IDENTIFICATION OF CRYSTAL GROWTH DYNAMICS IN CZOCHRALSKI SYSTEMS

MICHAEL JAMES HILL

Submitted For The Degree of Doctor of Philosophy
to
The University of Aston in Birmingham

October 1977

Summary

An experimental method is established to deduce the transfer function relating crystal radius to power changes in a Czochralski crystal growth system. The experimental method employs pseudo-random signal injection into the power controller of a gallium phosphide crystal puller and uses crystal radius measurements made on the cold crystal after growth is complete. This is the first known application of pseudo-random signal testing to crystal growth. Novel data processing techniques are developed to convert the crystal radius measurements to samples at equal time intervals. Fast Fourier transform processing is used to compute frequency responses for the process. An algorithm is developed to fit transfer functions to modulus frequency response data. A theoretical derivation of the transfer function is performed. This is based on the heat balance equation at the solid-liquid interface. Due to lack of data on gallium phosphide, the error bounds of the prediction are wide but it is shown that they may be reduced by using the measured transfer function. The work has application to the design of control systems for crystal growth and to a more fundamental understanding of the crystal growth process. This is the first known description of a theoretical analysis of Czochralski crystal growth dynamics.

CONTENTS

APPENDICES

ACKNOWLEDGMENTS

I would like to thank my Supervisor, Dr G K Steel, for many useful discussions and his encouragement.

I thank Metals Research Ltd, Melbourne, Herts for the experimental facilities and in particular Mr D Wingrove for his assistance with the crystal growing.

LIST OF DIAGRAMS

DIAGRAMS (Cont)

List of Principle Symbols

B_e	Estimation bandwidth
f	Frequency (Hz)
f_s	Sample frequency
G_i	Temperature gradient in melt
G_o	Temperature gradient in crystal
G_{xx}	Power spectrum
G_{xy}	Cross power spectrum
h_o	Steady state meniscus height
h_c	Melt depth in crucible
H	Capilliary constant
$H(s)$	Transfer function
J	Latent heat of solid
k_m	Thermal conductivity of melt
k_s	Thermal conductivity of solid
K	Meniscus curvature constant
K_o	Overall gain constant
P_f	Power due to fusion
P_i	Power input from melt
P_o	Power input to crystal
P_r	Power radiated from interface
r	Instantaneous crystal radius
r_o	Nominal crystal radius
R	Crucible radius
R_m	Meniscus radius of curvature
s	Laplace transform variable
t	Time
V	Growth rate
y	Instantaneous crystal length
z	Unit advance operator

α	Gradient variation factor
γ	Angle of contact
γ^2	Coherence function
ζ	Damping factor
η	Pull bar position
θ	Crystal surface angle
θ_a	Growth chamber ambient temperature
θ_i	Interface temperature
θ_m	Melt temperature
θ_s	Susceptor temperature
μ	Differential temperature ratio
ρ_s	Density of solid
ρ_m	Density of melt
$\varnothing$	Phase angle
Φ	Locally defined parameter
ω	Frequency (rad/sec)
Ω	Natural frequency

1. INTRODUCTION

1.1 GENERAL CZOCHRALSKI GROWTH

Crystal growing has been described as a craft, (Brice 1973)[1] and inasmuch as the process cannot be entirely predicted from theoretical or empirical considerations, this is an accurate description. One of the more important crystal growth methods is that due to Czochralski[2] in which the crystal is pulled from the melt at constant temperature. The normal crystal growing procedure with this method is to dip a seed into the melt and wait for thermal equilibrium. Then, while slowly withdrawing the seed from the melt, the melt temperature is increased, causing the growing crystal to decrease in diameter to as small a value as will be able to support the fully grown crystal. The purpose of this "neck in" portion of crystal growth is to eliminate crystalline imperfections that were developed at the initial solid-liquid interface when the seed was dipped into the melt. From this point, the melt temperature is slowly reduced while the crystal continues to be withdrawn from the melt. The reduction in melt temperature causes the diameter of the growing crystal to increase. This is continued until the required crystal diameter is reached, at which time the melt temperature is maintained relatively constant. The radial temperature gradient along the surface of the melt must be maintained such that the usual random temperature fluctuations due to imperfect temperature control will not cause the outer edge of the melt to solidify while the centre of the melt is too hot for proper crystal growth.

There are two possible rate-limiting steps in crystal growth. First is the time required for atoms within the liquid to migrate to a lattice site. This time is very short and although it affects the impurity distribution, it does not affect the crystal growing rate. Secondly, and most important, is the requirement for the removal of the latent heat of fusion. This heat is removed in three ways: by conduction through the crystal pull mechanism, by convection and

radiation from the body of the growing crystal itself and by conduction through the melt and the crucible support mechanism.

Crystal growing apparatus normally has three means of control: melt temperature control, seed and/or crucible rotation control, and crystal pull speed control. Figure 1.1 shows a cross section through a typical, pressurised Czochralski crystal puller and Figure 1.2 shows an enlarged section of the crucible with a growing crystal. During crystal growth a number of process variables must be monitored and controlled:

1. melt temperature must be precisely maintained to obtain the desired rate of solidification for the growing crystal. This is accomplished mainly by varying the power input to the heater.
2. crystal pull speed must be controlled because of its obvious effect on the rate of crystal growth.
3. crystal rotation speed has a shaping effect on the growing crystal and also has a great influence on stirring of the melt.
4. crucible lift speed: this is optional, to maintain the melt level at the same position relative to the heater as the crystal grows.
5. crucible rotation speed has a significant effect on the melt stirring which influences the temperature distribution in the melt.
6. cooling water flow rate affects the rate of heat removal from the process.
7. inert gas atmosphere, particularly at high pressures, is a significant convection medium to remove heat from the growing crystal.

1.1.1 Application and Objectives of Czochralski Growth

The Czochralski method has applicability to a wide range of materials. For instance Ga, with a melting point of 30C has been grown by Zimmerman[3] and La AlO_3 with a melting point of 2080C has been grown by Fay & Brandle[4]. An important extension to the method was made by Metz et al[5] when they introduced liquid encapsulated Czochralski (LEC) growth. This enabled compounds to be grown where one or more of the constituents was volatile. In particular this enabled the III-V

compounds to be grown. One of the more difficult of these inter-metallic materials, GaP, requires an inert gas pressure above the encapsulant of greater than 35 atmospheres (Bass & Oliver)[6]. Therefore it will be evident that the growth conditions in Czochralski growth systems are almost as diverse as the range of materials grown in them. In growing a Czochralski crystal there are usually several objectives:

a. to produce a single, strain free crystal
b. to produce a uniform diameter crystal
c. to produce uniform impurity distribution
d. to produce a low dislocation density

In practice, if the first two objectives are met, the last two will be satisfied. The most common method of Czochralski growth is to withdraw the crystal at constant speed and regulate the power input to the heater to produce an approximately uniform diameter crystal.

1.1.2 Automatic Growth

There have been numerous methods employed in attempts to transplant crystal growth from its status of that of an art to that of a science by using closed loop control techniques. The demand for single crystal silicon increased enormously in the late 1950s and much effort was spent on devising automatic production facilities. Levinson[7] described a control system for Czochralski growth which used a weight signal from either the growing crystal or the crucible to effect diameter control. Other weighing methods have been published by Reinert and Yatsko[8], Zinnes, Nevis and Brandle[9] and Bardsley et al[10,11] have derived the relationship between the apparent crystal weight and crystal diameter (albeit a non-unique relationship). Other control systems have exploited the fact that silicon has a high heat of fusion which results in a bright ring at the solid-liquid interface which is detectible by infra red sensors. This method has been published by Jen, Slocum and Valentino[12], Corburn, Seksinsky and Tucker[13], Patzner, Dessauer and Poponiak[14], and Digges, Hopkins and Seidensticker[15]. An optical method using a laser beam to reflect off the meniscus curve at the interface

has been published by Gross and Kersten[16]. O'Kane et al[17] describe an infra red TV system with electronic processing of the TV signal to give a direct measure of the crystal diameter. Similar systems, but using visible TV methods are described by Bachmann et al[18] and Gartner et al[19]. Possibly the most direct method of diameter control is to use an X-ray imaging technique. This was the subject of a patent by Phillips[20] and descriptions by Van Dijk et al[21] and Pruett and Lien[22]. In a rather different category, since it is only a means of varying the diameter and not of measuring it as well, is a description by Vojdani et al[23], of a method of extracting heat from the interface region by Peltier cooling. It will be evident that of the above methods, only the weighing and the X-ray systems can have any application to encapsulated crystal growth since all the others rely on a line of sight to the solid-liquid interface. The weighing method is restricted in that for certain materials, notably GaP, Ge, InP which expand on freezing it is not possible to obtain a unique value of diameter from the weight signal except at very low growth rates[10,24]. The X-ray techniques all need very high energy beams in view of the necessary pressure chamber wall thickness and thus are not viable production methods.

1.2 CRYSTAL GROWTH DYNAMICS

In none of the literature referenced above has there been any mention of the Czochralski crystal growth system dynamics in either an empirical or a theoretical sense. This is a serious gap and one that this thesis sets out to bridge. The difficulty of making on line diameter measurements for encapsulated growth would be less serious if a system model could be set up hence enabling a model-reference control system to be implemented. A further advantage of having a knowledge of the dynamics of Czochralski crystal growth systems in general would be that crystal diameter could be controlled by a strategy that would be least likely to cause grown-in stress concentrations or cause twinning.

1.2.1 Test Signal Injection

The system dynamics, as developed in Appendix 1, are shown in Figure 1.3. It will be evident that the only externally accessible parts are the power input and the diameter output. There is a noise input, due to changing convection patterns around the crystal, changing emissivity of the melt surface due to impurities and changes in coolant flow rates. This input point is however only conceptual and there would be little point in actually identifying the dynamics between it and the diameter output. There is therefore a requirement to inject some form of test signal into the power controller in order to be able to identify the complete system. The choice of this signal is discussed in Chapter 3.

1.2.2 Special Identification Problems

The nature of the Czochralski process introduces some special problems into the system identification procedure:

1. As it is a batch process, only a limited amount of data will be available for each crystal.

2. Because the thermal conditions change from the beginning to the end of a run the dynamics are likely to vary over the run, ie they will be non-stationary.

3. The process is non-linear, but it is intended to make a linear estimate. Test signals will thus need to be restricted in amplitude with attendant problems of having a poor signal to noise ratio at the output.

4. The crystal diameter will have to be measured from the cold crystal afterwards. In order to process the input/output records it will be necessary to convert the diameter samples (measured at equal distances along the crystal) into samples at equal time intervals and then to synchronise the two sequences.

This thesis is concerned with the system identification of a GaP LEC crystal growth system, kindly made available by Metals Research Ltd.[/]

/ Metals Research Ltd, Melbourn, Royston, Herts, England.

FIGURE 1.1

Cross Section Through a Typical Pressurised Czochralski Crystal Puller

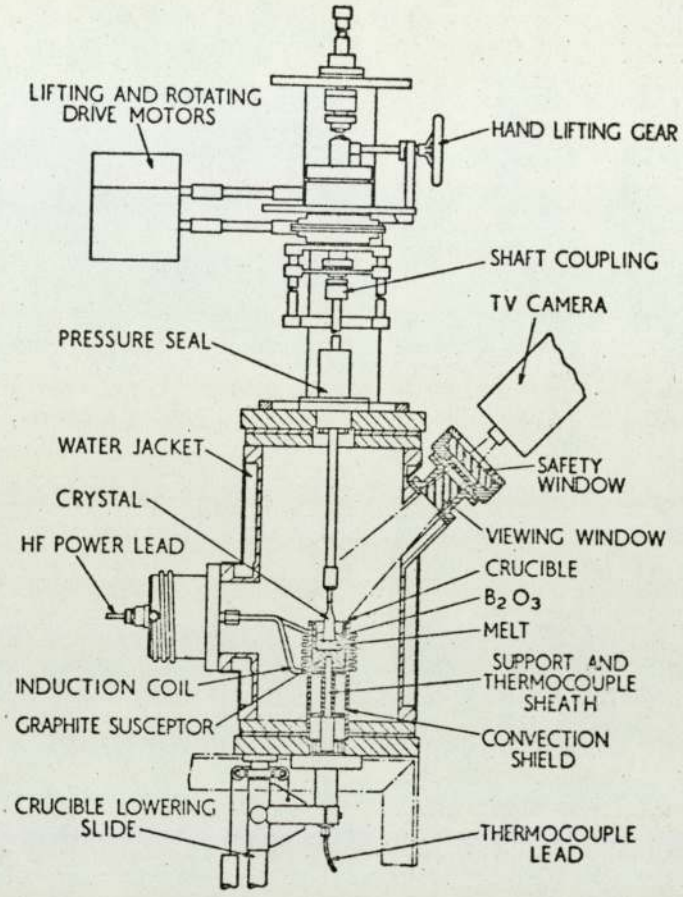


FIGURE 1.2

Cross Section Through Crucible With Growing Crystal

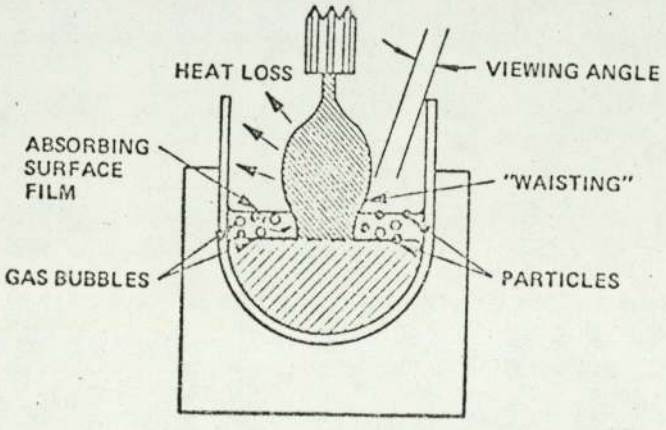
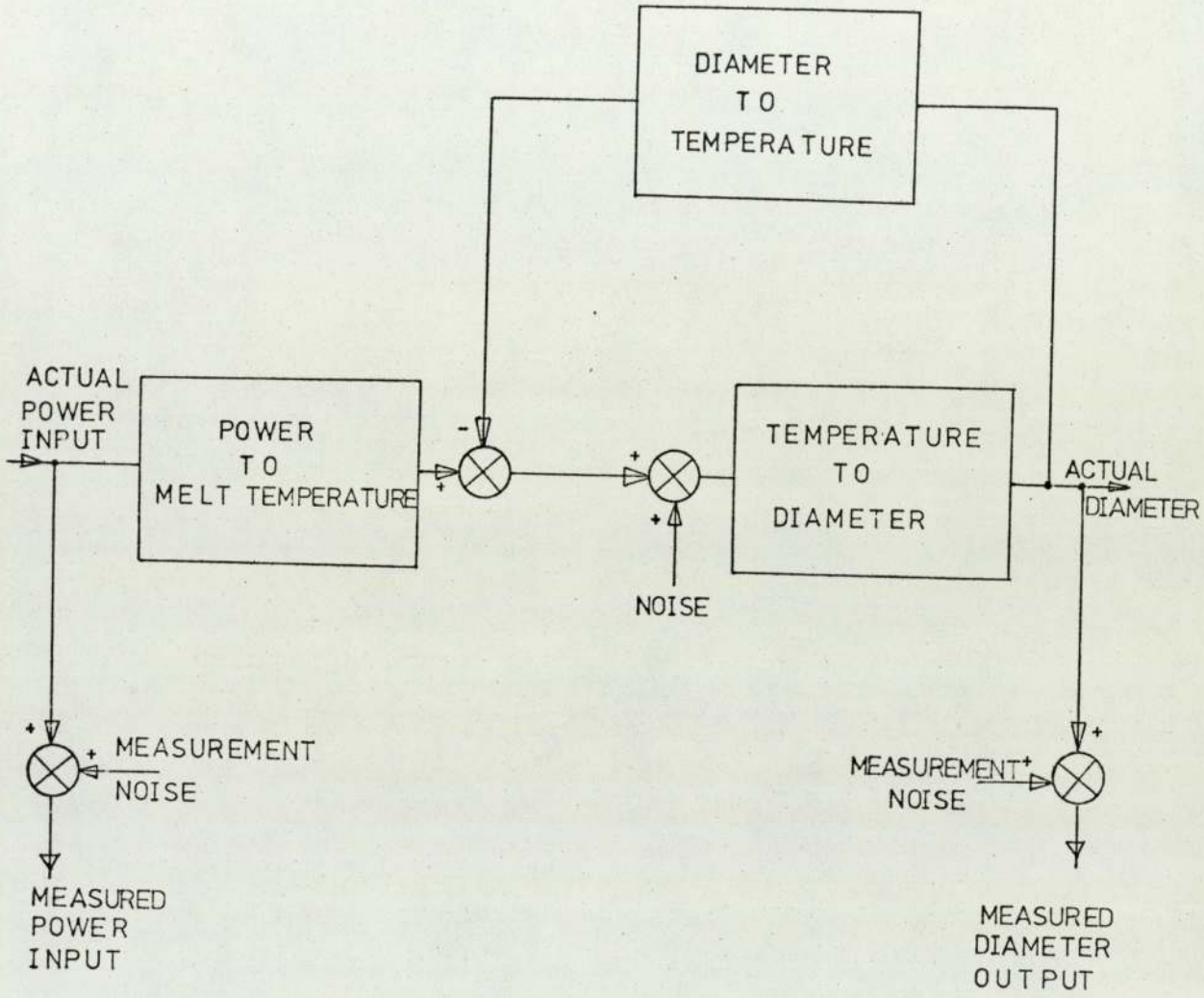


FIGURE 1.3 CRYSTAL GROWTH SYSTEM DYNAMICS



2. THEORETICAL DYNAMICS

2.1 INTRODUCTION

The dynamics of the Czochralski crystal growth system have received scant attention in the literature. This is surprising in view of the interest in automatic control systems[9,10,12,14,25]. There is also considerable literature on both experimental and analytical aspects of steady state Czochralski growth[24,26-34]. The nearest approach to a dynamic analysis of the Czochralski process is by Bardsley et al[10], but this is only concerned with the dynamic relationship between apparent crystal weight and diameter.

In this chapter a transfer function is developed that relates diameter changes to power changes based on heat balances at the solid-liquid interface. This follows the approach already published by Steel and Hill[35]. The dynamics may be conveniently divided into three sections: Power to temperature, temperature to radius, and radius to temperature as shown in Figure 4 of Appendix 1.

2.2 HEAT TRANSFER PROCESSES

The analysis is based on heat transfer mechanisms at the interface. These are very complex, due to the high pressure atmosphere, stirring effects in the melt and the presence of an encapsulant. The steady state temperature field in the crystal has been fairly comprehensively documented. The steady state heat conduction equation has been solved numerically by Sakharov et al[34]. Shaskov et al[32] extend the analysis to include reflected radiation and then use experimental observations to further refine the solution obtained. Experimental observations of the temperature field in a growing Si crystal have been made by Shaskov and Grishin[30] by means of a grown-in thermocouple. Numerical calculations relating particularly to the temperature field at the solid-liquid interface have been made by Arizumi and Kobayashi[36,37]. Analytical solutions to the thermal diffusion equation in the solid are obtained by Brice[38]. Kuo and Wilcox[39] derive analytic solutions for the temperature gradient at the interface. Despite these efforts,

the crystal growth conditions are mainly influenced by the hydrodynamics of the melt and of the growth atmosphere. Kobayashi and Arizumi[29,40] perform numerical calculations to derive the solid-liquid interface shape under steady state conditions. They take into account the melt temperature distribution for the cases of rotating crystal and rotating crucible. The same authors also perform a more comprehensive analysis of the hydrodynamics in the melt[41]. Experimental evidence is available[42], that the crystal growth interface is not stable, even in the short term, with rapid tilt and vertical displacements occurring under steady state growth. Nygren[43] also gives experimental evidence that the growth interface can be either concave or convex to the melt, depending upon the thermal growth conditions. Thus although the assumptions made in the analysis[35] regarding a flat interface and cylindrical interface are at variance with actuality, the errors so introduced are minimised by linearising the radius error about its nominal, steady state value and restricting the applicability to small deviations.

Since Appendix 1 was published, the paper by Kobayashi and Arizumi[41] has appeared. This consists of a numerical solution to the Navier-Stokes equations which govern the forced convection in the melt. Natural convection will also occur, but with normal crucible and crystal rotation rates it will be dominated by the forced convection. The results show that the time for complete circulation of the melt is very small, supporting the assumption made in Appendix 1 that the dynamic lag between power input to the crucible and temperature at the interface may be neglected.

2.3 MENISCUS CURVATURE CONSTANT

2.3.1 General

The meniscus curvature constant, K, is of importance in the time linearisation operation that is performed on the raw crystal data (see Chapter 4), and also in the overall transfer function derived in Appendix 1.

It is shown to be given by:

$$K = \frac{2h_o}{H^2} \qquad \text{..... 2.1 (from Appendix 1)}$$

This is a linearised approximation and a need exists to establish its range of validity. Newman and Searle[44] and Matejevic[45] show that Laplace's capilliary equation leads to:

$$\frac{1}{R_m} - \frac{1}{r} = \frac{\rho h}{\sigma} \qquad \text{..... 2.2}$$

where R_m = meniscus radius

r = crystal radius

$\rho = \rho_m - \rho_b$ = density difference between melt and encapsulant

h = meniscus height

σ = surface tension of melt

As it is intended to arrive at a linear approximation to the relationship between the crystal growth angle and the position of the solid-liquid interface, it is first necessary to assume that the $\frac{1}{r}$ term of equation 2.2 is neglible. That is, the family of meniscus curves implied in equation 2.2 are reduced to one corresponding to that given against a flat plate. The problem will be approached in two stages. In the first, the limits on r to ensure the validity of the "flat plate" approximation will be established. In the second part, the value of the meniscus curvature will be derived together with limits for its validity.

2.3.2 Flat Plate Approximation

Expressing the meniscus curvature in terms of cartesian co-ordinates and using Figure 3 of Appendix 1 as a reference:

$$\frac{1}{R_m} = \frac{\frac{d^2h}{dx^2}}{\left(1 + \left(\frac{dh}{dx}\right)^2\right)^{3/2}} \qquad \text{..... 2.3}$$

let $\frac{dh}{dx} = q$

then $\frac{d^2h}{dx^2} = \frac{dq}{dx} = q\frac{dq}{dh}$

substituting into equation 2.2 and integrating both sides with respect to h yields:

$$\frac{-1}{(1+q^2)^{\frac{1}{2}}} - \frac{h}{r} = \frac{\rho h^2}{2\sigma} + C \qquad \ldots\ldots 2.4$$

where C is a constant of integration

when $h = 0$, $q=0$ and $C=-1$. Let $H = \left(\frac{2\sigma}{\rho}\right)^{\frac{1}{2}}$

with $\theta=0$, $\frac{dh}{dx} = \cot\gamma$

Thus equation 2.4 leads to:

$$\frac{h}{H} = \frac{-H}{2r} + \left(\frac{H^2}{4r^2} + 1 - \frac{1}{(1+\cot^2\gamma)^{\frac{1}{2}}}\right)^{\frac{1}{2}} \qquad \ldots\ldots 2.5$$

writing this more concisely

$$\frac{h}{H} = -\frac{g}{2} + \left(\frac{g^2}{4} + u\right)^{\frac{1}{2}} \qquad \ldots\ldots 2.6$$

where $g = \frac{H}{r}$

and $u = 1 - (1 + \cot^2\gamma)^{-\frac{1}{2}}$

Equation 2.6 is plotted in Figure 2.1 as $\frac{h}{H}$ against $\frac{1}{g}$ for several values of γ. From these curves it may be seen that little error arises in assuming that h is independent of r for values of $\frac{r}{H}$ greater than 3.

Antonov[46] has observed values of γ of 10° to 20° for GaP and Gibbons[47] has observed a steady state value for h of 4mm for GaP.

Using $\gamma = 15^\circ$ and $\frac{r}{H} = 3$ in equation 2.5 yields $\frac{h}{H} = \cdot 71$.

Hence $r = 16\cdot 9$mm

Thus the approximation to a single meniscus curve independent of crystal radius is valid for GaP crystals of greater radius than 16.9mm

2.3.3 Range of Values

Having established the range of validity using the meniscus curve formed by the melt against a flat plate, the range of values of the meniscus curvature constant, K, will be evaluated.

From equation 16 of Appendix 1:

$$\text{Sin}(\theta+\gamma) = \frac{H^2-h^2}{H^2} \qquad \ldots\ldots 2.7$$

$$\therefore \tan(\theta+\gamma) = \frac{H^2-h^2}{(2H^2h^2-h^4)^{\frac{1}{2}}} \qquad \ldots\ldots 2.8$$

From equations 21 and 22 of Appendix 1, assuming negligible $\dot{h}$:

$$\tan\theta = K(h_o-h) \qquad \ldots\ldots 2.9$$

Note that h_o differs from H only as a function of the contact angle, γ (equation 17 of Appendix 1). As a first order approximation, equation 2.9 will be modified to include the contact angle, γ.

$$\tan(\theta+\gamma) = K(H-h) \qquad \ldots\ldots 2.10$$

Combining equations 2.10 and 2.8 yields:

$$K = \frac{H+h}{2(2H^2-h^2)^{\frac{1}{2}}} \qquad \ldots\ldots 2.11$$

Re-writing this in the form:

$$KH = \frac{1+W}{W(2-W^2)^{\frac{1}{2}}} \qquad \ldots\ldots 2.12$$

Where $W = \frac{h}{H}$ and re-writing equation 2.10 in the form:

$$\tan(\theta+\gamma) = KH(1-W) \qquad \ldots\ldots 2.13$$

enables a graph of KH against $(\theta+\gamma)$ to be plotted (Figure 2.2). This shows that over a range of $\theta+\gamma$ between -20° and 50°, KH = 2·06 within 7%. Hence if H = 4.65mm, within this range of $\theta+\gamma$, $K \simeq \cdot 43\text{mm}^{-1}$.

2.4 SYSTEM PARAMETERS

The system constants and derived parameters for Ge, Si and GaP are shown in Table 2.1 together with their error bounds. This is a more comprehensive version of Table 1 in Appendix 1. The error bounds on the GaP parameters are necessarily wider than for the other materials since its physical properties have not been as extensively investigated as with Si or Ge. Also, values of the temperature gradients G_o and G_i are not available for GaP. The values in table 2.1 are therefore estimates. The surface tension for GaP was not available, but this is not necessary since an experimental observation was available for the meniscus height, h_o.

The error bounds for the various derived parameters are calculated as follows:

From equation 40 of Appendix 1,

$$\mu = \frac{\theta_s - \theta_m}{\theta_m - \theta_i} \qquad \ldots\ldots\ldots 214$$

if the error in μ due to errors in θ_s, θ_m and θ_i of $\delta\theta_s$, $\delta\theta_m$ and $\delta\theta_i$ is $\delta\mu$, then

$$\delta\mu = \frac{\partial\mu}{\partial\theta_s}.\delta\theta_s + \frac{\partial\mu}{\partial\theta_m}.\delta\theta_m + \frac{\partial\mu}{\partial\theta_i}.\delta\theta_i \qquad \ldots\ldots\ldots 215$$

This leads to (for worst case errors):

$$\frac{\delta\mu}{\mu} = \frac{1}{\theta_s - \theta_m}\left\{(1+\mu)\delta\theta_m + \mu\,\delta\theta_i + \delta\theta_s\right\} \qquad \ldots\ldots\ldots 216$$

From equation 27 of Appendix 1:

$$P_i = \pi r^2 k_m G_i \qquad \ldots\ldots\ldots 217$$

so the error in P_i is given simply by

$$\frac{\delta P_i}{P_i} = \frac{\delta k_m}{k_m} + \frac{\delta G_i}{G_i} \qquad \ldots\ldots\ldots 218$$

From equation 14 of Appendix 1:

$$P_o = \pi r_o^2 k_s G_o \qquad \ldots\ldots\ldots 219$$

Hence:

$$\frac{\delta P_o}{P_o} = \frac{\delta k_s}{k_s} + \frac{\delta G_o}{G_o} \qquad \ldots\ldots 220$$

From equation 25 of Appendix 1:

$$P_f = \frac{\pi r_o^2 JV\rho_s}{60} \qquad \ldots\ldots 221$$

Hence;

$$\frac{\delta P_f}{P_f} = \frac{\delta J}{J} + \frac{\delta \rho_s}{\rho_s} \qquad \ldots\ldots 222$$

The power radiated, P_r, is given from the heat balance equation at the interface,

$$P_r = P_i + P_f - P_o \qquad \ldots\ldots 223$$

Hence:

$$\frac{\delta P_r}{P_r} = \frac{\delta P_i + \delta P_f + \delta P_o}{P_i + P_f - P_o} \qquad \ldots\ldots 224$$

The power density ratio, Q is given by equation 41 of Appendix 1,

$$Q = \frac{2h_o P_i}{r_o P_r} \qquad \ldots\ldots 225$$

Thus:

$$\frac{\delta Q}{Q} = \frac{\delta h_o}{h_o} + \frac{\delta P_i}{P_i} + \frac{\delta P_r}{P_r} \qquad \ldots\ldots 226$$

The final derived parameter, Φ, is given, from equation 44 of Appendix 1 as:

$$\Phi = \alpha + \frac{2G_i k_m}{G_o k_s}\left[\frac{h_o}{Qr_o} - \frac{\mu(Q-1)}{\left(\frac{R}{r_o}\right)^2 + Q(\mu+1) - 1}\right] \qquad \ldots\ldots 227$$

Which leads to:

$$\delta\Phi = \delta\alpha + \frac{2A}{G_o k_s}\left(G_i \delta k_m + k_m \delta G_i + \frac{G_i k_m}{k_s}\delta k_s + \frac{G_i k_m \delta G_o}{G_o}\right) + \frac{2G_i k_m \delta h_o}{G_o k_s Q r_o}$$

$$+ \frac{2G_i k_m}{G_o k_s}\left((Q-1)B(\mu QB-1)\delta\mu + ((Q-1)\mu^2 B^2 - \mu B - \frac{h_o}{r_o Q^2})\delta Q\right)$$

Where $A = \frac{h_o}{Qr_o} - \mu(Q-1)B$

And:

$$B = \left\{ \left(\frac{R}{r_o}\right)^2 + Q(\mu+1) - 1 \right\}^{-1} \qquad \ldots\ldots 228$$

The transfer function error bounds may now be calculated from equations 44 and 45 of Appendix 1:

$$\Omega = \left(\frac{G_o k_s K V \Phi}{r_o J \rho_s} \right)^{\frac{1}{2}} \qquad \ldots\ldots 229$$

$$\frac{\delta\Omega}{\Omega} \simeq \frac{1}{2}\left(\frac{\delta G_o}{G_o} + \frac{\delta k_s}{k_s} + \frac{\delta K}{K} + \frac{\delta \Phi}{\Phi} + \frac{\delta J}{J} + \frac{\delta \rho_s}{\rho_s} \right) \qquad \ldots\ldots 230$$

$$\zeta = \frac{G_i k_m}{Q} (r_o J \rho_s G_o k_s K V \Phi)^{-\frac{1}{2}} \qquad \ldots\ldots 231$$

$$\frac{\delta\zeta}{\zeta} \simeq \frac{\delta G_i}{G_i} + \frac{\delta k_m}{k_m} + \frac{\delta Q}{Q} + \frac{1}{2}\left(\frac{\delta J}{J} + \frac{\delta \rho_s}{\rho_s} + \frac{\delta G_o}{G_o} + \frac{\delta k_s}{k_s} + \frac{\delta K}{K} + \frac{\delta \Phi}{\Phi} \right) \qquad \ldots\ldots 232$$

$$K_o = \left(\frac{G_i k_m}{G_o k_s \Phi}\right)\left(\frac{\theta_s - \theta_a}{\theta_m - \theta_i}\right)\left[1 + \frac{\mu Q}{\left(\frac{R}{r}\right)^2 + Q - 1} \right]^{-1} \qquad \ldots\ldots 233$$

$$\delta K_o \simeq \left(1 + \frac{\mu Q}{E}\right)^{-1} \left\{ DC\left(\frac{\delta G_i}{G_i} + \frac{\delta k_m}{k_m} + \frac{\delta G_o}{G_o} + \frac{\delta k_s}{k_s} + \frac{\delta \Phi}{\Phi} \right) + D\delta C \right.$$

$$\left. + \left(1 + \frac{\mu Q}{E}\right)^{-1} \left[\frac{DQC\delta\mu}{E} + \frac{\mu DC}{E}\left(1 - \frac{Q}{E}\right)\delta Q \right] \right\} \qquad \ldots\ldots 234$$

With $C = \dfrac{\theta_s - \theta_a}{\theta_m - \theta_i}$

$D = \dfrac{G_i k_m}{G_o k_s \Phi}$

$E = \left(\left(\dfrac{R}{r_o}\right)^2 + Q - 1\right)$

$$\delta C \simeq \frac{1}{\theta_m - \theta_i}\left(\delta\theta_s + \delta\theta_a + C(\delta\theta_m + \delta\theta_i)\right)$$

The large variances on Q, Φ, Ω, ζ and K_o are primarily due to the temperature difference terms, μ and $\dfrac{\theta_s - \theta_a}{\theta_m - \theta_i}$, the errors being unavoidable due to differences between large numbers. This, together with the uncertainty on the values of $\theta_m, \theta_i, \theta_s$ and θ_a tends to make the derived transfer functions rather speculative. The accuracy of the parameters may be improved however by using the measured transfer

function. The approach relies on the assumption that the measured transfer function parameters have smaller error bounds than the most unreliable of the derived parameters of Table T2.1. Equation 419 indicates that Ω=.21 rad/min, ζ=1.14 and K_o= 45.91 Using this observed value of Ω in equation 229 and re-arranging for Φ yields a modified value for Φ of Φ'=1.59. Using this value of Φ and the measured value of ζ in equation 231 yields a modified value for Q of Q'=0.951. This is unrealistic, since as indicated by equation 42 of Appendix 1, a value less than unity implies an unstable mode in the power to diameter transfer function. However, the calculation has served to indicate that the actual value of Q lies somewhere below the value of 2.78 that the analytical derivation indicates. Note that Q is also given by the ratio of power transfer densities at the interface and the surface. The lower value would indicate that more heat loss occurs at the surface than the interface.

TABLE T2.1

Typical parameter values

System constants	G_e	± %	S_i	± %	G_aP	± %	G_aP	± %
r_o crystal radius (cm)	1	*	1	*	1	*	1.5	*
R crucible radius (cm)	2.3	*	4.6	*	4	*	4	*
V growth rate (cm/min)	0.17	*	0.30	*	.033	*	.033	*
J latent heat (J/g)	443	1	1800	1	1500	10	1500	10
k_s conductivity (W/cm K)	0.24	1	0.30	1	.3	1	.3	1
k_m conductivity (W/cm K)	0.71	1	0.67	1	.7	1	.7	1
ρ_s density (g/cm^3)	5.32	1	2.34	1	4.1	2	4.1	2
γ angle of contact (deg)	20	10	20	10	20	10	20	10
σ surface tension (dyne/cm)	620	5	720	5				
α gradient variation factor	1.5	33	1.5	33	1.5	33	1.5	33
G_o gradient in crystal (°C/cm)	103	10	110	10	100	20	100	20
G_i gradient in melt (°C/cm)	30	10	35	10	50	20	50	20
θ_i fusion temperature (°C)	937	.3	1410	.2	1470	.2	1470	.2
θ_m melt temperature (°C)	945	.3	1430	.2	1490	.2	1490	.2
θ_s susceptor temperature (°C)	960	.5	1457	.5	1545	.5	1545	.5
θ_a ambient temperature (°C)	300	10	300	10	300	10	300	10
Derived parameters								
h_o meniscus height (cm)	0.4	10	0.6	10	.5	10	.5	10
K meniscus curvature constant (cm^{-1})	3.2	3.5	2.0	3.5	4.3	3.5	4.3	3.5
μ differential temperature ratio	1.88	130	1.35	60	2.8	45	2.8	45
P_i power input from melt (W)	67.0	11	73.7	11	110	21	247	21
P_o power input to crystal (W)	77.7	11	103.7	11	94.2	21	212	21
P_f power due to fusion (W)	21.5	2	66.2	2	10.6	12	23.9	12
P_r power radiated (W)	10.8	150	36.2	58	26.4	167	59.2	169
Q power density ratio	4.96	171	2.44	79	4.17	198	2.78	200
Φ dimensionless parameter	0.95	219	1.74	51	1.11	130	1.08	151
Transfer function parameters								
Ω natural frequency (rad/min)	0.58	117	0.70	34	.21	78	.17	93
ζ damping ratio	0.19	300	0.20	124	.40	302	.46	314
K_o gain constant	49	420	23	35	40	310	35.8	309

* specified parameters

FIGURE 2.1

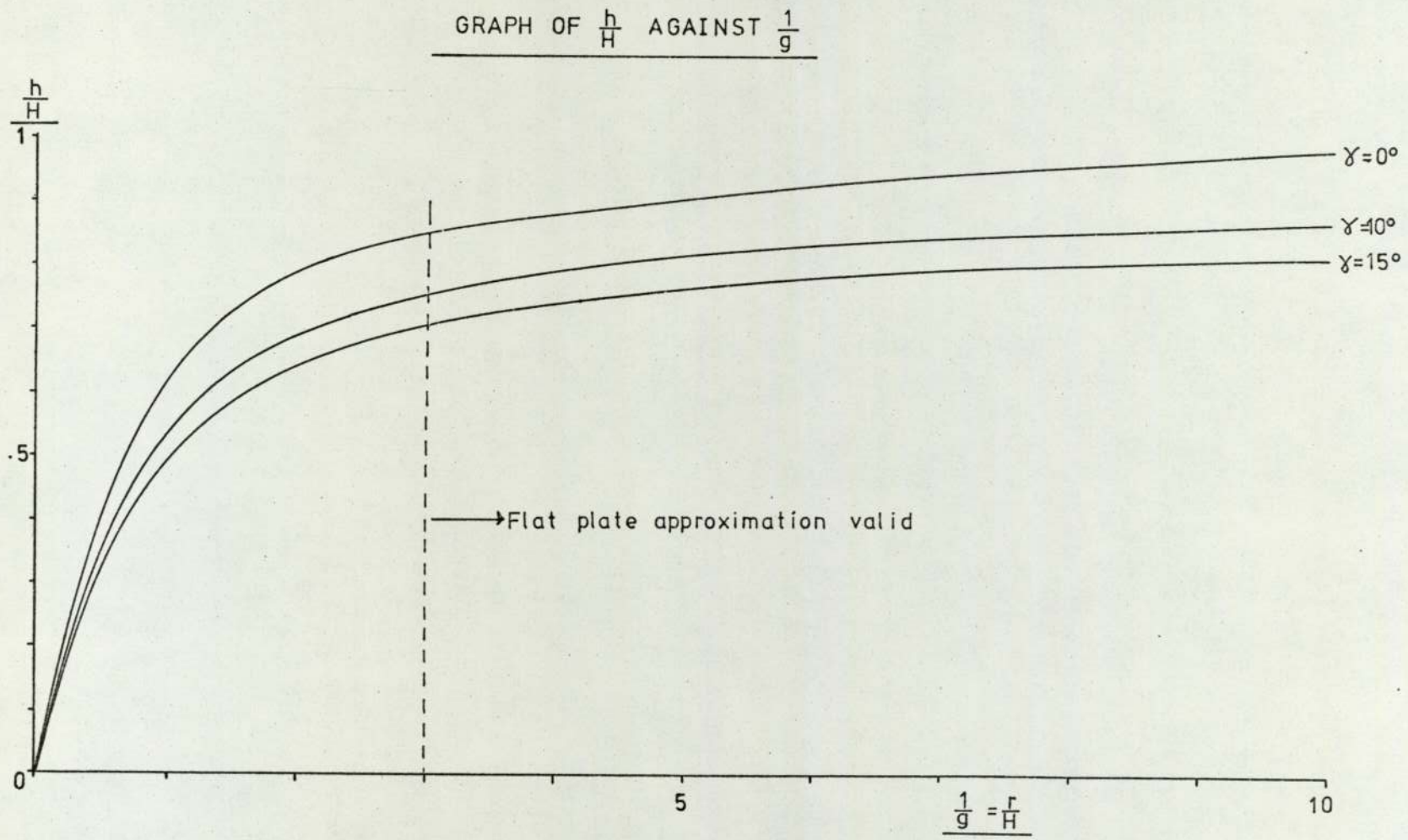
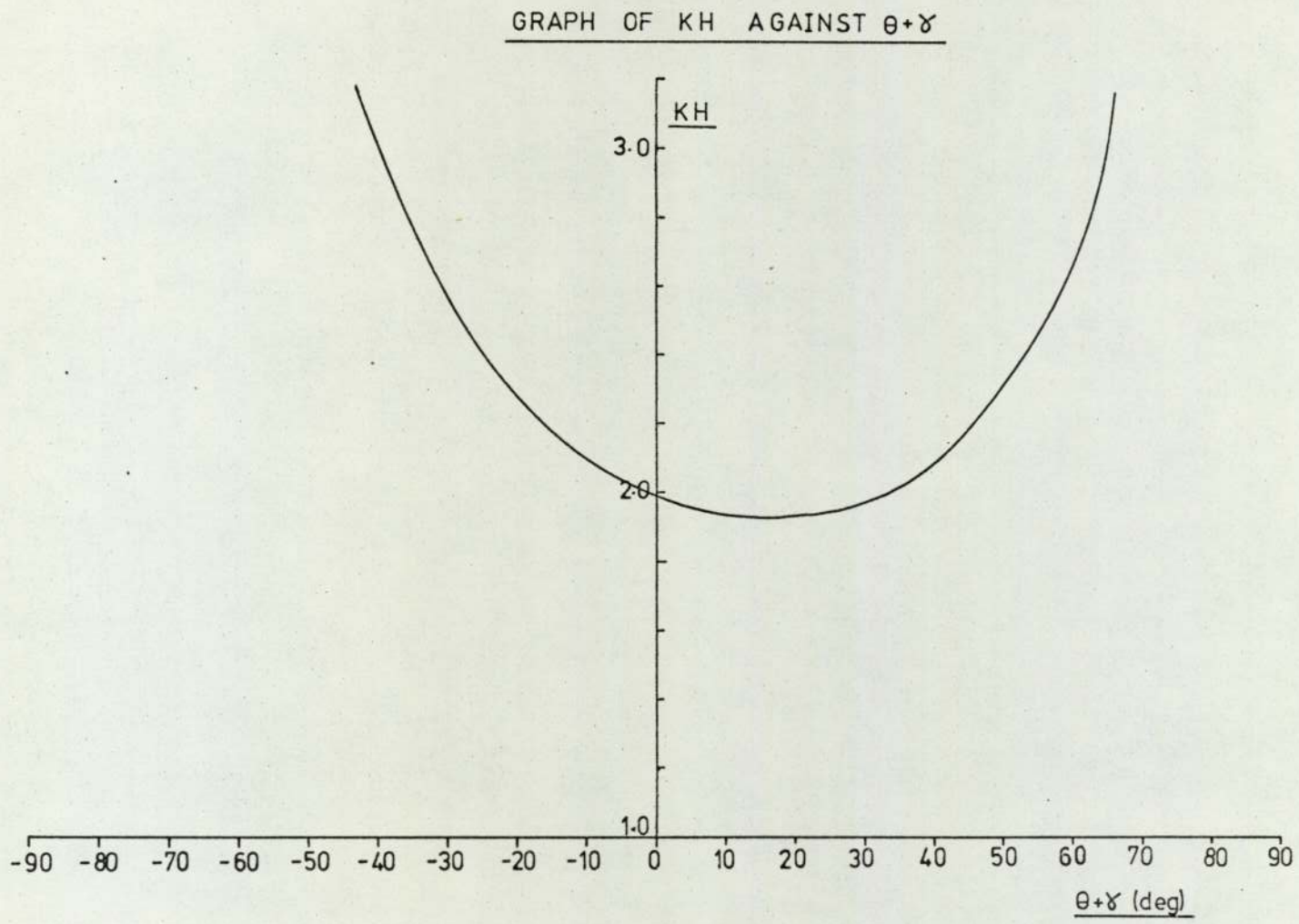

GRAPH OF $\frac{h}{H}$ AGAINST $\frac{1}{g}$
$\frac{h}{H}$
1
.5
0
5
10
$\frac{1}{g} = \frac{r}{H}$
γ=0°
γ=10°
γ=15°
Flat plate approximation valid

FIGURE 2.2


GRAPH OF KH AGAINST θ+γ
KH
3·0
2·0
1·0
-90 -80 -70 -60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90
θ+γ (deg)

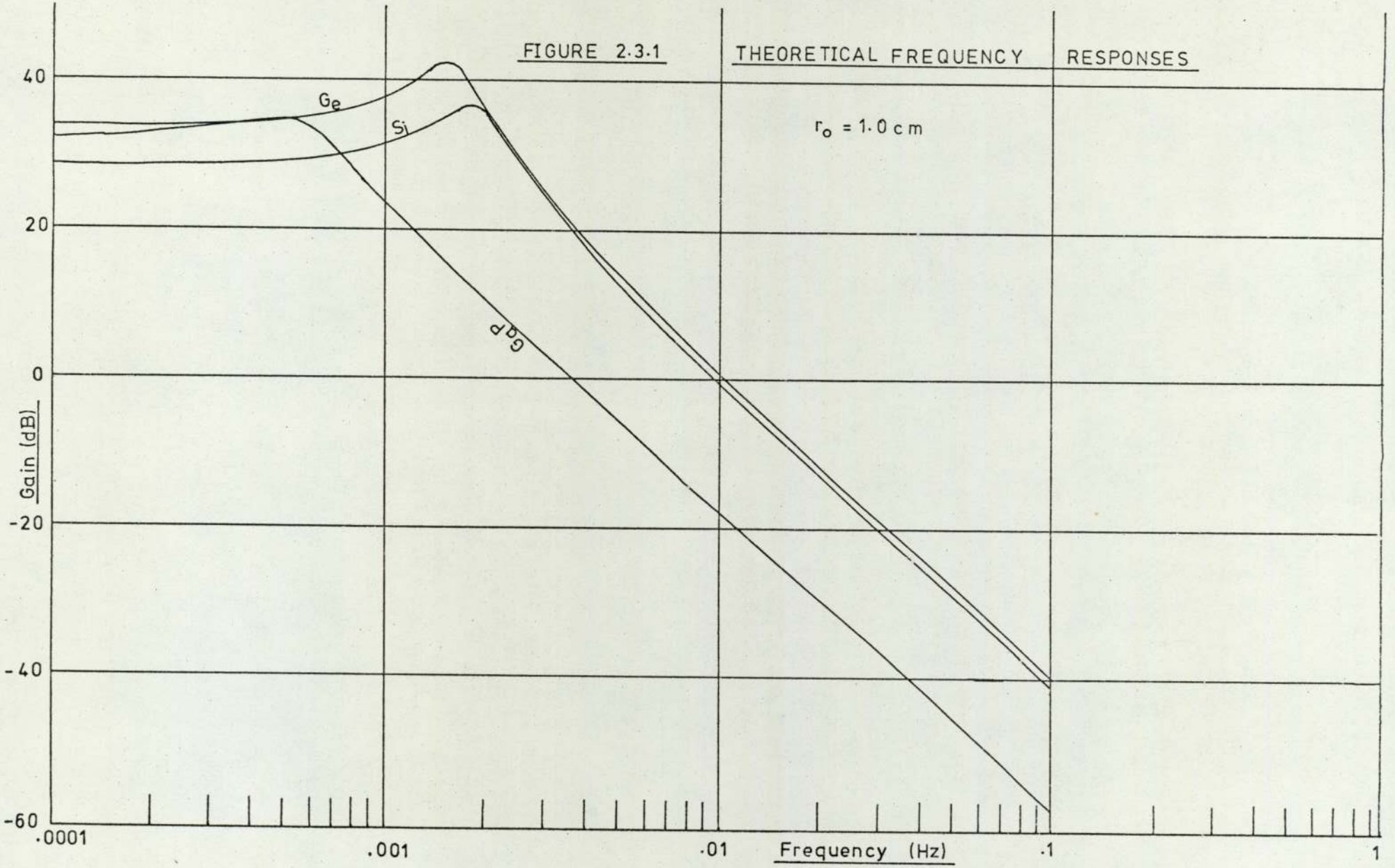


FIGURE 2.3.1
THEORETICAL FREQUENCY RESPONSES
r_o = 1.0 cm
Ge
Si
GaP
40
20
0
-20
-40
-60
Gain (dB)
.0001
.001
.01
.1
1
Frequency (Hz)

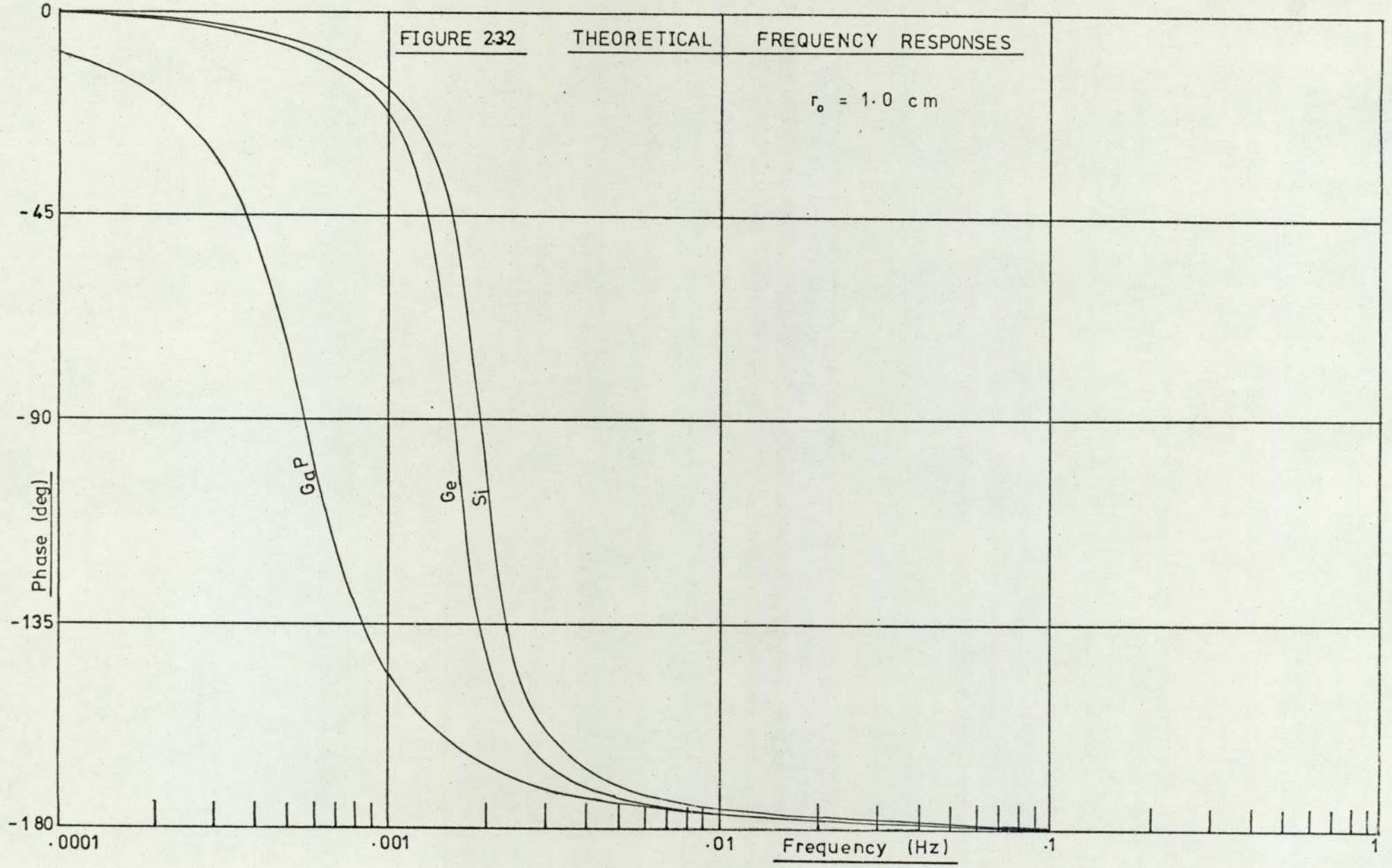


FIGURE 2.2 THEORETICAL FREQUENCY RESPONSES
r_o = 1.0 cm
0
-45
-90
-135
-180
Phase (deg)
GaP
Ge
Si
.0001
.001
.01
.1
1
Frequency (Hz)

3. EXPERIMENTAL PROCEDURE

3.1 ARRANGEMENT OF CRYSTAL PULLER

A type MSR6 pressurised crystal puller was made available by Metals Research Ltd for the experimental work. This was an RF heated puller, capable of taking a charge of 250gm. A systematic diagram of the puller is shown in Figure 3.1 and a picture of the puller in Figure 3.2. As can be seen from the picture, the viewing ports are inclined at approximately 30° to the vertical which gives a very restricted view of the growing crystal. Since encapsulation is used, the encapsulant, B_2O_3, very soon becomes opaque thereby obscuring the interface region of the growing crystal. Only one thermocouple is fitted, this being positioned in the base of the susceptor, 1mm from the bottom of the crucible. The pull rod has a position indicator with a digital read-out in units of ·0238mm and can be zeroed (eg at seed on). A contact device is fitted which enables contact between the seed crystal and the melt to be sensed. The pressure chamber is filled with nitrogen at a typical running pressure of 8×10^6 N/m^2. This pressure is not automatically controlled but for the duration of a typical crystal growth run it remains within 5% of its initial value. Crucible lift and rotation facilities are available although lift is not normally required, and was not used in any of the experiments. The pull bar also has lift and rotation facilities. Both the crucible and the pull bar movements are controlled by constant speed servo systems. The RF power input to the susceptor is controlled by a "Wattmaster" constant power regulator. This is fitted with a control potentiometer and may also be remotely controlled by a voltage signal. A closed circuit TV system exists for monitoring the progress of growth, the camera is pointed down one of the viewing ports.

3.2 EXPERIMENTAL SYSTEM

3.2.1 Equipment

This consisted of pseudo-random signal injection equipment. A special purpose signal generator had been built by the staff of

the University of Aston. This provided facilities for generating binary, inverse repeat binary and ternary pseudo-random sequences. The characteristic polynomials for the sequences could be specified. This signal generator required an external clock signal and this was provided by a Servomex LF141 signal generator. An interface unit to provide variable amplitude and variable offset facilities for the pseudo-random signal generator was designed and built. These three items are shown in Figure 3.3. A Solartron JM1861 pseudo-random signal generator was also available. This had a more restricted range of facilities and could only produce binary or inverse repeat binary pseudo-random sequences. It had an internal clock generator but this had widely spaced, fixed frequencies.

3.2.2 Choice of Test Signal

The necessity to inject some form of test signal into the system has been established in Chapter 1. There remains the choice of test signal to be made. Step or impulse inputs are precluded immediately in that any attempt to perform a linear system estimation about a mean operating point would fail because of the large input needed. (Such a signal injection of sufficient amplitude to be measurable in the output could well be fatal for the growth process). Sine wave testing is precluded with a batch process such as this; numerous crystals would need to be grown with different frequency inputs just to produce one frequency response. 'Gaussian' noise injection is precluded by the batch process limitations on the time that data can be gathered. The one remaining class of input signal, a pseudo-random sequence[47], is eminently suitable. The crystal growing process is a classic example of a requirement for pseudo-random signal testing in that a low amplitude of disturbance is mandatory and the time in which data can be collected is limited. However, most of the literature describing the use of pseudo-random sequences has been on

continuous processes[49,50,51]. Provided that enough data can be collected during the batch process then there is effectively no difference between the two cases. However, it is likely to be the case that in order to arrive at a reliable system estimate more data is required than is available. An ergodic hypothesis must therefore be adopted and the results from several runs regarded as an ensemble, with which the system estimate may be computed.

3.2.3 Model Structure

The crystal growth system dynamics are shown in block diagram form in Figure 1.3. The only intermediate stage in the dynamics is that of power to melt temperature. It is possible to perform a separate identification of this part of the dynamics by relating temperature changes to power changes. Unfortunately, the other half of the dynamics, temperature to diameter, is not separately identifiable. However, its dynamics may be deduced by measuring the overall dynamics and then subtracting the power to melt temperature dynamics. Note that if temperature measurements are made without a growing crystal present, then the diameter to temperature dynamics will not influence the result. There is therefore a case for performing a system identification experiment on the crystal puller without a crystal. However note that the melt temperature itself is not measurable, although it may be approximated by the temperature measured at the base of the susceptor. A further possibility exists, that of performing tests on the crucible both with and without a charge of GaP and encapsulant. The thermal capacity of the crucible would be expected to be increased by the presence of the charge and its heat dissipation characteristics would be expected to be influenced by the presence of the B_2O_3 layer.

3.2.4 Test on Crucible Alone

Although an empty crucible was being used (but with normal gas pressure and coolant flow rates etc), the dynamics would still be

expected to exhibit non-linear behaviour due to the mode dependence of the thermal response . (The crucible is likely to cool with a shorter time constant than that with which it heats up). Accordingly, an anti-symmetric sequence was chosen because of the property of non-correlation with even order harmonics[52]. This would enable a linear estimate to be made of a non-linear system. The transient of temperature produced by a step-change in power input had been observed to be settled in 5 minutes. To give ample time for the system impulse response to have decayed, a sequence of 10 minutes period was chosen. To provide an upper identifiable frequency that would reasonably be above any system noise, a bit frequency of ·3Hz was chosen. The sequence was now defined, since the inverse repeat sequence nearest to these two parameters has $2(2^7-1)$ bits (= 254). In this initial experiment, there was no limit to the length of data that could be collected. (Unlike the later experiments, where the data was to be limited by the length of crystal that could be grown). The pseudo-random signal generator was coupled directly into the power controller of the crystal puller. The peak to peak amplitude was set to 5.25 % of mean power. (This was considerably higher than the change used for the growth of a normal crystal). The test signal was run for three complete periods after a settling time of a half-period and the thermocouple output was recorded on a chart recorder for subsequent analysis. (It was thought that the high RF fields ambient to the crystal puller would make the use of a digital data logger impractical). The resultant analogue chart recording was hand digitised on to standard PDP9 punched tape, the thermocouple output being sampled at the bit frequency of the test signal, and in synchronism with it. The method of overlapping segment fast Fourier transformation was employed, (Carter, Knapp and Nuttal[53]), to evaluate the mean frequency response function from one period of inverse repeat pseudo-random signal and from three periods of

system output. An overlap fraction of 0·5 was used, yielding effectively five periods of overlapped data. The frequency response function so produced was then compensated for the effects of being preceded by a zero order hold, thus providing an estimate of the frequency response of the continuous system represented by sampled data. The resultant frequency response function (gain and phase) is shown in Figures 3.4.1 and 3.4.2. The coherence function is shown in Figure 3.5. The phase and the coherence function together indicate a very low correlation between the input and output signals above ·02Hz. The (almost) linear phase characteristic up to this frequency indicates the presence of a time lag between the input and output signals. In view of these factors a transfer function was fitted to the data by the method of fitting to modulus only, described in Chapter 5. The phase information was replaced by a negative-going, linear phase characteristic (initial phase), in order to start the iteration. The resultant fitted frequency responses are shown in Figures 3.6.1 and 3.6.2. The (first order) transfer function that was fitted was scaled to have units of $^{o}C/\%$ change of power and was:

$$H(s) = \frac{2.60}{52.08s+1} \qquad \ldots\ldots\ 301$$

with a sum of squares of error of fit = 0·061. (See description of FFIT in Chapter 5). The measured phase characteristic indicated a delay term of e^{-68s}.

3.2.5 Test on Crucible With Charge

A second experiment was devised to measure the dynamics of the susceptor and crucible with a melt present. It was further planned to go on to grow a crystal with a test signal superimposed. Since the first experiment had indicated that frequency components of the test signal above ·02 Hz were not detectable in the output and the presence of the melt would be expected to lower the natural frequency of the system, a longer bit period was indicated. The next increment of bit period that was available on the JM1861 was

0.1 Hz. This, with the same sequence as for the first experiment, but peak to peak amplitude of 3% of mean power, gave a sequence of duration 42.7 minutes. The sequence was run for a half period, as settling time, then for a further two periods. The data was processed in the same manner as for the first experiment except that since only two periods of data were available, only 3 overlapping segments could be used[53] with an overlap fraction of 0.5. The frequency response function is shown in Figures 3.7.1 and 3.7.2 and the coherence function is shown in Figure 3.8. A first order lag was fitted to the modulus, as for the first experiment, and the delay term was evaluated from the measured phase characteristic. The transfer function fitted was scaled to have units of $^{o}C/\%$ change of power and was:

$$H(s) = \frac{3.53}{49.75s+1} \qquad \ldots\ldots\ldots 302$$

with a sum of squares of error of fit = 0.120. The delay term was e^{-91s}. The fitted frequency response is shown in Figures 3.9.1 and 3.9.2. The delay terms were evaluated at the lowest frequency in the empty crucible test sequence,(.00118Hz). The difference between the measured and the fitted dynamics was assumed to be due to the delay term. The differences in gain between the two cases are possibly an indication of the non-linearity of the thermal process, the larger input signal for the empty crucible giving a lower gain. A further factor which could increase the gain would be the insulation afforded by the B_2O_3 layer, enabling a higher steady state temperature to be attained at a given power level. However, the B_2O_3 would also be expected to influence the time constant of the system, this is not evident between the two delays of the empty and full crucibles. It is concluded that the effect of the melt and B_2O_3 is insignificant, the differences in gain being due to non-linearity. The delay times must be considered unreliable since they are based on only 1 observation. The longer delay time with the smaller input signal is consistent with non-linear thermal behaviour.

3.2.6 Test on Crystal Growth

An initial crystal growth experiment was devised to measure the dynamics between power input and crystal radius. The same inverse repeat binary pseudo-random sequence was used as for the tests on the crucible. A crystal was seeded and at a pull speed of 2cm/hr was necked out to approximately 1cm diameter under manual power control. At this point the pseudo-random input signal was applied at 2.62% peak to peak amplitude Input power control was retained, (with the PRS superimposed), until the crystal finally broke at its seed-on point 141 minutes after starting the PRS. The resultant crystal showed no evidence at all of a diameter perturbation of high enough frequency to link it with the power perturbation. It was concluded that the bit period of the PRS used was too short for the dynamics of the crystal growth process to respond to it. In the choice of test signal, a dichotomy of interests exists, in that the low frequency end of the range should be taken down as low as possible in order to "illuminate" the dominant modes, while the period of the test signal should be as short as possible for two reasons:

1. The test signal should, at worst, be short enough to allow a half period settling time and one full period in the duration of the crystal run.

2. If possible, several periods of test signal should be accommodated in the crystal length to allow the dynamics to be separately identified at the beginning and end of the run.

A typical crystal grown in the MSR6 puller could take approximately three hours to grow (at 2cm/hour). Obviously the smaller the crystal diameter, the longer the growth time but in practice manual control to produce thin crystals is very difficult, the system seems to have an inbuilt bias to grow crystals of approximately 3cm diameter. The maximum length test sequence that can be accommodated on the crystal is therefore two hours. This leaves half a period as settling time and the remaining two

hours for identification. It has been demonstrated that a bit period of 10 seconds was too short for the crystal dynamics and theoretical considerations (Chapter 2) have indicated that a Nyquist frequency one tenth that given by a ten second bit period would be adequate. It was thought to be advantageous to use a ternary pseudo-random sequence. This gave the possibility of performing a non-linear estimation of the system at some later date (Barker, Obidegwu[54]). The overall period and bit frequency requirements were met by a fourth order ternary sequence with a bit period of 90 seconds. (This is 80 bits long and hence a complete period lasts 120 minutes). The magnitude of perturbation necessary to produce a measurable response was difficult to estimate, this being limited by the need to operate in the linear range without impairing the general growth conditions. A further point in favour of using a ternary sequence is that with a seventh order inverse repeat sequence, the longest time spent in one state is 8 bits but a fourth order ternary sequence only spends a maximum of four bits at an extreme. At this stage, no further systematic approach could be used to specify a 'safe' amplitude, so on purely empirical grounds the amplitude was set at a level that an experienced crystal grower thought was the maximum possible without causing the crystal to melt off or freeze out. Thus the next crystal was grown in the same manner as the first, but with a ternary pseudo-random power perturbation of 1.5 % peak to peak amplitude. There was very little evidence of diameter fluctuations. There were visible striae on the crystal but these were immeasurable by the dial gauge method (see Section 3.2.7). However, more seriously there was not a complete period of data available. Lamb and Rees[55] have shown that serious errors are introduced into system frequency response estimates by truncating or increasing the length of data streams obtained from pseudo-random sequences so that there was little

information to be gained from this crystal other than that the power perturbation was still too small. For the next crystal the power perturbation was increased to 2.75% peak to peak. This level of perturbation was satisfactory as a measurable signal was produced on the diameter profile of the finished crystal. However, as shown in Chapter 4 there was still insufficient data available to enable a transfer function estimate to be made. A subsequent crystal was grown at a slower growth rate and this yielded enough data to enable a transfer function to be estimated. A photograph of these last two crystals is shown in Figure 3.10.

3.2.7 Measurement of Crystal Diameter

The profile of the cold crystals was measured by means of a dial gauge and a lathe. The method is shown in Figure 3.11. The lathe provided a convenient vehicle for mounting the crystal and enabling the dial gauge to be incremented along the length of the crystal. The probe of the dial gauge was especially ground to a radius of ·01cm. The dial gauge itself was calibrated in increments of ·003cm but the reading could be estimated to ·0003cm. Readings were taken along the length of the crystal at intervals of ·005cm, starting at the seed-on point. The measured profile was then converted to diameter values by measuring the largest diameter on the crystal with a micrometer and applying a datum shift to the other profile measurements.

FIGURE 3.1

Crystal Puller Systematic Diagram

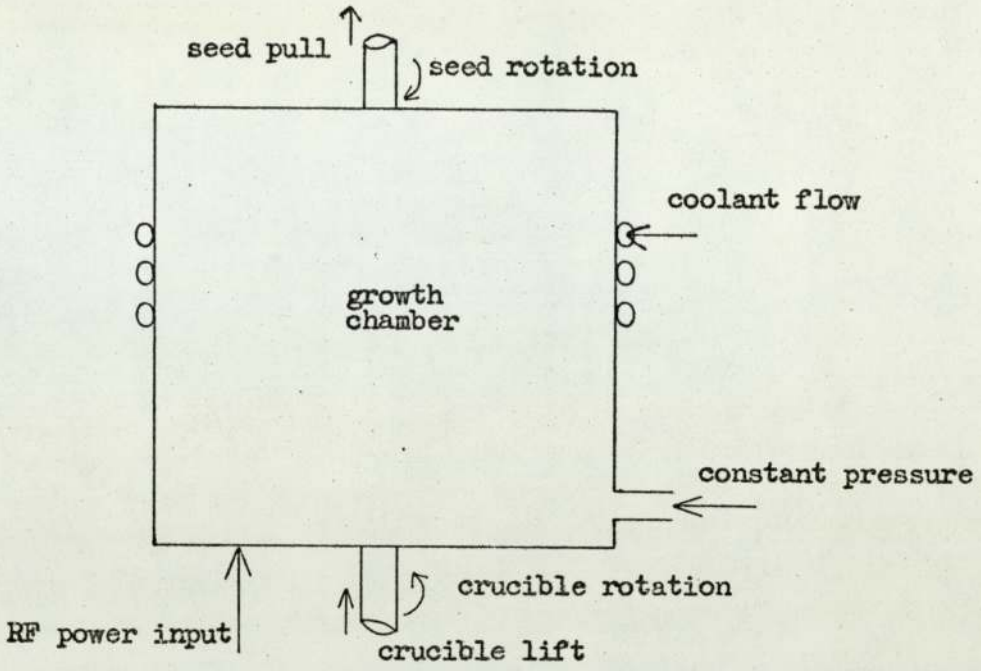


FIGURE 3.2

Crystal Puller Picture

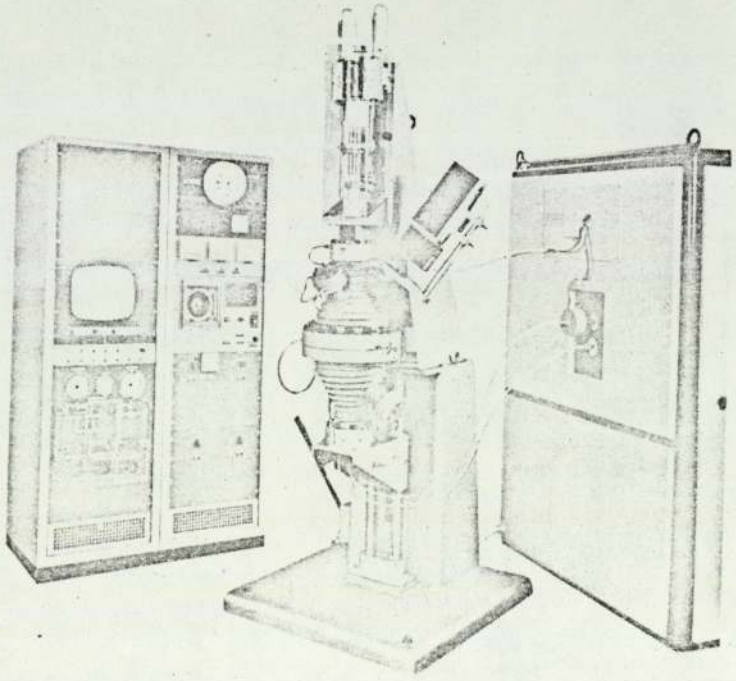


FIGURE 3.3 PSEUDO-RANDOM SIGNAL INJECTION EQUIPMENT

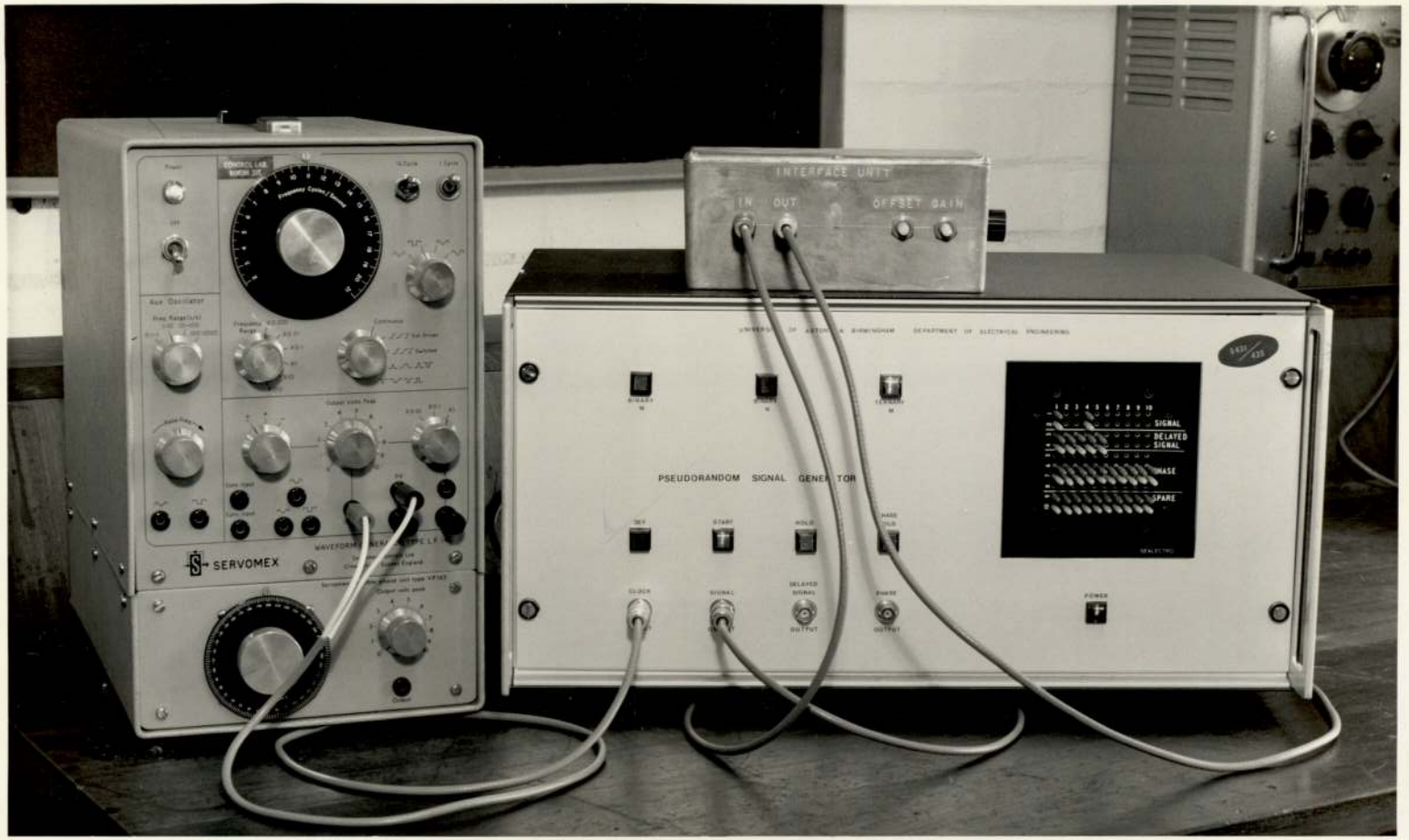


Figure 3.4.1 Frequency Response Function Of Empty Crucible (Modulus)

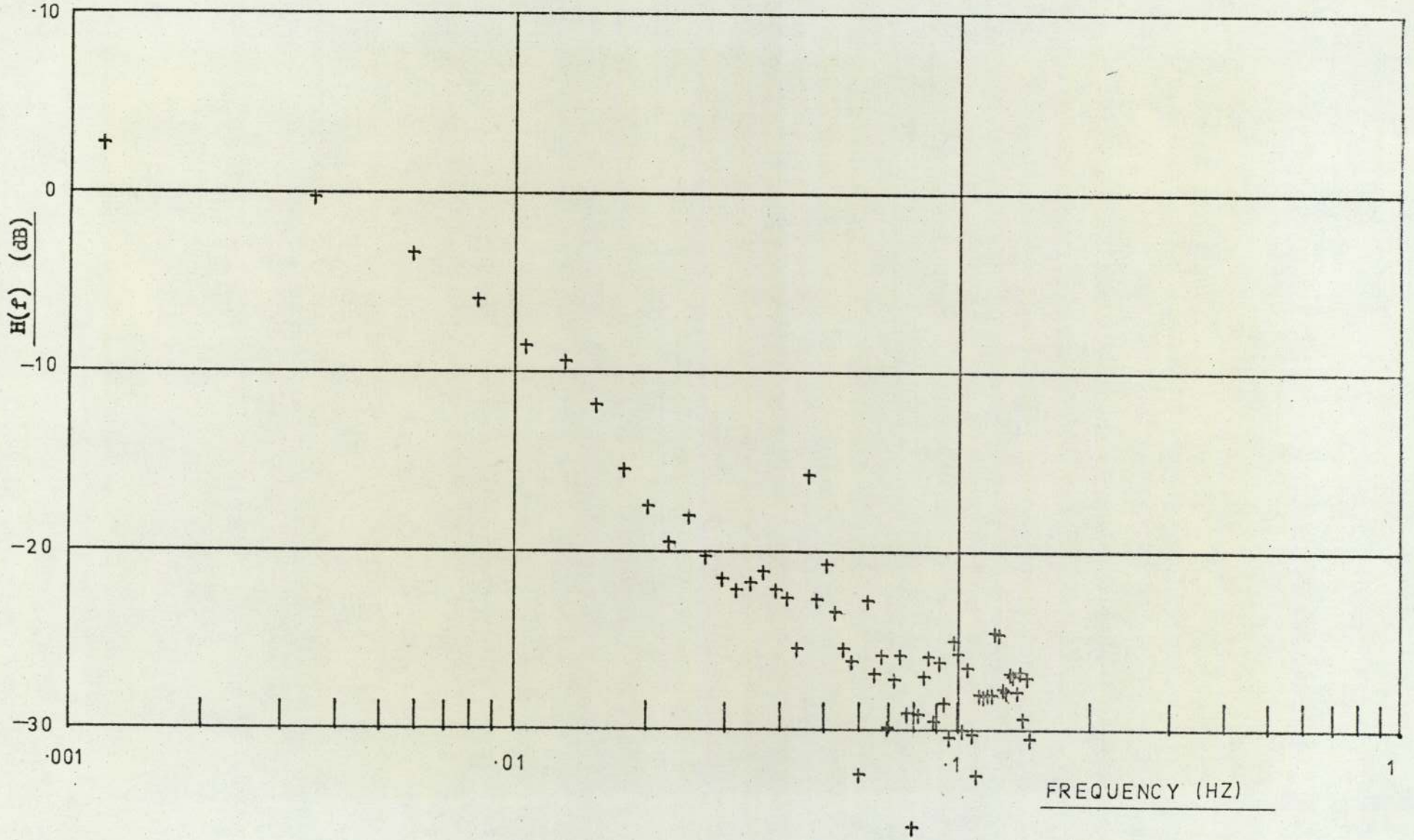


Figure 3.4.2 Frequency Response of Empty Crucible (Phase)

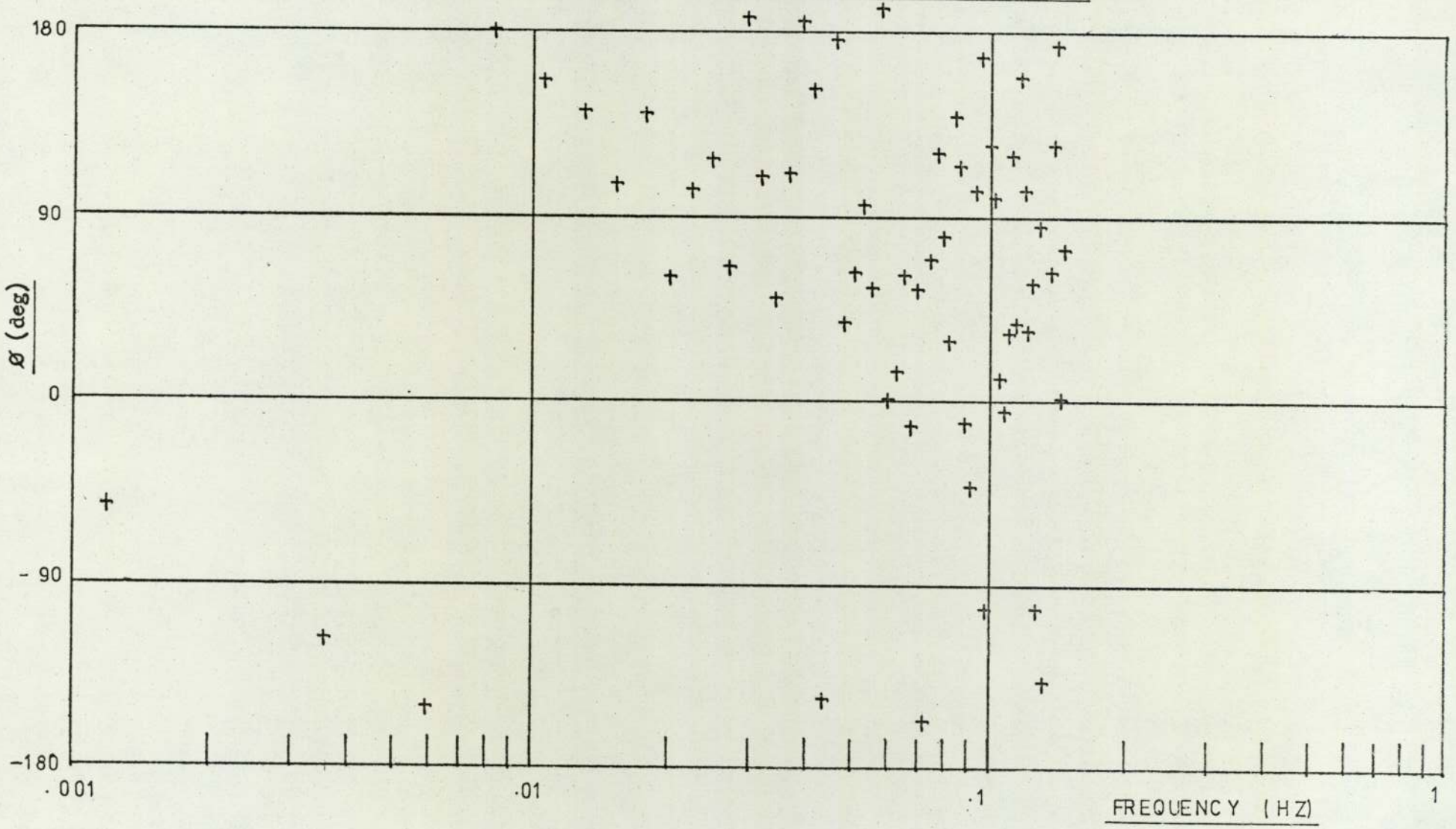


Figure 3.5 Coherence Function of Empty Crucible

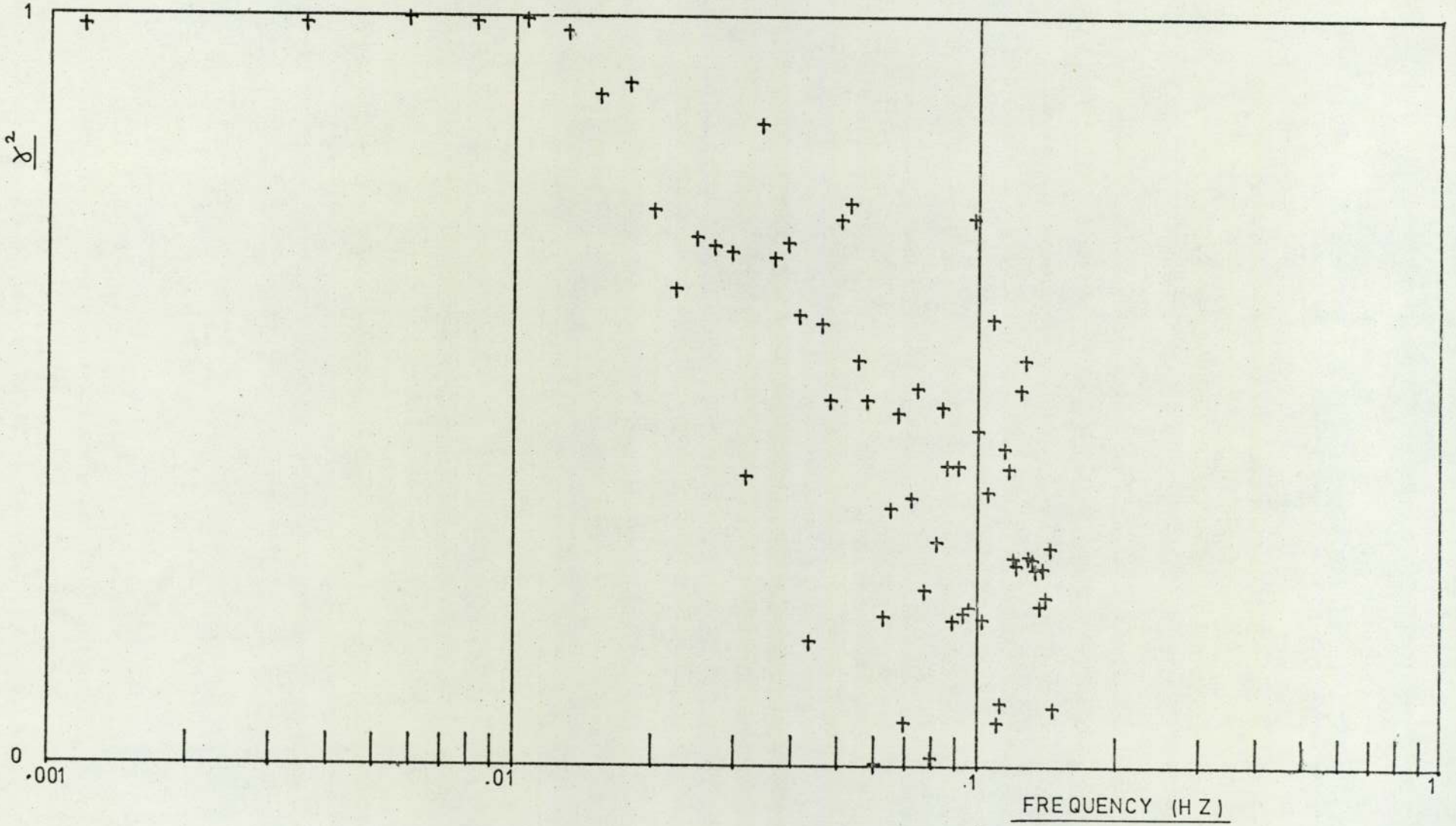


Figure 3.6.1 Fitted Frequency Response of Empty Crucible (Modulus)

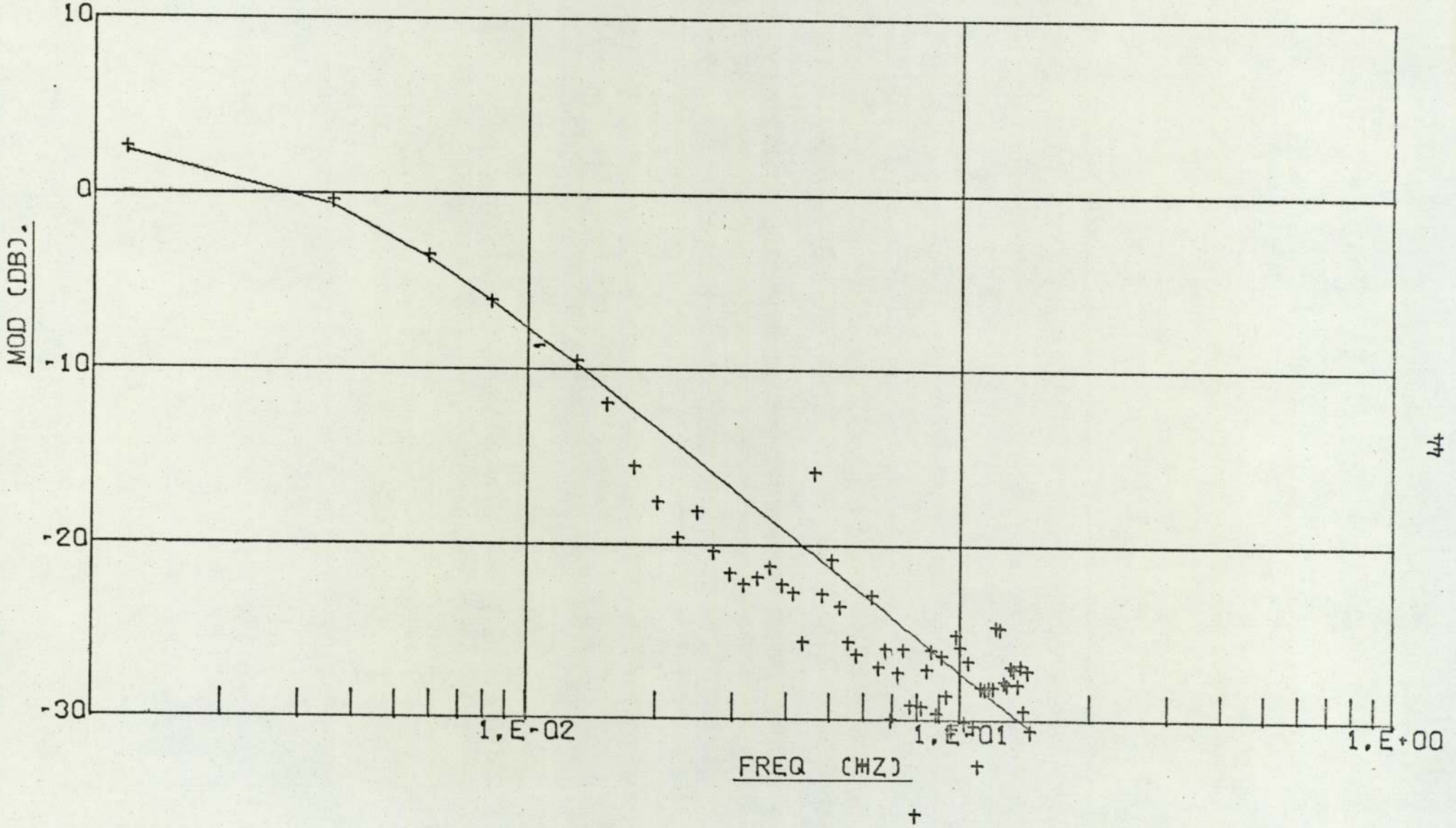


Figure 3.6.2 Fitted Frequency Response of Empty Crucible (Phase)

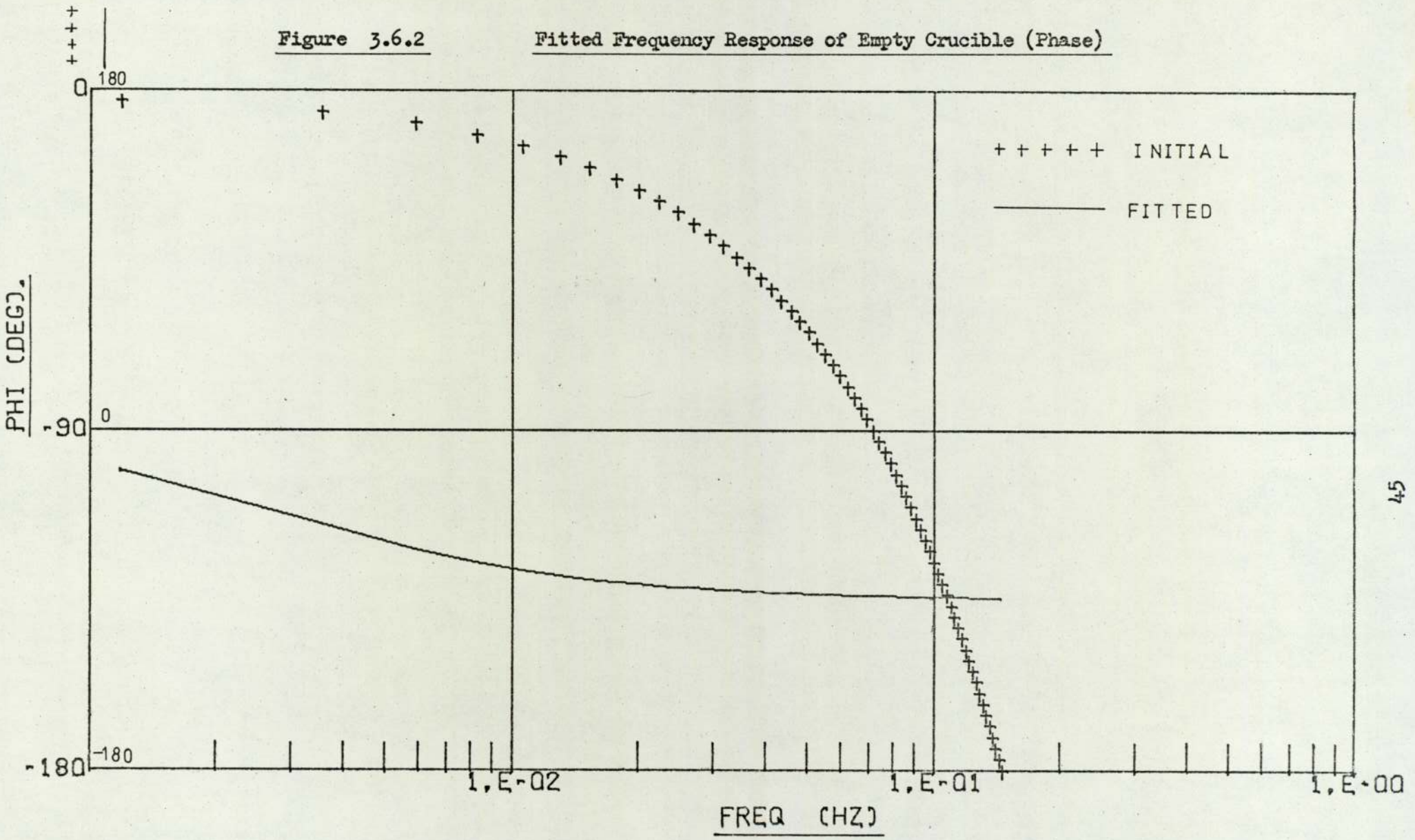


Figure 3.7.1 Frequency Response Function of Full Crucible (Modulus)

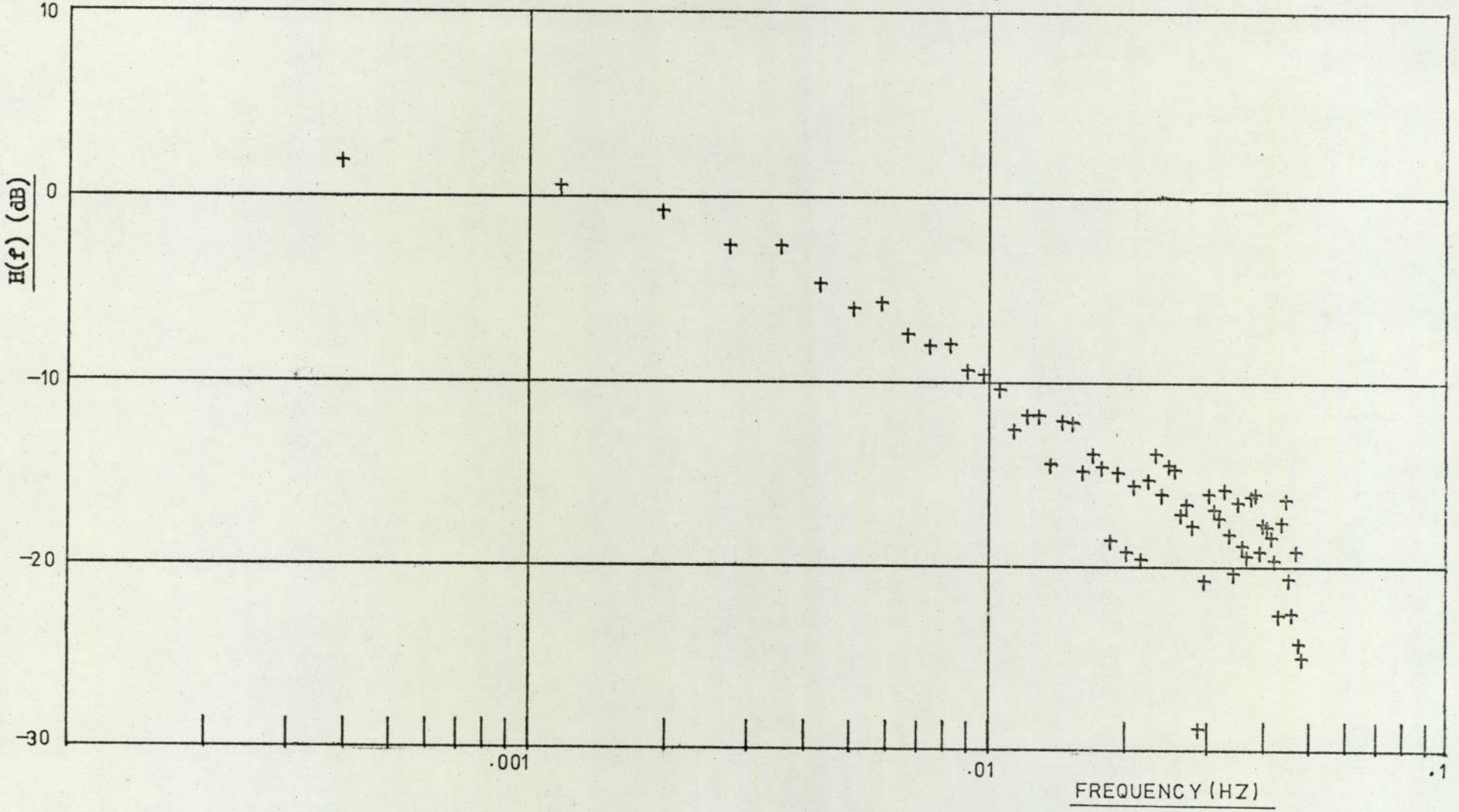


Figure 3.7.2 Frequency Response Function of Full Crucible (Phase)

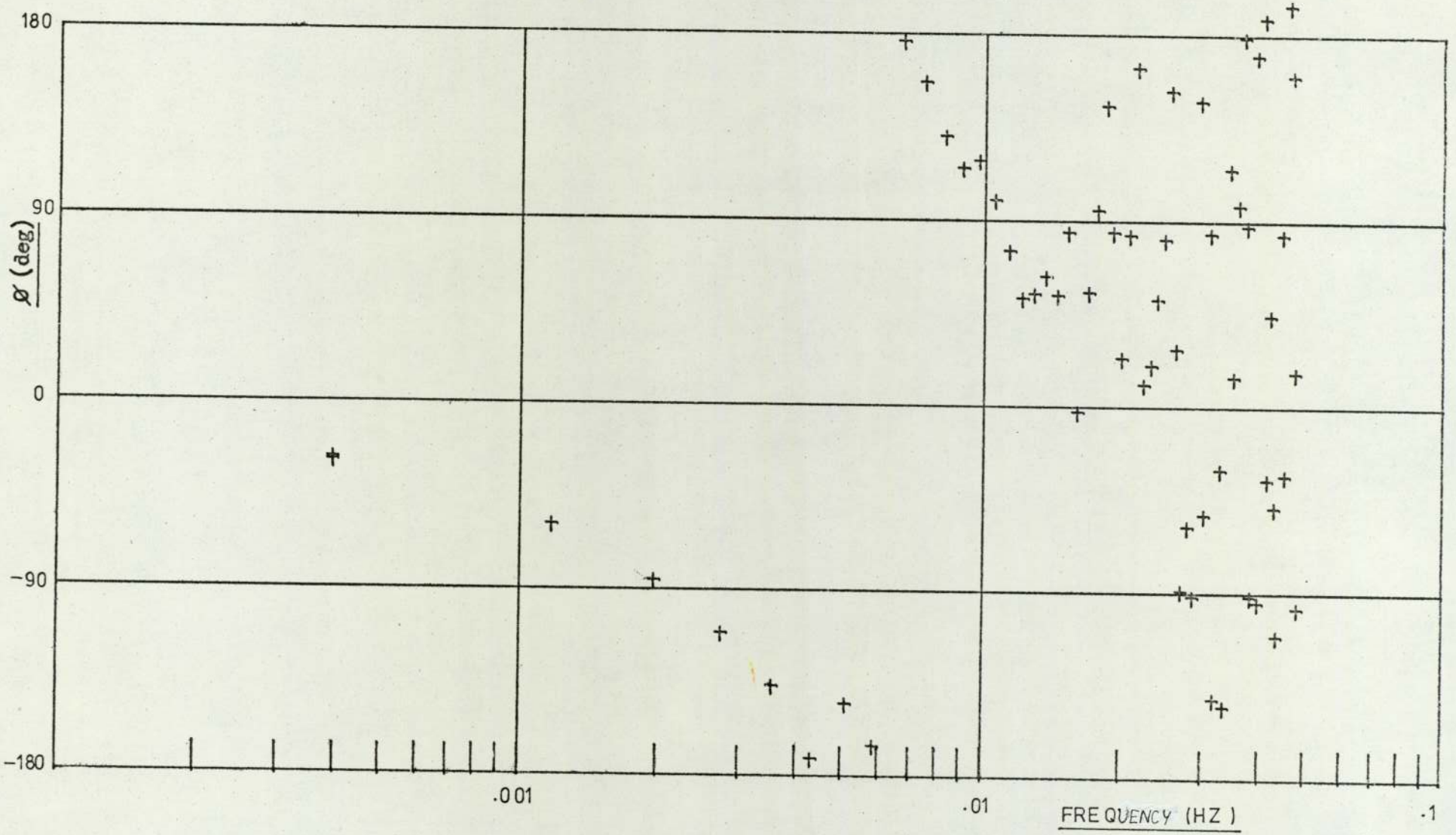


Figure 3.8 Coherence Function of Full Crucible

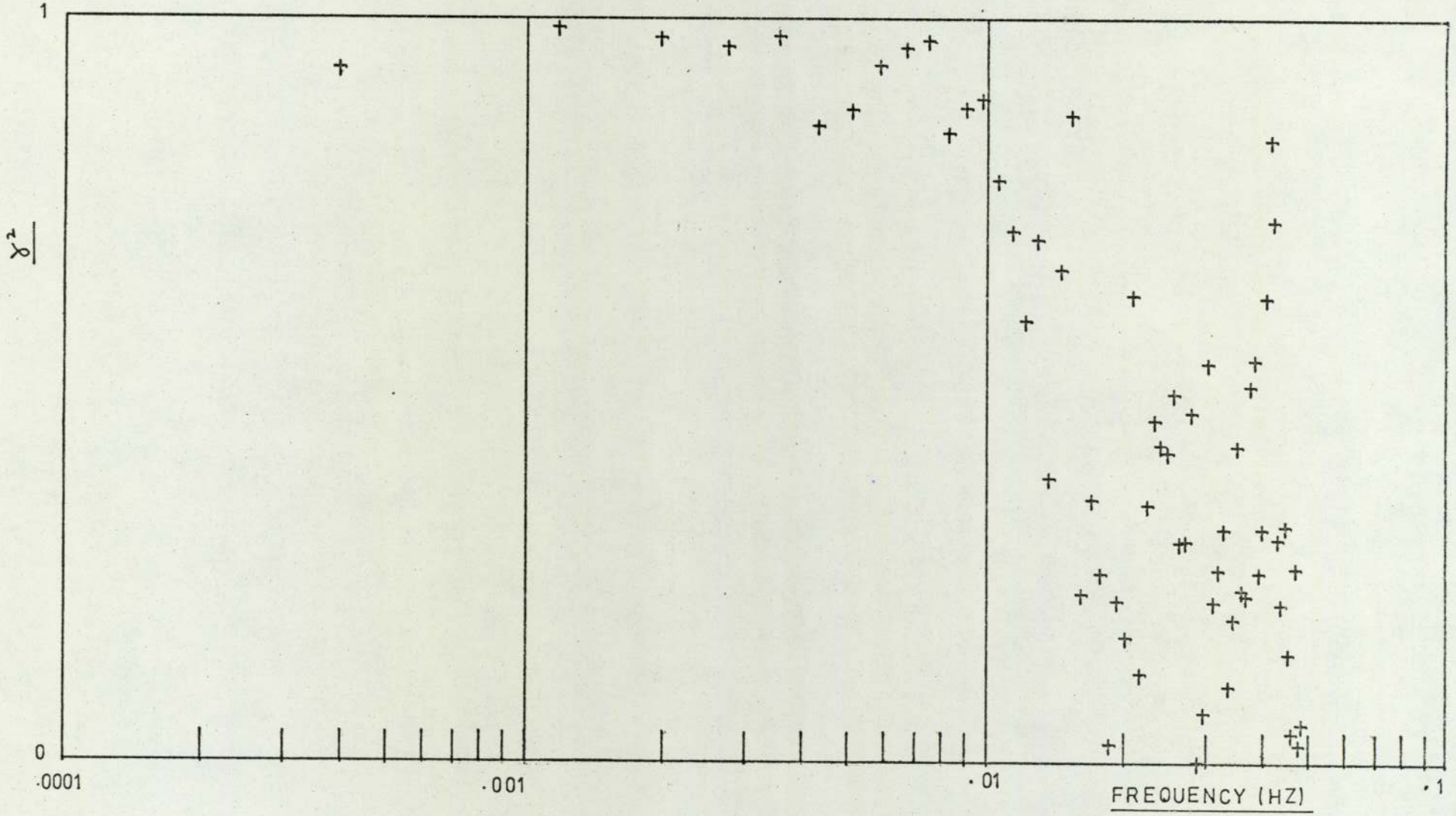


Figure 3.9.1 Fitted Frequency Response Function of Full Crucible (Modulus)

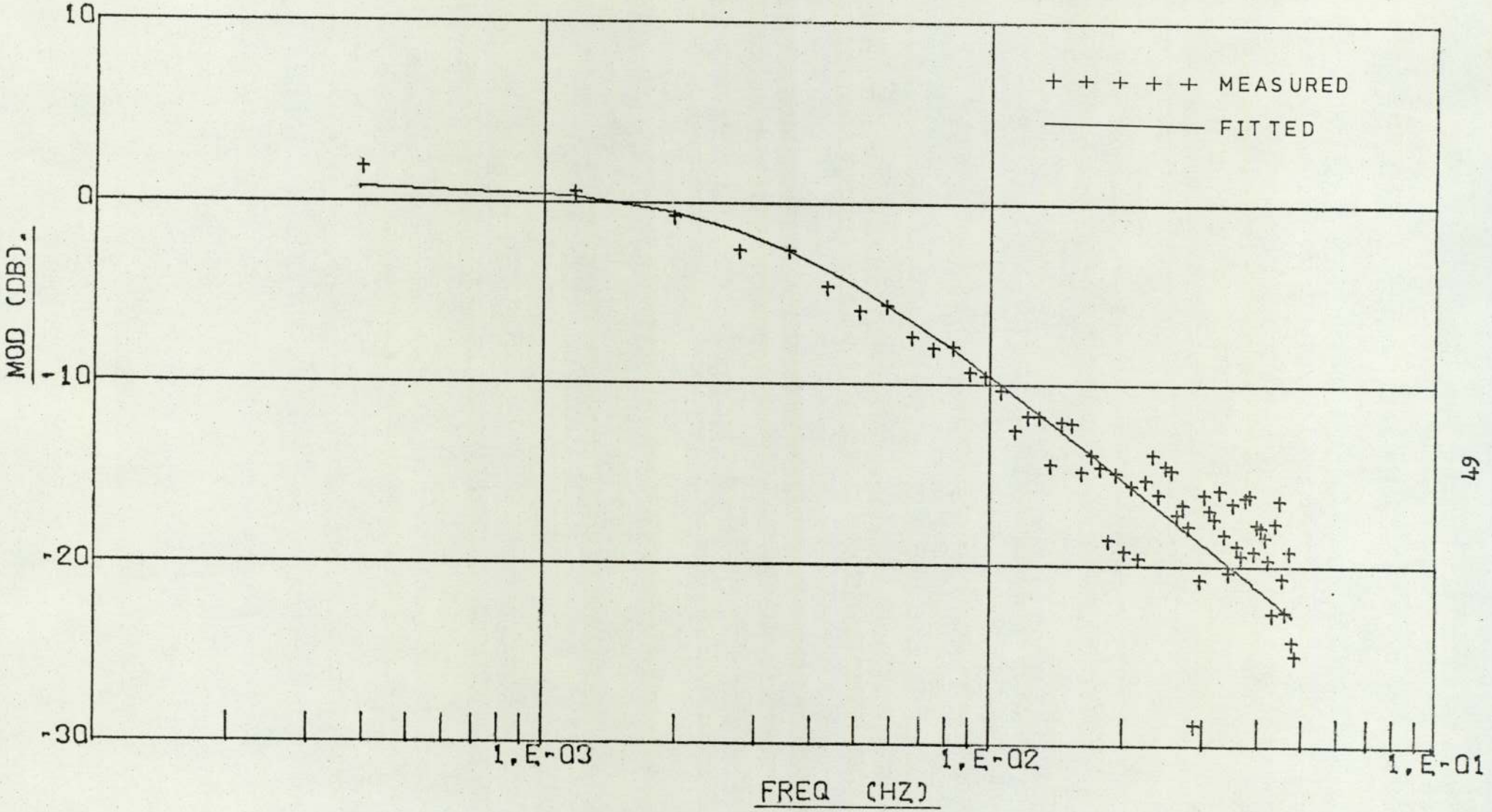


Figure 3.9.2 Fitted Frequency Response of Full Crucible (Phase)

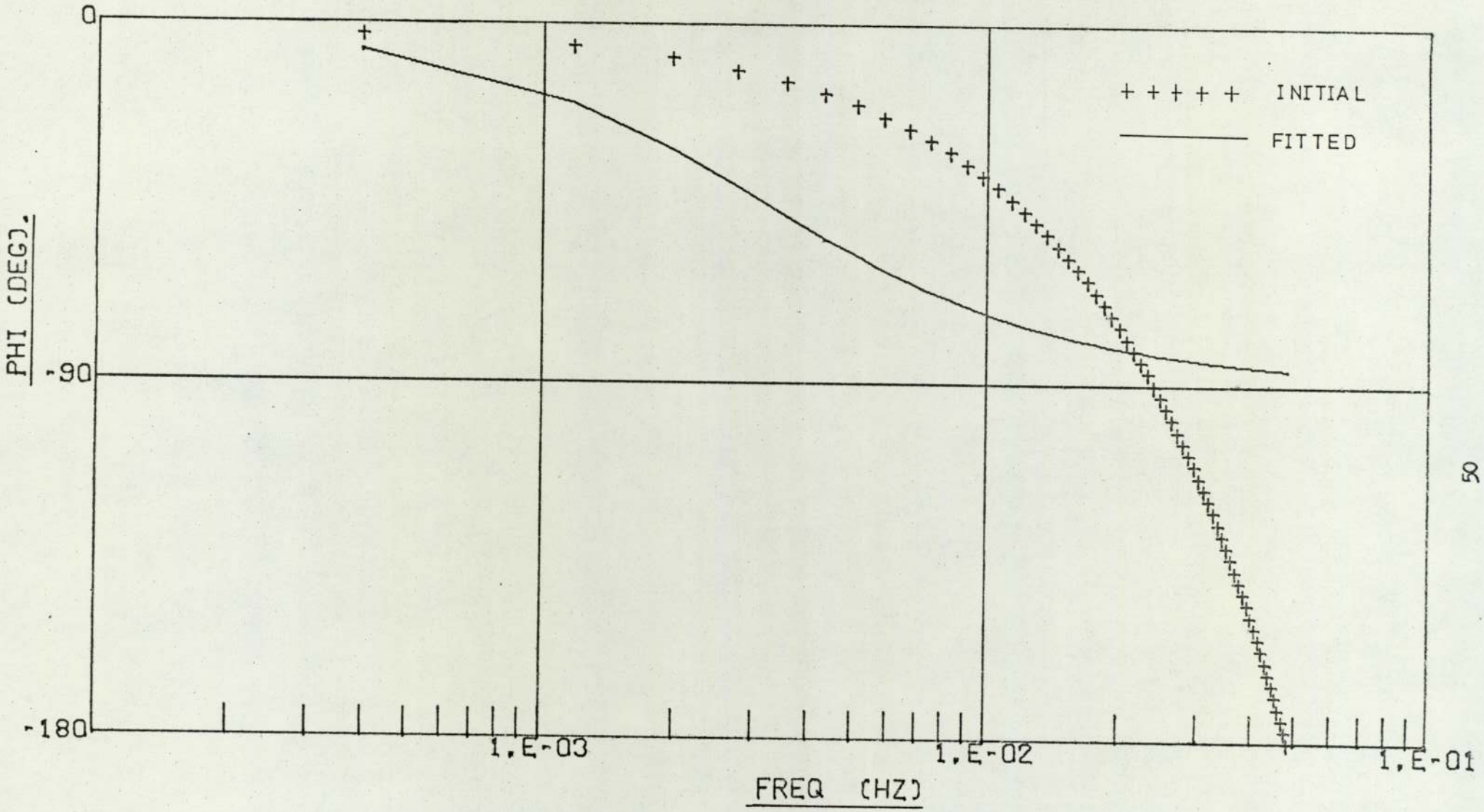


FIGURE 3.10

Crystals Number 3 and 4

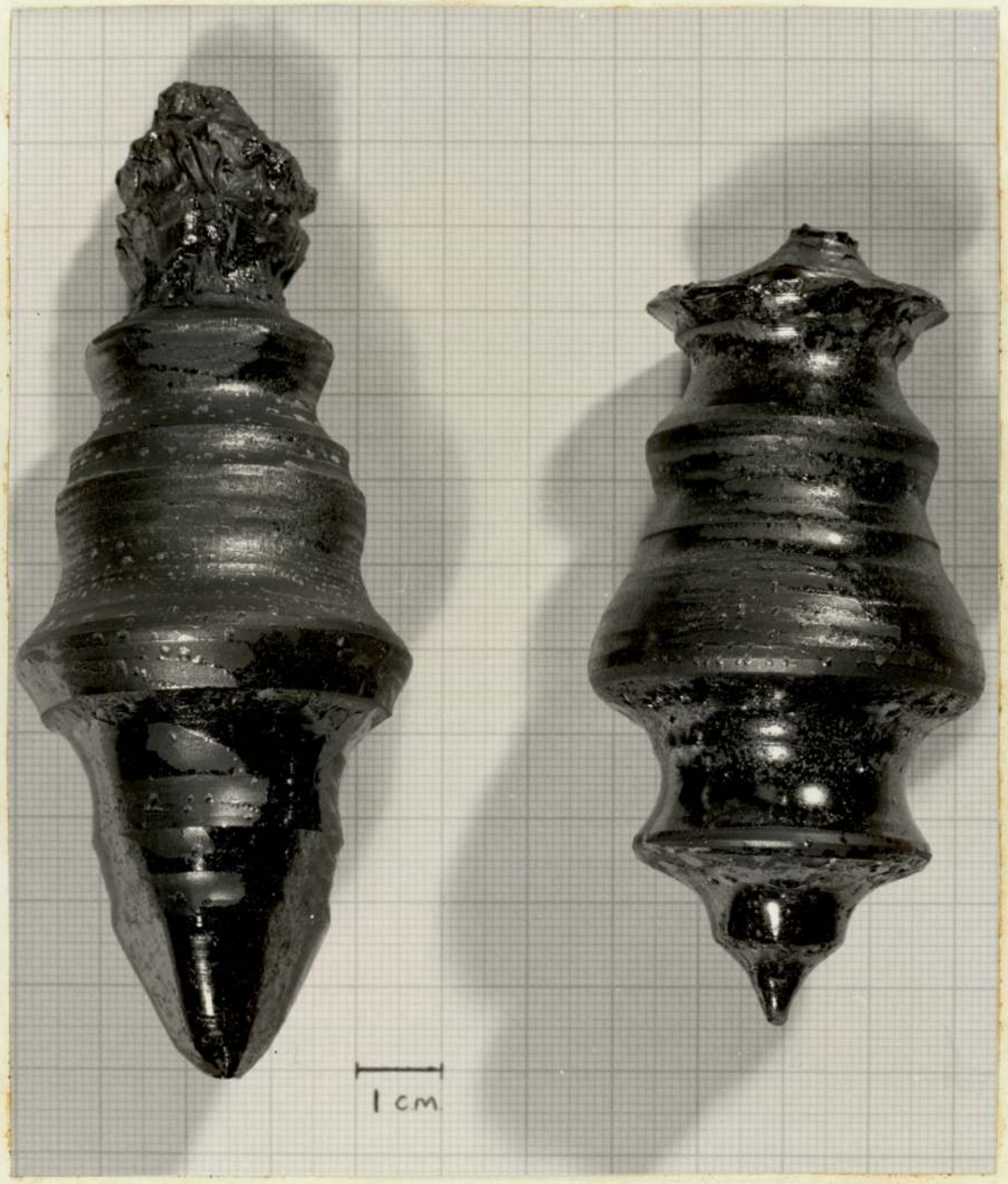
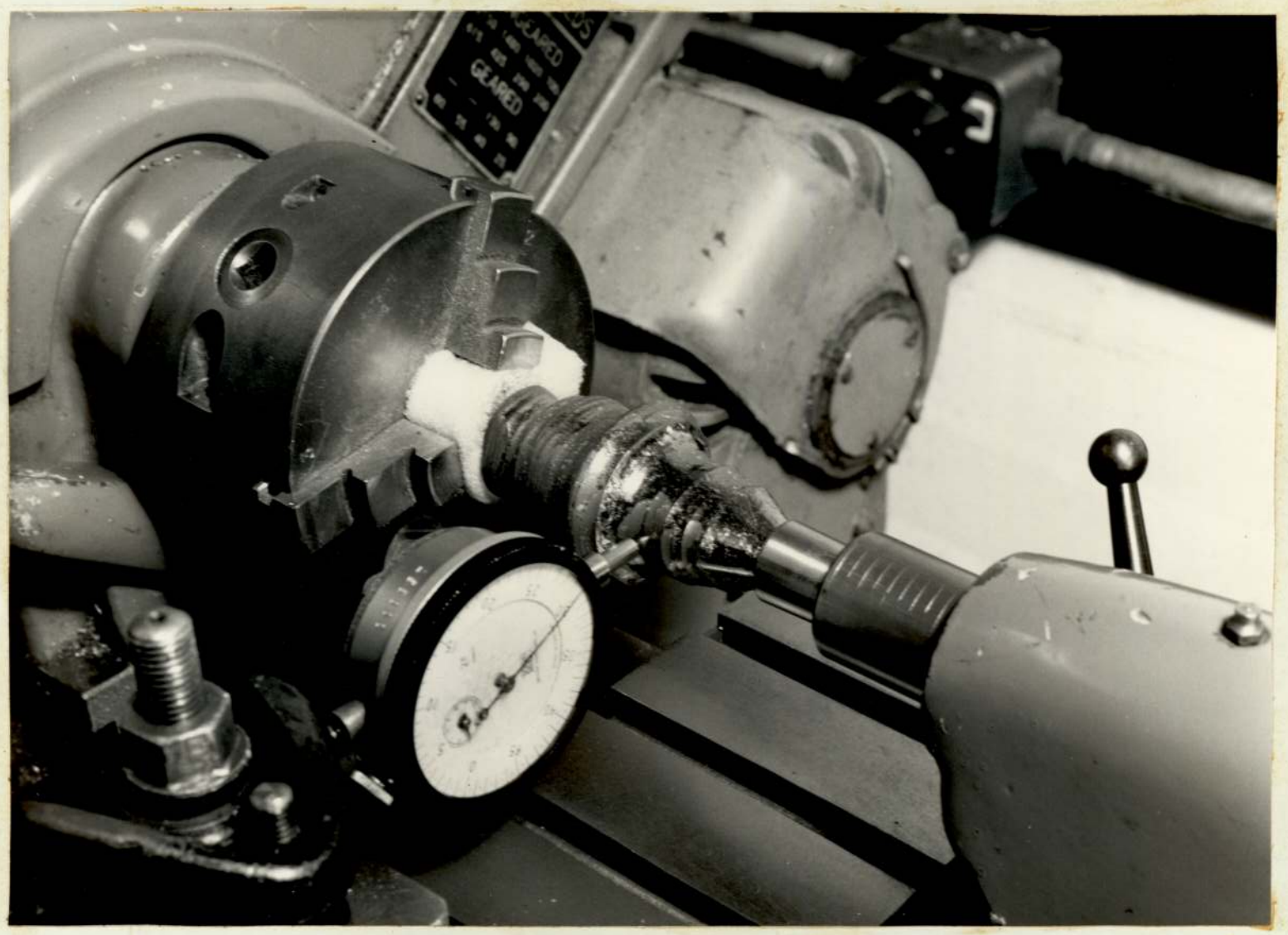


FIGURE 3.11 Crystal Diameter Measurement



4. DATA PROCESSING

4.1 THERMOCOUPLE DATA

4.1.1 General Processing

The input signal used for the first two experiments, (power to temperature runs) was an inverse repeat binary pseudo-random sequence derived from a seventh order characteristic polynomial:

$$F(z) = 1 \oplus z^3 \oplus z^7 \qquad \ldots\ldots\ldots 401$$

where $\oplus$ indicates modulo 2 addition.

There were therefore $2(2^7-1) = 254$ bits in each sequence. It may be shown[52] that even order harmonics are absent so that any frequency response estimates produced from the output data can only exist at odd order harmonics. It is highly desirable to remove any drift that is present in the output sequence[52], but the recommended method, that of data-differencing is wasteful of data so in this case polynomial drift elimination was used. This involved the fitting of a polynomial to the output data by least squares regression and using the perturbations about the fitted curve as the drift corrected data. In this case a first order polynomial fit was used. Bendat and Peirsol[56] show that for a limited amount of data, the spectral resolution varies inversely with the segment length. However the bias and variance of spectral estimates vary inversely with the number of segments. There is therefore a conflict between long segment lengths and large numbers of segments. Carter, Knapp and Nuttall[53] show how both the number of segments and the segment length may be increased by the use of overlapped fast Fourier transform processing. This method together with Hanning windowing[56] was used to produce averaged auto and cross power spectra from which an averaged frequency response estimate and a magnitude squared coherence (MSC) function for the system were calculated (see Chapter 5).

4.1.2 Errors

The errors involved in the estimation of a frequency response function may be categorised into two broad types: 1. Bias errors and 2. Random errors. Bias error is illustrated in Figure 4.1. It arises due to the unavoidable use of a finite width spectral window which can be caused by some or all of the following reasons:[56] 1. Inherent bias in the estimation procedure. 2. Bias due to false linearity assumptions, but note that the frequency response estimator used, $\hat{H}(f) = \frac{\hat{G}_{xy}(f)}{\hat{G}_{xx}(f)}$, gives a least squares optimal estimate of a non linear system. 3. Bias already present in spectral density estimates. 4. Bias due to measurement noise at the input. 5. Bias due to inputs correlated with the test signal. Errors from the first cause are usually neglible compared with errors from the other causes but it can be shown that $E\left[\hat{H}(f)\right] \longrightarrow H(f)$ as either $n = 2B_e T \longrightarrow \infty$ or $\gamma^2_{xy} \longrightarrow 1$ if no other bias errors are present. ($E\left[\hat{H}(f)\right]$ = expected value of $\hat{H}(f)$, n = number of degrees of freedom of each estimate, B_e = estimation bandwidth, T = length of data). Errors from source 2 may be minimised by ensuring the linearity of the process being measured by restricting the amplitude of perturbation. Source 3 may be minimised in a similar manner to 1, by using a narrow enough spectral window to accurately follow any spectral peaks in the power and cross spectra. The fourth source is perhaps rather obvious, in that if an estimate is made on the wrong assumption that the spectral density of the input was higher than it actually was, then the estimate of the system frequency response will be biased downwards by about the same amount. The final source of bias errors is in some sense the opposite of 4. in that it is due to noise present at the input that is not measured. If this noise is completely uncorrelated with the test signal then no bias errors will result. (Random errors will however).

Barker and Davy[52] show that in the absence of noise, system frequency response estimates obtained by the use of pseudo-random signals are deterministic, as opposed to estimates made by using random noise. This feature eliminates the need for large sample sizes and frequency smoothing techniques as described by Bendat and Peirsol[56]. Errors that are still present can be classified in five types:

1. Input errors, where for instance the input PRS is modified by the input transducer. In this case, because the power perturbation is a relatively 'clean' signal and the bit rate of the PRS is far below the cut off frequency of the power controller, this source of error can be neglected.

2. Output errors. These are principally due to aliasing. A general rule is given[52] that errors due to this source will be neglible if $|H(f)|$ of the system frequency response is decreasing at 40dB per decade for $\omega \geq \frac{\pi}{T}$. A preferable method of aliasing reduction would be to oversample, then use a low-pass digital filter and finally re-sample at the required frequency.

3. Noise errors. This includes the effects of disturbances and quantisation error. (In the crystal growing context, disturbances are likely to be the largest source of error). The magnitude error as a result of noise will be greatest at small phase estimates, while the phase error will be greatest at small magnitude errors. The disturbances in a crystal growing system are due to numerous factors, but are virtually impossible to quantify.

4. Drift errors. These will produce bias errors, but are easily avoided by using drift correction techniques on the raw data.

5. Non-linearity errors. As mentioned before, these are minimised by restricting the amplitude of the test signal and further reduced (avoiding even order non-linearities) by using an anti-symmetric test signal.

The noise errors can still be reduced by using more than one period of data for the spectral density estimates and averaging,

or, alternatively if several separate data streams are available, by adopting an ergodic hypothesis and averaging spectral estimates for each of the data streams to increase the number of degrees of freedom of the resultant frequency response estimate. This second method is preferable because it does not increase the estimation bandwidth, B_e and this has been shown to be desirable to reduce bias errors. B_e is a parameter that needs to be chosen with care to provide an optimal trade-off between random errors and bias errors. The overlapping segment method of Carter et al[53] falls into this second category. However, because the segments are overlapped, then the estimates produced from each segment are not entirely independent. A purely heuristic approach indicates that the number of degrees of freedom will be increased by a factor p:

where $p = m(1-q)$ 405

with m = number of overlapped segments

q = overlap fraction

4.2 CRYSTAL DATA

4.2.1 Crystal Time Linearisation

As described in Chapter 3, crystal diameter measurements are available at equal sample spacings along the length of the grown crystal. The only datum is the seed-on point and even this is subject to an uncertainty of $\delta\eta$. Before any frequency domain processing can be contemplated the diameter samples, $D(\eta)$, must be converted into diameter samples at equal time intervals, $D(t)$. Figure 4.2 shows the interface region of the growing crystal.

- L = melt level at which the crystal was seeded and growth commenced
- η = present position of the seed from L
- y = distance of the interface from the seed
- M = present melt level position from L
- r = present crystal radius
- R = crucible radius

By inspection, $y = \eta + M - h$ 406

$$\therefore \quad \frac{dy}{dt} = \frac{d\eta}{dt} + \frac{dM}{dt} - \frac{dh}{dt} \qquad \ldots\ldots 407$$

now $\frac{d\eta}{dt}$ is the pull rate, υ , which is constant during the growth cycle

$\frac{dM}{dt}$ is the melt level drop rate, $= q\left(\frac{dy}{dt} + \frac{dh}{dt}\right)$ 408

with $q = \frac{r^2}{R^2}\frac{\rho_s}{\rho_m}$ 409

where ρs = density of solid

ρm = density of melt

Now it can be shown from the meniscus geometry (Chapter 2), that

$$\frac{dr}{dt} = \frac{dy}{dt} K(H-h) \qquad \ldots\ldots 410$$

where K = meniscus curvature constant

H = Height to vertice of meniscus curve

hence $\frac{d}{dt}\left(\frac{dr}{dy}\right) = \frac{-Kdh}{dt}$ 411

substituting for $\frac{dh}{dt}$ and $\frac{dM}{dt}$ into equation 407 and re-arranging for $\frac{d\eta}{dt}$ yields:

$$\frac{d\eta}{dt} = (1-q)\left[\frac{dy}{dt} - \frac{1}{K}\frac{d}{dt}\left(\frac{dr}{dy}\right)\right] \qquad \ldots\ldots 412$$

In a short time interval Δt, the distance through which the pull rod moves is $\upsilon \Delta t$ and from Figure 4.2:

$$\Delta\eta = (1-q)\left[\Delta y - \frac{1}{K}\Delta_s\right] \qquad \ldots\ldots 413$$

where Δ_s is the change in surface slope, $\frac{dr}{dy}$ over the interval. This comprises an algorithm for computing the incremental slice thickness.

The summation $Z = \sum_{j=1}^{N} \Delta\eta_j$ gives a quantity which is directly proportional to the crystal growth time. The next part of the algorithm is required to divide Z into m equal divisions where m is the required sample time interval. This process is illustrated in Figure 4.3. Observe that $\frac{Z}{m}$ does not in general yield a sample interval that is in synchronism with the sample interval of $\sum\Delta\eta$. Linear interpolation is performed between the nearest two points of this function to give the sample number at which the diameter is required. Again, in general this sample number will be non-integer. A further linear interpolation is performed on the

measured crystal diameter to yield the required diameter at the linear time sample point. The flowchart and graphic illustrations of the time-linearisation algorithm are shown in Figure 5.6.

Some special points about equation 413 need emphasising. The term Δ_s is in effect a second derivative and as such is liable to be subject to high noise levels unless special precautions are taken. If $\frac{1}{K} \Delta_s > \Delta y$, then a negative slice thickness is computed, implying that crystal melt back has occurred. The effect of this on the time linearisation algorithm would be to produce a discontinuity in the modified diameter profile. This is unavoidable since if melt back had occurred, information would have been lost and it would be hardly surprising that the algorithm could not replace it. Maximising Δy minimises the danger of a computed melt back for any particular slice but Δy cannot be increased indefinitely as resolution would be lost. An optimum value of Δy must therefore exist, large enough to preclude melt back but small enough to maintain the bandwidth of the resultant diameter profile.

The crystal diameter measurements that were available had been made at as close a spacing as possible and were spaced at ·0051cm along each of the useable crystals. This was far higher than the pseudo-random ternary sequence bit rate demanded and hence gave the option of low pass filtering and then re-sampling before time linearisation. This is a highly desirable operation to carry out in any case to reduce aliasing in subsequent spectral estimates.

The second derivative term, Δ_s needs special attention, as indicated, to minimise the noise that will be amplified in the derivative operation. Accordingly a Lanczos 'low noise' digital differentiator was used. This is described by Kuo and Kaiser[57]. Strictly, an approximation is involved in assuming that the change of $\frac{dr}{dy}$ over a time interval Δt is given by the second derivative of r with respect to y. However this is only a second order error, it could be iteratively eliminated by computing a better

estimate of Δ_s from the time linearised profile that was produced using equation 413. However the error was considered small enough to be neglible. A seventh order Lanczos filter was chosen, having a differentiator cuttoff frequency of $\cdot 15 f_c$. The crystal perturbations in the frequency range of interest were therefore differentiated while the higher, (noise) frequencies were attenuated. Second derivative action was achieved by passing the crystal data through the differentiator twice. Since the Lanczos low noise differentiator is a non-recursive filter and it is not being used on 'on line' data then it is possible to centre its impulse response about zero time. The filter will not then produce any phase shift between the input and output sequences. This is the same principle as was used for the more conventional low-pass digital filters that were designed within the DPPP software package. (See Chapter 5). It was found that in order to avoid negative slice thicknesses being computed from equation 413, the slice thickness, Δy needed to be approximately three times as great as the basic, measured crystal slice. In summary the time linearisation strategy that has evolved is as follows:

1. Low pass filter the raw crystal profile
2. Re-sample and perform time linearisation
3. Low pass filter and re-sample to give crystal samples at bit rate of PRS.

while using a lowpass second derivative of the crystal profile in the time linearisation algorithm.

4.2.2 Time Synchronisation

Once the linear time profile has been produced, the problem remains of synchronising it with the injected PRS. Re-arranging equation 406:

$$\eta = y - M + h \qquad \ldots\ldots\ldots 414$$

The time from seed-on to the start and finish of the PRS is accurately known.

Let M_c = grown crystal mass and ρ_m = melt density

then the volume of liquid GaP removed from the crucible during growth, V_L is given by:

$$V_L = \frac{M_c}{\rho_m} \qquad \ldots\ldots\ldots 415$$

It has been shown in Appendix 2 that the depth of liquid in a round bottomed crucible is given by:

$$h_c^3 - 3Rh_c^2 + \frac{3V_L}{\pi} = 0 \qquad \ldots\ldots\ldots 416$$

Also, from the geometry of the crucible, the maximum depth possible in the curved part is given by:

$$\hat{h}_c = (2 - \sqrt{3})R$$

giving a maximum volume in the curved part of:

$$\hat{V}_L = \frac{\pi}{3} R^3(3\sqrt{3}-5) \qquad \ldots\ldots\ldots 417$$

Hence the depth of melt in the crucible may be calculated in two stages, that in the curved portion, and the remainder occupying the cylindrical part. Similarly, M, the drop in melt level that occurred as a result of growing the crystal may be calculated. If the interface height, h is assumed neglible compared with the length of crystal, then equation 414 may now be used to calculate the pull bar movement, η. The time taken to grow the measured length of crystal is now established since the pull bar lift rate is controlled at a constant value. The time axis of the diameter/time curve may now be calibrated.

4.2.3 Crystal Results

The measured crystal profiles for crystal nos 3 and 4 are shown in Figures 4.4 and 4.5 respectively. The intermediate stages involved in the time linearisation process are shown in Figures 4.6 and 4.7 respectively for crystals nos 3 and 4. The relevant parameters in the data processing operations are shown in Table 4.1. The final decimation factors were set at integers which allowed the initial decimation factors to be as near to 3 as possible for reasons discussed above. (The computer programs were arranged to allow

non integer decimation within the time linearisation program but only integer decimation within the digital filter program). As can be seen from the table, the length of data available for crystal No 3 was insufficient to accommodate a complete sequence of the PRS. It was reluctantly concluded that there was little point in proceeding any further with the data processing for this crystal. The final 80 data points for crystal No 4 were drift corrected by a second order polynomial curve fit and a transfer function, H(s) was calculated using the 'SPECTA' package as described in Chapter 5. (A Hanning window was used and zero order hold correction was applied to the resultant frequency response). The frequency response is shown in Figures 4.8.1 and 4.8.2. The fitted curves shown in these graphs are the result of 'FFIT' (see Chapter 5), fits to modulus only. The transfer function was found to be:

$$H(s) = \frac{\cdot 7707}{1 + 579 \cdot 03s - 26201s^2} \qquad \ldots\ldots\ldots 418$$

with a sum of squares of error of fit = ·00851

Note that H(s) has a right hand half s plane pole. This was considered to be unrealistic since the crystal did grow in a quite stable manner. It was concluded that the apparent unstable transfer function that was fitted by the 'FFIT' algorithm was purely as a result of excessive variance on the transfer function, particularly at high frequencies. A possible contribution to this variance was made by the fact that the data used for the Fourier transform was not settled. (The recommended half period settling time before data was collected could not be met due to lack of data). Although this factor threw doubt on the validity of the data, an attempt was made to fit a more meaningful transfer function by truncating the points on the modulus curve above the seventh. (At this point the variance becomes visibly large). The resultant frequency responses together with fitted transfer functions,

$(H'(s))$ are shown in Figures 4.9.1 and 4.9.2.

with $H'(s) = \dfrac{\cdot 7455}{1+641\cdot 53s+78{,}821s^2}$ 419

with a sum of squares of error of fit = ·00244.

The value of .7455 for the gain has units of inches diameter per 1.375% mean power. When normalised to cm/cm radius/per unit power change, a value for K_o of 45.91 is obtained. By comparison with the standard form of second order transfer function, $F(s) = \dfrac{\omega_n^2}{\omega_n^2 + 2\zeta\omega_n s + s^2}$ values for ζ and ω_n of 1.14 and 0.214 rad/min respectively are obtained. By comparing the measured phase lag at .000972 Hz with the phase lag given from the fitted transfer function at the same frequency, a phase difference of 146.4° remains. If this attributable to dead time then this term is e^{-418s}. At the growth rate used, this represents 0.26 cm along the crystal.

4.2.4 Crystal Conclusions.

This magnitude of dead time is feasible since it may have contributions from three sources:

1) Uncertainty on the exact seed-on position on the cold crystal.
2) Dead time between power changes and temperature changes in the melt. (The power to temperature experiments have indicated a dead time of up to 90 seconds, but the temperature measured was not in the melt itself).
3) A possible delay in the temperature to radius dynamics.

Most of the dead time is likely to be the result of source 1), as the seed-on position could only be identified to within 0.2cm. The measured natural frequency of .21 rad/min is 24% above the theoretically predicted value, well within the error bounds of the prediction. The damping factor is 148% above prediction and the gain constant is 28% above. The theoretical transfer function has been derived on the assumption of small deviations about a

mean radius. However the crystal that was grown exhibited comparatively large deviations about a mean. The two sets of parameters for GaP shown in Table T2.1 indicate the likely change of parameters with changes of radius. These changes are comparatively small but the violation of the small signal assumption is more serious since other parameters, assumed constant, will vary. For example, the gradient variation factor, α, that relates the crystal temperature gradient to the radius implies that the gradient increases with radius. This will not be the case because the crystal will be heated more by the walls of the crucible as it grows larger. This will tend to reduce the gradient. Similarly, the power radiated from the interface will be influenced by the aspect of the interface to the walls of the crucible. The radiation from the interface will also be influenced by the depth of B_2O_3 at any instant. Larger radii of crystal will cause a deeper layer of encapsulant, further tending to reduce the power radiated and also reducing the crystal temperature gradient. The effect, apart from that reported by Nygren[43] where the crystal shielding causes a rapid reduction in radius, will be to decrease the damping factor. Equation 233 indicates that the gain, K_o will be increased under these circumstances in that G_o and $\theta_m - \theta_i$ will be decreased. This is possibly an explanation of the observed difficulty of achieving uniform radius crystal growth in encapsulated systems under manual control. With the increased gain and reduced damping it is very easy to over-correct a large crystal.

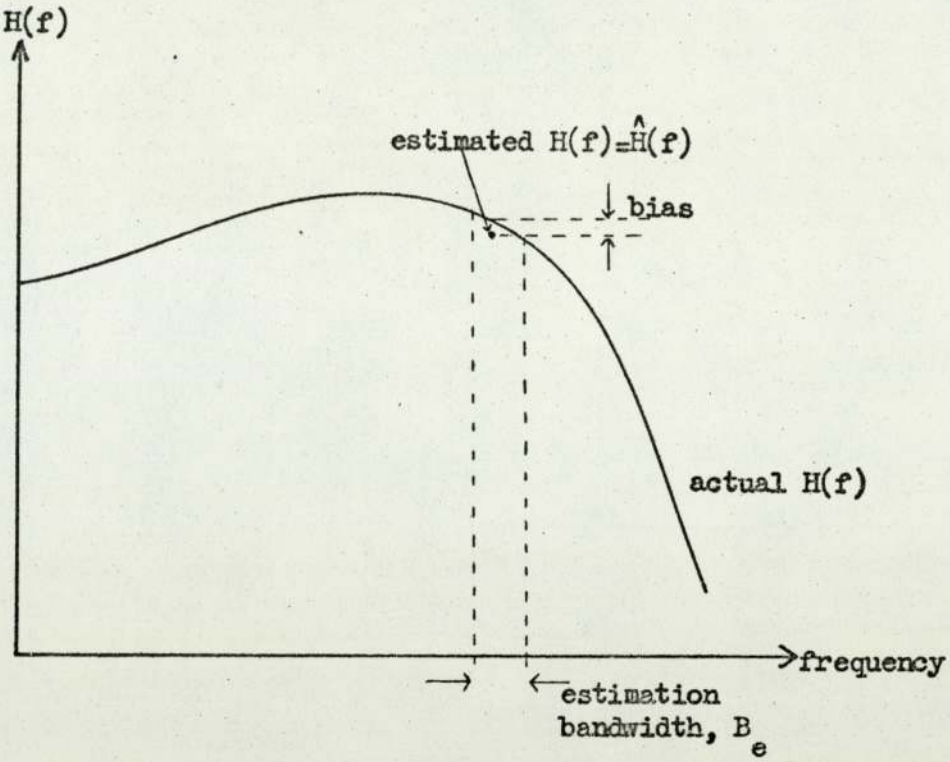
To summarise, the transfer function that has been identified from only one set of results is well within the error bounds of the predicted transfer function. However the prediction is necessarily widely defined due to lack of parameters. The accuracy of the derived parameters has been improved by using the measured transfer

function (Section 2.4). The qualitative effects of deviations from the assumptions made in the analysis have been examined.

TABLE T4.1 Crystal Processing Parameters

Parameter	Crystal 3	Crystal 4
Number of samples at •0051cm	1290	1025
Pull bar speed (cm/hr)	2•3919	1•857
Mass of crystal (gm)	245	228
Drop in melt level (cm)	1•685	1•6557
Time represented on linear time graph (min) from	25•8	62•40
to	136•53	182.52
Number of samples per PRS bit	17•2	12•8
Pre filter cuttoff $\left(\frac{fs}{fc}\right)$	•145	•156
Initial decimation factor (in linear time)	3•44	3•20
Final filter cuttoff $\left(\frac{fs}{fc}\right)$	•1	•125
Final decimation factor	5	4
Time linearisation 'K'(mm^{-1})	4.3	4.3

Figure 4.1 Illustration of Bias Errors



The estimate is averaged over B_e

Figure 4.2 Interface Region

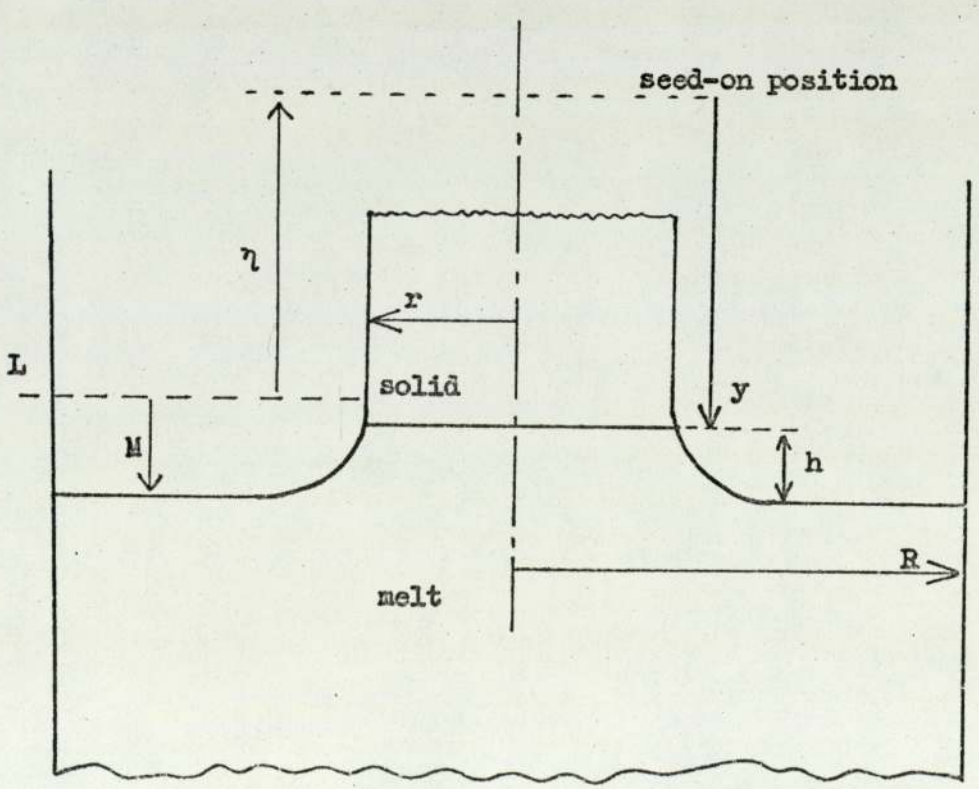
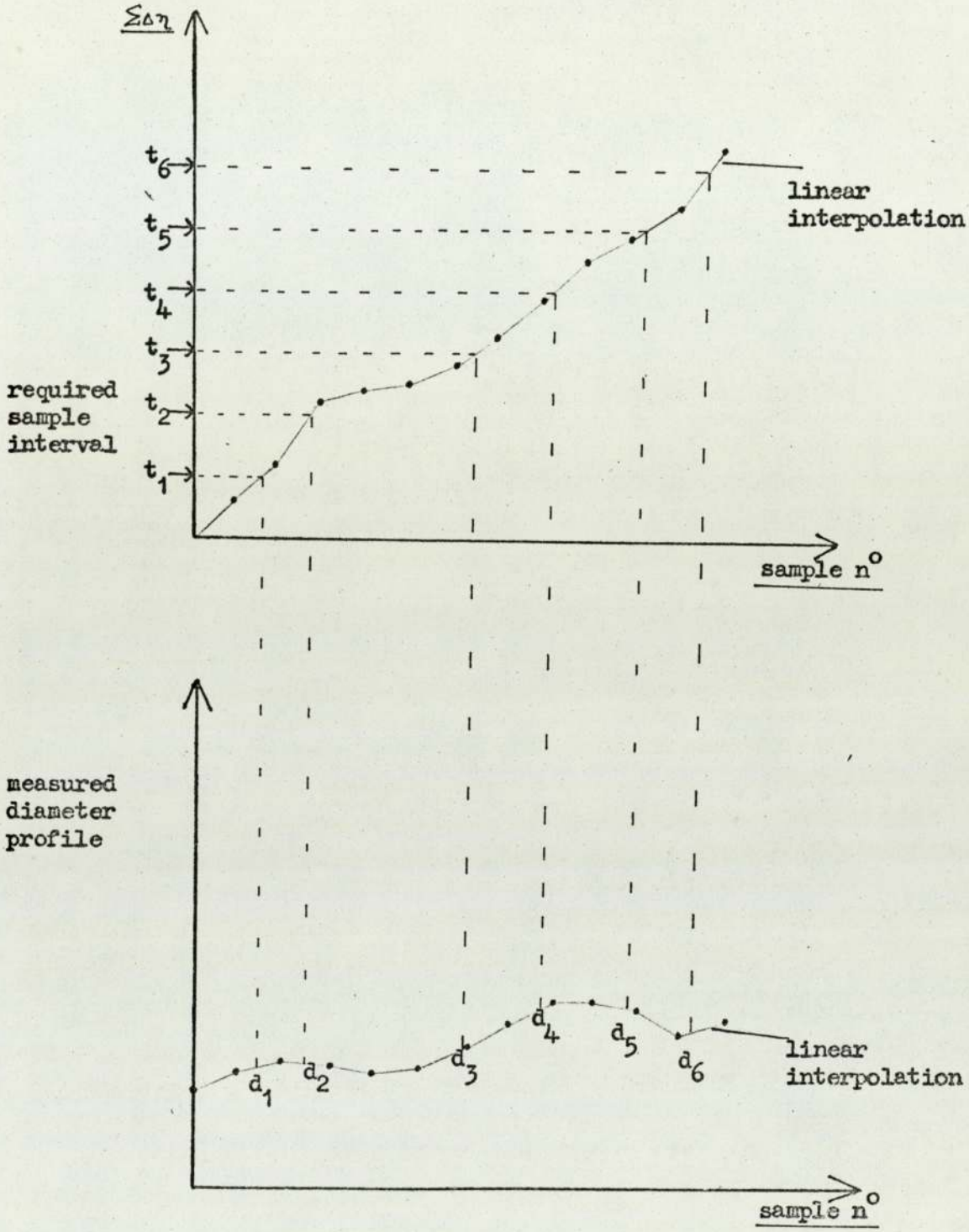


Figure 4.3 Time Linearisation Interpolation



The time linearised profile is comprised of the sequence

d_1 d_2 d_3 d_4 d_5 d_6 ..... at time intervals

t_1 t_2 t_3 t_4 t_5 t_6 .....

Figure 4.4 Diameter Profile of Crystal N° 3

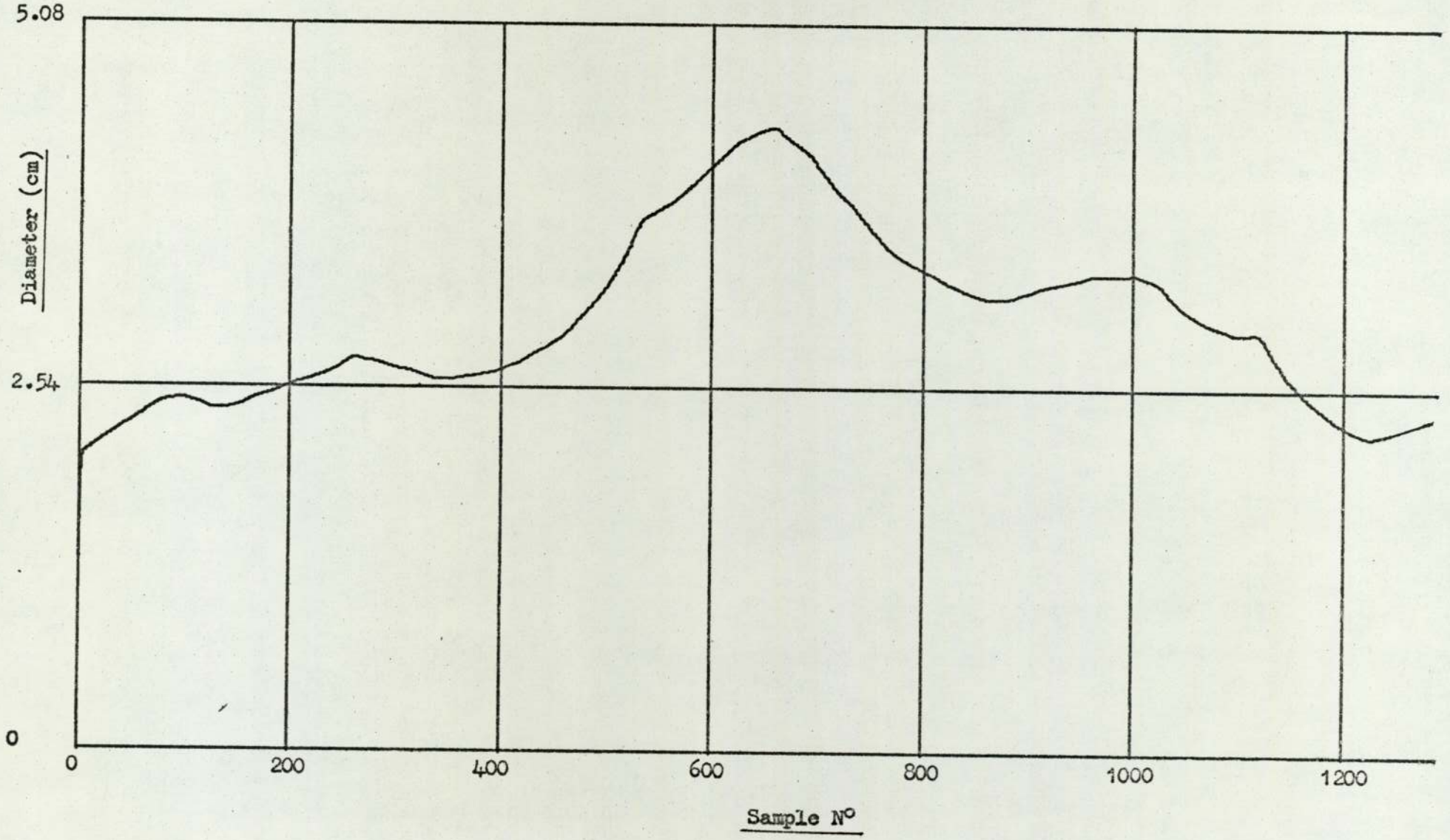


Figure 4.5 Profile of Crystal N° 4

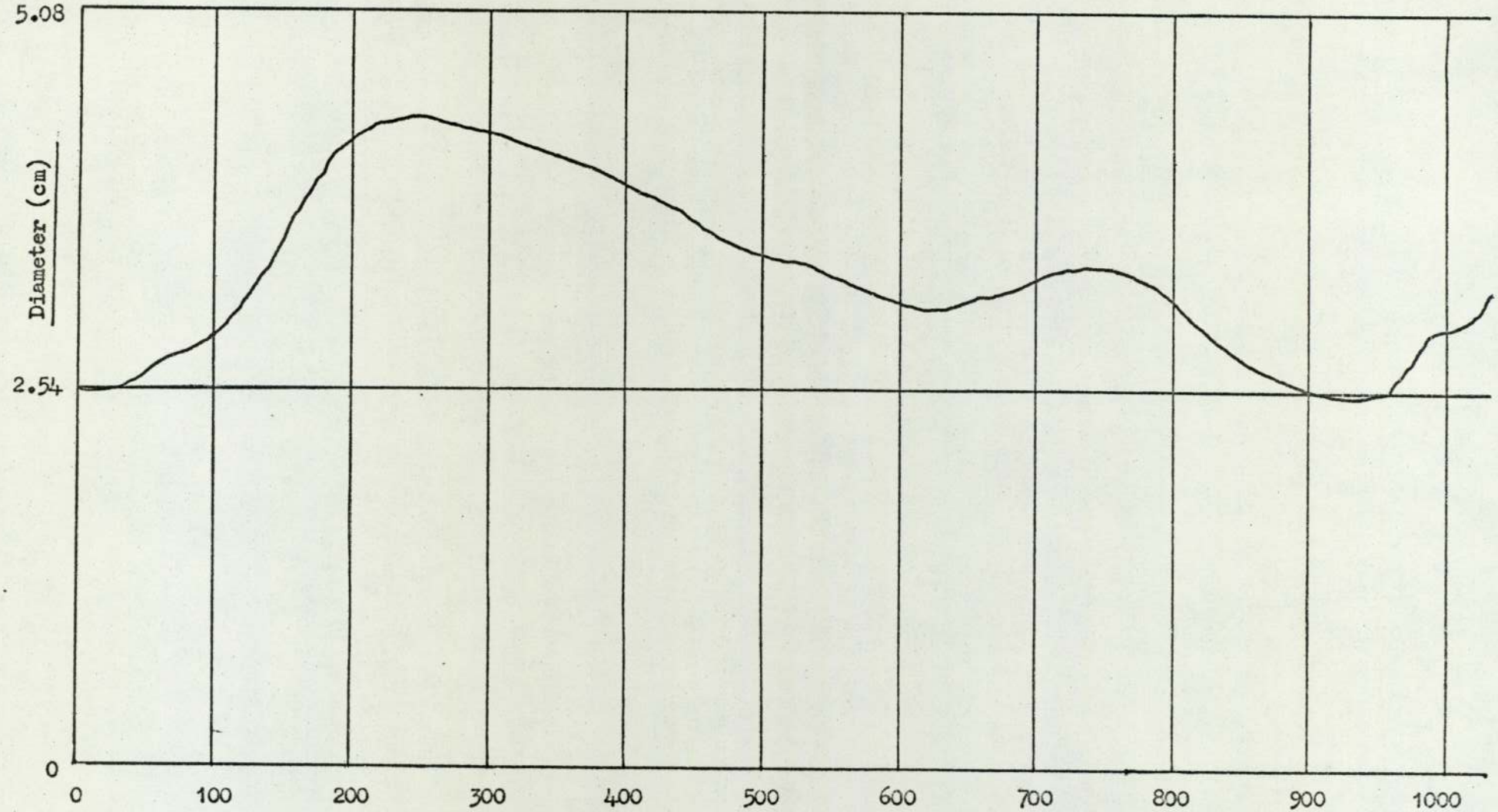


Figure 4.6 Time linearisation of Crystal N° 3

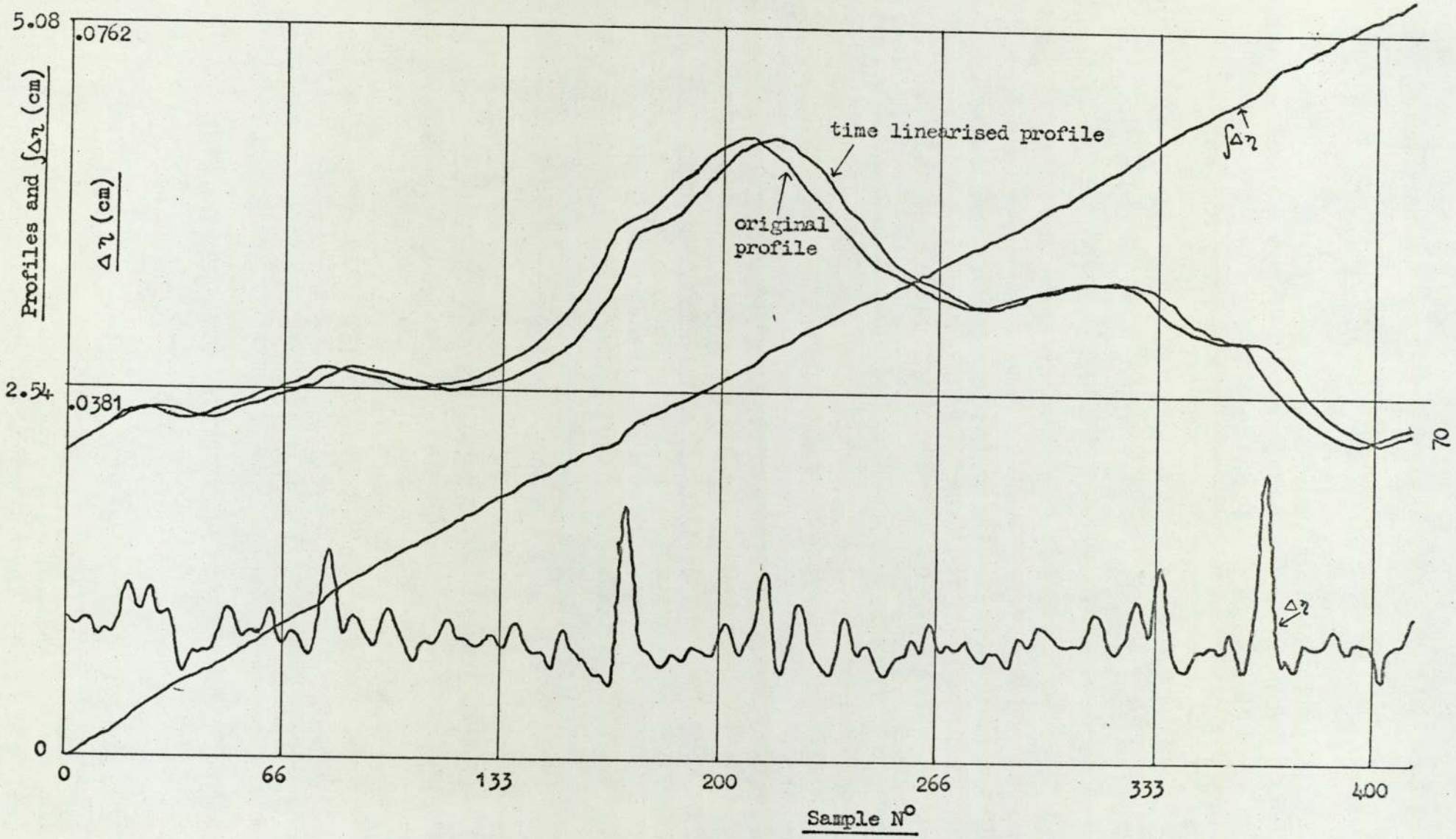


Figure 4.7 Time Linearisation of Crystal N° 4

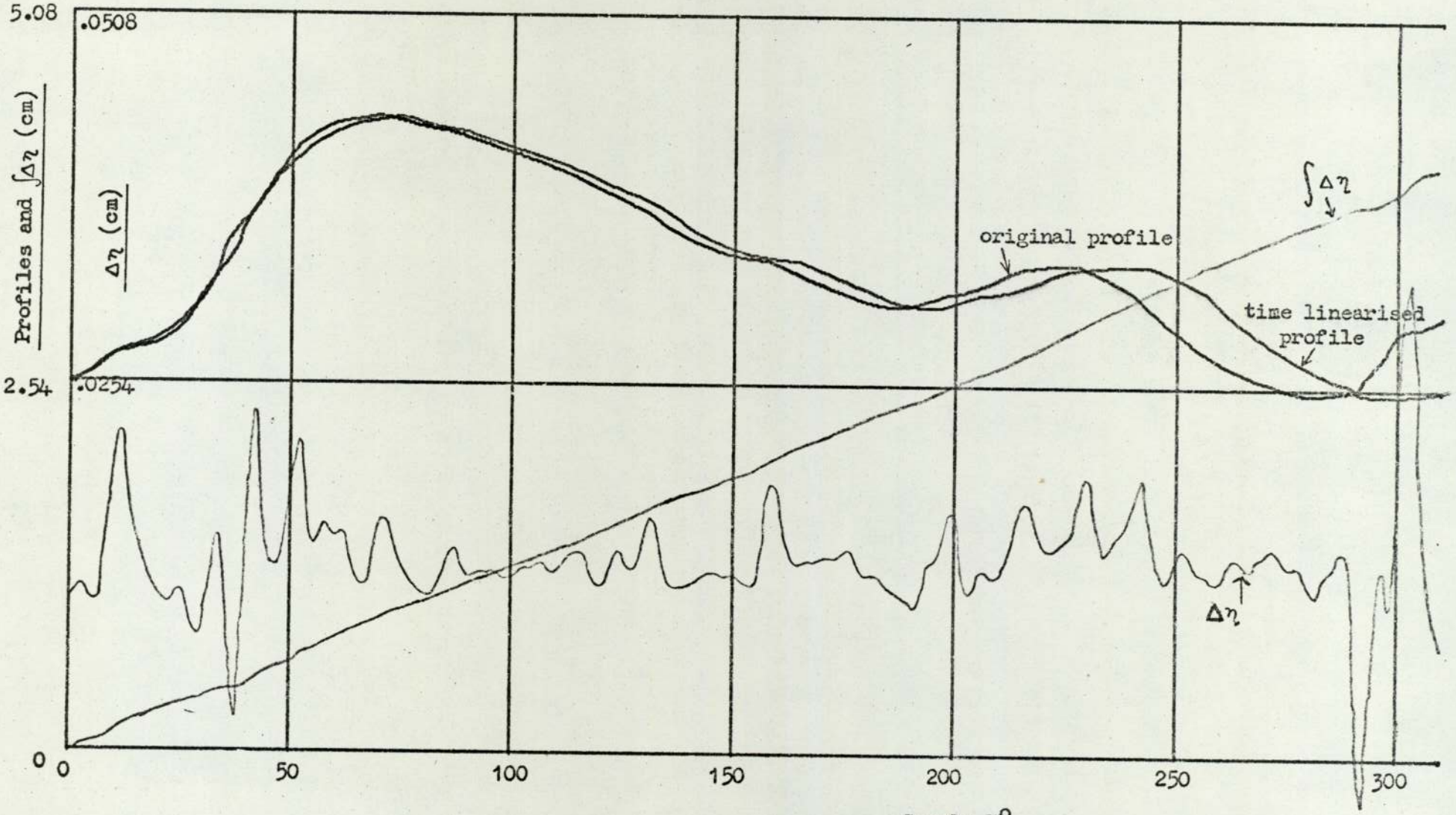


Figure 4.8.1 Power to Diameter Frequency Response For Crystal N° 4 (Modulus)

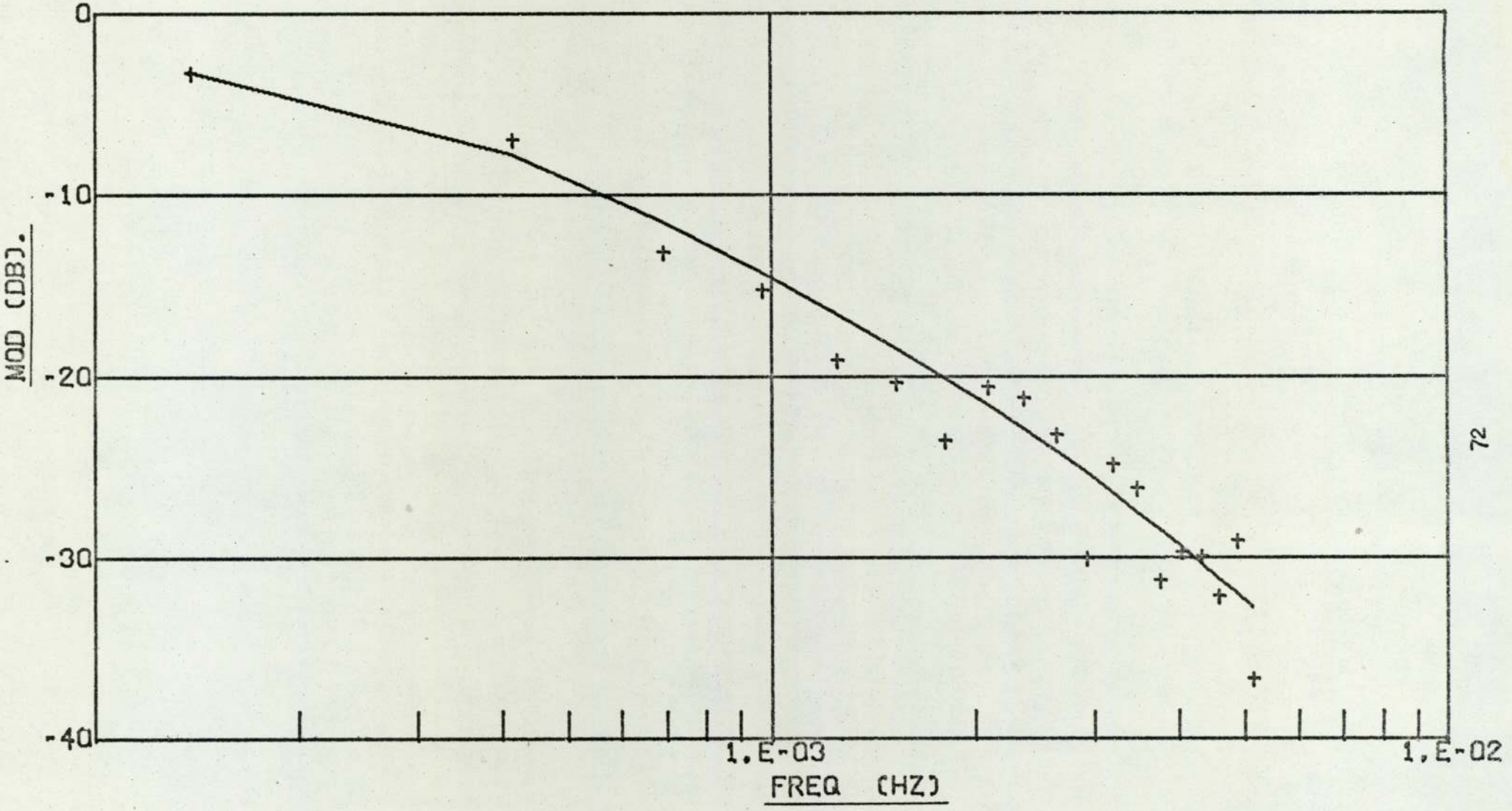


Figure 4.8.2 Power to Diameter Frequency Response For Crystal N° 4 (Phase)

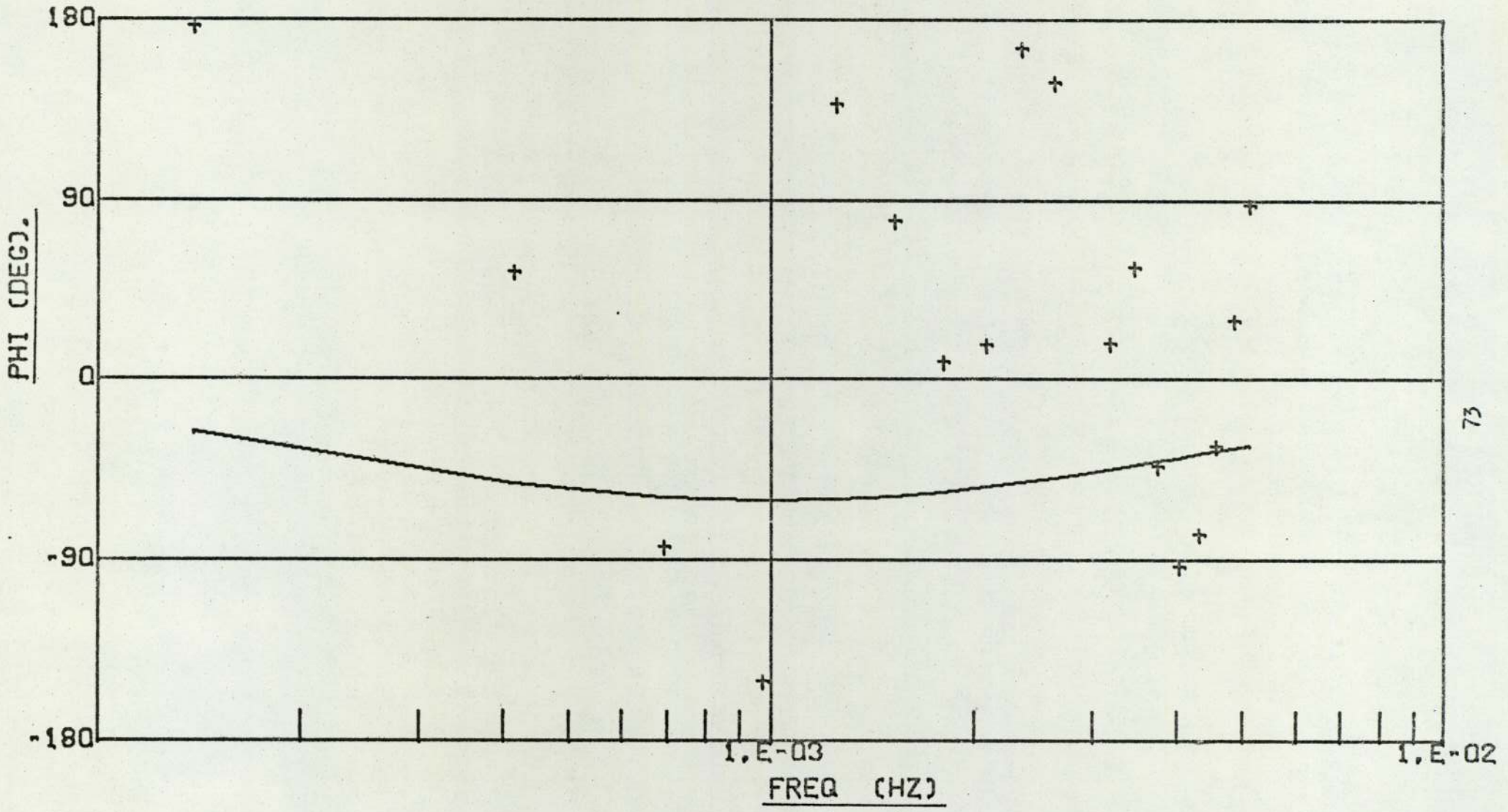


Figure 4.9.1 Modified Power to Diameter Transfer Function (Modulus)

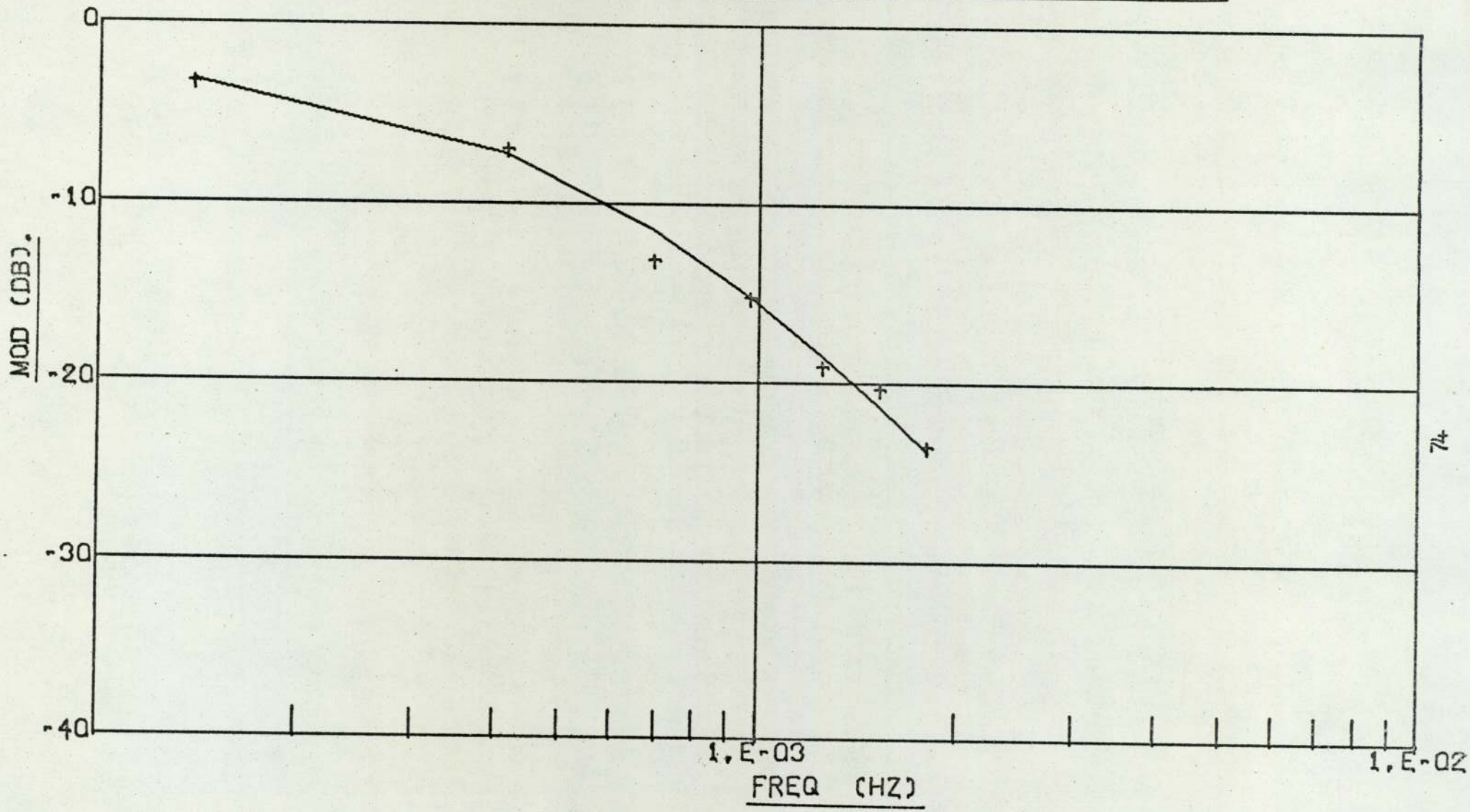
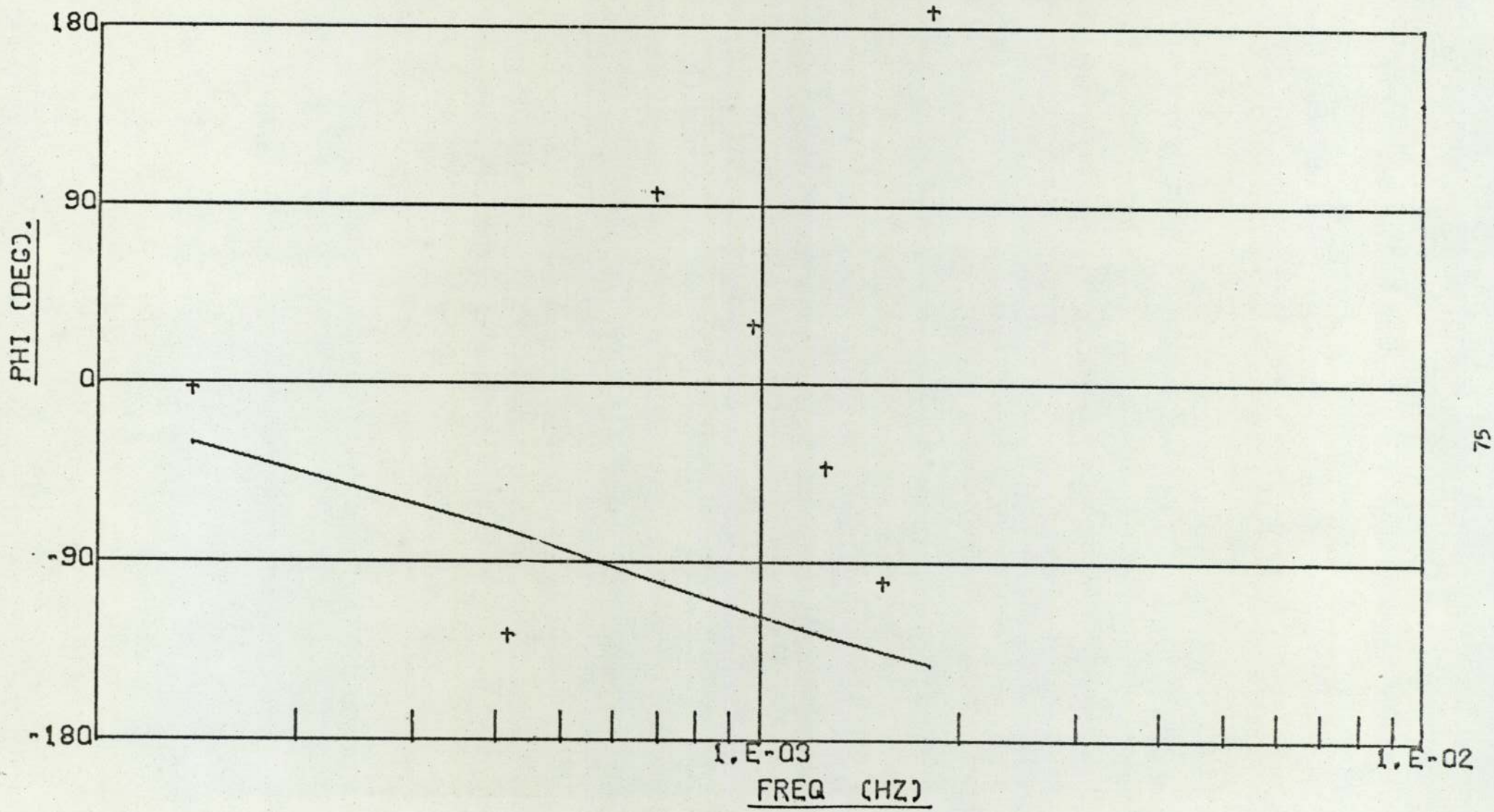


Figure 4.9.2 Modified Power to Diameter Transfer Function (Phase)



5. COMPUTER PROGRAM PACKAGES

5.1 GENERAL DESCRIPTION

All the programs used in this project were assembled into 3 distinct packages. They were designed to run on a PDP9 with 24k of store and the following peripherals

Teletype
Incremental Graph Plotter
3 Dec-Tape units
Paper tape punch and reader
D-A convertor (2 channels)

The 3 packages perform

a. General data manipulation
b. Spectral analysis of time series
c. Frequency response calculation and transfer function fitting

The 3 packages were designed to interface data into each other by means of paper tape or magnetic tape. For example, a frequency response function produced by b. could be output to a file on magnetic tape which could then be accessed by c.

5.2 DPPP - GENERAL DATA MANIPULATION PACKAGE

This consists of a short main program to read in data and enable a choice of subroutines to be made. The flowchart for DPPP is shown in Fig 5.1. The data may be input from any of the PDP9 input devices listed in section 5.1. Data is read into an array which may be rotated to enable, for instance, a mid-section of the data stream to be processed. With the core available, storage for 1300 single precision data points is possible. A further array of 200 is available for storage of digital filter coefficients or polynomial coefficients. Program control is effected by means of conversation with the teletype and by means of the accumulator switches. The sub-routines are selected via the teletype and after their execution a further subroutine may be selected.

The subroutines available are

DRITF - adds polynomials to data stream
FILTUR - non recursive, zero phase digital filter
FILD - non recursive, low pass digital filter design program
POLLY - fits curves to data by least squares regression
ENTRER - enables entry of coefficients
DIMOD3 - crystal time linearisation program
PUTOUT - general data output program
TRANS - transforms low pass digital filters to highpass or bandpass.

5.2.1 DRITF

The flowchart for this subroutine is shown in Fig 5.2. Note that if this subroutine is selected immediately after loading DPPP, no polynomial coefficients will exist. To avoid this, DRITF checks for the number of coefficients, N, to be non zero. If this test is not satisfied, an error message is output on the teletype and control returns to the main DPPP program. The drift polynomial may be added or subtracted from the data stream, the polarity being selected by means of the teletype.

If the data stream is a sequence $\{X\}$, of m elements and the drift polynomial is represented by a sequence $\{Y\}$, of n elements in descending order, then the drift correction is effected by the algorithm

$$X_j = X_j \pm \sum_{i=1}^{n} Y_i j^{n-i} \qquad \text{for} \quad j=1,m \qquad \ldots. 501$$

5.2.2 FILTUR

The structure of this program is shown in Fig 5.3. The coefficients are stored in the same array as was used for the polynomial coefficients in DRITF. They must either be entered by hand or calculated with program FILD. The structure of FILTUR is a non-recursive digital filter. The weighting function is convolved with the data stream. A special feature of this particular implementation is that the filtered sequence has zero phase shift at all frequencies. In a

physical system this would be unrealiseable since it requires the existence of a unit advance operator (z). In this off line case however, the entire data sequence is available simultaneously so that at a given point on the sequence, data can be both predicted and recalled. The convolution of the filter weighting function with the data stream is illustrated in Fig 5.4. Note that at each end of the data stream the weighting function overlaps into zeros, so that the filter has a settling time of half the width of the weighting function at each end of the data stream. In order to minimise the storage necessary, the program uses an auxiliary storage array of the same length as the filter coefficient array. This temporary storage is filled up from the original input data and convolution takes place between the temporary storage and the filter coefficients. The original data is replaced point by point as the temporary data frame is scanned along the input data. This method effectively avoids having storage for an extra 1300 points. When the convolution is completed, the filtered data sequence may be re-sampled at every Lth point. L is specified on the teletype and the new length of the data sequence is output on the teletype.

5.2.3 FILD

This program designs a digital filter approximation to an ideal low pass frequency response. The design is by the 'window' method, as described by Ackroyd[62]. For convenience, a brief description of the method is given here. An ideal lowpass frequency response is set up with a cut-off frequency f_c, assuming the approximation to be composed of n points and to have a clock frequency of f. These 3 parameters are specified during conversation. This frequency response is then transformed to the time domain using the fast fourier transform algorithm by Singleton[59]. This transformed frequency response is now the 'ideal' impulse response of the filter. However, this impulse response is necessarily truncated since it follows from

the Fourier transformation that the transform of a finite function will be an infinite function. The effects of this truncation are that the stop band attenuation of the filter will be limited and the frequncy response will have a ripple superimposed on it. Both these effects may be ameliorated by multiplying the truncated impulse response by a window function. By this means it is possible to improve the stop band attenuation at the expense of roll-off rate. For a given window the stop band attenuation is improved by increasing the width of the window (up to a limit of n). Of the 3 windows available within this program, the Blackman provides the greatest stop-band attenuation and the rectangular the converse. The Hamming window falls mid way. The window type and length are specified during conversation. The resultant windowed impulse response is the final set of coefficients for the digital filter. They are stored in array COEFF for use by other programs. The coefficients are also copied into a temporary storage array which is then transformed back to the frequency domain. The gain and phase response of the actual filter are output on the graph plotter,so that a comparison can be made between the ideal required and the actual realised frequency response. Note that a non-recursive digital filter will always have a phase characteristic that is linear with frequency since it is composed of a linear sum of weighted signal components that have been passed through unit delay (or advance) operators. It is therefore possible to shift the impulse response of the filter so that it is centred about zero time and hence produce a phase characteristic that is zero at all frequencies. The flowchart for FILD is shown in Fig 5.5

5.2.4 POLLY

This program fits polynomials to data by least squares regression. The maximum order of polynomial is ten. This is specified via the teletype. If n data points are to be fitted, the independent variable, X,ranges from 1 to n. The fitted coefficients are output on the

teletype and stored in array COEFF, in descending order, for further use. The actual least squares regression is performed by sub-routine STPLRG which is an adaptation of the Honeywell program STPLRG

5.2.5 ENTRER

This program permits entry of coefficients into array COEFF for use by other programs. The number of coefficients is requested via the teletype and they may be entered via either the teletype or the paper tape reader, the particular device being specified on the teletype.

5.2.6 DIMOD 3

This is a special purpose data transformation program intended to convert crystal diameter measurements made at equal distance increments along the crystal into diameter measurements made at equal time intervals while the crystal was growing. The flowchart for DIMOD 3 is shown in Fig 5.6. Note that as well as the measured diameter profile of the crystal, the program also requires its second derivative, this being in a file on magnetic or paper tape in the same format as the diameter. The algorithm used for the transformation is derived in the chapter on data processing. Note that in order to allow a data output representing diameter at time increments of $n \Delta t$ it is necessary to be able to specify a non-integer data sampling interval. The operation of this facility is illustrated in the flow chart. The algorithm is very noise sensitive, in that if due to noise on the second derivative, the slice thickness is calculated as being negative (ie computed melt back), a discontinuity will be produced on the output crystal profile. For this reason it has been found to be necessary to use as large as possible an incremental slice thickness, so that random noise does not produce negative thicknesses. The program uses internal storage for all calculations, after completion of time linearisation, the original

data input array is over-written with the modified profile and its length variable set to the new (re-sampled) length.

5.2.7 PUTOUT

This is a general purpose data output program. It outputs the data in array DATA (the length being automatically specified by the variable IR). It allows inspection of the data on a 'scope connected to the D-A channels of the PDP 9 and/or on the graph plotter. Initial option selection is via the accumulator switches, as shown in the flowchart of Fig 5.7. The normal operating procedure within DPPP would be to set up the accumulator switches before selecting PUTOUT. If 'Output Data' is selected, PUTOUT requests on which data channel the output is required. If Dectape is selected then the filename is required. Data is written in serial form on the specified device (ie teletype, paper punch, or Dectape).

5.2.8 TRANS

This program transforms the coefficients of a non-recursive, lowpass digital filter into the coefficients of either

1. Highpass
2. Bandpass
3. Bandstop

non-recursive digital filter. The transformations used are due to Constantinides.[58] The coefficients of z in the lowpass digital filter are transformed to $F(z)$.

For each of the 3 cases respectively

1. $F(z) = -z^{-1}$
2. $F(z) = -z^{-2}$
3. $F(z) = z^{-2}$

Note that in the case of the bandpass and bandstop transformations the number of coefficients are doubled, albeit by the insertion of zero valued coefficients.

If FN = Nyquist frequency of lowpass prototype

FL = Cut-off frequency of lowpass prototype

F0 = Centre frequency

F1 = Lower frequency

F2 = Upper frequency

Then for

Highpass F0 = FN - FL

Bandpass F1 = FN/2 - FL/2

F2 = FN/2 + FL/2

F0 = FN/2

Bandstop F1 = FL/2

F2 = FN - FL/2

F0 = FN/2

These 3 transformations are illustrated in Fig 5.8.

5.3 SPECTA - SPECTRAL ANALYSIS PACKAGE

SPECTA consists of a main program, GHOST, which is responsible for inputing data, calling the processing subroutines and enabling the output of data. The flowchart for GHOST is shown in Fig 5.11. In general the package computes 5 functions, these are

a. Power spectrum of data file 'A' (decibels)
b. Power spectrum of data file 'B' (decibels)
c. Cross power spectrum of 'A' and 'B' (decibels and degrees)
d. System frequency response based on input of 'A' and output of 'B' (decibels and degrees)
e. Coherence function of input and output of system

The data may be input from the teletype, paper tape reader or from Dectape, the particular device being chosen by selecting a data channel via the teletype immediately after program loading. Four different output devices may be selected at run time. Selection is by means of the accumulator switches. The 4 output devices are

a. Scope output
b. Graph plotter
c. Paper tape
d. Dectape.

SPECTA is dimensioned to accept data files 'A' and 'B' of 508 real numbers each.

The kernel of this package is a fast Fourier transform algorithm by Singleton.[59] This is a mixed radix transform and can handle data streams containing prime numbers up to and including 127. Data is read into 2 arrays, XIN and XOUT. The data remains untouched in these arrays throughout any program function, until new data is read in. Temporary internal arrays hold the data that is actually transformed and allow the phase of the input and output data streams to be adjusted. (To enable any system dead time to be trimmed out before transformation and hence allow better resolution of the phase characteristic).

5.3.1 POWER SPECTRA - G_{xx}, G_{yy}, G_{yx}

If U(k) and V(k) with k = 1, n are arrays storing n data points of the system input and output and $U'(k_1)$ and $V'(k_1)$ with $k_1 = 1, \frac{n}{2}$ are the fast Fourier transformed arrays, then let

$$k_2 = n + 2 - k_1 \quad \ldots\ldots 502$$

$$A_{k_1} = \cdot 5(U'(k_1) + U'(k_2)) \quad \ldots\ldots 503$$

$$B_{k_1} = \cdot 5(V'(k_1) - V'(k_2)) \quad \ldots\ldots 504$$

$$X_{k_1} = \cdot 5(V'(k_2) + V'(k_1)) \quad \ldots\ldots 505$$

$$Y_{k_1} = \cdot 5(U'(k_2) - U'(k_1)) \quad \ldots\ldots 506$$

then

$$G_{xx}(k_1) = A^2_{k_1} + B^2_{k_1} \quad \ldots\ldots 507$$

$$G_{yy}(k_1) = X^2_{k_1} + Y^2_{k_1} \quad \ldots\ldots 508$$

$$G_{yx}RE = X_{k_1} A_{k_1} + Y_{k_1} B_{k_1} \quad \ldots\ldots 509$$

$$G_{yx}IM = Y_{k_1} A_{k_1} - X_{k_1} B_{k_1} \quad \ldots\ldots 510$$

For odd harmonics k_1 increments from 2, $\frac{n}{2}$ by 2, for odd and even harmonics k_1 increments by 1 (reference Bendat and Peirsol).[56]

The power spectra are converted to dB and degrees before output on to the graph plotter, paper tape, magnetic tape or the C.R.O, as specified.

5.3.2 SYSTEM TRANSFER FUNCTION

Again, following Bendat and Peirsol,[56] the system transfer function, H(f) is estimated from the input power spectrum and the cross power spectrum.

$$H(f) = \frac{G_{yx}(f)}{G_{xx}(f)} \qquad \ldots\ldots\ 511$$

where $G_{yx}(f)$ and $G_{xx}(f)$ are smoothed estimates.

Hence the gain and phase factors of the system transfer function are given by

$$|H(f)| = \frac{G_{yx}(f)}{G_{xx}(f)} \qquad \ldots\ldots\ 512$$

$$\phi(f) = \operatorname{atan}\left(\frac{G_{yxIm}}{G_{yxRE}}\right) \qquad \ldots\ldots\ 513$$

5.3.3 COHERENCE FUNCTION

The ordinary coherence function between 2 stationary time records is defined by,[56]

$$\gamma^2_{xy}(f) = \frac{|G_{yx}(f)|^2}{G_{xx}(f)G_{yy}(f)} \qquad \ldots\ldots\ 514$$

In terms of digital estimates of power spectras,

$$\hat{\gamma}^2_k = \frac{|G_{yx}(f_k)|^2}{G_{xx}(f_k)G_{yy}(f_k)} \qquad \ldots\ldots\ 515$$

Note that the spectras must be smoothed estimates, ie derived from more than one block of data. Otherwise the coherency will always be unity. The computed coherency may be output on the graph plotter, paper tape, magnetic tape or the teletype, as before.

The structure of the program permits the use of overlapped fast Fourier transform processing, which together with the use of a Hanning data window reduces the variance of the coherency estimate for a given amount of data. (Carter, Knapp and Nuttall)[53]

5.3.4 HANNING WINDOW

The measured data may be multiplied by a Hanning window as shown in Fig 5.13 and defined below

$$W(nT) = \cdot 5(1-\text{Cos}(\frac{\pi nT}{a})) \qquad 0 \leq nT < a$$

$$W(nT) = 1 \qquad a \leq nT < (b-a) \qquad \ldots\ldots\ldots 516$$

$$W(nT) = \cdot 5\left\{1-\text{Cos}\,\pi\left(\frac{b-nT}{b-a}\right)\right\} \qquad (b-a) \leq nT < b$$

The percentage taper, P may be specified and is defined by

$$P = \frac{100(2a)}{b} \qquad \ldots\ldots\ldots 517$$

5.3.5 ZERO ORDER HOLD CORRECTION

Since a continuous system is being represented by a sampled data sequence, an implicit zero order hold is introduced at the system input. The subsequent estimation of the overall frequency response function will therefore include the frequency response of this zero order hold. An option is available within the package to remove this modification to the actual frequency response.

5.4 FITF - FREQUENCY RESPONSE CURVE FITTING PACKAGE

This package has 2 inter-related functions

1. To produce a graphical output of frequency response, (gain and phase) from a system transfer function
2. To fit transfer functions to frequency response data.

One of these modes is selected via the teletype immediately after program loading is completed. The overall flowchart is shown in Fig 5.9. The major segments of this flowchart are described in the following sections.

5.4.1 FREPIC

This subroutine calculates the modulus and phase from transfer functions which may be specified in any of 3 different forms. These are a. Ratios of polynomials in s

b. Poles and zeros with a gain constant
c. Time constants with a gain constant

Frequency responses are calculated at 200 logarithmically spaced frequencies between limits that are specified on the teletype. Where the characteristic polynomial of a transfer function is input, a check is made to see if the Routh stability criterion is met. If it

is not (ie RH half poles exist), then a message to this effect is output on the teletype. Computation then proceeds as normal. The actual frequency response computation is performed by subroutines QUFREQ, given polynomials, or PZFREQ given poles and zeros or time constants. Conversion from real and imaginary components to modulus and phase is performed by subroutine RIMP. There is storage for up to 10 numerator coefficients and 10 denominator coefficients.

5.4.2 QUFREQ

This subroutine accepts arrays of coefficients for the system transfer function in the form $\frac{X_1(s)}{X_2(s)}$ where X are polynomials in s. It also accepts a frequency array, of the frequencies at which the transfer function is to be evaluated. It calculates the vector numerator and denominator separately and then combines them into a rational vector quantity at each frequency. If the system transfer function is defined as

$$H(s) = \frac{X_1}{X_2} = \frac{\sum_{j=0}^{n} a_j s^j}{\sum_{j=0}^{m} a_j s^j} \qquad \ldots\ldots 518$$

$$\text{then } R_e(X_1) = \sum_{j=0}^{n} (-1)^j a_{2j} \omega^{2j} \qquad \ldots\ldots 519$$

$$I_m(X_1) = \sum_{j=0}^{n} (-1)^j a_{2j+1} \omega^{2j+1} \qquad \ldots\ldots 520$$

(similar expressions for X_2)

$$\text{and } R_e(H) = \frac{R_e(X_1)R_e(X_2) + I_m(X_1)I_m(X_2)}{R_e(X_2)^2 + I_m(X_2)^2} \qquad \ldots\ldots 521$$

$$I_m(H) = \frac{R_e(X_2)I_m(X_1) - I_m(X_2)R_e(X_1)}{R_e(X_2)^2 + I_m(X_2)^2} \qquad \ldots\ldots 522$$

5.4.3 PZFREQ

This subroutine accepts arrays of poles and zeros and a frequency array. It produces arrays of gain and phase in dB^s and degrees. If the transfer function is in the form

$$H(s) = \frac{k\prod_{j=1}^{n}(s+a_j)}{\prod_{j=1}^{m}(s+b_j)} \qquad \ldots\ldots\ldots 523$$

the subroutine calculates modulus and phase thus

$$|H| = 10^{\left(2\log_{10}K + \sum_{j=1}^{n}\log_{10}(a_j^2+\omega^2) - \sum_{j=1}^{m}\log_{10}(b_j^2+\omega^2)\right)} \qquad \ldots\ldots\ldots 524$$

$$\arg(H) = \frac{180}{\pi}\left(\sum_{j=1}^{n}\operatorname{atan}\left(\frac{\omega}{a_j}\right) - \sum_{j=1}^{m}\operatorname{atan}\left(\frac{\omega}{b_j}\right)\right) \qquad \ldots\ldots\ldots 525$$

5.4.4 FFIT

The basis of this subroutine is a complex curve fitting algorithm as described by Sanathanan and Koerner[61], which is itself an iterative form of the algorithm described by Levy[60]. Both these forms suffer from the disadvantage that transfer functions are fitted to vector, (real and imaginary) frequency responses. Their convergence is seriously affected if the frequency responses are corrupted by noise, which generally has a greater influence on the phase characteristic than the modulus. A better fit would be obtained if fitting to modulus only were performed. It will be shown that an analytical solution for the minimisation of the cost function of sum of squares of error does not exist for the case of a fit to modulus only. However the method of Sanathanan and Koerner has been adapted to provide an iterative method of fitting to modulus only.

To recap on the approach of Levy

If the measured frequency response is

$$F = R + jI \qquad \ldots\ldots\ldots 526$$

and the approximation to the measured frequency response is

$$H = \frac{N}{D} \qquad \ldots\ldots\ldots 527$$

with N and D being polynomials in s.

Under steady state conditions, replace s by $j\omega$ and re-group the polynomials thus

$$\frac{N}{D} = \frac{(A_o - A_2\omega^2 + A_4\omega^4 - \cdots) + j\omega(A_1 - A_3\omega^3 + A_5\omega^5 -)}{(B_o - B_2\omega^2 + B_4\omega^4 - \cdots) + j\omega(B_1 - B_3\omega^3 + B_5\omega^5 -)} \qquad \ldots\ldots 528$$

$$\frac{N}{D} = \frac{\alpha + j\omega\beta}{\sigma + j\omega\tau} \qquad \ldots\ldots\ldots 529$$

at each frequency, the error of fitting is therefore

$$\varepsilon = F - H \qquad \ldots\ldots\ldots 530$$

$$\varepsilon = (R+jI) - \frac{(\alpha + j\omega\beta)}{(\sigma + j\omega\tau)} \qquad \ldots\ldots\ldots 531$$

$$D\varepsilon = R\sigma - \omega\tau I - \alpha + j(\omega\tau R + \sigma I - \omega\beta) \qquad \ldots\ldots\ldots 532$$

Define E as the sum of squares of 516, summed over all frequency points

$$E = \sum_{k=o}^{m} \left[(R_k\sigma_k - \omega_k\tau_k I_k - \alpha_k)^2 + (\omega_k\tau_k R_k + I_k\sigma_k - \omega_k\beta_k)^2 \right] \qquad \ldots\ldots\ldots 533$$

The unknown coefficients of the fitted transfer function may now be derived by differentiating E with respect to the coefficients and equating to zero, thus minimising E. Observe that a set of linear simultaneous equations results from this procedure.

Note that in equation 532 it has been necessary to multiply out by D, the denominator polynomial. The result of this is that the error criterion that is minimised is in fact a weighted error, with a greater weighting to the higher frequency measured frequency response points. Sanathanan and Koerner[61] show how this weighting may be effectively removed by minimising a modified error criterion E', and using an iterative technique.

$$E' = \sum_{k=o}^{m}((R_k\sigma_k - \omega_k\tau_k I_k - \alpha_k)^2 + (\omega_k\tau_k R_k + I_k\sigma_k - \omega_k\beta_k)^2)W_{kL} \quad \ldots\ldots 534$$

where

$$W_{kL} = \frac{1}{|D_{L-1}|^2} \quad \ldots\ldots 535$$

with L being the iteration count.

Observe that for both Levy and Sanathanan and Koerner the crucial step in obtaining linear simultaneous equations to solve for the polynomial coefficients has occurred at equation 532. If a fit to modulus was required, then the error at each frequency would be defined as the difference between the moduli of the measured and fitted frequency responses, thus

$$\varepsilon'' = |F| - |H| \quad \ldots\ldots 536$$

$$\varepsilon'' = (R^2 + I^2)^{1/2} - (\alpha^2 + \omega^2\beta^2)(\sigma^2 + \omega^2\tau^2)^{1/2} \quad \ldots\ldots 537$$

there is no way that ε'' can be weighted to yield, when differentiated, a set of linear simultaneous equations. A hill climbing technique could be used to solve for the unknown coefficients. Since the dimensionality of such a hill climb could soon prove unweildy for high orders of fit, a new iterative technique has been developed. The error criterion of equation 534 is used, but with modified "measured" frequency response points, ie R_L' and I_L'. After the first set of transfer function coefficients has been produced, via equation 534, the phase information implied in R_k and I_k is updated to the phase implied in H (preserving the modulus). This is illustrated in Fig 5.10. It has been found that convergence of the algorithm is improved if the initial phase information of the measured frequency response is entered as a linear phase characteristic, from 0 to -180^o, even for higher order fits.

5.5 ILLUSTRATION OF OPERATION

The operation of FITF is illustrated in Figures 5.14.1 and 5.14.2. These are frequency responses computed within the package from a transfer function:

$$H(s) = \frac{135135-135135s+62370s^2-17325s^3+3150s^4+378s^5+28s^6-s^7}{135135+135135s+62370s^2+17325s^3+3150s^4+378s^5+28s^6+s^7} \quad \ldots\ldots 538$$

This function was computed at 200 logarithmically spaced points over a frequency range from ·01 to 10Hz.

The data was then fitted with a 7/7th order fit to modulus with 10 iterations. The sum of squares of error of fit was $8{\cdot}09\times10^{-5}$.

The fitted frequency responses are in fact superimposed on the original curves in Figures 5.14.1 and 5.14.2.

The operation of DPPP and SPECTA are illustrated in the following examples. Figure 5.15 shows the transform of a digital filter designed within DPPP. Its parameters are as follows:

Initial length 100 points

$$\frac{f_{cuttoff}}{f_{clock}} = 0{\cdot}1$$

Hamming windowed down to 8 points.[62]

This filter was then used to filter two periods of a ternary pseudo-random sequence. The input and output sequences are shown in Figure 5.16. Both these sequences were then read into SPECTA and their power spectras computed. These are shown in Figure 5.17 (complete periods from the centres of the original sequences were used).

FIGURE 5.1

FLOWCHART FOR DPPP

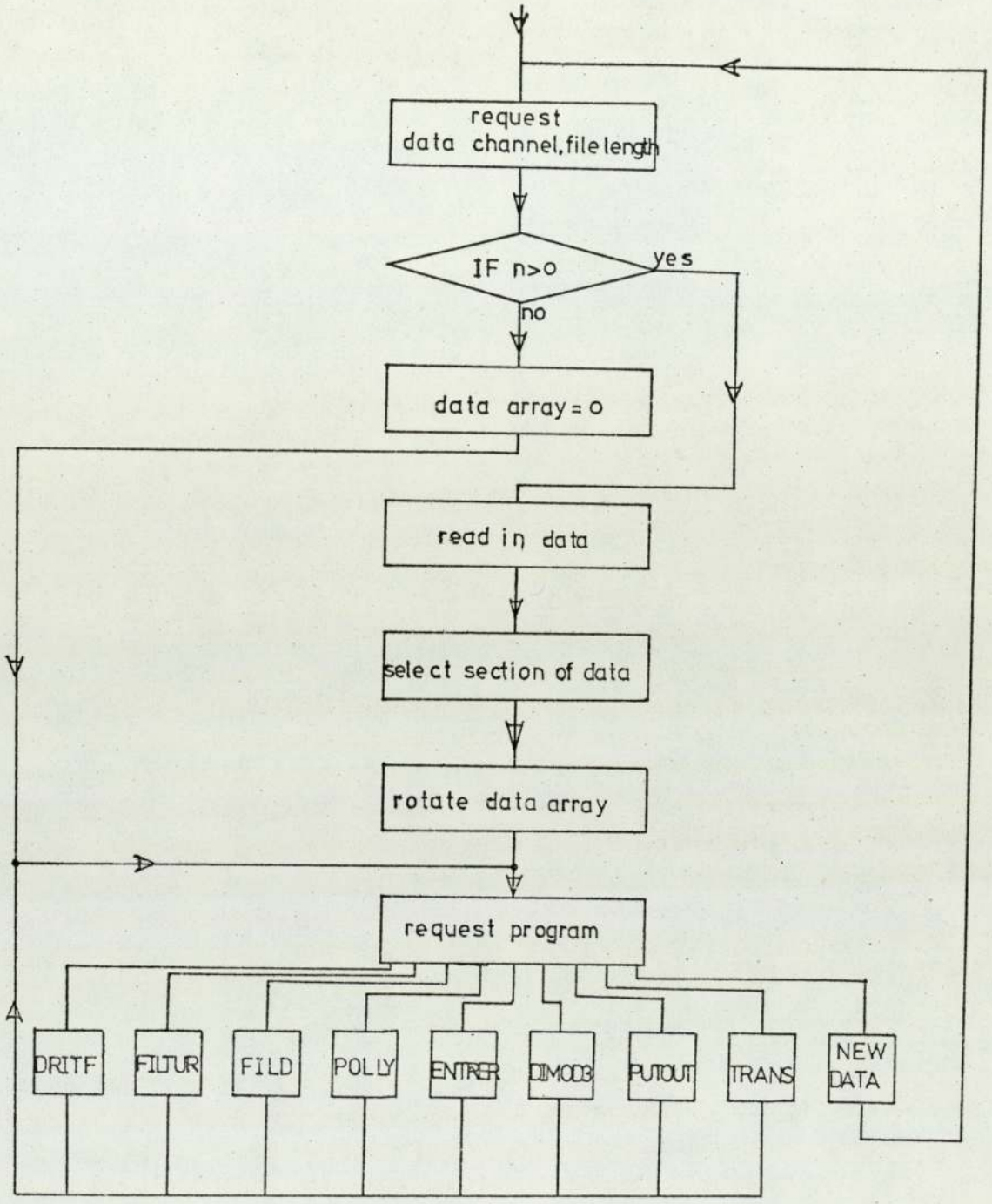


FIGURE 5.2

FLOWCHART FOR DRITF

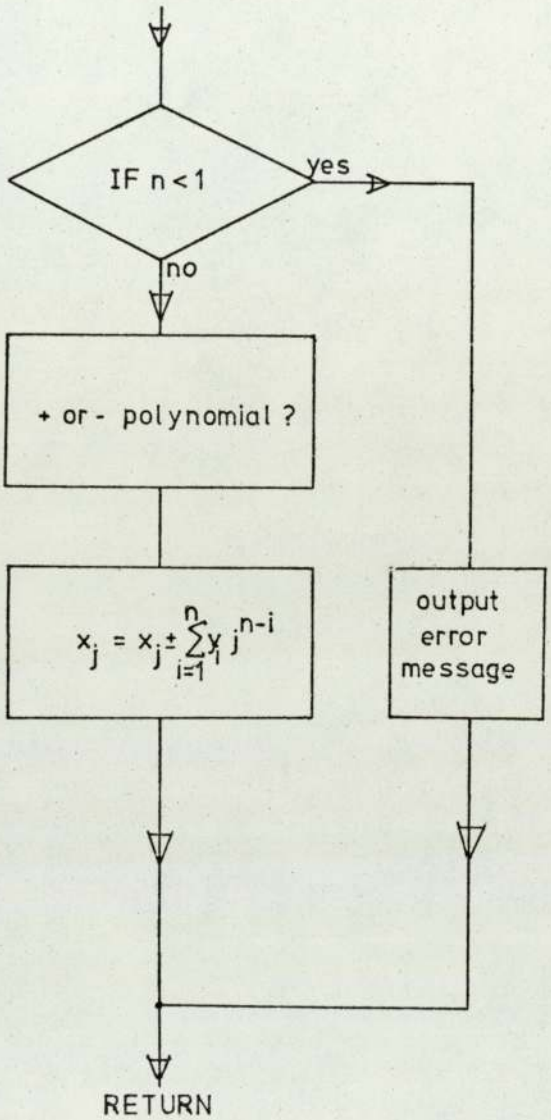


FIGURE 5.3
STRUCTURE OF FILTUR

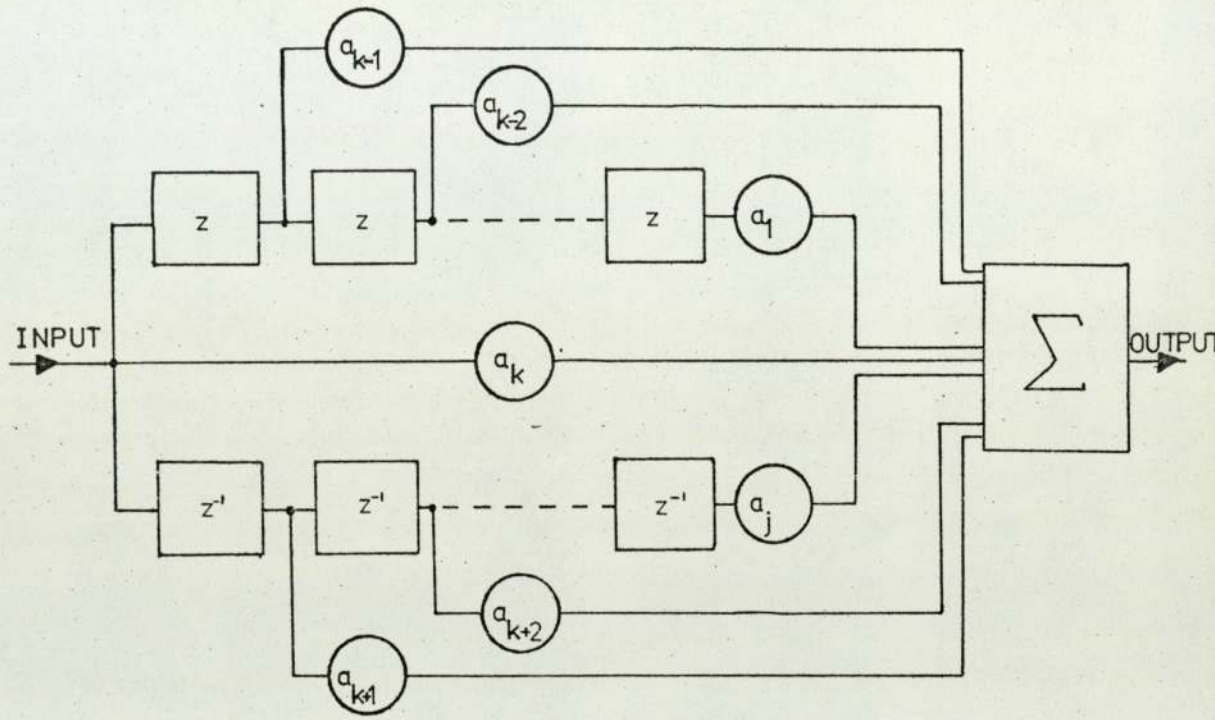


FIGURE 5.4
CONVOLUTION OF WEIGHTING FUNCTION WITH DATA STREAM

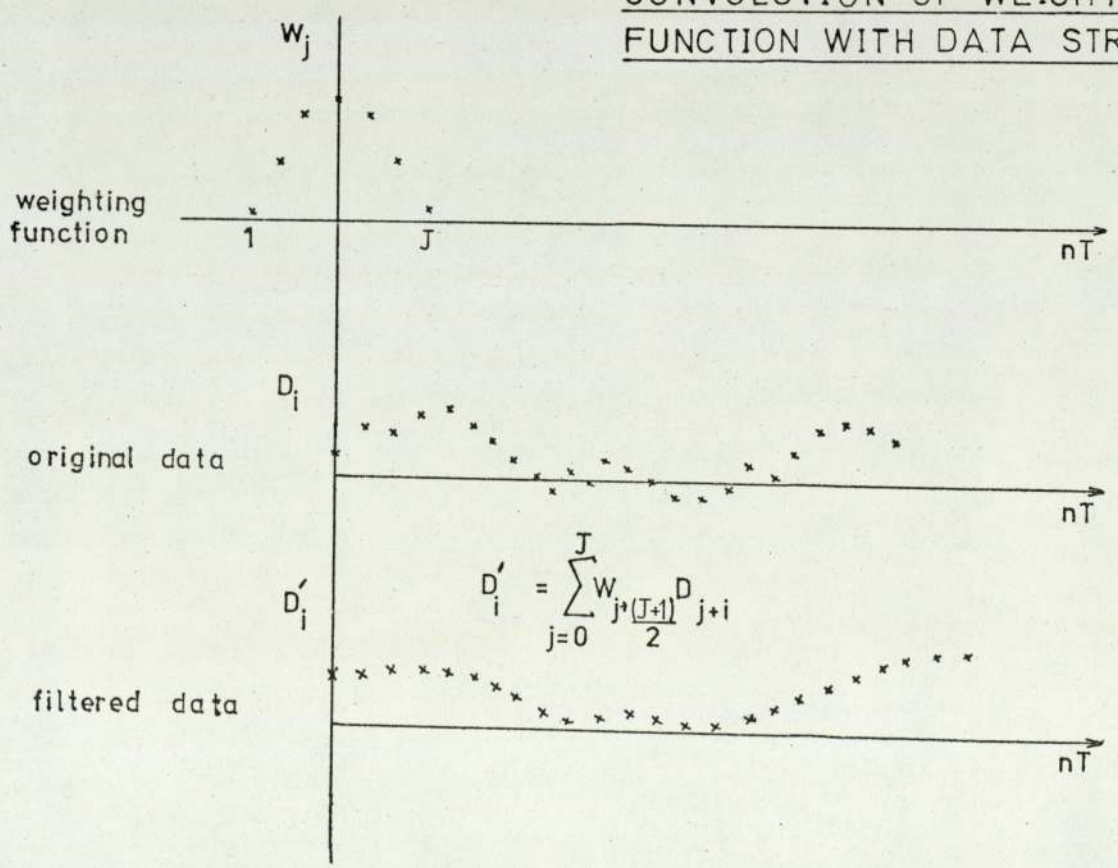
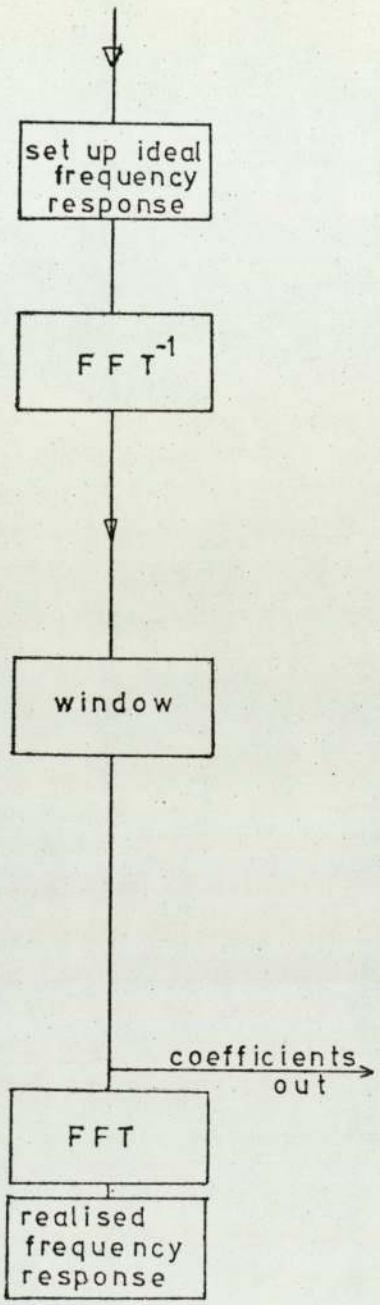


FIGURE 5.5

FLOWCHART FOR FILD



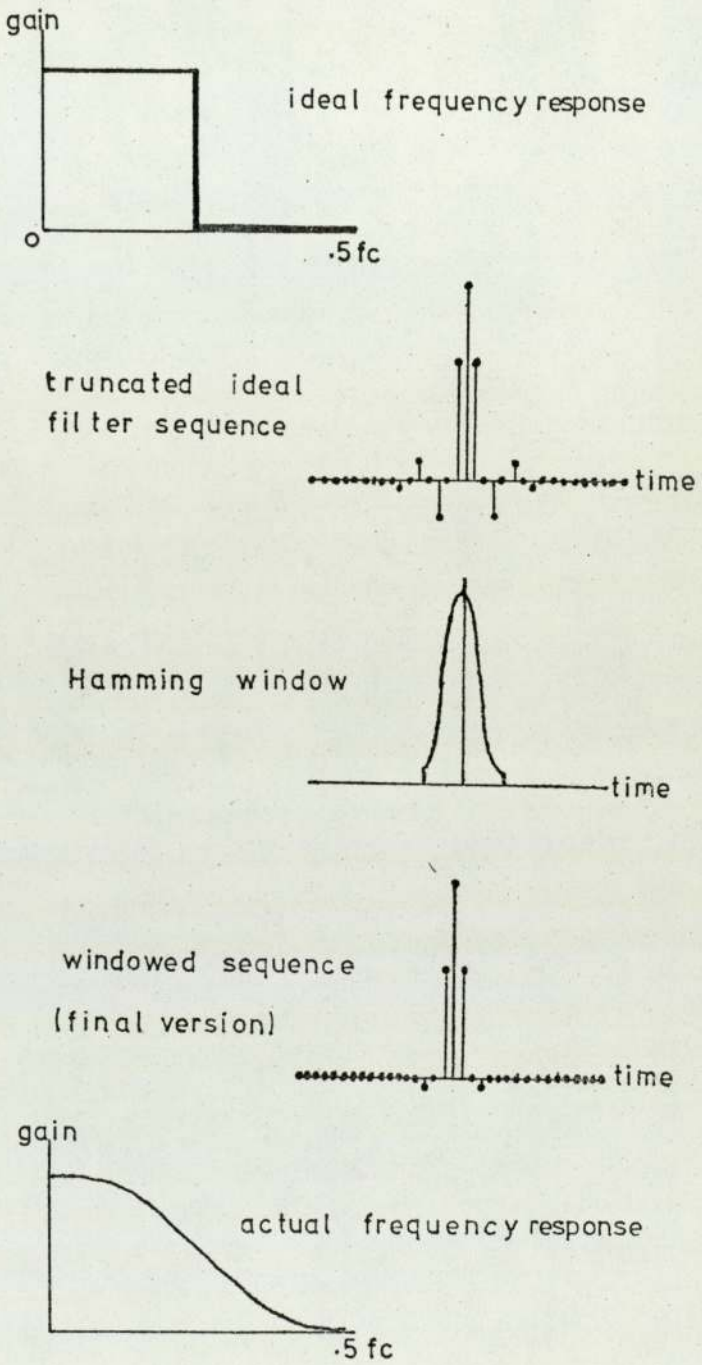
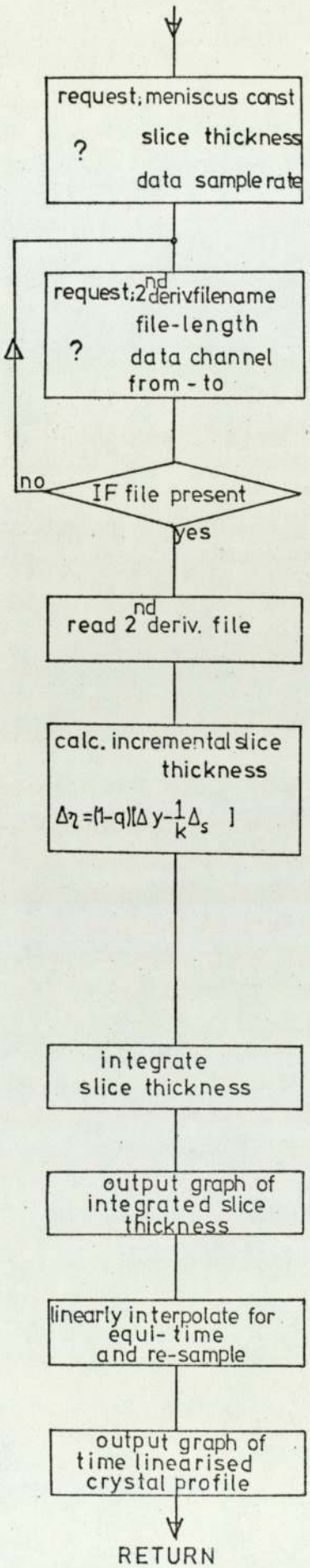


FIGURE 5.6
FLOWCHART FOR DIMOD3



diagramatic representation

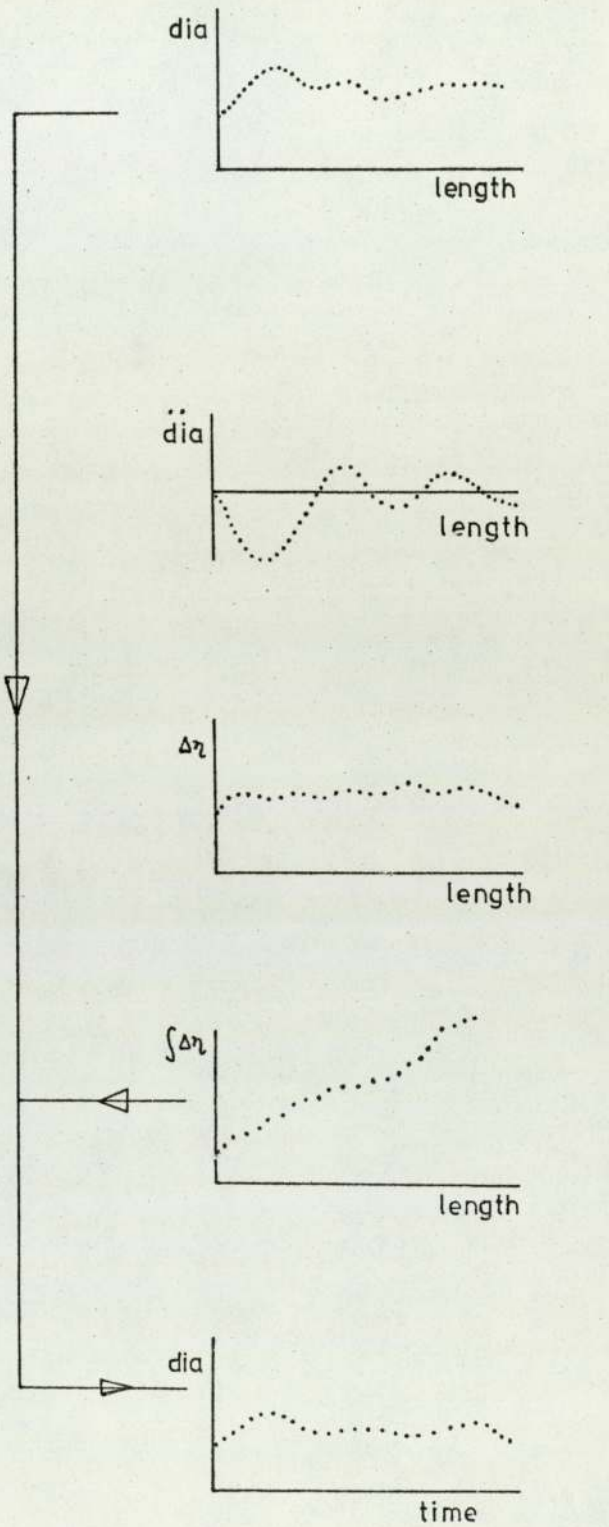


FIGURE 5.7

FLOWCHART FOR PUTOUT

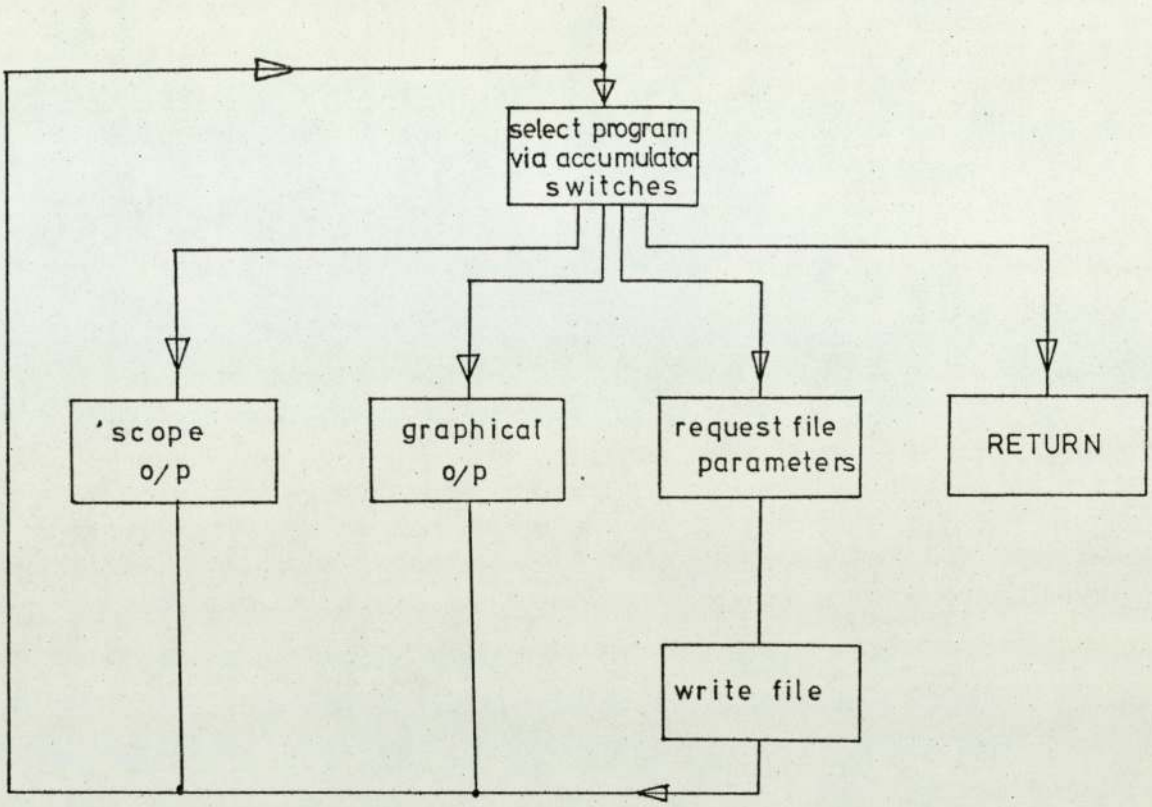


FIGURE 5.8

FREQUENCY TRANSFORMATIONS BY 'TRANS'

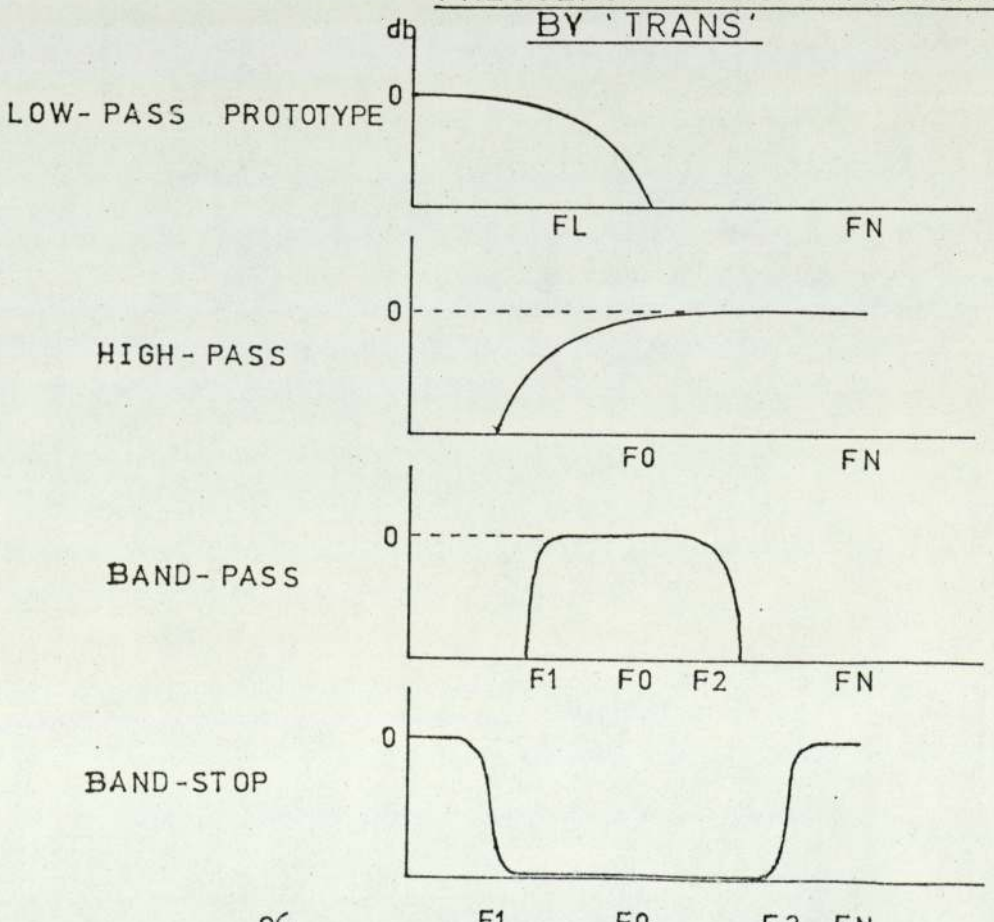


FIGURE 5.9

FLOWCHART FOR 'FITF'

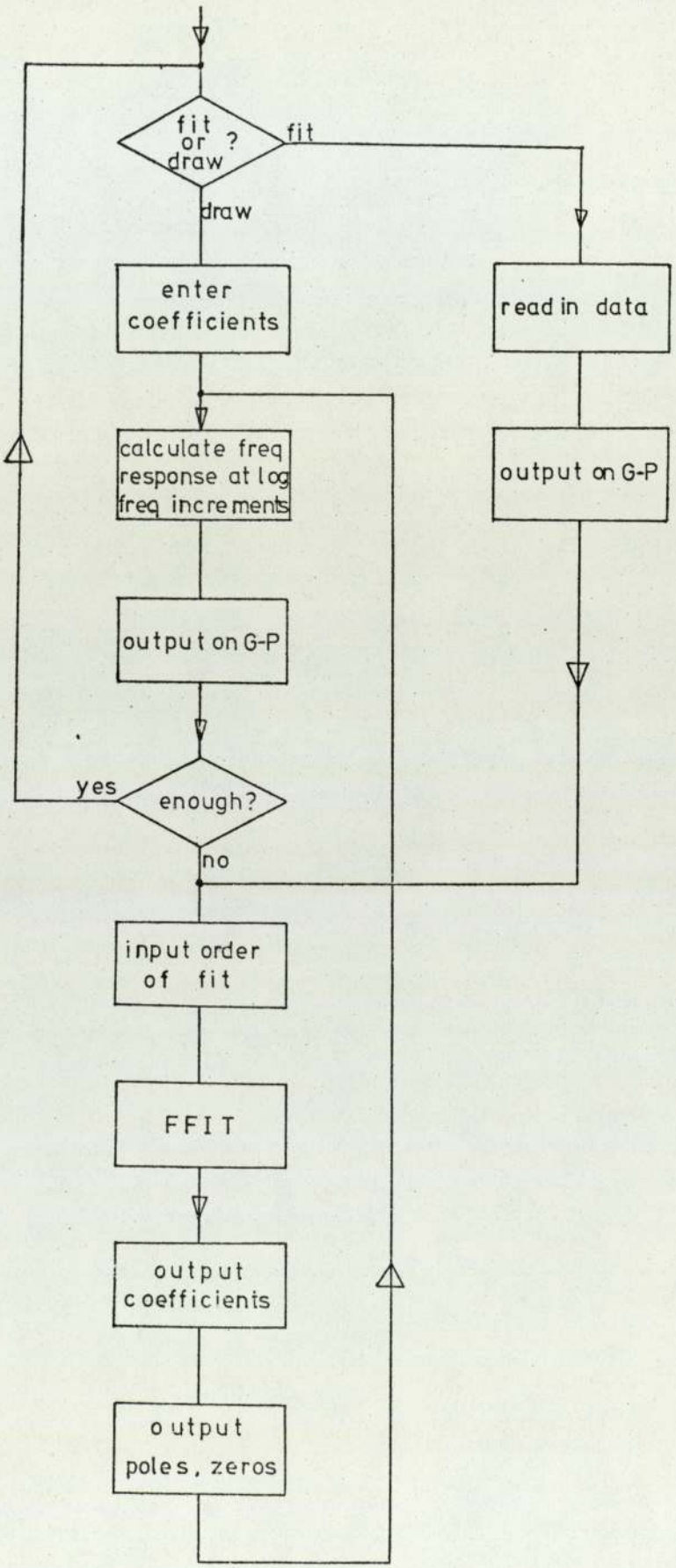


FIGURE 5.10

OPERATION OF 'FFIT'

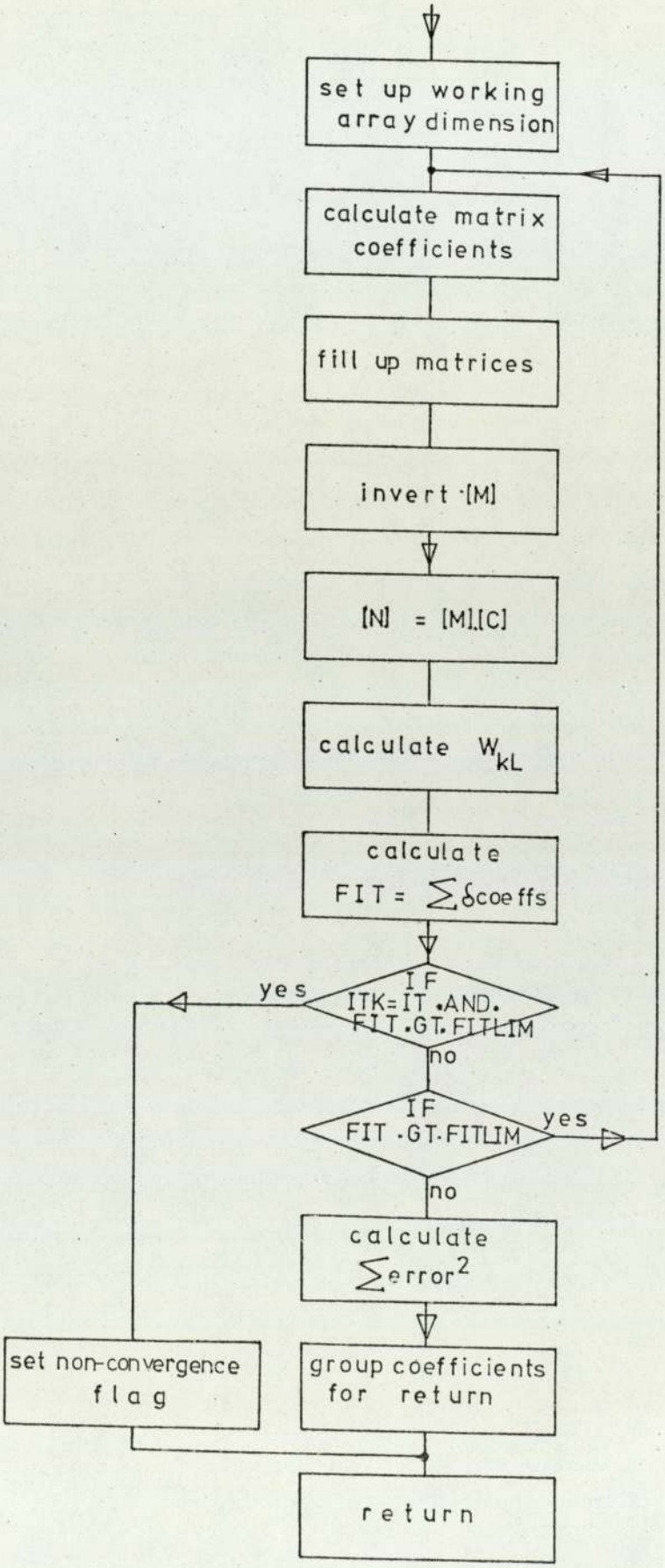


FIGURE 5·11

FLOWCHART FOR GHOST

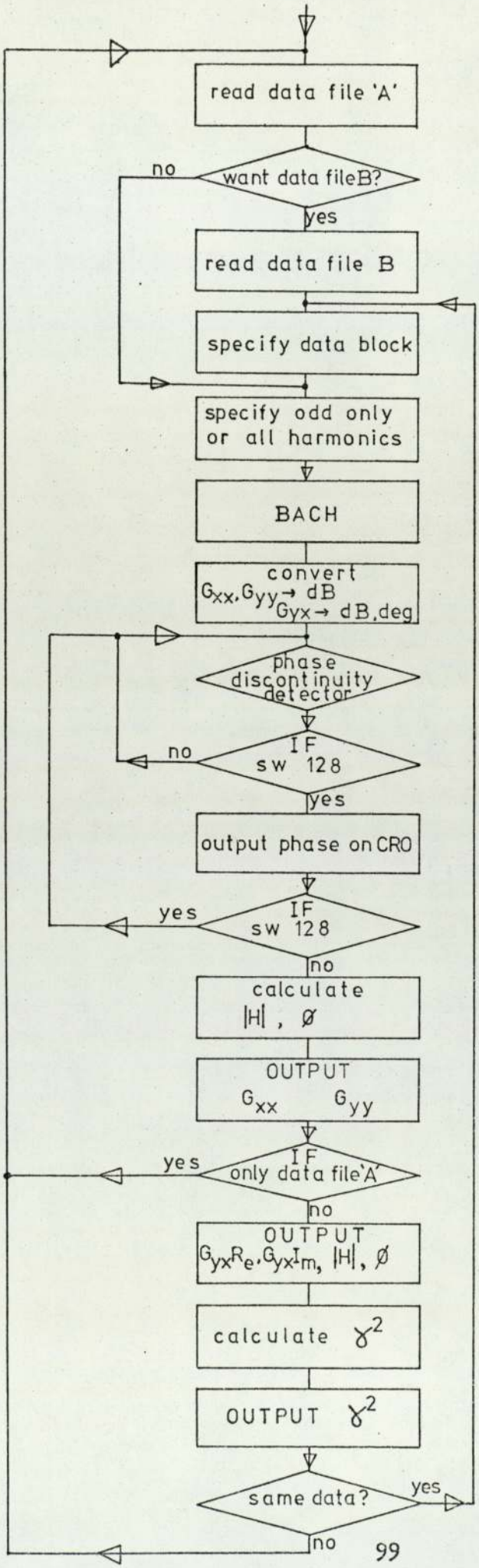


FIGURE 5.12

FLOWCHART FOR BACH

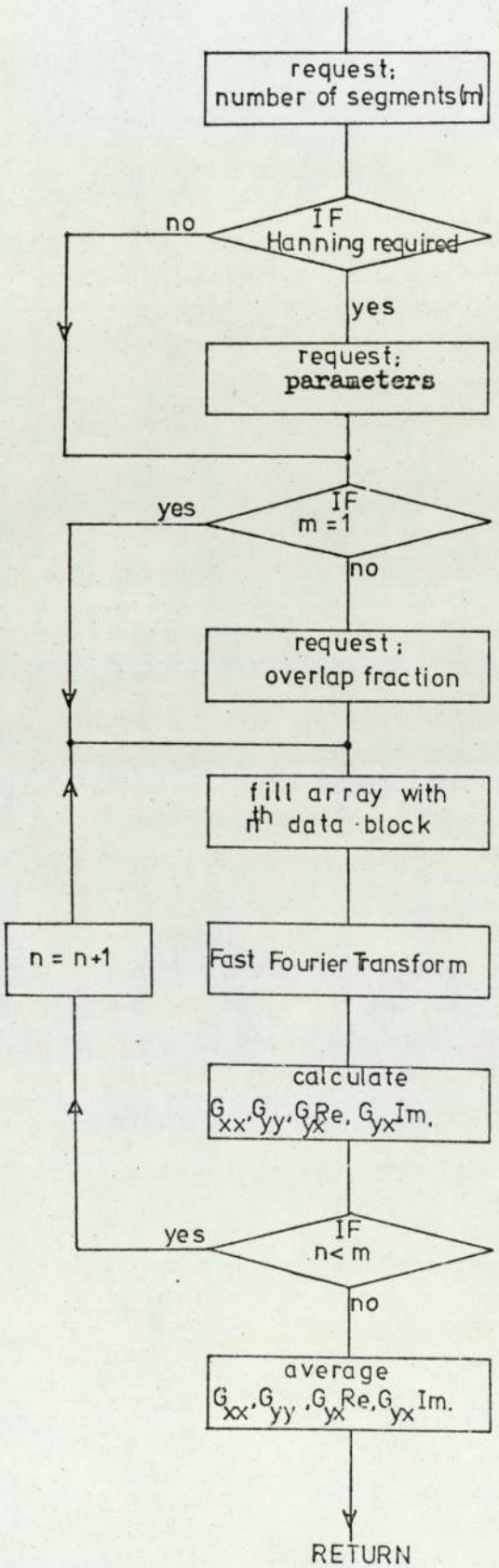


FIGURE 5.13

HANNING WINDOW

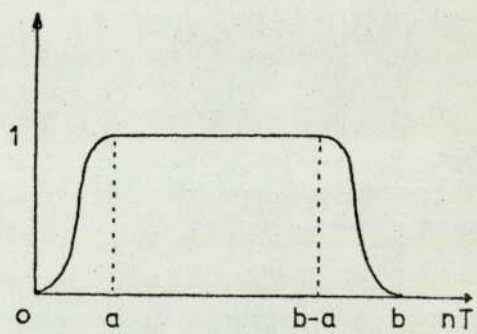


Figure 5.14.1 Initial And Fitted Frequency Responses (Modulus)

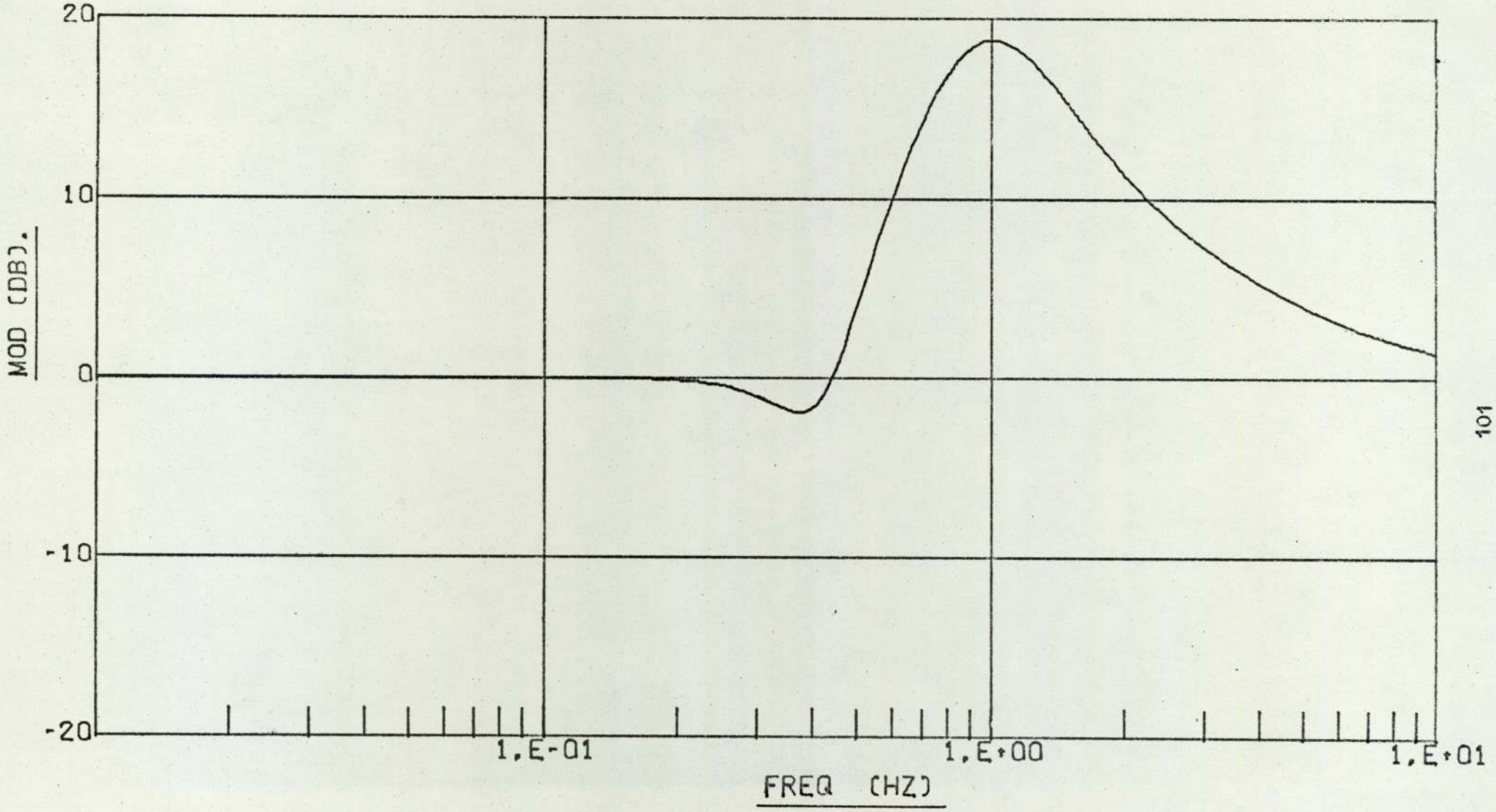


Figure 5.14.2 Initial And Fitted Frequency Responses (Phase)

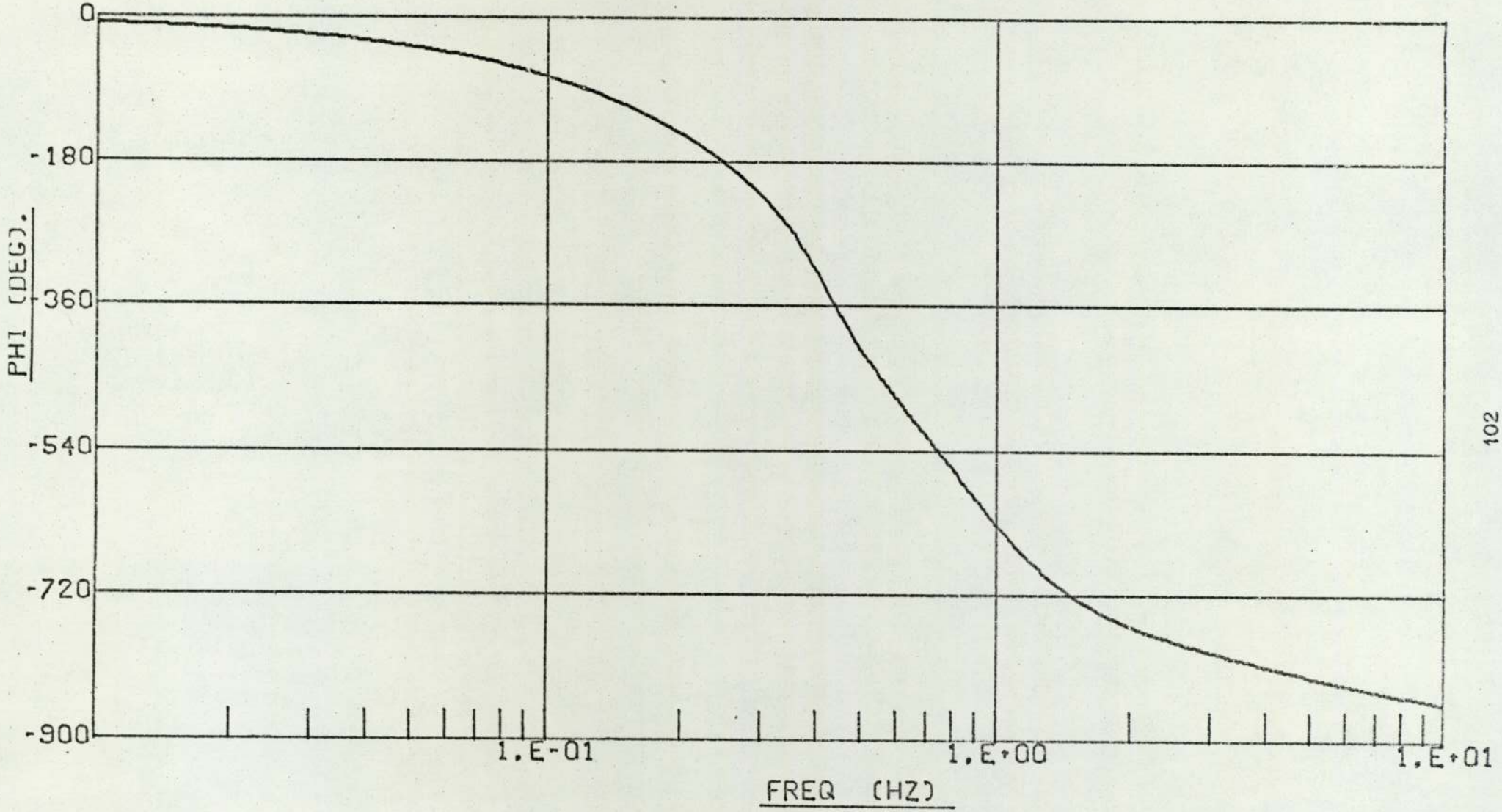


Figure 5.15 Frequency Response of Designed Digital Filter

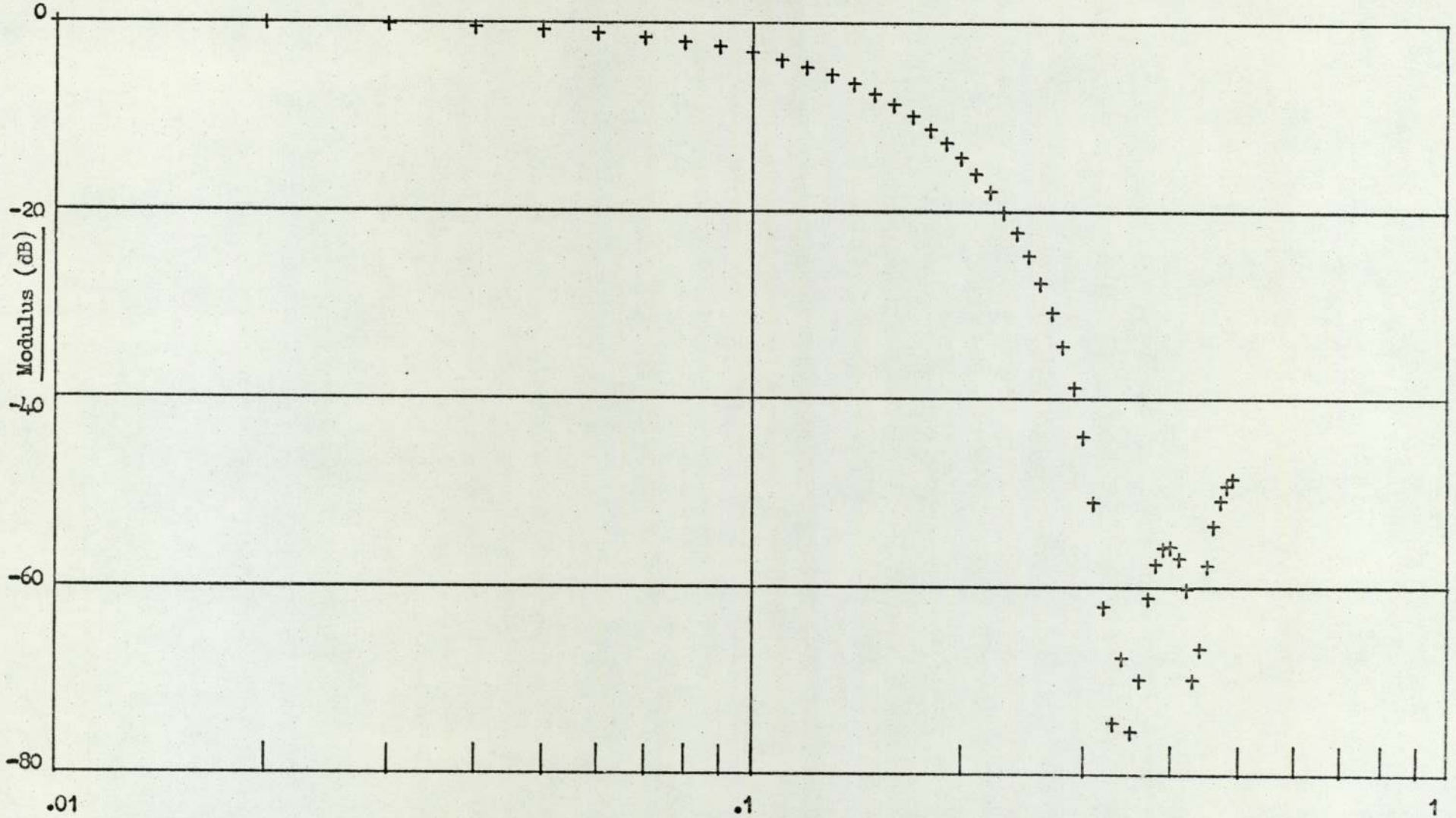
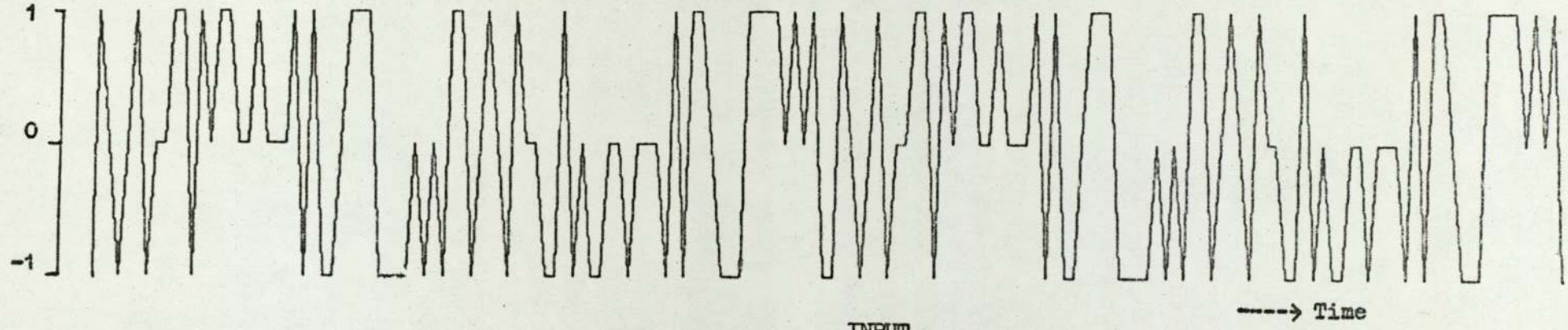
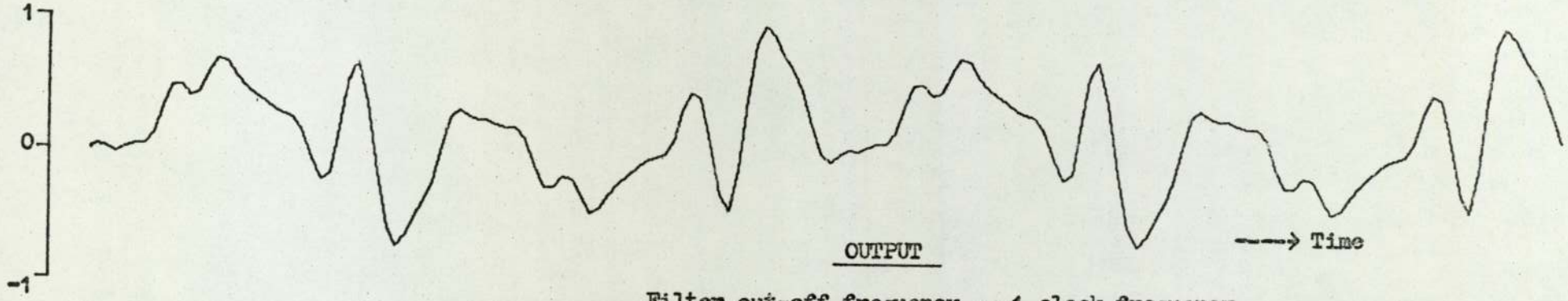


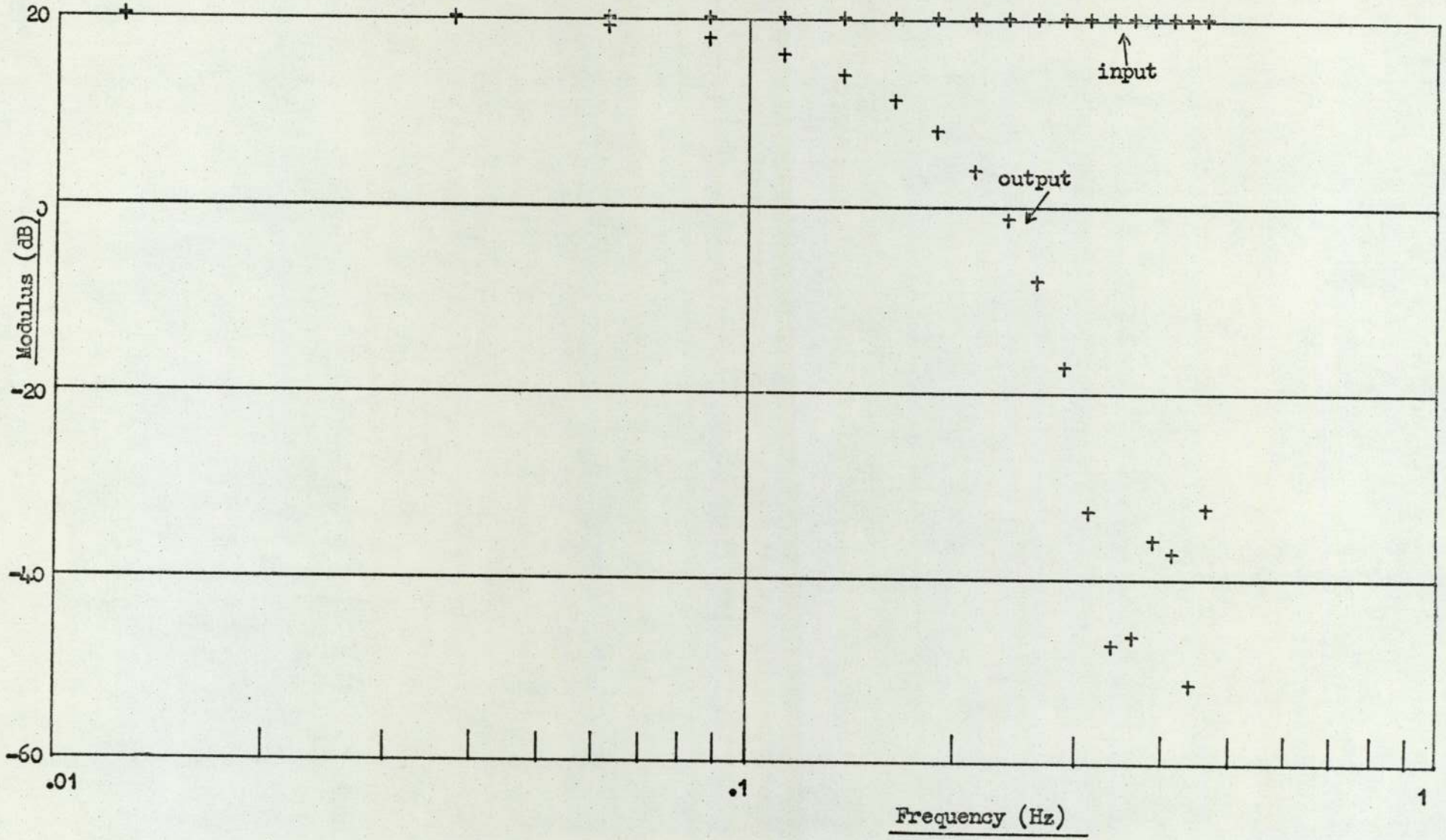
Figure 5.16 Input And Output Time Series of Digital Filter





Filter cut-off frequency = .1 clock frequency

Figure 5.17 Input And Output Power Spectra of Digital Filter



6. CONCLUSIONS

6.1 MAIN FEATURES OF METHOD

An experimental method has been established to measure the dynamics of Czochralski crystal growth systems, relating crystal radius to changes of heater power. This has involved the use of special techniques of data processing which are believed to be novel and original. The method involves the injection of a pseudo-random sequence into the power controller of the crystal pulling system. Subsequently, radius measurements are made along the length of the cold crystal. A new processing technique has been developed to enable the radius measurements made at equal distance intervals to be converted to radius measurements made at equal time intervals in order to perform frequency domain processing using the fast Fourier transform. This is believed to be the first application of pseudo-random signal testing to crystal growth systems.

A new method of transfer function curve fitting has been developed that enables transfer functions to be fitted to modulus data only. This method is of general application to systems having transport delay that would produce wide ranges of phase. A theoretical analysis of the Czochralski crystal growth process has been carried out and, within the limits of the experimental and theoretical accuracy, agreement has been achieved between the theoretical and practical results.

6.2 RELEVANCE AND LIMITATIONS

The experimental method that has been established is relevant to all Czochralski crystal growth systems. The limitations are:

1) The crystal must grow for long enough to enable at least one period of valid data to be collected.
2) The magnitude of the radius variation must be limited to ensure the validity of the assumption that the meniscus curve

between the melt and the crystal does not vary with crystal radius.

Since the Czochralski method is essentially a batch process and frequency domain processing requires averaged estimates obtained from several periods of data, there is a requirement to be able to grow at least five similar crystals with the same perturbation superimposed. The limited length of data from each crystal also means that the resultant transfer function will be a stationary approximation to a time varying process. The data limitation will in general not permit separate transfer function estimates to be made at the beginning and end of a crystal growth run.

In view of the above factors, the measured transfer function of the MSR6 crystal puller must be regarded as an illustration of the experimental method rather than a finalised measurement. More similar crystals are required for this.

The theoretical analysis is more restricted than the experimental method in that it assumes relatively slow growth and high thermal conductivity materials such as Si, Ge or GaP. (It is not applicable to oxide growth). The analysis shows how parameters from measurements on the steady state temperature fields in the crystal and the melt may be incorporated into the prediction of dynamic performance. The accuracy of the prediction for GaP has been limited by the lack of measurements for this material. It has been shown how the accuracy of the parameters involved in the prediction may be improved by using the measured transfer function. This work provides a basis for making comparisons between control systems for Czochralski crystal growth for different materials and configurations. The results have indicated that a significant dynamic lag exists between the melt temperature and the crystal radius.

6.3 SUGGESTIONS FOR FURTHER WORK

The most pressing requirement is to grow more GaP crystals to enable a reliable transfer function estimate to be made. It would be advantageous to grow a material such as Si in view of its considerably better documented parameters. This would also offer advantages of being easier to grow under manual control and crystals can also be produced at a higher growth rate. An ideal situation would be to use an automatically diameter controlled Si puller which would enable a particular nominal radius to be achieved. The effects of the B_2O_3 layer in GaP growth could be simulated by the addition of a suitable encapsulant. This would allow the theoretical structure described in Chapter 2 to be more positively verified.

A further area of work is in the measurement of the thermal field in the melt and GaP crystal. This could be achieved by the use of grown-in thermocouples.

Since a limitation of existing pressurised Czochralski crystal pullers is the lack of instrumentation, a useful application of the measured overall transfer function would be to a model reference control system. Any commanded power changes are fed to a real-time simulation of the process so that their likely effect can be examined. Such a simulation could be readily achieved with the use of a microprocessor.

Another possible area of work would be in the extension of the power to diameter transfer function to include the effects of a variable pull speed. This could find use in a control system in which rapid, short term adjustments were made to the pull speed but longer term control action was achieved by heater power variation.

APPENDIX I

Journal of Crystal Growth 30 (1975) 45–53

ANALYSIS OF THE TRANSFER FUNCTION GOVERNING CRYSTAL GROWTH IN THE CZOCHRALSKI PROCESS

G.K. STEEL and M.J. HILL
University of Aston, Electrical Engineering, Department, Birmingham, England

Received 14 March 1975; revised manuscript received 21 April 1975

A transfer function relating the crystal radius to the heater power is derived assuming small deviations from the steady state growth conditions. The derivation is based on the conditions of heat balance in the zone near the growth interface. The transfer function is expressed in terms of the physical constants of the material being grown together with values of the steady state temperature gradients in the crystal and the melt, which can be measured experimentally. It is shown that the dynamics of heat transfer in the interface zone represent a significant lag in the overall transfer function. The results also provide a means of estimating the effect of changes in the growth conditions on the dynamic response and of comparing the behaviour of different materials.

1. Introduction

The analysis of the dynamic response of the crystal growth process is of interest in forming the basis for the study of the design of closed-loop control systems. Such systems for the control of crystal diameter have been implemented in a variety of ways. Commonly the heater power or the pull speed have been used as controlling variables [1,2]. In each case the closed-loop system is formed by deriving a signal representing the crystal diameter and subtracting this from a signal indicating the required diameter to form the diameter error signal. This error signal is then used to activate a controller which manipulates the controlling variable. The need to measure the crystal diameter has been met by techniques which use a laser beam to monitor the movement of the meniscus [2]. Alternatively an indirect indication has been obtained by using a load cell to measure the changing weight of either the crystal or the crucible [1].

In order to design the controller for such a system by systematic analysis it is necessary to define the dynamic response characteristics of the crystal growth process. For this purpose the most generally useful description of the process dynamics is obtained by identifying the transfer function relating the crystal diameter to the controlling variable. The transfer function is defined as the ratio of the Laplace Transforms of the output and input variables [3]. This function is derived from the differential equations of the process which must be such that the changes in the input and output variables are linearly related. For nonlinear processes a linear relationship can be usually obtained by restriction of the range of variation of the variables to small changes about the nominal operating values.

The following analysis shows how the transfer function relating the changes of crystal radius and the heater power may be derived. The analysis is based on approximations to the thermal transfer conditions in the system. Previous work in this field has generally been restricted to studies of the steady state conditions in the melt [4,5] and in the crystal [6–8]. It is necessary to extend this work to include transient conditions and such an extension is only possible if a number of simplifying assumptions are made about the otherwise intricate thermal transfer conditions. To this extent the results derived here represent a first approximation with a limited range of applicability. Nevertheless they form a useful basis for comparison of different growth conditions.

The problem is approached in three stages. In the first stage the relationship between the crystal radius and the melt temperature is examined. This is done in terms of the heat balance obtained in the zone close to

the growth interface. The second stage takes account to the effect of the crystal radius and the heater power on the melt temperature and in the final stage these results are combined to give an overall view of the process dynamics.

2. The influence of melt temperature on crystal radius

The heat balance conditions in the zone near the growth interface may be used to formulate the transfer function relating crystal radius error to the melt temperature. In this instance the melt temperature referred to is the value at the surface of the boundary layer below the solid/liquid interface.

The interface zone is shown diagramatically in fig. 1 where p_0 is the power transferred to the crystal, p_f is the power released due to fusion, p_i is the power transferred from the melt and p_r is the power lost from the meniscus surface. It will be assumed that the interface is flat and that the meniscus can be approximated by a cylinder. The heat balance requires that

$$p_i + p_f = p_0 + p_r \,. \qquad (1)$$

The determining features of each of the terms in this relationship are now considered. Linearised approximations will be obtained by defining the radius error*

$$a = r - r_0 \,, \qquad (2)$$

where r_0 is the nomial radius value, and assuming that a is small enough for terms in a^2 to be neglected.

2.1. Power transferred to the crystal

If G is the temperature gradient in the crystal close to the interface we have

$$p_0 = \pi r^2 k_s G \,, \qquad (3)$$

where k_s is the thermal conductivity of the solid. In order to determine how p_0 varies with the radius it is necessary to observe that G is in general a function of the radius. Evidence of this can be found from analysis of the steady state thermal field in the crystal.

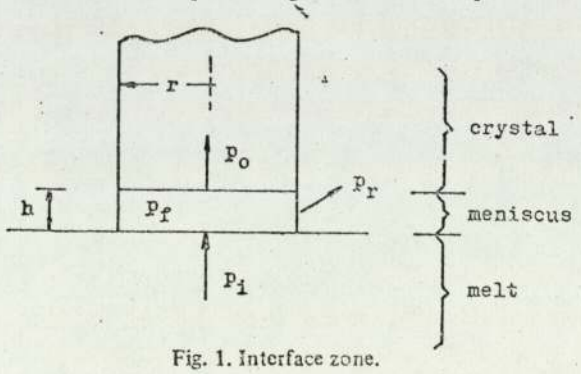


Fig. 1. Interface zone.

Transient changes in the radius will also set up transients in the thermal field. The fluctuations will propagate away from the interface zone with a delay corresponding approximately to a time constant $L^2\rho_s c/k_s$, where c is the specific heat and L is the distance from the interface. Materials of good conductivity, such as germanium and silicon, show high values of diffusivity $k_s/\rho_s c$ and the delay is then comparatively short; the rate of propagation of the thermal transients being large compared with the growth rate. In these cases we may assume that the changes of radius occur slowly in relation to the transients in the thermal field and the results of steady state analysis may be used.

Theoretical analyses of the temperature distribution in the crystal have shown that analytical results are available [6,7]. In terms of the numenclature used here, Kuo and Wilcox [7] derive the following expression for the temperature gradient on the axis at the interface

$$G = (\theta_a - \theta_f)/r\,(2B)^{1/2} \times \frac{(B_l/B)\cosh\lambda + (2/B)^{1/2}\sinh\lambda}{(B_l/B)\sinh\lambda + (2/B)^{1/2}\cosh\lambda} \,, \qquad (4)$$

where B is the Biot number $\epsilon r/k_s$ and λ is the dimensionless crystal length given by

$$\lambda = (L/r)\,(2B)^{1/2} \,. \qquad (5)$$

The more general results obtained by Brice [6] take account of the radial variation of the gradient. These show that the above expression is applicable to the gradient at the outer surface and is valid for materials of low Biot number. The range of validity has been given as $B < 0.2$ [7].

The value of B_l is determined by the heat transfer conditions at the top of the crystal. For a crystal which starts growth at the nominal radius r_0 and subsequently varies slowly from this value, the boundary conditions at the top correspond to writing $B_l = \epsilon r_0/k_s$, i.e.

* See section 7, list of symbols.

keeping B_l constant under changes in radius along the crystal.

Typical values of B for a semiconductor material of high conductivity such as germanium is 0.005, which is small enough for the following approximation to be used,

$$G = (\theta_a - \theta_i)\left[\frac{B_l}{r} + \frac{\lambda}{L}\tanh\lambda\right]. \tag{6}$$

For a short crystal, typically $L/r < 5$, we may approximate with $\tanh\lambda = \lambda$ and on differentiating eq. (6) we then have

$$\frac{dG}{dr} = (\theta_i - \theta_a)\left[\frac{B_l}{r^2} + \frac{\lambda^2}{Lr}\right], \tag{7}$$

which indicates the variation in G with radius changes. It is useful to write this in the form

$$\frac{dG}{dr}\frac{r}{G} = -1, \tag{8}$$

to show that the fractional change in G has the same magnitude as the fractional change in r and the opposite sign. Thus if G_0 is the nomial value of the gradient

$$G = G_0(1 - a/r_0) \tag{9}$$

which shows the gradient as a function of the radius deviation a.

The main assumptions used in arriving at this result are that the crystal length is short and that the radius is substantially constant throughout the length. Thus the approximation will be valid in the early stages of growth.

At the other extreme we may consider the effect of a change of radius near to the interface of a long crystal. The diagram of fig. 2 indicates a simplified view of this situation.

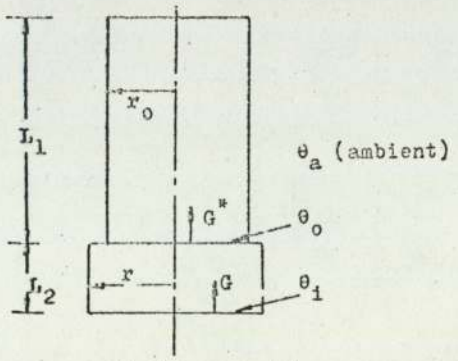


Fig. 2. Long crystal.

In the upper section the gradient G^* is given by an expression of the form shown in eq. (4) with the lower boundary temperature θ_i replaced by θ_0. The gradient G^* is therefore proportional to $(\theta_0 - \theta_a)$ and the change in G^* with θ_0 may be expressed as

$$\frac{dG^*}{d\theta_0}\frac{1}{G_0^*} = \frac{1}{\theta_0 - \theta_a}. \tag{10}$$

This show that if L_2 is short compared with L_1, so that $(\theta_0 - \theta_a)$ is large, the gradient G^* is insensitive to changes in θ_0 and therefore the total heat flow into the upper section remains substantially constant. Again for a material of low Biot number most of the heat is transferred by conduction rather than radiation in the lower section and therefore the total heat flow into the lower section is also constant. The gradient G in the lower section is then inversely proportional to r^2 and hence,

$$\frac{dG}{dr}\frac{r}{G} = -2, \tag{11}$$

which leads to,

$$G = G_0(1 - 2a/r_0). \tag{12}$$

If we compare this with eq. (9) it is appropriate to define

$$G = G_0(1 - \alpha a/r_0), \tag{13}$$

where α is a gradient variation factor, which varies between unity, in the initial stages of growth, and 2 in the final stages. It will be assumed that a value of 1.5 is representative of the general intermediate conditions.

When eq. (13) is used in conjunction with eq. (3) and terms in a^2 neglected we get,

$$p_0 = P_0(1 - \alpha a/r_0) + 2P_0 a/r_0, \tag{14}$$

where $P_0 = \pi r_0^2 k_s G_0$ is the nomial value of p_0.

2.2. Power due to fusion

Under steady state conditions the mass converted from the liquid to solid states is proportional to the interface area and the pull speed. But changes of radius are associated with changes in the meniscus height and there is a variation in the solidification rate with the movement of the interface relative to the melt surface. To examine this effect it is first nec-

essary to relate the changes of radius to the meniscus height.

2.2.1. Meniscus variations

The shape of the meniscus formed between a liquid surface and a cylindrical solid has been analysed with the assumption of zero contact angle [9]. It may be inferred from this that though the curvature of the meniscus is in general a function of the radius of the cylinder, it becomes substantially constant for large values of radius. The approximation to a constant curvature is valid when the radius exceeds $H/4$ where H is the capillary constant given by

$$H = (2\sigma/\rho g)^{1/2} , \tag{15}$$

where σ is the surface tension coefficients, ρ is the liquid density and g is the gravitational constant.

At large radius values the meniscus shape approaches that produced against a flat plate. The relationship between the height h and the surface angle may then be shown to be [10]

$$h = H[1 - \sin(\theta + \gamma)]^{1/2} , \tag{16}$$

where γ is the contact angle and θ is the surface angle as shown in fig. 3. Here H is seen to be the height at which the meniscus surface becomes vertical. When the solid surface is vertical $\theta = 0$ and the height is then given by

$$h_0 = H(1 - \sin\gamma)^{1/2} . \tag{17}$$

When angles θ and γ are small, eqs. (16) and (17) may be combined to give

$$\sin\theta = (2h_0/H^2)(h_0 - h) , \tag{18}$$

which expresses the slope of the solid surface in terms of the deviation of the meniscus height from the value at which parallel growth is maintained.

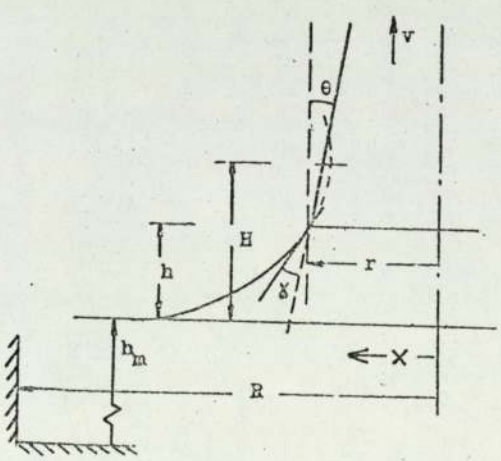


Fig. 3. Meniscus.

2.2.3. Movement of the interface

The diagram fig. 3 shows that the velocity of the interface relative to the crystal is $v - \dot{h} - \dot{h}_m$. The melt surface height h_m changes at the nominal rate

$$\dot{h}_m = -v_f/(1 - q) , \tag{19}$$

with

$$q = \frac{r_0^2 \rho_s}{R^2 \rho_m} . \tag{20}$$

From the diagram we also have

$$\dot{r} = (v - \dot{h} - \dot{h}_m)\tan\theta . \tag{21}$$

For small values of θ, $\tan\theta = \sin\theta$, and using eqs. (18), (19) and (21) with the restriction that $\dot{h}$ is small, we get

$$\dot{r} = KV(h_0 - h) , \tag{22}$$

in which V is the effective growth rate given by

$$V = v_f(1 - q) , \tag{23}$$

and

$$K = 2h_0/H^2 . \tag{24}$$

Eq. (22) represents the required general relationship between meniscus height and the crystal radius and is valid provided that the rate of change of radius is small compared with the growth rate.

2.2.4. Heat of fusion

The power released in the form of latent heat is given by $p_f = \pi r^2 \rho_s J(v - \dot{h} - \dot{h}_m)$ where J is the latent heat coefficient in J/g. On substituting for $\dot{h}$ in terms of $\ddot{r}$, obtained by differentiation of eq. (22), and approximating to small radius changes we get

$$p_f = P_f + \frac{2P_f}{r_0} a + \frac{\pi r_0^2 J \rho_s}{KV} \ddot{a} , \tag{25}$$

where P_f is the nomial value of the power released under steady state conditions.

2.3. Power transferred from the melt

The power transferred through the boundary layer is given approximately by,

$$p_i = (k_m/\delta)(\theta_m - \theta_i)\pi r^2 , \quad (26)$$

which shows the general dependence on the radius and the melt temperature. However the boundary layer thickness δ has been shown to be a function of the radius r and also to vary with the radial position under the interface [6]. The temperature at the boundary layer surface also varies. This interaction of effects complicates the interpretation of eq. (26).

Alternatively p_i can be expressed in terms of the temperature gradient in the boundary layer G_i, so that

$$p_i = \pi r^2 k_m G_i . \quad (27)$$

Experimental measurements of the temperature field in the melt have been made during the growth of silicon crystals over a wide range of conditions [4]. It is evident from these results that the gradient near the interface does not change significantly with changes of the crystal radius. Hence eq. (27) shows clearly the relationship between the power transferred and the radius while eq. (26) expresses the effect of changes in the melt temperature, in as much as these are due to factors other than radius changes; for example due to changes of the heater power input.

On combining these features and assuming small radius variations we get

$$p_i = P_i \left(1 + \frac{2a}{r_0} + \frac{\delta_m}{\theta_m - \theta_i}\right), \quad (28)$$

where δ_m is a small change in the melt temperature and P_i is the nomial value of the power transferred.

2.4. Power transferred from the meniscus

It is clearly difficult to be precise in evaluating the power radiated from the meniscus in view of the complex geometry involved. However it may be assumed that the total power is proportional to the surface area, which is in turn proportional to $2\pi rh$. The surface temperature is assumed to be constant at the fusion temperature.

The relationship between the radiated power p_r and the radius then follows as

$$p_r = P_r \left(1 + \frac{a}{r_0} - \frac{\dot{a}}{KVh_0}\right), \quad (29)$$

in which only terms corresponding to a first order variation in h and r have been included and eq. (22) is used to eliminate h. Here also P_r is the nomial value of the total power radiated.

To avoid the problem of computing P_r from the surface geometry, it is proposed to derive a value from the heat balance condition expressed in eq. (1) once the values of P_0, P_f and P_i have been established. This further ensures that the parameter values used in any derived results will be consistent with the steady state heat balance condition.

2.5. Heat balance equation

The heat balance condition has been formulated in eq. (1) and the separate terms evaluated in eqs. (14), (25), (28) and (29). When these expressions are substituted, the steady state values cancel to leave the incremental terms in the following differential equation,

$$\frac{-\delta_m P_i}{\theta_m - \theta_i} = \left(\frac{P_r + \alpha P_0}{r_0}\right)a + \frac{P_r}{KVh_0}\dot{a} + \frac{\pi r_0^2 J \rho_s}{KV}\ddot{a} . \quad (30)$$

This linear equation defines the mode of variation of the radius in response to changes in the melt temperature. The transfer function relating the two variables is then derived by taking the Laplace transform of the equation with zero initial conditions. The result may be written in the form

$$\bar{a}/\bar{\delta}_m = -D/(A + Bs + Cs^2) , \quad (31)$$

where $\bar{a}$ and $\bar{\delta}_m$ are transforms and the parameters are

$$A = (P_r + \alpha P_0)/r_0 , \quad B = P_r/KVh_0 , \quad (32)$$

$$C = \pi r_0^2 J \rho_s / KV , \quad D = P_i/(\theta_m - \theta_i) .$$

3. Effect of heater power changes

It is required to relate changes in the melt temper-

ature to changes in the heater power input. This relationship involves the heat transfer conditions in the melt.

The factors affecting the steady state temperature field in the melt have been examined [5], but due to the complex hydrodynamic effects induced by crucible and crystal rotation general analytical results are not available. Some useful experimental results have been obtained in silicon [4] and germanium [5] growth systems. For the present purposes an approximate view of the relationship between temperature and power flow may be developed as follows.

If p_h is the power input from the heater to the susceptor and θ_s is the susceptor temperature then

$$\theta_s - \theta_a = Z_1 p_h , \tag{33}$$

where Z_1 is the steady state value of the thermal impedance of heat transfer from the susceptor. In the absence of any heat loss from the surface of the melt the melt temperature would rise to equal the susceptor temperature in the steady state. If then power p_s is removed from the melt surface

$$\theta_s - \theta_m = Z_2 p_s , \tag{34}$$

where Z_2 is the steady state value of the thermal impedance presented to heat transfer through the melt. We note that in practice p_h is very much greater than p_s so that θ_s is independent of p_s.

Eliminating θ_s from eqs. (33) and (34) the relationship between small changes δ_h in p_h, δ_s in p_s and δ_m in θ_m is found to be

$$\delta_m = Z_1 \delta_h - Z_2 \delta_s . \tag{35}$$

This is the steady state relationship but changes of power do not have an immediate effect on the temperature; there is a transient delay due to thermal storage in the susceptor and the melt. If we define $f_1(s)$ to be the transfer function expressing the delay in response of the susceptor temperature to heater power changes and similarly $f_2(s)$ represents the delay between the melt temperature and the power loss from the surface, eq. (35) may be transformed to give

$$\bar{\delta}_m = \bar{\delta}_h Z_1 f_1(s) - \bar{\delta}_s Z_2 f_2(s) . \tag{36}$$

Now p_s consists of the power lost from the surface of the melt by radiation and that due to conduction to the crystal. Explicitly,

$$p_s = \pi R^2 \sigma_s + \pi r^2 (\sigma_i - \sigma_s) , \tag{37}$$

where σ_i and σ_s are the power transfer densities at the interface and the surface respectively, i.e.

$$\sigma_s = \epsilon_s (\theta_i - \theta_a), \quad \sigma_i = (k_m/a)(\theta_m - \theta_i) , \tag{38}$$

for small changes in the melt temperature and the radius this leads to

$$\bar{\delta}_s = \pi r_0^2 \frac{\sigma_i}{\theta_m - \theta_i} \bar{\delta}_m + 2\pi r_0 (\sigma_i - \sigma_s) \bar{a} . \tag{39}$$

Eliminating $\bar{\delta}_s$ between eqs. (36) and (39) and collecting terms in $\bar{\delta}_m$ we get an expression of the form

$$\bar{\delta}_m = F_1(s) \bar{\delta}_h - F_2(s) \bar{a} , \tag{40}$$

where

$$F_1(s) = \frac{\theta_s - \theta_a}{P_h} f_1(s) \left[1 + \frac{\mu Q f_2(s)}{(R/r_0)^2 + Q - 1} \right]^{-1} ,$$

$$F_2(s) = \frac{2(\theta_s - \theta_m)}{r_0} \left[\frac{(Q-1) f_2(s)}{(R/r_0)^2 + Q\,[1 + \mu f_2(s)] - 1} \right] ,$$

$$\mu = (\theta_s - \theta_m)/(\theta_m - \theta_i) ,$$

$$Q = \sigma_i/\sigma_s .$$

The value of the power transfer density ratio Q may be obtained by assuming that the emission at the meniscus is the same as over the rest of the melt surface so that

$$Q = 2h_0 P_i / r_0 P_r . \tag{41}$$

Eq. (40) now defines the required transfer functions relating the melt temperature to the heater power and the crystal radius.

4. Overall transfer function

The overall transfer function relating the radius to the heater power may be found by combining the results of sections 2 and 3. It is helpful to visualise the relationships in terms of the block diagram fig. 4 where F_1 and F_2 are as defined in eq. (40) and $-F_3$ is the function given in eq. (31). This block diagram clearly indicates the feedback effect by means of which the radius affects the melt temperature. Feedback occurs when the power density at the interface σ_i is different

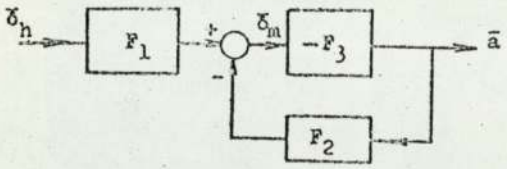


Fig. 4. Block diagram.

from that at the surface σ_s as expressed by the parameter Q. We note also that the sign of the feedback term depends on whether Q is greater or less than unity.

From fig. 4 the overall transfer function is seen to be

$$\bar{a}/\bar{\delta}_h = -F_1(s)\,F_3(s)/[1 - F_2(s)\,F_3(s)]\,. \qquad (42)$$

5. Transfer function under typical growth conditions

The form of the transfer function given by eq. (42) is complicated by the presence in F_1 and F_2 of terms f_1 and f_2 expressing the delay with which the melt temperature follows power changes. The function f_1 gives the delay between the heater power and the susceptor temperature and experimental results show that this involves time constants of one or two minutes in rf heated systems. The function f_2 relates the melt temperature to changes in the surface power loss and although this delay is difficult to measure experimentally it may be reliably taken to be less than in f_1. Under typical conditions the delay involved in the term F_3 is considerably greater than in either F_1 or F_2 and it is therefore proposed to examine the significance of the results on the assumption that the delays in F_1 and F_2 are negligible.

When this is done eq. (42) can be reduced to the form

$$\bar{a}/\bar{\delta}_h = \frac{-K_0(r_0/P_h)}{1 + 2\zeta\Omega^{-1}s + \Omega^{-2}s^2}\,, \qquad (43)$$

indicating a second order lag with natural frequence Ω and damping ratio ζ. The values of these parameters are given by

$$\Omega = \left[\frac{G_0 k_s K V \Phi}{r_0 J \rho_s}\right]^{1/2},$$

$$\zeta = \frac{G_i k_m}{Q}\,[r_0 J \rho_s G_0 k_s K V \Phi]^{-1/2}\,, \qquad (44)$$

with

$$\Phi = \alpha + 2\,\frac{G_i k_m}{G_0 k_s}\left[\frac{h_0}{Q r_0} - \frac{\mu(Q-1)}{(R/r_0)^2 + Q(\mu+1) - 1}\right].$$

The gain constant K_0 has been defined to express the ratio of the per-unit change in radius (a/r_0) to the per-unit change in heater power (δ_h/P_h) and takes the value

$$K_0 = \frac{G_i k_m}{G_0 k_s \Phi}\,\frac{\theta_s - \theta_a}{\theta_m - \theta_i}\left[1 + \frac{\mu Q}{(R/r_0)^2 + Q - 1}\right]^{-1}. \qquad (45)$$

These expressions are now in a form from which they may be evaluated using data obtainable from steady state measurements of the thermal field in the melt

Table 1
Typical parameter values

System constants		Ge	Si
r_0	crystal radius (cm)	1	1
R	crucible radius (cm)	2.3	4.6
V	growth rate (cm/min)	0.17	0.30
J	latent heat (J/g)	443	1800
k_s	conductivity (W/cm K)	0.24	0.30
k_m	conductivity (W/cm K)	0.71	0.67
ρ_s	density (g/cm^3)	5.32	2.34
γ	angle of contact (deg)	20	20
σ	surface tension (dyne/cm)	620	720
α	gradient variation factor	1.5	1.5
G_0	gradient in crystal (°C/cm)	103	110
G_i	gradient in melt (°C/cm)	30	35
θ_i	fusion temperature (°C)	937	1420
θ_m	melt temperature (°C)	945	1430
θ_s	susceptor temperature (°C)	960	1457
Derived parameters			
h_0	meniscus height (cm)	0.4	0.6
K	meniscus curvature constant (cm^{-1})	3.2	2.0
μ	differential temperature ratio	1.88	1.35
P_i	power input from melt (W)	67.0	73.7
P_0	power input to crystal (W)	77.7	103.7
P_f	power due to fusion (W)	21.5	66.2
P_r	power radiated (W)	10.8	36.2
Q	power density ratio	4.96	2.44
Φ	dimensionless parameter	0.95	1.74
Transfer function parameters			
Ω	natural frequency (rad/min)	0.58	0.70
ζ	damping ratio	0.19	0.20
K_0	gain constant	49	23

and the crystal. As an example, values for typical silicon and germanium growth systems have been computed and are shown in table 1.

The data for the silicon system were obtained from results given by Shashkov [4,8] and for germanium the results of Brice [6] were used.

The calculated values are generally in line with what has been observed in dynamic measurements made on growing crystals with the exception of K_0. The calculated values are approximately three times the observed values. This is attributable to the fact that in eq. (33) linearity has been assumed between the susceptor temperature and the heater power. In the presence of a substantial heat loss by radiation this is clearly an approximation which over estimates the sensitivity of the temperature to power changes. It is better to base K_0 on a measurement of the sensitivity and it can be shown that if ψ is the ratio of the susceptor temperature change to the per-unit change of heater power,

$$k_0 = \frac{G_i k_m}{G_0 k_s \Phi} \frac{\psi}{\theta_m - \theta_i} \left[1 + \frac{\mu Q}{(R/r_0)^2 + Q - 1}\right]^{-1}. \quad (46)$$

Values of ψ of the order of 500 C° have been observed in practice.

6. Conclusions

The results derived here represent a first approximation to the transfer function relating the crystal radius to changes of the heater power. They are valid provided that the changes of radius occur slowly and are small. Approximations to the changes in the thermal field in the crystal have been made which are adequate for materials such as germanium and silicon which have a low Biot number. This implies that the results are particular to materials showing high thermal conductivity.

With these limitations they offer a means of relating the dynamic response of a puller to the physical parameters of the material being grown. It has also been shown how data from steady state measurements on the temperature field in the melt and the crystal can be incorporated into the analysis.

Results of this type are important to the design of diameter control systems and the present work offers a means of making comparisons of the control system performance for different materials and system configurations. It is also possible to relate the required settings of controller parameters to the physical constants of the system and avoid the need for empirical adjustment.

Research is at present in progress leading to improved methods of measuring the dynamic response of crystal growth processes and in due course comparison between measured and theoretical results will be reported.

7. List of symbols

a	Crystal radius
f_1, f_2, F_1, F_2	Transfer function
G, G_i, G_0	Temperature gradients
h	Meniscus height
H	Capillary constant
J	Latent heat coefficient
k_s, k_m	Thermal conductivities
K	Meniscus curvature constant
K_0	Overall gain constant
$p_i, p_f, p_r, p_0, p_h, p_s$	Powers
P_i, P_f, P_r, P_0, P_h	Steady state powers
Q	Power transfer density ratio
r	Crystal radius
R	Crucible radius
s	Laplace transform variable
t	Time
v	Pull speed
V	Growth rate
α	Gradient variation factor
γ	Angle of contact
δ	Boundary layer thickness
δ_h, δ_s	Increments of p_h and p_s
δ_m	Increment of θ_m
ϵ, ϵ_s	Heat transfer coefficients
ζ	Damping ratio
$\theta_i, \theta_m, \theta_a, \theta_s$	Temperatures
μ	Differential temperature ratio
ρ_s, ρ_m	Densities
σ	Surface tension coefficient
σ_i, σ_s	Power transfer densities
Φ	Dimensionless parameter
ψ	Gain constant of susceptor
Ω	Natural frequency

Acknowledgements

The authors acknowledge with gratitude the support of the staff of Metals Research Ltd. on behalf of whom this work has been undertaken.

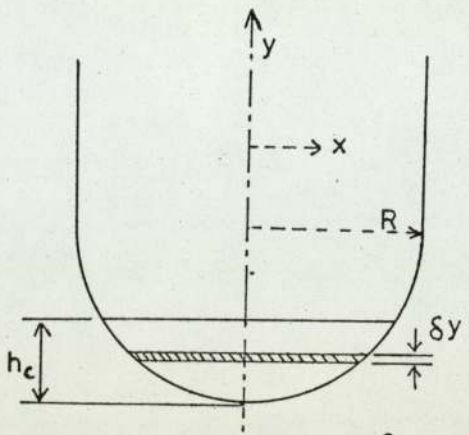
References

[1] A.E. Zinnes, B.E. Nevis and C.D. Brandle, J. Crystal Growth 19 (1973) 187.
[2] U. Gross and R. Kersten, J. Crystal Grwoth 15 (1972) 85.
[3] J.J. D'Azzo and C.H. Houpis, Feedback Control System Analysis and Synthesis (McGraw-Hill, New York, 1960).
[4] Y.M. Shashkov and V.M. Gurevich, Dokl. Akad. Nauk SSSR 187 (1969) 146.
[5] N. Kobayashi and T. Arizumi, Japan. J. Appl. Phys. 9 (1970) 1255.
[6] J.C. Brice, J. Crystal Growth 2 (1968) 395.
[7] V.H.S. Kuo and W.R. Wilcox, J. Crystal Growth 12 (1972) 191.
[8] Y.M. Shaskov and V.P. Grishin, Dokl. Akad. Nauk SSSR 179 (1968) 404.
[9] G.K. Gaule and J.R. Pastore, Met. Soc. Conf. (Interscience) 12 (1961) 201–225.
[10] H.M. Princen, in: Surface and Colloidal Science, Vol. 2, Ed. Matijevic (Wiley, New York, 1969).

APPENDIX II

MELT LEVEL IN SPHERICAL CRUCIBLES

It is required to find the depth of liquid that occupies a given volume in the bottom of a spherical bottomed crucible



volume of elemental slice = $\delta y \pi x^2$ A201

∴ total volume in depth h_c, V_s is found by integrating equation A201

$$V_s = \int_{R-h_c}^{R} \pi x^2 dy \quad \text{..... A202}$$

but since $x^2 = R^2 - y^2$ A203

$$V_s = \pi \left[R^2 y - \frac{y^3}{3} \right]_{R-h_c}^{R} \quad \text{..... A204}$$

$$V_s = \pi R h_c^2 - \frac{\pi h_c^3}{3} \quad \text{..... A205}$$

re-arranging A205

$$h_c^3 - 3rh_c^2 + \frac{3V_s}{\pi} = 0 \quad \text{..... A206}$$

which may be solved for h_c as a cubic.

APPENDIX 3

COMPUTER PROGRAM LISTINGS

A3.1 DPPP

DPP

```
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      N=-1
5     WRITE(1,13)
13    FORMAT(" DATA CHAN FOR INPUT?")
      READ(2,)NC
      WRITE(1,3)
3     FORMAT(" FILE LENGTH?")
      READ(2,)LF
      IF(NC.GT.0)GO TO 17
      DO 16 J=1,LF
16    DATA(J)=0.0
      IR=LF
      GO TO 11
17    IF(NC.LT.6) GO TO 15
14    WRITE(1,1)
1     FORMAT(" DATA INPUT FILENAME?")
      READ(2,2)FNAM
2     FORMAT(A5,A4)
      CALL FSTAT(NC,FNAM,J)
      IF(J.EQ.0) GO TO 14
      CALL SEEK(NC,FNAM)
15    READ(NC,)(DATA(J),J=1,LF)
      CALL CLOSE(NC)
6     WRITE(1,4)
4     FORMAT(" DATA FROM,TO?")
      READ(2,)ISTRT,IFIN
      IR=IFIN-ISTRT+1
      IF(IR.GT.LF) GO TO 6
      DO 7 J=1,IR
7     DATA(J)=DATA(ISTRT+J-1)
11    WRITE(1,8)
8     FORMAT(" SPECIFY PROGRAM")
      IF(ITOG(32)) GO TO 9
      WRITE(1,10)
10    FORMAT(" 1",T6,"ADD POLLY"/" 2",T6,"IMPLEMENT DIGIFILTER"/" 3"
      1"DESIGN DIGITAL FILTER"/" 4",T6,"FIT POLLY"/" 5",T6,"ENTER
      2COEFFS"/" 6",T6,"TIME LINEARISATION"/" 7",T6,"OUTPUT"/" 8",T
      3"NEW DATA"/" 9",T6,"TRANSFORM DIGIFILTER"//
      3" FUNCTIONS OF ACS ARE AS FOLLOWS:"//
      4" 12",T6,"SUPPRESS THIS PRINTOUT"/" 13",T6,"SELECT DATA
      5 OUTPUT"/" 14",T6,"RE READ ACS 13,15,16"/" 15",T6,"SELECT
      6 GRAPH PLOTTER"/" 16",T6,"SELECT GRAPHICS OUTPUT"/" 17",T6,
      7"TERMINATE GRAPHICS OUTPUT"/)
9     READ(2,)ISW
      IF(ISW.EQ.1) CALL DRITF
      IF(ISW.EQ.2) CALL FILTUR
      IF(ISW.EQ.3) CALL FILD
      IF(ISW.EQ.4) CALL POLLY
      IF(ISW.EQ.5) CALL ENTRER
      IF(ISW.EQ.6) CALL DIMOD3
      IF(ISW.EQ.7) CALL PUTOUT
      IF(ISW.EQ.8)GO TO 5
      IF(ISW.EQ.9) CALL TRANS
      GO TO 11
      END
```

DRITF

```
      SUBROUTINE DRITF
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      IF(N.GE.1) GO TO 3
      WRITE(1,4)
 4    FORMAT(" NO COEFFICIENTS PRESENT")
      RETURN
 3    WRITE(1,5)
 5    FORMAT(" ADD (+1) OR SUBTRACT (-1)")
      READ(1,)ISN
      DO 1 J=1,IR
      FJ=FLOAT(J)
      A1=0.0
      DO 2 K=1,N
      M2=N-K
 2    A1=A1+COEFF(K)*FJ**M2
 1    DATA(J)=DATA(J)+ISN*A1
      RETURN
      END
```

FILD

```
C PROGRAM TO DESIGN OFF LINE FINITE IMPULSE RESPONSE
C BRICKWALL LOWPASS DIGITAL FILTERS BY THE WINDOW METHOD
      SUBROUTINE FILD
      DOUBLE PRECISION ACC
      REAL IDEALR(100),IDEALI(100)
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
 14   WRITE(1,1)
 1    FORMAT(" CLOCK FREQ (HZ)?")
      READ(1,)CF
      WRITE(1,2)
 2    FORMAT(" NUMBER OF POINTS?")
      READ(1,)NP
 33   WRITE(1,3)
 3    FORMAT(" CUTOFF FREQ (HZ)?")
      READ(1,)FC
C PRODUCE BRICKWALL LOWPASS FREQ RESPONSE
      DO 4 J=1,NP
      IDEALR(J)=0.0
 4    IDEALI(J)=0.0
      CN=FC/CF*NP
      NC=CN+1
      IF(NP-NC+2.LT.1) GO TO 33
      IDEALR(1)=1.0
      DO 5 J=2,NC
      IDEALR(J)=1.0
 5    IDEALR(NP-J+2)=1.0
C CALC IDEAL IMPULSE RESPONSE
      CALL FFT1A(IDEALR,IDEALI,NP,NP,NP,1)
      DO 40 J=1,NP
      IDEALR(J)=IDEALR(J)/NP
 40   IDEALI(J)=IDEALI(J)/NP
      CALL GRUPAR(IDEALR,IDEALI,COEFF,NP,1)
C WINDOW IDEAL IMPULSE RESPONSE
      WRITE(1,6)
 6    FORMAT(" WINDOW TYPE:-RECTANGULAR(1),HAMMING(0),BLACKMAN(-1
      READ(1,)IWIND
      WRITE(1,7)
 7    FORMAT(" LENGTH OF WINDOW?")
      READ(1,)L
      CALL WINDOW(NP,L,COEFF,IWIND)
```

FILD (continued)

```
C SCALE TO GIVE GAIN OF 1 IN PASSBAND
      A=0.0
      DO 11 J=1,NP,2
 11   A=A+COEFF(J)
      DO 12 J=1,NP
 12   COEFF(J)=COEFF(J)/A
      CALL GRUPAR(IDEALR,IDEALI,COEFF,NP,0)
C STORE COEFFS IN ARRAY COEFF
      CALL FFT1A(IDEALR,IDEALI,NP,NP,NP,-1)
      N1=NP/2
C CALC DB,DEG OF WINDOWED SEQUENCE
      CALL RIMP(IDEALR,IDEALI,N1,1)
C CALC FREQ ARRAY.
      N1=N1-1
      DO 8 J=1,N1
      F(J)=FLOAT(J)*CF/NP
      IDEALR(J)=IDEALR(J+1)
 8    IDEALI(J)=IDEALI(J+1)
      WRITE(1,9)
 9    FORMAT(" GRAPH OF GAIN (DB)")
      CALL PLTFRG(IDEALR,F,N1,0,AL)
      WRITE(1,10)
 10   FORMAT(" GRAPH OF PHASE (DEG)")
      CALL PLTFRG(IDEALI,F,N1,0,AL)
C ROTATE FILTER COEFFICIENTS AND STORE IN ARRAY COEFF
      LL=L-1
      N=LL
      L2=NP-L/2+1
      DO 20 J=1,LL
      L3=L2+J
      IF(L3.GT.NP) L3=L3-NP
 20   IDEALR(J)=COEFF(L3*2-1)
      DO 27 J=1,LL
 27   COEFF(J)=IDEALR(J)
      WRITE(1,26)
 26   FORMAT(" PAPER TAPE OF COEFFS?")
      READ(1,)J
      IF(J.NE.1) GO TO 21
      WRITE(4,21)(IDEALR(J),J=1,LL)
      CALL CLOSE (4)
C OUTPUT FILTER COEFFS ON PP
 21   FORMAT(1X,1PE15.7)
      RETURN
      END
```

POLLY

```
C SUBROUTINE TO FIT POLYNOMIALS OF DEG N TO DATA
      SUBROUTINE POLLY
      REAL A(132)
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      WRITE(1,2)
 2    FORMAT(" ORDER OF POLY?")
      READ(1,)N
      IF(N.LT.0) RETURN
      DO 6 J=1,IR
 6    F(J)=J
      CALL STPLRG(IR,F,DATA,N,COEFF,IFLAG,A)
      IF(IFLAG.NE.1) WRITE(1,3)
 3    FORMAT(" POLY NOT FOUND")
      N=N+1
      WRITE(1,)(COEFF(J),J=1,N)
      RETURN
      END
```

DIMOD3

```
C SUBROUTINE TO MODIFY N EQUI-SPACED OBSERVATIONS TO M EQUI
C TIME SPACED OBSERVATIONS (RE GAP XTALS)
      SUBROUTINE DIMOD3
      REAL X1(1300)
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      ROM=4.5
      ROS=4.1
      DCRU2=9.0*ROM/ROS
12    IF(ITOG(16))RETURN
      WRITE(1,1)
1     FORMAT(" MENISCUS CONST? SLICE THICKNESS (IN)?")
      READ(1,)FK,XINC
      WRITE(1,5)
5     FORMAT(" SAMPLE DATA INPUT EVERY?")
      READ(2,)FKS
      KS=FKS
11    WRITE(1,6)
6     FORMAT(" ENTER 2ND DERIVATIVE FILENAME")
      READ(2,7)FNAM
7     FORMAT(A5,A4)
      WRITE(1,10)
10    FORMAT(" DATA CHANNEL? LENGTH? FROM? TO?")
      READ(2,)ICHAN,L,ISTRT,IFIN
      IR1=IFIN-ISTRT+1
      IF(IR1.NE.IR) GO TO 11
      CALL FSTAT(ICHAN,FNAM,J)
      IF(J.EQ.0) GO TO 11
      CALL SEEK(ICHAN,FNAM)
      READ(ICHAN,)(X1(J),J=1,L)
      CALL CLOSE(ICHAN)
      KL=IFIX(FLOAT(IR)/FKS)
      DO 4 J=1,IR
      X1(J)=X1(J+ISTRT-1)/(XINC)
4     F(J)=J
      XINC=XINC*FKS
      FK2=.5/FK
C CALC MODIFIED SLICE THICKNESS
      FJ=2.0
14    FJ=FJ+FKS
      IF(IFIX(FJ).GE.KL) GO TO 2
      J=IFIX(FJ)
      I1=(J-2)*FKS+FKS/2
      EXT=XINC-FK2*X1(I1)
      X1(J-1)=(1.0-(DATA(I1)**2)/DCRU2)*EXT
      GO TO 14
2     CONTINUE
      WRITE(1,8)
8     FORMAT(" GRAPH OF SLICE THICKNESS")
      IF(ITOG(2))CALL PICTUR(F,X1,KL-1,1)
      CALL PLOT3(X1,KL-1,1)
C INTEGRATE SLICE THICKNESS
      DO 3 J=3,KL
3     X1(J-1)=X1(J-2)+X1(J-1)
      WRITE(1,9)
9     FORMAT(" GRAPH OF INTEGRATED SLICE THICKNESS")
      IF(ITOG(2))CALL PICTUR(F,X1,KL-1,1)
      CALL PLOT3(X1,KL-1,1)
C FI=EQUI-TIME INTERVAL ON X1
```

DIMOD3 (continued)

```
      FI=X1(KL-1)/(KL-1)
C LIN INTERPOLATE DATA FOR EQUI-TIME
      K=1
      J=2
 16   FIJ=FI*FLOAT(J-1)
      IF(FIJ.LE.X1(K)) GO TO 17
      K=K+1
      IF(K-KL)16,18,18
 18   K=1
      F(J)=0.0
      J=J+1
      IF(J-KL)16,16,13
 17   T=(FIJ-X1(K-1))/(X1(K)-X1(K-1))
      TKS=T*FLOAT(KS)
      ITKS=TKS
      TKS1=TKS-ITKS
      F(J)=DATA(K*KS+ITKS)+TKS1*(DATA(K*KS+ITKS+1)-DATA(K*KS+ITKS)
      J=J+1
      IF(J.LE.KL) GO TO 16
 13   DO 19 J=2,KL
 19   DATA(J)=F(J)
      IR=KL
      LENGTH=KL
      WRITE(1,)LENGTH
      RETURN
      END
```

PUTOUT

```
      SUBROUTINE PUTOUT
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      DO 6 J=1,IR
 6    F(J)=J
 10   IF(ITOG(2))CALL PICTUR(F,DATA,IR,1)
      IF(ITOG(4))CALL PLOT3(DATA,IR,1)
      IF(ITOG(8))GO TO 10
      IF(ITOG(16)) GO TO 7
      RETURN
 7    WRITE(1,1)
 1    FORMAT(" DATCHAN FOR OUTPUT?")
      READ(2,)IK
      IF(IK.LT.6) GO TO 5
      WRITE(1,2)
 2    FORMAT(" FILENAME?")
      READ(2,3)FNAM
 3    FORMAT(A5,A4)
      CALL ENTER(IK,FNAM)
 5    WRITE(IK,4)(DATA(J),J=1,IR)
      IF(IK.GT.3) CALL CLOSE(IK)
 4    FORMAT(1X,1PE15.7)
      IF(ITOG(8))GO TO 10
      RETURN
      END
```

FILTUR

```
      SUBROUTINE FILTUR
      DOUBLE PRECISION ACC
      REAL IDEALR(100),IDEALI(100)
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      IF(N.GE.0) GO TO 1
      WRITE(1,2)
 2    FORMAT(" NO COEFFICIENTS PRESENT")
      RETURN
 1    LL=N
      L=N+1
      L4=L/2-1
C INITIALISE DATA STORE
      L5=L/2
      DO 9 J2=1,LL
      K1=IR+L4-J2+1
      IF(K1.GT.IR) GO TO 10
      IDEALI(J2)=DATA(K1)
      GO TO 9
 10   IDEALI(J2)=0.0
 9    CONTINUE
      DO 19 J1=1,IR
      J=IR-J1+1
      J2=J-L5
      ACC=0.0
C IMPLEMENT DIGIFILTER
      DO 18 K=1,LL
 18   ACC=ACC+IDEALI(K)*COEFF(K)
      DATA(J)=ACC
C ROTATE TEMP DATA STORE
      DO 23 J0=2,LL
 23   IDEALI(J0-1)=IDEALI(J0)
      IF(J2.LT.1) GO TO 24
      IDEALI(LL)=DATA(J2)
      GO TO 19
 24   IDEALI(LL)=0.0
 19   CONTINUE
C DECIMATE FILTERED DATA
      WRITE(1,22)
 22   FORMAT(" SAMPLE OUTPUT EVERY ?")
      READ(1,)NS
      LENGTH=IR/NS
      DO 25 J=1,LENGTH
 25   DATA(J)=DATA(J*NS)
      WRITE(1,)LENGTH
      IR=LENGTH
      RETURN
      END
```

ENTRER

```
      SUBROUTINE ENTRER
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      WRITE(1,1)
 1    FORMAT(" NUMBER OF COEFFICIENTS?")
      READ(1,)N
      WRITE(1,3)
 3    FORMAT(" INPUT COEFFS ON PAPER READER (3), OR TELETYPE (2)"
      READ(2,)IN
      WRITE(1,2)
 2    FORMAT(" ENTER COEFFICIENTS IN DESCENDING ORDER")
      READ(IN,)(COEFF(J),J=1,N)
      RETURN
      END
```

GRUPAR

```
      SUBROUTINE GRUPAR(REAL,IMAG,COMPLX,N,IFLAG)
C SUBROUTINE TO CONVERT TWO ARRAYS INTO ONE COMPLEX ARRAY OR TO
C CONVERT ONE COMPLEX ARRAY INTO TWO ARRAYS.
C  REAL,IMAG=TWO ARRAYS
C  COMPLX=COMPLEX ARRAY
C  N=LENGTH OF EACH OF TWO ARRAYS.
C  IFLAG=0 FOR 2-1
C  IFLAG=1 FOR 1-2
      REAL REAL(1),IMAG(1),COMPLX(1)
      IF(IFLAG)2,3,2
 2    DO 1 J=1,N
      COMPLX(J*2-1)=REAL(J)
 1    COMPLX(J*2)=IMAG(J)
      RETURN
 3    DO 4 J=1,N
      REAL(J)=COMPLX(J*2-1)
 4    IMAG(J)=COMPLX(J*2)
      RETURN
      END
```

WINDOW

```
      SUBROUTINE WINDOW(N,L,H,IWIND)
C
C SUBROUTINE APPLIES A WINDOW OF TOTAL LENGTH L TO A COMPLEX
C ARRAY
C  INPUTS ARE:   N=LENGTH OF INPUT/OUTPUT ARRAY.
C                IWIND=-1 FOR BLACKMAN WINDOW
C                     =0 FOR HAMMING WINDOW
C                     =1 FOR RECTANGULAR WINDOW
C                L=LENGTH OF WINDOW REQUIRED.
C                H=ARRAY OF DATA TO BE WINDOWED.
C
C  OUTPUT RETURNED IN ARRAY H
C
      REAL H(1)
      L2=L/2
      IF(IWIND)1,1,6
 1    DO 5 I=2,L2
      A=8.0*ATAN(1.0)*FLOAT(I-1)/FLOAT(L)
      IF(IWIND)3,2,2
 2    W=.54+.46*COS(A)
      GO TO 4
 3    W=.42+.5*COS(A)+.08*COS(2.0*A)
 4    H(I*2-1)=H(I*2-1)*W
      H(I*2)=0.0
      J=N-I+2
      H(J*2-1)=H(J*2-1)*W
 5    H(J*2)=0.0
 6    L3=L2+1
      N3=N-L/2+1
      DO 7 I=L3,N3
      H(I*2-1)=0.0
 7    H(I*2)=0.0
      RETURN
      END
```

TRANS

```
      SUBROUTINE TRANS
C TRANSFORMS LOWPASS DIGIFILTER PROTOTYPES TO HIGHPASS,BANDPASS
C OR BANDSTOP. FOR FINITE IMPULSE RESPONSE FILTERS.
      COMMON DATA(1300),F(1300),COEFF(200),N,IR,FNAM(2)
      IF(N.GE.1) GO TO 2
      WRITE(1,3)
 3    FORMAT(" NO COEFFICIENTS PRESENT")
      RETURN
 2    WRITE(1,1)
 1    FORMAT(" TRANSFORM LOWPASS PROTOTYPE DIGIFILTER INTO:"/
     1" 1",T6,"HIGHPASS"/" 0",T6,"BANDPASS"/" -1",T6,"BANDSTOP"/)
      READ(2,)IT
      DO 5 J=1,N
 5    F(J)=COEFF(J)
      IF(IT.LT.1)N=N*2-1
      INC=2
      IF(IT.EQ.1)INC=1
      DO 6 J=1,N,INC
      J1=J
      IF(INC.EQ.2)J1=(J+1)/2
      ISN=1
      IF(IT.GE.0)ISN=(-1.0)**J1
      COEFF(J)=ISN*F(J1)
      IF(IT.EQ.1) GO TO 6
      COEFF(J+1)=0.0
 6    CONTINUE
      RETURN
      END
```

A3·2 SPECTA

GHOST

```
C PROGRAM TO PRODUCE SPECTRUM OF SYSTEM FROM INPUT AND OUTPUT
C I/O DATA FROM FILES ON DATSLOT 7/6
C CALLS SUBROUTINES BACH,(FFT1A,HANN),SEGMEN,OUTPUT,(PICTUR,(GRAFIX,
C ITOG),PLTFRQ
      INTEGER OSTRT
      REAL ALAB(9),BLAB(9),CLB(9),GL(7),
     1D(9),E(9),F(508),FTEMP(508),FTI(254),FTO(254)
      COMMON TEMPI(254),TEMPO(254),XIN(508),XOUT(508),FILEN(2)
       DATA ALAB(1),ALAB(2)/5HFREQ ,5H(HZ) /
      DATA ALAB(4),ALAB(5)/5HMOD (,5HDB)  /
      DATA ALAB(7),ALAB(8),ALAB(9)/5HFREQ ,5HRESPO,5HNSE  /
      DATA BLAB(1),BLAB(2)/5HFREQ ,5H(HZ) /
      DATA BLAB(4),BLAB(5)/5HPHI (,5HDEG) /
      DATA BLAB(7),BLAB(8),BLAB(9)/5HFREQ ,5HRESPO,5HNSE   /
      DATA CLB(1),CLB(2)/5HFREQ ,5H(HZ) /
      DATA CLB(4),CLB(5),CLB(6)/5HGAMMA,5H SQUA,5HRED  /
      DATA CLB(7),CLB(8),CLB(9)/5HCOHER,5HENCE ,5HFUNCT/
      DATA D(1),D(2)/5HFREQ ,5H(HZ) /
      DATA D(4),D(5)/5HMOD (,5HDB)  /
      DATA D(7),D(8),D(9)/5HPOWER,5H SPEC,5HTRUM /
      DATA E(1),E(2)/5HFREQ ,5H(HZ) /
      DATA E(4),E(5)/5HPHI (,5HDEG) /
      DATA E(7),E(8),E(9)/5HPOWER,5H SPEC,5HTRUM /
      DATA GL(1),GL(2),GL(3),GL(4)/5HGXX  ,5HGYY  ,5HGYXPH,5HGYXDE
       DATA GL(5),GL(6),GL(7)/5HGAMSQ,5HMOD H,5HPHI H/
      RAD=45.0/ATAN2(1.0,1.0)
      PI=180.0/RAD
 9    DO 6 J=1,508
      XIN(J)=0.0
 6    XOUT(J)=0.0
      WRITE(1,1)
      READ(1,)IN
      READ(2,2)FILEN
      WRITE(1,10)
      READ(1,)X1
 31   WRITE(1,3)
      READ(1,)LIN
 44   WRITE(1,26)
      READ(1,)IL,ISTRT
      IF(IL.GT.LIN) GO TO 44
      J=1
      IF(IN.GT.5) CALL FSTAT(IN,FILEN,J)
      IF(J.EQ.0) GO TO 6
      CALL SEEK(IN,FILEN)
      READ(IN,)(XIN(J),J=1,LIN)
      CALL CLOSE(IN)
      WRITE(1,17)
      READ(1,)ID2
      IF(ID2-1)9,18,21
 18   WRITE(1,4)
      READ(2,2)FILEN
 34   WRITE(1,5)
      READ(1,)LOUT
      J=1
      IF(IN.GT.5) CALL FSTAT(IN,FILEN,J)
      IF(J.EQ.0) GO TO 18
```

```
      CALL SEEK(IN,FILEN)
      READ(IN,)(XOUT(J),J=1,LOUT)
      CALL CLOSE(IN)
43    WRITE(1,23)
      READ(1,)OSTRT
21    WRITE(1,12)
      READ(1,)INC
      CALL BACH(FTI,FTO,IL,INC,F,FTEMP,OSTRT,ISTRT,LIN,LOUT)
      IL2=(IL+1)/2
      IL4=IL2/INC
C CALC GXX(DB),GYY(DB) (ARRAYS TEMPI,TEMPO), AND GYX(DB),
C GYX(DEG)(ARRAYS FTI,FTO). (SUBROUTINE SEGMEN USED FOR CHAIN)
      CALL SEGMEN(TEMPI,TEMPO,IL,X1,FTO,FTI,F,INC)
C OUTPUT GXX,GYY
      CALL OUTPUT(TEMPI,TEMPO,F,IL4,GL(1),D)
      IF(ID2.EQ.2) GO TO 9
      CALL OUTPUT(TEMPO,TEMPI,F,IL4,GL(2),D)
C OUTPUT GYXRE,GYXIM (DB,DEG)
29    CALL OUTPUT(FTI,FTO,F,IL4,GL(4),D)
      CALL OUTPUT(FTO,FTI,F,IL4,GL(3),E)
C CALC MOD(H) AND OUTPUT TOGETHER WITH PHASE
      WRITE(1,25)
      READ(1,)KU
      DO 32 J=2,IL2,INC
      IX=J/INC+INC-1
      FTEMP(IX)=FTI(IX)-TEMPI(IX)
      IF(KU.EQ.2) GO TO 32
      A1=PI*FLOAT(IX)/FLOAT(IL)
      FTEMP(IX)=FTEMP(IX)+20.0*ALOG10(A1/SIN(A1))
      FTO(IX)=FTO(IX)+RAD*A1
32    CONTINUE
      FTEMP(1)=FTEMP(2)
      CALL OUTPUT(FTEMP,FTO,F,IL4,GL(6),ALAB)
      CALL OUTPUT(FTO,FTEMP,F,IL4,GL(7),BLAB)
C CALC GAMSQ AND OUTPUT
      DO 33 J=2,IL2,INC
      IX=J/INC+INC-1
      A1=.1*(2.0*FTI(IX)-TEMPI(IX)-TEMPO(IX))
      IF(A1.LE.-3.0) GO TO 49
      FTEMP(IX)=10.0**(A1)
      GO TO 33
49    FTEMP(IX)=0.0
33    CONTINUE
      FTEMP(1)=FTEMP(2)
      CALL OUTPUT(FTEMP,XOUT,F,IL4,GL(5),CLB)
      IF(ITOG(64)) 42,9,42
42    IF(ID2-1) 9,43,9
1     FORMAT(" INPUT CHANNEL?,INPUT FILENAME?")
2     FORMAT(A5,A4)
10    FORMAT(" SAMPLE T?")
3     FORMAT(" TOTAL INPUT LENGTH?")
26    FORMAT(" INPUT DATA BLOCK LENGTH? STARTING AT?")
17    FORMAT(" WANT 2ND DATA FILE?")
4     FORMAT(" SYSTEM OUTPUT FILENAME?")
5     FORMAT(" TOTAL OUTPUT LENGTH?")
23    FORMAT(" OUTPUT DATA STARTING AT?")
25    FORMAT(" ZOH CORRECTION ON TF?")
12    FORMAT(" ALL? (1); ODD ONLY? (2) HARMONICS")
      END
```

SEGMEN

```
C SEGMENT OF GHOST FOR CHAIN SEGMENTATION PURPOSES.
      SUBROUTINE SEGMEN(TEMPI,TEMPO,IL,X1,FTO,FTI,F,INC)
      REAL TEMPI(1),TEMPO(1),FTO(1),FTI(1),F(1)
      RAD=45.0/ATAN2(1.0,1.0)
      IL2=(IL+1)/2
      CALL ERRSET(0)
      DO 28 J=2,IL2,INC
      IX=J/INC+INC-1
      IF(TEMPI(J).LE.0.0) GO TO 35
      TEMPI(IX)=10.0*ALOG10(TEMPI(J))
      GO TO 36
35    TEMPI(IX)=-200.0
36    IF(TEMPO(J).LE.0.0) GO TO 37
      TEMPO(IX)=10.0*ALOG10(TEMPO(J))
      GO TO 47
37    TEMPO(IX)=-200.0
47    F(IX)=(J-1)/(X1*IL)
      A1=FTI(J)
      IF(A1.EQ.0.0.AND.FTO(J).EQ.0.0) GO TO 46
      FTI(IX)=5.0*ALOG10(A1*A1+FTO(J)*FTO(J))
      GO TO 28
46    FTI(IX)=-200.0
28    FTO(IX)=RAD*ATAN2(FTO(J),A1)
      CALL ERRSET(1)
      F(1)=F(2)
C REMOVE PHASE DISCONTINUITIES
2     WRITE(1,1)
1     FORMAT(" PHASE DISCONTINUITY DETECTION THRESHOLD?")
      READ(2,)FDDT
13    KU=0
      KFLAG=0
      IL4=IL2/INC
      N1=IL4-1
      DO 11 J=2,N1
      IF(FTO(J+1)-FTO(J).GT.FDDT) KU=-1
      IF(FTO(J+1)-FTO(J).LT.-FDDT) KU=1
      IF(KU.NE.0) KFLAG=J+1
      IF(KFLAG.NE.0) GO TO 16
11    CONTINUE
      GO TO 19
16    DO 20 J=KFLAG,IL4
20    FTO(J)=FTO(J)+KU*360.0
C INFINITE LOOP ESCAPE
      IF(ITOG(128))GO TO 19
      GO TO 13
19    CALL PICTUR(F,FTO,IL4,1)
      IF(ITOG(128)) GO TO 2
      TEMPI(1)=TEMPI(2)
      TEMPO(1)=TEMPO(2)
      FTI(1)=FTI(2)
      FTO(1)=FTO(2)
      RETURN
      END
```

HANN

```
      SUBROUTINE HANN(A,K,PCT)
      REAL A(1)
      PI=4.0*ATAN2(1.0,1.0)
      LPCT1=IFIX((PCT*K)/200.0)-1
      F=PI/LPCT1
      DO 1 J=0,LPCT1
      PI=.5*(1.0-COS(F*J))
      A(J+1)=A(J+1)*PI
1     A(K-J)=A(K-J)*PI
      RETURN
      END
```

BACH

```
      SUBROUTINE BACH(GYXRE,GYXIM,KM,IK,FFR,FFI,IOSTRT,ISTRT,LIN
     ,LOUT)
C SUBROUTINE TO CALC 1) POWER SPECTRUM OF INPUT
C                    2) POWER SPECTRUM OF OUTPUT
C                    3) CROSS POWER SPECTRUM
C   U=ARRAY FOR SYSTEM INPUT
C   V=ARRAY FOR SYSTEM OUTPUT
C
C   GYXRE=ARRAY TO RETURN REAL CROSSSPECTRUM
C   GYXIM=ARRAY TO RETURN IMAG CROSSSPECTRUM
C GXX=ARRAY TO RETURN INPUT POWER SPECTRUM
C GYY=ARRAY TO RETURN OUTPUT POWER SPECTRUM
C FFR=ARRAY FOR INTERNAL USE
C FFI=ARRAY FOR INTERNAL USE
C   KM=LENGTH OF DATA
C   IK=1 FOR ALL HARMONICS
C   IK=2 FOR ODD HARMONICS ONLY
C   IOSTRT=ADDRESS OF V AT WHICH DATA STREAM STARTS.
C   ISTRT=ADDRESS OF U AT WHICH DATA STREAM STARTS.
C   LIN=TOTAL AVAILABLE U DATA.
C   LOUT=TOTAL AVAILABLE V DATA.
C   SEGMENTATION,SEGMENT AVERAGING,HANNING ALL OPTIONIALLY DONE
      REAL FFR(1),FFI(1),GYXRE(1),GYXIM(1)
      COMMON GXX(254),GYY(254),U(508),V(508)
      KM2=KM/2
      DO 6 J=1,KM2
      GXX(J)=0.0
      GYY(J)=0.0
      GYXRE(J)=0.0
6     GYXIM(J)=0.0
      IHANI=0
      LS=KM
      B=0.0
      WRITE(1,2)
      READ(1,)NS,IHAN
      IF(IHAN.NE.1) GO TO 9
      WRITE(1,7)
      READ(1,)P,IHANI
9     IF(NS.EQ.1) GO TO 10
      LS=(KM+1)/NS
      WRITE(1,3)
      READ(1,)B
10    O=(1.0-B)*LS
      LSH=(LS+1)/2
      N=IFIX((-B+NS)/(1.0-B))
      DO 1 K=1,N
      KK1=ISTRT-1+IFIX((K-1)*O)
      KK2=IOSTRT-1+IFIX((K-1)*O)
      DO 5 J=1,LS
      K1=J+KK1
      K2=J+KK2
      IF(K1.GT.LIN) K1=K1-LIN
      IF(K2.GT.LOUT) K2=K2-LOUT
      FFR(J)=U(K1)
5     FFI(J)=V(K2)
      IF(IHANI.EQ.1) CALL HANN(FFR,LS,P)
      IF(IHAN.EQ.1) CALL HANN(FFI,LS,P)
C TRANSFORM AND CALCULATE GEES
      CALL FFT1A(FFR,FFI,LS,LS,LS,-1)
```

BACH (continued)

```
21      DO 1 I=2,LSH,IK
        K1=I
        K2=LS-I+2
        A=.5*(FFR(K1)+FFR(K2))
        B=.5*(FFI(K1)-FFI(K2))
        X=.5*(FFI(K2)+FFI(K1))
        Y=.5*(FFR(K2)-FFR(K1))
        GXX(I)=A*A+B*B+GXX(I)
        GYY(I)=X*X+Y*Y+GYY(I)
        GYXRE(I)=X*A+Y*B+GYXRE(I)
 1      GYXIM(I)=Y*A-X*B+GYXIM(I)
        DO 8 J=2,LSH,IK
        GXX(J)=GXX(J)/N
        GYY(J)=GYY(J)/N
        GYXRE(J)=GYXRE(J)/N
8       GYXIM(J)=GYXIM(J)/N
        KM=LS
2       FORMAT(" NUMBER OF SEGMENTS? HANNING?")
7       FORMAT(" %TAPER ON WINDOW? HANN INPUT AS WELL?")
3       FORMAT(" OVERLAP FRACTION?")
        RETURN
        END
```

OUTPUT

```
        SUBROUTINE OUTPUT(A,B,F,K,AL,BL)
        REAL A(1),B(1),F(1),BL(1),FIL(2)
        WRITE(1,1)AL
1       FORMAT(1X,A5)
        READ(1,)IG
        IF(IG.NE.1) RETURN
3       IF(ITOG(2))13,14,13
13      CALL PICTUR(F,A,K,1)
14      IF(ITOG(4))15,16,15
15      CALL PLTFRQ(A,F,K,0,BL)
16      IF(ITOG(8))4,5,4
4       WRITE(4,6)(F(J),A(J),B(J),J=2,K)
        CALL CLOSE(4)
6       FORMAT(1X,3E17.5)
5       IF(ITOG(16))7,8,7
7       WRITE(1,9)
9       FORMAT(" OUTPUT FILENAME?")
        READ(1,10)FIL
10      FORMAT(2A5)
        CALL ENTER(6,FIL)
        WRITE(6,6)(F(J),A(J),B(J),J=2,K)
        CALL CLOSE(6)
8       IF(ITOG(32))3,12,3
12      RETURN
        END
```

A3.3 FITF

```
C MASTER PROGRAM FOR FREQUENCY RESPONSE PACKAGE
C PACKAGE HAS TWO MODES A AND B
C
C CALLS SUBROUTINES FREPIC,QUFREQ,PZFREQ,ROUTH,RIMP,FFIT,MATRIX,INVMX
C,MXP,ADJ,PLTFRQ,SCPROD,PICTUR,GRAFIX,ITOG
C
C MODE A READS FREQ RESPO DATA IN VECTOR OR POLAR FORM FROM
C TT,PR,DT1,DT2.   DRAWS BODE PLOTS OF DATA. FITS POLYNOMIALS
C IN S.   DRAWS BODE PLOTS OF FITTED FUNCTIONS.
C CALCULATES COMPLEX POLES,ZEROS OF FUNCTIONS.
C
C
C MODE B DRAWS BODE PLOTS FROM POLYCOFFS OR PZS OR TIME CONSTS
C
C
C DATA IN SERIAL FORM FREQ,MOD,PHI..... K TRIADS
C FREQ IN HZ
C
C PACKAGE WRITTEN BY M.J.HILL  JAN 1975
      REAL R(200),I(200),F(200),SPN(400),SPD(200),AL(9),FIL(2),
     1PM(200),PP(200),BL(9)
      DATA AL(1),AL(2),AL(4),AL(5),AL(7),AL(8),AL(9),BL(1),BL(2),
     1BL(4),BL(5),BL(7),BL(8),BL(9)/
     25HFREQ ,5H (HZ),5HMOD (,5HDB). ,5HFREQ ,5HRESPO,5HNSE. ,
      35HFREQ ,5H (HZ),5HPHI (,5HDEG).,5HFREQ ,5HRESPO,5HNSE. /
  14   WRITE(1,19)
 19    FORMAT(" FIT=0,DRAW=1")
       READ(2,)IF
       IF(IF-1)22,20,22
 20    CALL FREPIC(R,I,F,K)
       GO TO 49
 22    WRITE(1,1)
 1     FORMAT(" K DATA?,DATCHAN?,POLAR=1,VECTOR=2")
       READ(2,)K,IN,K1
       IF(K.GT.200) GO TO 22
       IF(IN.LT.6) GO TO 15
 2     WRITE(1,16)
 16    FORMAT(" FILENAME?")
       READ(2,17)FIL
       CALL FSTAT(IN,FIL,J)
       IF(J.EQ.0) GO TO 2
 17    FORMAT(2A5)
       CALL SEEK(IN,FIL)
 15    READ(IN,)(F(J),R(J),I(J),J=1,K)
       CALL CLOSE(IN)
 49    IF(ITOG(2)) GO TO 44
       GO TO 24
 44    WRITE(1,46)
 46    FORMAT(" PHASE +/- RAMP?")
       READ(2,)P
       P=P*180.0/K
       DO 47 J=1,K
 47    I(J)=J*P
 24    WRITE(1,4)
 4     FORMAT(" IT FFIT ITERATIONS?")
       READ(2,) IT1
       IF(IT1)34,35,35
 35    WRITE(1,36)
```

```
36      FORMAT(" CONVERGENCE LIMIT?")
        READ(2,)FIT
34      WRITE(1,37)
37      FORMAT(" N/M TH ORDER POLLYFIT?")
        READ(2,)NP,MP
        KM=NP+MP+1
        IF(KM*(KM+5).GT.400) GO TO 34
        IF(K1.EQ.2) CALL RIMP(R,I,K,1)
21      CALL PLTFRQ(R,F,K,IF,AL)
        CALL PLTFRQ(I,F,K,IF,BL)
41      CALL RIMP(R,I,K,-1)
        IT=IT1
        CALL FFIT(F,R,I,K,SPN,SPD,400,FIT,IT,NP,MP)
        IF(IT.EQ.-1) WRITE(1,)FIT
        IF(IT.EQ.-1) GO TO 14
        WRITE(1,5) FIT
5       FORMAT(" SUM OF SQUARES OF ERROR OF FIT=",1PE12.4)
        ND=IFIX(SPD(1))
        NN=IFIX(SPN(1))
        DO 11 J=1,ND
11      SPD(J)=SPD(J+1)
        DO 12 J=1,NN
12      SPN(J)=SPN(J+1)
        CALL QUFREQ(SPD,ND,SPN,NN,F,K,PM,PP)
        CALL RIMP(PM,PP,K,1)
        CALL RIMP(R,I,K,1)
        IF(ITOG(2)) GO TO 18
        GO TO 42
18      KF=1
        DO 40 J=1,K
40      I(J)=PP(J)
        GO TO 41
42      WRITE(1,6)
6       FORMAT("/S↑N                DENOM COEFFS")
        DO 7 J=1,ND
        JJ=J-1
7       WRITE(1,8)JJ,SPD(J)
8       FORMAT(1X,I3,T18,1PE12.4)
        WRITE(1,9)
9       FORMAT("/S↑N                NUM COEFFS")
        DO 10 J=1,NN
        JJ=J-1
10      WRITE(1,8) JJ,SPN(J)
        CALL PLTFRQ(PM,F,K,1,AL)
        CALL PLTFRQ(PP,F,K,1,BL)
        IF(ND.EQ.1) GO TO 27
        ITER=1000
        CRIT=.0001
30      J=ND-1
        CALL NEWER(SPD,J,PM,ITER,CRIT)
        IF(J)28,29,28
28      WRITE(1,3)ITER,CRIT
        READ(2,)ITER,CRIT
        GO TO 30
29      J=ND-J-1
        J=J+J
        WRITE(1,25)
25      FORMAT("   REAL POLE          IMAG POLE")
```

FITF (continued)

```
      WRITE(1,13)(PM(I1),PM(I1+1),I1=1,J,2)
27    IF(NN.EQ.1) GO TO 13
      ITER=1000
      CRIT=.0001
31    J=NN-1
      CALL NEWER(SPN,J,PM,ITER,CRIT)
      IF(J)32,33,32
32    WRITE(1,3)ITER,CRIT
      READ(2,)ITER,CRIT
      GO TO 31
33    J=NN-J-1
      J=J+J
3     FORMAT(" ROOTS NOT FOUND WITH ITER=",I6,"CRIT=",E8.1,
     1"ENTER NEW ITER,CRIT")
      WRITE(1,26)
26    FORMAT("   REAL ZERO          IMAG ZERO")
      WRITE(1,13)(PM(I1),PM(I1+1),I1=1,J,2)
13    FORMAT(1X,1PE12.4,5X,1PE12.4)
      IF(KF.EQ.1) GO TO 14
      WRITE(1,23)
23    FORMAT(" FIT TO SAME DATA?")
      READ(2,)J
      IF(J-1) 14,24,14
      END
```

QUFREQ

```
C SUBROUTINE TO CALC REAL AND IMAGINARY PARTS OF N(JW)/D(JW)
C SUBROUTINE QUFREQ(A,JD,B,JN,F,N,X,Y)
C B=NUMERATOR COEFFICIENTS
C A=DENOM COEFFICIENTS
C JN=LENGTH OF NUM COEFFS
C JD=LENGTH OF DEN COEFFS
C F=FREQUENCY ARRAY (HZ)
C N=LENGTH OF F
C X,Y ARE ARRAYS TO RETURN RE AND IM
      SUBROUTINE QUFREQ(A,JD,B,JN,F,N,X,Y)
      DIMENSION A(1),B(1),F(1),X(1),Y(1)
      TPI=8.0*ATAN(1.0)
C CALC FREQ RESPO
      DO 9 J=1,N
      OMEG=F(J)*TPI
      RED=0.0
      XIMD=0.0
      REN=0.0
      XIMN=0.0
      JD1=JD-1
      JN1=JN-1
      DO 8 K=0,JD1,2
      EXP=(-1.0)**(K/2)
      RED=RED+(OMEG**K)*A(K+1)*EXP
      IF(K+2.GT.JD) GO TO 8
      XIMD=XIMD+(OMEG**(K+1))*A(K+2)*EXP
8     CONTINUE
      DO 6 K=0,JN1,2
      EXP=(-1.0)**(K/2)
      REN=REN+(OMEG**K)*B(K+1)*EXP
      IF(K+2.GT.JN) GO TO 6
      XIMN=XIMN+(OMEG**(K+1))*B(K+2)*EXP
6     CONTINUE
      DEN=RED*RED+XIMD*XIMD
      X(J)=(REN*RED+XIMN*XIMD)/DEN
9     Y(J)=(XIMN*RED-REN*XIMD)/DEN
      RETURN
      END
```

FFIT

```
C SUBROUTINE FFIT(F,R,I,N1,SPN,SPD,LIM,FIT,IT,NP,MP)
C FITS RATIO OF POLYNOMIALS TO FREQUENCY RESPONSE DATA.
C F=FREQUENCY POINTS (HZ)
C R,I HOLD REAL AND IMAG DATA
C N1=NUMBER OF FREQUENCY POINTS
C SPD IS WORKING ARRAY OF DIMENSION N1
C SPN IS WORKING ARRAY OF DIMENSION LIM
C FITTED NUM COEFFICIENTS RETURNED IN SPN
C SPN(1)=NUMBER OF NUM COEFFS.
C FITTED DENOMINATOR COEFFS RETURNED IN SPD.
C SPD(1)=NUMBER OF DEN COEFFS.
C FIT=SUM OF SQUARES OF ERROR OF FIT
C IT=MAXIMUM NUMBER OF ITERATIONS BEFOR BEING DEEMED INCONVERGENT
C -IT SPECIFIES NO OF ITERATIONS BEFORE NORMAL EXIT.
C FIT ON INPUT IS CONVERGENCE LIMIT.
C IT=1 FOR WEIGHTED LEAST SUM OF SQUARES OF ERROR
C IT RETURNED AS -1 IF ITERATIONS INCONVERGENT
C NP=NUMBER OF ZEROS TO BE FITTED
C MP=NUMBER OF POLES TO BE FITTED
C
C PROGRAM WRITTEN BY M.J.HILL USING MINIMUM MEAN SQUARE ERROR
C CRITERION. BASED ON SANATHANAN AND KOERNER (IEEE TRANS AUTO
C CONTROL JAN 1963.)
C OCTOBER 1974
C
      SUBROUTINE FFIT(F,R,I,N1,SPN,SPD,LIM,FIT,IT,NP,MP)
      REAL F(1),R(1),I(1),WKL(1),SPN(1),N(1),C(1),LAM(1),U(1),ST(1)
      1M(1,1),NN(1),SPD(1)
      TPI=8.0*ATAN2(1.0,1.0)
      MIT=ISIGN(1,IT)
      IT=IABS(IT)
      FITLIM=FIT
      ITK=0
      KM=MP+NP+1
      NP1=NP+1
      NMP=NP+MP
C FABRICATE WORKING ARRAYS FROM SP
  4   KMS=KM*KM+1
      KMS1=KMS+KM
      KMS2=KMS+2*KM
      KMS3=KMS+3*KM
      KMS4=KMS+4*KM
      CALL ADJ2(M,SPN(1),KMS-1)
      CALL ADJ(M,M(1,1),KM,KM,0)
      CALL ADJ(N,SPN(KMS),KM,0,0)
      CALL ADJ(C,SPN(KMS1),KM,0,0)
      CALL ADJ(LAM,SPN(KMS2),NP1,0,0)
      CALL ADJ(ST,SPN(KMS2+NP1),KM,0,0)
      CALL ADJ(U,SPN(KMS3+NP1),MP,0,0)
      CALL ADJ(NN,SPN(KMS4),KM,0,0)
      CALL ADJ(WKL,SPD(1),N1,0,0)
      DO 6 J=1,N1
      F(J)=F(J)*TPI
  6   WKL(J)=1.0
      DO 29 J=1,KM
 29   NN(J)=1.0E-6
      CALL ERRSET(0)
C CALC LAMBDAS
```

```
 61      FIT=0.0
         DO 7 IK=1,NP1
         LAM(IK)=0.0
         DO 7 K=1,N1
         I1=(IK-1)*2
 7       LAM(IK)=LAM(IK)+(F(K)**I1)*WKL(K)
C CALC S AND T
         DO 8 IK=1,KM,2
         ST(IK)=0.0
         I1=IK-1
         DO 8 K=1,N1
 8       ST(IK)=ST(IK)+(F(K)**I1)*R(K)*WKL(K)
         IF(KM.LT.2) GO TO 81
         DO 81 IK=2,KM,2
         ST(IK)=0.0
         I1=IK-1
         DO 81 K=1,N1
         ST(IK)=ST(IK)+(F(K)**I1)*I(K)*WKL(K)
 81      CONTINUE
C CALC U
         IF(MP.EQ.0) GO TO 9
         DO 9 IK=1,MP
         U(IK)=0.0
         DO 9 K=1,N1
         U(IK)=U(IK)+(F(K)**(2*IK))*(R(K)*R(K)+I(K)*I(K))*WKL(K)
 9       CONTINUE
C NOW FILL UP ARRAY M WITH LAM
         DO 10 IK=1,NP1,2
         DO 10 J=1,NP1,2
         MJ=(J-1)/2
         MK=(IK-1)/2
         M(IK,J)=LAM(MK+1+MJ)*((-1.0)**MJ)
         IF(J.GE.NP1) GO TO 10
         M(IK,J+1)=0.0
 10      CONTINUE
         IF(NP1.EQ.1) GO TO 11
         DO 11 IK=2,NP1,2
         DO 11 J=2,NP1,2
         MJ=J/2-1
         MK=IK/2-1
         M(IK,J)=LAM(MJ+2+MK)*((-1.0)**MJ)
         M(IK,J-1)=0.0
 11      CONTINUE
C FILL UP U
         IF(MP.EQ.0) GO TO 13
         DO 12 IK=1,MP,2
         DO 12 J=1,MP,2
         MS=(J+1)/2
         M(IK+NP1,J+NP1)=U(MS+(IK-1)/2)*((-1.0)**(MS+1))
         IF(J.GE.MP) GO TO 12
         M(IK+NP1,J+NP1+1)=0.0
 12      CONTINUE
         IF(MP.EQ.1) GO TO 13
         DO 13 IK=2,MP,2
         DO 13 J=2,MP,2
         MS=J/2
         M(IK+NP1,J+NP1)=U(MS+IK/2)*((-1.0)**(MS+1))
         M(IK+NP1,J+NP)=0.0
```

```
 13      CONTINUE
C FILL UP ST LEFT HAND
         IF(MP.EQ.0) GO TO 20
         DO 14 J=1,NP1
         DO 14 IK=1,MP
 14      M(IK+NP1,J)=ST(IK+J)
         K=1
         DO 16 J=1,NP1
         DO 15 IK=1,MP
         IF(K.EQ.2) M(IK+NP1,J)=M(IK+NP1,J)*((-1.0)**IK)
         IF(K.EQ.3) M(IK+NP1,J)=-M(IK+NP1,J)
 15      IF(K.EQ.4) M(IK+NP1,J)=M(IK+NP1,J)*((-1.0)**(IK-1))
         K=K+1
 16      IF(K.GT.4) K=K-4
C FILL UP ST RIGHT HAND
         DO 17 J=1,MP
         DO 17 IK=1,NP1
 17      M(IK,J+NP1)=ST(IK+J)
         K=1
         DO 19 J=1,MP
         DO 18 IK=1,NP1
         IF(K.EQ.1) M(IK,J+NP1)=M(IK,J+NP1)*((-1.0)**(IK-1))
         IF(K.EQ.3) M(IK,J+NP1)=M(IK,J+NP1)*((-1.0)**IK)
 18      IF(K.EQ.4) M(IK,J+NP1)=-M(IK,J+NP1)
         K=K+1
 19      IF(K.GT.4) K=K-4
C FILL UP C
         DO 20 J=1,MP,2
         C(J+NP1)=0.0
         IF(J.EQ.MP) GO TO 20
         C(J+NP1+1)=U((J+1)/2)
 20      CONTINUE
         DO 21 J=1,NP1
 21      C(J)=ST(J)
C INVERT M
         CALL SCPROD(M,2.0**(-30))
         CALL INVMX(M)
         CALL SCPROD(M,2.0**(-30))
         CALL MXP(N,M,C,KM)
C CALC WKL
         IF(MP.EQ.0) GO TO 28
         DO 23 J=1,N1
         QR=1.0
         QI=0.0
         DO 22 IK=1,MP,2
         QI=QI+N(NP1+IK)*(F(J)**IK)*((-1.0)**((IK-1)/2))
         IF(IK.EQ.MP) GO TO 22
         K=IK+1
         QR=QR+N(NP1+K)*(F(J)**K)*((-1.0)**(K/2))
 22      CONTINUE
 23      WKL(J)=1.0/(QR*QR+QI*QI)
C CALC SUM OF CHANGE IN COEFFICIENTS.
         DO 24 J=1,KM
         FIT=FIT+ABS((NN(J)-N(J))/NN(J))
 24      NN(J)=N(J)
         ITK=ITK+1
         IF(MIT)34,35,35
 34      IF(ITK-IT)61,28,28
```

FFIT (continued)

```
 35      IF(ITK.EQ.IT.AND.FIT.GT.FITLIM) GO TO 31
         IF(FIT.GT.FITLIM) GO TO 61
C CALC SUM O F SQUARES OF ERROR.
 28      FIT=0.0
         DO 25 K=1,N1
         RN=0.0
         XIN=0.0
         DO 26 J=1,NP1,2
         RN=RN+N(J)*(F(K)**(J-1))*((-1.0)**((J-1)/2))
         IF(J.EQ.NP1) GO TO 26
         XIN=XIN+N(J+1)*(F(K)**J)*((-1.0)**((J-1)/2))
 26      CONTINUE
         QR=SQRT(WKL(K)*(RN*RN+XIN*XIN))
         PR=SQRT(R(K)*R(K)+I(K)*I(K))
         FIT=FIT+(PR-QR)*(PR-QR)
 25      F(K)=F(K)/TPI
C INSERT COEFFS INTO SPN,SPD.
         MP1=MP+1
         SPN(1)=FLOAT(NP1)
         SPD(1)=FLOAT(MP1)
         DO 30 J=1,NP1
 30      SPN(J+1)=N(J)
         SPD(2)=1.0
         DO 32 J=1,MP
 32      SPD(J+2)=N(NP1+J)
         CALL ERRSET(1)
         RETURN
 31      IT=-1
         DO 33 J=1,N1
 33      F(J)=F(J)/TPI
         CALL ERRSET(1)
         RETURN
         END
```

ROUTH

```
         SUBROUTINE ROUTH(A,JD,IROT)
         REAL A(1),R(11,5)
         IROT=0
         DO 4 I=1,11
         DO 4 J=1,5
 4       R(I,J)=0.0
         DO 1 J=0,JD,2
         R(1,1+J/2)=A(JD-J)
         IF(JD-J.LT.2) GO TO 1
         R(2,1+J/2)=A(JD-J-1)
 1       CONTINUE
         DO 2 I=3,11
         DO 2 J=1,4
         IF(R(I-1,1).EQ.0.0) GO TO 2
         R(I,J)=(R(I-1,1)*R(I-2,J+1)-R(I-2,1)*R(I-1,J+1))/R(I-1,1)
  2       CONTINUE
         DO 3 J=1,10
 3       IF(R(J+1,1)*R(J,1).LT.0.0) IROT=-1
         RETURN
         END
```

PZFREQ

```
C SUBROUTINE TO PRODUCE MOD,PHASE FROM REAL POLES/ZEROS
C B=ZEROS
C M=NO OF ZEROS
C A=POLES
C N=NO OF POLES
C GK=GAIN CONST
C F=FREQ ARRAY (HZ)
C K=NO OF FREQ POINTS
C XMOD=MOD (DB)
C PHI=PHASE(DEG)
      SUBROUTINE PZFREQ(A,N,B,M,GK,F,K,XMOD,PHI)
      DIMENSION A(1),B(1),F(1),XMOD(1),PHI(1)
      RAD=45.0/ATAN2(1.0,1.0)
      TPI=360.0/RAD
      GKL=ALOG10(GK)*20.0
      DO 1 I=1,K
      XMOD(I)=0.0
      PHI(I)=0.0
      F1=F(I)*TPI
      F2=F1*F1
C CALC DEN CONTRIBUTION
      IF(N.EQ.0) GO TO 2
      DO 2 J=1,N
      XMOD(I)=XMOD(I)-ALOG10(A(J)*A(J)+F2)
      PHI(I)=PHI(I)-ATAN2(F1,-A(J))
2     CONTINUE
C CALC NUM CONTRIBUTION
      IF(M.EQ.0) GO TO 3
      DO 3 J=1,M
      XMOD(I)=XMOD(I)+ALOG10(B(J)*B(J)+F2)
      PHI(I)=PHI(I)+ATAN2(F1,-B(J))
3     CONTINUE
      XMOD(I)=XMOD(I)*10.0+GKL
1     PHI(I)=PHI(I)*RAD
      RETURN
      END
```

MXP

```
      SUBROUTINE MXP(A,B,C,KM)
      REAL A(1),B(1,1),C(1)
      DO 1 I=1,KM
      TEMP=0.0
      DO 2 J=1,KM
2     TEMP=TEMP+B(I,J)*C(J)
1     A(I)=TEMP
      RETURN
      END
```

FREPIC

```
C SUBROUTINE TO CALCULATE FREQ RESPO OF ANY SYSTEM.
C RETURNS MOD (DB) AND PHI (DEG)
C ASKS FOR POLYCOFFS OR PZS OR TIME CONSTS.
      SUBROUTINE FREPIC(XMOD,PHI,F,N)
      REAL A(10),B(10),XMOD(1),PHI(1),F(1)
      DO 7 J=1,10
      A(J)=0.0
 7    B(J)=0.0
      N=200
      GK1=1.0
      WRITE(1,8)
 8    FORMAT(" POLYCOFFS 1,PZS 2,TIME CONSTS 3")
      READ(2,)IPZ
C READ IN DENOM COEFFS
      WRITE(1,1)
 1    FORMAT(" ORDER OF DENOMINATOR?")
      READ(2,)JD
      IF(IPZ.LT.2) JD=JD+1
      IF(JD.EQ.0.AND.IPZ.GT.1) GO TO 15
      WRITE(1,9)
 9    FORMAT(" ENTER DENOM COEFFS")
      READ(2,)(A(J),J=1,JD)
      IF(IPZ.EQ.1) CALL ROUTH(A,JD,IROT)
      IF(IROT.EQ.-1) WRITE(1,15)
 15   FORMAT(" ROUTH DISSATISFIED")
      IROT=0
C READ IN NUMERATOR COEFFS
      WRITE(1,3)
 3    FORMAT(" ORDER OF NUMERATOR?")
      READ(2,)JN
      IF(IPZ.LT.2) JN=JN+1
      IF(JN.EQ.0.AND.IPZ.GT.1) GO TO 4
      WRITE(1,10)
 10   FORMAT(" ENTER NUM COEFFS")
      READ(2,)(B(J),J=1,JN)
 4    IF(IPZ.EQ.1) GO TO 5
      WRITE(1,13)
 13   FORMAT(" GAIN CONST?")
      READ(2,)GK
C CALCULATE FREQUENCY RANGE
 5    WRITE(1,6)
 6    FORMAT(" FREQ FROM, TO (HZ)")
      READ(2,)FL,FU
      FUL=ALOG10(FU/FL)
      N1=N-1
      RN=FLOAT(N1)
      DO 14 J=0,N1
      FJ=FLOAT(J)
 14   F(J+1)=FL*10.0**(FUL*FJ/RN)
      IF(IPZ-2)16,20,18
 16   CALL QUFREQ(A,JD,B,JN,F,N,XMOD,PHI)
      CALL RIMP(XMOD,PHI,N,1)
      RETURN
 18   IF(JN.EQ.0) GO TO 19
      DO 19 J=1,JN
      GK1=B(J)*GK1
      B(J)=1.0/B(J)
 19   CONTINUE
```

FREPIC (continued)

```
      IF(JD.EQ.0) GO TO 20
      DO 20 J=1,JD
      GK=GK/A(J)
      A(J)=1.0/A(J)
20    CONTINUE
      GK=GK*GK1
      CALL PZFREQ(A,JD,B,JN,GK,F,N,XMOD,PHI)
      RETURN
      END
```

RIMP

```
C SUBROUTINE TO CONVERT VECTOR DATA TO POLAR DATA OR VICE-VERSA
C RIMP(A,B,N,K)
C A=REAL OR DB ARRAY
C B=IMAG OR DEG ARRAY
C N=LENGTH OF ARRAYS
C K=1 FOR VECTOR TO POLAR
C K=-1 FOR POLAR TO VECTOR
C PHASE DISCONTINUITIES REMOVED
      SUBROUTINE RIMP(A,B,N,K)
      DIMENSION A(1),B(1)
      RAD=ATAN2(1.0,1.0)/45.0
      IF(K.EQ.1) GO TO 5
      DO 1 J=1,N
      A1=10.0**(A(J)/20.0)
      A(J)=A1*COS(RAD*B(J))
1     B(J)=A1*SIN(RAD*B(J))
      GO TO 9
5     DO 2 J=1,N
      A1=A(J)
      IF(ABS(A1)+ABS(B(J)).EQ.0.0) GO TO 11
      A(J)=10.0*ALOG10(A1*A1+B(J)*B(J))
      GO TO 2
11    A(J)=-200.0
2     B(J)=ATAN2(B(J),A1)/RAD
C REMOVE PHASE DISCONTINUITIES
3     FKU=0.0
      KFLAG=0
      N1=N-1
      DO 4 J=1,N1
      IF(B(J+1)-B(J).GT.160.0) FKU=-1.0
      IF(B(J+1)-B(J).LT.-160.0) FKU=1.0
      IF(FKU.NE.0.0) KFLAG=J+1
      IF(KFLAG.NE.0) GO TO 6
4     CONTINUE
      GO TO 9
6     DO 7 J=KFLAG,N
7     B(J)=B(J)+FKU*360.0
      GO TO 3
9     CONTINUE
      RETURN
      END
```

REFERENCES

1. BRICE J C 1965, "The Growth of Crystals from the Melt", N Holland
2. CZOCHRALSKI J 1918, Z Phys Chem 92 p219
3. ZIMMERMAN W 1954, Science 119 p411
4. FAY H and BRANDLE 1967, in Peiser p51
5. METZ E P A, MILLER R C, and MAZELSKY R 1962, J Appl Phys 33 p2016
6. BASS S J and OLIVER P E 1968, J Crystal Growth 3-4, pp286-290
7. LEVINSON J 1959, US Patent 2 908 044
8. REINERT R C and YATSKO M A 1974, J Crystal Growth 21, pp283-286
9. ZINNES, NEVIS and BRANDLE 1973, J Crystal Growth 19, pp187-192
10. BARDSLEY W et al 1974, J Crystal Growth 24-25, p 369
11. BARDSLEY W et al 1972, J Crystal Growth 16, pp277-279
12. JEN D, SLOCUM R and VALENTINO C 1971, US Patent 3 621 213
13. CORBURN D S, SEKSINSKY J J and TUCKER R B 1972, IBM Technical Disclosure Bull 15, 4, pp1334-1335
14. PATZNER E, DESSAUER R, and POPONIAK M 1967, SCP and Solid State Technology, Oct p25
15. DIGGES T G Jr, HOPKINS R H and SEIDENSTICKER R G 1975, J Crystal Growth 29, pp326-328
16. GROSS U and KERSTEN R 1972, J Crystal Growth 15, pp85-88
17. O'KANE D F, KWAP T W, GULITZ L and BEDNOWITZ A L 1972, J Crystal Growth 13-14, pp624-628
18. BACHMANN K J, KIRSCH H J and VETTER K J 1970, J Crystal Growth 7, pp 290-295
19. GARTNER K J, RITTINGHAUS K F, SEEGER A and UELHOFF W 1972 J Crystal Growth 13-14, pp619-623
20. PHILLIPS 1967, Netherlands Patent 6 512 921
21. VAN DIJK H J A, JOCHEM C M G, SCHOLL G J, VAN DER WERF P 1974, J Crystal Growth 21, pp310-312
22 PRUETT H D, LIEN S Y 1974, J Electrochem Soc June pp822-826
23. VOJDANI S, DABIRI A E and ASHOORI H 1974, J Crystal Growth 24-25 pp 374-375
24. HURLE D T J 1972, J Crystal Growth 13-14, pp39-43

25. BLOM G M 1972, Second National Conference on Crystal Growth, American Association for Crystal Growth, July-August, Princeton University.

26. BRICE J C 1973, Acta Electronica 16, 4, pp291-301

27. CHESSWAS M, COCKAYNE B, HURLE D T J, JAKEMAN E, MULLIN J B 1971, J Crystal Growth 11, pp225-232

28. SUREK T and CHALMERS B 1975, J Crystal Growth 29, pp1-11

29. KOBAYASHI N and ARIZUMI T 1970, Japan J Applied Physics 9, 4, pp 361-367

30. SHASHKOV M, GRISHIN V P 1968, Doklady Akad Nauk SSSR 179, 2,pp404-6

31. SHASKOV M, GUREVICH V M 1969, Doklady Akad Nauk SSSR 187, 1, pp146-148

32. SHASHKOV M, GUREVICH V M, SILKIN V B 1969, Izvest Akad Nauk SSSR 33, 12, pp1800-1805

33. AKIYAMA K, YAMAGUCHI J 1962, J Applied Physics 33, 5, pp1899-1900

34. SAKHAROV B A, SILKIN V B, SHASHKOV M 1969, Izv.An SSSR Neorgan Mater, 2

35. STEEL G K, HILL M J 1975, J Crystal Growth 30, pp45-53

36. ARIZUMI T, KOBAYASHI N 1969, Japan J Applied Physics 8,9,pp1091-1097

37. ARIZUMI T, KOBAYASHI N 1972, J Crystal Growth 13-14, pp615-618

38. BRICE J C 1968, J Crystal Growth 2, pp395-401

39. KUO V H S, WILCOX W R 1972, J Crystal Growth 12, pp191-194

40. KOBAYASHI N, ARIZUMI T 1970, Japan J Applied Physics 9,10,pp1255-1259

41. KOBAYASHI N, ARIZUMI T 1975, J Crystal Growth 30, pp177-184

42. SHASHKOV M, STEPANOVA G M, NIKITIN V M 1969, Izvest Akad Nauk SSSR 33,12, pp2027-30

43. NYGREN S F 1973, J Crystal Growth 19, p21

44. NEWMAN and SEARLE 1957, "The General Properties of Matter", Edition V, Arnold

45. MATEJEVIC (Editor) 1969, "Surface and Colloid Science" 2, Wiley

46. ANTONOV 1965, Sb Rost Kristallov, 6, p158

47. GIBBONS J 1974, Private communication

48. ZIERLER N 1959, J Soc Ind Appl Math 7, pp31-48

49. GODFREY K R 1969, Measurement and Control 2, May ppT65-T72

50. GODFREY K R 1969, Proc IEE, 116, 5, pp879-888

51. CHANG J A, OWEN A G, ZAMAN M, GRIFFIN A W J 1968, Measurement and Control 1, April ppT80-T84

52. BARKER H A, DAVY R W 1975, Proc IEE 122, 3, pp305-311

53. CARTER KNAPP AND NUTTAL 1973, IEEE Trans Audio and Electroacoustics, Au-21, 4, pp337-344

54. BARKER H A, OBIDEGWU S 1973, Bradford Symposium , IEE

55. LAMB J D, REES D 1973, IEE Conf Pub 103, pp141-146

56. BENDAT AND PIERSOL 1971, "Random Data Analysis and Measurement Procedures", Wiley

57. KUO F F and KAISER J F 1966, "System Analysis by Digital Computer" Wiley

58. CONSTANTINIDES A G 1970, Proc IEE 117, 6, pp1585-1590

59. SINGLETON R C 1969, IEEE Trans Audio and Electroacoustics Au-17 pp 93-102

60. LEVY E C 1959, IRE Trans Auto Control, May, pp37-43

61. SANATHANAN C K and KOERNER J 1963, IEEE Trans Auto Control, Jan, pp56-58

62. ACKROYD M H 1973, "Digital Filters", IEE Press