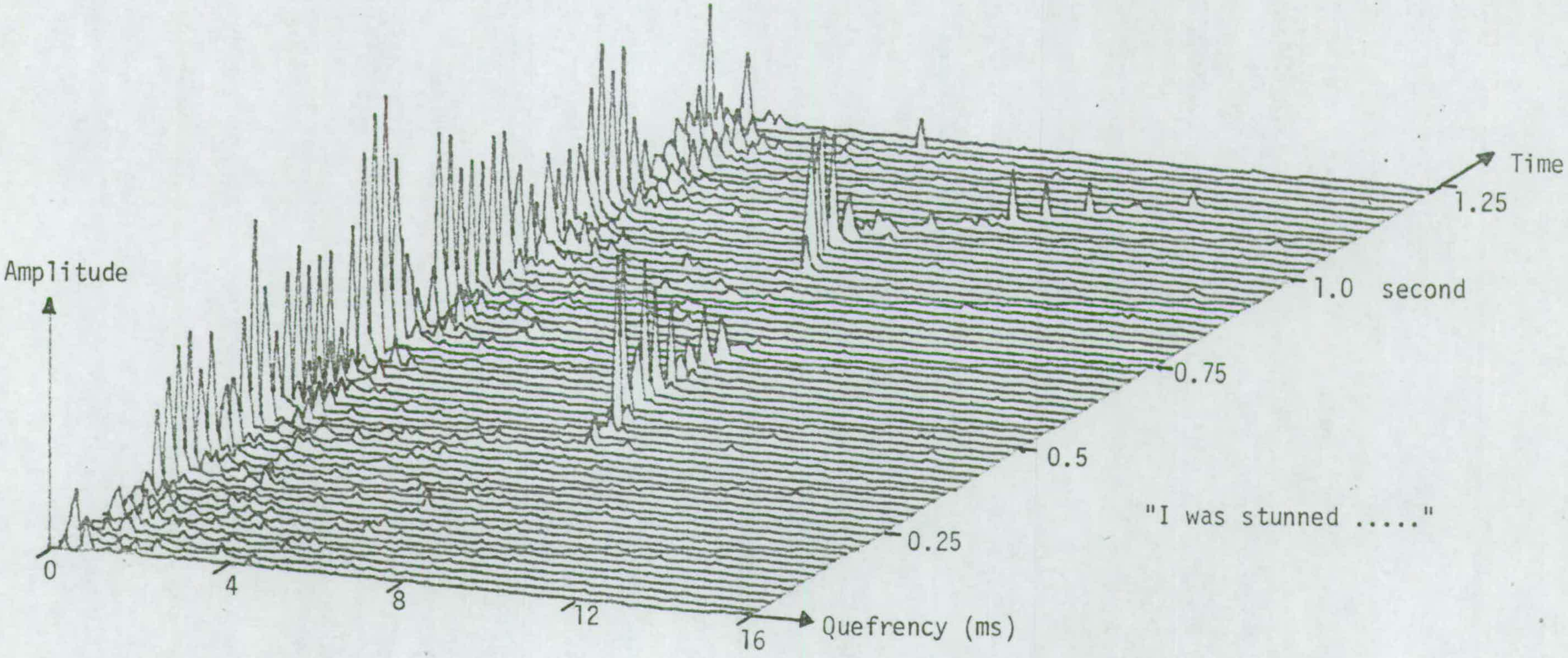


Fig.6.6 Three-dimensional Cepstrogram

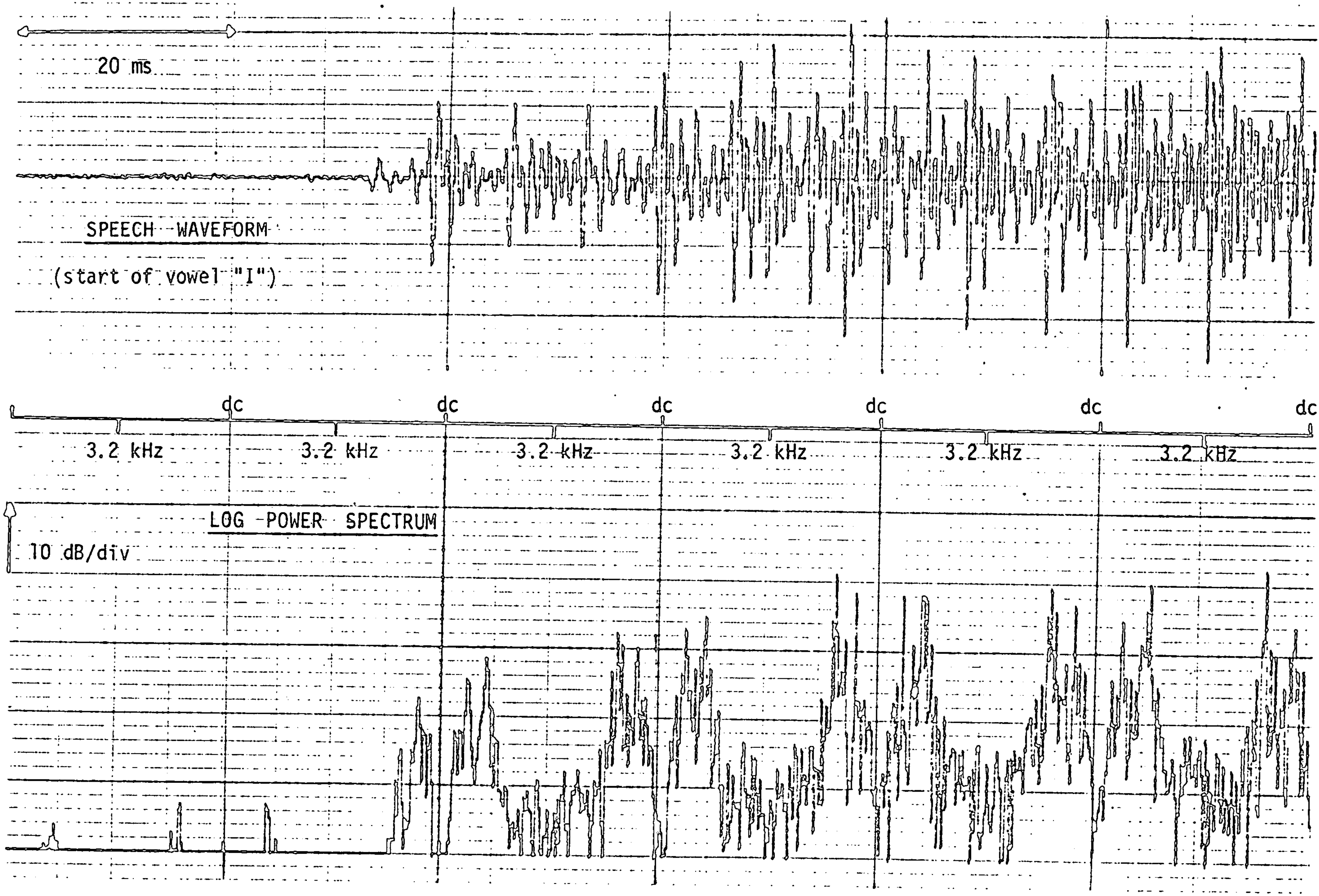


Fig.6.7 Sliding Transform of a Segment of Speech

a one frame delay. This is because the sliding transform convolvers have to be loaded with the signal before the transform is calculated. The correct results are obtained only during long vowels, implying that, in general, speech cannot be considered stationary for a 20ms frame rate. It is difficult to place any quantitative figure on the performance of the sliding algorithm in this application, but it is clear that the errors will cause severe distortion in synthetic speech. For this reason and others concerning hardware implementation (chapter 7) a direct transform is used in the following simulation work.

(It was originally intended to incorporate CZT hardware errors in the channel vocoder simulation. However, the time domain simulation of the CZT required approximately 20 minutes to process one second of speech. To conserve processing time and cost, an FFT algorithm (60 times as fast) was used in place of the CZT. The results are identical for the ideal case).

6.2.3 Data Reduction and Quantisation

The desired end product from this stage of the analyser processing is a digit stream which represents a quantised version of the speech spectral envelope. For an output data rate of 2400bps, each 20ms frame of spectral data has to be compressed into about 40 bits of information (2400bps = 48 bits per 20ms frame). The remaining bits in each frame are used for excitation source data.

The first step is to obtain a smoothed version of the speech spectrum (i.e. remove the spectral lines). Initial solutions to this problem made use of a low-pass filter in the frequency domain. The speech spectrum resulting from the DFT processor is in effect a time series representing frequency and tne filter is designed to remove the faster varying line components. The filter used in the simulation was a 51-tap FIR optimal filter designed using the Remez algorithm [67]. The number of taps was chosen to be odd so that the group delay was an integral number of clock periods and the cut-off frequency was selected to be 0.1 f_{sample}. A typical output from this filter is shown in Fig.6.8b (the appropriate group delay has been equalised). The formant structure of the vocal tract is now much clearer. An equivalent filtering operation could have been implemented using an accumulate and dump type of algorithm.

The spectral envelope in Fig.6.8b is grossly oversampled and it was chosen to down sample each spectral envelope to give 16 analogue frequency samples. These samples are equivalent to the output from a contiguous filter bank with 16 linearly spaced, equal bandwidth filters. Since the human ear is logarithmically sensitive

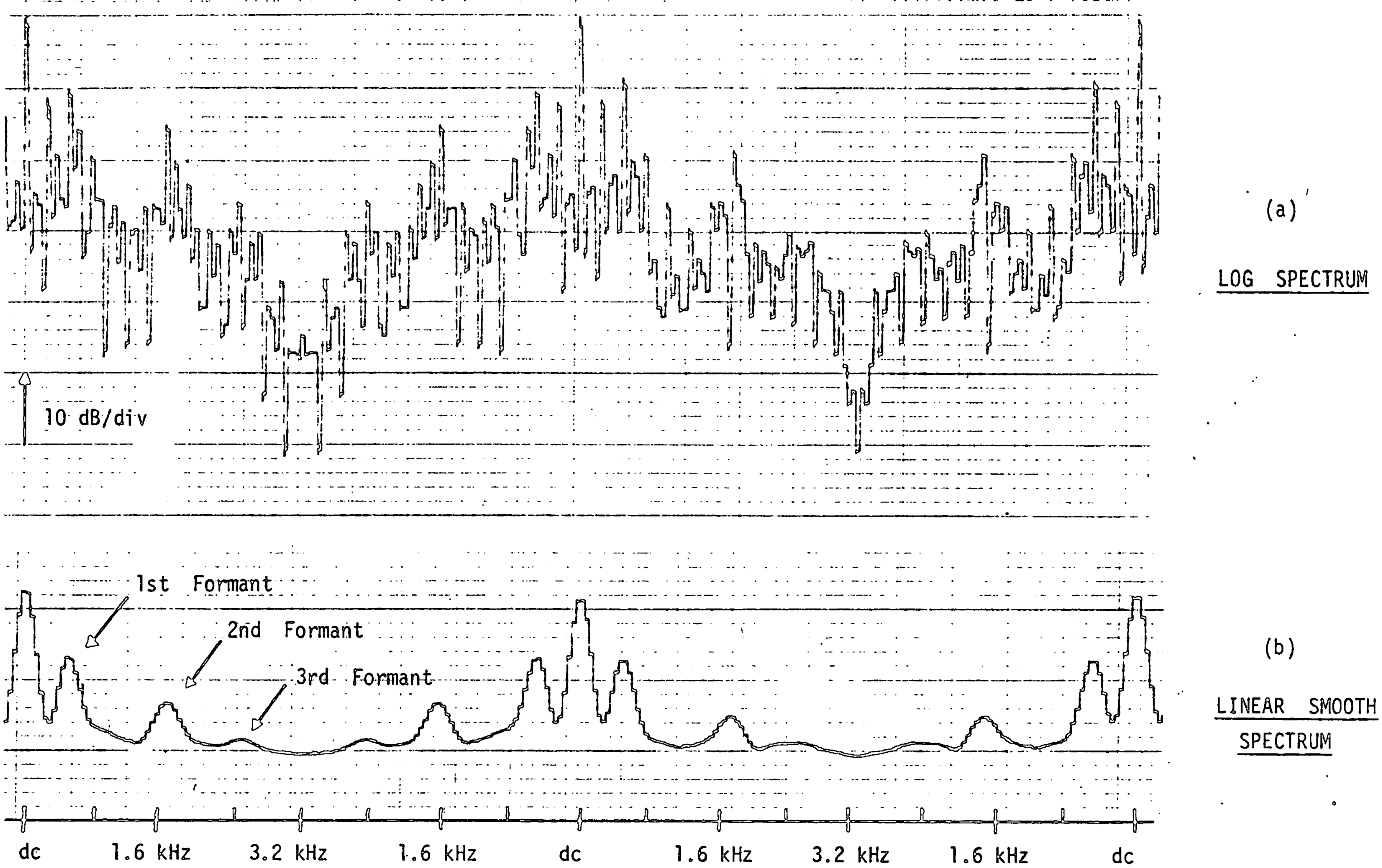


Fig.6.8 Spectral Smoothing (Low-pass Filter)

to amplitude the spectrum can be further compressed. The first 9 frequency points in each frame (low frequencies) were quantised to 3 bits each at 5dB per step and the remaining seven points were quantised to 2 bits at 10dB per step, allowing for a dynamic range of 40dB.

Subsequent synthesis from these data using techniques similar to those described in section 6.3 proved that the speech was of rather poor quality. Because the analyser and synthesiser were both new implementations, it was extremely difficult to pinpoint where the distortions originated; the bit stream between the analyser and synthesiser could not be compared against any standard. The poor quality appeared to be due to several factors, the most serious of which was the lack of frequency resolution at the low frequency end and the limited dynamic range. For these reasons, it was decided to adopt a logarithmic frequency compression method to allow direct comparison with an established hardware vocoder.

The channel vocoder available for experimental use, called the 'Marvox' was designed by the Joint Speech Research Unit, Cheltenham. This vocoder has logarithmically spaced 2nd order Butterworth filters as detailed in Table 2.I. Although the filter 'type' is not critical, the individual filters should be flat-topped and roll-off gently at the band edges [25]. The incorporation of this filter bank into the simulation may be accomplished in two ways:

1. time domain windowing before the DFT (one for each filter characteristic)

2. frequency domain convolution (one for each filter characteristic).

The first technique makes extremely inefficient use of processing power and is not a practical solution. The second does however have some attraction because each filter characteristic may be represented (to a first approximation) by relatively few weighting points.

The weighting values were obtained by evaluating each filter characteristic at frequencies corresponding to the DFT picket fence (multiples of 31.25Hz) and then quantising these values to the nearest dB. Only the most significant 20dB of each characterisic was considered. The equivalent filter outputs, F_k, are produced by summing the appropriately weighted frequency samples according to the relationship

$$F_k = \sum_{m=i}^{j} A(mR)\ 10^{-H_k(m-i+1)/20} \qquad \ldots (6.2)$$

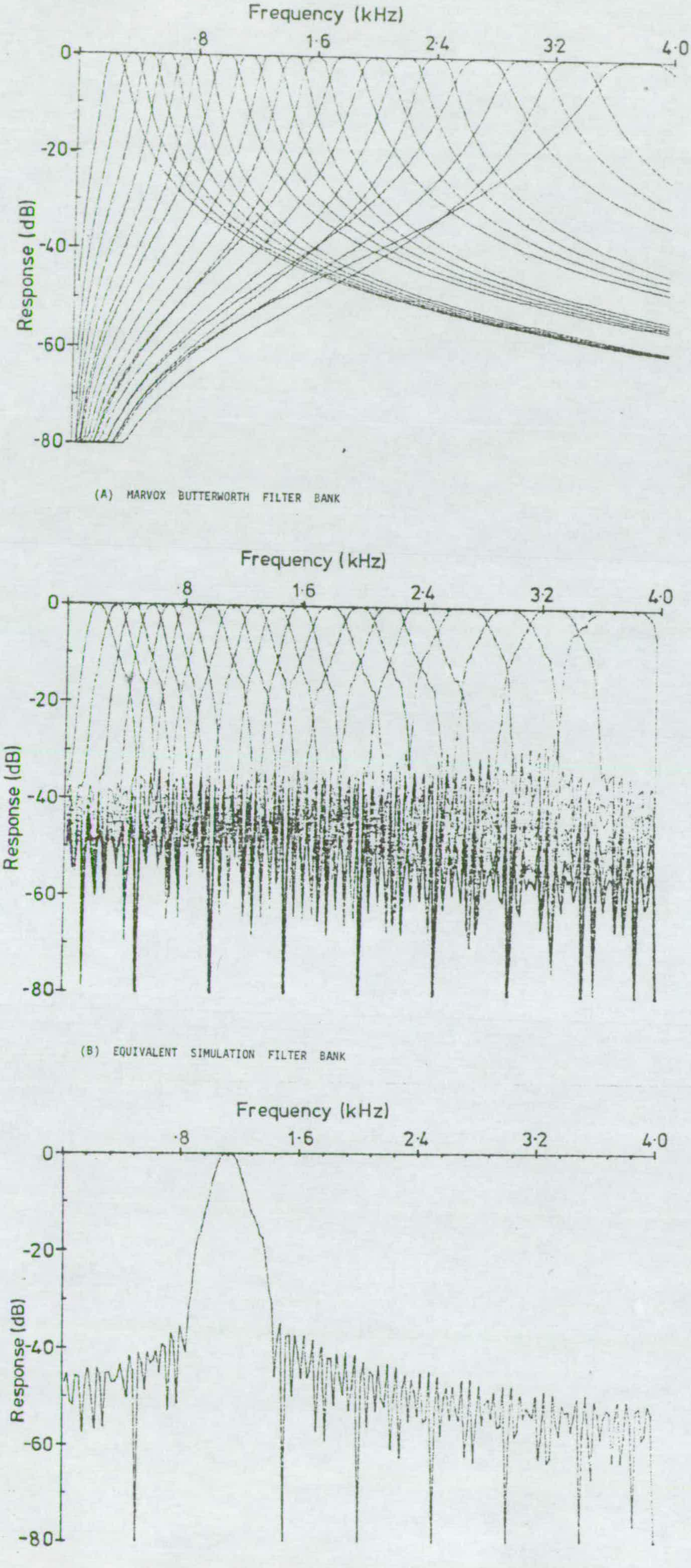
where R is the DFT resolution (31.25Hz), A is the spectral amplitude at the mth resolution bin and H_k are the logarithmic weighting coefficients as defined in Table 6.1.

k	i	j	H_k																			
1	4	15	18	10	3	0	0	1	4	8	11	14	16	18								
2	7	19	20	14	8	2	0	0	1	5	9	12	15	18	20							
3	11	22	17	12	6	1	0	0	2	6	10	13	16	19								
4	15	26	16	11	5	1	0	0	2	7	11	14	17	20								
5	18	29	19	15	10	4	0	0	0	3	7	12	15	18								
6	22	33	18	14	9	3	0	0	0	4	8	12	16	19								
7	26	40	18	14	10	6	2	0	0	0	2	5	9	12	15	17	19					
8	30	45	20	17	14	9	5	1	0	0	0	2	6	9	13	15	18	20				
9	35	49	19	16	13	9	4	1	0	0	0	3	6	10	13	16	18					
10	40	54	18	15	12	8	3	0	0	0	1	4	7	11	14	17	19					
11	45	59	18	15	11	7	2	0	0	0	1	4	8	12	15	17	19					
12	49	68	19	16	14	11	8	5	2	0	0	0	0	1	3	6	9	11	14	16	18	19
13	55	74	19	17	15	12	9	6	3	1	0	0	0	1	2	5	8	11	13	15	17	19
14	62	81	18	16	13	10	7	4	1	0	0	0	0	1	4	7	10	12	14	16	18	20
15	68	87	19	17	14	11	8	5	2	0	0	0	0	1	3	6	9	11	14	16	17	19
16	77	96	13	11	9	7	5	3	1	0	0	0	0	0	0	1	2	4	6	8	10	11
17	87	106	12	10	8	6	4	2	1	0	0	0	0	0	1	2	3	5	7	9	11	12
18	96	115	13	11	9	7	5	3	1	0	0	0	0	0	0	1	2	4	6	8	10	11
19	109	128	8	6	5	4	3	2	1	0	0	0	0	0	0	0	0	0	0	1	2	2

Table 6.1 Analyser Filter Coefficients

The amplitude transfer characteristic of the conventional filter bank is compared in Fig.6.9 with that produced by the above technique. It can be seen that the approximation matches closely within the first 20dB and then each filter falls off to a sidelobe level of -36dB. An individual filter (f_{centre} = 1150Hz) is shown separately in Fig.6.9c for clarity.

For the analyser computer simulation to be matched to the conventional Marvox synthesiser, the frame coding details (Fig.6.10) for each must be identical. The nineteen filter outputs are logarithmically quantised and compressed into 39 bits. The lowest frequency channel is coded as 3 bit direct PCM with 6dB/step (48dB dynamic range) and the remaining eighteen channels are each represented by two bit delta modulation. The delta modulation scheme permits more efficient coding by utilising the correlation between spectral channels. The 39 bits of spectral data are combined with two engineering bits for testing, 6 bits of pitch data (in reverse order) and one Voiced/Unvoiced (V/UV) bit to make a total of 48 bits per 20ms frame. Immediately before transmission, the final five bits in each frame are inverted for synchronisation in the receiver.

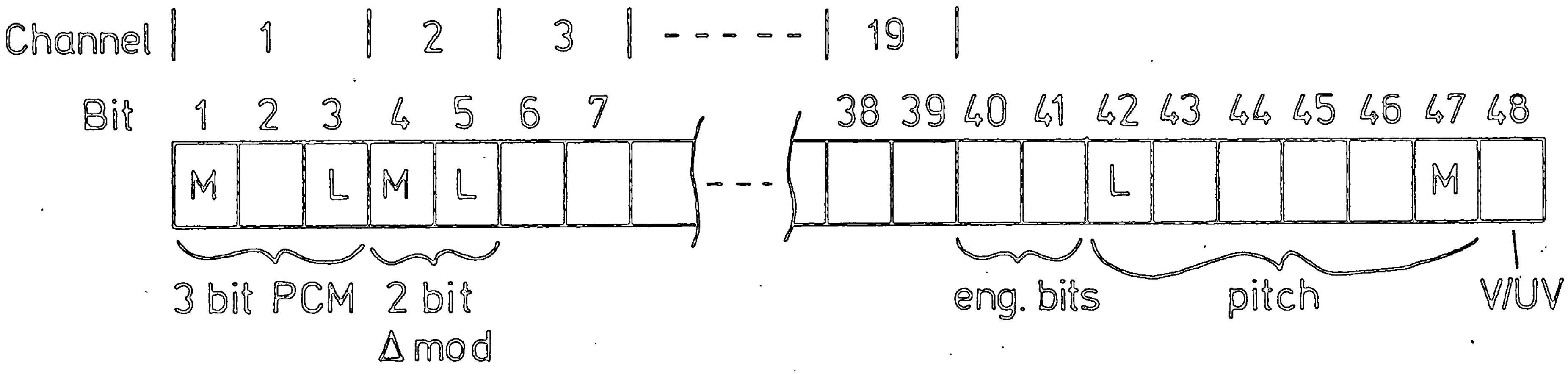


(A) MARVOX BUTTERWORTH FILTER BANK

(B) EQUIVALENT SIMULATION FILTER BANK

(C) RESPONSE OF CHANNEL 8 FILTER (Simulation)

Fig.6.9 Comparison of Analyser Filter Banks



NOTES : (1) 3 bit PCM code 111=0dB , 000=-42dB i.e. 6dB steps covering 48dB dynamic range

(2) 2 bit Δ Mod Code 11=+9dB , 10=+3dB , 01=-3dB , 00=-9dB (MSB represents +6dB , LSB represents +3dB)

(3) Pitch code in reverse order i.e. LSB first

(4) V/UV = 1 for voiced , V/UV = 0 for unvoiced

(5) Bits 44 to 48 are inverted prior to transmission for sync. purposes

(6) M = Most Significant Bit , L = Least Significant Bit

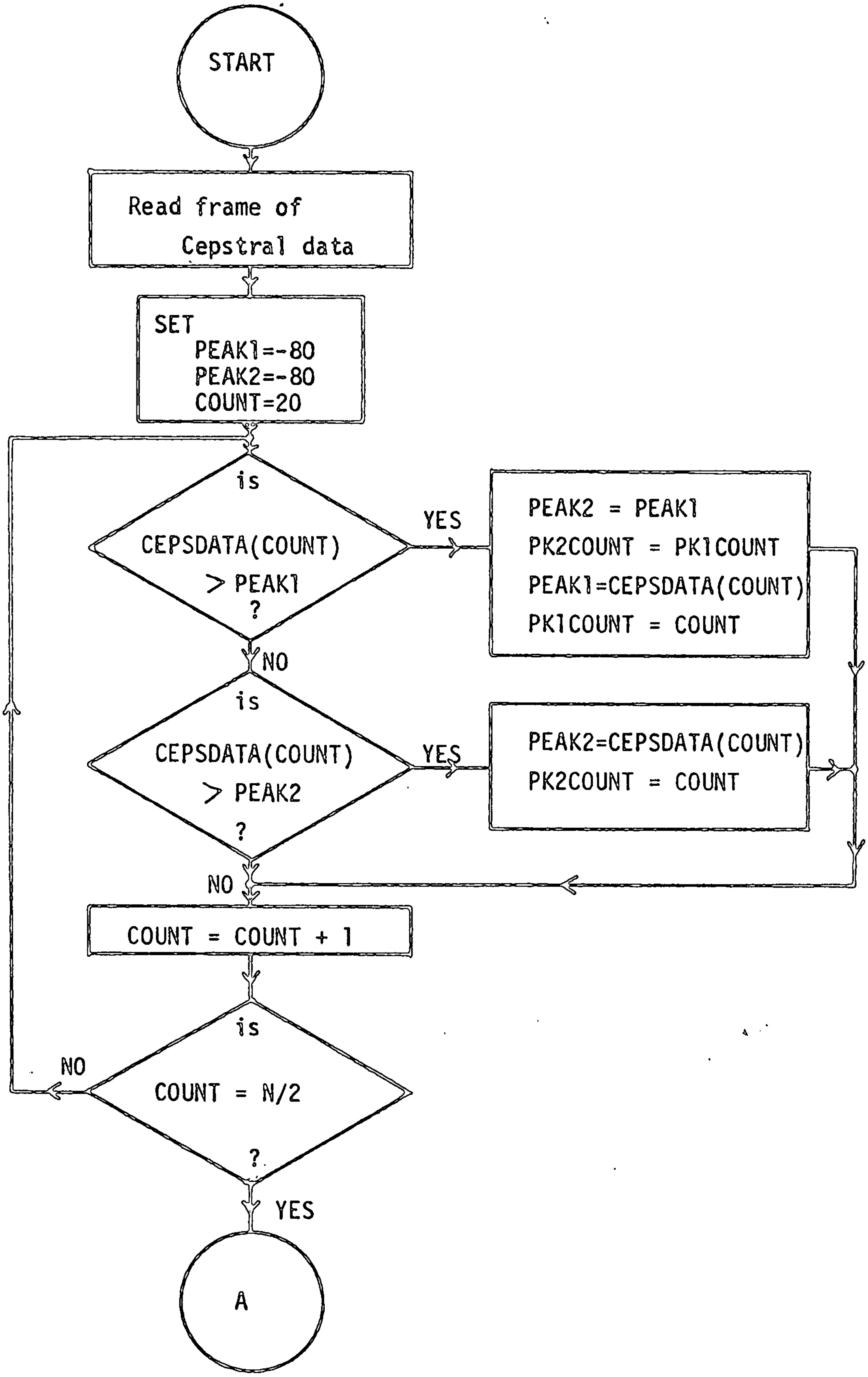
Fig.6.10 Channel Vocoder Frame Format

6.2.4 Pitch Extraction

The pitch extractor makes use of a peak picker to scan the cepstra and detect the peaks which are most likely to have been caused by the speech pitch periodicity. After a peak has been located, its position within the frame is converted into a digital word which is passed to the output stages for transmission. If a peak with the correct characteristics cannot be found, the peak picker assumes that the speech is unvoiced and a binary zero is transmitted.

One of the first cepstral peak picking algorithms was developed by Noll [21] and, although the algorithm works well, it is too involved for efficient hardware implementation. Two alternative algorithms have therefore been developed for the peak picker. The first (referred to as algorithm A) is extremely simple but does not eliminate spurious effects whereas the second (referred to as algorithm B) is more sophisticated and can consequently operate with a higher fidelity.

Both algorithms employ an identical peak detection stage and its operation is illustrated by the flow diagram in Fig.6.11. The result produced is the magnitude and position of the largest and second largest peaks within a


START
Read frame of Cepstral data
SET PEAK1=-80 PEAK2=-80 COUNT=20
is CEPSDATA(COUNT) > PEAK1 ?
YES
PEAK2 = PEAK1
PK2COUNT = PK1COUNT
PEAK1=CEPSDATA(COUNT)
PK1COUNT = COUNT
NO
is CEPSDATA(COUNT) > PEAK2 ?
YES
PEAK2=CEPSDATA(COUNT)
PK2COUNT = COUNT
NO
COUNT = COUNT + 1
is COUNT = N/2 ?
NO
YES
A

restricted portion of the cepstrum. The lower quefrencies are disregarded because they are outside the expected range of pitch periods and in this example the lower limit is chosen to be 20*0.125=2.5ms (400Hz). In addition, the second N/2 points in each frame are rejected, being a reflection of the first N/2 points.

After the two major peaks have been located, various tests are performed to ascertain whether or not the largest peak indicates a meaningful pitch period. In the simplified algorithm (algorithm A), Fig.6.12, the magnitude difference between the two peaks is calculated and if this exceeds 12dB, the speech is classed as voiced and the position of the largest peak is output. If this test is unsuccessful, the position of the second peak is compared with that of the first to determine whether or not the pitch peak has fallen between two resolution cells of the DFT processors. All peaks which fail both of the above tests are classed as unvoiced.

As it stands, algorithm A cannot distinguish spurious peaks. A decision of this type can be made only if information concerning the history of the pitch contour has been stored. The human pitch varies slowly and smoothly and it is therefore possible to predict the next pitch period based on past experience. Information concerning the cepstrum which immediately follows the present frame is also

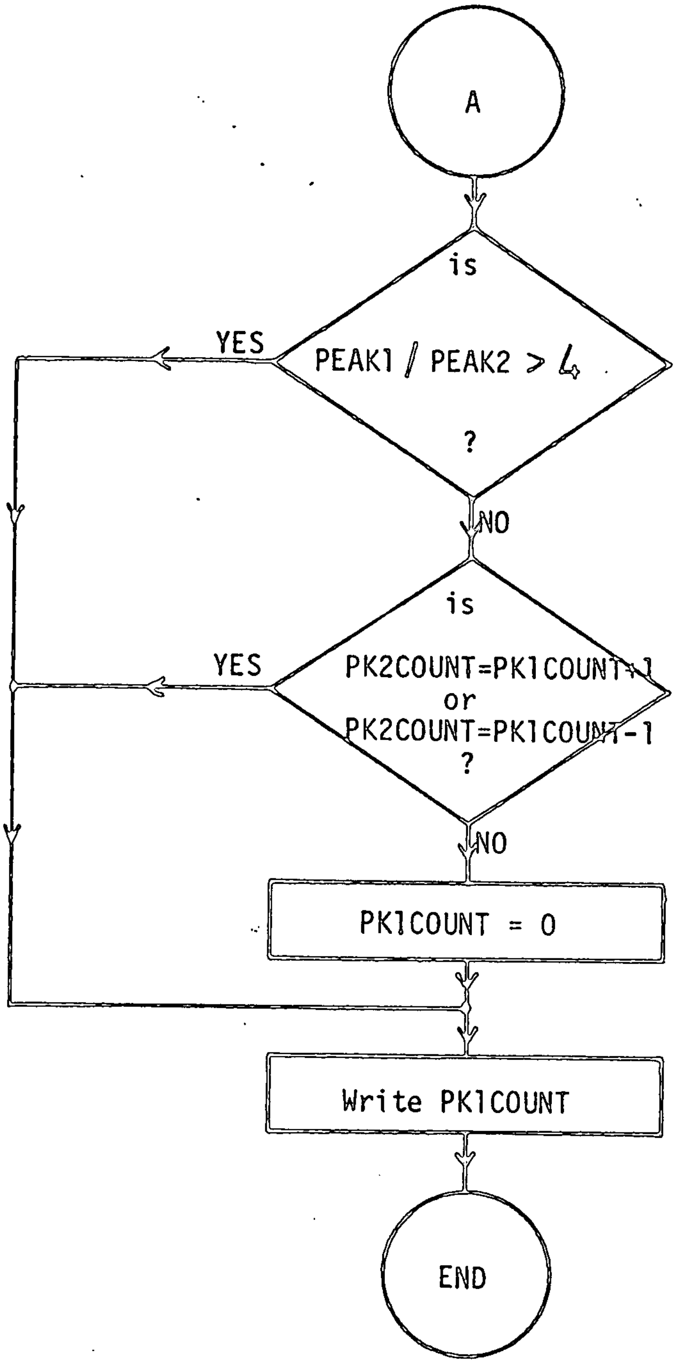


Fig.6.12 Simple Peak-picker Decision Algorithm

helpful. (Note that 'looking into the future' implies a one frame delay in the system). For example, if the cepstra on either side of the present frame indicate unvoiced speech, it is most unlikely that the present frame is voiced.

The following modifications to the simple algorithm help to eliminate spurious pitch estimates. During voiced speech the modified algorithm rejects pitch periods which do not appear within 10 resolution cells (1.25ms) of the previous pitch measurement. If there is a large peak in the cepstrum outside the above limits, it is assumed that the speech is voiced and the previous pitch estimate is repeated. Unfortunately, this modification leads to an undesirable effect. On a change from voiced to unvoiced speech, the peak picker occasionally latches on to the previous voiced estimate because of spurious peaks in the unvoiced speech. To compensate for this, an additional V/UV indicator is incorporated, which compares the channel 19 filter output with that from channel 2. If the ratio exceeds 5, the speech is classed as unvoiced and if the ratio is less than 0.5, the speech is classed as voiced. Should the energy ratio fall between these two limits, the classification depends on measurements from the cepstrum. The flow diagram for this more complex algorithm (algorithm B) is given in Fig.6.13.

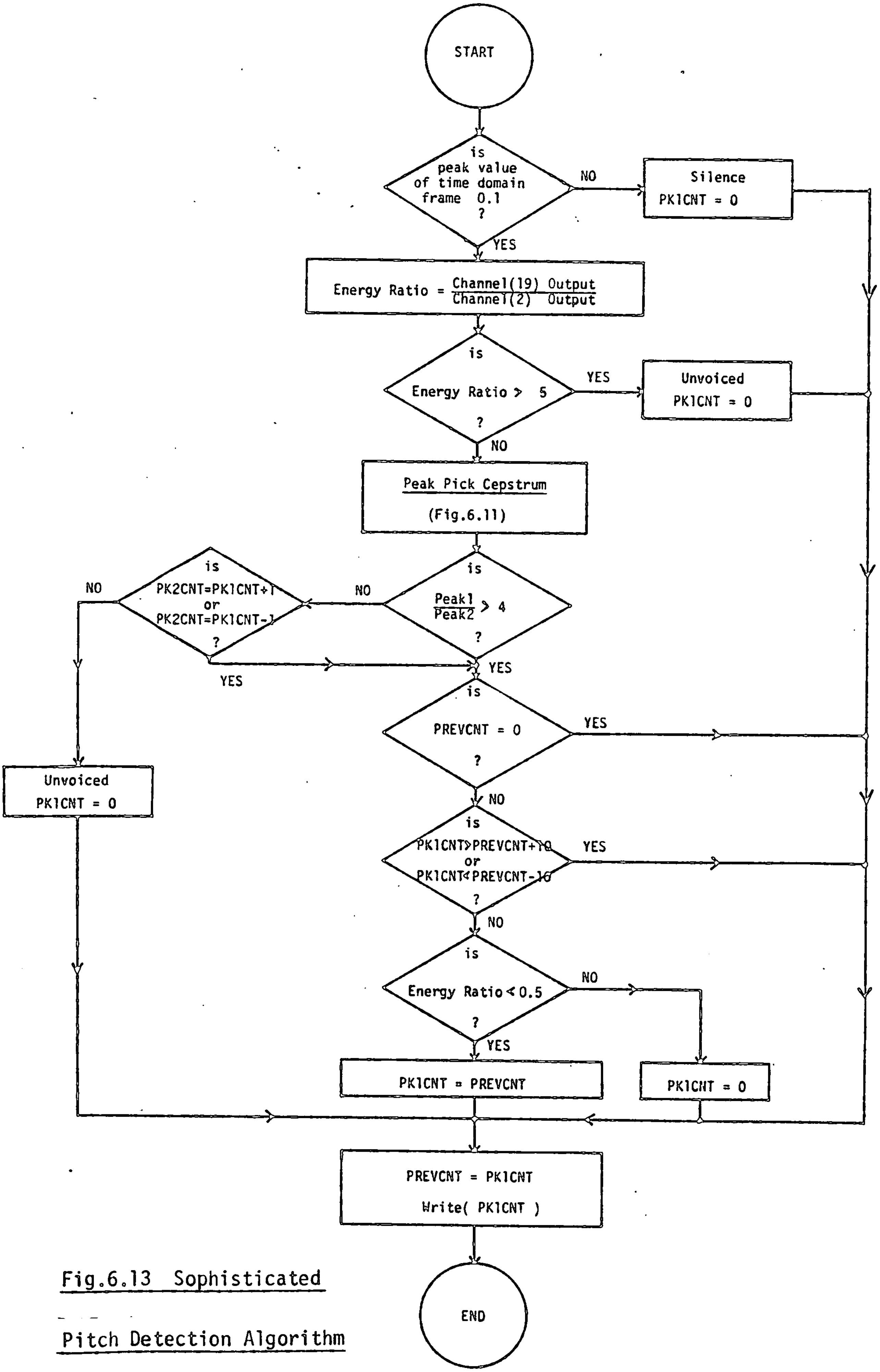

START
is peak value of time domain frame 0.1 ?
NO
Silence
PK1CNT = 0
YES
Energy Ratio = $\frac{\text{Channel(19) Output}}{\text{Channel(2) Output}}$
is Energy Ratio > 5 ?
YES
Unvoiced
PK1CNT = 0
NO
Peak Pick Cepstrum
(Fig.6.11)
is $\frac{\text{Peak1}}{\text{Peak2}}$ > 4 ?
NO
is PK2CNT=PK1CNT+1 or PK2CNT=PK1CNT-1 ?
NO
YES
Unvoiced
PK1CNT = 0
YES
is PREVCNT = 0 ?
YES
NO
is PK1CNT>PREVCNT+10 or PK1CNT<PREVCNT-10 ?
YES
NO
is Energy Ratio < 0.5 ?
NO
YES
PK1CNT = PREVCNT
PK1CNT = 0
PREVCNT = PK1CNT
Write(PK1CNT)
END

Fig.6.13 Sophisticated Pitch Detection Algorithm

For male input speech, the pitch extractor operated with very few gross errors. To assess the performance of the cepstral pitch extractor, the output data were compared with manually extracted pitch data from the time domain speech waveform, "I was stunned by the beauty of the view". Out of a total of 192 frames of speech data, 13 errors occurred, some of which were more serious than others. By far the most common error (8) was caused by rapid pitch inflexions, either at the beginning or at the end of words. When the pitch changes quickly in relation to the frame period, the cepstral peak disappears and the speech is classed as unvoiced. On two occasions, pitch inflexions during a voiced segment resulted in a similar classification. Apart from pitch contour smoothing, very little can be incorporated to prevent this effect. Because of the imposed dynamic range limit of 50dB in the simulation, one frame of low-level voiced speech was classed as unvoiced. It is possible that this may be improved by agc. One other error resulted from the peak picking process; the second rahmonic was selected in preference to the fundamental (pitch doubling) at the end of a voiced segment. This may have been a legitimate decision since visual inspection of the time domain waveform showed that a component at half the fundamental period was increasing in strength. However, synthetic speech produced from these data sounded rather unnatural. The remaining errors were due to the energy comparator selecting unvoiced instead of voiced during a voiced segment.

When the pitch extractor was input with female speech, a large number of errors occurred. Most of these were due to the high pitch frequencies produced by this speaker. On several occasions, the pitch contour exceeded the upper limit scanned by the peak picker (2.5ms) and, on others, the pitch peak became distorted by the formant information. These errors highlighted a serious defect in the cepstral pitch extractor; for pitch frequencies of greater than 400Hz, the excitation source and the vocal tract are not properly deconvolved, which makes pitch extraction very difficult. The majority of the remaining errors were caused by the energy comparator classifying voiced as unvoiced because, overall, this speaker had a larger high frequency content in the speech waveform. Readjustment of the V/UV comparator limits is necessary to alleviate this problem.

The Marvox channel vocoder employs a 6-bit code for representation of the logarithmically coded (lumped linear) pitch data. The 6-bit code covers four octaves of frequency ranging from 37.5Hz up to 600Hz with 16 levels per octave. Since the cepstrum gives linear resolution, an encoder is necessary to convert the cepstral pitch data to a logarithmic scale. The encoder is designed to map the cepstral resolution bins on to the nearest Marvox quantisation level and the mapping function used in this simulation is given in Table 6.2.

	Period (ms)		Period (ms)		Period (ms)		Period (ms)
0	1.7 (600Hz)	16	3.3 (300Hz)	32	6.7 (150Hz)	48	13.3 (75Hz)
1	1.8	17	3.5	33	7.1	49	14.2
2	1.9	18	3.7	34	7.5	50	15.1
3	2.0	19	3.9	35	7.9	51	16.0
4	2.1	20	4.1	36	8.3	52	16.9
5	2.2	21	4.3	37	8.7	53	17.8
6	2.3	22	4.5	38	9.1	54	18.7
7	2.4	23	4.7	39	9.5	55	19.6
8	2.5	24	4.9	40	9.9	56	20.5
9	2.6	25	5.1	41	10.3	57	21.4
10	2.7	26	5.3	42	10.7	58	22.3
11	2.8	27	5.5	43	11.1	59	23.2
12	2.9	28	5.7	44	11.5	60	24.1
13	3.0	29	5.9	45	11.9	61	25.0
14	3.1	30	6.1	46	12.3	62	25.9
15	3.2	31	6.3	47	12.7	63	26.8

"Marvox" Pitch Detector Code

"Marvox"	Cepstrum	"Marvox"	Cepstrum
0	0 to 14	27	45
1	15	28	46,47
2	16	29	48
3	17	30	49,50
4	18	31	51 to 53
6	19	32	54 to 56
7	20	33	57 to 59
8	21	34	60 to 62
9	22	35	63 to 65
11	23	36	66 to 68
12	24	37	69 to 71
13	25	38	72 to 75
14	26	39	76 to 78
15	27	40	79 to 81
16	28	41	82 to 84
17	29	42	85 to 88
18	30,31	43	89 to 91
19	32,33	44	92 to 94
20	34	45	95 to 97
21	35,36	46	98 to 101
22	37,38	47	102 to 105
23	39	48	106 to 111
24	40,41	49	112 to 118
25	42	50	119 to 125
26	43,44	51	126 to 129

Period = cepstrum * 0.125ms

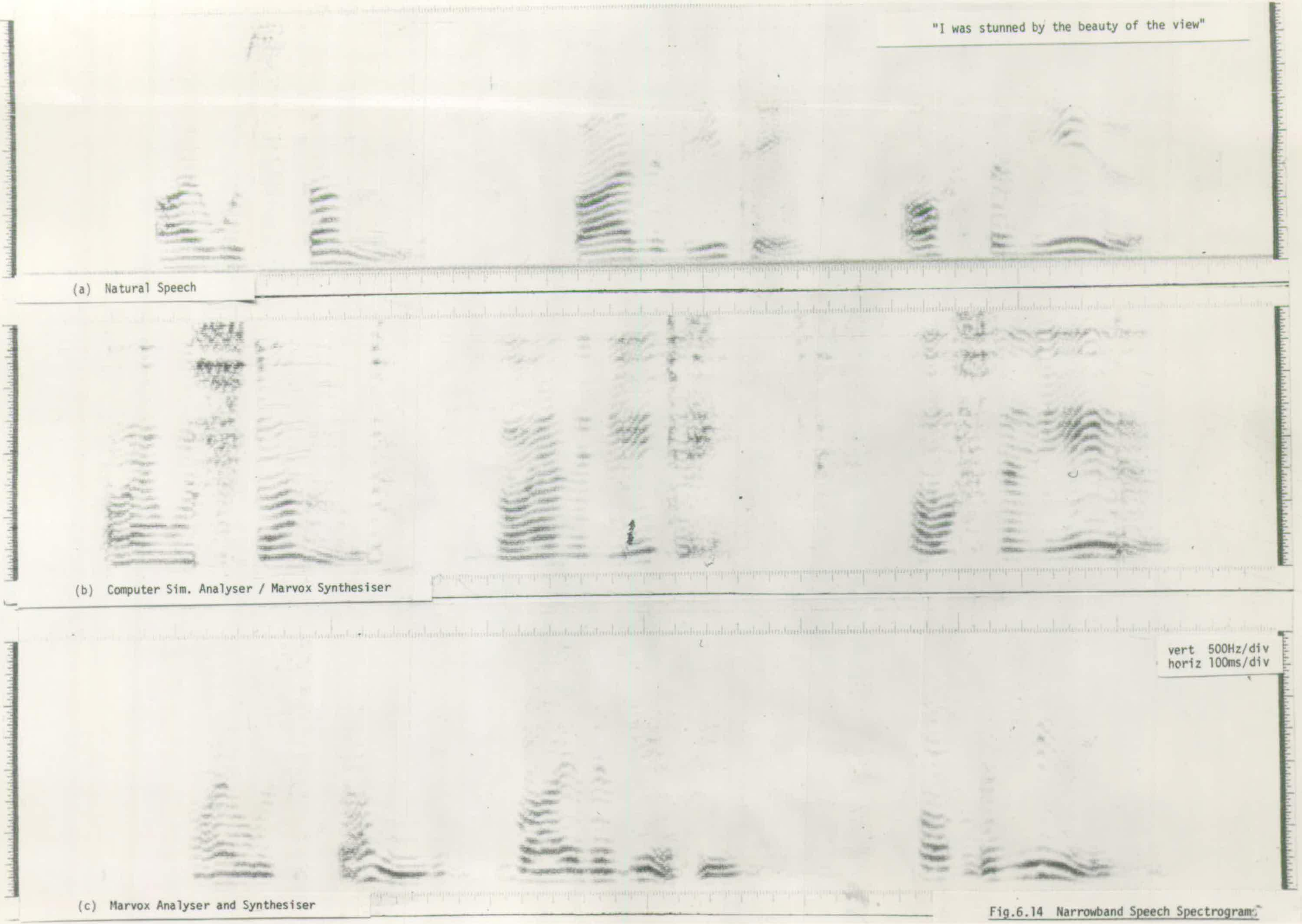
Cepstrum to "Marvox" Encoder

Table 6.2 Linear to Logarithmic Pitch Encoder

6.2.5 Performance Comparison

The channel analyser computer simulation was programmed to generate a serial data stream at 2400bps which could be input to a hardwired Marvox channel synthesiser. This facilitated a direct comparison between the parallel filter bank analyser and the computer simulation.

Fig.6.14 illustrates three narrowband (50Hz) spectrograms comparing the pitch variations in (a) the original speech (b) the synthetic speech generated by the computer simulation/Marvox synthesiser combination and (c) the synthetic speech produced by the Marvox analyser and synthesiser. (The sentence is divided into three separate segments because of the restricted computing facilities). It can be seen that the cepstral pitch detector out-performs the phase-locked loop in the Marvox analyser in several aspects. In particular, the pitch tracking is extremely good. For example, at the beginning of "beauty", the cepstrum follows the pitch which is starting to increase rapidly, whereas the phase-locked loop moves in the opposite direction. At the end of "beauty", however, the cepstrum classifies the "y" as unvoiced. Apart from this mistake, the V/UV decisions appear to be accurate, which is in contrast to the Marvox analysis, where the pitch contour almost breaks up during "I" and "by". (Note that the pitch tracking is actually aided by the Marvox synthesiser because



(a) Natural Speech

(b) Computer Sim. Analyser / Marvox Synthesiser

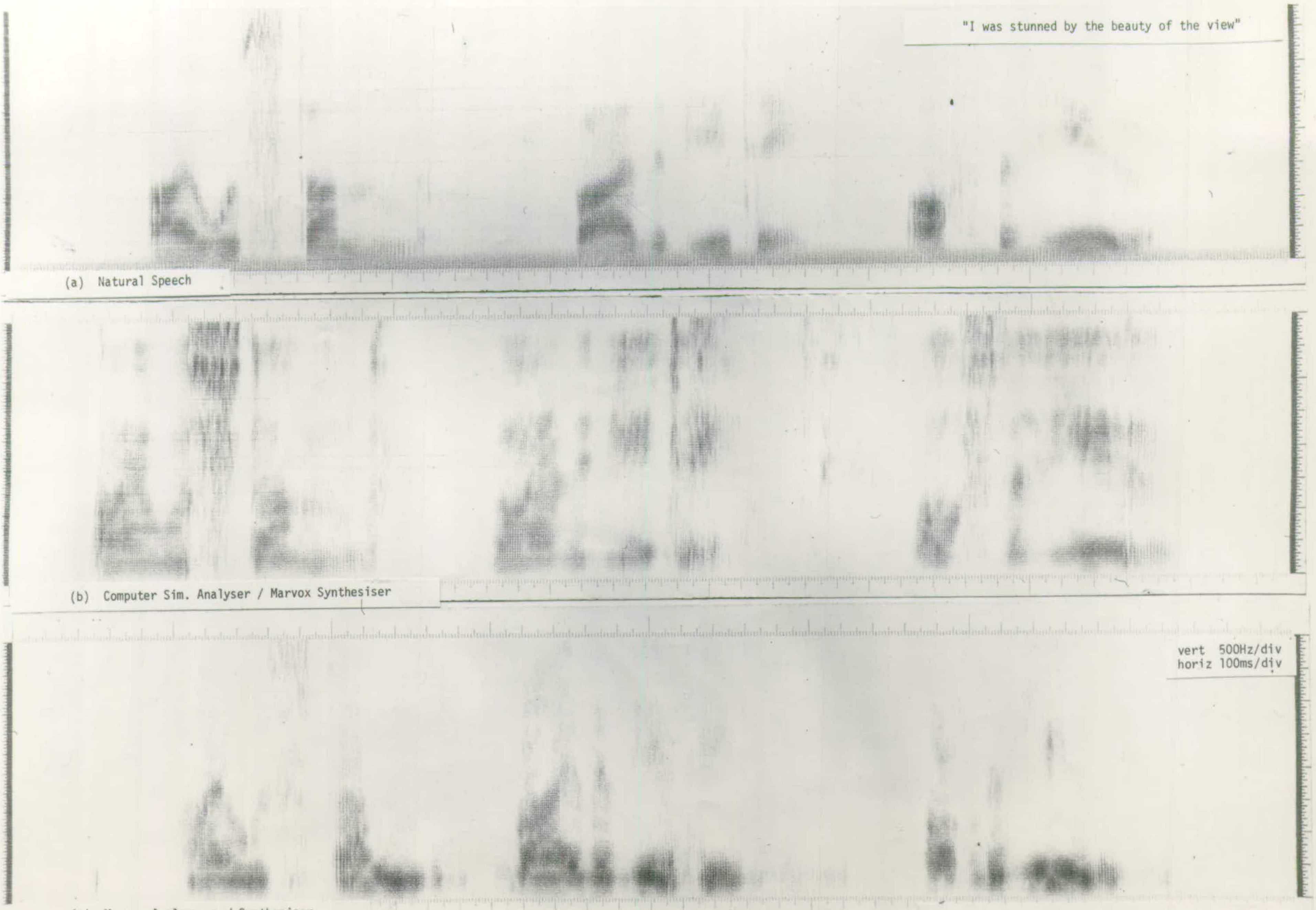
(c) Marvox Analyser and Synthesiser

Fig.6.14 Narrowband Speech Spectrograms

all pitch data are smoothed by a 5Hz, one-pole low-pass filter. This filter is switched out of circuit for an unvoiced to voiced transition).

Wideband analyses (200Hz) for the same speech segments are shown in Fig.6.15. These analyses permit the formant movements and bandwidths to be examined. As expected, both the computer simulation and the Marvox analyser produce similar results (apart from the stronger high frequency emphasis in Fig.6.15b). In comparison to the original speech, the synthetic formants are less well defined. This is due to the coarse spectral quantisation.

Informal listening tests indicated that the synthetic speech generated from the computer simulation was superior to that from the Marvox analysis. The main reason for this was the more accurate pitch extraction. Overall, the synthetic speech from the computer analysis had a sharper, more natural sound than the Marvox, although the intelligibility was good for both analyses. A sufficient quantity of data has not as yet been processed to permit full intelligibility testing using phonetically balanced word lists.



(a) Natural Speech

(b) Computer Sim. Analyser / Marvox Synthesiser

(c) Marvox Analyser and Synthesiser

Fig.6.15 Wideband Speech Spectrograms

6.2.6 Summary of Analyser Simulation Conclusions

The following conclusions were drawn from this section:

1. a sampling frequency of 8kHz permits sufficient analyser bandwidth

2. pre-equalisation should consist of +6dB/octave boost from 1kHz

3. the spectrum resolution should be at least 50Hz to allow logarithmic frequency compression (i.e. 160-point DFT processor)

4. to cover the full pitch range of 2ms to 20ms, the cepstrum resolution should be at least 0.1ms (i.e. 400-point DFT processor) and be averaged over a period of 40ms

5. a good compromise to (3) and (4) above may be achieved using a 256-point processor operating on a 32ms segment of speech

6. the sliding CZT is not useful for speech spectrum or cepstrum analysis

7. weighting functions are essential for both the spectrum and cepstrum calculations (the Hamming window is suitable)

8. to achieve a frame rate of 20ms, overlapping techniques are required in the DFT computation

9. standard channel vocoder data compression techniques may be employed in this implementation

10. for stand alone performance, the analyser should be designed to interface with conventional synthesiser designs

11. the cepstral pitch detector enables high fidelity channel vocoder operation.

6.3 CHANNEL SYNTHESISER SIMULATION

The synthesiser has to convert the received frames of quantised spectral data and excitation source information into a continuous synthetic speech waveform. Conventional synthesisers achieve this by filtering an excitation signal, generated from the control parameters, through a bank of contiguous band-pass filters whose overall transfer characteristic has been made to look like the vocal tract. In the implementation simulated here, this filter bank has been replaced by an inverse DFT processor.

The computer simulation block diagram is shown in Fig.6.16. The quantised spectral data received from the

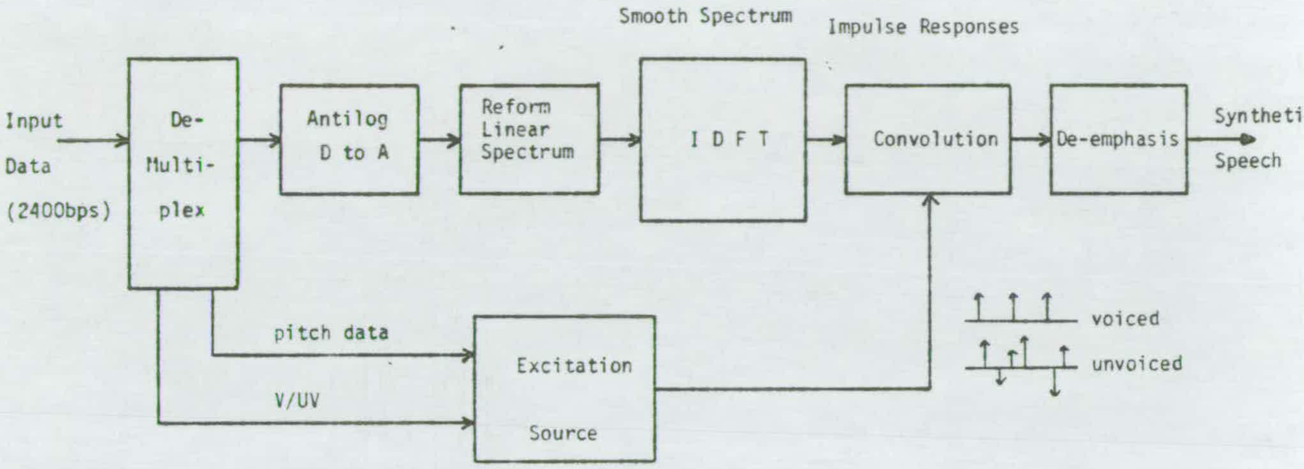


Fig.6.16 Synthesiser Simulation Block Diagram

demultiplexer are processed by operations, which are the inverse of those performed in the analyser, to form an impulse response function in each frame. These impulse responses are an approximate reconstruction of the vocal tract impulse response at certain discrete instants of time, and to regenerate the speech waveform requires the excitation signal for a particular period to be convolved with the correct impulse response.

6.3.1 Impulse Response Generation

The synthesiser receives one 48-bit frame of data every 20ms which is composed of 39 bits for spectral data, 6 bits for pitch, 1 bit for the V/UV control and two engineering bits. To generate an appropriate impulse response, the 39 spectral bits have to be demodulated (using the first channel 3-bit PCM as a reference), anti-logarithmically converted, reformed into a smooth spectrum and finally inverse transformed.

The inverse DFT (IDFT) does not demand as high a resolution as the forward DFT in the analyser because only the spectral envelope is to be transformed. However, to enable both the analyser and synthesiser to make use of the same central processor in a hardware implementation, a 256-point IDFT is chosen. The IDFT requires real and imaginary inputs, $\{R_k\}$ and $\{I_k\}$, of the form

$$R_k = A_k \cos(\emptyset_k) \qquad \ldots (6.3)$$

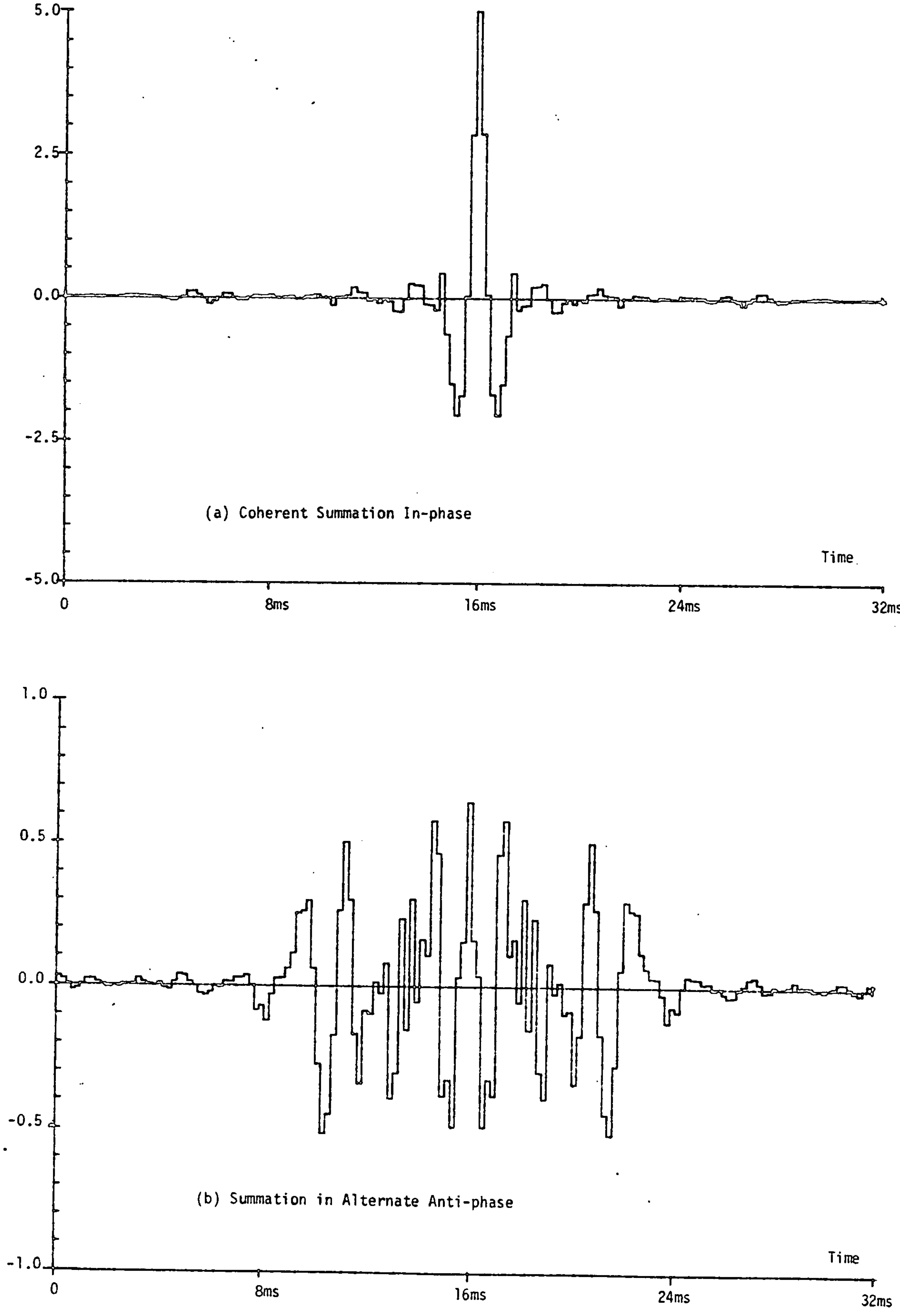
$$I_k = A_k \sin(\emptyset_k) \qquad \ldots (6.4)$$

where the sequence $\{A_k\}$ is the amplitude spectrum and the sequence $\{\emptyset_k\}$ is the phase spectrum. Since only the amplitude spectrum is transmitted to the synthesiser, the phase spectrum is arbitrary. If it is chosen that $\emptyset_k=0$, for all k, the IDFT inputs reduce to $R_k=A_k$ and $I_k=0$, and the resulting impulse response is termed zero phase. This zero phase impulse response is real and even and does not possess an imaginary part.

The 256-point amplitude spectrum (128 points reflected), must be reconstructed from the nineteen logarithmically spaced spectral components received by the synthesiser. In order to retain the characteristics of the conventional filter bank synthesiser, the reconstruction is achieved by summing together weighted versions of each filter amplitude transfer characteristic. The weighting factors are given by the received channel amplitudes. In the Marvox synthesiser, alternate filter outputs are summed in anti-phase to prevent coherent summation giving large spikes. The same can be achieved in this simulation by alternating the sign of each weighting coefficient. Fig.6.17 demonstrates the difference between summing in-phase and summing in alternate anti-phase.

Unfortunately, the band-pass filters in the Marvox synthesiser are not identical to those in the analyser, implying that an alternative set of filter coefficients have to be stored. Although the centre frequencies are the same, the filters are single tuned and the bandwidths are much narrower. The narrower bandwidths give a resonant quality to the synthetic speech. Also, during unvoiced synthesis, an additional wide-band filter is substituted for the channel 19 filter to provide extra high frequency energy.

The inverse transformation of a spectrum with 31.25Hz nominal frequency resolution results in an impulse response which lasts for 32ms. However, an impulse response which



(a) Coherent Summation In-phase

(b) Summation in Alternate Anti-phase

Fig.6.17 Illustration of Impulse Responses Generated by Coherent Phase Summation and Alternate Anti-phase Summation

lasts for only 20ms is necessary for the speech reconstruction algorithm described in section 6.3.3. This modification can be accomplished simply by truncating the 32ms (256 point) impulse response to 20ms (160 point). The same effect could have been achieved by reducing the frequency resolution to 50Hz. In addition, each impulse response is rotated by N/2 points (equivalent to a phase shift) to make the main peak appear in the middle of a frame, thereby reducing discontinuities when the impulse responses are convolved with the excitation source.

6.3.2 Excitation Sources

The two main excitation sources used in the production of human speech are (a) periodic impulses generated by the vocal cords (voiced) and (b) random noise created by air turbulence (unvoiced). The voiced excitation source in this simulation consists of a fixed amplitude pulse train with the pulse rate controlled by the pitch input data. The pulse width is equal to the time resolution of the system i.e. 20ms/160 = 0.125ms. Although the spectral decay in this signal is not matched to the human source, an appropriate compensation may be incorporated by a post-equalisation stage.

The random noise source is simply a random sequence of 1's and 0's and is selected by the V/UV control input.

6.3.3 Reconstruction by Convolution

Two alternative techniques are possible for the reinsertion of the excitation source data into the speech. The first is to multiply the smooth spectrum by the DFT of the excitation source before the inverse transformation is performed, and the second is to convolve the speech impulse responses with the excitation source in the time domain.

At first glance, the former technique appears to be simpler but, in fact, there are two major problems. Firstly, the DFT of the excitation source has to be generated. The obvious approach is to use an extra DFT processor but this would not be a desirable solution. Since the continuous transform of the voiced excitation source is an impulse train with repetition rate 1/T, where T is the pitch period, it seems likely that it would be relatively easy to generate this signal without the use of a DFT processor. However, the vocoder operates with finite and not infinite transforms, which means that a (sin x)/x picket fence would have to be multiplied into the impulse train. This procedure would require complex hardware. The only other way to implement this scheme would be to store the excitation signal transforms in ROM and multiply them into the spectrum through an MDAC. To store the appropriate signals for 64 different pitch periods would require approximately 8k bytes of ROM. The second problem with this frequency domain multiplication technique is that it would be very difficult to remove the frame to frame discontinuities in the resulting synthetic speech.

The alternative reconstruction method is the time domain convolution of the excitation source and the vocal tract impulse response. The convolution of a signal with a train of impulses is equivalent to the addition of delayed replicas of that signal with the delay equalling the impulse repetition rate. This operation may be conveniently implemented using a tapped CCD delay line with on-off switches at each tap.

The first attempt using this technique is shown in Fig.6.18. Here, the convolution reference is calculated from the pitch data and the resulting sequence of 1's and 0's loaded into the tap switches, starting at the first tap. One 20ms frame of impulse response is then clocked through the delay line before a new excitation sequence is loaded. The synthetic speech in Fig.6.18 demonstrates the main defect of this fixed convolution approach. Discontinuities in pitch arise because the delay between the last excitation impulse of one frame and the first of the next frame is not an exact pitch period but is the remaining delay after an integral number of pitch periods have been subtracted from

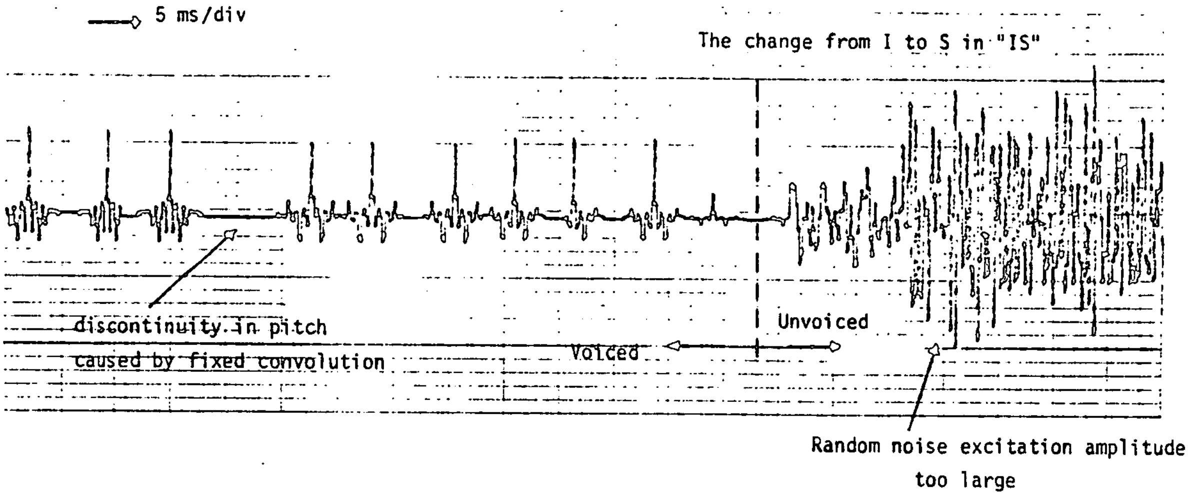


Fig.6.18 Fixed Convolution of Impulse Responses (Handset processing)

the fixed frame delay. A scheme is therefore required which takes into account the pitch period from the previous frame.

The flow diagram for an improved reconstruction algorithm is given in Fig.6.19. The controlling element in this scheme is a counter which is clocked synchronously with the CCD delay line and is reset at the start of each frame. In other words, the counter effectively follows the leading edge of each frame as it moves along the delay line register. At each address, the delay line tap switch is interrogated and, if the switch is found to be closed, it is opened. The counter address is then compared with the "next pulse" address and, if there is agreement, the tap which has been closed remains closed for a full cycle of the counter,

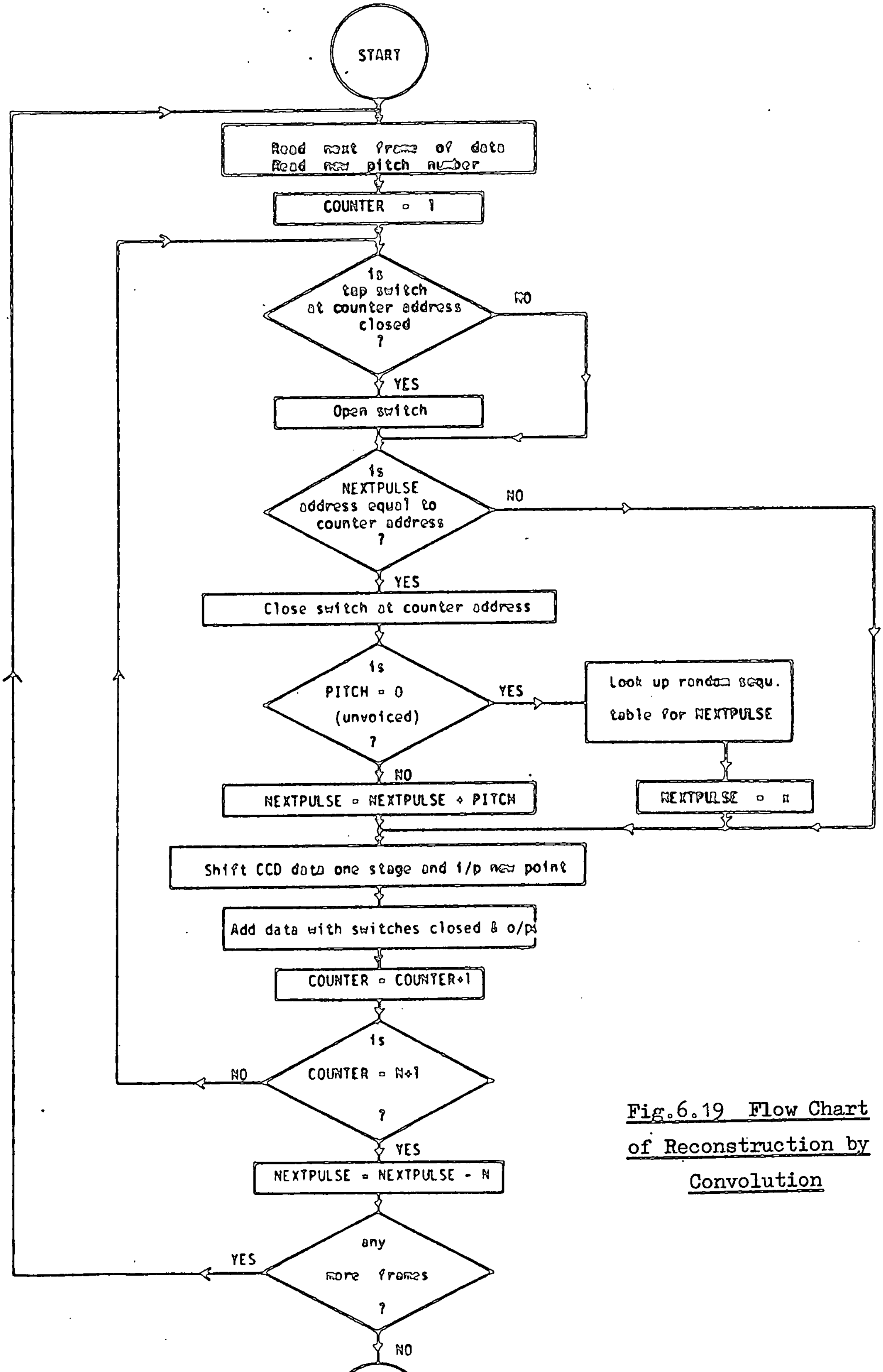

START
Read next frame of data
Read new pitch number
COUNTER = 1
is tap switch at counter address closed ?
NO
YES
Open switch
is NEXTPULSE address equal to counter address ?
NO
YES
Close switch at counter address
is PITCH = 0 (unvoiced) ?
YES
NO
Look up random sequ. table for NEXTPULSE
NEXTPULSE = NEXTPULSE + PITCH
NEXTPULSE = [illegible]
Shift CCD data one stage and i/p new point
Add data with switches closed & o/p
COUNTER = COUNTER+1
is COUNTER = N+1 ?
NO
YES
NEXTPULSE = NEXTPULSE - N
any more frames ?
YES
NO

Fig.6.19 Flow Chart of Reconstruction by Convolution

thereby allowing one complete frame of impulse data to be output from each selected tap. After a switch has been closed, the new "next pulse" address is calculated, which, in the case of voiced speech, is the old address added to the pitch input number. For unvoiced speech, the "next pulse" address is accessed from a random sequence look-up table. At the end of each frame, the "next pulse" address will point to a location outwith the range of the delay line and, in order to reset for the next impulse response frame, M tap locations are subtracted from this address (M is the number of CCD taps). This method ensures smooth reconstruction of the speech.

Fig.6.20 shows the reconstruction of voiced speech using the improved algorithm. Each impulse response has been smoothly connected at the pitch rate so that frame discontinuities are avoided. The change from voiced to unvoiced speech is demonstrated in Fig.6.21 and again there is no interference from the frame periodicity. The amplitude of the unvoiced segment of speech has been adjusted for optimum audio clarity.

6.3.4 Synthesiser Performance

The synthesiser's performance was assessed by comparing the output from a Marvox synthesiser with that from the

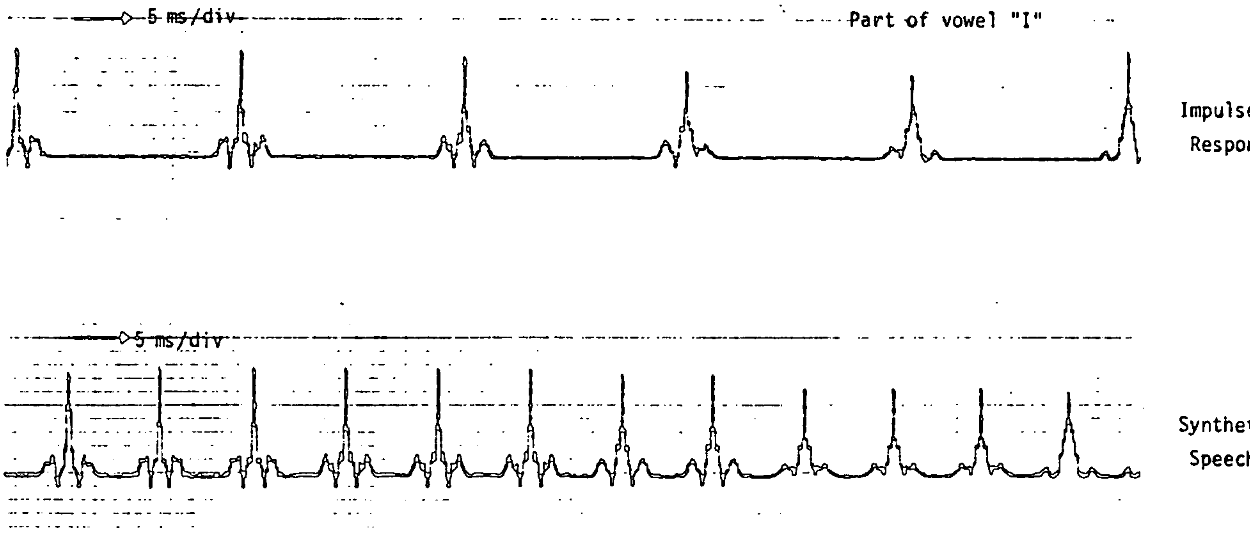


Fig.6.20 Interpolation of Pitch to Remove Frame Discontinuities

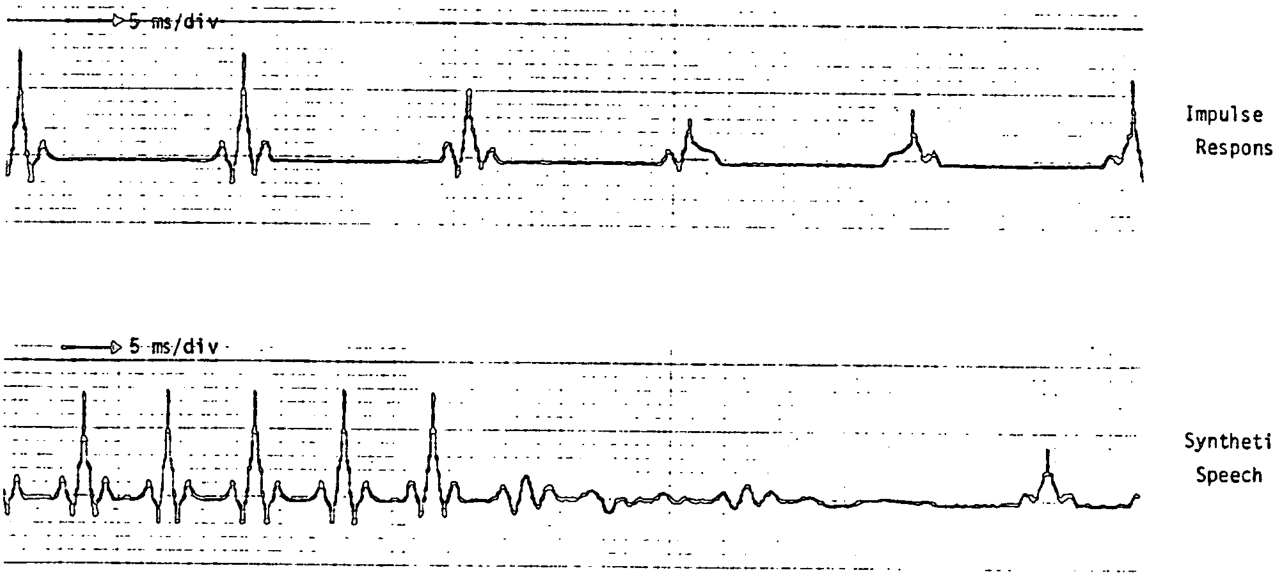


Fig.6.21 Change from Voiced to Unvoiced Speech

simulation when the common input was a digit stream generated by the analyser computer simulation. The spectrograms for the original speech and the Marvox synthesiser's output are shown in Fig.6.14 and Fig.6.15.

Fig.6.22 shows the corresponding narrowband and wideband spectrograms for the computer generated synthetic speech. Although these spectrograms are of rather poor quality, it can be seen from the narrowband analysis that the pitch contour is completely broken up and shows little resemblance to the original. Several explanations are possible. Firstly, there is no pitch smoothing low-pass filter in the computer simulation and, secondly, the pitch period is fixed for each 20ms frame, whereas in the Marvox synthesiser the pitch period is continuously changing. However, both of these effects will produce only small steps in the pitch contour and are not responsible for the rather large deviations in Fig.6.22. The main cause of distortion results from the reconstruction by convolution. When each 20ms impulse response is convolved with the periodic pulse train (which has a period of less than 20ms), consecutive impulse responses overlap, thereby creating discontinuities and generating spurious periodicities. The amount of distortion depends on the pitch period and on the width of the most significant part of the impulse response. The wider impulse responses, generated from spectral summation in alternate anti-phase, have a significant width of

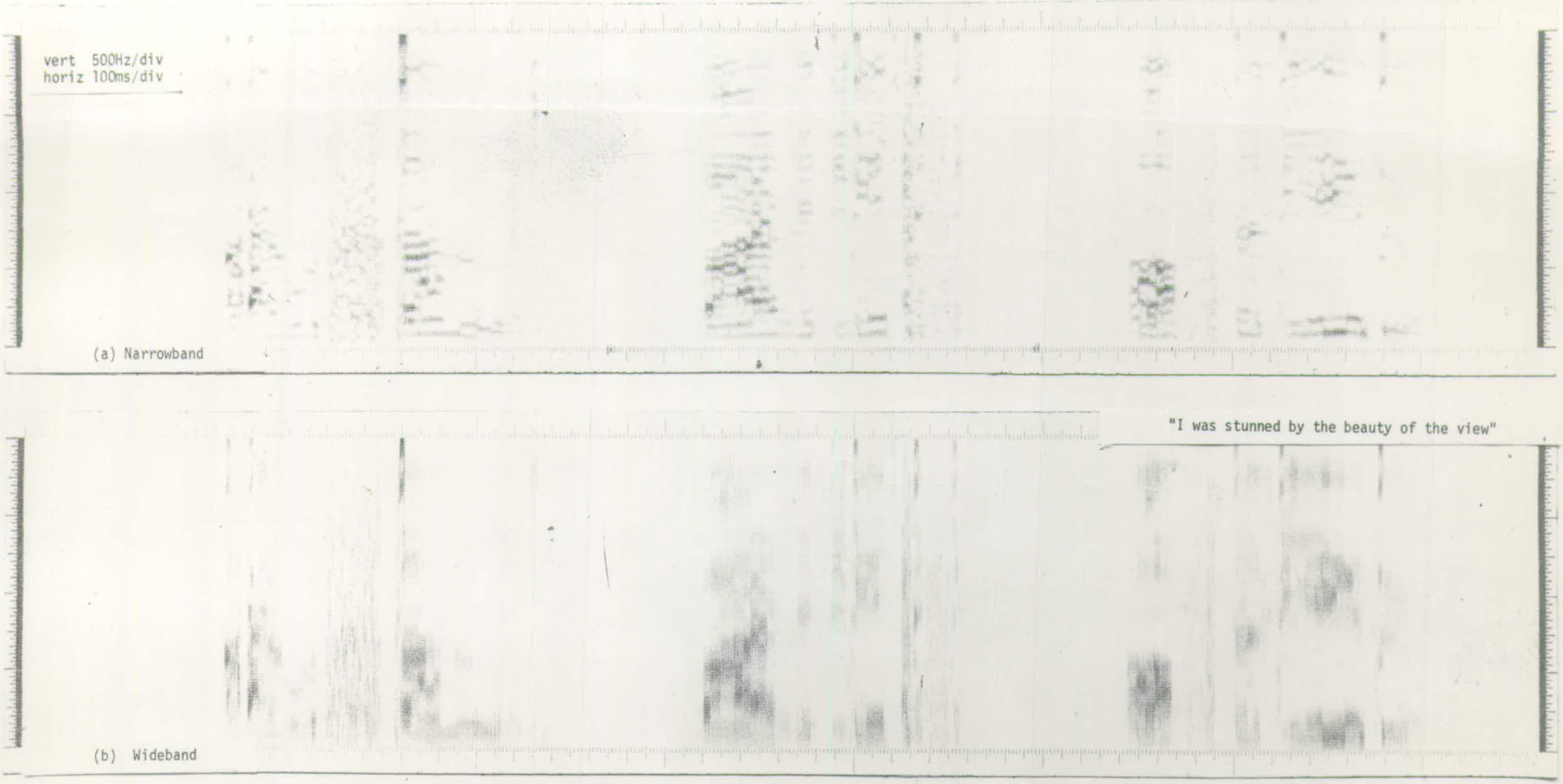


Fig.6.22 Spectrograms from the Analyser / Synthesiser Simulation

typically 10ms, whereas those from the in-phase summation last for 2ms. Since the wider impulse responses have been employed here, the distortion is severe. Reconstruction using the narrow impulse responses has been attempted, but, although the pitch is improved, the speech has a very mechanical sound.

The wideband analysis in Fig.6.22 shows that the formants are not as well defined as those obtained from the Marvox. Also, the coarse time quantisation (20ms) is apparent since continuous smoothing is not possible when the speech is reconstructed in frames. (The spectral data are smoothed from frame to frame by a low-pass filter in the Marvox synthesiser).

Listening tests have shown that although the speech is intelligible, the rough nature of the pitch detracts considerably from the synthetic speech quality. Unfortunately, the available time did not permit more detailed investigation of these problems.

6.3.5 Summary of Synthesiser Simulation Conclusions

The following conclusions were drawn from the synthesiser simulation:

1. the inverse DFT processor should have the same characteristics as the forward processor in the analyser (i.e. 256-point), but, in addition, should be capable of providing the real part of the output and not just the modulus (the sliding transform is therefore not applicable)

2. the frequency domain multiplication of the excitation source is not easy to implement and may cause frame discontinuities

3. time domain convolution of the excitation source data can remove frame discontinuities but introduces pitch discontinuities if the impulse responses are of longer duration than the pitch period

4. to date, the synthetic speech quality obtained from this simulation is not comparable with that from the Marvox synthesiser

5. further simulation is necessary to eliminate completely discontinuity problems in this synthesiser.

CHAPTER 7

THE OPTIMAL DESIGN OF A CCD CHANNEL VOCODER

This chapter is intended to merge the channel vocoder simulation results with the experience gained from CCD hardware. A possible implementation of a novel channel analyser is described in section 7.1 and the corresponding synthesiser is presented in the following section. Both are designed to interface with a conventional channel vocoder so that each may be operated independently. Section 7.3 compares the advantages and disadvantages of this particular implementation with the alternative CCD parallel filter bank vocoder.

7.1 ANALYSER CONFIGURATION

In order to maximise hardware efficiency, it is necessary to multiplex a single DFT processor. Because sliding processors cannot be multiplexed (without defeating their main advantages), the choice for central processor is therefore between the direct Chirp Z-transform (CZT) [6] and the Prime Transform (PT) [88]. Since in this particular application the input data do not have an imaginary part, the PT architecture (Fig.4.14) is chosen in preference to the CZT. This choice reduces the overall hardware configuration, by two CCD convolvers and six analogue multipliers. The PT input and output permutations can be incorporated into the addressing codes for exsisting input and output analogue storage without further hardware cost. In accordance with the computer simulation conclusions from chapter 6, a 257-point transform is selected, the nearest prime number to 256. (It is assumed here that the accuracy of Analogue Random Access Memory (ARAM) will be increased through development from 5% to 2%).

The complete analyser simulation is outlined in Fig.7.1. The input speech is amplified, emphasised at +6dB/octave from 1kHz and low-pass filtered to prevent aliasing (3.8kHz). This pre-processed speech is then sampled at 8kHz and passed to the first of three main processing stages.

The first stage, which is shown in more detail in Fig.7.2, is designed to time-compress and permute the input data. Time compression is necessary since the analyser computes two DFTs (each with 50% duty cycle) on 32ms segments of data (256 points at 8kHz) per 20ms frame. Two 256 stage ARAMs are connected in parallel to permit overlapping of data; the registers are loaded at a clock rate of 8kHz and are emptied every alternate frame at 51.2kHz (see timing diagram, Fig.7.5). These signals are therefore time compressed by a factor of 6.4. The order in which the data are read out is controlled by a ROM containing the permutation code (eqn.4.34) for a 257-point PT. Two extra sample and hold circuits are required to

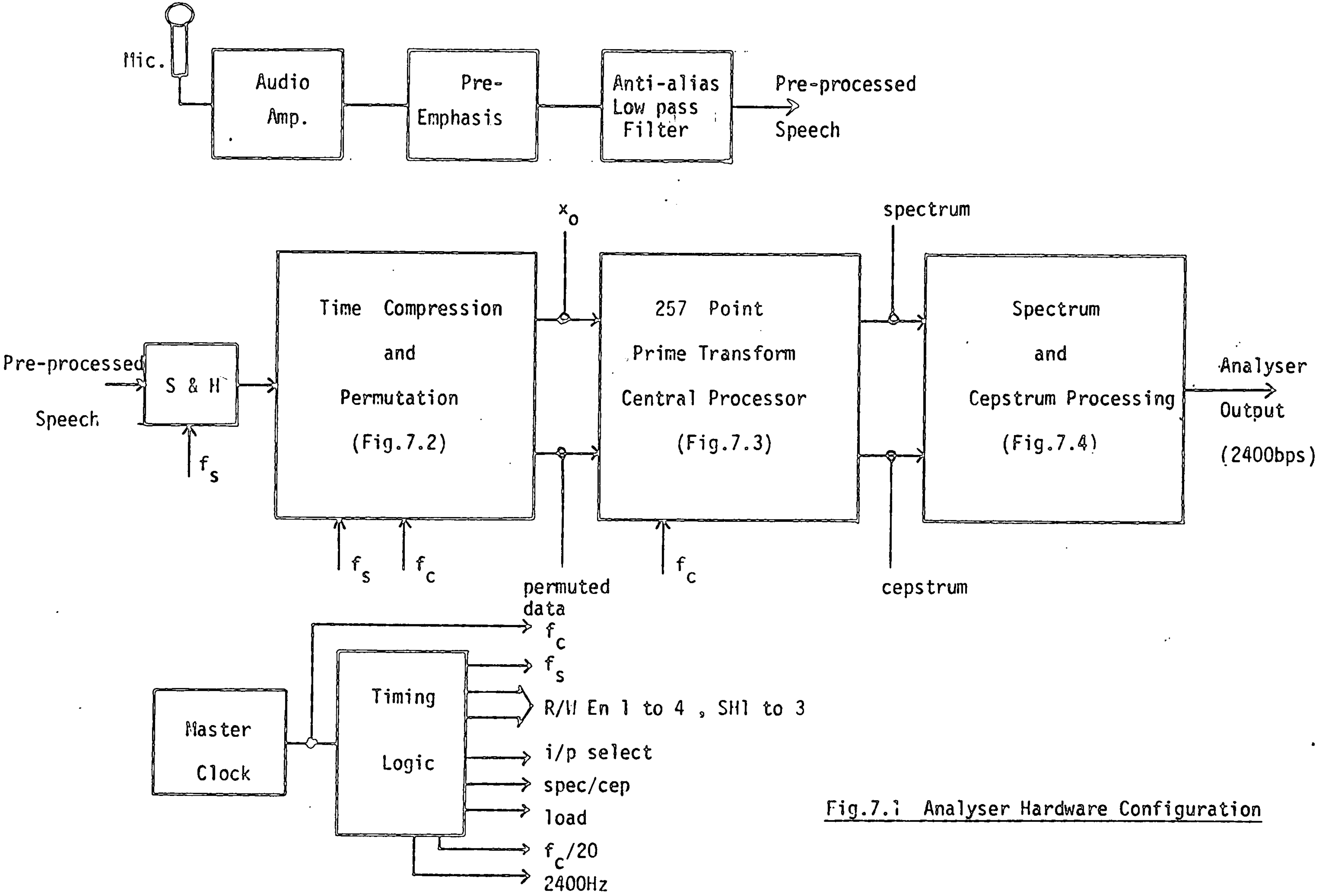


Fig.7.1 Analyser Hardware Configuration

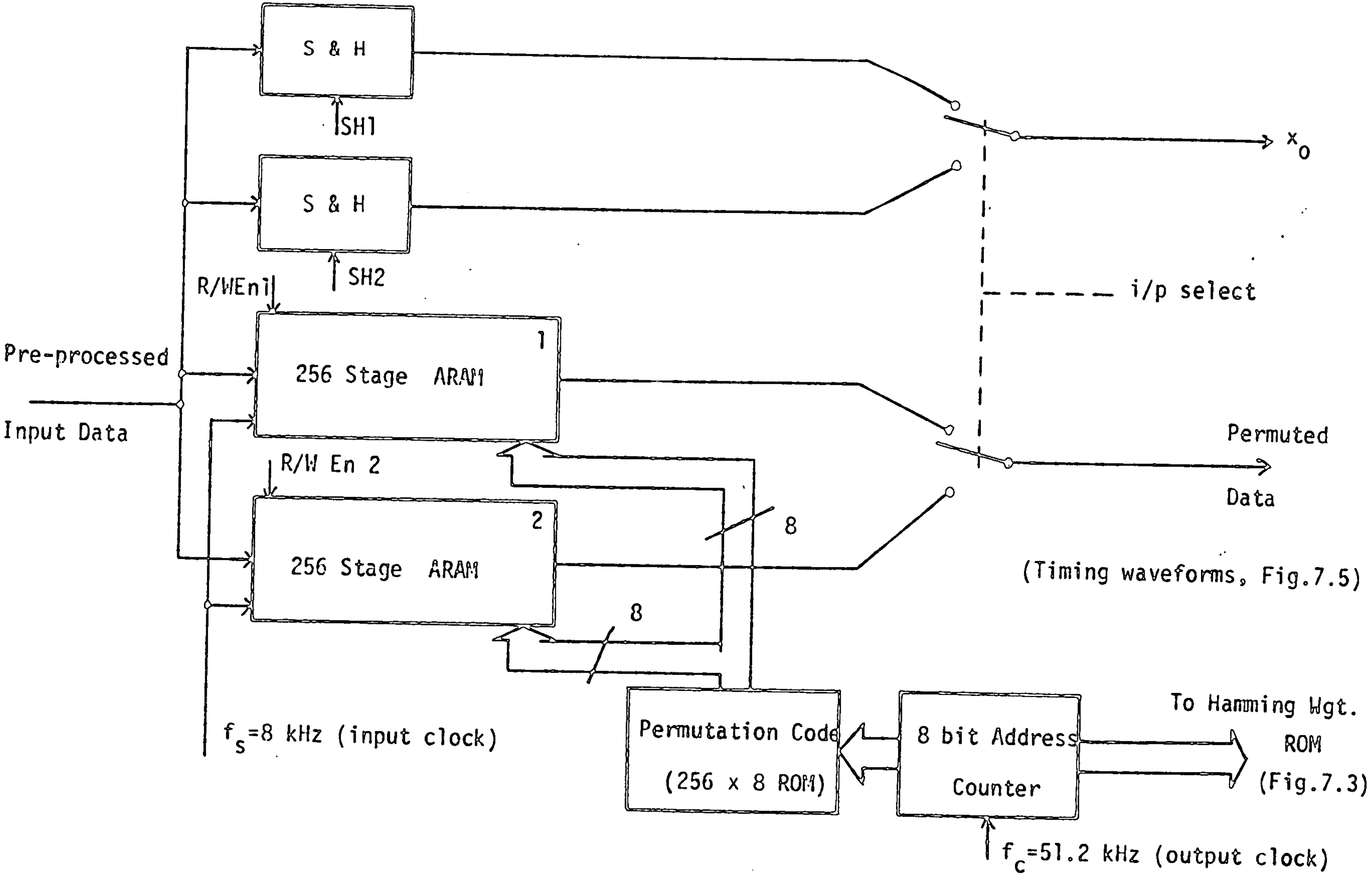


Fig.7.2 Time Compression and Data Permutation

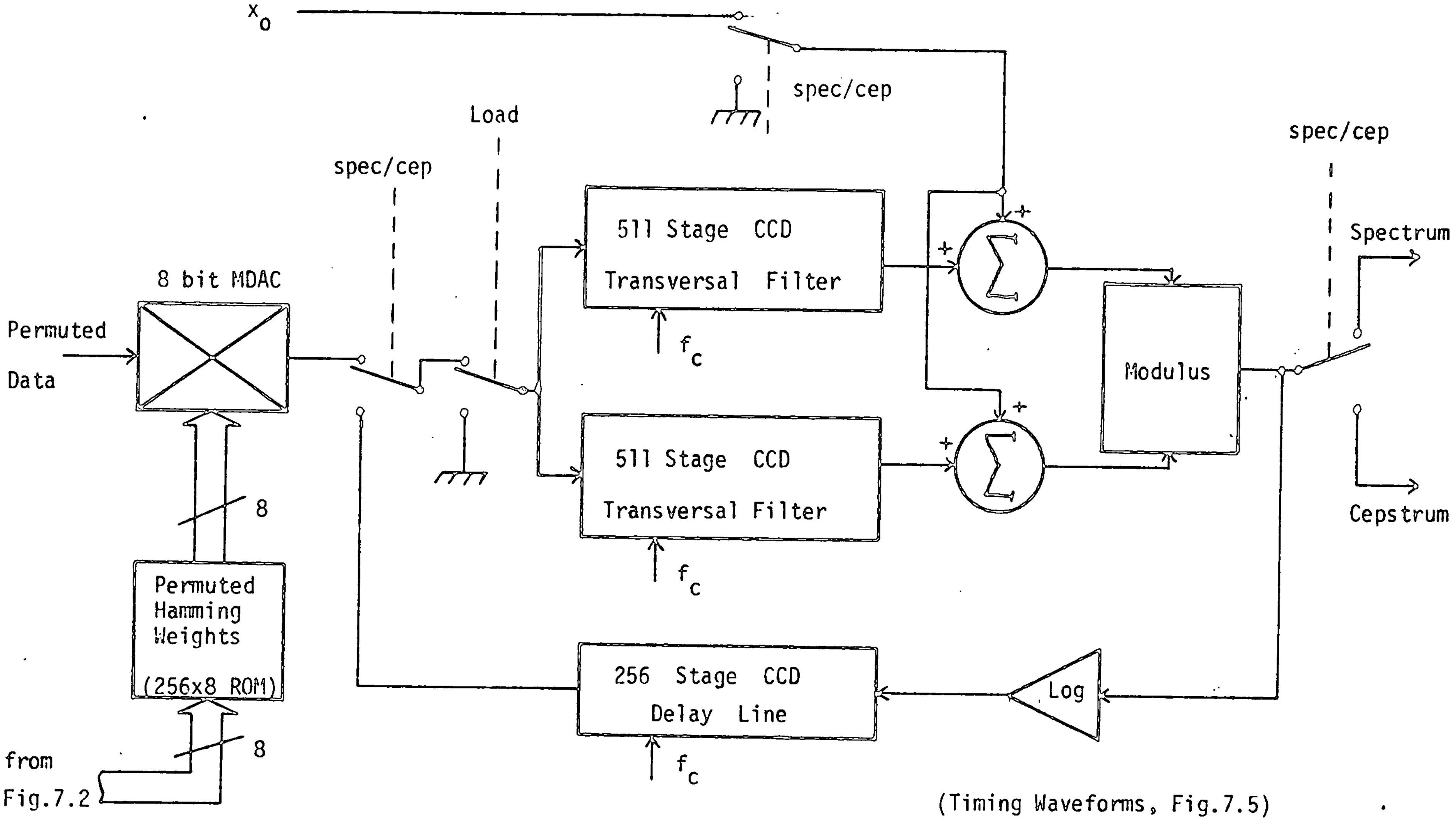


Fig.7.3 Prime Transform Central Processor

store the first data points in each frame (x_0) for the PT algorithm (see eqn.4.33).

The next processing section is the heart of the PT (Fig.7.3). The time-compressed data (256-points in 5ms) are multiplied by a Hamming window (permuted) to reduce frequency domain leakage and then loaded into two 511 stage (2N-3) CCD split gate transversal filters. Equations 4.37 and 4.38 give the transversal filter weighting coefficients. The transversal filter outputs are added to the spectral offset, x_0, before the two channels are combined by a modulus circuit to produce a permuted amplitude spectrum. An approximation to the modulus function (section 5.3.3) is sufficient here because of the coarse spectral quantisation which follows. The amplitude spectrum is passed to the output stages for coding and is also fed back to the correlator inputs via a logarithmic amplifier and a 256-stage delay line. Inverse permutation is not necessary at the output of the modulus stage since the feedback path also misses out the forward permutation. The nominal PT clock rate is 51.2kHz, allowing the calculation of two 257-point transforms per 20ms frame (actually 256 points because X_0, the dc coefficient is not computed). The processor's input/output sequence is as follows:

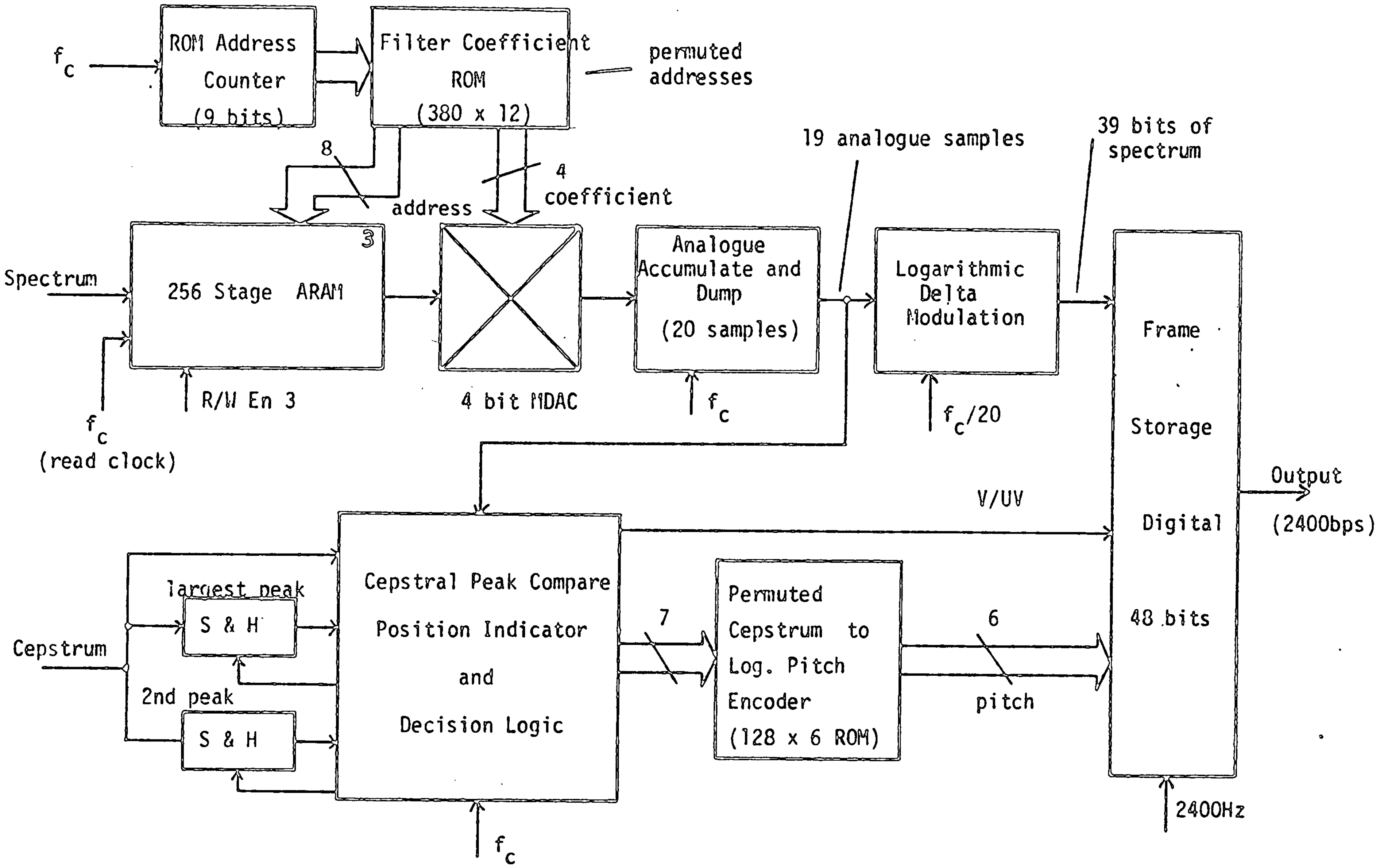

f_c
ROM Address Counter (9 bits)
Filter Coefficient ROM (380 x 12)
permuted addresses
8
address
4
coefficient
3
Spectrum
256 Stage ARAM
R/W En 3
f_c (read clock)
4 bit MDAC
Analogue Accumulate and Dump (20 samples)
f_c
19 analogue samples
Logarithmic Delta Modulation
f_c/20
39 bits of spectrum
Frame Storage Digital 48 bits
Output (2400bps)
2400Hz
V/UV
Cepstrum
largest peak
S & H
2nd peak
S & H
Cepstral Peak Compare Position Indicator and Decision Logic
f_c
7
Permuted Cepstrum to Log. Pitch Encoder (128 x 6 ROM)
6
pitch

Fig.7.4 Spectrum and Cepstrum Processing

TIME	PT INPUT	PT OUTPUT
0-5ms	Speech data	Undefined
5-10ms	Zero	Amp. spec.
10-15ms	Log. amp. spec.	Undefined
15-20ms	Zero	Cepstrum

The 256-point spectra and cepstra are each gated to their respective output processing stages (Fig.7.4). The spectra are stored in a 256-stage ARAM to facilitate the channel compression scheme detailed in section 6.2.3. A 380x12 bit ROM stores the permuted frequency bin addresses (8 bits) and also the 4-bit weighting coefficients (Table 6.1) which are multiplied into the frequency components by a 4-bit multiplying D to A converter. An analogue accumulate and dump circuit sums 20 weighted samples to give an approximation to each of the 19 Butterworth filter characteristics. The channel 2 and channel 19 outputs are fed to the pitch detector to assist in the V/UV decision. Finally, the channel outputs are coded by a logarithmic delta modulation scheme, normally implemented by a comparator, an up-down counter and a D to A converter in the feedback loop.

The pitch extraction process consists of detecting and storing the largest and second largest peaks in the cepstrum. These peaks are selected according to the flow diagram in Fig.6.11 and stored in two sample and hold circuits. Decision logic (Fig.6.13) then estimates the pitch information, which is converted to the appropriate

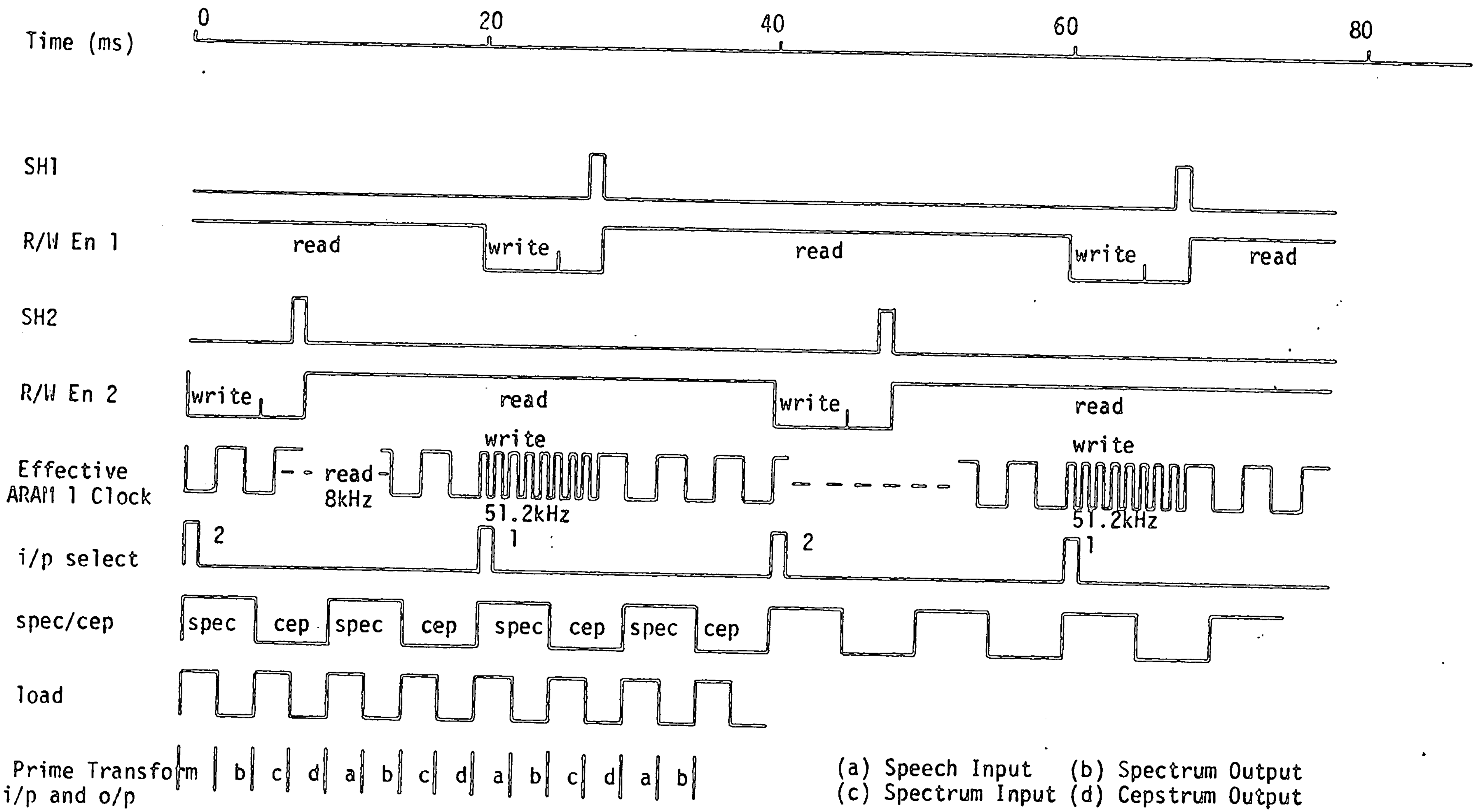


Fig.7.5 Analyser Timing Diagram

output coding (Table 6.2) by a decoder. The analyser frame is completed by emptying the frame storage register in serial at 2400bps.

7.2 SYNTHESISER CONFIGURATION

The input data are demultiplexed at 2.4kHz to provide spectral, pitch and sync. information (Fig.7.6). The sync. bit is used to control the master clock from which all timing functions are generated. After one frame of spectral data has been received and stored, the 39 data bits are anti-logarithmically D to A converted to form 19 channel amplitudes which are subsequently stored in ARAM. A 256-point permuted spectrum is then reformed by weighting and accumulating a set of contiguous filter bank characterisics (section 6.3.1). Three samples are accumulated for each spectral point since, at most, three filter characteristics overlap.

Only one 257-point inverse PT is required for each 20ms frame in the synthesiser and, because the spectral data are real and even, the inverse PT reduces to a single channel. The transform clock rate is therefore chosen to be 25.6kHz so that the spectral data are read in during the first 10ms of each frame and the time-compressed impulse response data are output into ARAM during the second 10ms period. At the beginning of the next frame, one impulse response is

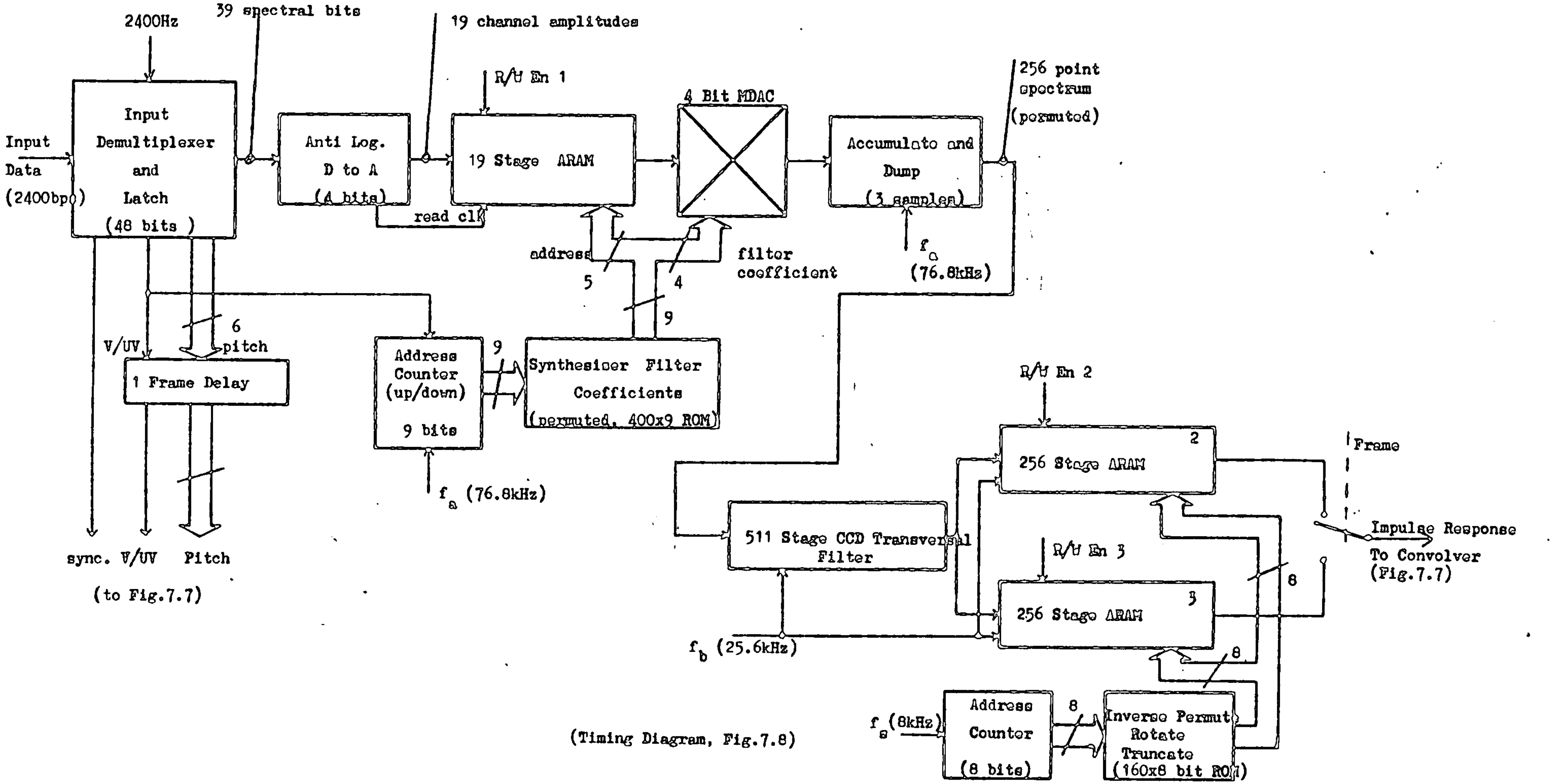

2400Hz
39 spectral bits
19 channel amplitudes
R/W En 1
4 Bit MDAC
256 point spectrum (permuted)
Input Data (2400bps)
Input Demultiplexer and Latch (48 bits)
Anti Log. D to A (4 bits)
19 Stage ARAM
Accumulate and Dump (3 samples)
read clk
address
5
4
filter coefficient
f_a (76.8kHz)
9
6
V/UV
pitch
1 Frame Delay
Address Counter (up/down) 9 bits
9
Synthesiser Filter Coefficients (permuted, 400x9 ROM)
f_a (76.8kHz)
sync. V/UV Pitch
(to Fig.7.7)
R/W En 2
256 Stage ARAM
2
Frame
511 Stage CCD Transversal Filter
R/W En 3
Impulse Response To Convolver (Fig.7.7)
256 Stage ARAM
3
8
f_b (25.6kHz)
8
f_s (8kHz)
Address Counter (8 bits)
8
Inverse Permut Rotate Truncate (160x8 bit ROM)
(Timing Diagram, Fig.7.8)

Fig.7.6 Synthesiser - Impulse Response Generation

transferred to the speech reconstruction algorithm (see timing diagram in Fig.7.8). The order and rate of this data transfer is controlled by an address ROM clocked at 8kHz which combines four functions: (1) inverse permutation, (2) impulse response rotation, (3) truncation to 160-points and (4) time expansion to the original speech sampling rate.

The main component in the speech reconstruction algorithm (Fig.7.7) is a 160-stage CCD transversal filter with binary weights. The operation of this stage is well described by the flow diagram in Fig.6.19 and its accompanying text. Note that the gain of the transversal filter is reduced for unvoiced reconstruction to ensure proper balance in the synthetic speech. A sample and hold circuit is included at the output of the transversal filter to remove CCD clock breakthrough. The synthetic speech is output after low-pass filtering at 3.8kHz and frequency de-emphasis.

A half-duplex vocoder may be constructed by multiplexing the major processing blocks in the analyser with those in the synthesiser. Alternatively, a full duplex vocoder may be achieved by increasing the analyser throughput rate and combining the central processors.

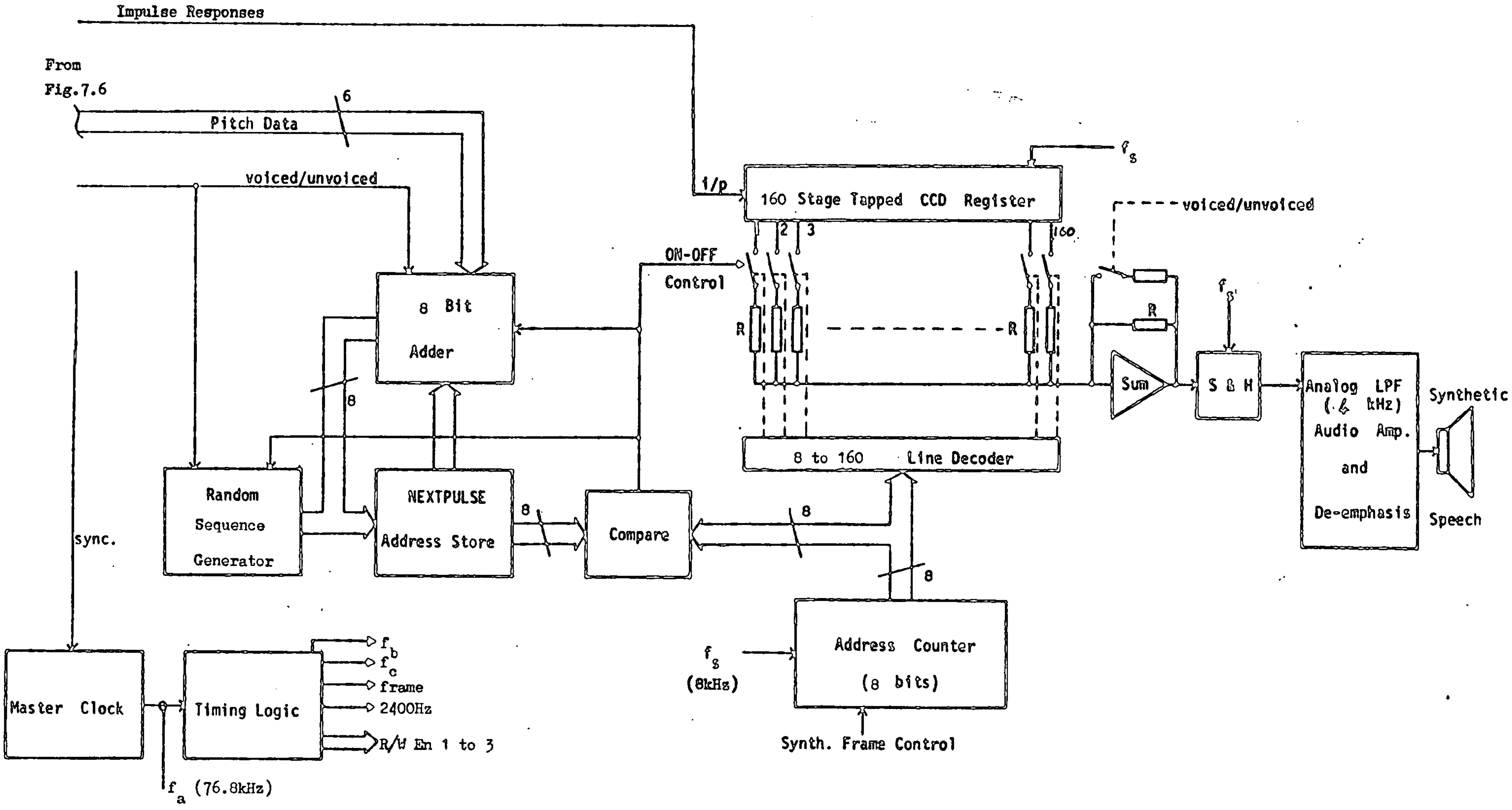


Fig.7.7 Synthetic Speech Reconstruction

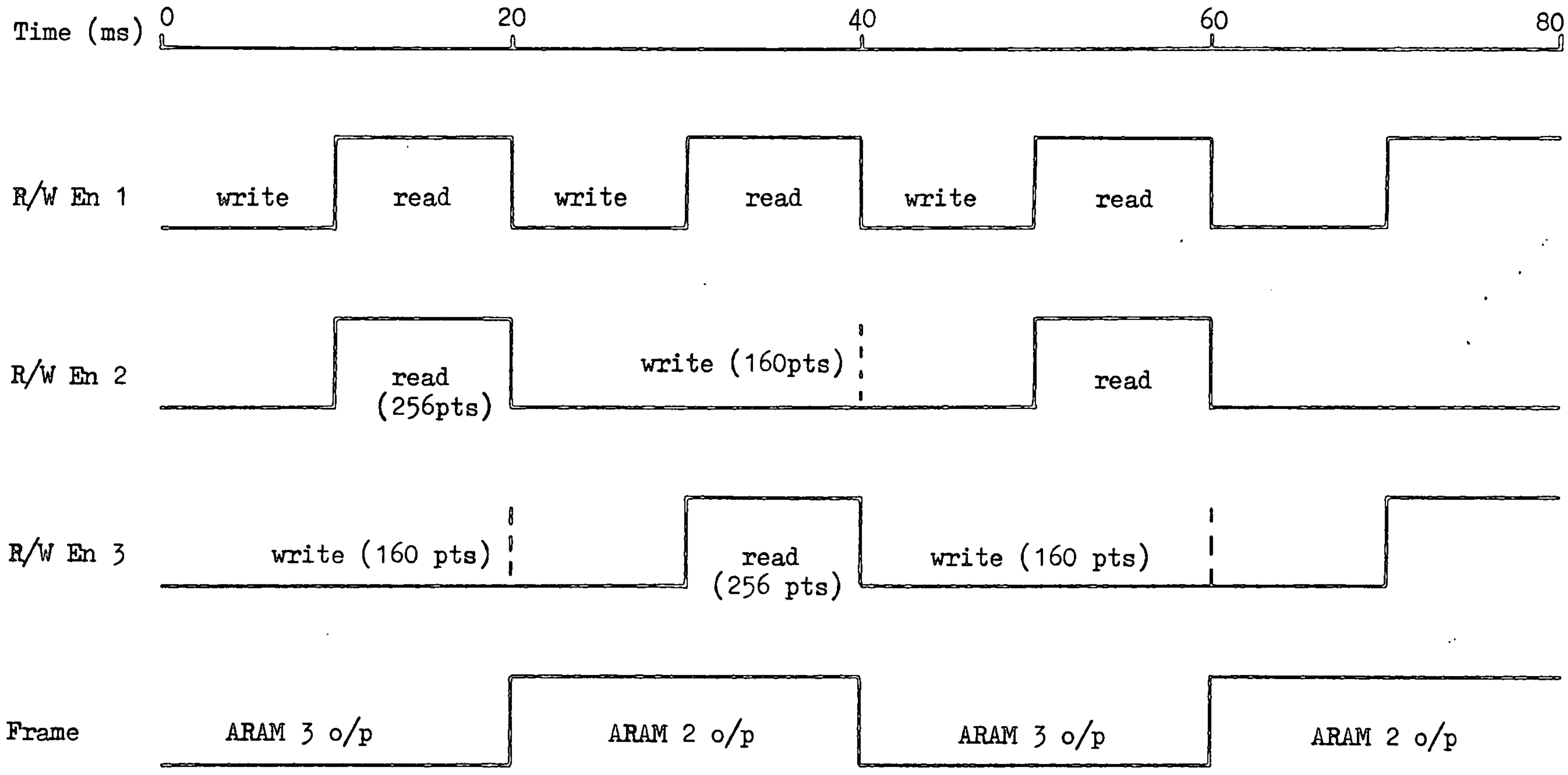


Fig.7.8 Synthesiser Timing

7.3 COMPARISON WITH A CCD PARALLEL FILTER BANK VOCODER

In 1978, Hewes et.al. [30] from Texas Instruments published details of a channel vocoder implemented using CCD and switched capacitor technology. This vocoder, almost an exact copy of the Marvox algorithm, is implemented with two custom designed CCD/NMOS integrated circuits and 5 microcomputers (TMS9940). The CCD analyser chip contains 19 parallel CCD band-pass filters, 19 full-wave rectifiers, 19 switched capacitor low-pass filters, a multiplexer and a logarithmic A to D converter. Three of the microcomputers are utilised in a pitch detector while the other two handle data coding. The synthesiser chip houses the full Marvox synthesiser.

Table 7.1 summarises and compares the component requirements for the analyser with that for the cepstral vocoder. In terms of analogue storage, each of the analyser implementations is comparable; however, ARAM consumes more chip space than does CCD. On the other hand, 19 CCD transversal filters need 19 summing amplifiers consuming both power and chip space. The other analogue circuitry in each analyser may be compared in terms of the total number of amplifiers. The filter bank analyser uses almost double the amplifiers employed by the cepstral channel analyser. In addition to the analogue components, the cepstral

Comparison with a CCD Parallel Filter Bank Vocoder

(a) CCD Cepstral Vocoder

	Analyser Chip	Synthesiser Chip
Analogue		
	1278 stages of CCD + 3 amps.	511 stages of CCD + 1 amp.
	768 stages of ARAM	160 stages tapped CCD + 1 amp.
	5 Sample and Holds	531 stages of ARAM
	3 Summers	1 Sample and Hold
	3 Comparators	1 Accumalate and Dump
	1 Log. amp.	
	2 Rectifiers	
	1 Accumulate and Dump	
Digital		
	1170 bytes ROM	610 bytes ROM
	6 bytes RAM	8 bytes RAM
	1 - 8 bit MDAC	1 - 4 bit MDAC
	1 - 4 bit MDAC	1 - 4 bit anti-log DAC
	1 - 3 bit DAC	1 - 8 to 160 line decoder
	Misc. Logic	Misc. Logic
External Componentry for Complete System		
	Amplifier	Amplifier
	Low-pass Filter	Low-pass Filter
	Pre-emphasis Filter	De-emphasis Filter

(b) CCD Filter Bank Vocoder

	Analyser Chip	Synthesiser Chip
Analogue		
	1900 stages of CCD + 19 Amps.	1 anti. log. DAC (5 bits)
	19 Rectifiers	1 Channel Demultiplexer
	19 Switched Cap. LPFs (38 amps.)	19 Sample and Holds
	1 - 19 channel analog multiplexer	19 Switched.Cap. LPFs (38 amps.)
	1 - 5 bit log. A to D converter	1 Summer
Digital		
	Misc. Logic	Misc. Logic
External Componentry for Complete System		
	Amplifier	Amplifier
	Low-pass Filter	Low-pass Filter
	Pre-emphasis Filter	De-emphasis Filter
	3 Microcomputers (pitch detector)	1 Microcomputer (frame format)
	1 Microcomputer (frame format)	

implementation requires a considerable amount of digital circuitry. By far the largest item is the 1170 bytes of ROM, of which almost half is used to represent the 19 Butterworth filter characteristics. It seems likely that further development would permit a more efficient approximation to these filter characteristics. For example, if only one characteristic was stored for each filter with equal bandwidth, the filter coefficient storage could be reduced by approximately 80%. It should be noted that the extra digital hardware in the cepstral analyser should be compared with the four microcomputers employed by the filter bank analyser since pitch detection and frame coding are included in the components list.

Comparing the synthesiser component counts, it can be seen that the major item in the filter bank implementation is the number of amplifiers whereas, because processing is serial in the other synthesiser, the major cost is analogue and digital storage. Although the total chip areas may prove similar, the switched capacitor filter bank will always have an advantage, since the performance is limited only by the operational amplifiers.

In summary, it is considered that the CCD DFT based analyser offers an equivalent performance to that of the CCD filter bank approach [30] but will provide a more efficient hardware implementation. The new synthesiser, however, will operate with decreased fidelity (without further development) due to pitch discontinuities and is unlikely to achieve vastly superior engineering benefits. The optimum analogue channel vocoder implementation points to a combination of the cepstral analyser and the filter bank synthesiser. It appears possible that this system will enable the construction of a two chip vocoder.

CHAPTER 8

CONCLUSIONS

The design of CCD Fourier transform processors and their application in low bit rate speech communication systems have been investigated in this thesis. In particular, a novel implementation of the channel vocoder nas been developed.

In the author's experience, many people have a poor understanding of the DFT and its derivatives; chapter 4 has been written bearing this in mind and attempts to clarify some of the main areas of confusion. The mathematical concepts are translated into realisable hardware structures and attention is drawn to cases where suitable restrictions permit hardware reduction. Practical considerations are emphasised throughout and performance limitations are compared. It is shown that in real-time applications demanding high accuracy coupled with high resolution, there is at present no alternative to the digital FFT. However, in cases where reduced accuracy can be tolerated, analogue CCD Fourier transform processors such as the CZT and the PT offer up to 512 point transforms on a single IC. Thus the module size and power consumption are reduced considerably when compared to current digital FFTs. This is achieved at the expense of reduced accuracy (e.g. 1%) which gives a typical processor error or sidelobe level of approximately -40dB. Considering noise only, the output exhibits 50-60dB signal to noise ratio. A comparison between the CZT and the

PT has shown that in terms of accuracy, there is little difference between the CZT and the PT, but with restricted input conditions (e.g. a typical speech waveform which is real), the PT configuration reduces the hardware by a factor of two.

Chapter 5 described the design and construction of a CCD CZT processor. The computer simulation employed here permitted the selection of component tolerances for any given transform accuracy. The spectrum analyser, which was constructed in CCD hardware computed either a 32-point direct CZT or a 64-point sliding transform, with or without Hamming weighting. It met all of the initial design specifications except the output linearity, which was limited to 30dB by the output transconductance multipliers in the modulus circuit. The alternative linear approximation circuit (+1/2dB ripple) increased the dynamic range to 50-60dB but limited the processor speed to 100kHz. From this discussion, it is clear that the practical design of the processor output stage requires further attention if 40dB linear dynamic range is to be achieved. The author considers that chapters 4 and 5 together form a comprehensive guide to the design of CCD Fourier transform processors.

On-line computer simulations, detailed in chapter 6, permitted the design of a channel vocoder based on DFT techniques. This clearly showed that the sliding transform was not suitable for the short-time spectrum analysis of speech. The simulation further showed that our DFT based cepstral pitch detector technique provided superior results on all speech except for very high pitched female voices. This was verified by comparative listening tests with speech synthesised from a discrete filter bank. The author therefore believes that CCD DFT processors offer significant practical advantages when used for pitch detection. The simulated channel synthesiser was not as successful as the analyser due to discontinuities arising from the overlapping of impulse responses. Further simulation work is therefore necessary to improve the synthetic speech quality.

Finally, the simulation results and practical experience gained from CCD DFT processors have been combined in chapter 7 to show how the channel vocoder might best be implemented using sample-data analogue signal processing techniques. As discussed in chapter 7, a PT configuration was selected for the central processing unit. From an engineering point of view, the cepstral analyser compares favourably with an integrated CCD filter bank in terms of chip size, power consumption and hence cost. However, since the DFT based analyser performs both the required spectrum analysis and pitch detection without the need for external microcomputers, it is now believed to be the more attractive implementation. In contrast, the synthesiser does not promise any particular performance advantage and is more expensive in chip area than the parallel filter bank. It is therefore considered that the optimum CCD channel vocoder configuration will be based on a combination of the cepstral analyser and the filter bank synthesiser. Using this approach, a two chip vocoder implementation is feasible. However, before chip design can commence, a discrete hardware model of the vocoder should be built to permit the optimisation of the analyser filter bank coefficients thereby reducing storage requirements. This will also allow thorough intelligibility testing to be undertaken with a large sample of male and female speakers.

In summary, this thesis has shown conclusively that, at the present time, the analogue CCD has considerable importance in speech processing systems. However this conclusion must be treated with caution since the rate of progress in digital signal processing (e.g. high speed bit slice microprocessors) is such that within a few years most speech processing may be performed digitally.

REFERENCES

1. CUCCIA, C.L.: "Bandwith Conservation is Essential", Microwave Systems News, Oct.1978, pp.67-72.

2. GOLD, B. and RADER, C.M.: "The Channel Vocoder", IEEE Trans. Audio and Electroacoustics, Dec.1967, Vol.AU-15, No.4, pp.148-161.

3. ATAL, B.S. and HANAUER, S.L.: "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave", J. Acoust. Soc. Amer., 1971, Vol.50, pp.637-655.

4. HOWES, M.J.. and MORGAN, D.V.: "Charge Coupled Devices and Systems", John Wiley and Sons, 1979.

5. TURIN, G.L.: "An Introduction to Matched Filters", IRE Trans.Inform.Theory, June 1960, Vol.IT-6, pp.311-329.

6. EVERSOLE, W.L. et.al.: "Spectral Analysis using the CCD Chirp Z-Transform", AGARD Conf.Proc. No.230, Oct.1977, Paper No.5.3.

7. WHITE, M.H. et.al.: "CCD Analog Adaptive Signal Processing", Proc. CCD Applications Conf., San Diego, 1978, pp.3A1-3A14.

8. DENYER, P.B. et.al.: "A Programmable CCD Transversal Filter: Design and Application", Proc. CCD Applications Conf., San Diego, 1978, pp.3B11-3B21.

9. WILKINSON, R.M.: "Delta Modulation Techniques for Analogue to Digital Conversion of Speech Signals", Signals Research and Development Establishment Report no.69022, Apr.1969.

10. KING, R.A. and GOSLING, W.: "Time Encoded Speech (TES)", IEE Int. Specialist Seminar on Case Studies In Advanced Signal Processing, Conf. proc., Peebles, Sept.1979, to be published.

11. FLANAGAN, J.L: "Speech Analysis, Synthesis and Perception", Springer-Verlag, New York 1972.

12. HOLMES, J.N.: "Speech Synthesis", Mills and Boon Monograph EE/7, 1972.

13. GILL, J.S.: "Improvements in or relating to Larynx Excitation Period Detectors", U.K. Patent Applic. No. 10525/65, May 1965.

15. GOLD, B. and RABINER, L.R.: "Parallel Processing Techniques for Estimating Pitch Periods of Speech in the Time Domain", J. Acoust. Soc. Amer., Aug.1969, Vol.46, pp.442-448.

16. SONDHI, M.N.: "New Methods of Pitch Extraction", IEEE Trans. Audio Electroacoust., June 1968, Vol.AU-16, pp.262-266.

17. DUBNOWSKI, J.J., SCHAFER, R.W. and RABINER, L.R.: "Real-time Digital Hardware Pitch Detector", IEEE Trans. Acoust., Speech and Sig.Proc., Feb.1976, Vol.ASSP-24, pp.2-8.

18. ROSS, M.J. et.al.: "Average Magnitude Difference Function Pitch Extractor", IEEE Trans.Acoust., Speech and Sig. Proc., Oct.1974, Vol.ASSP-22, pp.353-362.

19. MAKSYM, J.N.: "Real-time Pitch Extraction by Adaptive Prediction of the Speech Waveform", IEEE Trans. Audio and Electroacoust., June 1973, Vol.AU-21, No.3, pp.149-154.

20. MOORER, J.A.: "The Optimum Comb Method of Pitch Period Analysis of Continuous Digitized Speech", IEEE Trans.Acoust., Speech and Sig. Proc., Oct.1974, Vol.ASSP-22, No.5, pp.

21. NOLL, A.M.: "Cepstrum Pitch Determination", J. Acoust. Soc. Amer., Feb.1967, Vol.41, No.2, pp.293-309.

22. RABINER, L.R. et.al.: "A Comparitive Performance Study of Several Pitch Detection Algorithms", IEEE Trans.Acoust., Speech and Sig. Proc., Oct.1976, Vol.ASSP-24, No.5, pp.399-418.

23. DUDLEY, H.W.: "The Vocoder", Bell Labs. Rec., 1939, Vol.17, pp.122-126.

24. HOLMES, J.N.: "A Variable Frame Rate Coding Scheme for Speech Analysis-Synthesis Systems", Electronic Letters, Apr.1974, Vol.10, No.7, pp.101-102.

25. HOLMES, J.N., private communication.

26. RADER, C.M.: "Spectra of Vocoder Channel Signal", J. Acoust. Soc. Amer., 1963, Vol.35, p805.

27. KELLY, L.C.: "Speech and Vocoders", The Radio and Electronic Engineer, Aug.1970, Vol.40, No.2, pp.73-82.

28. BIALLY,T. and ANDERSON,W.M.:"A Digital Channel Vocoder", IEEE Trans.Comm.Tech., Aug.1970, Vol.COM-18, No.4, pp.435-442.

29. KINGSBURY,N.G. and KELLY,L.C.:"A Digital Filter Bank for Real-time Speech Analysis and Synthesis using Logarithmically Quantised Signals", Proc. Digital Processing of Signals in Communications, IERE Conf. Proc. No.37, pp.81-96.

30. HEWES,C.R. et.al.:"A CCD/NMOS Channel Vocoder", Proc. CCD Applications Conf., San Diego, 1978, pp.3A17-3A24.

31. MAKHOUL,J.:"Linear Prediction: A Tutorial Review", Proc. IEEE, Apr.1975, Vol.63, pp.561-580.

32. WIGGINS,R. and BRANTINGHAM,L.:"Three Chip System Synthesizes Human Speech", Electronics, Aug.1978, pp.109-116.

33. VISWANATHAN,R. and MAKHOUL,J.:"Quantisation Properties of Transmission Parameters in Linear Predictive Systems", IEEE Trans.Acoust., Speech and Sig. Proc., Vol.ASSP-23, No.3, June 1975, pp.309-321.

34. OPPENHEIM,A.V.:"Speech Analysis-Synthesis System Based on Homomorphic Filtering", J. Acoust. Soc. Amer., 1969, Vol.45, No.2, pp.458-465.

35. OPPENHEIM,A.V. and SCHAFER,R.W.:"Homomorphic Analysis of Speech", IEEE Trans. Audio and Electroacoustics, June 1968, Vol.AU-16, No.2, pp.221-226.

36. WEINSTEIN,C.J. and OPPENHEIM,A.V.:" Predictive Coding in a Homomorphic Vocoder", IEEE Trans. Audio and Electroacoustics, Sept.1971, Vol.AU-19, No.3, pp.243-248.

37. IMAI,S.:"Low Bit Rate Cepstral Vocoder Using the Log Magnitude Approximation Filter", IEEE Conf.Proc. CH1285-6/78/0000-0441$00.75@1978, 1978.

38. SCHAFER,R.W. and RABINER,L.R.:"System for Automatic Analysis of Voiced Speech", J. Acoust. Soc. Amer., 1970, Vol.47, pp.634-648.

39. MARKEL,J.D.:"Application of a Digital Inverse Filter for Automatic Formant and Fo Analysis", IEEE Trans. Audio and Electroacoustics, June 1973, Vol.AU-21, No.3, pp.154-160.

40. McCANDLESS,S.S.:"An Algorithm for Automatic Formant Extraction Using Linear Prediction Spectra", IEEE Trans.Acoust., Speech and Sig. Proc., Apr.1974, Vol.ASSP-22, No.2, pp.135-141.

41. BELL,G.G. et.al.:" Reduction of Speech Spectra By Analysis-by-Synthesis Techniques", J. Acoust. Soc. Amer., 1961, Vol.33, pp.1725-1736.

42. BOYLE,W.S. and SMITH,G.E.: "Charge-Coupled Semiconductor Devices", Bell Syst. Tech. Journ., 1970, 49, pp.587-593.

43. SEQUIN,C.H. and TOMPSETT,M.F.: "Charge Transfer Devices", Academic Press, Inc., 1975.

44. TOMPSETT,M.F.: "Charge Transfer Devices", J. Vac. Sci. Technol., July-Aug 1972, 9, No.4, pp.1166-1181.

45. SZE,S.M.: "Physics of Semiconductor Devices", John Wiley and Sons, 1969.

46. BEYNON,J.D.E.: "The Basic Principles of Charge-Coupled Devices", Microelectronics, 1975, 7, No.2, pp.7-13.

47. TOMPSETT,M.F.,AMELIO,G.F. and SMITH,G.E.: "Charge Coupled 8-Bit Shift Register", Appl. Phys. Lett., 1970, 17, pp.111-115.

48. SEQUIN,C.H. and MOHSEN,A.M.: "Linearity of Electrical Charge Injection into Charge-Coupled Devices", IEEE J. of Solid State Circuits, Apr.1975, SC-10, No.2, pp.81-92.

49. TOMPSETT,M.F. and ZIMANY,E.J.,Jr.: "Use of Charge-Coupled Devices for Delaying Analog Signals", IEEE J. Solid State Circuits, Apr.1973, SC-8, pp.151-157.

50. TOMPSETT,M.F.: "Surface Potential Equilibration Method of Setting Charge in Charge Coupled Devices", IEEE Trans. Electron Devices, June 1972, ED-22, No.6, pp.305-309.

51. BAERTSCH,R.D. et. al.: "The Design and Operation of Practical Charge Transfer Transversal Filters", IEEE Trans. Electron Devices, Feb.1976, ED-23, No.2, pp.133-142.

52. MACLENNAN,D.J. and MAVOR,J.: "Novel Technique for the Linearisation of Charge Coupled Devices", Electronics Letters, May 1975, 11, No.10,

pp222-223.

53. ARTHUR,J.W., private communication.

54. BUSS,D.D. et. al.: "Transversal Filtering Using Charge Transfer Devices", IEEE J. of Solid State Circuits, Apr.1973, SC-8, No.2, pp.138-146.

55. MACLENNAN,D.J. et. al.: "Techniques for Realising Transversal Filters using Charge-Coupled Devices", Proc. IEE, June 1975, 122, No.6, pp.615-619.

56. DENYER,P.B. and MAVOR,J.: "Design of CCD Delay Lines with Floating Gate Taps", Solid State and Electron Devices, July 1977, 1, No.4, pp.121-129.

57. DENYER,P.B. and MAVOR,J.: "Design and Development of CCD Programmable Transversal Filters", Electronic Circuits and Systems, Jan.1978, 2, No.1, pp.1-8.

58. TOMPSETT,M.F.: "The Quantitative Effects of Interface States on the Performance of Charge-Coupled Devices", IEEE Trans. Electron Devices, 1973, ED-20, pp.44-45.

59. CARNES,J.E.,KOSONOCKY,W.F. and RAMBERG,E.G.: "Free Charge Transport in Charge-Coupled Devices", IEEE Trans. Electron Devices, 1972, ED-19, pp.798-808.

60. VANSTONE,G.F, ROBERTS,J.B.G. and LONG.A.E.:"The Measurement of the Charge Residual for CCD Transfer Using Impulse and Frequency Responses", Solid State Electronics, 1974, 17, pp.889-895.

61. DUTTA ROY,S.C. and DAS,V.G.: "On Exact Compensation of Transfer Inefficiency in a Charge Transfer Delay Line", Electronics Letters, Feb.1978, 14, No.4, pp.115-116.

62. TOZER,R.C. and HOBSON,G.S.: "Reduction of High-Level Nonlinear Smearing in CCDs", Electronic Letters, July 1976, 12, No.14, pp.355-356.

63. MAVOR,J.,DAVIE,M.C. and DENYER,P.B.: "Techniques for Increasing the Effective Charge Transfer Efficiency of Tapped CCD Registers", Electronics Letters, Jan.1977, 13, No.1, pp.31-33.

64. MOHSEN,A.M.,TOMPSETT,M.F. and SEQUIN,C.H.: "Noise Measurements in Charge-Coupled Devices", IEEE Trans. Electron Devices, May 1975, ED-22, No.5, pp.209-218.

65. WESTE,N. and MAVOR,J.: "MOST Amplifiers for Performing Peripheral Integrated Circuit Functions", IEE J. Electron. Circuits and Syst., 1977, 1, pp.165-172.

66. CAVES,J.T. et. al.: "Sampled Analog Filtering Using Switched Capacitors as Resistor Equivalents", IEEE J. Solid State Circuits, Dec.1977, SC-12, No.6, pp.592-599.

67. RABINER,L.R. and GOLD,B.: "Theory and Application of Digital Signal Processing", Prentice-Hall Inc., 1975.

68. DENYER,P.B.,MAVOR,J. and ARTHUR,J.W.,:"Miniature Programmable Transversal Filter Using CCD/MOS Technology", Proc. IEEE, 1979, to be published.

69. BRIGHAM,E.O.: "The Fast Fourier Transform", Prentice-Hall Inc., 1974.

70. BRACEWELL,R.:"The Fourier Transform and its Applications", McGraw-Hill Inc., 1965.

71. COOLEY,J.W. and TUKEY,J.W.: "An algorithm for Machine Calculation of Complex Fourier Series", Math. Computation, Apr.1965, Vol.19, pp. 297 - 301.

72. BRIGHAM,E.O. and MORROW,R.E.: "The Fast Fourier Transform", IEEE Spectrum, Dec.1967, Vol.4, pp. 63 - 70.

73. WELCH,P.D.: "A Fixed Point Fast Fourier Transform Error Analysis", IEEE Trans. Audio and Electroacoustics, June 1969, Vol.AU-17, pp. 151 - 157.

74. PLESSEY MICROSYSTEMS : "SPM FFT Spectrum Analysers", Plessey Data Sheet, Pub. No. PS4703.

75. RISK,R.J.: "Efficient Hard Wired Digital Fast Fourier Transform Processor", Electronic Letters, Aug.1977, Vol.13, No.16, pp. 458 - 459.

76. CASPE,R.A.:"Array Processors", Mini Micro Systems, July 1978, pp. 51 - 83.

77. HARRIS,F.J.:"On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform", Proc. IEEE, Jan.1978, Vol.66, No.1, pp. 51 - 83.

78. RABINER,L.R.,SCHAFER,R.W. and RADER,C.M.: "The Chirp Z-Transform Algorithm", IEEE Trans. Audio and Electroacoustics, Jun.1969, Vol. AU-17, No.2, pp.

86 - 92.

79. BAILEY,W.H. et. al.: "Radar Video Processing using the Chirp Z-Transform", CCD '75, Int. Conf. on the Applic. of CCD, Oct.1975, pp. 283 - 290.

80. WARDROP,B. and BULL,E.: "A Discrete Fourier Transform Processor using Charge Coupled Devices", The Marconi Review, 1977, Vol.XL, No.204, pp. 1 - 41.

81. MAYER,G.J.: "The Chirp Z-Transform - A CCD Implementation", RCA Review, Dec.1975, Vol.36, pp. 759 - 773.

82. BLUESTEIN,L.I.: "A Linear Filtering Approach to the Computation of the Discrete Fourier Transform", 1968 Northeast Elec. Research and Eng. Meeting Record, Nov.1968, pp.218 - 219.

83. RABINER,L.R.,SCHAFER,R.W. and RADER,C.M.: "The Chirp Z-Transform Algorithm and its Applications", BSTJ, Jun.1969, pp. 1249 - 1292.

84. BERGLAND,G.D.: "A Guided Tour of the Fast Fourier Transform", IEEE Spectrum, Jul.1969, Vol.6-2, pp. 41 - 52.

85. BUSS,D.D. et. al.: "Comparison between the CCD CZT and the Digital FFT", CCD '75, Int. Conf. on the Application of CCD, Oct.1975, pp. 267 - 281.

86. CAMPBELL,J.G.,TAO,T.F. and POLLACK,M.A.: "Sensitivity Study of the Chirp Z-Transform and the Prime Transform as Sampled Analog Discrete Fourier Transform Algorithms", 10th Asilomar Conf. on Circuits, Systems and Computers, Nov.1976, Asilomar,California, Paper No.7.

87. DAVIE,M.C.: "Optimisation of Componentry in a Surface Acoustic Wave Discrete Fourier Transform Processor", Hons. Degree Special Projects Report, Elec. Eng. Dept., Univ. of Edinburgh, Ref.HSP182, May 1976,pp.55-65.

88. RADER,C.M.: "Discrete Fourier Transforms when the Number of Data Samples is Prime", Proc. IEEE, Jun.1968, Vol.56, pp. 1107 - 1108.

89. RETICON: "ARAM-64. Analog Random Access Memory", Reticon Corp. Data Sheet, 1976, No. CA94086.

90. JACK,M.A.,PARK,D.G. and GRANT,P.M.: "CCD Spectrum Analyser using Prime Transform Algorithm",

Electronic Letters, Jul.1977, Vol.13, No.15, pp. 431 - 432.

91. BARRITT,M.M. et.al.: "Edinburgh IMP Language Manual", (Edinburgh Regional Computing Centre,1970)

92. PARK,D.G.: "The Construction and Computer Somulation of a CCD Fourier Transform Processor using the Prime Transform Algorithm", Hons. Degree Special Projects Report, Elec. Eng. Dept., Univ. of Edin., Ref.HSP215, May 1977, pp.24-27.

93. FOSS,R.C. and GREEN,B.J.: "Design Data for High and Low-pass Active Filters", Technical Communication, The Plessey Company Ltd.

94. KELLY,L.C., private communication.

95. GRADSHTEYN,I.S. and RYZHIK,I.M.: "Tables of Integral Series and Products", Academic Press, New York and London, 1965, pp.29-30.

96. ORCHARD,H.J.: "The Synthesis of RC Networks to have Prescribed Transfer Functions", Proc.IRE,vol.39,Apr.1951,pp.428-432.

97. SARAGA,W.: "The Design of Wide-band Phase Splitting Networks", Proc.IRE,vol.38,Jul.1950,pp.754-770.

98. WEAVER,Jr.,D.K.: "Design of RC Wide-band 90-Degree Phase-Difference Network", Proc.IRE,Apr.1954,pp.671-676.

99. DAVIE,M.C.:"Speech Storage Handler", Internal Documentation, Edinburgh University, 1978.

APPENDIX A

RELEVANT PUBLICATION

TECHNIQUES FOR INCREASING THE EFFECTIVE CHARGE-TRANSFER EFFICIENCY OF TAPPED C.C.D. REGISTERS

Indexing terms: Charge-coupled-device circuits, Delay lines

A design technique for multitap c.c.d. delay lines is discussed in which the effective charge transfer efficiency is increased over its intrinsic process-dependent value. The technique involves locating tap amplifiers at every alternate bit, and operating the device at twice the normal clock rate. The advantages of the technique are discussed with reference to a 32-tap, *n*-channel c.c.d. delay line.

Introduction: Recent work has shown how an improvement in the effective transfer efficiency of c.c.d.s may be obtained by the introduction of cell redundancy and circuit complexity.[1, 2] Techniques reported here employ cell redundancy, but involve a minimum of peripheral circuit complexity; these are especially suitable for multitap delay lines, as well as single-output registers.

Consider a c.c.d. register operated in the conventional mode, as shown in Fig. 1*a*. The impulse-response sequence, allowing for transfer inefficiency, has been well studied and an adaption of the result obtained by Vanstone[3] is used here. The *r*th residual of the impulse response sequence at a nondestructive tap *n* can be shown to be

$$\frac{(n+r)!}{r!n!}\varepsilon^r(1-\varepsilon)^n \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (1)$$

where ε is the effective transfer inefficiency per cell and $r = 0$ indicates the main charge packet, $r = 1$ the first residual etc.

The effect of ε is to smear a single charge packet into following signal samples. For low $n\varepsilon$ products, this is limited to a predominant first residual contribution to the immediately following signal sample. This letter discusses two circuit techniques which may be used to reduce the effect of transfer inefficiency, and results are presented for a multitap c.c.d. delay line designed to employ these principles.

Description: The first scheme, shown in Fig. 1*b*, employs alternate input sampling, in which 'fat zeros' are interposed between the signal packets. By sampling only the signal packets at output taps, the contribution of the preceding signal sample is reduced from a first to second residual effect, the intervening 'zero' having absorbed the comparatively large first residual.

The second scheme, shown in Fig. 1*c*, provides self cancellation of the first residual loss, as well as a reduction in the contribution of preceding signal packets. Each input signal sample is injected in two successive charge packets, and the second charge packet of each pair is sampled at output taps. The reduction of the second packet by transfer inefficiency is compensated by the addition of the (ideally) identical loss of the leading packet during each transfer. The leading charge packet also reduces the residual effect of preceding signal samples.

As both schemes halve the effective data rate, it is necessary to double the clock frequency and the number of stages to achieve the same sampling criteria and time–bandwidth product, the [illegible] signal charge packets may be suitably sampled at alternate stages. For convenience, the schemes are referred to as alternate zero alternate tap (a.z.a.t.) and double sample alternate tap (d.s.a.t.), respectively.

The impulse-response sequence of the a.z.a.t. scheme may be obtained from the conventional response (expr. 1), considering alternate terms only and allowing for 2*n* stages to tap *n*:

$$\frac{(2n+2r)!}{2r!2n!}\varepsilon_2^{2r}(1-\varepsilon_2)^{2n} \quad . \; . \; . \; . \; . \; . \; . \; . \quad (2)$$

where ε_2 is the value of ε at the new clock frequency $2f_s$ (ε_1 will be taken as the value of ε at f_s). That of the d.s.a.t. scheme is obtained by considering alternate terms of the conventional response to two impulses delayed by one clock period with respect to one another, again allowing for 2*n* stages:

$$\frac{(2n+2r)!}{2r!2n!}(1+2n\varepsilon_2/(2r+1)+\varepsilon_2)\varepsilon_2^{2r}(1-\varepsilon_2)^{2n} \quad . \quad (3)$$

A summary of these results is given in Table 1, where the techniques are compared in terms of a quality factor *R*, defined as the magnitude ratio of the first residual to the main charge packet.

The results for the d.s.a.t. scheme are identical to those achieved by the scheme proposed by Tozer and Hobson;[1] in fact, the principle is similar. However, addition of the main charge packet and its first residual is here accomplished automatically during each transfer, rather than by peripheral circuitry.

Limitations and comparisons: The benefit obtained from these techniques is reduced at high operating frequencies where doubling the clock frequency may degrade ε_2 significantly. Indeed, an upper limit on the operating frequency may be determined by the criterion

$$\varepsilon_2^2 < \varepsilon_1/2n \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (4)$$

Clearly, the schemes perform best at clock frequencies where ε is effectively constant; it is convenient to compare the results under these conditions.

Table 1 shows that, where $\varepsilon_2 \simeq \varepsilon_1$, both schemes offer an improvement over conventional operation. For practical $n\varepsilon$ values, the improvement in quality factor *R* is approximately $1/2n\varepsilon$. Comparison of the two proposed techniques shows that double sampling provides a marginal performance improvement on alternate zero operation. Reference to Fig. 1 also shows that implementation of the d.s.a.t. scheme is slightly simpler. It is thus concluded that double sampling is preferable to alternate zero operation.

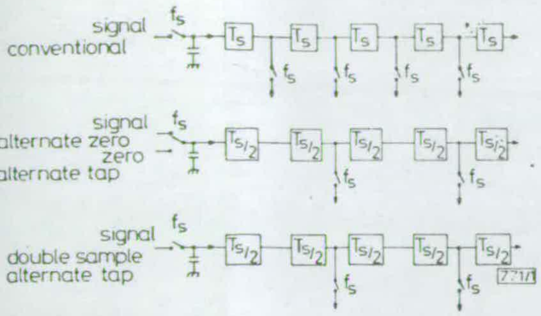
Device considerations: The increased number of transfer cells appears initially to be a disadvantage. For multitap-register applications, however, the increase in available silicon area may be advantageous where posttap signal processing is required on chip. The distance between tap outputs is often

Table 1 COMPARISON OF THREE TECHNIQUES

	Impulse response (residual *r*) (tap *n*)	$r_1 : r_0$ performance ratio		
		General	Large *n*	$n\varepsilon = 0{\cdot}1$
Conventional	$\frac{(n+r)!}{r!n!}\varepsilon_1^r(1-\varepsilon_1)^n$	$\varepsilon_1(n+1)$	$n\varepsilon_1$	0·100
Alternate zero alternate tap	$\frac{(2n+2r)!}{2r!2n!}\varepsilon_2^{2r}(1-\varepsilon_2)^{2n}$	$\varepsilon_2^2(n+1)(2n+1)$	$2(n\varepsilon_2)^2$	0·020
Double sample alternate tap	$\frac{(2n+2r)!}{2r!2n!}(1+2n\varepsilon_2/(2r+1)+\varepsilon_2)\varepsilon_2^{2r}(1-\varepsilon_2)^{2n}$	$\varepsilon_2^2(n+1)(2n+1)\frac{(1+\varepsilon_2(n+1))}{(1+\varepsilon_2(2n+1))}$	$2(n\varepsilon_2)^2(1-n\varepsilon_2)$	0·018

Reprinted from ELECTRONICS LETTERS 6th January 1977 Vol. 13 No. 1 pp. 31–33

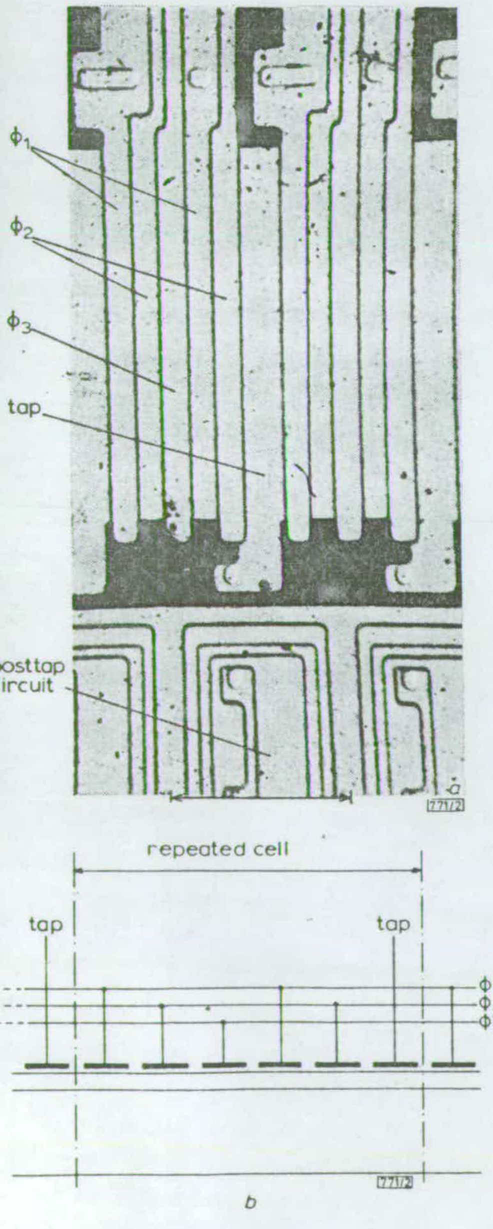
termined by transfer efficiency and process considerations d can be restrictive where identical signal-processing rcuits are required at every tap. The schemes described re may be used to double the area available for peripheral gnal processing circuitry, for a given c.c.d. cell length.



g. 1 *Three modes of c.c.d. operation*
= sample frequency
= sample period

onversely, where the distance between taps is determined posttap circuitry, the schemes allow gate lengths to be lved for a given circuit configuration, permitting higher-equency operation.

Implementation of both schemes involves little peripheral mplexity, the only additional circuit requirement being the neration of sample pulses at half the clock frequency. This practical simplicity makes the schemes very attractive where n improvement in efficiency or increase in available peri-heral circuit area is desired without loss in device perfor-nance. Where the increase in clock frequency and device ength is impractical, it is possible to implement the schemes nd preserve the lower clock frequency by multiplexing two parallel registers.



ig. 2 *Experimental device*
Photomicrograph of c.c.d.
Schematic of cell structure

The principle may be extended to include higher-order sampling and cell redundancy where greater improvements in efficiency or available peripheral circuit area are necessary.

Experimental results: A 64-bit (32-tap) c.c.d. and its peripheral f.g.r.[4] tapping circuitry (Fig. 2) was fabricated with a 'shadow-etch' (s.e.t.) process.[5] The device has gate lengths of 5 μm, with 10 μm tap gates and tap sense amplifiers of 35 μm pitch.

The device was operated with a fill-and-spill input technique. Fig. 3*a* shows the sampled output at tap 13 in response to a pulse input (shorter in duration than the clock period) which was adjusted to give 90% of full well capacity. The quality factor *R* is estimated from the photograph to be 0·04. Fig. 3*b* shows the corresponding output at tap 26, employing the d.s.a.t. technique, and quite clearly a significant improvement in the quality factor has been achieved; in fact, the improvement is such that the effective first residual is difficult to measure. Theoretically, the improved quality factor is approximately 0·003. It is interesting to note that the main response in Fig. 3*b* is the sum of the main response and the first residual in Fig. 3*a*, as predicted by eqn. 3.

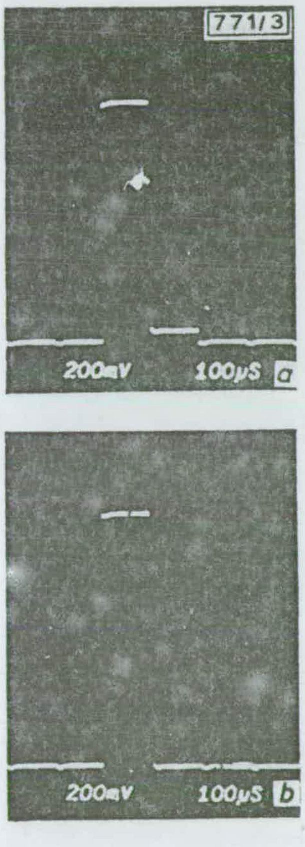


Fig. 3 *Impulse responses*
a Normal operation ($f_c = f_s = 10$ kHz)
b Double sampling (d.s.a.t.) operation ($f_c = 20$ kHz, $f_s = 10$ kHz)

Conclusions: Two techniques have been discussed for improving the effective charge transfer efficiency of multitapped c.c.d. delay lines. Both can be implemented with little increase in peripheral circuitry. The apparent disadvantages of the techniques, twice the clock frequency and device length, should be outweighed in the majority of applications requiring high-efficiency values. The increases in device area may help the layout of the tap amplifiers. The d.s.a.t. technique is marginally better than the a.z.a.t. approach and the efficacy of the former technique has been demonstrated for a 32-bit c.c.d. delay line.

Acknowledgments: The authors gratefully acknowledge the support of the UK SRC, DCVD MOD(PE) and the Wolfson Microelectronics Liaison Unit, University of Edinburgh, for device design facilities. The devices were designed by D. J. MacLennan and fabricated at The Plessey Co. Ltd.

22nd November 1976

J. MAVOR
M. C. DAVIE

Department of Electrical Engineering
School of Engineering Science
University of Edinburgh
Edinburgh EH9 3JL, Scotland

P. B. DENYER

Wolfson Microelectronics Liaison Unit
School of Engineering Science
Mayfield Road, Edinburgh EH9 3JL, Scotland

References

1 TOZER, R. C., and HOBSON, G. S.: 'Reduction of high-level nonlinear smearing in c.c.d.s', *Electron. Lett.*, 1976, **12**, pp. 355–356
2 COOPER, D. C., DARLINGTON, E. H., PETFORD, S. M., and ROBERTS, J. B. G.: 'Reducing the effect of charge-transfer inefficiency in a c.c.d. video integrator', *ibid.*, 1975, **11**, pp. 384–385
3 VANSTONE, G. F., ROBERTS, J. B. G., and LONG, A. E.: 'The measurement of the charge residual for c.c.d. transfer using impulse and frequency responses', *Solid-State Electron.*, 1974, **17**, pp. 889–895
4 MCLENNAN, D. J., MAVOR, J., VANSTONE, G. F., and WINDLE, D. J.: 'Novel tapping technique for charge-coupled devices', *Electron. Lett.*, 1973, **9**, pp. 610–611
5 PERKINS, K. D., and BROWNE, V. A.: 'Sub-micron gap metal gate technology for CCDs', *Microelectron.* 1975, **7**, (2), pp. 14–22

APPENDIX B

MATHEMATICAL ANALYSIS OF CHIRP FILTERING

In the block diagram shown in Fig.B.1 an input sequence is multiplied by a discrete chirp and circularly convolved in a chirp transversal filter. Using the principle of superposition, Fig.B.1 may be regarded as part of the complex CZT processor (Fig.4.7).

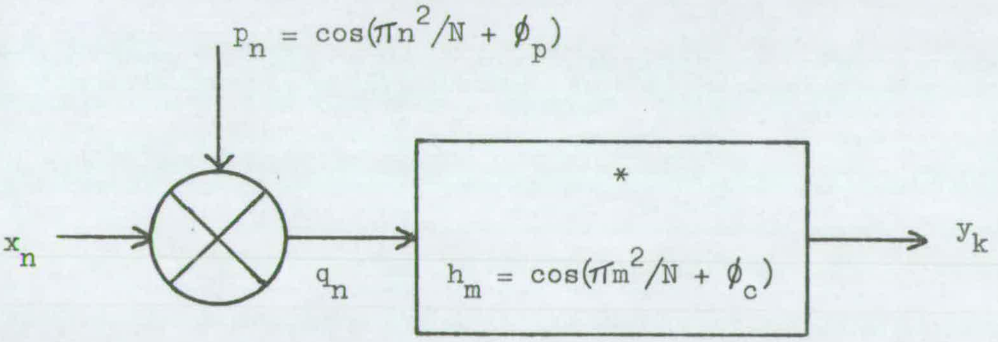

$p_n = \cos(\pi n^2/N + \phi_p)$
x_n
q_n
$*$
$h_m = \cos(\pi m^2/N + \phi_c)$
y_k

Fig.B.1 Chirp Filtering

Consider an input sequence $\{x_n\}$ of the form

$$x_n = A \cos\left[2\pi fn/N + \phi_s\right] \quad n=0,1..N-1 \qquad \ldots \text{(B.1)}$$

where f is the normalised frequency and ϕ_s is an arbitrary phase factor. The multiplication of $\{x_n\}$ by $\{p_n\}$ where

$$p_n = \cos(\pi n^2/N + \phi_p) \qquad n=0,1..N-1 \qquad \ldots \text{(B.2)}$$

results in an intermediate sequence $\{q_n\}$

$$\begin{aligned} q_n = {} & \frac{A}{2}\cos\left[\frac{2\pi}{N}\left(fn + n^2/2\right) + \phi_s + \phi_p\right] \\ & + \frac{A}{2}\cos\left[\frac{2}{N}\left(fn - n^2/2\right) + \phi_s - \phi_p\right] \qquad \ldots \text{(B.3)} \end{aligned}$$

The phase term ϕ_p is either 0 or $-\pi/2$ depending on whether COS or SIN premultiplication is intended.

The filter output sequence $\{y_k\}$ is given by the discrete convolution

$$y_k = \sum_{\lambda=0}^{N-1} q_\lambda \cos\left[\pi(k-\lambda)^2/N + \phi_c\right] \qquad k=0,1..N-1 \quad \ldots \text{(B.4)}$$

where ϕ_c is either 0 or $-\pi/2$. Expansion of the cosine products gives an output consisting of four terms

$$y_k = a_k + b_k + c_k + d_k \qquad k=0,1..N-1 \quad \ldots \text{(B.5)}$$

where

$$a_k = \frac{A}{4}\sum_{\lambda=0}^{N-1}\cos\left\{\frac{2\pi}{N}\left[\lambda^2 + (f-k)\lambda + k^2/2\right] + \phi_s + \phi_p + \phi_c\right\}$$

$$b_k = \frac{A}{4}\sum_{\lambda=0}^{N-1}\cos\left\{\frac{2\pi}{N}\left[\quad + (f+k)\lambda - k^2/2\right] + \phi_s + \phi_p - \phi_c\right\}$$

$$c_k = \frac{A}{4}\sum_{\lambda=0}^{N-1}\cos\left\{\frac{2\pi}{N}\left[\quad + (f-k)\lambda + k^2/2\right] + \phi_s - \phi_p + \phi_c\right\}$$

$$d_k = \frac{A}{4}\sum_{\lambda=0}^{N-1}\cos\left\{\frac{2\pi}{N}\left[-\lambda^2 + (f+k)\lambda - k^2/2\right] + \phi_s - \phi_p - \phi_c\right\}$$

Terms b_k and c_k have linear cosine arguments and can be evaluated using a result from Ref.[95] to give

$$b_k = \frac{A}{4}\cos\left[\frac{(N-1)\pi}{N}(f+k) + \phi_s + \phi_p - \phi_c - \pi k^2/N\right] . \sin\left[\pi(f+k)\right] \operatorname{cosec}\left[\pi(f+k)/N\right] \qquad \ldots (B.6)$$

and

$$c_k = \frac{A}{4}\cos\left[\frac{(N-1)\pi}{N}(f-k) + \phi_s - \phi_p + \phi_c + \pi k^2/N\right] . \sin\left[\pi(f-k)\right] \operatorname{cosec}\left[\pi(f-k)/N\right] \qquad \ldots (B.7)$$

However, both a_k and d_k have quadratic cosine arguments and are therefore similar to discrete forms of the cosine Fresnel integral. No closed solution has been found and a_k and d_k can be evaluated only by numerical analysis.

The four individual terms are plotted in Fig.B.2 for N=64, f=16 (i.e. a basis vector input) and $\phi_s = \phi_p = \phi_c = 0$. Note that although each term is defined only for integer values of k, outputs are also shown for real values. This allows a better understanding of the function type. As expected, b_k and c_k are (sin x)/x functions centred at k=-f and k=+f, each modulated by quadratic phase terms. For integer values of k, only the nulls and peaks of these terms are displayed.

a_k and d_k produce functions which can be likened to Fresnel ripples modulated by quadratic phase factors. These terms are undesirable in matched filtering, but for large time-bandwidth products the filter processing gain tends to make this distortion insignificant. In the CZT, the complex convolver adds and subtracts appropriate filter outputs so

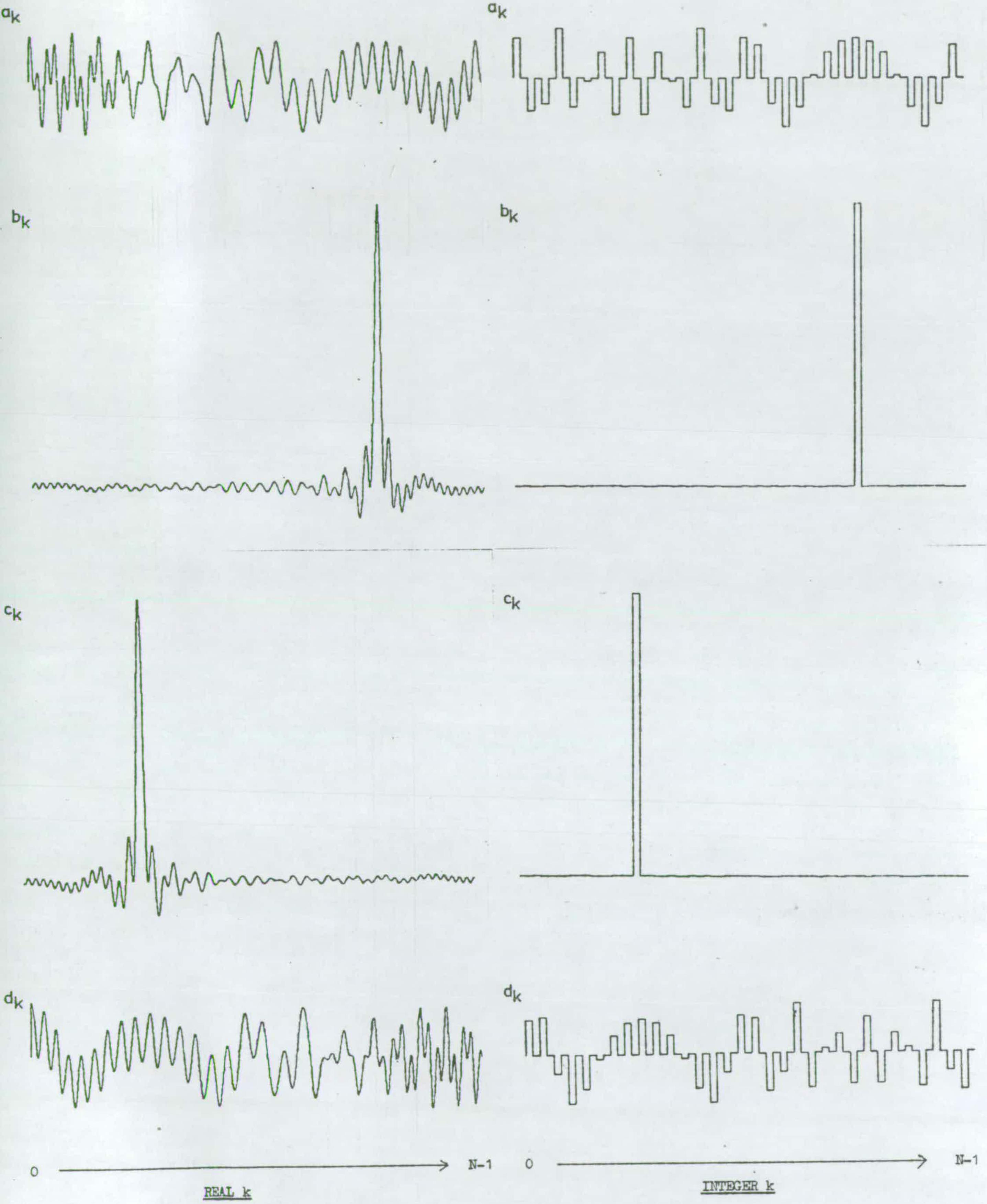


Fig.B.2 Chirp Filter Outputs

that a_k and d_k disappear, and b_k and c_k are reinforced. When the CZT is supplied with a complex input, the even and odd properties of COS and SIN combine to cancel either b_k or c_k, leaving only a single (sin x)/x in the output. The redundant quadratic phase term modulating the (sin x)/x is removed in the CZT either by postmultiplication or by taking the modulus.

APPENDIX C

THE DESIGN OF A 90 DEGREE PHASE DIFFERENCE NETWORK

C.1 INTRODUCTION

The 90° phase difference network described in this appendix has been designed for use in a speech processing system where there is need to generate pseudo complex input data for a real-time CZT processor. The provision of complex data effectively doubles the available processing bandwidth by cancelling the image frequencies present in the spectrum produced by real data only.

The phase difference method of generating complex data is valid for speech since the absolute phase of the voice signal is unimportant to the human ear. Also, in most speech processing systems, only the magnitude spectra are required.

Phase difference errors in the practical configuration will give rise to suppressed image frequencies in the spectrum. To suppress these images by at least 40dB requires a phase difference accuracy of approximately 1° (section 5.2.6).

The main design specifications are therefore:

1. Operating Bandwidth: 50Hz to 3200Hz

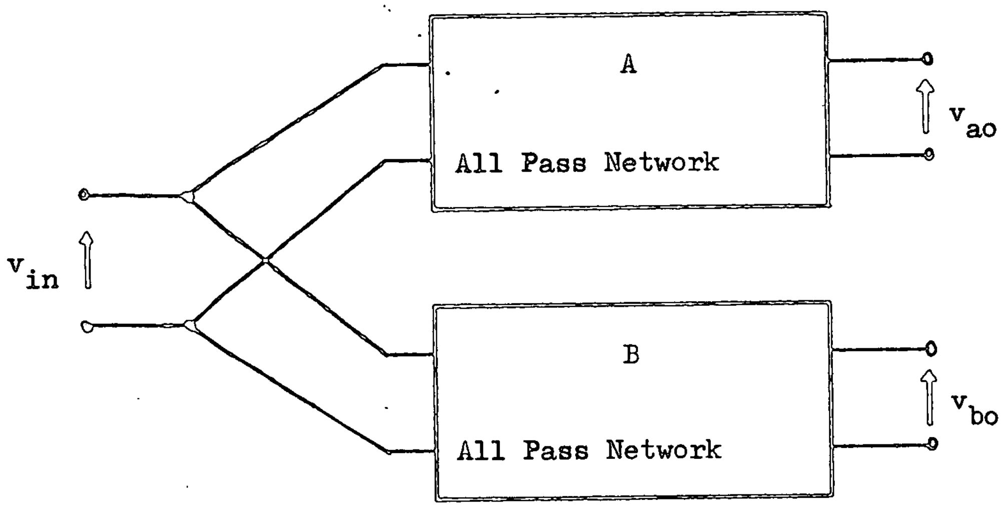
2. Phase Difference Accuracy: 1°

3. In-band Gain: Unity

C.2 THEORETICAL BACKGROUND

The network theory which is relevant to the design and construction of constant phase difference networks has been well understood for many years [96] and has been widely used in single sideband modulation schemes.

It is possible to show [97] that a 90° phase splitting circuit can be designed to operate over a large frequency range by connecting two all-pass networks as shown in Fig.C.1. To make these networks physically realisable, their idealised transfer functions can be approximated by equal-ripple Tschebyscheff functions to give the phase difference function illustrated in Fig.C.2. The two main design parameters of this configuration are (a) the maximum phase difference deviation, ϵ , and (b), the bandwidth ratio f_H / f_L. (a) and (b) together allow the network complexity, n, to be determined.



$v_{in} = A\cos(wt + \theta)$ $v_{ao} = A\cos(wt + \phi)$ $v_{bo} = A\cos(wt + \phi + 90^{\circ})$

Fig.C.1 Phase Difference Networks

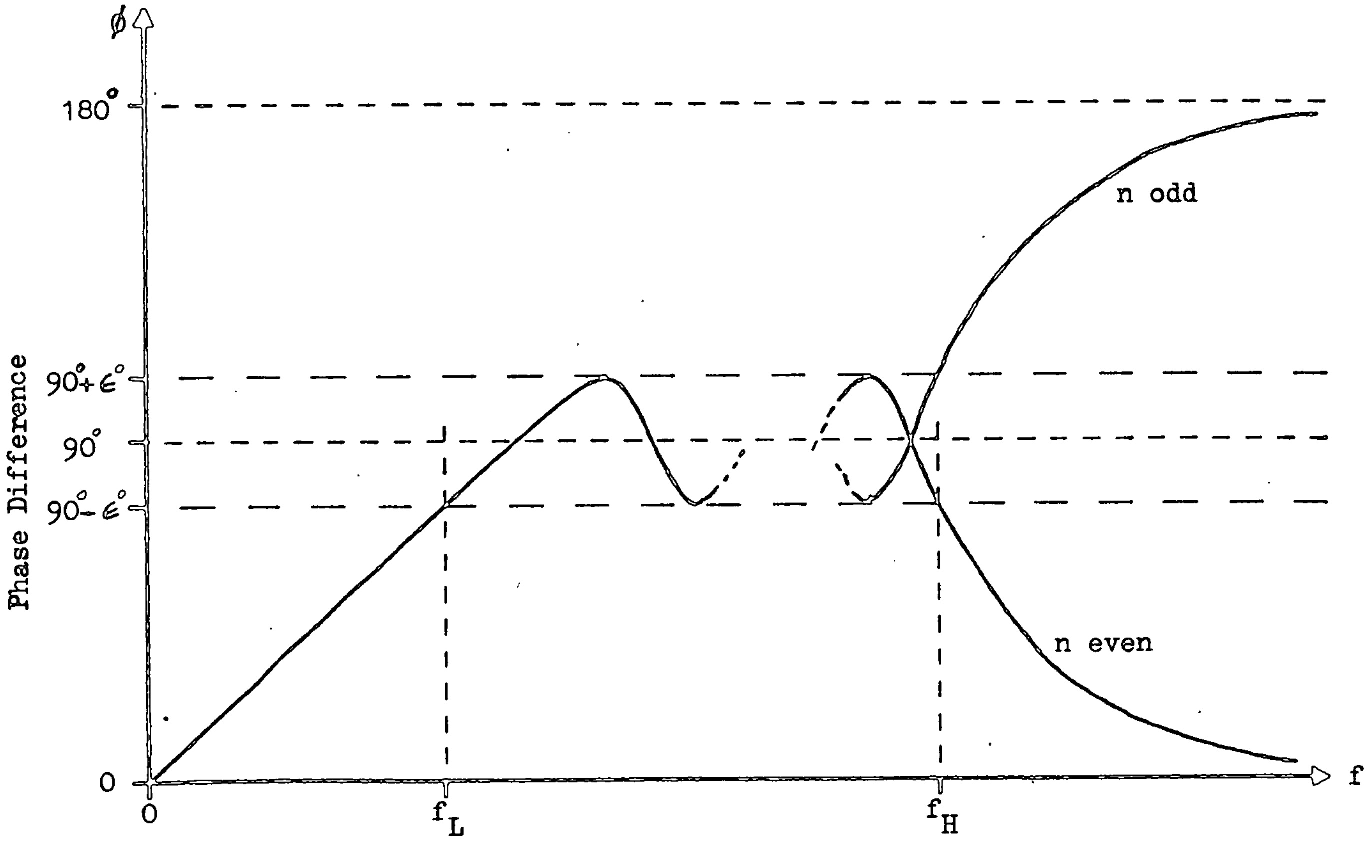


Fig.C.2 Tschebyscheff Phase Difference Function

For circuit operation at low frequencies it is desirable to sysnthesise the transfer functions using resistance and capacitance elements only, since it is difficult to manufacture inductors of adequate quality. This practical restriction requires that the poles of the individual all-pass transfer functions lie on the negative real axis in the complex frequency plane.

The general response functions of networks A and B (Fig.C.1) having the RC restriction are given by

$$H_a(s) = K\,\frac{(s-\sigma_{a1})\,(s-\sigma_{a2})\,(\qquad)\,(\qquad)}{(s+\sigma_{a1})\,(s+\sigma_{a2})\,(\qquad)\,(\qquad)} \qquad \ldots\ (C.1)$$

$$H_b(s) = K\,\frac{(s-\sigma_{b1})\,(s-\sigma_{b2})\,(\qquad)\,(\qquad)}{(s+\sigma_{b1})\,(s+\sigma_{b2})\,(\qquad)\,(\qquad)} \qquad \ldots\ (C.2)$$

where the values of σ are real and positive.

The synthesis problem is therefore to determine the pole-zero pairs for 90° phase difference between $H_a(s)$ and $H_b(s)$ (simplified by the use of the elliptic tangent transformation) and to find RC networks that realise the response function. Note that the realisation problem does not have a unique result since there are many different yet equivalent network configurations and so it is normally necessary to select the most convenient circuit.

A design procedure for the above sysnthesis is given by D.K.Weaver,Jr.[98]. In this design, the all-pass networks A and B are represented by half-lattice configurations (Fig.C.3) where the impedance functions Z_x and Z_y are constructed from canonical RC dipoles.

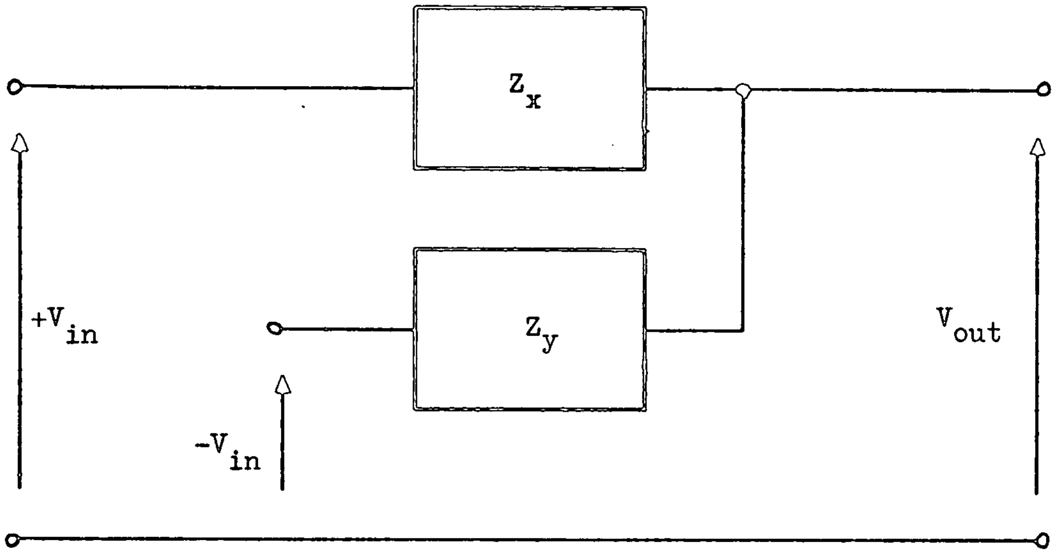


Fig.C.3 All Pass Half Lattice

These networks are driven by balanced inputs.

C.3 DESIGN

C.3.1 Network Synthesis

A computer programme was written in the Edinburgh IMP language to perform the calculations required in the design procedure given by D.K.Weaver,Jr.[98]. This programme is divided into two sections.

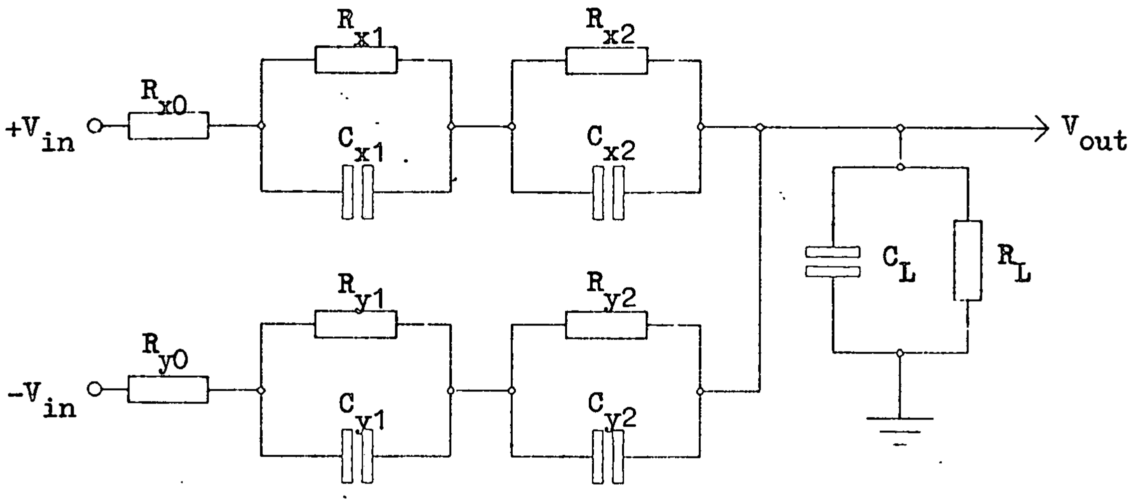
The first section computes the desired network transfer functions from the bandwidth ratio and the network complexity factor, n. The resulting transfer functions for the specifications given in the introduction (n=7) are as follows:

$$H_a(s) = \frac{0.1202\ (s - 0.4030)\ (s - 3.5655)\ (s - 17.9496)\ (s-158.8233)}{(s + 0.4030)\ (s + 3.5655)\ (s + 17.9496)\ (s+158.8233)} \quad \ldots\ (C.3)$$

$$H_b(s) = \frac{0.1202\ (s - 1.4861)\ (s - 8.0000)\ (s - 43.0656)}{(s + 1.4861)\ (s + 8.0000)\ (s + 43.0656)} \quad \ldots\ (C.4)$$

The normalised component values for the chosen network configuration (Fig.C.3) are calculated from the above transfer functions by the second section of the computer programme. The resulting numbers are then multiplied by an impedance factor to give practical component values (Cmin>100pF, Rmax<1M). The final network configuration with the appropriate component values is shown in Fig.C.4.

C.3.2 Network Analysis



Element	A Network	B Network
R_{x0}	SC	SC
R_{x1}	1000.0 k	OC
R_{x2}	19.6 k	100.5 k
R_{y0}	5.4 k	39.0 k
R_{y1}	OC	946.3 k
R_{y2}	35.7 k	SC
R_L	139.3 k	134.6 k
C_{x1}	19.0 nF	6.64 nF
C_{x2}	1.64 nF	761.3 pF
C_{y1}	28.9 nF	3.36 nF
C_{y2}	7.39 nF	OC
C_L	11.1 nF	5.0 nF

OC = Open Circuit SC = Short Circuit

all resistors in ohms

Fig.C.4 Detailed Network Configuration

The normalised frequency response of these networks, $H_a(w)$ and $H_b(w)$ can be obtained by evaluating their transfer functions on the imaginary axis in the complex frequency plane. This is accomplished using the substitution s=jw.

The amplitude frequency response is formed by taking the modulus of H(w) and the phase response by taking the argument. The following equations result from the synthesised transfer functions:

$$|H_a(jw)| = 0.1202 \qquad \ldots (C.5)$$

$$\phi_a(jw) = \tan^{-1}(\,2ab\,/(a^2 - b^2)\,) \qquad \ldots (C.6)$$

where $a = w^4 - 3553.8\,w^2 + 4096$

and $b = 180.74\,w^3 - 11567.4\,w$

$$|H_b(jw)| = -0.1202 \qquad \ldots (C.7)$$

$$\phi_b(jw) = \tan^{-1}(\,2cd\,/(d^2 - c^2)\,) \qquad \ldots (C.8)$$

where $c = 52.55\,w^2 - 512$

and $d = w^3 - 420.41\,w$

The amplitude functions $|H_a(jw)|$ and $|H_b(jw)|$ are the expected all-pass responses with constant attenuation of 0.1202. The phase difference function, ϕ_d, which is $\phi_a - \phi_b$, is plotted in Fig.5.28. (Note that the frequency ordinate has been scaled by f_L). The theoretical phase ripples inside the pass-band are well within the 1° tolerance required.