APPLICATION OF ANALOGUE

CHARGE-COUPLED DEVICES

TO ADAPTIVE SIGNAL PROCESSING

by

C.F.N. Cowan

Thesis submitted for the degree

of Doctor of Philosophy to the

Faculty of Science,

University of Edinburgh

1980

DECLARATION OF ORIGINALITY

This thesis, composed entirely by myself reports work conducted in the Department of Electrical Engineering at the University of Edinburgh exclusively by myself, with the exception of the section mentioned below.

The result presented in section 6.6 is reproduced by kind permission of Dr. J.W. Arthur and H.M.M. Reekie.

signed

C.F.N. Cowan

date: 21 July 1980

ACKNOWLEDGEMENTS

CONTENTS

LIST OF ABBREVIATIONS

ADC   Analogue to Digital Converter

agc   automatic gain control

ARAM   Analogue Random Access Memory

BBD   Bucket Brigade Device

CCD   Charge Coupled Device

CMOS   Complementary Metal Oxide Silicon

CTD   Charge Transfer Device

CZT   Chirp-Z Transform

DAC   Digital to Analogue Converter

DCT   Discrete Cosine Transform

FFT   Fast Fourier Transform

FIR   Finite Impulse Response

FM   Frequency Modulation

LCD   Liquid Crystal Display

LED   Light Emitting Diode

LSB   Least Significant Bit

LSI   Large Scale Integration

MOS   Metal Oxide Semiconductor

MSB   Most Significant Bit

SNR   Signal to Noise Ratio

prf   pulse repetition frequency

PT   Prime Transform

PTF   Programmable Transversal Filter

RAM   Random Access Memory

rms   root mean square

ROM   Read Only Memory

SAW   Surface Acoustic Wave

SPS   Serial Parallel Serial

TDL   Tapped Delay Line

TDI   Time Delay and Integrate

TTL   Transistor Transistor Logic

VLSI   Very Large Scale Integration

CHAPTER 1:   INTRODUCTION

A key element in many signal processing applications is one which is capable of delaying signals in a controlled manner.   Until the invention of the charge-coupled device (CCD)[1,2] in 1970 the circuit techniques used to do this were digital shift registers, or bucket-brigade devices (BBD) in the analogue domain.   The CCD provided an efficient, compact technique for the production of analogue signal delays on a single integrated circuit which opened the way to effective analogue implementations of convolvers and correlators at baseband frequencies.

It is the use of analogue CCD convolutional processors applied to analogue signal processing which is addressed in this thesis.   The particular type of systems considered here may be generally grouped under the heading of adaptive signal processing[3] modules since, in all the experimental systems described in this thesis, the general aim is to make use of particular measured statistical properties of an incoming signal in order to determine some form of filtering or decision function. This class of system has been chosen because of the possibility of enhanced performance[4] which could be available when the filtering characteristics are not fixed (ie variable or self-varying).

1.1 Signal processing with charge-coupled devices

One of the most important characteristics of the CCD is its formation as a straightforward analogue delay line. In this form it finds uses in such diverse areas as audio[5] and video[6] delay lines and simple noise integrators[7] and comb filters.

The applicability of this type of delay is strictly limited unless the signal may be accessed at a number of points along the delay line and then weighted and summed to form a so-called transversal filter structure. Two types of transversal filter have been previously advocated which will be divided into the two groups of committed and uncommitted filtering.

(1) Committed filtering: Here the filtering function is fixed at the processing stage. The commonest type of committed filtering function is the split-gate filter[8], in which the individual weights are implemented by physically dividing the CCD transfer electrode into two portions. The ratio of the lengths of the two portions determines the tap weight for that particular electrode. The signal under each electrode is extracted (non-destructively) and weighted by a current sensing technique[9] and the summation of the weighted signal values is then simply achieved by using a common summing node and current-to-voltage conversion. This is one of the commonest types of fixed filter implementation

using CCD's and some typical applications for these devices will be considered later.

(2) Uncommitted filtering: In this case the filter architecture used is again that of the transversal filter but with the difference that the filter weighting function is not specified at the time of device manufacture, ie the filter is programmable. Three basic structures have been used to implement this approach and these structures are illustrated in figure 1.1.

(a) the 'organ-pipe' structure[10]: This structure basically uses a number of CCD delay lines of different lengths which have a common input node. The delay line outputs then yield values of the input signal at various delay times. These outputs may then be weighted using a resistor summing network or an array of multipliers (cf. figure 1.1(a)). This structure is unwieldy due to repetition of identical delay paths many times in the overall filter.

(b) the tapped delay line[11] structure (TDL): In this structure only one delay line is used and the signals are made available for weighting by the use of non-destructive tapping[12] of the delay line which replicates the signal under an electrode without destroying the charge packet. The resulting signal samples may again be weighted by resistive summing networks or by the use of multipliers (cf. figure 1.1(b)). This structure does have the disadvantage of requiring extra silicon area to accommodate the tapping amplifiers
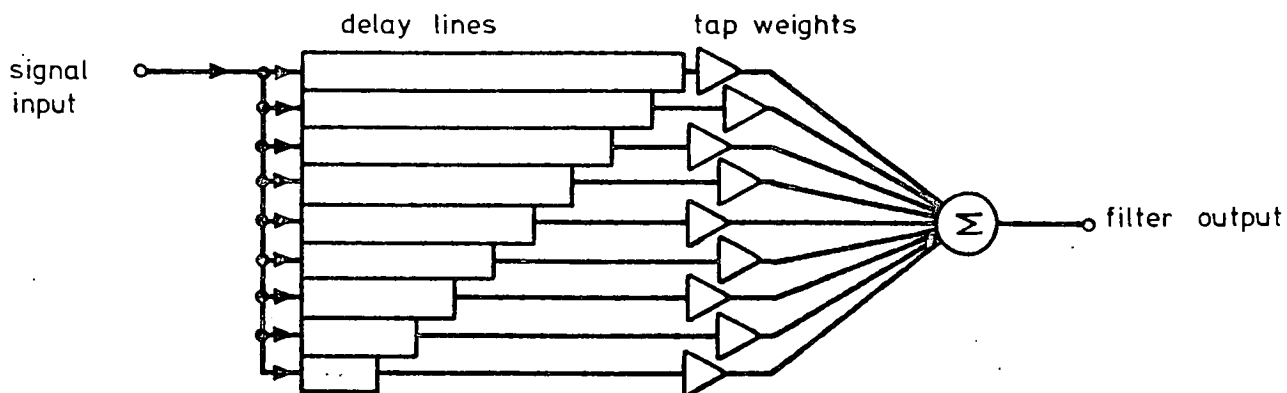
(a) organ-pipe structure
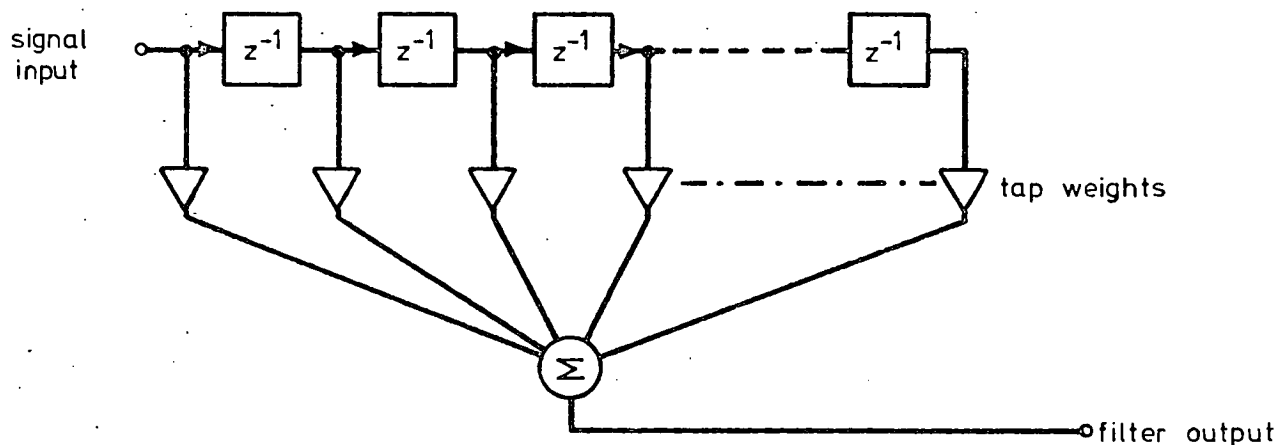
(b) tapped delay line structure
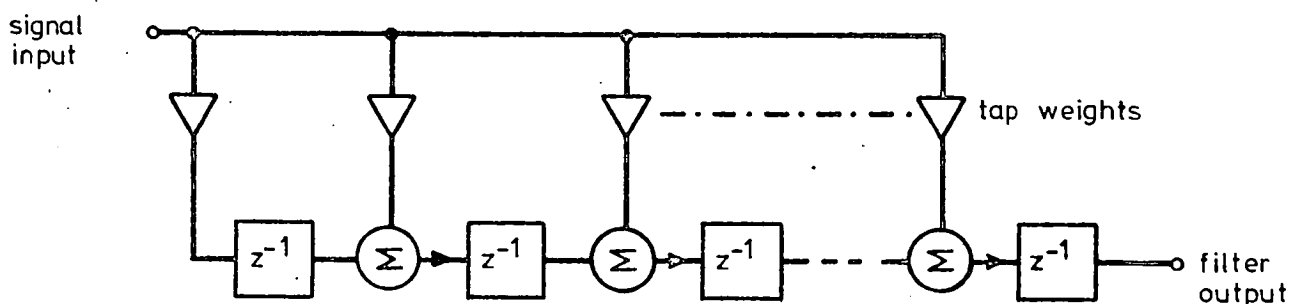
(c) time delay and integrate structure

Figure 1.1 : Block diagrams of three alternative schemes
for programmable transversal filter
implementation.

and the resulting tapped delay lines do tend to have slower clocking rates than the other two alternatives. This structure does however make more efficient use of silicon area than the 'organ-pipe' structure.

(c) time delay and integrate (TDI)[13]: The TDI structure works by reversing the order of operations considered in the previous two cases. All inputs are taken as being common but signal weighting is done before entry into the CCD. Each input then occurs at successive delay points with the charge packet from the previous stage being added in one clock cycle later so that the weighted input signals progressively accumulate (or integrate) with the filter result being available on the final transfer electrode (cf. figure 1.1(c)). This structure does have the requirement of a progressive increase in size of transfer electrodes due to the accumulating size of the charge packets and this yields the characteristic triangular shape of the TDI. This particular characteristic provides an upper limit on the length (ie. the time-bandwidth product) of the TDI.

Although the TDI and 'organ-pipe' structures have faster clocking rates (due to the fact that they have no tapping points in their structure) than the TDL structure they are not so suitable for long filters since the TDL size increases by a factor of 1/N for each successive filter stage added (N = the total number of filter stages) whereas the TDI and 'organ-pipe' size increases by a

factor of 2/N and the overall area is proportional to $N^2/2$ against the TDL area which is proportional to N. It is therefore clear that the TDL structure is more suited to long (high time-bandwidth product) filters with relatively low bandwidths as it makes more economical use of silicon area than either of the above alternatives.

Two distinct types of uncommitted filter may be identified and these have been termed as field programmable[10] filters. The field programmable filter is typified by a TDL with a resistive weighting function, where the filter characteristics may only be changed by altering the resistor values, this type of system is then directly comparable to a split-gate filter where the weights are readily programmed by the user. The real-time programmable filter may be defined as a filter in which the tap weights may be altered at a rate similar to the clock rate of the CCD. This is done by weighting with multipliers using coefficients stored on some external circuit (RAM or ROM usually). Many possible schemes have been proposed for this type of structure (which will be referred to as a programmable transversal filter or PTF) which use either analogue multipliers[14,15] or multiplying digital to analogue converters[16] (MDAC's) to perform the weighting function.

A great deal of work has been done on the filtering structures just described with particular reference to

their application in a number of areas.

(1) Spectrum analysis: The CCD has made a significant contribution in the area of spectrum analysis, particularly to the implementation of the chirp-z transform (CZT) using split-gate chirp filters[17]. Efforts have also been made to implement the prime transform algorithm[18] and more recently PTFs have been applied to yield a very efficient realisation of the discrete cosine transform[19] (DCT).

(2) Frequency filtering: This is an area almost entirely dominated by the split-gate CCD filter which has been used to implement anti-aliasing filters[20] and band-pass filters for use in spectrum analyser filter banks and vocoders[21].

(3) Matched filtering and pulse compression: In this area the CCD based PTF has found application to pulse compression of sonar return pulses[22] and matched filtering in spread spectrum communications equipment[23]. The small size and low power consumption of the CCD has made it a most attractive alternative to digital solutions in this application area.

(4) Adaptive filtering[3]: It is in this particular application that the PTF has its most important role since it is essential here that the filter tap weights should be updated at the CCD clock rate. This area is a relatively new one but a few schemes for adaptive filter implementation have been published[24-27].

Many of the applications mentioned above are being
approached using entirely digital circuit techniques and
with the ever shrinking size and cost of digital
components it appears more and more likely that they will
be used in prefernce to the more elegant CCD solutions in
many areas of signal processing. This is particularly
true when the poor temperature stability and difficult
'setting-up' procedures of contemporary CCDs is taken into
account. It would then appear that the CCD will only be
used in areas where space and power are primary
considerations or where the structures involved are
insensitive to temperature drift due to the inherent
systematic structure.

This is the specific category of system which is
considered in this thesis under the general heading of
adaptive signal processors. Two classes of processor are
considered: (1) open loop adaptive systems which have
specific tasks assigned to them, and (2) closed loop or
true adaptive systems which are much more general purpose
machines.

The open loop processor is one in which a number of a
priori assumptions are made about an incoming signal
sequence and the filtering function is in fact fixed.
The first is a statistical analyser module[28] which is
capable of measuring the statistics of an incoming signal
and then making a decision, with a defined confidence

level, on the presence or absence of a signal peak. This
processor was specifically conceived as a decision making
circuit to follow a sonar matched filter in an environment
where size and power consumption are of primary
consideration.

The second module was a noise canceller[7] used to cancel
unwanted fixed pattern noise on the output of a serial-
parallel-serial (SPS) delay line. The technique used was
a simple coherent integration technique. The important
point here was that the canceller should be implemented in
a technology which was compatible with the SPS delay line
to permit integration on the same integrated circuit and
CCD techniques therefore had to be used.

Although both the processors just described were
necessarily implemented using CCD circuits the stability
of both systems was not entirely satisfactory. This
prompted the construction of the second system type
mentioned above, the closed loop adaptive filter.

Four experimental adaptive filter modules were constructed,
two of which were based around a monolithic 64-point CCD
PTF[14]. These two prototypes[27,29] demonstrated the
basic feasibility of using CCD elements for adaptive
filtering. However, quantitative measurements were
difficult to perform on these modules due to their lack of
controllability. The third prototype[30] which was based

around a monolithic 256-point PTF[31] was constructed, which incorporated the necessary controllability and therefore allowed a quantitative comparison of filter performance with theoretical predictions. The fourth prototype was an all-digital module[32] which was used to provide a comparison with results obtainable using conventional technology. The results generated with the 256-point adaptive filter in particular may be readily compared with theory to demonstrate the quality of performance available using CCD based adaptive systems. An additional adavantage of the closed loop system is that it is insensitive to temperature drift effects and a corresponding decrease in 'setting-up' procedure complexity is achieved.

1.2 Layout of thesis

A number of integrated circuits not commonly available commercially were used in the construction of the experimental systems just described. A brief description of the more important characteristics of these devices is given in chapter 2 together with an overview of the basic operating principles of the charge-coupled device.

Chapter 3 deals with the theory and construction of the two open loop systems described in the last section. Experimental results from the two prototype systems are also presented which demonstrate the principles of

operation of these systems.

The remainder of the thesis is devoted to a description of the closed loop experimental systems. Chapter 4 introduces the basic principles of the operation of an adaptive filter and describes some of their possible applications. In chapter 5 a number of solutions to the problem of adaptive filter implementation are presented with particular reference to specific circuit techniques. System architectures which are specifically suited to analogue and digital implementation are advocated and the actual system structures used in the construction of the three 64-point adaptive filter prototypes are described.

One of the most intriguing possibilities with an adaptive filter structure is that of inherent correction of errors within the filter itself. This aspect of adaptive filter operation is investigated theoretically in chapter 6 with specific reference to error sources such as charge transfer inefficiency, tap gain errors and multiplier inaccuracies. Experimental verification of some of the theoretical predictions in this chapter is also included in the last section of this chapter.

Chapter 8 is devoted to a description of the 256-point adaptive filter prototype (full details of the construction of this module are given in Appendix A). Experimental results for this module are also presented in

chapter 8 demonstrating the high degree of predictability of this processor from theoretically ideal models.

Chapter 9 contains a discussion of the results presented in the previous chapters and the author's conclusions.

CHAPTER 2:   CCD CIRCUITS

In the work described in the following chapters a number
of novel charge-coupled device integrated circuits were
used which are not commercially available. A large part
of this chapter is, therefore, devoted to an explanation
of the particular operational aspects of these devices
which are relevant to the work considered. In addition,
the following two sections briefly review the basic
operation of the charge-coupled device in terms of charge
transfer and, input and output mechanisms.

2.1   Charge transport mechanisms

The basic function of the CCDs considered in this section
is that of delay of sampled analogue data. In a CCD,
analogue data samples are stored as charge packets held
in the depletion region under an electrode with a positive
voltage bias (for an n-channel device). If it is first
assumed that no signal charge is available then the
depletion region may be conveniently viewed as a potential
'well' of a depth which is proportional to the gate bias
applied to the CCD electrode. Introducing a signal charge
packet into this well causes the well to fill up, with a
resulting drop in the surface potential. This is
illustrated diagramatically in figure 2.1 which shows how
the depth of the potential well decreases as the amount of
signal charge stored under the electrode increases[33].
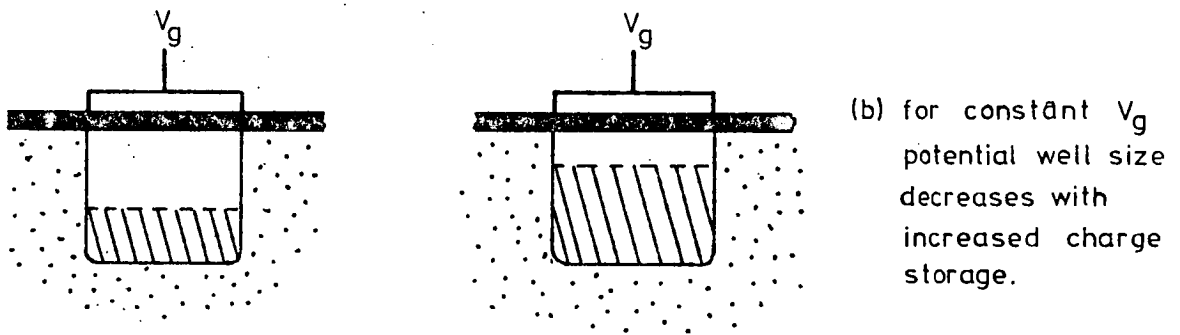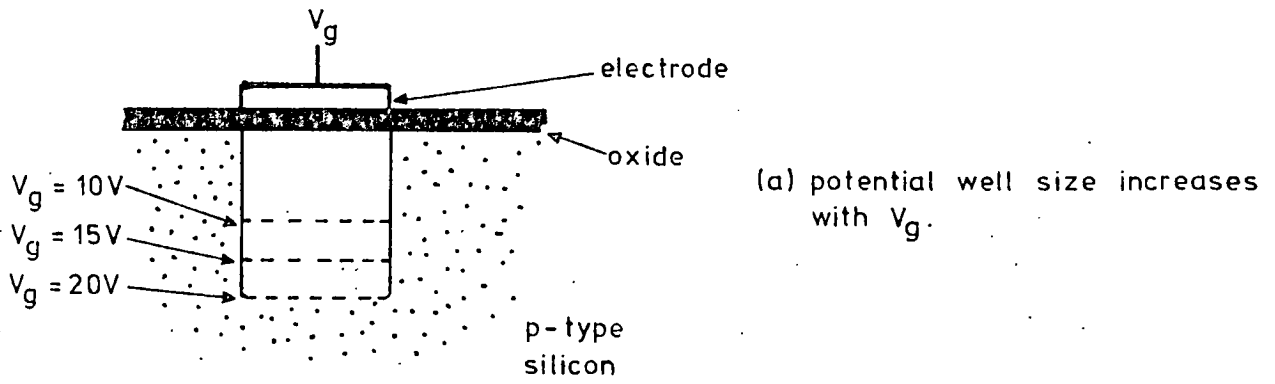
Figure 2.1: Diagram illustrating the potential well concept
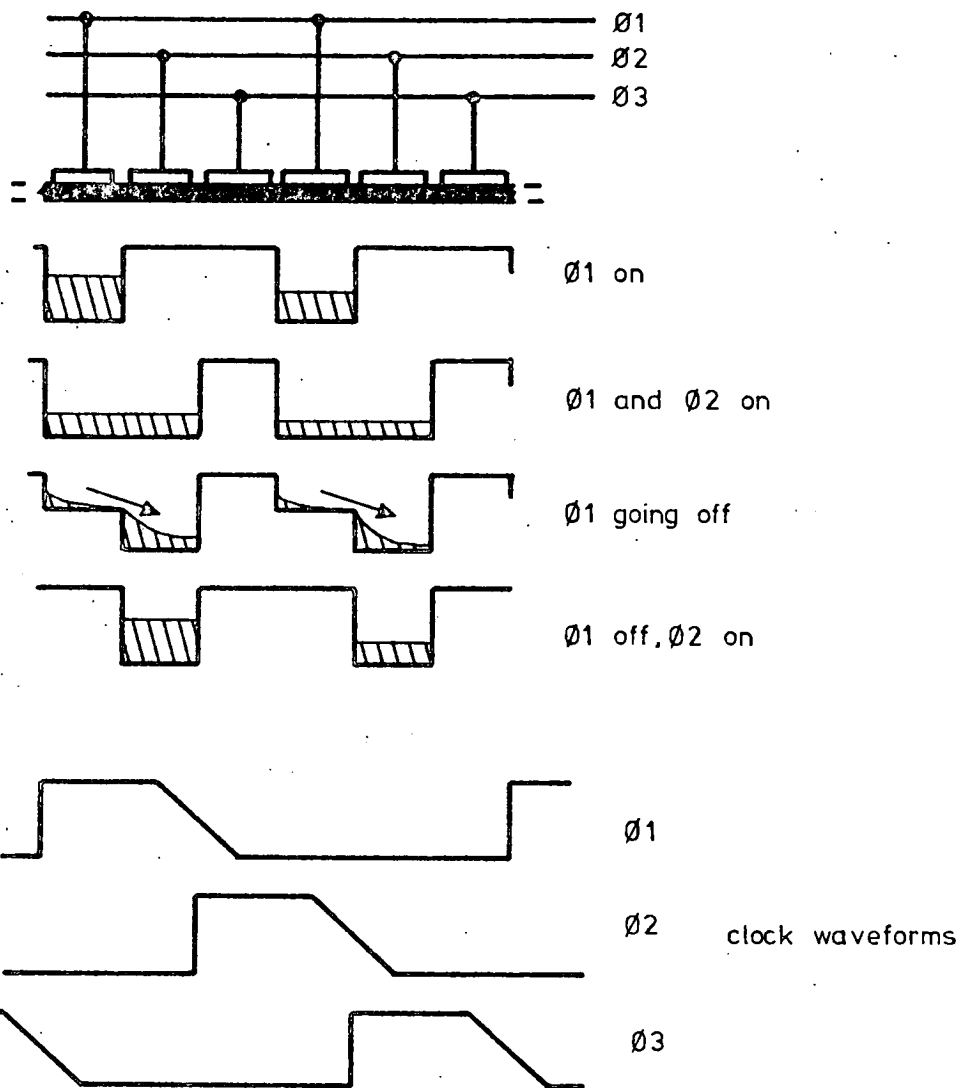
Figure 2.2: Transfer of charge using a 3-phase CCD clocking structure.

The transfer of charge packets from one potential well to another may be achieved in a number of ways, usually described in terms of the clock structure needed to perform a unit delay. In this section two charge transfer schemes will be described, one using a 3-phase clock structure and the other using a 2-phase clock structure. In the following discussion the word 'phase' will be replaced by the symbol '∅'.

Charge transfer with a 3∅ clock structure is achieved by the use of 3 successive CCD electrodes forming a single delay 'cell'. With reference to figure 2.2, the signal charge packet is initially stored under electrode 1 and its corresponding clock ∅1 is therefore on (positive potential) while the other two electrode clocks, ∅2 and ∅3, are off (zero volts). The second clock, ∅2, now comes on so that the signal charge is shared between electrodes 1 and 2, ∅1 now switches off and the charge remaining under electrode 1 spills into the well under electrode 2. No reverse charge flow occurs since ∅3 is off during this transfer sequence and charge flow therefore occurs from left to right only. The charge transfer is completed by transferring the signal charge packet from electrode 2 to electrode 3, following the same procedure as that described above for transfer between electrodes 1 and 2.

The main reason for employing this 3∅ clock structure is to prevent the reverse flow of charge along the delay line.

It is clearly demonstrated how this is achieved with the 3∅ structure in figure 2.2 since a barrier, in the form of at least one transfer electrode in the off state, always exists between adjacent transfer cells.

Charge transfer using only 2 clock phases may however be achieved by the use of implanted barriers or stepped oxide thickness. The stepped oxide thickness technique will be the one described here. The important point here is that the threshold voltage of two adjacent electrodes differs, so that when they are switched on together the depth of the potential well under a given electrode pair is stepped. This is shown in figure 2.3, where the threshold voltage difference is achieved by using a thick and then a thin oxide layer covered by one electrode.

The transfer of a charge packet from one well to the next is shown in figure 2.4 where the charge is initially held under the second half of the ∅1 electrode with ∅1 on and ∅2 off. As ∅2 comes on and ∅1 goes off the charge is dumped over the first potential barrier under the ∅2 electrode and accumulates in the lower part of the 'step'. In this way backward flow of charge is minimised in a 2∅ CCD structure, as long as the signal charge packet size is significantly smaller than the size of the potential step between the two half-electrodes.

Despite the structural and clocking details just described

off   on
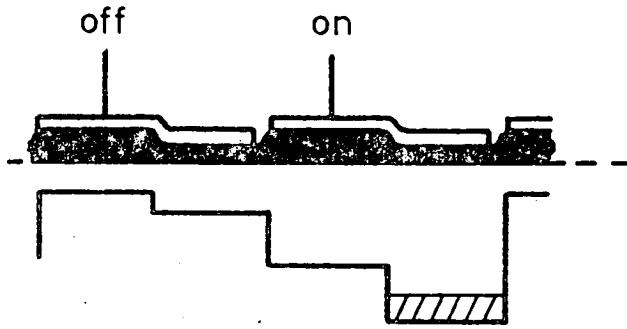
Figure 2.3: 2∅ CCD structure illustrating stepped well
sizes due to different oxide thickness.

∅1
∅2

∅1 on

∅1 going off

∅1 off
transfer completed
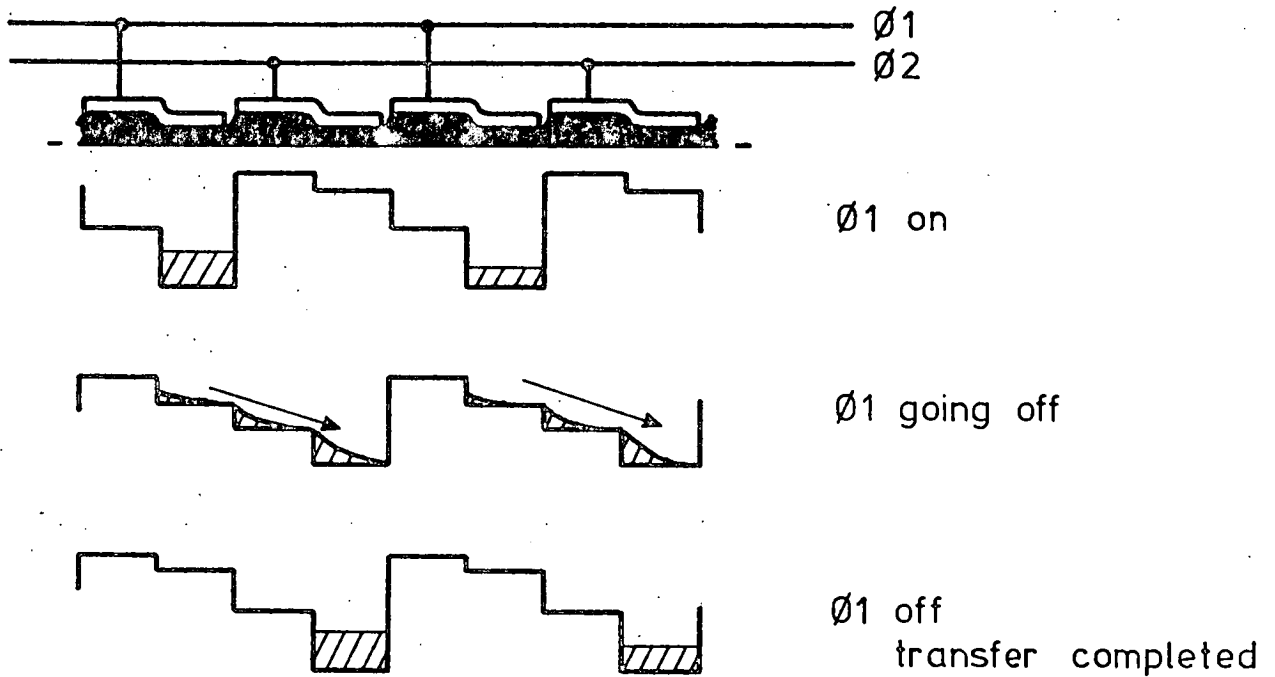
Figure 2.4: Transfer of charge in a 2∅ CCD structure.

backward flow can still occur in both 2∅ and 3∅ structures and this leads to the phenomenon known as charge transfer inefficiency[34]. This is the incomplete transfer of a signal charge packet from one CCD electrode to the next. An additional factor which leads to transfer inefficiency is the existance of surface trapping states under an electrode. Charge carriers are trapped by these states and remain until the next charge packet is transferred in. The overall effect is that when a charge packet is transferred a small proportion ξQ (where Q represents the signal charge and ξ is a constant usually between 10^{-3} and 10^{-4}) is left behind and combines with the following charge packet. All the charge eventually leaves the delay line but the signal is distorted or 'smeared' by the inefficiency effect. The only way in which charge is actually destroyed is by carrier recombination in the semiconductor bulk material.

An overall figure of merit used to describe the charge transfer inefficiency is the $n\xi$ product, where n is the total number of transfers which take place in the delay line. The effect of transfer inefficiency is illustrated in figure 2.5 where the input to the CCD is a single pulse and the form of the delay line output is shown for progressively larger $n\xi$ products. This effect clearly imposes an upper limit on the length of delay achievable for acceptable signal distortion.
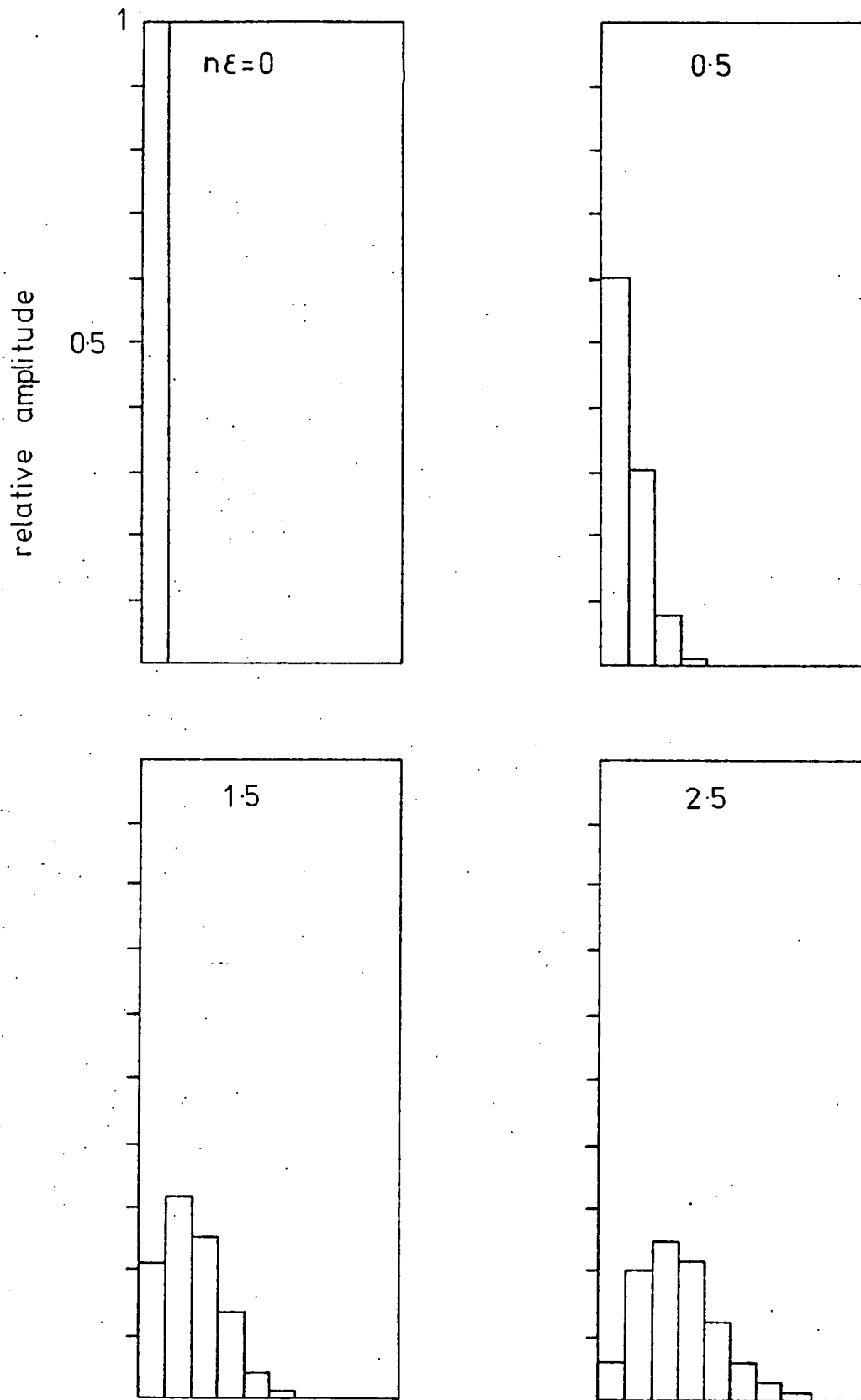
Figure 2.5: Output of a CCD delay line with a unit pulse input for varying $n\xi$ product.

2.2 Signal input techniques

A variety of signal input techniques may be used with CCD structures of which two will be described in this section; diode cut-off[35] and 'fill and spill'[36].

The structure at the CCD input is similar for both input techniques and is shown in figure 2.6. This consists of an implanted diode followed by a gate electrode immediately preceding the first transfer electrode. In the diode cut-off technique the input signal is applied directly to the diode. Initially the $\emptyset 1$ clock is on, therefore a potential well exists under the $\emptyset 1$ electrode. However, no charge is transferred from the input diode as long as the input gate clock remains off. When the input gate is pulsed high an amount of charge proportional to the signal amplitude is dumped under the $\emptyset$ electrode and the input gate is switched off before the $\emptyset 1$ to $\emptyset 2$ transfer takes place, therefore isolating the input diode from the transfer process. This technique is illustrated in figure 2.7. Diode cut-off does result in some non-linearity due to the fact that when the input gate switches off the charge under it tends to divide between the input diode and the $\emptyset 1$ transfer well resulting in an inaccuracy in the amount of the charge injected.

The fill and spill technique is shown in figure 2.8, where it can be seen that the arrangement of electrodes
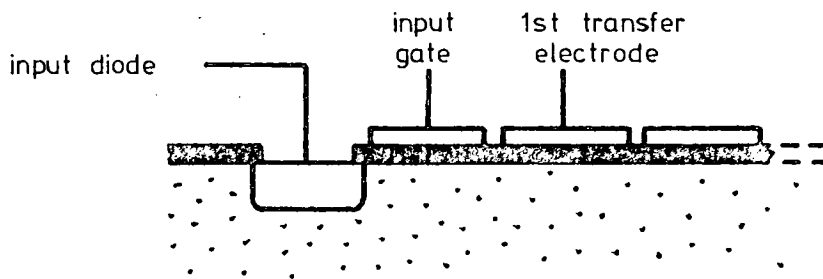
Figure 2.6: CCD input structure.
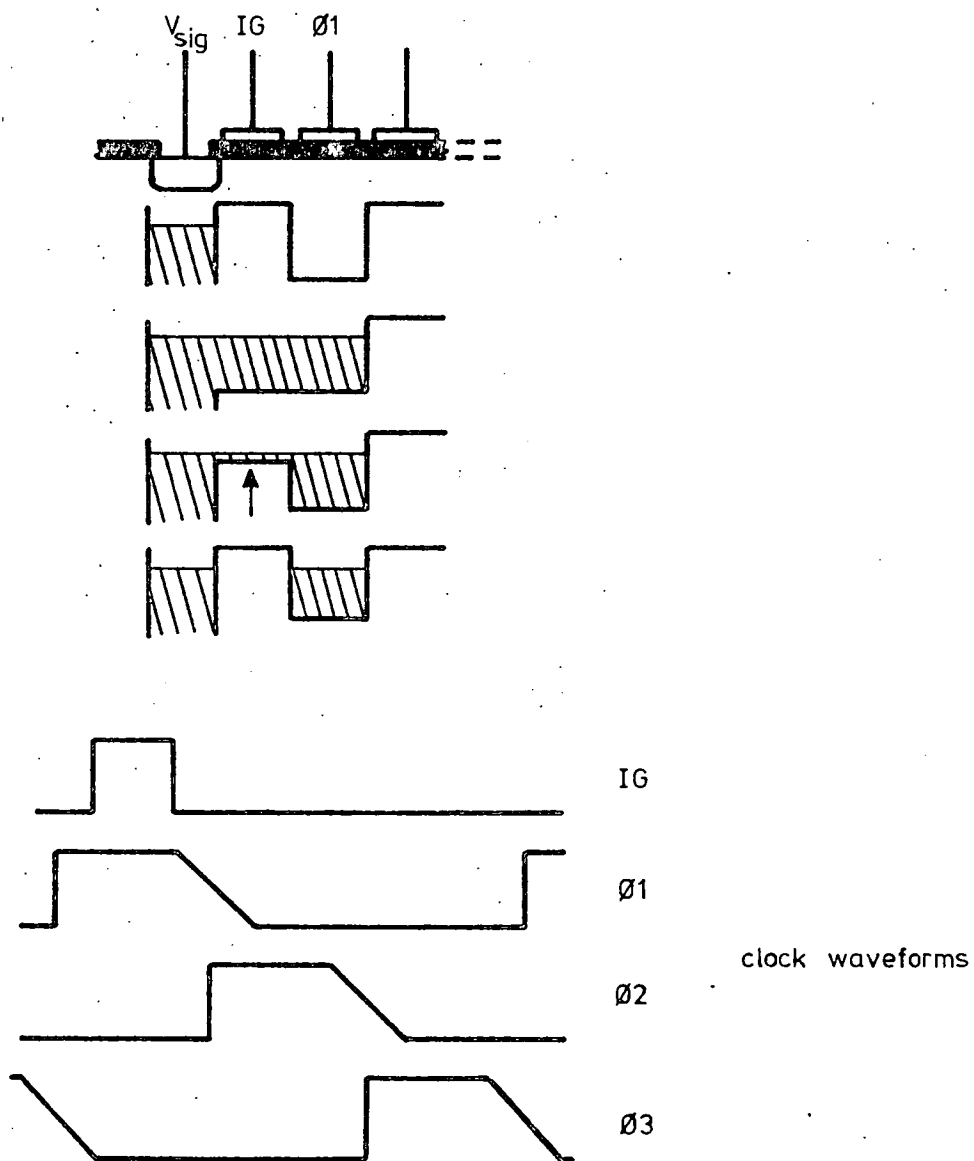
Figure 2.7: Diode cut-off input technique.
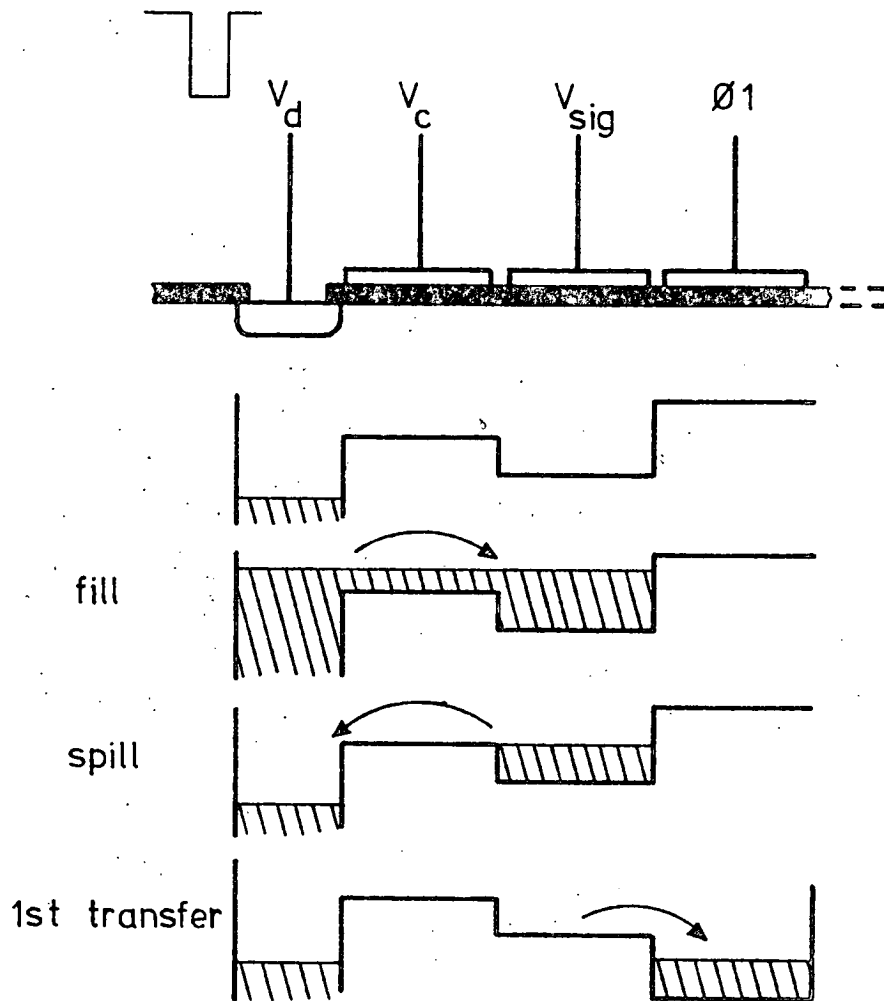
Figure 2.8: Fill and spill input technique.

at the input differs from that described above. The
input signal voltage is actually applied to a second gate
electrode immediately before the first transfer electrode.
The first input gate is held at a constant potential to
form a potential barrier between the input diode and the
input gate. With $\emptyset 1$ held off the diode is pulsed on,
filling the potential well under the input gate (this
potential well depth is proportional to the input
voltage). When the diode goes off the excess charge under
the input gate spills back over the first gate barrier
leaving a charge packet proportional in size to the input
signal. Normal charge transfer may then take place to
the $\emptyset 1$ electrode position. Since there is no cut-off
applied here there can be no forward spillage of charge
into the input gate potential well and this input
technique therefore provides a more linear input scheme
than the diode cut-off.

2.3 Tapped delay line implementation

In the previous sections the basic mechanisms for charge
input and transfer in a CCD were considered; in this
section a technique for non-destructively sensing the
amount of charge under an electrode is described. The
reason for doing this is to form a tapped delay line
structure which is the basic element required to form a
programmable transveral filter (as described in chapter 1).

Two tapped delay line devices were used in the work
described here both of which used a floating gate reset[12]
(f.g.r.) tapping circuit implementation. One of the
devices used a 3ϕ clocking structure (the M55 tapped delay
line) with two complete delay stages between each tapping
point while the other was a 2ϕ structure with 1 delay
stage between each tapping point (the WM2010 tapped delay
line).

The mechanism of f.g.r. tapping will be described here
with reference to the 3ϕ clocking structure used in the
M55 TDL. The basic technique relies on the use of a
slightly different transfer mechanism than that already
stated, in that one of the clock phases is held at a
constant voltage about half way between the on and off
levels. For the M55 device an electrode is on when +25V
is applied to it and off at +2V, the intermediate voltage
used on the tapping electrode was about +15V. The charge
transfer mechanism using this arrangement is shown in
figure 2.9 for a group of 3 electrodes showing the pseudo-
2ϕ operation now obtained.

The charge is sensed while under this second electrode
using the circuit shown in figure 2.10. The potential on
the sense electrode is reset during the ϕ2 phase by
pulsing high the gate of the reset transistor with ϕ2.
When the transfer of charge from the ϕ2 to the sense
electrode takes place the sense electrode is floating and
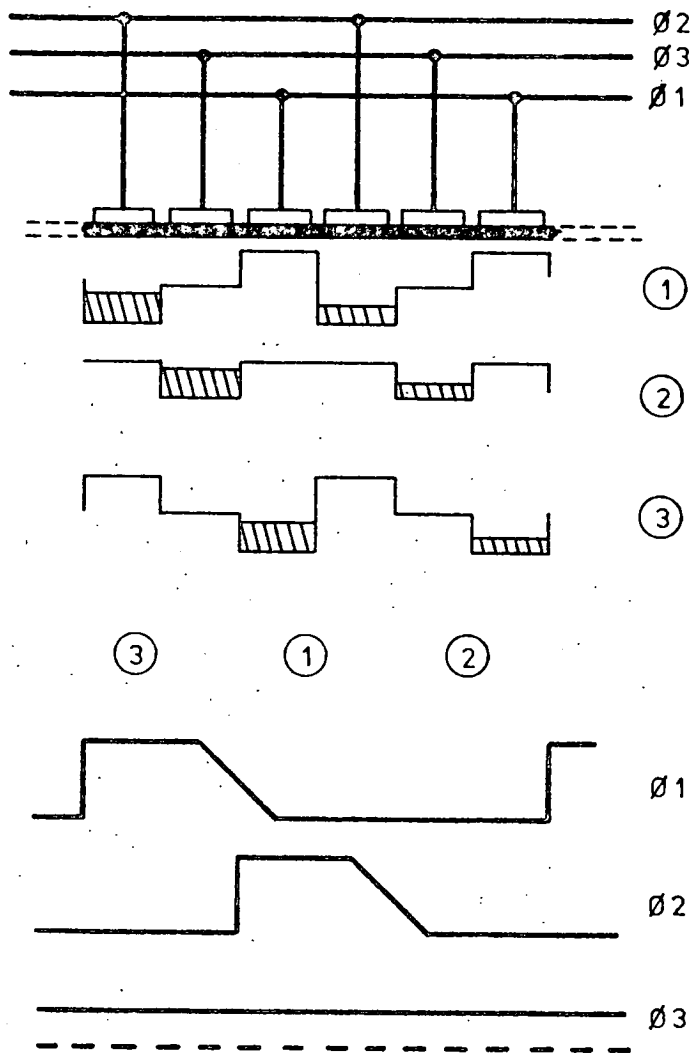
Figure 2.9: Pseudo-2∅ charge transfer.
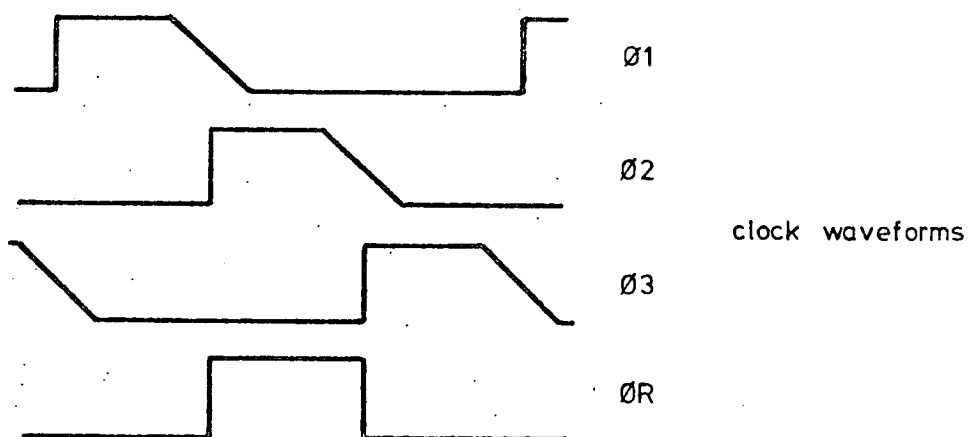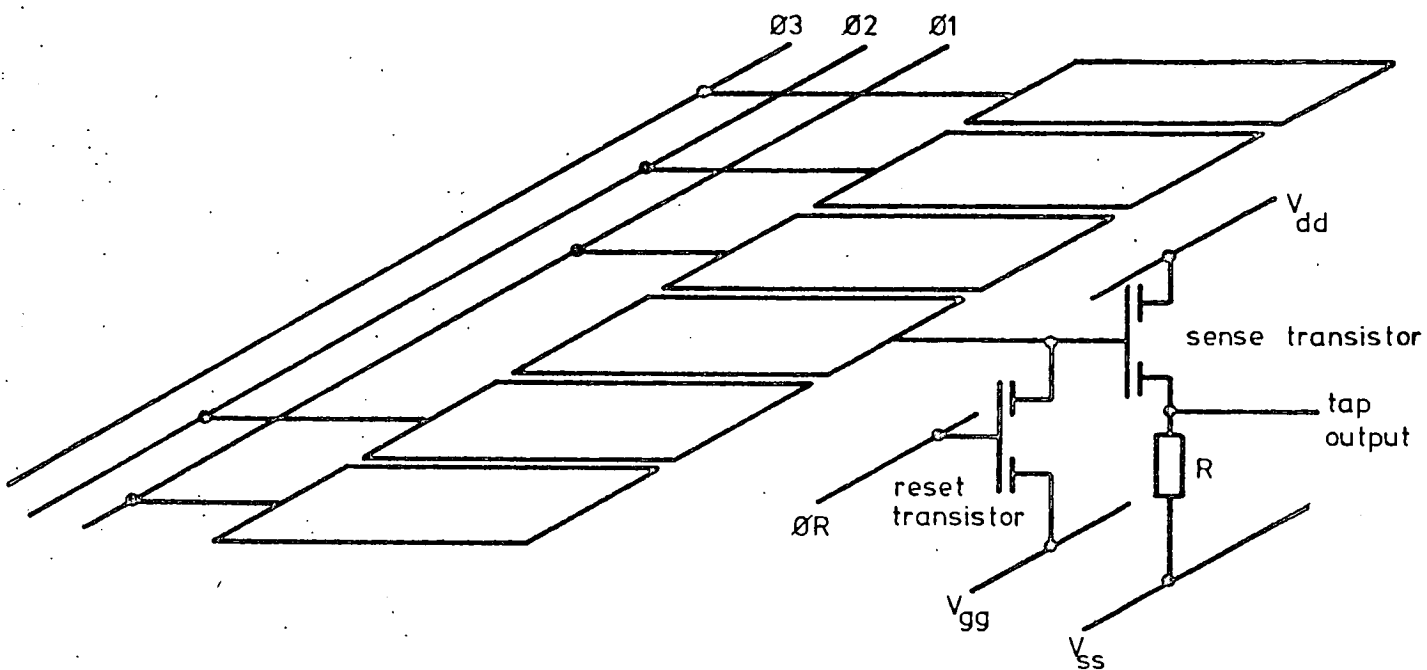
Figure 2.10: Floating-gate-reset charge sensing.

its potential is modified by the signal charge packet transferred under it. This modified electrode potential is sensed by the source follower shown in figure 2.10 and a voltage proportional to the signal size may be measured at the sense amplifier output.

The main disadvantage with this sensing technique is that the dynamic range of signals is halved since the shallow potential well under the sense electrode now determines the maximum signal size. The main characteristics of the two TDLs described in this section are summarised in table I.

2.4 SPS delay line implementation

The serial-parallel-serial[36] configuration of delay line is the one most commonly used to implement long delays, usually associated with the storage of digital information. If a normal linear CCD delay were to be used in this application P delay stages would be required to store P bits of data with a resultant inefficiency product of $P\xi$. With $P\simeq10^3$ this inefficiency product becomes unacceptable and some other technique for delaying signals must be found.

The SPS realisation is shown diagramatically in figure 2.11. This consists of a set of N parallel delay lines each of lenght M delay stages. The inputs to these delay

	M55	WM2010
delay stages	32	32
clock phases	3	2
maximum sampling frequency	100 kHz	2 MHz
delay cells/tap	2	1
dynamic range	50 dB	50 dB

Table I: Tapped delay lines characteristics.
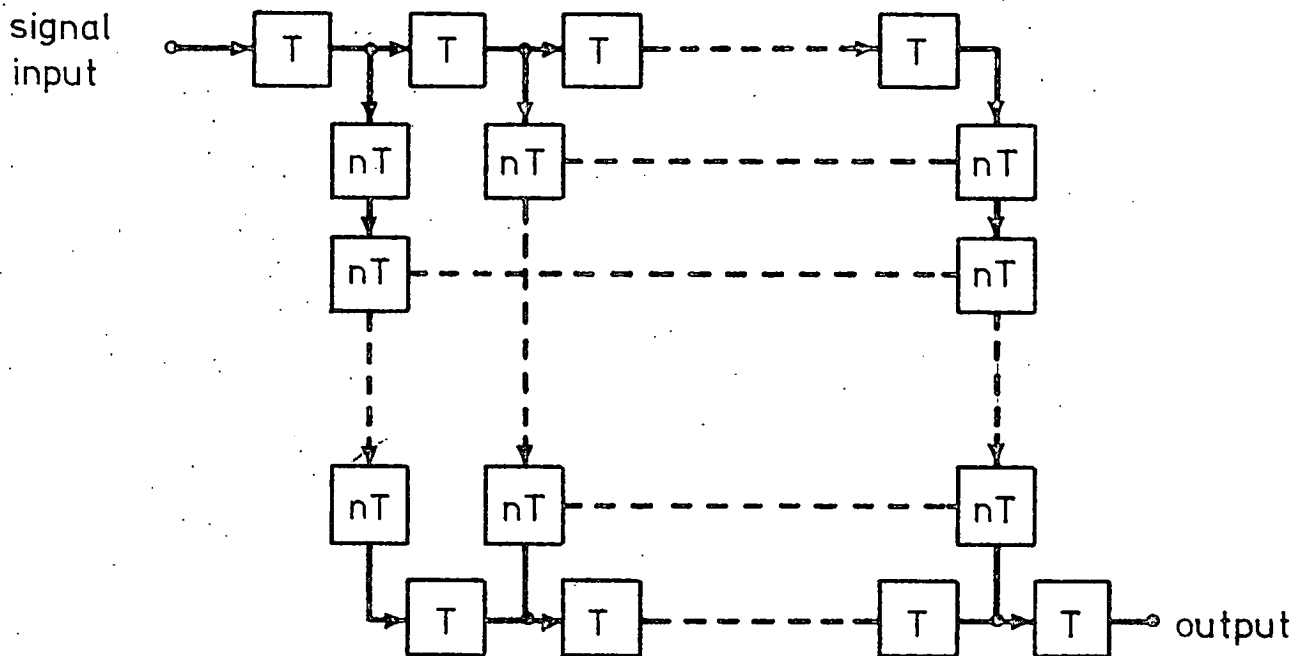
signal
input

Figure 2.11: General configuration of an SPS delay line.

lines are supplied from N parallel outputs on an N stage delay line which has a clock rate N times faster than the parallel delay line clocks. Therefore a set of N input signals are clocked into the first serial delay line at clock rate f_c and are then transferred into the N parallel delay lines. The signals are clocked along these parallel lines at a clock rate of f_c/N. At the end of these delay lines the signals are transferred into another N point serial delay line and clocked out at clock rate f_c. The total delay for any given signal is then N.M stages but total number of transfers is only N + M.

Although signal distortion due to transfer inefficiency is reduced to a minimum using this architecture the effects of dark current noise are accentuated. Dark current[37] noise arises from thermally generated minority carriers which are injected into the signal charge packet. At the output of the CCD the contribution from dark current is indistinguishable from the actual signal charge.

In a linear CCD this noise source does not usually present a serious problem provided that the device clocks operate at a constant rate. This is because each signal packet will accumulate the same amount of dark current generated charge during its passage along the delay line resulting in only an extraneous d.c. bias at the output. This, however, is not the case in an SPS device since a signal charge packet may be transferred along any one of N
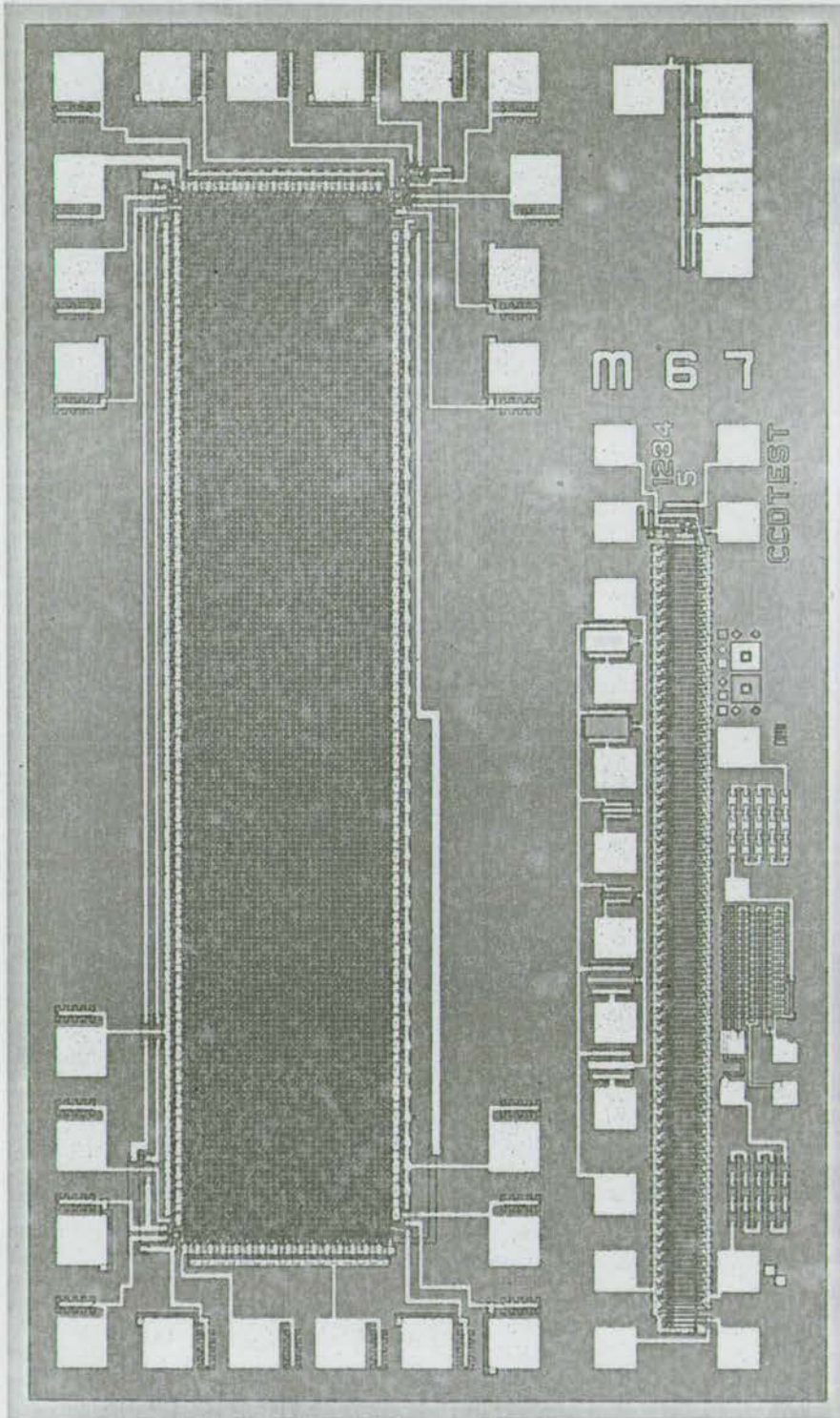
Figure 2.12: Photomicrograph of the Plessey M67 SPS
delay line.

different delay paths. Differences in device structure such as local stacking faults and carrier concentration which tend to vary across the device surface mean that each delay path will have a different dark current contribution. Therefore at the SPS output the signal appears with an additive noise component which repeats every n clock cycles.

In chapter 3 a scheme for the cancellation of this fixed pattern noise is presented[29]. The SPS delay line used was a Plessey M67 SPS delay line which is a 4028 stage delay arranged as a 32 x 128 stage SPS block with a specified maximum clock rate of 2 MHz. A photograph of the M67 integrated circuit is shown in figure 2.12.

2.5 Programmable transversal filter implementation

Two monolithic CCD programmable transversal filter realisations were used in the work described in chapters 5 to 8. These were the WM2100 64-point PTF[14] and the WM2110 256-point PTF[31]. In this section the basic operational characteristics of each of these devices are described with particular attention being paid to the method of weight multiplication.

Both of the PTF circuits mentioned above were of the TDL type described in section 1.1 with analogue weight storage where the weighting operation is achieved using a

set of parallel analogue multipliers. In the case of the WM2100 device the TDL structure was essentially a 64-point 3∅ structure of the same type as that used in the M55 TDL. A block diagram of the 64-point PTF is shown in figure 2.13. Storage of the 64 weight coefficients was achieved using a set of 64 storage capacitors linked by MOS switches to a common analogue input bus. Any one of these switches could be opened, at random, by the application of a 6 bit address code to the input function decoder. However, once read into the sampling sites it was impossible to retrieve the information again due to the fact that the capacitance of the input bus line was approximately 100 times greater than the storage capacitance. It is therefore necessary to store the filter weight values on some external medium, usually digital RAM or ROM.

The signal weighting was achieved using a single MOS multiplying transistor at each CCD tap output, connected as shown in figure 2.14. The signal at the CCD tap output comprises the signal v_x plus a constant bias voltage V_x and the weighting signal, or reference, applied to the transistor drain was comprised of the weighting signal v_y plus a constant bias V_y. The source of each transistor provides an output current I_{DS} where:

$$I_{DS} \propto v_x v_y + (V_x - V_y - V_T)v_y - \tfrac{1}{2}v_y^2 \quad \ldots\ldots\ldots\ldots\ldots(2.1)$$

$$= v_x v_y + (Kv_y - \tfrac{1}{2}v_y^2) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2.2)$$
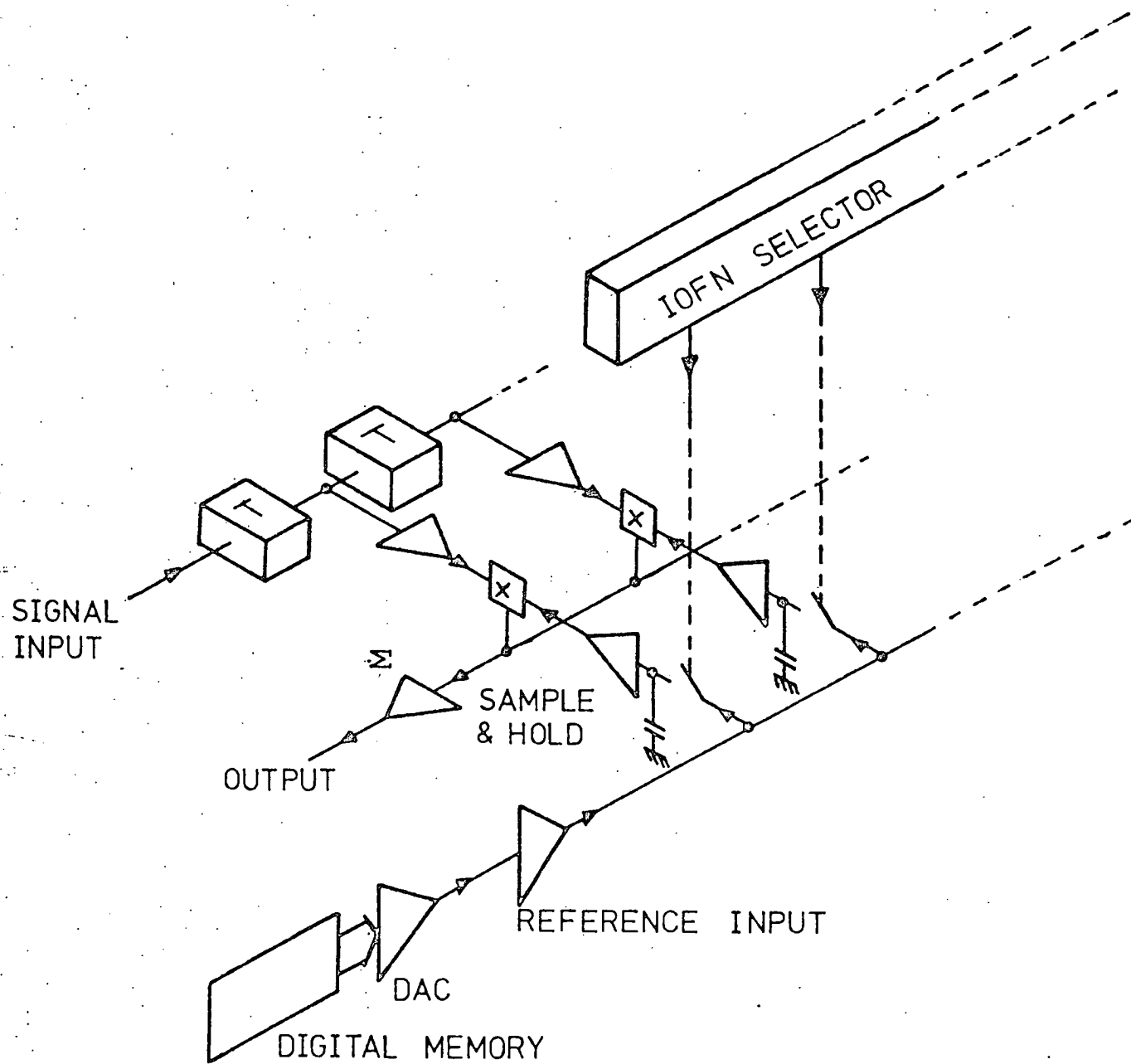
Figure 2.13: Block diagram of the 64-point programmable
transversal filter.

(from CCD tap output)

$v_x + V_x$

virtual V_y

I_{ds}

$v_y + V_y$

(from weight
sample hold)
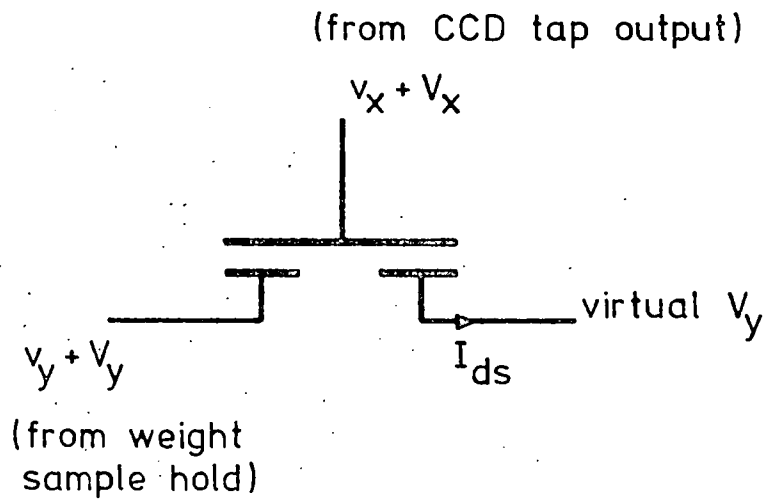
Figure 2.14: Schematic diagram of the single MOST
weighting multiplier.
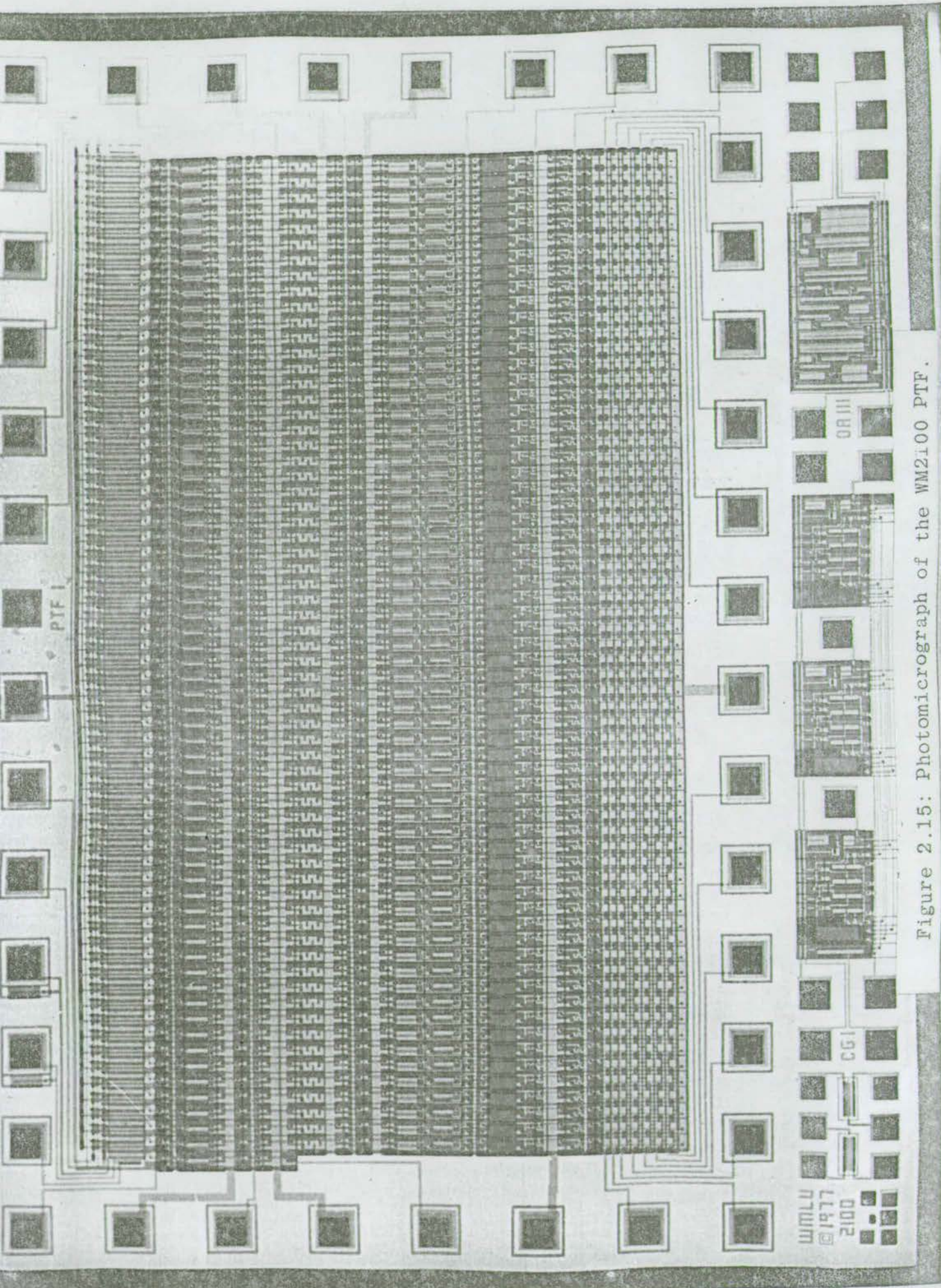
Figure 2.15: Photomicrograph of the WM2100 PTF.

For any given set of reference weights v_y does not change and the bracketed term in equation (2.2) can be viewed as a constant d.c. offset at the output. All I_{DS} currents are summed by connecting the transistor sources to a common summing bus to yield the result

$$v_o = \sum v_x v_y + \sum (kv_y - \tfrac{1}{2}v_y^2) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.3)$$

One technique which was used to subtract the unwanted d.c. term in equation (2.2) was to form the convolution sum described in equation (2.3) and store this result for half a clock cycle on a sample and hold circuit at the output of the device. On one of the following CCD reset clock cycles the CCD sense gate was forced to a constant voltage equal to V_x. The filter output was sampled again at this point yielding the result:

$$v_o = \sum (kv_y - \tfrac{1}{2}v_y^2) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.4)$$

Subtracting equation (2.4) from equation (2.3) yields the required output $\sum v_x v_y$.

It was possible to use this zeroing technique with the WM2100 PTF since the CCD tap amplifiers were coupled directly to the multiplier transistor gates. However, with the WM2110 PTF this was not possible since each CCD tap output was sampled at the tap amplifier output to allow a larger amount of time for output amplifier

settling. The WM2110 device was based on a 256-point 2∅ TDL of the same type as the WM2010 TDL. It differed in a few significant ways from the WM2100 PTF. A block diagram of the 256-point PTF chip is shown in figure 2.16. The weighting sample and holds were implemented in the same way as the WM2100 but here the sampling switches were activated by a pulse propagating along a 256 stage shift register rather than by the random access technique used in the WM2100.

As mentioned before the CCD tap outputs are all sampled on the chip and the zeroing technique described above could not therefore be used. The mechanism by which the residual d.c. errors caused by this are compensated for in an adaptive system is described in section 6.4. The reason for not including the facility for zeroing in this chip was that in many filtering applications d.c. terms on the PTF output are not used and the excess d.c. offset shown in equation (2.3) may be nulled simply by a.c. coupling the filter output. However, in the implementation of adaptive systems d.c. terms on the output must be used and a.c. coupling would result in a failure to converge.

A comparison of the characteristics of the WM2100 and the WM2110 PTF devices is given in table II.

	WM2100	WM2110
no. of filter points	64	256
tap weight update mode	random	serial
CCD clocking	3∅	2∅
delay stages/tap	2	1
Max. CCD sampling rate	100 kHz	2 MHz
Max. weight update	1 MHz	5 MHz
Harmonic distrotion	-40 dB	-50 dB
output dynamic range	65 dB	70 dB

Table II: Programmable transversal filter
characteristics.
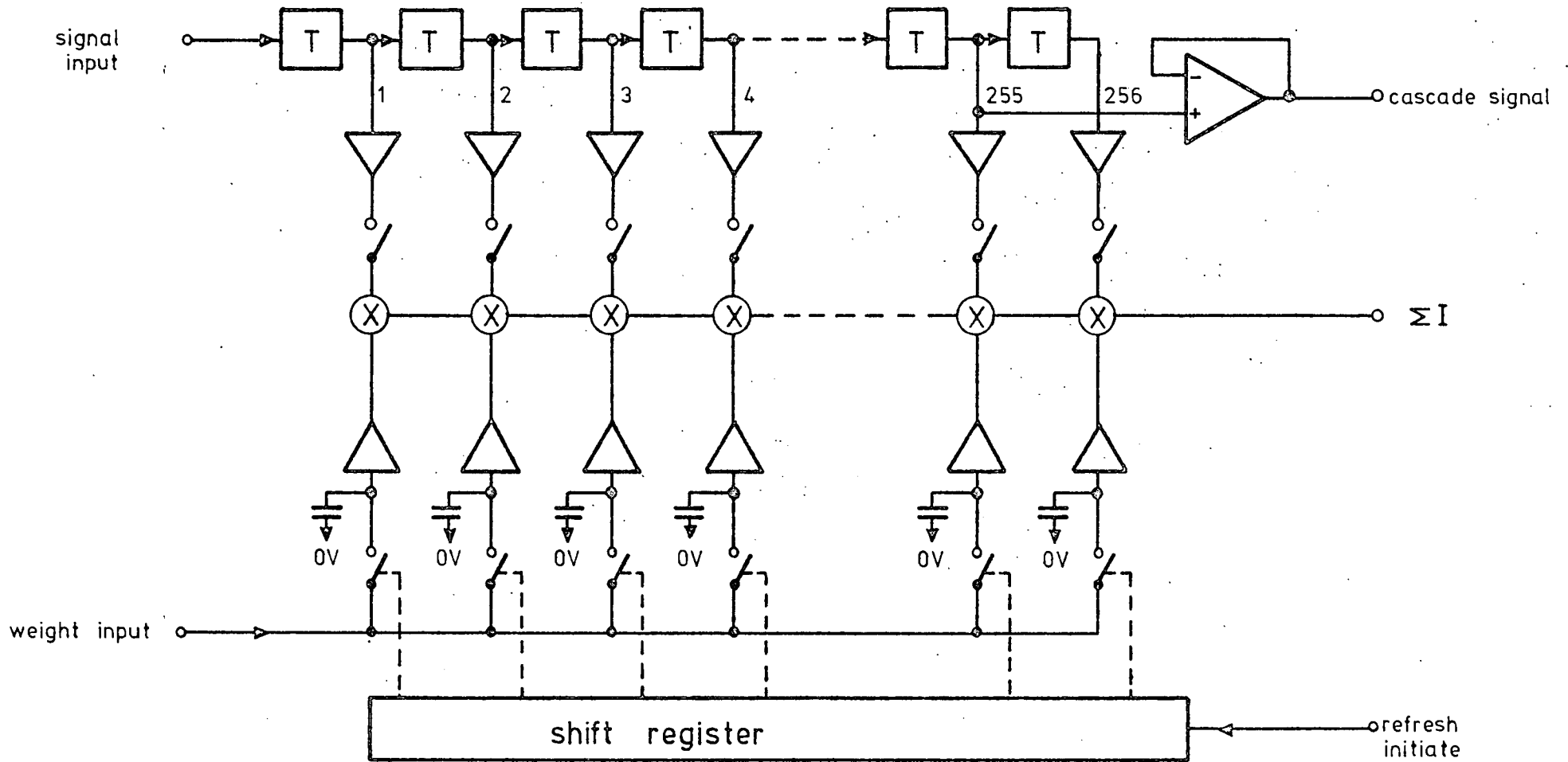
Figure 2.16: Block diagram of the 256-point PTF.
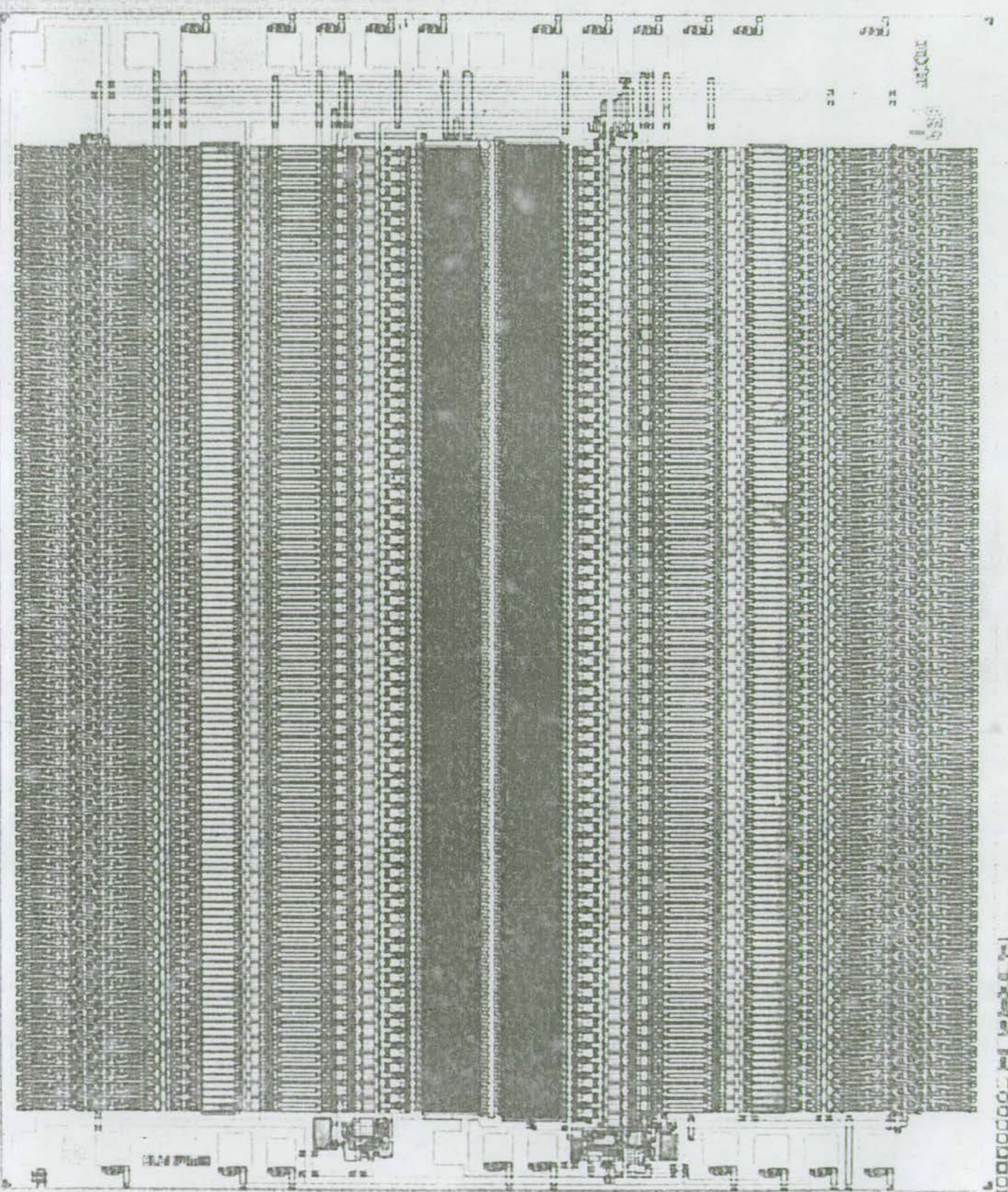
Figure 2.17: Photomicrograph of the WM2110 PTF.

CHAPTER 3:   OPEN LOOP SYSTEMS

In this chapter two systems are considered under the broad definition of open loop signal processing systems. This means that the processor has a fixed, and therefore predictable, response to any input signal and a certain number of a-priori assumptions must be made about the form of incoming signals. The two systems considered are:-

(1) A statistical analyser module which is capable of detecting signal peaks immersed in noise by the use of an adaptive thresholding technique.

(2) A noise canceller used to eliminate the fixed pattern noise associated with the output of a CCD serial-parallel-serial delay line; the delay line used was described in chapter 2.

The systems described above are adaptive in that they are capable of adjusting their characteristics to suit any situation in a given set of possibilities. However the filtering functions used are fixed, ie. no feedback to filter tap weights is provided, and therefore the systems are described as open loop.

3.1 Linear Regression Analyser Module

A common problem encountered in signal processing is that of detecting a signal against a background of noise. If the expected signal waveform is known then matched

filtering techniques may be used, and if the expected signal is a precisely known pattern (eg. a PN-sequence) then the optimum detection technique is correlation. However, if relatively little information is available about the expected signal (that is, it is restricted within a certain frequency band but may assume any combination of frequency components within that bandwidth) then the problem becomes more complex. For instance, for the events in the time domain using conventional filtering and thresholding techniques the threshold must be matched to the noise level which can vary making the set threshold invalid, and a slowly varying baseline also adds to this problem.

The ideal system would be one which is capable of evaluating the mean-square noise level and using this figure to automatically adjust the thresholding level to attain a constant error rate[39]. A slowly varying baseline would still lead to detection problems but this could be overcome by evaluating the signal validity from its derivative rather than its amplitude, in which case d.c. and low-frequency noise would have no effect on the detection system. A block diagram of this basic, idealised system is shown in figure 3.1.

This type of signal analysis is ideally suited to the use of statistical techniques and the system presented in this section is based on linear regression analysis[40].
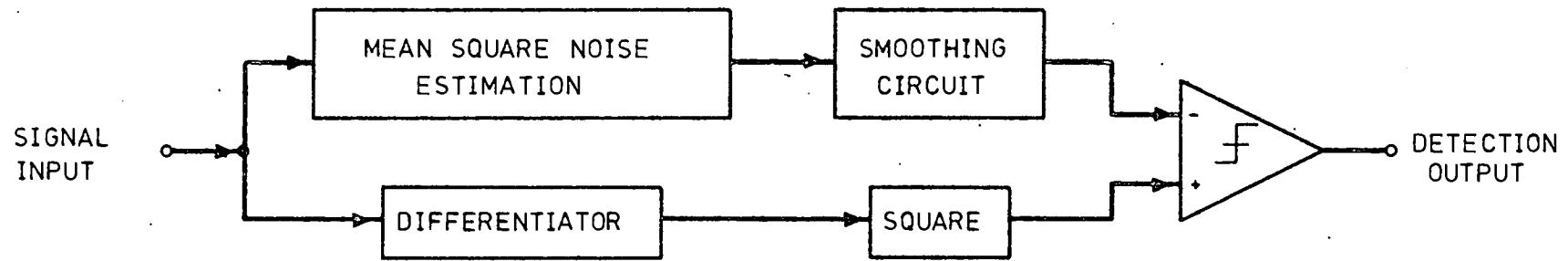
Figure 3.1: Block diagram of an idealised adaptive threshold
detection system.

NB. It should be noted that where the signal is
known exactly, beforehand, matched filtering is
still the optimum filtering technique. This type
of filter is really only valid where a pulse of
unknown form occurs within specified limits.

This technique[41] has been used before in the form of a computer program for the 'non real-time' analysis of experimental data. The system described here, based on analogue charge-coupled device (CCD) tapped delay lines, implements similar algorithms providing a 'real-time' output at the sampling rate[39]. Certain restrictions on bandwidth and accuracy do exist using this method and these are described in greater detail in section 3.1.2, devoted to systems design.

3.1.1 Theoretical Analysis

A slowly varying signal, y, having values y_n at times t_n may be estimated over a given interval by the equation, $y = mt_n + c$; where m and c are the derivative and average of the signal, respectively. The error in the value of y, at any given time, may be expressed as:

$$e_n = y_n - y$$
$$= y_n - (mn\tau + c)$$

where the samples are evenly spaced in time by intervals τ. The sum of the squared errors over a fixed number of samples $(2p + 1)$ is then given by:

$$S^2 = \sum_{n=-p}^{p} e_n^2 = \sum (y_n - (mn\tau + c))^2 \quad \ldots\ldots(3.1)$$

Using the least-squares method, the best estimates of the quantities m and c are given by minimising the sum of the errors squared e_n^2. That is:

$$\frac{\partial S^2}{\partial m} = \frac{\partial S^2}{\partial c} = 0$$

Hence:

$$m = \frac{\sum\limits_{n=-p}^{p} n\tau y_n}{\sum\limits_{n=-p}^{p} (n\tau)^2} = y' \quad \dots\dots\dots\dots(3.2)$$

$$c = \frac{\sum\limits_{n=-p}^{p} y_n}{2p + 1} = \overline{y} \quad \dots\dots\dots\dots\dots(3.3)$$

The minimised form of equation (3.1) is then:

$$S^2 = \sum y_n^2 - m^2 \sum (n\tau)^2 - c^2(2p + 1) \quad \dots\dots\dots(3.4)$$

where m and c are given by equations (3.2) and (3.3).

Assuming the data to be normal with standard deviation σ,

then σ^2 may be estimated by:

$$\sigma^2 = \frac{\sum(y_n - y)^2}{2p}$$

and the variances of m and c are given by:

$$\sigma_m^2 = \frac{\sum(n\tau)^2 \sigma^2}{(\sum(n\tau)^2)^2} = \frac{\sigma^2}{\sum(n\tau)^2}$$

$$\sigma_c^2 = \frac{\sum \sigma^2}{(2p + 1)^2} = \frac{\sigma^2}{2p + 1}$$

Hence:

$$\sigma_c^2 = \frac{p + 1}{2} \left\{ \frac{1}{p(p + 1)} \left[(\overline{y^2}) - c^2 \right] - \frac{\tau^2 m^2}{3} \right\} \dots(3.5)$$

$$\sigma_m^2 = \frac{3}{2p} \left\{ \frac{1}{p(p+1)\tau^2} \left[(\overline{y^2}) - c^2 \right] - \frac{m^2}{3} \right\} \quad \ldots \ldots (3.6)$$

where
$$\overline{y^2} = \frac{1}{2p+1} \sum_{n=-p}^{p} y_n^2 .$$

That is, the variances of m and c are connected simply by a factor of proportionality, $\frac{1}{3}p(p+1)\tau^2$.

Intuitively, it may be argued that the value of m^2 must be greater than its probable error σ_m^2 to register a significant rate of change in the data, and hence the presence of a signal (see figure 3.1). More rigorously, the significance of the regression coefficient m may be determined by the application of the 'Student-t' test of significance to the derivative. That is:

$$t = m \left[(2p - 1) \sum (n\tau)^2 / \sum (y_n - y)^2 \right]^{\frac{1}{2}}$$

or
$$\sum (y_n - y)^2 = \frac{m^2(2p - 1) \sum (n\tau)^2}{t^2}$$

where the value of t may be found from the Student-t table,[40] dependent on the chosen level of significance. Using equation (3.4) an inequality:

$$T(m,\beta) = \frac{1}{p(p+1)^2} \left[(\overline{y^2}) - c^2 \right] - (\beta + 1) \frac{m^2}{3} < 0 \quad (3.7)$$

may be determined which indicates when a signal is significant at the chosen level. Equation (3.7) is similar to (3.6) except for the factor β where:

$$\beta = \frac{2p - 1}{t^2}$$

This test is independent of the value of τ, which may be arbitrarily set at unity. We now have a test which yields a signal detection system which automatically compensates for the noise level, the function which can not be carried out using conventional filtering and thresholding techniques.

A number of output parameters may be readily evaluated from the well known results[39]. For instance, it can be predicted that the averager and differentiator frequency responses will closely follow the sinc(x) function and its derivative. Further, the averager response will have a maximum at d.c. and its first null when one complete cycle is stored in the CCD register (ie. at $1/27f_c = 0.037 \ f_c$, where f_c is the sampling rate of the 27 point delay lines used in this experiment).

It is also a simple matter to calculate the maximum processing gain (minimum signal to noise ratio) for the system by the choice of two distinct waveforms representing maximum signal and maximum noise respectively. Furthermore, if the noise signal is chosen so that it affects only the y^2 value and the signal has no effect on the variance then the calculation is greatly simplified. Examination of the relevant variance and significance formulae confirms that these criteria are satisfied for the case where the noise signal is a Nyquist-rate square wave and the signal is another square wave at less than $0.018 \ f_c$ (ie. well within the range where the variance

output is not affected). The processing gain may then be evaluated for various levels of significance. Figure 3.2 shows a plot of minimum signal to noise ratio against significance level for the stated case. It should be noted that the processing gain is critically dependent on the signal and noise spectral characteristics and the case presented here is for maximum processing gain.

In all instances shown in figure 3.2, where the minimum signal to noise ratio is less than 0 dB, a white noise background may not be used since components of the noise spectrum in the signal detection region will yield a significant output. Therefore, for a white noise background it is necessary to use a value of β which gives a minimum signal to noise ratio of 0 dB, or greater, at the input. In the example stated in section 3.1.2 the significance level used was the 1% level and hence the noise spectrum was limited to about 0.04 f_c, upward.

In the case where a white noise background may be used the components of noise in the signal region both add and subtract from the actual signal. Therefore, to ensure signal detection the minimum signal to noise ratio is raised to about 3 dB.

3.1.2 System Implementation

The implementation of the system described in the
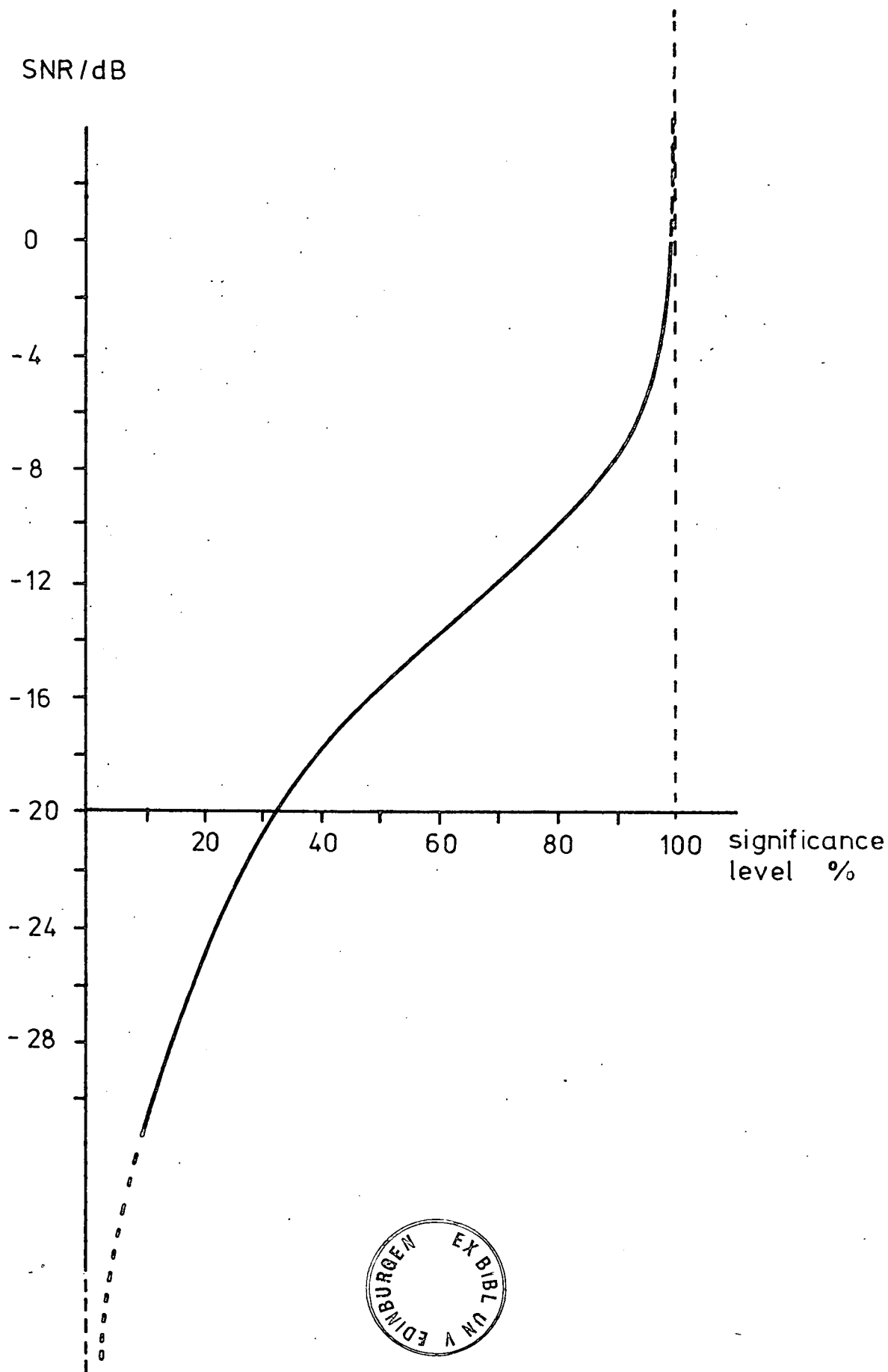
Figure 3.2: Graph of minimum signal to noise ratio plotted
against theoretical significance level.

previous section is dependent on the ability to store

several successive samples of the incoming data, and make

them available for manipulation simultaneously, to aid

the speed of operation. In addition, the accuracy of the

system is dependent on the number of samples used, and

the bandwidth of detectable signals is dependent on the

sampling frequency and the number of samples N, such that:

Bandwidth $\alpha f_c/N$

Accuracy $\alpha \sqrt{N}$

Therefore to make a fast, versatile system with

potentially programmable bandwidth the following

attributes are required:

(i)   Variable sampling rate;

(ii) The availability of several successive

samples simultaneously (ie. serial-to-

parallel operation); and

(iii) The ability to process varying numbers of

samples.

One circuit which offers all of these features is the CCD

analogue tapped delay line, which was the basic circuit

element used to implement the experimental system.

The delay lines used in the experimental system to be

described here were the M55 CCD tapped delay lines

described in chapter 2, with 27 separately accessible taps.

The use of these devices means that the averaging and

differentiating function described mathematically in

equations (3.2) and (3.3) may be implemented directly by weighting the tap outputs and summing. Circuit diagrams demonstrating the circuits used to implement these functions are shown in figure 3.3. The register values R_1 to R_{27} may be evaluated directly from equations (3.2) and (3.3). In figure 3.3(a) the averager weighting resistors are all identical and the differentiator weights in figure 3.3(b) form a linear ramp with R_{14} being the centre point, not contributing.

A complete block diagram of the experimental system is shown in figure 3.4. It can be seen from this that the system is a literal interpretation of the variance and significance formulae, making system performance exactly predictable. However, the variance output is filtered before application to the significance testing circuit to eliminate variance errors. This means that the variance is averaged over a greater number of data points than the basic 27 and the significance testing circuit now more closely resembles a 'Chi-square' test than a 'Student-t' test of significance. Nevertheless, equation (3.7) may still be applied and in this case for 27 data points at the 1% significance level the value of β is approximately 4.2.

Although the significance output gives a positive indication of a valid signal at the input, it conveys only the information that a significant derivative is
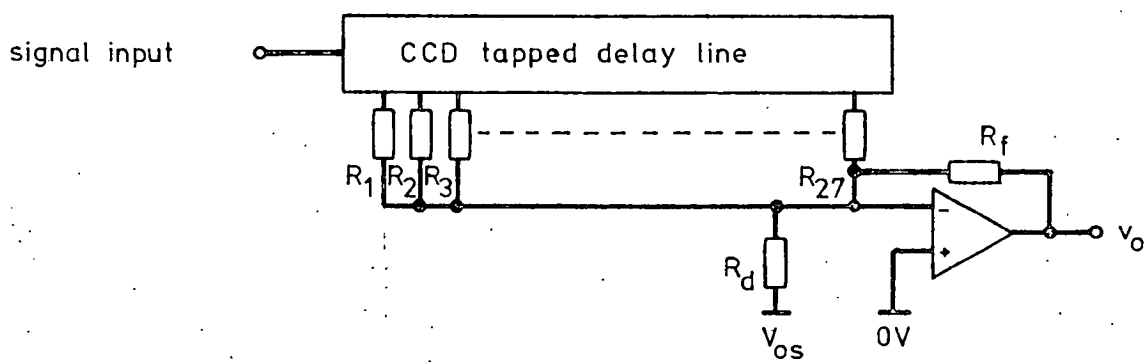
(a) Averager circuit.
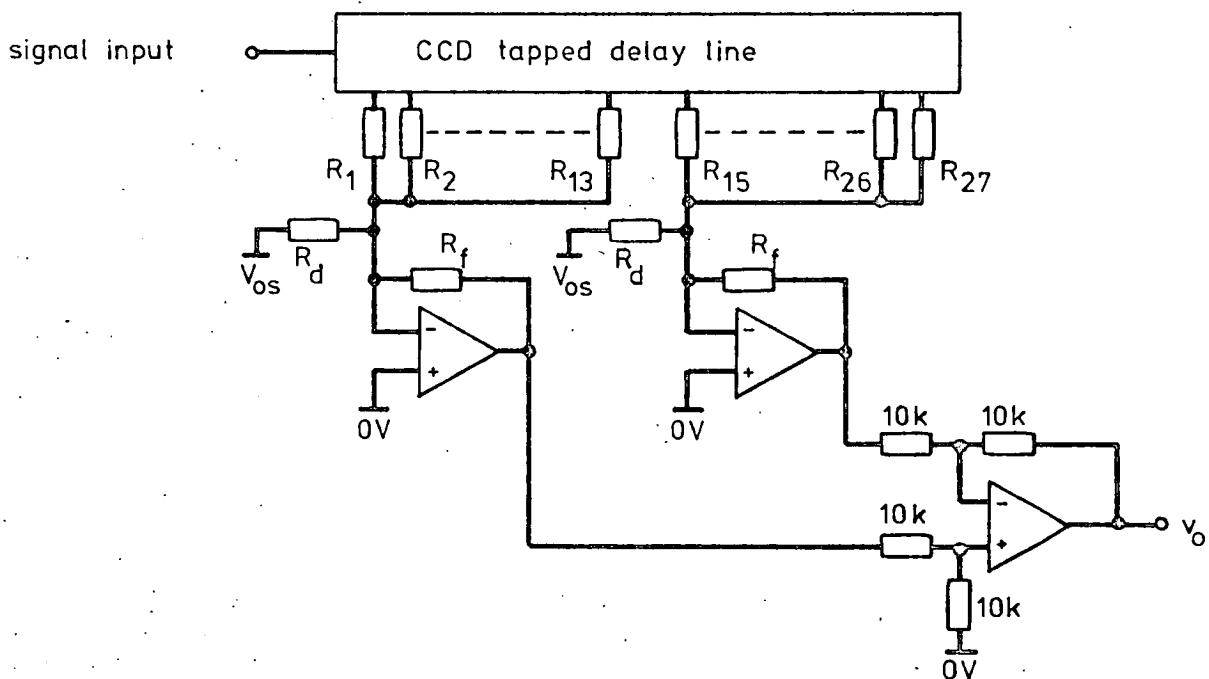
(b) Differentiator circuit.

Figure 3.3: Schematic diagrams of the CCD averager and differentiator circuits.
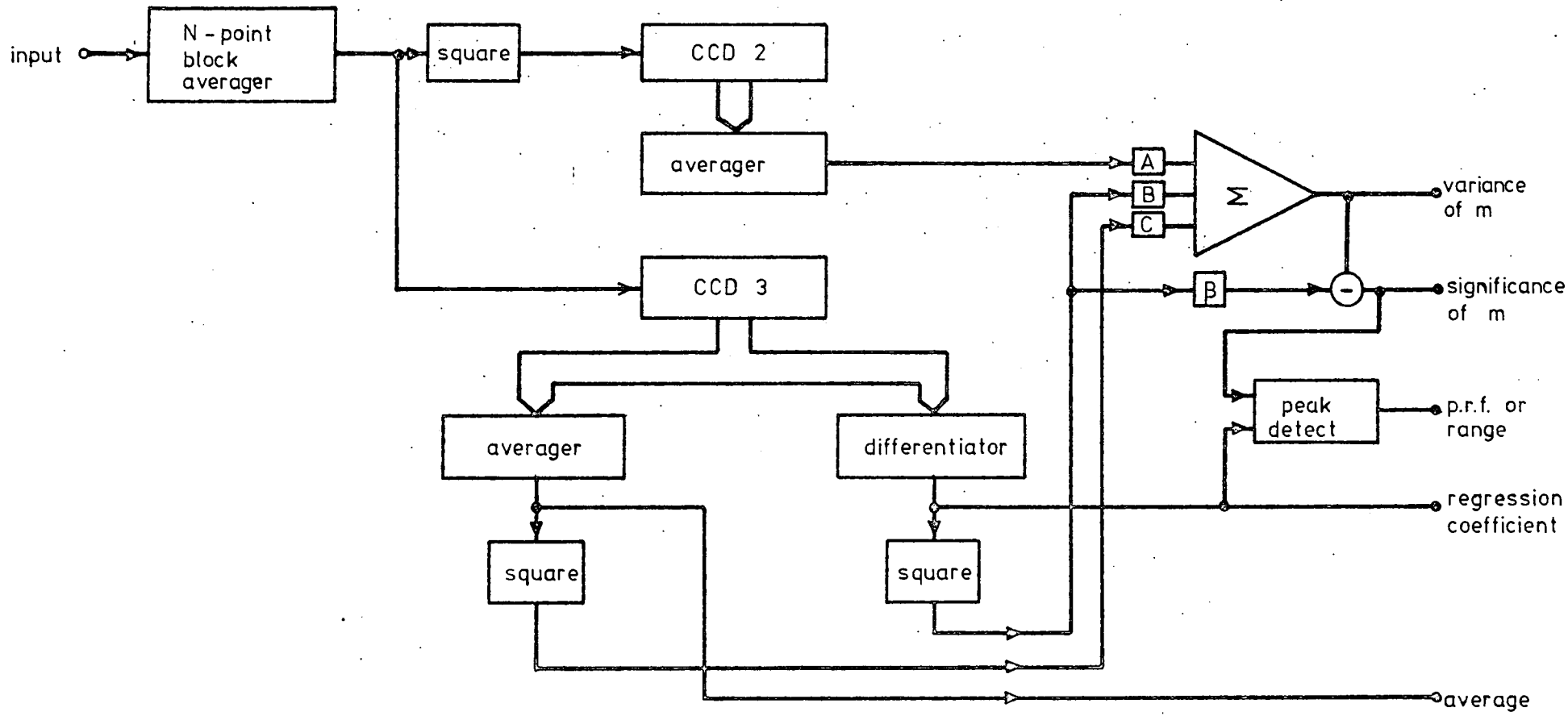
Figure 3.4: Block diagram of the experimental linear regression analyser.

present, whilst no information on the polarity of that derivative is available. If however, the significance output is used in conjunction with the derivative output then significant positive and negative derivatives may be separately resolved. This information may then be applied to indicate the presence of a significant signal peak.

Information derived from this circuit may further be used to find the peak position[41] and the peak repetition period. In the experimental system the peak position was taken as the point half-way between the last significant positive derivative and the first significant negative derivative. The peak repetition period may then be calculated by counting between these values. It is possible to initiate this count sequence from any arbitrary point so if the peak position with reference to an arbitrary synch. pulse were required (ie. a range figure) then the algorithm could be easily modified to provide this facility.

The experimental system was constructed as a modular system in a 19" rack: figure 3.5 shows a photograph of the system. The complete system as described had overall power requirement of 12 Watts, however, the CCD processing elements individually consume 500 mW. It is estimated that an equivalent digital realisation would require 50 times as much power.(i.e. in the region of 500W).
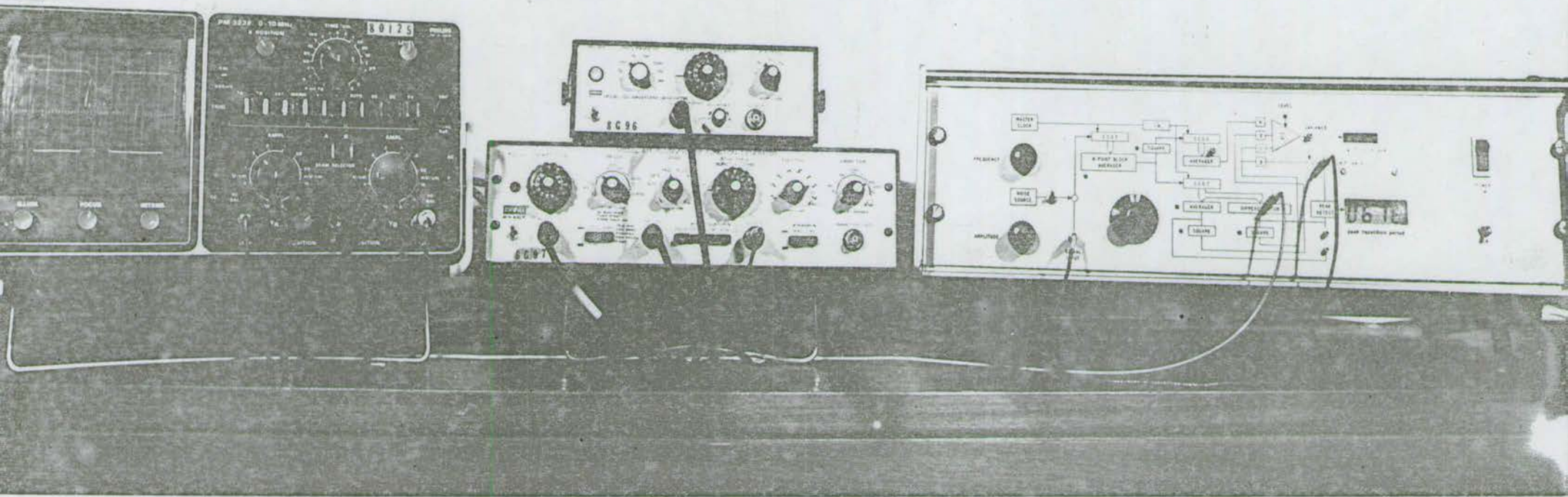
Figure 3.5: Photograph of the experimental system.

3.1.3 Experimental Results

Figure 3.6 shows a series of photographs demonstrating the basic system outputs for a triangular wave input. It should be noted that the differentiator weighting does not make full use of the possible system bandwidth but is rather a least-square estimate.

Since the variance is (by definition) the mean-square variation of the signal, then a graph of the square root of the variance plotted against the rms noise input should yield a linear relationship. This was plotted using a 31-bit pseudo-random sequence as the noise source and the result is shown in figure 3.7. It can be seen that, within the limits of reading error, a linear relationship was maintained.

Figure 3.8 shows a plot of averager and differentiator frequency responses for a sine wave swept from d.c. to 0.06 f_c, approximately. These responses display the expected sinc(x) shape and its derivative (cf. section 3.1.1).

Also shown in figure 3.8 is the envelope of the unfiltered variance output, for the same input, demonstrating a relatively constant output above about 0.04 f_c. Hence, frequencies above this point are detected as noise yielding a predictable d.c. offset at the
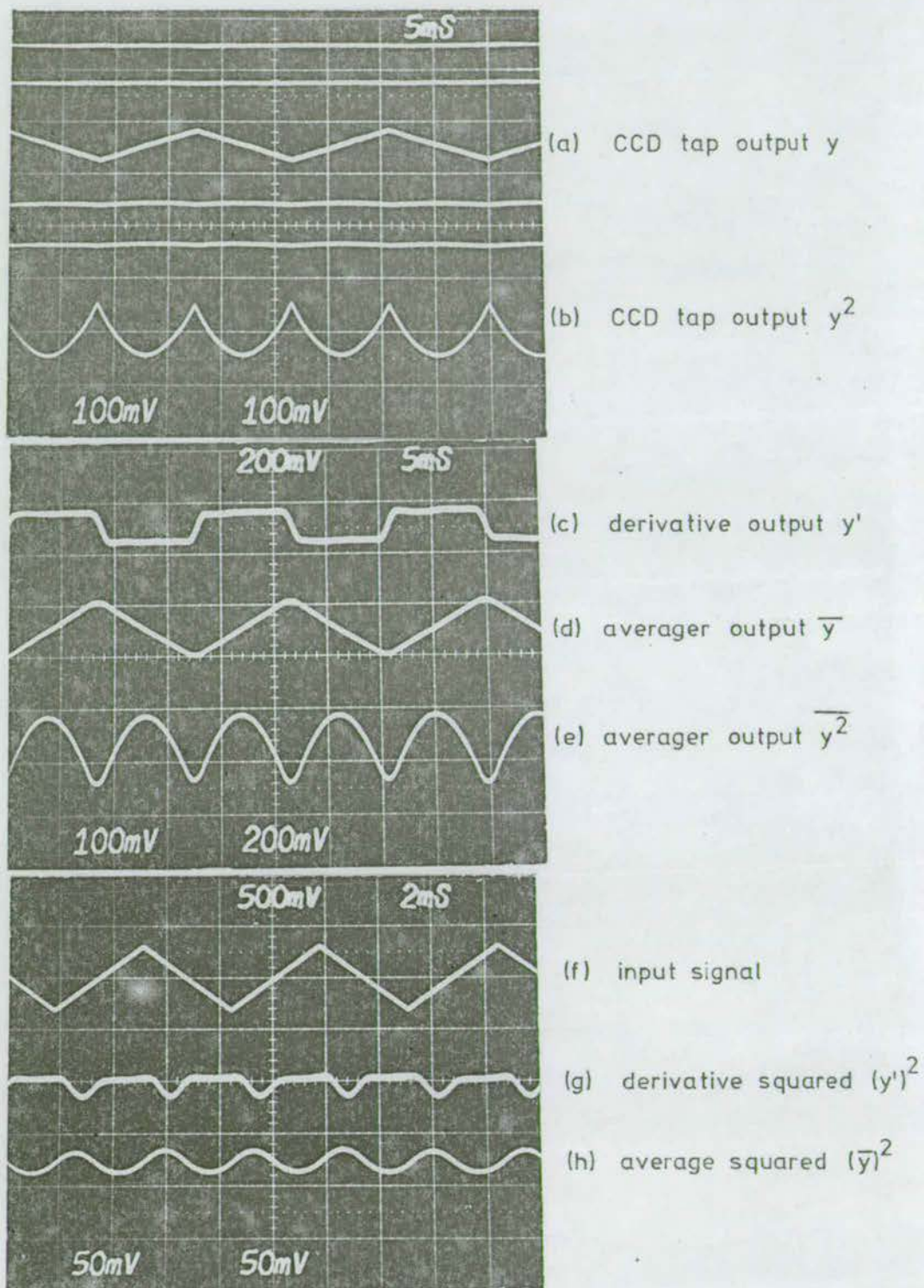
Figure 3.6: Basic system outputs for a triangular waveform input.
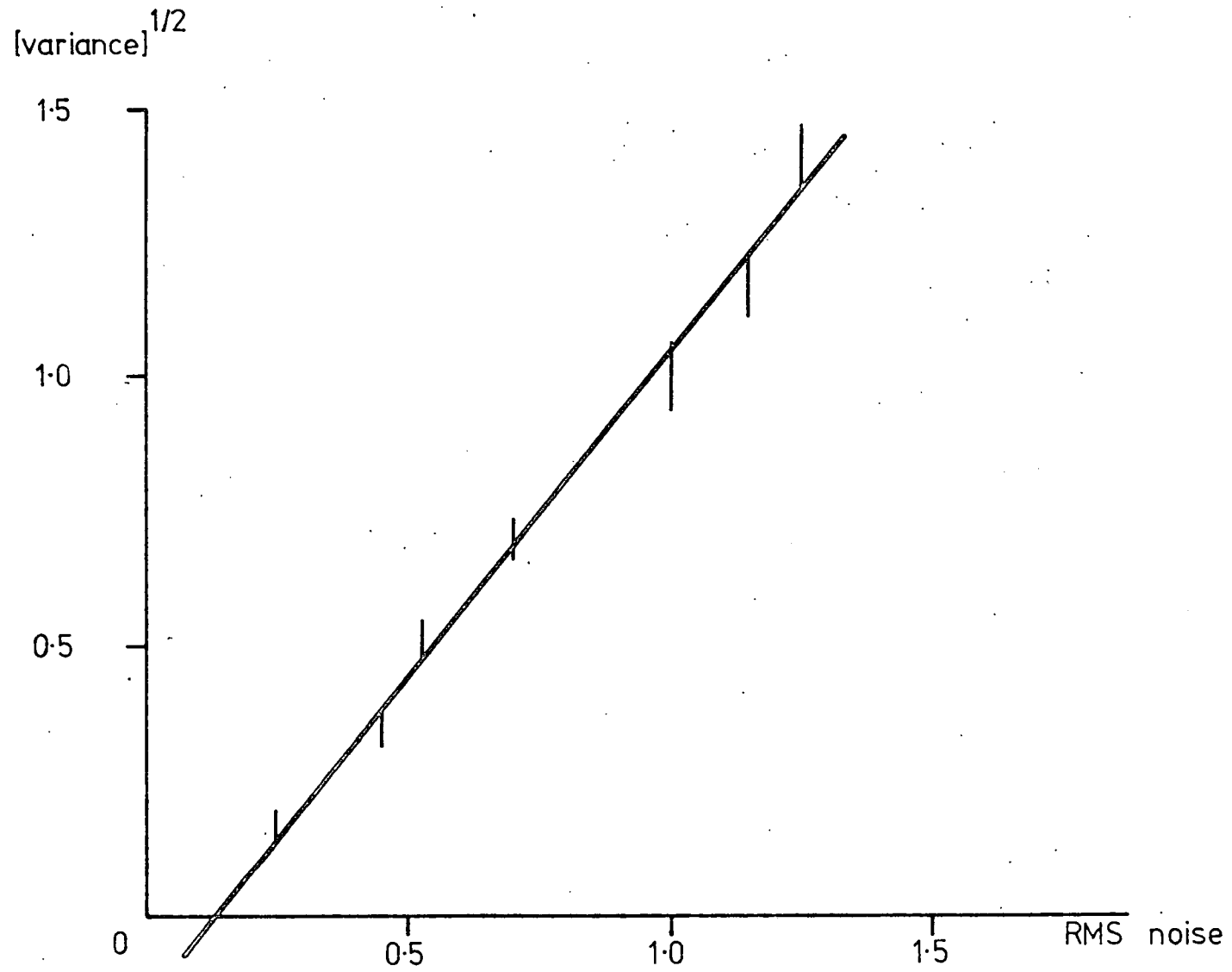
Figure 3.7: Plot of the square root of the variance output
against input signal to noise ratio.
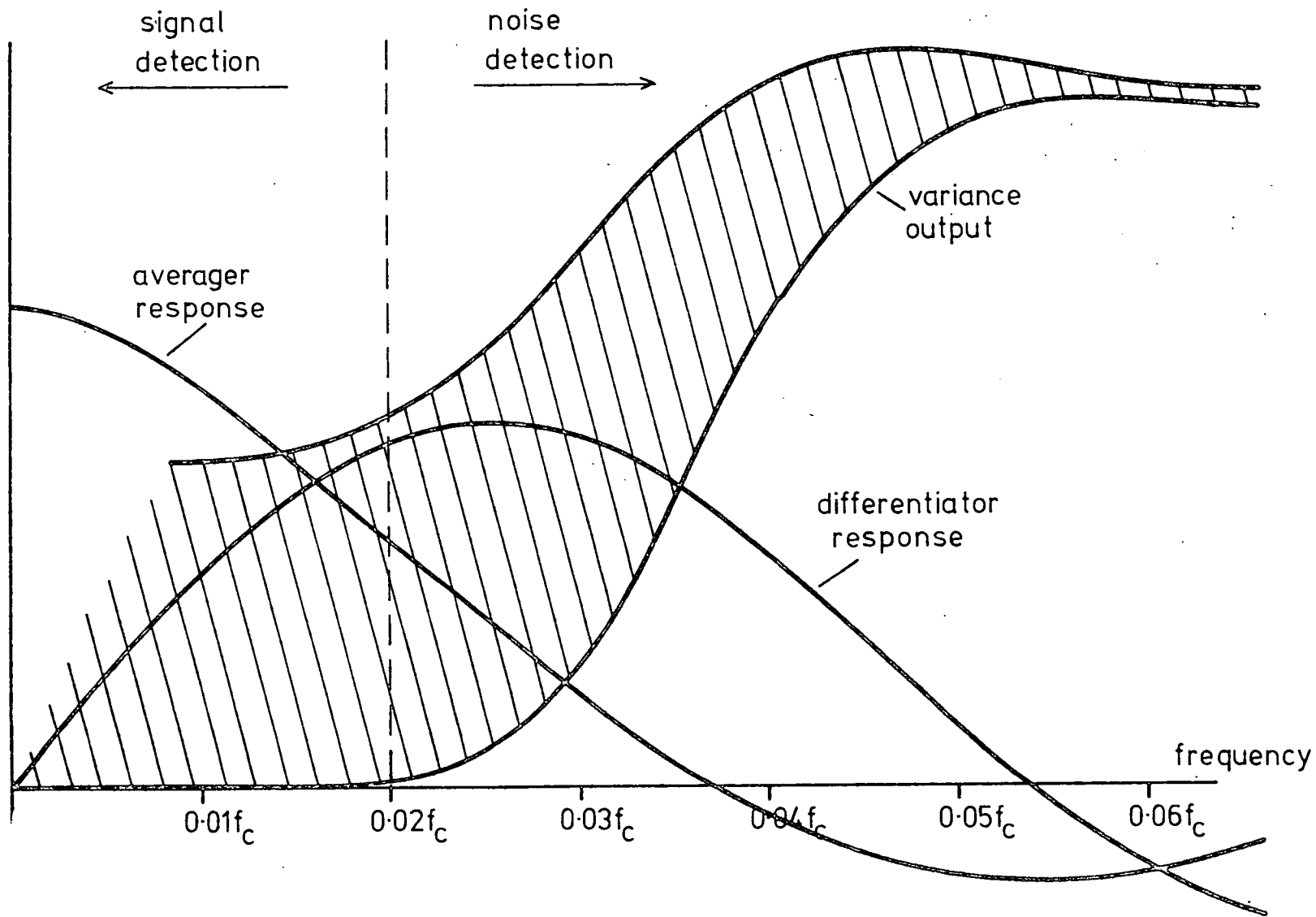
Figure 3.8: Averager, differentiator and variance output
amplitudes for a sinusoidal input swept from
d.c. to 0.06 f_c.

variance output and signals below this frequency are regarded as valid signals (weighted according to their derivative).

Figure 3.9 shows some typical significance outputs for a triangular pulse, first with no noise and then with an estimated 0 dB signal to noise ratio. In this case the noise source used was a 127-bit pseudo-random sequence filtered to give about 10 dB rejection at 0.04 f_c. Therefore noise components are present in the signal region and these may be clearly seen in figure 3.9 although they never reach the detection threshold level. It can be seen from these outputs that even with a considerable noise background the significant peaks still reach above the set threshold level and a consistent peak detection output is available.

A major brawback in the system described here was a low dynamic range which was measured to be about 25 dB at the input. This is due mainly to the fact that the signal is squared before entering CCD2 which results in halving the available input dynamic range (the range of the CCD register is approximately 50 dB)[42]. That is, for normal operation the dynamic range;

$$20 \log_{10} \left(\frac{A}{B}\right) = 50 \text{ dB}$$

but for the case where the signal is squared

$$20 \log_{10} \left(\frac{A'}{B'}\right)^2 = 50 \text{ dB}$$

Therefore, at the input the dynamic range is approximately

increasing
significance

non-significant
region

peak detection output

significance output

input signal

no noise   estimated 0dB SNR
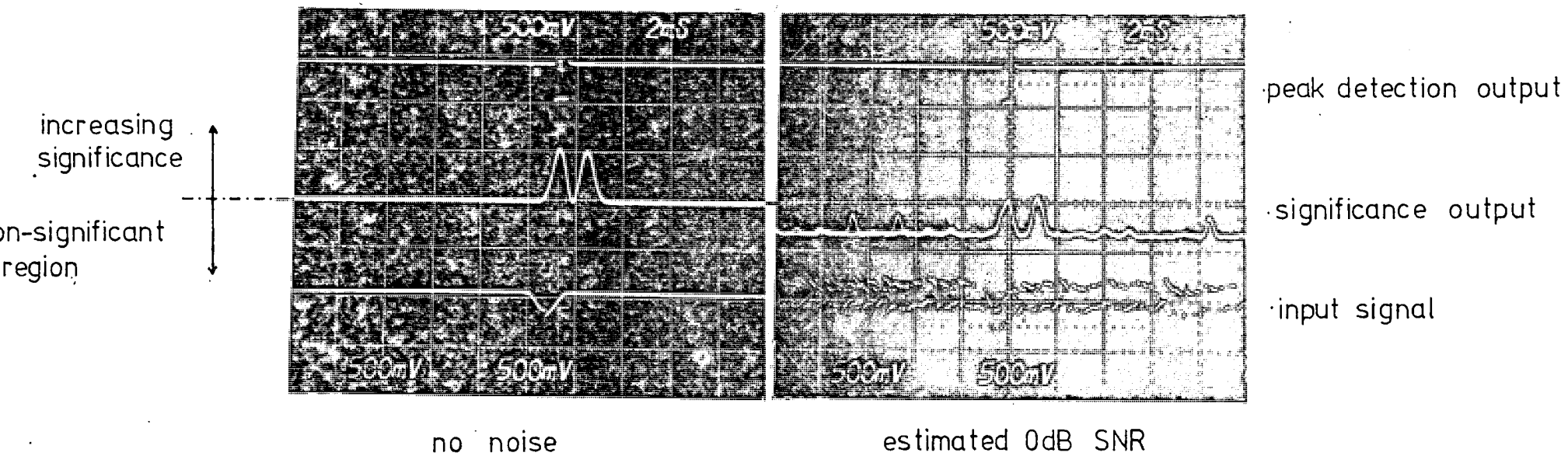
Figure 3.9: Results showing the significance output for a
triangular pulse input signal.

25 dB.

3.1.4 Summary

It has been demonstrated in principle that it is feasible to apply statistical analysis technique to the problem of non-matched filtering, and that the use of these techniques have certain basic advantages over more conventional methods. These advantages mainly concern the automatic, noise controlled, thresholding technique and the exclusion of low frequency noise from the signal by the use of derivative detection rather than by the more normal amplitude detection.

In addition, it had been shown that charge-transfer devices provide a suitable basic building block for such a system providing for easy adjustment of the system characteristics. A small, low power system of this type is ideally suited to use as a constant false alarm rate decision module following a noise reduction system, in particular communications receiver applications (eg. sonar-return processing). Applications to scientific instruments used inherently in 'noisy' experiments is also a possibility, replacing non real-time techniques[41].

A major problem associated with the processor described here is that of stability. This is due to the high dependence of the processor on d.c. levels produced at the

output of various analogue squaring and filtering circuits.
It was found that the outputs tended to drift to
unacceptable levels for quite small temperature changes,
of the order of 1 or 2^0 C. This type of behaviour is
clearly unacceptable in a module of this type since
drifts of this kind seriously affect the accuracy of pulse
detection.

3.2   Cancellation of fixed pattern noise in SPS delay lines

The basic operation of a serial-parallel-serial (SPS) CCD
delay line has been discussed in chapter 2. Although the
structure does have the advantage of reducing the overall
charge-transfer inefficiency involved in the total delay
it does tend to accentuate the problems associated with
dark current noise[43]. This is due to the fact that an
incoming signal may be transferred over any one of 32
separate delay paths (using the M67 SPS delay line
described in section 2.4). Each of these delay paths will
contribute different amounts of dark current to the signal
charge packet due to differences in the device structure,
such as local stacking faults and carrier concentration
variations in the silicon substrate. The overall effect
is a fixed pattern noise which is added to the output
signal.

The one fact which is known about this noise pattern is
its repetition period which is fixed at 32 clock cycles

for the M67 device since there are only 32 individual
signal paths within the device. This means that it is
possible to coherently integrate over the period of the
fixed pattern noise and then cancel it at the SPS delay
line output. In this section the design of such a
canceller is described and some typical results are
presented to demonstrate the operation of the experimental
canceller.

3.2.1   System Implementation

It is important if a system such as this is to be
commercially viable that the noise cancellation scheme
should be physically compatable with the technology used
in the SPS device manufacture and should be simple enough
to enable integration on the same chip as the SPS device.

The cancellation scheme chosen for the experimental
system is shown in figure 3.10. The principle of
operation is that the output of the SPS device is
integrated by the linear (n stage) CCD delay line loop so
that a replica of the output noise pattern builds up in
the linear CCD. The pattern is then subtracted from the
SPS output. The result shown in figure 3.11 shows a
typical output from the experimental integration loop
(which was constructed using the WM2100 TDL described in
chapter 2) when the input to the SPS delay line was
grounded. The trace shown in figure 3.11(b) shows the
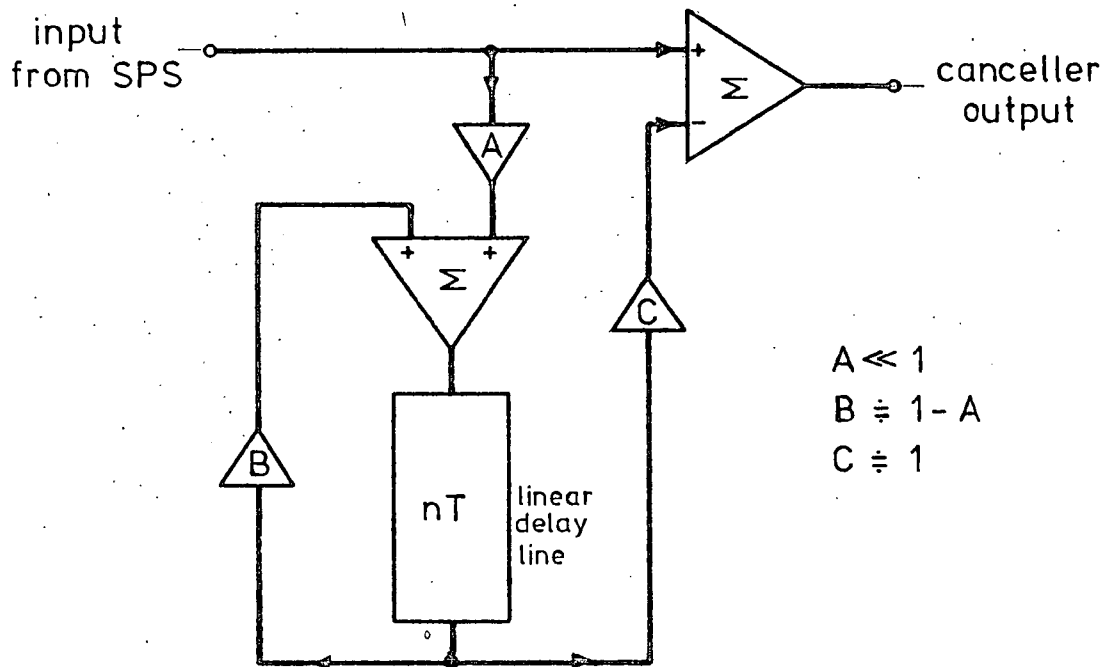
Figure 3.10: Block diagram of the experimental noise canceller.

vert. scale   200 mV/div
horiz. scale   0·1 ms/div

(a)   output from SPS delay

(b)   cancellation output

(c)   integrated noise pattern
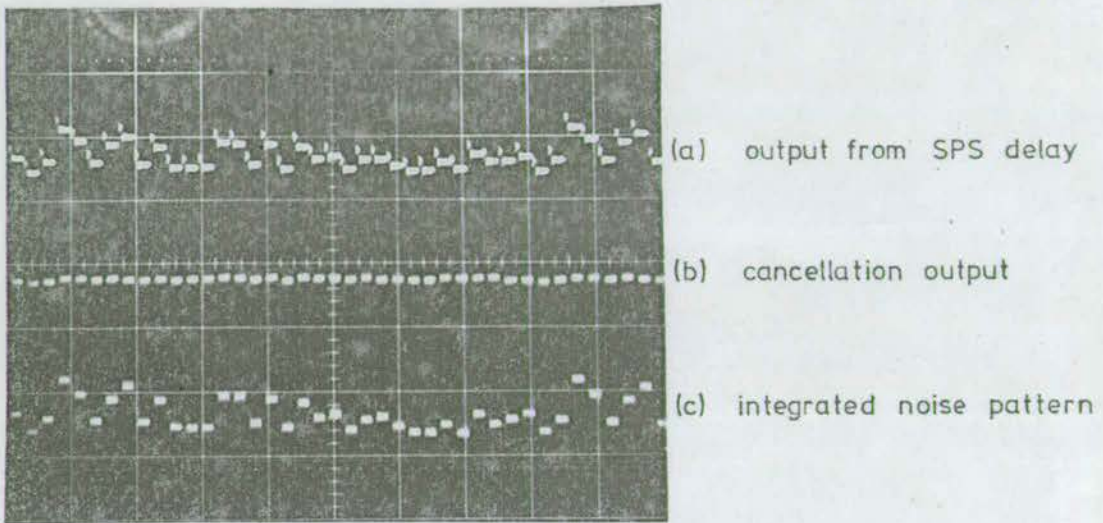
Figure 3.11: Result demonstrating the canceller performance
for a zero signal input.

integrated version of the SPS noise pattern and figure 3.11(c) shows the canceller output which may be seen to be relatively noise free, in comparison with the original output.

One defect which does occur using this cancellation scheme is that the integration loop imposes a comb filter response on the output with nulls at frequencies which are integer multiples of f_c/n. However for speech or video signals it will be assumed that truly periodic signals do not exist for long enough for a notch to form at the canceller output. The graph shown in figure 3.12 illustrates the measured frequency response of the system when the input signals are pure sinusoids.

3.2.2 Experimental Results

The result shown in figure 3.13 is for a sinusoidal input to the SPS device (figure 3.13(a)). The SPS output is shown in figure 3.13(b) and the effect of dark current fixed pattern noise is clearly visible here.

The integrated noise pattern in shown in figure 3.13(d) and the actual canceller output is shown in figure 3.13(c). From the frequency spectra shown in figure 3.13(c) and (f) the improvement in effective signal to noise ratio is approximately 20 dB.
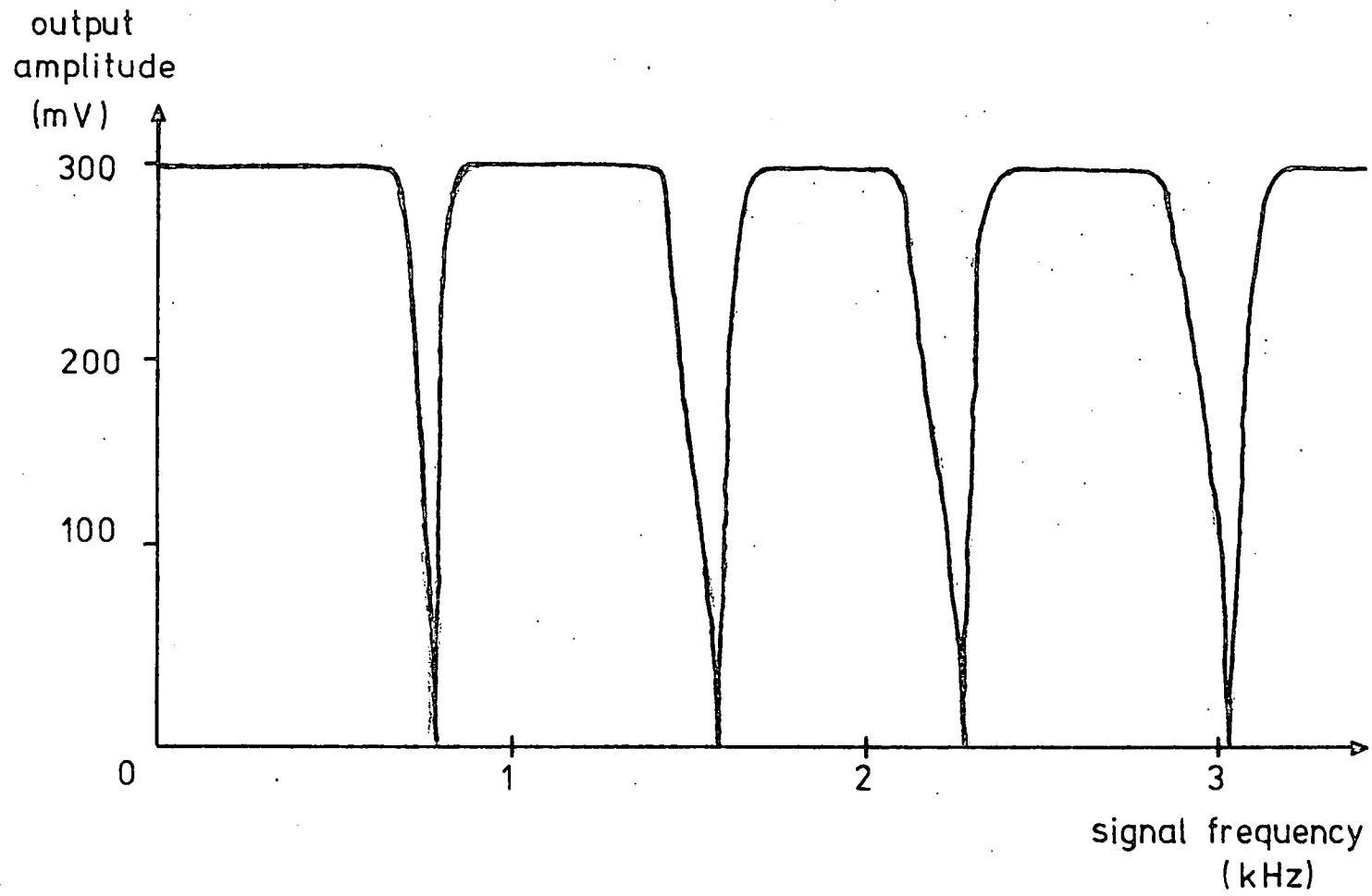
Figure 3.12: Frequency response of the canceller circuit.
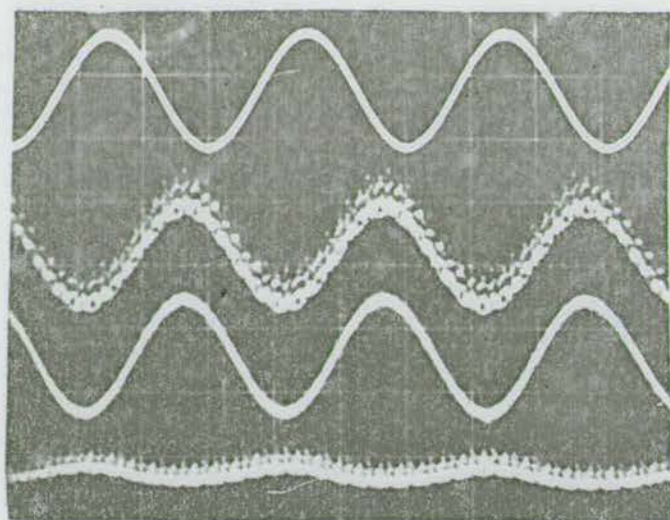
A signal input

B SPS output

C canceller output

D integrated noise
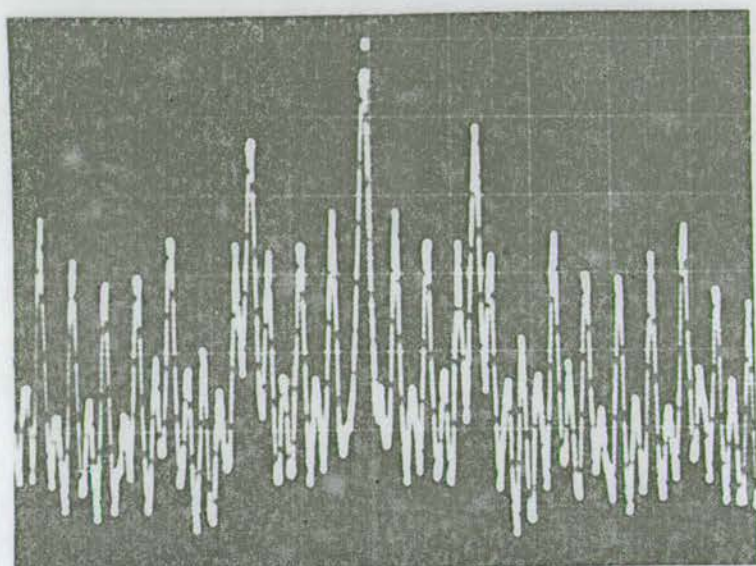
E input signal spectrum

10 dB/div

2 kHz/div
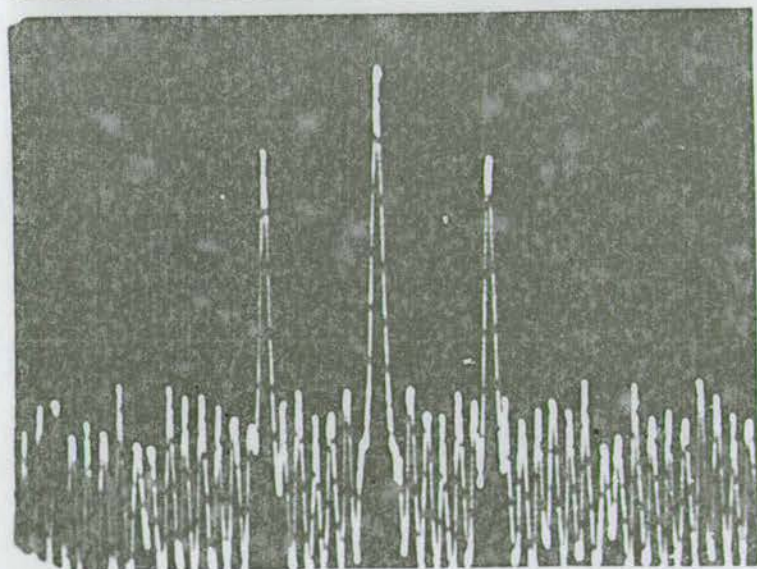
F canceller output

Figure 3.13: Result showing fixed pattern noise cancellation in the presence of a sine wave input signal.

A certain amount of signal distortion was obvious in the test signals used mainly due to the relatively high values of the gain A necessary at the integrator input. This high gain was necessary due to the cumulative transfer inefficiency in the noise integrator meaning that integration had to take place over a relatively short period of time. This situation could be improved either by using a buried channel CCD in the integrator (the experimental module used a surface channel device), or by using a simple transfer inefficiency compensation filter[34].

3.2.3 Summary

The feasibility of SPS fixed pattern noise cancellation has been demonstrated showing that it is possible to considerably enhance the effective signal-to-noise ratio of a signal at the output of an analogue SPS delay line. Potential future developments of this concept are:

(1) Monolithic integration: This circuit could be integrated monolithically with the SPS device, since the technology for the cancellation scheme is compatible and all gains in the integration loop are less than unity. Thus simple MOS buffers and charge manipulation techniques may be employed to complete the cancellation loop.

(2) Overall feedback: The present cancellation system is an open loop processor depending on externally applied gains. A desirable system would

be a closed loop system employing signal feedback to minimise dependence on loop gains.

(3) Alternative cancellation schemes: Such as the use of delta-modulation at the CCD output to cancel fixed noise components with minimal effect on required signal components.

Again the major problem found here was that of stability due to the d.c. terms in the CCD integration loop. This problem could be solved to some extent here by a.c. coupling of signals around the integration loop, but a more attractive solution would be to use the averaging techniques advocated by Traynar and Beynon[44] which suffer from none of the inherent stability problems of recursive filter structures.

CHAPTER 4:   CLOSED LOOP SYSTEMS

In the previous chapter the processing systems under consideration were so called 'open loop' systems, meaning that no continuous dynamic control is available over their operating gain and bias levels. In the following chapters the systems which will be considered are closed loop; i.e. some form of external feedback is applied around the filtering device to provide continuous dynamic optimisation of gain and bias parameters. This means that the closed loop system may be designed to be stable and free from drift due to temperature changes. An added advantage is a reduction in the required accuracy of 'setting-up' procedures due to the inherent ability of the closed loop system to adjust its own bias levels. These characteristics will be demonstrated theoretically and experimentally in the following chapters.

4.1 General adaptive systems

In general a closed loop filtering system is known as an adaptive filter and it involves several fundamental differences in overall operation from a conventional open loop processing system. With a normal open loop system the statistics of incoming signals and any accompanying noise or distortion must be known a-priori. It is then necessary to define exactly how the noise and/or distortion may be optimally reduced and this a-priori

information must be used in the design of a dedicated processing system.

However, with a true adaptive system an absolute minimum of a-priori information is necessary about the incoming signal. This is because the adaptive filter operates by measuring the statistics of the incoming signal and adjusting its own impulse response in such a way as to minimise some cost function. This cost function may be derived in a number of ways depending on the intended application but normally it is derived by the use of a second signal source (see the general block diagram in figure 4.1). This secondary signal input d(t) may be defined as the desired output of the filter, in which case the task of the adaptive algorithm is to adjust the weights in the programmable filter device in such a way as to minimise the difference between the filter output and the secondary d(t) input.

It can be shown that when the filtering device is a finite impulse response transversal filter the optimum weight vector is the Weiner[45] solution given by:

$$\underline{H}_W = \underline{R}^{-1} \; \underline{P} \; \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots(4.1)$$

where $\underline{H}_W$ is a column vector of the weight values supplied to the transversal filter;

$$\underline{R} = E\left(\underline{S}\underline{S}^T\right)$$

Where $E()$ denotes the expected value, $\underline{S}$ is a column vector of signal samples stored in the filter and the
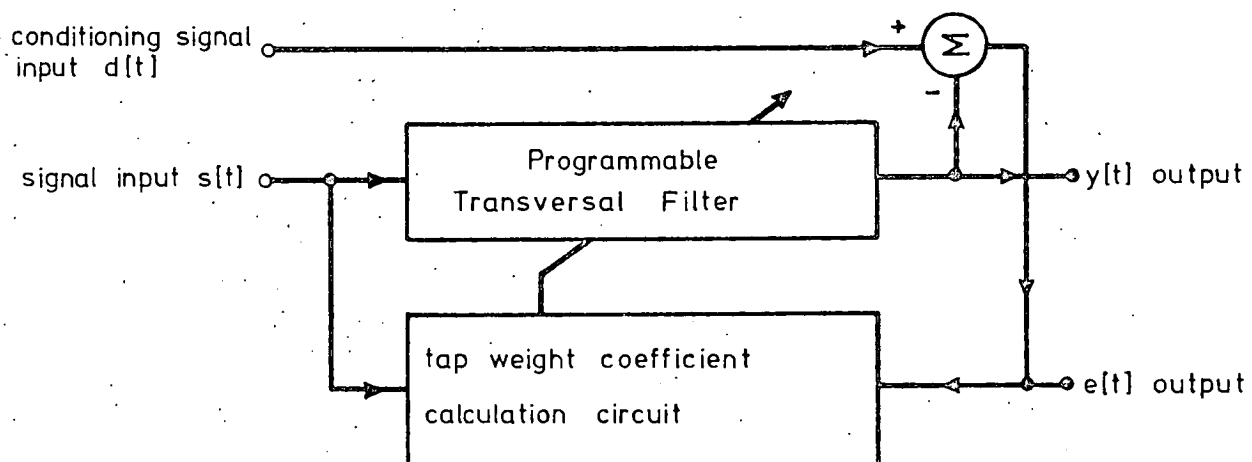
Figure 4.1: Block diagram showing the construction of a general adaptive processing element.

superscripted T represents the transpose of the matrix; and

$$\underline{P} = E\bigl(d(t)\ \underline{S}\bigr)$$

One of the earliest true adaptive systems was designed by Gabor[46] and used a complex analogue computer system with a capability to adaptively adjust 18 filter weight coefficients so that the output approached as closely as possible to some target function.

Many of the later developments in adaptive filter structures were aimed specifically at the equalisation of telephone channels for digital data transmission. This class of system is typified by the filters designed by Lucky[47] which used a least-mean square criterion to minimise the error function. The least-mean square technique was later refined and developed by Widrow[48-50] et al at Stanford and later work by Moschner[51] investigated the effects of simplifying the Widrow algorithm.

Most of the work carried out at this point was primarily on the theoretical aspect of the adaptive filter and one of the first attempts at a realistic implementation of the Widrow algorithm was undertaken by Neissen[52] and also by Morgan and Craig[53] using an 8 tap analogue transversal filter with digital circuitry used to update and store the weight values. Although this work proved

successful, the size and power consumption of the adaptive

filter module was to prevent its general use. Later

attempts at realising these systems, notably the systems

designed by Corl[54], White[24] and Copeland[25], were

concentrated on minimising the size and power consumption

of the adaptive filter by the use of CCDs and LSI

techniques. However, in the work mentioned above no

attempt was made to optimise the design technique by

examination of the error sources within the analogue

devices used in the various realisations. This thesis

represents the first realistic attempt to evaluate the

effect of CCD error sources on the operation of a CCD-

based adaptive processor.

4.2 Adaptive algorithms

It is clearly possible to calculate the value of $\underline{H}$ given

by equation 4.1 directly, however this would involve a

processor of considerable complexity and power. Many

algorithms have been proposed which approximate the

result and a considerable effort has gone into the

theoretical evaluation of these algorithms both by the use

of analytical techniques and computer simulations. Using

the techniques of direct calculation of $\underline{H}$ described above

is to return essentially to the use of an open loop

processor; therefore, by definition, the method of

calculation of $\underline{H}$ to be used in this work was restrained

to being an iterative closed loop technique. It was

additionally required that the computation needed in order to implement the algorithm should be as simple as possible. The basic algorithm chosen to implement the CCD-based adaptive filter was the Widrow least-mean square (LMS) adaption algorithm[48] given by:

$$\underline{H}(t + 1) = \underline{H}(t) + 2\mu e(t)\underline{S}(t) \quad \ldots\ldots\ldots\ldots\ldots(4.2)$$

where $e(t) = d(t) - y(t)$;

   $y(t)$ is the filter output; and

   μ is a convergence factor which controls the stability and rate of convergence of the algorithm.

This algorithm uses a single value of the error $e(t)$ to estimate the gradient of a parabolic error surface and then uses this gradient estimate to reduce the error (in a least-mean square sense) by descending towards the minimum point of the error surface: the rate of descent and the eventual accuracy of the result being dependent on the size of μ, the convergence factor. A full derivation of the Widrow LMS algorithm may be found in reference 48 and a general block diagram of this system is shown in figure 4.2.

4.3   Adaptive filter applications

The same basic adaptive filter unit may be used in many important application areas essentially by changing only the way in which the input and output ports are used. These applications principally involve situations where it
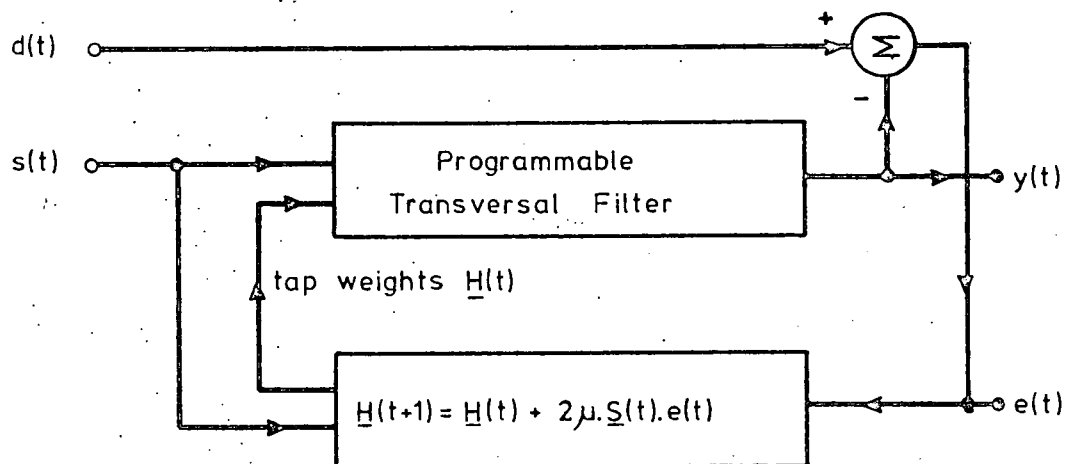
Figure 4.2: Block diagram of an adaptive transversal
filter using the linear Widrow LMS
adaption algorithm.

is impossible to predict the characteristics of the
environment through which a signal must travel.

There are two basic characteristics of an adaptive filter
which must be considered when such a system is applied in
a practical manner:

(1) Any component of the s(t) input correlated
with any component of the d(t) input will be reproduced
at y(t) in such a way to subtract coherently from d(t).
Any component of s(t) uncorrelated with d(t) will be
suppressed by the filter. Any component of d(t)
uncorrelated with s(t) will remain at the e(t) output.

(2) Many applications essentially use the
system modelling ability of the filter (see figure 4.3).
That is, if there is only one primary signal which is the
input to an unknown system and both input and output of
this system are available, then if s(t) is the primary
signal source and d(t) is the system output the optimum
adaptive filter impulse response is the same as that of
the unknown system. If s(t) is the unknown system output
and d(t) the primary input signal then the optimum
adaptive filter transfer function is the inverse of that of
the unknown system.

Using these two basic characteristics we may now consider
some of the major application areas to which the adaptive
filter may be applied.

(A) Medical electronics: This was one of the first areas

Primary signal

$H_W = H(s)$

Primary signal

$H_W = H^{-1}(s)$
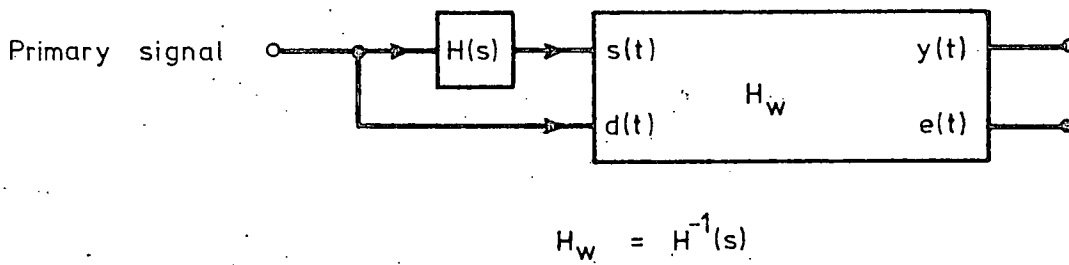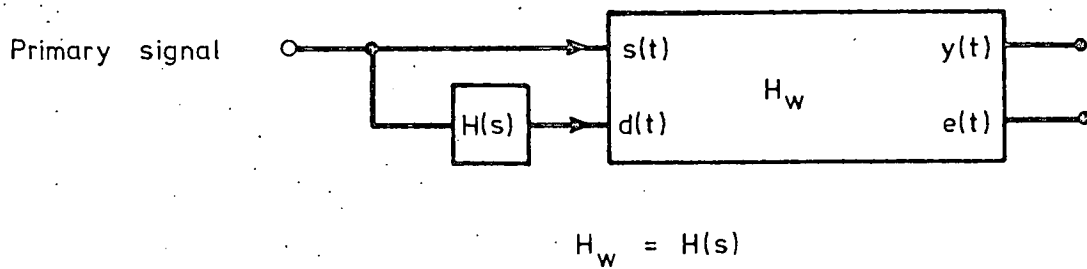
Figure 4.3: Configuration of the idealised adaptive filter in the two system modelling modes of operation.

in which adaptive filters were applied[48], principally
in the role of noise cancellers (see figure 4.4(b)). In
this configuration the primary signal (that containing
the desired information) is applied to the d(t) input.
This signal is contaminated by a spurious signal from a
secondary source. A sample of this secondary signal may
be obtained from another source (uncontaminated by the
required signal) which is correlated with the interference
on the primary signal. This correlated signal is applied
to the s(t) input and the filter forms an impulse
response which yields an output y(t) which subtracts
coherently the unwanted component from d(t) leaving the
desired signal on the e(t) output.

An example of the use of this technique is in the
monitoring of foetal heartbeat, where the primary signal
is obtained from a transducer array on the mother's
abdomen. There transducers yield a signal containing the
foetal heart signal heavily masked by the mother's heart-
beat. The secondary signal registering only the mother's
heartbeat is then obtained from a transducer array on the
mother's chest. The adaptive filter then models the
distorting path from the chest to the abdominal
transducers to produce the signal which coherently
subtracts from the abdominal signal.

(B) Speech processing: Here the major impact (aside from
communications processing) occurs when speech is
contaminated by periodic (or pseudo-periodic) signals or
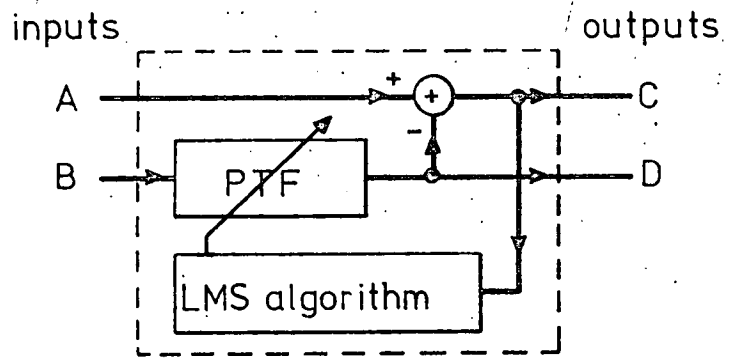
(a) basic adaptive filter configuration
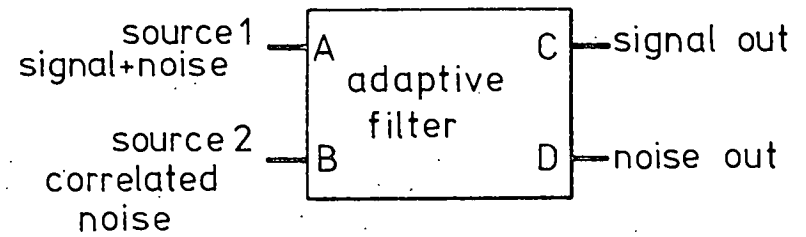
(b) noise canceller
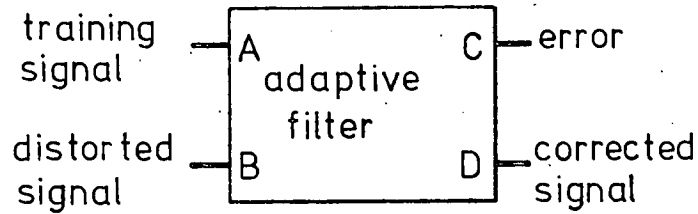
(c) inverse filter
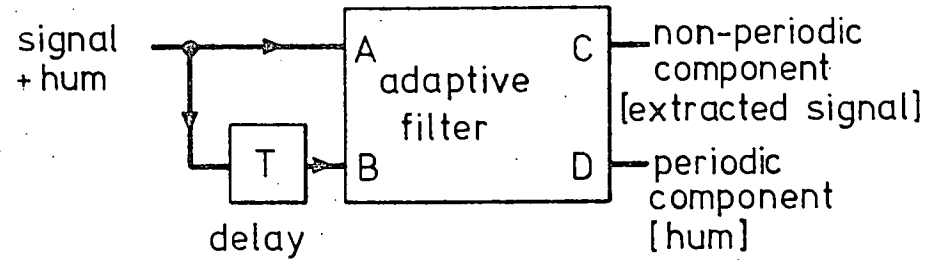
(d) self-tuning filter

Figure 4.4: Use of an adaptive filter in three basic applications

by so-called 'convolutional' noise[55]. The first
problem occurs for instance in trying tc record speech in
a room where music is being played (the musical tones in
general last long enough to be considered periodic) or
where constant interference exists (eg. hum or whistle on
HF radio reception). This type of problem may combated by
the use of the self-tuning filter configuration shown in
figure 4.4(c). Here the delay between the d(t) and the
s(t) inputs is sufficient to decorrelate any non-periodic
signal existing at the input. The filter then suppresses
the non-periodic signal leaving the periodic component
on the y(t) output, which subtracts coherently from the
d(t) output. The speech may be viewed as an essentially
non-periodic random signal and can therefore be stripped
of interfering hum or musical tones by using this
configuration of adaptive filter.

(C) Communications processing: Applications for adaptive
filtering in communications processing exist mainly in
the field of digital data transmission over switched
voice-bandwidth telephone lines[47,52,56]. The specific
problems encountered are those of echo and distortion.
With fixed telephone links these effects are not really
problematical since fixed equalisers may be constructed
to compensate for a known data link. However, when the
switched network is used each reconnection results in a
different set of distortion and echo parameters. It is
therefore necessary to use an adaptive equalisation
scheme which is capable of learning the optimum

equalisation impulse response on each reconnection in order to make full use of the available information rate on the communications links.

The problem with distortion is overcome in a relatively straightforward manner (see figure 4.4(d)) by transmitting a known data sequence at the beginning of each transmission for a short 'learning' period. The output of the transmission line is applied as the s(t) input and the receiver generates a perfect version of the known input signal as the d(t) input. After convergence the filter impulse response $\underline{H}_W$ is the inverse of the channel impulse response. At this point the adaption cycle may be stopped and the filter weights frozen to accommodate further, unknown, data transmission and a decision-directed feedback technique[66] may be used which automatically updates the weight vector to compensate for small changes in line characteristics due to temperature drift etc.

Echo effects take place primarily because of impedance mismatches between various sections of transmission line. This means that a terminal's own transmission is likely to interfere with the signal which it is attempting to receive. Hence, without some form of compensation, full duplex data transmission over the two-wire telephone link cannot take place. An adaptive filter connected as shown in figure 4.5 will allow full duplex transmission over a
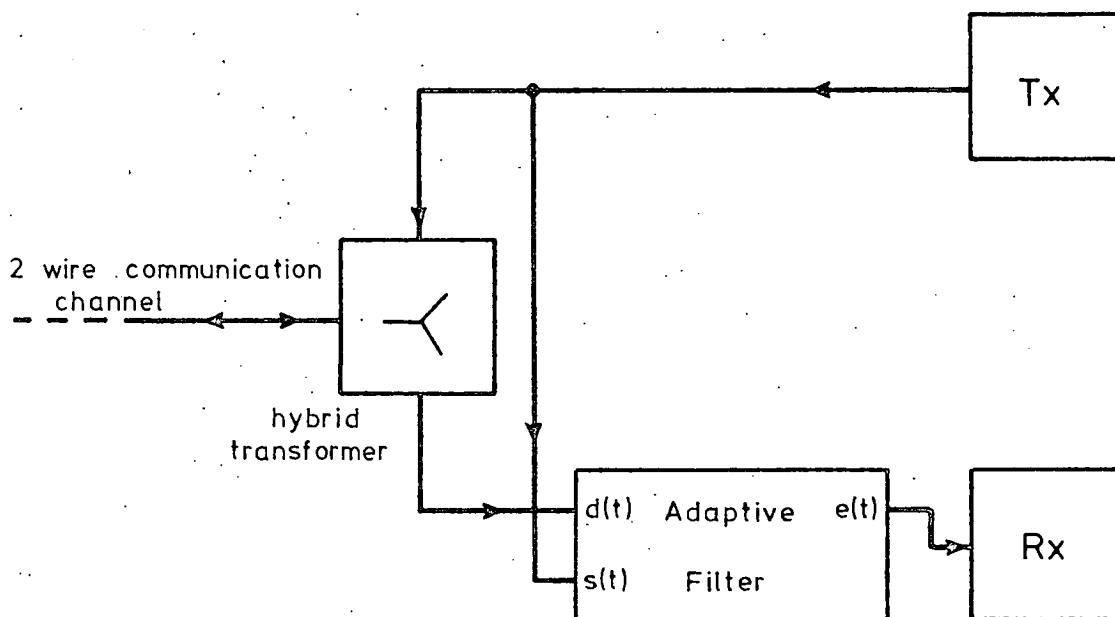
Figure 4.5: Adaptive filter configuration allowing full duplex data transmission over a single two-wire telephone channel with mismatched hybrids.

mismatched two-wire link[57]. The system configuration
in this case is essentially the same as that considered
for the detection of foetal heartbeat in section 4.1.

Adaptive filters may also be employed to overcome similar
echo problems in true voice loud-speaking telephones
(LSTs) and to compensate for multipath interference in
high-bandwidth satellite and ground-based microwave links.

An important point to note is that all these applications
use essentially the same type of processor with one of
three different external configurations:

      (1)   the echo cancellation configuration;

      (2)   the self-tuning filter; and

      (3)   the specifically trained adaptive equaliser.

Depending upon the application, the required bandwidth,
the convergence rate and the number of required filter
points all vary. A very desirable processor is therefore
one which has a very wide bandwidth range, and easily
adjustable convergence factor, μ, and is easily
cascadable. It is a processor having this general type
of specification which is investigated in the following
sections. The eventual objective is to produce a module
having the above specifications and, additionally,
consuming a minimum amount of power and having a minimum
space requirement.

CHAPTER 5: SYSTEM IMPLEMENTATIONS

5.1 Variants of the LMS adaption algorithm

Given the objective of constructing an adaptive filter of minimum size and power consumption it is clearly impossible to use high order adaptive algorithms which commonly use matrix manipulation techniques to calculate the weight vector $\underline{H}$. Therefore algorithms of the type shown in equation 4.2 only were used in this work. The algorithm of equation 4.2 may be degraded in various ways to alter its characteristics[32]. If the algorithm is rewritten in the following form:

$$\underline{H}(t + 1) = \underline{H}(t) + 2\mu(inc) \dots\dots\dots\dots\dots\dots(5.1)$$

then the algorithm type is defined by the form if the weight increment (inc) which will always be a function of s(t) and e(t). Four forms of (inc)[24,47,51,58] are commonly used and these are listed in Table I. The choice of algorithm is restricted by the type of circuit used to implement it and by the desired convergence properties of the adaptive filter. For instance the clipped LMS[51] algorithm is frequently preferred for speech processing since clipping s(t) tends to whiten its frequency spectrum[55] and therefore enhances convergence rate. However the linear LMS algorithm[48] (stated in equation 4.2) is the most flexible of the above variants since it is easily degradable to form the other three variants.

DESCRIPTION	FORM OF (inc)	CONVERGENCE DEPENDENCE	CIRCUIT TECHNIQUE
LINEAR LMS	$e(t)\ s(t)$	POWER	ANALOGUE
CLIPPED LMS	$e(t)\ \text{sgn}\{s(t)\}$	AMPLITUDE	ANALOGUE/DIGITAL
HYBRID LMS	$\text{sgn}\{e(t)\}\ s(t)$	AMPLITUDE	ANALOGUE/DIGITAL
ZERO FORCING	$\text{sgn}\{e(t)\}\ \text{sgn}\{s(t)\}$	ZERO CROSSING	ANALOGUE/DIGITAL

Table I: Basic characteristics of the LMS algorithm variants.

The objection most commonly levelled against the use of the linear LMS algorithm, particularly with reference to its application to speech processing, is that its convergence properties are degraded by the dependence of convergence on the incoming signal power as opposed to the input signal amplitude dependence of the clipped LMS variant. This characteristic can however be counteracted by modulating the convergence factor μ by an estimate of the signal power inverse[32] such that μ becomes:

$$\{ \frac{1}{N} \sum_{n=1}^{N} s^2(t) \}^{-1} .$$

This effectively widens the range of convergence of the algorithm in terms of input signal dynamic range.

However in considering the available circuit techniques there are obvious limitations on algorithm choice when using analogue and digital circuitry. Since multiplication is of primary importance with digital circuitry, figuring very prominantly in determining overall size and power consumption, it is clearly undesirable to have extra linear multiplications which can be avoided. For this reason it is desirable to use one of the algorithms involving clipping of one or both of the multiplicands when the implementation to be used is digital.

On the contrary, linear multiplication using analogue circuitry (to a low accuracy) is relatively easy and the linear LMS algorithm is therefore more suited to an analogue circuit implementation. Figure 5.1 shows the basic filter cell for the linear LMS and the clipped LMS adaption algorithm. It can be seen from this that the clipped multiplication of the latter algorithm involves considerable complexity for an analogue circuit (the exclusive OR alone requires 11 MOS transistors) whereas the alternative linear multiplier may be implemented using 4 MOS transistors (see chapter 9).

5.2 Adaptive system architectures

Having defined the type of adaptive algorithm to be used there still remains the question of the optimum system architecture to yield maximum efficiency for a given range of applications. In this section three alternative forms of adaptive processing implementation are considered with reference to their possible implementation using analogue or digital circuitry. The three system architectures[32] are:

(1) Parallel processing;

(2) Burst mode processing; and

(3) Serial mode processing.

These architectures are listed in Table II which also summarises the important characteristics associated with each.

ALGORITHM IMPLEMENTATION	CIRCUIT TECHNIQUE	FILTER POINTS PER DEVICE	BANDWIDTH	NORMALISED CONVERGENCE RATE
PARALLEL	ANALOGUE	>100		
	DIGITAL	NOT FEASIBLE	HIGH	FAST
	HYBRID	≈32		
BUSRT	ANALOGUE			
	DIGITAL	>100	LOW	FAST
	HYBRID			
SERIAL	ANALOGUE		HIGH	
	DIGITAL	>100	LOW	SLOW
	HYBRID			

Table II: Comparison of possible system architectures using the Widrow LMS adaption algorithm.

(1) Parallel processing: This is an implementation in which the tap weights are updated at each tap by a dedicated update circuit. The total filter would then consist of N cascaded filter cells such as those illustrated in figure 5.1. The total structure is shown schematically for the linear LMS algorithm in figure 5.1(b). This may be achieved relatively easily using analogue techniques due to the simplicity of the analogue multiplier and offers the most compact processor with no bandwidth dependence on filter length. Implementation of this type of processor in digital form is however prohibitively complex for any appreciable time-bandwidth product.

(2) Burst mode processing: This architecture is applicable to both digital and analogue realisations and the two realisations will be dealt with separately:

(i) Digital realisation; the complexity of the digital multiplier means that it is desirable to use as few hardware multipliers as possible in the digital approach to filtering. Therefore only one multiplier is used and it is time multiplexed to perform each signal/tap weight multiplication sequentially. The multiplier output is then accumulated to form the convolution result. This is the so-called DELTIC[59] processor which is illustrated schematically in figure 5.2. The addition of an adaptive algorithm to the DELTIC processor is a trivial problem since only one extra multiply and
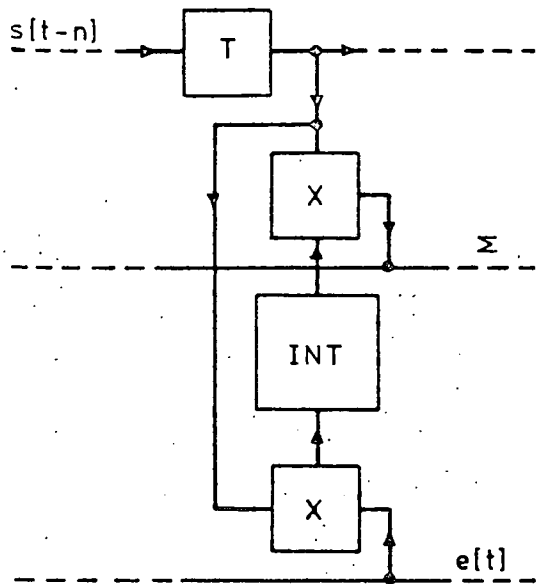
(a) Adaptive filter cell using the linear LMS update algorithm.
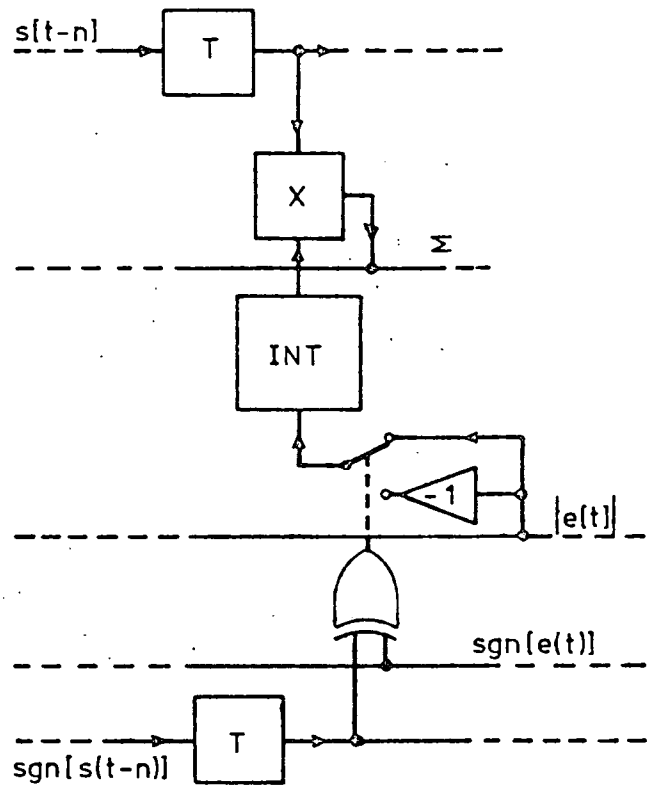
(b) Adaptive filter cell using the clipped LMS update algorithm.
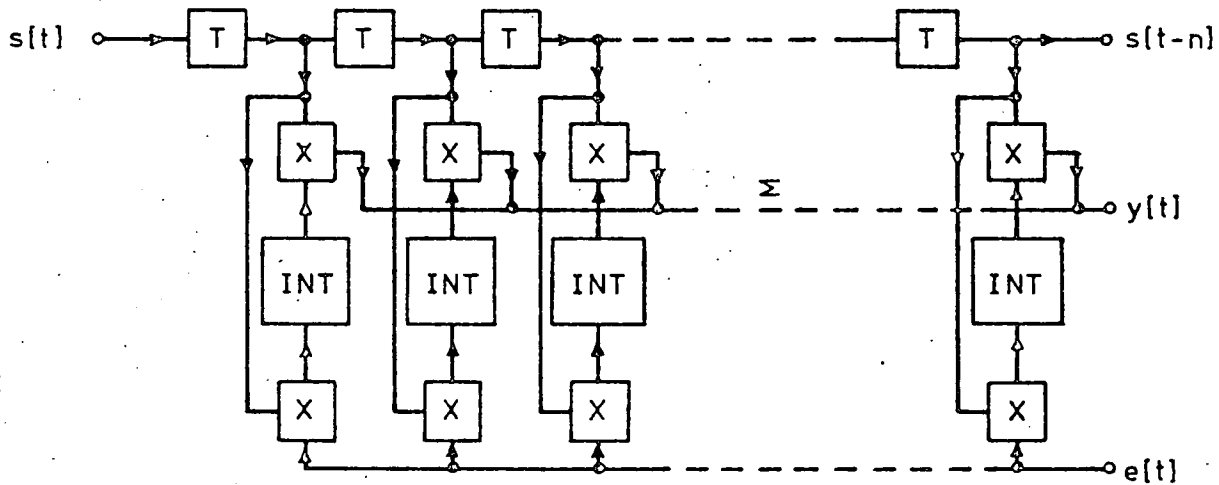
Figure 5.1: Parallel update adaptive filter configurations using the Widrow LMS adaption algorithm.
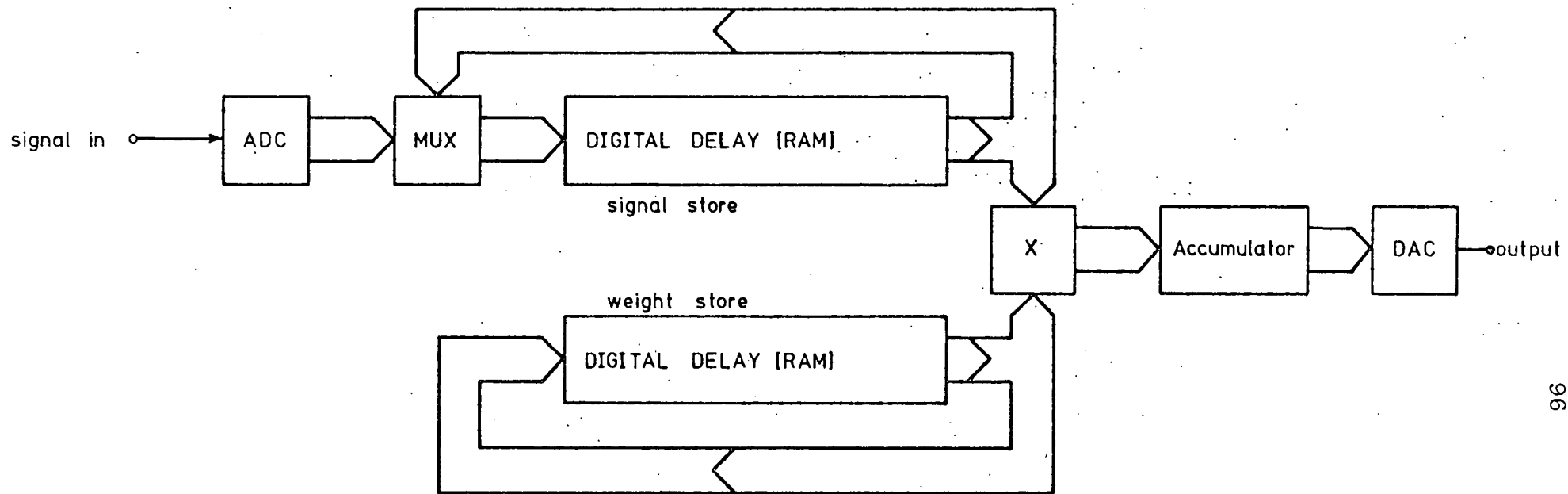
Figure 5.2: Block diagram of the deltic digital transversal
filter.

accumulate circuit need be added to derive the (inc)

function and add it to the weight which has already been

serialised in the filter element. Due to this

serialisation there is an inherent loss of bandwidth in

this processor proportional to the time-bandwidth product.

A more detailed design for this type of processor is

considered in section 5.3.

(ii) Analogue realisation; here we assume that

the filtering function had already been implemented in

analogue form as a parallel processor such as the CCD

programmable transversal filters described in chapter 2.

However, the PTF does not contain the necessary circuitry

to perform the update needed to implement the adaptive

algorithm. It is therefore necessary to perform this

updating function using digital circuitry outside the PTF

integrated circuit. To do this each individual product,

s(t-n).e(t), is calculated within one clock period (N

products where N is the filter length). These products

are added to the existing tap weights which are stored

digitally in a RAM and the result is fed back into the

PTF in analogue form before the next clock cycle begins.

A block diagram showing one version of this type of

system is shown in figure 5.3. Again the bandwidth of the

system is degraded in direct proportion to the total time-

bandwidth product due to the time required to calculate

the series of products s(t-n).e(t).

(3) Serial mode processing[27]: In this type of
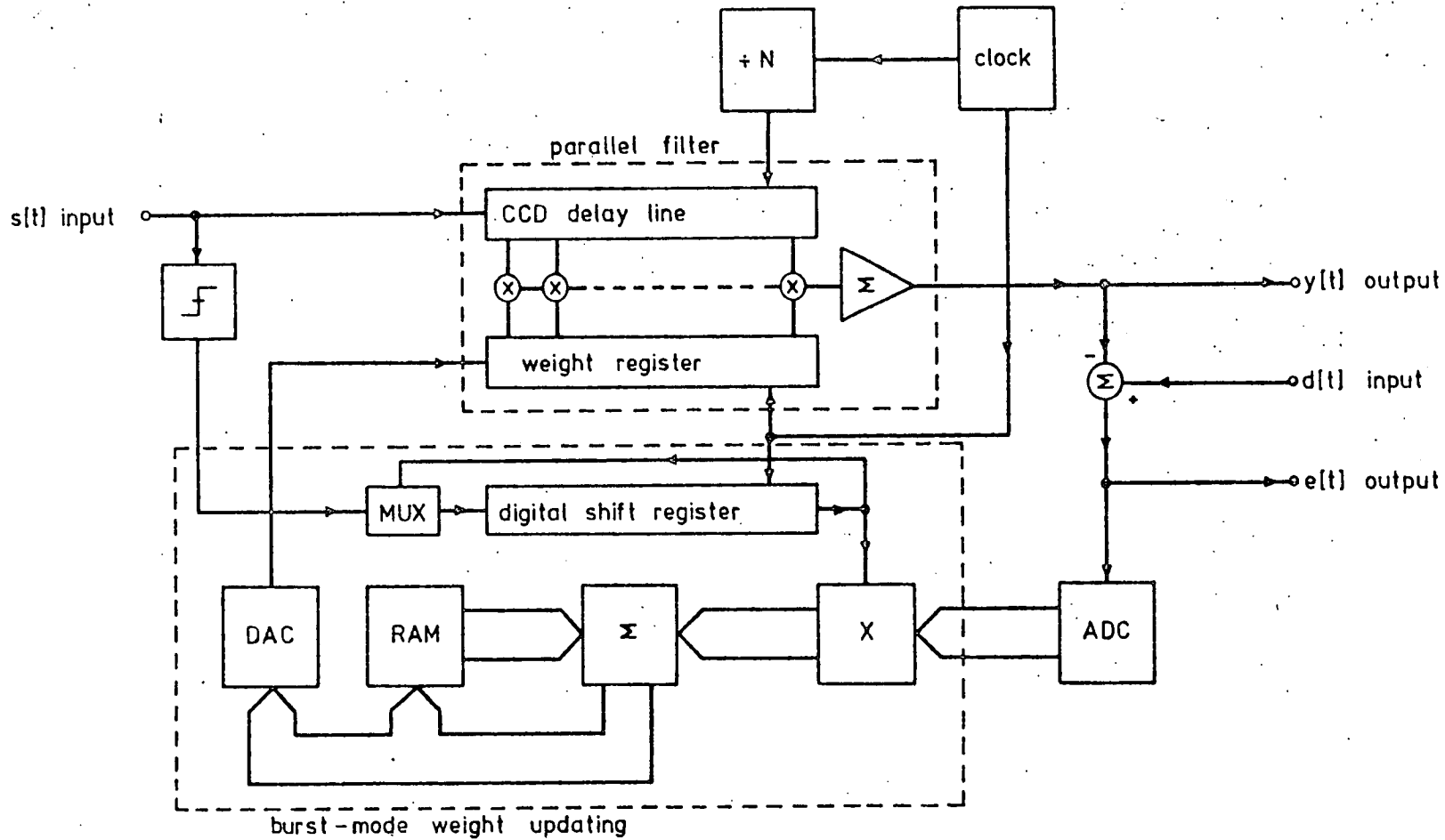
Figure 5.3: Block diagram of an analogue burst mode update adaptive filter based on a CCD PTF.

implementation the circuitry used is mainly analogue.
The analogue PTF mentioned above is again used to
implement the filtering function. However here, rather
than sacrifice overall system bandwidth to implement the
adaptive algorithm, convergence rate is degraded by only
updating a single tap weight at each input sample period.
This means that it takes N input sample periods to update
the entire weight vector once, therefore the convergence
rate of the filter is N times slower (where N is the
filter length). It can be shown that the other
convergence properties of the filter are essentially the
same as for the parallel update filter and this is dealt
with in some detail in chapter 6. The use of this
technique means that CCD PTFs with large numbers of filter
points may be used without loss of bandwidth. The loss in
convergence rate can be tolerated in systems where the
input signal statistics remain stationary for long periods
of time. The serial update mode has the additional
advantage that it ensures the uncorrelated nature of the
input samples, which is one of the conditions imposed on
the convergence of the LMS algorithms, to a greater
extent than a continuously sampled version of the filter.
This arises from the fact that more data passes through
the filter for each full update cycle, an additional
implied assumption is that the data is random.

Adaptive filters in general are easily cascaded provided
that the error derivation part of the circuit may be

readily separated from the main system. This is easily achieved with the systems architectures described above. The filtering sections are cascaded by cascading the delayed output signal from one section to the signal input of the next and adding all the filter outputs together. This cumulative output is then used to derive the error output, e(t), which is fed back to all the weight-updating circuits in parallel. Doing this involves greater difficulty using digital circuitry than with analogue circuitry owing to the greater complexity of digital arithmetic operations. A practical difficulty with the addition of digital quantities tends to be the shift in position of the significant bit patterns as more filter sections are added in. Figure 5.4 illustrates how cascading of adaptive filter modules may be achieved using an arbitrary circuit architecture.

5.3 Experimental realisations

Two experimental modules were constructed in order to evaluate the performance of a CCD-based adaptive filter module. The first was an analogue CCD version[27] using the linear LMS algorithm with a system architecture based on the serial mode of weight updating described above. The second module was an all-digital version[32] of the type described as a burst mode processor using the zero-forcing adaption algorithm. The primary objective of constructing this second system (using the Deltic approach)
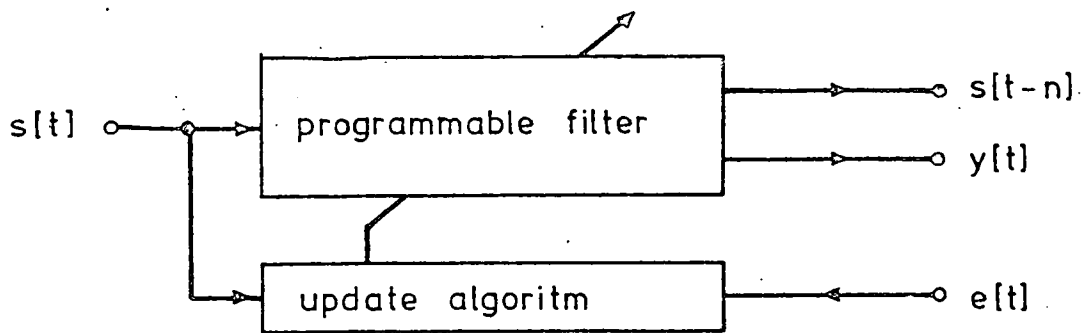
Figure 5.4: (a) General adaptive filter without internal error derivation.
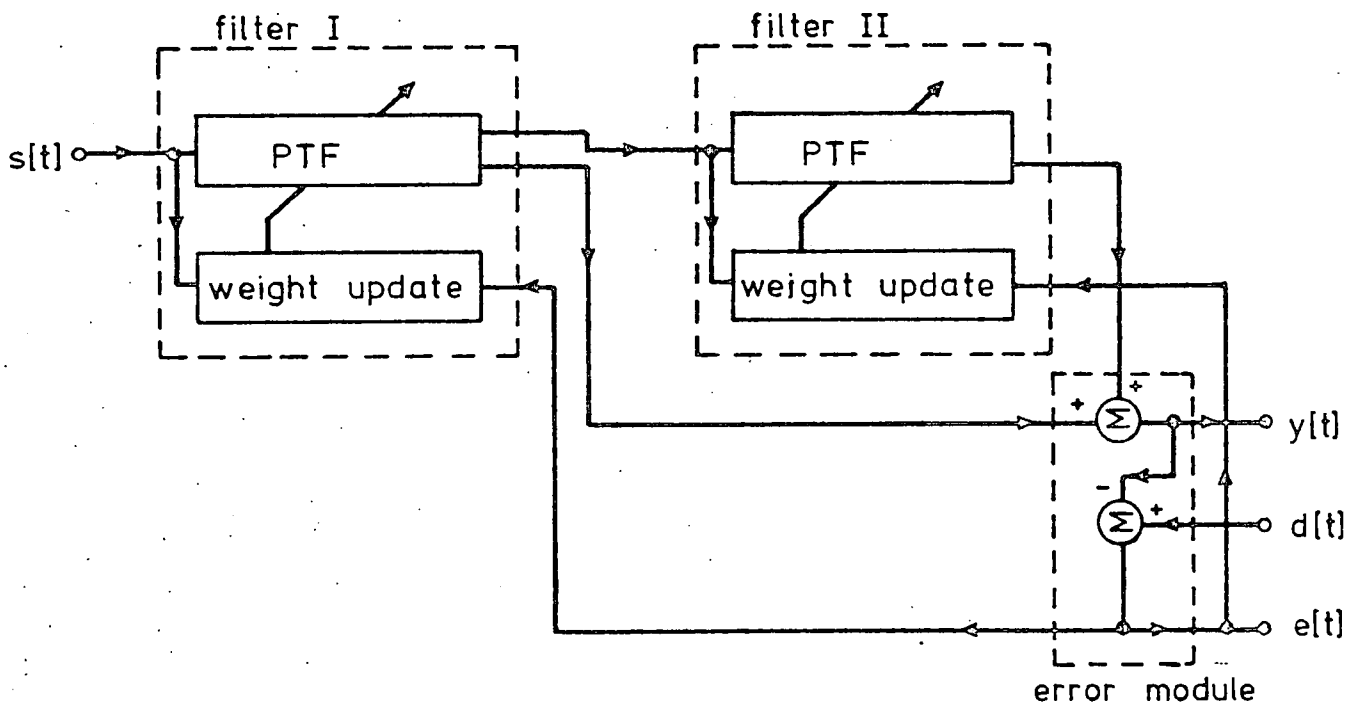
Figure 5.4: (b) Two adaptive filter modules cascaded using a third error derivation module to form a filter of length 2N.

was to provide a means by which the performance of the analogue CCD version could be compared with a known benchmark.

5.3.1 64-point adaptive filter prototype

This prototype system was based on the WM2100 programmable transversal filter described in chapter 2. The main objective in building this unit was to demonstrate the feasibility of using analogue CCD components, having relatively high error bounds, in the realisation of a practical adaptive filter structure. In using this PTF circuit it is clearly impossible to implement the parallel update circuit described in section 5.2 and the choice of architecture therefore lies between the burst mode and the serial mode processor. The burst mode update scheme proved to be unsuitable for this device due to the relatively long access time of the individual reference weighting cells. Use of this update technique would in fact have reduced the overall system bandwidth to around 2 kHz.

An additional problem with the device was that weight values stored in the reference could not be read out again due to the very large attenuation caused by the capacitive ratio which exists between the reference storage capacitors and the common reference input line. It was therefore necessary to duplicate the reference store outside the

filtering device using a memory device which could be both written to and read from.

In the case of one of the CCD prototypes this external memory took the form of a digital random access memory capable of storing 64 words each to 16 bits accuracy. However only the most significant 8 bits of each word was actually used to supply the analogue weights to the PTF. The reason for the excess length in the weight vector words was to allow more accurate determination of the significant weight value. This means that the error term $2\mu e(t)s(t)$ can be determined to 8 bit accuracy and any individual error applied to the weight integration cell will have a minimal effect on the actual weight value. Therefore, in effect, the weight value only changes after a period of integration of error increments which has the effect of performing a sliding average of the increments therefore yielding a more accurate estimate of the error gradient.

A general block diagram of this prototype is shown in figure 5.5. The error output $e(t)$ is sampled at one point in every 65 successive samples and is then multiplied in analogue form, by the signal on the last tap of the CCD delay line. This quantity is attenuated by the factor and converted to digital form. The result is then added to the last value of the relevant tap weight which is stored in digital RAM. This quantity is converted back to analogue
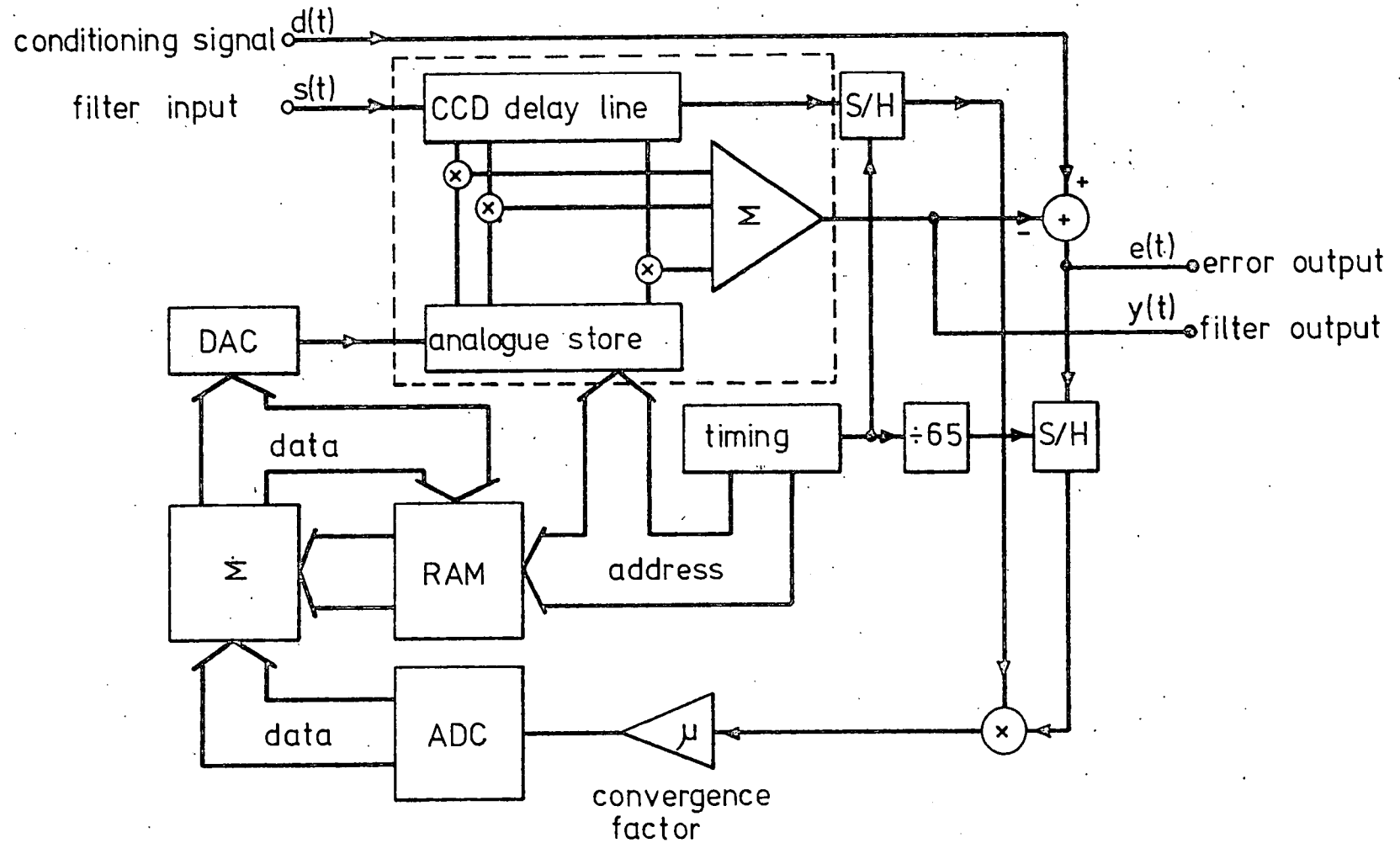
Figure 5.5: Block diagram of the experimental CCD adaptive
filter with digital weight update circuitry.

form and stored in the PTF weight register. Weight
updating takes place starting from the last tap and
progressing to the first tap to maintain the correct
ordering of information to the adaption algorithm. One
effect which the serial update has is to prevent adaption
to periodic inputs which have a period corresponding to
the update period. This is because, in this instance all
the information on the incoming signal is not available
to the processor and it therefore fails to converge.
This effect is explained in greater detail in the next
chapter and a solution to the problem is presented in
chapter 8. The reason for sampling e(t) at one point in
65 was that many test signals with a period of 64 samples
were used to test this structure and sampling e(t) at 1
point in 64 would have resulted in failure to converge
with these test signals.

A similar system to the one just described was also
constructed using external analogue storage of the weight
values instead of digital RAM. This was done primarily
to evaluate the effects of weight vector decay on the
adaptive algorithm. The analogue equivalent of the
digital RAM was constructed using an analogue multiplexor
system which was capable of switching a single input/
output line on to any one of 64 storage capacitors (see
figure 5.6). The very high ratio of 'off' to 'on'
resistance of these switches minimises leakage and a reset
cycle which occurs before the read part of the operation
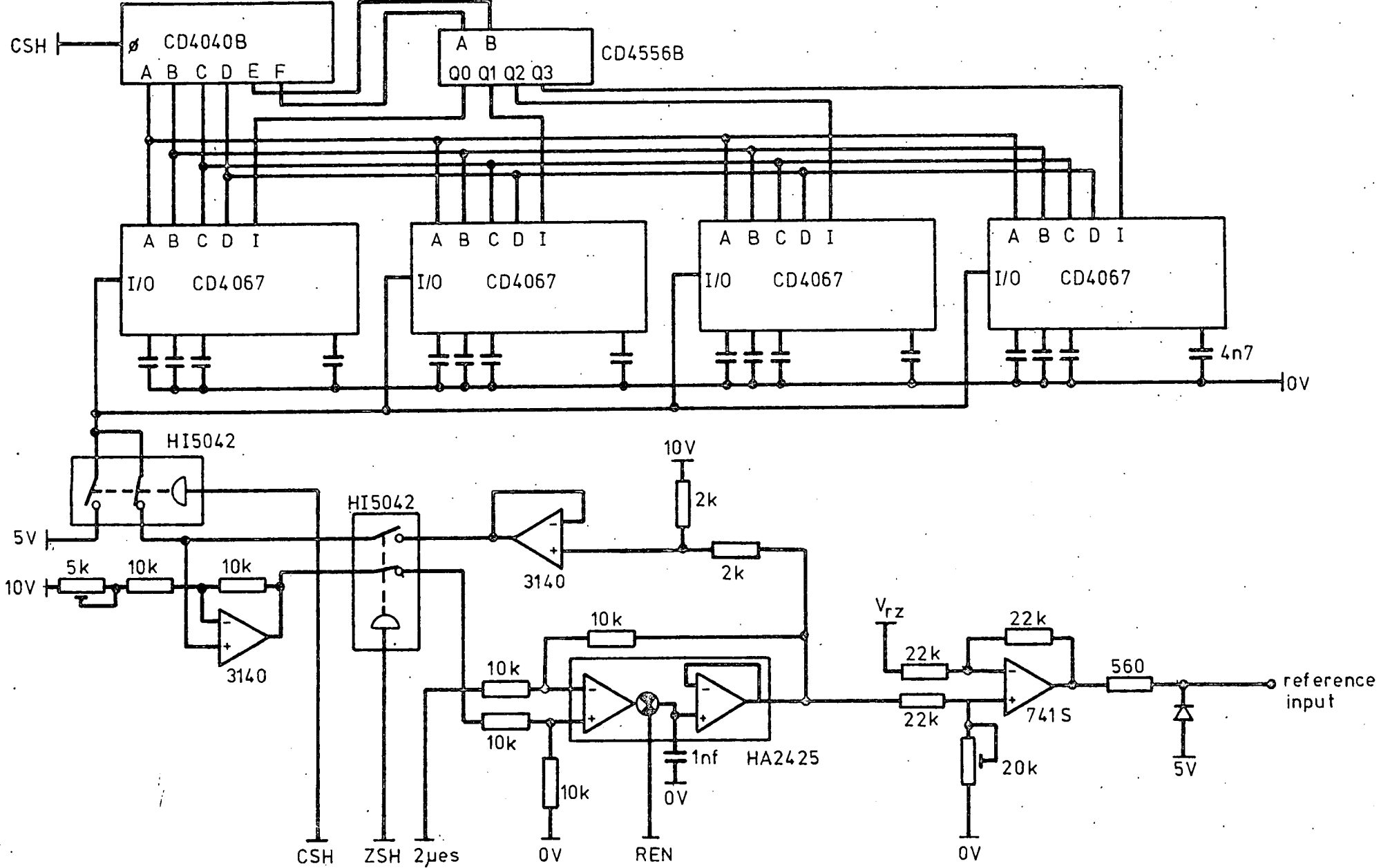
Figure 5.6: Schematic diagram showing the analogue weight storage and
update algorithm implementation used in the all-analogue
CCD adaptive filter realisation.

cycle minimises cross-talk between neighbouring samples.

The adaptive filter incorporating the analogue memory circuit was constructed on a single double-Eurocard circuit board using wire-wrap interconnect with a total power consumption of approximately 10 Watts. A photograph of this system is shown in figure 5.7. The circuitry necessary to implement the digital external storage and arithmetic was constructed on a separate circuit card which interfaced with the PTF on the analogue mother board. The achieved bandwidth of the system using digital weight storage was 25 kHz, the bandwidth being eventually limited by the conversion time of the analogue to digital converter. An agravating factor was the small proportion of the total clock cycle time (approximately 1/6) available for this conversion.

5.3.2 64-point digital adaptive filter prototype

The digital prototype adaptive filter[32] used the burst mode circuit architecture, the so-called DELTIC filter, to implement a zero-forcing algorithm. The use of the zero-forcing algorithm meant that only the sign bits of the signal and error needed to be multiplied to produce the required increment for the weight update. In this case a fixed convergence factor μ of 2^{-7} was used due to the difficulty in implementing an effective digital attenuator circuit. Therefore the total resolution of
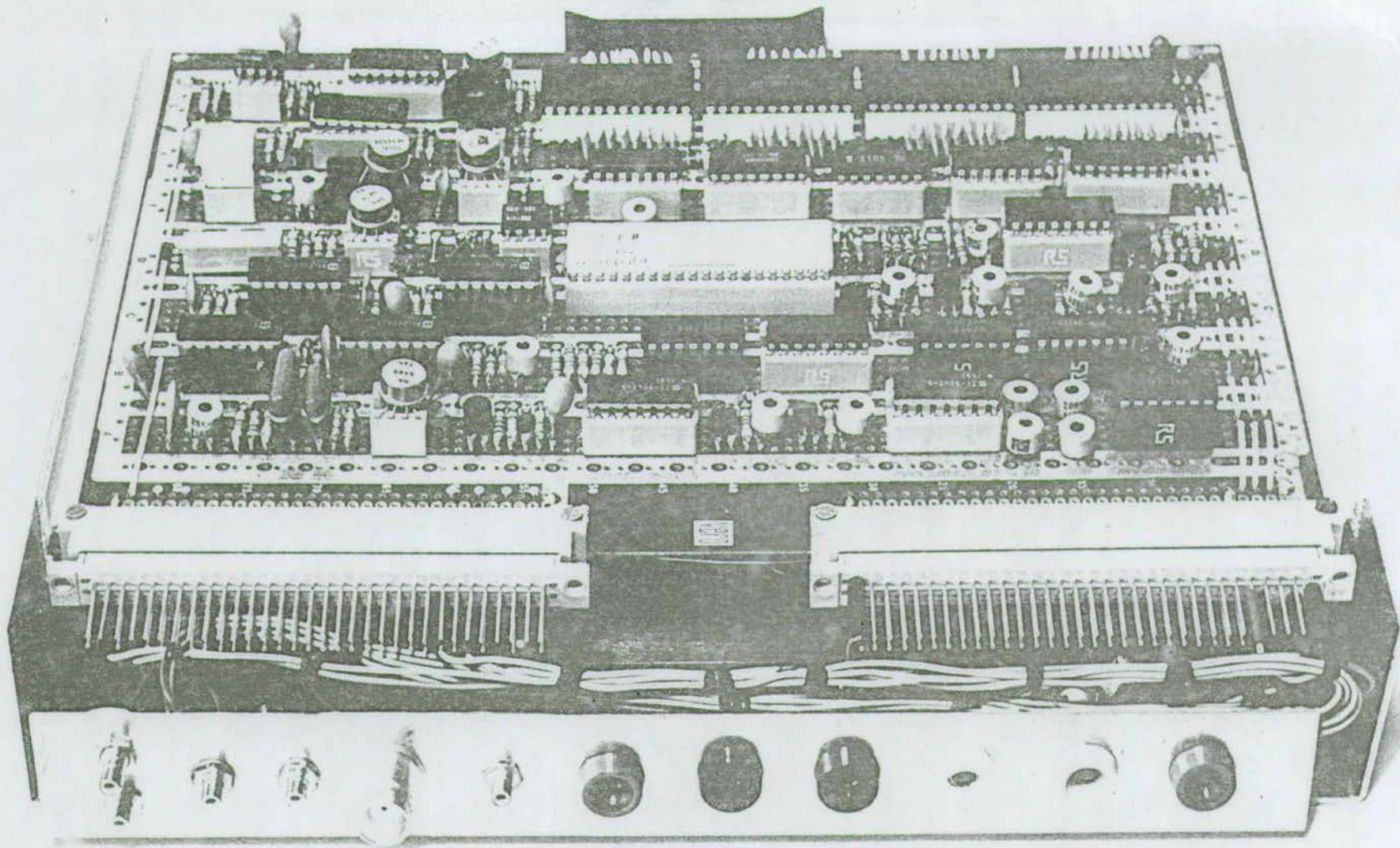
Figure 5.7: Photograph of the analogue CCD adaptive filter module.

each weight value was limited to 8 bits. This means that no increment integration takes place in this case since all 8 bits of each weight value are used by the filter multiplier. It is to be expected therefore that the accuracy of the weight vector solutions will be somewhat worse in this case than for the CCD system described in the last section using digital weight storage, particularly in the case where the filter is to be used in the noise cancellation mode of operation (c.f. figure 4). This may be seen to be true simply by considering the effect of the excess error $e(t)$ (i.e. the output signal) which is inherently present at high levels in a cancellation system. In the case of the CCD-based system described in section 5.2.1 these errors will tend to integrate in a non-coherent manner and therefore have little or no effect on the weight value. However, in the DELTIC system, these errors are added directly into the significant weight value and coherent integration over as few as 5 sample points will put the 3 least significant bits of the 8 bit weight words in error.

A block diagram of the prototype is shown in figure 5.8 which illustrates the ease with which a DELTIC filter may be made adaptive. The recirculating memories used to implement the signal and weight registers were both 64 x 8 bit MOS RAMs and the digital multiplier was a parallel 8 x 8 bit device with a total multiplication time of 100 nsec. The rest of the circuitry was implemented using low power
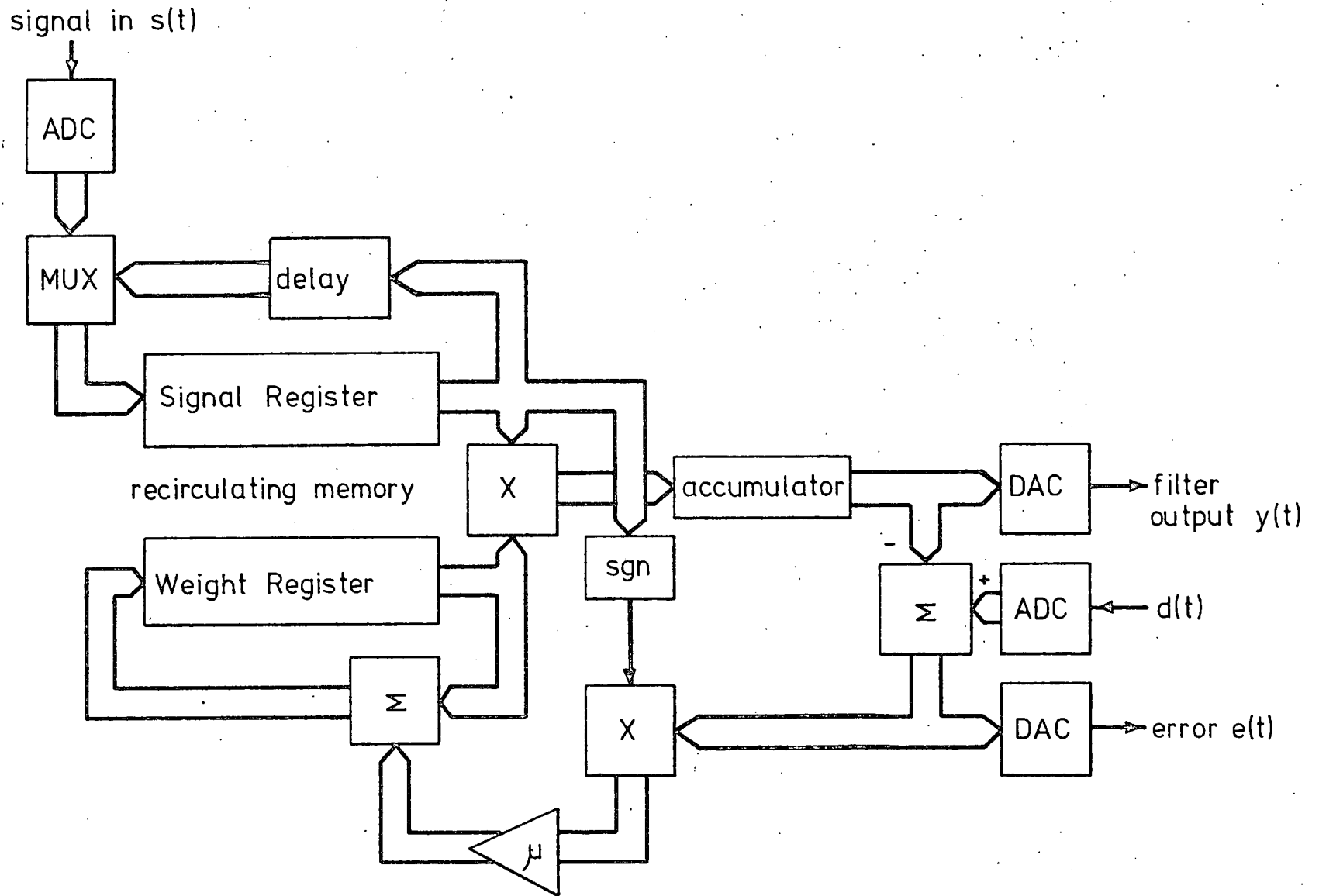
Figure 5.8: Block diagram of the deltic adaptive filter.

Schottky TTL and was constructed on a single double-
Eurocard circuit board using wire-wrap interconnect with
an ancilliary board to accommodate the two analogue-to-
digital converters required to supply the s(t) and d(t)
inputs. Total power consumption was less than 10 Watts
and the maximum sampling rate of the system was 10 kHz.
A photograph of the complete system is shown in figure 5.9.

5.4   Adaptive recursive filters

Recursive filters implemented using CCDs have in general
been unsuccessful due to the cumulative effects of charge
transfer inefficiency and dark current generation.   In
this section a design for an adaptive recursive filter is
considered which uses two transversal adaptive filters of
the type illustrated in figure 5.4(a).

Given the classic construction of a recursive filter with
feedback taps a_n and feedforward taps b_n the signal x(t)
in the delay line is given by:

$$x(t) = s(t) + \sum_{n=1}^{N} a_n x(t-n) \quad \dots\dots\dots\dots\dots\dots(5.2)$$

where s(t) is the input signal. The filter output is then:

$$y(t) = \sum_{n=1}^{N} b_n x(t-n)$$

$$= \sum b_n [s(t-n) + \sum a_n x(t-2n)]$$

$$= \sum b_n s(t-n) + \sum b_n [\sum a_n x(t-2n)]$$

$$= \sum b_n s(t-n) + \sum a_n c(t-n) \quad \dots\dots\dots\dots(5.3)$$
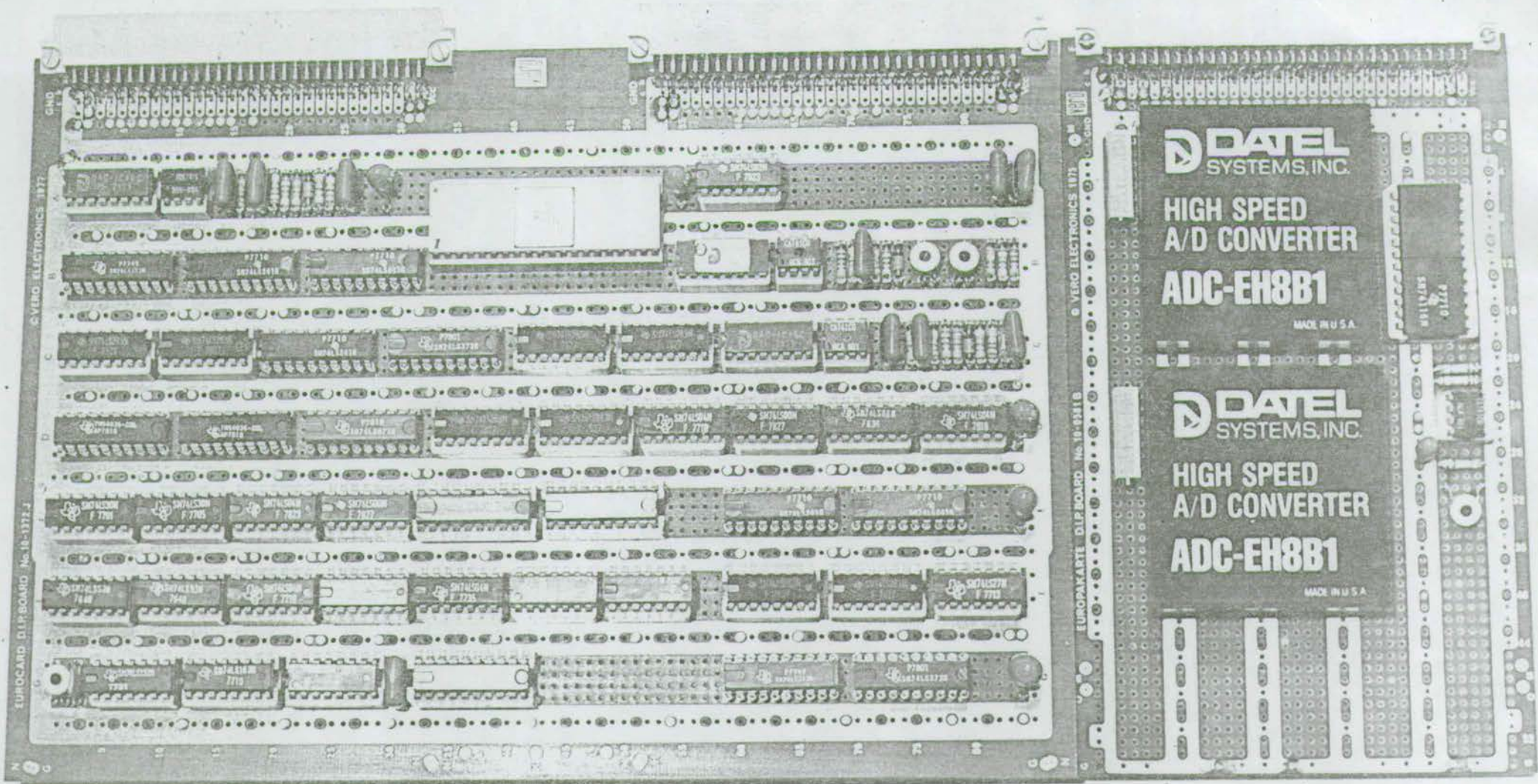
Figure 5.9: Photograph of the experimental deltic adaptive filter module.

The individual summations in equation (5.3) may be modelled by transversal filters. If we then define an error $e(t)$, as before, such that;

$$e(t) = d(t) - y(t) \quad \dots\dots\dots\dots\dots\dots\dots(5.4)$$

and use the Widrow LMS algorithm to update a_n and b_n:

$$b_n(t + 1) = b_n(t) + \frac{\partial e^2(t)}{\partial b_n(t)} \quad \dots\dots\dots\dots\dots(5.5)$$

$$a_n(t + 1) = a_n(t) + \frac{\partial e^2(t)}{\partial b_n(t)} \quad \dots\dots\dots\dots\dots(5.6)$$

then:

$$\frac{\partial e^2(t)}{\partial b_n(t)} = 2e(t)\frac{\partial y(t)}{\partial b_n(t)} \simeq 2e(t)s(t-n) \quad \dots\dots(5.7)$$

$$\frac{\partial e^2(t)}{\partial a_n(t)} = 2e(t)\frac{\partial y(t)}{\partial a_n(t)} \simeq 2e(t)y(t-n) \quad \dots\dots(5.8)$$

Substituting (5.7) and (5.8) into equations (5.5) and (5.6) yields the recursive LMS algorithm proposed by Feintuch[60]:

$$b_n(t + 1) = b_n(t) + 2\mu e(t)s(t-n) \quad \dots\dots\dots(5.9)$$

$$a_n(t + 1) = a_n(t) + 2\mu e(t)y(t-n) \quad \dots\dots\dots(5.10)$$

The gradient estimates given in equations (5.7) and (5.8) are at best crude approximtions but to proceed to higher order estimates yields a great increase in hardware complexity without any guaranteed increase in convergence reliability. Since the error surface associated with the recursive filter is complex, rather than a simple quadratic as in the case of the transversal filter, convergence is not guaranteed and convergence to local minima is possible. However results published by Feintuch[60] and Ahmed[61] tend to show that convergence does occur for a wide range of input conditions.

A block diagram of a system based on the ideas outlined
in this section is shown in figure 5.10 although this was
not constructed as an experimental system due to the lack
of the necessary equipment.
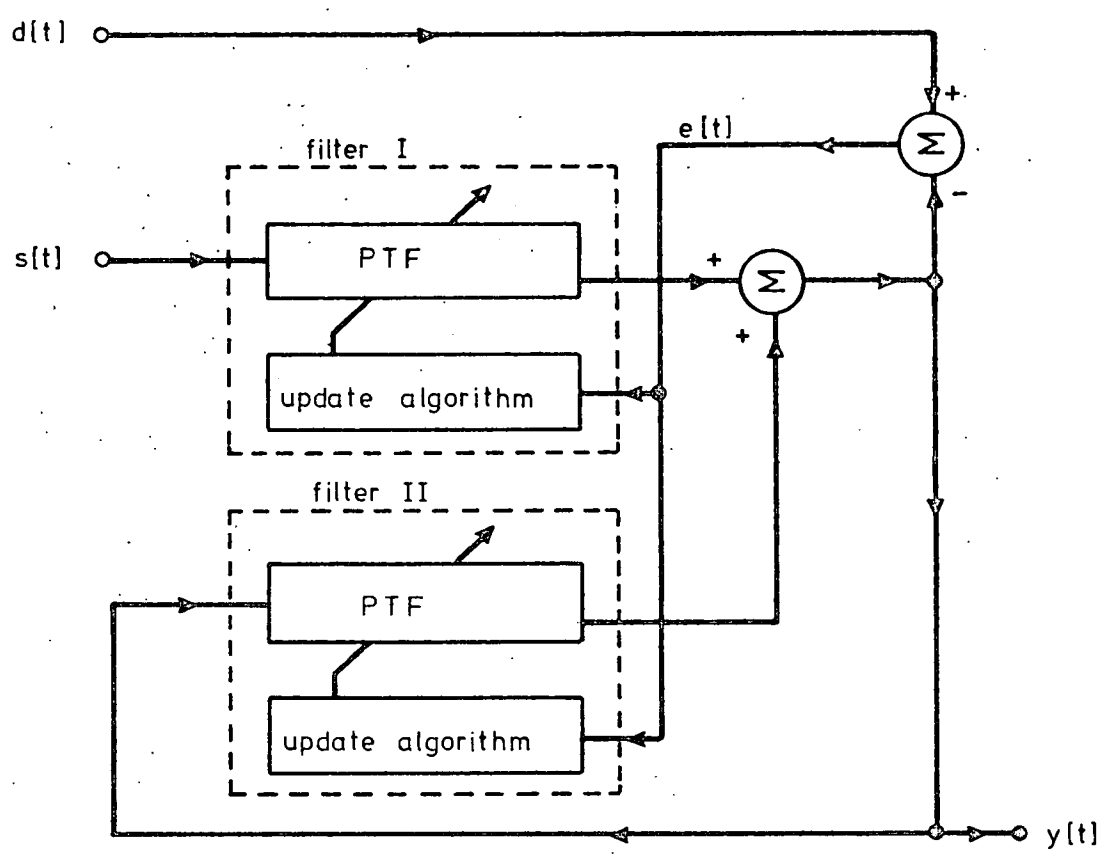
Figure 5.10: Schematic diagram showing the use of two adaptive transversal filters to form a single recursive adaptive filter.

CHAPTER 6: ADAPTIVE ERROR CORRECTION

6.1 Analogue system errors

The Widrow LMS algorithm discussed in the previous two chapters is guaranteed to converge, given certain constraints for a theoretically perfect system. However, in real systems error sources do arise and since our prototype use analogue CCD circuitry a range of errors peculiar to that implementation must be taken into account. In this chapter the primary objective is to evaluate the effects of these errors on an analogue CCD adaptive filter from a mainly theoretical point of view[62]. The errors which will be examined in the following sections are:

> (1) Charge transfer inefficiency;

> (2) Weight vector decay;

> (3) Multiplier errors; and

> (4) Gain errors in both the weighting circuits

and the CCD tap output circuits.

An additional factor which will be considered in the following section is the effect of updating the filter weight vector in the serial fashion described in section 5.2.

6.2 Charge transfer efficiency effects

In the CCD signal register charge is not transferred from
one cell to the next with complete efficiency; instead an
amount of charge ηQ is transferred (η is the transfer
efficiency) and an amount ξQ ($\xi = 1-\eta$) is left behind.
Therefore the actual signal dealt with by the filter
multipliers is $\underline{\xi}S$ where $\underline{\xi}$ is an n x n matrix of
inefficiency coefficients. Further the signal value used
in the update algorithm is corrupted by inefficiency
effects since it is taken from the last tap of the CCD.
The effective signal value used in the algorithm is then
$\underline{E}S$ where $\underline{E}$ is another square matrix of inefficiency
coefficients. Approximations to $\underline{\xi}$ and $\underline{E}$ ignoring terms
occuring before $s(t-n)$ are:

$$
\underline{\xi} =
\begin{bmatrix}
\eta & \xi\eta & \xi^2\eta & \xi^3\eta & \cdots\cdots\xi^{n-1}\eta \\
0 & \eta^2 & 2\xi\eta^2 & 3\xi^2\eta^2 & \vdots \\
0 & 0 & \eta^3 & 3\xi\eta^3 & \vdots \\
0 & 0 & 0 & \eta^4 & \vdots \\
\vdots & \vdots & \vdots & 0 & \vdots \\
0 & 0 & 0 & 0\cdots\cdots & \eta^n
\end{bmatrix}
$$

$$
\underline{E} =
\begin{bmatrix}
\eta^n & \binom{n}{n-1}\xi\eta^n & \binom{n+1}{n-1}\xi^2\eta^n \cdots \binom{2(n-1)}{n-1}\xi^{n-1}\eta^n \vdots \\
0 & \eta^n & \binom{n}{n-1}\xi\eta^n \cdots\cdots\cdots\cdots\cdots\vdots \\
0 & 0 & \eta^n \cdots\cdots\cdots\cdots\cdots\cdots\vdots \\
\vdots & & & \vdots \\
0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \eta^n
\end{bmatrix}
$$

The algorithm, taking into account these effects, then

becomes:

$$\underline{H}(t) = \underline{H}(t-1) - 2\mu\underline{X}(t-t_N) \ \underline{ES}(t_N) \ \{d(t_N) - \underline{S}^T(t_N)\underline{\xi}^T \ \underline{H}(t_N)\}\ldots$$

$$\ldots\ldots\ldots\ldots\ldots\ldots(6.1)$$

where, $N = n+1$ (n = the number of filter points used),

and $\quad t_N = t - t \bmod N$

$$\underline{X}(a) = \begin{bmatrix} \delta(a-n+1) & 0 & 0 \cdot \cdots \cdot 0 \\ 0 & \delta(a-n+2) & 0 \cdot \cdots \cdot 0 \\ \vdots & \vdots & \cdots \cdots 0 \\ \vdots & \vdots & \\ 0 & 0 \cdot \cdots \cdots \cdots \delta(a) \end{bmatrix}$$

$$\delta(a) = \begin{array}{l} 1, \ a = 0 \\ 0, \ a \neq 0 \end{array}$$

If times t', where t' is an integer multiple of N, only

are considered then (6.1) becomes:

$$\underline{H}(t') = \underline{H}(t'-N) - 2\mu\underline{ES}(t-N) \ \{d(t'-N) - \underline{S}^T(t-N) \ \underline{\xi}^T\underline{H}(t'-N)\}$$

which is the same as the parallel-update algorithm,

ignoring times t which are non-integer multiples of N.

Taking expected values:

$$E\{\underline{H}(t')\} = [\underline{I} + 2\mu\underline{ER}\underline{\xi}^T] \ E \ \{H(t'-N)\} \ -2\underline{EF} \ \ldots\ldots\ldots(6.2)$$

where,

$$\underline{R} = E \ \{\underline{S}(t) \ \underline{S}^T(t)\} \ , \ \text{and}$$

$$\underline{P} = E \ \{d(t) \ \underline{S}(t)\} \ \ldots\ldots\ldots\ldots\ldots\ldots(6.3)$$

assuming $\underline{H}$ is uncorrelated with $\underline{R}$.

$$E\{\underline{H}(t')\} = [\underline{I} + 2\mu\underline{ER}\underline{\xi}^T]^{(\frac{t'}{N}+1)} E\{\underline{H}(0)\} - 2\mu \sum_{i=0}^{\frac{t'}{N}+1} [\underline{I} + 2\mu\underline{ER}\underline{\xi}^T]^i \underline{EP} \ \ldots\ldots$$

$$\ldots\ldots\ldots\ldots\ldots(6.4)$$

$$\lim_{t' \to \infty} [\underline{I} + 2\mu \underline{E} \underline{R} \underline{\xi}^T]^{\frac{t'}{N} + 1} \to 0$$

if $\dfrac{-1}{n^n \lambda_{max}} < \mu < 0$

(where λ_{max} is the largest eigen vlaue of $\underline{R}$) and provided that $\underline{S}$ does not contain a periodic frequency component which is an integer multiple of 1/N.

Also

$$\lim_{t' \to \infty} \sum_{i=0}^{\frac{t'}{N}+1} [\underline{I} + 2\mu \underline{E} \underline{R} \underline{\xi}^T]^i = \frac{1}{2\mu} [\underline{E} \underline{R} \underline{\xi}^T]^{-1}$$

$$= -\frac{1}{2\mu} (\underline{\xi}^T)^{-1} \underline{R}^{-1} \underline{E}^{-1} \dots$$

$$\dots \dots \dots \dots \dots \dots (6.5)$$

Substituting (6.5) into (6.4)

$$\lim_{t' \to \infty} E\{\underline{H}(t)\} = (\underline{\xi}^T)^{-1} \underline{R}^{-1} \underline{E}^{-1} \underline{E} \underline{P}$$

$$= (\underline{\xi}^T)^{-1} \underline{R}^{-1} \underline{P}$$

The filter output is given by:

$$y = \underline{S}^T \underline{\xi}^T \underline{H} \quad \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (6.6)$$

and substituting the value of $\underline{H}$ into equation (6.6)

$$y = \underline{S}^T (\underline{\xi}^T) (\underline{\xi}^T)^{-1} \underline{R}^{-1} \underline{P}$$

$$= \underline{S}^T \underline{H}_W$$

That is, the effective weight vector is the same optimum Weiner solution, $\underline{H}_W$ as that found in the case where no errors are taken into account. A more detailed treatment of transfer inefficiency correction is given in reference 34. It should be noted that the inversion of $\underline{\xi}^T$ is only meaningful where the input signal is repetitive or when the optimum impulse response is significant over a time interval less than that available in the PTF.

This example serves to illustrate that from the point of
view of ultimate convergence, the serial algorithm may be
treated in the same way as the parallel update version
with the condition that $\underline{S}$ should not have a periodic
frequency component which is an integer multiple if 1/N.
Therefore, in all further analysis convergence properties
will be treated as for a parallel update system.

6.3 Reference signal decay

In the systems used in these experiments (i.e. the two
CCD 64-point prototype modules) decay of the reference
weight vector $\underline{H}$ may be viewed in two parts. First decay
of the weight vector stored in the filter device, assuming
that the weight vector is stored digitally for use in the
update algorithm. The algorithm then becomes

$$\underline{H}(t+1) = \underline{H}(t) - 2\mu\underline{S}(t)\left[d(t) - \underline{S}^T(t)k\underline{H}(t)\right] \quad \ldots\ldots(6.7)$$

then
$$\lim_{t' \to \infty} E\{\underline{H}(t+1)\} = \frac{1}{k}\underline{H}_W \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(6.8)$$

$$\text{given} \quad -\frac{1}{k\lambda_{max}} < \mu < 0$$

That is, following the same analysis as in section 6.2
the final weight vector is a multiple of the Weiner
weights. In most cases this is acceptable and has no
significant effect on system performance when decay rates
($\simeq 1$ V/sec) available on monolithic designs are used.

However, when a decay term is considered in the weight

vector used in the update algorithm, assuming the filter reference decay to be negligible, the algorithm then becomes

$$\underline{H}(t+1) = k\underline{H}(t) - 2\mu\underline{S}(t)\left[d(t) - \underline{S}^T(t)\,\underline{H}(t)\right] \quad\ldots\ldots(6.9)$$

Then, again following the analysis of section 6.2:

$$\lim_{t' \to \infty} E\{\underline{H}(t+1)\} = \left(\frac{(k-1)}{2\mu}\,\underline{I} + \underline{R}\right)^{-1}\underline{P} \quad\ldots\ldots\ldots\ldots(6.10)$$

where $-\dfrac{k+1}{2\lambda_{max}} < \mu < \dfrac{1-k}{2\lambda_{max}}$

That is, the converged weight vector is biased and not the optimum solution $\underline{H}_W$. An unexpected effect also exists since the minimum error occurs when μ takes its largest value of $k + 1/2\lambda_{max}$ (which also has the effect of maximising the weight variance). Although this effect can lead to gross errors in the system performance when k assumes a significantly small value it also causes the convergence speed to be maximised with, in some cases, such as adaptive line tracking[63], is desirable.

6.4 Multiplier errors

In the construction of the WM2100 programmable filter device a single MOS transistor is used to perform the multiplication of signal and weight values at each filter point. In normal operation these (transistor) multipliers are operated in two cycles where the signal on the CCD tap is presented to the transistor gate during the first cycle and a constant bias level equivalent to the signal zero

point is forced on the gate during the second time cycle.
The first cycle produces as output given by:

$$y = s.h + k.h - \tfrac{1}{2}h^2 \quad \dots\dots\dots\dots\dots\dots\dots(6.11)$$

where s and h are the signal and weight values at the
relevant CCD tapping point, and k is a constant. This
equation may be obtained from the drain/source current
equation for a MOST operating in the unsaturated region.
During the second operation cycle s = 0, therefore:

$$y_2 = k.h - \tfrac{1}{2}h^2 \quad \dots\dots\dots\dots\dots\dots\dots\dots(6.12)$$

Subtracting (6.12) from (6.11) yields the required result

$$y = s.h$$

In later design (WM2110) it was not possible to perform
this two cycle zeroing operation. Therefore a d.c. error
is generated at the output every time the weight vector
$\underline{H}$ is changed. It is difficult to analyse this error in
the same way as was done in the previous two sections,
mainly because of the term involving h^2 in equation (6.11).
However the earlier prototypes described in section 5.2
used a degraded form of multiplier zeroing such that in
the second cycle of operation an offset d.c. bias was used
on the multiplier gate rather than the actual signal zero
level. This yields the output;

$$y(t) = \underline{S}^T\underline{H} - \underline{K}^T\underline{H} \quad \dots\dots\dots\dots\dots\dots\dots(6.13)$$

Substituting this value of y(t) into the Widrow LMS
adaption algorithm yields a solution for the expected
value of $\underline{H}$ given by:

$$E\{H(t)\} = (\underline{R} - E\{\underline{S}\,\underline{K}^T\})^{-1}\underline{P} \quad \dots\dots\dots\dots\dots(6.14)$$

This result is somewhat difficult to analyse heuristically

but it can be seen to deal only with the d.c. terms
present in the system. The action of this particular
correction mechanism can be viewed from three separate
instances:

(1) $\underline{S}(t)$ has no d.c term: In this case the
d.c. terms in $\underline{H}$ have no valid contribution to $y(t)$ and $\underline{H}_W$
therefore has no net d.c. term. The product $\underline{K}^T\underline{H}$ then
has a zero result.

(2) $\underline{S}(t)$ has a d.c. term and $d(t)$ also has a
d.c. term: In this case the d.c. level of $\underline{H}$ is shifted
until the result of multiplying the d.c. terms of signal
and reference cancels the $\underline{K}^T\underline{H}$ error term.

(3) $\underline{S}(t)$ has no d.c. term but $d(t)$ does: Here
the net d.c. term on $d(t)$ should appear as an error on
$e(t)$. However because of the $\underline{K}^T\underline{H}$ term the filter can
always find a d.c. term in the output, it therefore
reproduces a d.c. output in error.

These three cases also apply to the situation where the h^2
term is taken into account (equation (6.11)). The
overall result is that the filter can still be relied upon
to produce a valid output despite the multiplier errors,
except in the third case quoted above. Therefore true
baseband operation of the system as a whole is suspect.

6.5 General error correction mechanism

In general it can be said that if a source of error can be
designated by a matrix $\underline{G}$ such that:

$$y = \underline{S}^T\underline{G}\ \underline{H} \quad \text{or} \quad y = \underline{S}^T\underline{H}\ \underline{G}$$

and it is possible to find an inverse matrix $\underline{G}^{-1}$ then the adaptive filter will create a weight vector given by:

$$\underline{H} = \underline{G}^{-1}\underline{H}_W \quad \text{or} \quad \underline{H} = \underline{H}_W\ \underline{G}^{-1}$$

such that $y = \underline{S}^T\underline{H}_W$.

A trivial example of this is where the error source is a set of gain mismatches along the row of tap amplifiers, in which case $\underline{G}$ is a square matrix where the diagonal terms represent the tap gains and all other terms are zero.

More complex error effects present greater difficulty in analysis and the solutions given in the previous sections are at best approximations to the action of the filter.

An additional bonus which should be true from the previous analysis relates to the temperature stability of the filter. Since drifts in d.c. bias levels are long term effects in a CCD this means that even for a serialised update scheme the filter adaption should be a great deal faster than significant drifts in PTF parameters due to temperature changes. This would mean that the CCD-based adaptive filter would be free from serious temperature drift problems which means, in addition, that setting up procedures may be greatly simplified.

6.6 Experimental verification of error correction

In order to verify the error correction mechanisms

mentioned above it would be necessary to isolate each individual type of error on its own. Unfortunately this is impossible to do and checking that error correction actually takes place must be inferred by a suitable interpretation of filtering results. For instance, tap gain errors are quite easy to pick out from an impulse response of one of the PTF devices but these errors tend to be easily confused with transfer inefficiency effects which are only manifested clearly when long delays are used with a clear high frequency component (a step or pulse) in the impulse response. On the other hand, weight biasing due to tap weight decay produces a clearly visible effect which will be described in the next chapter and the correction of multiplier errors is clearly demonstrated by the consistent maintenance of correct d.c. bias levels at the output of the filter.

The experimental evidence presented in this section†
demonstrates the ability of a pseudo-adaptive filter to compensate for tap gain errors and transfer inefficiency when specifically trained to do so[64]. The experimental system used in this case was a WM2100 64-point PTF which was controlled by a Zilog Z80 microprocessor system. The microprocessor program was to force the impulse response of the PTF to resemble, as closely as possible, an ideal

impulse response $\underline{H}^*$ by iteratively adjusting the tap weights at the PTF reference input. The algorithm used was the Widrow LMS algorithm, but since the input signal was a single pulse repeated every 64 sample periods two special effects come into play:

(1) The signal is zero at 63 points out of 64, which means that the LMS algorithm is automatically serialised because 63 signal data points out of 64 carry no information. Therefore a serial version of the LMS algorithm, as described in section 5.2 may be used without any loss in convergence time or other error effects. In this case in fact the algorithm serialisation is effectively implemented by holding one signal value in every 64 and continuously monitoring the error at the output[64].

(2) The significant 64th signal sample is known to always have value one. This means that it is not in fact necessary to sample the CCD signal at all. In fact the incoming signal pulse is generated by the micro-processor system.

The adaption algorithm may then be written as:

$$\underline{H}(t + N) = \underline{H}(t) + \mu\left[r(t) - y(t)\right] \quad \ldots\ldots\ldots(6.15)$$

where $r(t)$ is the ideal impulse response stored in the microprocessor, and $y(t)$ is the impulse response of the PTF.

A block diagram of this system is shown in figure 6.1 and
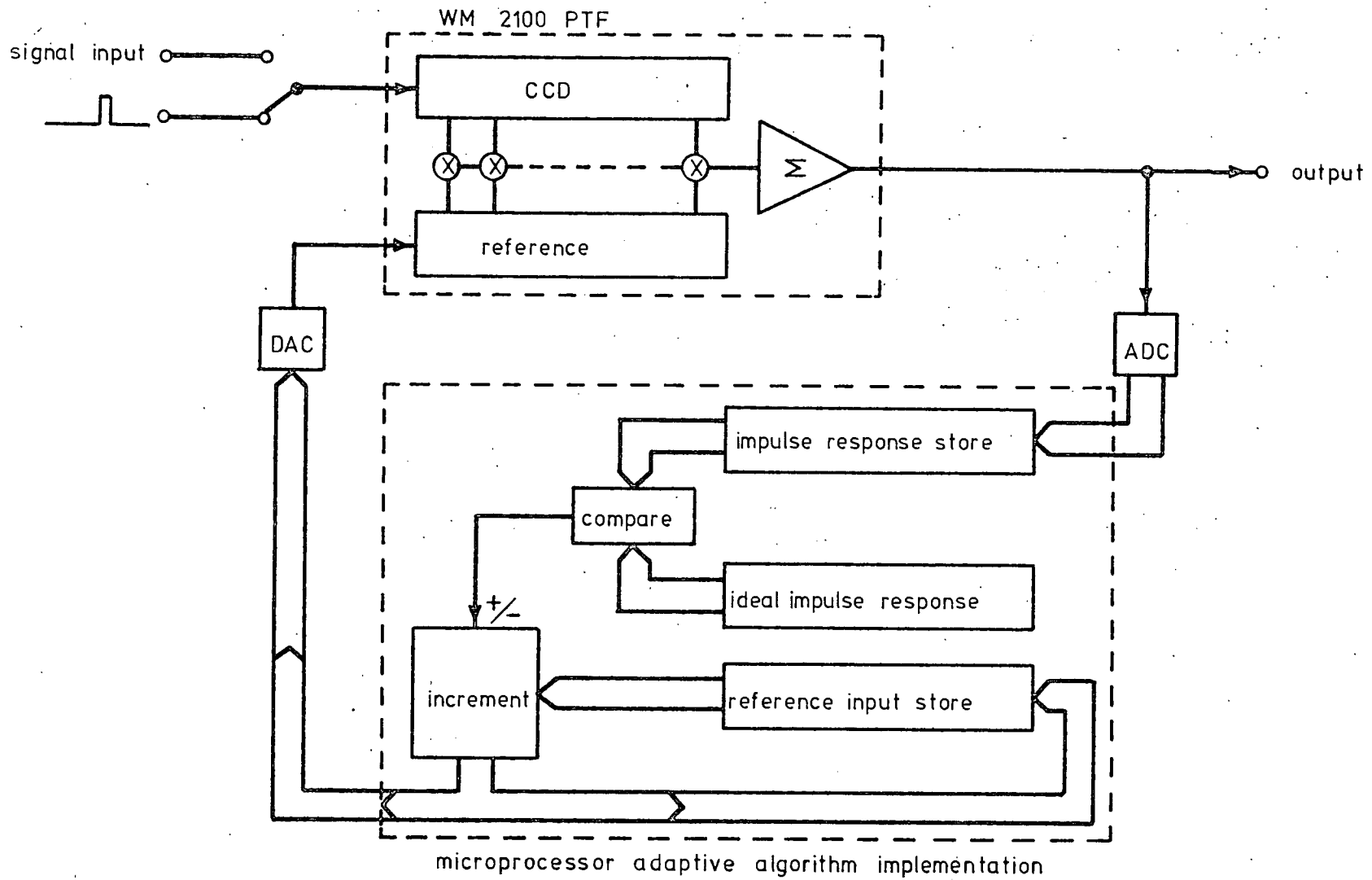
Figure 6.1: Block diagram of the microprocessor-controlled CCD adaptive filter.

a typical experimental result is shown in figure 6.2.
Figure 6.2(a) shows the input signal to the PTF and (b) is
the signal initially supplied at the reference input.
Figure 6.2(c) shows the corresponding output y(t) (before
adaption) in which tap gain variation is clearly visible
(the uneven nature of the linear part of the ramp) and
transfer inefficiency manifests itself as a smearing of
the stepped sdge of the ramp. Figure 6.2(d) shows the
signal appearing at the CCD 64th tap. Figures 6.2 (e)
and (f) show the result obtained after the system has been
allowed to adapt where (e) shows the revised reference
input and (f) is the y(t) output. It can be seen quite
clearly from figure 6.2(f) that the adaption process has
to a large degree, managed to compensate for errors due to
tap gain errors and transfer inefficiency since the ramp
is now linear with a clearly defined sharp return step.
Also clearly demonstrated is one non-working tap, the low
value in the first ramp section. This was due to a fault
on the PTF device used for this experiment. Even here it
can be seen that some attempt was made to raise this
output value by raising the corresponding reference input
value to its maximum value. It should be noted here that
the PTF was being operated as a correlator and y(t) is
therefore the time reverse of the reference input.

The result quoted here, although it does demonstrate one
aspect of adaptive error correction, does not necessarily
prove that correction of device errors is inherent to an

(a) s(t) input

(b) reference input

(c) impulse response

(d) CCD output

[i] before adaption

(e) reference input

(f) impulse response
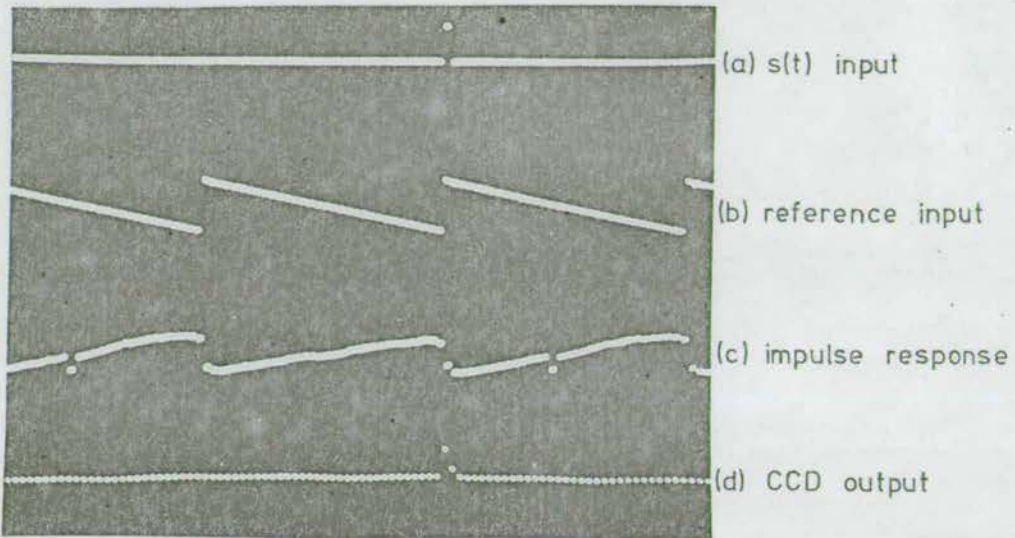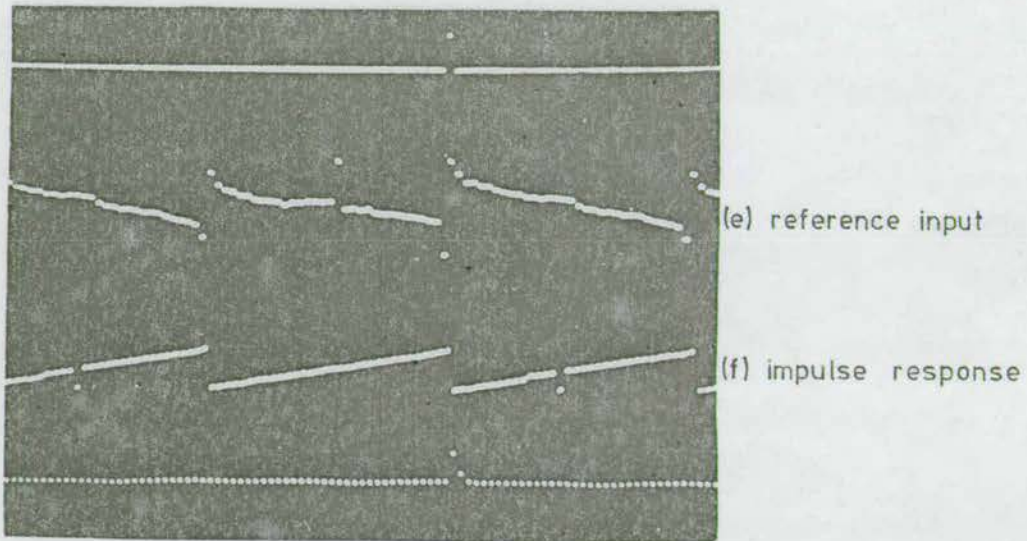
[ii] after adaption

Figure 6.2: Result demonstrating the ability of the
adaptive system to compensate for gain
and inefficiency errors in the PTF device.

adaptive filter operating normally.   In the succeeding

chapters however evidence of weight vector optimisation

in the course of normal adaptive filter operation will be

demonstrated using the prototype systems described in

chapter 5.

CHAPTER 7:   EXPERIMENTAL RESULTS

In this chapter experimental results are presented which are representative of the operation of the three prototype adaptive filters described in chapter 5.   It has been attempted, as far as possible, to relate these results to the systematic errors described in the last chapter and the individual advantages of each of the circuit techniques are highlighted with reference to the specific modes of operation in which the results were generated.   Due to the difficulty in isolating specific error sources a high reliance on appropriate interpretation of qualitative experimental data has been necessary; though this does have the advantage that the resulting conclusions have a much greater relevance to the practical utilisation of adaptive sub-systems.

7.1   Convergence time characteristics

Rigorously, the convergence time of a linear LMS adaptive filter is dependent on the maximum eigen value of the input covariance matrix $\underline{R}$[48].   However a reasonable approximation for the convergence time constant τ has been derived by Widrow et al[49] to yield the result:

$$\tau = \frac{n}{4\mu \, tr\underline{R}} \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots(7.1)$$

where n is the number of weights in the transversal filter, $\underline{R}$ is the input covariance matrix, and $tr\underline{R}$ is the

trace of $\underline{R}$ (the input signal power).

For instance if the input to both s(t) and d(t) input ports is a sinusoid with an integral number of full cycles within the filter delay time then:

$$tr\underline{R} = \sum \sin^2 \omega_c$$
$$= \sum \tfrac{1}{2} - \tfrac{1}{2} \sum \cos 2\omega_c \quad \dots\dots\dots\dots\dots\dots\dots(7.2)$$

The second term in equation (7.2) sums to zero, given the constraints already placed on the input signal.

Therefore, using a filter with 64 taps:

$$tr\underline{R} = 32$$

Therefore, from equation (7.1):

$$\tau = \frac{1}{2\mu}$$

For a linear LMS adaptive filter having a convergence factor $\mu = 2^{-7}$:

$$\tau = \frac{1}{0.016} = 62 \text{ samples}$$

The time taken to converge to the minimum error state is usually taken to be 4τ.[49] Therefore the total convergence time for the case just stated is 240 samples.

However, when the algorithm used is of the zero-forcing type used to implement the DELTIC prototype then the input signals seen by the update algorithm are effectively square waves. This means that the dependence on input power is removed and:

$$tr\underline{R} = \sum_{n=1}^{N} 1 = 64$$

that is $\qquad \tau = \tfrac{1}{4}\mu$

For the digital prototype with a convergence factor of 2^{-7} this yields a convergence time of approximately 120 samples. The result shown in figure 7.1 demonstrates the typical convergence properties of the DELTIC adaptive filter. Here the s(t) input was a sinusoid with a period of 16 samples (i.e. the signal fulfils the necessary criterion of having an integral number of full cycles within the filter delay time). The d(t) input is initially grounded so that the filter is trained to reject the s(t) input. The d(t) input then becomes a replica of s(t) and the filter is allowed to converge to this new training signal. The y(t) output in figure 7.1(c) shows the expected gradual rise from a zero level to the required sinusoidal shape. However it is easier to measure the convergence time from the error output shown in figure 7.1(d). Here the exponential decay of the envelope of e(t) is clearly visible and the convergence time may be seen to be close to that stated above from theoretical considerations.

A peculiarity of the zero-forcing algorithm is that the convergence time of the filter when trained as a square wave matched filter should be the same as that stated for the above case because the apparent input power levels are the same. Figure 7.2 shows the typical convergence characteristics for this case, and from this it may be seen that the convergence time is roughly equivalent to that in figure 7.1.
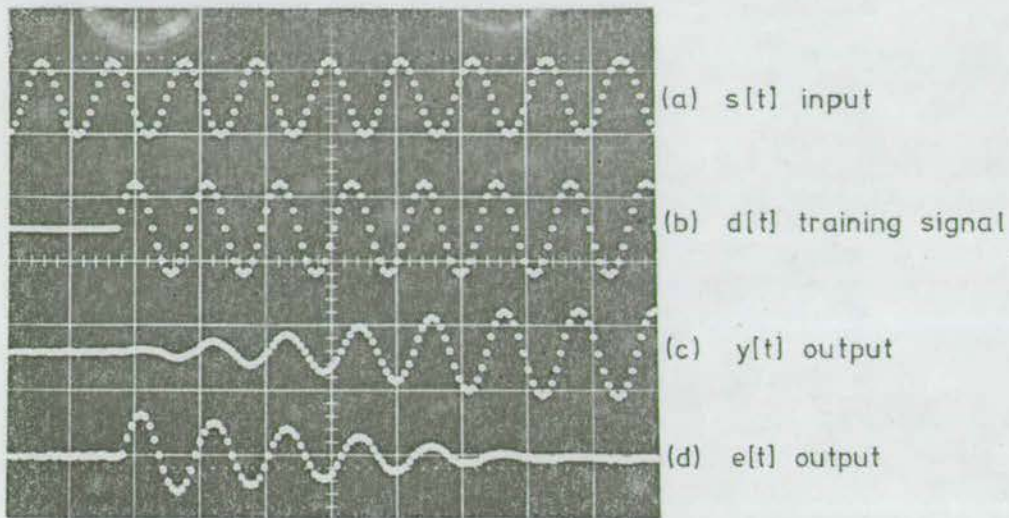
Figure 7.1: Characteristic convergence properties
of the DELTIC adaptive filter with
sinusoidal inputs.
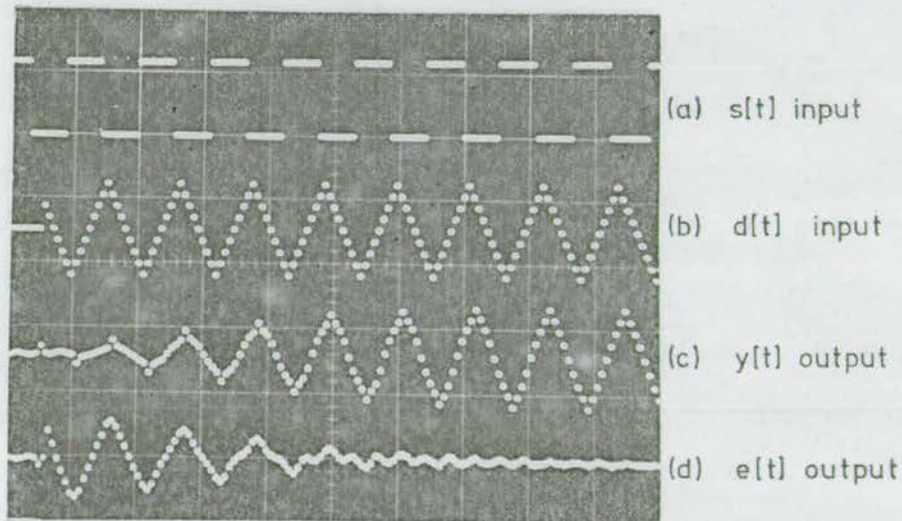
Figure 7.2: Convergence properties of the DELTIC
prototype when trained as a square
wave matched filter.

The convergence properties of the CCD-based prototypes are somewhat complicated by the serial nature of the convergence algorithm which means that convergence should occur approximately 65 times slower than for the case stated above. Experimental verification of this convergence property is complicated by the analogue nature of the convergence factor setting in the two CCD prototype modules which makes it difficult to be sure of the exact convergence factor being used. However, these problems were solved in a later CCD-based prototype described in the next chapter and a rigorous evaluation of convergence properties may be found in section 8.2.

Figure 7.3 demonstrates a typical convergence characteristic for the CCD adaptive filter prototype having analogue weight storage. In this case the s(t) and d(t) inputs were identical at all times and at one time they were both grounded and then they were both switched to a sinusoidal input. The effect of the grounded inputs on the analogue storage adaptive filter is that the tap weights tend to decay to a negative d.c. level and there is a resulting excess d.c. error when the input becomes a sinusoid due to multiplier errors. This is clearly evidenced in figure 7.3(a) which shows the e(t) output during convergence. The need to restore proper d.c. bias levels in this case results in an even longer convergence time than expected. In fact the total convergence time was almost twice as long as that expected from theory.
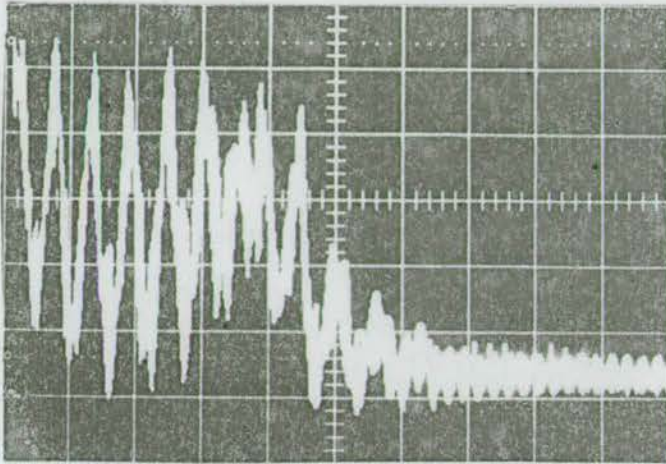
(a) error output during convergence

timebase - 100 msec/div
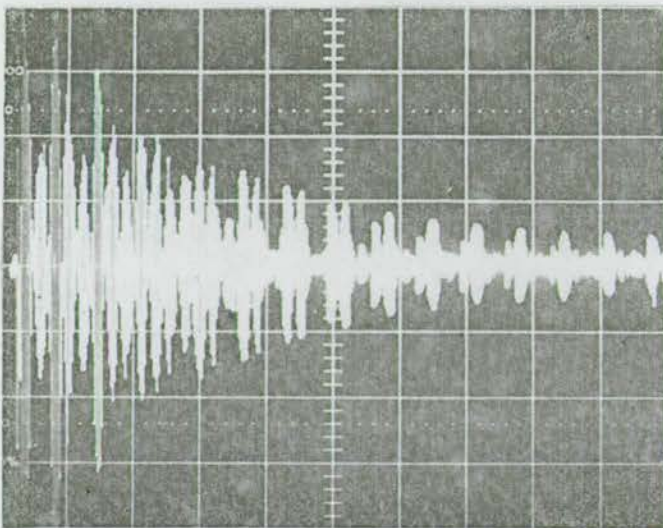
(b) weight increments during convergence

Figure 7.3: Convergence characteristics of the CCD adaptive filter prototype with analogue weight storage when the input signals are sinusoidal.

The signal shown in figure 7.3(b) is the actual increment supplied (serially) to each tap weight during convergence. In this case the exponential nature of error decay is once again clearly visible. The fact that the filter can still converge with these multiplier generated d.c. errors is an empirical proof of the error correction mechanism stated in section 6.4.

The serial nature of the update algorithm in the CCD-based systems has one further effect on convergence characteristics. That is that the weights diverge for periodic inputs having a frequency which is an integral multiple of the inverse of the total update period (65 samples in this case). This imposes a comb frequency response on the filter with 64 nulls between the Nyquist frequency and the minimum possible input frequency (a signal having one cycle within the filter delay time in this case). The nulls are however extremely sharp, as a signal having a period of 64 sample tends to show near optimum convergence characteristics and apart from these nulls the frequency response (after convergence) is flat over the region just stated. In many cases this characteristic is clearly undesirable and a solution to this problem is proposed in chapter 8.

A point which should be noted here is that the alternative serialised update scheme described in section 6.6 does not suffer from this problem of weight divergence at

particular frequencies. This is because the e(t) output is continuously sampled in the alternative scheme meaning that the cross-correlation matrix $\underline{P}$ is fully defined for all d(t). In the case of the system described here $\underline{P}$ is not fully defined if d(t) is synchronous with the update period.

7.2 Cancellation characteristics

It is the ability of an adaptive filter to cancel unwanted components of a signal which is often quoted as its most useful characteristic. Indeed the degree to which the filter is capable of cancelling an unwanted signal component is often quoted as a figure of merit called the adaptivity χ where:

$$\chi = 10 \log_{10} \frac{\overline{[d'(t)]^2}}{\overline{[e'(t)]^2}} \text{ dB}$$

and d'(t) is the component of d(t) which it is desired to cancel and e'(t) is the remaining unwanted component at the e(t) output.

The cancellation performance of an adaptive filter may be conveniently demonstrated by a very simple experiment where the incoming signal comprises two sinusuids at approximately the same level, one at a frequency f and the second at frequency 2f. It is this second signal component which we desire to cancel and therefore the s(t) input is supplied with a version of this sinusoid which is of the

wrong phase and amplitude to subtract coherently from d(t). The task of the filter is then to form an impulse response which modifies the phase and amplitude of s(t) such that y(t) subtracts coherently from d(t) leaving the required signal component, at frequency f, on the e(t) output.

The result shown in figure 7.4 demonstrates this type of operation using the CCD adaptive filter prototype with digital weight updating circuitry. In this case the convergence factor μ was adjusted to its minimum value of approximately 2^{-10} to yield the maximum possible degree of cancellation. Examination of the d(t) and e(t) spectra shown in figure 7.4(e) and 7.4(f) reveals that the adaptivity in this case was approximately 50 dB. This result was generated for the slowest convergence time case and decreasing the convergence time (i.e. increasing μ) resulted in a corresponding drop in the adaptivity. This is because of the increasing interference, by the excess error e(t), with the weight values as μ increases (see section 5.3.2). A more detailed experimental treatment of this characteristic is given in chapter 8.

In section 6.3 it was shown theoretically that the weight vector error was minimised for the shortest convergence time in the case where the weight values constantly decay. This is the case for the CCD prototype module having analogue circuitry to calculate and store the weight

(a) s[t] input

(b) d[t] input

(c) y[t] output

(d) e[t] output

(e) d[t] input spectrum

(f) e[t] output spectrum
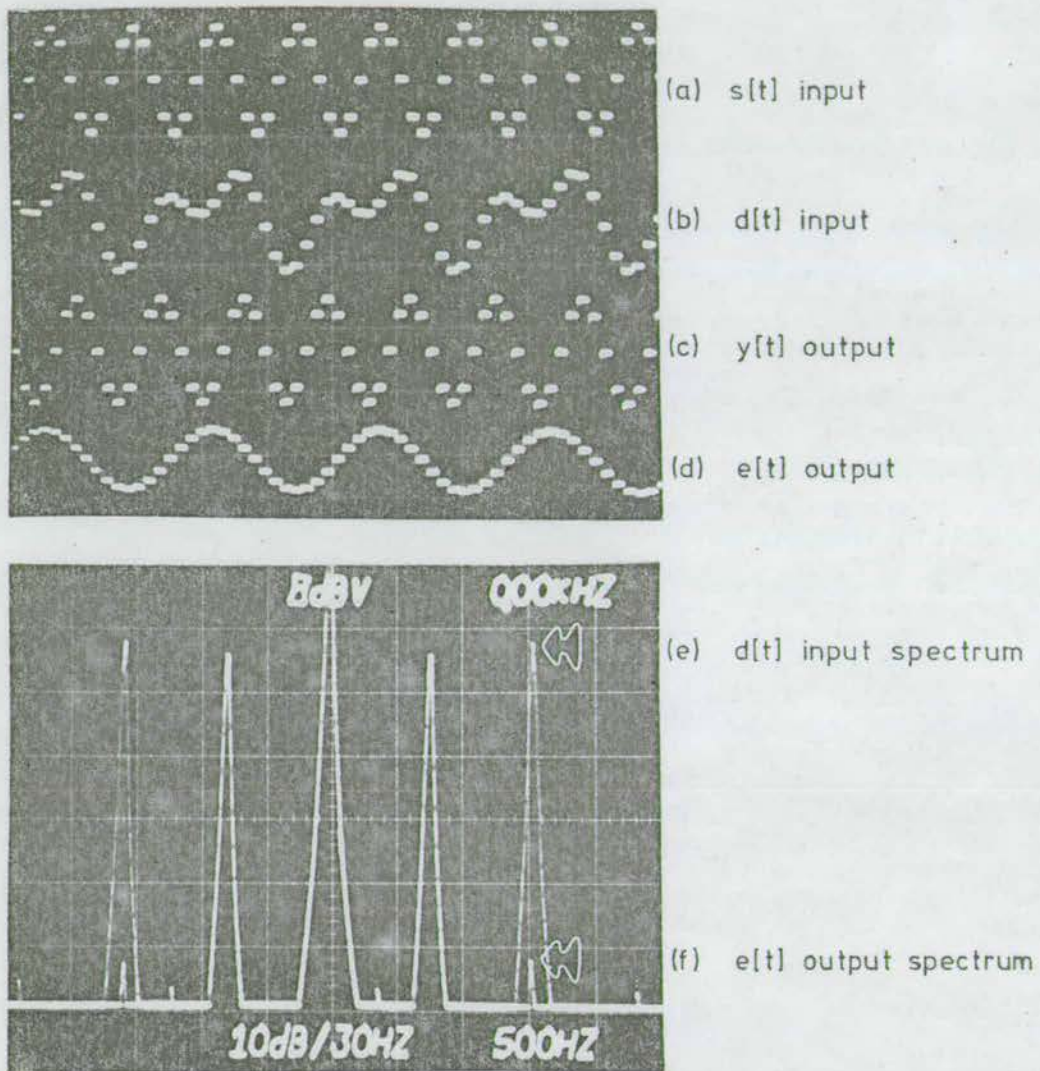
Figure 7.4: Performance of the analogue CCD adaptive
filter prototype with digital weight
storage as a canceller.

vector $\underline{H}$. The result shown in figure 7.5 is for the case where μ was maximised, yielding an adaptivity of approximately 25 dB. Increasing the convergence time, in this case, was found to degrade the adaptivity, as predicted in section 6.3. The adaptivity figure of 25 dB does in fact correspond to that obtained for the fastest convergence case using the digital weight storage prototype. It is thought that the poor result quoted here was due, principally, to excessive weight decay values of about 1 V/sec achieved using the discrete storage capacitors and switches in this prototype. Achievable storage times on monolithic devices are a factor of 10 better than this and would yield a proportionally better performance.

The same set of test signals applied to the all-digital DELTIC prototype yielded an adaptivity of about 30 dB at best (see figure 7.6). This is due entirely to the poorer weight resolution available on this prototype. To achieve a better result using this type of processor would involve a large increase in hardware complexity and it was felt that this was not justified in meeting the needs of this project.

However, the few results discussed in this section have shown that the needs of the adaptive processor in terms of the weight vector updating circuitry are in fact relatively complex and a flexible processor offering

(a) d[t] input spectrum

(b) e[t] output spectrum
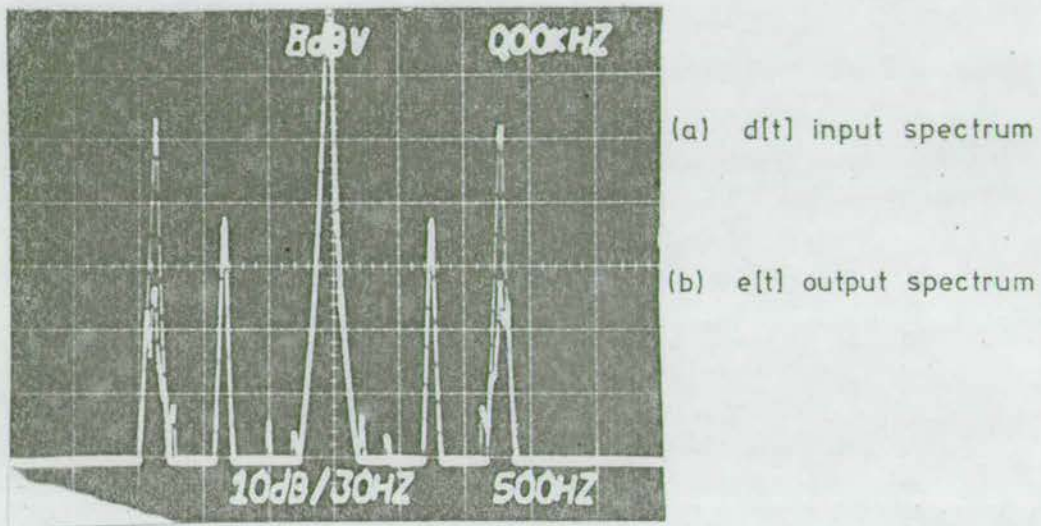
Figure 7.5: Cancellation performance of the CCD prototype
with analogue weight storage.

(a) d[t] input spectrum

(b) e[t] output spectrum
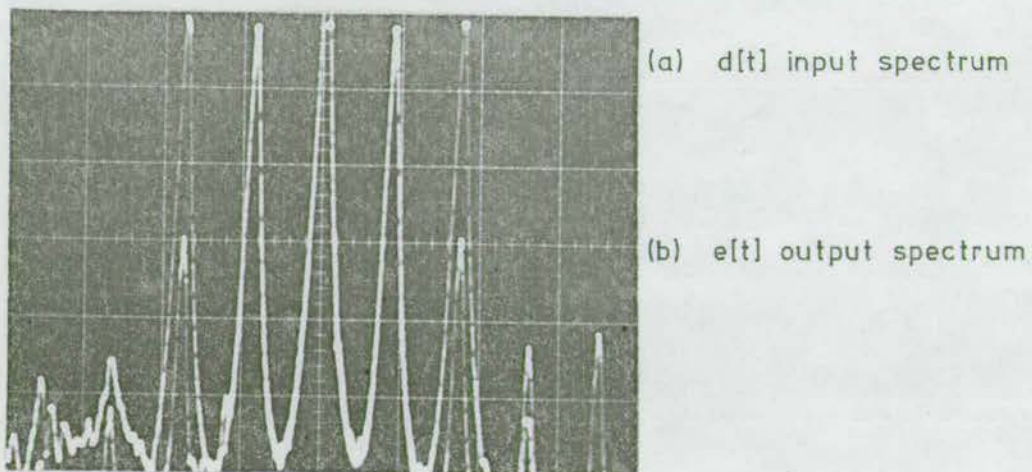
Figure 7.6: Cancellation performance of the DELTIC
prototype adaptive filter.

either fast convergence times or very accurate weight
vector solutions must have a very high weight vector
resolution with the facility of easy and precise
controllability of the convergence factor.

7.3 Matched and inverse filtering

For any filter the impulse response may take two extreme
forms where the total power content of the weight vector
is either maximised or minimised: these two cases are
matched filtering and inverse filtering, respectively.
The matched filter, by maximising the output signal power,
is specifically aimed at increasing signal-to-noise ratio
and therefore increasing the probability of signal
detection. The inverse filter, on the other hand, is
intended to counteract distortion where noise is not a
major problem. Both of these filter types prove to be
useful in evaluating adaptive filter performance and some
selected results illustrating these performance aspects
are presented in the next two sections.

7.3.1 Matched filtering characteristics

One of the major problems in evaluating the performance
of an adaptive filter is the difficulty in predicting the
exact form which the weight vector will take under any
given set of circumstances. This problem does not arise
where the tap weights are stored on analogue sample-and-

holds since the weight values always tend to decay towards zero and the filter therefore always chooses an impulse response in which every weight is as close as possible to zero. It will be found that this always corresponds to the theoretically expected solution. However, where the weight storage is digital no such decay exists and an important constraint has therefore been removed. For instance if the adaptive filter is trained as a sine wave matched filter with s(t) and d(t) being identical sinusoids (no other signals being present at either input port) the expected weight vector is the matching sine wave. Although this indeed happens with analogue weight storage it is not the case with the filters using digital weight storage. This is simply because an inadequate amount of information has been given to the filter and it therefore has many degrees of freedom in its choice of weight vector; this means that the converged weight vector tends to take what looks like a totally arbitrary form. This was in fact found to be the case in tests carried out with the prototype adaptive filters described here.

Therefore, in order to define fully the expected weight vector the adaptive filter must receive information covering the entire operating frequency spectrum at the s(t) input, this condition is fulfilled for example in the test arrangement described in section 6.6. One example of this is when the filter is specifically trained to form

the matched filter response to a linear f.m. sweep signal
(a 'chirp') swept from the Nyquist frequency to d.c. where
the total duration of the signal is equivalent to the
filter time delay. Given that the signal just described
is applied to the s(t) input of the filter and the
training signal applied to the d(t) input is a single
pulse positioned at the end of this signal sequence, then
the impulse response of the filter should be the time
reverse of the s(t) input signal.

The result shown in figure 7.7 demonstrates the operation
of the CCD adaptive filter with digital weight storage
as a chirp matched filter. It can be seen from a
comparison of figure 7.7(b) and (c) that the matching
of the d(t) input and the y(t) output after convergence
is good and in fact the residual sidelobe level in this
case was measured to be less than -25 dB. The trace
shown in figure 7.7(d) is the impulse response of the
device for this case which was generated by freezing the
weights (reducing μ to zero) and putting a pulse on the
s(t) input of the filter. The impulse response shown here
is the expected time reverse of the chirp shown in figure
7.7(a).

The example shown in figure 7.8 is the same result
generated using the DELTIC prototype module. Again the
y(t) output shows a high degree of correspondence to the
training signal and this serves to demonstrate that the

(a) s[t] input

(b) d[t] input (training signal)

(c) y[t] output

(d) impulse response
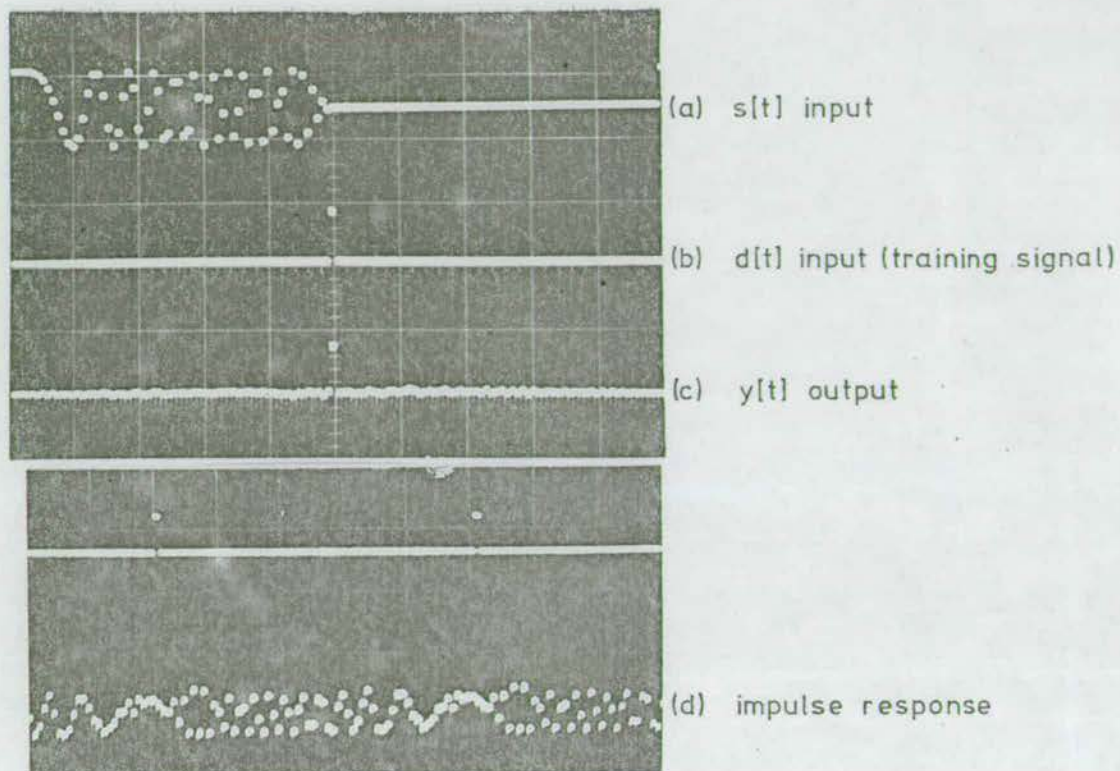
Figure 7.7: Example of matched filter training for the CCD prototype with digital update circuitry.

(a) s[t] input

(b) d[t] input

(c) y[t] output

(d) e[t] output
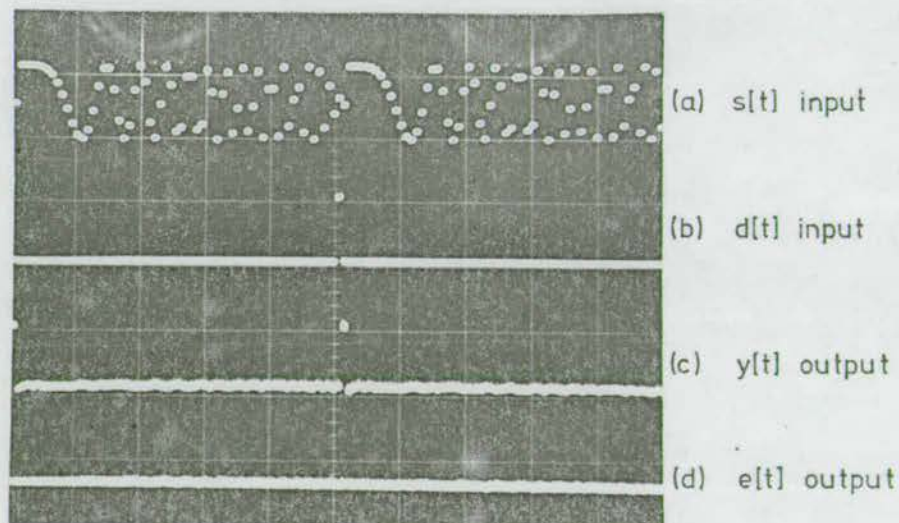
Figure 7.8: Chirp matched filtering using the DELTIC prototype adaptive filter.

straightforward filtering operation of the adaptive filter is not affected in the same adverse way as the cancellation performance by coarseness in the convergence factor.

A peculiarity of the adaptive filter operation is that the filter is capable of shaping the weighting window function to suit the particular set of training signals being used. For instance, if the s(t) input is a full bandwidth chirp with a sine start phase and the weight function is a similar chirp with a cosine start phase the theoretical filter output is dominated by a dual pulse of the form 1, 0,-1 with slowly decaying sidelobes at alternate sample positions on either side of this dual pulse. However if this s(t) input is used with a training signal which contains only the dual pulse and no sidelobes, as shown for the DELTIC prototype in figure 7.9, then the output specified can be achieved by applying a weighting function to the weight vector, as shown in figure 7.9(d). A similar result occurs, for instance, when the same s(t) input is used but the training signal, d(t), is a pulse of duration 4 samples. The result shown in figure 7.10 shows the operation of the CCD prototype (with digital weight storage) with these input conditions. Figure 7.10 (b) shows that the long pulse is indeed reproduced using the weight vector shown in figure 7.10(c) which is the time reversed input chirp which has essentially been low pass filtered to tailor the frequency response of the

(a) s[t] input

(b) d[t] input [training signal]

(c) y[t] output

(d) e[t] output

(e) weight vector input
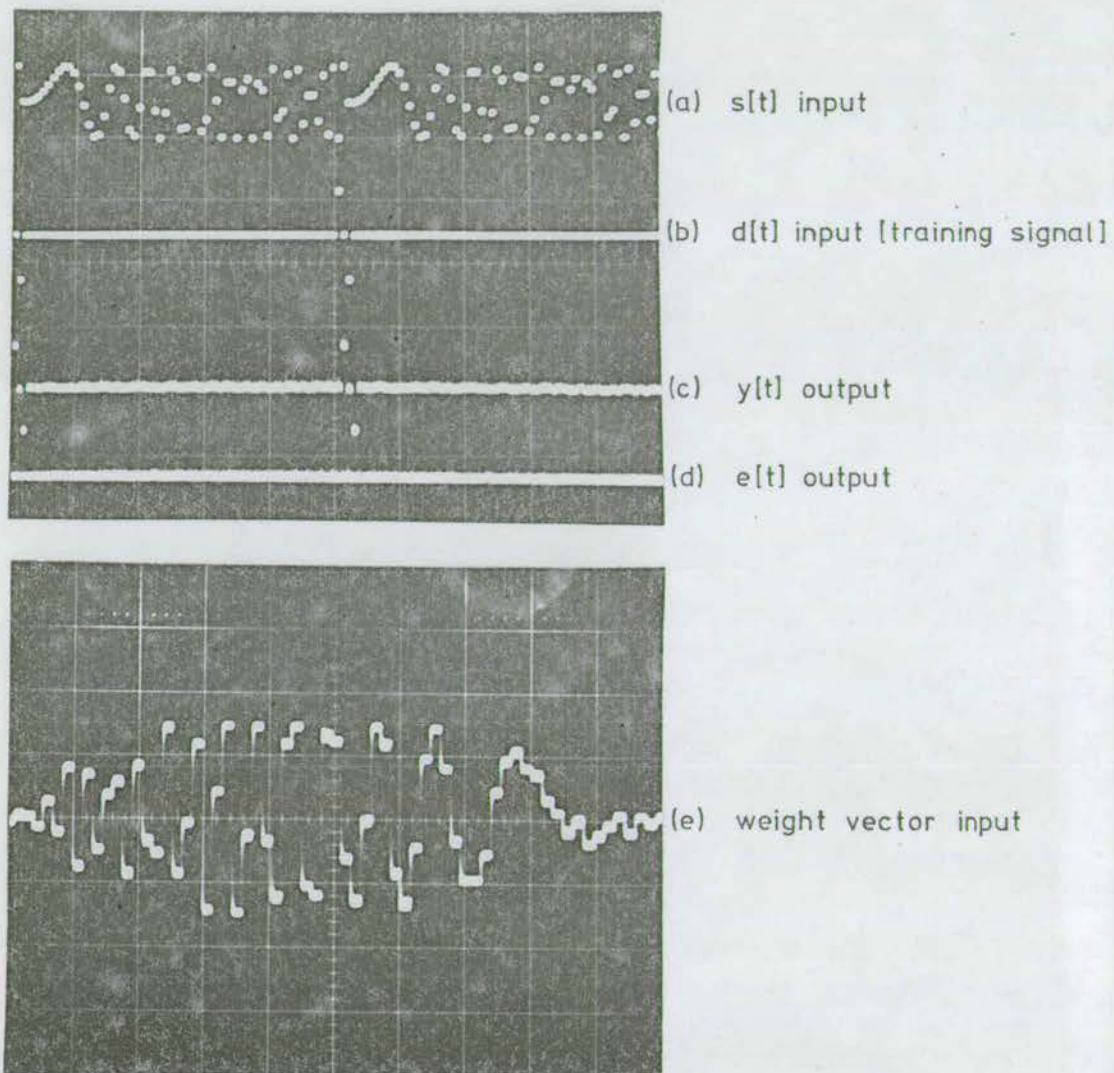
Figure 7.9: Derivation of optimum windowing by using
a training signal stripped of sidelobes
in the DELTIC adaptive filter.

(a) s[t] input

(b) y[t] output
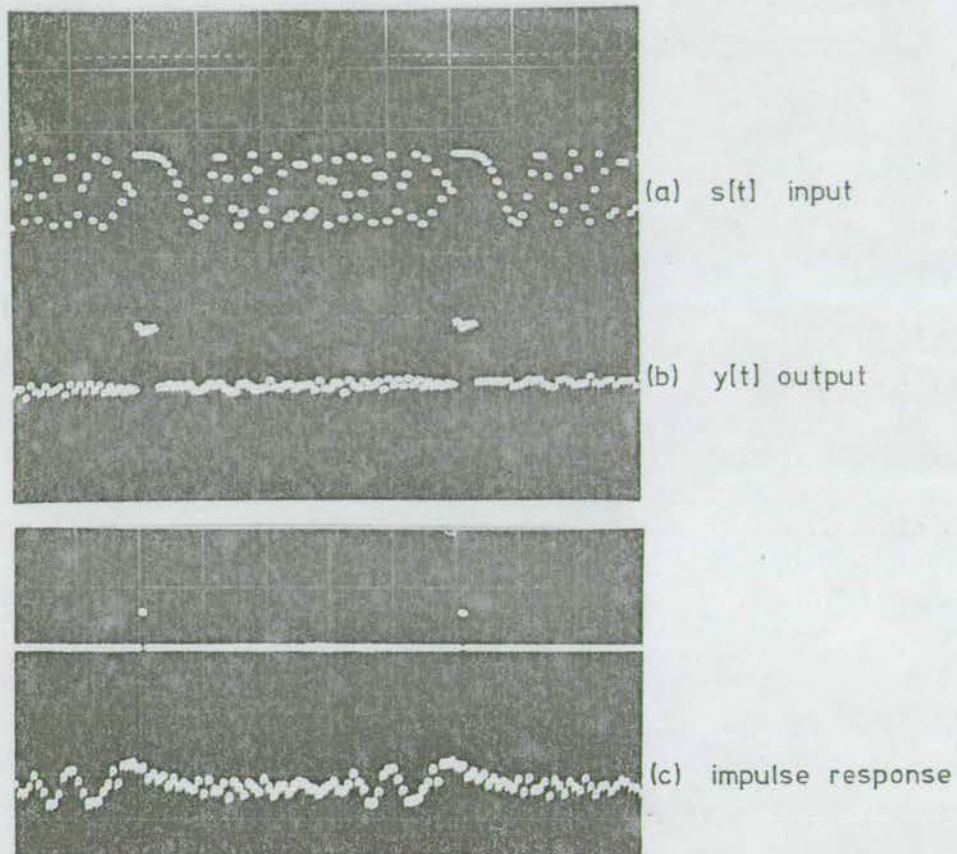
(c) impulse response

Figure 7.10: Derivation of an optimally shaped impulse response using the CCD prototype adaptive filter.

filter.

What these results demonstrate is the inherent ability of
an adaptive filter to generate an optimum weight vector
for any input and desired output combination automatically.
The above statement only holds true on the conditions that
such a weight vector actually exists and that the s(t)
input covers the entire frequency spectrum of the filter.

7.3.2   Inverse filtering characteristics

It was found in the last section that in operation as a
matched filter there was little difference between the
operational characteristics of the analogue and digital
adaptive filter prototypes. In this section the
alternative inverse filter structure is considered, where
weight vector power levels are considerably lower than
those considered in the last section.

The result shown in figure 7.11 demonstrates the action of
both the DELTIC prototype system and the CCD system with
digital weight storage. The inputs to s(t) and d(t) in
this case were identical square waves. The optimum
weight vector in this case is a series of pulses of
alternating sign at a spacing corresponding to half a
cycle of one of the square waves. The trace shown in
figure 7.11(c) is the output of the DELTIC filter, which
shows that although the general shape of the d(t) signal

(a) s[t] input

(b) d[t] input

(c) y[t] output

(d) weight vector
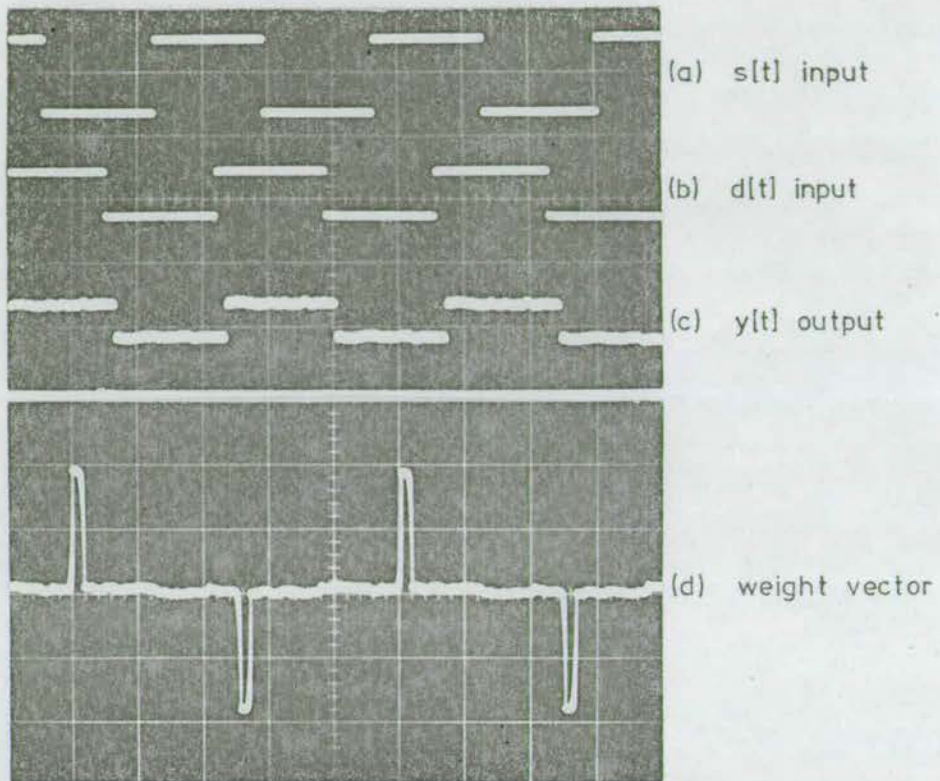
(i) DELTIC system result

(e) s[t] input

(f) y[t] output
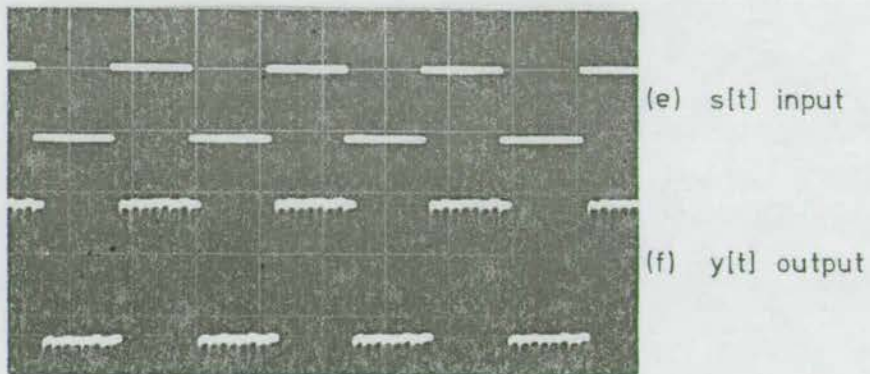
(ii) CCD system result

Figure 7.11: Result showing the use of both prototypes
in an inverse filter mode of operation.

is replicated considerable error still exists in the form
of ripple on the flat part of the waveform. However the
weight vector for the DELTIC system (shown on an expanded
scale in figure 7.11(d)) does show a good correspondence
to that predicted. The corresponding output for the CCD
filter shown in figure 7.11(f) demonstrates a much higher
quality of output than was the case with the DELTIC
filter with little residual error present on the flat part
of the output waveform. Also the sharp definition of the
square wave edges tend to confirm that transfer
inefficiency compensation is in fact taking place.

This fundamental difference in performance may be easily
explained in terms of the basic design of the filters
themselves. Both filters were specifically designed as
matched filters, but this fact has a different effect in
each case:

      (1) the DELTIC filter: Here the output word
length of the digital multiplier is 16 bits and
accumulation of 64 serial multiplications results in a
maximum word length of 22 bits. However it is clearly
impossible to deal with all 22 bits at the output and they
must therefore be truncated in some way since the actual
output word length used must match the word length of the
incoming d(t) signal. In the system described in chapter
5 the resolution of the ADC on the d(t) input was 8 bits
and it was necessary, therefore, to choose 8 bits of the
output word to represent the final result. The 8 bits

actually chosen were the top, i.e. most significant, 8 of the 22 bit result. This is ideal for matched filter performance since the result is always guaranteed to be significant in this region due to the high power content of the weight vector. However, in operation as as inverse filter only a small number of weighting points actually contribute to the final result and the magnitude of the final result will be relatively small in the top 8 bits being used here. This was the case in the result quoted in figure 7.11 and the poor result shown in figure 7.11(c) results directly from this range deficiency. In order to restore the proper range to the filter output it is necessary to choose 8 output bits in a position which is lower in the significant order of the 22 bit accumulator output. This means physically rewiring the system (or installing sets of 8 bit buffers) and it is also necessary in this case to construct an overflow detection and saturation circuit to prevent erroneous outputs occurring due to 'false' overflow.

(2) the CCD filter: This filter was also constructed as a matched filter, but here the limiting factor on operation is the noise level in the device, which is fixed. This is because each multiplier contributes a certain level of noise to the output even if it has no significant weight value assigned to it. However, for the filter considered here, the total dynamic range (taken as the ratio of maximum output level over maximum spurious noise level) is in the region of 75 dB.

This means that relatively low power weight vectors such as that used in figure 7.11 may be used by a relatively simple alteration in output gain level. It is clearly shown in figure 7.11 that the residual dynamic range of the CCD filter is still much greater than that for the equivalent case with the DELTIC filter.

It is important to note here that the operation of a practical adaptive filter in the inverse filter mode is critically dependent on the power level of the d(t) input. For instance, if the level of the d(t) input is too high then the pulses described in figure 7.11(d) will be at saturation level and a large residual error will still exist. This means that the filter will try to achieve a better result (in terms of decreasing the e(t) power level) by using more of the weight vector points. This is illustrated in figure 7.12 for the DELTIC module. It can be seen here that the d(t) input was in fact many times the maximum possible amplitude for inverse filter operation and as a result the adaptive filter has formed a matched filter response in order to reduce the error power as much as possible. Although this occurs with square wave inputs it would not always be the case since signals with white frequency spectra tend to have very narrow pulse-like matched filter outputs. Therefore, for many input signals, a compromise is reached which is neither inverse nor matched filter response. This result will be of particular interest in considering adaptive filter

(a) s[t] input

(b) d[t] input

(c) y[t] output

(d) e[t] output
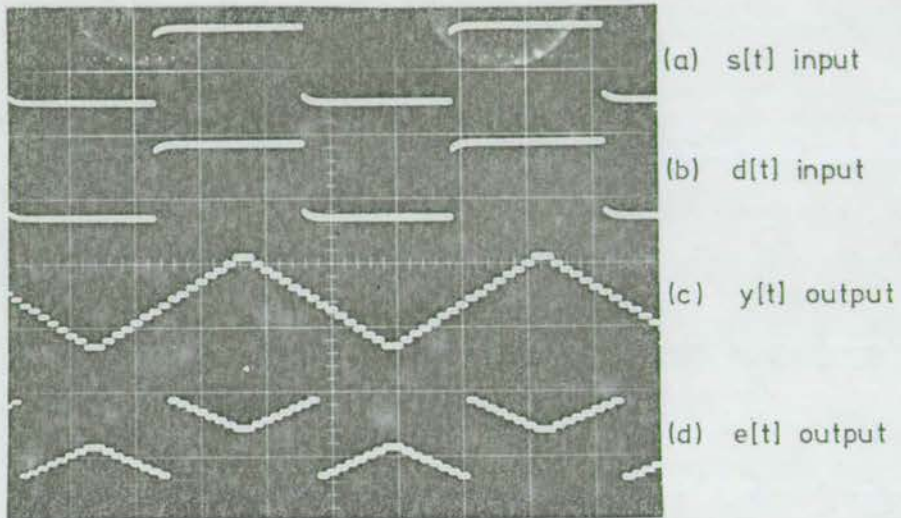
Figure 7.12: Result for the DELTIC filter with an inverse
filter training signal with excess power
content forcing a matched filter response.

operation in the self-tuning mode of operation (see section 7.4).

Operation in this mode is particularly useful in the application to adaptive equalisation of communications channels, e.g. switched telephone lines. One such experiment was carried out to test the operation of the CCD adaptive filter in this particular application. The experimental configuration is shown in figure 7.13 where the signal source generates a simple 2-level digital pseudo-random code which is passed directly through a simulated 5 miles of telephone cable. The sort of signal received at the channel output is shown in figure 7.14(a) with corresponding eye pattern shown in figure 7.14(d). The signal is clearly very heavily distorted and the eye pattern shows it to be only marginally detectable. The asymmetry of the eye pattern in figure 7.14(d) is due to the existance of a small d.c. term in the test signal. For the purpose of this experiment the d(t) input was taken as being a delayed (and undistorted) version of the channel input (figure 7.14(b)). This in fact does serve as a valid test arrangement since it is possible to set up the adaptive filter at the start of transmission by sending a known data sequence for a short time to allow the tap weights to converge and thereafter freezing the weights or using a decision-directed feedback technique[66] to compensate for slow drifts in channel characteristics. The output of the adaptive filter after convergence is
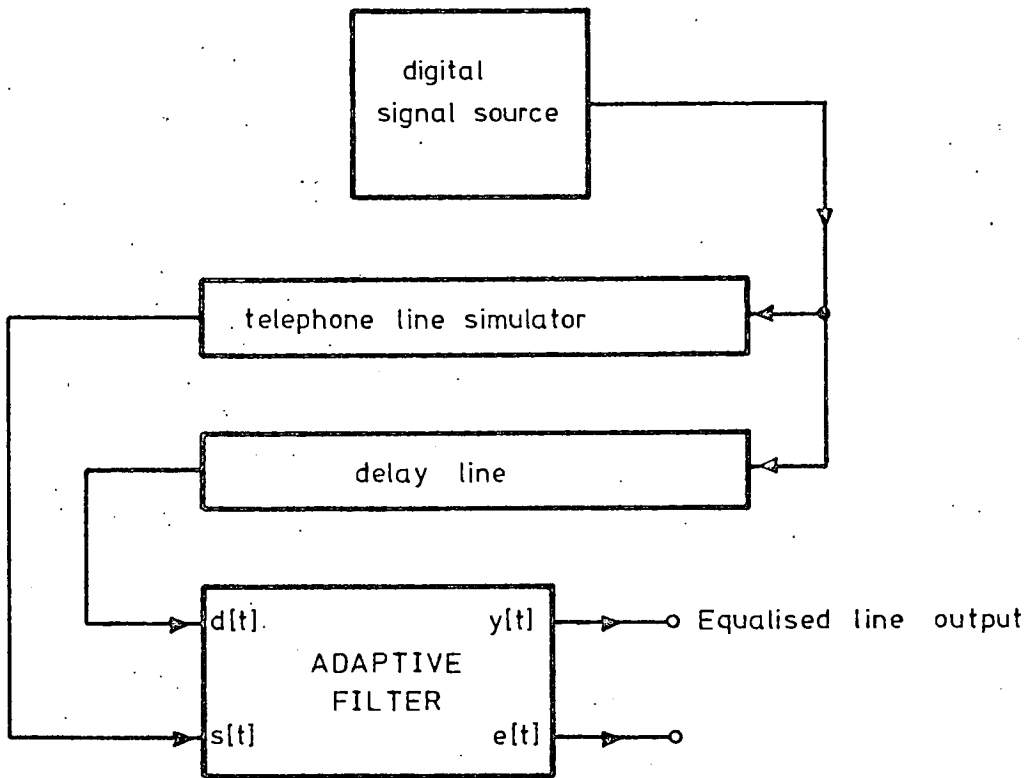
Figure 7.13: Experimental configuration used to demonstrate telephone line equalisation using the CCD-based adaptive filter.

(a) line output

(b) d[t], training signal

(c) y[t] output

(d) line output eye pattern

(e) y[t] output eye pattern
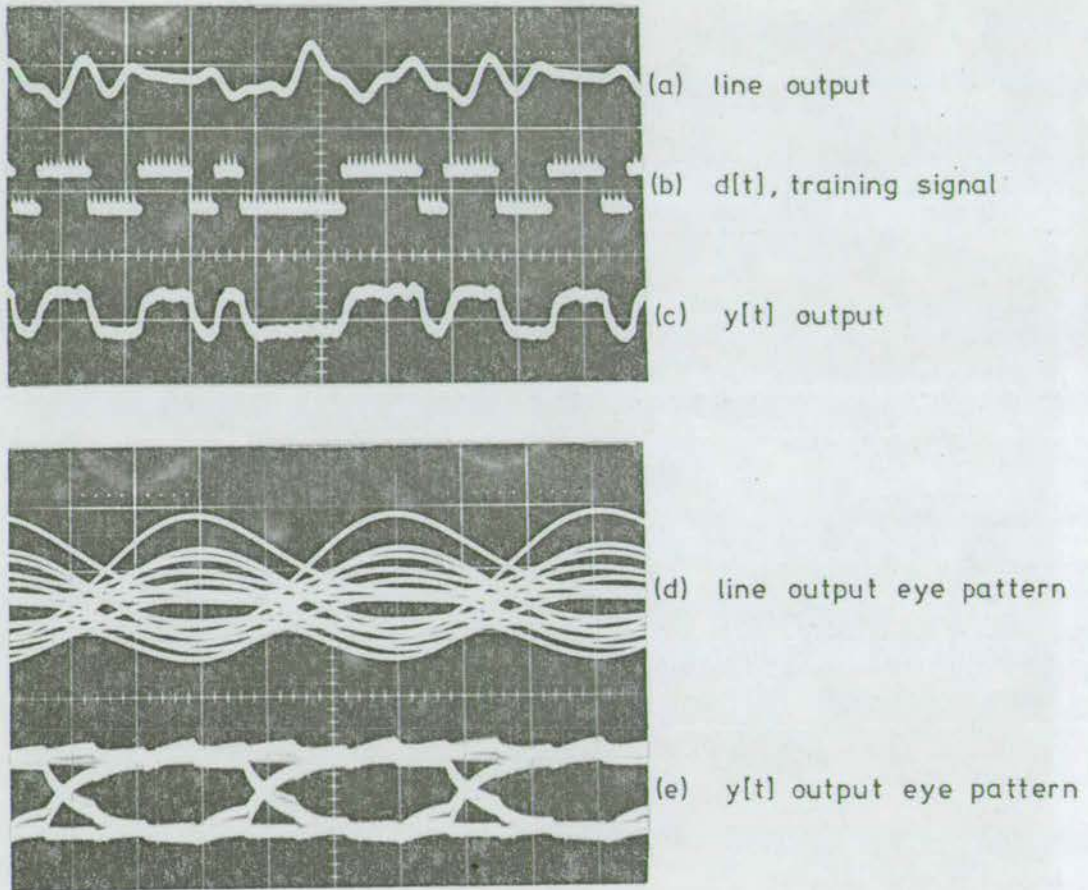
Figure 7.14: Result demonstrating adaptive
line equalisation.

shown in figure 7.14(c) with a corresponding eye pattern in figure 7.14(e). The filter output has been low pass filtered here purely in order to clarify the edges in the eye pattern. It can be seen here that a considerable improvement in detectability has been achieved with the zero crossings much better defined and the overall eye opening is greatly improved. It is clear that channel equalisation is one area where CCD adaptive filters could make a major impact.

7.4 Self-tuning filter characteristics

In the last section the performance of the adaptive filter in the two extreme cases of matched and inverse filtering were considered with reference to specific training signals. The self-tuning filter[48] on the other hand needs no external training signal since it uses the same signal as both s(t) and d(t) input, relying on the filter delay to decorrelate unwanted non-periodic signal components. The basic operation of this class of filter has already been discussed in section 4.3. A typical result generated using this filter configuration is shown in figure 7.15 where the input signal (figure 7.15(a)) is a sinusoid with added random noise. At the y(t) output (figure 7.15(c)) the sinusoid is reproduced, stripped of the noise which is left at the e(t) output (figure 7.15 (d)). Frequency spectra of the s(t) input and the y(t) output are shown in figures 7.15 (e) and (f) showing the

(a) s[t] input

(b) d[t] input

(c) y[t] output

(d) e[t] output

(e) spectrum of s[t]

(f) spectrum of y[t]
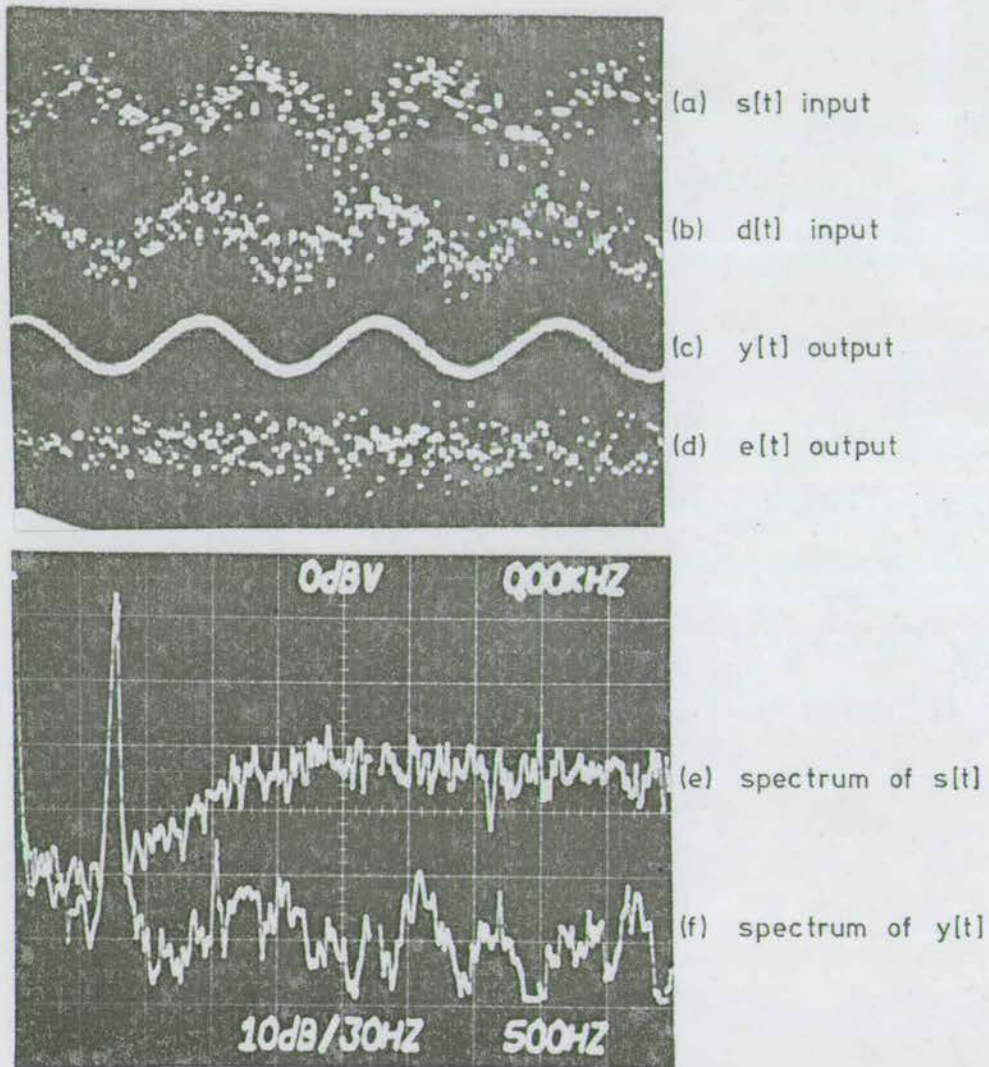
Figure 7.15: Result showing the operation of the CCD adaptive filter as a self-tuning filter.

average noise suppression on the y(t) output to be about 25 dB. This type of filter has many possible applications in removing unwanted noise on periodic tones or, alternatively, for cancelling unwanted tones on non-periodic signals, for example: whistle on high frequency radio reception or for removing musical interference on speech signals.

7.5 Summary

The results presented in this chapter have demonstrated the ability of the CCD adaptive processor to accomplish many of the classical adaptive filtering tasks as well as, and in some cases better than, an equivalent digital processor. One of the more important aspects which has not been mentioned before in this chapter is the stability of the CCD system. Despite continuous operation over a period of several months in environments involving temperature changes of at least $20^{\circ}C$ no observable change in operational characteristics of the prototype filters took place, despite the fact that the PTF itself was known to be particularly susceptible to temperature drift. This is due entirely to the stabilising effect of the adaptive loop itself which automatically compensated for bias changes due to temperature drift. Apart from this type of error compensation, compensation of errors due to multiplier operation and transfer inefficiency have also been seen to be achieved.

CHAPTER 8:   256-POINT ADAPTIVE FILTER

Although the results presented in chapter 7 demonstrated the basic feasibility of using analogue CCD/MOS technology in the realisation of adaptive filters the systems described suffered from several shortcomings. The most successful of the three prototypes described in chapter 5 was the CCD module using digital tap weight update but this was limited in performance by 3 major defects:-

(1)   Weight divergence at certain periodic input frequencies due to the serial mode of update.

(2)   Bandwidth reduction due to the necessity to perform an analogue to digital conversion within each sample period.

(3)   Lack of accurate controllability of the convergence factor, meaning that a rigorous comparison of convergence performance with theoretical prediction was impossible.

In this chapter details of a 256-point adaptive filter prototype based around the WM2110 PTF (described in chapter 2) are given. This module was designed with the intention of solving the problems stated above, therefore yielding a more flexible system.

8.1 256-point system implementation

A basic block diagram of the experimental 256-point adaptive filter system is shown in figure 8.1 and detailed circuit diagrams for this system may be found in Appendix A. The structure of this prototype differs in 4 major ways from the CCD systems described in chapter 5.

(1) The clipped LMS algorithm is used.

(2) The clipped signal is delayed further at the CCD output to allow time for ADC settling.

(3) The update period is varied between 257 and 258 samples.

(4) The convergence factor is digitally controlled

The update mode was however still of the serial type described in chapter 5, to reduce the amount of digital hardware required. The first two points listed above are directly related to the proposed increase in bandwidth for this system. The major bandwidth limitation on the previous CCD prototype was that the multiplication of $e(t)$ and $s(t-n)$ was performed using an analogue circuit and then the result was converted to digital form. This meant that one analogue to digital conversion had to take place within each sample period since $s(t-n)$ changes for each successive sample. However, $e(t)$ is constant (for this module) for 257 or 258 sample periods and by converting $e(t)$ to digital form, before multiplication with $s(t-n)$ a much longer ADC settling time is available. It would not be desirable to then convert $s(t-n)$ to the same type
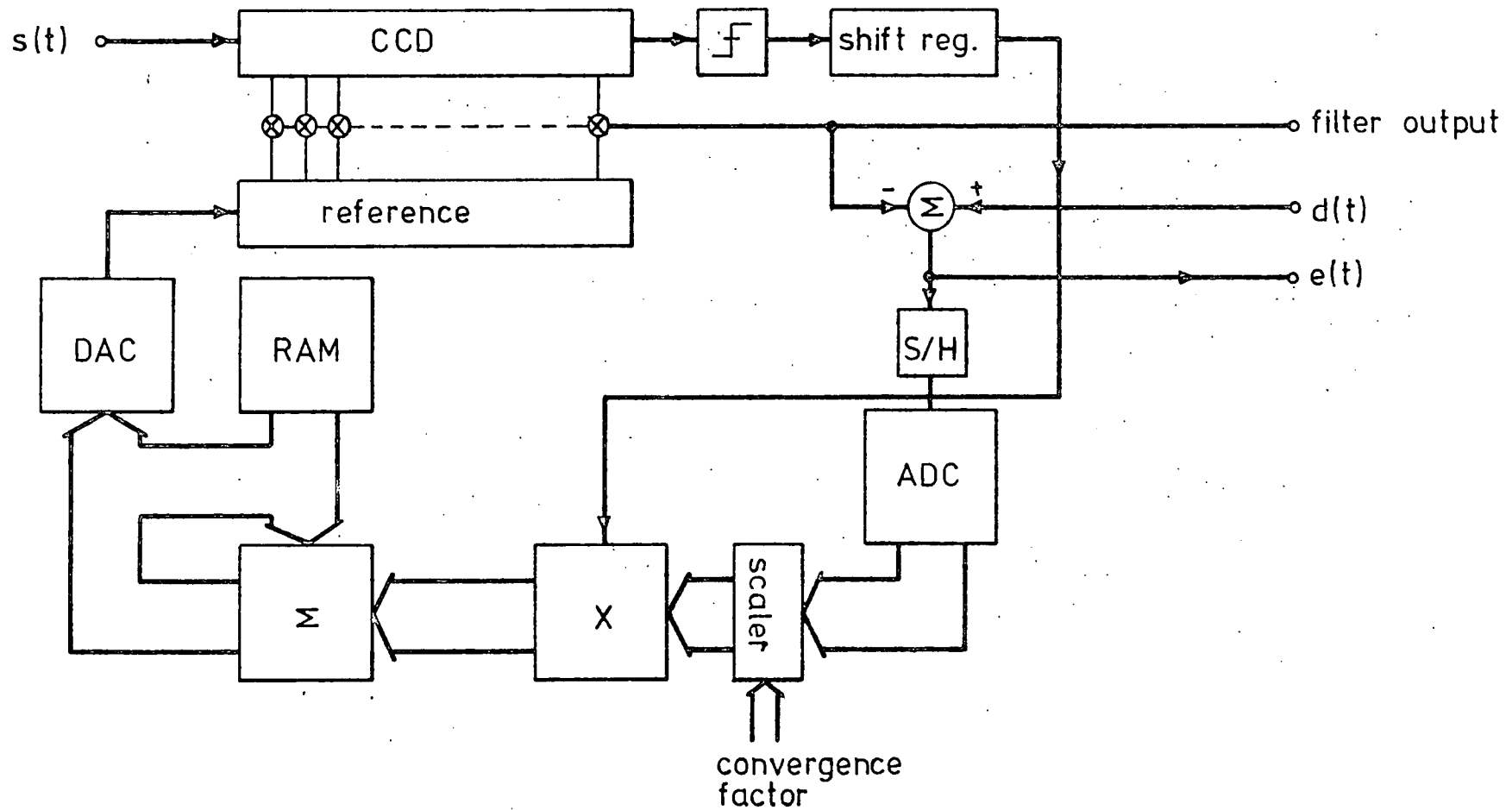
Figure 8.1: Block diagram of the 256-point adaptive filter.

of digital form (an 8 bit word in this case) since this would require a fast settling ADC and a parallel digital multiplier (8 x 8 bits) to form the e(t).s(t-n) product. It was therefore decided to use the clipped LMS algorithm which only requires the sign of s(t-n), i.e. a one bit quantity, and this also means that the error multiplication can be done using a set of exclusive OR gates. The reason for the further delay of the s(t-n) signal (actually a 100 point delay in the experimental system) was that the last signal on the CCD delay line part of the PTF must be used in the update process and must therefore be preserved until the e(t) analogue to digital conversion is complete. Therefore the delay time (of this extra delay) corresponds to the time allowed for ADC settling. The e(t) output is sampled and applied to the ADC input at a time when s(t-n) is on the last tap of the CCD and the output of the ADC is latched at a time $100f_c^{-1}$ later (f_c = the sampling frequency) when $\text{sgn}[s(t-n)]$ is on the output of the 100 point digital shift register. The two quantities, e(t) and $\text{sgn}[s(t-n)]$, are then multiplied to begin a new weight vector update cycle. The use of this scheme meant that a relatively slow monolithic ADC could be used without degrading the overall system bandwidth.

The point of varying the weight vector update period was to remove the possibility of weight divergence at certain periodic input frequencies, experienced with the previous CCD systems (see section 7.1). The weight update period

was made to be 257 and 258 samples alternatively which means that all signals are entirely sampled for this system, though the time required to fully define signals having a period of 257 or 258 samples is twice as long as the time normally required for other signals since an individual error term will be sampled twice on every other update cycle at these frequencies.

The convergence factor in this system was controlled digitally by the insertion of a simple digital scaling circuit after the e(t) ADC. This circuit allowed a range of 8 discrete scaling factors between 2^{-5} and 2^{-12} in integral powers of two. This was easily achieved digitally by taking the 8 input bits from the ADC and shifting their position in a 16 bit output word. This 16 bit output word was then applied to a 24 bit wide digital adder circuit of which the top 5 bits were zero (and also the last 3 bits). This adder used as its other input 256 different locations in a 24 bit wide RAM array which was the external weight store in this case. The output (most significant) 10 bits were converted to analogue form and supplied to the PTF reference input. By using this entirely digital integration loop the precise control required over the convergence factor is attained and quantitative comparison of convergence characteristics with theoretical predictions may be achieved.

The whole system was built on 4 international size circuit

cards with wire-wrap interconnect (see figure 8.2) and

the system was housed in a standard 19 inch rack (see

figure 8.3) with its own power supplies. The total power

consumption of the unit was approximately 7.5 Watts with

a maximum bandwidth of about 200 kHz.

8.2 Convergence characteristics

The results presented in this section demonstrate the

theoretical predictability of the convergence

characteristics of the 256-point adaptive filter

prototype. The input signals used were the same as those

described in section 7.1, i.e. the convergence time for

the case where both inputs are identical sinusoids was

measured. The convergence time constant defined in

equation (7.1) defines the convergence time for an ideal

parallel update adaptive filter, but for this prototype

the convergence time will be 256 times as long, due to

the serialised update scheme.

From the premises stated above the convergence time of the

filter for a convergence factor of 2^{-5} should be about

0.5 secs. with a sampling rate of 50 kHz. The

convergence time should then double for each successive

halving of μ. The results illustrated in figure 8.4 show

a graph of experimentally measured convergence times

plotted against the convergence factor μ over its full

range from 2^{-5} to 2^{-12}. It can be seen from this graph
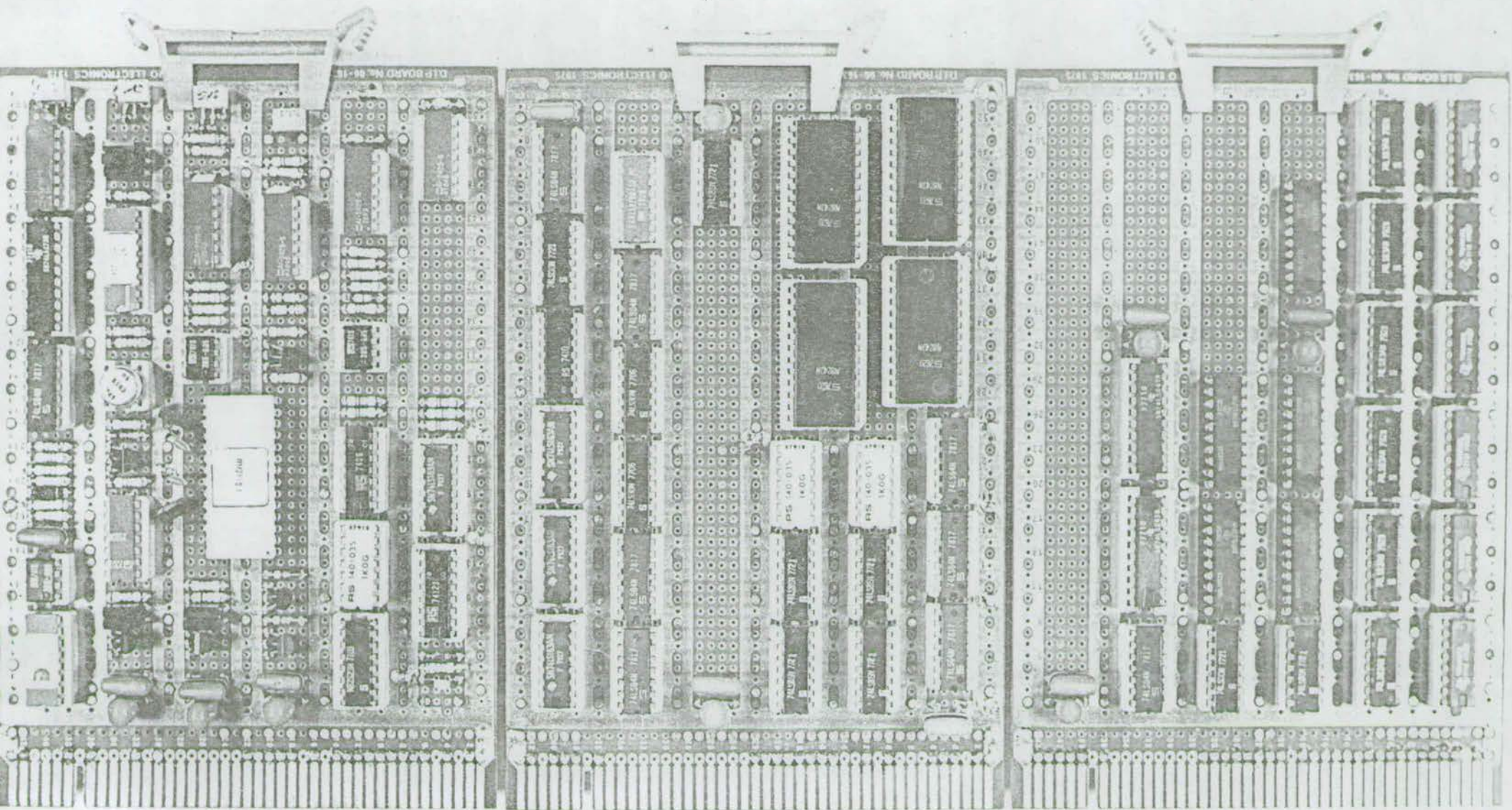
Figure 8.2: Photograph of the 256-point filter circuit boards.
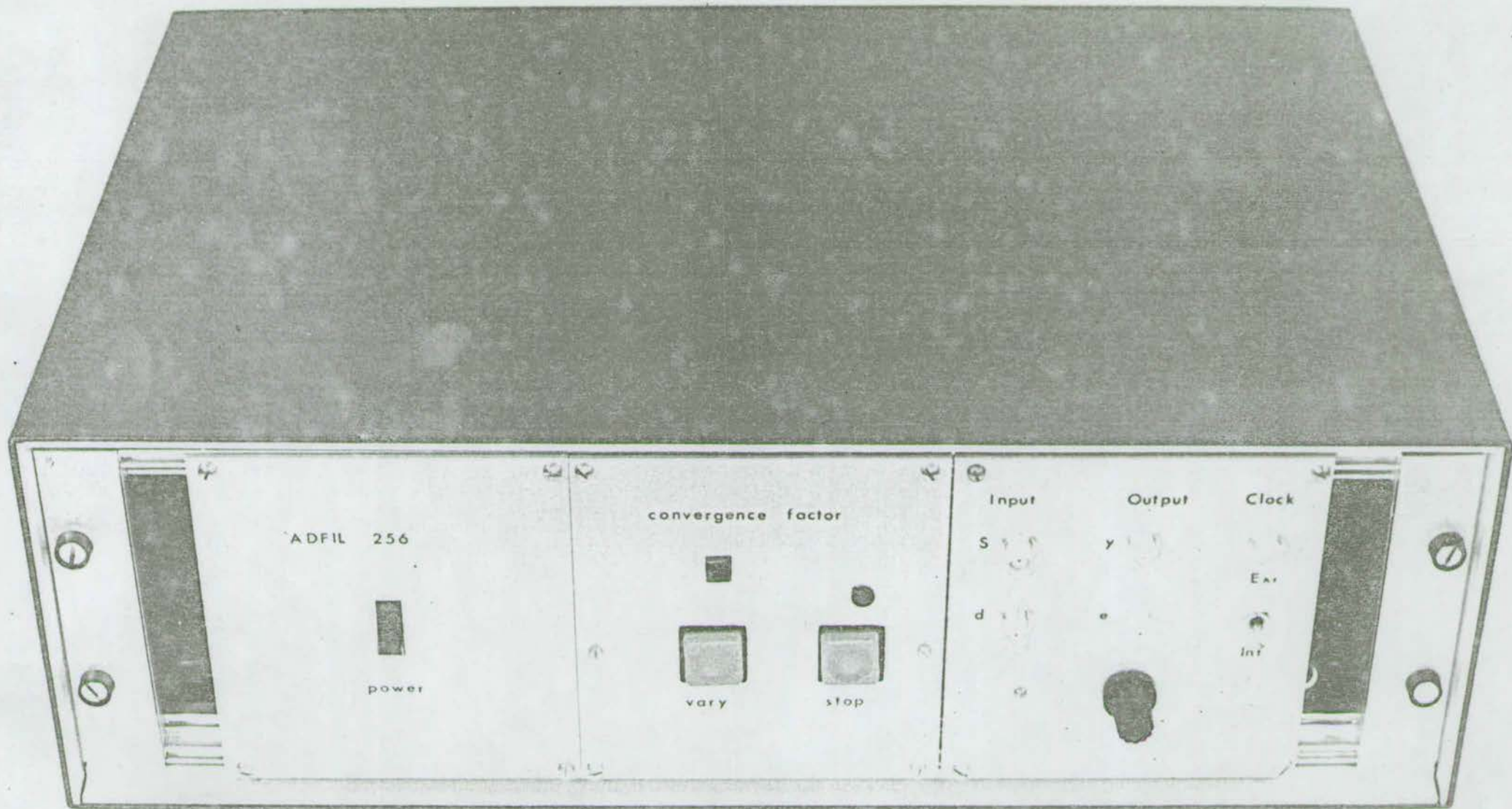
Figure 8.3: Photograph of the complete 256-point adaptive filter.
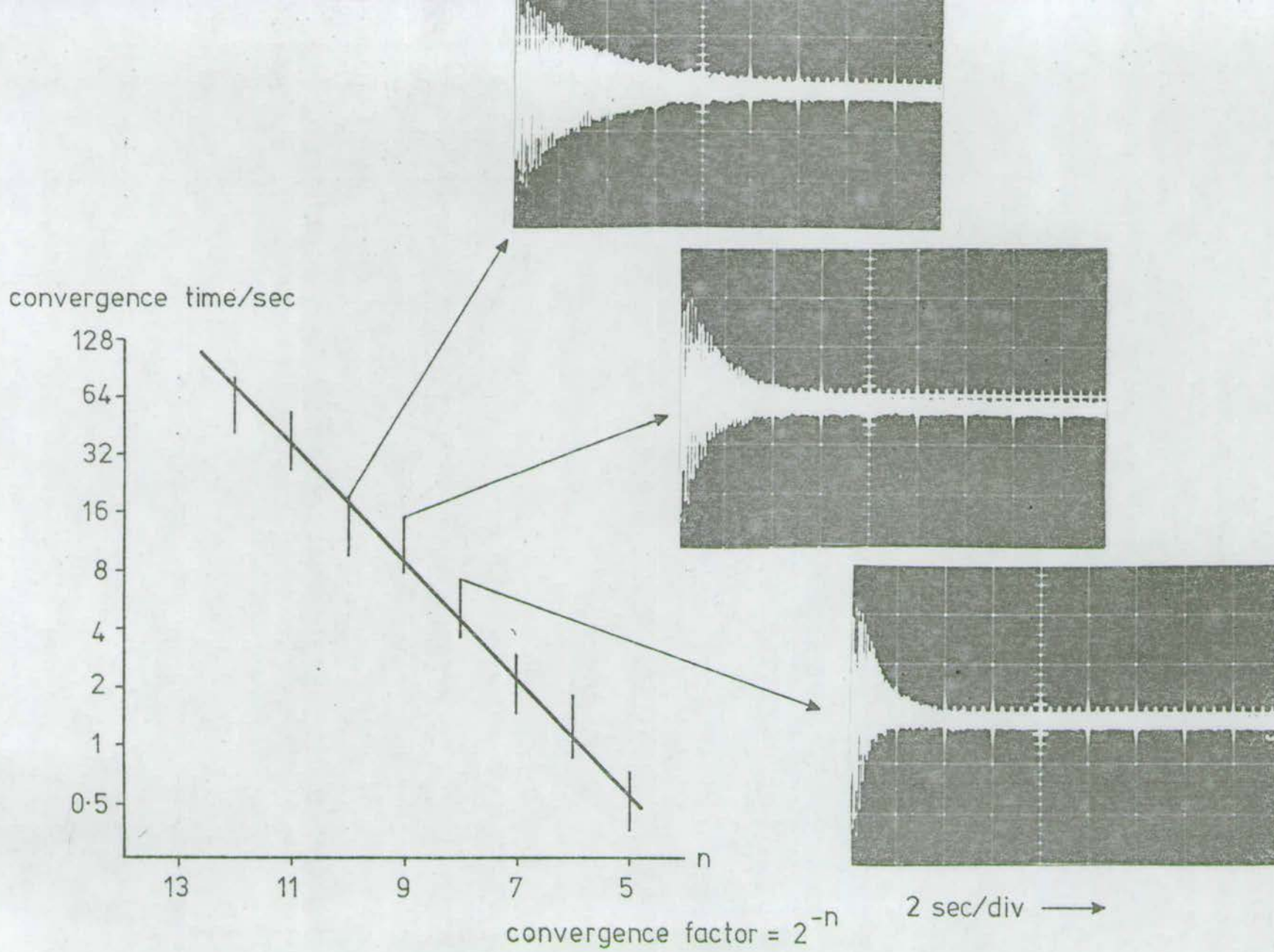
Figure 8.4: Results showing the variation in convergence
time with convergence factor.

that the measured convergence times fall very close to
the expected theoretical values. The inset photographs
in figure 8.4 show the e(t) output during convergence for
some selected values of convergence factor. These results
show typical convergence characteristics achieved with the
e(t) output decaying experimentally to zero with time.
Since convergence of the filter is a statistical process
the convergence times tend to deviate around the expected
values to some extent but the general trend is shown here
to adhere very closely to the theoretically expected
values. The data bars in the graph illustrate this
spread of results over 10 estimates of convergence time.

Apart from these specific convergence characteristics it
was found that this filter using varying weight vector
update periods did in fact converge to all periodic inputs
within the operating frequency band and the comb frequency
spectrum described in section 7.1 did not exist with this
module.

8.3   Cancellation characteristics

The objective in this section is to show experimentally
the trade-off between convergence time and converged
weight accuracy with varying convergence factor μ.   A
method for measuring the adaptivity has already been
discussed in section 7.2 and the same set of test signals
will be used here to measure the cancellation performance

of the 256-point adaptive filter.

The result shown in figure 8.5 shows a typical cancellation result for a convergence factor of 2^{-12}. Trace (a) shows the s(t) input consisting of a sinusoid at frequency f while trace (b) is the d(t) input comprising a combination of sinusoids at frequencies f and 2f. The resultant cancellation (e(t) output) is shown in trace (d) and the spectra of d(t) and e(t) are shown, superimposed, in figure 8.5(e) and (f). The cancellation of the unwanted component was about 50 dB in this case. The additional spectral peaks which can be seen close to the d.c. component of the e(t) spectrum in figure 8.5 are due to a fixed pattern noise component on the y(t) output. This fixed pattern noise is due to the non-zeroed mode of operation of the PTF multipliers. However, since the noise components occupy only a very small part of the total available bandwidth they may be easily removed, for instance by using a switched-capacitor post processing filter on the output. The graph shown in figure 8.6 shows the variation in adaptivity with convergence factor over the full range of μ. The inset photographs show typical residual error levels for a few selected points on the graph. Comparing figures 8.6 and 8.4 it is clear that the expected trade-off does exist between convergence time and the accuracy of the filter output.
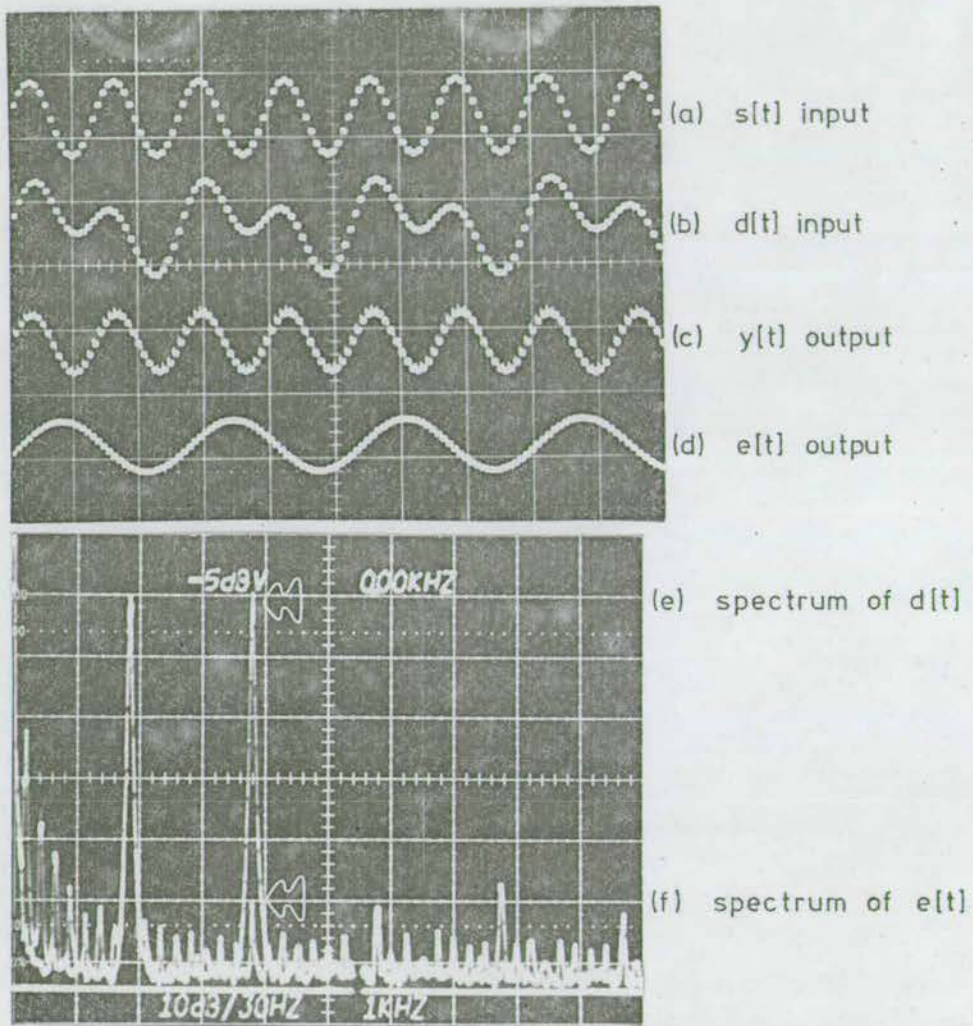
Figure 8.5: Operation of the filter as a canceller.
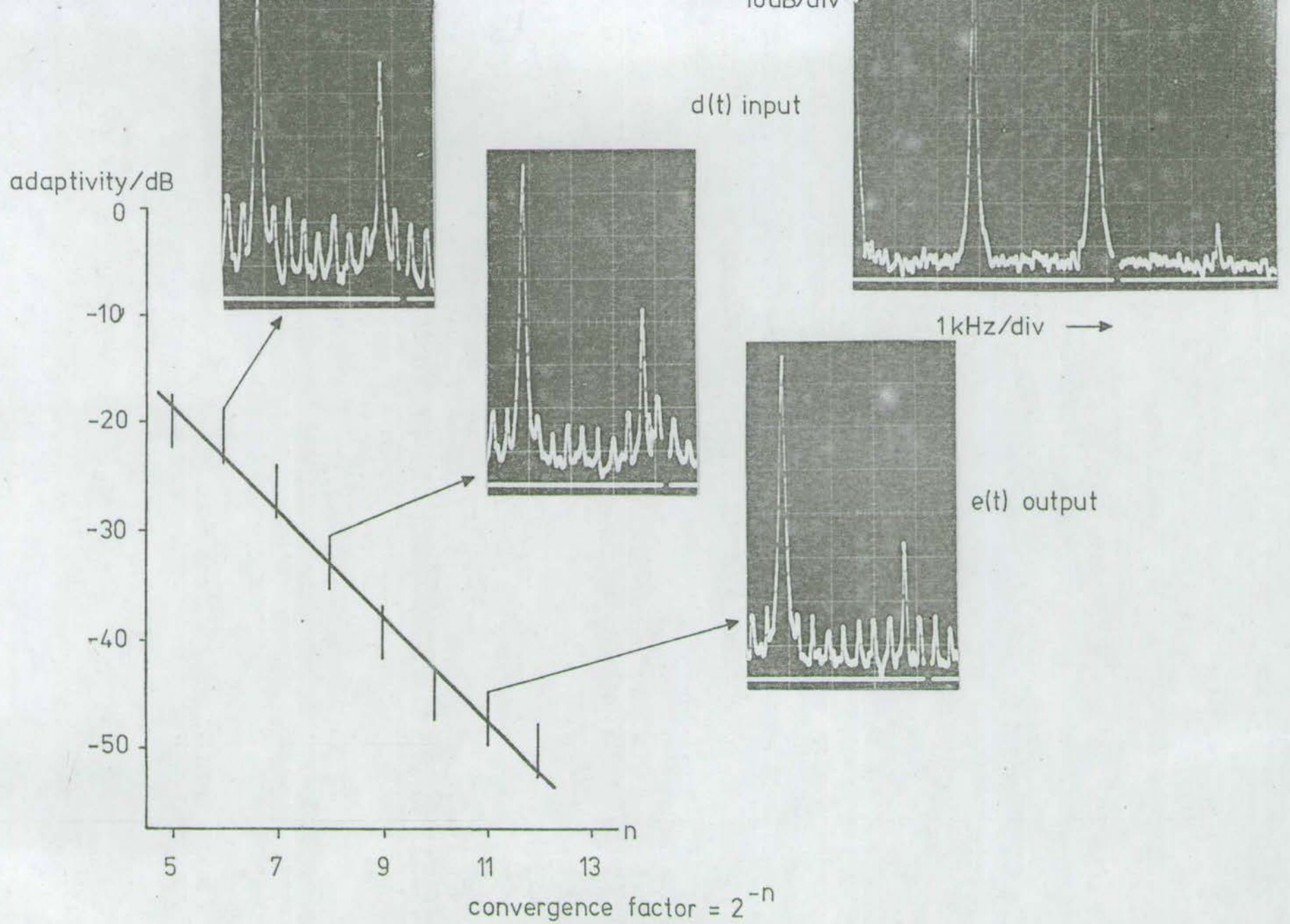
Figure 8.6: Results showing the variation of adaptivity with convergence factor.

8.4 Matched and inverse filter results

It has been shown in the last two sections that the 256-point filter does in fact operate in a way which corresponds very closely with theoretical predictions in terms of convergence time and filter accuracy. In this section results are presented which show the performance of the 256-point system as a matched filter and as an inverse filter.

The results shown in figure 8.7 demonstrate the operation of the 256-point system as a matched filter where the input signal, s(t), is a linear f.m. sweep from d.c. to the Nyquist frequency (figure 8.7(a)). The d(t) input is a single pulse shown in figure 8.7(b) and the converged output shown in figure 8.7(c) shows a good correspondence to the d(t) training signal. The area immediately around this output pulse is shown on an expanded time scale in figure 8.7(d) showing that sidelobes have been almost totally suppressed, as demanded by the d(t) training signal. The actual sidelobe level was measured to be about -25 dB in this case. The impulse response of the PTF is not shown in this case because the output signal is effectively masked by the PTF fixed pattern noise which appears at approximately the same level as the impulse response.

The trace shown in figure 8.7(e) is the same signal as
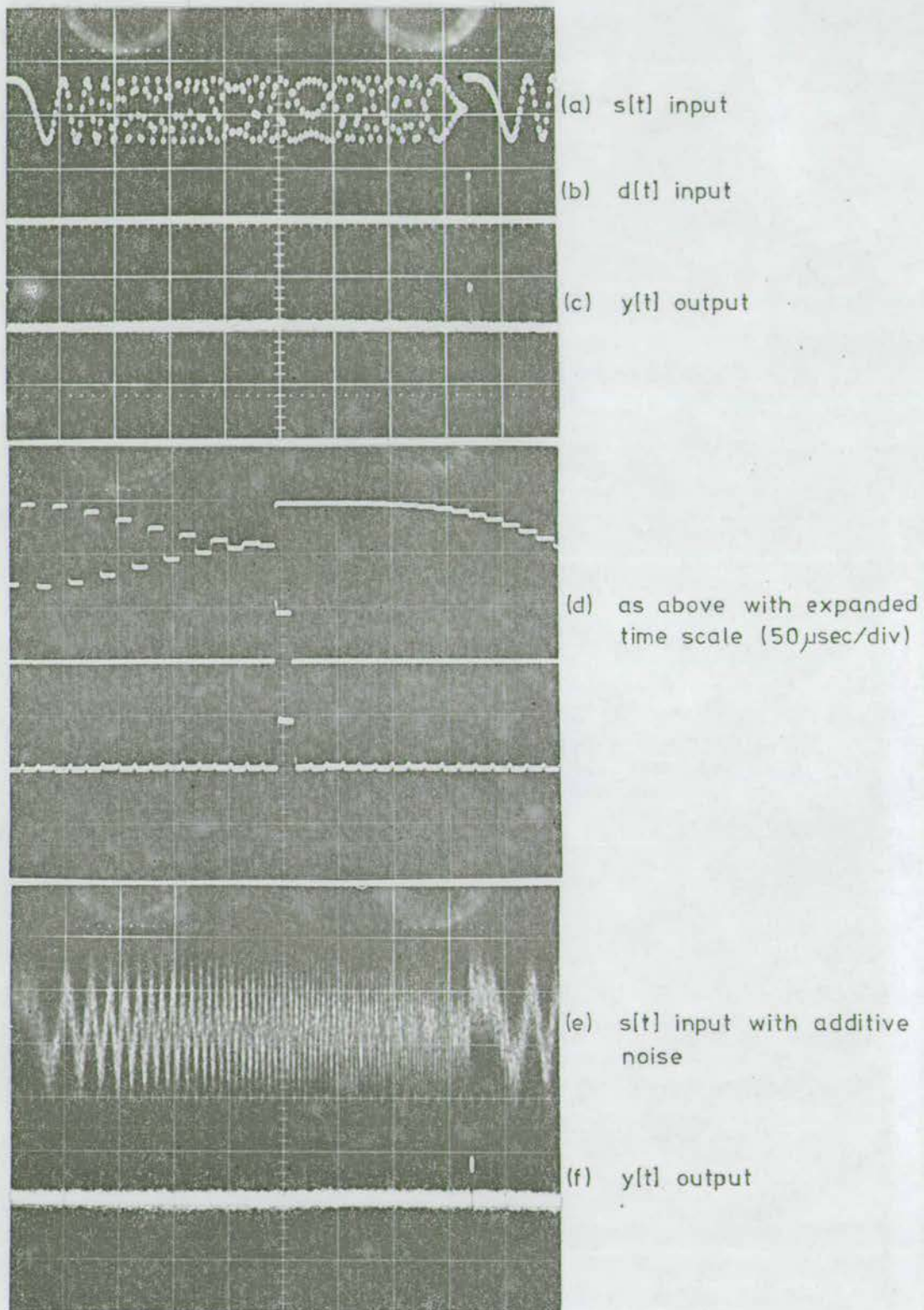
Figure 8.7: Operation of the 256-point filter as a
matched filter.

(a) s[t] input

(b) d[t] input

(c) y[t] output

(d) as above with expanded
time scale (50 μsec/div)

(e) s[t] input with additive
noise

(f) y[t] output

that in figure 8.7(a) with additive white noise which was supplied to the s(t) input after the adaption loop had been switched off, i.e. μ was reduced to zero. The corresponding filter output is shown in figure 8.7(f) which shows clearly detectable pulse in the same position as the training pulse d(t). This is a clear demonstration of effective matched filtering achieved through adaptive training with high time-bandwidth product signals.

The result shown in figure 8.8 demonstrates the operation of the 256-point system in the inverse filter mode where the two inputs (s(t) and d(t)) were identical square waves. The filter output (shown in figure 8.8(c)) again shows a high degree of correspondence with the d(t) input (shown in figure 8.8(b)). However, in this case the total period of the input square wave was only 8 samples meaning that it was possible for the filter to use 64 active weighting points (25% of full weight power). When lower weighting powers are forced by increasing the period of the input square wave the quality of the y(t) output is severely degraded. This is due primarily to the relatively high noise floor on the output of this device which tends to have a more pronounced effect on this device due to the higher time-bandwidth product of 128 (as opposed to 32 on the devices considered in the previous chapters). It can be said, therefore, that the operation of this system in the inverse filter mode is poor in comparison with the quality of operation shown by

178

(a) s[t] input

(b) d[t] input

(c) y[t] output
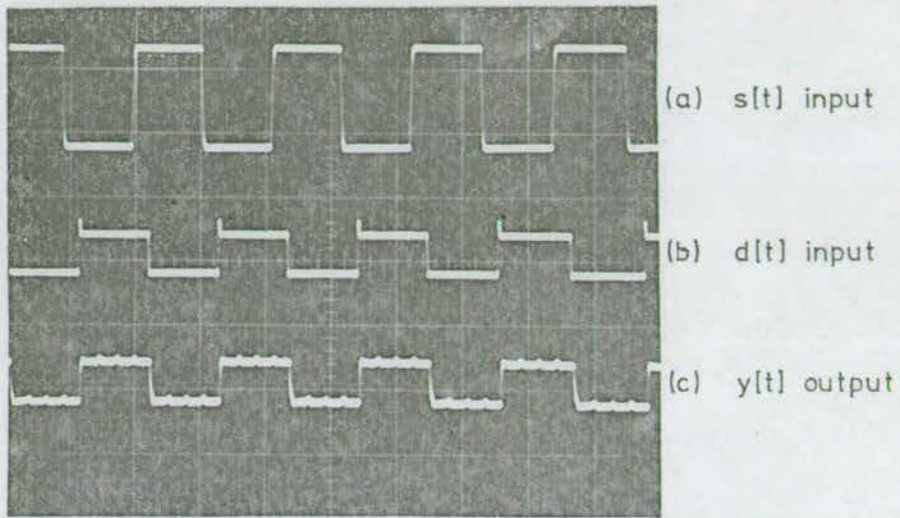
Figure 8.8: Operation as an inverse filter.

the previous 64-point system in figure 7.11.

The final result, shown in figure 8.9, demonstrates the operation of the 256-point prototype system as a self-tuning filter. The input signal in this case was a pair of sinusoids at frequency f and 4f with additive white noise. The y(t) output of the filter is shown in figure 8.9(b) with corresponding frequency spectrum in figure 8.9(e). It can be seen from this that the noise component of the signal is suppressed at the y(t) output by an average of about 25 dB. The effect of the higher time-bandwidth product is clearly endorsed by the sharp definition of the two discrete passband frequencies. In fact, here, the total system bandwidth was about 30 kHz and the actual width of the passbands around the two accepted sinusoids is less than the 300 Hz resolution of the spectrum analyser.

The trace shown in figure 8.9(f) illustrates the degree of cancellation of the sinusoids of the e(t) output of the filter. This was performed by allowing the filter to adapt in the self-tuning mode of operation, and then stopping the adaption (by reducing μ to zero) and then switching off the noise component of the signal while monitoring the e(t) output. The cancellation of the two sinusoids in this case can be seen to be approximately 38 dB.

(a) s[t] input

(b) y[t] output

(c) e[t] output

(d) spectrum of s[t]

(e) spectrum of y[t]

(f) spectrum of e[t] when the noise signal is removed
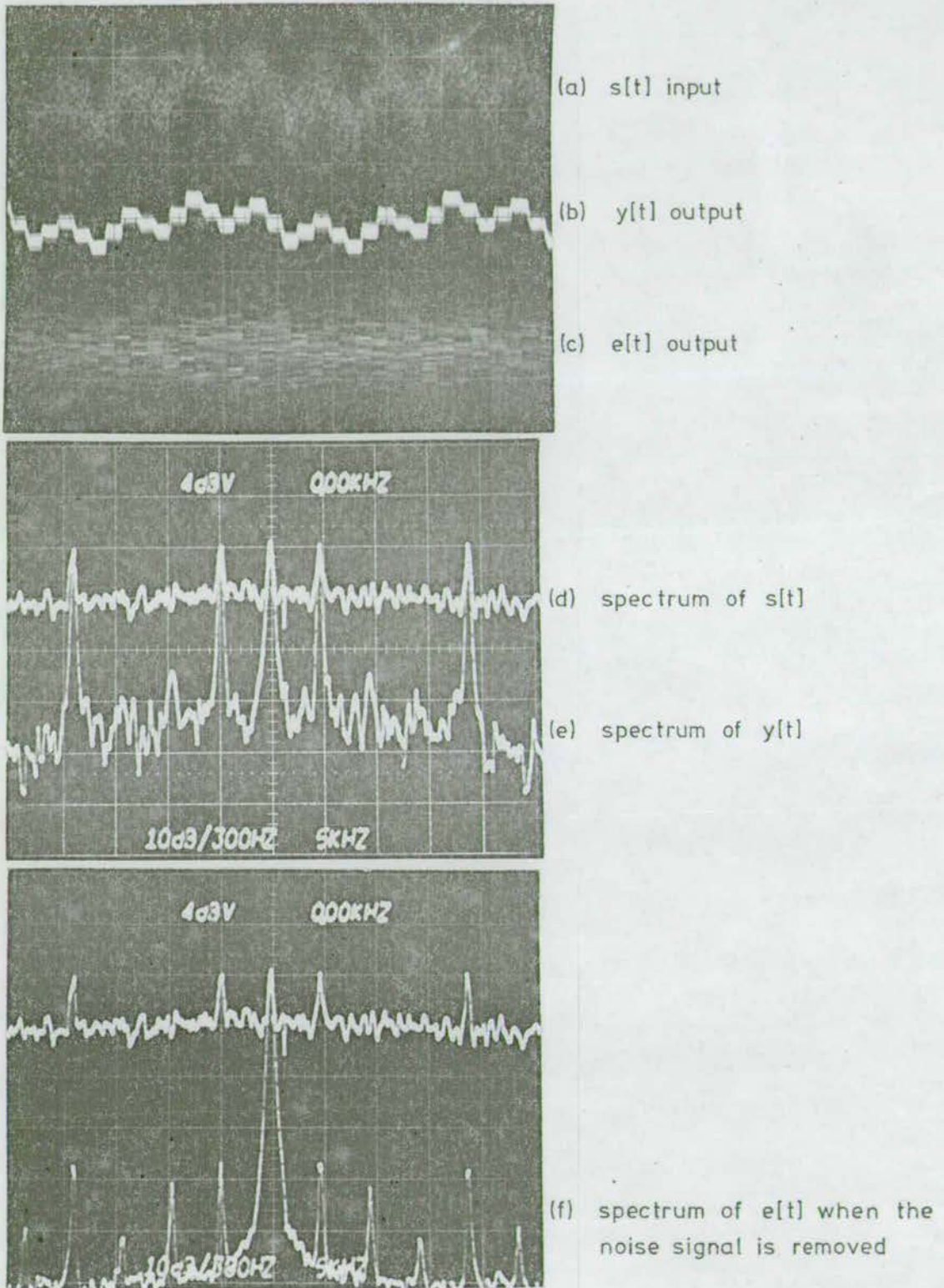
Figure 8.9: Operation as a self-tuning filter.

8.5 Summary

It has been shown in this chapter that the convergence
and weight accuracy characteristics of this 256-point
adaptive filter module correspond closely to theoretically
predicted values. Since this particular filtering device
contained all the possible error sources, such as transfer
inefficiency, gain errors and multilpier errors, it is
clear that these errors make very little difference to
the overall operational characteristics of the CCD adaptive
filter system. The main objectives of increasing the
bandwidth and smoothing the overall convergence character-
istics of earlier filters have been achieved with an
increase in bandwidth of an order of magnitude from 25 kHz
to 200 kHz over that achieved with the 64-point system,
and the system was found to converge for all continuously
periodic input signals.

The only undesirable characteristic still maintained by
this system was the fixed pattern noise at the filter
output which was due to breakthrough on the reference
sample and hold sites together with the inability to
'zero' the multipliers with this device. Another error
source which was found with this system was that of system
generated d.c. offsets at the y(t) output when s(t) was
a.c. coupled, but d(t) had a d.c. offset. This was
explained in section 6.4 as being an error effect caused
by the single transistor multiplier scheme. It would,

therefore, seem to be desirable to use a multiplier

zeroing scheme (also described in section 6.4) with an

adaptive filter in order to remove these d.c. errors at

the outputs, this would also effectively remove the

fixed pattern noise problem mentioned above.

CHAPTER 9: CONCLUSIONS

In this thesis two types of system have been considered
with reference to their implementation using analogue CCD/
MOS circuitry: that is, open and closed loop systems.

In chapter 3 the open loop systems described demonstrated
the power of the analogue processor in terms of its
parallel processing ability when applied to the solution
of complex problems requiring the rapid manipulation of
signal vectors. This was particularly true of the linear
regression analyser described in section 3.1. A small,
low power system of this type is ideally suited to use as
a constant false alarm rate decision module following a
noise reduction system, in particular for use in
communications receivers (e.g. sonar return processing).
The system described here could be modified in a number
of ways to extend the basic performance characteristics
described in section 3.1.

(i) The dynamic range could be extended to a considerable
     extent either by the use of compression techniques
     at the input to CCD2 or by performing the squaring
     operation, not prior to the input to CCD2 but after-
     wards at each tap output. The dynamic range would
     then be limited by the squaring circuits and not by
     the CCD dynamic range. A very simple multiplier
     circuit using a single MOST has already been mentioned

in chapter 2.

(ii)   Dark current effects in the differentiator circuit
       could be overcome by the use of an alternative zero
       signal in the CCD3 register giving a reference level
       which could be subtracted from the final value so
       compensating for effects not attributable fo the
       signal. The use of this technique would however half
       the available maximum signal frequency.

(iii) A more accurate indication of peak position could be
       obtained by the use of a more complex algorithm. Such
       a technique would use the interpolation on actual
       values of the significant derivative on both sides of
       the peak.

The second loop system described in section 3.2 differs
from that considered above as it addresses the specific
problem of dark current noise cancellation in a CCD SPS
memory. Since this is a required peripheral to the memory
device it is important that the processor be simple and
that it should be easily implemented monolithically on the
same chip as the memory, using the same technology. All
these requirements were achieved using the canceller
described in section 3.2 and a good noise cancellation
performance was observed

Both the systems mentioned above demonstrated the ability

of the CCD signal processing module to handle a high rate
of data throughput with a minimum expenditure on power
consumption and size. At present these functions could
not be handled efficiently by digital processors due to
the high cost of digital multiplication. However, with
the approaching prospect of VLSI devices making digital
systems smaller and the increasing use of new algorithms
to reduce the complexity of digital filters, the analogue
device cannot guarantee to compete against the arbitrarily
high accuracy using digital computational elements. Taken
in this context the parameters of prime importance in
considering the acceptability of analogue CCD systems must
be overall system stability and component accuracy.

The bulk of this thesis was centred around the consider-
ation of adaptive filters using analogue CCD programmable
transversal filters. It was shown in chapters 6 and 7
that the CCD filters were inherently stable when used in
a closed loop adaptive processing situation and that the
algorithm considered (the Widrow LMS adaption algorithm)
contained the inherent ability to compensate for many of
the errors which arise in the analogue filter structure.
Also considered in chapter 7 was a digital adaptive filter
based on the conventional DELTIC digital filter
architecture. It was shown here that the digital system
was particularly lacking in terms of application
flexibility due to the difficulty in performing digital
scaling (simple attenuation in analogue terms). In

addition certain other basic differences were illustrated, such as the lower bandwidth of the digital processor due to the requirement to use a single multiplexed multiplier and the necessary degradation of the adaption algorithm in order to avoid the need for a second digital multiplication at each filter point. However, the digital system is much more efficient for the processing of very low frequency signals due to the dark current problem using CCD. It is also true to say that when using the DELTIC approach to digital filtering it is comparatively simple to implement an adaptive system since the signal and weight vectors have already been serialised. The graph shown in figure 9.1 illustrates the general ranges of applicability of digital and analogue CCD adaptive filters in the author's opinion (with the addition of possible SAW implementations)[67]. It can be seen from this that CCD adaptive filters occupy a very significant range of operation between a few kHz of signal bandwidth up to about 6 MHz. They could therefore offer a viable approach to many adaptive filtering problems from voice-band echo cancellation to sonar beam forming and video satellite link equalisation. At the present the digital processor offers the only really viable approach to low bandwidth problems (for instance in processing seismic data) and digital processors may also supply an alternative to CCD at higher sampling rates for low time-bandwidth products.

In chapter 7, although a great deal of qualitative data
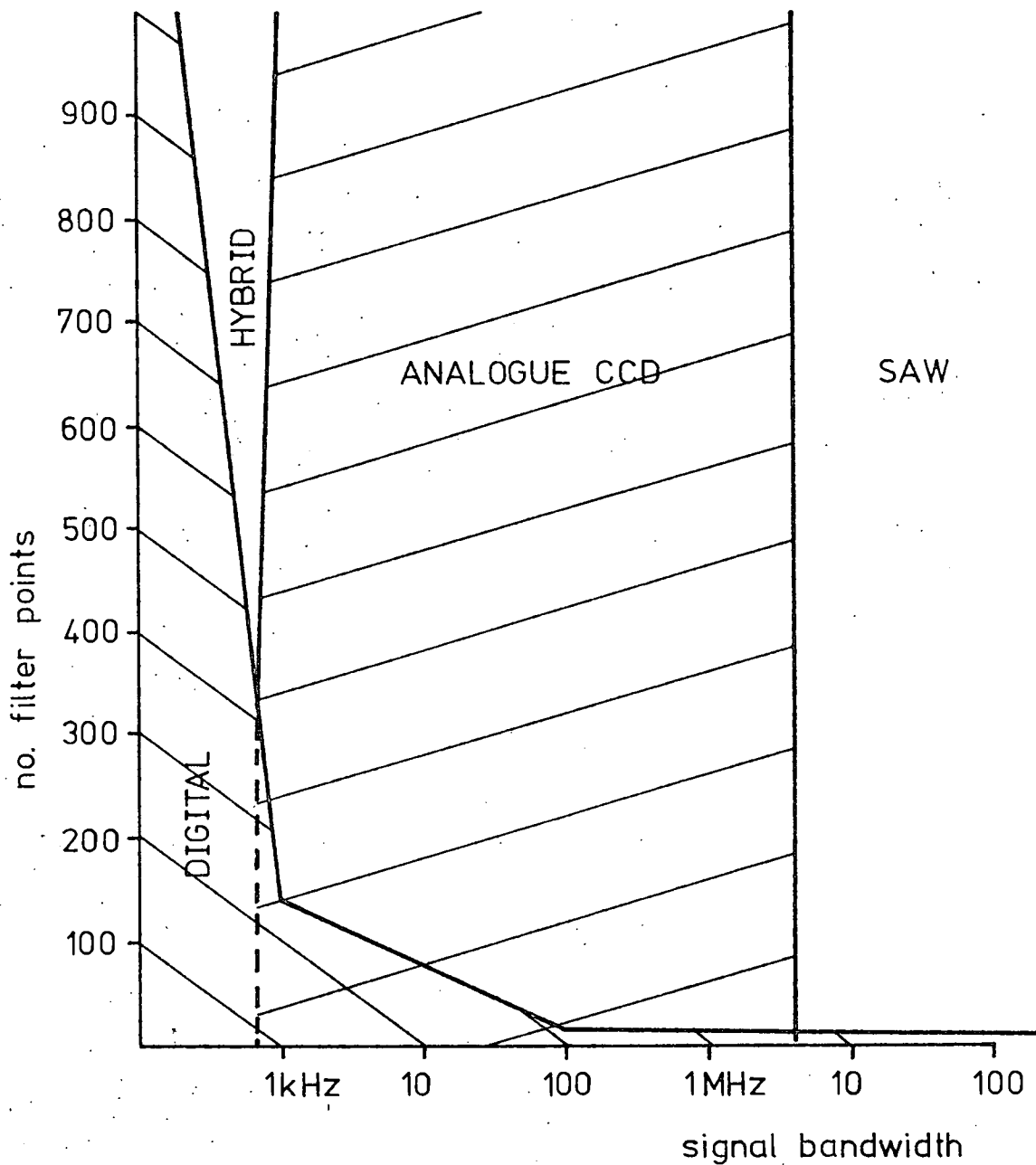
Figure 9.1: Graph illustrating the applicability of
various circuit techniques to adaptive
filtering.

was generated which illustrated the general feasibility
of CCD adaptive processors it was generally impossible to
compare the data generated with theoretical predictions in
any meaningful way. This was because of the design of the
original prototypes which were not easily controllable and
the system parameters were diffcult to measure. The 256-
point adaptive filter described in chapter 8 was designed
to overcome these problems, providing a flexible unit with
known (preprogrammed) system parameters allowing easy
comparison with concurrent theoretical predictions. The
results presented in this chapter showing the very close
correspondence between theoretical predictions and
experimentally measured values for convergence properties
serves to reinforce the conclusion that the adaptive CCD
filter is indeed error tolerant. The system also
demonstrated a much higher bandwidth capability of about
200 kHz and a 'lower' power consumption of about 7.5W.

In conclusion, the final chapters of this thesis have
demonstrated the basic feasibility of constructing
flexible, low power adaptive filters using analogue CCD/
MOS circuit techniques. In order to have a maximum impact
on the signal processing market it would be very desirable
to reduce the complete structure to a single integrated
circuit. Considering the fact that the major component
of the adaptive filter (the programmable transversal
filter itself) has already been integrated, this presents
no major obstacle as the remaining structure of a second

multiplier and an integrator for each filter cell presents no real problem for monolithic implementation. The design of such a filter having 65 filter points is in fact now being commissioned by the Wolfson Microelectronics Institute[32].

The use of the type of system described in this thesis using a digital implementation for the algorithm calculation circuit still remains a viable alternative, however, since the programmable transversal filter itself has a much wider potential range of use.

REFERENCES

1. Boyle, W.S., and Smith, G.E.; "Charge-coupled
    semiconductor devices", Bell Syst. Tech. J., 49;
    pp. 587-593, April 1970

2. Amelio, G.F., Tompsett, M.F., and Smith, G.E.;
    "Experimental verification of the charge coupled
    device concept", ibid., pp. 593-600

3. Masenten, W.K.; "Adaptive signal processing", IEE
    Conf. Proc. no. 180 on "Case studies in advanced
    signal processing", Peebles, Scotland, 1979, pp.
    168-177

4. Krammer, J.D.R.; "Adaptive antenna array processing:
    a study of weight error effects", ibid., pp. 184-
    189

5. Schiffman, M.; "Playback control speeds or slows
    taped speech without distortion", Electronics, 47,
    pp. 87-94, August 1974

6. Wen, D.D.; "A CCD video delay line", IEEE Int.
    Solid-State Circ. Conf., Philadelphia, pp. 204-205,
    February 1976

7. Geddes, R.C., Cowan, C.F.N., Mavor, J., and Dickson,
    J.K.; "Adaptive cancellation of fixed-pattern noise
    in CCD serial-parallel-serial memory", Electron.
    Lett., 15, pp. 505-507, August 1979

8. Collins, D.R., Bailey, W.H., Gosney, W.M., and Buss,
    D.D.; "Charge-coupled-device analogue matched
    filters", Electron. Lett., 8 pp. 328-329, June 1972

9.   Baertsch, R.D., Engler, W.E., Goldberg, H.S., Puckette, C.M., and Tiemann,J.J.; "The design and operation of practical charge-transfer transversal filters", IEEE Trans. Electron Dev., 23, pp. 133-141, February 1976

10.   Weckler, G.P., and Walby, M.D.; "Programmable transversal filters: design tradeoffs", Proc. 5th Int. Conf. on CCDs, Edinburgh, 1979, pp. 211-221

11.   MacLennan, D.J., Mavor, J., and Vanstone, G.F.; "Technique for realising transversal filters using charge-coupled devices", Proc. IEE, 122, 1975, pp. 615-619

12.   Denyer, P.B., and Mavor, J.,; "Desing of CCD delay lines with floating gate taps", IEE J. Solid-state and Electron Devices, 1, pp. 121-129, July 1977

13.   Walmsley, C.F., and Gooding, J.N.; "50 MHz. time delay and integration CCD", Proc. 5th Int. Conf. on CCDs, Edinburgh, 1979, pp. 341-346

14.   Denyer, P.B., Mavor, J., and Arthur, J.W.; "Miniature programmable transversal filter using CCD/MOST technology", Proc. IEEE, 67, pp. 42-50, January 1979

15.   Pelgrom, M.J.M., Wallinga , H., and Holleman, J.; "The electrically programmable split-electrode CCD transversal filter (EPSEF)", Proc. 5th Int. Conf. on CCDs, Edinburgh, 1979, pp. 254-260

16.   Sunter, S., Chowaniec, A., and Little, T.,; "Adaptive filtering in CCD and MOS technologies", ibid., pp. 261-267

17.   Bailey, W.H., Buss, D.D., Hite, L.R., and Whatley,
      M.W.; "Radar video processing using the chirp-Z
      transform", Proc. Int. Conf. on Applications of CCD,
      San Diego, 1975, pp.283-290

18.   Jack, M.A., Park, D.G., and Grant, P.M.; "CCD
      spectrum analyser using prime-transform algorithm",
      Electron. Lett., 1977, 13, pp. 431-432

19.   Kapur, N., Mavor, J., and Jack, M.A.; "Discrete
      cosine transform processor using a CCD programmable
      transversal filter", Electron. Lett., 1980, 16, pp.
      139-141

20.   Keutgens, W.J., Simons, R.R., Declerck, G.J., and
      Janssens, E.J.; "Fully-integrated charge-coupled
      PCM line filters", Proc. 5th Int. Conf. on CCDs,
      Edinburgh, 1979, pp. 176-181

21.   Hewes, C.R., Hester, R.K., Pettengill, R.C., Eversole,
      W.L., Mayer, D.J., and Hui, T.K.; "A CCD/NMOS channel
      vocoder", IEE Conf. Proc. no. 180 on 'Case studies
      in advanced signal processing', Peebles, Scotland,
      1979, pp. 134-139

22.   Arthur, J.W., Dix, J.F., Harland, E., Widdowson, J.W.,
      Denyer, P.B., Mavor, J., and Milne, A.D.; "Large time-
      bandwidth product CCD correlators for sonar", ibid.,
      pp. 62-68

23.   Gandolfo, D.A., Tower, J.R., Elliott, L.D., Nossen,
      E.J. and Martinson, L.W.; "CCDs for spread spectrum
      applications", ibid., pp. 90-96

24. White, M.H., Mack, I.A., Borsuk, G.M., Lampe, D.R., and Kub, F.J.; "Charge-coupled device (CCD) adaptive discrete analog signal processing", IEEE J. of Solid-State Circ., 14, pp. 132-147, February 1979

25. Ahuja, B.K., Copeland, M.A., and Chan, C.H.; "A samples analog MOS LSI adaptive filter", ibid., pp. 148-154

26. Schwarz, T., and Malah, D.; "Hybrid realisation of an adaptive filter for real-time noise-cancelling applications", Electron. Lett., 15, 1979, pp. 671-672

27. Cowan, C.F.N., Mavor, J., Arthur, J.W.; "Implementation of a 64-point adaptive filter using an analogue CCD programmable filter", Electron. Lett., 14, 1978, pp. 568-569

28. Arthur, J.W., Cowan, C.F.N., and Mavor, J.; "Novel linear regression analyser using analogue CCD", Electron. Lett., 13, 1977, pp. 751-753

29. Cowan, C.F.N., Mavor, J., and Arthur, J.W.; "Noise cancellation and inverse filtering using a compact high-performance CCD adaptive filter", Electron. Lett., 15, 1979, pp. 35-37

30. Cowan, C.F.N., and Mavor, J.; "Miniature CCD-based analog adaptive filters", IEEE ICASSP80, pp. 474-477, April 1980

31. Denyer, P.B., and Mavor, J.; "256-point programmable transversal filter", Proc. 5th Int. Conf. on CCDs, Edinburgh, 1979, pp. 253-254

32. Cowan, C.F.N., Mavor, J., Arthur, J.W., and Denyer, P.B.; "An evaluation of analogue and digital adaptive filter realisations", IEE Conf. Proc. no. 180 on 'Case studies in advanced signal processing', pp. 178-183, September 1979

33. Beynon, J.D.E., and Lamb, D.R.; "Charge-coupled devices and their applications", McGraw Hill (UK), 1980

34. Arthur, J.W.; "Signal design to minimise inefficiency effects in CCD matched filters", WMI internal report, March 1978

35. Wallinga, H.; "A comparison of CCD analogue input circuit characteristics", Proc. Int. Conf. on CCDs, Edinburgh, 1974, pp. 13-21

36. Rosenbaum, S.D., Chan, C.H., Caves, J.T., Poon, S.C., and Wallace, R.W.;"A 16384 bit high-density CCD memory", IEEE Trans. Electron Devices, 1976, 23, pp. 101-107

37. Tasch, A.F., Broderson, R.W., Buss, D.D., and Bate, R.T.; "Dark current and storage-time considerations in charge-coupled devices", Proc. Int. Conf. on CCDs, San Diego, 1973

38. Mavor, J., and Denyer, P.B.; "The dependence of CCD dark current upon power dissipation", Microelectron. & Reliab., 1978, 17, pp. 403-404

39. Arthur, J.W., Cowan, C.F.N., and Mavor, J.; "Analogue real-time, statistical analyser utilising CCDs to implement a decision algorithm", IEEE Trans. on Inst. & Meas., 1980, 29, pp. 58-66

40. Weatherburn, C.E.; "First course in mathematical statistics", Cambridge University Press, 1952, pp. 194-195

41. Arthur, J.W.; "Digital processing of spectra", J. Raman Spectrosc., 1976, $\underline{5}$, pp. 9-20

42. Bennet, W.R.; "Introduction to signal transmission", McGraw Hill, New York, 1970, pp. 64-71

43. Weimer, P.K., Pike, W.S., Shallcross, F.V., and Kovac, M.G.; "Video processing in charge-transfer image sensors by recycling signals through the sensor", RCA Rev., 1974, $\underline{35}$, pp. 341-354

44. Traynar, C.P., and Beynon, J.D.E.; "A nonrecursive signal integrator using a parallel CCD structure", IEEE J. of Solid-State Circ., 1979, $\underline{14}$, pp. 742-746

45. Wiener, N.; "Extrapolation, interpolation and smoothing of stationary time series", J. Wiley & Sons, New York, 1949

46. Gabor, D., Wilby, W.P.L., and Woodcock, R.; "A universal non-linear filter, predictor and simulator which optimises itself by a learning process", Proc. IEE, 1961, $\underline{109}$, pp. 422-438

47. Lucky, R.W.; "Techniques for adaptive equalisation of digital communications systems", BSTJ, 1966, $\underline{45}$, pp. 255-286

48. Widrow, B., Glover, J.R., McCool, J.M., Kaunitz, J., Williams, C.S., Hearn, R.H., Zeidler, J.R., Dong, E., and Goodlin, R.C.; "Adaptive noise cancelling: principles and applications", Proc. IEEE, 1975, $\underline{63}$, pp. 1692-1716

49. Widrow, B., McCool, J.M., Larimore, M.G., and Johnson, C.R.; "Stationary and non-stationary learning characteristics of the LMS adaptive filter", Proc. IEEE, 1976, 64, pp. 1151-1161

50. Widrow, B., and McCool, J.M.; "A comparison of adaptive algorithms based on the methods of steepest descent and random search", IEEE Trans. Ant. & Prop., 1976, 24, pp. 615-637

51. Moschner, J.L.; "Adaptive filter with clipped input data", Stanford University tech. report no. 6796-1

52. Neissen, C.W., and Willim, D.K.; "Adaptive equaliser for pulse transmission", IEEE Trans. on Comm. Tech., 1970, 18, pp. 377-395

53. Morgan, D.R., and Craig, S.E.; "Real-time adaptive linear prediction using the least mean square gradient algorithm", IEEE Trans. ASSP, 1976, 24, pp. 494-507

54. Corl, D.; "A CTD adaptive inverse filter", Electron. Lett., 1978, 14, pp. 60-62

55. Reid, J.R.C.; "Adaptive filters for speech enhancement: a feasibility study", private communication, Nov. 1979

56. South, C.R., Hoppitt, C.E., and Lewis, A.V.; "Adaptive filters to improve loudspeaker telephone", Electron. Lett., 1979, 15, pp. 673-674

57. Verhoeckx, N.A.M, van den Elzen, H.C., Snijders, F.A.M., and van Gerwen, P.J.; "Digital echo cancellation for baseband data transmission", IEEE Trans. ASSP, 1979, 27, pp. 768-781

58. Sondhi, M.N., and Debasis, M.; "New results on the performance of a well-known class of adaptive filters", Proc. IEEE, 1976, <u>64</u>, pp. 1583-1597

59. Mavor, J., Jack, M.A., Saxton, D., and Grant, P.M.; "Design and performance of a programmable, real-time charge-coupled device recirculating delay-line correlator", IEE J. Electron. Circ. and Syst., 1977, <u>1</u>, pp. 137-143

60. Feintuch, P.L.; "An adaptive recursive LMS filter", Proc. IEEE, Nov. 1976, pp. 1622-1624

61. Ahmed, N.; "A study of adaptive digital filters", Sandia Labs int. report, SAND77-0102

62. Cowan, C.F.N., Arthur, J.W., and Mavor, J.; "CCD based adaptive filters; realisation and analysis", University of Edinburgh, Int. report, Oct. 1978

63. Treichler, J.R.; "The spectral line enhancer - the concept, an implementation, and an application", Ph.D. dissertation, Stanford University (CA), June 1977

64. Arthur, J.W., Reekie, H.M., and Mavor, J.; "A novel technique for enhancing CCD programmable transversal filter performance", University of Edinburgh, Int. report, March 1979

65. White, M.H., Mack, I.A.G., Lewis, L.L., and Borsuk, G.M.; "CCD adaptive discrete analog signal processing", Westinghouse corp. final report, contract no. N00173-76-C-0147

66. Murano, K., Unagami, S., and Tsuda, T.; "LSI processor for digital signal processing and its application to 4800 bit/s modem", IEEE Trans. on Comm. 1978, 26, pp. 499-506

67. Grant, P.M., and Kino, G.S.; "Adaptive filter based on SAW monolithic storage correlator", Electron. Lett., 1978, 14, pp. 562-564

68. Behar, D., Kino, G.S., and Bowers, J.E.; "The storage correlator as an adaptive inverse filter", Electron. Lett., 1979, 15

APPENDIX A: 256-POINT ADAPTIVE FILTER CONSTRUCTION

In chapter 8 a general system outline for the experimental 256-point adaptive filter was described. In this appendix specific details of the actual circuit design are described. Detailed circuit diagrams of the system are shown in figures A.1 to A.4 and a clock timing diagram in figure A.5.

The 256-point adaptive filter was based around the WM2110 monolithic 256-point PTF, described in section 2.5. A circuit diagram showing the main analogue circuit components servicing the PTF is shown in figure A.1. The main clocks for the system are generated using a 4 bit counter and ROM, giving a possible 16 phases within the sampling clock frequency. The clocks generated thus for the PTF are translated to a 15V level simply using open collecter TTL inverters to drive the CCD clocks with bipolar clocks ϕ_r and $\overline{\phi}_r$. A diagram showing the actual clock structure used is given in figure A.5. The signal input to the CCD in the PTF is sampled using a HA2425 sample/hold, which also supplies the input bias level, and the output of this circuit supplies the input diode of the CCD which uses a straightforward diode cut-off input technique (see section 2.1). The last tap of the CCD, which is available on one IC pin is clipped using an offset biased comparator. The output of this comparator, converted to TTL level, is then supplied as the input to a
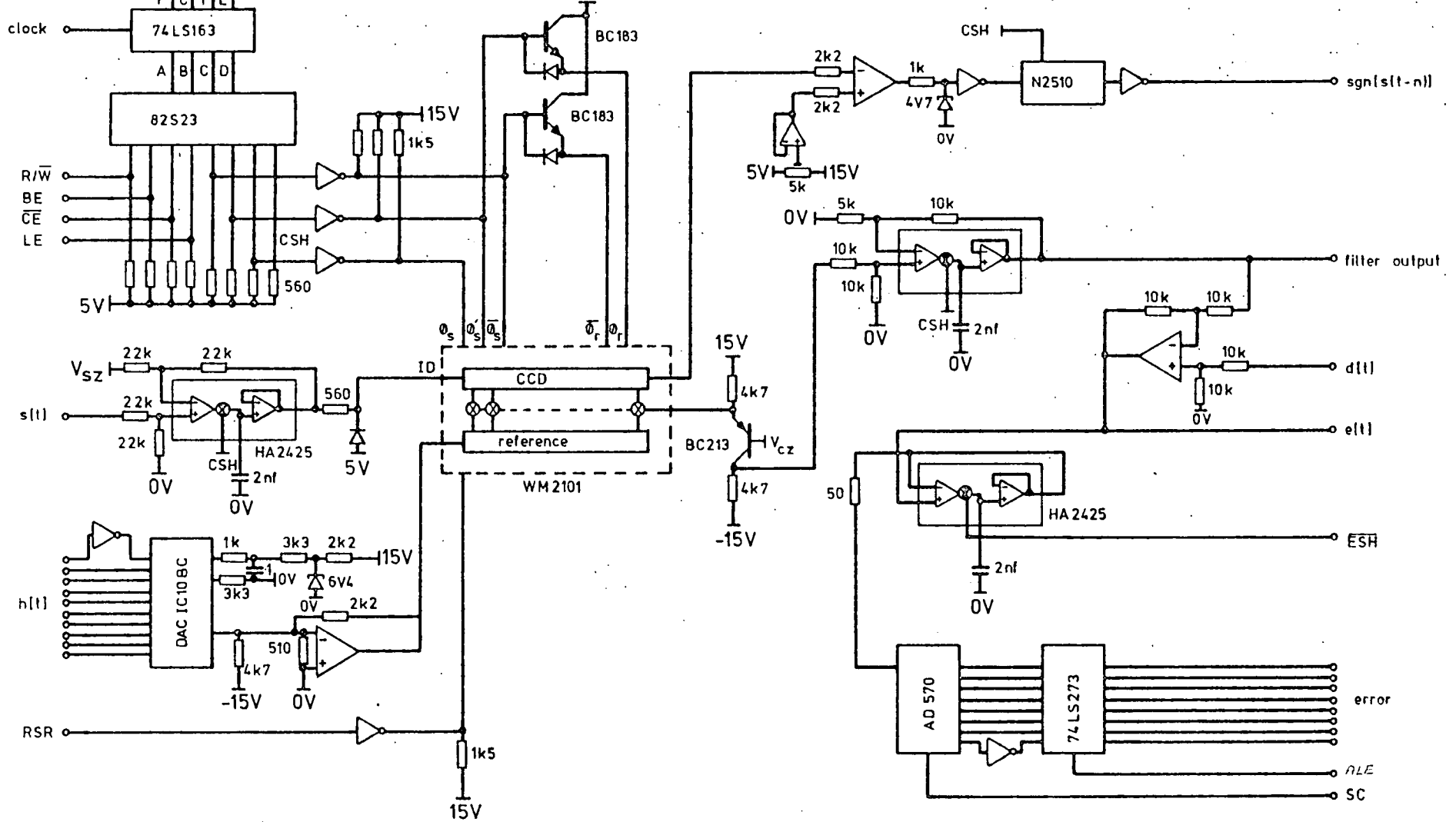
Figure A.1: Circuit diagram of the analogue part of the
256-point adaptive filter.

200

100-point MOS digital shift register. The actual filter output is in the form of a current at the PTF output and this is converted to a voltage using the BC213L common base transistor circuit shown in figure A.1. The output d.c. level is not clamped here and a median level is set using V_{cz}. The result of this current to voltage conversion is sampled on the CSH clock and amplified using another HA2425 circuit to provide the system y(t) output. The error output is then derived using a simple operational amplifier circuit to find the difference between the d(t) input and the y(t) output. This error is sampled using a clock (ESH) which will be described in the next paragraph. This sample/hold output is then the input to an AD570 ADC which is followed by an 8 bit latch supplying the update error to the scaling circuit.

The overall system timing circuit is shown in figure A.2 with a general timing diagram shown in figure A.5. The input to the counter string is the CSH clock which means that the counters advance by one at each sample input period. The repetition rate of these counters is jittered so that reset pulses appear alternately on a count of 257 or 258. This is achieved using the JK flip-flop circuit and the exclusive ORs shown in figure A.2, the reasons for doing this have already been outlined in chapter 8. The main clocks generated by this circuit are:

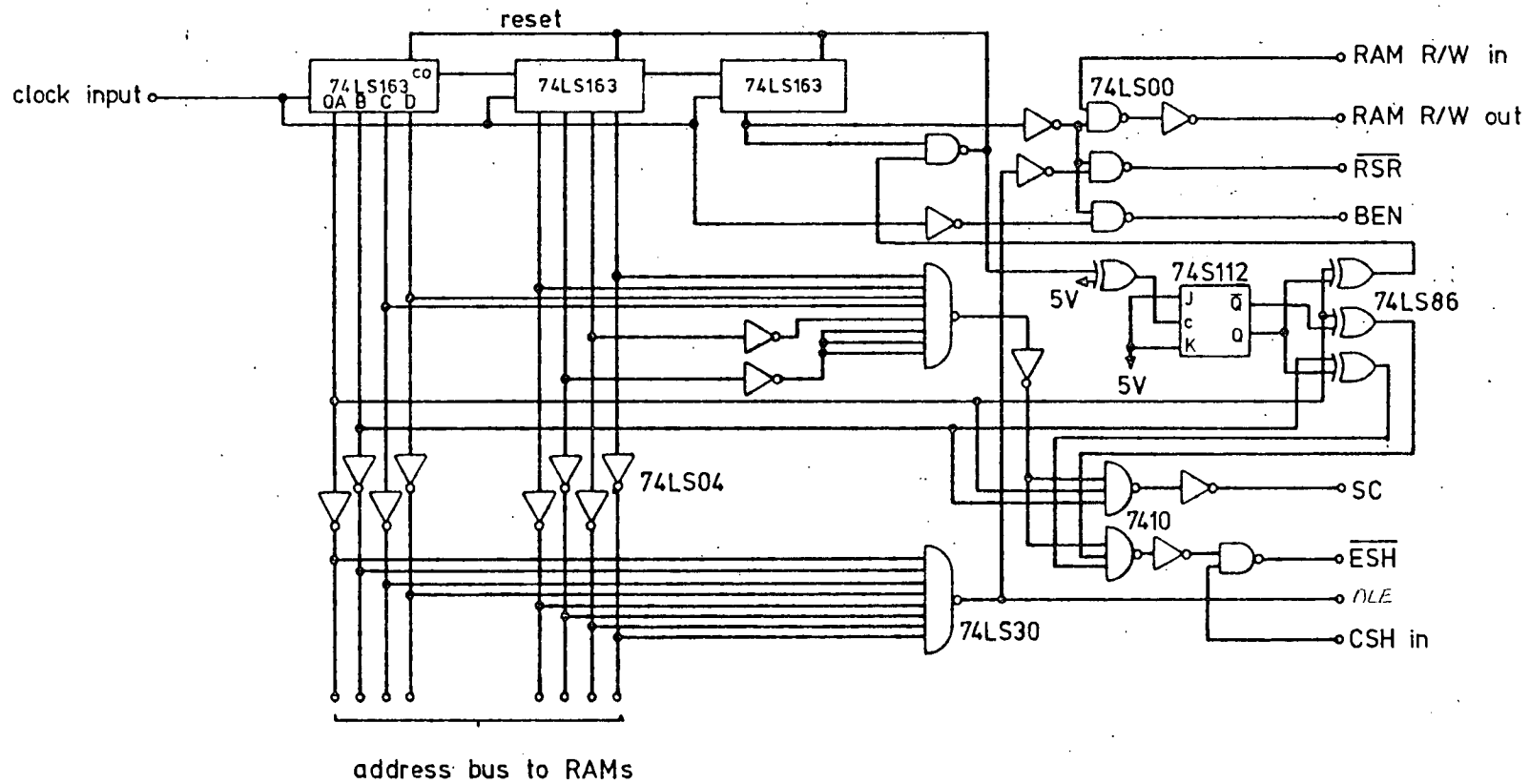(1) A start convert command, SC, to the ADC in figure A.1 which occurs on a count of 160.

Figure A.2: Schematic diagram of the timing circuitry.

(2) An error sample/hold pulse, ESH, which occurs
just before SC on a count of 158 or 159
depending on the overall clock cycle time.

(3) A latch command, ALE, which enables the latch on
the ADC output (see figure A.1), which always
appears on a count of 1.

(4) The reference shift register input for the PTF
which always occurs on the last count of the
sequence.

(5) The $R/\overline{W}$ command to the RAMs and BE clocks are
also inhibited by this circuit on counts of 257
and/or 258.

(6) The addresses to the RAMs in figure A.4.

The digital scaling and multiplication circuits are shown
in figure A.3. The scaling circuit takes the error output
(from the ADC latch in figure A.1) which occupies a width
of 8 bits and shifts its position over a 16 bit output word
according to a 3 bit scaling input command. This is
achieved using 4 N8243 scaling circuits as shown in figure
A.3. The outputs of this scaling circuit are applied as
one input vector to an array of exclusive OR gates with
the other, common, input being the output of the MOS shift
register at the CCD output shown in figure A.1. In the 2's
complement coding in use here this gate array performs a 1
bit by 16 bit multiplication giving the result 2ue(t)sgn(s
(t-n)) which is the desired weight vector increment. The
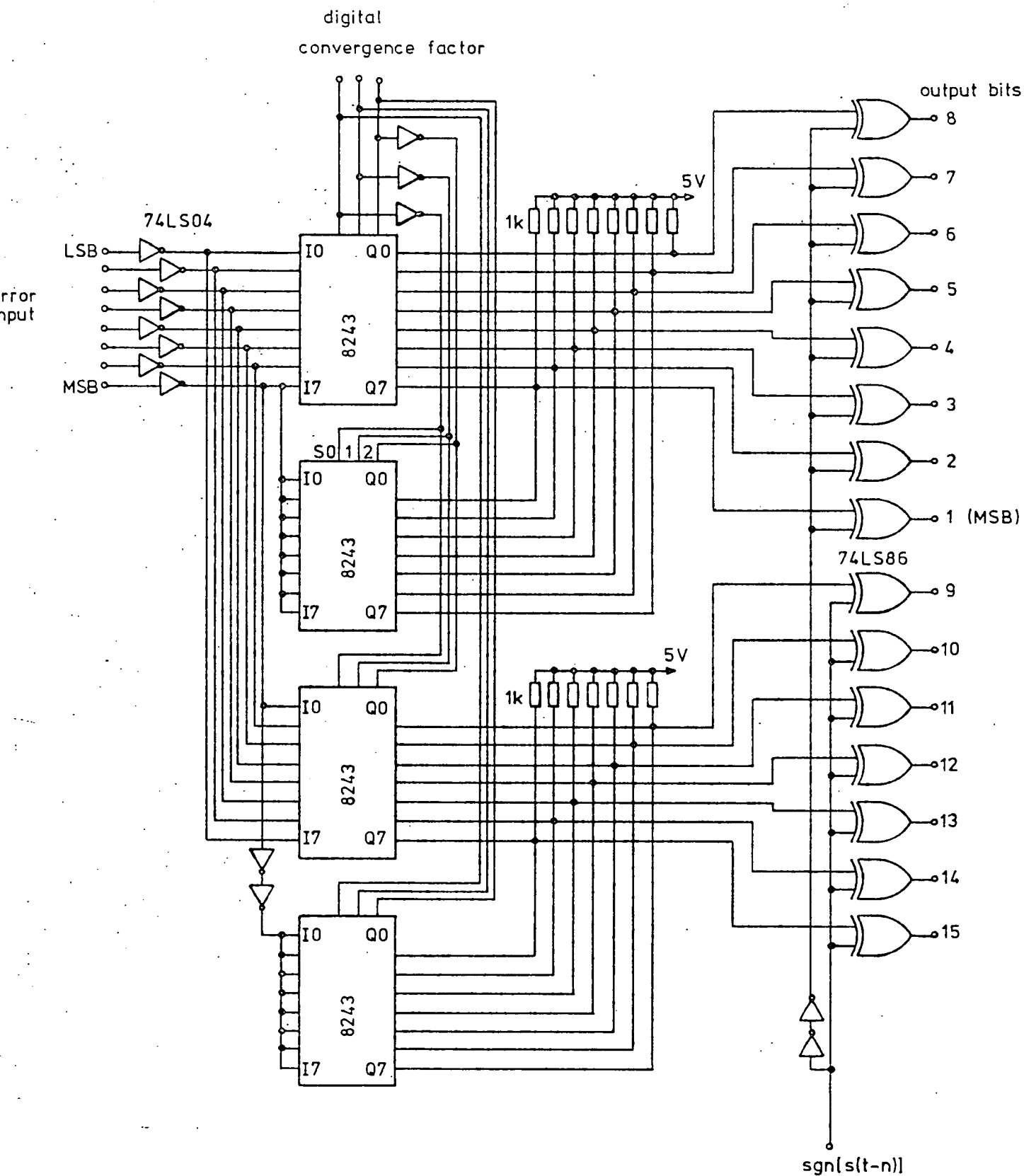reason for having the multiplication after the scaling

Figure A.3: Digital scaling and multiplication circuit.

function is in order to avoid the scaling circuit delay at every increment.

The final circuit diagram shown in figure A.4 shows the weight storage and integration circuit. Weight storage is achieved using 5 TMS4043 256x4 bit MOS RAMs giving a weight word resolution of 20 bits. The outputs of these RAMs supply one port of a 20 bit TTL adder using 5 74LS83 4 bit adders. The other input to these adders is the weight increment resulting from the circuit in figure A.3. The most significant bit of this increment occurs on the 6th most significant bit of the adder input giving the maximum convergence factor of 2^{-5}. The outputs of these adders are made to be saturating by incorporating an overflow detection circuit and a secondary set of output latches which always hold a saturated value for the adder output (either positive or negative, as appropriate). The appropriate latch output is then enabled to form the saturating output according to the state of the overflow detector output. The output of these latches is then fed back into the RAMs during the write cycle before the next RAM address change. At the same time the 10 most significant bits of the adder outputs are taken to represent the new value for the weight at the address indicated by the present RAM address. These 10 bits supply the input to a 10 bit DAC which provides the analogue weight input to the PTF, shown in figure A.1.
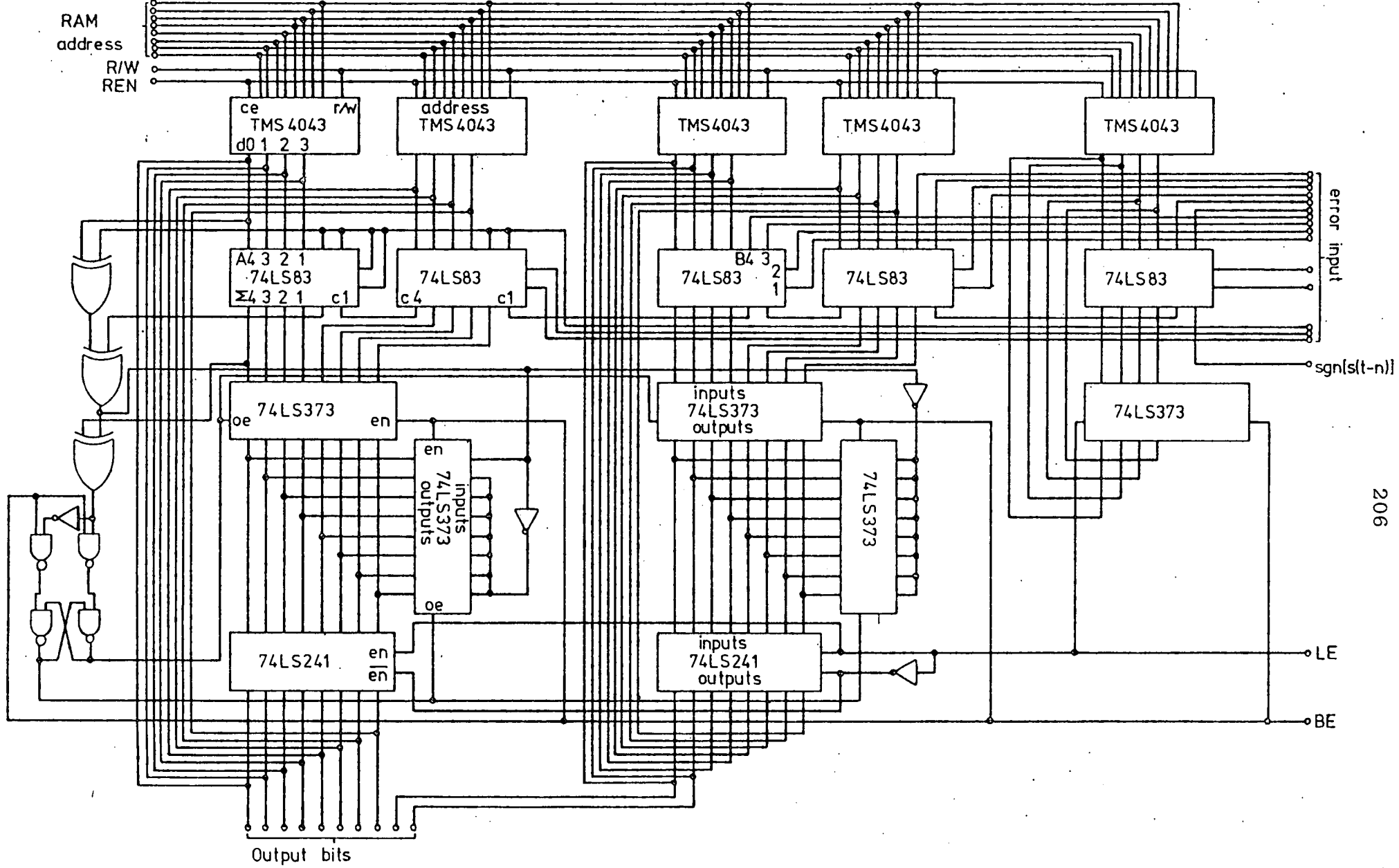
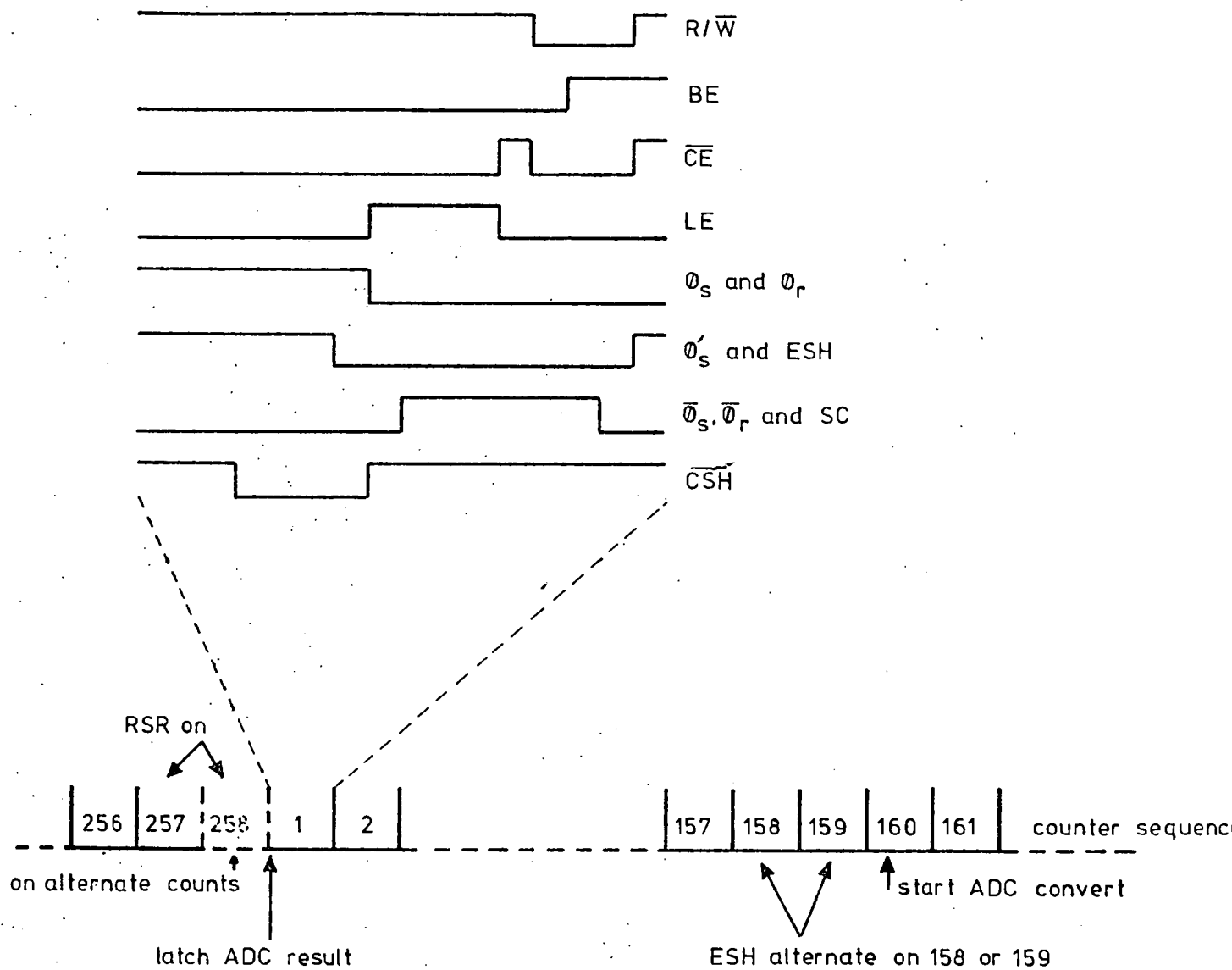Figure A.4: Digital storage and integration circuit.

Figure A.5: Detailed system clocking diagram.

APPENDIX B: RELEVANT PUBLICATIONS

1. *Arthur, J.W., Cowan, C.F.N., Mavor, J.; "Novel
    linear regression analyser using analogue CCD",
    Electronics Letters, 24, (13), pp. 751-753

2. *Cowan, C.F.N., Mavor, J., and Arthur, J.W.;
    "Implementation of a 64-point adaptive filter using
    an analogue CCD programmable filter", Electronics
    Letters, 14, (17), pp. 568-569

3. Cowan, C.F.N., Mavor, J., and Arthur, J.W.; "Noise
    cancellation and inverse filtering using a compact,
    high-performance CCD adaptive filter", Electronics
    Letters, 15, (1), pp. 35-37

4. *Geddes, R.C., Cowan, C.F.N., Mavor, J., and Dickson,
    J.F.; "Adaptive cancellation of fixed-pattern noise
    in CCD serial-parallel-serial memory", Electronics
    Letters, 15, (16), pp. 505-507

5. Denyer, P.B., Mavor, J., Arthur, J.W., and Cowan,
    C.F.N.; "A programmable CCD transversal filter:
    design and application", Proc. 1978 Int. Conf. on
    'The Application of CCDs', San Diego, pp. 3B-11 –
    3B-21

6. Cowan, C.F.N., Arthur, J.W., Mavor, J., and Denyer,
    P.B.; "CCD-based analogue adaptive processing", Proc.
    1979 Int. Conf. on CCDs, Edinburgh, pp. 334-340

7. *Cowan, C.F.N., Mavor, J., Arthur, J.W., and Denyer,
    P.B.; "An evaluation of analogue and digital adaptive
    filter realisations", IEE Int. Spec. Sem. on 'Case
    Studies in Advanced Signal Processing', Peebles,
    Scotland, 1979, pp. 178-183

8. *Arthur, J.W., Cowan, C.F.N., and Mavor, J.;
    "Analogue real-time statistical analyser utilising
    CCDs to implement a decision algorithm", IEEE Trans.
    Ints. and Meas., 1980, 29, pp. 58-66

9. Mavor, J., Denyer, P.B., Arthur, J.W., and Cowan,
    C.F.N.; "A monolithic programmable transversal filter
    for analogue signal processing", the Radio and Elec.
    Eng., 1980, 50, pp. 213-225

10. *Cowan, C.F.N., and Mavor, J.; "Miniature CCD-based
    analogue adaptive filters", IEEE ICASSP-80, 1980,
    Denver, pp. 474-478

11. Cowan, C.F.N., and Mavor, J.; "A 256-point adaptive
    filter: design and performance evaluation", Proc.
    IEE, CRSP, 127, 1980, pp. 179-184

12. *Cowan, C.F.N., and Mavor, J.; "Analogue signal
    processing using charge-coupled devices", Proc.
    EUSIPCO 80, (to be published, Sept. 1980)

* Papers reprinted in this appendix.

NOVEL LINEAR REGRESSION ANALYSER USING ANALOGUE C.C.D.

Linear regression analysis is normally implemented by using digital computing techniques. The letter presents a system to perform the regression function by using analogue c.c.d. components, giving a real-time output at the sampling rate. Results are presented to illustrate the principles of the system.

Introduction: A common requirement in signal processing is that of detecting a slowly varying unknown signal against a background of noise; this letter describes a novel approach to this problem, based on analogue c.c.d. transversal filters.[1] In a typical situation, if a system of lowpass filtering and threshold detection is used, there is the problem of matching the threshold to the noise level, and a slowly changing baseline upsets the process entirely. If, however, several successive samples of the signal are available over the region of interest, statistical techniques such as straight-line fitting (linear regression analysis) provide an approach that makes the most use of the available information.

General theory: For a signal y having values y_n at the sampling times t_n, and approximated by $y = mt_n + c$, where m is the time derivative and c is a constant baseline, formulas exist giving the values of m and c for a best fit.[2,3] The equations may be greatly simplified when the samples are evenly spaced in time, say by τ, and $t = 0$ may be arbitrarily set at the centre sample. In this case, m and c are given by

$$\left. \begin{array}{l} m = \dfrac{3}{p(p+1)(2p+1)\tau} \displaystyle\sum_{n=-p}^{p} ny_n \\[4mm] c = \dfrac{1}{2p+1} \displaystyle\sum_{n=-p}^{p} y_n \end{array} \right\} \quad (1)$$

where $2p + 1$ is the total number of samples (odd). The variances of these quantities are given by

$$\sigma_c^2 = \frac{p+1}{2}\left\{\frac{1}{p(p+1)}[(\overline{y^2}) - c^2] - m^2\tau^2/3\right\} \quad (2)$$

$$\sigma_m^2 = \frac{3}{2p}\left\{\frac{1}{p(p+1)\tau^2}[(\overline{y^2}) - c^2] - m^2/3\right\} \quad (3)$$

where

$$\overline{y^2} = \frac{1}{2p+1}\sum_{n=-p}^{p} y_n^2$$

Heuristically, one can reason that the value of $|m|$ has to be significantly greater than its probable error σ_m in order to register a rate of change in the data and hence the presence of a time-varying signal. In fact, the inequality

$$T(m, \beta) = \frac{1}{p(p+1)\tau^2}[(\overline{y^2}) - c^2] - \beta m^2 < 0 \quad (4)$$

which is similar to eqn. 2, apart from the factor β, determines when m is significantly different from zero. The value of β is obtained from the appropriate 'student-t' test with parameters m and σ_m. The appropriate value may, however, be readily found empirically. In the example stated below, for 27 samples at the 1% level of significance, β is approximately 4·2. This test is independent of τ, which may be arbitrarily set at unity, resulting in a simple thresholding method which automatically adapts to the noise level.

Experimental realisation: The above equations reduce to the manipulation of weighted sums of sampled data, for which the arithmetic may be performed by c.c.d. transversal filters, giving a real-time output at the sampling rate. Such a system, based on 27 data points and having a variable sampling rate, is now described. It comprises a number of interesting and potentially useful elements which could function individually

or together as

(*a*) an averager (unweighted)

(*b*) a differentiator (least-squares weighted)

(*c*) a mean-square noise element

(*d*) a significance tester or threshold-level detector.

A block diagram of the experimental system is shown in Fig. 1. The weights for the f.g.r. tap c.c.d. transversal filters[1] were realised by conductances chosen in accordance with eqn. 1. An output corresponding to one of eqns. 2–4 is obtained by the choice of the weights A, B and C in the final summing circuit.

Since the variance is, in fact, the mean-square variation of the signal (by definition), a graph of the square root of the variance output against r.m.s. noise input should yield a linear relationship. This was plotted by using a 31 bit pseudorandom binary sequence as the noise source, and the result is shown in Fig. 2a. It can be seen that, within the limits of reading errors, a linear relationship was maintained.
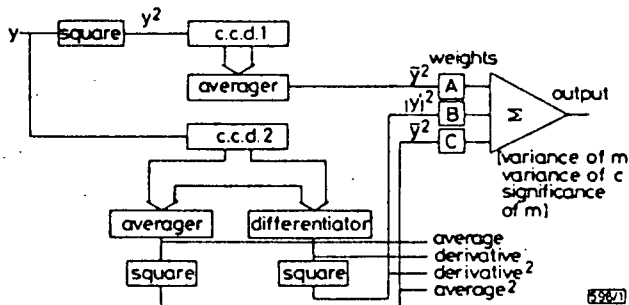

Fig. 1 *System block diagram*

By varying the weights A, B and C different outputs may be obtained, as indicated
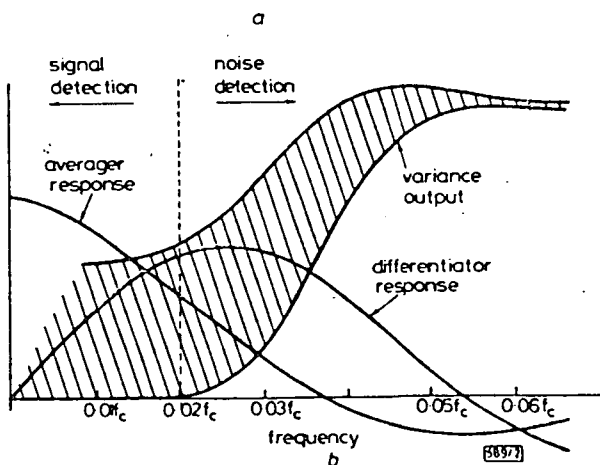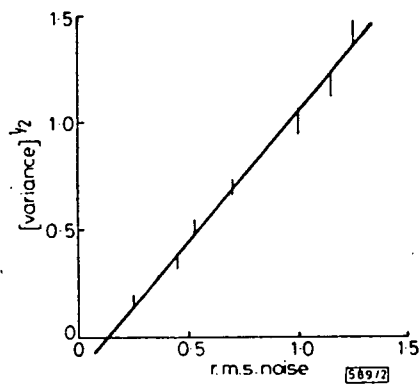

Fig. 2

(*a*) Plot of the square root of the variance output against the r.m.s. noise amplitude at the input

(*b*) Plot of the averager and differentiator responses and the variance output against frequency for a linear f.m. chirp input swept from d.c. to 0·06f_c

Fig. 2b shows plots of the averager and differentiator responses obtained from the experimental system when the input signal is a sinewave swept from d.c. to $0.06f_c$, where f_c is the c.c.d. sampling rate. These responses closely follow the

accurate squaring circuits and an improved gain structure. The operation could also be extended to include parabolic or higher-order components by the addition of further delay-line sections dealing with y^3 and higher orders. In certain applica-
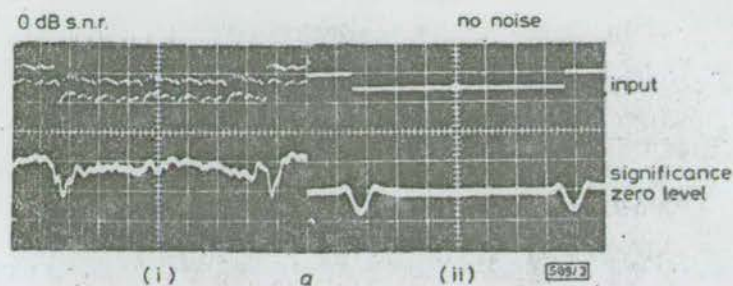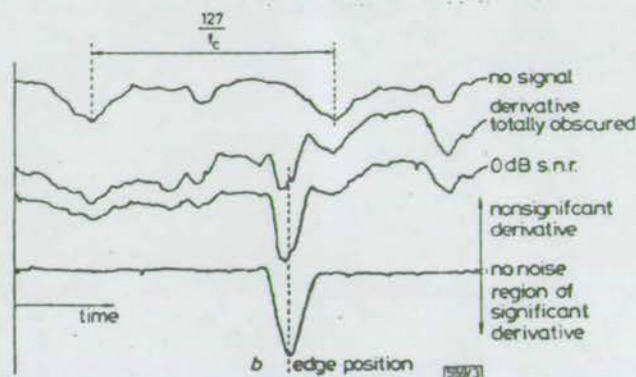

Fig. 3

(a) Outputs showing the derivative level of significance for a square-wave input having (i) a 0 dB signal to noise ratio and (ii) no noise. Timebase 2 ms/division, signal 0·5V/division, output 0·2V/division

(b) Outputs for a step waveform with various levels of noise, showing the level of significance of the derivative

theoretically predicted results, such that the averager and differentiator responses closely approximate the sinc x function and its derivative. An envelope of the variance output is also shown, demonstrating a relatively constant output at frequencies above about $0.044f_c$. Thus noise frequencies which will give a predictable d.c. level at the variance output, without spurious periodic components, range from about $0.044f_c$ to the Nyquist $f_c/2$, and the actual signal which can be dealt with lies below $0.02f_c$. The different operating regions for signal and noise are shown in Fig. 2b.

Finally, Fig. 3 shows the type of output which is obtained when the system is set up to detect $T(m, 4·2)$, the significance of the derivative output at the 1% level. In Fig. 3a, outputs are shown for the case where no noise is present and for the case where the input signal/noise ratio is 0 dB. In the latter example, it can be seen that the significance peak just cuts the zero level established in the previous case.

Fig. 3b shows a series of outputs which have been plotted on a pen recorder. The noise source used here was a 127 bit pseudorandom sequence filtered to give about 10 dB rejection at $0.04f_c$. Periodic components of the noise below $0.04f_c$ are therefore present at the output and are clearly distinguishable in the top trace. Computer simulation of this situation has proved that the patterns obtained can, in fact, be wholly accredited to the p.n. sequence. By comparison with the other traces, it may be seen that this pattern does, in fact, recur when the step signal is also present. The signal used was a square wave at $0.001f_c$.

Conclusions: It has been shown that analogue transversal filtering can be successfully applied to the problem of constructing a 'real-time' linear regression analyser. The performance of the system could be upgraded by the use of more

tions, the noise statistics are independent of time and circuit simplification may result. A potential application to peak detection would result in applying the algorithm given in Reference 2 to the significance output.

J. W. ARTHUR   13th October 1977

Wolfson Microelectronics Liaison Unit
University of Edinburgh
Mayfield Road, Edinburgh EH9 3JL, Scotland

C. F. N. COWAN
J. MAVOR

Electrical Engineering Department
University of Edinburgh
Mayfield Road, Edinburgh EH9 3JL, Scotland

References
1 MACLENNAN, D. J., MAVOR, J., and VANSTONE, G. F.: 'Technique for realising transversal filters using charge-coupled devices', *Proc. IEE*, 1975, **122**, (6) pp. 615–619
2 ARTHUR, J. W.: 'Digital processing of spectra', *J. Raman Spectrosc.*, 1976, **5**, pp. 9–20
3 WEATHERBURN, C. E.: 'First course in mathematical statistics', (Cambridge University Press, 1952), pp. 194–195

Analog Real-Time Statistical Analyzer Utilizing CCD's to Implement a Decision Algorithm

JOHN W. ARTHUR, COLIN F. N. COWAN, AND JOHN MAVOR, MEMBER, IEEE

Abstract—A system is presented to realize the function of a linear regression analyzer using analog charge-coupled device (CCD) tapped delay lines to perform fast parallel processing. Such systems are normally implemented using digital computing techniques giving a nonreal-time output. The analog approach discussed in this paper is capable of providing results in real time at the sampling rate of several hundred kilohertz, with the added advantage of reduction in power consumption and physical size. Results obtained from a prototype system are presented to demonstrate the principles of the system operation.

I. INTRODUCTION

A COMMON PROBLEM encountered in signal processing is that of detecting a signal against a background of noise. If the expected signal waveform is known, then matched filtering techniques may be used, and if the expected signal is a precisely known pattern (e.g., a PN sequence) then the optimum detection technique is correlation. However, if relatively little information is available about the expected signal (that is, it is restricted within a certain frequency band but may assume any combination of frequency components within that bandwidth) then the problem becomes more complex. For instance, for events in the time domain using conventional filtering and thresholding techniques the threshold must be matched to the noise level which can vary making the set threshold invalid, and a slowly varying baseline also adds to this problem.

The ideal system would be one which is capable of evaluating the mean-square noise level and using this figure to automatically adjust the thresholding level to attain a constant error rate. A slowly varying baseline would still lead to detection problems but this could be overcome by evaluating the signal validity from its derivative rather than its amplitude, in which case dc and low-frequency noise would have no effect on the detection system. A block diagram of this basic idealized system is shown in Fig. 1.
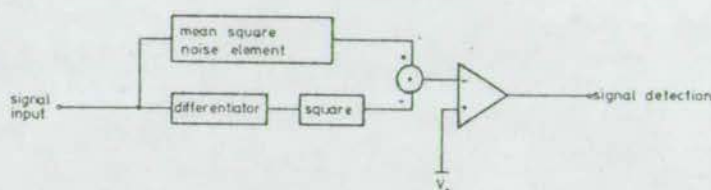
Fig. 1. Block diagram of an idealized nonmatched filter system.

This type of signal analysis is ideally suited to the use of statistical techniques and the system presented in this paper is based on linear regression analysis [1]. This technique [2] has been used before in the form of a computer program for the "nonreal-time" analysis of experimental data. The system described here, based on analog charge-coupled device (CCD) tapped delay lines, implements similar algorithms providing a "real-time" output at the sampling rate [3]. Certain restrictions on bandwidth and accuracy do exist using this method and these are described in greater detail in Section III, devoted to systems design.

II. THEORETICAL ANALYSIS

A slowly varying signal y having values y_n at times t_n may be estimated over a given interval by the equation $y = mt_n + c$, where m and c are the derivative and average of the signal, respectively. The error in the value of y, at any given time, may be expressed as

$$e_n = y_n - y$$
$$= y_n - (mn\tau + c)$$

where the samples are evenly spaced in time by intervals τ. The sum of the squared errors over a fixed number of samples $(2p + 1)$ is then given by

$$S^2 = \sum_{n=-p}^{p} e_n^2 = \sum (y_n - (mn\tau + c))^2. \tag{2.1}$$

Using the least-squares method, the best estimates of the quantities m and c are given by minimizing the sum of the errors squared $\sum e_n^2$, i.e.,

$$\frac{\partial S^2}{\partial m} = \frac{\partial S^2}{\partial c} = 0.$$

Hence

$$m = \sum_{n=-p}^{p} n\tau y_n / \sum_{n=-p}^{p} (n\tau)^2 = y' \qquad (2.2)$$

$$c = \sum_{n=-p}^{p} y_n / (2p + 1) = \bar{y}. \qquad (2.3)$$

The minimized form of (2.1) is then

$$S^2 = \sum y_n^2 - m^2 \sum (n\tau)^2 - c^2 (2p + 1) \qquad (2.4)$$

where m and c are given by (2.2) and (2.3). Assuming the data to be normal with standard deviation σ, σ^2 may be estimated by

$$\sigma^2 = \frac{\sum (y_n - y)^2}{2p}.$$

and the variances of m and c are given by

$$\sigma_m^2 = \frac{\sum (n\tau)^2 \sigma^2}{(\sum (n\tau)^2)^2} = \frac{\sigma^2}{\sum (n\tau)^2}$$

$$\sigma_c^2 = \frac{\sum \sigma^2}{(2p + 1)^2} = \frac{\sigma^2}{2p + 1}.$$

Hence

$$\sigma_c^2 = \frac{p+1}{2} \left\{ \frac{1}{p(p+1)} [(\overline{y^2}) - c^2] - \frac{\tau^2 m^2}{3} \right\} \qquad (2.5)$$

$$\sigma_m^2 = \frac{3}{2p} \left\{ \frac{1}{p(p+1)\tau^2} [(\overline{y^2}) - c^2] - \frac{m^2}{3} \right\} \qquad (2.6)$$

where

$$\overline{y^2} = \frac{1}{2p + 1} \sum_{n=-p}^{p} y_n^2$$

i.e., the variances of m and c are connected simply by a factor of proportionality $\frac{1}{3} p(p + 1)\tau^2$.

Intuitively, it may be argued that the value of m^2 must be greater than its probable error σ_m^2 to register a significant rate of change in the data, and hence the presence of a signal (cf. Fig. 1). More rigorously, the significance of the regression coefficient m may be determined by the application of the "Student-t" test of significance to the derivative, i.e.,

$$t = m [(2p - 1) \sum (n\tau)^2 / \sum (y_n - y)^2]^{1/2}$$

or

$$\sum (y_n - y)^2 = \frac{m^2 (2p - 1) \sum (n\tau)^2}{t^2}$$

where the value of t may be found from the Student-t table, dependent on the chosen level of significance. Using (2.4) an inequality

$$T(m, \beta) = \frac{1}{p(p+1)^2} [(\overline{y^2}) - c^2] - (\beta + 1) \frac{m^2}{3} < 0 \qquad (2.7)$$

may be determined which indicates when a signal is significant at the chosen level. Equation (2.7) is similar to (2.6) except for the factor β where
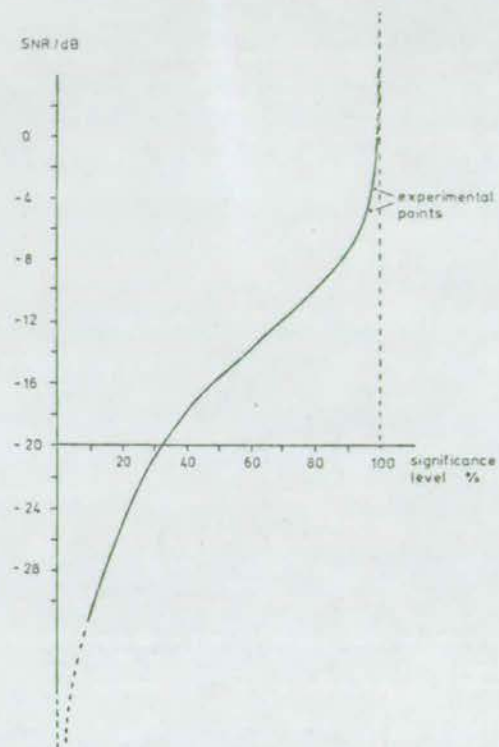
$$\beta = \frac{2p - 1}{t^2}.$$

Fig. 2. Graph of predicted noise rejection plotted against significance level for a signal at $0.01 f_c$ and simulated noise at $f_c/2$.

This test is independent of the value of τ, which may be arbitrarily set at unity. We now have a test which yields a signal detection system which automatically compensates for the noise level, the function which cannot be carried out using conventional filtering and thresholding techniques.

A number of output parameters may be readily evaluated from the previous equations. For instance, it can be predicted that the averager and differentiator frequency responses will closely follow the sinc (x) function and its derivative. Further, the averager response will have a maximum at dc and its first null when one complete cycle is stored in the CCD register (i.e. at $0.037 f_c$, where f_c is the sampling rate).

It is also a simple matter to calculate the maximum processing gain (minimum signal-to-noise ratio) for the system by the choice of two distinct waveforms representing maximum signal and maximum noise, respectively. Furthermore, if the noise signal is chosen so that it affects only the $\overline{y^2}$ value and the signal has no effect on the variance then the calculation is greatly simplified. Examination of the relevant variance and significance formulas confirms that these criteria are satisfied for the case where the noise signal is a Nyquist-rate square wave and the signal is another square wave at less than $0.018 f_c$. The processing gain may then be evaluated for various levels of significance. Fig. 2 shows a plot of minimum signal-to-noise ratio against significance level for the stated case. It should be noted that the processing gain is critically dependent on the signal and noise spectral characteristics and the case presented here is for *maximum* processing gain.

In all instances shown in Fig. 2, where the minimum signal-to-noise ratio is less than 0 dB, a white noise background may not be used since components of the noise spectrum in the signal detection region will yield a significant output. Therefore, for a white noise background it is necessary to use a value of β which gives a minimum signal-to-noise ratio of 0 dB, or greater, at the input. In the examples stated in Section IV the significance level used was the 1-percent level and hence the noise spectrum was limited to about $0.04f_c$ upward.

In the case where a white noise background may be used the components of noise in the signal region both add and subtract from the actual signal. Therefore, to ensure signal detection the minimum signal-to-noise ratio is raised to about 3 dB.

The operation of the basic system may be extended by the addition of a block averaging function at the input. This means that each data point at the input of the basic system is represented by the average of N points at the block averager input. Hence, the total number of data points used to derive a single output data point is $(2p + 1)N$ giving a consequent increase in system accuracy, provided that there is sufficient redundancy in the data to allow time compression in the block averager (i.e. the bandwidth is reduced).

The system bandwidth could be increased by using cubic and higher order terms in the regression calculation. This, however, reduces the measured level of noise since the signal bandwidth is increased, so reducing the proportion of the noise seen by the variance estimator.

III. SYSTEMS DESIGN

The implementation of the system described in the previous section is dependent on the ability to store several successive samples of the incoming data, and make them available for manipulation simultaneously, to aid speed of operation. In addition, the accuracy of the system is dependent on the number of samples used, and the bandwidth of detectable signals is dependent on the sampling frequency and the number of samples N such that

$$\text{Bandwidth} \propto f_c/N$$

$$\text{Accuracy} \propto \sqrt{N}.$$

Therefore, to make a fast versatile system with potentially programmable bandwidth the following attributes are required:

i) variable sampling rate,

ii) the availability of several successive samples simultaneously (i.e. serial to parallel operation),

iii) the ability to process varying numbers of samples.

One circuit which offers all of these features is the CCD analog tapped delay line, which was the basic circuit element used to implement the experimental system described in Section IV.

Certain alternatives to this structure could be used and these are listed below.

i) The use of continuous analog filters is possible if a high-pass filter is used to determine the average noise level
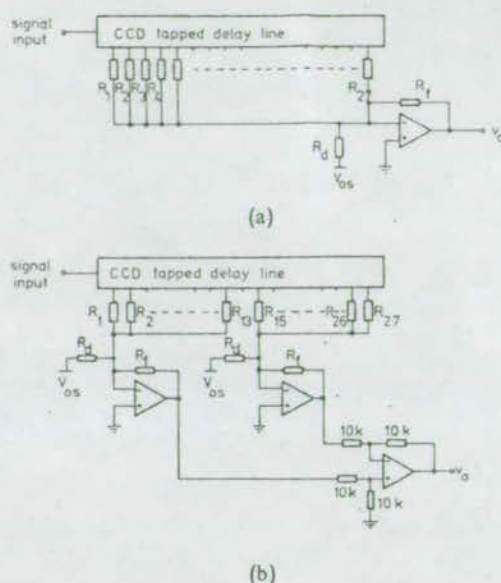

Fig. 3. (a) Circuit diagram of a basic CCD averager. (b) Circuit diagram of a CCD differentiator.

and set the threshold level, and the actual signal is determined from a combination of low-pass filtering and differentiating. This method, however, would not be a direct implementation of the statistical analysis presented in Section III and would, in addition, yield a system of fixed bandwidth and accuracy.

ii) The system could be implemented using a microprogrammed digital system. However, the normal MPU instruction execution time of about 10 μs means that a long series of complex operations may take a considerable time to execute. For the function described here it is necessary, for each output point, to store N input samples, and their squares, averages, and derivatives must be found, and further squaring operations must be implemented after this point. Given the amount of computations just listed a MPU would probably be capable of delivering one output point per t seconds, where t is of the order of one (i.e. the second criterion of serial to parallel operation is violated).

iii) A hardwired digital system could be used providing the availability of parallel operations with a consequent increase in speed. However, digital operations again impose serious restraints on bandwidth unless hardware multiplying circuits are used, in which case it is possible that a digital system could operate much faster than the analog equivalent. However, the cost of such a system would be considerable.

IV. PRACTICAL REALIZATION

The delay lines used in the experimental system to be described here were analog CCD registers with 27 separately accessible taps using floating gate reset (FGR)[1] tapping [4].

[1] FGR tapping is a scheme which allows a signal to be extracted nondestructively from the CCD delay line at any point by maintaining one of the transfer gates at an arbitrary "floating" voltage which is periodically reset. A full explanation of the technique may be found in reference [4].
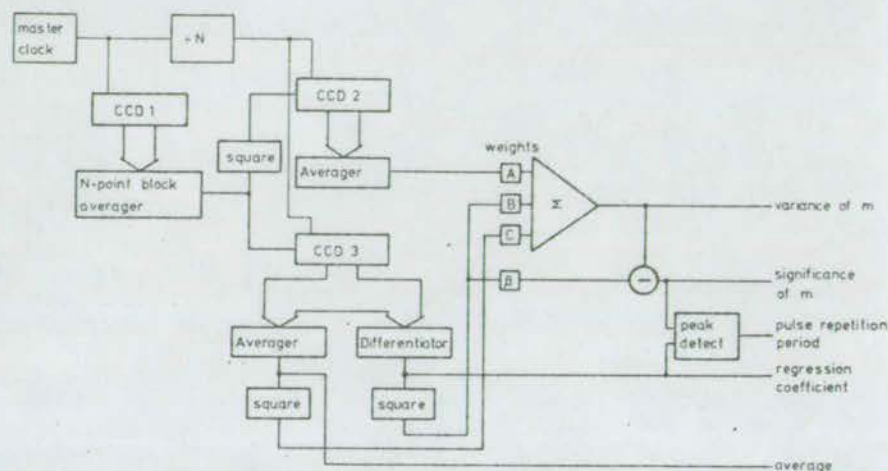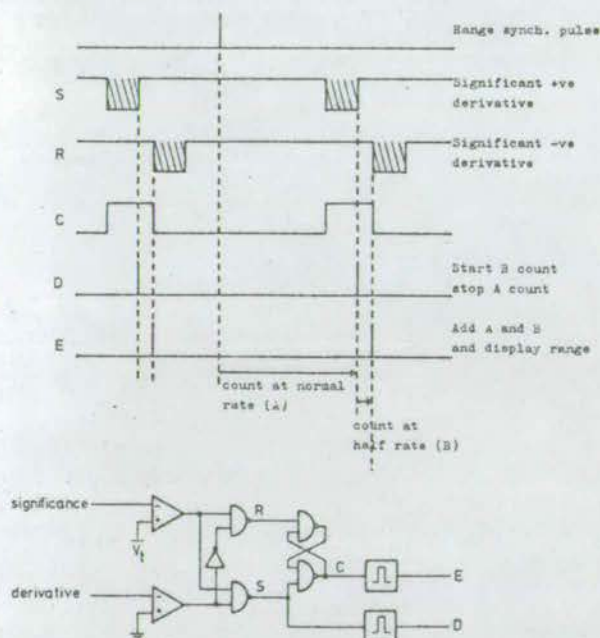
Fig. 4.   System block diagram.

Fig. 5.   Circuit for the peak detection system and examples of some basic waveforms.

The use of these devices means that the averaging and differentiating functions described mathematically in (2.2) and (2.3) may be implemented directly by weighting the tap outputs and summing [5]. Circuit diagrams demonstrating the circuits used to implement these functions are shown in Fig. 3. The resistor values R_1 to R_{27} may be evaluated directly from (2.2) and (2.3). In Fig. 3(a) the averager weighting resistors are all identical and the differentiator weights (Fig. 3(b)) form a linear ramp with R_{14}, being the center point, not contributing.

A complete block diagram of the experimental system is shown in Fig. 4. It can be seen from this that the system is a *literal* interpretation of the variance and significance formulas, making system performance exactly predictable. However, the variance output is filtered before application

to the significance testing circuit to eliminate variance errors. This means that the variance is averaged over a greater number of data points than the basic 27 and the significance testing circuit now more closely resembles a "χ^2" test than a "Student-t" test of significance. Nevertheless (2.7) may still be applied; and, in this case, for 27 data points at the 1-percent significance level the value of β is approximately 4.2.

Although the significance output gives a positive indication of a valid signal at the input, it conveys only the information that a significant derivative is present, while no information on the polarity of that derivative is available. If, however, the significance output is used in conjunction with the derivative output then significant positive and negative derivatives may be separately resolved. This information may then be applied to indicate the presence of a significant signal peak. A circuit diagram and appropriate waveforms for this function are shown in Fig. 5.

Information derived from this circuit may further be used to find the peak position [2] and the peak repetition period. In the experimental system the peak position was taken as the point half-way between the last significant positive derivative and the first significant negative derivative. The peak repetition period may then be calculated by counting between these values. It is possible to initiate this count sequence from any arbitrary point so if the peak position with reference to an arbitrary synch pulse were required (i.e. a range figure) then the algorithm could be easily modified to provide this facility. A flow diagram illustrating the peak repetition period algorithm is shown in Fig. 6. This was implemented in the system by causing a counter to count every output sample starting from the synch pulse. When the D pulse comes on this figure is preset into a second counter, counting every other sample, and the result is displayed by the E pulse coming on.

The block averager at the system input was constructed in a similar way to that shown in Fig. 3 but with variable numbers of weighted tap outputs switched through to the summing circuit (cf. Fig. 7).
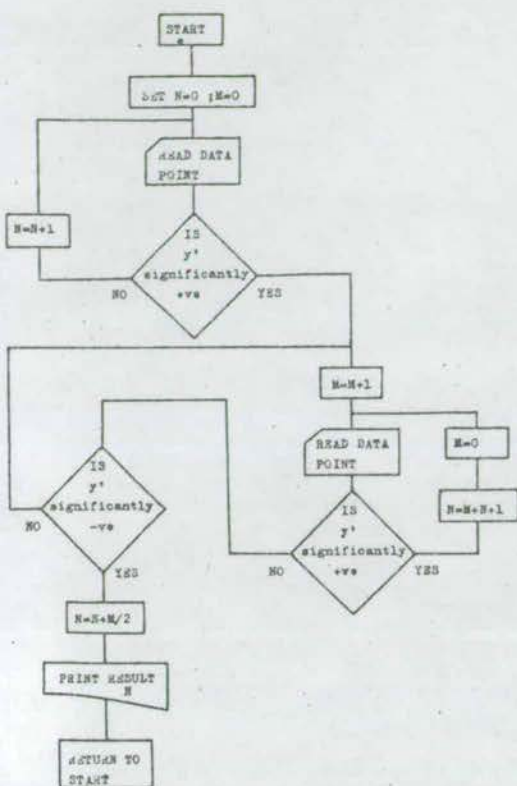
Fig. 6.   Flow diagram illustrating the algorithm used to calculate the peak repetition period.

Fig. 7.   Circuit diagram of the block averager.

The experimental system was constructed as a modular system in a 19″ rack. Fig. 8 shows a photograph of the complete system and also a photograph of one of the CCD tapped delay lines. The complete system as described had an overall power requirement of 12 W. The CCD processing elements, however, individually consume 500 mW, whereas it is estimated that an equivalent digital realization would require 50 times as much.

V. EXPERIMENTAL RESULTS

Fig. 9 shows a series of photographs demonstrating the basic system outputs for a triangular wave input. It should be noted that the differentiator weighting does not make full use of the possible system bandwidth but is rather a least-square estimate.

The photograph in Fig. 10 shows the variance output for a 31-bit pseudorandom sequence based on f_c as the input. The irregular pattern obtained in this case was matched precisely by a computer simulation of the theoretical system (also shown) demonstrating that this output pattern may be entirely accounted for by the known statistics.

Since the variance is (by definition) the mean-square variation of the signal, then a graph of the square root of the variance plotted against the rms noise input should yield a linear relationship. This was plotted using the 31-bit pseudorandom sequence as the noise source and the result is shown in Fig. 11. It can be seen that, within the limits of reading error, a linear relationship was maintained.

Fig. 12 shows a plot of averager and differentiator frequency responses for a sine wave swept approximately from
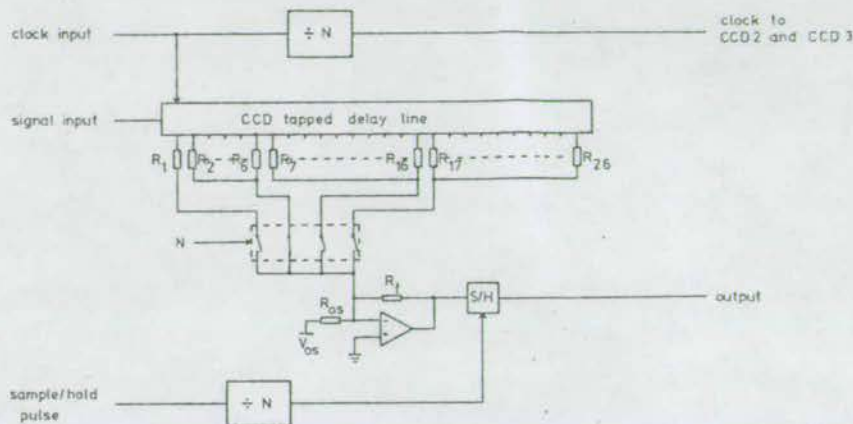
dc to $0.06f_c$. These responses display the expected sinc (x) shape and its derivative (cf. Section II).

Also shown in Fig. 12 is the envelope of the unfiltered variance output, for the same input, demonstrating a relatively constant output above about $0.04f_c$. Hence, frequencies above this point are detected as noise yielding a predictable dc offset at the variance output and signals below this frequency are regarded as valid signals (weighted according to their derivative). The photograph in Fig. 13 shows the filtered version of the variance for an input sine wave swept from dc to Nyquist $(f_c/2)$. This demonstrates the consistent output maintained by the variance above $0.04f_c$.

Fig. 14 shows some typical significance outputs for a triangular pulse, first with no noise and then with an estimated 0-dB signal-to-noise ratio. In this case the noise source used was a 127-bit pseudorandom sequence filtered to give about 10-dB rejection at $0.04f_c$. Therefore, noise components are present in the signal region and these may be clearly seen in Fig. 14 although they never reach the detection threshold level. It can be seen from these outputs that even with a considerable noise background the significant peaks still reach above the set threshold level and a consistent peak detection output is available. The peak detection output shown corresponds to the complement of that marked C in Fig. 15.

Since the significance output is primarily dependent on the derivative squared it naturally follows a similar frequency response. Fig. 15 shows the significance output for three different sinusoidal pulsewidths showing that an optimum sinusoidal pulsewidth exists at about $1/0.05f_c$, and frequencies above and below this level yield degraded noise rejection performance.

The use of the block averager may be demonstrated by reference to Fig. 16 where two examples of the significance output are shown for $N = 1$ and $N = 5$. In each case the same input signal (a sinusoidal pulse) and noise sequence (the 127-bit pseudorandom sequence) were used. The improvement in noise rejection may be clearly seen here.

A major drawback in the system described here was a low dynamic range which was measured to be about 25 dB at the
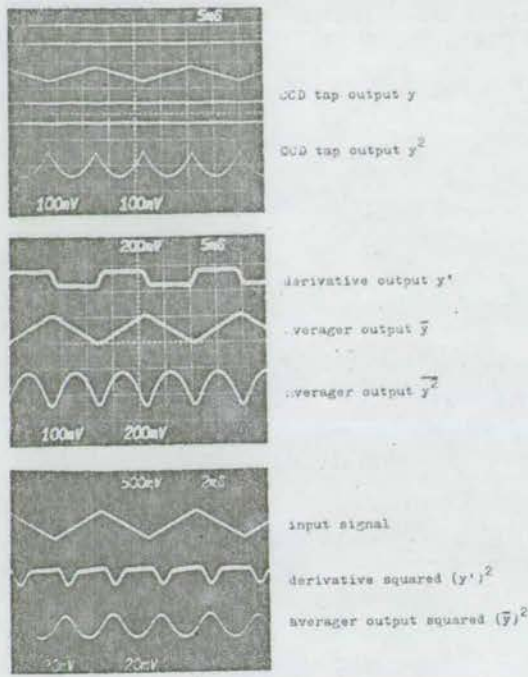
(a)

(b)

Fig. 8. (a) Photograph showing the complete experimental system. (b) Photograph showing surface detail of one of the CCD FGR tapped delay lines.

input. This is due mainly to the fact that the signal is squared before entering CCD2 which results in halving the available input dynamic range (the range of the CCD register is approximately 50 dB) [6], i.e., for normal operation, the dynamic range is given by

$$20 \log_{10} \left(\frac{A}{B} \right) = 50 \text{ dB}$$

but for the case where the signal is squared

$$20 \log_{10} \left(\frac{A'}{B'} \right)^2 = 50 \text{ dB}.$$

Therefore, at the input the dynamic range is approximately 25 dB.

Further, the maximum clocking frequency of the CCD is about 100 kHz yielding a maximum detectable frequency at about 500 Hz; at this frequency, with the block averager set at $N = 25$, the basic system is clocking at 4 kHz meaning that the total delay time in CCD2 and CCD3 is about 40 ms. This delay value results in a considerable contribution to the output from CCD dark current, which provides a cumulative effect along the line of taps. This is compensated for between the $\bar{y}^2$ and $\overline{y^2}$ outputs but it does contribute a significant dc offset to the derivative output since an offset-

CCD tap output y

CCD tap output y^2

100mV   100mV

derivative output y'

averager output $\bar{y}$

averager output $\overrightarrow{y^2}$

100mV   200mV

input signal

derivative squared $(y')^2$

averager output squared $(\bar{y})^2$

Fig. 9.   Basic system outputs for a triangular wave input.

CCD tap output y for a 31-bit
pseudo-random sequence input

Variance output

(a)

Computer simulation of variance
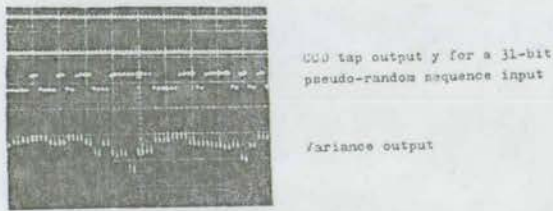output for theoretical system

(b)

Fig. 10.   (a) Variance output for a 31-bit pseudorandom sequence input.
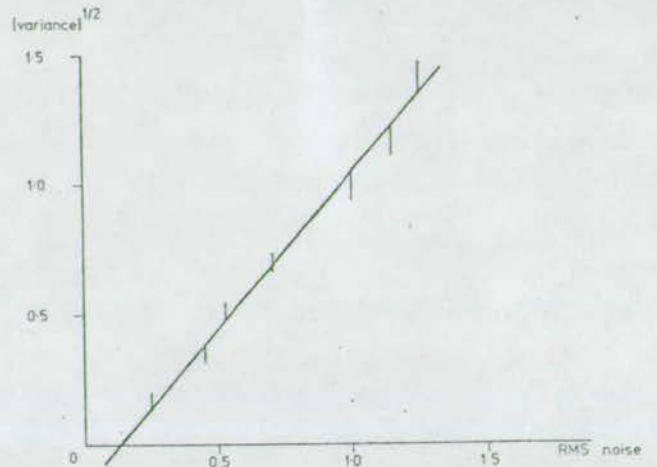(b) Computer simulation of the variance output for the above case.

Fig. 11.   Plot of rms variance output of the analyzer against rms noise
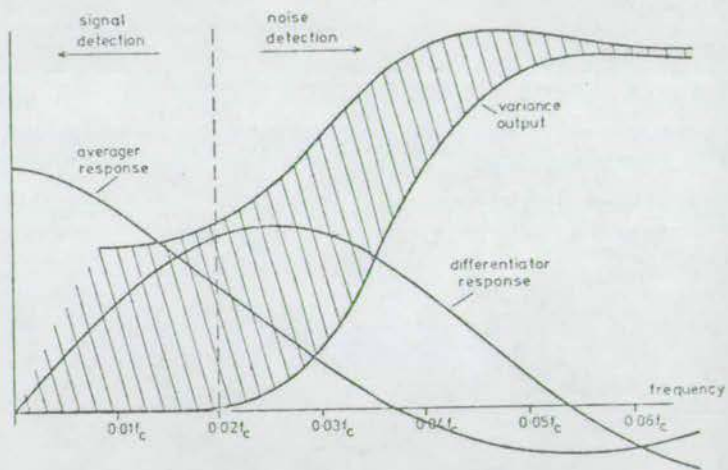input.

Fig. 12.   Averager and differentiator responses and variance output as a
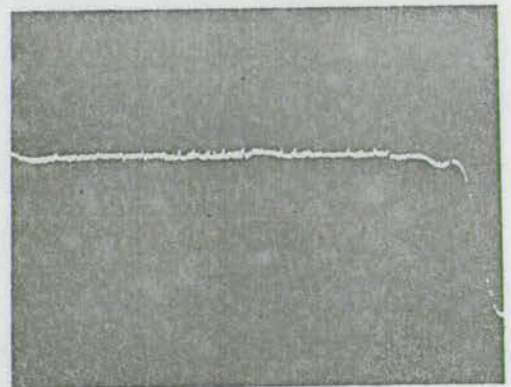function of frequency.

Fig. 13.   Filtered version of the variance output for a sine wave input
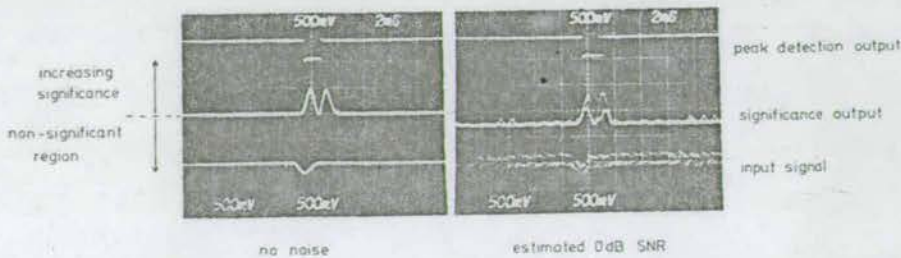swept from $f_c/2$ to dc.

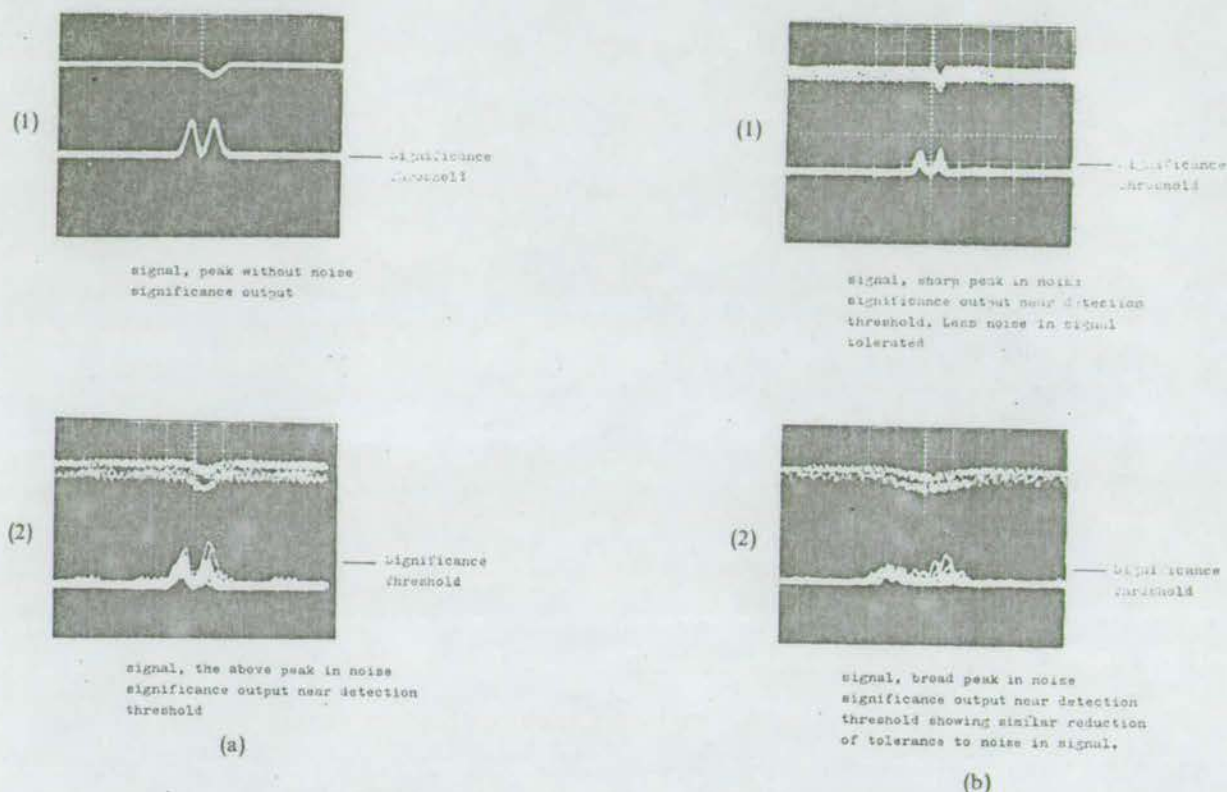Fig. 14.   Examples of the significance and major peak detection outputs.

signal, peak without noise
significance output

signal, the above peak in noise
significance output near detection
threshold

(a)

signal, sharp peak in noise
significance output near detection
threshold. Less noise in signal
tolerated

signal, broad peak in noise
significance output near detection
threshold showing similar reduction
of tolerance to noise in signal.

(b)

Fig. 15. (a) Peak detection with peak width matched to the filter length.
(b) Peak detection with wider and narrower peaks.

signal, peak in noise
significance output near detection
threshold. Block averager N=1

signal, peak in noise (identical to
above)
significance output displays greater
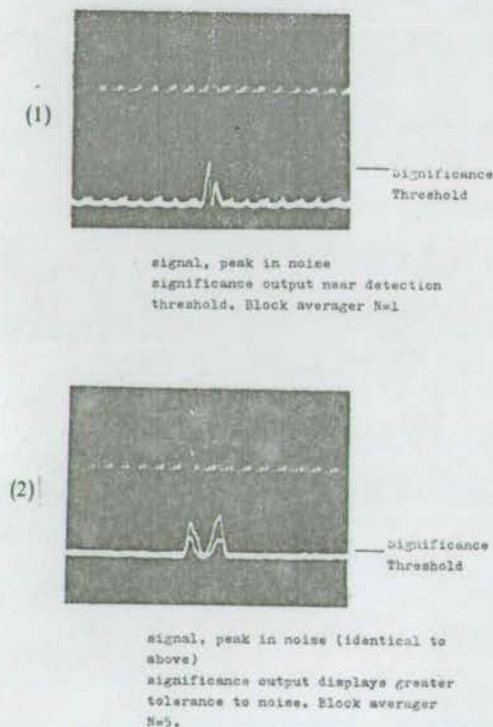tolerance to noise. Block averager
N=5.

Fig. 16. Significance outputs for identical signal and noise inputs with
the block averager set at $N = 1$ and $N = 5$.

ting ramp is produced from the input to output of the register CCD3, which produces a response near the maximum of the differentiator response curve but is not related to the signal input.

VI. Conclusions

It has been demonstrated in principle that it is feasible to apply statistical analysis techniques to the problem of nonmatched filtering, and that the use of these techniques has certain basic advantages over more conventional methods. These advantages mainly concern the automatic noise-controlled thresholding technique and the exclusion of low-frequency noise from the signal by the use of derivative detection rather than by the more normal amplitude detection.

In addition, it has been shown that charge-transfer devices provide a suitable basic building block for such a system providing for easy adjustment of the system characteristics. A small low power system of this type is ideally suited to use as a constant false alarm rate decision module following a noise reduction system, in particular, communications receiver applications (e.g., sonar-return processing). Applications to scientific instruments used inherently in "noisy" experiments are also a possibility, replacing nonreal-time techniques [2].

The system described here could be modified in a number of ways to extend the basic performance characteristics described in Section V.

i) Dynamic range could be extended to a considerable extent either by the use of compression techniques at the input to CCD2 or by performing the squaring operation not prior to input to CCD2 but afterwards at each tap output. Dynamic range would then be limited by the squaring circuits and not by the CCD dynamic range. A very simple

multiplier circuit using a single MOST has been reported in recent literature [7].

ii) Dark current effects in the differentiator circuit could be overcome by the use of an alternate-zero signal in the CCD3 register giving a reference level which could be subtracted from the final value so compensating for effects not attributable to the signal. The use of this technique would, however, halve the available maximum signal frequency.

iii) A more accurate indication of peak position could be obtained by the use of a more complex algorithm. Such a technique would use interpolation on actual values of the significant derivative on both sides of the peak.

iv) Microprocessor-controlled programmable transversal filters [8], [9] could be used to replace the existing averaging and differentiating circuits. A reduction in the number of units required would result from time multiplexing such a system, though in this case the bandwidth would be limited by the excess processing time required. Additionally, linearization of the impulse response could be used to match more precisely the averaging and differentiating weighting coefficients.

v) Given the system described in iv) it would be possible to scan the basic system clocks in frequency to optimize significant outputs at an arbitrary frequency.

ACKNOWLEDGMENT

The authors acknowledge the valuable contributions by P. B. Denyer, formerly with the Wolfson Microelectronics Institute (now with Denyer–Walmsley Microelectronics, Limited), and by Dr. D. J. MacLennan who designed the CCD delay lines.

REFERENCES

[1] C. E. Weatherburn, *First Course in Mathematical Statistics.* Cambridge, MA: Cambridge University Press, 1952, pp. 194–195.
[2] J. W. Arthur, "Digital processing of spectra," *J. Raman Spectrosc.*, vol. 5, pp. 9–20, 1976.
[3] J. W. Arthur, C. F. N. Cowan, and J. Mavor, "Novel linear regression analyser using analogue CCD," *Electron. Lett.*, vol. 13, pp. 751–753, 1977.
[4] P. B. Denyer and J. Mavor, "Design of CCD delay lines with floating gate taps," *IEE J. Solid-State Electron Dev.*, vol. 1, pp. 121–129, July 1977.
[5] D. J. MacLennan, J. Mavor, and G. F. Vanstone, "Technique for realising transversal filters using charge-coupled devices," *Proc. IEE*, vol. 122, pp. 615–619.
[6] W. R. Bennet, *Introduction to Signal Transmission.* New York: McGraw-Hill 1970, pp. 64–71.
[7] P. B. Denyer, J. W. Arthur, and J. Mavor, "Monolithic, programmable analogue CCD transversal filter," *Electron. Lett.*, vol. 13, pp. 739–741, 1977.
[8] J. E. Dilley, M. Naughton, R. C. S. Morling, G. D. Cain, and A. H. Abed, "A microcomputer-controlled adaptive CCD transversal filter," in *Proc. 3rd Int. Conf. Techn. Applications CCD's*, pp. 269–276, 1976.
[9] J. Mavor and P. B. Denyer, "Design and development of CCD programmable transversal filters," *Electron. Circuits Syst.*, vol. 2, pp. 1–8, 1978.

BIBLIOGRAPHY

The bibliography references are given as general background material which may be useful to the reader not acquainted with the technology or system concepts used in the text of this paper.
[1] W. S. Boyle and G. E. Smith, "Charge-coupled semiconductor devices," *Bell Syst. Tech. J.*, vol. 49, pp. 587–593, 1970.
[2] L. S. Schwartz, *Principles of Coding, Filtering and Information Theory.* Cleaver-Hume Press, 1963, pp. 157–158.

ADAPTIVE CANCELLATION OF FIXED-PATTERN NOISE IN C.C.D. SERIAL-PARALLEL-SERIAL MEMORY

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

The letter addresses the problem of the effective implementation of long analogue delay lines. Charge-coupled devices (c.c.d.) appear attractive for this application but suffer from the problem of signal smearing due to imperfect charge transfer. Although s.p.s. delay-line configurations minimise charge-transfer inefficiency, they emphasise the significance of dark current which can cause severe fixed-pattern noise problems in this architecture. A scheme is proposed here for the cancellation of fixed-pattern noise in s.p.s. delay lines and results are presented to demonstrate the effectiveness of the system.

Introduction: The production of analogue delay lines for use in audio and video processing has presented a major problem for many years. The use of charge-coupled-device delay lines seems an attractive solution, since the c.c.d. is inherently an analogue device and therefore needs no analogue-digital conversion. However, the problem arises here of signal smearing due to charge-transfer inefficiency in a c.c.d. delay line having many delay stages. This results in loss of fidelity on high-frequency audio signals and smearing of sharp edges in video images. The solution to the transfer inefficiency problem is to use a serial-parallel-serial (s.p.s.) configuration of the c.c.d. delay line (cf. Fig. 1a) which is commonly used in the implementation of planar c.c.d. imaging devices. In this device n samples are clocked along a normal linear n-stage tapped delay line and then once in every n input sample periods the contents of this line are transferred into a set of n parallel delay lines, m stages long. The n signal samples are then passed along these parallel lines at a clock rate of f_c/n (f_c = original sampling rate). At the end of these lines the signal samples are transferred into successive stages of a second serial delay line of length n which clocks the signal samples out at the original sampling rate f_c. Thus every signal sample is subjected to a total of nm stages of delay (normalising $1/f_c$ to 1) but is only transferred $n + m$ times, resulting in a much reduced cumulative charge-transfer inefficiency. However, the use of this type of transfer geometry results in the enhancement of a second fundamental source of signal degradation in c.c.d.s, that is dark current.

Dark-current[2] noise arises from thermally generated minority carriers which are injected into the signal charge packet. At the output of the c.c.d. the contribution from dark current is indistinguishable from the actual signal charge.

In a linear c.c.d. this noise source does not usually present a serious problem provided that the device clocks operate at a constant rate. This is because each signal packet will accumulate the same amount of dark-current-generated charge during its passage along the delay line, resulting only in an extraneous d.c. bias at the output. This bias may vary depending on working temperature and power dissipation[3] but may be compensated for, in most cases, simply by a.c. coupling the device output. This, however, is not the case in an s.p.s. device, since a signal charge packet may be transferred along any one of n different delay paths. Differences in device structure such as local stacking faults and carrier concentration, which tend to vary across the device surface, mean that each delay path will have a different dark-current contribution. Therefore at the s.p.s. output the signal appears with an additive random noise component which repeats every n clock cycles; and this noise 'signature' will be different for every s.p.s. device.

It is the objective of this letter to present a scheme for the cancellation of this fixed noise which relies on the use of signal-processing techniques applied to the s.p.s. delay-line output rather than manipulation of the basic structure or manufacturing parameters.

Cancellation system: Of fundamental importance is the recognition that the noise pattern at the s.p.s device output is repetitive at every n clock cycles. Thus if a replica of the noise pattern could be stored permanently it would be possible to subtract it coherently from the s.p.s. signal output. To do this literally, however, would result in a complex module relying on fast digital circuitry not compatible in terms of manufacturing technology with the s.p.s. device. In addition, such a system would not compensate for variations due to long-term temperature drift.

It is important, if a device such as this is to be commercially viable, that the noise cancellation scheme should be physically compatible with the technology used in the s.p.s device manufacture and should be simple enough to enable integration on the same chip as the s.p.s. device.

The cancellation scheme chosen for the experimental system is shown in Fig. 1b. The principle of operation is that the output of the s.p.s. device is integrated by the linear (n-stage) c.c.d. delay-line loop so that a replica of the output noise pattern builds up in the linear c.c.d. This pattern is then subtracted from the s.p.s. output.

One defect which does occur using this cancellation scheme is that the integration loop imposes a comb-filter response on the output with nulls at frequencies which are integer multiples of f_c/n. However, for speech or video signals it will be assumed that truly periodic signals do not exist for long enough for a notch to form at the canceller output.

Results: The s.p.s. delay line used in these experiments was a 4 Kbit device arranged as a 32×128 stage block, clocked at 50 kHz. The delay line used to demonstrate the noise integration technique was a 32-stage c.c.d. tapped delay line; for the purposes of this experiment only the last tap was used.

The result shown in Fig. 2 is for a sinusoidal input to the s.p.s. device (Fig. 2a). The s.p.s. output is shown in Fig. 2b and the effect of dark-current fixed-pattern noise is clearly visible here.

The integrated noise pattern is shown in Fig. 2d and the actual canceller output is shown in Fig. 2c. From the frequency spectra shown in Fig. 2e and f the improvement in effective signal/noise ratio is approximately 20 dB.

A certain amount of signal distortion was obvious in the test signals used, mainly due to the relatively high values of the gain A necessary at the integrator input. This high gain was
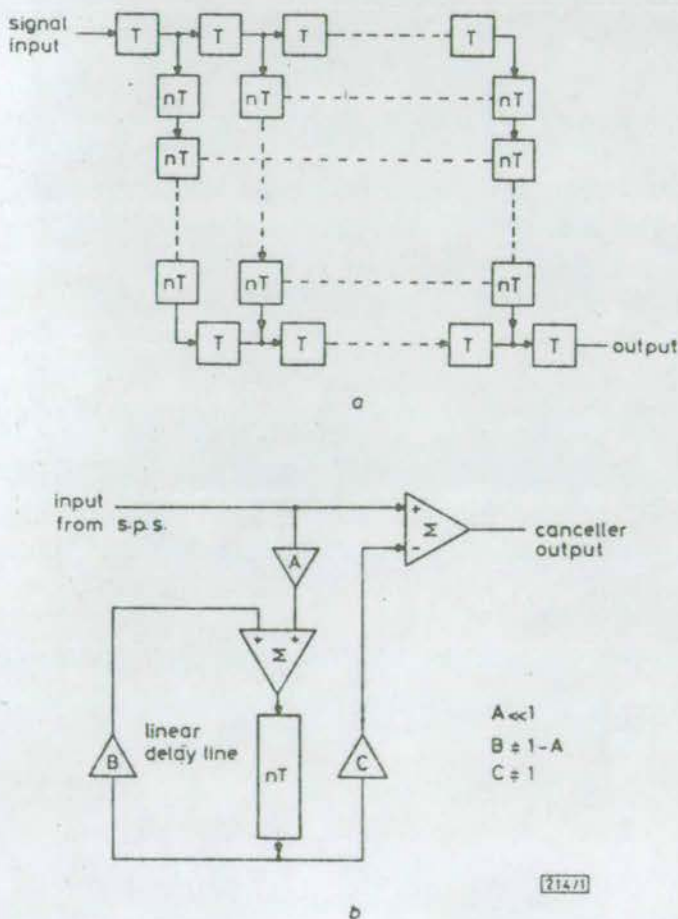

Fig. 1

a S.P.S. delay line functional block diagram
b Block diagram of the experimental s.p.s. noise cancellation scheme

necessary due to the cumulative transfer inefficiency in the noise integrator, meaning that integration had to take place over a relatively short period. This situation could be improved either by using a buried-channel c.c.d. in the integra-tor (the experimental module used a surface-channel device), or by using a simple transfer inefficiency compensation filter.

Conclusions: The feasibility of s.p.s. fixed-pattern noise cancel-lation has been demonstrated, showing that it is possible to considerably enhance the effective signal/noise ratio of a signal at the output of an analogue s.p.s. delay line. Potential future developments of this concept are:

(*a*) *Monolithic integration:* This circuit could be integrated monolithically with the s.p.s. device, since the technology for the cancellation scheme is compatible and all gains in the inte-gration loop are less than unity. Thus, simple m.o.s. buffers and charge-manipulation techniques may be employed to complete the cancellation loop.

(*b*) *Overall feedback:* The present cancellation system is an open-loop processor depending on externally applied gains. A desirable system would be a closed-loop system employing signal feedback to minimise dependence on loop gains.

(*c*) *Alternative cancellation schemes:* such as the use of delta modulation at the c.c.d. output to cancel fixed noise compon-ents with minimal dependence on the cancellation circuit defects, i.e. gain and transfer inefficiency.

The cancellation scheme described here results in an effective analogue delay line, for use in audio circuits or as a video frame store, which suffers from none of the problems of signal smearing associated with a long linear c.c.d. delay line or the fixed-pattern noise normally accompanying the s.p.s. realisation.

R. C. GEDDES   *19th June 1979*
C. F. N. COWAN
J. MAVOR

University of Edinburgh
School of Engineering Science
Mayfield Road, Edinburgh EH9 3JL, Scotland

J. F. DICKSON

Allen Clark Research Centre
Plessey (Research) Ltd.
Caswell, Northants., England

References

1 WEIMER, P. K., PIKE, W. S., SHALLCROSS, F. V., and KOVAC, M. G.: 'Video processing in charge-transfer image sensors by recycling signals through the sensor', *RCA Rev.*, 1974, **35**, pp. 341–354
2 TASCH, A. F., BRODERSON, R. W., BUSS, D. D., and BATE, R. T.: 'Dark current and storage-time considerations in charge-coupled devices'. Proceedings of the CCD applications conference, San Diego, 1973
3 MAVOR, J., and DENYER, P. B.: 'The dependence of CCD dark current upon power dissipation', *Microelectron. & Reliab.*, 1978, **17**, pp. 403–404
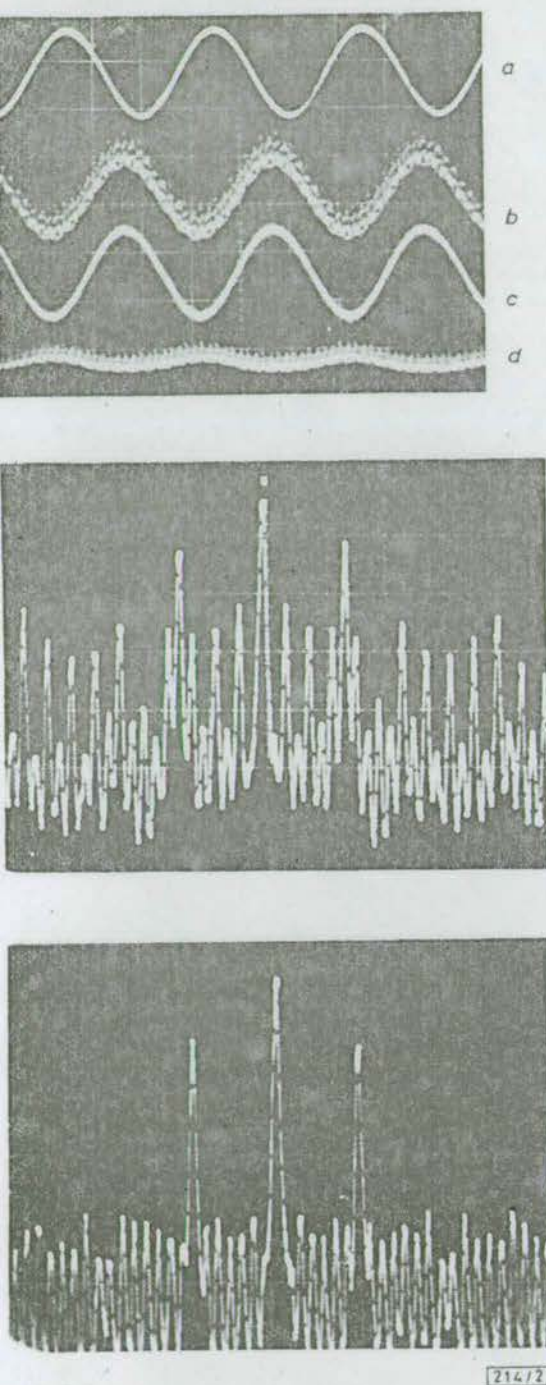
Fig. 2 *Experimental results for cancellation of fixed-pattern noise at the s.p.s. device output*

a Input signal
b S.P.S. output (unprocessed)
c Canceller output
d Integrated noise pattern
e Input-signal spectrum
f Canceller output spectrum (10 dB/div, 2 kHz/div)

IMPLEMENTATION OF A 64-POINT ADAPTIVE FILTER USING AN ANALOGUE C.C.D. PROGRAMMABLE FILTER

Indexing terms: Charge-coupled devices, Filters

This letter describes the construction and operation of an adaptive filter based on the Widrow least-mean-square adaption algorithm using a 64-point analogue programmable c.c.d. transversal filter as the main processing element. Initial test results are presented to confirm the principle of the system and these results may be readily compared with previously published simulation results.

Introduction: A useful device for many applications in signal processing would be a filter needing the minimum *a priori* information about an incoming signal immersed in noise to detect and reproduce it. The adaptive filter is probably the optimum form for this type of application and it has been extensively studied in theory and modelled on computers. However, little is known about the actual physical implementation of this class of filter. The purpose of this letter is to present an account of the design and operation of a compact 64-stage adaptive filter based on entirely analogue processing elements, the central element being an analogue programmable transversal filter based on a charge-coupled device (c.c.d.) tapped delay line.

Implementation: A block diagram of the basic adaptive filter element is shown in Fig. 1. The system is supplied with two inputs; the input signal $s(t)$ and the desired filter response to this input, $d(t)$. The aim is to force the output of the filter to resemble, as closely as possible, the desired response $d(t)$.
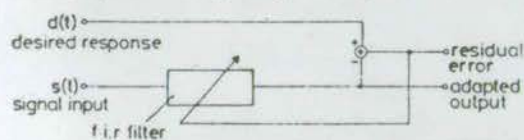

Fig. 1 *Basic block diagram of an adaptive filter*

To achieve this the filter weight vector H is updated continuously using the following algorithm:

$$h_k(t + 1) = h_k(t) + 2\mu\epsilon(t) s(t - k)$$

where $\epsilon(t) = d(t) - r(t)$ is the output error, $r(t)$ is the actual filter output and μ is a selected convergence factor. The subscript k refers to the weight or tap position in the filter, the terms in brackets are the time or delay indices and t is in normalised units of one delay time.

This is the Widrow least-mean-square adaption algorithm, and a detailed derivation and discussion may be found in Reference 1.
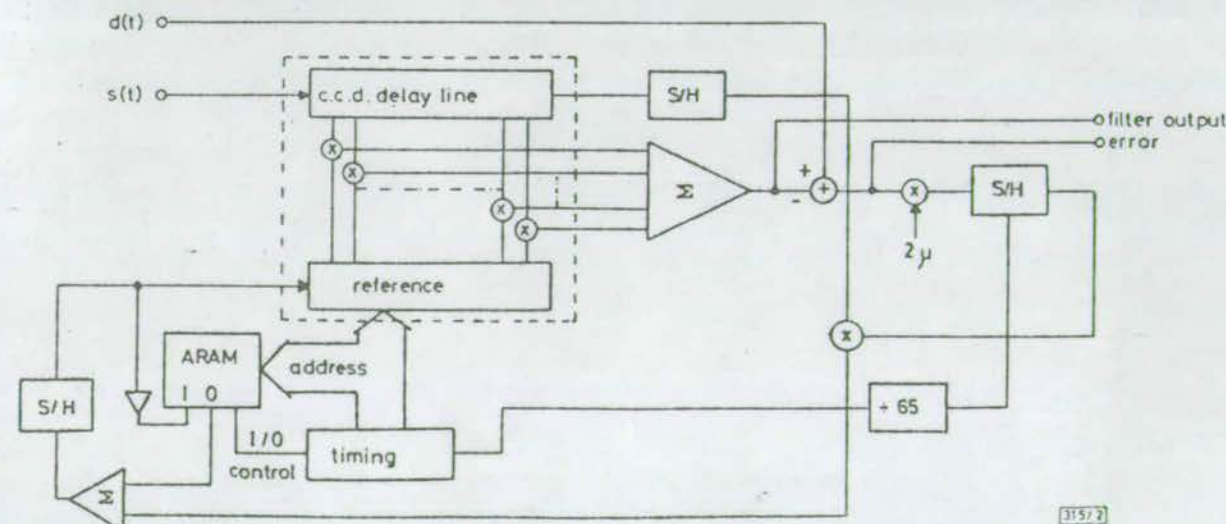
Fig. 2 shows a block diagram of the system used to implement this algorithm. The programmable transversal filter used (shown enclosed in the dotted box in Fig. 2) was a monolithic 64-stage device using a tapped c.c.d. to delay the input signal and a bank of analogue storage capacitors to hold the weight vector; the actual weighting was achieved by parallel multiplication using a single m.o.s.t. four-quadrant multiplier at each weighting point. The design and operation of this filter has been described in detail in recent literature.[2] Since with this filter it is impossible to update all 64 weights in the time between successive output data points, the output error $\epsilon(t)$ is sampled at only one point in 65 and the weights are updated *sequentially*. In addition, it is not possible to read individual weight values once they have been stored in the filter holding register. It was therefore necessary to provide a separate analogue memory (the ARAM in Fig. 2), which could be both read from and written to at any time, to provide temporary storage of weight values during the weight-vector updating cycle.

Results: All the experimental results given in this Section are for the case where the desired response input is fed with an in-phase uncorrupted version of the input signal. The simplest case is when both the input signal and the desired response are the same pure sinusoid (nonsynchronous with the c.c.d. clocks). For this case, the filter output was found to track the desired response perfectly in frequency and phase from the Nyquist frequency (for all results presented here the Nyquist frequency is 6·4 kHz) to about 200 Hz, which corresponds to the periodicity matching the filter length. At this low frequency, phase tracking is degraded since it is impossible to form an accurate matched response in the reference weights.

Fig. 3(i) shows the effect of using an input signal that is a pure sine wave at 1 kHz corrupted by a second sine of higher frequency (5·4 kHz). Fig. 3(i)*a* shows the input signal $s(t)$, Fig. 3(i)*c* shows the desired output $d(t)$ and Fig. 3(i)*b* shows the actual filter output. It can be seen that considerable rejection of the interfering signal takes place. Measurements on a spectrum analyser showed the rejection of the 5·4 kHz signal to be approximately 35 dB. Fig. 3(i)*d* shows the corresponding weight vector H that forms the expected sinusoid (appropriately adjusted in phase to maintain correct phase tracking at the output).

Generally it was found that the filter behaved well under conditions where the desired output was a near monochromatic signal. However, when a signal of wide bandwidth is used (e.g. a linear f.m. waveform), the required impulse response is forced to approximate more closely to a single impulse. The system then becomes more susceptible to noise since the lower amplitude tap weights still add the same degree of noise to the filter output as before.

Fig. 3(ii)*a* shows the spectrum of a monochromatic signal at 900 Hz contaminated by broadband noise that was applied to the filter input; the desired output is the uncorrupted 900 Hz

Fig. 2 *System block diagram*

signal. The filter output spectrum is shown in Fig. 3(ii)b. The minimum noise rejection (at frequencies close to the desired output frequency) was 20 dB with higher-frequency components being rejected by about 40 dB. The expected increase in noise rejection on both the high- and low-frequency sides of the desired frequency may be clearly seen.
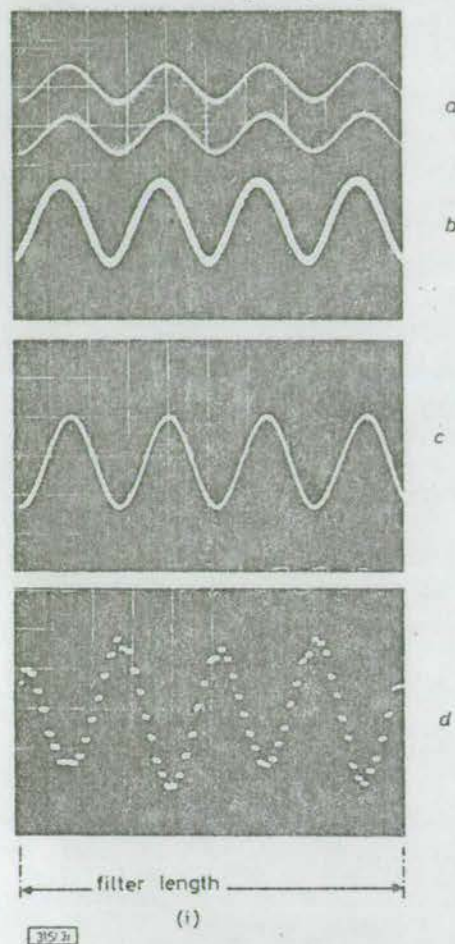

(i)

Fig. 3(i) *Adaptive filter performance for a sinusoidal input with high-frequency interference*

a Filter input, 5 V/div
b Filter output, 2 V/div
c Desired response, 2 V/div
d Weight vector, 2 V/div
Timebase: 0·5 ms/div

using analogue c.c.d. has been a two-stage device using a clipped form of the Widrow algorithm,[3] compared with the 64-point system using unclipped error samples reported here. The adaptive inverse filter described by Corl[4] used a computer to update the filter tap weights using a zero-forcing algorithm.

This type of system will have many possible applications in the field of baseband signal processing, such as e.c.g. noise cancelling, voice processing and sonar. Further work is presently being undertaken on the optimisation of this system (particularly in increasing its tolerance to broadband signals), and it is hoped that an integrated version may be designed and built in the near future.

Conclusions: It has been demonstrated that a 64-stage adaptive filter may be realised using analogue m.o.s. and charge-coupled-device technology resulting in a compact low-power unit. This is thought to be the first realistic demonstration of such a filtering system using this technology, which could be integrated monolithically. The only previous implementation
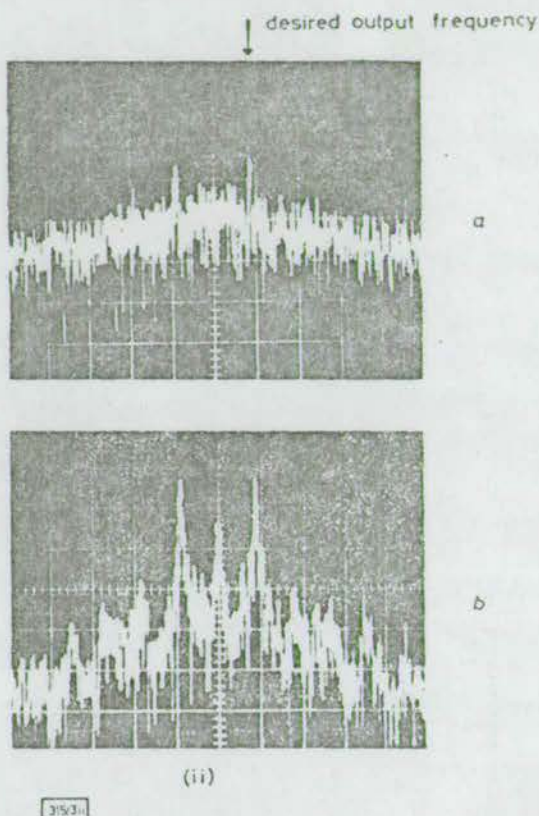
desired output frequency

(ii)

Fig. 3(ii) *Filter spectral characteristics for a monochromatic signal immersed in broadband noise*

a Signal spectrum at the filter input
b Filter output frequency spectrum
Centre frequency 0 Hz
Horizontal scale: 1 kHz/div
Vertical scale: 10 dB/div

C. F. N. COWAN   13th July 1978
J. MAVOR

Electrical Engineering Department
University of Edinburgh
Mayfield Road
Edinburgh EH9 3JL, Scotland

J. W. ARTHUR

Wolfson Microelectronics Liaison Unit
University of Edinburgh
Mayfield Road
Edinburgh EH9 3JL, Scotland

References
1 WIDROW, B., *et al.*: 'Adaptive noise cancelling: principles and applications', *Proc. IEEE*, 1975, 63, pp. 1692–1716
2 MAVOR, J., and DENYER, P. B.: 'Design and development of c.c.d. programmable transversal filters', *IEE J. Electron. Circ. & Syst.*, 1978, 2, (1), pp. 1–8
3 WHITE, M. H., *et al.*: 'Adaptive CCD signal processing for sonar applications'. Proceedings of the IEEE international symposium on circuits & systems, Phoenix, 1977
4 CORL, D.: 'A CTD adaptive inverse filter', *Electron. Lett.*, 1978, 14, pp. 60–62

AN EVALUATION OF ANALOGUE AND DIGITAL ADAPTIVE FILTER REALISATIONS

C.F.N. Cowan*, J. Mavor*, J.W. Arthur††, and P.B. Denyer†

* Department of Electrical Engineering, University of Edinburgh, U.K.
† Wolfson Microelectronics Institute, University of Edinburgh, U.K.
†† Wolfson Microelectronics Institute, University of Edinburgh, U.K., now with MESL, Edinburgh.

ABSTRACT

This paper presents an evaluation of analogue and digital circuit techniques when applied to the realisation of real-time miniature adaptive filters. Various adaptive algorithms are investigated with relevance to their suitability for integration using the two approaches, and three possible circuit architectures are evaluated to determine the optimum processor parameters in each case. Overall, a general view of optimum adaptive filter architecture in either analogue or digital circuit form is described and the relative merits of the alternative systems discussed.

INTRODUCTION

Adaptive processors have been recognised as potentially powerful signal processing modules for some time. An extensive investment, in terms of theoretical analysis and computer modelling (1-4) has already been made in the field of adaptive processors but the expense involved in their implementation has hitherto prohibited their general use in modern systems design. This is because it has been necessary to design systems with significant numbers of filter points, as complex digital modules which are bulky, expensive and dissipate high power levels. However, with the availability of digital LSI sub-systems, microprocessors and the entry of CCD analogue circuit elements (5) as viable signal processing components, the realisation of such systems appears much more feasible.

There are many potential applications for such an adaptive processor with conflicting needs in terms of bandwidth, overall storage time and dynamic range requirements. Also the desired flexibility of the processor may predominantly dictate which approach is ultimately more applicable. Some of the more important application areas (8) are telephone line equalisation, echo and multipath cancellers, noise cancellers in speech processing and medical electronics, and optimum filtering.

The objective of this paper is to examine the alternative techniques available for the construction of a commercially viable adaptive filter sub-system in a single monolithic device, concentrating on custom digital LSI circuitry and analogue CCD techniques.

The suitability of various forms of adaptive algorithms are investigated in relation to the conflicting needs of the two circuit approaches and various possible systems architectures to implement the selected algorithms are examined. The designs of two prototype systems, one analogue (6,7) and one digital, are described and experimental results are presented for the breadboarded prototypes to illustrate the constraints associated with each design. Finally, results are presented for the CCD system used as an adaptive equaliser (7) to compensate for distortion causing intersymbol interference in digital transmission over switched voice bandwidth telephone lines, such as those used in the public subscriber network.

SYSTEM IMPLEMENTATION

From our goal of providing a fully monolithic adaptive filter module it is clearly impossible to use high order adaption algorithms which commonly require the use of matrix manipulation techniques. We will therefore concentrate on variants of a simple, first order iterative algorithm where each tap weight is updated according to the following algorithm (8):

$$h(t + 1) = h(t) + 2\mu(inc) \qquad \ldots (1)$$

The increment (inc) in Eqn. 1 is commonly derived by using a function of the signal $s(t)$ and the error $e(t)$ = $d(t) - c(t)$ where $d(t)$ is a target signal and $c(t)$ is the filter output. Four forms of (inc) are commonly used and these are shown in Table I (9-12).

All four of these variants may potentially be integrated but only those which incorporate the sgn function are particularly suited to digital implementation where binary information is to be processed. However, the analogue counterpart is most suited to the linear LMS algorithm which may prove the most flexible since in its normal mode of operation it offers the fastest convergence, and the amplitude of the factor μ may be modulated by the signal power to remove the convergence time dependence of this algorithm on signal power levels. Fig. 1 shows single cells of the adaptive filter processor implementing the linear variant and the clipped variant; it illustrates that whereas the linear variant needs only a 4-quadrant multiplier to implement the (inc) function, the clipped version needs a shift register, an exclusive-or and a set of polarity switches. A 4-quadrant multiplier may be implemented using 4 MOS transistors, whereas the clipped version requires many more than this (certainly 11 for the exclusive-or alone). It was therefore decided that an analogue implementation should use the linear LMS adaption algorithm; thereby yielding the general form of the algorithm in a minimum amount of hardware. However, using digital circuitry, the linear variant would require the use of a second multiplier in the adaptive filter cell which is undesirable because the multiplier figures largely in speed, power and cost. The difference between the other three variants for a digital processor is insignificant and it was decided to use the clipped LMS variant since this algorithm offers the most reliable convergence properties.

Using either of these update schemes there are still three possible circuit architectures which may be used and these are summarised in Table II.

The optimum processor offering maximum bandwith, maximum packing density and maximum convergence rate, employs the parallel update scheme and, conveniently, using analogue CCD technology. This technique is prohibitively complex in digital circuitry but may be implemented in analogue form using cascaded sections such as that shown in Fig. 1(a). The ideal digital processor used the burst mode processing technique, (the so-called DELTIC filter (13)) where data is stored in RAM and a single, fast multiplier is time shared to provide the convolution sum and tap weight updates are calculated at the same time as accessing signal samples from RAM. This results in a loss of bandwidth proportional to the number of filter stages (i.e. to the resolution), but this is inherent in the design of

TABLE I : Basic characteristics of the LMS algorithm variants

DESCRIPTION	FORM OF (inc)	CONVERGENCE DEPENDENCE	CIRCUIT TECHNIQUE
LINEAR LMS	$e(t)\ s(t)$	POWER	ANALOGUE
CLIPPED LMS	$e(t)\ \text{sgn}\,[s(t)]$	AMPLITUDE	ANALOGUE/DIGITAL
HYBRID LMS	$\text{sgn}\,[e(t)]\ s(t)$	AMPLITUDE	ANALOGUE/DIGITAL
ZERO FORCING	$\text{sgn}\,[e(t)]\ \text{sgn}\,[s(t)]$	ZERO CROSSINGS	ANALOGUE/DIGITAL

TABLE II : Comparison of possible system archiectures using the Widrow LMS adaption algorithm.

ALGORITHM IMPLEMENTATION	CIRCUIT TECHNIQUE	FILTER POINTS PER DEVICE	BANDWIDTH	NORMALISED CONVERGENCE RATE
PARALLEL	ANALOGUE DIGITAL HYBRID	>100 NOT FEASIBLE ~32	HIGH	FAST
BURST	ANALOGUE DIGITAL HYBRID	>100	LOW	FAST
SERIAL	ANALOGUE DIGITAL HYBRID	>100	HIGH LOW HIGH	SLOW

DELTIC type filters and the incorporation of adaptability incurs no additional bandwidth penalty. The burst mode processor was the circuit architecture used in our implementation of the prototype, digital, adaptive filter: which was constructed as a 64-stage filter with a 10 kHz maximum sampling rate using the clipped LMS algorithm. A block diagram of the experimental system is shown in Fig. 2.

In the analogue prototype it was desired to use a monolithic 64-point CCD programmable transversal filter which, designed by WMI at Edinburgh, was already available (14). Therefore, in order to evaluate the possible performance of an all-analogue device with its associated filter errors, the third scheme using serial weight updating was used to yield maximum bandwidth operation. A block diagram of the prototype is shown in Fig. 3. In this processor, only one weight is up-dated at every output sample, meaning that only one integration circuit is used in a time shared manner (similar to the digital system) but the filter itself may operate at its maximum bandwidth: the only penalty being a reduction in convergence speed. The prototype system was constructed on a double Eurocard circuit board, consumed less than 10 watts of power and had an operational bandwidth of 50 kHz.

EXPERIMENTAL RESULTS

The results presented here for the two prototype modules are intended to provide an indication of the basic operational characteristics of the two systems under similar conditions. The sampling rate of the analogue system was limited, in these measurements, to 16 kHz so that its results may be reasonably compared with its digital counterpart.

Fig. 4 shows typical convergence paths taken by the two processors when the filter input is a sinusoid and the filter is trained initially to have a zero output and then trained to reproduce the input sinusoid at its output. For the digital system convergence occurs within 15 ms whereas the analogue system takes approximately 500 ms. This difference is due to the fact that the analogue system only updates its weight vector once in every 65 output periods and an additional amount of time is needed to reconstruct the correct d.c. bias points. Obviously, a full parallel update

analogue system would have much faster convergence rates.

Of greater significance than the convergence speeds is the ultimate accuracy which may be achieved in the converged state. In order to evaluate this effectively the external storage on the analogue adaptive filter is implemented digitally - as this would closely approximate the available decay times possible on a monolithic device (equating 16 bit quantisation noise to the expected signal decay noise). A very stringent test of the processor's ability to cope with a variety of signal conditions is to use high bandwidth input signals in a mode which forces the filter pulse response to assume minimum power content. One such input condition exists when both the system inputs are identical square waves. In this case the filter pulse response is found to be a set of pulses of alternating sign at the square wave edge positions, and the performance of the two proto-types under this condition is illustrated in Fig. 5.

The digital processor results showing the typical pulse response are given in Fig. 5(a). Here it can be seen that the actual filter output is of relatively poor quality (i.e. there is a high noise background) due to the fact that only the top eight bits of the filter accumulator were used to describe the filter output and to derive the error. Whereas this is perfectly satisfactory, when the system is trained as a matched filter it yields this poor result in any other operational mode. This is acceptable if the processor is designed to meet a known set of input signal characteristics but if these are not known a-priori then design is much more complex since more output bits must be accounted for, either by using different significant bit patterns or extending the resolution of the ADC associated with d(t). The same result for the analogue system, given in Fig. 5(b), shows that the filter output was considerably better and, in fact, there the continuity of the analogue circuitry lends itself to a more general application without the need for drastic recalibration for varying input signal conditions. This is due to the dynamic range of about 70 dB available at the PTF output which need not be transmitted to the digital integrator (i.e. only an 8-bit error increment is generated and only an 8-bit weight value is needed.)

These two results illustrate the main defferences between the two systems. The matter of convergence speed is not really at issue since the monolithic adaptive processor would in practice almost certainly be a parallel update system. The second result concerning output dynamic range would be a serious drawback for any digital system used in an unknown or rapidly varying signal environment.

The final result presented in Fig 6 demonstrates the operation of the analogue module in a practical signal processing environment. The incoming signal is the output of a telephone line (a simulation of 5 miles of telephone cable) carrying a basedband digital pulse train. It is desired to compensate for the signal distortion introduced by the line, therefore reconstituting the original data. One technique used to achieve this is to transmit a known data sequence for a short time during which the adaptive filter is trained with the known sequence. After training, the receiver-generated training signal is replaced by a signal derived from a decision circuit at the filter's own output which allows the filter to track minor variations in transmission line characteristics. The result, shown in Fig 6, is for the part of the cycle where the filter is being specifically trained so that it may approximate a suitable equaliser impulse response. The eye pattern of the filter input and output (Fig 6(e) and (f)) show that a considerable improvement in signal detectability is achieved.

CONCLUSIONS

The relative advantages of analogue and digital design have been evaluated with respect to the design of a monolithic adaptive filter. It has been shown that certain advantages such as higher packing density and improved bandwidth may accrue from the use of CCD-based analogue transversal filters in this system structure.

As part of an ongoing research project, these results cannot be taken as conclusive as the ultimate feasibility of analogue weight integration has not yet been sufficiently investigated. However, it is apparent that the iterative nature of the LMS algorithm can impart considerable improvement to the stability and error characteristics of the analogue CCD filter, but it cannot be determined just how much this stability may be attributed to the partially digital nature of the feedback loop and it may be desirable to retain this digital integration in the final design. However, work is presently under way to evaluate the true performance characteristics of the analogue weight update circuit described in Fig 1(a) and theoretical predictions show that achievable analogue storage times on a monolithic device can in fact equal the performance of 16-bit digital storage (roughly equating decay noise to quantisation noise). However, overall noise and drift characteristics may not be accurately evaluated at present. Linearity is not an issue provided continuity can be maintained over a reliable 60 or 70 dB dynamic range.

Sufficient information is available to predict that the monolithic integration of these sub-systems is possible with considerable numbers (>100) of filter stages on a single chip.

Fig 7 is a diagram which illustrates the applicability of various circuit techniques for different bandwidths and numbers of filter points. It can be seen from this that CCD occupies a very important area between bandwidth of about 1 kHz to about 6 MHz. Above 6 MHz the technology used must be surface acoustic wave, or alternatively, very short digital filters. Below 1kHz bandwidth, for any significant length of filter, dark current begins to have a severly degrading effect on CCD based filters and digital techniques must be used here or hybrids using discrete analogue circuitry with digital updating circuits.

Potential application areas such as telecommunications, medical electronics and speech bandwidth processing ensure that adaptive processors will have a firm future in modern signal processing systems design. With the development of monolithic fast ADC's and future VLSI technology, digital implementations will rapidly close the gap between low-frequency baseband and IF bandwidth processors. However, until then, the only viable technology which appears to be available to implement adaptive filters suitable for speech and sonar bandwidth processing and for use in high speed data modems in compact, low-power form is CCD-based integrated circuits.

REFERENCES

(1) Widrow, B. et al.: "Stationary and non-stationary learning characteristics of the LMS adaptive filter", Proc. IEEE, 1976, 64, pp. 1151-1162.

(2) Widrow, B. & McCool, J.M.: "A comparison of adaptive algorithms based on the methods of steepest descent and random search", IEEE Trans. AP, 1976, 24, pp. 615-637.

(3) Ahmed, N.: "A study of adaptive digital filters", Sandia Labs. Report, SAND 77-0102.

(4) Shensa, M.: "Time constraints and learning curves of LMS adaptive filters", Naval Ocean Systems Centre, San Diego, 1978, Technical Report No. 312.

(5) Boyle, W.S. & Smith, G.E.: "Charge-coupled semiconductor devices", BSTJ, 1970, 49, pp. 587-593.

(6) Cowan, C.F.N., Mavor, J. & Arthur, J.W.: "Noise cancellation and inverse filtering using a compact high performance CCD adaptive filter", Electronics Letters, 1979, 15, pp. 35-37.

(7) Cowan, C.F.N., Arthur, J.W., Mavor, J. & Denyer, P.B.: "CCD based analogue adaptive processing", Proc. 5th Int. Conf. on "The Technology and Application of Charge-Coupled Devices", Edinburgh, Sept. 1979.

(8) Widrow, B. et al.: "Adaptive noise cancelling: principles and applications", Proc. IEEE, 1975, 63, pp. 1692-1716.

(9) White, M.H. et al,: "Charge-coupled device (CCD) adaptive discrete analogue signal processing", IEEE J. of Solid-State Circuits, 1979, 14, pp. 132-147.

(10) Lucky, R.W.: "Techniques for adaptive equalisation of digital communications systems", BSTJ, 1966, pp. 255-284.

(11) Sondhi, M.N. & Debasis, M.: "New results on the performance of a well-known class of adaptive filters", Proc. IEEE, 1976, 64, pp. 1583-1597.

(12) Moscher, J.L.: "Adaptive filter with clipped input data", Stanford University, Technical Report No. 6796-1.

(13) Mavor, J. et al.: "Design and performance of a programmable, real-time charge-coupled device recirculating daly-line correlator", IEE J. Electron. Circ. & Syst., 1977, 1, (4), pp. 137-143.

(14) Denyer, P.B., Mavor, J. & Arthur, J.W.: "Miniature programmable transversal filter using CCD/MOS techology", Proc. IEEE, 1979, 67, pp. 42-50.
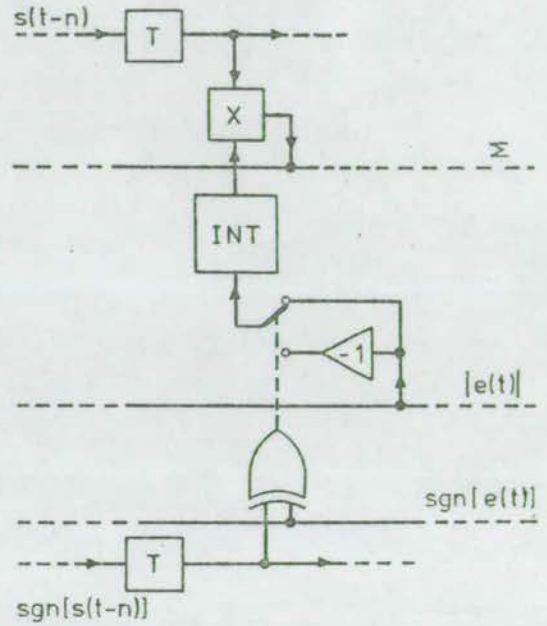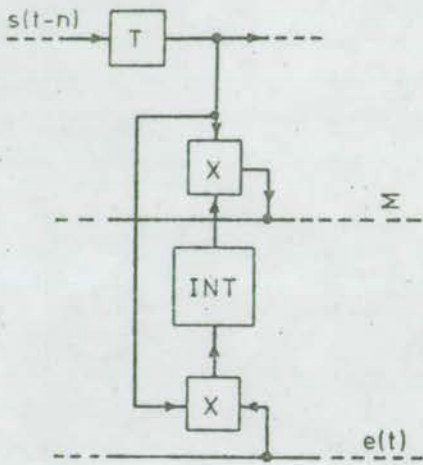
Fig 1 (a) Parallel update filter stage for the
linear LMS algorithm.

(b) Parallel update filter stage for the clipped
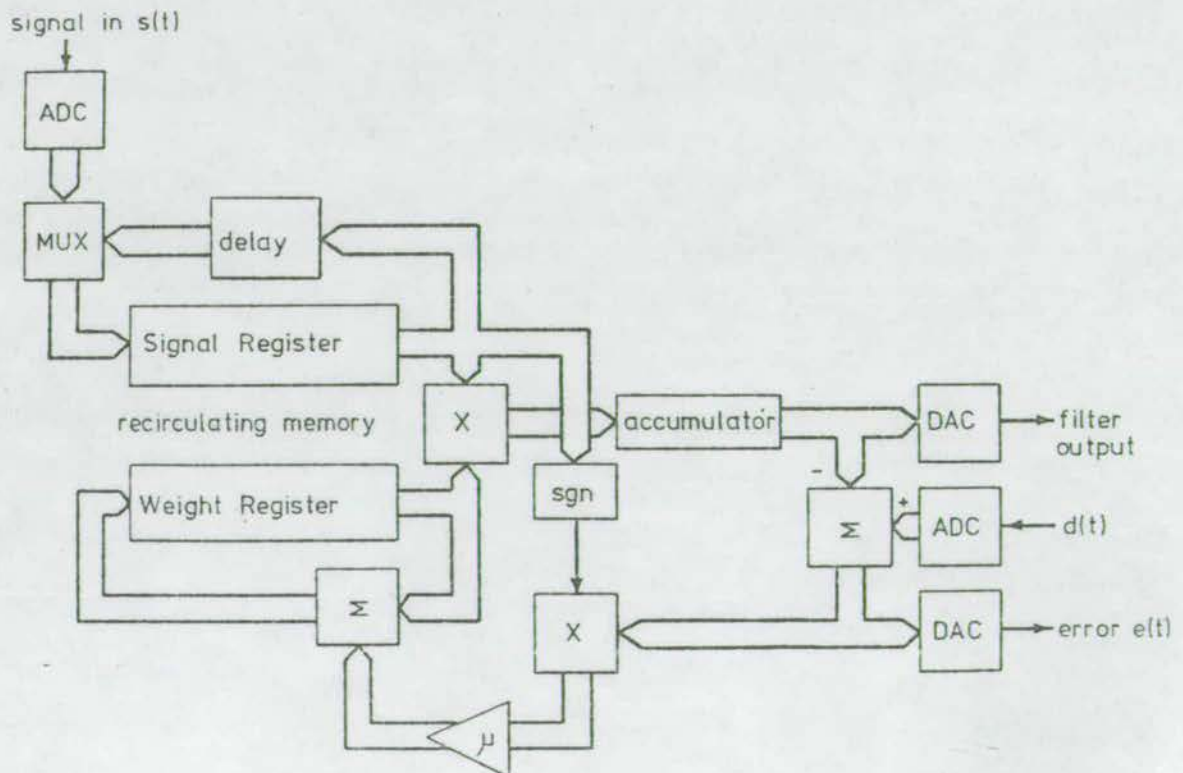LMS algorithm.

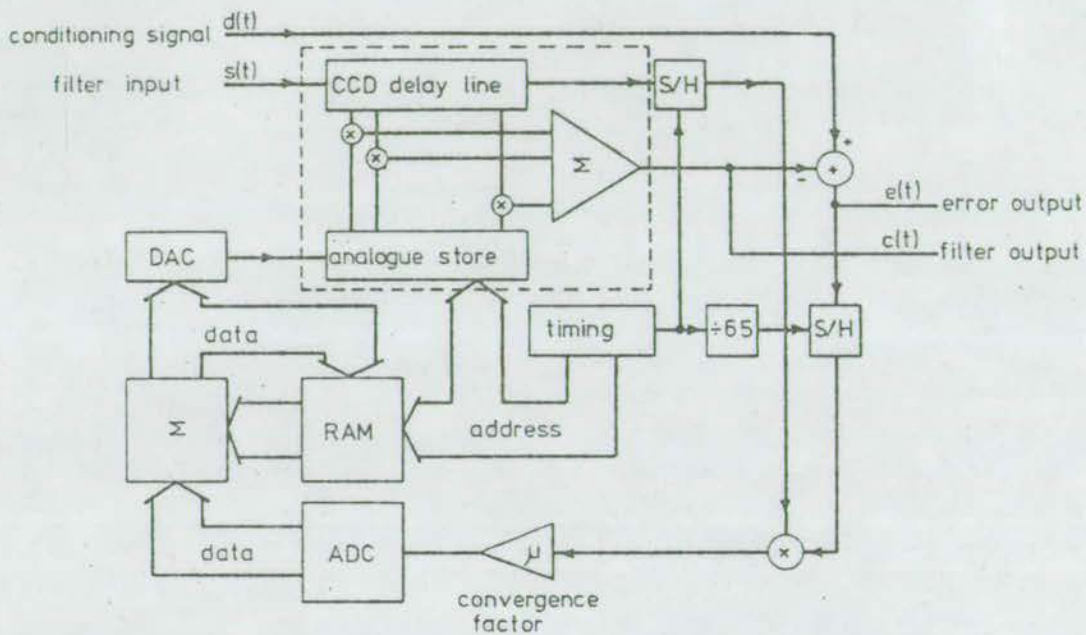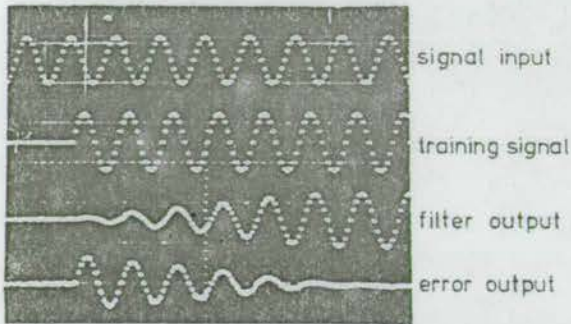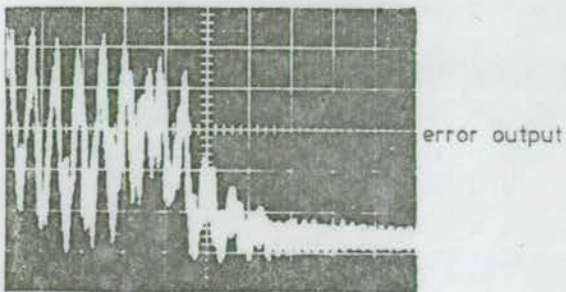Fig. 2 . Digital adaptive filter system based on the DELTIC processor.

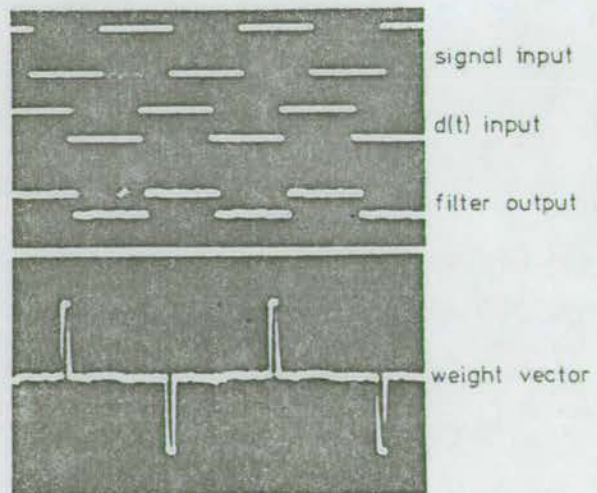Fig. 3 : Block diagram of the analogue system using serial update.

(a) convergence of the digital adaptive filter.
timebase – 2 ms/div

error output
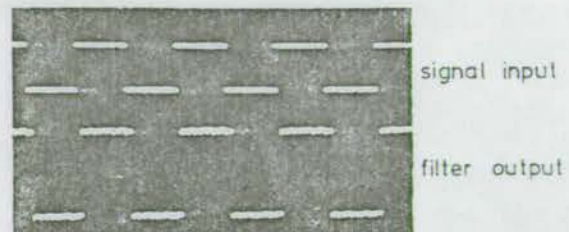
(b) analogue system convergence.
timebase –100 ms/div

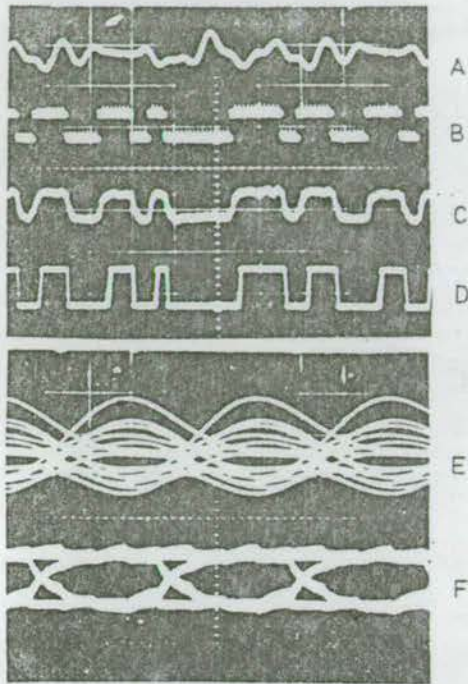Fig. 4 : Convergence characteristics of the two prototype systems.

signal input

d(t) input

filter output

weight vector

(a) result from the digital filter.

signal input

filter output

(b) result from the analogue filter.

Fig. 5 : Results from the two prototype filters for minimum power pulse response training.

A simulated line output.
B training signal.
C filter output.
D threshold detect output.
E line output eye pattern.
F filter output eye pattern.
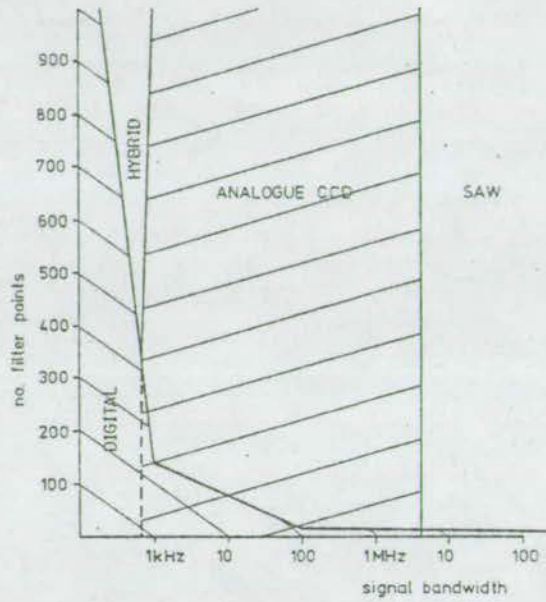
Fig. 6 : Adaptive line equalisation.

Fig. 7 : Graph illustrating the applicability of various circuit techniques to compact, low power adaptive filter implementation.

MINIATURE CCD-BASED ANALOG ADAPTIVE FILTERS

C.F.N. Cowan*, J. Mavor*

*Department of Electrical Engineering, University of Edinburgh,
The King's Buildings, Mayfield Road, Edinburgh, EH9 3JL.

Abstract

The paper describes the design and operation
of a 256-point adaptive filter, based on a
monolithic 256-point charge-coupled device
programmable transversal filter. Results
are presented showing the characteristic
performance obtained with the prototype
system.

1. Introduction

Adaptive filter design has mainly centred
around the use of digital circuit elements[1-3]
due to the inherently complex nature of adaptive
algorithms. In general, they require that the
transversal filter tap weights be calculated to
a high precision. However, at high time-
bandwidth products it becomes difficult to
realise digital adaptive filters having
appreciable bandwidth in a compact, low-
power form due to the complexity of the
digital multiplier[4]. It is in this area that
the fast parallel processing ability of the
charge-coupled device[5] (CCD) programmable trans-
versal filter[6] (PTF) may be best utilised. In
this paper the design and characteristic per-
formance of a 256-point analog adaptive filter
based on a monolithic 256-point CCD PTF[7] is
described. The adaptive weight updating
algorithm is implemented using digital cir-
cuitry but since the parallel processing
associated with the filtering operation
is carried out at high speed by the analog
CCD element there is no requirement for the
limiting digital elements[8] such as fast parallel
multipliers and fast analogue to digital con-
verters (ADC's).

The use of a simple iterative adaption
algorithm (the clipped Widrow least-mean-
square (LMS) algorithm[9,10] in this case) has
the additional advantage of ensuring stable
operation of the CCD element, since any drift
in d.c. bias points is automatically compen-
sated for by the adaptive loop.

Such a processor has many potential appli-
cations in high bandwidth systems, particularly
in the areas of equalisation and echo-cancell-
ation in communications channels. In many
areas, such as echo cancellation for satellite
communications and equalisation of long-distance
telephone lines, size and power consumption are
not serious limitations, but for many applica-
tions such as echo cancellation in loud-speak-
ing telephones and digital modems power con-
sumption plays a major role in determining
the type of circuit technique which is used to
implement the adaptive filter element.

2. CCD Adaptive filter implementation

A general block diagram of the experimental
system is shown in Figure 1. The system is
based around a monolithic, all-analog, 256-
point CCD programmable transversal filter[7].
This filter implements a transversal filter
literally (as shown inside the dotted box in
Figure 1) where the signal is delayed by a CCD
analog delay line which has 256 non-destructive
tapping points each separated by a discrete
time delay f_c^{-1} (where f_c is the sampling
frequency). The filter weighting[6] is achieved
by multiplying each tap output by a weight
value which is stored on a simple, capacitive,
sample-and-hold site at each tapping point on
the circuit. Multiplication of each signal
and weight value is achieved by a single MOST
at each tapping point which has a current out-
put allowing summing simply by commoning all
the multiplier outputs to a single current-
summing bus to yield the output:

$$y(t) = \sum_{n=0}^{255} s(t-n) \, h_n(t) \qquad (1)$$

where $h_n(t)$ is the tap weight value at time t on
tap n; and $s(t-n)$ is the signal value at time
t on tap n.

The adaption algorithm used in this imple-
mentation was the clipped Widrow LMS algorithm[10]
given by:

$$\underline{H}(t+1) = \underline{H}(t) + 2\mu \, . \mathrm{sgn}[\,\underline{S}(t)]\, .e(t) \quad (2)$$

where $\underline{H}(t)$ is the vector of weight values at
time t; $\mathrm{sgn}[\,\underline{S}(t)]$ is the vector of signal
signs at time t; $e(t) = d(t) - y(t)$;
$d(t)$ is a secondary 'conditioning' signal
input; and μ is a convergence factor controll-
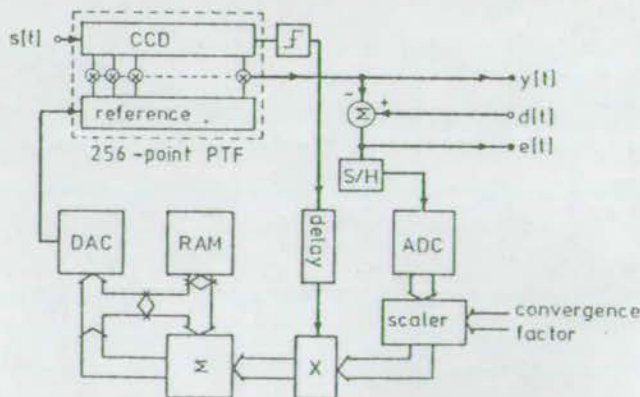ing the rate of convergence and the accuracy

Figure 1 : Block diagram of the experimental
256-point adaptive filter system.

and stability of the algorithm.

Owing to the construction of the PTF (which was not designed as a dedicated adaptive filter), it was necessary to update weight values in a *serial* fashion. The algorithm stated in equation (2) inherently implies that every weight is updated for each input sample period. However to do this with the PTF described above would result in a severe loss of bandwidth. It was, therefore, decided in this prototpye module to retain the filter bandwidth and sacrifice convergence time by updating only one weight value at each input sample period. This is done by sampling the e(t) output at one point in 256 and then multiplying it by the sign of the signal on the last tap of the CCD. This increment is then used to update the tap weights, proceeding from right to left, in sequence, at the sampling rate. If this update period were maintained at a period of 256 f_c^{-1} there would be a problem with adaption to periodic waveforms having a period equal to any harmonic of the update period. To solve this problem the update period is 'jittered' between 257 f_c^{-1} and 258 f_c^{-1}.

A previous deïsgn based on a 64-point [8] PTF was limited in bandwidth by the convergence time of the ADC. This problem was solved here by inserting a 100 point digital delay in the sgn[s(t-n)] path before multiplication by the error. This means that the error may be sampled 100 f_c^{-1} secs. early, leaving that amount of time for ADC settling before the result is digitally latched and used to multiply

by the signal sign bits. Therefore, at a 1 MHz sampling rate 100 μsec is available for ADC settling meaning that a low power, mono-lithic voice bandwidth ADC could be used. The multiplication sgn[s(t-n)].e(t) is achieved simply by using exclusive-OR gates and the convergence factor μ is implemented by a digital scaling circuit which shifts the position of the error bits over 8 discrete positions corresponding to 8 convergence factors from 2^{-5} to 2^{-12}. The result 2μ. sgn[s(t-n)].e(t) is added to the digitally stored weight value $h_n(t)$, converted to analog and stored in the PTF weight store.

The system was constructed on 3 international-size circuit cards, consumed 7.5 W and had a bandwidth of 200 kHz (which should be extendable to 0.5 MHz).

3. Experimental results

In this section experimental results from the prototype system, described above, are presented which are compared where possible with theoretically predicted values to demonstrate the predictability of the performance of this predominantly analog processor.

The result shown in Figure 2 is for the adaptive filter operated in a noise-cancellation mode, where the 'noise' in this case is represented as a sinusoid of frequency 2f and the required signal is a sinusoid of frequency f. The d(t) input to the filter is the incoming signal comprising both sinusoids at
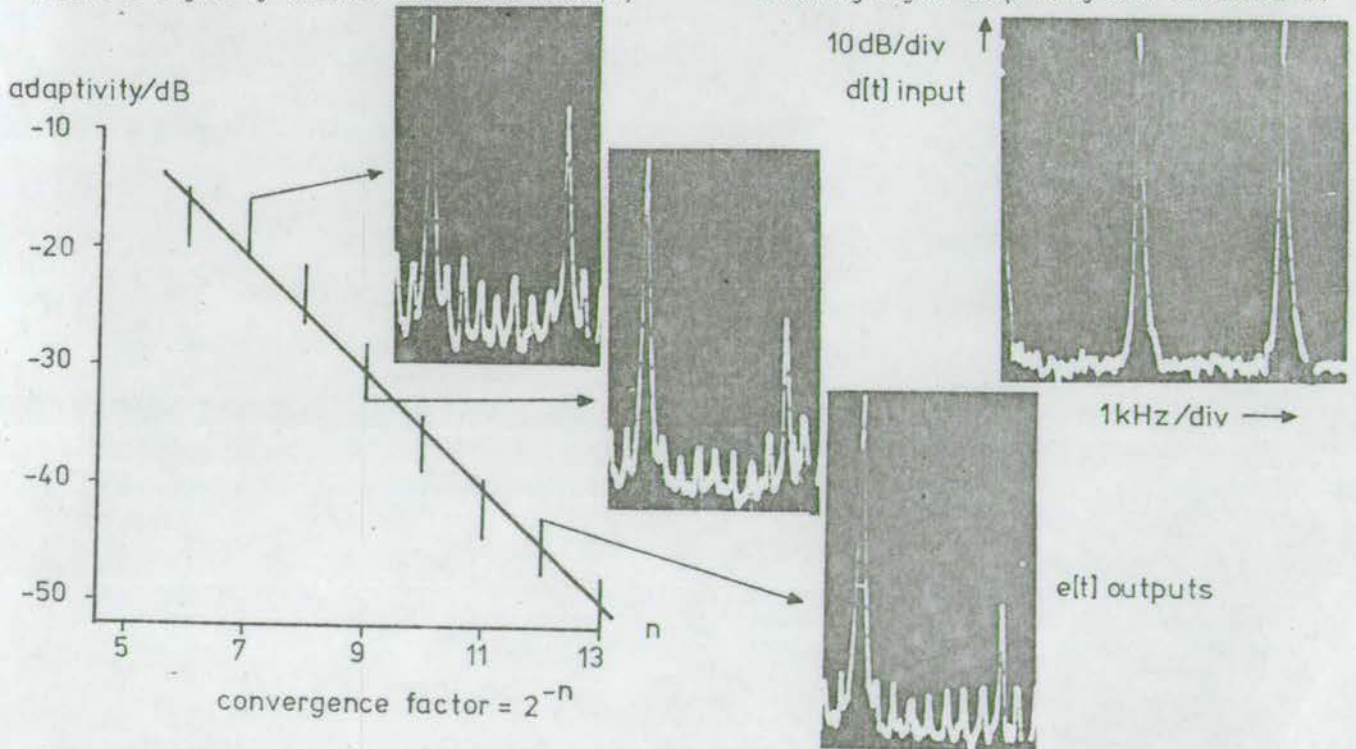

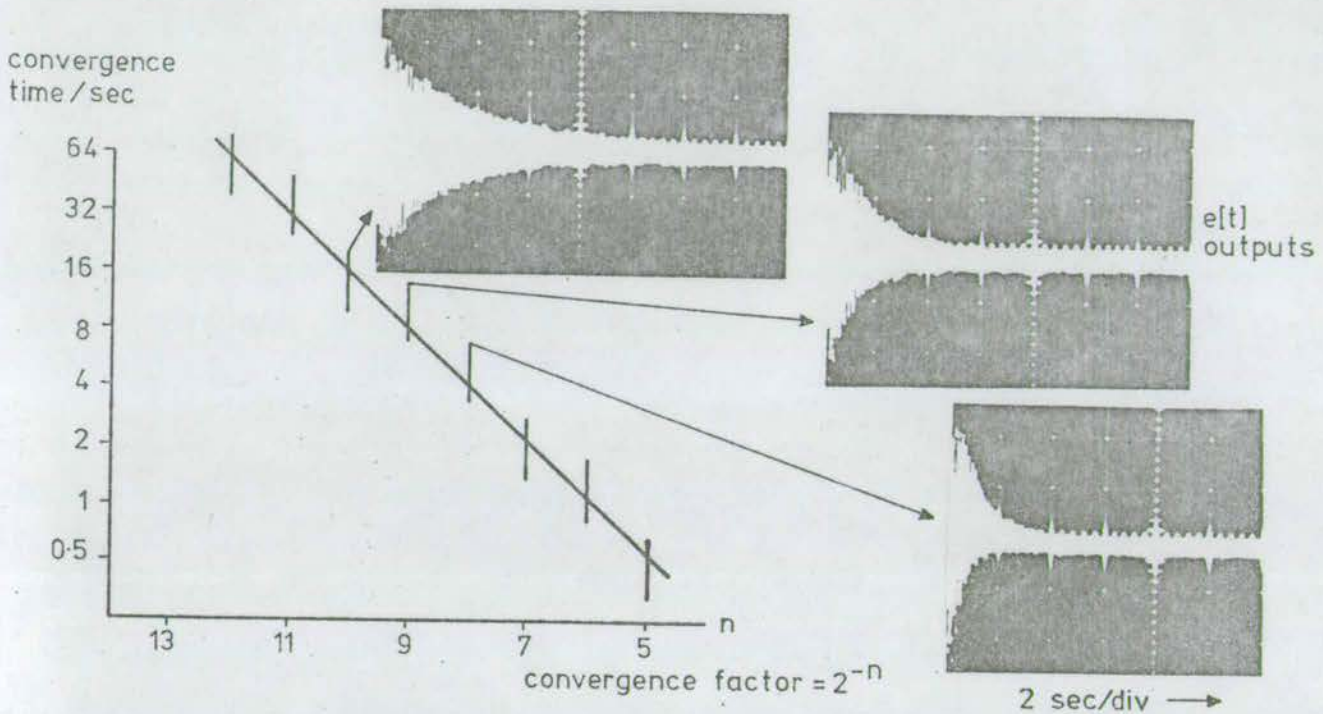Figure 2 : Experimental results showing the variation in adaptivity with the convergence factor μ.

Figure 3 : Experimental results showing the variation in convergence time with convergence factor μ.

the same level (this is represented spectrally in the inset photograph) and the s(t) input is a version of the interfering sinusoid which is of a different _phase_ and amplitude. In order to minimise $e^2(t)$ the filter must form an impulse response which adjusts the phase and amplitude of s(t) in such a way as to subtract coherently from d(t) leaving the required sinusoid of frequency f at the e(t) output. The degree of cancellation may be defined as a general figure of merit called the _adaptivity_ χ where:

$$\chi = 20 \log_{10} \frac{e'(t)}{d'(t)}$$

where the superscripted dash implies the spectral component of e(t) and d(t) which is actually cancelled.

Theoretical predictions indicate that due to the misadjustment effect the adaptivity should be best for small μ. The experimentally derived graph shown in Figure 2 confirms that this is the case with the adaptivity varying between -20 dB and -50 dB for a variation in μ from 2^{-5} to 2^{-12}. Misadjustment in this case may heuristically be seen to be due to the interference of the sinusoid on e(t) with the weight vector update, and as the size of μ is decreased this has less effect on the weight values. Typical experimental results showing spectra of the e(t) output are shown for selected points on the graph.

Although lower convergence factors result in enhanced adaptivity performance there is a trace-off in that reducing μ causes a corresponding increase in convergence time. Theoretically the convergence time constant

for an ideal, parallel-update filter is given by[11]:

$$\tau = \frac{n}{4\mu \, trR} \qquad (3)$$

where n is the total number of filter points; R is the autocorrelation matrix of the input signal $E\{SS^T\}$, and trR is the trace of R. The total convergence time for an ideal parallel-update filter is approximately $4\tau f_c^{-1}$. For our serial update prototype the convergence time should be approximately 256 times this value. Therefore for adaption to a situation where both inputs are identical sinusoids the convergence time for $\mu = 2^{-5}$ should be about 0.5 secs. with a sampling frequency of 50 kHz. The convergence time should then double for each successive halving of μ. Figure 3 shows a set of results which illustrate the behaviour of the e(t) output during convergence when μ is varied between 2^{-5} and 2^{-12}. The input conditions in this case are that the s(t) input is always a sinusoid of frequency f, the filter is first trained to have an output of zero, d(t) then becomes a replica of s(t) and the filter is allowed to converge to this new training signal. As expected the convergence times shown in Figure 3 roughly double for each halving of μ.

The result shown in Figure 4 demonstrates the filter operation in a slightly more complex mode. In Figure 4 the input signal (supplied to both s(t) and d(t)) is made up of a pair of sinusoids at frequencies f and 3f plus a certain amount of random white noise. The noise component can not be correlated by the filter and the optimum result is therefore the case where the two periodic components are reproduced
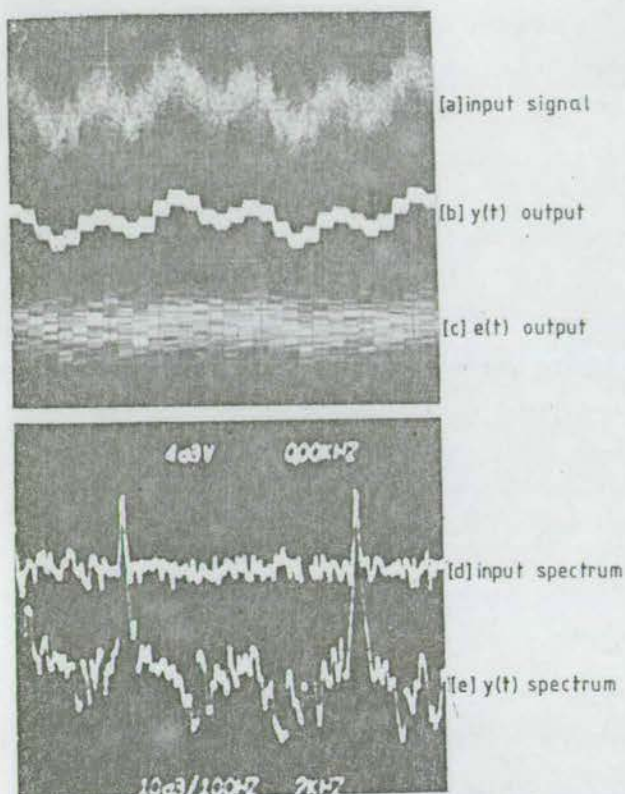
[a] input signal

[b] y(t) output

[c] e(t) output

[d] input spectrum

[e] y(t) spectrum

Figure 4 : Operation of the prototype filter as a self-tuning filter.

at the y(t) output with the noise suppressed (see Figure 4(b)) and with the correct phases and amplitudes to subtract coherently from d(t) leaving the broadband component on the e(t) output (see Figure 4(c)). Spectra of d(t) and y(t) are shown in Figure 4(d) and (e) showing that the noise component on the y(t) output is rejected on average by about 25 dB. This type of configuration has possible application for the cancellation of hum or background music on speech signals.

4. Conclusions

The feasibility of constructing high time-bandwidth product adaptive transversal filters using analog CCD/MOS circuit techniques has been demonstrated. It has been shown that considerable advantage in terms of circuit complexity, power consumption and operating bandwidth may be achieved in comparison with equivalent digital systems. This means that adaptive filters of this type may be applied to echo cancellation and equalisation tasks in mobile communications equipment where physical size and power consumption are at a premium.

The main problem with present prototypes is the slow convergence time available due to the serial mode of tap update. In many applications this is acceptable but it would be possible to achieve higher convergence times by using a a burst-mode updating scheme which serially updates all weights within one sample period,

though this naturally incurs a great loss of bandwidth. The alternative is to design an all-analog adaptive filter cell which allows full parallel updating without the requirement for seriallising the weight vector. This work, however, demonstrates the basic feasibility of using analog CCD elements in the adaptive filter without incurring any serious penalty because of the inherent inaccuracies of the analog components. Work is presently being undertaken at Edinburgh University to develop a fully monolithic, all-analog CCD adaptive filter having 64-filter points on a single chip[8].

References

1. Hinden, H.J.; "Echo-cancelling chip opens way to increased use of satellite channels", Electronics, Elec.Review, Aug.2, 1979, pp.41-42.
2. Ahmed, N.; "A study of adaptive digital filters", Sandia Labs. Report, SAND 77-0102.
3. Lucky, R.W.; "Techniques for adaptive equalisation of digital communications systems", BSTJ, 1966, pp.255-284.
4. South, C.R., Hoppitt, C.E., and Lewis, A.V.; "Adaptive filters to improve loudspeaker telephone", Electron Lett., 1979, 15, pp.673-674.
5. Boyle, W.S., and Smith, G.E.; "Charge coupled semiconductor devices", BSTJ, 49, 1970, pp.587-593.
6. Denyer, P.B., Mavor, J., and Arthur J.W.; "Miniature programmable transversal filter using CCD/MOS technology", Proc. IEEE, 1979. 67, pp.42-50.
7. Denyer, P.B. and Mavor, J.; "256-point programmable transversal filter", Proc. 5th Int. Conf. on CCD's, Edinburgh, 1979, pp.253-254.
8. Cowan, C.F.N., Mavor, J., Arthur, J.W., and Denyer, P.B.; "An evaluation of analogue and digital adaptive filter realisations", IEE, Int. Spec. Sem. on 'Case studies on advanced signal processing", Peebles, Scotland, Sep., 1979.
9. Moscher, J.L.; "Adaptive filter with clipped input data", Stanford University, Tech. Report no. 6796-1.
10. Widrow, B., et al; "Adaptive noise cancellating : principles and applications", Proc. IEEE, 1975, 63, pp.1692-1716.
11. Widrow, B., et al; "Stationary and non-stationary learning characteristics of the LMS adaptive filter", Proc. IEEE, 1976, 64, pp.1151-1162.

ANALOGUE SIGNAL PROCESSING USING CHARGE-COUPLED DEVICES

C.F.N. Cowan and J. Mavor

Department of Electrical Engineering, University of Edinburgh, Edinburgh, Scotland

In this paper the application to signal processing of a monolithic programmable transversal filter (PTF), using a charge-coupled device (CCD) as the basic delay element, is considered. In particular the relevance of the highly flexible programmable nature of the device in conjunction with its high speed of operation makes it generally applicable to a number of key applications. The applications described to illustrate these attributes are matched filtering, a prime transform implementation and adaptive filtering.

INTRODUCTION

A function which is common to many signal processing tasks is that of *convolution*, which implies the calculation of a scalar product of a column vector and a row vector of the same order where at least one of the vectors is not normally known *a priori*. The major problem involved in performing this operation using digital circuit techniques is the number of multiplications needed to form a single convolution result given by:

$$Y_{out} = [S_1 \ S_2 \ S_3 \ \dots \ S_N] \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ \vdots \\ h_n \end{bmatrix} = \sum_{n=1}^{N} s_n h_n \quad (1)$$

The classical digital solution to this problem is to use the so-called DELTIC [1] circuit architecture which is illustrated in figure 1. In this approach the two vectors S and H are stored in a recirculating digital memory and each individual partial product, $s_n h_n$, is formed sequentially using a single, hardware, parallel multiplier. The output of this multiplier is then accumulated over N points (where N is the order of the filter) to form the convolution result Y_{out}. This approach has two major disadvantages:

(1) It is necessary to use a hardware parallel multiplier which uses a large amount of power if fast operation is required.

(2) The bandwidth of the system is inversely proportional to the time-band-width product due to the serial nature of the calculation, and it therefore becomes increasingly difficult to perform real-time operations as the filter order increases.
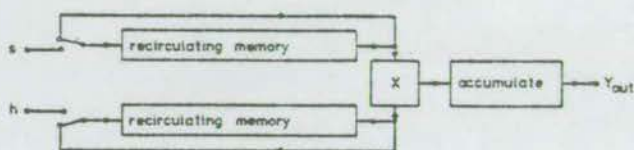
The approach does however have the advantage

Figure 1 : Conventional DELTIC filter structure.

that both the S and H vectors may be easily altered or reconfigured which is important for the convolution module to have as wide a range of application as possible.

A second digital circuit architecture which may be used in preference to the DELTIC system is the 'bit slice' architecture proposed by Peled and Liu [2]. In order to evaluate the way in which this technique operates we must first examine the mechanism of conventional multiplication. The multiplier s_n may be viewed as:

$$s_n = \sum_{i=0}^{b=1} s_{ni} \ 2^i \quad (2)$$

where b = the number of bits forming the s_n or h_n word.

S_{ni} = the value of the $(i+1)^{th}$ bit of the s_n word.

The result of the multiplication $s_n h_n$ for a conventional multiplication is then given by:

$$s_n h_n = \sum_{i=0}^{b-1} [(S_{ni}.2^i).h_n] \quad (3)$$

and the convolution sum is then:

$$Y_{out} = \sum_{n=1}^{N} s_n h_n = \sum_{n=1}^{N} \sum_{i=0}^{b-1} [S_{ni}.2^i .h_n] \quad (4)$$

The 'bit slice' approach to convolution

operates by simply reversing the order of
the summations in equation (4) so that the
sum over n is performed first with i held
constant, and then the sum over i is calculated
to form the eventual output.

In hardware terms this is done by
storing the sum of the partial products
(sum over n) in a ROM which is addressed by
the set of ith bits of $\underline{S}$. This means that
all possible outputs for any combination of
bit patterns must be stored in ROM meaning
that the vector $\underline{H}$ is permanently stored and
can not be easily changed. A schematic
diagram illustrating the operation of this
system is shown in figure 2. This system
has the advantage that no hardware multiplier
is required and the speed of operation is
independent of time-bandwidth product.
Increasing the number of filter points used
obviously causes a steady increase in the
amount of memory required, but the relative
cheapness of memory as opposed to LSI
multipliers makes this approach very
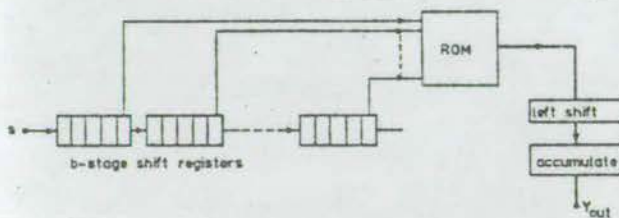attractive for fixed filtering tasks.

Figure 2 : 'Bit slice' filter structure.

The technique usually preferred for
convolutional filter implementation using
analogue devices is a straightforward literal
interpretation of the convolution product
given in equation (1) where all N products
are formed simultaneously using a set of
N multipliers in parallel, see figure 3.
A very attractive way of doing this is
evidenced in the use of charge-coupled
devices (CCD) and analogue MOS. In the
following sections, the basic construction
of a monolithic 256-point CCD programmable
transversal filter [3] (PTF) will be
described and its performance in a number
of key areas will be demonstrated. Although
the analogue approach does not have the
desirable characteristic of arbitrarily
accurate multiplication it does contain the
desirable attribute of a flexible structure
allowing easy alteration of both multiplying
vectors. It is, therefore, ideally suited
to the performance of tasks such as matched
filtering, pulse compression, transform
implementation (in particular CZT and Prime
Transform) and adaptive filtering. The
processor to be described is capable of
performing all these tasks at high speed
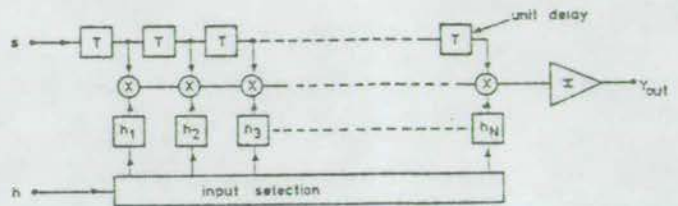with low power consumption.

Figure 3 : Direct parallel transversal filter structure.

MATCHED FILTERING

Our realisation of transveral filter
architecture is a direct implementation of
the classic transversal filter (figure 3) in
integrated circuit form. We have fabricated
integrated matched filters, in linear CTD and
MOS technology, with 64-points and 256-points,
and we here present some details and
applications of these structures.

A block diagram of the device
architecture is given in figure 4 [4]. The
signal register is formed from a tapped
analogue CCD-delay line which simultaneously
realizes the signal storage and time-shift
operations required in the convolution sum.
CTD realization of this element is optimum
for these applications in that the analogue
time delay and shift processes are achieved
naturally, with the most economical use of silicon
area. Because the CCD signal register provides
the necessary time-shift process, a stationary
(spatially "stationary" relative to the signal
information) analogue reference register is
sufficient, supplying the weighting values to the
multipliers in parallel form. An electrically
simple and physically compact realization of this
element uses discrete MOS capacitors for analogue
voltage memory. These feed the multiplier
reference terminals via buffer amplifiers,
and reference values are up-dated individually
via a single digitally multiplexed analogue
input bus. This static analogue reference
memory implies a simpler chip structure than
filters which employ both dynamic signal and
reference registers. Note that it is possible
to exchange the signal and reference register
functions (such that the signal is loaded into
the static register and the reference co-
efficients are time shifted and recirculated)
without changing its function.

However, analogue coefficient
storage is by nature dynamic and thus may
require some form of dedicated, external
memory for refresh purposes. Despite this
two-level memory requirement, the arrangement
is still optimally compact for filters of
more than (approximately) 32 points because
of the relative simplicity of the analogue
multipliers compared with the multiplying DAC
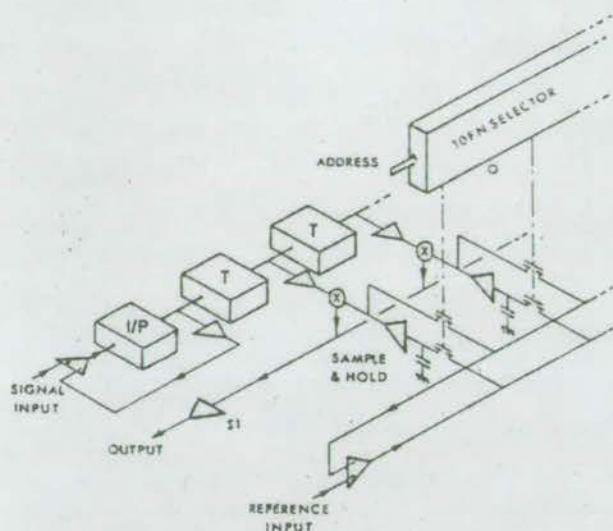structures implied with a single digital-
reference memory architecture.

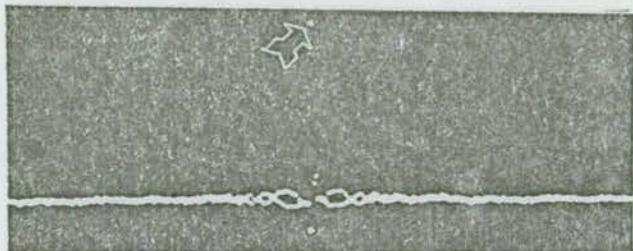Figure 4 : CCD programmable filter block diagram

Figure 5 : 256-point matched filter response for a
full bandwidth chirp input.

For general purpose signal processing,
accurate multiplication of the signal and
reference (weighting coefficient) samples is
required at each filter point.   It is well
known that MOS transistors operating in the
"triode" region potentially offer a compact
multiplying element.   We have used this
approach to realise 4-quadrant analogue
multipliers based on a *single* MOS transistor [4].

The simplicity of our integrated filter
and the minimal number of components involved
are of particular note, and have enabled us
to produce (a) a 64-point PTF with 56μm cell
pitch, having a chip size of 4.5 x 3.3 mm and
dissipating 300 mW, and (b) a 256-point PTF
with 28μm cell pitch, having a chip size of
4 x 3.8 mm and dissipating less than 300 mW.

A prime application of the programmable
filter is as a matched-filter, or correlation
detector.   The achieved bandwidth of prototype
filters is currently into the MHz range, making
their application suitable for sonar and high
speed instrumentation.   Essentially, the
impulse response of the filter is chosen to
be the time-inverse of the waveform to be
coherently detected, and such a pattern of
signals is loaded into the reference capacitor

array.   Suitable waveforms may be chosen such
that the signal energy is time compressed into
a single output peak.   The compression factor
is a function of the time-bandwidth (TB) of
the signal.   Frequently, linearly modulated
FM waveforms or "chirps" are used for sonar

and, for a filter of N-points, the maximum
product that is achievable $TB_{MAX} = N/2$.
Clearly high TB figures are desirable and
may be achieved by cascading our integrated
filters through the CCD signal register.
A 256-point programmable filter module con-
taining four cascaded 64-point devices has
been constructed on a printed circuit board
which measures 16 x 23 cm and dissipates
3 W (of which 1.2 W is contributed by the
correlator chips).   The matched filter
detection of a chirp of TB = 128 using the
256-point module is demonstrated in figure 5.

Charge transfer inefficiency (CTI) in
the CTD delay line ultimately limits the
number of filter points that may be cascaded
with this filter approach, as the signal
disperses along the CCD register.   However,
computer simulation of this effect shows that
results are adequate for $TB_\varepsilon < 1$, where ε is
the CTI value per stage.   Thus for a typical
ε per stage of 10^{-3}, chirps of TB = 1000 may
be processed, implying a total filter length
of at least 2000 points.

From measurements made upon prototype
integrated filters we have found that their
total storage time exceeds 1 s at room
temperature, and the effective tap weight
tolerance is about 2%.   The accuracy of the
filters may be improved by compensating for
multiplier and other inaccuracies using
microprocessor control and has been treated
elsewhere [4].

PRIME TRANSFORM IMPLEMENTATION

The discrete cosine transform (DCT) of
a sequence $\{g_n\}$, n = 0,1,...., (N-1) is
defined as the N-point sequence $\{G_k\}$, K = 0,1,...,
(N-1) where

$$G_0 = \frac{g_0}{2} + \sum_{n=1}^{N-1} g_n \qquad (5)$$

$$G_k = \frac{g_0}{2} + \sum_{n=1}^{N-1} g_n \cos\left(\frac{2\pi nK}{2N-1}\right), \qquad (6)$$

$$K = 1,2,...,(N-1)$$

For a prime-transform (PT) implementation,
(2N - 1) must be prime.

Three operations are involved in using the PT algorithm. The first is permutation (reordering) of the input data using a code derived from a primitive root R of the prime number P (here $P = 2N - 1 = 127$, $R = 3$). Subsequent circular correlation of this reordered sequence with a permuted discrete-cosine sequence yields the DCT coefficients $\{G_k\}$ is permuted order. Reordering the DCT coefficients obtained gives the $\{G_k\}$ in order of increasing frequency.

The significant feature of the PT implementation discussed here is that both the reordering of the input data and the subsequent circular correlation are performed in a single CCD device [6]. Figure 4 shows the CCD filter arrangement, permutation of the data samples. $\{g_n\}$ is achieved by storing these samples in permuted order on the sample-and-hold capacitors under the control of a permutation code r.o.m. Implementation of the circular correlation is achieved by passing the basis vector discrete-cosine sequence $\{W_1\}$ through the CCD channel twice to generate one set of DCT coefficients, while the signal under test is retained on the capacitors. Passing two cycles of the discrete-cosine sequence does not increase the computation time, since the reference register and the channel, start loading simmultaneously, and after N clock cycles the two sequences to be correlated have entered the CCD in their respective positions. This gives the first spectral coefficient G_1. Thereafter, one spectral component (in permuted order) is produced for every subsequent clock cycle. In practice the discrete-cosine sequence cycles continuously through the CCD with suitable time markers and the signal under test can be loaded under command from a timing pulse.

Figure 6 shows the configuration of the processor considered here. The permuter code is stored in r.o.m. 1 and fed to 6 input address lines of the CCD filter. The tap-weight (basis vector) sequence $\{W_1\}$ has values given by

$$W_1 = \cos \left(\frac{2\pi m}{127}\right) \quad 1 = 1,2,3,\ldots, 63$$

where $m = ((3^1))_{127}$ and $((\ ))$ denotes modulo.

This sequence is stored in r.o.m. 2 and, since correlation is required, is fed in reverse order $W_{63}, W_{62},\ldots, W_1$, through the d.a. convertor to the CCD filter.

Computation of the d.c. coefficient (G_0) and the addition of the constant transform offset $(g_0/2)$ has not been considered here.

Figure 7(a) shows the DCT of a signal which consists of one cycle of a cosinusoid within the data frame ($T = 5$ ms) and Figure 7(b) shows the DCT of a signal which consists of four cycles of a cosinusoid within the data frame. The spectral peak is clear, and closer inspection of the output shows the peak to be in the expected permuted position [55] for the one-cycle and [37] for the four-cycle signal. For both of these signals, G_0 equals zero. The frame reference for one output frame is at the left edge of the graticule and is again at the centre line for the next output frame. The nonoptimum sidelobe level has been traced to two defective taps on the CCD transversal filter used for this demonstration. Measurements were made at a clock frequency of 15 kHz and the total power consumption was 4 W. The maximum frequency of operation of the CCD filter is 50 kHz.

ADAPTIVE FILTERING

The final application of the PTF which will be considered here is that of adaptive filtering [7]. This is a particularly attractive mode of operation for an analogue
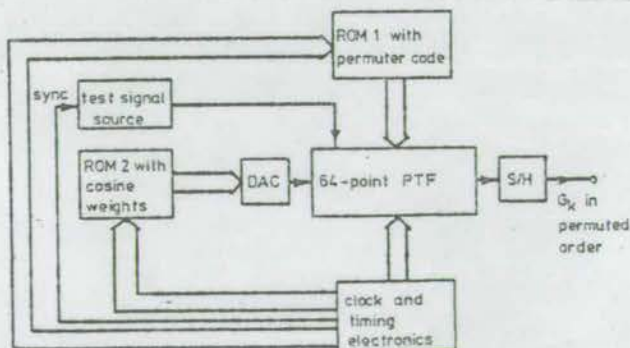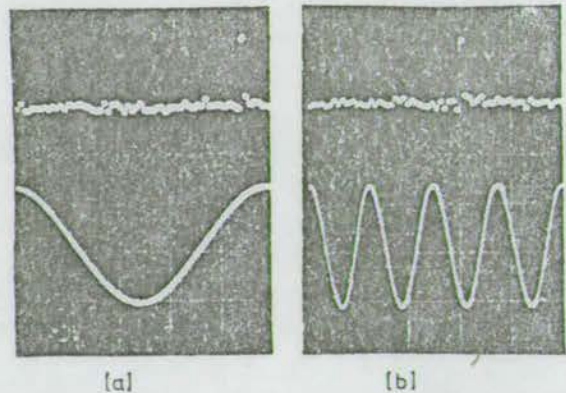

Figure 6 : Block diagram of the DCT processor.

[a]   [b]

Figure 7 : DCT of a signal which consists of
[a] a single cycle of a cosinusoid.
[b] four cycles of a cosinusoid.

CCD device since by its nature an *iterative* adaptive algorithm is tolerant of errors within the filter itself and is capable, to a large degree, of compensating for those errors.

The adaptive algorithm chosen for this system was the clipped version of the Widrow LMS algorithm given by:

$$\underline{H}(t+1) = H(t) + 2\mu \; sgn[\underline{S}(t)] \; e(t) \quad (7)$$

where, $\underline{H}(t)$ is the vector of weight values at time t,
   $S(t)$ is the vector of signal values at time t,
   $e(t)$ is the difference between a target signal d(t) and the filter output at time t, and
   μ is a convergence factor which controls the stability and eventual accuracy of the solution for $\underline{H}(t)$.

The objective of the algorithm is to minimise the difference between the filter output and a target signal d(t) in a least-mean-square sense by iteratively adjusting the weight vector $\underline{H}$ using a gradient descent technique.

Figure 8 shows a general block diagram of the adaptive filter implementation using the 256-point CCD programmable filter device [3] (shown in the dotted box in figure 8). Owing to the fact that only one weight value may be read into the PTF reference at a time there is available a trade-off between convergence time and overall system bandwidth. It is possible to update all the weight values sequentially within a single sample period, as the algorithm implies, but the time necessary to do this would severely restrict the total system bandwidth. It was, therefore, decided to use a 'serial' version of the update algorithm where only one weight value is updated at each sample period. This does of course result in an increase in convergence time but the eventual converged value of the weights is the same as for a parallel system. The time coherence of the algorithm is maintained by sampling the error at one point in every 256 and then multiplying this error by the sign of the signal sample present at the last tap of the CCD. This product is then added to the appropriate weight value in a digital accumulator. In this module the error e(t) is converted directly to digital form; and this operation only has to be done once in every 256 input sample periods, which means that a long conversion time is available to the ADC. The signal on the CCD last tap is clipped and digitally delayed to allow time for the ADC convergence, and multiplication is then performed digitally using exclusive-OR gates (sgn[s(t)].e(t)). We are therefore able to use a slow, low power, monolithic ADC while still retaining the full bandwidth of the PTF. The convergence factor μ is simply implemented using a digital
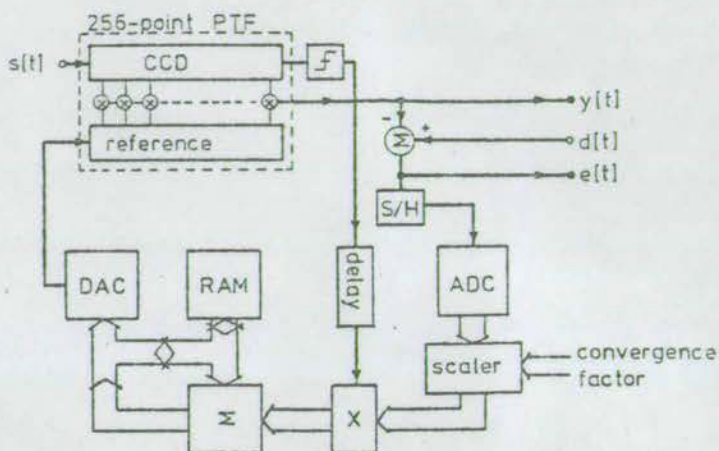

Figure 8 : Adaptive filter block diagram

scaling function which yields eight distinct convergence factors between 2^{-5} and 2^{-12}, in integral powers of 2.

The complete system described above was constructed on 3 international-sized circuit cards, had a bandwidth of 200 kHz and a power consumption of 7.5 W. A more detailed description of the system and results demonstrating its performance may be found in reference 7.

The final result presented in figure 9 demonstrates the operation of the adaptive filter in a practical signal processing environment. The incoming signal is the output of a telephone line (a simulation of 5 miles of telephone cable) carrying a base - band digital pulse train. It is desired to compensate for the signal distortion introduced by the line, therefore reconstituting the original data. One technique used to achieve this is to transmit a known data sequence for a short time during which the adaptive filter is trained with the known sequence. After training, the receiver-generated training signal is replaced by a signal derived from a decision circuit at the filter's own output which allows the filter to track minor variations in transmission line characteristics. The results, shown in figure 9, is for the part of the cycle where the filter is being specifically trained so that it may approximate a suitable equaliser impulse response. The eye pattern of the filter input and output (figure 9(d) and (e)) show that a considerable improvement in signal detectability is achieved.

CONCLUSIONS

It has been shown that the CCD programmable transversal filter offers a viable and attractive alternative to digital processors in the field

(a) line output

(b) training signal

(c) filter output

(d) eye pattern of line output
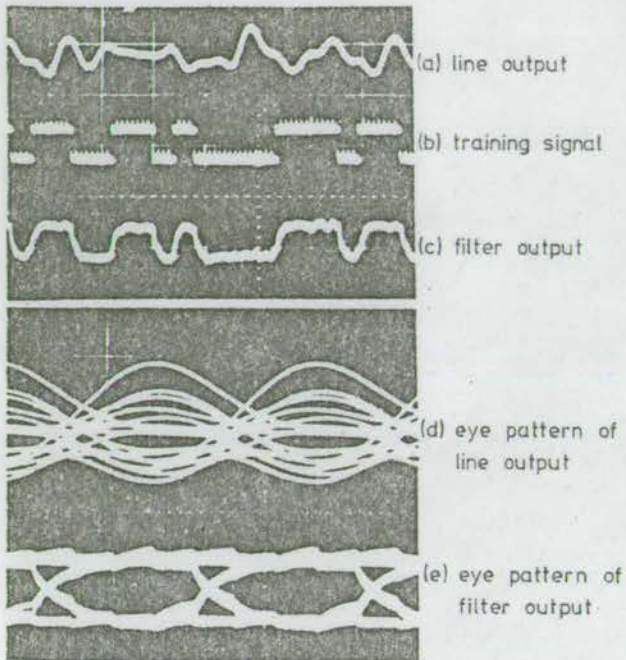
(e) eye pattern of filter output

Figure 9 : Performace of the adaptive filter as a line equaliser.

of fast, programmable, convolutional processors. The programming flexibility of the CCD filters described means that they are ideally suited to a number of important applications in signal processing. In particular, the tasks demonstrated in this paper of matched filtering, adaptive filtering and DCT implementation may be performed using a minimum of peripheral circuitry, apart from the PTF itself, and therefore speed and power consumption is optimised. The results presented here indicate that compact, lower power, flexible, analogue PTF sub-systems with TB products of 1000 at power levels less than 10 W will be a reality in the near future. These are expected to impact such applications areas as communications and sonar signal processing, particularly where space and power are at a premium.

In conclusion, it has been demonstrated that analogue charge-coupled device technology does have a significant role to play in the design of future signal processing systems. This is particularly true for applications where processing precision is secondary to high bandwidth requirements and power and processor size are crucial.

REFERENCES

[1] Mavor, J., Jack, M.A., Saxton, D., and Grant, P.M.: 'Design and performance of a programmable, real-time charge-coupled device recirculating delay-line correlator', IEE J. Electron. Circ. and Syst., 1977, 1, (4), pp.137-143.

[2] Peled, A., and Liu, B.: 'A new hardware realization of digital filters', IEEE Trans. ASSP, 1974, 22, (6), pp.456-462.

[3] Denyer, P.B. and Mavor, J.: '256-point programmable transversal filter', Proc. 5th Int. Conf. on CCD, Edinburgh, 1979, pp.253-254.

[4] Denyer, P.B., Mavor, J. and Arthur, J.W.: 'Miniature programmable transversal filter using CCD/MOS technology', Proc. IEEE, 67, (1), Jan. 1979, pp.42-50.

[5] Bailey, W.H., Buss, D.D., Hite, L.R., and Whatley, M.W.: 'Radar video processing using the chirp Z-transform', Proc. Int. Conf. on Applications of CCD, San Diego, 1975, pp.283-290.

[6] Kapur, N., Mavor, J., and Jack, M.A.: 'Discrete cosine transform processor using a CCD programmable transversal filter', Electronics Letters, 1980, 16, (4), pp.139-141.

[7] Cowan, C.F.N. and Mavor, J.: 'Miniature CCD-based analogue adaptive filters', IEEE ICASSP80, April 1980, pp.474-478.