THE UNIVERSITY OF HULL

ANALYSIS AND SYNTHESIS OF DIGITAL ACTIVE NETWORKS

being a Thesis submitted for the Degree of

Doctor of Philosophy

in the University of Hull

by

Francis Geoffrey Armstrong Coupe, B.Sc. (Hons.)

September 1979

To Briony, Helen

and Andrew

ABSTRACT

The analysis of digital active networks is developed in this thesis, starting from the definitions of digital amplifiers and digital amplifier arrays and concluding with the presentation of general analysis techniques for N-port digital active networks. The analysis techniques are then tested by comparing the results of practical experiments with numerical evaluations of the derived transfer functions using a computer.

The basic techniques necessary for the synthesis of digital active networks are described with an example, and the thesis is concluded with a discussion of the advantages of digital active networks over their analogue equivalents.

ACKNOWLEDGEMENTS

I should like to thank my wife Briony for her patience and endurance while this thesis was being prepared.

I should also like to thank Dr. J. I. Sewell for supervising the work presented in this thesis, and to acknowledge SRC sponsorship for part of the period of preparation of this thesis.

Finally I should like to thank Mrs. Marian Smith for typing this thesis.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 OVERALL APPROACH

Analogue active networks and their associated mathematics have been very extensively investigated in recent years and may now be said to be classical. The same may also be said for digital filters and sampled data systems in general. The purpose of this thesis is not to explore any particular and small part of the above topics but to explore the possibilities of combining analogue active and passive networks with sampled data systems and digital filters to make a new range of circuit realisations possible.

One possible approach could have been to investigate the design of conventional digital filters with N input ports fed from voltage sensing A/D converters and N output ports feeding current generating D/A converters, completed by strapping the equivalent analogue inputs and outputs thus making an N-port network. This approach, however, only solves half the original problem, neatly avoiding the problem of mixing passive or active analogue components with digital amplifiers. The approach chosen in this thesis was therefore more general and could easily be applied to the simpler all-digital matrix realisation.

It was decided to tackle this problem by simulating the digital admittance matrix using digital transadmittance amplifiers mixed with ordinary passive analogue components, thus marrying digital active networks to analogue active and passive networks. A commensurate mathematical approach was also developed to describe this matrix and any of the other matrices which may be derived from it, in particular the digital impedance matrix.

The most significant advantage of simulating digital active networks is that the individual digital amplifiers may be made deliberately

non-linear or parametric and thus functions can be simulated which are just not feasible in ordinary analogue networks.

1.2 SURVEY OF RELEVANT WORK

The previous work published in the specific field of research considered in this thesis is sparse, though Pim and Bullingham [1] [2] have presented a technique for the simulation of digital components which is different from that presented in Chapter 2. However Pim and Bullingham cite no previous research in this field.

The synthesis of active networks from the admittance matrix has been considered by Yanagisawa and Kanbayashi [3] [4]. In their first paper they considered synthesis based on the reversing of the process of nullator-norator analysis. In their second paper they base their synthesis technique on the scattering matrix S which is first derived for the required network and the admittance matrix Y calculated from S. Having then obtained Y they use the nullator-norator synthesis technique of their first paper to produce the required network.

Now it is perfectly possible to apply these synthesis techniques to digital active networks, and this is discussed in Chapter 10.

As this thesis is concerned with combining analogue and digital networks it is necessary to consider the definition of the Z-transform used in the analyses [5]. The work presented here uses the standard definition of the Z-transform [6] which is described in Appendix A. This is because this problem involves analysing circuits containing ordinary analogue components in conjunction with digital amplifiers, rather than trying to emulate the operation of an analogue active or passive filter by a digital or sampled data filter. Thus the capacitively loaded 2-port digital gyrator analysed in Chapter 4 will show an impulse invariant response when compared with an analogue gyrator.

It would not have been correct to have used the 'bilinear' Z-transform [7] [8] or the 'matched' Z-transform [9] for the digital

gyrator analysis because the resulting z-plane polynomials would not have described what actually occurs if analogue active or passive components are mixed with digital amplifiers in a network.

However when digital amplifier arrays that include digital filters are to be used to model an analogue active or passive network, then it may be better to use the 'bilinear' or 'matched' Z-transform rather than the 'standard' Z-transform.

If the bilinear Z-transform had been used then the frequency response of the original analogue networks could have been more closely approximated. The matched Z-transform could have been chosen, and this would have accurately matched the pole and zero locations of the digital active network with its analogue equivalent.

The standard Z-transform has been described in many papers and books, such as [6], [10] and [11].

A useful advantage of using the standard Z-transform is that the modified or advanced Z-transform may then be used to compute the time domain output of a sampled data machine or digital filter between sampling pulses, as described in [11].

Another problem that arises in any sampled data system or digital filter is that of non-ideal sampling, that is when the sample pulse is of a distinctly finite width. However [12] shows that provided the sampler is followed by a hold stage this problem does not matter, and this is so in this case.

The effect of using digital filters with inevitably finite register lengths introduces either round-off or truncation errors into the digital filters output, which may in turn be interpreted as a form of noise. This problem has been extensively discussed in papers such as [13] [14] [15] [16] and [17]. However it is shown in Chapter 3 that the coefficients in digital active networks can be derived from passive components which do not have their component values

quantised. Nevertheless there are quantisers associated with every A/D converter in each digital amplifer and Chapter 3 shows that a signal may experience repeated amplitude quantisation in a network such as the digital gyrator thus introducing an effect similar to coefficient quantisation or round-off error. This in turn may give rise to limit cycle oscillations, and this was in fact observed in the machine described in Chapter 6 when it was connected as a capacitively loaded 2-port digital gyrator.

1.3 CHAPTER DESCRIPTIONS

Chapter 2 introduces and describes the stages necessary to the construction of a digital amplifier. The conservation of units under the sampling process and the transformability of analogue transfer functions are explored in depth. The effect of quantisation noise in a digital amplifier is then studied, followed by a description of techniques to enhance the digital amplifier. The digital amplifier is then applied to the simulation of analogue circuit components and networks.

Chapter 3 introduces and describes digital active arrays. A 2-port digital active network is analysed and this analysis is then extended to an N-port network. In particular the problem of finding the Z-transform of an N-port admittance matrix when the Z-transform of some of the elements cannot be directly found has been solved for the general case. Matrix stability, limit cycle noise amplitude and element resolution have all been considered.

Chapter 4 contains an example of the analysis techniques derived in Chapter 3. The digital admittance and impedance matrices are derived for a capacitively loaded 2-port digital gyrator. The stability and limit cycle noise amplitude is then considered for this example.

Chapter 5 briefly describes the four FORTRAN computer programs written to analyse the expressions derived in Chapter 4.

Chapter 6 describes the construction and use of a digital machine

built as part of the research work to verify the mathematical analysis of Chapters 3 and 4 by practical experiment.

Chapter 7 describes the practical results obtained by making measurements on the digital machine described in Chapter 6 when it was set up to simulate a capacitively loaded 2-port digital gyrator. The presence of limit cycle oscillations is noted, and the practical results are corrected to eliminate the effect of the presence of this noise.

Chapter 8 describes the results obtained from the computer programs for the case of the capacitively loaded 2-port digital gyrator for all the elements of the 2-port digital admittance and impedance matrices. The quantisation voltage transfer matrix is also evaluated using the same parameters as in the digital admittance and impedance matrices.

Chapter 9 compares the experimental and computer results and shows that the mathematical analysis is correct.

Chapter 10 shows that the analysis of digital active networks derived in Chapter 3 can be applied to help in the synthesis of these networks.

Chapter 11 summarises the work presented, and possible future areas of research are suggested, thus concluding this thesis.

1.4 GENERAL POINTS

In this thesis the term 'digital' is used to indicate an analogue to digital (A/D) conversion process which incorporates sampling and amplitude quantisation in the signal path, and this was in fact used to make the digital amplifiers. Nevertheless the analysis presented is effectively true for sample data systems where only sampling is used without the presence of an A/D converter which would incorporate a quantisation stage. If sample data amplifiers were to be used then limit cycle noise would entirely vanish.

In the sampling processes described in this thesis, all the sampling is assumed to be uniform and synchronous.

In order to avoid confusion it is also necessary to clarify the use of the letter z in this thesis. The Z-transform is used extensively such that h(z) would be an arbitary function of z, the Z-transform variable. However when considering the Z-transform of an impedance such as z_{11} (z), then the z's need to be clearly distinguished. By the general conventions in electronic engineering it was not felt wise to either redefine the Z-transform variable or the letter used for impedance. Thus z_{11} (z) is the digital input impedance at port 1 of a network in the z-plane.

CHAPTER 2

DIGITAL AMPLIFIERS

2.1 AMPLIFIER COMPARISON

The conventional analogue amplifier can only perform two operations in its input, namely scaling and convolution. The scaling factor may or may not be unitless depending upon which of the four types of amplifier is involved. The four types of amplifier and their idealised input and output impedances are listed in Table 2.1.

However there are two other devices which may be added to the signal path of any of these four amplifiers, a sampler and a quantiser. Introducing a sampler makes the machine operate in discrete time thereby becoming a sampled data machine and adding a quantiser creates a digital amplifier. Theoretically a quantiser could be added on its own, but this would be difficult to realise in practice, and would only decrease the stability of a network including this amplifier.

When a sampler and a quantiser is included in the signal path to make a digital amplifier then quantisation is achieved as an integral part of analogue-to-digital conversion (A/D). (See Appendix A for an analysis of sampling and Appendix C for an analysis of quantisation.) After signal processing the signal has to undergo digital-to-analogue conversion (D/A). Thus the signal processing is achieved using conventional digital logic.

2.2. DIGITAL AMPLIFIER SIGNAL PATH

Fig. 2.1 shows the block diagram of the signal path of a digital amplifier. The amplifier is really a hybrid of analogue and digital stages. Although the circuit blocks are described individually it can occur that in practice two or more stages are merged into one practical circuit.

	VOLTAGE AMPLIFIER	CURRENT AMPLIFIER	TRANSIMPEDANCE AMPLIFIER	TRANSADMITTANCE AMPLIFIER
Theoretical SENSOR Input impedance	∞	0	0	∞
Theoretical GENERATOR Output impedance	0	∞	0	∞

TABLE 2.1

Amplifier Comparison

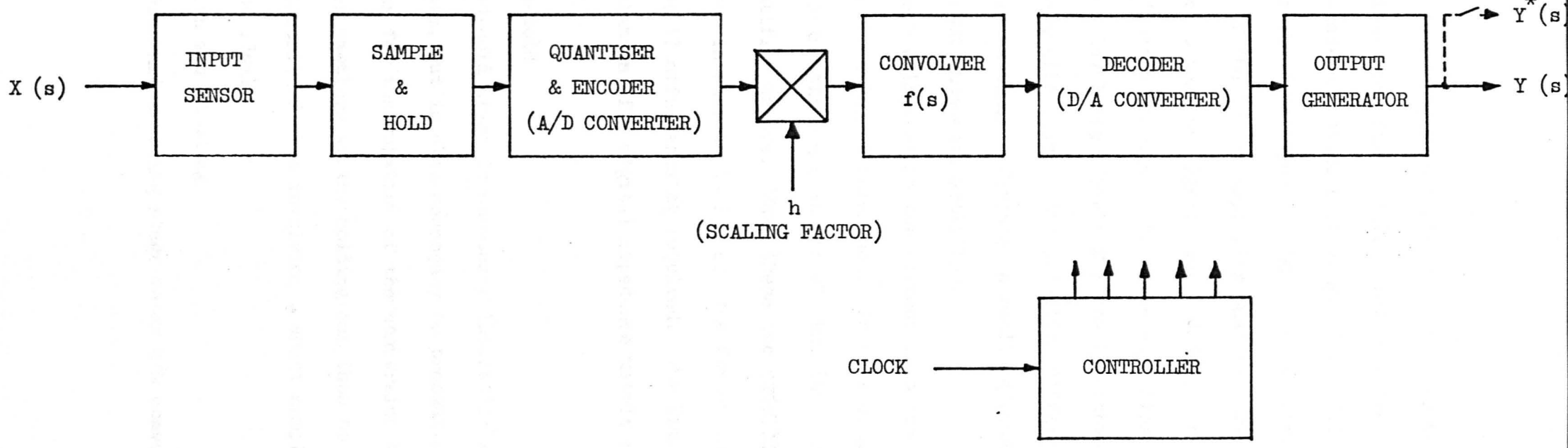


FIG. 2.1 Digital Amplifier Signal Path

2.2.1 Sensor

The sensor is an analogue amplifier with the correct input impedance for the type of digital amplifier required and an output impedance suitable to drive the sample-and-hold stage following. Table 2.1 summarises the input impedances for the four types of digital amplifier.

The bandwidth of the sensor amplifier must be very much greater than the sample rate so that no significant modification of the digital amplifier transfer function occurs. The sense amplifier may also amplify the input signal to a level appropriate for the A/D converter.

A voltage sensor will measure the potential difference between the input node and an external reference (normally ground) and is a conventional high input impedance amplifier.

A current sensor will measure the current into the input node, or out of that node to the external reference. In this simple amplifier these two possible connections are identical, but this is not so in the case of digital amplifier arrays. When these two cases do not merge the latter current sensor is easy to design but the former is much more difficult due to the floating sensing required. The limitation which this imposes on the simulation of a digital impedance matrix is discussed in Section 3.3.

2.2.2 Sample-and-Hold

The sample-and-hold stage is necessary theoretically to make this a sampled data system, but is also necessary to present a stable input to the A/D converter for the duration of the conversion time. It is normal to use series samplers as the holding can then be done in a capacitor; the dual circuit of this involving a shunt sampler and an inductor is far from ideal.

2.2.3 Quantisation and Encoding

The quantisation and encoding stage is an A/D converter normally

arranged to produce a signed binary output. It is theoretically necessary to make the digital amplifier. The transfer function is a staircase which in this case must be linearly weighted because of the succeeding mathematical operations to be performed. The quantisation process is discussed in Appendix C. Although it is possible to build an analogue network with a staircase transfer function, in this case the instantaneous signal level captured by the sample-and-hold stage is converted to the equivalent signed binary number representing the interval in which that signal level fell.

2.2.4 Scaling Factor

The scaling factor is included for two purposes. All numerical ratios between the input and output of the stages of the amplifier are gathered together as a hypothetical multiplier. A real multiplier may also be added to allow variable scaling of the amplifier or parametric operation, and would be implemented using digital logic. The types of multiplier which could be used are discussed in Sections 6.3.3 and 6.8.4.

2.2.5 Convolver

The convolver may include a digital filter, but must include all the lumped time delays in the signal path, which in any actual example are determined by inspection of the circuit arrangement. The digital filter would be implemented using digital logic.

2.2.6 Decoder

The decoder is a digital to analogue (D/A) converter. The various techniques for D/A conversion are described and discussed in Chapter 6.

2.2.7 Generator

The generator is an analogue amplifier with the correct output impedance, as summarised in Table 2.1. The generator amplifier bandwidth must be very much larger than the sample rate and must also scale the output signal range of the decoder to the actual dynamic range required. It is often the case that the decoder and generator can be

merged.

2.2.8 Controller

The controller is necessary to sequence the processing of the data in the signal path and is driven by a clock signal.

2.3 DIGITAL AMPLIFIER ANALYSIS

The block diagram of a simple sample data system is shown in Fig. 2.2. T_s is the sampling period, and H(s) is assumed to be the transfer function for an active or passive analogue network. The transfer function will be unitless if the input and output variables (x, y) have the same units, namely voltage or current. However H(s) will not be unitless if x and y represent variables with opposite units implying that H(s) will have units of impedance or admittance.

2.3.1 Conservation of Units

From Appendix A, equation (A.8) the Laplace transform of the input signal after sampling will be:

$$X^*(s) = \frac{1}{T_s} \sum_{r=-\infty}^{\infty} X(s + j\,r\,w_s) \qquad (2.1)$$

where the asterisk implies that the variable has been sampled, and w_s is the angular sampling frequency. It should be noted that the units of $X^*(s)$ have been reduced by dividing by the sampling period T_s.

Thus the Laplace transform of the output variable will be:

$$Y(s) = H(s)\,X^*(s) \qquad (2.2)$$

In order to analyse the state of Y(s) at the sampling instants it is expedient to hypothetically sample Y(s) thus:

$$Y^*(s) = \left[H(s)\ X^*(s) \right]^* \qquad (2.3)$$

and this is conventionally rewritten as:

$$Y^*(s) = H^*(s)\ X^*(s) \qquad (2.4)$$

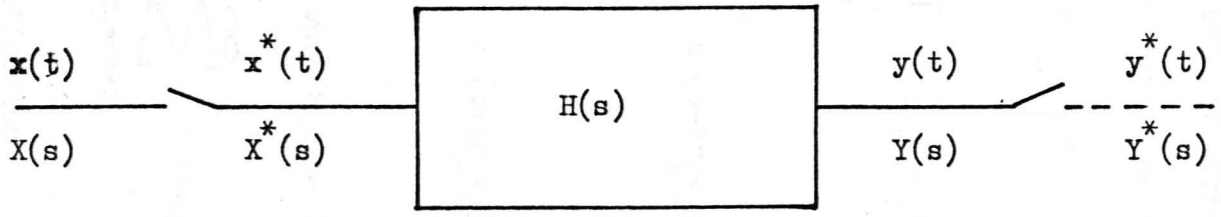


FIG. 2.2 Sample Data System

However the action of sampling has divided each variable in equation (2.4) by the sample period T_s and thus (2.4) is no longer consistent in units. It is necessary therefore to rewrite equation (2.4) as:

$$Y^*(s) = T_s H^*(s) \; X^*(s) \tag{2.5}$$

and the justification for this will now be derived. This point is mentioned by Blackman [18] but not derived.

Equation (2.3) may be rewritten as using equation (A.8) thus:

$$Y^*(s) = \left[H(s) \; X^*(s) \right]^*$$

$$= \frac{1}{T_s} \sum_{r_1=-\infty}^{\infty} \left[H(s+jr_1w_s) \; X^*(s+jr_1w_s) \right]$$

$$= \frac{1}{T_s} \sum_{r_1=-\infty}^{\infty} \left[H(s+jr_1w_s) \frac{1}{T_s} \sum_{r_2=-\infty}^{\infty} X(s+j(r_1+r_2)w_s) \right]$$

$$= \frac{1}{T_s} \sum_{r_1=-\infty}^{\infty} H(s+jr_1w_s) \frac{1}{T_s} \sum_{r_1=-\infty}^{\infty} \sum_{r_2=-\infty}^{\infty} X(s+j(r_1+r_2)w_s) \tag{2.6}$$

In the trivial case when $H(s) = 1$, equation (2.6) becomes:

$$\left[X^*(s) \right]^* = \frac{1}{T_s^2} \sum_{r_1=-\infty}^{\infty} \sum_{r_2=-\infty}^{\infty} X(s+j(r_1+r_2)w_s) \tag{2.7}$$

$$= X^*(s) \text{ by definition}$$

This result may be applied to equation (2.6) thus:

$$Y^*(s) = \frac{1}{T_s} \sum_{r_1=-\infty}^{\infty} H(s+jr_1w_s) \; T_s \; X^*(s)$$

$$= T_s H^*(s) \; X^*(s) \tag{2.8}$$

and equation (2.5) is therefore justified.

2.3.2 Conditions for Transformability

Equation (2.8) may now be rewritten as a pulse transfer function (PTF) thus:

$$\frac{Y^*(s)}{X^*(s)} = T_s H^*(s) \tag{2.9}$$

The Z-transform of this PTF may be taken provided $H^*(s)$ represents a convergent series. Now H(s) may be defined as:

$$H(s) = \frac{\sum_{k=o}^{n} a_k s^k}{\sum_{k=o}^{m} b_k s^k} \tag{2.10}$$

After sampling and using equation (A.8) H(s) becomes:

$$H^*(s) = \frac{1}{T_s} \sum_{r=-\infty}^{\infty} \left[\frac{\sum_{k=o}^{n} a_k (s+jrw_s)^k}{\sum_{k=o}^{m} b_k (s+jrw_s)^k} \right] \tag{2.11}$$

Equation (2.10) may be divided out thus:

$$H(s) = \sum_{k=o}^{n-m} c_k s^k + \sum_{k=1}^{\infty} d_k s^{-k} \qquad \text{when } n > m \tag{2.12}$$

$$= c_o + \sum_{k=1}^{\infty} d_k s^{-k} \qquad \text{when } n = m \tag{2.13}$$

$$= 0 + \sum_{k=1}^{\infty} d_k s^{-k} \qquad \text{when } n < m \tag{2.14}$$

QUOTIENT + REMAINDER

The order of the remainder polynomial will be finite only if the denominator polynomial in equation (2.10) is a factor of the numerator polynomial.

To find $H^*(s)$, the quotient and remainder may be transformed independently but must first be proved to be transformable. To do this the transformed expression must be proved to consist of a convergent series.

From equation (2.12), with $n > m$, the quotient expression is:

$$\sum_{k=0}^{n-m} c_k s^k \tag{2.15}$$

After transforming this becomes:

$$\sum_{k=0}^{n-m} c_k \cdot \frac{1}{T_s} \sum_{r=-\infty}^{\infty} (s + j\, rw_s)^k$$

which may be rewritten thus:

$$\sum_{k=0}^{n-m} c_k \frac{1}{T_s} \left[\sum_{r=0}^{\infty} (s + j\, rw_s)^k + \sum_{r=0}^{\infty} (s - j\, rw_s)^k - s^k \right] \tag{2.16}$$

If both the terms in the square brackets are convergent, then so is the whole expression. Hence consider the first term of expression (2.16):

$$\sum_{r=0}^{\infty} (s + j\, rw_s)^k \tag{2.17}$$

Now apply the ratio test [19] to expression (2.17) and let the ratio of the $(r+1)^{th}$ and r^{th} terms be R where, by definition,

$$R = |R| \ (\cos \Theta + j \sin \Theta)$$

Now:

$$R = \left(\frac{s + j\,(r+1)w_s}{s + j\,rw_s} \right)^k \tag{2.18}$$

$$|R| = \left(\frac{\sigma^2 + (w + (r+1)w_s)^2}{\sigma^2 + (w + rw_s)^2} \right)^{k/2} \tag{2.19}$$

$$\theta = k \left[\mathrm{Tan}^{-1} \left(\frac{w + (r+1)w_s}{\sigma} \right) - \mathrm{Tan}^{-1} \left(\frac{w + rw_s}{\sigma} \right) \right] \tag{2.20}$$

The limit when $r \Rightarrow \infty$ must now be taken for the real and imaginary parts of R separately by finding the limits of $|R|$ and θ.

$$\underset{r \Rightarrow \infty}{\mathrm{Lt}} \; (|R|) = 1 \tag{2.21}$$

$$\underset{r \Rightarrow \infty}{\mathrm{Lt}} \; (\theta) = 0 \tag{2.22}$$

Hence:

$$\underset{r \Rightarrow \infty}{\mathrm{Lt}} \; (R) = 1 + j\,0 = \underset{r \Rightarrow \infty}{\mathrm{Lt}} \; (|R|) \tag{2.23}$$

This implies that the imaginary series is absolutely convergent but leaving the convergence of the real series undefined.

However, from reference [19] , the $(r+1)^{th}$ is always slightly greater than the r^{th} term, namely:

$$\sigma^2 + (w + (r+1)w_s)^2 > \sigma^2 + (w + rw_s)^2$$

As all variables are positive this reduces to:

$$w_s^2\,(2r+1) + 2ww_s > 0 \tag{2.24}$$

which now must be valid. Hence the real series is divergent.

The second term in the square brackets in expression (2.16) can now be considered, namely:

$$\sum_{r=0}^{\infty} (s - j\,rw_s)^k \tag{2.25}$$

Following the argument developed for expression (2.17):

$$R = \left(\frac{s - j\,(r+1)w_s}{s - j\,rw_s} \right)^{k} \tag{2.26}$$

$$|R| = \left(\frac{\sigma^2 + (w + (r+1)w_s)^2}{\sigma^2 + (w + rw_s)^2} \right)^{k/2} \tag{2.27}$$

$$\theta = -k \left[\mathrm{Tan}^{-1} \left(\frac{w + (r+1)w_s}{\sigma} \right) - \mathrm{Tan}^{-1} \left(\frac{w + rw_s}{\sigma} \right) \right] \tag{2.28}$$

Now taking limits gives:

$$\lim_{r \to \infty} \left(|R| \right) = 1 \tag{2.29}$$

$$\lim_{r \to \infty} \left(\theta \right) = 0 \tag{2.30}$$

This is identical with the previous result and hence the real series diverges and the imaginary series converges. Hence, overall, expression (2.16) is divergent, and thus the Z-transform of expression (2.15) is impossible.

From equation (2.12) the remainder expression is:

$$\sum_{k=1}^{\infty} d_k s^{-k} \tag{2.31}$$

After transforming this becomes:

$$\sum_{k=1}^{\infty} d_k \frac{1}{T_s} \sum_{r=-\infty}^{\infty} (s + jrw_s)^{-k}$$

which may be rewritten thus:

$$\sum_{k=1}^{\infty} d_k \frac{1}{T_s} \left[\sum_{r=0}^{\infty} (s + jrw_s)^{-k} + \sum_{r=0}^{\infty} (s - jrw_s)^{-k} - s^{-k} \right] \tag{2.32}$$

Again, both terms in the square brackets must be tested for convergence. Consider first:

$$\sum_{r=0}^{\infty} (s + j\, rw_s)^{-k} \tag{2.33}$$

Following the same argument developed for expression (2.17):

$$R = \left(\frac{s + j\,(r+1)\, w_s}{s + j\, rw_s} \right)^{-k} \tag{2.34}$$

Hence:

$$|R| = \left(\frac{\sigma^2 + (w + rw_s)^2}{\sigma^2 + (w + (r+1)w_s)^2} \right)^{k/2} \tag{2.35}$$

and the arg. is:

$$\theta = -k \left[\mathrm{Tan}^{-1} \left(\frac{w + (r+1)w_s}{\sigma} \right) - \mathrm{Tan}^{-1} \left(\frac{w + rw_s}{\sigma} \right) \right] \tag{2.36}$$

Now taking the limits gives:

$$\underset{r \to \infty}{\mathrm{Lt}} \; (\, |R| \,) = 1 \tag{2.37}$$

$$\underset{r \to \infty}{\mathrm{Lt}} \; (\, \theta \,) = 0 \tag{2.38}$$

Thus the imaginary series is again convergent. However the $(r+1)^{th}$ term is not now greater than the r^{th} term thus:

$$\sigma^2 + (w + rw_s)^2 < \sigma^2 + (w + (r+1)w_s)^2$$

which reduces to:

$$w_s^2 \, (2r+1) + 2ww_s < 0 \tag{2.39}$$

As all the variables are positive, by definition, this inequality is not true and hence the real series is now convergent.

Now consider the second term in the square brackets in expression (2.32), namely:

$$\sum_{r=0}^{\infty} (s - j\,rw_s)^{-k} \qquad (2.40)$$

Following the same argument as developed for expression (2.25):

$$R = \left(\frac{s + j\,(r+1)w_s}{s + j\,rw_s} \right)^{-k} \qquad (2.41)$$

$$|R| = \left(\frac{\sigma^2 + (w + rw_s)^2}{\sigma^2 + (w + (r+1)w_s)^2} \right)^{k/2} \qquad (2.42)$$

$$\theta = k \left[\mathrm{Tan}^{-1} \left(\frac{w + (r+1)w_s}{\sigma} \right) - \mathrm{Tan}^{-1} \left(\frac{w + rw_s}{\sigma} \right) \right] \qquad (2.43)$$

Now taking the limits gives:

$$\lim_{r \Rightarrow \infty} \left(|R| \right) = 1 \qquad (2.44)$$

$$\lim_{r \Rightarrow \infty} \left(\theta \right) = 0 \qquad (2.45)$$

Again, both the real and imaginary series converge and thus the remainder expression is unconditionally convergent.

Thus the Z-transform of $H^*(s)$ can only be found when $n \leqslant m$ because $H^*(s)$ then represents a convergent series, that is the numerator polynomial order does not exceed the denominator polynomial order.

It is further possible to say that $\left[1 / H(s) \right]^*$ may be found provided that $m \leqslant n$.

2.3.3 Rational Transfer Function Sampling

It is also necessary to investigate the transformability of the ratio of two transfer functions when the individual transfer functions themselves may or may not be transformable.

Now let the rational transfer function of H(s) now be:

$$H(s) = A(s) \;/\; B(s)$$

$$H(s) = \frac{\sum_{k=0}^{n_A} a_{Ak} s^k}{\sum_{k=0}^{m_A} b_{Ak} s^k} \cdot \frac{\sum_{k=0}^{n_B} a_{Bk} s^k}{\sum_{k=0}^{m_B} b_{Bk} s^k}$$

$$= \frac{\sum_{k=0}^{n_A+m_B} c_k s^k}{\sum_{k=0}^{n_B+m_A} d_k s^k} \qquad (2.46)$$

For the Z-transform of $H^*(s)$ to be able to be found, and from Section 2.3.2:

$$n_A + m_B \leqslant m_A + n_B$$

or

$$p = \frac{n_A + m_B}{m_A + n_B} \leqslant 1 \qquad (2.47)$$

Table 2.2 lists the conditions under which the Z-transform of $H^*(s)$ may be found. It should be particularly noted that H(s) is not necessarily transformable when $A^*(s)$ and $B^*(s)$ are individually transformable, $A^*(s)$ should be transformable but $B^*(s)$ untransformable.

2.4 PRACTICAL DIGITAL AMPLIFIER

A practical and typical digital amplifier as shown in Fig. 2.1 may now be analysed. The following assumptions will be made:

2.4.1 Input

The input signal will be a voltage, thereby replacing X(s) by V(s), and requiring the input sensor to have a theoretically infinite input impedance.

A(s)	B(s)	p	H(s) TRANSFORMABLE
$n_A > m_A$	$n_B > m_B$	Undefined	Possibly
$n_A > m_A$	$n_B \leqslant m_B$	> 1	No
$n_A \leqslant m_A$	$n_B > m_B$	< 1	Yes
$n_A \leqslant m_A$	$n_B \leqslant m_B$	Undefined	Possibly

TABLE 2.2

Rational Transfer Function Transformability

2.4.2 Scaling Factor

The scaling factor will have units of Siemens (mhos) as the output signal will be a current, and h will therefore be replaced by a conductance g. An actual multiplier will also be included with an external input, via its own A/D converter.

2.4.3 Convolver

The convolver must gather together the transfer function of the sample-and-hold stage (zero-order hold) and the overall time delay in the signal path. Hence:

$$f(s) = \underbrace{\frac{(1 - \exp(-s T_s))}{s}}_{\text{ZERO-ORDER HOLD}} \underbrace{\exp(-sk T_s)}_{\text{TIME DELAY}} \qquad (2.48)$$

where T_s is the uniform sampling period, and kT_s is the fractional time time delay in the signal path and $k > 0$.

2.4.4 Generator

The output current generator has a theoretically infinite output impedance and Y(s) is therefore replaced by I(s).

2.4.5 Pulse Transfer Function (PTF)

From the results of Section 2.3.2, $f^*(s)$ can be found directly. Hence equation (2.48) becomes:

$$f(z) = (1-z^{-1})\,(1-z^{-1})^{-1} z^{-k} = z^{-k} \qquad (2.49)$$

Hence from equation (2.49) the PTF becomes:

$$\frac{I(z)}{V(z)} = g z^{-k} = y(z) \qquad (2.50)$$

where $z = \exp(sT_s)$ and y(z) is a digital transadmittance.

From equation (2.51) it can be seen that the PTF is a function of g, k and T_s only. The sample-and-hold circuit has no effect in this case on the PTF and the PTF has a single pole at the origin of the z-plane.

This basic digital amplifier will be used to introduce the digital admittance matrix in Chapter 3.

2.5 QUANTISATION NOISE

The actions and effects of sampling, that is conversion to discrete time, have been considered. In a practical realisation A/D conversion will be necessary if a digital amplifier is to be used rather than a sampled data amplifier and this process will produce amplitude quantisation.

The effect of this quantisation process may be represented by a noise signal adding to the input signal [20]. Fig. 2.3 shows this arrangement.

Thus from Fig. 2.3:

$$I(s) = g(s)\left(V_s(s) + V_N(s) \right)^*$$

$$= I_s(s) + I_N(s) \qquad (2.51)$$

where:

$$V_N = \sqrt{V_{STEP}^2 / 12} \qquad (2.52)$$

Hypothetically sampling the output current and taking the Z-transform gives:

$$I_s(z) + I_N(z) = g(z)\left(V_s(z) + V_N(z) \right) \qquad (2.53)$$

and:

$$I_N(z) = g(z)\ V_N(z)$$

Hence the total output current contains a quantisation noise component $I_N(z)$.

2.6 MODIFIED DIGITAL AMPLIFIERS

There are various modifications which can easily be incorporated in the signal path of the basic digital amplifier.

2.6.1 Combined Inputs

The output of the A/D converter in the signal path of a digital

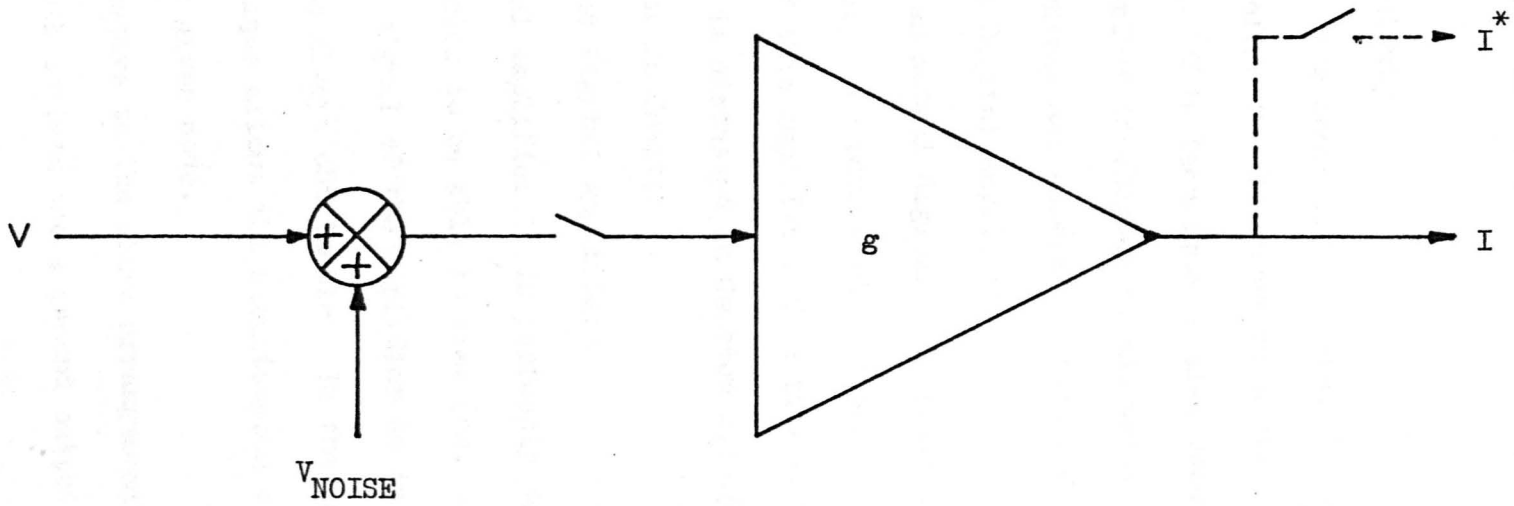


FIG. 2.3 Quantisation Noise Input

amplifier may be used to feed more than one signal path with no intrinsic degradation of the amplifiers performance. This technique is exploited to make the Digital Active Network described in Chapter 6, in order to rationalise the number of A/D converters, sample-and-hold circuits and sensors required.

2.6.2 Combined Outputs

The outputs of the convolvers of several digital amplifiers may be added together before the D/A conversion stage when appropriate to the output variable. This technique is also needed to make the Digital Active Network described in Chapter 3, and again this rationalises the number of D/A converters and generators required.

2.6.3 Parametric Digital Amplifers

By including an actual digital multiplier in the signal path of the digital amplifer, then parametric effects may be created by feeding the second input of this amplifer with a time varying signal. An example of the use of this is discussed in Section 2.10.6 and in the analysis of a digital gyrator in Chapter 4.

2.6.4 Dual Output Digital Amplifier

In any digital amplifier it is perfectly feasible to have a second D/A converter in order to be able to view (on, say, an oscilloscope) the time domain output signal of the amplifier as in Fig. 2.4. The input signal is of course always obtainable. In the case of a Digital Active Network this technique allows the simultaneous viewing of the current and voltage at any given node.

As an alternative to the above arrangement, the same D/A converter may be used for both outputs and a second output generator introduced instead.

2.6.5 Generalised Digital Amplifier

By extending the technique described in Section 2.6.4, a generalised

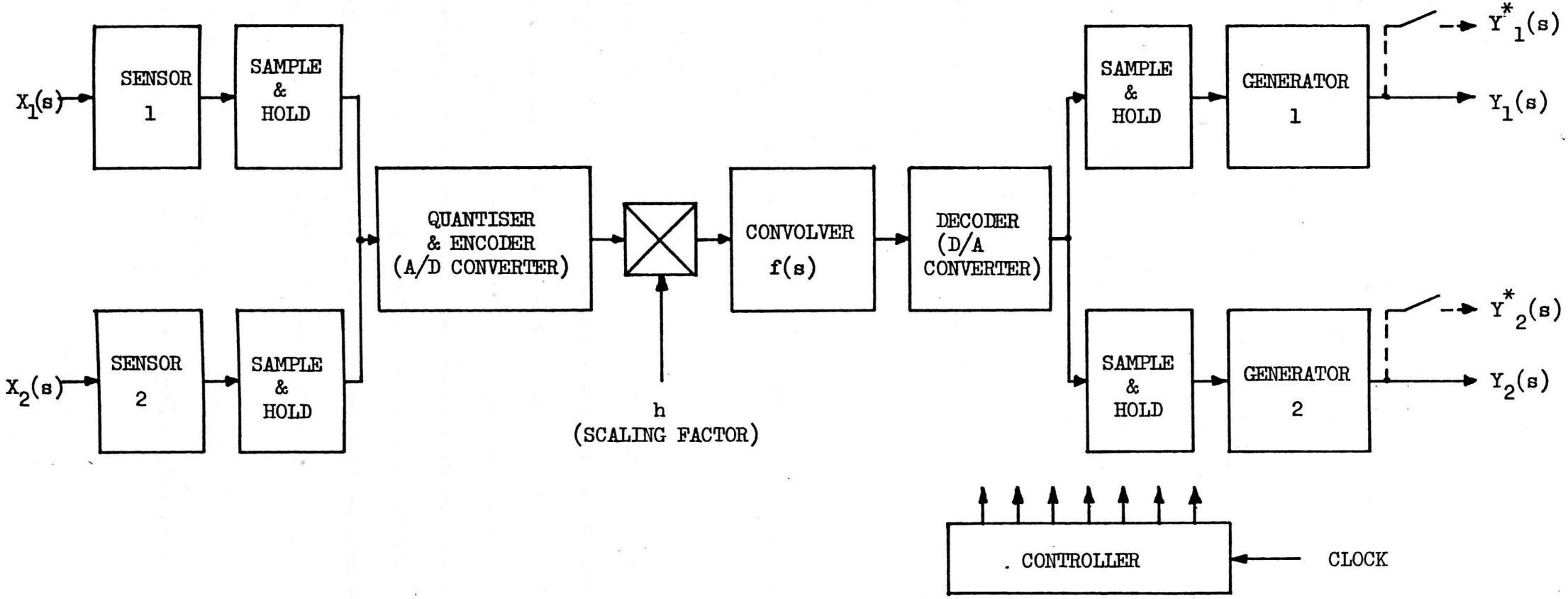


FIG. 2.4 Generalised Digital Amplifier

digital amplifier may be designed, as in Fig. 2.4. Two types of sensor and generator are included in the signal path, and by adjustments to the controller any of the four types of digital amplifier may be made, and the amplifier may even be multiplexed by judicious control of the four sample-and-hold stages.

2.6.6 Computer Controlled Digital Amplifier

Those sections of the signal path which are implemented in digital logic, namely the encoder, scaling factor, convolver and decoder, and also the sample-and-hold circuits are all easily controlled through a suitably interfaced digital computer. By this technique all the parameters of the digital amplifier may be made adjustable through software programming, and this becomes very important when a full digital active network is constructed. With suitable analysis of the output of a digital active network, the parameters of this network may be optimised.

It is possible for a microprocessor to perform this controlling action and to be integral with the digital active network. See Section 6.8.7.

2.7 ALIASING

Aliasing is due to the sampling process, and involves the folding of any part of the incoming frequency spectrum above half the sample rate into the amplifier passband which lies beneath half the sampling rate. The incoming signal can be filtered to remove the portion of its spectrum which lies above half the sampling rate. However if this technique is applied to the amplifiers in an array then the convolving properties of the amplifiers in the array will be considerably modified. The only solution is to filter the network input signals, be they currents or voltages.

2.8 IMPROVED DIGITAL AMPLIFIER ARCHITECTURE

The digital amplifier may be significantly improved by using the

same D/A converter in the encoder and decoder, as shown in Fig. 2.5. The disadvantage of needing a more complicated controller and an extra sample-and-hold circuit are easily outweighed by this improvement. Because the D/A converter cannot be used for its dual purpose simultaneously, a time delay is inherent in this arrangement, but some time delay is always inherent in a digital amplifier.

There are three principal advantages:

2.8.1 Simplification

The D/A converter not only governs the whole linearity of the amplifier, but is also likely to be the most costly single component.

Both these properties are very dependent on the number of bits to be decoded, and rapidly increase with an increase in the number of bits. Using a single D/A converter considerably simplifies the problem.

2.8.2 Modular Implementation

Fig. 2.6 shows the four modules making up this digital amplifier. The contents of each module are shown in Fig. 2.5. By changing input or output modules the type of amplifier can be changed, and by changing the convolver and scaler module, the whole transfer function can be changed. When assembling the digital admittance matrix realisation of a digital active network, as described in Chapter 6, this approach allows easy modifications to the convolver and scaler module only.

2.8.3 Floating point Arithmetic

It would be very advantageous to implement all the digital processes in the amplifier with floating point hardware, as this would greatly increase the amplifier's dynamic range, but not the resolution. However the complexity and cost would also increase greatly.

2.9 AUTO-CALIBRATION

By connecting the output signal Y(z) through a suitable scaling

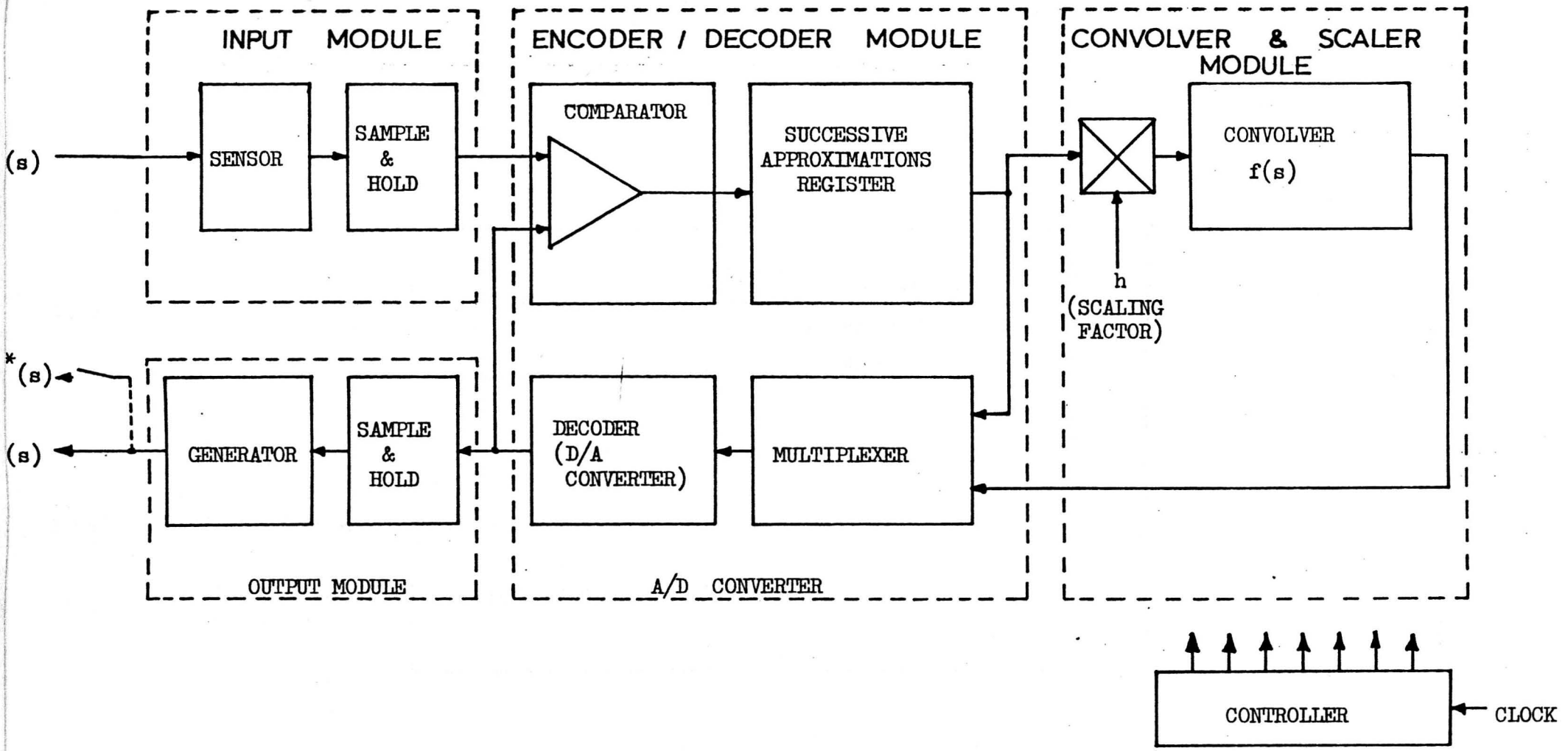


FIG. 2.5 Modified Digital Amplifier

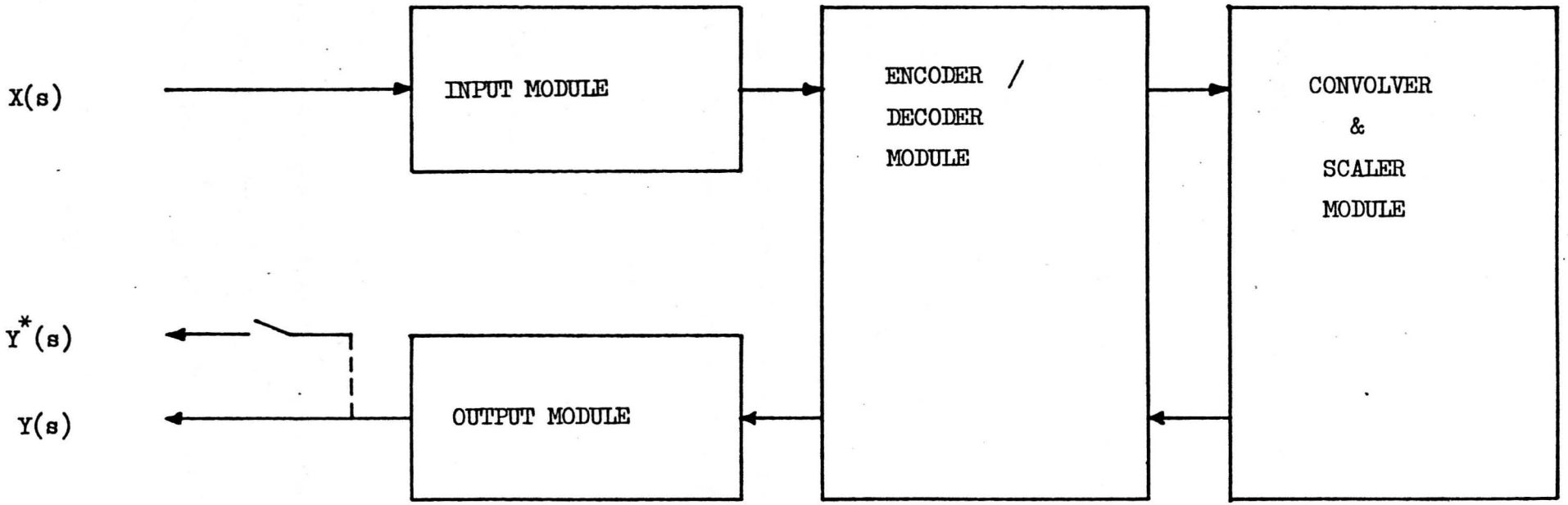


FIG. 2.6 Modular Digital Amplifier

device (H) back to the input (X(z)), then auto-calibration is possible. This may of course be done with any digital amplifier, but with only one D/A converter, the setting up is considerably easier. Fig. 2.7 shows the block diagram of such an arrangement, and E(z) is included as a notional input to aid analysis.

Considering Fig. 2.7 the pulse transfer function (PTF) may be:

$$\frac{Y(z)}{E(z)} = \frac{h}{z^k - Hh} \tag{2.54}$$

and the characteristic equation is:

$$z^k - Hh = 0 \tag{2.55}$$

If H and h are both assumed to be real and non-zero but with only h having either sign, it is clear from equation (2.55) that the PTF has k poles on the real axis when:

$$z_p = Hh \tag{2.56}$$

However, for stability, this multiple pole should lie within the unit circle. Hence:

$$Hh < 1 \tag{2.57}$$

The DC stability is of particular interest (as E(z) is only a notional input) and may be found by substituting z = 1 into equation (2.54).

$$\frac{X(1)}{E(1)} = \frac{h}{1-Hh}$$

and $\frac{h}{1-Hh} < 1$ for stability

which implies that :

$$h < \frac{1}{1+H} \tag{2.58}$$

This will always be so if the signs of H and h are different. This second inequality is slightly more stringent than the first.

The initial value theorem [10] may also be applied to equation (2.54) thus :

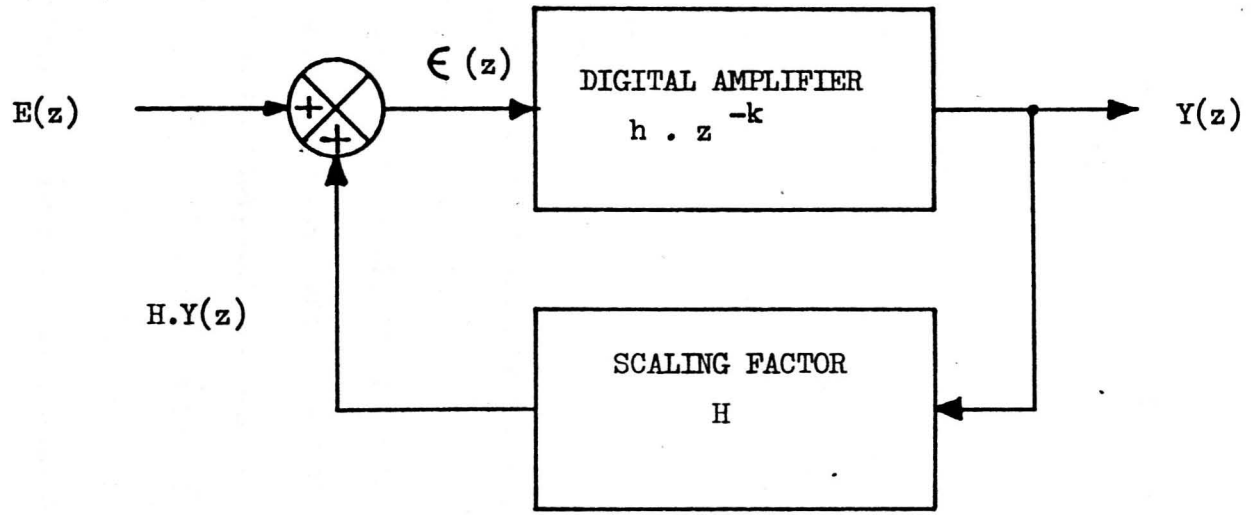


FIG. 2.7 Digital Amplifier arranged for Auto-calibration

$$\lim_{z \to \infty} \left(\frac{h}{z^k - Hh} \right) = 0 \qquad (2.59)$$

Hence the initial value of the PTF is 0.

The final value theorem may now be applied to equation (2.54) thus:

$$\lim_{z \to 1} \left(\left(\frac{z-1}{z} \right) \left(\frac{h}{z^k - Hh} \right) \right) = 0 \quad \text{when} \quad Hh \neq 1$$

$$= \frac{h}{k} \quad \text{when} \quad Hh = 1 \qquad (2.60)$$

As E(z) is a notional input, practical calibration of this amplifier implies checking that X(z) is zero under the conditions derived. The inequality (2.58) allows h to be measured by increasing H until latch-up occurs at $h = \frac{1}{H}$, assuming H and h to have the same sign.

2.10 DIGITAL COMPONENT SIMULATION

Techniques have been developed for simulating digital inductors and capacitors [1], [2] by Pim and Bullingham using binary rate multipliers. These techniques have shown severe bandwidth limitations and error problems, coupled with very difficult analysis.

An alternative technique applied here is to use a digital amplifier with input and output strapped and the appropriate transfer function defined by an internal digital filter. Such an amplifier is shown in Fig. 2.8 and differs from the basic digital amplifier as Fig. 2.1 in that the delay stage has been replaced by a simple digital filter. As an example the structure of this digital filter may be chosen to represent the digital equivalent of an inductor or capacitor.

The transfer function of the digital amplifier of Fig. 2.8 may be written down using equation (2.48):

$$f(z) = g\, Z \left\{ (1 - \exp(-sT_s)) \;/\; s \right\} h(z) \qquad (2.61)$$

where h(z) is the internal digital filter transfer function. After taking

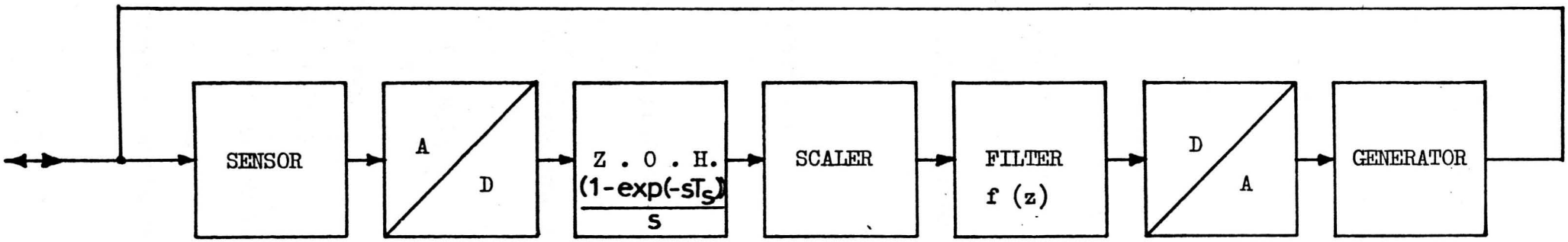


FIG. 2.8 Simulation of Grounded Component

the Z-transform equation (2.61) becomes:

$$f(z) = g\,h(z). \tag{2.62}$$

The form of equation (2.62) is valid because the digital filter will be clocked at the sample rate implying that it cannot change state between clock pulses. Thus a hypothetical sampler has been added between the zero-order hold and the digital filter.

2.10.1 Digital Capacitor

The admittance of an ordinary capacitor is:

$$y_c(s) = sC \tag{2.63}$$

This admittance may not be simply Z-transformed due to the rules described in section 2.3. However, by the duality of the Thevenin and Norton equivalent circuits, either the admittance or the impedance of this capacitor may be simulated. Now considering the impedance of this capacitor:

$$z_c(s) = 1/sC \tag{2.64}$$

which may be readily transformed to:

$$z_c(z) = \frac{T_s}{C}\left(\frac{z}{z-1}\right) \tag{2.65}$$

A digital transimpedance amplifier may be readily used to simulate $z_c(z)$.

Again by the duality of the Thevenin and Norton equivalent circuits the admittance may be simulated thus:

$$y_c(z) = \frac{I(z)}{V(z)} = \frac{C}{T_s}\left(\frac{z-1}{z}\right) \tag{2.66}$$

Thus it is expedient to say:

$$Z\left\{y_c(s)\right\} = 1/Z\left\{z_c(s)\right\} \tag{2.67}$$

This technique is developed further in succeeding chapters, particularly for the case of digital amplifier arrays.

Fig. 2.9 shows a realisation of the required digital filter using

a feed-forward first order digital filter where:

$$h(z) = \left(\frac{z-1}{z}\right) \tag{2.68}$$

and:

$$g = \frac{C}{T_s} \tag{2.69}$$

The frequency response of this digital capacitor only approximately matches that of a real capacitor. The DC response for $y_c(z)$ will be:

$$y_c(1) = 0 \tag{2.70}$$

which is correct. The response at the Nyquist rate will be:

$$y_c(-1) = \frac{C}{T_s} 2 \tag{2.71}$$

whereas that of an ordinary capacitor would be:

$$y_c(f_s) = 2\pi C/T_s \tag{2.72}$$

Thus the admittance of the digital capacitor has fallen short of an ordinary capacitor by a factor π at the sample rate.

2.10.2 Digital Inductor

The admittance of an analogue inductor is:

$$y_L(s) = \frac{1}{sL} \tag{2.73}$$

and this may be simply Z-transformed to:

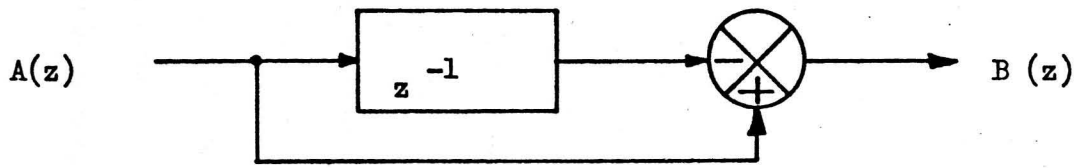
$$y_L(z) = \frac{T_s}{L} \left(\frac{z}{z-1}\right) \tag{2.74}$$

Thus the digital inductor may be simply realised as in Fig. 2.10 using a first order recursive digital filter thus:

$$h(z) = \frac{z}{z-1} \tag{2.75}$$

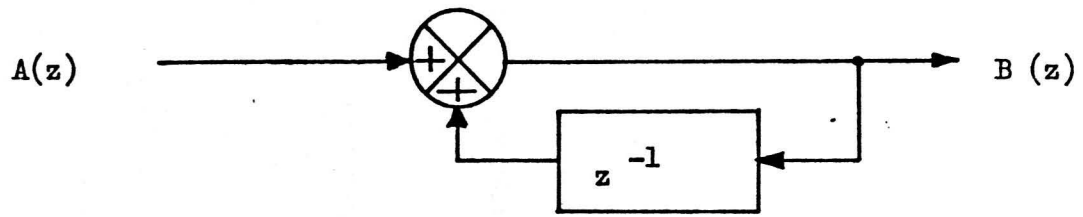
2.10.3 Further Digital Networks

The combinations of analogue components that may be simulated are clearly boundless and it is not the purpose of this thesis to describe interminable combinations but rather to explore the possibilities of



$f(z) = B(z) / A(z) = (z-1) / z$

FIG. 2.9



$f(z) = B(z)/ A(z) = z/ (z -1)$

FIG. 2.10

interconnecting networks of these digital amplifiers. However, it is interesting to consider the case of the parallel tuned circuit shunted by a conductance g_3 as in Fig. 2.11.

From equations (2.66) and (2.74) the overall impedance of this damped parallel tuned circuit will be:

$$z_p(z) = \frac{z\,(z-1)}{z^2(g_1+g_2) - 2g_1 z + g_1+g_3} \qquad (2.76)$$

where:

$$g_1 = C / T_s$$

$$g_2 = T_s / L$$

The complex conjugate poles of $z_p(z)$ will be at:

$$z = \frac{g_1}{g_1+g_2} \pm j \left(\sqrt{\frac{g_1g_2 + g_1g_3 + g_2g_3}{g_1 + g_2}} \right) \qquad (2.77)$$

The poles will lie on the z-plane unit circle when $| z | = 1$.

It can be shown that this condition will be true when:

$$(g_1+g_2)\,(g_2-g_3) = 0 \qquad (2.78)$$

The first term comes from having to square the denominator of equation (2.77). Thus the useful result is that the complex conjugate poles will lie on the unit circle when $g_2 = g_3$.

The argand of equation (2.77) may be used to calculate the resonant frequency from equation (2.78) thus:

$$\angle(z) = \mathrm{Tan}^{-1} \left(\sqrt{\frac{g_1g_2 + g_2g_3 + g_1g_3}{g_1}} \right) = w\,T \qquad (2.79)$$

Thus the natural oscillating frequency will be:

$$f_o = \frac{1}{2\pi T_s}\ \mathrm{Tan}^{-1} \left(\frac{\sqrt{2g_1g_2 + g_2^{\,2}}}{g_1} \right) \qquad (2.80)$$

Hence this novel circuit has achieved oscillation by making the

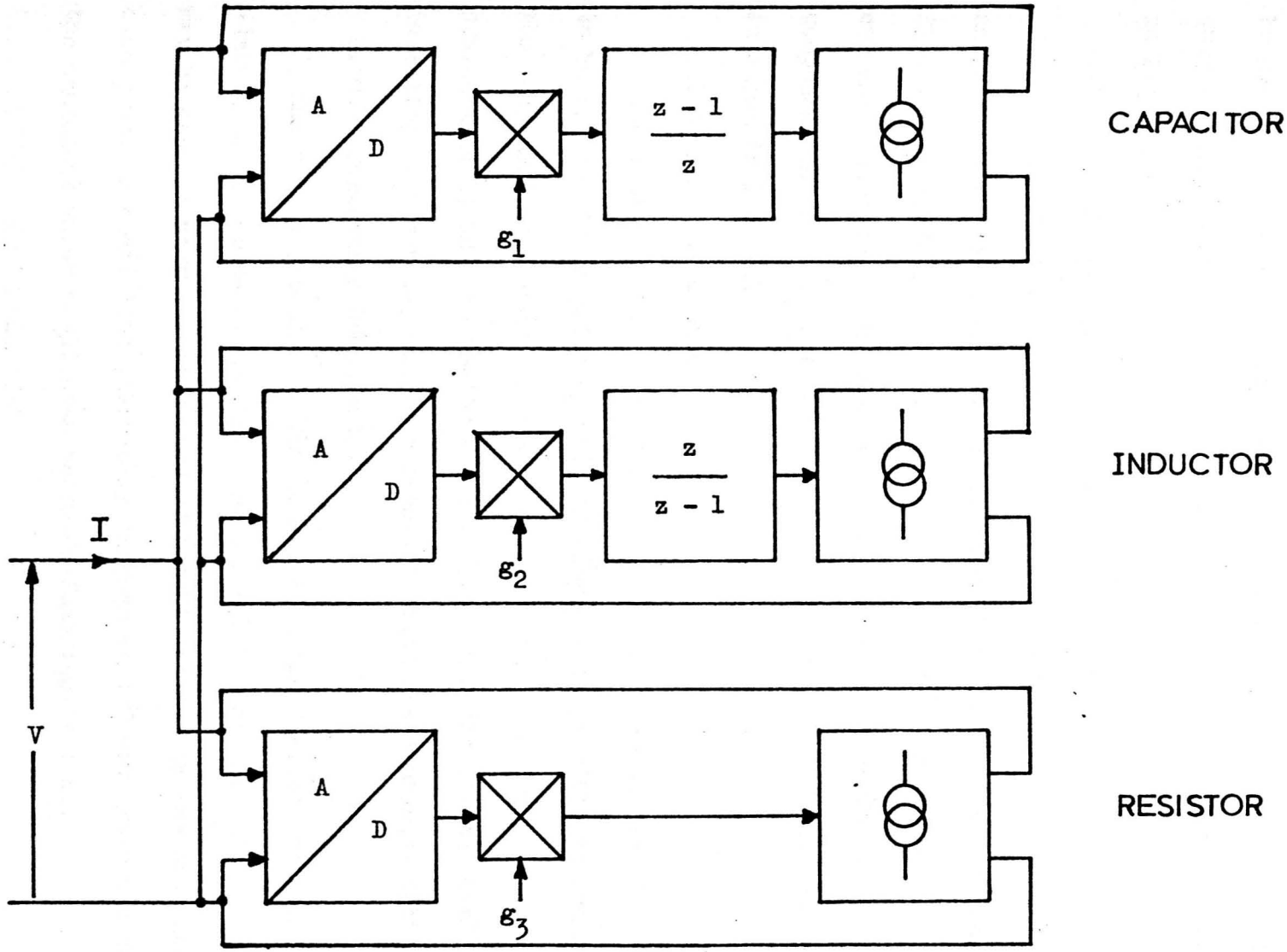


FIG. 2.11 Damped Parallel Tuned Circuit

dissipative element g_3 deliberately non-zero, and this is a notable contrast to the conventional parallel tuned circuit.

The block diagram shown in Fig. 2.11 may be considerably simplified by merging the input voltage sensors, A/D converters, D/A converters and current generators in the three parallel signal paths, and this is shown in Fig. 2.12.

2.10.4 Floating Digital Components

If the digital component or network of components is simulated using a digital transadmittance amplifier then it is simple to design a floating or differential voltage sensor and a floating or complementary current generator. Thus with a suitable internal transfer function the component or network of floating components may be simulated and this is shown in Fig. 2.13.

2.10.5 Negative Components

The scaler within a digital amplifier has been assumed to be able to be set to either polarity and therefore it is possible to simulate the component or network of components with the sign opposite to that conventionally used in analogue circuitry. Thus the digital amplifier is acting in a similar way to a negative impedance converter (NIC).

2.10.6 Parametric Components

Due to the presence of the scaler g in the digital component admittance or impedance and that intrinsically g is controllable, it may be used to vary the simulated component value. In the practical case g may be controlled externally through an A/D converter and thus the component value could even become a function of time.

2.11 SUMMARY AND CONCLUSIONS

The concept of a digital amplifier has been introduced, explained and analysed mathematically. In particular the possibilities of finding the Z-transform of a transfer function have been thoroughly explored.

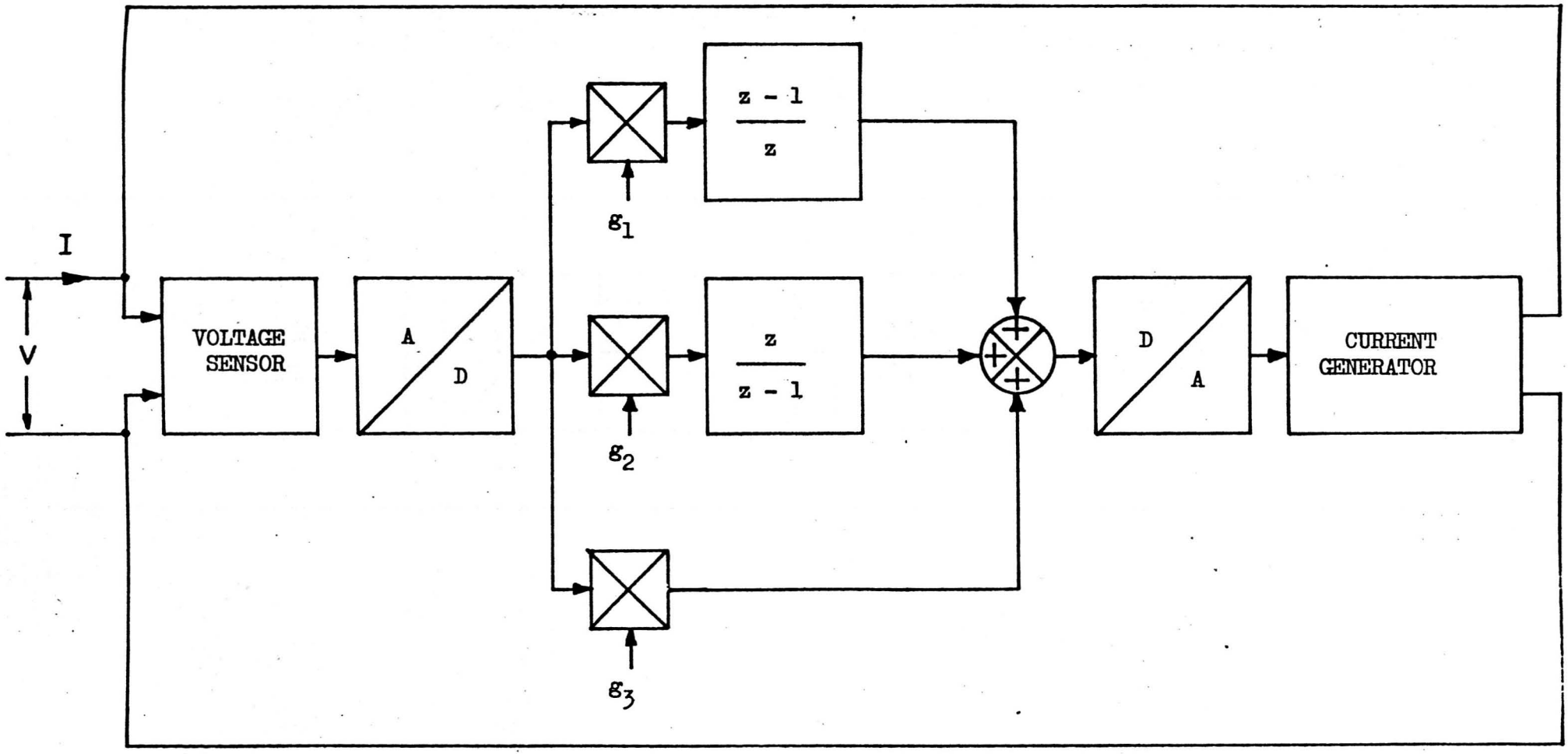


FIG. 2.12 Improved Damped Parallel Tuned Circuit

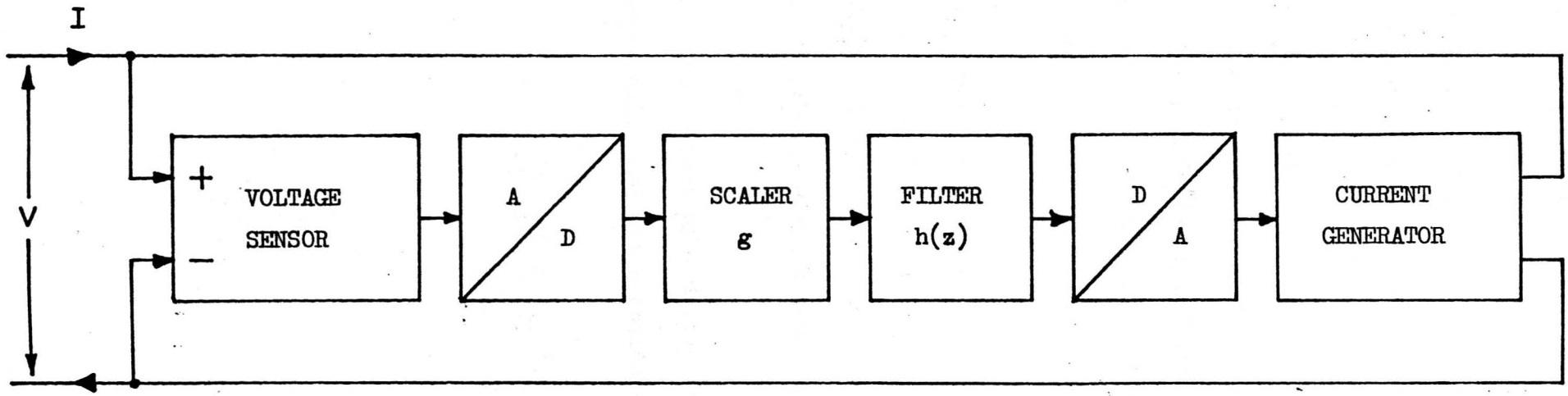


FIG. 2.13 Simulation of Floating Component

Several possible improvements to the design of a digital amplifier have been stated, including the possibility of auto-calibration of the digital amplifier.

The Z-transform of a rational transfer function has been shown to be finite when the numerator order is less than or equal to the denominator order, and this result is of importance in the next Chapter when considering the Z-transform of ordinary analogue circuit components.

The concept of digital circuit components has then been introduced and explained. The transfer functions of the digital filters necessary to simulate these components have been derived. As an example, a digital parallel tuned circuit has been analysed and the stability conditions evaluated.

Digital circuit components with negative and parametric values have also been explained.

The correction factor to conserve units of time when finding the Z-transform of analogue circuit components has been shown to be T_s, the sampling period.

Overall the concept of a digital amplifier has been fully considered theoretically and its associated problems analysed.

CHAPTER 3

DIGITAL AMPLIFIER ARRAYS

3.1 INTRODUCTION

The purpose of the array of digital amplifiers described here is to simulate the digital equivalent of a generalised active network, that is a digital active network. It is necessary firstly to consider the limitations that the mathematics which describe these arrays place upon the types of amplifier and the types of matrix which can be easily simulated. It is assumed initially that no passive or active analogue networks are connected to the nodes of the array. Secondly, in order to analyse mathematically all the possible variations with digital active networks, these networks will be considered in order of complexity.

All sampling in the digital networks that follow is assumed to be uniform and synchronous, though these parameters could be varied in a more general case.

3.2 GENERALISED ACTIVE NETWORKS

Any active network of N identifiable nodes may be fully described in N linear independent equations in the Laplace domain. At each node it is necessary to specify a voltage relative to a common external reference (often ground) and an input current. Hence the N linear independent equations describe the relationship between the N voltages and N currents. The N equations may be structured arbitrarily in the voltages and currents, but a regular structure is of great assistance in the mathematical analysis and of necessity in the practical realisation with (N + 1) voltages and/or currents in each equation.

The j^{th} equation may therefore be written:

$$P_i = \sum_{j=1}^{N} a_{ij} Q_j \qquad (3.1)$$

where P_i and the Q's are voltages or currents (and have been assumed so far to be mixed) and a_{ij} is a complex rational polynomial in s, the Laplace variable. Hence the equation (3.1) may be expanded to a matrix equation thus:

$$\begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_N \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_N \end{bmatrix} \qquad (3.2)$$

$$\text{or } P = A.Q \qquad (3.3)$$

As a second restriction on the formation of this matrix equation, it is convenient to assume that all the elements of the column vector in P are of one type, and all elements in the volumn vector Q are of the other type. If so, then P represents voltages or currents, and Q the opposite.

These two constraints produce the two well known matrices, admittance and impedance. The practical realisation of these two matrices is now easily derived.

For an arbitary element a_{ij} of a square matrix, the following equation applies:

$$P_{ij} = a_{ij} Q_j$$

which is easily simulated by an amplifier with a transfer function a_{ij}. The application of the second restriction implies that all the amplifiers must be either trans-impedance or trans-admittance.

3.3 DIGITAL IMPEDANCE AMPLIFIER ARRAY

The matrix equations which must be simulated by a digital impedance amplifier array are:

$$\begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_N \end{bmatrix} = \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1N} \\ z_{21} & z_{22} & \cdots & z_{2N} \\ \vdots & \vdots & & \vdots \\ z_{N1} & z_{N2} & \cdots & z_{NN} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_N \end{bmatrix} \qquad (3.4)$$

where each z off the leading diagonal will be simulated by a digital transimpedance amplifier. As discussed in Chapter 2, these amplifiers are very difficult to construct due to the problem of realising a suitable sensor and generator. If the leading diagonal impedances are also to be simulated by amplifiers as described in Chapter 6, the the problem is even more complicated. Hence the impedance matrix is not a convenient choice for practical simulation. However, the equivalent impedance matrix may always be derived from the admittance matrix.

3.4 DIGITAL ADMITTANCE AMPLIFIER ARRAY

The matrix equations which must be simulated by a digital admittance amplifier array are:

$$\begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_N \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1N} \\ y_{21} & y_{22} & \cdots & y_{2N} \\ \vdots & \vdots & & \vdots \\ y_{N1} & y_{N2} & \cdots & y_{NN} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_N \end{bmatrix} \qquad (3.5)$$

where each y off the leading diagonal will be simulated by a digital transadmittance amplifier. As discussed in Section 2.10.4 these amplifiers are very much easier to construct both in the grounded and floating forms and the leading diagonal admittances are also easy to simulate.

It was therefore decided that the admittance matrix would be simulated, and the impedance matrix derived theoretically where necessary by conventional matrix inversion.

Due to the choice of an amplifier array as described, the regular

construction of the array may be exploited. Firstly, each column of the admittance matrix has the same associated voltage variable, and each row is summed to produce a current. The array is shown in Fig. 3.1. It can be seen that each array element is simulated by an array module containing a scaler (g_{ij}), a convolver ($f_{ij}(z)$) and an adder. This structure therefore lends itself to a modular construction, assuming that each scaler and convolver are basically the same. The disadvantage is that for a general purpose machine N^2 array modules are needed.

If the N^2 array modules are considered separately, then they constitute a multi-input, multi-output digital filter. [3] [4] However once the overall admittance matrix is simulated, interaction with passive and active analogue components is easy. Furthermore standard analogue active networks such as gyrators and circulators may also be simulated.

By employing the technique described in section 2.8, an improved amplifier array may be made, and this is shown in Fig. 2.5. Both the voltage encoding and current decoding are done in the same encoder/decoder module. N such modules are required, thereby completing the modular approach to the simulation of the admittance matrix.

If the required bandwidth of this digital active network is low, then various components in the digital amplifier array may be multiplexed, in particular the A/D and D/A converters, the scalers and the convolvers. There are many possible combinations, but these will not be discussed because it is intended that the matrix will be studied in general.

3.5 DIGITAL TRANSADMITTANCE AMPLIFIERS

The transfer function of the digital transadmittance amplifiers used in this analysis are assumed to be of the form:

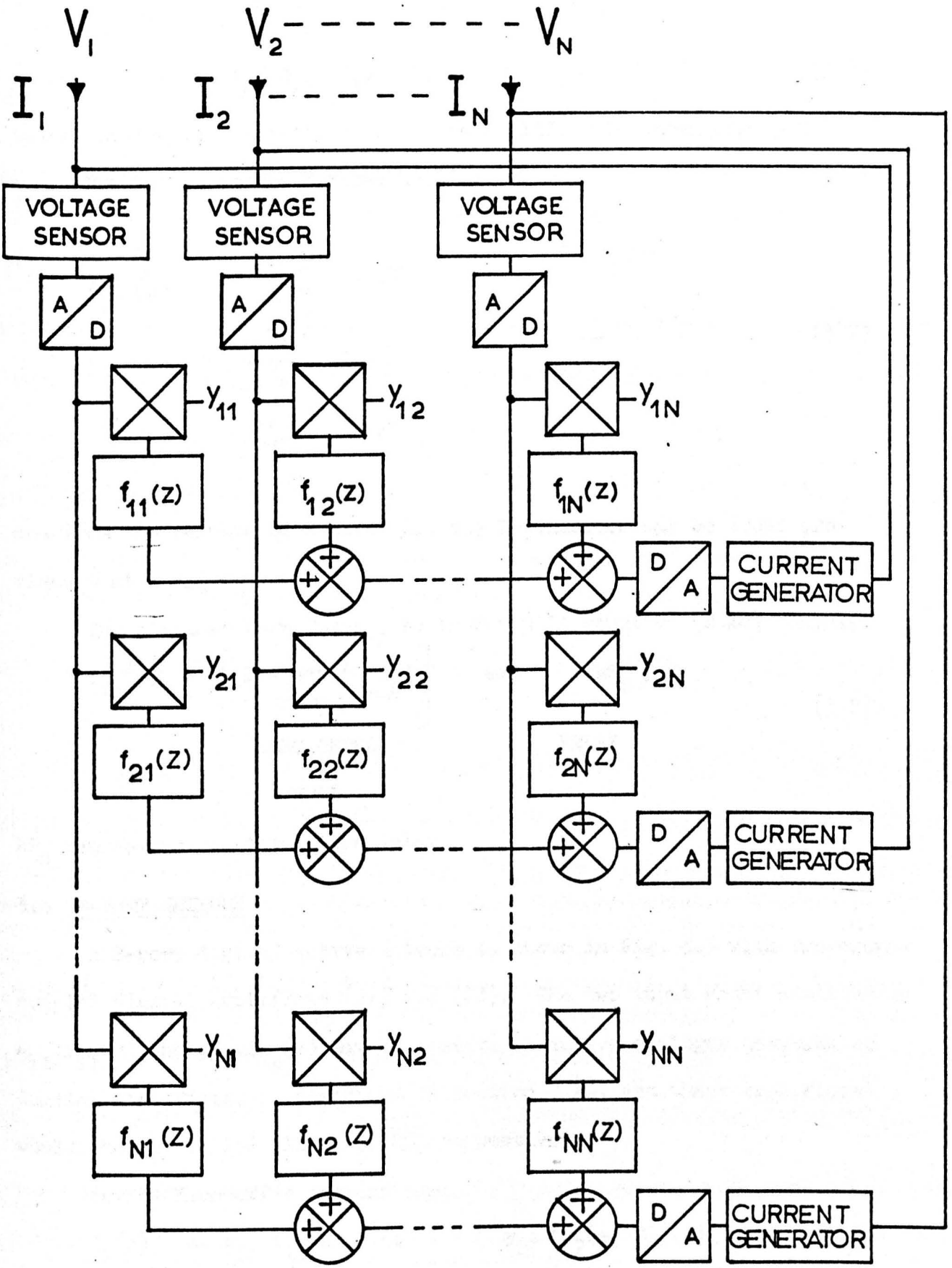


FIG. 3.1 Generalised Digital Admittance Amplifier Array

$$\frac{I_i^*(s)}{V_j^*(s)} = \left[g_{ij}\, f_{ij}(s) \right]^* = T_s\, g_{ij}\, f_{ij}^*(s) \qquad (3.6)$$

using the results of Section 2.3. The amplifier is shown Fig. 3.2.

Now let $f_{ij}(s)$ be defined as:

$$f_{ij}(s) = \frac{\sum_{k=0}^{n} a_k s^k}{\sum_{k=0}^{m} b_k s^k} \qquad (3.7)$$

and from the results of Section 2.3 the Z-transform may be found provided that $n \leqslant m$.

The simplest form for $f_{ij}(s)$ is shown in equation (2.48), namely:

$$f_{ij}(s) = \underbrace{\left[\frac{1 - \exp(-sT_s)}{s} \right]}_{\text{ZERO-ORDER HOLD}} \underbrace{\exp(-skT_s)}_{\text{DELAY}} \qquad (3.8)$$

kT_s represents the total time delay.

3.6 2-PORT NETWORK

A 2-port digital active network is shown in Fig. 3.3 with two cross-coupled digital amplifiers (21) and (12). The two input shunt admittances $g_{11}.f_{11}(s)$ and $g_{22}.f_{22}(s)$ may be passive or active analogue networks or digital amplifiers, as described in Section 2.10, and these amplifiers would then be called (11) and (22) respectively.

Now by Kirchoff's current law:

$$I_1(s) = g_{11}.f_{11}(s)\, V_1(s) + g_{12}\, f_{12}(s)\, V_2^*(s) \qquad (3.9)$$

$$I_2(s) = g_{21}\, f_{21}(s)\, V_1^*(s) + g_{22}\, f_{22}(s)\, V_2(s) \qquad (3.10)$$

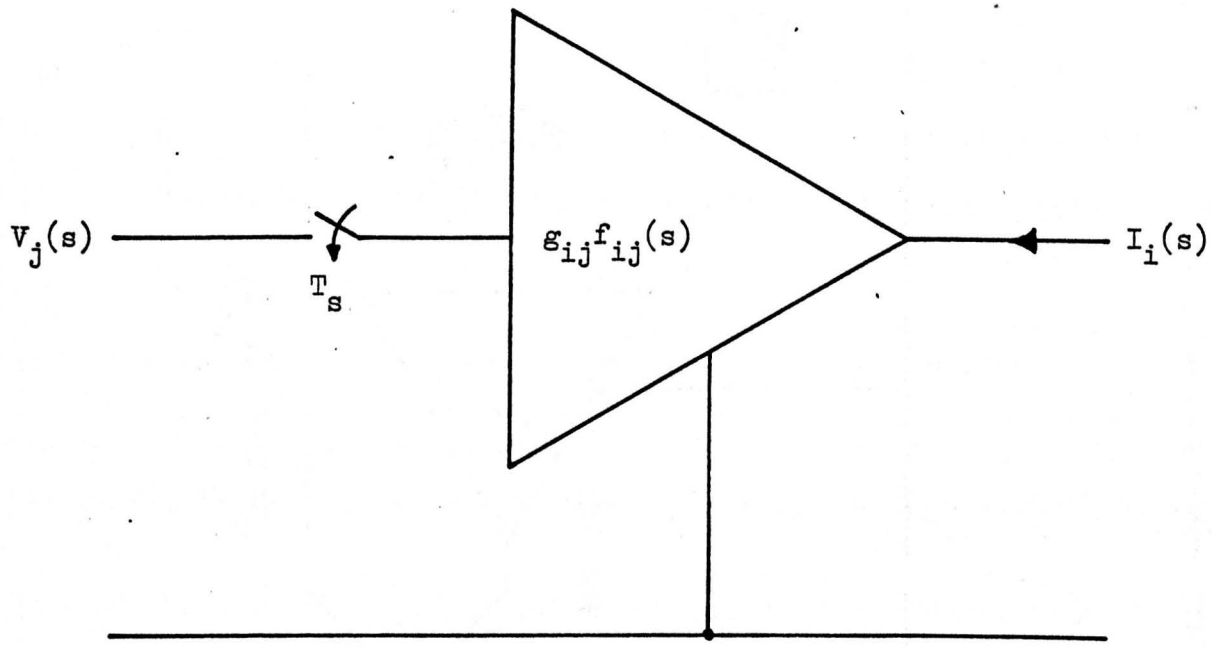


FIG. 3.2 General Transadmittance Amplifier

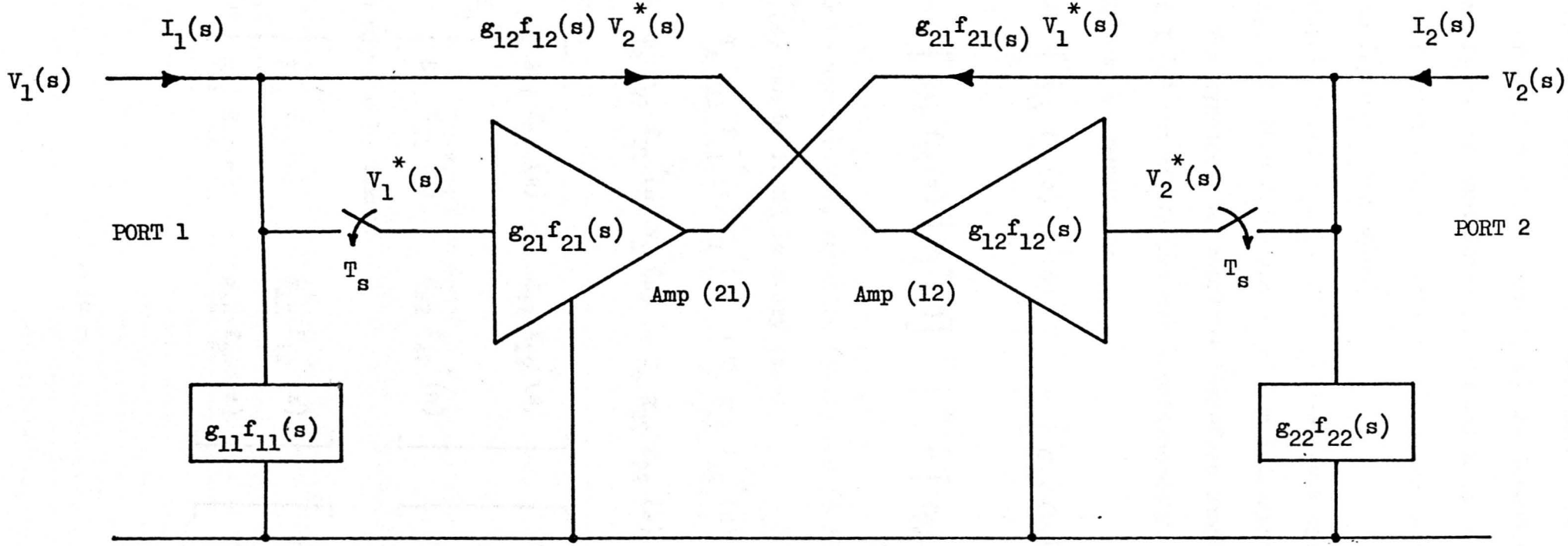


FIG. 3.3. Digital 2-port Network

where s is the complex variable of the Laplace Transform, and * implies sampling. In the above equations it is important to note that $V_j(s)$ is the independent or input variable, and $I_i(s)$ the dependent or output variable because it is the admittance matrix which is being simulated. However there must exist a matrix where $I_i(s)$ is the independent variable and $V_j(s)$ is the dependent variable.

If the output variables $I_1(s)$ and $I_2(s)$ are now hypothetically sampled (see Fig. 3.4) then $V_1(s)$ and $V_2(s)$ are intrinsically hypothetically sampled. The hypothetical samplers therefore sample $V_i(s)$ into the network and $I_i(s)$ out of the network simultaneously. Hence equations (3.9) and (3.10) may be rewritten:

$$I_1^*(s) = \left[g_{11}\, f_{11}(s)\, V_1^*(s) \right]^* + \left[g_{12}\, f_{12}(s) \left[V_2^* \right]^* \right]^* \tag{3.11}$$

$$I_2^*(s) = \left[g_{21}\, f_{21}(s) \left[V_1^*(s) \right]^* \right]^* + \left[g_{22}\, f_{22}(s)\, V_2^*(s) \right]^* \tag{3.12}$$

According to Section 2.3, equation (3.11) and (3.12) may be rewritten if $f_{ij}^*(s)$ can be found in every case:

$$I_1^*(s) = T_s\, g_{11}\, f_{11}^*(s)\, V_1^*(s) + T_s\, g_{12}\, f_{12}^*(s)\, V_2^*(s) \tag{3.13}$$

$$I_2^*(s) = T_s\, g_{21}\, f_{21}^*(s)\, V_1^*(s) + T_s\, g_{22}\, f_{22}^*(s)\, V_2^*(s) \tag{3.14}$$

or in matrix form:

$$\begin{bmatrix} I_1^*(s) \\ I_2^*(s) \end{bmatrix} = T_s \begin{bmatrix} g_{11}\, f_{11}^*(s) & g_{12}\, f_{12}^*(s) \\ g_{21}\, f_{21}^*(s) & g_{22}\, f_{22}^*(s) \end{bmatrix} \begin{bmatrix} V_1^*(s) \\ V_2^*(s) \end{bmatrix} \tag{3.15}$$

The Z-transform may now be taken:

$$\begin{bmatrix} I_1(z) \\ I_2(z) \end{bmatrix} = T_s \begin{bmatrix} g_{11}\, f_{11}(z) & g_{12}\, f_{12}(z) \\ g_{21}\, f_{21}(z) & g_{22}\, f_{22}(z) \end{bmatrix} \begin{bmatrix} V_1(z) \\ V_2(z) \end{bmatrix} \tag{3.16}$$

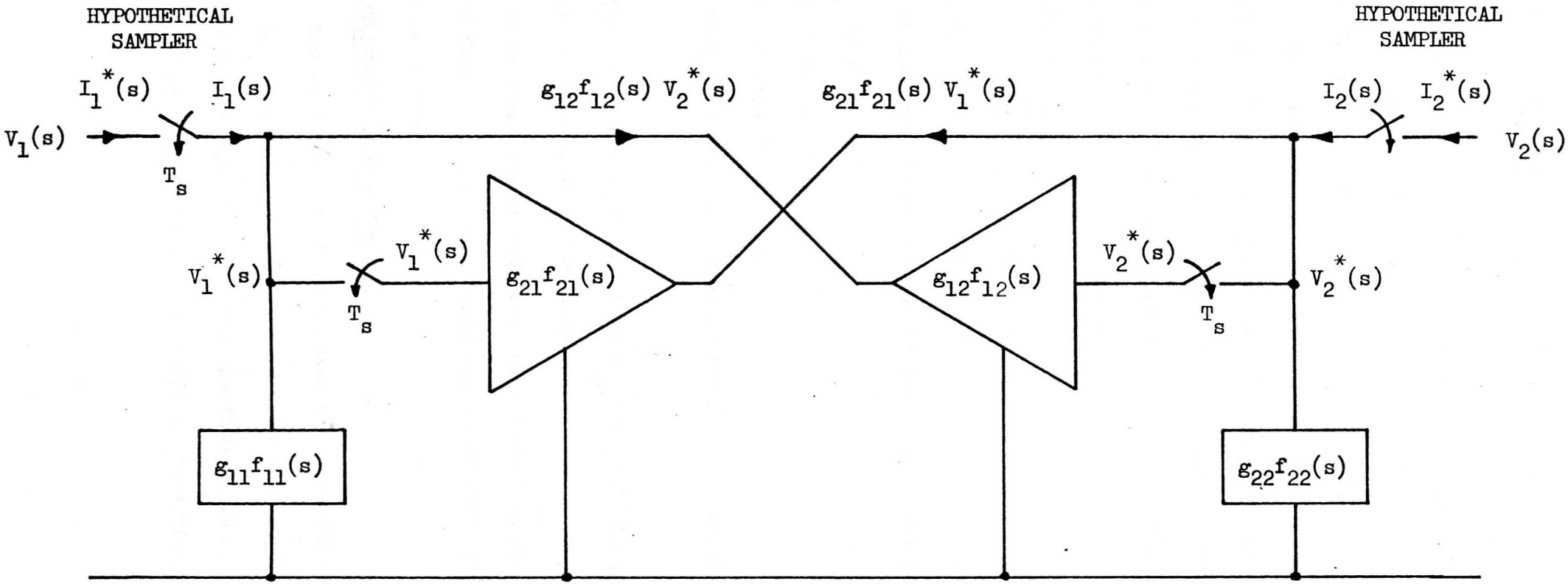


FIG. 3.4 Digital 2-Port Network with Hypothetical Samplers

Where:

$$y_{11}(z) = \left.\frac{I_1(z)}{V_1(z)}\right|_{V_2(z)=0} = T_s\, g_{11}\, f_{11}(z) \qquad (3.17)$$

and similarly for the other three elements.

Hence the sampled admittance matrix has been found and this has elements defined in a manner similar to the analogue admittance matrix.

This 2-port network has been fully described by the admittance matrix which it in turn is simulating. However the network is also described by a sampled impedance matrix despite being a simulation of the admittance matrix. The elements are defined in exactly the same way as the analogue impedance matrix, namely:

$$z_{11}(z) = \left.\frac{V_1(z)}{I_1(z)}\right|_{I_2(z)=0}$$

and similarly for the other three elements.

As the form of the definition is the same as the analogue case it follows in general that:

$$Y(z)\ Z(z) = U = Z(z)\ Y(z)$$

where U is the identity matrix.

The properties of these sampled matrices are considered in Chapter 4.

So far it has been assumed that $f_{ij}^{*}(s)$ could be found directly, but this is not necessarily always the case. In the 2-port case, either the first or second rows or both rows of the admittance may not be able to be found directly, and these cases are considered after the general case has been derived.

3.7 TRANSFORMABLE N-PORT NETWORK

The case of the 2-port network with all elements transformable including the analogue shunt admittances connected to the 2-ports has been studied in the previous section. Section 2.3.3 derives the ratios of polynomials which will be either wholly or conditionally transformable.

The results can now be used to establish which equations are wholly or conditionally transformable. An N-port network is shown in Fig. 3.5.

To analyse completely the N-port network it would be necessary to consider all combinations of transformable and untransformable matrix elements. However the analyses presented in the 6 cases studied cover examples of each combination, including digital amplifiers as shunt circuit components at the ports.

Case 1

Conditions: 1) All elements transformable.

2) Leading diagonal elements derived from digital shunt admittances created by digital transadmittance amplifiers with input and output linked.

The matrix equations are:

$$\begin{bmatrix} I_1(s) \\ I_2(s) \\ \vdots \\ I_N(s) \end{bmatrix} = \begin{bmatrix} y_{11}(s) & y_{12}(s) & \cdots\cdots & y_{1N}(s) \\ y_{21}(s) & y_{22}(s) & \cdots\cdots & y_{2N}(s) \\ \vdots & \vdots & & \vdots \\ y_{N1}(s) & y_{N2}(s) & \cdots\cdots & y_{NN}(s) \end{bmatrix} \begin{bmatrix} V_1^*(s) \\ V_2^*(s) \\ \vdots \\ V_N^*(s) \end{bmatrix}$$

These equations are derived from Fig. 3.5:

$$y_{ij}(s) = g_{ij} \cdot f_{ij}(s) \qquad (3.18)$$

Hence in matrix terms:

$$I(s) = Y(s)\ V^*(s) \qquad (3.19)$$

As all $f_{ij}(s)$ are transformable, simultaneous hypothetical sampling at each node gives:

$$I^*(s) = T_s\ Y^*(s)\ V^*(s) \qquad (3.20)$$

The Z-transform may now be taken:

$$I(z) = T_s\ Y(z)\ V(z) \qquad (3.21)$$

There must in general exist a matrix Z(z) where:

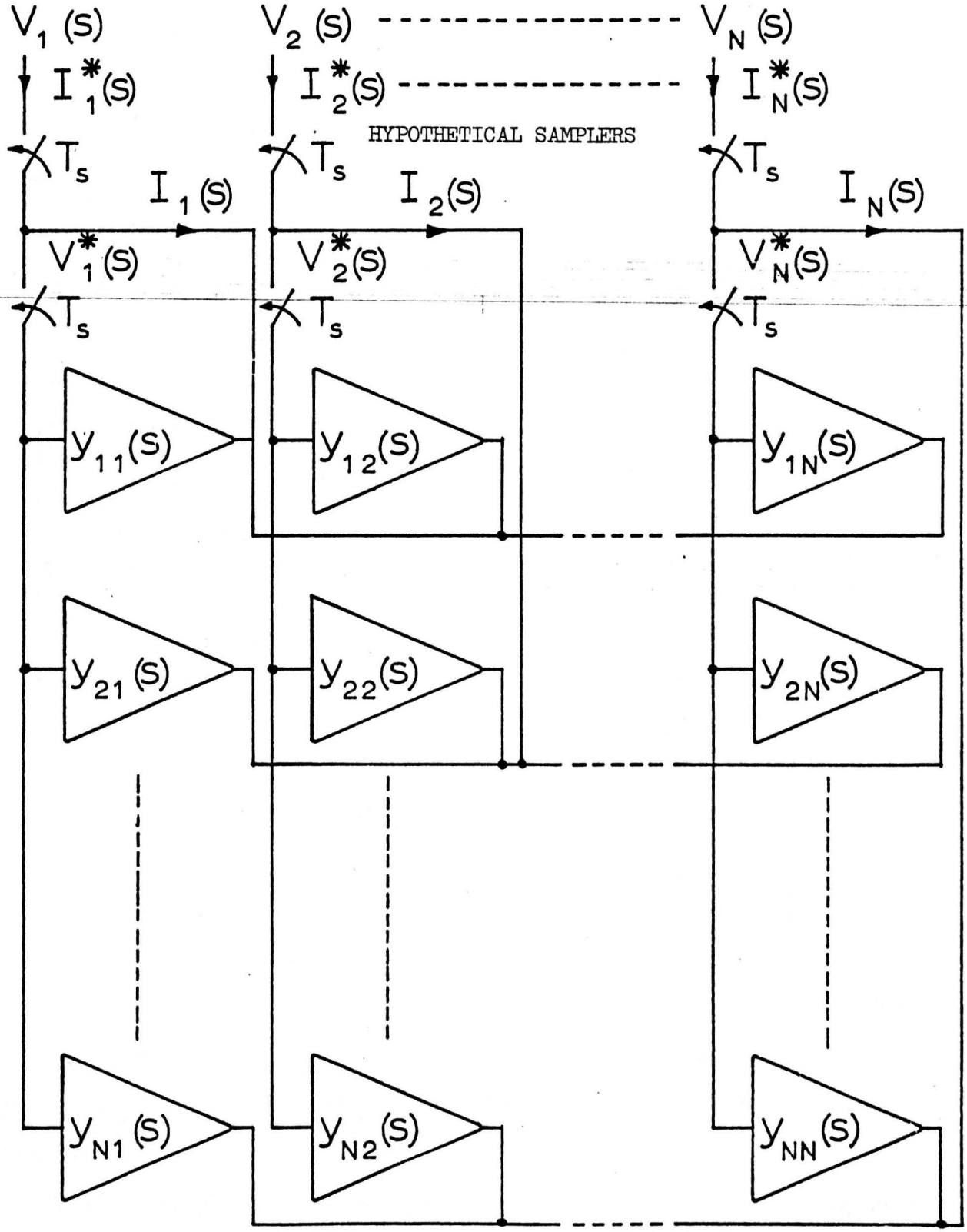


FIG. 3.5 Digital N-port Network

$$Z(z)\ Y(z) = U$$

(unless Y(z) is singular).

Z(z) is the equivalent digital impedance matrix.

Case 2

Conditions: 1) All elements transformable

2) Leading diagonal elements derived from analogue shunt admittances.

The matrix equations are:

$$\begin{bmatrix} I_1(s) \\ I_2(s) \\ \vdots \\ I_N(s) \end{bmatrix} = \begin{bmatrix} 0 & y_{12}(s) & \cdots & y_{1N}(s) \\ y_{21}(s) & 0 & \cdots & y_{2N}(s) \\ \vdots & \vdots & & \vdots \\ y_{N1}(s) & y_{N2}(s) & \cdots & 0 \end{bmatrix} \begin{bmatrix} V_1^*(s) \\ V_2^*(s) \\ \vdots \\ V_N^*(s) \end{bmatrix}$$

$$+ \begin{bmatrix} y_{11}(s) & 0 & \cdots & 0 \\ 0 & y_{22}(s) & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & y_{NN}(s) \end{bmatrix} \begin{bmatrix} V_1(s) \\ V_2(s) \\ \vdots \\ V_N(s) \end{bmatrix} \qquad (3.22)$$

The amplifier array is similar to Fig. 3.5 but with all the amplifiers on the leading diagonal removed. Let equation (3.22) be rewritten:

$$I(s) = Y_1(s)\ V^*(s) + Y_2(s)V(s) \qquad (3.23)$$

where $Y_1(s) + Y_2(s) = Y(s)$

Due to the simultaneous hypothetical sampling which is necessary to analyse the network, $I_j(s)$ and $V_j(s)$ are both sampled. Hence the matrix equation reduces to:

$$I^*(s) = T_s\ Y^*(s)\ V^*(s) \qquad (3.24)$$

and the Z-transform is:

$$I(z) = T_s\, Y(z)\; V(z) \tag{3.25}$$

which is identical with the previous result (3.21). This implies that it is immaterial whether analogue networks of sampled digital admittance amplifiers are in shunt with each port. Hence these two types may be arbitarily mixed in any practical array, always assuming that every element is transformable.

Case 3

Conditions: 1) All off diagonal elements are transformable.

2) All leading diagonal elements derived from analogue shunt admittances which are not transformable.

The matrix equations may be separated as in (3.22), and rewritten as in (3.23). By the above definition, $Y_1(s)$ is transformable, and $Y_2(s)$ not transformable. Now in this case, instead of directly finding $Y(z)$, find $Z(z)$ first and then invert to find $Y(z)$. This implies that the equations must be rearranged to make $V(s)$ the output column vector instead of $I(s)$.

Now $V(s)$ may be found by multiplying equation (3.23) by $Y_2^{-1}(s)$ which by definition is transformable. Thus:

$$V(s) = Y_2^{-1}(s)\, I(s) - Y_2^{-1}(s)\, Y_1(s)\, V^*(s) \tag{3.26}$$

where $Y_2^{-1}(s)$ is a diagonal matrix thus:

$$Y_2^{-1}(s) = \begin{bmatrix} \frac{1}{y_{11}(s)} & 0 & \cdots & 0 \\ 0 & \frac{1}{y_{22}(s)} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \frac{1}{y_{NN}(s)} \end{bmatrix} \tag{3.27}$$

The inverse of each element of the original matrix is bound to be

transformable.

Further, $Y_2^{-1}(s)\, Y_1(s)$ is also transformable due to its form:

$$Y_2^{-1}(s)\, Y_1(s) = \begin{bmatrix} 0 & \frac{y_{12}(s)}{y_{11}(s)} & \cdots\cdots\cdots & \frac{y_{1N}(s)}{y_{11}(s)} \\ \frac{y_{21}(s)}{y_{22}(s)} & 0 & \cdots\cdots\cdots & \frac{y_{2N}(s)}{y_{22}(s)} \\ \vdots & \vdots & & \vdots \\ \frac{y_{N1}(s)}{y_{NN}(s)} & \frac{y_{N2}(s)}{y_{NN}(s)} & \cdots\cdots\cdots & 0 \end{bmatrix} \qquad (3.28)$$

As $y_{ij}(s)$ is transformable and $y_{ii}(s)$ is not transformable by definition, then from Section 2.2.3, $y_{ij}(s) \,/\, y_{ii}(s)$ is always transformable.

Hence the matrix output column vector $V(s)$ in equation (3.26) is now able to be hypothetically sampled, which in turn also samples the input column vector $I(s)$:

$$V^*(s) = T_s\left[Y_2^{-1}(s)\right]^* I^*(s) - \left[Y_2^{-1}\, Y_1(s)\right]^* V^*(s) \qquad (3.29)$$

Only the first term incorporates the sampling period T_s, because the matrix product $Y_2^{-1}(s)\, Y_1(s)$ is unitless by definition, as every element is an admittance, and the product yields the ratio of these admittances as in equation (3.28).

Now equation (3.29) may be rearranged:

$$V^*(s)\left[U + \left[Y_2^{-1}(s)\;\; Y_1(s)\right]^*\right] = \left[Y_2^{-1}(s)\right]^* I^*(s)\, T_s$$

(where U is the identity matrix), and hence:

$$V^*(s) = T_s\left[U + \left[Y_2^{-1}(s)\, Y_1(s)\right]^*\right]^{-1}\left[Y_2^{-1}(s)\right]^* I^*(s)$$

Taking the Z-transform, the impedance matrix may be written:

$$Z(z) = T_s\left[U + Z\left\{Y_2^{-1}(s)\, Y_1(s)\right\}\right]^{-1} Z\left\{Y_2^{-1}(s)\right\} \qquad (3.30)$$

This matrix may now be inverted to give Y(z):

$$Y(z) = \frac{1}{T_s}\left[Z\left\{Y_2^{-1}(s)\right\}\right]^{-1}\left[U + Z\left\{Y_2^{-1}(s)\,Y_1(s)\right\}\right] \qquad (3.31)$$

Case 4

Conditions: 1) All off-diagonal elements transformable.

2) All leading diagonal elements derived from shunt admittances, some of which are transformable.

From matrix equation (3.23) and the above definition, $Y_1(s)$ is transformable and $Y_2(s)$ non-transformable. Now define a diagonal matrix A, such that:

$$A_1\,I(s) = A_1\,Y_1(s)\,V^*(s) + A_1\,Y_2(s)\,V(s) \qquad (3.32)$$

where $A_1\,Y_2(s)$ is transformable. A_1 is a 'row-selecting' matrix and also symmetrical. Define secondly another diagonal matrix A_2 such that:

$$A_2 = U - A_1 \qquad (3.33)$$

giving:

$$A_2\,I(s) = A_2\,Y_1(s)\,V^*(s) + A_2\,Y_2(s)\,V(s) \qquad (3.34)$$

where $A_2\,Y_2(s)$ is entirely non-transformable.

Matrices A_1 and A_2 have some rows which are all-zero by definition, and their inverses do not exist. However equation (3.32) can be transformed by definition now:

$$\left[A_1\,I(s)\right]^* = T_s\left[A_1\,Y_1(s)\right]^*V^*(s) + T_s\left[A_1\,Y_2(s)\right]^*V^*(s) \qquad (3.35)$$

Although the elements of A_1 are not a function of s, it is the matrix products now formed which are important. However, as I(s) and $Y_1(s)$ are transformable by definition, this equation may be rewritten:

$$A_1\,I^*(s) = T_s\,A_1\,Y_1^*(s)\,V^*(s) + T_s\left[A_1\,Y_2(s)\right]^*V^*(s) \qquad (3.36)$$

Now taking the Z-transform:

$$A_1\,I(z) = T_s A_1\,Y_1(z)\;V(z) + T_s\;Z\left\{A_1Y_2(s)\right\}V(z)$$

$$= T_s\left[A_1\,Y_1(z) + Z\left\{A_1\,Y_2(s)\right\}\right]V(z) \qquad (3.37)$$

Now by definition A_1, A_2 and $Y_2(s)$ are diagonal matrices, that is:

$$A_1 = A_1^T$$

$$A_2 = A_2^T$$

$$Y_2(s) = Y_2(s)^T$$

and hence :

$$A_2\, Y_2(s) = Y_2(s)\, A_2 \tag{3.38}$$

Hence equation (3.34) may be rearranged thus:

$$A_2 I(s) = A_2\, Y_1(s)\, V^*(s) + Y_2(s)\, A_2\, V(s) \tag{3.39}$$

Now multiplying through by $Y_2^{-1}(s)$ and rearranging gives:

$$A_2\, Y_2^{-1}(s)\, A_2 I(s) = A_2\, Y_2^{-1}(s)\, Y_1(s)\, V^*(s) + A_2 V(s) \tag{3.40}$$

V(s) and I(s) may now be hypothetically sampled:

$$\frac{1}{T_s}\left[A_2\, Y_2^{-1}(s)\right]^* A_2 I^*(s) = \left[A_2\, Y_2^{-1}(s) Y_1(s)\right]^* V^*(s) + A_2 V^*(s) \tag{3.41}$$

Taking the Z-transform and rearranging further:

$$A_2\, I(z) = T_s \left[Z(A_2\, Y_2^{-1}(s))\right]^{-1} \left[Z(A_2\, Y_2^{-1}(s)\, Y_1(s)) + A_2\right] V(z) \tag{3.42}$$

Now $A_2\, Y_2^{-1}(s)$ and $A_2\, Y_2^{-1}(s)\, Y_1(s)$ are both transformable by definition.

Equations (3.37) and (3.42) may now be added to make I(z):

$$I(z) = A_1\, I(z) + A_2\, I(z)$$

$$= T_s \left[A_1\, Y_1(z) + Z\left\{A_1\, Y_2(s)\right\} + \left[Z\left\{A_2\, Y_2^{-1}(s)\right\}\right]^{-1} \cdot \left[Z\left\{A_2\, Y_2^{-1}(s)\, Y_1(s)\right\} + A_2\right]\right] V(z) \tag{3.43}$$

and hence:

$$Y(z) = T_s \left[A_1\, Y_1(z) + Z\left\{A_1\, Y_2(s)\right\} + \left[Z\left\{A_2\, Y_2^{-1}(s)\right\}\right]^{-1} \cdot \left[Z\left\{A_2 Y_2^{-1}(s)\, Y_1(s)\right\} + A_2\right]\right] \tag{3.44}$$

Case 5

Conditions: 1) Complete array of digital amplifiers such that all amplifier transfer functions are transformable.

2) Shunt admittances at each port, all of which are transformable.

The matrix equations may be written:

$$\begin{bmatrix} I_1(s) \\ I_2(s) \\ \vdots \\ I_N(s) \end{bmatrix} = \begin{bmatrix} Y_{11}(s) & y_{12}(s) & \cdots & y_{1N}(s) \\ y_{21}(s) & y_{22}(s) & \cdots & y_{2N}(s) \\ \vdots & \vdots & & \vdots \\ y_{N1}(s) & y_{N2}(s) & \cdots & y_{NN}(s) \end{bmatrix} \begin{bmatrix} V_1^*(s) \\ V_2^*(s) \\ \vdots \\ V_N^*(s) \end{bmatrix}$$

$$+ \begin{bmatrix} y'_{11}(s) & 0 & \cdots & 0 \\ 0 & y'_{22}(s) & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & y'_{NN}(s) \end{bmatrix} \begin{bmatrix} V_1(s) \\ V_2(s) \\ \vdots \\ V_N(s) \end{bmatrix} \qquad (3.45)$$

or:

$$I(s) = Y_3(s)\, V^*(s) + Y_4(s)\, V(s) \qquad (3.46)$$

where $Y_3(s)$ and $Y_4(s)$ are transformable.

Hence by hypothetically sampling I(s), the following results:

$$I^*(s) = T_s Y_3^*(s)\, V^*(s) + T_s Y_4(s)^* V^*(s) \qquad (3.47)$$

Hence:

$$Y(z) = T_s \left[Y_3(z) + Y_4(z) \right] \qquad (3.48)$$

and:

$$Z(z) = \frac{1}{T_s} \left[Y_3(z) + Y_4(z) \right]^{-1} \qquad (3.49)$$

Case 6

Conditions: 1) Complete array of digital amplifiers such that all

amplifier transfer functions are transformable.

2) Shunt admittances at each port, none of which are transformable.

Referring to equation (3.46), none of the elements of $Y_4(s)$ are transformable. Therefore because $Y_4(s)$ is a diagonal matrix, $Y_4^{-1}(s)$ is transformable. Note that none of the shunt admittances may be zero in this case.

Thus rearrange (3.46):

$$Y_4^{-1}(s)\, I(s) = Y_4^{-1}(s)\, Y_3(s)\, V^*(s) + V(s) \qquad (3.50)$$

now making V(s) the output column vector. Now sample hypothetically V(s), thus also sampling I(s):

$$\left[Y_4^{-1}(s) \right]^* I^*(s) = \left[Y_4^{-1}(s)\, Y_3(s) \right]^* V^*(s) + V^*(s) \qquad (3.51)$$

Case 6 is now following Case 3 and thus:

$$Y(z) = T_s \left[Z\left\{ Y_4^{-1}(s) \right\} \right]^{-1} \left[U + Z\left\{ Y_4^{-1}(s)\, Y_3(s) \right\} \right] \qquad (3.52)$$

$$Z(z) = \frac{1}{T_s} \left[U + Z\left\{ Y_4^{-1}(s)\, Y_3(s) \right\} \right]^{-1} Z\left\{ Y_4^{-1}(s) \right\} \qquad (3.53)$$

The techniques described do not cover every possible combination of transformable and untransformable network. However the technique of separating the matrix into transformable and untransformable portions will allow all patterns to be analysed.

The resulting digital admittance matrix may of course have any node suppressed by pivotal condensation.

3.8 INTRINSIC REALISABILITY

In Section 3.7 it was shown that the original equations are always transformable provided that all the off-diagonal elements are transformable, which in turn may be established by inspection. If each off-diagonal element has been simulated by a digital amplifier then it is intrinsically possible to find the transform of the transfer function. Thus the cases studied in Section 3.7 are those which naturally occur.

3.9 MATRIX STABILITY

For a digital amplifier array described by an N x N matrix in general called X(z) to be stable, the poles of every element must lie within the z-plane unit circle. Rather than considering every element independently, all the elements may be tested simultaneously by analysing the pole positions of the determinant of X(z). This assertion may be proved as follows by defining X(z) thus:

$$X(z)=\begin{bmatrix} \frac{A_{11}(z)}{B_{11}(z)} & \frac{A_{12}(z)}{B_{12}(z)} & \cdots\cdots\cdots\cdots & \frac{A_{1N}(z)}{B_{1N}(z)} \\ \frac{A_{21}(z)}{B_{21}(z)} & \frac{A_{22}(z)}{B_{22}(z)} & \cdots\cdots\cdots\cdots & \frac{A_{2N}(z)}{B_{2N}(z)} \\ \vdots & \vdots & & \vdots \\ \frac{A_{N1}(z)}{B_{N1}(z)} & \frac{A_{N2}(z)}{B_{N2}(z)} & \cdots\cdots\cdots\cdots & \frac{A_{NN}(z)}{B_{NN}(z)} \end{bmatrix} \qquad (3.54)$$

By multiplying out, the determinant of X(z) may be derived:

$$\left| X(z) \right| = \frac{C(z)}{\prod_{i=1}^{N} \left[\prod_{j=1}^{N} B_{ij}(z) \right]} \qquad (3.55)$$

and $C(z) = f(A_{11}(z), A_{12}(z), \ldots\ldots, A_{NN}(z), B_{11}(z), B_{12}(z), \ldots . B_{NN}(z))$

Thus the denominator of the determinant of X(z) is the product of the denominators of each element of X(z) and thus consists of every pole of every element of X(z) which must lie within the z-plane unit circle for stability.

From [15] the accuracy by which the coefficients of the denominator of the determinant of X(z) can be known also contributes to the stability of the digital amplifier array. Thus the denominator of equation (3.55) may be rewritten thus:

$$\prod_{i=1}^{N} \left[\prod_{j=1}^{N} B_{ij}(z) \right] = \sum_{i=o}^{m} b_i z^{i} \qquad (3.56)$$

Now let:

$$b_k = \text{Max}\ (b_0, b_1, \ldots\ldots b_m) \qquad (3.57)$$

$$b_{sum} = \sum_{i=o}^{m} b_i \qquad (3.58)$$

To maintain stability, any error Δb_k in b_k due to coefficient truncation must be less than half a quantisation step, thus:

$$\frac{\Delta b_k}{b_{sum}} < \frac{1}{2} (2^n) \qquad (3.59)$$

where n is the number of binary bits in the coefficient word.

Thus:

$$n > \log_2 (\Delta b_k / b_{sum}) + 1 \qquad (3.60)$$

However in the case of digital amplifier arrays some or all the coefficients may be derived from external real components having intrinsically analogue values, and this last criterion will not apply to these coefficients and therefore the value for n may be considerably less than inequality (3.60) would suggest.

However, each digital amplifier contains a quantisation stage and thus the output variable at a port will contain the results of the quantisation of all the input variables which in turn may depend on other output variables. Hence a signal may be quantised, filtered, quantised and so on quite a number of times, thereby introducing a different form of round-off or truncation error.

3.10 INITIAL AND FINAL VALUES

The initial and final value theorems [11] may be applied to the matrix of the digital amplifier array by calculating the initial and

final values of the output vector.

3.10.1 Initial Value Theorem

The initial value theorem may be used to determine the initial output state of a digital machine.

The theorem states:

$$c(0) = \underset{|z| \Rightarrow \infty}{Lt} \; (C(z)) \tag{3.61}$$

where c(0) is the time domain initial output state and C (z) is the PTF.

To conserve units in (3.61), it is necessary to include a notional constant input of unit magnitude with appropriate units.

Equation (3.61) may now be extended to the matrix case thus:

$$\left[d(0)\right] = \underset{|z| \Rightarrow \infty}{Lt} \left\{D(z)\right\} \quad S \tag{3.62}$$

where $\left[d(0)\right]$ is the initial value of the output rector, D(z) the PTF matrix and S is a unitary matrix.

Thus the initial value may be calculated by taking the limit for each element of D and then summing each row of D to make $\left[d(0)\right]$.

3.10.2 Final Value Theorem

The final value theorem may be used to determine the final state of the output of a digital machine an infinite time after setting up.

The theorem states:

$$c(\infty) = \underset{z \Rightarrow 1}{Lt} \; ((1 - z^{-1}) \; C(z)) \tag{3.63}$$

where c (∞) is the time domain final value and C(z) is the PTF.

Again it is necessary to include a notional constant input of unit magnitude to conserve units.

Equation (3.63) may now be extended to the matrix case thus:

$$\left[d(\infty)\right] = \underset{z \Rightarrow 1}{Lt} \left\{D(z)\right\} \; S \tag{3.64}$$

where $\left[d(\infty)\right]$ is the final value of the output vector, D (z) is the PTF matrix and S is a unitary matrix.

Thus the final value may be calculated by taking the limit for each element of D and then summing each row of D to make $\left[d(\infty) \right]$.

3.11 LIMIT CYCLE NOISE

The effects of amplitude quantisation within a digital amplifier have been discussed in Section 2.5.

In general, a digital amplifier array will consist of digital amplifiers simulating each matrix element, and each of these amplifiers will introduce quantisation noise.

In the case of a 2-port digital active network represented by a digital admittance matrix, this quantisation becomes noise voltages convolving with the matrix admittances. Hence let V_{Nij} be the quantisation noise voltage introduced into the admittance y_{ij} producing an output noise current thus:

$$I_{Nij} = y_{ij} \; V_{Nij} \tag{3.65}$$

This noise voltage V_{ij} is intimately associated with the generation of y_{ij} and hence only convolves with y_{ij}. Furthermore the quantisation noise can only be present when a signal is present as it is created by that signal crossing quantisation boundaries.

Applying the results from Section 2.5 to equations (3.9) and (3.10) gives:

$$I_1(s) = g_{11}f_{11}(s)\, V_1(s) + g_{12}f_{12}(s)\, V_2^*(s) + g_{12}f_{12}(s)V^*_{N2}(s) \tag{3.66}$$

$$I_2(s) = g_{21}f_{21}(s)\, V_1^*(s) + g_{21}f_{21}(s)\, V^*_{N1}(s) + g_{22}f_{22}(s)\, V_2(s) \tag{3.67}$$

Manipulating these equations, rewriting them as matrices and taking the Z-transform gives:

$$\begin{bmatrix} I_1(z) \\ I_2(z) \end{bmatrix} = \begin{bmatrix} g_{11}f_{11}(z) & g_{12}f_{12}(z) \\ g_{21}f_{21}(z) & g_{22}f_{22}(z) \end{bmatrix} \begin{bmatrix} V_1(z) \\ V_2(z) \end{bmatrix} + \begin{bmatrix} 0 & g_{12}f_{12}(z) \\ g_{21}f_{21}(z) & 0 \end{bmatrix} \begin{bmatrix} V_{N1}(z) \\ V_{N2}(z) \end{bmatrix} \quad (3.68)$$

Equation (3.65) may be rewritten as:

$$I(z) = Y(z)\, V(z) + Y_1(z)\, V_N(z) \quad (3.69)$$

where $Y(z)$ is assumed to have been found by whichever technique is applicable, and $Y_1(z)$ is the Z-transform of $Y_1(s)$ as defined in equation (3.23).

Thus the output current vector $I(z)$ contains a noise current $I_N(z)$:

$$I_N(z) = Y_1(z)\, V_N(z) \quad (3.70)$$

This is equivalent to regarding the noise as coming from external current generators in shunt with the ports.

When the output vector is to be the voltage $V(z)$ then equation (3.69) may be rearranged thus:

$$V(z) = Z(z)\, I(z) - Z(z)\, Y_1(z)\, V_N(z) \quad (3.71)$$

The output voltage vector $V(z)$ now contains a noise voltage $V_Q(s)$ where:

$$V_Q(z) = Z(z)\, Y_1(z)\, V_N(z) \quad (3.72)$$

$$= A(z)\, V_N(z)$$

where $A(z) = Z(z)\, Y_1(z)$ (3.73)

$V_Q(z)$ may be considered to have been generated by noise current generators, in shunt with any port which has a digital amplifier output connected to it, convolving with the digital impedance matrix.

Now in general the number of bits and the dynamic range of each A/D converter in a digital amplifier array will be the same and hence the noise voltage vector will become:

$$V_N(z) = V_N \begin{bmatrix} 1 \\ 1 \\ . \\ . \\ . \\ 1 \end{bmatrix} \qquad (3.74)$$

$V_Q(z)$ may then be seen to be the sum of each row of A(z) scaled by V_N in equation (3.72).

Now equation (3.74) may be simplfied using equation (2.52) to:

$$V_N(z) = \sqrt{\frac{V_{STEP}^2}{12}} \quad S \qquad (3.75)$$

where S is a unit column vector. Thus equation (3.72) becomes:

$$V_Q(z) = \sqrt{\frac{V_{STEP}^2}{12}} \quad A(z) \cdot S \qquad (3.76)$$

This allows the R.M.S. noise voltage at any given port to be computed.

Furthermore the resulting quantisation noise voltage vector $V_Q(z)$ may have elements large enough to appear as a signal input to the network. This may be expressed thus:

$$V_{Qi}(z) > V_{Nj}(z) \qquad (3.77)$$

As $V_{Nj}(z)$ represents a quantised voltage, equation (3.77) will be true if:

$$V_{Qi}(z) = q \; V_{Nj}(z) \qquad (3.78)$$

where:

$$q = 2, 3, 4, ... \qquad (3.79)$$

To satisfy equation (3.79) the value of any element of A(z) would have to be:

$$a_{ij}(z) \geq 2 \qquad (3.80)$$

Thus if any element of $A(z)$ has a magnitude which equals or exceeds 2 at any frequency or range of frequencies then a limit cycle noise voltage will be present at each port of the digital amplifier array. However due to the constraint of equation (3.77) it is possible to set up a digital amplifier array in a quiescent state such that there has not been an input signal to start off the process of limit cycle noise generation. Once any input is fed into the digital amplifier array at any port such that that input signal exceeds V_N then the limit cycle noise will start up automatically and be self-sustaining.

Now this limit cycle noise will not build up to cause limiting within the digital amplifier array because however large the voltage $V_{Qi}(s)$ is at the i^{th} port, the amplitude of $V_{Nj}(z)$ is unaffected, being merely made up of many equal amplitude steps. Thus limiting can only occur if:

$$a_{ij}(z) \geqslant M \tag{3.81}$$

where M is the number of quantisation levels present in the code used in the digital amplifiers.

When equation (3.80) is true then limit cycle noise will be present. However even when $a_{ij} < 2$, noise will be present whenever a signal is injected into the digital amplifier array.

Now the presence of a limit cycle noise voltage V_{Qi} at the i^{th} port will affect the accuracy of any measurements of the transimpedances to that port. Thus letting the total observed voltage be V_i' :

$$V_i' = V_i + V_{Qi} \tag{3.82}$$

The transimpedance to that port from the j^{th} port can be measured when I_j is known:

$$\frac{V_i'}{I_j} = z_{ij} + \Delta z_{ij} \tag{3.83}$$

where Δz_{ij} is the error in z_{ij} caused by this limit cycle noise. Thus the fractional error in z_{ij} is:

$$\frac{\Delta z_{ij}}{z_{ij}} = \frac{V_{Qi}}{V_i} \tag{3.84}$$

The fractional error will depend entirely on the relative signal and limit cycle noise amplitudes. Furthermore the fractional error in the transimpedance is the inverse of the signal-to-limit cycle noise ratio at the i^{th} port.

3.12 ELEMENT RESOLUTION

The input voltage vector to the digital admittance matrix in equation (3.5) inevitably represents voltages which are quantised into M levels by the A/D converters at the input to each digital amplifiers. Thus because the admittance matrix elements offer no quantisation, the output vector will contain currents quantised into M levels and to evaluate the digital admittances by measuring input voltages and currents introduces an uncertainty.

Hence from equation (3.5):

$$y_{ij} = I_i / V_j$$

Thus after manipulation the fractional error in the digital trans-admittance will be:

$$\left.\frac{\Delta y_{ij}}{y_{ij}}\right|_{MAX} = \frac{\Delta I}{I_i} + \frac{\Delta V}{V_j} \tag{3.85}$$

where Δy_{ij} is the error in the digital admittance y_{ij} caused by the quantised voltage (ΔV). However, by definition:

$$\Delta V < \tfrac{1}{2} \text{ LSB} \tag{3.86}$$

and thus provided the quantised voltage obeys this inequality then:

$$\Delta I = 0$$

If the magnitude of the input voltage V_j is adjusted to be maximum then:

$$\frac{\Delta V}{V_j} = \frac{1}{M} \tag{3.87}$$

and equation (3.73) may be rewritten:

$$\left.\frac{\Delta y_{ij}}{y_{ij}}\right|_{MAX} = \frac{1}{M} \tag{3.88}$$

which is the quantisation resolution. The measured admittance $y_{ij}(z)$ will vary by $\Delta y_{ij}(z)$ as the input voltage varies across one quantisation level.

This same calculation may be repeated for the digital impedance matrix in equation (3.4), giving:

$$\left.\frac{\Delta z_{ij}}{z_{ij}}\right|_{MAX} = \frac{1}{M} \tag{3.89}$$

Thus the matrix elements can be found only to the nearest 100/M percent. This gives a powerful reason for making M as large as possible by designing the A/D converter to quantise with as many levels as possible.

The computed impedances and admittances reported in Chapter 8 will therefore vary over the range of $\pm$ 100/M percent and thus any practical measurement which lies within this range may be regarded as valid.

3.13 SUMMARY AND CONCLUSIONS

The digital admittance and impedance matrices have been defined. Appropriate analysis techniques have been developed for an N-port digital active network constructed of a mixture of digital amplifiers and analogue active or passive components. It has been shown that either the digital impedance or admittance matrices may be directly found and the other matrix obtained by inversion.

The overall stability of the whole digital admittance or impedance

matrix has been shown to depend on the denominator of the determinant of that matrix.

A limit cycle signal has been shown to be present at the ports of a digital network under certain conditions and this will restrict the use of digital active networks. However the conditions which cause this problem are easy to derive from the appropriate matrix and therefore may be determined before a network is constructed.

CHAPTER 4

2-PORT DIGITAL GYRATOR ANALYSIS

4.1 INTRODUCTION

In order to verify the mathematics presented in Chapter 3, a digital amplifier array (as described in Chapter 6) was constructed and arranged as a 2-port digital gyrator. Both ports were capacitively loaded and the network was fed from a current source shunted by a conductance. Fig. 4.1 shows the actual arrangement. The digital transadmittance amplifiers have a block diagram as in Fig. 4.2. The Sample-and-Hold stage contributes a Zero-Order-Hold to the transfer function, and the lumped time delay stage includes all actual time delays.

The digital gyrator was fed from a current source deliberately so that the voltages at the ports could be observed. Thus as the input current was defined, the digital impedance matrix was effectively being tested.

4.2 GYRATOR ANALYSIS

The transadmittance amplifiers of Fig. 4.2 have the following transfer functions:

$$g_{21}.f_{21}(s) = -g_2(1-\exp(-sT_s)).\exp(-sk_2T_s) / s \qquad (4.1)$$

$$g_{12}.f_{12}(s) = g_1(1-\exp(-sT_s)).\exp(-sk_1T_s) / s \qquad (4.2)$$

where $(1-\exp(-sT_s)) / s$ is the transfer function of a zero-order hold and:

$\exp(-skT_s)$ represents a time delay of kT_s seconds and T_s is the sampling period.

From Section 2.3, it can be seen that $f_{21}(s)$ and $f_{12}(s)$ are transformable.

The overall equations describing the gyrator in Fig. 4.1 are:

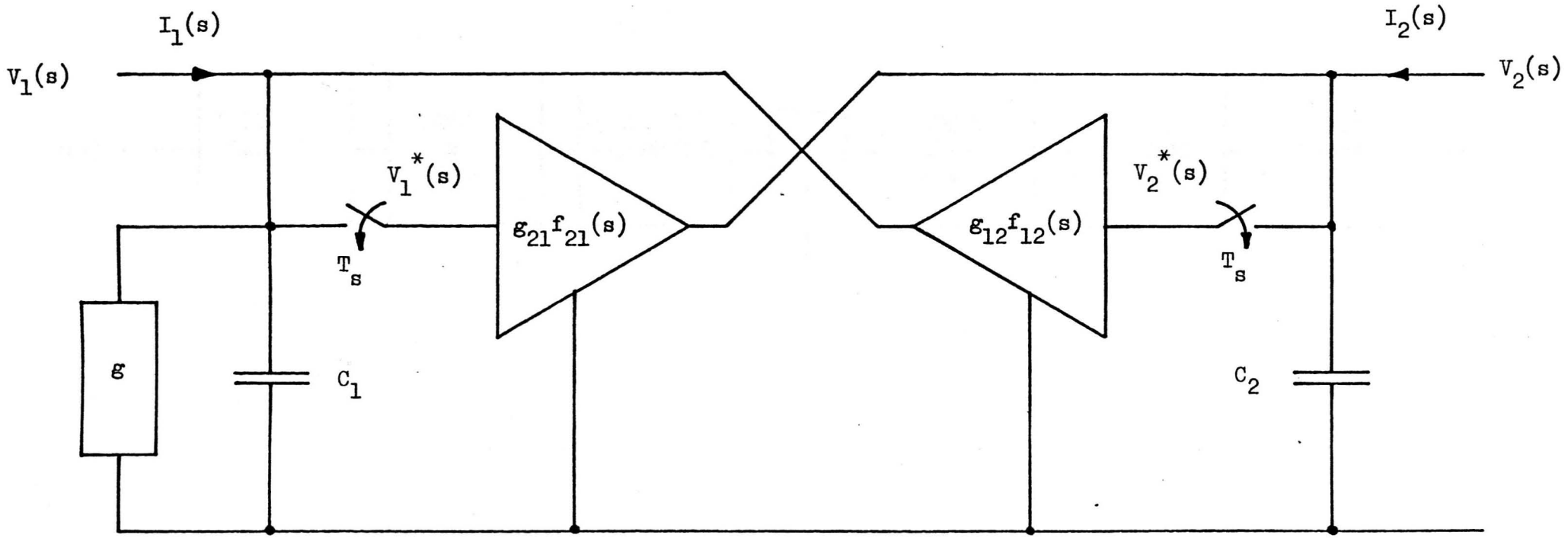


FIG. 4.1 2-port Digital Gyrator

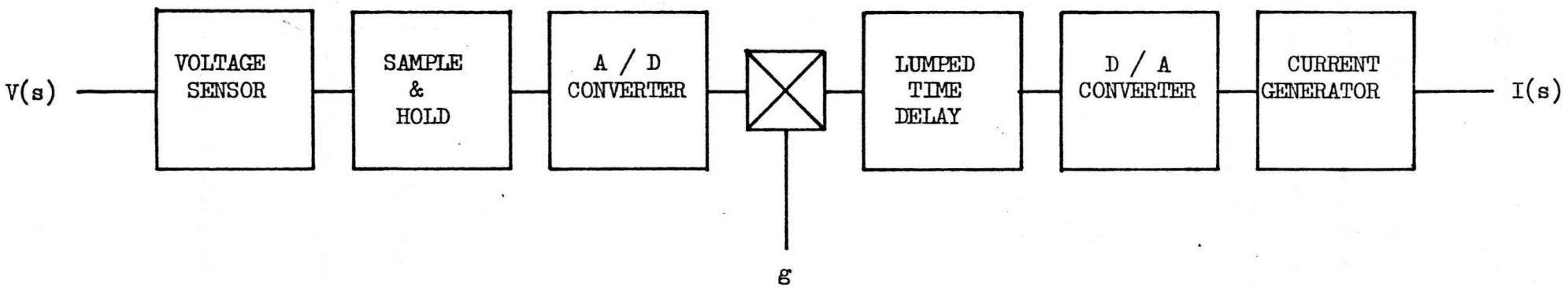


FIG. 4.2 Digital Gyrator Transadmittance Amplifiers

$$I_1(s) = (sC_1+g)\, V_1(s) + g_1\,(1-\exp(-sT_s))\exp(-sk_1T_s)V_2^*(s)/s \tag{4.3}$$

$$I_2(s) = -g_2(1-\exp(-sT_s))\exp(-sk_2T_s)\, V_1^*(s)/s + sC_2V_2(s) \tag{4.4}$$

By inspection it can be seen that $(sC_1 + g)$ and sC_2 are not transformable. Hence the result of Case 3 in Chapter 3 is used to find Y(z) from equation (3.31) and Z(z) from equation (3.30) thus:

$$Y(z) = \frac{1}{T_s}\left[\, Z\left\{ Y_2^{-1}(s)\right\}\right]^{-1}\left[\, U + Z\left\{ Y_2^{-1}(s)Y_1(s)\right\}\right] \tag{4.5}$$

where:

$$Y_1(s) = \begin{bmatrix} 0 & g_1(1-\exp(-sT_s))\exp(-skT_s)/s \\ -g_2(1-\exp(-sT_s))\exp(-sk_2T_s)/s & 0 \end{bmatrix} \tag{4.6}$$

and:

$$Y_2(s) = \begin{bmatrix} sC_1 + g & 0 \\ 0 & sC_2 \end{bmatrix} \tag{4.7}$$

Hence:

$$Y_2^{-1}(s) = \begin{bmatrix} \dfrac{1}{sC_1+g} & 0 \\ 0 & \dfrac{1}{sC_2} \end{bmatrix} \tag{4.8}$$

Therefore:

$$Y_2^{-1}(s)Y_1(s) = \begin{bmatrix} 0 & g_1(1-\exp(-sT_s))\exp(-sk_1T_s)/s(sC_1+g) \\ -g_2(1-\exp(-sT_s))\exp(-sk_2T_s)/s^2C_2 & 0 \end{bmatrix} \tag{4.9}$$

Hence the Z-transform of matrix (4.8) is:

$$\begin{bmatrix} \frac{1}{C_1} Z\left\{\frac{1}{s+g/C_1}\right\} & 0 \\ 0 & \frac{1}{C_2} Z\left\{\frac{1}{s}\right\} \end{bmatrix} \qquad (4.10)$$

Now, letting $T = C_1/g$, matrix (4.10) becomes:

$$\begin{bmatrix} z / C_1(z-\exp(-T_s / T)) & 0 \\ 0 & z / C_2(z-1) \end{bmatrix} \qquad (4.11)$$

Taking the Z-transform of matrix (4.9) gives:

$$\begin{bmatrix} 0 & g_1 z^{-k_1} (1-\exp(-T_s/T))/g(z-\exp(-T_s/T)) \\ -g_2 T_s z^{-k_2}/C_2(z-1) & 0 \end{bmatrix} \qquad (4.12)$$

Now substituting into equation (4.5) gives:

$$Y(z) = \frac{1}{T_s} \begin{bmatrix} C_1(z-\exp(-T_s/T))/z & 0 \\ 0 & C_2(z-1)/z \end{bmatrix} \begin{bmatrix} 1 & \left[\frac{g_1 z^{-k_1}(1-\exp(-T_s/T))}{g\ (z-\exp(-T_s/T))}\right] \\ -g_2 T_s z^{-k_2}/C_2(z-1) & 1 \end{bmatrix}$$

Multiplying out and replacing $\exp(-T_s/T)$ by α gives:

$$Y(z) = \frac{1}{T_s} \begin{bmatrix} C_1\ (z-\alpha)/z & g_1 \frac{C_1}{g} z^{-k_1-1} (1-\alpha) \\ -g_2 T_s z^{-k_2-1} & C_2(z-1)\ /z \end{bmatrix} \qquad (4.13)$$

Equation (3.30) may be used to find Z(z):

$$\begin{aligned} Z(z) &= T_s \left[U + Z\left\{ Y_2^{-1}(s)\ Y_1(s) \right\} \right]^{-1} Z\left\{ Y_2^{-1}(s) \right\} \\ &= T_s \begin{bmatrix} 1 & \frac{g_1}{g} z^{-k_1} \frac{(1-\alpha)}{(z-\alpha)} \\ -\frac{g_2}{C_2} \frac{T_s z^{-k_2}}{(z-1)} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \frac{1}{C_1} \frac{z}{(z-\alpha)} & 0 \\ 0 & \frac{1}{C_2} \frac{z}{(z-1)} \end{bmatrix} \end{aligned} \qquad (4.14)$$

$$Z(z) = T_s \left[\begin{array}{cc} z^{k_1+k_2+1}(z-1)/C_1 & -g_1 z^{k_2+1}(1-\alpha)/gC_2 \\ g_2 T_s z^{k_1+1}/C_1 C_2 & z^{k_1+k_2+1}(z-\alpha)/C_2 \end{array} \right] \Big/ M(z) \qquad (4.15)$$

where:

$$M(z) = z^{k_1+k_2}(z-1)(z-\alpha) + T_s(1-\alpha)g_1 g_2/gC_2 \qquad (4.16)$$

4.3 GYRATOR STABILITY

The overall stability of the matrices Y(z) and Z(z) may be found from studying the values of z which cause these matrices to be singular. The values of z are the poles of the determinants of Y(z) and Z(z). If all these poles lie within the unit-circle, then the matrix concerned is unconditionally stable.

From (4.13):

$$\left| Y(z) \right| = \frac{C_1 C_2}{T_s^2} \cdot \frac{M(z)}{z^{k_1+k_2+2}} \qquad (4.17)$$

where M(z) is defined in (4.16). $\left|Y(z)\right|$ has multiple poles only at the origin z = 0, and is thus unconditionally stable.

From (4.15):

$$\left| Z(z) \right| = \frac{T_s^2}{C_1 C_2} \cdot \frac{z^{k_1+k_2+2}}{M(z)} \qquad (4.18)$$

which is also the inverse of $\left| Y(z) \right|$

$\left| Z(z) \right|$ has poles when M(z) = 0

This is the same condition as applies to any of the individual elements of Z(z). Hence the matrix Z(z) may be unstable, depending upon the roots of M(z).

Now consider M(z) from equation (4.16):

$$M(z) = z^{k_1+k_2}(z-1)(z-\alpha) + \frac{g_1 g_2}{g\, C_2} G_s(1-\alpha)$$

The roots of M(z) will be simplified if $\alpha \Rightarrow 1$. Hence consider the limit:

$$\underset{\alpha \Rightarrow 1}{Lt} \left\{ M(z) \right\} = z^{k_1+k_2}(z-1)^2 \qquad (4.19)$$

In this case M(z) has 2 roots at z = 1 and $k_1 + k_2$ roots at the origin. However the roots at the origin are cancelled by the numerator of (4.2). Thus when $\alpha \Rightarrow 1$, this digital impedance matrix becomes unstable at D.C, that is it will latch up.

However by definition:

$$\alpha = \exp(-T_s/T) = \exp(-T_s\, g/C_1)$$

and therefore α will tend to unity when g or T_s tend to zero or C_1 tends to infinity, thereby making the digital impedance matrix unstable.

4.4 INITIAL AND FINAL VALUES

The initial and final output vector values may be found using the simple algorithms defined in Section 3.10.

4.4.1 Digital Admittance Matrix

The initial output current from the digital admittance matrix (4.13) will be:

$$\left[i\,(o) \right] = \underset{z \Rightarrow \infty}{Lt} \left| Y(z) \right| = \begin{bmatrix} C_1 / T_s \\ C_2 / T_s \end{bmatrix} \qquad (4.20)$$

and thus the initial output current will be finite and the matrix initially stable.

The final output current from the digital admittance matrix will be:

$$\left[i\,(\infty) \right] = \underset{z \Rightarrow 1}{Lt} \left[\left(\frac{z-1}{z} \right) Y(z) \right] = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad (4.21)$$

and thus the final output current is zero making the matrix unconditionally stable.

4.4.2 Digital Impedance Matrix

The initial output voltage from the digital impedance matrix (3.15) will be:

$$\begin{bmatrix} v(o) \end{bmatrix} = \underset{z \to \infty}{Lt} \left\{ Z(z) \right\} = \begin{bmatrix} T_s / C_1 \\ T_s / C_2 \end{bmatrix} \qquad (4.22)$$

and thus the initial output voltage will be finite and the matrix initially stable.

The final value of the output voltage vector from the digital impedance matrix will be:

$$\begin{bmatrix} v(\infty) \end{bmatrix} = \underset{z \to 1}{Lt} \left\{ \left(\frac{z-1}{z} \right) Z(z) \right\} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad (4.23)$$

irrespective of the value of α .

This result would suggest a more optimistic solution than that found in practice. Both the computer results of Chapter 8 and the practical results of Chapter 7 indicate that the output voltage will oscillate under certain parameter values. Effectively the final value has only found the mean of the output voltages.

Alternatively the limit taken for the final value should only be taken when all the poles lie within the unit circle. When they migrate outside this circle then this test becomes meaningless.

4.5 LIMIT CYCLE NOISE

The actual quantisation noise currents and voltages may be calculated from Section 3.11. However $Y_1(z)$ must first be calculated:

$$Y_1(s) = \begin{bmatrix} 0 & g_1(1-\exp(-sT_s))\exp(-sk_1T_s)/s \\ -g_2(1-\exp(-sT_s))\exp(-sk_2T_s)/s & 0 \end{bmatrix} \qquad (4.24)$$

which is directly transformable to:

$$Y_1(z) = \begin{bmatrix} 0 & g_1 z^{-k_1} \\ -g_2 z^{-k_2} & 0 \end{bmatrix} \tag{4.25}$$

Hence from equation (3.70) the noise current vector $I_N(z)$ is:

$$I_N(z) = \begin{bmatrix} 0 & g_1 z^{-k_1} \\ -g_2 z^{-k_2} & 0 \end{bmatrix} V_n(z) \tag{4.26}$$

From equation (3.70) the noise voltage transfer function A(z) is:

$$A(z) = T_s \frac{\begin{bmatrix} g_1 g_2 (1 - \alpha) z / g C_2 & g_1 z^{k_2+1} (z-1)/C_1 \\ -g_2 (z - \alpha) z^{k_1+1} / C_2 & g_1 g_2 T_s z / C_1 C_2 \end{bmatrix}}{M(z)} \tag{4.27}$$

where M(z) is defined in equation (4.16).

The noise voltage at ports 1 and 2 of the digital gyrator may be calculated using equation (2.52):

$$V_Q(z) = \sqrt{\frac{V_{STEP}^2}{12}} \cdot A(z) \cdot S \tag{4.28}$$

which may be written:

$$V_Q(z) = \sqrt{\frac{\frac{V_{STEP}^2}{12}}{M(z)}} \begin{bmatrix} \frac{T_s g_1}{C_1} z^{k_2+2} & -\frac{T_s g_1}{C_1} z^{k_2+1} & +\frac{T_s g_1 g_2}{g C_2} (1 - \alpha) z \\ -\frac{T_s g_2}{C_2} z^{k_1+2} & + \frac{T_s g_2}{C_2} z^{k_1+1} & + \frac{g_1 g_2 T_s^2}{C_1 C_2} z \end{bmatrix} \tag{4.29}$$

4.6 SUMMARY AND CONCLUSIONS

A two port capacitively loaded digital gyrator has been analysed

to give both its digital impedance and admittance matrix from the results of Chapter 3. The stability of these actual matrices has been derived and it has been shown that the admittance matrix is unconditionally stable, whereas the impedance matrix is only conditionally stable.

The quantisation noise transfer matrix has also been derived for this gyrator ready for computer evaluation.

CHAPTER 5

COMPUTER PROGRAMS

5.1 INTRODUCTION

Four computer programs were written to enable the analysis of the gyrator to be compared with the practical results described in Chapter 7. It is not intended to describe each step of each program as these programs were necessary to verify the analysis, but not innovative.

The programs were written in FORTRAN IV on a mini-computer with a disk-based operating system. The operating system required that programs producing executable code in excess of 8K words should be overlaid in segments, and thus various libraries of subroutines were also written as overlay segments.

All the listings are presented in Appendix D and grouped according to main program or overlay segment. Where an overlay segment was used by more than one main program, the segment is still only listed once. Each main program has a list near its beginning of all the overlay segments it uses. Each main program or overlay segment listed in Appendix D was written specifically for the work presented in this thesis.

The actual graph plotting was programmed by calling a standard graph plotting library available on this mini-computer and the FORTRAN subroutines of this library are listed under PLOTTER.

5.2 GYRATOR DATA PREPARATION PROGRAM (GDP1)

This program was written to evaluate the constituent polynomials of the digital impedance, admittance and quantisation matrix elements for a selection of values of each of the parameters (in Table 5.1) where the meanings of the parameters are defined in Fig. 4.1. GDP1 was intended to calculate only intermediate results for later processing.

The rational polynomial of a matrix element was calculated and

PARAMETER	PARAMETER MEANING	VALUE
f_s	Sampling frequency	26.6 KHz
T_s	Sampling period	18.75 microseconds
C_1	Port 1 shunt capacitor	9.6 microfarads
C_2	Port 2 shunt capacitor	10 microfarads
g_1, g_2	Transconductances	10 millisiemens
R_s	Port 1 shunt resistor	50, 100, 200, 300, 400, 500 ohms
k_1, k_2	Fractional delay	1

TABLE 5.1

Digital Gyrator Component Values

stored in an unformatted (binary) disk file for use by the remaining three programs, namely PZP1, FRA1 and IZT1.

5.3 POLE-ZERO PLOTTING PROGRAM (PZP1)

This program was written to calculate the poles and zeroes of rational polynomials of up to fourth order, the results being listed on a line-printer and plotted on a graph plotter. First and second order polynomials were factorised conventionally while third order polynomials were factorised by finding a single real root and then reducing the polynomial to second order by dividing through by this root. Fourth order polynomials were factorised by Brown's method (see Appendix B) which involves finding the two quadratic factors of this polynomial and then factorising these quadratic factors.

This program could have been enhanced by implementing Bairstow's method [21], to enable n^{th} order polynomials to be factorised, the order being limited then by available time and the computer speed.

The polynomial order must be integral for this program, though it is clear from matrices (4.13) and (4.15) that the order need not be integral in practice if the fractional delays k_1 and k_2 are not integral.

The actual polynomial variable is arbitrary but the graph plotting was designed to accept the Laplace variable s or the Z-transform variable z, a unit circle being plotted on the complex plane for the latter case.

5.4 FREQUENCY RESPONSE ANALYSIS PROGRAM (FRA1)

This program was written to find the frequency response for a rational polynomial in the Laplace variable s or the Z-transform variable z between any two given frequencies in logarithmic or linear frequency increments. FRA1 was written to read data from the control terminal or from one of the disk files made by GDP1. The output data can be listed on the line printer and drawn by the graph plotter.

FRA1 uses a scratch disk file (on unit 6) to store the results of the complete analysis as an unformatted (binary) disk file which is then rewound and read back for the line-printer output, for the Bode magnitude andphase plots, and for the Nyquist plot, as required.

The algorithm used to calculate the frequency response of a z-plane polynomial is explained in Appendix E.

5.5 INVERSE Z-TRANSFORM PROGRAM (IZT1)

This program was written to calculate the inverse Z-transform and impulse response of a z-plane polynomial PTF. The input data was able to be read from the control terminal or from one of the disk files made by GDP1 and the output data was then listed on the line-printer and drawn by the graph plotter.

The algorithm to find the inverse Z-transform [11] involves dividing out the rational transfer function ad infinitum and this was implemented by repeated polynomial long division of the numerator by the denominator. This algorithm suffers from the disadvantage that rounding errors slowly accumulate in the remainder thereby sometimes causing gross errors after several thousand iterations. This is particularly prevalent in the infinite impulse response filter defined by the digital gyrator.

By premultiplying the rational PTF with the Z-transform of a unit step z/(z-1) and repeating the above algorithm, the step response could be found. By premultiplying with other functions the time domain output could have been calculated for these functions but this would have been of decreasing usefulness.

The initial and final value theorems [11] were also implemented to calculate these values for both the rational polynomial PTF and the impulse response. This served to confirm that rounding errors were sometimes seriously affecting the inversion algorithm. The algorithm

for the inverse Z-transform could have been modified to use double precision arithmetic but this was not considered to be necessary in the cases studied for this thesis.

CHAPTER 6

EXPERIMENTAL MACHINE

6.1 INTRODUCTION

An experimental machine was constructed to verify the theoretical analysis described in the previous chapters. The structure of this machine is shown in Fig. 6.1.

The design and construction of this machine is described along with the use of the various printed circuit boards to simulate a 2-port digital admittance matrix.

Various boards were constructed to test the machine and external equipment built to simplify the operation of the matrix.

The descriptions and drawings of the printed circuit boards in this chapter are not presented with the intention of providing a maintenance manual.

6.2 MACHINE ARCHITECTURE

6.2.1 Overall Structure

The machine as constructed consisted of 10 independent multiplexed channels with 2 voltage sensing A/D converters, 2 current generating D/A converters and 10 voltage generating D/A converters. Further, any two adjacent channels could be multiplied together, stored and added to another product pair provided that both pairs were within one frame. This could then be used to simulate one row of the 2-port digital admittance matrix. This structure is shown in Fig. 6.2.

6.2.2 Sample Rate

The input sampling rate was designed to be 1/60 of the master clock frequency, which in turn could be set to any one of five frequencies as listed in Table 6.1. All sampling was designed to be uniform and synchronous.

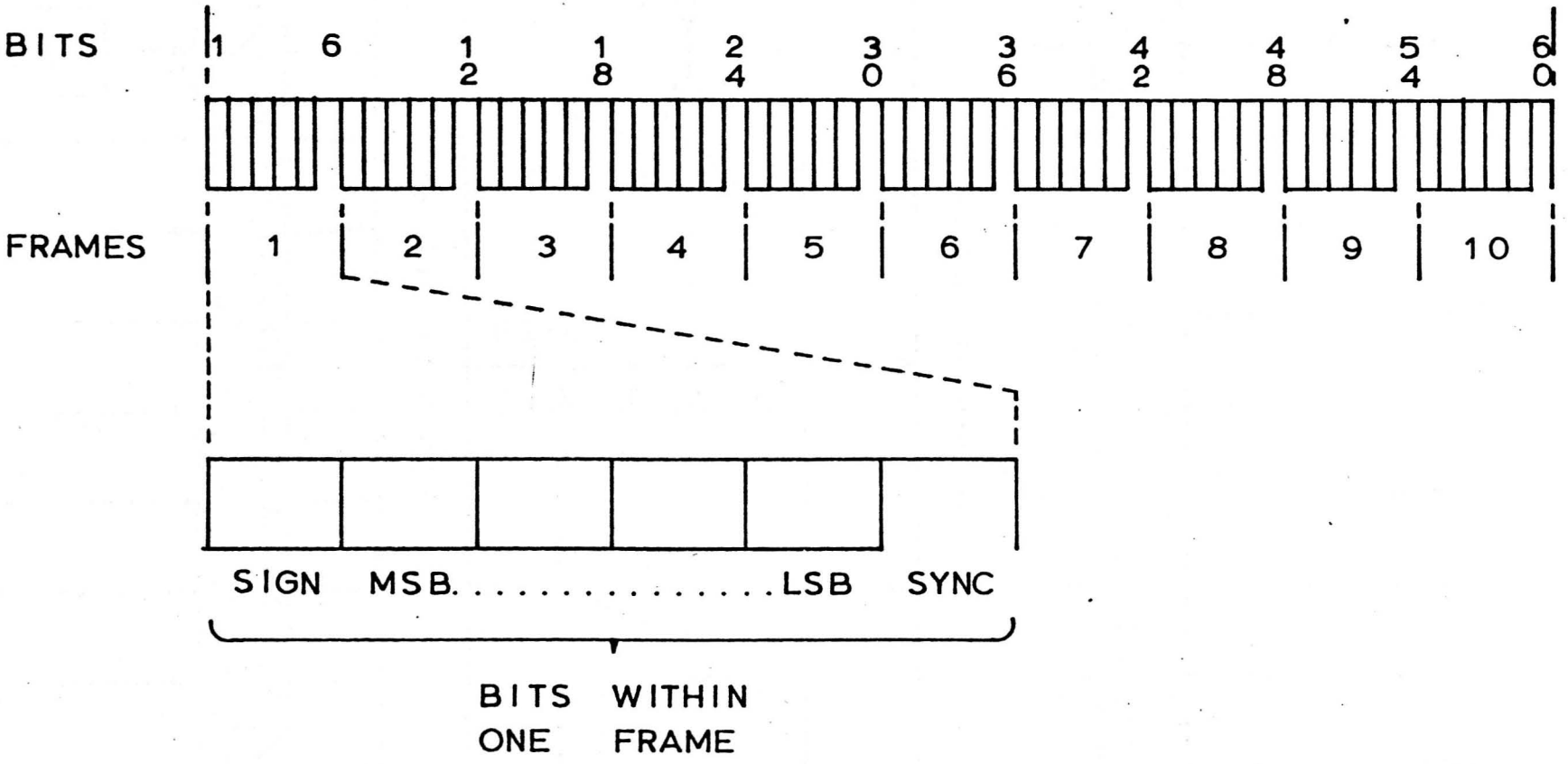


Fig. 6.1 Serial Data Structure

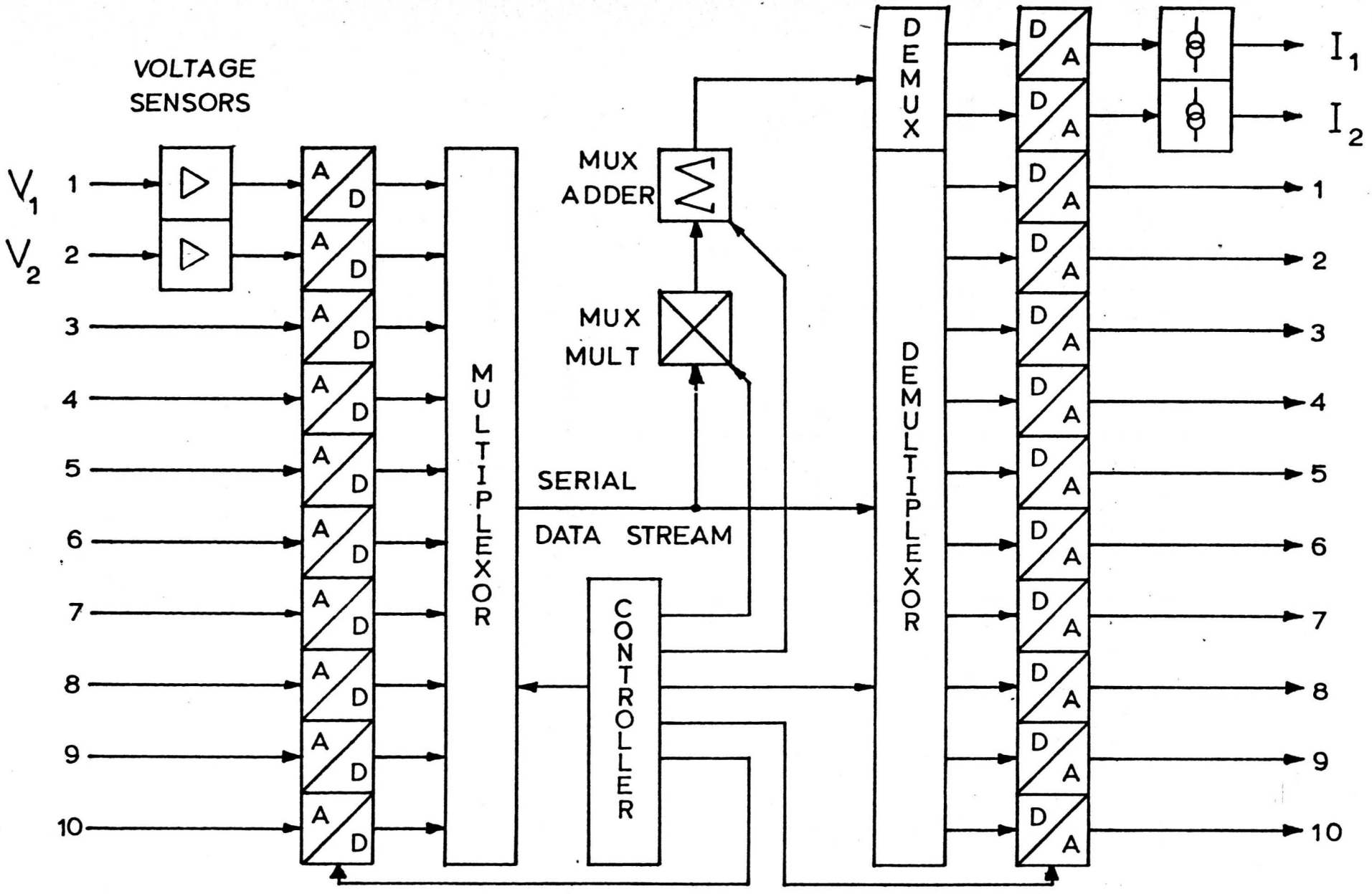


Fig. 6.2 Digital Machine Architecture

BASIC CLOCK FREQUENCY kHz	CONVERSION TIME / BIT ns	SAMPLING FREQUENCY kHz	SAMPLING PERIOD μs	NYQUIST RATE kHz
3200	312.5	53.3˙	18.75	26.6˙
1600	625	26.6˙	37.5	13.3˙
800	1250	13.3˙	75	6.6˙
400	2500	6.6˙	150	3.3˙
200	5000	3.3˙	300	1.6˙

TABLE 6.1

Sampling Parameters

The absolute maximum sampling rate was determined ultimately by the conversion time per bit of the A/D converter which in turn was governed by the total settling time of the D/A converter and the comparator in this converter. For this particular machine the minimum conversion time per bit was about 250 nanoseconds. Hence the equivalent single channel non-multiplexed A/D converter sample frequency would have been 10 times the sampling frequency shown in Table 6.1, due to the multiplexing of 10 channels.

6.2.3 Data Conversion

The input A/D conversion was performed on the 10 input channels by a multiplexed successive approximations A/D converter. (See Appendix C). This converter produced a 5-bit serial output consisting of a sign bit followed by 4 magnitude bits.

6.2.4 Serial Data Structure

Fig. 6.2 shows the serial data output produced by the multiplexed A/D converter. The 10 channels each consisting of 5 data bits and one synchronisation bit made 1 frame.

6.2.5 Input Quantisation

The A/D converter was designed to produce a 5-bit code consisting of a sign bit and 4 bits of magnitude. A two's complement code was not chosen to represent negative input values, but in retrospect the machine would have been a little simpler if this approach had been adopted.

From Appendix C the parameter values for this converter have been calculated and are listed in Table 6.2.

6.2.6 Output Quantisation

The output D/A converter was designed to accept a parallel 10-bit word consisting of a sign bit and 9 bits of magnitude. Table 6.3 shows the parameter values calculated from Appendix C.

The longer word length was required because the preceding multiplication and addition stages effectively doubled the work length.

EQUATION	PARAMETER	VALUE	UNITS	MEANING
C.1	M	31	-	Number of quantisation levels
C.2	ΔV	129	mV	Quantisation step size
C.3	A	3.23	%	Quantisation accuracy
C.4	R	29.8	dB	Dynamic Range
C.6	Sq	31.59	dB	Maximum signal / noise ratio

TABLE 6.2

A/D Input Converter Parameter Values

EQUATION	PARAMETER	VALUE	UNITS	MEANING
C.1	M	1023	-	Number of quantisation levels
C.2	Δ I	7.8	mA	Maximum Quantisation step size
C.3	A	0.098	%	Quantisation accuracy
C.4	R	60.2	dB	Dynamic range
C.6	Sq	62	dB	Maximum signal / noise ratio

TABLE 6.3

A/D Output Converter Parameter Values

However the overall parameter values were limited by the input converter as the state of the 10 bit word feeding this converter could only be in 1 of 31 possible states at any given instant.

The absolute current generated was directly proportional to the D/A converter reference voltage and the maximum output current was calculated from equation (6.24) as 399mA.

6.2.7 Delay Time

The delay time in the signal path from the voltage input to the current output was a minimum of sixth tenths of the sample period. However this could be increased and was normally set to one complete sample period, in order to make the mathematics considerably simpler. If the delay was not an integral number of sample periods then the polynomials in the matrix elements would have a non-integral order.

6.3 2-PORT MATRIX

The structure for the experimental machine to simulate a 2-port matrix is shown in Fig. 6.3. In this structure it has been assumed that all the matrix elements are simple digital transadmittance amplifiers (see Section 2.4), which are fed in through the A/D converters. The 2-port gyrator analysed in Chapter 4 only requires g_{12} and g_{21}, and hence a simpler structure. In practice the A/D converter, multiplier and adder were all multiplexed.

6.3.1 Sampling Input Buffer

In order to make digital transadmittance amplifiers it is necessary to present a very high input impedance. Fig. 6.4 shows the input buffer and sample-and-hold stage. A second buffer amplifier is included to present a low output source impedance.

6.3.2 Encoder

The principle of operation of the encoder portion is described in Appendix C.

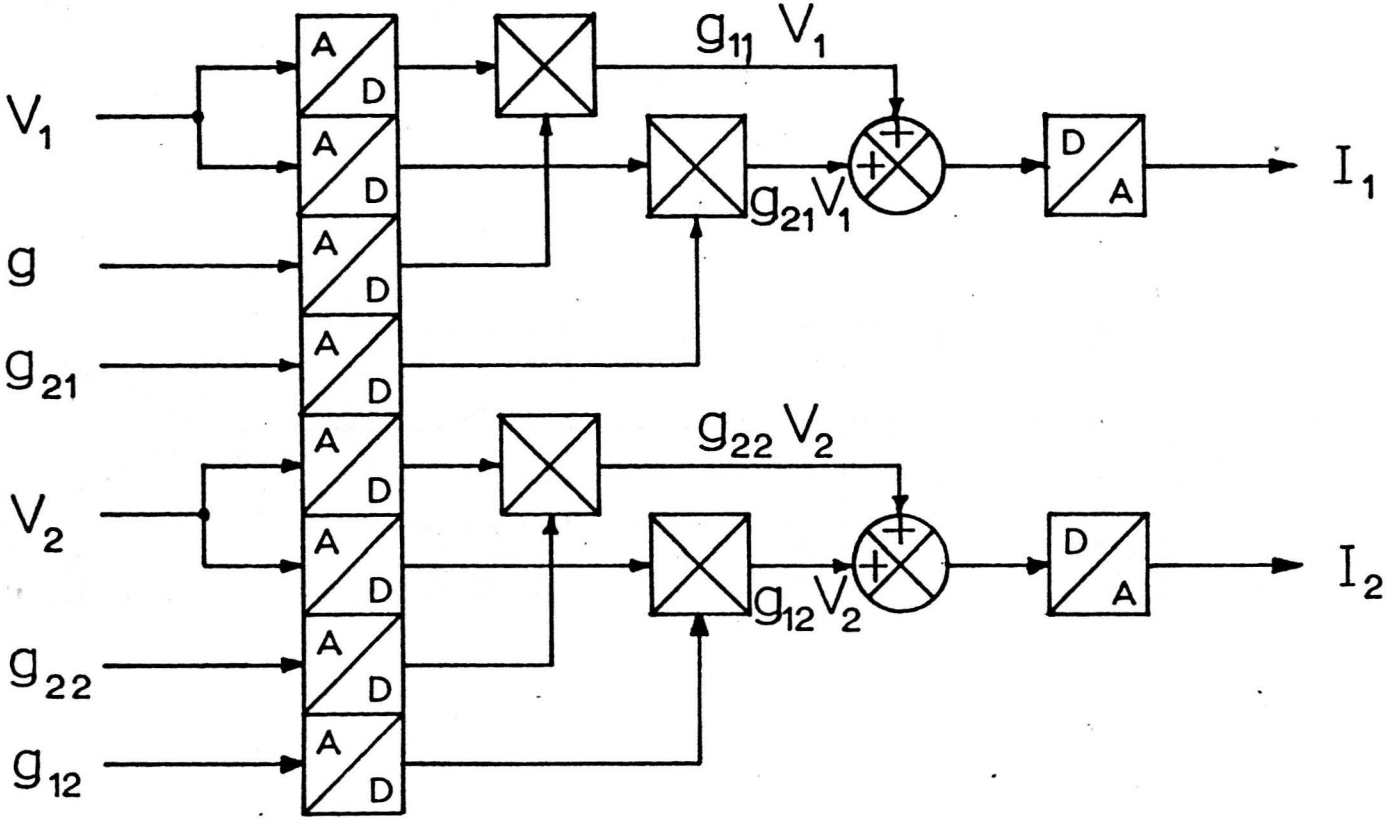


Fig. 6.3 2-port Digital Admittance Matrix

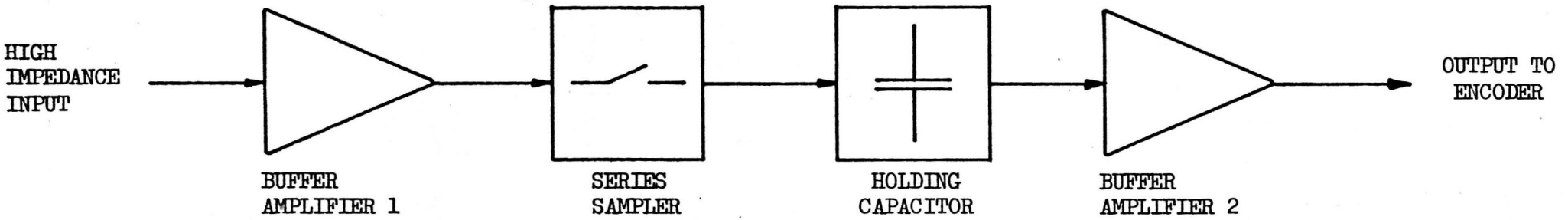


Fig. 6.4 Sampling Input Buffer

6.3.3 Multiplier

One multiplier is multiplexed to create all the products needed for the matrix simulation. The multiplier uses the shift-and-add algorithm to find the product of the magnitudes, with a separate circuit to multiply the signs.

The general multiplication algorithm is described below, and in this case m = 4, and the structure is shown in Fig. 6.5.

Let the two numbers to be multiplied be:

$$N_a = S_a \sum_{i=o}^{m-1} a_i 2^i \tag{6.1}$$

and

$$N_b = S_b \sum_{j=o}^{m-1} b_j 2^j \tag{6.2}$$

Hence the product may be written:

$$S_c \sum_{k=o}^{2(m-1)} c_k 2^k = S_a S_b \sum_{i=o}^{m-1} a_i 2^i \sum_{j=o}^{m-1} b_j 2^j$$

$$= S_a S_b \sum_{i=o}^{m-1} \left[a_i 2^i \sum_{j=o}^{m-1} b_j 2^j \right] \tag{6.3}$$

$$= S_a S_b \left[a_o \sum_{j=o}^{m-1} b_j 2^j + a_1 \sum_{j=o}^{m-1} b_j 2^j . 2 \right.$$

$$+ a_2 \sum_{j=o}^{m-1} b_j 2^j . 4$$

$$+ \ldots\ldots\ldots\ldots$$

$$\left. + a_{m-1} \sum_{j=o}^{m-1} b_j 2^j 2^{m-1} \right]$$

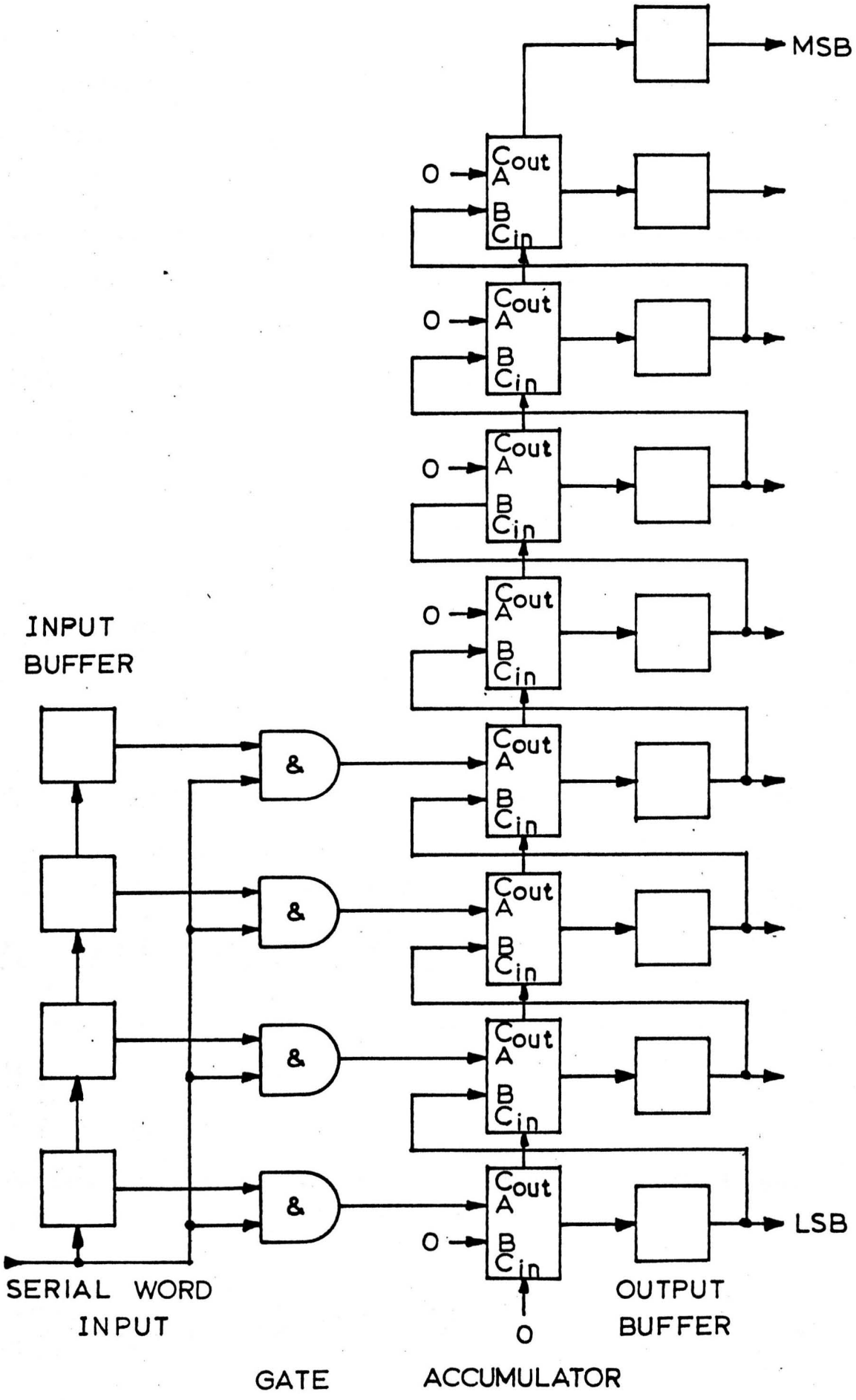


Fig. 6.5 Shift-and-Add Multiplier

Thus the shift-and-add algorithm sequentially creates each term in the square brackets and adds it to the accumulated total. This process takes m iterations to complete and creates a word 2m bits long.

6.3.4 Adder

The adder is multiplexed to add two results from the multiplexed multiplier to create a 10-bit word represented by a sign and 9 bits magnitude ready for feeding to the output D/A conyerter. The adder algorithm shown below works by converting each input to an all positive word, adding the inputs together, then subtracting twice the offset.

Let the two numbers to be added be N_f and N_g where :

$$N_h = N_g + N_g \tag{6.4}$$

and :

$$N_f = S_f \sum_{i=o}^{n-1} f_i 2^i \tag{6.5}$$

$$N_g = S_g \sum_{i=o}^{n-1} g_i 2^i \tag{6.6}$$

Add a constant offset to N_f and N_g of K. Therefore:

$$N_h = (N_f + K) + (N_g + K) - 2K$$

$$= Q_f + Q_g - 2K \tag{6.7}$$

and let:

$$Q_h = Q_f + Q_g$$

The manipulations which follow are all based on the small theorem:

THEOREM

$$\sum_{i=o}^{p-1} a_i 2^i + \sum_{i=o}^{p-1} \overline{a_i} 2^i = 2^p - 1 \tag{6.8}$$

where $\overline{a_i} = 1 - a_i$. This is the 1's complement of a_i.

PROOF

The LHS may be rewritten :

$$(a_i + \overline{a_i})\ 2^i = \sum_{i=o}^{p-1} 2^i$$

$$= 2^p - 1 \qquad (6.9)$$

Add an offset K to N_f and N_g:

$$Q_f = S_f \sum_{i=o}^{n-1} f_i 2^i + K \qquad (6.10)$$

$$Q_g = S_g \sum_{i=o}^{n-1} g_i 2^i + K \qquad (6.11)$$

and define:

$$Q_f \Big|_{\min} = Q_g \Big|_{\min} = 0 .$$

Therefore :

$$K = -(-1) \sum_{i=o}^{n-1} 2^i$$

$$= 2^n - 1 \qquad (6.12)$$

Now consider Q_f (from (6.10) and (6.12)):

$$Q_f = S_f \sum_{i=o}^{n-1} f_i 2^i + \sum_{i=o}^{n-1} 2^i \qquad (6.13)$$

If $S_f = +1$,

$$Q_f = \sum_{i=o}^{n-1} (1 + f_i)\, 2^i \qquad (6.14)$$

and when $S_f = -1$,

$$Q_f = \sum_{i=o}^{n-1} (1 + f_i)\, 2^i = \sum_{i=o}^{n-1} \overline{f_i} 2^i \qquad (6.15)$$

A realisation of this algorithm is shown in Fig. 6.6. Exactly the same algorithm must be applied to Q_g, and in this machine the circuitry to offset Q_f and Q_g was multiplexed.

Now Q_f and Q_g may be added. The maximum value of the sum Q_h will be:

$$\begin{aligned} Q_h \Big|_{max} &= 4 \sum_{i=o}^{n-1} 2^i \\ &= 2^{n+2} - 4 \\ &< \sum_{i=o}^{n-1} 2^i \end{aligned} \qquad (6.16)$$

Now:

$$\begin{aligned} Q_h &= S_f \sum_{i=o}^{n-1} f_i 2^i + S_g \sum_{i=o}^{n-1} g_i 2^i + 2(2^n - 1) \\ &= \sum_{i=o}^{n+1} m_i 2^i \end{aligned} \qquad (6.17)$$

$$\text{as } Q_h \Big|_{max} < \sum_{i=o}^{n+1} 2^i$$

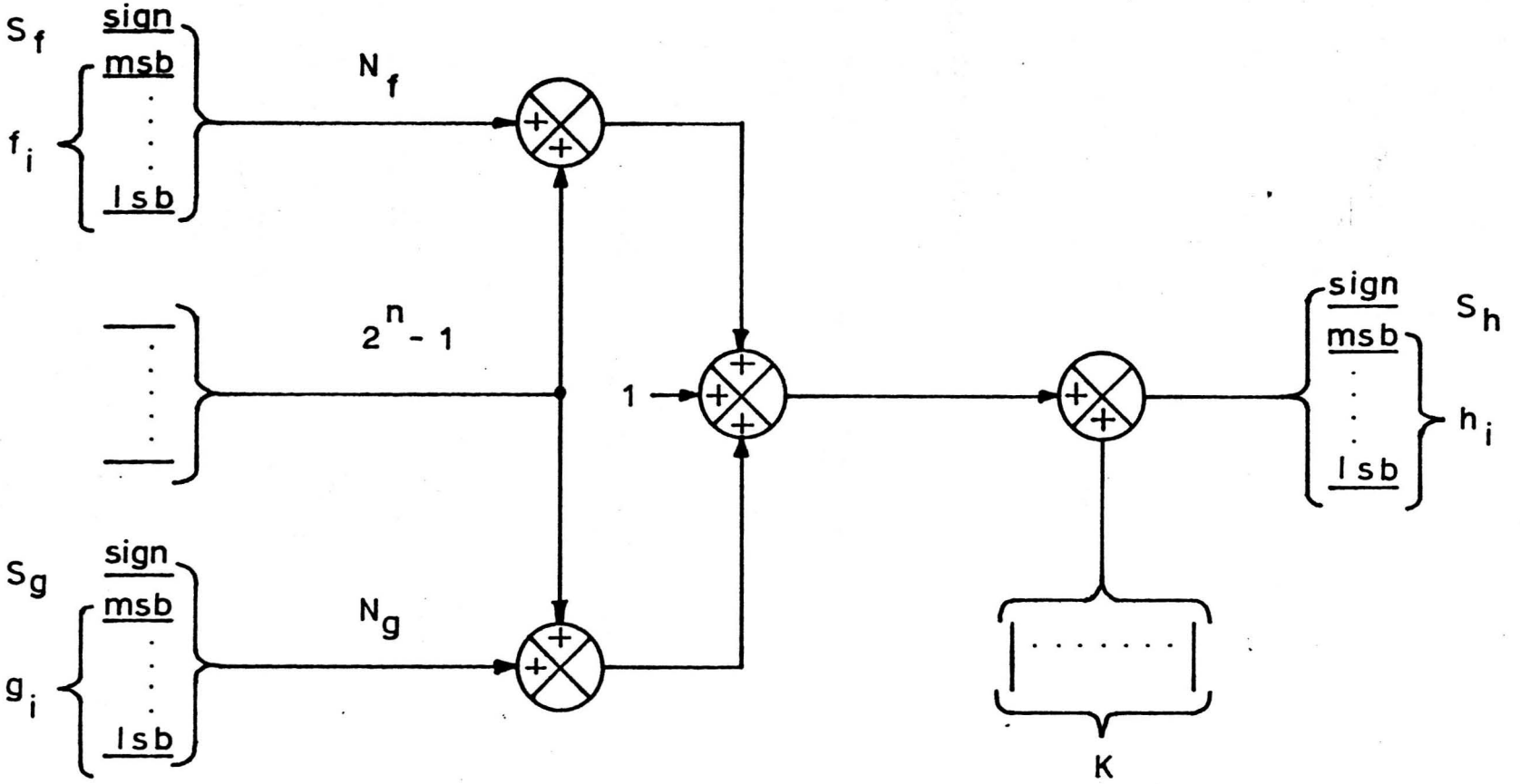


Fig. 6.6 Structure of Adder

Therefore:

$$N_h = \sum_{i=o}^{n+1} m_i 2^i - 2(2^n - 1)$$

This may be rearranged thus:

$$N_h = \left(1 + \sum_{i=o}^{n+1} m_i 2^i\right) - \left(2^{n+1} - 1\right)$$

$$= \sum_{i=o}^{n+1} x_i 2^i - \left(2^{n+1} - 1\right) \qquad (6.18)$$

and again from equation (6.16):

$$1 + \sum_{i=o}^{n+1} m_i 2^i < \sum_{i=o}^{n+1} 2^i$$

Hence:

$$N_h = x_{n+1} 2^{n+1} + \sum_{i=o}^{n} x_i 2^i - (2^{n+1} - 1)$$

When :

$x_{n+1} = -1$, $S_h = +1$ and:

$$N_h = 1 + \sum_{i=o}^{n} x_i 2^i \qquad (6.19)$$

When :

$x_{n+1} = 0$, $S_h = -1$ and:

$$N_h = \sum_{i=o}^{n} x_i 2^i \qquad (6.20)$$

This may be implemented as shown in Fig. 6.6.

6.3.5 D/A Converter Current Generator

A block diagram of the D/A converter current generator is shown in Fig. 6.7. The buffer register is necessary to hold the output current constant between computations.

The basic circuit of a bidirectional current generating D/A converter is shown in Fig. 6.8. By analogy from Appendix F equation (F.5), I_1 may be written as:

$$I_1 = \frac{V_{ref}}{R_o} \tag{6.21}$$

However from the resistive shunt ladder shown in Fig. 6.9 :

$$\frac{1}{R_o} = \frac{1}{R} \sum_{i=o}^{n} a_i 2^i$$

and thus equation (6.21) may be rewritten:

$$I_1 = \frac{V_{ref}}{R} \sum_{i=o}^{n} a_i 2^i \tag{6.22}$$

Thus I_1 is controlled by the state of the binary word representing the magnitude of the output current. Now in Fig. 6.8 the sign bit (S_h) is used to control the direction of the output current I_o. Thus the output current will be:

$$I_o = S_h \cdot \frac{V_{ref}}{R} \cdot \frac{R_1}{R_2} \cdot \sum_{i=o}^{n} a_i 2^i \tag{6.23}$$

Therefore the absolute maximum value of I_o is:

$$I_c \Big|_{max} = \frac{V_{ref}}{R} \cdot \frac{R_1}{R_2} \cdot (2^n - 1) \tag{6.24}$$

Table 6.4 shows the component values, parameters and maximum output

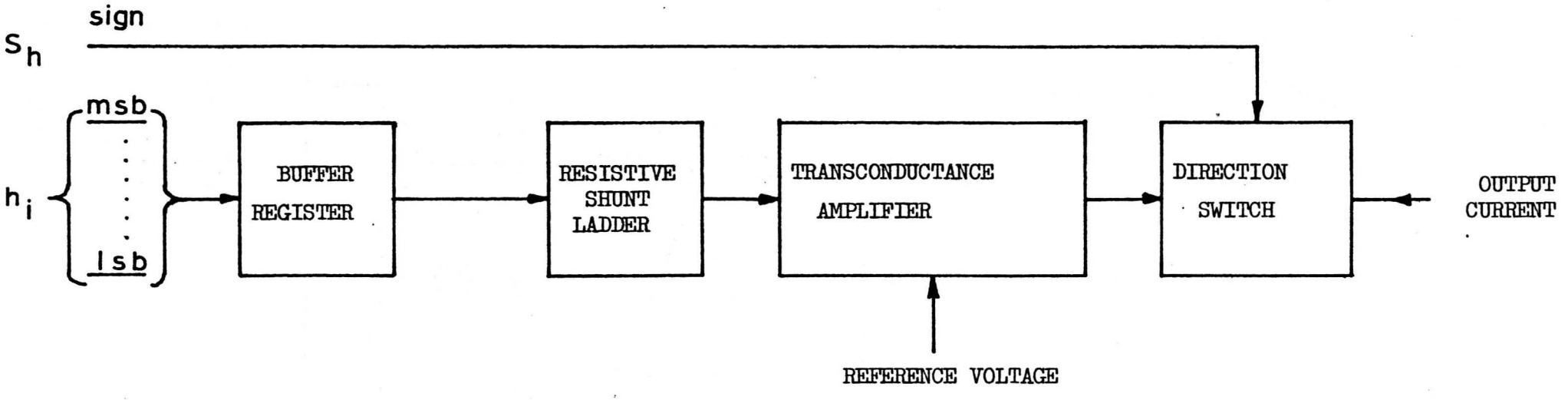


FIG. 6. 7 D/A Decoder and Current Generator

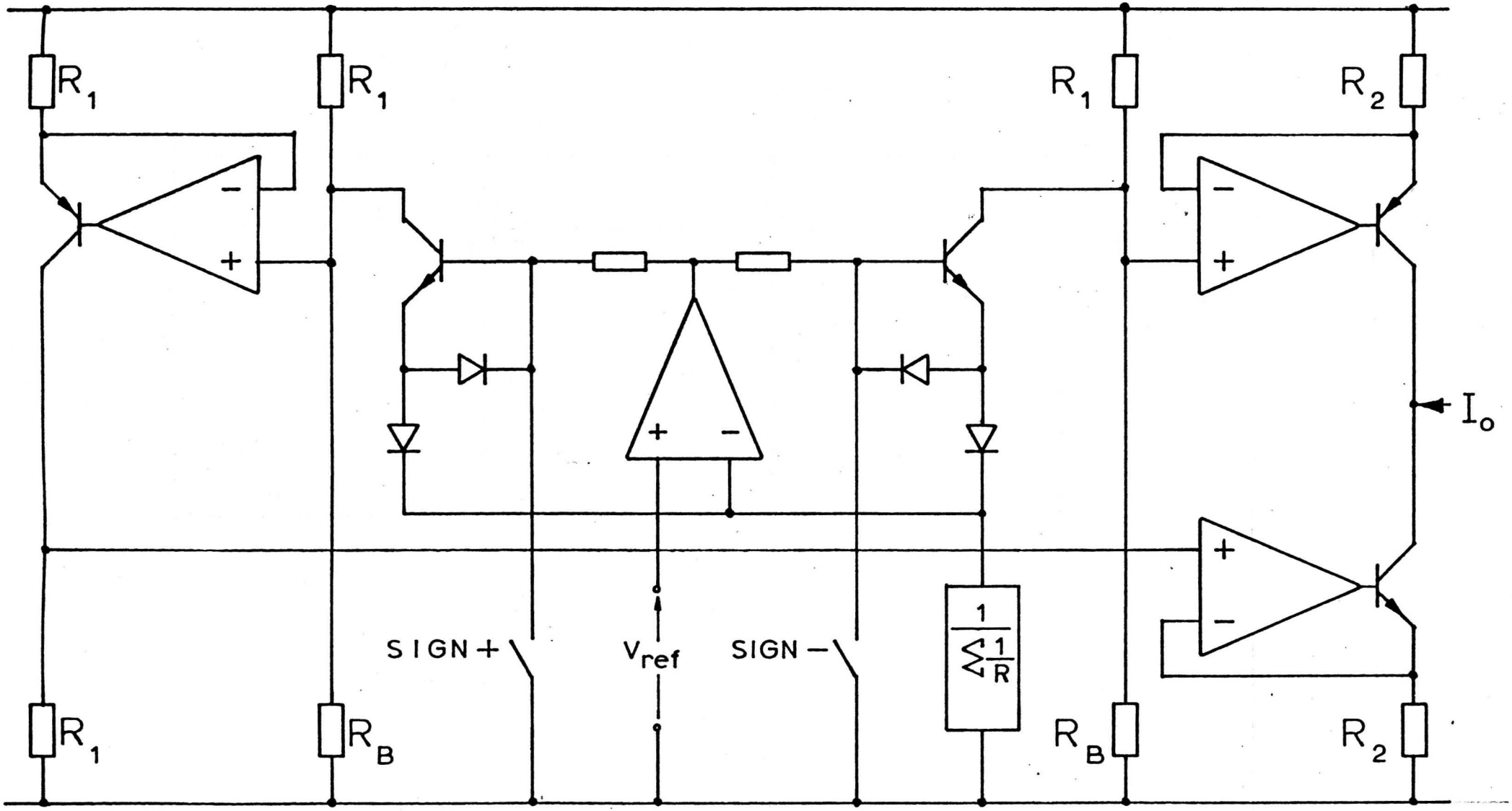


Fig. 6.8 Bidirectional Current generating D/A Converter

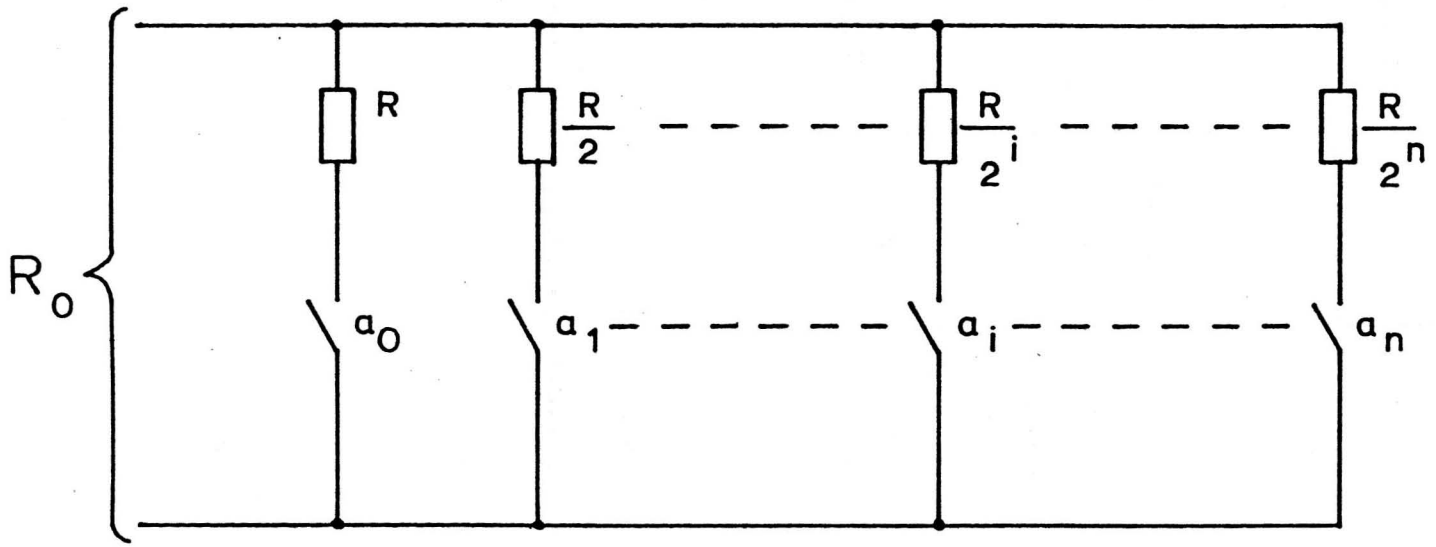


Fig. 6.9 Resistive Shunt Ladder

VARIABLE	VALUE	UNITS	COMMENTS
V_{ref}	5	VOLTS	Maximum Reference Voltage
R	64000	OHMS	
R_1	75	OHMS	
R_2	7.5	OHMS	
n	9	-	Number of bits
$I_o \Big\vert_{MAX}$	0.399	AMPS	Maximum output current

TABLE 6.4

Output Transconductance Amplifier Parameters

current.

The sign of the output current is controlled by grounding the line corresponding to the other sign. The circuitry on boards 12 and 18 (Circuits 6.9 and 6.11) is arranged so that the upper and lower output stages of the current generator are on simultaneously for the minimum amount of time. The only effect of simultaneous operation of the upper and lower output stages is to cause the output current to cease for a very short time because these output stages are protected by having their output current controlled.

The output transconductance amplifier is biased into class AB to overcome crossover distortion caused when the sign bit changes state. The resistors R_B in Fig. 6.8 set the bias current in the output stage to :

$$I_{BIAS} = 2V_{cc} \cdot \frac{1}{R_2} \cdot \frac{R_1}{R_1 + R_B} \qquad (6.25)$$

Table 6.5 shows the component values used in the experimental machine and hence the bias current.

The settling time of the current generators was controlled by the type of operational amplifiers and transistors chosen. The slew rate (0.5V/ μS) of the operational amplifier (SN72741N) was found to be adequate. However the transition frequency (f_T) of the power transistors (TR8 and TR10 in Circuit 6.20) which were chosen first was only 3 MHz and this was found to cause severe settling time degradation. These transistors were types TIP 29 and TIP 30 and they were therefore replaced by types BD 131 and BD 132 which have 60 MHz transition frequencies [22], [23] thus curing the problem. All the other transistors involved were small signal types having transition frequencies in excess of 100 MHz.

VARIABLE	VALUE	UNITS	COMMENTS
V_{cc}	10	Volts	Power Supply Voltage
R_1	75	OHMS	
R_2	7.5	OHMS	
R_B	20000	OHMS	
I_{BIAS}	9.96	Milliamps	Bias Current

TABLE 6.5

Output Transconductance Amplifier Bias Current

6.4 MACHINE CONSTRUCTION

The whole experimental machine was constructed in a type 4 U 30-slot Vero rack designed to accommodate boards 203 mm x 159 mm with 60 pin, 0.1 inch pitch, single-sided edge connectors. An additional rack was constructed and attached to the top of the Vero rack to hold all the input and output sockets, and provision for reversing the sign of of the data encoded from each channel. As many signals as possible were bus-wired across the edge-connectors at the back of the Vero-rack. All the digital logic was implemented with 74 series Transistor-Transistor-Logic (TTL) [24] .

Table 6.6 shows the power supplies used by the machine.

The board numbers used represent the actual socket numbers in the card frame. Some of these sockets positions were deliberately not used either to allow for the addition of new cards or because the present cards occupied more width than one socket spacing allowed.

6.5 BOARDS FOR DIGITAL ADMITTANCE MATRIX

The boards necessary to organise and operate the machine as a 2-port digital admittance matrix are described here, including the clock and control boards.

6.5.1 Master Clock Board

The master clock boards (Board 1 and Fig. 6.10) contains the master oscillator, the sampling rate selector and the circuitry to generate the control signals.

The master oscillator is set to run at 6.4 MHz and is crystal controlled. The two digital inverters in IC1 are biased into the linear mode by the feedback through TR_1 (P346A). The output is taken from between the two inverters and is buffered before feeding the 4 stage synchronous binary counter (IC2). By selecting the clock signal or any one of the outputs from the four stages the basic operating frequency

VOLTAGE VOLTS	CURRENT AMPS
+ 15	0.5
+ 10	1.0
+ 5	2.5
- 6	0.1
- 10	1.0
- 15	0.5

TABLE 6.6

Power Supplies

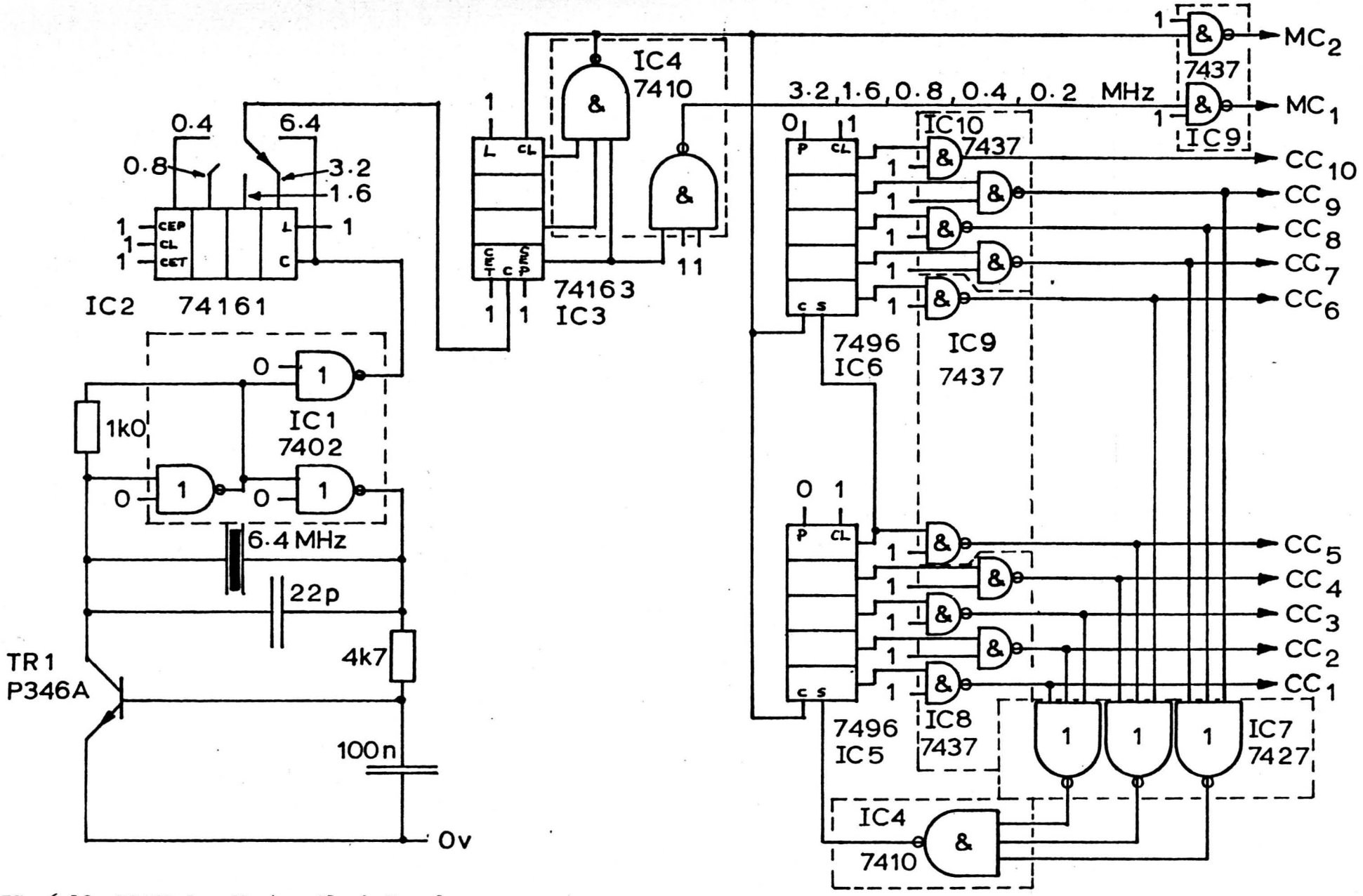


FIG. 6.10 BOARD 1 Master Clock Board

of the whole machine could be set, thus controlling the sample rate. The selected output feeds the master control counter.

The master control counter (IC3) is a fully synchronous 4 stage counter arranged to divide the selected input clock rate by 2 to make MC_1 and by 12 to make MC_2. MC_2 is then used to generate the channel control signals.

The channel control signals CC_1 to CC_{10} are generated by a 10-stage shift register (IC5 and IC6) which is clocked by MC_2 and thus the maximum frequency for CC_i is 53.3 kHz, the maximum sampling rate.

MC_1, MC_2 and CC_1 to CC_{10} are all buffered onto the busses connected across the back of the machine by IC_8, IC_9 and IC_{10}.

Part of IC_4 and IC_7 are used to maintain the 10-stage shift register propagating a solitary zero.

MC_1, MC_2 and CC_1 to CC_{10} are shown in Fig. 6.11. MC_1 and MC_2 are shown in Photo 6.1 and MC_2 and CC_1 in Photo 6.2.

6.5.2 Comparator Board (4)

The comparator board (Board 4 and Fig. 6.12) is part of the multiplexed A/D converter and contains 10 comparators, one for each analogue input. One comparator at a time is selected using the channel control signals, the comparator outputs being internally OR'ed in pairs and buffered by transistors TR1 to TR5, then collectively OR'ed.

The transistors are necessary to remove the loading effects of the inverters (IC6) on the comparator outputs (IC1 to IC5).

6.5.3 Encoder Board (5)

The encoder board (Board 5 and Fig. 6.13) contains the 5-bit test and storage registers and the encoder's D/A converter. The test register propagates a single high state and is kept in this mode by the feedback to its serial input. The input of the storage register

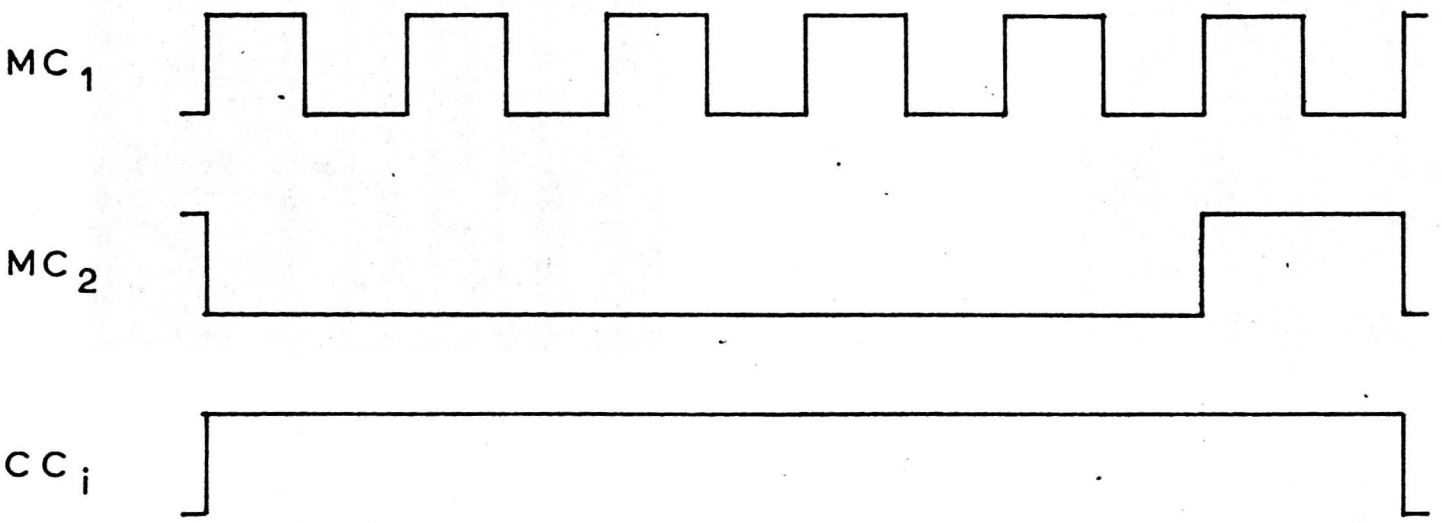


FIG. 6.11 Clock Signals

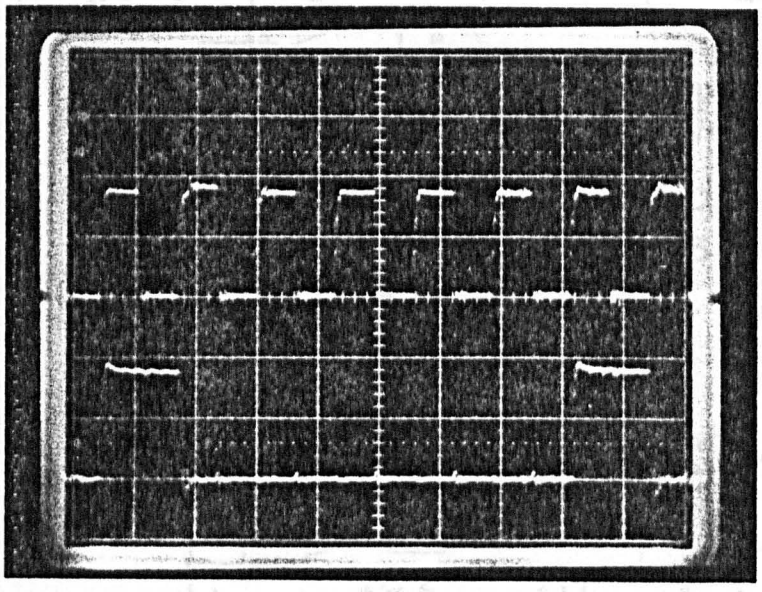


PHOTO 6.1

X 200ns / div

Y_1 2V / div

Y_2 2V / div

Master Clock Signals

Y_1 - MC 1

Y_2 - MC 2

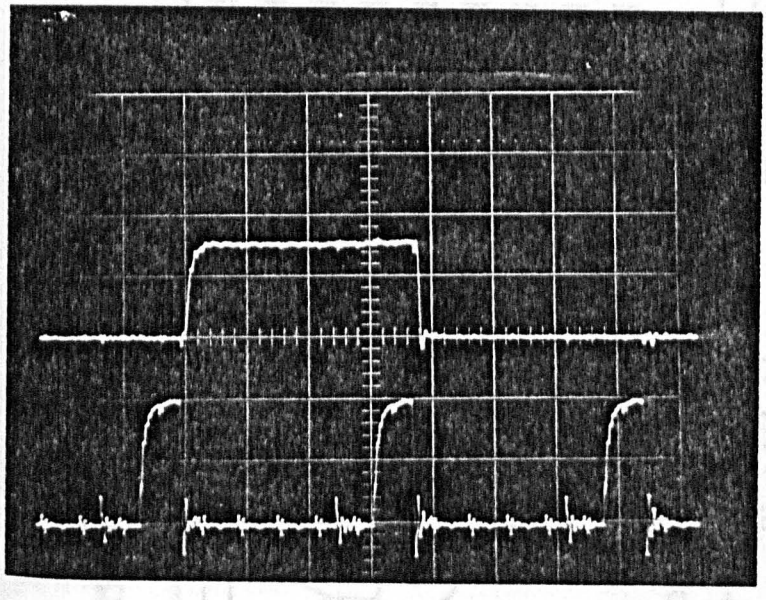


PHOTO 6.2

X 500 ns / div

Y_1 2V / div

Y_2 2V / div

Master Clock and Channel Control Signals

Y_1 - CC_1

Y_2 - MC_2

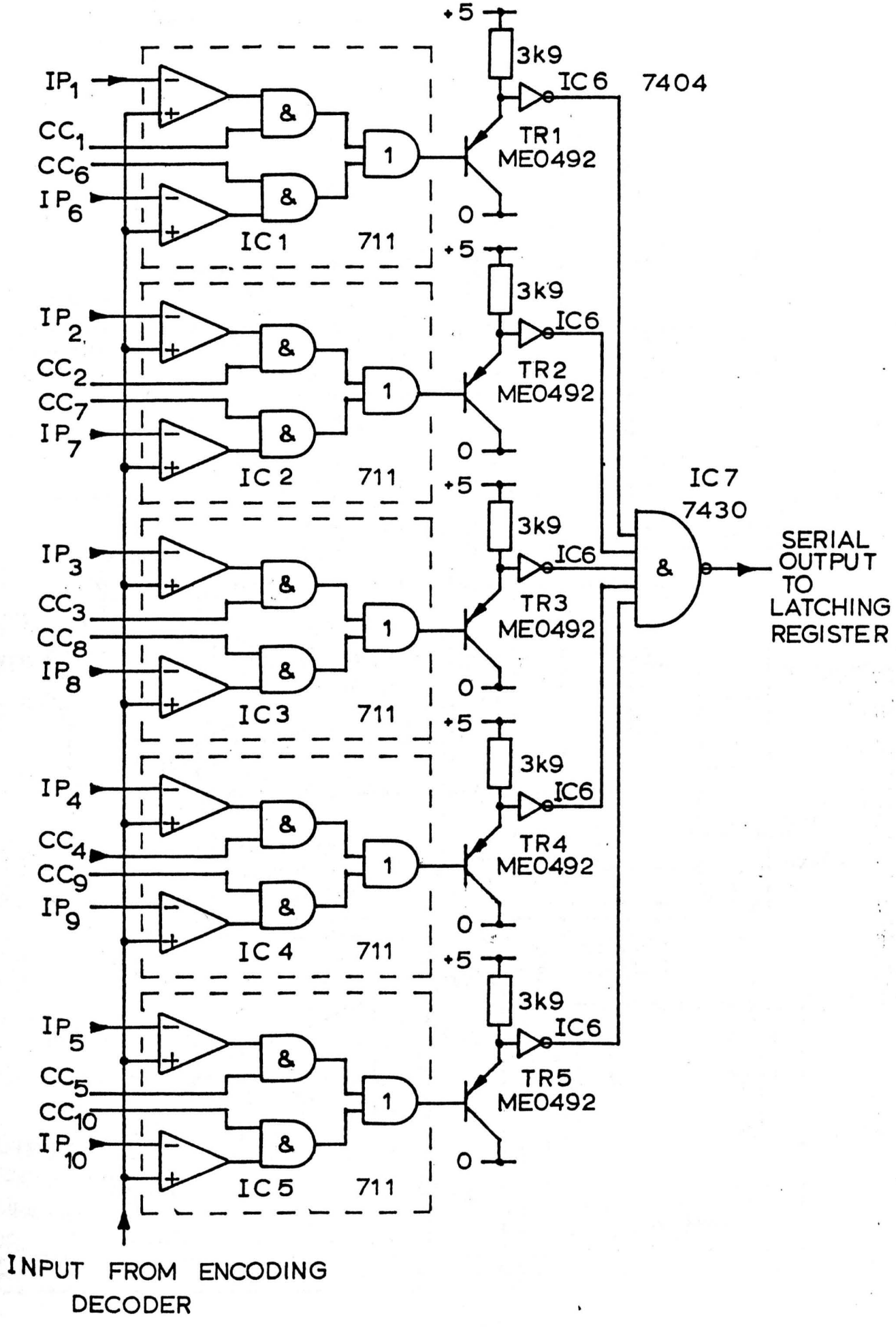


FIG. 6.12 BOARD 4 Comparator Board

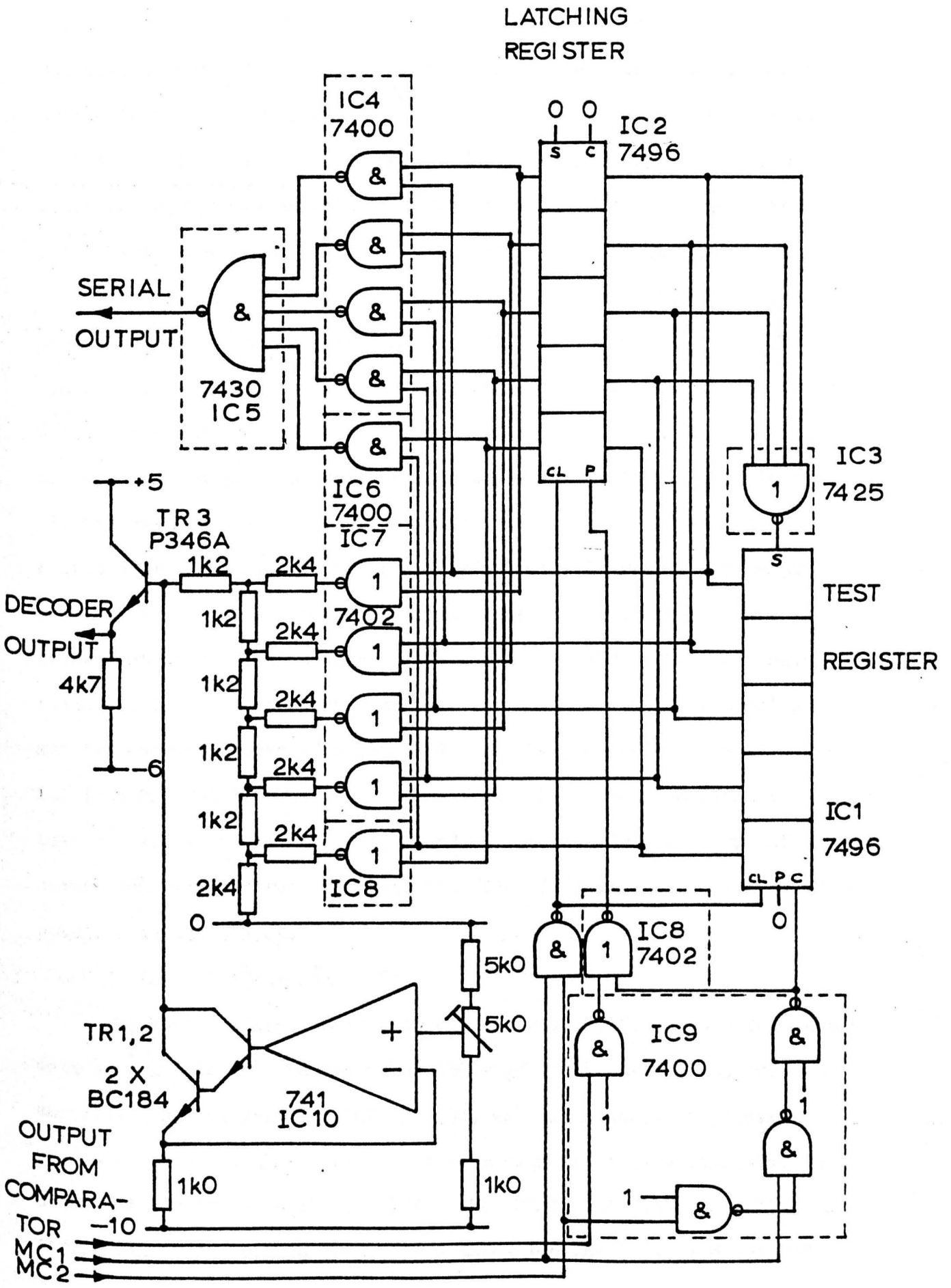


FIG. 6.13 BOARD 5 Encoder Board

is indexed by the test register so that if a decision to store a high state has been made by the comparator (Photo 6.3) then the correct bit will be set. This output of the storage register is also indexed by the test register using 5 NAND gates and the result OR'ed to produce the output serial bit stream. (Photo 6.4 and 6.5). This was found to be superior to using the direct signal from the comparator due to the presence of spurious pulses in the latter signal.

Each stage of the test register is also OR'ed with the corresponding stage of the storage register, the outputs driving the D/A converter. This simple D/A converter consists of an R/2R ladder (Fig. 6.14) and a current source to remove any D.C. offset in the converter output. The output impedance of an R/2R ladder is 2R provided that all the switches have negligible series resistance. Hence a current source sinking current from the ladder output will reduce any output voltage by I_b. 2R, independent of the state of the switches. The ladder output voltage is then buffered with an emitter follower which feeds the 10 comparators on board 4. Photo 6.6 shows CC1 (uppertrace) and the output of this R/2R ladder during the conversion of a sinewave fed into Channel 1. The lower trace shows the successive decision levels clearly, namely 1, 2, 4, 8 and 16 levels. As the output of the comparator will be true or false, the final 16 levels give rise to 32 possible output states.

6.5.4 Level Shifter Board (6)

The level shifter board (Board 6 and Fig. 6.15) accepts the serial data stream into IC1 and generates the sign of the product of two successive serial words in IC2 and IC4, and simultaneously converts the data format in this serial word from sign and 4 bits magnitude to 5 bits of magnitude in IC3 and IC4. (See Section 6.3.3). Finally the serial magnitude word is transmitted using IC6 and IC7. IC8 and IC9

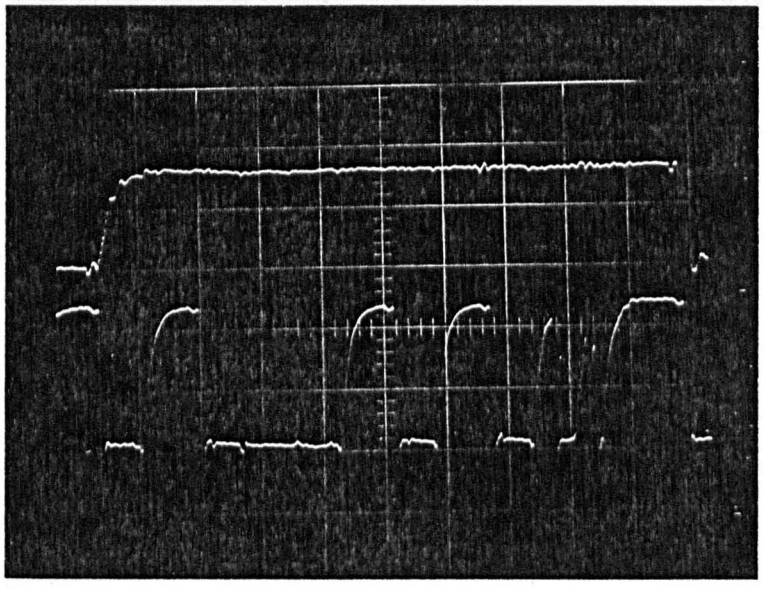


PHOTO 6.3

X - 200ns / div

Y_1 - 2V / div

Y_2 - 2V / div

Voltage comparator output

Y_1 - CC_1

Y_2 - Channel 1 comparator output when encoding level 23

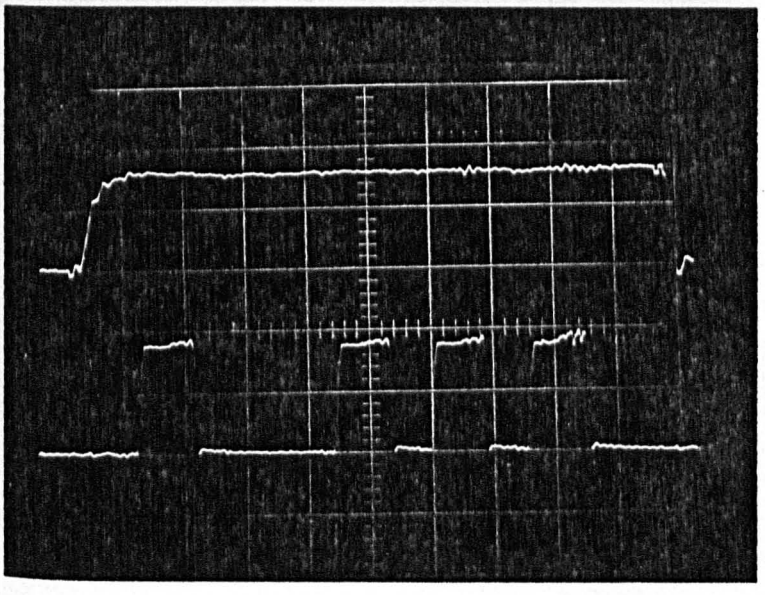


PHOTO 6.4

X - 200ns / div

Y_1 - 2V /div

Y_2 - 2V /div

Serial Transmitted Data

Y_1 - CC_1

Y_2 - Channel 1 serial output when encoding level 23

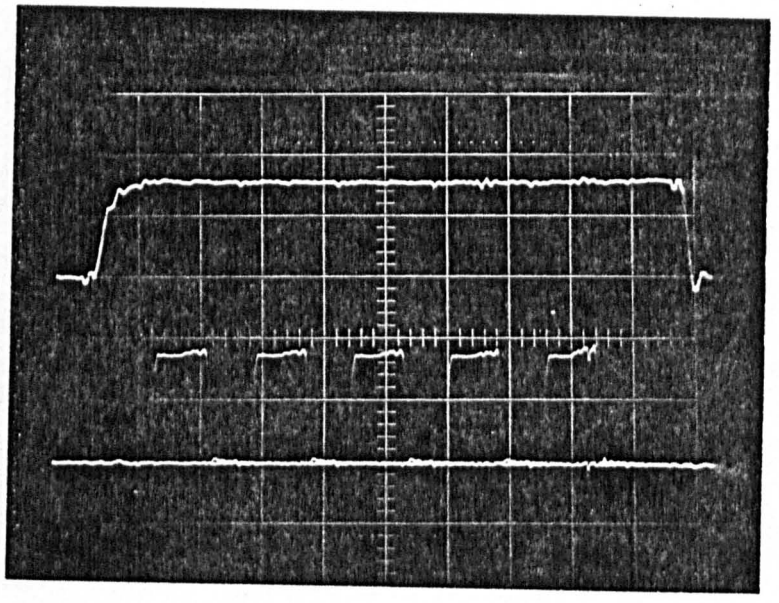


PHOTO 6.5

X - 200ns / div

Y_1 - 2V / div

Y_2 - 2V / div

Serial transmitted data when encoding a sinewave of maximum amplitude

Y_1 - CC_1

Y_2 - Serial transmitted data from a sinewave

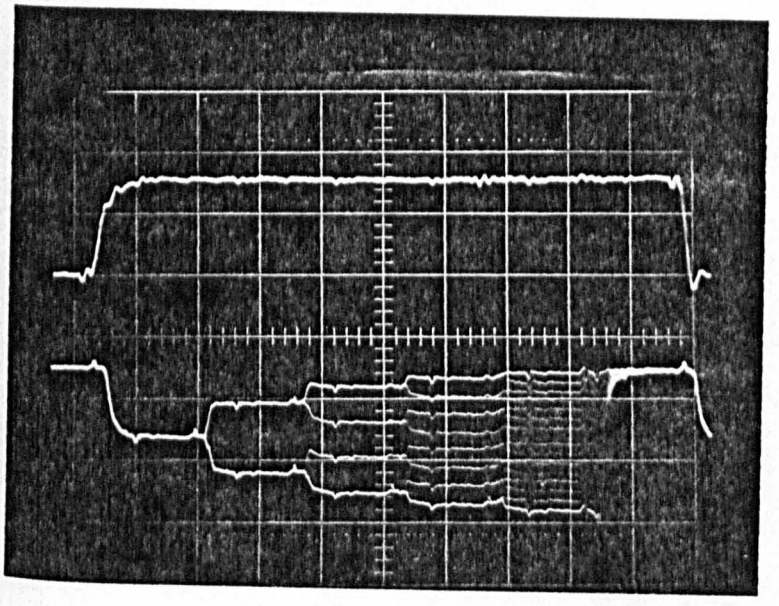


PHOTO 6.6

X - 200ns / div

Y_1 - 2V / div

Y_2 - 2V / div

Branching in A/D converter

Y_1 - CC_1

Y_2 - Output of encoder D/A converter

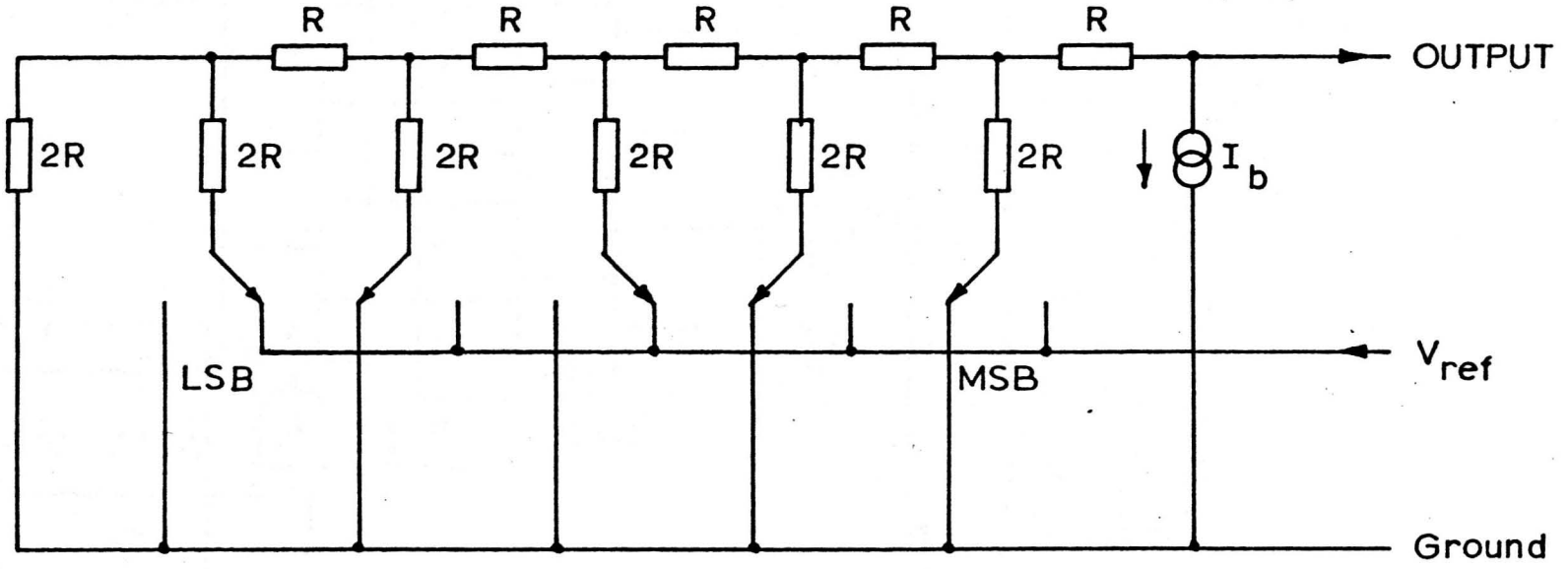


FIG. 6.14 R/2R D/A Converter

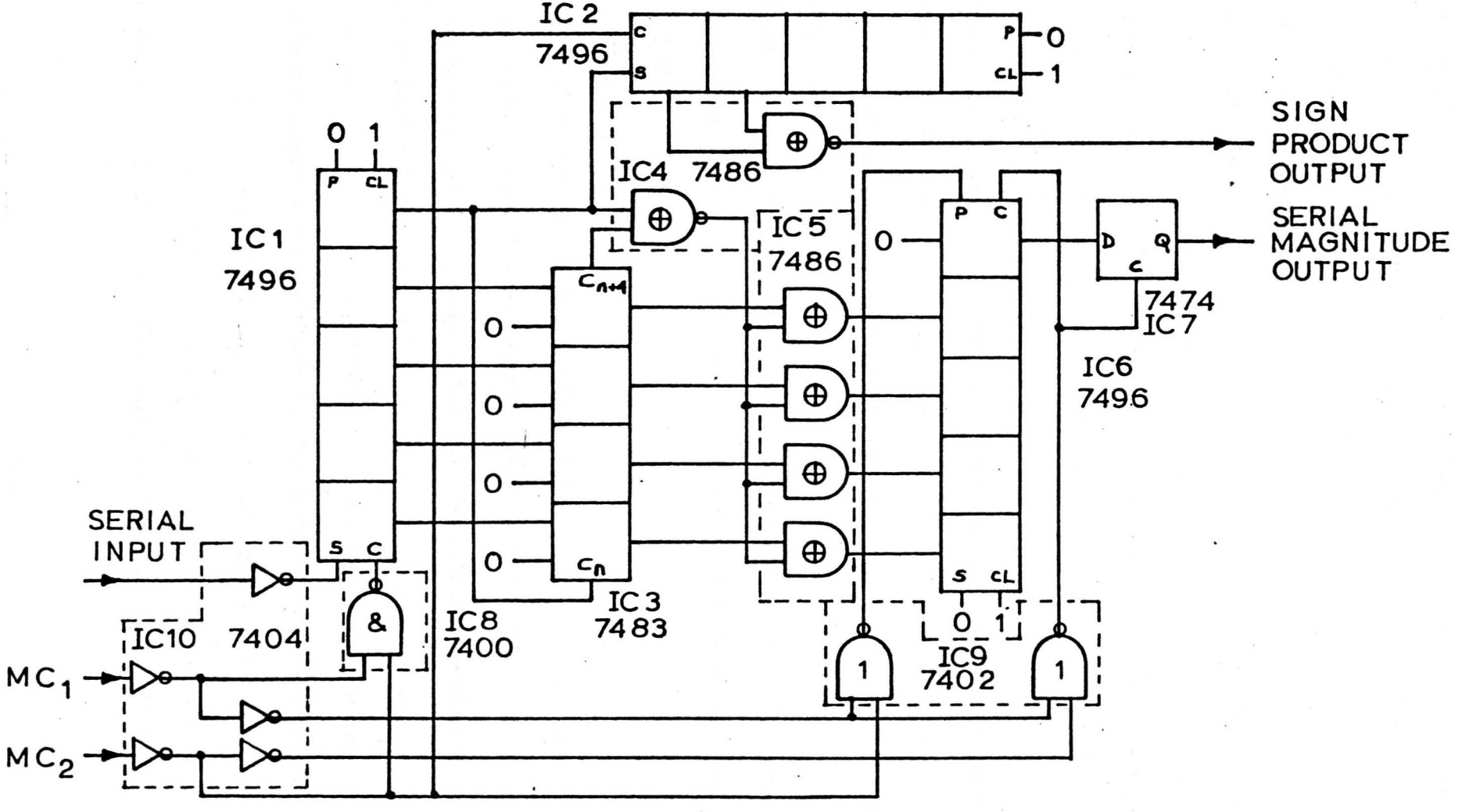


FIG. 6.15 BOARD 6 Level Shifter Board

generate the necessary on-board control signals.

6.5.5 Control Signal Generator Board (8)

The control signal generator board (Board 8 and Fig. 6.16) generates 7 control signals from the two clock signals MC1 and MC2 and the 10 channel control signals.

6.5.6 Multiplier Board (9)

The multiplier board (Board 9 and Fig. 6.17) performs sequential multiplication on 2 successive serial data words from board 6, which in turn has removed the sign bit from the serial stream. The first word is clocked into the multiplier buffer register (IC1) by LOADC. The second word is used bit by bit to gate (in IC2) the first word into the multiplier adder (IC3 and IC4). The adder output is clocked in parallel into the multiplier main register (IC5, 6, 7) and the output from every bit of the multiplier main register is connected back to the next higher input of the adder, thus handwiring the shift requirement. After 4 iterations the main register contains the product of the magnitudes. This is then converted back to sign and magnitude in IC8, 9, 10.

6.5.7 Adder Board (10a)

The adder board (Board 10a and Fig. 6.18) adds together two successive products from the multiplier sequentially. 2555 is added to each product in IC1 and 2 before the main addition is performed. The first product after shifting is loaded into the adder register (IC3, 4, 5) through the adder by ACCMC. The output of each stage of the accumulator register is fed back to the input to the equivalent stage of the adder. When the second product is ready, this is offset and then added to the first product and loaded into the adder register again by ACCMC. The adder register now contains a 10-bit word.

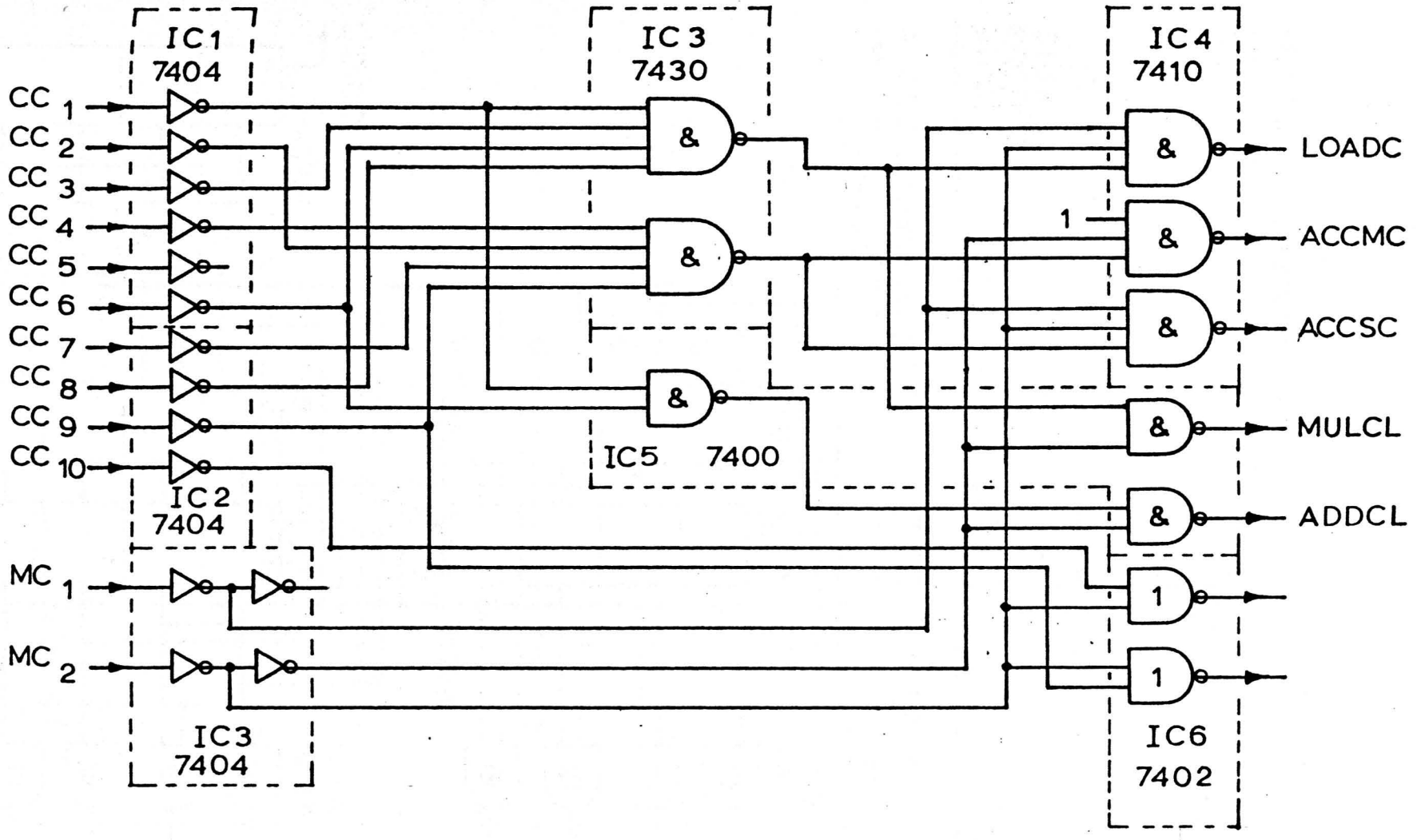


FIG. 6.16 BOARD 8 Control Signal Generator Board

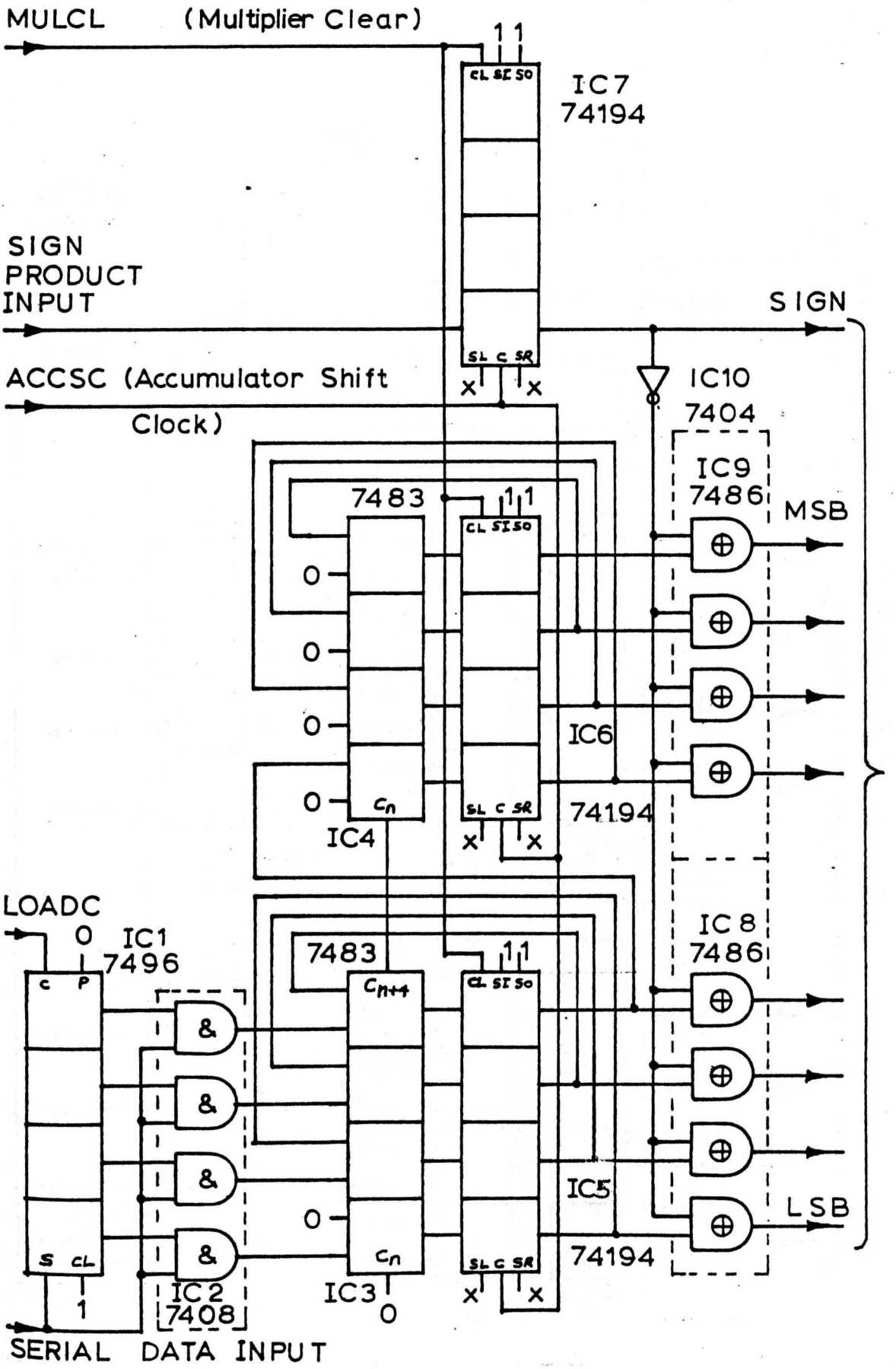


FIG. 6.17 BOARD 9 Multiplier Board

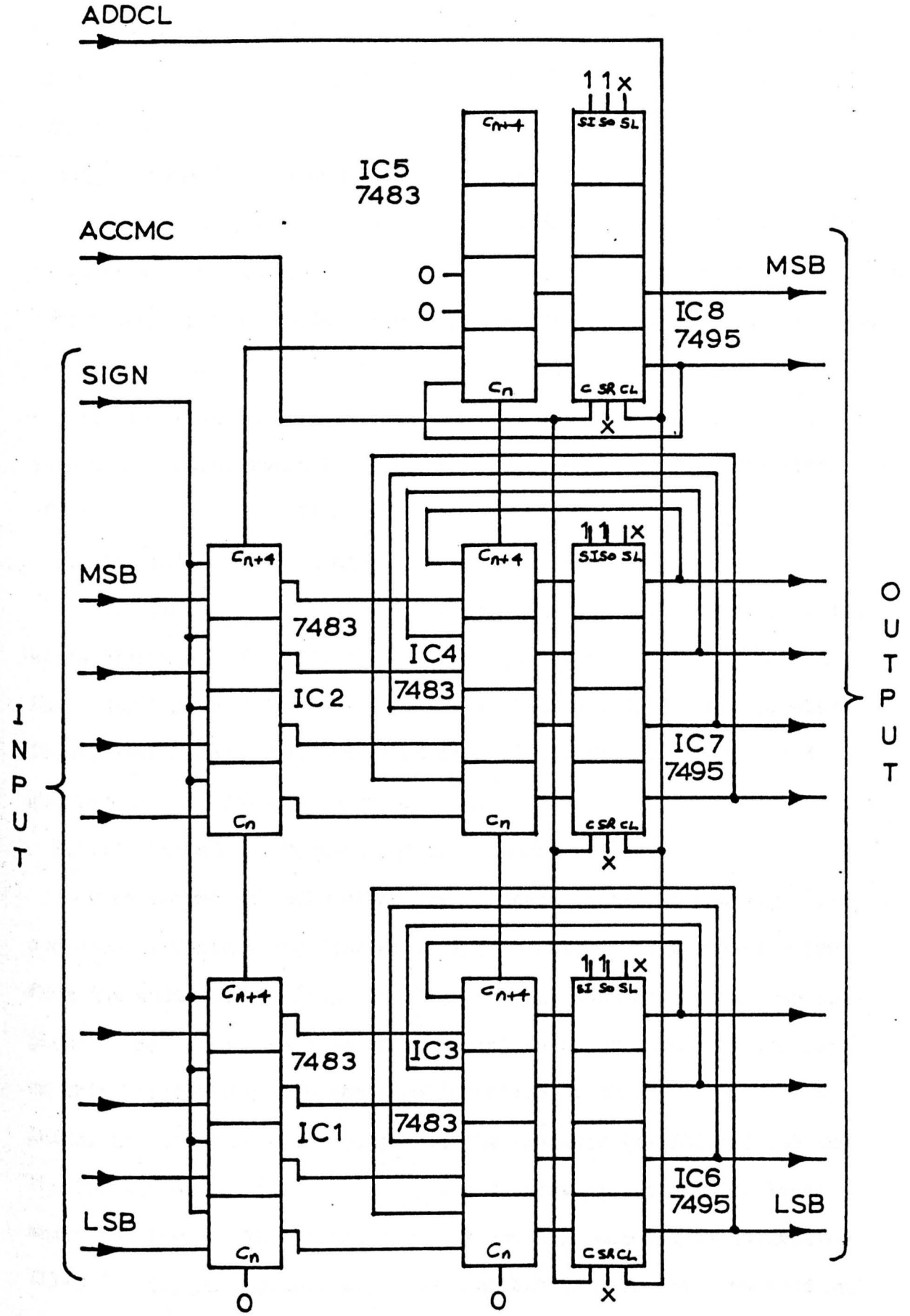


FIG. 6.18 BOARD 10a Adder Board

6.5.8 Adder Shift Board (11a)

The adder shift board (Board 11a and Fig. 6.19) accepts the 10-bit all magnitude word from the adder board (10a) and subtracts 2046 from the word to generate an output word of a sign bit and 9-bits of magnitude.

6.5.9 First D/A Current Generator Board (12)

The first D/A current generator board (Board 12 and Fig. 6.20) performs the D/A conversion and current generation. The binary weighted resistive shunt ladder is controlled from the 10-bit register on board 14 and determines the magnitude of the current generated, along with the reference voltage V_{X1}. (See Appendix F). The sign bit is used to select the output stage to be controlled, thereby setting the direction of the output current flow.

6.5.10 Buffer Store Board (14a)

The buffer store board (Board 14a and Fig. 6.21) contains a 10-stage clocked latch (IC1, 2, 3) to accept the parallel output from the adder boards (10a and 11a) and hold this steady for one complete frame, thus holding the output current steady for one frame. IC4 generates the necessary control signals.

6.5.11 Second D/A Current Generator Board (16)

The second D/A current generator board (Board 16 and Fig. 6.22) contains a 10-stage register (IC1, 2,3) to store the parallel output from the adder boards (10a and 11a) and hold this steady for one complete frame. The binary weighted resistive shunt ladder is included on this board along with the sign selection circuitry. This shunt ladder is driven from the output of the register via IC4 and IC5 and the PNP transistor (2N3702) (TR13-22) discrete inverting and level shifting stages. The stages of the shunt are selected by saturating TR1-9 (BFY51) as appropriate. The sign bit is also level shifted and

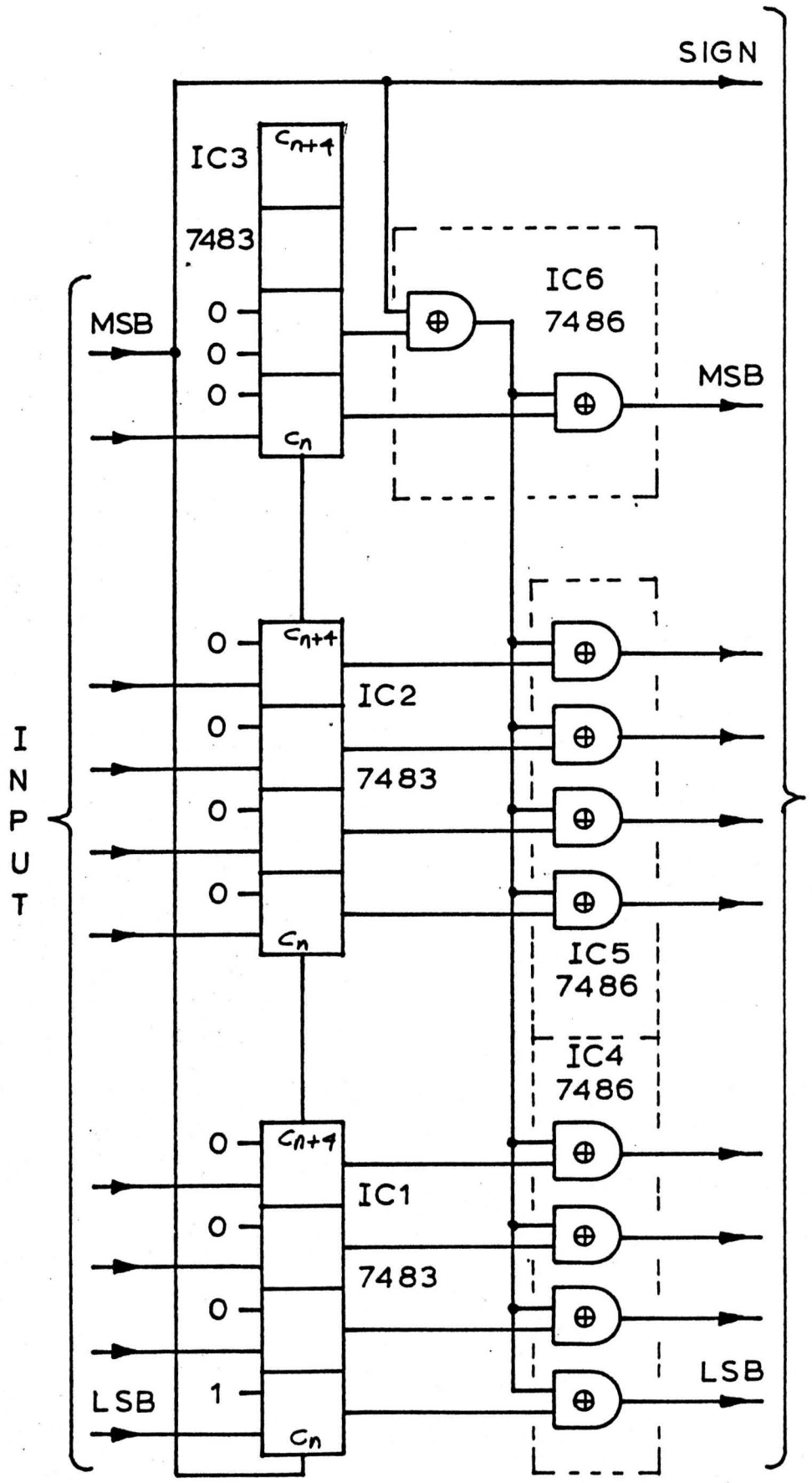


FIG. 6.19 BOARD 11a Adder Shift Board

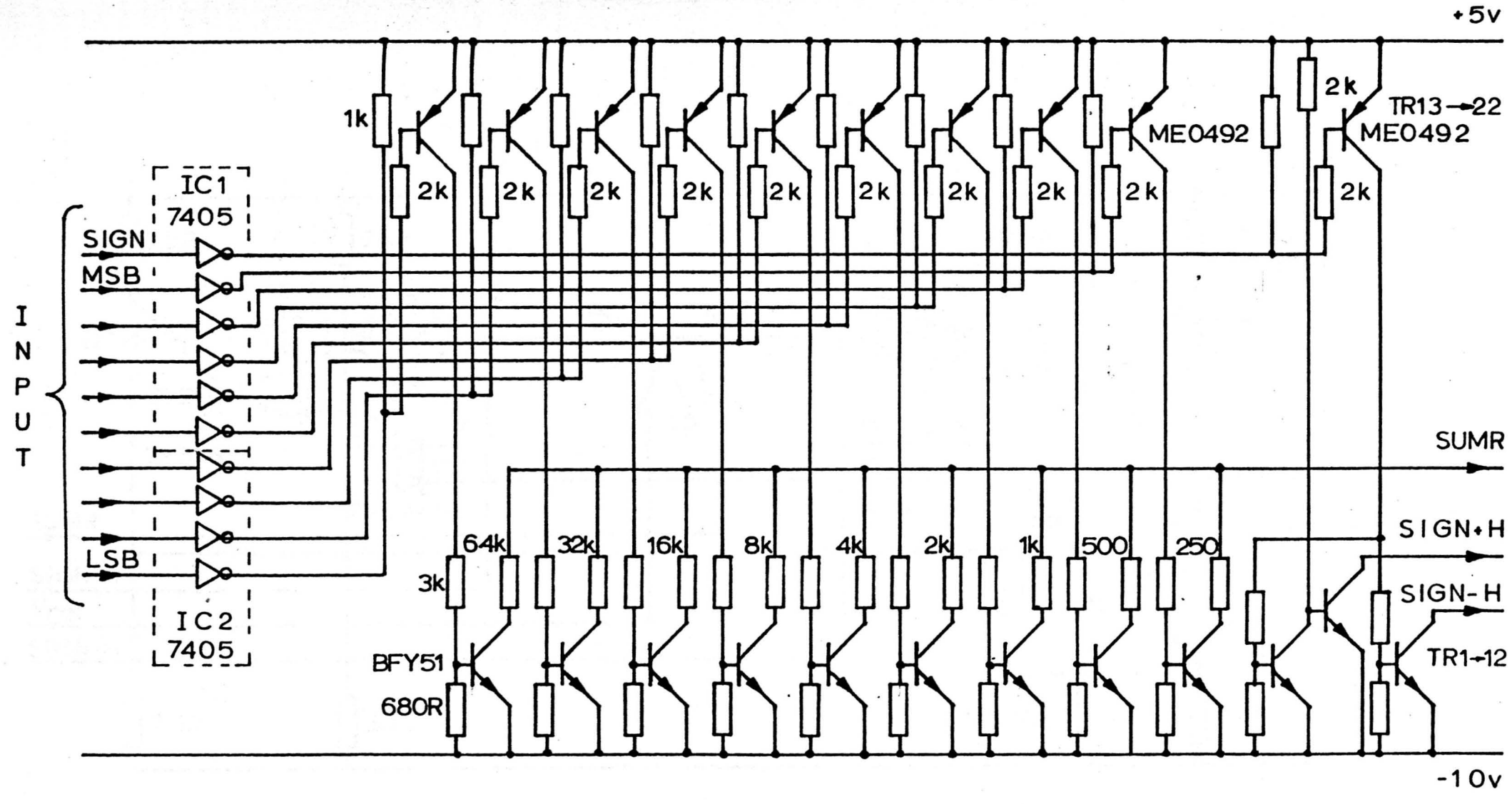


FIG. 6.20a BOARD 12 Output Current Decoder and Generator 1

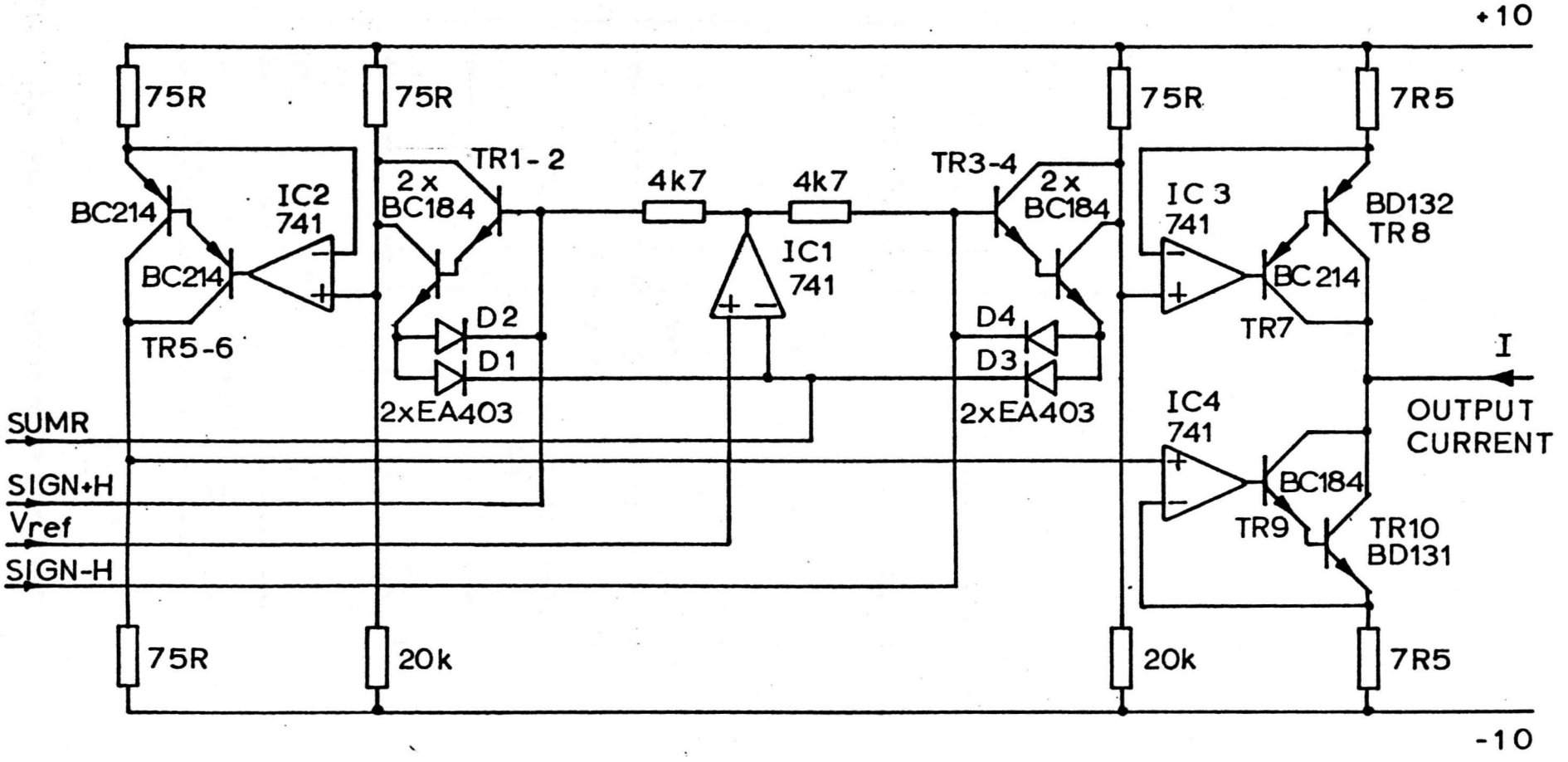


FIG. 6.20b BOARD 12 Output Current Decoder and Generator 1

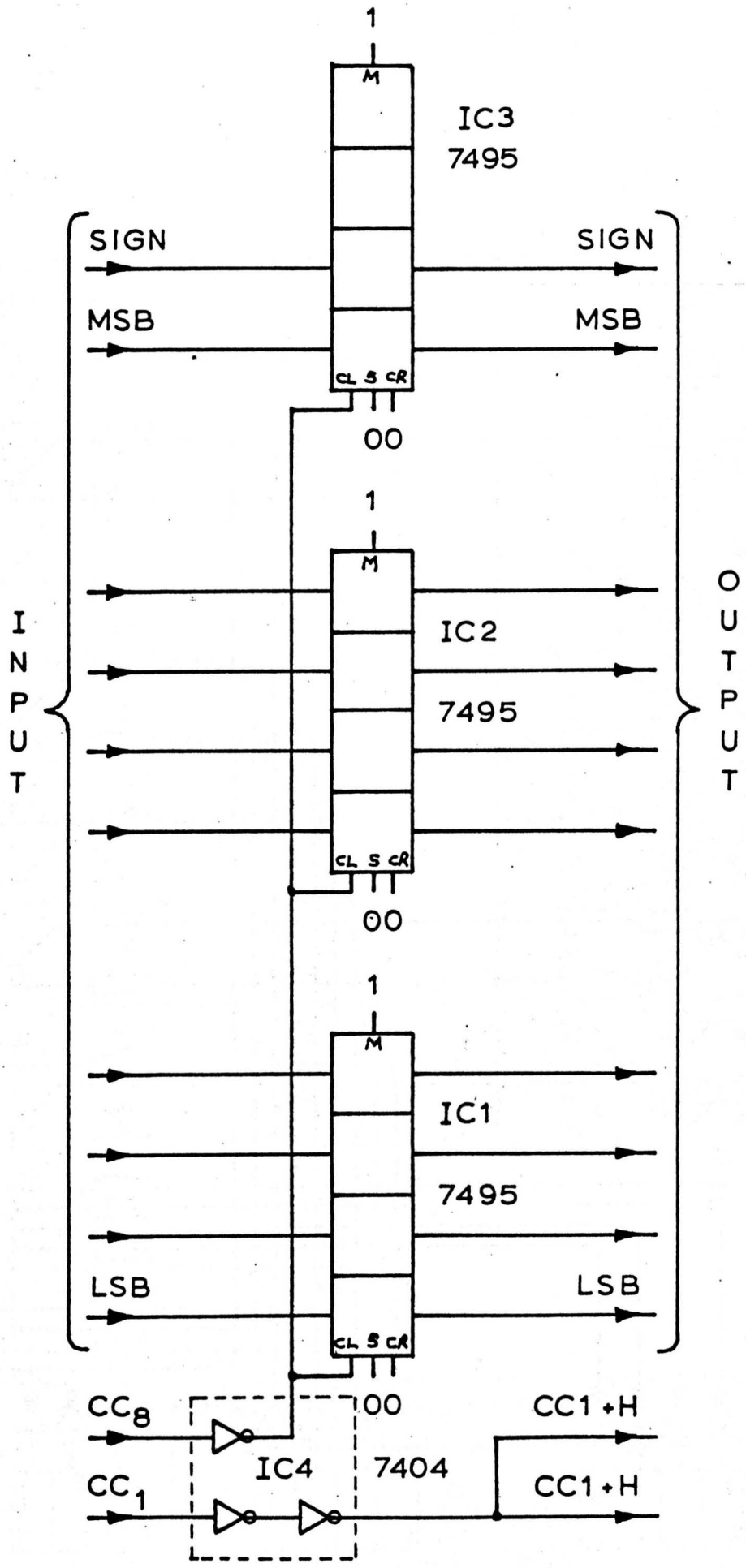


FIG. 6.21 BOARD 14a Buffer Store for BOARD 12

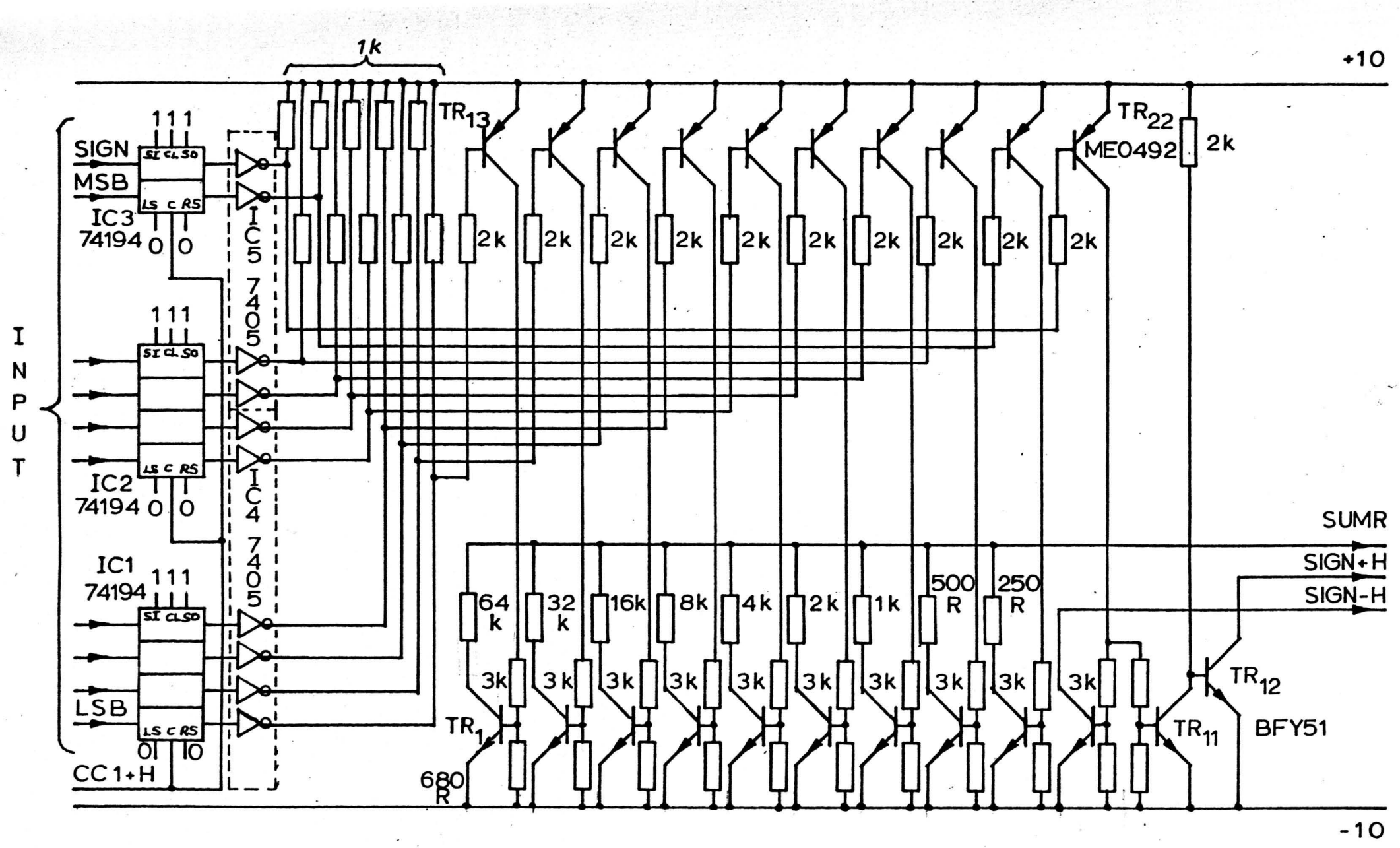


FIG. 6.22 BOARD 16 Buffer Store and Output Current Decoder 2

used to saturate TR11 and TR12 (BFY51).

6.5.12 Second Current Generator Board (18)

The second current generator board (Board 18 and Fig. 6.23) is fed with the resistance set by the shunt resistive ladder and the sign bit from Board 16. The basic circuit is the same as that used on Board 12.

6.5.13 Synchronous Sampler Board (22)

The synchronous sampler board (Board 22 and Fig. 6.24) has 2 sample and hold circuits each with a series MOSFET sampler. According to the delay required, one of the channel control signals could be selected to trigger a monostable (IC1) which in turn switched on the samplers (TR1 and 2) for a predetermined time. This sampling time could be adjusted by VR1 from 0.5 to 2 microseconds. Photo 6.7 shows the sample pulse derived from sampling a sinewave at the input to IC5 or 6 without the hold capacitor. Normally the hold capacitors C1 and C2 were present.

The input voltages were sensed by voltage followers (IC3 and 4) which in turn changed the hold capacitors when the series sampler was switched on.

The voltages across the hold capacitors were sensed by voltage follower (IC5 and 6) which in turn fed the appropriate A/D converter.

As the maximum input frequency was limited to 26.6˙ kHz (Table 7.1) the use of operational amplifiers (SN 7274IN) with a limited slew rate and a 1 MHz unity gain-bandwidth did not prove a drawback.

The use of a finite sample pulse width did not cause any problems because a duty ratio greater than 30:1 could easily be achieved.

6.5.14 Channel Selector Board (28)

The channel selector board (Board 28 and Fig. 6.25) de-multiplexes the serial 60-bit frame, feeding each word to the register

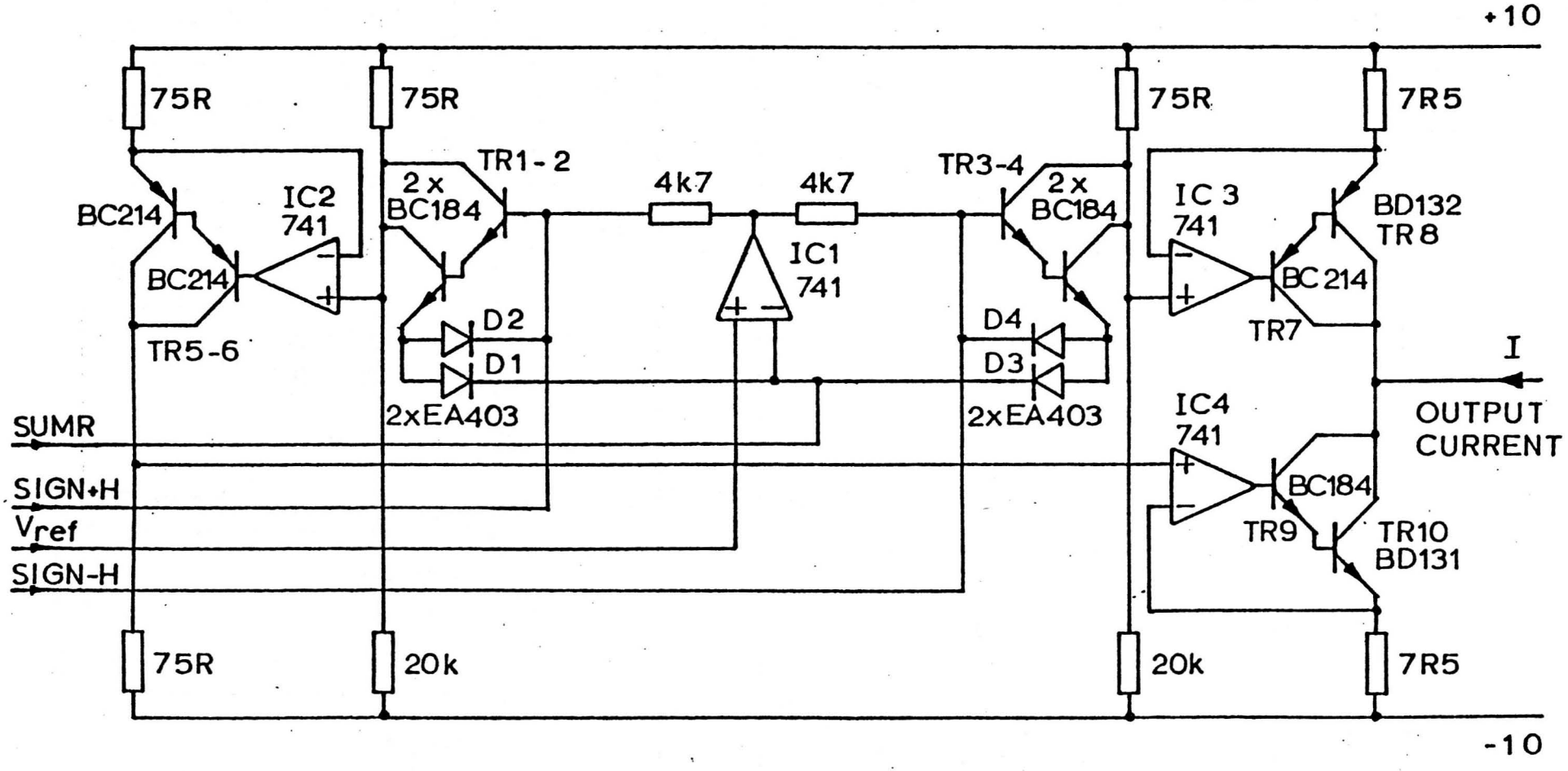


FIG. 6.23 BOARD 18 Output Current Decoder and Generator 1

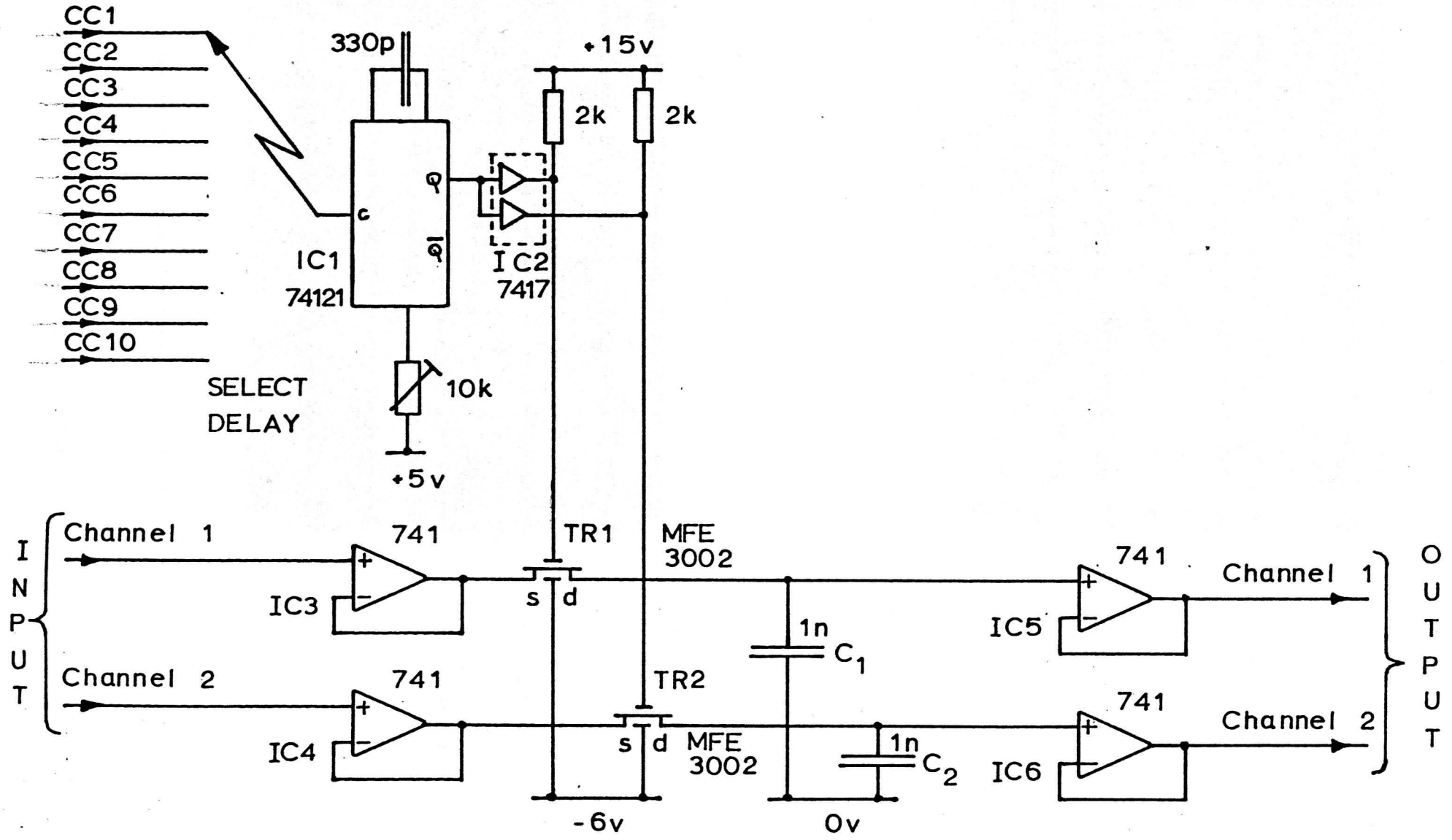


FIG. 6.24 BOARD 22 Synchronous Samplers and Input Buffer Amplifiers

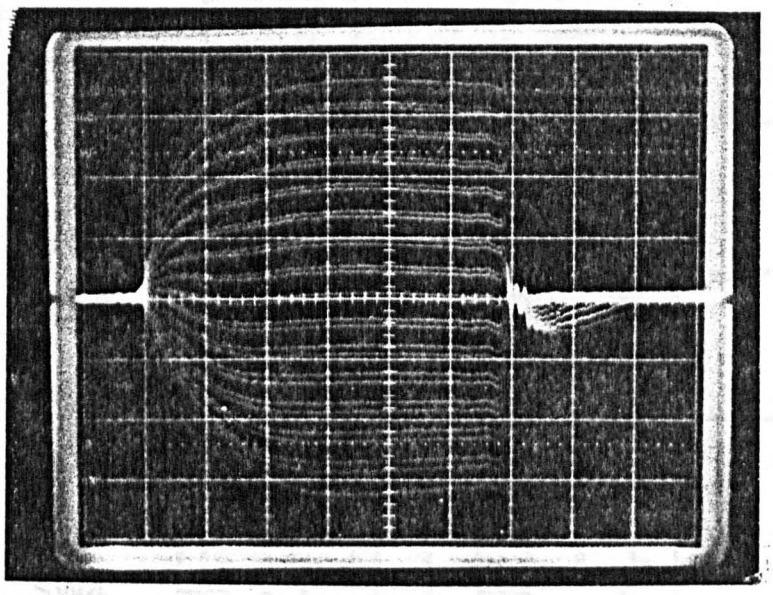


PHOTO 6.7

X 100ns / div

Y 2V / div

Sample Pulse with Sinewave Input

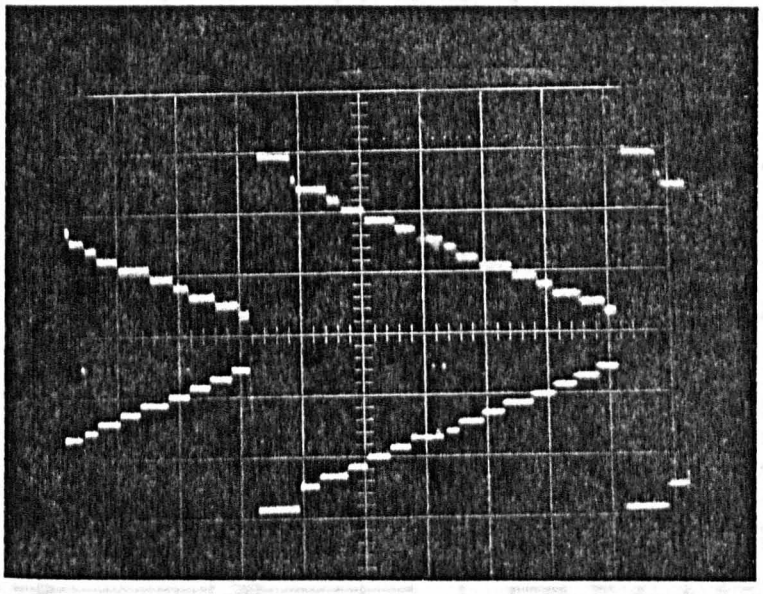


PHOTO 6.8

X 250 μs / div

Y_1, Y_2 1V / div

Sawtooth and Inverted Sawtooth waveform

Upper trace: Channel 4 Inverted

Lower trace: Channel 1 Non-inverted

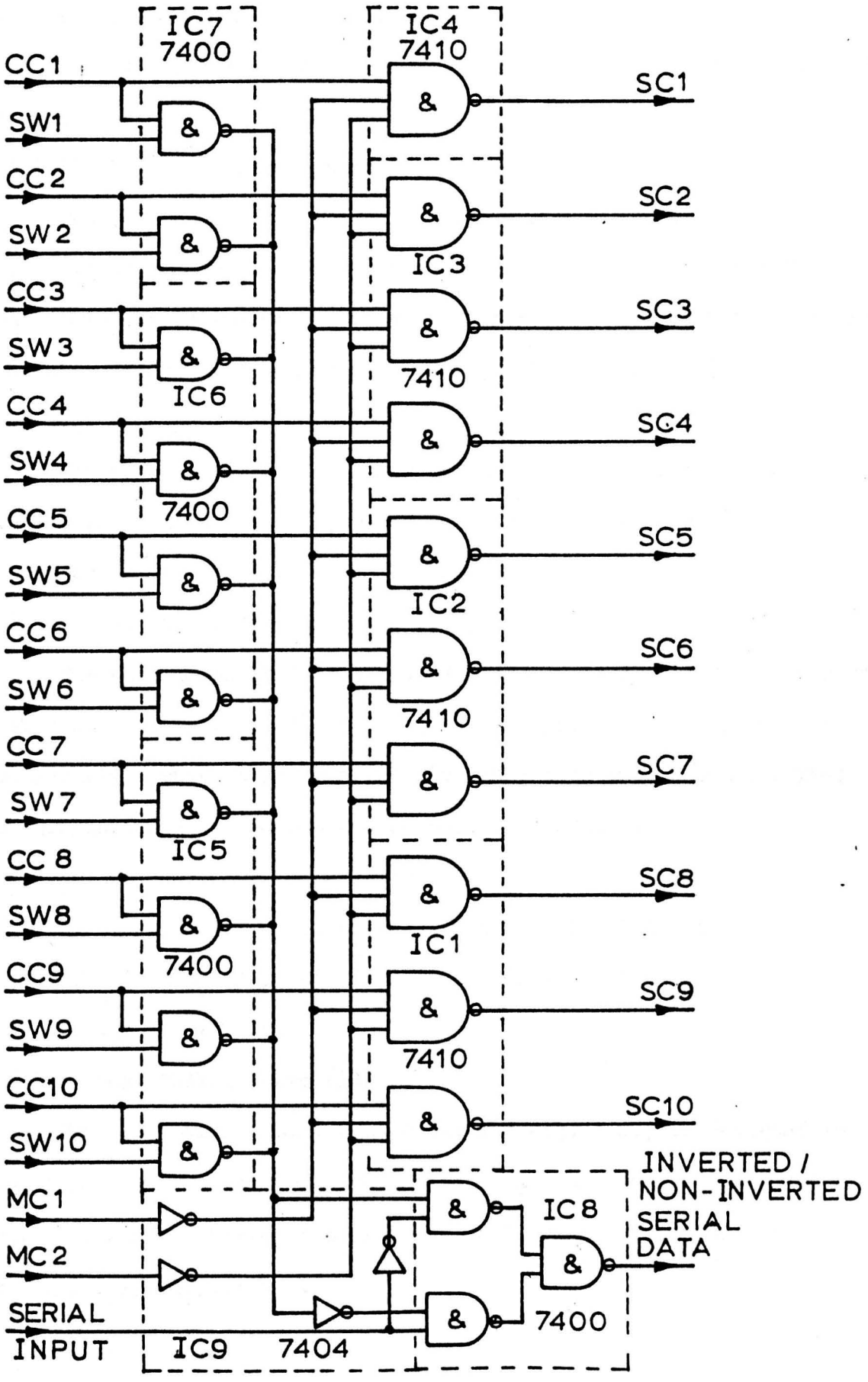


FIG. 6.25 BOARD 28 Channel Selector Board

in its respective D/A converter on boards 20 and 30. SW1 to SW10 are used to invert the bits in their equivalent word if required. Because each word is being transmitted as 5 bits of absolute magnitude, then inverting the bits will invert the analogue signal, and this is shown in Photo 6.8.

6.5.15 Channel Decoder Board (29 and 30)

The channel decoder boards (Board 29 and 30 and Fig. 6.26) are identical and hence interchangeable. Each contains five 5 bit D/A converters consisting of a storage register (IC1 to 5), an R/2R ladder, a level shifter and an emitter follower buffer amplifier (TR1 to TR5).

6.6 TEST BOARDS

The purpose of the following boards was to be able to test the machine in stages. The multiplier was always assumed to be present but the adder and the D/A converters could be replaced.

6.6.1 Test Level Shifter Board (10b)

The test level shifter board (Board 10b and Fig. 6.27) in conjunction with board 11b (Fig. 6.28) is designed to replace the adder on boards 10a and 11a. This board level shifts the 9 bits in IC1 and IC2 from the multiplier but does not add successive products.

6.6.2 Test Inverter Board (11b)

The test inverter board (Board 11b and Fig. 6.28) is designed to invert the 10-bit all magnitude word in IC1 and IC2 and is used in conjunction with Board 10b.

6.6.3 Test Buffer Board (14b)

The test buffer board (Board 14b and Fig. 6.29) is designed to act as a dummy buffer store board (14a). IC1 generates the necessary control signals, thereby removing the hold stage before the current generating D/A converters.

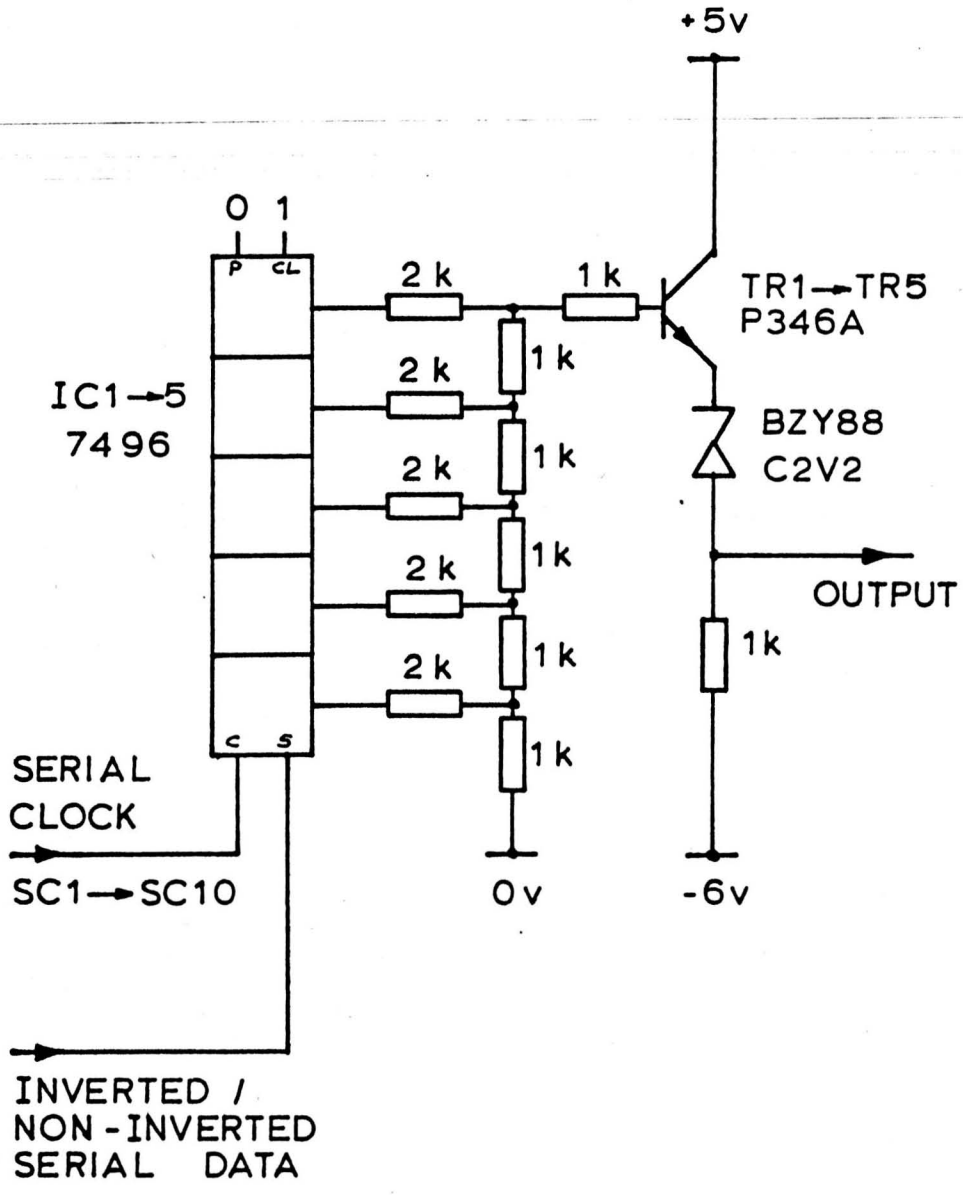


FIG. 6.26 BOARDS 29 and 30 Channel Decoder Boards

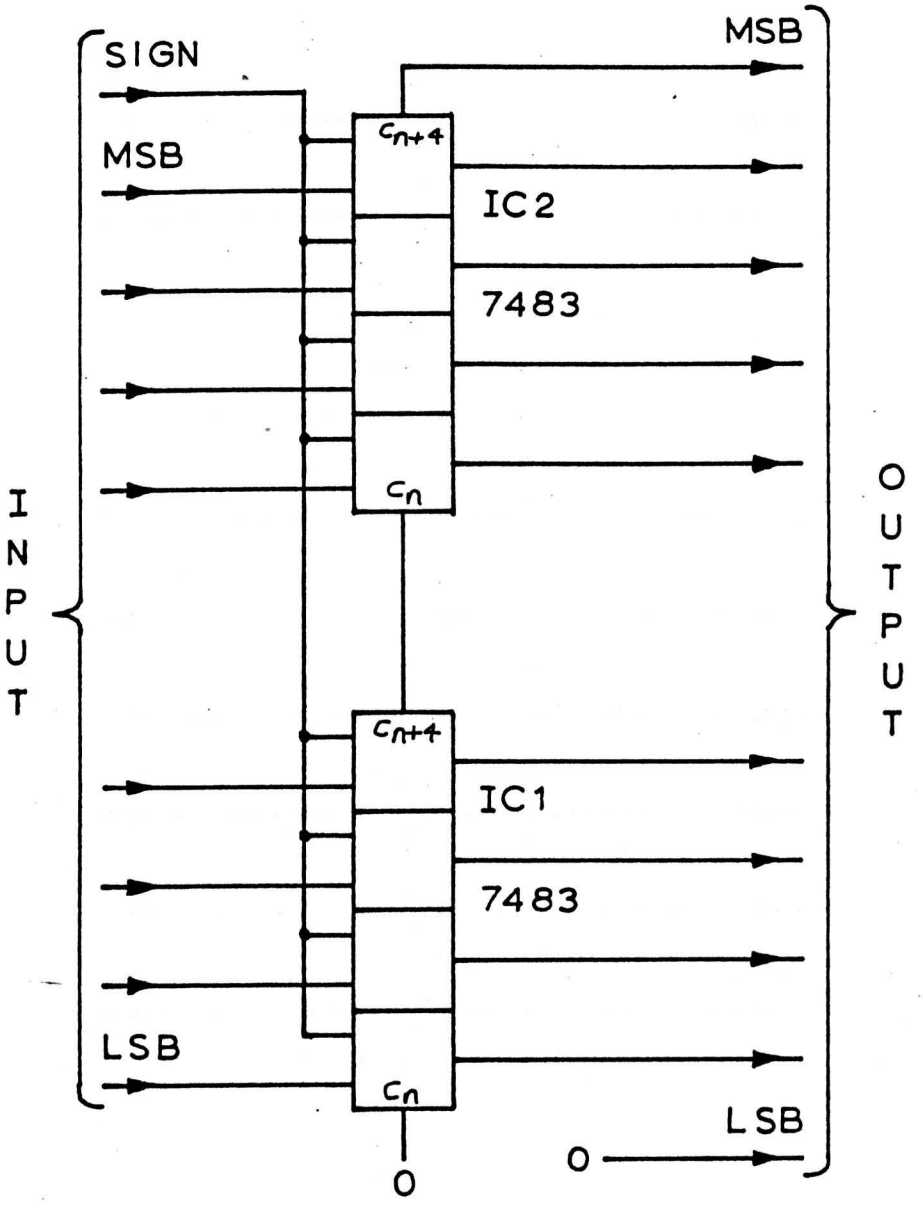


FIG. 6.27 BOARD 10b Test Level Shifter Board

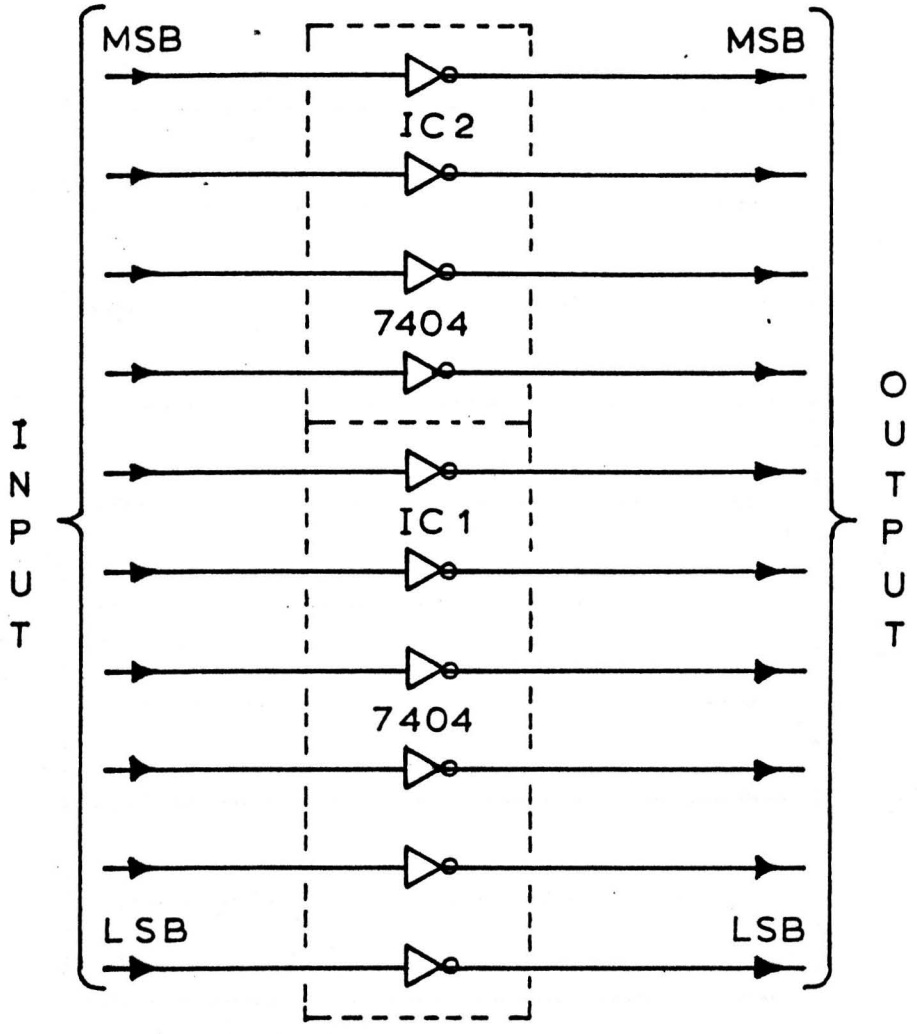


FIG. 6.28 BOARD 11b Test Inverter Board

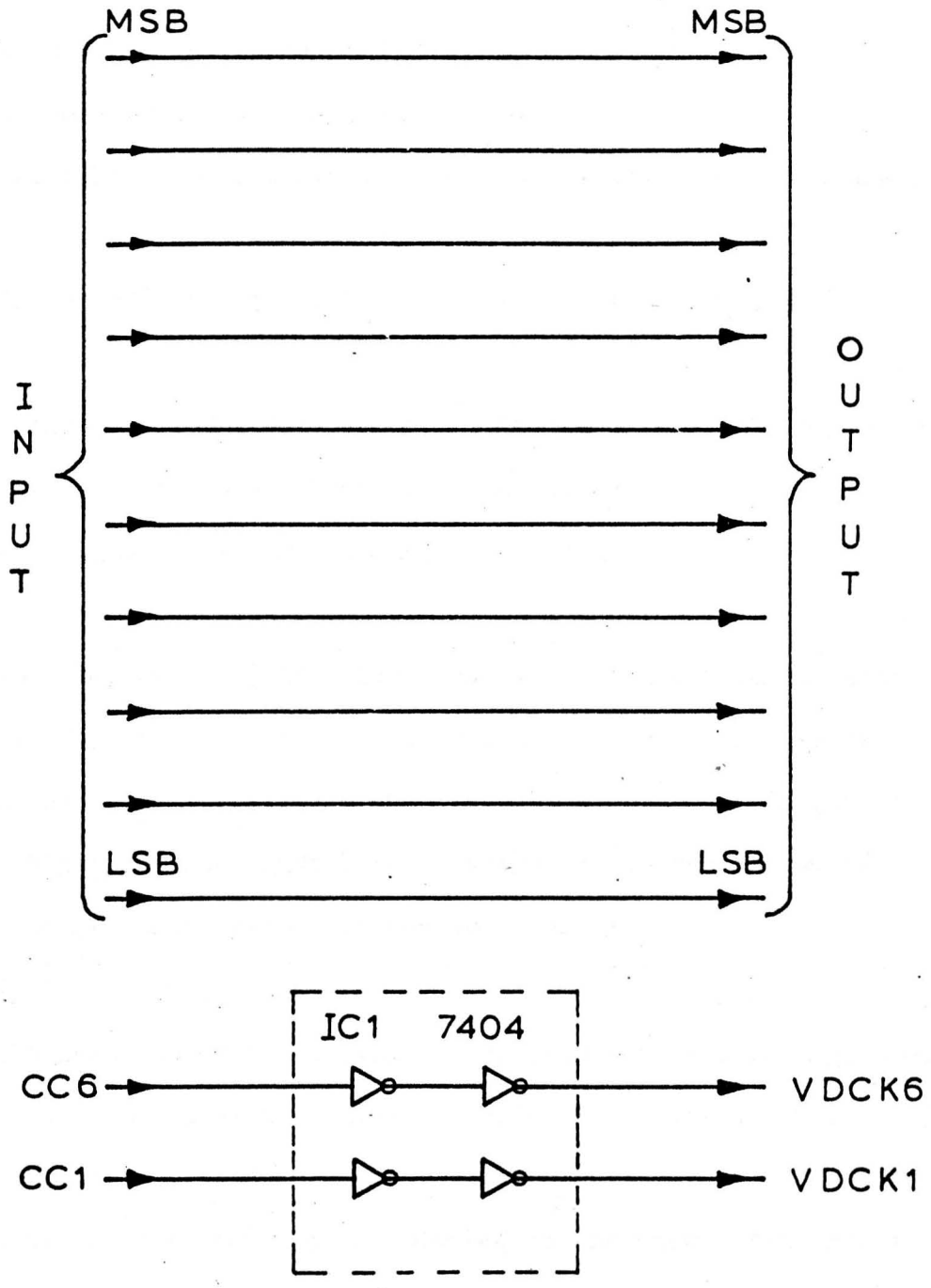


FIG. 6.29 BOARD 14B Test Buffer Board

6.6.4 Visible Decoder Board

The Visible Decoder Board (Fig. 6.30) may be inserted in sockets 14 or 16 and replaces one or other of the D/A current sources. The board contains 1 red and 9 green LEDs (LED 1 - 10) and the appropriate 10 bit buffer register (IC1 to IC3) and inverters (IC4 and 5). Thus for DC inputs to the machine the output word may be read and checked. This board was used in checking the multiplication and addition algorithms used, and in fault finding as necessary.

6.6.5 Substitute Voltage Decoder Board

The substitute voltage decoder board (Fig. 6.31) is designed to substitute for boards 13, 14, 15 or 16. A 10-bit register (IC1 to 3) is loaded by the correct channel control signal. The register outputs are then inverted by IC4 to IC6 (SN7437N) buffer NAND gates which drive an R/2R ladder for D/A conversion. Thus the signal path may be tested without the D/A current source being necessary.

6.6.6 Channel Tester Board (26)

The channel tester board (Board 26 and circuit 6.32) contains a 5-bit serial register (IC1) which may be selectively loaded with any one of the 10 words in the serial word frame. The contents of this register are then inverted in IC2 and displayed by 5 LEDs (LED 1 to 5) thus enabling any one channel to be either monitored or tested. IC3 and 4 generate the appropriate control signals.

6.6.7 Ramp Tester Board

The ramp tester board (Fig. 6.33) consists of a 7-stage binary counter (IC1 and 2) with ancillary combinational logic (IC3, 4, 5) and is designed to produce a waveform, after D/A conversion, as shown in Fig. 6.34. The converter is clocked by the combination $\overline{MC_2} + \overline{CC_{10}}$.

The 5-bit word from the combinational logic described above is simultaneously clocked into two 5-bit registers (IC7 and 8). One

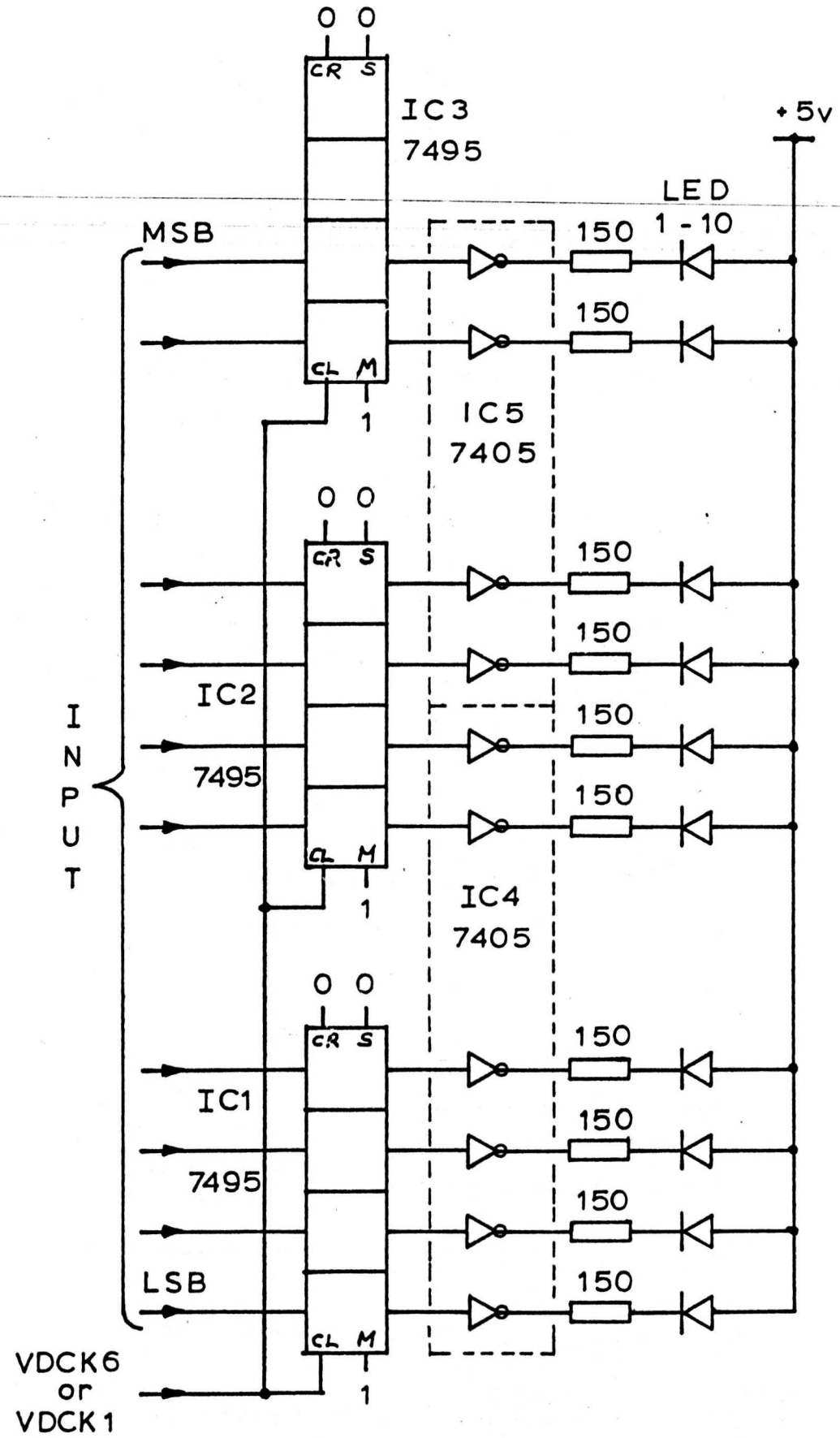


FIG. 6.30 BOARD 14 or 16 Visible Decoder Board

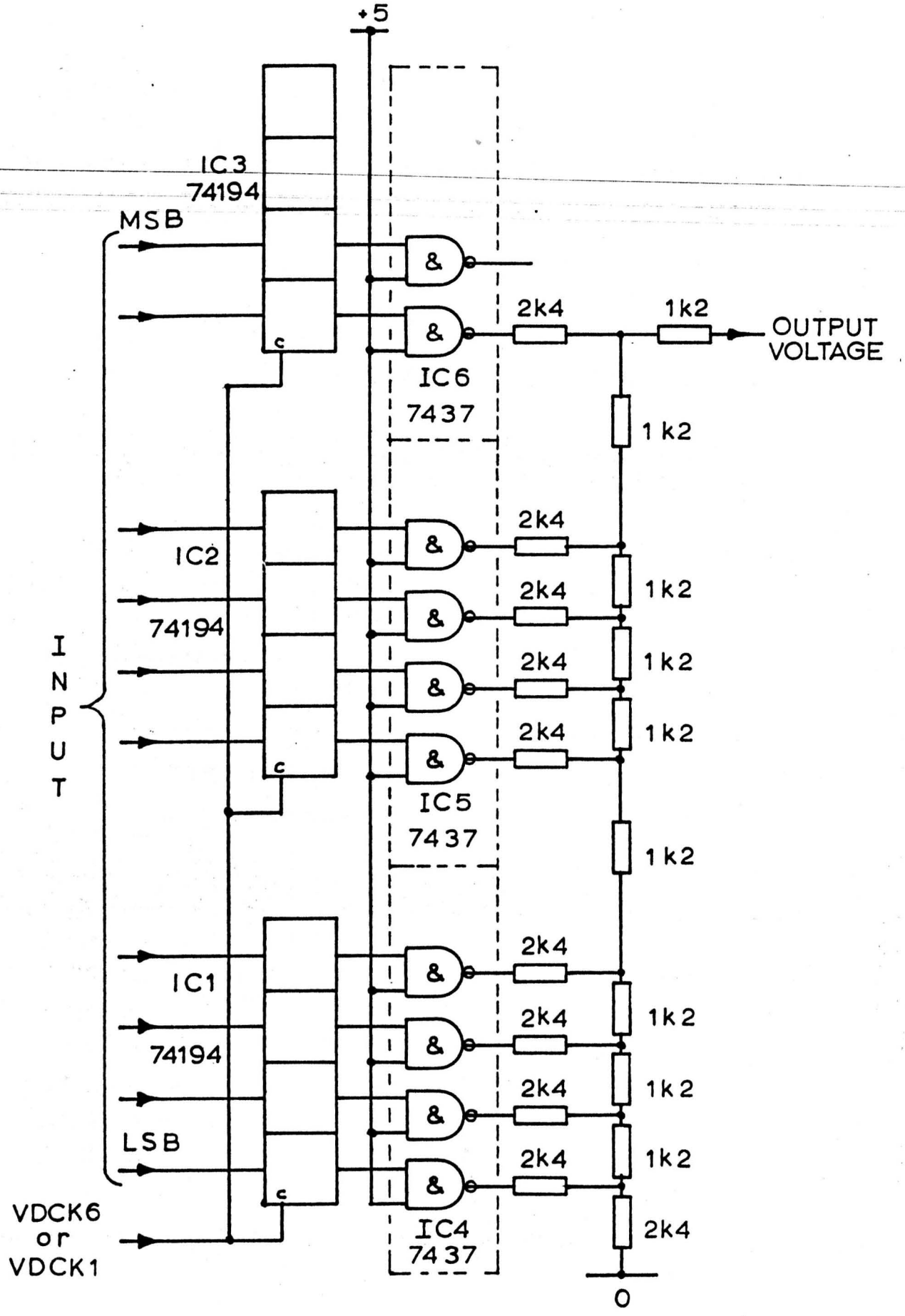


FIG. 6.31 BOARDS 13, 14, 15 or 16
Substitute Voltage Decorder Board

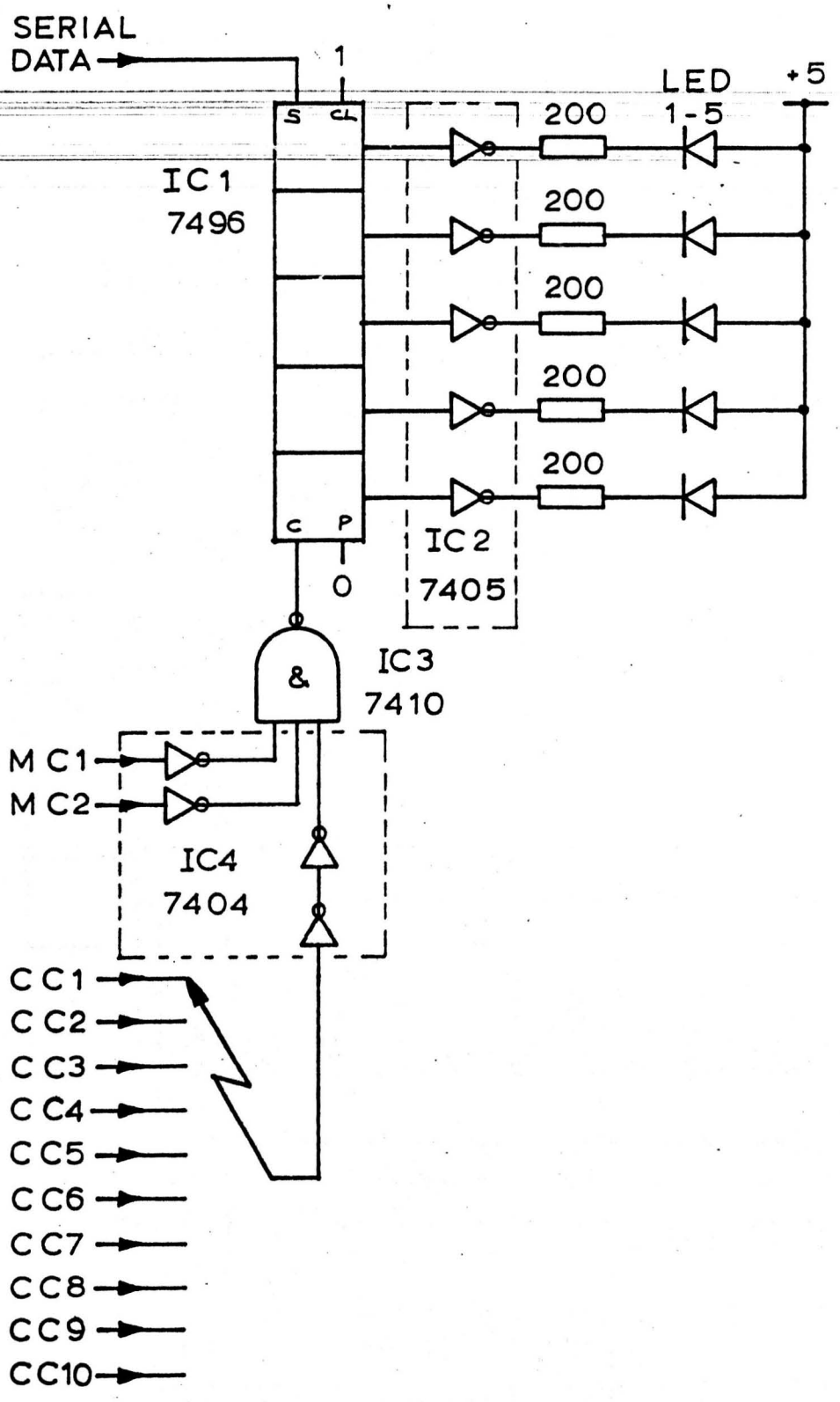


FIG. 6.32 BOARD 26 Channel Tester Board

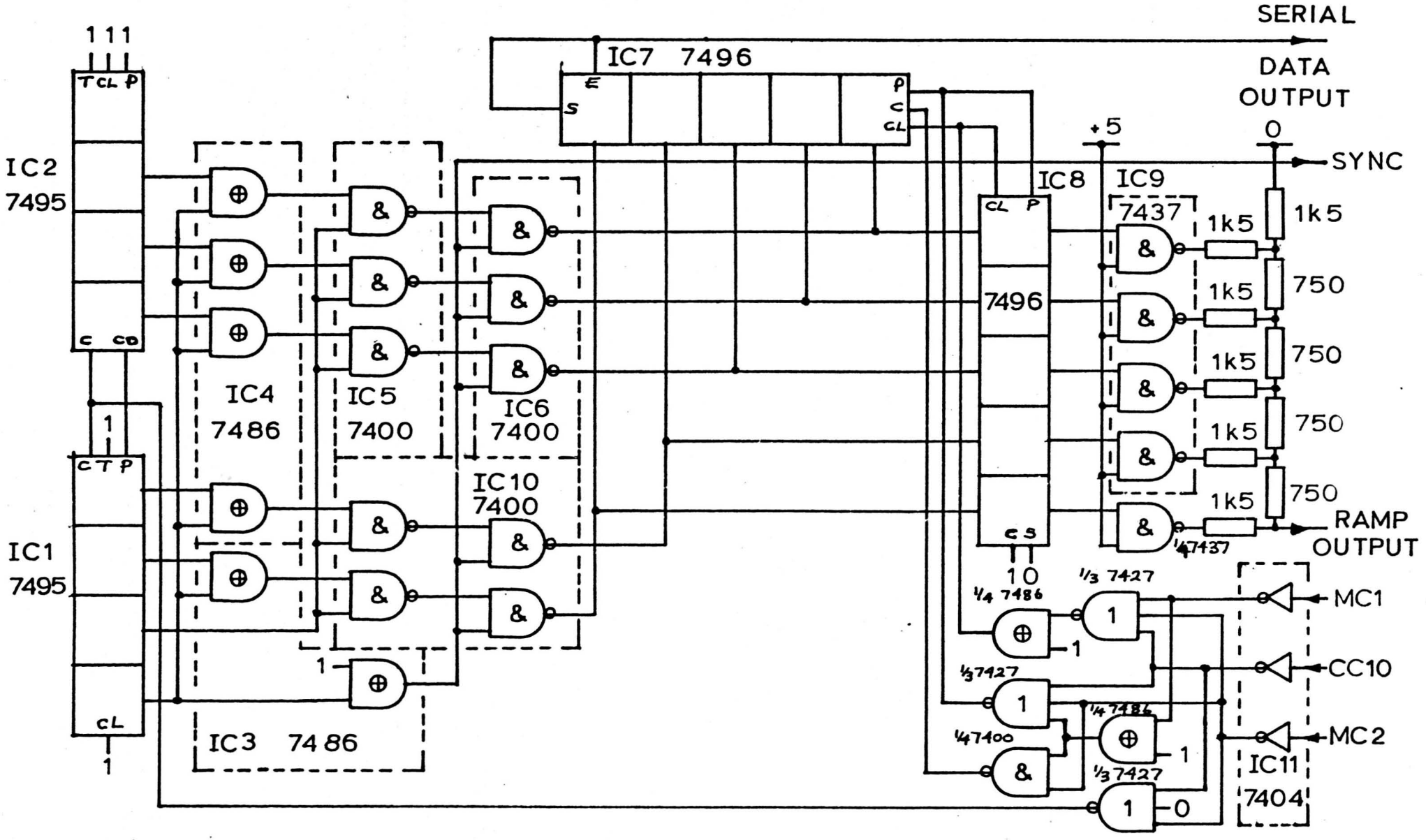


FIG. 6.33 Ramp Tester Board

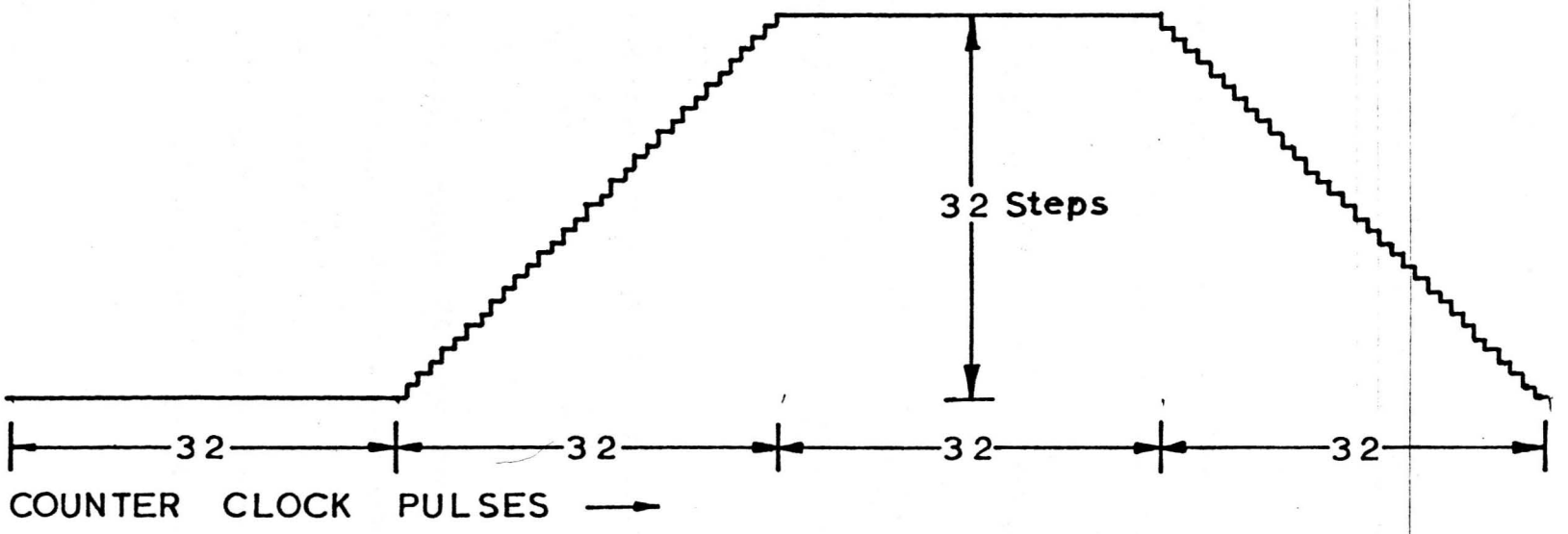


FIG. 6.34 Ramp Tester Wareform

register transmits this word in word 10 of the serial word frame. The other register drives an R/2R ladder through IC6 and 9 in order to make this test waveform externally available as an analogue signal.

6.7 EXTERNAL ANCILLARY EQUIPMENT

6.7.1 Parameter Adjustment Box

A simple variable buffered bipolar voltage source (Fig. 6.35) was made in order to be able to control the parameters of the matrix. The circuit shown in Fig. 6.35 was repeated 4 times within that box. The slide switch was included to enable a signal or a DC level to be buffered by IC1 and fed into the experimental machine.

6.7.2 Binary Attenuator Box

A binary attenuator box (Fig. 6.36) was built to act as a simple calibrated attenuator with a constant source impedance. The circuit is in principle the same as Fig. 6.31 with the reference voltage replaced by a signal input and the semiconductor switches replaced by slide switches.

6.8 INPUT CURRENT SOURCE

In order to test the digital gyrator by defining the input current (see Chapter 7) an analogue differential transconductance amplifier had to be built (Fig. 6.37). The circuit used closely follows that of the current source D/A converters (Figs. 6.20, 6.22 and 6.23). The input current source frequency response was tested and the results are given and described in Section 7.4.

From Fig. 6.37 and Appendix F the transconductance of this amplifier may be derived assuming the following conditions:

$$R_{1L} = R_{1R} = R_1$$

$$R_{2L} = R_{2R} = R_2$$

$$R_3 = R_4$$

$$R_{5L} = R_{5R} = R_5$$

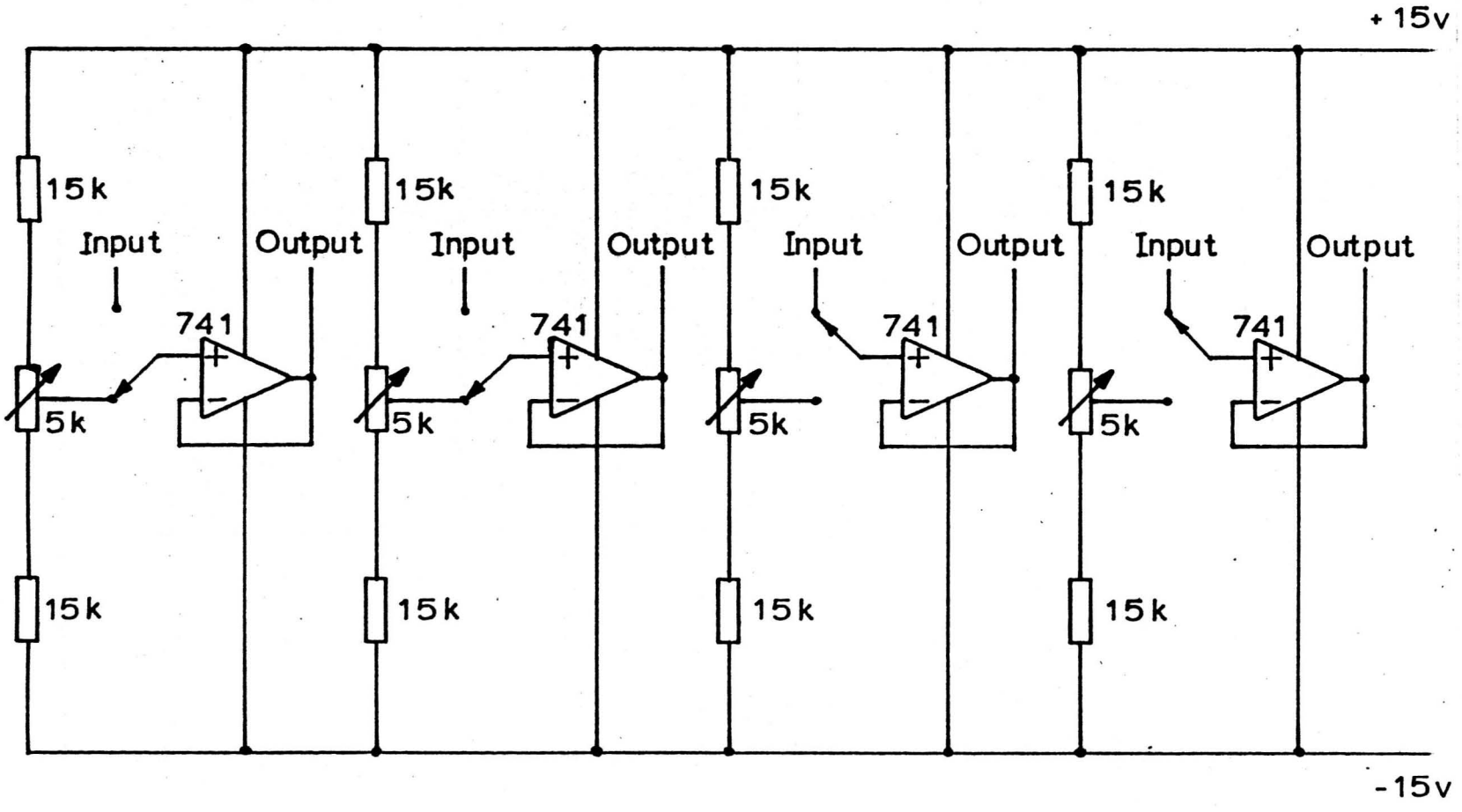


FIG. 6.35 Parameter Adjustment Box

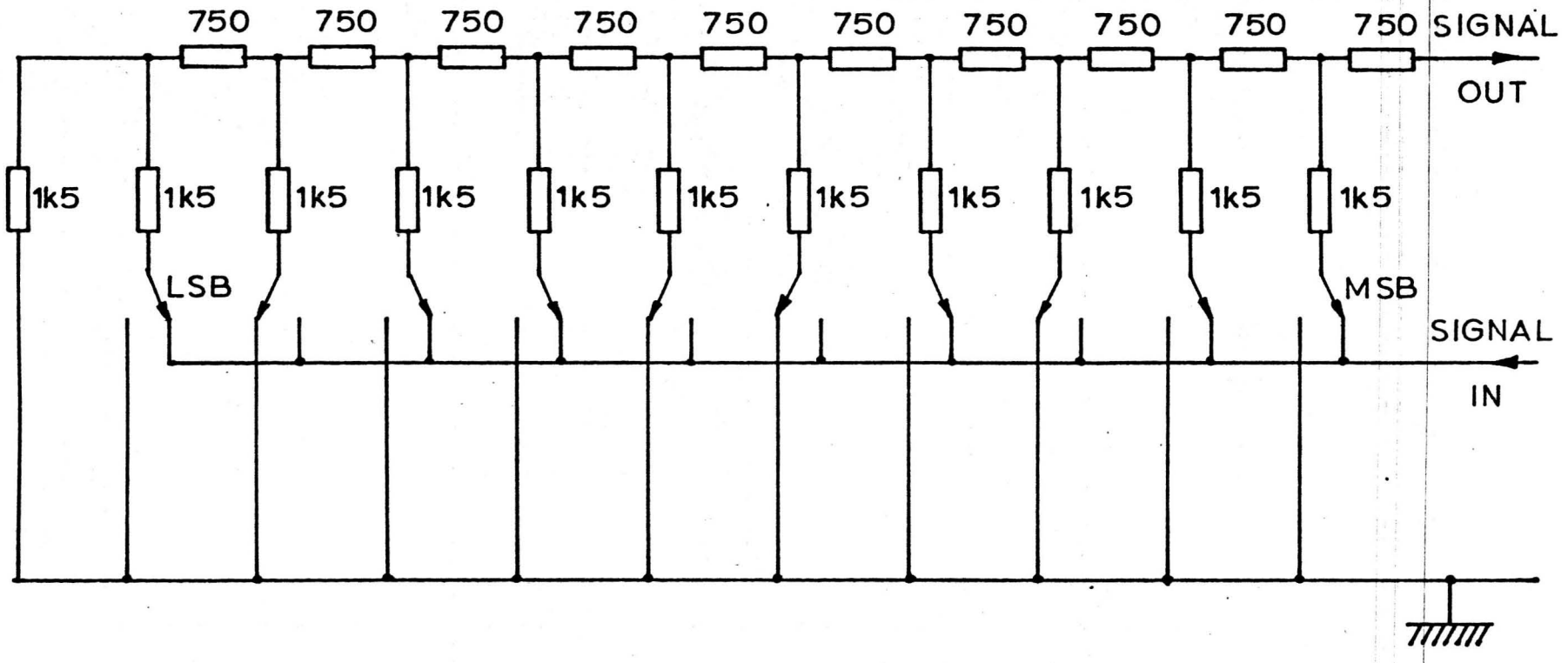


FIG. 6.36 R/2R Attenuator

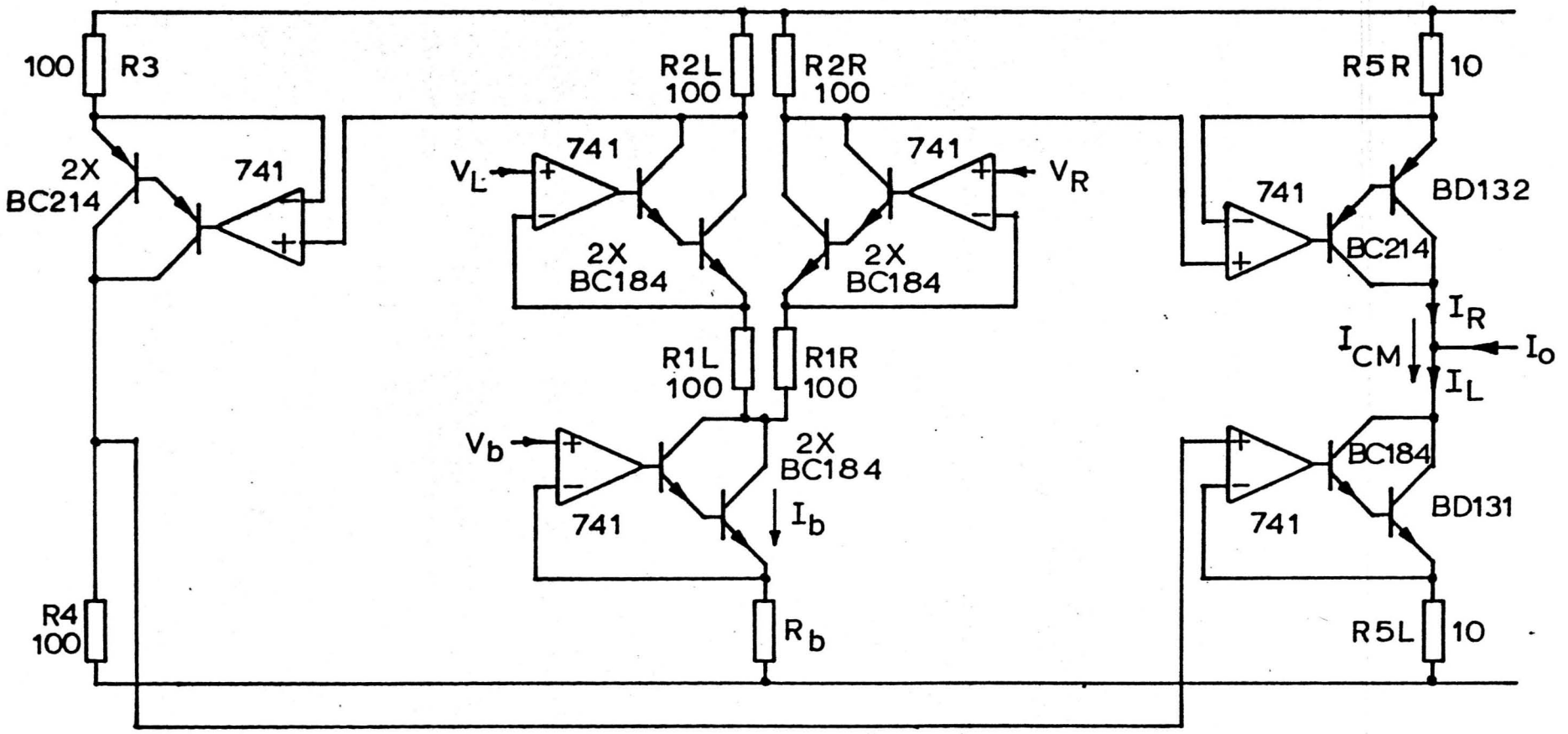


FIG. 6.37 Analogue Differential Transconductance Amplifier

The transconductance will be:

$$g = R_2 / R_1 R_5 \tag{6.26}$$

The common mode current (I_{CM}) in the output stage is controlled by I_b thus:

$$I_{CM} = \frac{I_b}{2} \cdot \frac{R_2}{R_5} \tag{6.27}$$

The maximum output current will therefore be:

$$I_o \Big|_{max} = 2\, I_{CM} \tag{6.28}$$

The various component values used in this amplifier are also shown in Table 7.14 and 7.21.

The characteristics of this amplifier are described further in Section 7.4.

6.9 MACHINE OPERATION

The quality of operation of this machine as a signal sampler, quantiser, multiplier adder and D/A current and voltage generator is of importance before use as a 2-port digital active network.

6.9.1 Voltage Transfer Function

The basic voltage transfer function is shown in Photo 6.9 for a 50 Hz sinewave input. All 32 quantisation levels are visible, though the slope is not entirely linear. This is due to the use of NAND buffer gates (SN7437) to drive the R/2R ladders in the D/A converter in the multiplexed A/D converter, and in particular the variable high and low state voltages between each gate.

A more sophisticated design of A/D converter, or a complete module was not used because of the cost.

Photos 6.10 and 6.11 show the voltage transfer functions for a 1kHz sinewave and squarewave respectively. The elliptical shape is due to the intrinsic time delay through the digital amplifier

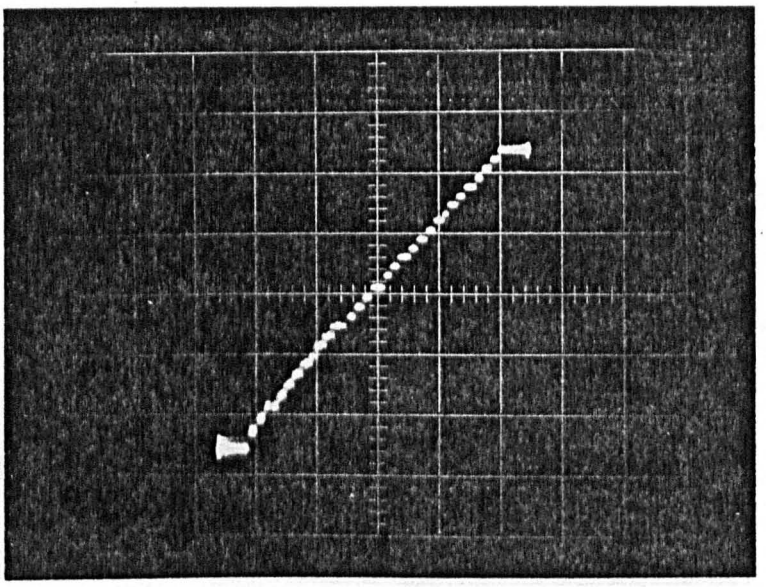


PHOTO 6.9

X 1V / div

Y 1V / div

Transfer function of one channel fed with

5 0 H z sinewave

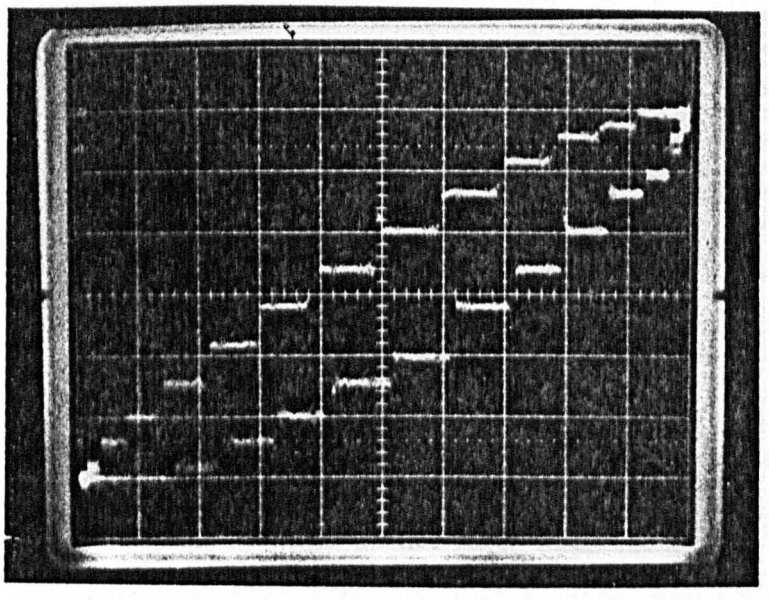


PHOTO 6.10

X 2.5V pp input

Y_1 0.5V / div

Transfer function for second D/A current generator

fed with 1 k H z sinewave

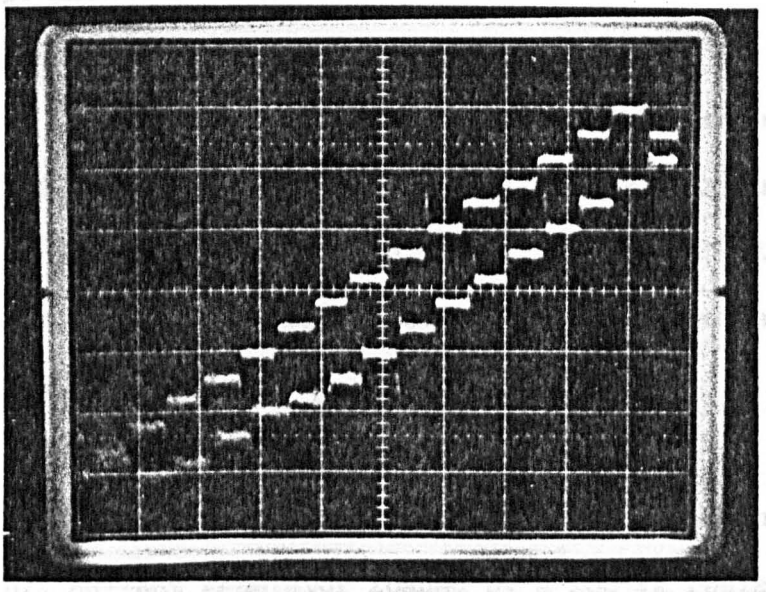


PHOTO 6.11

X 2.5V pp input

Y 0.5V / div

Transfer function of second D/A current generator fed with 1 kHz triangular wave

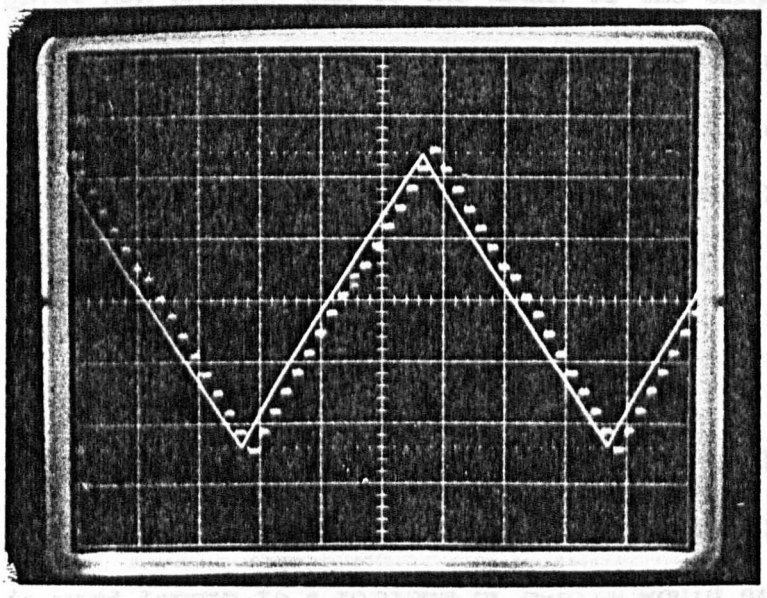


PHOTO 6.12

X 200 μs / div

Y_1 0.5V / div

Y_2 1 V / div

Channel 6 output for triangular input signal

introducing a group delay, that is a frequency dependent phase shift. This is not visible in Photo 6.9 due to the low input frequency of 50 Hz.

6.9.2 Output Converter Linearity

The output waveforms for a simple channel output through an R/2R ladder, and the outputs of the two current generating D/A converters are shown in Photos 6.12, 6.13 and 6.14 for a triangular input waveform. The current generating D/A converters were loaded with 100 Ω resistors.

6.9.3 Multiplier Transfer Function

The square law transfer function from the output of the second D/A current generator loaded by a 100 Ω resistor is shown in Photo 6.15. The square law was obtained by feeding a 50 Hz sinewave into two inputs simultaneously and arranging the machine to compute the product.

Photo 6.16 shows the transfer function for a 1 kHz sinewave input together with the frequency doubled output signal. Again group delay can be seen due to the delay of one sample period in the machine.

Photo 6.17 shows the input and output waveforms when squaring, showing in particular the frequency doubling without any significant D.C. offset. In fact the squaring arrangement was very useful for trimming out the A/D converter D.C. offset.

6.10 POSSIBLE IMPROVEMENTS

6.10.1 Word Length

It was found by experiment that the most serious shortcoming of this machine was the 5-bit word length (see Chapter 7). An increase in word length to a minimum of 8-bits would considerably improve the dynamic range and signal/noise ratio of this machine (see Table 6.2). However, this would imply a significant increase in the component cost

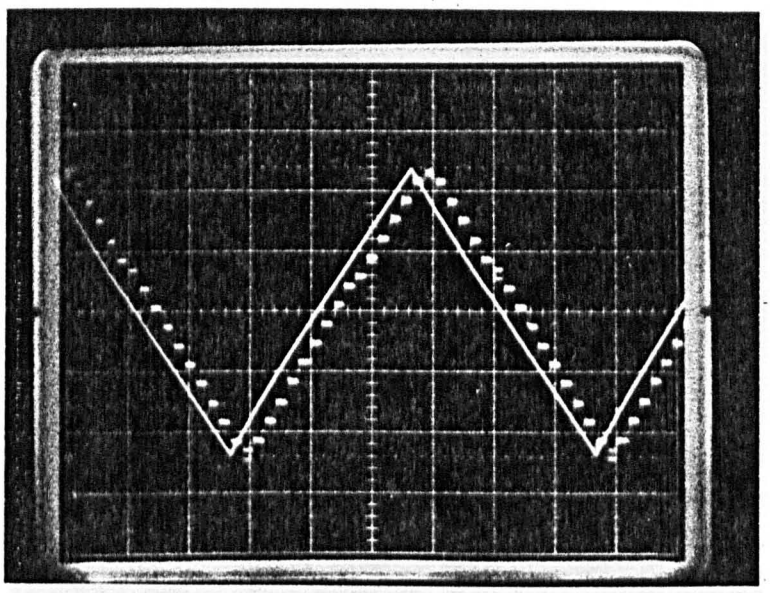


PHOTO 6.13

X - 200 μ s / div

Y_1 - 0.5V / div

Y_2 - 0.5V / div

Output current from first digital amplifier D/A current converter.

Y_1 - Triangular input signal

Y_2 - Output current from first D/A converter

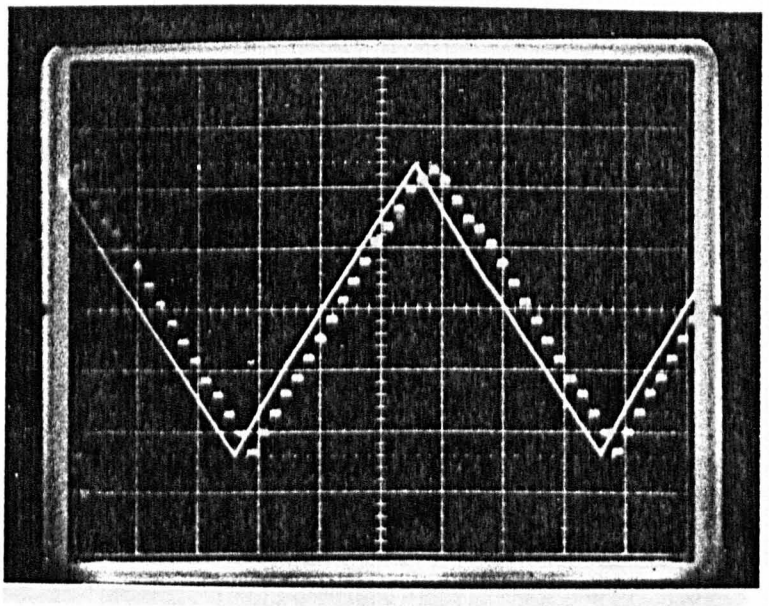


PHOTO 6.14

X - 200 μ s / div

Y_1 - 0.5V / div

Y_2 - 0.5V / div

Output current from second digital amplifier second D/A current converter

Y_1 - Triangular input signal

Y_2 - Output current from D/A converter

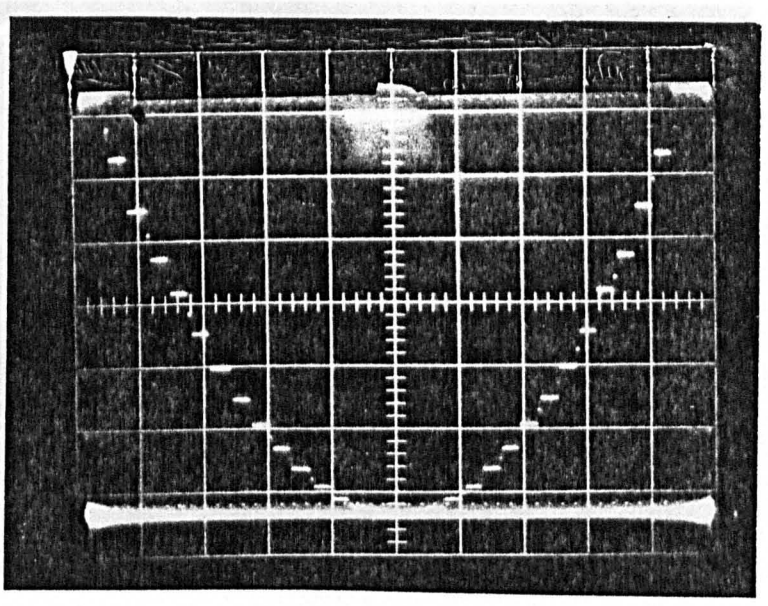


PHOTO 6.15

X 2.5V pp

Y 0.1V / div

Square law transfer function.
Output from second D/A current generator loaded by 100 Ω resistor with 5 0 H z sineware input

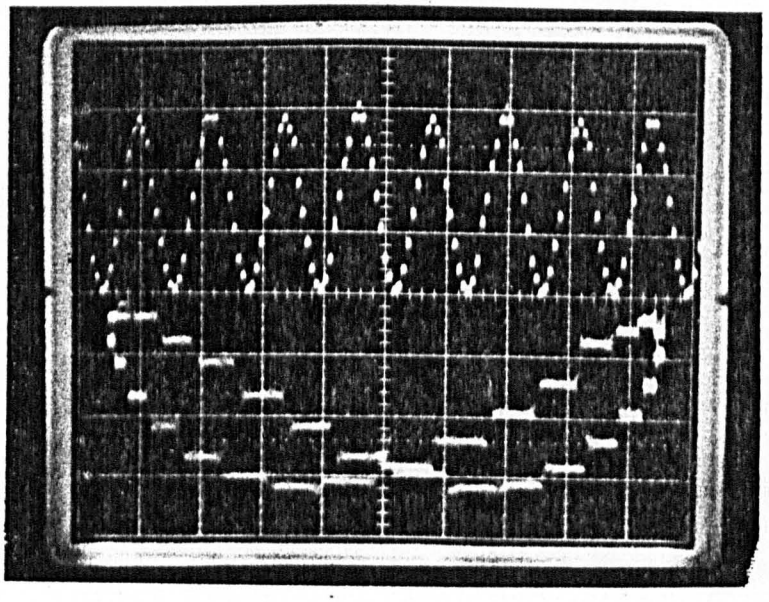


PHOTO 6.16

X_1 0.5ms / div

X_1 0.5V / div

X_2 2.5 V pp

Y_2 0.5V / div

Square law transfer function and squared output signal from second D/A current generator loaded by 100 Ω resistor and fed with 1 kH z sineware input

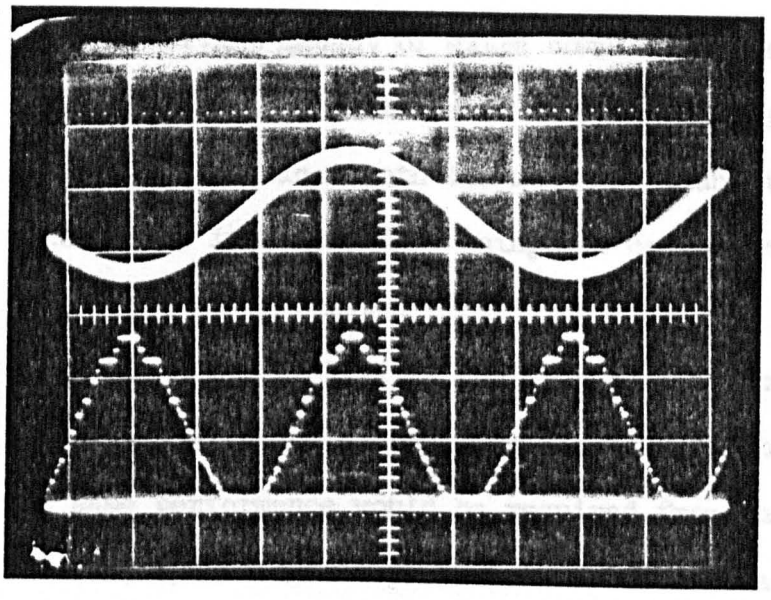


PHOTO 6.17

X 1 ms / div

Y_1, Y_2 0.5V / div

Input and output wareforms to squaring circuit.

if the present machine architecture were to be kept. Sections 6.10.4 and 6.10.7 suggest improvements to this architecture.

6.10.2 Two's Complement Arithmetic

A certain amount of extra complexity was caused by performing the arithmetic calculations in this machine by other than 2's complement arithmetic. Both the multiplier and the adder could have been more easily implemented by using 2's complement arithmetic.

6.10.3 A/D Conversion

For a 2-port digital admittance matrix only 6 A/D converters are required. With an increased number of bits in a word approximately the same performance would be required from the multiplexed converter. However, by having a separate converter for each input, the sample period could be reduced to at least one-sixth of the present maximum rate.

Also, it is now possible to buy successive approximation A/D converters. [25], [26], [27]. This type of converter would greatly simplify the construction of the input A/D converters and the output D/A converters.

6.10.4 Multiplication

The present multiplication scheme uses the shift-and-add algorithm (see Section 6.3.3). It would now be more economical and much simpler to use a combinational multiplier such as described in [28].

6.10.5 Multiplexing

Currently the machine uses a multiplexed A/D converter, multiplier and adder. For a specific application this could be replaced by individual stages in each signal path, as in Fig. 6.38 which would of course increase the maximum speed of the machine. However, in the general case this arrangement would be totally inflexible.

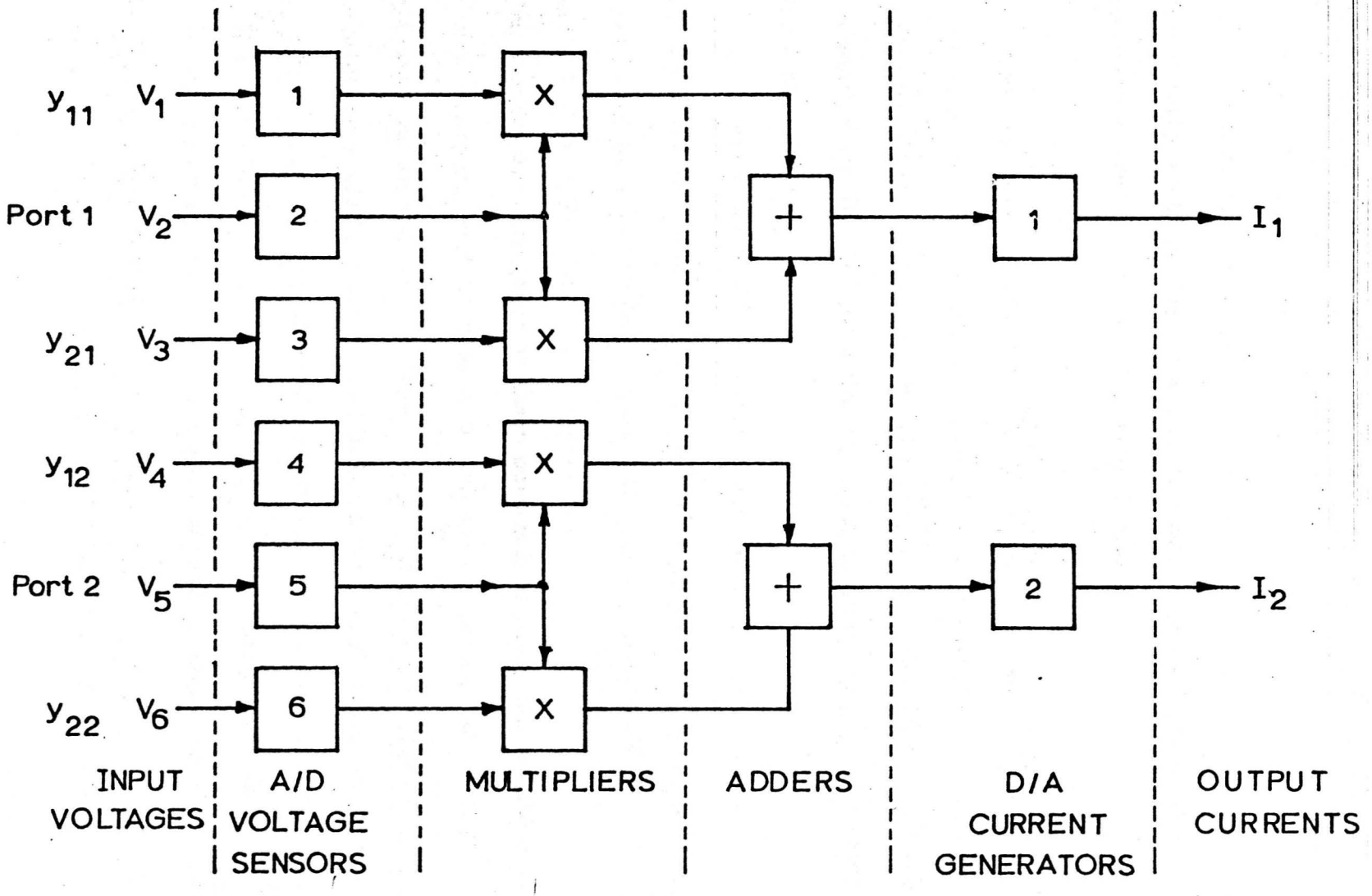


FIG. 6.38 Alternative Machine Structure

6.10.6 Data Transmission

The present multiplexed A/D converter produces a serial output which is then multiplexed through the rest of the machine. A better solution which could be incorporated with new A/D converters would be to feed these A/D converter outputs via a tri-state, bi-directional bus. This would then make multiplexing easier and more sophisticated. A block diagram of this structure is shown in Fig. 6.39. With this structure it would then be easy to interface this machine with a digital computer thereby enabling the computer to control parameters such as the transconductance of the digital amplifiers.

6.10.7 Microprocessor Control

By developing the concept in the previous section the digital amplifier array could be constructed using a multiplicity of peripherals on a microprocessor bus system, thereby giving over the control of the total structure of the array to the resident program. Fig. 6.40 shows a typical structure.

The microprocessor CPU is assumed not to do any arithmetic calculations for the arrays, merely controlling the passing of data from one peripheral to another. Addition could be performed by the microprocessor and hence this peripheral would be optional.

A significant advantage of this approach is that the matrix element parameters could be stored in the memory instead of having to be fed in through A/D converters. This of course only applies to the simpler amplifier arrays and would not be applicable when elements are to be rapidly changed or controlled parametrically.

The flow chart for calculating the output current for one equation or row of the digital admittance matrix is shown in Fig. 6.41.

6.10.8 D/A Current Source

An improved D/A current source may be designed by using the

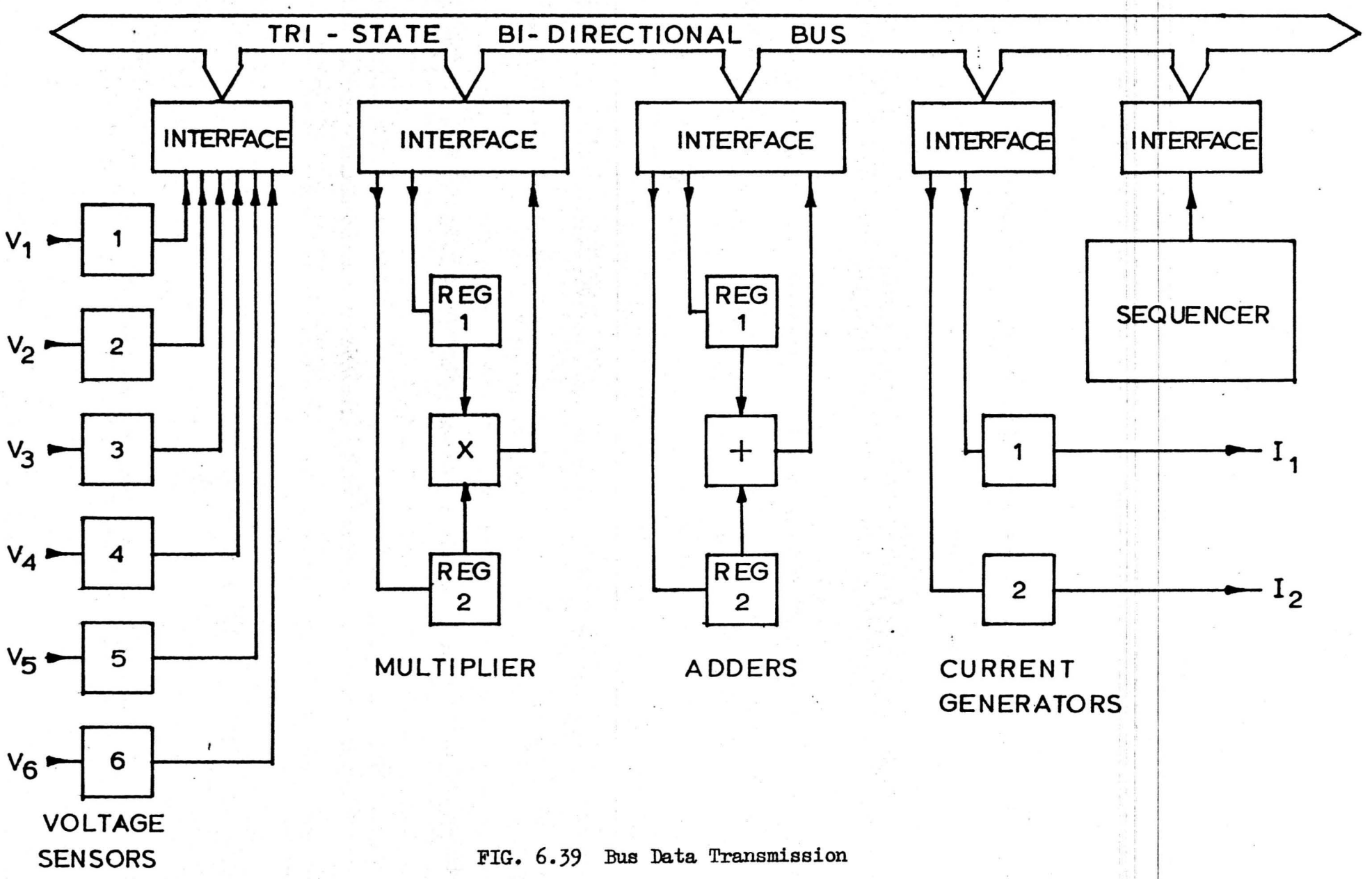


FIG. 6.39 Bus Data Transmission

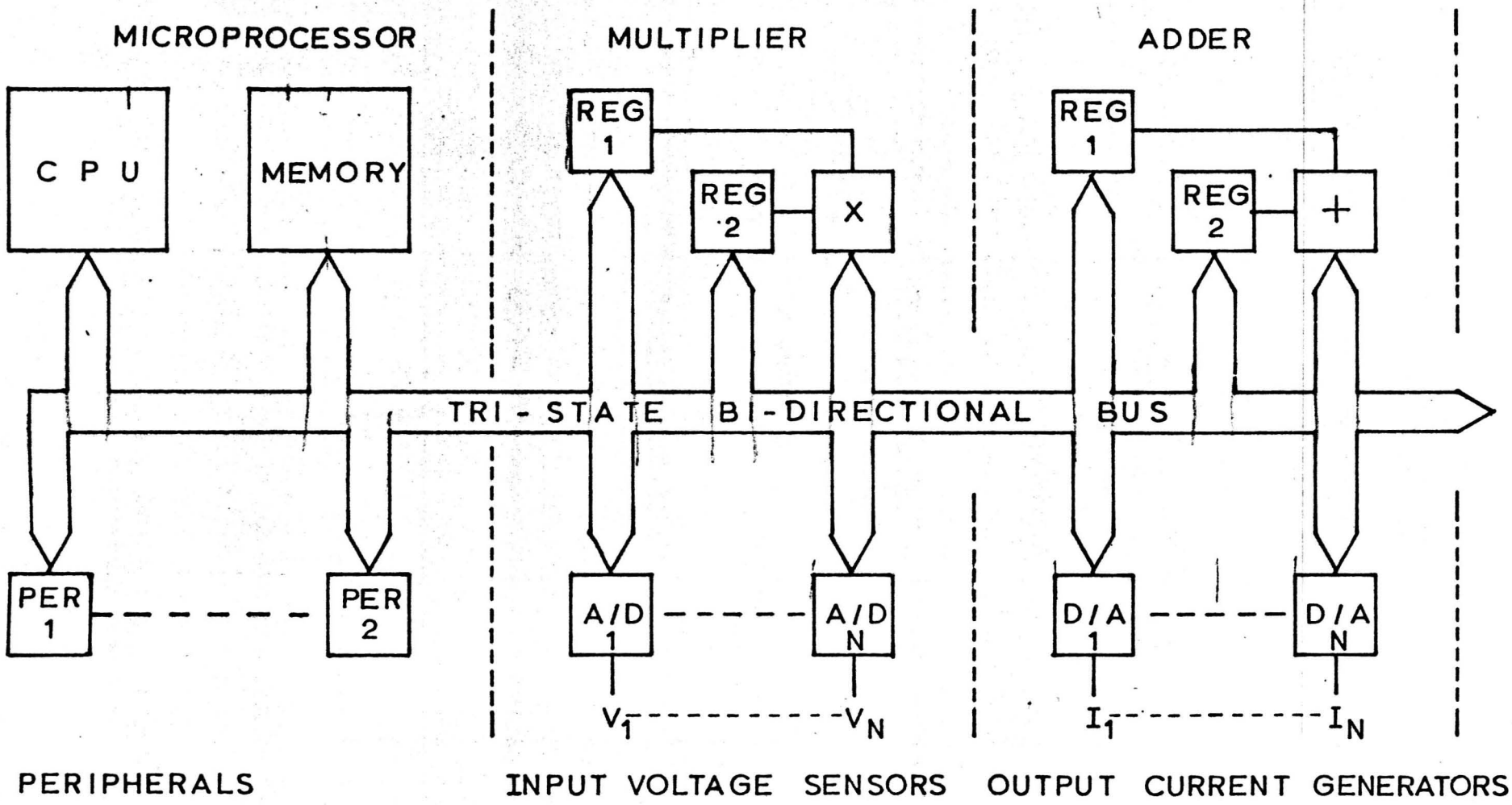


FIG. 6.40 Microprocessor Control of Digital Amplifier Array

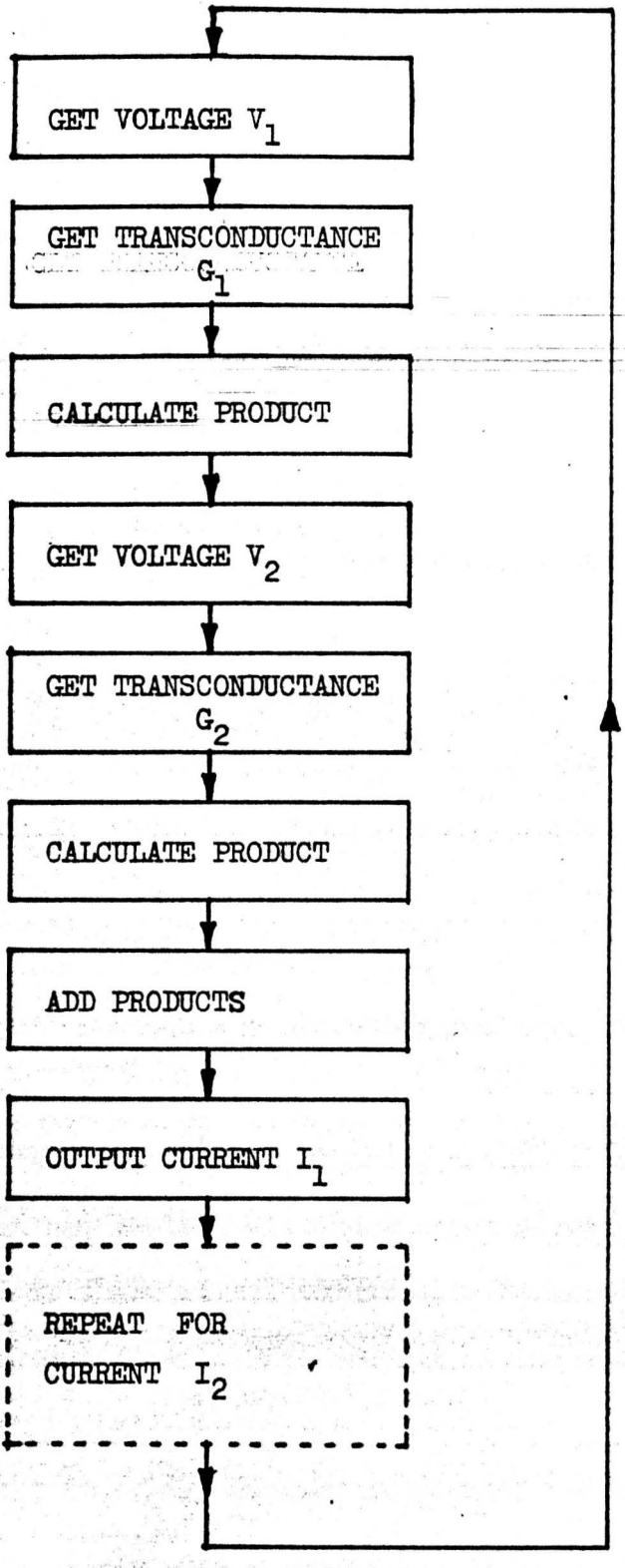


FIG. 6.41 Calculating Flowchart

alternative R/2R ladder to create current sources with a binary weighting. Circuit 6.42 shows a 10-bit D/A current source. The circuit is designed so that all the current sources operate continuously rather than being switched on and off. The diodes form current steering switches; the currents into the current sources are switched so as to flow from the output node or from a saturated transistor switch.

The input word has to be arranged to be offset binary coded because the output current is the difference between the fixed current from the upper current source and the steered current sunk by the binary weighted current sources.

The reference voltage (V_b) setting the scale factor of the decoder is converted to a current by transconductance amplifier 1 and reflected by amplifier 2. Thus the upper current source and the lower binary weighted current sources both have the same reference voltage.

The first advantage of this improved D/A current generator is that because the current sources are operating continuously, at fixed values of current, the limitation on the settling time of this circuit is the speed that the current steering switches can operate.

The second advantage is that all the current sources are biased in class A and therefore there will intrinsically be no crossover distortion between current flowing into and out of the output node.

6.11 SUMMARY AND CONCLUSIONS

A 10-channel digital signal processing machine has been described, containing a multiplexed A/D converter, multiplier and adder. This machine was used to simulate two digital transadmittance amplifiers each with a bandwidth in excess of 25 kHz. The machine was designed to simulate the 2-port capacitively loaded digital gyrator which was analysed in Chapter 4.

The major drawbacks in the design of this machine have been

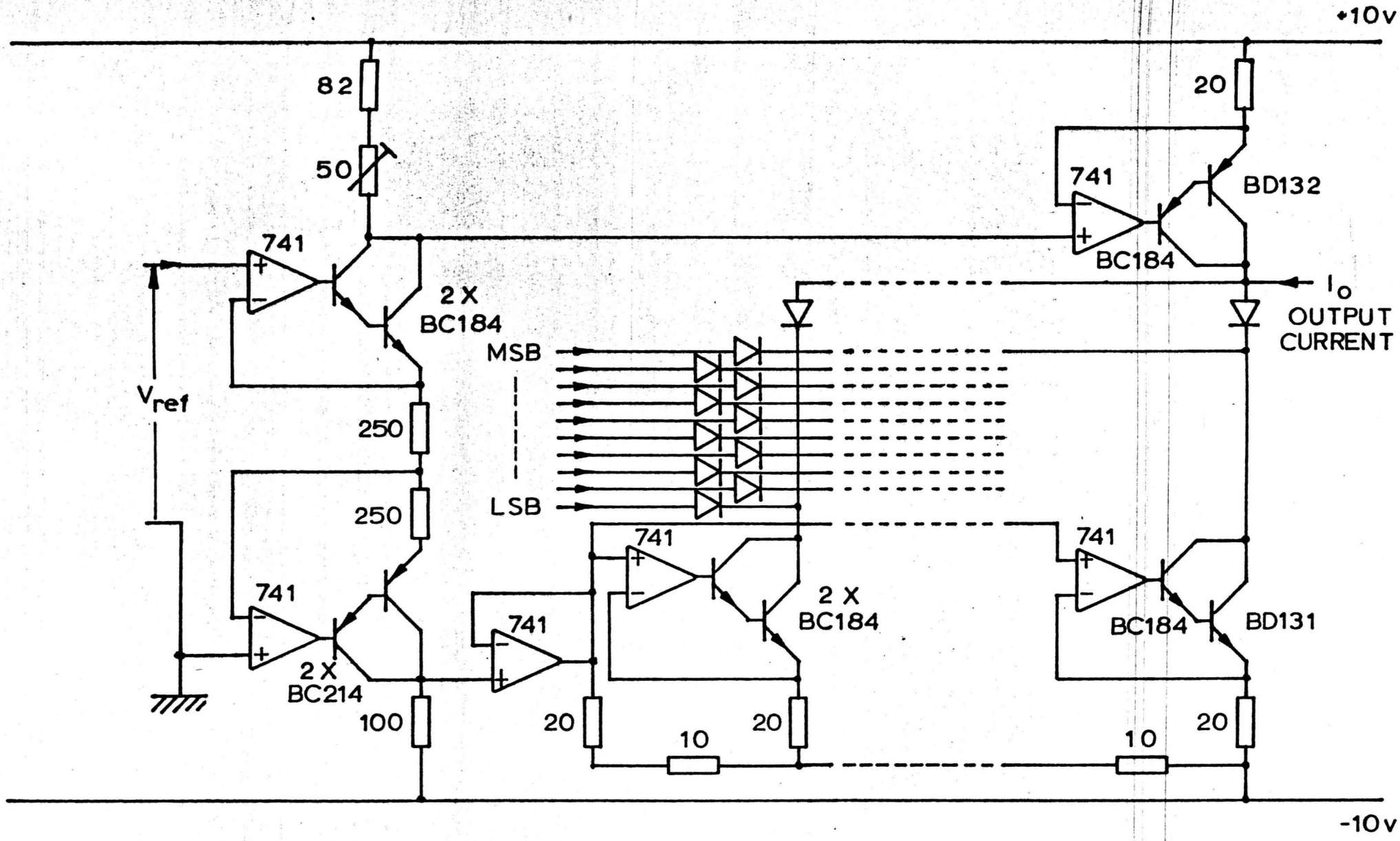


FIG. 6.42 Improved D/A Current Source

described and considerable improvements suggested. In particular the number of bits in each digital word was found to be barely sufficient and the arithmetic algorithms used were cumbersome.

Nevertheless a working digital admittance matrix has been designed, constructed and tested, then used as a digital gyrator.

CHAPTER 7

PRACTICAL RESULTS

7.1 INTRODUCTION

The experimental machine was set up as described in Chapter 6 to act as a 2-port capacitively loaded digital gyrator. Measurements were then taken for the magnitude and phase frequency responses for the digital input impedance (z_{11}) and forward transimpedance (z_{21}). These two elements were studied because they represented the two basic types of element as seen in matrix (4.15). Their denominators were theoretically equal and therefore they effectively represented the practical outcome of all four digital impedance matrix elements. Furthermore, the impedance matrix was derived in Chapter 4 from the admittance matrix, and thus to attempt to verify the digital impedance matrix would also intrinsically verify the digital admittance matrix.

The practical measurements were made with the sampling period fixed at 18.75 microseconds because the amplitude of limit cycle oscillations increased in rough proportion to the sample period, as shown later in Table 8.35.

7.2 MEASUREMENT TECHNIQUE

The frequency responses of digital input impedance z_{11} and forward transfer impedance z_{21} were measured using the standard technique described in Appendix G. The test arrangement is shown in Fig. 7.1. The amplifier transconductance g was set to $^1/_{75}$ siemens, and this is equivalent in magnitude to - 37.5 dB.

The circuit of the digital gyrator is shown in Fig. 7.2 and the relevant circuit parameters are listed in Table 7.1.

The apparent advantage of using this magnitude comparison method is that the absolute signal levels do not have to be known,

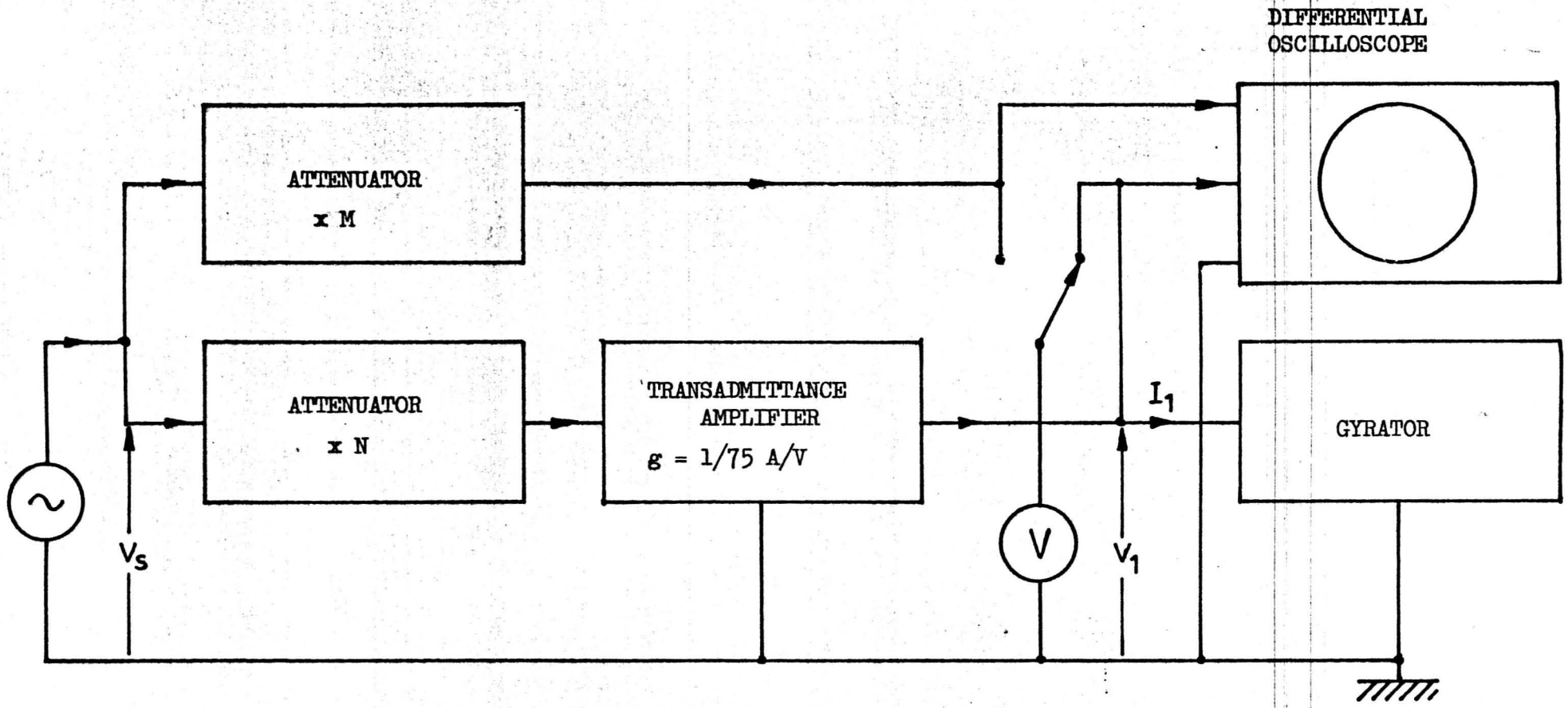


FIG. 7.1 z_{11} - Input Impedance Measurement

PARAMETER	PARAMETER MEANING	VALUE
f_s	Sampling frequency	26.6 KHz
T_s	Sampling period	18.75 microseconds
C_1	Port 1 shunt capacitor	9.6 microfarads
C_2	Port 2 shunt capacitor	10 microfarads
g_1, g_2	Transconductances	10 millisiemens
R_s	Port 1 shunt resistor	50, 100, 200, 300, 400, 500 ohms
k_1, k_2	Fractional delay	1

TABLE 7.1

Digital Gyrator Component Values

only the gains or losses of each stage in each limb. However, the presence of a finite dynamic range, 32 quantisation levels and limit cycle noise meant that the signal needed to be kept near the maximum possible level.

7.2.1 Magnitude Response

The magnitude response was obtained using equation (G.4) and with the terms defined in Fig. 7.1, thus:

$$\left| z \right| = N / Mg \qquad (7.1)$$

However, the attenuator values M and N were measured in decibels, and hence equation (7.1) may be rewritten:

$$\left| z \right| = 10^{(M_{dB} - N_{dB} - g_{dB}) / 20} \qquad (7.2)$$

where g_{dB} was set to - 37.5 dB.

7.2.2 Phase Response

The phase shift $\angle z$ was calculated directly from equation (G.10) thus:

$$\angle z = -2 \operatorname{Tan}^{-1} \left[\frac{\left| V_D \right|_{pp}}{\sqrt{32 \left| V_M \right|^2_{RMS} - \left| V_D \right|^2_{pp}}} \right] \qquad (7.3)$$

Thus in this case both a voltage difference and an absolute voltage had to be measured.

7.3 INPUT IMPEDANCE (z_{11}) OF DIGITAL GYRATOR

The frequency response of the input impedance z_{11} of the 2-port capacitively loaded digital gyrator was measured using the test arrangement shown in Fig. 7.1 at port 1 or the digital gyrator. The circuit of the digital gyrator is shown in Fig. 7.2 and the relevant circuit parameters are listed in Table 7.1

M_{dB}, N_{dB}, $\left| V_D \right|_{pp}$ and $\left| V_M \right|_{RMS}$ were measured and the values for $\left| z_{11} \right|$ and $\angle(z_{11})$ were calculated using the formulae in Section 7.2 and all are listed in Tables 7.2 to 7.7. Graphs 7.1 to 7.6 show the

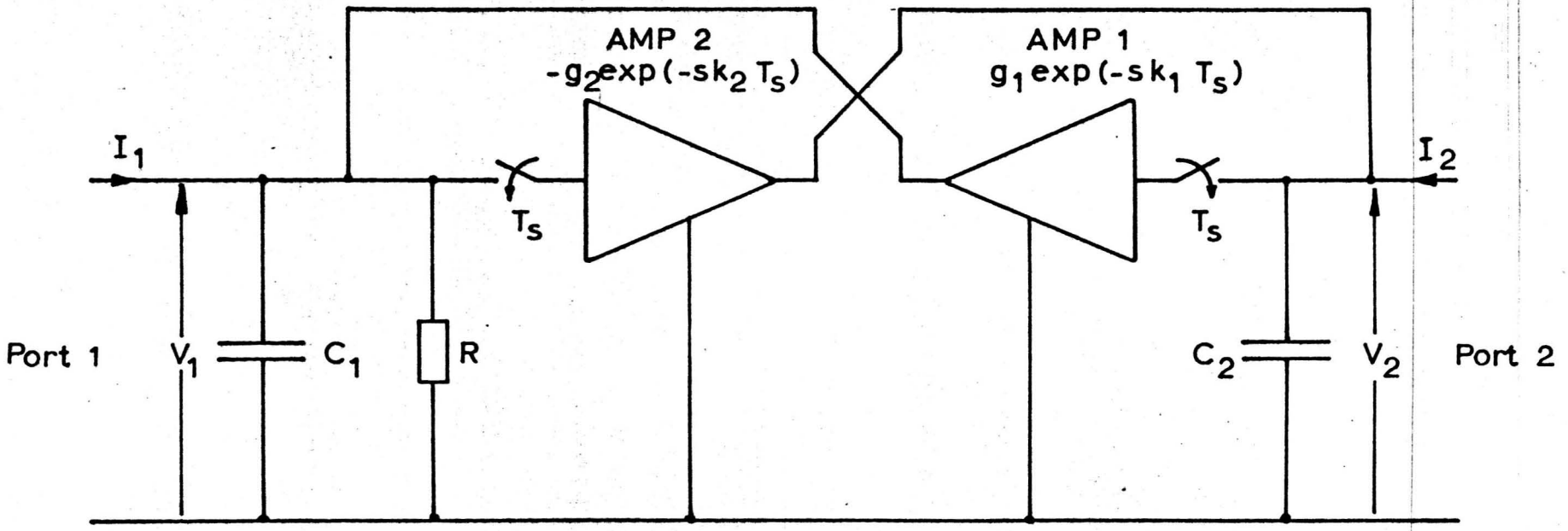


FIG. 7.2 Circuit of Digital Gyrator

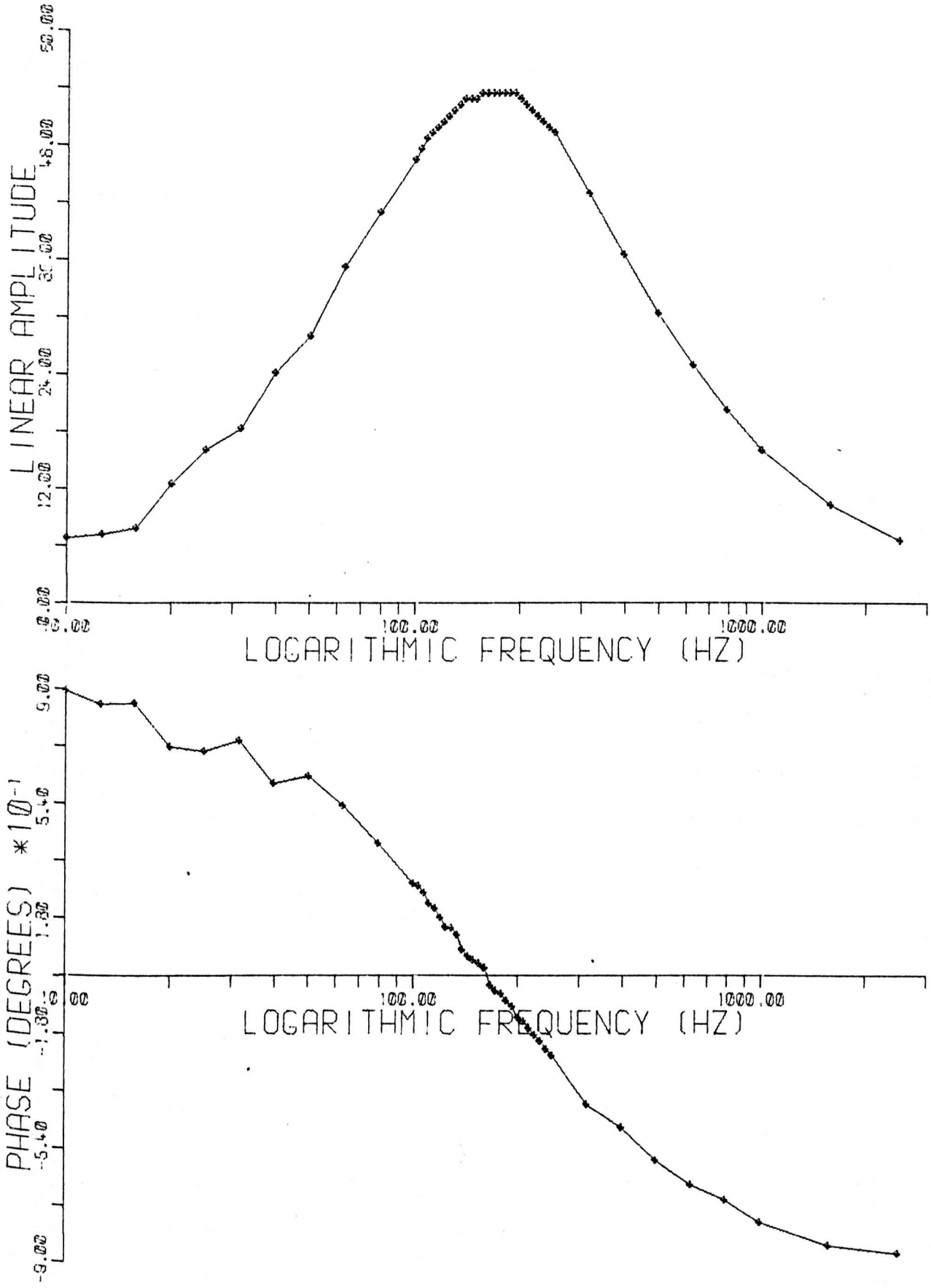
EXPERIMENTAL RESULTS TABLE 7.2

DIGITAL INPUT IMPEDANCE Z11

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	50.0 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-30.8	10.0	16.7	0.049	0.300	6.8400	90.0000
12.60	-30.3	10.0	17.2	0.052	0.200	7.2400	85.6697
15.80	-29.5	10.0	18.0	0.057	0.220	7.9400	86.0496
20.00	-25.5	10.0	22.0	0.090	0.300	12.5900	72.2100
25.10	-23.3	10.0	24.2	0.110	0.360	16.2200	70.7000
31.60	-22.2	10.0	25.3	0.132	0.400	18.4100	74.1200
39.80	-19.8	10.0	27.7	0.175	0.500	24.2700	60.6700
50.10	-18.5	10.0	29.0	0.203	0.600	28.1800	63.0000
63.10	-16.5	10.0	31.0	0.255	0.650	35.4800	53.5700
79.40	-15.2	10.0	32.3	0.298	0.600	41.2100	41.7000
100.00	-14.1	10.0	33.4	0.338	0.480	46.7700	29.0800
103.70	-13.9	10.0	33.6	0.345	0.480	47.8600	28.4800
107.60	-13.7	10.0	33.8	0.350	0.450	48.9800	26.2700
111.60	-13.6	10.0	33.9	0.356	0.400	49.5500	22.9100
115.80	-13.5	10.0	34.0	0.361	0.380	50.1200	21.4500
120.10	-13.4	10.0	34.1	0.366	0.330	50.7000	18.3400
124.60	-13.3	10.0	34.2	0.370	0.280	51.2900	15.3800
129.20	-13.2	10.0	34.3	0.375	0.280	51.8800	15.1700
134.10	-13.1	10.0	34.4	0.378	0.240	52.4800	12.8900
139.10	-13.0	10.0	34.5	0.381	0.160	53.0900	8.5100
144.30	-13.0	10.0	34.5	0.380	0.120	53.0900	6.4000
149.60	-13.0	10.0	34.5	0.383	0.100	53.0900	5.2900
155.20	-12.9	10.0	34.6	0.385	0.080	53.7000	4.2100
160.60	-12.9	10.0	34.6	0.388	0.050	53.7000	2.6100
161.00	-12.9	10.0	34.6	0.387	0.050	53.7000	2.6200
167.10	-12.9	10.0	34.6	0.385	0.050	53.7000	-2.6300
173.30	-12.9	10.0	34.6	0.382	0.080	53.7000	-4.2400
179.80	-12.9	10.0	34.6	0.386	0.100	53.7000	-5.2500
186.50	-12.9	10.0	34.6	0.385	0.140	53.7000	-7.3700
193.40	-12.9	10.0	34.6	0.385	0.175	53.7000	-9.2200
200.60	-13.0	10.0	34.5	0.381	0.240	53.0900	-12.7900
208.10	-13.1	10.0	34.4	0.379	0.260	52.4800	-13.9300
215.90	-13.2	10.0	34.3	0.376	0.300	51.8800	-16.2200
223.90	-13.3	10.0	34.2	0.370	0.330	51.2900	-18.1400
232.30	-13.4	10.0	34.1	0.367	0.360	50.7000	-19.9700
241.00	-13.5	10.0	34.0	0.363	0.400	50.1200	-22.4700

250.00	-13.6	10.0	33.9	0.358	0.430	49.5500	-24.5200
316.00	-14.8	10.0	32.7	0.311	0.600	43.1500	-39.8800
398.00	-16.2	10.0	31.3	0.265	0.600	36.7300	-47.1900
501.00	-17.8	10.0	29.7	0.221	0.600	30.5500	-57.3600
631.00	-19.5	10.0	28.0	0.181	0.550	25.1200	-64.9800
794.00	-11.3	0.0	26.2	0.463	0.500	20.4200	-69.8800
1000.00	-13.3	0.0	24.2	0.370	0.300	16.2200	-76.7900
1580.00	-17.1	0.0	20.4	0.237	0.900	10.4700	-84.3398
2510.00	-20.9	0.0	16.6	0.154	0.600	6.7600	-87.0595



Z11 R_s = 50 OHMS T_s = 18.75E-6 SECONDS

Measured Input Impedance GRAPH 7.1

EXPERIMENTAL RESULTS — TABLE 7.3

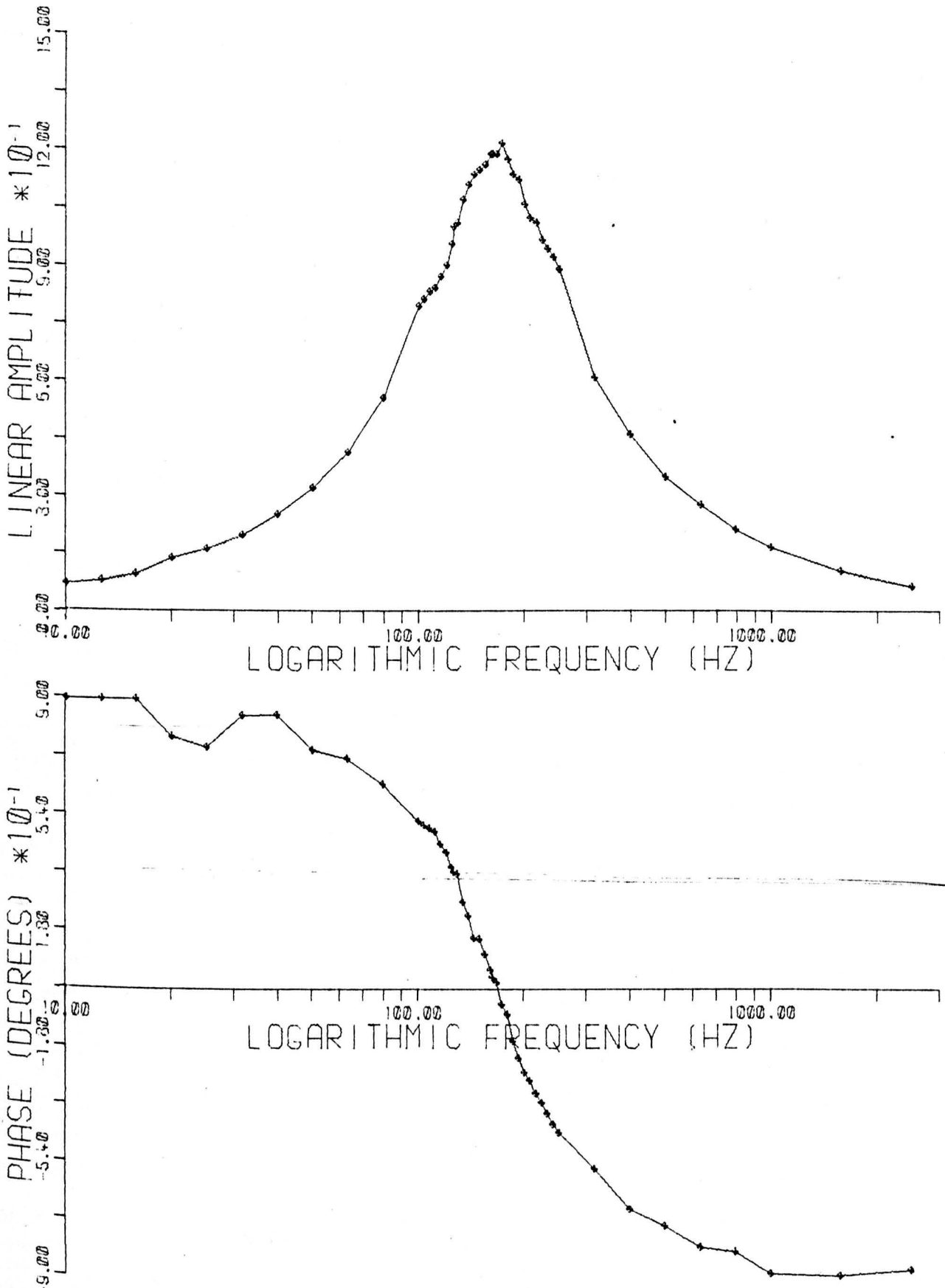
DIGITAL INPUT IMPEDANCE Z11

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	100.0 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-30.6	10.0	16.9	0.050	0.350	7.0000	90.0000
12.60	-29.5	10.0	18.0	0.057	0.400	7.9400	90.0000
15.80	-27.8	10.0	19.7	0.069	0.300	9.6600	90.0000
20.00	-24.7	10.0	22.8	0.098	0.350	13.8000	78.2999
25.10	-23.3	10.0	24.2	0.116	0.400	16.2200	75.1201
31.60	-21.5	10.0	26.0	0.144	0.550	19.9500	84.9397
39.80	-19.4	10.0	28.1	0.183	0.700	25.4100	85.0897
50.10	-17.3	10.0	30.2	0.234	0.800	32.3600	74.3700
63.10	-15.1	10.0	32.4	0.302	1.000	41.6900	71.6600
79.40	-12.6	10.0	34.9	0.402	1.200	55.5900	63.7000
100.00	-19.5	20.0	38.0	0.181	0.450	79.4300	52.1400
103.70	-19.3	20.0	38.2	0.185	0.450	81.2800	50.9300
107.60	-19.1	20.0	38.4	0.188	0.450	83.1800	50.0700
111.60	-19.0	20.0	38.5	0.192	0.450	84.1400	48.9500
115.80	-18.7	20.0	38.8	0.199	0.430	87.1000	44.9100
120.10	-18.4	20.0	39.1	0.205	0.420	90.1600	42.4700
124.60	-17.9	20.0	39.6	0.218	0.400	95.5000	37.8500
126.00	-17.5	20.0	40.0	0.227	0.400	100.0000	36.3000
129.20	-17.4	20.0	40.1	0.230	0.400	101.1599	35.8100
134.10	-16.9	20.0	40.6	0.242	0.320	107.1500	27.0400
139.10	-16.6	20.0	40.9	0.252	0.280	110.9200	22.6600
144.30	-16.4	20.0	41.1	0.258	0.200	113.5000	15.7500
149.60	-16.3	20.0	41.2	0.260	0.200	114.8200	15.6300
155.20	-16.2	20.0	41.3	0.265	0.140	116.1400	10.7200
161.00	-16.0	20.0	41.5	0.270	0.080	118.8500	6.0000
163.00	-16.0	20.0	41.5	0.271	0.050	118.8499	3.7400
167.10	-16.0	20.0	41.5	0.269	0.030	118.8500	2.2600
173.30	-15.8	20.0	41.5	0.274	0.060	118.8500	-4.4400
179.80	-16.1	20.0	41.4	0.265	0.100	117.4900	-7.6500
186.50	-16.4	20.0	41.1	0.259	0.200	113.4999	-15.6900
193.40	-16.5	20.0	41.0	0.254	0.260	112.1999	-20.8500
200.60	-17.0	20.0	40.5	0.241	0.300	105.9300	-25.4200
208.10	-17.3	20.0	40.2	0.234	0.320	102.3300	-27.9800
215.90	-17.4	20.0	40.1	0.232	0.360	101.1599	-31.8400
223.90	-17.8	20.0	39.7	0.219	0.370	96.6100	-34.7500
232.30	-18.0	20.0	39.5	0.216	0.400	94.4100	-38.2200
241.00	-18.2	20.0	39.3	0.210	0.420	92.2600	-41.4100
250.00	-18.5	20.0	39.0	0.203	0.430	89.1300	-43.9800

SOME PAGES BOUND INTO/CLOSE TO SPINE.

316.00	-11.8	10.0	35.7	0.439	1.150	60.9500	-55.1700
398.00	-14.2	10.0	33.3	0.333	1.050	46.2400	-67.7500
501.00	-16.6	10.0	30.9	0.253	0.850	35.0800	-72.8700
631.00	-18.6	10.0	28.9	0.199	0.720	27.8600	-79.5200
794.00	-10.9	0.0	26.6	0.490	1.800	21.3800	-80.9899
1000.00	-13.0	0.0	24.5	0.382	1.500	16.7900	-87.9193
1580.00	-17.0	0.0	20.5	0.240	0.950	10.5900	-88.8090
2510.00	-20.9	0.0	16.6	0.154	0.600	6.7600	-87.0595



Z11 R_s = 100 OHMS T_s = 18.75E-6 SECONDS

Measured Input Impedance GRAPH 7.2

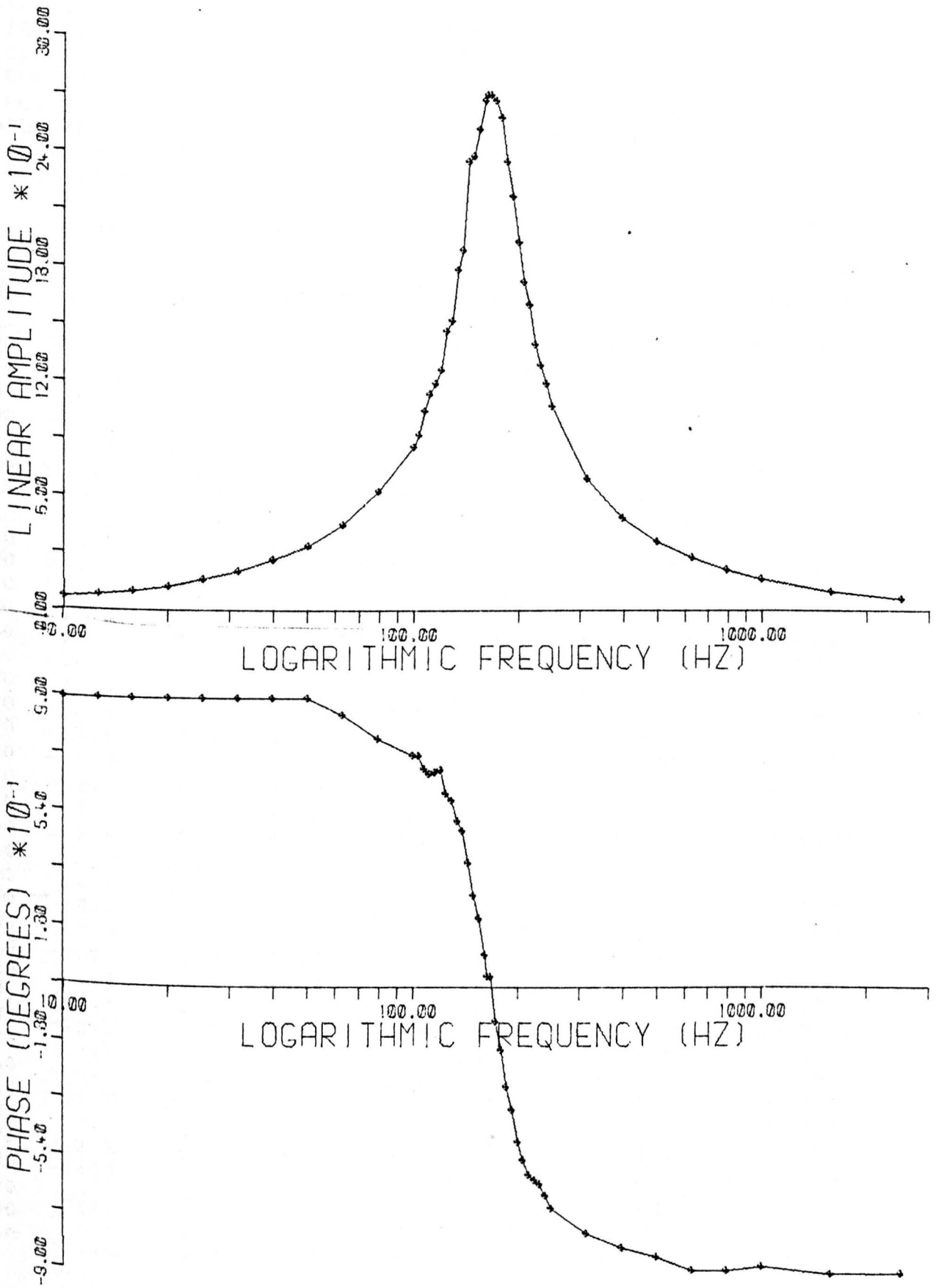
EXPERIMENTAL RESULTS TABLE 7.4

DIGITAL INPUT IMPEDANCE Z11

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	200 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	|Z11| DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-30.3	10.0	17.2	0.052	0.000	7.2400	90.0000
12.60	-28.8	10.0	18.7	0.062	0.000	8.6100	90.0000
15.80	-27.2	10.0	20.3	0.074	0.000	10.3500	90.0000
20.00	-25.2	10.0	22.3	0.093	0.350	13.0300	90.0000
25.10	-23.0	10.0	24.5	0.120	0.550	16.7900	90.0000
31.60	-18.9	10.0	26.5	0.151	0.650	21.1300	90.0000
39.80	-16.8	10.0	28.6	0.192	0.800	26.9200	90.0000
50.10	-14.3	10.0	30.7	0.248	1.000	34.2800	90.0000
63.10	-11.5	10.0	33.2	0.328	1.250	45.7100	84.7098
79.40	-18.8	10.0	36.0	0.452	1.600	63.1000	77.4800
100.00	-18.8	20.0	38.7	0.195	0.650	86.1000	72.2100
103.70	-18.2	20.0	39.3	0.210	0.700	92.2600	72.2100
107.60	-17.1	20.0	40.4	0.237	0.750	104.7100	68.0300
111.60	-16.4	20.0	41.1	0.258	0.800	113.5000	66.4800
115.80	-16.0	20.0	41.5	0.269	0.840	118.8499	67.0100
120.10	-15.5	20.0	42.0	0.286	0.900	125.8900	67.6000
124.60	-14.2	20.0	43.3	0.334	0.950	146.2200	60.3700
129.20	-13.9	20.0	43.6	0.346	0.950	151.3600	58.0700
134.10	-12.5	20.0	45.0	0.406	1.000	177.8300	51.6200
139.10	-12.0	20.0	45.5	0.429	1.000	188.3599	48.6700
144.30	-11.1	20.0	47.4	0.481	0.900	234.4199	38.6300
149.60	-20.0	10.0	47.5	0.540	0.750	237.1400	28.4300
155.20	-9.5	20.0	48.0	0.570	0.600	251.1900	21.4500
161.00	-9.0	20.0	48.5	0.602	0.300	266.0700	10.1100
163.30	-8.9	20.0	48.6	0.611	0.100	269.1500	3.5200
167.10	-8.9	20.0	48.6	0.609	0.110	269.1500	3.6600
173.30	-9.0	20.0	48.5	0.602	0.300	266.0700	-10.1100
179.80	-9.3	20.0	48.2	0.586	0.550	257.0400	-19.1000
186.50	-10.1	20.0	47.4	0.535	0.800	234.4199	-30.6600
193.40	-10.8	20.0	46.7	0.492	0.900	216.2700	-37.7300
200.60	-11.8	20.0	45.7	0.437	1.000	192.7499	-47.7200
208.10	-12.8	20.0	44.7	0.394	1.000	171.7899	-53.3200
215.90	-13.4	20.0	44.1	0.365	1.000	160.3200	-57.9400
223.90	-14.6	20.0	42.9	0.320	0.900	139.6400	-59.6300
232.30	-15.3	20.0	42.2	0.293	0.840	128.8200	-60.9000
241.00	-16.0	20.0	41.5	0.272	0.820	118.8500	-64.4100
250.00	-16.9	20.0	40.6	0.245	0.780	107.1500	-68.5000
316.00	-10.7	10.0	36.8	0.500	1.750	69.1800	-76.4500
398.00	-13.7	10.0	33.8	0.354	1.300	48.9800	-80.9599

501.00	-16.2	10.0	31.3	0.265	1.000	36.7300	-83.6798
631.00	-18.5	10.0	29.0	0.204	0.800	28.1800	-87.7693
794.00	-10.7	0.0	26.8	0.497	1.950	21.8800	-87.8293
1000.00	-12.9	0.0	24.6	0.387	1.500	16.9800	-86.4995
1580.00	-17.0	0.0	20.5	0.242	0.960	10.5900	-89.0589
2510.00	-20.8	0.0	16.7	0.156	0.620	6.8400	-89.2689



Z11 R_s = 200 OHMS T_s = 18.75E-6 SECONDS

Measured Input Impedance GRAPH **7.3**

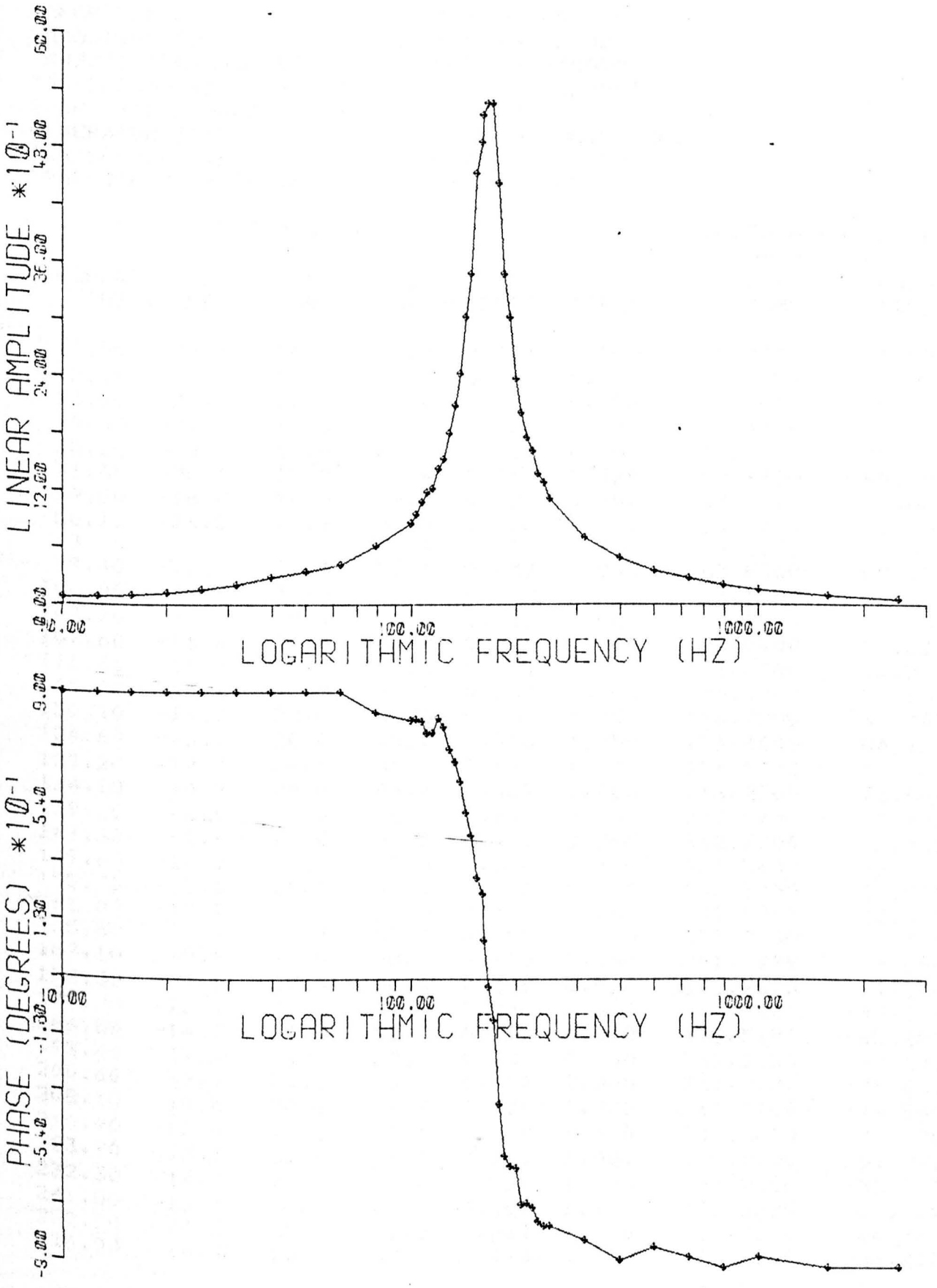
EXPERIMENTAL RESULTS — TABLE 7.5

DIGITAL INPUT IMPEDANCE Z11

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	300 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-29.2	10.0	18.3	0.059	0.000	8.2200	90.0000
12.60	-28.1	10.0	19.4	0.067	0.000	9.3300	90.0000
15.80	-26.6	10.0	20.9	0.079	0.000	11.0900	90.0000
20.00	-24.9	10.0	22.6	0.096	0.000	13.4900	90.0000
25.10	-23.1	10.0	24.4	0.119	0.000	16.6000	90.0000
31.60	-20.9	10.0	26.6	0.154	0.700	21.3800	90.0000
39.80	-18.1	10.0	29.4	0.191	0.800	29.5100	90.0000
50.10	-16.5	10.0	31.0	0.255	1.050	35.4800	90.0000
63.10	-14.9	10.0	32.6	0.319	1.300	42.6600	90.0000
79.40	-11.6	10.0	35.9	0.451	1.700	62.3700	83.5698
100.00	-18.8	20.0	38.7	0.196	0.720	86.1000	80.9900
103.70	-17.9	20.0	39.6	0.217	0.800	95.5000	81.3399
107.60	-16.8	20.0	40.7	0.246	0.900	108.3900	80.5899
111.60	-16.0	20.0	41.5	0.270	0.950	118.8500	76.9200
115.80	-15.7	20.0	41.8	0.284	1.000	123.0300	76.9900
120.10	-14.4	20.0	43.1	0.325	1.200	142.8900	81.4899
124.60	-13.8	20.0	43.7	0.348	1.250	153.1100	78.8399
129.20	-12.4	20.0	45.1	0.407	1.350	179.8899	71.8000
134.10	-11.1	20.0	46.4	0.475	1.500	208.9300	67.8700
139.10	-9.8	20.0	47.7	0.552	1.600	242.6599	61.6500
144.30	-7.9	20.0	49.6	0.688	1.700	302.0000	51.8000
149.60	-6.7	20.0	50.8	0.790	1.700	346.7400	44.7200
155.20	-14.4	30.0	53.1	0.325	0.500	451.8600	31.5600
161.00	-13.8	30.0	53.7	0.347	0.450	484.1699	26.5100
162.60	-13.3	30.0	54.2	0.370	0.220	512.8600	12.0700
167.10	-13.1	30.0	54.4	0.377	0.040	524.8099	-2.1500
173.30	-13.1	30.0	54.4	0.375	0.230	524.8101	-12.4500
179.80	-14.6	30.0	52.9	0.318	0.600	441.5700	-38.9700
186.50	-16.7	30.0	50.8	0.249	0.650	346.7400	-54.9600
193.40	-17.9	30.0	49.6	0.218	0.600	302.0000	-58.2300
200.60	-10.0	20.0	47.5	0.541	1.500	237.1400	-58.7000
208.10	-11.4	20.0	46.1	0.461	1.500	201.8400	-70.2300
215.90	-12.6	20.0	44.9	0.402	1.300	175.7900	-69.7300
223.90	-13.3	20.0	44.2	0.371	1.220	162.1800	-71.0900
232.30	-14.7	20.0	42.8	0.317	1.100	138.0399	-75.6700
241.00	-15.3	20.0	42.2	0.295	1.040	128.8200	-77.1000
250.00	-16.5	20.0	41.0	0.256	0.900	112.2000	-76.8500
316.00	-10.4	10.0	37.1	0.514	1.900	71.6100	-81.6099

398.00	-13.6	10.0	33.9	0.357	1.400	49.5500	-87.7693
501.00	-16.2	10.0	31.3	0.265	1.000	36.7300	-83.6798
631.00	-18.4	10.0	29.1	0.206	0.800	28.5100	-86.7095
794.00	-10.7	0.0	26.8	0.498	2.000	21.8800	-90.0000
1000.00	-12.9	0.0	24.6	0.387	1.500	16.9800	-86.4995
1580.00	-17.0	0.0	20.5	0.241	1.000	10.5900	-90.0000
2510.00	-20.7	0.0	16.8	0.158	0.650	6.9200	-90.0000



Z11 R_s = 300 OHMS T_s = 18.75E-6 SECONDS

Measured Input Impedance GRAPH **7.4**

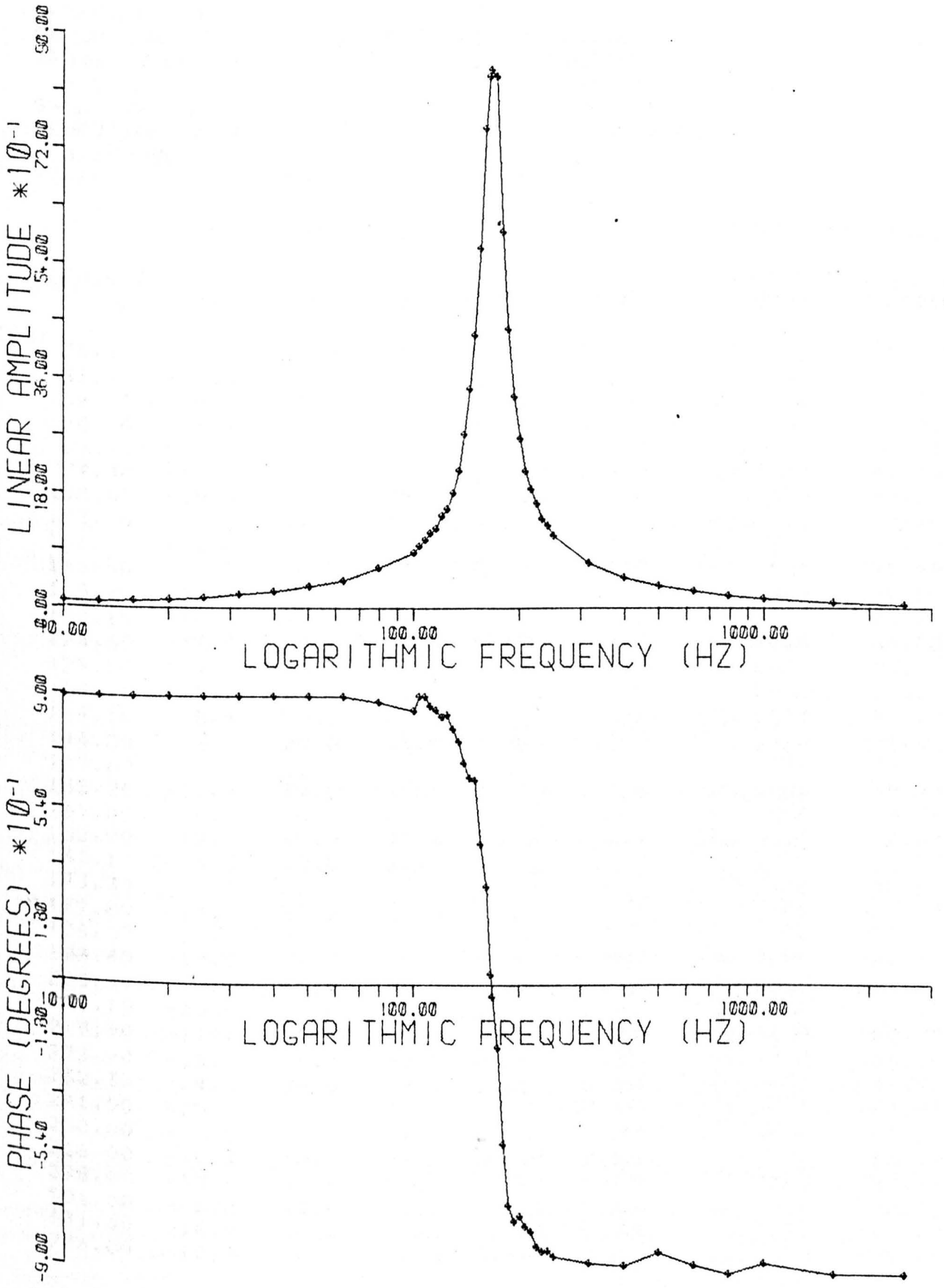
EXPERIMENTAL RESULTS TABLE 7.6

DIGITAL INPUT IMPEDANCE Z11

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	400 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-27.1	10.0	20.4	0.075	0.000	10.4700	90.0000
12.60	-26.5	10.0	21.0	0.080	0.000	11.2200	90.0000
15.80	-25.7	10.0	21.8	0.088	0.000	12.3000	90.0000
20.00	-24.3	10.0	23.2	0.103	0.000	14.4500	90.0000
25.10	-22.8	10.0	24.7	0.123	0.000	17.1800	90.0000
31.60	-20.6	10.0	26.9	0.158	0.800	22.1300	90.0000
39.80	-18.8	10.0	28.7	0.194	0.900	27.2300	90.0000
50.10	-16.5	10.0	31.0	0.255	1.100	35.4800	90.0000
63.10	-14.6	10.0	32.9	0.318	1.300	44.1600	90.0000
79.40	-11.4	10.0	36.1	0.457	1.800	63.8300	88.2592
100.00	-18.7	20.0	38.8	0.198	0.760	87.1000	85.4597
103.70	-17.7	20.0	39.8	0.223	0.900	97.7200	90.0000
107.60	-16.8	20.0	40.7	0.245	1.000	108.3900	90.0000
111.60	-16.0	20.0	41.5	0.270	1.050	118.8500	86.8595
115.80	-15.5	20.0	42.0	0.286	1.100	125.8900	85.6697
120.10	-14.2	20.0	43.3	0.332	1.250	146.2200	83.4498
124.60	-13.6	20.0	43.9	0.356	1.350	156.6800	84.1898
129.20	-12.3	20.0	45.2	0.414	1.500	181.9700	79.6600
134.10	-10.8	20.0	46.7	0.489	1.700	216.2700	75.8400
139.10	-8.8	20.0	48.7	0.623	2.000	272.2700	69.1500
144.30	-6.8	20.0	50.7	0.779	2.350	342.7700	64.4500
149.60	-14.9	30.0	52.6	0.308	0.920	426.5800	63.7500
155.20	-12.5	30.0	55.0	0.405	0.850	562.3400	43.5600
161.00	-10.0	30.0	57.5	0.540	0.800	749.8899	30.3600
165.60	-9.1	30.0	58.4	0.600	0.100	831.7600	3.3800
167.10	-9.0	30.0	58.5	0.600	0.100	841.3999	-3.3800
173.30	-9.1	30.0	58.4	0.595	0.560	831.7600	-19.1500
179.80	-12.1	30.0	55.4	0.425	1.000	588.8400	-49.1600
186.50	-14.7	30.0	52.8	0.315	1.000	436.5199	-68.2800
193.40	-17.1	30.0	50.4	0.237	0.800	331.1300	-73.2700
200.60	-9.0	20.0	48.5	0.604	2.000	266.0700	-71.6600
208.10	-10.8	20.0	46.7	0.494	1.700	216.2700	-74.9400
215.90	-12.0	20.0	45.5	0.428	1.500	188.3600	-76.5700
223.90	-13.1	20.0	44.4	0.380	1.400	165.9600	-81.2799
232.30	-14.5	20.0	43.0	0.321	1.200	141.2500	-82.7298
241.00	-15.1	20.0	42.4	0.300	1.120	131.8299	-82.5899
250.00	-16.3	20.0	41.2	0.263	1.000	114.8200	-84.4698
316.00	-10.4	10.0	37.1	0.516	2.000	71.6100	-86.4995

398.00	-13.6	10.0	33.9	0.358	1.400	49.5500	-87.4694
501.00	-16.2	10.0	31.3	0.266	1.000	36.7300	-83.2998
631.00	-18.4	10.0	29.1	0.205	0.800	28.5100	-87.2394
794.00	-10.7	0.0	26.8	0.498	2.000	21.8800	-90.0000
1000.00	-12.9	0.0	24.6	0.387	1.500	16.9800	-86.4995
1580.00	-17.0	0.0	20.5	0.242	1.000	10.5900	-90.0000
2510.00	-20.6	0.0	16.9	0.159	0.000	7.0000	-90.0000



Z11 R_s = 400 OHMS T_s = 18.75E-6 SECONDS

Measured Input Impedance GRAPH **7.5**

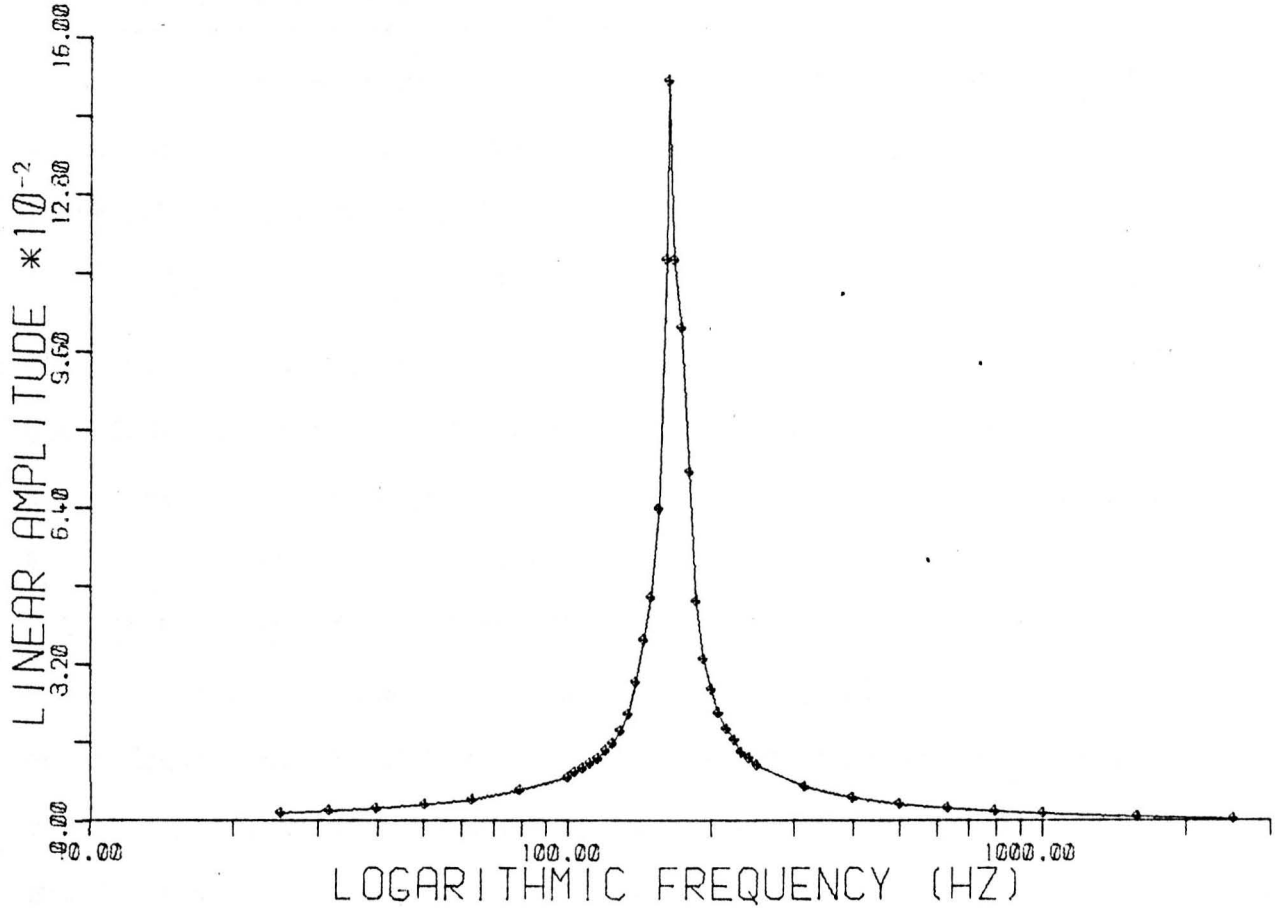
EXPERIMENTAL RESULTS TABLE 7.7

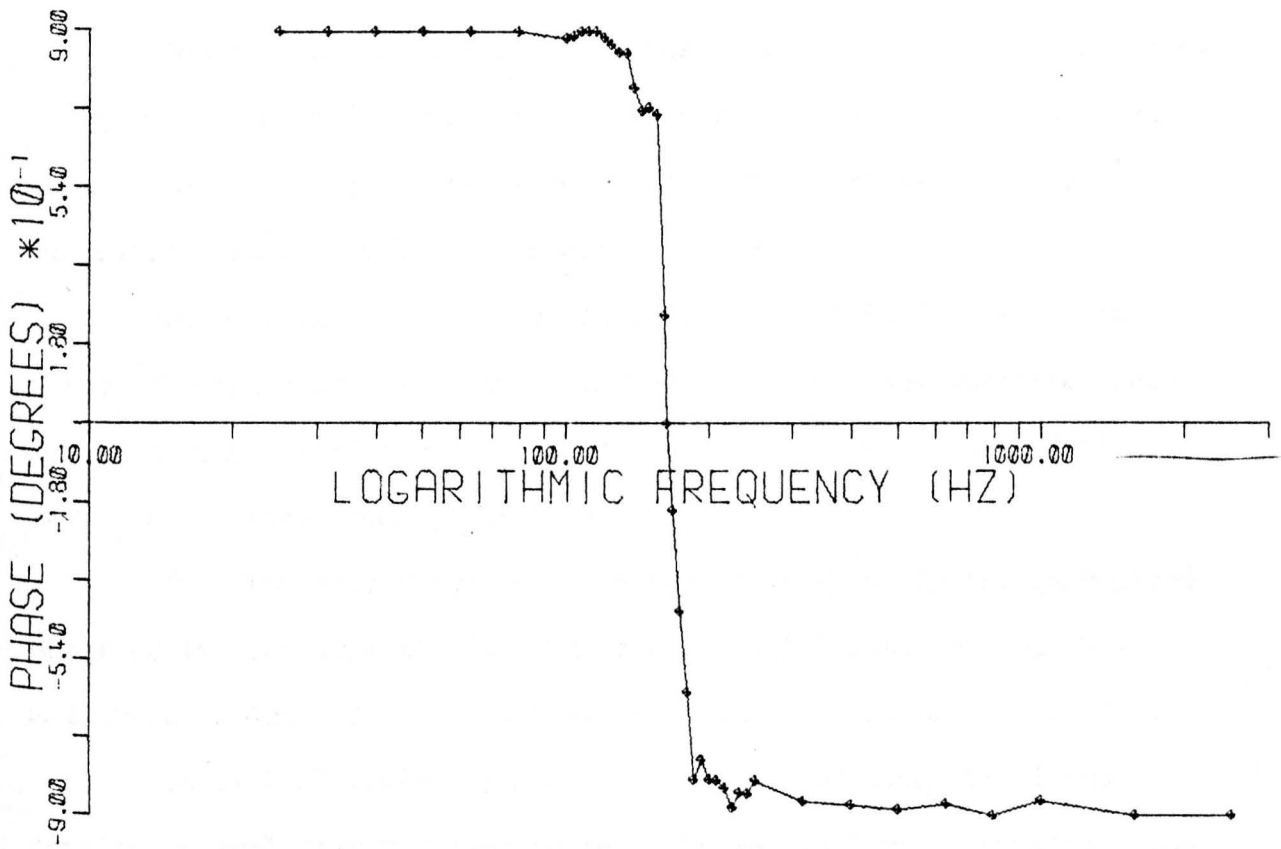
DIGITAL INPUT IMPEDANCE Z11

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	500 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
25.10	-22.4	10.0	25.1	0.129	0.000	17.9900	90.0000
31.60	-20.4	10.0	27.1	0.162	0.000	22.6500	90.0000
39.80	-18.7	10.0	28.8	0.197	1.000	27.5400	90.0000
50.10	-16.4	10.0	31.1	0.258	1.200	35.8900	90.0000
63.10	-14.6	10.0	32.9	0.319	1.300	44.1600	90.0000
79.40	-11.3	10.0	36.2	0.467	1.950	64.5700	90.0000
100.00	-18.5	20.0	39.9	0.203	0.800	89.1300	88.3192
103.70	-17.5	20.0	40.0	0.227	0.900	100.0000	88.9890
107.60	-16.8	20.0	40.7	0.246	1.000	108.3900	90.0000
111.60	-16.0	20.0	41.5	0.271	1.100	118.8500	90.0000
115.80	-15.4	20.0	42.1	0.290	1.200	127.3500	90.0000
120.10	-14.3	20.0	43.2	0.329	1.300	144.5400	88.6191
124.60	-13.5	20.0	44.0	0.360	1.400	158.4900	86.8595
129.20	-12.2	20.0	45.3	0.418	1.600	184.0800	85.1697
134.10	-10.7	20.0	46.8	0.497	1.900	218.7800	85.0297
139.10	-8.4	20.0	49.1	0.651	2.300	285.1000	77.0300
144.30	-6.1	20.0	51.4	0.843	2.800	371.5400	71.9100
149.60	-14.3	30.0	53.2	0.329	1.100	457.0899	72.4600
155.20	-11.4	30.0	56.1	0.457	1.500	638.2600	70.9300
161.00	-6.3	30.0	61.2	0.830	1.000	1148.1500	24.5900
162.90	-13.9	40.0	63.6	0.346	1.000	1513.5600	0.0000
167.10	-6.3	30.0	61.2	0.830	0.800	1148.1500	-19.6200
173.30	-7.4	30.0	60.1	0.726	1.500	1011.5800	-42.8400
179.80	-10.4	30.0	57.1	0.519	1.500	716.1400	-61.4500
186.50	-14.4	30.0	53.1	0.324	1.200	451.8601	-81.7999
193.40	-17.0	30.0	50.5	0.241	0.850	334.9700	-77.1400
200.60	-8.8	20.0	48.7	0.622	2.300	272.2700	-81.6399
208.10	-10.5	20.0	47.0	0.512	1.900	223.8700	-81.9899
215.90	-11.9	20.0	45.6	0.437	1.650	190.5500	-83.7398
223.90	-13.0	20.0	44.5	0.381	1.500	167.8800	-88.2092
232.30	-14.4	20.0	43.1	0.325	1.240	142.8900	-84.8297
241.00	-15.1	20.0	42.4	0.301	1.150	131.8300	-84.9697
250.00	-16.1	20.0	41.4	0.269	1.000	117.4900	-82.1699
316.00	-10.4	10.0	37.1	0.515	2.000	71.6100	-86.7095
398.00	-13.6	10.0	33.9	0.358	1.400	49.5500	-87.4694
501.00	-16.2	10.0	31.3	0.266	1.050	36.7300	-88.4991
631.00	-18.4	10.0	29.1	0.205	0.800	28.5100	-87.2394
794.00	-10.7	0.0	26.8	0.498	2.050	21.8800	-90.0000

1000.00	-12.9	0.0	24.6	0.387	1.500	16.9800	-86.4995
1580.00	-16.9	0.0	20.6	0.244	1.000	10.7200	-90.0000
2510.00	-20.6	0.0	16.9	0.160	0.000	7.0000	-90.0000





Z11 R_s = 500 OHMS T_s = 18.75E-6 SECONDS

Measured Input Impedance GRAPH 7.6

frequency responses for the magnitude and phase for all 6 values of the shunt resistance R_s with the sampling period set to 18.75 microseconds.

7.4 FORWARD TRANSIMPEDANCE (z_{21}) OF DIGITAL GYRATOR

The frequency response of the forward transimpedance z_{21} of the 2-port capacitively loaded digital gyrator was measured in a manner identical to z_{11}, but at port 2 of the digital gyrator. See Fig. 7.3.

M_{dB}, N_{dB}, $\left|V_D\right|_{pp}$ and $\left|V_M\right|_{RMS}$ were measured and the values for $\left|z_{21}\right|$ and $\measuredangle(z_{21})$ were calculated using the above formulae, and are listed in Tables 7.8 to 7.13. Graphs 7.7 to 7.12 show the gain and phase frequency responses for all 6 values of the shunt resistance R_s with the sampling period set to 18.75 microseconds.

7.5 INPUT TRANSADMITTANCE AMPLIFIER

The input signal was coupled to the digital gyrator by way of an analogue transadmittance amplifer in order to measure z_{11} and z_{21} of the digital impedance matrix by injecting a known current. This amplifier was described in Section 6.8.

Because this amplifier was necessarily in the main signal path, its gain and phase frequency responses were also determined to prove that these had a negligible effect on the measurements taken because the output current could not be directly monitored.

The circuit of this amplifier is repeated in Fig. 7.4, the component values used are listed in Table 7.14 and the amplifier parameters in Table 7.15 for the response measured. The transconductance may be calculated from equation (6.26).

The frequency response was measured using a digital gain-phase meter (Solartron type 1170). The results are listed in Table 7.16 and shown in Graph 7.13. The test arrangement is shown in Fig. 7.5.

Table 7.17 lists the resistor values used when this transadmittance amplifier was used to test the 2-port digital gyrator. The

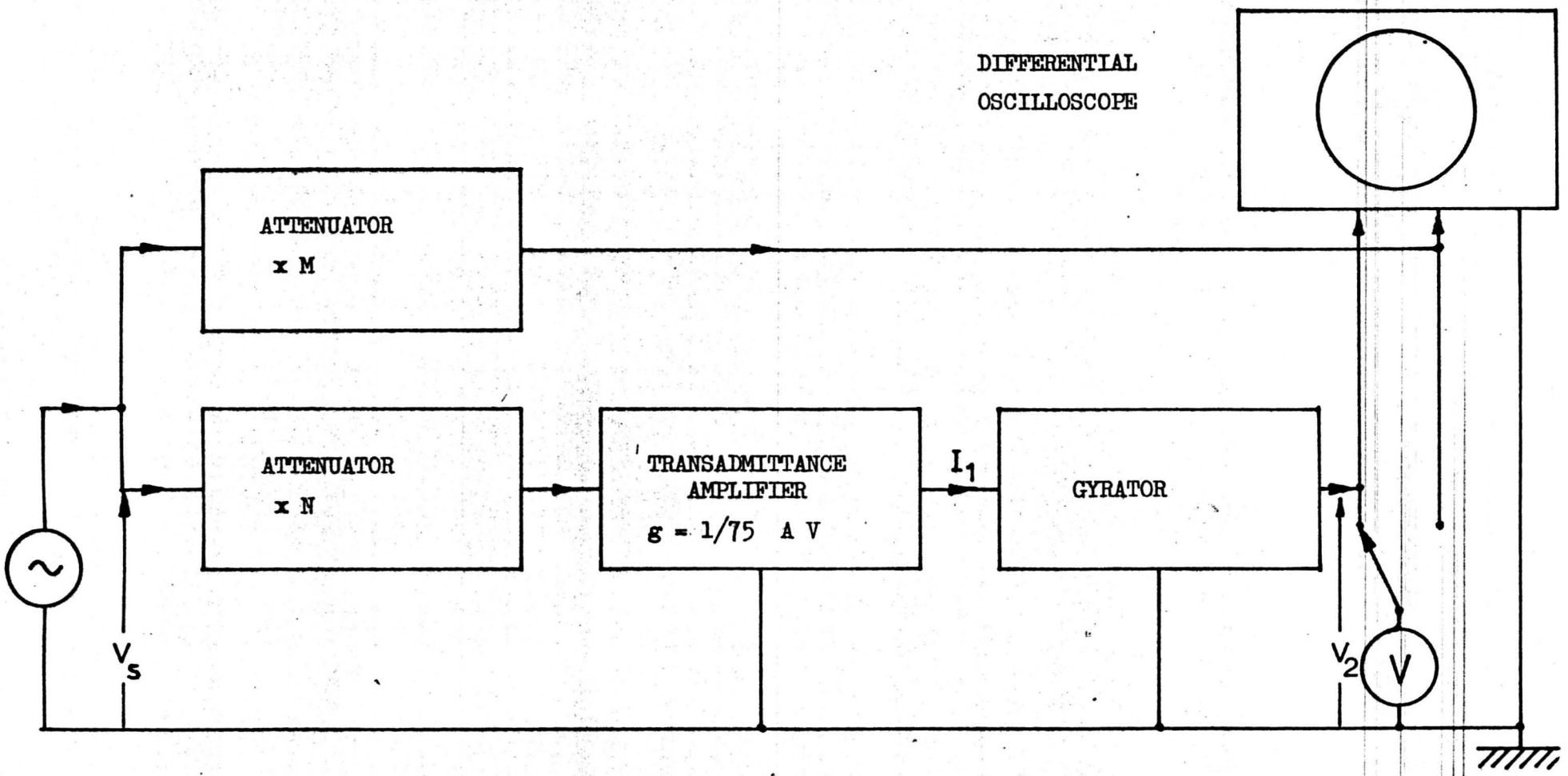


FIG. 7.3 z_{21} - Forward Transimpedance Measurement

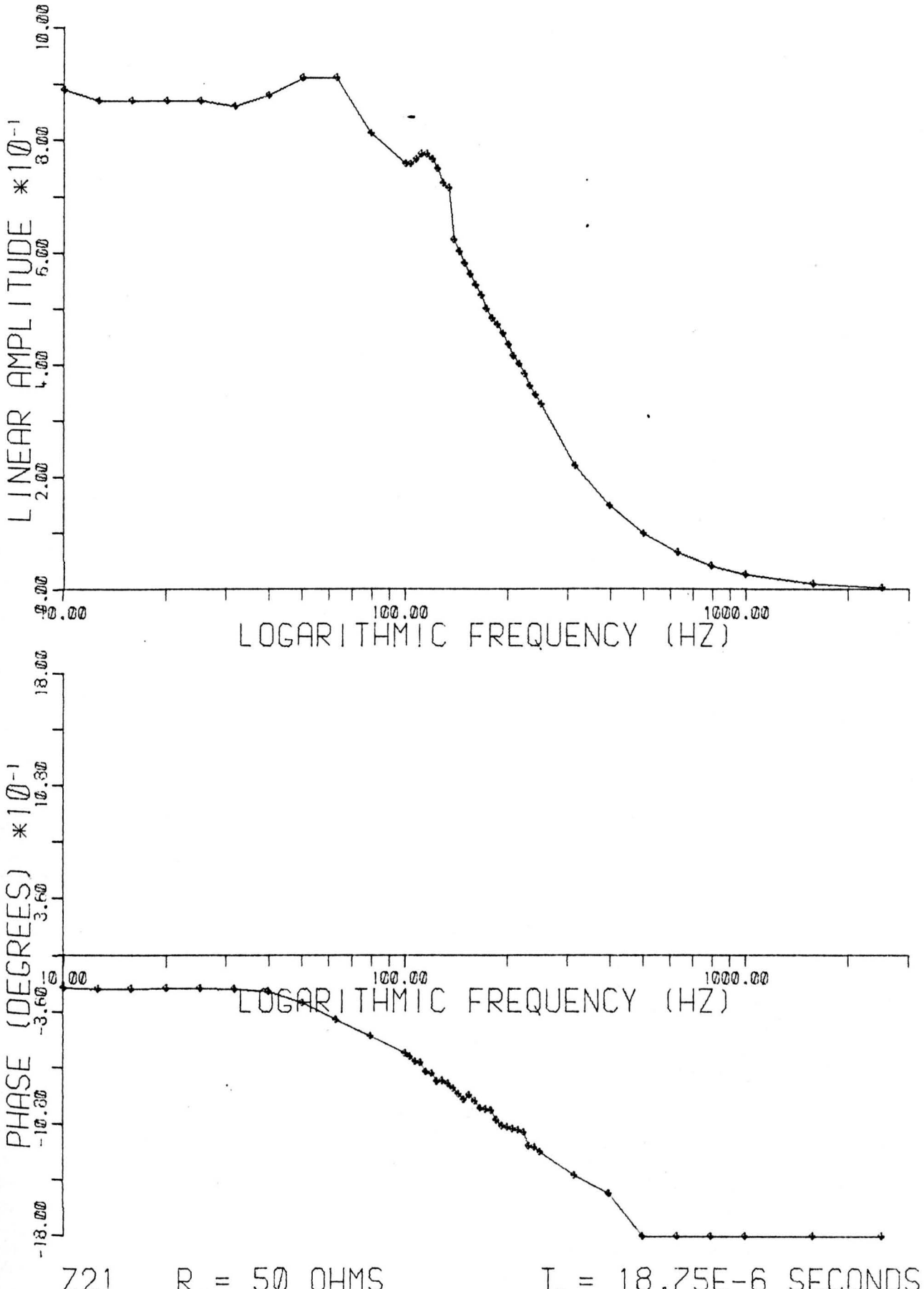
EXPERIMENTAL RESULTS TABLE 7.8

DIGITAL FORWARD TRANSFER IMPEDANCE Z21

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	50 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	|Z11| DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-18.5	20.0	39.0	0.201	0.200	89.1300	-20.2600
12.60	-18.7	20.0	38.8	0.195	0.200	87.1000	-20.8900
15.80	-18.7	20.0	38.8	0.196	0.200	87.1000	-20.7800
20.00	-18.7	20.0	38.8	0.199	0.200	87.1000	-20.4700
25.10	-18.7	20.0	38.8	0.198	0.200	87.1000	-20.5700
31.60	-18.8	20.0	38.7	0.194	0.200	86.1000	-21.0000
39.80	-18.6	20.0	38.9	0.200	0.220	88.1000	-22.4300
50.10	-18.3	20.0	39.2	0.208	0.300	91.2000	-29.5400
63.10	-18.3	20.0	39.2	0.206	0.400	91.2000	-40.1500
79.40	-19.3	20.0	38.2	0.185	0.450	81.2800	-50.9300
100.00	-19.9	20.0	37.6	0.172	0.500	75.8600	-61.8500
103.70	-19.9	20.0	37.6	0.173	0.520	75.8600	-64.1900
107.60	-19.8	20.0	37.7	0.175	0.550	76.7400	-67.5000
111.60	-19.7	20.0	37.8	0.177	0.560	77.6200	-68.0100
115.80	-19.7	20.0	37.8	0.176	0.600	77.6200	-74.1200
120.10	-19.8	20.0	37.7	0.174	0.600	76.7400	-75.1200
124.60	-20.0	20.0	37.5	0.170	0.620	74.9900	-80.2899
129.20	-20.3	20.0	37.2	0.165	0.600	72.4400	-80.0099
134.10	-20.4	20.0	37.1	0.162	0.600	71.6100	-81.7999
139.10	-11.6	10.0	35.9	0.447	1.700	62.3700	-84.4898
144.30	-11.9	10.0	35.6	0.432	1.700	60.2600	-88.1592
149.60	-12.2	10.0	35.3	0.418	1.700	58.2100	-91.9408
155.20	-12.5	10.0	35.0	0.404	1.600	56.2300	-88.8690
161.00	-12.8	10.0	34.7	0.390	1.600	54.3300	-92.9805
167.10	-13.1	10.0	34.4	0.376	1.600	52.4800	-97.5702
173.30	-13.5	10.0	34.0	0.362	1.550	50.1200	-98.3901
179.80	-13.8	10.0	33.7	0.349	1.500	48.4200	-98.8901
186.50	-14.0	10.0	33.5	0.334	1.500	47.3200	-105.1000
193.40	-14.3	10.0	33.2	0.326	1.500	45.7100	-108.8600
200.60	-14.7	10.0	32.8	0.314	1.450	43.6500	-109.4400
208.10	-15.1	10.0	32.4	0.301	1.400	41.6900	-110.6100
215.90	-15.4	10.0	32.1	0.289	1.350	40.2700	-111.3300
223.90	-15.8	10.0	31.7	0.276	1.300	38.4600	-112.7400
232.30	-16.3	10.0	31.2	0.263	1.300	36.3100	-121.8099
241.00	-16.7	10.0	30.8	0.252	1.250	34.6700	-122.5300
250.00	-17.1	10.0	30.4	0.239	1.200	33.1100	-125.1399
316.00	-10.6	0.0	26.9	0.507	2.700	22.1300	-140.5799
398.00	-13.9	0.0	23.6	0.346	1.900	15.1400	-152.2099

501.00	-17.4	0.0	20.1	0.231	3.250	10.1200	-179.9994
631.00	-21.0	0.0	16.5	0.153	0.900	6.6800	-179.9991
794.00	-24.9	0.0	12.6	0.097	0.600	4.2700	-179.9986
1000.00	-28.5	0.0	9.0	0.064	0.400	2.8200	-179.9980
1580.00	-36.2	0.0	1.3	0.026	0.170	1.1600	-179.9950
2510.00	-43.4	0.0	-5;9	0.012	0.060	0.5100	-179.9887



Measured Forward Transimpedance GRAPH 7.7

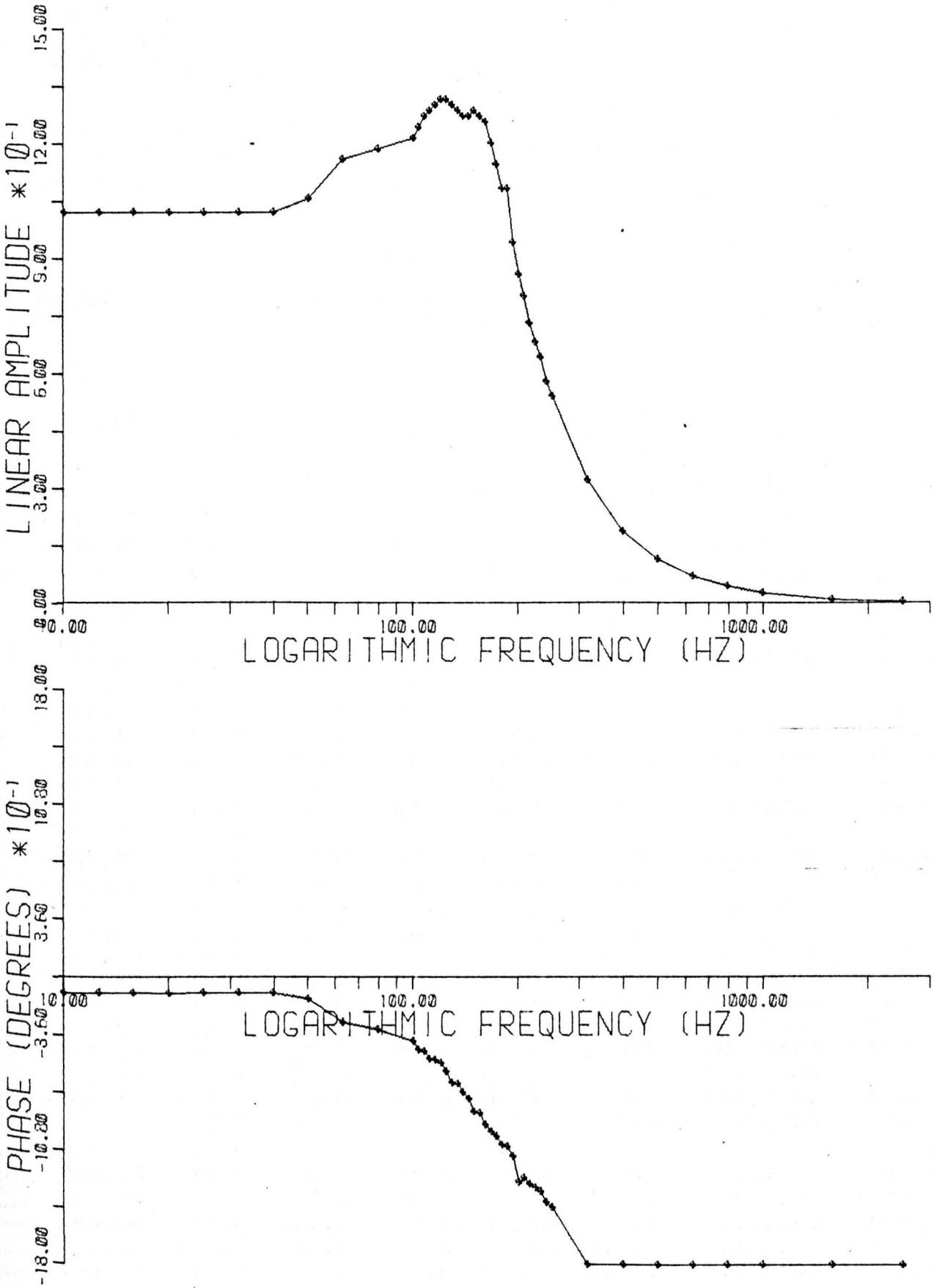
EXPERIMENTAL RESULTS TABLE 7.9

DIGITAL FORWARD TRANSFER IMPEDANCE Z21

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	100 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-17.3	20.0	40.2	0.230	0.100	102.3300	-8.8200
12.60	-17.3	20.0	40.2	0.231	0.100	102.3300	-8.7800
15.80	-17.3	20.0	40.2	0.230	0.100	102.3300	-8.8200
20.00	-17.3	20.0	40.2	0.229	0.100	102.3300	-8.8500
25.10	-17.3	20.0	40.2	0.230	0.100	102.3300	-8.8200
31.60	-17.3	20.0	40.2	0.234	0.100	102.3300	-8.6700
39.80	-17.3	20.0	40.2	0.232	0.100	102.3299	-8.7400
50.10	-17.0	20.0	40.5	0.239	0.150	105.9299	-12.7400
63.10	-16.2	20.0	41.3	0.262	0.350	116.1400	-27.3200
79.40	-16.0	20.0	41.5	0.271	0.420	118.8500	-31.8000
100.00	-15.8	20.0	41.7	0.277	0.520	121.6200	-38.7600
103.70	-15.6	20.0	41.9	0.281	0.600	124.4500	-44.3500
107.60	-15.4	20.0	42.1	0.289	0.630	127.3500	-45.3300
111.60	-15.3	20.0	42.2	0.293	0.700	128.8200	-49.9600
115.80	-15.2	20.0	42.3	0.296	0.720	130.3200	-50.9300
120.10	-15.1	20.0	42.4	0.299	0.750	131.8299	-52.6400
124.60	-15.1	20.0	42.4	0.299	0.820	131.8300	-58.0000
129.20	-15.2	20.0	42.3	0.296	0.900	130.3200	-65.0300
134.10	-15.3	20.0	42.2	0.292	0.900	128.8200	-66.0300
139.10	-15.4	20.0	42.1	0.288	0.950	127.3500	-71.3400
144.30	-15.4	20.0	42.1	0.290	0.950	127.3499	-75.1200
149.60	-15.3	20.0	42.2	0.292	1.100	128.8200	-83.5098
155.20	-15.4	20.0	42.1	0.290	1.100	127.3500	-84.2198
161.00	-15.5	20.0	42.0	0.284	1.150	125.8900	-91.4209
167.10	-15.9	20.0	41.6	0.274	1.150	120.2300	-95.7902
173.30	-16.3	20.0	41.2	0.261	1.120	114.8200	-98.6801
179.80	-16.8	20.0	40.7	0.247	1.100	108.3900	-103.8600
186.50	-16.8	20.0	40.7	0.245	1.100	108.3900	-105.0600
193.40	-18.0	20.0	39.5	0.214	1.000	94.4100	-111.3900
200.60	-18.8	20.0	38.7	0.197	1.000	86.1000	-127.6199
208.10	-9.4	10.0	38.1	0.579	2.900	80.3500	-124.6000
215.90	-10.2	10.0	37.3	0.530	2.700	73.2800	-128.4600
223.90	-10.8	10.0	36.7	0.496	2.550	68.3900	-130.6900
232.30	-11.3	10.0	36.2	0.462	2.400	64.5700	-133.3600
241.00	-12.2	10.0	35.3	0.423	2.250	58.2100	-140.1999
250.00	-12.8	10.0	34.7	0.391	2.100	54.3300	-143.4100
316.00	-17.3	10.0	30.2	0.233	1.350	32.3600	-180.0000
398.00	-11.9	0.0	25.6	0.434	2.500	19.0500	-180.0000

501.00	-16.1	0.0	21.4	0.269	1.550	11.7500	-180.0000
631.00	-20.3	0.0	17.2	0.166	0.950	7.2400	-180.0000
794.00	-24.2	0.0	13.3	0.105	0.620	4.6200	-180.0000
1000.00	-28.4	0.0	9.1	0.065	0.400	2.8500	-180.0000
1580.00	-36.2	0.0	1.3	0.026	0.170	1.1600	-180.0000
2510.00	-42.9	0.0	-5.4	0.012	0.070	0.5400	-180.0000



Z21 R_s = 100 OHMS T_s = 18.75E-6 SECONDS

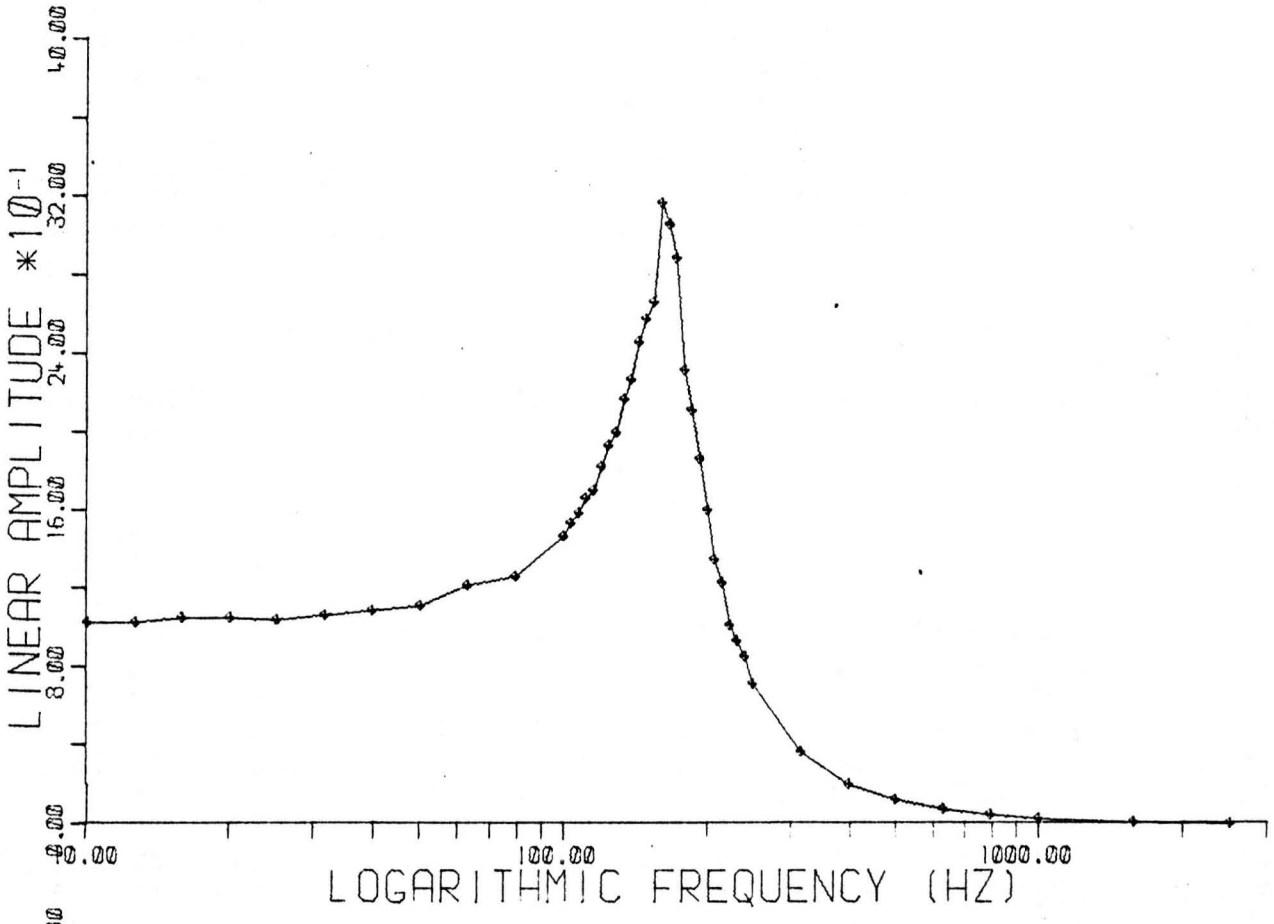
Measured Forward Transimpedance GRAPH 7.8

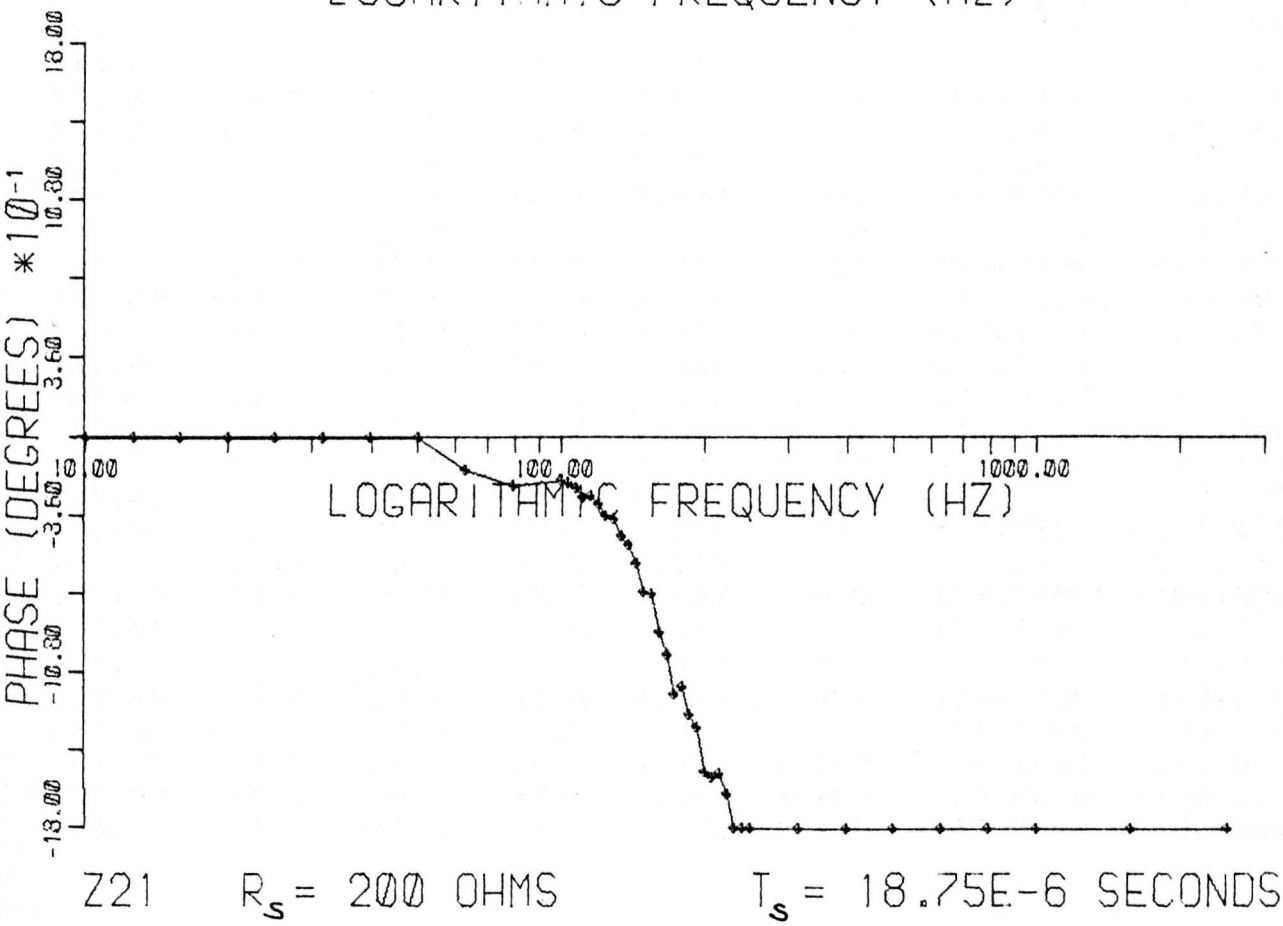
EXPERIMENTAL RESULTS TABLE 7.10
DIGITAL FORWARD TRANSFER IMPEDANCE Z21

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	200 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	|Z11| DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-17.3	20.0	40.2	0.230	0.000	102.3300	0.0000
12.60	-17.3	20.0	40.2	0.231	0.000	102.3300	0.0000
15.80	-17.1	20.0	40.4	0.238	0.000	104.7100	0.0000
20.00	-17.1	20.0	40.4	0.238	0.000	104.7100	0.0000
25.10	-17.2	20.0	40.3	0.236	0.000	103.5100	0.0000
31.60	-17.0	20.0	40.5	0.241	0.000	105.9300	0.0000
39.80	-16.8	20.0	40.7	0.246	0.000	108.3900	0.0000
50.10	-16.6	20.0	40.9	0.252	0.000	110.9200	0.0000
63.10	-14.3	20.0	41.7	0.275	0.200	121.6200	-14.7700
79.40	-11.5	20.0	42.0	0.284	0.300	125.8900	-21.5200
100.00	-14.2	20.0	43.3	0.333	0.320	146.2200	-19.5600
103.70	-13.8	20.0	43.7	0.347	0.350	153.1100	-20.5400
107.60	-13.5	20.0	44.0	0.358	0.400	158.4900	-22.7800
111.60	-13.1	20.0	44.4	0.377	0.500	165.9600	-27.1200
115.80	-12.9	20.0	44.6	0.387	0.500	169.8199	-26.4100
120.10	-12.3	20.0	45.2	0.413	0.600	181.9699	-29.7600
124.60	-11.8	20.0	45.7	0.436	0.750	192.7499	-35.4100
129.20	-11.5	20.0	46.0	0.454	0.810	199.5300	-36.7700
134.10	-10.8	20.0	46.7	0.490	1.050	216.2700	-44.5200
139.10	-10.4	20.0	47.1	0.515	1.200	226.4600	-48.6500
144.30	-9.7	20.0	47.8	0.555	1.500	245.4700	-57.0800
149.60	-9.3	20.0	48.2	0.582	1.900	257.0400	-70.4900
155.20	-9.0	20.0	48.5	0.607	2.000	266.0700	-71.2500
161.00	-17.5	20.0	50.0	0.227	0.900	316.2300	-88.9890
167.10	-17.8	20.0	49.7	0.220	0.950	305.4900	-99.5201
173.30	-18.3	20.0	49.2	0.207	1.000	288.4000	-117.2999
179.80	-10.2	20.0	47.3	0.528	2.500	231.7400	-113.6500
186.50	-11.0	20.0	46.5	0.475	2.400	211.3500	-126.5500
193.40	-12.1	20.0	45.4	0.425	2.200	186.2100	-132.4300
200.60	-13.4	20.0	44.1	0.363	2.000	160.3200	-153.7999
208.10	-14.9	20.0	42.6	0.307	1.700	134.9000	-156.4099
215.90	-15.7	20.0	41.8	0.281	1.550	123.0300	-154.3799
223.90	-17.4	20.0	40.1	0.232	1.300	101.1600	-164.2399
232.30	-18.1	20.0	39.4	0.212	1.200	93.3300	-180.0000
241.00	-18.9	20.0	38.6	0.194	1.100	85.1100	-180.0000
250.00	-10.4	10.0	37.1	0.518	2.900	71.6100	-180.0000
316.00	-16.2	10.0	31.3	0.266	1.550	36.7300	-180.0000
398.00	-11.5	0.0	26.0	0.457	2.650	19.9500	-180.0000

501.00	-15.7	0.0	21.8	0.279	1.600	12.3000	-180.0000
631.00	-19.9	0.0	17.6	0.172	1.000	7.5900	-180.0000
794.00	-24.1	0.0	13.4	0.107	0.620	4.6800	-180.0000
1000.00	-28.1	0.0	9.4	0.067	0.400	2.9500	-180.0000
1580.00	-36.1	0.0	1.4	0.027	0.170	1.1700	-180.0000
2510.00	-42.4	0.0	-4.9	0.013	0.070	0.5700	-180.0000





Z21 R_s = 200 OHMS T_s = 18.75E-6 SECONDS

Measured Forward Transimpedance GRAPH 7.9

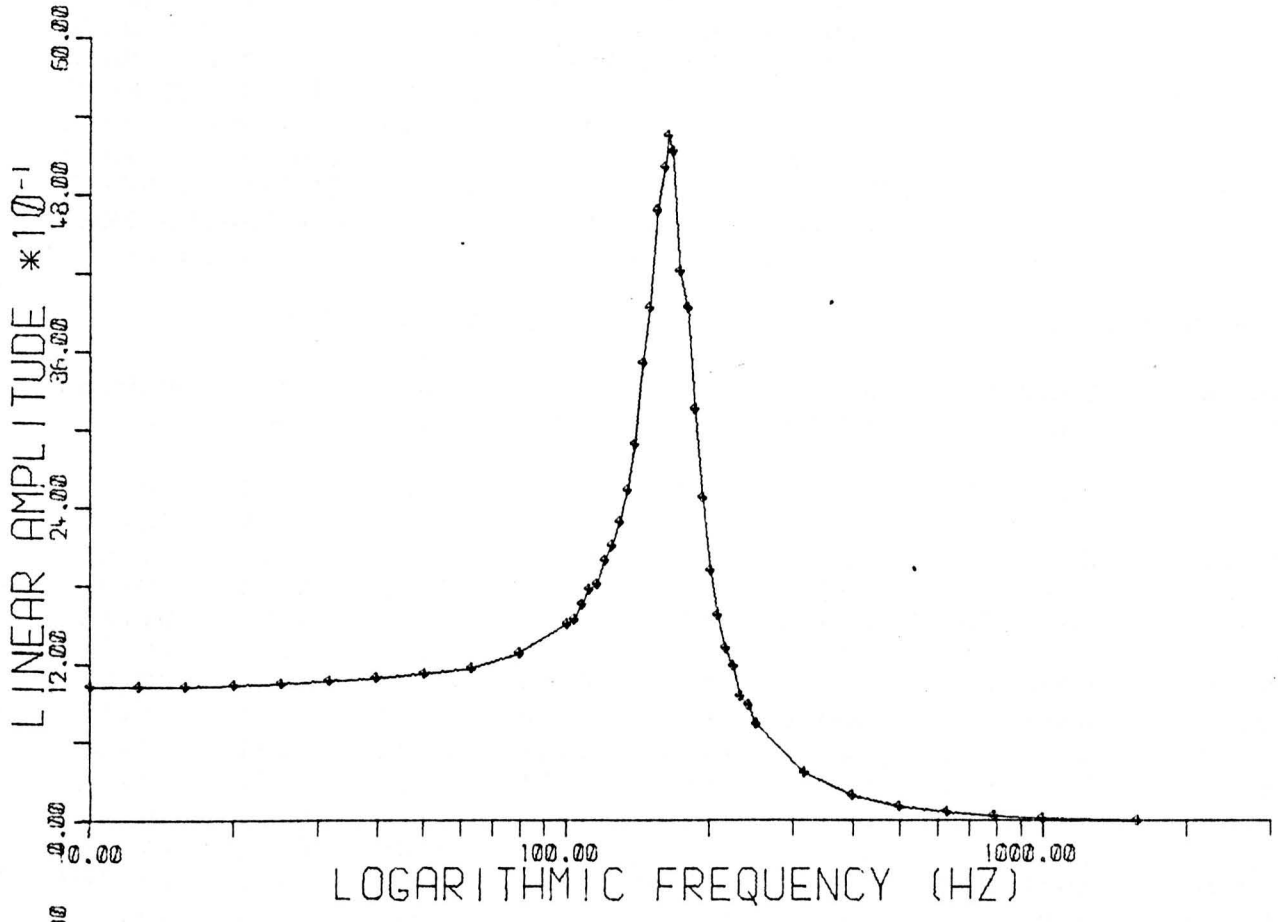
EXPERIMENTAL RESULTS TABLE 7.11

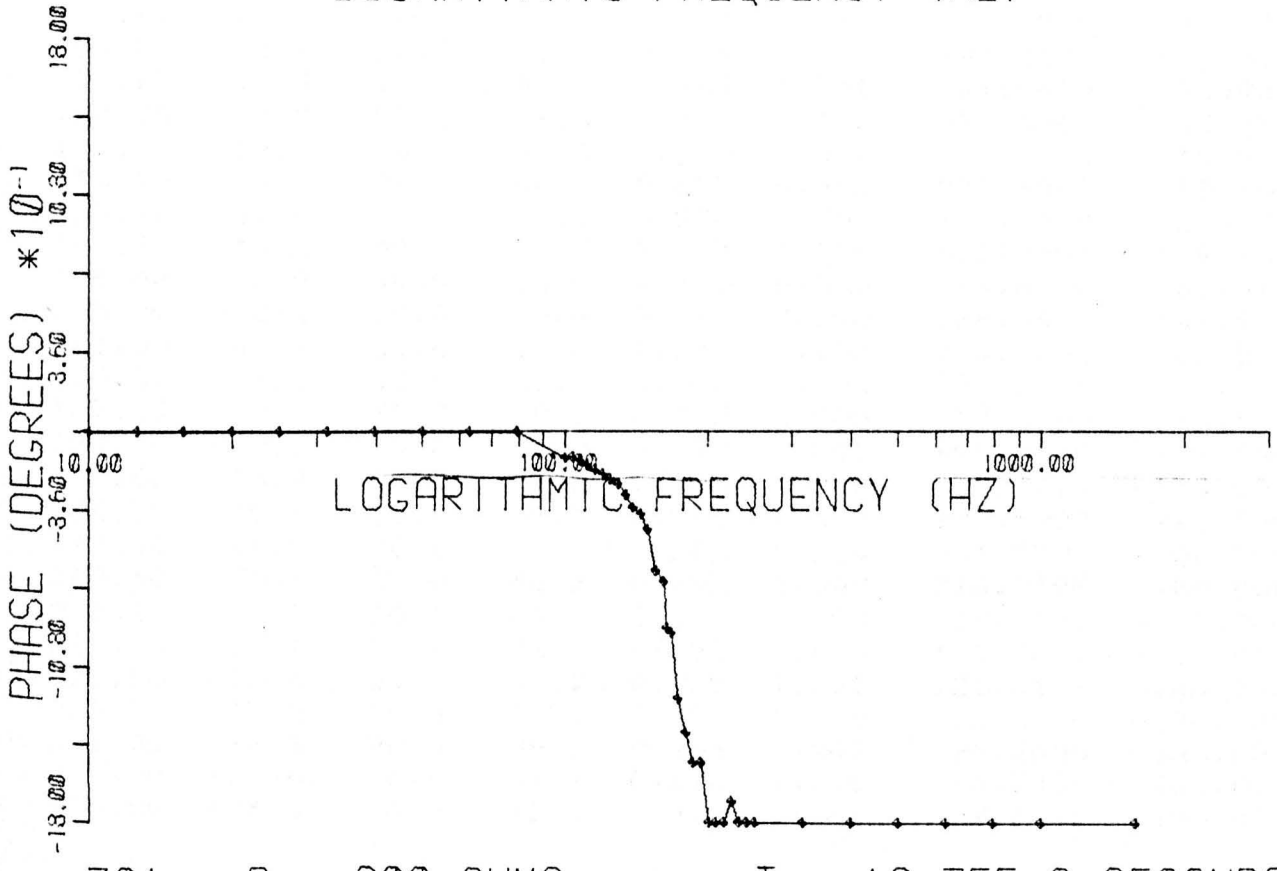
DIGITAL FORWARD TRANSFER IMPEDANCE Z21

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	300 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-17.3	20.0	40.2	0.232	0.000	102.3300	0.0000
12.60	-17.3	20.0	40.2	0.232	0.000	102.3300	0.0000
15.80	-17.3	20.0	40.2	0.234	0.000	102.3300	0.0000
20.00	-17.2	20.0	40.3	0.235	0.000	103.5100	0.0000
25.10	-17.1	20.0	40.4	0.239	0.000	104.7100	0.0000
31.60	-16.9	20.0	40.6	0.242	0.000	107.1500	0.0000
39.80	-16.7	20.0	40.8	0.248	0.000	109.6500	0.0000
50.10	-16.4	20.0	41.1	0.258	0.000	113.5000	0.0000
63.10	-16.1	20.0	41.4	0.266	0.000	117.4900	0.0000
79.40	-15.3	20.0	42.2	0.293	0.000	128.8200	0.0000
100.00	-13.9	20.0	43.6	0.343	0.200	151.3600	-11.8300
103.70	-13.7	20.0	43.8	0.353	0.200	154.8800	-11.5000
107.60	-13.1	20.0	44.4	0.379	0.250	165.9600	-13.3900
111.60	-12.5	20.0	45.0	0.402	0.300	177.8299	-15.1600
115.80	-12.3	20.0	45.2	0.412	0.350	181.9700	-17.2700
120.10	-11.5	20.0	46.0	0.451	0.420	199.5300	-18.9500
124.60	-11.0	20.0	46.5	0.477	0.500	211.3500	-21.3600
129.20	-10.3	20.0	47.2	0.521	0.600	229.0899	-23.4900
134.10	-9.4	20.0	48.1	0.578	0.800	254.1000	-28.3300
139.10	-8.3	20.0	49.2	0.658	1.100	288.4000	-34.3800
144.30	-16.6	30.0	50.9	0.251	0.450	350.7499	-36.9500
149.60	-15.6	30.0	51.9	0.282	0.600	393.5500	-44.1900
155.20	-14.1	30.0	53.4	0.337	1.000	467.7399	-63.2800
161.00	-13.5	30.0	54.0	0.361	1.150	501.1899	-68.5500
163.60	-13.3	30.0	54.4	0.376	1.500	524.8102	-89.6987
167.10	-15.0	30.0	54.2	0.369	1.500	512.8599	-91.8808
173.30	-15.6	30.0	52.5	0.303	1.500	421.7000	-122.1199
179.80	-17.5	30.0	51.9	0.284	1.500	393.5500	-138.0300
186.50	-9.6	30.0	50.0	0.226	1.240	316.2300	-151.8300
193.40	-11.8	20.0	47.9	0.565	3.100	248.3100	-151.8300
200.60	-13.5	20.0	45.7	0.439	2.500	192.7499	-180.0000
208.10	-15.0	20.0	44.0	0.361	2.050	158.4899	-180.0000
215.90	-15.9	20.0	42.5	0.303	1.700	133.3499	-180.0000
223.90	-17.8	20.0	41.6	0.275	1.550	120.2300	-170.2400
232.30	-18.4	20.0	39.7	0.221	1.300	96.6100	-180.0000
241.00	-9.9	20.0	39.1	0.205	1.200	90.1600	-180.0000
250.00	-16.0	10.0	37.6	0.544	3.200	75.8600	-180.0000
316.00	-16.0	10.0	31.5	0.272	1.600	37.5800	-180.0000

398.00	-11.4	0.0	26.1	0.462	2.700	20.1800	-180.0000
501.00	-15.7	0.0	21.8	0.281	1.600	12.3000	-180.0000
631.00	-19.9	0.0	17.6	0.172	1.000	7.5900	-180.0000
794.00	-24.1	0.0	13.4	0.107	1.000	4.6800	-180.0000
1000.00	-28.3	0.0	9.2	0.066	0.400	2.8800	-180.0000
1580.00	-36.1	0.0	1.4	0.027	0.170	1.1700	-180.0000





Z21 R_s = 300 OHMS T_s = 18.75E-6 SECONDS

Measured Forward Transimpedance GRAPH 7.10

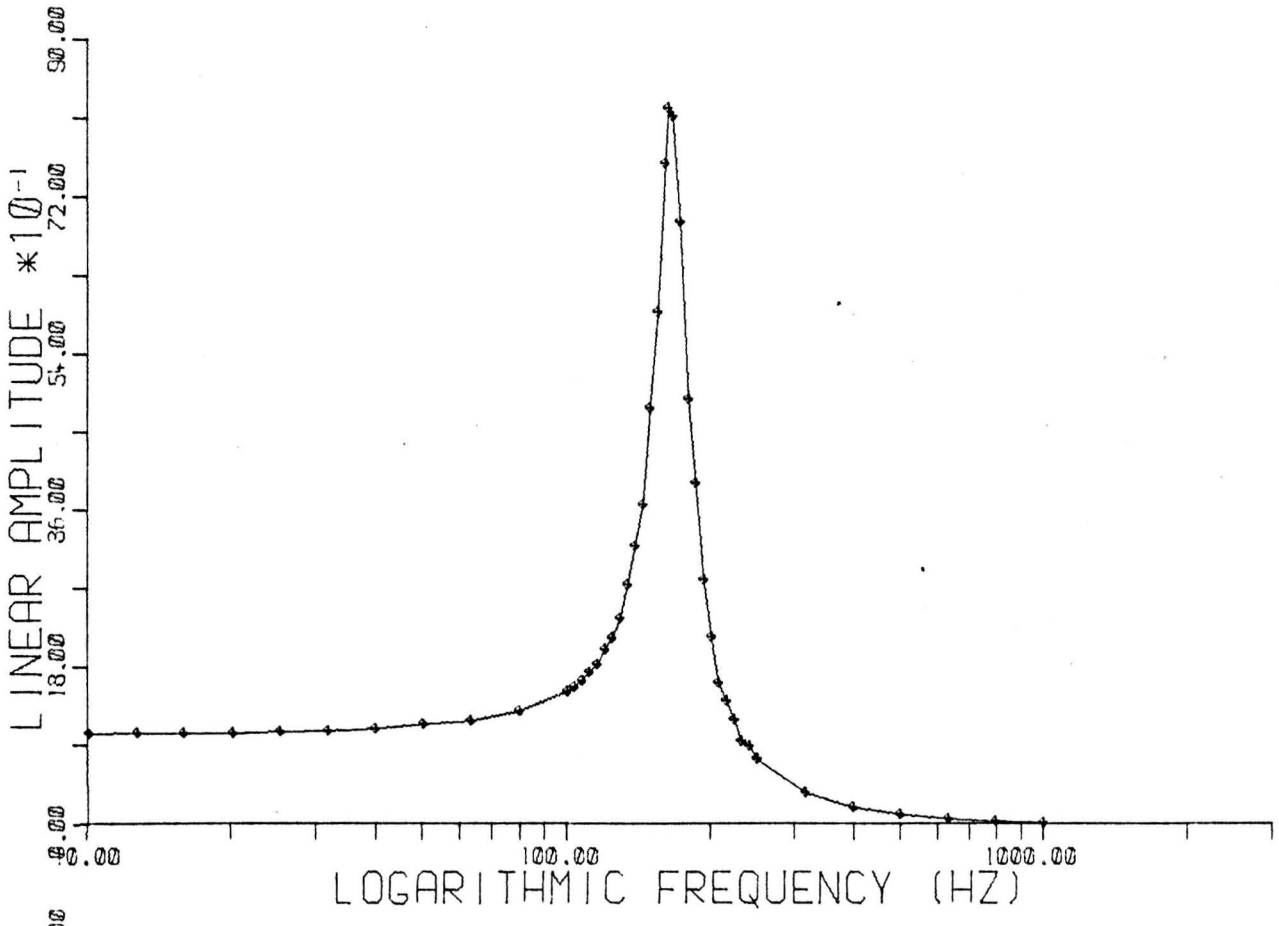
EXPERIMENTAL RESULTS TABLE 7.12

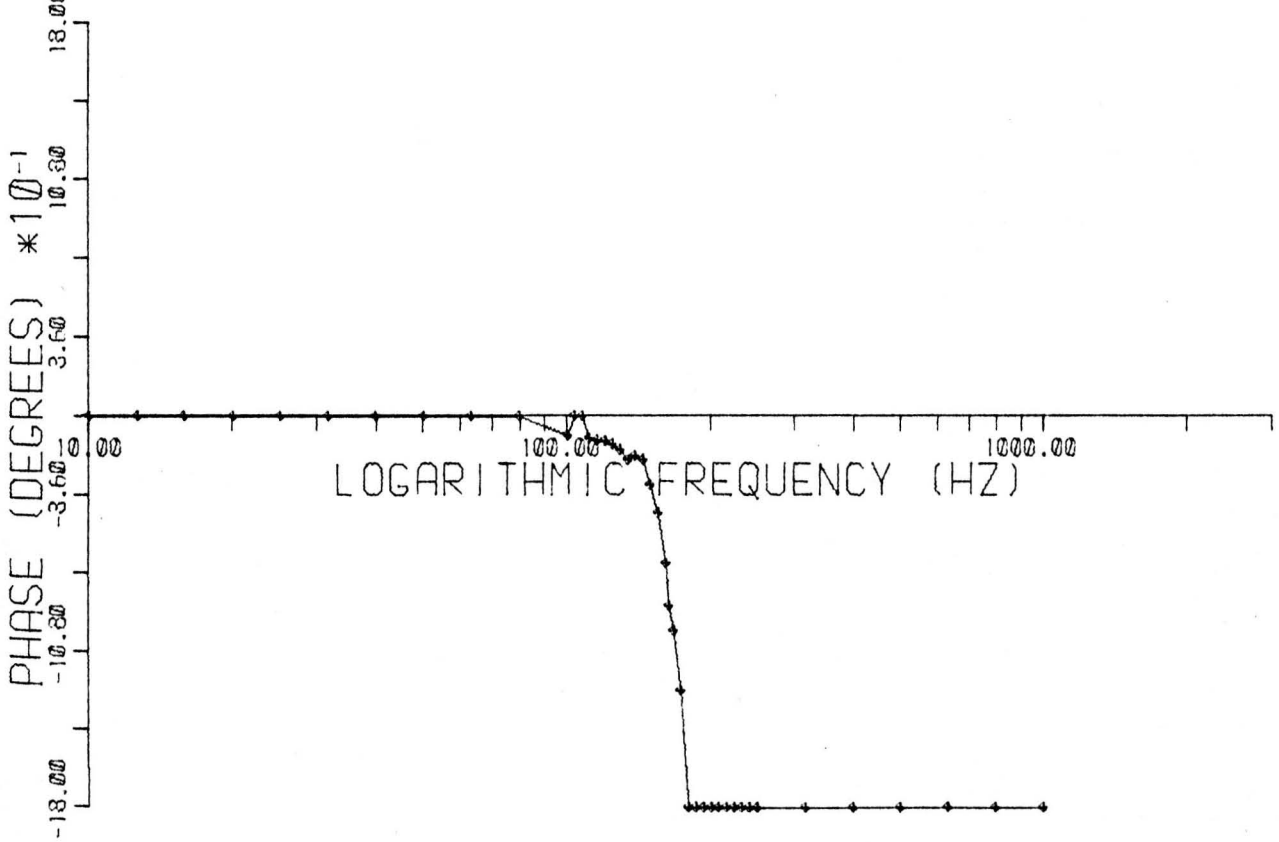
DIGITAL FORWARD TRANSFER IMPEDANCE Z21

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	400 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	|Z11| DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-17.2	10.0	40.3	0.235	0.000	103.5100	0.0000
12.60	-17.1	10.0	40.4	0.238	0.000	104.7100	0.0000
15.80	-17.1	10.0	40.4	0.236	0.000	104.7100	0.0000
20.00	-17.1	10.0	40.4	0.236	0.000	104.7100	0.0000
25.10	-17.0	10.0	40.5	0.240	0.000	105.9300	0.0000
31.60	-16.9	10.0	40.6	0.243	0.000	107.1500	0.0000
39.80	-16.7	10.0	40.8	0.246	0.000	109.6500	0.0000
50.10	-16.3	10.0	41.2	0.261	0.000	114.8200	0.0000
63.10	-16.0	10.0	41.5	0.269	0.000	118.8500	0.0000
79.40	-15.2	10.0	42.3	0.297	0.000	130.3200	0.0000
100.00	-13.8	20.0	43.7	0.347	0.150	153.1100	-8.7700
103.70	-13.5	20.0	44.0	0.361	0.000	158.4900	0.0000
107.60	-13.1	20.0	44.4	0.378	0.000	165.9600	0.0000
111.60	-12.6	20.0	44.9	0.399	0.200	175.7900	-10.1700
115.80	-12.2	20.0	45.3	0.419	0.220	184.0800	-10.6500
120.10	-11.4	20.0	46.1	0.460	0.250	201.8400	-11.0300
124.60	-10.9	20.0	46.6	0.487	0.300	213.8000	-12.5000
129.20	-10.0	20.0	47.5	0.540	0.400	237.1400	-15.0500
134.10	-8.7	20.0	48.8	0.623	0.600	275.4200	-19.6000
139.10	-17.4	30.0	50.1	0.230	0.200	319.8900	-17.6900
144.30	-16.2	30.0	51.3	0.263	0.250	367.2801	-19.3500
149.60	-13.9	30.0	53.6	0.346	0.520	478.6300	-30.8100
155.20	-12.1	30.0	55.4	0.425	0.900	588.8400	-43.9700
161.00	-9.9	30.0	57.6	0.543	1.700	758.5798	-67.2100
163.80	-9.2	30.0	58.3	0.592	2.300	822.2399	-86.7495
167.10	-9.3	30.0	58.2	0.585	2.500	812.8300	-98.1301
173.30	-10.7	30.0	56.8	0.497	2.500	691.8300	-125.5500
179.80	-13.7	30.0	53.8	0.350	2.000	489.7799	-180.0000
186.50	-15.6	30.0	51.9	0.281	1.600	393.5499	-180.0000
193.40	-18.5	20.0	49.0	0.202	1.150	281.8399	-180.0000
200.60	-10.8	20.0	46.7	0.492	2.800	216.2699	-180.0000
208.10	-13.2	20.0	44.3	0.375	2.150	164.0599	-180.0000
215.90	-14.4	20.0	43.1	0.328	1.850	142.8899	-180.0000
223.90	-15.8	20.0	41.7	0.279	1.600	121.6199	-180.0000
232.30	-17.8	20.0	39.7	0.222	1.300	96.6100	-180.0000
241.00	-18.3	20.0	39.2	0.208	1.200	91.2000	-180.0000
250.00	-19.8	20.0	37.7	0.176	1.000	76.7400	-180.0000
316.00	-16.0	10.0	31.5	0.272	1.600	37.5800	-180.0000

398.00	-11.4	0.0	26.5	0.463	2.700	21.1300	-180.0000
501.00	-15.7	0.0	21.8	0.282	1.600	12.3000	-180.0000
631.00	-19.9	0.0	17.6	0.172	1.000	7.5900	-180.0000
794.00	-24.0	0.0	13.5	0.108	1.000	4.7300	-180.0000
1000.00	-28.2	0.0	9.3	0.066	0.400	2.9200	-180.0000





Z21 R_s = 400 OHMS T_s = 18.75E-6 SECONDS

Measured Forward Transimpedance GRAPH 7.11

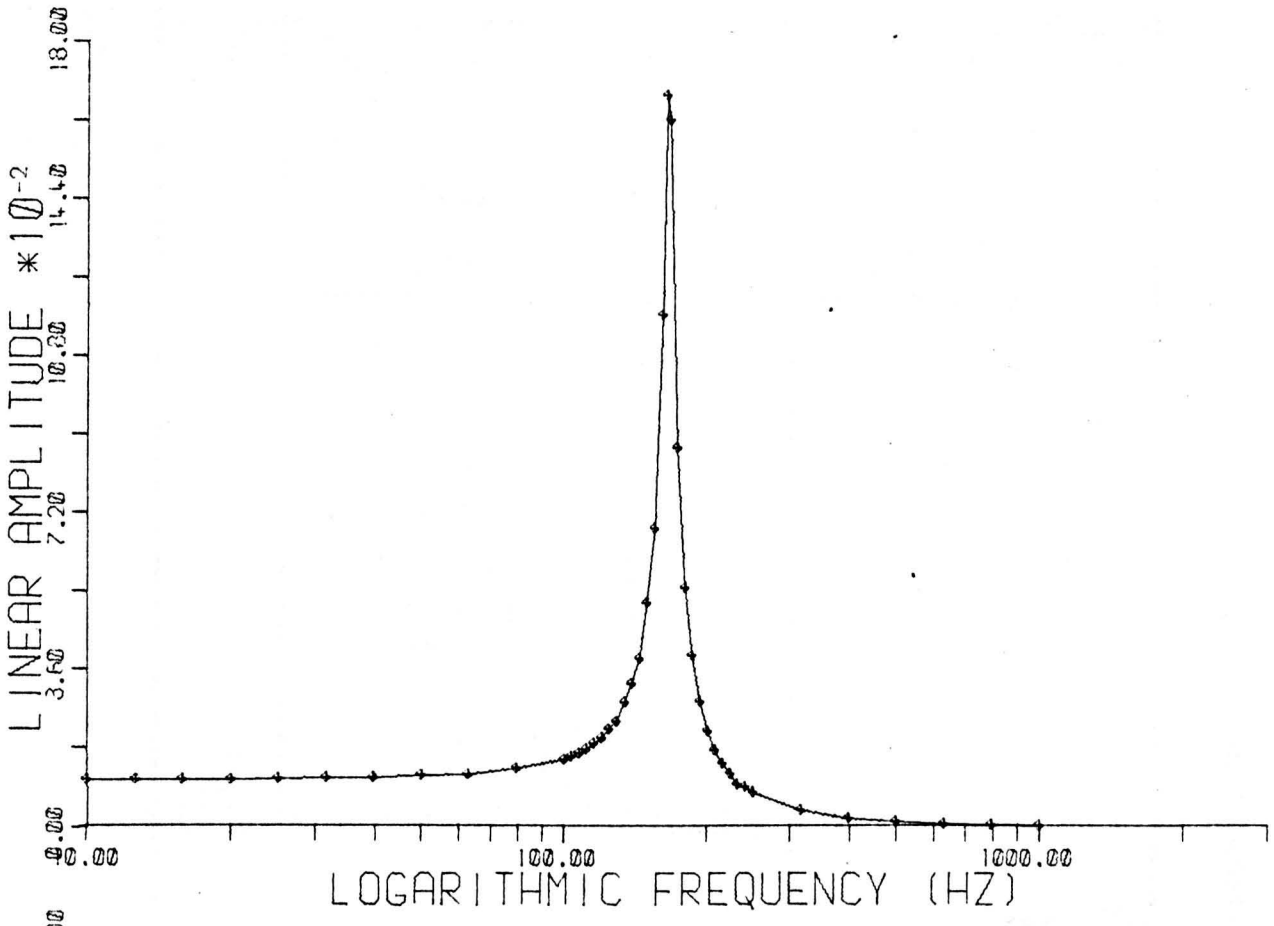
EXPERIMENTAL RESULTS TABLE 7.13

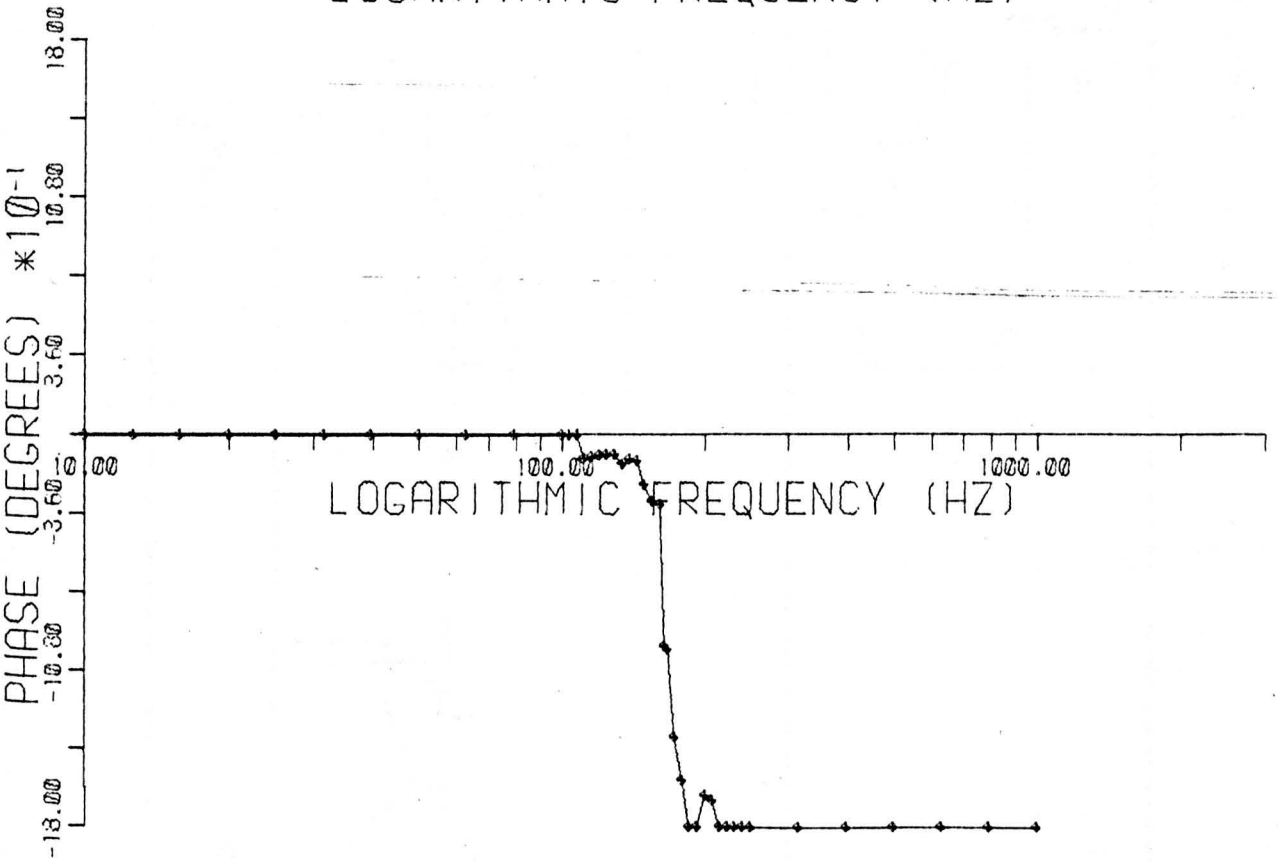
DIGITAL FORWARD TRANSFER IMPEDANCE Z21

SAMPLING PERIOD T	:	0.1875E-04 SECONDS
CAPACITOR C1	:	0.9600E-05 FARADS
CAPACITOR C2	:	0.1000E-04 FARADS
TRANSCONDUCTANCE G1	:	0.1000E-01 SIEMENS
TRANSCONDUCTANCE G2	:	0.1000E-01 SIEMENS
SHUNT RESISTANCE R	:	500 OHMS
GENERATOR INPUT	:	1.5 VOLTS (RMS)
TRANSCONDUCTANCE AMP	:	0.1333E-01 SIEMENS
TRANSCONDUCTANCE AMP	:	-37.5 DBS

	MEASURED VALUES					CALCULATED RESULTS	
FREQUENCY HZ	M DB	N DB	[Z11] DB	V1 V(RMS)	VD1 V(P-P)	MODULUS OHMS	ARGAND DEGREES
10.00	-16.8	20.0	40.7	0.244	0.000	108.3900	0.0000
12.60	-16.8	20.0	40.7	0.246	0.000	108.3900	0.0000
15.80	-16.8	20.0	40.7	0.247	0.000	108.3900	0.0000
20.00	-16.8	20.0	40.7	0.247	0.000	108.3900	0.0000
25.10	-16.7	20.0	40.8	0.249	0.000	109.6500	0.0000
31.60	-16.6	20.0	40.9	0.253	0.000	110.9200	0.0000
39.80	-16.6	20.0	40.9	0.253	0.000	110.9200	0.0000
50.10	-16.2	20.0	41.3	0.264	0.000	116.1400	0.0000
63.10	-16.0	20.0	41.5	0.271	0.000	118.8500	0.0000
79.40	-15.0	20.0	42.5	0.302	0.000	133.3500	0.0000
100.00	-13.8	20.0	43.7	0.349	0.000	153.1100	0.0000
103.70	-13.4	20.0	44.1	0.364	0.000	160.3200	0.0000
107.60	-13.0	20.0	44.5	0.380	0.000	167.8800	0.0000
111.60	-12.6	20.0	44.9	0.399	0.200	175.7900	-10.1700
115.80	-12.0	20.0	45.5	0.426	0.200	188.3600	-9.5200
120.10	-11.4	20.0	46.1	0.460	0.200	201.8400	-8.8200
124.60	-10.6	20.0	46.9	0.500	0.200	221.3100	-8.1100
129.20	-9.9	20.0	47.6	0.545	0.220	239.8800	-8.1800
134.10	-8.4	20.0	49.1	0.651	0.400	285.1000	-12.4700
139.10	-17.2	30.0	50.3	0.236	0.120	327.3400	-10.3100
144.30	-15.8	30.0	51.7	0.277	0.150	384.5900	-10.9900
149.60	-13.3	30.0	54.2	0.369	0.400	512.8600	-22.1000
155.20	-10.8	30.0	56.7	0.488	0.700	683.9100	-29.3800
161.00	-16.1	40.0	61.4	0.265	0.400	1174.9000	-30.9500
164.80	-13.0	40.0	64.5	0.380	1.600	1678.8000	-96.2002
167.10	-11.6	30.0	64.2	0.376	3.300	1621.8090	-97.5702
173.30	-8.7	30.0	58.8	0.624	3.300	870.9598	-138.4199
179.80	-12.7	30.0	54.8	0.395	2.200	549.5400	-158.2800
186.50	-15.6	30.0	51.9	0.282	1.600	393.5499	-180.0000
193.40	-18.3	30.0	49.2	0.207	1.200	288.3999	-180.0000
200.60	-10.7	20.0	46.8	0.499	2.800	218.7799	-165.4400
208.10	-12.6	20.0	44.9	0.400	2.250	175.7899	-167.8300
215.90	-14.2	20.0	43.3	0.333	1.900	146.2199	-180.0000
223.90	-15.7	20.0	41.8	0.280	1.600	123.0299	-180.0000
232.30	-17.8	20.0	39.7	0.224	1.300	96.6100	-180.0000
241.00	-18.3	20.0	39.2	0.208	1.200	91.2000	-180.0000
250.00	-19.5	20.0	38.0	0.181	1.000	79.4300	-180.0000
316.00	-16.0	10.0	31.5	0.272	1.600	37.5800	-180.0000

398.00	-11.4	0.0	26.1	0.463	2.700	20.1800	-180.0000
501.00	-15.6	0.0	21.9	0.283	1.600	12.4500	-180.0000
631.00	-19.9	0.0	17.6	0.172	1.000	7.5900	-180.0000
794.00	-23.8	0.0	13.7	0.110	1.000	4.8400	-180.0000
1000.00	-28.2	0.0	9.3	0.066	0.400	2.9200	-180.0000





Z21 R_s = 500 OHMS T_s = 18.75E-6 SECONDS

Measured Forward Transimpedance GRAPH 7.12

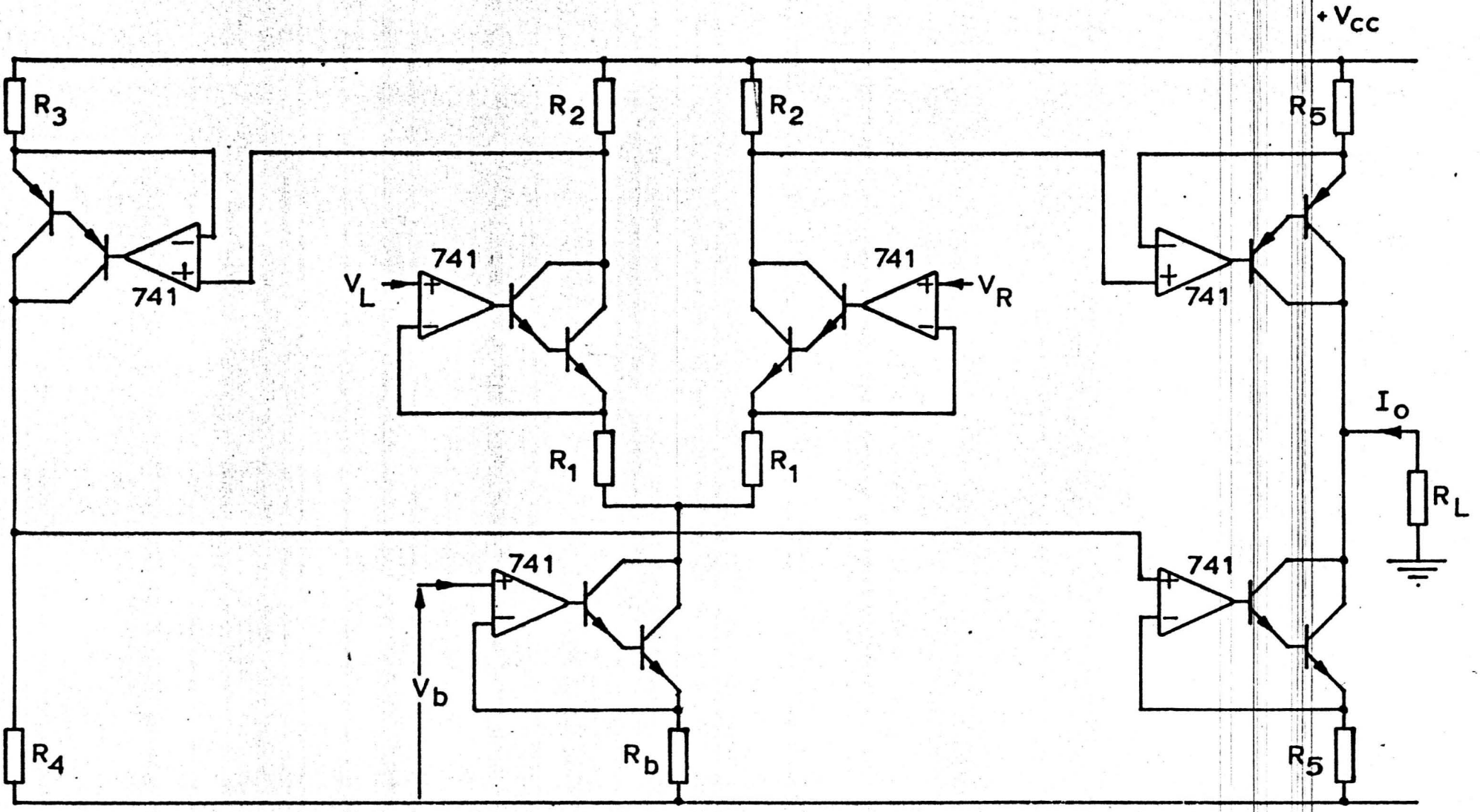


FIG. 7.4 Input Transadmittance Amplifier

RESISTOR	VALUE (OHMS)
R_1	100
R_2	100
R_3	100
R_4	100
R_5	10
R_B	100

TABLE 7.14

Component values for analogue transadmittance amplifier

PARAMETER	VALUE	UNITS
g	0.1	Siemens
I_B	250	mA
V_{ref}	2.5	VOLTS
R_L	50	OHMS

TABLE 7.15

Parameters for analogue transadmittance amplifier

EXPERIMENTAL RESULTS TABLE 7.16

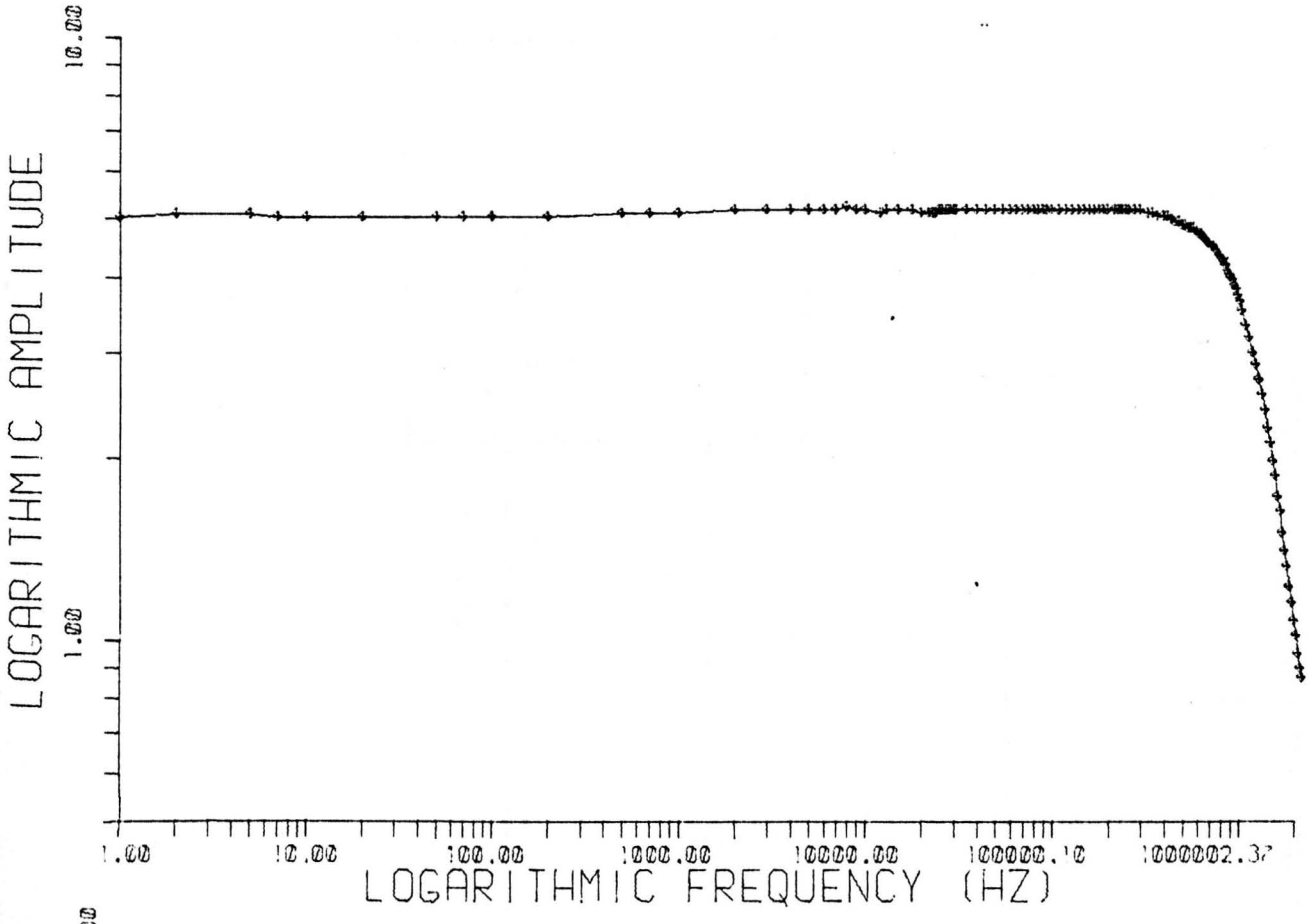
GAIN OF TRANSCONDUCTANCE AMPLIFIER LOADED BY 50 OHMS

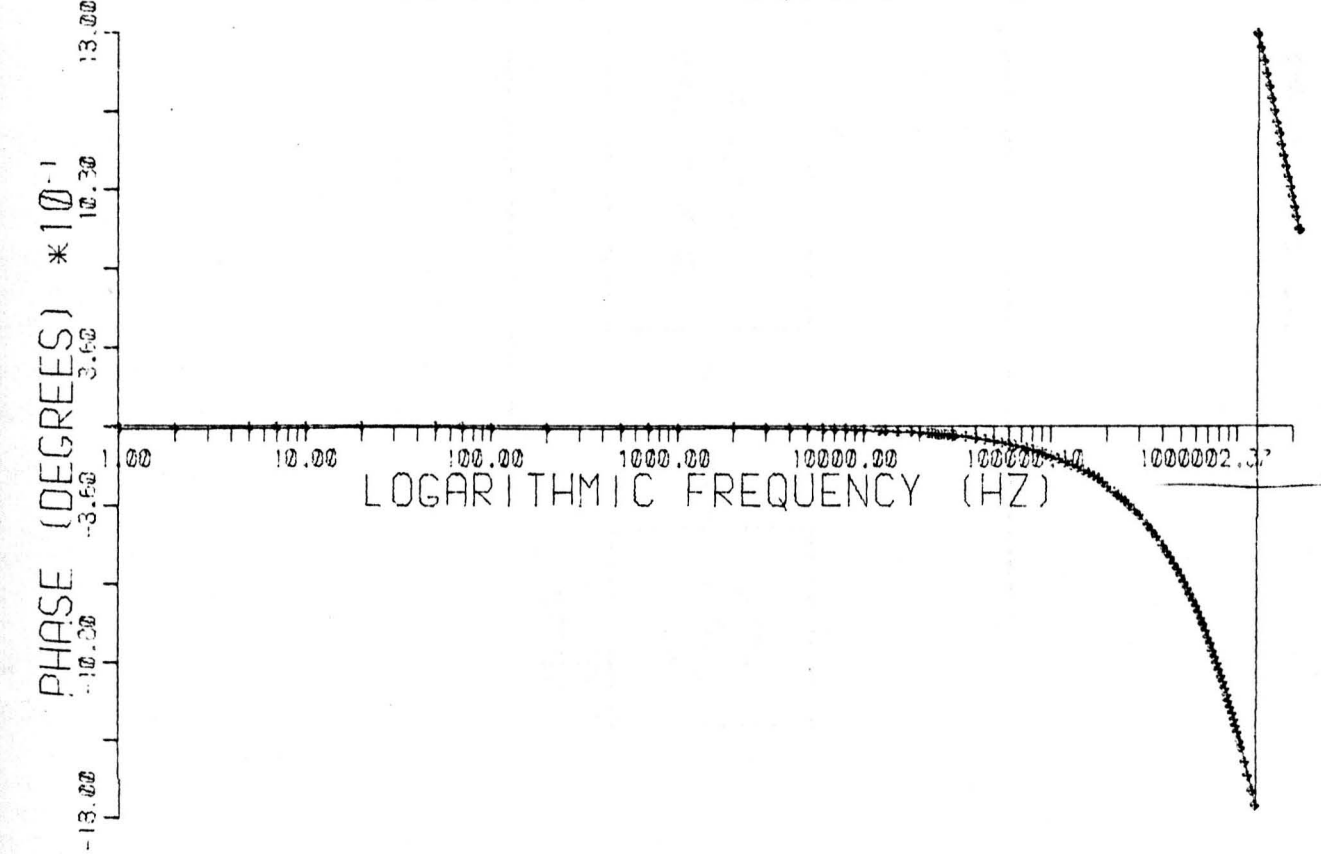
GENERATOR INPUT : 2.6 VOLTS (RMS)

	MEASURED VALUES	
FREQUENCY HZ	MODULUS	ARGAND DEGREES
1	5.0699	-0.2000
2	5.1286	-0.2000
5	5.1286	-0.1000
7	5.0699	-0.1000
10	5.0699	-0.1000
20	5.0699	0.0000
50	5.0699	0.0000
70	5.0699	-0.1000
100	5.0699	-0.2000
200	5.0699	-0.3000
500	5.1286	-0.3000
700	5.1286	-0.2000
1000	5.1286	-0.2000
2000	5.1880	-0.1000
3000	5.1880	-0.5000
4000	5.1880	-0.3000
5000	5.1880	-0.5000
6000	5.1880	-0.7000
7000	5.1880	-0.8000
8000	5.2481	-0.8000
9000	5.1880	-1.0000
10000	5.1880	-1.2000
12000	5.1286	-1.5000
13000	5.1880	-1.6000
15000	5.1880	-1.7000
18000	5.1880	-1.8000
20000	5.1286	-2.2000
22000	5.1286	-2.5000
23000	5.1286	-2.7000
24000	5.1286	-2.9000
25000	5.1880	-3.1000
26000	5.1880	-3.2000
27000	5.1880	-3.2000
28000	5.1880	-3.3000
29000	5.1880	-3.4000
30000	5.1880	-3.4000
31000	5.1880	-3.5000
35000	5.1880	-4.0000
40000	5.1880	-4.6000
45000	5.1880	-5.3000
50000	5.1880	-6.1000
55000	5.1880	-6.8000
60000	5.1880	-7.5000
65000	5.1880	-8.1000
70000	5.1880	-8.8000
75000	5.1880	-9.4000

80000	5.1880	-10.1000
85000	5.1880	-10.8000
90000	5.1880	-11.5000
95000	5.1880	-12.1000
100000	5.1880	-12.8000
110000	5.1880	-14.2000
120000	5.1880	-15.5000
130000	5.1880	-16.8000
140000	5.1880	-18.1000
150000	5.1880	-19.4000
160000	5.1880	-20.7000
170000	5.1880	-21.9000
180000	5.1880	-23.2000
190000	5.1880	-24.9000
200000	5.1880	-27.0000
220000	5.1880	-29.6000
230000	5.1880	-30.8000
240000	5.1880	-32.2000
250000	5.1880	-33.5000
260000	5.1880	-34.9000
280000	5.1880	-37.6000
300000	5.1880	-40.1000
330000	5.1286	-44.3000
350000	5.1286	-47.0000
370000	5.0699	-49.9000
400000	5.0699	-53.9000
420000	5.0699	-56.8000
440000	5.0119	-59.5000
460000	4.9545	-62.4000
480000	4.9545	-65.1000
500000	4.8978	-68.0000
520000	4.8978	-70.8000
540000	4.8417	-73.6000
560000	4.8417	-76.3000
580000	4.8417	-79.1999
600000	4.7863	-81.8998
620000	4.7315	-84.6998
640000	4.7315	-87.5994
657500	4.6774	-90.0000
660000	4.6774	-90.5013
680000	4.6238	-93.4005
700000	4.5709	-96.3002
720000	4.5186	-99.1001
740000	4.5186	-102.1000
760000	4.4668	-105.1000
780000	4.4157	-108.0000
800000	4.3652	-111.0000
820000	4.3152	-113.8000
840000	4.2658	-116.7999
860000	4.2170	-119.6000
880000	4.1210	-122.6999
900000	4.0738	-125.6000
920000	4.0272	-128.4000
940000	3.9811	-131.2000
960000	3.8905	-134.3000
980000	3.8459	-137.0999

1000000	3.7584	-139.9999
1030000	3.6728	-144.1000
1050000	3.5481	-146.9000
1100000	3.3497	-153.6998
1150000	3.1989	-160.3000
1200000	3.0200	-166.9000
1250000	2.8840	-173.8000
1295000	2.7227	180.0000
1300000	2.7227	179.3000
1350000	2.5704	173.5000
1400000	2.4266	167.4000
1450000	2.2646	161.5000
1500000	2.1380	155.8000
1550000	1.9953	150.0999
1600000	1.8836	144.8000
1650000	1.7378	139.4000
1700000	1.6406	134.2000
1750000	1.5136	129.0000
1800000	1.4125	124.0000
1850000	1.3335	119.0000
1900000	1.2303	114.2000
1950000	1.1614	109.7000
2000000	1.0839	105.0000
2050000	1.0233	100.4000
2100000	0.9550	95.7003
2150000	0.9016	90.0015
2200000	0.8710	90.0000





ANALOGUE TRANSADMITTANCE AMPLIFIER RESPONSE.

Used to test Digital Gyrator. GRAPH 7.13

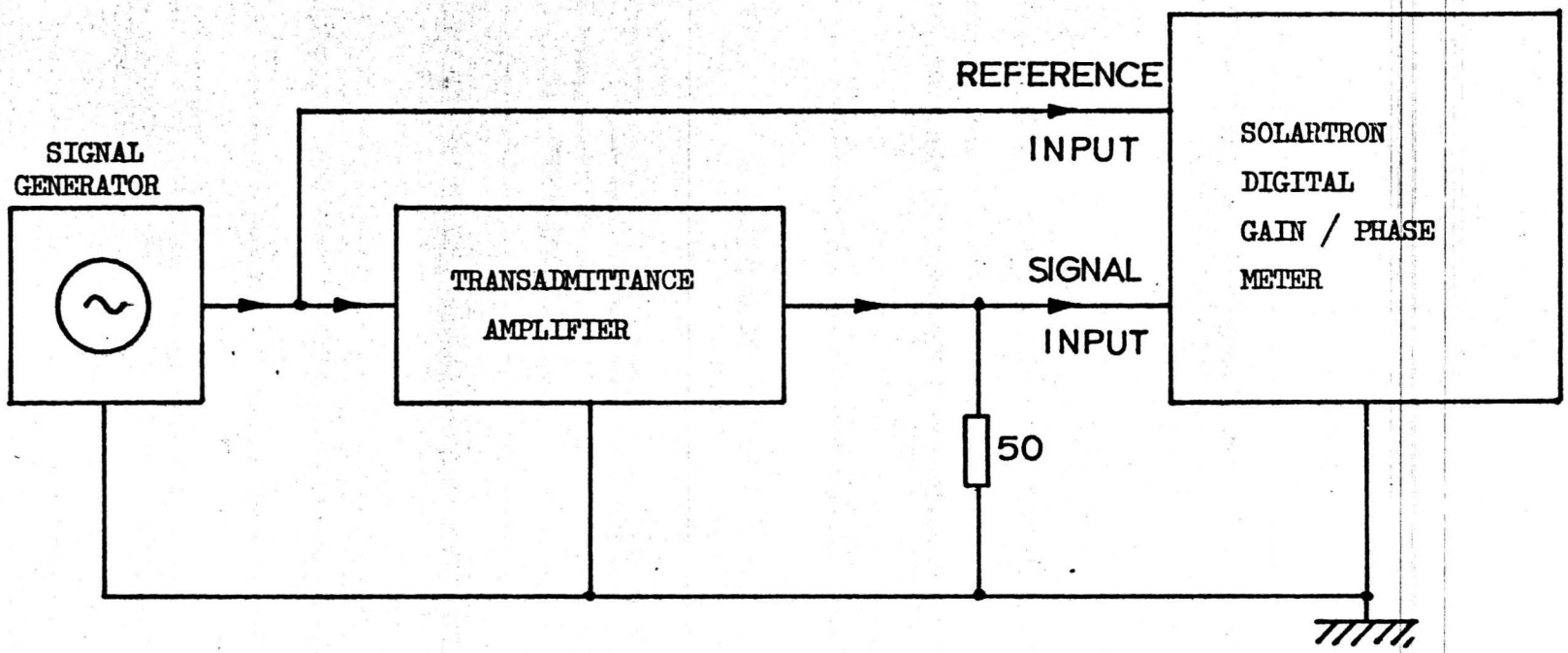


FIG. 7.5 Transadmittance Amplifier Test Arrangement

RESISTOR	VALUE (OHMS)
R_1	100
R_2	100
R_3	100
R_4	100
R_5	7.5
R_B	100

TABLE 7.17

Component Values for analogue transadmittance amplifier used in experiments.

transconductance was set to 0.013335 siemens.

7.6 DIGITAL AMPLIFIER CALIBRATION

The digital amplifiers used in the digital gyrator needed calibrating before the measurements of Sections 7.3 and 7.4 could be taken. Fig. 7.6 shows the block diagram of a digital amplifier. Thus:

$$g = g_a g_b \tag{7.7}$$

where:

$$g_a \propto V_a$$

$$g_b \propto V_{REF}$$

To calibrate the amplifiers V_a was set to its maximum value and V_{REF} adjusted until $\left| g \right|$ = 0.01 siemens by measuring V and I.

7.7 LIMIT CYCLE NOISE

The impulse responses shown in Photos 7.14 to 7.21 clearly show that a limited amplitude oscillation is present all the time provided that the shunt resistance is greater than 100 ohms. The significance of this oscillation was not realised during the time of practical experimentation and thus no accurate quantitative measurements of the amplitude and frequency of this oscillation were obtained.

Nevertheless the analyses in Sections 3.11 and 4.5 clearly show that this oscillation is to be expected under certain component conditions, and that the measurement error that is introduced depends on the relative limit cycle noise and signal level (equation 3.84)).

A correction factor (k_q) to be applied to the measured results may be calculated from these results. Let the total measured voltage at port i be V_i and the limit cycle noise voltage be V_{qi}. From equation (3.84):

$$\frac{\triangle z_{ij}}{z_{ij}} = \frac{V_{qi}}{V_i - V_{qi}} \tag{7.8}$$

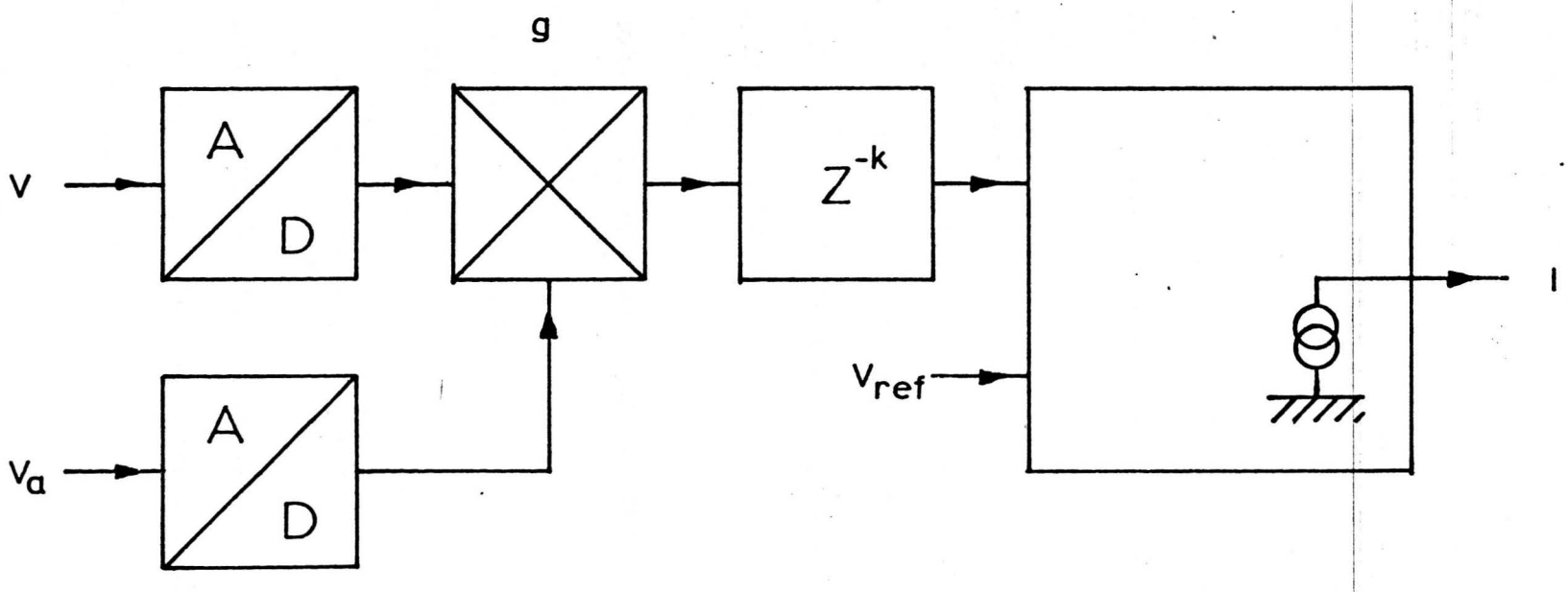


FIG. 7.6 Digital Amplifier

and k_q may be defined thus:

$$(z_{ij} + \triangle z_{ij})\ k_q = z_{ij} \tag{7.9}$$

By rearranging equation (7.9):

$$k_q = \frac{V_i - V_{qi}}{V_i} \tag{7.10}$$

The limit cycle noise voltages have been measured from Photos 7.16, 7.17, 7.20 and 7.21 and are listed in Table 7.18 for shunt resistances of 200 and 300 ohms. However for the purposes of deriving the correction factor k_q the computed noise voltages from Tables 8.35 and 8.36 were used.

7.8 ACCURACY OF RESULTS

The accuracy of the measured results depended on two factors, the test equipment accuracy, and the limit cycle oscillation amplitude. Thus the test equipment inaccuracy may be applied to the measured results as shown in Tables 7.19 and 7.20.

7.8.1 Test Equipment Accuracy

The effects of the inaccuracy of the test equipment may be derived from equation (G.7) with reference to the test arrangements shown in Figs 7.1 and 7.3. Thus for the digital input impedance the fractional error will be:

$$\frac{\triangle z_{11}}{z_{11}} = \frac{\triangle M}{M} + \frac{\triangle N}{N} + \frac{\triangle g}{g} \tag{7.11}$$

The error $\triangle g$ in the transconductance amplifier depended on the component values used. From equation (6.26):

$$g = R_2 / R_1 R_5 \tag{7.12}$$

and thus:

$$\frac{\triangle g}{g} = \frac{\triangle R_2}{R_2} + \frac{\triangle R_1}{R_1} + \frac{\triangle R_5}{R_5} \tag{7.13}$$

PHOTO	SAMPLING PERIOD μs	SHUNT RESISTANCE Ω	NOISE VOLTAGE mV (rms)	PORT
7.16	18.75	200	70.7	1
7.17	18.75	300	141	1
7.20	18.75	200	88	2
7.21	18.75	300	141	2

TABLE 7.18

Quantisation Noise Voltages from Photographic Results

SHUNT RESISTANCE	MEASURED PEAK IMPEDANCE	MEASURED PEAK IMPEDANCE RANGE (± 8.32%)		COMPUTED LIMIT CYCLE	MEASURED SIGNAL	CORRECTED PEAK IMPEDANCE RANGE (± 8.32%)	
		MIN	MAX			MIN	MAX
Ω	Ω	Ω	Ω	mV (rms)	mV (rms)	Ω	Ω
50	53.70	49.23	58.18	0	388	49.23	58.18
100	118.85	108.96	128.74	0	271	108.96	128.74
200	269.15	246.76	291.54	113.3	611	201.11	237.61
300	512.86	470.19	555.53	180.5	370	240.81	284.52
400	831.76	762.56	900.96	257.7	600	435.42	514.45
500	1513.56	1387.63	1639.49	347.0	346	-	-

Table 7.19

Corrected Peak Impedance Ranges for z_{11}

SHUNT RESISTANCE	MEASURED PEAK IMPEDANCE	MEASURED PEAK IMPEDANCE RANGE (± 8.32%)		COMPUTED LIMIT CYCLE NOISE	MEASURED SIGNAL	CORRECTED PEAK IMPEDANCE RANGE (±8.32%)	
		MIN	MAX			MIN	MAX
Ω	Ω	Ω	Ω	mV (rms)	mV(rms)	Ω	Ω
50	No peak	-	-	0	-	-	-
100	No peak	-	-	0	-	-	-
200	316.23	289.92	342.54	90.0	227	175.11	206.89
300	524.81	481.15	568.47	154.6	376	283.40	334.83
400	822.24	753.83	890.65	229.8	592	461.34	545.08
500	1678.80	1539.12	1818.48	317.0	380	255.49	301.87

Table 7.20

Corrected Peak Impedance Ranges for z_{21}

As all the resistors used had a 2% tolerance then:

$$\frac{\triangle g}{g} = 6\% \qquad (7.14)$$

The minimum attenuator step size was 0.1 decibels, and hence:

$$\frac{\triangle N}{N} = \frac{\triangle M}{M} = 1.16\% \qquad (7.15)$$

Thus the overall test equipment inaccuracy was 8.32%.

7.8.2 Limit Cycle Noise Error

The correction factor k_q for the measured peak impedances is listed with the corrected impedances in Tables 7.19 and 7.20 for the digital input impedance z_{11} and forward transfer impedance z_{21} respectively.

7.9 PHOTOGRAPHIC RESULTS

The qualitative magnitude frequency responses for the digital gyrator (Fig. 7.2) were also obtained using the arrangement of equipment shown in Fig. 7.7 to 7.9 and with the sample period set to 18.75 μS. The sweep generator was set to logarithmically scan the useful operating frequency range. Provided that the sweep rate was slow enough, the envelope of the voltage waveform was also the shape of the amplitude frequency response. The slow sweep rate was necessary to minimise the sidebands caused by sweeping this oscillator, thus making the input approximate to a sinusoid of constant frequency.

The input to the transadmittance amplifier was set to 1 volt peak-peak, and $g = 0.0133$ siemens making the output current $I_1 = 13.3$mA.

7.9.1 Magnitude Response - z_{11}

Fig. 7.7 shows the arrangement of equipment to measure qualitatively the voltage at Port 1 of the digital gyrator. The input voltage to the transadmittance amplifier was measured because the input current

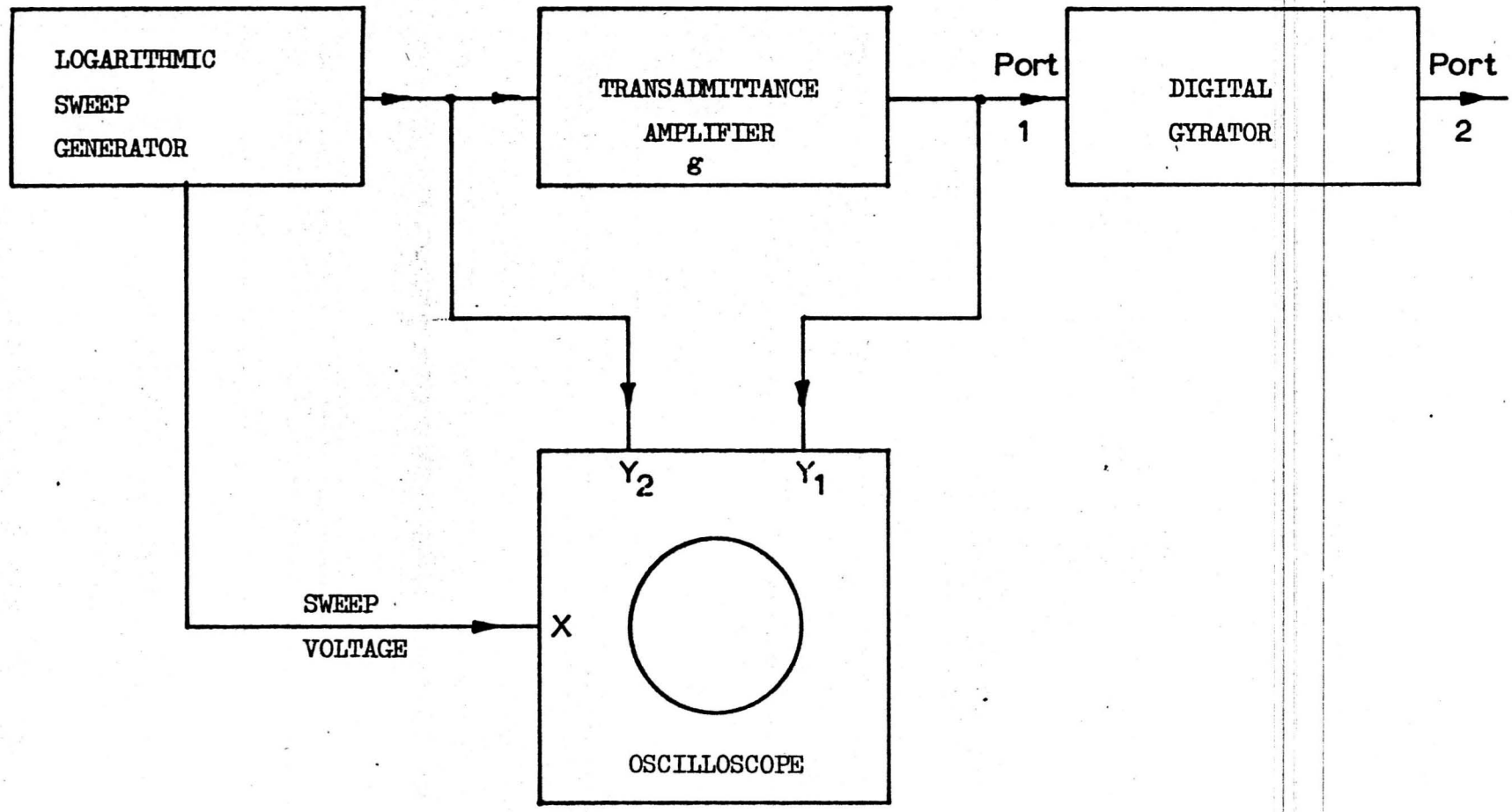


FIG. 7.7 Qualitative Measurement of Digital Input Impedance (z_{11})

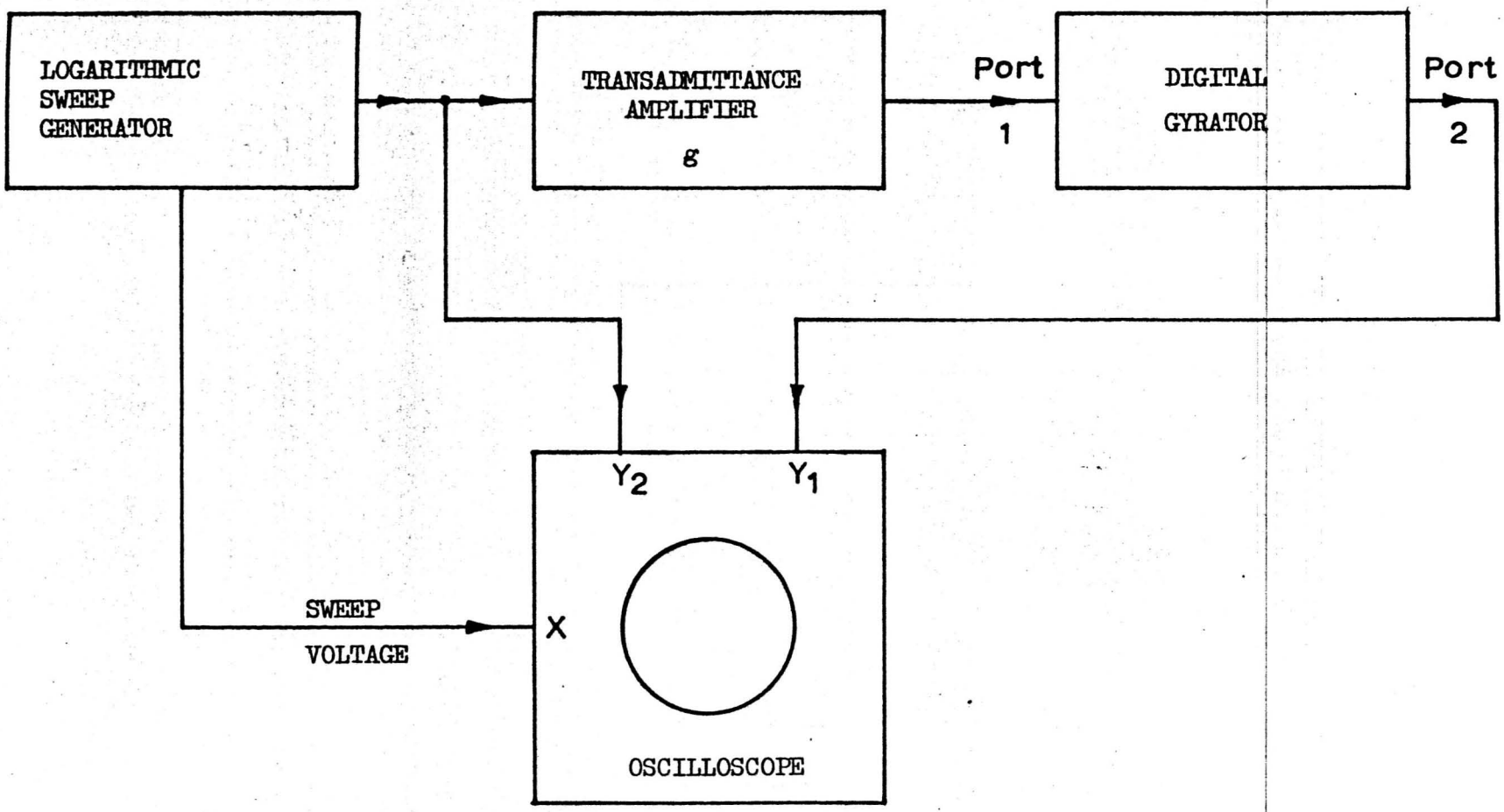


FIG. 7.8 Qualitative Measurement of Digital Forward Transimpedance (z_{21})

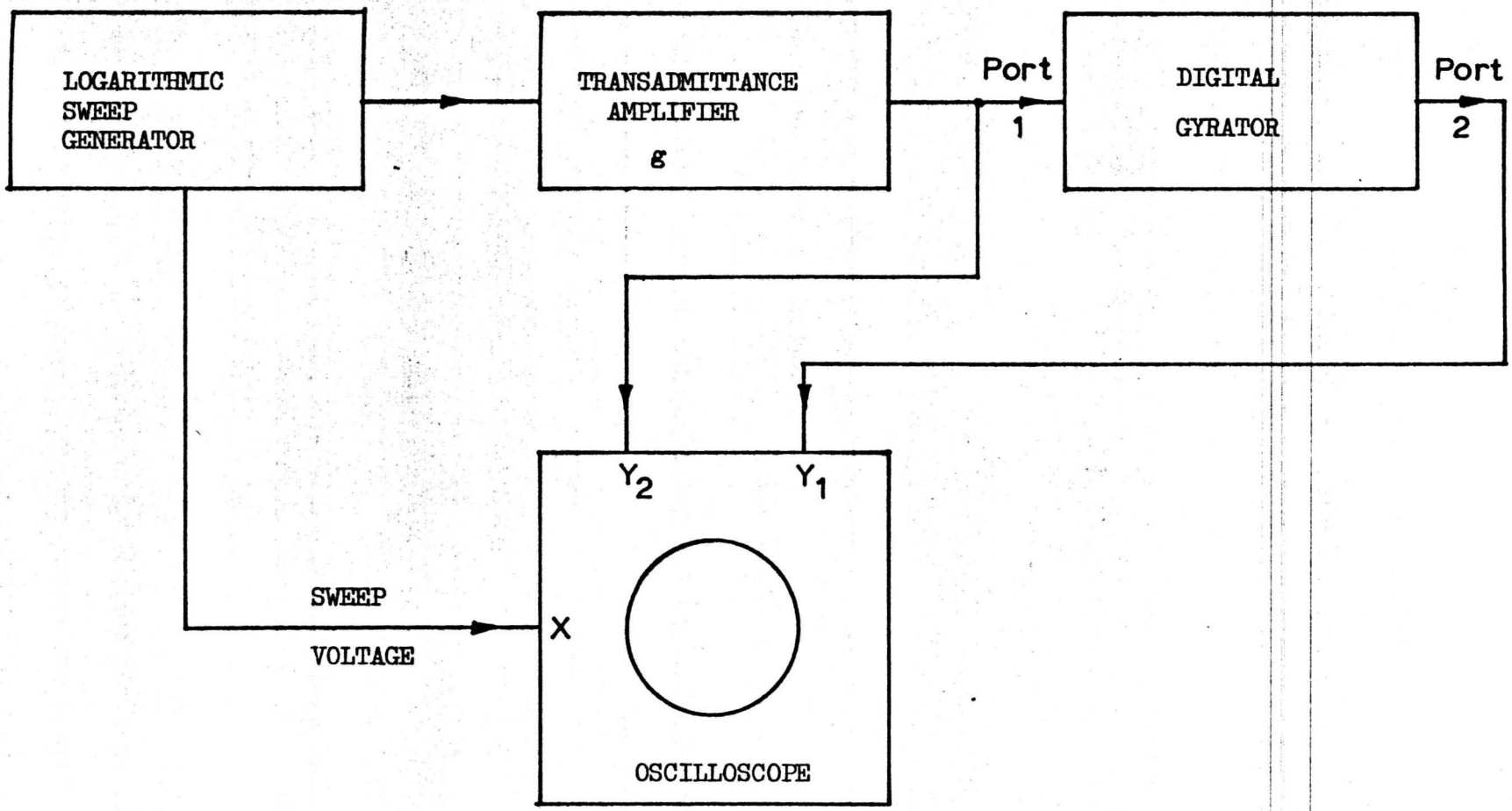


FIG. 7.9 Qualitative Measurement of Digital Voltage Gain

to the gyrator was directly proportional to this voltage. Photos 7.1 to 7.4 show the voltage on Port 1 (upper trace) against the input voltage to the transadmittance amplifier for 4 values of R_S.

Photos 7.3 and 7.4 show a beating effect between the sweep signal and the limit cycle oscillation present because the shunt resistance was 200 and 300 ohms respectively.

7.9.2 Magnitude Response - z_{21}

Fig. 7.8 shows the arrangement of equipment to measure qualitatively the voltage at Port 2. Photos 7.5 to 7.8 show the voltage at Port 2 (upper trace) against the input voltage to the transadmittance amplifier for 4 values of R_S.

Photos 7.7 and 7.8 show a beating effect between the sweep signal and the limit cycle oscillation present.

7.9.3 Relative Magnitude Response

Fig. 7.9 shows the arrangement of equipment to measure qualitatively the voltages at both ports. Photos 7.9 to 7.13 show these voltages with the voltage at Port 1 as the upper trace for 5 values of R_2.

7.9.4 Impulse Response of z_{11}

The impulse response of z_{11} was measured qualitatively using the arrangement of equipment in Fig. 7.7. Photos 7.14 to 7.17 show the voltage at Port 1 (upper trace) in response to a step change in input current to Port 1 for 4 values of R_S.

Photos 7.16 and 7.17 show the presence of a limit cycle oscillation because the shunt resistance is 200 and 300 ohms respectively. The rms limit cycle noise voltages from the photos are listed in Table 7.18.

7.9.5 Impulse Response of z_{21}

The impulse response of z_{21} was measured qualitatively using

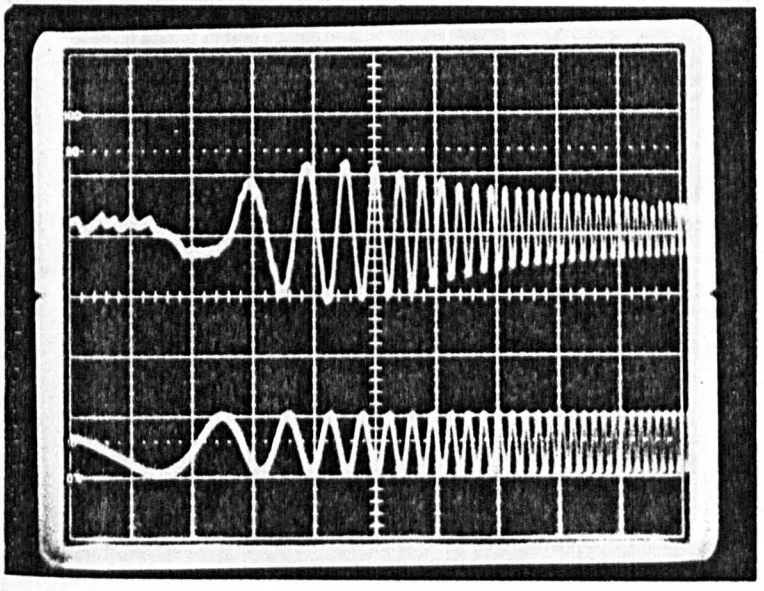


PHOTO 7.1

X - Sweep Voltage

Y_1 - 0.2V / div

Y_2 - 1.0V / div

Envelope Response at Port 1 with R = 50 ohms

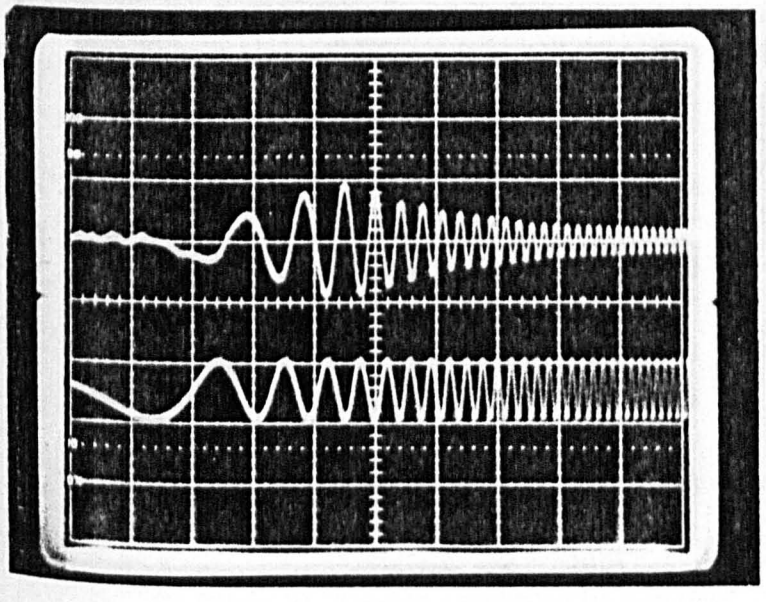


PHOTO 7.2

X - Sweep Voltage

Y_1 - 0.5V / div

Y_2 - 1.0V / div

Envelope Response at Port 1 with R = 100 ohms

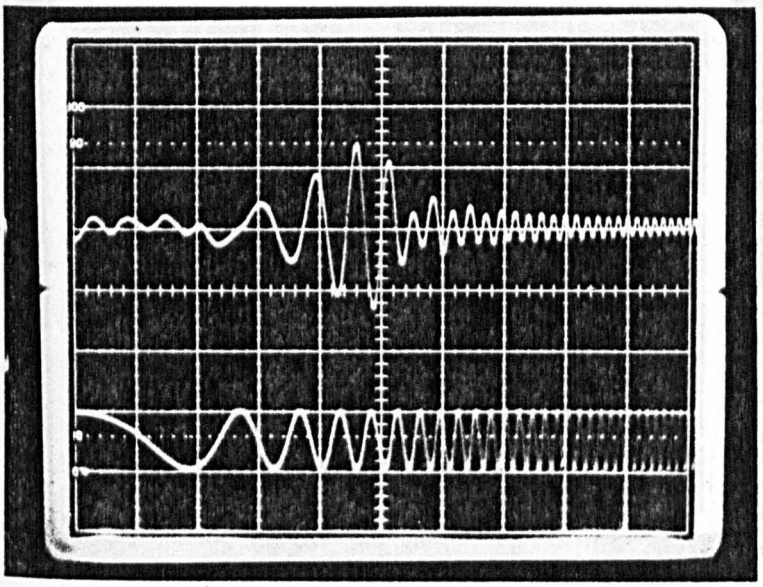


PHOTO 7.3

X - Sweep Voltage

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Envelope Response at Port 1 with R = 200 ohms

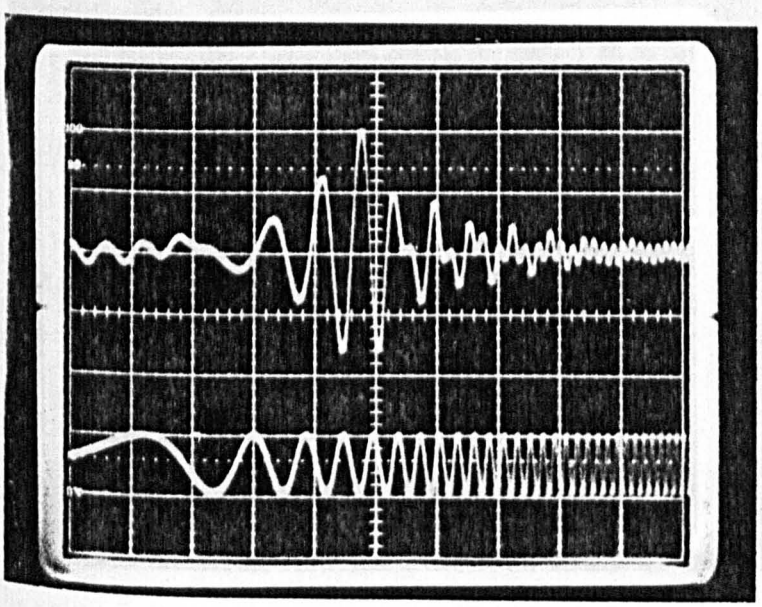


PHOTO 7.4

X - Sweep Voltage

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Envelope Response at Port 1 With R = 300 ohms

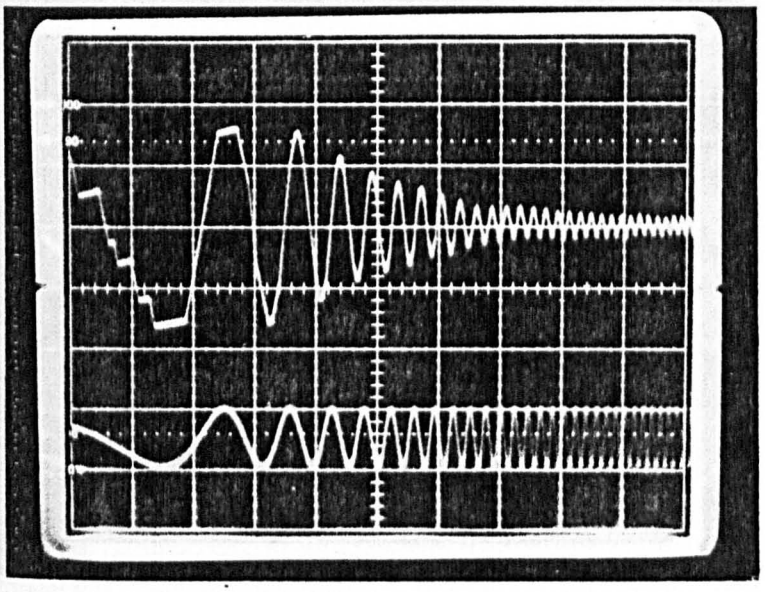


PHOTO 7.5

X - Sweep Voltage

Y_1 - 0.2V / div

Y_2 - 1.0V / div

Envelope Response at Port 2 with R = 50 ohms

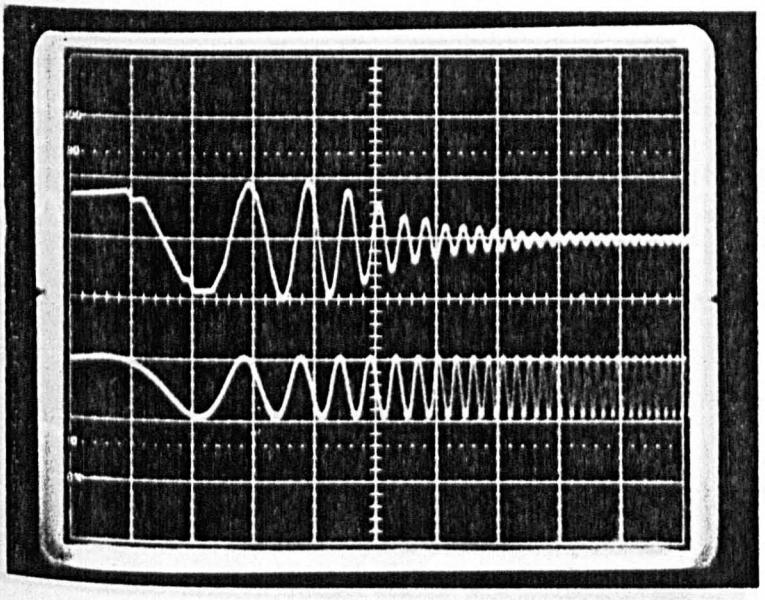


PHOTO 7.6

X - Sweep Voltage

Y_1 - 0.5V / div

Y_2 - 1.0V / div

Envelope Response at Port 2 with R = 100 ohms

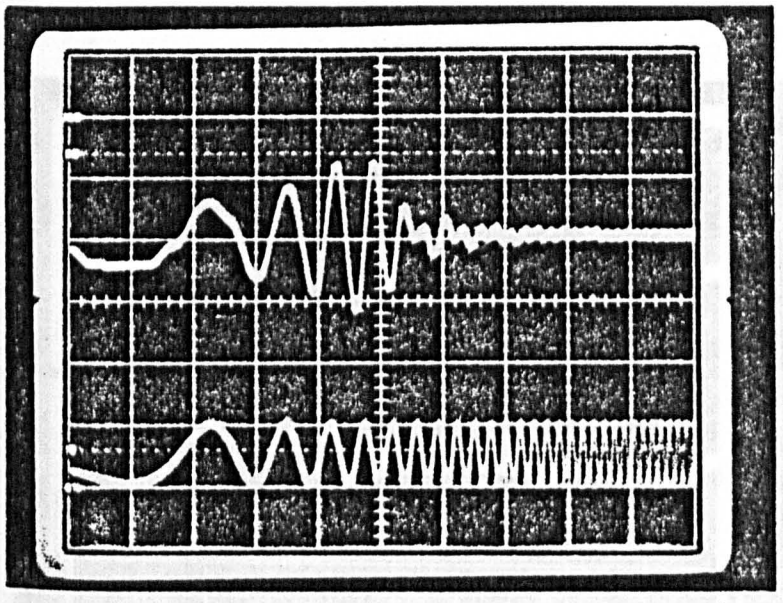


PHOTO 7.7

X - Sweep Voltage

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Envelope Response at Port 2 with R = 200 ohms

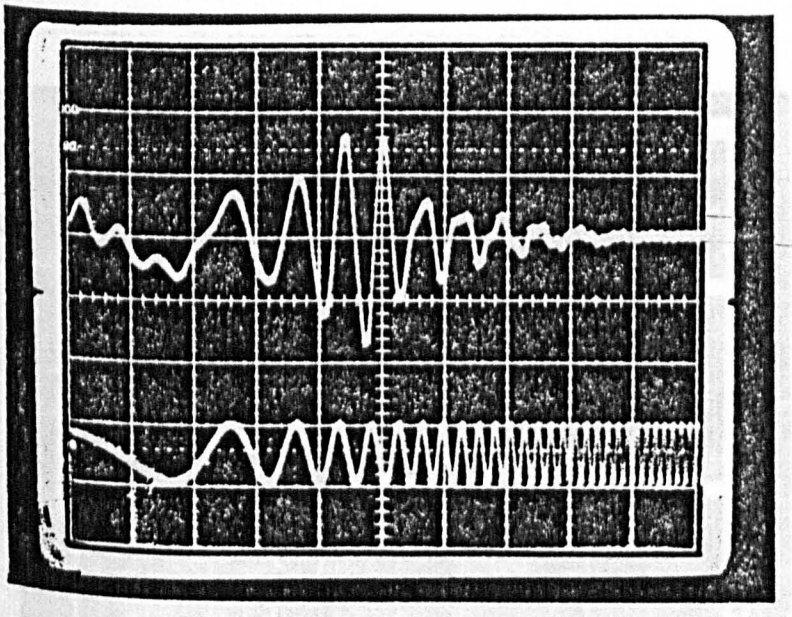


PHOTO 7.8

X - Sweep Voltage

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Envelope Response at Port 2 with R = 300 ohms

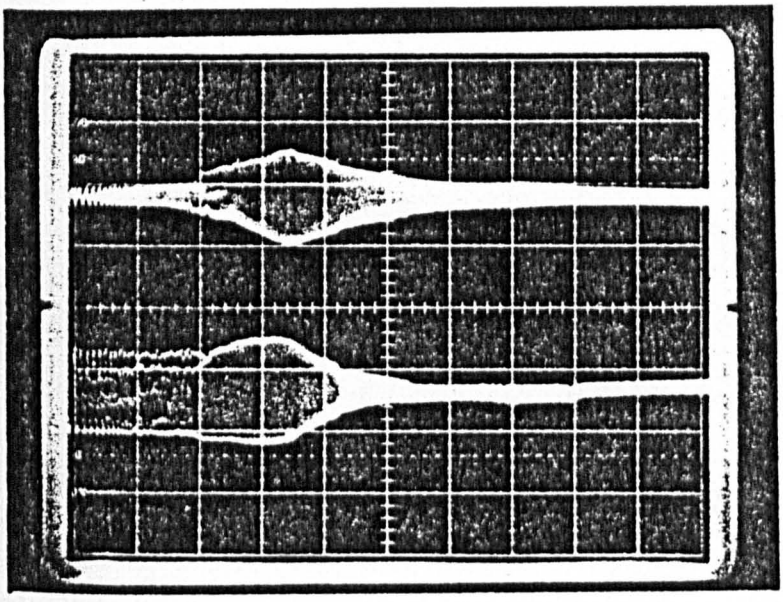


PHOTO 7.9

X - Sweep Voltage

Y_1 - 0.5V / div

Y_2 - 0.5V / div

Relative Envelope Response between Ports 1 and 2 with R = 100 ohms

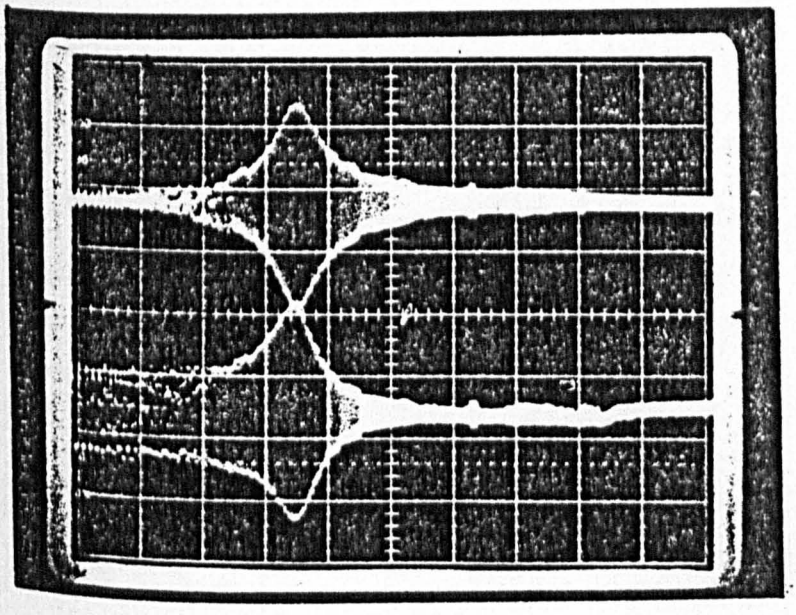


PHOTO 7.10

X - Sweep Voltage

Y_1 - 0.5V / div

Y_2 - 0.5V / div

Relative Envelope Response between Ports 1 and 2 with R = 200 ohms

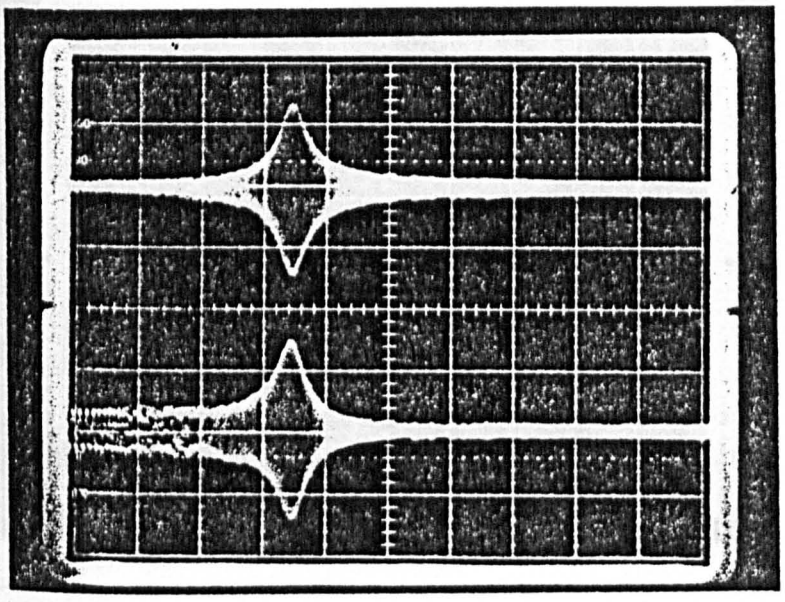


PHOTO 7.11

X - Sweep Voltage

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Relative Envelope Response between Ports 1 and 2 with R = 300 ohms

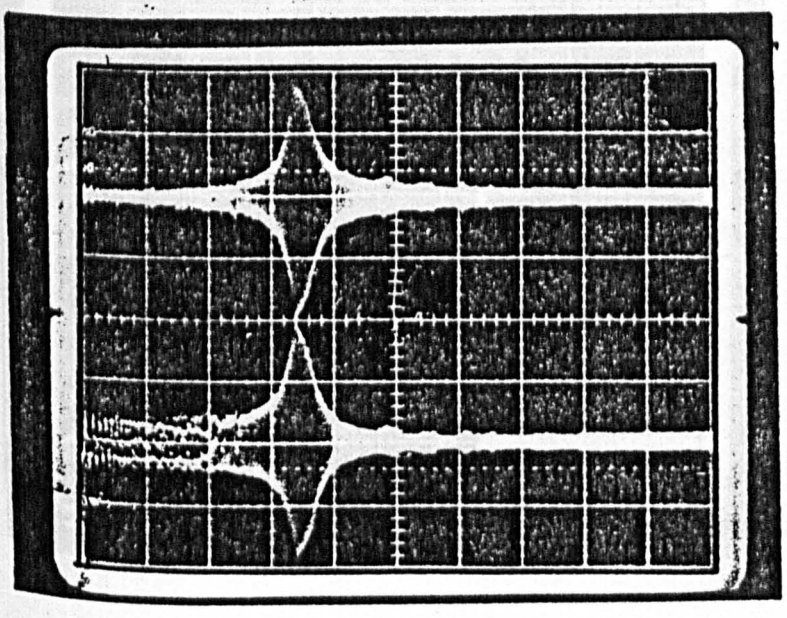


PHOTO 7.12

X - Sweep Voltage

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Relative Envelope Response between Ports 1 and 2 with R = 400 ohms

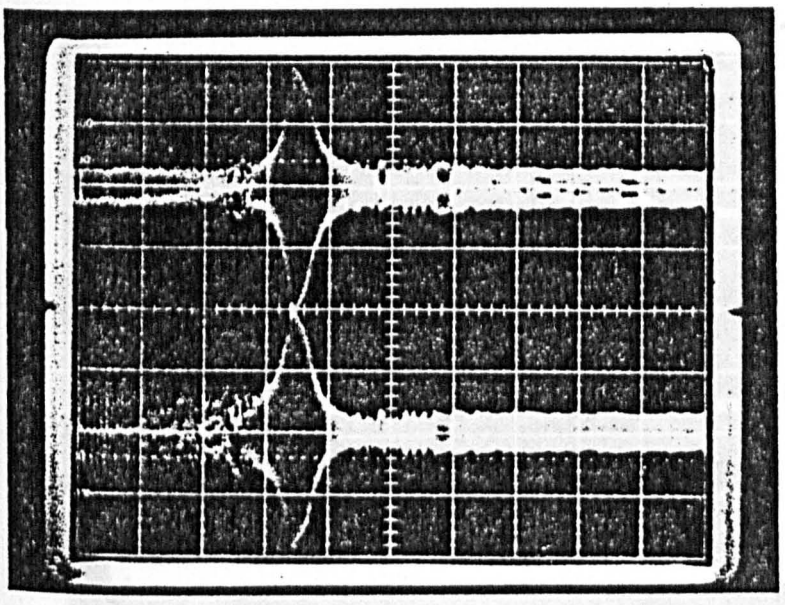


PHOTO 7.13

X - Sweep Voltage

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Relative Envelope Response between Ports 1 and 2 with R = 500 ohms

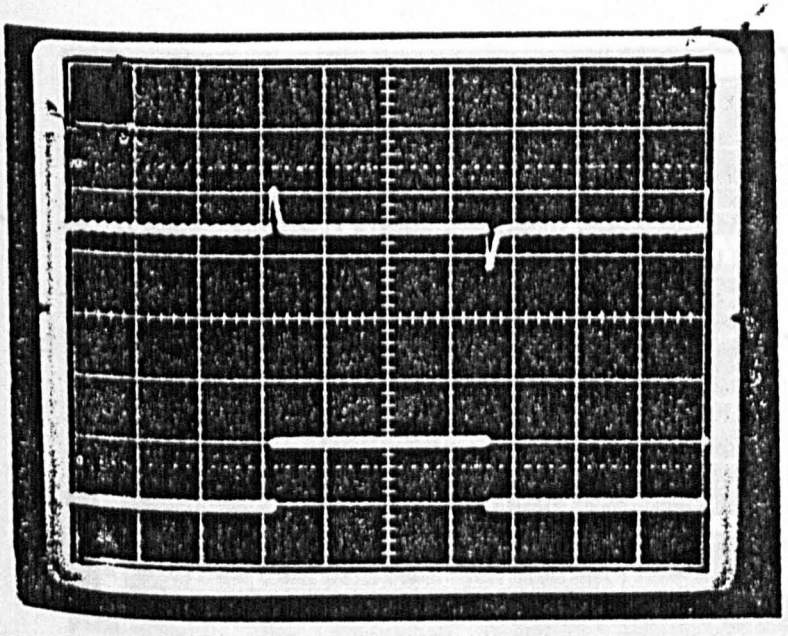


PHOTO 7. 14

X - 5ms / div

Y_1 - 0.5V / div

Y_2 - 0.5V / div

Impulse Response at Port 1 with R = 50 ohms

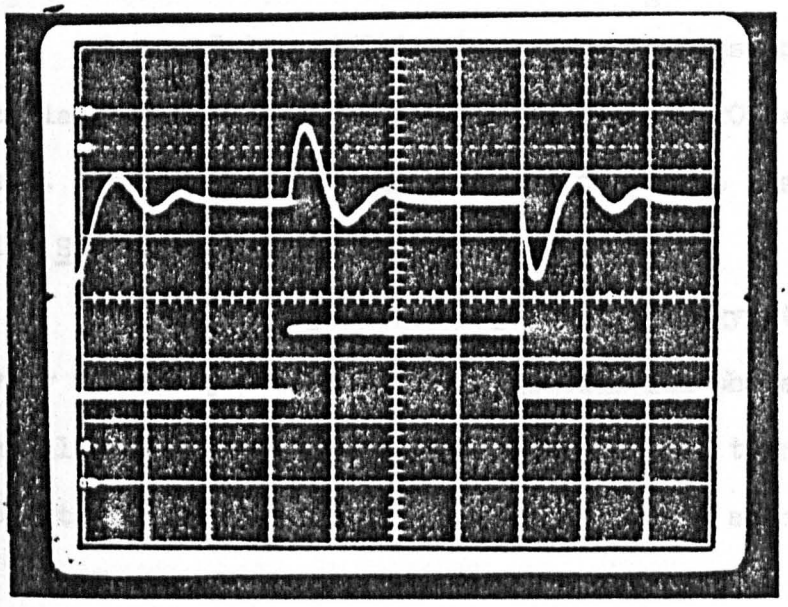


PHOTO 7.15

X - 5ms / div

Y_1 - 0.5V / div

Y_2 - 1.0V / div

Impulse Response at Port 1 with R = 100 ohms

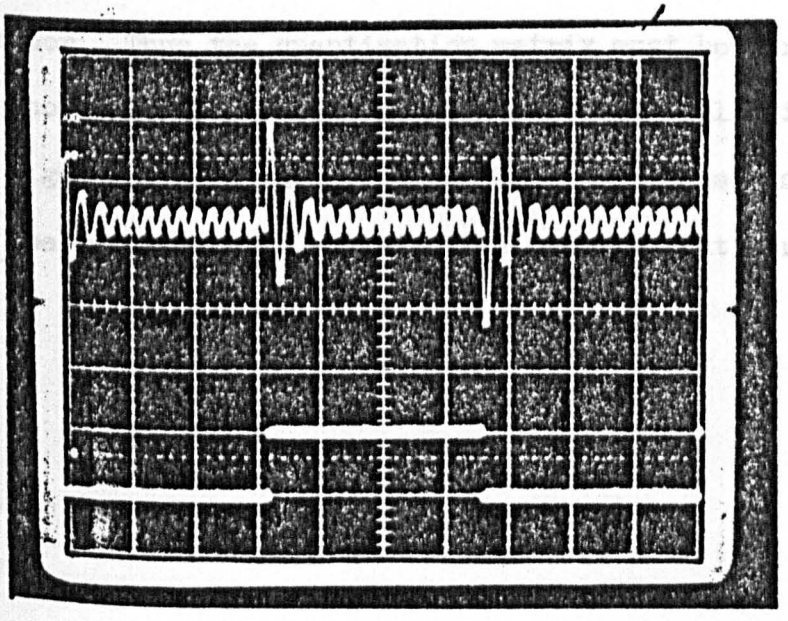


PHOTO 7.16

X - 20ms / div

Y_1 - 0.5V / div

Y_2 - 1.0V / div

Impulse Response at Port 1 with R = 200 ohms

the arrangement of equipment in Fig. 7.8. Photos 7.18 to 7.21 show the voltage at Port 2 (upper trace) in response to a step change in input current to Port 1 for 4 values of R_S.

Photos 7.20 and 7.21 also show the presence of limit cycle oscillations because the shunt resistance is 200 and 300 ohms respectively. The rms limit cycle noise voltages are listed in Table 7.18.

7.10 SUMMARY AND CONCLUSIONS

A capacitively loaded 2-port digital gyrator has been quantitatively tested by measuring the magnitude and phase responses of the digital input impedance (z_{11}) and the forward transimpedance (z_{21}) with a constant sampling frequency and 6 values of shunt resistance.

Photographs have been used to demonstrate the machine performance and to qualitatively verify the experimental frequency responses and impulse responses.

The presence of a quantisation oscillation has been verified and shown photographically and the effects of this on the measured results calculated. It is clear from these results that this quantisation oscillation sets an awkward limit on the usefulness of the digital gyrator. Thus the quantisation matrix must be derived and evaluated to show whether any form of quantisation oscillation will be present in a given digital active network. An increase in the number of bits in the word will not obviate this problem, but merely reduce it.

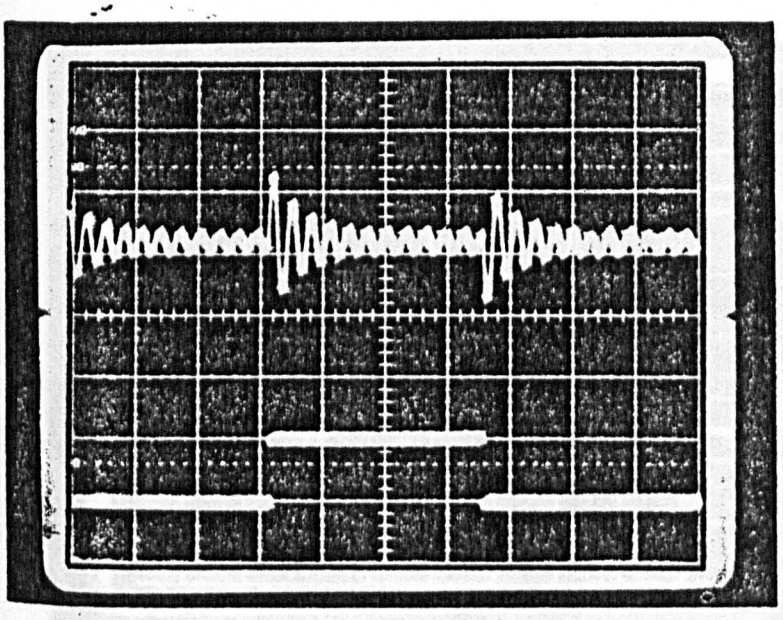


PHOTO 7.17

X - 20ms / div

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Impulse Response at Port 1 with R = 300 ohms

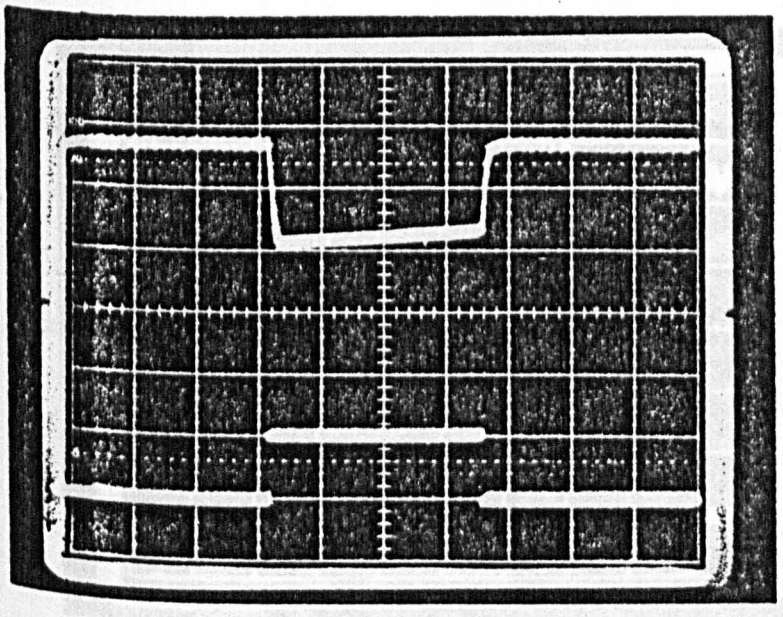


PHOTO 7.18

X - 5ms / div

Y_1 - 0.5V / div

Y_2 - 0.5V / div

Impulse Response at Port 2 with R = 50 ohms

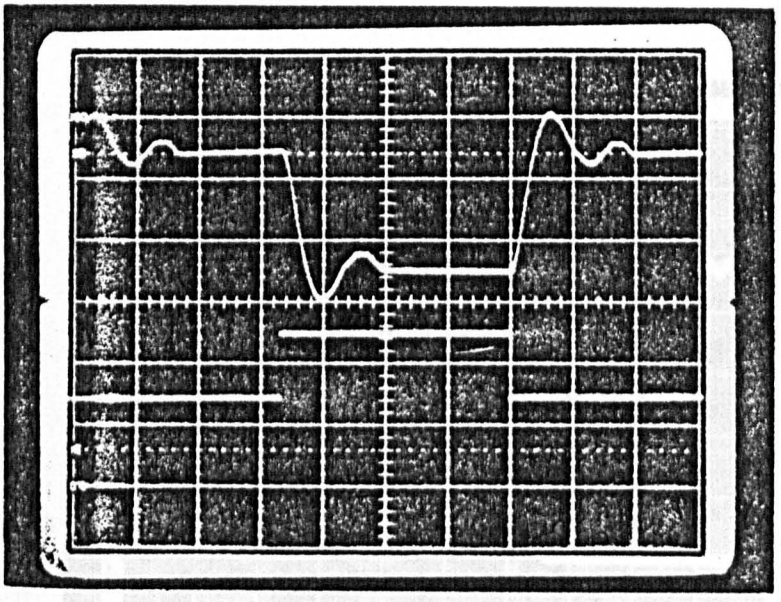


PHOTO 7.19

X - 5ms / div

Y_1 - 0.5V / div

Y_2 - 1.0V / div

Impulse Response at Port 2 with R = 100 ohms

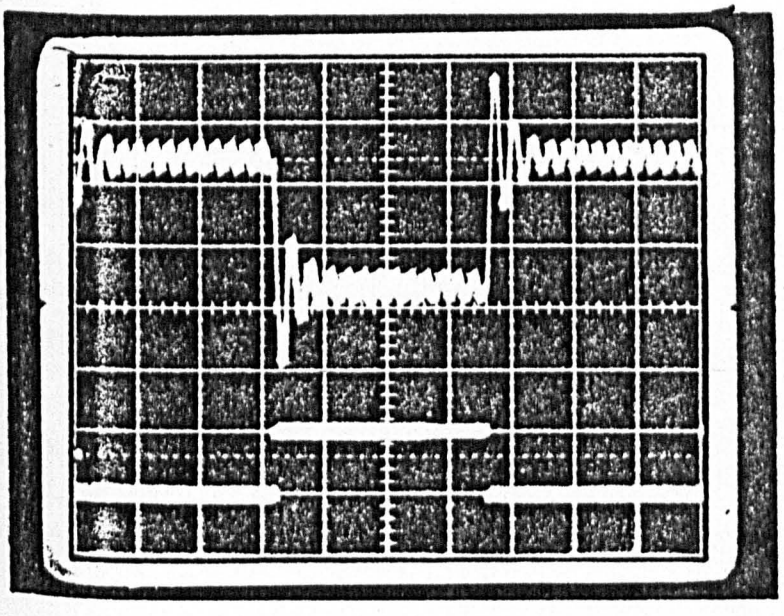


PHOTO 7.20

X - 20ms / div

Y_1 - 0.5V / div

Y_2 - 0.5V / div

Impulse Response at Port 2 with R = 200 ohms

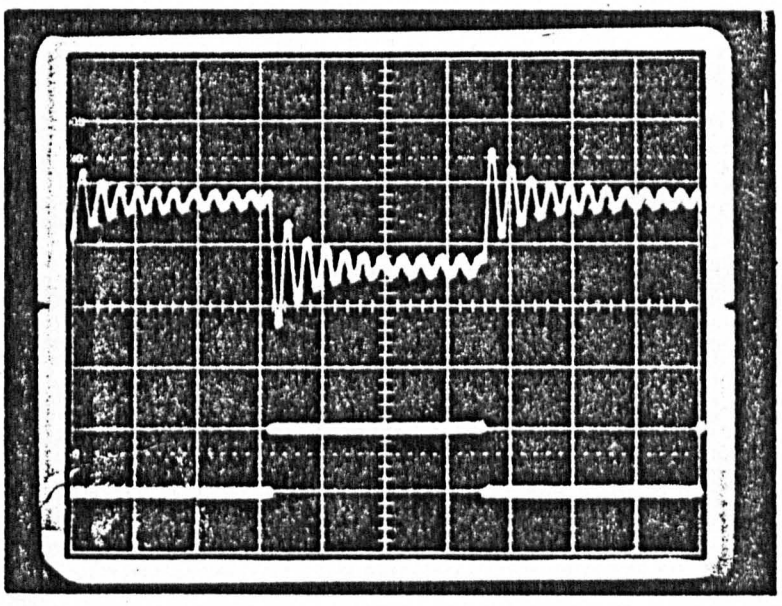


PHOTO 7.21

X - 20ms / div

Y_1 - 1.0V / div

Y_2 - 1.0V / div

Impulse Response at Port 2 with R = 300 ohms

CHAPTER 8

COMPUTER RESULTS

8.1 INTRODUCTION

The results obtained from the computer programs described in Chapter 5 are presented in this chapter in the same sequence in which the programs were described.

It should be noted that in the graphical results the linear axes are marked with scale factors having powers of the opposite sign to that expected. This is a deliberate property of the graphics plotting library PLOTTER package (Appendix D) which is installed on the mini-computer system. The scale factor as marked should be interpreted as a factor that has been used to scale the data plotted against that axis rather than as a factor to multiply the scale markings on that axis.

8.2 GYRATOR DATA PREPARATION PROGRAM (GDP1)

GDP1 was written specifically to evaluate the elements of the 2-port digital admittance, impedance and quantisation matrices for the digital gyrator analysed in Chapter 4, and shown in Fig. 4.1.

The digital admittance and impedance matrices (4.13) and (4.15) are repeated here for convenient reference:

$$Y(z) = \begin{bmatrix} \dfrac{C_1 (z - \alpha)}{T_s z} & \dfrac{g_1 C_1 (1 - \alpha)}{g T_s z^{k_1 + 1}} \\ -g_2 \dfrac{1}{z^{k_2 + 1}} & \dfrac{C_2}{T_s} \dfrac{(z - 1)}{z} \end{bmatrix} \tag{8.1}$$

CHAPTER 8

COMPUTER RESULTS

8.1 INTRODUCTION

The results obtained from the computer programs described in Chapter 5 are presented in this chapter in the same sequence in which the programs were described.

It should be noted that in the graphical results the linear axes are marked with scale factors having powers of the opposite sign to that expected. This is a deliberate property of the graphics plotting library PLOTTER package (Appendix D) which is installed on the mini-computer system. The scale factor as marked should be interpreted as a factor that has been used to scale the data plotted against that axis rather than as a factor to multiply the scale markings on that axis.

8.2 GYRATOR DATA PREPARATION PROGRAM (GDP1)

GDP1 was written specifically to evaluate the elements of the 2-port digital admittance, impedance and quantisation matrices for the digital gyrator analysed in Chapter 4, and shown in Fig. 4.1 .

The digital admittance and impedance matrices (4.13) and (4.15) are repeated here for convenient reference:

$$Y(z) = \begin{bmatrix} \dfrac{C_1 (z - \alpha)}{T_s z} & \dfrac{g_1 C_1 (1 - \alpha)}{g T_s z^{k_1 + 1}} \\ -g_2 \dfrac{1}{z^{k_2 + 1}} & \dfrac{C_2}{T_s} \dfrac{(z - 1)}{z} \end{bmatrix} \qquad (8.1)$$

$$Z(z) = \frac{\begin{bmatrix} \frac{T_s}{C_1} z^{k_1+k_2+1}(z-1) & \frac{-g_1 T_s(1-\alpha)}{g C_2} z^{k_2+1} \\ \frac{g_2 T_s^2}{C_1 C_2} z^{k_1+1} & \frac{T_s}{C_2}(z-\alpha) z^{k_1+k_2+1} \end{bmatrix}}{z^{k_1+k_2+2} - z^{k_1+k_2+1}(1+\alpha) + z^{k_1+k_2} + \frac{T_s(1-\alpha)g_1 g_2}{g C_2}} \quad (8.2)$$

where $\alpha = \exp(-T_s/T)$

$= \exp(-T_s g/C_1)$

The range of values chosen for study are listed in Table 8.1. The shunt conductance g was replaced by the equivalent resistance R where $R = 1/g$. 30 different polynomials were calculated for each of the 8 matrix elements, giving 240 polynomials in all.

GDP1 was written to produce only intermediate results ready for analysis by PZP1, FRA1 and IZT1 but the actual calculated denominator polynomial coefficients for the impedance matrix are listed in Tables 8.2 to 8.31 under the results for PZP1. The remainder of the polynomial coefficients are not listed because they in themselves convey little information.

8.3 POLE-ZERO PLOTTING PROGRAM (PZP1)

PZP1 calculated the roots of the polynomials evaluated by GDP1 and then tabulated and plotted these results.

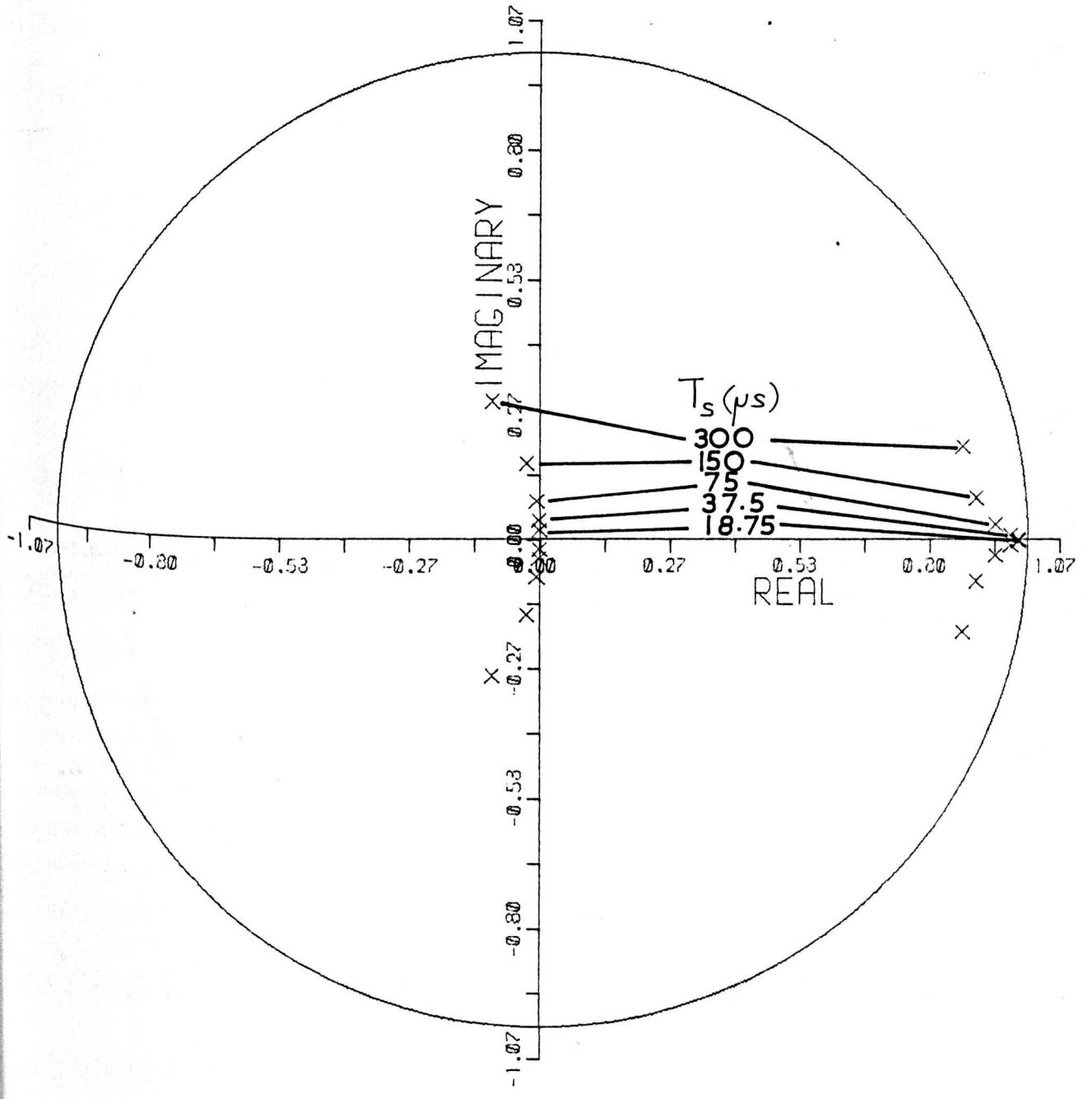
From the admittance matrix (8.1) it can be seen that the poles and zeroes of $y_{12}(z)$, $y_{21}(z)$ and $y_{22}(z)$ and the zero of $y_{11}(z)$ may be obtained by inspection. The pole of $y_{11}(z)$ may also be obtained by inspection by observing that α is the coefficient of the second term of the denominator of the impedance matrix (8.2) and this is listed

PARAMETER	PARAMETER MEANING	VALUE
f_s	Sampling frequency	26.6 KHz
T_s	Sampling period	18.75 microseconds
C_1	Port 1 shunt capacitor	9.6 microfarads
C_2	Port 2 shunt capacitor	10 microfarads
g_1, g_2	Transconductances	10 millisiemens
R_s	Port 1 shunt resistor	50, 100, 200, 300, 400, 500 ohms
k_1, k_2	Fractional delay	1

TABLE 8.1

Digital Gyrator Component Values

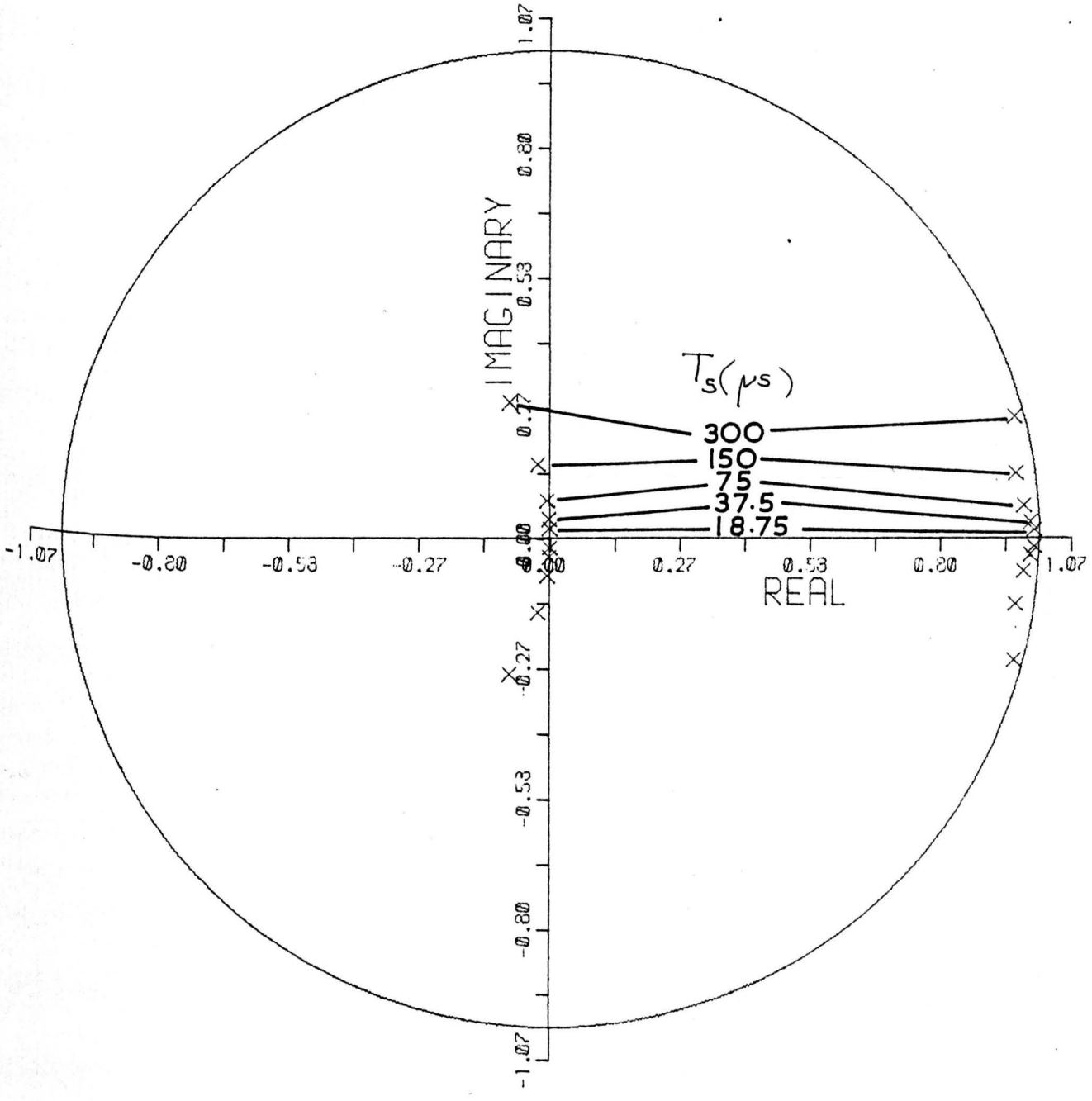
Z-PLANE AND UNIT-CIRCLE



POLE PLOT WITH SHUNT RESISTOR = 50 OHMS
showing pole migration with all 5 values of T_s.

GRAPH 8.1

Z-PLANE AND UNIT-CIRCLE

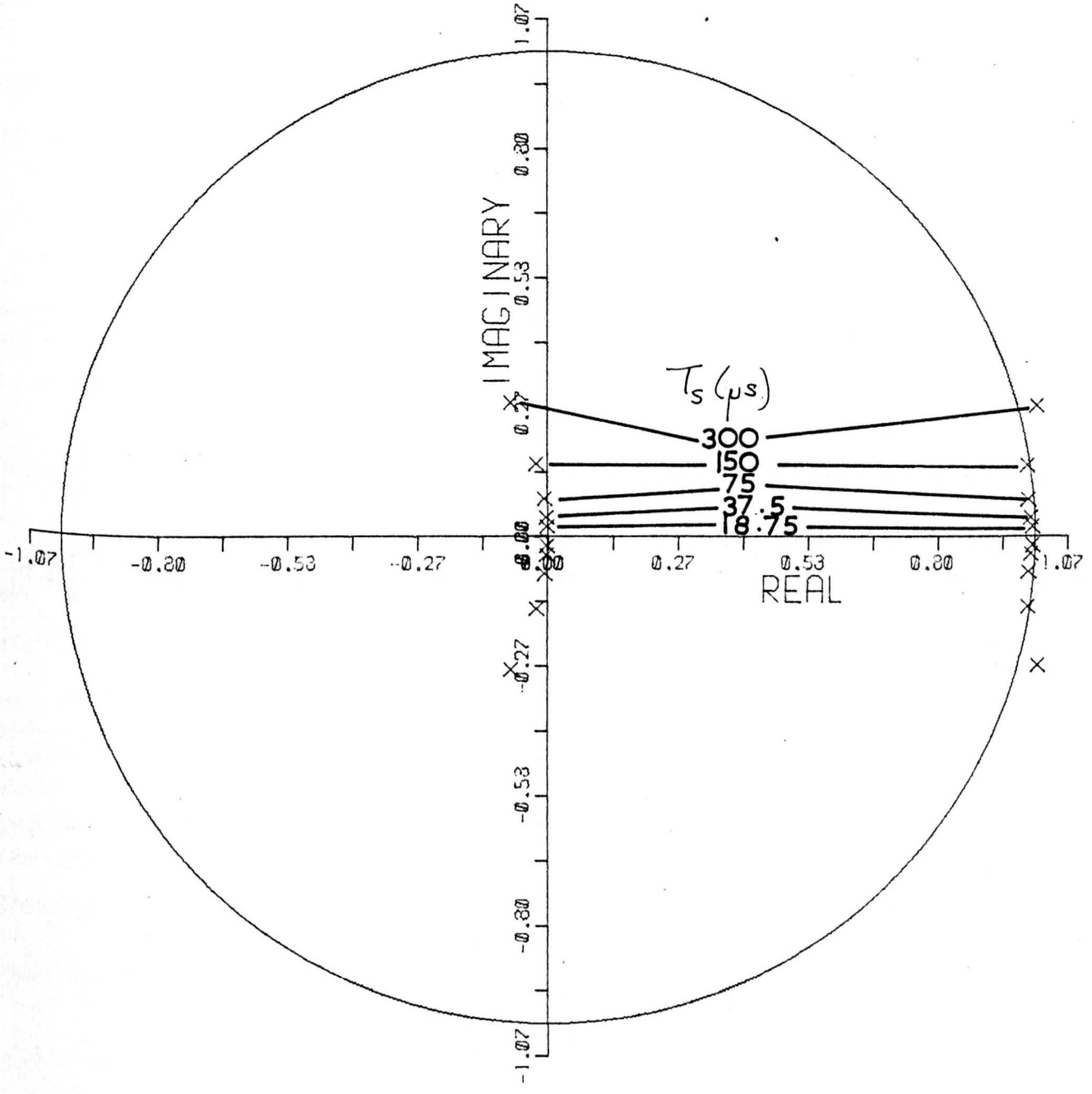


POLE PLOT WITH SHUNT RESISTOR = 100 OHMS

showing pole migration with all 5 values of T_s

GRAPH 8.2

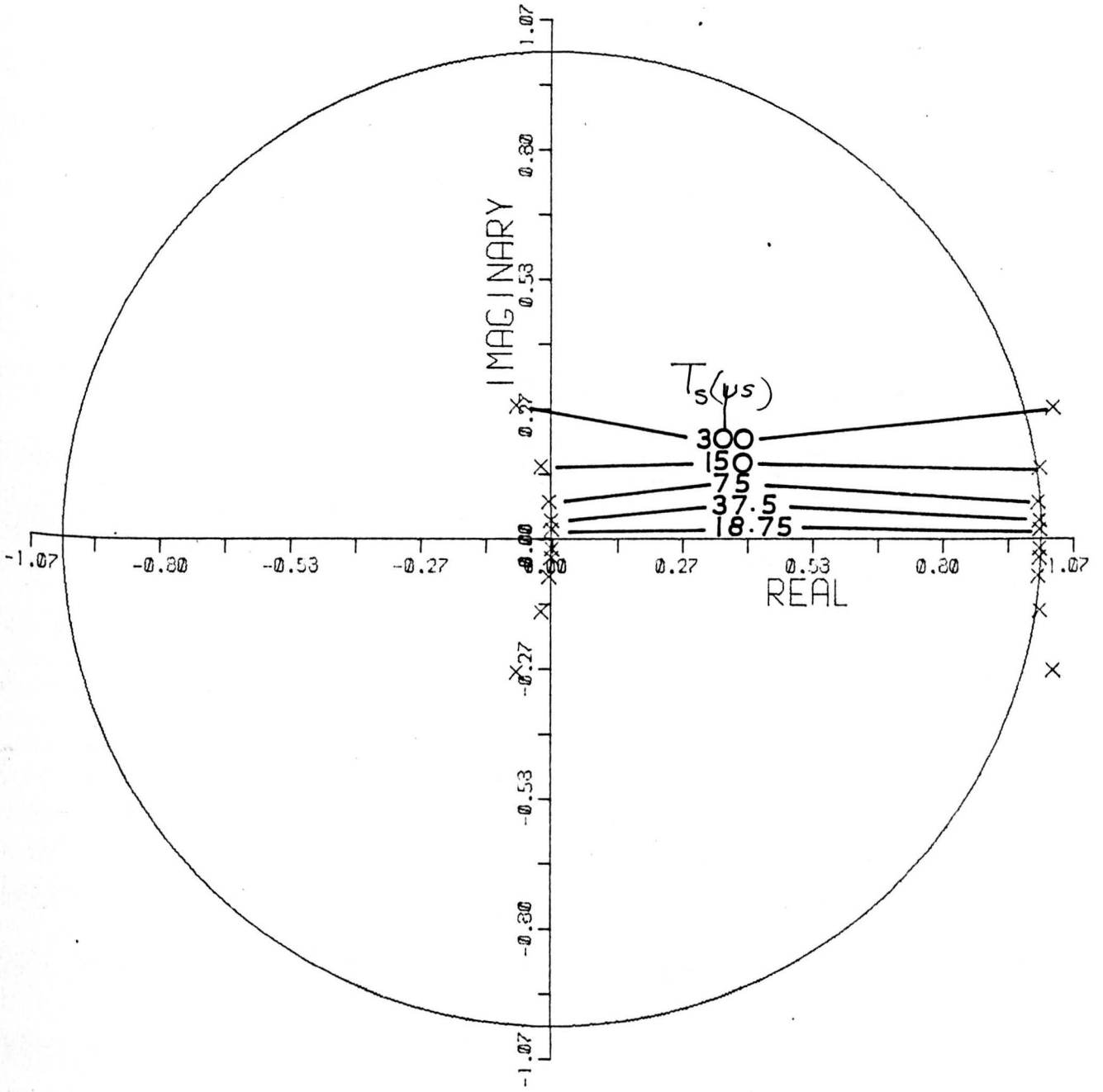
Z-PLANE AND UNIT-CIRCLE



POLE PLOT WITH SHUNT RESISTOR = 200 OHMS
showing pole migration with all 5 values of T_s

GRAPH 8.3

Z-PLANE AND UNIT-CIRCLE

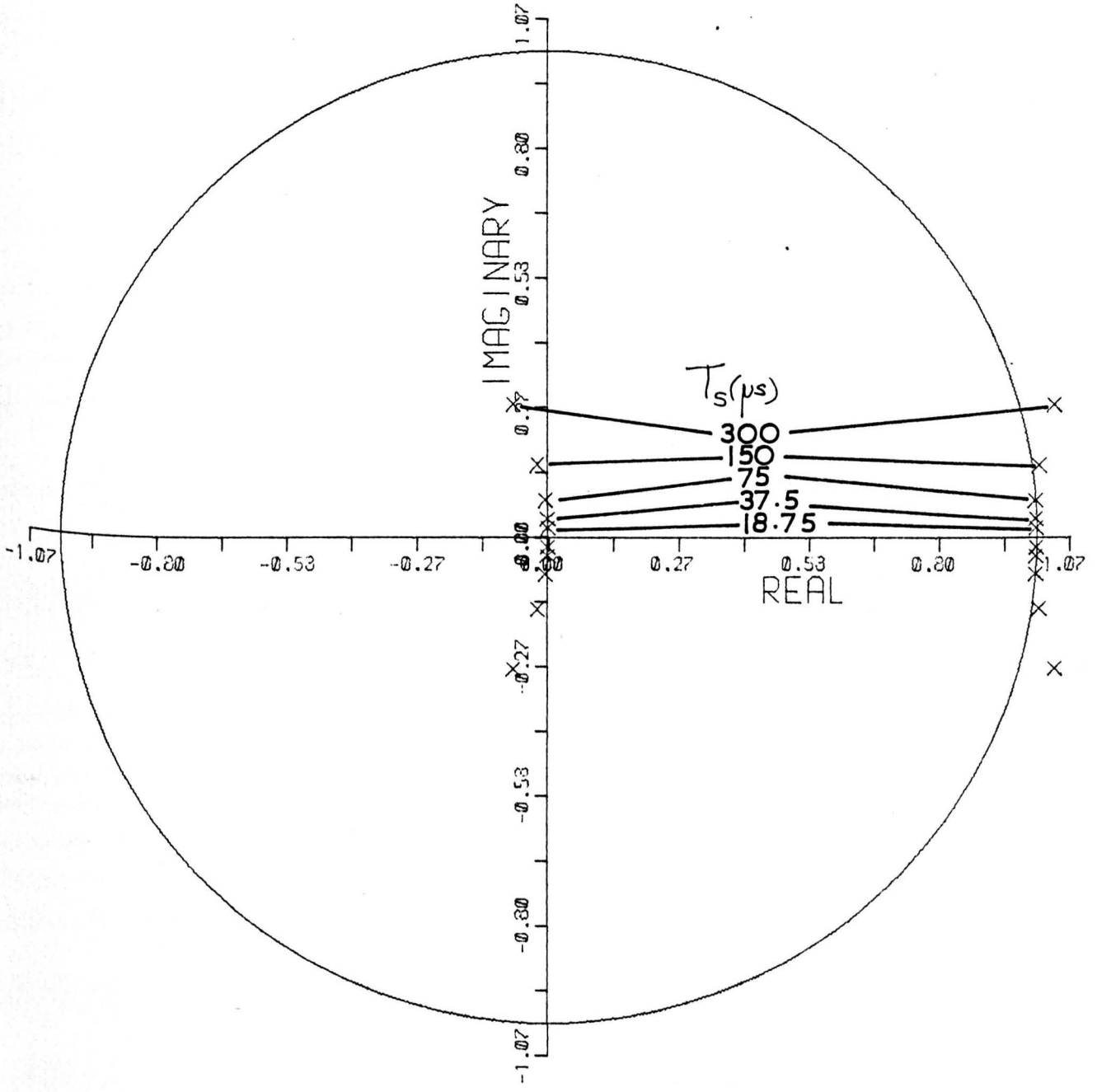


POLE PLOT WITH SHUNT RESISTOR = 300 OHMS

showing pole migration with all 5 values of T_s

GRAPH 8.4

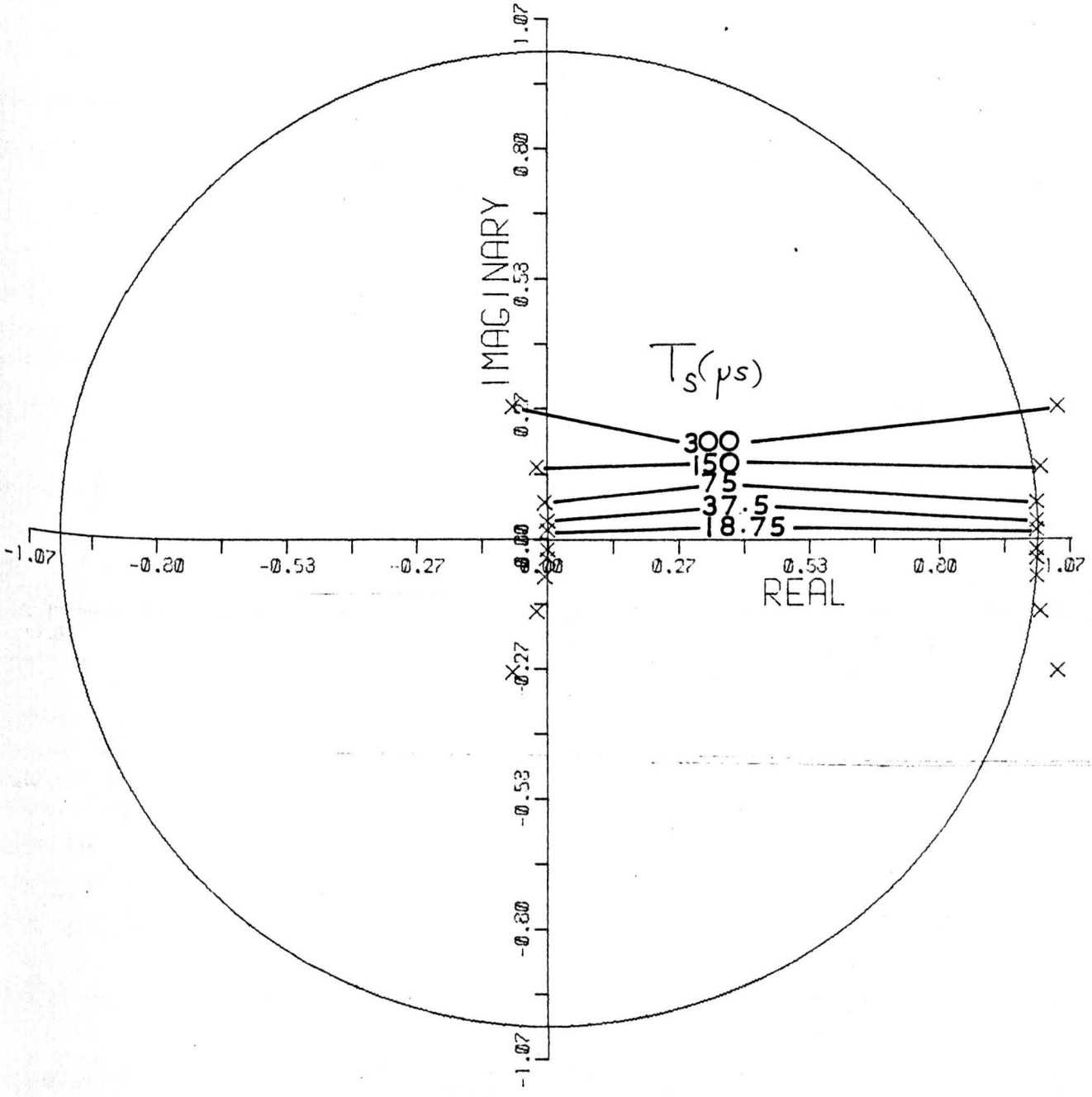
Z-PLANE AND UNIT-CIRCLE



POLE PLOT WITH SHUNT RESISTOR = 400 OHMS
showing pole migration with all 5 values of T_s

GRAPH 8.5

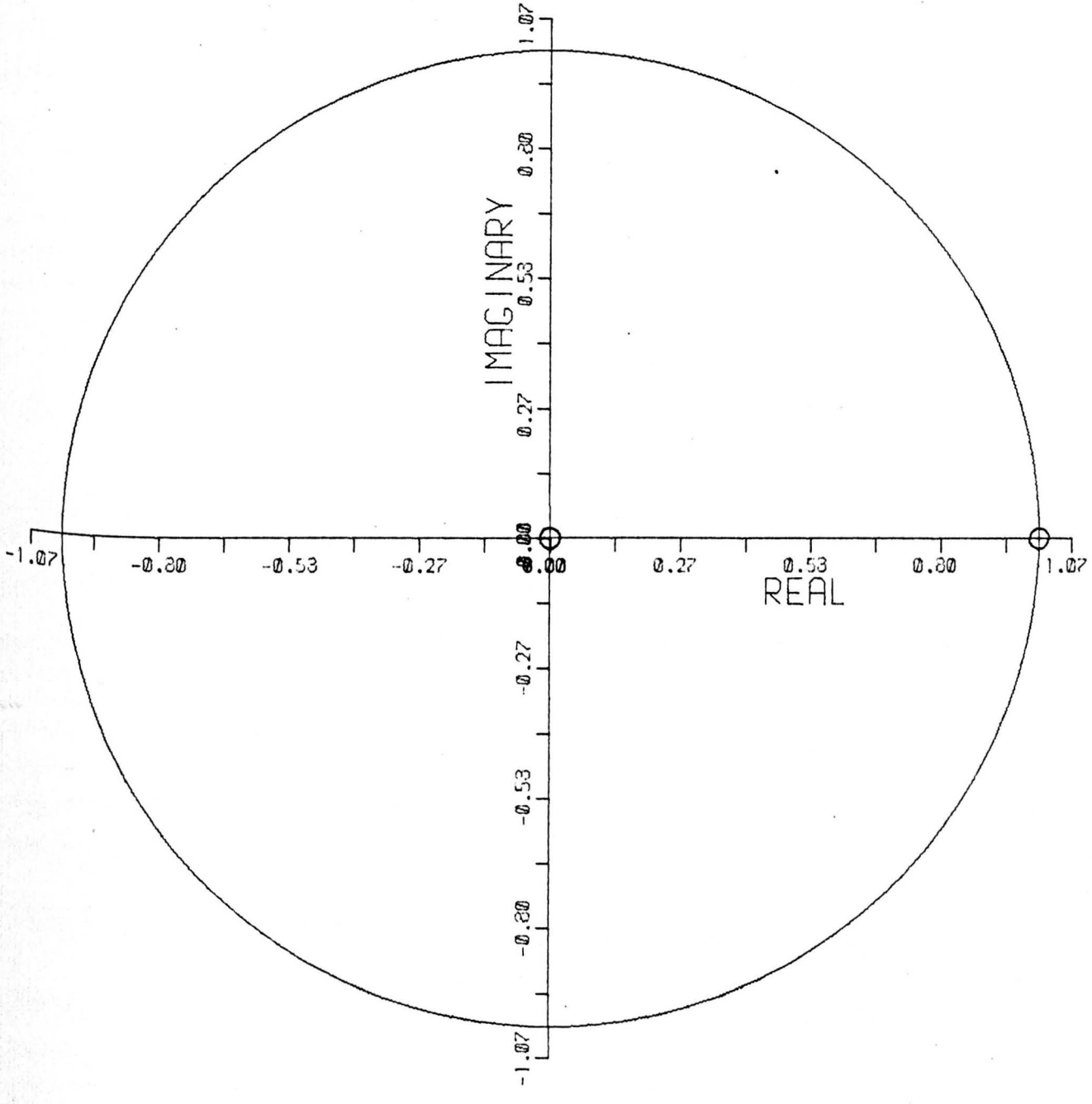
Z-PLANE AND UNIT-CIRCLE



POLE PLOT WITH SHUNT RESISTOR = 500 OHMS
showing pole migration with all 5 values of T_s

GRAPH 8.6

Z-PLANE AND UNIT-CIRCLE



ZERO PLOT FOR ALL SAMPLE FREQUENCIES
and all 6 values of shunt resistance R_s

GRAPH 8.7

in Tables 8.2 to 8.31.

From the impedance matrix (8.2) the zeros may all be found by inspection, α being found as described above. However, the denominator of this matrix is a fourth order polynomial when $k_1 = k_2 = 1$ (Table 8.1) and hence the poles cannot be found by inspection. Thus for brevity only the poles of the impedance matrix are listed as results.

Tables 8.2 to 8.31 list the parameters, coefficients and roots of the denominator of the digital impedance matrix (8.2). The roots are listed in cartesian and polar coordinates, the modulus column being particularly useful for checking whether the pole lies outside the unit circle and Table 8.32 lists the parameters for the cases when this is true. Graphs 8.1 to 8.7 show the poles and zeroes.

8.4 FREQUENCY RESPONSE ANALYSIS PROGRAM (FRA1)

FRA1 was written to calculate the frequency response of any s or z-plane polynomial, but in particular the polynomials evaluated by GDP1, and then to list and plot these results. For the sake of brevity, only the Bodé magnitude and phase plots are presented here.

Each frequency response shown has a logarithmic frequency axis with values between 5 Hz and 5 kHz, but a linear amplitude axis in ohms or siemens as appropriate between zero and a convenient maximum.

FRA1 evaluated the frequency response of all four elements of the digital impedance matrix (8.2) for all the parameter values listed in Table 8.1 and plotted them on Graphs 8.8 to 8.27. Each graph shows six curves for the six values of the shunt resistance R with a constant value of sampling period T_s which is stated on the bottom of each graph.

Each plot for each of the digital impedance matrix elements shows (Graphs 8.8 to 8.27) a peak at around 160 Hz, the exact peaks being shown in Tables 8.33 and 8.34. It should be noted from these graphs

PARAMETERS

SAMPLING PERIOD	T	=	0.1875E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	50.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.196169E+01
2	0.961691E+00
1	0.0
0	0.359151E-03

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.981226E+00	J*	0.248976E-02	0.981229E+00	0.145382E+00
0.981226E+00	J*	-.248976E-02	0.981229E+00	-.145382E+00
-.380158E-03	J*	0.193096E-01	0.193134E-01	0.911289E+02
-.380158E-03	J*	-.193096E-01	0.193134E-01	-.911289E+02

TABLE 8.2

PARAMETERS

SAMPLING PERIOD	T	=	0.1875E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	100.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.198066E+01
2	0.980658E+00
1	0.0
0	0.362657E-03

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.990702E+00	J*	0.166087E-01	0.990841E+00	0.960452E+00
0.990702E+00	J*	-.166087E-01	0.990841E+00	-.960452E+00
-.372767E-03	J*	0.192169E-01	0.192206E-01	0.911123E+02
-.372767E-03	J*	-.192169E-01	0.192206E-01	-.911123E+02

TABLE 8.3

PARAMETERS

SAMPLING PERIOD	T	=	0.1875E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	200.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.199028E+01
2	0.990282E+00
1	0.0
0	0.364427E-03

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.995510E+00	J*	0.185483E-01	0.995683E+00	0.106741E+01
0.995510E+00	J*	-.185483E-01	0.995683E+00	-.106741E+01
-.369072E-03	J*	0.191689E-01	0.191724E-01	0.911040E+02
-.369072E-03	J*	-.191689E-01	0.191724E-01	-.911040E+02

TABLE 8.4

PARAMETERS

SAMPLING PERIOD	T	=	0.1875E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	300.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.199351E+01
2	0.993511E+00
1	0.0
0	0.365022E-03

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.997123E+00	J*	0.188858E-01	0.997302E+00	0.108507E+01
0.997123E+00	J*	-.188858E-01	0.997302E+00	-.108507E+01
-.367940E-03	J*	0.191533E-01	0.191569E-01	0.911016E+02
-.367940E-03	J*	-.191533E-01	0.191569E-01	-.911016E+02

TABLE 8.5

PARAMETERS

SAMPLING PERIOD	T	=	0.1875E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	400.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.199513E+01
2	0.995129E+00
1	0.0
0	0.365317E-03

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.997932E+00	J*	0.189928E-01	0.998113E+00	0.109033E+01
0.997932E+00	J*	-.189928E-01	0.998113E+00	-.109033E+01
-.367343E-03	J*	0.191456E-01	0.191491E-01	0.911002E+02
-.367343E-03	J*	-.191456E-01	0.191491E-01	-.911002E+02

TABLE 8.6

PARAMETERS

SAMPLING PERIOD	T	=	0.1875E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	500.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.199610E+01
2	0.996101E+00
1	0.0
0	0.365496E-03

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.998418E+00	J*	0.190461E-01	0.998599E+00	0.109286E+01
0.998418E+00	J*	-.190461E-01	0.998599E+00	-.109286E+01
-.366926E-03	J*	0.191409E-01	0.191444E-01	0.910993E+02
-.366926E-03	J*	-.191409E-01	0.191444E-01	-.910993E+02

TABLE 8.7

PARAMETERS

SAMPLING PERIOD	T	=	0.3750E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	50.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.192485E+01
2	0.924849E+00
1	0.0
0	0.140908E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.963997E+00	J*	0.103177E-01	0.964052E+00	0.613213E+00
0.963997E+00	J*	-.103177E-01	0.964052E+00	-.613213E+00
-.157261E-02	J*	0.389062E-01	0.389379E-01	0.923153E+02
-.157261E-02	J*	-.389062E-01	0.389379E-01	-.923153E+02

TABLE 8.8

PARAMETERS

SAMPLING PERIOD	T	=	0.3750E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	100.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.196169E+01
2	0.961691E+00
1	0.0
0	0.143660E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.982357E+00	J*	0.334517E-01	0.982927E+00	0.195031E+01
0.982357E+00	J*	-.334517E-01	0.982927E+00	-.195031E+01
-.151187E-02	J*	0.385314E-01	0.385611E-01	0.922477E+02
-.151187E-02	J*	-.385314E-01	0.385611E-01	-.922477E+02

TABLE 8.9

PARAMETERS

SAMPLING PERIOD	T =	0.3750E-04	SECONDS
SHUNT CAPACITOR	C1 =	0.9600E-05	FARADS
SHUNT CAPACITOR	C2 =	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1 =	0.0100	SIEMENS
TRANSCONDUCTANCE	G2 =	0.0100	SIEMENS
SHUNT RESISTOR	R =	200.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.198066E+01
2	0.980658E+00
1	0.0
0	0.145063E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.991812E+00	J*	0.371142E-01	0.992506E+00	0.214304E+01
0.991812E+00	J*	-.371142E-01	0.992506E+00	-.214304E+01
-.148267E-02	J*	0.383457E-01	0.383744E-01	0.922150E+02
-.148267E-02	J*	-.383457E-01	0.383744E-01	-.922150E+02

TABLE 8.10

PARAMETERS

SAMPLING PERIOD	T =	0.3750E-04	SECONDS
SHUNT CAPACITOR	C1 =	0.9600E-05	FARADS
SHUNT CAPACITOR	C2 =	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1 =	0.0100	SIEMENS
TRANSCONDUCTANCE	G2 =	0.0100	SIEMENS
SHUNT RESISTOR	R =	300.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.198706E+01
2	0.987064E+00
1	0.0
0	0.145535E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.995005E+00	J*	0.377385E-01	0.995720E+00	0.217207E+01
0.995005E+00	J*	-.377385E-01	0.995720E+00	-.217207E+01
-.147313E-02	J*	0.382846E-01	0.383130E-01	0.922043E+02
-.147313E-02	J*	-.382846E-01	0.383130E-01	-.922043E+02

TABLE 8.11

PARAMETERS

SAMPLING PERIOD	T	=	0.3750E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	400.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.199028E+01
2	0.990282E+00
1	0.0
0	0.145771E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.996609E+00	J*	0.379480E-01	0.997332E+00	0.218060E+01
0.996609E+00	J*	-.379480E-01	0.997332E+00	-.218060E+01
-.146836E-02	J*	0.382537E-01	0.382818E-01	0.921989E+02
-.146836E-02	J*	-.382537E-01	0.382818E-01	-.921989E+02

TABLE 8.12

PARAMETERS

SAMPLING PERIOD	T	=	0.3750E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	500.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.199222E+01
2	0.992218E+00
1	0.0
0	0.145914E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.997575E+00	J*	0.380374E-01	0.998300E+00	0.218362E+01
0.997575E+00	J*	-.380374E-01	0.998300E+00	-.218362E+01
-.146556E-02	J*	0.382359E-01	0.382639E-01	0.921957E+02
-.146556E-02	J*	-.382359E-01	0.382639E-01	-.921957E+02

TABLE 8.13

PARAMETERS

SAMPLING PERIOD	T	=	0.7500E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	50.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.185535E+01
2	0.855345E+00
1	0.0
0	0.542455E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.934309E+00	J*	0.317101E-01	0.934847E+00	0.194385E+01
0.934309E+00	J*	-.317101E-01	0.934847E+00	-.194385E+01
-.663579E-02	J*	0.785049E-01	0.787848E-01	0.948319E+02
-.663579E-02	J*	-.785049E-01	0.787848E-01	-.948319E+02

TABLE 8.14

PARAMETERS

SAMPLING PERIOD	T	=	0.7500E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	100.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.192485E+01
2	0.924849E+00
1	0.0
0	0.563634E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.968568E+00	J*	0.674350E-01	0.970913E+00	0.398270E+01
0.968568E+00	J*	-.674350E-01	0.970913E+00	-.398270E+01
-.614339E-02	J*	0.770803E-01	0.773248E-01	0.945572E+02
-.614339E-02	J*	-.770803E-01	0.773248E-01	-.945572E+02

TABLE 8.15

PARAMETERS

SAMPLING PERIOD	T	=	0.7500E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	200.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.196169E+01
2	0.961691E+00
1	0.0
0	0.574642E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.986760E+00	J*	0.739670E-01	0.989528E+00	0.428684E+01
0.986760E+00	J*	-.739670E-01	0.989528E+00	-.428684E+01
-.591421E-02	J*	0.763786E-01	0.766073E-01	0.944281E+02
-.591421E-02	J*	-.763786E-01	0.766073E-01	-.944281E+02

TABLE 8.16

PARAMETERS

SAMPLING PERIOD	T	=	0.7500E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	300.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.197429E+01
2	0.974295E+00
1	0.0
0	0.578373E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.992987E+00	J*	0.750676E-01	0.995821E+00	0.432321E+01
0.992987E+00	J*	-.750676E-01	0.995821E+00	-.432321E+01
-.584012E-02	J*	0.761464E-01	0.763700E-01	0.943861E+02
-.584012E-02	J*	-.761464E-01	0.763700E-01	-.943861E+02

TABLE 8.17

PARAMETERS

SAMPLING PERIOD	T	=	0.7500E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	400.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.198066E+01
2	0.980658E+00
1	0.0
0	0.580251E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.996133E+00	J*	0.754209E-01	0.998984E+00	0.432981E+01
0.996133E+00	J*	-.754209E-01	0.998984E+00	-.432981E+01
-.580365E-02	J*	0.760305E-01	0.762517E-01	0.943655E+02
-.580365E-02	J*	-.760305E-01	0.762517E-01	-.943655E+02

TABLE 8.18

PARAMETERS

SAMPLING PERIOD	T	=	0.7500E-04	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	500.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.198450E+01
2	0.984497E+00
1	0.0
0	0.581382E-02

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.998030E+00	J*	0.755646E-01	0.100089E+01	0.432982E+01
0.998030E+00	J*	-.755646E-01	0.100089E+01	-.432982E+01
-.578177E-02	J*	0.759612E-01	0.761809E-01	0.943530E+02
-.578177E-02	J*	-.759612E-01	0.761809E-01	-.943530E+02

TABLE 8.19

PARAMETERS

SAMPLING PERIOD	T	=	0.1500E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	50.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.173162E+01
2	0.731616E+00
1	0.0
0	0.201288E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.893516E+00	J*	0.853948E-01	0.897588E+00	0.545927E+01
0.893516E+00	J*	-.853948E-01	0.897588E+00	-.545927E+01
-.277085E-01	J*	0.155616E+00	0.158064E+00	0.100096E+03
-.277085E-01	J*	-.155616E+00	0.158064E+00	-.100096E+03

TABLE 8.20

PARAMETERS

SAMPLING PERIOD	T	=	0.1500E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	100.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.185535E+01
2	0.855345E+00
1	0.0
0	0.216982E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.951865E+00	J*	0.133890E+00	0.961236E+00	0.800674E+01
0.951865E+00	J*	-.133890E+00	0.961236E+00	-.800674E+01
-.241924E-01	J*	0.151322E+00	0.153243E+00	0.990833E+02
-.241924E-01	J*	-.151322E+00	0.153243E+00	-.990833E+02

TABLE 8.21

PARAMETERS

SAMPLING PERIOD	T =	0.1500E-03	SECONDS
SHUNT CAPACITOR	C1 =	0.9600E-05	FARADS
SHUNT CAPACITOR	C2 =	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1 =	0.0100	SIEMENS
TRANSCONDUCTANCE	G2 =	0.0100	SIEMENS
SHUNT RESISTOR	R =	200.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.192485E+01
2	0.924849E+00
1	0.0
0	0.225454E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.985030E+00	J*	0.144519E+00	0.995575E+00	0.834664E+01
0.985030E+00	J*	-.144519E+00	0.995575E+00	-.834664E+01
-.226054E-01	J*	0.149115E+00	0.150818E+00	0.986204E+02
-.226054E-01	J*	-.149115E+00	0.150818E+00	-.986204E+02

TABLE 8.22

PARAMETERS

SAMPLING PERIOD	T =	0.1500E-03	SECONDS
SHUNT CAPACITOR	C1 =	0.9600E-05	FARADS
SHUNT CAPACITOR	C2 =	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1 =	0.0100	SIEMENS
TRANSCONDUCTANCE	G2 =	0.0100	SIEMENS
SHUNT RESISTOR	R =	300.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.194925E+01
2	0.949250E+00
1	0.0
0	0.228376E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.996726E+00	J*	0.146282E+00	0.100740E+01	0.834929E+01
0.996726E+00	J*	-.146282E+00	0.100740E+01	-.834929E+01
-.221009E-01	J*	0.148373E+00	0.150010E+00	0.984723E+02
-.221009E-01	J*	-.148373E+00	0.150010E+00	-.984723E+02

TABLE 8.23

PARAMETERS

SAMPLING PERIOD	T	=	0.1500E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	400.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.196169E+01
2	0.961691E+00
1	0.0
0	0.229857E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.100270E+01	J*	0.146808E+00	0.101339E+01	0.832965E+01
0.100270E+01	J*	-.146808E+00	0.101339E+01	-.832965E+01
-.218535E-01	J*	0.148002E+00	0.149607E+00	0.983995E+02
-.218535E-01	J*	-.148002E+00	0.149607E+00	-.983995E+02

TABLE 8.24

PARAMETERS

SAMPLING PERIOD	T	=	0.1500E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	500.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.196923E+01
2	0.969233E+00
1	0.0
0	0.230750E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.100632E+01	J*	0.147008E+00	0.101700E+01	0.831124E+01
0.100632E+01	J*	-.147008E+00	0.101700E+01	-.831124E+01
-.217065E-01	J*	0.147779E+00	0.149365E+00	0.983562E+02
-.217065E-01	J*	-.147779E+00	0.149365E+00	-.983562E+02

TABLE 8.25

PARAMETERS

SAMPLING PERIOD	T	=	0.3000E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	50.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.153526E+01
2	0.535261E+00
1	0.0
0	0.697108E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.865480E+00	J*	0.190201E+00	0.886134E+00	0.123945E+02
0.865480E+00	J*	-.190201E+00	0.886134E+00	-.123945E+02
-.978498E-01	J*	0.281430E+00	0.297955E+00	0.109172E+03
-.978498E-01	J*	-.281430E+00	0.297955E+00	-.109172E+03

TABLE 8.26

PARAMETERS

SAMPLING PERIOD	T	=	0.3000E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	100.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.173162E+01
2	0.731616E+00
1	0.0
0	0.805153E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.948437E+00	J*	0.248604E+00	0.980478E+00	0.146879E+02
0.948437E+00	J*	-.248604E+00	0.980478E+00	-.146879E+02
-.826295E-01	J*	0.277355E+00	0.289402E+00	0.106590E+03
-.826295E-01	J*	-.277355E+00	0.289402E+00	-.106590E+03

TABLE 8.27

PARAMETERS

SAMPLING PERIOD	T	=	0.3000E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	200.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.185535E+01
2	0.855345E+00
1	0.0
0	0.867928E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.100283E+01	J*	0.265516E+00	0.103738E+01	0.148298E+02
0.100283E+01	J*	-.265516E+00	0.103738E+01	-.148298E+02
-.751545E-01	J*	0.273865E+00	0.283990E+00	0.105345E+03
-.751545E-01	J*	-.273865E+00	0.283990E+00	-.105345E+03

TABLE 8.28

PARAMETERS

SAMPLING PERIOD	T	=	0.3000E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	300.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.190108E+01
2	0.901075E+00
1	0.0
0	0.890324E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.102327E+01	J*	0.268583E+00	0.105793E+01	0.147070E+02
0.102327E+01	J*	-.268583E+00	0.105793E+01	-.147070E+02
-.727297E-01	J*	0.272506E+00	0.282044E+00	0.104944E+03
-.727297E-01	J*	-.272506E+00	0.282044E+00	-.104944E+03

TABLE 8.29

PARAMETERS

SAMPLING PERIOD	T	=	0.3000E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	400.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.192485E+01
2	0.924849E+00
1	0.0
0	0.901814E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.103396E+01	J*	0.269519E+00	0.106851E+01	0.146100E+02
0.103396E+01	J*	-.269519E+00	0.106851E+01	-.146100E+02
-.715334E-01	J*	0.271792E+00	0.281048E+00	0.104745E+03
-.715334E-01	J*	-.271792E+00	0.281048E+00	-.104745E+03

TABLE 8.30

PARAMETERS

SAMPLING PERIOD	T	=	0.3000E-03	SECONDS
SHUNT CAPACITOR	C1	=	0.9600E-05	FARADS
SHUNT CAPACITOR	C2	=	0.1000E-04	FARADS
TRANSCONDUCTANCE	G1	=	0.0100	SIEMENS
TRANSCONDUCTANCE	G2	=	0.0100	SIEMENS
SHUNT RESISTOR	R	=	500.0000	OHMS

DENOMINATOR POLYNOMIAL COEFFICIENTS

POWER	COEFFICIENT
4	0.100000E+01
3	-.193941E+01
2	0.939413E+00
1	0.0
0	0.908804E-01

DENOMINATOR POLYNOMIAL ROOTS

REAL		IMAGINARY	MODULUS	ARG(DEG)
0.104053E+01	J*	0.269874E+00	0.107496E+01	0.145400E+02
0.104053E+01	J*	-.269874E+00	0.107496E+01	-.145400E+02
-.708209E-01	J*	0.271354E+00	0.280443E+00	0.104627E+03
-.708209E-01	J*	-.271354E+00	0.280443E+00	-.104627E+03

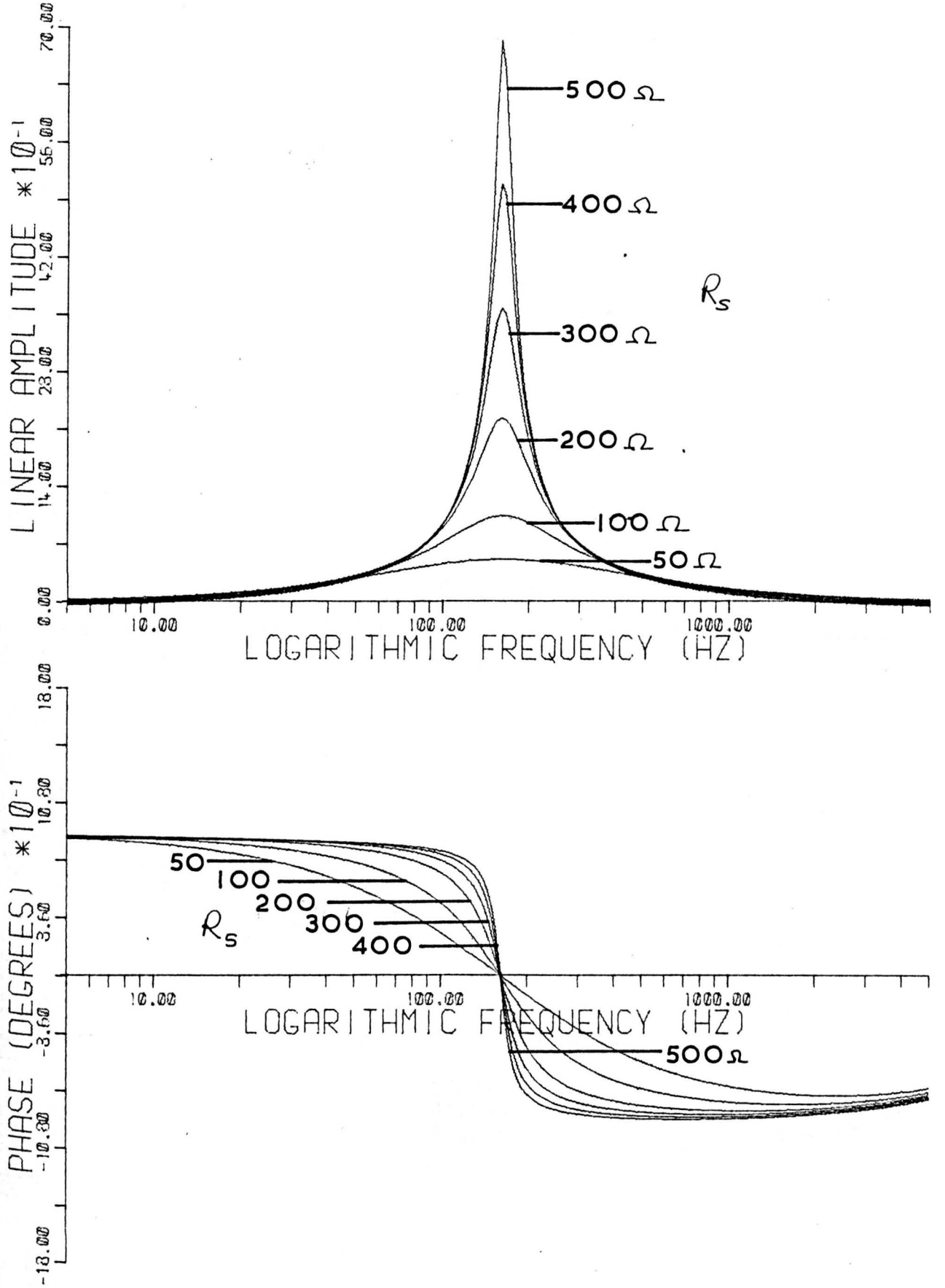
TABLE 8.31

SAMPLING PERIOD (μ s)	SHUNT RESISTANCE (Ω)					
	50	100	200	300	400	500
18.75	0.98123	0.99084	0.99568	0.99730	0.99811	0.99860
37.5	0.96405	0.98293	0.99251	0.99572	0.99733	0.99830
75.0	0.93485	0.97092	0.98953	0.99582	0.99898	1.00089
150.0	0.89759	0.96124	0.99558	1.00740	1.01339	1.01700
300.00	0.88613	0.98048	1.03738	1.05793	1.06851	1.07496

UNSTABLE REGION

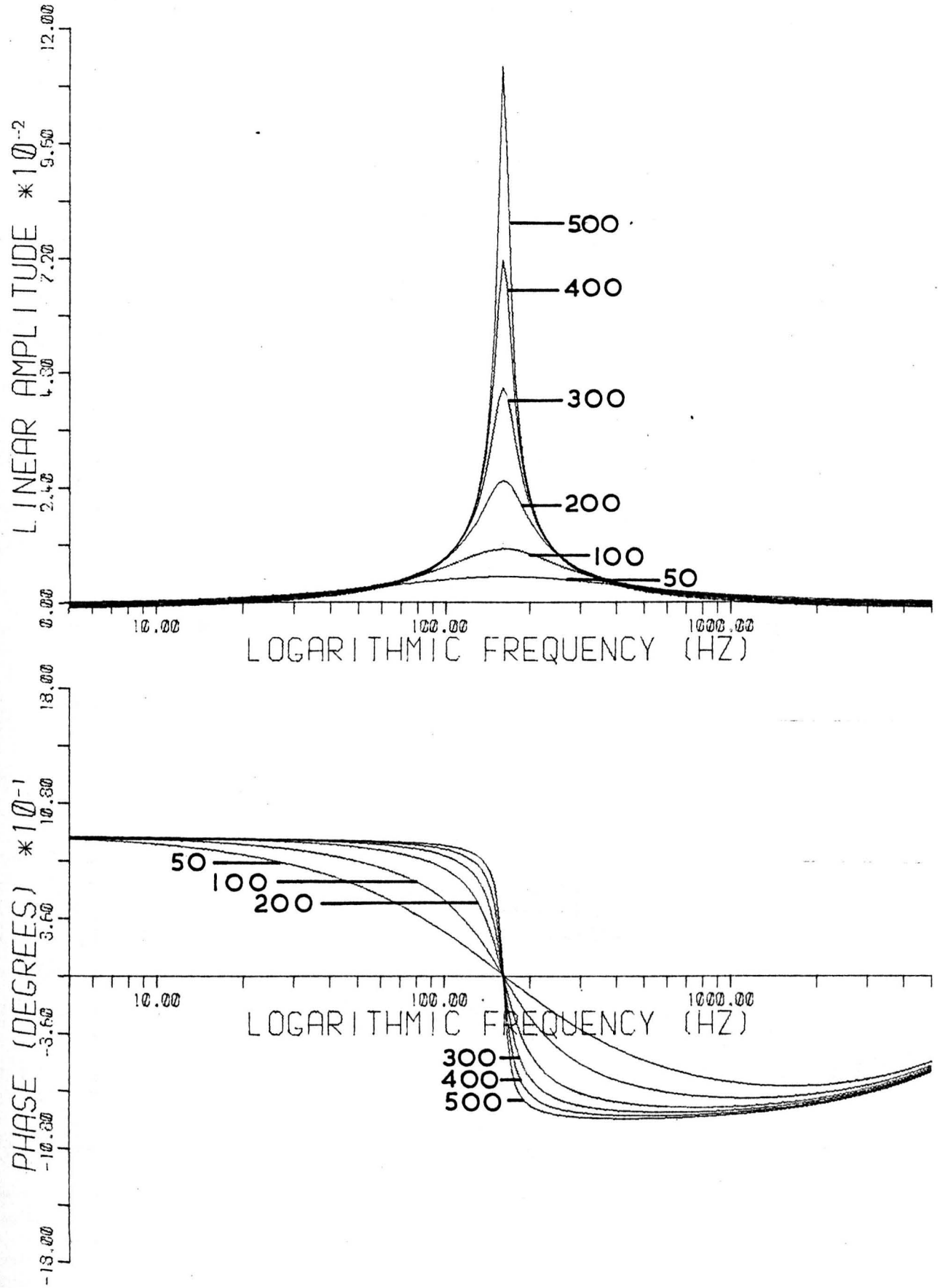
TABLE 8.32

Peak Pole Magnitude for Digital impedance matrix



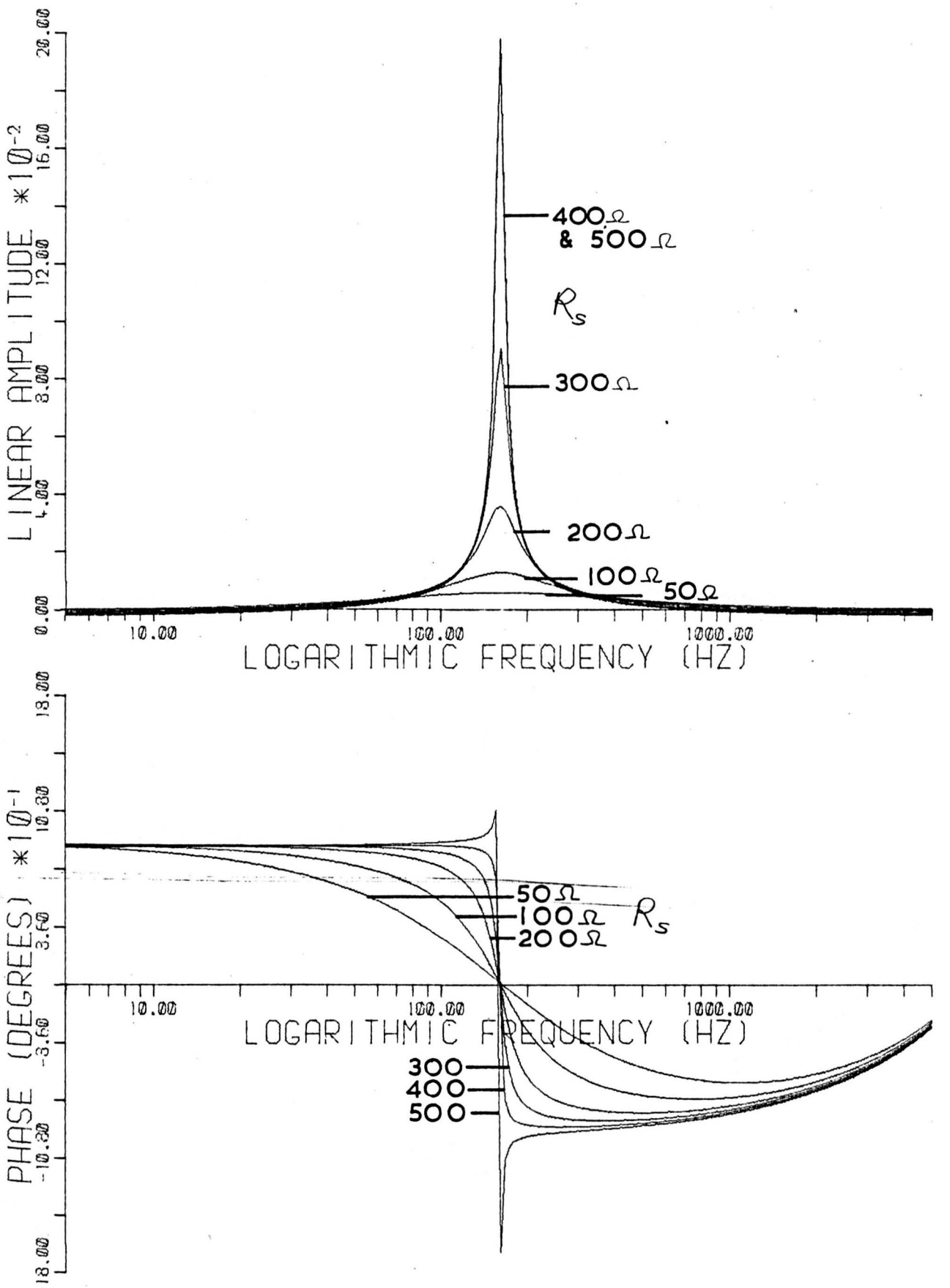
Z11 - SAMPLE PERIOD = 18.75E-6 SECONDS

Computed Input Impedance GRAPH 8.8



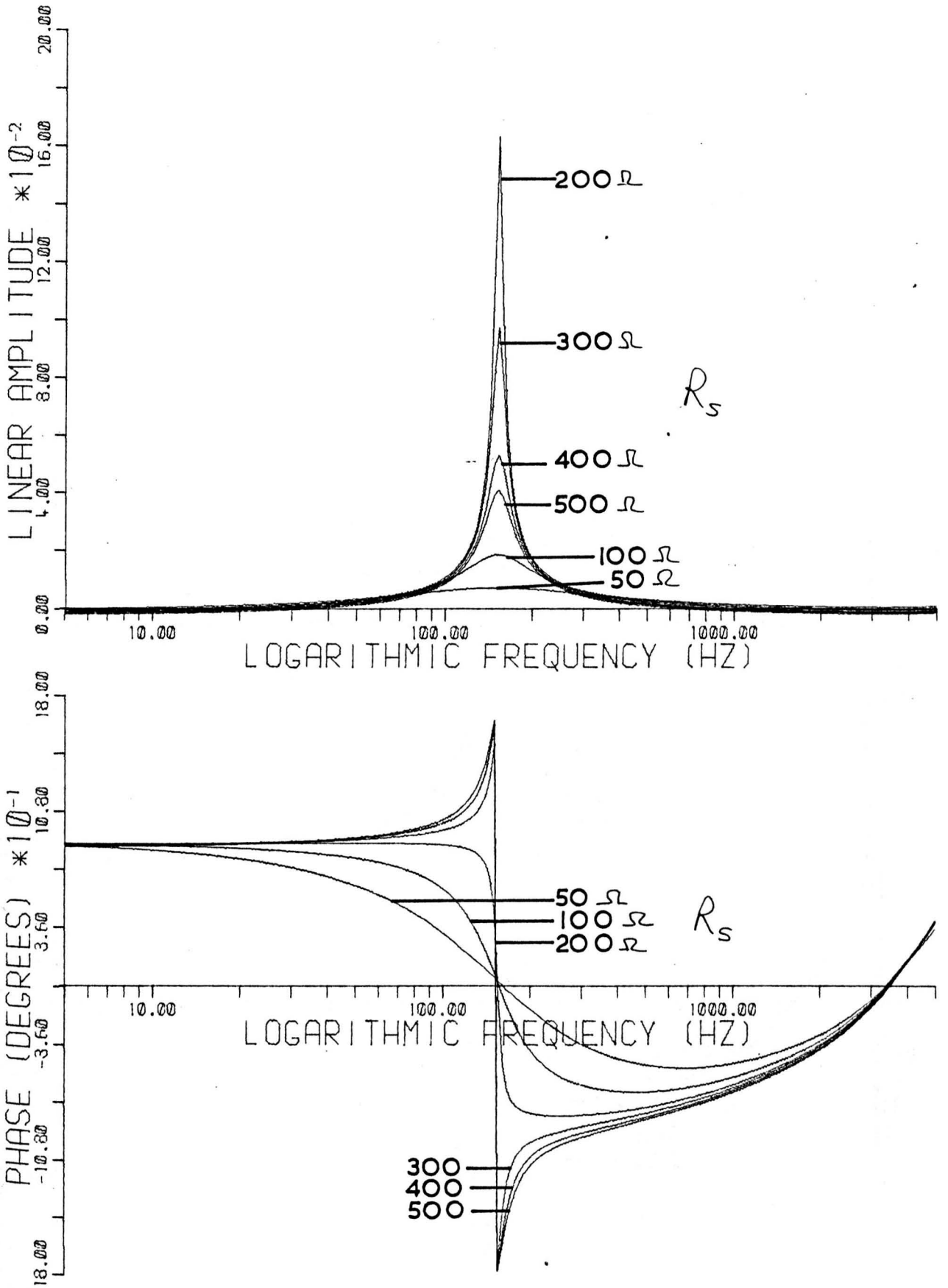
Z11 - SAMPLE PERIOD = 37.5 SECONDS

Computed Input Impedance GRAPH 8.9



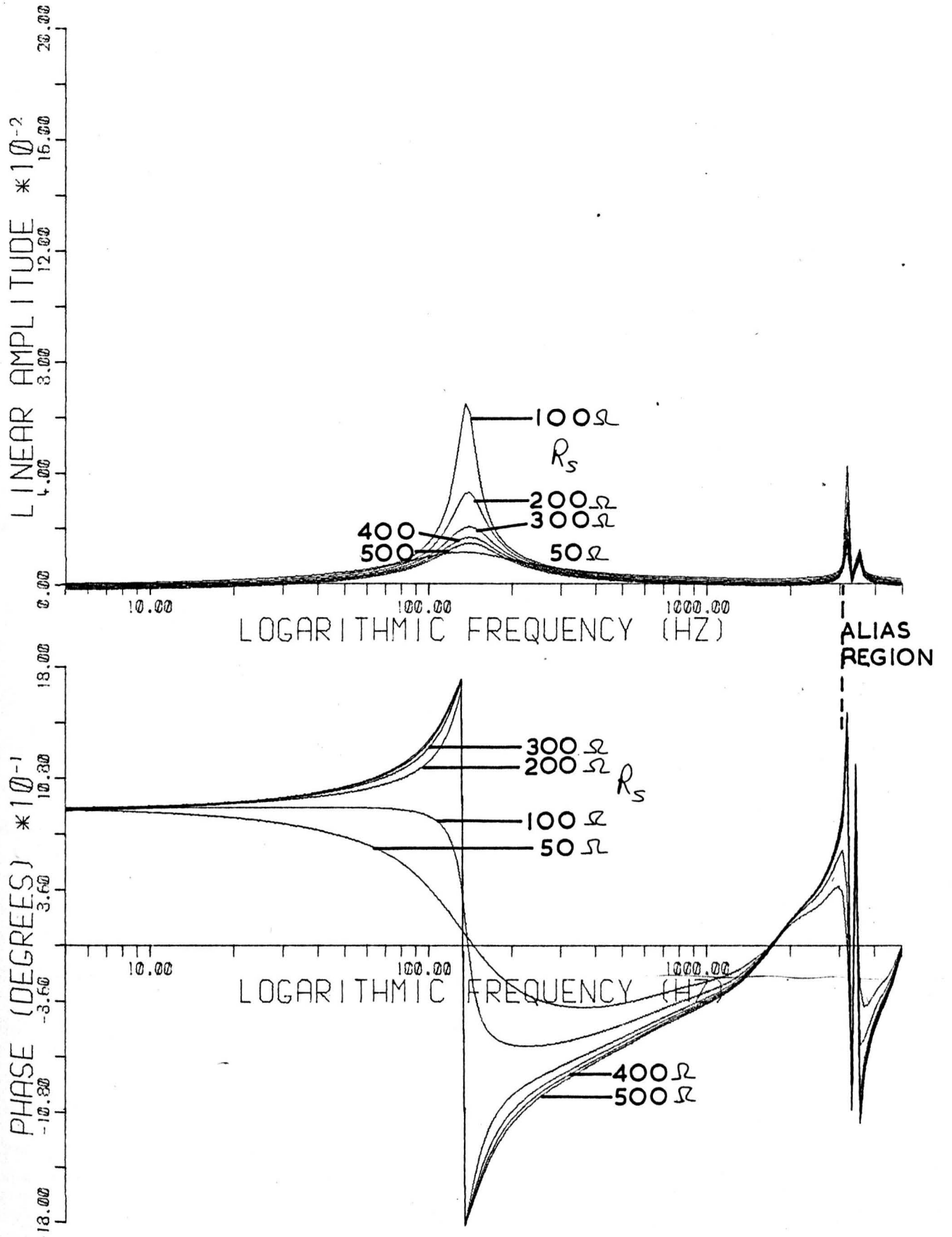
Z11 - SAMPLE PERIOD = 75E-6 SECONDS

Computed Input Impedance GRAPH 8.10



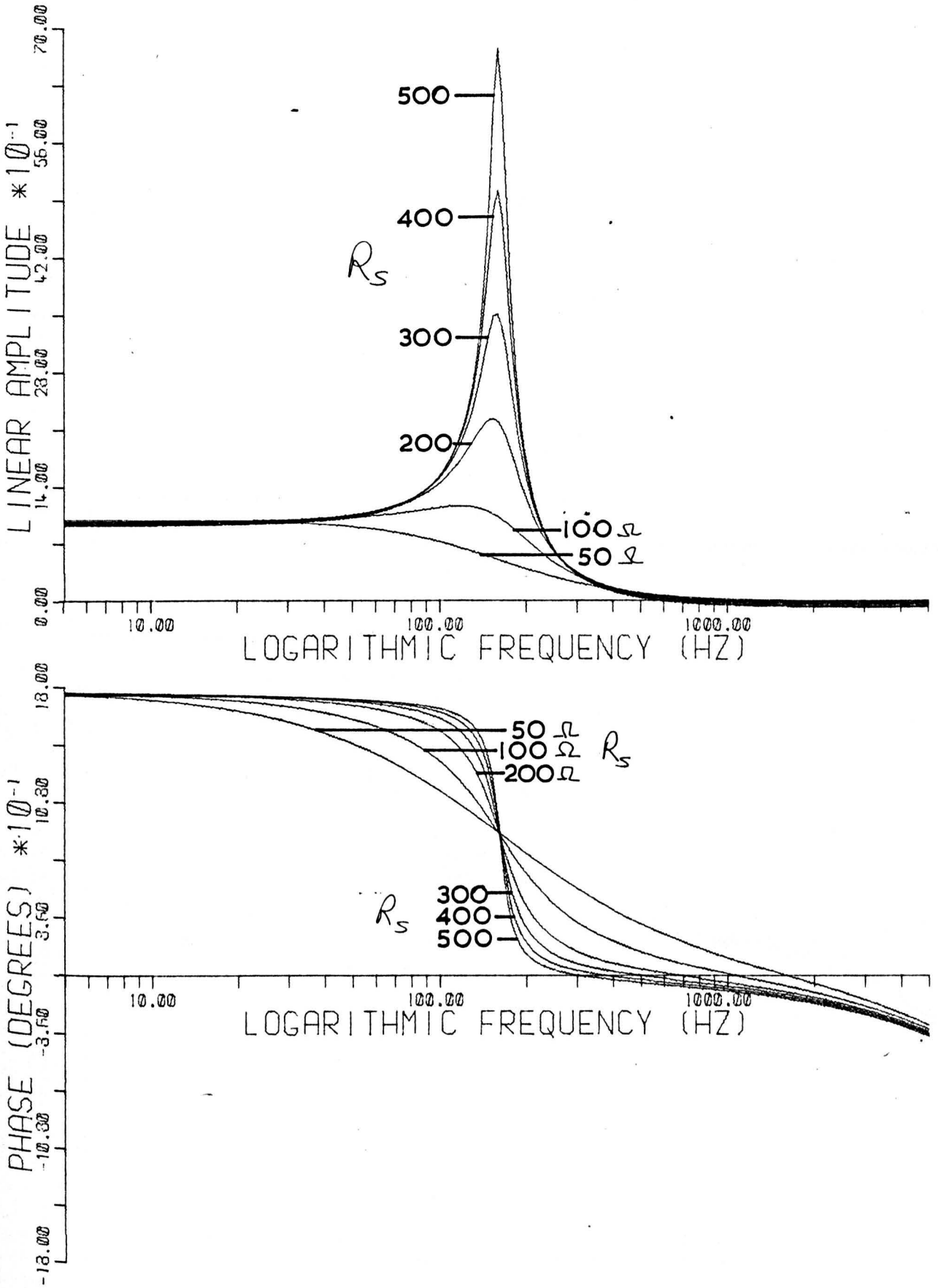
Z11 - SAMPLE PERIOD = 150E-6 SECONDS

Computed Input Impedance GRAPH 8.11



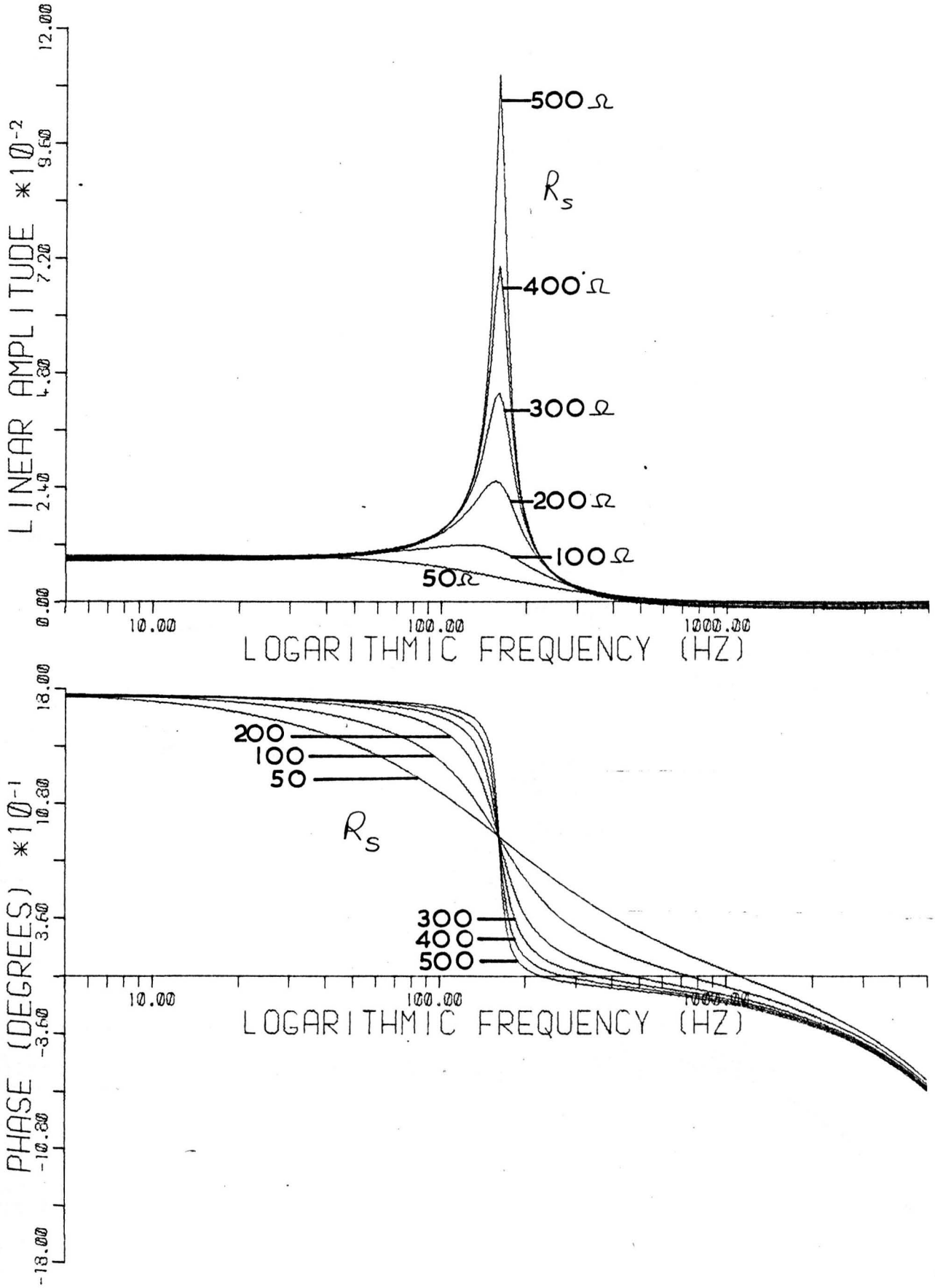
Z11 - SAMPLE PERIOD = 300E-6 SECONDS

Computed Input Impedance GRAPH 8.12



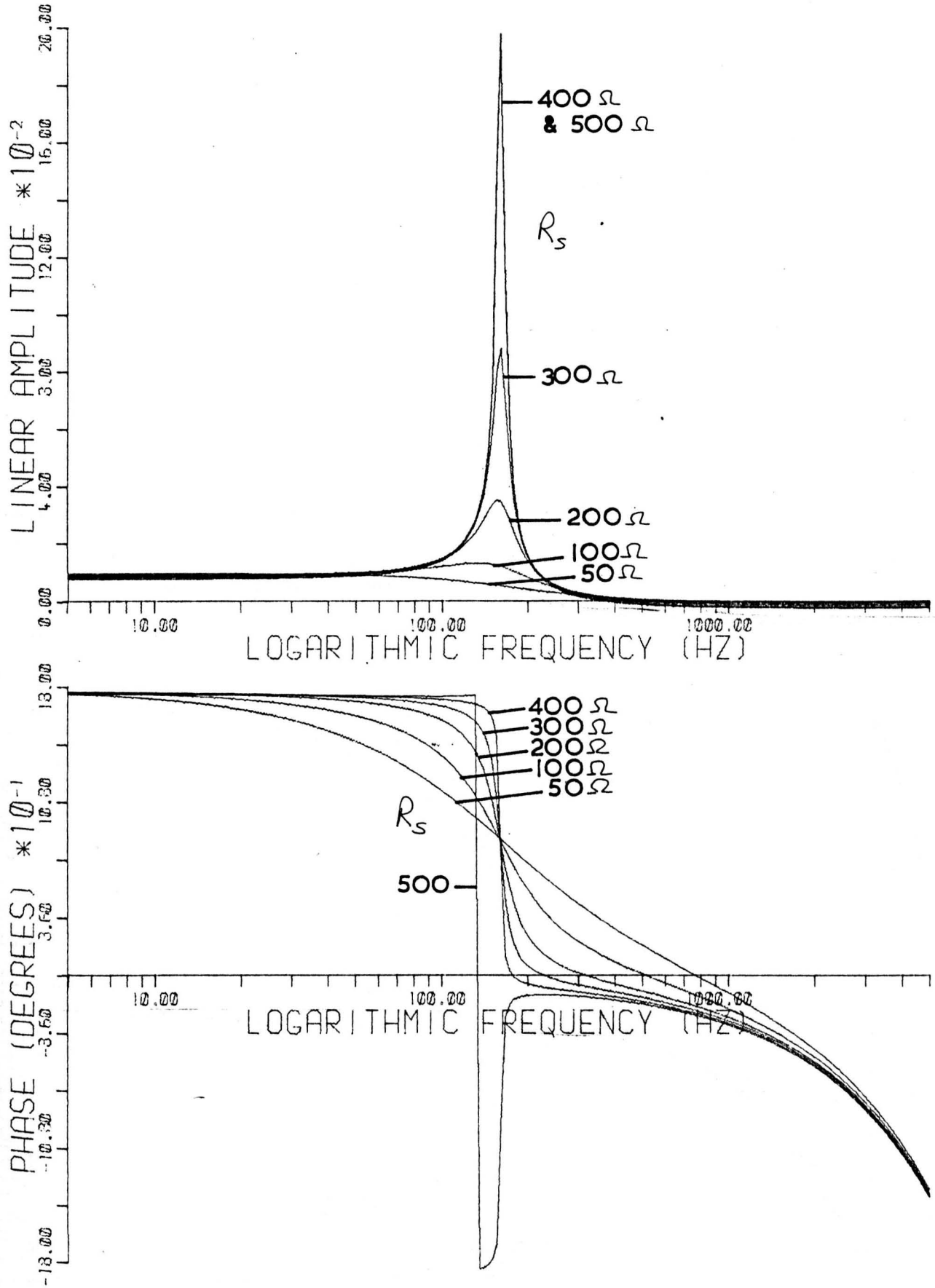
Z12 - SAMPLE PERIOD = 18.75E-6 SECONDS

Computed Reverse Transimpedance GRAPH 8.13



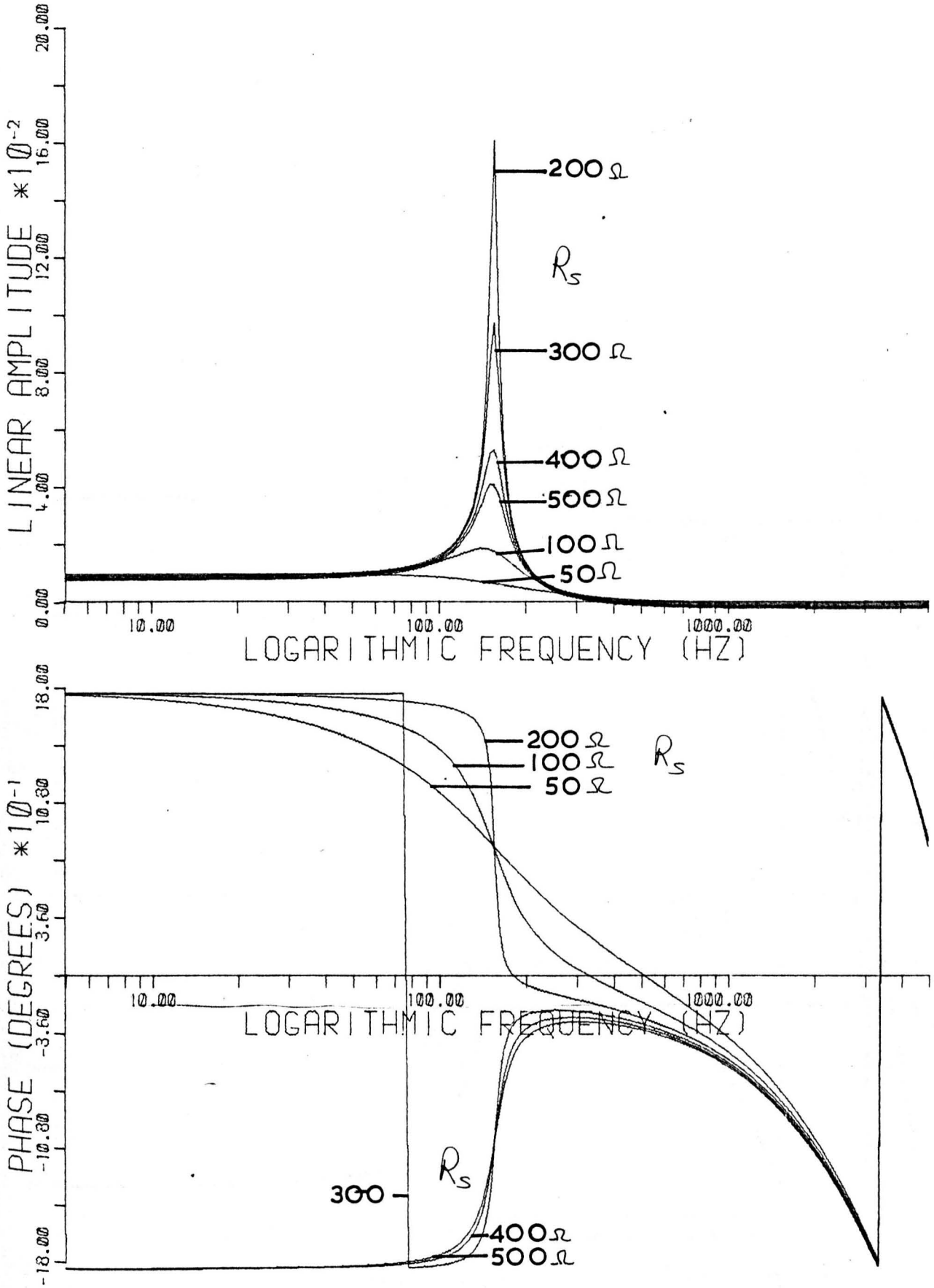
Z12 - SAMPLE PERIOD = 37.5E-6 SECONDS

Computed Reverse Transimpedance GRAPH 8.14



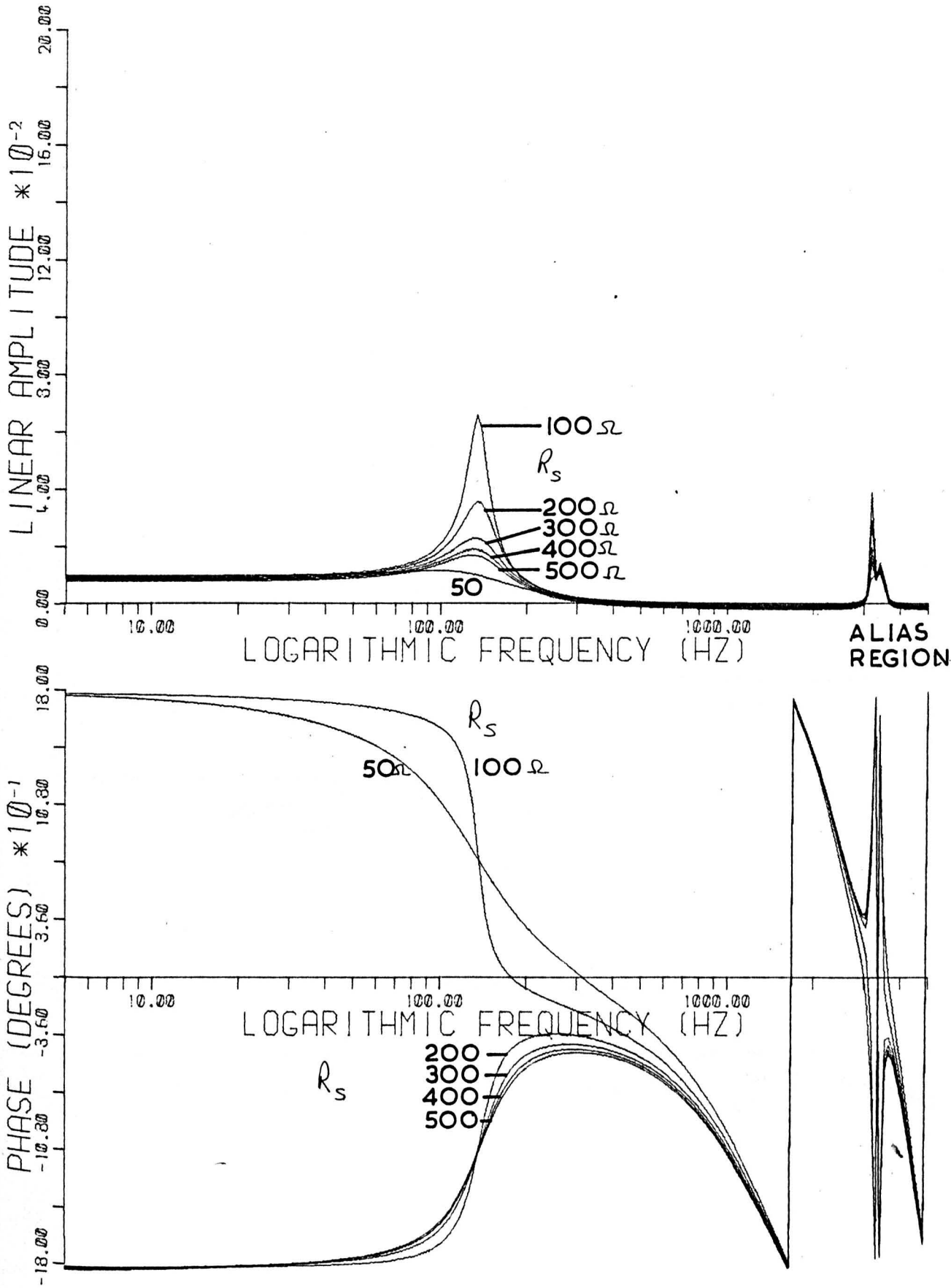
Z12 - SAMPLE PERIOD = 75E-6 SECONDS

Computed Reverse Transimpedance GRAPH 8.15



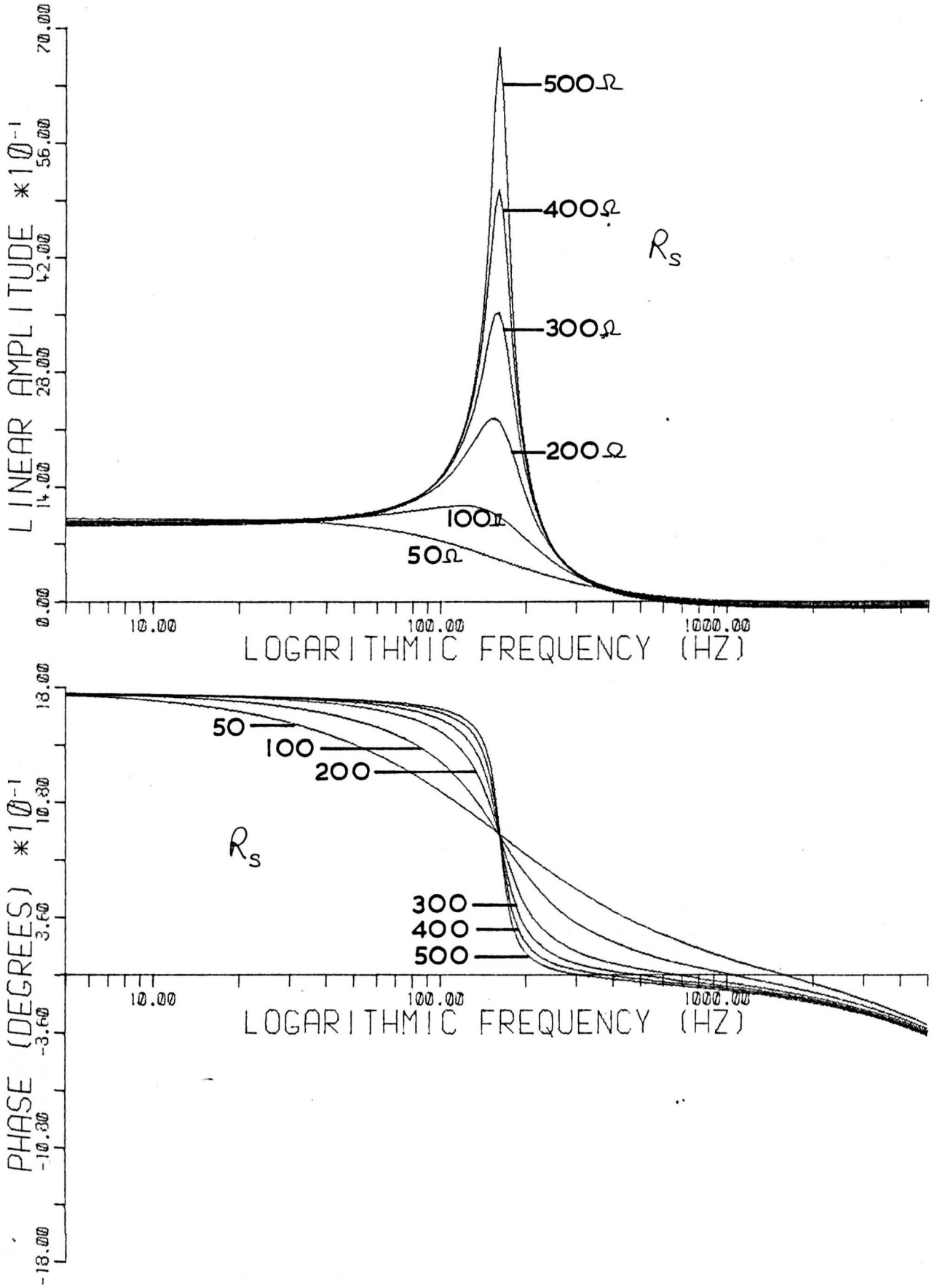
Z12 - SAMPLE PERIOD = 150E-6 SECONDS

Computed Reverse Transimpedance GRAPH 8.16



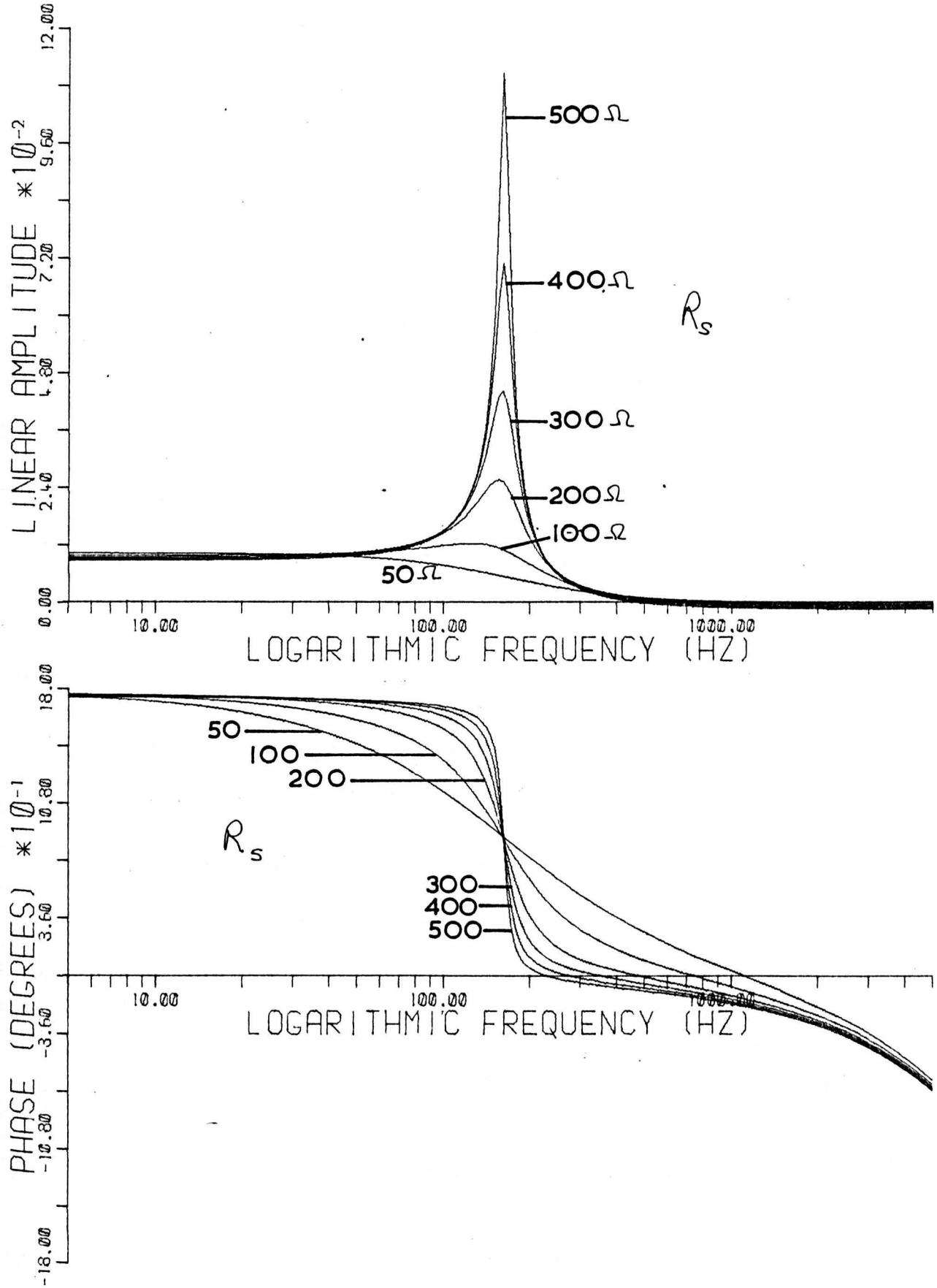
Z12 - SAMPLE PERIOD = 300E-6 SECONDS

Computed Reverse Transimpedance GRAPH 8.17



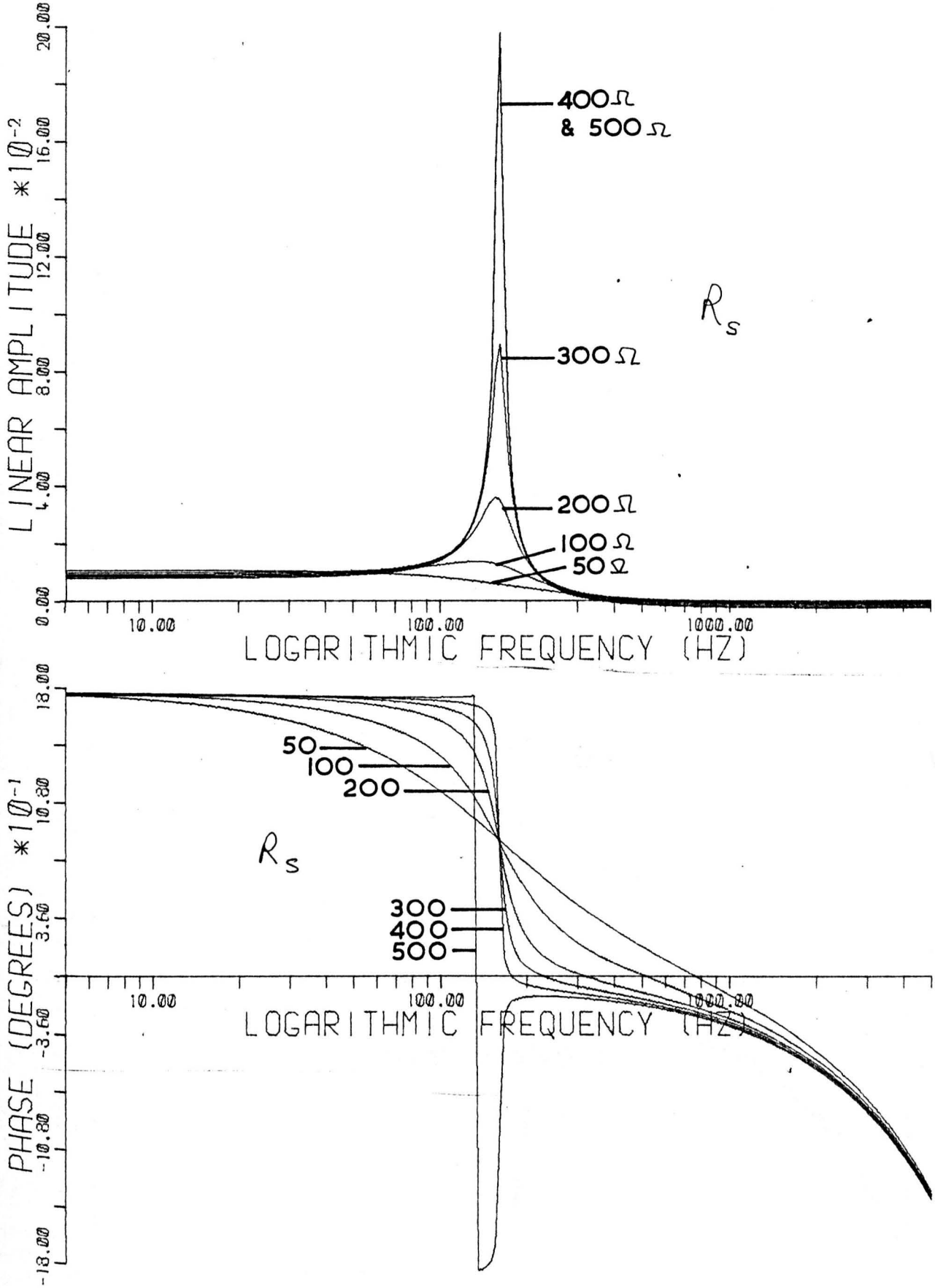
Z21 - SAMPLE PERIOD = 18.75E-6 SECONDS

Computed Forward Transimpedance GRAPH 8.18



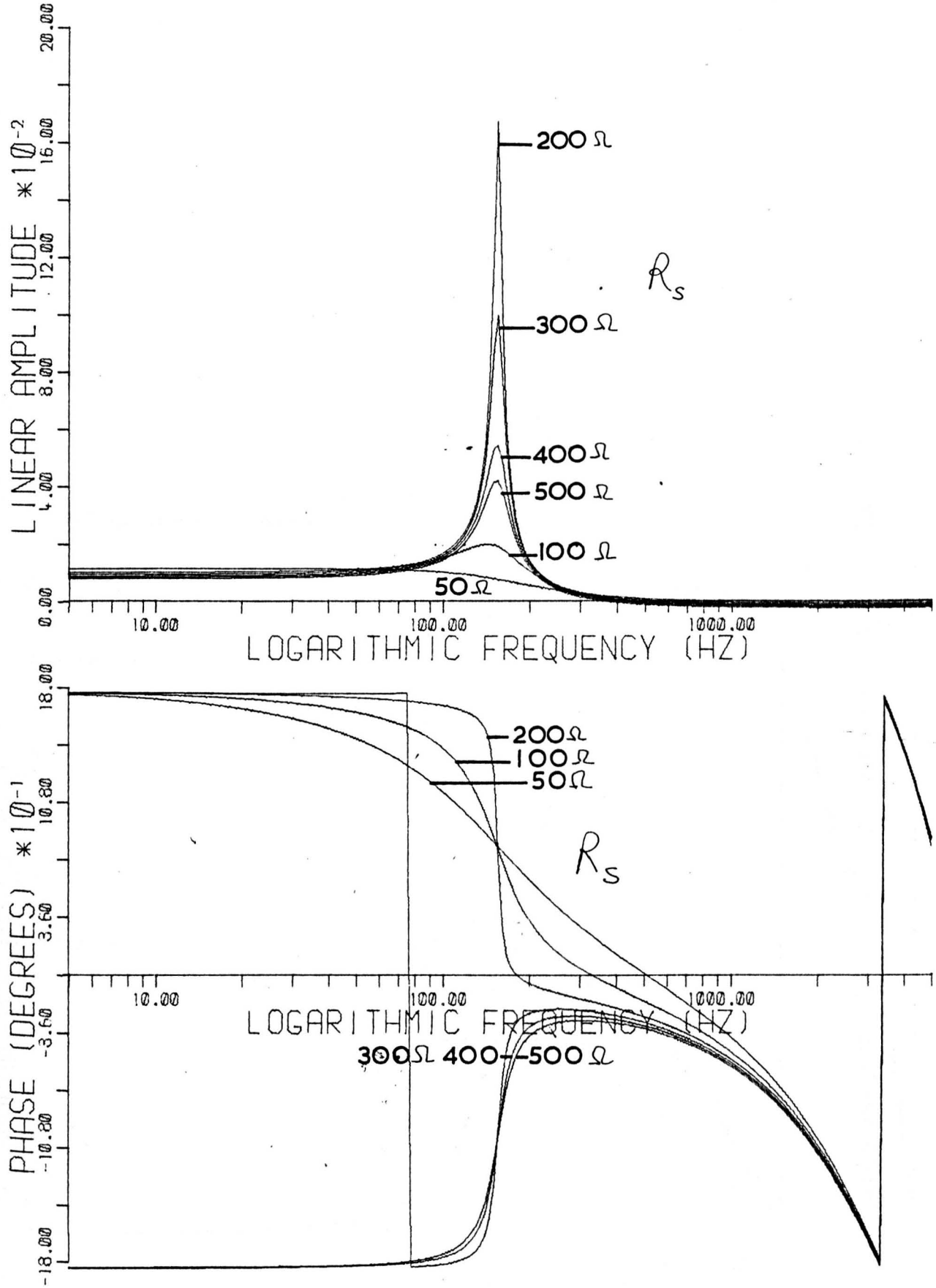
Z21 - SAMPLE PERIOD = 37.5E-6 SECONDS

Computed Forward Transimpedance GRAPH 8.19



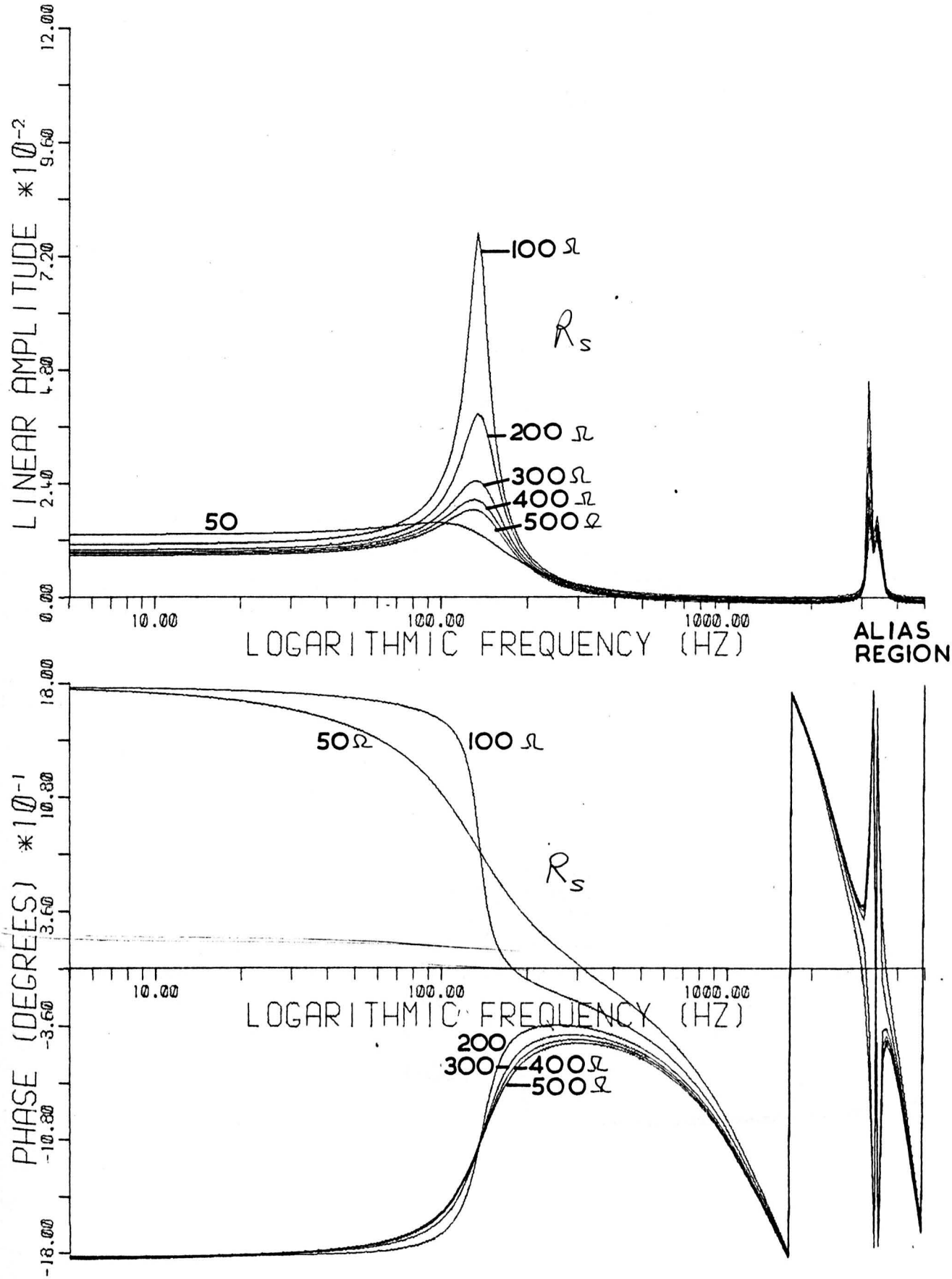
Z21 - SAMPLE PERIOD = 75E-6 SECONDS

Computed Forward Transimpedance GRAPH 8.20



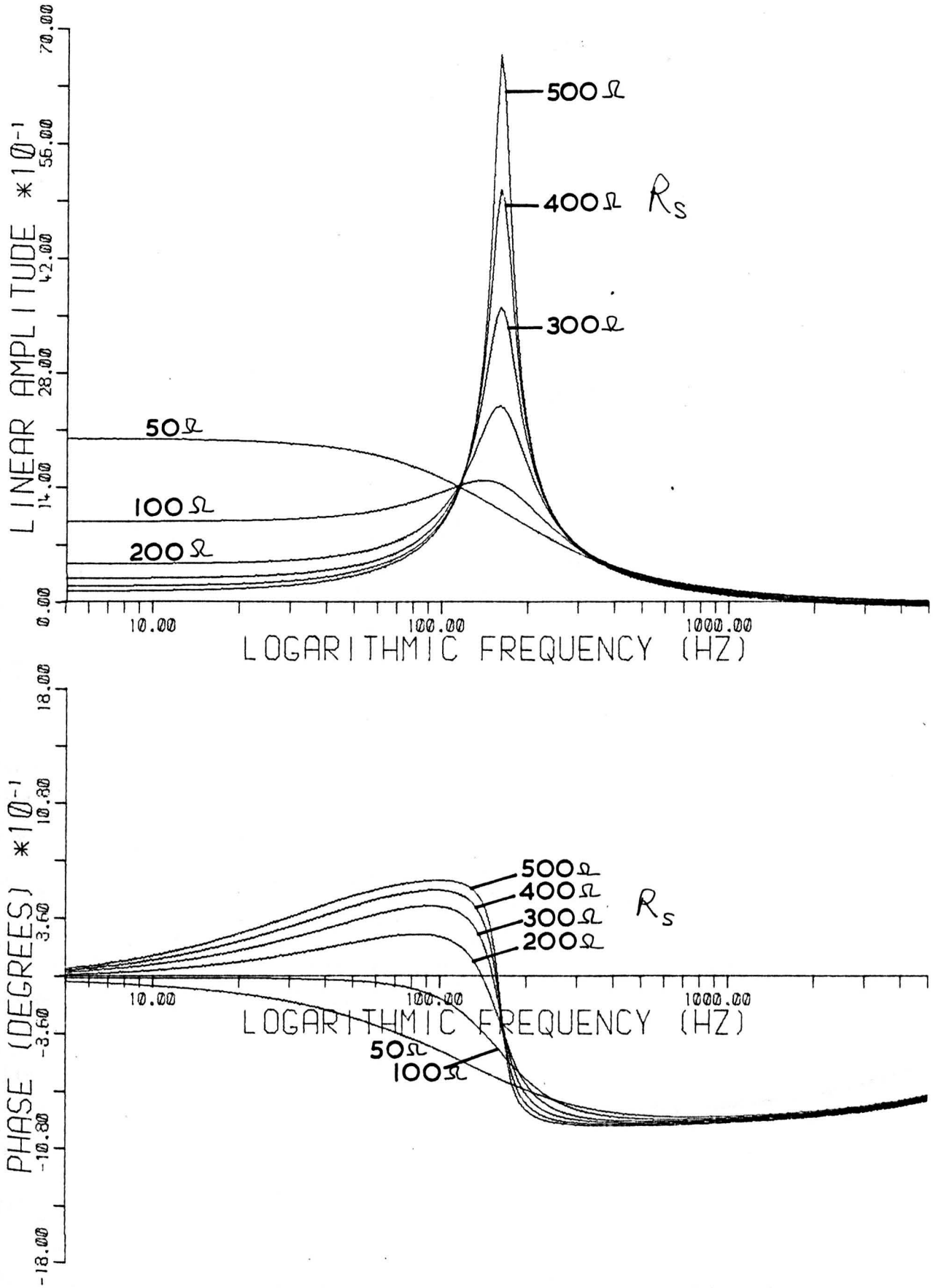
Z21 - SAMPLE PERIOD = 150E-6 SECONDS

Computed Forward Transimpedance GRAPH 8.21



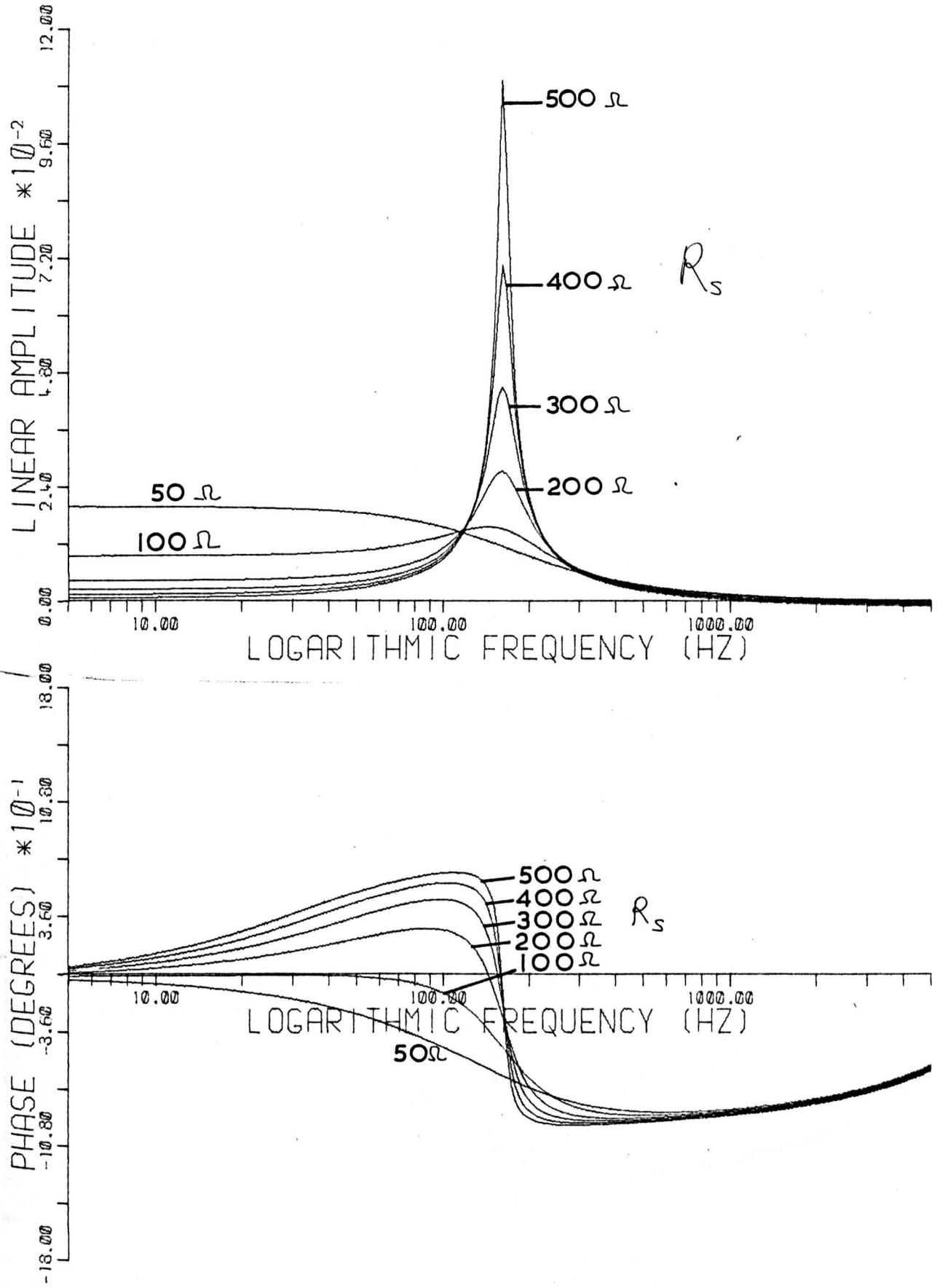
Z21 - SAMPLE PERIOD = 300E-6 SECONDS

Computed Forward Transimpedance GRAPH 8.22



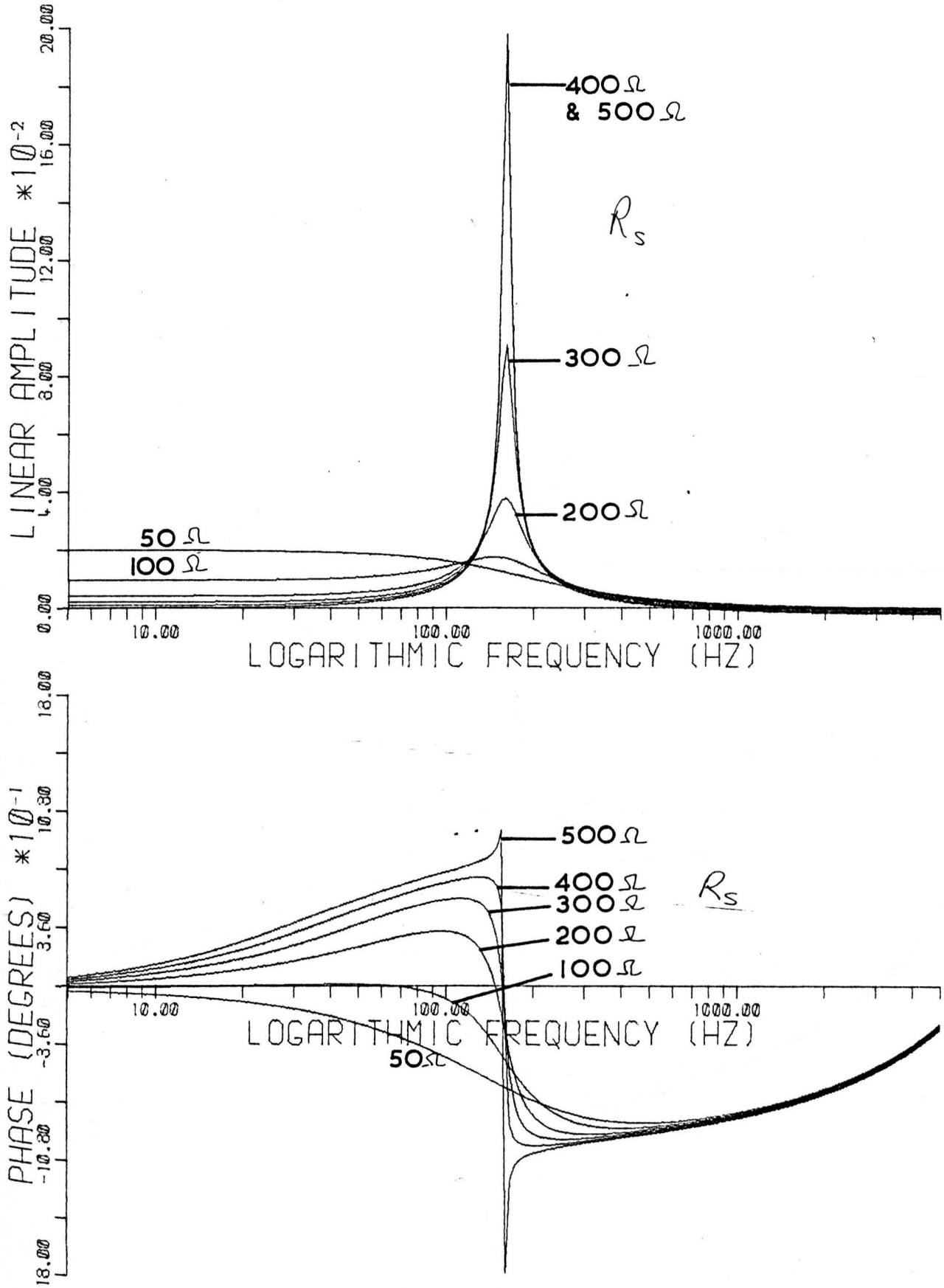
Z22 - SAMPLE PERIOD = 18.75E-6 SECONDS

Computed Output Impedance GRAPH 8.23



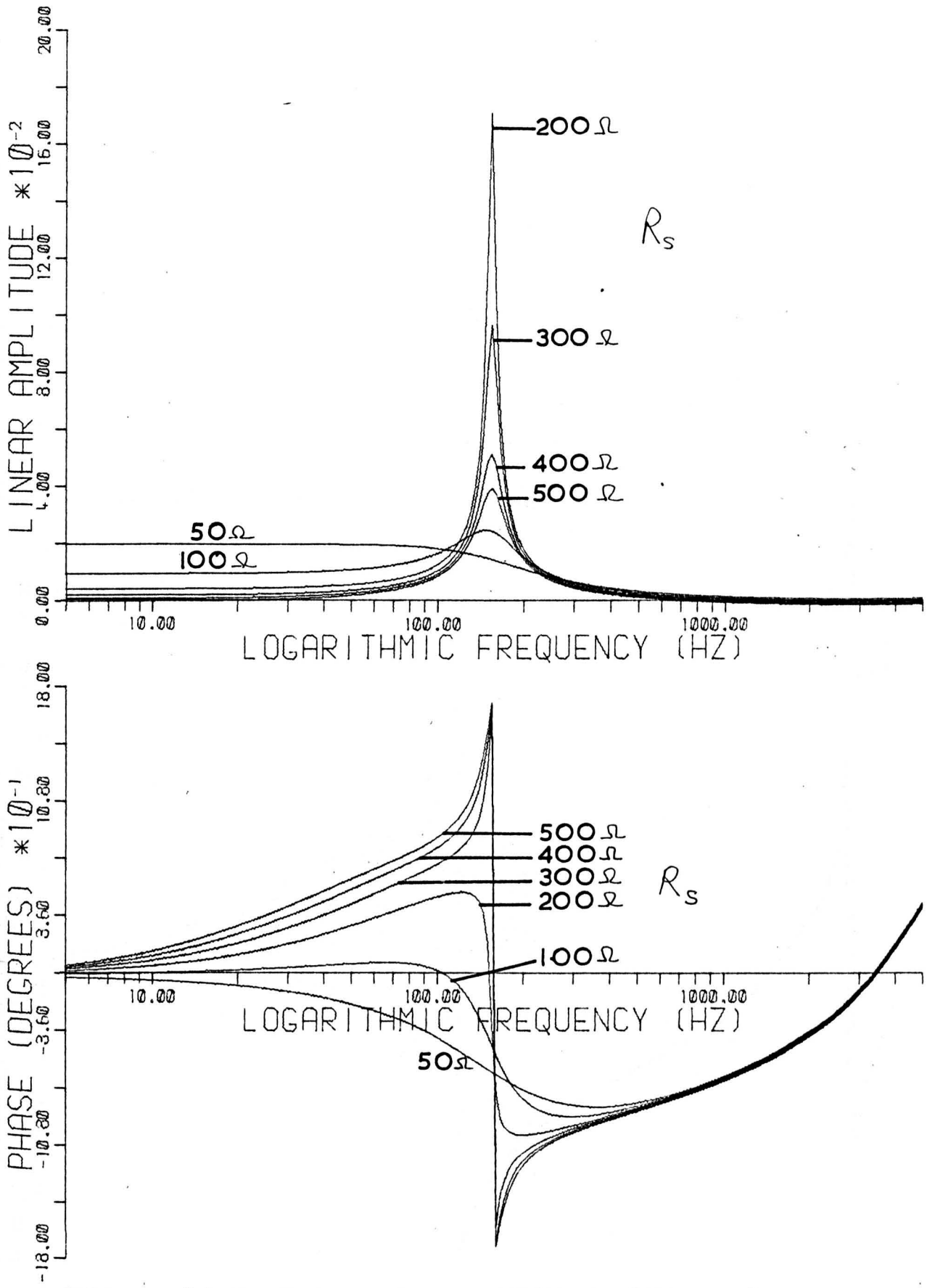
Z22 - SAMPLE PERIOD = 37.5E-6 SECONDS

Computed Output Impedance GRAPH 8.24



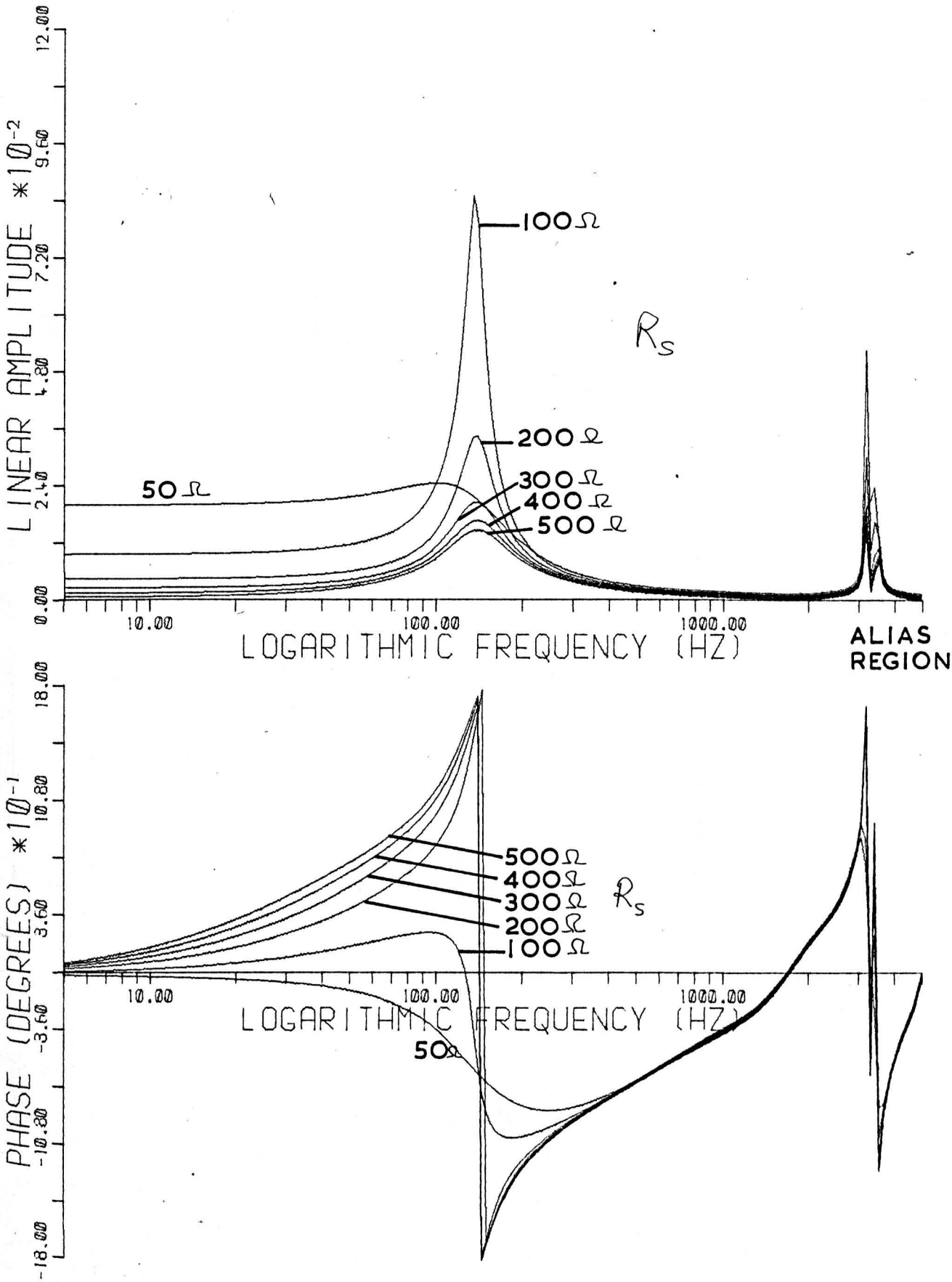
Z22 - SAMPLE PERIOD = 75E-6 SECONDS

Computed Output Impedance GRAPH 8.25



Z22 - SAMPLE PERIOD = 150E-6 SECONDS

Computed Output Impedance GRAPH 8.26



Z22 - SAMPLE PERIOD = 300E-6 SECONDS

Computed Output Impedance GRAPH 8.27

VALUES	SAMPLING PERIOD (μs)	SHUNT RESISTANCE (Ω)					
		50	100	200	300	400	500
Peak	18.75	162.11	162.11	162.11	162.11	162.11	162.11
Frequency	37.5	162.11	162.11	162.11	162.11	162.11	162.11
(Hz)	75.0	159.74	160.53	160.53	160.53	160.53	160.53
(No Peak)	150.0	-	-	-	-	-	-
	300.0	-	-	-	-	-	-
				UNSTABLE REGION			
Peak	18.75	52.46	107.00	226.46	362.08	517.26	697.07
Impedance	37.5	55.08	114.89	260.51	455.40	729.97	1144.60
	75.0	60.84	133.94	368.73	920.98	3772.65	4318.11
(No Peak)	150.0	-	-	-	-	-	-
	300.0	-	-	-	-	-	-
				UNSTABLE REGION			

TABLE 8.33

Peak Frequencies and Impedances for z_{11}

VALUES	SAMPLING PERIOD (μs)	SHUNT RESISTANCE (Ω)					
		50	100	200	300	400	500
Peak	18.75	-	-	153.2	158.95	160.53	161.32
Frequency	37.5	-	-	155.0	159.74	161.32	161.32
(Hz)	75.0	-	-	157.37	159.74	160.53	160.53
(No Peak)	150.0	-	-	-	-	-	-
	300.0	-	-	-	-	-	-
					UNSTABLE REGION		
Peak	18.75	-	-	227.97	358.64	509.79	685.13
Impedance	37.5	-	-	261.18	450.60	719.00	1125.76
	75.0	-	-	369.71	915.91	3741.32	4282.25
(No Peak)	150.0	-	-	-	-	-	-
	300.0	-	-	-	-	-	-
					UNSTABLE REGION		

TABLE 8.34

Peak Frequencies and Impedances for z_{21}

that the resonant frequency is virtually independent of the sample rate, being principally defined by the external component values as listed in Table 8.1.

By inspection of the digital admittance matrix (8.1) it can be seen that all the polynomials in z are first order and hence none of these elements will demonstrate any form of resonance. However, the results for $y_{11}(z)$ are shown in Graphs 8.28 to 8.32 to demonstrate the form these frequency responses take.

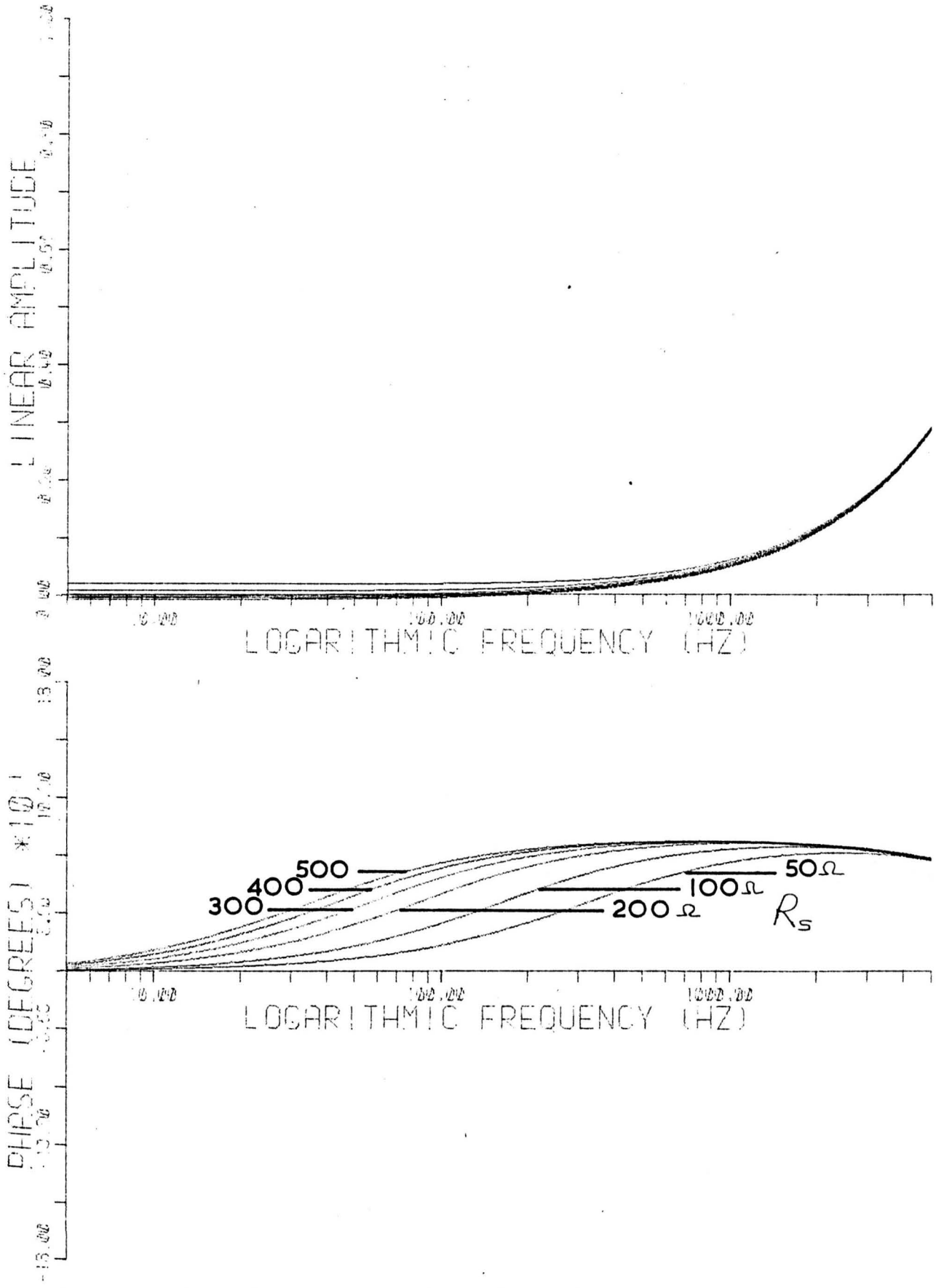
The quantisation noise voltage transfer function for Ports 1 and 2 were evaluated using equation (4.35) and are shown in Graphs 8.33 to 8.42 for all the component values listed in Table 8.1. Tables 8.35 and 8.36 show the peak frequency, gain and noise voltage for the quantisation transfer function at both Ports 1 and 2 respectively. From equation (3.77), any combination of shunt resistance and sampling period which gives a peak amplitude which is greater than or equal to 2 will mean that the 2-port capacitively loaded digital gyrator will display finite limit cycle oscillations with a frequency equal to the peak frequency as listed. From Tables 8.35 and 8.36 this will occur when the shunt resistance is 200 ohms or greater.

8.5 INVERSE Z-TRANSFORM PROGRAM (IZT1)

IZT1 was written to calculate the inverse Z-transform and impulse response for any Z-plane polynomial, but in particular the polynomials evaluated by GDP1, and then to list and plot these results. For the sake of brevity, only the graphical plots are presented here.

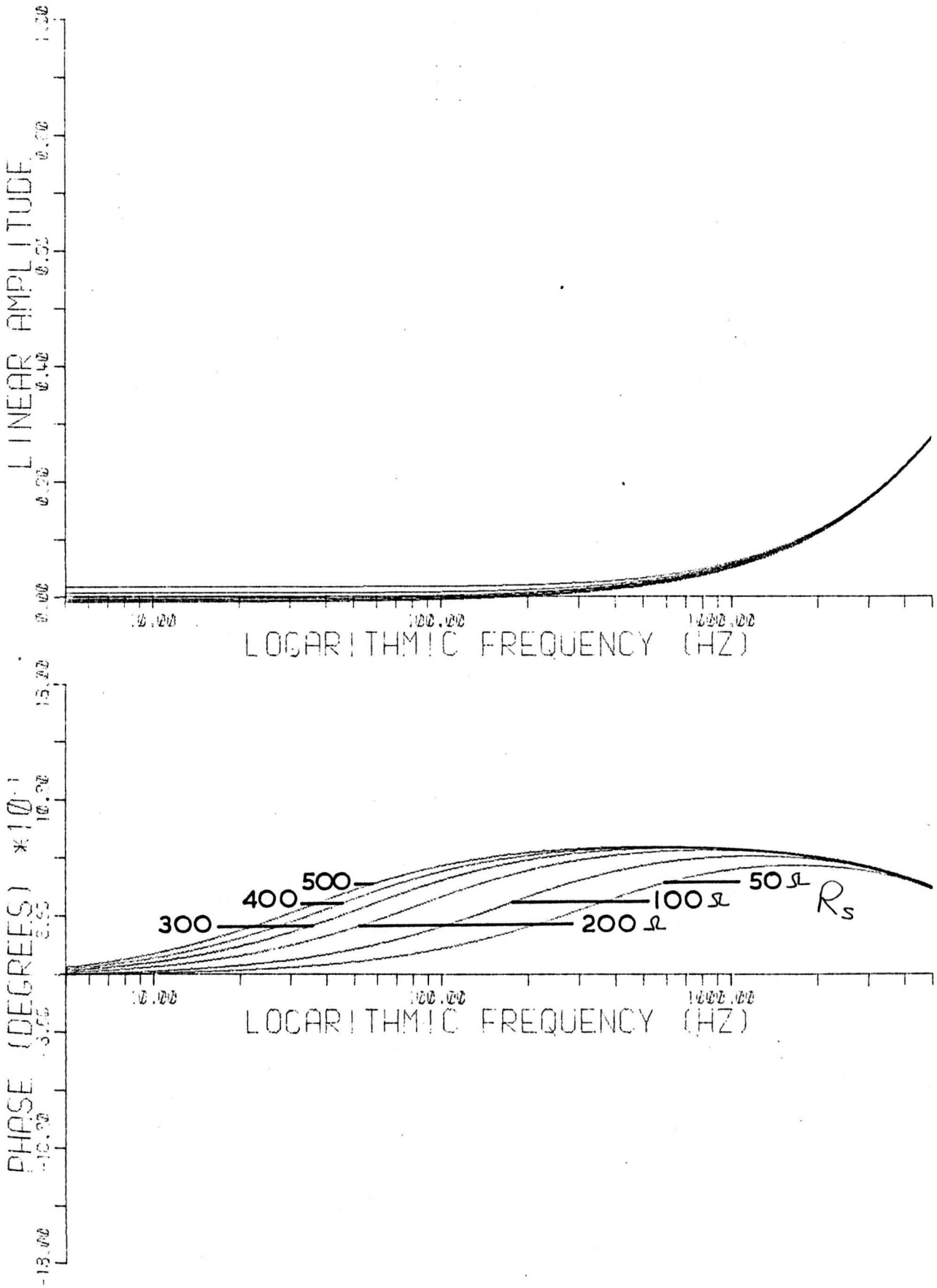
The inverse Z-transforms and impulse responses for the digital input impedance (z_{11}) are shown in Graphs 8.43 to 8.47 and in Graphs 8.48 to 8.52 for the digital forward transfer impedance (z_{21}).

Each inverse transform response has a horizontal axis marked in the number of iterations from an arbitrary initial iteration and



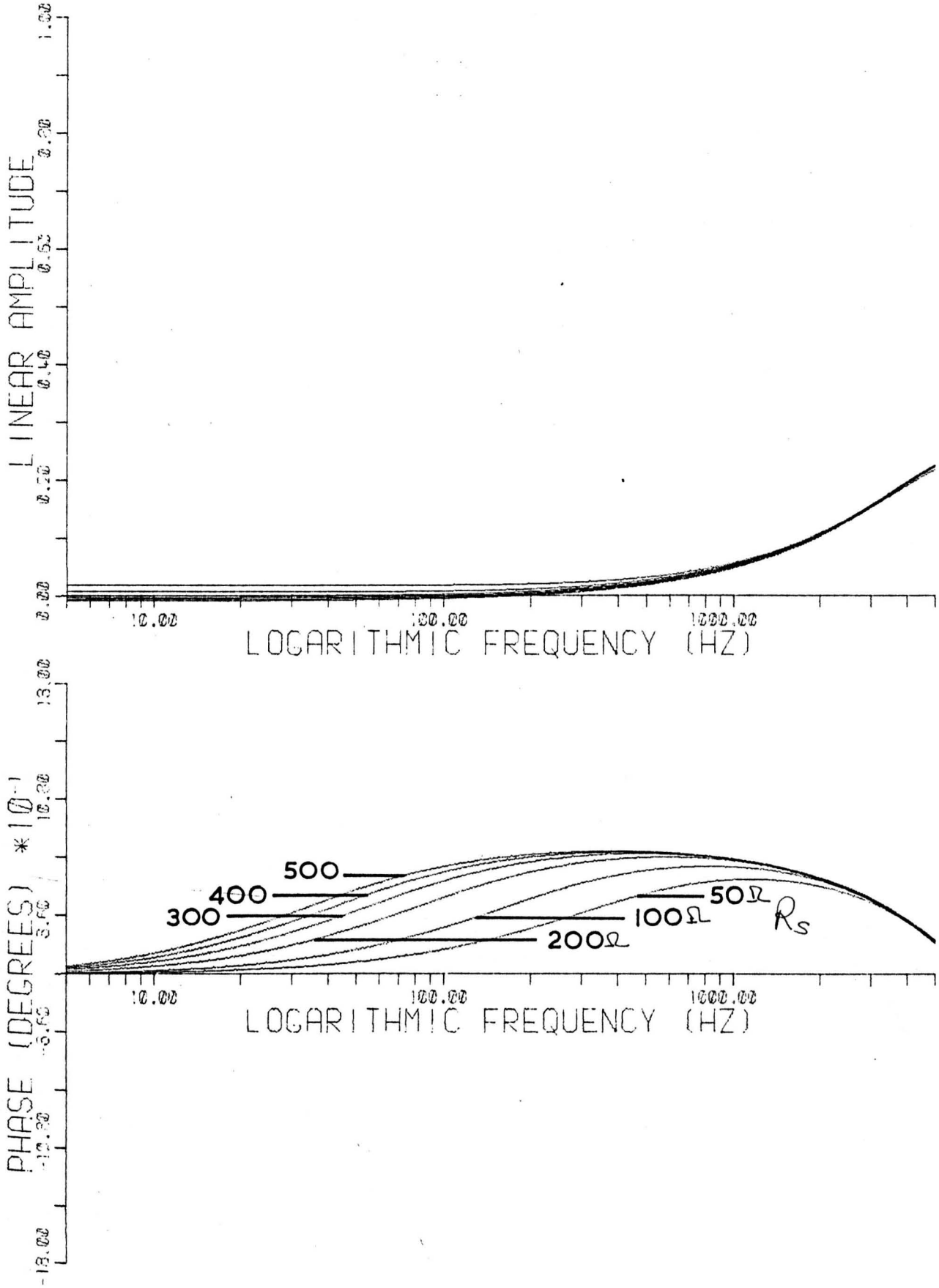
Y11 - SAMPLE PERIOD = 18.75E-6 SECONDS

Computed Input Admittance GRAPH 8.28



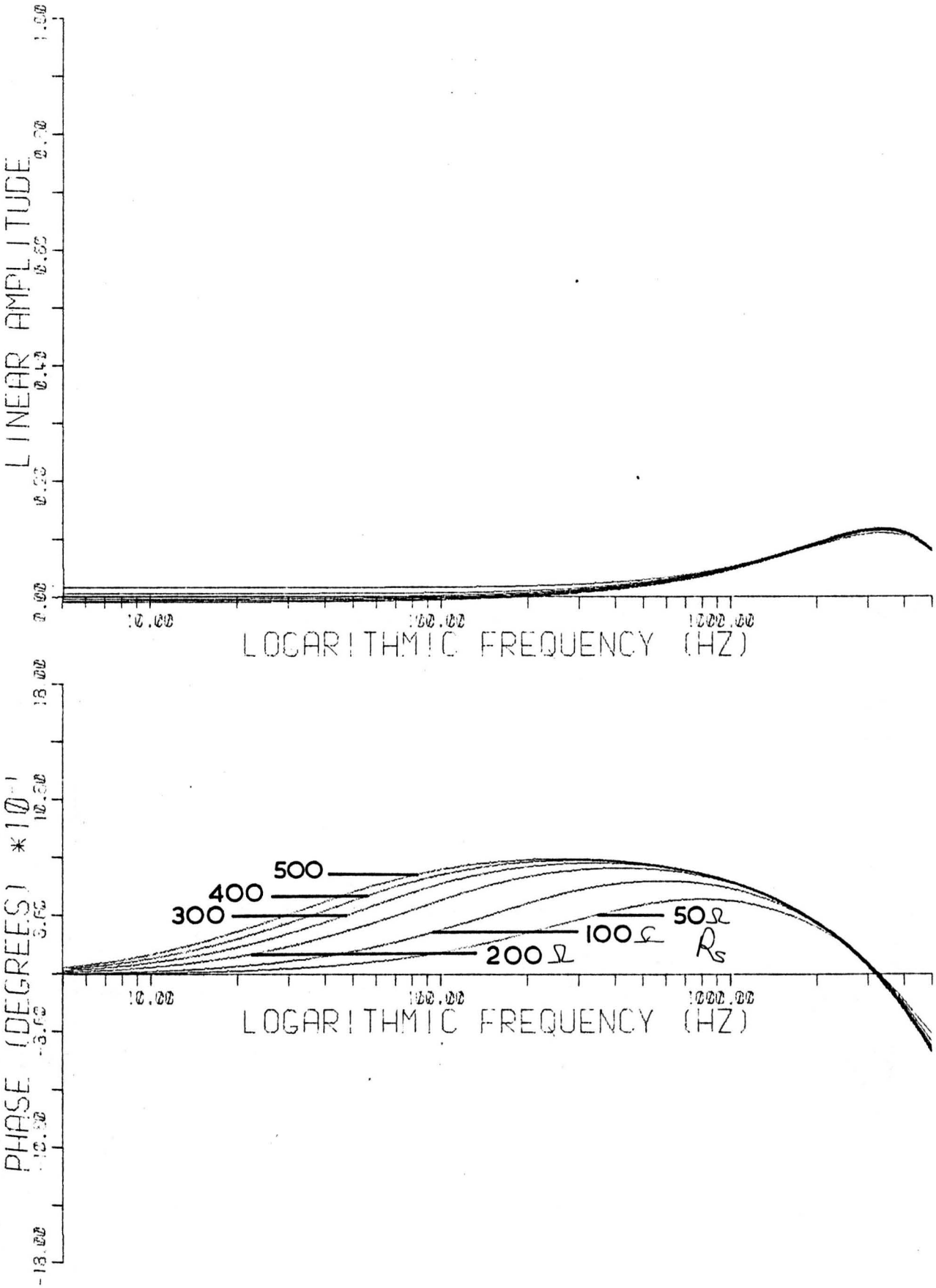
Y11 - SAMPLE PERIOD = 37.5E-6 SECONDS

Computed Input Admittance GRAPH 8.29



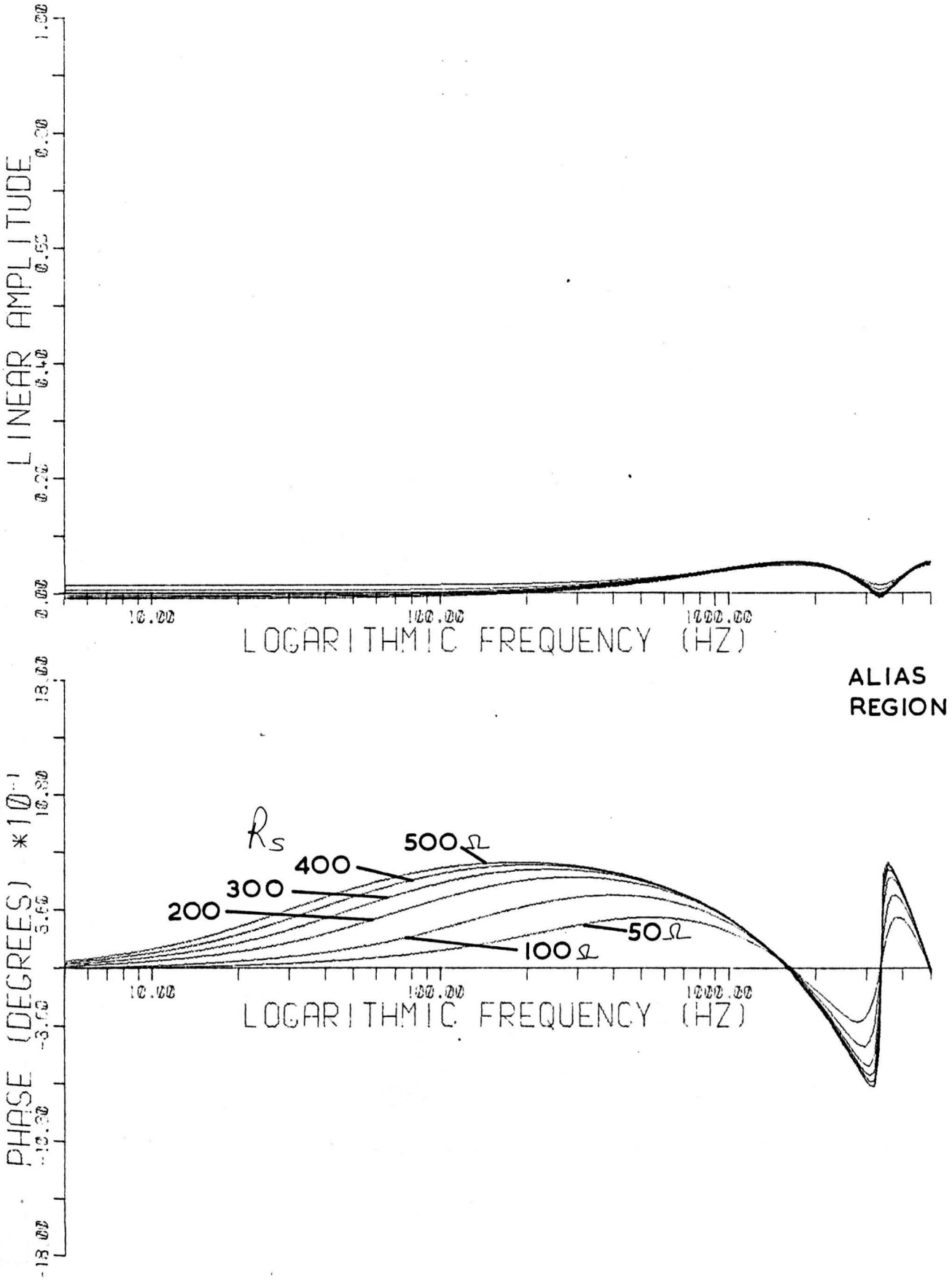
Y11 - SAMPLE PERIOD = 75E-6 SECONDS

Computed Input Admittance GRAPH 8.30



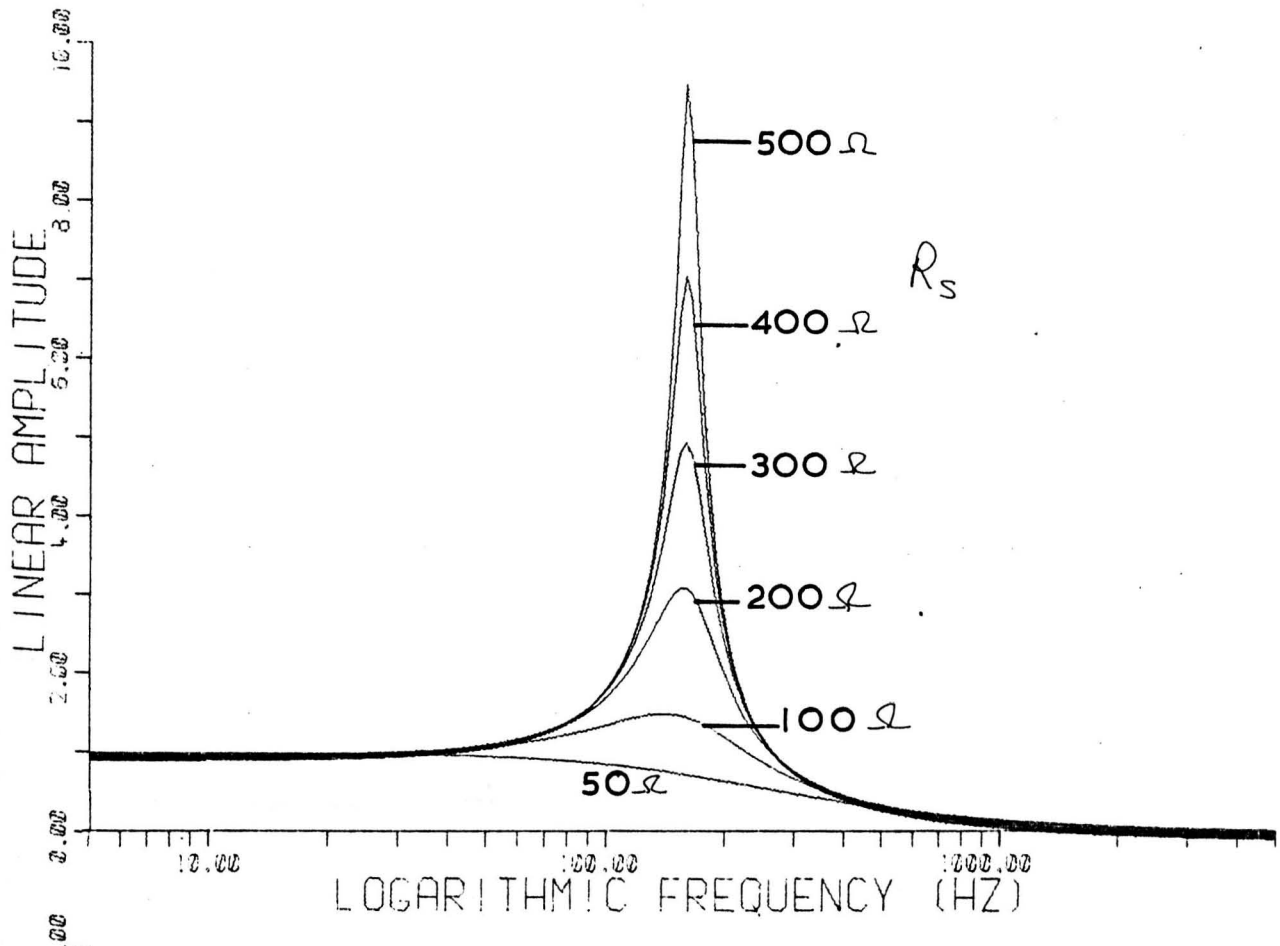
Y11 - SAMPLE PERIOD = 150E-6 SECONDS

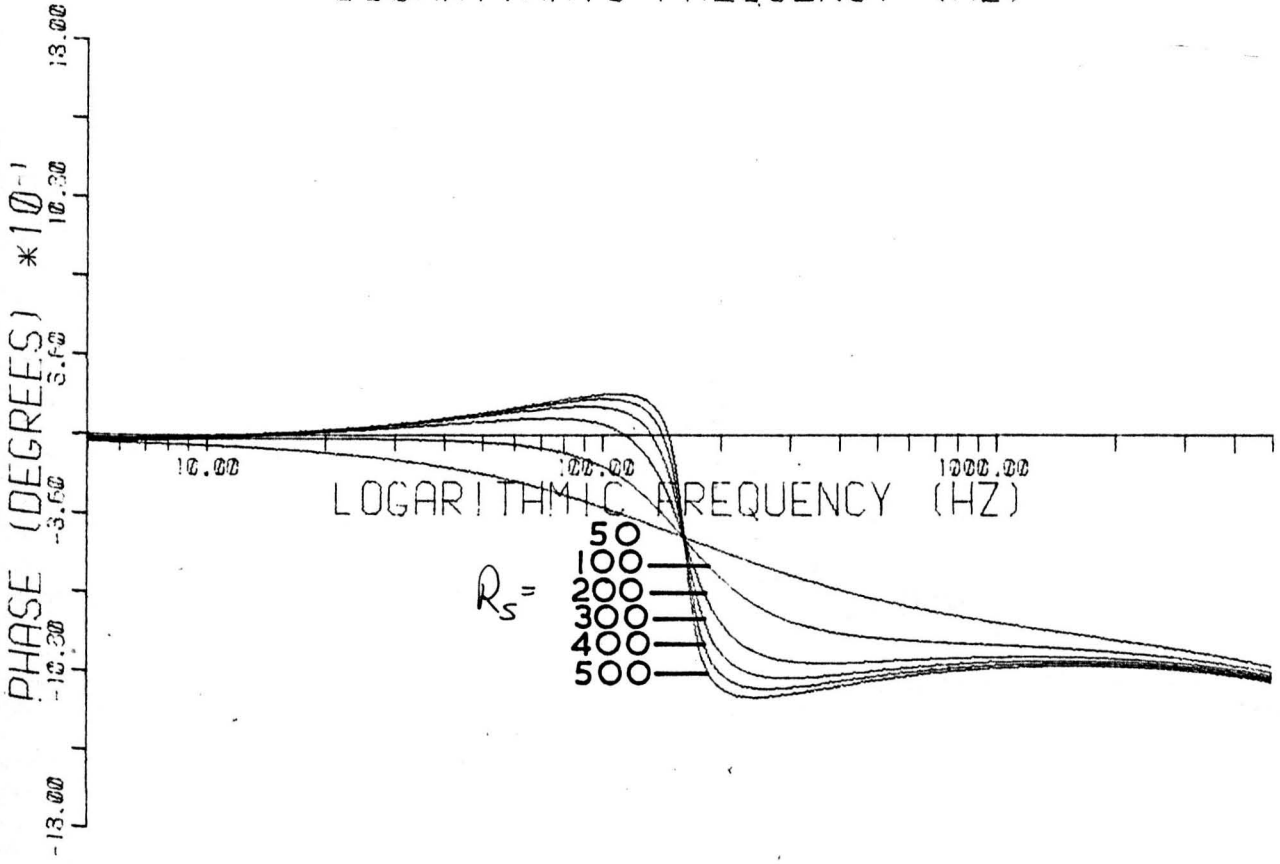
Computed Input Admittance GRAPH 8.31



Y11 - SAMPLE PERIOD = 300E-6 SECONDS

Computed Input Admittance GRAPH 8.32

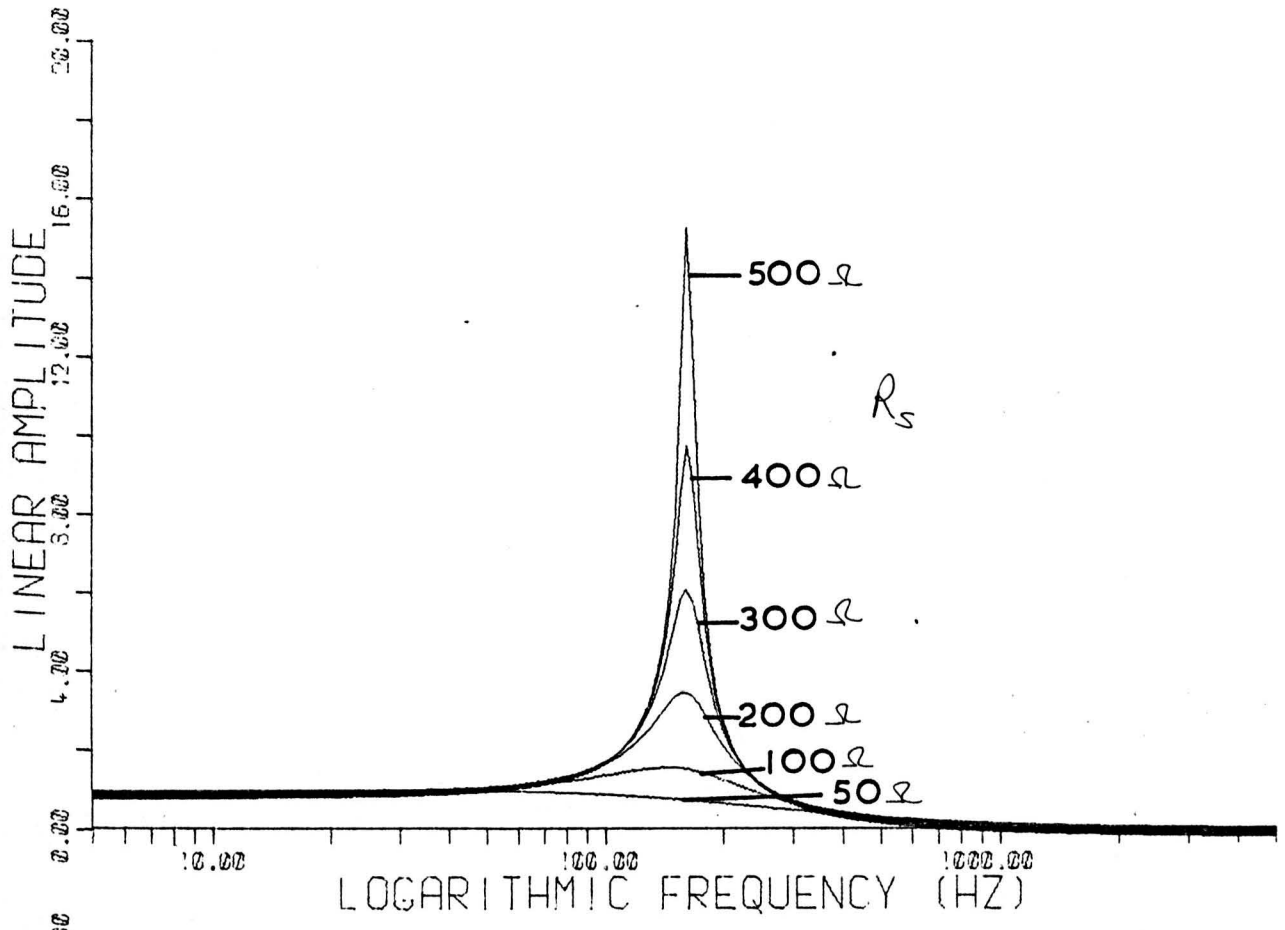


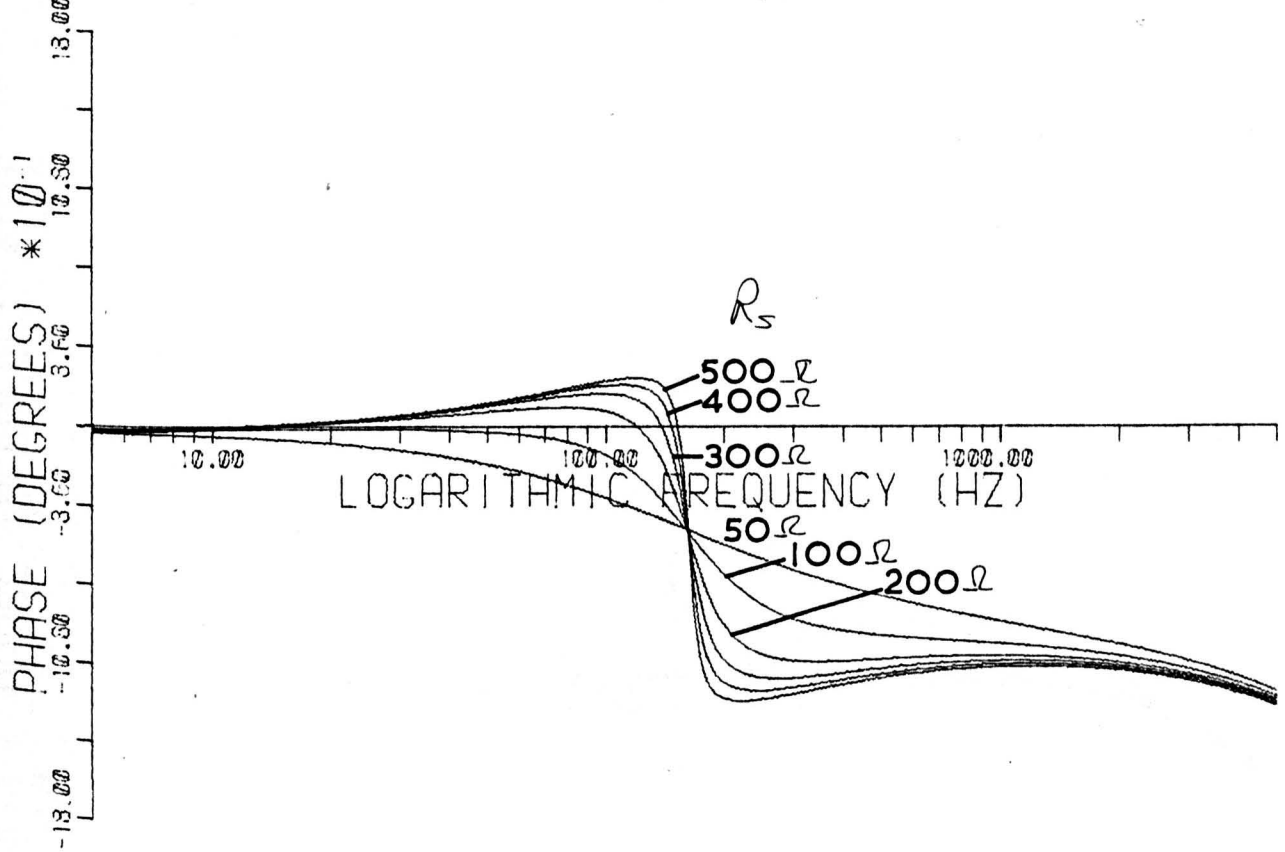


QUANTISATION TRANSFER FUNCTION AT PORT 1

T_s = 18.75 μs

GRAPH 8.33

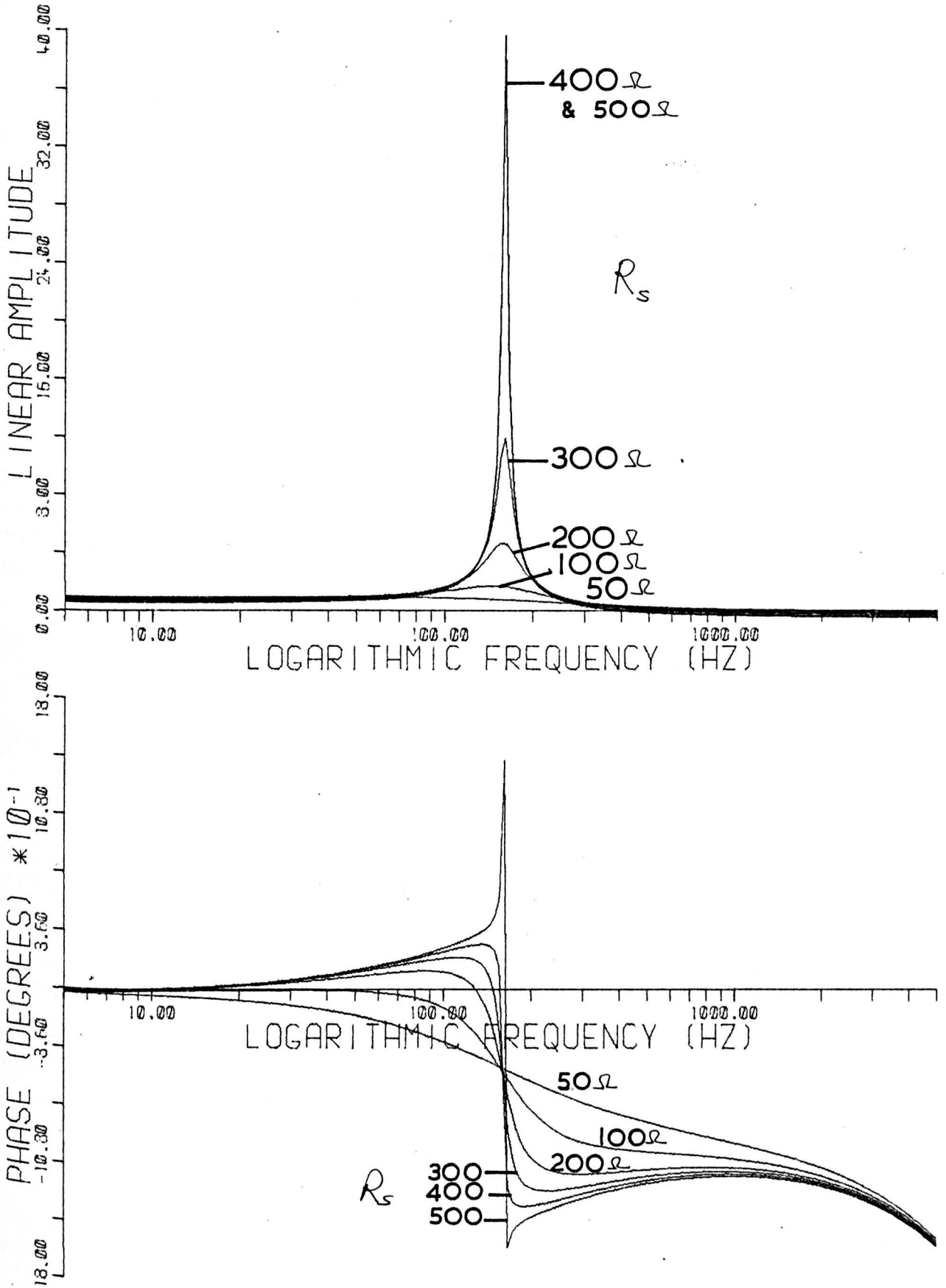




QUANTISATION TRANSFER FUNCTION AT PORT 1

T_s = 37.5 μs

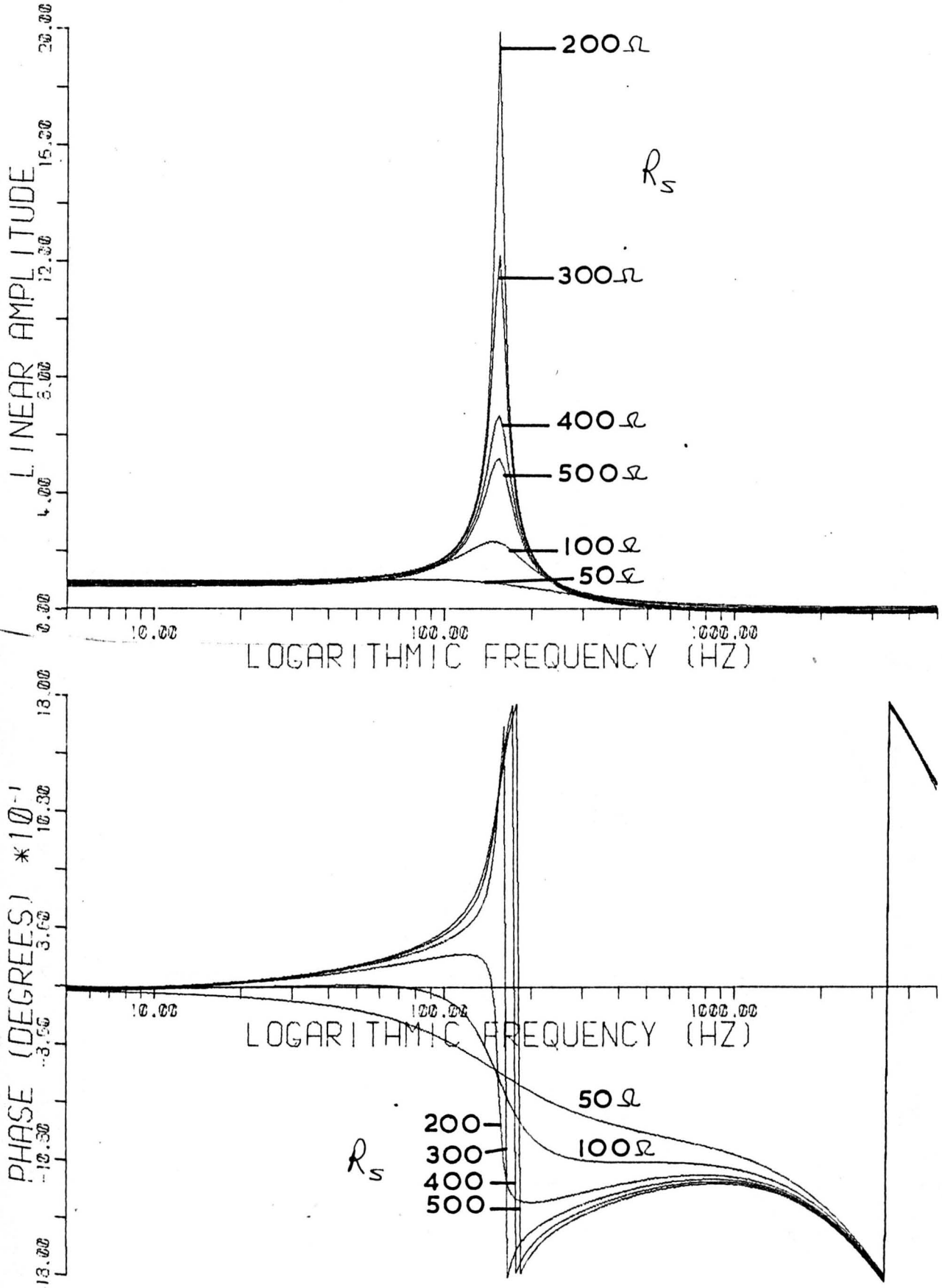
GRAPH 8.34



QUANTISATION TRANSFER FUNCTION AT PORT 1

$T_s = 75\ \mu s$

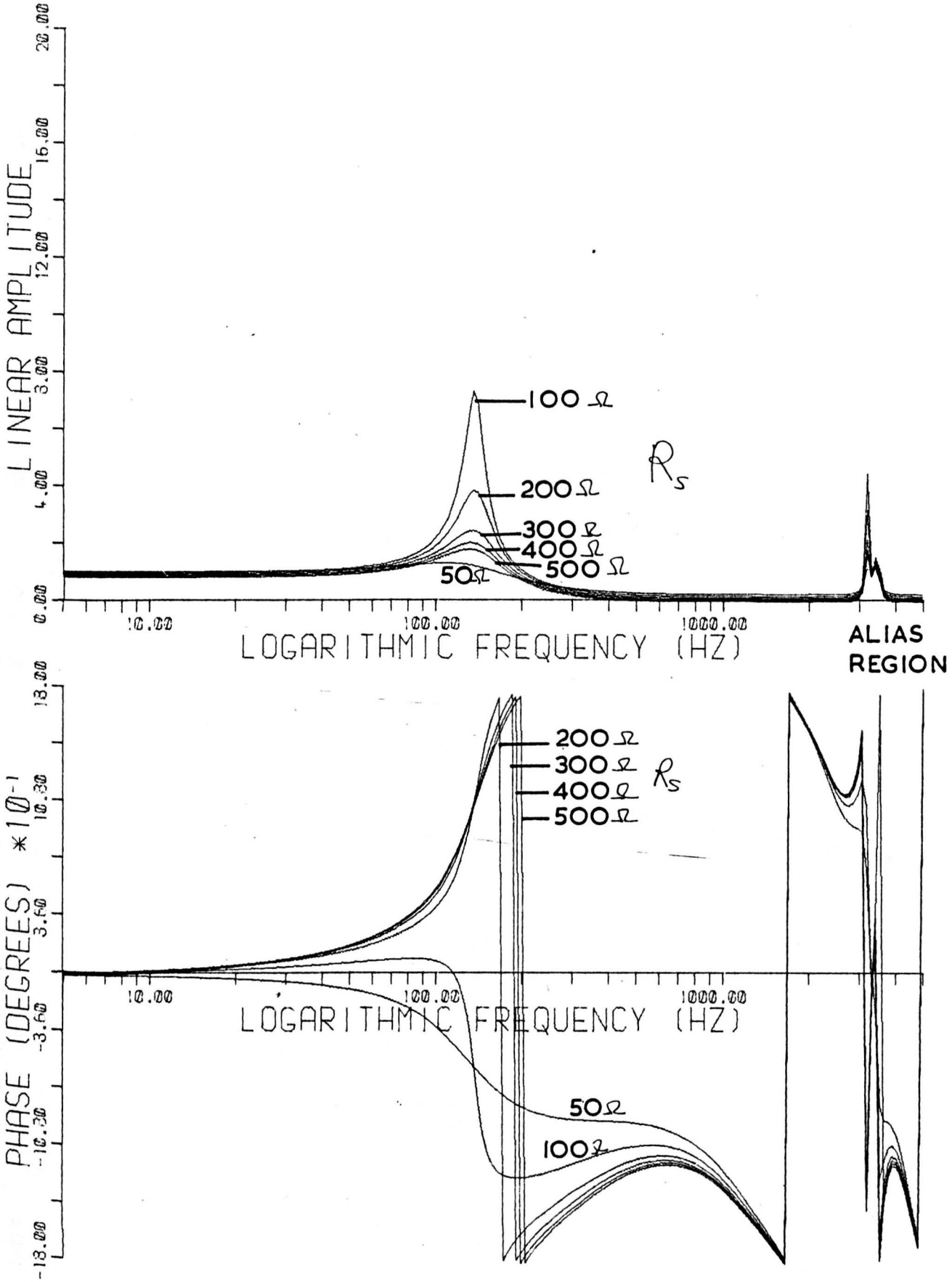
GRAPH 8.35



QUANTISATION TRANSFER FUNCTION AT PORT 1

$T_s = 150\ \mu s$

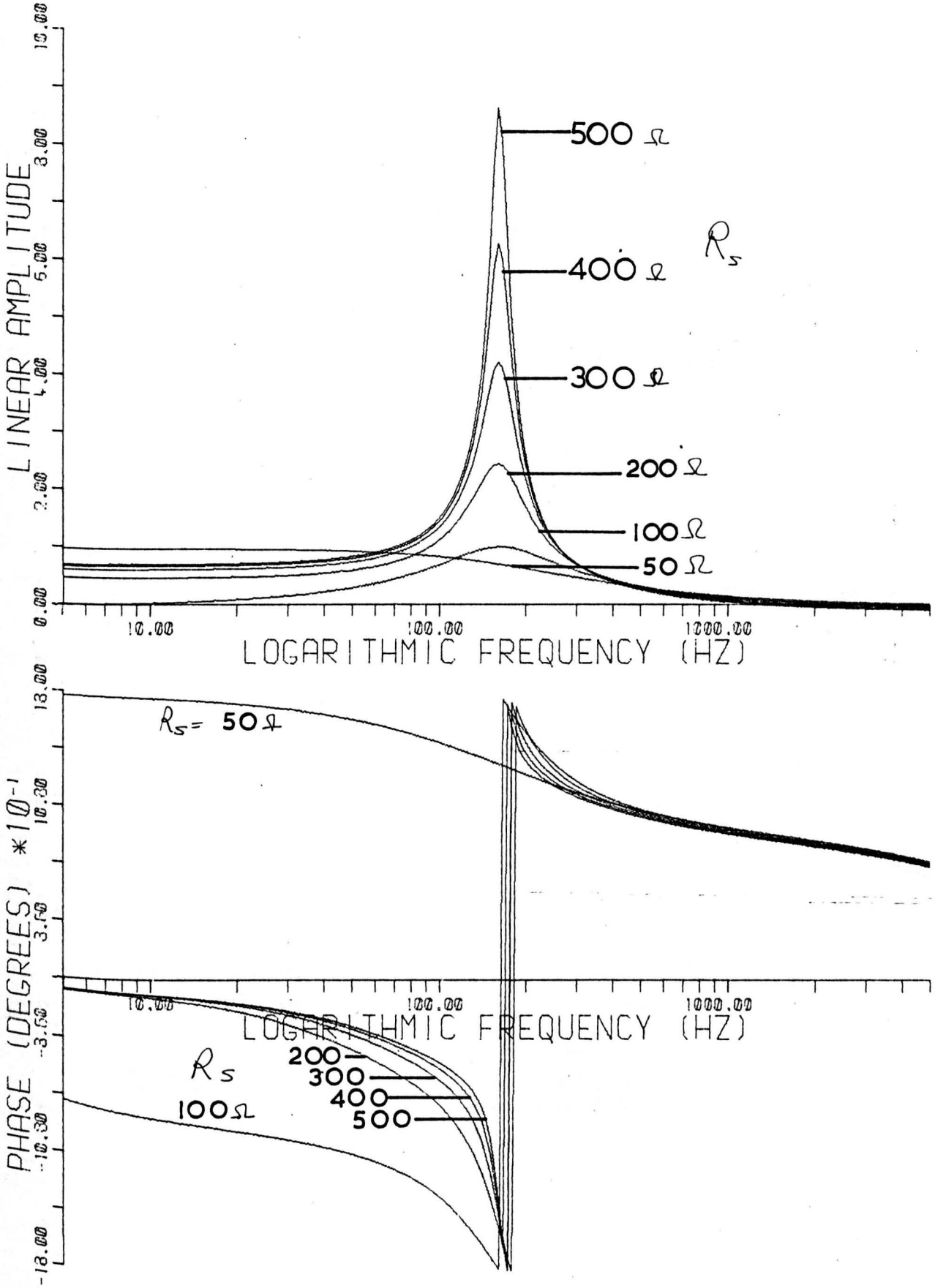
GRAPH 8.36



QUANTISATION TRANSFER FUNCTION AT PORT 1

T_s = 300 μs

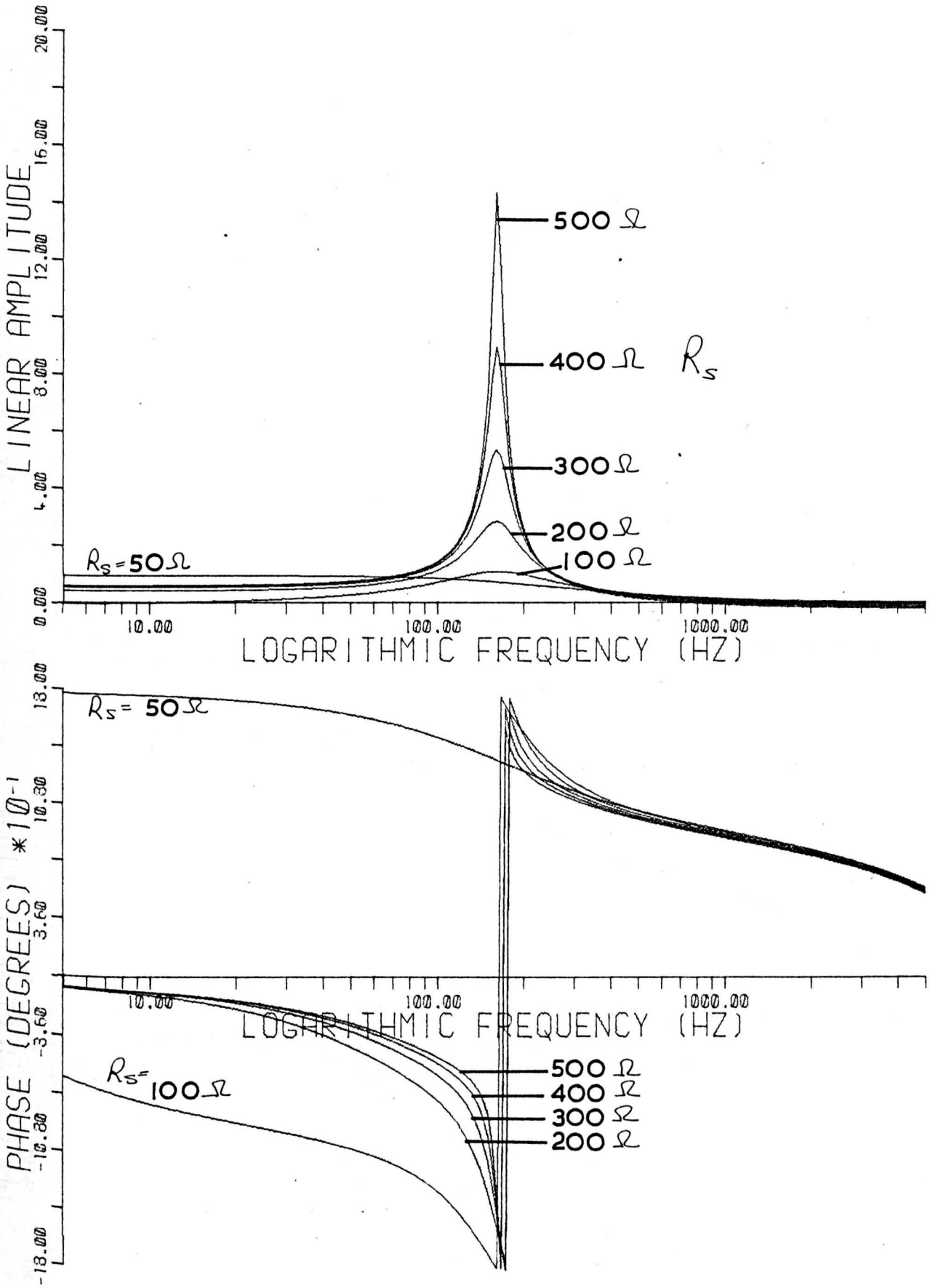
GRAPH 8.37



QUANTISATION TRANSFER FUNCTION AT PORT 2

T_s = 18.75 μs

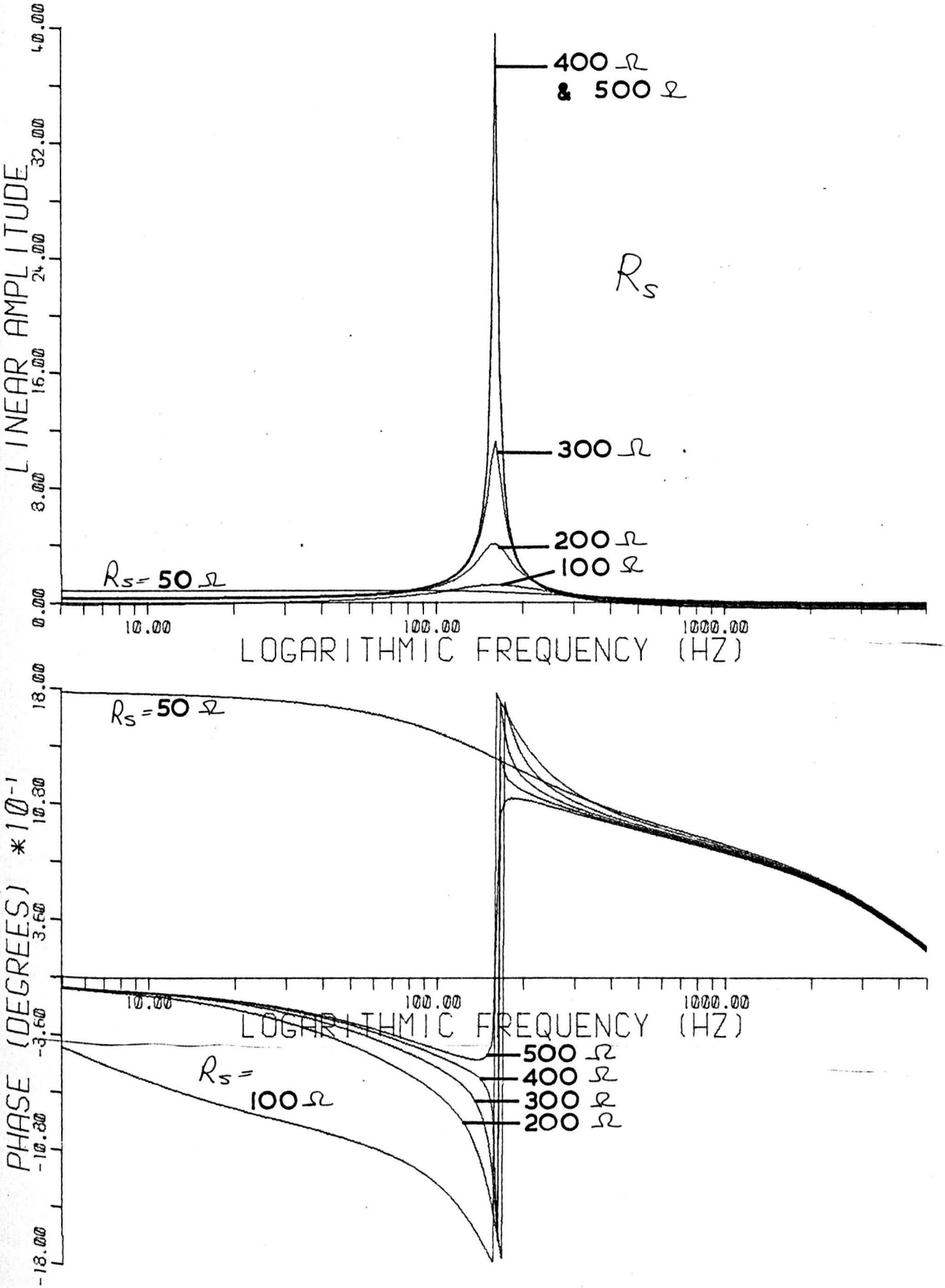
GRAPH 8.38



QUANTISATION TRANSFER FUNCTION AT PORT 2

T_s= 37.5 μs

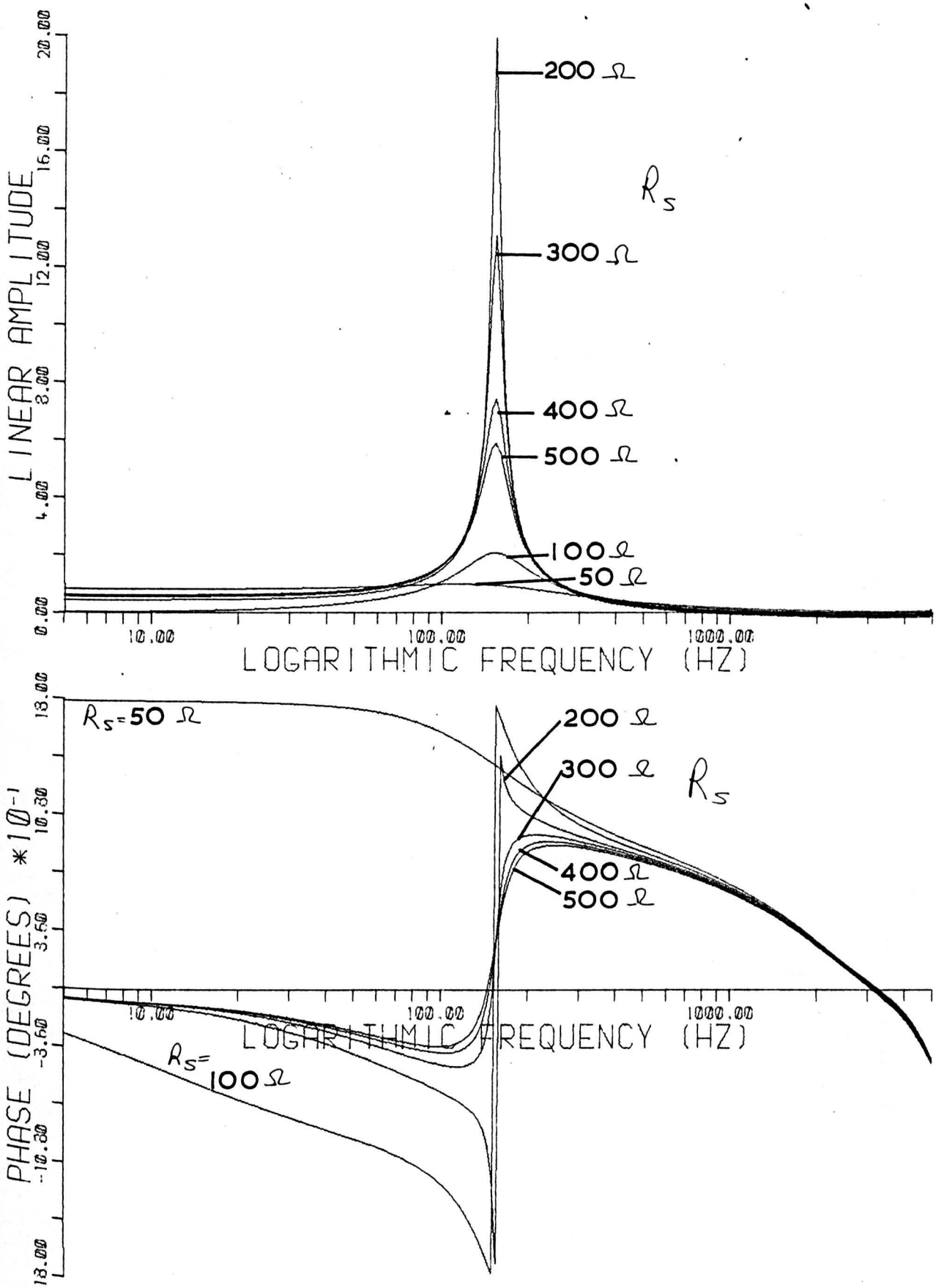
GRAPH 8.39



QUANTISATION TRANSFER FUNCTION AT PORT 2

T_s = 75 μs

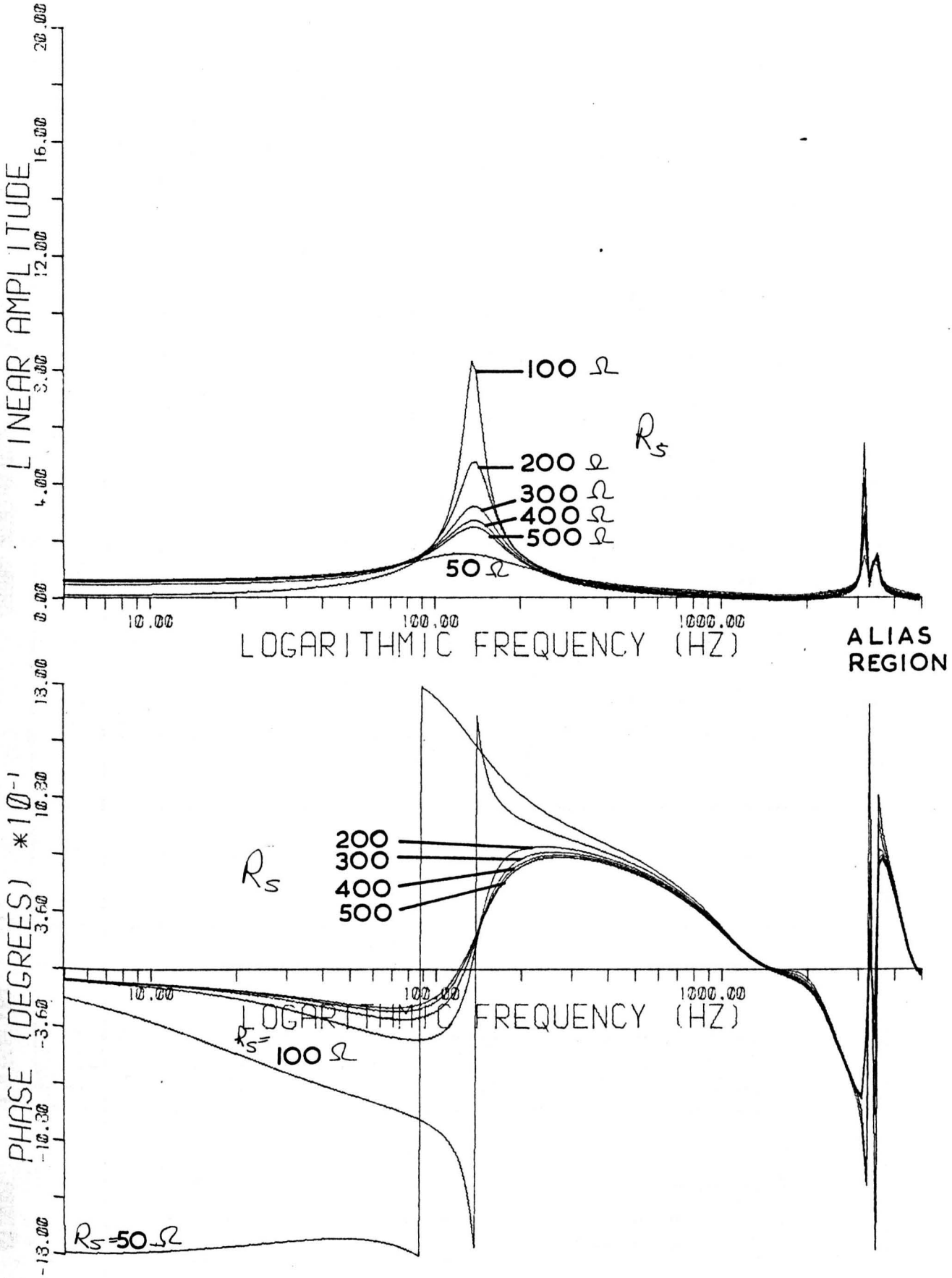
GRAPH 8.40



QUANTISATION TRANSFER FUNCTION AT PORT 2

T_s = 150 μs

GRAPH 8.41



QUANTISATION TRANSFER FUNCTION AT PORT 2

T_s = 300 μs

GRAPH 8.42

VALUES	SAMPLING PERIOD (μ s)	SHUNT RESISTANCE (Ω)					
		50	100	200	300	400	500
	18.75	-	140.5	158.7	160.7	161.7	161.7
Peak	37.5	-	143.7	158.9	161.1	161.1	161.1
Frequency	75.0	-	145.8	158.9	161.1	161.1	161.1
(Hz)	150.0	-	146.8	154.4	154.4	154.4	154.4
	300.00	-	135.3	136.3	134.2	132.1	131.1
					UNSTABLE REGION		
	18.75	-	1.5209	3.1389	5.0019	7.1406	9.6166
	37.5	-	1.5977	3.5489	6.1913	9.9148	15.5050
Peak	75.0	-	1.7881	4.8614	12.0710	47.1618	53.2516
Gain	150.0	-	2.3931	20.7234	12.4024	6.8675	5.4111
	300.0	-	7.3895	3.9318	2.5742	2.1973	2.0212
			OSCILLIATORY REGION		UNSTABLE REGION		
Peak	18.75	-	54.9	113.3	180.5	257.7	347.0
Noise	37.5	-	57.7	128.1	223.4	357.8	559.5
Voltage	75.0	-	64.5	175.4	435.6	1701.8	1921.6
(mVrms)	150.0	-	86.4	747.8	447.5	247.8	195.3
(for 36.1	300.0	-	266.6	141.9	92.9	79.3	72.9
mVrms input)					UNSTABLE REGION		

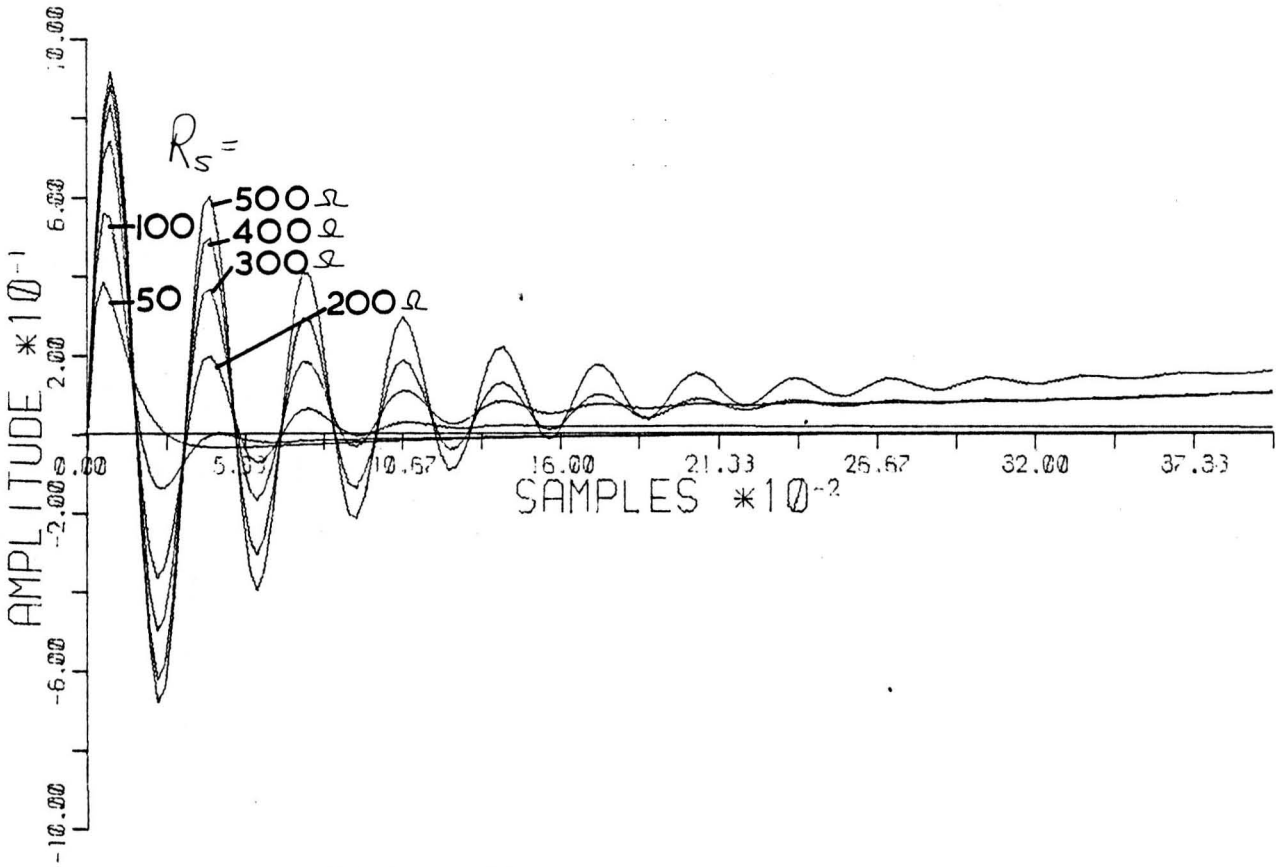
TABLE 8.35

Limit cycle Peak Frequencies, gains and noise Voltages for Port 1

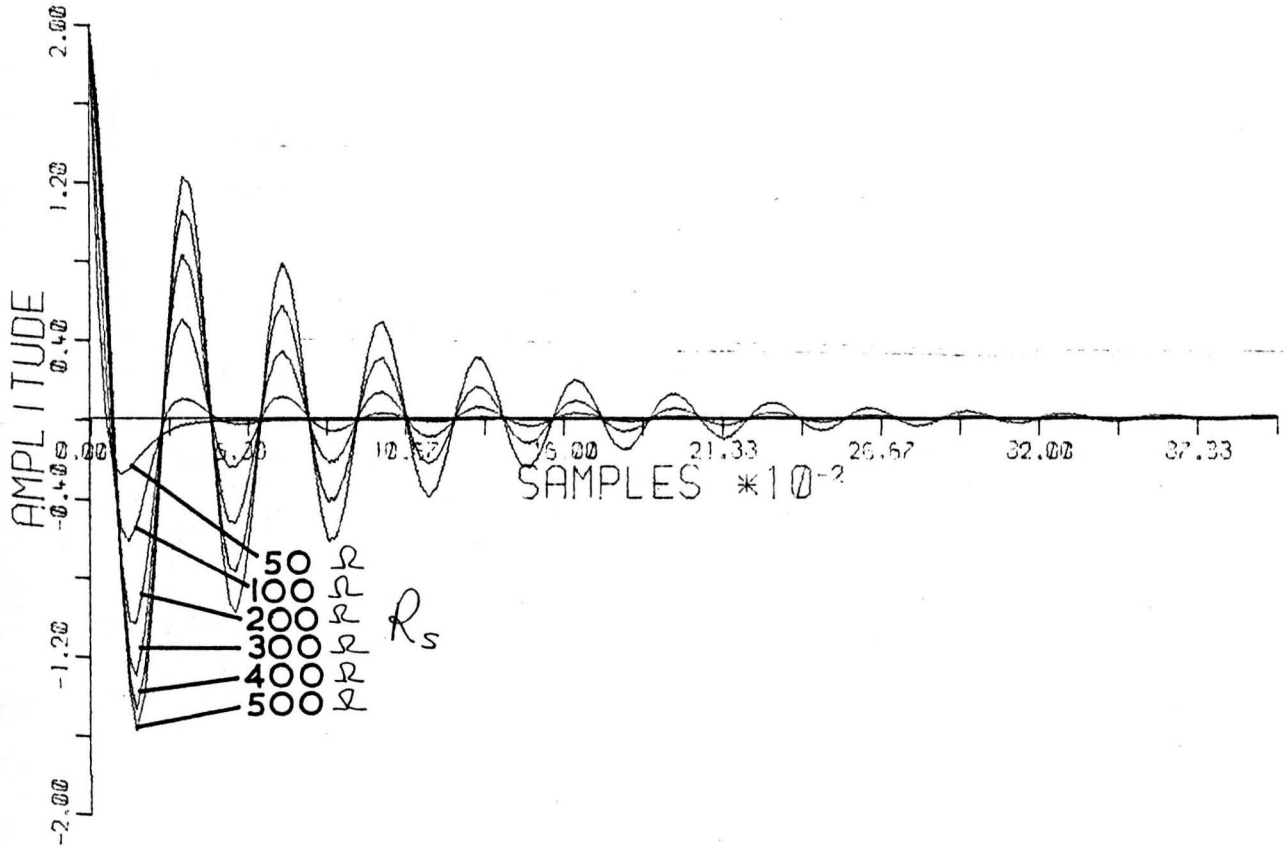
VALUES	SAMPLING PERIOD (μ s)	SHUNT RESISTANCE (Ω)					
		50	100	200	300	400	500
	18.75	-	162.6	160.7	161.7	161.7	161.7
Peak	37.5	-	161.8	161.1	161.1	161.1	161.1
Frequency	75.0	-	160.3	158.9	161.1	161.1	161.1
(Hz)	150.0	-	154.0	154.4	154.4	154.4	154.4
	300.0	-	136.3	138.4	138.4	136.3	136.3
					UNSTABLE REGION		
	18.75	-	1.0474	2.4929	4.2852	6.3671	8.7841
	37.5	-	1.1465	2.9246	5.4876	9.1274	14.6110
Peak	75.0	-	1.3885	4.9999	11.4450	46.3171	53.4487
Gain	150.0	-	2.1470	20.8838	13.2456	7.5721	6.0857
	300.0	-	8.4053	4.8748	3.3242	2.8989	2.7028
			OSCILLIATORY REGION		UNSTABLE REGION		
Peak Noise	18.75	-	37.8	90.0	154.6	229.8	317.0
Voltage	37.5	-	41.4	105.5	198.0	329.4	527.2
(mVrms)	75.0	-	50.1	180.4	413.0	1671.3	1928.7
(for 36.1	150.0	-	77.5	753.6	478.0	273.2	219.6
mVrms input)	300.0	-	303.3	175.9	120.0	104.6	97.5
					UNSTABLE REGION		

TABLE 8.36

Limit cycle Peak Frequencies, gains and noise Voltages for Port 2



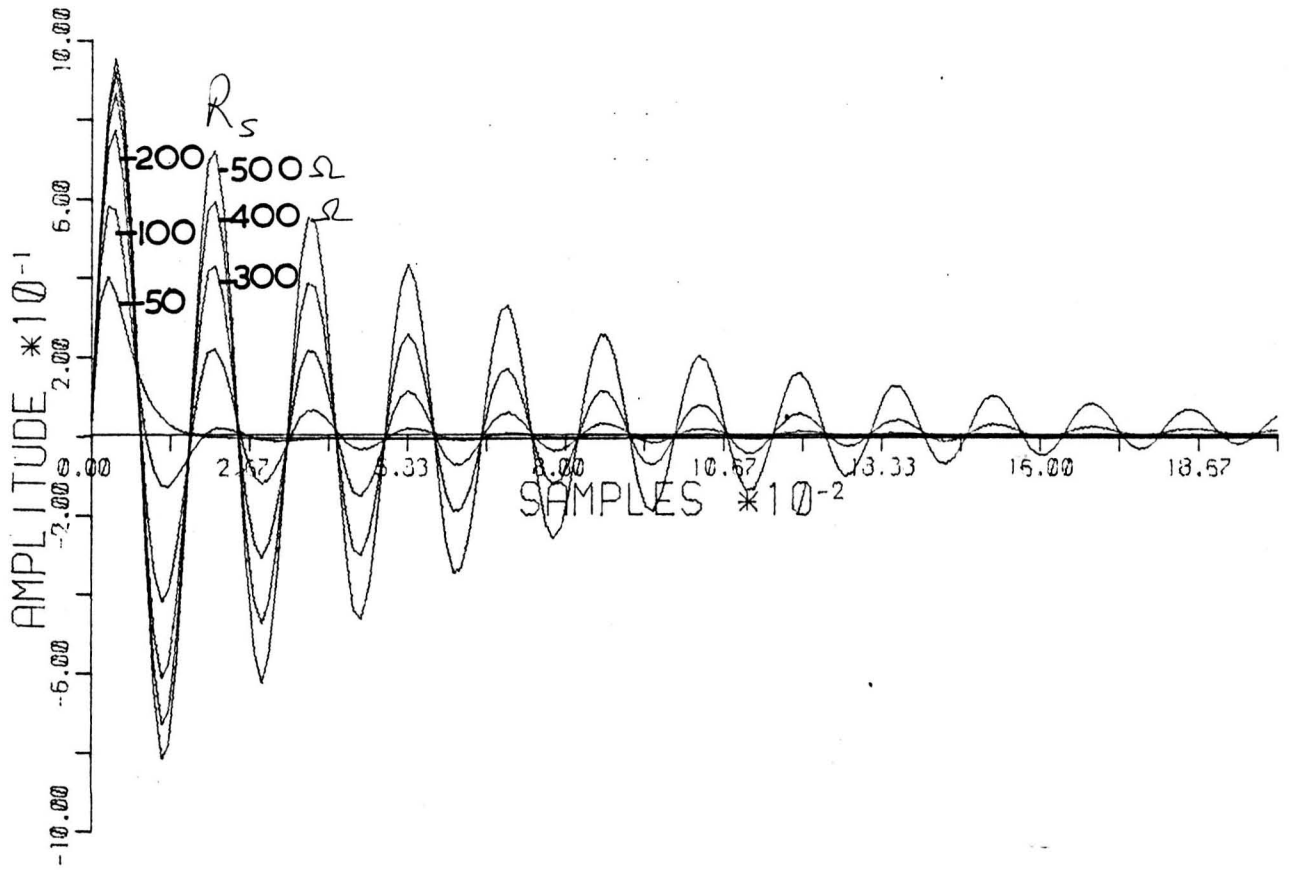
IMPULSE RESPONSE



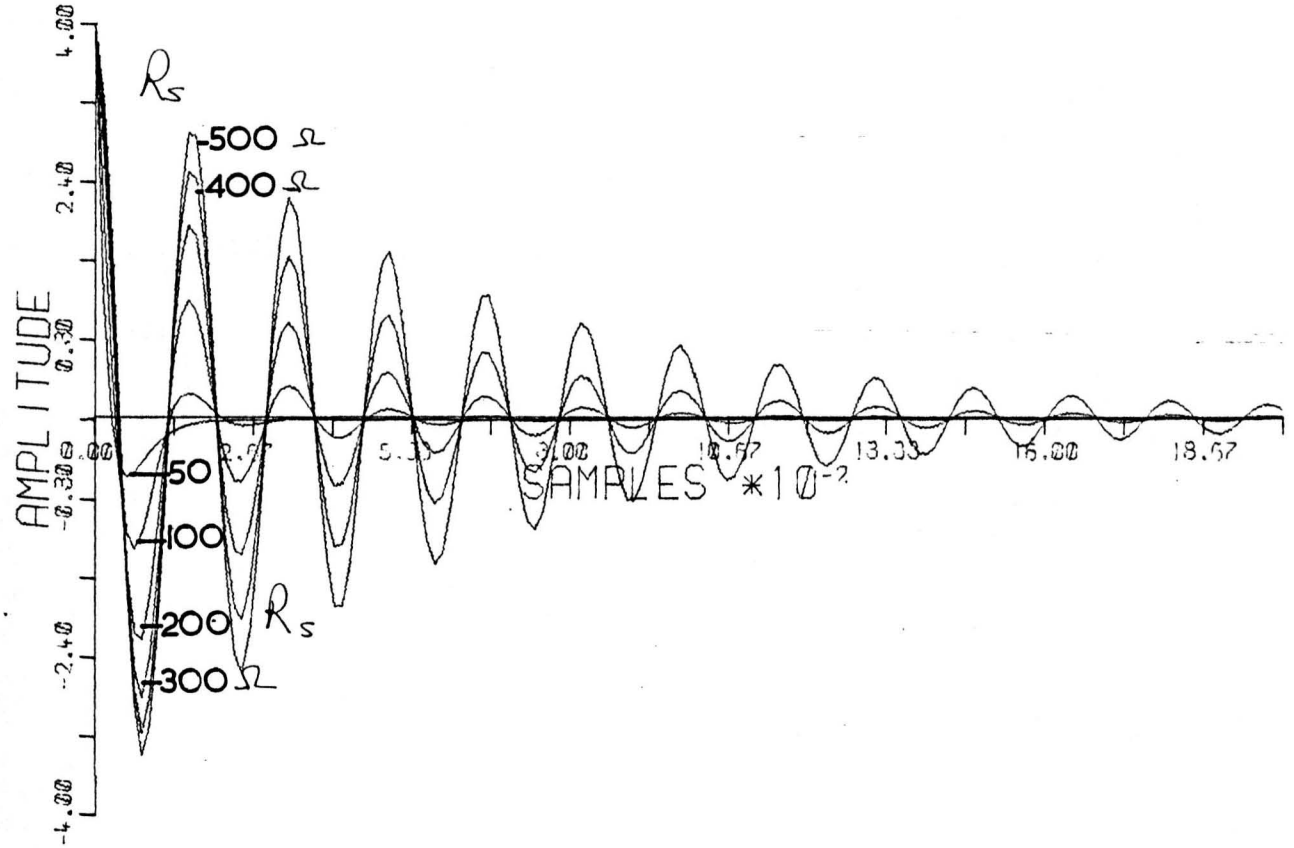
INVERSE TRANSFORM

Z11 - SAMPLE PERIOD = 18.75E-6 SECONDS

GRAPH 8.43



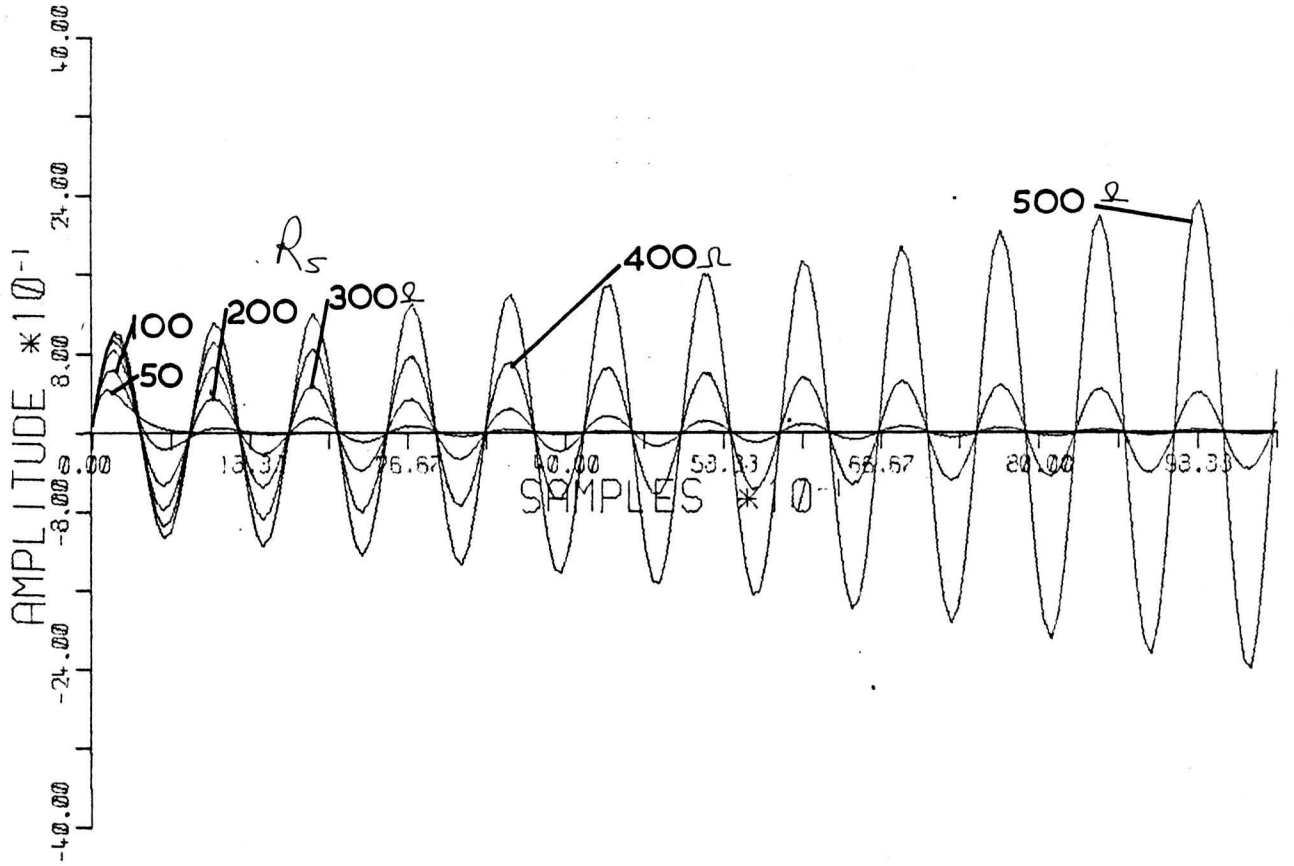
IMPULSE RESPONSE



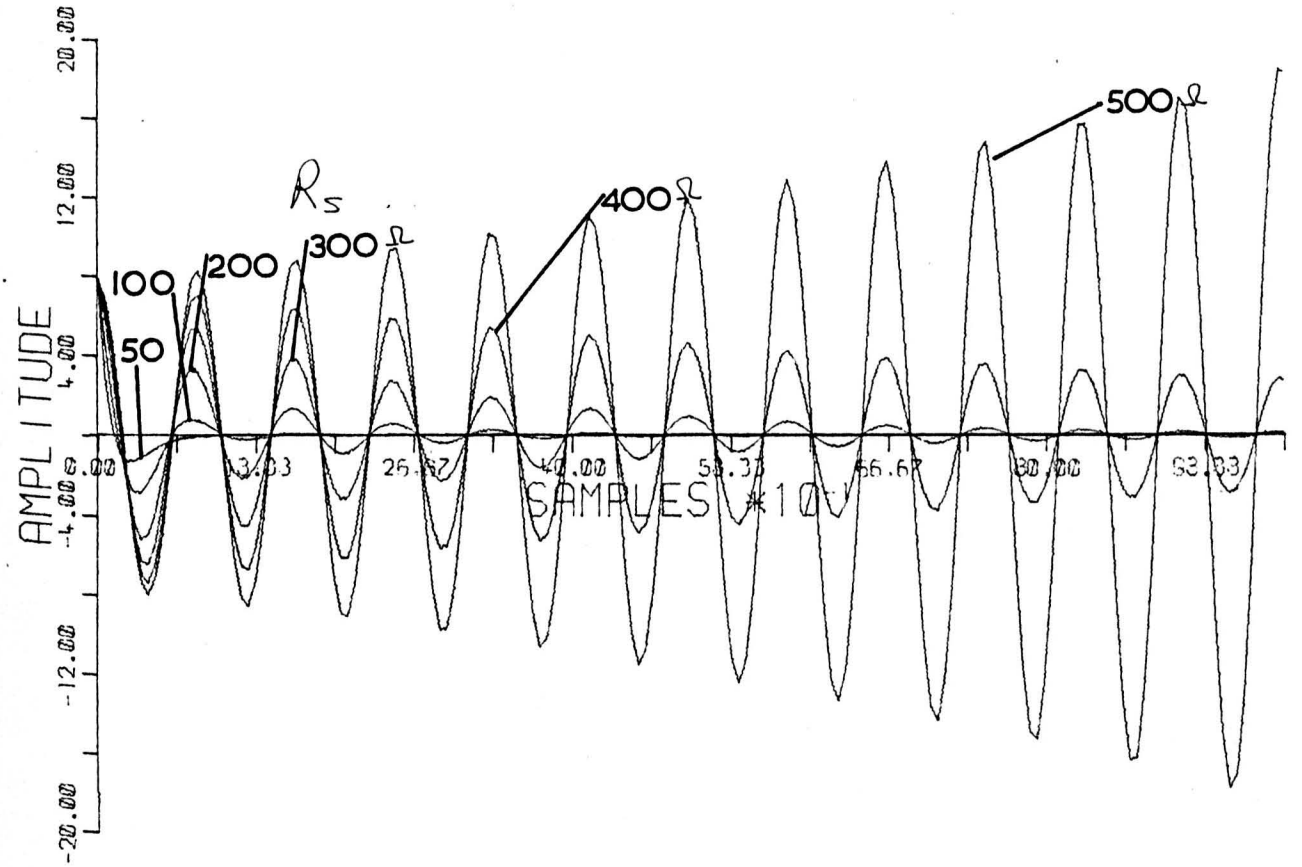
INVERSE TRANSFORM

Z11 - SAMPLE PERIOD = 37.5E-6 SECONDS

GRAPH **8.44**



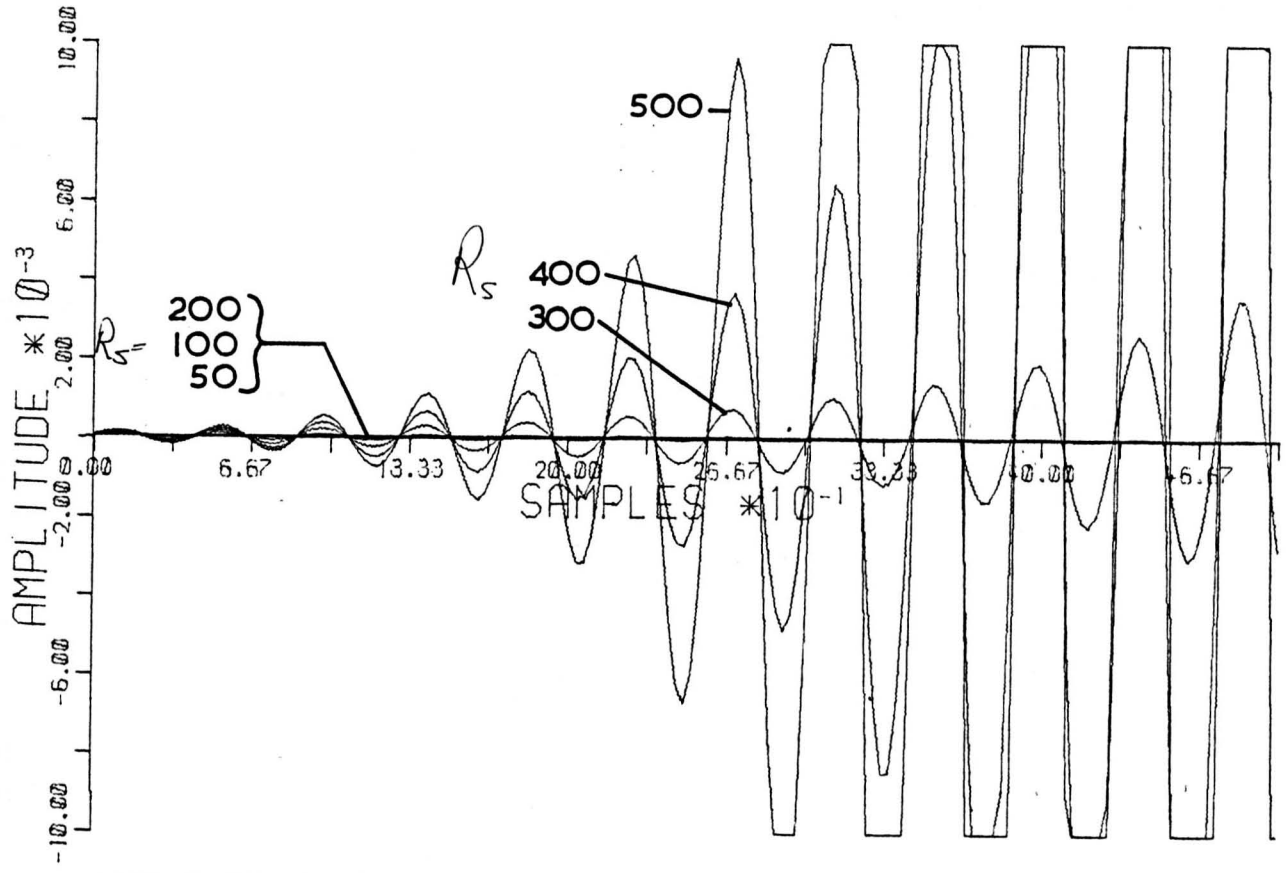
IMPULSE RESPONSE



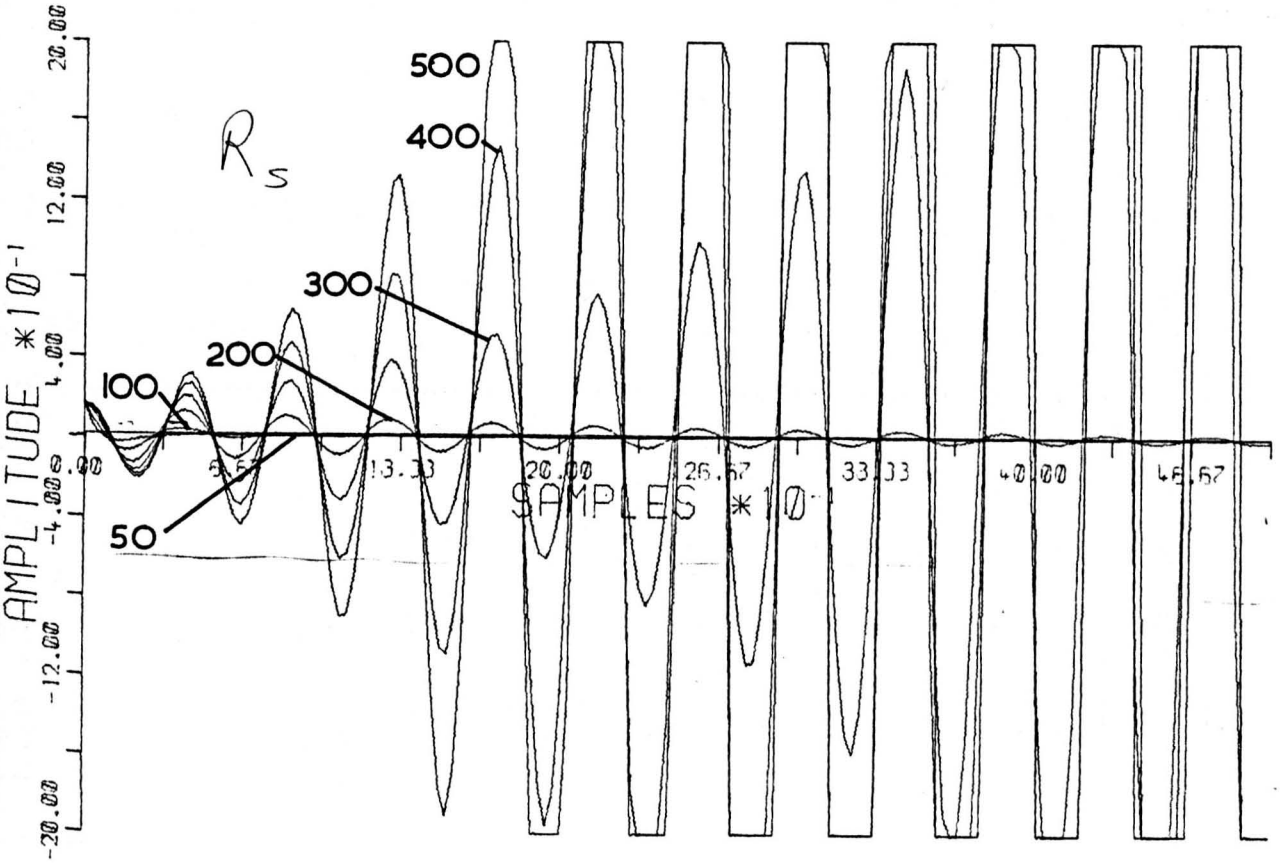
INVERSE TRANSFORM

Z11 - SAMPLE PERIOD = 75E-6 SEÇONDS

GRAPH **8.45**



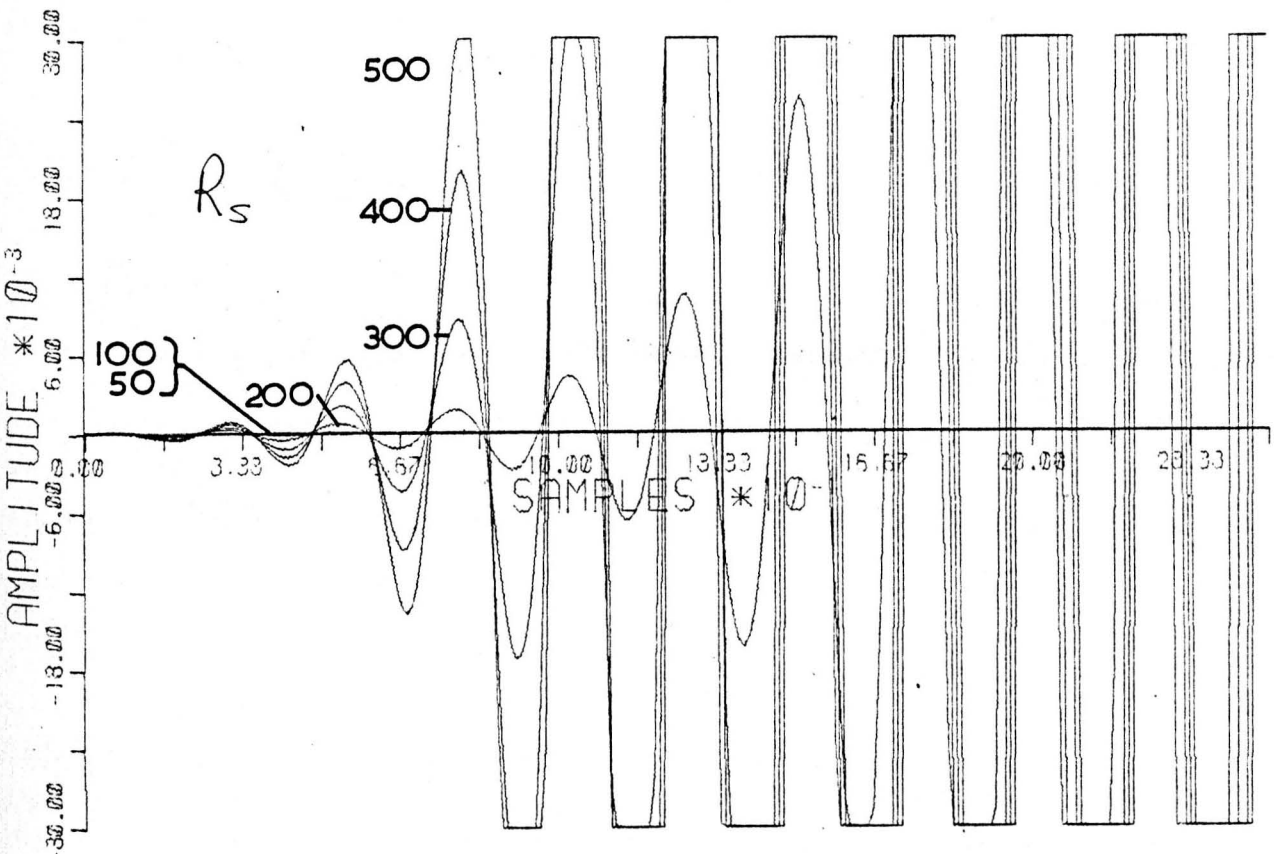
IMPULSE RESPONSE



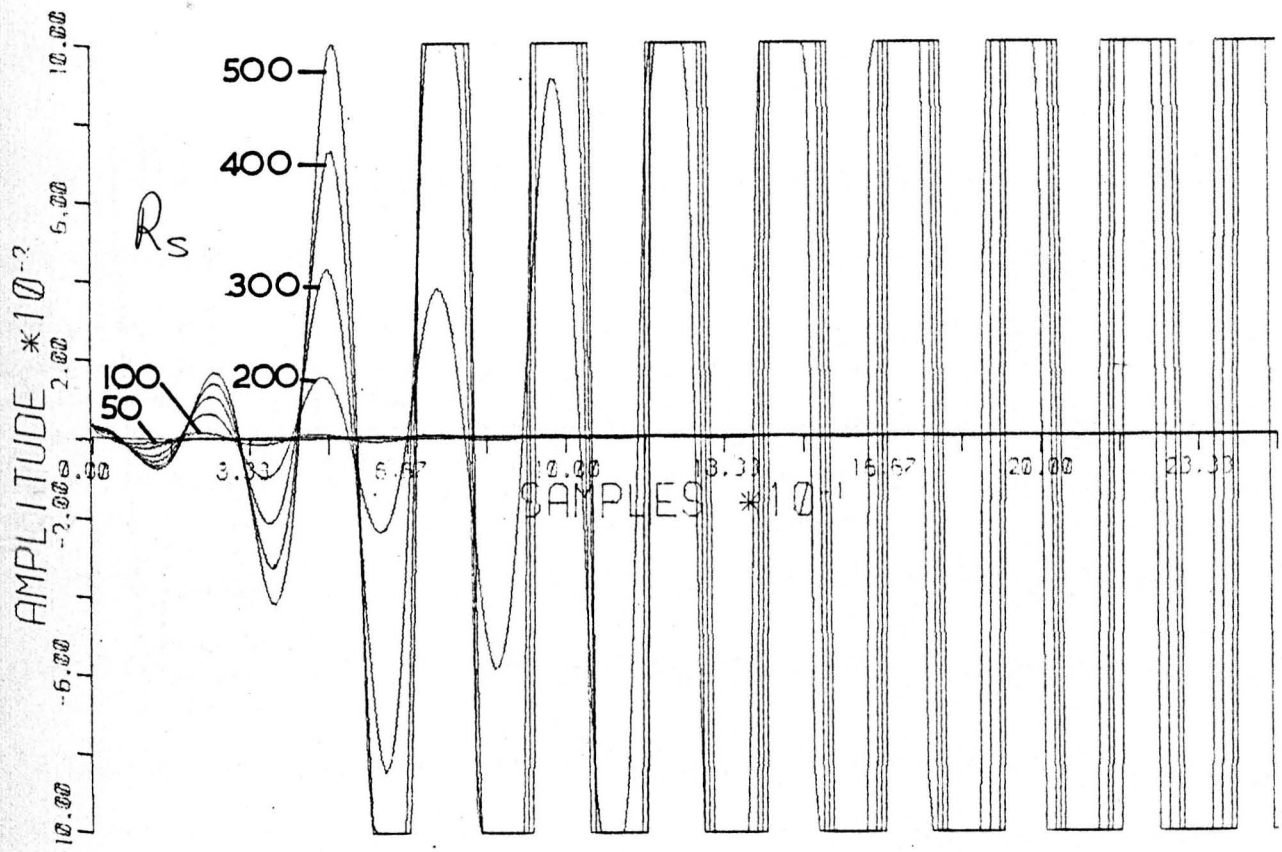
INVERSE TRANSFORM

Z11 - SAMPLE PERIOD = 150E-6 SECONDS

GRAPH 8.46



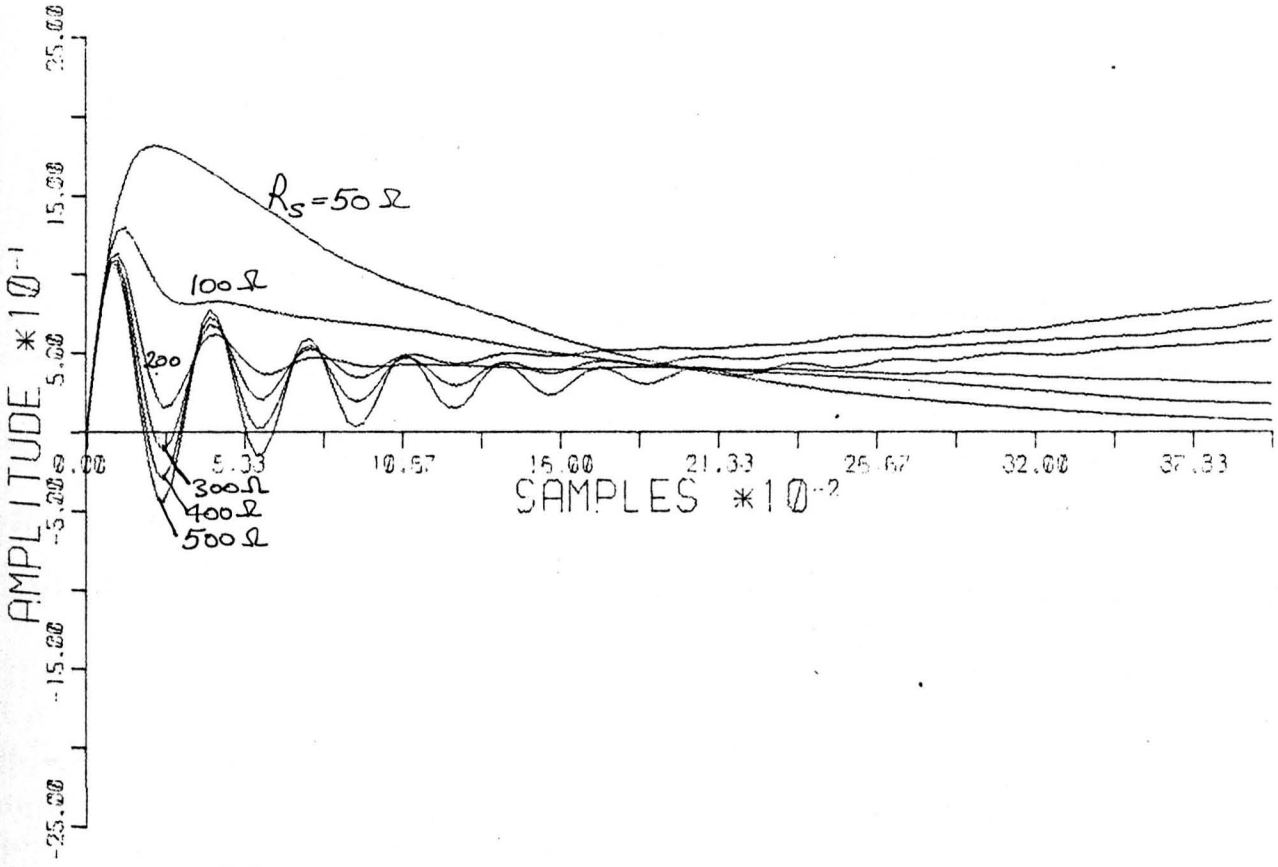
IMPULSE RESPONSE



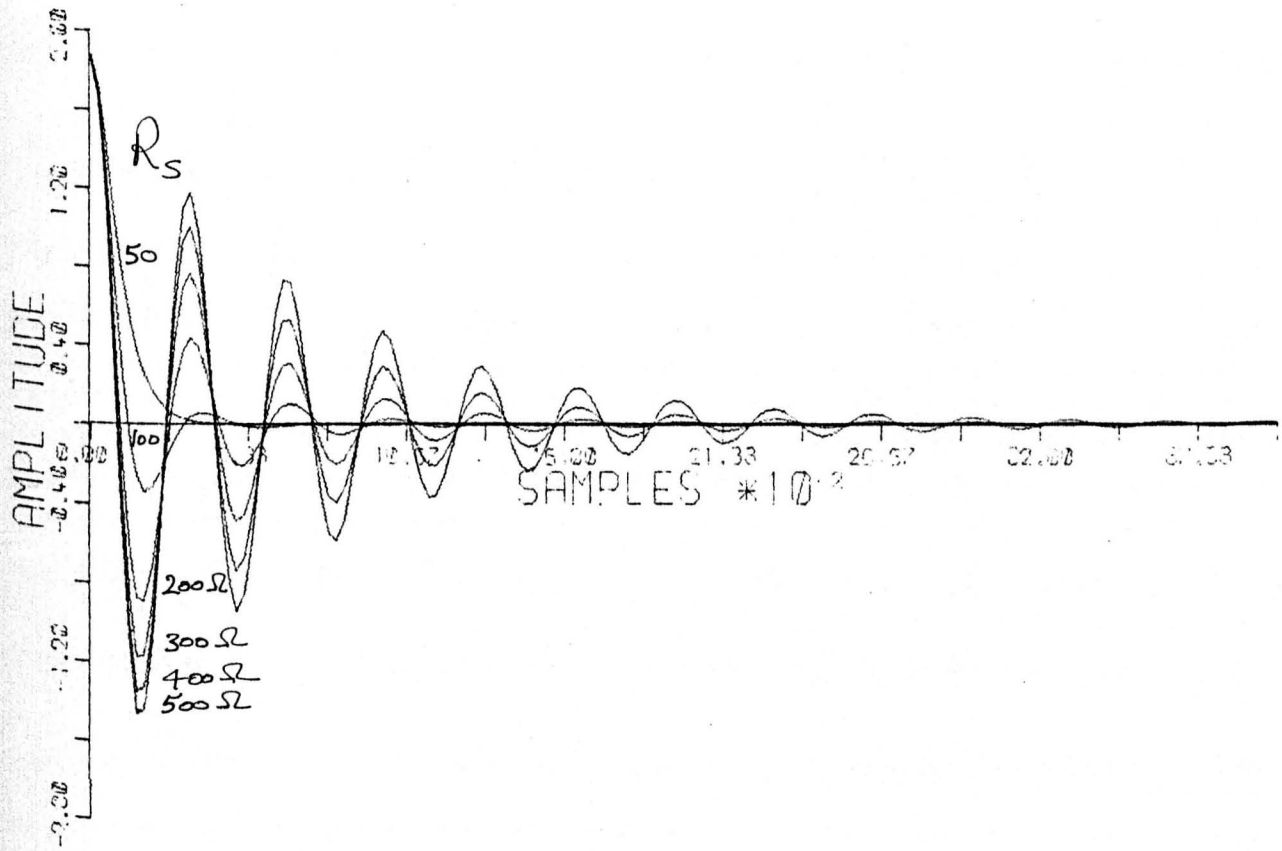
INVERSE TRANSFORM

Z11 - SAMPLE PERIOD = 300E-6 SECONDS

GRAPH 8.47



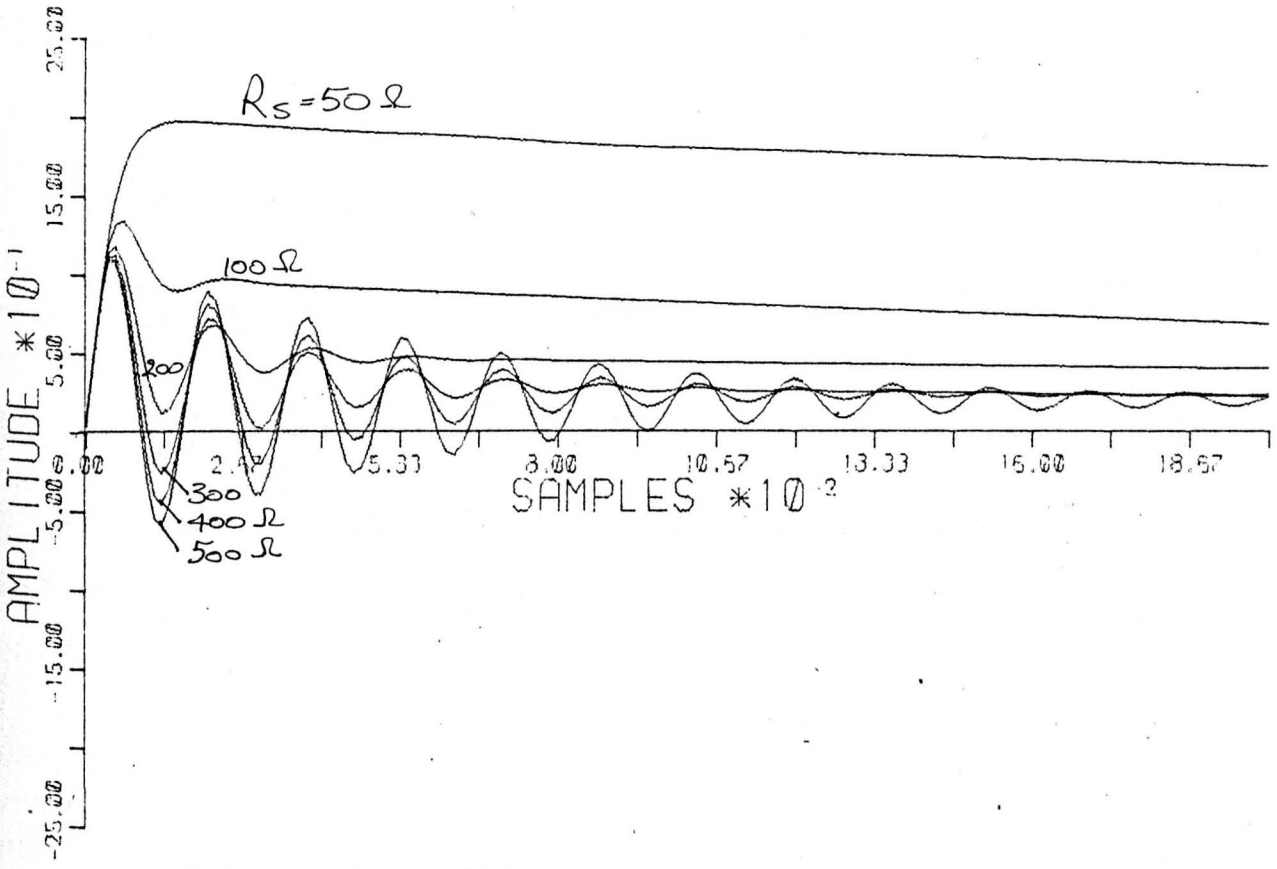
IMPULSE RESPONSE



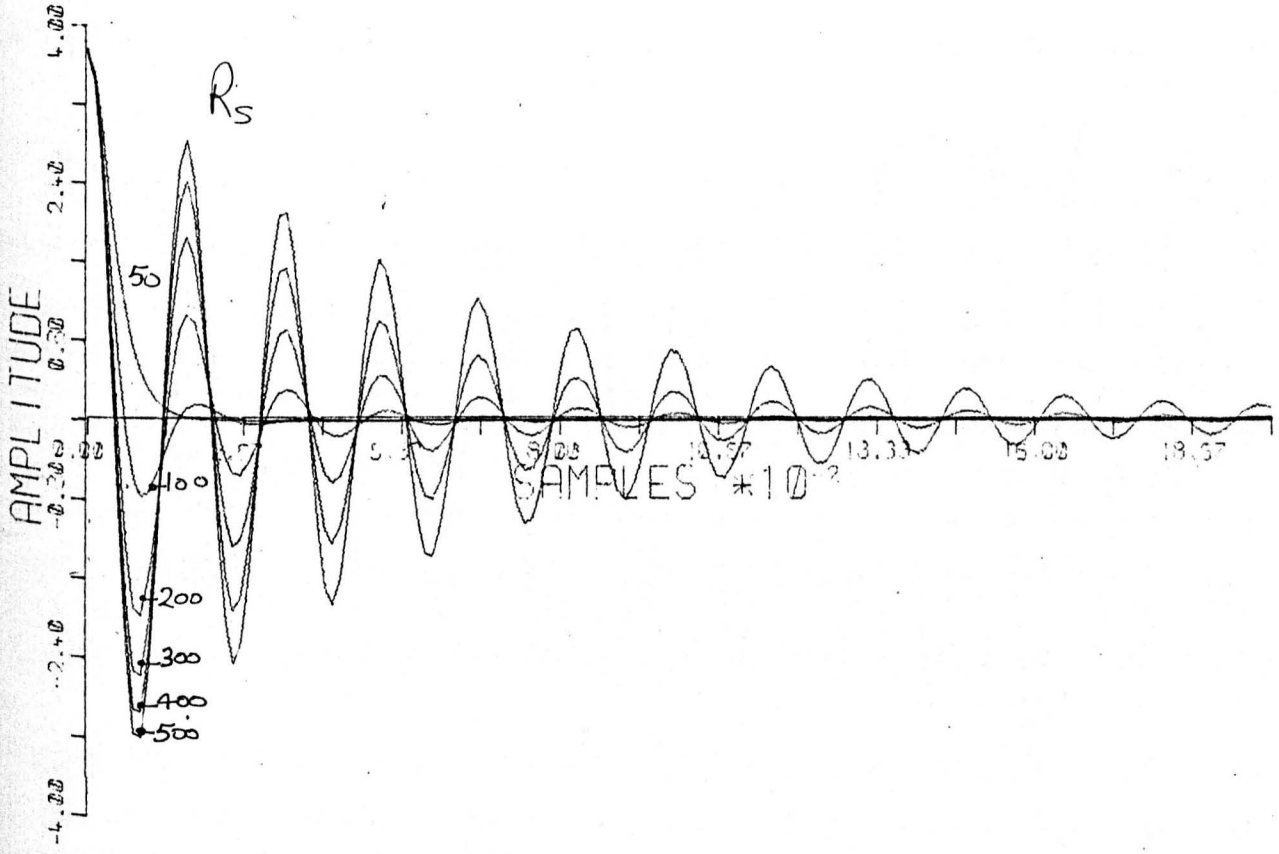
INVERSE TRANSFORM

Z21 - SAMPLE PERIOD = 18.75E-6 SECONDS

GRAPH 8.48



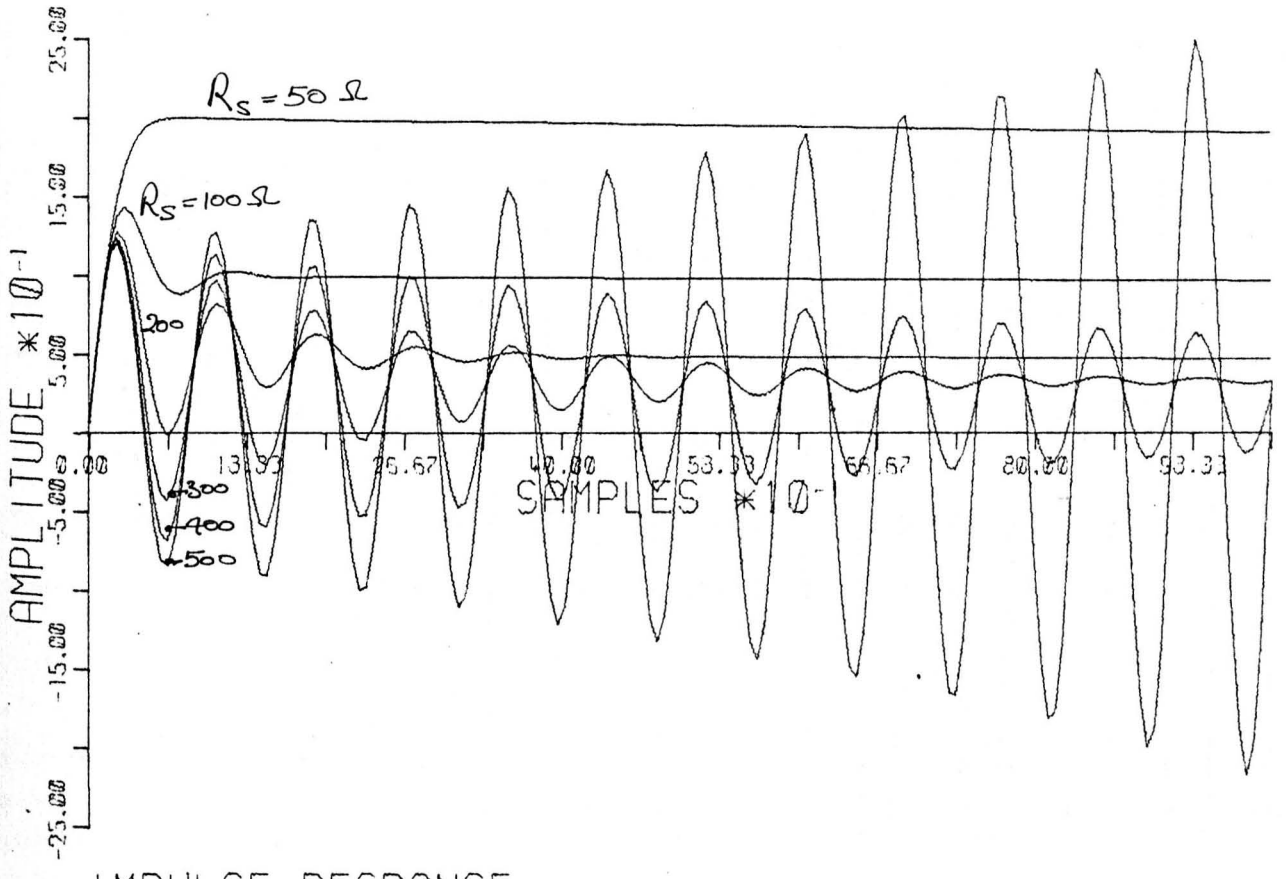
IMPULSE RESPONSE



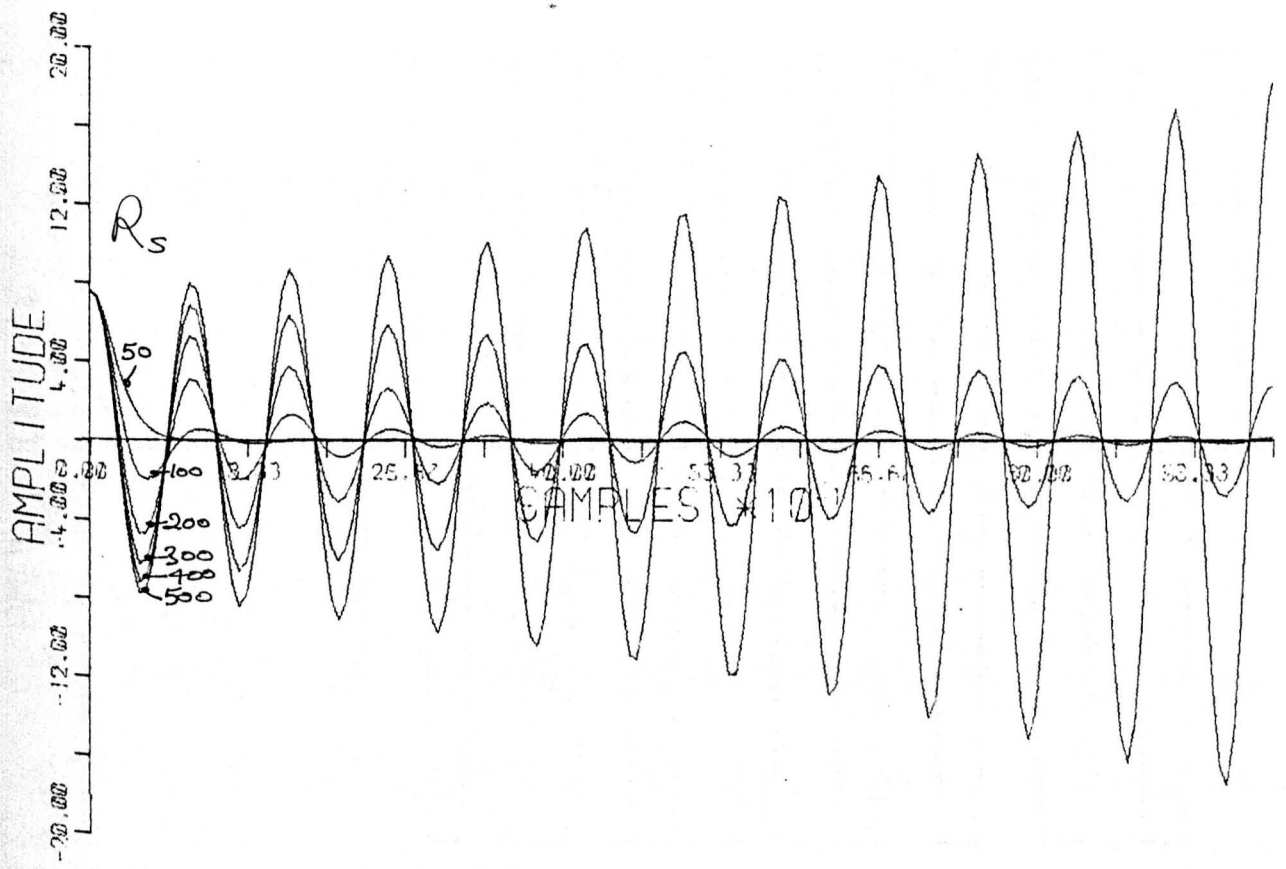
INVERSE TRANSFORM

Z21 - SAMPLE PERIOD = 37.5E-6 SECONDS

GRAPH 8.49



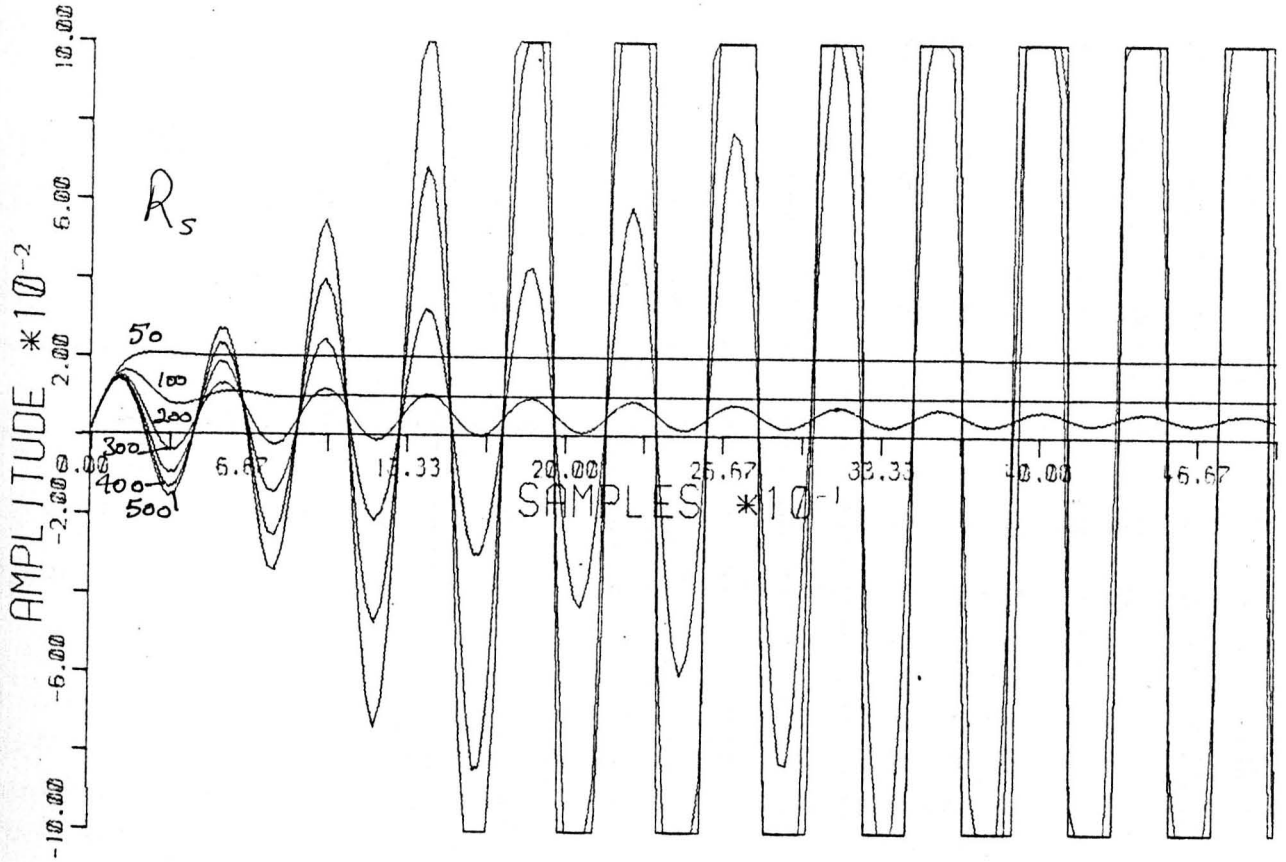
IMPULSE RESPONSE



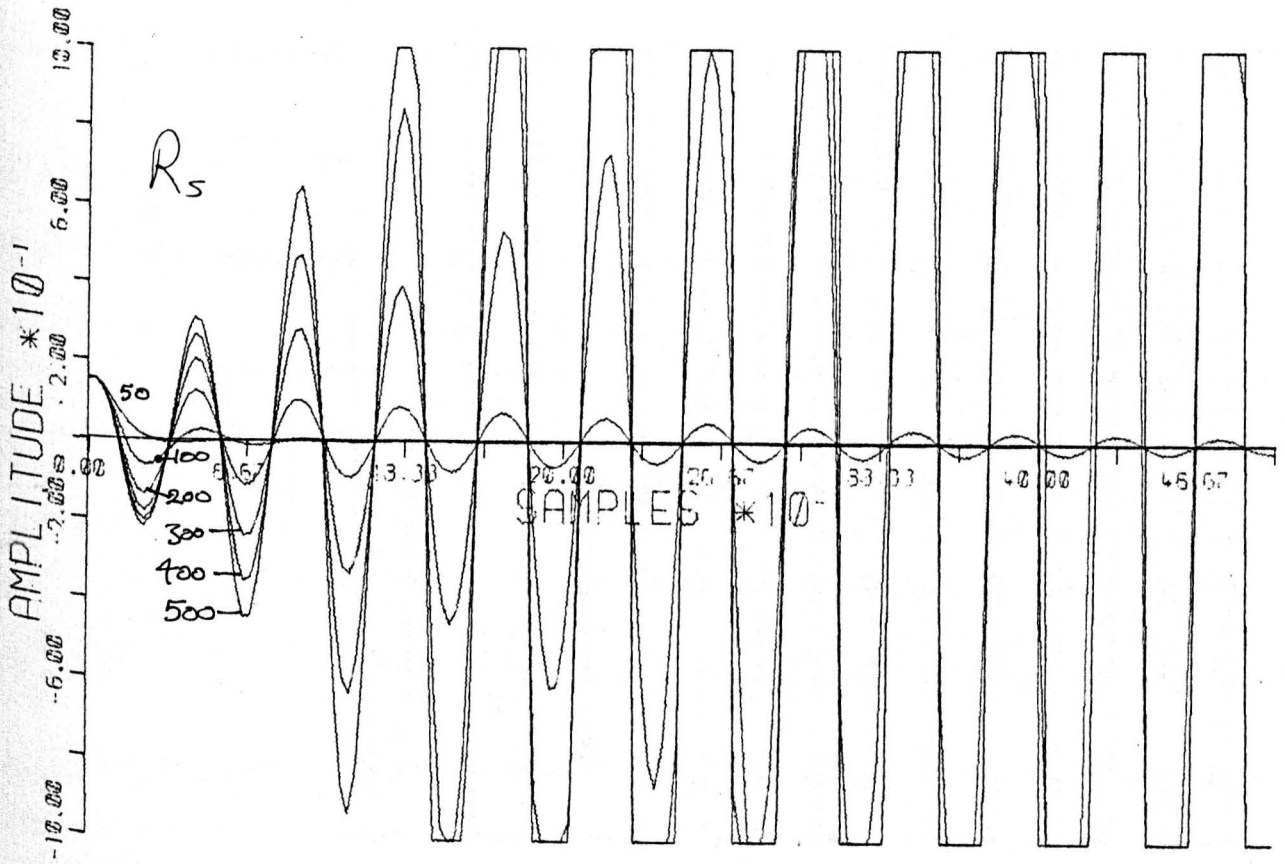
INVERSE TRANSFORM

Z21 - SAMPLE PERIOD = 75E-6 SECONDS

GRAPH 8.50



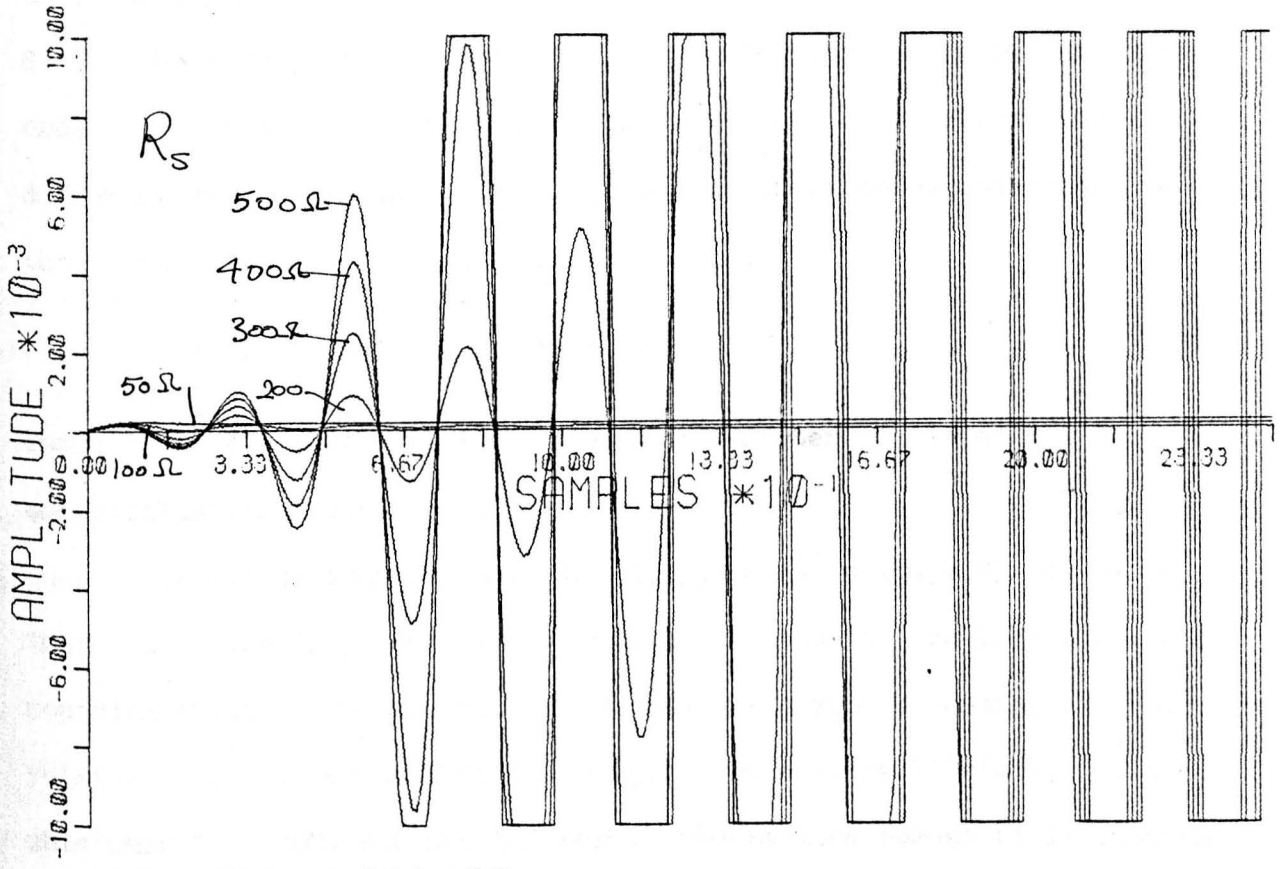
IMPULSE RESPONSE



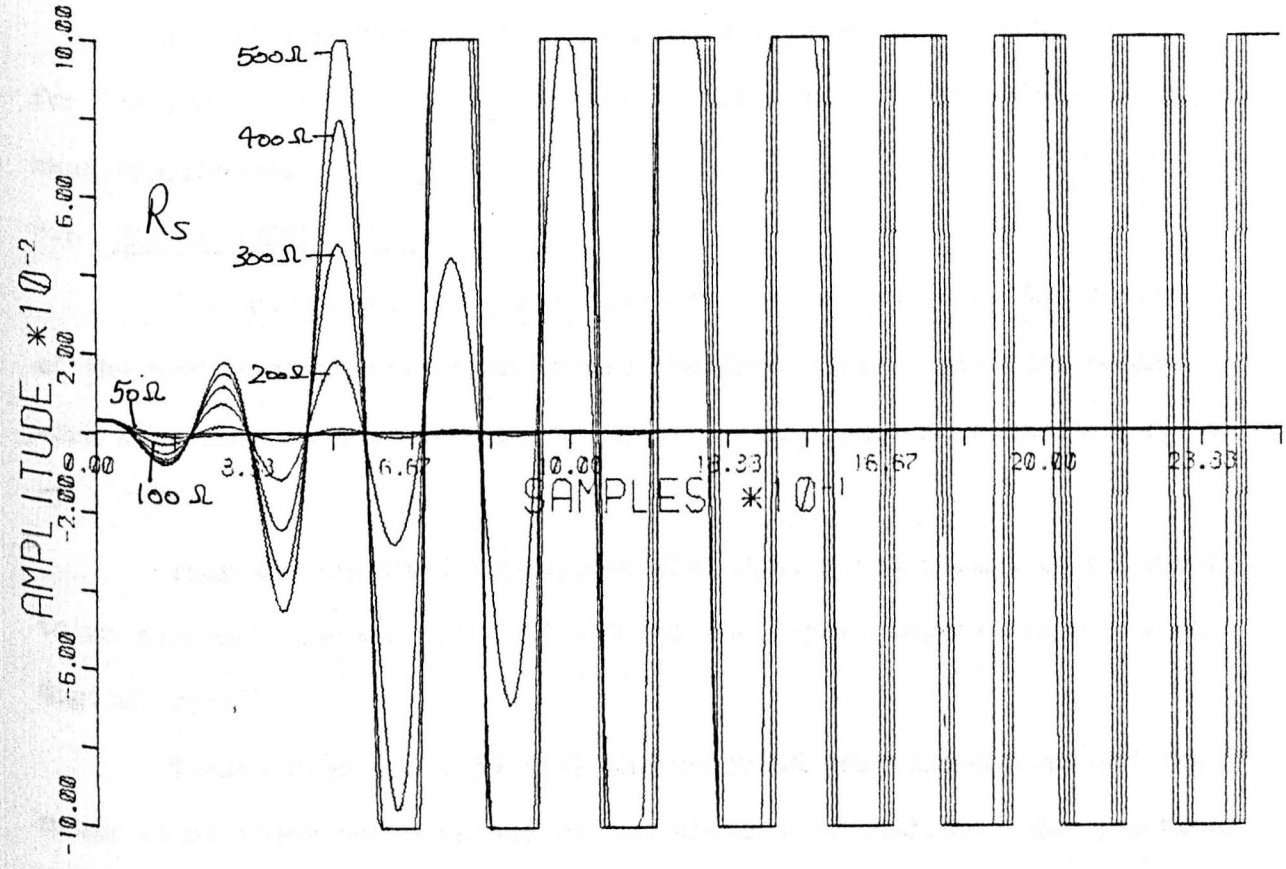
INVERSE TRANSFORM

Z21 - SAMPLE PERIOD = 150E-6 SECONDS

GRAPH 8.51



IMPULSE RESPONSE



INVERSE TRANSFORM

Z21 - SAMPLE PERIOD = 300E-6 SECONDS

GRAPH 8.52

this was adjusted for each of the five sample rates according to Table 8.37. The iteration size was reduced with increasing sample period in order to keep the plotting size of the oscillations identical for different sample periods. The frequency of oscillation observable in these responses can be calculated as follows:

$$f_{osc} = \frac{1}{T_s} \cdot \frac{1}{n} = f_s / n \qquad (8.3)$$

where T_s is the sample period and n is the number of iterations in one oscillation cycle.

It can be observed from the Graphs 8.43, 8.48, 8,49 and 8.50 that due to the large number of iterations needed by the algorithm that rounding errors have occurred in the impulse response causing the calculated values to drift from the origin. This is particularly noticeable when R_S = 300, 400 and 500 ohms. The reduced number of iterations in the other graphs have masked this effect.

The inverse Z-transform and impulse responses are not shown for the admittance matrix due to the simple form of polynomials making response finite.

8.6 ELEMENT RESOLUTION

The matrix element resolution has been shown to be the inverse of the number of quantisation levels (Section 3.12). With the 5-bit word size used in the experimental machine this gave an element resolution of 3.23%.

Thus the computed impedances and admittances cannot be measured to an accuracy greater than ± 3.23% for the 2-port capacitively loaded digital gyrator.

Tables 8.38 and 8.39 list the computed peak impedances and the range about these peaks caused by the element resolution. The practical results, after suitable correction, are expected to lie somewhere in

SAMPLE PERIOD T_s μs	NO. OF ITERATIONS IN INVERSE TRANSFORM
18.75	4000
37.5	2000
75.0	1000
150.0	500
300.0	250

TABLE 8.37

Inverse Transform Iteration Size.

SHUNT RESISTANCE	COMPUTED PEAK IMPEDANCE	COMPUTED PEAK IMPEDANCE RANGE (± 3.23%)	
		MIN	MAX
Ω	Ω	Ω	Ω
50	52.46	50.77	54.15
100	107.01	103.55	110.56
200	226.51	219.99	233.83
300	362.15	350.45	373.85
400	517.47	500.76	534.18
500	697.30	674.78	719.82

TABLE 8.38

Computed Peak Impedance Range for z_{11}

SHUNT RESISTANCE	COMPUTED PEAK IMPEDANCE	COMPUTED PEAK IMPEDANCE RANGE (± 3.23%)	
		MIN	MAX
Ω	Ω	Ω	Ω
50	No peak	-	-
100	No peak	-	-
200	316.23	220.61	235.33
300	524.81	347.00	370.16
400	822.24	493.25	526.17
500	1678.80	662.90	707.16

TABLE 8.39

Computed Peak Impedance Range for z_{21}

this range.

8.7 SUMMARY AND CONCLUSIONS

The results presented in this chapter have calculated the pole and zero positions for the elements of both the digital impedance and admittance matrices for a range of test parameters for the capacitively loaded 2-port digital gyrator, calculating the frequency response, inverse Z-transform and impulse response of these same elements. More values of test parameters could have been chosen but this would have only produced an overwhelming quantity of results giving less and less extra information.

The quantisation transfer function has been evaluated for this gyrator and it has been shown that the digital active network should demonstrate quantisation oscillation when the chosen shunt resistance is 200 ohms or greater. The value of shunt resistance which just causes limit cycle oscillation to be sustained will lie between 100 and 200 ohms.

CHAPTER 9

COMPARISON OF RESULTS

9.1 INTRODUCTION

In this chapter, the results of the practical experiments with the capacitively loaded 2-port digital gyrator of Chapter 7 are compared with the computed results from Chapter 8 for the same circuit. The comparison of results is based entirely on the z_{11} and z_{21} elements of the digital impedance matrix (8.2), as those were the two elements investigated practically. Furthermore, only the frequency responses are used for this comparison as it is impossible in practice to measure the inverse Z-transform or impulse responses for these elements.

The two elements from the digital impedance matrix were chosen for comparison because the digital admittance matrix was derived in Chapter 4 and the digital impedance matrix then calculated by straightforward matrix inversion. As each element of the impedance matrix depends on all four admittance matrix elements it follows that a comparison through the impedance matrix will verify all elements of the admittance matrix. Further, if all those admittance elements are correct then the other two impedance elements will also be correct.

The results used for comparison were obtained using the six values of input shunt resistance listed in Table 8.31, but with only one value of sampling period, 18.75 microseconds.

9.2 DIGITAL INPUT IMPEDANCE (z_{11})

The frequency response for the digital input impedance $z_{11}(z)$ is shown for the practical results in Graphs 7.1 to 7.6 and for the computed results in Graph 8.8. These both show resonance at nominally the same frequency, but the impedance magnitude is not equal.

Table 9.1 shows the corrected and computed peak impedance

Shunt Resistance Ω	Measured Input Impedance Ω	Corrected Measured Input Impedance Range		Computed Input Impedance Range	
		Min. Ω	Max. Ω	Min. Ω	Max. Ω
50	53.70	49.23	58.18	50.77	54.15
100	118.85	108.96	128.74	103.55	110.56
200	269.15	201.11	237.61	219.99	233.83
300	512.86	240.81	284.52	350.45	373.85
400	831.76	435.42	514.45	500.76	534.18
500	1513.56	-	-	674.78	719.82

Table 9.1 Corrected and Computed Peak Impedances for z_{11}

ranges for the six values of shunt resistance with the sampling period set to 18.75 microseconds.

It can be seen from these results that the corrected impedance range overlaps the computed impedance range when the shunt resistance was set to 50, 100, 200 and 400 ohms. Table 9.1 shows discrepancies between the corrected and expected impedances when the measured signal voltage in Table 7.19 was of a similar magnitude to the limit cycle noise voltage.

If the true effect of this limit cycle noise had been predicted before the practical experiment then clearly this problem could have been drastically reduced by deliberately increasing the input signal to the digital gyrator to the maximum level that could be linearly handled.

9.3 DIGITAL FORWARD TRANSIMPEDANCE (z_{21})

The frequency response for the digital forward transfer impedance $z_{21}(z)$ is shown for the practical results in Graphs 7.7 to 7.12 and for the computed results in Graph 8.18. These both show a finite response at D.C. and resonance at nominally the same frequency, but the impedance magnitude is again not equal.

Table 9.2 shows the corrected and computer peak impedance ranges for the six values of shunt resistance with the sampling set to 18.75 microseconds.

It can be seen from these results that the corrected impedance range overlaps the computed impedance range only when the shunt resistance was set to 400 ohms. Table 9.2 shows discrepancies between the corrected and computed impedance ranges when the measured signal voltage in Table 7.20 was of a similar magnitude to the limit cycle noise voltage.

As with the measurement of z_{11}, if the effect of this limit

Shunt Resistance Ω	Measured Forward Transimpedance Ω	Corrected Measured Forward Transimpedance Range		Computed Forward Transimpedance Range	
		Min. Ω	Max. Ω	Min. Ω	Max. Ω
50	No peak	-	-	-	-
100	No peak	-	-	-	-
200	316.23	175.11	206.89	220.61	235.33
300	524.81	283.40	334.83	347.00	370.16
400	822.24	461.34	545.08	493.25	526.17
500	1678.80	255.49	301.87	662.90	707.16

Table 9.2 Corrected and Computed Peak Impedances for z_{21}

cycle noise had been predicted then the input signal to the digital gyrator could have been increased to maximise the signal-to-limit cycle noise ratio.

9.4 LIMIT CYCLE NOISE VOLTAGES

The limit cycle noise voltages measured from photographs are listed in Table 7.18, and the computed voltages are listed in Tables 8.35 and 8.36 for shunt resistances of 200 and 300 ohms. It is clear from these results that the predicted noise voltages are significantly greater than the measured voltages.

However, as the basic process of limit cycle noise generation is a non-linear process, it is very difficult to derive an accurate expression to allow the limit cycle noise voltage to be calculated. However it may be that the phase response of the quantisation voltage transfer function plays a significant part. At resonance at port 1 this phase shift is independent of the shunt resistance, as can be seen in Graph 8.33, with a magnitude of about 41^{o}. At port 2 this shift is about 180^{o}. Thus if this phase shift is involved in the expression for the limit cycle noise voltage then a different relationship between the photographic results and the computed results would be expected. From Tables 7.18, 8.35 and 8.36 the results for port 2 are much closer than port 1.

However no definite relationship can be established with only two results for each port.

9.5 SUMMARY AND CONCLUSIONS

The measured digital input impedance and forward transfer impedance have been shown to be approximately equal to the equivalent computed impedances when the measurement errors, limit cycle noise and element resolution have been taken into account. However the effects of these three types of errors has so reduced the accuracy of the

measured results that the practical experiment cannot be said to be absolutely conclusive with the values obtained.

Clearly the experiment could have been repeated with the machine as described in Chapter 6, but a better solution would have been to have rebuilt the machine with at least 8 bits in the internal digital word. This would have immediately increased the signal to limit cycle noise ratio and the element resolution eight-fold.

Having extensively demonstrated the analysis of digital active networks, the way is now open to consider the synthesis of digital active networks.

CHAPTER 10

DIGITAL NETWORK SYNTHESIS

10.1 INTRODUCTION

The purpose of this chapter is to show the advantages and limitations of synthesising digital active networks.

10.2 GENERAL SYNTHESIS TECHNIQUES

In order to synthesisedigital active networks it is simplest to use the techniques available to synthesis analogue active networks and then convert the resulting network to contain digital amplifiers. This may be done by taking the Z-transform of the synthesised admittance matrix and realising the resulting digital admittance matrix using digital amplifiers.

One starting point in the synthesis of analogue active networks is to specify an external set of properties, such as transfer functions or input impedances, and from these to generate a circuit realisation. An alternative starting point for synthesis which is often used is an N x N admittance matrix describing the external properties at the N ports of the network. If the circuit is derived by some other technique then the admittance matrix may still be found by conventional analysis techniques.

If the resulting admittance matrix is symmetrical then the network can be realised using only passive components, and as such would not be sensible to synthesise with digital amplifiers. However the network will be passive in a general sense if there is no net power gain, which can be the case with positive or negative impedance converters or inverters which do require internal amplifiers, and are conveniently suited to conversion to use digital amplifiers.

The 2-port capacitively loaded digital gyrator analysed in Chapter

4 would have been passive if the off-diagonal elements in the digital admittance matrix (4.13) had been equal in magnitude at all frequencies.

The admittance matrix may be divided into a symmetrical (or passive) matrix and an asymmetrical (or active) matrix thus:

$$Y(s) = Y_p(s) + Y_A(s) \tag{10.1}$$

where by definition:

$$Y_p(s) = Y_p^T(s) \tag{10.2}$$

Various techniques have been proposed to enable an active admittance matrix to be synthesised from the original admittance matrix, such as those by Yanagisawa and Kanbayashi [3] [4] and also by Thielmann [31].

10.3 DIGITAL ADMITTANCE MATRICES

Before digital admplifiers can be introduced into the synthesis of digital admittances arrays it is necessary to investigate further the implications of taking the Z-transform of an admittance matrix. From the six case studies in Chapter 3 it can be seen that although by definition:

$$Y(z) = Z\{Y(s)\} \tag{10.3}$$

that this does not in general apply at an element level.

Cases 3 and 4 in Chapter 3 show that the rows in the admittance matrix with all transformable elements may be transformed directly thus:

$$y_{ij}(z) = Z\{y_{ij}(s)\} \tag{10.4}$$

If even one element in a row is untransformable then that row must be manipulated (through matrices) before transforming. If $y_{ij}(s)$ is untransformable then from Chapter 3 $y_{ij}(z)$ was shown to be:

$$y_{ij}(z) = \frac{1}{Z\left\{\frac{1}{y_{ij}(s)}\right\}} \tag{10.5}$$

All the other elements in that row are then transformed thus:

$$y_{ik}(z) = \frac{1}{Z\left\{\frac{1}{y_{ij}(s)}\right\}} \; Z\left\{\frac{y_{ik}(s)}{y_{ij}(s)}\right\} \tag{10.6}$$

Unfortunately equation (10.6) does not reduce any more and in particular y_{ij} in any form does not cancel unless y_{ik} is not a function of s. It is at this point that the synthesis in discrete time departs from the continuous time case.

A general element of the digital admittance matrix may be written thus:

$$y_{ik}(z) = Z\left\{ f\left(y_{i1}(s),\ y_{i2}(s),\ \ldots\ldots\ y_{iN}(s)\right)\right\} \tag{10.7}$$

Now when two analogue admittance networks with matrices $Y_1(s)$ and $Y_2(s)$ are connected node for node then by definition:

$$Y(s) = Y_1(s) + Y_2(s) \tag{10.8}$$

The Z-transform of the resulting analogue admittance matrix is therefore:

$$Y(z) = Z\left\{ Y_1(s) + Y_2(s)\right\} \tag{10.9}$$

If equation (10.4) applies to $Y_1(s)$ and $Y_2(s)$ completely then equation (10.9) may be rewritten:

$$Y(z) = Y_1(z) + Y_2(z) \tag{10.10}$$

However if equation (10.7) has to be applied then Y(z) in equation (10.9) becomes a complicated function of the transform of the sum of $Y_1(s)$ and $Y_2(s)$. This then means that unless equation (10.4) applies to the elements of the combined admittance matrix, that the elements in some of the rows of the admittance matrix will be considerably modified when an external admittance network is connected. Thus the

range of networks which may be usefully synthesised is restricted to those networks which are fully transformable.

10.4 SELECTION OF Z-TRANSFORM

In order to apply digital amplifiers to the synthesis of active networks it is necessary to consider the type of Z-transform of the s-plane transfer functions which is to be used.

In Chapter 3, six different combinations of transformable and untransformable admittance matrix elements were considered and it was shown that the 'standard' Z-transform [6] could always be taken and the digital admittance matrix therefore always found. In the case of synthesis, the 'bilinear' Z-transform [7] [8] or the 'matched' Z-transform [9] may be used as well, whereas in the analysis case the 'standard' Z-transform had to be used in order to exactly calculate the digital admittance or impedance matrices ready for a comparison with the results from the practical experiments.

However with synthesis the choice must be made according to the properties of the original analogue transfer function which are to be most closely reproduced.

10.4.1 Standard Z-transform

The standard Z-transform is described in Appendix A, and it is sufficient to quote two simple results:

$$\frac{1}{s} \Rightarrow \frac{z}{z-1} \tag{10.11}$$

and in the case of digital active networks it was shown in Chapter 2 that

$$s \Rightarrow \frac{z-1}{z} \tag{10.12}$$

In this transform the impulse response is invariant, whereas the frequency response and pole and zero locations may differ considerably.

10.4.2 Bilinear Z-transform

The bilinear Z-transform copies the frequency response of the s-plane network much more closely. Here:

$$s \Longrightarrow \frac{z - 1}{z + 1} \qquad (10.13)$$

This may be compared with equation (10.12), and it can be seen that the pole has migrated from z = 0 to z = -1, which corresponds to the nyquist frequency.

Thus this transform has normalised the transfer function response at infinite frequency to the nyquist rate.

The impulse response over the transform is now no longer invariant, and the pole and zero locations do not match.

10.4.3 Matched Z-transform

The matched Z-transform is intended to match the pole and zero locations of the discrete time network to that of the original untransformed network.

This may be simply achieved thus:

$$s \Longrightarrow z \qquad (10.14)$$

Again, the impulse response is no longer invariant, and the frequency response will also be different.

Once the Z-transform has been taken then the structure for the required digital filter can be derived by rearranging the pulse transfer function as a linear difference equation from which a digital filter structure may be readily obtained.

The synthesis of the convolving elements in the digital filters has now reached the standard digital filter design techniques such as described in [5], [7], [8], [9] and [10].

10.5 APPLICATION OF DIGITAL AMPLIFIERS

Digital amplifiers may be applied in two ways in network synthesis;

either a direct substitution of a digital amplifer for an analogue amplifier may be made, or an array of digital admittance amplifiers used where their arrangement is derived from the asymmetric admittance matrix.

The former use of digital amplifiers does not exploit their ability to have digital filters included in their signal path whereas in the latter case the digital amplifier may be designed to conveniently simulate a wide variety of complicated transfer functions, and it is to this role that the digital amplifier is best suited.

One of the main advantages of using digital amplifiers in network synthesis is that their internal digital filters can easily be designed to have a linear phase response by using a transversal filter with symmetrical tapping weights which will give a finite impulse response (FIR).

Now the transfer function of a standard digital amplifier was derived in equation (2.50). Due to the presence of the denominator factor 's', it is necessary to introduce the digital amplifier transfer function before taking the Z-transform because:

$$Z = \left\{ \left(\frac{1 - e^{-sT_s}}{s} \right) e^{-skT_s} H(s) \right\}$$

$$\neq z^{-k} H(z) \qquad (10.15)$$

Thus the digital amplifier position has to be defined before the Z-transform is taken.

10.6 LIMIT CYCLE NOISE

It is necessary to test the digital admittance matrix derived during digital active network synthesis to check if limit cycle oscillations can occur in order to prevent a wasted implementation. The technique was fully explained in section 3.11. However from equation (3.72) the noise vector at the ports of the digital amplifier

array will be:

$$V_Q(z) = Z(z)\ Y_1(z)\ V_N(z) \tag{10.16}$$

where $Y_1(z)$ is the digital admittance matrix of all the noise generating elements. If all the elements are simulated using digital amplifiers which all introduce quantisation noise then:

$$Y_1(z) = Y(z) \tag{10.17}$$

and equation (10.16) reduces by definition to:

$$V_Q(z) = V_N(z) \tag{10.18}$$

Thus A (z) in equation (3.73) becomes:

$$A(z) = U \tag{10.19}$$

where U is the identity matrix.

Now it is very often the case that when equation (10.17) is not true that the elements of A(z) are considerably greater or equal to two at certain frequencies. By definition the identity matrix has unity elements and thus a digital active network using digital amplifiers for every element cannot show limit cycle oscillations, merely simple quantisation noise. Thus an all-digital amplifier realisation is very preferable.

However the lack of limit cycle oscillations due to the quantisation process in an all-digital amplifier realisation does not mean that limit cycle oscillations cannot occur due to truncation of word lengths within each digital amplifier. This effect is described in such papers as [13], [14], [15], [16], [17].

10.7 DIGITAL GYRATOR SYNTHESIS

The 2-port capacitively loaded digital gyrator which was studied for analysis purposes in Chapter 4 may also be studied as a synthesis example. The circuit is shown in Fig. 4.1 and the analysis was based on equation (3.31) derived in case 3 in Chapter 3.

An analogue 2-port capacitively loaded gyrator (Fig. 10.1) will be described by an admittance matrix thus:

$$Y(s) = \begin{bmatrix} g + s\,C_1 & g_1 \\ -g_2 & sC_2 \end{bmatrix} \qquad (10.20)$$

The equivalent digital gyrator was formed in Chapter 4 by replacing the analogue cross-coupled transconductance amplifiers by digital amplifiers.

After replacing the off-diagonal terms in Y(s) by the transfer function of digital amplifier, and then taking the Z-transform according to case 3 in Chapter 3, Y(z) was found using the standard Z-transform:

$$Y(z) = \begin{bmatrix} \frac{C_1}{T_s}\left(\frac{z-\alpha}{z}\right) & \frac{g_1 C_1}{gT_s} z^{-k_1-1}(1-\alpha) \\ -g_2 z^{-k_2-1} & \frac{C_2}{T_s}\left(\frac{z-1}{z}\right) \end{bmatrix} \qquad (10.21)$$

When using digital amplifiers to simulate the leading diagonal elements as well as the off-diagonal elements, it is necessary to include the effects of the zero order hold stage and the amplifier delay, as in equation (2.48). Thus matrix (10.20) becomes:

$$Y(s) = \begin{bmatrix} \frac{(g+sC_1)(1-e^{-sT_s})}{s} e^{-sk_{11}T_s} & g_1\frac{(1-e^{-sT_s})}{s} e^{-sk_{12}T_s} \\ -g_2\frac{(1-e^{-sT_s})}{s} e^{-sk_{21}T_s} & sC_2\frac{(1-e^{-sT_s})}{s} e^{-sk_{22}T_s} \end{bmatrix} \qquad (10.22)$$

which may be transformed to:

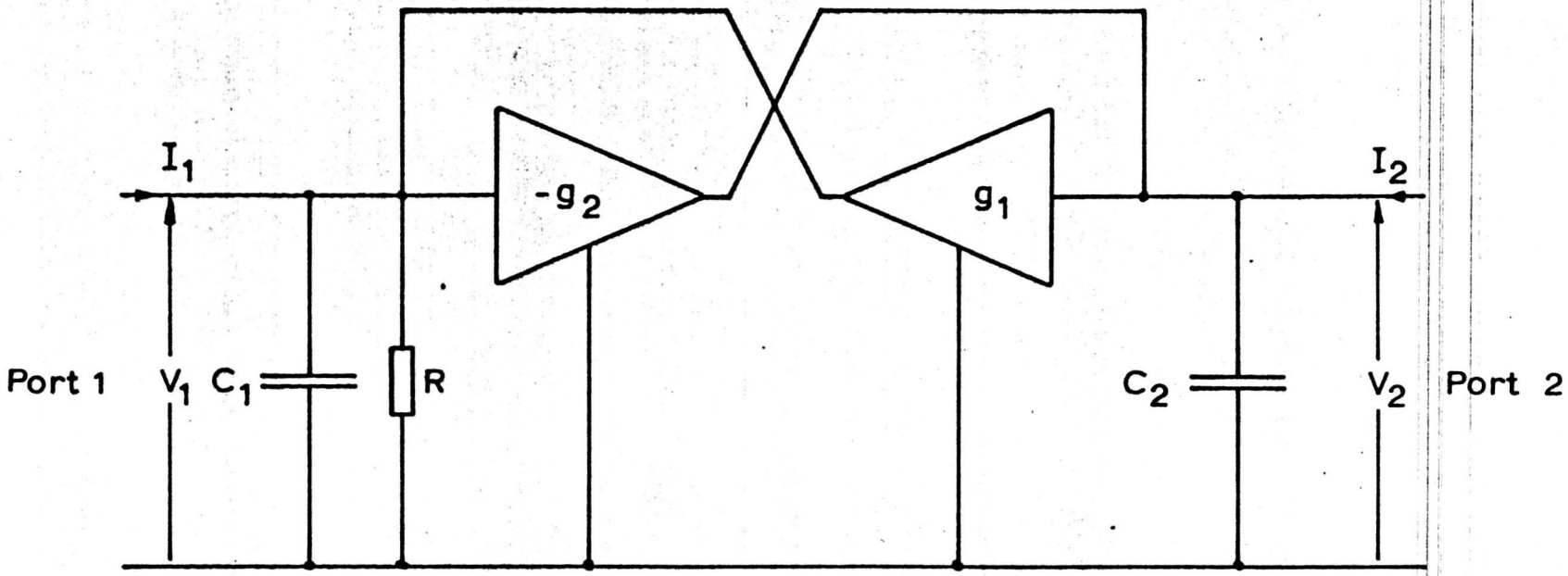


FIG.10.1 Capacitively Loaded Analogue Gyrator

$$Y(z) = \begin{bmatrix} z^{-k_{11}} \left(g + \frac{C_1}{T_s}\left(\frac{z-1}{z}\right)\right) & g_1 z^{-k_{12}} \\ -g_2 z^{-k_{21}} & \frac{C_2}{T_s}\left(\frac{z-1}{z}\right) z^{-k_{22}} \end{bmatrix} \qquad (10.23)$$

The elements in the above matrix contain only one mention of each of the original components, g, g_1, g_2, C_1 and C_2. Thus the realisation of this matrix will be able to have each component controlled independently as with the original analogue network.

It should also be noted that compared with matrix (10.21), α is no longer present and all the elements are simplified. Furthermore the capacitance terms in $y_{11}(z)$ and $y_{22}(z)$ can be clearly identified.

In effect matrix (10.23) is showing that the digital components developed and described in section 2.10 may be combined to form an all-digital gyrator.

The leading diagonal terms of matrix (10.23) may now be simulated using digital amplifiers with input and output strapped and with simple feed forward digital amplifiers in their signal path. The off-diagonal terms can be simulated using simple digital amplifiers. The resulting digital amplifier array is shown in Fig. 10.2. The duplication of A/D and D/A converters has been removed.

10.8 CONCLUSIONS

It has been shown that it is possible to synthesise digital active networks using digital amplifiers, and that for two reasons it is particularly beneficial to use an all-digital amplifier approach. Firstly, an all digital approach eliminates the problem of limit cycle oscillation caused by quantisation noise. Secondly the practical

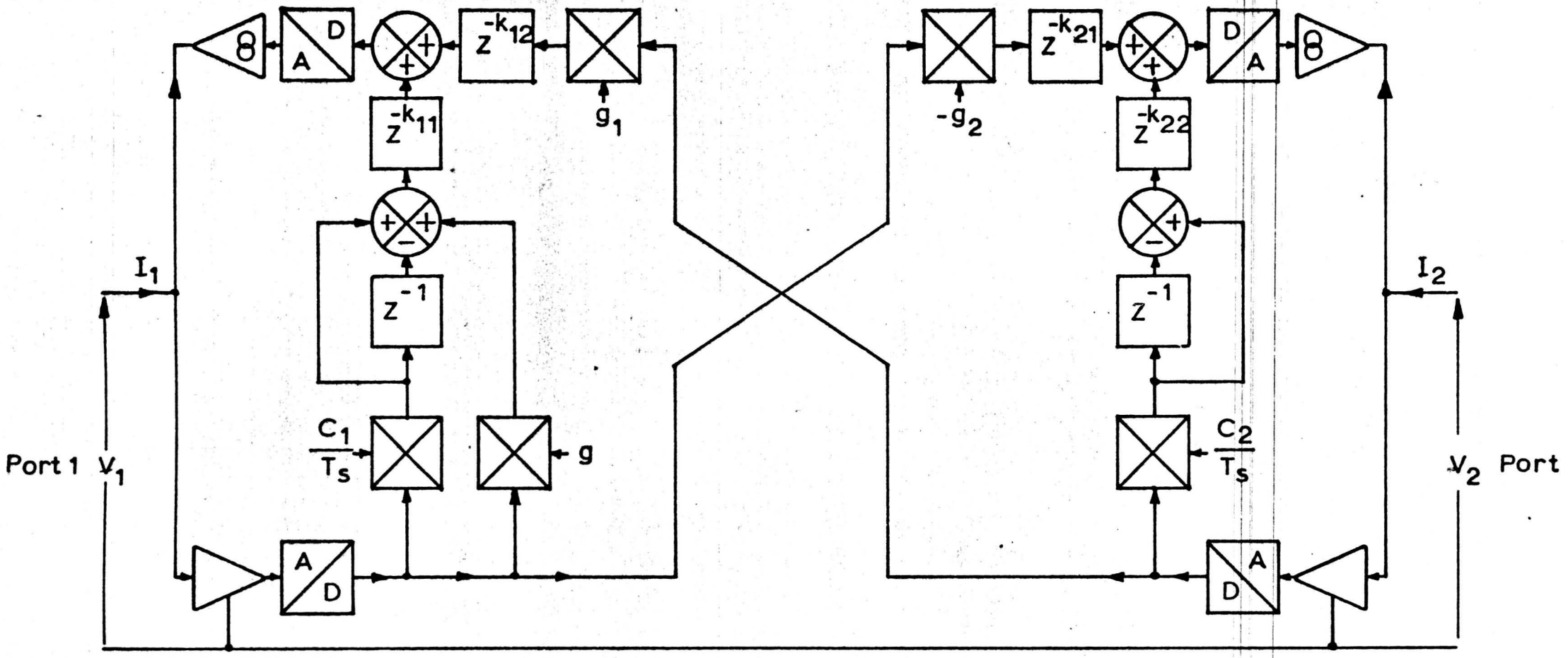


FIG. 10.2 All-Digital Gyrator

example showed that the all-digital approach produces realisable transformed admittance matrix elements which are easily related to the original analogue components, thereby allowing simple control of each digital component value.

CHAPTER 11

SUMMARY AND CONCLUSIONS

11.1 INTRODUCTION

The purpose of this thesis has been to investigate the analysis and synthesis of digital active networks. This has been achieved both in theory and practice.

11.2 SUMMARY

The term 'digital active network' has been defined to mean an array consisting either partly or wholly of digital amplifiers, which in turn have been defined to contain a current or voltage sensor, an A/D converter, a scaler, a convolver, a D/A converter and a current or voltage generator.

It was shown that for practical reasons it is much easier to design digital transadmittance amplifiers than any other type, and that digital admittance arrays were therefore the simplest to simulate.

The conservation of units under the sampling process was considered and it was shown that it was necessary to introduce the sampling period as a correction factor.

Next the conditions under which a Laplace domain transfer function could be Z-transformed were considered and it was shown that it was valid to use the reciprocal of the Z-transform of the reciprocal of a transfer function if that transfer function was not transformable.

The concept of a digital circuit component was then introduced and its construction by coupling the output and input of a digital transadmittance amplifier described. A few simple digital component simulations were then considered.

The next important step was to develop techniques to analyse mixed arrays of digital amplifiers and analogue circuit components.

Six typical case studies were investigated and equations derived to allow the digital admittance and impedance matrices to be found for each case.

From a knowledge of the digital admittance matrix for a digital active network a quantisation matrix was calculated and it was shown that the introduction of amplitude quantisation in each A/D converter would lead to the generation of limit cycle oscillations under certain conditions, which were in turn derived. It was shown that the amplitude of these limit cycle oscillations could be obtained by evaluating the frequency response of the elements of a quantisation matrix, and from the quantisation step size.

A secondary effect of amplitude quantisation, namely element resolution, was then investigated and shown to be the reciprocal of the number of quantisation levels used.

The matrix analysis was then applied to a capacitively loaded 2-port digital gyrator, and the digital admittance and impedance matrices found. From these matrices it was predicted that limit cycle oscillations would be present using certain analogue component values.

This analysis also formed the basis for the construction of a capacitively loaded 2-port digital gyrator and for the writing of a suite of FORTRAN computer programs to evaluate these matrices. Limit cycle oscillations were shown to exist with certain component values. Results obtained from the practical machine and the computer results were compared and after correction for limit cycle oscillations, a reasonable but not close match was found.

Finally the problems and advantages behind the synthesis of digital active networks were investigated, and it was shown that it was preferable to design an all-digital network. This was because an all-digital network could not show limit cycle oscillation caused

by amplitude quantisation, and furthermore the elements of the admittance matrix of an all-digital realisation were simple transforms of the original component types, whereas the elements of the admittance matrix derived from a mixed analogue and digital realisation were not the simple transform of each element.

11.3 CONCLUSIONS

From the work presented in this thesis it is clear that analogue active and passive networks may be combined with sampled data systems and digital filters, and that the resulting digital active networks may be both analysed and synthesised. Thus digital active networks are feasible.

The advantages of digital active networks really lie in being able to use digital filters in place of analogue filters, rather than restricting digital filters to the role of simple signal path processing.

Digital active networks intrinsically contain complicated circuitry but whereas that presented a problem at the time when the practical work for this thesis was started, it is no longer as great a problem with the increase in the number of types of integrated circuits. However digital active networks will always cost more per network node than their analogue near equivalents.

REFERENCES

1 PIM D. N. and BULLINGHAM J. M.
"Digital Simulation of Electronic Components"
Electronics Letters. 4 Sept 1975 Vol 11 No 18
pp 441 - 442

2 PIM D. N. and BULLINGHAM J. M.
"Digital Inductors and Capacitors"
Electronics Letters. 19 Aug 1976 Vol 12 No 17
pp 425 - 427

3 YANAGISAWA T. and KANBAYASHI N.
"Realisation of Arbitary Conductance Matrix using Operational Amplifiers"
IEEE Proceedings of International Symposium on Circuits and Systems 1976
pp 532 - 535

4 YANAGISAWA T. and KANBAYASHI N.
"Multi-Port Multi- Directional Amplifier using Operational Amplifiers"
IEEE Proceedings of International Symposium on Circuits and Systems 1978
pp 866 - 870

5 GOLDEN R. M.
"Digital Filter synthesis by sampled-data Transformation"
IEEE Transactions on Audio and Electroacoustics
Sept 1968 Vol AU - 16 No 3
pp 321 - 329

6 JURY E. I.
'Theory and Application of the Z-transform Method'
Wiley 1964

7 GOLDEN R. M. and KAISER J. F.
"Design of wideband sampled-data filters"
Bell System Technical Journal. July 1964 Vol 43 Part 2
pp 1533 - 1546

8 LEWIS C. P. and MATHEW M. I.
"The design of digital filters in the w-plane"
Electronic Engineering. Aug 1974
pp 12 - 15

9 KELLY L. C. and HOLMES J. N.
"Computer processing of signals with particular reference to simulation of electric filter networks"
Joint Speech Research Unit, Ruislip, Middlesex
Research Report JU14 - 1 Nov 1964

10 RADER C. M. and GOLD B.
"Digital filter design techniques in the frequency domain"
Proc IEEE Vol 55 No 2 Feb 1967
pp 149 - 171

11 SHINNERS S. M.
'Control System Design'
Wiley International 1964

12 TOU J. T.
'Digital Sampled Data Control Systems'
McGraw-Hill 1959 pp 572 - 576

13 OPPENHEIM A. V. and WERNSTEIN C. J.
"Effects of finite register length in digital filtering and the fast fourier transform"
Proc. IEEE Vol 60 No 8 Aug 1972
pp 957 - 976

14 AVENHAUS E.
"On the design of digital filters with coefficients of limited word length"
IEEE Transactions on Audio and Electroacoustics
Vol AU - 20 No 3 Aug 1972
pp 206 - 212

15 JACKSON L. B.
"Roundoff-noise analysis for fixed point digital filters realised in cascade or parallel form"
IEEE Transactions on Audio and Electroacoustics
Vol AU - 18 No 2 June 1970
pp 107 - 122

16 SANDBERG I. W. and KAISER J. F.
"A bound on limit-cycles in fixed point implementations of digital filters"
IEEE Transactions on Audio and Acoustics
Vol AU - 20 No 2 June 1972
pp 110 - 112

17 WHITING P. A.
"Finite word length effects in digital processes"
Electron Nov 1974
pp 46 - 51

18 BLACKMAN R. B.
'Data Smoothing and Prediction'
Addison-Wesley 1965
p 83

19 GEARY A., LOWRY H. V. and HAYDEN H. A.
'Advanced Mathematics for Technical Students. Part 1'
Longmans 1945

20 BETTS J. A.
'Signal Processing, Modulation and Noise'
English Universities Press 1970
p 139

21 ISAACSON E. and KELLER H. B.
'Analysis of Numerical Methods'
Wiley. New York. 1966
pp 131 - 133

22 'Semiconductor Components Data Book Three - Transistors'
Texas Instruments 1971
pp 122 - 129

23 'Consumer Electronics Handbook Part 1'
Mullard 1970 - 71
pp BD 131 Page 1 - BD 132 Page 8

24 'Semiconductor Components Data Book Two - Digital Integrated Circuits'
Texas Instruments 1971

25 'D/A and A/D Short Form Catalogue'
Micro Networks Corporation May 1974

26 'Data Sheet for Am 2502, Am 2503 and Am 2504. Successive approximations registers'
Advanced Micro Devices Inc.

27 'Data Acquisition Products Catalog'
Analog Devices Inc 1978
pp 381 - 472

28 'Data Sheet for Am 2505 Four-bit by Two-bit 2's complement multiplier'
Advanced Micro Devices Inc.

29 SNEDDON I. N.
'Fourier Series'
Routledge & Kegan Paul Ltd. 1961

30 'Math Pac 1 for HP - 65 Calculator'
Hewlett-Packard
p 22 (Math 1 - 08A)

31 THIELMANN H.
"Synthese linearer aktiver RC - Netzwerke durch Matrixerweiterung"
AEU Band 28 (1974) Heft 6
pp 245 - 256

APPENDIX A

BASIC SAMPLING THEORY

A.1 THE Z-TRANSFORM

The basic action of sampling is the multiplication in the time domain of a signal x(t) with a unit impulse train i(t). These signals can be seen in Fig. A.1. The unit impulse function has a period of T_s starting at t=0. Hence i(t) may be written:

$$i(t) = \sum_{r=0}^{\infty} \delta (t - rT_s) \qquad (A.1)$$

where $\delta (t - rT_s)$ is a unit impulse occurring when $t = rT_s$. Further the impulses themselves are assumed to have unit area and infinite height coupled with infinitesimal duration.

Let x(t) be any arbitrary waveform and $x^*(t)$ be defined as the result of sampling x(t), then:

$$x^*(t) = x(t).\, i(t) \qquad (A.2)$$

From equation (A.1), (A.2) may be rewritten:

$$x^*(t) = \sum_{r=0}^{\infty} x(rT_s)\ \delta (t - rT_s) \qquad (A.3)$$

This equation may now be Laplace transformed:

$$\mathcal{L}\left\{x^*(t)\right\} = \sum_{r=0}^{\infty} x(rT_s) \int_0^{\infty} \delta(t - rT_s) e^{-st}\, dt$$

$$= \sum_{r=0}^{\infty} x(rT_s)\ e^{-rsT_s} = X^*(s) \qquad (A.4)$$

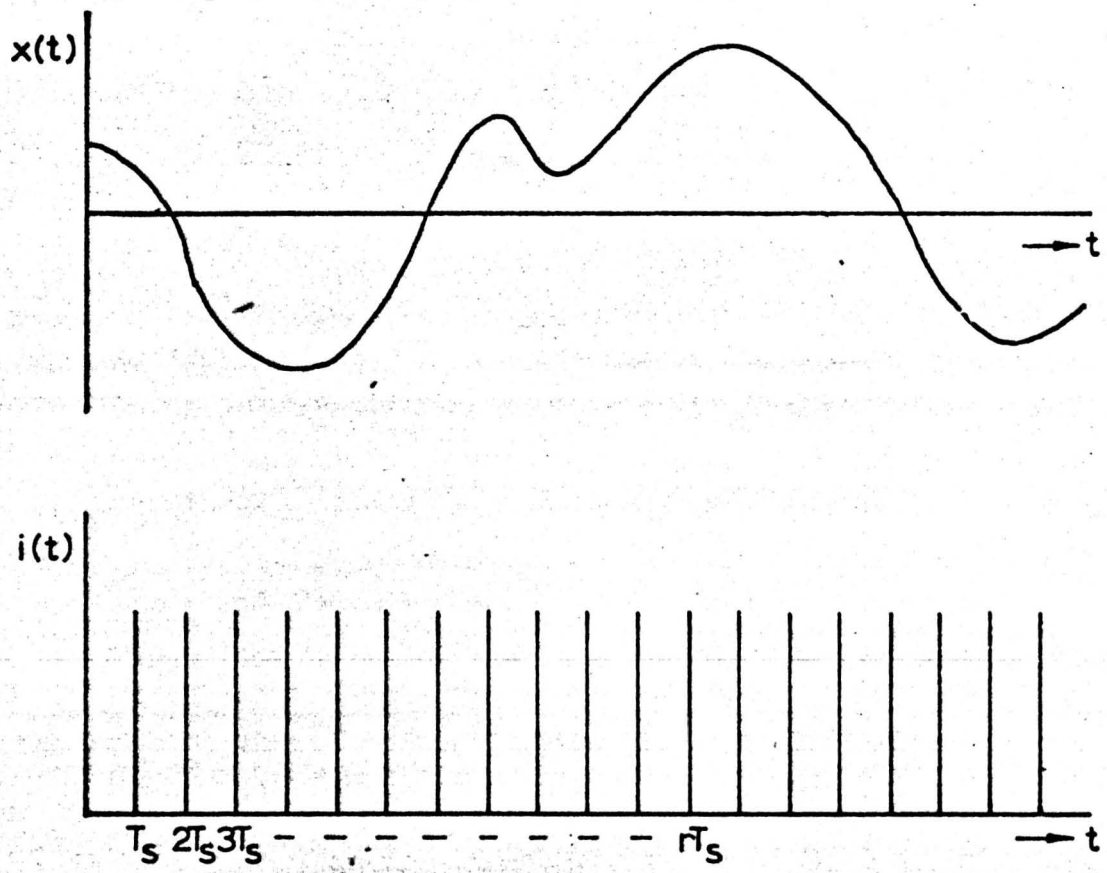


FIG. A.1 Signal and Sampling Waveforms

Equation (A.4) allows the time domain signal x(t) to be written as a sampled Laplace transform.

However other results can also be obtained. Consider the Complex Fourier Analysis of i(t) [29] :

$$i(t) = \frac{1}{T_s} \sum_{r=0}^{\infty} C_r \; e^{jrw_s t} \tag{A.5}$$

$$\text{where } C_r = \int_{\frac{-T_s}{2}}^{\frac{T_s}{2}} i(t) \; e^{-jrw_s t} \, dt \tag{A.6}$$

By substituting (A.6) in (A.5) it can be shown that :

$$i(t) = \frac{1}{T_s} \sum_{r=-\infty}^{\infty} e^{jrw_s t} \tag{A.7}$$

It is important to notice that the term $\frac{1}{T_s}$ has appeared, giving i(t) the units of Time $^{-1}$. Substituting (A.7) in (A.2) gives:

$$x^*(t) = \frac{1}{T_s} \sum_{r=-\infty}^{\infty} x(t) \; e^{jrw_s t}$$

and hence the Laplace transform becomes :

$$\mathcal{L}\left\{ x^*(t) \right\} = \frac{1}{T_s} \sum_{r=-\infty}^{\infty} \int_{0}^{\infty} x(t) \; e^{(s+jrw_s)t} \, dt$$

$$X^*(s) = \frac{1}{T_s} \sum_{r=-\infty}^{\infty} X(s + jrw_s) \tag{A.8}$$

Equation (A.8) gives an expression for the sampling of a Laplace transformed variable x(s).

Now the units of s are radians sec^{-1}. X(s) will be a polynomial with units which are an integer power of (radians sec^{-1}). The term $\frac{1}{T_s}$ will reduce this integer power by one. Hence when applying (A.8) to any equation in s, it is important to consider the conservation of units.

From equation (A.8), $X^*(s)$ may be replaced by X(z) (where $z = \exp(sT_s)$) provided that the series for $X^*(s)$ is convergent and this is called the Z-transform. Table A.1 shows some standard s and z plane transforms with their equivalent time domain functions.

In the practical case no real sampler can be ideal and in this case the previous analysis must be repeated. However, as shown in reference [18] if even a non-ideal sampler is followed by a holding capacitor, then the transfer function is not affected.

A.2 INVERSE Z-TRANSFORM

There are three techniques for taking the inverse Z-transform, and these are discussed in the succeeding sub-sections.

A.2.1 Partial Fraction Expansion

The function of z to be inverted must first have its denominator factorised so that the partial fraction expansion may be found. Each term in the partial fraction expansion is then compared with a list of standard transforms such as Table A.1.

The advantage of this method is that the function of z may be inverted to the s^* (sampled s) plane or the time domain.

A.2.2 Power Series Expansion

The function of z to be inverted is divided out thus:

TIME FUNCTION	LAPLACE TRANSFORM	Z-TRANSFORM
$\delta(t)$	1	1
$\delta(t-nT)$	e^{-snT}	z^{-n}
$U(t)$	$\frac{1}{s}$	$\frac{z}{z-1}$
t	$\frac{1}{s^2}$	$\frac{Tz}{(z-1)^2}$
$\exp(-at)$	$\frac{1}{s+a}$	$\frac{z}{z-\exp(-aT)}$
$t\exp(-at)$	$\frac{1}{(s+a)^2}$	$\frac{Tz\exp(-aT)}{(z-\exp(-aT))^2}$
$\text{Sin } a\,t$	$\frac{a}{s^2+a^2}$	$\frac{z \text{ Sin } a\,T}{z^2-2z \text{ Cos } a\,T+1}$
$\text{Cos } a\,t$	$\frac{s}{s^2+a^2}$	$\frac{z^2-z \text{ Cos } a\,T}{z^2-2z \text{ Cos } a\,T+1}$

TABLE A.1

Z - Transforms

$$F(z) = \frac{\sum_{i=0}^{n} a_i z^i}{\sum_{i=0}^{m} b_i z^i}$$

$$= \sum_{i=0}^{n-m} c_i z^i + \sum_{i=1}^{\infty} d_i z^{-i} \qquad (A.9)$$

The coefficients c_i and d_i then represent the time domain value when $t = i\,T$ because z^i is a pure time delay.

Thus the time domain value of the function at each sampling instant is available by this method.

A.2.3 The Inversion Integral

The inversion integral may be applied to the function of t to be inverted to give the time domain equivalent function :

$$f(nT) = \frac{1}{2\pi j} \oint F(z)\, z^{n-1}\, dz \qquad (A.10)$$

The line integral must be made large enough to include all the roots of F (z). The unit circle is normally used.

A.3 THE ADVANCED Z-TRANSFORM

For the ordinary Z-transform the results of sampling are only defined at the sampling instants which are by definition an integral multiple of the sampling period. The advanced Z-transform allows the computation of the results of sampling between the sampling instants. [11]

The advanced Z-transform has not been applied to any problems encountered in this thesis although the experimental machine's intermediate output state could be calculated using this technique.

APPENDIX B

BROWN'S METHOD FOR FACTORISING FOURTH ORDER POLYNOMIALS

B.1 INTRODUCTION

Brown's Method [30] was implemented as a computer algorithm in Chapter 5 to solve fourth order polynomials. This method requires the solving of an intermediate and associated third order polynomial from which the coefficients of a pair of simultaneous quadratic equations can be found. The roots of these simultaneous quadratic equations are also the roots of the fourth order polynomial.

B.2 ALGORITHM

Let the fourth order polynomial f(x) be:

$$f(x) = x^4 + a_3 x^3 + a_2 x^2 + a_1 x + a_o \qquad (B.1)$$

and define an associated third order polynomial f(y) such that:

$$f(y) = y^3 + b_2 y^2 + b_1 y + b_o \qquad (B.2)$$

where: $b_2 = - a_2$

$$b_1 = a_3 a_1 - 4 a_o$$

$$b_o = a_o (4 a_2 - a_3^2) - a_1^2$$

Let y_o be the largest real root of f(y). The coefficients of the simultaneous quadratic equations may then be defined:

$$x^2 + (A + C) x + B + D = 0 \qquad (B.3)$$

$$x^2 + (A - C) x + B - D = 0 \qquad (B.4)$$

where $A = a_3/2$

$$B = y_o/2$$

$$D = \sqrt{B^2 - a_o}$$

$$C = (A\,B - a_1/2)/D \qquad \text{if } D \neq 0$$

$$\text{or } C = \sqrt{A^2 - a_2 + y_o} \qquad \text{if } D = 0$$

equations (B.3) and (B.4) can be easily solved thus giving the four roots of equation (B.1)

APPENDIX C

A/D CONVERSION

C.1 A/D CONVERSION PROCESS

The process of linear A/D conversion involves deciding which one of M quantisation levels is nearest to the signal at an instant in time (the sampling instant). Hence in a binary A/D converter the signal is quantised to the level which is within $\pm \frac{1}{2}$ LSB of the signal.

For practical operation it is necessary to hold the sampled signal constant between sampling instants. These two operations are combined in a sample-and-hold stage.

The whole A/D conversion process is therefore equivalent to a sample-and-hold stage followed by a quantiser and is shown in Fig. C.1. The former stage is described in Appendix A and the latter stage in Section C.4. The full process of sampling and quantisation is shown graphically in Fig. C.2.

C.2 QUANTISATION PROCESS

The process of quantisation is used to decide which level of a code (normally binary) is nearest to the sampled signal. If the input can be both positive and negative and the binary code is used then the optimum choice for M is :

$$M = 2^n - 1 \qquad (C.1)$$

where n is the number of bits in the code, including the sign bit. Hence the size of any decision level will be :

$$\triangle_V = V / (2^n - 1) \qquad (C.2)$$

where V is the maximum absolute input signal.

From equation (C.2) the accuracy (A) in percent and dynamic range (R) in decibels of the converter can be found :

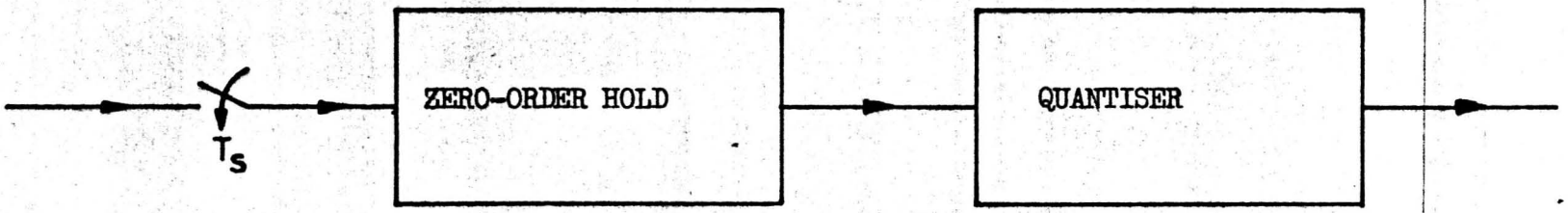


FIG. C.1 Block diagram of basic A/D converter

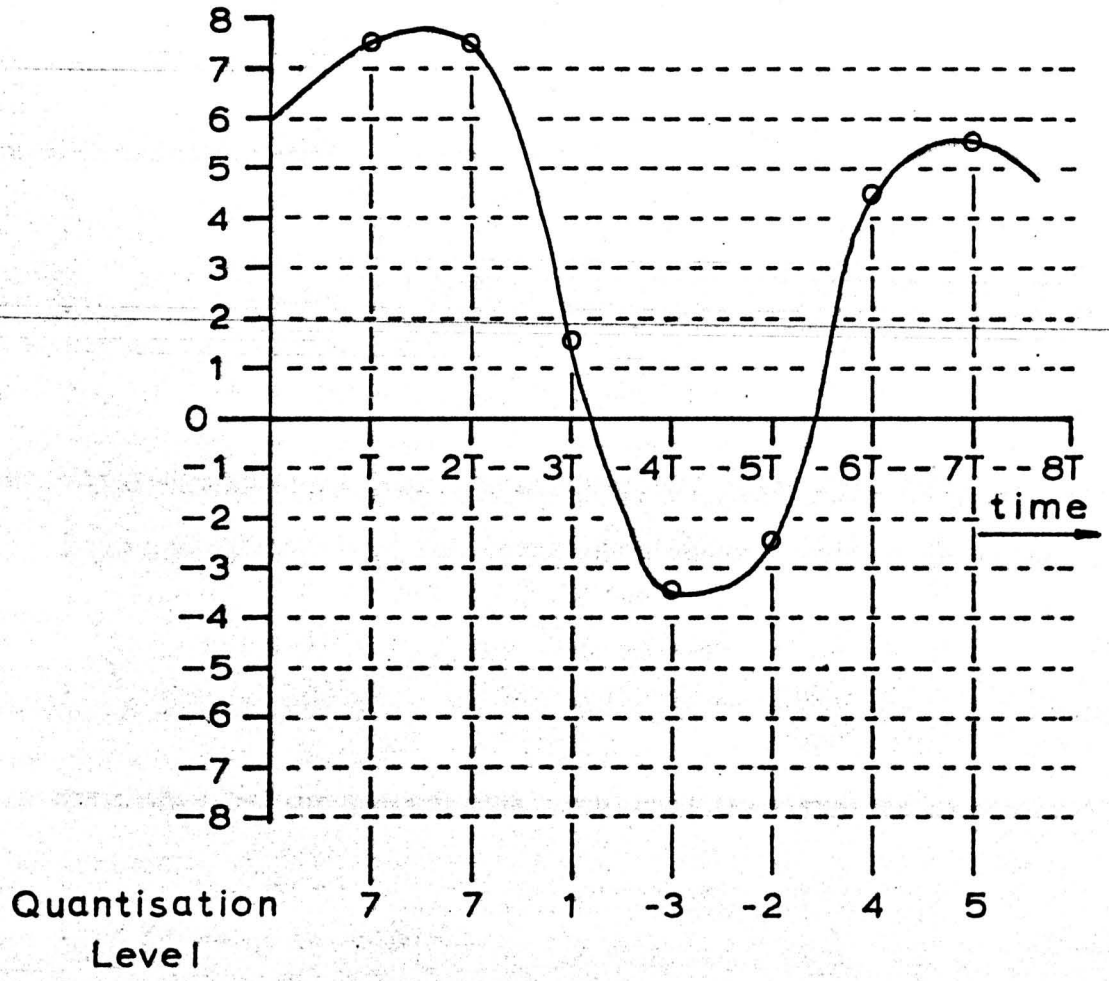


FIG. C.2 Sampling and Quantising

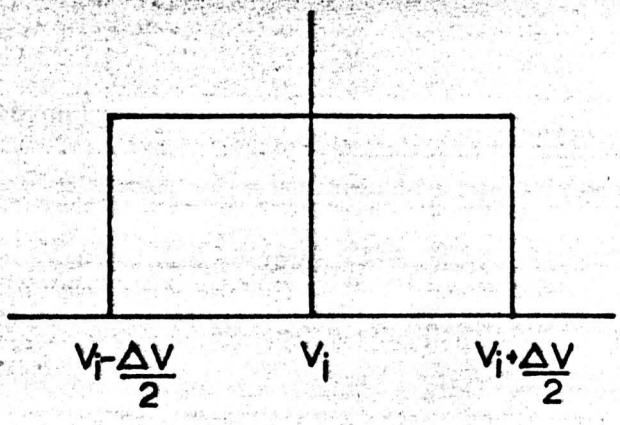


FIG. C.3 Noise Distribution

$$A = 100 / (2^n - 1) \ \% \qquad (C.3)$$

$$R = 20 \log_{10} (2^n - 1) \text{ d B} \qquad (C.4)$$

C.3 QUANTISATION NOISE

Due to this quantisation process, noise is introduced into the signal path due to the intrinsic approximation made of the instantaneous signal amplitude.

From [18] provided that $\triangle v \ll V$ this noise may be assumed to be uniformly distributed over any given step. (See Fig. C.3).

From reference [18] the total mean square quantising noise voltage σ^2 or variance is derived as :

$$\sigma^2 = \frac{(\triangle v)^2}{12} \qquad (C.5)$$

From this result the maximum quantisation signal/noise ratio can be derived :

$$S_q = 20 \log_{10} (1.225M) \qquad (C.6)$$

C.4 QUANTISATION ALGORITHM

A convenient and fast method of quantising a signal into typically a binary code is to use the successive approximation algorithm. [26] This algorithm has the advantage that it takes only N iterations to quantise a signal into one of $2^N - 1$ levels.

C.4.1 Operation

The block diagram of a successive approximations A/D converter is shown in Fig. C.4. The output from this converter can be as a serial bit stream or as sequential parallel words.

The algorithm tests the input signal in successive half ranges and sets the bits in a storage register accordingly.

The N-stage test register is designed as a shift register to

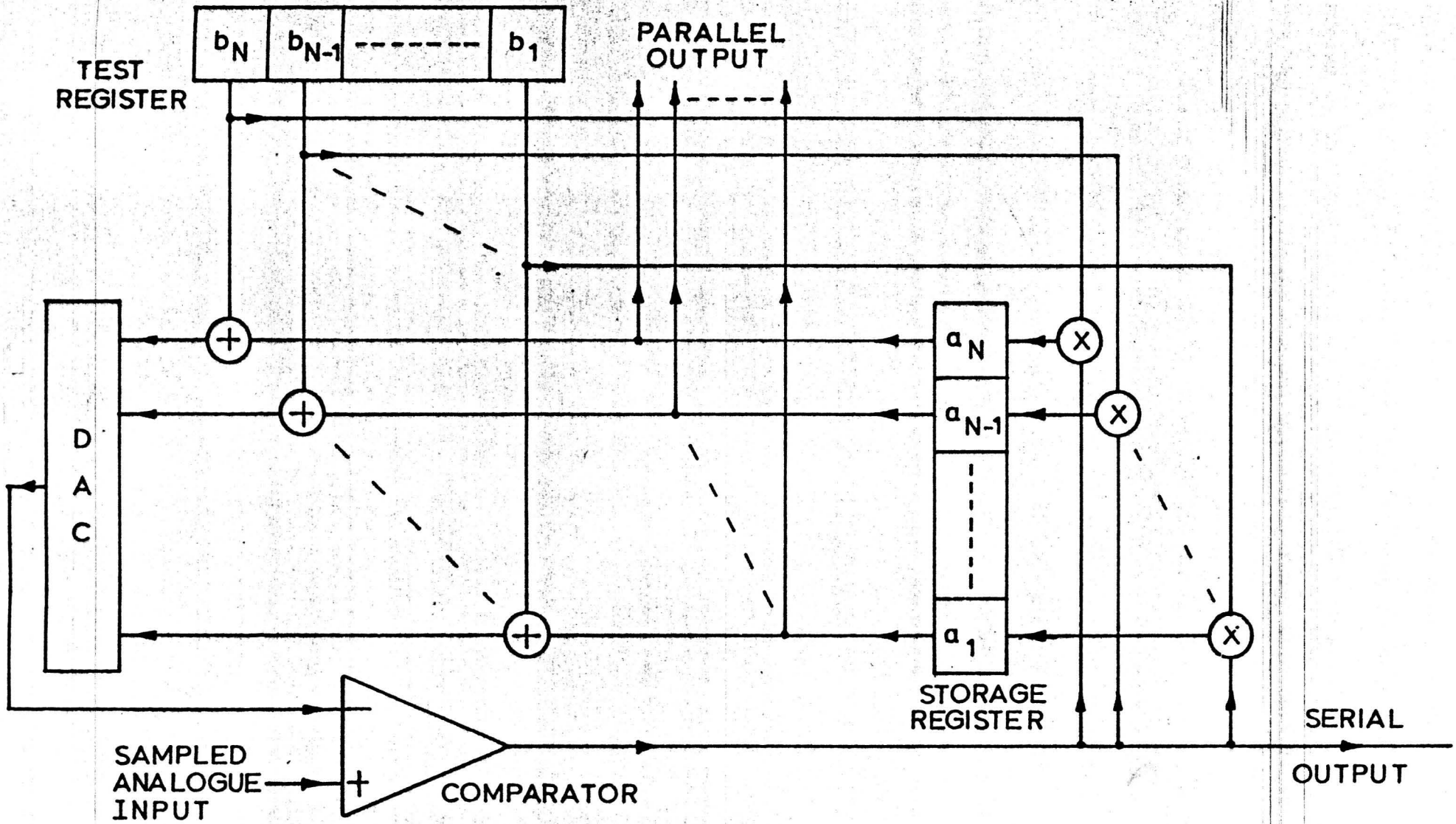


FIG. C.4 Successive Approximation A/D converter

propagate a solitary high state starting at b_N. The direction is from b_N to b_1. The parallel output is inclusive OR'ed to the storage register output and fed to the DAC to test the next interval.

The N-stage storage register has its stages sequentially indexed by the test register. If the comparator output indicates that the signal is in the current upper half-range then that stage is set high.

The D/A converter (DAC) is needed to convert the testing binary pattern from the test and storage registers to an analogue signal.

The comparator gives a high or low output depending on the relative polarity of the analogue signal to be converted and the current binary testing pattern.

APPENDIX D

COMPUTER PROGRAM LISTINGS

The FORTRAN source text of the 5 main programs and their overlay segments are listed here. All the routines were written solely to verify the analyses presented in this thesis.

Any subprograms that are called in these listings but are not listed here belonged to the various system libraries. In particular PLOTTER (Segment 18) is not listed.

```
      PROGRAM GDP1
C GYRATOR DATA PREPARATION PROGRAM.
C
C SUBROUTINES USED :
C       REDATA
C       DATOUT
C
C 0 COMMAND STREAM
C 1 MONITOR STREAM
C 2 ASR KEYBOARD
C 3 ASR PRINTER
C 4 L/P
C 5 DISK OUTPUT FILE
C
      INTEGER NT,NC1,NC2,NG1,NG2,NG,NEL,
     1 NCFN,NCFD,IRES,STAR,NO,YES
C
      LOGICAL TERM,PERPH
C
      DIMENSION T(100),C1(100),C2(100),G1(100),G2(100),
     1 RS(100),ITITLE(72),PERPH(3),IPERPH(3),DATN(6),DATD(6)
C
      DATA YES,NO,STAR,IMP,IADM /1HY,1HN,1H*,1HZ,1HY/
      DATA NCON,NEND /0,9/
      DATA ISPACE,IQUANT /1H ,1HQ/
C
C INTRODUCTION :-
9     WRITE(3,90)
90    FORMAT(' GYRATOR DATA PREPARATION PROGRAM'/)
      IREPLY=1
C
C PERIPHERALS :
10    WRITE(3,100)
100   FORMAT(' L/P,ASR,DISK FILE REQUIRED? (Y/N) :'/)
      READ(2,101) IPERPH
101   FORMAT(3(A1))
      DO 11 I=1,3
      PERPH(I)=.FALSE.
      IF(IPERPH(I).EQ.STAR) GO TO 40
      IF(IPERPH(1).NE.YES.AND.IPERPH(I).NE.NO) GO TO 10
      IF(IPERPH(I).EQ.YES) PERPH(I)=.TRUE.
11    CONTINUE
C
C GET HEADER FOR DISK FILE IF REQUIRED
      IF(.NOT.PERPH(3)) GO TO 50
      WRITE(3,55)
55    FORMAT(' GIVE HEADING FOR DISK FILE'/)
      NCHARS=72
      DO 58 I=1,NCHARS
      ITITLE(I)=ISPACE
58    CONTINUE
C
      CALL GETLIN(2,ITITLE,NCHARS)
C NOW WRITE TO DISK FILE
      WRITE(5) NCHARS,ITITLE
```

```
      DO 200 IT=1,NT
      TIMCON=T(IT)
      IF(PERPH(1)) WRITE(4,1500) TIMCON
      IF(PERPH(2)) WRITE(3,1500) TIMCON
      DO 201 IC1=1,NC1
      CAP1=C1(IC1)
      IF(PERPH(1)) WRITE(4,1501) CAP1
      IF(PERPH(2)) WRITE(3,1501) CAP1
      DO 202 IC2=1,NC2
      CAP2=C2(IC2)
      IF(PERPH(1)) WRITE(4,1502) CAP2
      IF(PERPH(2)) WRITE(3,1502) CAP2
      DO 203 IG1=1,NG1
      COND1=G1(IG1)
      IF(PERPH(1)) WRITE(4,1503) COND1
      IF(PERPH(2)) WRITE(3,1503) COND1
      DO 204 IG2=1,NG2
      COND2=G2(IG2)
      IF(PERPH(1)) WRITE(4,1504) COND2
      IF(PERPH(2)) WRITE(3,1504) COND2
      DO 205 IRS=1,NRS
      RSHUNT=RS(IRS)
      IF(PERPH(1)) WRITE(4,1505) RSHUNT
      IF(PERPH(2)) WRITE(3,1505) RSHUNT
C
C
1500  FORMAT(' T   :',E12.4)
1501  FORMAT(' C1  :',E12.4)
1502  FORMAT(' C2  :',E12.4)
1503  FORMAT(' G1  :',E12.4)
1504  FORMAT(' G2  :',E12.4)
1505  FORMAT(' R   :',E12.4/)
C
      ALPHA=EXP(-1.0*TIMCON/CAP1/RSHUNT)
C
      GO TO (210,211,212,213,214,215,216,217,
     1218,219,220,221,222,223),ISWTCH
C
C Y11 :-
210   NCFN=1
      DATN(2)=CAP1/TIMCON
      DATN(1)=-1.0*DATN(2)*ALPHA
      NCFD=1
      DATD(2)=1.0
      DATD(1)=ZERO
      GO TO 228
C
C Y12 :-
211   NCFN=0
      DATN(1)=(1.0-ALPHA)*COND1*CAP1/RSHUNT/TIMCON
      NCFD=2
      DATD(3)=1.0
      DATD(2)=ZERO
      DATD(1)=ZERO
      GO TO 228
C
```

```
C Y21 :-
212   NCFN=0
      DATN(1)=-1.0*COND2
      NCFD=2
      DATD(3)=1.0
      DATD(2)=ZERO
      DATD(1)=ZERO
      GO TO 228
C
C Y22 :-
213   NCFN=1
      DATD(2)=CAP2/TIMCON
      DATN(1)=-1.0*CAP2/TIMCON
      NCFD=1
      DATD(2)=1.0
      DATD(1)=ZERO
      GO TO 228
C
C Z11 :-
214   NCFN=4
      DATN(5)=TIMCON/CAP1
      DATN(4)=-1.0*TIMCON/CAP1
      DATN(3)=ZERO
      DATN(2)=ZERO
      DATN(1)=ZERO
      GO TO 229
C
C Z12 :-
215   NCFN=2
      DATN(3)=-1.0*COND1*TIMCON*(1.0-ALPHA)*RSHUNT/CAP2
      DATN(2)=ZERO
      DATN(1)=ZERO
      GO TO 229
C
C Z21 :-
216   NCFN=2
      DATN(3)=-1.0*COND2*TIMCON*TIMCON/CAP1/CAP2
      DATN(2)=ZERO
      DATN(1)=ZERO
      GO TO 229
C
C Z22 :-
217   NCFN=4
      DATN(5)=TIMCON/CAP2
      DATN(4)=-1.0*DATN(5)*ALPHA
      DATN(3)=ZERO
      DATN(2)=ZERO
      DATN(1)=ZERO
      GO TO 229
C
C Q11 :-
218   NCFN=1
      DATN(2)=TIMCON*COND1*COND2*(1.0-ALPHA)*RSHUNT/CAP2
      DATN(1)=ZERO
      GO TO 229
C
```

```
C Q12 :-
219    NCFN=3
       DATN(4)=COND1*TIMCON/CAP1
       DATN(3)=-1.0*DATN(4)
       DATN(2)=ZERO
       DATN(1)=ZERO
       GO TO 229
C
C Q21 :-
220    NCFN=3
       DATN(4)=-1.0*COND2*TIMCON/CAP2
       DATN(3)=TIMCON*COND2*ALPHA/CAP2
       DATN(2)=ZERO
       DATN(1)=ZERO
       GO TO 229
C
C Q22 :-
221    NCFN=1
       DATN(2)=TIMCON*TIMCON*COND1*COND2/CAP1/CAP2
       DATN(1)=ZERO
       GO TO 229
C
C Q1 :-
222    NCFN=3
       DATN(4)=COND1*TIMCON/CAP1
       DATN(3)=-1.0*DATN(4)
       DATN(2)=TIMCON*COND1*COND2*RSHUNT*(1.0-ALPHA)/CAP2
       DATN(1)=ZERO
       GO TO 229
C
C Q2 :-
223    NCFN=3
       DATN(4)=-1.0*TIMCON*COND2/CAP2
       DATN(3)=TIMCON*COND2*ALPHA/CAP2
       DATN(2)=TIMCON*TIMCON*COND1*COND2/CAP1/CAP2
       DATN(1)=ZERO
       GO TO 229
C
C DENOMINATOR POLYNOMIAL :-
229    NCFD=4
       DATD(5)=1.0
       DATD(4)=-1.0-ALPHA
       DATD(3)=ALPHA
       DATD(2)=ZERO
       DATD(1)=COND1*COND2*TIMCON*(1.0-ALPHA)*
      * RSHUNT/CAP2
C
228    CONTINUE
C
C OUTPUT STATUS WORD
       WRITE(5) NCON
C
C OUTPUT PARAMETERS
       WRITE(5) TIMCON,CAP1,CAP2,COND1,COND2,RSHUNT
C
C OUTPUT NUMERATOR COEFFICIENTS
```

```
C
50     IF(IREPLY.EQ.3) GO TO 26
C
C GET COMPONENT VALUES :
       CALL REDATA('T ',NT,T(1))
       CALL REDATA('C1',NC1,C1(1))
       CALL REDATA('C2',NC2,C2(1))
       CALL REDATA('G1',NG1,G1(1))
       CALL REDATA('G2',NG2,G2(1))
       CALL REDATA('R ',NRS,RS(1))
C
C NUMBER OF RESULTS :-
26     IRES=NT*NC1*NC2*NG1*NG2*NRS
       WRITE(3,116) IRES
116    FORMAT(1H ,'NUMBER OF RESULTS WILL BE :',I8)
C
C TYPE OF MATRIX :-
30     WRITE (1,120)
120    FORMAT(' IMPEDANCE, ADMITTANCE ',
      1'OR QUANTISATION MATRIX',
      2' (Z/Y/Q) :'/)
       READ(2,121) MATRIX
121    FORMAT(A1)
       IF(MATRIX.EQ.STAR) GO TO 40
       IF(MATRIX.NE.IADM.AND.MATRIX.NE.IMP.
      1AND.MATRIX.NE.IQUANT) GO TO 30
       WRITE(3,1200)
1200   FORMAT(' DIVE ELEMENT :'/)
       READ(2,2200) NEL
2200   FORMAT()
C
       IF(MATRIX.EQ.IADM) IS1=0
       IF(MATRIX.EQ.IMP) IS1=4
       IF(MATRIX.EQ.IQUANT) IS1=8
       IF(NEL.EQ.11) IS2=1
       IF(NEL.EQ.12) IS2=2
       IF(NEL.EQ.21) IS2=3
       IF(NEL.EQ.22) IS2=4
       IF(NEL.EQ.1) IS2=5
       IF(NEL.EQ.2) IS2=6
       IF(IS2.LT.1.OR.IS2.GT.6) GO TO 30
       ISWTCH=IS1+IS2
C
       IF(PERPH(3)) WRITE(5) IRIX(MATRIX,NEL
       IF(PERPH(2)) WRITE(3,151) MATRIX,NEL
       IF(PERPH(1)) WRITE(4,150) MATRIX,NEL
150    FORMAT(1H1,' MATRIX ELEMENT :',A1,I3)
151    FORMAT(' MATRIX ELEMENT :',A1,I3)
C
C CLEAR NUMERATOR & DENOMINATOR ARRAYS
       DO 310 I=1,6
       DATN(I)=ZERO
       DATD(1)=ZERO
310    CONTINUE
C
C
```

```
230    CALL DATOUT(NCFN,DATN,1,PERPH)
C
C OUTPUT DENOMINATOR COEFFICIENTS
       CALL DATOUT(NCFD,DATD,2,PERPH)
C
C NOW CALCULATE THE MINIMUM NUMBER OF BITS NECESSARY :
       DBIT=ZERO
       DO 320 I=1,6
       DBIT=DBIT+DATD(I)
320    CONTINUE
       BIT=AMAX1(DATD(1),DATD(2),DATD(3),DATD(4),
      1DATD(5),DATD(6))
       BIT=ALOG10(ABS(BIT/2./DBIT))/ALOG10(2.)
       NUMBIT=IFIX(BIT)+1
       IF((BIT-FLOAT(NUMBIT)).GT.0.0) NUMBIT=NUMBIT+1
       IF(PERPH(1)) WRITE(4,300) NUMBIT
       IF(PERPH(2)) WRITE(3,300) NUMBIT
300    FORMAT(' NUMBER OF BITS NECESSARY =',I4/)
C
C
205    CONTINUE
204    CONTINUE
203    CONTINUE
202    CONTINUE
201    CONTINUE
200    CONTINUE
C
C REWIND DISK FILE IF REQUIRED
40     IF(.NOT.PERPH(3)) GO TO 45
       WRITE(5) NEND
C LOADZE IS NECESSARY BEFORE FILE REWIND
       CALL LOADZE
       ENDFILE 5
       REWIND 5
C
C
45     WRITE(3,130)
130    FORMAT(' REPEAT, RESTART,',
      1' CHANGE PERPHS OR END? (1,2,3,4) :'/)
       READ(2,131) IREPLY
131    FORMAT()
C
       GO TO (30,10,10,500),IREPLY
C
500    CONTINUE
       STOP
       END
       FINISH
```

```
      PROGRAM PZP1
C POLE-ZERO PLOTTING PROGRAM
C
C PERIPHERALS USED
C 2=ASR READ
C 3=ASR PRINT
C 4=L/P
C 5=INPUT DISK FILE
C 6=OUTPUT DISK FILE
C 7=GRAPH PLOTTER
C
C SEGMENTS USED
C  7 - TEKLIB
C  8 - PLOTLIB
C  9 - ROOTLIB
C 10 - POLYLIB
C 15 - PZPLIB
C 16 - POLENT
C
      INTEGER YES
      LOGICAL PERPH,NOTPER
      DIMENSION IPERPH(5)
      COMMON /PLTINC/ XYINC,XYMAX
      COMMON /MATELM/ NUMRES,MATRIX,NELEMS
      COMMON /POLYNS/ NDPOLY(60),NPOLY,MPOLY,IPYCNT
      COMMON /COEFFS/ ARRNUM(5),NORNUM,ARRDEN(5),NORDEN
      COMMON /TEKREC/ IRECRD(72),NREC,MREC
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      COMMON /ANGLES/ PI,DEGRAD
      COMMON /PLOTYP/ NUMDEN,NAUTO,INPDEV,ITYPE
      COMMON /SCALES/ XORG,YORG,XYSIZE,XYSCAL,PTSIZE
      COMMON /PERPHS/ PERPH(5),NOTPER(5)
      COMMON /TEK/ IXVAL,IYVAL,LINPAG,LINCNT
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
C
      EXTERNAL POLE,ZERO
C
      DATA YES,NO /1HY,1HN/
      DATA MZERO,MPOLE /1,2/
      DATA NEND /9/
C
      MTITLE=72
C
C CALCULATE BASIC ANGLES
      PI=4.0*ATAN(1.0)
      DEGRAD=180.0/PI
C
C INITIALISE
      MREC=72
      NREC=0
```

```
C
      WRITE(3,1501)
1501  FORMAT(' POLYNOMIAL ROOT FINDING PROGRAM'/
     1' ALL THE COEFFICIENTS MUST BE REAL',
     2 ' & GIVEN IN DESCENDING ORDER.'/)
C
C
C ASK FOR PERIPHERAL DEVICES :
10    WRITE(3,1000)
1000  FORMAT(' OUTPUT PERIPHERALS REQUIRED? (Y/N)'/
     1' L/P, ASR, TEK(DATA), PLOTTER OR DISK FILE :'/)
      READ(2,2000) IPERPH
2000  FORMAT(5(A1))
      DO 3000 I=1,5
      PERPH(I)=.FALSE.
      NOTPER(I)=.TRUE.
      IF(IPERPH(I).EQ.NO) GO TO 3000
      IF(IPERPH(I).NE.YES) GO TO 10
      PERPH(I)=.TRUE.
      NOTPER(I)=.FALSE.
3000  CONTINUE
C
C IF TEK. DATA REQUIRED THEN GET STARTING COORDINATES
C       IF(PERPH(3)) CALL TKSTRT
C GET DETAILS OF GRAPH TO BE PLOTTED
100   IF(PERPH(4)) CALL GPSTRT
C
C INITIALISATION :
200   IERASE=YES
      ANG=45.0/ATAN(1.0)
C
C INITIALISE PARAMETER ARRAY
      DO 3200 IPAR=1,6
      PARNEW(IPAR)=0.0
3200  CONTINUE
C
C ASK FOR INPUT DEVICE :
C DISK FILE = 1
C KEYBOARD  = 2
210   WRITE(3,1200)
1200  FORMAT(' DISK FILE OR KEYBOARD INPUT? (1/2) :'/)
      READ(2,2010) INPDEV
2010  FORMAT()
      IF(INPDEV.LT.1.OR.INPDEV.GT.2) GO TO 210
C
220   WRITE(3,1210)
1210  FORMAT(' NUM.=1, DENOM.=2, BOTH=3 :'/)
      READ(2,2010) NUMDEN
      IF(NUMDEN.LT.1.OR.NUMDEN.GT.3) GO TO 220
C
      WRITE(3,1220)
```

```
1220  FORMAT(' HOW MANY POLYNOMIALS ARE TO BE ANALYSED? :'/)
      READ(2,2010) NPOLY
C
C INITIALISE DISK INPUT
      IF(INPDEV.EQ.1) CALL DISTRT
C
C INITIALISE OUTPUT DEVICES
      IF(PERPH(5)) CALL DOSTRT
      IF(PERPH(1).AND.INPDEV.EQ.1) CALL LPSTRT
C
C INITIALISE VDU IF REQUIRED
C      IF(PERPH(3)) CALL SETVDU
C
C START CALCULATING LOOP
      DO 3300 IPOLY=1,NPOLY
C
C NOW SWITCH ACCORDING TO INPUT DEVICE
      GO TO (300,310),INPDEV
C
C DISC-FILE INPUT
300   CALL DISCIN(NORNUM,ARRNUM(1),NORDEN,
     1ARRDEN(1),NDPOLY(IPOLY))
      IF(PERPH(1)) CALL LPARMS(PARNEW)
      GO TO 320
C
C KEYBOARD INPUT
310   CALL KEYBIN(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),NUMDEN)
C
C ANALYSE NUMERATOR POLYNOMIAL
320   GO TO (330,340,330),NUMDEN
330   IF(PERPH(1)) CALL LPCOEF(NORNUM,ARRNUM(1),1)
      CALL RTFIND(NORNUM,ARRNUM(1),ZERO,1)
340   GO TO (3300,350,350),NUMDEN
C ANALYSE DENOMINATOR POLYNOMIAL
350   IF(PERPH(1)) CALL LPCOEF(NORDEN,ARRDEN(1),2)
      CALL RTFIND(NORDEN,ARRDEN(1),POLE,2)
C
3300  CONTINUE
C END OF MAIN LOOP
C
C
C CLOSE GRAPH PLOTTER
400   IF(PERPH(4)) CALL PLOT(XORG,YORG,1000)
      IF(NOTPER(5)) GO TO 410
C
C REWIND DISK OUTPUT FILE
      CALL LOADZE
      ENDFILE 6
      REWIND 6
      CALL LOADFP
C
```

```
C REWIND DISK INPUT FILE
410    GO TO (420,430),INPDEV
420    CALL LOADZE
       REWIND 5
       CALL LOADFP
C
430    IF(NAUTO.EQ.1.AND.PERPH(3)) CALL COPY
       WRITE(3,1400)
1400   FORMAT(' REPEAT=1, RESTART=2, END=3 :'/)
       READ(2,2010) IREPLY
       GO TO (100,10,999),IREPLY
C
999    CONTINUE
       STOP
       END
```

```
      PROGRAM FRA1
C PROGRAM TO CALCULATE DATA FOR FREQUENCY RESPONSE ANALYSIS.
C
C PERIPHERALS USED
C 2=ASR READ
C 3=ASR WRITE
C 4=L/P
C 5=INPUT DISK FILE
C 6=SCRATCH OUTPUT DISK FILE
C 7=GRAPH PLOTTER
C
C SEGMENTS USED
C  7 - TEKLIB
C 10 - POLYLIB
C 11 - FREQLIB
C 12 - RESPLIB
C 13 - FRALIB
C 15 - PZPLIB
C 16 - POLENT
C 18 - PLOTTER
C
      INTEGER YES,NO
      COMPLEX POINT
      DIMENSION IARRAY(72),IPERPH(2)
      LOGICAL PLTPTS,PERPH,NOTPER
      COMMON /COORDS/ PLTPTS,PTSIZE,PTANG,IPTTYP
      COMMON /MATELM/ NUMRES,MATRIX,NELEMS
      COMMON /POLYNS/ NDPOLY(60),NPOLY,MPOLY,IPYCNT
      COMMON /LABELS/ LOGF(13),LINF(11)
      COMMON /ORIGIN/ XORGB,YORGB,XORGP,YORGP,XORGN,YORGN
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
      COMMON /COEFFS/ ARRNUM(60),NORNUM,ARRDEN(60),NORDEN
      COMMON /ANGLES/ PI,PI2
      COMMON /CIL/ IPENX,IPENY,IPS,IPC,IPCN,FACR,
     1XM,SIZES,SIZEN,SIZEL,TICK,STEP,XSPAC,
     2IPNAB,ITAPE,IB,IBC,IBYTE,IBASE,IMMET
      COMMON /CILUNT/ IPLUNT
      COMMON /SCALES/ XORG,YORG,SIZE,SCALEX,SCALEY,
     1PLTSIZ,IPLANE
      COMMON /PERPHS/ PERPH(2),NOTPER(2)
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      COMMON /FQAXIS/ FQMIN,FQMAX,FQINC
      COMMON /FQAXBM/ FQMINB,FQMAXB,FQINCB
      COMMON /FQAXBP/ FQMINP,FQMAXP,FQINCP
      COMMON /MAGAX/ AMPMIN,AMPMAX,AMPINC,IBPLOT
      COMMON /PHASAX/ PHAMIN,PHAMAX,PHAINC,IPPLOT
      COMMON /NYQUAX/ REMAX,RIMAX,RINCX,RINCY,INPLOT
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      EQUIVALENCE (PARNEW(1),TSAMP)
      EXTERNAL ZPOLY,SPOLY,MAGPLT,PHAPLT,NYQPLT
```

```
      DATA ZERO /0.0/
      DATA YES,NO /1HY,1HN/
      DATA NEND,IZERO,MAXORD,MAXPLY /9,0,60,60/
C
C
      MPOLY=60
      PI=4.0*ATAN(1.0)
      PI2=2.0*PI
      PLTPTS=.FALSE.
      PTSIZE=0.2
      PTANG=0.0
      IPTTYP=-6
C
C
1000  WRITE(3,2000)
2000  FORMAT(' FREQUENCY RESPONSE CALCULATION PROGRAM'/)
C
C
1010  WRITE(3,2010)
2010  FORMAT(' L/P AND PLOTTER REQUIRED? (Y/N):'/)
      READ(2,2020) IPERPH
2020  FORMAT(2A1)
      DO 1030 I=1,2
      PERPH(I)=.FALSE.
      NOTPER(I)=.FALSE.
      IF(IPERPH(I).NE.YES.AND.IPERPH(I).NE.NO) GO TO 1010
      IF(IPERPH(I).EQ.YES) PERPH(I)=.TRUE.
      IF(IPERPH(I).EQ.NO) NOTPER(I)=.TRUE.
1030  CONTINUE
C
1100  WRITE(3,2100)
2100  FORMAT(' Z OR S PLANE (1/2) :'/)
      READ(2,3210) IPLANE
3210  FORMAT()
      IF(IPLANE.LT.1.OR.IPLANE.GT.2) GO TO 1100
C
C
1140  WRITE(3,2140)
2140  FORMAT(' HOW MANY TEST FREQUENCIES? :'/)
      READ(2,2145) NPOINT
2145  FORMAT()
      IF(NPOINT.LT.1) GO TO 1140
      POINTS=FLOAT(NPOINT-1)
C
1150  WRITE(3,2150)
2150  FORMAT(' LOG OR LIN FREQUENCY SCAN? (1/2) :'/)
      READ(2,3210) ISCAN
      IF(ISCAN.LT.1.OR.ISCAN.GT.2) GO TO 1150
1120  WRITE(3,2120)
2120  FORMAT(' MINIMUM & MAXIMUM FREQUENCY :'/)
      READ(2,2125) FQMIN,FQMAX
```

```
2125  FORMAT()
      IF(FQMAX.LT.0.0) GO TO 1120
      IF(FQMIN.LT.0.0) GO TO 1120
      IF(FQMAX.GE.FQMIN) GO TO 1200
C
C SWOP OVER :
      TEMP=FQMAX
      FQMAX=FQMIN
      FQMIN=TEMP
C
1200  IF(NOTPER(2)) GO TO 1300
      CALL PLOTS(7)
      IF(NPOINT.LT.0) CALL AXIS(0.,
     10.,0.,0.,0.,0.,IARRAY(1),NARRAY)
C
C NOW GET GRAPH COMMENT
      WRITE(3,1210)
1210  FORMAT(' GIVE GRAPH COMMENT :'/)
      NARRAY=72
      CALL GETLIN(2,IARRAY(1),NARRAY)
C NOW CONVERT TO A2-FORMAT
      CALL A1A2ST(IARRAY(1),NARRAY,ITITG(1),NTITG)
C
      CALL SETUP
C
C ASK FOR INPUT DEVICE :
1300  WRITE(3,2300)
2300  FORMAT(' DISC-FILE OR KEYBOARD INPUT (1/2) :'/)
      READ(2,3210) IDEV
      IF(IDEV.LT.1.OR.IDEV.GT.2) GO TO 1300
1310  WRITE(3,2310)
2310  FORMAT(' HOW MANY POLYNOMIALS? (<=60) :'/)
      READ(2,3210) NPOLY
      IF(NPOLY.LT.1.OR.NPOLY.GT.MAXPLY) GO TO 1310
C
      GO TO (1320,1330),IDEV
C
C READ TITLE & HEADING FROM DISK FILE
1320  CALL DISTRT
      GO TO 1400
C
1330  IF(IPLANE.NE.1) GO TO 1400
      WRITE(3,2330)
2330  FORMAT(' GIVE SAMPLING PERIOD :'/)
      READ(2,3210) TSAMP
C
C START POLNOMIAL ANALYSIS LOOP
1400  DO 3400 IPOLY=1,NPOLY
C DISC-FILE INPUT
      IF(IDEV.EQ.1) CALL DISCIN(NORNUM,ARRNUM(1),
     1 NORDEN,ARRDEN(1),NDPOLY(IPOLY))
```

```
C KEYBOARD INPUT
      IF(IDEV.EQ.2) CALL KEYBIN(NORNUM,ARRNUM(1),
     1NORDEN,ARRDEN(1),3)
C
C
C CALCULATE FREQUENCY INCREMENTS
1600  GO TO (1610,1620),ISCAN
1610  FQINC=EXP(ALOG(FQMAX/FQMIN)/POINTS)
      GO TO 1630
1620  FQINC=(FQMAX-FQMIN)/POINTS
C
C NOW FIND FREQUENCY RESPONSE
1630  GO TO (1640,1650),IPLANE
C Z-PLANE
1640  CALL FQRESP(ZPOLY,NPOINT)
      GO TO 1660
C S-PLANE
1650  CALL FQRESP(SPOLY,NPOINT)
C
C REWIND TEMPORARY DISK FILES
1660  CALL LOADZE
      REWIND 6
      CALL LOADFP
C
C INITIALISE OUTPUT DEVICES :
C
C LINE PRINTER :
1700  IF(NOTPER(1)) GO TO 1800
      CALL LPFST(IDEV)
      CALL LPFOUT
C REWIND DISK FILE :
      CALL LOADZE
      REWIND 6
      CALL LOADFP
C
C GRAPH PLOTTER OUTPUT
1800  IF(NOTPER(2)) GO TO 1900
C
C BODE MAGNITUDE PLOT
      IF(IBPLOT.EQ.NO) GO TO 1830
      CALL GRAPLT(MAGPLT,XORGB,YORGB)
C
C NOW REWIND THE TEMPORARY DISK FILE
      CALL LOADZE
      REWIND 6
      CALL LOADFP
C
C BODE PHASE PLOT
1830  IF(IPPLOT.EQ.NO) GO TO 1860
      CALL GRAPLT(PHAPLT,XORGP,YORGP)
C
```

```
C REWIND TEMPORARY DISK FILE
      CALL LOADZE
      REWIND 6
      CALL LOADFP
C
C NYQUIST PLOT
1860  IF(INPLOT.EQ.NO) GO TO 1900
      CALL GRAPLT(NYQPLT,XORGN,YORGN)
C
C REWIND TEMPORARY DISK FILE
      CALL LOADZE
      REWIND 6
      CALL LOADFP
C
1900  GO TO (3400,1990),IDEV
C
C REWIND DATA FILE
1990  IF(IDEV.NE.1) GO TO 3400
      CALL LOADZE
      REWIND 5
      CALL LOADFP
C
C
3400  CONTINUE
C
C FINISH DRAWING GRAPHS
      IF(PERPH(2)) CALL PLOT(ZERO,-100.0,999)
C
1995  WRITE(3,2990)
2990  FORMAT(' RE-RUN, RESTART, CHANGE PERIPHERALS',
     1' OR END (1,2,3,4)'/)
      READ(2,2995) IEND
2995  FORMAT()
      IF(IEND.LT.1.OR.IEND.GT.4) GO TO 1995
      GO TO (1300,1010,1010,1999),IEND
C
1999  CONTINUE
      STOP
C
      END
```

```
      PROGRAM IZT1
C FINDS INVERSE Z-TRANSFORM FROM Z-PLANE TO DISCRETE
C TIME DOMAIN.
C
C PERIPHERALS USED
C 2=ASR READ
C 3=ASR WRITE
C 4=L/P
C 5=INPUT DISK FILE
C 6=GRAPH PLOTTER
C
C SEGMENTS USED
C  7 - TEKLIB
C 10 - POLYLIB
C 15 - PZPLIB
C 16 - POLENT
C 17 - IZTLIB
C
      INTEGER YES,NO
      DIMENSION IMPTIT(8),INVTIT(9),IARRAY(72)
      DIMENSION IPERPH(2)
      DIMENSION ARRNUM(60),ARRDEN(60)
      LOGICAL PERPH,NOTPER,AXDRAW,DRAW
      COMMON /PRINTS/ PRINT(10),IPRINT,MPRINT
      COMMON /PLTST/ AXDRAW,DRAW
      COMMON /CIL/ IPENX,IPENY,IPS,IPC,IPCN,FACR,
     1XM,SIZES,SIZEN,SIZEL,TICK,STEP,XSPAC,
     2IPNAB,ITAPE,IB,IBC,IBYTE,IBACT,IMMET
      COMMON /CILUNT/ IPLUNT
      COMMON /PERPHS/ PERPH(2),NOTPER(2)
      COMMON /TIMORG/ XORGT,YORGT,SIZEX,SIZEY,SCALEX,SCALEY,
     1XINC,YINC,XMAXT,YMAXT
      COMMON /POLYNS/ NDPOLY(60),NPOLY,MPOLY,IPYCNT
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      COMMON /MATELM/ NUMRES,MATRIX,NELEMS
C
      DATA IMPTIT /2HIM,2HPU,2HLS,2HE ,2HRE,2HSP,2HON,2HSE/
      DATA INVTIT /2HIN,2HVE,2HRS,2HE ,2HTR,
     12HAN,2HSF,2HOR,2HM /
      DATA IMPNUM,INVNUM /8,9/
      DATA YES,NO,MAXPTS /1HY,1HN,1000/
      DATA GAP /2.0/
C
C SET UP PRINT BUFFER
      IPRINT=0
      MPRINT=10
C
C OUTPUT TITLE
1000  WRITE(3,2000)
```

```
2000  FORMAT(' INVERSE Z-TRANSFORM PROGRAM'/)
C
1010  WRITE(3,2010)
2010  FORMAT(' L/P & PLOTTER REQUIRED? (Y/N) :'/)
      READ(2,2020) IPERPH
2020  FORMAT(2A1)
      DO 3010 I=1,2
      PERPH(I)=.FALSE.
      NOTPER(I)=.FALSE.
      IF(IPERPH(I).NE.YES.AND.IPERPH(I).NE.NO) GO TO 1010
      IF(IPERPH(I).EQ.YES) PERPH(I)=.TRUE.
      IF(IPERPH(I).EQ.NO) NOTPER(I)=.TRUE.
3010  CONTINUE
      IF(NOTPER(1).AND.NOTPER(2)) GO TO 1010
C
1200  WRITE(3,2200)
2200  FORMAT(' DISC-FILE OR KEYBOARD INPUT? (1/2) :'/)
      READ(2,2210) IDEV
2210  FORMAT()
      IF(IDEV.LT.1.OR.IDEV.GT.2) GO TO 1200
C
C ASSIGN GRAPH PLOTTER TO DEVICE 6
      IF(PERPH(2)) CALL PLOTS(6)
1210  WRITE(3,2220)
2220  FORMAT(' HOW MANY POLYNOMIALS? (<=60) :'/)
      READ(2,2210) NPOLY
      IF(NPOLY.LT.1.OR.NPOLY.GT.60) GO TO 1210
C
C ASCERTAIN HOW MANY POINTS TO FIND
1300  WRITE(3,2300)
2300  FORMAT(' HOW MANY INVERSE & IMPULSE',
     1' ITERATIONS? (>=0) :'/)
      READ(2,2210) INVCNT,IMPCNT
      IF(INVCNT.LE.0.OR.IMPCNT.LE.0) GO TO 1300
1320  IF(NOTPER(2)) GO TO 1350
C
      WRITE(3,2320)
2320  FORMAT(' GIVE MAXIMUM INVERSE &',
     1' IMPULSE AMPLITUDES :'/)
      READ(2,2210) AMPINV,AMPIMP
      IF(AMPINV.LE.0.0.OR.AMPIMP.LE.0.0) GO TO 1320
1330  WRITE(3,2330)
2330  FORMAT(' GIVE SCALES OF AXES :'/)
      READ(2,2210) SCALEX,SCALEY
C
C SET UP VARIABLES
1350  CALL SETIZT
C
C NOW GET COMMENT FOR GRAPH
      IF(NOTPER(2)) GO TO 1400
      NARRAY=72
```

```
      WRITE(3,2350)
2350  FORMAT(' GIVE GRAPH COMMENT :'/)
      CALL GETLIN(2,IARRAY(1),NARRAY)
C NOW CONVERT 'IARRAY' TO A2-FORMAT
      CALL A1A2ST(IARRAY(1),NARRAY,ITITG(1),NTITG)
C
1400  IF(NOTPER(2)) GO TO 1410
      CALL TIMPLT(10.0,SIZEY*SCALEY,FLOAT(INVCNT),
     1AMPINV,INVTIT(1),INVNUM)
C
C NOW ANNOTATE GRAPH
      CALL ANNOTE
C
C CALCULATE AXIS SIZES IN MILLIMETRES
      INVPTS=IFIX(10.0*SIZEX*SCALEX)
      IMPPTS=INVPTS
C
C BRANCH IF DISK FILE OR KEYBOARD ENTRY
1410  GO TO (1415,1490),IDEV
C
C DISK FILE
C NOW READ HEADER OFF FILE
1415  CALL DISTRT
C
C NOW START CALCULATING LOOP
      DO 3400 IPOLY=1,NPOLY
C
C READ DISK RECORD
      CALL DISCIN(NORNUM,ARRNUM(1),NORDEN,
     1ARRDEN(1),NDPOLY(IPOLY))
C
C NOW CALCULATE THE INVERSE TRANSFORM
      CALL ZTRINV(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),
     1INVPTS,INVCNT)
C
3400  CONTINUE
C
C
C NOW REWIND THE FILE
      REWIND 5
C
C NOW RE-READ THE FILE HEADER
      CALL DISTRT
C
C NOW DRAW NEW AXES
      IF(NOTPER(2)) GO TO 1420
      CALL TIMPLT(10.0,2.0*SIZEY*SCALEY+GAP,FLOAT(IMPCNT),
     1AMPIMP,IMPTIT(1),IMPNUM)
C
C & RESTART LOOP
1420  DO 3410 IPOLY=1,NPOLY
```

```
C
C READ DISK RECORD
      CALL DISCIN(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),
     1NDPOLY(IPOLY))
C
C NOW CALCULATE THE IMPULSE RESPONSE
      CALL ZTRIMP(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),
     1IMPPTS,IMPCNT)
C
3410  CONTINUE
C
      GO TO 1800
C
C KEYBOARD ENTRY
1490  IF(NOTPER(2)) GO TO 1495
      CALL TIMPLT(10.0,2.0*SIZEY*SCALEY+GAP,FLOAT(IMPCNT),
     1AMPIMP,IMPTIT(1),IMPNUM)
C
C START CALCULATING LOOP
1495  DO 3500 IPOLY=1,NPOLY
C
C SET TO INVERSE TRANSFORM PLOT
      CALL SETPLT(10.0,SIZEY*SCALEY,FLOAT(INVCNT),AMPINV)
C
C READ KEYBOARD RECORD
      CALL KEYBIN(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),3)
C
C NOW CALCULATE THE INVERSE TRANSFORM
      CALL ZTRINV(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),
     1INVPTS,INVCNT)
C
C NOW SET TO THE IMPULSE RESPONSE GRAPH
      CALL SETPLT(10.0,2.0*SIZEY*SCALEY+GAP,FLOAT(IMPCNT),
     1AMPIMP)
C
C NOW CALCULATE THE IMPULSE RESPONSE
      CALL ZTRIMP(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),
     1IMPPTS,IMPCNT)
C
3500  CONTINUE
C
1800  IF(PERPH(2)) CALL PLOT(XORGT,YORGT,-999)
C
1900  WRITE(3,2900)
2900  FORMAT(' RESTART, REPEAT OR END (1,2 OR 3) :'/)
      READ(2,2210) IEND
      IF(IEND.LT.1.OR.IEND.GT.3) GO TO 1900
      GO TO (1010,1300,1999),IEND
C
1999  STOP
      END
```

```
C       TEKLIB
        SEGMENT 7
        REPEAT INTRINSICS

        SUBROUTINE A2A1ST(LIST1,NUMIN,LIST2,NUMOUT)
C CONVERTS LIST1 (A2-FORMAT) TO LIST2 (A1-FORMAT)
C
        DIMENSION LIST1(1),LIST2(1)
        DATA IBYTE /256/
C
C
        NUMOUT=0
        DO 10 I=1,NUMIN
        NUMOUT=NUMOUT+1
        M=LIST1(I)
        CALL MASK(M,-256)
        LIST2(NUMOUT)=M+160
C
        NUMOUT=NUMOUT+1
        M=LIST1(I)
        CALL MASK(M,255)
        M=M*IBYTE
        CALL MASK(M,-256)
        LIST2(NUMOUT)=M+160
10      CONTINUE
        RETURN
        END
```

```
      SUBROUTINE A1A2ST(LIST1,NUMIN,LIST2,NUMOUT)
C CONVERTS LIST1 (A1-FORMAT) TO LIST2 (A2-FORMAT)
C
      DIMENSION LIST1(1),LIST2(1)
      DATA IBYTE,ISPACE /256,32/
C
      NUMOUT=0
      I=0
C
10    I=I+1
      NUMOUT=NUMOUT+1
      M=LIST1(I)
      CALL MASK(M,-256)
      LIST2(NUMOUT)=M
      IF(I.GE.NUMIN) GO TO 30
      I=I+1
      M=LIST1(I)
      CALL MASK(M,32512)
      M=M/IBYTE
      CALL MASK(M,255)
      LIST2(NUMOUT)=LIST2(NUMOUT)+M
      IF(I.GE.NUMIN) GO TO 20
      GO TO 10
C
20    RETURN
C
C NOW TEST FOR NUMIN ODD
30    IF(2*(NUMIN/2).NE.NUMIN) LIST2(NUMOUT)=LIST2(NUMOUT)+ISPACE
C
      RETURN
      END
```

```
C PLOTLIB
      SEGMENT 8
      REPEAT INTRINSICS
      SUBROUTINE PLANE(X,Y,IPLANE)
C PLOTS CARTESIAN COORDINATES ON THE GRAPH PLOTTER WITH THE
C ORIGIN AT (XORG,YORG).
C
      INTEGER SPLANE,ZPLANE
      REAL NINETY
      DIMENSION SPLANE(4),ZPLANE(12),IMAGIN(5),IARRAY(36)
      DIMENSION IGRAPH(3)
      COMMON /PLTINC/ XYINC,XYMAX
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      COMMON /SCALES/ XORG,YORG,XYSIZE,XYSCAL,PTSIZE
C
      DATA IGRAPH /2HGR,2HAP,2HH /
      DATA SPLANE /2HS-,2HPL,2HAN,2HE /
      DATA ZPLANE /2HZ-,2HPL,2HAN,2HE ,2HAN,
     12HD ,2HUN,2HIT,2H-C,2HIR,2HCL,2HE /
      DATA NINETY,CHSIZE /90.0,0.4/
      DATA ZERO,ROUND,ONEPT1 /0.0,360.0,1.1/
      DATA IMAGIN /2HIM,2HAG,2HIN,2HAR,2HY /
C
      XYMAX=1.0
      XORG=X
      YORG=Y
C DRAW AXES
C INITIALISE PEN POSITION
      CALL PLOT(XORG,YORG,-3)
C
C NOW DRAW THE AXES
      SIZE=XYSIZE*XYSCAL
      XYINC=XYMAX/SIZE
      SHIFT=AINT(ONEPT1*SIZE)
C
C DRAW POSITIVE X-AXIS
      CALL AXIS(ZERO,ZERO,SHIFT,ZERO,ZERO,XYINC,'REAL',4)
C
C DRAW NEGATIVE X-AXIS
      CALL AXIS(-SHIFT,ZERO,SHIFT,ZERO,
     1-SHIFT*XYINC,XYINC,1H ,1)
C
C DRAW POSITIVE Y-AXIS
      CALL AXIS(ZERO,ZERO,SHIFT,NINETY,ZERO,XYINC,
     1IMAGIN(1),-10)
C
C DRAW NEGATIVE Y-AXIS
      CALL AXIS(ZERO,-SHIFT,SHIFT,NINETY,
     1-SHIFT*XYINC,XYINC,1H ,-1)
C
```

```
C NOW DRAW UNIT CIRCLE IF REQUIRED
      IF(IPLANE.EQ.1) CALL CIRCLE(ZERO,ZERO,
     1SIZE,ROUND)
C
C CONVERT THE TITLE ARRAY FROM A1 TO A2 FORMAT.
      CALL A1A2ST(ITITG(1),NTITG,IARRAY(1),NCHARS)
C
C NOW ADD THE GRAPH TITLE
      YPOSN=SHIFT+2.0
      CALL SYMBOL(-SHIFT,-YPOSN,CHSIZE,IARRAY(1),ZERO,NTITG)
C
C NOW ADD 'GRAPH'
      CALL SYMBOL(2.0,-YPOSN-2.0,CHSIZE,IGRAPH(1),ZERO,6)
C
C NOW ADD THE GRAPH NAME
      GO TO (10,20),IPLANE
10    CALL SYMBOL(-4.0,YPOSN,CHSIZE,ZPLANE(1),ZERO,24)
      GO TO 30
20    CALL SYMBOL(ZERO,YPOSN,CHSIZE,SPLANE(1),ZERO,8)
C
C NOW RESET THE PEN TO THE ORIGIN
30    CALL PLOT(ZERO,ZERO,3)
C
      RETURN
      END
```

```
      SUBROUTINE CIRCLE(XCENT,YCENT,RADIUS,STEPS)
C DRAWS CIRCLE ON GRAPH PLOTTER WITH CENTRE (XCENT,YCENT).
C
      STEP=8.0*ATAN(1.0)/STEPS
C
      CALL PLOT(XCENT+RADIUS,YCENT,3)
C
      ISTEP=IFIX(STEPS+0.5)
      DO 100 I=0,ISTEP
      X=XCENT+RADIUS*COS(STEP*FLOAT(I))
      Y=YCENT+RADIUS*SIN(STEP*FLOAT(I))
      CALL PLOT(X,Y,2)
100   CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE POLE(NORDER,ROOTS)
C PLOTS A CROSS AT THE POLE POSITIONS.
C
      COMPLEX ROOTS
      DIMENSION ROOTS(1)
      COMMON /PLTINC/ XYINC,XYMAX
      COMMON /SCALES/ XORG,YORG,XYSIZE,XYSCAL,PTSIZE
      DATA ROOT2 /1.4142316/
C
      DO 10 I=1,NORDER
      X=REAL(ROOTS(I))/XYINC
      Y=AIMAG(ROOTS(I))/XYINC
      S=PTSIZE/ROOT2
C
      CALL PLOT(X-S,Y-S,3)
      CALL PLOT(X+S,Y+S,2)
      CALL PLOT(X+S,Y-S,3)
      CALL PLOT(X-S,Y+S,2)
10    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE ZERO(NORDER,ROOTS)
C PLOTS A CIRCLE AT THE ZERO POSITIONS.
C
      COMPLEX ROOTS
      DIMENSION ROOTS(1)
      COMMON /PLTINC/ XYINC,XYMAX
      COMMON /SCALES/ XORG,YORG,XYSIZE,XYSCAL,PTSIZE
C
      DO 10 I=1,NORDER
      X=REAL(ROOTS(I))/XYINC
      Y=AIMAG(ROOTS(I))/XYINC
      CALL CIRCLE(X,Y,PTSIZE,18.0)
10    CONTINUE
C
      RETURN
      END
```

```
C ROOTLIB
      SEGMENT 9
      REPEAT INTRINSICS
      SUBROUTINE RTFIND(NORDER,ARRAY,GRAPH,ITYPE)
C ANALYSES POLYNOMIAL WHOSE COEFFICIENTS ARE STORED IN ARRAY
C
      LOGICAL PERPH,NOTPER
      COMPLEX ROOTS,CHECK
      DIMENSION ARRAY(1),ROOTS(4),CHECK(4)
      COMMON /PERPHS/ PERPH(5),NOTPER(5)
C
C DIVIDE THROUGH BY LEADING COEFFICIENT :
      NUMELS=NORDER+1
      CFLEAD=ARRAY(NUMELS)
      DO 10 I=1,NUMELS
      ARRAY(I)=ARRAY(I)/CFLEAD
10    CONTINUE
C
C SWITCH ACCORDING TO POLYNOMIAL ORDER :
      IF(NORDER.LT.1.OR.NORDER.GT.4) RETURN
      GO TO (110,120,130,140),NORDER
C
C FIRST ORDER :
110   CALL RT1ST(ARRAY(1),ROOTS(1))
      GO TO 200
C
C SECOND ORDER :
120   CALL RT2ND(ARRAY(1),ROOTS(1))
      GO TO 200
C
C THIRD ORDER :
130   CALL RT3RD(ARRAY(1),ROOTS(1))
      GO TO 200
C
C FOURTH ORDER :
140   CALL RT4TH(ARRAY(1),ROOTS(1))
C
C CHECK RESULT
200   CALL POLYCX(NORDER,ARRAY(1),ROOTS(1),CHECK(1))
C
C NOW OUTPUT TO PERIPHERALS
      IF(PERPH(2)) CALL ASROUT(NORDER,ROOTS(1),CHECK(1))
C
      IF(PERPH(5)) CALL DISCOP(NORDER,ROOTS(1))
C
      IF(PERPH(1)) CALL LPROOT(NORDER,ROOTS(1),
     1ARRAY(1),ITYPE)
C
      IF(PERPH(3)) CALL LISTLP(NORDER,ROOTS(1),
     1CHECK(1),ARRAY(1))
```

```
C
      IF(PERPH(4)) CALL GRAPH(NORDER,ROOTS(1))
C
      RETURN
      END
```

```
      SUBROUTINE RT1ST(ARRAY,ROOT)
C TRIVIAL ROUTINE TO FIND ROOT OF FIRST ORDER POLYNOMIAL
C
      COMPLEX ROOT
      DIMENSION ARRAY(1)
C
      ROOT=CMPLX(-1.0*ARRAY(1)/ARRAY(2),0.0)
      RETURN
      END
```

```
      SUBROUTINE RT2ND(ARRAY,ROOTS)
C FINDS ROOTS OF SECOND ORDER POLYNOMIAL
      COMPLEX ROOTS
      DIMENSION ARRAY(1),ROOTS(1)
C
      IF(ARRAY(3).EQ.0.0) GO TO 30
      TEMP1=ARRAY(2)*ARRAY(2)-4.0*ARRAY(3)*ARRAY(1)
      TEMP2=SQRT(ABS(TEMP1))*0.5/ARRAY(3)
      TEMP3=-0.5*ARRAY(2)/ARRAY(3)
C
      IF(TEMP1) 20,10,10
10    ROOTS(1)=CMPLX(TEMP3+TEMP2,0.0)
      ROOTS(2)=CMPLX(TEMP3-TEMP2,0.0)
C
      RETURN
C
20    ROOTS(1)=CMPLX(TEMP3,TEMP2)
      ROOTS(2)=CMPLX(TEMP3,-1.0*TEMP2)
C
      RETURN
C
30    ROOTS(1)=CMPLX(-ARRAY(1)/ARRAY(2),0.0)
      ROOTS(2)=(0.0,0.0)
      RETURN
      END
```

```
      SUBROUTINE RT3RD(ARRAY,ROOTS)
C FINDS ROOTS OF THIRD ORDER POLYNOMIALS
C
      COMPLEX ROOTS
      DIMENSION ARRAY(1),CUBCFS(4),ROOTS(3),
     1DIVIDE(2),QUOTNT(3)
C
C MAKE COPY OF ARRAY
      DO 10 I=1,4
      CUBCFS(I)=ARRAY(I)
10    CONTINUE
C
C FIND SINGLE REAL ROOT :
      CALL RTHUNT(CUBCFS(1),ROOT)
      DIVIDE(2)=1.0
      DIVIDE(1)=-ROOT
C DIVIDE THROUGH BY THIS ROOT :
      CALL POLDIV(3,CUBCFS(1),1,DIVIDE(1),NORQNT,QUOTNT(1))
C NOW FIND ROOTS OF RESULTING SECOND ORDER POLYNOMIAL :
      CALL RT2ND(QUOTNT(1),ROOTS(1))
      ROOTS(3)=CMPLX(ROOT,0.0)
      RETURN
      END
```

```
      DFOUR=0.25*BROOT*BROOT-ARRAY(1)
      IF(DFOUR) 60,70,70
60    DFOUR=0.0
      GO TO 80
C
70    DFOUR=SQRT(ABS(DFOUR))
      IF(DFOUR) 90,80,90
80    CFOUR=AFOUR*AFOUR-ARRAY(3)+BROOT
      CFOUR=SQRT(ABS(CFOUR))
      GO TO 100
90    CFOUR=(AFOUR*BFOUR-ARRAY(2)*0.5)/DFOUR
C
100   G(3)=1.0
      G(2)=AFOUR+CFOUR
      G(1)=BFOUR+DFOUR
      CALL RT2ND(G(1),ROOTS(1))
C
210   G(2)=AFOUR-CFOUR
      G(1)=BFOUR-DFOUR
      CALL RT2ND(G(1),ROOTS(3))
C
      RETURN
      END
```

```
      SUBROUTINE RTHUNT(CUBCFS,ROOT)
C FINDS SINGLE REAL ROOT OF THIRD ORDER POLYNOMIAL
C
      DIMENSION CUBCFS(1)
      DATA SIZEXP,ERROR /1E-8,1E-20/
C
      IF(CUBCFS(4)) 5,16,5
5     DO  10  I=1,4
10    CUBCFS(I)=CUBCFS(I)/CUBCFS(4)
      IF(CUBCFS(1).NE.0.0) GO TO 11
      ROOT=0.0
      RETURN
11    SCALE=0.0
      CFSIGN=SIGN(1.0,CUBCFS(1))
      VALUE=ABS(CUBCFS(1)+CUBCFS(3))
      POWER=SIZEXP
12    POWER=10.0*POWER
      IF(VALUE.GE.POWER) GO TO 12
      ORDER=POWER
13    ORDER=ORDER/10.0
      CFSIGN=-1.0*CFSIGN
14    VALUE=SCALE+CFSIGN*ORDER
C
      IF(VALUE.EQ.SCALE) GO TO 15
      SIZE=((VALUE+CUBCFS(3))*VALUE+CUBCFS(2))*VALUE+
     1CUBCFS(1)
      IF (ABS(SIZE).LT.ERROR) GO TO 15
      SCALE=VALUE
      IF (CFSIGN*SIZE.GE.0.0) GO TO 13
      GO TO 14
C
15    ROOT=VALUE
C
      RETURN
16    ROOT=0.0
      RETURN
      END
```

```
      SUBROUTINE RT4TH(ARRAY,ROOTS)
C SUBROUTINE TO FIND ROOTS OF A FOURTH ORDER POLYNOMIAL
C
      COMPLEX ROOTS,QUOTRT,XROOT,YROOT
      DIMENSION ARRAY(1),RELARY(4),FACTOR(2),QUOTNT(2),
     1 G(3),ROOTS(1),QUOTRT(2),XROOT(2),YROOT(3)
C
C
C FIND COEFFS. OF RELATED 3RD ORDER POLYN. :-
      RELARY(4)=1.0
      RELARY(3)=-ARRAY(3)
      RELARY(2)=ARRAY(4)*ARRAY(2)-4.0*ARRAY(1)
      RELARY(1)=ARRAY(1)*(4.0*ARRAY(3)-ARRAY(4)**2)
     1-ARRAY(2)**2
      ROOT=0.0
      IF(RELARY(1)) 10,20,10
C
C FIND ROOTS OF RELATED POLYNOMIAL
10    CALL RTHUNT(RELARY(1),ROOT)
C
      FACTOR(2)=1.0
      FACTOR(1)=-ROOT
      YROOT(1)=CMPLX(ROOT,0.0)
C
C DIVIDE THROUGH TO REMOVE FACTOR
      CALL POLDIV(3,RELARY(1),1,FACTOR(1),NORQNT,QUOTNT(1))
C
C & FIND ROOTS
      CALL RT2ND(QUOTNT(1),QUOTRT(1))
C
      GO TO 30
C
C CONSTANT COEFFICIENT = 0. HENCE REDUCE POLYNOMIAL
C TO SECOND ORDER.
20    YROOT(1)=(0.0,0.0)
      DO 3000 I=1,3
      RELARY(I)=RELARY(I+1)
3000  CONTINUE
C
      CALL RT2ND(RELARY(1),QUOTRT(1))
C
30    IF(AIMAG(QUOTRT(1)).EQ.0.0.AND.
     1AIMAG(QUOTRT(2)).EQ.0.0) GO TO 40
      BROOT=ROOT
      GO TO 50
C
C FIND THE LARGEST REAL ROOT:-
40    BROOT=AMAX1(ROOT,REAL(QUOTRT(1)),REAL(QUOTRT(2)))
50    AFOUR=0.5*ARRAY(4)
      BFOUR=BROOT*0.5
```

```
      DFOUR=0.25*BROOT*BROOT-ARRAY(1)
      IF(DFOUR) 60,70,70
60    DFOUR=0.0
      GO TO 80
C
70    DFOUR=SQRT(ABS(DFOUR))
      IF(DFOUR) 90,80,90
80    CFOUR=AFOUR*AFOUR-ARRAY(3)+BROOT
      CFOUR=SQRT(ABS(CFOUR))
      GO TO 100
90    CFOUR=(AFOUR*BFOUR-ARRAY(2)*0.5)/DFOUR
C
100   G(3)=1.0
      G(2)=AFOUR+CFOUR
      G(1)=BFOUR+DFOUR
      CALL RT2ND(G(1),ROOTS(1))
C
210   G(2)=AFOUR-CFOUR
      G(1)=BFOUR-DFOUR
      CALL RT2ND(G(1),ROOTS(3))
C
      RETURN
      END
```

```
C POLYLIB
      SEGMENT 10
      REPEAT INTRINSICS
      SUBROUTINE POLMLT(NA,COEFFA,NB,COEFFB)
C MULTIPLIES 2 POLYNOMIALS TOGETHER.
C COEFFA & COEFFB ARE THE INPUT POLYNOMIAL
C  COEFFICIENT ARRAYS.
C COEFFA ALSO RETURNS THE PRODUCT ARRAY. THUS COEFFA WILL BE
C SUPERCEDED.
C
C NA & NB ARE THE POLYNOMIAL ORDERS
C                            ------
C
      DIMENSION COEFFA(1),COEFFB(1),COEFFC(256)
C
C CALCULATE OUTPUT ARRAY ORDER
      NC=NA+NB
      NPOLYA=NA+1
      NPOLYB=NB+1
      NPOLYC=NC+1
C
C CLEAR OUTPUT ARRAY
      DO 3000 I=1,NPOLYC
      COEFFC(I)=0.0
3000  CONTINUE
C
C NOW MULTIPLY ARRAYS
      DO 3010 IA=1,NPOLYA
      IREVA=NPOLYA-IA+1
      DO 3020 IB=1,NPOLYB
      IREVB=NPOLYB-IB+1
      IREVC=IREVA+IREVB-1
      COEFFC(IREVC)=COEFFA(IREVA)*COEFFB(IREVB)+
     1COEFFC(IREVC)
3020  CONTINUE
3010  CONTINUE
C
C COPY COEFFC INTO COEFFA
      DO 3100 IC=1,NPOLYC
      COEFFA(IC)=COEFFC(IC)
3100  CONTINUE
      NA=NC
C
      RETURN
      END
```

```
      SUBROUTINE POLDIV(NORNUM,ARRNUM,NORDEN,ARRDEN,
     1NORQNT,ARRQNT)
C DIVIDES 'ARRNUM' BY 'ARRDEN'
C NOTE :- ARRAY 'ARRNUM' WILL BE DESTROYED IN THIS ROUTINE
C
      DIMENSION ARRNUM(1),ARRDEN(1),ARRQNT(1)
C
      NORQNT=NORNUM-NORDEN
      IF(NORQNT) 200,10,10
10    N=NORQNT+1
C
      DO 20 I=1,N
      J=N+1-I
      ARRQNT(J)=QUOTNT(NORNUM,ARRNUM,NORDEN,ARRDEN)
20    CONTINUE
C
200   RETURN
      END
```

```
      REAL FUNCTION REPDIV(NORNUM,ARRNUM,NORDEN,ARRDEN)
C PERFORMS REPEATED DIVISION USING 'QUOTNT'
C
      DIMENSION ARRNUM(1),ARRDEN(1)
C
C THE NUMBER OF NUMERATOR COEFFICIENTS MUST BE
C KEPT CONSTANT, HENCE THE NUMERATOR POLYNOMIAL
C MUST BE SHIFTED UP ONE ELEMENTC AFTER EACH
C ITERATION & THE ORDER INCREMENTED.
C
C FIRST FIND THE QUOTIENT :
      RESULT=QUOTNT(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1))
C
C NEXT INCREMENT THE NUMERATOR ORDER
      NORNUM=NORNUM+1
C
C NOW SHIFT THE ARRAY
      DO 10 I=1,NORNUM
      J=NORNUM+1-I
      ARRNUM(J+1)=ARRNUM(J)
10    CONTINUE
C
      ARRNUM(1)=0.0
      REPDIV=RESULT
      RETURN
      END
```

```
      REAL FUNCTION QUOTNT(NORNUM,ARRNUM,NORDEN,ARRDEN)
C PERFORMS ONE ITERATION IN POLYNOMIAL DIVISION
C ARRNUM IS RETURNED AS THE QUOTIENT POLYNOMIAL.
C
      DIMENSION ARRNUM(1),ARRDEN(1)
C
      NCFNUM=NORNUM+1
      NCFDEN=NORDEN+1
C
C FIRST CALCULATE QUOTIENT
      RESULT=ARRNUM(NCFNUM)/ARRDEN(NCFDEN)
      NORNUM=NORNUM-1
      IF(NORNUM.LT.0) NORNUM=0
C
C NOW SUBTRACT RESULT*(DENOMINATOR POLYNOMIAL) FROM
C NUMERATOR POLYNOMIAL
      N=NCFNUM
      IF(NCFNUM.LE.NCFDEN) N=NCFDEN
      DO 10  I=2,N
      ICFNUM=NCFNUM+1-I
      ICFDEN=NCFDEN+1-I
      J=N+1-I
      VALUEN=0.0
      IF(ICFNUM.GT.0) VALUEN=ARRNUM(ICFNUM)
      VALUED=0.0
      IF(ICFDEN.GT.0) VALUED=ARRDEN(ICFDEN)
      ARRNUM(J)=VALUEN-RESULT*VALUED
10    CONTINUE
      ARRNUM(N)=0.0
C
      QUOTNT=RESULT
      RETURN
      END
```

```
      SUBROUTINE POLDIF(NORDER,ARRAY)
C DIFFERENTIATES POLYNOMIAL ARRAY.
C
      DIMENSION ARRAY(1)
C
      IF(NORDER) 30,30,10
C
10    DO 20 I=1,NORDER
      ARRAY(I)=ARRAY(I+1)*FLOAT(I)
20    CONTINUE
C
30    NORDER=NORDER-1
      IF(NORDER.LT.0) NORDER=0
      RETURN
      END
```

```
      SUBROUTINE POLYCX(NORDER,COEFFS,ROOTS,CHECK)
C EVALUATES POLYNOMIAL FROM COEFFS & ROOTS AND RETURNS
C CHECK CONTAINING THE RESULTS.
C
C NORDER  - POLYNOMIAL ORDER
C
      COMPLEX ROOT,ROOTS,CHECK
      DIMENSION COEFFS(1),ROOTS(1),CHECK(1)
C
      NUMELS=NORDER+1
      DO 10 I=1,NORDER
      CHECK(I)=(0.0,0.0)
      ROOT=ROOTS(I)
      RMAG=CABS(ROOT)
      THETA=ARCTAN(AIMAG(ROOT),REAL(ROOT))
      DO 20 J=1,NUMELS
      THETA=THETA*FLOAT(J-1)
      ROOT=(1.0,0.0)
      IF(THETA.NE.0.0) ROOT=CEXP(CMPLX(0.0,THETA))
      R=RMAG
      IF(R.NE.0.0) R=R**(J-1)
      CHECK(I)=CHECK(I)+CMPLX(COEFFS(J)*R,0.0)*ROOT
20    CONTINUE
10    CONTINUE
      RETURN
      END
```

```
      REAL FUNCTION ARCTAN(YREAL,XREAL)
C FINDS ATAN2 OF ARGUMENTS, COVERING SITUATION
C WHEN BOTH OF THESE ARGUMENTS ARE ZERO.
C
      RESULT=0.0
      IF(XREAL.NE.0.0.OR.
     1YREAL.NE.0.0) RESULT=ATAN2(YREAL,XREAL)
      ARCTAN=RESULT
      RETURN
      END
```

```
      SUBROUTINE CHCKRT(NUMRTS,ROOTS,ICHECK)
C CHECKS COMPLEX ROOTS TO SEE IF THEY
C ARE COMPLEX CONJUGATE. ICHECK CARRIES THE ROOT TYPE THUS :
C ICHECK(I) = 1 CONJUGATE ROOT NOT TO BE PRINTED
C ICHECK(I) = 2 NORMAL ROOT
C ICHECK(I) = 3 CONJUGATE ROOT TO BE PRINTED
C
      COMPLEX ROOTS
      DIMENSION ICHECK(1),ROOTS(1)
C
C CLEAR ICHECK
      DO 10 I=1,4
      ICHECK(I)=2
10    CONTINUE
C
C NOW CHECK THE ROOTS
      M2=NUMRTS-1
      DO 20 I=1,M2
      M1=I+1
      DO 30 J=M1,NUMRTS
      IF(ICHECK(J).NE.2) GO TO 30
      IF(ROOTS(I)-CONJG(ROOTS(J))) 30,40,30
C
C CONJUGATE ROOTS
40    ICHECK(I)=3
      ICHECK(J)=1
30    CONTINUE
20    CONTINUE
      RETURN
      END
```

```
C FREQLIB
      SEGMENT 11
      REPEAT INTRINSICS
      SUBROUTINE FQRESP(POLY,NUMPTS)
C CALCULATES FREQUENCY RESPONSE & STORES RESULTS IN DISK
C FILE ON UNIT 6.
C
      COMPLEX TSTANG,VALUEN,VALUED
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
      COMMON /COEFFS/ ARRNUM(60),NORNUM,ARRDEN(60),NORDEN
      COMMON /ANGLES/ PI,PI2
      COMMON /FQAXIS/ FQMIN,FQMAX,FQINC
      DATA SIGMA /0.0/
      DATA IZERO,NEND /0,9/
C
C START FREQUENCY SCAN LOOP
      TESTFQ=FQMIN
100   DO 200 IFREQ=1,NUMPTS
      TSTANG=CMPLX(0.0,TESTFQ*PI2)
C
C EVALUATE NUMERATOR POLYNOMIAL :
110   CALL POLY(NORNUM,ARRNUM(1),TSTANG,VALUEN)
C
C EVALUATE DENOMINATOR POLYNOMIAL :
120   CALL POLY(NORDEN,ARRDEN(1),TSTANG,VALUED)
C
      VALUEN=VALUEN/VALUED
C
C DISK STORAGE :
C FIRST WRITE THE STATUS
      WRITE(6) IZERO
C THEN THE DATA
      WRITE(6) VALUEN,TESTFQ
C
      GO TO (180,190),ISCAN
180   TESTFQ=TESTFQ*FQINC
      GO TO 200
190   TESTFQ=TESTFQ+FQINC
C
200   CONTINUE
C
C WRITE END STATUS
      WRITE(6) NEND
      RETURN
      END
```

```
      SUBROUTINE ZPOLY(NORDER,COEFFS,TESTFQ,ZVALUE)
C ZPOLY TAKES TESTFQ & EVALUATES VALUE OF POLYNOMIAL, ZVALUE
C
C TSAMP  - SAMPLE PERIOD (T)
C COEFFS - COEFFICIENTS OF Z-PLANE POLYNOMIAL
C
      COMPLEX ANGLE,TESTFQ,ZVALUE
      DIMENSION COEFFS(1)
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      EQUIVALENCE (PARNEW(1),TSAMP)
C
C EVALUATE ANGLE :
      ANGLE=CMPLX(TSAMP,0.0)*TESTFQ
C
C INITIALISE :
      ZVALUE=CMPLX(COEFFS(1),0.0)
C
C TEST ORDER
      IF(NORDER) 20,20,10
C
C NOW EVALUATE ZVALUE :
10    DO 100 I=1,NORDER
      ZVALUE=ZVALUE+CMPLX(COEFFS(I+1),0.0)*
     1 CEXP(ANGLE*CMPLX(FLOAT(I),0.0))
100   CONTINUE
20    RETURN
      END
```

```
      SUBROUTINE SPOLY(NORDER,COEFFS,TESTFQ,SVALUE)
C SPOLY TAKES TESTFQ & EVALUATES SVALUE.
C COEFFS - COEFFICIENTS OF S-PLANE POLYNOMIAL
C
      COMPLEX TESTFQ,SVALUE
      DIMENSION COEFFS(1)
C
C INITIALISE :
      SVALUE=CMPLX(COEFFS(1),0.0)
      ANGLE=ARCTAN(AIMAG(TESTFQ),REAL(TESTFQ))
C
C TEST ORDER
      IF(NORDER) 20,20,10
C
C NOW EVALUATE SVALUE
10    DO 100 I=1,NORDER
      SVALUE=SVALUE+CMPLX(COEFFS(I+1)*CABS(TESTFQ)**I,0.0)*
     1 CEXP(CMPLX(0.0,ANGLE*FLOAT(I)))
100   CONTINUE
20    RETURN
      END
```

```
C RESPLIB
      SEGMENT 12
      REPEAT INTRINSICS,EXTERNALS
      SUBROUTINE BODEM
C PLOTS & LABELS AXES FOR BODE MAGNITUDE PLOT
C
      COMMON /LABELS/ LOGF(13),LINF(11),LOGAMP(11),
     1LINAMP(8),LPHAD(8),LPHAR(8)
      COMMON /ORIGIN/ XORGB,YORGB,XORGP,YORGP,XORGN,YORGN
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
      COMMON /SCALES/ XORG,YORG,SIZE,SCALEX,SCALEY,
     1PLTSIZ,IPLANE
      COMMON /MAGAX/ AMPMIN,AMPMAX,AMPINC,IBPLOT
      COMMON /FQAXBM/ FQMINB,FQMAXB,FQINCB
      DATA ZERO,NINETY,NEND /0.0,90.0,9/
C
C INITIALISE PLOT POSITION
      CALL PLOT(XORGB,YORGB,-3)
C & PLOT SIZE
      SIZEX=SCALEX*SIZE
C
C NOW DRAW THE HORIZONTAL AXIS
      GO TO (100,200),IFBM
C LOGARITHMIC
100   IF(FQMINB.LE.0.0) FQMINB=1.0
      FQINCB=ALOG10(FQMAXB/FQMINB)/SIZEX
      CALL LOGAX(ZERO,ZERO,SIZEX,FQMINB,FQMAXB,
     1ZERO,LOGF(1),26)
      GO TO 300
C
C LINEAR
200   FQINCB=(FQMAXB-FQMINB)/SIZEX
      CALL AXIS(ZERO,ZERO,SIZEX,ZERO,FQMINB,FQINCB,
     1LINF(1),21)
C
C DRAW VERTICAL AXIS
300   SIZEY=SCALEY*SIZE
      GO TO (400,500),IMBM
C LOGARITHMIC
400   IF(AMPMIN.LE.0.0) AMPMIN=1.0
      AMPINC=ALOG10(AMPMAX/AMPMIN)/SIZEY
      CALL LOGAX(ZERO,ZERO,SIZEY,AMPMIN,AMPMAX,
     1NINETY,LOGAMP(1),-21)
      GO TO 600
C
C LINEAR
500   AMPINC=(AMPMAX-AMPMIN)/SIZEY
      CALL AXIS(ZERO,ZERO,SIZEY,NINETY,AMPMIN,
     1AMPINC,LINAMP(1),-16)
C RESET TO ORIGINAL ORIGIN
```

```
600     CALL PLOT(-XORGB,-YORGB,-3)
        RETURN
        END
```

```
      SUBROUTINE BODEP
C PLOTS & LABELS AXES FOR BODE PHASE PLOT
C
      COMMON /LABELS/ LOGF(13),LINF(11),LOGAMP(11),
     1LINAMP(8),LPHAD(8),LPHAR(8)
      COMMON /ORIGIN/ XORGB,YORGB,XORGP,YORGP,XORGN,YORGN
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
      COMMON /SCALES/ XORG,YORG,SIZE,SCALEX,SCALEY,
     1PLTSIZ,IPLANE
      COMMON /PHASAX/ PHAMIN,PHAMAX,PHAINC,IPPLOT
      COMMON /FQAXBP/ FQMINP,FQMAXP,FQINCP
      DATA ZERO,NINETY,NEND /0.0,90.0,9/
C
C INITIALISE PLOT POSITION
      CALL PLOT(XORGP,YORGP,-3)
C & PLOT SIZE
      SIZEX=SCALEX*SIZE
C
C NOW DRAW THE HORIZONTAL AXIS
      GO TO (100,200),IFBP
C LOGARITHMIC
100   IF(FQMINP.LE.0.0) FQMINP=1.0
      FQINCP=ALOG10(FQMAXP/FQMINP)/SIZEX
      CALL LOGAX(ZERO,ZERO,SIZEX,FQMINP,FQMAXP,
     1ZERO,LOGF(1),26)
      GO TO 300
C
C LINEAR
200   FQINCP=(FQMAXP-FQMINP)/SIZEX
      CALL AXIS(ZERO,ZERO,SIZEX,ZERO,FQMINP,FQINCP,
     1LINF(1),21)
C
C DRAW VERTICAL AXIS
300   SIZEY=SCALEY*SIZE
      HALFY=SIZEY/2.0
      PHAINC=PHAMAX/HALFY
      GO TO (310,320),IPDR
C PHASE IN DEGREES
310   CALL AXIS(ZERO,-HALFY,SIZEY,NINETY,-PHAMAX,
     1PHAINC,LPHAD(1),-15)
      GO TO 600
C PHASE IN RADIANS
320   CALL AXIS(ZERO,-HALFY,SIZEY,NINETY,-PHAMAX,
     1PHAINC,LPHAR(1),-15)
C
C RESET TO ORIGINAL ORIGIN
600   CALL PLOT(-XORGP,-YORGP,-3)
      RETURN
      END
```

```
      SUBROUTINE NYQST
C PLOTS & LABELS AXES FOR NYQUIST PLOT
C
      DIMENSION IMAGIN(5)
      COMMON /LABELS/ LOGF(13),LINF(11),LOGAMP(11),
     1LINAMP(8),LPHAD(8),LPHAR(8)
      COMMON /ORIGIN/ XORGB,YORGB,XORGP,YORGP,XORGN,YORGN
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
      COMMON /SCALES/ XORG,YORG,SIZE,SCALEX,SCALEY,
     1PLTSIZ,IPLANE
      COMMON /NYQUAX/ REMAX,RIMAX,RINCX,RINCY,INPLOT
      DATA ZERO,NINETY,NEND /0.0,90.0,9/
      DATA IMAGIN /2HIM,2HAG,2HIN,2HAR,2HY /
C
C INITIALISE PLOT POSITION
      CALL PLOT(XORGN,YORGN,-3)
C & PLOT SIZE
      SIZEX=SCALEX*SIZE
      SIZEY=SCALEY*SIZE
      HALFX=SIZEX/2.0
      HALFY=SIZEY/2.0
C
C NOW DRAW THE HORIZONTAL AXIS
      RINCX=REMAX/HALFX
      CALL AXIS(-HALFX,ZERO,SIZEX,ZERO,-REMAX,RINCX,1H ,1)
C & LABEL THIS AXIS
      CALL SYMBOL(HALFX,-1.0,0.4,'REAL',0.0,4)
C
C DRAW VERTICAL AXIS
      RINCY=RIMAX/HALFY
      CALL AXIS(ZERO,-HALFY,SIZEY,NINETY,-RIMAX,
     1RINCY,1H ,-1)
C & LABEL THIS AXIS
      CALL SYMBOL(0.3,HALFY,0.4,IMAGIN(1),0.0,10)
C
C RESET TO ORIGINAL ORIGIN
600   CALL PLOT(-XORGN,-YORGN,-3)
      RETURN
      END
```

```
      SUBROUTINE LOGAX(XSTRT,YSTRT,AXLTH,RMIN,RMAX,
     1THETA,IBCD,N)
C DRAWS LOGARITHMIC AXES ON GRAPH PLOTTER
C
      DIMENSION IBCD(1)
      DATA GRAD,RLABEL,DEGRAD,PIBY2 /0.2,0.5,0.01745328,
     11.570795/
      DATA WIDLET,SHIFT /0.33548387,0.16666666/
C
C INITIALISE
      SIDE=FLOAT(ISIGN(1,N))
      NUMPTS=IABS(N)
C CALCULATE INITIAL & FINAL DECADES
      RINC=ALOG10(RMAX/RMIN)
      ILAST=IFIX(RINC)
C CORRECT FOR ROUND-OFF ERROR
      IF(RINC.GT.AINT(RINC)) ILAST=ILAST+1
      IFIRST=IFIX(ALOG10(RMIN))
      ILAST=IFIRST+ILAST
C CALCULATE DECADE SIZE
      DECSIZ=AXLTH/RINC
C
C INITIALISE PLOT POSITION
      CALL PLOT(XSTRT,YSTRT,3)
      ANGLE=THETA*DEGRAD
      COSANG=COS(ANGLE)
      SINANG=SIN(ANGLE)
      RITANG=ANGLE+SIDE*PIBY2
      XSIDE=COS(RITANG)
      YSIDE=SIN(RITANG)
      XLABEL=XSIDE*RLABEL
      YLABEL=YSIDE*RLABEL
      XGRAD=XSIDE*GRAD
      YGRAD=YSIDE*GRAD
C
C NOW DRAW AXIS
      DO 10 I=IFIRST,ILAST
      DECADE=10.0**I
C CALCULATE LABEL SHIFT
      DECSFT=SHIFT*FLOAT(I+1)+0.075
      XSHIFT=DECSFT*COSANG
      YSHIFT=DECSFT*SINANG
C
C NOW PLOT THE POINTS IN ONE DECADE
      DO 20 J=1,10
      POINT=DECADE*FLOAT(J)
      IF(POINT.LT.RMIN.OR.POINT.GT.1.01*RMAX) GO TO 20
      R=DECSIZ*(ALOG10(POINT/RMIN))
      X=XSTRT+R*COSANG
      Y=YSTRT+R*SINANG
```

```
      CALL PLOT(X,Y,2)
      CALL PLOT(X-XGRAD,Y-YGRAD,2)
      CALL PLOT(X,Y,3)
      IF(J.GT.1) GO TO 20
      CALL NUMBER(X-XLABEL-XSHIFT,Y-YLABEL-YSHIFT,
     10,2,DECADE,THETA,2)
      CALL PLOT(X,Y,3)
20    CONTINUE
10    CONTINUE
C
C NOW LABEL AXIS
      OFFSET=(AXLTH-WIDLET*FLOAT(NUMPTS))/2.0
      X=XSTRT+COSANG*OFFSET-XSIDE
      Y=YSTRT+SINANG*OFFSET-YSIDE
      CALL SYMBOL(X,Y,0.4,IBCD,THETA,NUMPTS)
C
      RETURN
      END
```

```
      SUBROUTINE GRAPLT(GRAPH,X,Y)
C READS DISK FILE & DRAWS GRAPH.
C 'GRAPH' IS THE DUMMY NAME FOR THE REAL PLOT ROUTINE, VIZ:
C MAGPLT
C PHAPLT
C NYQPLT
C
      COMPLEX POINT
      DATA NEND /9/
C
      CALL PLOT(X,Y,-3)
C
      NTYP=3
C
C READ STATUS WORD
10    READ(6) ISTOP
C & CHECK
      IF(ISTOP.EQ.NEND) GO TO 40
C
C NOW READ THE VALUES
      READ(6) POINT,TESTFQ
      CALL GRAPH(POINT,TESTFQ,NTYP)
      NTYP=2
      GO TO 10
C
40    CALL PLOT(-X,-Y,-3)
      RETURN
      END
```

```
      SUBROUTINE MAGPLT(POINT,TESTFQ,N)
C BODE MAGNITUDE PLOT
C
      LOGICAL PLTPTS
      COMPLEX POINT
      COMMON /COORDS/ PLTPTS,PTSIZE,PTANG,IPTTYP
      COMMON /FQAXBM/ FQMINB,FQMAXB,FQINCB
      COMMON /MAGAX/ AMPMIN,AMPMAX,AMPINC
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
C
C CALCULATE MAX & MIN FREQUENCIES
      FREQ=AMAX1(TESTFQ,FQMINB)
      FREQ=AMIN1(FREQ,FQMAXB)
C CALCULATE MAX & MIN AMPLITUDES
      RMAG=AMAX1(CABS(POINT),AMPMIN)
      RMAG=AMIN1(RMAG,AMPMAX)
C
      GO TO (110,120),IFBM
C LOGARITHMIC FREQUENCY
110   FREQ=ALOG10(FREQ/FQMINB)/FQINCB
      GO TO 130
C LINEAR FREQUENCY
120   FREQ=FREQ/FQINCB
130   GO TO (210,220),IMBM
C LOGARITHMIC AMPLITUDE
210   RMAG=ALOG10(RMAG/AMPMIN)/AMPINC
      GO TO 230
C LINEAR MAGNITUDE
220   RMAG=RMAG/AMPINC
230   CALL PLOT(FREQ,RMAG,N)
C
C CHECK IF POINT TO BE MARKED
      IF(PLTPTS) CALL SYMBOL(FREQ,RMAG,PTSIZE,1H ,PTANG,IPTTYP)
C
      RETURN
      END
```

```
      SUBROUTINE PHAPLT(POINT,TESTFQ,N)
C BODE PHASE PLOT
C
      COMPLEX POINT
      LOGICAL PLTPTS
      COMMON /COORDS/ PLTPTS,PTSIZE,PTANG,IPTTYP
      COMMON /ANGLES/ PI,PI2
      COMMON /FQAXBP/ FQMINP,FQMAXP,FQINCP
      COMMON /PHASAX/ PHAMIN,PHAMAX,PHAINC,IPPLOT
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
C
C CALCULATE MAX & MIN FREQUENCIES
      FREQ=AMAX1(TESTFQ,FQMINP)
      FREQ=AMIN1(FREQ,FQMAXP)
C
      GO TO (110,120),IFBP
C LOGARITHMIC FREQUENCY
110   FREQ=ALOG10(FREQ/FQMINP)/FQINCP
      GO TO 130
C LINEAR FREQUENCY
120   FREQ=FREQ/FQINCP
C
C NOW CALULATE THE PHASE
130   PHASE=ARCTAN(AIMAG(POINT),REAL(POINT))/PHAINC
      IF(IPDR.EQ.1) PHASE=PHASE*180.0/PI
      CALL PLOT(FREQ,PHASE,N)
C
C CHECK IF POINT TO BE MARKED
      IF(PLTPTS) CALL SYMBOL(FREQ,PHASE,PTSIZE,1H ,PTANG,IPTTYP)
C
      RETURN
      END
```

```
      SUBROUTINE NYQPLT(POINT,TESTFQ,N)
C NYQUIST PLOT
C
      LOGICAL PLTPTS
      COMPLEX POINT
      COMMON /COORDS/ PLTPTS,PTSIZE,PTANG,IPTTYP
      COMMON /NYQUAX/ REMAX,RIMAX,RINCX,RINCY,INPLOT
C
      X=REAL(POINT)/RINCX
      Y=AIMAG(POINT)/RINCY
      CALL PLOT(X,Y,N)
C
C CHECK IF POINT TO BE MARKED
      IF(PLTPTS) CALL SYMBOL(X,Y,PTSIZE,1H ,PTANG,IPTTYP)
C
      RETURN
      END
```

```
C FRALIB
      SEGMENT 13
      REPEAT INTRINSICS
      SUBROUTINE SETUP
C GETS PLOTTING PARAMETERS
C
      INTEGER YES
      DIMENSION IGRAPH(3)
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      COMMON /SCALES/ XORG,YORG,SIZE,SCALEX,SCALEY,
     1PLTSIZ,IPLANE
      COMMON /LABELS/ LOGF(13),LINF(11),LOGAMP(11),
     1LINAMP(8),LPHAD(8),LPHAR(8)
      COMMON /ORIGIN/ XORGB,YORGB,XORGP,YORGP,XORGN,YORGN
      COMMON /FQAXBM/ FQMINB,FQMAXB,FQINCB
      COMMON /FQAXBP/ FQMINP,FQMAXP,FQINCP
      COMMON /MAGAX/ AMPMIN,AMPMAX,AMPINC,IBPLOT
      COMMON /PHASAX/ PHAMIN,PHAMAX,PHAINC,IPPLOT
      COMMON /NYQUAX/ REMAX,RIMAX,RINCX,RINCY,INPLOT
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
C
      DATA IGRAPH /2HGR,2HAP,2HH /
      DATA YES,NO /1HY,1HN/
      DATA GAP /1.5/
C
C SET OVERALL ORIGIN
      XORG=10.0
      YORG=10.0
      CALL PLOT(XORG,YORG,-3)
C
C INITIALISE PARAMETERS
      SIZE=10.0
      WRITE(3,100)
100   FORMAT(' GIVE HORIZONTAL & VERTICAL SCALES :'/)
      READ(2,3210) SCALEX,SCALEY
      SCALEX=ABS(SCALEX)
      SCALEY=ABS(SCALEY)
C
      PLTSIZ=0.2
C
C SET UP AXIS LABEL ARRAYS
      LOGF(1)='LO'
      LOGF(2)='GA'
      LOGF(3)='RI'
      LOGF(4)='TH'
      LOGF(5)='MI'
      LOGF(6)='C '
      LOGF(7)='FR'
      LOGF(8)='EQ'
      LOGF(9)='UE'
```

```
      LOGF(10)='NC'
      LOGF(11)='Y '
      LOGF(12)='(H'
      LOGF(13)='Z)'
C
      LINF(1)='LI'
      LINF(2)='NE'
      LINF(3)='AR'
      LINF(4)=' F'
      LINF(5)='RE'
      LINF(6)='QU'
      LINF(7)='EN'
      LINF(8)='CY'
      LINF(9)=' ('
      LINF(10)='HZ'
      LINF(11)=') '
C
      LOGAMP(1)='LO'
      LOGAMP(2)='GA'
      LOGAMP(3)='RI'
      LOGAMP(4)='TH'
      LOGAMP(5)='MI'
      LOGAMP(6)='C '
      LOGAMP(7)='AM'
      LOGAMP(8)='PL'
      LOGAMP(9)='IT'
      LOGAMP(10)='UD'
      LOGAMP(11)='E '
C
      LINAMP(1)='LI'
      LINAMP(2)='NE'
      LINAMP(3)='AR'
      LINAMP(4)=' A'
      LINAMP(5)='MP'
      LINAMP(6)='LI'
      LINAMP(7)='TU'
      LINAMP(8)='DE'
C
      LPHAD(1)='PH'
      LPHAD(2)='AS'
      LPHAD(3)='E '
      LPHAD(4)='(D'
      LPHAD(5)='EG'
      LPHAD(6)='RE'
      LPHAD(7)='ES'
      LPHAD(8)=') '
C
      LPHAR(1)='PH'
      LPHAR(2)='AS'
      LPHAR(3)='E '
      LPHAR(4)='(R'
```

```
      LPHAR(5)='AD'
      LPHAR(6)='IA'
      LPHAR(7)='NS'
      LPHAR(8)=') '
C
C CALCULATE AXIS LENGTH
      SIZEX=SCALEX*SIZE
      SIZEY=SCALEY*SIZE
C
10    WRITE(3,200)
200   FORMAT(' BODE MAGNITUDE PLOT? (Y/N) :'/)
      READ(2,210) IBPLOT
210   FORMAT(A1)
      IF(IBPLOT.NE.YES.AND.IBPLOT.NE.NO) GO TO 10
      IF(IBPLOT.EQ.NO) GO TO 40
      XORGB=0.0
      YORGB=SIZEY+2.0+GAP
C
20    WRITE(3,220)
220   FORMAT(' LOG OR LIN FREQUENCY AXIS? (1/2) :'/)
      READ(2,3210) IFBM
3210  FORMAT()
      IF(IFBM.NE.1.AND.IFBM.NE.2) GO TO 20
C
      WRITE(3,240)
      READ(2,3210) FQMINB,FQMAXB
C
30    WRITE(3,230)
230   FORMAT(' LOG OR LIN MAGNITUDE AXIS? (1/2) :'/)
      READ(2,3210) IMBM
      IF(IMBM.NE.1.AND.IMBM.NE.2) GO TO 30
C
      WRITE(3,240)
240   FORMAT(' MINIMUM & MAXIMUM VALUE? :'/)
      READ(2,3210) AMPMIN,AMPMAX
      IF(IMBM.EQ.1.AND.AMPMIN.EQ.0.0) GO TO 30
C
C
40    WRITE(3,250)
250   FORMAT(' BODE PHASE PLOT? (Y/N) :'/)
      READ(2,210) IPPLOT
      IF(IPPLOT.NE.YES.AND.IPPLOT.NE.NO) GO TO 40
      IF(IPPLOT.EQ.NO) GO TO 70
      XORGP=0.0
      YORGP=0.5*SIZEY+2.0
C
50    WRITE(3,220)
      READ(2,3210) IFBP
      IF(IFBP.NE.1.AND.IFBP.NE.2) GO TO 50
C
      WRITE(3,240)
```

```
      READ(2,3210) FQMINP,FQMAXP
C
60    WRITE(3,260)
260   FORMAT(' PHASE IN DEGREES OR RADIANS? (1/2) :'/)
      READ(2,3210) IPDR
      IF(IPDR.LT.1.OR.IPDR.GT.2) GO TO 60
C
      WRITE(3,270)
270   FORMAT(' MAXIMUM PHASE ANGLE :'/)
      READ(2,3210) PHAMAX
      PHAMAX=ABS(PHAMAX)
      PHAMIN=0.0
C
C
70    WRITE(3,280)
280   FORMAT (' NYQUIST PLOT? (Y/N) :'/)
      READ(2,210) INPLOT
      IF(INPLOT.NE.YES.AND.INPLOT.NE.NO) GO TO 70
      IF(INPLOT.EQ.NO) GO TO 80
C
      XORGN=SIZEX
      IF(IPPLOT.EQ.YES.OR.IBPLOT.EQ.YES) XORGN=XORGN+10.0
      YORGN=0.5*SIZEY
      WRITE(3,290)
290   FORMAT(' MAXIMUM REAL & IMAGINARY VALUE :'/)
      READ(2,3210) REMAX,RIMAX
      REMAX=ABS(REMAX)
      RIMAX=ABS(RIMAX)
C
C
80    IF(IBPLOT.EQ.NO.AND.IPPLOT.EQ.NO.
     1AND.INPLOT.EQ.NO) GO TO 10
C
      IF(IBPLOT.EQ.YES) CALL BODEM
      IF(IPPLOT.EQ.YES) CALL BODEP
      IF(INPLOT.EQ.YES) CALL NYQST
C
C NOW WRITE GRAPH COMMENT
      CALL SYMBOL(0.0,1.0,0.4,ITITG(1),0.0,NTITG*2)
C
C NOW WRITE 'GRAPH'
      CALL SYMBOL(10.0,0.0,0.4,IGRAPH(1),0.0,6)
      RETURN
      END
```

```
      SUBROUTINE LPFST(IDEV)
C ROUTINE TO WRITE THE L/P HEADINGS
C
      COMMON /PLOTYP/ ISCAN,IFBM,IMBM,IFBP,IPDR
      COMMON /SCALES/ XORG,YORG,SIZE,SCALEX,SCALEY,PLTSIZ,
     1IPLANE
      COMMON /COEFFS/ ARRNUM(60),NORNUM,ARRDEN(60),NORDEN
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      EQUIVALENCE (PARNEW(1),TSAMP)
C
      WRITE(4,1000)
1000  FORMAT(1H1)
      IF(IDEV.EQ.1) WRITE(4,1010) ITITD
1010  FORMAT(1H1,72A1/)
      IF(IPLANE.EQ.1.OR.IDEV.EQ.3) WRITE(4,1030) PARNEW
1030  FORMAT(' SAMPLING PERIOD T :',E12.4,' SECONDS'/
     1' CAPACITOR C1       :',E12.4,' FARADS'/
     2' CAPACITOR C2       :',E12.4,' FARADS'/
     3' TRANSCONDANCE G1   :',E12.4,' SIEMENS'/
     4' TRANSCONDANCE G2   :',E12.4,' SIEMENS'/
     5' SHUNT RESISTANCE R:',F8.1,' OHMS'/)
      IF(IDEV.EQ.3) GO TO 30
C
      IF(ISCAN.EQ.2) WRITE(4,1040)
      IF(ISCAN.EQ.1) WRITE(4,1050)
      WRITE(4,1060)
      DO 10 L=0,NORNUM
      L1=NORNUM-L
      L2=L1+1
      WRITE(4,1100) L1,ARRNUM(L2)
1100  FORMAT(I4,4X,F12.6)
10    CONTINUE
C
      WRITE(4,1110)
      DO 20 L=0,NORDEN
      L1=NORDEN-L
      L2=L1+1
      WRITE(4,1100) L1,ARRDEN(L2)
20    CONTINUE
C
30    WRITE(4,1120)
1120  FORMAT(' RESULT FREQUENCY',29X,'COMPLEX VALUE'/
     1 1H ,6(1H-),1X,9(1H-),29X,7(1H-),1X,5(1H-)/
     2 1H ,2X,'NO.',6X,'HZ',11X,'REAL',7X,
     3'IMAGINARY',7X,'MODULUS',
     4 3X,'ARGAND(RAD)',3X,'ARGAND(DEG)'/)
C
C
1040  FORMAT(' LINEAR FREQUENCY SCAN'/)
```

```
1050  FORMAT(' LOGARITHMIC FREQUENCY SCAN'/)
1060  FORMAT(' NUMERATOR POLYNOMIAL COEFFICIENTS :'/)
1110  FORMAT(' DENOMINATOR POLYNOMIAL COEFFICIENTS :'/)
C
      RETURN
      END
```

```
      SUBROUTINE LPFOUT
C ROUTINE TO OUTPUT DATA FROM TEMPORARY DISK FILE TO
C L/P.
C
      COMPLEX POINT
      COMMON /ANGLES/ PI,PI2
C
      DATA NEND /9/
C
      IFQ=0
C
C NOW READ DISK FILE & OUTPUT THE RESULTS UNTIL
C  THE TERMINATING CHARACTER IS FOUND
C FIRST READ STATUS
10    READ(6) ISTOP
      IF(ISTOP.EQ.NEND) RETURN
C NOW READ THE DATA
      READ(6) POINT,TESTFQ
C
C CALCULATE THE REAL & IMAGINARY PARTS
      RREAL=REAL(POINT)
      RIMAG=AIMAG(POINT)
      RMOD=CABS(POINT)
      RARG=ARCTAN(RIMAG,RREAL)
      RDEG=RARG*180.0/PI
      IFQ=IFQ+1
C
C LINE-PRINTER OUTPUT :
1000  WRITE(4,2000) IFQ,TESTFQ,RREAL,RIMAG,RMOD,RARG,RDEG
2000  FORMAT(1H ,I5,1X,F10.2,5(2X,F12.4))
C
      GO TO 10
C
      END
```

```
C PZPLIB
      SEGMENT 15
      REPEAT INTRINSICS
C
      SUBROUTINE DISTRT
C READS HEADER OFF DISK FILE
C
      COMMON /POLYNS/ NDPOLY(60),NPOLY,MPOLY,IPYCNT
      COMMON /MATELM/ NUMRES,MATRIX,NELEMS
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
C
      MPOLY=60
      IPYCNT=0
      IF(NPOLY.GT.MPOLY) RETURN
      WRITE(3,300)
300   FORMAT(' GIVE POLYNOMIALS TO BE ANALYSED :'/)
      READ(2,200) ((NDPOLY(I)),I=1,NPOLY)
200   FORMAT()
C
C READ TITLES
      READ(5) NTITD,ITITD
C
C READ NUMBER OF RESULTS, MATRIX TYPE & ELEMENT
      READ(5) NUMRES,MATRIX,NELEMS
C
      RETURN
      END
```

```
      SUBROUTINE DISCIN(NORNUM,ARRNUM,NORDEN,ARRDEN,INPOLY)
C READS POLYNOMIAL COEFFICIENTS FROM DISC FILE
C
      INTEGER ENDFIL
      DIMENSION ARRNUM(1),ARRDEN(1)
      COMMON /POLYNS/ NDPOLY(60),NPOLY,MPOLY,IPYCNT
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      DATA ENDFIL /9/
C
C STORE OLD PARAMETERS
10    DO 20 IPAR=1,6
      PAROLD(IPAR)=PARNEW(IPAR)
20    CONTINUE
C
C READ STATUS WORD
      READ(5) ISTDSK
      IF(ISTDSK.NE.ENDFIL) GO TO 30
      IPYCNT=-1
      RETURN
C
C NOW READ PARAMETERS
30    READ(5) PARNEW
C
C NOW READ COEFFICIENTS
      READ(5) NORNUM,IPLOTA
      READ(5) ((ARRNUM(I+1)),I=0,NORNUM)
      READ(5) NORDEN,IPLOTB
      READ(5) ((ARRDEN(I+1)),I=0,NORDEN)
      IPYCNT=IPYCNT+1
      IF(IPYCNT.NE.INPOLY) GO TO 10
C
      RETURN
      END
```

```
      SUBROUTINE LPSTRT
C OUTPUTS HEADER TO L/P.
C
      COMMON /MATELM/ NUMRES,MATRIX,NELEMS
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
C
      WRITE(4,100) ((ITITD(I)),I=1,NTITD)
100   FORMAT(1H1,72A1)
      IF(NUMRES.EQ.0) RETURN
      WRITE(4,110) NUMRES,MATRIX,NELEMS
110   FORMAT(' NUMBER OF RESULTS =',I6/
     1' MATRIX ELEMENT ',A1,I2/)
C
      RETURN
      END
```

```
      SUBROUTINE DISCOP(NUMBER,ROOTS)
C OUTPUTS DATA TO BINARY DISK FILE.
C
      INTEGER ZERO
      COMPLEX ROOTS
      DIMENSION ROOTS(1)
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      DATA ZERO /0/
C
C FIRST WRITE STATUS WORD
      WRITE(6) ZERO
C
C NOW WRITE PARAMETERS
      WRITE(6) PARNEW
C
C NOW WRITE ROOTS
      WRITE(6) (ROOTS(I),I=1,NUMBER)
C
      RETURN
      END
```

```
      SUBROUTINE ASROUT(NUMBER,ROOTS,CHECK)
C OUTPUTS ROOTS TO ASR
C
      COMPLEX ROOTS,CHECK
      DIMENSION ROOTS(1),CHECK(1)
      COMMON /ANGLES/ PI,DEGRAD
C
      WRITE(3,100)
100   FORMAT(1HO,'CARTESIAN ROOTS :'/2X,
     1'REAL',16X,'IMAGINARY')
      WRITE(3,110) (ROOTS(I),I=1,NUMBER)
110   FORMAT(1H ,E14.6,4X,'J*',E14.6)
      WRITE(3,120)
120   FORMAT(1HO,'CHECK :')
C
      WRITE(3,110) CHECK
C
      WRITE(3,140)
140   FORMAT(1HO,'POLAR ROOTS :'/
     1 2X,'MAGNITUDE',11X,'RADIANS',13X,'DEGREES')
C
      DO 200 I=1,NUMBER
      R=CABS(ROOTS(I))
      ANGRAD=ARCTAN(AIMAG(ROOTS(I)),REAL(ROOTS(I)))
      ANGDEG=DEGRAD*ANGRAD
      WRITE(3,150) R,ANGRAD,ANGDEG
150   FORMAT(1H ,3(E14.6,6X))
200   CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE LPROOT(NORDER,ROOTS,ARRAY,ITYPE)
C OUTPUTS ROOTS TO L/P
C
      COMPLEX ROOTS
      DIMENSION ROOTS(1),ARRAY(1)
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      COMMON /ANGLES/ PI,DEGRAD
C
      IF(ITYPE.EQ.1) WRITE(4,100)
100   FORMAT(' NUMERATOR POLYNOMIAL ROOTS')
      IF(ITYPE.EQ.2) WRITE(4,110)
110   FORMAT(' DENOMINATOR POLYNOMIAL ROOTS')
      WRITE(4,120)
120   FORMAT('   REAL',16X,'IMAGINARY',
     17X,'MODULUS',9X,'ARG(DEG)')
      DO 200 I=1,NORDER
      R=CABS(ROOTS(I))
      X=REAL(ROOTS(I))
      Y=AIMAG(ROOTS(I))
      ANGDEG=DEGRAD*ARCTAN(Y,X)
      WRITE(4,140) X,Y,R,ANGDEG
140   FORMAT(1H ,E14.6,4X,'J*',3(E14.6,2X))
200   CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE LPARMS(PARAMS)
C ROUTINE TO OUTPUT PARAMETERS FOR DISK FILE INPUT
C
      DIMENSION PARAMS(6)
C
      WRITE(4,100) PARAMS
100   FORMAT(1H1,'PARAMETERS'/
     1' SAMPLING PERIOD   T = ',E10.4,2X,'SECONDS'/
     2' SHUNT CAPACITOR  C1 = ',E10.4,2X,'FARADS'/
     3' SHUNT CAPACITOR  C2 = ',E10.4,2X,'FARADS'/
     4' TRANSCONDUCTANCE G1 = ',F10.4,2X,'SIEMENS'/
     5' TRANSCONDUCTANCE G2 = ',F10.4,2X,'SIEMENS'/
     6' SHUNT RESISTOR    R = ',F10.4,2X,'OHMS'/)
      RETURN
      END
```

```
      SUBROUTINE LPCOEF(NORDER,ARRAY,ITYPE)
C ROUTINE TO OUTPUT COEFFICIENTS OF ARRAY BEING
C ANALYSED.
C
      DIMENSION ARRAY(1)
C
      IF(ITYPE.EQ.1) WRITE(4,100)
100   FORMAT(' NUMERATOR POLYNOMIAL COEFFICIENTS'/
     1' POWER',6X,'COEFFICIENT')
      IF(ITYPE.EQ.2) WRITE(4,110)
110   FORMAT(' DENOMINATOR POLYNOMIAL COEFFICIENTS'/
     1' POWER',6X,'COEFFICIENT')
C
      DO 10 I=0,NORDER
      J=NORDER-I
      WRITE(4,120) J,ARRAY(J+1)
120   FORMAT (1H ,3X,I2,4X,E14.6)
10    CONTINUE
C
      WRITE(4,130)
130   FORMAT(1H )
C
      RETURN
      END
```

```
      SUBROUTINE SETVDU
C SETS VDU FOR DATA OUTPUT
C
      CALL OPVDU
      RETURN
      END
```

```
      SUBROUTINE TKSTRT
C GET STARTING COORDINATES FOR WRITING DATA TO TEK.
C
      INTEGER YES,NO
      COMMON /PLOTYP/ NUMDEN,NAUTO,INPDEV,ITYPE
      COMMON /TEK/ IXVAL,IYVAL,LINPAG,LINCNT
C
      DATA MAXX,MINX,MAXY,MINY /1023,0,760,0/
      DATA LINMAX /40/
      DATA YES,NO /1HY,1HN/
C
10    WRITE(3,1010)
1010  FORMAT(' X-COORD =')
      READ(2,2010) IXVAL
2010  FORMAT()
      IF(IXVAL.LT.MINX.OR.IXVAL.GT.MAXX) GO TO 10
20    WRITE(3,1020)
1020  FORMAT(' Y-COORD =')
      READ(2,2010) IYVAL
      IF(IYVAL.LT.MINY.OR.IYVAL.GT.MAXY) GO TO 20
30    WRITE(3,1030)
1030  FORMAT(' NUMBER OF LINES/PAGE =')
      READ(2,2010) LINPAG
      IF(LINPAG.LE.0.OR.LINPAG.GT.LINMAX) GO TO 30
      LINCNT=0
      WRITE(3,1040)
1040  FORMAT(' AUTO-LISTING REQUIRED? (Y/N) :')
40    READ(2,2040) IAUTO
2040  FORMAT(A1)
      IF(IAUTO.NE.YES.AND.IAUTO.NE.NO) GO TO 40
      NAUTO=0
      IF(IAUTO.EQ.YES) NAUTO=1
      RETURN
      END
```

```
      SUBROUTINE VDUOUT(NORDER,ROOTS,CHECK,ARRAY)
C OUTPUTS TO TEKTRONIX TERMINAL.
C
      INTEGER YES,STAR,BOTH
      LOGICAL TERM,AUTORP,PERPH,NOTPER,PARWRT
      COMPLEX ROOTS,CHECK
      DIMENSION ARRAY(1),ROOTS(1),CHECK(1)
      DIMENSION IPRINT(4),MSCALE(4)
      DIMENSION MICRO(3),MILLI(3),IFARAD(3),ISEC(4),ICON(4)
      COMMON /ANGLES/ PI,DEGRAD
      COMMON /PLOTYP/ NUMDEN,NAUTO,INPDEV,ITYPE
      COMMON /PERPHS/ PERPH(5),NOTPER(5)
      COMMON /TEK/ IXVAL,IYVAL,LINPAG,LINCNT
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      DATA MICRO /2HMI,2HCR,2HO-/
      DATA MILLI /2HMI,2HLL,2HI-/
      DATA IFARAD /2HFA,2HRA,2HDS/
      DATA ISEC /2HSE,2HCO,2HND,2HS /
      DATA ICON /2HSI,2HEM,2HEN,2HS /
      DATA IUNIT,IDATPT /16,29/
      DATA YES,NO /1HY,1HN/
      DATA SMICRO,SMILLI /1.E6,1.E3/
      DATA NCON,NEND /0,9/
C
C CHECK ROOTS
      CALL CHCKRT(NORDER,ROOTS(1),IPRINT(1))
      IPAR=0
      IROOT=0
      PARWRT=.FALSE.
C
C CHECK IF NEW PAGE NEEDED
      IF(LINCNT.LE.0) CALL VDSTRT
C
C SKIP PARAMETER LISTING IF DISK FILE NOT BEING PROCESSED
      IF(INPDEV.EQ.2) GO TO 200
C
C SKIP PARAMETER LISTING FOR DENOMINATOR IF BOTH
C ARE BEING LISTED.
C       IF(NUMDEN.EQ.3.AND.IPLOT.EQ.2) GO TO 200
C
C COMPARE PARAMETERS; IF DIFFERENT THEN PRINT NEW VALUES
100   PARWRT=.FALSE.
      IPAR=IPAR+1
      IF(IPAR.GT.6.AND.IROOT.GT.NORDER) RETURN
      IF(IPAR.GT.6) GO TO 200
      IF(PARNEW(IPAR).EQ.PAROLD(IPAR)) GO TO 100
      PARWRT=.TRUE.
C
      GO TO (110,120,130,140,150,160),IPAR
110   CALL OUTST('TS = ',5,2)
```

```
      SUBROUTINE VDUOUT(NORDER,ROOTS,CHECK,ARRAY)
C OUTPUTS TO TEKTRONIX TERMINAL.
C
      INTEGER YES,STAR,BOTH
      LOGICAL TERM,AUTORP,PERPH,NOTPER,PARWRT
      COMPLEX ROOTS,CHECK
      DIMENSION ARRAY(1),ROOTS(1),CHECK(1)
      DIMENSION IPRINT(4),MSCALE(4)
      DIMENSION MICRO(3),MILLI(3),IFARAD(3),ISEC(4),ICON(4)
      COMMON /ANGLES/ PI,DEGRAD
      COMMON /PLOTYP/ NUMDEN,NAUTO,INPDEV,ITYPE
      COMMON /PERPHS/ PERPH(5),NOTPER(5)
      COMMON /TEK/ IXVAL,IYVAL,LINPAG,LINCNT
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
      DATA MICRO /2HMI,2HCR,2HO-/
      DATA MILLI /2HMI,2HLL,2HI-/
      DATA IFARAD /2HFA,2HRA,2HDS/
      DATA ISEC /2HSE,2HCO,2HND,2HS /
      DATA ICON /2HSI,2HEM,2HEN,2HS /
      DATA IUNIT,IDATPT /16,29/
      DATA YES,NO /1HY,1HN/
      DATA SMICRO,SMILLI /1.E6,1.E3/
      DATA NCON,NEND /0,9/
C
C CHECK ROOTS
      CALL CHCKRT(NORDER,ROOTS(1),IPRINT(1))
      IPAR=0
      IROOT=0
      PARWRT=.FALSE.
C
C CHECK IF NEW PAGE NEEDED
      IF(LINCNT.LE.0) CALL VDSTRT
C
C SKIP PARAMETER LISTING IF DISK FILE NOT BEING PROCESSED
      IF(INPDEV.EQ.2) GO TO 200
C
C SKIP PARAMETER LISTING FOR DENOMINATOR IF BOTH
C ARE BEING LISTED.
C       IF(NUMDEN.EQ.3.AND.IPLOT.EQ.2) GO TO 200
C
C COMPARE PARAMETERS; IF DIFFERENT THEN PRINT NEW VALUES
100   PARWRT=.FALSE.
      IPAR=IPAR+1
      IF(IPAR.GT.6.AND.IROOT.GT.NORDER) RETURN
      IF(IPAR.GT.6) GO TO 200
      IF(PARNEW(IPAR).EQ.PAROLD(IPAR)) GO TO 100
      PARWRT=.TRUE.
C
      GO TO (110,120,130,140,150,160),IPAR
110   CALL OUTST('TS = ',5,2)
```

```
      CALL OUTF(PARNEW(1)*SMICRO,2)
      CALL TAB(IUNIT)
      CALL OUTST(MICRO(1),6,2)
      CALL OUTST(ISEC(1),7,2)
      GO TO 200
120   CALL OUTST('C1 = ',5,2)
      CALL OUTF(PARNEW(2)*SMICRO,2)
      CALL TAB(IUNIT)
      CALL OUTST(MICRO(1),6,2)
      CALL OUTST(IFARAD(1),6,2)
      GO TO 200
130   CALL OUTST('C2 = ',5,2)
      CALL OUTF(PARNEW(3)*SMICRO,2)
      CALL TAB(IUNIT)
      CALL OUTST(MICRO(1),6,2)
      CALL OUTST(IFARAD(1),6,2)
      GO TO 200
140   CALL OUTST('G1 = ',5,2)
      CALL OUTF(PARNEW(4)*SMILLI,2)
      CALL TAB(IUNIT)
      CALL OUTST(MILLI(1),6,2)
      CALL OUTST(ICON(1),8,2)
      GO TO 200
150   CALL OUTST('G2 = ',5,2)
      CALL OUTF(PARNEW(5)*SMILLI,2)
      CALL TAB(IUNIT)
      CALL OUTST(MILLI(1),6,2)
      CALL OUTST(ICON(1),8,2)
      GO TO 200
160   CALL OUTST('RS = ',5,2)
      CALL OUTF(PARNEW(6),2)
      CALL TAB(IUNIT)
      CALL OUTST('OHMS',4,2)
C
C NOW OUTPUT ROOTS
200   IROOT=IROOT+1
      IF(IROOT.GT.NORDER.AND.PARWRT) GO TO 320
      IF(IROOT.GT.NORDER) GO TO 100
300   IF(IPRINT(IROOT).LE.1) GO TO 200
310   CALL TAB(IDATPT)
      CALL RTOUT(ROOTS(IROOT),IPRINT(IROOT))
320   CALL NEWLIN(1,1)
      LINCNT=LINCNT-1
      GO TO 100
      END
```

```
      SUBROUTINE RTOUT(ROOT,IPAIR)
C OUTPUTS REAL & IMAGINARY PARTS OF ROOT DEPENDING ON THE
C VALUE OF PAIR :
C IPAIR = 1 OMIT
C IPAIR = 2 PRINT NORMALLY
C IPAIR = 3 PRINT AS COMPLEX CONJUGATE
C
      COMPLEX ROOT
C
      GO TO (40,10,10),IPAIR
10    CALL TAB(32)
      CALL OUTF(REAL(ROOT),4)
      GO TO (40,30,20),IPAIR
C
20    CALL TAB(40)
      CALL OUTST('+/-',3,2)
30    CALL TAB(45)
      CALL OUTST('J*(',3,2)
      CALL OUTF(AIMAG(ROOT),4)
      CALL OUTST(')',1,2)
40    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VDSTRT
C OUTPUTS VDU PAGE HEADINGS FOR PZP1
C
      DIMENSION IPARMS(5),IREAL(2),IMAGIN(5)
      COMMON /PLOTYP/ NUMDEN,NAUTO,INPDEV,ITYPE
      COMMON /TEK/ IXVAL,IYVAL,LINPAG,LINCNT
      DATA IPARMS /2HPA,2HRA,2HME,2HTE,2HRS/
      DATA IREAL /2HRE,2HAL/
      DATA IMAGIN /2HIM,2HAG,2HIN,2HAR,2HY /
      DATA MALPHA /31/
C
C SET TO ALPHA MODE
      CALL CHOUT(31)
      IF(NAUTO.EQ.1) CALL COPY
C
C NOW CLEAR SCREEN
      CALL CLEAR
      WRITE(4,100)
100   FORMAT(' DONE')
C
C INITIALISE WRITING POINT
      CALL TPLOT(0,IXVAL,IYVAL)
      WRITE(4,100)
C
C NOW RESET TO ALPHA MODE
      CALL CHOUT(MALPHA)
      WRITE(4,100)
C
C NOW OUTPUT HEADINGS
      CALL SPACES(4)
      CALL OUTST(IPARMS(1),10,2)
      CALL TAB(30)
      CALL OUTST(IREAL(1),4,2)
      CALL SPACES(9)
      CALL OUTST(IMAGIN(1),10,2)
      CALL NEWLIN(2,1)
      LINCNT=LINPAG
C
      WRITE(4,100)
      RETURN
      END
```

```
      SUBROUTINE GPSTRT
C INITIALISES GRAPH PLOTTER.
C
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      COMMON /SCALES/ XORG,YORG,XYSIZE,XYSCAL,PTSIZE
C
C ASSIGN GRAPH PLOTTER TO ADDRESS 7
      CALL PLOTS(7)
110   WRITE(3,1100)
1100  FORMAT(' Z OR S PLANE? (1 OR 2) :'/)
      READ(2,2010) IPLANE
2010  FORMAT()
      IF(IPLANE.LT.1.OR.IPLANE.GT.2) GO TO 110
C
C GET SCALE & POLE/ZERO SIZE
120   WRITE(3,1110)
1110  FORMAT(' SCALE FACTOR & POLE/ZERO PLOT SIZE:'/)
      READ(2,2010) XYSCAL,PTSIZE
      IF(XYSCAL.LT.0.0.OR.PTSIZE.LT.0.0) GO TO 120
C
C NOW GET GRAPH TITLE
      WRITE(3,1120)
1120  FORMAT(' TITLE:'/)
      NTITG=MTITLE
      CALL GETLIN(2,ITITG,NTITG)
      XYSIZE=7.5
      CALL PLANE(16.0,16.0,IPLANE)
      RETURN
      END
```

```
      SUBROUTINE LISTLP(NORDER,ROOTS,CHECK,ARRAY)
C OUTPUTS RESULTS IN COMPRESSED FORM
C
      DIMENSION ROOTS(1),CHECK(1),ARRAY(1)
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
C
C OUTPUT PARAMETERS
      WRITE(4,100) PARNEW
100   FORMAT(' PARAMETERS'/1H ,E10.4,2X,'SECONDS'/
     11H ,E10.4,2X,'FARADS'/1H ,E10.4,2X,'FARADS'/
     21H ,F10.4,2X,'SIEMENS',1H ,F10.4,2X,'SIEMENS'/
     31H ,F10.4,2X,'OHMS'/)
      WRITE(4,200)
200   FORMAT(' NUMERATOR',20X,'DENOMINATOR'/
     11H ,'COEFFICIENTS',17X,'COEFFICIENTS')
      RETURN
      END
```

```
C POLENT
      SEGMENT 16
      REPEAT INTRINSICS,READWRITE
C
      SUBROUTINE KEYBIN(NORNUM,ARRNUM,NORDEN,ARRDEN,NUMDEN)
C INPUTS RATIONAL POLYNOMIAL.
C
      DIMENSION ARRNUM(1),ARRDEN(1)
C
C SWITCH ACCORDING TO POLYNOMIAL TYPE
      GO TO (100,110,100),NUMDEN
C
100   WRITE(3,300)
300   FORMAT(' NUMERATOR'/)
      CALL POLGET(NORNUM,ARRNUM(1))
C
      GO TO (120,110,110),NUMDEN
110   WRITE(3,305)
305   FORMAT(' DENOMINATOR'/)
      CALL POLGET(NORDEN,ARRDEN(1))
C
120   RETURN
      END
```

```
      SUBROUTINE POLGET(NORDER,ARRAY)
C GETS COMPLETE POLYNOMIAL WHICH CAN CONSIST OF FACTORS.
C
      DIMENSION ARRAY(1),COEFFS(40)
      DATA NUMCFS /40/
C
10    WRITE(3,1000)
1000  FORMAT(' HOW MANY FACTORS? :'/)
      READ(2,2000) NFACTS
2000  FORMAT()
      IF(NFACTS.GT.NUMCFS) GO TO 10
      IF(NFACTS) 10,10,20
C
C CLEAR COEFFICIENTS IN ARRAY
20    NORDER=0
      ARRAY(1)=1.0
C
C NOW GET EACH POLYNOMIAL ARRAY & MULTIPLY WITH THAT ALREADY
C OBTAINED.
C
      DO 3010 IPOLY=1,NFACTS
      CALL POLYIN(N,COEFFS(1))
      CALL POLMLT(NORDER,ARRAY(1),N,COEFFS(1))
3010  CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE POLYIN(NORDER,ARRAY)
C ROUTINE TO ENTER RANDOM POLYNOMIAL COEFFICIENTS
C
      DIMENSION ARRAY(1)
C
10    WRITE(3,1000)
1000  FORMAT(' GIVE FACTOR ORDER :'/)
      READ(2,2000) NORDER
2000  FORMAT()
C
      IF(NORDER) 10,20,20
20    NUMCFS=NORDER+1
C
C FIRST CLEAR ARRAY
      DO 3000 I=1,NUMCFS
      ARRAY(I)=0.0
3000  CONTINUE
C
C NOW GET POWERS & COEFFICIENTS.
C COEFFICIENTS OF UNSPECIFIED POWERS ARE
C AUTOMATICALLY SET TO ZERO.
C ENTRY MAY BE TERMINATED BY TYPING NEGATIVE POWER.
      DO 3010 I=1,NUMCFS
      WRITE(3,1010)
1010  FORMAT(' GIVE POWER & COEFFICIENT :'/)
30    READ(2,2000) IPOWER,VALUE
      IF(IPOWER) 50,40,40
40    IF(IPOWER.GT.NUMCFS) GO TO 30
      ARRAY(IPOWER+1)=VALUE
3010  CONTINUE
C
50    RETURN
      END
```

```
C IZTLIB
      SEGMENT 17
      REPEAT INTRINSICS
      SUBROUTINE SETIZT
C SETS UP ALL THE GRAPHICS VARIABLES FOR IZTLIB
C
      COMMON /PRINTS/ PRINT(10),IPRINT,MPRINT
      COMMON /TIMORG/ XORGT,YORGT,SIZEX,SIZEY,SCALEX,SCALEY,
     1XINC,YINC,XMAXT,YMAXT
      DATA TEN /10.0/
C
      IPRINT=0
      MPRINT=10
C
C SET GRAPH SIZE
      SIZEX=TEN
      SIZEY=TEN
C
C SET INITIAL ORIGIN
      XORGT=0.0
      YORGT=0.0
C
      RETURN
      END
```

```
      SUBROUTINE ZTRINV(NORNUM,ARRNUM,NORDEN,ARRDEN,
     1INVPTS,INVCNT)
C CALCULATES INVERSE TRANSFORM
C *
      DIMENSION ZLESS1(2),ZALONE(2)
      DIMENSION ARRAYN(60),ARRAYD(60)
      DIMENSION ARRNUM(1),ARRDEN(1)
      LOGICAL PERPH,NOTPER
      COMMON /PERPHS/ PERPH(2),NOTPER(2)
C
      DATA ZLESS1 /-1.0,1.0/
      DATA ZALONE /0.0,1.0/
C
C FIRST MAKE A COPY OF NUMERATOR POLYNOMIAL
      CALL ARRSAV(NORNUM,ARRNUM(1),NORDN,ARRAYN(1))
C
C NOW CALCULATE THE INVERSE TRANSFORM
      CALL TIMRES(NORDN,ARRAYN(1),NORDEN,ARRDEN(1),INVPTS,
     1INVCNT)
C
C NOW FIND INITIAL VALUE
      VINIT=VSTART(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1),INITAL)
C
20    IF(PERPH(1)) WRITE(4,110) INITAL,VINIT
110   FORMAT(' INITIAL ORDER =',I3,2X,'VALUE =',F10.4/)
      WRITE(3,110) INITAL,VINIT
C
C NOW FIND FINAL VALUE
30    CALL ARRSAV(NORNUM,ARRNUM(1),NORDN,ARRAYN(1))
      CALL ARRSAV(NORDEN,ARRDEN(1),NORDD,ARRAYD(1))
C
      CALL POLMLT(NORDN,ARRAYN(1),1,ZLESS1(1))
      CALL POLMLT(NORDD,ARRAYD(1),1,ZALONE(1))
C
      VEND=VFINAL(NORDN,ARRAYN(1),NORDD,ARRAYD(1),IFINAL)
      GO TO (50,40),IFINAL
40    IF(PERPH(1)) WRITE(4,120)
120   FORMAT(' INFINITE FINAL VALUE'/)
      WRITE(3,120)
      RETURN
50    IF(PERPH(1)) WRITE(4,130) VEND
130   FORMAT(' FINAL VALUE ='F10.4/)
      WRITE(3,130) VEND
C
      RETURN
      END
```

```
      SUBROUTINE ZTRIMP(NORNUM,ARRNUM,NORDEN,ARRDEN,
     1IMPPTS,IMPCNT)
C CALCULATES THE IMPULSE RESPONSE
C
      DIMENSION ZLESS1(2),ZALONE(2)
      DIMENSION ARRAYN(60),ARRAYD(60)
      DIMENSION ARRNUM(1),ARRDEN(1)
      LOGICAL PERPH,NOTPER
      COMMON /PERPHS/ PERPH(2),NOTPER(2)
C
      DATA ZLESS1 /-1.0,1.0/
      DATA ZALONE /0.0,1.0/
C
C NOW FIND THE IMPULSE RESPONSE
C FIRST MAKE A COPY OF THE NUMERATOR
      CALL ARRSAV(NORNUM,ARRNUM(1),NORDN,ARRAYN(1))
      CALL ARRSAV(NORDEN,ARRDEN(1),NORDD,ARRAYD(1))
C
C NOW MULTIPLY BY (Z/(Z-1))
      CALL POLMLT(NORDN,ARRAYN(1),1,ZALONE(1))
      CALL POLMLT(NORDD,ARRAYD(1),1,ZLESS1(1))
C
C NOW CALCULATE THE INVERSE TRANSFORM
      CALL TIMRES(NORDN,ARRAYN(1),NORDD,ARRAYD(1),IMPPTS,
     1IMPCNT)
C
C NOW FIND INITIAL IMPULSE RESPONSE VALUE
      CALL ARRSAV(NORNUM,ARRNUM(1),NORDN,ARRAYN(1))
      CALL ARRSAV(NORDEN,ARRDEN(1),NORDD,ARRAYD(1))
      CALL POLMLT(NORDN,ARRAYN(1),1,ZALONE(1))
      CALL POLMLT(NORDD,ARRAYD(1),1,ZLESS1(1))
      VINIT=VSTART(NORDN,ARRAYN(1),NORDD,ARRAYD(1),INITAL)
C
20    IF(PERPH(1)) WRITE(4,110) INITAL,VINIT
110   FORMAT(' INITIAL ORDER =',I3,2X,'VALUE =',F10.4/)
      WRITE(3,110) INITAL,VINIT
C
30    CALL POLMLT(NORDN,ARRAYN(1),1,ZLESS1(1))
      CALL POLMLT(NORDD,ARRAYD(1),1,ZALONE(1))
C
      VEND=VFINAL(NORDN,ARRAYN(1),NORDD,ARRAYD(1),IFINAL)
      GO TO (50,40),IFINAL
40    WRITE(4,120)
120   FORMAT(' INFINITE FINAL VALUE'/)
      WRITE(3,120)
      RETURN
50    WRITE(4,130) VEND
130   FORMAT(' FINAL VALUE =',F10.4/)
      WRITE(3,130) VEND
C
```

```
RETURN
END
```

```
      SUBROUTINE TIMRES(NORNUM,ARRNUM,NORDEN,ARRDEN,
     1NUMPTS,NUMITN)
C COMPUTES INVERSE Z-TRANSFORM & OUTPUTS ON L/P OR G/P.
C
      LOGICAL PERPH,NOTPER
      DIMENSION ARRNUM(1),ARRDEN(1)
C
      COMMON /CIL/ IPENX,IPENY,IPS,IPC,IPCN,FACR,
     1XM,SIZES,SIZEN,SIZEL,TICK,STEP,XSPAC,
     2IPNAB,ITAPE,IB,IBC,IBYTE,IBASE,IMMET
      COMMON /PERPHS/ PERPH(2),NOTPER(2)
      DATA ZERO /0.0/
C
C OUTPUT L/P HEADINGS
      IF(PERPH(1)) CALL INVPRT(ZERO,ZERO,0)
C
C SET PLOT MODE TO INVISIBLE
      ISTATE=3
C
C NOW CALCULATE THE INVERSE TRANSFORM
      NUMSTP=NUMITN/NUMPTS
      IF(NUMSTP.LE.0) NUMSTP=1
      DO 10 I=0,NUMITN
      TIME=FLOAT(I)
      AMP=REPDIV(NORNUM,ARRNUM(1),NORDEN,ARRDEN(1))
      IF(PERPH(1)) CALL INVPRT(TIME,AMP,1)
      IF(NOTPER(2)) GO TO 10
      IF(MOD(I,NUMSTP).EQ.0) CALL INVPLT(TIME,AMP,ISTATE)
C SET PLOT MODE TO VISIBLE
      ISTATE=2
10    CONTINUE
C
C NOW FORCE PLOTTING OF THE LAST RESULT
      IF(PERPH(2)) CALL INVPLT(TIME,AMP,ISTATE)
C
C NOW RESET PLOT MODE TO INVISIBLE AGAIN
      ISTATE=3
C
C NOW CLOSE L/P OUTPUT
      IF(PERPH(1)) CALL INVPRT(ZERO,ZERO,-1)
C
      RETURN
      END
```

```
      SUBROUTINE INVPRT(TIME,AMP,ICNTRL)
C PRINTS TIME & AMPLITUDE ON L/P.
C HOWEVER, THIS ROUTINE IS INTENDED TO SAVE PAPER
C BY SAVING UP THE OUTPUT UNTIL A FULL LINE'S WORTH
C IS READY OR THE BUFFER IS TO BE FLUSHED.
C
C ICNTRL < 0 FLUSH BUFFER
C ICNTRL = 0 OUTPUT HEADINGS
C ICNTRL > 0 STORE DATA
C
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      COMMON /MATELM/ NUMRES,MATRIX,NELEMS
      COMMON /PRINTS/ PRINT(10),IPRINT,MPRINT
      COMMON /PARAMS/ PAROLD(6),PARNEW(6)
C
C CHECK IF ICNTRL <0
      IF(ICNTRL) 20,40,10
C
10    IPRINT=IPRINT+2
      PRINT(IPRINT-1)=TIME
      PRINT(IPRINT)=AMP
C
C NOW TEST IPRINT
      IF(IPRINT.LT.MPRINT) RETURN
C
C NOW OUTPUT PRINT
20    WRITE(4,100) (PRINT(I),I=1,IPRINT)
100   FORMAT(1H ,5(F8.0,2X,F10.4))
      IPRINT=0
C
30    RETURN
C
C OUTPUT HEADINGS
40    WRITE(4,110) (ITITD(I),I=1,NTITD)
110   FORMAT(1H1,72A1)
      WRITE(4,120) NUMRES,MATRIX,NELEMS
120   FORMAT(' NUMBER OF RESULTS =',I6/
     1' MATRIX ELEMENT ',A1,I2/)
      WRITE(4,130) PARNEW
130   FORMAT(' SAMPLE PERIOD  ',E13.6/' C1',13X,E13.6/
     1' C2',13X,E13.6/' G1',13X,E13.6/' G2',13X,E13.6/
     2' SHUNT RESISTOR ',E13.6/)
      WRITE(4,140)
140   FORMAT(1H ,5(4X,'TIME',3X,'AMPLITUDE',3X))
      RETURN
      END
```

```
      SUBROUTINE TIMPLT(XORG,YORG,XMAX,YMAX,ITITLE,NTITLE)
C DRAWS AXES FOR INVERSE Z-TRANSFORM PLOTS
C
      REAL NINETY
      DIMENSION INVTIM(4),INVAMP(5),ITITLE(1)
      COMMON /TIMORG/ XORGT,YORGT,SIZEX,SIZEY,SCALEX,SCALEY,
     1XINC,YINC,XMAXT,YMAXT
      DATA ZERO,NINETY,CHSIZE /0.0,90.0,0.4/
      DATA INVTIM /2HSA,2HMP,2HLE,2HS /
      DATA INVAMP /2HAM,2HPL,2HIT,2HUD,2HE /
C
C INITIALISE
      CALL SETPLT(XORG,YORG,XMAX,YMAX)
C
C DRAW TIME AXIS
      XSIZE=AINT(SIZEX*SCALEX)
      CALL AXIS(ZERO,ZERO,XSIZE,ZERO,ZERO,
     1XINC,INVTIM(1),7)
C
C DRAW AMPLITUDE AXIS
      YSIZE=AINT(SIZEY*SCALEY)
      HALFY=YSIZE/2.0
      CALL AXIS(ZERO,-HALFY,YSIZE,NINETY,-YMAXT,
     1YINC,INVAMP(1),-9)
C
C NOW OUTPUT THE GRAPH TITLE
      CALL SYMBOL(ZERO,-(HALFY+1.0),CHSIZE,ITITLE(1),
     1ZERO,2*NTITLE)
C
      RETURN
      END
```

```
      SUBROUTINE SETPLT(XORG,YORG,XMAX,YMAX)
C SETS NEW ORIGINS
C
      COMMON /TIMORG/ XORGT,YORGT,SIZEX,SIZEY,SCALEX,SCALEY,
     1XINC,YINC,XMAXT,YMAXT
C
C FIRST RESET ORIGIN
      CALL PLOT(-XORGT,-YORGT,-3)
C
C NOW STORE NEW VALUES
      XORGT=XORG
      YORGT=YORG
      XMAXT=XMAX
      YMAXT=YMAX
      XINC=XMAXT/AINT(SIZEX*SCALEX)
      YINC=2.0*YMAXT/AINT(SIZEY*SCALEY)
C
C NOW SET NEW ORIGIN
      CALL PLOT(XORGT,YORGT,-3)
C
      RETURN
      END
```

```
      SUBROUTINE ANNOTE
C ANNOTATES GRAPH BY WRITING A COMMENT &
C THE WORD 'GRAPH'.
C
      DIMENSION IGRAPH(3)
      COMMON /TITLES/ ITITD(72),NTITD,ITITG(72),NTITG,MTITLE
      COMMON /TIMORG/ XORGT,YORGT,SIZEX,SIZEY,SCALEX,SCALEY,
     1XINC,YINC,XMAXT,YMAXT
      DATA ZERO,CHSIZE /0.0,0.4/
      DATA IGRAPH /2HGR,2HAP,2HH /
C
C WRITE THE GRAPH COMMENT
      HALFY=0.5*AINT(SIZEY*SCALEY)
      CALL SYMBOL(ZERO,-(HALFY+2.0),CHSIZE,
     1ITITG(1),ZERO,2*NTITG)
C
C NOW WRITE 'GRAPH'
      CALL SYMBOL(10.0,-HALFY-3.0,CHSIZE,IGRAPH(1),ZERO,6)
C
      RETURN
      END
```

```
      SUBROUTINE INVPLT(XPOINT,YPOINT,ISTATE)
C PLOTS POINTS ON Z-TRANSFORM PLOT
C
      COMMON /TIMORG/ XORGT,YORGT,SIZEX,SIZEY,SCALEX,SCALEY,
     1XINC,YINC,XMAXT,YMAXT
C
      X=AMIN1(XPOINT,XMAXT)
      Y=SIGN(AMIN1(ABS(YPOINT),YMAXT),YPOINT)
C
      CALL PLOT(X/XINC,Y/YINC,ISTATE)
C
C
      RETURN
      END
```

```
      SUBROUTINE ARRSAV(NORD1,ARRAY1,NORD2,ARRAY2)
C SAVES ARRAY1 INTO ARRAY2.
C
      DIMENSION ARRAY1(1),ARRAY2(1)
C
      NORD2=NORD1
      DO 10 I=0,NORD1
      ARRAY2(I+1)=ARRAY1(I+1)
10    CONTINUE
      RETURN
      END
```

```
      REAL FUNCTION VSTART(NORNUM,ARRNUM,NORDEN,ARRDEN,
     1ISTART)
C FINDS INITIAL VALUE OF RATIONAL POLYNOMIAL IN Z DOMAIN
C BY TAKING LIMIT AS Z TENDS TO INFINITY.
C
C
      DIMENSION ARRNUM(1),ARRDEN(1)
C
      NCFNUM=NORNUM+1
      NCFDEN=NORDEN+1
C
C GET FIRST NON-ZERO NUMERATOR COEFFICIENT
      DO 10 I=1,NCFNUM
      INUM=NCFNUM+1-I
      VALNUM=ARRNUM(INUM)
      IF(VALNUM) 20,10,20
10    CONTINUE
C
C NOW GET FIRST NON-ZERO DENOMINATOR COEFFICIENT
20    DO 30 I=1,NCFDEN
      IDEN=NCFDEN+1-I
      VALDEN=ARRDEN(IDEN)
      IF(VALDEN) 40,30,40
30    CONTINUE
C
C NOW CALCULATE THE ORDER OF THE INITIAL VALUE
40    ISTART=INUM-IDEN
C
C NOW CALCULATE THE INITIAL VALUE ITSELF
      VSTART=VALNUM/VALDEN
C
      RETURN
      END
```

```
      REAL FUNCTION ARRSUM(NORDER,ARRAY)
C SUMS COEFFICIENTS OF ARRAY
C
      DIMENSION ARRAY(1)
C
      SUM=0.0
      DO 10 I=0,NORDER
      SUM=SUM+ARRAY(I+1)
10    CONTINUE
C
      ARRSUM=SUM
      RETURN
      END
```

```
      REAL FUNCTION VFINAL(NORNUM,ARRNUM,NORDEN,ARRDEN,
     1IFINAL)
C FINDS FINAL VALUE OF POLYNOMIAL IN Z DOMAIN.
C IFINAL = 1 OK
C IFINAL = 2 INFINITE FINAL VALUE
C
      DIMENSION ARRNUM(1),ARRDEN(1)
C
C START BY EVALUATING NUMERATOR & DENOMINATOR SEPARATELY :
10    VALUEN=ARRSUM(NORNUM,ARRNUM(1))
      VALUED=ARRSUM(NORDEN,ARRDEN(1))
C
C NOW TEST RESULTS
      IF(VALUED) 20,30,20
20    VFINAL=VALUEN/VALUED
      IFINAL=1
      RETURN
C
C DENOMINATOR =0. INFINITE LIMIT IF NUMERATOR<>0
30    IF(VALUEN) 40,50,40
40    VFINAL=0.0
      IFINAL=2
      RETURN
C
C NUMERATOR & DENOMINATOR =0.
C DIFFERENTIATE NUMERATOR & DENOMINATOR :
50    CALL POLDIF(NORNUM,ARRNUM(1))
      CALL POLDIF(NORDEN,ARRDEN(1))
C
C NOW TRY AGAIN
      GO TO 10
      END
```

```
C FRPLIB
      SEGMENT 19
      REPEAT INTRINSICS
C
      SUBROUTINE FQPLOT(NUMPTS)
C COLLECTS EXPERIMENTAL DATA & STORES IT ON
C DISK FILE ON UNIT 6
C
      INTEGER YES
      LOGICAL ANGDEG,AMPDBS
      COMPLEX VALUE
      DATA IABSOL,IDBS,IDEG,IRAD /1HA,1HD,1HD,1HR/
      DATA NEND,IZERO /9,0/
      DATA YES,NO /1HY,1HN/
      DATA PI /3.1415926/
C
      FACTOR=ALOG(10.0)/20.0
C
C GET DATA TYPES
      AMPDBS=.FALSE.
100   WRITE(3,1000)
1000  FORMAT(' AMPLITUDE AS ABSOLUTE OR DECIBELS (A/D):'/)
      READ(2,2000) ITYPE
2000  FORMAT(A1)
      IF(ITYPE.NE.IABSOL.AND.ITYPE.NE.IDBS) GO TO 100
      IF(ITYPE.EQ.IDBS) AMPDBS=.TRUE.
C
110   ANGDEG=.FALSE.
      WRITE(3,1010)
1010  FORMAT(' PHASE IN DEGREES OR RADIANS (D/R) :'/)
      READ(2,2000) IANG
      IF(IANG.NE.IDEG.AND.IANG.NE.IRAD) GO TO 110
      IF(IANG.EQ.IDEG) ANGDEG=.TRUE.
C
C NOW GET DATA
      WRITE(3,1040)
1040  FORMAT(' GIVE FREQUENCY, AMPLITUDE & PHASE'/)
C
C GET DATA
      DO 10 I=1,NUMPTS
20    WRITE(3,1100) I
1100  FORMAT(1H ,I6/)
      READ(2,2100) TESTFQ,AMP,PHASE
2100  FORMAT()
C
C NOW CHECK IF THESE VALUES ARE REALLY REQUIRED
30    WRITE(3,1200) TESTFQ,AMP,PHASE
1200  FORMAT(' ARE THESE VALUES CORRECT? :'/
     13(F12.4,2X)/)
      READ(2,2200) IANS
```

```
2200  FORMAT(A1)
      IF(IANS.EQ.NO) GO TO 20
      IF(IANS.NE.YES) GO TO 30
C
C CONVERT PHASE TO RADIANS
      IF(ANGDEG) PHASE=PHASE*PI/180.0
C
C CONVERT AMPLITUDE TO ABSOLUTE
      IF(AMPDBS) AMP=EXP(AMP*FACTOR)
C
C CALCULATE COMPLEX VALUE
      VALUE=CMPLX(AMP*COS(PHASE),AMP*SIN(PHASE))
C
C NOW WRITE TO DISK
C FIRST STATUS
      WRITE(6) IZERO
C
C NOW WRITE DATA
      WRITE(6) VALUE,TESTFQ
C
10    CONTINUE
C
C NOW WRITE END STATUS
      WRITE(6) NEND
      RETURN
      END
```

APPENDIX E

FREQUENCY RESPONSE ALGORITHM

E.1 FREQUENCY RESPONSE ALGORITHM

The Z- transform variable z is defined as:

$$z = e^{sT} \tag{E.1}$$

where T_s is the uniform sampling period and s is the Laplace variable:

$$s = \sigma + j\,w \tag{E.2}$$

To study the frequency response of a digital active network, a steady sinusoid must be applied, thus making $\sigma = 0$ and:

$$z = e^{j\,wTs}$$

$$= \text{Cos } wT_s + j \text{ Sin } wT_s \tag{E.3}$$

and further by De Moivre's theorem :

$$z^r = \text{Cos } r\,wt_s + j \text{ Sin } r\,wT_s \tag{E.4}$$

Now consider a rational polynomial in z, A(z):

$$A(z) = \frac{\sum_{i=1}^{n} a_i z^i}{\sum_{i=1}^{m} b_i z^i} \tag{E.5}$$

Thus A(jw) becomes:

$$A(jw) = \frac{\sum_{i=1}^{n} a_i \left[\text{Cos } i\,wT_s + j \text{ Sin } i\,wT_s\right]}{\sum_{i=1}^{m} b_i \left[\text{Cos } i\,wT_s + j \text{ Sin } i\,wT_s\right]}$$

$$= \frac{\sum_{i=1}^{n} a_i \quad \text{Cos } i\, wT_s + j \sum_{i=1}^{n} a_i \text{ Sin } i\, wT_s}{\sum_{i=1}^{m} b_i \quad \text{Cos } i\, wT_s + j \sum_{i=1}^{m} b_i \text{ Sin } i\, wT_s}$$

$$= \frac{B_{11} + j\, B_{12}}{B_{w1} + j\, B_{22}} \qquad \text{(E.6)}$$

where:

$$B_{11} = \sum_{i=1}^{n} a_i \text{ Cos } iwT_s$$

$$B_{12} = \sum_{i=1}^{n} a_i \text{ Sin } iw\, T_s$$

$$B_{21} = \sum_{i=1}^{m} b_i \text{ Cos } iw\, T_s$$

$$B_{22} = \sum_{i=1}^{m} b_i \text{ Sin } iw\, T_s$$

The modulus of A(jw) is thus:

$$\left| A(jw) \right| = \frac{\left| B_{11} + j\, B_{12} \right|}{\left| B_{21} + j\, B_{22} \right|} \qquad \text{(E.7)}$$

and the arg. of A(jw) is thus:

$$A(jw) = \text{Tan}^{-1} (B_{12}/B_{11}) - \text{Tan}^{-1} (B_{22}/B_{21}) \qquad \text{(E.8)}$$

hence in order to compute $\left| A(jw) \right|$ and $\angle A(jw)$, B_{11}, B_{12}, B_{21}

and B_{22} must be found for each value of w. Further the complex components of A(jw) may now be calculated:

$$\mathrm{Re}\left[A(j\,w)\right] = \left|A\,(j\;\;w)\right| \quad \mathrm{Cos}\,(\,\angle\, A(j\,w)\,) \qquad \text{(E.9)}$$

$$\mathrm{Im}\left[A(j\,w)\right] = \left|A\,(j\;\;w)\right| \quad \mathrm{Sin}\,(\,\angle\, A(j\,w)\,) \qquad \text{(E.10)}$$

This algorithm has been implemented as subroutine ZPOLY.

APPENDIX F

ANALOGUE TRANSCONDUCTANCE AMPLIFIER

F.1 INTRODUCTION

The analogue transconductance amplifier shown in Fig. F.1 is analysed here because of the extensive use made of this circuit in the experimental machine.

F.2 ANALYSIS

Consider the circuit in Fig. F.1. The operational amplifier is assumed to be ideal, namely the input impedance tends to infinity and the output impedance to zero. However, the differential voltage gain (m) is assumed to be finite initially.

Now, by definition:

$$V_o = m\,(V_1 - V_g) \tag{F.1}$$

$$I_2 = h_{fe}\, i_b \tag{F.2}$$

where h_{fe} is the transistor current gain, and by observation:

$$V_o = V_{be} + V_g \tag{F.3}$$

$$V_g = \frac{1}{g}\,(i_b + I_2) \tag{F.4}$$

where V_{be} is the base-emitter forward bias voltage drop.

Equations (F.1) to (F.4) may be rearranged to give:

$$I_2 = g\,(mV_1 - V_{be}) \left(\frac{1}{1+m}\right) \left(\frac{1}{1+1/h_{fe}}\right)$$

If the voltage gained $m \gg 1$ and the transistor small signal current gain $h_{fe} \gg 1$ then the following limit may be taken :

$$\underset{h_{fe},\, m \Rightarrow \infty}{Lt} \left\{ I_2 \right\} = g\, V_1 \tag{F.5}$$

The two conditions can be easily met by choosing an operational amplifier with a large open-loop gain (m) and employing a darlington transistor pair in place of the single transistor shown in Fig. F.1.

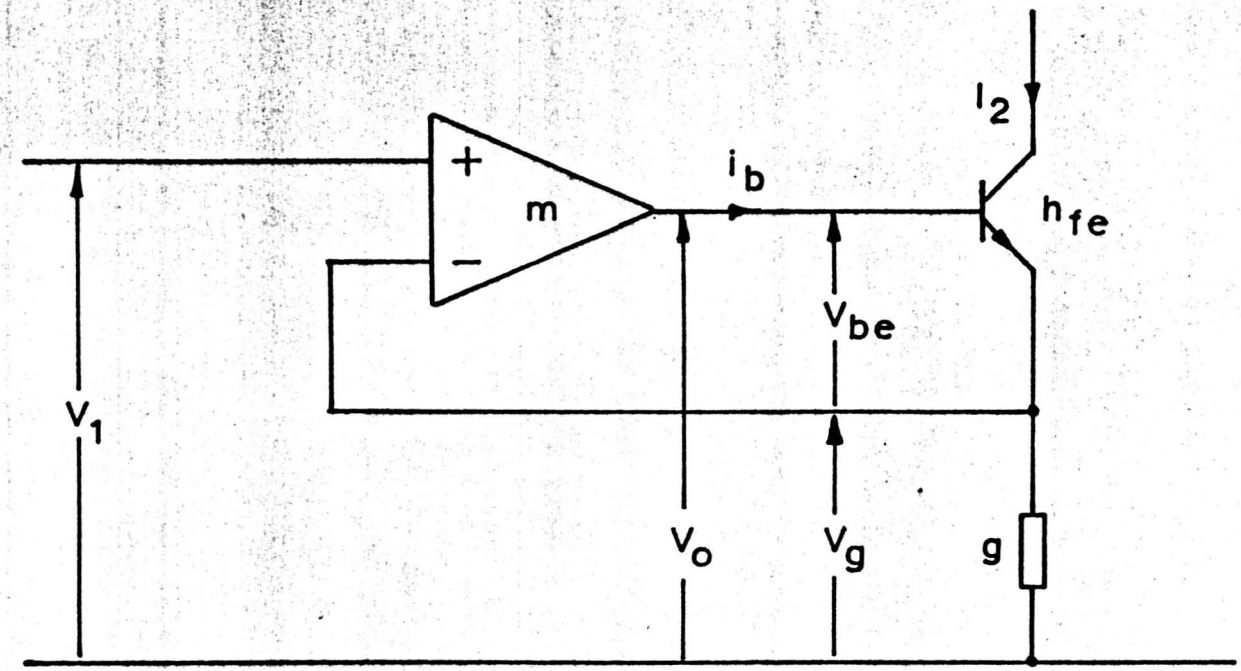


FIG. F.1 Analogue Transconductance Amplifier

APPENDIX G

MEASUREMENT OF GAIN AND PHASE

G.1 INTRODUCTION

The practical measurement of voltage gain and phase of an active or passive, digital or analogue network is given here. The method was used in Chapter 7 to obtain the practical results.

G.2 VOLTAGE GAIN

The arrangement of equipment to measure the voltage gain is shown in Fig. G.1 .

From Fig. G.1 the magnitude of V_N and V_M may be written thus:

$$\left| V_N \right| = N \left| h(s) \right| \left| V_s \right| \tag{G.1}$$

$$\left| V_M \right| = M \left| V_s \right| \tag{G.2}$$

If the two attenuators are adjusted so that the voltmeter reads the same value independent of switch setting then:

$$\left| V_N \right| = \left| V_M \right| \tag{G.3}$$

and hence:

$$\left| h(s) \right| = M/N \tag{G.4}$$

If the range of values which $\left| h(s) \right|$ may take is limited then the test arrangement may be simplified. For instance let:

$$\left| h(s) \right| \leqslant 1 \tag{G.5}$$

which will be so in a passive R-C network then the attenuator M may be set to 1, or removed entirely. If $\left| h(s) \right| \geqslant 1$ then the converse will apply.

G.2.1 Logarithmic Attenuators

In most practical situations the two attenuators will be

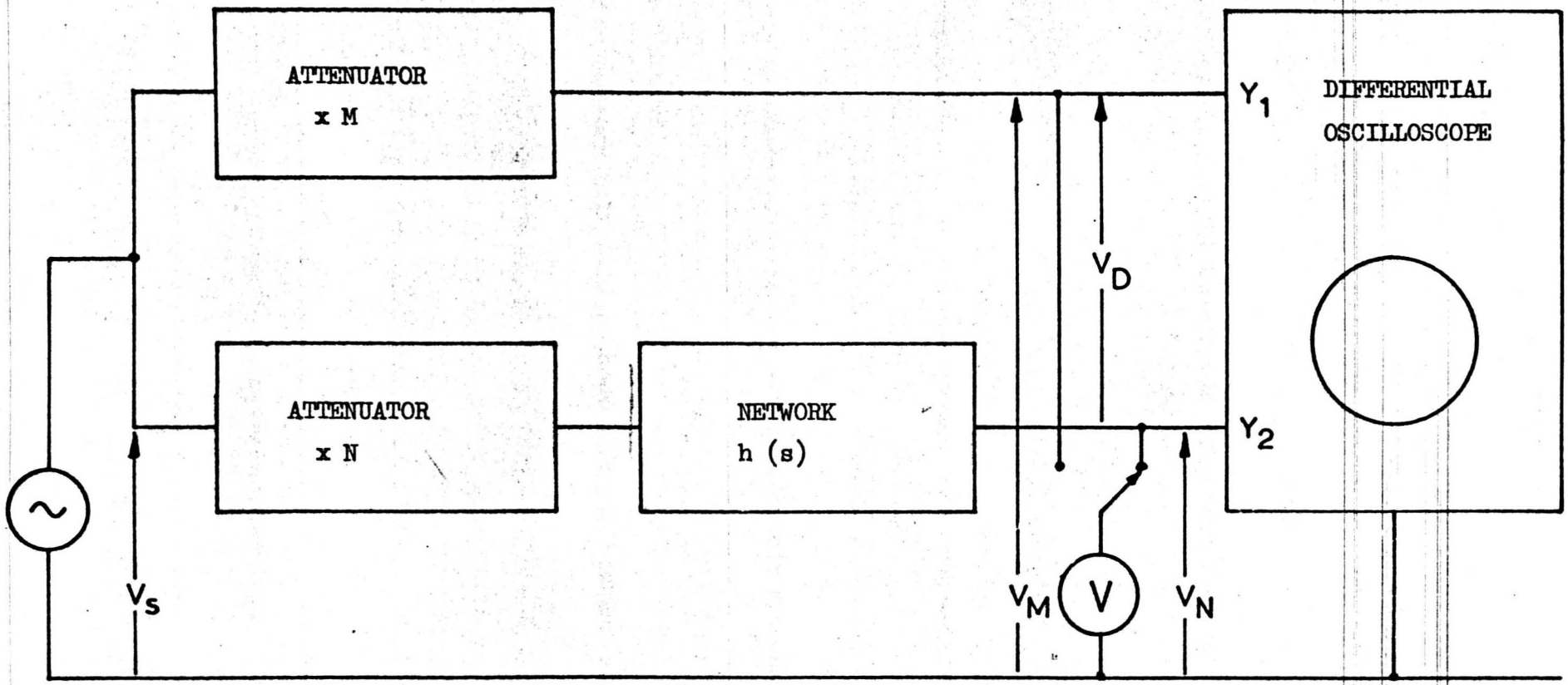


FIG. G.1 Voltage Gain Measurement

calibrated in decibels. Hence equation (G.4) may be rewritten:

$$|h(s)| = 10^{(M_{dB} - N_{dB}) / 20} \qquad (G.6)$$

G.2.2 Voltage Gain Accuracy

The maximum fractional error in $|h(s)|$ may be derived from equation (G.4)

$$\left| \frac{|\triangle h(s)|}{|h(s)|} \right|_{max} = \frac{\triangle M}{M} + \frac{\triangle N}{N} \qquad (G.7)$$

The expression for the fractional error may also be derived from equation (G.6):

$$\left| \frac{|\triangle h(s)|}{|h(s)|} \right|_{max} = 10^{(\triangle M_{dB} - \triangle N_{dB}) / 20} - 1 \qquad (G.8)$$

G.3 PHASE RESPONSE

The difference voltage V_D from Fig. G.1 is:

$$V_D = V_N - V_M$$

The signal source must be sinusoidal for this method to work. Hence:

$$V_S = V \text{ Cos } wt$$

Let the network introduce a phase shift ϕ. Thus:

$$V_D = N \left| h(s) \right| V \text{ Cos } (wt + \phi) - M V \cos w t$$

However from equation (G.4) and after manipulation:

$$\left| V_D \right| = 2 \left| V_M \right| \text{ Sin } (\phi/2)$$

and therefore:

$$\phi = -2 \text{ Tan}^{-1} \left[\frac{\left| V_D \right|}{(4 \left| V_M \right|^2 - \left| V_D \right|^2)^{\frac{1}{2}}} \right] \qquad (G.9)$$

Note that $\left| V_D \right|$ is often measured as peak volts, but $\left| V_N \right|$ and V_M as R.M.S. volts.

Thus equation (G.9) can be rewritten thus:

$$\phi = -2\mathrm{Tan}^{-1}\left[\frac{|V_D|_{PP}}{(32|V_M|^2_{RMS} - |V_D|^2_{PP})^{\frac{1}{2}}}\right] \qquad \text{(G.10)}$$

The derivation of accuracy of ϕ from equations (G.9) and (G.10) cannot be simply found.

G.4 CONCLUSIONS

It has been shown that the gain and phase of a network may be measured simply. Equations (G.7) and (G.10) are used in Chapter 7.