SAND76-8200
Unlimited Release

ESR Powder Line Shape Calculations

J. Vitko, Jr., R. E. Huddleston

Prepared by Sandia Laboratories, Albuquerque New Mexico 87115
and Livermore, California 94550 for the United States Energy Research
and Development Administration under Contract AT (29-1)-789

Printed May 1976

Sandia Laboratories

SF 2900 Q(7-73)

When printing a copy of any digitized SAND Report, you are required to update the markings to current standards.

Issued by Sandia Laboratories, operated for the United States Energy Research and Development Administration by Sandia Corporation.

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

SAND76-8200
Unlimited Release
Printed May 1976

ESR POWDER LINE SHAPE CALCULATIONS

J. Vitko, Jr.
Physical Research Division 8334

R. E. Huddleston
Numerical Applications Division 8322
Sandia Laboratories, Livermore

ABSTRACT

A program has been developed for computing the ESR spectrum of a collection of randomly oriented spins subject only to an electronic Zeeman interaction and having a Lorentzian single crystal line shape. Other single crystal line shapes, including numerical solutions of the Bloch equations, can be accommodated with minor modifications. The program differs in several features from those existing elsewhere, thus enabling one to study saturation effects, over-modulation effects, both absorptive and dispersive signals, and second and higher order derivative signals.

TABLE OF CONTENTS

	Page
Introduction	7
Approximations	8
Powder Line Shapes	9
Additional Program Features	13
Effect of Overmodulation	13
The Case of Gaussian Broadening	14
Future Directions	15
Use of Program	16
Appendix I	23
The Effective g Value	23
Orientation Dependence of Transition Probability	25
The Powder Line Shape	28
Appendix II	33

ESR POWDER LINE SHAPE CALCULATIONS

I. Introduction

In extending the applicability of ESR from the study of radiation defects in single crystal material to similar studies in amorphous insulating materials, one must try to understand and model the resulting change in line shape. Specifically, it was decided to model as simply as possible those interesting features which presented themselves during a recent study[1] of the E_1' defect center (an electron trapped at an oxygen vacancy in the SiO_2 network) in amorphous silica.

This effort has resulted in the development of a flexible computer program for calculating the ESR spectrum of a collection of randomly oriented spins, with the approximation that the only interaction present is the electronic Zeeman interaction and that the single crystal line shape is Lorentzian. This program, though similar to those existing elsewhere and described in various professional journals, differs significantly in several features, enabling one to study saturation effects, overmodulation effects, both absorptive and dispersive signals, and second and higher order derivative signals. Further, the program is written to facilitate changing the single crystal line shape function, with the future possibility of inputting a numerical solution of the Bloch equations, thus enabling the study of passage effects.

---

[1] J. Vitko, Jr., (SLL), M. Moran, (LLL), and M. Schwab, (LLL), to be published.

This report is a description of the resulting program. Sections II-III trace the development of the line shape equation, explicitly stating the underlying assumptions. Additional program features, as well as future directions, are discussed in Section IV. User intructions and sample graphical output appear in Section V. For those desiring additional detail, Appendix I contains the derivation of the line shape equation for the axially symmetric case, and Appendix II contains various program details, including flow chart, variable list, and a discussion of numerical techniques used. Microfiche containing a program listing and sample numerical output is attached to the inside back cover of this report.

II. Approximations

A. In general the effective spin Hamiltonian can be quite complicated, e.g., Eq. (1):

$$\mathcal{H} = \beta\, \overline{S}\cdot\overline{g}\cdot\overline{H} + \overline{S}\cdot\overline{D}\cdot\overline{S} + \overline{S}\cdot\overline{A}\cdot\overline{I} + \overline{I}\cdot\overline{Q}\cdot\overline{I} - \gamma\hbar\, \overline{I}\cdot\overline{H} \tag{1}$$

where the first term describes the electronic Zeeman interaction, second term the fine structure, third term the hyperfine interaction, fourth term the nuclear quadrupole interaction, and the fifth term the nuclear Zeeman interaction.

We shall ignore all but the electronic Zeeman interaction, i.e., we treat

$$\mathcal{H} = \beta\, \overline{S}\cdot\overline{g}\cdot\overline{H} \tag{2}$$

B. The ESR line shape in an amorphous material may differ from that in a single crystal for two reasons:

1. If the $\bar{g}$ tensor is anisotropic there will be an orientational broadening due to the fact that the defect is randomly oriented with respect to the applied magnetic field;

2. There may now be a distribution of effective spin Hamiltonian parameters which reflect the distribution of some physical property in the glass structure.

In this treatment we ignore (2), i.e., we assume that the defect is characterized by a unique set of spin Hamiltonian parameters. Further, we assume that the principal axes of the g tensor are randomly oriented with respect to the applied field.

C. In general, for an anisotropic $\bar{g}$ tensor the transition probability (due to the microwave field H_1^*) is also a function of the orientation, and hence its affect on the line shape must also be considered. In Appendix I, we show that for small anisotropies this effect is small. In the program we assume the transition probability is independent of orientation.

D. An isotropic Lorentzian has been assumed for the single crystal line shape.

III. Powder Line Shapes

Having explicitly stated our approximations, we may proceed to the calculation of the powder line shape.

Let $Y(H-H')$ be the line shape in the single crystal, where H' is the resonant magnetic field and is given by

$$H' = h\nu/g'\beta \tag{3}$$

where

$$g' = \left(g_1^2 \cos^2 \Theta_1 + g_2^2 \cos^2 \Theta_2 + g_3^2 \cos^2 \Theta_3\right)^{1/2} \tag{4}$$

and Θ_i is the angle between the applied magnetic field and i^{th} principal axis of the g tensor.

The line shape in the powder is then given by

$$I(H) = \frac{1}{4\pi}\int_{4\pi \text{ solid angle}} Y(H-H')\, d\Omega \tag{5}$$

By careful application of Eq. (3) and (4) one can transform[2] the integral in Eq. (5) from an integral over all orientations to an integral over all allowed values of g' or equivalently to an integral over all possible resonant fields H'. The results of such a transformation are conveniently written as

$$I(H) = \int_{H_3}^{H_1} S(H')\, Y(H-H')\, dH' \tag{6}$$

where

(i) $S(H')$ is known as the shape function, and $S(H')dH'$ may be thought of as the probability that a randomly oriented molecule of $\bar{g}$ tensor $g_3 > g_2 > g_1$ will have its resonant field between H' and $H' + dH'$;

(ii) the g values are relabeled, if need be, so that $g_3 > g_2 > g_1$. Naturally the labeling does not affect the physics in any way, however, the ordering $g_3 > g_2 > g_1$ has been assumed in deriving an explicit function for $S(H')$, and

[2]For details see F. K. Kneubühl, J. Chem. Phys. 33, 1074, 1960, and C. P. Poole, Electron Spin Resonance, John Wiley & Sons, c 1967, p. 826 ff.

(iii) $H_1 = \frac{h\nu}{\beta g_1}$; $H_2 = \frac{h\nu}{\beta g_2}$; $H_3 = \frac{h\nu}{\beta g_3}$.

The shape factor $S(H')$ is given by

$$S(H') = 0 \qquad 0 \le H' \le H_3 \qquad (7)$$

$$S(H') = \frac{2}{\pi} \frac{H_1 H_2 H_3}{H'^2} \frac{K(k)}{\left(H_1^2 - H'^2\right)^{1/2}\left(H_2^2 - H_3^2\right)^{1/2}} \qquad H_3 \le H' \le H_2$$

$$S(H') = \frac{2}{\pi} \frac{H_1 H_2 H_3}{H'^2} \frac{K(1/k)}{\left(H_1^2 - H_2^2\right)^{1/2}\left(H'^2 - H_3^2\right)^{1/2}} \qquad H_2 \le H' \le H_1$$

$$S(H') = 0 \qquad H_1 \le H'$$

where

$$K(k) = \int_0^1 \frac{dx}{\left(1-x^2\right)^{1/2}\left(1-k^2x^2\right)^{1/2}} \text{ , i.e., an elliptic integral} \qquad (8)$$

of the first kind,

and

$$k^2 = \frac{\left(H_1^2 - H_2^2\right)\left(H'^2 - H_3^2\right)}{\left(H_1^2 - H'^2\right)\left(H_2^2 - H_3^2\right)}$$

For a single crystal line shape we have used the standard Lorentzian[3] expressions for the absorptive (χ'') and the dispersive (χ') components, i.e.,

---

[3] A more general treatment would have assumed an inhomogeneous single crystal line shape which is given by the convolution of a Gaussian distribution with a Lorentzian homogeneous line shape. Such a procedure introduces another fitting parameter (in addition to the T_1, T_2 of the Lorentzian line), i.e., the width of the Gaussian distribution.

$$Y(H-H') = \frac{\dfrac{1}{\frac{1}{2}\Delta H_{1/2}}}{1 + s + \left(\dfrac{H'-H}{\frac{1}{2}\Delta H_{1/2}}\right)^2} \qquad \text{Absorption} \qquad (10)$$

$$Y(H-H') = \frac{\left(\dfrac{1}{\frac{1}{2}\Delta H_{1/2}}\right)\left(\dfrac{H'-H}{\frac{1}{2}\Delta H_{1/2}}\right)}{1 + s + \left(\dfrac{H'-H}{\frac{1}{2}\Delta H_{1/2}}\right)^2} \qquad \text{Dispersion} \qquad (11)$$

where

$\Delta H_{1/2}$ = full width at half maximum and is related to the transverse relaxation time T_2 through the expression

$$T_2 = \frac{2}{(\gamma\ \Delta H_{1/2})} \qquad (12)$$

and s is the saturation parameter

$$s = \gamma H_1^* T_1 \left(\frac{H_1^*}{\frac{1}{2}\Delta H_{1/2}}\right) \qquad (13)$$

where γ, the gyromagnetic ratio, = $g\beta/\hbar$ (β = Bohr magneton) and the microwave field H_x is given by

$$H_x = 2H_1^* \cos \omega t \ .$$

It should be noted that we have assumed the single crystal line shape and its saturation behavior to be isotropic, specifically we have assumed that T_1 and T_2 are not a function of orientation and that

$$\gamma = \frac{g\beta}{\hbar} \simeq \frac{g_3\beta}{\hbar} \qquad (14)$$

The inclusion of the saturation parameter s allows us to examine the microwave saturation behavior of the powder line shape, an area that to date has been largely ignored. Besides being of intrinsic interest, this serves as an additional test of the fitting parameters obtained in the unsaturated regime.

IV. Additional Program Features

A. Effect of Overmodulation

For reasons of signal to noise considerations, one does not detect I(H) but instead superimposes a weak modulating field ($H_m \sin \omega_m t$) parallel to the applied static field and detects the component of $I(H + H_m \sin \omega_m t)$ that appears at a frequency ω_m. Expanding $I(H + H_m \sin \omega_m t)$ in a Fourier series we have

$$I(H + H_m \sin \omega_m t) = \frac{1}{2} a_o + \sum_{N=1}^{\infty} \left[a_N \sin (N\omega_m t) + b_N \cos (N\omega_m t) \right] \quad (15)$$

thus the in phase signal (phase referenced to modulating field) will be given by $a_N(H)$ and the out of phase signal will be given by $b_N(H)$.

Equivalently, for H_m small we could have expanded $I(H + H_m \sin \omega_m t)$ in a Taylor series in which case

$$I(H + H_m \sin \omega_m t) = I + I' \times (H_m \sin \omega_m t) + \frac{1}{2} I'' \times (H_m \sin \omega_m t)^2 + .. \quad (16)$$

where $' = \frac{d}{dH}$, and all derivatives are evaluated at H

or, making use of the trignometric identity

$$\sin^2 \Theta = \frac{1}{2} - \frac{1}{2} \cos 2\Theta \tag{17}$$

$$I(H + H_m \sin \omega_m t) = I + \frac{1}{4} I'' H_m^2 + I' H_m \sin \omega_m t$$

$$- \frac{1}{4} I'' H_m^2 \cos 2\omega_m t + \ldots \quad . \tag{18}$$

Thus for small H_m one can make the identification

$$a_1 (H) = H_m I' \qquad a_2 (H) = 0$$

$$b_1 (H) = 0 \qquad b_2 (H) = -\frac{1}{4} H_m^2 I'' \tag{19}$$

It is because of this identification of a_1 (H) with I' that one says he is doing 1st derivative spectroscopy. Clearly this is only an approximation and as H_m increases in magnitude the observed signal, i.e., a_1 (H), begins to depart from the signal calculated by simply taking the first derivative of I(H). This departure is known as an "overmodulation effect." Because this computer program allows us to calculate a_N, b_N through use of the subroutine FØUR, one can study overmodulation effects in powder line shapes and compare this to overmodulation effects in Lorentzian and Gaussian line shapes. Other areas where this feature will prove particularly useful is in 2nd (and higher order) derivative spectroscopy.

B. The Case of Gaussian Broadening

Many ESR experimentalists have developed an "intuitive" feel for the effects of overmodulation and saturation on inhomogeneous broadening in single crystals. This inevitably results in comparing the behavior of a powder broadened line with that of a Gaussian broadened single crystal line. To facilitate this comparison we have included the possibility of substituting

the Gaussian shape function of Eq. 20

$$S(H') = \frac{2}{\Delta H^*_{1/2}} \left(\frac{\ln 2}{\pi}\right)^{1/2} \exp\left[-\ln 2 \left(\frac{H' - H_o}{\frac{1}{2}\Delta H^*_{1/2}}\right)^2\right] \tag{20}$$

where

$\Delta H^*_{1/2}$ = full width at half maximum

$H_o = \frac{h\nu}{g_3\beta}$, assuming isotropy, i.e., $g_1 = g_2 = g_3$

in lieu of the powder shape function of Eq. 7. This substitution is accomplished by appropriate choice of the program parameter IFUNC (to be described later).

C. Future Directions

One obvious future direction points to the inclusion of more terms in the Hamiltonian, e.g., Eq. (1). This could not be done by simply extending this program, but would call for a different approach to line shape calculations (for a review of different approaches see, e.g., P. C. Taylor, J. F. Baugher, and H. M. Kirz, Chem. Rev. 75, 203, 1975).

Another, less obvious extension, is to take a more general approach to the single crystal line shape, initially along the lines of footnote #3, and more ambitiously to numerically solve the Bloch equations (using as input the experimental operating parameters) so that one can adequately describe passage effects.[4] Because these dynamic effects can be complicated, it is suggested that they first be studied in a simple system, for which corresponding single crystal data is available or readily attainable; the E_1' defect in silica appears to be such a system.

---

[4] M. Weger, Bell System Tech. J., 1013, 1960.

Finally, one may wish to relax the assumption that the defect is characterized by a unique set of parameters g_1, g_2, g_3 and instead allow for a distribution of these parameters. One must then make the connection between a distribution of physical parameters (e.g., bond lengths and bond angles) and the resulting distribution in g values, and examine the effects of correlation (e.g., depending on the microscopic physical origin of the distribution in g values, the distribution of g_1 may be correlated with the distribution of g_2, etc.), on the observed line shape. This kind of study is currently being done by G. E. Peterson, C. R. Kurkjian, and A. Carnevale (e.g., Phys. and Chem. of Glasses 15, 52, 1974).

V. Use of Program

To insure maximum flexibility, the user must write his own program consisting mainly of appropriate calls to the subroutines CØNSTNT, SIGNAL, or FØUR (which may be obtained by attaching the file REHFØRVITKØ, ID = VITKØ) followed by desired plotting calls.

The call to CØNSTNT must come before all other calls since this reads in input data from NAMELIST VITKØ and also supplies the constants necessary for subsequent calculations. Required input parameters including computer name, conventional symbol, and appropriate units are specified in Table I.

To calculate χ', χ'', $\frac{d\chi'}{dH}$, or $\frac{d\chi''}{dH}$ simply specify the appropriate value of IFUNC in NAMELIST VITKØ and CALL SIGNAL (XPLØT, YPLØT, NPTS, FAC). The user must specify

1. NPTS - the number of equidistant points at which the signal will be calculated, currently limited by dimension statements within REHFØRVITKØ not to exceed 501 points.

TABLE I

Name List Variables

1. G1 $\equiv g_1$
2. G2 $\equiv g_2$
3. G3 $\equiv g_3$

   where g_1, g_2, g_3 are the eigenvalues of the g tensor. Because calculation of a powder line shape involves a mapping from "g space" to "H space" we have arbitrarily assumed

   $$g_3 > g_2 > g_1 \ .$$

   It is essential that the user follow this same convention in ordering g values.
4. TONE $\equiv T_1$ (sec) where T_1 characterizes the spin-lattice relaxation of the single crystal line shape and is assumed isotropic.
5. DELTAH $\equiv \Delta H_{1/2}$ (gauss), full width at half maximum of the single crystal line shape which has been assumed to be Lorentzian.
6. DELTAHS $\equiv \Delta H^*_{1/2}$ (gauss), the full width at half maximum of a Gaussian inhomogeneous distribution.
7. H1STAR $\equiv H^*_1$ (gauss), amplitude of the rotating component of the microwave field, i.e., the cavity field H_x is given by

   $$H_x = 2H^*_1 \cos \omega t \ .$$
8. GNU $\equiv \nu$(Hz), the microwave frequency.
9. IFUNC $\equiv$ 1 Gaussian Shape Function = Absorption Case

   2 Gaussian Shape Function = Dispersion Case

   3 Gaussian (Derivative) = Absorption Case

   4 Gaussian (Derivative) = Dispersion Case

   5 Powder Lineshape = Absorption Case

   6 Powder Lineshape = Dispersion Case

   7 Powder Lineshape (Derivative) = Absorption Case

   8 Powder Lineshape (Derivative) = Dispersion Case

NOTE: When using SUBRØUTINE FØUR, IFUNC should be 1, 2, 5, or 6 only. The "derivative" signals may be obtained by specifying the appropriate Fourier coefficient.

2. FAC - a parameter specifying the initial and final points of the magnetic field interval, H_i and H_f, over which the signal is to be calculated. For the powder line shape case

$$H_i = H_3 - FAC \times (H_1 - H_3) \tag{21a}$$

$$H_f = H_1 + FAC \times (H_1 - H_3) \tag{21b}$$

where $H_j = \frac{h\nu}{g_j \beta}$.

For the Gaussian case

$$H_i = H_o - FAC \times (\tfrac{1}{2} \Delta H^*_{1/2}) \tag{22a}$$

$$H_f = H_o + FAC \times (\tfrac{1}{2} \Delta H^*_{1/2}) \tag{22b}$$

where the Gaussian is centered at a field H_o (in gauss) and is of full width at half maximum $\Delta H^*_{1/2}$ (in gauss).

SIGNAL returns two arrays, both of which must be dimensioned in the calling program.

1. XPLØT - a user dimensioned array, of dimension NPTS or greater, containing the magnetic field values, in gauss, at which the signal was calculated.
2. YPLØT - a user dimensioned array, of dimension NPTS or greater, containing the calculated signal. YPLØT (I) is the signal amplitude at the magnetic field given by XPLØT (I).

If one wishes to calculate the Fourier series expansion of the field modulated signal, i.e.,

$$(H + H_m \sin \omega_m t) = \frac{1}{2} a_o + \sum_{N=1}^{\infty} \left[a_N \sin (N \omega_m t) + b_N \cos (N \omega_m t)\right] \tag{23}$$

the appropriate call is CALL FØUR (XPLØT,YPLØT,NPTS,IFLAG,HSUBM,ØMEGAM,N,FAC). The user must specify:

1. NPTS - description same as for SIGNAL[5]
2. IFLAG - if IFLAG = 1, YPLØT = a_N

   IFLAG = 2, YPLØT = b_N

   where a_N, b_N are defined as in Eq. 23.
3. HSUBM - the amplitude of the modulation field, H_m, in gauss
4. ØMEGAM - the modulation frequency, ω_m, in Hz. Note that ØMEGAM should not affect the calculation and may be left at some arbitrary value, say 1 kHz.
5. N - specifies Fourier coefficient of order N.
6. FAC - description same as for SIGNAL.

FØUR returns two arrays, XPLØT and YPLØT, the descriptions of which are the same as for SIGNAL.

To verify that we have correctly programmed the basic lineshape calculations, we have hand digitized a powder lineshape for the m-complex in GeO_2 as calculated by T. L. Purcell,[6] and compared it with our own calculation. Since both calculations use identical lineshape equations and input parameters, the computed signals should agree. This comparison is shown in Figure 1.[7]

---

[5]FOUR calculates I(H) at NP points, does a cubic spline fit to I(H) and then calculates the Fourier coefficients at NPTS. NP is a parameter internal to the program and has a current default value of 101. For further discussion of NP, as well as some subleties about the interval over which the calculation is performed, the user should consult Appendix II.

[6]T. L. Purcell, Thesis, University of Tennessee, 1968.

[7]Figures 1 and 2 are indicative of program capabilities when used with DISSPLA graphical software on a Tektronix 4014 Intercom Terminal tied into a CDC 6600 computer.

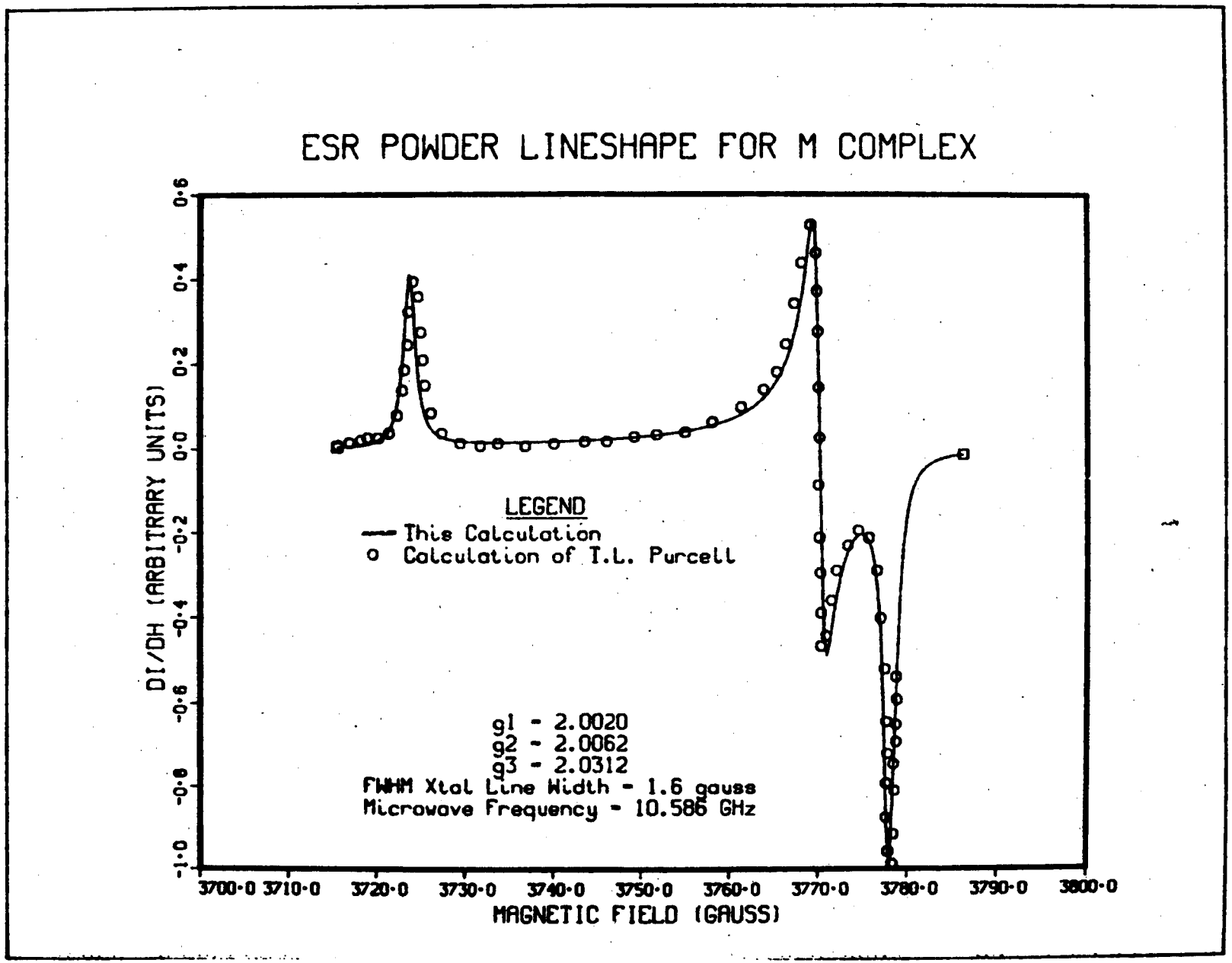

ESR POWDER LINESHAPE FOR M COMPLEX
DI/DH (ARBITRARY UNITS)
0·6
0·4
0·2
0·0
-0·2
-0·4
-0·6
-0·8
-1·0
3700·0
3710·0
3720·0
3730·0
3740·0
3750·0
3760·0
3770·0
3780·0
3790·0
3800·0
MAGNETIC FIELD (GAUSS)
LEGEND
This Calculation
Calculation of T.L. Purcell
g1 - 2.0020
g2 - 2.0062
g3 - 2.0312
FWHM Xtal Line Width - 1.6 gauss
Microwave Frequency - 10.586 GHz

Figure 1

Given the inaccuracies involved in hand digitizing a published figure, and the fact that the only adjustable parameter in the comparison was the overall signal amplitude, the agreement is found to be quite satisfactory.

Figure 2 illustrates the effect of overmodulation on an ESR power signal, in this case the E'_1 center in SiO_2. The first derivative calculation and the exact lineshape calculation for $H_m = 10^{-6}$ gauss given identical results, as well they should (note the solid line actually gives the result of both of these calculations at some 201 points; the squares are merely labels and are placed at every tenth point). However, as one increases the modulation field, to one gauss say, the first derivative calculation is no longer sufficient for computing lineshapes, and one must do the exact calculation as given by the dashed curve.

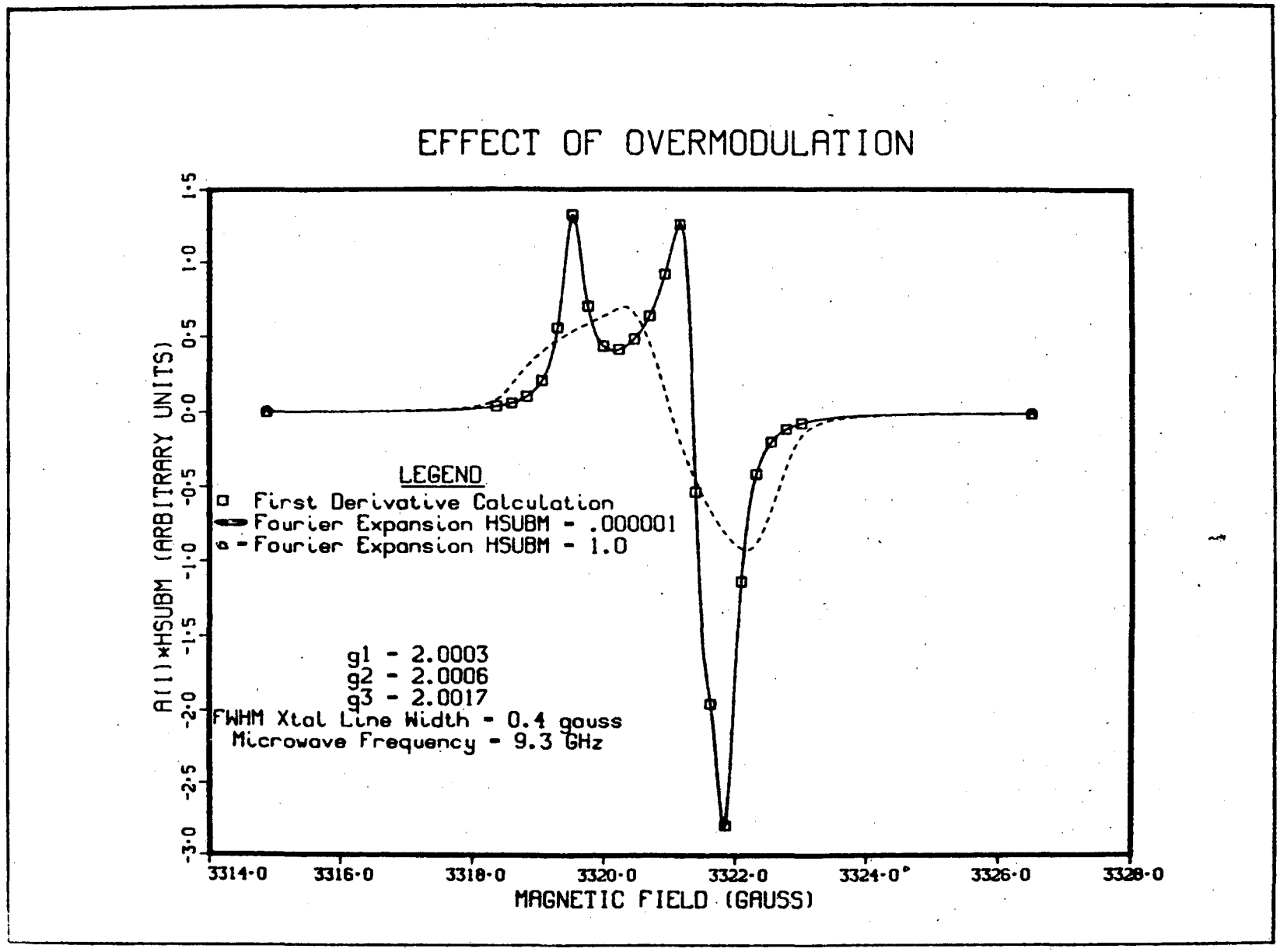

EFFECT OF OVERMODULATION
A(1)*HSUBM (ARBITRARY UNITS)
1·5
1·0
0·5
0·0
-0·5
-1·0
-1·5
-2·0
-2·5
-3·0
3314·0
3316·0
3318·0
3320·0
3322·0
3324·0
3326·0
3328·0
MAGNETIC FIELD (GAUSS)
LEGEND
First Derivative Calculation
Fourier Expansion HSUBM - .000001
Fourier Expansion HSUBM - 1.0
g1 - 2.0003
g2 - 2.0006
g3 - 2.0017
FWHM Xtal Line Width - 0.4 gauss
Microwave Frequency - 9.3 GHz

Figure 2

APPENDIX I

We shall try to illustrate some of the mathematics behind the program by treating the somewhat simpler example of a spin 1/2 system possessing axial symmetry.

A. <u>The Effective g Value</u>

For a totally anisotropic g

$$\mathcal{H}_{Zeeman} = \bar{S}\cdot\bar{g}\cdot\bar{H} = \beta\left[g_{pp} H_p S_p + g_{qq} H_q S_q + g_{rr} H_r S_r\right] \tag{1}$$

where the $\hat{p}$, $\hat{q}$, $\hat{r}$ are the principal axes of the g tensor; H_p, H_q, H_r are the components of the applied magnetic field along these axes. S_p, S_q, S_r are the usual spin operators.

For the case of axial symmetry let us

1. Call the unique axis $\hat{r}$
2. Choose H to lie in the pr plane, and to make an angle Θ with respect to the $\hat{r}$ axis (see Fig. A-I-1a)
3. Let $g_{pp} = g_{qq} = g_{\perp}$; $g_{rr} = g_{||}$
   then Eq. (1) becomes

$$\mathcal{H} = \beta H\left[g_{\perp}\sin\Theta\, S_p + g_{||}\cos\Theta\, S_r\right] \tag{2}$$

One way to diagonalize this Hamiltonian is to transform to a new coordinate system $\hat{p}'$, $\hat{q}'$, $\hat{r}'$ that is obtained by rotating $\hat{p}$, $\hat{q}$, $\hat{r}$ about the $\hat{q}$ axis through an angle ϕ (see Fig. A-I-1b) defined by

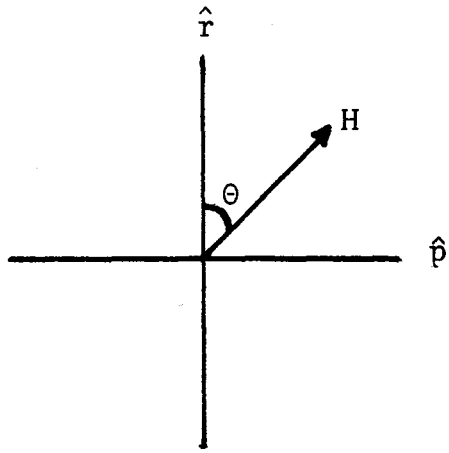


Figure A-I-1a

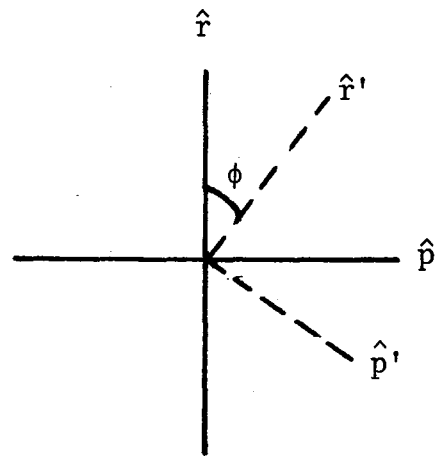


Figure A-I-1b

$$\tan \phi = \frac{g_{\perp}}{g_{||}} \frac{\sin \Theta}{\cos \Theta} = \frac{g_{\perp}}{g_{||}} \tan \Theta \tag{3}$$

then

$$S_p = \cos \phi \, S_{p'} + \sin \phi \, S_{r'} \tag{4a}$$

$$S_q = S_{q'} \tag{4b}$$

$$S_r = -\sin \phi \, S_{p'} + \cos \phi \, S_{r'} \tag{4c}$$

where

$$\sin \phi = \frac{g_{\perp} \sin \Theta}{g}; \quad \cos \phi = \frac{g_{||} \cos \Theta}{g}$$

and

$$g = \sqrt{g_{||}^{2} \cos^2 \Theta + g_{\perp}^{2} \sin^2 \Theta}$$

Substituting Eq. (4) into Eq. (2) we get

$$= g\beta H\, S_{r'} \tag{5}$$

For a spin 1/2 system this gives two energy levels

$$E\left(+\frac{1}{2}\right) = \frac{1}{2}\, g\beta H; \quad E\left(-\frac{1}{2}\right) = -\frac{1}{2}\, g\beta H \tag{6}$$

The resonance condition is then simply given by

$$h\nu = E\left(+\frac{1}{2}\right) - E\left(-\frac{1}{2}\right) = g\beta H \tag{7}$$

Thus g is the effective g value and depends on the orientation of the magnetic field with respect to the defect axes.

B. Orientation Dependence of Transition Probability

Here we follow the treatment of B. Bleany (Proc. Phys. Soc. A 75, 621, 1960).

Let us apply some linearly polarized microwave field, given by $H_1 \cos \omega t$. Further, let H_1 make an angle η with the $\hat{q}$ axis and let its projection on the pr plane make an angle Θ' with respect to the $\hat{r}$ axis.

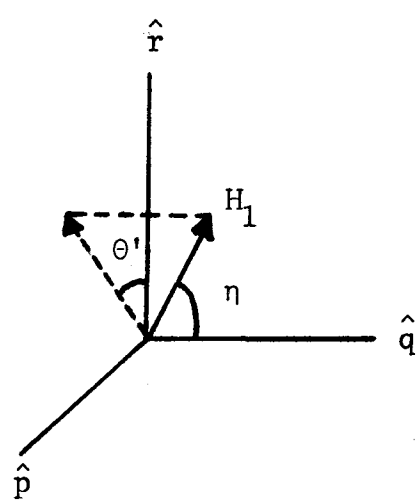


Figure A-I-1c

(See Figure A-I-1c.) Then

$$H_{1p} = H_1 \sin \eta \sin \Theta' \tag{8}$$

$$H_{1q} = H_1 \cos \eta$$

$$H_{1r} = H_1 \sin \eta \cos \Theta'$$

Then the perturbing Hamiltonian, $\mathcal{H}_{RF}$, due to the application of the microwave field is given by

$$\mathcal{H}_{RF} = \frac{\beta H_1}{2}\left(g_\perp \sin\eta \sin\Theta' \, S_p + g_\perp \cos\eta \, S_q + g_{||} \sin\eta \cos\Theta' \, S_r\right) \tag{9}$$

where the factor of $\frac{1}{2}$ comes from resolving $\cos\omega t$ into $\frac{1}{2}\left(e^{i\omega t} + e^{-i\omega t}\right)$ and retaining the appropriate component.

Using Eq. (4) we transform to the $\hat{p}'$, $\hat{q}'$, $\hat{r}'$ coordinate system and get

$$\mathcal{H}_{RF} = \frac{\beta H_1}{2}\left\{\frac{g_\perp g_{||}}{g} \sin\eta \sin(\Theta-\Theta') \, S_{p'} + g_\perp \cos\eta \, S_{q'} + \frac{\sin\eta}{g} \times \left[g^2_\perp \sin\Theta' \sin\Theta + g^2_{||} \cos\Theta' \cos\Theta\right] S_{r'}\right\} \tag{10}$$

Since we are interested in transitions only we may ignore the term in $S_{r'}$, which corresponds to a shift in spin energy levels due to the application of the microwave field. Introducing the standard raising and lowering operators, i.e.,

$$S_+ = \left(S_{p'} + iS_{q'}\right) \; ; \quad S_- = \left(S_{p'} - iS_{q'}\right) \tag{11}$$

Eq. (10) becomes

$$\mathcal{H}_{RF} \simeq \frac{\beta H_1}{2}\left\{\left[\frac{g_\perp g_{||}}{2g} \sin\eta \sin(\Theta-\Theta') - i\,\frac{g_\perp}{2} \cos\eta\right] S_+ + \left[\frac{g_\perp g_{||}}{2g} \sin\eta \sin(\Theta-\Theta') + i\,\frac{g_\perp}{2} \cos\eta\right] S_-\right\} \tag{12}$$

The transition probability, W, between the states $|1/2\rangle$ and $|-1/2\rangle$ is given by Fermi's golden rule,

$$W = \frac{2\pi}{\hbar}\,\rho\Big(|1/2\rangle\Big)\,|\langle 1/2|\mathscr{H}_{RF}|-1/2\rangle|^2 \tag{13}$$

where $\rho\Big(|1/2\rangle|\Big)$ is the density of final states per unit energy. If we assume that $\rho\Big(|1/2\rangle|\Big)$ dose not depend on the orientation of the defect (this assumption is not always valid, e.g., for homogeneous line broadening $\rho \sim 1/T_1$, so that if T_1 is anisotropic ρ will be anisotropic) then the orientational dependence of W is given by the orientational dependence of $|\langle 1/2|\mathscr{H}_{RF}|-1/2\rangle|^2$. Now

$$|\langle 1/2|\mathscr{H}_{RF}|-1/2\rangle|^2 \simeq \frac{\beta^2 H_1^{\,2}}{16}\left\{\left(\frac{g_\perp g_{||}}{g}\right)^2 \sin^2\eta\,\sin^2(\Theta'-\Theta) + g_\perp^{\,2}\cos^2\eta\right\} \tag{14}$$

The common experimental configuration is to have H_1 perpendicular to the applied static field, i.e., $\Theta'-\Theta = \pi/2$, in this case

$$|\langle 1/2|\mathscr{H}_{RF}|-1/2\rangle|^2 \sim \left\{\left(\frac{g_\perp g_{||}}{g}\right)^2 \sin^2\eta + g_\perp^{\,2}\cos^2\eta\right\} \tag{15}$$

In a powder all values of η will be equally likely, therefore

$$|\langle 1/2|\mathscr{H}_{RF}|-1/2\rangle|^2 \sim g_\perp^{\,2}\left(\frac{g_{||}^2}{g^2} + 1\right) \tag{16}$$

Now g varies monotonically from $g_{||}(\Theta = 0)$ to $g_\perp$ $(\Theta = \pi/2)$. Thus the maximum variation in transition probability as one scans the lineshape is

$$\frac{W(\Theta=0)}{W(\Theta=\pi/2)} = \frac{2g_{\perp}^{2}}{g_{||}^{2} + g_{\perp}^{2}} \tag{17}$$

Silsbee (J. App. Phys. 32, 1459, 1961) has found that the g values for the E_1' defect are

$$g_1 = 2.00029 \; ; \quad g_2 = 2.00049 \; ; \quad g_3 = 2.00176 \tag{18}$$

If we approximate this by an axial g tensor, e.g.,

$$g_{\perp} = 2.00029 \qquad g_{||} = 2.00176 \tag{19}$$

We find that

$$\left.\frac{W(\Theta=0)}{W(\Theta=\pi/2)}\right]_{max} = 0.99927 \tag{20}$$

i.e., the transition probability varies by less than 0.1% across the line shape. Therefore, it is reasonable to neglect this variation.

C. The Powder Line Shape

As we have stated earlier, the powder line shape I(H) is given by

$$I(H) = \frac{1}{4\pi}\int_{4\pi \text{ solid angle}} Y(H-H')\, d\Omega = \frac{1}{4\pi}\int_{0}^{2\pi} d\phi \int_{0}^{\pi} Y(H-H') \sin\Theta \, d\Theta \tag{21}$$

where Y(H-H') is a single crystal line shape.

For axial symmetry this reduces to

$$I(H) = \frac{1}{2} \int_0^{\pi} Y(H-H') \sin \Theta \, d\Theta \tag{22}$$

further, the defect cannot tell whether H is parallel or anti-parallel to the $\hat{r}$ axis, therefore,

$$I(H) = \int_0^{\pi/2} Y(H-H') \sin \Theta \, d\Theta \tag{23}$$

What we would like to do is to transform this from an integral over Θ, to one over H'. To do this, recall from Eq. (7) that

$$H' = \frac{h\nu}{\beta} \frac{1}{\left(g_{||}^{2} \cos^2 \Theta + g_{\perp}^{2} \sin^2\Theta\right)^{1/2}} \tag{24}$$

Therefore,

$$I(H) = \int_{H' = H_{||}}^{H' = H_{\perp}} Y(H-H') \sin \Theta \left(\frac{d\Theta}{dH'}\right) dH' \tag{25}$$

where $H_{||} = \frac{h\nu}{\beta g_{||}}$ and $H_{\perp} = \frac{h\nu}{\beta g_{\perp}}$.

But from Eq. (24) we see that

$$\sin \Theta \frac{d\Theta}{dH'} = \frac{\alpha^2}{H'^3 \left[g_{||}^{2} - g_{\perp}^{2}\right] \cos \Theta}$$

$$= \frac{\alpha^2}{H'^3 \left[g_{||}^{2} - g_{\perp}^{2}\right]} \sqrt{\frac{\left[g_{||}^{2} - g_{\perp}^{2}\right] H'^2}{\alpha^2 - g_{\perp}^{2} H'^2}} \tag{26}$$

where $\alpha = \frac{h\nu}{\beta}$.

Simplifying the above expression and defining the shape function S(H') gives, for $g_{||} > g_{\perp}$,

$$S(H') = \frac{\alpha^2}{H'^2 g_{\perp}\left(g_{||}^2 - g_{\perp}^2\right)^{1/2}\left(H_{\perp}^2 - H'^2\right)^{1/2}} \qquad H_{||} \leq H' \leq H_{\perp} \tag{27}$$

$$S(H') = 0 \qquad \text{elsewhere}$$

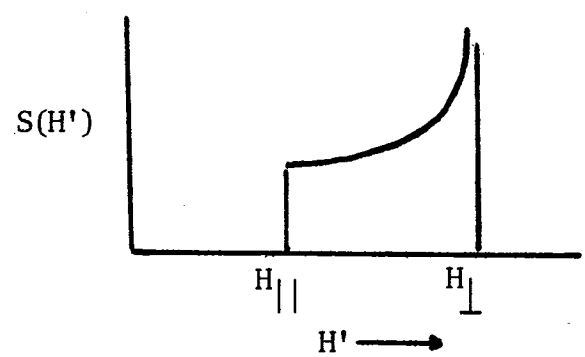
A crude feel for the behavior of S(H') can be gotten by examining S(H') in the limit of small g anisotropy.

In this case $H_{||} \sim H_{\perp} \sim H'$, and

$$S(H') \sim \frac{1}{\left(H_{\perp}^2 - H'^2\right)^{1/2}} \qquad H_{||} \leq H' \leq H_{\perp} \tag{28}$$

Thus S(H') looks something like Figure A-I-2

Figure A-I-2



The divergence at $H' = H_{\perp}$ can be traced back to the factor sin Θ in Eq. (22). Physically it arises because the number of spins whose $\hat{r}$ axis make an angle Θ with respect to the applied field is proportional to sin Θ dΘ; thus the largest number of spins have $\hat{r}$ perpendicular to H and hence are characterized by $g_{\perp}$.

The powder line shape is then given by

$$I(H) = \int_{H_{||}}^{H_{\perp}} S(H')\ Y(H-H')\ dH' \tag{29}$$

The procedure is very similar for the case of complete g anisotropy, only S(H') becomes somewhat messier and the underlying ideas are obscured by the algebra.

APPENDIX II

This suite of codes performs two tasks, one accomplished by calling CØNSTNT and then SIGNAL and the other accomplished by calling CØNSTNT and then FØUR.

Subroutine CØNSTNT reads in the data (see comment cards at the beginning of each routine for more details) and from this calculates certain quantities which are stored in common block /VITKØ1/. (The method of data transmission - in this case NAMELIST statements - will have to be modified at most installations.)

Subroutine SIGNAL returns an array of values

$$I(H) = \int_0^\infty SY(H')dH' = \begin{cases} \int_0^\infty S(H')\ Y(H-H')\ dH' \\ \int_0^\infty S(H')\ \frac{d}{dH}\ Y(H-H')\ dH' \end{cases}$$

the selection for SY(H') depending upon an input parameter IFUNC (see comments at the beginning of SIGNAL for more details).

Subroutine SIGNAL calls subroutine CØNVØL which determines the useful finite subintervals of $[0,\infty)$ for calculating $\int_0^\infty SY(H')\ dH'$. CØNVØL then calls on a numerical integration routine GAUS8 to compute the finite integrals. More detailed information on CØNVØL appears in this appendix.

The CØNSTNT, FØUR sequence calculates the Fourier coefficients of $G(t) = I(H + H_m \sin \omega_m t)$ as a function of H. Details of FØUR are given in this appendix.

Figures A-II-1 and A-II-2 show the subroutine linkages. Figure A-II-3 is a list of the variables as they are used in the codes.

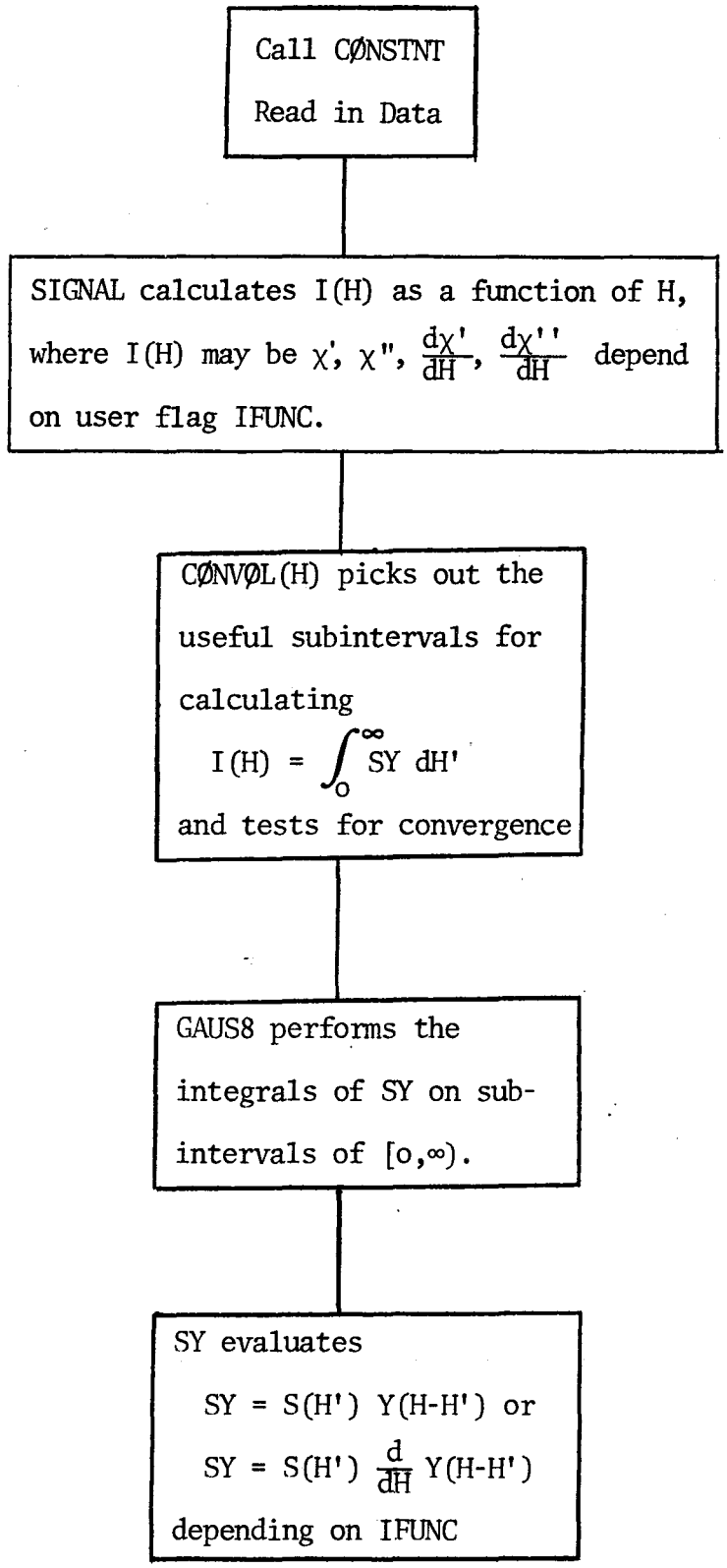

Call CØNSTNT
Read in Data
SIGNAL calculates I(H) as a function of H, where I(H) may be χ', χ'', $\frac{d\chi'}{dH}$, $\frac{d\chi''}{dH}$ depend on user flag IFUNC.
CØNVØL(H) picks out the useful subintervals for calculating
$I(H) = \int_0^\infty SY\, dH'$
and tests for convergence
GAUS8 performs the integrals of SY on sub-intervals of $[o,\infty)$.
SY evaluates
$SY = S(H')\, Y(H-H')$ or
$SY = S(H')\, \frac{d}{dH}\, Y(H-H')$
depending on IFUNC

Figure A-II-1

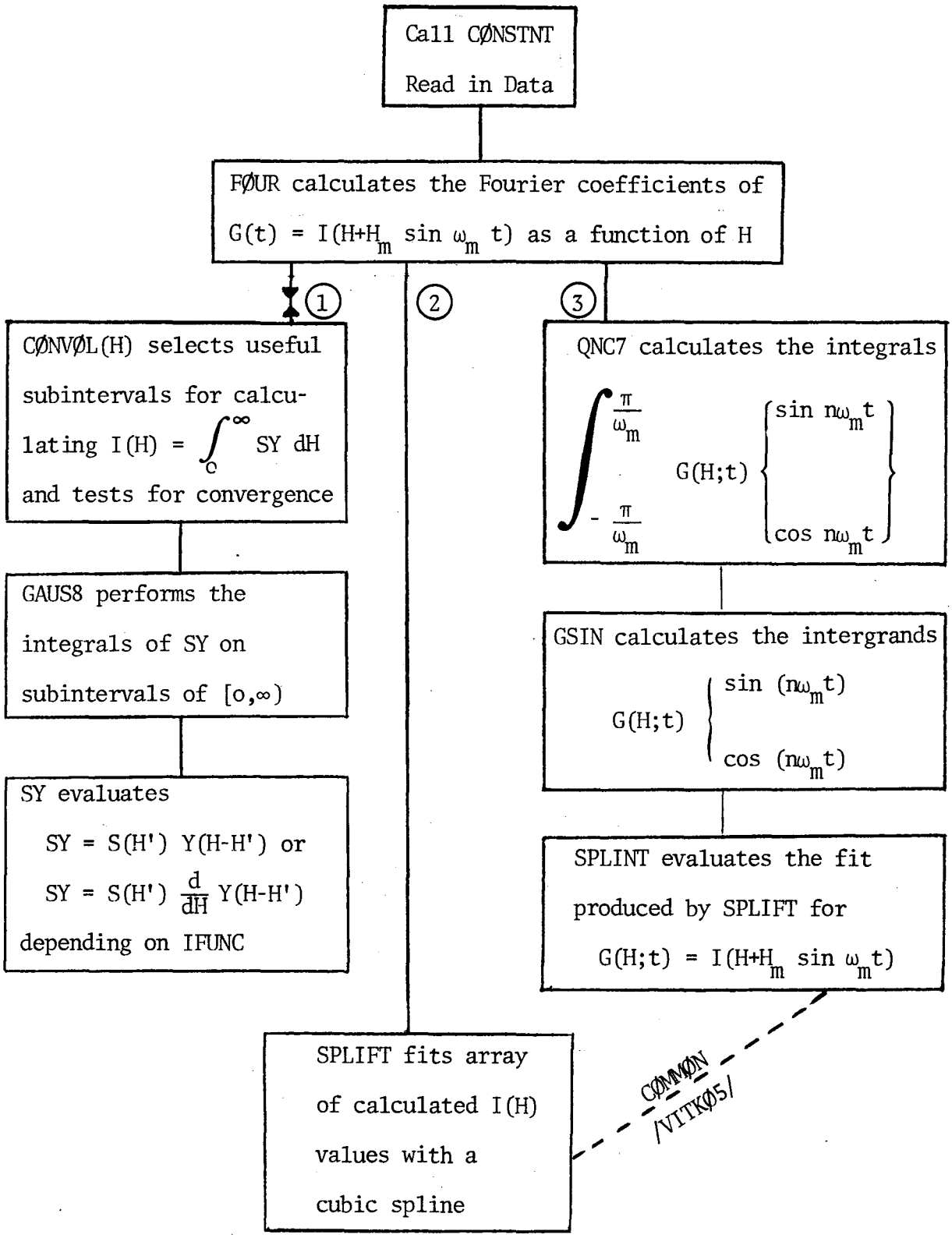

Call CØNSTNT
Read in Data
FØUR calculates the Fourier coefficients of
G(t) = I(H+H_m sin ω_m t) as a function of H
1
2
3
CØNVØL(H) selects useful subintervals for calculating I(H) = ∫_0^∞ SY dH and tests for convergence
GAUS8 performs the integrals of SY on subintervals of [o,∞)
SY evaluates
SY = S(H') Y(H-H') or
SY = S(H') d/dH Y(H-H')
depending on IFUNC
QNC7 calculates the integrals
∫_{-π/ω_m}^{π/ω_m} G(H;t) {sin nω_m t, cos nω_m t}
GSIN calculates the intergrands
G(H;t) {sin (nω_m t), cos (nω_m t)}
SPLINT evaluates the fit produced by SPLIFT for
G(H;t) = I(H+H_m sin ω_m t)
SPLIFT fits array of calculated I(H) values with a cubic spline
CØMMØN /VITKØ5/

Figure A-II-2

VARIABLE LIST

HPRIM $= H'$

HALF $= \frac{1}{2}\Delta H_{1/2}$

HLFSTR $= \frac{1}{2}\Delta H^*_{1/2}$

BETA $= \beta$

PLANC $= \hbar$

ØVRHALF $= \dfrac{1}{\frac{1}{2}\Delta H_{1/2}}$

ØHLFSTR $= \dfrac{1}{\frac{1}{2}\Delta H^*_{1/2}}$

GAMMA $= g_3\ \beta/\hbar = \gamma$

PI $= \pi$

XLØG2 $= \ln(2)$

PILØG $= \dfrac{\ln(2)^{1/2}}{\pi} \Big/ \left(\frac{1}{2}\Delta H^*_{1/2}\right)$

TEMP $= (2\pi\hbar\nu)/\beta$

HZERØ $= H_0 = (2\pi\hbar\nu)/(\beta g_3)$

HØNE $= H_1 = (2\pi\hbar\nu)/(\beta g_1)$

H2 $= H_2 = (2\pi\hbar\nu)/(\beta g_2)$

H3 $= H_3 = (2\pi\hbar\nu)/(\beta g_3)$

H1S $= H_1^{\ 2}$

H2S $= H_2^{\ 2}$

H3S $= H_3^{\ 2}$

H1STAR $= H_1^{\ *}$

S $= \gamma(H_1^{\ *})^2\ T_1 \Big/ \left(\frac{1}{2}\Delta H_{1/2}\right)$

HSUBM $= H_m$

ØMEGAM $= \omega_m$

DELTAH $= \Delta H_{1/2}$

DELTAHS $= \Delta H^*_{1/2}$

SY(HPRIM) $= S(H')\ Y(H-H')$

GSIN(X) $= I\left(H + H_m \sin(\omega_m x)\right) \sin(n\omega_m x)$

GCØS(X) $= I\left(H + H_m \sin(\omega_m x)\right) \cos(n\omega_m x)$

Figure A-II-3

Subroutine CØNVØL (and CNVLERR)

Subroutine CØNVØL(H) calculates the function

$$I(H) = \int_0^\infty S(H')\ Y(H-H')\ dH'$$

or

$$I(H) = \int_0^\infty S(H')\ \frac{d}{dH}\ Y(H-H')\ dH'$$

(Which of the two is calculated and the definitions of S and Y depends upon IFUNC.)

The particular definitions of S and Y assure us that the products

$$S(H')\ Y(H-H')$$

or

$$S(H')\ \frac{d}{dH}\ Y(H-H')$$

approach zero rapidly outside of certain intervals. The intervals are usually so small that a standard Laguerre integration formula for the interval $[0,\infty)$ would have to be of very high degree to detect the substantially non-zero portion of the curve.

Thus, the method used for evaluating the integrals depends on specific knowledge of the intervals over which $S \cdot Y$ is active. We perform finite integrals in these known regions and check for convergence by evaluating integrals on nearby intervals and comparing with the original interval. The specifics can be given as:

1. If $1 \leq IFUNC \leq 4$, let

$$DIF = \frac{1}{2} \Delta H^*_{1/2}$$

$$HLØ = H_0 - FAC \cdot DIF \qquad FAC = 3.0$$

$$HUP = H_0 + FAC \cdot DIF$$

$$HLØLØ = HLØ - XFAC \cdot DIF \qquad XFAC = 10.0$$

$$HUPUP = HUP + XFAC \cdot DIF$$

2. If $5 \leq IFUNC \leq 8$, let

$$DIF = (H_1 - H_3) + .001$$

$$HLØ = H_3 - FAC * DIF \qquad FAC = 0.1$$

$$HLØLØ = \text{Maximum } (0, HLØ - XFAC * DIF) \qquad XFAC = 20.0$$

$$HUP = H_1 + FAC * DIF$$

$$HUPUP = \text{Minimum } (10^{20}, HUP + XFAC * DIF)$$

NOTE: The values of FAC, XFAC, and the upper bound were chosen after experimenting with the code to achieve convergence. They are not calculated bounds.

3. Calculate

$$\text{QLEFT} = \int_{\text{HLØLØ}}^{\text{HLØ}} \text{SY}$$

$$\text{QMIDDLE} = \int_{\text{HLØ}}^{\text{HUP}} \text{SY}$$

$$\text{QRIGHT} = \int_{\text{HUP}}^{\text{HUPUP}} \text{SY}$$

by calling an adaptive 8-point Gaussian routine GAUS8.

4. If $|\text{QLEFT} + \text{QRIGHT}| \leq (.01)\cdot|\text{QMIDDLE}|$, then we accept the approximation

$$\int_0^\infty \text{SY} = \text{QLEFT} + \text{QMIDDLE} + \text{QRIGHT}$$

If $|\text{QLEFT} + \text{QRIGHT}| \geq (.01)\cdot|\text{QMIDDLE}|$,

then replace HLØ by HLØ - DIF

HUP by HUP + DIF

HLØLØ = HLØ - XFAC * DIF

HUPUP = HUP + XFAC * DIF

and return to step 3 for a possible second and third try.

5. If the test in step 4 fails three times, CNVLERR is called, which prints out the information about the three tries and calls a halt to program execution. If this happens, call the author.

Subroutine FØUR

Let $G(t) = I(H + H_m \sin \omega_m t)$

and expand G in a Fourier series

$$G(t) = \frac{1}{2} a_0 + \sum_{N=1}^{\infty} \left[a_N \sin (N \omega_m t) + b_N \cos (N \omega_m t) \right]$$

where

$$a_N = \frac{\omega_m}{\pi} \int_{-\frac{\pi}{\omega_m}}^{\frac{\pi}{\omega_m}} G(t) \sin (N \omega_m t)\, dt$$

$$N = 0, 1, 2, \ldots$$

and

$$b_N = \frac{\omega_m}{\pi} \int_{-\frac{\pi}{\omega_m}}^{\frac{\pi}{\omega_m}} G(t) \cos (N \omega_m t)\, dt \; .$$

We are interested in obtaining values of a_N and b_N as functions of H over the range

$$\text{HSTART} \leq H \leq \text{HFINAL}$$

where, if IFUNC is 1 or 2,

$$\text{HSTART} = H_0 - \text{FAC}\left(\frac{1}{2}\,\Delta H^*_{1/2}\right)$$

$$\text{HFINAL} = H_0 + \text{FAC}\left(\frac{1}{2}\,\Delta H^*_{1/2}\right)$$

(FAC is an input variable to FØUR)

and, if IFUNC is 5 or 6,

$$\text{HSTART} = H_3 - \text{FAC} \cdot (H_1 - H_3)$$

$$\text{HFINAL} = H_1 + \text{FAC} \cdot (H_1 - H_3)\ .$$

The calling sequence for FØUR is

FØUR(HVEC,CØFVEC,IPTS,IFLAG,HSUBM,ØMEGAM,N,FAC)

The user is expected to input N, HSUBM = H_m, ØMEGAM = ω_m, IFLAG, and IPTS according to the rules below. Also, the user must supply dimensioned arrays HVEC, and CØFVEC of length at least IPTS. The routine will return

$$\text{HVEC(I)} = \text{HSTART} + (\text{I-1})\left(\frac{\text{HFINAL} - \text{HSTART}}{\text{IPTS} - 1}\right), \qquad \text{I} = 1, \ldots, \text{IPTS}$$

and

$$\text{CØFVEC(I)} = \begin{cases} a_N(\text{HVEC(I)})\ , & \text{if IFLAG} = 1 \\ b_N(\text{HVEC(I)})\ , & \text{if IFLAG} = 2 \end{cases} \qquad \text{I} = 1, \ldots, \text{IPTS.}$$

The integration of the intergrals which define the Fourier coefficients are carried out by an adaptive Newton-Cotes quadrature routine, QNC7. The procedure of QNC7 requires many evaluations of the integrands

$$G(t) \begin{Bmatrix} \sin\ (N\ \omega_m t) \\ \cos\ (N\ \omega_m t) \end{Bmatrix} = I(H + H_m \sin \omega_m t) \begin{Bmatrix} \sin\ (N\ \omega_m t) \\ \cos\ (N\ \omega_m t) \end{Bmatrix}$$

and therefore many evaluations of I(H) each of which is an integral itself. For the sake of efficiency, I(H) is evaluated NP (default for NP is 101) times over the range HSTART - $2H_m$ to HFINAL + $2H_m$ and these points are then fit by a cubic interpolatory spline (through subroutine SPLIFT). The fit is then used in the evaluation of the integrals for a_N and b_N. Since the integrals require evaluation of $I(H + H_m \sin \omega_m t)$, the fit certainly has to extend $\pm H_m$ beyond the range of H, which is of interest. The spline fit has an artificially produced zero second derivative at the end points. To keep this from having much effect on the fit near the end points of interest, the fitting points are extended to $\pm 2H_m$ beyond the range of H. This extension of length H_m on either end may be neither necessary nor sufficient, but it seems to work in practice.

The evaluation of I(H) over HSTART - $2H_m \leq H \leq$ HFINAL + $2H_m$ and the call to the spline fitting routine are carried out in subroutine FØUR. The evaluations of the fit are carried out in subroutine GSIN. The spline is fit to NP points (NP = 101 by default). To change the number of fitting points, the user may do the following:

1. Change common block VITKØ5 in FØUR and GSIN according to the rule
   CØMMØN/VITKØ5/XH(NP), YH(NP), YP(NP), YPP(NP), WH(3•NP), NP.

2. In the calling program, provide the common statement referred to in (1) and set the value of NP.

Subroutine FØUR then makes the calls to the adaptive Newton-Cote quadrature routine QNC7. The calling sequence for QNC7 requires that the integrand, which is computed by GSIN, be a function of only one variable. Therefore, FØUR passes copies of the arguments H, HSUBM, N, and ØMEGAM to GSIN through the common blocks VITKØ2 and VITKØ4.

UNLIMITED RELEASE

INITIAL DISTRIBUTION

Lawrence Livermore Laboratories
University of California
P. O. Box 808
Livermore, CA 94550
Attn: M. Schwab/M. J. Moran, L-503

H. T. Weaver, 2354
F. L. Vook, 5110
J. E. Schirber, 5150
E. P. EerNisse, 5112
K. L. Brower, 5112
P. M. Richards, 5132
C. J. MacCallum, 5231
T. B. Cook, 8000; Attn: L. Gutierrez, 8100
C. H. DeSelm, 8200
W. C. Scrivner, 8400
B. F. Murphey, 8300; Attn: R. H. Meinken, 8310
J. L. Wirth, 8340
J. F. Barham, 8360
T. S. Gold, 8320
R. Y. Lee, 8322
R. E. Huddleston, 8322 (10)
G. W. Anderson, Jr., 8330; Attn: J. E. Marion, 8332
C. W. Robinson, 8333
P. D. Gildea, 8335
W. Bauer, 8334
P. L. Mattern, 8334
J. Vitko, Jr., 8334 (10)
C. F. Coll, III, 8341
J. J. Iannucci, 8341
W. D. Wilson, 8341
E. H. Barsis, 8342
T. A. Dellin, 8342
C. Hartwig, 8342
Technical Publications and Art Division 8265, for TIC (2)
F. J. Cupps, 8265/Library Systems & Technical Processes Div., 3141
Library Systems & Technical Processes Division, 3141 (4)
Library & Security Classification Division, 8266-2 (5)

FOLLOWING ARE THE IMAGES FROM THE MICROFICHE ATTACHED TO THE BACK OF THIS DOCUMENT

```
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
***   UNCLASSIFIED   ***          REHVT,T300,KUNC.     REHFORVITKO CY=46   AND MICROFICHE
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    02/05/76      13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
          PROGRAM TEST       74/74   OPT=1                          FTN 4.5+R406       02/05/76  13.03.47       PAGE     2

VARIABLES      SN  TYPE           RELOCATION
 6305  XPLOT      REAL       ARRAY                  REFS          2         14         15
 6370  YPLOT      REAL       ARRAY                  REFS          2         14         15

FILE NAMES         MODE
    0  INPUT
 2041  OUTPUT
    0  TAPE1
 4102  TAPE50

EXTERNALS           TYPE    ARGS       REFERENCES
       CONSTNT                0              7
       ENDPLOT                0             31
       EOF        REAL        1              8
       EXIT                   0             32
       FOUR                   8             26
       PRMODE                 0              3
       QUICK                  9             15         27
       SIGNAL                 4             14

STATEMENT LABELS              DEF LINE   REFERENCES
 6156  10                        5         29
    0  11           INACTIVE     9          8
 6214  15                       28         21         22         23         24
 6215  20                       30          8

STATISTICS
  PROGRAM LENGTH                511B      329
  BUFFER LENGTH                6144B     3172
```

```
      DIF = HLFSTR                                                  SIGNAL     42
      HSTART = HZERO - FAC*DIF                                      SIGNAL     43
      HFINAL = HZERO + FAC*DIF                                      SIGNAL     44
      GO TO 15                                                      SIGNAL     45
   10 CONTINUE                                                      SIGNAL     46
      DIF = HONE - H3                                               SIGNAL     47
      HSTART = H3 - FAC*DIF                                         SIGNAL     48
      HFINAL = HONE + FAC*DIF                                       SIGNAL     49
   15 CONTINUE                                                      SIGNAL     50
      HINC  = (HFINAL - HSTART)/(NPTS-1)                            SIGNAL     51
      H =  HSTART -HINC                                             SIGNAL     52
      DO 20 J=1,NPTS                                                SIGNAL     53
      H = H + HINC                                                  SIGNAL     54
      XPLOT(J) = H                                                  SIGNAL     55
      YPLOT(J) = CONVOL(H)                                          SIGNAL     56
   20 CONTINUE                                                      SIGNAL     57
      RETURN                                                        SIGNAL     58
      END                                                           SIGNAL     59
```

SYMBOLIC REFERENCE MAP (R=3)

ENTRY POINTS		DEF LINE	REFERENCES
3	SIGNAL	1	74

VARIABLES		SN	TYPE	RELOCATION								
2	BETA		REAL	VITK01	REFS	48						
31	DELTAH		REAL	VITK01	REFS	48						
32	DELTAHS		REAL	VITK01	REFS	48						
54	DIF		REAL		REFS	59	60	64	65	DEFINED	58	63
0	FAC		REAL	F.P.	REFS	59	60	64	65	DEFINED	1	
25	GAMMA		REAL	VITK01	REFS	48						
0	GNU		REAL	VITK01	REFS	48						
15	G1		REAL	VITK01	REFS	48						
16	G2		REAL	VITK01	REFS	48						
17	G3		REAL	VITK01	REFS	48						
60	H		REAL		REFS	70	71	72	DEFINED	68	70	
4	HALF		REAL	VITK01	REFS	48						
56	HFINAL		REAL		REFS	67	DEFINED	60	65			
57	HINC		REAL		REFS	68	70	DEFINED	67			
26	HISTAR		REAL	VITK01	REFS	48						
27	HLFSTR		REAL	VITK01	REFS	48	58					
7	HONE		REAL	VITK01	REFS	48	63	65				
55	HSTART		REAL		REFS	67	68	DEFINED	59	64		
6	HZERO		REAL	VITK01	REFS	48	59	60				
12	H1S		REAL	VITK01	REFS	48						
10	H2		REAL	VITK01	REFS	48						
13	H2S		REAL	VITK01	REFS	48						
11	H3		REAL	VITK01	REFS	48	63	64				
14	H3S		REAL	VITK01	REFS	48						
21	IFUNC		INTEGER	VITK01	REFS	48	57					
61	J		INTEGER		REFS	71	72	DEFINED	69			
0	NPTS		INTEGER	F.P.	REFS	67	69	DEFINED	1			
30	OHLFSTR		REAL	VITK01	REFS	48						

```
        SUBROUTINE FOUR        74/74   OPT=1                          FTN 4.5+R406        02/05/76  13.03.47        PAGE      6

  LOOPS  LABEL     INDEX      FROM-TO     LENGTH    PROPERTIES
     25  8       * I            93 96        5B                EXT REFS
     65  110     * J           123 127      10B                EXT REFS
    117  20      * J           154 172      25B                EXT REFS

  COMMON BLOCKS   LENGTH      MEMBERS - BIAS NAME(LENGTH)
         VITK02        1                 0 CH      (1)
         VITK04        3                 0 CHSUBM (1)                  1 COMEGAM(1)                  2 NN      (1)
         VITK05      708                 0 XH      (101)             101 YH      (101)             202 YP      (101)
                                       303 YPP     (101)             404 WH      (303)             707 NP      (1)
         VITK01       27                 0 GNU     (1)                 1 PLANC   (1)                 2 BETA    (1)
                                         3 PI      (1)                 4 HALF    (1)                 5 OVRHALF(1)
                                         6 HZERO   (1)                 7 HONE    (1)                 8 H2      (1)
                                         9 H3      (1)                10 H1S     (1)                11 H2S     (1)
                                        12 H3S     (1)                13 G1      (1)                14 G2      (1)
                                        15 G3      (1)                16 S       (1)                17 IFUNC   (1)
                                        18 XLOG2   (1)                19 PILOG   (1)                20 TONE    (1)
                                        21 GAMMA   (1)                22 HISTAR  (1)                23 HLFSTR  (1)
                                        24 OHLFSTR(1)                 25 DELTAH  (1)                26 DELTAHS(1)

  STATISTICS
    PROGRAM LENGTH                 315B      205
    CM LABELED COMMON LENGTH      1343B      739
```

```
VARIABLES      SN  TYPE            RELOCATION
    2  BETA       REAL                VITK01       REFS        29
  176  CONVOL     REAL                            DEFINED      101
   31  DELTAH     REAL                VITK01       REFS        29
   32  DELTAHS    REAL                VITK01       REFS        29
    5  DIF        REAL                VITK03       REFS        43        57        58        60        61        68        69
                                                      70        71        93        94        95        96
                                                  DEFINED       54        67
    3  ERR        REAL                VITK03       REFS        43        75        83        85   DEFINED        52
    0  FAC        REAL                VITK03       REFS        43        57        60        68        69
                                                  DEFINED       55        65
   25  GAMMA      REAL                VITK01       REFS        29
    0  GNU        REAL                VITK01       REFS        29
   15  G1         REAL                VITK01       REFS        29
   16  G2         REAL                VITK01       REFS        29
   17  G3         REAL                VITK01       REFS        29
    0  H          REAL                VITK02       REFS        39   DEFINED        49
    4  HALF       REAL                VITK01       REFS        29
    0  HH         REAL                 F.P.        REFS        49        99   DEFINED         1
   26  HISTAR     REAL                VITK01       REFS        29
   27  HLFSTR     REAL                VITK01       REFS        29        67
  177  HLO        REAL                             REFS        58        70        75        83        93        95
                                                  DEFINED       57        68        93
  200  HLOLO      REAL                             REFS        59        75   DEFINED        58        59        70        95
    7  HONE       REAL                VITK01       REFS        29        54        60
  201  HUP        REAL                             REFS        61        71        83        85        94        96
                                                  DEFINED       60        69        94
  202  HUPUP      REAL                             REFS        62        85   DEFINED        61        62        71        96
    6  HZERO      REAL                VITK01       REFS        29        68        69
   12  H1S        REAL                VITK01       REFS        29
   10  H2         REAL                VITK01       REFS        29
   13  H2S        REAL                VITK01       REFS        29
   11  H3         REAL                VITK01       REFS        29        54        57
   14  H3S        REAL                VITK01       REFS        29
  203  ICOUNT     INTEGER                          REFS        91        92   DEFINED        73        91
  205  IERR       INTEGER                          REFS        75        76        77        83        84        85        86
   21  IFUNC      INTEGER             VITK01       REFS        29        53        79
    4  NPTS       INTEGER             VITK03       REFS        43
   30  OHLFSTR    REAL                VITK01       REFS        29
    5  OVRHALF    REAL                VITK01       REFS        29
    3  PI         REAL                VITK01       REFS        29
   23  PILOG      REAL                VITK01       REFS        29
    1  PLANC      REAL                VITK01       REFS        29
  210  Q          REAL                             REFS        90   DEFINED        87
  204  QLEFT      REAL                             REFS        75        87       101
  206  QMIDDLE    REAL                             REFS        83        89       101
  207  QRIGHT     REAL                             REFS        85        87       101
   20  S          REAL                VITK01       REFS        29
  211  TEST       REAL                             REFS        90   DEFINED        89
    2  TESTFAC    REAL                VITK03       REFS        43        89   DEFINED        88
   24  TONE       REAL                VITK01       REFS        29
    1  XFAC       REAL                VITK03       REFS        43        58        61        70        71        95        96
                                                  DEFINED       56        66
   22  XLOG2      REAL                VITK01       REFS        29

FILE NAMES         MODE
       OUTPUT      FMT                    WRITES       77        79
```

```
        FUNCTION GSIN        74/74   OPT=1                                  FTN 4.5+R406       02/05/76  13.03.47       PAGE     2

ENTRY POINTS      DEF LINE      REFERENCES
   14  GCOS            37          49
    4  GSIN             1

VARIABLES      SN  TYPE            RELOCATION
  100  ARG        REAL                                  REFS        43         45    DEFINED        41
    0  CHSUBM     REAL                  VITK04          REFS        32         41
    1  COMEGAM    REAL                  VITK04          REFS        32         35         39         41
  102  G          REAL                                  REFS        43         48
  103  GP       * REAL                                  REFS        43
  104  GPP      * REAL                                  REFS        43
   75  GSIN       REAL                                 DEFINED      48
    0  H          REAL                  VITK02          REFS        29         41
  105  KERR       INTEGER                               REFS        43         44
  101  NI         INTEGER                               REFS        43    DEFINED        42
    2  NN         INTEGER               VITK04          REFS        32         34         38
 1303  NP         INTEGER               VITK05          REFS        33         43
   77  S          REAL                                  REFS        48    DEFINED        35         39
  624  WH         REAL        ARRAY     VITK05          REFS        33
    0  X          REAL                   F.P.           REFS        35         39         41    DEFINED         1
    0  XH         REAL        ARRAY     VITK05          REFS        33         43
   76  XN         REAL                                  REFS        35         39    DEFINED        34         38
  145  YH         REAL        ARRAY     VITK05          REFS        33         43
  312  YP         REAL        ARRAY     VITK05          REFS        33
  457  YPP        REAL        ARRAY     VITK05          REFS        33         43

FILE NAMES         MODE
       OUTPUT     FMT                     WRITES        45

EXTERNALS            TYPE    ARGS        REFERENCES
       COS        REAL        1 LIBRARY      39
       SIN        REAL        1 LIBRARY      35         41
       SPLINT                10              43

INLINE FUNCTIONS    TYPE    ARGS         DEF LINE   REFERENCES
       FLOAT      REAL        1  INTRIN                34         38

STATEMENT LABELS                DEF LINE    REFERENCES
   30  10                          40          36
   66  15       FMT                46          45
   43  20                          47          44

COMMON BLOCKS    LENGTH      MEMBERS - BIAS NAME(LENGTH)
       VITK02        1               0 H       (1)
       VITK04        3               0 CHSUBM (1)                  1 COMEGAM(1)                  2 NN      (1)
       VITK05      708               0 XH      (101)             101 YH      (101)             202 YP      (101)
                                   303 YPP     (101)             404 WH      (303)             707 NP      (1)

STATISTICS
  PROGRAM LENGTH                 106B       70
  CM LABELED COMMON LENGTH      1310B      712
```

```
                GO TO 50                                                NATSEK   173
          C     ---------- ENTRY FOR ELIKM ----------                   NATSEK   174
            130 IF (CAYSQ .GT. 1.0E0) GO TO 200                         NATSEK   175
                ETA = CAYSQ                                             NATSEK   176
                GO TO 50                                                NATSEK   177
          C     ---------- ERROR RETURN FOR K**2 .LT. 0 ----------      NATSEK   178
            200 JK = 1                                                  NATSEK   179
                GO TO 260                                               NATSEK   180
          C     ---------- ERROR RETURN FOR K**2 .EQ. 1 ----------      NATSEK   181
            220 IF (ETA .LT. 0.0E0) GO TO 240                           NATSEK   182
                JK = 2                                                  NATSEK   183
                GO TO 260                                               NATSEK   184
            240 JK = 3                                                  NATSEK   185
          C     ---------- UPDATE ERROR COUNTS, ETC. ----------        NATSEK   186
            260 RESULT = XIND                                           NATSEK   187
                CALL FCNMON(IFCN,JK,FCN(INT),ARG,RESULT)                NATSEK   188
            300 RETURN                                                  NATSEK   189
          C     ---------- LAST CARD OF NATSEK ----------               NATSEK   190
                END                                                     NATSEK   191
```

CARD NR.	SEVERITY	DETAILS	DIAGNOSIS OF PROBLEM
147	I		AN IF STATEMENT MAY BE MORE EFFICIENT THAN A 2 OR 3 BRANCH COMPUTED GO TO STATEMENT.

SYMBOLIC REFERENCE MAP (R=3)

ENTRY POINTS		DEF LINE	REFERENCES
3	NATSEK	4	188

VARIABLES		SN	TYPE		RELOCATION								
125	A		REAL	ARRAY		REFS	6	152	156	164	DEFINED	135	
0	ARG		REAL		F.P.	REFS	6	146	187	DEFINED	4		
137	B		REAL	ARRAY		REFS	6	153	157	164	DEFINED	139	
115	CAY		REAL			REFS	6	170	171	DEFINED	169		
116	CAYSQ		REAL			REFS	6	148	149	169	174	175	
						DEFINED	146						
117	DELIPK		REAL			REFS	6	161	DEFINED	160			
112	EPS		REAL			REFS	6	151	DEFINED	128			
120	ETA		REAL			REFS	6	150	151	156	157	160	164
						2*171	181	DEFINED	149	170	171	175	
151	FCN		REAL	ARRAY		REFS	6	187	DEFINED	128			
123	I		INTEGER			REFS	8	156	157	DEFINED	155		
111	IFCN		INTEGER			REFS	8	187	DEFINED	128			
0	INT		INTEGER		F.P.	REFS	8	147	187	DEFINED	4		
124	JK		INTEGER			REFS	8	187	DEFINED	178	182	184	
0	RESULT		REAL		F.P.	REFS	6	187	DEFINED	4	161	164	186
121	SUMA		REAL			REFS	6	156	160	DEFINED	152	156	
122	SUMB		REAL			REFS	6	157	160	DEFINED	153	157	
110	XIND		REAL			REFS	6	186	DEFINED	128			

BLOCK	ADDRESS	LENGTH	FILE	PREFIX TABLE CONTENTS
				INTEGER TO INTEGER EXPONENTIATION.
SINCOS=	36347	66	SL-FORTRAN	10/24/75 11.09.47. SCOPE 3.4 COMPASS 3.75203 2
				TRIGONOMETRIC SINE OR COSINE OF X. OPT=ALL.
SYSAID=	36435	1	SL-FORTRAN	10/24/75 11.09.50. SCOPE 3.4 COMPASS 3.75203 2
				LINK BETWEEN SYS=AID AND INITIALIZATION CODE.
FLTIN=	36436	154	SL-FORTRAN	10/24/75 11.02.47. SCOPE 3.4 COMPASS 3.75203 2
				COMMON FLOATING INPUT CONVERTER.
FMTAP=	36612	351	SL-FORTRAN	10/24/75 11.02.53. SCOPE 3.4 COMPASS 3.75203 2
				CRACK APLIST AND FORMAT FOR KODER/KRAKER.
FORUTL=	37163	16	SL-FORTRAN	10/24/75 11.03.30. SCOPE 3.4 COMPASS 3.75203 2
				FCL MISC. UTILITIES.
GETFIT=	37201	42	SL-FORTRAN	10/24/75 11.03.32. SCOPE 3.4 COMPASS 3.75203 2
				LOCATE AN FIT GIVEN A FILE NAME.
NAMOUT=	37243	274	SL-FORTRAN	10/24/75 11.05.44. SCOPE 3.4 COMPASS 3.75203 2
				NAMELIST OUTPUT ROUTINE.
OUTC=	37537	175	SL-FORTRAN	10/24/75 11.06.31. SCOPE 3.4 COMPASS 3.75203 2
				FORMATTED WRITE FORTRAN RECORD.
SQRT	37734	43	SL-FORTRAN	10/24/75 11.09.49. SCOPE 3.4 COMPASS 3.75203 2
				COMPUTE THE SQUARE ROOT OF X. OPT=ALL.
SYS-1ST	37777	62	SL-FORTRAN	10/24/75 11.09.50. SCOPE 3.4 COMPASS 3.75203 2
				MATH LIBRARY LINK TO ERROR MESSAGE PROCESSOR.
XTOI=	40061	10	SL-FORTRAN	10/24/75 11.09.55. SCOPE 3.4 COMPASS 3.75203 2
				REAL TO INTEGER EXPONENTIATION.
SYS.RM	40071	37	SL-SYSIO	05/20/75 16.56.29. SCOPE 3.4 COMPASS 3.75050
				PROCESS SYSTEM REQUEST.
/CON.RM/	40130	6		
CIO.RM	40136	40	SL-SYSIO	09/20/75 17.27.39. SCOPE 3.4 COMPASS 3.2-406
/AOB.RM/	40176	10		
MOVE.RM	40206	64	SL-SYSIO	09/20/75 17.27.44. SCOPE 3.4 COMPASS 3.2-406
MCT.RM	40272	227	SL-SYSIO	09/20/75 17.27.46. SCOPE 3.4 COMPASS 3.2-406
/JMPS.RM/	40521	11		
/MEMC.RM/	40532	3		
/OPES.FO/	40535	1		
/OPEN.FO/	40536	7		
OPEN.RM	40545	231	SL-SYSIO	09/20/75 17.28.01. SCOPE 3.4 COMPASS 3.2-406
/TERM.RM/	40776	1		
/PUT.FO/	40777	7		
PUT.SQ	41006	1316	SL-SYSIO	09/20/75 17.28.45. SCOPE 3.4 COMPASS 3.2-406
WAR.SQ	42324	260	SL-SYSIO	09/20/75 17.29.37. SCOPE 3.4 COMPASS 3.2-406
/CLSF.FO/	42604	7		
CLSF.RM	42613	23	SL-SYSIO	09/20/75 17.30.18. SCOPE 3.4 COMPASS 3.2-406
/GET.BT/	42636	5		
BTRT.SQ	42643	114	SL-SYSIO	09/20/75 17.31.28. SCOPE 3.4 COMPASS 3.2-406
/SKFL.FO/	42757	7		
SKFL.SQ	42766	47	SL-SYSIO	09/20/75 17.32.23. SCOPE 3.4 COMPASS 3.2-406
ERR.RM	43035	404	SL-SYSIO	09/20/75 17.27.40. SCOPE 3.4 COMPASS 3.2-406
CHWR.SQ	43441	7	SL-SYSIO	09/20/75 17.27.46. SCOPE 3.4 COMPASS 3.2-406
OSUB.RM	43450	73	SL-SYSIO	09/20/75 17.28.07. SCOPE 3.4 COMPASS 3.2-406
OPEN.SQ	43543	262	SL-SYSIO	09/20/75 17.28.13. SCOPE 3.4 COMPASS 3.2-406
OPEX.SQ	44025	14	SL-SYSIO	09/20/75 17.28.31. SCOPE 3.4 COMPASS 3.2-406
/PUT.RT/	44041	11		
RLEQ.RM	44052	42	SL-SYSIO	09/20/75 17.28.42. SCOPE 3.4 COMPASS 3.2-406
CLSF.SQ	44114	131	SL-SYSIO	09/20/75 17.30.19. SCOPE 3.4 COMPASS 3.2-406
/CLSV.FO/	44245	7		
CLSV.SQ	44254	123	SL-SYSIO	09/20/75 17.30.23. SCOPE 3.4 COMPASS 3.2-406

ENTRY	ADDRESS	PROGRAM	REFERENCES							
			ANOT	22425	22434	22440	22471			
			SCALEB	26772	27025					
			NUMDIS	31420	31440	31451				
SYS=	40073	SYS.RM	PLADJ	26704	26711					
			Q8.IO.	32602	32610	32620				
			FORSYS=	33421	33440	33473	33532			
			NAMOUT=	37307						
			ERR.RM	43246	43265	43416				
RCL=	40106									
WNB=	40112		PUT.SQ	42120						
			GET.SQ	44444						
MSG=	40122		Q8.IO.	32616						
			FORSYS=	33465	33471	33521	33523	33530	34040	34060
				34066						
			REMARK	36137						
			PUT.SQ	41537	41573					
			ERR.RM	43203	43223	43274				
RM.CIO	40137	CIO.RM	PUT.SQ	41055	41500	41565	41605	42016	42026	42106
				42314						
			WAR.SQ	42426	42602					
			ERR.RM	43262						
			OPEN.SQ	43666	44003					
			OPEX.SQ	44032						
			CLSF.SQ	44156	44243					
			CLSV.SQ	44307	44317	44333	44343	44355		
			GET.SQ	44654	45041	45133	45136	45273	45305	
RM.RCLA	40147		PUT.SQ	41777	42136	42303				
			WAR.SQ	42526						
			BTRT.SQ	42743						
			ERR.RM	43252						
			CLSF.SQ	44144						
			GET.SQ	44521	44576					
RM.RCLP	40154		PUT.SQ	41513	41533					
			GET.SQ	45015						
RM.SYS=	40167									
MOVE.RM	40206	MOVE.RM	PUT.SQ	41366						
			FSU.SQ	45630						
MCT.RM	40273	MCT.RM	OPEN.RM	40567	40571	40671	40673	40741	40745	
			CLSF.RM	42620						
OPEN.RM	40545	OPEN.RM	FORSYS=	34201						
PUT.SQ	41007	PUT.SQ	FORSYS=	34103						
			INCOM=	34470						
			NAMOUT=	37455						
			OUTC=	37711						
FLSH.SQ	42115									
WAR.SQ	42324	WAR.SQ	PUT.SQ	41077						
REPO.SQ	42523									
CLSF.RM	42613	CLSF.RM	FORSYS=	33651	33711					
			NAMOUT=	37272						
			OUTC=	37612						
PUT.E	42643	BTRT.SQ								
PUT.I	42643									
BTRT.SQ	42643									
PUT.C	42643		FCL.C.	32543						
			NAMIN=	35600						

```
B 10

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 2,

$END
 IFUNC    =   2
 GNU      =        .93000000000000E+10
 PLANC    =        .10545887000000E-26
 BETA     =        .92740780000000E-20
 PI       =        .31415926535898E+01
 HALF     =        .10000000000000E+00
 OVRHALF  =        .99999999999999E+01
 HZERO    =        .33195269359482E+04
 HONE     =        .33218502563053E+04
 H2       =        .33213521282053E+04
 H3       =        .33195269359482E+04
 H1S      =        .11034689125316E+08
 H2S      =        .11031379959534E+08
 H3S      =        .11019259078486E+08
 G1       =        .20003000000000E+01
 G2       =        .20006000000000E+01
 G3       =        .20017000000000E+01
 S        =        .44007492998454E-05
 XLOG2    =        .69314718055995E+00
 PILOG    =        .93943727869965E-01
 TONE     =        .25000000000000E-03
 GAMMA    =        .17602997199382E+08
 H1STAR   =        .10000000000000E-04
 HLFSTR   =        .50000000000000E+01
 OHLFSTR  =        .20000000000000E+00
 DELTAH   =        .20000000000000E+00
 DELTAHS  =        .10000000000000E+02
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
 ***  UNCLASSIFIED  ***      REHVT,T300,KUNC.   REHFORVITKO CY=46  AND MICROFICHE
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   02/05/76   13.00.00.   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
      SUBROUTINE  CONSTNT                                               CONSTNT     2
C                                                                       CONSTNT     3
C         SUBROUTINE CONSTNT READS IN DATA FROM THE NAMELIST VITKO.     CONSTNT     4
C         THE VARIABLES READ IN ARE                                     CONSTNT     5
C               G1          TONE                H1STAR                  CONSTNT     6
C               G2          DELTAH              GNU                     CONSTNT     7
C               G3          DELTAHS             IFUNC                   CONSTNT     8
C     THESE VALUES ARE WRITTEN ON LOGICAL UNIT 50. SETTING TAPE50=OUTPUT CONSTNT    9
C     ON THE PROGRAM CARD WILL CAUSE THE VALUES TO BE PRINTED.          CONSTNT    10
C                                                                       CONSTNT    11
C     SUBROUTINE CONSTNT CALCULATES A NUMBER OF CONSTANTS USED IN OTHER CONSTNT    12
C     SUBROUTINES OF THIS PACKAGE, STORES THE CALCULATED VALUES IN      CONSTNT    13
C     COMMON BLOCK VITKO1, AND WRITES THE VALUES ON LOGICAL UNIT 50.    CONSTNT    14
C                                                                       CONSTNT    15
C                                                                       VITKO1      2
      COMMON /VITKO1/ GNU,    PLANC,   BETA,    PI,    HALF,  OVRHALF,  VITKO1      3
     *                                                                  VITKO1      4
     * HZERO, HONE,   H2 ,  H3 ,   H1S,   H2S,  H3S,                    VITKO1      5
     *                                                                  VITKO1      6
     * G1,  G2,  G3,  S,   IFUNC,    XLOG2,   PILOG,    TONE ,  GAMMA   VITKO1      7
     *                                                                  VITKO1      8
     * ,HISTAR, HLFSTR, OHLFSTR, DELTAH, DELTAHS                        VITKO1M1    1
C                                                                       VITKO1     12
C                                                                       VITKO1     13
      DIMENSION SETCOM(27)                                              CONSTNTM1   1
      EQUIVALENCE(SETCOM(1),GNU)                                        CONSTNT    19
      DATA  SECTOM /27*1.0/                                             CONSTNTM1   2
C                                                                       CONSTNTM1   3
C                                                                       CONSTNTM1   4
C     THE ABOVE DATA STATEMENT AND THE EQUIVALENCE STATEMENT HAVE THE   CONSTNTM1   5
C     EFFECT OF SETTING EACH VARIABLE IN COMMON BLOCK VITKO1 TO THE     CONSTNTM1   6
C     VALUE OF 1.0 . IT IS QUITE REASONABLE TO RUN A PROBLEM FOR THE    CONSTNTM1   7
C     GAUSSIAN CASE WITHOUT HAVING READ IN VALUES OF G1 AND G2, BUT     3JAN(76)    3
C     G1 AND G2 APPEAR IN THE DENOMINATOR IN THE CALCULATIONS OF        3JAN(76)    4
C     CONSTANTS IN CONSTNT. HENCE, THERE WOULD BE AN ERROR. SKIPPING    CONSTNTM1  10
C     OVER THESE CALCULATIONS BY KEYING ON IFLAG IS NOT DESIRABLE SINCE CONSTNTM1  11
C     ONE MAY WISH TO READ IN THE DATA FOR BOTH THE GAUSSIAN AND        CONSTNTM1  12
C     POWDER LINESHAPES IN ONE CALL TO CONSTNT.                         CONSTNTM1  13
C                                                                       CONSTNTM1  14
C                                                                       CONSTNTM1  15
      NAMELIST/VITKO/  G1,G2,G3,TONE,DELTAH,DELTAHS,H1STAR ,GNU,IFUNC   CONSTNT    21
C                                                                       CONSTNT    22
C     IFUNC  =  1   GAUSSIAN SHAPE FUNCTION        -   ABSORPTION CASE  IFUNC       2
C            =  2   GAUSSIAN SHAPE FUNCTION        -   DISPERSION CASE  IFUNC       3
C            =  3   GAUSSIAN (DERIVATIVE)          -   ABSORPTION CASE  IFUNC       4
C            =  4   GAUSSIAN (DERIVATIVE)          -   DISPERSION CASE  IFUNC       5
C            =  5   POWDER LINESHAPE               -   ABSORPTION CASE  IFUNC       6
C            =  6   POWDER LINESHAPE               -   DISPERSION CASE  IFUNC       7
C            =  7   POWDER LINESHAPE (DERIVATIVE)  -   ABSORPTION CASE  IFUNC       8
C            =  8   POWDER LINESHAPE (DERIVATIVE)  -   DISPERSION CASE  IFUNC       9
C                                                                       IFUNC      10
      READ VITKO                                                        CONSTNT    24
      WRITE (50,VITKO)                                                  CONSTNT    25
C                                                                       CONSTNT    26
C                                                                       CONSTNT    27
C                                                                       CONSTNT    28
C       SET  PARAMETERS                                                 CONSTNT    29
```

```
        SUBROUTINE SIGNAL     74/74   OPT=1                          FTN 4.5+R406       02/05/76  13.03.47       PAGE     3

 VARIABLES      SN  TYPE           RELOCATION
     5  OVRHALF    REAL                VITK01       REFS        48
     3  PI         REAL                VITK01       REFS        48
    23  PILOG      REAL                VITK01       REFS        48
     1  PLANC      REAL                VITK01       REFS        48
    20  S          REAL                VITK01       REFS        48
    24  TONE       REAL                VITK01       REFS        48
    22  XLOG2      REAL                VITK01       REFS        48
     0  XPLOT      REAL        ARRAY    F.P.        REFS        46    DEFINED        1         71
     0  YPLOT      REAL        ARRAY    F.P.        REFS        46    DEFINED        1         72

 EXTERNALS            TYPE    ARGS       REFERENCES
        CONVOL      REAL        1              72

 STATEMENT LABELS              DEF LINE   REFERENCES
    25  10                        62         57
    32  15                        66         61
     0  20                        73         69

 LOOPS  LABEL     INDEX      FROM-TO     LENGTH     PROPERTIES
    40  20      * J            69 73       10B                EXT REFS

 COMMON BLOCKS    LENGTH      MEMBERS - BIAS NAME(LENGTH)
        VITK01       27                 0 GNU    (1)                 1 PLANC  (1)                 2 BETA   (1)
                                        3 PI     (1)                 4 HALF   (1)                 5 OVRHALF(1)
                                        6 HZERO  (1)                 7 HONE   (1)                 8 H2     (1)
                                        9 H3     (1)                10 H1S    (1)                11 H2S    (1)
                                       12 H3S    (1)                13 G1     (1)                14 G2     (1)
                                       15 G3     (1)                16 S      (1)                17 IFUNC  (1)
                                       18 XLOG2  (1)                19 PILOG  (1)                20 TONE   (1)
                                       21 GAMMA  (1)                22 HISTAR (1)                23 HLFSTR (1)
                                       24 OHLFSTR(1)                25 DELTAH (1)                26 DELTAHS(1)

 STATISTICS
   PROGRAM LENGTH                 72B       58
   CM LABELED COMMON LENGTH       33B       27
```

```
      FUNCTION SY(HPRIM)                                                SY          2
C                                                                       SY          3
C       SUBROUTINE SY(HPRIM) CALCULATES THE FUNCTIONS                   SY          4
C               SY(HPRIM) = S(HPRIM)*Y(H-HPRIM)                         SY          5
C          OR                                                           SY          6
C               SY(HPRIM) = S(HPRIM)*(D/DH)(Y(H-HPRIM))                 SY          7
C     FOR VARIOUS DEFINITIONS OF S AND Y AS DETERMINED BY IFUNC         SY          8
C                                                                       SY          9
C     IFUNC  =  1   GAUSSIAN SHAPE FUNCTION        -   ABSORPTION CASE  IFUNC       2
C            =  2   GAUSSIAN SHAPE FUNCTION        -   DISPERSION CASE  IFUNC       3
C            =  3   GAUSSIAN (DERIVATIVE)          -   ABSORPTION CASE  IFUNC       4
C            =  4   GAUSSIAN (DERIVATIVE)          -   DISPERSION CASE  IFUNC       5
C            =  5   POWDER LINESHAPE               -   ABSORPTION CASE  IFUNC       6
C            =  6   POWDER LINESHAPE               -   DISPERSION CASE  IFUNC       7
C            =  7   POWDER LINESHAPE (DERIVATIVE)  -   ABSORPTION CASE  IFUNC       8
C            =  8   POWDER LINESHAPE (DERIVATIVE)  -   DISPERSION CASE  IFUNC       9
C                                                                       IFUNC      10
C                                                                       SY         11
C     THE VALUE OF H IN Y(H-HPRIM) IS PASSED THROUGH COMMON BLOCK       SY         12
C     VITKO2                                                            SY         13
C                                                                       SY         14
C                                                                       VITKO1      2
      COMMON /VITKO1/ GNU,    PLANC,   BETA,    PI,    HALF,  OVRHALF,  VITKO1      3
     *                                                                  VITKO1      4
     * HZERO, HONE,   H2 ,  H3 ,   H1S,   H2S,  H3S,                    VITKO1      5
     *                                                                  VITKO1      6
     * G1,  G2,  G3,  S,   IFUNC,    XLOG2,   PILOG,    TONE ,  GAMMA  VITKO1      7
     *                                                                  VITKO1      8
     * ,HISTAR, HLFSTR, OHLFSTR, DELTAH, DELTAHS                        VITKO1M1    1
C                                                                       VITKO1     12
C                                                                       VITKO1     13
C                                                                       VITKO2      2
      COMMON /VITKO2/ H                                                 VITKO2      3
C                                                                       VITKO2      4
C                                                                       VITKO2      5
C                                                                       SY         17
C                 ******                                                SY         18
C                                                                       SY         19
C         CALCULATE  Y                                                  SY         20
C                                                                       SY         21
      X = HPRIM - H                                                     SY         22
      T1 = X * OVRHALF                                                  SY         23
      F  = T1*T1                                                        SY         24
      YA  =  OVRHALF/(1.0 + S + F)                                      SY         25
      GO TO(1,2,3,4,1,2,3,4)  IFUNC                                     SY         26
    1 CONTINUE                                                          SY         27
      YFUNC =  YA                                                       SY         28
      GO TO 70                                                          SY         29
    2 CONTINUE                                                          SY         30
      YFUNC = YA*T1                                                     SY         31
      GO TO 70                                                          SY         32
    3 CONTINUE                                                          SY         33
      YFUNC =  2.0*T1*YA*YA                                             SY         34
      GO TO 70                                                          SY         35
    4 CONTINUE                                                          SY         36
      YFUNC = YA*(2.0*F*YA-OVRHALF)                                     SY         37
   70 CONTINUE                                                          SY         38
```

```
        FUNCTION CONVOL      74/74   OPT=1                          FTN 4.5+R406       02/05/76  13.03.47       PAGE      4

 EXTERNALS            TYPE   ARGS       REFERENCES
       CNVLERR                1             99
       EXIT                   0             81
       GAUS8                  6             75        83        85
       SY                     0             27        75        83        85

 INLINE FUNCTIONS    TYPE   ARGS        DEF LINE  REFERENCES
       ABS        REAL        1  INTRIN                87        89
       AMAX1      REAL        0  INTRIN                59
       AMIN1      REAL        0  INTRIN                62

 STATEMENT LABELS              DEF LINE   REFERENCES
    42  1                         74         97
    46  2                         77         84        86
   146  3        FMT              78         77
    54  4                         82         76
   104  5                         98         92
   107  7                        100         90
   161  10       FMT              80         79
    30  110                       64         53
    41  150                       72         63

 COMMON BLOCKS    LENGTH      MEMBERS - BIAS NAME(LENGTH)
       VITKO1        27                 0 GNU    (1)                 1 PLANC  (1)                 2 BETA   (1)
                                        3 PI     (1)                 4 HALF   (1)                 5 OVRHALF(1)
                                        6 HZERO  (1)                 7 HONE   (1)                 8 H2     (1)
                                        9 H3     (1)                10 H1S    (1)                11 H2S    (1)
                                       12 H3S    (1)                13 G1     (1)                14 G2     (1)
                                       15 G3     (1)                16 S      (1)                17 IFUNC  (1)
                                       18 XLOG2  (1)                19 PILOG  (1)                20 TONE   (1)
                                       21 GAMMA  (1)                22 HISTAR (1)                23 HLFSTR (1)
                                       24 OHLFSTR(1)                25 DELTAH (1)                26 DELTAHS(1)
       VITKO2         1                 0 H      (1)
       VITKO3         6                 0 FAC    (1)                 1 XFAC   (1)                 2 TESTFAC(1)
                                        3 ERR    (1)                 4 NPTS   (1)                 5 DIF    (1)

 STATISTICS
   PROGRAM LENGTH                 212B      138
   CM LABELED COMMON LENGTH        42B       34
```

```
      C
      C     ------- FUNPACK ------- CDC 6000 ----- SINGLE PRECISION ---------
      C
            SUBROUTINE FCNMON(IFCN,JERR,FCN,ARG,RESULT)
      C
            REAL ARG,FCN,RESULT
            INTEGER ERCONT(20),IERCNT(20,5),IFCN,J,JERR,JK,K,NERR(20),
           1        NPRINT
            LOGICAL IPRINT(20)
      C
      C     THIS ROUTINE IS INTENDED FOR USE ONLY BY OTHER ELEMENTS OF
      C     FUNPACK.  ALL ERROR DIAGNOSTIC FACILITIES AND PRINT STATEMENTS
      C     ARE CONCENTRATED WITHIN THIS SUBROUTINE.  CALLS BY INDIVIDUAL
      C     FUNPACK  FUNCTION PACKETS USE THE PARAMETERS
      C
      C     IFCN    - AN INTEGER IDENTIFYING THE CALLING PACKET.
      C
      C     JERR    - AN INTEGER IDENTIFYING THE ERROR DETECTED WITHIN THE
      C               CALLING PACKET.
      C
      C     FCN     - A 6 CHARACTER HOLLERITH STRING IDENTIFYING THE ACTIVE
      C               ENTRY IN THE CALLING PACKET.
      C
      C     ARG     - THE REAL ARGUMENT LEADING TO THE ERROR CONDITION.
      C
      C     RESULT  - THE REAL FUNCTION VALUE BEING RETURNED BY THE
      C               CALLING PACKET.
      C
      C     CALLS TO THIS ROUTINE FROM THE ERROR MONITORING SUBPROGRAM
      C     MONERR  SPECIFY ONLY THE PARAMETERS
      C
      C     IFCN    - THE PARAMETER OF THE SAME NAME INPUT TO  MONERR,
      C
      C     JERR    - THE VALUE  -KK-2, WHERE  KK  IS AN INPUT PARAMETER TO
      C               MONERR,
      C
      C     ALL OTHER PARAMETERS BEING DUMMIES.
      C
      C     ERROR INFORMATION IS STORED IN  FCNMON  IN ARRAYS WITH THE FIRST
      C     SUBSCRIPT KEYED TO  IFCN.  THESE ARRAYS ARE DESCRIBED BELOW.
      C
      C     IPRINT    - A LOGICAL ARRAY AUTHORIZING THE PRINTING OF ERROR
      C                 MESSAGES.
      C
      C     ERCONT    - AN INTEGER ARRAY AUTHORIZING THE TERMINATION (VALUE
      C                 OF -1) OR CONTINUATION (ANY OTHER VALUE) OF THE
      C                 COMPUTER RUN AFTER ERROR DETECTION.
      C
      C     IERCNT    - AN INTEGER ARRAY TABULATING THE FREQUENCY OF THE
      C                 VARIOUS ERRORS REPORTED BY EACH FUNCTION PACKET.
      C
      C     NERR      - AN INTEGER ARRAY DESIGNATING THE LAST ERROR REPORTED
      C                 BY EACH FUNCTION PACKET.
      C
      C     ----------
      C
      C     QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW
```

```
EXTERNALS          TYPE    ARGS        REFERENCES
       ALOG       REAL       1 LIBRARY     160        164
       FCNMON                5             187

INLINE FUNCTIONS   TYPE    ARGS         DEF LINE   REFERENCES
       ABS        REAL       1  INTRIN                169

STATEMENT LABELS              DEF LINE    REFERENCES
   16  20                       148         147
   22  50                       150         172        176
    0  100                      158         155
   47  105                      164         151
   53  120                      169         147
   60  130                      174         147
   64  200                      178         148        174
   66  220                      181         150
   71  240                      184         181
   72  260                      186         179        183
  101  300                      188         162        165

LOOPS  LABEL     INDEX      FROM-TO     LENGTH     PROPERTIES
   33  100       I          155 158        6B       INSTACK

STATISTICS
  PROGRAM LENGTH                154B      108
```

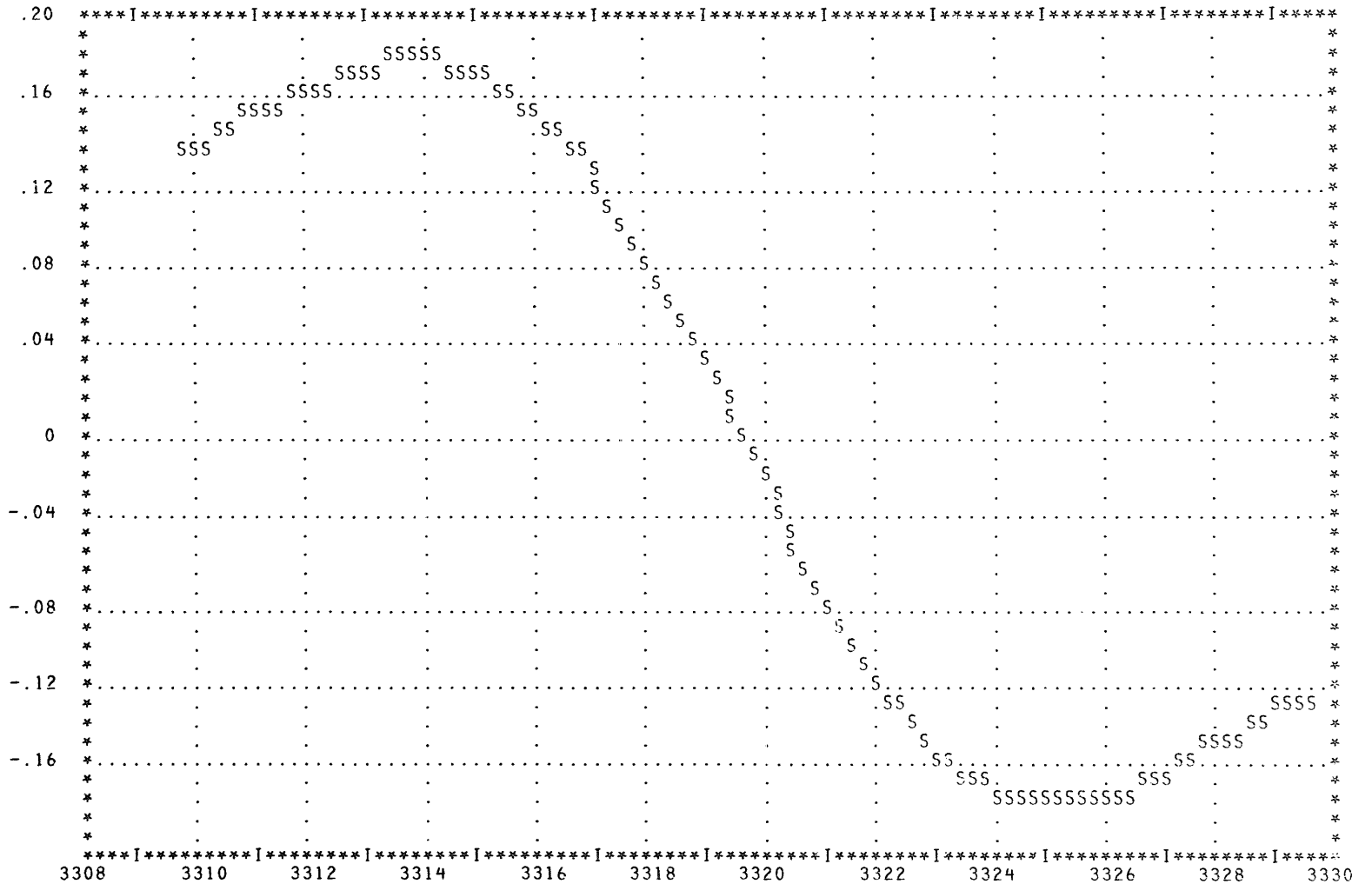
BLOCK	ADDRESS	LENGTH	FILE		PREFIX TABLE CONTENTS
/GET.FO/	44377	7			
/GET.RT/	44406	11			
GET.SQ	44417	1031	SL-SYSIO	09/20/75	17.30.32. SCOPE 3.4 COMPASS 3.2-406
Z.SQ	45450	101	SL-SYSIO	09/20/75	17.31.00. SCOPE 3.4 COMPASS 3.2-406
FSU.SQ	45551	106	SL-SYSIO	09/20/75	17.31.24. SCOPE 3.4 COMPASS 3.2-406
CPC	45657	251	SL-NUCLEUS	05/20/75	16.55.14. SCOPE 3.4 COMPASS 3.75050
IO	46130	203	SL-NUCLEUS	05/20/75	16.55.17. SCOPE 3.4 COMPASS 3.75050

ENTRY POINTS.

ENTRY	ADDRESS	PROGRAM	REFERENCES						
TAPE1]	101	TEST							
INPUT]	101		CONSTNT	7171					
OUTPUT]	2142		FOUR	11360					
			CONVOL	12004	12017				
			CNVLERR	12232	12266				
			GSIN	12400					
			ERRCHK	15543	15556	15563	15572	15601	15611
			ABTERM	15751					
			CURSES	16037					
			ONECHK	21434	21443				
TAPE50]	4203		CONSTNT	7175	7200	7207	7217	7227	7237	7247
				7257	7267	7277	7307	7317	7327	7337
				7347	7357	7367	7377	7407	7417	7427
				7437	7447	7457	7467	7477	7507	7517
			FOUR	11346					
			CNVLERR	12245					
TEST	6253								
CONSTNT	7012	CONSTNT	TEST	6261					
SIGNAL	7542	SIGNAL	TEST	6271					
FOUR	11144	FOUR	TEST	6312					
SY	11462	SY	CONVOL	11755	11764	11773			
			CNVLERR	12166	12175	12202	12211	12216	12225
CONVOL	11646	CONVOL	SIGNAL	7603					
			FOUR	11232					
CNVLERR	12057	CNVLERR	CONVOL	11750					
GSIN	12322	GSIN	FOUR	11327					
GCOS	12332		FOUR	11336					
FCNMON	12427	FCNMON	NATSEK	13213					
NATSEK	13116	NATSEK	ELIPK	13300					
ELIPK	13273	ELIPK	SY	11565	11577				
REHPLOT	13315	REHPLOT	CNVLERR	12112	12124	12132			
ERRCHK	15372	ERRCHK	GAUS8	16655					
			QNC7	17676					
			SPLIFT	21277	21303				
			SPLINT	21722					
ERXSET	15714	ERXSET	FOUR	11256					
ABTERM	15731	ABTERM	ERRCHK	15523					
			CURSES	16033					
REMARX	15775	REMARX	ERRCHK	15410					
CURSES	16024	CURSES	GAUS8	16170					
			QNC7	17353					

ENTRY	ADDRESS	PROGRAM	REFERENCES							
PUT.K	42643									
SKFL.SQ	42766	SKFL.SQ	FORSYS=	33644						
ERR.RM	43113	ERR.RM	JMPS.RM	40522	40523	40524	40525	40526	40527	40531
			MEMC.RM	40532	40533	40534				
			OPES.FO	40535						
			OPEN.FO	40536	40537	40540	40541	40542	40543	40544
			OPEN.RM	40547	40637	40704	40750	40757	40761	40765
				40767	40770					
			PUT.FO	40777	41000	41001	41002	41003	41004	41005
			PUT.SQ	41006	41013	41047	41070	41073	41100	41147
				41211	41245	41274	41502	41531	41532	41552
				42311	42317					
			CLSF.FO	42604	42605	42606	42607	42610	42611	42612
			CLSF.RM	42615	42626	42634				
			SKFL.FO	42757	42760	42761	42762	42763	42764	42765
			SKFL.SQ	42770	42773	42776				
			OSUB.RM	43531	43532	43533	43534	43535	43536	43537
				43540	43541					
			OPEN.SQ	43617	43621	43624	43665	43732	43734	43766
				44016	44017					
			OPEX.SQ	44026						
			RLEQ.RM	44055						
			CLSF.SQ	44122	44201	44244				
			CLSV.FO	44245	44246	44247	44250	44251	44252	44253
			GET.FO	44377	44400	44401	44402	44403	44404	44405
			GET.SQ	44417	44423	44440	44447	44454	45157	45301
				45334	45357					
ERR1.RM	43117									
ERR2.RM	43275									
CHWR.SQ	43441	CHWR.SQ	PUT.SQ	41303						
			OPEN.SQ	43553						
OSUB.RM	43450	OSUB.RM	OPEN.RM	40705						
OPEN.SQ	43543	OPEN.SQ	FORSYS=	34201						
OPXX.SQ	43660		OPEX.SQ	44040						
OPEX.SQ	44025	OPEX.SQ	OPEN.SQ	43657						
RLEQ.RM	44052	RLEQ.RM	PUT.SQ	41121	41203					
			WAR.SQ	42325						
CLSF.SQ	44114	CLSF.SQ								
RSPT.SQ	44202		OPEN.SQ	43743						
			OPEX.SQ	44027						
			CLSV.SQ	44273	44367					
			GET.SQ	44577	45265					
CLSV.SQ	44256	CLSV.SQ	PUT.SQ	41555	42007					
			GET.SQ	45374						
GET.SQ	44420	GET.SQ	NAMIN=	35601						
SKGT.SQ	44471		SKFL.SQ	43014	43022					
GCNT.SQ	44530									
GXIT.SQ	44600		Z.SQ	45477	45534	45546				
			FSU.SQ	45605	45621	45644	45646			
GRTJ.SQ	44661		FSU.SQ	45606	45651					
AINI.SQ	44713									
ANBL.SQ	44720		BTRT.SQ	42655	42673	42700	42756			
AMAC.SQ	44724		BTRT.SQ	42643	42656	42702	42703	42712	42713	42714
				42722	42730	42733	42741			
			FSU.SQ	45607	45651	45656				

C 10


.20
.16
.12
.08
.04
0
-.04
-.08
-.12
-.16
3308
3310
3312
3314
3316
3318
3320
3322
3324
3326
3328
3330

PLOT 3

D 1

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX 02/05/76 13.00.00. XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```
      HALF    = .5*DELTAH                                          CONSTNT    30
      HLFSTR  = .5*DELTAHS                                         CONSTNT    31
      BETA    =  9.274078E-21                                      CONSTNT    32
      PLANC   =  1.0545887E-27                                     CONSTNT    33
      OVRHALF =  1.0/HALF                                          CONSTNT    34
      OHLFSTR = 1.0/HLFSTR                                         CONSTNT    35
      GAMMA   = G3*BETA/PLANC                                      CONSTNT    36
      PI      = 3.141592653589793                                  CONSTNT    37
      XLOG2   = ALOG(2.0)                                          CONSTNT    38
      PILOG = OHLFSTR*SQRT(XLOG2/PI)                               CONSTNT    39
      TEMP    = 2.0*PI*PLANC*GNU/BETA                              CONSTNT    40
      HZERO  =  TEMP/G3                                            CONSTNT    41
      HONE   =  TEMP/G1                                            CONSTNT    42
      H2     =  TEMP/G2                                            CONSTNT    43
      H3     =  TEMP/G3                                            CONSTNT    44
      H1S = HONE*HONE                                              CONSTNT    45
      H2S = H2*H2                                                  CONSTNT    46
      H3S = H3*H3                                                  CONSTNT    47
      S = GAMMA*H1STAR*H1STAR*TONE*OVRHALF                         CONSTNT    48
      WRITE(50,116)  IFUNC                                         CONSTNT    49
  116 FORMAT(*  IFUNC    = *, I3     )                             CONSTNT    50
      WRITE(50,101) GNU                                            CONSTNT    51
  101 FORMAT(*  GNU      = *, E25.14)                              CONSTNT    52
      WRITE(50,102) PLANC                                          CONSTNT    53
  102 FORMAT(*  PLANC    = *, E25.14)                              CONSTNT    54
      WRITE(50,103) BETA                                           CONSTNT    55
  103 FORMAT(*  BETA     = *, E25.14)                              CONSTNT    56
      WRITE(50,104) PI                                             CONSTNT    57
  104 FORMAT(*  PI       = *, E25.14)                              CONSTNT    58
      WRITE(50,105) HALF                                           CONSTNT    59
  105 FORMAT(*  HALF     = *, E25.14)                              CONSTNT    60
      WRITE(50,106) OVRHALF                                        CONSTNT    61
  106 FORMAT(*  OVRHALF = *, E25.14)                               CONSTNT    62
      WRITE(50,107) HZERO                                          CONSTNT    63
  107 FORMAT(*  HZERO    = *, E25.14)                              CONSTNT    64
      WRITE(50,108) HONE                                           CONSTNT    65
  108 FORMAT(*  HONE     = *, E25.14)                              CONSTNT    66
      WRITE(50,127) H2                                             CONSTNT    67
  127 FORMAT(*  H2       = *, E25.14)                              CONSTNT    68
      WRITE(50,109) H3                                             CONSTNT    69
  109 FORMAT(*  H3       = *, E25.14)                              CONSTNT    70
      WRITE(50,110) H1S                                            CONSTNT    71
  110 FORMAT(*  H1S      = *, E25.14)                              CONSTNT    72
      WRITE(50,111) H2S                                            CONSTNT    73
  111 FORMAT(*  H2S      = *, E25.14)                              CONSTNT    74
      WRITE(50,126) H3S                                            CONSTNT    75
  126 FORMAT(*  H3S      = *, E25.14)                              CONSTNT    76
      WRITE(50,112) G1                                             CONSTNT    77
  112 FORMAT(*  G1       = *, E25.14)                              CONSTNT    78
      WRITE(50,113) G2                                             CONSTNT    79
  113 FORMAT(*  G2       = *, E25.14)                              CONSTNT    80
      WRITE(50,114) G3                                             CONSTNT    81
  114 FORMAT(*  G3       = *, E25.14)                              CONSTNT    82
      WRITE(50,115) S                                              CONSTNT    83
  115 FORMAT(*  S        = *, E25.14)                              CONSTNT    84
      WRITE(50,117) XLOG2                                          CONSTNT    85
  117 FORMAT(*  XLOG2    = *, E25.14)                              CONSTNT    86
```

```
      SUBROUTINE FOUR(HVEC,COFVEC,IPTS,IFLAG,HSUBM,OMEGAM,N,FAC)
C
C
C        LET I(H) = INTEGRAL(0,INFINITY)( SY(HPRIM) ) WHERE SY IS GIVEN
C     BY SUBROUTINE SY ,THE VALUE OF WHICH IS DETERMINED BY IFUNC
C     ACCORDING TO THE RULE
C
C     IFUNC  =  1   GAUSSIAN SHAPE FUNCTION        -   ABSORPTION CASE
C            =  2   GAUSSIAN SHAPE FUNCTION        -   DISPERSION CASE
C            =  5   POWDER LINESHAPE               -   ABSORPTION CASE
C            =  6   POWDER LINESHAPE               -   DISPERSION CASE
C
C
C         LET  G(T) =  I(H + HSUBM*SIN(OMEGAM*T) )
C
C     AND EXPAND G IN A FOURIER SERIES
C
C      G(T) = .5FCOS(0) + SUM(N=1,...)(FSIN(N)*SIN(N*OMEGAM*T)
C                                    + FCOS(N)*COS(N*OMEGAM*T)
C
C
C        CLEARLY THE FOURIER COEFFICIENTS FSIN(N) AND FCOS(N) ARE
C     FUNCTIONS OF H (AND NOT OF T).  SUBROUTINE FOUR RETURNS COFVEC
C     (READ AS FOURIER COEFFICIENT VECTOR)  AND  HVEC (READ AS VECTOR
C     OF H VALUES )  WHERE
C        HVEC(I) = HSTART +(I-1)*(HFINAL-HSTART)/(IPTS-1) ,I=1,...,IPTS
C
C                 WHERE HSTART AND HFINAL ARE GIVEN BY
C                       IF IFUNC = 1 OR 2 (GAUSSIAN CASE)
C                            HSTART = HZERO - FAC*DELTAHS/2.0
C                         AND
C                            HFINAL = HZERO + FAC*DELTAHS/2.0
C                          GIVEN THAT
C                               HZERO = PLANC*GNU/(BETA*G3)
C                    OR
C                       IF IFUNC = 5 OR 6
C                            HSTART = H3 - FAC*(HONE - H3)
C                         AND
C                            HFINAL = HONE + FAC*(HONE - H3)
C
C     AND
C        COFVEC(I) = FSIN(N) EVALUATED AT HVEC(I)  IF IFLAG = 1
C      OR
C        COFVEC(I) = FCOS(N) EVALUATED AT HVEC(I)  IF IFLAG = 2 ,
C
C
C     *****
C     NOTE-
C         SUBROUTINE CONSTANT MUST HAVE BEEN CALLED BEFORE A CALL TO
C         FOUR (TO SET THE VALUES IN COMMON BLOCK VITK01)
C     *****
C
C
C
C         THE CALCULATION OF THE FOURIER COEFFICIENTS REQUIRES MANY
C     EVALUATIONS OF I(H). FOR EFFICIENCY, I(H) IS FIT WITH A CUBIC
```

```
C                                                                       SY        39
C                  ******                                               SY        40
C                  ******                                               SY        41
C                  ******                                               SY        42
C                  ******                                               SY        43
C                  ******                                               SY        44
C                  ******                                               SY        45
C                                                                       SY        46
      IF( IFUNC .GE. 5) GO TO 100                                       SY        47
C                                                                       SY        48
C                  ******                                               SY        49
C                                                                       SY        50
C                       CALCULATE S                                     SY        51
C                                                                       SY        52
C                                                                       SY        53
C      GAUSSIAN CASE                                                    SY        54
C                                                                       SY        55
      T3 = (HPRIM - HZERO)*OHLFSTR                                      SY        56
      T3 = T3*T3*XLOG2                                                  SY        57
      IF( T3 .GE.  675.0 )   GO TO 80                                   SY        58
      SFUNC = PILOG*EXP(-T3)                                            SY        59
      GO TO 90                                                          SY        60
   80 SFUNC = 0.0                                                       SY        61
   90 CONTINUE                                                          SY        62
      SY = YFUNC * SFUNC                                                SY        63
      RETURN                                                            SY        64
C                                                                       SY        65
C         POWDER LINESHAPE                                              SY        66
C                                                                       SY        67
  100 CONTINUE                                                          SY        68
      IF( HPRIM .GE. H3  .AND.   HPRIM .LE. HONE ) GO TO 120            SY        69
      SY = 0.0                                                          SY        70
      RETURN                                                            SY        71
  120 CONTINUE                                                          SY        72
      HS = HPRIM*HPRIM                                                  SY        73
      T4 =  (2.0/PI)*(HONE*H2*H3/HS)                                    SY        74
      T5 = H1S - H2S                                                    SY        75
      T6 = HS  - H3S                                                    SY        76
      T7 = H1S - HS                                                     SY        77
      T8 = H2S - H3S                                                    SY        78
      XK = (T5*T6) / (T7*T8)                                            SY        79
      IF( HPRIM .GT. H2 ) GO TO 140                                     SY        80
C                                                                       SY        81
C     CALL ELLIPTIC INTEGRAL ROUTINE                                    SY        82
      ELLIP = ELIPK( XK)                                                SY        83
      SFUNC =  T4 * ELLIP / SQRT( T7*T8)                                SY        84
      SY = SFUNC * YFUNC                                                SY        85
      RETURN                                                            SY        86
  140 CONTINUE                                                          SY        87
      XK = 1.0/XK                                                       SY        88
      ELLIP  = ELIPK( XK)                                               SY        89
      SFUNC = T4 * ELLIP / SQRT( T5*T6)                                 SY        90
      SY = SFUNC*YFUNC                                                  SY        91
      RETURN                                                            SY        92
      END                                                               SY        93
```

```
      SUBROUTINE CNVLERR(HH)                                            CNVLERR     2
C       SUBROUTINE CNVLERR IS CALLED BY CONVOL ONLY IF CONVOL FAILS TO  CNVLERR     3
C     ACHIEVE CONVERGENCE OF THE SEMI-INFINITE INTEGRALS. CNVLERR WRITES CNVLERR    4
C     SOME CRYPTIC MESSAGES ON LOGICAL UNIT 50 AND HALTS EXECUTION.     CNVLERR     5
C     ON OCCURENCE OF SUCH MESSAGES, CONTACT                            CNVLERR     6
C                                                                       CNVLERR     7
C                 R. E. HUDDLESTON                                      CNVLERR     8
C                 NUMERICAL APPLICATIONS DIVISION  8322                 CNVLERR     9
C                 EXT. 2120                                             CNVLERR    10
C                                                                       CNVLERR    11
C                                                                       CNVLERR    12
C                                                                       CNVLERR    13
      EXTERNAL SY                                                       CNVLERR    14
C                                                                       VITK01      2
      COMMON /VITK01/ GNU,    PLANC,   BETA,    PI,    HALF,  OVRHALF,  VITK01      3
     *                                                                  VITK01      4
     * HZERO, HONE,   H2 ,  H3 ,   H1S,   H2S,  H3S,                    VITK01      5
     *                                                                  VITK01      6
     * G1,  G2,  G3,  S,   IFUNC,    XLOG2,   PILOG,    TONE ,  GAMMA  VITK01      7
     *                                                                  VITK01      8
     * ,HISTAR, HLFSTR, OHLFSTR, DELTAH, DELTAHS                        VITK01M1    1
C                                                                       VITK01     12
C                                                                       VITK01     13
C                                                                       VITK02      2
      COMMON /VITK02/ H                                                 VITK02      3
C                                                                       VITK02      4
C                                                                       VITK02      5
C                                                                       VITK03      2
      COMMON /VITK03/  FAC,XFAC,TESTFAC,ERR,NPTS,DIF                    VITK03      3
C                                                                       VITK03      4
C                                                                       VITK03      5
C     IFUNC  =  1   GAUSSIAN SHAPE FUNCTION        -   ABSORPTION CASE  IFUNC       2
C            =  2   GAUSSIAN SHAPE FUNCTION        -   DISPERSION CASE  IFUNC       3
C            =  3   GAUSSIAN (DERIVATIVE)          -   ABSORPTION CASE  IFUNC       4
C            =  4   GAUSSIAN (DERIVATIVE)          -   DISPERSION CASE  IFUNC       5
C            =  5   POWDER LINESHAPE               -   ABSORPTION CASE  IFUNC       6
C            =  6   POWDER LINESHAPE               -   DISPERSION CASE  IFUNC       7
C            =  7   POWDER LINESHAPE (DERIVATIVE)  -   ABSORPTION CASE  IFUNC       8
C            =  8   POWDER LINESHAPE (DERIVATIVE)  -   DISPERSION CASE  IFUNC       9
C                                                                       IFUNC      10
C                                                                       CNVLERR    19
C     PASS H TO FUNCTION SY                                             CNVLERR    20
C                                                                       CNVLERR    21
      H = HH                                                            CNVLERR    22
C                                                                       CNVLERR    23
C                                                                       CNVLERR    24
      IF(IFUNC .LE. 4) GO TO 110                                        CNVLERR    25
      HLO = H3 - FAC*DIF                                                CNVLERR    26
      HLOLO  = HLO  -XFAC*DIF                                           CNVLERR    27
      HLOLO  = AMAX1(0.0,HLOLO)                                         CNVLERR    28
      HUP = HONE + FAC*DIF                                              CNVLERR    29
      HUPUP  =HUP  + XFAC*DIF                                           CNVLERR    30
      HUPUP  = AMIN1(1.0E20,HUPUP)                                      CNVLERR    31
      GO TO 150                                                         CNVLERR    32
  110 CONTINUE                                                          CNVLERR    33
      DIF = HLFSTR                                                      CNVLERR    34
      HLO =  HZERO - FAC*DIF                                            CNVLERR    35
```

```
C     APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY
C
C     LATEST REVISION  JUNE, 1973
C
C     ----------------------------------------------------------------
C
C
      DATA NPRINT/6/,IPRINT/20*.TRUE./,ERCONT/20*0/,IERCNT/100*0/,
     1     NERR/20*0/
C
C     ----------------------------------------------------------------
C
      IF (JERR .LT. 0) GO TO 500
C     ---------- ERROR MESSAGE FROM FUNCTION ROUTINE ----------
      IERCNT(IFCN,JERR) = IERCNT(IFCN,JERR) + 1
      IF (IPRINT(IFCN)) WRITE(NPRINT,1001) FCN,ARG,ARG
      NERR(IFCN) = JERR
      IF (ERCONT(IFCN) .EQ. -1) GO TO 200
      IF (IPRINT(IFCN)) WRITE(NPRINT,1002) RESULT,RESULT
      IF (ERCONT(IFCN) .LT. 1) IPRINT(IFCN) = .FALSE.
      GO TO 600
C     ---------- PREPARE FOR TERMINATION AND TRACEBACK ----------
  200 DO 210 I = 1, 5
        K = 6 - I
        IF (IERCNT(IFCN,K) .NE. 0) GO TO 220
  210 CONTINUE
C     --------- TERMINATE AND TRACEBACK ----------
  220 WRITE(NPRINT,1003) (J,IERCNT(IFCN,J), J=1,K)
      CALL EXIT
C     ---------- USER INTERROGATION AND RESETTING OF FLAGS ----------
  500 JK = -JERR - 2
      IF (JK .GT. 3) GO TO 510
      J = JK + 2
      GO TO (530,520,530,550,540), J
  510 JK = IERCNT(IFCN,JK-3)
      GO TO 580
C
  520 DO 525 K = 1, 5
  525   IERCNT(IFCN,K) = 0
C
  530 IPRINT(IFCN) = .TRUE.
      GO TO 570
  540 JK = NERR(IFCN)
      NERR(IFCN) = 0
      GO TO 580
  550 IPRINT(IFCN) = .FALSE.
  570 ERCONT(IFCN) = JK
  580 JERR = JK
  600 RETURN
 1001 FORMAT(/21H ILLEGAL ARGUMENT IN ,A6,8H, ARG = ,E22.13,022)
 1002 FORMAT(6X,35HEXECUTION CONTINUING WITH RESULT = ,E22.13,022)
 1003 FORMAT(6X,43HEXECUTION TERMINATING. ERROR COUNTS FOLLOW,/
     1  (6X,10HERROR NO. ,I2,3H = ,I6))
C     ---------- LAST CARD OF FCNMON ----------
      END
```

```
      C                                                                 ELIPK      2
      C     ------- FUNPACK ------- CDC 6000 ----- SINGLE PRECISION ---------  ELIPK      3
      C                                                                 ELIPK      4
            REAL FUNCTION ELIPK(CAYSQ)                                  ELIPK      5
      C                                                                 ELIPK      6
            REAL CAYSQ,RESULT                                           ELIPK      7
            INTEGER INT                                                 ELIPK      8
      C                                                                 ELIPK      9
      C     THIS SINGLE PRECISION SUBROUTINE COMPUTES APPROXIMATE VALUES  ELIPK     10
      C     FOR THE COMPLETE ELLIPTIC INTEGRAL OF THE FIRST KIND GIVEN THE  ELIPK   11
      C     ARGUMENT  K**2.                                             ELIPK     12
      C                                                                 ELIPK     13
      C     OTHER  FUNPACK  SUBPROGRAMS REQUIRED -  NATSEK, FCNMON, AND,  ELIPK   14
      C                                                OPTIONALLY, MONERR  ELIPK  15
      C                                                                 ELIPK     16
      C     ----------------------------------------                    ELIPK     17
      C                                                                 ELIPK     18
            INT = 1                                                     ELIPK     19
            CALL NATSEK(CAYSQ,RESULT,INT)                               ELIPK     20
            ELIPK = RESULT                                              ELIPK     21
            RETURN                                                      ELIPK     22
      C     ---------- LAST CARD OF ELIPK ----------                    ELIPK     23
            END                                                         ELIPK     24
```

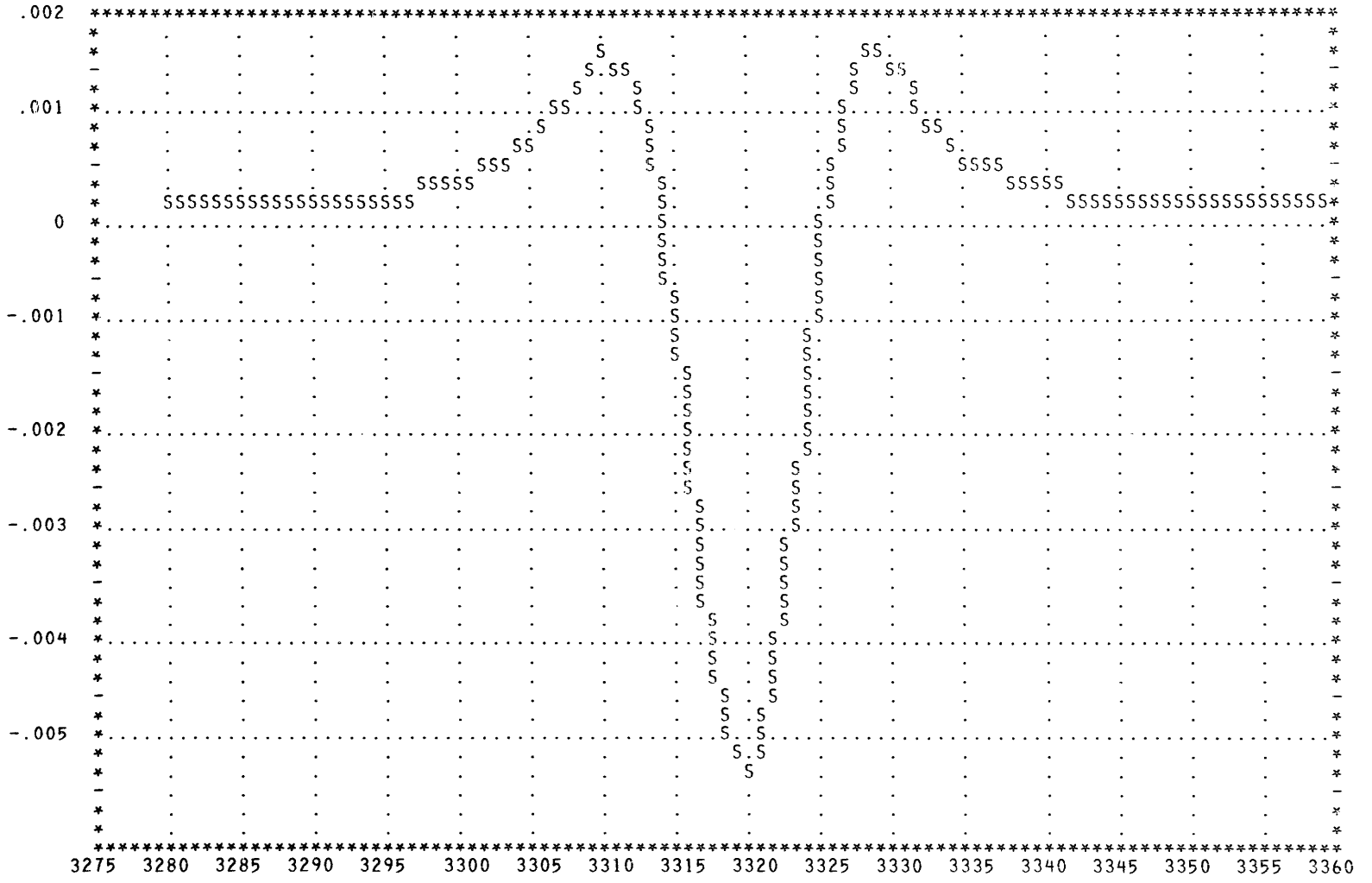
SYMBOLIC REFERENCE MAP (R=3)

```
ENTRY POINTS     DEF LINE     REFERENCES
    4  ELIPK          4          21

VARIABLES     SN  TYPE          RELOCATION
    0  CAYSQ     REAL                F.P.        REFS        6        19    DEFINED        4
   20  ELIPK     REAL                          DEFINED      20
   22  INT       INTEGER                         REFS        7        19    DEFINED       18
   21  RESULT    REAL                            REFS        6        19        20

EXTERNALS          TYPE   ARGS       REFERENCES
       NATSEK                3             19

STATISTICS
  PROGRAM LENGTH                 23B      19
```

ENTRY	ADDRESS	PROGRAM	REFERENCES							
MLUSED	16064	MLUSED	ERRCHK	15405						
			ERXSET	15717						
			CURSES	16027						
			GAUS8	16165						
			QNC7	17350						
			SPLIFT	21051						
			SPLINT	21716						
MLDATA	16112	MLDATA	MLUSED	16070						
XPUNGE	16137	XPUNGE	MLUSED	16072						
			ONECHK	21431						
GAUS8	16152	GAUS8	CONVOL	11705	11717	11723				
			CNVLERR	12110	12122	12130				
QNC7	17335	QNC7	FOUR	11265	11271					
SPLIFT	21034	SPLIFT	FOUR	11242						
ONECHK	21403	ONECHK	GAUS8	16662						
			QNC7	17703						
LUKBAK	21467	LUKBAK	ERRCHK	15427	15504					
			MLUSED	16066						
			MLDATA	16115						
MLMESS	21510	MLMESS	MLDATB	21547						
MLDATB	21542	MLDATB	ABTERM	15734						
			MLDATA	16124						
IGNORE	21564	IGNORE	MLDATA	16117						
SPLINT	21713	SPLINT	GSIN	12354						
FIXFLT	22114	FIXFLT	CLOSZIP	25421	25503	25513				
			XYPLOT	27705						
			NUMDIS	31410						
			QUICK	31702	31715	31730	31734	31746	31752	31756
				31762						
LSHIFT	22123	LSHIFT	CALCMP	25105	25142					
			PCVOUT	26442	26445	26465	26503	26526		
			SYMBOL	27427	27432					
			NUMDIS	31532						
MOD2	22131	MOD2	CALCMP	24647	25144					
			GRIDGN	31013						
MOD2R	22134		GRIDGN	31016	31020					
NEGZERO	22145	NEGZERO	SYMBOL	27316	27322					
			XYPLOT	27712	27716					
			MINMAX	30506						
			QUICK	31775	32063	32111				
NUMTYPE	22151	NUMTYPE	CALCMP	24742	24744					
			XYPLOT	27654						
WRPLT	22164	WRPLT	CALCMP	24725	24727	24731				
			PCVOUT	26554	26570					
PLTND	22200		CALCMP	25173						
ANOT	22341	ANOT	AXIS	23450						
AXIS	23236	AXIS	QUICK	32075						
GRID	23242		QUICK	32105						
CALCMP	24626	CALCMP	CLOSZIP	25443	25461	25510	25517	25535	25541	25546
				25552	25565	25571	25576	25602	25604	
			DSHPLOT	25773	26002	26013	26024	26123	26127	26143
				26150	26153	26156	26161	26163		
			SYMBOL	27340						
			XYPLOT	27601	27647	30006	30014	30034	30042	30056
				30071	30144	30173				

ENTRY	ADDRESS	PROGRAM	REFERENCES			
DXIT.SQ	45423		SKFL.SQ	43004		
GET.Z	45450	Z.SQ	FORSYS=	33644		
			NAMIN=	35601		
GET.S	45551	FSU.SQ				
GET.U	45555					
GET.F	45557					
RMU.SQ	45563					
RMU0.SQ	45570		GET.SQ	44572		
RMU2.SQ	45610		GET.SQ	44566		
RMU1.SQ	45611					
CPC03	45657	CPC	IO	46142		
CPC04	45676		IO	46324		
CPC	45725		WRPLT	22175	22201	22203
			RDPLID	32343		
			IO	46266		
CPC02	46004					
CPC999	46120		IO	46154		
IOSAV	46141	IC				
IOREAD	46160		RDPLID	32341		
IOREWRT	46163					
IOWRITE	46166					
IOIO	46171					
IOZZ	46324					
IOZW	46330					

2.602 CP SECONDS 63437B CM STORAGE USED 153 TABLE MOVES


.002
.001
0
-.001
-.002
-.003
-.004
-.005
3275 3280 3285 3290 3295 3300 3305 3310 3315 3320 3325 3330 3335 3340 3345 3350 3355 3360

BATCH CREATED 02/04/76 TODAY IS 02/05/76
THIS MESSAGE WILL APPEAR ON ALL PRINTOUTS

****

****

****

TO USERS WITH PERMANENT FILES ON MFA.

CDC INADVERTENTLY OVERWROTE A FILE ON ONE OF OUR DISK PACKS DURING MAINTENANCE THIS MORNING.

THE FILE KNOWN TO BE LOST WAS RECOVERED AND RESTORED, BUT IT IS POSSIBLE ANOTHER FILE WAS INVOLVED.

1) IF YOU HAVE PERMANENT FILE ACCESS PROBLEMS, PLEASE CHECK THE AUDIT TO DETERMINE IF THE FILE INVOLVED WAS ON UNIT 1.

IF SO, PLEASE ADVISE JOHN BENAPFL, FRED WHITWORTH, OR DICK ISLER OF THE SITUATION.

2) IF YOU HAVE AN IMPORTANT FILE ON A PERMANENT FILE DEVICE, IT SHOULD BE BACKED UP ON TAPE.

PERMANENT FILES ARE NEVER *PERMANENT*.

****

****

****

```
115                 WRITE(50,118) PILOG                                  CONSTNT    87
                118 FORMAT(*  PILOG   = *, E25.14)                       CONSTNT    88
                    WRITE(50,119) TONE                                   CONSTNT    89
                119 FORMAT(*  TONE    = *, E25.14)                       CONSTNT    90
                    WRITE(50,120) GAMMA                                  CONSTNT    91
120             120 FORMAT(*  GAMMA   = *, E25.14)                       CONSTNT    92
                    WRITE(50,121) H1STAR                                 CONSTNT    93
                121 FORMAT(*  H1STAR  = *, E25.14)                       CONSTNT    94
                    WRITE(50,122) HLFSTR                                 CONSTNT    95
                122 FORMAT(*  HLFSTR  = *, E25.14)                       CONSTNT    96
125                 WRITE(50,123)OHLFSTR                                 CONSTNT    97
                123 FORMAT(*  OHLFSTR = *, E25.14)                       CONSTNT    98
                    WRITE(50,124) DELTAH                                 CONSTNT    99
                124 FORMAT(*  DELTAH  = *, E25.14)                       CONSTNT   100
                    WRITE(50,125) DELTAHS                                CONSTNT   101
130             125 FORMAT(*  DELTAHS = *, E25.14)                       CONSTNT   102
                    RETURN                                               CONSTNT   103
                    END                                                  CONSTNT   104
```

CARD NR.	SEVERITY	DETAILS	DIAGNOSIS OF PROBLEM
27	I	1.0	DATA ITEM LIST EXCEEDS VARIABLE LIST, EXCESS CONSTANTS IGNORED.
65	I		CONSTANT TOO LONG. HIGH ORDER DIGITS RETAINED, BUT SOME PRECISION LOST.

SYMBOLIC REFERENCE MAP (R=3)

ENTRY POINTS		DEF LINE	REFERENCES
1	CONSTNT	1	131

VARIABLES		SN	TYPE	RELOCATION							
2	BETA		REAL	VITK01	REFS	16	64	68	83	DEFINED	60
31	DELTAH		REAL	VITK01	REFS	16	41	58	127		
32	DELTAHS		REAL	VITK01	REFS	16	41	59	129		
25	GAMMA		REAL	VITK01	REFS	16	76	119	DEFINED	64	
0	GNU		REAL	VITK01	REFS	16	26	41	68	79	
15	G1		REAL	VITK01	REFS	16	41	70	105		
16	G2		REAL	VITK01	REFS	16	41	71	107		
17	G3		REAL	VITK01	REFS	16	41	64	69	72	109
4	HALF		REAL	VITK01	REFS	16	62	87	DEFINED	58	
26	HISTAR		REAL	VITK01	REFS	16					
27	HLFSTR		REAL	VITK01	REFS	16	63	123	DEFINED	59	
7	HONE		REAL	VITK01	REFS	16	2*73	93	DEFINED	70	
6	HZERO		REAL	VITK01	REFS	16	91	DEFINED	69		
12	H1S		REAL	VITK01	REFS	16	99	DEFINED	73		
524	H1STAR		REAL		REFS	41	2*76	121			
10	H2		REAL	VITK01	REFS	16	2*74	95	DEFINED	71	
13	H2S		REAL	VITK01	REFS	16	101	DEFINED	74		
11	H3		REAL	VITK01	REFS	16	2*75	97	DEFINED	72	
14	H3S		REAL	VITK01	REFS	16	103	DEFINED	75		
21	IFUNC		INTEGER	VITK01	REFS	16	41	77			

```
C     INTERPOLATORY SPLINE AT 101 POINTS AND THE FIT IS USED FOR          FOUR       35
C     SUBSEQUENT EVALUATIONS. TO CAUSE THE FIT TO HAVE OTHER THAN         FOUR       36
C     101 POINTS  ONE MUST CHANGE THE COMMON STATMENT  VITK05  (IN        FOUR       37
C     BOTH FOUR AND  GSIN   ) ACCORDING TO THE FOLLOWING RULE --          FOUR       38
C        NP =  NUMBER OF FITTING POINTS                                   FOUR       39
C     COMMON /VITK05/  XH(NP),YH(NP),YP(NP),YPP(NP),WH(3*NP),NP           FOUR       40
C        AND THE VALUE OF NP MUST BE SET EITHER DIRECTLY IN FOUR OR IN    FOUR       41
C        THE USER CALLING PROGRAM THROUGH COMMON BLOCK VITK05             3JAN(76)    1
C                                                                         FOUR       43
C      USEFUL LIMITS FOR NP PROBABLY LIE BETWEEN 51 AND 501.              FOUR       44
C                                                                         FOUR       45
C                                                                         FOUR       46
C                                                                         FOUR       47
C                                                                         FOURM2      1
      EXTERNAL GSIN,GCOS                                                  FOURM2      2
      COMMON /VITK02/ CH                                                  FOURM2      3
C                                                                         FOURM2      4
      COMMON /VITK04/ CHSUBM,COMEGAM,NN                                   VITK04      2
C                                                                         FOURM2      6
      DIMENSION  HVEC(1),COFVEC(1)                                        FOUR       48
      COMMON /VITK05/  XH(101),YH(101),YP(101),YPP(101),WH(303),NP        VITK05      2
C                                                                         VITK01      2
      COMMON /VITK01/ GNU,    PLANC,   BETA,    PI,    HALF,  OVRHALF,    VITK01      3
     *                                                                    VITK01      4
     * HZERO, HONE,   H2 ,  H3 ,   H1S,   H2S,  H3S,                      VITK01      5
     *                                                                    VITK01      6
     * G1,  G2,  G3,  S,   IFUNC,    XLOG2,   PILOG,    TONE ,  GAMMA    VITK01      7
     *                                                                    VITK01      8
     * ,HISTAR, HLFSTR, DHLFSTR, DELTAH, DELTAHS                          VITK01M1    1
C                                                                         VITK01     12
C                                                                         VITK01     13
C                                                                         FOURM3     21
C                                                                         FOURM3     22
      IF(IFUNC .EQ. 1  .OR. IFUNC .EQ. 2  .OR. IFUNC .EQ. 5               FOURM3     23
     *   .OR. IFUNC .EQ. 6 )   GO TO 9                                    FOURM3     24
      DO 8  I=1,20                                                        FOURM3     25
      WRITE(50,7)                                                         FOURM3     26
    7 FORMAT(* IFUNC = 3,4,7 AND 8 ARE INVALID IN CALLS TO FOUR *,/)      FOURM3     27
    8 CONTINUE                                                            FOURM3     28
      CALL EXIT                                                           FOURM3     29
    9 CONTINUE                                                            FOURM3     30
      IF(IFUNC .GE. 5) GO TO 10                                           FOUR       51
      DIF = HLFSTR                                                        FOUR       53
      HSTART = HZERO - FAC*DIF                                            FOUR       54
      HFINAL = HZERO + FAC*DIF                                            FOUR       55
            GO TO 12                                                      FOURM2      7
   10 CONTINUE                                                            FOUR       57
      DIF = HONE - H3                                                     FOUR       58
      HSTART = H3 - FAC*DIF                                               FOUR       60
      HFINAL = HONE + FAC*DIF                                             FOUR       61
   12 CONTINUE                                                            FOURM2      8
      HINC = (HFINAL - HSTART)/(IPTS-1)                                   FOUR       63
      H =  HSTART -HINC                                                   FOUR       64
C                                                                         FOUR       65
C                                                                         FOUR       66
C                                                                         FOUR       67
C           FIT  I(H) = CONVOL(H)  FROM    HSTART -2.0HSUBM               FOUR       68
```

SYMBOLIC REFERENCE MAP (R=3)

ENTRY POINTS		DEF LINE	REFERENCES								
4	SY	1	83	90	105	111					

VARIABLES		SN	TYPE	RELOCATION								
2	BETA		REAL	VITK01	REFS	23						
31	DELTAH		REAL	VITK01	REFS	23						
32	DELTAHS		REAL	VITK01	REFS	23						
155	ELLIP		REAL		REFS	103	109	DEFINED	102	108		
141	F		REAL		REFS	44	56	DEFINED	43			
25	GAMMA		REAL	VITK01	REFS	23						
0	GNU		REAL	VITK01	REFS	23						
15	G1		REAL	VITK01	REFS	23						
16	G2		REAL	VITK01	REFS	23						
17	G3		REAL	VITK01	REFS	23						
0	H		REAL	VITK02	REFS	33	41					
4	HALF		REAL	VITK01	REFS	23						
26	HISTAR		REAL	VITK01	REFS	23						
27	HLFSTR		REAL	VITK01	REFS	23						
7	HONE		REAL	VITK01	REFS	23	88	93				
0	HPRIM		REAL	F.P.	REFS	41	75	2*88	2*92	99		
					DEFINED	1						
146	HS		REAL		REFS	93	95	96	DEFINED	92		
6	HZERO		REAL	VITK01	REFS	23	75					
12	H1S		REAL	VITK01	REFS	23	94	96				
10	H2		REAL	VITK01	REFS	23	93	99				
13	H2S		REAL	VITK01	REFS	23	94	97				
11	H3		REAL	VITK01	REFS	23	88	93				
14	H3S		REAL	VITK01	REFS	23	95	97				
21	IFUNC		INTEGER	VITK01	REFS	23	45	66				
30	OHLFSTR		REAL	VITK01	REFS	23	75					
5	OVRHALF		REAL	VITK01	REFS	23	42	44	56			
3	PI		REAL	VITK01	REFS	23	93					
23	PILOG		REAL	VITK01	REFS	23	78					
1	PLANC		REAL	VITK01	REFS	23						
20	S		REAL	VITK01	REFS	23	44					
145	SFUNC		REAL		REFS	82	104	110	DEFINED	78	80	103
					109							
136	SY		REAL		DEFINED	82	89	104	110			
24	TONE		REAL	VITK01	REFS	23						
140	T1		REAL		REFS	2*43	50	53	DEFINED	42		
144	T3		REAL		REFS	2*76	77	78	DEFINED	75	76	
147	T4		REAL		REFS	103	109	DEFINED	93			
150	T5		REAL		REFS	98	109	DEFINED	94			
151	T6		REAL		REFS	98	109	DEFINED	95			
152	T7		REAL		REFS	98	103	DEFINED	96			
153	T8		REAL		REFS	98	103	DEFINED	97			
137	X		REAL		REFS	42	DEFINED	41				
154	XK		REAL		REFS	102	107	108	DEFINED	98	107	
22	XLOG2		REAL	VITK01	REFS	23	76					
142	YA		REAL		REFS	47	50	2*53	2*56	DEFINED	44	
143	YFUNC		REAL		REFS	82	104	110	DEFINED	47	50	53
					56							

```
                HUP =  HZERO + FAC*DIF                                  CNVLERR    36
                HLOLO = HLO - FAC*DIF                                   CNVLERR    37
                HUPUP = HUP + XFAC*DIF                                  CNVLERR    38
            150 CONTINUE                                                CNVLERR    39
                ICOUNT = 0                                              CNVLERR    40
              1 CONTINUE                                                CNVLERR    41
                CALL GAUS8(SY,HLOLO,HLO,ERR,QLEFT,IERR)                 CNVLERR    42
                CALL  REHPLOT(SY,HLOLO,HLO,NPTS)                        CNVLERR    43
                IF(IERR .EQ.  1) GO TO  4                               CNVLERR    44
              2 PRINT 3, IERR                                           CNVLERR    45
              3 FORMAT(*  TROUBLE IN GAUS8 -- CALLED FROM CONVOL -- IERR =*,I3)  CNVLERR    46
                CALL EXIT                                               CNVLERR    47
              4 CONTINUE                                                CNVLERR    48
                CALL GAUS8(SY,HLO,HUP,ERR,QMIDDLE,IERR)                 CNVLERR    49
                CALL REHPLOT(SY,HLO,HUP,NPTS)                           CNVLERR    50
                IF(IERR .NE. 1) GO TO  2                                CNVLERR    51
                CALL GAUS8(SY,HUP,HUPUP,ERR,QRIGHT,IERR)                CNVLERR    52
                CALL REHPLOT(SY,HUP,HUPUP,NPTS)                         CNVLERR    53
                IF(IERR .NE. 1) GO TO 2                                 CNVLERR    54
                Q = ABS( QLEFT + QRIGHT)                                CNVLERR    55
                TEST = TESTFAC*ABS(QMIDDLE)                             CNVLERR    56
                IF( Q .LT. TEST) GO TO 7                                CNVLERR    57
                WRITE(50,10010)   QLEFT,QMIDDLE,QRIGHT,IFUNC            CNVLERR    58
          10010 FORMAT(///,* QLEFT =*,E25.14, 10X, *QMIDDLE = *,  E25.14,  CNVLERR    59
               *           * QRIGHT = *,E25.14, 10X, * IFUNC = *,I3,//)  CNVLERR    60
                ICOUNT = ICOUNT + 1                                     CNVLERR    61
                IF(ICOUNT .EQ. 3 )  GO TO  5                            CNVLERR    62
                HLO = HLO -DIF                                          CNVLERR    63
                HUP = HUP + DIF                                         CNVLERR    64
                HLOLO = HLO - XFAC*DIF                                  CNVLERR    65
                HUPUP = HUP + XFAC*DIF                                  CNVLERR    66
                GO TO 1                                                 CNVLERR    67
              5 CONTINUE                                                CNVLERR    68
                PRINT 6                                                 CNVLERR    69
              6 FORMAT(* THREE PASSES AT INTEGRATION IN CONVOL FAILED  * )  CNVLERR    70
                CALL ENDPLOT                                            CNVLERR    71
                CALL EXIT                                               CNVLERR    72
              7 CONTINUE                                                CNVLERR    73
                RETURN                                                  CNVLERR    74
                END                                                     CNVLERR    75
```

SYMBOLIC REFERENCE MAP (R=3)

```
ENTRY POINTS     DEF LINE     REFERENCES
    3  CNVLERR        1          96

VARIABLES     SN  TYPE           RELOCATION
    2  BETA       REAL               VITK01       REFS        15
   31  DELTAH     REAL               VITK01       REFS        15
   32  DELTAHS    REAL               VITK01       REFS        15
    5  DIF        REAL               VITK03       REFS        29        48        49        51        52        57        58
                                                     59        60        85        86        87        88
                                                 DEFINED       56
```

```
        SUBROUTINE FCNMON     74/74   OPT=1                          FTN 4.5+R406       02/05/76  13.03.47       PAGE     3

        SYMBOLIC REFERENCE MAP (R=3)

  ENTRY POINTS     DEF LINE     REFERENCES
      3  FCNMON          4        106

  VARIABLES     SN  TYPE           RELOCATION
      0  ARG        REAL                  F.P.        REFS          6       2*73   DEFINED        4
    214  ERCONT     INTEGER    ARRAY                  REFS          7         75        77   DEFINED       65       104
      0  FCN        REAL                  F.P.        REFS          6         73   DEFINED        4
    213  I          INTEGER                           REFS         81    DEFINED        80
    240  IERCNT     INTEGER    ARRAY                  REFS          7         72        82        85        92
                                                    DEFINED         65         72        96
      0  IFCN       INTEGER               F.P.        REFS          7       2*72        73        74        75        76      2*77
                                                        82         85         92        96        98       100       101       103
                                                       104    DEFINED          4
    430  IPRINT     LOGICAL    ARRAY                  REFS          9         73        76   DEFINED       65        77        98
                                                       103
    210  J          INTEGER                           REFS          7       2*85        91   DEFINED       85        90
      0  JERR       INTEGER               F.P.        REFS          7         70      2*72        74        88
                                                    DEFINED          4        105
    211  JK         INTEGER                           REFS          7         89        90        92       104       105
                                                    DEFINED         88         92       100
    212  K          INTEGER                           REFS          7         82        85        96   DEFINED       81        95
    404  NERR       INTEGER    ARRAY                  REFS          7        100   DEFINED        65        74       101
    135  NPRINT     INTEGER                           REFS          7    DEFINED        65   I/O REFS       73        76        85
      0  RESULT     REAL                  F.P.        REFS          6       2*76   DEFINED        4
         VARIABLES USED AS FILE NAMES, SEE ABOVE

  EXTERNALS          TYPE   ARGS       REFERENCES
         EXIT                 0             86

  STATEMENT LABELS              DEF LINE   REFERENCES
     40  200                       80         75
      0  210                       83         80
     50  220                       85         82
     67  500                       88         70
    104  510                       92         89
    110  520                       95         91
      0  525                       96         95
    120  530                       98        2*91
    122  540                      100         91
    125  550                      103         91
    127  570                      104         99
    131  580                      105         93        102
    133  600                      106         78
    160  1001     FMT             107         73
    166  1002     FMT             108         76
    175  1003     FMT             109         85

  LOOPS  LABEL     INDEX     FROM-TO    LENGTH    PROPERTIES
     41  210     * I          80 83        7B     INSTACK    EXITS
     53          * J          85 85       11B               EXT REFS
    115  525       K          95 96        2B     INSTACK

  STATISTICS
    PROGRAM LENGTH                467B     311
```

```
      SUBROUTINE  REHPLOT(FUNC,A,B,NPTS)                                REHPLOT     2
      DIMENSION   XP(501),YP(501)                                       REHPLOT     3
      AI = (B-A)/(NPTS-1)                                               REHPLOT     4
      AA = A                                                            REHPLOT     5
      DO 10010  K=1,NPTS                                                REHPLOT     6
      XP(K)  = AA                                                       REHPLOT     7
      YP(K) = FUNC(AA)                                                  REHPLOT     8
      AA=AA+AI                                                          REHPLOT     9
10010 CONTINUE                                                          REHPLOT    10
      L = +1                                                            REHPLOT    11
      ISYM = 51                                                         REHPLOT    12
      CALL QUICK(L,XP,YP,NPTS,ISYM,1H ,1,1H ,1)                         REHPLOT    13
      RETURN                                                            REHPLOT    14
      END                                                               REHPLOT    15
```

```
SYMBOLIC REFERENCE MAP (R=3)

ENTRY POINTS     DEF LINE     REFERENCES
    3  REHPLOT        1          13

VARIABLES     SN  TYPE           RELOCATION
    0  A         REAL                F.P.        REFS        3        4    DEFINED        1
   61  AA        REAL                            REFS        6        7        8    DEFINED        4        8
   60  AI        REAL                            REFS        8    DEFINED        3
    0  B         REAL                F.P.        REFS        3    DEFINED        1
   64  ISYM      INTEGER                         REFS       12    DEFINED       11
   62  K         INTEGER                         REFS        6        7    DEFINED        5
   63  L         INTEGER                         REFS       12    DEFINED       10
    0  NPTS      INTEGER             F.P.        REFS        3        5       12    DEFINED        1
   65  XP        REAL       ARRAY                REFS        2       12    DEFINED        6
 1052  YP        REAL       ARRAY                REFS        2       12    DEFINED        7

EXTERNALS          TYPE   ARGS       REFERENCES
       FUNC      REAL       1   F.P.       7
       QUICK                9             12

STATEMENT LABELS             DEF LINE   REFERENCES
    0  10010                     9          5

LOOPS  LABEL    INDEX     FROM-TO    LENGTH    PROPERTIES
   23  10010  * K            5 9       11B              EXT REFS

STATISTICS
  PROGRAM LENGTH              2053B    1067
```

ENTRY	ADDRESS	PROGRAM	REFERENCES						
			FRMADV	30721					
			GRIDGN	30771	30773	30775	30777	31001	31144	31164
				31166	31170	31172	31174			
			QUICK	31767						
			TIC	32464	32475	32477				
PLOTS	24703		PCVOUT	26423						
CHKID	25312	CHKID	CALCMP	24721						
CLOSZIP	25413	CLOSZIP	GRIDGN	31074						
ZIP	25471		GRIDGN	31102						
DSHPLOT	25736	DSHPLOT	XYPLOT	30037	30147					
NUMBER	26334	NUMBER	ANOT	22612	22674	22747				
PCVOUT	26377	PCVOUT	CALCMP	25075	25151	25165				
			SYMBOL	27437	27445					
			ENDPLOT	30647						
			PRMODE	31607						
PLADJ	26700	PLADJ	CALCMP	24717						
SCALEB	26720	SCALEB	AXIS	23332						
SYMBOL	27306	SYMBOL	ANOT	22572	22622	22677	22730	22774		
			DSHPLOT	26146						
			NUMBER	26353						
			XYPLOT	30054						
XYPLOT	27534	XYPLOT	QUICK	32135						
DASH	30175									
PLTDATA	30442	PLTDATA	CHKID	25312						
			ENDPLOT	30703						
			FRMADV	30722						
MASKER	30445	MASKER	NUMDIS	31513						
MINMAX	30461	MINMAX	QUICK	31675	31707					
NARG	30557	NARG	DSHPLOT	25747						
			XYPLOT	30207						
			MINMAX	30475						
			QUICK	31646						
NDEC	30567	NDEC	ANOT	22461						
			NUMDIS	31414						
NHOL	30624	NHOL	NUMBER	26343						
ENDPLOT	30644	ENDPLOT	TEST	6317						
			CNVLERR	12162						
FILMAV	30717	FRMADV								
FRMADV	30717		QUICK	32027						
APERTUR	30723									
GRIDGN	30737	GRIDGN	AXIS	23412						
AXISGN	31176		AXIS	23415						
NUMDIS	31374	NUMDIS	NUMBER	26341						
PRMODE	31601	PRMODE	TEST	6255						
PRPAGE	31610									
QUICK	31623	QUICK	TEST	6273	6314					
			REHPLOT	13352						
QUICKMP	32137									
QUICKML	32147									
RDPLID	32340	RDPLID	CHKID	25347						
TIC	32460	TIC	GRIDGN	31104						
RANDOM.	32526	FORUTL=								
Q8NTRY.	32613	Q8NTRY=	TEST	6253						
FECCHR.	32773	COMIO=	KODER=	34605	34672					
			NAMOUT=	37501						

```
E 10

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 3,

$END
 IFUNC    =    3
 GNU      =        .93000000000000E+10
 PLANC    =        .10545887000000E-26
 BETA     =        .92740780000000E-20
 PI       =        .31415926535898E+01
 HALF     =        .10000000000000E+00
 OVRHALF  =        .99999999999999E+01
 HZERO    =        .33195269359482E+04
 HONE     =        .33218502563053E+04
 H2       =        .33213521282053E+04
 H3       =        .33195269359482E+04
 H1S      =        .11034689125316E+08
 H2S      =        .11031379959534E+08
 H3S      =        .11019259078486E+08
 G1       =        .20003000000000E+01
 G2       =        .20006000000000E+01
 G3       =        .20017000000000E+01
 S        =        .44007492998454E-05
 XLOG2    =        .69314718055995E+00
 PILOG    =        .93943727869965E-01
 TONE     =        .25000000000000E-03
 GAMMA    =        .17602997199382E+08
 H1STAR   =        .10000000000000E-04
 HLFSTR   =        .50000000000000E+01
 OHLFSTR  =        .20000000000000E+00
 DELTAH   =        .20000000000000E+00
 DELTAHS  =        .10000000000000E+02
```

```
/////    *COMPILE   TEST
*****    *COMPILE     CONSTNT
*****    *COMPILE     SIGNAL
/////    *COMPILE    FOUR
*****    *COMPILE     SY
*****    *COMPILE     CONVOL
*****    *COMPILE     CNVLERR
/////    *COMPILE    GSIN
*****    *COMPILE     FCNMON
*****    *COMPILE     NATSEK
*****    *COMPILE     ELIPK
*****    *COMPILE     REHPLOT
```

MODIFICATIONS / CONTROL CARDS

```
CONSTNT  *CALL   VITK01
CONSTNT  *CALL  IFUNC
SIGNAL   *CALL IFUNC
SIGNAL   *CALL   VITK01
FOUR     *CALL  VITK04
FOUR     *CALL  VITK05
FOUR     *CALL   VITK01
SY       *CALL IFUNC
SY       *CALL   VITK01
SY       *CALL   VITK02
CONVOL   *CALL IFUNC
CONVOL   *CALL   VITK01
CONVOL   *CALL    VITK02
CONVOL   *CALL  VITK03
CNVLERR  *CALL   VITK01
CNVLERR  *CALL    VITK02
CNVLERR  *CALL  VITK03
CNVLERR  *CALL IFUNC
GSIN     *CALL VITK02
GSIN     *CALL VITK04
GSIN     *CALL VITK05
```

```
CONSTNT      17
CONSTNT      23
SIGNAL        8
SIGNAL       40
FOURM2        5
FOUR         49
FOUR         50
SY           10
SY           15
SY           16
CONVOL        8
CONVOL       21
CONVOL       22
CONVOL       23
CNVLERR      15
CNVLERR      16
CNVLERR      17
CNVLERR      18
GSIN         29
GSIN         30
GSIN         31
```

CORRECTION IDENTS ARE LISTED IN CHRONOLOGICAL ORDER OF INSERTION

```
IFUNC      VITK01     VITK02     VITK03     VITK04     VITK05     SIGNAL     FOUR
FOURIER    GSIN       CONSTNT    CONVOL     CNVLERR    SY         MONERR     FCNMON
NATSEK     ELIPK      ELIK1      ELIKM      REHPLOT    VITO       TEST5      FOURM1
CONSTNTM1  VITK01M1   FOURM2     FOURM3     TEST       3JAN(76)
```

```
      SUBROUTINE CONSTNT    74/74   OPT=1                          FTN 4.5+R406       02/05/76  13.03.47       PAGE     4

VARIABLES      SN  TYPE            RELOCATION
   30  OHLFSTR    REAL                 VITK01        REFS        16        67       125    DEFINED       63
    5  OVRHALF    REAL                 VITK01        REFS        16        76        89    DEFINED       62
    3  PI         REAL                 VITK01        REFS        16        67        68        85    DEFINED       65
   23  PILOG      REAL                 VITK01        REFS        16       115    DEFINED        67
    1  PLANC      REAL                 VITK01        REFS        16        64        68        81    DEFINED       61
   20  S          REAL                 VITK01        REFS        16       111    DEFINED        76
  133  SECTOM   * REAL                              DEFINED        27
    0  SETCOM     REAL        ARRAY    VITK01        REFS        25        26
  525  TEMP       REAL                              REFS        69        70        71        72    DEFINED       68
   24  TONE       REAL                 VITK01        REFS        16        41        76       117
   22  XLOG2      REAL                 VITK01        REFS        16        67       113    DEFINED       66

FILE NAMES         MODE
       INPUT      NAME                    READS        52
       TAPE50     MIXED                  WRITES        53        77        79        81        83        85        87        89
                                             91        93        95        97        99       101       103       105       107
                                            109       111       113       115       117       119       121       123       125
                                            127       129

EXTERNALS            TYPE    ARGS       REFERENCES
       ALOG        REAL       1 LIBRARY      66
       SQRT        REAL       1 LIBRARY      67

NAMELISTS          DEF LINE     REFERENCES
       VITKO           41         52         53

STATEMENT LABELS                DEF LINE    REFERENCES
  202  101      FMT                 80          79
  212  102      FMT                 82          81
  222  103      FMT                 84          83
  232  104      FMT                 86          85
  242  105      FMT                 88          87
  252  106      FMT                 90          89
  262  107      FMT                 92          91
  272  108      FMT                 94          93
  312  109      FMT                 98          97
  322  110      FMT                100          99
  332  111      FMT                102         101
  352  112      FMT                106         105
  362  113      FMT                108         107
  372  114      FMT                110         109
  402  115      FMT                112         111
  173  116      FMT                 78          77
  412  117      FMT                114         113
  422  118      FMT                116         115
  432  119      FMT                118         117
  442  120      FMT                120         119
  452  121      FMT                122         121
  462  122      FMT                124         123
  472  123      FMT                126         125
  502  124      FMT                128         127
  512  125      FMT                130         129
  342  126      FMT                104         103
  302  127      FMT                 96          95
```

```
      C                                                                 FOUR        69
      C                 TO  HFINAL + 2.0HSUBM                           FOUR        70
      C                                                                 FOUR        71
            HLO = HSTART - 2.0*HSUBM                                    FOUR        72
            HUP = HFINAL + 2.0*HSUBM                                    FOUR        73
            DATA  NP /101/                                              FOUR        74
            HHINC = (HUP-HLO)/(NP-1)                                    FOUR        75
            HH  =  HLO - HHINC                                          FOUR        76
            DO 110 J=1,NP                                               FOUR        77
            HH = HH +HHINC                                              FOUR        78
            XH(J) = HH                                                  FOUR        79
            YH(J) = CONVOL(HH)                                          FOUR        80
        110 CONTINUE                                                    FOUR        81
      C                                                                 FOUR        82
      C                                                                 FOUR        83
            ISX = 0                                                     FOUR        84
            A1 = 0.0                                                    FOUR        85
            AN = 0.0                                                    FOUR        86
            B1 = 0.0                                                    FOUR        87
            BN = 0.0                                                    FOUR        88
            CALL SPLIFT(XH,YH,YP,YPP,NP,WH,IERRH,ISX,A1,B1,AN,BN)       FOUR        89
      C                                                                 FOUR        90
      C                                                                 FOUR        91
      C         END OF FITTING OF I(H)                                  FOUR        92
      C                                                                 FOUR        93
      C                                                                 FOUR        94
      C                                                                 FOURM2       9
      C     COMPUTE THE INTEGRALS WHICH DEFINE THE FOURIER COEFFICIENTS FOURM2      10
      C                                                                 FOURM2      11
            DATA PI /3.141592653589793/                                 FOURM2      12
      C                                                                 FOURM2      13
            CHSUBM  =  HSUBM                                            FOURM2      14
            COMEGAM =  OMEGAM                                           FOURM2      15
            NN      =  N                                                FOURM2      16
            B       =  PI/OMEGAM                                        FOURM2      17
            A       =  -B                                               FOURM2      18
            C       =  1.0/B                                            FOURM2      19
            ERR     =  1.0E-5                                           FOURM2      20
            CALL ERXSET(10,0)                                           FOURM2      21
            DO 20  J=1,IPTS                                             FOUR        95
            H = H + HINC                                                FOUR        96
            CH = H                                                      FOURM2      22
            IF(IFLAG .EQ. 1)   CALL QNC7(GSIN,A,B,ERR,FCOF,IERR)        FOURM2      23
      C                                                                 FOURM2      24
            IF(IFLAG .EQ. 2)   CALL QNC7(GCOS,A,B,ERR,FCOF,IERR)        FOURM2      25
      C                                                                 FOURM2      26
            IF(IERR .EQ. 1) GO TO 15                                    FOURM2      27
            PRINT 5 , N,IFLAG,H                                         FOURM2      28
          5 FORMAT(//,* TROUBLE IN QNC7 FOR *,/,                        FOURM2      29
           1          *  N    = *,I3,/,                                 FOURM2      30
           2          *  IFLAG=*,I3,/,                                 FOURM2      31
           3          *  H    = *,E25.14,///)                           FOURM2      32
         15 CONTINUE                                                    FOURM2      33
            FCOF = C*FCOF                                               FOURM2      34
      C                                                                 FOURM2      35
            HVEC(J) = H                                                 FOUR        98
            COFVEC(J) = FCOF                                            FOUR        99
```

```
        FUNCTION SY          74/74   OPT=1                                  FTN 4.5+R406       02/05/76  13.03.47       PAGE     4

 EXTERNALS            TYPE    ARGS        REFERENCES
        ELIPK       REAL        1             102        108
        EXP         REAL        1 LIBRARY      78
        SQRT        REAL        1 LIBRARY     103        109

 STATEMENT LABELS                DEF LINE    REFERENCES
    31  1                           46        2*45
    33  2                           49        2*45
    36  3                           52        2*45
    41  4                           55        2*45
    45  70                          57         48         51         54
    57  80                          80         77
    60  90                          81         79
    63  100                         87         66
    70  120                         91         88
   117  140                        106         99

 COMMON BLOCKS    LENGTH      MEMBERS - BIAS NAME(LENGTH)
        VITKO1        27                 0 GNU     (1)                 1 PLANC  (1)                 2 BETA   (1)
                                         3 PI      (1)                 4 HALF   (1)                 5 OVRHALF(1)
                                         6 HZERO   (1)                 7 HONE   (1)                 8 H2     (1)
                                         9 H3      (1)                10 H1S    (1)                11 H2S    (1)
                                        12 H3S     (1)                13 G1     (1)                14 G2     (1)
                                        15 G3      (1)                16 S      (1)                17 IFUNC  (1)
                                        18 XLOG2   (1)                19 PILOG  (1)                20 TONE   (1)
                                        21 GAMMA   (1)                22 HISTAR (1)                23 HLFSTR (1)
                                        24 OHLFSTR(1)                 25 DELTAH (1)                26 DELTAHS(1)
        VITKO2         1                 0 H       (1)

 STATISTICS
   PROGRAM LENGTH                   156B      110
   CM LABELED COMMON LENGTH          34B       28
```

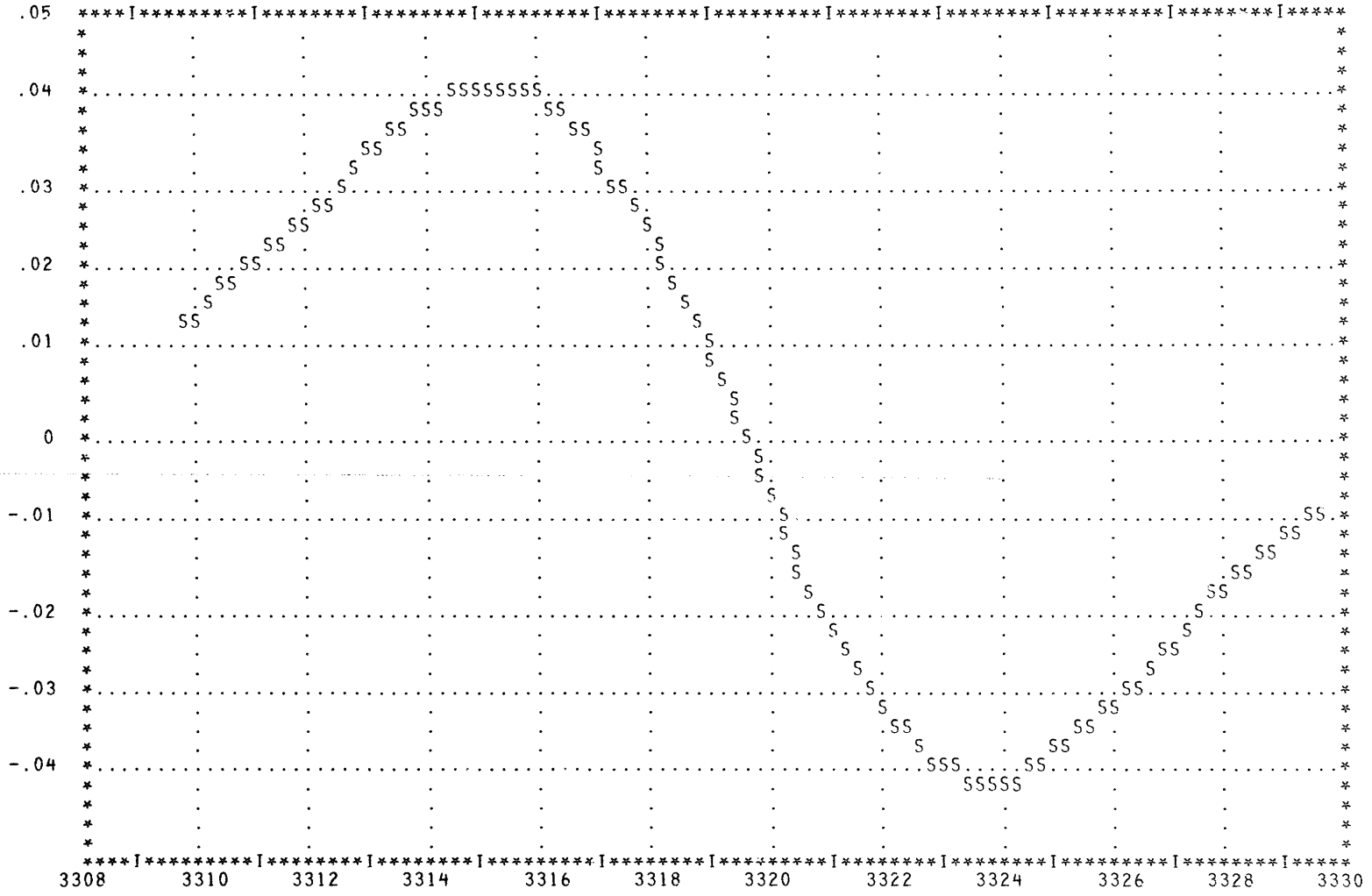
```
VARIABLES      SN  TYPE          RELOCATION
     3  ERR        REAL              VITK03      REFS        29        64        71        74
     0  FAC        REAL              VITK03      REFS        29        48        51        57        58        59
    25  GAMMA      REAL              VITK01      REFS        15
     0  GNU        REAL              VITK01      REFS        15
    15  G1         REAL              VITK01      REFS        15
    16  G2         REAL              VITK01      REFS        15
    17  G3         REAL              VITK01      REFS        15
     0  H          REAL              VITK02      REFS        25   DEFINED        44
     4  HALF       REAL              VITK01      REFS        15
     0  HH         REAL               F.P.       REFS        44   DEFINED         1
    26  HISTAR     REAL              VITK01      REFS        15
    27  HLFSTR     REAL              VITK01      REFS        15        56
   227  HLO        REAL                          REFS        49        59        64        65        71        72        85
                                                   87   DEFINED        48        57        85
   230  HLOLO      REAL                          REFS        50        64        65   DEFINED        49        50        59
                                                   87
     7  HONE       REAL              VITK01      REFS        15        51
   231  HUP        REAL                          REFS        53        60        71        72        74        75        86
                                                   88   DEFINED        51        58        86
   232  HUPUP      REAL                          REFS        53        74        75   DEFINED        52        53        60
                                                   88
     6  HZERO      REAL              VITK01      REFS        15        57        58
    12  H1S        REAL              VITK01      REFS        15
    10  H2         REAL              VITK01      REFS        15
    13  H2S        REAL              VITK01      REFS        15
    11  H3         REAL              VITK01      REFS        15        48
    14  H3S        REAL              VITK01      REFS        15
   233  ICOUNT     INTEGER                       REFS        83        84   DEFINED        62        83
   235  IERR       INTEGER                       REFS        64        66        67        71        73        74        76
    21  IFUNC      INTEGER           VITK01      REFS        15        47        80
     4  NPTS       INTEGER           VITK03      REFS        29        65        72        75
    30  OHLFSTR    REAL              VITK01      REFS        15
     5  OVRHALF    REAL              VITK01      REFS        15
     3  PI         REAL              VITK01      REFS        15
    23  PILOG      REAL              VITK01      REFS        15
     1  PLANC      REAL              VITK01      REFS        15
   240  Q          REAL                          REFS        79   DEFINED        77
   234  QLEFT      REAL                          REFS        64        77        80
   236  QMIDDLE    REAL                          REFS        71        78        80
   237  QRIGHT     REAL                          REFS        74        77        80
    20  S          REAL              VITK01      REFS        15
   241  TEST       REAL                          REFS        79   DEFINED        78
     2  TESTFAC    REAL              VITK03      REFS        29        78
    24  TONE       REAL              VITK01      REFS        15
     1  XFAC       REAL              VITK03      REFS        29        49        52        60        87        88
    22  XLOG2      REAL              VITK01      REFS        15

FILE NAMES        MODE
        OUTPUT    FMT                 WRITES      67        91
        TAPE50    FMT                 WRITES      80

EXTERNALS          TYPE   ARGS       REFERENCES
        ENDPLOT            0            93
        EXIT               0            69        94
        GAUS8              6            64        71        74
        REHPLOT            4            65        72        75
        SY                 0            13        64        65        71        72        74        75
```

```
      C                                                                 NATSEK     2
      C     ------- FUNPACK ------- CDC 6000 ----- SINGLE PRECISION ---------  NATSEK     3
      C                                                                 NATSEK     4
            SUBROUTINE NATSEK(ARG,RESULT,INT)                           NATSEK     5
      C                                                                 NATSEK     6
            REAL A(10),ARG,B(10),CAY,CAYSQ,DELIPK,EPS,ETA,FCN(3),RESULT,  NATSEK     7
           1      SUMA,SUMB,XIND                                        NATSEK     8
            INTEGER I,IFCN,INT,JK                                       NATSEK     9
      C                                                                 NATSEK    10
      C     THIS  FUNPACK  PACKET CONTAINS THREE FUNCTION TYPE SUBPROGRAMS  NATSEK    11
      C     ELIPK, ELIK1, AND  ELIKM, AND ONE SUBROUTINE TYPE           NATSEK    12
      C     SUBPROGRAM, NATSEK.  THE PACKET COMPUTES SINGLE PRECISION VALUES  NATSEK    13
      C     OF THE COMPLETE ELLIPTIC INTEGRALS OF THE FIRST KIND        NATSEK    14
      C                                                                 NATSEK    15
      C            K(CAYSQ),  K(CAY**2), AND  K(1-ETA)                  NATSEK    16
      C                                                                 NATSEK    17
      C     FOR THE PARAMETERS  CAYSQ, CAY, AND  ETA, RESPECTIVELY, WHERE  NATSEK    18
      C                                                                 NATSEK    19
      C            K(K**2)=INTEGRAL (FROM PHI=0 TO PHI=PI/2)            NATSEK    20
      C                          1/SQRT(1-(K**2)*(SIN(PHI))**2)         NATSEK    21
      C                                   0 .LE. K**2 .LT. 1.           NATSEK    22
      C                                                                 NATSEK    23
      C     THE FORTRAN CALLING STATEMENTS FOR THE PRIMARY ENTRIES TO THIS  NATSEK    24
      C     PACKET ARE                                                  NATSEK    25
      C                                                                 NATSEK    26
      C                  Y = ELIPK(CAYSQ),                              NATSEK    27
      C                                                                 NATSEK    28
      C                  Y = ELIK1(CAY),                                NATSEK    29
      C     AND                                                         NATSEK    30
      C                  Y = ELIKM(ETA),                                NATSEK    31
      C                                                                 NATSEK    32
      C     WHERE THE ENTRY POINTS CORRESPOND TO THE FUNCTIONS K(CAYSQ),  NATSEK    33
      C     K(CAY**2) AND K(1-ETA), RESPECTIVELY.  THE ROUTINE  NATSEK  IS  NATSEK    34
      C     INTENDED FOR INTERNAL PACKET USE ONLY, ALL COMPUTATIONS WITHIN  NATSEK    35
      C     THE PACKET BEING CONCENTRATED IN THIS ROUTINE.  THE FUNCTION  NATSEK    36
      C     SUBPROGRAMS INVOKE  NATSEK  WITH THE FORTRAN STATEMENT      NATSEK    37
      C          CALL NATSEK(ARG,RESULT,INT)                            NATSEK    38
      C     WHERE THE PARAMETER USAGE IS AS FOLLOWS                     NATSEK    39
      C                                                                 NATSEK    40
      C      FUNCTION                  PARAMETERS FOR  NATSEK           NATSEK    41
      C        CALL                 ARG                    RESULT     INT  NATSEK    42
      C      ELIPK(CAYSQ)   CAYSQ, 0 .LE. CAYSQ .LT. 1    K(CAYSQ)     1  NATSEK    43
      C      ELIK1(CAY)      CAY, -1 .LT. CAY .LT. 1      K(CAY**2)    2  NATSEK    44
      C      ELIKM(ETA)      ETA,  0 .LT. ETA .LE. 1      K(1-ETA)     3  NATSEK    45
      C                                                                 NATSEK    46
      C     THE MAIN COMPUTATION INVOLVES EVALUATION OF NEAR MINIMAX    NATSEK    47
      C     APPROXIMATIONS PUBLISHED IN MATH. COMP. 19, 105-112(1965)   NATSEK    48
      C     BY CODY.                                                    NATSEK    49
      C                                                                 NATSEK    50
      C     OTHER SUBPROGRAMS REQUIRED                                  NATSEK    51
      C                                                                 NATSEK    52
      C     FROM  FUNPACK  -  FCNMON  AND, OPTIONALLY, MONERR  (BOTH    NATSEK    53
      C                        CONTAINED IN THE  FUNPACK  PACKET  MONEPR)  NATSEK    54
      C     OTHER          -  ABS, ALOG                                 NATSEK    55
      C                                                                 NATSEK    56
      C     ACCURACY                                                    NATSEK    57
      C                                                                 NATSEK    58
```

```
SCOPE 3.4.3    -    CONTROL-CARD-INITIATED LOAD            LOADER    P393       02/05/76  13.04.25.      PAGE     1


FWA OF THE LOAD            101
LWA+1 OF THE LOAD        46333

TRANSFER ADDRESS -- TEST            6253


PROGRAM AND BLOCK ASSIGNMENTS.

BLOCK         ADDRESS    LENGTH          FILE                        PREFIX TABLE CONTENTS

TEST              101      6655          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
/VITK01/         6756        33
CONSTNT          7011       526          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
SIGNAL           7537        72          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
/VITK02/         7631         1
/VITK04/         7632         3
/VITK05/         7635      1304
FOUR            11141       315          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
SY              11456       156          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
/VITK03/        11634         6
CONVOL          11642       212          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
CNVLERR         12054       242          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
GSIN            12316       106          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
FCNMON          12424       467          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
NATSEK          13113       154          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
ELIPK           13267        23          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
REHPLOT         13312      2053          LGO        02/05/76  13.03.47  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
/MLBLK0/        15365         2
ERRCHK          15367       322      UL-MATHFTN     02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
ERXSET          15711        15      UL-MATHFTN     02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
ABTERM          15726        47      UL-MATHFTN     02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
REMARX          15775        24      UL-MATHFTN     02/04/76  12.06.24. SCOPE 3.4 COMPASS 3.75203    2
CURSES          16021        42      UL-MATHFTN     02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
MLUSED          16063        24      UL-MATHFTN     02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
MLDATA          16107        30      UL-MATHFTN     02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                     OPT=1
XPUNGE          16137        10      UL-MATHFTN     02/04/76  12.06.43. SCOPE 3.4 COMPASS 3.75203    2
GAUS8           16147      1163      UL-MATHFTN     02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
```

ENTRY	ADDRESS	PROGRAM	REFERENCES						
FECPRT.	33013		INCOM=	34425					
			FMTAP=	37101					
FECC63.	33051		OUTC=	37702					
EOF	33070	EOF	TEST	6263					
FEOFAL.	33103	FLTOUT=	KODER=	34767	35006	35027			
			NAMOUT=	37433					
FEOEOV.	33125		KODER=	35006					
FEOEXP.	33127		KODER=	34770					
			NAMOUT=	37435					
FEORND.	33164		KODER=	34755	35003	35024			
			NAMOUT=	37425					
FEOSCA.	33221		KODER=	34745	35000	35020			
			NAMOUT=	37422	37437				
FEOZRO.	33305		KODER=	34756	35004	35025			
MEMIT.	33413	FORSYS=	Q8.IO.	32605					
END.	33454		TEST	6322					
EXIT	33501		TEST	6321					
			FOUR	11173					
			CONVOL	11715					
			CNVLERR	12120	12164				
			FCNMON	12512					
STOP.	33503		ABTERM	15742					
ABNORM.	33512		GOTOER=	36127					
SYSARG=	33543								
IOERR.	33562		NAMIN=	35636					
			NAMOUT=	37472					
			OUTC=	37721					
SYSEND.	33606		SYSTEM	36117					
SYP=5	33607								
CLSLNK.	33623								
SYSERR.	33665		Q8.IO.	32563					
			COMIO=	33034	33047				
			INCOM=	34432					
			NAMIN=	35645	35664				
			SYSTEM	36113	36114				
			GOTOER=	36126					
			FMTAP=	37120					
			GETFIT=	37230					
			NAMOUT=	37467					
			OUTC=	37726					
SYST1A.	33726								
SYP=1	33732								
SYP=3	34007								
SYP=4	34015								
SYS=6	34045								
SYS2=	34071		INCOM=	34437					
FECOPE.	34140		NAMIN=	35333					
			NAMOUT=	37300					
			OUTC=	37620					
FEIGNC.	34232	INCOM=	NAMIN=	35537	35550	35555			
			FLTIN=	36440	36444	36447	36450	36451	
FEISBL.	34247		NAMIN=	35375	35422	35513	35532	35544	35574 35607
				35610	35723				
FEINUM.	34270		NAMIN=	35521	35524	35530			
FEIFSG.	34301		NAMIN=	35567					

```
F  9

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 1,

$END
 IFUNC    =    1
 GNU      =        .93000000000000E+10
 PLANC    =        .10545887000000E-26
 BETA     =        .92740780000000E-20
 PI       =        .31415926535898E+01
 HALF     =        .10000000000000E+00
 OVRHALF  =        .99999999999999E+01
 HZERO    =        .33195269359482E+04
 HONE     =        .33218502563053E+04
 H2       =        .33213521282053E+04
 H3       =        .33195269359482E+04
 H1S      =        .11034689125316E+08
 H2S      =        .11031379959534E+08
 H3S      =        .11019259078486E+08
 G1       =        .20003000000000E+01
 G2       =        .20006000000000E+01
 G3       =        .20017000000000E+01
 S        =        .44007492998454E-05
 XLOG2    =        .69314718055995E+00
 PILOG    =        .93943727869965E-01
 TONE     =        .25000000000000E-03
 GAMMA    =        .17602997199382E+08
 H1STAR   =        .10000000000000E-04
 HLFSTR   =        .50000000000000E+01
 OHLFSTR  =        .20000000000000E+00
 DELTAH   =        .20000000000000E+00
 DELTAHS  =        .10000000000000E+02
```


.05
.04
.03
.02
.01
0
-.01
-.02
-.03
-.04
3308
3310
3312
3314
3316
3318
3320
3322
3324
3326
3328
3330

PLOT 5

DECK LIST AS READ FROM OLDPL PLUS ADDED NEW DECKS

YANK$$$	IFUNC	VITK01	VITK02	VITK03	VITK04	VITK05	SIGNAL
FOUR	GSIN	CONSTNT	CONVOL	CNVLERR	SY	FCNMON	NATSEK
ELIPK	REHPLOT	TEST					

DECKS ARE LISTED IN THE ORDER OF THEIR OCCURRENCE ON A NEW PROGRAM LIBRARY IF ONE IS CREATED BY THIS UPDATE

YANK$$$	IFUNC	VITK01	VITK02	VITK03	VITK04	VITK05	SIGNAL
FOUR	GSIN	CONSTNT	CONVOL	CNVLERR	SY	FCNMON	NATSEK
ELIPK	REHPLOT	TEST					

COMMON DECKS ENCOUNTERED

IFUNC	VITK01	VITK02	VITK03	VITK04	VITK05

DECKS WRITTEN TO COMPILE FILE

TEST	CONSTNT	SIGNAL	FOUR	SY	CONVOL	CNVLERR	GSIN
FCNMON	NATSEK	ELIPK	REHPLOT				

THIS UPDATE REQUIRED 34100B WORDS OF CORE.

```
 COMMON BLOCKS   LENGTH     MEMBERS - BIAS NAME(LENGTH)
        VITK01      27               0 GNU     (1)          1 PLANC  (1)          2 BETA   (1)
                                     3 PI      (1)          4 HALF   (1)          5 OVRHALF(1)
                                     6 HZERO   (1)          7 HONE   (1)          8 H2     (1)
                                     9 H3      (1)         10 H1S    (1)         11 H2S    (1)
                                    12 H3S     (1)         13 G1     (1)         14 G2     (1)
                                    15 G3      (1)         16 S      (1)         17 IFUNC  (1)
                                    18 XLOG2   (1)         19 PILOG  (1)         20 TONE   (1)
                                    21 GAMMA   (1)         22 HISTAR (1)         23 HLFSTR (1)
                                    24 OHLFSTR(1)          25 DELTAH (1)         26 DELTAHS(1)

 EQUIV CLASSES   LENGTH     MEMBERS - BIAS NAME(LENGTH)
GNU     GNU          27               0 SETCOM (27)

 STATISTICS
   PROGRAM LENGTH                 526B      342
   CM LABELED COMMON LENGTH        33B       27
```

```
      20 CONTINUE                                                    FOUR      100
         RETURN                                                      FOUR      101
         END                                                         FOUR      102
```

CARD NR. SEVERITY DETAILS DIAGNOSIS OF PROBLEM

144 I CONSTANT TOO LONG. HIGH ORDER DIGITS RETAINED, BUT SOME PRECISION LOST.

SYMBOLIC REFERENCE MAP (R=3)

ENTRY POINTS		DEF LINE	REFERENCES
3	FOUR	1	173

VARIABLES		SN	TYPE		RELOCATION								
267	A		REAL			REFS	157	159	DEFINED	150			
262	AN		REAL			REFS	135	DEFINED	132				
261	A1		REAL			REFS	135	DEFINED	131				
266	B		REAL			REFS	150	151	157	159	DEFINED	149	
2	BETA		REAL		VITK01	REFS	80						
264	BN		REAL			REFS	135	DEFINED	134				
263	B1		REAL			REFS	135	DEFINED	133				
270	C		REAL			REFS	168	DEFINED	151				
0	CH		REAL		VITK02	REFS	73	DEFINED	156				
0	CHSUBM		REAL		VITK04	REFS	75	DEFINED	146				
0	COFVEC		REAL	ARRAY	F.P.	REFS	77	DEFINED	1	171			
1	COMEGAM		REAL		VITK04	REFS	75	DEFINED	147				
31	DELTAH		REAL		VITK01	REFS	80						
32	DELTAHS		REAL		VITK01	REFS	80						
246	DIF		REAL			REFS	101	102	106	107	DEFINED	100	105
271	ERR		REAL			REFS	157	159	DEFINED	152			
0	FAC		REAL		F.P.	REFS	101	102	106	107	DEFINED	1	
272	FCOF		REAL			REFS	157	159	168	171	DEFINED	168	
25	GAMMA		REAL		VITK01	REFS	80						
0	GNU		REAL		VITK01	REFS	80						
15	G1		REAL		VITK01	REFS	80						
16	G2		REAL		VITK01	REFS	80						
17	G3		REAL		VITK01	REFS	80						
252	H		REAL			REFS	155	156	162	170	DEFINED	110	155
4	HALF		REAL		VITK01	REFS	80						
250	HFINAL		REAL			REFS	109	119	DEFINED	102	107		
256	HH		REAL			REFS	124	125	126	DEFINED	122	124	
255	HHINC		REAL			REFS	122	124	DEFINED	121			
251	HINC		REAL			REFS	110	155	DEFINED	109			
26	HISTAR		REAL		VITK01	REFS	80						
27	HLFSTR		REAL		VITK01	REFS	80	100					
253	HLO		REAL			REFS	121	122	DEFINED	118			
7	HONE		REAL		VITK01	REFS	80	105	107				
247	HSTART		REAL			REFS	109	110	118	DEFINED	101	106	
0	HSUBM		REAL		F.P.	REFS	118	119	146	DEFINED	1		
254	HUP		REAL			REFS	121	DEFINED	119				

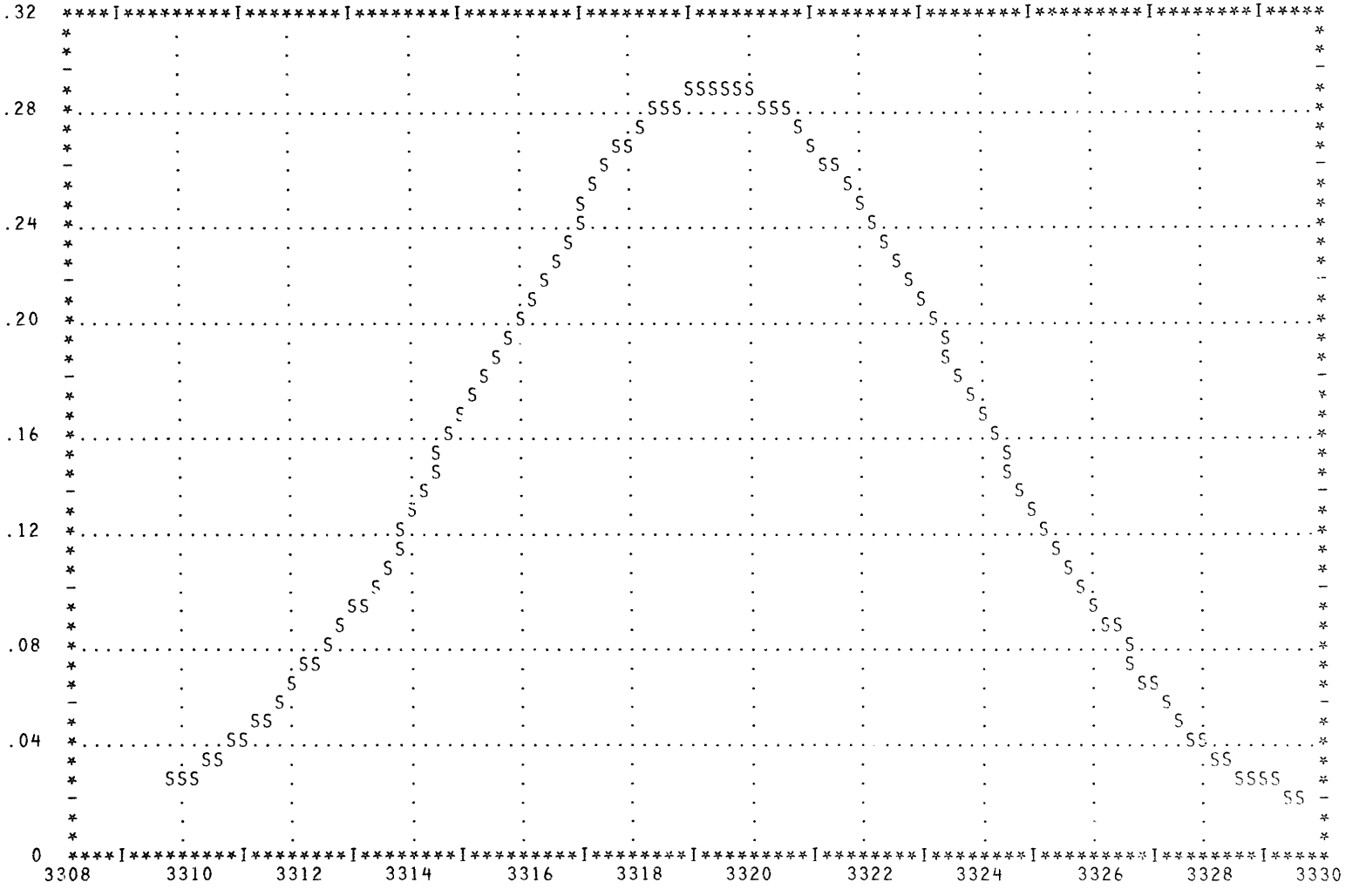
```
      FUNCTION   CONVOL(HH)                                             CONVOL     2
C                                                                       CONVOL     3
C       LET I(H) = INTEGRAL(0,INFINITY)( SY(HPRIM) ) WHERE SY IS GIVEN  CONVOL     4
C     BY SUBROUTINE SY ,THE VALUE OF WHICH IS DETERMINED BY IFUNC       CONVOL     5
C     ACCORDING TO THE RULE                                             CONVOL     6
C                                                                       CONVOL     7
C     IFUNC  =  1   GAUSSIAN SHAPE FUNCTION        -   ABSORPTION CASE  IFUNC      2
C            =  2   GAUSSIAN SHAPE FUNCTION        -   DISPERSION CASE  IFUNC      3
C            =  3   GAUSSIAN (DERIVATIVE)          -   ABSORPTION CASE  IFUNC      4
C            =  4   GAUSSIAN (DERIVATIVE)          -   DISPERSION CASE  IFUNC      5
C            =  5   POWDER LINESHAPE               -   ABSORPTION CASE  IFUNC      6
C            =  6   POWDER LINESHAPE               -   DISPERSION CASE  IFUNC      7
C            =  7   POWDER LINESHAPE (DERIVATIVE)  -   ABSORPTION CASE  IFUNC      8
C            =  8   POWDER LINESHAPE (DERIVATIVE)  -   DISPERSION CASE  IFUNC      9
C                                                                       IFUNC     10
C                                                                       CONVOL     9
C     FUNCTION CONVOL(H) CALCULATES I(H). TO BE MORE PRECISE, CONVOL    CONVOL    10
C     DETERMINES THE PORTION OF (0,INFINITY) OVER WHICH THE FUNCTION(S) CONVOL    11
C     SY SIGNIFICANTLY CONTRIBUTE TO THE INTEGRAL DEFININING I(H).      CONVOL    12
C     CONVOL CALLS UPON SUBROUTINE GAUS8 TO CARRY OUT THE FINITE        CONVOL    13
C     INTERVAL INTEGRATIONS.                                            CONVOL    14
C                                                                       CONVOL    15
C                                                                       CONVOL    16
C                                                                       CONVOL    17
C                                                                       CONVOL    18
C                                                                       CONVOL    19
      EXTERNAL SY                                                       CONVOL    20
C                                                                       VITK01     2
      COMMON /VITK01/ GNU,    PLANC,   BETA,    PI,    HALF,  OVRHALF,  VITK01     3
     *                                                                  VITK01     4
     * HZERO, HONE,   H2 ,  H3 ,   H1S,   H2S,  H3S,                    VITK01     5
     *                                                                  VITK01     6
     * G1,  G2,  G3,  S,   IFUNC,    XLOG2,   PILOG,    TONE ,  GAMMA  VITK01     7
     *                                                                  VITK01     8
     * ,HISTAR, HLFSTR, OHLFSTR, DELTAH, DELTAHS                        VITK01M1   1
C                                                                       VITK01    12
C                                                                       VITK01    13
C                                                                       VITK02     2
      COMMON /VITK02/ H                                                 VITK02     3
C                                                                       VITK02     4
C                                                                       VITK02     5
C                                                                       VITK03     2
      COMMON /VITK03/  FAC,XFAC,TESTFAC,ERR,NPTS,DIF                    VITK03     3
C                                                                       VITK03     4
C                                                                       VITK03     5
C                                                                       CONVOL    25
C     PASS H TO FUNCTION SY                                             CONVOL    26
C                                                                       CONVOL    27
      H = HH                                                            CONVOL    28
C                                                                       CONVOL    29
C                                                                       CONVOL    30
      ERR = 1.0E-6                                                      CONVOL    31
      IF(IFUNC .LE. 4) GO TO 110                                        CONVOL    32
      DIF = (HONE - H3) +.001                                           CONVOL    33
      FAC  = 0.1                                                        CONVOL    34
      XFAC = 20.0                                                       CONVOL    35
      HLO  =  H3 - FAC*DIF                                              CONVOL    36
```

```
      SUBROUTINE CNVLERR    74/74   OPT=1                          FTN 4.5+R406       02/05/76  13.03.47       PAGE     4

INLINE FUNCTIONS   TYPE   ARGS        DEF LINE  REFERENCES
       ABS        REAL      1  INTRIN                77          78
       AMAX1      REAL      0  INTRIN                50
       AMIN1      REAL      0  INTRIN                53

STATEMENT LABELS              DEF LINE   REFERENCES
   33  1                         63         89
   41  2                         67         73         76
  162  3        FMT              68         67
   45  4                         70         66
  103  5                         90         84
  215  6        FMT              92         91
  111  7                         95         79
   22  110                       55         47
   32  150                       61         54
  200  10010    FMT              81         80

COMMON BLOCKS   LENGTH     MEMBERS - BIAS NAME(LENGTH)
       VITK01       27              0 GNU    (1)                1 PLANC  (1)                2 BETA   (1)
                                    3 PI     (1)                4 HALF   (1)                5 OVRHALF(1)
                                    6 HZERO  (1)                7 HONE   (1)                8 H2     (1)
                                    9 H3     (1)               10 H1S    (1)               11 H2S    (1)
                                   12 H3S    (1)               13 G1     (1)               14 G2     (1)
                                   15 G3     (1)               16 S      (1)               17 IFUNC  (1)
                                   18 XLOG2  (1)               19 PILOG  (1)               20 TONE   (1)
                                   21 GAMMA  (1)               22 HISTAR (1)               23 HLFSTR (1)
                                   24 OHLFSTR(1)               25 DELTAH (1)               26 DELTAHS(1)
       VITK02        1               0 H      (1)
       VITK03        6               0 FAC    (1)                1 XFAC   (1)                2 TESTFAC(1)
                                     3 ERR    (1)                4 NPTS   (1)                5 DIF    (1)

STATISTICS
  PROGRAM LENGTH                 242B      162
  CM LABELED COMMON LENGTH        42B       34
```

```
C     THIS SUBROUTINE IS DESIGNED TO GIVE NEAR OPTIMAL ACCURACY
C     IN SINGLE PRECISION ARITHMETIC ON A CDC 6000 SERIES COMPUTER.
C     THE FOLLOWING TABLE GIVES THE RESULTS OF RANDOM ARGUMENT
C     ACCURACY TESTS, FOR ARGUMENTS DRAWN FROM THE INTERVAL (0, 1),
C     WHICH WERE RUN AT THE U. OF TEXAS IN OCTOBER, 1972.
C
C        ENTRY POINT   FREQUENCY OF BIT ERRORS   MAX        RMS
C                       NO. OF BITS IN ERROR      REL.       REL.
C                          0     1     2          ERROR      ERROR
C
C          ELIPK        1247   751     2        8.906E-15  2.956E-15
C          ELIK1        1243   757     0        7.104E-15  2.842E-15
C          ELIKM        1238   762     0        7.104E-15  2.954E-15
C       ----------
C
C     ERROR MONITORING
C
C     THE FOLLOWING TABLE INDICATES THE TYPES OF ERROR THAT MAY
C     BE ENCOUNTERED IN THIS ROUTINE AND THE FUNCTION VALUE SUPPLIED
C     IN EACH CASE (XIND IS THE INDEFINITE OPERAND).
C
C       ERROR       ERROR                       FUNCTION VALUES FOR
C        NO.                                      ELIPK      ELIK1
C                                                OR ELIKM
C         1   K**2 .LT. 0, OR ETA .GT. 1           XIND        -
C         2   K**2 .EQ. 1, OR ETA .EQ. 0           XIND       XIND
C         3   K**2 .GT. 1, OR ETA .LT. 0           XIND       XIND
C
C     ALL ERROR PROCESSING IS HANDLED THROUGH THE SEPARATE  FUNPACK
C     PACKET  MONERR.  THE DEFAULT ERROR PROCEDURE IS TO PRINT
C     AN ERROR MESSAGE ONLY FOR THE FIRST ERROR ENCOUNTERED.  THE
C     FREQUENCY OF EACH TYPE OF ERROR IS ACCUMULATED, AND EXECUTION
C     IS CONTINUED.  FOR THE  ELIPK  PACKET,  MONERR  MAY BE INVOKED
C     TO ALTER THESE CONVENTIONS, OR TO INTERROGATE THE ERROR STATUS,
C     BY THE FORTRAN STATEMENT
C                JK = MONERR(KK,2),
C     WHERE  JK  AND  KK  ARE INTEGERS, RELATED ACCORDING TO THE
C     FOLLOWING TABLE.  EC-N DENOTES THE ERROR COUNT FOR ERROR
C     NUMBER  N  IN THE ABOVE TABLE, AND  --  DENOTES NO CHANGE
C     FROM THE PREVIOUS SETTING.
C
C                                       SUBSEQUENT ACTION
C          KK    JK     ERROR           UPON FINDING ERROR
C                       COUNTS           PRINT    TRACEBACK
C                                        MSSG.     AND STOP
C       .LT. -1  0       --               --         --
C         -1     KK      --               YES        YES
C          0     KK   RESET TO 0        1 ONLY      NO
C          1     KK      --            EVERY TIME   NO
C          2     KK      --               NO        NO
C          3      *      --               --        --
C          4     EC-1    --               --        --
C          5     EC-2    --               --        --
C          6     EC-3    --               --        --
C       .GT. 6   0       --               --        --
C
C     *  THE RETURN FOR KK=3 IS THE ERROR NUMBER N (FROM THE
```

```
BLOCK        ADDRESS   LENGTH        FILE                      PREFIX TABLE CONTENTS

                                                   OPT=1
QNC7           17332     1477     UL-MATHFTN      02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                   OPT=1
SPLIFT         21031      347     UL-MATHFTN      02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                   OPT=1
ONECHK         21400       65     UL-MATHFTN      02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                   OPT=1
LUKBAK         21465       23     UL-MATHFTN      02/04/76  12.06.23. SCOPE 3.4 COMPASS 3.75203    2
MLMESS         21510       27     UL-MATHFTN      02/04/76  12.06.24. SCOPE 3.4 COMPASS 3.75203    2
MLDATB         21537       21     UL-MATHFTN      02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                   OPT=1
IGNORE         21560      130     UL-MATHFTN      02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                   OPT=1
SPLINT         21710      203     UL-MATHFTN      02/04/76  12.06.13  SCOPE 3.4 FTN    4.5R406 666X  I
                                                   OPT=1
FIXFLT         22113        7     SL-FTNLIBY      05/31/75  11.54.39. SCOPE 3.4 COMPASS 3.75050    2
LSHIFT         22122        6     SL-FTNLIBY      05/31/75  11.54.58. SCOPE 3.4 COMPASS 3.75050    2
                                                   SLL LSHIFT.  FTN VERSION (YOU SHOULD USE SHIFT)
MOD2           22130       14     SL-FTNLIBY      05/31/75  11.55.03. SCOPE 3.4 COMPASS 3.75050    2
NEGZERO        22144        5     SL-FTNLIBY      05/31/75  11.55.08. SCOPE 3.4 COMPASS 3.75050    2
NUMTYPE        22151       11     SL-FTNLIBY      05/31/75  11.55.09. SCOPE 3.4 COMPASS 3.75050    2
/SET835/       22162        1
WRPLT          22163       30     SL-FTNLIBY      05/31/75  11.55.11. SCOPE 3.4 COMPASS 3.75050    2
/PLOTTER/      22213      113
/PL835ID/      22326       10
ANOT           22336      657     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
/SQUARE/       23215        1
/INCSZ/        23216       15
AXIS           23233      351     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
/LOCAT/        23604        2
/PLTBUF/       23606     1000
/IPLT/         24606        1
/PLFACT/       24607       10
/RASTER/       24617        4
CALCMP         24623      466     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
CHKID          25311       72     SL-FTNLIBY      05/31/75  11.55.56. SCOPE 3.4 COMPASS 3.75050    2
/BLANK/        25403        5
CLOSZIP        25410      323     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
DSHPLOT        25733      376     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
NUMBER         26331       43     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
PCVOUT         26374      303     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
PLADJ          26677       16     SL-FTNLIBY      05/31/75  11.56.36. SCOPE 3.4 COMPASS 3.75050    2
SCALEB         26715      366     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
SYMBOL         27303      226     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
XYPLOT         27531      700     SL-FTNLIBY      05/31/75  11.54.36. SCOPE 3.4 FTN    4.3P393 666X  I
                                                   OPT=2
```

ENTRY	ADDRESS	PROGRAM	REFERENCES						
FEIBLK=	34344		NAMIN=	35341	35343				
FEIBLK.	34354		NAMIN=	35337					
FEIFST.	34370		NAMIN=	35347					
			FLTIN=	36527	36531	36536	36537	36542	36553
ERRSET	34373								
FEIERR.	34403		NAMIN=	35674	35675	35676	35700		
			FLTIN=	36445	36546				
KOJPT.	34530	KODER=	OUTC=	37552					
KODWRT=	35171		OUTC=	37632					
KOREP.	35176		OUTC=	37553					
NAMIN.	35305	NAMIN=	CONSTNT	7014					
FEOL.	35731	OUTCOM=	KODER=	34545	34610				
			NAMOUT=	37417					
FEOI.	35734		KODER=	34542	34543				
			NAMOUT=	37420					
FEOXFL.	36003		FLTOUT=	33126	33316				
			KODER=	35031	35100				
FEOAFM.	36011		FLTOUT=	33152	33155	33157	33163	33314	
			NAMOUT=	37431					
FEOBLS.	36016		FLTOUT=	33106	33107	33110	33112	33302	33303 33307
				33311					
			KODER=	34762	35107				
FEOCNV.	36031		FLTOUT=	33124					
FEOR1F.	36062		FLTOUT=	33300					
FEOR10.	36066		NAMOUT=	37364	37406	37427	37441	37444	37450
FEONTL.	36073		NAMOUT=	37434					
SYSTEM	36110	SYSTEM	ABTERM	15741					
GOTOER.	36123	GOTOER=	SY	11476					
			FCNMON	12522					
			NATSEK	13125					
			ANOT	22450	23002				
			AXIS	23406	23431				
			CALCMP	24670					
			GRIDGN	31065					
			QUICK	31723					
REMARK	36136	REMARK	CALCMP	24733	25171				
			SCALEB	26755					
			QUICK	32160	32163				
ALOG	36142	ALOG							
ALOG10	36146								
ALOG.	36153		CONSTNT	7032					
			NATSEK	13155	13162				
			GAUS8	16211	16213				
			QNC7	17402	17404				
ALOG10.	36173		ANOT	22417	22635				
			SCALEB	26764					
			GRIDGN	31121					
			NUMDIS	31433					
EXP	36235	EXP							
EXP.	36253		SY	11532					
ITOJ.	36335	ITOJ=	NUMDIS	31460					
COS	36350	SINCOS=							
SIN	36353								
COS.	36356		GSIN	12344					
			SYMBOL	27352					


.32
.28
.24
.20
.16
.12
.08
.04
0
3308
3310
3312
3314
3316
3318
3320
3322
3324
3326
3328
3330

```
G 10

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 4,

$END
 IFUNC    =   4
 GNU      =        .93000000000000E+10
 PLANC    =        .10545887000000E-26
 BETA     =        .92740780000000E-20
 PI       =        .31415926535898E+01
 HALF     =        .10000000000000E+00
 OVRHALF  =        .99999999999999E+01
 HZERO    =        .33195269359482E+04
 HONE     =        .33218502563053E+04
 H2       =        .33213521282053E+04
 H3       =        .33195269359482E+04
 H1S      =        .11034689125316E+08
 H2S      =        .11031379959534E+08
 H3S      =        .11019259078486E+08
 G1       =        .20003000000000E+01
 G2       =        .20006000000000E+01
 G3       =        .20017000000000E+01
 S        =        .44007492998454E-05
 XLOG2    =        .69314718055995E+00
 PILOG    =        .93943727869965E-01
 TONE     =        .25000000000000E-03
 GAMMA    =        .17602997199382E+08
 H1STAR   =        .10000000000000E-04
 HLFSTR   =        .50000000000000E+01
 OHLFSTR  =        .20000000000000E+00
 DELTAH   =        .20000000000000E+00
 DELTAHS  =        .10000000000000E+02
```

```
      PROGRAM TEST(INPUT,OUTPUT,TAPE1=INPUT,TAPE5=)                     TEST      2
      DIMENSION  XPLOT(51),YPLOT(51),HVEC(61),COFVEC(61)                TEST      3
      CALL  PRMODE                                                      TEST      4
      IFUNC = 0                                                         TEST      5
   10 CONTINUE                                                          TEST      6
      IFUNC = IFUNC + 1                                                 TEST      7
      CALL CONSTNT                                                      TEST      8
      IF(EOF(1))  20,11                                                 TEST      9
   11 CONTINUE                                                          TEST     10
      NPTS = 51                                                         TEST     11
      FAC = 2.0                                                         TEST     12
      L=+1                                                              TEST     13
      ISYM = 51                                                         TEST     14
      CALL SIGNAL(XPLOT,YPLOT,NPTS,FAC)                                 TEST     15
      CALL QUICK(L,XPLOT,YPLOT,NPTS,ISYM,1H ,1,1H ,1)                   TEST     16
      HSUBM   =  0.1                                                    TEST     17
      OMEGAM  =  1000.0                                                 TEST     18
      N=1                                                               TEST     19
      IPTS = 61                                                         TEST     20
      IFLAG= 1                                                          TEST     21
      IF(IFUNC .EQ. 3) GO TO 15                                         TEST     22
      IF(IFUNC .EQ. 4) GO TO 15                                         TEST     23
      IF(IFUNC .EQ. 7) GO TO 15                                         TEST     24
      IF(IFUNC .EQ. 8) GO TO 15                                         TEST     25
      FAC = 4.0*FAC                                                     TEST     26
      CALL FOUR(HVEC,COFVEC,IPTS,IFLAG,HSUBM,OMEGAM,N,FAC)              TEST     27
      CALL QUICK(L,HVEC,COFVEC,IPTS,ISYM,1H ,1,1H ,1)                   TEST     28
   15 CONTINUE                                                          TEST     29
      GO TO 10                                                          TEST     30
   20 CONTINUE                                                          TEST     31
      CALL  ENDPLOT                                                     TEST     32
      CALL EXIT                                                         TEST     33
      END                                                               TEST     34
```

SYMBOLIC REFERENCE MAP (R=3)

```
ENTRY POINTS     DEF LINE     REFERENCES
 6152  TEST          1

VARIABLES      SN  TYPE          RELOCATION
 6550  COFVEC     REAL       ARRAY                REFS        2        26        27
 6275  FAC        REAL                            REFS       14        25        26    DEFINED       11        25
 6300  HSUBM      REAL                            REFS       26   DEFINED        16
 6453  HVEC       REAL       ARRAY                REFS        2        26        27
 6304  IFLAG      INTEGER                         REFS       26   DEFINED        20
 6273  IFUNC      INTEGER                         REFS        6        21        22        23        24
                                                DEFINED        4         6
 6303  IPTS       INTEGER                         REFS       26        27   DEFINED        19
 6277  ISYM       INTEGER                         REFS       15        27   DEFINED        13
 6276  L          INTEGER                         REFS       15        27   DEFINED        12
 6302  N          INTEGER                         REFS       26   DEFINED        18
 6274  NPTS       INTEGER                         REFS       14        15   DEFINED        10
 6301  OMEGAM     REAL                            REFS       26   DEFINED        17
```

```
      SUBROUTINE  SIGNAL(XPLOT,YPLOT,NPTS,FAC)                          SIGNAL      2
C                                                                       SIGNAL      3
C       LET I(H)=INTEGRAL(0,INFINITY)(SY(HPRIM)) WHERE SY IS GIVEN      SIGNAL      4
C     BY SUBROUTINE SY ,THE VALUE OF WHICH IS DETERMINED BY IFUNC       SIGNAL      5
C     ACCORDING TO THE RULE                                             SIGNAL      6
C                                                                       SIGNAL      7
C     IFUNC  =  1   GAUSSIAN SHAPE FUNCTION        -   ABSORPTION CASE  IFUNC       2
C            =  2   GAUSSIAN SHAPE FUNCTION        -   DISPERSION CASE  IFUNC       3
C            =  3   GAUSSIAN (DERIVATIVE)          -   ABSORPTION CASE  IFUNC       4
C            =  4   GAUSSIAN (DERIVATIVE)          -   DISPERSION CASE  IFUNC       5
C            =  5   POWDER LINESHAPE               -   ABSORPTION CASE  IFUNC       6
C            =  6   POWDER LINESHAPE               -   DISPERSION CASE  IFUNC       7
C            =  7   POWDER LINESHAPE (DERIVATIVE)  -   ABSORPTION CASE  IFUNC       8
C            =  8   POWDER LINESHAPE (DERIVATIVE)  -   DISPERSION CASE  IFUNC       9
C                                                                       IFUNC      10
C                                                                       SIGNAL      9
C                                                                       SIGNAL     10
C      SUBROUTINE SIGNAL RETURNS THE VECTORS XPLOT AND YPLOT OF LENGTH  SIGNAL     11
C     NPTS WHERE                                                        SIGNAL     12
C                                                                       SIGNAL     13
C        XPLOT(I) = HSTART +(I-1)*(HFINAL-HSTART)/(NPTS-1), I=1,...,NPTS SIGNAL     14
C     AND                                                               SIGNAL     15
C        YPLOT(I) = I(XPLOT(I)).                                        SIGNAL     16
C                                                                       SIGNAL     17
C       IF IFUNC .LE. 4 (GAUSSIAN CASE),                                SIGNAL     18
C           HSTART=  HZERO  -  FAC*DELTAHS/2.0                          SIGNAL     19
C     AND                                                               SIGNAL     20
C           HFINAL=  HZERO  +  FAC*DELTAHS/2.0                          SIGNAL     21
C     WHERE                                                             SIGNAL     22
C           HZERO =   PLANC*GNU/(BETA*G3)                               SIGNAL     23
C                                                                       SIGNAL     24
C                                                                       SIGNAL     25
C        IF IFUNC  .GE. 5  AND  .LE.  8 ,                               SIGNAL     26
C            HSTART =   H3 - FAC*(HONE - H3)                            SIGNAL     27
C     AND                                                               SIGNAL     28
C            HFINAL = HONE + FAC*(HONE - H3).                           SIGNAL     29
C                                                                       SIGNAL     30
C                                                                       SIGNAL     31
C     *****                                                             SIGNAL     32
C     NOTE-                                                             SIGNAL     33
C         SUBROUTINE CONSTANT MUST HAVE BEEN CALLED BEFORE A CALL TO    SIGNAL     34
C         SIGNAL (TO SET THE VALUES IN COMMON BLOCK VITK01)             SIGNAL     35
C     *****                                                             SIGNAL     36
C                                                                       SIGNAL     37
C                                                                       SIGNAL     38
      DIMENSION   XPLOT(1),YPLOT(1)                                     SIGNAL     39
C                                                                       VITK01      2
      COMMON /VITK01/ GNU,    PLANC,   BETA,    PI,    HALF,  OVRHALF,  VITK01      3
     *                                                                  VITK01      4
     * HZERO, HONE,   H2 ,  H3 ,   H1S,   H2S,  H3S,                    VITK01      5
     *                                                                  VITK01      6
     * G1,  G2,  G3,  S,   IFUNC,    XLOG2,   PILOG,    TONE ,  GAMMA  VITK01      7
     *                                                                  VITK01      8
     * ,HISTAR, HLFSTR, OVLFSTR, DELTAH, DELTAHS                        VITK01M1    1
C                                                                       VITK01     12
C                                                                       VITK01     13
      IF(IFUNC .GE. 5) GO TO 10                                         SIGNAL     41
```

```
VARIABLES      SN  TYPE            RELOCATION
     0  HVEC       REAL       ARRAY     F.P.        REFS        77    DEFINED        1        170
     6  HZERO      REAL                 VITK01      REFS        80       101        102
    12  H1S        REAL                 VITK01      REFS        80
    10  H2         REAL                 VITK01      REFS        80
    13  H2S        REAL                 VITK01      REFS        80
    11  H3         REAL                 VITK01      REFS        80       105        106
    14  H3S        REAL                 VITK01      REFS        80
   245  I        * INTEGER                          DEFINED      93
   273  IERR       INTEGER                            REFS       157       159        161
   265  IERRH    * INTEGER                            REFS       135
     0  IFLAG      INTEGER               F.P.        REFS       157       159        162    DEFINED        1
    21  IFUNC      INTEGER              VITK01       REFS        80      4*91         99
     0  IPTS       INTEGER               F.P.        REFS       109       154    DEFINED        1
   260  ISX        INTEGER                            REFS       135    DEFINED      130
   257  J          INTEGER                            REFS       125       126        170        171    DEFINED      123       154
     0  N          INTEGER               F.P.        REFS       148       162    DEFINED        1
     2  NN         INTEGER              VITK04       REFS        75    DEFINED      148
  1303  NP         INTEGER              VITK05       REFS        78       121        123        135    DEFINED      120
    30  OHLFSTR    REAL                 VITK01       REFS        80
     0  OMEGAM     REAL                  F.P.        REFS       147       149    DEFINED        1
     5  OVRHALF    REAL                 VITK01       REFS        80
     3  PI         REAL                 VITK01       REFS        80       149    DEFINED      144
    23  PILOG      REAL                 VITK01       REFS        80
     1  PLANC      REAL                 VITK01       REFS        80
    20  S          REAL                 VITK01       REFS        80
    24  TONE       REAL                 VITK01       REFS        80
   624  WH         REAL       ARRAY     VITK05       REFS        78       135
     0  XH         REAL       ARRAY     VITK05       REFS        78       135    DEFINED      125
    22  XLOG2      REAL                 VITK01       REFS        80
   145  YH         REAL       ARRAY     VITK05       REFS        78       135    DEFINED      126
   312  YP         REAL       ARRAY     VITK05       REFS        78       135
   457  YPP        REAL       ARRAY     VITK05       REFS        78       135

FILE NAMES        MODE
       OUTPUT     FMT                    WRITES      162
       TAPE50     FMT                    WRITES       94

EXTERNALS          TYPE    ARGS        REFERENCES
       CONVOL     REAL       1             126
       ERXSET                2             153
       EXIT                  0              97
       GCOS                  0              72        159
       GSIN                  0              72        157
       QNC7                  6             157        159
       SPLIFT               12             135

STATEMENT LABELS              DEF LINE    REFERENCES
   225  5        FMT             163         162
   210  7        FMT              95          94
     0  8                        96          93
    33  9                        98          91
    43  10                      104          99
    50  12                      108         103
   135  15                      167         161
     0  20                      172         154
     0  110                     127         123
```

```
      HLOLO  = HLO  -XFAC*DIF                                           CONVOL     37
      HLOLO  = AMAX1(0.0,HLOLO)                                         CONVOL     38
      HUP =  HONE + FAC*DIF                                             CONVOL     39
      HUPUP  =HUP  + XFAC*DIF                                           CONVOL     40
      HUPUP  = AMIN1(1.0E20,HUPUP)                                      CONVOL     41
      GO TO 150                                                         CONVOL     42
  110 CONTINUE                                                          CONVOL     43
      FAC = 3.0                                                         CONVOL     44
      XFAC= 10.0                                                        CONVOL     45
      DIF = HLFSTR                                                      CONVOL     46
      HLO = HZERO - FAC*DIF                                             CONVOL     47
      HUP = HZERO + FAC*DIF                                             CONVOL     48
      HLOLO = HLO - XFAC*DIF                                            CONVOL     49
      HUPUP = HUP + XFAC*DIF                                            CONVOL     50
  150 CONTINUE                                                          CONVOL     51
      ICOUNT = 0                                                        CONVOL     52
    1 CONTINUE                                                          CONVOL     53
      CALL GAUS8(SY,HLOLO,HLO,ERR,QLEFT,IERR)                           CONVOL     54
      IF(IERR .EQ.  1) GO TO  4                                         CONVOL     55
    2 PRINT 3, IERR                                                     CONVOL     56
    3 FORMAT(*  TROUBLE IN GAUS8 -- CALLED FROM CONVOL -- IERR =*,I3)   CONVOL     57
      PRINT 10, IFUNC                                                   CONVOL     58
   10 FORMAT(* IFUNC = *,I3)                                            CONVOL     59
      CALL EXIT                                                         CONVOL     60
    4 CONTINUE                                                          CONVOL     61
      CALL GAUS8(SY,HLO,HUP,ERR,QMIDDLE,IERR)                           CONVOL     62
      IF(IERR .NE. 1) GO TO  2                                          CONVOL     63
      CALL GAUS8(SY,HUP,HUPUP,ERR,QRIGHT,IERR)                          CONVOL     64
      IF(IERR .NE. 1) GO TO 2                                           CONVOL     65
      Q = ABS( QLEFT + QRIGHT)                                          CONVOL     66
      TESTFAC = .01                                                     CONVOL     67
      TEST = TESTFAC*ABS(QMIDDLE)                                       CONVOL     68
      IF( Q .LT. TEST) GO TO 7                                          CONVOL     69
      ICOUNT = ICOUNT + 1                                               CONVOL     70
      IF(ICOUNT .EQ. 3 )  GO TO  5                                      CONVOL     71
      HLO = HLO -DIF                                                    CONVOL     72
      HUP = HUP + DIF                                                   CONVOL     73
      HLOLO = HLO - XFAC*DIF                                            CONVOL     74
      HUPUP = HUP + XFAC*DIF                                            CONVOL     75
      GO TO 1                                                           CONVOL     76
    5 CONTINUE                                                          CONVOL     77
      CALL CNVLERR(HH)                                                  CONVOL     78
    7 CONTINUE                                                          CONVOL     79
      CONVOL  =   QMIDDLE  + QRIGHT + QLEFT                             CONVOL     80
      RETURN                                                            CONVOL     81
      END                                                               CONVOL     32
```

SYMBOLIC REFERENCE MAP (R=3)

ENTRY POINTS	DEF LINE	REFERENCES
4 CONVOL	1	102

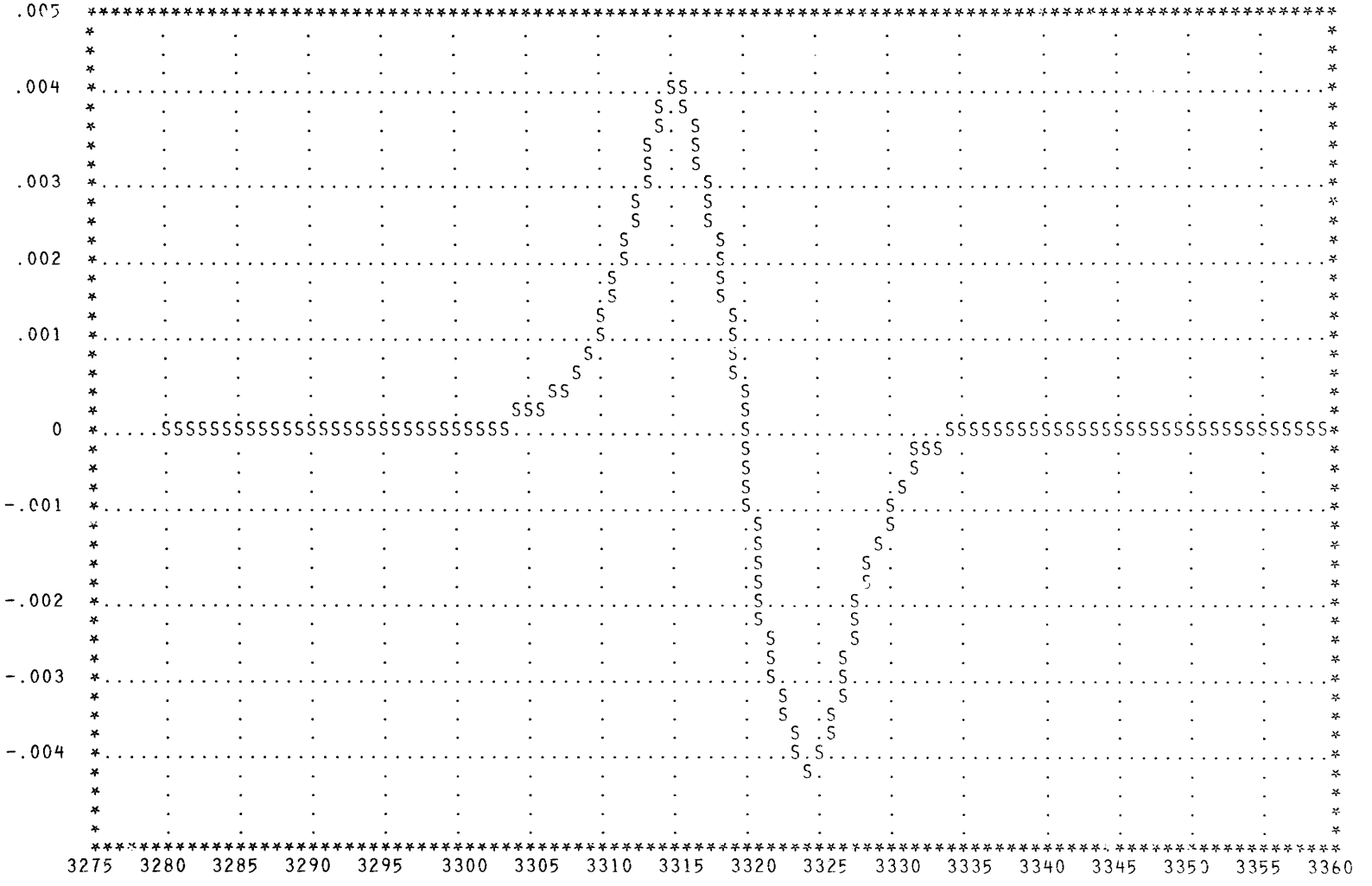
```
      FUNCTION GSIN(X)                                                  GSIN      2
C       FUNCTION GSIN CALCULATES                                        GSIN      3
C            GSIN(X) = I(H + HSUBM*SIN(OMEGAM*X))*SIN(N*OMEGAM*X)       GSIN      4
C           WHERE N IS PASSED TO GSIN FROM FOUR (TO FOURIER AND THEN    GSIN      5
C           THROUGH COMMON TO GSIN).                                    GSIN      6
C                                                                       GSIN      7
C       GCOS IS AN ALTERNATE ENTRY POINT TO GSIN WHICH CALCULATES       GSIN      8
C            GCOS(X) = I(H + HSUBM*SIN(OMEGAM*X))*COS(N*OMEGAM*X)       GSIN      9
C                                                                       GSIN     10
C                                                                       GSIN     11
C                                                                       GSIN     12
C          THE CALCULATION OF THE FOURIER COEFFICIENTS REQUIRES MANY    GSIN     13
C     EVALUATIONS OF I(H). FOR EFFICIENCY, I(H) IS FIT WITH A CUBIC     GSIN     14
C     INTERPOLATORY SPLINE AT 101 POINTS AND THE FIT IS USED FOR        GSIN     15
C     SUBSEQUENT EVALUATIONS. TO CAUSE THE FIT TO HAVE OTHER THAN       GSIN     16
C     101 POINTS  ONE MUST CHANGE THE COMMON STATMENT  VITK05  (IN      GSIN     17
C     BOTH FOUR AND  GSIN   ) ACCORDING TO THE FOLLOWING RULE --        GSIN     18
C        NP =  NUMBER OF FITTING POINTS                                 GSIN     19
C     COMMON /VITK05/  XH(NP),YH(NP),YP(NP),YPP(NP),WH(3*NP),NP         GSIN     20
C        AND THE VALUE OF NP MUST BE SET EITHER DIRECTLY IN FOUR OR IN  GSIN     21
C        THE USER CALLING PROGRAM THROUGH COMMON BLOCK VITK05           3JAN(76)  2
C                                                                       GSIN     23
C      USEFUL LIMITS FOR NP PROBABLY LIE BETWEEN 51 AND 501.            GSIN     24
C                                                                       GSIN     25
C                                                                       GSIN     26
C                                                                       GSIN     27
C                                                                       GSIN     28
C                                                                       VITK02    2
      COMMON /VITK02/ H                                                 VITK02    3
C                                                                       VITK02    4
C                                                                       VITK02    5
      COMMON /VITK04/ CHSUBM,COMEGAM,NN                                 VITK04    2
      COMMON /VITK05/  XH(101),YH(101),YP(101),YPP(101),WH(303),NP      VITK05    2
      XN = FLOAT(NN)                                                    GSIN     32
      S = SIN(XN*COMEGAM*X)                                             GSIN     33
      GO TO 10                                                          GSIN     34
      ENTRY GCOS                                                        GSIN     35
      XN = FLOAT(NN)                                                    GSIN     36
      S = COS(XN*COMEGAM*X)                                             GSIN     37
   10 CONTINUE                                                          GSIN     38
      ARG = H + CHSUBM*SIN(COMEGAM*X)                                   GSIN     39
      NI = 1                                                            GSIN     40
      CALL SPLINT(XH,YH,YPP,NP,ARG,G,GP,GPP,NI,KERR)                    GSIN     41
      IF(KERR .EQ  1) GO TO 20                                          GSIN     42
      PRINT 15, ARG                                                     GSIN     43
   15 FORMAT(///,* IN SPLINT, ARG =*,E25.14,* IS OUT OF BOUNDS *,//)    GSIN     44
   20 CONTINUE                                                          GSIN     45
      GSIN = S*G                                                        GSIN     46
      RETURN                                                            GSIN     47
      END                                                               GSIN     48
```

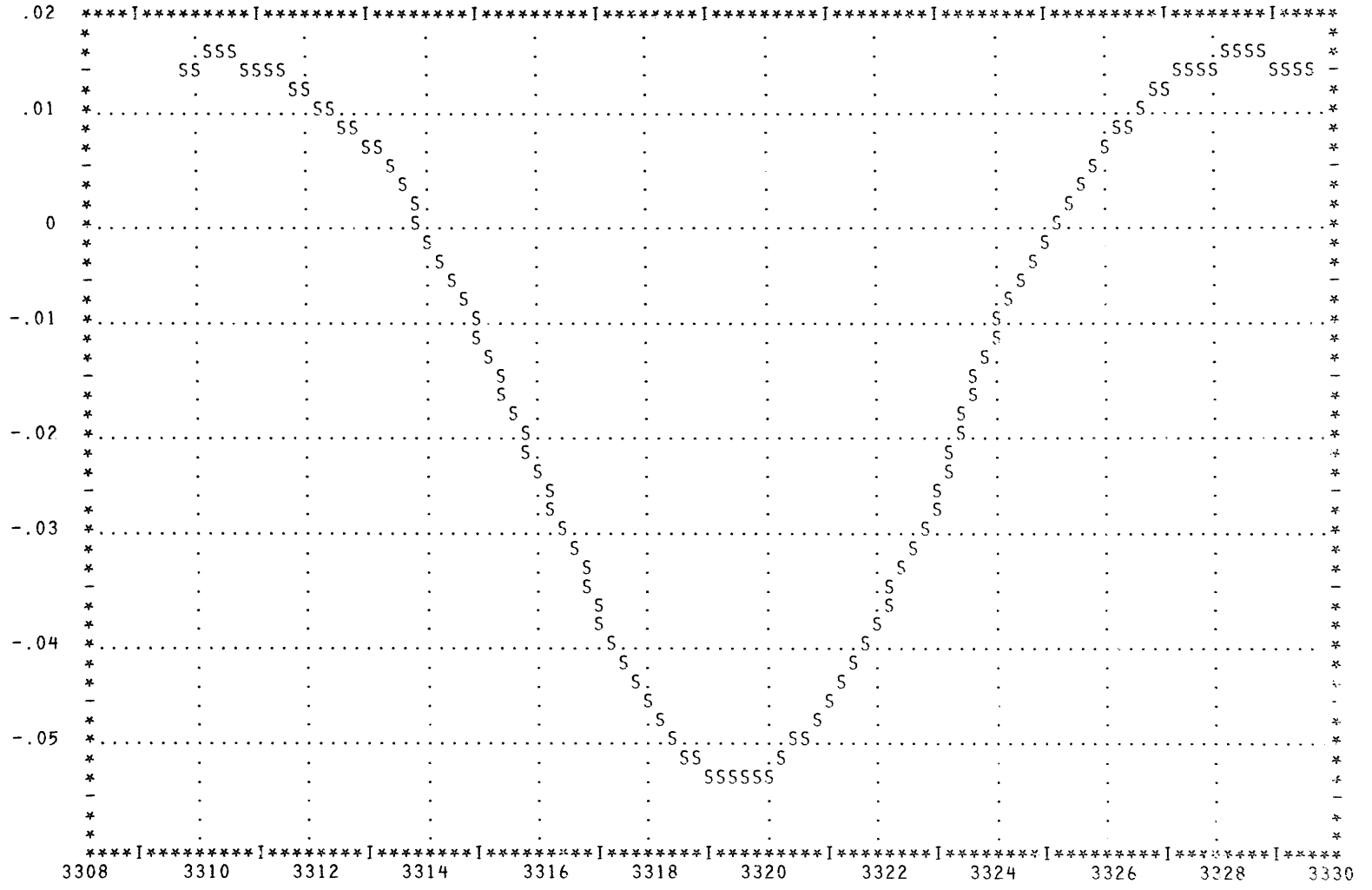
SYMBOLIC REFERENCE MAP (R=3)

```
C        PREVIOUS TABLE) FOR THE LATEST ERROR ENCOUNTERED.                NATSEK    116
C        AT THE SAME TIME, THE INTERNAL ERROR INDICATOR IS                NATSEK    117
C        RESET TO 0.  A RETURN OF 0 INDICATES NO ERROR HAS                NATSEK    118
C        OCCURRED.                                                        NATSEK    119
C                                                                         NATSEK    120
C        ----------                                                       NATSEK    121
C                                                                         NATSEK    122
C     QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW           NATSEK    123
C     APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY           NATSEK    124
C                                                                         NATSEK    125
C     LATEST REVISION  JUNE, 1973                                         NATSEK    126
C                                                                         NATSEK    127
C     ------------------------------------------------------------------  NATSEK    128
      DATA XIND/17770000000000000000B/,IFCN/2/,FCN/5HELIPK,5HELIK1,        NATSEK    129
     1     5HELIKM/,EPS/16354000000000000000B/                            NATSEK    130
C     ----------                                                          NATSEK    131
C                                                                         NATSEK    132
C     COEFFICIENTS FOR THE A(9) + B(9) * LN(1/ETA) APPROXIMATION.         NATSEK    133
C                                                                         NATSEK    134
C     ----------                                                          NATSEK    135
      DATA  A/17044732524712154654B,17104040476227526235B,                NATSEK    136
     1171154160652412703 60B,17115330057407374272B,17107546246001627667B, NATSEK    137
     2171114444042136275043B,17117514755150020163B,17127720134677627312B, NATSEK    138
     3171461344137700032 73B,17205427102775750717B/                       NATSEK    139
      DATA  B/17024273624624722514B,17067032371557771231B,                NATSEK    140
     1171114601000205012375B,17125227722401261265B,17127433071605232165B, NATSEK    141
     2171134616645037724301B,17136177737350265012B,17144377777730776700B, NATSEK    142
     3171477777777777765477B,17174000000000000000B/                       NATSEK    143
C                                                                         NATSEK    144
C     ------------------------------------------------------------------  NATSEK    145
C                                                                         NATSEK    146
      CAYSQ = ARG                                                         NATSEK    147
      GO TO (20,120,130), INT                                             NATSEK    148
   20 IF (CAYSQ .LT. 0.0E0) GO TO 200                                     NATSEK    149
      ETA = (0.5E0 - CAYSQ) + 0.5E0                                       NATSEK    150
   50 IF (ETA .LE. 0.0E0) GO TO 220                                       NATSEK    151
      IF (ETA .LT. EPS) GO TO 105                                         NATSEK    152
      SUMA = A(1)                                                         NATSEK    153
      SUMB = B(1)                                                         NATSEK    154
C                                                                         NATSEK    155
      DO 100 I = 2, 10                                                    NATSEK    156
         SUMA = SUMA * ETA + A(I)                                         NATSEK    157
         SUMB = SUMB * ETA + B(I)                                         NATSEK    158
  100 CONTINUE                                                            NATSEK    159
C                                                                         NATSEK    160
      DELIPK = SUMA - ALOG(ETA) * SUMB                                    NATSEK    161
      RESULT = DELIPK                                                     NATSEK    162
      GO TO 300                                                           NATSEK    163
C     ---------- RETURN FOR SMALL ETA ----------                          NATSEK    164
  105 RESULT = A(10) - ALOG(ETA) * B(10)                                  NATSEK    165
      GO TO 300                                                           NATSEK    166
C     ---------- ENTRY FOR ELIK1.  CALCULATION OF                         NATSEK    167
C                1 - CAY**2 IS REFORMULATED                               NATSEK    168
C                      TO PRESERVE ACCURACY ----------                    NATSEK    169
  120 CAY = ABS(CAYSQ)                                                    NATSEK    170
      ETA = (0.5E0 - CAY) + 0.5E0                                         NATSEK    171
      ETA = ETA + CAY * ETA                                               NATSEK    172
```

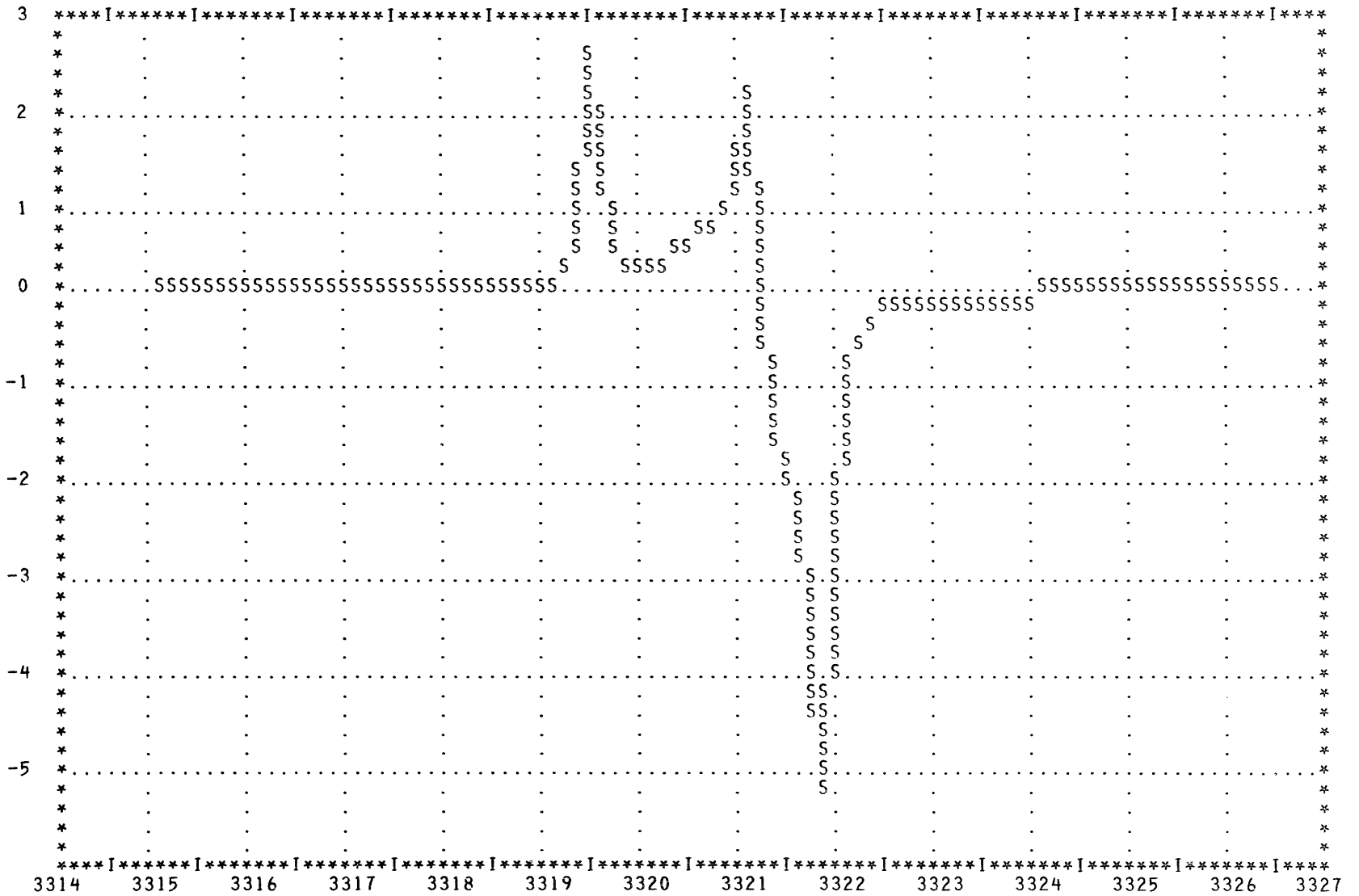
BLOCK	ADDRESS	LENGTH	FILE	PREFIX TABLE CONTENTS
/GRD/	30431	4		
/CLASS/	30435	5		
PLTDATA	30442	2	SL-FTNLIBY	01/19/76 08.18.59. SCOPE 3.4 COMPASS 3.75203 2
MASKER	30444	12	SL-FTNLIBY	05/31/75 11.54.58. SCOPE 3.4 COMPASS 3.75050 2
MINMAX	30456	100	SL-FTNLIBY	05/31/75 11.54.36. SCOPE 3.4 FTN 4.3P393 666X I OPT=2
NARG	30556	5	SL-FTNLIBY	05/31/75 11.55.03. SCOPE 3.4 COMPASS 3.75050 2
NDEC	30563	40	SL-FTNLIBY	05/31/75 11.54.36. SCOPE 3.4 FTN 4.3P393 666X I OPT=2
NHOL	30623	20	SL-FTNLIBY	05/31/75 11.55.56. SCOPE 3.4 COMPASS 3.75050 2
ENDPLOT	30643	53	SL-FTNLIBY	05/31/75 11.56.09. SCOPE 3.4 COMPASS 3.75050 2
FRMADV	30716	16	SL-FTNLIBY	05/31/75 11.56.12. SCOPE 3.4 COMPASS 3.75050 2
GRIDGN	30734	434	SL-FTNLIBY	05/31/75 11.54.36. SCOPE 3.4 FTN 4.3P393 666X I OPT=2
NUMDIS	31370	210	SL-FTNLIBY	05/31/75 11.54.36. SCOPE 3.4 FTN 4.3P393 666X I OPT=2
PRMODE	31600	17	SL-FTNLIBY	05/31/75 11.56.25. SCOPE 3.4 COMPASS 3.75050 2
/NOPLT/	31617	1		
QUICK	31620	517	SL-FTNLIBY	05/31/75 11.54.36. SCOPE 3.4 FTN 4.3P393 666X I OPT=2
RDPLID	32337	116	SL-FTNLIBY	05/31/75 11.56.36. SCOPE 3.4 COMPASS 3.75050 2
/XY/	32455	2		
TIC	32457	43	SL-FTNLIBY	01/19/76 08.18.59. SCOPE 3.4 COMPASS 3.75203 2
/FCL.C./	32522	24		
/Q8.IO./	32546	157		
Q8NTRY=	32725	0	SL-FORTRAN	10/24/75 11.01.57. SCOPE 3.4 COMPASS 3.75203 FCL INITILALIZATION ROUTINE.
/IOCON./	32725	41		
COMIO=	32766	77	SL-FORTRAN	10/24/75 11.02.31. SCOPE 3.4 COMPASS 3.75203 2 COMMON CODED I/O ROUTINES AND CONSTANTS.63-CHAR
EOF	33065	16	SL-FORTRAN	10/24/75 11.02.46. SCOPE 3.4 COMPASS 3.75203 2 TEST FOR END OF FILE STATUS.
FLTOUT=	33103	310	SL-FORTRAN	10/24/75 11.02.49. SCOPE 3.4 COMPASS 3.75203 2 COMMON FLOATING OUTPUT CODE
FORSYS=	33413	615	SL-FORTRAN	10/24/75 11.02.57. SCOPE 3.4 COMPASS 3.75203 2 FORTRAN OBJECT LIBRARY UTILITIES.
INCOM=	34230	277	SL-FORTRAN	10/24/75 11.03.34. SCOPE 3.4 COMPASS 3.75203 2 COMMON INPUT FORMATTING CODE
KODER=	34527	457	SL-FORTRAN	10/24/75 11.04.45. SCOPE 3.4 COMPASS 3.75203 2 OUTPUT FORMAT INTERPRETER.
NAMIN=	35206	523	SL-FORTRAN	10/24/75 11.05.33. SCOPE 3.4 COMPASS 3.75203 2 NAMELIST INPUT ROUTINE.
OUTCOM=	35731	154	SL-FORTRAN	10/24/75 11.06.38. SCOPE 3.4 COMPASS 3.75203 2 COMMON OUTPUT CODE
SYSTEM	36105	15	SL-FORTRAN	10/24/75 11.07.19. SCOPE 3.4 COMPASS 3.75203 2 USER CALLABLE ERROR PROCESSOR.
GOTOER=	36122	14	SL-FORTRAN	10/24/75 11.07.38. SCOPE 3.4 COMPASS 3.75203 2 COMPUTED GO TO ERROR PROCESSOR.
REMARK	36136	3	SL-FORTRAN	10/24/75 11.07.45. SCOPE 3.4 COMPASS 3.75203 2 ISSUE A MESSAGE TO THE DAYFILE.
ALOG	36141	73	SL-FORTRAN	10/24/75 11.07.51. SCOPE 3.4 COMPASS 3.75203 2 COMPUTE COMMON AND NATURAL LOGARITHMS. OPT=ALL.
EXP	36234	75	SL-FORTRAN	10/24/75 11.09.38. SCOPE 3.4 COMPASS 3.75203 2 EXPONENTIAL FUNCTION. E TO POWER X. OPT=ALL.
ITOJ=	36331	16	SL-FORTRAN	10/24/75 11.09.44. SCOPE 3.4 COMPASS 3.75203 2

ENTRY	ADDRESS	PROGRAM	REFERENCES							
SIN.	36366		GSIN	12330	12347					
			SYMBOL	27354						
SYSAID=	36435	SYSAID=	Q8.IO.	32612						
			SYS=1ST	40016						
FEIEXP.	36436	FLTIN=	INCOM=	34333	34352					
FEIFSC.	36460		INCOM=	34362						
FECNAP.	36622	FMTAP=	KODER=	34612	34667	34673	34713	34730	35170	35177
FECAP.	36630		OUTC=	37643						
FECFMT.	36647		KODER=	34607	35133	35135	35145	35175		
FECFMU.	36652		KODER=	34546	34652					
FECJP=	36755		OUTC=	37636						
FECLP.	36756		KODER=	34601						
FECRP.	37001		KODER=	34602						
FECEE.	37016		KODER=	34733						
FECV.	37064		KODER=	34556						
FECBUG.	37073		KODER=	34527	34530					
CBD.	37166	FORUTL=	FORSYS=	33450	33572	33751	33772	34043		
			INCOM=	34421						
BFN.	37173		FORSYS=	33564	33744					
			NAMOUT=	37331	37341					
GETFIT.	37204	GETFIT=	EOF	33072						
			NAMIN=	35312						
			NAMOUT=	37255						
			OUTC=	37562						
NAME.	37241									
NAMOUT.	37251	NAMOUT=	CONSTNT	7016						
OUTCI.	37554	OUTC=	CONSTNT	7056	7060	7062	7064	7066	7070	7072
				7074	7076	7100	7102	7104	7106	7110
				7112	7114	7116	7120	7122	7124	7126
				7130	7132	7134	7136	7140	7142	
			FOUR	11167	11275					
			CONVOL	11711	11713					
			CNVLERR	12116	12144	12160				
			GSIN	12360						
			FCNMON	12450	12457	12475				
			ERRCHK	15437	15462	15474	15511	15516	15521	
			ABTERM	15736						
			CURSES	16031						
			ONECHK	21420	21427					
			PCVOUT	26575						
OUTCR.	37652		FCNMON	12504	12510					
			ERRCHK	15444	15450					
SQRT	37735	SQRT								
SQRT.	37754		CONSTNT	7034						
			SY	11570	11602					
			DSHPLOT	26032	26113					
			SYMBOL	27360	27370					
			XYPLOT	30021						
SYS1ST.	40002	SYS=1ST	ALOG	36207						
			EXP	36263						
			SINCOS=	36414						
			SQRT	37767						
MORGUE.	40004									
XTOI.	40064	XTOI=	GAUS8	16230						
			QNC7	17421						


.005
.004
.003
.002
.001
0
-.001
-.002
-.003
-.004
3275 3280 3285 3290 3295 3300 3305 3310 3315 3320 3325 3330 3335 3340 3345 3350 3355 3360

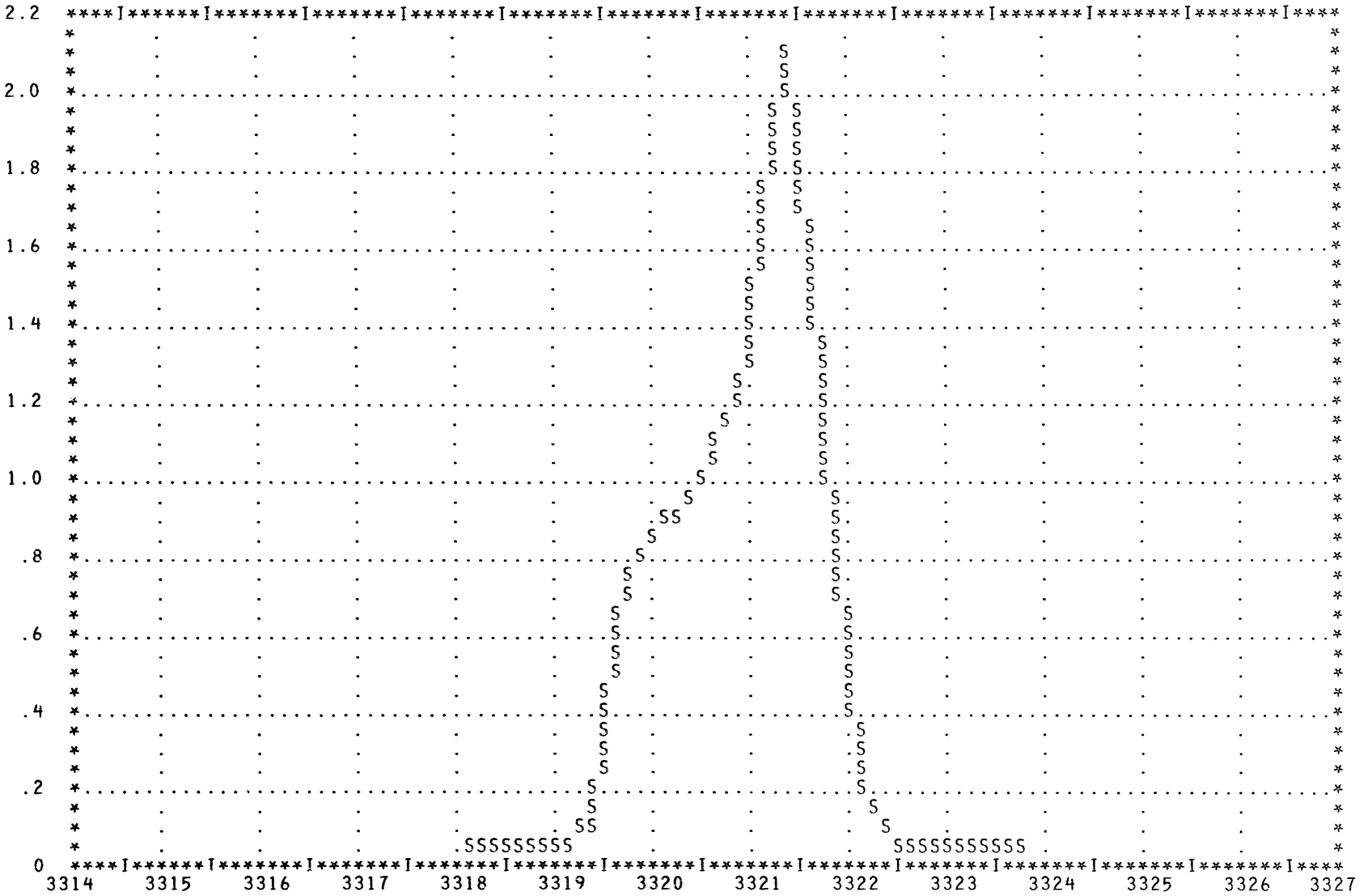

.02
.01
0
-.01
-.02
-.03
-.04
-.05
3308
3310
3312
3314
3316
3318
3320
3322
3324
3326
3328
3330

```
B  1

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 5,

$END
 IFUNC   =   5
 GNU     =        .93000000000000E+10
 PLANC   =        .10545887000000E-26
 BETA    =        .92740780000000E-20
 PI      =        .31415926535898E+01
 HALF    =        .10000000000000E+00
 OVRHALF =        .99999999999999E+01
 HZERO   =        .33195269359482E+04
 HONE    =        .33218502563053E+04
 H2      =        .33213521282053E+04
 H3      =        .33195269359482E+04
 H1S     =        .11034689125316E+08
 H2S     =        .11031379959534E+08
 H3S     =        .11019259078486E+08
 G1      =        .20003000000000E+01
 G2      =        .20006000000000E+01
 G3      =        .20017000000000E+01
 S       =        .44007492998454E-05
 XLOG2   =        .69314718055995E+00
 PILOG   =        .93943727869965E-01
 TONE    =        .25000000000000E-03
 GAMMA   =        .17602997199382E+08
 H1STAR  =        .10000000000000E-04
 HLFSTR  =        .50000000000000E+01
 OHLFSTR =        .20000000000000E+00
 DELTAH  =        .20000000000000E+00
 DELTAHS =        .10000000000000E+02
```


3
2
1
0
-1
-2
-3
-4
-5
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327

PLOT 11

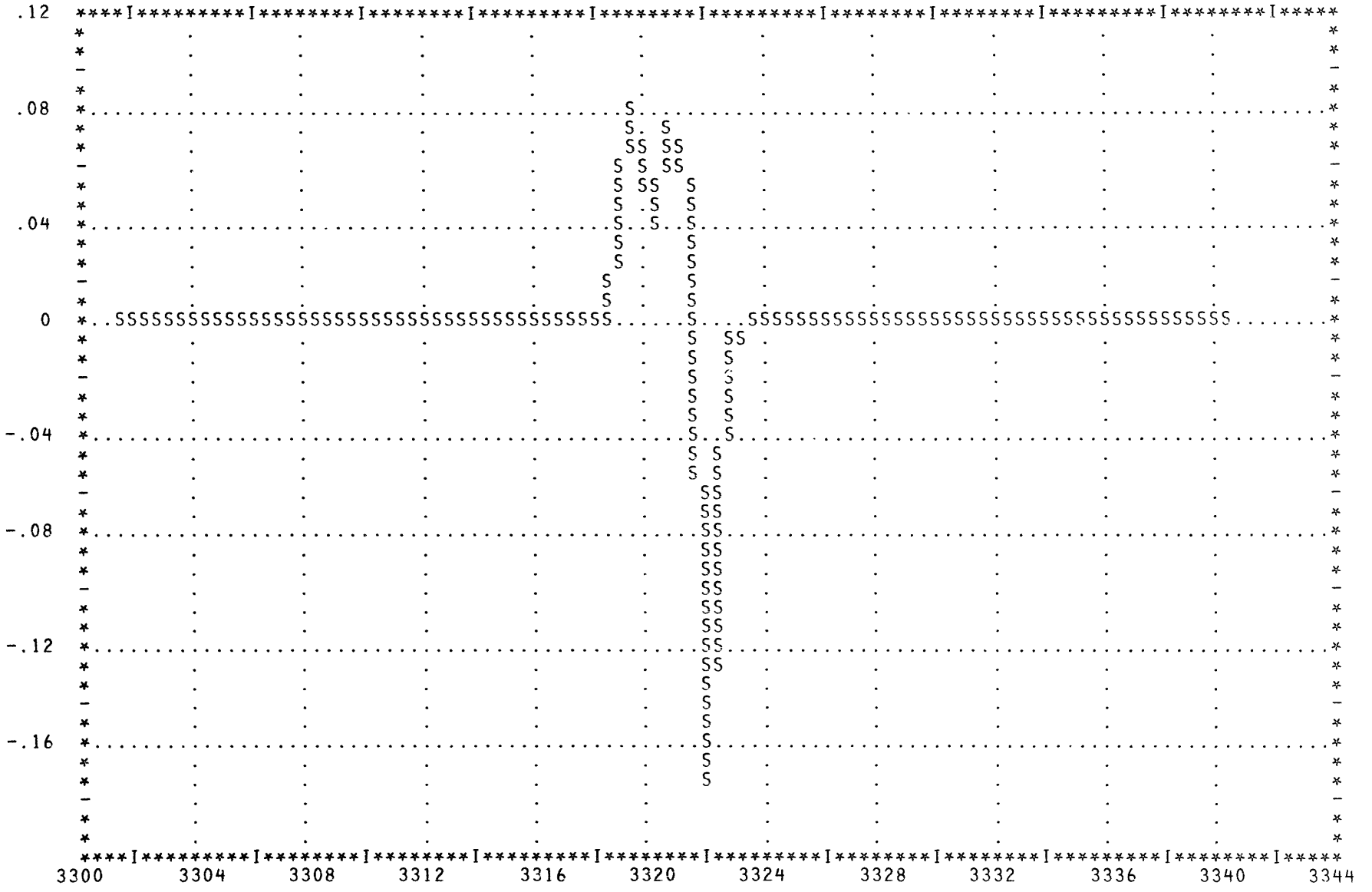
C 1

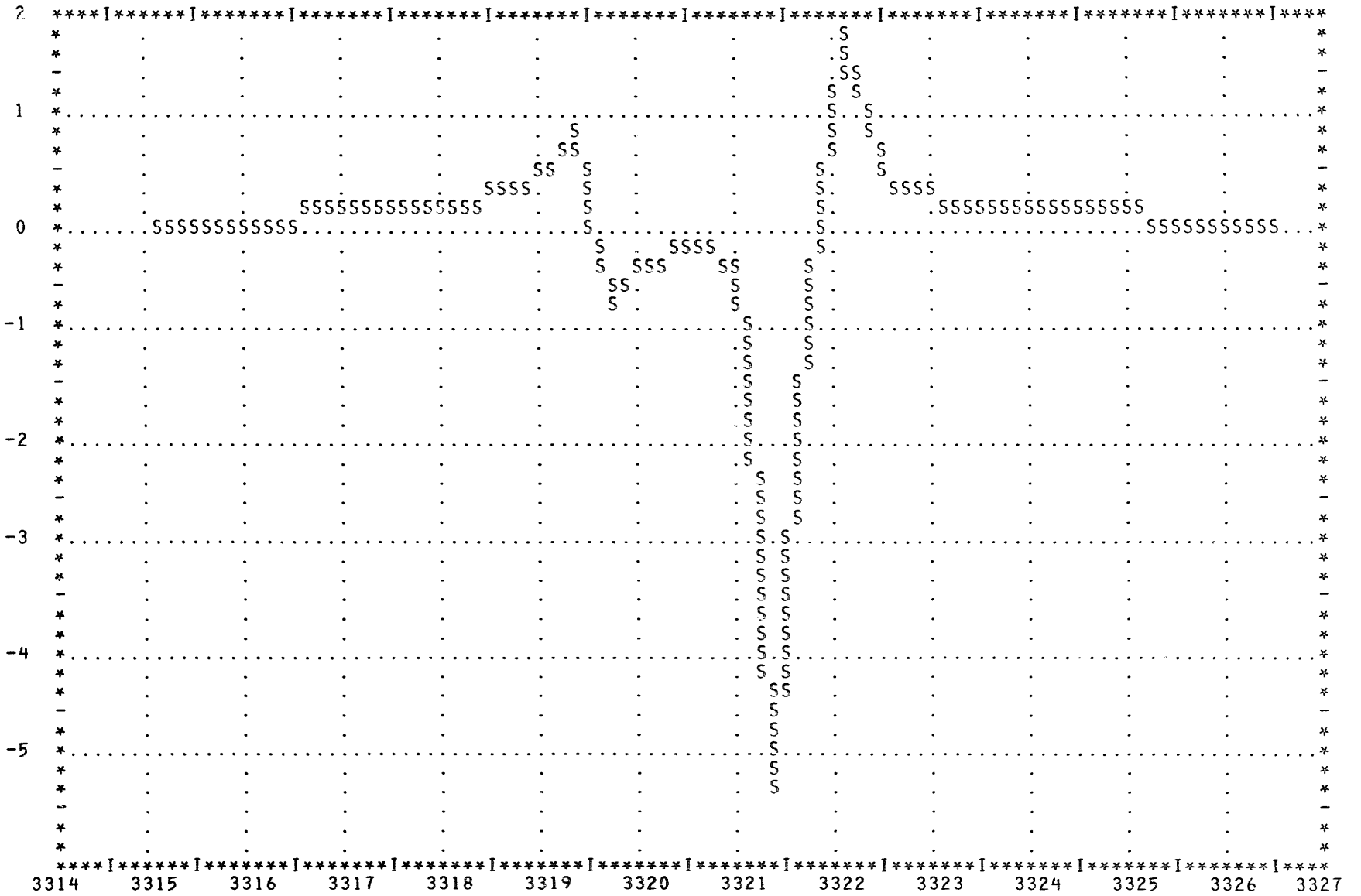

2.2
2.0
1.8
1.6
1.4
1.2
1.0
.8
.6
.4
.2
0
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327

PLOT 7

```
C  2

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 8,

$END
 IFUNC   =   8
 GNU     =        .93000000000000E+10
 PLANC   =        .10545887000000E-26
 BETA    =        .92740780000000E-20
 PI      =        .31415926535898E+01
 HALF    =        .10000000000000E+00
 OVRHALF =        .99999999999999E+01
 HZERO   =        .33195269359482E+04
 HONE    =        .33218502563053E+04
 H2      =        .33213521282053E+04
 H3      =        .33195269359482E+04
 H1S     =        .11034689125316E+08
 H2S     =        .11031379959534E+08
 H3S     =        .11019259078486E+08
 G1      =        .20003000000000E+01
 G2      =        .20006000000000E+01
 G3      =        .20017000000000E+01
 S       =        .44007492998454E-05
 XLOG2   =        .69314718055995E+00
 PILOG   =        .39943727869965E-01
 TONE    =        .25000000000000E-03
 GAMMA   =        .17602997199382E+08
 H1STAR  =        .10000000000000E-04
 HLFSTR  =        .50000000000000E+01
 OHLFSTR =        .20000000000000E+00
 DELTAH  =        .20000000000000E+00
 DELTAHS =        .10000000000000E+02
```

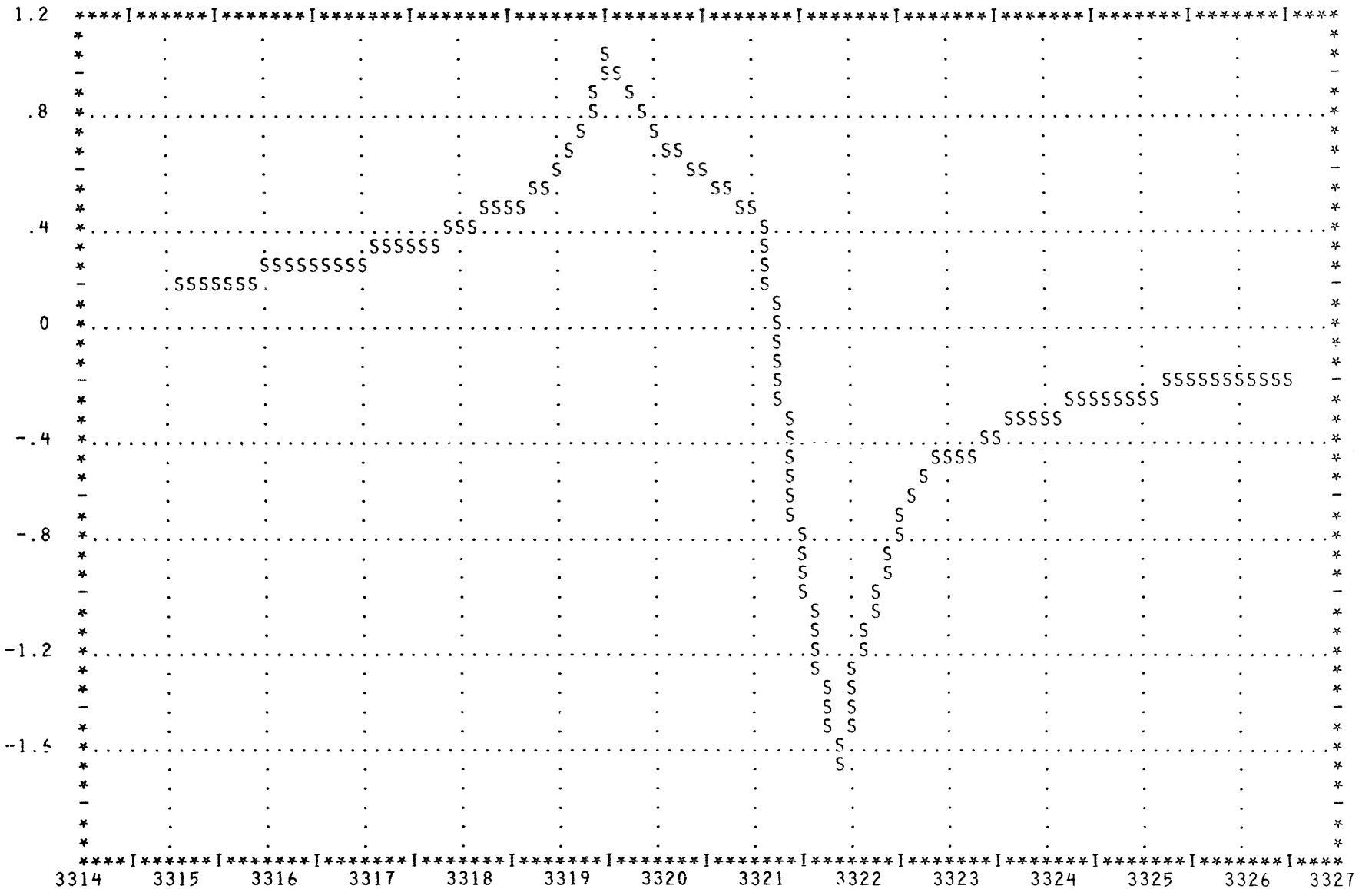
D 1

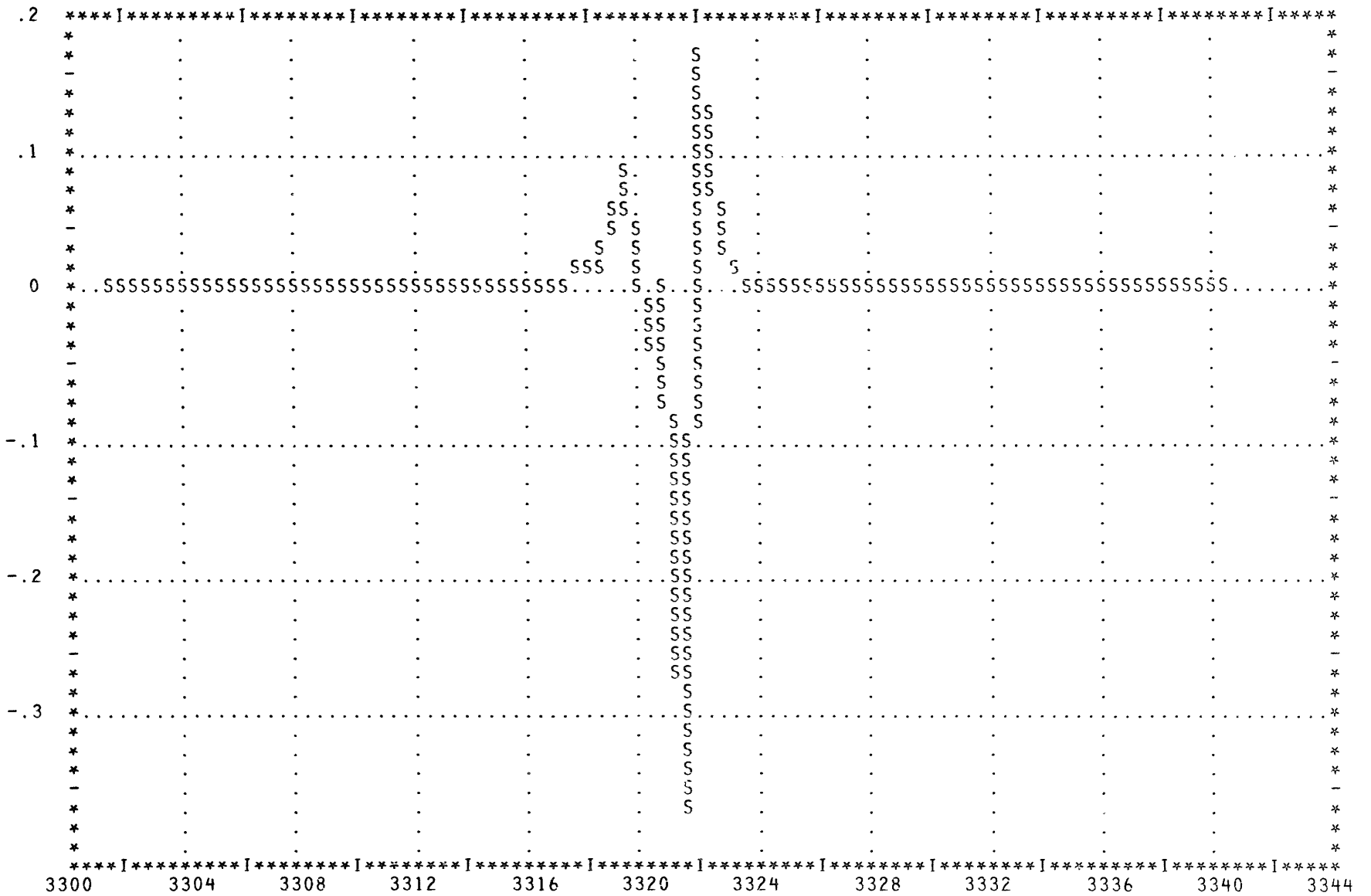

.12
.08
.04
0
-.04
-.08
-.12
-.16
3300
3304
3308
3312
3316
3320
3324
3328
3332
3336
3340
3344


2
1
0
-1
-2
-3
-4
-5
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327

```
E  1

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 6,

$END
 IFUNC   =   6
 GNU     =        .93000000000000E+10
 PLANC   =        .10545887000000E-26
 BETA    =        .92740780000000E-20
 PI      =        .31415926535898E+01
 HALF    =        .10000000000000E+00
 OVRHALF =        .99999999999999E+01
 HZERO   =        .33195269359482E+04
 HONE    =        .33218502563053E+04
 H2      =        .33213521282053E+04
 H3      =        .33195269359482E+04
 H1S     =        .11034689125316E+08
 H2S     =        .11031379959534E+08
 H3S     =        .11019259078486E+08
 G1      =        .20003000000000E+01
 G2      =        .20006000000000E+01
 G3      =        .20017000000000E+01
 S       =        .44007492998454E-05
 XLOG2   =        .69314718055995E+00
 PILOG   =        .93943727869965E-01
 TONE    =        .25000000000000E-03
 GAMMA   =        .17602997199382E+08
 H1STAR  =        .10000000000000E-04
 HLFSTR  =        .50000000000000E+01
 OHLFSTR =        .20000000000000E+00
 DELTAH  =        .20000000000000E+00
 DELTAHS =        .10000000000000E+02
```

```
E  2

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 8,

$END
 IFUNC    =    8
 GNU      =        .93000000000000E+10
 PLANC    =        .10545887000000E-26
 BETA     =        .92740780000000E-20
 PI       =        .31415926535898E+01
 HALF     =        .10000000000000E+00
 OVRHALF  =        .99999999999999E+01
 HZERO    =        .33195269359482E+04
 HONE     =        .33218502563053E+04
 H2       =        .33213521282053E+04
 H3       =        .33195269359482E+04
 H1S      =        .11034689125316E+08
 H2S      =        .11031379959534E+08
 H3S      =        .11019259078486E+08
 G1       =        .20003000000000E+01
 G2       =        .20006000000000E+01
 G3       =        .20017000000000E+01
 S        =        .44007492998454E-05
 XLOG2    =        .69314718055995E+00
 PILOG    =        .93943727869965E-01
 TONE     =        .25000000000000E-03
 GAMMA    =        .17602997199382E+08
 H1STAR   =        .10000000000000E-04
 HLFSTR   =        .50000000000000E+01
 OHLFSTR  =        .20000000000000E+00
 DELTAH   =        .20000000000000E+00
 DELTAHS  =        .10000000000000E+02
```


1.2
.8
.4
0
-.4
-.8
-1.2
-1.6
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327


.2
.1
0
-.1
-.2
-.3
3300
3304
3308
3312
3316
3320
3324
3328
3332
3336
3340
3344

```
H  1

$VITKO

G1      = .20003E+01,

G2      = .20006E+01,

G3      = .20017E+01,

TONE    = .25E-03,

DELTAH  = .2E+00,

DELTAHS = .1E+02,

H1STAR  = .1E-04,

GNU     = .93E+10,

IFUNC   = 7,

$END
 IFUNC    =    7
 GNU      =        .93000000000000E+10
 PLANC    =        .10545887000000E-26
 BETA     =        .92740780000000E-20
 PI       =        .31415926535898E+01
 HALF     =        .10000000000000E+00
 OVRHALF  =        .99999999999999E+01
 HZERO    =        .33195269359482E+04
 HONE     =        .33218502563053E+04
 H2       =        .33213521282053E+04
 H3       =        .33195269359482E+04
 H1S      =        .11034689125316E+08
 H2S      =        .11031379959534E+08
 H3S      =        .11019259078486E+08
 G1       =        .20003000000000E+01
 G2       =        .20006000000000E+01
 G3       =        .20017000000000E+01
 S        =        .44007492998454E-05
 XLOG2    =        .69314718055995E+00
 PILCG    =        .93943727869965E-01
 TONE     =        .25000000000000E-03
 GAMMA    =        .17602997199382E+08
 H1STAR   =        .10000000000000E-04
 HLFSTR   =        .50000000000000E+01
 OHLFSTR  =        .20000000000000E+00
 DELTAH   =        .20000000000000E+00
 DELTAHS  =        .10000000000000E+02
```