UNIVERSITY OF SOUTHAMPTON

FREQUENCY SAMPLING DIGITAL FILTERS

by

Alexander Hugh Campbell Fraser

Thesis submitted for the degree of Master of Philosophy
November 1977

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND APPLIED SCIENCE

ELECTRONICS

Master of Philosophy

FREQUENCY SAMPLING DIGITAL FILTERS

by Alexander Hugh Campbell Fraser

The aims of this investigation were to examine alternative realisation of frequency sampling digital filters, to analyse and predict the noise performance and overflow behaviour of each realisation, to devise an efficient method of measuring filter performance and to examine the effects of changes in filter parameters upon the performance.

Three bandpass digital filter realisations were examined, simulated on a general purpose digital computer, and their performance compared on the basis of a measurement yielding gain, phase and output noise level. Effects upon performance of rounding, truncating, signal amplitude and phase, and resonator damping were also investigated. Provided the arithmetic wordlength was set at 16 bits, good agreement between the predicted and measured performance was obtained; shorter wordlengths gave a degradation in performance and less consistent noise measurements. Overflow analysis showed that two of the realisations suffered from severe limitation of the useful frequency range.

The results of the investigation have shown that the frequency sampling technique suffers from several major drawbacks, in particular a high noise level at the output of the filter which is maintained at an acceptably low level at the expense of wordlength, a need for finely quantised coefficients to give satisfactory approximation to the desired filter characteristic and an inordinately long period of measurement to give consistent values of output noise. A brief review of some possible hardware implementations showed that the frequency sampling technique demanded a high degree of complexity.

It is concluded that the frequency sampling technique does not compare favourably with other modern filtering techniques.

Acknowledgements

The author wishes to extend his thanks to the Captain, Dean, Deputy Dean and Director of Electrical Engineering at the Royal Naval Engineering College, Manadon for their support and encouragement during the period of this project and also to Mr D G Appleby of the Department of Electronics for his supervision and help.

In addition, the author wishes to thank all those with whom he has discussed problems raised during the course of the investigation, in particular Lieutenant J Riggs, Royal Navy for his contribution to software development as part of his final year undergraduate project at the College.

LIST OF SYMBOLS

$a(i)$	real constant coefficient
A_0	dc component of filter output
A	gain of filter measured by best-fit method
$A_1(k)$	weighting factor of the Type I and Type II filters
$A_1'(k)$	modified weighting factor (Types I and II filters)
$A_2(k)$	weighting factor of the Type III filter
$A_2'(k)$	modified weighting factor (Type III filter)
$b(i)$	real constant coefficient
B	filter bandwidth
c	coefficient wordlength
$e(n)$	filter output noise due to $\varepsilon(n)$
$e(q)$	analogue-to-digital converter error
$\left\{E_n\right\}_{rms}$	rms noise output
f_0	resonator frequency
F	noise figure in dB
$g(n)$	general impulse response
$G(z)$	general pulse transfer function
$h(n)$	filter impulse response
$h_r(n)$	resonator impulse response
$H(j\omega)$	continuous frequency response
$H(k)$	sampled value of continuous frequency response
$H(z)$	filter pulse transfer function
$H_r(z)$	resonator pulse transfer function

k	sample number = $\frac{\text{frequency} \times \text{number of samples}}{\text{sampling frequency}}$
M	number of samples used in best fit analysis
N	number of samples
NG	noise gain
NSR	noise-to-signal ratio
q	analogue-to-digital converter step size
r	radial distance of pole from origin = $1 - \delta$
t	arithmetic wordlength
$t(n)$	sampling instant
T	sampling period
$u(n)$	intermediate signal sequence
$w(n)$	intermediate signal sequence
$x(n)$	input sequence
$X(z)$	z-transform of $x(n)$
$y(n)$	output sequence
$Y(z)$	z-transform of $y(n)$
z^{-1}	unit delay
z^{-i}	i-th order delay
z^{-N}	N-th order delay.

γ	attenuation factor
δ	radial distance of pole from unit circle
$\varepsilon(n)$	input noise
$\varepsilon_i(n)$	noise from i-th noise source
θ	resonator angle = $2\pi k/N = 2\pi f_0 T$
σ^2_{suff}	variance of signals or noise, 'suff' defined in text.
τ	filter time constant
ϕ	phase of filter measured by best fit method
$\phi(k)$	phase at k-th sample
ω	continuous angular frequency

CONTENTS

CHAPTER ONE

AN OUTLINE OF THE PROBLEM

Out of all the literature available on the subject of digital filtering, the frequency sampling technique has received relatively little attention. The technique is only one of several employed to solve the approximation problem for finite impulse response digital filters but it does offer some interesting aspects for investigation as it utilises components which have infinite impulse responses. However, it is not a technique that lends itself readily to the formulation of a set of design rules as does for example a recursive digital filter modelled on well-known analogue filter designs. However, it could be a useful technique for the design of filters having arbitrary amplitude response together with linear phase. Its major disadvantages compared to other design techniques for finite impulse response filters are,

a. it relies upon exact pole-zero cancellations to give the desired frequency response,

b. high gain, high-Q resonators form an integral part of the design,

c. it is economically feasible only when few frequency samples are non-zero compared with the possible number of frequency samples.

Considerations equally applicable to other methods of digital signal processing include the choice of word length, the choice of the type of arithmetic, the method of multiplication and the effect of non-linearities in the processor. Of particular importance in any system employing registers whose bit lengths are limited is the problem of noise at the output of the system due to rounding or truncating the products of multiplication. This aspect has received a good deal of attention and models have been proposed which enable predictions of noise performance to be made with a fair degree of accuracy. The noise performance of frequency sampling digital filters is particularly important since they employ digital resonators whose properties are such that it is reasonable to expect high quantisation noise levels.

Noise analyses are readily available in the literature albeit mainly confined to first-order and second-order recursive sections. Even so, by making suitable assumptions, it is relatively straightforward to adapt the analyses to high gain, high-Q resonators.

As will be shown later, a compromise must be struck between maintaining the basis of the technique of frequency sampling while at the same time attempting to obtain satisfactorily low noise levels.

Having made predictions of noise performance, it is of course essential that they be tested in a practical filter. As far as can be ascertained, no noise performance figures have been given in the literature, nor have any comparisons been made between different realisations of frequency sampling filters. In order to make measurements on a filter without having to actually build one requires the use of a large computer having double precision facilities. Ideally, the computer should offer a real-time capability but such facilities are not readily available. However, provided the simulation programs offer sufficient flexibility so that changes in word length, signal amplitude, signal frequency etc may be effected, the loss of real-time operation is not so serious. Nevertheless, as it must always be borne in mind that the eventual object of any investigation in signal processing is to meet the optimum method of implementing the system, the simulation should wherever possible be geared to that end. For example, a simulation using 24-bit words would be unrealistic and wasteful when present hardware allows a maximum of 16-bits for digital words. Specially made hardware for particular applications is, of course, excluded from this premise.

Thus in this investigation, it is essential that alternative realisations be examined, noise performance predictions be made for each realisation, a method of measurement be devised to effect a suitable flexible simulation and hardware proposals be made which utilise up-to-date technology.

The aims for this investigation based upon the foregoing requirements are:

a. to examine alternative realisation of frequency sampling digital filters,

b. to analyse and predict the noise performance for each realisation,

c. to simulate the realisation on a general purpose digital computer,

d. to devise an efficient method of testing the filter performance to yield gain, phase and noise information,

e. to examine the effects upon filter performance of changes in parameters ie word lengths, rounding or truncating, etc,

f. to suggest possible methods of hardware implementation.

In order to use a realistic example of a filter for investigation, a post-detection radar doppler filter specified in Figure 1.1 was chosen. This filter is made up of four contiguous bandpass filters to yield an overall bandwidth of 1600 Hz. In a radar system it would be necessary to provide a high-pass filter enveloping the doppler filter in order to exclude close range clutter. As one of the advantages of digital filters is that arbitrary responses may be approximated, there is no need to provide such a filter, the leading edge of the first band-pass section sufficing for this purpose.

The thesis presented hereon gives in Chapter 2 a general introduction to those aspects of digital filtering relevant to the investigation. It is inappropriate to give a self-contained and comprehensive theoretical introduction to the subject but some aspects must be included to form a foundation for the theory presented in later chapters. Chapter 3 presents a brief review of previous work in this field, mentioning in particular the theory of frequency sampling filters, quantisation noise analysis and hardware implementation. The theory of frequency sampling filters is given in Chapter 4 where three different filter realisations are examined. The realisations are based upon proposals made in the existing literature, the aim of this chapter being to collect together the results of previous work rather than to derive the transfer functions ab initio.

Having now presented the relevant theory leading up to the formulation of transfer functions, a theoretical noise analysis on each realisation is performed in Chapter 5. The majority of the analyses are based on work by other researchers in the field, the major addition to this being an analysis of noise performance using double precision accumulators.

Also included in Chapter 5 is the theory of a best-fit method of filter performance measurement. The method utilises both input signal and output signal to yield gain response, phase response and noise output of the filter. It assumes a sinusoidal input. In summary, Chapters 4 and 5 lay down the theory of the filters to be tested and a method of testing suitable for use on a general purpose computer simulation.

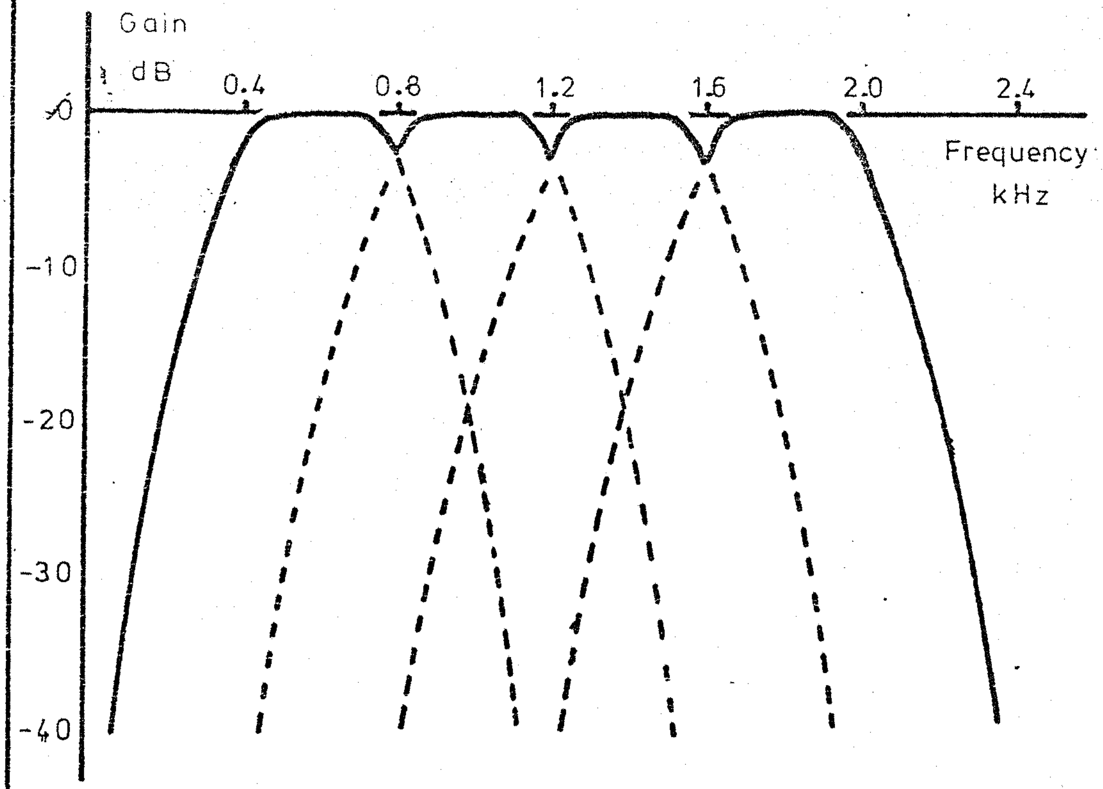
Chapter 6 discusses the factors which must be taken into account in designing a suitable simulation. As already stated, frequency sampling filters do not lend themselves easily to the formulation of general design rules so that it is necessary to fix some of the parameters which are dictated by the filter specification.

For example, the minimum number of resonators necessary to meet the specification must be determined, the resonator damping factor must be decided and, with hardware implementation in mind, the maximum word significance must be given careful consideration. It is perfectly feasible to devise a general purpose simulation program but such a program would probably be very cumbersome requiring a mass of input data in addition to that specifying the frequency and weighting factor of each resonator. For this reason, the simulation programs described in Chapter 7 and listed in the Appendix have been written specifically for the filter described in Figure 1.1.

Details of the measurements and tests performed on the simulation are given in Chapter 8.

Chapter 9 presents a brief survey of the possibilities for filter implementation using read-only-memory table look up for multiplication and using microprocessors in conjunction with pre-packaged multipliers. In addition mention is made of the residue number system which is currently being proposed as an alternative method in fast multiplication and addition implementations.

Finally, Chapter 10 gives a summary of the conclusions which have been drawn from the results of the investigations and makes recommendations concerning further work in this field.



Filter Specification

Sampling Frequency	4600 Hz
Doppler Filter Spacing	400 Hz
Centre Frequency of First Filter	600 Hz
Response w.r.t. Centre Frequency	
	-3 dB at ±200 Hz
	-20 dB at ±400 Hz
Minimum Stopband Attenuation	-40 dB
Passband Ripple	1 dB

Figure 1.1 Doppler Filter Specification

CHAPTER TWO

GENERAL INTRODUCTION

2.1 Principles of Digital Filtering.

Systems having sampled i.e. discrete, signals as inputs and outputs are characterised by difference equations as distinct from differential equations which characterise continuous i.e. analogue systems. In the case of digital filters, the difference equations describing the filters express the output sequence, y(n) as either

a. the sum of weighted delayed versions of the input sequence x(n)

$$y(n) = \sum_{i=0}^{p} a(i)x(n-i) \qquad 2.1$$

or b. the sum of weighted delayed versions of the output and the input

$$y(n) = x(n) - \sum_{i=1}^{p} a(i)y(n-i) \qquad 2.2$$

or c. a combination of a. and b.

$$\sum_{i=0}^{m} b(i)y(n-i) = \sum_{i=0}^{p} a(i)x(n-i) \qquad 2.3$$

In the equations given, m and p are positive integers, a(i) and b(i) are real constants. For the sake of brevity, the sampling period T has been set to unity.

Equation 2.3 represents the most general expression for the difference equation of a digital filter from which, by applying z-transform theory, the pulse transfer function of the filter may be derived. The pulse transfer function H(z) is defined as the ratio of the z-transform of the output sequence Y(z) to the z-transform of the input sequence X(z),

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^{p} a(i)z^{-i}}{b(0) + \sum_{i=1}^{m} b(i)z^{-i}} \qquad 2.4$$

This pulse transfer function is realisable and stable if

a. a(i) and b(i) are real time-invariant constants. These coefficients are equivalent to the coefficients in analogue filters determined by the circuit components which are rarely time-invariant.

b. the poles of the transfer function defined by the roots of the denominator polynomial lie within the unit circle (Reference 2.1). The analogue equivalent is the condition that the poles must lie in the left-hand region of the s-plane.

Two cases of equation 2.4 are of interest. The first is when a(i) and b(i) are non-zero. This condition yields an output sequence which is a function of past and present values of both the output and input sequences. This implies a feedback system, usually termed a recursive system. The pulse transfer function includes zeros and poles, and under certain circumstances, the filter can become unstable. The impulse response of a recursive filter will be of infinite duration.

The second case is when b(0) = 1, all other coefficients b(i) being zero. The pulse transfer function contains zeros only so that the output sequence is a function of the input sequence alone. The system is described as non-recursive, it is inherently stable and, provided p is finite, the impulse response is finite. Although it is often possible to define filters in terms of the duration of their impulse response, it does not necessarily follow that recursive filters are always infinite impulse response filters. For example, the frequency sampling technique utilises a recursive realisation preceded by a comb filter so producing a finite response. This arrangement is the subject of the present investigation. Thus, it is recommended that the terms recursive and non-recursive be used to define filter realisations.

Gold and Rader (Reference 2.2) have shown that a second-order transfer function offers distinct noise advantages and may be used as a basic element to realise higher-order filters by connecting the elements in a cascade or parallel arrangement. Such a second-order recursive element having a weighting factor of unity applied to the output ie b(0) = 1, is given by

$$H(z) = \frac{a(o) + a(1)z^{-1} + a(2)z^{-2}}{1 + b(1)z^{-1} + b(2)z^{-2}} \qquad 2.5$$

This second-order section is characterised by the difference equation

$$y(n) = a(o)x(n) + a(1)x(n-1) + a(2)x(n-2) - b(1)y(n-1) - b(2)y(n-2) \qquad 2.6$$

Figure 2.1 shows the direct realisation of equation 2.6 in which individual delays, z^{-1}, are used for the zeros and for the poles. In hardware terms as well as in computational speed, this realisation is uneconomic and can suffer from severe coefficient sensitivity (Reference 2.8).

An alternative realisation, usually described as the canonic form is shown in Figure 2.2. It has a minimum number of delays, multipliers and adders and is characterised by the difference equation pair

$$\begin{aligned} w(n) &= x(n) + b(1)w(n-1) + b(2)w(n-2) \\ y(n) &= a(0)w(n) + a(1)w(n-1) + a(2)w(n-2) \end{aligned} \qquad 2.7$$

Examination of the two realisations show that the poles precede the zeros in the canonic form whereas in the direct form, the reverse is true. The direct form can have certain noise advantages, the trade-off being in computational speed and coefficient sensitivity.

Having defined two types of digital filter form i.e. recursive and non-recursive, it is usual to define performance in terms of the duration of the impulse response. The design techniques are well documented, for example by Rabiner and Gold (Reference 2.3). Since this investigation is concerned with a particular form of finite impulse response filter, a survey of the design techniques for this form follows. Only a brief description of design techniques for infinite impulse response filters will be given.

2.2 Design Techniques for Finite Impulse Response Filters.

Finite impulse response filters are particularly useful in approximating filters of arbitrary amplitude/frequency response coupled with linear phase. They are also designed without recourse to classical filter theory.

One method of design starts with the determination of the impulse response coefficients, h(n), which are then set into the expression

$$H(z) = \sum_{n=0}^{N-1} h(n)z^{-n} \qquad 2.8$$

where N is the number of coefficients being considered. Given the desired frequency response, the impulse response is found from

$$h(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(j\omega)\ \exp(j\omega n)\ d\omega \qquad 2.9$$

The desired continuous frequency response being given by

$$H(j\omega) = \sum_{n=-\infty}^{\infty} h(n)\ \exp(-j\omega n) \qquad 2.10$$

Two major difficulties arise in the computation of the impulse response. In the first place, the Fourier series coefficients from equation 2.9 will in general extend to infinity but in any case will be significant beyond the number N. Secondly, the more complicated the desired frequency response, the more difficult it becomes to describe H(jω).

Obviously, the Fourier series must be truncated but simple truncation introduces the overshoot and ripple associated with Gibb's phenomenon. A solution may be found by causing the Fourier series to converge rapidly by applying a time-limited weighting function, or window, which is multiplied with the Fourier coefficients. This is the process of convolving the desired frequency response with the window frequency response, and is described in detail in Reference 2.11.

The second difficulty may be reduced in magnitude by making use of the discrete Fourier transform of sampled values of the frequency response. Suppose N samples, H(k), of the desired frequency response are used to compute the impulse response via

$$h(n) = \frac{1}{N} \sum_{n=0}^{N-1} H(k)\ \exp(j2\pi k/N) \qquad 2.11$$

The values of h(n) so computed are set into equation 2.8 to give the pulse transfer function of the filter. This technique is called the frequency sampling technique. By means of a linear programming procedure, optimum values of the frequency samples in the transition regions may be computed (Reference 2.5). This procedure is in effect equivalent to finding an optimum window for the Fourier series method of design (Reference 2.6). Chapter 4 will show how this technique can be developed to give a pulse transfer function having a linear phase characteristic.

2.3 Design Techniques for Infinite Impulse Response Filters.

A very comprehensive survey of design techniques has been presented by Rabiner and Gold (Reference 2.3). In essence, two approaches can be made (Reference 2.4). In the first, use is made of the well-established design methods used for analogue filters having relatively simple closed-form design formulae. For example, if the desired frequency response of the filter is approximated satisfactorily by a second-order Butterworth analogue filter, it is very straightforward to make s-plane to z-plane substitution, very often using the bilinear z-transform, to determine the pulse transfer function. If, on the other hand, the frequency response is arbitrary, design may be effected by means of computer aided methods although this seems to be rarely employed. One particular form of an infinite impulse response system is a digital resonator, which is a system intended to produce an output at a particular frequency. Its analogue equivalent is the high-Q bandpass filter which if incorporated into a suitable feedback network, will yield self-sustaining oscillations. However, the analogue system is limited inasmuch as its parameters are not time-invariant and high gain, high-Q circuits are very difficult to implement.

2.4 Digital Resonators.

The general expression for a digital resonator pulse transfer function is given by Gold and Rader (Reference 2.2) as,

$$H(z) = \frac{1 - vz^{-1}}{1 - 2r.z^{-1}\cos\theta + r^{2}z^{-2}} \qquad 2.12$$

The frequency of the resonator is defined by the resonator angle θ

$$\text{where} \quad \theta = 2\pi f_0 T = \omega_0 T \qquad 2.13$$

f_0 being the resonator frequency. The coefficient r is the radial distance of the complex conjugate poles from the origin of the unit circle ie the poles are given by $p_{1,2} = r\exp(\pm j\theta)$.

Obviously, if the resonator is to be high-Q, r must lie very close to unity. The gain of the resonator is a function of r and of the value assigned to v in equation 2.12. as the following cases illustrate.

Case 1 $v = r$

$$H_r(z) = \frac{1 - rz^{-1}}{1 - 2r\cos\theta z^{-1} + r^2 z^{-2}} \qquad 2.14$$

Using conventional techniques, or geometrical techniques shown in Figures 2.3 to 2.5, the gain is found to be approximated by

$$\text{Gain at resonance} \approx \frac{\left[2(1 - \cos\theta)\right]^{\frac{1}{2}}}{2\delta\sin\theta} = \frac{1}{2\delta\cos\frac{\theta}{2}} \qquad 2.15$$

where $r = 1 - \delta$ and $\delta \ll 1$

It is evident that the gain is frequency dependent.

Case 2 $v = 0$

Here the zero lies at the origin of the unit circle. Making the same approximations as before,

$$\text{Gain at resonance} \approx \frac{1}{2\delta\sin\theta} \qquad 2.16$$

Case 3 $v = r\cos\theta$

The zero in this case is now a function of frequency which, as it appears in the numerator will offset the frequency dependence given by the previous cases.

$$\text{Gain at resonance} \approx \frac{1}{2\delta} \qquad 2.17$$

Thus, the introduction of the frequency dependent zero yields a resonator with constant gain.

Analytical methods of determining the bandwidth of the resonator are cumbersome, the geometrical interpretation yielding a good approximation for all three cases as

$$\text{Bandwidth} = \frac{2\delta \times \text{sampling frequency}}{2\pi} \text{ Hz} \qquad 2.18$$

In terms of computational speed, Cases 1 and 3 are very similar. Case 1 will be discarded as it has the disadvantage of a severely frequency dependent gain. Only Cases 2 and 3 will be considered in the following chapters.

In this part of the chapter, principles of digital filtering together with methods of obtaining approximations to continuous frequency responses have been presented briefly. Chapter 4 will return to this aspect when the pulse transfer function of the frequency sampling digital filter will be derived. Having determined the pulse transfer function, it is necessary to consider word length limitations and their effects upon filter performance.

2.5 Quantisation Effects in Digital Filters.

Chapter 5 of Reference 2.3 gives a thorough treatment of finite word length effects inherent in digital systems employing registers of finite length to store filter coefficients and signals. In this section, a brief resumé of some of the effects will be presented, more detailed analysis applied to digital resonators being given later in Chapter 5. The analysis will be based upon those presented by Weinstein (Reference 2.7).

In many cases, digital filters will be required to process data derived from an analogue source when it is necessary to employ an analogue-to-digital converter. Since the output word length is limited, it consists of a sequence of numbers together with an error $e(q)$ bounded by

$$-\frac{q}{2} < e(q) < \frac{q}{2} \qquad 2.19$$

q is the step size taken by the converter and is assumed constant. If speech were being processed, then step size might not be constant, non-linear quantisation being employed in the same way as in PCM systems.

It is justifiable to regard the error as a white noise having zero mean and variance $\sigma_e^2 = q^2/12$. Further, it is assumed that the error is uncorrelated from sample to sample and that it is uncorrelated with the signal.

Turning now to the filter coefficients, these are initially specified with so-called infinite precision to give the desired approximation. It is these coefficients that define the filter response so that any departure from the precise values will cause differences in the filter response, even to the extent of causing instability. It is usual to estimate the sensitivity of the filter related to the pole positions by examining, in the case of the digital resonator, the points of intersection of concentric circles representing the quantisation of r^2 and vertical lines corresponding to the quantisation of $r \cos \theta$ in equation 2.12. This is shown in Figure 2.6.

For the Case 2 resonator having two complex conjugate poles at $r\exp(\pm j\theta)$ and a zero at $z = 0$, the difference equation is

$$\begin{aligned} y(n) &= x(n) + b(1)y(n-1) - b(2)y(n-2) \\ \text{where}\quad b(1) &= + 2r\cos\theta \quad\text{and}\quad b(2) = r^2 \end{aligned} \qquad 2.20$$

Sensitivities are defined by differentiating r and θ with respect to $b(1)$ and $b(2)$.

$$\Delta r = \frac{\Delta b(2)}{2r}\ ;\ \Delta\theta = \frac{\Delta b(2)}{2r^2\tan\theta} - \frac{\Delta b(1)}{2r\sin\theta} \qquad 2.21$$

It can be seen from Figure 2.6a that for resonators with θ close to zero or to 180°, the angular error is large giving rise to problems in low pass filters, Ideally, the coefficient sensitivity should be constant (and small) over the full range of θ and to this end, a coupled form of resonator has been proposed (Reference 2.7).

The characterising difference equations are

$$\left.\begin{aligned} y(n) &= r\cos\theta\ y(n-1) - r\sin\theta\ u(n-1) + x(n) \\ u(n) &= r\sin\theta\ y(n-1) + r\cos\theta\ u(n-1) \end{aligned}\right) \qquad 2.22$$

As seen from Figure 2.6b, this form offers a distinct improvement in coefficient sensitivity over the direct form albeit at reduced computational speed.

Turning now to the signal word length, the major problem arises at the output of the multipliers where a double length word is generally to be expected. Rounding or truncation may be used, or even a combination of both. Truncation is the more difficult method of word length reduction to analyse because the error is very dependent upon the type of arithmetic being used. A full treatment of this topic can be found in Chapter 9 of Reference 2.4. Whatever the method of limitation used, certain assumptions must be made. In addition to those already given for analogue-to-digital conversion, the assumption that the signal passes through several quantisation levels from sample to sample is made. Chapter 5 will present a fuller analysis of the effect of quantisation error due to rounding upon the performance of a frequency sampling digital filter.

If the input to a filter is suddenly reduced to zero, an output can appear which oscillates about a particular non-zero level. This phenomenon is called limit cycling and has been analysed for rounding (Reference 2.8) and for truncation (References 2.9 and 2.10). In all analyses, dead-band regions are defined in which limit cycling can take place, as shown in Figure 2.7 where the cross-hatched regions represent values of b(1) and b(2) in Equation 2.20 for which limit cycling is absent.

Certain advantages are offered by truncation. Firstly, limit cycles present in a second-order filter using truncation are similar to those in a rounded first-order filter. Secondly, the ranges of b(1) and b(2) for no limit cycling are greater than for rounding. However, the disadvantage is that the signal-to-noise ratio falls by 6 dB which may be recouped only at the expense of increasing the signal word length by 1 bit.

Alternatively, limit cycling may be avoided by ensuring that the input signal never falls to zero i.e. by applying a dummy signal at the filter input. In practical cases, the signal input is rarely likely to fall to zero, there being noise usually present to prevent this.

Finally, there is the problem of overflow when the register length is exceeded. One approach to avoid this condition is to scale the input to the filter so that the maximum input signal can never cause any node in the filter to overflow, even though the probability that the input will reach its maximum permitted value is remote. Alternatively, the input may be allowed to exceed the maximum permitted by the scaling criterion and to employ saturating adders which if overflow occurs, clamp the sum to a maximum value. Allowing saturation to occur means that the output of the filter is allowed to be distorted.

2.6 Applications of Digital Filters.

Digital signal processing, embracing digital filtering, is being applied to fields which have in the past relied upon analogue signal processing. This is in no small part due to the rapid development of digital integrated circuit technology which now allows stringent system performance specifications to be met. Digital integrated circuits have inherent advantages over analogue circuits, in particular, high accuracy, parameter stability, high reliability and low cost.

Digital filters have a wide variety of applications either in the bettering of performance of analogue filters in situations where conventional R, L, C filters are either insufficiently stable or inordinately cumbersome as in the case of very low frequency filters, or where the signal processing requirement is beyond the scope of analogue techniques, for example very narrow-band notch filters.

Other advantages of digital filters lie in their ability to be rapidly reprogrammed to yield filters of different response while employing the same hardware, and in their suitability for use in a multiplexed role.

On the other hand, the disadvantages are cost, complexity and operating speed although over the past few years, all these important factors have been improved. For example, microprocessor development is now offering 16-bit systems with clock rates of over 10 MHz, programmable read-only-memories allow more efficient use to be made of look-up table methods of multiplication and direct-memory-access facilities offer faster computational speeds.

2.7 Summary.

Some of the topics on digital filtering considered to be relevant to the frequency sampling design technique have been presented. Much of the material is applicable to all digital filtering design techniques, but in the frequency sampling techniques employing as it does high gain, high-Q resonators, the effects of quantisation are perhaps the most important. The available literature, well referenced in References 2.3 and 2.4, indicates that in such systems, a large word length is essential. Bearing in mind that many of the papers were written at a time when fast 16-bit processors were considered to be unattainable for some time, there tends to be an emphasis on the examination of systems having shorter word lengths. Now that 16-bit processors are relatively common the emphasis is now moving towards methods of increasing computational speed.

2.8 References.

2.1 Bogner, R.E. and Constantinides, A.G., 'Introduction to Digital Filtering, Chapter 2, Wiley, 1975.

2.2 Gold, B. and Rader, C.M., 'Digital Processing of Signals' McGraw-Hill, 1969.

2.3 Rabiner, L.R. and Gold, B., 'Theory and Applications of Digital Signal Processing', Prentice-Hall, 1975.

2.4 Oppenheim, A.V. and Schafer, R.W., 'Digital Signal Processing', Prentice-Hall, 1975.

2.5 Rabiner, L.R., Gold, B. and McGonegal, C., 'An Approach to the Approximation Problem for Non-recursive Digital Filters', IEEE, Trans. Audio and Electroacoustics, AU-18, No 2, June 1970, 83-106.

2.6 McCreary, T.J., 'On Frequency Sampling Digital Filters', IEEE Trans. Audio and Electroacoustics, AU-20, No 3, August 1972, 222-223.

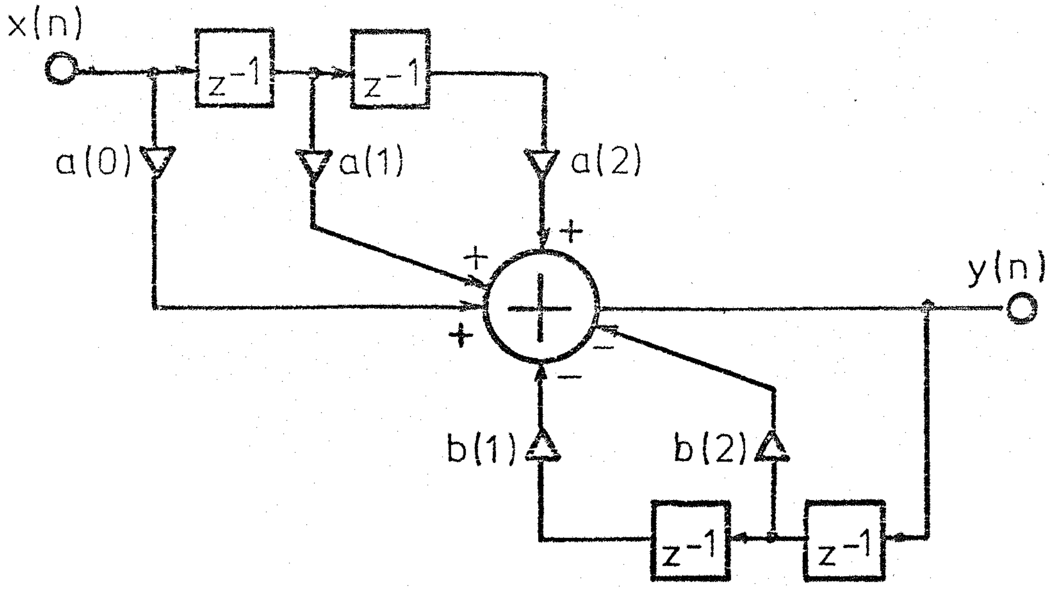
2.7 Weinstein, C.J., 'Quantisation Effects in Digital Filters', MIT Lincoln Lab. Report 468, November 1969.

2.8 Jackson, L.B., 'Roundoff Noise for Fixed-Point Digital Filters Realised in Cascade or Parallel Form', IEEE Trans. Audio and Electroacoustics, AU-18, No 2, June 1970, 107-122.

2.9 Chih-yu Kao, 'An Analysis of Limit Cycles due to Sign-Magnitude Truncation and Multiplication in Recursive Digital Filters', Proc. ASILOMAR Conf. Circuits and Systems, November 1971, 349-353.

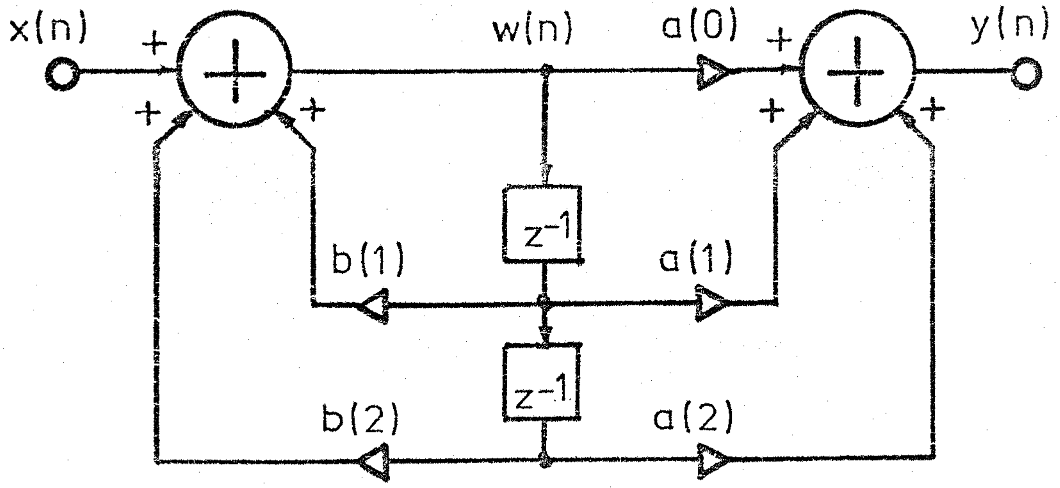
2.10 Thong, T. and Liu, B., 'Limit Cycles in Combinatorial Filters using Two's-Complement Truncated Arithmetic', 8th ASILOMAR Conf. Circuits, Systems and Computers, 51-55.

2.11 Rabiner, L.R., 'Techniques for Designing FIR Digital Filters', IEEE Trans. Communication Technology, COM-19, No 2, April 1971, 188-195.



$$y(n) = a(0)x(n) + a(1)x(n-1) + a(2)x(n-2) - b(1)y(n-1) - b(2)y(n-2)$$

Figure 2.1 Direct Realisation



$$w(n) = x(n) + b(1)w(n-1) + b(2)w(n-2)$$
$$y(n) = a(0)w(n) + a(1)w(n-1) + a(2)w(n-2)$$

Figure 2.2 Canonic Realisation

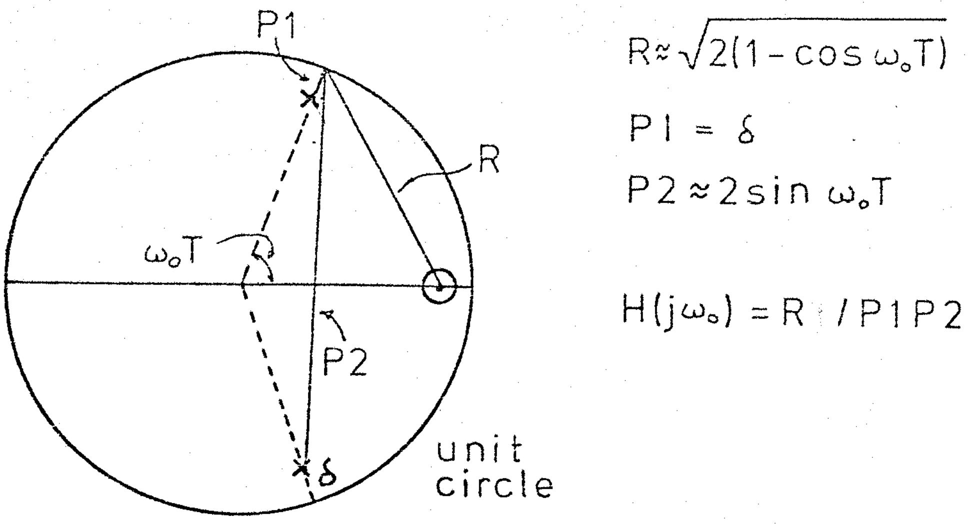

P1
R
ω_oT
P2
δ
unit circle
$R \approx \sqrt{2(1-\cos\omega_oT)}$
$P1 = \delta$
$P2 \approx 2\sin\omega_oT$
$H(j\omega_o) = R\ /P1P2$

Figure 2.3 Resonator with Zero at $z=r$

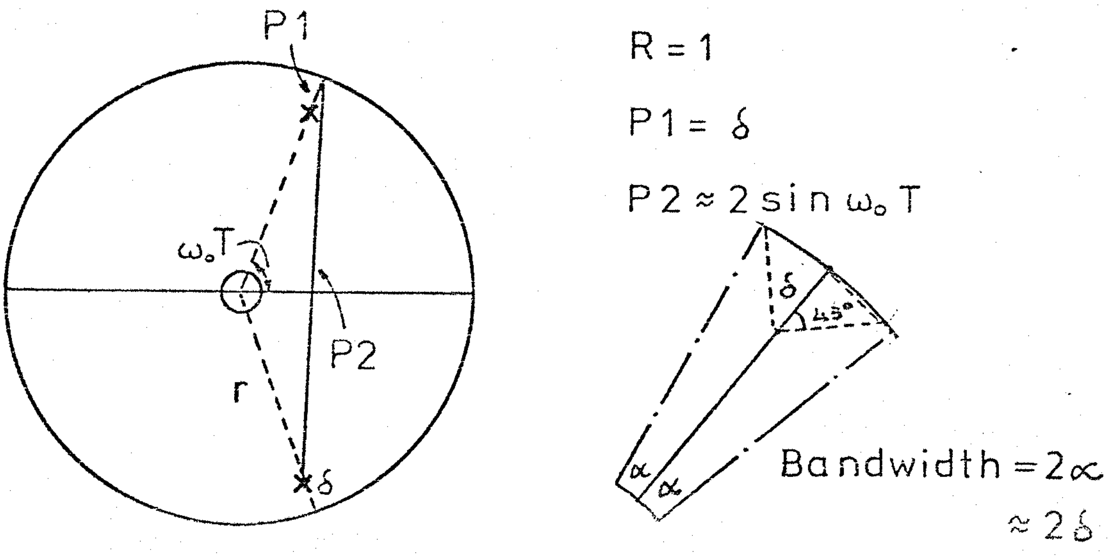


Figure 2.4 Resonator with Zero at $z=0$

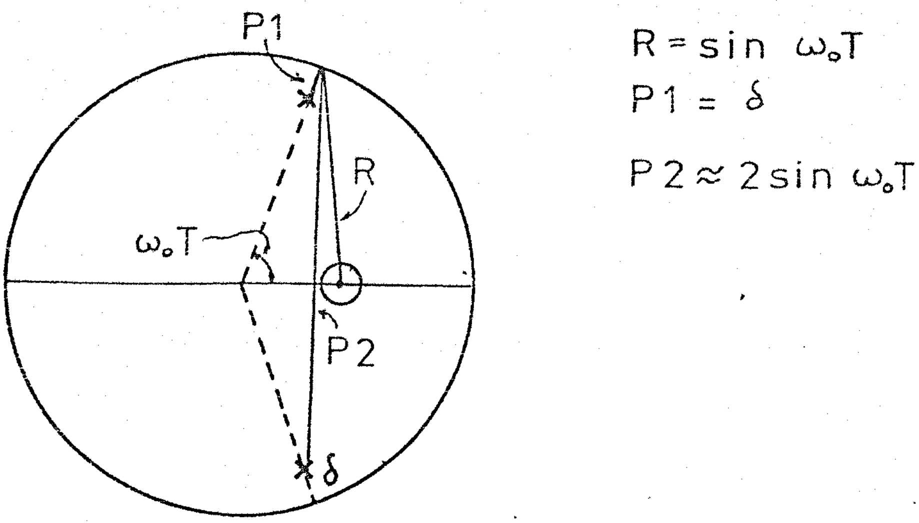
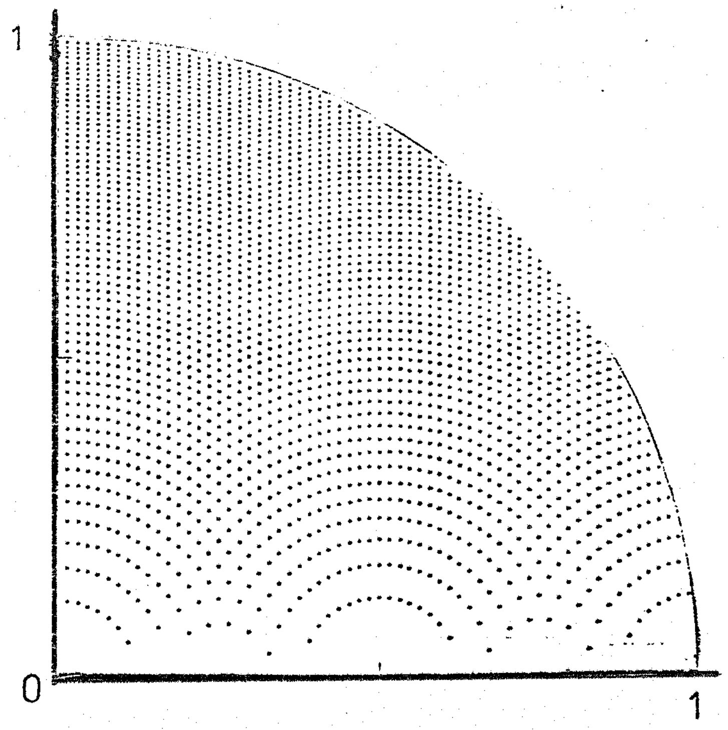
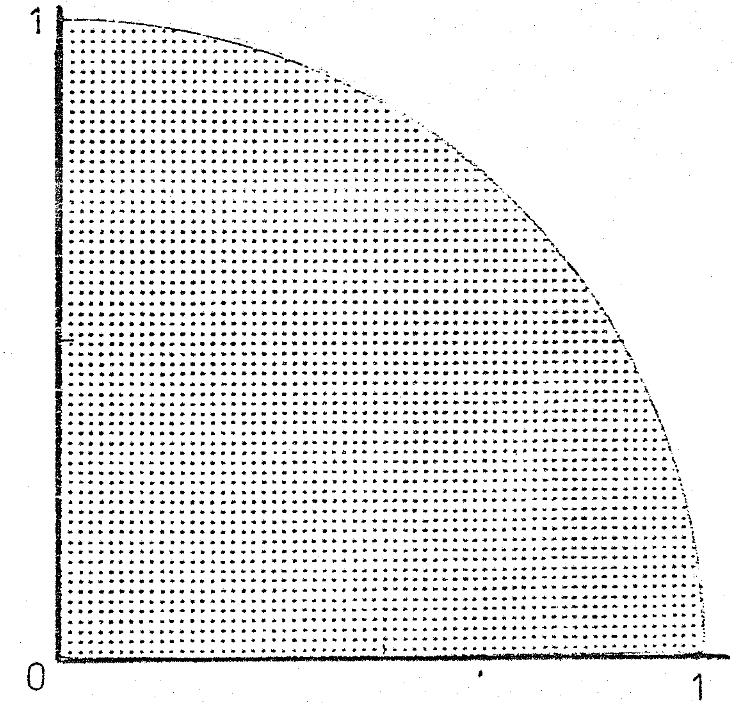


Figure 2.5 Resonator with Zero at $z=r\cos\omega_oT$

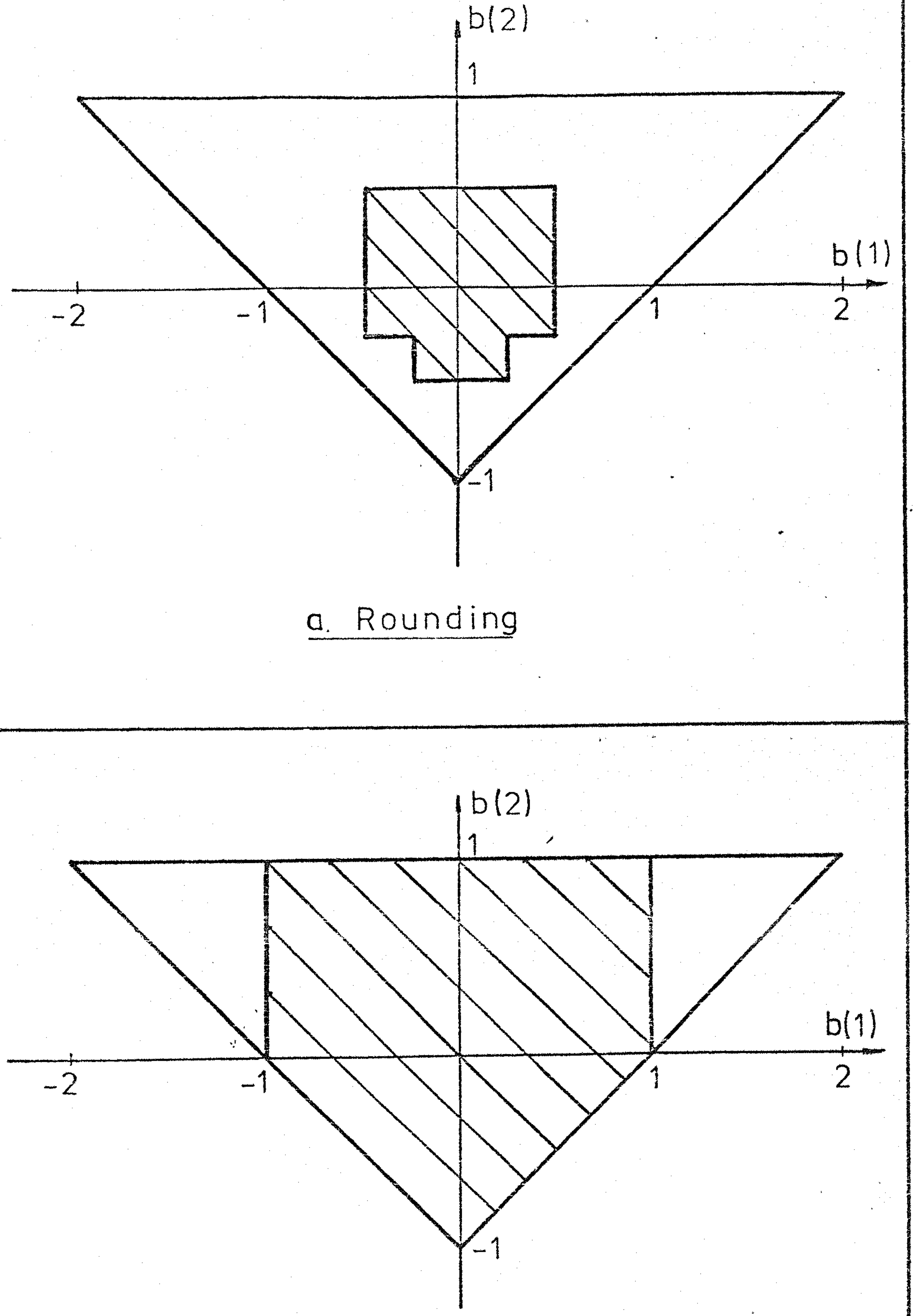


a. Direct Form Second-Order Section
(6-bit quantisation)



b. Coupled Form Second-Order Section
(6-bit quantisation)

Figure 2.6 Grid of Allowable Pole Positions



a. Rounding

b. Truncation

Figure 2.7 Limit Cycling Deadband Regions

CHAPTER THREE

REVIEW OF PREVIOUS WORK

3.1 Introduction.

The quantity of literature available on the subject of digital signal processing and digital filtering is considerable and no attempt will be made to give a comprehensive survey covering the field. On the other hand, the amount of literature specifically devoted to the frequency sampling technique is somewhat limited, almost to the point where it appears that the technique is of only academic interest. This is perhaps because of the relative inflexibility of the technique, other methods such as the equiripple algorithm or the use of windows offering better methods for general frequency selective filter design. Notwithstanding, the frequency sampling technique provides an interesting example of recursive section being used to realise filters having a finite impulse response and concentrates attention on the problem of the quantisation noise/word length compromise essential in systems which have poles lying close to the unit circle.

Whatever particular design method is chosen, there is an inevitable interdependence between allied techniques. This review is consequently a summary of those areas which relate specifically to frequency sampling filters and is divided into four main topics, namely, theory of frequency sampling filters, quantisation effects, realisations, hardware implementations. Of these, the last named topic is currently receiving renewed interest even though the volume of literature covering digital filtering has decreased sharply of late.

3.2 Theory of Frequency Sampling Digital Filters.

The frequency sampling technique is theoretically possible using analogue techniques but because of the high demands placed upon the resonators has been feasible only since the advent of digital techniques. Gold and Rader (Reference 3.1) describe the principle of operation quite simply as a system in which a comb filter having m zeros is followed by a bank of n resonators such that $m > n$, each resonator possessing complex conjugate pole pairs which cancel a corresponding comb filter zero. The input is applied to the comb filter, the outputs of each resonator are weighted and added together, the resulting filter having a finite impulse response of duration $m \times$ (sampling period), a frequency response with linear phase, and an amplitude response which agrees with the specification of the filter at the sampling points.

Bogner (Reference 3.2) likens the technique to nailing down the comb response at regular intervals and then pulling out some of the nails. Whatever the particular imagery associated with this technique, all authors agree that frequency sampling filtering is the equivalent of filtering by Fourier transforming, multiplying by a filter frequency function and inverse transforming.

Following closely behind Gold and Rader, Weinstein (Reference 3.3) extended the theory to include specific linear phase constraints, at the same time examining the problems of noise in such a system. Rabiner et al. (Reference 3.4) developed a general theory, making comparisons between this frequency sampling technique and the search for an optimum window used in non-recursive design, a conclusion which was supported by McCreary (Reference 3.5).

In a later paper by Rabiner and Schafer (Reference 3.6), a more detailed analysis showed that subject to certain constraints, the filter realisation proposed by Weinstein could be simplified to reduce hardware requirements and to increase computational speed. However, as Rabiner and Gold (Reference 3.7) and Burlage et al (Reference 3.8) point out, the resultant filter impulse response is symmetrical about a non-integer number of samples.

On a more general basis, Oppenheim and Schafer (Reference 3.9) and Rabiner, Kaiser et al (Reference 3.10) have compared infinite impulse response filters with those having finite impulse response. No definite conclusion is reached as to which type is superior, only a suggestion that a figure of merit may be defined in terms of the number of multiplications per sample leading to a bias in favour of infinite impulse response realisations. If linear phase is essential, the finite impulse response realisations may be justified despite the extra cost.

The theory of the frequency sampling filter was laid down at a very early stage in the evolution of digital filtering, other aspects which follow being the subjects of wider investigations.

3.3 Quantisation Effects.

There is no doubt that this aspect of digital signal processing has received and continues to receive a marked degree of attention. Weinstein (Reference 3.3) and Liu (Reference 3.11) give detailed summaries of the literature up to 1971, a sequel to those being published in a paper by Claasen et al (Reference 3.12).

In parallel with these, Oppenheim and Weinstein (Reference 3.13) give a tutorial review of the theory covering representation of numbers, truncation, rounding and the formulation of noise-to-signal ratios.

As already discussed in Chapter 2, quantisation effects in digital filters can be classified into those caused by coefficient quantisation and those caused by multiplier roundoff. This does not imply that each cause is independent for Fettweis (Reference 3.14) has shown that there is a direct and indirect connection between the two. He interprets the direct connection of the rounding process as a fluctuation in the coefficients of ideal multipliers producing finite length results even in the absence of rounding while the indirect connection is a result of the fact that the multiplier complexity is solely a function of the signal word length. Because of this, it is possible to trade off coefficient word length against signal word length without seriously impairing the frequency characteristic. In practical terms this means that the signal word length may be increased at no extra cost in complexity with the additional advantage that the roundoff noise is reduced.

A further reduction in roundoff noise may be effected by accumulating partial sums in an extended length register prior to final rounding (or truncating).

In the case of coefficient quantisation, it has been found difficult to give an analytical treatment although an estimate of the displacement of poles and zeros due to multiplier coefficient quantisation has been made (Reference 3.15). In Reference 3.21, relationships are given which allow estimates to be made of the relevant word lengths given the analogue-to-digital converter characteristics, the noise gain of the filter and the desired noise figure due to roundoff.

3.4 Realisations.

As a consequence of investigations into the effects of pole displacements, a search has been made for realisations which exhibit better pole distributions without increasing filter complexity. By constructing a grid of allowable pole positions (Reference 3.13), it is possible to define a pole sparsity factor (Reference 3.16) which in turn leads to a search for realisations having a low pole-sparsity factor. Some new realisations have been proposed (Reference 3.17) but it is unfortunate that all increase the filter complexity.

Pole sparsity factors are of particular significance when designing low-pass filters but are of little importance in the case of bandpass filters when direct realisation may be employed (Reference 3.18).

3.5 Hardware Implementations.

So far, the work described has been of a theoretical nature. The next major step is to implement the chosen realisation as effectively as possible. In view of the regular advances in hardware technology, the designer is likely to be faced with the prospect of having to use obsolescent techniques. Assuming for the moment that the pace of development is not a major design factor, attention focusses onto the potentially slowest element in the system; the multiplier. Early attempts at increasing multiplier speed (Reference 3.19) used conventional logic circuitry which was quickly overtaken by the introduction of integrated circuit multipliers which are at present capable of producing a 32-bit product in less than half a microsecond. Other researchers (Reference 3.20) have attempted to devise special representation of the processing coefficients by coding them in such a way that a minimum number of add/subtract operations are required in the multiplication process. This represents a move away from the pre-packaged multiplier chip. A figure of merit has been proposed, defined as the ratio of the input data rate in kHz to the number of MSI TTL integrated circuits required, thus giving a good measure of the cost of the implementation. Figures of merit of 3 times greater than pre-packaged multiplier implementations have been claimed.

Progressing from MSI to LSI, attention has been given to the implementation of digital filters using microprocessors (Reference 3.21). First-order filters have been implemented using 8-bit microprocessors and fixed point arithmetic capable of sampling rates up to 1 kHz. The sampling rate would be very much lower for higher-order or more complex filters although it is possible to improve performance by using pre-packaged hardware multipliers as peripheral devices.

Departing from the LSI approach, the residue number system has re-emerged after lying dormant for some time (Reference 3.22). This system provides a capability for high speed multiplication when used in conjunction with fast access read-only-memories. Computational speeds using this system compare very favourably with conventionally implemented filters.

3.6 Summary.

In view of the overwhelming quantity of literature available on the subject of digital filtering, it has only been possible to present a brief review of certain aspects particularly relevant to frequency sampling filtering. The theory of this technique is well documented as is the theory of quantisation effects due to limited word lengths. Realisations are very often dictated by the particular filter application. The problems of implementation are, on the other hand, not yet solved, several options being open to the designer.

3.7 References.

3.1 Rader, C.M. and Gold, B., 'Digital Filter Design Techniques in the Frequency Domain', Proc. IEEE, Vol 55, No 2, February 1967, 149-171.

3.2 Bogner, R.E. and Constantinides, A.G., 'Introduction to Digital Filtering', Wiley, 1975.

3.3 Weinstein, C.J., 'Quantisation Effects in Digital Filters', MIT Lincoln Lab. Report 468, November 1969.

3.4 Rabiner, L.R., Gold, B. and McGonegal, C., 'An Approach to the Approximation Problem for Non-recursive Digital Filters', IEEE Trans. Audio and Electroacoustics, AU-18, No 2, June 1970, 83-106.

3.5 McCreary, T.J., 'On Frequency Sampling Digital Filters', IEEE Trans. Audio and Electroacoustics, AU-20, No 3, August 1972, 222-223.

3.6 Rabiner, L.R. and Schafer, R.W., 'Recursive and Non-recursive Realisations in Digital Filters designed by Frequency Sampling Techniques', IEEE Trans. Audio and Electroacoustics, AU-19, No 3, September 1971, 200-207.

3.7 Rabiner, L.R., and Gold, B., 'Theory and Application of Digital Signal Processing', Prentice-Hall, 1975.

3.8 Burlage, D.W., Houts, R.C. and Vaughan, G.L., 'Time Domain Design of Frequency Sampling Digital Filters for Pulse Shaping using Linear Programming Techniques'. IEEE Trans. Acoustics, Speech and Signal Processing, ASSP-22, No 3, June 1974, 180-185.

3.9 Oppenheim, A.V. and Schafer, R.W., 'Digital Signal Processing', Prentice-Hall, 1975.

3.10 Rabiner, L.R., Kaiser, J.F., Harmann, O. and Dolen, M.T., 'Some Comparisons between FIR and IIR Digital Filters', BSTJ, Vol 53, No 2, February 1974, 305-331.

3.11 Liu, B., 'Effect of Finite Word length on the Accuracy of Digital Filters - A Review', IEEE Trans. Circuit Theory, CT-18, No 6, November 1971, 670-677.

3.12 Claasen, T.A.C.M., Mecklenbräuker, W.F.G. and Peek, J.B.H., 'Effects of Quantisation and Overflow in Recursive Digital Filters', IEEE Trans. Acoustics, Speech and Signal Processing, ASSP-24, No 6, December 1976, 517-529.

3.13 Oppenheim, A.V. and Weinstein, C.J., 'Effect of Finite Register Length in Digital Filtering and Fast Fourier Transform', Proc. IEEE, Vol 60, No 8, August 1972, 957-976.

3.14 Fettweis, A., 'On the Connection between Multiplier Word Length Limitation and Roundoff Noise in Digital Filters', IEEE, Trans. Circuit Theory, CT-19, No 5, September 1972, 486-491.

3.15 Mitra, S.K. and Sherwood, R.J., 'Estimation of Pole-Zero Displacements of a Digital Filter due to Coefficient Quantisation', IEEE Trans. Circuits and Systems, CAS-21, No 1, January 1974, 116-124.

3.16 Rahman, M.H. and Fahmy, M.M., 'On Pole Distribution in Digital Filters', Circuit Theory and Applications, Vol 3, 1975, 95-100.

3.17 Agarwal, R.C., 'On the Realisation of Digital Filters' PhD Thesis, Rice University, 1974.

3.18 Jackson, L.B., 'Roundoff Noise Analysis for Fixed-Point Digital Filters Realised in Cascade or Parallel Form', IEEE Trans. Audio and Electroacoustics, AU-18, No 2, June 1970, 107-122.

3.19 Jackson, L.B., Kaiser, J.F. and McDonald, H.S., 'An Approach to the Implementation of Digital Filters', IEEE Trans. Audio and Electroacoustics, AU-16, No 3, September 1968, 413-421.

3.20 Peled, A. 'On the Hardware Implementation of Digital Signal Processors', IEEE Trans. Acoustics, Speech and Signal Processing, ASSP-24, No 1, February 1976, 76-86.

3.21 Ahmed, N. and Jayapalan, J.P., 'On Digital Filter Implementation via Microprocessors', IEEE Trans. Industrial Electronics and Control Instrumenaation, IECI-23, No 3, August 1976, 249-253.

3.22 Jenkins, W.K. and Leon, B.J., 'The Use of Residue Number Systems in the Design of Finite Impulse Response Digital Filters', IEEE Trans. Circuits and Systems, CAS-24, No 4 April 1977, 191-201.

CHAPTER FOUR

THEORY OF THE FREQUENCY SAMPLING DIGITAL FILTER

4.1 Introduction.

The theory of the frequency sampling digital filter has already been discussed briefly in Chapters 2 and 3 and is covered comprehensively in the literature (References 4.1 to 4.9).

The principle underlying this technique is that a comb filter having N equispaced zeros around the unit circle is followed by a bank of resonators having poles which cancel exactly some of the zeros in the comb filter, each resonator output being weighted by an amount equal to the value of the desired frequency response at the resonator frequency, as shown in Figure 4.1. The values of the frequency samples H(k) are chosen as zero in the stopband, unity in the passband and fractions in the transition bands. This technique is particularly attractive for narrow-band frequency selective filters where only a few of the frequency response samples are non-zero. In this application, the technique can be more efficient than other forms of filter even though the frequency sampling implementation will in general require more hardware complexity, particularly in the form of data storage.

Furthermore, because of its modular nature, the filter lends itself particularly well to time multiplexing of the resonator sections and, as multiplier speeds increase, to time multiplexing within each section.

Its main disadvantage lies in the inflexibility of specifying the passband and stopband cutoff frequencies since the placing of the frequency samples is constrained to integer multiples of (sampling frequency ÷ number of samples). Increasing the number of samples improves the flexibility but reduces the efficiency. This technique is therefore not suitable for general frequency selective filter designs.

In the analyses which follow, consideration of bandpass frequency sampling digital filters will be given more attention than general filter specifications.

4.2 The General Frequency Sampling Filter Transfer Function.

A finite duration sequence can be represented by its Discrete Fourier Transform so that if $H(k)$ are the samples of a desired continuous frequency response $H(j\omega)$ taken at N equispaced frequencies $f(k) = k/NT$ where T is the sampling period and $k = 0, 1, 2 \ldots\ldots N - 1$ as shown in Figure 4.1a,

$$H(k) = H(j\omega)\Big|_{\omega = 2\pi k/NT} \qquad 4.1$$

Using the Discrete Fourier Transform, the impulse response

$h(n)$ is given by

$$h(n) = \frac{1}{N}\sum_{k=0}^{N-1} H(k)\exp(j2\pi kn/N) \qquad 4.2$$

$$n = 0, 1, \ldots\ldots N - 1$$

whence by taking the z-transform of equation 4.2, the pulse transfer function $H(z)$ is yielded,

$$H(z) = \sum_{n=0}^{N-1} h(n)z^{-n} \qquad 4.3$$

which has the property that

$$H(z)\Big|_{z = \exp(j2\pi k/N)} = H(k) \qquad 4.4$$

Substituting equation 4.2 into equation 4.3 and re-arranging terms gives

$$H(z) = \frac{1 - z^{-N}}{N}\sum_{k=0}^{N-1}\frac{H(k)}{1 - \exp(j2\pi k/N)z^{-1}} \qquad 4.5$$

As shown in Figure 4.1b, equation 4.5 may be realised using a comb filter, $(1 - z^{-N})$, having zeros at $z(k) = \exp(j2\pi k/N)$ in cascade with a parallel connection of resonators having complex pole pairs occurring at the zeros of the comb filter.

The impulse response of a filter described by equation 4.5 will be complex but in practical systems it is essential that the impulse response be real. This requirement imposes a symmetry upon the frequency samples such that

$$|H(k)| = |H(N - k)| \tag{4.6}$$

and also upon the phase of the samples

$$\phi(k) = - \phi(N - k) \tag{4.7}$$

Assuming for the moment that N, the number of samples, is even and applying equations 4.6 and 4.7 to equation 4.5,

$$\begin{aligned} H(z) = \frac{1 - z^{-N}}{N} \Bigg\{ & \sum_{k=1}^{\frac{N}{2}-1} \frac{|H(k)| \exp\left[j\phi(k)\right]}{1 - \exp(j2\pi k/N)z^{-1}} \\ & + \sum_{k=\frac{N}{2}+1}^{N-1} \frac{|H(k)| \exp\left[j\phi(k)\right]}{1 - \exp(j2\pi k/N)z^{-1}} \\ & + \frac{H(0)}{1 - z^{-1}} + \frac{H\left(\frac{N}{2}\right)}{1 + z^{-1}} \Bigg\} \end{aligned} \tag{4.8}$$

Equation 4.8 may be simplified by combining the complex conjugate terms to give

$$\begin{aligned} H(z) = \frac{1 - z^{-N}}{N} \Bigg\{ & \sum_{k=1}^{\frac{N}{2}-1} \frac{2|H(k)| \left[\cos \phi(k) - \cos(\phi(k) - 2\pi k/N)\right] z^{-1}}{1 - 2z^{-1} \cos(2\pi k/N) + z^{-2}} \\ & + \frac{H(0)}{1 - z^{-1}} + \frac{H\left(\frac{N}{2}\right)}{1 + z^{-1}} \Bigg\} \end{aligned} \tag{4.9}$$

Equation 4.9 represents the general expression for the transfer function of a frequency sampling digital filter. It may be further simplified by imposing additional constraints upon the phase characteristics of the filter.

4.3 Linear Phase Constraints.

In this section, slightly different linear phase constraints will be applied to equation 4.9. Each resulting filter will be described by a Type I, II or III filter which will also contain information regarding the form of realisation.

The Type I Filter

The Type I filter is defined as that filter which has a linear phase response specified by

$$\phi(k) = -k\pi \qquad k = 0, 1, \ldots.. \left(\frac{N}{2} - 1\right) \qquad 4.10$$

Substituting equation 4.10 into equation 4.9, at the same time setting $H(0) = 0$ for the case of the bandpass filter, the pulse transfer function becomes

$$H(z) = \frac{1 - z^{-N}}{N} \sum_{k=1}^{\frac{N}{2}-1} \frac{(-1)^k 2|H(k)| \left[1 - \cos(2\pi k/N)z^{-1}\right]}{1 - 2\cos(2\pi k/N)z^{-1} + z^{-2}} \qquad 4.11$$

Note that in order to satisfy the linear phase constraint of equation 4.10, $H\left(\frac{N}{2}\right)$ must be zero. Linear phase highpass filters are not easily designed using this technique.

In the case when N is even, the impulse response is symmetrical about $h\left(\frac{N}{2}\right)$ and in the case when N is odd, the upper limit of summation in equation 4.11 is replaced by $(N - 1)/2$, the impulse response now possessing no unique peak, the centre of symmetry lying halfway between two samples.

Figure 4.2 shows the realisation of the Type I filter defined by equation 4.11. It is the direct form in which the zero preceeds the poles within the resonator section.

Type II Filter

The Type II filter differs from the Type I filter solely in its realisation. Figure 4.3 shows the realisation which is of the canonic form in which the zero follows the poles in the resonator section.

Both Type I and Type II filters come under the Case 3 digital resonator described in Chapter 2. The gain of each resonator at resonance is independent of the frequency of operation because the zero itself is frequency dependent.

Type III Filter

Rabiner and Schafer (Reference 4.4) have presented an analysis of the frequency sampling filter based upon a different phase constraint to that used in the Type I filter analysis, namely

$$\phi(k) = \frac{\pi k(N - 1)}{N} \qquad k = 0, 1, \ldots\ldots \frac{N}{2} - 1 \qquad 4.12$$

Here the phase characteristic has been shifted by one half sample which should give directly opposite characteristics of the impulse response from those given by the Type I filter. The reasoning upon which the choice of phase characteristic is given in Reference 4.2. If the set of frequency samples $H(k)$ is a real symmetric sequence, then the interpolated frequency response derived from equation 4.5 contains a small imaginary component. This component will disappear if the symmetry of the impulse response is shifted by one half sample but in the case of a symmetrical bandpass filter would vanish in any case. More important perhaps is that the half sample shift simplifies the expression for the pulse transfer function.

Substituting equation 4.12 into equation 4.9 yields

$$H(z) = \frac{(1 - z^{-N})}{N} (1 - z^{-1}) \sum_{k=1}^{\frac{N}{2}-1} \frac{(-1)^k 2|H(k)|\cos(\pi k/N)}{1 - 2\cos(2\pi k/N)z^{-1} + z^{-2}} \qquad 4.13$$

Once again, for N odd, the upper limit of summation becomes $(N - 1)/2$.

In this type, the resonator zero becomes common to all resonators and may therefore be implemented as a common element lumped with the comb filter. Such a strategy naturally simplifies the structure of the resonator as shown in Figure 4.3 but at the expense of a frequency dependent gain as shown in Chapter 2 under the Case 1 digital resonator. As will be shown later in Chapter 6, the frequency dependence is a disadvantage when resonators operating beyond the quarter sampling frequency are required.

In practice, the poles and zeros of the system are moved slightly inside the unit circle, so avoiding inexact cancellations and possible instability.

This is effected by replacing z^{-1} in the pulse transfer function expressions by rz^{-1}, where r is very close to unity such that $r = 1 - \delta$, δ being of the order of 2^{-c} where c is the coefficient word length. However, the choice of c is critical and will be considered later.

4.4 Summary.

Transfer functions for frequency sampling digital filters have been derived using two linear phase constraints suggested in the literature. Both derived transfer functions have the same general form consisting of a comb filter followed by a bank of resonators, the essential differences being the delay of the filter, the symmetry properties of the impulse response and the gain of the individual resonators. Of those, the last is considered to be the most important for, despite the hardware and computational speed advantages of the Type III filter, it is susceptible to overflow at particular frequencies which rather restricts its scope of application. However, in Chapter 6 consideration will be given to a method of avoiding the overflow problem.

4.5 References.

4.1 Gold, B. and Rader, C.N., 'Digital Processing of Signals', McGraw-Hill, 1969.

4.2 Rabiner, L.R., Gold, B. and McGonegal, C., 'An Approach to the Approximation Problem for Non-recursive Digital Filters'. IEEE Trans. Audio and Electroacoustics, AU-18, No 2, June 1970, 83-106.

4.3 Rabiner, L.R., 'Techniques for Designing FIR Digital Filters', IEEE Trans. Communication Tecnniques, COMM-19, No 2, April 1971, 185-195.

4.4 Rabiner, L.R. and Schafer, R.W., 'Recursive and Non-recursive Realisations of Digital Filters Designed by Frequency Sampling Techniques', IEEE Trans. Audio and Electroacoustics, AU-19, No 3, September 1971, 200-207.

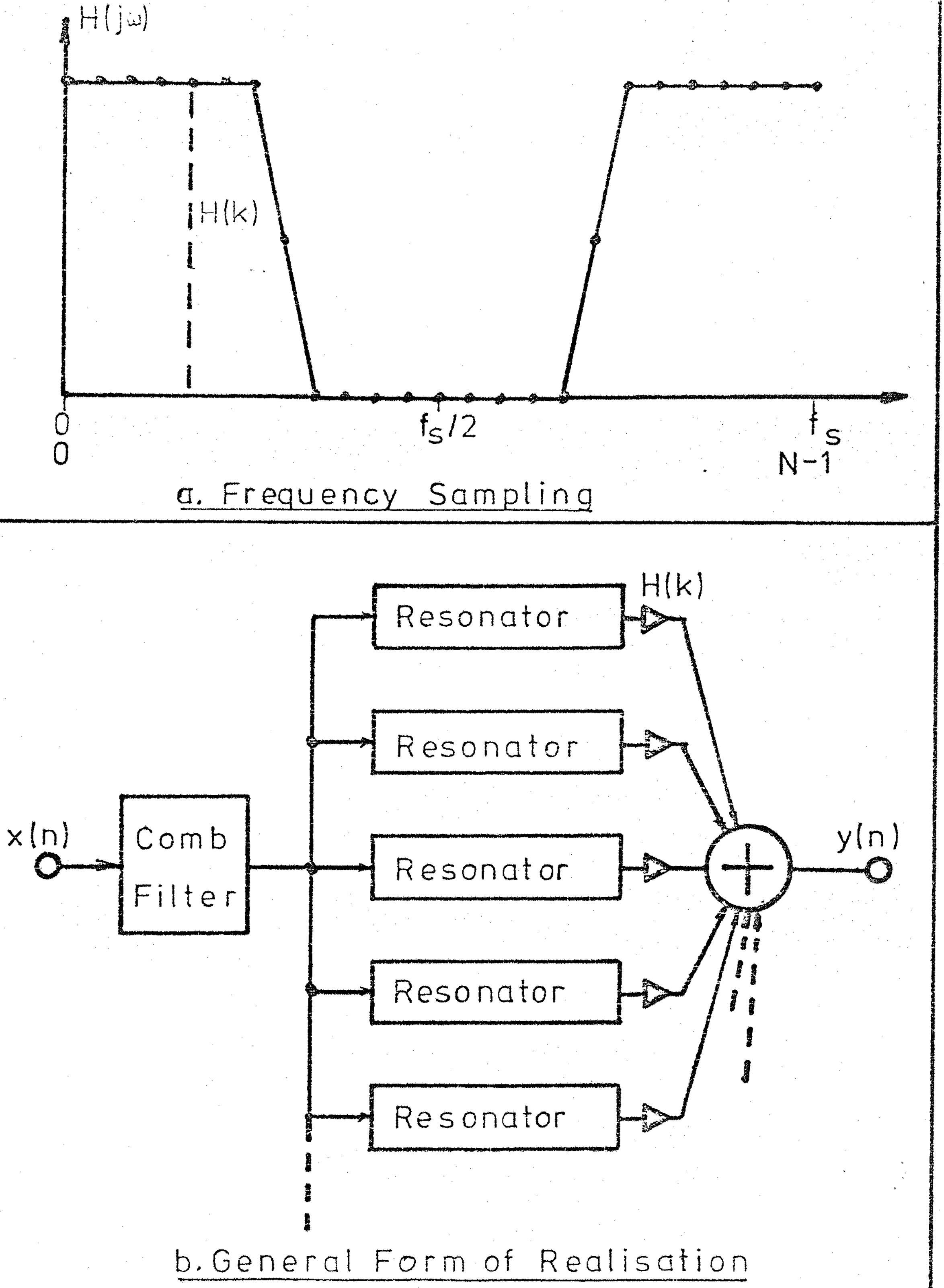
4.5 Weinstein, C.J., 'Quantisation Effects in Digital Filters', MIT Lincoln Lab Report 468, November 1969.

4.6 Echard, J.D. and Boorstyn, R.R., 'Digital Filtering for Radar Signal Processing Applications', IEEE Trans. Audio and Electroacoustics, AU-20, No 1, March 1972, 42-52.

4.7 Rabiner, L.R. and Gold, B., 'Theory and Applications of Digital Signal Processing', Prentice-Hall, 1975.

4.8 Oppenheim, A.V. and Schafer, R.W., 'Digital Signal Processing', Prentice-Hall, 1975.

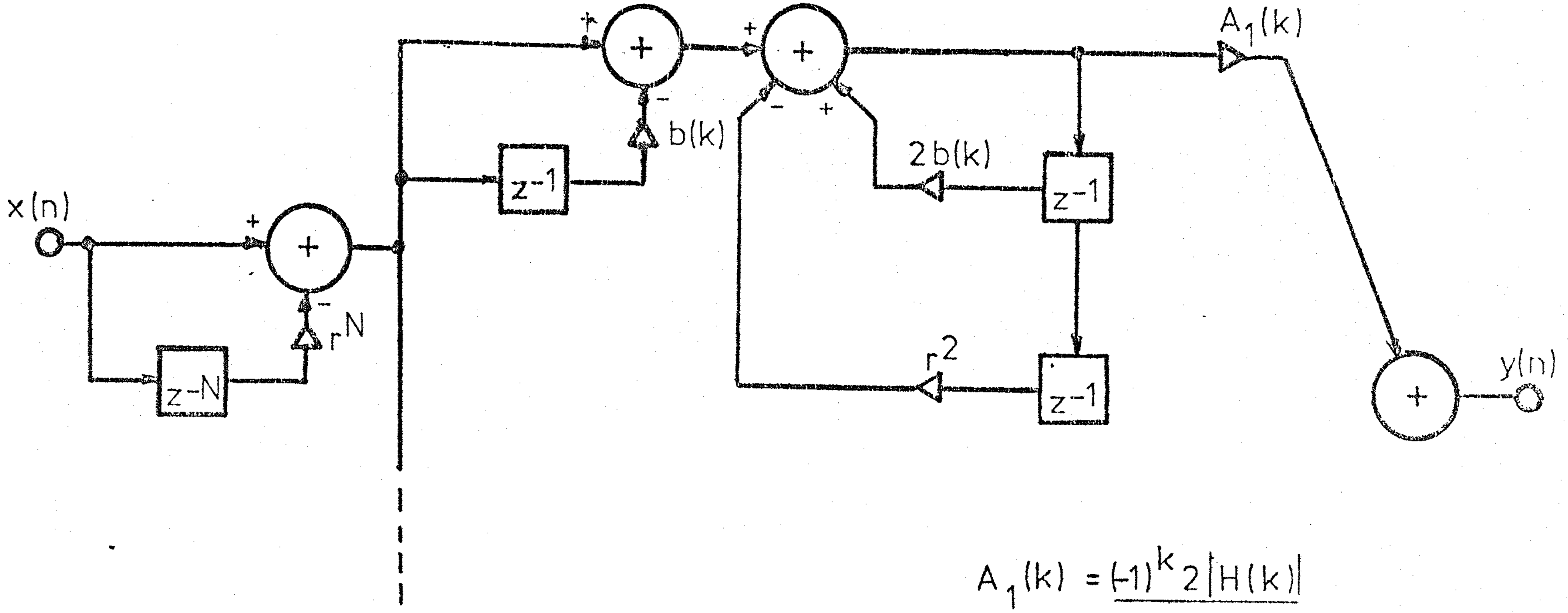
4.9 Bogner, R.E. and Constantimides, A.G., 'Introduction to Digital Filtering', Wiley, 1975.



a. Frequency Sampling

b. General Form of Realisation

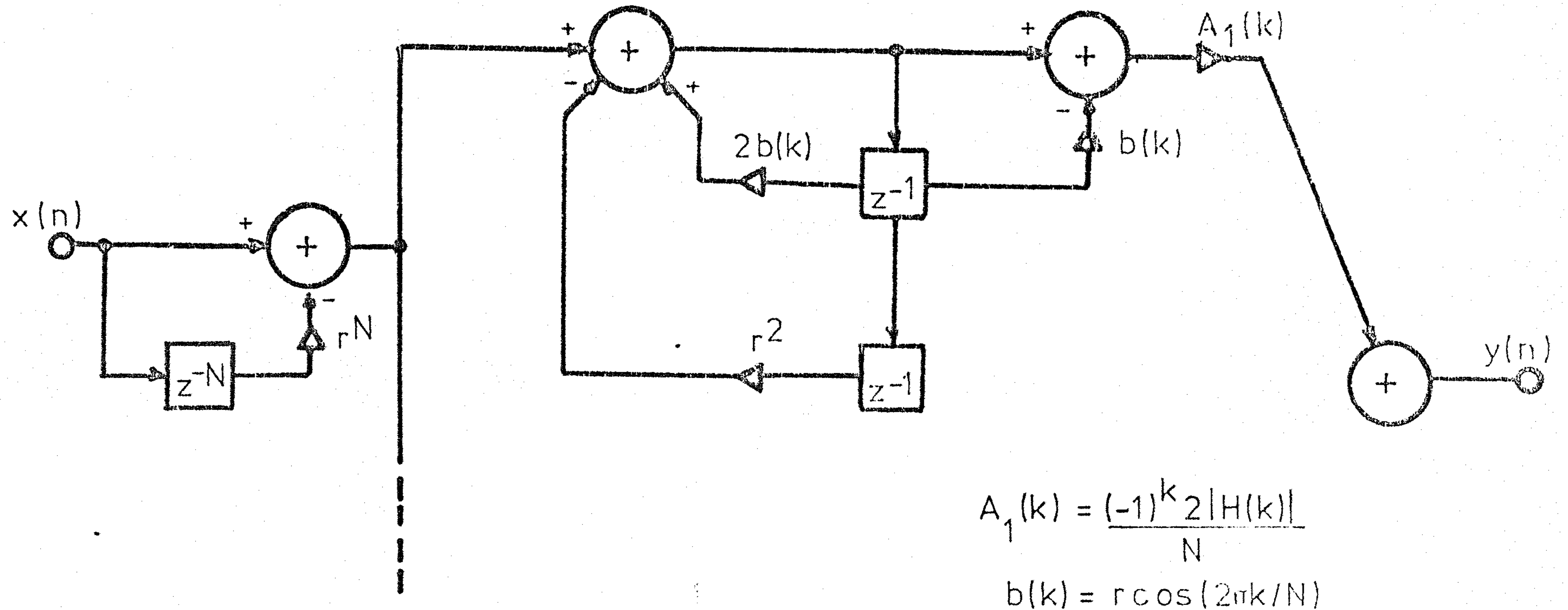
Figure 4.1 Frequency Sampling Filter



$$A_1(k) = \frac{(-1)^k 2|H(k)|}{N}$$

$$b(k) = r\cos(2\pi k/N)$$

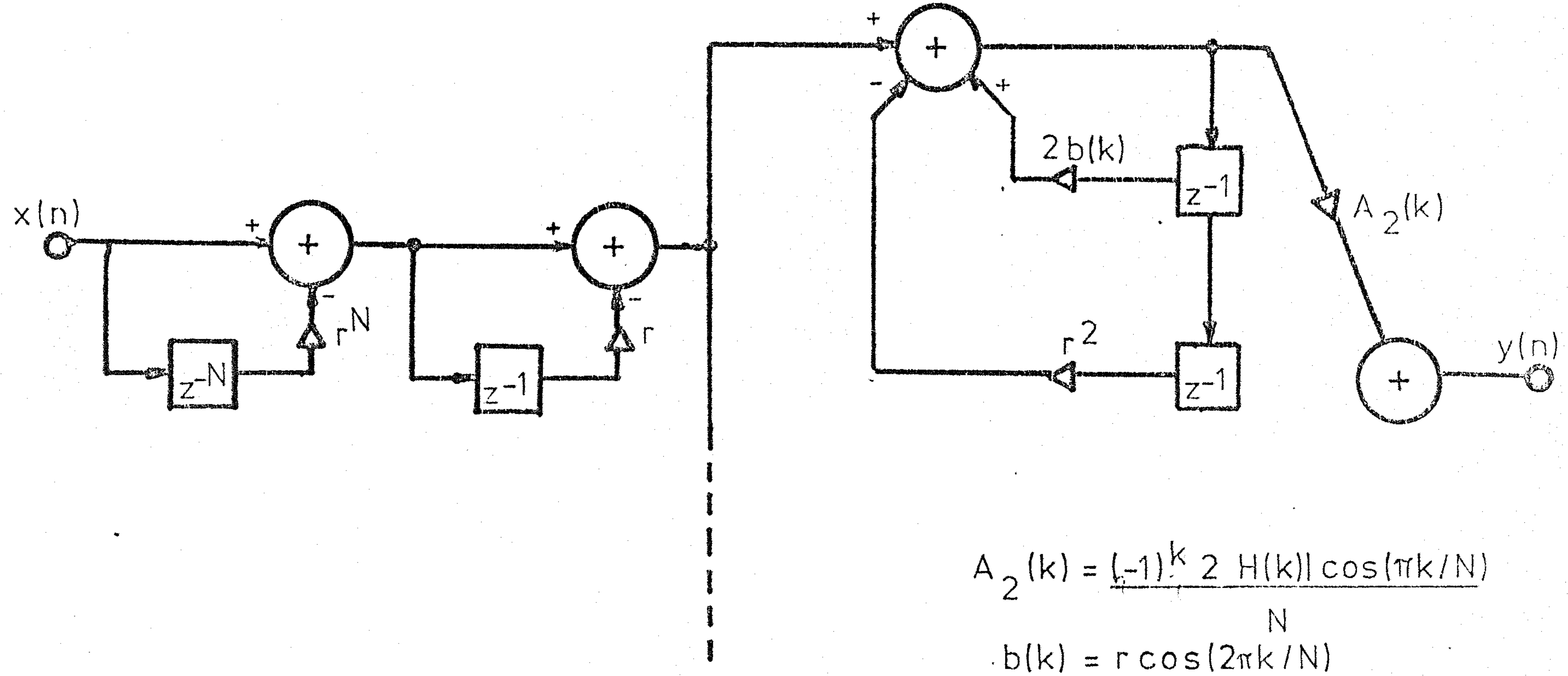
Figure 4.2 TYPE I Filter Realisation



$$A_1(k) = \frac{(-1)^k 2|H(k)|}{N}$$

$$b(k) = r\cos(2\pi k/N)$$

Figure 4.3 TYPE II Filter Realisation



$$A_2(k) = \frac{(-1)^k\, 2\; H(k)|\cos(\pi k/N)}{N}$$

$$b(k) = r\cos(2\pi k/N)$$

Figure 4.4   TYPE III Filter Realisation

CHAPTER FIVE

QUANTISATION NOISE IN FREQUENCY SAMPLING DIGITAL FILTERS

5.1 Introduction.

Several comprehensive surveys of the sources of error in digital filters caused by finite word lengths are available, References 5.1 to 5.8. There is general agreement that the major effects of quantisation upon filter performance fall in four main categories,

a. Quantisation of the filter coefficients in the difference equations defining the filter. Any departure from the design coefficients will cause deviations in the filter characteristics.

b. Roundoff or truncation errors which occur when multiplications are performed.

c. Input quantisation errors introduced by analogue-to-digital conversion.

d. Overflow, which places a bound upon the maximum input signal amplitude.

In addition, under certain circumstances, the phenomenon of limit cycling may appear; this is covered under b. above.

Coefficient quantisation can best be treated on the basis of sensitivity (Reference 5.7) and from such considerations it can be shown that several filter realisations exist which reduce the sensitivity of the filters to coefficient quantisation for those filters whose poles lie near to zero frequency and almost on the unit circle. In the case of bandpass filters, many of the advantages of such realisations are not of such great significance as the sensitivity is lower than that of low-pass filters of the same bandwidth. In the case of frequency sampling filters, it is necessary to determine by how much the poles may be moved inside the unit circle before significant changes in the filter characteristics are detected.

Roundoff errors demand close attention as they are dependent upon the choice of arithmetic, the word length and the filter realisation. Fixed point additions are errorless provided no overflow occurs, the sole source of roundoff error being found in the multipliers. In addition, in the case of fixed point arithmetic, the noise produced by rounding is uncorrelated with the sign of the signal. For these reasons, fixed point arithmetic will be assumed in the analyses which follow.

Input quantisation has been discussed extensively based on the assumption that it may be treated as white noise. This assumption has been shown to be valid in the majority of cases (Reference 5.2). Overflow presents problems but may be countered by the use of suitable scaling (Reference 5.8) or by the use of saturating adders.

The following sections present analyses of noise performance for the types of filter defined in Chapter 4 based upon the noise models and techniques available in the literature. Finally a best-fit method of noise measurement suitable for simulation measurements is proposed.

5.2 The Noise Model.

In common with other workers in the field, it is assumed that the multipliers in the filter can be represented by an ideal noiseless device together with an additive white noise source. Using linear systems theory, the output of a system having a transfer function $G(z)$, which is not necessarily the same as the transfer function between output and input $H(z)$, produced when a white noise $\varepsilon(n)$ is applied to the system input can be determined. It is assumed that $\varepsilon(n)$ does not exist for $n < 0$, that it has zero mean and variance σ_ε^2 given by

$$\sigma_\varepsilon^2 = 2^{-2t}/12 \qquad 5.1$$

where t is the word length. The variance of the filter output is then obtained by applying the convolution theorem which, provided $\varepsilon(n)$ is independent from sample to sample, is given by

$$\sigma_e^2 = \sigma_\varepsilon^2 \sum_{n=0}^{m} g^2(n) \qquad 5.2$$

Here, $g(n)$ is the impulse response of the system found by transforming the transfer function $G(z)$. Calculation of σ_e^2 can be simplified by assuming that the system is stable having all its poles within the unit circle. The righthand side of Equation 5.2 is then convergent and the steady-state variance of the system output is found by letting m tend to infinity, ie,

$$\sigma_e^2 \approx \sigma_\varepsilon^2 \sum_{n=0}^{\infty} g^2(n) \qquad 5.3$$

which may be evaluated by the relationship

$$\sum_{n=0}^{\infty} g^2(n) = \frac{1}{2\pi j} \oint G(z)G(z^{-1})z^{-1}dz \qquad 5.4$$

The integral in equation 5.4 appears in many applications in control and communication problems and may be solved most easily by a partial fraction expansion of $G(z)G(z^{-1})$ and the application of Cauchy's Residue Theorem.

In all the filter realisations to be considered, more than one noise source is present. On the assumption that each of the noise sources are uncorrelated with each other and if there are i noise sources, the total output noise variance σ_0^2 is given by

$$\sigma_0^2 = \sum_i \sigma_e^2(i) \qquad 5.5$$

With this noise model, it is possible to determine the contribution to the total noise at the output from all the noise sources in the system. It must be emphasised that the $G(z)$ of the system might not be the same for all noise sources and that their contributions will be weighted as shown in equation 5.3.

5.3 Noise Analysis of the Type I Filter.

The transfer function of a single Type I filter resonator element is

$$H_r(z) = \frac{1 - r\cos\theta\, z^{-1}}{1 - 2r\cos\theta\, z^{-1} + r^2z^{-2}} \qquad 5.6$$

where $\theta = 2\pi k/N$.

The impulse response of this resonator is

$$h_r(n) = r^n \cos n\theta \qquad 5.7$$

The resonator possesses a single zero on the real axis at $z = r\cos\theta$ and a complex conjugate pole pair at $z = r\exp(\pm j\theta)$. The resonant frequency is determined by the resonator angle θ and the sampling rate $1/T$ determines the resonator angle.

Figure 5.1 shows the realisation of the Type I filter including the noise sources in the filter. Inspection reveals that as far as the resonator is concerned, the noise outputs of the multipliers, $\varepsilon_2(n)$, $\varepsilon_3(n)$ and $\varepsilon_4(n)$, pass through the poles of the resonator ie through the recursive section of the resonator. Using equation 5.4 in which $G(z)$ is now

$$G(z) = \frac{1}{1 - 2r\cos\theta\, z^{-1} + r^2z^{-2}} \qquad 5.8$$

it can be shown that

$$\sum_{n=0}^{\infty} g^2(n) = \frac{1+r^2}{1-r^2} \cdot \frac{1}{r^4 + 1 - 2r^2 \cos 2\theta} \qquad 5.9$$

$$\approx 1/4\delta \sin^2 \theta \qquad 5.10$$

In order to avoid overflow within the resonator, it is essential to scale the input signal. This is effected by introducing an attenuator $1/\gamma$ at the input to the bank of resonators as shown. The attenuator reduces both signal and noise, the latter to such an extent that the roundoff noise from the comb filter may be ignored, leaving only the roundoff produced by the attenuator itself.

Because the attenuator has been introduced, it is necessary to multiply the output of the resonator by a factor γ in order to retain consistency. Combining this factor with the weighting factor $A_1(k)$, the modified weighting factor becomes

$$A_1'(k) = \gamma A_1(k) = \frac{(-1)^k \, 2|H(k)|\gamma}{N} \qquad 5.11$$

It is assumed that the noise component at the output contributed by the weighting factor is negligible.

There are therefore four main sources of noise in the realisation, namely the input noise passing through the resonators which contributes a component $\sigma_{0_1}^{\,2}$ to the output noise variance, and the noise from the resonator itself produced by the generators $\varepsilon_2(n)$, $\varepsilon_3(n)$ and $\varepsilon_4(n)$ passing through the poles of the resonator contributing a component $\sigma_{0_2}^{\,2}$ to the output noise.

Then using the noise model defined earlier, the noise component $\sigma_{0_1}^{\,2}$ is given by

$$\sigma_{0_1}^{\,2} = \frac{2^{-2t}}{12} \sum_{n=0}^{\infty} h_r^{\,2}(n) \sum_k \left[A_1'(k)\right]^2$$

$$= \frac{2^{-2t}}{12} \left[\frac{1}{1-r^2} \left\{ 1 - \frac{r^2 \sin^2 \theta (1+r^2)}{1 + r^4 - 2r^2 \cos 2\theta} \right\} \right] \sum_k \left[A_1'(k)\right]^2 \qquad 5.12$$

$$\approx \frac{2^{-2t}}{12} \; \frac{1}{4\delta} \sum_k \left[A_1'(k)\right]^2$$

The term h(n) is the impulse response of the filter defined by the transfer function H(z) derived in Chapter 4 for the Type I and Type II filters. It has been assumed that the resonators have a sufficiently narrow bandwidth to prevent any significant overlap between them.

The three noise sources within the resonator contribute the component $\sigma_{0_2}{}^2$ where

$$\sigma_{0_2}{}^2 = \frac{2^{-2t}}{12} \cdot 3 \sum_k \frac{\left[A_1'(k)\right]^2}{4\delta \sin^2\theta(k)} \tag{5.13}$$

Since each component has been assumed to be uncorrelated, the sum of the variances gives the variance of the total output noise $\sigma_{0_1}{}^2$

$$\sigma_{0_I}{}^2 = \frac{2^{-2t}}{12} \cdot \frac{1}{4\delta} \left\{ \sum_k \left[A_1'(k)\right]^2 + 3 \sum_k \frac{\left[A_1'(k)\right]^2}{\sin^2\theta(k)} \right\} \tag{5.14}$$

The noise variance at the output can be used as a measure of the noise power present at the output, but the more usual way of expressing noise performance is to formulate expressions for the output noise-to-signal ratio, NSR, for two forms of input, one a white noise, the other a sinusoid.

The noise is defined as having a uniform amplitude distribution between ± 1 with zero mean and variance given by

$$\sigma_{xn}{}^2 = 1/3 \tag{5.15}$$

This signal passes through the attenuator, through the filter appearing at the output with a variance

$$\sigma_{yn}{}^2 = \frac{1}{3\gamma} \sum_k \left[A_1'(k)\right]^2 \tag{5.16}$$

On the other hand, the sinusoid has zero mean and a variance given by

$$\sigma_{xs}{}^2 = 1/2 \tag{5.17}$$

yielding an output signal of variance

$$\sigma_{ys}{}^2 = \frac{1}{2\gamma} \sum_k \left[A_1'(k)\right]^2 \tag{5.18}$$

The noise-to-signal ratios for both form of input can now be formulated.

$$(NSR_n)_I = \frac{2^{-2t}}{12} \cdot \frac{3\gamma}{4\delta} \left[1 + 3 \frac{\sum\limits_k \frac{\left[A_1'(k)\right]^2}{\sin^2\theta(k)}}{\sum\limits_k \left[A_1'(k)\right]^2} \right]$$

$$= \frac{2^{-2t}}{12} \cdot \frac{3\gamma}{4\delta} \left[1 + 3 \frac{\sum\limits_k \left[\frac{H(k)}{\sin\theta(k)}\right]^2}{\sum\limits_k \left[H(k)\right]^2} \right] \qquad 5.19$$

and for the sinusoidal input,

$$(NSR_s)_I = \frac{2^{-2t}}{12} \cdot \frac{\gamma}{2\delta} \left[1 + 3 \frac{\sum\limits_k \left[\frac{H(k)}{\sin\theta(k)}\right]^2}{\sum\limits_k \left[H(k)\right]^2} \right] \qquad 5.20$$

5.4 Noise Analysis of the Type II Filter.

In the canonic realisation shown in Figure 5.2, the noise passes first through the poles of the resonator and then through the zero, which is the reverse of the situation in the Type I filter.

Using the same procedure as in the Type I filter and making the same assumptions, it can be seen that the three significant noise sources ie the attenuator noise, $\varepsilon_2(n)$ and $\varepsilon_3(n)$ pass through the resonator as a whole so that the variance of the output noise for the Type II filter is

$$\sigma_{0_{II}}^2 = \frac{2^{-2t}}{12} \cdot \frac{3}{4\delta} \sum_k \left[A_1'(k)\right]^2 \qquad 5.21$$

since

$$\sum_{n=0} h_r^2(n) \approx 1/4\delta \qquad 5.22$$

The noise-to-signal ratios for noise input and sinusoidal input are respectively

$$(NSR_n)_{II} = \frac{2^{-2t}}{12} \cdot \frac{9\gamma}{4\delta} \tag{5.23}$$

$$(NSR_s)_{II} = \frac{2^{-2t}}{12} \cdot \frac{3\gamma}{2\delta} \tag{5.24}$$

5.5 Noise Analysis of the Type III Filter.

Examination of the realisation shown in Figure 5.3 shows that the noise from the attenuator and from the internally generated noise sources pass only through the poles of the resonator because the common zero has been removed from the resonator bank. Using the same analysis method as before, the output noise variance is

$$\sigma_{0_{III}}^{2} = \frac{2^{-2t}}{12} \cdot \frac{3}{4\delta} \sum_k \frac{\left[A_2'(k)\right]^2}{\sin^2\theta(k)} \tag{5.25}$$

where

$$A_2'(k) = \frac{\gamma(-1)^k\, 2|H(k)|\cos\left[\theta(k)/2\right]}{N} \tag{5.26}$$

whence

$$(NSR_n)_{III} = \frac{2^{-2t}}{12} \cdot \frac{3\gamma}{4\delta} \cdot 3\, \frac{\sum_k \left[\dfrac{H(k)\ \cos\left[\theta(k)/2\right]}{2\ \sin\left[\theta(k)/2\right]}\right]^2}{\sum_k \left[H(k)\ \cos\left[\theta(k)/2\right]\right]^2} \tag{5.27}$$

$$(NSR_s)_{III} = 2(NSR_n)_{III}/3 \tag{5.28}$$

5.6 Double Precision Accumulator.

In the foregoing analyses, it has been assumed that the product of each multiplication is rounded before being passed onto the accumulator. An alternative method is to accumulate the products in a double precision register, the output of which is then reduced to the specified word length. Intuitively, this should improve the noise performance of the filter because the noise sources within the resonator have been reduced.

In the case of the Type I filter, the significant source of noise is at the output of the adder as shown in Figure 5.4. Then

$$\sigma_{0Ia}{}^{2} = \frac{2^{-2t}}{12} \cdot \frac{1}{4\delta} \sum_{k} \frac{\left[A_1'(k)\right]}{\sin^2\theta(k)} \qquad 5.29$$

whence

$$(NSR_n)_{Ia} = \frac{2^{-2t}}{12} \cdot \frac{3\gamma}{4\delta} \frac{\sum\limits_{k} \left[\dfrac{H(k)}{\sin\ \theta(k)}\right]}{\sum\limits_{k} \left[H(k)\right]^2} \qquad 5.30$$

$$(NSR_s)_{Ia} = \frac{2}{3}\left[(NSR_n)_{Ia}\right] \qquad 5.31$$

In the Type II filter shown in Figure 5.5, the noise output from the double precision accumulator passes through the poles and through the zero of the resonator so that

$$\sigma_{0IIa}{}^{2} = \frac{2^{-2t}}{12} \cdot \frac{1}{4\delta} \sum_{k} \left[A_1'(k)\right]^2 \qquad 5.32$$

and

$$(NSR_n)_{IIa} = \frac{2^{-2t}}{12} \cdot \frac{3\gamma}{4\delta} = \frac{3}{2}\left[(NSR_s)_{IIa}\right] \qquad 5.33$$

Finally, Figure 5.6 shows the double precision accumulator realisation of the Type III filter.

$$\sigma_{0IIIa}^{2} = \frac{2^{-2t}}{12} \cdot \frac{1}{4\delta} \sum_{k} \left[A_2'(k)\right]^2 \qquad 5.34$$

whence

$$(NSR_n)_{IIIa} = \frac{2^{-t}}{12} \cdot \frac{3\gamma}{4\delta} \frac{\sum_k \left[\frac{H(k)}{2 \tan \left[\theta(k)/2\right]}\right]^2}{\sum_k \left[H(k) \cos \left[\theta(k)/2\right]\right]^2}$$

$$= \frac{3}{2}\left[(NSR_s)_{IIIa}\right] \qquad 5.35$$

Having formulated expressions for the noise output and output noise-to-signal ratios, the problem arises of how to measure the noise output of a particular filter without resorting to cumbersome techniques. To this end, a best-fit method of filter performance measurement has been devised.

5.7 Best-Fit Filter Performance Measurement.

Gold and Rader (Reference 5.5) have proposed a method of measuring quantisation noise in a digital filter using the system shown in Figure 5.7. In a simulation, this method is quite satisfactory if only noise is to be measured. It does not, however, given any measure of the gain and phase of the filter since this information contained in the output of the filter is virtually destroyed in the subtraction process. On a broader basis, therefore, the proposed method of measurement is inefficient since for two passes of the same data, a single result only is yielded.

The best-fit method can be used with deterministic signals, or more precisely, with sinusoid inputs by assuming that the filter output y(n) contains a d.c. component, a desired signal component phase shifted by the filter together with other components which may be treated as noise.

If the output noise is defined as E_n, then

$$E_n = y-A_0-A_1\cos(\omega t+\phi) \qquad 5.36$$

where y is the filter output, A_0 is the d.c. component, A_1 is the filter gain, ϕ in the filter phase and ω in the frequency of measurement. Equation 5.36 has been written assuming the output is continuous for the sake of clarity. In practice, the digital filter is allowed to reach a steady-state, the time for this being discussed in Chapter 6, when M samples of the output $y(n)$ are stored. The root mean square value of the output noise is then calculated from

$$(E_n)_{rms} = \left[\frac{1}{M}\sum_{n=1}^{M}\left[E(n)\right]^2\right]^{\frac{1}{2}} \qquad 5.37$$

where $E(n)$ is the sampled noise value.
Rearranging equation 5.37

$$M\left[(E_n)_{rms}\right]^2 = \sum_{n=1}^{M}\left[E(n)\right]^2$$

$$= E$$

$$= \sum_{n=1}^{M}\left\{y(n)-A_0-A_1\ \cos\left[\omega t(n)+\phi\right]\right\}^2 \qquad 5.38$$

Equation 5.38 may be expanded by making the following substitutions.

$$x = A_1\cos\phi,\ y = A_1\sin\phi,\ z = y(n),$$
$$c = \cos\omega t(n),\ s = -\sin\omega t(n)$$

whence $$E = M\left[(E_n)_{rms}\right]^2 = \sum_{n=1}^{M}(z-A_0-xc-ys)^2 \qquad 5.39$$

Now if the correct values of A_0 and $A_1\ \cos\left[\omega t(n)+\phi\right]$ have been subtracted from the output signal z, the noise E will be a minimum so that

$$\frac{\partial E}{\partial A_0} = \frac{\partial E}{\partial x} = \frac{\partial E}{\partial y} = 0 \qquad 5.40$$

Expanding equation 5.39.

$$
\begin{aligned}
E = \sum z^2 + MA_0{}^2 + x^2\sum c^2 + y^2\sum s^2 \\
+ 2\Big[A_0x\sum c + A_0y\sum s + xy\sum cs \\
- A_0\sum z - x\sum cz - y\sum sz\Big]
\end{aligned}
\qquad 5.41
$$

The limits of summation have been omitted for the sake of convenience. Applying the constraints defined in equation 5.40

$$
\left.
\begin{aligned}
2MA_0 + 2x\sum c + 2y\sum s - 2\sum z = 0 \\
2x\sum c^2 + 2A_0\sum c + 2y\sum cs - 2\sum cz = 0 \\
2y\sum s^2 + 2A_0\sum s + 2x\sum cs - 2\sum sz = 0
\end{aligned}
\right\}
\qquad 5.42
$$

whence

$$
A_0 = \frac{1}{M}\Big[\sum z - x\sum c - y\sum s\Big] \qquad 5.43
$$

To simplify the expressions for gain and phase, let

$$
\tilde{c} = M\sum c^2 - \sum c\sum c, \quad \tilde{s} = M\sum s^2 - \sum s\sum s,
$$

$$
\widetilde{pq} = M\sum pq - \sum p\sum q
$$

whence

$$
\begin{aligned}
x &= \frac{1}{\tilde{c}}\Big[\widetilde{cz} - y.\widetilde{cs}\Big] \\
y &= \frac{\tilde{c}.\widetilde{sz} - \widetilde{cs}.\widetilde{cz}}{\tilde{c}.\tilde{s} - \widetilde{cs}.\widetilde{cs}}
\end{aligned}
\qquad 5.44
$$

The gain and phase of the filter may now be calculated from equation 5.44,

$$
\left.
\begin{aligned}
\text{Gain} \quad A_1 &= (x^2+y^2)^{\frac{1}{2}} \\
\text{Phase} \quad \phi &= \tan^{-1}(y/x)
\end{aligned}
\right\}
\qquad 5.45
$$

and the noise can be calculated from equation 5.41

5.8 Discussion.

Examination of the expressions derived for the output noise-to-signal ratios of the three types of filter, both for rounding after the multiplier and for rounding after the adder, equations 5.19 to 5.31, show that in the Type I filter and the Type III filter there is a distinct frequency dependence. If the operating frequency lies close to one quarter of the sampling frequency, the noise performance of the three filters is much the same but as the frequencies of operation more further away from the quarter sampling frequency, large differences are to be expected.

It is difficult to draw any general quantitative conclusions regarding the merits or demerits of particular realisations without resorting to specific filter examples, preferably chosen to cover the range from just above zero frequency to just below the half sampling frequency. Figure 5.8 shows the frequency specifications of three such filters which have not been chosen to satisfy the doppler filter specification shown in Figure 1.1 but to deliberately place the centre frequency of the extreme filters so that errors will be introduced due to coefficient quantisation. Figure 2.6a shows that at about one tenth (or four tenths) of the sampling frequency, the coefficient quantisation is becoming less severe.

It will be shown in Chapter 6 that a suitable value for the constant term in the equations $3\gamma/48\delta$ is 512 and using this as a basis for calculation together with the parameters shown in Figure 5.8, a comparison may be made which can then be tested by measurement using the best fit method. Figure 5.9 presents a summary of the theoretical noise-to-signal ratios for each type of filter.

These figures show that the rounded double precision accumulator has an advantage over the rounded multiplier realisation of about 5 dB and that the noise-to-signal ratio pattern for each rounding method is identical. Individually, the Type I filter gives the poorest result while the Type III filter has the advantage over the Type II filter at mid-frequency range and above.

It has been assumed in the calculation of the figures that no overflow occurs but this might not be the case and will be investigated in Chapter 6. As already mentioned, if overflow does occur, the filter becomes less satisfactory as non-linearities will be introduced.

However, if the frequency of overflow is very low, this might not be too great a disadvantage although it will depend upon the particular filter application.

Turning now to the proposed best-fit method of filter performance, its major advantage lies in the economy of computation. It is prone to a measurement confusion particularly at the resonator frequencies for the resonators have very narrow bandwidths which make the noise component appear like a signal component or vice versa. This is not a serious disadvantage particularly when register lengths are long.

A much more important limitation is that the input signal must be deterministic, or more exactly, sinusoidal because a phase shifted version of the input is necessary for the computations of gain phase and noise.

The validity of the analyses presented in this Chapter will be tested by measurement via computer simulations using programs described in Chapter 7.

5.9 Summary.

Noise-to-signal analyses have been presented for three types of filter realisation and for two methods of rounding. They show that there are distinct differences in noise performance which are not in themselves conclusive until an overflow investigation has been carried out. Overflow may of course be avoided completely but this requires that the filter is always started from zero initial conditions and that any scaling used is such that the largest possible signal is smaller in magnitude than the overflow level.

5.10 References.

5.1 Weinstein, C.J., 'Quantisation Effects in Digital Filters', MIT Lincoln Lab. Report 468, November 1969.

5.2 Hadjifotiou, A., 'Design and Simulation of Digital Filters for MTI and Pulse Doppler Radars', PhD Thesis, University of Southampton, October 1974.

5.3 Liu, B., 'Effect of Finite Word Length on the Accuracy of Digital Filters - A Review', IEEE Trans. Circuit Theory, CT-18, No 6, November 1971, 670-677.

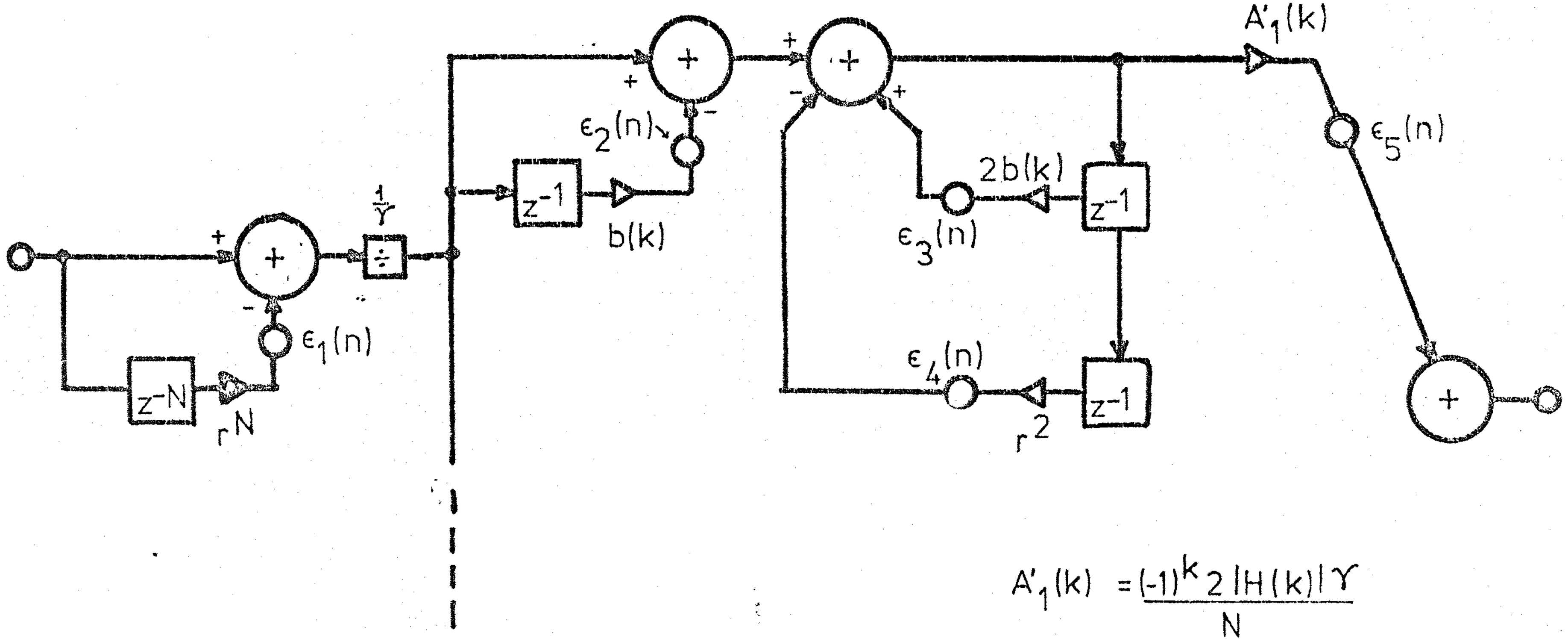
5.4 Oppenheim, A.V., and Weinstein, C.J., 'Effect of Finite Register Length in Digital Filters and Fast Fourier Transforms', Proc. IEEE, Vol 60, No 8, August 1972, 957-976.

5.5 Gold, B. and Rader, C.M., 'Digital Processing of Signals' McGraw-Hill, 1969.

5.6 Claasen, T.A.C.M., Macklenbräuker, W.F.G., and Peek, J.B.H., 'Effects of Quantisation and Overflow in Recursive Digital Filters', IEEE Trans. Acoustics, Speech and Signal Processing ASSP-24, No 6, December 1976, 517-529.

5.7 Agarwal, R.C., 'On the Realisation of Digital Filters' PhD Thesis, Rice University, 1974.

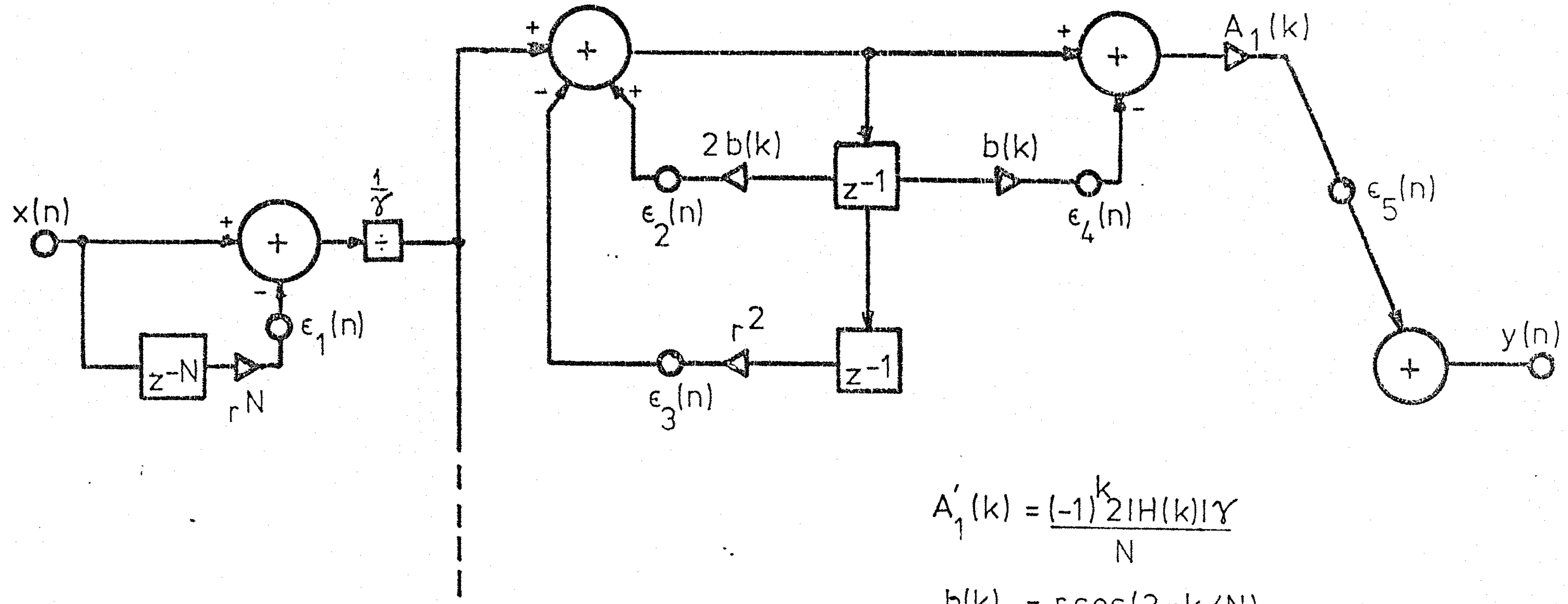
5.8 Jackson, L.B., 'Roundoff Noise Analysis for Fixed Point Digital Filters realised in Cascade or Parallel Form', IEEE Trans. Audio and Electroacoustics, AU-18, No 2, June 1970, 107-122.



$$A'_1(k) = \frac{(-1)^k 2\,|H(k)|\,\Upsilon}{N}$$

$$b(k) = r\cos(2\pi k/N)$$

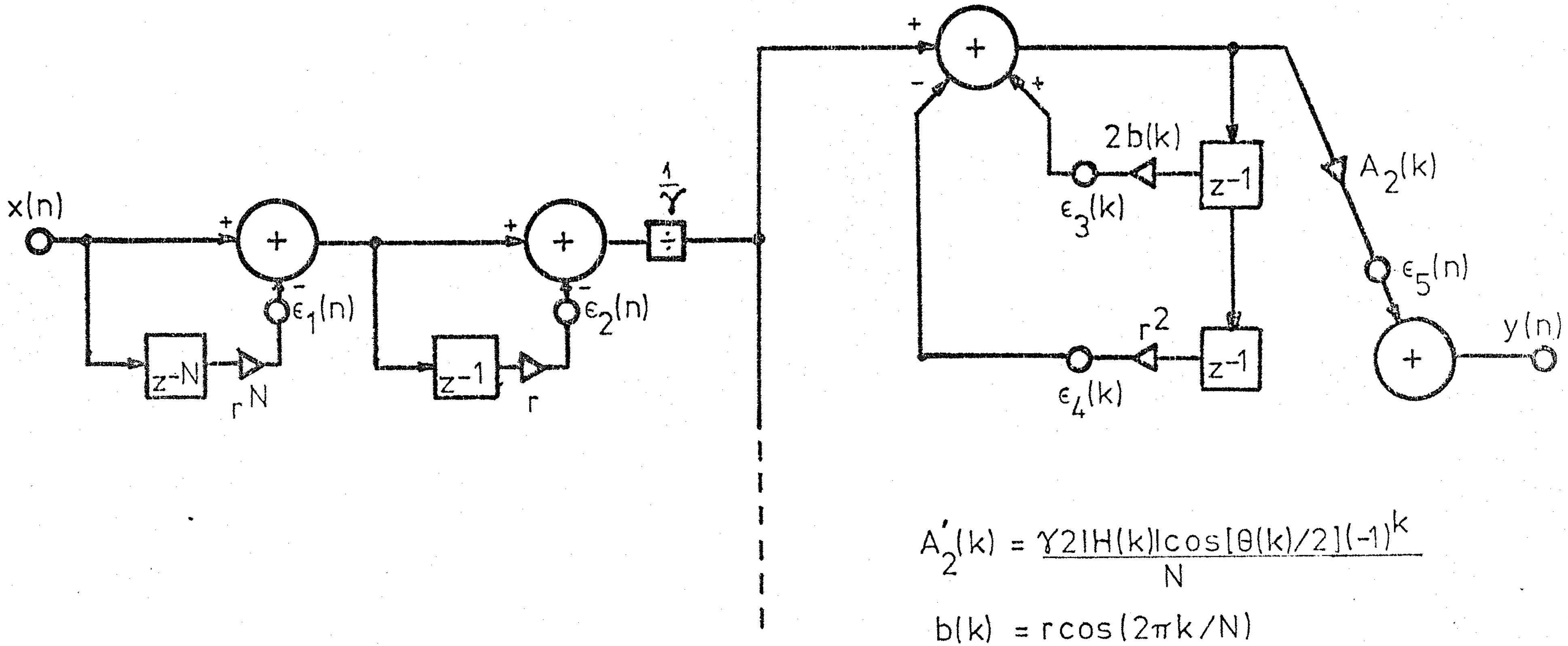
Figure 5.1 TYPE I Filter Noise Sources - Rounded Multiplier



$$A_1'(k) = \frac{(-1)^k 2|H(k)|\gamma}{N}$$

$$b(k) = r\cos(2\pi k/N)$$

Figure 5.2 TYPE II Filter Noise Sources -Rounded Multiplier



$$A_2'(k) = \frac{\gamma 2|H(k)|\cos[\theta(k)/2](-1)^k}{N}$$

$$b(k) = r\cos(2\pi k/N)$$

Figure 53 TYPE III Filter Noise Sources - Rounded Multiplier

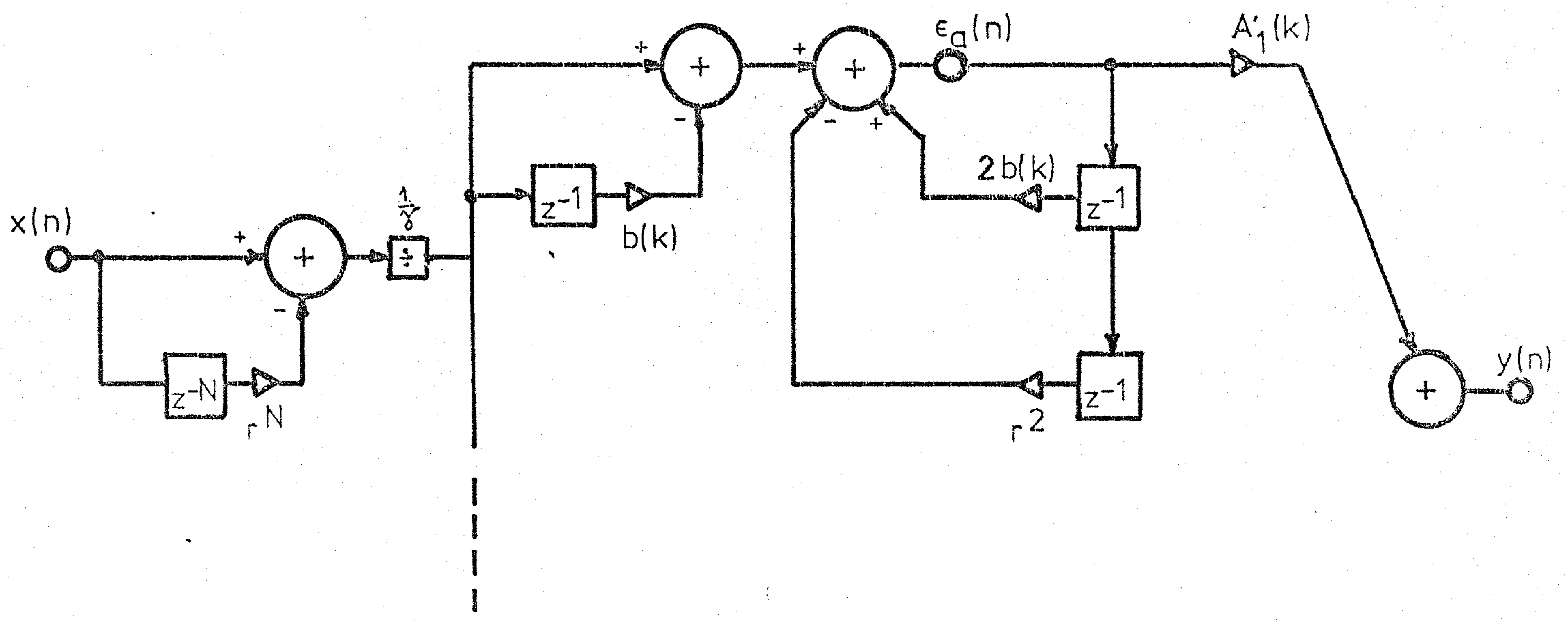


Figure 5.4 TYPE I Filter Noise Sources - Rounded Adder

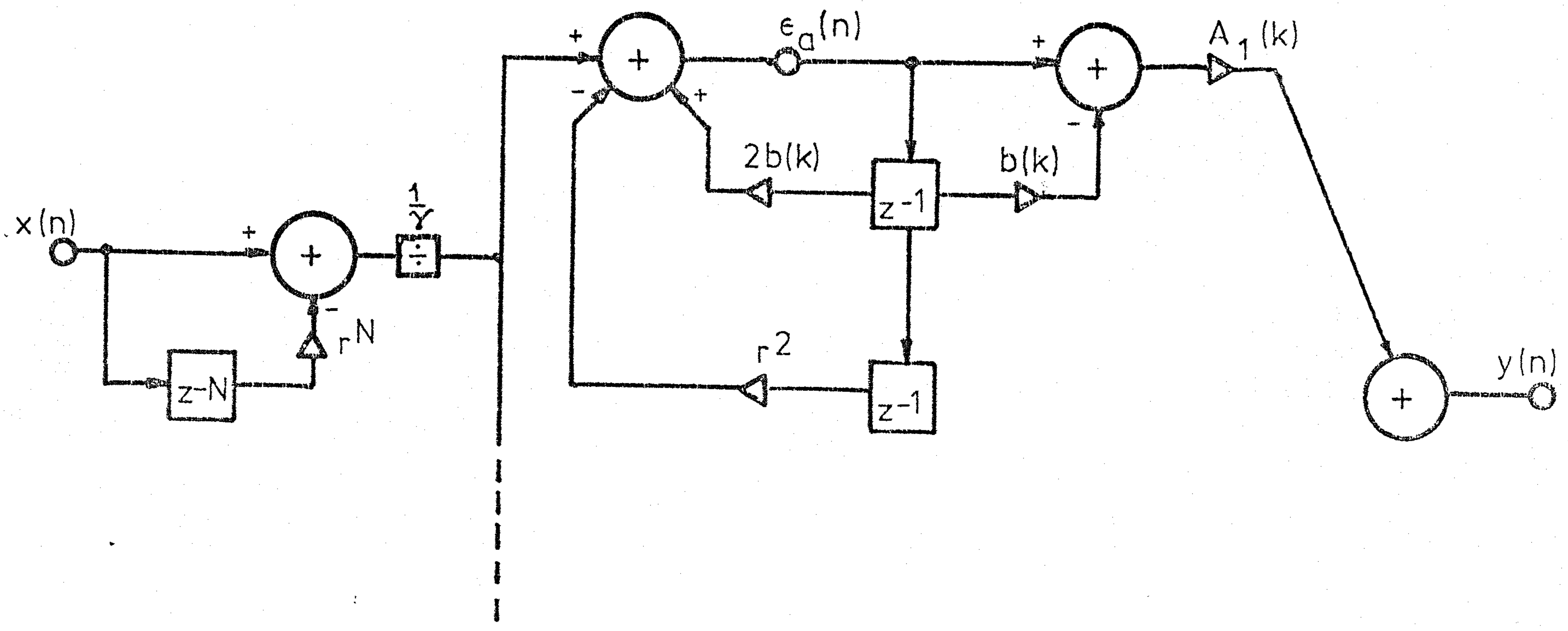


Figure 5.5 TYPE II Filter Noise Sources - Rounded Adder

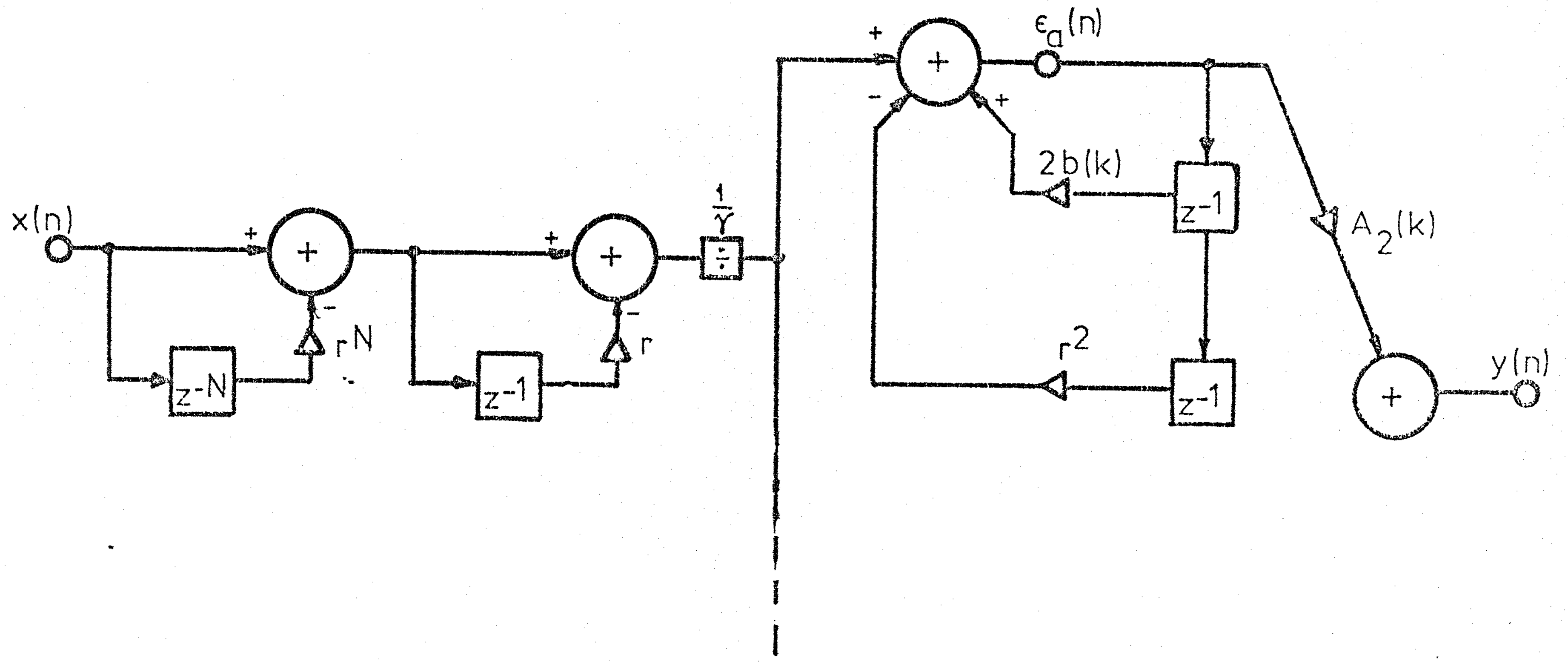


Figure 5.6 TYPE III Filter Noise Sources - Rounded Adder

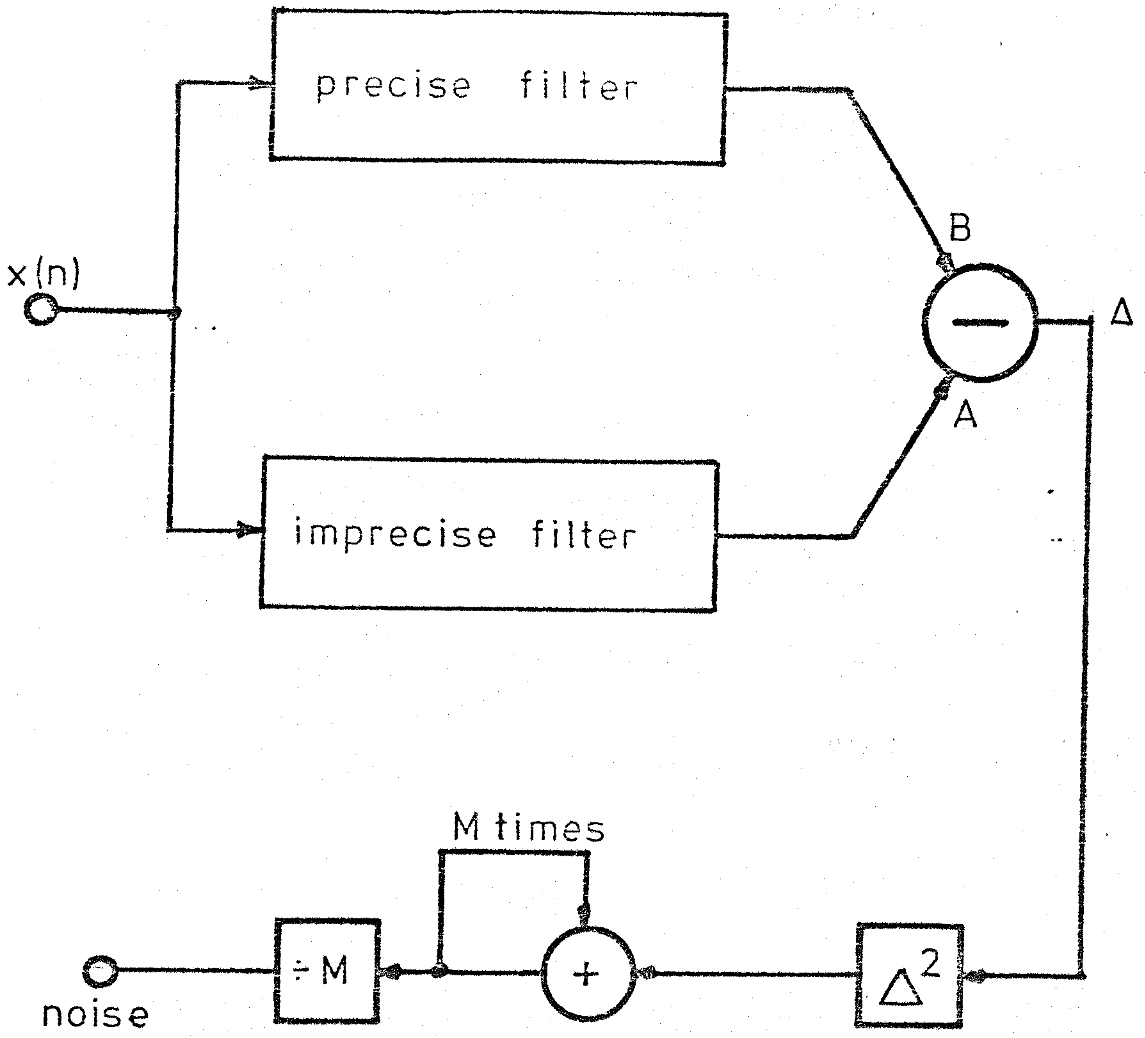


Figure 5.7 Two-pass Noise Measurement

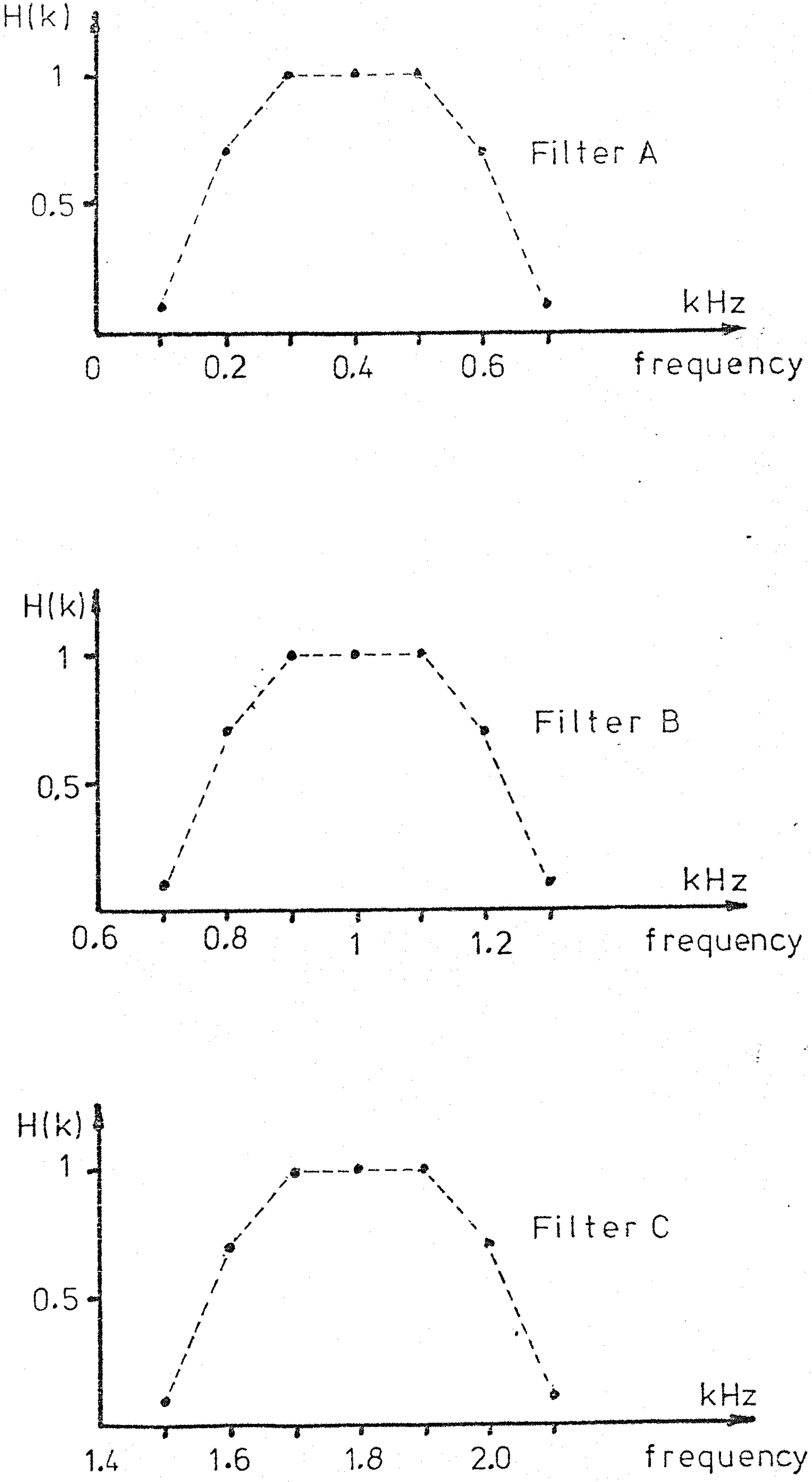


Figure 5.8 Filter Specifications

Signal	TYPE I			TYPE II			TYPE III		
	A	B	C	A	B	C	A	B	C
Noise	-56.5	-62.6	-58.8	-64.1	-64.1	-64.1	-56.8	-65.9	-69.5
Sinusoid	-58.2	-64.4	-60.6	-65.9	-65.9	-65.9	-58.6	-67.6	-71.3

a. Rounded Multipliers (16-bit word)

Signal	TYPE I			TYPE II			TYPE III		
	A	B	C	A	B	C	A	B	C
Noise	-61.5	-68.6	-64.0	-68.9	-68.9	-68.9	-61.5	-70.6	-74.3
Sinusoid	-63.3	-70.4	-65.8	-70.1	-70.1	-70.1	-63.3	-72.4	-76.1

b. Rounded Accumulators (16-bit word)

Figure 5.9 Theoretical Noise-to-signal Ratios in dB

CHAPTER SIX

FILTER SIMULATION

6.1 Introduction.

In Chapter 2 it was stated that the frequency sampling technique of approximation to arbitrary frequency responses does not lend itself readily to hard and fast design rules. In addition, it was shown in Chapter 5 that general analyses of noise performance are possible but that comparisons between realisations are difficult to make without resorting to a particular filter specification. This chapter describes the procedure employed to determine the fixed parameters of the filter simulation given a specification as a basis for design.

The simulation can be conveniently divided into two phases:

Phase 1 - the determination of the minimum number of resonators required to meet the specification, the minimum coefficient word length and the degree of resonator damping δ. These parameters may be found by direct computation of the transfer function in the frequency domain.

Phase 2 - investigation of the noise performance, the gain and phase responses performed via a simulation in the time domain.

Phase 1 was carried out using a programmable desk-top calculator having a word length of 24 bits. This enabled coefficient investigations to be made on a filter that was effectively 'precise' in terms of the arithmetic word length. The program is given in Appendix B together with instructions for its use. This method is considered to be more efficient than that proposed by Weinstein (Reference 6.1) which entails a computation of the impulse response followed by a discrete Fourier Transformation to yield the frequency response.

Phase 2 used a general purpose computer with batch facilities. The suite of programs developed is described in Chapter 7.

6.2 Determination of the Minimum Number of Resonators.

The minimum number of resonators required to satisfy the given specification depends upon the number of frequency samples N, the maximum allowable sidelobe amplitude and the steepness of the transition band.

From the results given by Rabiner (Reference 6.2), it is evident that those parameters are themselves interdependent and that to achieve a particular percentage bandwidth, the number of resonators increases as N increases. Chapter 5 has shown that an increase in the number of resonators can lead to an increase in the noise-to-signal ratio and will inevitably lead to a lower computation speed. It is therefore essential to choose as low a value of N as possible bearing in mind that a low value of N increases the frequency spacing between samples leading to unacceptably high ripple in the passband.

In the case of non-optimum filters, it is necessary to experiment to determine the minimum number of resonators. Figure 6.1 shows frequency responses of the filter realised with 3, 5 and 7 resonators, each with a frequency spacing of 100 Hz. The 3-resonator realisation is patently inadequate while the 5-resonator realisation meets the bandwidth requirement but not the sidelobe constraint ie less than -40 dB in the stopband. The 7-resonator realisation meets the specification in all respects.

6.3 Coefficient Wordlength.

Coefficient quantisation causes the pole pair defined by $r \exp(\pm j\theta)$ to deviate in the z-plane from the corresponding zeros of the comb filter. Oppenheim (Reference 6.3) and Agarwal (Reference 6.4) have considered this problem from the point of view of pole sensitivity to coefficient errors. Whatever the criteria used, it is essential that the coefficient word length be large enough to minimise errors in the filter response while at the same time maintaining an economical implementation.

Figure 6.2 shows filter responses for coefficient word lengths of 4, 8 and 12 bits. The 4-bit response is of little value and can be discarded. The 8-bit response is reasonably satisfactory except for a ripple in the passband and could be used. There is however another factor to be considered; namely the bandwidth of the resonators defined by the damping factor. The damping factor δ is of the order of 2^{-c} where c is the coefficient word length. For a coefficient of 8-bits, $\delta = 2^{-8}$ which gives a bandwidth of about 6 Hz at a sampling frequency of 4600 Hz. Such a bandwidth could begin to invalidate the assumption in the noise analysis that there should be no resonator response overlap. The 12-bit coefficient is seen to give a satisfactory response.

6.4 Resonator Damping.

In Chapter 5 it was shown that the noise performance of the filter depends to a great extent upon the resonator damping, that is upon δ, which also determines the gain and bandwidth of the resonator. As a result, the problem consists in striking a compromise between a value of δ small enough to satisfy the high gain high-Q requirement of the resonators and a value which is low enough to ensure a satisfactory noise performance. The effect of δ upon the noise performance can be virtually eliminated by choosing a large arithmetic word length, the decision on a value for δ then being made by observing the effect of changing δ upon the frequency response. Ripple and other deviations from the desired response will be observed if δ is made too small. Several tests were made using a coefficient word length of 12 bits, with no restriction being placed upon the arithmetic word length except by the calculator itself. A value of $\delta = 2^{-9}$ was chosen as the optimum value which agrees with the figure given by Weinstein in Reference 6.1.

6.5 Multiplier Word Length.

The choice is made primarily upon the basis of acceptable output noise as described and analysed in Chapter 5. Reference 6.5 quotes a design formula to determine the minimum arithmetic word length t in terms of the roundoff noise figure F in dB and the noise gain NG defined in Equation 5.4.

$$t \approx (0.8 + \frac{F}{6}) + \frac{10}{6}\log(NG) \qquad 6.1$$

Now, in the case of the resonator from which the major sources of noise are derived, the noise gain NG can be approximated to $1/2\delta = 256$. Rewriting Equation 6.1 gives

$$t \approx \frac{F}{6} + 5 \qquad 6.2$$

If a noise figure of 60 dB is required, t must be at least 15 bits. This shows a very close agreement with the figures given in Chapter 5.

Inspection of Equation 6.2 shows that an increase of 4-bits in the arithmetic word length yields an improvement of 24 dB in the noise figure.

6.6 Overflow, Maximum Coefficient Significance and Scaling.

In a filter employing fixed point arithmetic, the signals are assumed to be signed fractions. Jackson (Reference 6.6) has shown that in order to avoid overflow, a scaling factor must be applied at the input to the bank of digital resonators, even though such a strategy will have a deleterious effect upon the dynamic range of the filter. The output of each resonator in the frequency sampling technique is weighted and in order to maintain the validity of the transfer function, the weighting coefficient must be increased as described in Chapter 5.

Weinstein (Reference 6.1) suggests a scaling factor of $1/N$ but if N is large, the input to the resonator bank would be severely attenuated. In addition, if N is not an integer power of 2, division is accompanied by error. If N is an integer power of 2, right shifting may be employed so dispensing with the need for an additional multiplier. But, right shifting reduces the signal word length which argues for a closer examination of the overflow conditions in particular realisations.

The scaling factor should also be such that the modified weighting coefficient, $A'(k)$, does not itself overflow.

In the Type I and Type II realisations, the weighting coefficient is defined by

$$A_1(k) = (-1)^k \, 2|H(k)|/N \qquad 6.3.$$

and in the Type III realisation

$$A_2(k) = (-1)^k \, 2|H(k)|\cos(\pi k/N)/N \qquad 6.4$$

In both cases, if the weighting coefficients are to remain as fractions, the scaling factor must not be less than $2/N$.

Consider first the Type III realisation. The comb filter common zero combination has a maximum gain of 4 and if overflow is to be avoided at the input to the resonator bank, the input signal, $x(n)$, must be reduced by a factor of 4, effected by a 2-bit right shift. The gain of the comb filter/common zero combination and a single resonator is then given by

$$\frac{N}{8\cos(\pi k/N)} \qquad 6.5$$

which exhibits a frequency dependence best investigated by examining signal levels at particular nodes within the resonator as shown in Figure 6.5.

In order to compensate for the division of the input signal by a factor of 4, the output of the filter must be increased by a factor of 4 so that at a frequency of k/NT

$$|4.b_D.A_2(k)| = |H(k)| \qquad 6.6$$

where b_D is the signal value at point D in Figure 6.5.

Now

$$b_D = \frac{N}{8\cos(\pi k/N)} \qquad 6.7$$

which confirms Equation 6.5 so that the signal value at point E is

$$b_E = \frac{2rN}{8}.\frac{\cos(2\pi k/N)}{\cos(\pi k/N)} \qquad 6.8$$

Point E is a much more critical point since it is here that overflow is most likely to occur. Choosing a scaling factor of 2/N, b_E must be not greater than N/2 ie.

$$\frac{N}{2} \geqslant \frac{2rN}{8}\frac{\cos(2\pi k/N)}{\cos(\pi k/N)} \qquad 6.9$$

whence 6.10

$$\left(\frac{k}{N}\right)_{max} = 0.38$$

This limits the upper working frequency of the Type III realisation to 0.38 × (sampling frequency); in the particular filter chosen as a basis of comparison, the upper frequency limit for no overflow is 1748 Hz.

In the case of the Type I realisation shown in Figure 6.3

$$|2.b_F.A_1(k)| = |H(k)| \qquad 6.11$$

The factor of 2 in Equation 6.11 derives from the maximum comb filter gain of 2. The signal value at point F is

$$b_F = N/4 \qquad 6.12$$

and at point G is

$$b_G = \frac{2rN}{4} \cos(2\pi k/N) \qquad 6.13$$

whence

$$\left(\frac{k}{N}\right)_{max} = 0.5 \qquad 6.14$$

Thus the Type I realisation will not overflow up to the half sampling frequency.

A similar analysis for the Type II realisation shown in Figure 6.4 shows that

$$b_J = N/4 \qquad 6.15$$

$$b_L = \frac{N}{4 \sin(2\pi k/N)} \qquad 6.16$$

and

$$b_M = \frac{2rN \cos(2\pi k/N)}{4 \sin(2\pi k/N)} \qquad 6.17$$

whence

$$\left(\frac{k}{N}\right) = 0.125 \quad \text{or} \quad 0.375 \qquad 6.18$$

This places an even more severe limitation upon the usable frequency range. The results of the analyses are shown pictorially in Figure 6.6 where the severe frequency limitation of the Type II realisation is striking.

In practice, a scaling factor would be chosen as an integer power of 2 less than N/2, which would only serve to reduce the usable frequency range. For example if the scaling factor were chosen as 1/16, the Type I realisation would not overflow over the half sampling frequency range, the Type II realisation would be limited to a range of 700 Hz to 1500 Hz and the Type III realisation would have a usable frequency range up to 1400 Hz. These figures assume the sampling frequency to be 4600 Hz, the number of frequency samples N to be 46 and the damping factor δ as 1/512.

The range of the Type II realisation may be increased before overflow occurs by dividing the input signal by 4 as for the Type III realisation.

Referring to Figure 6.4, the relevant signal values are

$$\left.\begin{aligned} b_J &= \frac{N}{8} \\ b_L &= \frac{N}{8\ \sin(2\pi k/N)} \\ b_M &= \frac{2rN}{8\ \tan(2\pi k/N)} \end{aligned}\right\}$$

whence $$\frac{k}{N} = 0.07 \quad \text{or} \quad 0.43 \qquad 6.20$$

which at a sampling frequency of 4600 Hz gives a range of frequencies free of overflow from 322 Hz to 1978 Hz. This strategy does have the disadvantage that an extra attenuation of the signal has been introduced.

6.7 Summary and Recapitulation.

This chapter has investigated the choice of parameters used in a simulation to be based upon a particular frequency specification. Examination of Figure 6.6 shows quite conclusively that the Type I realisation avoids the overflow condition over the whole working frequency range. The severely limited range of the Type II realisation can be increased but at the expense of dynamic range. The Type III realisation is satisfactory up to a frequency of the order of 0.4 of the sampling frequency.

In the simulation which will be used for measurement and validation of the noise performance analysis and proposed best fit method of filter testing, the following parameters will be used

Number of resonators	7
Coefficient word length	12 bits
Analogue-to-digital word length	12 bits
Damping factor	2^{-9}
Scaling factor	16

It is appropriate at this point to recapitulate and to draw up a comparison between the three filter realisations. In Chapter 5 it was shown that the Type II realisation possessed the most attractive noise performance in that the noise-to-signal ratio was independent of frequency. The Type I realisation produced the poorest noise performance while the Type III realisation gave satisfactory performance above a frequency about one-fifth of the sampling frequency. In addition, the use of a double precision accumulator was shown to have significant advantages.

In the light of the overflow analysis, which is a general analysis not related to any particular filter specification, the choice of the filter with the best performance in all respects becomes difficult. If overflow can be tolerated provided the frequency of such an occurrence is low, the Type II realisation appears to be the obvious choice. If overflow is not tolerable, the choice devolves upon the Type I realisation. But it really depends upon the siting of the centre frequency of the filter. Thus, once again, there is no hard and fast set of rules which can be laid down to lead the designer to a firm choice of realisation. In the case of the doppler filter specification given in Figure 1.1, the Type I realisation would appear to be the most appropriate choice.

In the simulation tests described in Chapter 8, all three realisations will be investigated.

6.8 References.

6.1 Weinstein, C.J., 'Quantisation Effects in Digital Filters', MIT Lincoln Lab. Report 468, November 1969.

6.2 Rabiner, L.R., Gold, B. and McGonegal, C., 'An Approach to the Approximation Problem for Non-recursive Digital Filters', IEEE Trans. on Audio and Electroacoustics, AU-18, No 2, June 1970, 83-106.

6.3 Oppenheim, A.V. and Weinstein C.J., 'Effect of Finite Register Length in Digital Filters and the Fast Fourier Transform', Proc. IEEE, Vol 60, No 8, August 1972, 957-976.

6.4 Agarwal, R.C., 'On the Realisation of Digital Filters' PhD Thesis, Rice University, 1974.

6.5 Ahmad, N. and Jayapalan J.P., 'On Digital Filter Implementation via Microprocessors', IEEE Trans. Industrial Electronics and Control Instrumentation', IECI-23, No 3, August 1976, 249-253.

6.6 Jackson, L.B. 'Roundoff Noise Analysis for Fixed Point Digital Filters realised in Cascade or Parallel Form', IEEE Trans. Audio and Electroacoustics, AU-18, No 2, June 1970, 107-122.

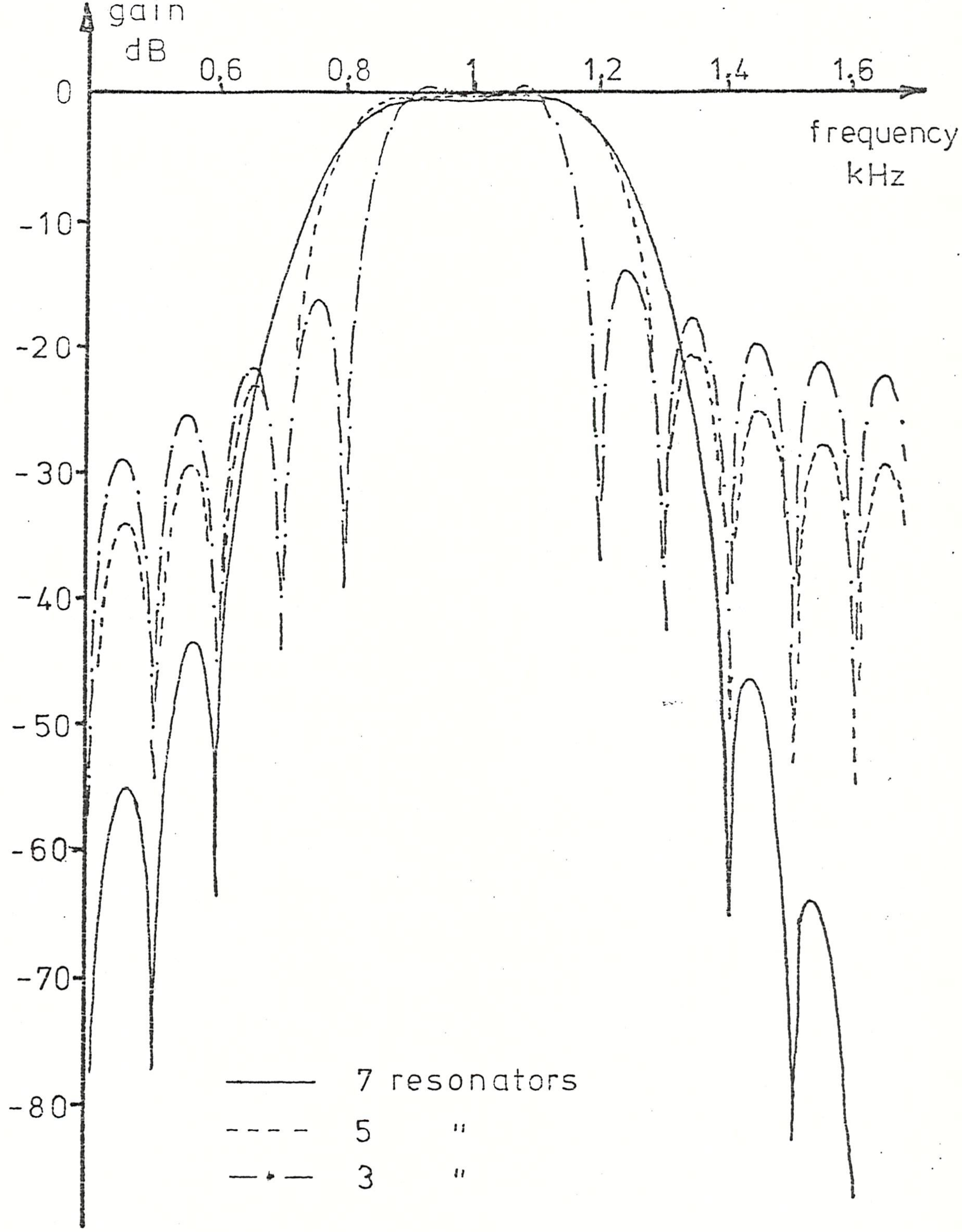


Figure 6.1 Effect of Number of Resonators upon the Frequency Response

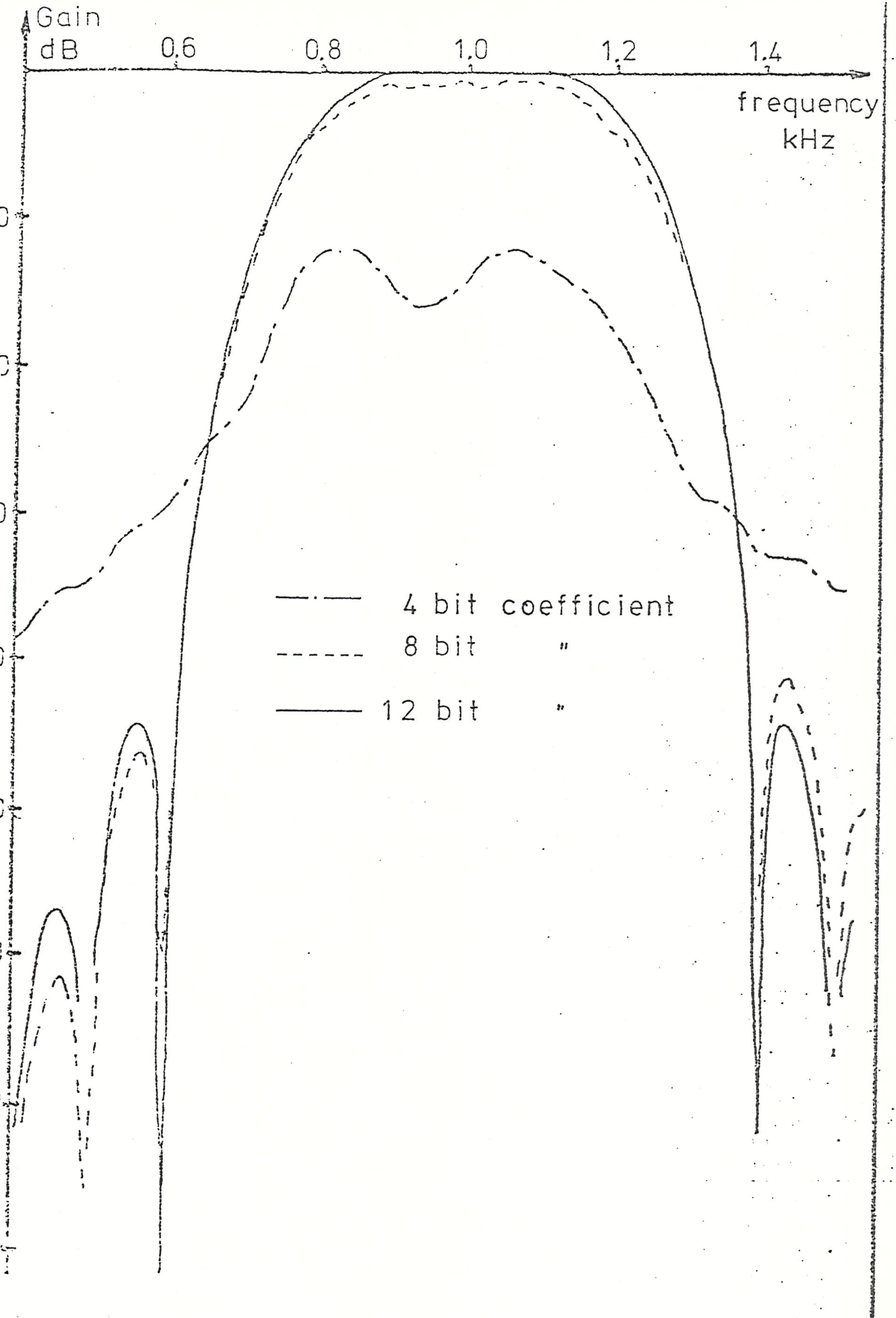


Figure 6.2 Effect of Coefficient Wordlength upon the Frequency Response

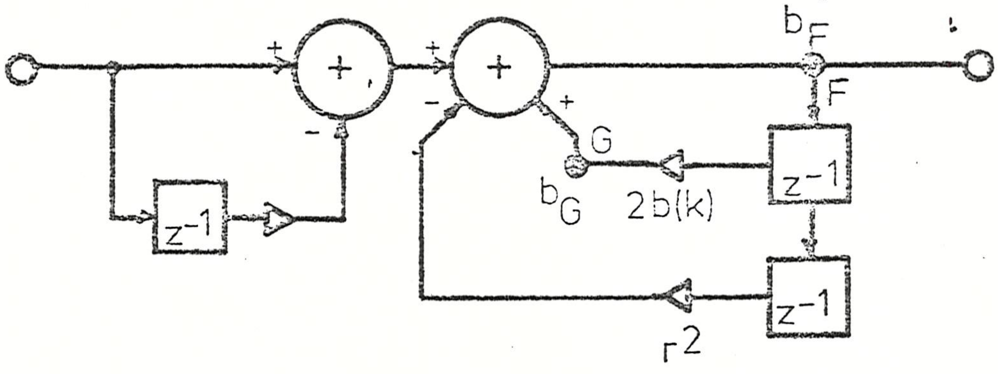


Figure 6.3 TYPE I Resonator

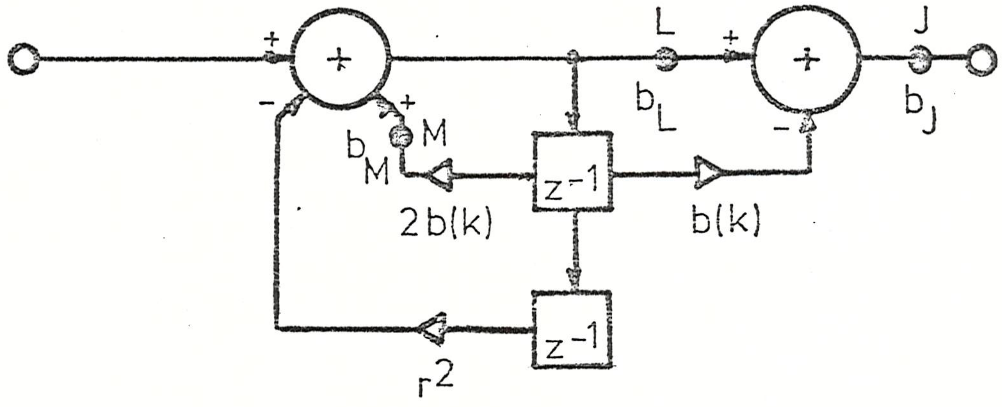


Figure 6.4 TYPE II Resonator

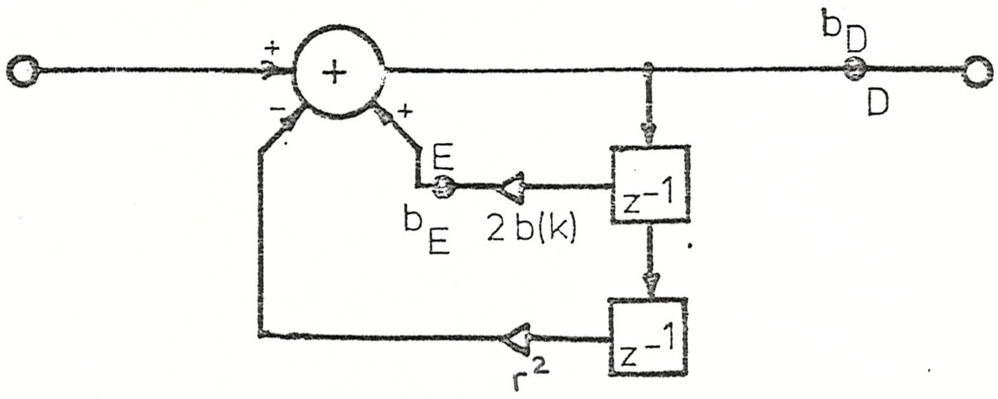


Figure 6.5 TYPE III Resonator

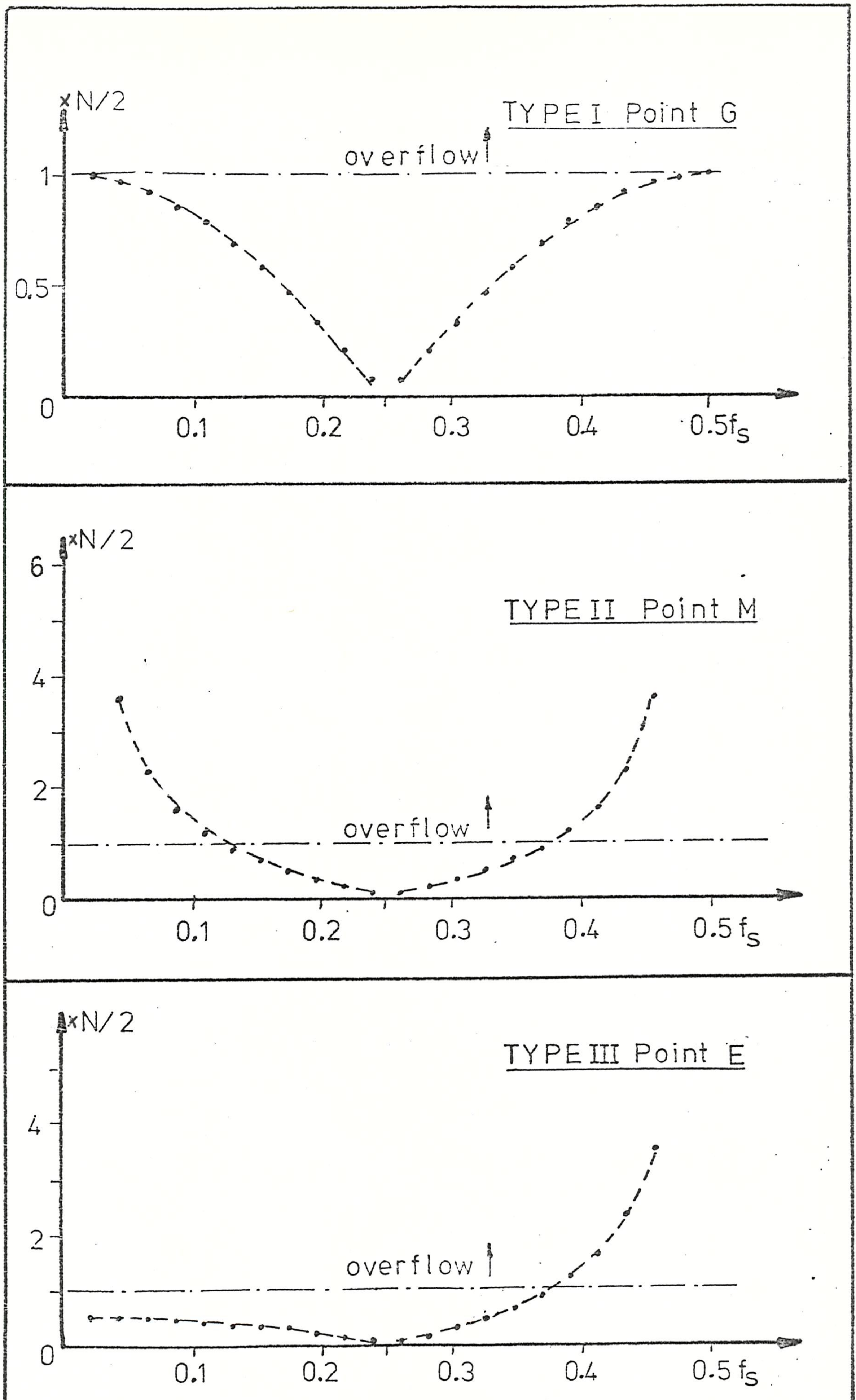


Figure 6.6 Signal Level in Resonators

CHAPTER SEVEN

SIMULATION PROGRAMS

7.1 Introduction.

Previous chapters have presented the theory, noise analysis and simulation parameters required in the investigation of frequency sampling digital filters. It is now necessary to write a series of general purpose computer programs which allow sufficient flexibility to enable parameter changes to be effected easily while at the same time maintaining a reasonable degree of economy in the computational time required.

As several of the components of the simulation are common, it was decided to build the program in the form of a main program together with a series of sub routines appropriate to the filter realisation being simulated. The flow-charts for the best-fit analysis, the noise comparison and the impulse response simulations are shown in Figures 7.1, 7.2 and 7.3, the listing of each main program and sub routine being given in Appendix A.

The language used for the simulation programs is FORTRAN IV.

7.2 Description of Sub-routines.

Two distinct categories of sub routine are used in the simulation, the first comprising those common to all filter realisations, the second being those peculiar to a particular realisation, as shown in Table 7.1.

Common Sub-routines	Particular Sub-routines
SIGMA	FILCON
SREGRES	INITDLY
VV	DIGIFILT
FFT	

Table 7.1 Simulation Sub-routines.

With the exception of SUB-ROUTINE FFT which is a modified version of the program suggested by Coates (Reference 7.1) all sub-routines have been developed specially for this investigation.

SUB ROUTINE SIGMA (Figure A1a)

This sub-routine stores the summations of the input and output sequences required to carry out the best-fit analysis described in Chapter 5. It is not called in the main program until the filter output has reached a steady-state and sufficient time has elapsed for the noise to have built up to its maximum value.

SUB ROUTINE SREGRES (Figure A1b)

The summations performed in SUB ROUTINE SIGMA are used by this sub-routine to calculate the gain, phase and noise output of the filter using the formulae derived in Chapter 5. Initially, a diagnostic routine was used to identify errors in the results but this was removed once the programs had been validated.

DOUBLE PRECISION FUNCTION VV (Figure A1c)

The products of multiplication and the output of the analogue-to-digital converter may be rounded or truncated by this sub-routine. Rounding is effected by setting OX to Ø.5DØ, truncated by setting OX to ØDØ.

SUB ROUTINE FFT (Figure A2)

Used in conjunction with the impulse response main program, this sub-routine accepts 512 samples of the filter output and performs a Fast Fourier Transform to give the frequency response. The frequency spacing between each output value is given by sampling frequency ÷ number of samples, which in this investigation is almost 9 Hz.

SUB ROUTINE FILCON (Figure A3a and A4a)

This sub-routine calculates the filter coefficients, as defined in the previous chapters, and initiates a printout of the rounded values. Two versions of this sub routine are necessary, FILCON 12 for the Type I and Type II filters, FILCON 3 for the Type III filter.

SUB ROUTINE INITDLY (Figure A3b and A4b)

Before a measurement is made on the filter simulation, the filter delay elements must be initialised by setting the stores representing the delays to zero. If other preconditions are required, the sub routine must be modified accordingly.

SUB ROUTINE DIGIFILT (Figures A5, A6, A7)

This represents the simulation of the actual filter and is available in three versions corresponding to the three filter types. Explanatory diagrams have been provided in the Appendix in an attempt to clarify the program.

7.3 References.

7.1 Bogner, R.E. and Constantinides, A.G., 'Introduction to Digital Filtering' Wiley, 1975.

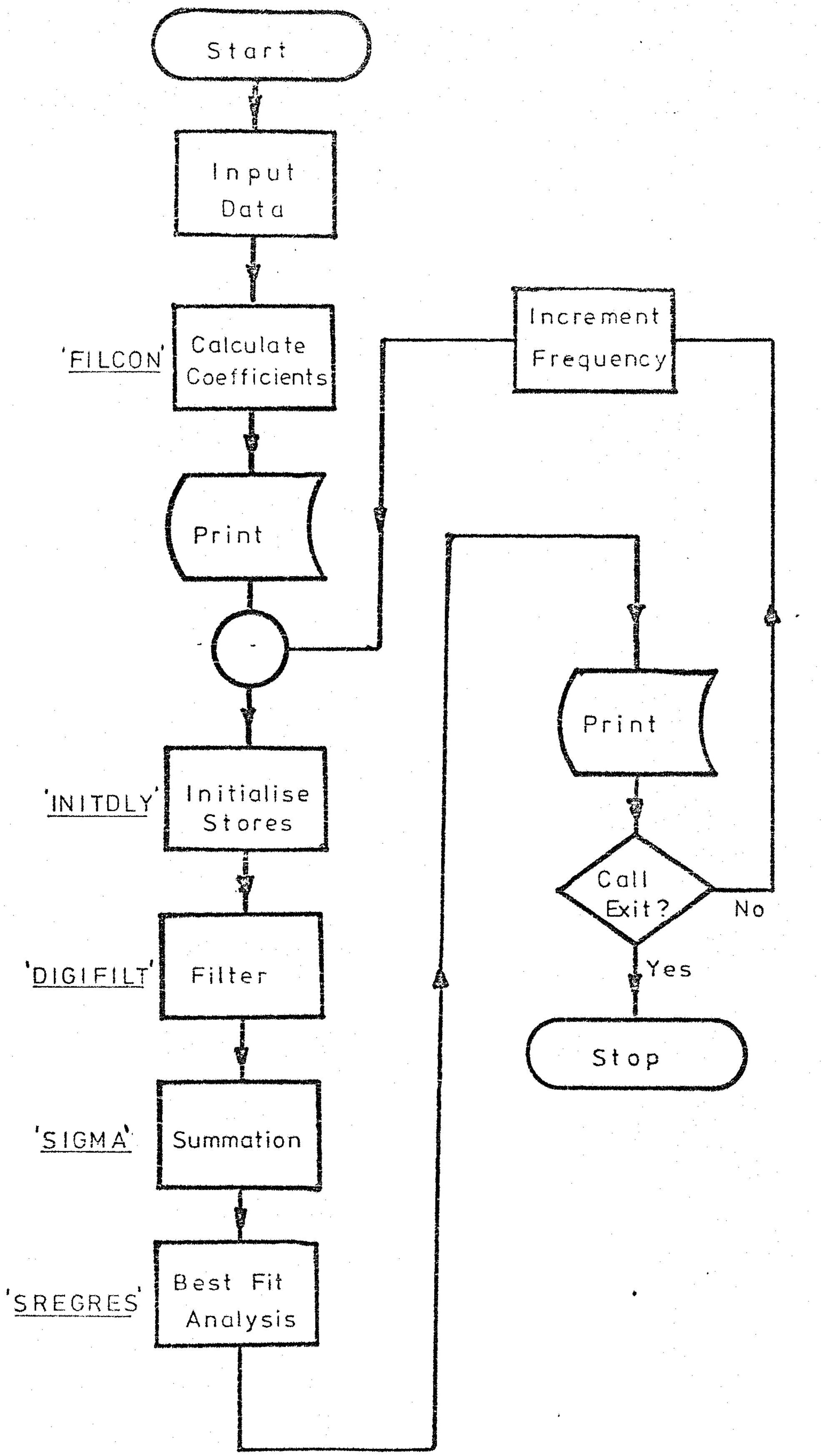


Figure 7.1 Flowchart for Best-Fit Program

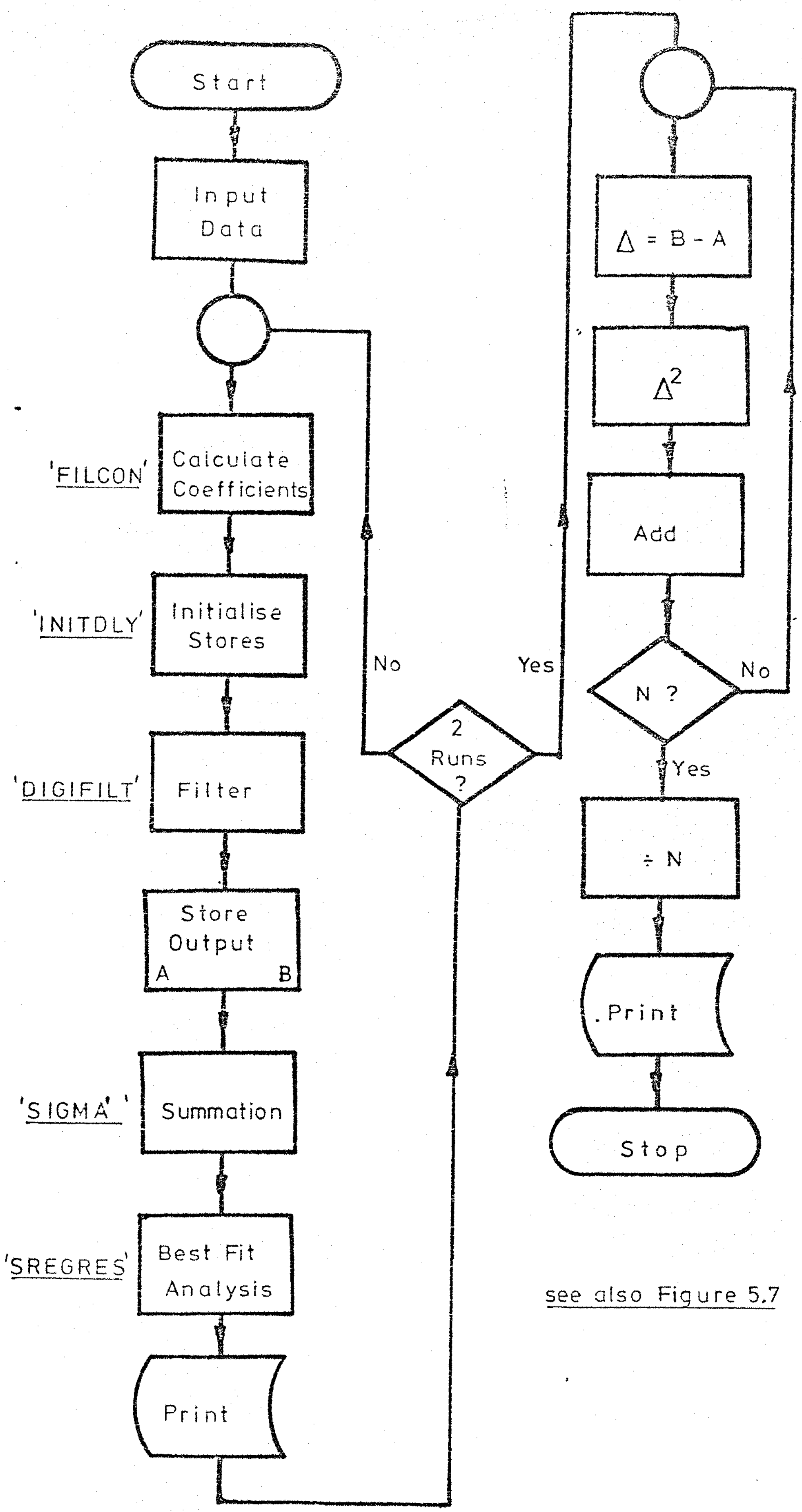


Figure 7.2 Flowchart for Noise Comparison

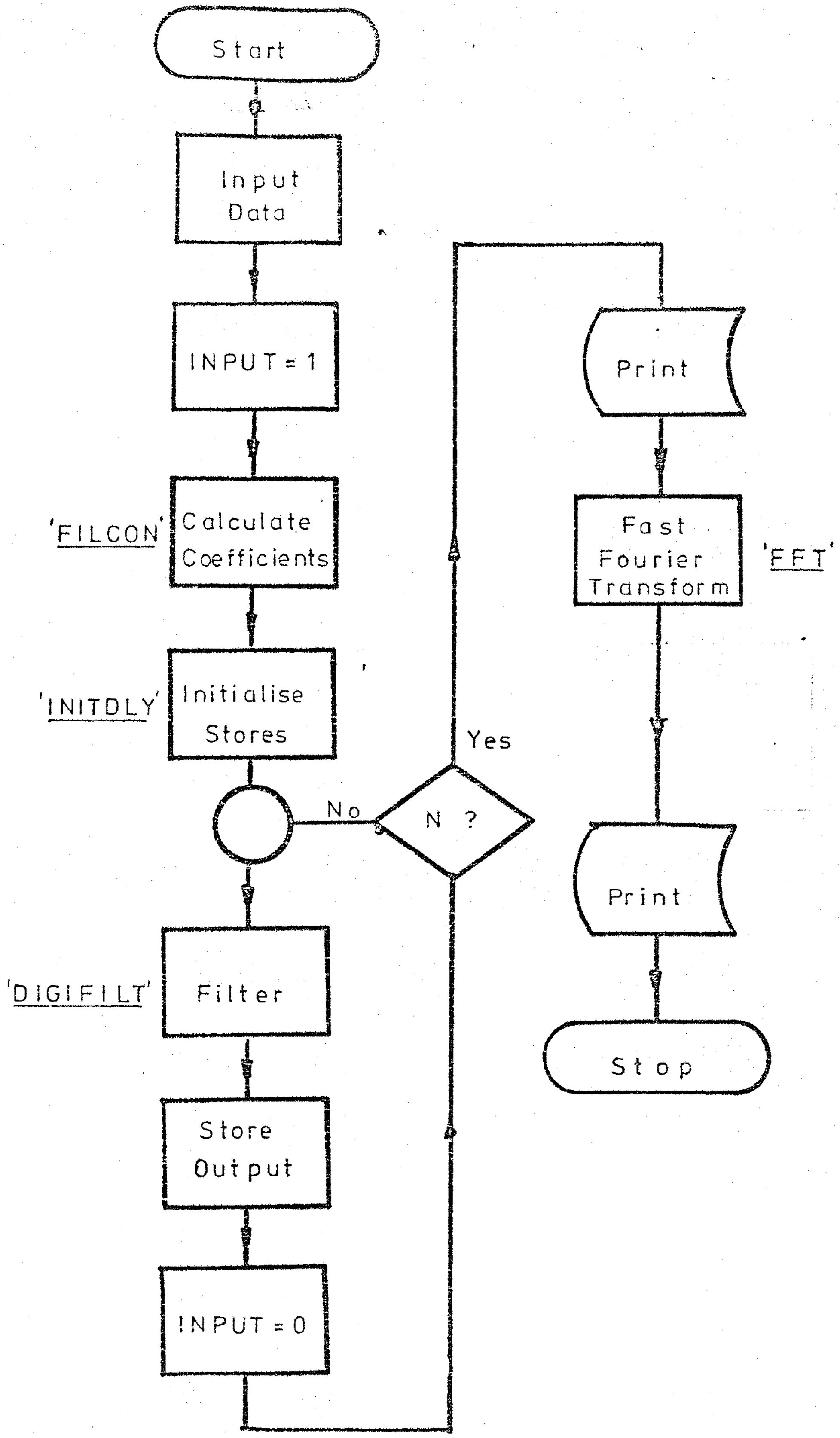


Figure 7.3 Flowchart for Impulse Response and Fast Fourier Transform

CHAPTER EIGHT

MEASUREMENTS ON THE SIMULATION

8.1 Introduction.

Preliminary measurements made by a direct computation of the pulse transfer function in the frequency domain have determined the parameters of the digital filter in respect of the number of resonators, the degree of resonator damping and the coefficient word length. Such measurements give no indication of the noise performance of the filter when finite word lengths are used to represent the arithmetic. The time domain simulation programs described in Chapter 7 allow noise measurements to be made and offer the opportunity to validate the theories presented in earlier chapters. Before effective measurements can be made, it is necessary to determine when measurement may be started, since the filter must first be allowed to settle ie the filter output must have reached the steady-state condition. Because the resonators represent the source of most of the noise output, initial measurements to determine the settling time are made on the resonator section of the filter.

The proposed best-fit method of filter performance measurement must then be validated. This is effected by comparing this method with the conventional two-pass method using a so-called precise filter as a standard.

Limit cycling and its effects upon performance are investigated as well as the effects of rounding or truncation of the resonator arithmetic.

8.2 Resonator Performance.

A single resonator having a centre frequency of 1000 Hz, a damping factor of 2^{-9} and a word length of 16-bits was used as a vehicle for this investigation. The resonator coefficients were quantised to a 12-bit word length. The damping factor controls the gain at resonance of the resonator together with its bandwidth although the Type III realisation employs resonators with frequency dependent gain. However, at the frequency chosen ie 1000 Hz, it has been assumed that the resonator gain is approximately 256 and the resonator bandwidth is 2.9 Hz for all three realisations.

According to the classical filter theory, the settling time required to reach steady-state conditions can be estimated from a knowledge of the filter bandwidth, B. The theory states that if a sinusoid is suddenly applied to the input of a filter, the output will rise from zero to a steady-state value with a time constant τ such that

$$\begin{aligned} \tau &= 1/\pi B \\ &= 110 \text{ milliseconds} \end{aligned} \qquad 8.1$$

having already assumed that the bandwidth is set at 2.9 Hz and each sample occurs every 1/4600 seconds. The filter must be allowed to run for at least 3τ before measurements can be taken, which corresponds to about 1500 samples. In the simulation, considerably more central processor time is required.

It is suggested that the classical theory can be applied to white noise generated in the resonators for they possess such a narrow bandwidth that the white noise approximates to a summation of sinusoids within a very narrow band. This assumption was tested in an analogue system in which white noise was passed through a narrow bandpass filter, the probability density function of the filter output resembling that of a sinusoid.

Figure 8.1a shows how the resonator signal output behaves with time while Figure 8.1b shows the buildup of output noise from the resonator. Both sets of results support the above theory and show that in the frequency sampling filter, the bandwidth of the individual elements within the filter realisation are of more consequence than the parameters of the overall filter. As a contrast to the behaviour just described, the output of a complete filter from zero to 30 milliseconds was plotted as shown in Figure 8.2. The output is delayed by the delay of the filter equal to half the number of samples, after which the amplitude rises to a steady-state value. Such a measurement could in itself be misleading in that it could be assumed that if the filter output reaches a steady-state in a short time, the noise output should behave in the same way. But because the noise derives in the main from the resonator having a much higher Q-factor, this assumption is erroneous.

Returning to the resonator, Figure 8.3 shows the frequency response of a 1000 Hz resonator. The measured bandwidth is 2.7 Hz and the measured gain is 261 showing that the approximations made in the theoretical analyses have been justified.

8.3 Validation of the Best-Fit Method.

To test the validity of the best-fit method presented in Chapter 5, a Type I filter realisation was chosen having a centre frequency of 1000 Hz and a bandwidth of 400 Hz. The coefficients were quantised to 12-bits, the arithmetic word length was rounded to 16-bits and the resonator damping factor was set at 2^{-9}. Test frequencies were chosen at random in the stopband and in the passband, the standard 'precise' filter having an arithmetic word length of 30 bits.

The results of the comparison of the best-fit method against the two-pass method showed a good agreement except at the resonator frequencies when the best-fit method gave a higher noise output about 3% above the value given by the two-pass method. As already discussed, because of the high gain at resonator frequencies, anomalies are to be expected. No attempt has been made to offer an explanation of these anomalies because of the relatively low percentage error which does not materially alter any of the theory presented. A closer investigation of the comparison about the resonator frequency showed that the difference was significant only within the bandwidth of the resonator.

As a final check, the number of samples used to compute the noise output was increased. A close agreement was obtained between the two methods except at exactly the resonator frequency. The actual values of noise at particular frequencies was different from the smaller number of samples although this is to be expected. If the noise process is truly random, then the period of measurement must be equal to at least the reciprocal of the resonator bandwidth. Alternatively, several measurements of smaller duration may be made and the average of the measurements taken. In either case, inordinately long computing times will be required.

Assuming that the scatter in the noise measurements may be tolerated, smaller measurement periods may be used to give an approximate value of the output noise which for all practical purposes will suffice, precise measurements being unnecessary.

8.4 Filter Performance Measurements.

Considering first the phase characteristic of the digital filter realisations, it was shown in Chapter 4 that the phase difference between the Types I and II and the Type III realisations is $k\pi/N$. Figure 8.4a shows the measured phase response of the Types I and II, Figure 8.4b shows the phase response of the Type III realisation. In both cases the phase response is linear within the passband of the filter and in Figure 8.4a it is seen that linear phase is produced in the stopband with a phase excursion of 180° rather than 360° as in the passband. The phase response of the Type III filter realisation exhibits unexpected characteristics in the upper stopband for which no satisfactory explanation has been found.

The gain response of the filter realisations measured by the best fit method satisfies the filter specification irrespective of the arithmetic word length for it is the coefficient word length which determines the closeness of the approximation to the desired gain response. Slight discrepancies were observed in the case of resonator arithmetic truncated to 12-bits. Closer examination in the close vicinity of the resonator frequencies once again showed that the discrepancies are confined to the resonator bandwidth and are not significant in practice.

Chapter 5 presented noise performance analyses which assumed that the analogue-to-digital converter output noise was severely attenuated by the scaling factor, reducing it to negligible proportions. In order to avoid any effects caused by the analogue-to-digital converter, a signal word-length of 12-bits was chosen. Reference 8.1 gives an approximate formula for the determination of the converter word length given the desired signal-to-noise ratio at the output of a digital filter. It does not include any account of scaling factors but shows that a 12-bit converter would give at least a 80 dB signal-to-noise ratio. This figure, checked using the two-pass method was found to be approximately 82 dB, showing a very close agreement with the predicted value.

With this choice of converter word length, the simulation parameters have been determined and it is now possible to validate the predictions of filter performance made in previous chapters.

The first series of measurements was designed to validate the predictions summarised in Figure 5.9. The input signal used was a sinusoid whose frequency was varied over the ranges of interest. The peak signal amplitude was set to unity.

Using the best fit method, measurements of gain, phase and rms noise output were made, the results of the noise measurement summarised in Table 8.1.

Filter	Realisation		
	Type I	Type II	Type III
A	54 dB	62 dB	56 dB
B	61 dB	63 dB	64 dB
C	56 dB	63 dB	67 dB

Table 8.1 RMS noise output (16-bit word length) in dB.

The figures shown in Table 8.1 are produced by averaging the measurements over the passband. An additional measurement was made on a Type II filter simulation having rounded double precision accumulators, the rms noise output averaging at 71 dB below 1 V. Comparing these results with the predictions for noise-to-signal ratio shown in Figure 5.9, it can be seen that the general pattern of behaviour is reflected in the measured results although noise-to-signal ratio was not itself measured. For example, the Type II filter realisation exhibits an almost constant noise output over the frequency range of interest while the Type III filter yields less noise output as the frequency increases.

The predictions made in Chapter 5 would therefore appear to have been justified and the best-fit method validated once again.

8.5 Further Measurements.

At this stage, the majority of the aims of the investigation had been achieved and a satisfactory frequency sampling digital filter simulation effected. For the sake of completeness, other aspects of filter performance may be investigated by utilising the flexibility of the simulation programs.

As the filter performance has already been tested over a range of frequencies, further measurements may be made on the filter realisations using frequencies which are least likely to include effects such as overflow or severe coefficient quantisation sensitivity. The frequency range chosen was that employed for the B filter specification ie a filter having a centre frequency of 1000 Hz and a bandwidth of 400 Hz. The following tests were made:

a. Effects of truncation of the resonator arithmetic upon the noise output.

b. Effects of rounding and truncation upon the impulse response of the filter realisations,

c. Effects of rounding and truncation upon the Fast Fourier Transform derived frequency response.

d. Effects of reduction in arithmetic word length upon filter performance, impulse response and Fast Fourier Transform derived frequency response.

e. Effects of signal amplitude and phase upon filter performance.

f. Validation of the effect of changes in damping factor upon the noise performance.

Some of these effects may be predicted with a fair degree of certainty. For example, truncation of the arithmetic in the resonators will be accompanied by an increase in the noise output. Nevertheless, it is considered that it is prudent to take advantage of the opportunity to verify the predictions in case anomalies are present which could bring the initial assumptions into doubt.

According to the accepted theories, truncating the multiplier outputs in the resonators should have two major effects. In the first place, the noise performance should deteriorate by 6 dB and in the second place, the phenomenon of limit cycling should be significantly reduced.

The initial series of tests employed a 16-bit arithmetic word length, the outputs of the resonator multipliers being rounded. The three filter realisations were tested for gain, phase and noise performance using the best-fit method.

An impulse response measurement was made, 512 samples of the impulse response being used as input points to a Fast Fourier Transform routine to yield the interpolated frequency response, each frequency spaced at approximately 9 Hz. The results of these tests are shown in Figures 8.5 to 8.13. In all cases, the noise output lay below the maximum sidelobe amplitude with an average level of about -60 dB referred to 1 volt. As predicted, the Type II filter realisation gives the best noise performance. The impulse responses shown in Figures 8.6, 8.9 and 8.12 appear to be of finite duration due to the scales used, the actual computer results showing a cycling of very small amplitude as seen in the inset of Figure 8.6. Figure 8.12 shows the non-integer delay of the Type III filter. The FFT derived frequency responses are very similar to the best-fit measured responses.

This procedure was then applied to the three filter realisations using truncated arithmetic in the resonators, the results being given in Figures 8.14 to 8.22. The noise outputs under this condition of truncation were found to be greater than those produced in the case of rounding and the impulse response was of finite duration lasting exactly 46 samples. One minor difference between rounding and truncation is that the sidelobe pattern produced from the Fourier analysis does not exhibit uneveness for truncation ie spurious sharp responses superimposes upon the sidelobe. It is assumed that the spurious responses are due to the low level limit cycling associated with the rounded resonators.

Thus, the measurement made on the filter realisations employing a 16-bit arithmetic word length have borne out the predictions given by a theoretical analysis and, other than a slightly increase noise output which still lies below the maximum sidelobe amplitude, give virtually the same performance.

The arithmetic word length was then reduced to 12-bits and the series of tests repeated, the results being shown in Figures 8.23 to 8.40. In all cases, the gain response satisfactorily met the specifications laid down but the noise output lay above the maximum sidelobe amplitude, a condition which could render the filter unsatisfactory for certain applications. It would certainly not meet the specification used in this investigation which requires the output of the filter to be less than -40 dB in the filter stopbands.

The behaviour of the noise output over the frequency range is strikingly different from that of the 16-bit arithmetic word length filters. Apart from the expected increase in noise output of about 24 dB, the variation of noise output is much larger making the comparisons between rounding and truncation difficult and showing, if anything, that the noise output does not appear to increase in the case of truncation. No satisfactory explanation of this can be offered.

The impulse responses for the case of the rounded arithmetic resonators show distinct evidence of limit cycling with a prounounced beat effect which has not been analysed except that these effects appear clearly in the FFT-derived frequency responses. Bearing in mind that another method of implementing digital filters uses Fourier Transform techniques, the limit cycling and beating renders the filter virtually unusable. In the case of truncated arithmetic resonators, the impulse responses, while being of finite duration, exhibit a form of reduced amplitude 'echo' so extending the response to twice that expected and causing severe ripple in the filter passband.

As a result of these measurements, it may be concluded that a filter of this type must be implemented using 16-bit arithmetic in order to give a satisfactory performance in all respects. When such word lengths are used, the difference between rounding and truncation becomes marginal.

Finally, in this series of measurements, a simulation of the Type II filter employing rounded double precision accumulators was made, the results being shown in Figures 8.41 to 8.43. The results confirm earlier predictions and indicate that this method offers the better solution in all respects.

Varying the damping factor had little effect upon the gain response of the filter and showed that the noise output increased by about 3 dB as the damping factor was decreased by a factor of 2 so confirming earlier predictions.

Decreasing the signal amplitude had no significant effect upon the filter performance until it approached the level of a single quantising step in the analogue-to-digital converter. At this point, the best-fit method yielded erratic results from which no significant conclusions could be drawn. A variation of the phase of the input signal appeared to have negligible effect upon the filter performance.

8.6 Summary.

A series of measurements have been made on simulations of the three filter realisation types in which the effects of arithmetic word length, rounding, truncating, damping factor, signal amplitude and phase upon the performance were investigated. Further investigations have been made into the impulse response and Fourier transform derived frequency response.

In the case of 16-bit arithmetic word lengths, the predictions of performance presented in earlier chapters have been confirmed, particularly the results of the noise analyses. If 12-bit arithmetic word-lengths are employed, the filter performance is seriously degraded to make it virtually unusable.

The results have highlighted a major disadvantage of the frequency sampling technique in that it demands long register lengths by virtue of the high gain, high-Q resonator elements necessary for its realisation.

8.7 Reference.

8.1 Ahmed, N. and Jayapalan, J.P., 'On Digital Filter Implementations via Microprocessors', IEEE Trans. Industrial Electronics and Control Instrumentation, IECI-23, No 3, August 1976, 249-253.

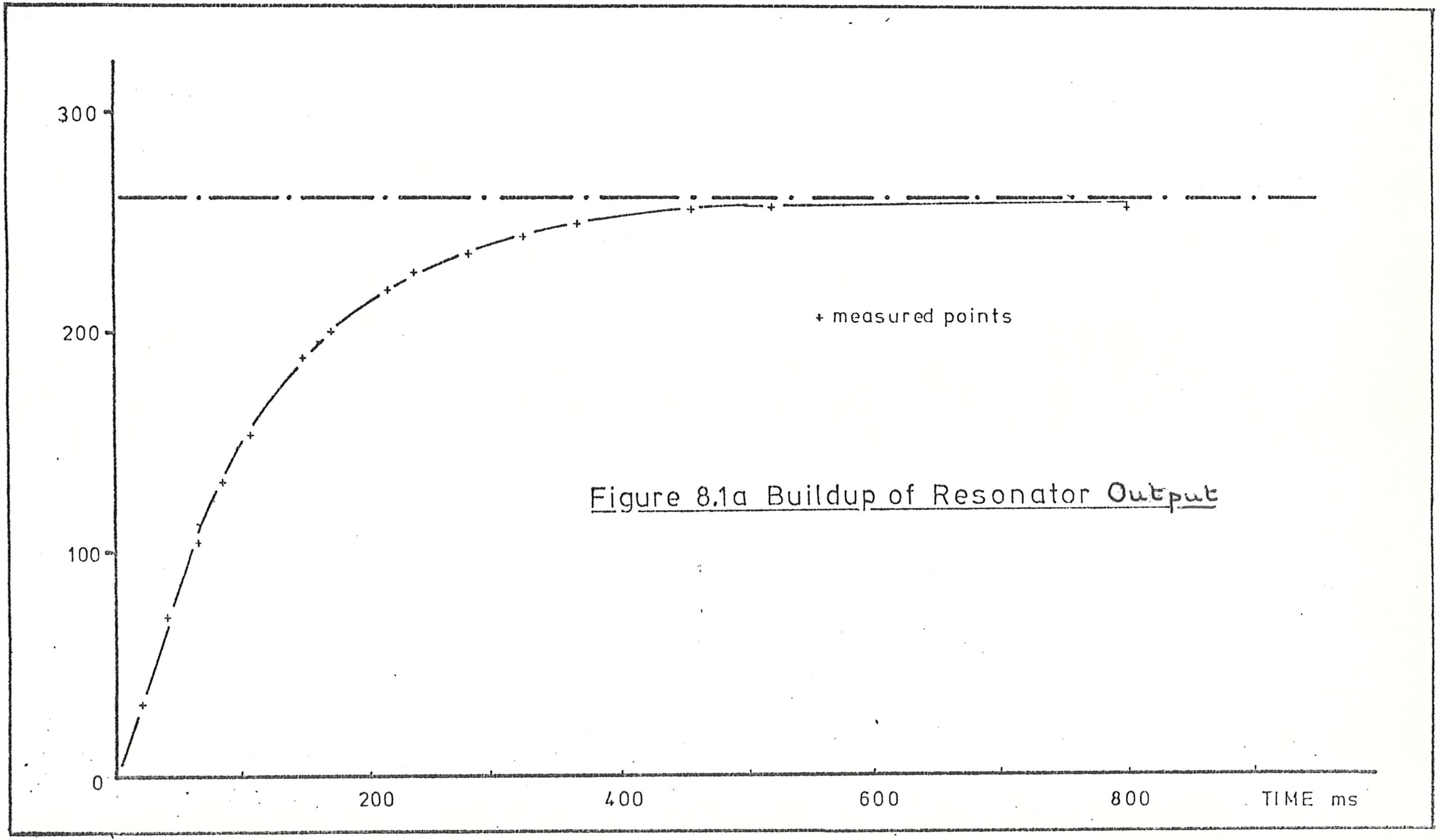


Figure 8.1a Buildup of Resonator Output

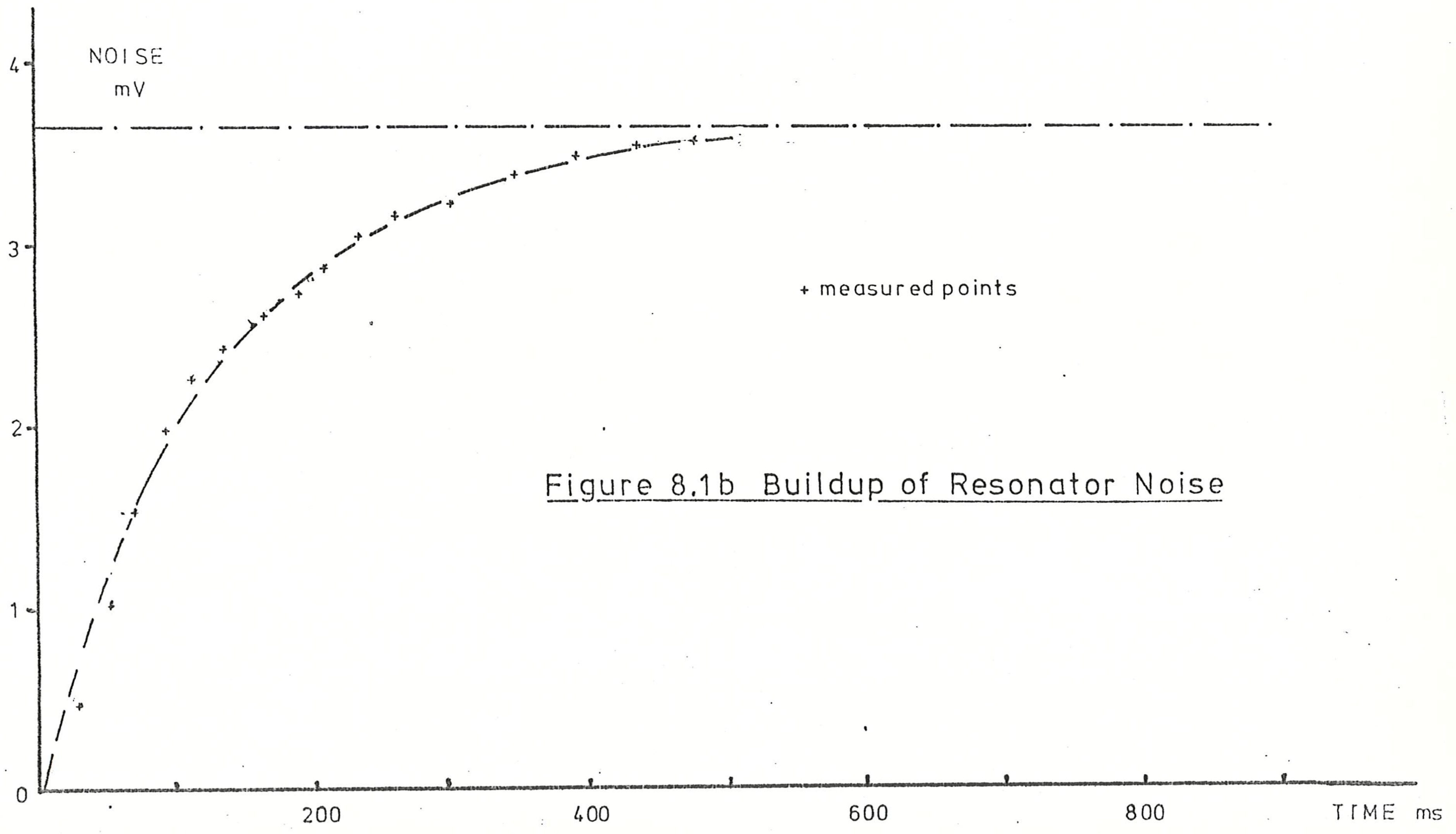


Figure 8.1b Buildup of Resonator Noise

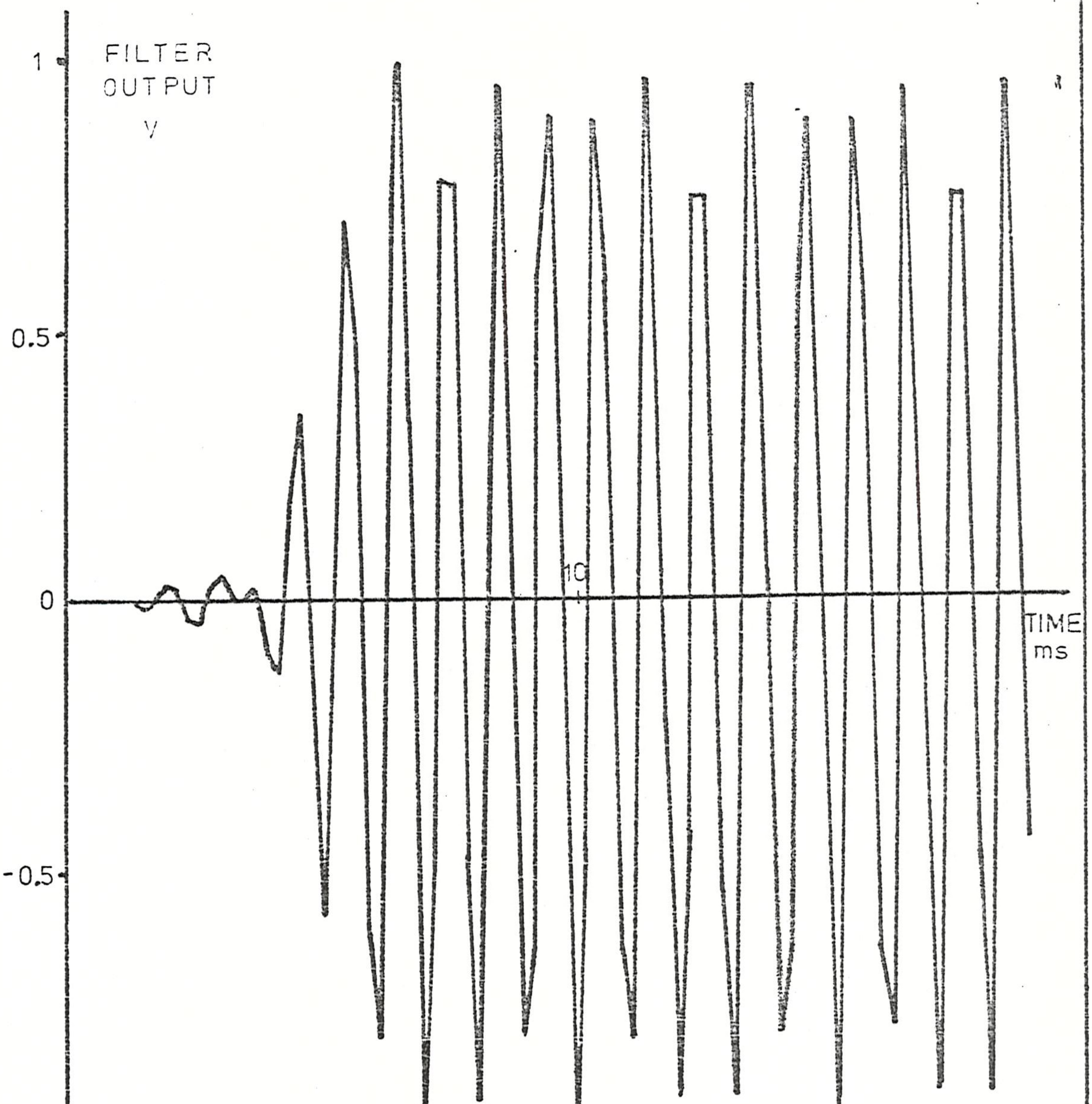


Figure 8.2 Filter Output at 1000 Hz

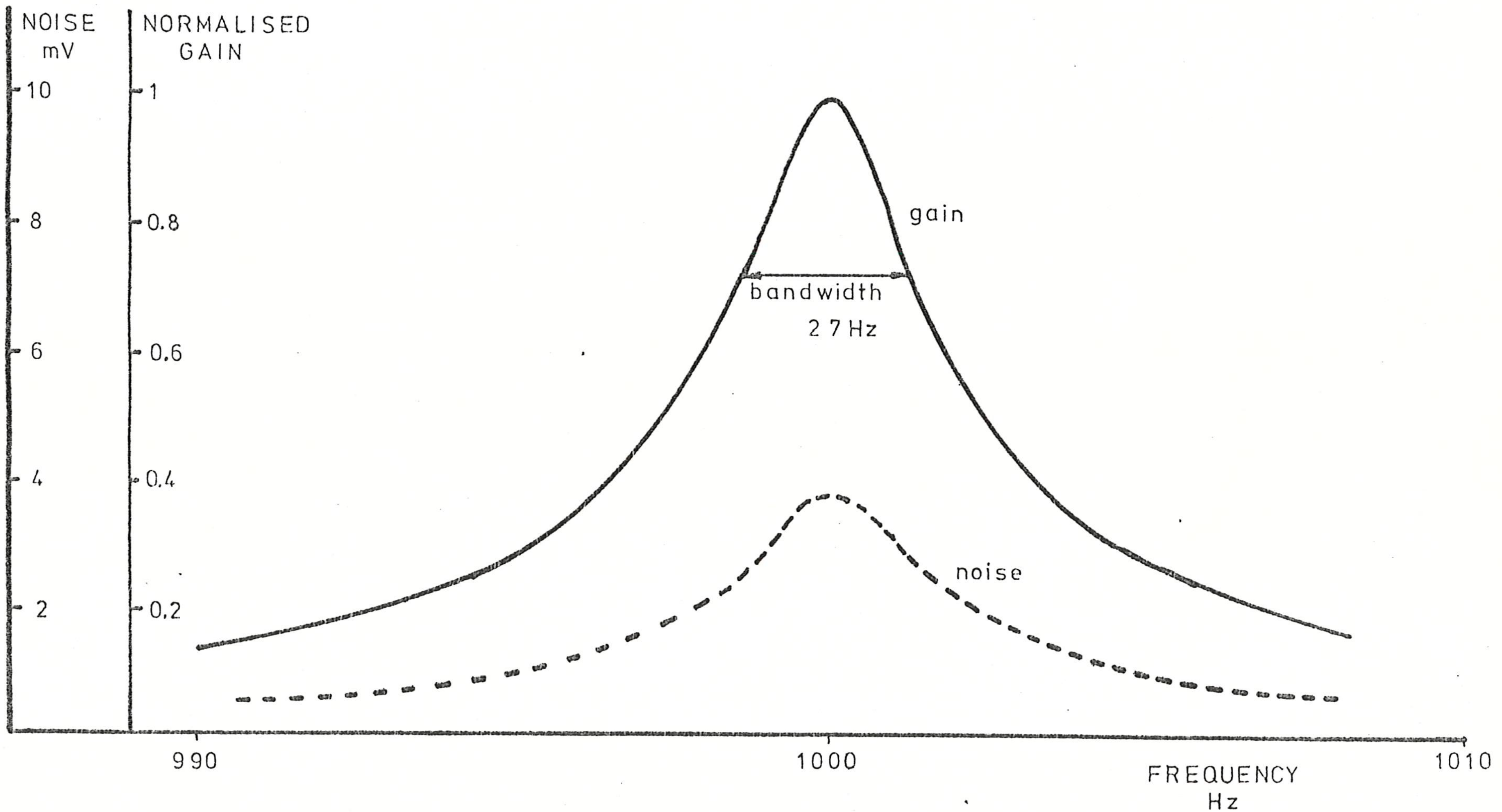


Figure 8.3 Gain and Noise Behaviour of 1000 Hz Resonator

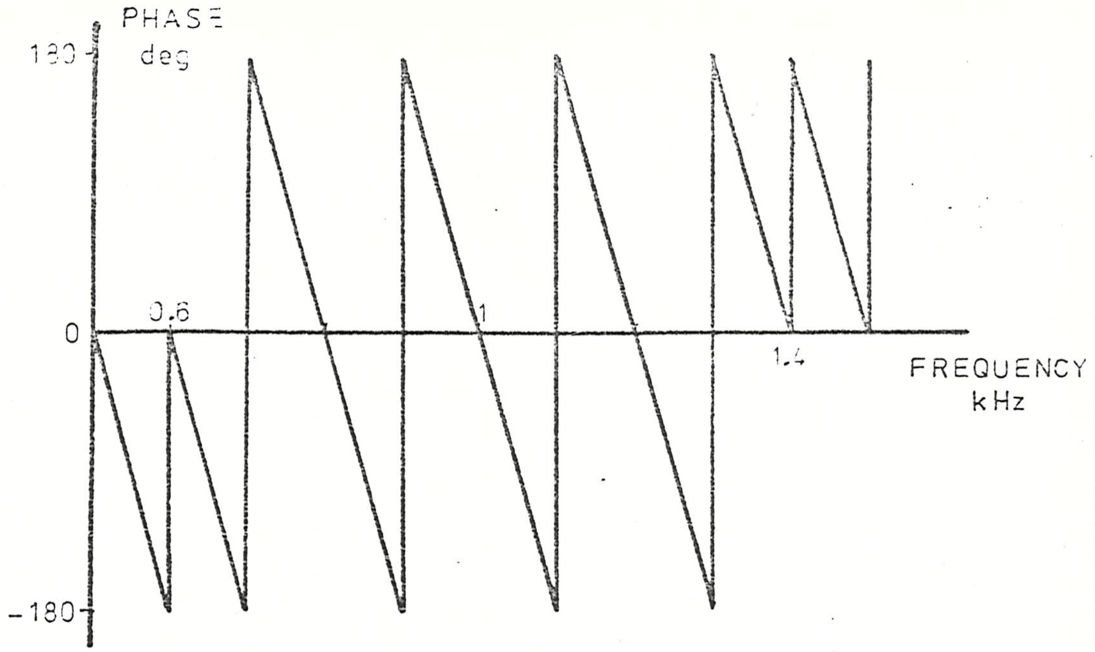


Figure 8.4a Phase Response - Types I & II

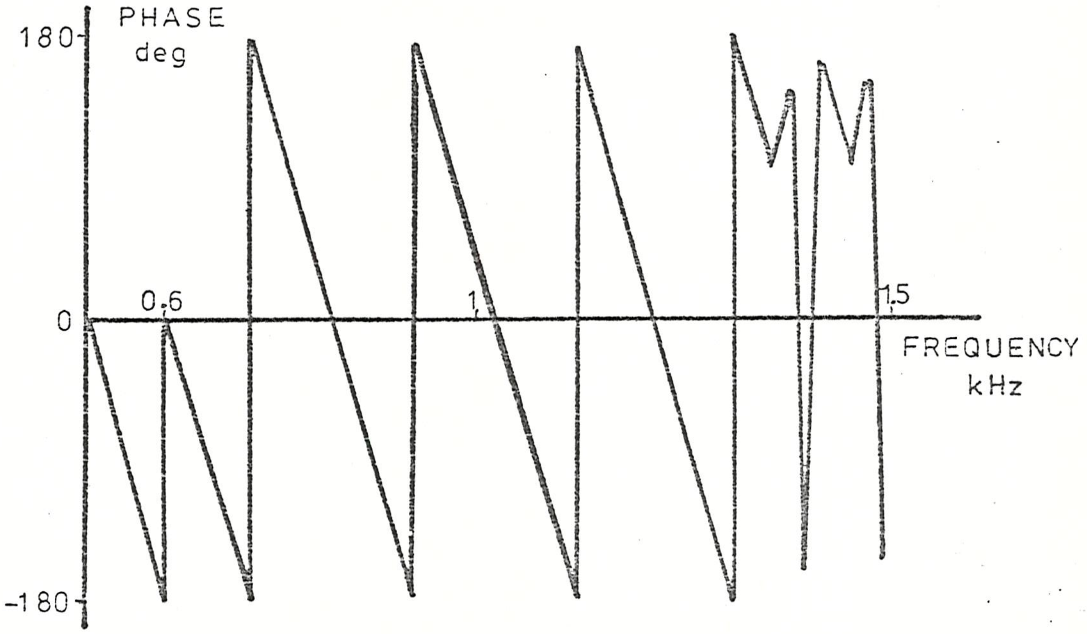


Figure 8.4b Phase Response - Type III

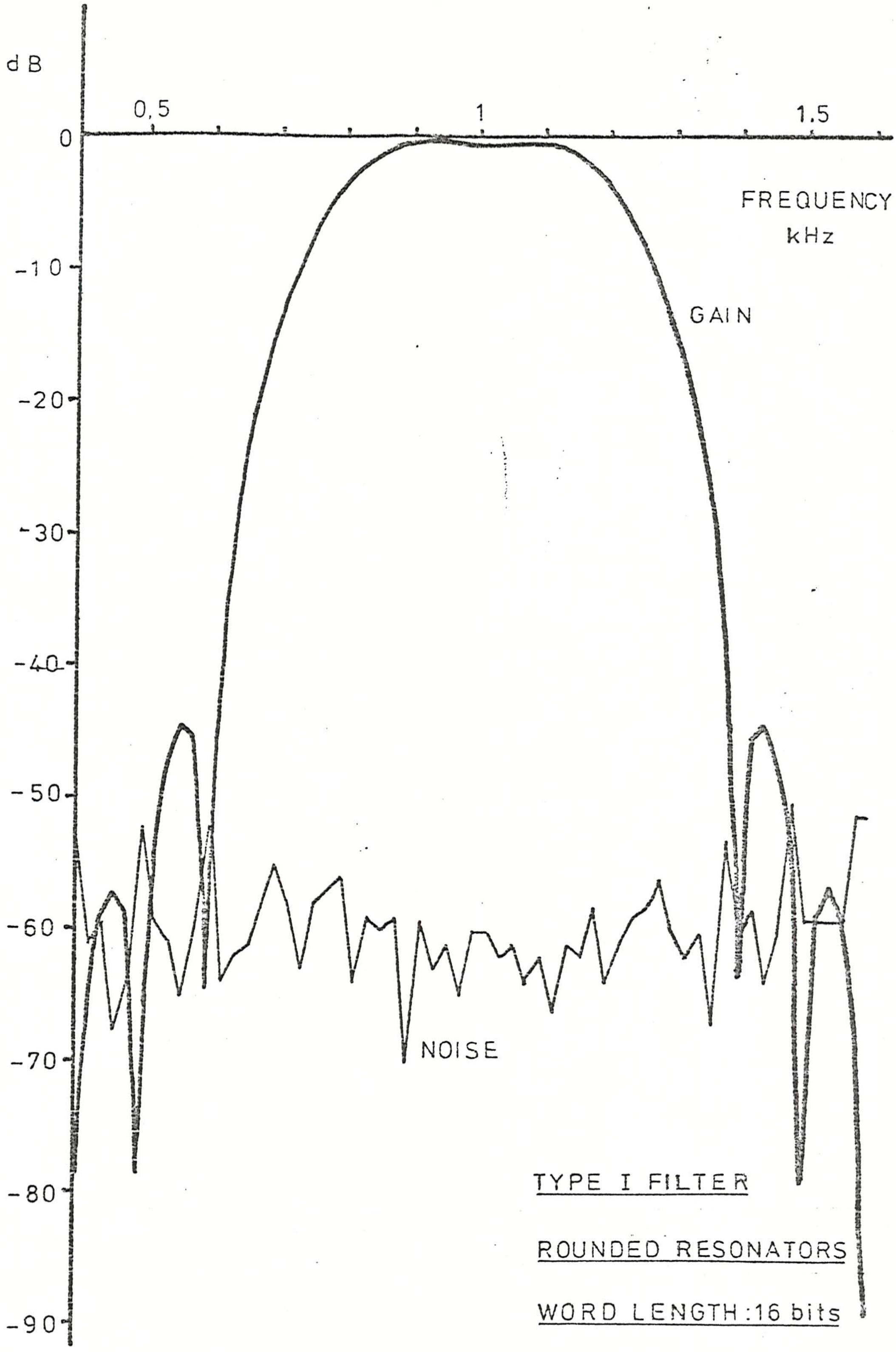


Figure 8.5 Best-fit Frequency Response and Noise Output

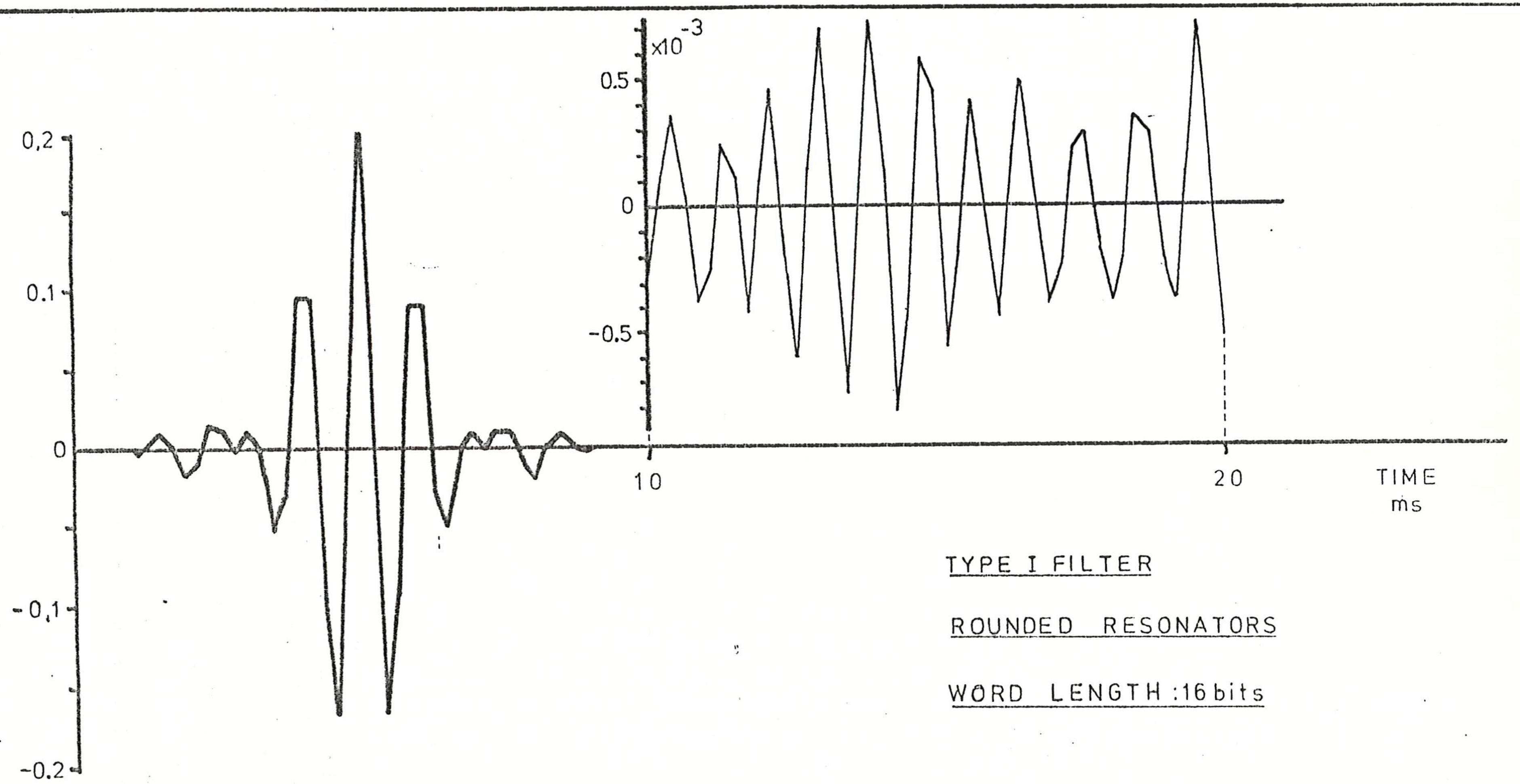


Figure 8.6 Impulse Response (showing cycling phenomenon)

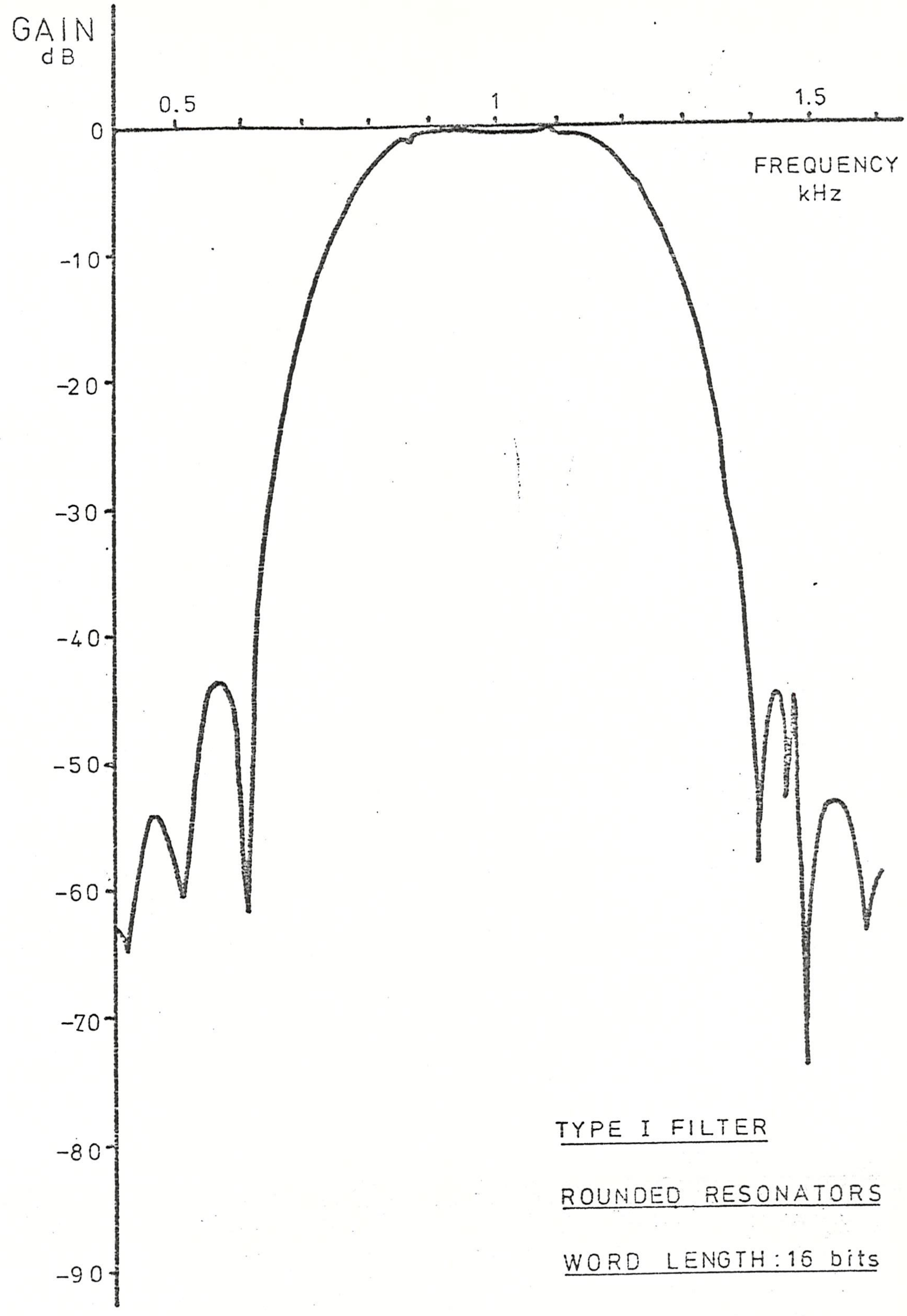


Figure 8.7 FFT-derived Frequency Response

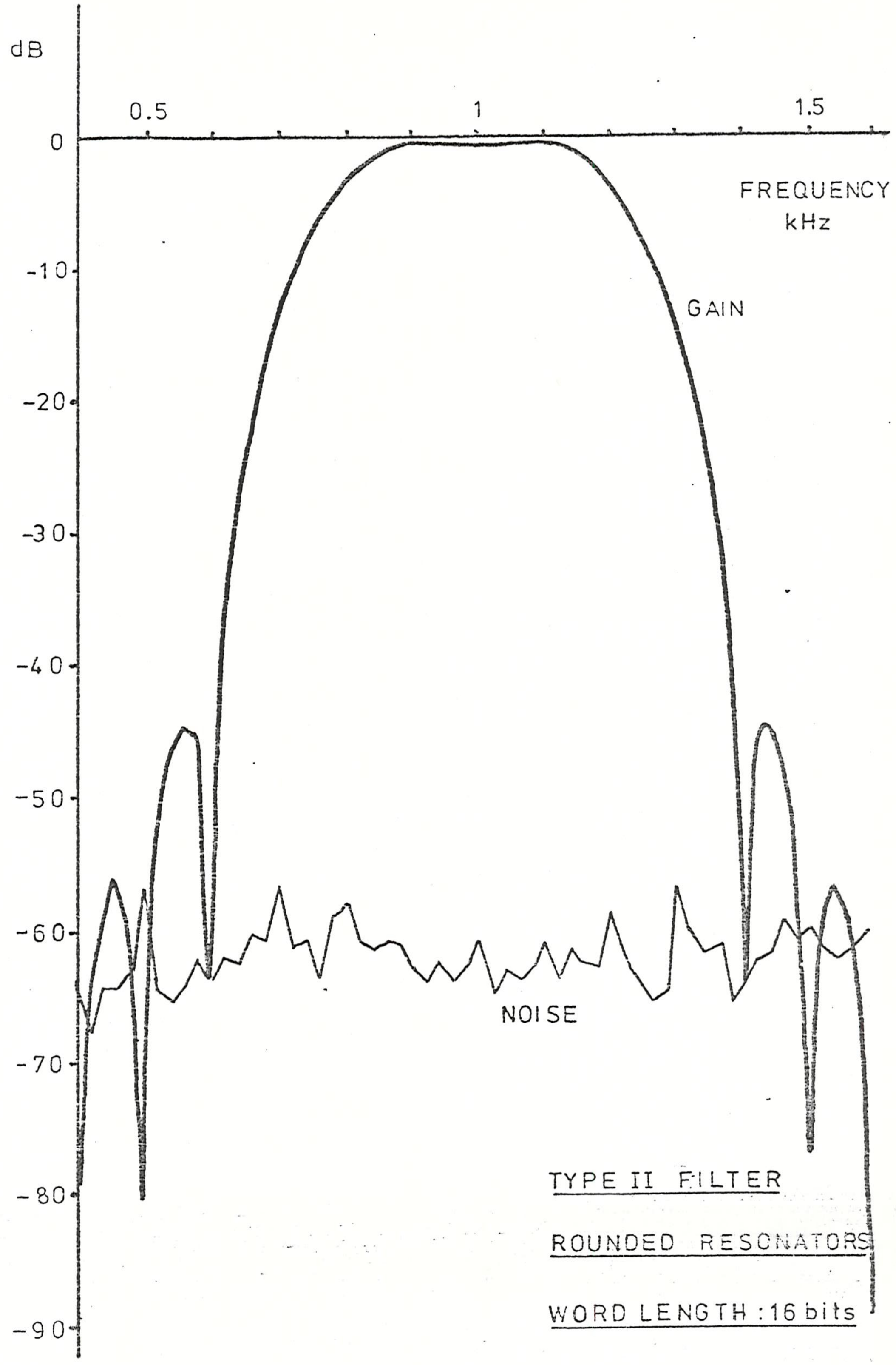


Figure 8.8 Best-fit Frequency Response and Noise Output

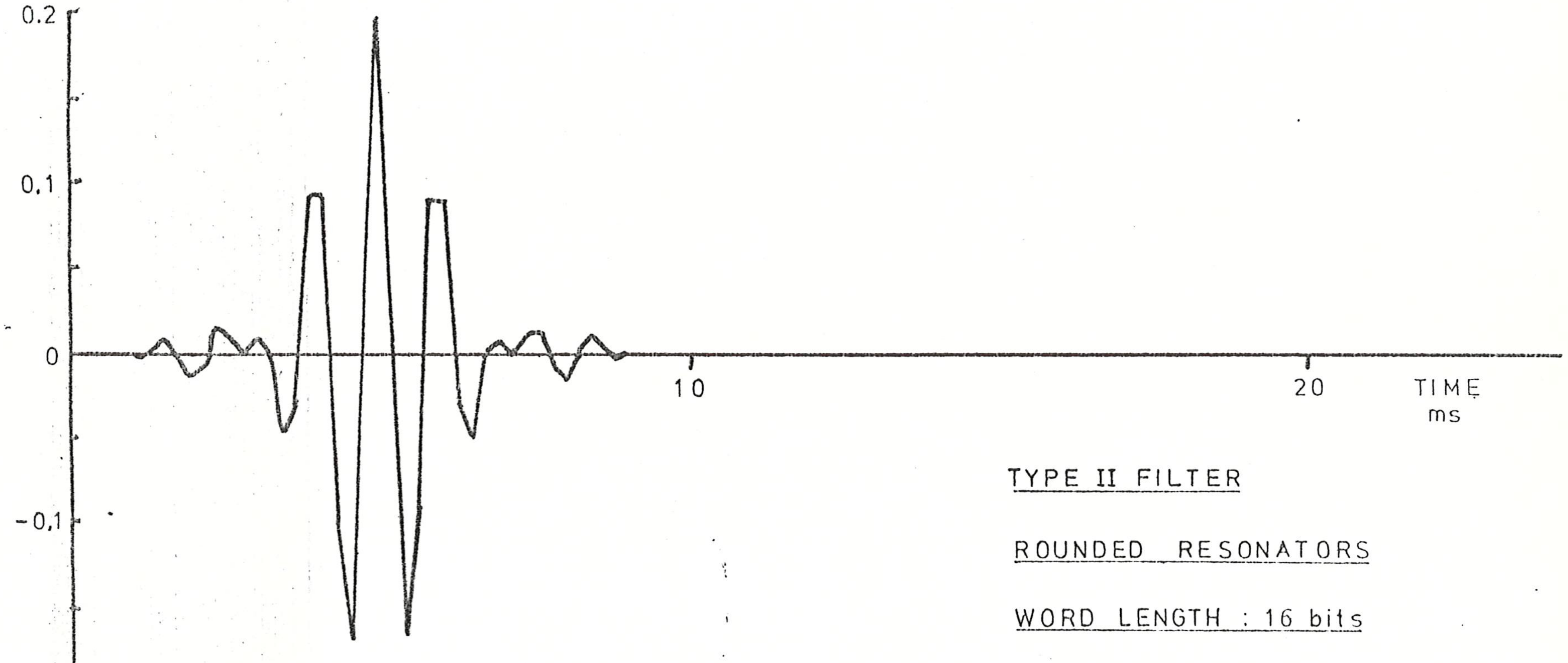


Figure 8.9 Impulse Response

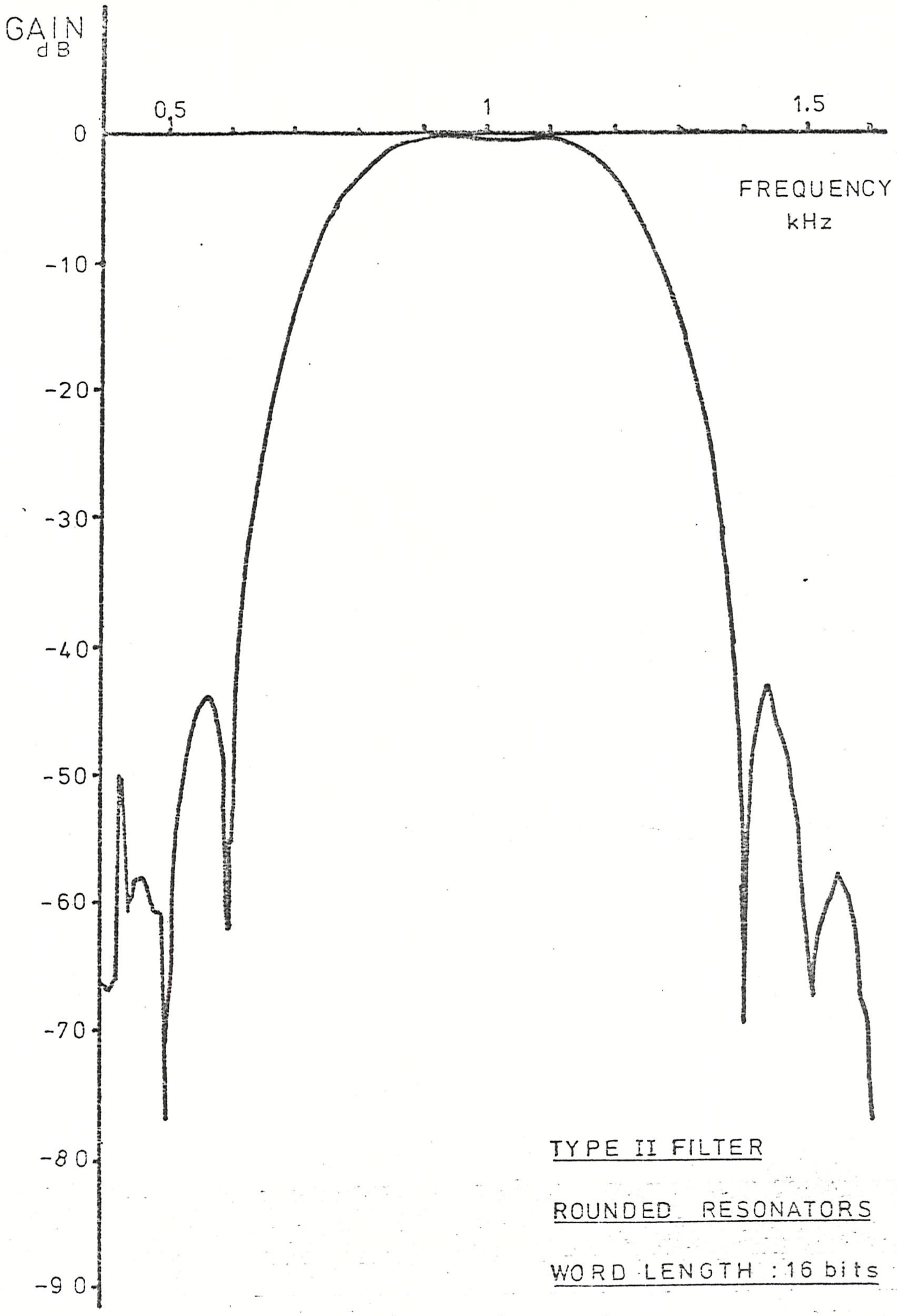


Figure 8.10 FFT-derived Frequency Response

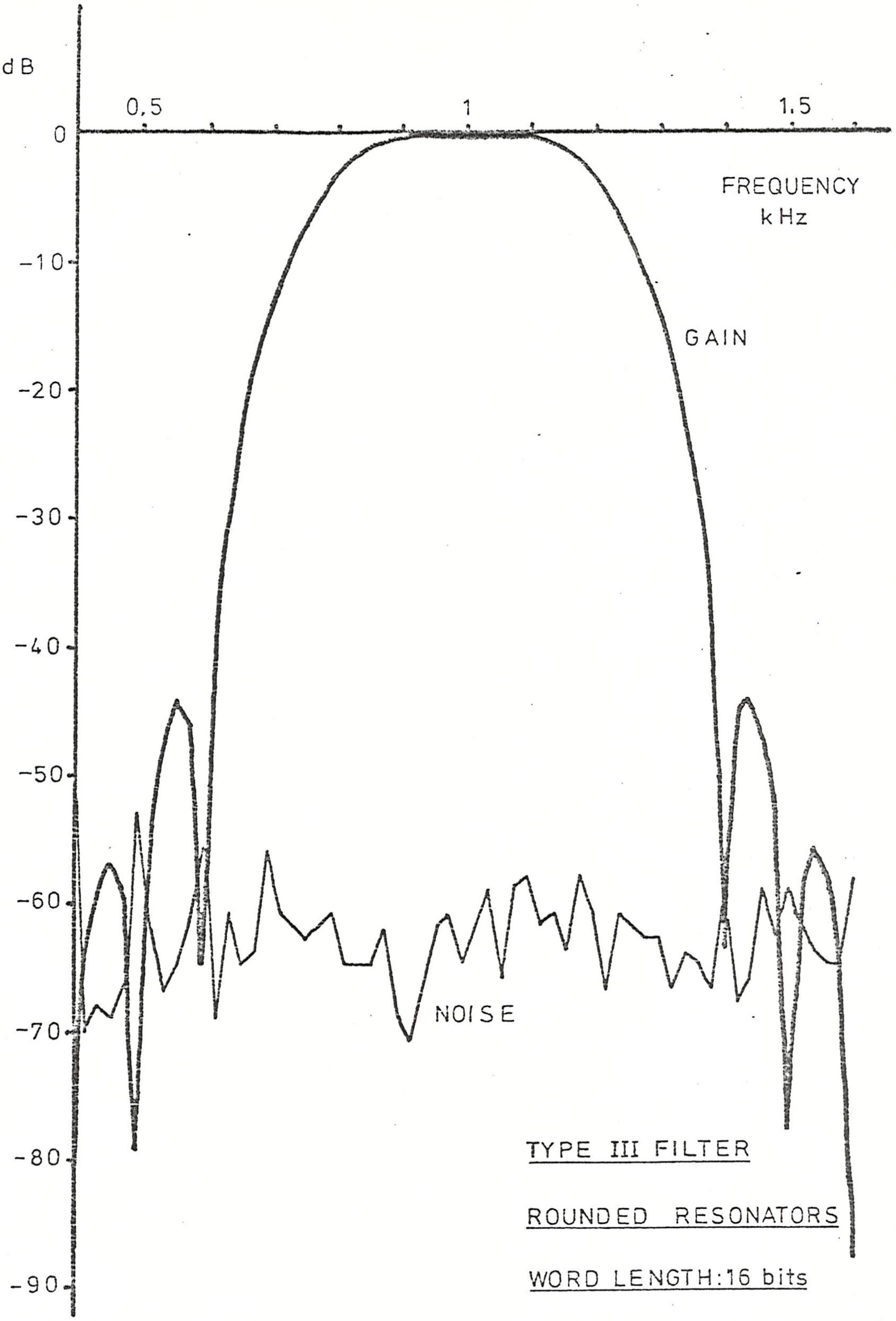


Figure 8.11 Best-fit Frequency Response and Noise Output

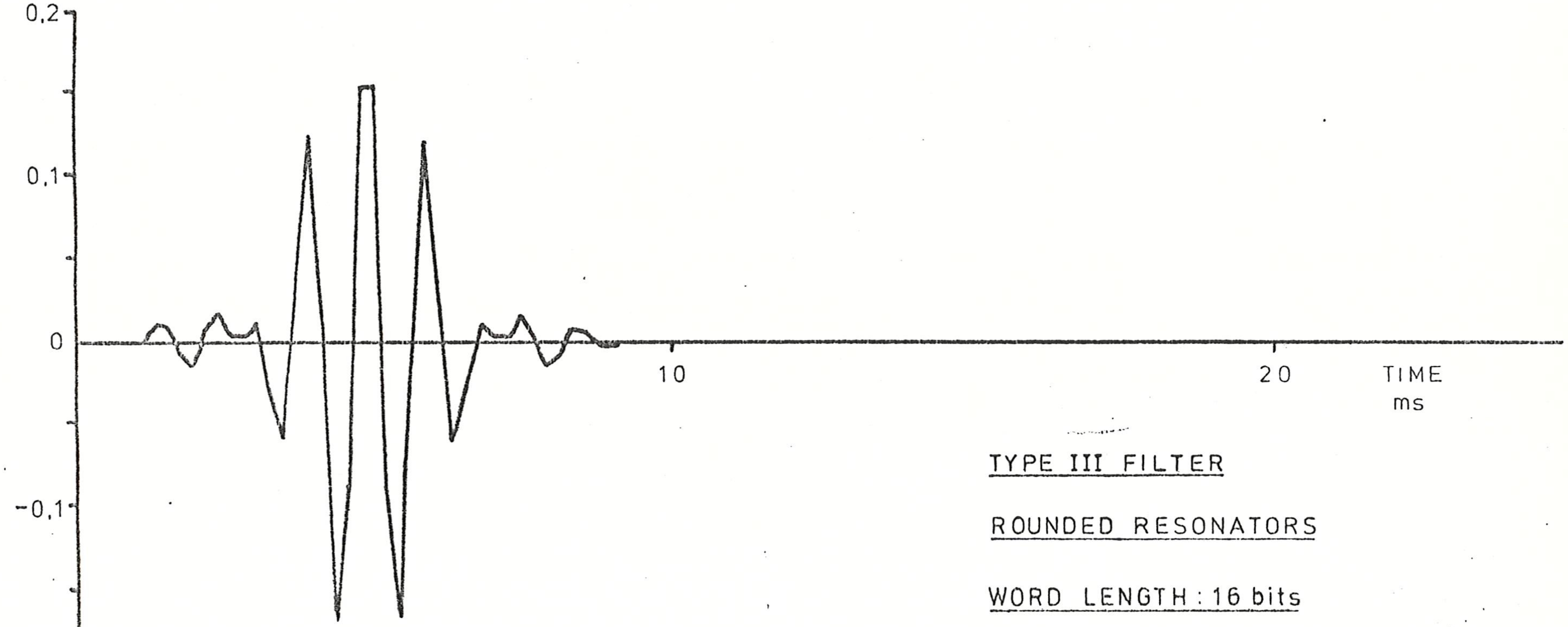


Figure 8.12 Impulse Response

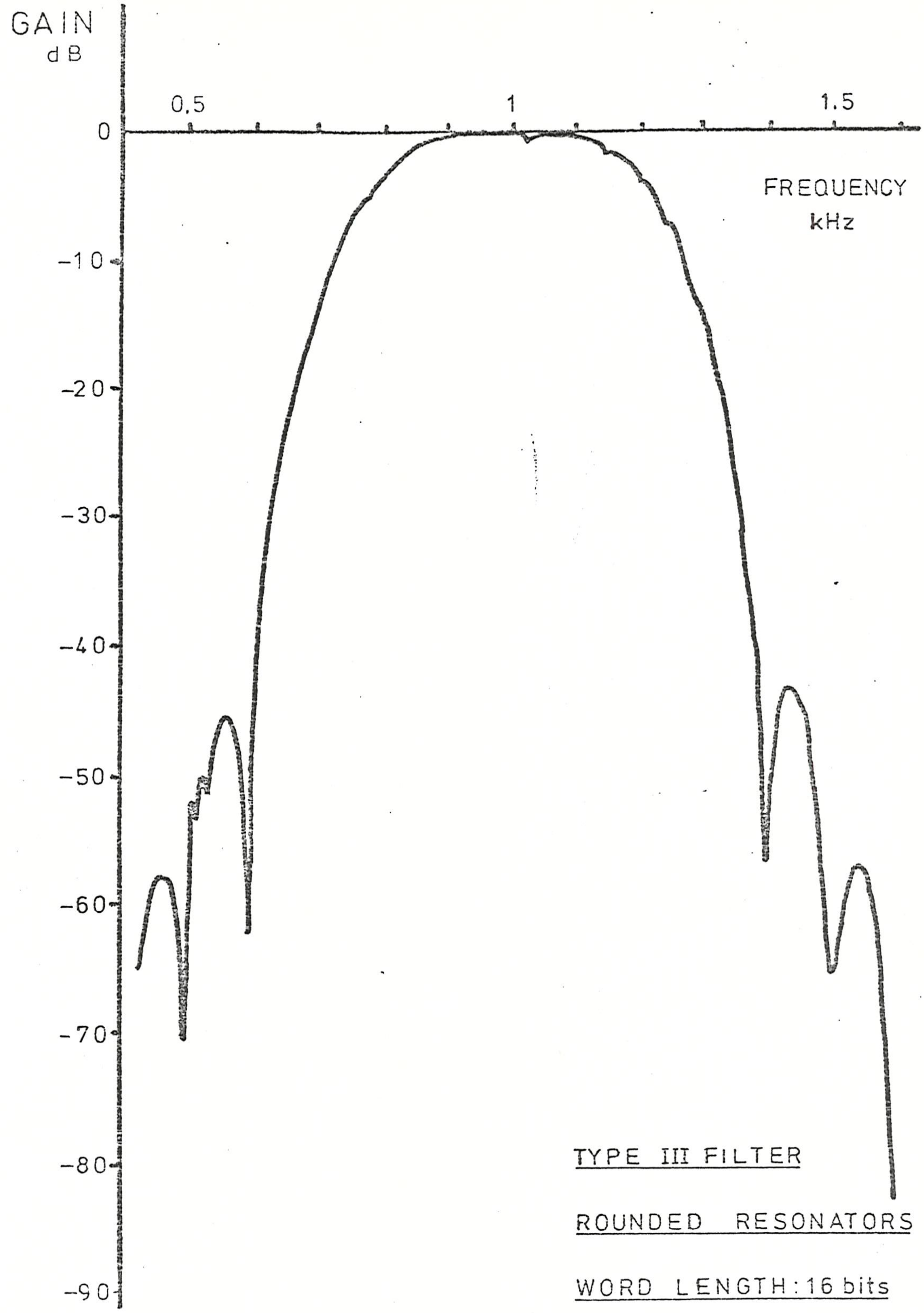


Figure 8.13 FFT-derived Frequency Response

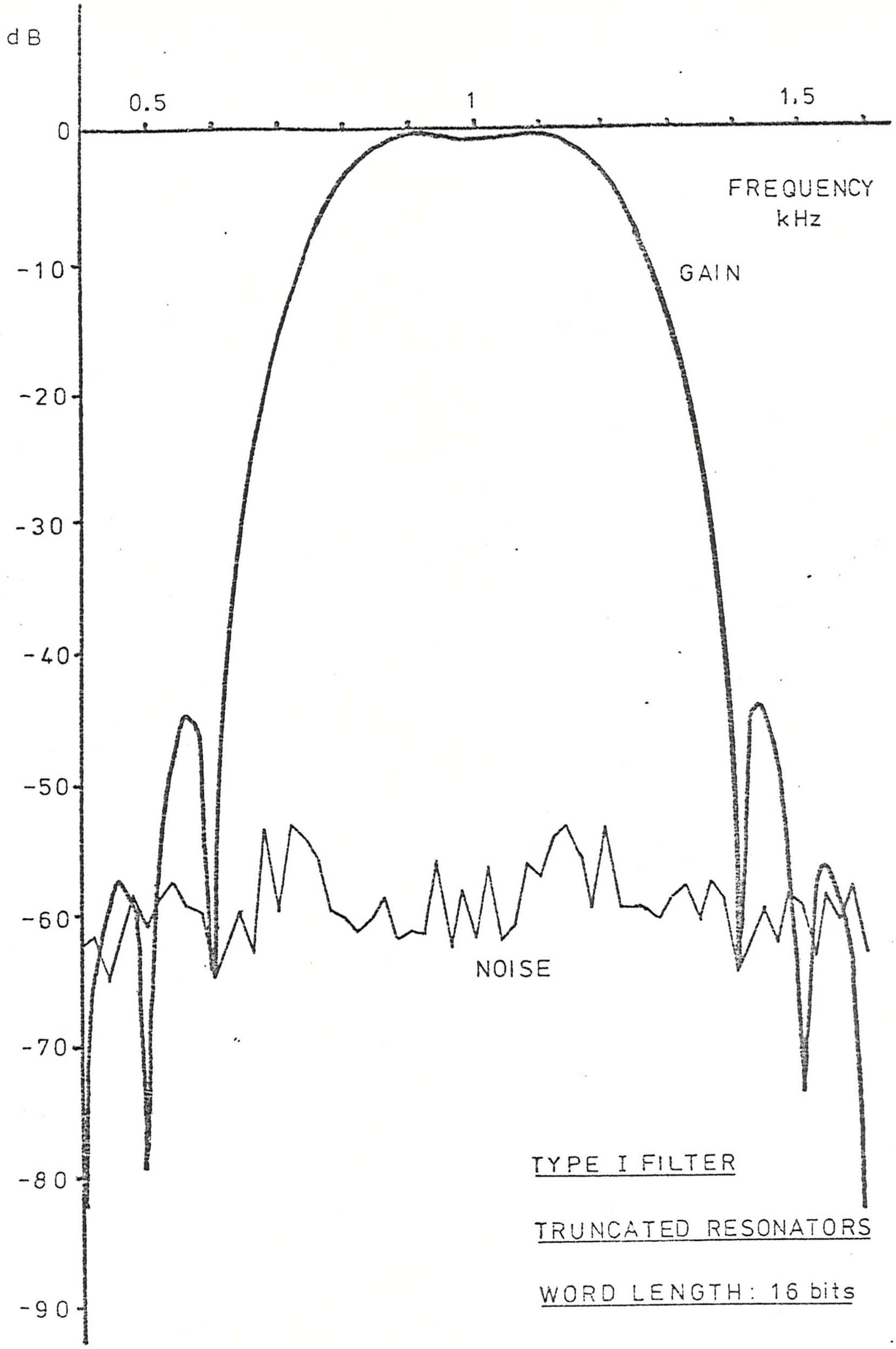


Figure 8.14 Best-fit Frequency Response and Noise Output

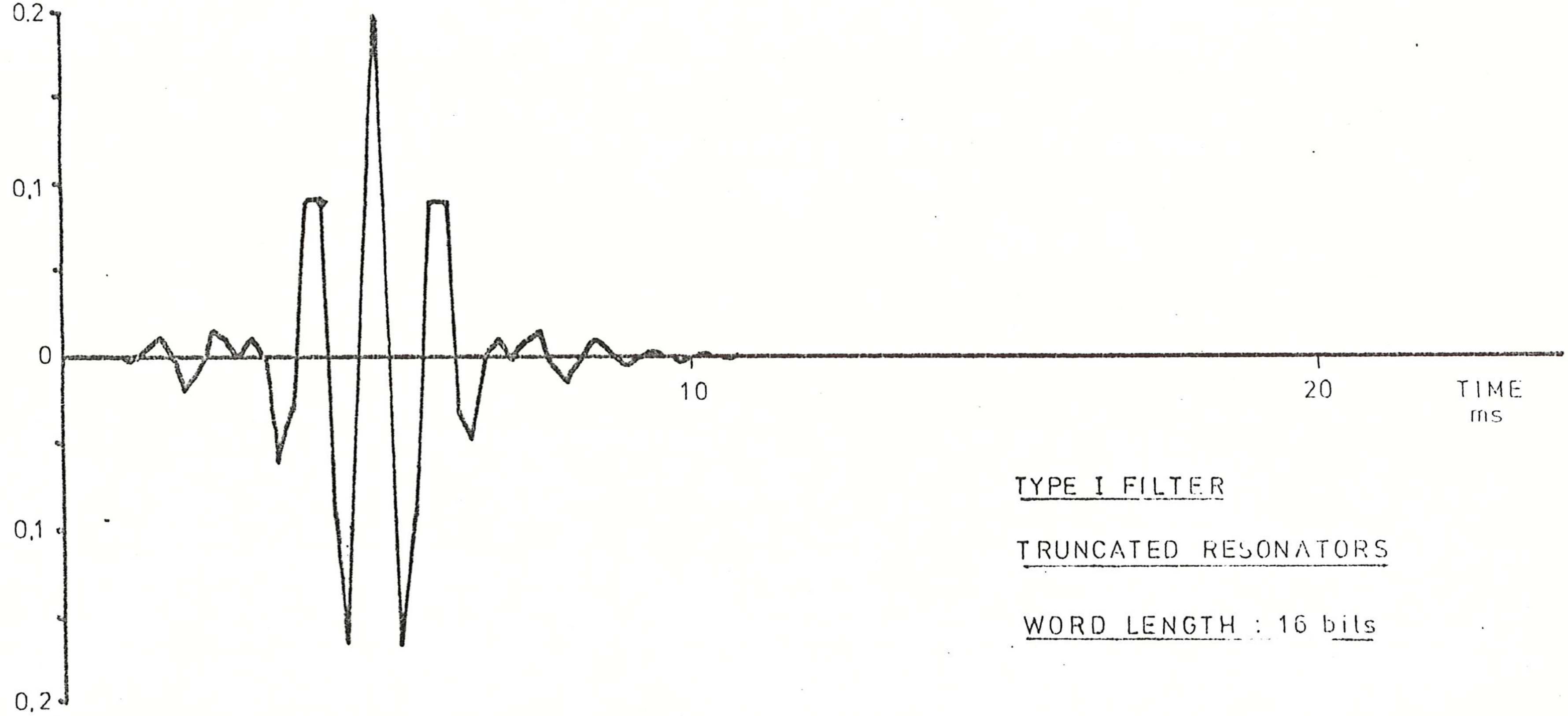


Figure 8.15 Impulse Response

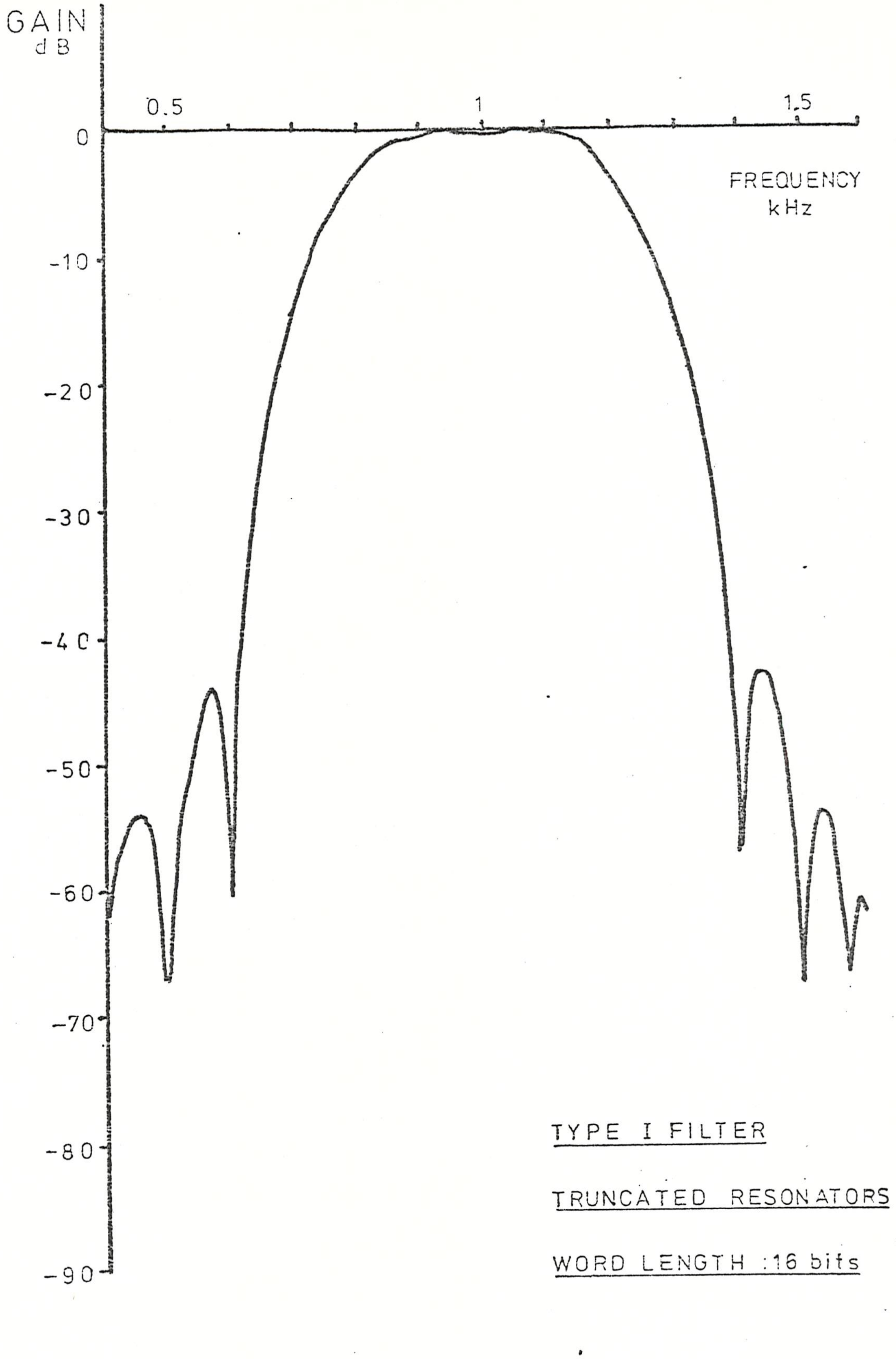


Figure 8.16 FFT-derived Frequency Response

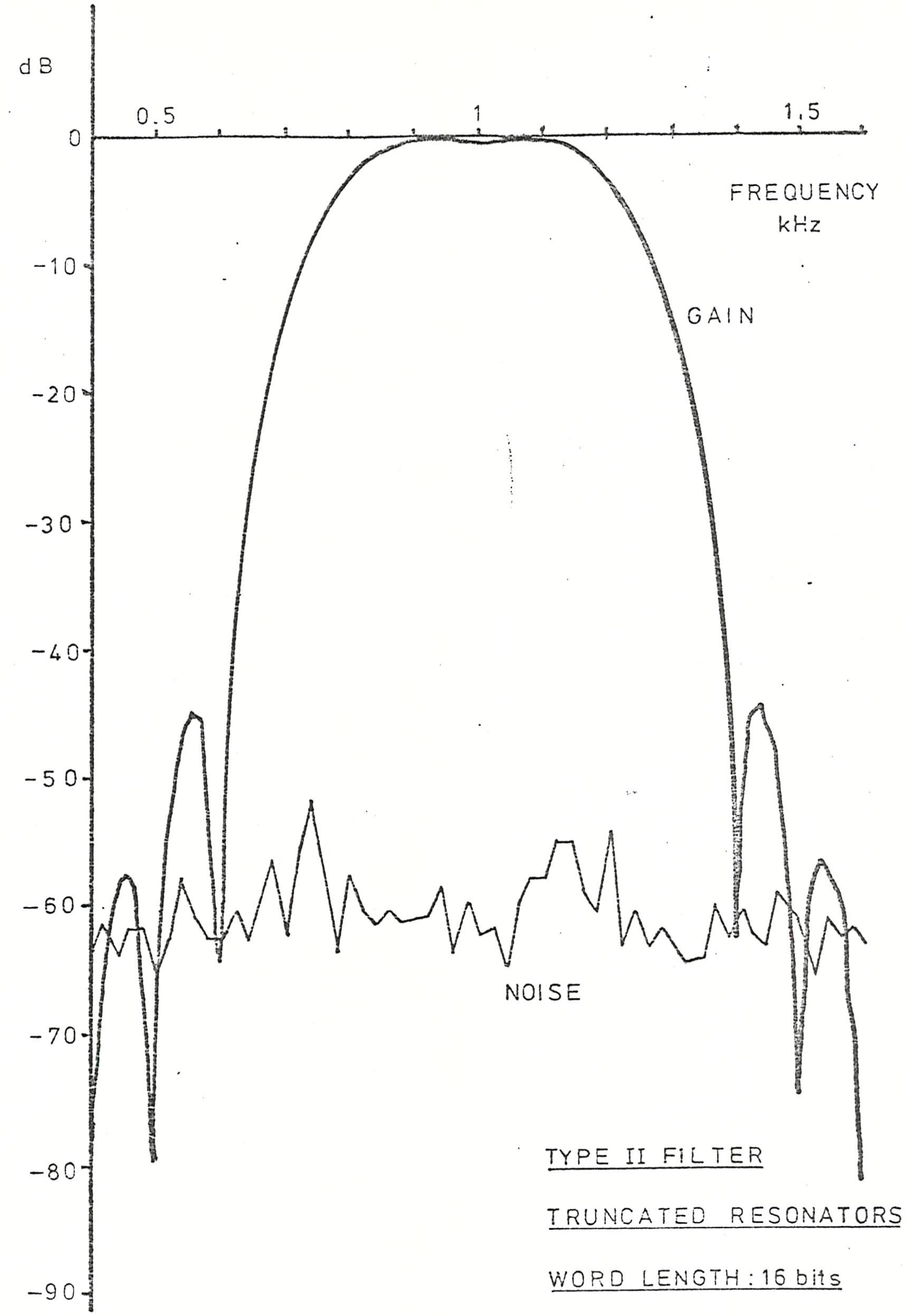


Figure 8.17 Best-fit Frequency Response and Noise Output

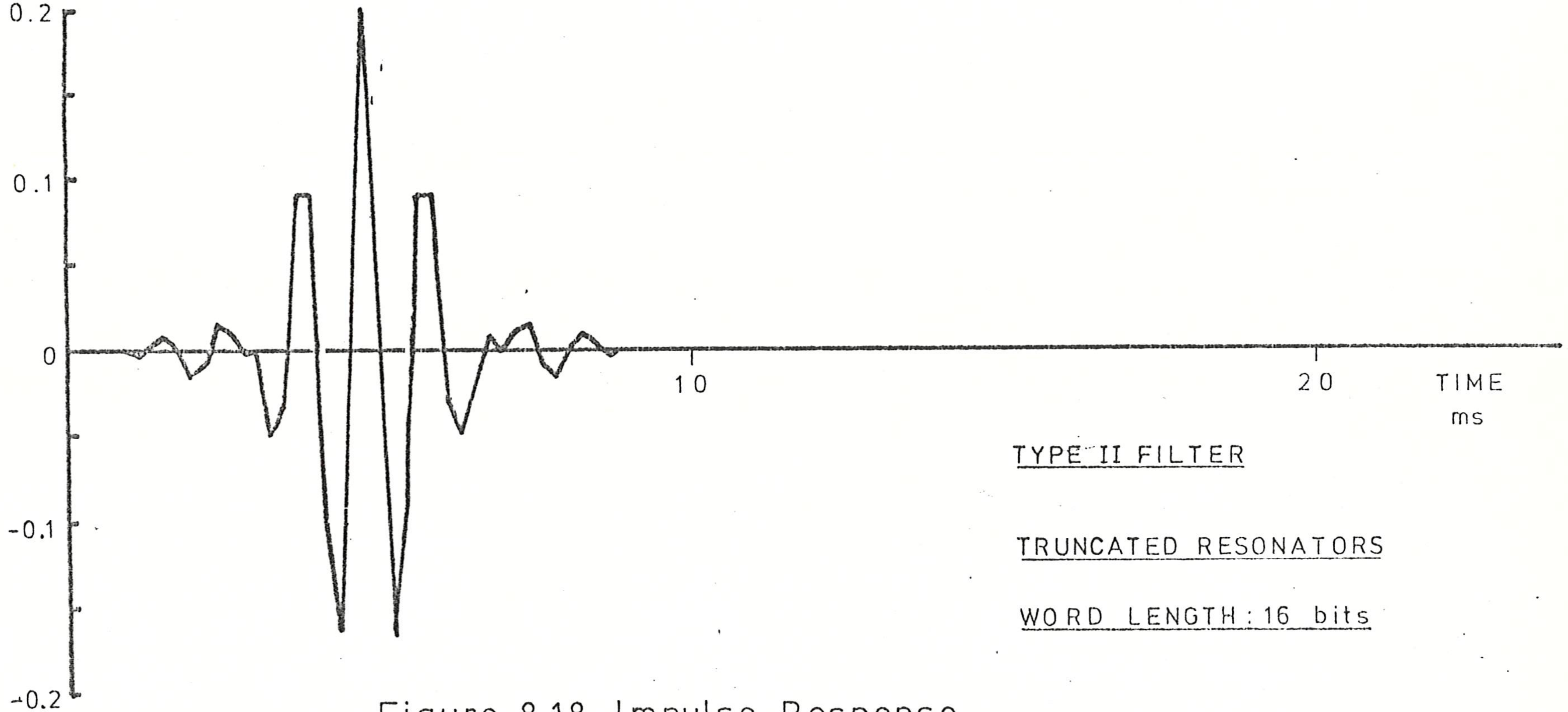


Figure 8.18 Impulse Response

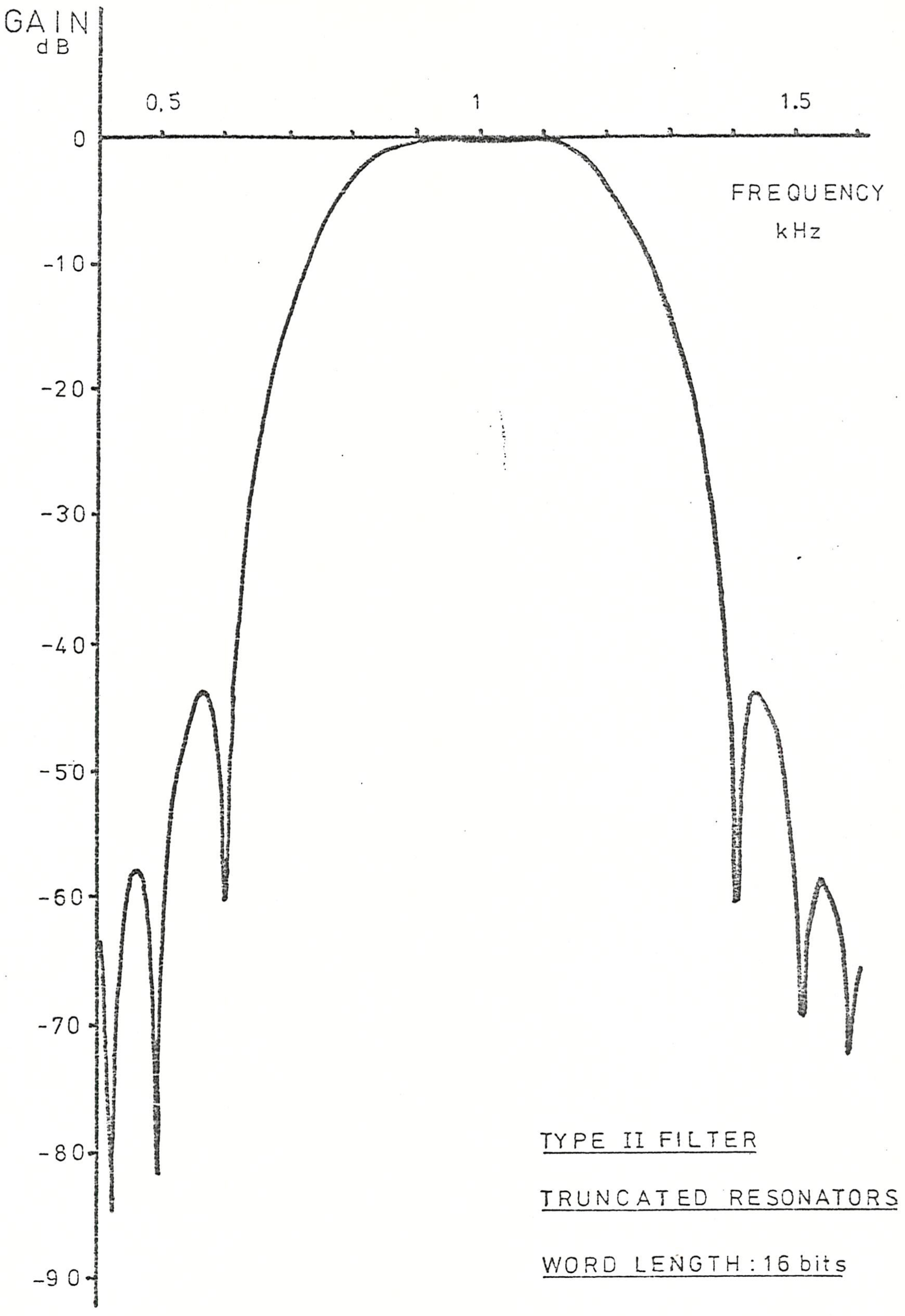


Figure 8.19 FFT-derived Frequency Response

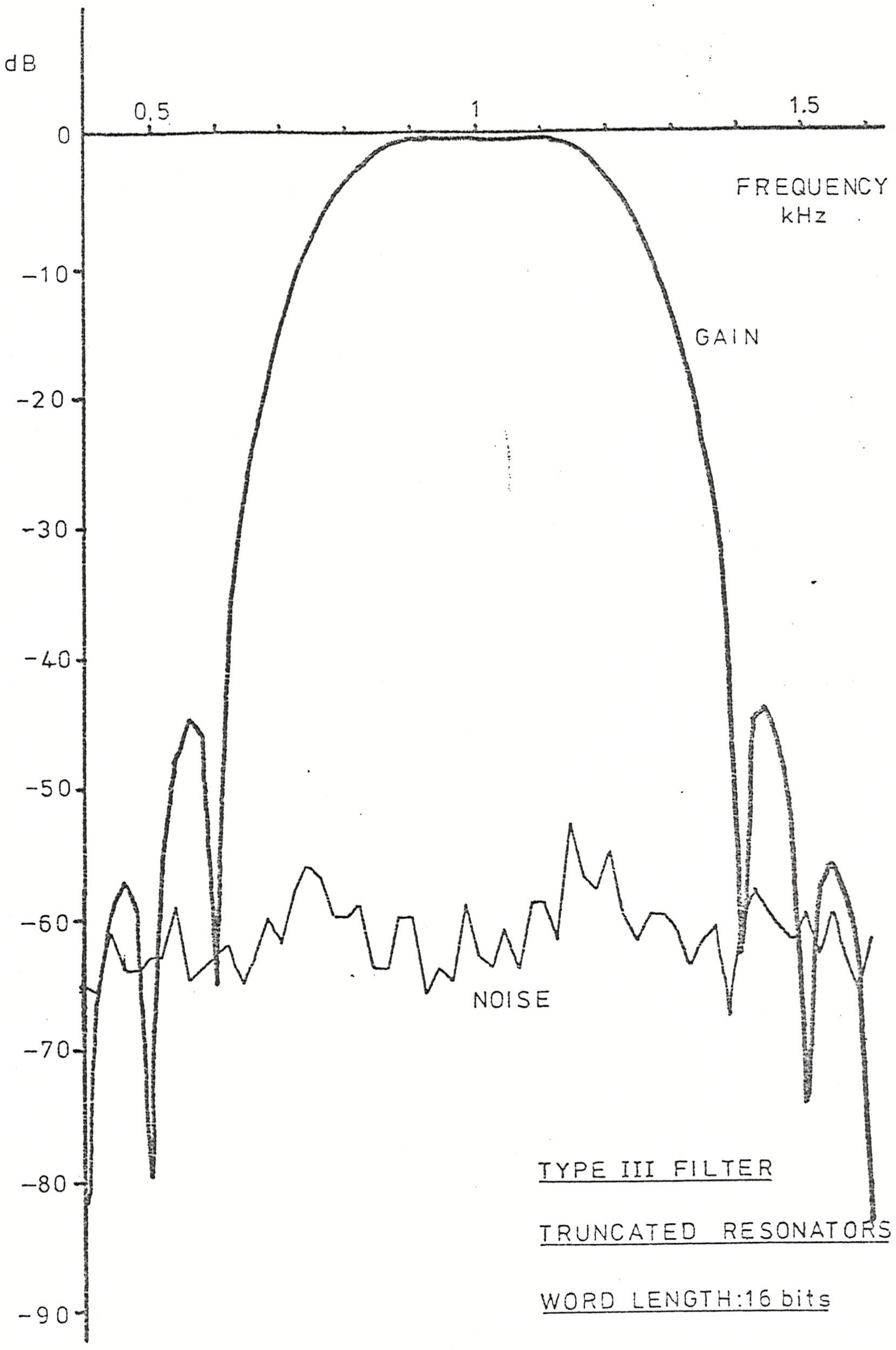


Figure 8.20 Best-fit Frequency Response and Noise Output

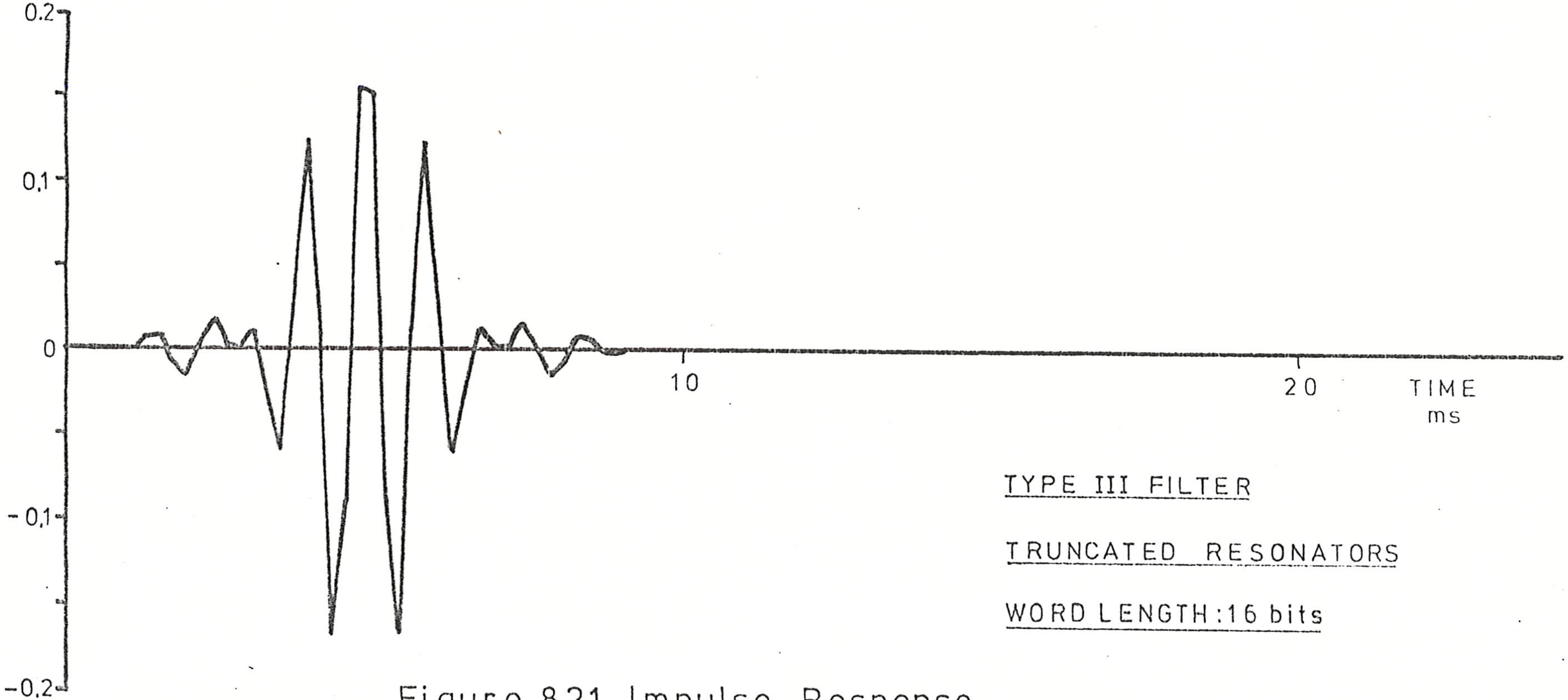


Figure 8.21 Impulse Response

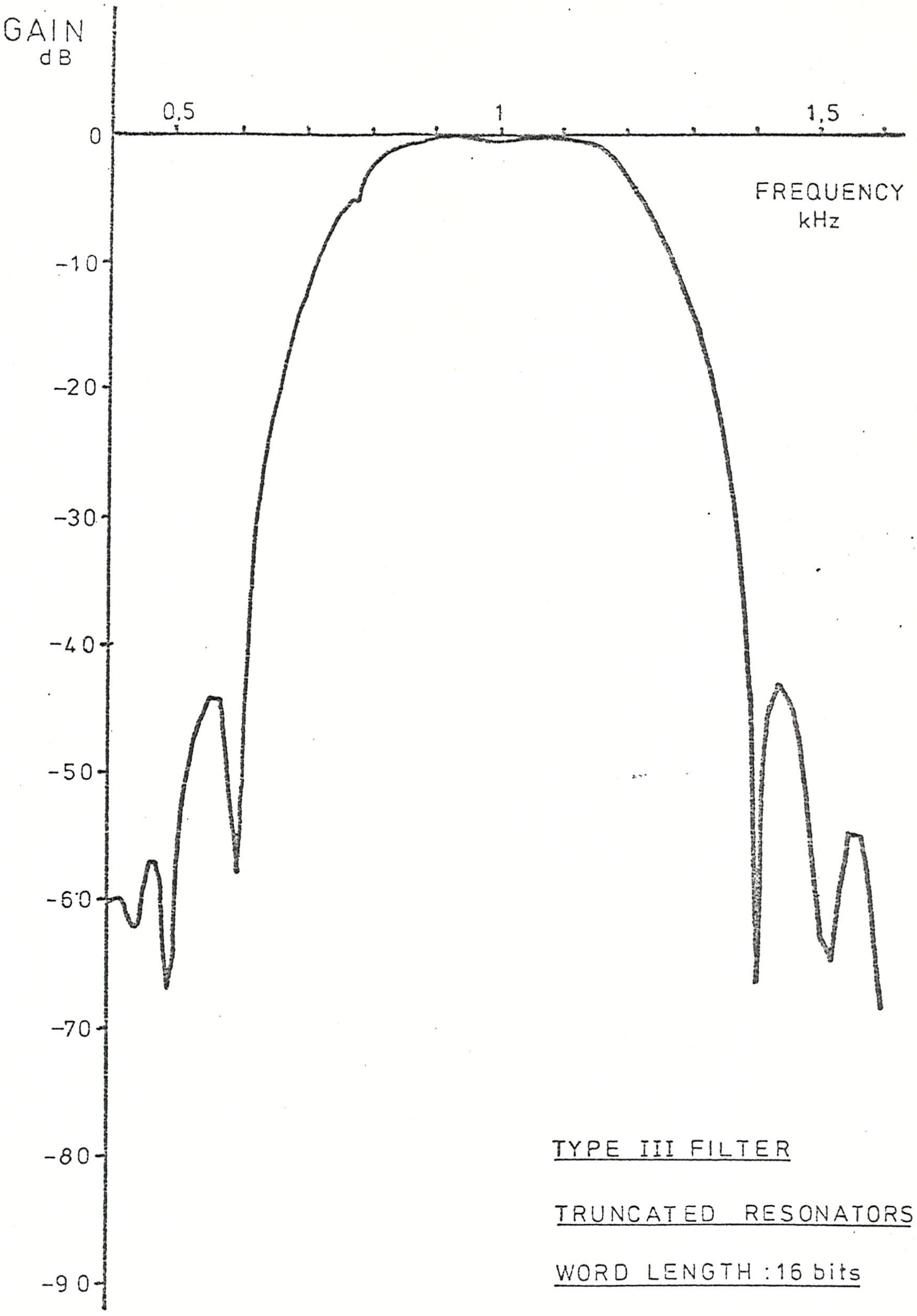


Figure 8.22 FFT-derived Frequency Response

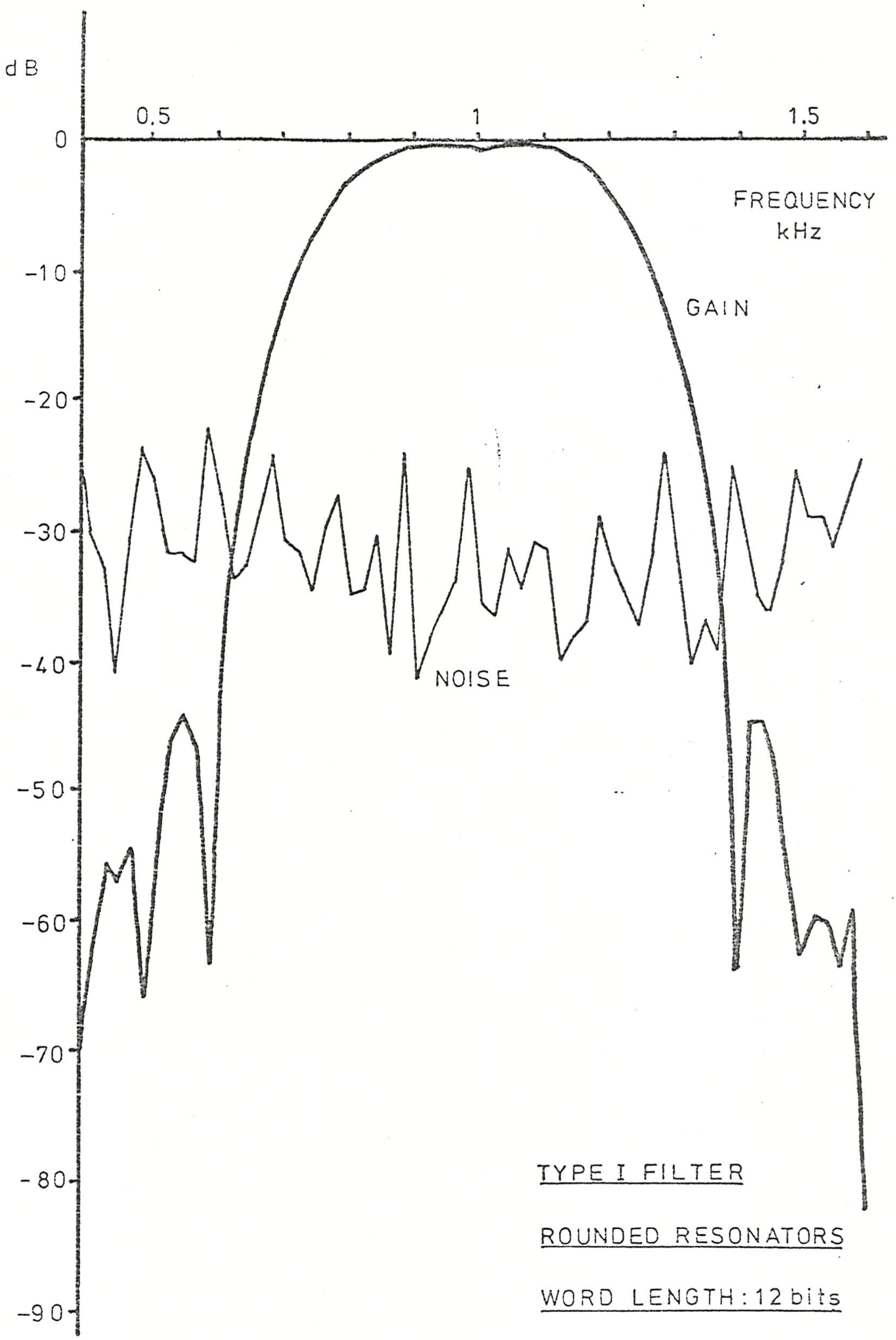


Figure 8.23 Best-fit Frequency Response and Noise Output

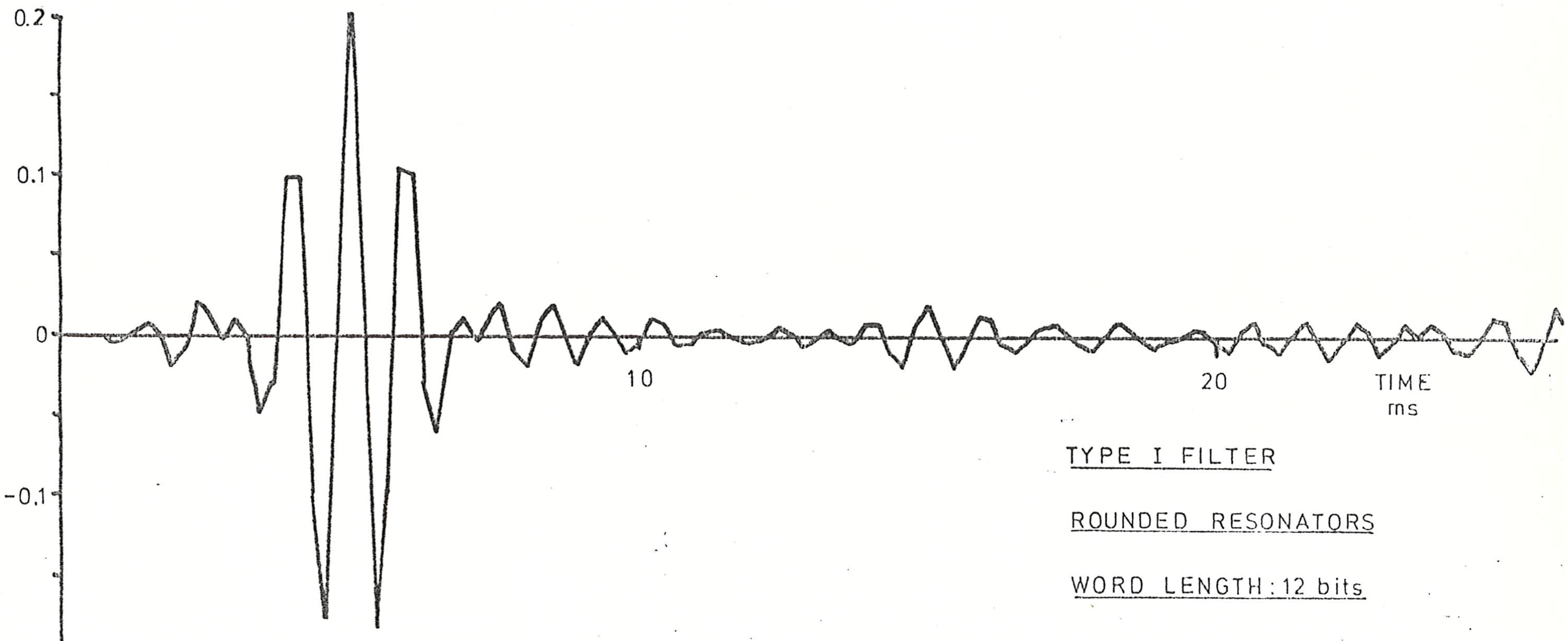


Figure 8.24 Impulse Response

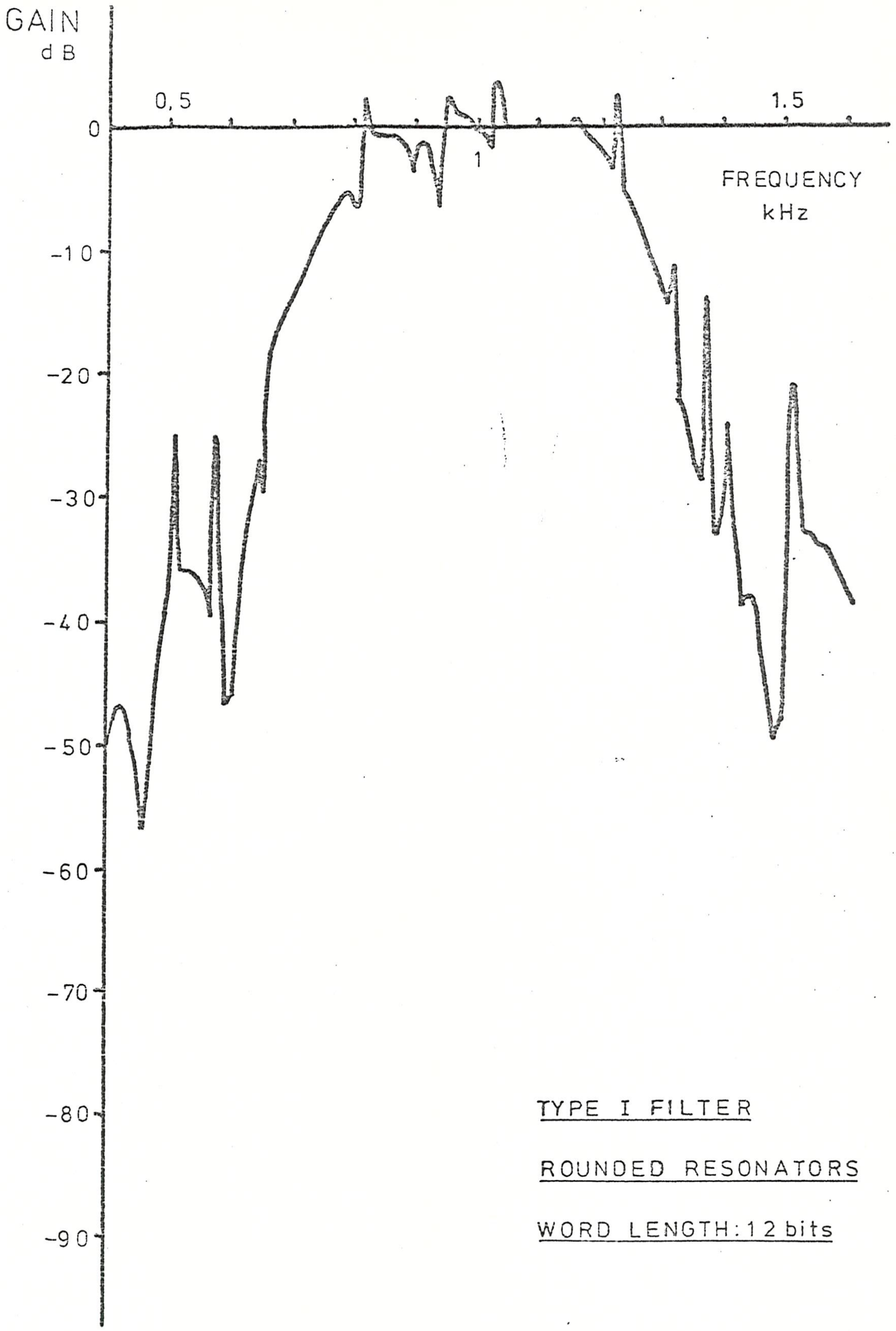


Figure 8.25 FFT-derived Frequency Response

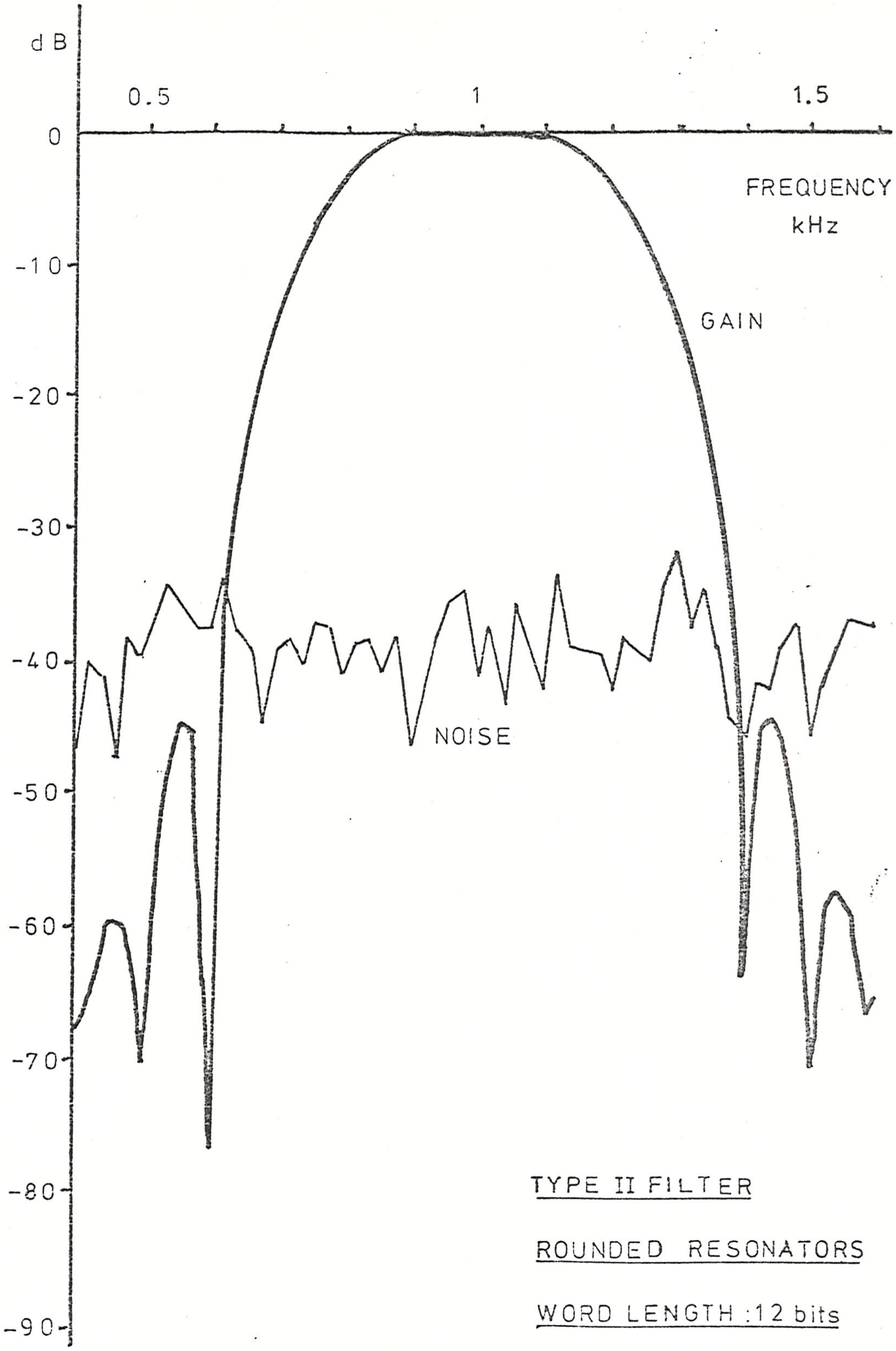


Figure 8.26 Best-fit Frequency Response and Noise Output

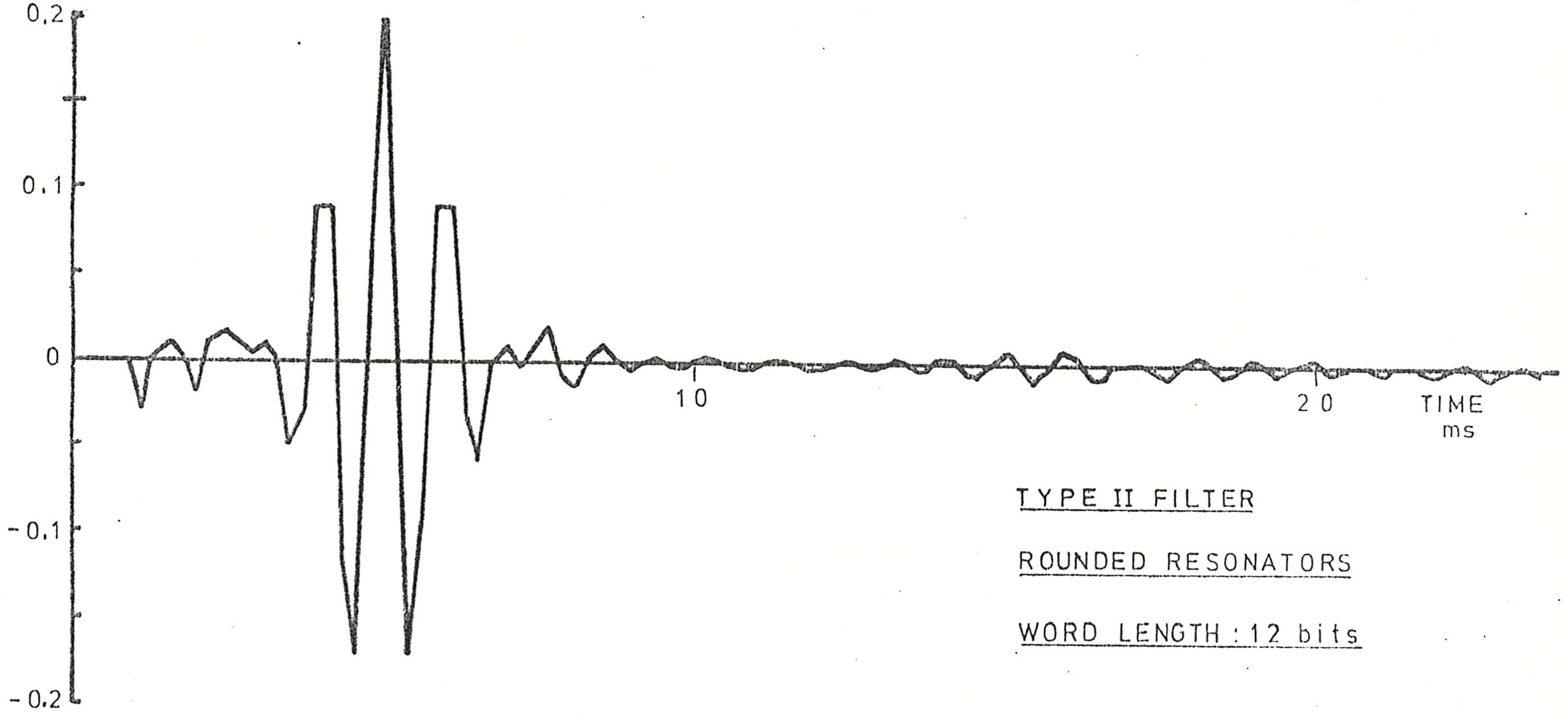


Figure 8.27 Impulse Response

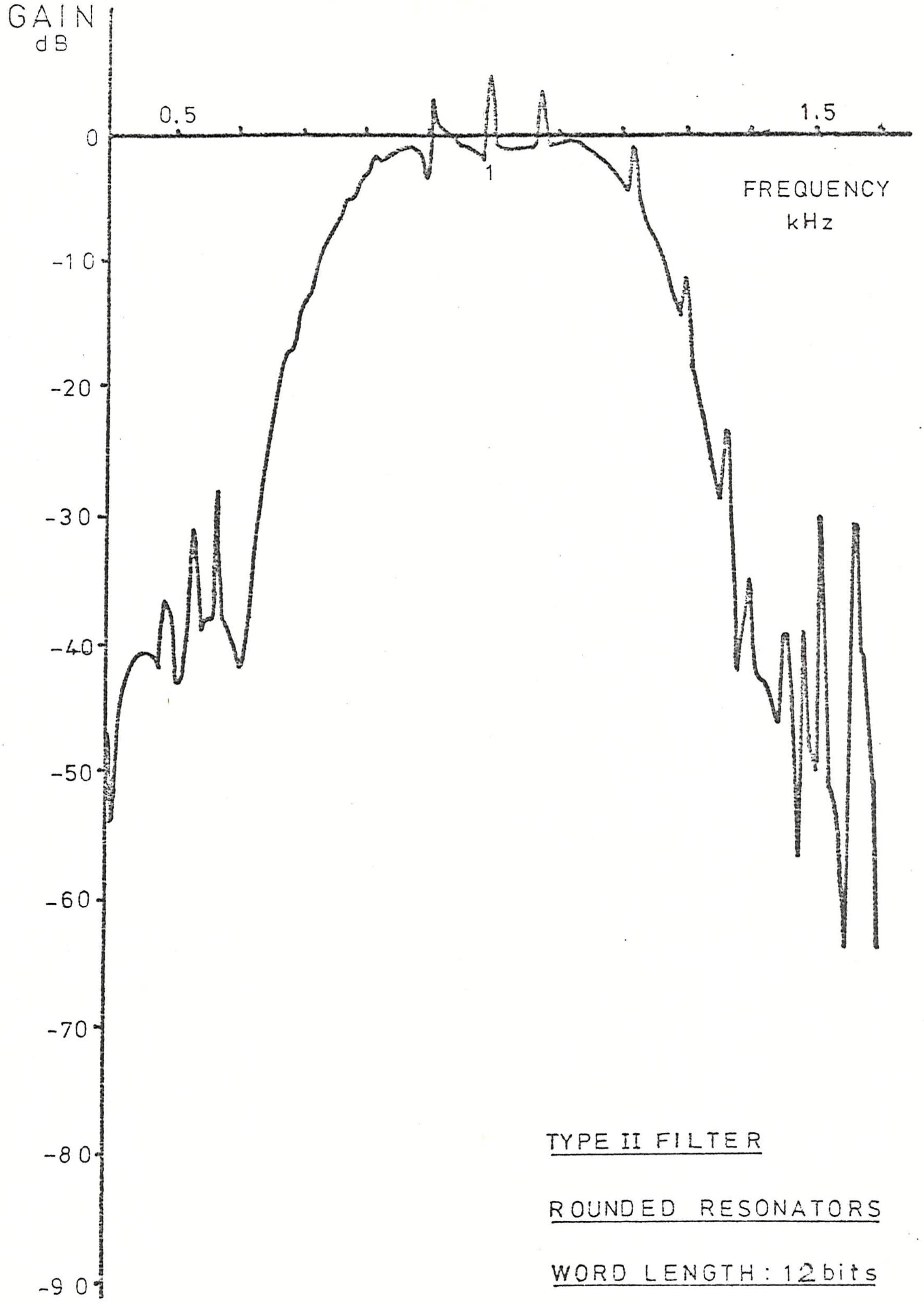


Figure 8.28 FFT-derived Frequency Response

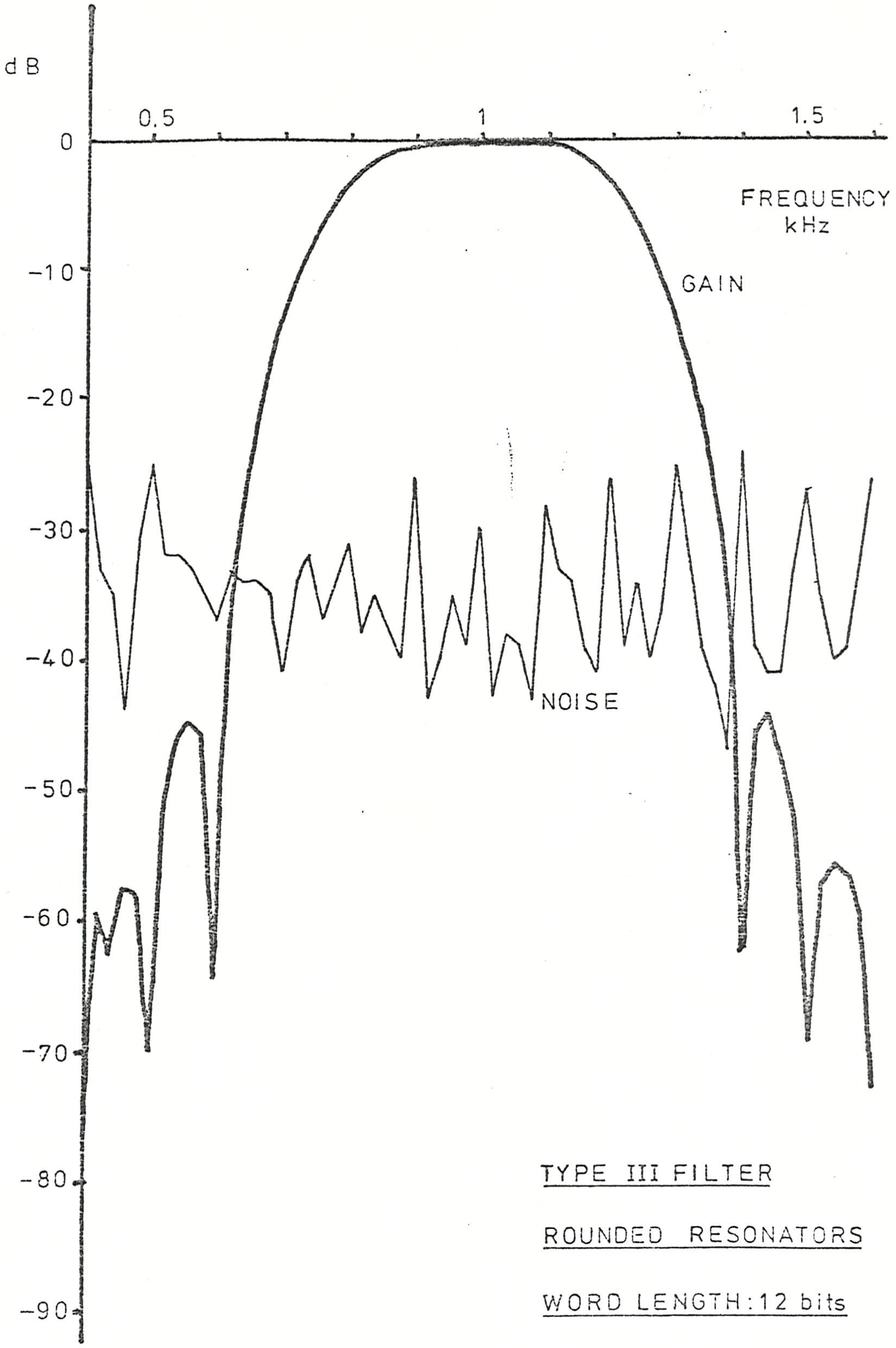


Figure 8.29 Best-fit Frequency Response and Noise Output

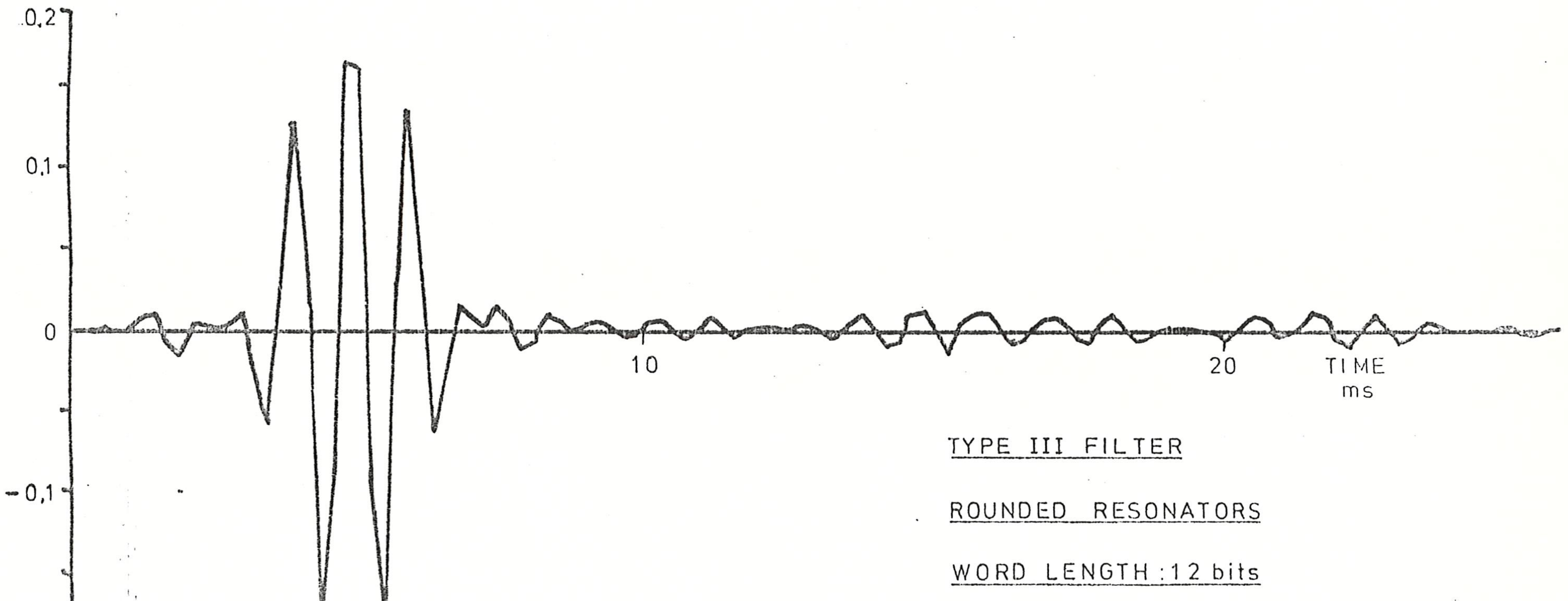


Figure 8.30 Impulse Response

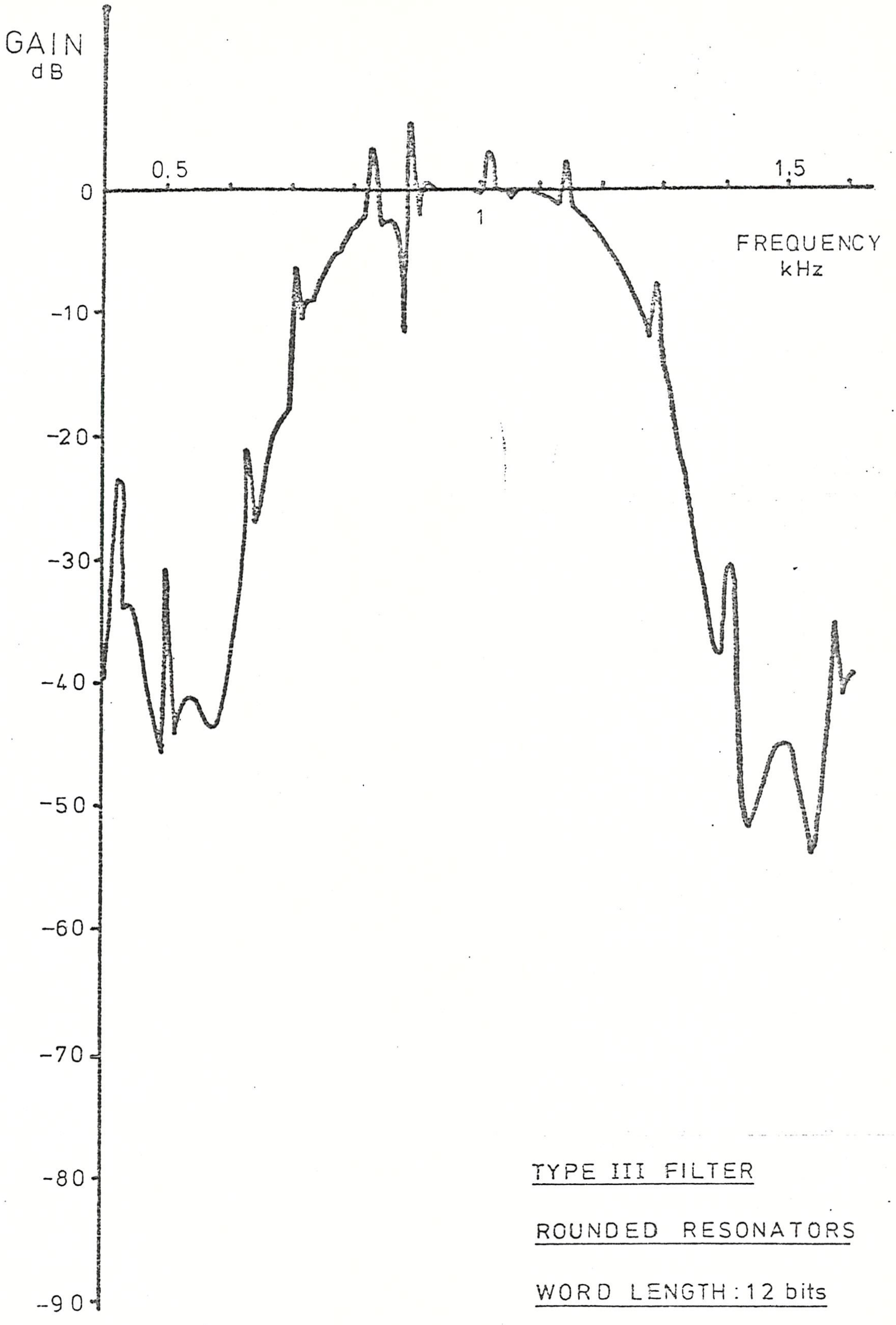


Figure 8.31 FFT-derived Frequency Response

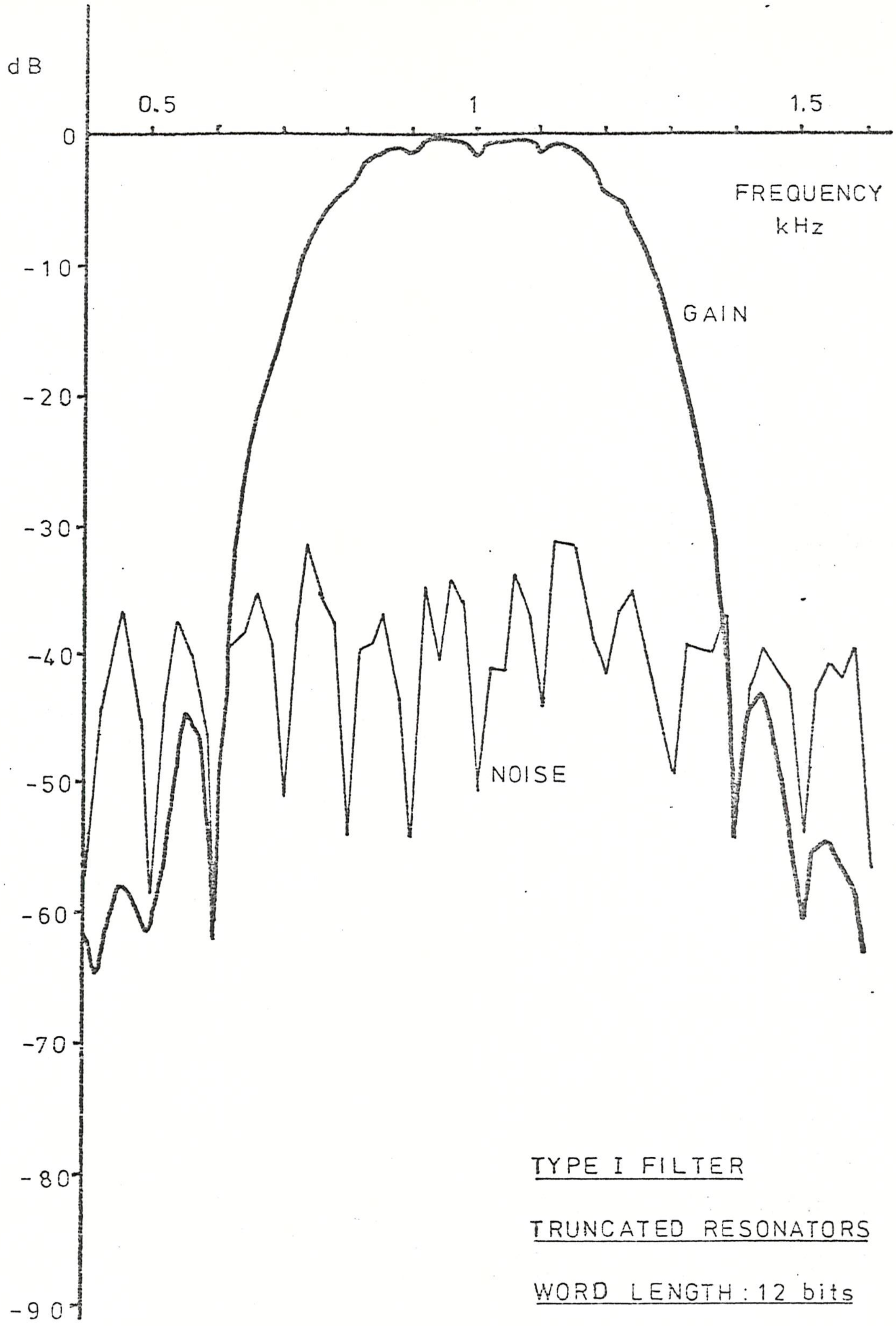


Figure 8.32 Best-fit Frequency Response and Noise Output

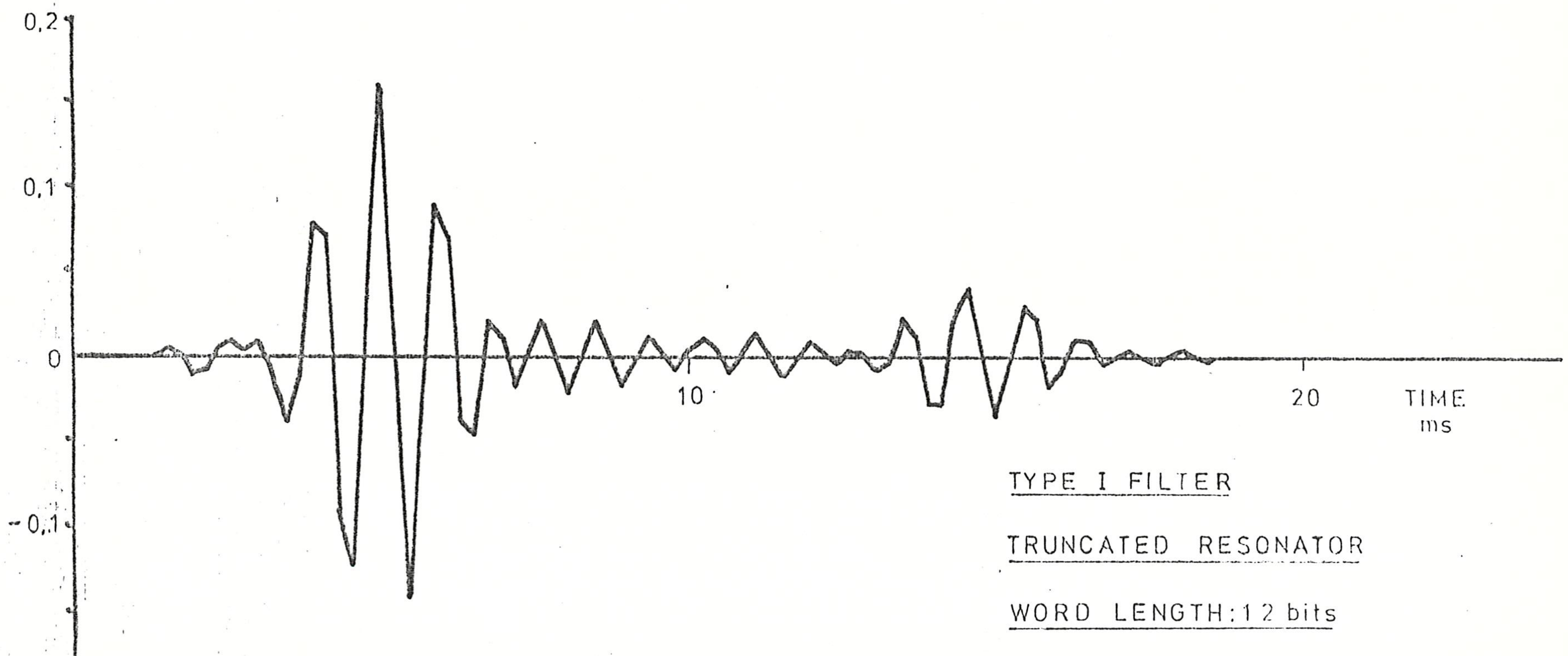


Figure 8,33 Impulse Response

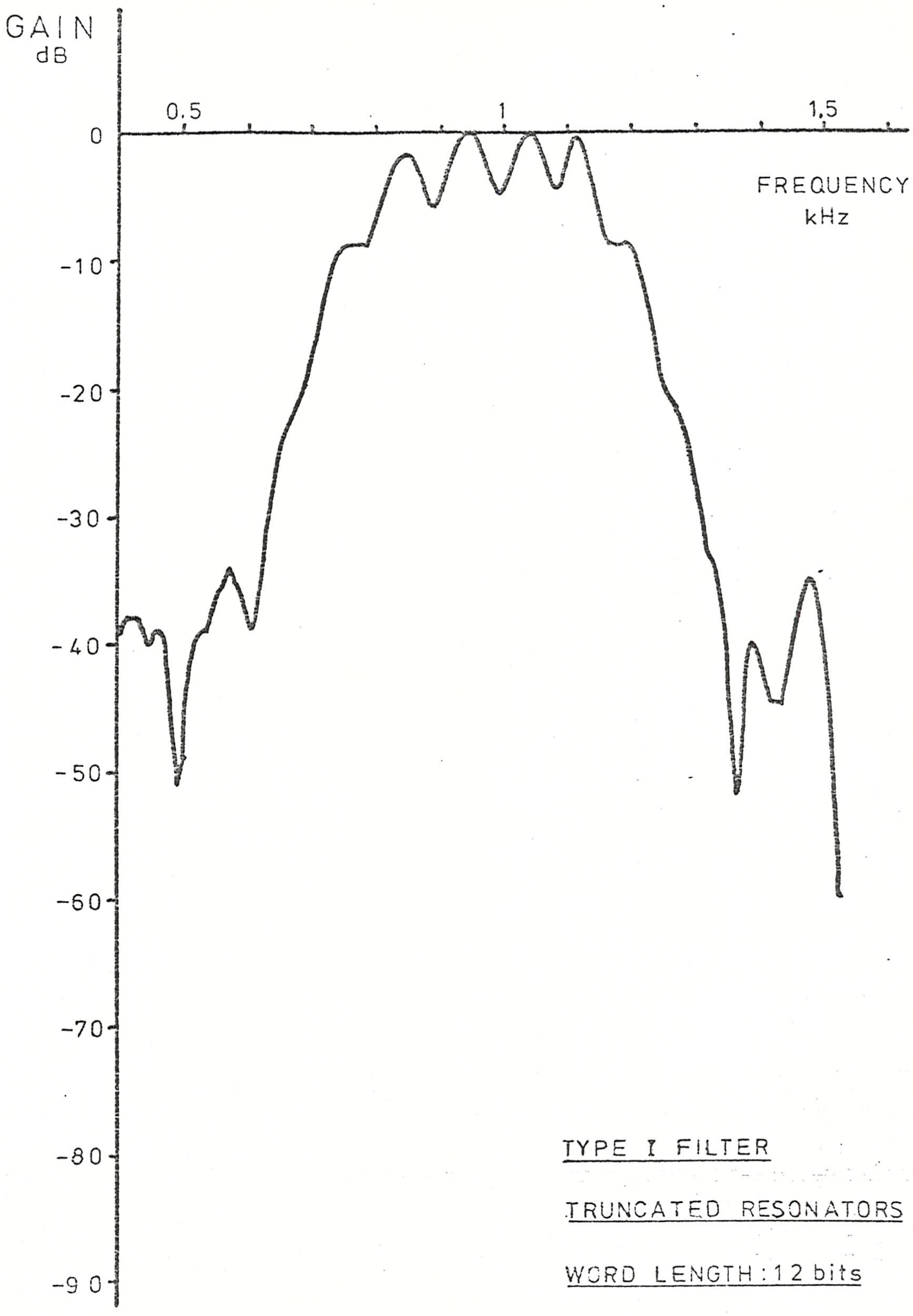


Figure 8.34 FFT-derived Frequency Response

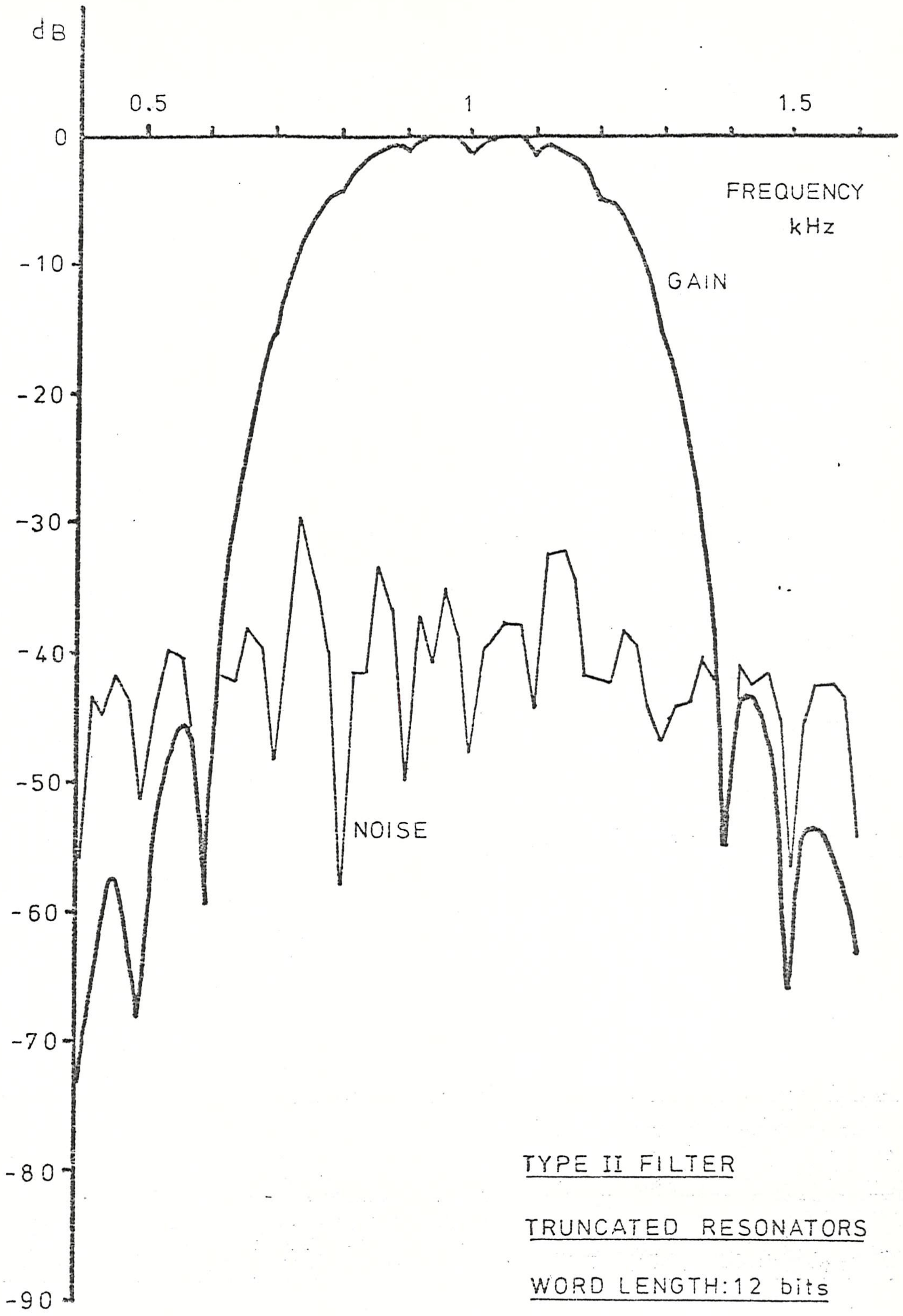


Figure 8.35 Best-fit Frequency Response and Noise Output

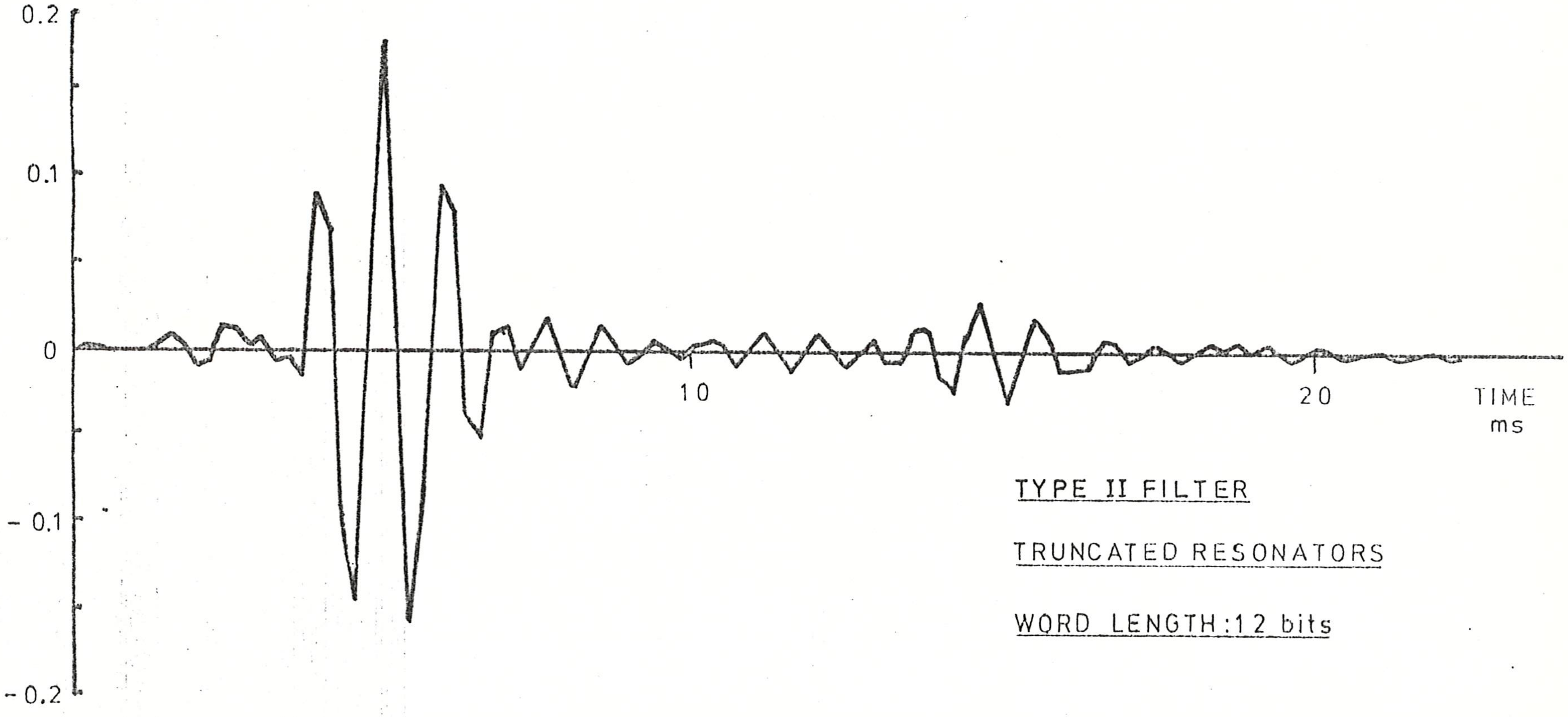


Figure 8.36 Impulse Response

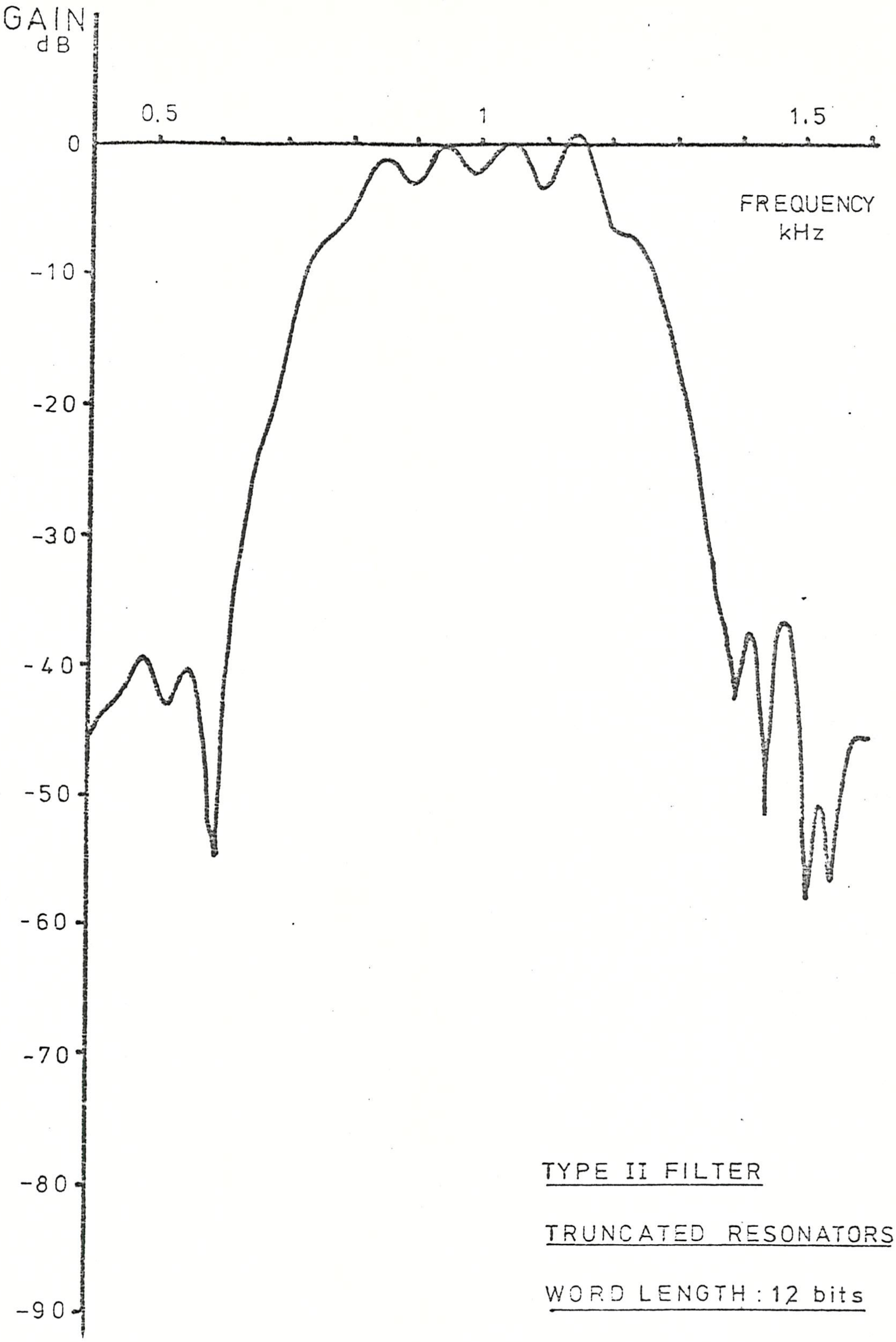


Figure 8.37 FFT-derived Frequency Response

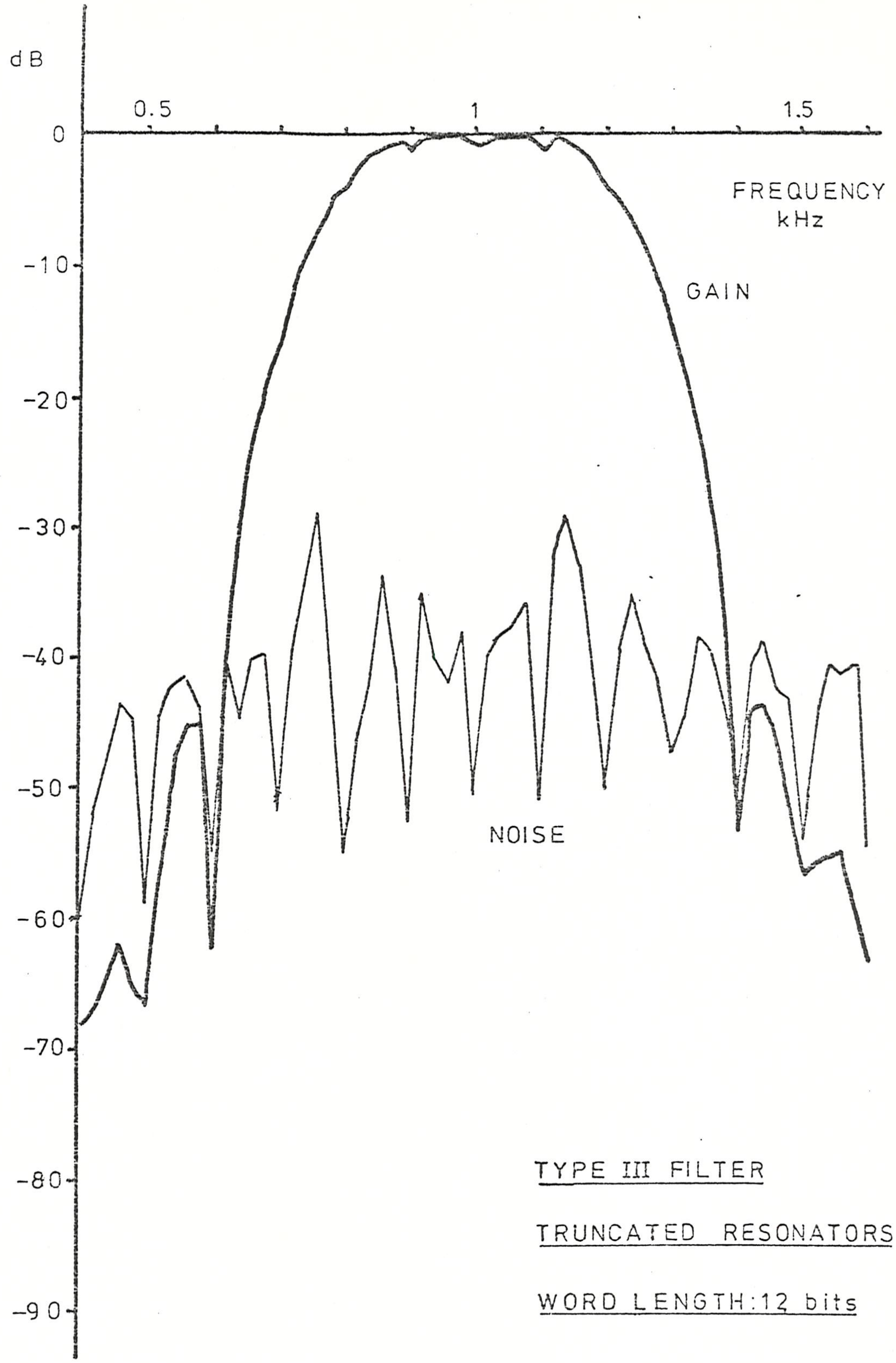


Figure 8.38 Best-fit Frequency Response and Noise Output

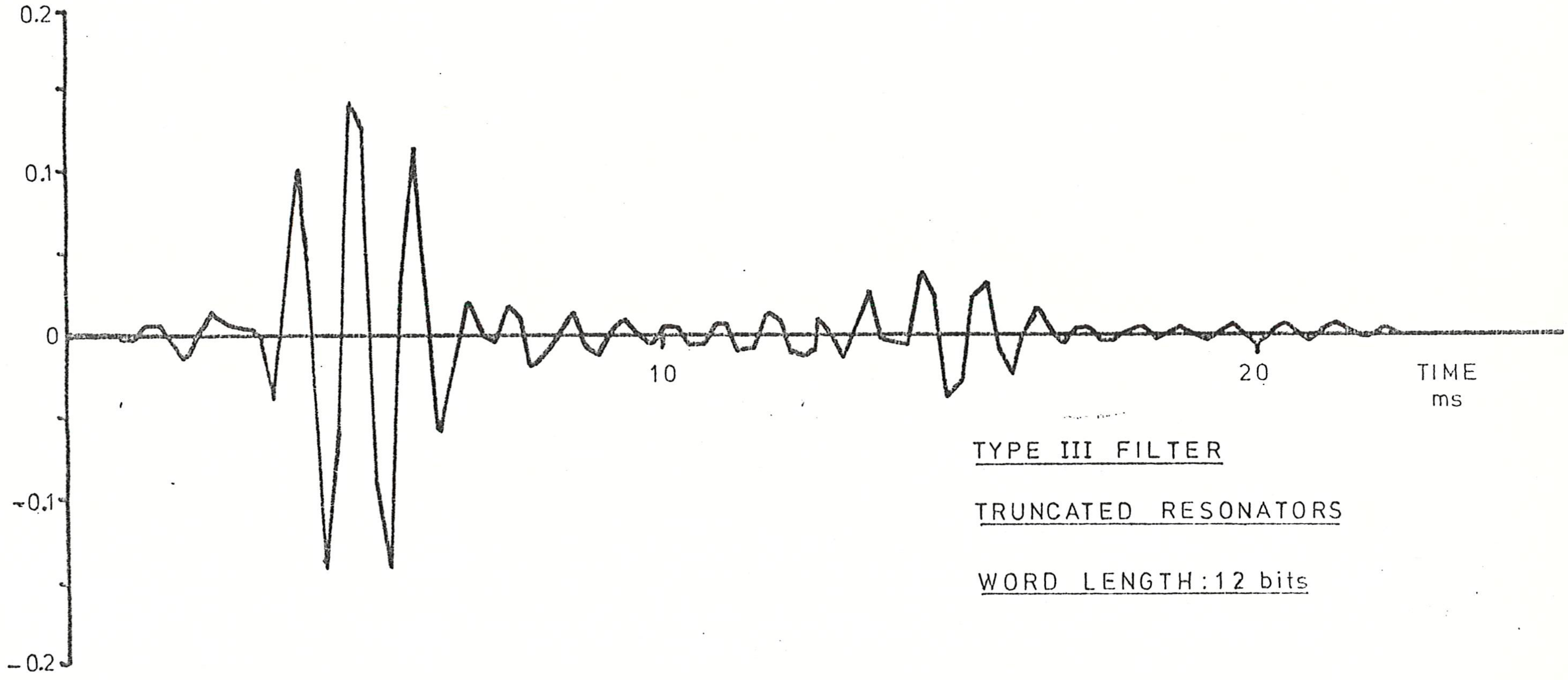


Figure 8.39 Impulse Response

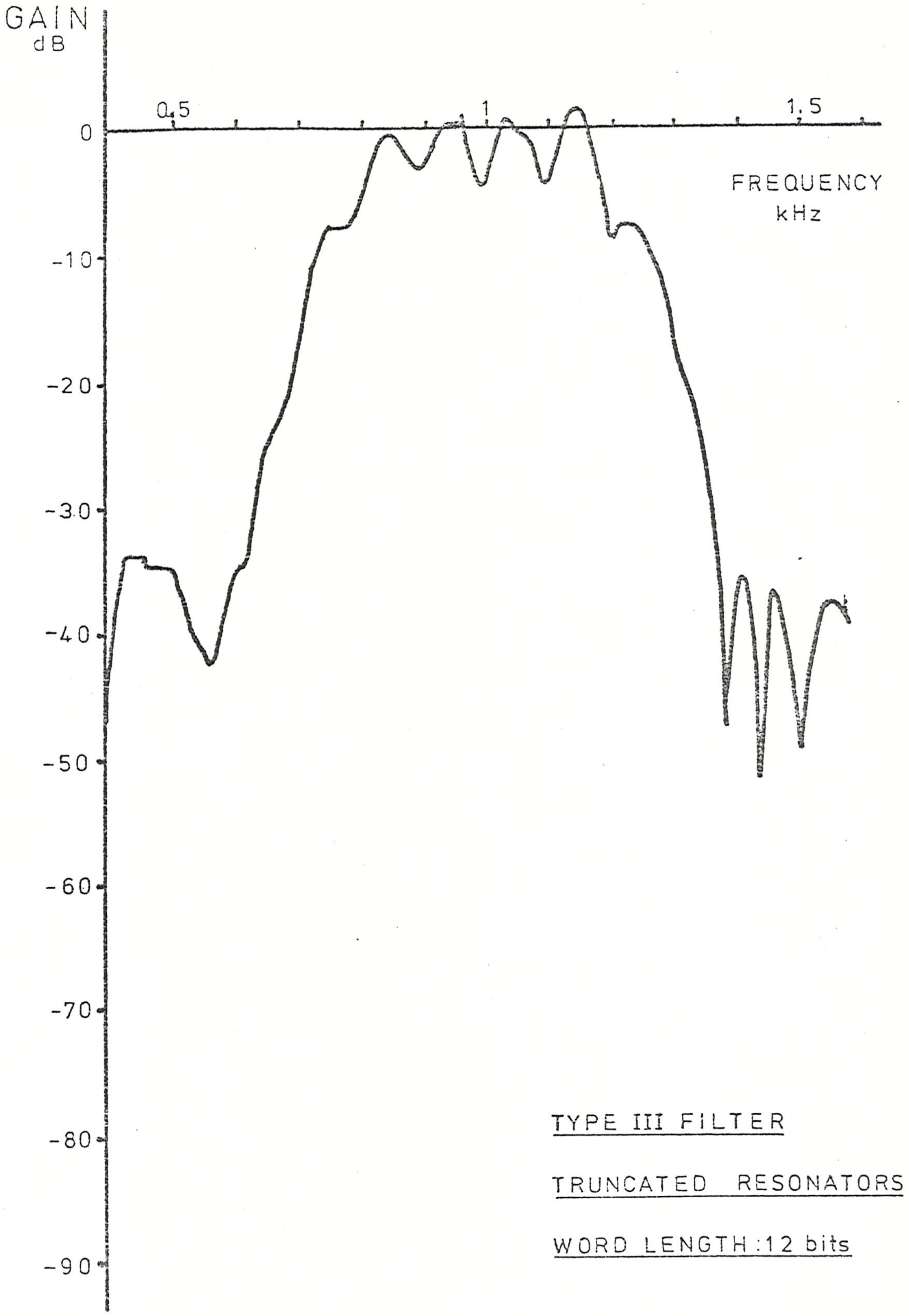


Figure 8.40 FFT-derived Frequency Response

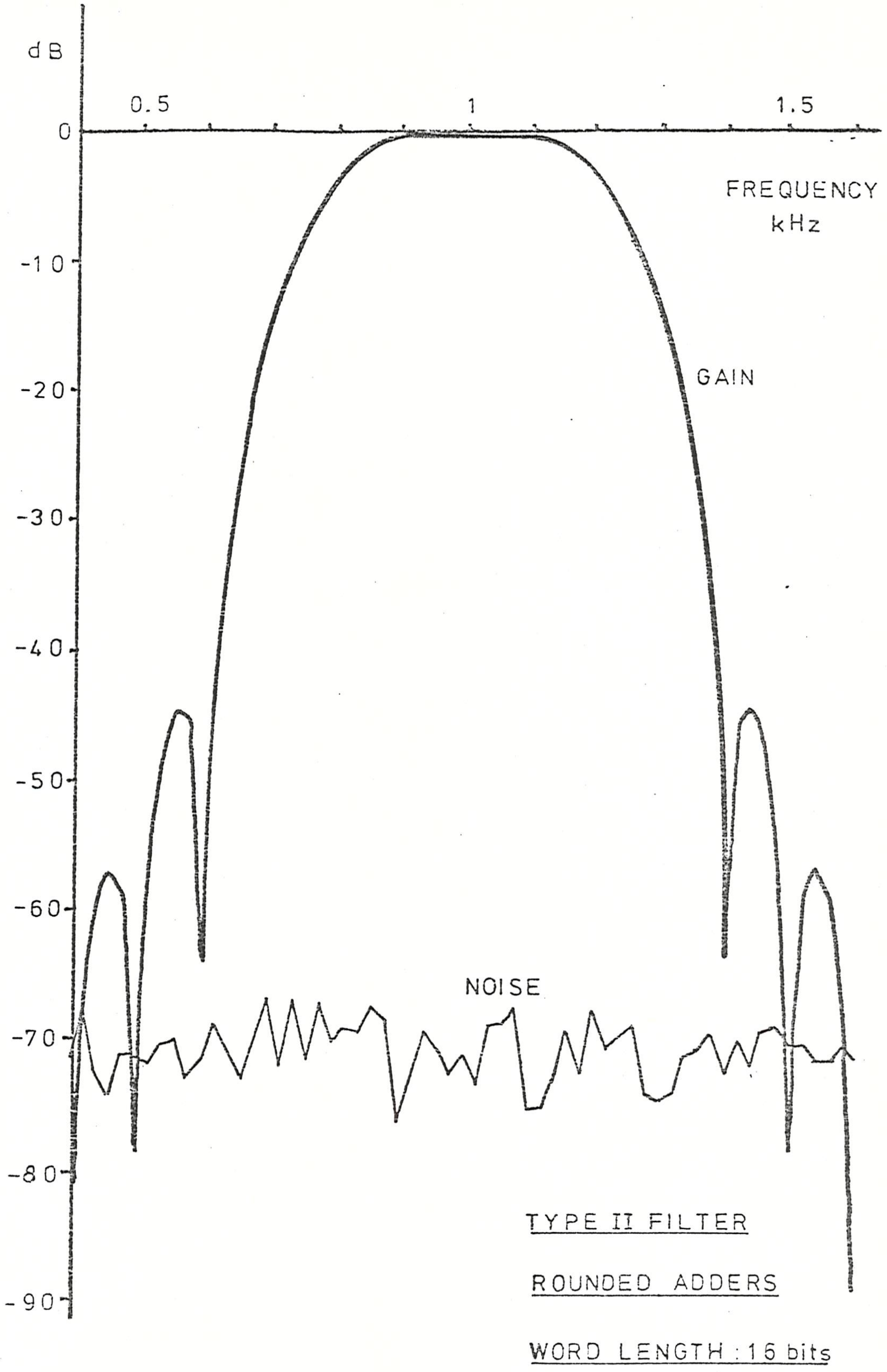
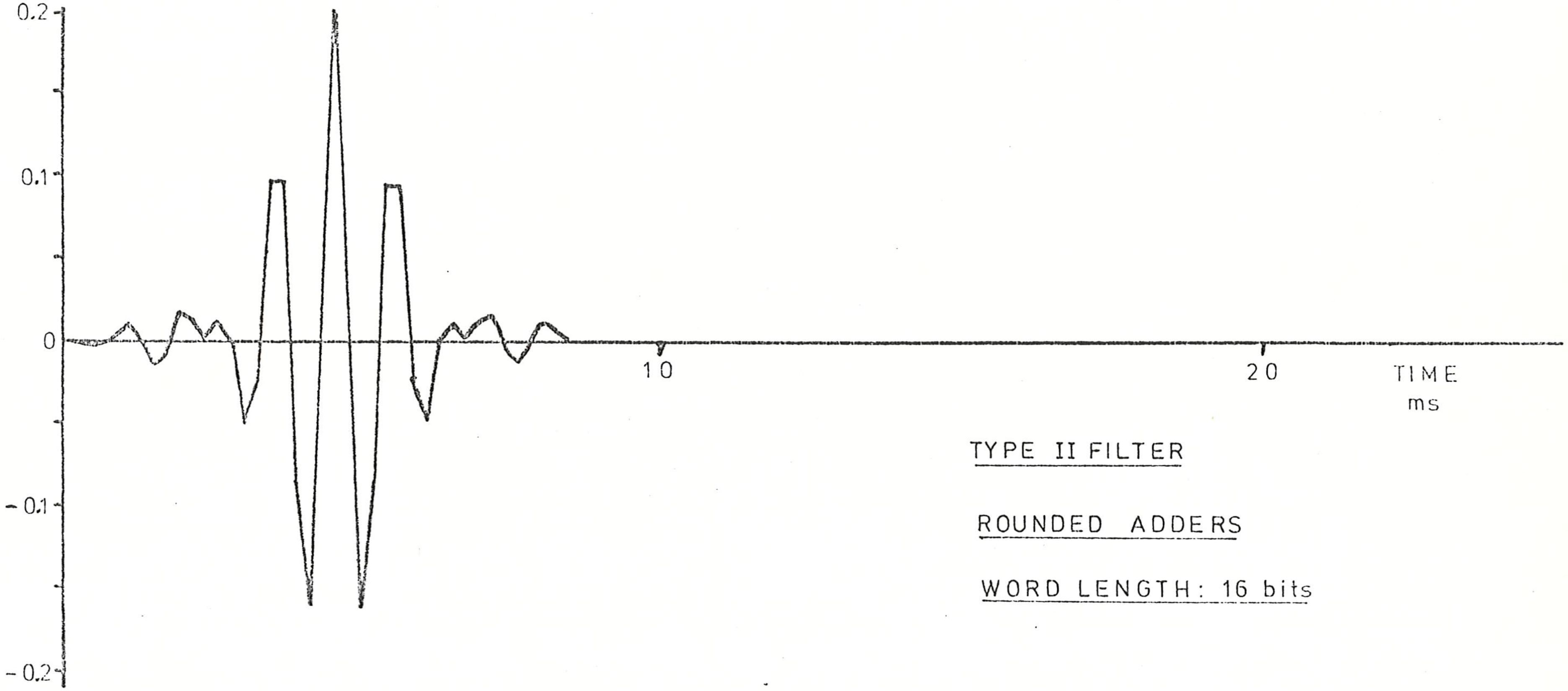


Figure 8.41 Best-fit Frequency Response



TYPE II FILTER

ROUNDED ADDERS

WORD LENGTH: 16 bits

Figure 8.42 Impulse Response

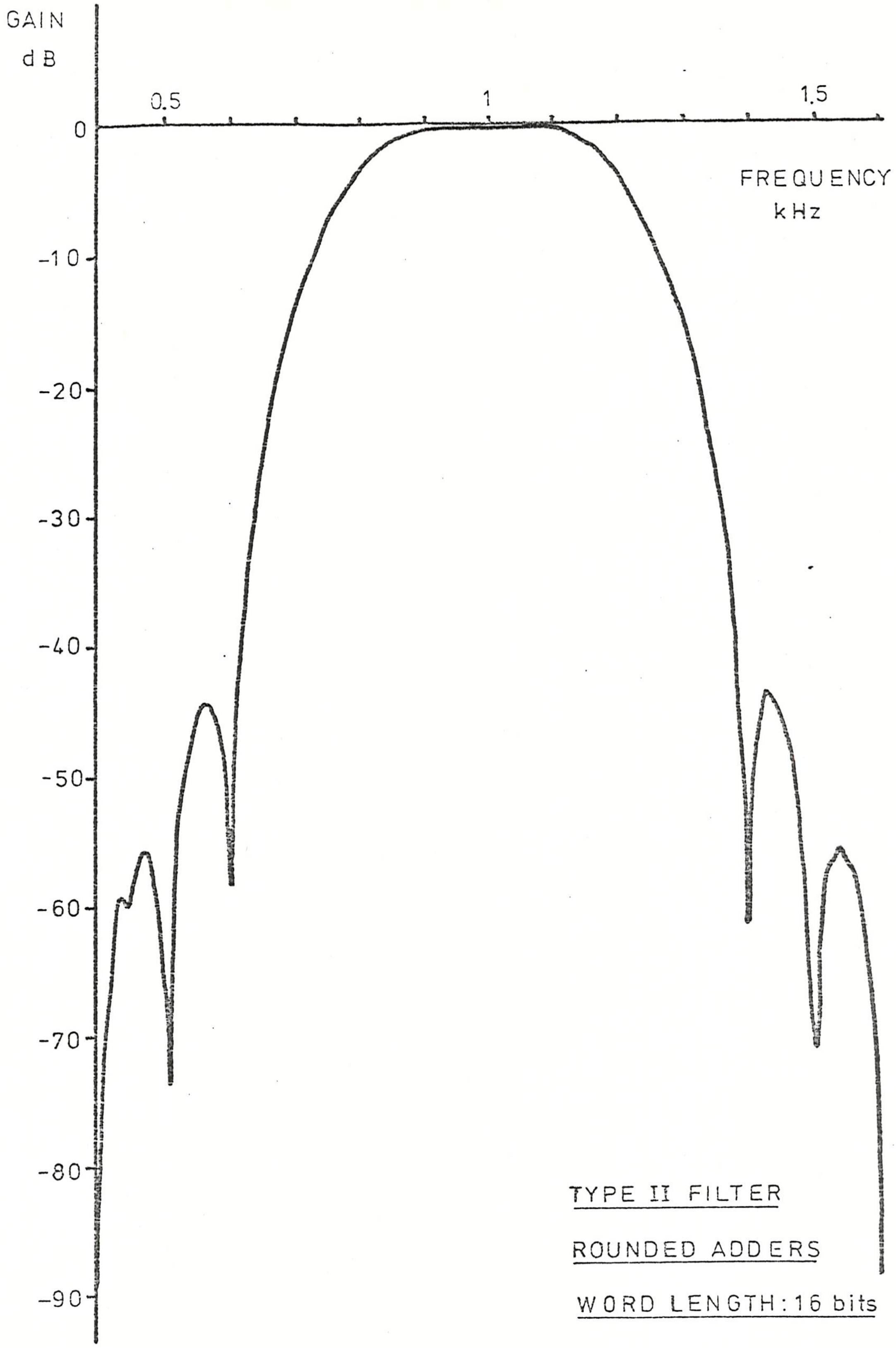


Figure 8.43 FFT-derived Frequency Response

CHAPTER NINE

SOME HARDWARE CONSIDERATIONS

9.1 Introduction.

Up to this point, attention has been confined almost exclusively to the derivation of filter transfer functions, and to the investigation of possible realisations and their relative performance. It is considered that no investigation such as this is complete unless some attention has been given to the implementation of the filter algorithms. In this context, implementation is taken to mean design and construction of special purpose hardware.

Figure 9.1 shows the functional block diagram of a digital signal processor containing random access memory (RAM) which holds the data to be processed, the arithmetic unit capable of performing the necessary additions and multiplications, a high-speed temporary store (scratch-pad) and a control unit containing a set of micro-instructions defining the particular processing function. The coefficient storage would most likely be a read-only memory (ROM) or in the case of programmable filters, a programmable read-only memory (PROM) although these tend to be somewhat expensive. Figure 9.1 is effectively a rearrangement of the traditional block diagram of a digital computer which is equally applicable to large main-frame computers, minicomputers or microprocessors. The major drawback of any processor employing general purpose processors is that the computational speed is inadequate for use with high data rates, even though reductions in cycle times are almost everyday occurrences. Thus, the only feasible course of action open to the engineer faced with the problem of a signal processor is to design and construct special purpose hardware.

In this chapter, no attempt has been made to lay down guidelines for design procedures or to actually effect a design for a frequency sampling filter. Such an undertaking would probably constitute a special investigation by itself and it is considered not to be a major part of this investigation.

The major effort being made in the field of implementation appears to be a search for faster and more economical digital multipliers, both in terms of cost and power dissipation. In addition, papers are beginning to appear which investigate the use of microprocessors in digital signal processing and it is perhaps worth looking at these relatively new devices for, as speeds increase, their use will become increasingly more widespread in this field.

It may be safely asserted that the computationally slowest element in digital filter implementations is the multiplication instruction. It is at present also by far the most expensive element and the engineer is forced to employ multiplexing if the cost of the implementation is not to become prohibitive and non-competitive. Using the frequency sampling filter realisations discussed in this paper and supposing a Type I or Type II realisation were chosen, it can be seen that the comb filter section requires one multiplication and one addition per sample while the bank of resonators requires 4 multiplications and 4 additions/sample. However, on closer examination it can be seen that one of the recursive section coefficients is exactly twice the forward path coefficient and that a simple hardwired bit shift could be used in the Type II filter. This is not the case in the Type I filter. Taking the worst case, the filter requires 29 multiplications and 29 additions per sample so that at a sampling rate of 4600 samples per second, each multiply-and-add function must be completed in less than 7.5 microseconds if a single multiplier is used. It must be emphasised that this does not take any account of other system overheads such as access times, shifting times and multiplexing.

Such a multiplication time does not preclude the use of a single multiplier for these are now available 16 × 16-bit multipliers producing a full 32-bit product in less than 200 nanoseconds but this time does not include the transfer of data from store to the multiplier inputs or the transfer of the product after rounding back to store. But speed, or data rate, is not the only criterion. The number of integrated circuits required together with the amount of power dissipated are also significant criteria and it is upon these three criteria that Jenkins and Leon (Reference 9.1) have defined a figure of merit as the data rate per unit cost.

In Reference 9.1, a dual bandpass discrete convolution filter was chosen as a basis of comparison, the results showing that for the particular form of filter chosen, multiplications effected by bit-slice techniques or by using residue number systems are far superior to the conventional multiplying system using pre-packaged multipliers with a system of integrated circuits. The comparison does not include any results for a microprocessor system although this device is becoming a serious contender for a prominent place in signal processing hardware.

9.2 Hardware employing Read-Only-Memories.

The bit-slice technique presented by Peled and Liu (Reference 9.2) is claimed to offer significant savings in cost and power consumption together with a higher operating speed as compared to existing multiplying systems. It employs a read-only-memory which generates a function ψ determined by the binary arguments corresponding to the signals at nodes in the resonator.

Let the difference equation of the digital resonator be

$$\begin{aligned} y(n) = a(0)x(n) &+ a(1)x(n-1) \\ &- b(1)y(n-1) - b(2)y(n-2) \end{aligned} \qquad 9.1$$

Choosing the 1000 Hz resonator as an example of this technique, the coefficients of the difference equation when rounded to 12-bits are

$$a(0) = 1 \qquad b(1) = -0.40625$$
$$a(1) = -0.203125 \qquad b(2) = 0.99609375$$

It is assumed that no overflow occurs and that all signals are represented by B-bits including the sign bit in 2's complement code. A function ψ is generated such that

$$\psi(p,q,s,t) = a(0)p + a(1)q - b(1)s - b(2)t \qquad 9.2$$

The function ϕ can take $2^4 = 16$ distinct values as shown in Figure 9.2. The left-hand column represents the memory address 'pqst' and the right-hand column is the contents of the memory, the first digit being the sign bit and the decimal point appearing immediately to the right of the sign bit.

The procedure is shown in diagrammatic form in Figure 9.3. Data enters serially into shift registers SR1-SR3, the least significant bit leading. At each shift instant, new values appear at the input of the ROM so defining a new address. The output of the ROM is loaded into R4 which is connected to one input of the accumulator. The other accumulator input is hard wired to the output register R5 with a 1-bit right shift. After B such shifts the value in register R5 is rounded, is passed on as the output and the accumulator is cleared. The output of R5 is passed into the shift register SR2 when the processor is again ready to compute the output due to the next sample.

One of the advantages of this procedure is that the ROM automatically converts between 2's complement and sign magnitude; a greater advantage is that a multiplier and coefficient storage has been replaced by a read-only-memory and shift registers. Each resonator section would of course require its own ROM system.

The resonator chosen has not been weighted as required in the frequency sampling realisation. This presents no difficulty for each coefficient would be scaled appropriately, the output string in Figure 9.2 being different in this case.

A similar system could be made for the implementation of the comb filter. This would require one additional ROM as the comb filter is a common element. Thus, a frequency sampling filter implemented using this bit-slice technique and serial processing, eight ROM's are required together with associated circuitry.

The speed of operation of the system depends upon the acesss time of the ROM and the addition time of the adder, the slower element being the more important.

Times quoted in Reference 9.2 indicate that a digital resonator implemented by this technique will operate at a bit-rate of 20 MHz so that if the word length is 16-bits, the resonator can operate in real time on signals having a bandwidth of some 600 kHz. Errors due to the finite word length used in this system can be investigated using a similar procedure to that presented in Chapter 5.

9.3 Hardware employing Microprocessors.

The microprocessor has replaced the conventional circuitry of a digital processor consisting of small scale integrated circuits by a single large scale integrated circuit containing registers, program counters and arithmetic logic circuits as shown in Figure 9.4 which is a block diagram of a typical microprocessor. Commercially available microprocessors might deviate from this diagram in ways which enhance some particular aspect of the device.

The word lengths and computational speeds or instruction times vary from fast 2-bit bit slice processors to fast 16-bit processors and include the now familiar 8-bit processor really intended for controller applications as well as recently introduced 16-bit processors containing a hardwired multiplier. This last type of processor is capable of executing a multiplication instruction in 32 microseconds which compared with a software execution time of 1.2 milliseconds shows that the use of hardwired multipliers significantly increases the permissible data rate of the device. Microprocessors used with an external hardwired multiplier can offer even faster execution times, a typical system being shown in Figure 9.5. The decisive factor here is the speed of the microprocessor itself for the multiplier is capable of performing the multiplication in 200 nanoseconds in the case of 16-bit words. A modern fast 16-bit microprocessor can perform a memory read operation in about 15 clock cycles which at a clock frequency of 10 MHz corresponds to 1.5 microseconds. Assuming that this is a typical figure, then a multiply sub-routine requiring 15 instructions would be executed in 23 microseconds or thereabouts, which is almost exactly the same as the time required in an 8-bit processor system (References 9.3 and 9.4).

The general trend shows improved performances for 16-bit microprocessors and it is certain that advances in technology will soon yield a microprocessor capable of a multiplication sub-routine execution time of 10 microseconds or less. When such a processor is available, it will be feasible to consider real time processing of high data rate signals.

No attempt has been made to write a program for a microprocessor system to implement a digital filter structure as it is considered that such an exercise constitutes a topic for further investigation.

9.4 Hardware employing Residue Number Systems.

Residue number coding has been recognised as an efficient implementation of high speed multiplication and addition for some time, awaiting only the introduction of fast-access electrically programmable read-only-memories for fast look-up operations (Reference 9.1). The theory of residue number arithmetic is well documented (References 9.5 and 9.6) and will not be repeated here.

The overhead costs required for encoding and decoding the signals form a significant fraction of the overall cost although the increase is offset to some extent by the savings in arithmetic hardware. Estimates show that the performance of residue number system compares very favourably with conventional implementations which could include microprocessors and with the bit-slice method using read-only-memory lookup tables. Once again, no detailed feasibility study has been carried out.

9.5 Summary.

Several proposals for the implementation of digital filters have appeared in the literature since Jackson et al (Reference 9.7) presented an approach which is well suited to implementation using large scale integration. Many of the proposals have been overtaken by events, the results of further research and increased complexity of digital circuitry leading to, on the one hand, systems which avoid multipliers by using read-only memory table look-up and on the other hand, development of asynchronous array multipliers having multiplication times of 200 manoseconds or less.

Both techniques offer their own particular advantages and it must be left to the engineer to decide upon the particular form of implementation needed to meet the given filter specification. However, the present rapid development of fast microprocessors indicates that it is highly likely that these devices will take the lead in signal processing applications in real time.

9.6 References.

9.1 Jenkins, W.K. and Leon, B.J., 'The Use of Residue Number Systems in the Design of Finite Impulse Response Digital Filters', IEEE Trans. Circuits and Systems, CAS-24, No 4, April 1977, 191-201.

9.2 Peled, A. and Liu, B., 'A New Hardware Realisation of Digital Filters', IEEE Trans. Acoustics, Speech and Signal Processing, ASSP-22, No 6, December 1974, 456-462.

9.3 Wilnai, D. and Verhofstabt, P.W.J., 'One-chip CPU Packs Power of General-purpose Minicomputers', Electronics, 23 June 1977, 113-117.

9.4 Geist, D.J. 'MOS Processor Picks Up Speed with Bipolar Multipliers', Electronics, 7 July 1977, 113-115.

9.5 Knuth, D.E., 'Seminumerical Algorithms', The Art of Computer Programming, Vol 2, Addison-Wesley, 1969, 248-257.

9.6 Garner, H.L. 'The Residue Number System', IRE Trans. Electronic Computers, June 1959, 140-147.

9.7 Jackson, L.B. Kaiser, J.F. and McDonald, H.S. 'An Approach to the Implementation of Digital Filters', IEEE Trans. Audio and Electroacoustics, Vol AU-16, No 3, September 1968, 413-421.

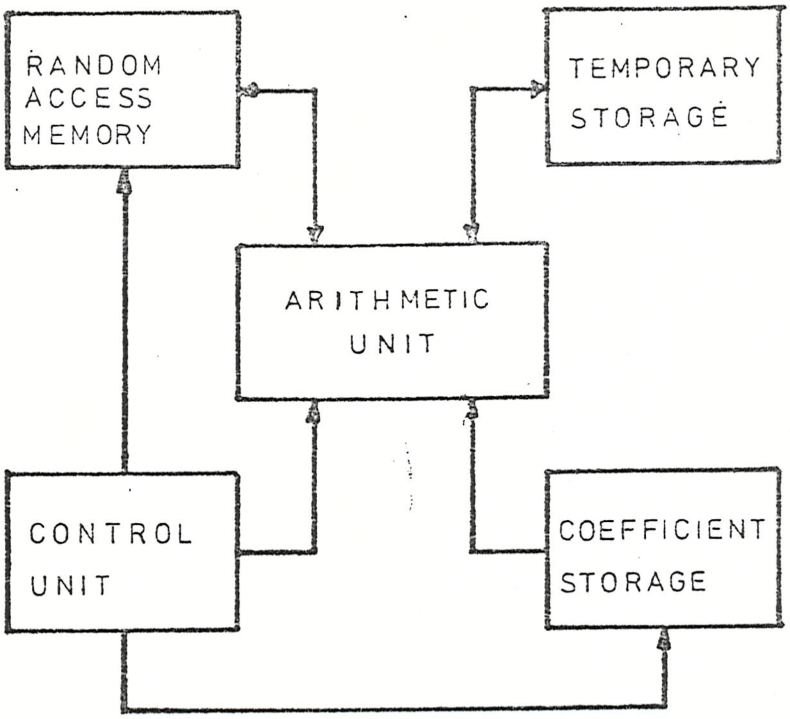


Figure 9.1 Digital Signal Processor Schematic

Memory Address				Contents ψ							
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	1
0	0	1	0	0	0	0	1	1	0	1	0
0	0	1	1	1	1	0	1	1	0	1	1
0	1	0	0	1	1	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	0	0
0	1	1	0	0	0	0	1	1	1	0	0
0	1	1	1	1	1	0	0	1	1	1	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0
1	0	1	0	0	1	0	1	1	0	1	0
1	0	1	1	0	0	0	1	1	0	1	0
1	1	0	0	0	0	1	1	0	0	1	0
1	1	0	1	1	1	1	1	0	1	0	0
1	1	1	0	0	1	0	0	1	1	0	1
1	1	1	1	0	0	0	0	1	1	0	0

Figure 9.2 ROM Lookup for Resonator

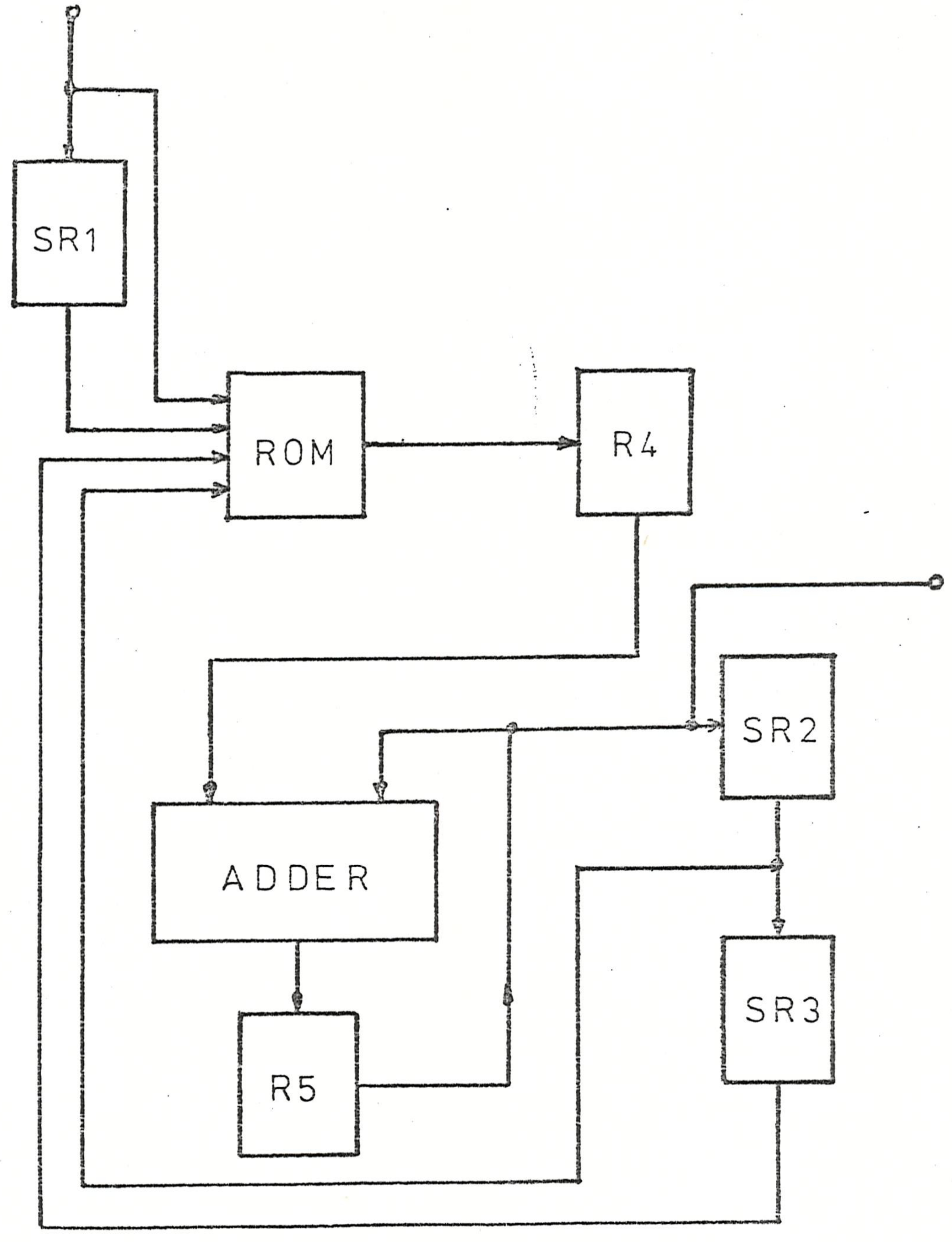


Figure 9.3 Realisation of Digital Resonator using Read-Only-Memory

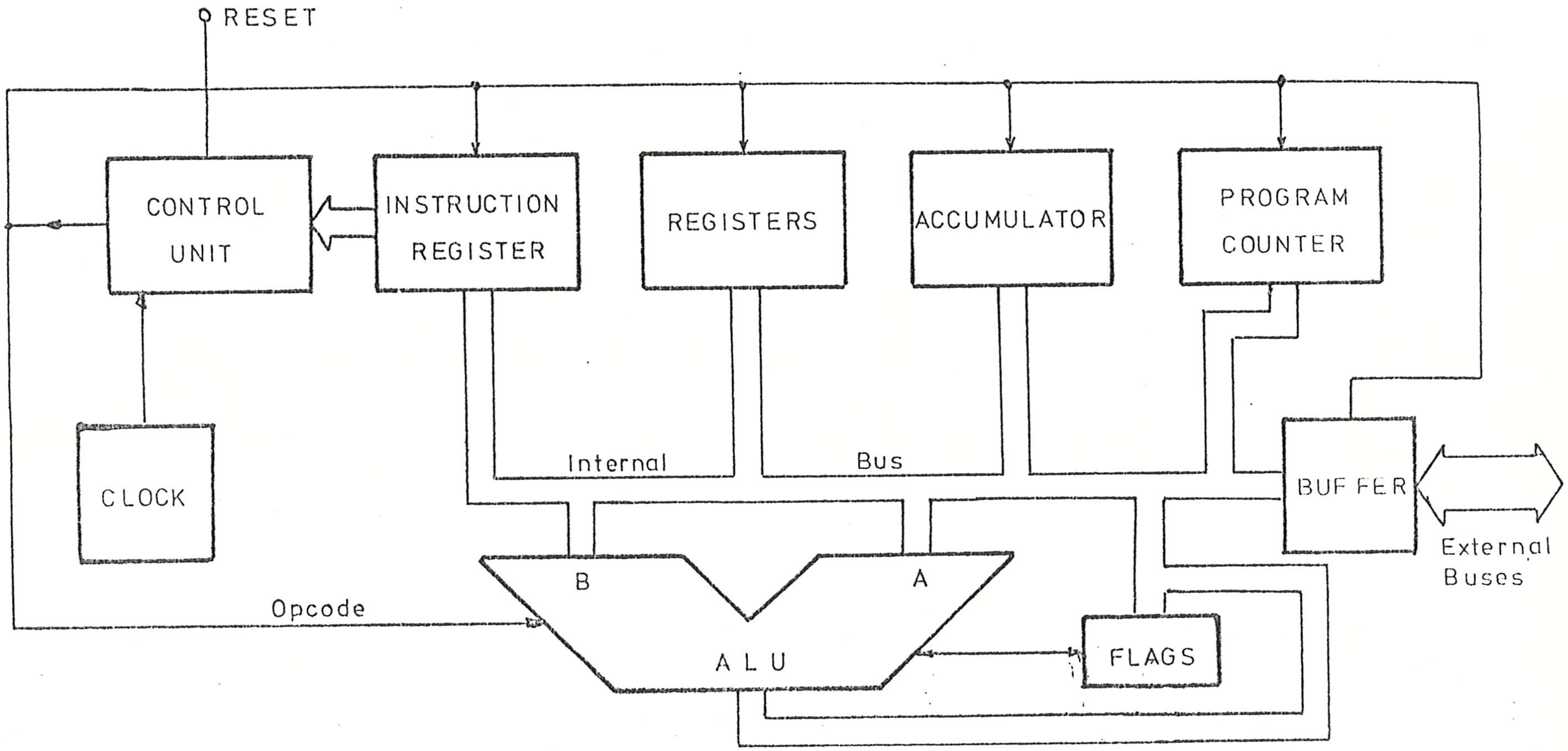


Figure 9.4 Block Diagram of Microprocessor System

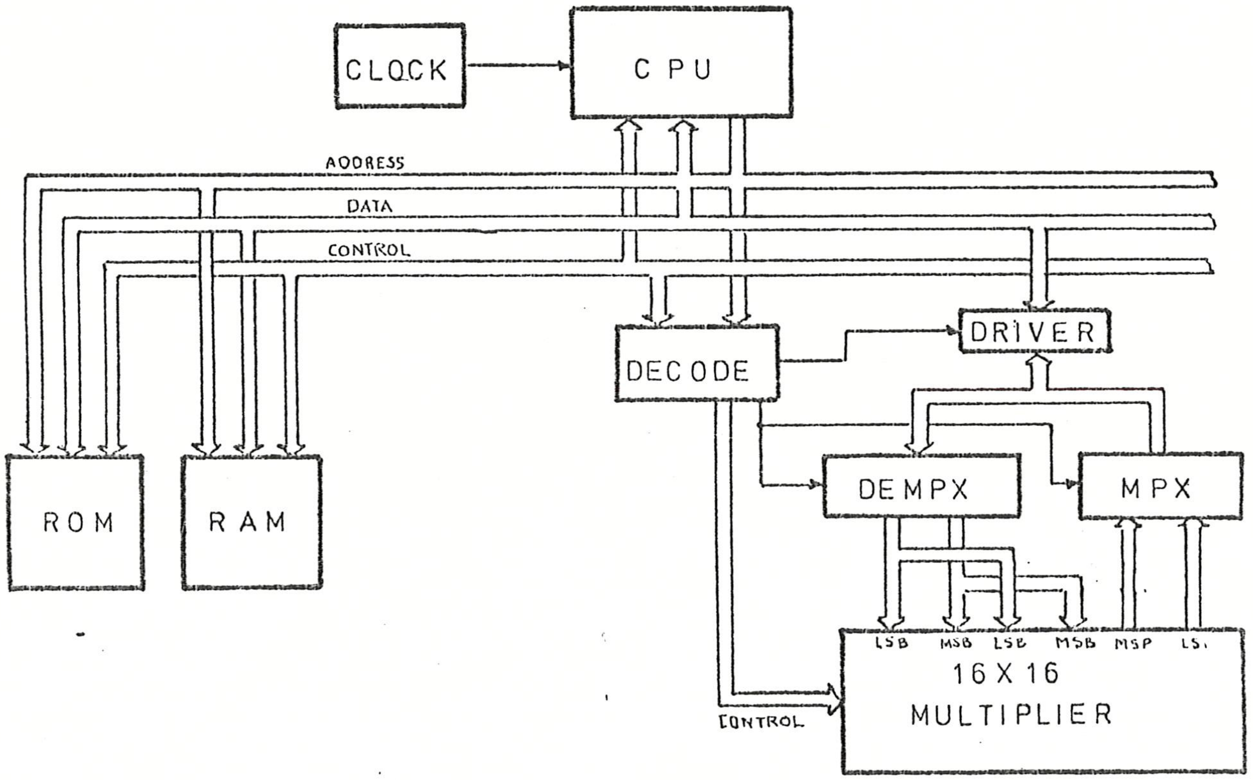


Figure 9.5 Microprocessor System with Hardwired Multiplier

CHAPTER TEN

CONCLUSIONS AND RECOMMENDATIONS

10.1 Conclusions

Three bandpass filter realisations have been investigated, each being based upon proposals presented in the existing literature. Although only two general forms of pulse transfer function were derived, the realisations differed essentially in the structure of the digital resonator, viz

Type I	-	zero preceding the poles
Type II	-	poles preceding the zero
Type III	-	poles only, the common frequency independent zero being lumped with the comb filter element.

The gain and phase responses calculated directly from the pulse transfer functions were in very close agreement with those predicted. Analyses of the noise performance and overflow conditions were undertaken for each realisation, the results of the analyses being summarised in Table 10.1.

Filter Type	Noise Performance	Overflow
I	Frequency dependent and poorest	None up to half sampling frequency
II	Frequency independent	Very limited useful frequency range
III	Improving with increase in frequency	Limited useful frequency range

Table 10.1 Summary of Noise Performance and Overflow

At approximately one quarter of the sampling frequency, the individual noise performances were almost identical. An improvement in noise performance was achieved by employing double precision accumulators, rounding being performed at the output of each accumulator - this is to be expected for there are fewer accumulators than multipliers ie fewer noise sources.

The overflow analysis showed striking differences between the realisations and placed severe limitations upon the useful frequency range of two of the filter realisations if overflow were to be avoided. The effect of overflow could be reduced by employing saturating accumulators but this would be at the expense of linearity.

The validity of the analyses was tested by developing a suite of simulation programs written in FORTRAN IV, particular attention being paid to the method of filter performance measurement. The results yielded by this method were in very close agreement with predicted figures for gain, phase and output noise. Anomalous results were obtained at test frequencies lying within the passband of the resonator but provided these very narrow frequency bands are avoided, the method is very satisfactory and has been used in the measurement of performance in other digital filter simulations.

The major drawback in any measurement of noise is the very long time required for the output of the filter to reach its steady-state. This time is completely dependent upon the damping of the resonators and can be expressed as the number of samples prior to the start of measurement given by $3 \div \delta$. Since δ is very much less than unity, the number of samples will be unavoidably large.

The results of measurements made upon all three realisations with the object of examining the effects of parameter changes upon filter performance were generally in agreement with the predicted behaviour. Difficulty was experienced in identifying definite noise characteristics when the arithmetic wordlength used was 12 bits. This was in the main due to the very large excursions of output noise and to the distinct increase in noise level observed at exactly resonator frequencies. Had longer measurement time been allowed, it is expected that the noise behaviour would have been less erratic.

The results of this investigation have supported the general concensus of opinion that the frequency sampling technique suffers from the following major drawbacks:-

a. the noise level at the output of the filter is high because of the nature of the resonators used in this technique,

b. the arithmetic wordlength is necessarily long if the output noise level is to be acceptable,

c. coefficient wordlengths must be long in order to achieve a good approximation to the desired filter characteristic,

d. an inordinately long measurement time is required if noise level measurements are to be consistent,

e. the hardware complexity and computational time increase with the number of non-zero frequency samples.

In addition, it has been shown that two of the three realisations proposed in the literature are very susceptible to overflow.

To summarise, the aims of the investigation have been achieved, the results supporting existing theories and analyses of frequency sampling filters. A satisfactory method of filter performance measurement has been developed which can be used with other linear digital signal processors.

10.2 Recommendations

In view of the complexity and demanding nature of the frequency sampling technique, it is doubtful whether any further examination of the technqiue would produce results which would indicate that this technique has distinct advantages over other techniques. Closer investigation of the actual behaviour of noise within the resonators could prove interesting but of little practical significance as digital hardware is presently available having wordlengths which reduce the noise problem to very minor proportions.

In the brief review of some possible hardware implementations contained in Chapter 9, it became evident that the designer has a wide range of implementation methods to choose from and that the actual method finally employed would be dictated by a variety of technical and economic considerations. Indeed, the question as to the use of digital circuits to implement filters may be called into question, other techniques such as Surface Acoustic Waves and Charge Controlled Devices competing for a place in pre-detection and post-detection filtering.

It is therefore recommended that further work in the examination of the frequency sampling technique per se should not be undertaken but that a profitable area of research lies in the investigation of the relative merits of other techniques for the approximation of filter characteristics.

APPENDIX A

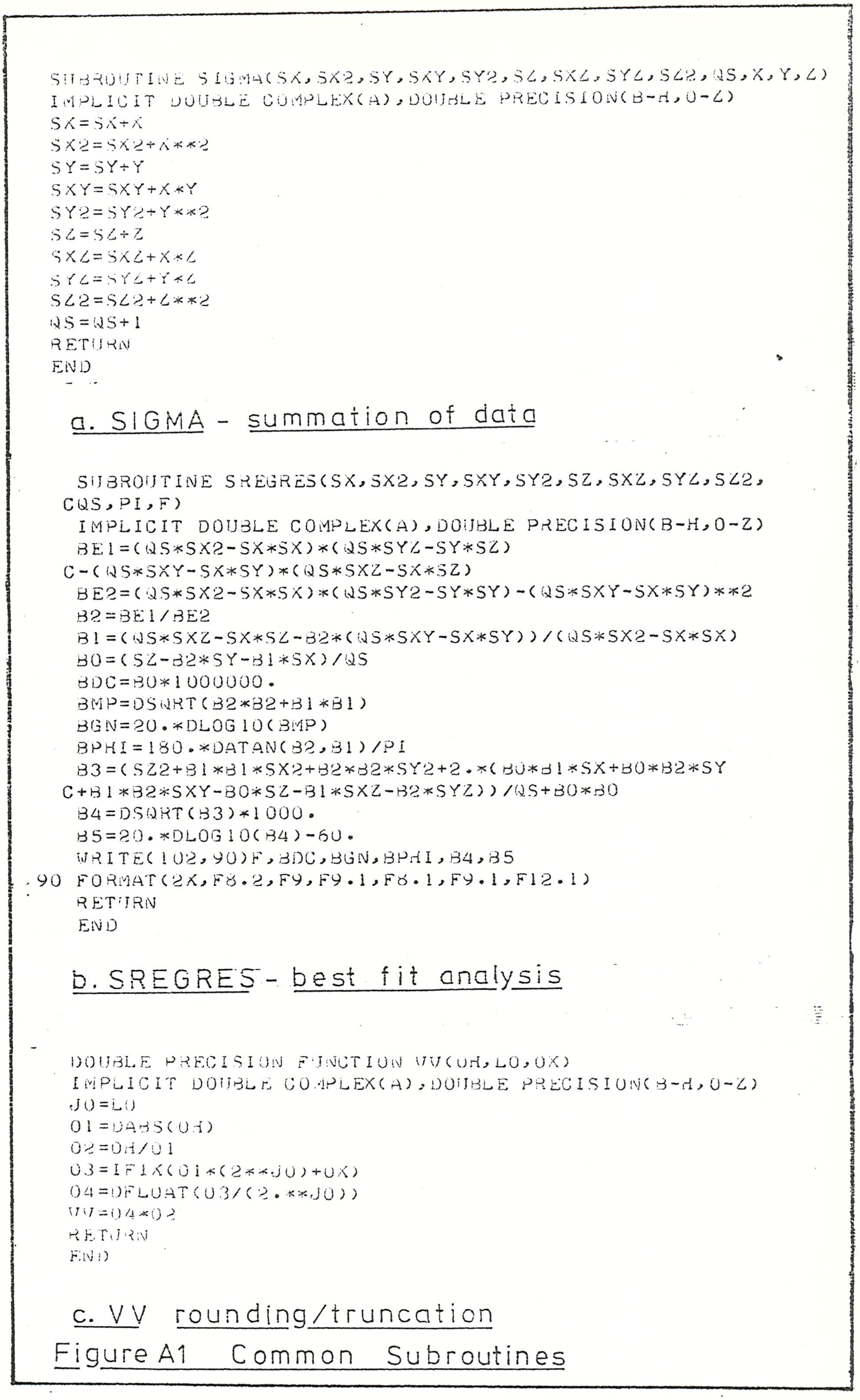
DESCRIPTION OF SIMULATION PROGRAMS

Figures A1 to A7 give details of the sub-routines used in the simulation of the bandpass digital filters under investigation, Figures A8 to A10 giving details of typical main programs. For example the main program in Figure A9 uses sub-routines appropriate to the Type III filter to produce a noise comparison. If other filter types are to be investigated, the relevant sub-routine calls must be amended and the main program merged with the appropriate sub-routines. It will be seen that the array dimensions are specified having values appropriate to a bank of seven resonators, 46 frequency samples, 512 samples of impulse response and 460 samples of filter output. By a simple editing process, these dimensions can, of course, be varied as desired, care being taken to ensure that all the sub-routines used are also edited in the same way.

The language used is FORTRAN IV and the computer employed throughout the investigation was a Rank-Xerox Sigma Six.

The variables used in the data input to the simulations are defined below:

N	-	number of runs of the simulation
F	-	single test frequency
FMIN	-	minimum test frequency
FMAX	-	maximum test frequency
FSTEP	-	frequency increment
FS	-	sampling frequency
SA	-	input signal amplitude
RL	-	settling period measured as number of samples before measurements commence
DE	-	damping factor exponent $\delta = 2^{-DE}$
LA	-	analogue-to-digital wordlength
LC	-	coefficient wordlength
LW	-	arithmetic wordlength
LX	-	arithmetic wordlength of the 'precise' filter
CAB	-	a 7 by 2 array containing the resonator number k and the modulus of the frequency sample $\|H(k)\|$.



```
 SUBROUTINE SIGMA(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,QS,X,Y,Z)
 IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
 SX=SX+X
 SX2=SX2+X**2
 SY=SY+Y
 SXY=SXY+X*Y
 SY2=SY2+Y**2
 SZ=SZ+Z
 SXZ=SXZ+X*Z
 SYZ=SYZ+Y*Z
 SZ2=SZ2+Z**2
 QS=QS+1
 RETURN
 END
```

a. SIGMA - summation of data

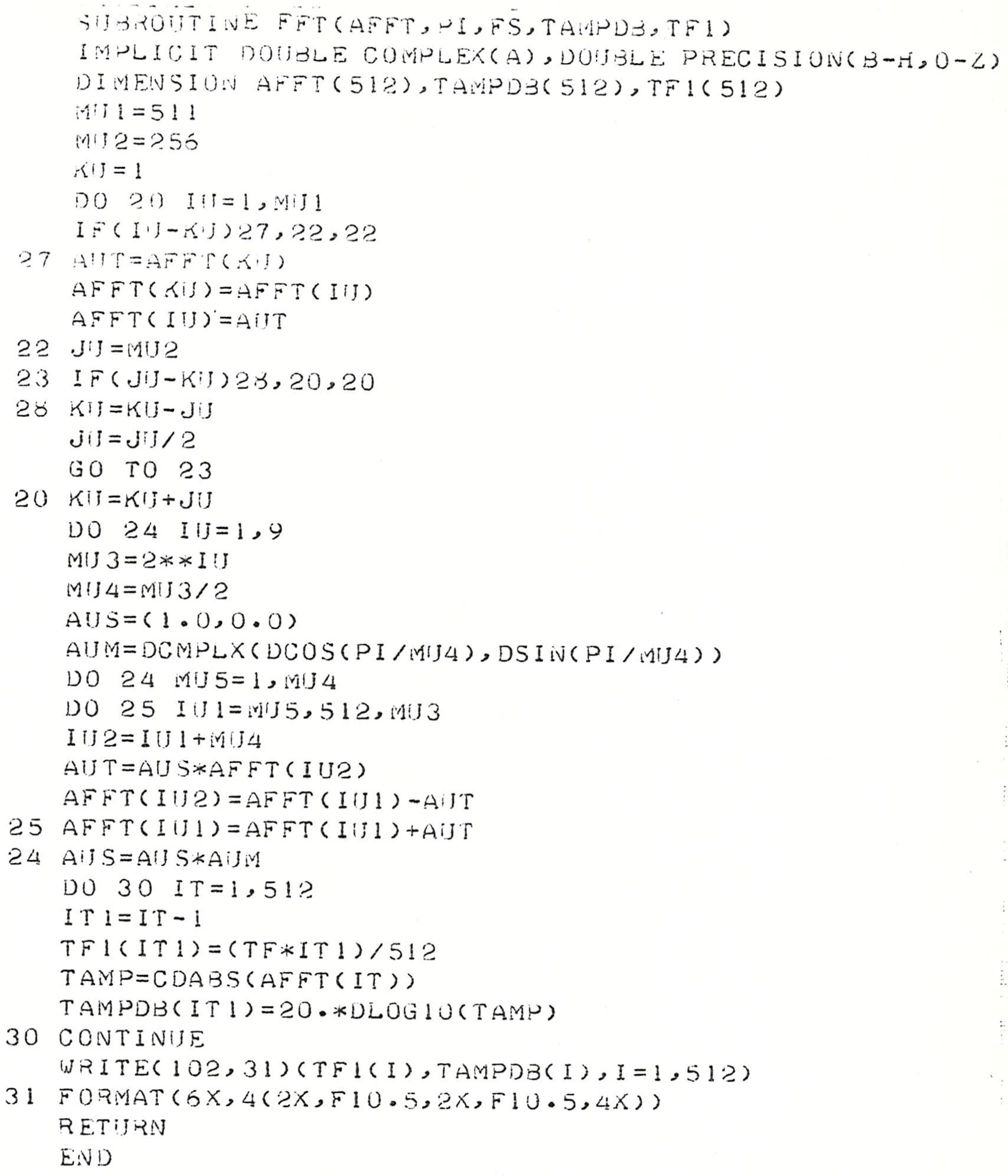
```
   SUBROUTINE SREGRES(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,
  CQS,PI,F)
   IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
   BE1=(QS*SX2-SX*SX)*(QS*SYZ-SY*SZ)
  C-(QS*SXY-SX*SY)*(QS*SXZ-SX*SZ)
   BE2=(QS*SX2-SX*SX)*(QS*SY2-SY*SY)-(QS*SXY-SX*SY)**2
   B2=BE1/BE2
   B1=(QS*SXZ-SX*SZ-B2*(QS*SXY-SX*SY))/(QS*SX2-SX*SX)
   B0=(SZ-B2*SY-B1*SX)/QS
   BDC=B0*1000000.
   BMP=DSQRT(B2*B2+B1*B1)
   BGN=20.*DLOG10(BMP)
   BPHI=180.*DATAN(B2,B1)/PI
   B3=(SZ2+B1*B1*SX2+B2*B2*SY2+2.*(B0*B1*SX+B0*B2*SY
  C+B1*B2*SXY-B0*SZ-B1*SXZ-B2*SYZ))/QS+B0*B0
   B4=DSQRT(B3)*1000.
   B5=20.*DLOG10(B4)-60.
   WRITE(102,90)F,BDC,BGN,BPHI,B4,B5
90 FORMAT(2X,F8.2,F9,F9.1,F8.1,F9.1,F12.1)
   RETURN
   END
```

b. SREGRES - best fit analysis

```
  DOUBLE PRECISION FUNCTION VV(OH,LO,OX)
  IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
  JO=LO
  O1=DABS(OH)
  O2=OH/O1
  O3=IFIX(O1*(2**JO)+OX)
  O4=DFLOAT(O3/(2.**JO))
  VV=O4*O2
  RETURN
  END
```

c. VV rounding/truncation

Figure A1 Common Subroutines



```
      SUBROUTINE FFT(AFFT,PI,FS,TAMPDB,TF1)
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION AFFT(512),TAMPDB(512),TF1(512)
      MU1=511
      MU2=256
      KU=1
      DO 20 IU=1,MU1
      IF(IU-KU)27,22,22
   27 AUT=AFFT(KU)
      AFFT(KU)=AFFT(IU)
      AFFT(IU)=AUT
   22 JU=MU2
   23 IF(JU-KU)28,20,20
   28 KU=KU-JU
      JU=JU/2
      GO TO 23
   20 KU=KU+JU
      DO 24 IU=1,9
      MU3=2**IU
      MU4=MU3/2
      AUS=(1.0,0.0)
      AUM=DCMPLX(DCOS(PI/MU4),DSIN(PI/MU4))
      DO 24 MU5=1,MU4
      DO 25 IU1=MU5,512,MU3
      IU2=IU1+MU4
      AUT=AUS*AFFT(IU2)
      AFFT(IU2)=AFFT(IU1)-AUT
   25 AFFT(IU1)=AFFT(IU1)+AUT
   24 AUS=AUS*AUM
      DO 30 IT=1,512
      IT1=IT-1
      TF1(IT1)=(TF*IT1)/512
      TAMP=CDABS(AFFT(IT))
      TAMPDB(IT1)=20.*DLOG10(TAMP)
   30 CONTINUE
      WRITE(102,31)(TF1(I),TAMPDB(I),I=1,512)
   31 FORMAT(6X,4(2X,F10.5,2X,F10.5,4X))
      RETURN
      END
```

FFT - Fast Fourier Transform

Figure A2 Common Subroutine

```
      SUBROUTINE FILCON12(LC,CAB,CR,CR2,CRN,PI,DE,ZAB)
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION CAB(7,2),ZAB(7,2)
      CR=(2.**DE-1.)/(2.**DE)
      CR2=VV(CR**2,LC,.5D0)
      CRN=VV(CR**46,LC,.5D0)
      DO75 IH=1,7
      H1=1
      IF(CAB(IH,1)/2.-DFLOAT(IFIX(CAB(IH,1)/2.))-0.25)77,77,76
   76 H1=-H1
   77 H1=H1*CAB(IH,2)*16/23
      H2=CR*DCOS(PI*CAB(IH,1)/23.)
      ZAB(IH,1)=VV(H1,LC,.5D0)
      ZAB(IH,2)=VV(H2,LC,.5D0)
   75 WRITE(102,74)ZAB(IH,1),ZAB(IH,2)
   74 FORMAT(4X,F12.9,4X,F12.9)
      RETURN
      END
```

a. Calculation of Filter Coefficients

```
      SUBROUTINE  INITDLY12(ZN,ZRES,Z1)
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION  ZN(46),ZRES(7,2),Z1(7)
      DO 1Ø IE=1,46
   1Ø ZN(IE)=Ø
      DO 11 IE=1,7
   11 ZRES(IE,2)=ZRES(IE,1)=Z1(IE)=Ø.
      RETURN
      END
```

b. Initialisation of Stores

Figure A3 Subroutines for Type III

```
      SUBROUTINE FILCON3(LC,CAB,CR,CR2,CRN,PI,DE,ZAB)
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION CAB(7,2),ZAB(7,2)
      CR=(2.**DE-1.)/(2.**DE)
      CR2=VV(CR**2,LC,.5D0)
      CRN=VV(CR**46,LC,.5D0)
      DO75 IH=1,7
      H1=1
      IF(CAB(IH,1)/2.-DFLOAT(IFIX(CAB(IH,1)/2.))-0.25)77,77,76
76    H1=-H1
77    H1=H1*CAB(IH,2)*DCOS(HPI*CAB(IH,1)/46.)*16/23
      H2=CR*DCOS(PI*CAB(IH,1)/23.)
      ZAB(IH,1)=VV(H1,LC,.5D0)
      ZAB(IH,2)=VV(H2,LC,.5D0)
75    WRITE(102,74)ZAB(IH,1),ZAB(IH,2)
74    FORMAT(4X,F12.9,4X,F12.9)
      RETURN
      END
```

a. Calculation of Filter Coefficients

```
      SUBROUTINE  INITDLY3(ZN,ZRES,Z1)
      IMPLICIT DOUBLE  COMPLEX(A),DOUBLE    PRECISION(B-H,O-Z)
      DIMENSION  ZN(46),ZRES(7,2)
      Z1=0.
      DO 10 IE=1,46
10     ZN(IE)=0.
      DO 11 IE=1,7
11    ZRES(IE,2)=ZRES(IE,1)=0.
      RETURN
      END
```

b. Initialisation of Stores

Figure A4 Subroutines for Type III

```
   SUBROUTINE DIGIFILT1(ZN,ZRES,ZAB,X,CR,CR2,CRN,GZ,GO,LG,LA)
   IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
   DIMENSION ZN(46),ZRES(7,2),ZAB(7,2),Z1(7)
   GI=VV(X,LA,.5D0)/2.
   G=ZN(1)
   DO 80 IG=1,45
   IGN=IG+1
80 ZN(IG)=ZN(IGN)
   ZN(46)=GI
   GJ=(GI-VV(G*CRN),LV,.5D0)/16.
   Z=0
   DO 81 IG=1,7
   GK=GJ-VV((Z1(IG)*ZAB(IG,2)),LV,.5D0)
   Z1(IG)=GJ
   GP=GK-VV((ZRES(IG,2)*CR2),LV,0D0)
  C+VV((ZRES(IG,1)*ZAB(IG,2)),LV,0D0)*2.
   ZRES(IG,2)=ZRES(IG,1)
   ZRES(IG,1)=GP
81 Z=Z+VV((GP*ZAB(IG,1)),LV,.5D0)
   Z=Z*2.
   RETURN
   END
```

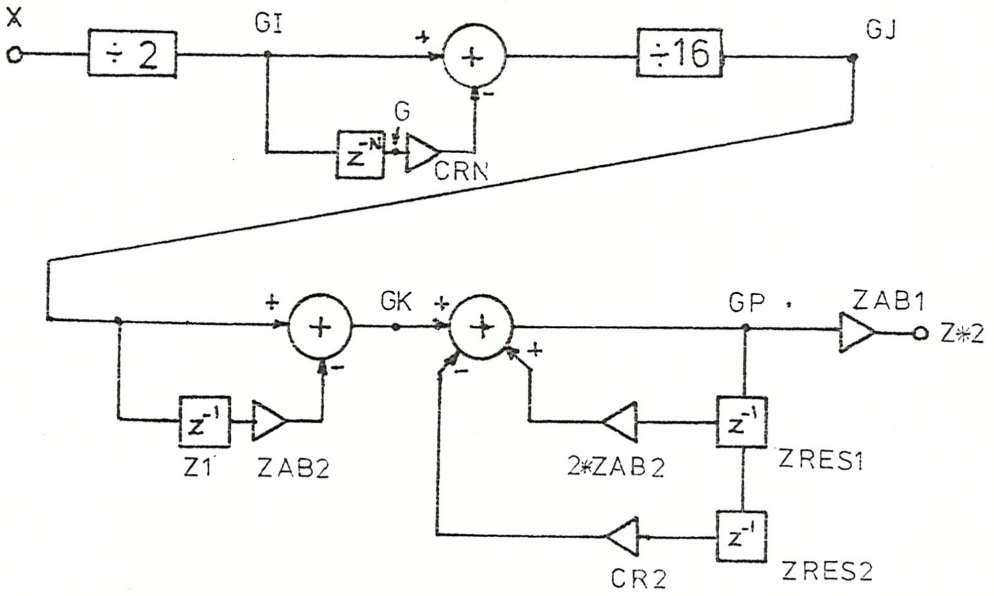
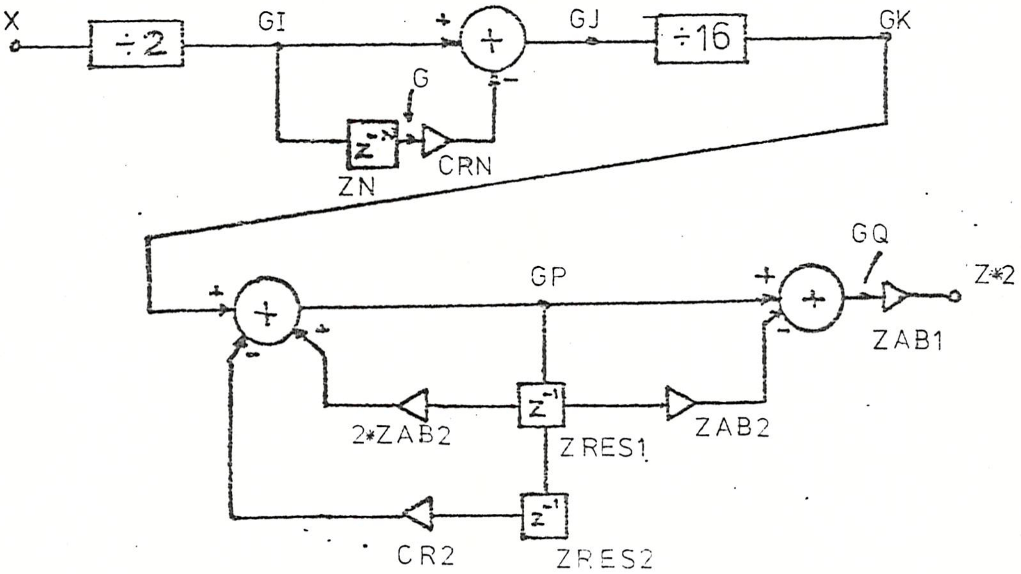


Figure A5 Type I Filter Simulation

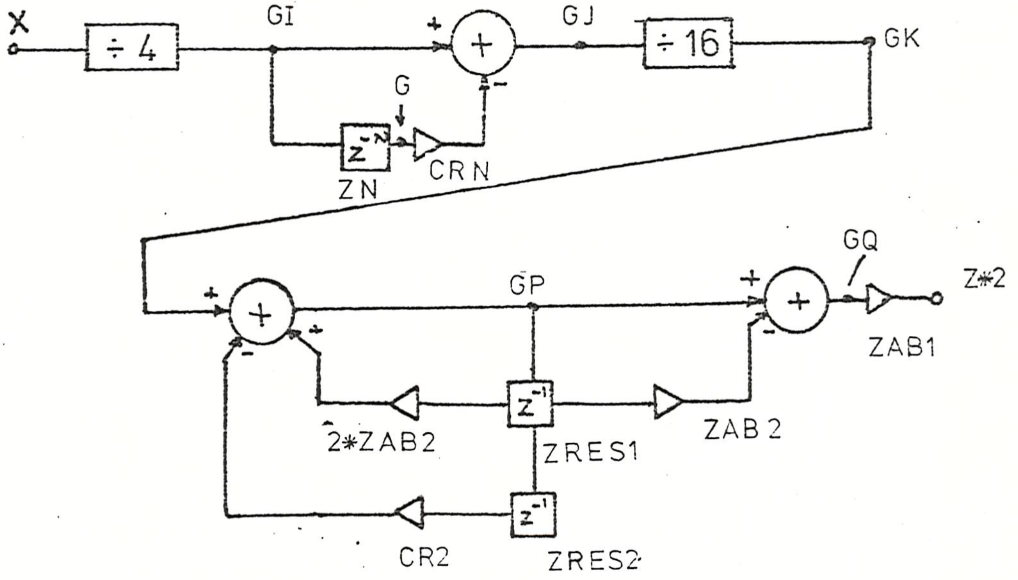
```
      SUBROUTINE DIGIFILT2(ZN,ZRES,ZAB,X,CR,CR2,CRN,Z1,Z,LV,LA)
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION ZN(46),ZRES(7,2),ZAB(7,2),Z1(7)
      GI=VV(X,LA,.5D0)/2.
      G=ZN(1)
      DO 80 IG=1,45
      IGN=IG+1
   80 ZN(IG)=ZN(IGN)
      ZN(46)=GI
      GJ=GI-VV((G*CRN),LV,.5D0)
      Z=0.
      DO 81 IG=1,7
      GK=GJ/16.
      GP=GK-VV((ZRES(IG,2)*CR2),LV,0D0)
     C+VV((ZRES(IG,1)*ZAB(IG,2)),LV,0D0)*2.
      GQ=GP-VV((ZRES(IG,1)*ZAB(IG,2)),LV,.5D0)
      ZRES(IG,2)=ZRES(IG,1)
      ZRES(IG,1)=GP
   81 Z=Z+VV((GQ*ZAB(IG,1),LV,.5D0)
      Z=Z*2.
      RETURN
      END
```



Rounded Multipliers

Figure A6a Type II Filter Simulation

```
      SUBROUTINE DIGIFILT2A(ZN,ZRES,ZAB,X,CR,CR2,CRN,Z1,Z,LW,LA)
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION ZN(46),ZRES(7,2),ZAB(7,2),Z1(7)
      GI=VV(X,LA,.5D0)/2.
      G=ZN(1)
      DO 80 IG=1,45
      IGN=IG+1
   80 ZN(IG)=ZN(IGN)
      ZN(46)=GI
      GJ=GI-VV((G*CRN),LW,.5D0)
      Z=0.
      DO 81 IG=1,7
      GK=GJ/16.
      GP=GK-(ZRES(IG,2)*CR2)+(ZRES(IG,1)*ZAB(IG,2))*2
      GP=VV(GP,LW,.5D0)
      GQ=VV(GP-ZRES(IG,1)*ZAB(IG,2),LV,.5D0)
      ZRES(IG,2)=ZRES(IG,1)
      ZRES(IG,1)=GP
   81 Z=Z+(GQ*ZAB(IG,1))
      Z=(VV(Z,LW,.5D0))*2.
      RETURN
      END
```



Rounded Accumulators

Figure A6b Type II Filter Simulation

```
      SUBROUTINE DIGIFILT3(ZN,ZRES,ZAB,X,CR,CR2,CRN,Z1,Z,LV,LA)
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION ZN(46),ZRES(7,2),ZAB(7,2)
      GI=VV(X,LA,.5D0)/4.
      GJ=GI-VV((Z1*CR),LV,.5D0)
      Z1=GI
      G=ZN(1)
      DO 80 IG=1,45
      IGN=IG+1
 80   ZN(IG)=ZN(IGN)
      ZN(46)=GJ
      GK=GJ-VV((G*CRN),LV,.5D0)
      Z=0.
      DO 81 IG=1,7
      GP=GK/16-VV((ZRES(IG,2)*CR2),LV,.5D0)
     C+VV((ZRES(IG,1)*ZAB(IG,2)),LV,.5D0)*2
      ZRES(IG,2)=ZRES(IG,1)
      ZRES(IG,1)=GP
 81   Z=Z+VV((GP*ZAB(IG,1)),LV,.5D0)
      Z=Z*4.
      RETURN
      END
```

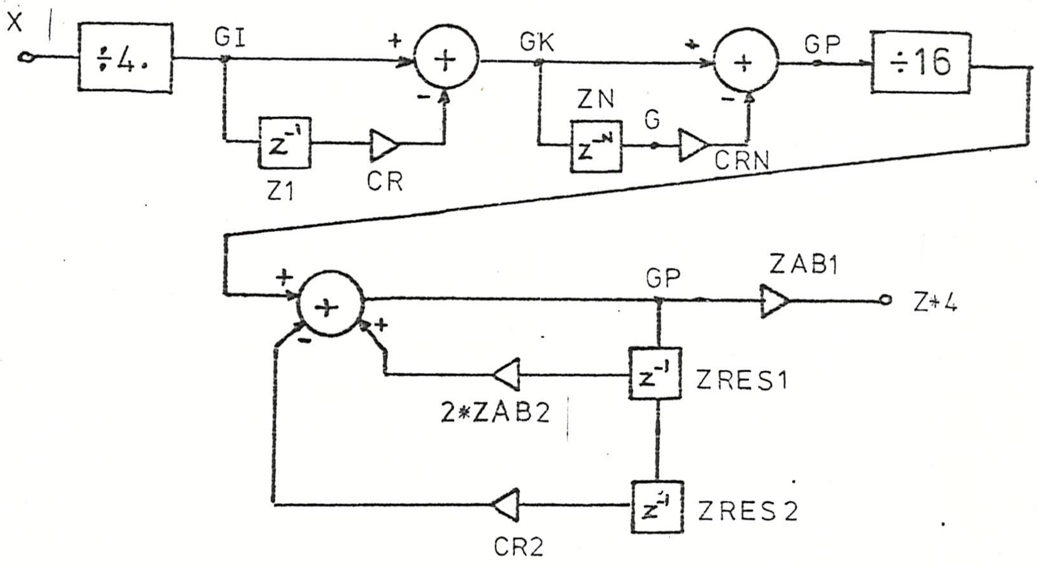
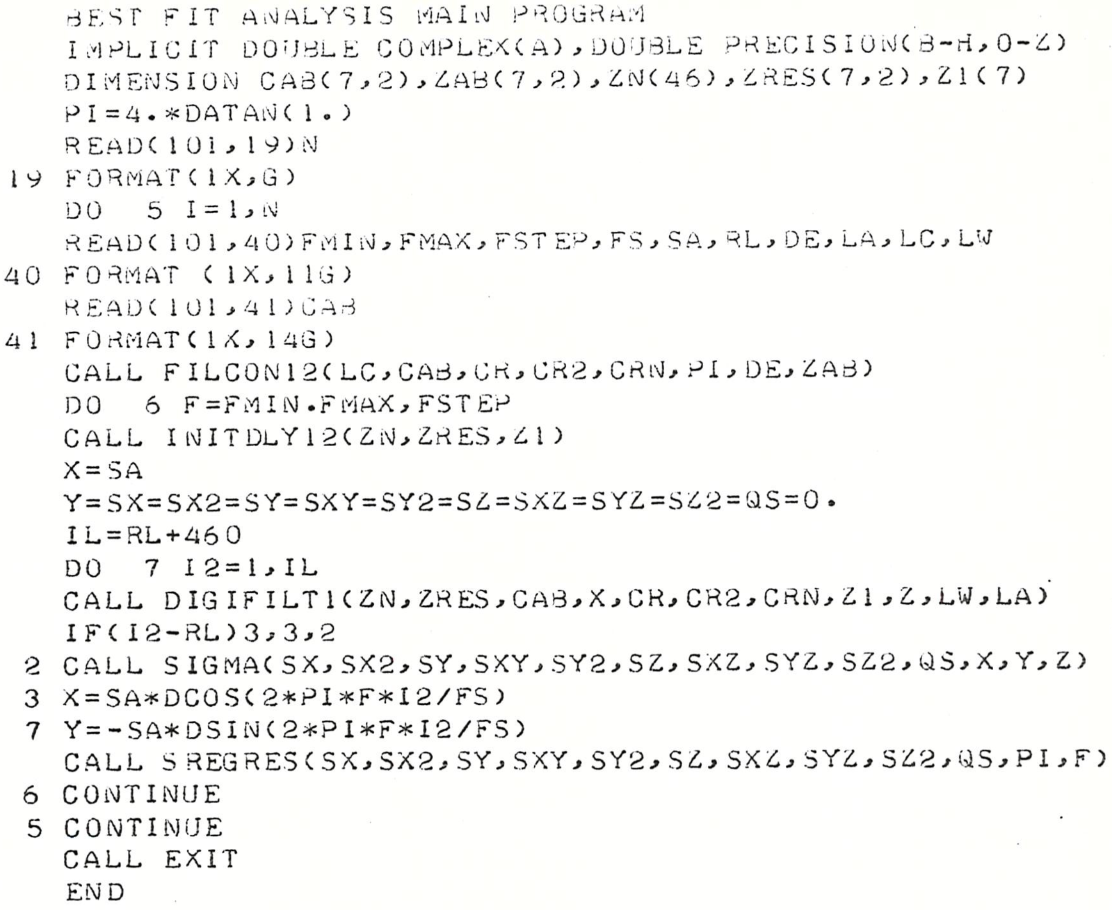
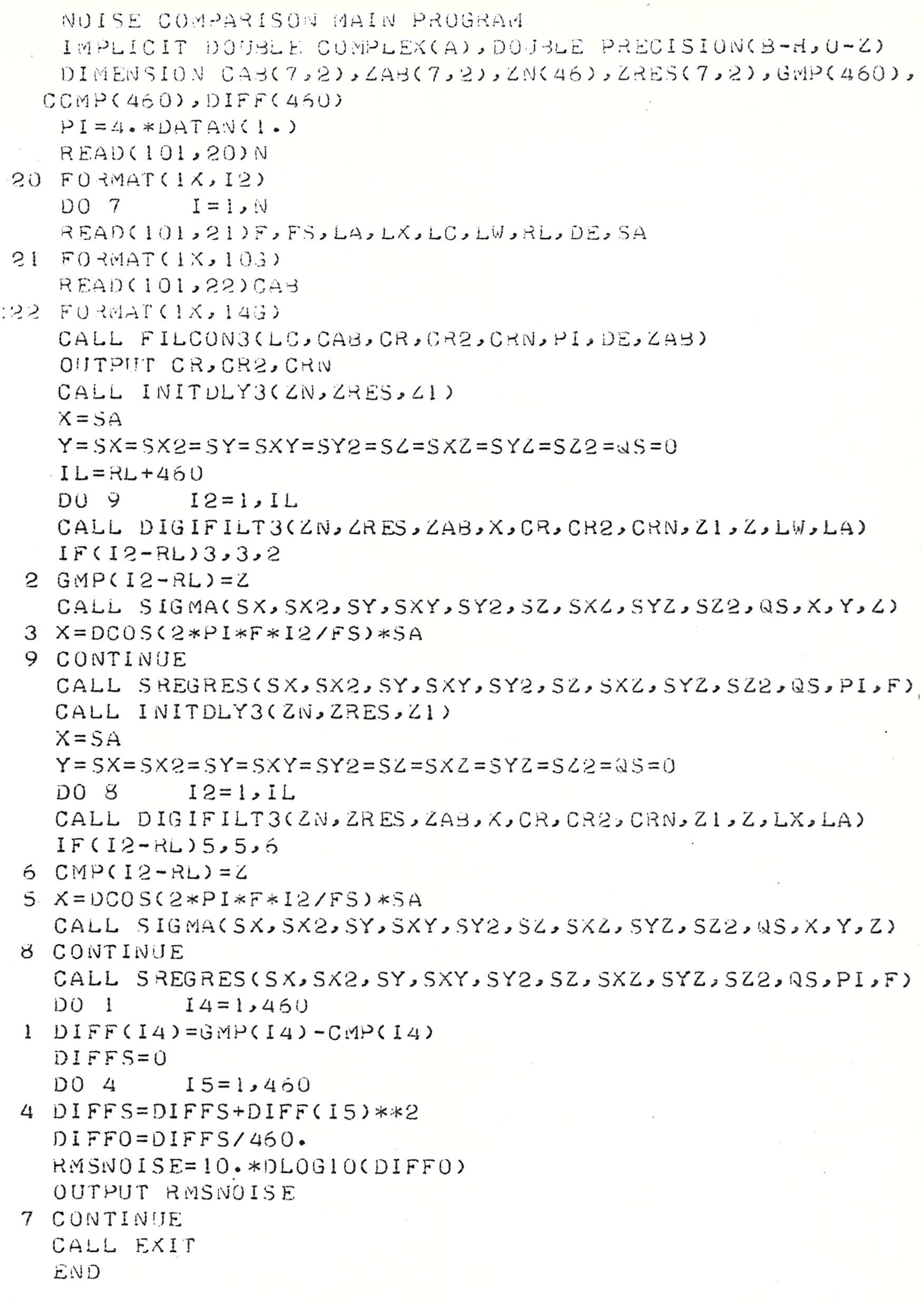


Figure A7 Type III Filter Simulation



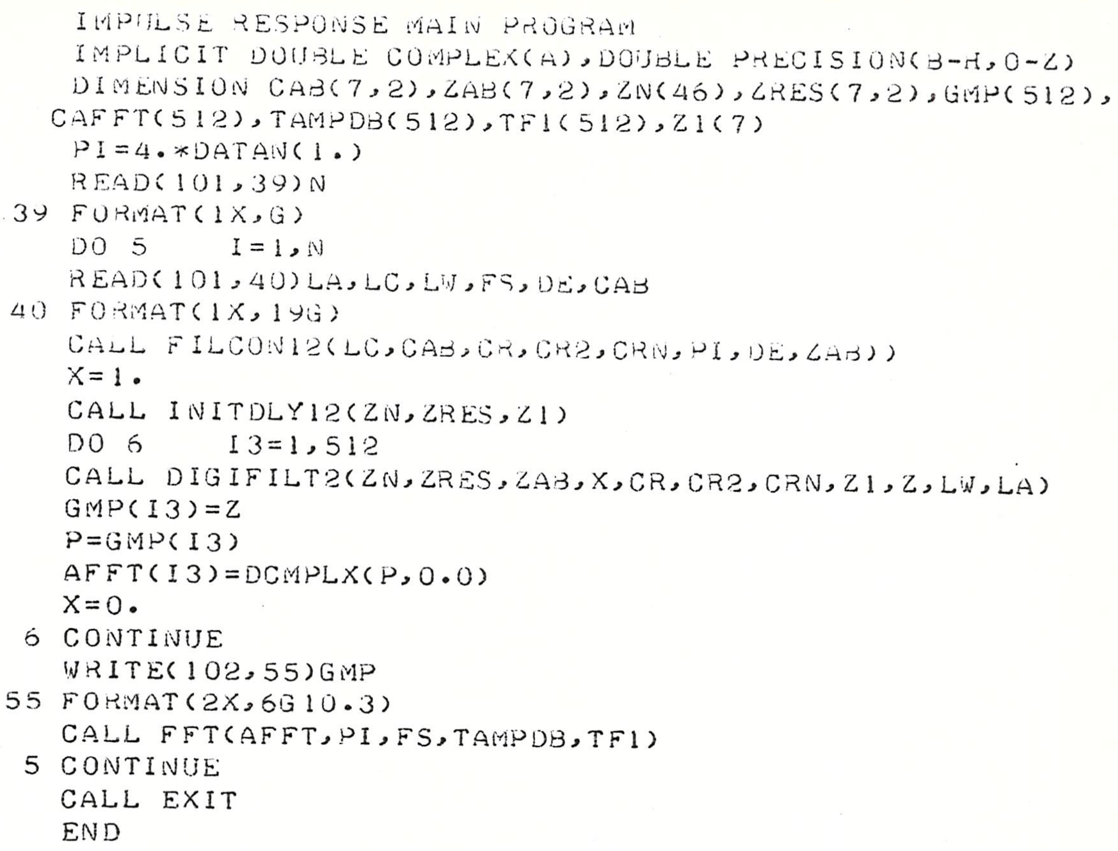
```
      BEST FIT ANALYSIS MAIN PROGRAM
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION CAB(7,2),ZAB(7,2),ZN(46),ZRES(7,2),Z1(7)
      PI=4.*DATAN(1.)
      READ(101,19)N
   19 FORMAT(1X,G)
      DO   5 I=1,N
      READ(101,40)FMIN,FMAX,FSTEP,FS,SA,RL,DE,LA,LC,LW
   40 FORMAT (1X,11G)
      READ(101,41)CAB
   41 FORMAT(1X,14G)
      CALL FILCON12(LC,CAB,CR,CR2,CRN,PI,DE,ZAB)
      DO   6 F=FMIN.FMAX,FSTEP
      CALL INITDLY12(ZN,ZRES,Z1)
      X=SA
      Y=SX=SX2=SY=SXY=SY2=SZ=SXZ=SYZ=SZ2=QS=0.
      IL=RL+460
      DO   7 I2=1,IL
      CALL DIGIFILT1(ZN,ZRES,CAB,X,CR,CR2,CRN,Z1,Z,LW,LA)
      IF(I2-RL)3,3,2
    2 CALL SIGMA(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,QS,X,Y,Z)
    3 X=SA*DCOS(2*PI*F*I2/FS)
    7 Y=-SA*DSIN(2*PI*F*I2/FS)
      CALL S REGRES(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,QS,PI,F)
    6 CONTINUE
    5 CONTINUE
      CALL EXIT
      END
```

Figure A8 Best Fit Analysis Main Program



```
      NOISE COMPARISON MAIN PROGRAM
      IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
      DIMENSION CAB(7,2),ZAB(7,2),ZN(46),ZRES(7,2),GMP(460),
    CCMP(460),DIFF(460)
      PI=4.*DATAN(1.)
      READ(101,20)N
   20 FORMAT(1X,I2)
      DO 7      I=1,N
      READ(101,21)F,FS,LA,LX,LC,LW,RL,DE,SA
   21 FORMAT(1X,10G)
      READ(101,22)CAB
   22 FORMAT(1X,14G)
      CALL FILCON3(LC,CAB,CR,CR2,CRN,PI,DE,ZAB)
      OUTPUT CR,CR2,CRN
      CALL INITDLY3(ZN,ZRES,Z1)
      X=SA
      Y=SX=SX2=SY=SXY=SY2=SZ=SXZ=SYZ=SZ2=QS=0
      IL=RL+460
      DO 9      I2=1,IL
      CALL DIGIFILT3(ZN,ZRES,ZAB,X,CR,CR2,CRN,Z1,Z,LW,LA)
      IF(I2-RL)3,3,2
    2 GMP(I2-RL)=Z
      CALL SIGMA(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,QS,X,Y,Z)
    3 X=DCOS(2*PI*F*I2/FS)*SA
    9 CONTINUE
      CALL SREGRES(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,QS,PI,F)
      CALL INITDLY3(ZN,ZRES,Z1)
      X=SA
      Y=SX=SX2=SY=SXY=SY2=SZ=SXZ=SYZ=SZ2=QS=0
      DO 8      I2=1,IL
      CALL DIGIFILT3(ZN,ZRES,ZAB,X,CR,CR2,CRN,Z1,Z,LX,LA)
      IF(I2-RL)5,5,6
    6 CMP(I2-RL)=Z
    5 X=DCOS(2*PI*F*I2/FS)*SA
      CALL SIGMA(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,QS,X,Y,Z)
    8 CONTINUE
      CALL SREGRES(SX,SX2,SY,SXY,SY2,SZ,SXZ,SYZ,SZ2,QS,PI,F)
      DO 1      I4=1,460
    1 DIFF(I4)=GMP(I4)-CMP(I4)
      DIFFS=0
      DO 4      I5=1,460
    4 DIFFS=DIFFS+DIFF(I5)**2
      DIFFO=DIFFS/460.
      RMSNOISE=10.*DLOG10(DIFFO)
      OUTPUT RMSNOISE
    7 CONTINUE
      CALL EXIT
      END
```

Figure A9 Noise Comparison Main Program



```
   IMPULSE RESPONSE MAIN PROGRAM
   IMPLICIT DOUBLE COMPLEX(A),DOUBLE PRECISION(B-H,O-Z)
   DIMENSION CAB(7,2),ZAB(7,2),ZN(46),ZRES(7,2),GMP(512),
  CAFFT(512),TAMPDB(512),TF1(512),Z1(7)
   PI=4.*DATAN(1.)
   READ(101,39)N
39 FORMAT(1X,G)
   DO 5     I=1,N
   READ(101,40)LA,LC,LW,FS,DE,CAB
40 FORMAT(1X,19G)
   CALL FILCON12(LC,CAB,CR,CR2,CRN,PI,DE,ZAB))
   X=1.
   CALL INITDLY12(ZN,ZRES,Z1)
   DO 6     I3=1,512
   CALL DIGIFILT2(ZN,ZRES,ZAB,X,CR,CR2,CRN,Z1,Z,LW,LA)
   GMP(I3)=Z
   P=GMP(I3)
   AFFT(I3)=DCMPLX(P,0.0)
   X=0.
 6 CONTINUE
   WRITE(102,55)GMP
55 FORMAT(2X,6G10.3)
   CALL FFT(AFFT,PI,FS,TAMPDB,TF1)
 5 CONTINUE
   CALL EXIT
   END
```

Figure A10. Impulse Response Main Program

APPENDIX B

DESKTOP CALCULATOR PROGRAM FOR TYPE III

BANDPASS FILTER

This program solves the pulse transfer function equation of a Type III bandpass filter,

$$H(z) = \frac{(1-r^N z^{-N})(1-rz^{-1})}{N} \sum_k \frac{(-1)^k 2|H(k)| \cos(\pi k/N)}{1-2r \cos(2\pi k/N)z^{-1}+r^2 z^{-2}}$$

and plots the interpolated frequency response are the range 200 Hz to 1800 Hz. By making suitable modifications to the program, a plot of the phase response is obtained.

Figure B1 shows the flow chart and method of calculation, Figure B2 gives the program listings, Figure 3 shows the gain response and Figure 4 the phase response.

The desktop calculator used was the Hewlett-Packard 9810, which requires the Mathematics and Plotter ROMS together with the Hewlett-Packard plotter, 9862.

STEP USER INSTRUCTION

1. Enter filter parameters.

Parameter	to Store	Example
Number of samples N	040	46
Sampling period T	041	1/4600
Number of resonators (< 9)	046	7
Coefficient wordlength	047	12
Damping factor exponent	048	9
Frequency of resonator having H(k) = 0 in upper stopband	070	1400
Frequency increment	071	10
Resonator spacing	072	100

2. Enter values of frequency samples 030-038

$(-1)^k |H(k)|$

eg

- .2	030
.7	031
- 1	032
1	033
- 1	034
.7	035
- .2	036

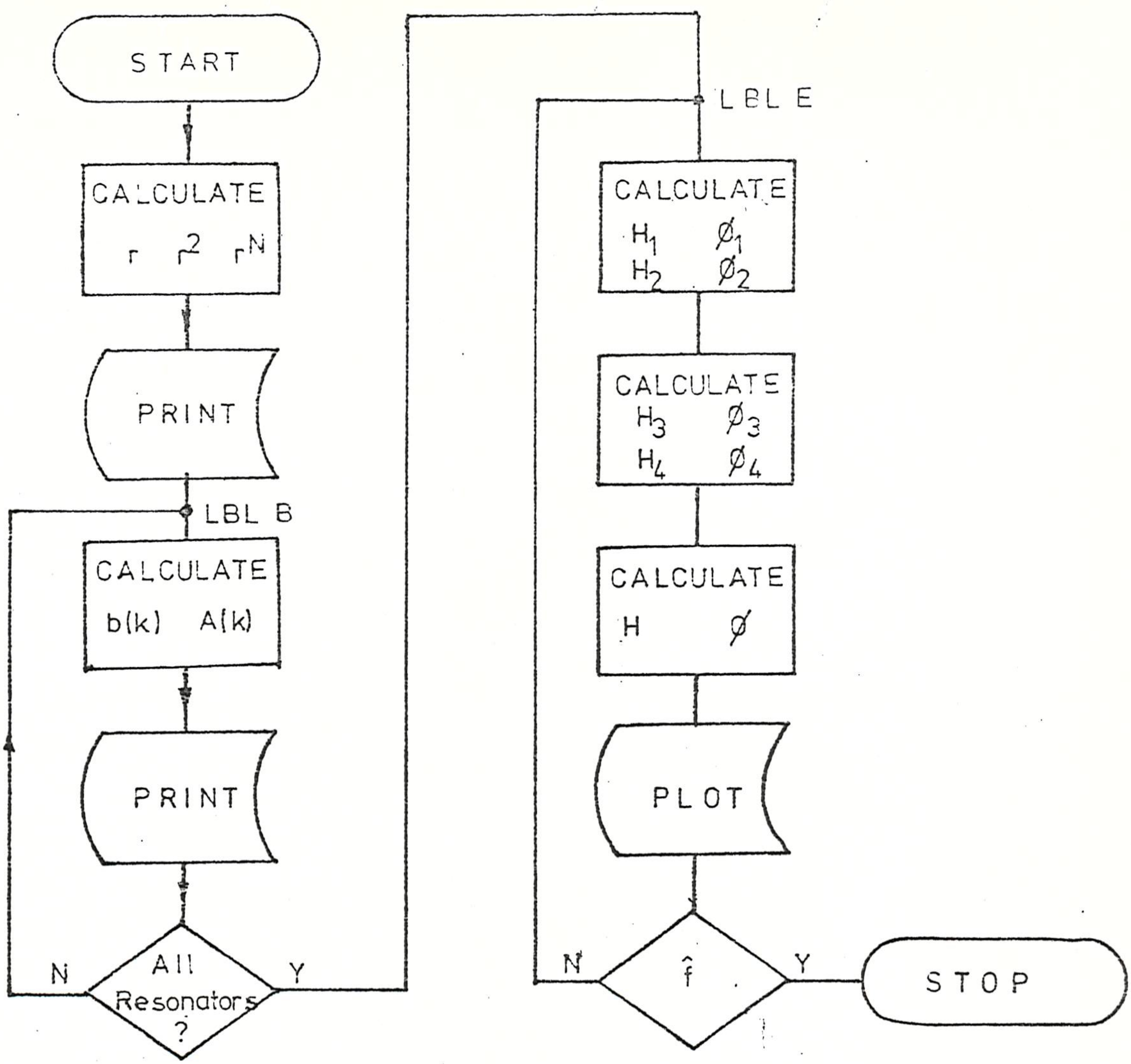
3. Enter start frequency to store a

4. GO TO LBL G

CONTINUE

The axes will be plotted

5. On completion of axes, enter the resonator frequencies pressing CONTINUE after each entry. A printout of coefficients will be given and plotting will commence immediately after the final printout.



f = frequency variable

$$1 - r^N z^{-N} \rightarrow [1 - r^N\cos(2\pi f TN)] - jr^N\sin(2\pi f TN) \begin{smallmatrix}\nearrow H_1 \\ \searrow \phi_1\end{smallmatrix}$$

$$1 - rz^{-1} \rightarrow [1 - r\cos(2\pi f T)] - j\sin(2\pi f T) \begin{smallmatrix}\nearrow H_2 \\ \searrow \phi_2\end{smallmatrix}$$

$$\frac{1}{1 - 2r\cos(2\pi k/N)z^{-1} + r^2 z^{-2}}$$

$$\rightarrow \frac{\cos(2\pi f T) + j\sin(2\pi f T)}{[(1+r^2)\cos(2\pi f T) - 2r\cos(2\pi k/N)] + j(1-r^2)\sin(2\pi f T)} \begin{smallmatrix}\nearrow H_3 \\ \rightarrow \phi_3 \\ \rightarrow H_4 \\ \searrow \phi_4\end{smallmatrix}$$

Figure B1 Program Flowchart

STEP	0	1	2	3	4	5	6	7	8	9
000	LBL	F	XFR	4	9	X	0	X<Y	GTO	LBL
001	+	CNT	GTO	LBL	-	LBL	+	0	.	5
002	+	DN	INT	XFR	DIV	4	9	S/R	LBL	-
003	0	.	5	-	DN	INT	XFR	DIV	4	9
004	S/R	LBL	G	CNT	1	0	UP	9	0	CHS
005	FMT	1	3	1	8	0	0	UP	2	0
006	0	FMT	1	2	0	UP	2	0	0	FMT
007	1	UP	5	0	FMT	1	5	0	UP	2
008	0	0	FMT	1	UP	5	FMT	1	6	K
009	2	0	1	0	XTO	0	0	1	0	2
010	0	XTO	0	0	2	0	3	0	XTO	0
011	0	3	XFR	4	8	CHS	UP	2	H	UP
012	1	XEY	-	YTO	4	3	DN	PNT	XFR	4
013	7	UP	2	H	XTO	4	9	XFR	4	3
014	XSQ	UP	F	XTO	4	4	PNT	XFR	4	0
015	UP	XFR	4	3	H	UP	F	XTO	4	5
016	PNT	PNT	LBL	B	STP	XTO	b	UP	2	X
017	π	X	DN	XFR	X	4	1	XTO	6	9
018	N	UP	2	X	XFR	4	3	X	2	DIV
019	F	UP	2	X	DN	XTO	IND	0	0	1
020	UP	b	PNT	DN	PNT	XFR	6	9	UP	2
021	DIV	DN	N	UP	2	X	DN	XFR	DIV	4
022	0	XFR	IND	X	0	0	3	UP	2	DIV
023	F	UP	2	X	DN	XTO	IND	0	0	2
024	PNT	PNT	1	XTO	+	0	0	1	XTO	+
025	0	0	2	XTO	+	0	0	3	XFR	7
026	2	XTO	+	b	b	UP	XFR	7	0	X=Y

*

Figure B2 Calculator Program

STEP	0	1	2	3	4	5	6	7	8	9
027	GTO	LBL	E	CNT	GTO	LBL	B	LBL	E	0
028	1	0	XTO	0	0	1	0	2	0	XTO
029	0	0	2	0	XTO	6	1	XTO	6	2
030	LBL	N	a	UP	2	X	π	X	DN	XFR
031	X	4	0	XFR	X	4	1	XTO	6	3
032	N	XTO	6	4	XFR	6	3	M	XTO	6
033	5	XFR	4	5	XFR	X	6	4	UP	1
034	XEY	-	XFR	6	5	XFR	X	4	5	XEY
035	A	XTO	5	0	YTO	5	1	XFR	6	3
036	XFR	DIV	4	0	XTO	6	6	N	XTO	6
037	7	XFR	6	6	M	XTO	6	8	XFR	4
038	3	XFR	X	6	7	UP	1	XEY	-	XFR
039	6	8	XFR	X	4	3	XEY	A	XTO	5
040	2	YTO	5	3	LBL	C	XFR	6	8	UP
041	XFR	6	7	A	XTO	5	4	YTO	0	5
042	5	1	XFR	+	4	4	XFR	X	6	7
043	XFR	IND	-	0	0	1	XTO	0	5	8
044	1	XFR	-	4	4	XFR	X	6	8	UP
045	XFR	5	8	A	XTO	5	6	YTO	5	7
046	XFR	5	0	XFR	X	5	2	XFR	X	5
047	4	XFR	DIV	5	6	XFR	IND	X	0	0
048	2	XTO	5	9	XFR	5	1	XFR	+	5
049	3	XFR	+	5	5	XFR	-	5	7	XTO
050	6	0	UP	XFR	5	9	B	XTO	+	6
051	1	YTO	+	0	6	2	1	XTO	+	0
052	0	1	XTO	+	0	0	2	1	0	XFR
053	+	4	6	UP	XFR	0	0	1	X=Y	GTO

Figure B2 Calculator Program(contd)

STEP	0	1	2	3	4	5	6	7	8	9	
054	LBL	D	CNT	GTO	LBL	C	LBL	D	XFR	6	
055	2	UP	XFR	6	1	A	K	4	UP	2	✳
056	0	X	a	FMT	1	DN	XFR	7	1	XTO	
057	+	a	GTO	LBL	E	STP	END				
004		LBL	H	CNT	π	UP	CHS	FMT	1	3	
005	1	8	0	0	UP	2	0	0	FMT	1	
006	2	0	UP	2	0	0	FMT	1	UP	5	
007	0	FMT	1	5	0	UP	2	0	0	FMT	
008	1	UP	π	FMT	1	6	CNT	CNT	CNT		
055							a	FMT	1	DN	
056	XFR	7	1	XTO	+	a	GTO	LBL	E	STP	
057	END										

✳ Modifications

Figure B2 Calculator Program(contd.)

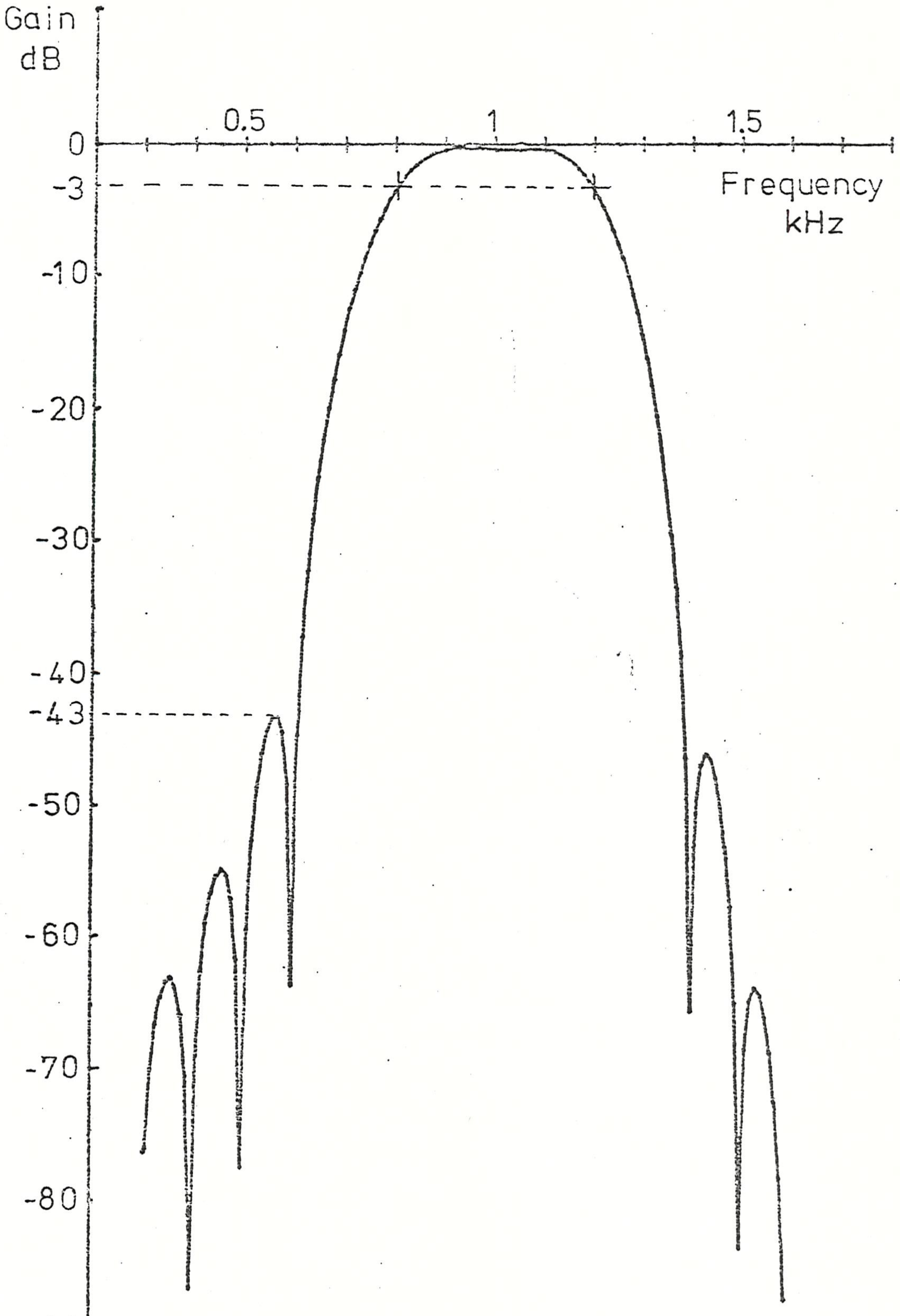


Figure B3 Frequency Response

compare with Figure 8.4b

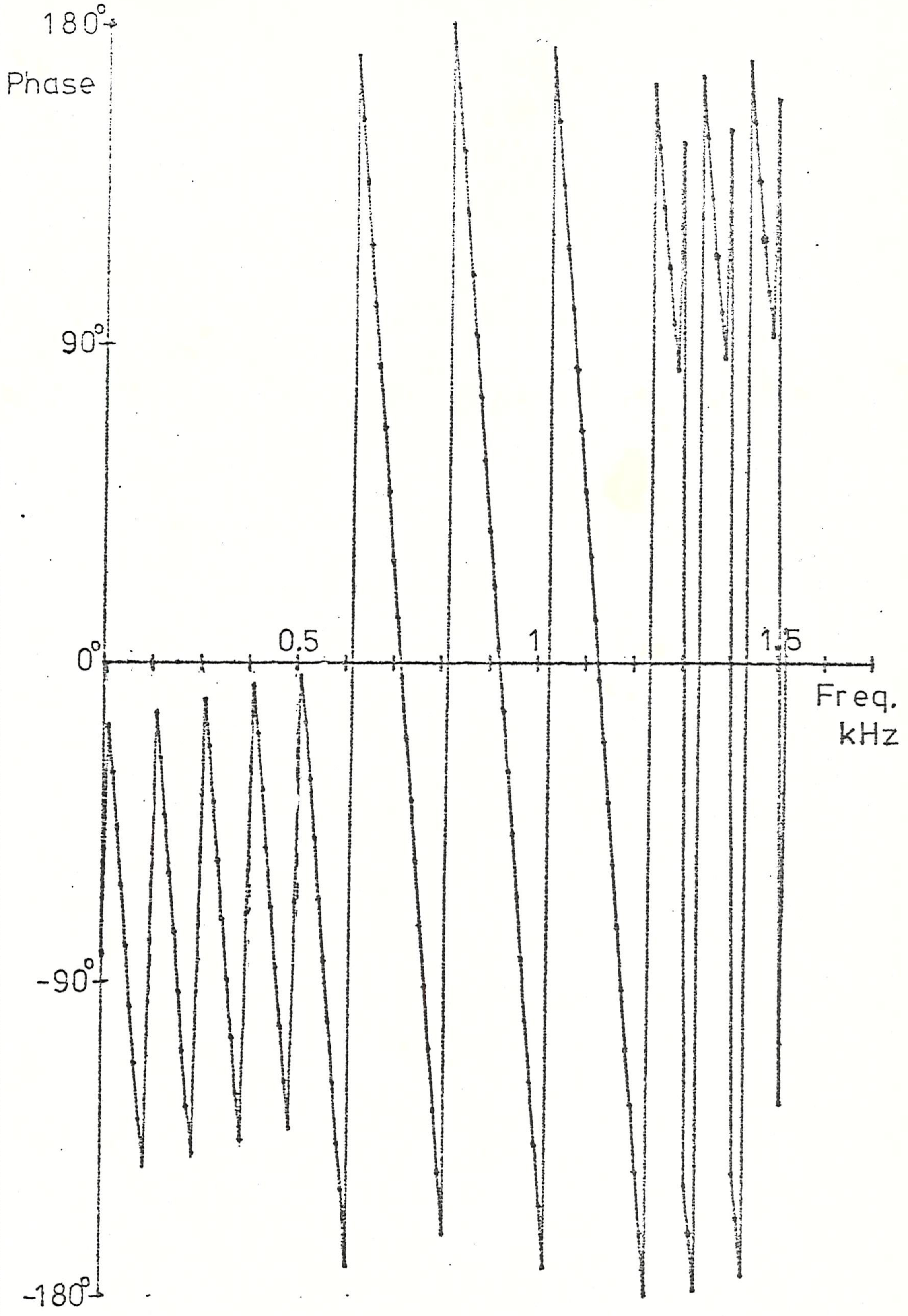


Figure B4 Phase Response