JUL 11 1963

325

7-11-63

ORNL-3429
UC-34 - Physics
TID-4500 (20th ed.)

MASTER

A FORTRAN PROGRAM FOR CALCULATING THE
SCATTERING OF NUCLEONS FROM A
NONLOCAL OPTICAL POTENTIAL

F. G. Perey

OAK RIDGE NATIONAL LABORATORY

operated by

UNION CARBIDE CORPORATION

for the

U.S. ATOMIC ENERGY COMMISSION

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of 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. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Printed in USA. Price: \$1.75 Available from the
Office of Technical Services
U. S. Department of Commerce
Washington 25, D. C.

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

- A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or
- B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

Contract No. W-7405-eng-26

Neutron Physics Division

A FORTRAN PROGRAM FOR CALCULATING
THE SCATTERING OF NUCLEONS FROM A NONLOCAL OPTICAL POTENTIAL

F. G. Perey

Date Issued

JUL 11 1963

OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee
Operated by
UNION CARBIDE CORPORATION
for the
U. S. ATOMIC ENERGY COMMISSION

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

ABSTRACT

In this report is given the listing of a FORTRAN program for calculating the scattering of nucleons from a nonlocal optical potential. The mathematical formulation of the problem is given, together with the numerical methods used in the code. The input to the program is explained and a brief functional description of each subroutine of the code is given.

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

Table of Contents

	<u>Page No.</u>
Abstract.....	iii
Introduction.....	1
Mathematical Formulation.....	1
Description of Input to the Code.....	9
Brief Description of the Function of Each Subroutine.....	14
Acknowledgments.....	18
Appendix.....	19

INTRODUCTION

This version of the FORTRAN program for calculating the elastic scattering of nucleons by a nonlocal optical potential grew out of many attempts at the numerical solution of the problem. The author regrets that many of the idiosyncrasies of the earlier versions of the code have been retained (originally, to facilitate intercomparison of the codes), but he did not think it worthwhile to rewrite a code of this magnitude to eliminate some of the clumsiness in the programming.

The physical justifications, mathematical formulation, and comparison with neutron experimental data up to 24 Mev are given elsewhere.¹ The mathematical formulation of the problem is given again in this report to facilitate understanding of the limitations of the numerical solution and of the coding. Also given are a description of the input parameters and their restrictions, a brief description of each subroutine, and the flow of logic of the calculation. The listing of each subroutine and some test cases are presented as an appendix.

MATHEMATICAL FORMULATION

The integrodifferential Schrödinger equation is

$$\left(\frac{\hbar^2}{2M} \nabla^2 + E \right) \Psi(\underline{r}) = - \left[(U_{SP} + iW_{SP}) S(r) \underline{L} \cdot \underline{\sigma} + V_C(r) \right] \Psi(\underline{r}) + \int V(\underline{r}, \underline{r}') \Psi(\underline{r}') d\underline{r}', \quad (1)$$

where M is the reduced mass of the incident particle and E its energy in the center-of-mass system of coordinates.

The kernel $V(\underline{r}, \underline{r}')$ is taken to be separable and is written in the form

$$V(\underline{r}, \underline{r}') = U \left[\frac{1}{2} (r + r') \right] H(|\underline{r} - \underline{r}'|). \quad (2)$$

1. F. G. Perey and B. Buck, Nuclear Phys. 32, 353 (1962).

It would have been preferable to take $\frac{1}{2} |\underline{r} + \underline{r}'|$ for argument of U , but this complicates very much the calculation, and the simpler choice indicated in Eq. 2 was made. Hence $V(\underline{r}, \underline{r}')$ depends on the variables r, r' and $\cos X$, where X is the angle between the vectors $\underline{r}$ and $\underline{r}'$.

$H(\underline{r}, \underline{r}')$ is chosen to be a gaussian function:

$$H(|\underline{r} - \underline{r}'|) = \frac{\exp \left[- \left(\frac{r - r'}{\beta} \right)^2 \right]}{\pi^{3/2} \beta^3}, \quad (3)$$

which is normalized so that

$$\int H(|\underline{r} - \underline{r}'|) d\underline{r}' = 1. \quad (4)$$

We define

$$p = \frac{1}{2} (r + r'). \quad (5)$$

The form chosen for $U(p)$ in Eq. 2 is similar to those employed in local optical model calculations:

$$- U(p) = (V_S + iW_S) f_S(p) + iW_D f_D(p), \quad (6)$$

where

$$f_S(p) = \left[1 + \exp \left(\frac{p - R_S}{a_S} \right) \right]^{-1}, \quad R_S = r_{OS} A^{1/3}, \quad (7)$$

$$f_D(p) = 4 \exp \left(\frac{p - R_D}{a_D} \right) \left[1 + \exp \left(\frac{p - R_D}{a_D} \right) \right]^{-2}, \quad R_D = r_{OD} A^{1/3}. \quad (8)$$

Here $f_S(p)$ is a Saxon form factor, while $f_D(p)$ gives a surface imaginary term of the Saxon derivative type which has unity at its maximum value. Note that the variable is p and not r , as in local calculations.

The spin-orbit function in Eq. 1 is

$$S(r) = \left(\frac{\hbar}{2Mc} \right)^2 \frac{1}{a_S r} \frac{\exp \left(\frac{r - R_S}{a_S} \right)}{\left[1 + \exp \left(\frac{r - R_S}{a_S} \right) \right]^2}, \quad (9)$$

where M is the nucleon mass. The function $S(r)$ is often defined with M as the pi meson mass; in this case the coefficients U_{SP} and W_{SP} in Eq. 1 are smaller by a factor of 180.59.

The coulomb potential in Eq. 1 is assumed to come from a uniform charge distribution of radius R_S and is therefore

$$V_C(r) = \frac{Ze^2}{2R_S} \left(3 - \frac{r^2}{R_S^2} \right), \quad \text{for } r \leq R_S,$$

$$V_C(r) = \frac{Ze^2}{r}, \quad \text{for } r > R_S. \quad (10)$$

The functions $\Psi(\underline{r})$, $\Psi(\underline{r}')$, and $V(\underline{r}, \underline{r}')$ in Eq. 1 are expanded in partial waves:

$$\Psi(\underline{r}) = \sum_{j\ell m} \frac{u_{j\ell}(r)}{r} C_{m\lambda\sigma}^{j\ell s} \left\{ i^\ell Y_\ell^\lambda(\theta, \phi) \right\} X_s^\sigma, \quad (11)$$

$$\Psi(\underline{r}') = \sum_{j'\ell'm'} \frac{u_{j'\ell'}(r')}{r'} C_{m'\lambda'\sigma'}^{j'\ell's'} \left\{ i^{\ell'} Y_{\ell'}^{\lambda'}(\theta', \phi') \right\} X_{s'}^{\sigma'}, \quad (12)$$

$$V(\underline{r}, \underline{r}') = \sum_{LM} \frac{g_L(r, r')}{rr'} \left\{ i^L Y_L^M(\theta, \phi) \right\} \left\{ i^L Y_L^M(\theta', \phi') \right\}^*. \quad (13)$$

Substitution of Eqs. 11-13 in Eq. 1 yields the following equation for the radial functions $u_{j\ell}(r)$:

$$\begin{aligned} \frac{\hbar^2}{2M} \left[\frac{d^2}{dr^2} - \frac{\ell(\ell+1)}{r^2} \right] u_{j\ell}(r) + E u_{j\ell}(r) \\ = - \left[(U_{SP} + iW_{SP}) S(r) T(j\ell) + V_C(r) \right] u_{j\ell}(r) + \int_0^\infty g_\ell(r, r') u_{j\ell}(r') dr', \end{aligned} \quad (14)$$

where

$$T(j\ell) = j(j+1) - \ell(\ell+1) - \frac{3}{4}. \quad (15)$$

From the expansion of $V(\underline{r}, \underline{r}')$ we easily obtain

$$\begin{aligned} g_\ell(r, r') = 2\pi r r' U \left[\frac{1}{2} (r + r') \right] \\ \times \int_{-1}^{+1} \frac{\exp \left[- \left(\frac{r - r'}{\beta} \right)^2 \right]}{\pi^{3/2} \beta^3} P_\ell(\cos X) d(\cos X). \end{aligned} \quad (16)$$

Now let

$$g_\ell(r, r') = U \left[\frac{1}{2} (r + r') \right] h_\ell(r, r'). \quad (17)$$

Then, a straightforward evaluation of the integral in Eq. 16 gives

$$h_\ell(r, r') = \frac{1}{\pi^{1/2} \beta} \exp \left[- \left(\frac{r^2 + r'^2}{\beta^2} \right) \right] K_\ell(z), \quad (18)$$

where

$$K_\ell(z) = 2 i^\ell z j_\ell(-iz), \quad (19)$$

$$z = 2 \frac{rr'}{\beta}. \quad (20)$$

In Eq. 19 $j_\ell(x)$ is a spherical Bessel function. The following expression for $K_\ell(z)$ is analytically exact for all z :

$$K_\ell(z) = Q_\ell(z) \exp(z) + (-)^{\ell+1} Q_\ell(-z) \exp(-z), \quad (21)$$

where $Q_\ell(z)$ is a finite polynomial in z^{-1} , given by

$$Q_\ell(z) = \sum_{S=0}^{\ell} A_S(\ell) (-z)^{-S}, \quad A_0(\ell) = 1, \quad (22)$$

$$A_S(\ell) = \left[\frac{\ell(\ell+1) - S(S+1)}{2S} \right] A_{S-1}(\ell). \quad (23)$$

Thus we obtain the exact expression

$$\begin{aligned} h_\ell(r, r') = \frac{1}{\pi^{1/2} \beta} \left[Q_\ell \left(2 \frac{rr'}{\beta^2} \right) \exp \left\{ - \left(\frac{r - r'}{\beta} \right)^2 \right\} \right. \\ \left. + (-)^{\ell+1} Q_\ell \left(- 2 \frac{rr'}{\beta^2} \right) \exp \left\{ - \left(\frac{r + r'}{\beta} \right)^2 \right\} \right]. \quad (24) \end{aligned}$$

When z is small, the exact expression (21) is not useful for numerical work because of large cancellations between the component terms. Hence for small z it is better to expand $K_\ell(z)$ as a power series in z :

$$K_\ell(z) = C(\ell) z^{\ell+1} \sum_{S=0}^{\infty} B_S(\ell) z^S, \quad (25)$$

$$C(\ell) = \frac{2}{(2\ell + 1)!!}, \quad B_0(\ell) = 1, \quad B_1(\ell) = 0, \quad (26)$$

$$B_S(\ell) = \frac{B_{S-2}(\ell)}{S(S + 2\ell + 1)}. \quad (27)$$

The expansion (25) is used until more than a preassigned number of terms are required to obtain an accuracy of one part in 10^3 . Then the exact form (24) is employed.

Various methods were tried for solving the integrodifferential Eq. 14; the one which proved the fastest was an iteration procedure. In order to start the iteration and to improve on the convergence of the method, use is made of the local WKB (Wentzel, Kramers, and Brillouin) approximation to the nonlocal potential. We are going to replace the integral of Eq. 1 by an approximate local expression involving only r . The integral in Eq. 1 can be written

$$I = \int U \left[\frac{1}{2} (\underline{r} + \underline{r}') \right] \frac{\exp \left[- \left(\frac{\underline{r} - \underline{r}'}{\beta} \right)^2 \right]}{\pi^{3/2} \beta^3} \bar{\Psi}(\underline{r}') d\underline{r}'. \quad (28)$$

Note that in order to derive the WKB approximation the argument of U is $\frac{1}{2} (\underline{r} + \underline{r}')$. The approximate expression we want is of the form

$$I \approx U_L(r) \bar{\Psi}(\underline{r}). \quad (29)$$

Let

$$\underline{r}' - \underline{r} = \beta \underline{s}, \quad (30)$$

so that $\frac{1}{2} (\underline{r} + \underline{r}') = \underline{r} + \frac{1}{2} \beta \underline{s}$. Substitution in the integral gives

$$I = \int U \left(\underline{r} + \frac{1}{2} \beta \underline{s} \right) \frac{\exp(-s^2)}{\pi^{3/2}} \bar{\Psi}(\underline{r} + \beta \underline{s}) d\underline{s}. \quad (31)$$

Using the operator form of the Taylor expansion, the integral I can be written as an operator expression:

$$I = \left\{ \int \exp \left[\beta \underline{s} \cdot \left(\frac{1}{2} \nabla_1 + \nabla_2 \right) \right] \frac{\exp(-s^2)}{\pi^{3/2}} d\underline{s} \right\} U(\underline{r}) \Psi(\underline{r}), \quad (32)$$

where

$$\nabla_1 \text{ operates only on } U(\underline{r}), \nabla_2 \text{ operates only on } \Psi(\underline{r}). \quad (33)$$

Treating the expression $\frac{1}{2} \nabla_1 + \nabla_2$, for the moment, as an algebraic quantity, the integral over $d\underline{s}$ is easily evaluated and the following operator form is obtained for the integral:

$$I = \exp \left[\frac{\beta^2}{4} \left(\frac{1}{2} \nabla_1 + \nabla_2 \right)^2 \right] U(\underline{r}) \Psi(\underline{r}). \quad (34)$$

Making the approximation that ∇_1 operating on U can be neglected, since in the interior of the nucleus U(r) is isotropic, then

$$I \cong U(\underline{r}) \exp \left(\frac{\beta^2}{4} \nabla^2 \right) \Psi(\underline{r}). \quad (35)$$

By substituting the form in Eq. 29 for the integral in Eq. 1 and neglecting the spin-orbit potential,

$$\nabla^2 \Psi(\underline{r}) = - \frac{2M}{\hbar^2} \left[E - V_C(r) - U_L(r) \right] \Psi(\underline{r}). \quad (36)$$

In Eq. 35 the approximation is made that the operator ∇^2 can be replaced by its numerical value as given in Eq. 36; the desired result is then obtained:

$$U_L(r) = U(r) \exp \left[- \frac{M\beta^2}{2\hbar^2} \left\{ E - V_C(r) - U_L(r) \right\} \right]. \quad (37)$$

In the program Eq. 37 is solved by an iteration procedure at the first mesh point and calculated at all other mesh points by integrating the differential equation obtained by the differentiation of Eq. 37.

The iteration scheme will now be explained, but to simplify the notation, let

$$E_{j\ell}^* = E - \frac{\ell(\ell+1)}{r^2} + (U_{SP} + iW_{SP}) S(r) T(j\ell) - V_C(r); \quad (38)$$

then Eq. 14 is rewritten as

$$\frac{\hbar^2}{2M} \frac{d^2}{dr^2} u_{j\ell}(r) + E_{j\ell}^*(r) u_{j\ell}(r) = \int_0^{R_M} g_\ell(r, r') u_{j\ell}(r') dr'. \quad (39)$$

The upper limit R_M is the radius at which the kernel function becomes negligible; that is, $g_\ell(r, r') \approx 0$ for $r, r' \geq R_M$. Thus R_M is the radius for which the solutions $u(r)$ are matched to the free-state wave functions in order to obtain the scattering matrix elements. The details of the computation of the free-state Coulomb functions and of the S-matrix elements are very similar to those of ref 2.

When the local WKB potential U_L is subtracted from both sides of Eq. 39 and the subscripts $j\ell$ are dropped for clarity, then

$$\begin{aligned} \frac{\hbar^2}{2M} \frac{d^2}{dr^2} u(r) + [E^*(r) - U_L(r)] u(r) \\ = \left[\int_0^{R_M} g_\ell(r, r') u(r') dr' - U_L(r) u(r) \right]. \end{aligned} \quad (40)$$

If the suffix n denotes the n th-order approximation to the correct solution, the iteration procedure is written

2. B. Buck, R. N. Maddison, and P. E. Hodgson, Phil. Mag. 5, 1181 (1960).

$$\frac{\hbar^2}{2M} \frac{d^2}{dr^2} u_0(r) + \left[E^*(r) - U_L(r) \right] u_0(r) = 0, \quad (41)$$

$$\begin{aligned} \frac{\hbar^2}{2M} \frac{d^2}{dr^2} u_n(r) + \left[E^*(r) - U_L(r) \right] u_n(r) \\ = \left[\int_0^{R_M} g_\ell(r, r') u_{n-1}(r') dr' - U_L(r) u_{n-1}(r) \right]. \end{aligned} \quad (42)$$

Notice that Eq. 41 for $u_0(r)$ is a homogeneous local wave equation so that the scheme is easily started. The inhomogeneous iterations, Eq. 42, now yield a set of functions $u_n(r)$ which rapidly converge to the true solution $u(r)$. The iteration is continued until the logarithmic derivative at R_M , obtained from u_n , agrees to within a preassigned percentage with that obtained from u_{n-1} .

DESCRIPTION OF INPUT TO THE CODE

Since the nonlocal matrices $h_\ell(r, r')$ (Eq. 18) are only a function of β , the nonlocality range, they need be calculated only once for all computations using the same value of β and the same integration mesh size. Therefore provision is made in the code to generate and update a "library tape" of matrices $h_\ell(r, r')$. Since, frequently, several calculations are made successively while the code is in core and since searching the library tape for the appropriate $h_\ell(r, r')$ would be time-consuming, provision is made in the code to write the matrices $h_\ell(r, r')$ on a scratch tape to be used for several computations.

The code has therefore three functions: to generate nonlocal matrices to be added to a library tape of nonlocal matrices (on logical tape 3), to transfer nonlocal matrices from the library tape to a scratch tape (logical tape 1), and to do a nonlocal optical model calculation using the nonlocal matrices on logical tape 1.

Most of the input numbers to the code are in floating-point format (the number may be anywhere in the field indicated but must have a decimal point punched), except a few specifically indicated control integers, which are in fixed-point format (no decimal point, and the number must occupy the right-most position in the field).

The three functions of the program are controlled by a Master Control Card, which must precede every calculation. The system is described below:

MASTER CONTROL CARD

Column 1

If = 1: Function I

If = 2: Function II

If = 3: Function III

Function I: Calculation of nonlocal matrices

First Card: Master control card

Second Card: Matrix parameters (format 4F10.8,3I2)

Column 1-10: h_a , step length (in fermi) of last set of matrices on tape

11-20: β_a , nonlocality range (in fermi) of last set of matrices on tape (to start a new tape $h_a = \beta_a = 0$ or blank)

21-30: h_b , step length (in fermi) of new matrices to be added to tape

31-40: β_b , nonlocality range (in fermi) of new matrices to be added to tape

41-42: N_T (fixed point), number of partial waves (a maximum of 25 partial waves may be used)

43-44: N_B (fixed point), number of terms in asymptotic expansion of $h_\ell(r, r')$; may be left blank, in which case 5 terms will be used but up to 10 terms may be used

45-46: I_p , if = 01, a list of the matrices on the tape will
be printed;

if = 00 or blank, no list will be printed

Function II: Transferring nonlocal matrices from logical tape 3 to
logical tape 1

First Card: Master control card

Second Card: Matrix parameters (format 2F10.8)

Column 1-10: h_a , step length (in fermi) of matrices to be
transferred

11-20: β_a , nonlocality range (in fermi) of matrices to be
transferred

Note: The matrices must be on logical tape 3; otherwise, a tape error
will occur

Function III: Nonlocal optical model calculation

First Card: Master control card

Second Card: Experimental parameters (format 5F10.8)

Column 1-10: ELAB, laboratory energy (in MeV) of incident particle

11-20: AMT, mass of target nuclei (in AMU)

21-30: AMI, mass of incident particle (in AMU)

31-40: ZT, charge of target nuclei (in electron units)

41-50: ZI, charge of incident particle (in electron units)

Third Card: Optical potential parameters I (format 7F10.8)

Column 1-10: US, Saxon potential real-well depth (in MeV)

11-20: RS, Saxon potential radius parameter (r_{OS} in fermi)

21-30: AS, Saxon potential diffuseness parameter (a_S in fermi)

31-40: WS, Saxon potential imaginary-well depth (in MeV)

41-50: WD, Derivative Saxon potential imaginary-well depth
(in MeV)

51-60: RD, Derivative Saxon potential radius parameter
(r_{OD} in fermi)

61-70: AD, Derivative Saxon potential diffuseness parameter
(a_D in fermi)

Fourth Card: Optical potential parameters II (format 3F10.8)

Column 1-10: VSP, real-well depth of spin-orbit potential
(in MeV)

11-20: WSP, imaginary-well depth of spin-orbit potential
(in MeV)

21-30: BETA, nonlocality range (in fermi)

Notes: VSP positive gives the same convention as normal shell model;
BETA is not used except for output purposes; the calculation is
made with whatever matrices are on tape 1

Fifth Card: Numerical parameters (format 6F10.8, I2)

Column 1-10: H, integration step length (in fermi) -- must be the
same as the step length of nonlocal matrices on
logical tape 1

11-20: HC, Coulomb subroutine integration step length (in
fermi) (may be different from H; a value of 0.1
is satisfactory)

21-30: C, cutoff parameter in iteration scheme (percentage
difference in the logarithmic derivative between two
successive iterations to accept convergence; the value
0.0005 is usually sufficient)

31-40: THETA1 ($\theta_{\min}$)	}	$\sigma(\theta)$ will be computed from $\theta_{\min}$ to $\theta_{\max}$ in $\Delta\theta$ steps; the θ 's are in degrees; not more than 100 angles are allowed; $\theta_{\min} \neq 0$ for charged particles
41-50: THETA2 ($\Delta\theta$)		
51-60: THETA3 ($\theta_{\max}$)		

61-62: NC (fixed point), should be greater than the number
of partial waves needed for the calculation (see
limitations below)

Sixth Card: Derivative control card (format 11,8X,11,9I3); numbers are all in fixed point on this card

Column 1: ISP, if = 0 or blank, no spin-orbit potential
if = 1 for spin 1/2 projectile and with a spin-orbit potential

10: NV = number of parameters with respect to which the derivatives of the S-matrix elements are desired; if none desired, leave blank

11-13: IPA(1) } the IPA's are key numbers to
14-16: IPA(2) } indicate the parameter (see below);
17-19: IPA(3) } there should be NV entries in the IPA
 ⋮ } vector
up to IPA(9)

Key numbers for the derivatives:

1 ≡ US
2 ≡ RS
3 ≡ AS
4 ≡ WS
5 ≡ WD
6 ≡ RD
7 ≡ AD
8 ≡ USP
9 ≡ WSP

Example: if the derivatives with respect to RS and WD are wanted, NV = 2
IPA(1) = 2, IPA(2) = 5.

Limitations in Values of Input Parameters and Error Returns

Because of the various computational techniques used and because of the space allotted to the various arrays in the computer, the calculation is either unreliable or cannot proceed if the input parameters are not within a certain range. To the extent that those restrictions were known to the author at the time of programming, several checks have been incorporated in the code to test the calculation, and various messages are

printed out when an error condition is detected. The following limitations are known to apply to this version of the code:

1. The matching radius R_M is taken as the largest of the two following quantities:

$$r_{OS} A^{1/3} + 7a_S, \quad r_{OD} A^{1/3} + 7a_D.$$

The limitation on step length for a given matching radius is

$$H \leq \frac{R_M}{97}.$$

If this condition is not met, the code word DATA is printed.

2. For a given step length H there is a limitation on the size of β , the nonlocality range, since at most 30 off-diagonal elements are used in the matrix $g_\ell(r, r')$ (Eq. 17). For the calculation to proceed, the criterion $g_\ell(r, r + 30H) \leq g_0(R_M, R_M)$ must be met for every partial wave. This criterion may be changed by altering the value of GRM and GIM in subroutine NEWMAT. When the above criterion is not met, the code word MATRIX is printed, and the calculation proceeds to the next case.

3. The value of NC should be greater than $kR_M - \eta$, where η is the Coulomb parameter. A value of $kR_M + 5$ is usually satisfactory. In the Coulomb subroutine if the Wronskian of the free-state equation is not 1, the message COULOMB is printed and the value of NC should be checked.

4. When the iteration scheme has not converged after 25 iterations, the message INTEGRATION is printed, indicating a breakdown of the method. The value of the potential parameters should be modified.

5. If the value of NC is too small, the message LMAX TOO SMALL is printed, and NC should be increased, 25 being the maximum value it may have.

BRIEF DESCRIPTION OF THE FUNCTION OF EACH SUBROUTINE

The program contains a short main program to control the flow of logic and a series of subroutines to perform the calculations. With one exception the subroutines have no arguments transmitted by the calling

program. The quantities of interest are stored in common. The main program reads the Master Control Card and transfers control to subroutine ADTAPE for Function I, to subroutine GETHL2 for Function II, and to subroutine INPUT4 for Function III.

Function I: Calculation of Nonlocal Matrices

The nonlocal matrices $h_{\ell}(r, r')$ are symmetric with the largest elements along the diagonal, the magnitude of the elements decreasing rapidly as one moves off the diagonal. Because of storage limitation, only a band 30 elements wide on one side of the diagonal is stored in the computer. The length of the diagonal is 100 elements.

ADTAPE: Sets up the library tape and handles the writing of the matrices, which are obtained by calling MATRHL.

MATRHL: Controls the packing and unpacking of the matrices and calls the subroutine HL(I,J), which computes the elements by using Eqs. 21 or 25, depending on the size of $\frac{rr'}{\beta^2}$.

Function II: Transfer of Matrices

GETHL2: Searches the logical tape 3 for the appropriate matrices and transfers them to logical tape 1.

Function III: Nonlocal Optical Model Calculation

INPUT4: Reads the input cards described in "Description of Input to the Code."

SANGLE: Sets up a vector with the angles at which the cross sections are to be calculated.

PRINPU: Prints the input parameters.

DATAPR: Converts the various input parameters to the same unit system in the center-of-mass coordinate system. The matching radius is calculated and if there is not enough space, the code word DATA is printed.

VECTOR: Computes the various potentials at each mesh point, as well as their derivatives with respect to the parameters. The local WKB approximation is also calculated here.

COULOM: Calculates the Coulomb phase shifts and the regular and irregular Coulomb functions and their derivatives at the matching radius. If the Wronskian fails to be 1, the calculation stops and the code word COULOMB is printed.

At this point a loop is entered to calculate the S-matrix elements for each partial wave. The loop terminates when the S-matrix elements have reached their asymptotic value of 1. With $\ell = 0$ the loop is entered at subroutine NEWMAT.

NEWMAT: Sets up the matrix $g_\ell(r, r') = U \left[\frac{1}{2} (r + r') \right] h_\ell(r, r')$, reading the $h_\ell(r, r')$ matrices from logical tape 1. The matrix $g_\ell(r, r')$ is stored in the computer in the same manner as the matrix $h_\ell(r, r')$. If, because of the step length used and the nonlocality range, more than 30 off-diagonal elements are required, the code word MATRIX is printed.

INTEGR: The radial equation is solved by the iteration scheme given previously. After each iteration the logarithmic derivative of the wave function is calculated and the percentage deviation from the value obtained in the previous iteration is compared with the input number C. If the percentage deviation is less than C, the solution is accepted; if it is greater than C, another iteration is done, and so on until the criterion is met. If the criterion has not been met after 25 iterations, the code word INTEGRATION is printed.

SCMATR: The logarithmic derivative of the radial wave function is matched to the logarithmic derivative of the free-state wave function to obtain the scattering matrix element.

- DERIVS: If the derivatives of the scattering matrix elements are required, they are calculated in this subroutine. The method of calculation is given elsewhere.³
- CLEAN: If spin $1/2$ projectile is used, the first pass in subroutine integration is done with $j = \ell + 1/2$, and control is returned to subroutine INTEGR for the case $j = \ell - 1/2$. The S-matrix elements are checked to determine whether they have reached their asymptotic value ($\text{Re } S_\ell > 0.999$, $\text{Im } S_\ell < 0.001$). If they have not, ℓ is increased by 1 and control is returned to NEWMAT for a new partial wave. If the scattering matrix elements have reached their asymptotic value, control is transferred to subroutine SIGMAS. Each time ℓ is increased in the above loop, it is verified that enough free-state Coulomb functions have been calculated; if not, the code word LMAX TOO SMALL is printed.
- SIGMAS: Calculates the differential cross sections and the polarization at the angle assigned by SANGLE.
- PRSIGM: Prints the differential cross section and the S-matrix elements (next to the S-matrix elements, under the column IT the number of iterations for each partial wave is given); if required, the derivatives of the S-matrix elements are printed.

3. F. G. J. Perey and B. Buck, Neutron Phys. Div. Ann. Progr. Rept. Sept. 1, 1961, ORNL-3193, p 250.

ACKNOWLEDGMENTS

It is with great pleasure that I acknowledge the collaboration of B. Buck in the formulation of the problem and the investigation of the numerical methods to solve the integrodifferential Schrödinger equation. I would like also to thank G. R. Satchler for many discussions and suggestions on the scattering of nucleons from nonlocal potentials.

APPENDIX

Listing of FORTRAN Subroutines

and

Test Cases for IBM-7090 and CDC-1604

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

*TYPE(FORTRAN)	000
DIMENSION F(200)	001
COMMON F	002
EQUIVALENCE (F(61),K0),(F(29),J3),(F(51),KS),	003
I(F(47),IL),(F(45),LC),(F(63),I7)	004
1 READ INPUT TAPE 10,2,K0,I7	005
2 FORMAT (11)	006
IF (K0-1) 100,100,101	007
100 CALL ADTAPE	008
GO TO 1	009
101 IF (K0-2) 3,3,4	010
3 CALL GETHL2	011
GO TO 1	012
4 CALL INPUT4	013
J3=0	014
CALL SANGLE	015
CALL PRINPU	016
CALL DATAPR	017
IF (LC) 5,5,12	018
5 CALL VECTOR	019
CALL COULOM	020
IF (LC-1) 6,6,14	021
6 KS=0	022
IL=0	023
7 CALL NEWMAT	024
IF (LC) 8,8,16	025
8 CALL INTEGR	026
IF (LC) 9,9,18	027
9 CALL SCMATR	028
CALL DERIVS	029
CALL CLEAN	030
IF (LC) 7,8,10	031
10 IF (LC-1) 11,11,20	032
11 CALL SIGMAS	033
CALL PRSIGM	034
GO TO 1	035

12 WRITE OUTPUT TAPE 9,13	036
GO TO 1	037
13 FORMAT (5H0DATA)	038
14 WRITE OUTPUT TAPE 9,15	039
15 FORMAT(8H0COULOMB)	040
GO TO 1	041
16 WRITE OUTPUT TAPE 9,17	042
17 FORMAT(7H0MATRIX)	043
GO TO 1	044
18 WRITE OUTPUT TAPE 9,19	045
19 FORMAT(12H0INTEGRATION)	046
GO TO 1	047
20 WRITE OUTPUT TAPE 9,21	048
21 FORMAT(15H0LMAX TOO SMALL)	049
GO TO 1	050
END	051

*TYPE(FORTRAN)	052
SUBROUTINE ADTAPE	053
DIMENSION D(15000),X(300),HLMAT(100,30)	054
COMMON D	055
EQUIVALENCE (D(15),H),(D(10),BETA),(D(17),M),	056
(D(16),NB),(D(51),IL),(D(201),X(1)),(D(2111),HLMAT)	057
READ INPUT TAPE 10,1,H1,BETA1,H,BETA,M,NB,LIST	058
1 FORMAT (4F10.8,3I2)	059
REWIND 3	060
IF (H1) 22,22,21	061
21 IF (LIST) 4,4,2	062
2 WRITE OUTPUT TAPE 9,3	063
3 FORMAT (36H1 NON-LOCALITY MATRICES ON TAPE/	064
148H0 STEP LENGTH BETA NO.OF MATRICES)	065
4 DO 11 I=1,300	066
READ TAPE 3,H2,BETA2,M2	067
IF (LIST) 7,7,5	068
5 WRITE OUTPUT TAPE 9,6,H2,BETA2,M2	069
6 FORMAT (1H0,F17.8,F16.8,I9)	070
7 IF (H2-H1) 10,8,10	071
8 IF (BETA1-BETA2) 10,9,10	072
9 GO TO 12	073
10 DO 11 J=1,M2	074
11 READ TAPE 3	075
12 DO 13 I=1,M2	076
13 READ TAPE 3	077
22 WRITE TAPE 3,H,BETA,M	078
X(1)=H	079
DO 14 I=2,300	080
14 X(I)=X(I-1)+H	081
IL=0	082
IF (NB) 15,15,16	083
15 NB=5	084
16 DO 17 I6=1,M	085
CALL MATRHL	086
WRITE TAPE 3,((HLMAT(I,J),I=1,100),J=1,30)	087


```
17 IL=IL+1
   IF (LIST) 20,20,18
18 WRITE OUTPUT TAPE 9,19,H,BETA,M
19 FORMAT (4H0NEW,F14.8,F16.8,19)
20 RETURN
   END
```

```
088
089
090
091
092
093
```

*TYPE(FORTRAN)	094
SUBROUTINE MATRHL	095
DIMENSION D(15000),X(300),EX(300),	096
HLMAT(100,30),SMZ(900),A(30),B(30)	097
COMMON D	098
EQUIVALENCE (D(64),AL),(D(51),IL),(D(10),BETA),(D(52),LO),	099
1(D(16),NB),(D(53),FAC),(D(54),IZ1),(D(55),IZ2),	100
2(D(59),J1),(D(58),IS),(D(60),HLR),(D(201),X(1)),	101
3(D(1811),EX(1)),(D(2111),HLMAT),(D(11561),SMZ(1)),	102
4(D(12461),A(1)),(D(12491),B(1))	103
AL=IL	104
IF (IL) 1,1,4	105
1 B1=1./(1.7724539*BETA)	106
DO 2 I=1,300	107
2 EX(I)=0.	108
DO 3 I=1,300	109
B2=(X(I)/BETA)**2	110
IF (B2-80.) 3,3,4	111
3 EX(I)=B1*EXP(-B2)	112
4 DO 5 I=1,100	113
DO 5 J=1,30	114
5 HLMAT(I,J)=0.	115
DO 6 I=1,900	116
6 SMZ(I)=0.	117
DO 7 I=1,30	118
A(I)=0.	119
7 B(I)=0.	120
DO 8 I=1,NB	121
AI=1	122
8 B(I)=1./((2.*AI)*(2.*AI+1.+2.*AL))	123
IF (IL) 100,100,101	124
100 LO=1	125
GO TO 102	126
101 LO=2	127

DO 9 I=1,IL	128
AI=1	129
9 A(I)=(AL*(AL+1.)-AI*(AI-1.))/(2.*AI)	130
102 FAC=2.	131
BI=2.*AL+1.	132
10 FAC=FAC/BI	133
BI=BI-2.	134
IF (BI) 11,11,10	135
11 IZ1=900	136
IZ2=2500	137
DO 14 I=1,100	138
DO 13 J=1,100	139
J1=J-1+1	140
IF (J1-50) 12,12,14	141
12 IS=I+J	142
CALL HL(I,J)	143
HLMAT(I,J1)=HLR	144
13 CONTINUE	145
14 CONTINUE	146
RETURN	147
END.	148

*TYPE(FORTRAN)	149
SUBROUTINE HL(I,J)	150
DIMENSION D(200),X(750),FR(300),FI(560),EX(300),	151
IGR(150,30),GI(150,30),IUP(150),ILO(150),JLA(150),	152
2SMZ(900),A(30),B(30)	153
COMMON D,X,FR,FI,EX,GR,GI,IUP,ILO,JLA,SMZ,	154
IA,R	155
EQUIVALENCE (D(10),BETA),(D(16),NB),(D(51),IL),	156
1(D(52),LU),(D(53),FAC),(D(54),IZ1),(D(55),IZ2),	157
2(D(58),IS),(D(59),JI),(D(60),HLR)	158
IS=IS	159
J4=JI-1	160
IZ=I*J	161
IF (IZ-IZ1) 1,7,7	162
1 IF (SMZ(IZ)) 2,2,6	163
2 Z=2.*X(1)*X(J)/(BETA**2)	164
Z2=Z**2	165
BA=1.	166
BB=1.	167
DO 4 K=1,NB	168
BB=BB*B(K)*Z2	169
IF (BA*.0001-BB) 4,4,3	170
3 BA=BA+BB	171
GO TO 5	172
4 BA=BA+BB	173
IZ1=IZ	174
GO TO 8	175
5 SMZ(IZ)=EXP(-Z)*FAC*(Z**(IL+1))*BA	176
6 IF (JI-1) 21,21,22	177
21 HLR=SMZ(IZ)/(1.7724539*BETA)	178
RETURN	179
22 HLR=EX(J4)*SMZ(IZ)	180
RETURN	181
7 IF (IZ-IZ2) 8,17,17	182
8 Z=-2.*X(1)*X(J)/(BETA**2)	183
BA=1.	184
BB=1.	185
GO TO (11,9),LU	186

9	D0 10 K=1,IL	187
	BB=BB*A(K)/Z	188
10	BA=BA+BB	189
11	IF (J1-1) 23,23,24	190
23	HLR=BA/(1.7724539*BETA)	191
	G0 T0 25	192
24	HLR=EX(J4)*BA	193
25	Z=-Z	194
	BA=1.	195
	BB=1.	196
	G0 T0 (14,12),L0	197
12	D0 13 K=1,IL	198
	BB=BB*A(K)/Z	199
13	BA=BA+BB	200
14	BA=BA*EX(15)	201
	IF (HLR*.0001-BA) 16,16,15	202
15	IZ2=IZ	203
16	IW=IL/2	204
	IF (2*IW-IL) 29,28,28	205
28	HLR=HLR-BA	206
	RETURN.	207
29	HLR=HLR+BA	208
	RETURN.	209
17	Z=-2.*X(1)*X(J)/(BETA**2)	210
	BA=1.	211
	BB=1.	212
	G0 T0 (20,18),L0	213
18	D0 19 K=1,IL	214
	BB=BB*A(K)/Z	215
19	BA=BA+BB	216
20	IF (J1-1) 26,26,27	217
26	HLR=BA/(1.7724539*BETA)	218
	RETURN	219
27	HLR=BA*EX(J4)	220
	RETURN	221
	END	222

*TYPE(FORTRAN)	223
SUBROUTINE GETHL2	224
DIMENSION F(12262),HLMAT(100,30)	225
EQUIVALENCE (F(32),BETA),(F(17),H),(F(12261),HLMAT)	226
COMMON F	227
REWIND 3	228
REWIND 1	229
READ INPUT TAPE 10,6,H,BETA	230
6 FORMAT (2F10.8)	231
DO 3 I=1,300	232
READ TAPE 3,H1,BETA1,M1	233
IF (H1-H) 2,1,2	234
1 IF (BETA1-BETA) 2,4,2	235
2 DO 3 J=1,M1	236
3 READ TAPE 3	237
4 DO 5 I6=1,M1	238
READ TAPE 3,((HLMAT(I,J),I=1,100),J=1,30)	239
5 WRITE TAPE 1,((HLMAT(I,J),I=1,100),J=1,30)	240
REWIND 1	241
RETURN	242
END	243

*TYPE(FORTRAN)	244
SUBROUTINE INPUT4	245
DIMENSION F(15265),IPA(10),DSRU(9,25),	246
IDSIU(9,25),DSRD(9,25),DSID(9,25)	247
COMMON F	248
EQUIVALENCE (F(21),E),(F(22),AMT),(F(23),AMI),	249
1(F(24),ZT),(F(25),ZI),(F(1),U),(F(2),RR),(F(3),AR),	250
2(F(4),WS),(F(5),WD),(F(6),RI),(F(7),AI),(F(8),USP),	251
3(F(9),WSP),(F(32),RE),(F(17),H),(F(18),HC),(F(55),TI),	252
4(F(26),AI),(F(27),A2),(F(28),A3),(F(19),LM),(F(56),NV),	253
5(F(20),ISP),(F(57),NDS),(F(62),IPR),(F(15241),IPA),	254
6(F(4661),DSRU),(F(4886),USIL),(F(5111),DSRD),(F(5336),	255
7DSID)	256
READ INPUT TAPE 10,1,E,AMT,AMI,ZT,ZI,U,RR,	257
1AR,WS,WD,RI,AI,USP,WSP,RE,H,HC,TI,AI,A2,A3,	258
2LM,ISP,NV,(IPA(I),I=1,9)	259
1 FORMAT (5F10.8/7F10.8/3F10.8/6F10.8,12/11,8X,	260
11,9I3)	261
NDS=NV	262
IPR=NV	263
DO 2 I=1,9	264
DO 2 J=1,25	265
DSRU(I,J)=0.	266
DSRD(I,J)=0.	267
DSIU(I,J)=0.	268
2 DSID(I,J)=0.	269
RETURN	270
END	271

*TYPE(FORTRAN)	344
SUBROUTINE SANGLE	345
DIMENSION D(6000),ANG(100),ANGEXP(100)	346
COMMON D	347
EQUIVALENCE (D(26),A1),(D(27),A2),(D(28),A3),(D(5561),ANG),	348
1(D(29),J3),(D(30),KM),(D(31),KMEXP),(D(5961),ANGEXP)	349
IF (J3-1) 1,4,4	350
1 ANG(1)=A1	351
DO 3 I=2,100	352
ANG(I)=ANG(I-1)+A2	353
IF (ANG(I)-A3) 3,2,2	354
2 KM=1	355
GO TO 6	356
3 CONTINUE	357
GO TO 6	358
4 KM=KMEXP	359
DO 5 I=1,KM	360
5 ANG(I)=ANGEXP(I)	361
6 RETURN	362
END	363

*TYPE(FORTAN)	272
SUBROUTINE PRINPU	273
DIMENSION F(150)	274
COMMON F	275
EQUIVALENCE (F(21),E),(F(1),UR),(F(4),WS),(F(2),RS),(F(3),AS),	276
1(F(5),WD),(F(6),RD),(F(7),AD),(F(8),USP),(F(9),WSP),(F(32),BETA	277
2),(F(23),AI),(F(22),AT),(F(25),ZI),(F(24),7T),(F(17),H),(F(18),HC)	278
3,(F(19),LM),(F(61),K0)	279
IF (K0-2) 1,1,3	280
1 WRITE OUTPUT TAPE 9,2	281
2 FORMAT(44HI NON-LOCAL OPTICAL POTENTIAL CALCULATION)	282
GO TO 5	283
3 WRITE OUTPUT TAPE 9,4	284
4 FORMAT(39HI SEARCH NON-LOCAL OPTICAL POTENTIAL)	285
5 WRITE OUTPUT TAPE 9,6,E,UR,USP,H,AT,WS,WD,WSP,HC	286
6 FORMAT(15H PARAMETERS/12H EXPERIMENTAL,5X,11HTHEORETICAL,	287
155X,9HNUMERICAL/12H ELAB =,F7.3,8X,7HUS =,F7.3,30X,	288
27HUSP =,F8.2,7X,4HH =,F7.3/12H MT =,F7.3,8X,7HWS =,	289
3F7.3,8X,7HWD =,F7.3,8X,7HWSP =,F8.2,7X,4HHC =,F7.3)	290
WRITE OUTPUT TAPE 9,7,ZT,RS,RD,BETA,LM,AI,AS,AD,7T	291
7 FORMAT(12H ZT =,F7.3,8X,7HRS =,F7.3,8X,7HRD =,F7.3,	292
18X,7HBETA =,F8.2,7X,4HLM =,14/12H MI =,F7.3,	293
28X,7HAS =,F7.3,8X,7HAD =,F7.3/12H ZI =,F7.3)	294
RETURN	295
END	296

*TYPE(FORTRAN)	297
SUBROUTINE DATAPR	298
DIMENSION F(150),X(200)	299
COMMON F,X	300
EQUIVALENCE (F(23),AMI),(F(22),AMT),(F(33),R),	301
1(F(11),RR),(F(2),ROR),(F(14),RI),(F(6),ROI),(F(35),AMSP),	302
2(F(36),AMSS),(F(37),ES),(F(21),ELAB),(F(39),ETA),(F(25),ZI),	303
3(F(24),ZT),(F(40),EKS),(F(41),EK),(F(10),U),(F(1),UL),	304
4(F(12),WS),(F(4),WSL),(F(13),WD),(F(5),WDL),(F(15),VSP),	305
5(F(8),VSPL),(F(16),WSP),(F(9),WSPL),(F(3),AR),(F(7),AI),	306
6(F(42),RMM),(F(17),H),(F(43),N),(F(44),NN),(F(45),LC)	307
IF (AMI-4.) 1,1,2	308
1 B=AMT**(.1/3.)	309
GO TO 3	310
2 B=AMT**(.1/3.)+AMI**(.1/3.)	311
3 RR=ROR*B	312
RI=ROI*B	313
AMSP=AMI*AMT/(AMI+AMT)	314
ES=ELAB*AMT/(AMI+AMT)	315
ETA=0.157454*ZI*ZT*SQRTF(AMSP/ES)	316
AMSS=0.04783258*AMSP	317
EKS=AMSS*ES	318
EK=SQRTF(EKS)	319
U=UL*AMSS	320
WS=WSL*AMSS	321
WD=WDL*AMSS	322
WSP=WSPL*AMSS*.011057	323
VSP=VSPL*AMSS*.011057	324
RRM=RR+7.*AR	325
RIM=RI+7.*AI	326
IF (RRM-RIM) 4,4,5	327
4 RMM=RIM	328
GO TO 6	329
5 RMM=RRM	330
6 N=RMM/H	331
IF (N-97) 8,8,7	332
7 LC=1	333
GO TO 10	334

```
8 LC=0
  AWN=N
  RMM=AWN*H
  NN=2*N
  X(1)=H
  DO 9 I=2,200
9 X(I)=X(I-1)+H
10 RETURN
  END
```

```
335
336
337
338
339
340
341
342
343
```

*TYPE(FORTRAN)	498
SUBROUTINE VECTOR	499
DIMENSION F(4000),BT(200),X(200),AT(200),FRV(200),	500
1FR(200),FRR(200),FRA(200),FI(200),FISR(200),FISA(200),	501
2VRSP(100),VSPV(100),VSPR(100),VSPA(100),WRSP(100),WSPR(100),	502
3WSPA(100),VC(200),VCR(200),FIDW(200),FIDR(200),	503
4FIDA(200),UL(200),WL(200)	504
COMMON F	505
EQUIVALENCE (F(3711),BT),(F(151),X),(F(3911),AT),	506
1(F(1711),FRV),(F(1511),FR),(F(1911),FRR),(F(2111),FRA),	507
2(F(2311),FI),(F(2511),FISR),(F(2711),FISA),(F(811),VRSP),	508
3(F(911),VSPV),(F(1011),VSPR),(F(1111),VSPA),(F(1211),WRSP),	509
4(F(1311),WSPR),(F(1411),WSPA),(F(611),VC),(F(15871),VCR),	510
5(F(2911),FIDW),(F(3111),FIDR),(F(3311),FIDA),(F(43),I7),(F(40),E),	511
6(F(43),N),(F(44),NN),(F(11),RR),(F(3),AR),(F(10),V),(F(33),B),	512
7(F(12),WS),(F(20),ISP),(F(15),VSP),(F(16),WSP),(F(39),ETA),	513
8(F(41),EK),(F(17),H),(F(13),WD),(F(14),RI),(F(7),AI),	514
9(F(32),BET),(F(15471),UL),(F(15671),WL)	515
N=N	516
NN=NN	517
B1=2.*RR	518
B2=2.*AR	519
DO 4 I=1,NN	520
BT(I)=(X(I)-B1)/B2	521
IF (BT(I)-86.) 2,1,1	522
1 AT(I)=1.0E37	523
GO TO 3	524
2 AT(I)=EXP(B1(I))	525
3 FRV(I)=1./(1.+AT(I))	526
FR(I)=FRV(I)*V	527
FRR(I)=FRV(I)*FR(I)*AT(I)*B/AR	528
FRA(I)=FRR(I)*BT(I)/B	529
FI(I)=FRV(I)*WS	530
FISR(I)=FRR(I)*WS/V	531
4 FISA(I)=FRA(I)*WS/V	532
IF (ISP-1) 18,5,5	533

18 D0 19 I=1,100	534
VSPV(I)=0.	535
VRSP(I)=0.	536
VSPR(I)=0.	537
VSPA(I)=0.	538
WRSP(I)=0.	539
WSPR(I)=0.	540
19 WSPA(I)=0.	541
G0 T0 7	542
5 D0 6 I=1,N	543
VSPV(I)=AT(2*I)*FRV(2*I)*FRV(2*I)/(AR*X(I))	544
VRSP(I)=VSPV(I)*VSP	545
VSPR(I)=VRSP(I)*B*FRV(2*I)*(AT(2*I)-1.)/AR	546
VSPA(I)=VSPR(I)*BT(2*I)/B-VRSP(I)/AR	547
WRSP(I)=VRSP(I)*WSP/VSP	548
WSPR(I)=VSPR(I)*WSP/VSP	549
6 WSPA(I)=VSPA(I)*WSP/VSP	550
7 IF (ETA) 20,20,8	551
20 D0 21 I=1,200	552
VC=0.	553
21 VCR=0.	554
G0 T0 11	555
8 B1=ETA*EK/RR	556
B2=3.*B*B1/RR	557
I1=2.*RR/H	558
D0 9 I=1,I1	559
B3=.25*(X(I)/RR)**2	560
VC(I)=B1*(3.-B3)	561
9 VCR(I)=B2*(B3-1.)	562
B1=ETA*EK*4.	563
I1=I1+1	564
N3=2*(N+3)	565
D0 10 I=11,N3	566
VC(I)=B1/X(I)	567
10 VCR(I)=0.	568
11 IF (WD) 22,22,12	569

22	D0 23 I=1,200	570
	FIDW(I)=0.	571
	FIDR(I)=0.	572
23	FIDA(I)=0.	573
	G0 T0 17	574
12	B1=2.*RI	575
	B2=2.*AI	576
	D0 16 I=1,NN	577
	BT(I)=(X(I)-B1)/B2	578
	IF (BT(I)-88.) 14,13,13	579
13	AT(I)=1.0E37	580
	G0 T0 15	581
14	AT(I)=EXP(-BT(I))	582
15	FIDW(I)=4.*AT(I)/((AT(I)+1.))**2)	583
	FI(I)=FI(I)+FIDW(I)*WD	584
	FIDR(I)=FIDW(I)*WD*B*(AT(I)-1.)/(AI *(AT(I)+1.))	585
16	FIDA(I)=FIDR(I)*BT(I)/B	586
17	IF (I7-1) 25,31,31	587
25	BE=(BET/2.))**2	588
	UL(I)=FR(I)/2.	589
	WL(I)=FI(I)/2.	590
26	EX=EXP(-BE*(E+UL(I)-VC(I)))	591
	CAS=COSF(BE*WL(I))	592
	SAN=SINF(BE*WL(I))	593
	FRN=EX*(FR(I)*CAS+FI(I)*SAN)	594
	FIN=EX*(FI(I)*CAS-FR(I)*SAN)	595
	IF (ABS((FRN-UL(I))/UL(I))-0.00005) 27,27,28	596
27	IF (ABS((FIN-WL(I))/WL(I))-0.00005) 29,29,28	597
28	UL(I)=FRN	598
	WL(I)=FIN	599
	G0 T0 26	600
29	D0 30 I=1,NN	601
	EX=EXP(-BE*(E+UL(I)-VC(I)))	602
	DI=BE*WL(I)	603
	CAS=COSF(DI)	604
	SAN=SINF(DI)	605
	DUN=FR(I+1)-FR(I)	606

DWN=FI(I+1)-FI(I)	607
ANR=EX*(DUN*CAS+DWN*SAN)+UL(I)*BE*(VC(I+1)-VC(I))	608
ANI=EX*(DWN*CAS-DUN*SAN)+WL(I)*BE*(VC(I+1)-VC(I))	609
DR=1.+BE*UL(I)	610
D2=DR**2+DI**2	611
DUL=(DR*ANR+DI*ANI)/D2	612
DWL=(DR*ANI-DI*ANR)/D2	613
UL(I+1)=UL(I)+DUL	614
30 WL(I+1)=WL(I)+DWL	615
31 RETURN	616
END	617

*TYPE(FORTRAN)	364
SUBROUTINE COULOM	365
DIMENSION AB(350),SIGMA(51),F(56),FP(51),	366
IG(51),GP(51)	367
COMMON AB,SIGMA,F,FP,G,GP	368
EQUIVALENCE (AB(19),LMAX),(AB(18),H),	369
1(AB(39),HETA),(AB(40),AKS),(AB(41),AK),	370
2(AB(45),J),(AB(42),RM),(AB(46),RA)	371
J=0	372
23 HETAS=HETA**2	373
RA=RM	374
N=0	375
AL=50.	376
L=51	377
CI=AL+1.	378
PHI=SQRTF(CI**2+HETAS)	379
CHI=ATANF(HETA/CI)	380
SIGMA(L)=(AL+.5)*CHI+HETA*LOGF(PHI)-HETA-SINF(CHI)/(12.*PHI)+	381
1SINF(3.*CHI)/(360.*PHI**3.)-SINF(5.*CHI)/(1260.*PHI**5.)+	382
2SINF(7.*CHI)/(1680.*PHI**7.)-SINF(9.*CHI)/(1188.*PHI**9.)	383
SIGMA(L-1)=SIGMA(L)-ATANF(HETA/AL)	384
AL=AL-1.	385
L=L-1	386
IF(L-2)2,1,1	387
2 I=0	388
R=RA+H	389
9 CI=AK*R	390
C2=2.*CI	391
SS=1.	392
SI=SS	393
TS=0.	394
TI=TS	395
SL=0.	396
S2=SL	397
TL=1.-HETA/CI	398

T2=TL	399
AN=0.	400
NI=20	401
3 C3=AN+1.	402
C4=C3*C2	403
S4=S1	404
C5=(2.*AN+1.)*HETA/C4	405
C6=(HETAS-AN*C3)/C4	406
S1=C5*S1-C6*T1	407
T1=C5*T1+C6*S4	408
S3=S2	409
S2=C5*S2-C6*T2-S1/C1	410
T2=C5*T2+C6*S3-11/C1	411
AN=C3	412
NI=NI-1	413
SS=SS+S1	414
TS=TS+T1	415
SL=SL+S2	416
TL=TL+T2	417
IF(NI)4,4,3	418
4 W=ABSF(SS*TL-SL*TS-1.)	419
IF(W-.0001)6,6,5	420
5 RA=RA+20.*H	421
N=N+10	422
GO TO 2	423
6 THETA=C1-HETA*LOGF(C2)+SIGMA(1)	424
C7=COSF(THETA)	425
C8=SINF(THETA)	426
G(1)=SS*C7-TS*C8	427
GP(1)=AK*(SL*C7-TL*C8)	428
IF(1)8,7,8	429
7 G1=G(1)	430
I=1	431
R=RA	432
G3=G1	433
GO TO 9	434

8	IF(RM-RA)10,11,10	435
10	C3=2.*HEIA*AK	436
	C4=10.*H**2/12.	437
	C5=C4/10.	438
	G7=G(1)	439
	M=N	440
	R=RA+H	441
	AK1=AKS-C3/R	442
	R=RA	443
	AK2=AKS-C3/R	444
	AK4=AK2*G7	445
14	IF(M)13,13,12	446
12	R=R-H	447
	AK3=AKS-C3/R	448
	G2=((2.-C4*AK2)*G7-(1.+C5*AK1)*G1)/(1.+C5*AK3)	449
	R=R-H	450
	AK1=AKS-C3/R	451
	G1=((2.-C4*AK3)*G2-(1.+C5*AK2)*G7)/(1.+C5*AK1)	452
	AK4=AK4+4.*G2*AK3+2.*G1*AK1	453
	G7=G1	454
	AK2=AK1	455
	G1=G2	456
	AK1=AK3	457
	M=M-1	458
	G0 T0 14	459
13	G(1)=G7	460
	GP(1)=H*(AK4-G7*AK2)/3.+GP(1)	461
11	I=IMAX+4	462
	C5=AK*RM	463
	F(L+2)=0.	464
	F(L+1)=10.0E-36	465
	AL=L	466
15	C3=AL*2.+1.	467
	C4=AL+1.	468
	F(L)=AL/SQRTF(AL**2+HETAS)*((C3*HETA/	469
	1(AL*C4)+C3/C5)*F(L+1)-SQRTF(C4**2+HETAS)/	470

2C4*F(L+2))	471
AL=AL-1.	472
L=L-1	473
IF(L)16,16,15	474
16 FP(1)=AK*((HETA+1./C5)*F(1)-SQRTF(HETAS+1.)*F(2))	475
ALPHA=AK/(FP(1)*G(1)-F(1)*GP(1))	476
FP(1)=ALPHA*FP(1)	477
L1=LMAX+1	478
D0 17 L=1,L1	479
17 F(L)=ALPHA*F(L)	480
AL=0.	481
D0 18 L=1,L1	482
AL=AL+1.	483
C3=HETA/AL+AL/C5	484
C4=SQRTF(AL**2+HETAS)/AL	485
FP(L+1)=AK*(C4*F(L)-C3*F(L+1))	486
G(L+1)=C3*G(L)/C4-GP(L)/(C4*AK)	487
GP(L+1)=AK*(C4*G(L)-C3*G(L+1))	488
W=ABSF((FP(L)*G(L)-F(L)*GP(L))/AK-1.)	489
IF(W-.0001)18,18,20	490
18 CONTINUE	491
G0 TO 21	492
20 G1=G3	493
J=J+1	494
IF(J-2)6,21,21	495
21 RETURN	496
END	497

*TYPE(FORTRAN)	618
SUBROUTINE NEWMAT	619
DIMENSION F(100),HLMAT(100,30),FI(200),	620
IGR(100,30),GI(100,30),IUP(100),ILO(100),FR(200)	621
COMMON F	622
EQUIVALENCE (F(43),N),(F(44),NN),(F(49),AL),	623
1(F(47),IL),(F(45),IC),(F(12261),HLMAT),(F(4261),GR),	624
2(F(9261),GI),(F(3511),IUP),(F(3611),ILO),(F(1511),FR),	625
3(F(2311),FI)	626
NN=NN	627
N=N	628
AL=1L	629
IF (AL) 1,1,2	630
1 REWIND 1	631
LO=1	632
GO TO 26	633
2 LO=2	634
26 READ TAPE 1,((HLMAT(I,J),I=1,100),J=1,30)	635
3 DO 4 I=1,100	636
DO 4 J=1,30	637
GR(I,J)=0.	638
4 GI(I,J)=0.	639
DO 5 I=1,100	640
IUP(I)=0	641
5 ILO(I)=0	642
GO TO (6,7),LO	643
6 GRM=FR(NN)*HLMAT(N,1)	644
GIM=FI(NN)*HLMAT(N,1)	645
7 DO 8 I=1,10	646
DO 8 J=1,10	647
J1=J-1+1	648
IS=J+1	649
GR(I,J1)=FR(IS)*HLMAT(I,J1)	650
8 GI(I,J1)=FI(IS)*HLMAT(I,J1)	651
DO 14 I=1,11	652
DO 13 J=11,N	653
J1=J-1+1	654
IF (J1-30) 10,10,9	655

9	LC=JI	656
	GO TO 25	657
10	IS=J+1	658
	GR(I,JI)=FR(IS)*HLMAT(I,JI)	659
	GI(I,JI)=FI(IS)*HLMAT(I,JI)	660
	IF (GR(I,JI)-GRM) 11,13,13	661
11	IF (GI(I,JI)-GIM) 12,13,13	662
12	IUP(I)=J	663
	GO TO 14	664
13	CONTINUE	665
14	CONTINUE	666
	DO 20 I=12,N	667
	DO 19 J=I,N	668
	JI=J-I+1	669
	IF (JI-30) 16,16,15	670
15	LC=JI	671
	GO TO 25	672
16	IS=I+J	673
	GR(I,JI)=FR(IS)*HLMAT(I,JI)	674
	GI(I,JI)=FI(IS)*HLMAT(I,JI)	675
	IF (GR(I,JI)-GRM) 17,19,19	676
17	IF (GI(I,JI)-GIM) 18,19,19	677
18	IUP(I)=J	678
	GO TO 20	679
19	CONTINUE	680
20	CONTINUE	681
	K0=IUP(I)	682
	DO 21 I=1,K0	683
21	IL0(I)=1	684
	DO 24 I=2,N	685
	K1=IUP(I)-IUP(I-1)	686
	IF (K1) 24,24,22	687
22	K2=K0+1	688
	K3=K0+K1	689
	DO 23 J=K2,K3	690
23	IL0(J)=1	691

K0=K3
24 CONTINUE
LC=0
25 RETURN
END

692
693
694
695
696

*TYPE(FORTRAN)	697
SUBROUTINE INTEGR	698
DIMENSION F(9265),UR(100),UI(100),FRH(100),	699
FIH(100),PR(100),PI(100),QR0(100),QI0(100),IL0(100),	700
2IUP(100),GR(100,30),GI(100,30),X(200),VC(100),IU(25),	701
3ID(25),VRSP(100),WRSP(100),	702
4FRHL(100),FIHL(100),UL(200),WL(200)	703
COMMON F	704
EQUIVALENCE (F(48),ILI),(F(49),AL),(F(47),IL),	705
1(F(43),N),(F(50),NI),(F(17),H),(F(51),KSP),(F(40),EK2),	706
2(F(52),DERR),(F(53),DERI),(F(54),DERN),(F(55),TIOF),	707
3(F(45),LC),(F(4311),UR),(F(4411),UI),(F(3711),FRH),	708
4(F(3811),FIH),(F(3911),PR),(F(4011),PI),(F(4111),QR0),	709
5(F(4211),QI0),(F(3611),IL0),(F(6261),GR),(F(9261),GI),	710
6(F(3511),IUP),(F(151),X),(F(611),VC),(F(811),VRSP),	711
7(F(1211),WRSP),(F(4511),IU),(F(4536),ID)	712
8,(F(15271),FRHL),(F(15371),FIHL),(F(15471),UL),	713
9(F(15671),WL),(F(63),I7)	714
ILI=IL+1	715
AL=IL	716
N=N	717
NI=0	718
NI=N+1	719
N2=N+2	720
N3=N+3	721
N4=N-1	722
N5=N-2	723
N6=N-3	724
H2=H**2	725
H3=H2*H	726
IF (KSP) 1,1,3	727
1 DO 2 I=1,100	728
UR(I)=0.	729
2 UI(I)=0.	730
3 DO 4 I=1,100	731
FRH(I)=0.	732

FIH(I)=0.	733
PR(I)=n.	734
PI(I)=0.	735
QR0(I)=0.	736
4 QI0(I)=0.	737
IF (I7-I) 41,40,40	738
40 D0 7 I=1,N	739
KI=IL0(I)	740
D0 5 J=KI,I	741
J1=1-J+1	742
QR0(I)=QR0(I)+GR(J,J1)	743
5 QI0(I)=QI0(I)+GI(J,J1)	744
K2=IUP(I)	745
K1=I+1	746
D0 6 J=KI,K2	747
J1=J-1+1	748
QR0(I)=QR0(I)+GR(I,J1)	749
6 QI0(I)=QI0(I)+GI(I,J1)	750
QR0(I)=QR0(I)*H3	751
7 QI0(I)=QI0(I)*H3	752
G0 T0 43	753
41 D0 42 I=1,N	754
QR0(I)=UL(2*I)*H2	755
42 QI0(I)=WL(2*I)*H2	756
43 IF (KSP) 8,8,11	757
8 D0 9 I=1,N	758
FRH(I)=QR0(I)+H2*(EK2-AL*(AL+1.)/(X(I)**2)-VC(2*I)	759
I+AL*VRSP(I))	760
9 FIH(I)=QI0(I)+H2*AL*WRSP(I)	761
D0 10 I=N1,N3	762
10 FRH(I)=H2*(EK2-AL*(AL+1.)/(X(I)**2)-VC(2*I))	763
G0 T0 14	764
11 D0 12 I=1,N	765
FRH(I)=QR0(I)+H2*(EK2-AL*(AL+1.)/(X(I)**2)-VC(2*I)	766
I-(AL+1.)*VRSP(I))	767
12 FIH(I)=QI0(I)-H2*(AL+1.)*WRSP(I)	768
D0 13 I=N1,N3	769

13	FRH(I)=H2*(EK2-AL*(AL+1.)/(X(I)**2)-VC(2*I))	770
	IF (17-I) 14,19,19	771
14	UR(I)=H*(IL+1)	772
	UR(2)=(2.-FRH(I))*UR(I)	773
	UI(2)=-FIH(I)*UR(I)	774
	DO 15 I=2,4	775
	BLA=2.-FRH(I)	776
	UR(I+1)=BLA*UR(I)+FIH(I)*UI(I)-UR(I-1)	777
15	UI(I+1)=BLA*UI(I)-FIH(I)*UR(I)-UI(I-1)	778
	DO 18 I=5,N2	779
	B3R=12.+FRH(I+1)	780
	B1R=24.-10.*FRH(I)	781
	B1I=-10.*FIH(I)	782
	B2R=12.+FRH(I-1)	783
	CR=B1R*UR(I)-B1I*UI(I)-B2R*UR(I-1)+FIH(I-1)*UI(I-1)	784
	CI=B1I*UR(I)+B1R*UI(I)-FIH(I-1)*UR(I-1)-B2R*UI(I-1)	785
	B3N=B3R**2+FIH(I+1)**2	786
	UR(I+1)=(CR*B3R+CI*FIH(I+1))/B3N	787
18	UI(I+1)=(CI*B3R-CR*FIH(I+1))/B3N	788
19	DERNM=1000000.	789
20	URP=(UR(N3)-UR(N6)+9.*(UR(N5)-UR(N2))+45.*(UR(N1)-UR(N4)))/(60.*H)	790
	UIP=(UI(N3)-UI(N6)+9.*(UI(N5)-UI(N2))+45.*(UI(N1)-UI(N4)))/(60.*H)	791
	U2=UR(N)**2+UI(N)**2	792
	DERR=(URP*UR(N)+UIP*UI(N))/U2	793
	DERI=(UIP*UR(N)-URP*UI(N))/U2	794
	DERN=DERR**2+DERI**2	795
	IF (ARSF((DERN-DERNM)/DERN)-T10F) 30,21,21	796
21	IF (NI-25) 22,22,31	797
22	NI=NI+1	798
	DERNM=DERN	799
	DO 25 I=1,N	800
	PR(I)=0.	801
	PI(I)=0.	802
	KI=IL0(I)	803
	K3=I+1	804
		805
		806

	D0 23 J=K1,I	807
	J1=K3-J	808
	PR(I)=PR(I)+GR(J,J1)*UR(J)-GI(J,J1)*UI(J)	809
23	PI(I)=PI(I)+GI(J,J1)*UR(J)+GR(J,J1)*UI(J)	810
	K2=1UP(I)	811
	K1=I-I	812
	D0 24 J=K3,K2	813
	J1=J+K1	814
	PR(I)=PR(I)+GR(I,J1)*UR(J)-GI(I,J1)*UI(J)	815
24	PI(I)=PI(I)+GI(I,J1)*UR(J)+GR(I,J1)*UI(J)	816
	PR(I)=H3*PR(I)-QR0(I)*UR(I)+QI0(I)*UI(I)	817
25	PI(I)=H3*PI(I)-QI0(I)*UR(I)-QR0(I)*UI(I)	818
	UR(2)=(2.-FRH(I))*UR(I)+PR(I)	819
	UI(2)=-FIH(I)*UR(I)+PI(I)	820
	D0 26 I=2,4	821
	BIA=2.-FRH(I)	822
	UR(I+1)=BIA*UR(I)+FIH(I)*UI(I)-UR(I-1)+PR(I)	823
26	UI(I+1)=BIA*UI(I)-FIH(I)*UR(I)-UI(I-1)+PI(I)	824
	D0 29 I=5,N2	825
	B3R=12.+FRH(I+1)	826
	BIR=24.-10.*FRH(I)	827
	BII=-10.*FIH(I)	828
	B2R=12.+FRH(I-1)	829
	CR=BIR*UR(I)-BII*UI(I)-B2R*UR(I-1)+FIH(I-1)	830
	I*UI(I-1)-PR(I+1)-10.*PR(I)-PR(I-1)	831
	CI=BII*UR(I)+BIR*UI(I)-FIH(I-1)*UR(I-1)-B2R	832
	I*UI(I-1)-PI(I+1)-10.*PI(I)-PI(I-1)	833
	B3N=B3R**2+FIH(I+1)**2	834
	UR(I+1)=(CR*B3R+CI*FIH(I+1))/B3N	835
29	UI(I+1)=(CI*B3R-CR*FIH(I+1))/B3N	836
	G0 T0 20	837
30	LC=0	838
	IF (KSP) 32,32,33	839
32	IU(ILI)=NI	840
	ID(ILI)=0	841
	G0 T0 34	842

```
33 ID(ILI)=NI  
34 RETURN  
31 LC=1  
GO TO 34  
END
```

843
844
845
846
847

*TYPE(FORTRAN)	848
SUBROUTINE SCMATR	849
DIMENSION F(4700),FC(56),FCP(51),GC(51),	850
IGCP(51),SRU(25),SRD(25),SIU(25),SID(25)	851
COMMON F	852
EQUIVALENCE (F(48),ILI),(F(51),KSP),(F(52),DERR),	853
1(F(53),DERI),(F(402),FC),(F(458),FCP),(F(519),GC),	854
2(F(560),GCP),(F(4561),SRU),(F(4586),SIU),	855
3(F(4611),SRD),(F(4636),SID)	856
ILI=ILI	857
SNR=FCP(ILI)-FC(ILI)*DERR+GC(ILI)*DERI	858
SNI=GCP(ILI)-FC(ILI)*DERI-GC(ILI)*DERR	859
SDR=FC(ILI)*DERR+GC(ILI)*DERI-FCP(ILI)	860
SDI=GCP(ILI)+FC(ILI)*DERI-GC(ILI)*DERR	861
SNN=SDR**2+SDI**2	862
SRR=(SNR*SDR+SNI*SDI)/SNN	863
SRI=(SNI*SDR-SNR*SDI)/SNN	864
IF (KSP) 1,1,2	865
1 SRU(ILI)=SRR	866
SIU(ILI)=SRI	867
GO TO 3	868
2 SRD(ILI)=SRR	869
SID(ILI)=SRI	870
3 RETURN	871
END	872

*TYPE(FORTRAN)	916
SUBROUTINE DERIVS	917
DIMENSION F(12265),SRU(25),SIU(25),SRD(25),	918
ISID(25),FC(56),GC(51),UR(100),UI(100),IPA(10),	919
2FRH(200), QR0(100),Q10(100),IL0(100),	920
3HLMAT(100,30),IUP(100),FRR(200),PR(200),FIDW(200),	921
4FISR(200),FRA(200),FISA(200),FRV(200),FIDR(200),	922
5FIDA(200),VCR(200),VSPR(100),WSPR(100),VSPA(100),	923
6WSPA(100),VSPV(100),DSRU(9,25),DSIU(9,25),DSRD(9,25),	924
7DSID(9,25)	925
COMMON F	926
EQUIVALENCE (F(44),NN),(F(43),N),(F(48),ILI),(F(51),KSP),(F(17),H	927
1),(F(41),EK),(F(20),ISP),(F(49),AL),(F(56),NV), (F(101	928
21),VSPR),(F(1311),WSPR),(F(1111),VSPA),(F(1411),WSPA),(F(911),VSPV	929
3),(F(4661),DSRU),(F(4886),DSIU),(F(5111),DSRD),(F(5336),DSID),(F(4	930
4561),SRU),(F(4586),SIU),(F(4511),SRD),(F(4636),SID),(F(402),FC),(F	931
5(509),GC),(F(4311),UR),(F(4411),UI),(F(15261),IPA),(F(3711),FRH),(	932
6F(1711),FRV),(F(4111),QR0),(F(4211),Q10),(F(3611),IL0),(F(12261),H	933
7LMAT),(F(3511),IUP),(F(1911),FRR),(F(3911),PR),(F(2511),FISR),(F(2	934
8111),FRA),(F(2711),FISA),(F(2911),FIDW),(F(3111),FIDR),(F(3311),FI	935
9DA),(F(57),NDS),(F(15871),VCR)	936
IF (NDS) 51,51,52	937
52 NN=NN	938
NN=NN	939
N=N	940
ILI=ILI	941
IF (KSP) 100,100,101	942
100 SOUR=SRU(ILI)	943
S0UI=SIU(ILI)	944
G0 T0 102	945
101 SOUR=SRD(ILI)	946
S0UI=SID(ILI)	947
102 TAR=FC(ILI)*(1.+S0UR)+S0UI*GC(ILI)	948
TAI=GC(ILI)*(1.-S0UR)+S0UI*FC(ILI)	949
URN=UR(N)**2+UI(N)**2	950
CONR=(TAR*UR(N)+TAI*UI(N))/URN	951
CONI=(TAI*UR(N)-TAR*UI(N))/URN	952

TAR=(-CONR*CONI*H)/EK	953
TAI=(CONR**2-CONI**2)*H/(2.*EK)	954
IF (ISP) 103,103,104	955
103 TU=0.	956
GO TO 107	957
104 IF (KSP) 105,105,106	958
105 TU=AL	959
GO TO 107	960
106 TU=-(AL+1.)	961
107 DO 50 K=1,NV	962
KID=IPA(K)	963
IF (KID-1) 1,1,6	964
1 DO 2 J=1,NN	965
2 FRH(J)=FRV(J)*.04783258	966
DO 5 I=1,N	967
QR0(I)=0.	968
Q10(I)=0.	969
K1=1L0(I)	970
K3=I+1	971
DO 3 J=K1,I	972
J1=K3-J	973
K4=1+J	974
TUTR=FRH(K4)*HLMAT(J,J1)	975
QR0(I)=QR0(I)+TUTR*UR(J)	976
3 Q10(I)=Q10(I)+TUTR*UI(J)	977
K2=IUP(I)	978
K1=I-I	979
DO 4 J=K3,K2	980
J1=J+K1	981
K4=J+1	982
TUTR=FRH(K4)*HLMAT(I,J1)	983
QR0(I)=QR0(I)+TUTR*UR(J)	984
4 Q10(I)=Q10(I)+TUTR*UI(J)	985
QR0(I)=QR0(I)*H	986
5 Q10(I)=Q10(I)*H	987
GO TO 34	988

6 IF (K10-2) 7,7,9	989
7 D0 8 J=1,NN	990
FRH(J)=FRR(J)	991
8 PR(J)=FISR(J)	992
G0 T0 12	993
9 IF (K10-3) 10,10,16	994
10 D0 11 J=1,NN	995
FRH(J)=FRA(J)	996
11 PR(J)=FISA(J)	997
12 D0 15 I=1,N	998
QR0(I)=0.	999
QI0(I)=0.	1000
K1=IL0(I)	1001
K3=I+1	1002
D0 13 J=K1,1	1003
J1=K3-J	1004
K4=I+J	1005
TUTR=FRH(K4)*HLMAT(J,J1)	1006
TUTI=PR(K4)*HLMAT(J,J1)	1007
QR0(I)=QR0(I)+TUTR*UR(J)-TUTI*UI(J)	1008
13 QI0(I)=QI0(I)+TUTR*UI(J)+TUTI*UR(J)	1009
K2=IUP(I)	1010
K1=I-I	1011
D0 14 J=K3,K2	1012
J1=J+K1	1013
K4=J+1	1014
TUTR=FRH(K4)*HLMAT(I,J1)	1015
TUTI=PR(K4)*HLMAT(I,J1)	1016
QR0(I)=QR0(I)+TUTR*UR(J)-TUTI*UI(J)	1017
14 QI0(I)=QI0(I)+TUTR*UI(J)+TUTI*UR(J)	1018
QR0(I)=QR0(I)*H	1019
15 QI0(I)=QI0(I)*H	1020
G0 T0 34	1021
16 IF (K10-4) 17,17,19	1022
17 D0 18 J=1,NN	1023
18 PR(J)=FRV(J)*.04783258	1024
G0 T0 28	1025

19 IF (K10-5) 20,20,22	1026
20 DO 21 J=1,NN	1027
21 PR(J)=FIDW(J)*.04783258	1028
GO TO 28	1029
22 IF (K10-6) 23,23,25	1030
23 DO 24 J=1,NN	1031
24 PR(J)=FIDR(J)	1032
GO TO 28	1033
25 IF (K10-7) 26,26,32	1034
26 DO 27 J=1,NN	1035
27 PR(J)=FIDA(J)	1036
28 DO 31 I=1,N	1037
QR0(I)=0.	1038
QI0(I)=0.	1039
K1=ILO(I)	1040
K3=I+1	1041
DO 29 J=K1,1	1042
J1=K3-J	1043
K4=1+J	1044
TUTI=PR(K4)*HLMAT(J,J1)	1045
QR0(I)=QR0(I)-UI(J)*TUTI	1046
29 Q19(I)=QI0(I)+UR(J)*TUTI	1047
K2=IUP(I)	1048
K1=I-I	1049
DO 30 J=K3,K2	1050
J1=J+K1	1051
K4=1+J	1052
TUTI=PR(K4)*HLMAT(I,J1)	1053
QR0(I)=QR0(I)-TUTI*UI(J)	1054
30 QI0(I)=Q19(I)+TUTI*UR(J)	1055
QR0(I)=QR0(I)*H	1056
31 QI0(I)=Q19(I)*H	1057
GO TO 34	1058
32 DO 33 J=1,N	1059
QR0(J)=0.	1060
33 QI0(J)=0.	1061

34 IF (K10-1) 46,46,35	1062
35 IF (K10-2) 36,36,38	1063
36 DO 37 I=1,N	1064
TUTR=-VCR(2*I)+TU*VSPR(I)	1065
TUTI=TU*WSPR(I)	1066
QR0(I)=QR0(I)+TUTR*UR(I)-TUTI*UI(I)	1067
37 QI0(I)=QI0(I)+TUTR*UI(I)+TUTI*UR(I)	1068
GO TO 46	1069
38 IF (K10-3) 39,39,41	1070
39 DO 40 J=1,N	1071
TUTR=TU*VSPA(I)	1072
TUTI=TU*WSPA(I)	1073
QR0(I)=QR0(I)+TUTR*UR(I)-TUTI*UI(I)	1074
40 QI0(I)=QI0(I)+TUTR*UI(I)+TUTI*UR(I)	1075
GO TO 46	1076
41 IF (K10-8) 46,42,44	1077
42 DO 43 I=1,N	1078
QR0(I)=TU*VSPV(I)*UR(I)*.04783258*.011057	1079
43 QI0(I)=TU*VSPV(I)*UI(I)*.04783258*.011057	1080
GO TO 46	1081
44 DO 45 I=1,N	1082
QR0(I)=-TU*VSPV(I)*UI(I)*.04783258*.011057	1083
45 QI0(I)=TU*VSPV(I)*UR(I)*.04783258*.011057	1084
46 TUTR=0.	1085
TUTI=0.	1086
DO 47 I=1,N	1087
TUTR=TUTR+QR0(I)*UR(I)-QI0(I)*UI(I)	1088
47 TUTI=TUTI+QR0(I)*UI(I)+QI0(I)*UR(I)	1089
IF (KSP) 48,48,49	1090
48 DSRU(K10,ILI)=TAR*TUTR-TAI*TUTI	1091
DSIU(K10,ILI)=TAR*TUTI+TAI*TUTR	1092
GO TO 50	1093
49 DSRD(K10,ILI)=TAR*TUTR-TAI*TUTI	1094
DSID(K10,ILI)=TAR*TUTI+TAI*TUTR	1095
50 CONTINUE	1096
51 RETURN	1097
END	1098

*TYPE(FORTRAN)	873
SUBROUTINE CLEAN	874
DIMENSION SRD(25),SRU(25),SIU(25),SID(25),	875
1DSRD(9,25),DSID(9,25),DSRU(9,25),DSIU(9,25),	876
2F(5400)	877
COMMON F	878
EQUIVALENCE (F(48),ILI),(F(20),ISP),(F(51),KSP),(F(47),IL),	879
1(F(45),LC),(F(19),ILM),(F(4611),SRD),(F(4541),SRU),	880
2(F(4586),SIU),(F(4636),SID),(F(5111),DSRD),(F(5336),DSID),	881
3(F(4661),DSRU),(F(4886),DSIU)	882
ILI=ILI	883
IF (ISP) 1,1,3	884
1 SRD(ILI)=SRU(ILI)	885
SID(ILI)=SIU(ILI)	886
DO 2 I=1,9	887
DSRD(I,ILI)=DSRU(I,ILI)	888
2 DSID(I,ILI)=DSIU(I,ILI)	889
GO TO 9	890
3 IF (ILI-1) 4,4,6	891
4 SRD(1)=SRU(1)	892
SID(1)=SIU(1)	893
DO 5 I=1,9	894
DSRD(I,1)=DSRU(I,1)	895
5 DSID(I,1)=DSIU(I,1)	896
GO TO 14	897
6 IF (KSP) 7,7,8	898
7 KSP=1	899
LC=0	900
RETURN	901
8 KSP=0	902
9 IF (ABSF(SRD(ILI))-0.999) 14,10,10	903
10 IF (ABSF(SRU(ILI))-0.999) 14,11,11	904
11 IF (ABSF(SID(ILI))-0.001) 12,12,14	905
12 IF (ABSF(SIU(ILI))-0.001) 13,13,14	906
13 LC=1	907
RETURN	908

```
14 IL=IL+1
   IF (IL-ILM) 16,16,15
15 LC=2
   RETURN
16 LC=-1
   RETURN
   END
```

```
909
910
911
912
913
914
915
```

*TYPE(FORTRAN)	1099
SUBROUTINE SIGMAS	1100
DIMENSION F(5900),CAS(25),SAN(25),PL(25),PLM(25),	1101
1SIGC(25),SRU(25),SIU(25),SRD(25),SID(25),ANG(100),	1102
2SIG(100),P0(100),SIGC00(100)	1103
COMMON F	1104
EQUIVALENCE (F(30),KM),(F(34),ILM),(F(47),IL),(F(58),SIGA),	1105
1(F(40),AKS),(F(39),ETA),(F(59),SIGE),(F(60),SIGT),(F(41),AK),	1106
2(F(351),SIGC),(F(4561),SRU),(F(4586),SIU),(F(4611),SRD),	1107
3(F(4636),SID),(F(5561),ANG),(F(5661),SIG),(F(5761),P0),	1108
4(F(5861),SIGC00)	1109
KM=KM	1110
ILM=IL+1	1111
B1=1.	1112
B2=0.	1113
SIGA=0.	1114
DO 1 I=1,ILM	1115
SIGA=SIGA+B1*(1.-SRU(I)**2-SIU(I)**2)+B2*(1.-	1116
1SRD(I)**2-SID(I)**2)	1117
B1=B1+1.	1118
1 B2=B2+1.	1119
SIGA=SIGA*31.415927/AKS	1120
IF (ETA) 2,2,5	1121
2 B1=1.	1122
B2=0.	1123
SIGE=0.	1124
DO 3 I=1,ILM	1125
SIGE=SIGE+B1*((1.-SRU(I))**2+SIU(I)**2)+	1126
1B2*((1.-SRD(I))**2+SID(I)**2)	1127
B1=B1+1.	1128
3 B2=B2+1.	1129
SIGE=SIGE*31.415927/AKS	1130
B1=1.	1131
B2=0.	1132
SIGT=0.	1133
DO 4 I=1,ILM	1134

SIGT=SIGT+B1*(1.-SRU(1))+B2*(1.-SRD(1))	1135
B1=B1+1.	1136
4 B2=B2+1.	1137
SIGT=SIGT*62.831854/AKS	1138
5 D0 6 I=1,ILM	1139
CAS(1)=COSF(2.*SIGC(1))	1140
6 SAN(1)=SINF(2.*SIGC(1))	1141
B1=2.*AK	1142
B2=2.*SIGC(1)	1143
B3=2.*ETA	1144
D0 12 I=1,KM	1145
PL(1)=1.	1146
PL(2)=COSF(ANG(1)*0.0174532925)	1147
PLM(1)=0.	1148
PLM(2)=SINF(ANG(1)*0.0174532925)	1149
C1=3.	1150
C2=1.	1151
C3=2.	1152
D0 7 J=3,ILM	1153
PL(J)=(C1*PL(2)*PL(J-1)-C2*PL(J-2))/C3	1154
PLM(J)=PL(2)*PLM(J-1)+C3*PLM(2)*PL(J-1)	1155
C1=C1+2.	1156
C2=C2+1.	1157
7 C3=C3+1.	1158
C1=0.	1159
SUMAR=0.	1160
SUMAI=0.	1161
SUMBR=0.	1162
SUMBI=0.	1163
D0 8 J=1,ILM	1164
SRM=(C1+1.)*SRU(J)+C1*SRD(J)-2.*C1-1.	1165
SIM=(C1+1.)*SIU(J)+C1*SID(J)	1166
SUMAR=SUMAR+(SRM*SAN(J)+SIM*CAS(J))*PL(J)	1167
SUMAI=SUMAI-(SRM*CAS(J)-SIM*SAN(J))*PL(J)	1168
SRM=SRU(J)-SRD(J)	1169
SIM=SIU(J)-SID(J)	1170

SUMBR=SUMBR+(SRM*SAN(J)+SIM*CAS(J))*PLM(J)	1171
SUMBI=SUMBI-(SRM*CAS(J)-SIM*SAN(J))*PLM(J)	1172
8 CI=CI+1.	1173
FCR=0.	1174
FCI=0.	1175
IF (ETA) 9,10,9	1176
9 CI=SINF(ANG(1)*0.0087266463)	1177
C2=B2-B3*LOGF(CI)	1178
FCR=-ETA*CMSF(C2)/(CI**2)	1179
FCI=-ETA*SINF(C2)/(CI**2)	1180
10 SUMAR=SUMAR+FCR	1181
SUMAI=SUMAI+FCI	1182
SIG(I)=(SUMAR**2+SUMAI**2+SUMBR**2+SUMBI**2)*10.	1183
1/(BI**2)	1184
P0(1)=20.*(SUMAI*SUMBR-SUMBI*SUMAR)/(SIG(I)*	1185
1(BI**2))	1186
IF (ETA) 11,12,11	1187
11 SIGCOU(I)=SIG(I)*(BI**2)*.1/(FCR**2+FCI**2)	1188
12 CONTINUE	1189
RETURN	1190
END	1191

*TYPE(FORTRAN)	1192
SUBROUTINE PRSIGM	1193
DIMENSION F(5665),ANG(100),SIG(100),PB(100),ID(25),IL(25),	1194
1SIGC(100),SRU(25),SIU(25),SRD(25),SID(25),DSRU(9,25),	1195
2DSIU(9,25),DSRD(9,25),DSID(9,25)	1196
COMMON F	1197
EQUIVALENCE (F(39),ETA),(F(60),SIGT),(F(59),SIGE),(F(58),SIGA),	1198
1(F(30),KM),(F(34),LM),(F(62),IPR),(F(5561),ANG),(F(5661),SIG),	1199
2(F(5761),PB),(F(5861),SIGC),(F(4561),SRU),(F(4586),SIU),	1200
3(F(4611),SRD),(F(4636),SID),(F(4661),DSRU),(F(4886),DSIU),	1201
4(F(5111),DSRD),(F(5336),DSID),(F(4511),IU),(F(4536),ID)	1202
IF (ETA) 1,1,15	1203
1 DO 2 I=1,KM	1204
2 SIGC(I)=0.	1205
GO TO 3	1206
15 SIGT=0.	1207
SIGE=0.	1208
3 WRITE OUTPUT TAPE 9,4,SIGA,SIGE,SIGT	1209
4 FORMAT(12H0 RESULTS/19H0 SIG.REACTION =,	1210
10PE14.6,5X,13HSIG.ELASTIC =,0PE14.6,5X,	1211
211HSIG.TOTAL =,0PE14.6/85X,9HS MATRIX/	1212
36X,5HTHETA,8X,5HSIGE.,10X,4HP0L.,9X,8HSIG/SIGC,	1213
411X,1HL,4X,3HSR+,6X,3HSI+,4X,2HIT,6X,3HSR-,6X,	1214
53HSI-,4X,2HIT)	1215
IX=0	1216
DO 5 I=1,LM	1217
WRITE OUTPUT TAPE 9,6,ANG(I),SIG(I),PB(I),SIGC(I),	1218
1IX,SRU(I),SIU(I),ID(I),SRD(I),SID(I),ID(I)	1219
5 IX=IX+1	1220
6 FORMAT(5X,0PF6.2,4X,0PE12.4,4X,0PE12.4,4X,0PE12.4,	1221
19X,12,2X,0PF7.4,2X,0PF7.4,2X,12,4X,0PF7.4,2X,0PF7.4,	1222
22X,12)	1223
IX=IX+1	1224
DO 7 I=IX,KM	1225
7 WRITE OUTPUT TAPE 9,8,ANG(I),SIG(I),PB(I),SIGC(I)	1226

8	FORMAT(5X,0PF6.2,4X,0PE12.4,4X,0PE12.4,4X,0PE12.4)	1227
	IF (IPR-1) 13,9,9	1228
9	WRITE OUTPUT TAPE 9,10	1229
10	FORMAT(1H1,30X,	1230
	141HDERIVATIVES OF SCATTERING MATRIX ELEMENTS/	1231
	27X,1HL,6X,2HUS,10X,2HRS,10X,2HAS,10X,2HWS,10X,2HWD,	1232
	310X,2HRD,10X,2HAD,10X,3HUSP,9X,3HWSP)	1233
	IX=0	1234
	DO 12 I=1,LM	1235
	WRITE OUTPUT TAPE 9,11,IX,(DSRU(J,I),J=1,9),	1236
	1(DSIU(J1,I),J1=1,9),(DSRD(J2,I),J2=1,9),	1237
	2(DSID(J3,I),J3=1,9)	1238
11	FORMAT(6H0 SR+ ,13,9(0PE12.3)/2X,7HSI+ ,	1239
	19(0PE12.3)/2X,7HSR- ,9(0PE12.3)/2X,7HSI- ,	1240
	29(0PE12.3))	1241
12	IX=IX+1	1242
13	RETURN	1243
	END	1244

TEST CASE FOR IBM-7090

SEARCH NON-LOCAL OPTICAL POTENTIAL

PARAMETERS

EXPERIMENTAL

THEORETICAL

ELAB	# 4.100	US	# 71.000		
MT	# 56.000	WS	# -0.	WD	# 15.000
ZT	# -0.	RS	# 1.220	RD	# 1.220
MI	# 1.000	AS	# 0.650	AD	# 0.470
ZI	# -0.				

RESULTS

SIG.REACTION # D.157058E 04

SIG.ELASTIC # D.207966E 04

THETA	SIG.	POL.	SIG/SIGC
0.	0.1740E 04	0.	0.
5.00	0.1706E 04	0.2637E-02	0.
10.00	0.1609E 04	0.4911E-02	0.
15.00	0.1458E 04	0.6420E-02	0.
20.00	0.1267E 04	0.6683E-02	0.
25.00	0.1054E 04	0.5075E-02	0.
30.00	0.8373E 03	0.7307E-03	0.
35.00	0.6315E 03	-0.7619E-02	0.
40.00	0.4489E 03	-0.2196E-01	0.
45.00	0.2972E 03	-0.4561E-01	0.
50.00	0.1797E 03	-0.8463E-01	0.
55.00	0.9581E 02	-0.1513E-00	0.
60.00	0.4220E 02	-0.2730E-00	0.
65.00	0.1358E 02	-0.4958E-00	0.
70.00	0.3970E 01	-0.2167E-00	0.
75.00	0.7396E 01	0.7078E 00	0.
80.00	0.1849E 02	0.5770E 00	0.
85.00	0.3278E 02	0.4472E-00	0.
90.00	0.4682E 02	0.3554E-00	0.
95.00	0.5817E 02	0.2806E-00	0.
100.00	0.6534E 02	0.2104E-00	0.
105.00	0.6765E 02	0.1375E-00	0.
110.00	0.6517E 02	0.5636E-01	0.
115.00	0.5854E 02	-0.3798E-01	0.
120.00	0.4890E 02	-0.1510E-00	0.
125.00	0.3768E 02	-0.2886E-00	0.
130.00	0.2645E 02	-0.4548E-00	0.
135.00	0.1672E 02	-0.6336E 00	0.
140.00	0.9704E 01	-0.7024E 00	0.
145.00	0.6164E 01	-0.2332E-00	0.
150.00	0.6284E 01	0.7039E 00	0.
155.00	0.9634E 01	0.9861E 00	0.
160.00	0.1525E 02	0.8296E 00	0.
165.00	0.2180E 02	0.6022E 00	0.
170.00	0.2784E 02	0.3869E-00	0.
175.00	0.3206E 02	0.1888E-00	0.
180.00	0.3357E 02	0.6393E-07	0.

TEST CASE FOR IBM-7090 (CONTINUED)

			NUMERICAL	
USP	#	1300.00	H	# 0.150
WSP	#	-0.	HC	# 0.100
BETA	#	0.85	LM	# 20

SIG.TOTAL # 0.365024E 04

S MATRIX

L	SR+	SI+	IT	SR-	SI-	IT
0	0.2740	-0.1583	6	0.2740	-0.1583	0
1	-0.2011	-0.6271	5	-0.3219	-0.5925	5
2	0.0039	-0.4952	6	-0.0053	-0.2687	6
3	0.7897	0.1612	4	0.8948	0.0196	6
4	0.9694	-0.0048	4	0.9492	0.0038	5
5	0.9957	0.0049	4	0.9984	0.0026	4
6	0.9999	0.0003	4	0.9999	0.0001	3

TEST CASE FOR CDC-1604

SEARCH NON-LOCAL OPTICAL POTENTIAL

PARAMETERS

EXPERIMENTAL

THEORETICAL

ELAR = 4.100

US = 71.000

MT = 56.000

WS = .000

WD = 15.000

ZT = .000

BS = 1.220

RD = 1.220

MI = 1.000

AS = .650

AD = .470

ZI = .000

RESULTS

SIG. REACTION = 1.569143E 003

SIG. ELASTIC = 2.081708E 003

THETA	SIG.	POL.	SIG/SIGC
.00	1.7415E 003	.0000E 000	.0000E 000
5.00	1.7079E 003	2.6111E-003	.0000E 000
10.00	1.6104E 003	4.8573E-003	.0000E 000
15.00	1.4590E 003	6.3370E-003	.0000E 000
20.00	1.2682E 003	6.5670E-003	.0000E 000
25.00	1.0554E 003	4.9215E-003	.0000E 000
30.00	8.3823E 002	5.3469E-004	.0000E 000
35.00	6.3218E 002	-7.8648E-003	.0000E 000
40.00	4.4936E 002	-2.2263E-002	.0000E 000
45.00	2.9747E 002	-4.5988E-002	.0000E 000
50.00	1.7982E 002	-8.5118E-002	.0000E 000
55.00	9.5863E 001	-1.5197E-001	.0000E 000
60.00	4.2179E 001	-2.7416E-001	.0000E 000
65.00	1.3535E 001	-4.9876E-001	.0000E 000
70.00	3.9236E 000	-2.1806E-001	.0000E 000
75.00	7.3713E 000	7.1254E-001	.0000E 000
80.00	1.8495E 001	5.7799E-001	.0000E 000
85.00	3.2818E 001	4.4734E-001	.0000E 000
90.00	4.6883E 001	3.5523E-001	.0000E 000
95.00	5.8254E 001	2.8025E-001	.0000E 000
100.00	6.5429E 001	2.0998E-001	.0000E 000
105.00	6.7743E 001	1.3698E-001	.0000E 000
110.00	6.5251E 001	5.5782E-002	.0000E 000
115.00	5.8611E 001	-3.8626E-002	.0000E 000
120.00	4.8952E 001	-1.5170E-001	.0000E 000
125.00	3.7714E 001	-2.8950E-001	.0000E 000
130.00	2.6474E 001	-4.5603E-001	.0000E 000
135.00	1.6730E 001	-6.3565E-001	.0000E 000
140.00	9.7040E 000	-7.0620E-001	.0000E 000
145.00	6.1564E 000	-2.3916E-001	.0000E 000
150.00	6.2690E 000	7.0049E-001	.0000E 000
155.00	9.6139E 000	9.8514E-001	.0000E 000
160.00	1.5226E 001	8.2922E-001	.0000E 000
165.00	2.1774E 001	6.0198E-001	.0000E 000
170.00	2.7808E 001	3.8674E-001	.0000E 000
175.00	3.2030E 001	1.8871E-001	.0000E 000
180.00	3.3543E 001	7.8020E-009	.0000E 000

TEST CASE FOR CDC-1604 (CONTINUED)

NUMERICAL

USP	=	1300.00	H	=	.150
WSP	=	.00	HC	=	.100
BETA	=	.85	LM	=	20

SIG.TOTAL = 3.650851E 003

S MATRIX

L	SR+	SI+	IT	SR-	SI-	IT
0	.2744	-.1584	6	.2744	-.1584	0
1	-.2026	-.6268	5	-.3233	-.5920	5
2	.0037	-.4962	3	-.0054	-.2693	6
3	.7902	.1603	4	.8950	.0187	6
4	.9694	-.0050	4	.9493	.0036	5
5	.9957	.0049	4	.9984	.0025	4
6	.9999	.0003	4	.9999	.0001	4

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

ORNL-3429
UC-34 - Physics
TID-4500 (20th ed.)

INTERNAL DISTRIBUTION

1. Biology Library	70. F. K. McGowan
2-3. Central Research Library	71. F. C. Maienschein
4. Reactor Division Library	72. C. D. Moak
5-6. ORNL - Y-12 Technical Library	73. C. W. Nestor
Document Reference Section	74. S. K. Penny
7-56. Laboratory Records Department	75-84. F. G. Perey
57. Laboratory Records, ORNL R.C.	85. A. M. Perry
58. Nancy Alexander	86. C. A. Preskitt
59. J. B. Ball	87. M. J. Skinner
60. R. Bender	88. C. D. Susano
61. R. E. Biggers	89. J. A. Swartout
62. E. P. Blizard	90. D. K. Trubey
63. A. C. Downing	91. A. M. Weinberg
64. C. B. Fulmer	92. C. D. Zerby
65. C. D. Goodman	93. R. A. Charpie (consultant)
66. F. F. Haywood	94. P. F. Gast (consultant)
67. W. H. Jordan	95. R. F. Taschek (consultant)
68. W. E. Kinney	96. T. J. Thompson (consultant)
69. C. E. Larson	

EXTERNAL DISTRIBUTION

- 97-98. University of California, Los Angeles (1 copy each to M. A. Melkanoff and D. S. Saxon)
- 99-100. Argonne National Laboratory (1 copy each to P. A. Moldauer and A. Elwin)
101. R. S. Caswell, National Bureau of Standards, Washington
102. L. Cranberg, Los Alamos Scientific Laboratory
- 103-104. Brookhaven National Laboratory (1 copy each to E. Averbach and C. E. Porter)
105. M. H. Kalos, United Nuclear Corporation, White Plains, New York
106. Research and Development Division, AEC, ORO
- 107-743. Given distribution as shown in TID-4500 (20th ed.) under Physics category (75 copies - OTS)