Title: A Fortran Program for Elastic Scattering Analyses with the
Nuclear Optical Model

Author: Michel A. Melkanoff
David S. Saxon
John S. Nodvik
David G. Cantor

Release Date: August 24, 2009 [EBook #29784]

Language: English

Character set encoding: ISO-8859-1

Produced by David Starner, Andrew D. Hwang, and the Online
Distributed Proofreading Team at http://www.pgdp.net

TRANSCRIBER'S NOTE

Minor typographical corrections, changes to the presentational style, and regularizations of spelling and hyphenation have been made without comment.

Every effort has been made to remove OCR errors from the `FORTRAN` code.

This PDF file is formatted for screen viewing, but may be easily recompiled for printing. Please see the preamble of the LaTeX source file for instructions.

A FORTRAN Program for Elastic Scattering Analyses with the Nuclear Optical Model

MICHEL A. MELKANOFF
University of California, Los Angeles

DAVID S. SAXON
University of California, Los Angeles

JOHN S. NODVIK
University of Southern California

DAVID G. CANTOR
University of California, Los Angeles

UNIVERSITY OF CALIFORNIA PRESS
BERKELEY AND LOS ANGELES 1961

UNIVERSITY OF CALIFORNIA PUBLICATIONS IN AUTOMATIC COMPUTATION

NUMBER 1

UNIVERSITY OF CALIFORNIA PRESS, Berkeley and Los Angeles, California
CAMBRIDGE UNIVERSITY PRESS, London, England

$4.50

Second Printing, 1961

PRINTED IN THE UNITED STATES OF AMERICA

Acknowledgements

The authors would like to express their sincere appreciation to the Western Data Processing Center, Graduate School of Business Administration, UCLA, for the use of their IBM 709 computer. Special thanks are due to Mrs. Lisa Greenstadt and Mrs. Lois Holloway who have worked intensively and skillfully to prepare the program.

This program is largely based on experience gained on the SWAC, and the authors recall this with gratitude to Numerical Analysis Research, Department of Mathematics, UCLA.

Finally the authors would like to express their appreciation to the National Science Foundation and the Office of Naval Research for financial support.

Table of Contents

I. Introduction

The purpose of the present report is to describe in complete detail a FORTRAN code named Program SCAT 4 written by the UCLA group in order to analyze elastic scattering of various particles against complex nuclei by means of the diffuse surface optical model of the nucleus.

While a number of similar programs have been prepared and used by other groups, there have been many requests for the UCLA program because of its flexibility and the availability of IBM 704 and 709 computers for which the program is written.

The present program still contains some undesirable features and the UCLA group is constantly modifying it to make it more efficient and flexible. However, a "final" program will probably never be reached and it was decided to release Program SCAT 4 without further delay; as they develop, modifications and additions will be described in later reports.

Other laboratories will probably add further modifications and the UCLA group will be grateful for description of such modifications as well as for any suggestions in this regard. Modifications and additions deemed worthwhile will be passed on to other users of the program but while the UCLA group is willing to serve partially as a central clearing house, the entire clerical responsibility cannot be assumed by the UCLA group.

It should also be noted that, while every effort has been made to check out the program, the UCLA group cannot guarantee its complete correctness.

Program SCAT 4 is available on a symbolic deck and will be mailed on request. Air mailing will require prepaid postage by requesting parties.

Potential users of program SCAT 4 may find it useful to follow these suggestions in reading the present report:

1) If the potential user is only interested in analyses with standard potentials he may proceed as follows:

   a) Read the introduction to the mathematical description.

   b) Consider the fundamental equations: (34), (35), (51), (78) through (85), (132), (137) through (139) in chapter II.

   c) Read chapter III, section A and the general flow chart.

   d) Read the description of subroutines INPT4 and OUTPT4 in chapter III, section B.

   e) Read chapter IV and VII.

2) If the potential user is interested in all the features of the program, then a perusal of the whole report is advisable. The mathematical description of chapter II is a brief review of the theory and the basic equations are all listed there. Symbolic FORTRAN variables are indicated in capital letters and may be looked up in the glossary making up chapter V.

Note that the program may be used for incident neutral particle by letting $ZZ' = 0$.

II. Mathematical Description

Program `SCAT 4` calculates in the center-of-mass system the differential elastic scattering cross sections $\sigma(\theta)$, the polarization $P(\theta)$, and the total reaction cross section σ_R for particles of spin 0 or 1/2 having any mass, charge and (non-relativistic) energy scattered by spinless nuclei of any mass and charge for various sets of diffuse surface optical model parameters. The incident and target particles are assumed to interact through a two-body potential consisting of a complex nuclear potential which includes spin-orbit interaction and whose shape can be specified by input parameters. When the incident particle is charged, the two body potential contains, in addition, the coulomb potential between an incident point charge and an extended, constant charge density target.

The calculations include numerical integrations of the radial Schroedinger equations for the effective partial waves. The complex phase shifts are obtained as usual by matching the logarithmic derivatives of the numerically obtained nuclear wave functions to that of the coulomb (or spherical Bessel) functions. The phase shifts are then used to compute polarizations and cross sections which may be compared to the experimental values by means of the χ^2 test.

A. General Formulation

We begin with a brief review of the basic theory relating to the scattering of spin 1/2 particles by a zero spin target[1]. We shall first consider the case of an uncharged incident particle and indicate later the modifications necessary if the incident particle is charged.

The interaction is assumed to be of the form

$$V_T = V_1 + V_2 \, \vec{S} \cdot \vec{L} \tag{1}$$

where V_1 and V_2 are complex quantities depending only on the distance r between the incident particle and the target particle. In terms of the Pauli spin operator $\vec{\sigma}$, the spin operator of the incident particle, $\vec{S}$, is given by

$$\vec{S} = \frac{1}{2}\hbar\vec{\sigma} \tag{2}$$

and the (relative) orbital angular momentum operator is given by

$$\vec{L} = \vec{r} \times \left(\frac{\hbar}{i} \vec{\nabla} \right). \tag{3}$$

The Schroedinger equation is then

$$\left[-\frac{\hbar^2}{2\mu} \vec{\nabla}^2 + V_1(r) + V_2(r) \, \vec{S} \cdot \vec{L} \right] \Psi = E\Psi \tag{4}$$

[1] See J. Lepore, Phys. Rev. **79**, 137 (1950).

where

$$\mu = \frac{m_i\, m_b}{m_i + m_b} \tag{5}$$

is the reduced mass, m_i and m_b being respectively the masses of the incident and target particles in atomic mass units.

$$E = \frac{m_b}{m_i + m_b} E_{\rm LAB} \tag{6}$$

is the energy in the center of mass system, $E_{\rm LAB}$ being the lab energy of the incident particle in MeV.

1. Uncharged Incident Particles

The wave function corresponding to a wave incident in the positive z direction and normalized to one incident particle per unit time per unit area is

$$\Psi_{\rm inc} = \frac{1}{\sqrt{v}} e^{ikz} \chi_{\rm inc} \tag{7}$$

where v is the relative velocity, the wave number k is given by

$$k = \sqrt{\frac{2\mu E}{\hbar^2}} = 0.2195376\sqrt{\mu E}\ \text{fermi}^{-1} \tag{8}$$

and the incident spin function is

$$\chi_{\rm inc} = a_{1/2}\alpha + a_{-1/2}\beta \tag{9}$$

where α and β are normalized spin eigenfunctions of S_z and $a_{1/2}$, $a_{-1/2}$ the corresponding amplitudes.

The partial wave expansion corresponding to (7) is given by:

$$\Psi_{\rm inc} = \frac{1}{\sqrt{v}} \sum_{\ell=0}^{\infty} (2\ell+1) i^\ell j_\ell(kr) \sqrt{\frac{4\pi}{2\ell+1}} Y_\ell^0(\theta,\varphi) \left[a_{1/2}\alpha + a_{-1/2}\beta\right] \tag{10}$$

where $j_\ell(kr)$ is the regular spherical Bessel function of order ℓ and the normalized spherical harmonics are defined as

$$Y_\ell^m(\theta,\varphi) = (-1)^{\frac{m+|m|}{2}} \sqrt{\frac{2\ell+1}{4\pi}} \sqrt{\frac{(\ell-|m|)!}{(\ell+|m|)!}} P_\ell^{|m|}(\cos\theta) e^{im\varphi} \tag{11}$$

where $P_\ell^{|m|}(\cos\theta)$ are the associated Legendre polynomials.

The product functions $Y_\ell^0\alpha$ and $Y_\ell^0\beta$ which appear in (10) are simultaneous eigenfunctions of the operators $\vec{L}^2$, L_z, $\vec{S}^2$, and S_z but not of the operator $\vec{L}\cdot\vec{S}$ which appears in the spin-orbit interaction. This may be remedied by introducing functions $\mathscr{Y}_{j\ell s}^{m_j}$ which

are simultaneous eigenfunctions of $\vec{L}^2$, $\vec{S}^2$, $\vec{J}^2$, and J_z and thus of $\vec{L}\cdot\vec{S}$ where $\vec{J}$ is the total angular momentum,

$$\vec{J} = \vec{L} + \vec{S}. \tag{12}$$

Since $s = 1/2$, the possible values of j are $j = \ell + 1/2$ and $j = \ell - 1/2$; the corresponding eigenfunctions are given by

$$\left.\begin{aligned} \mathscr{Y}^{m_j}_{\ell+1/2,\ell,s} &= \sqrt{\frac{\ell+m_j+1/2}{2\ell+1}}\,Y_\ell^{m_j-1/2}\alpha + \sqrt{\frac{\ell-m_j+1/2}{2\ell+1}}\,Y_\ell^{m_j+1/2}\beta, \text{ for } j=\ell+1/2 \\ \mathscr{Y}^{m_j}_{\ell-1/2,\ell,s} &= -\sqrt{\frac{\ell-m_j+1/2}{2\ell+1}}\,Y_\ell^{m_j-1/2}\alpha + \sqrt{\frac{\ell+m_j+1/2}{2\ell+1}}\,Y_\ell^{m_j+1/2}\beta, \text{ for } j=\ell-1/2 \end{aligned}\right\} \tag{13}$$

The incident wave function may now be written as

$$\begin{aligned} \Psi_{\text{inc}} = &\sqrt{\frac{4\pi}{V}}\sum_{\ell=0}^{\infty}\sqrt{\ell+1}\,i^\ell\,j_\ell(kr)\left[\;\; a_{1/2}\,\mathscr{Y}^{1/2}_{\ell+1/2,\ell,1/2} + a_{-1/2}\,\mathscr{Y}^{-1/2}_{\ell+1/2,\ell,1/2}\right] \\ &+\sqrt{\frac{4\pi}{V}}\sum_{\ell=0}^{\infty}\quad\sqrt{\ell}\;\; i^\ell\,j_\ell(kr)\left[-a_{1/2}\,\mathscr{Y}^{1/2}_{\ell-1/2,\ell,1/2} + a_{-1/2}\,\mathscr{Y}^{-1/2}_{\ell-1/2,\ell,1/2}\right] \end{aligned} \tag{14}$$

The total wave function can be written in a form similar to (14):

$$\begin{aligned} \Psi_{\text{total}} &= \Psi_{\text{inc}} + \Psi_{\text{scatt}} \\ &= \sqrt{\frac{4\pi}{V}}\sum_{\ell=0}^{\infty}\sqrt{\ell+1}\,i^\ell\,\frac{\Psi^+_\ell(r)}{kr}\left[\;\; a_{1/2}\,\mathscr{Y}^{1/2}_{\ell+1/2,\ell,1/2} + a_{-1/2}\,\mathscr{Y}^{-1/2}_{\ell+1/2,\ell,1/2}\right] \\ &+\sqrt{\frac{4\pi}{V}}\sum_{\ell=0}^{\infty}\quad\sqrt{\ell}\;\; i^\ell\,\frac{\Psi^-_\ell(r)}{kr}\left[-a_{1/2}\,\mathscr{Y}^{1/2}_{\ell-1/2,\ell,1/2} + a_{-1/2}\,\mathscr{Y}^{-1/2}_{\ell-1/2,\ell,1/2}\right] \end{aligned} \tag{15}$$

where Ψ^+_ℓ is the radial function associated with $j = \ell + 1/2$ and Ψ^-_ℓ is associated with $j = \ell - 1/2$.

The terms appearing in (15) are not coupled by the spin-orbit interaction, and substitution into the Schroedinger equation (4) yields the following radial equations:

$$\frac{d^2\Psi^\pm_\ell}{dr^2} + \left\{k^2 - \frac{2\mu}{\hbar^2}\left[V_1 + \frac{\hbar^2}{2}\begin{pmatrix}\ell\\ \text{or}\\ -\ell-1\end{pmatrix}V_2\right] - \frac{\ell(\ell+1)}{r^2}\right\}\Psi^\pm_\ell = 0 \tag{16}$$

where the quantity ℓ appears in the equation for $\Psi^\pm_\ell$ and $-\ell-1$ appears in the equation for Ψ^-_ℓ.

The radial wave function $\Psi^\pm_\ell$ must reduce to that of the incident wave, $kr\,j_\ell(kr)$, when there is no interaction and must be such that only the outgoing wave is modified by the interaction. These conditions are satisfied by the asymptotic expression

$$\Psi^\pm_\ell \cong kr\,j_\ell(kr) + C^\pm_\ell\left[-y_\ell(kr) + i\,j_\ell(kr)\right] \tag{17}$$

which reduces to

$$\Psi_\ell^\pm \cong kr\, j_\ell(kr) + C_\ell^\pm\, e^{i(kr-\ell\pi/2)} \tag{18}$$

or equivalently

$$\Psi_\ell^\pm \cong \sin(kr - \frac{\ell\pi}{2}) + C_\ell^\pm e^{i(kr-\ell\pi/2)} \tag{19}$$

as may be seen by applying the asymptotic expression for the regular and irregular spherical Bessel functions:

$$\left.\begin{aligned} kr\, j_\ell(kr) &\cong \quad \sin(kr - \ell\pi/2) \\ kr\, y_\ell(kr) &\cong -\cos(kr - \ell\pi/2). \end{aligned}\right\} \tag{20}$$

On the other hand, in terms of complex phase shifts $\delta_\ell^\pm$, (19) must be of the form

$$\Psi_\ell^\pm \cong A_\ell^\pm \sin(kr - \ell\pi/2 + \delta_\ell^\pm) \tag{21}$$

Comparison of the coefficients of e^{ikr} and e^{-ikr} in eqs. (21) and (19) yields

$$C_\ell^\pm = \frac{1}{2i}(e^{2i\delta_\ell^\pm} - 1) \tag{22}$$

$$A_\ell^\pm = e^{i\delta_\ell^\pm} \tag{23}$$

Substituting (18) into (15) and subtracting $\Psi_{\rm inc}$ as given by (14), yields for $\Psi_{\rm scatt}$ the asymptotic form:

$$\Psi_{\rm scatt} \cong \frac{1}{\sqrt{V}}\frac{e^{ikr}}{r}\left\{A(\theta)\left[a_{1/2}\alpha + a_{-1/2}\beta\right] + iB(\theta)\left[a_{-1/2}e^{-i\varphi}\alpha - a_{1/2}e^{i\varphi}\beta\right]\right\} \tag{24}$$

where

$$\left.\begin{aligned} A(\theta) &= \quad \frac{1}{k}\sum_{\ell=0}^{\infty}\left[(\ell+1)C_\ell^+ + \ell C_\ell^-\right] P_\ell(\cos\theta) \\ B(\theta) &= -\frac{i}{k}\sum_{\ell=0}^{\infty}\left[C_\ell^+ - C_\ell^-\right] P_\ell^1(\cos\theta) \end{aligned}\right\} \tag{25}$$

The wave function of the scattered wave can more conveniently be expressed in terms of $\vec{\sigma}$ and $\vec{n}$, the unit vector normal to the scattering plane defined by

$$\vec{n}\sin\theta = \vec{k_1} \times \vec{k_0} \tag{26}$$

where $\vec{k}_0$ and $\vec{k}_1$ are unit vectors in the direction of propagation before and after scattering; thus

$$\Psi_{\rm scatt} \cong \frac{1}{\sqrt{V}}\frac{e^{ikr}}{r}\left[A(\theta) + B(\theta)\vec{\sigma}\cdot\vec{n}\right]\chi_{\rm inc} = \frac{1}{\sqrt{V}}\frac{e^{ikr}}{r}f(\theta)\chi_{\rm inc} \tag{27}$$

where $f(\theta)$ is the operator

$$f(\theta) = A(\theta) + B(\theta)\vec{\sigma}\cdot\vec{n}. \tag{28}$$

The differential elastic scattering cross section and polarization vector which are given by

$$\sigma(\theta) = \left\langle [f(\theta)\chi_{\rm inc}]^\dagger \, [f(\theta)\chi_{\rm inc}] \right\rangle \tag{29}$$

$$\vec{P}(\theta) = \frac{\left\langle [f(\theta)\chi_{\rm inc}]^\dagger \, [f(\theta)\chi_{\rm inc}] \right\rangle}{\sigma(\theta)} \tag{30}$$

thus become

$$\sigma(\theta) = |A|^2 + |B|^2 + (A^*B + AB^*)\vec{n}\cdot\vec{P}_0 \tag{31}$$

$$\vec{P}(\theta) = \frac{(|A|^2 - |B|^2)\vec{P}_0 + \left[A^*B + AB^* + 2|B|^2\vec{P}_0\cdot\vec{n}\right]\vec{n} + i(A^*B - AB^*)\vec{n}\times\vec{P}_0}{|A|^2 + |B|^2 + (A^*B + AB^*)\vec{P}_0\cdot\vec{n}} \tag{32}$$

where the incident polarization vector $\vec{P}_0$, is given by

$$\vec{P}_0 = \left\langle \chi^\dagger_{\rm inc} \; \vec{\sigma}\chi_{\rm inc} \right\rangle \tag{33}$$

If the incident beam is unpolarized, i.e., $\vec{P}_0 = 0$, the scattered beam is polarized along the direction $\vec{n}$, perpendicular to the scattering plane and

$$\sigma(\theta) = |A|^2 + |B|^2 \tag{34}$$

$$\vec{P}(\theta) = P(\theta)\vec{n} = \frac{(A^*B + AB^*)}{|A|^2 + |B|^2}\vec{n} \tag{35}$$

Experimentally, the polarization is sometimes obtained from a double scattering experiment in the same plane wherein the polarization in the first scattering is known[2].
The differential elastic scattering cross section for the second scattering may then be obtained from (31) and (35):

$$\begin{aligned}\sigma_2(\theta) &= (|A|^2 + |B|^2)\left[1 + \frac{A^*B + AB^*}{|A|^2 + |B|^2}\vec{n}_2\cdot\vec{P}_1\right] \\ &= (|A|^2 + |B|^2)(1 + \vec{P}_2\cdot\vec{P}_1).\end{aligned} \tag{36}$$

Referring to FIGURE 1, it is clear that

$$\vec{n}_1 = \vec{n}^r_2 = -\vec{n}^\ell_2, \tag{37}$$

so that the differential scattering cross sections along the r and ℓ beams are as follows:

$$\left.\begin{aligned}\sigma^r_2(\theta) &= (|A|^2 + |B|^2)(1 + P_2P_1) \\ \sigma^\ell_2(\theta) &= (|A|^2 + |B|^2)(1 - P_2P_1),\end{aligned}\right\} \tag{38}$$

[2]L. Rosen, Proceedings of the International Conference on the Nuclear Optical Model, Florida State University, Tallahassee, 1959, pp. 72–90.

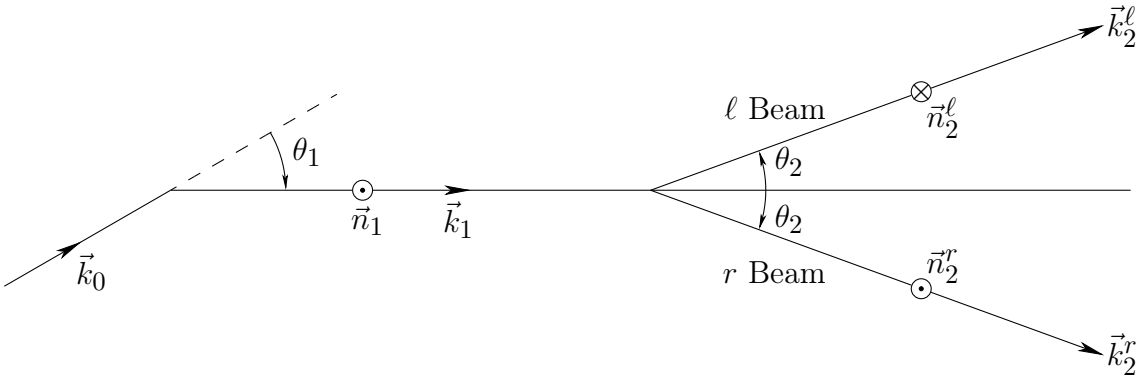


FIG. 1

the ratio of the scattering intensities becomes

$$\frac{\sigma_2^{\ell}(\theta)}{\sigma_2^{r}(\theta)} = \frac{1 - P_2 P_1}{1 + P_2 P_1}, \tag{39}$$

and solving for P_2:

$$P_2 = \frac{1}{P_1} \frac{\sigma_2^{\ell} - \sigma_2^{r}}{\sigma_2^{\ell} + \sigma_2^{r}} \tag{40}$$

which reduces when $P_1 = 1$ to

$$P_2 = \frac{\sigma_2^{\ell} - \sigma_2^{r}}{\sigma_2^{\ell} + \sigma_2^{r}} \tag{41}$$

2. Charged Incident Particles

We next consider the case in which the incident particle has charge Ze and the target particle has charge $Z'e$. The potential $V(r)$ must now include a term $V_c(r)$ which describes the coulomb interaction. For small values of r, V_c will depend on the assumed charge distribution, while for large values of r, we must have

$$V_c = \frac{ZZ'e^2}{r} \quad (r \text{ large}). \tag{42}$$

It is convenient to introduce the parameter η,

$$\eta = \frac{\mu\ ZZ'e^2}{\hbar^2 k} = 0.15805086\, ZZ' \sqrt{\frac{m_i}{E_{\rm LAB}}} \tag{43}$$

For the "incident wave" we take $\Psi_c(r)\chi_{\rm inc}$, where Ψ_c is the solution to the Schroedinger equation

$$-\frac{\hbar}{2\mu} \vec{\nabla}^2 \Psi_c + \frac{ZZ'e^2}{r} \Psi_c = E\Psi_c \tag{44}$$

corresponding to the scattering of two point charges.

It is well known that in that case

$$\Psi_c = \frac{1}{\sqrt{V}}\,\Gamma(1+i\eta)e^{-1/2\eta\pi}\,e^{ikz}F(-i\eta,1,ik\xi) \tag{45}$$

where $\xi = r - z$ and F is the confluent hypergeometric function.

It is important to note that Ψ_c includes a distorted incoming wave *plus* a scattered wave due to the point charge potential, and as such is not strictly an incident wave.

The asymptotic form of Ψ_c is given by

$$\begin{aligned}\Psi_c \cong \frac{1}{\sqrt{V}}\Bigg\{ & e^{i[kz\ -\eta\,\ell n\,k(r-z)]}\left(1-\frac{\eta^2}{ik(r-z)}\right) \\ & + \frac{1}{r}f_c(\theta)\,e^{i(kr-\eta\,\ell n\,2kr)}\Bigg\}\end{aligned} \tag{46}$$

where

$$f_c(\theta) = -\frac{\eta}{2k\sin^2\theta/2}\,e^{-i\eta\,\ell n(\sin^2\theta/2)+2i\ \sigma_0} \tag{47}$$

is the Rutherford scattering amplitude and σ_0 is given by equation (49), below, with $\ell = 0$.

The partial wave expansion of Ψ_c is given by

$$\Psi_c = \frac{1}{\sqrt{V}}\sum_{\ell=0}^{\infty}(2\ell+1)\,i^\ell\,e^{i\sigma_\ell}\,\frac{F_\ell(\eta,kr)}{kr}\sqrt{\frac{4\pi}{2\ell+1}}Y_\ell^0(\theta,\varphi) \tag{48}$$

where $F_\ell(\eta,kr)$ is the regular coulomb function and σ_ℓ is the usual coulomb phase shift given by

$$\sigma_\ell = \arg\Gamma(\ell+1+i\eta) \tag{49}$$

Comparing equation (48) with (10) we see that in equation (14) it is necessary to replace $j_\ell(kr)$ by $e^{i\sigma_\ell}\frac{F_\ell(\eta,kr)}{kr}$; thus, in this case,

$$\begin{aligned}\Psi_{\rm inc} = & \sqrt{\frac{4\pi}{V}}\sum_{\ell=0}^{\infty}\sqrt{\ell+1}\,i^\ell\,e^{i\sigma_\ell}\,\frac{F_\ell(\eta,kr)}{kr}\Big[\ a_{1/2}\,\mathscr{Y}^1_{\ell+1/2,\ell,1/2}/2 + a_{-1/2}\,\mathscr{Y}^{-1/2}_{\ell+1/2,\ell,1/2}\Big] \\ & + \sqrt{\frac{4\pi}{V}}\sum_{\ell=0}^{\infty}\quad\sqrt{\ell}\ \ i^\ell\,e^{i\sigma_\ell}\,\frac{F_\ell(\eta,kr)}{kr}\Big[-a_{1/2}\,\mathscr{Y}^1_{\ell-1/2,\ell,1/2}/2 + a_{-1/2}\,\mathscr{Y}^{-1/2}_{\ell-1/2,\ell,1/2}\Big]\end{aligned} \tag{50}$$

The total wave function can be written as a sum of the "incident" wave, $\Psi_{\rm inc}$, plus a "scattered" wave, $\Psi_{\rm scatt}$, where $\Psi_{\rm scatt}$ now includes only interference terms and deviations

from pure Rutherford scattering:

$$\begin{aligned}\Psi_{\text{total}} &= \Psi_{\text{inc}} + \Psi_{\text{scatt}} \\ &= \sqrt{\frac{4\pi}{V}} \sum_{\ell=0}^{\infty} \sqrt{\ell+1}\, i^{\ell}\, e^{i\sigma_\ell}\, \frac{\Psi_\ell^+(r)}{kr} \left[\; a_{1/2}\, \mathscr{Y}^{1}_{\ell+1/2,\ell,1/2}/2 + a_{-1/2}\, \mathscr{Y}^{-1/2}_{\ell+1/2,\ell,1/2} \right] \\ &+ \sqrt{\frac{4\pi}{V}} \sum_{\ell=0}^{\infty} \quad \sqrt{\ell}\;\; i^{\ell}\, e^{i\sigma_\ell}\, \frac{\Psi_\ell^-(r)}{kr} \left[-a_{1/2}\, \mathscr{Y}^{1}_{\ell-1/2,\ell,1/2}/2 + a_{-1/2}\, \mathscr{Y}^{-1/2}_{\ell-1/2,\ell,1/2} \right]\end{aligned} \tag{51}$$

This wave function, Ψ_{total}, is formally almost identical to the expression given by equation (15) and the radial wave functions $\Psi_\ell^\pm$ obey an equation which is formally identical to equation (16) except that $V_1(r)$ must now include the coulomb potential $V_c(r)$ which may differ from a point charge potential at close distances.

The radial wave function $\Psi_\ell^\pm$ must now reduce to the "incident" wave, $F_\ell(\eta, kr)$, when the potential becomes a coulomb point charge potential, and must be such that only the outgoing wave is modified by the non-coulomb interaction. These conditions are satisfied by the asymptotic expression:

$$\Psi_\ell^\pm \cong F_\ell(\eta, kr) + C_\ell^\pm \left[G_\ell(\eta, kr) + i F_\ell(\eta, kr) \right] \tag{52}$$

which reduces to

$$\Psi_\ell^\pm \cong F_\ell(\eta, kr) + C_\ell^\pm e^{i(kr - \eta\, \ell n\, 2kr - \ell\pi/2 + \sigma_\ell)} \tag{53}$$

or equivalently

$$\Psi_\ell^\pm \cong \sin(kr - \eta\, \ell n\, 2kr - \ell\pi/2 + \sigma_\ell) + C_\ell^\pm e^{i(kr - \eta\, \ell n\, 2kr - \ell\pi/2 + \sigma_\ell)} \tag{54}$$

as may be seen by introducing the asymptotic expressions for the regular and irregular coulomb functions:

$$\left.\begin{aligned} F_\ell(\eta, kr) &\cong \sin(kr - \eta\, \ell n\, 2kr - \ell\pi/2 + \sigma_\ell) \\ G_\ell(\eta, kr) &\cong \cos(kr - \eta\, \ell n\, 2kr - \ell\pi/2 + \sigma_\ell) \end{aligned}\right\} \tag{55}$$

In this case, the "nuclear phase shift" $\delta_\ell^\pm$ is taken to be such that the asymptotic form of $\Psi_\ell^\pm$ is given by

$$\Psi_\ell^\pm \cong A_\ell^\pm \sin(kr - \eta\, \ell n\, 2kr - \ell\pi/2 + \sigma_\ell + \delta_\ell^\pm) \tag{56}$$

Comparison of the coefficients of $e^{i(kr - \eta\, \ell n\, 2kr)}$ and $e^{-i(kr - \eta\, \ell n\, kr)}$ in equations (54) and (56) yields

$$C_\ell^\pm = \frac{1}{2i} \left[e^{2i\, \delta_\ell^\pm} - 1 \right] \tag{57}$$

$$A_\ell^\pm = e^{i\, \delta_\ell^\pm} \tag{58}$$

Substituting (53) into (51) and making use of (46) and (50) we obtain for the asymptotic form of the total wave function

$$\Psi_{\text{total}} \cong \frac{1}{\sqrt{V}} \left\{ e^{i[kz - \eta\, \ell n\, k(r-z)]} \left[1 - \frac{\eta^2}{ik(r-z)} \right] \right\} \chi_{\text{inc}} + \frac{1}{\sqrt{V}} \frac{e^{i(kr - \eta\, \ell n\, 2kr)}}{r} \left\{ A(\theta) \left[a_{1/2}\alpha + a_{-1/2}\beta \right] + iB(\theta) \left[a_{-1/2} e^{-i\varphi}\alpha - a_{1/2} e^{i\varphi}\beta \right] \right\} \tag{59}$$

where

$$\begin{aligned} A(\theta) &= f_c(\theta) + \frac{1}{k} \sum_{\ell=0}^{\infty} e^{2i\sigma_\ell} \left[(\ell+1) C_\ell^+ + \ell C_\ell^- \right] P_\ell(\cos\theta) \\ B(\theta) &= -\frac{i}{k} \sum_{\ell=0}^{\infty} e^{2i\sigma_\ell} \left[C_\ell^+ - C_\ell^- \right] P_\ell^1(\cos\theta) \end{aligned} \tag{60}$$

and $f_c(\theta)$ is given by equation (47).

From this point, the formulation follows through as in the case of uncharged particles.

B. Optical Model Potential

1. Diffuse Surface Optical Model with Volume Absorption and Coulomb Spin-Orbit.

The interaction (1) is assumed to have the form

$$V_T = V_{\text{CN}} + V_{\text{SO}} + V_{\text{Coul}} + V_{\text{Coul SO}} \tag{61}$$

where the terms appearing in equation (61) are respectively the central nuclear, spin-orbit nuclear, coulomb, and coulomb spin-orbit potentials.

We shall first consider the case for which the real and imaginary parts of the central potential have a special common form factor (corresponding to volume absorption), and the spin-orbit potential is of the Thomas type. This particular central potential form factor has been used extensively and will be referred to as the standard form factor. We shall then discuss other form factors available in the program.

(a) Central nuclear potential

$$V_{\text{CN}} = (-V - iW) \frac{1}{(1 + e^{(r-R_N)/a})} \tag{62}$$

where V and W are respectively the depths of the real and imaginary part of the nuclear potential in MeV (V and W are positive for an attractive, absorbing potential), and a common volume absorption form factor is assumed, where

$$R_N = R_{\text{ON}} m_b^{1/3} \times 10^{-13} \text{ cm} \tag{63}$$

R_{ON} being the nuclear radius constant and a is the rounding parameter in 10^{-13} cm.

(b) **Nuclear spin-orbit potential**

The nuclear spin-orbit potential is often written in the Thomas form

$$V_{\rm SO} = \lambda \frac{1}{2M_p^2 c^2} \left\{ \frac{1}{r} \frac{d}{dr} \left[\frac{-V}{1 + e^{(r-R_N)/a}} \right] \right\} \vec{S} \cdot \vec{L} \tag{64}$$

where M_p is the proton test mass and c the velocity of light. If λ were 1, the spin-orbit term would be that predicted by the Dirac equation. To provide more freedom in the model one writes

$$\lambda = 4 \left(\frac{M_p}{M_\pi} \right)^2 \frac{V_S + iW_S}{V} \tag{65}$$

where M_π is the pion rest mass and V_S and W_S are respectively the strengths of the real and imaginary parts of the nuclear spin-orbit potential in MeV.

It may be noted that a negative value of the real part of λ would be in accordance with the shell model of the nucleus where a (real) negative spin-orbit term is required to give the proper level sequence in contra-distinction to the atomic case.

(c) **Coulomb potential**

The coulomb potential is taken here to correspond to a constant charge density within the nucleus extending to a distance R_c given by

$$R_c = R_{\rm oc} m_b^{1/3} \times 10^{-13} \text{ cm} \tag{66}$$

where $R_{\rm oc}$ is the coulomb radius constant; thus

$$\begin{aligned} V_{\rm Coul} &= (ZZ'e^2/2R_c)(3 - r^2/R_c^2) \qquad \text{for } r \le R_c \\ &= ZZ'e^2/r \qquad\qquad\qquad\qquad \text{for } r \ge R_c \end{aligned} \tag{67}$$

(d) **Coulomb spin-orbit potential**

The coulomb spin-orbit term is assumed to have the form[3]

$$V_{\rm Coul\ SO} = (\mu_P - \tfrac{1}{2}) \frac{1}{M_P^2 c^2} \left[\frac{1}{r} \frac{d}{dr} V_{\rm Coul} \right] \vec{S} \cdot \vec{L} \tag{68}$$

where μ_P is the proton magnetic moment in nuclear magnetons. It may be noted that the coulomb spin-orbit term is negligible except at very high energies.

Substituting equations (62), (64), (67), and (68) into equation (16) and transforming to the dimensionless variable

$$\rho = kr \tag{69}$$

[3]W. Heckrotte, Phys. Rev. **101**, l406 (1956).

we find

$$\left\{ -\frac{d^2}{d\rho^2} + \frac{\ell(\ell+1)}{\rho^2} - \left(\frac{V+iW}{E}\right)\left(\frac{1}{1+e^{(\rho-\bar{\rho}_N)/ka}}\right) + \left(\frac{\hbar}{M_\pi c}\right)^2 \left(\frac{V_S+iW_S}{E}\right) k^2 \left[-\frac{1}{\rho}\frac{d}{d\rho}\left(\frac{1}{1+e^{(\rho-\bar{\rho}_N)/ka}}\right)\right] \begin{pmatrix} \ell \\ \text{or} \\ -\ell-1 \end{pmatrix} + U_{\text{Coul}} + U_{\text{Coul SO}} - 1 \right\} \Psi_\ell^\pm(\rho) = 0 \quad (70)$$

where

$$\begin{aligned} U_{\text{Coul}} &= \frac{\eta}{\bar{\rho}_c}\left(3 - \frac{\rho^2}{\bar{\rho}_c^2}\right) \quad \text{for } \rho \le \bar{\rho}_c \\ &= 2\eta/\rho \qquad\qquad \text{for } \rho \ge \bar{\rho}_c \end{aligned} \quad (71)$$

$$\begin{aligned} U_{\text{Coul SO}} &= -\frac{1}{2}\left(\frac{\hbar}{M_P c}\right)^2 (\mu_P - \tfrac{1}{2})(2\eta)\left(k^2/\bar{\rho}_c^3\right) \begin{pmatrix} \ell \\ \text{or} \\ -\ell-1 \end{pmatrix} \quad \text{for } \rho \le \bar{\rho}_c \\ &= -\frac{1}{2}\left(\frac{\hbar}{M_P c}\right)^2 (\mu_P - \tfrac{1}{2})(2\eta)\left(k^2/\rho^3\right) \begin{pmatrix} \ell \\ \text{or} \\ -\ell-1 \end{pmatrix} \quad \text{for } \rho \ge \bar{\rho}_c \end{aligned} \quad (72)$$

and where

$$\bar{\rho}_N = kR_N \quad (73)$$

$$\bar{\rho}_c = kR_c. \quad (74)$$

Substituting now

$$\left(\frac{\hbar}{M_\pi c}\right)^2 = 2.00 \times 10^{-26}\ \text{cm}^2 \quad (75)$$

$$2\eta\, k^2 \cdot \tfrac{1}{2}\left(\frac{\hbar}{M_P c}\right)^2 \cong 2\eta\left(\frac{E}{M_P c^2}\right) = 2\eta\,\frac{E}{931} \quad (76)$$

$$\mu_P - \tfrac{1}{2} = 2.7934 - 0.5 = 2.2934 \quad (77)$$

into equation (70) yields:

$$\begin{aligned} \frac{d^2}{d\rho^2}\Psi_\ell^\pm(\rho) &= \left\{-1 + \frac{\ell(\ell+1)}{\rho^2} - \left(\frac{V+iW}{E}\right)\left(\frac{1}{1+e^{(\rho-\bar{\rho}_N)/ka}}\right) + \frac{\eta}{\bar{\rho}_c}\left(3 - \frac{\rho^2}{\bar{\rho}_c^2}\right)\right. \\ &\left. + \left[2\left(\frac{V_S+iW_S}{E}\right)\left(\frac{k}{a}\right)\left(\frac{1}{\rho}\,\frac{e^{(\rho-\bar{\rho}_N)/ka}}{(1+e^{(\rho-\bar{\rho}_N)/ka})^2}\right) - 0.004926\,\frac{\eta E}{\bar{\rho}_c^3}\right] \begin{pmatrix} \ell \\ \text{or} \\ -\ell-1 \end{pmatrix} \right\} \Psi_\ell^\pm(\rho), \ \text{for } \rho \le \bar{\rho}_c \\ &= \left\{-1 + \frac{\ell(\ell+1)}{\rho^2} - \left(\frac{V+iW}{E}\right)\left(\frac{1}{1+e^{(\rho-\bar{\rho}_N)/ka}}\right) + \frac{2\eta}{\rho}\right. \\ &\left. + \left[2\left(\frac{V_S+iW_S}{E}\right)\left(\frac{k}{a}\right)\left(\frac{1}{\rho}\,\frac{e^{(\rho-\bar{\rho}_N)/ka}}{(1+e^{(\rho-\bar{\rho}_N)/ka})^2}\right) - 0.004926\,\frac{\eta E}{\rho^3}\right] \begin{pmatrix} \ell \\ \text{or} \\ -\ell-1 \end{pmatrix} \right\} \Psi_\ell^\pm(\rho), \ \text{for } \rho \ge \bar{\rho}_c \end{aligned} \quad (78)$$

2. Nuclear Form Factors

Equation (78) may be rewritten in such a way as to display explicitly the various nuclear form factors:

$$\frac{d^2}{d\rho^2}\Psi_\ell^\pm(\rho) = \left\{-1 + \frac{\ell(\ell+1)}{\rho^2} - \frac{V}{E}\,f_{\rm CR}(\rho) - i\frac{W}{E}\,f_{\rm CI}(\rho) + \frac{\eta}{\bar\rho_c}\left(3 - \frac{\rho^2}{\bar\rho_c^2}\right)\right.$$
$$\left. + \left[\frac{V_S}{E}\,\frac{2k}{a}\,f_{\rm SR}(\rho) + i\frac{W_S}{E}\,\frac{2k}{a}\,f_{\rm SI}(\rho) - 0.004926\frac{\eta E}{\bar\rho_c^3}\right]\begin{pmatrix}\ell\\ \text{or}\\ -\ell-1\end{pmatrix}\right\}\Psi_\ell^\pm(\rho),\ \text{for } \rho \le \bar\rho_c$$
$$= \left\{-1 + \frac{\ell(\ell+1)}{\rho^2} - \frac{V}{E}\,f_{\rm CR}(\rho) - i\frac{W}{E}\,f_{\rm CI}(\rho) + \frac{2\eta}{\rho}\right. \tag{79}$$
$$\left. + \left[\frac{V_S}{E}\,\frac{2k}{a}\,f_{\rm SR}(\rho) + i\frac{W_S}{E}\,\frac{2k}{a}\,f_{\rm SI}(\rho) - 0.004926\frac{\eta E}{\rho^3}\right]\begin{pmatrix}\ell\\ \text{or}\\ -\ell-1\end{pmatrix}\right\}\Psi_\ell^\pm(\rho),\ \text{for } \rho \ge \bar\rho_c$$

Three basic nuclear form factors and some special modifications of them are presently available in the program. In addition the coulomb spin-orbit term may be excluded at will. The required form factors may be chosen by assigning the proper values to the symbolic quantities KTRL as described on pages 33 *ff*.

(a) Basic Form Factors

(i) Volume absorption (KTRL(I) = 0, I = 1, 7, 8, 9, 10)

$$f_{\rm CR}(\rho) = f_{\rm CI}(\rho) = \frac{1}{(1 + e^{(\rho-\bar\rho_N)/ka})} \tag{80}$$

$$f_{\rm SR}(\rho) = f_{\rm SI}(\rho) = \frac{1}{\rho}\,\frac{e^{(\rho-\bar\rho_N)/ka}}{(1 + e^{(\rho-\bar\rho_N)/ka})^2} \tag{81}$$

(ii) Gaussian absorption (KTRL(1) = 1)
$f_{\rm CR}$ is given by (80), $f_{\rm SR}$ and $f_{\rm SI}$ are given by (81) and

$$f_{\rm CI}(\rho) = e^{-[(\rho-\bar\rho_G)/kb]^2} \tag{82}$$

where

$$\bar\rho_G = kR_{\rm OG}m_b^{1/3}, \tag{83}$$

$R_{\rm OG}$ being the nuclear Gaussian radius constant, and b determines the Gaussian width.

(iii) Square well (KTRL(1) = 2)

$$\begin{aligned} f_{\rm CR}(\rho) = f_{\rm CI}(\rho) &= 1 \qquad \text{for } \rho \le \bar\rho_N \\ &= 0 \qquad \text{for } \rho \ge \bar\rho_N \end{aligned} \tag{84}$$

$$f_{\rm SR}(\rho) = f_{\rm SI}(\rho) = 0. \tag{85}$$

(b) **Special Central Nuclear Form Factors**[4]

(KTRL(1) = 0)

The purpose of these form factors is to allow one to modify the knee or tail of the potential curve and produce central rises or depressions in the real and/or imaginary parts of the central nuclear potential, as specified by proper choice of the KTRL's.

(i) Form A (KTRL(7) = 1 for real part, KTRL(8) = 1 for imaginary part).

$$\left.\begin{aligned} f_{\mathrm{CR}}(\rho) \text{ and/or } f_{\mathrm{CI}}(\rho) &= [1 + h_A(\rho)]\, f_{nA_1}(\rho) && 0 < \rho \le \rho_{m_A} \\ &= f_{nA_1}(\rho) && \rho_{m_A} \le \rho \le \bar{\rho}_N \\ &= f_{nA_2}(\rho) && \bar{\rho}_N \le \rho \le \rho_{\max} \end{aligned}\right\} \tag{86}$$

(ii) Form B (KTRL(7) = 2 for real part, KTRL(8) = 2 for imaginary part).

$$\left.\begin{aligned} f_{\mathrm{CR}}(\rho) \text{ and/or } f_{\mathrm{CI}}(\rho) &= [1 + h_B(\rho)]\, f_{nB_1}(\rho) && 0 < \rho \le \rho_{m_B} \\ &= \quad f_{nB_1}(\rho) && \rho_{m_B} \le \rho \le \bar{\rho}_N \\ &= \quad f_{nB_2}(\rho) && \bar{\rho}_N \le \rho \le \rho_{\max} \end{aligned}\right\} \tag{87}$$

The presence of forms A and B allows distinct form factors in the real and imaginary parts. The presence of A_1, A_2 and B_1, B_2 allows distinct shapes in the knee and tail of the form factors. Letting x be either A or B, and n be either nA_1, nA_2, nB_1, or nB_2,

$$h_x(\rho) = h_{0x}\left[2\left(\frac{\rho}{\rho_{m_x}}\right)^3 - 3\left(\frac{\rho}{\rho_{m_x}}\right)^2 + 1\right] = h_{0x}\left(1 - \frac{\rho}{\rho_{m_x}}\right)^2\left(1 + \frac{\rho}{\rho_{m_x}}\right) \tag{88}$$

$$f_n(\rho) = \frac{1}{1 + g_n(\rho)} \tag{89}$$

where

$$g_n(\rho) = \exp\left\{\frac{1}{n}\left(\frac{\bar{\rho}_N}{ka}\right)\left[\left(\frac{\rho}{\bar{\rho}_N}\right)^n - 1\right]\right\} \tag{90}$$

where h_{0A}, h_{0B}, nA_1, nA_2, nB_1, nB_2, ρ_{m_A}, ρ_{m_B} are selected constants. (The n's are always taken as ≥ 0.)

Note 1: If h_{0x} is taken to be zero and nx_1, nx_2 are taken to be 1, forms A and B reduce to the volume absorption form.

Note 2: The three curves defined by equations (86) and (87) join smoothly with continuous derivatives as long as ρ_{m_x} is chosen less than $\bar{\rho}_N$.

Note 3: Positive values of h_{0x} will produce central rises in the form factors while negative values will produce a central depression.

[4] J.S. Nodvik, Proceedings of the International Conference on the Nuclear Model, Florida State University, Tallahassee, 1959, pp. 16–23.

Note 4: If $nx_1 > 1$, the knee of the potential will be sharper than for the usual volume absorption case, while $0 \le nx_1 \le 1$ will soften the knee of the curve.

Note 5: If $nx_2 > 1$, this will shorten the potential tail while $0 \le nx_2 \le 1$ will extend it.

Some typical shapes are presented in FIGURES 2, 3, and 4.

(c) SPECIAL NUCLEAR SPIN-ORBIT FORM FACTORS (KTRL(1) = 0)

Two special nuclear spin-orbit form factors are available. They can be applied to the real and/or imaginary parts of the nuclear spin-orbit potential. The first of these form factors corresponds to the Thomas term applied to form A in the central nuclear potential, while the second uses form B itself; this permits one to study the result of deviations from the Thomas form.

(i) DERIVATIVE FORM FACTOR A (KTRL(9) = 1 for real part, KTRL(1O) = 1 for imaginary part)

$$\left.\begin{aligned} f_{\mathrm{SR}}(\rho) \text{ and/or } f_{\mathrm{SI}}(\rho) &= (ka)\left[-\frac{1}{\rho}\frac{d}{d\rho}(\text{form factor A})\right] \\ &= (ka)\left[-\left(\frac{1}{\rho}\frac{dh_A(\rho)}{d\rho}\right) f_{nA_1}(\rho) - (1+h_A(\rho))\left(\frac{1}{\rho}\frac{df_{nA_1}(\rho)}{d\rho}\right)\right] \\ & \qquad\qquad \text{for } 0 \le \rho \le \rho_{m_a} \\ &= (ka)\left[-\frac{1}{\rho}\frac{df_{nA_1}(\rho)}{d\rho}\right] \qquad \text{for } \rho_{m_a} \le \rho \le \bar{\rho}_N \\ &= (ka)\left[-\frac{1}{\rho}\frac{df_{nA_2}(\rho)}{d\rho}\right] \qquad \text{for } \bar{\rho}_N \le \rho \le \rho_{\max} \end{aligned}\right\} \tag{91}$$

where

$$-\frac{1}{\rho}\frac{dh_A(\rho)}{d\rho} = \frac{6h_{0A}}{\rho_{m_A}^2}\left(1 - \frac{\rho}{\rho_{m_A}}\right) \tag{92}$$

$$-\frac{1}{\rho}\frac{df_n\rho}{d\rho} = \left(\frac{\bar{\rho}_N}{ka}\right)\frac{1}{\rho^2}\left(\frac{\rho}{\bar{\rho}_N}\right)^n g_n(\rho)\,[f_n(\rho)]^2 \tag{93}$$

and $f_n(\rho)$ and $g_n(\rho)$ are given by equations (89) and (90).

(ii) FORM FACTOR B (KTRL(9) = 2 for real part, KTRL(10) = 2 for imaginary part)

$$f_{\mathrm{SR}}(\rho) \text{ and/or } f_{\mathrm{SI}}(\rho) = \frac{1}{2}.\ [\text{form factor B as per equation (87)}] \tag{94}$$

Note: If h_{0A} is taken to be zero while nA_1 and nA_2 are taken to be 1, the derivative form factor in (91) becomes identical to the usual spin-orbit form factor (81).

Some typical shapes are presented in FIGURES 5, 6, and 7.

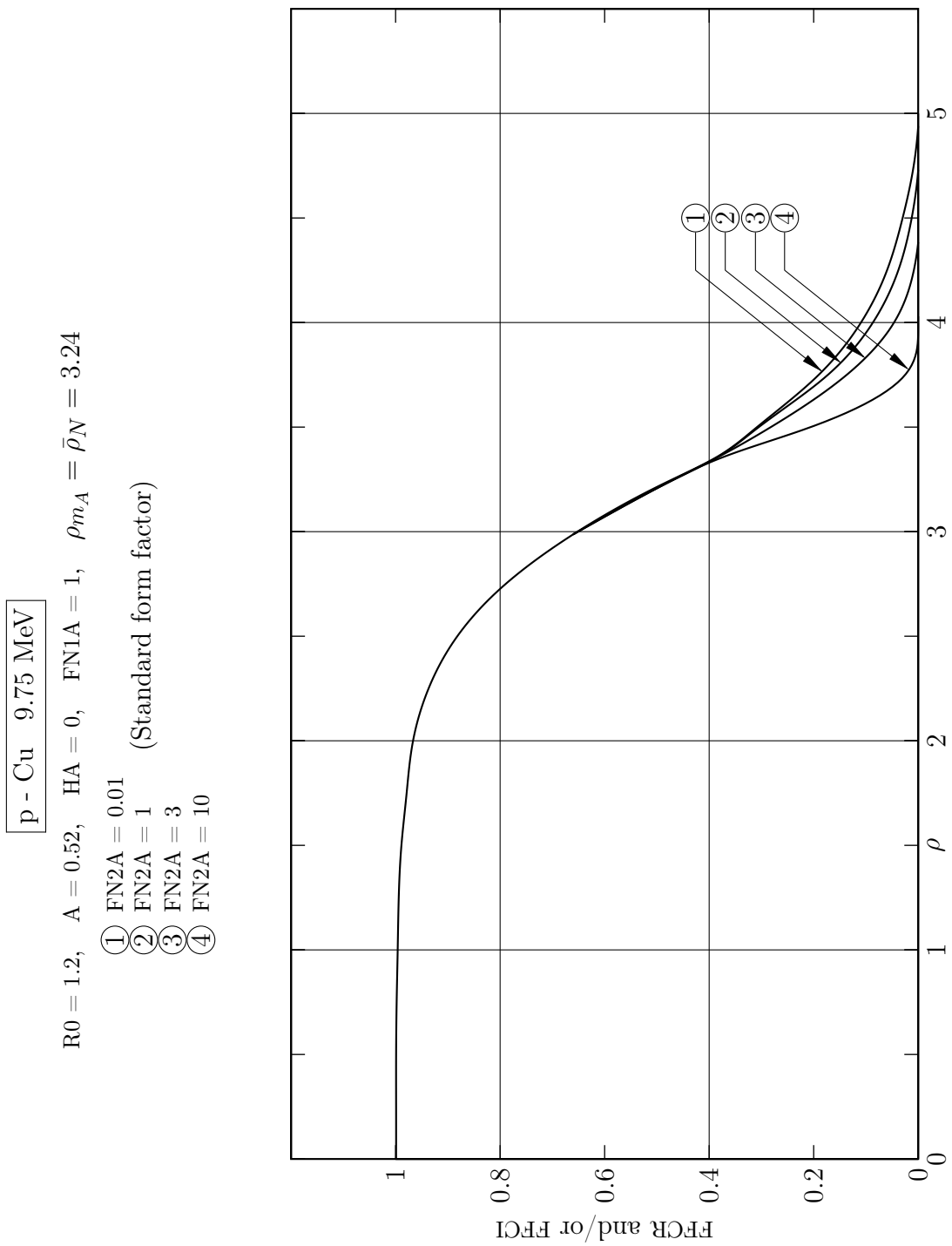


FIG. 2 – Central form factors, tail variation.

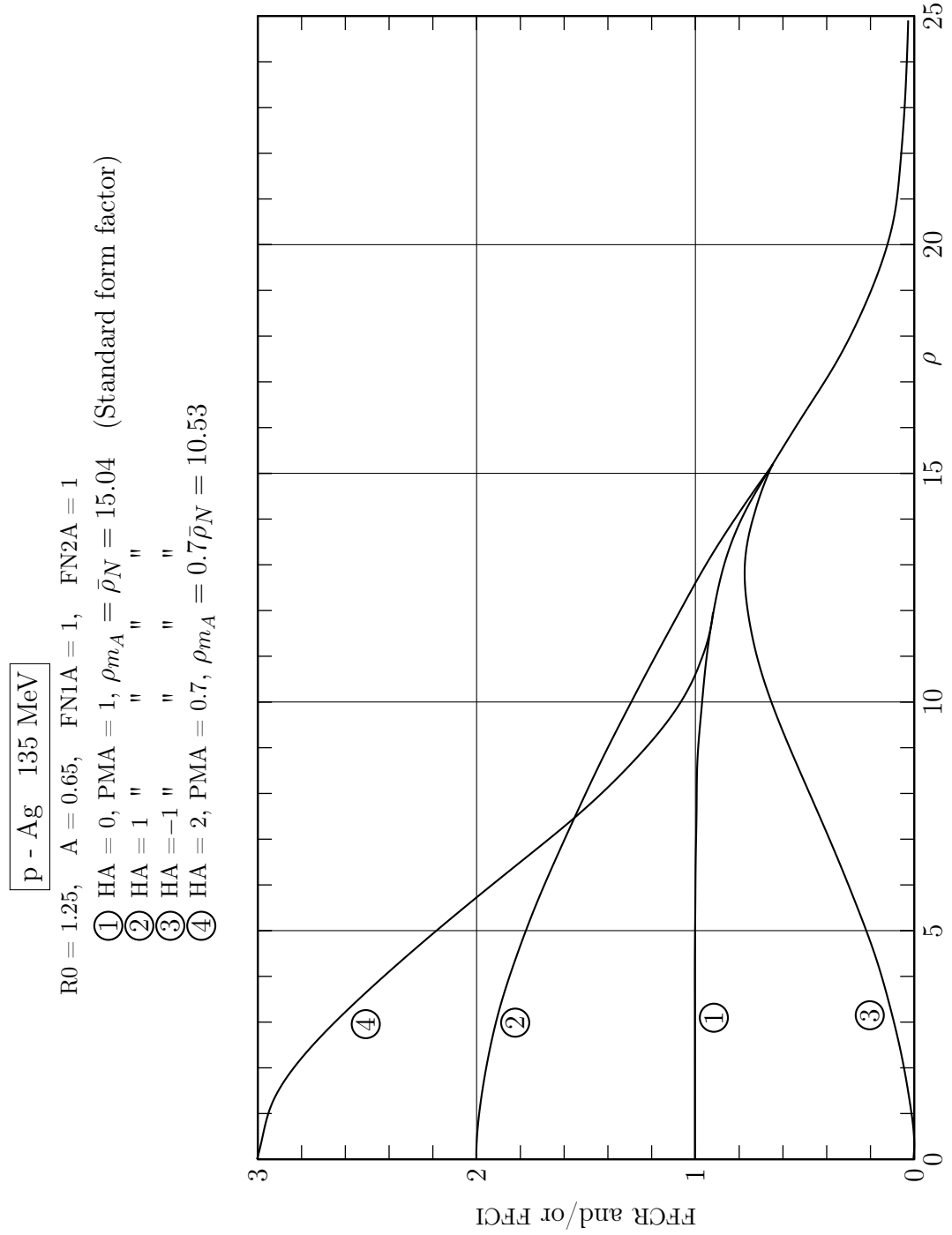


FIG. 3 – Central form factors; central dips and rises.

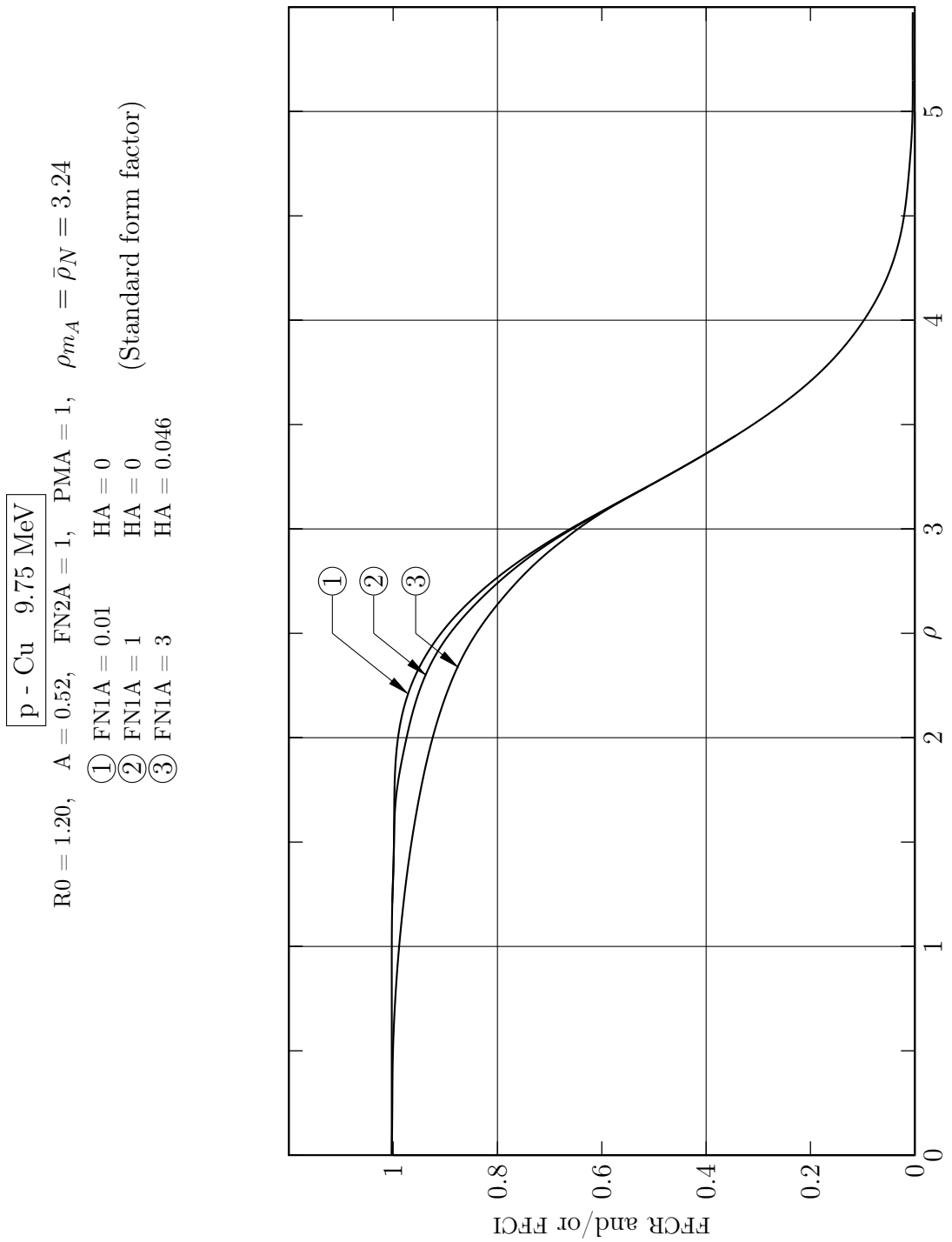


FIG. 4 – Central form factors, knee variation.

3. Final Formulation for Machine Calculation

The complex radial wave function $\Psi_\ell^\pm(\rho)$ may be written as

$$\Psi_\ell^\pm(\rho) = x_\ell^\pm(\rho) + iy_\ell^\pm(\rho) \tag{95}$$

and equation (79) for $\vec{\sigma}\cdot\vec{\ell} = \ell$ or $-\ell-1$ can now be separated into two real coupled differential equations, and dropping the subscripts and superscripts for convenience:

$$\left.\begin{aligned} \frac{d^2x}{d\rho^2} &= px - qy \\ \frac{d^2y}{d\rho^2} &= qx + py \end{aligned}\right\} \tag{96}$$

where

$$\left.\begin{aligned} p &= U_{\rm CR} + U_{\rm SR}\begin{pmatrix} \ell \\ \text{or} \\ -\ell-1 \end{pmatrix} + \frac{\ell(\ell+1)}{\rho^2} \\ q &= U_{\rm CI} + U_{\rm SI}\begin{pmatrix} \ell \\ \text{or} \\ -\ell-1 \end{pmatrix} \end{aligned}\right\} \tag{97}$$

Formulas (97) are convenient for programming purposes as the U's are now independent of ℓ, indeed:

$$\left.\begin{aligned} U_{\rm CR} &= -1 - \frac{V}{E}f_{\rm CR} + \frac{\eta}{\bar\rho_c}\left(3 - \frac{\rho^2}{\bar\rho_c^2}\right) && \text{for } \rho \le \bar\rho_c \\ &= -1 - \frac{V}{E}f_{\rm CR} + \frac{2\eta}{\rho} && \text{for } \rho \ge \bar\rho_c \end{aligned}\right\} \tag{98}$$

$$U_{\rm CI} = -\frac{W}{E}f_{\rm CI} \tag{99}$$

$$\left.\begin{aligned} U_{\rm SR} &= \frac{V_S}{E}\frac{2k}{a}f_{\rm SR} - 0.004926\frac{\eta E}{\bar\rho_c^3} && \text{for } \rho \le \bar\rho_c \\ &= \frac{V_S}{E}\frac{2k}{a}f_{\rm SR} - 0.004926\frac{\eta E}{\rho^3} && \text{for } \rho \ge \bar\rho_c \end{aligned}\right\} \tag{100}$$

$$U_{\rm SI} = \frac{W_S}{E}\frac{2k}{a}f_{\rm SI} \tag{101}$$

4. Numerical Integration

Equations (96) must be integrated numerically twice for each $\ell = 0$ to $\ell_{\max}$ where $\ell_{\max+1}$ corresponds to a partial wave negligibly disturbed by the scattering.

The method chosen for numerical integration is the 3-point Runge-Kutta method: it lends itself to easy starting, permits one to change the interval quite easily and gives excellent accuracy with relatively large steps.

p - Cu 9.75 MeV

R0 = 1.2, A = 0.52, FN1A = 3, PMA = 1, $\rho_{m_A} = \bar{\rho}_N = 3.24$

① FN2A = 0.01
② FN2A = 1 (Standard derivative form factor)
③ FN2A = 3
④ FN2A = 10

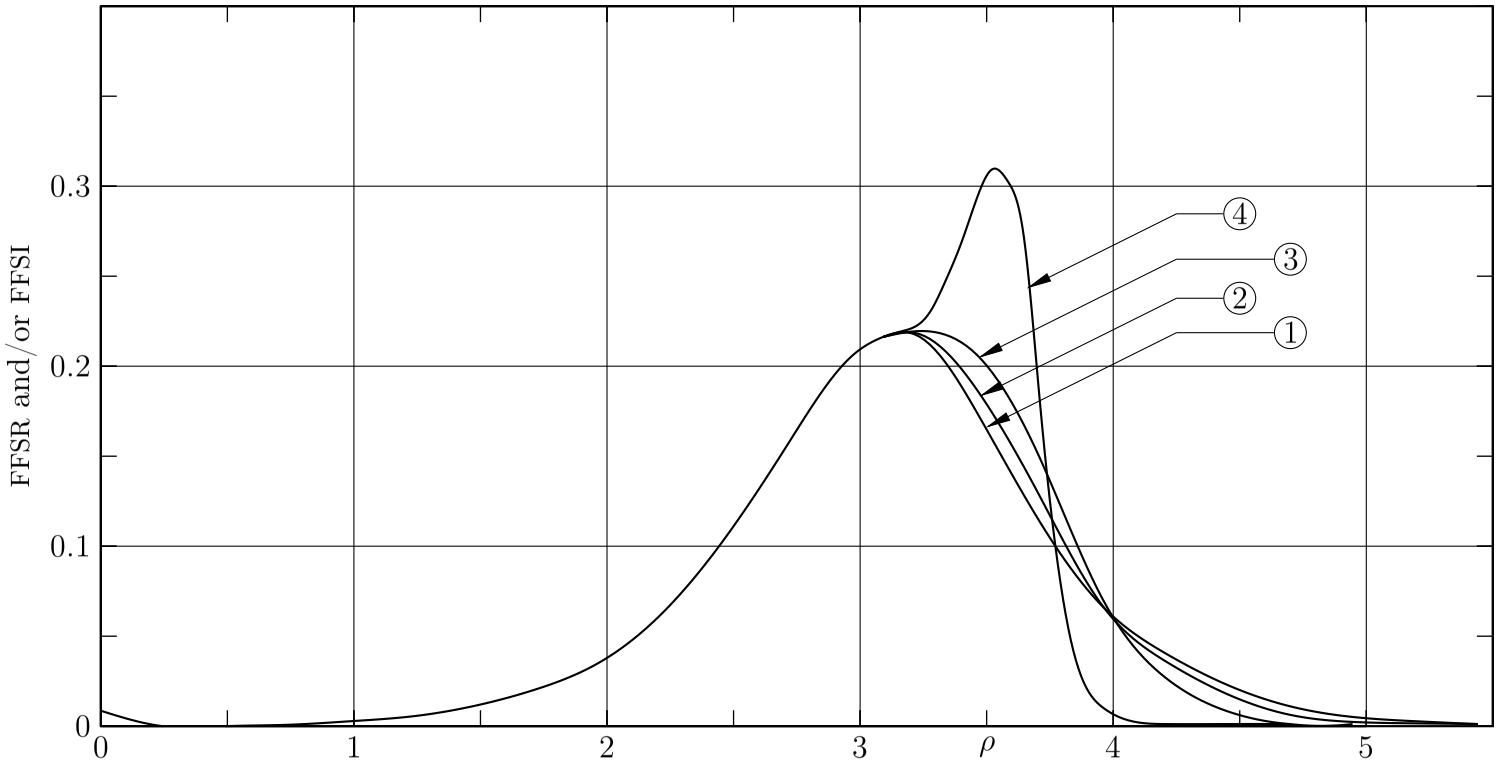


FIG. 5 – Spin-orbit form factor (derivative form). Tail variation.

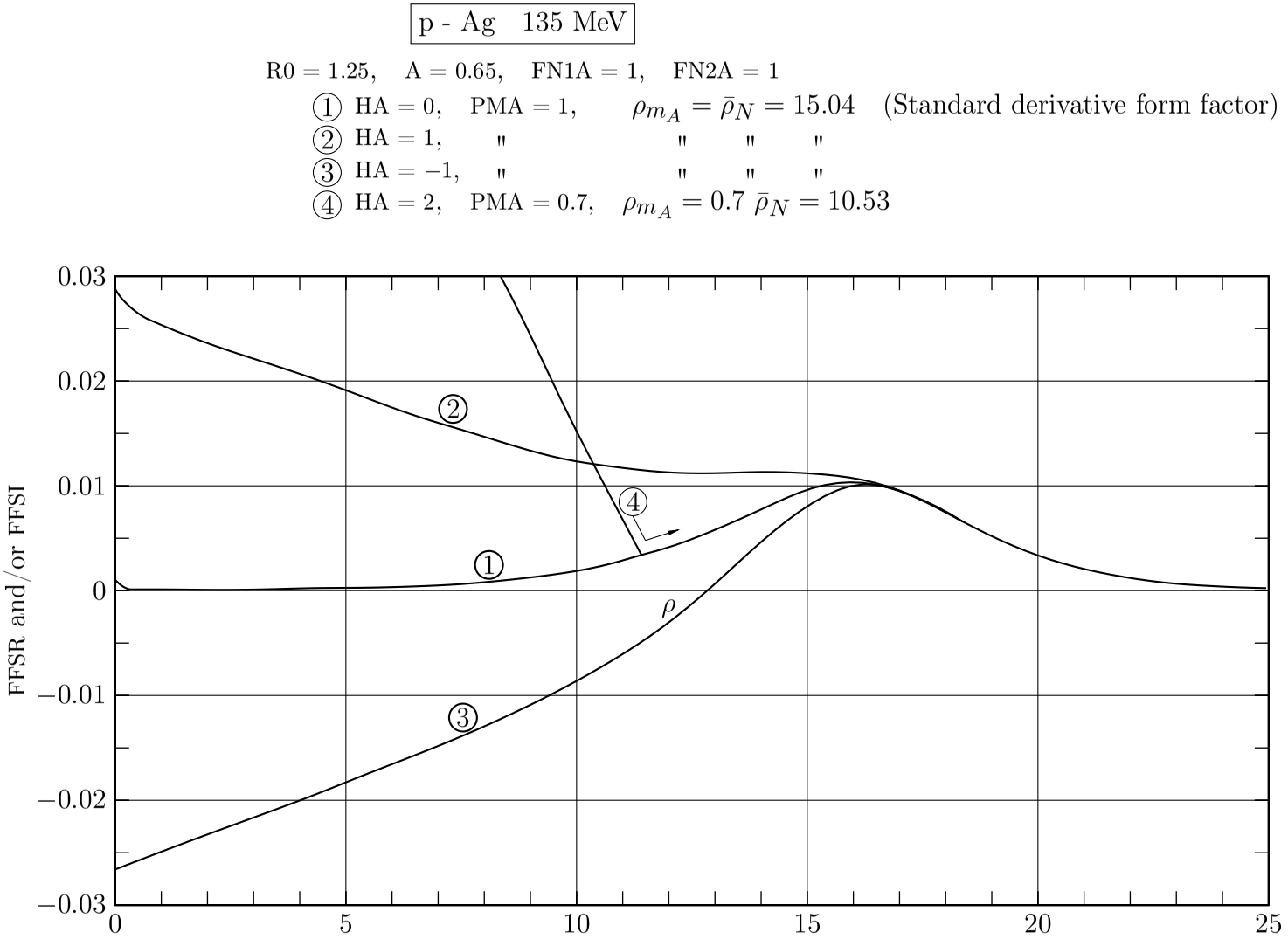


FIG. 6 – Spin-orbit form factors (derivative form). Effect of dips and rises in central form factor.

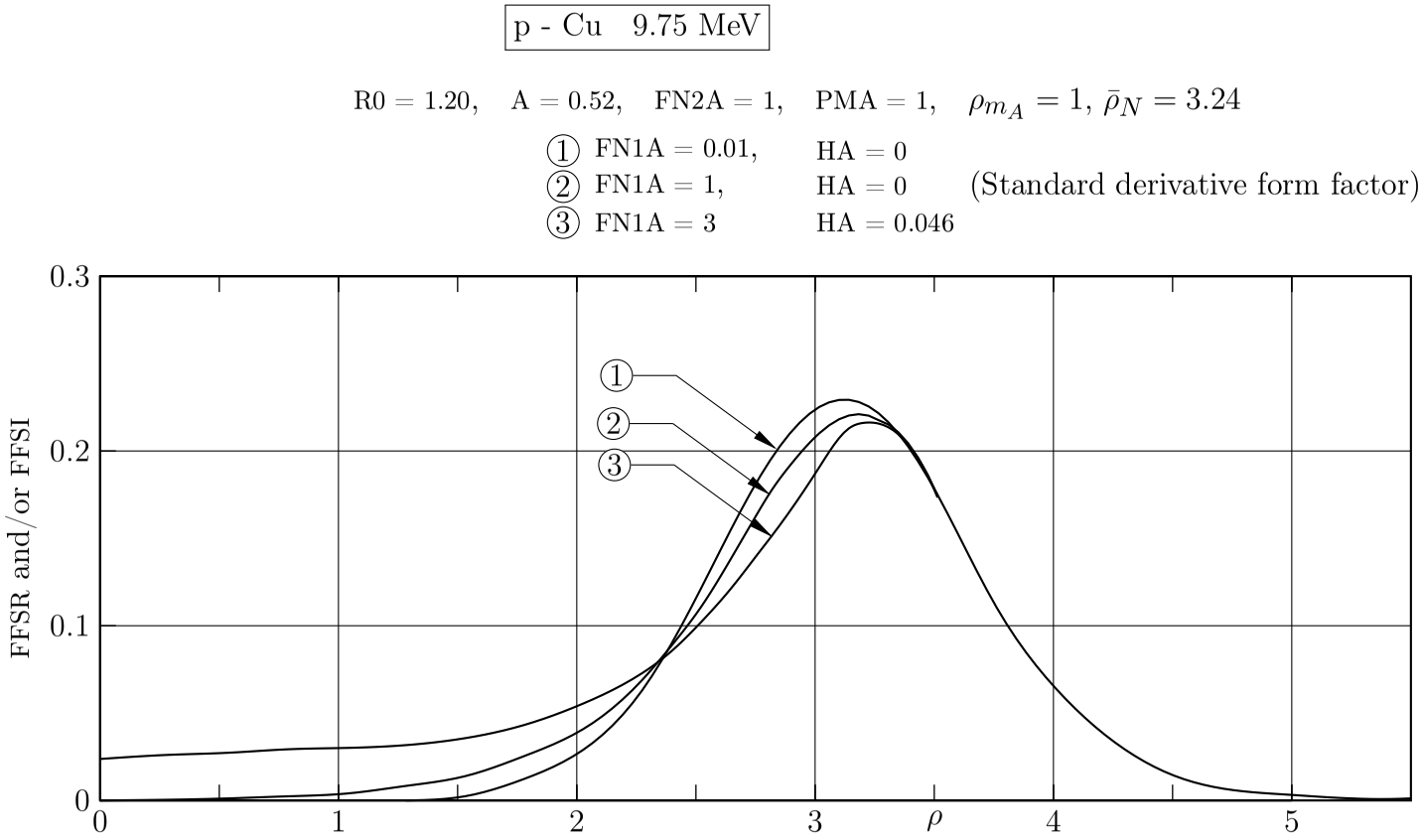


FIG. 7 – Spin-orbit form factor (derivative form). Knee variation.

Given x_{i1}, y_{i1}, $\dot{x}_{i1}$, $\dot{y}_{i1}$, at ρ_i, where $\dot{x}_{i1} \equiv \left(\frac{dx}{d\rho}\right)_{i,1}$ etc.

$$\ddot{x}_{i1} = f(x_{i1}, y_{i1}, \rho_i); \quad \ddot{y}_{i1} = g(x_{i1}, y_{i1}, \rho_i) \tag{102}$$

$$x_{i2} = x_{i1} + \dot{x}_{i1}\frac{\Delta\rho}{2}; \quad y_{i2} = y_{i1} + \dot{y}_{i1}\frac{\Delta\rho}{2} \tag{103}$$

$$\ddot{x}_{i2} = f(x_{i2}, y_{i2}, \rho_i + \frac{\Delta\rho}{2}); \quad \ddot{y}_{i2} = g(x_{i2}, y_{i2}, \rho_i + \frac{\Delta\rho}{2}) \tag{104}$$

$$x_{i3} = x_{i2} + \ddot{x}_{i1}\frac{(\Delta\rho)^2}{4}; \quad y_{i3} = y_{i2} + \ddot{y}_{i1}\frac{(\Delta\rho)^2}{4} \tag{105}$$

$$\ddot{x}_{i3} = f(x_{i3}, y_{i3}, \rho_i + \frac{\Delta\rho}{2}); \quad \ddot{y}_{i3} = g(x_{i3}, y_{i3}, \rho_i + \frac{\Delta\rho}{2}) \tag{106}$$

$$x_{i4} = x_{i2} + \dot{x}_{i1}\frac{\Delta\rho}{2} + \ddot{x}_{i2}\frac{(\Delta\rho)^2}{2}; \quad y_{i4} = y_{i2} + \dot{y}_{i1}\frac{\Delta\rho}{2} + \ddot{y}_{i2}\frac{(\Delta\rho)^2}{2} \tag{107}$$

$$\ddot{x}_{i4} = f(x_{i4}, y_{i4}, \rho_i + \Delta\rho); \quad \ddot{y}_{i4} = g(x_{i4}, y_{i4}, \rho_i + \Delta\rho) \tag{108}$$

and finally

$$x_{i+1,1} = x_{i1} + \Delta x_i = x_{i1} + \frac{(\Delta\rho)^2}{6}(\ddot{x}_{i1} + \ddot{x}_{i2} + \ddot{x}_{i3}) + \Delta\rho\, \dot{x}_{i1} \tag{109}$$

$$\dot{x}_{i+1,1} = \dot{x}_{i1} + \Delta\dot{x}_i = \dot{x}_{i1} + \frac{\Delta\rho}{6}(\ddot{x}_{i1} + 2\ddot{x}_{i2} + 2\ddot{x}_{i3} + \ddot{x}_{i4}) \tag{110}$$

$$y_{i+1,1} = y_{i1} + \Delta y_i = y_{i1} + \frac{(\Delta\rho)^2}{6}(\ddot{y}_{i1} + \ddot{y}_{i2} + \ddot{y}_{i3}) + \Delta\rho\dot{y}_{i1} \tag{111}$$

$$\dot{y}_{i+1,1} = \dot{y}_{i1} + \Delta\dot{y}_i = \dot{y}_{i1} + \frac{\Delta\rho}{6}(\ddot{y}_{i1} + 2\ddot{y}_{i2} + 2\ddot{y}_{i3} + \ddot{y}_{i4}) \tag{112}$$

The process is continued until the nuclear potential becomes negligible at which time the wave functions and their first derivatives must be saved for later matching with those of the coulomb function.

Starting values: If ρ_{initial} is very small, the following starting values may be used:

$$\left.\begin{aligned} x_\ell(\rho = \rho_{\text{initial}}) = (\Delta\rho_1)^{\ell+1}; \quad \dot{x}_\ell(\rho = \rho_{\text{initial}}) = (\ell+1)(\Delta\rho_1)^\ell \\ y_\ell(\rho = \rho_{\text{initial}}) = 0; \quad \dot{y}_\ell(\rho = \rho_{\text{initial}}) = 0 \end{aligned}\right\} \tag{113}$$

5. Coulomb Functions

The regular and irregular coulomb functions are given by the following asymptotic formulas which may be used successfully for large values of ρ:

$$\left.\begin{aligned} F_0 &\sim \sin[\mathrm{Re}(\varphi_0)]e^{-\mathrm{Im}(\varphi_0)} \\ F_1 &\sim \sin[\mathrm{Re}(\varphi_1)]e^{-\mathrm{Im}(\varphi_1)} \\ G_0 &\sim \cos[\mathrm{Re}(\varphi_0)]e^{-\mathrm{Im}(\varphi_0)} \\ G_1 &\sim \cos[\mathrm{Re}(\varphi_1)]e^{-\mathrm{Im}(\varphi_1)} \end{aligned}\right\} \tag{114}$$

where

$$\left.\begin{aligned}\varphi_0 &= \rho - \eta\,\ell n\,2\rho + \sigma_0 + \sum_{k=2}^{\infty} \frac{a_k}{\rho^{k-1}} \left(\frac{1}{1-k}\right) \\ \varphi_1 &= \rho - \eta\,\ell n\,2\rho + \sigma_1 - \frac{\pi}{2} + \sum_{k=2}^{\infty} \frac{b_k}{\rho^{k-1}} \left(\frac{1}{1-k}\right)\end{aligned}\right\} \tag{115}$$

and where

$$\left.\begin{aligned} a_1 = -\eta, \quad a_2 &= \frac{-\eta^2}{2} + i\eta \\ b_1 = -\eta, \quad b_2 &= -\frac{2+\eta^2}{2} + i\frac{\eta}{2} \\ a_k &= -\left(\frac{1}{2}\sum_{m=1}^{k-1} a_m\, a_{k-m}\right) - i\frac{k-1}{2}\, a_{k-1}\end{aligned}\right\} \tag{116}$$

with a similar recurrence formula holding for b_k

$$\left.\begin{aligned}\sigma_0 &= \arg\Gamma(1+i\eta) \\ \sigma_1 &= \sigma_0 + \tan^{-1}\eta\end{aligned}\right\} \tag{117}$$

Furthermore the quantity σ_0 may be successfully approximated over the whole range of η by the following formula:

$$\sigma_0 = -\eta + \left(\frac{\eta}{2}\right)\ln(\eta^2+16) + \frac{7}{2}\tan^{-1}\left(\frac{\eta}{4}\right) - \left[\tan^{-1}\eta + \tan^{-1}\left(\frac{\eta}{2}\right) + \tan^{-1}\left(\frac{\eta}{3}\right)\right] \tag{118}$$

$$- \frac{\eta}{12(\eta^2+16)}\left[1 + \frac{1}{30}\,\frac{\eta^2-48}{(\eta^2+16)^2} + \frac{1}{105}\,\frac{\eta^4 - 160\eta^2 + 1280}{(16+\eta^2)^4}\right].$$

The above formulas which can of course be generalized for any value of ℓ are equivalent though not formally identical to the formulas listed by Abramowitz[5] and by Fröberg[6].

Rather than use these formulas for obtaining F_ℓ and G_ℓ for any value of $\ell > 1$, it is preferable to make use of recurrence formulas.

The following upward recurrence formula is suitable for finding G_ℓ:

$$G_{\ell+1} = \frac{(2\ell+1)\left[\eta + \frac{\ell(\ell+1)}{\rho}\right] G_\ell - (\ell+1)\left[\ell^2+\eta^2\right]^{1/2} G_{\ell-1}}{\ell\left[(\ell+1)^2+\eta^2\right]^{1/2}}. \tag{119}$$

[5]Tables of Coulomb Wave Functions, Vol. I, National Bureau of Standards, Applied Mathematics Series 17, Washington, 1952, p. XV.

[6]C. E. Fröberg, Rev. Mod. Phys. **27**, 399 (1955).

A similar recurrence relation can only be used for downward recurrence on the F_ℓ's, otherwise results rapidly lose all significance. This may be done by means of a method due to Stegun and Abramowitz[7] and which is essentially as follows.

Let it be required to compute F_ℓ from $\ell = 0$ to $\ell = \ell_{\max}$.

(1) Let $\ell^{(1)} = \ell_{\max} + 10$

(The number 10 is arbitrary but has found satisfactory from practical experience)

Let $F^{(1)}_{\ell^{(1)}+1} = 0$ and $F^{(1)}_{\ell^{(1)}} = 0.1$. Successive values of $F^{(1)}_\ell$ can be computed from $\ell = 0$ to $\ell = \ell^{(1)} - 1$ by means of the downward recurrence formula:

$$F^{(1)}_{\ell-1} = \frac{(2\ell+1)\left[\eta + \frac{\ell(\ell+1)}{\rho}\right] F^{(1)}_\ell - \ell\left[(\ell+1)^2 + \eta^2\right]^{1/2} F^{(1)}_{\ell+1}}{(\ell+1)\left[\ell^2+\eta^2\right]^{1/2}}. \tag{120}$$

Letting the constant

$$\alpha = (F^{(1)}_0 G_1 - F^{(1)}_1 G_0)(1+\eta^2)^{1/2} \tag{121}$$

one may compute successively

$$F_\ell = F^{(1)}_\ell \alpha^{-1} \tag{122}$$

for $\ell = \ell_{\max} + 1$ to $\ell = 0$.

(2) To verify the accuracy of the F_ℓ's obtained above one may compute as above a new set of functions $F^{(2)}_\ell$ starting perhaps from $\ell^{(2)} = \ell^{(1)} + 5$ (again the number 5 is obtained from practical experience) and letting now $F^{(2)}_{\ell^{(2)}+1} = 0$, $F^{(2)}_{\ell^{(2)}} = 0.1$. This yields a new set of F_ℓ's.

(3) Comparison of the two sets of F_ℓ's obtained in (1) and (2) above indicates the accuracy of the computation. If this proves insufficient, let $\ell^{(3)} = \ell^{(2)} + 5$ and starting from $F^{(3)}_{\ell^{(3)}+1} = 0$, $F^{(3)}_{\ell^{(3)}} = 0.1$ one may obtain a third set set of F_ℓ's which is to be compared with the second set.

This procedure may be continued until two successive sets of F_ℓ's are found to agree. The derivatives of the coulomb functions may be obtained from the formula

$$Y'_\ell = \frac{\left[\frac{(\ell+1)^2}{\rho} + \eta\right] Y_\ell - \left[(\ell+1)^2+\eta^2\right]^{1/2} Y_{\ell+1}}{(\ell+1)} \tag{123}$$

where Y_ℓ stands for either F_ℓ or G_ℓ.

[7]Stegun and Abramowitz, Phys. Rev. **98**, 1851 (1955).

6. Phase Shifts

The phase shifts are obtained in the usual fashion by matching the logarithmic derivatives of the coulomb functions with those of the numerically integrated functions at a value of ρ sufficiently large so that the nuclear potential becomes negligible.

Matching the logarithmic derivative of the nuclear function $\Psi_\ell = x_\ell + iy_\ell$ with that of its asymptotic form

$$F_\ell + (G_\ell + iF_\ell)C_\ell$$

yields

$$\frac{\Psi'_\ell}{\Psi_\ell} = \frac{F'_\ell + (G'_\ell + iF'_\ell)C_\ell}{F_\ell + (G_\ell + iF_\ell)C_\ell} \tag{124}$$

which lead to

$$C_\ell^\pm = \frac{\Psi_\ell^\pm F'_\ell - \Psi_\ell^{\pm\prime} F_\ell}{\Psi_\ell^{\pm\prime} G_\ell - \Psi_\ell^\pm G'_\ell + i(\Psi_\ell^{\pm\prime} F_\ell - \Psi_\ell^\pm F'_\ell)} \tag{125}$$

the quantities C_ℓ being related to the complex phase shifts through equation (57).

7. Cross Section and Polarization

The differential elastic scattering cross section $\sigma(\theta)$ and the polarization $P(\theta)$ for an unpolarized incident beam are obtained from equations (34) and (35) while the reaction cross section may be obtained as follows.

$$\sigma_R = \frac{N_{\text{abs}}}{N_{\text{inc}}} \tag{126}$$

where N_{abs} is the absorbed flux, and N_{inc} is the incident flux which was assumed to be 1 (see equation (7)). By definition,

$$N_{\text{abs}} = -\frac{\hbar}{2i\mu} \int \left[\Psi^\dagger_{\text{total}} \frac{\partial \Psi_{\text{total}}}{\partial r} - \Psi_{\text{total}} \frac{\partial \Psi^\dagger_{\text{total}}}{\partial r} \right] r_0^2 \sin\theta \, d\theta \, d\varphi \tag{127}$$

where the integral is taken over the surface of a large sphere of radius $r = r_0$. Substituting equation (51) for Ψ_{total} into equation (127) and making use of the orthonormality of the $\mathscr{Y}^{m_j}_{j,\ell,s}$'s and of the relation

$$\left|a_{1/2}\right|^2 + \left|a_{-1/2}\right|^2 - 1, \tag{128}$$

yields after carrying out the surface integration:

$$\sigma_R = N_{\text{abs}} = \frac{4\pi}{V} \sum_{\ell=0}^{\infty} (\ell+1) \left\{ r^2 \left(-\frac{\hbar}{2i\mu} \right) \left[\frac{\Psi_\ell^{+*}}{kr} \frac{\partial}{\partial r} \left(\frac{\Psi_\ell^+}{kr} \right) - \frac{\Psi_\ell^+}{kr} \frac{\partial}{\partial r} \left(\frac{\Psi_\ell^{+*}}{kr} \right) \right] \right\}_{r=r_0} \tag{129}$$

$$- \frac{4\pi}{V} \sum_{\ell=0}^{\infty} \ell \left\{ r^2 \left(-\frac{\hbar}{2i\mu} \right) \left[\frac{\Psi_\ell^{-*}}{kr} \frac{\partial}{\partial r} \left(\frac{\Psi_\ell^-}{kr} \right) - \frac{\Psi_\ell^-}{kr} \frac{\partial}{\partial r} \left(\frac{\Psi_\ell^{-*}}{kr} \right) \right] \right\}_{r=r_0}$$

Now substituting the asymptotic form (52) for $\Psi_\ell^\pm$ and making use of the Wronskian relations

$$G_\ell F_\ell' - F_\ell G_\ell' = 1 \tag{130}$$

we are led to the following:

$$\frac{4\pi}{V}\left\{r^2\left(-\frac{\hbar}{2i\mu}\right)\left[\frac{\Psi_\ell^{\pm *}}{kr}\frac{\partial}{\partial r}\left(\frac{\Psi_\ell^\pm}{kr}\right) - \frac{\Psi_\ell^\pm}{kr}\frac{\partial}{\partial r}\left(\frac{\Psi_\ell^{\pm *}}{kr}\right)\right]\right\}_{r=r_0} = \frac{4\pi}{k^2}\left[\mathrm{Im}(C_\ell^\pm) - |C_\ell^\pm|^2\right]. \tag{131}$$

Finally, substitution of (131) into (129) yields

$$\sigma = \frac{4\pi}{k^2}\sum_{\ell=0}^{\infty}\Big\{(\ell+1)\Big[\mathrm{Im}(C_\ell^+) - \big(\mathrm{Im}(C_\ell^+)\big)^2 - \big(\mathrm{Re}(C_\ell^+)\big)^2\Big] + \ell\Big[\mathrm{Im}(C_\ell^-) - \big(\mathrm{Im}(C_\ell^-)\big)^2 - \big(\mathrm{Re}(C_\ell^-)\big)^2\Big]\Big\}. \tag{132}$$

Note: The quantities $e^{2i\sigma_\ell}$ appearing in equation (60) may be obtained by the following recurrence formulas:

$$\begin{aligned}\mathrm{Re}(e^{2i\sigma_{\ell+1}}) = \cos 2\sigma_{\ell+1} &= \left[\frac{(\ell+1)^2-\eta^2}{(\ell+1)^2+\eta^2}\cos 2\sigma_\ell\right] - \left[\frac{2\eta(\ell+1)}{(\ell+1)^2+\eta^2}\sin 2\sigma_\ell\right] \\ \mathrm{Im}(e^{2i\sigma_{\ell+1}}) = \sin 2\sigma_{\ell+1} &= \left[\frac{(\ell+1)^2-\eta^2}{(\ell+1)^2+\eta^2}\sin 2\sigma_\ell\right] + \left[\frac{2\eta(\ell+1)}{(\ell+1)^2+\eta^2}\cos 2\sigma_\ell\right]\end{aligned} \tag{133}$$

while the Legendre polynomials obey the usual relations

$$P_0(\cos\theta) = 1, \quad P_1(\cos\theta) = \cos\theta$$

$$P_{\ell+1}(\cos\theta) = \frac{1}{\ell+1}\left[(2\ell+1)\cos\theta P_\ell(\cos\theta) - \ell P_{\ell-1}(\cos\theta)\right] \tag{134}$$

$$P_\ell^{(1)}(\cos\theta) = \frac{\ell+1}{\sin\theta}\left[\cos\theta P_\ell(\cos\theta) - P_{\ell+1}(\cos\theta)\right]. \tag{135}$$

One may also compute the Rutherford scattering cross section:

$$\sigma_c(\theta) = |f_c(\theta)|^2. \tag{136}$$

8. Chi Square Deviation

Experimental and theoretical quantities may be compared by means of the chi square deviation:

$$\chi_T^2 = \chi_\sigma^2 + \chi_P^2 \tag{137}$$

where

$$\chi_\sigma^2 = \sum_\theta \chi_\sigma^2(\theta) = \sum_\theta \left[\frac{\sigma^{\mathrm{th}}(\theta) - \sigma^{\mathrm{ex}}(\theta)}{\Delta\sigma^{\mathrm{ex}}(\theta)}\right]^2 \tag{138}$$

$$\chi_P^2 = \sum_\theta \chi_P^2(\theta) = \sum_\theta \left[\frac{P^{\mathrm{th}}(\theta) - P^{\mathrm{ex}}(\theta)}{\Delta P^{\mathrm{ex}}(\theta)}\right]^2 \tag{139}$$

where the $\sigma^{\rm th}(\theta)$ and $P^{\rm th}(\theta)$ are the theoretically obtained cross sections and polarizations while $\sigma^{\rm ex}(\theta)$, $\Delta\sigma^{\rm ex}(\theta)$, $P^{\rm ex}(\theta)$, $\Delta P^{\rm ex}(\theta)$ are respectively the experimentally given cross sections, standard deviations in the cross sections, polarization and standard deviations in the polarization.

It should be noted that the constants were chosen such that the differential and reaction cross section will be obtained in units of 10^{-26} cm^2. The polarizations are of course dimensionless ratios.

9. Normalization

The radial wave functions $\Psi_\ell^\pm$ and their derivatives obtained from numerical integration of the radial Schroedinger equation contain an arbitrary normalization factor, $1/M_\ell^\pm$. This factor however does not affect the cross section and polarization since these are obtained from the phase shifts which in turn are obtained from ratios of logarithmic derivatives (see equation (125)) wherein the M_ℓ's cancel out. If on the other hand the normalized radial wave functions and their derivatives are required, the normalization terms may be obtained as follows:

The asymptotic form of $\Psi_\ell^\pm$ must obey equation (52) but improper normalization results in the fact that the calculated wave functions are actually given by

$$x_\ell^\pm(\rho) + iy_\ell^\pm(\rho) = M_\ell^\pm \left\{F_\ell(\eta,\rho) + C_\ell^\pm \left[G_\ell(\eta,\rho) + iF_\ell(\eta,\rho)\right]\right\} \tag{140}$$

Now, for $\rho \le \rho_{\max}$ the nuclear potentials are negligible and equation (52) represents the exact solution; in particular, at $\rho = \rho_{\max}$, we must have

$$x_\ell^\pm(\rho_{\max}) + iy_\ell^\pm(\rho_{\max}) = M_\ell^\pm \left\{F_\ell(\eta,\rho_{\max}) + C_\ell^\pm \left[G_\ell(\eta,\rho_{\max}) + iF_\ell(\eta,\rho_{\max})\right]\right\} \tag{141}$$

whereby

$$M_\ell^\pm = \frac{x_\ell^\pm(\rho_{\max}) + iy_\ell^\pm(\rho_{\max})}{F_\ell(\eta,\rho_{\max}) + C_\ell^\pm \left[G_\ell(\eta,\rho_{\max}) + iF_\ell(\eta,\rho_{\max})\right]} \tag{142}$$

and the normalized radial wave functions and their derivatives are given by

$$\left.\begin{aligned} \Psi_\ell^\pm(\rho) &= \frac{1}{M_\ell^\pm}\left[x_\ell^\pm(\rho) + iy_\ell^\pm(\rho)\right] \\ \frac{d\Psi_\ell^\pm(\rho)}{d\rho} &= \frac{1}{M_\ell^\pm}\left[\dot{x}_\ell^\pm(\rho) + i\dot{y}_\ell^\pm(\rho)\right] \end{aligned}\right\} \tag{143}$$

and the complete normalized wave function is given in equation (51) with $\Psi_\ell^\pm$ as above in equation (143).

Note: During the numerical integration the program may renormalize the wave functions and their derivatives at any value of ρ for which overflow takes place by dividing the functions and their derivatives by the largest of these. This is accompanied by an explicit printout as explained in the description of subroutine RKINT. Such occasional internal renormalization must of course be taken into account if correctly normalized functions are required.

III. Program Description

A. General Description

1. Machine Specifications

Program SCAT 4 has been written for an IBM 704 with floating point traps or an IBM 709, with a 32,768 words memory, no drum and a minimum of two tape units.

The program can probably be modified for a 16K memory by reducing the number of θ's (up to 75 allowed here) and the number of ℓ's (up to 50 allowed here). A large part of the memory (7500 words) is occupied by the Legendre polynomials and this may also be reduced by computing the polynomials as required. Furthermore, the program contains a large number of printouts which may be abbreviated to save storage space.

2. General Program Description

The program was designed to compute cross sections, polarizations and chi square deviations at a number of specified points in the space of the optical model parameters V, W, A, VS, WS, and if needed BG (RO, RC and RG are kept fixed), for a given set of input data.

The time to carry out a run for a single set of parameters depends of course upon the maximum values of ℓ and ρ; for p-Cu at 10 MeV ($\ell_{\max} = 10$, $\rho = 0.0625$ (.0625) 0.50 (0.25) 10.0) a run takes about 45 seconds including about 15 seconds for maximum output to tape.

The program has been written in the form of subroutines to allow easy checking and modification. Some of these subroutines are not yet available, but some provision have been made to include them in the future. The following subroutines written in FORTRAN are specific (sub)routines of the program:

Main routine	–	MAIN4			
Subroutine	–	*CTRL4*			
Subroutine	–	INPT4	Subroutine	–	PGEN4
Subroutine	–	POT1CH	Subroutine	–	INTCTR
Subroutine	–	POP1	Subroutine	–	RKINT
Subroutine	–	SIGZRO	Subroutine	–	*CSUBL*
Subroutine	–	FSUBC	Subroutine	–	AB
Subroutine	–	EXSGML	Subroutine	–	SGSGCP
Subroutine	–	RHOTB	Subroutine	–	SIGMAR
Subroutine	–	*COULFN*	Subroutine	–	*CHISQ*
Subroutine	–	RMXINC	Subroutine	–	OUTPT4

The following subroutines are general utility routines used by the program:

Subroutine - SKIP written in FORTRAN

Subroutine - LEAVE written in FORTRAN

Subroutine - SPILL written in FAP

The following subroutines are used in conjunction with the Load-and-Go system in use at WDPC (Western Data Processing Center, UCLA). The effect of using this system is described in section III-A-3 below.

Subroutine - **SAVE**

Subroutine - PDUMP

Subroutine - EXIT

The program assumes the presence of the following Fortran elementary function subroutines:

LOGF	–	(natural logarithm)
SINF	–	(sine)
COSF	–	(cosine)
EXPF	–	(exponential)
SQRTF	–	(square root)
ATANF	–	(arc tangent)

3. Use of the WDPC Load-and-Go System

Program SCAT 4 has been written for the Load-and-Go system in use at the WDPC, UCLA. This *only* affects it as follows:

(i) Special subroutines of the load-and-go system.

Subroutine SAVE

The purpose of this subroutine is to allow the operator to interrupt the calculation without loss. The program is normally run with Sense Switch 1 off; turning on Sense Switch 1 will cause the program to call **SAVE** after completing the innermost **DO** loop of subroutine *CTRL4*. **SAVE** then writes on tape the content of the core memory as well as all other information required to continue the computation such as the contents of the AC, MQ, index registers, etc. . . .

A restart routine will then later reload the core memory, reset all registers etc. . . , and return right after the *CALL SAVE* statement. The following statements up to statement number 66 are then required to properly position the input data tape as the latter was probably rewound when the computation was interrupted.

To eliminate the use of subroutine **SAVE**, remove from subroutine *CTRL4* all statements from statement number 118 to statement number 66 inclusive.

Subroutine PDUMP(α,β)

The purpose of this subroutine is to provide a partial core dump of all quantities between the location of the arguments in the call statement. Subroutine PDUMP is called by subroutine LEAVE whenever difficulties such as overflow or division by zero take place.

To eliminate subroutine PDUMP, replace in subroutine LEAVE the statement *CALL* PDUMP(A,ZZ) by whatever statements will cause the required core dump.

Subroutine EXIT

This subroutine terminates the program.

To eliminate subroutine EXIT, replace statement number 151 in subroutine INPT4 by whatever statement will be used to terminate the program.

(ii) **END** Statements.

The usual `FORTRAN` **END** statements do not appear in the program as the load-and-go system provides them automatically.

(iii) Input and Output Statements.

In conjunction with the load-and-go system, the program is input from tape, while the input data is brought in from tape 7 and all the output is to tape 6.

All these particular features can of course be easily modified to use the program either directly or in conjunction with any other system.

4. Error Indications:

(i) Division by zero.

Every division which could conceivably have a zero divisor either because of the range of numbers used or because of an error in the input data is followed by an **IF** DIVIDE CHECK. Detection of a zero denominator is then followed by an explicit print out and a *CALL LEAVE* statement which leads to the next set of input data. In order to be sure that no division by zero remains undetected, every subroutine which contains an **IF** DIVIDE CHECK statement also begins with an **IF** DIVIDE CHECK to verify that the trigger is off at the start of the subroutine; if the divide check trigger is found on at the start, there is an explicit printout to that effect followed by a *CALL LEAVE* statement.

(ii) Overflow. Underflow.

Overflow and underflow are monitored by subroutine SPILL (JSPILL, ISPILL, x, y) which needs only be called once by MAIN4. When SPILL is called, it replaces the quantities JSPILL and ISPILL by zeros. Thereafter, in case of overflow (underflow) the subroutine replaces the overflowed (underflowed) quantity with x (y) and places into JSPILL (ISPILL) the address of the command which caused overflow (underflow) to occur for the first time. Program `SCAT 4` uses $x = y = 0$.

Every subroutine in which computations are carried out starts by setting ISPILL and JSPILL equal to zero to insure correct identification of possible subsequent overflow or underflow. The subroutine then ends with a check of ISPILL and JSPILL. If either of these is not zero, there is an explicit printout describing the overflow or underflow. Underflow results therefore in substituting zero for the underflowed quantity, but the computation

proceeds. Overflow on the other hand results in substituting zero for the overflowed quantity and leads to a *CALL LEAVE* statement to stop the computation.

B. Detailed Descriptions of the Specific Routines of the Program

MAIN4

The main routine which is only used at the start of the program carries out the following steps:

1) Calls SPILL which controls overflow and underflow (see III-A-4-ii). One such call statement is sufficient to put SPILL in permanent control for all subroutines.

2) Sets up EPS1, EPS2, EPS3, which are constants used to control the accuracy of the Coulomb functions computations, and EPS4 which is used in subroutine POT1CH.

3) Inputs identification and program numbers.

4) Calls *CTRL4*.

***CTRL4* (Control 4)**

This subroutine controls the whole flow of the program. It was coded as a subroutine to allow it to be called by subroutine LEAVE. It carries out the following steps:

1) Advances group identification and resets run identification numbers.

2) Call INPT4.

3) Calls POT1CH.

4) If KTRL(5) = 1, calls POP1
   if KTRL(5) = 0, proceeds.

5) Calls SIGZRO, FSUBC, EXSGML.

6) Sets up five (or six) nested **DO** loops for varying successively V, W, a, V_s, W_s (and b for a surface absorption potential). The following steps are always done within the innermost **DO** loop:

   a) If Sense Switch 1 is on, calls **SAVE**
      if Sense Switch 1 is off, proceeds.
   b) Advances run identification number.
   c) Calls RHOTB, *COULFN*, RMXINC, PGEN4, INTCTR, *CSUBL*, AB, SGSGCP, SIGMAR.
   d) If KTRL(2) = 0, proceeds
      if KTRL(2) = 1, calls *CHISQ*.
   e) Calls OUTPT4.

7) When all the **DO** loops have been completed, returns to step 1.

INPT4 (Input 4)

1) Inputs KTRL(1); if KTRL(1) = 100, calls EXIT
   if KTRL(1) ≠ 100, proceeds.

2) Inputs KTRL(I), I = 2 **to** 13.

3) Inputs FMI, FMB, ELAB, ZZ, RC, V, W, RO, A, VS, WS, RG, BG, DV, DW, DA, DVS, DWS, DBG, HA, PMA, FN1A, FN2A, HB, PMB, FN1B, FN2B, NVMAX, NWMAX, NAMAX, NVSMAX, NWSMAX, NBGMAX.

4) Sets up TV = V to TBG = BG (starting values of the parameters).

5) Inputs NMAX, forms NMAXP = NMAX−1.

6) Inputs RHOIN(I), I = 1 **to** NMAX and DRHOIN(I), I = 1 **to** NMAXP.

7) Computes FMU as per equation (5)
   Computes ECM as per equation (6)
   Computes FKAY as per equation (8)
   Computes RHOBN as per equation (73)
   Computes RMA and RMB (see Glossary, under PMA, PMB)
   Computes RHOBC as per equation (74)
   Computes ETA as per equation (43).

8) Inputs LMAXM, forms IMAX = LMAXM + 1.

9) Sets IIN(J) = 1, J = 1 **to** LMAX (see description of subroutine INTCTR)

10) If KTRL(5) = 0, proceeds
    if KTRL(5) ≠ 0:a) inputs JMAX
    b) inputs THETAD(I), I = 1 **to** JMAX
    c) computes THETA(I), I = 1 **to** JMAX.

11) If KTRL(2) = 0 and/or KTRL(3) = 0, proceeds,
    if KTRL(2) ≠ 0 and KTRL(3) ≠ 0, inputs
    SGMARX(I), DSGMEX(I), POLEX(I), DPOLEX(I), I = 1 **to** JMAX.

12) Returns to *CTRL4*.

POT1CH (potential 1 check)

The purpose of this subroutine is to check whether $\ell_{\max}$ is sufficiently large so that all the partial waves sensibly affected by the potential are included and to check whether $\rho_{\max}$ (the point at which the coulomb functions will be matched to the nuclear wave functions) is sufficiently large to insure that the non-coulomb part of the potential is negligible. If $\ell_{\max}$ and/or $\rho_{\max}$ are too small, the subroutine increases them, and sets IIN($\ell_{\max}$)= 1. The quantities $\rho_{\max}$ and $\ell_{\max}$ may be checked or not according to the value assigned to KTRL(13):

KTRL(13) = 1: check both $\ell_{\max}$ and $\rho_{\max}$
KTRL(13) = 2: check $\rho_{\max}$ only
KTRL(13) = 3: check $\ell_{\max}$ only
KTRL(13) = 4: do not check either.

$\rho_{\max}$ and $\ell_{\max}$ are checked in various ways depending upon the potential form. The routine operates as follows:

1) The routine first calculates the maximum values of V, W, A, VS, WS, and, in the case of a Gaussian absorption, of BG over the specified grid of these parameters.

2) If KTRL(1) = 0, standard potential (or variation thereof), the routine checks, if required, that:

   a) $\rho_{\max}$ is sufficiently large so that

$$\frac{(V^2+W^2)^{1/2}}{E}\,\frac{1}{(1+e^{(\rho_{\max}-\bar{\rho}_N)/ka})} \le \epsilon_4. \tag{144}$$

   If this condition is not met, $\rho_{\max}$ is increased by the last value of $\Delta\rho$ and the check is repeated. This is accompanied by the print out:
   RHOIN(NMAX) = (value of old $\rho_{\max}$) + (last value of DRHOIN)
   RHOIN(NMAX) IS TOO SMALL IN NUCLEAR POTENTIAL.

   b) The routine also checks, if required, that $\ell_{\max}$ is sufficiently large so that

$$\frac{\sqrt{V^2+W^2}}{E}\,\frac{1}{(1+e^{(\ell_{\max}-\bar{\rho}_N)/ka})} \le \epsilon_4. \tag{145}$$

   If this condition is not met, $\ell_{\max}$ is increased by 1 and the check is repeated; this is accompanied by the following printout:
   LMAXM = (value of old LMAXM) + 1
   LMAXM TOO SMALL BECAUSE OF CENTRAL POTENTIAL.
   The routine then checks that $\ell_{\max}$ is sufficiently large so that

$$2k^2\frac{\sqrt{V_S^2+W_S^2}}{E}\,\frac{1}{(1+e^{(\ell_{\max}-\bar{\rho}_N)/ka})} \le \epsilon_4. \tag{146}$$

   If this condition is not met, $\ell_{\max}$ is increased by 1 and the check is repeated; this is accompanied by the following printout:
   LMAXM = (value of old LMAXM) + 1
   LMAXM TOO SMALL BECAUSE OF SPIN ORBIT POTENTIAL.

3) If KTRL(1) = 1, Gaussian absorption,

   a) The check on $\rho_{\max}$ is as follows:

$$\frac{V}{E}\,\frac{1}{(1+e^{(\rho_{\max}-\bar{\rho}_N)/ka})} \le \epsilon_4; \tag{147}$$

and

$$\frac{W}{E}\, e^{-(\rho_{\max}-\bar{\rho}_G/kb)^2} \leq \epsilon_4. \tag{148}$$

If these conditions are not met $\rho_{\max}$ is increased as before and the checks are repeated; this is accompanied by the same printout as above.

b) The check on $\ell_{\max}$ is as follows:

$$\frac{V}{E}\,\frac{1}{(1+e^{(\ell_{\max}-\bar{\rho}_N)/ka})} \leq \epsilon_4; \tag{149}$$

and

$$\frac{W}{E}\, e^{-(\ell_{\max}-\bar{\rho}_G/kb)^2} \leq \epsilon_4 \tag{150}$$

and as in equation (146).

If these conditions are not met $\ell_{\max}$ is increased by 1 and the checks repeated. The prints-out are given on the previous page.

4) If KTRL(1) = 2, Square well

a) The check on $\rho_{\max}$ is as follows

$$\rho_{\max} > \bar{\rho}_N \tag{151}$$

b) The check on $\ell_{\max}$ is as follows

$$\ell_{\max} > \bar{\rho}_N + 3. \tag{152}$$

Failure to meet these conditions leads to increases in $\rho_{\max}$ and/or $\ell_{\max}$ accompanied by the same printouts as given above, after which the checks are repeated.

The program uses EPS4 = 0.001. This quantity is specified in the MAIN4 routine.

The checks described above are based on a rough estimate of the phase shifts using a WKB expression.

POP1

Computes P(L,J), PP(L,J), L = 1 **to** LMAXP, J = 1 **to** JMAX as per equations (134) and (135) and returns to *CTRL4*.

SIGZRO (Sigma zero)

Computes SIGMA0 and SIGMA1 as per equations (117) and (118) and returns to *CTRL4*.

FSUBC

Computes FCR(J) and FCI(J), J = 1 **to** JMAX as per equation (47) and returns to *CTRL4*.

EXSGML (Exponential sigma ℓ)

Computes EXSGMR(J), EXSGMI(J) for J = 1 **to** LMAX as per equation (133) and returns to *CTRL4*.

RHOTB (Rho tabulation)

The purpose of this subroutine is to construct a table of ρ's and $\Delta\rho$'s corresponding to each step of the numerical integration. This table is formed from the arrays of RHOIN(I) and DRHOIN(I) which are input by subroutine INPT4

INPUT ARRAYS

RHOIN(I)	DRHOIN(I)
RHOIN(1)	DRHOIN(1)
RHOIN(2)	DRHOIN(2)
.	.
.	.
.	.
.	.
RHOIN(NMAX−1)	DRHOIN(NMAX−1)
RHOIN(NMAX)	

COMPUTED TABLES

RHO(I)	DRHO(I)
RHO(1)	DRHO(1)
RHO(2)	DRHO(2)
.	.
.	.
.	.
.	.
RHO(ILAST−1)	DRHO(ILAST−1)
RHO(ILAST)	

ρ= RHOIN(1) (DRHOIN(1)) RHOIN(2) ... (DRHOIN(NMAX−1)) RHOIN(NMAX)
RHO(I+1) = RHO(I) + DRHO(I)
DRHO(1) = DRHO(2) = ⋯ = DRHO(I) = DRHOIN(1)
up **to** RHO(I) = RHOIN(2), etc....
RHO(1) = RHOIN(1); RHO(ILAST) = RHO(NMAX)
ILAST≥NMAX.

If RHOIN(NMAX) is given in such a way that it cannot be reached by an integral number of DRHO(I)'s, the last interval is shortened (up to 50%) or lengthened (by no more than 50%) so that RHO(ILAST) = RHOIN(NMAX).

***COULFN* (Coulomb functions)**

This is the most complex subroutine of the program. It computes the regular and irregular coulomb functions and their derivatives for L = 1 **to** LMAXM at ρ= RHOMAX by means of asymptotic formulas. The main steps are as follows:

1) The a and b series appearing in equation (115) are calculated according to equations (116) and are cut off when either:

   (a) The term N_a (or N_b) is such that the next term exceeds in magnitude the previous one, i.e., when

   $$[\mathrm{Re}(U_{N_a}+1)]^2+[\mathrm{Im}(U_{N_a}+1)]^2 \geq [\mathrm{Re}(U_{N_a})]^2+[\mathrm{Im}(U_{N_a})]^2 \tag{153}$$

   where

   $$U_k = \frac{a_k}{(k-1)\rho_{\max}^{k-1}} \tag{154}$$

   and similarly for the b series.

   (b) The contributions of both the real and imaginary terms give undetectable contributions to the real and imaginary parts of φ_0 (and similarly for φ_1). During these computations, the value of $\rho_{\max}$ may be increased by addition of the last value of DRHOIN and the computation starts all over again under the following condition:

a) The a or b series is identically equal to zero. This is accompanied by the printout:

SERIES IN PHI0 OR PHI1 IS ZERO, CHECK **DATA**, **IF** OK

INCREASE RHOMAX = (value of old RHOMAX) + (value of last DRHOIN)

b) Either of the two series diverges too quickly, i.e., the N_a-th (or N_b-th) term still gives a non-negligible contribution to the series obtained so far, viz.

$$\left| \frac{\left[\mathrm{Re}(U_{N_a})\right]^2 + \left[\mathrm{Im}(U_{N_a})\right]^2}{\left[\mathrm{Re}\left(\sum_{k=2}^{N_a-1} U_k\right)\right]^2 + \left[\mathrm{Im}\left(\sum_{k=2}^{N_a-1} U_k\right)\right]^2} \right| \geq EPS3 \tag{155}$$

(EPS3 is given the value 0.00001 in the MAIN4 routine.)
This is accompanied by the printout:

IF OK A OR B SERIES DIVERGES TOO QUICKLY

INCREASE RHOMAX = (value of old RHOMAX) +(value of last DRHOIN).

c) Over 48 terms are required in either the a or b series. This is accompanied by the printout:

INCREASE RHOMAX = (value of old RHOMAX) + (value of last DRHOIN)

A OR B SERIES CONVERGES TOO SLOWLY.

2) The quantities φ_0, φ_1, F_0, F_1, G_0, G_1 are formed according to equations (114) and (115), and the Wronskian is checked for accuracy requiring that

$$\left| \mathscr{W} - \left[1+\eta^2\right]^{-1/2} \right| = \left| F_0 G_1 - F_1 G_0 - \left[1+\eta^2\right]^{-1/2} \right| \leq EPS1 \tag{156}$$

(EPS1 is given the value 0.00001 in the MAIN4 routine.)

If this condition is violated $\rho_{\max}$ is increased and the computation starts all over again; this is accompanied by the following printout:

INCREASE RHOMAX = (old value of RHOMAX) + (last value of DRHOIN)

BAD INITIAL WRONSKIAN.

3) The regular coulomb functions are formed by downward recurrence as per equations (120) and (122) according to the accompanying description.

Agreement between successive sets of F_ℓ's is verified by checking that

$$\left| (F_\ell^{(n)} / F_\ell^{(n+1)}) - 1 \right| \leq EPS2 \tag{157}$$

(EPS2 is given the value 0.00001 in the MAIN4 routine) for $\ell = 0$ to $\ell_{\max}$.

During this computation the value of $\rho_{\max}$ is increased and the computation starts all over if it turns out that $\ell_{(1)} > \ell_{\max} + 40$. This is accompanied by the printout:

INCREASE RHOMAX = (old value of RHOMAX) + (last value of DRHOIN)

L TOO LARGE IN FBAR(L).

4) The irregular coulomb functions are formed by upward recurrence as per equation (119) and the Wronskian for every $\ell = 0$ to $\ell_{\max} + 1$ is checked for accuracy requiring that

$$\left| F_\ell G_{\ell+1} - F_{\ell+1} G_\ell - \frac{\ell + 1}{\left[(\ell+1)^2 + \eta^2\right]^{1/2}} \right| \leq EPS1 \tag{158}$$

(EPS1 is given the value 0.00001 in the MAIN4 routine.)

If this condition is violated the value of $\rho_{\max}$ is increased and the computation starts all over again; this is accompanied by the printout:
INCREASE RHOMAX = (old value of RHOMAX) + (last value of DRHOIN)
BAD WRONSKIAN FOR L = (value of $\ell + 1$ for which equation (158) failed).

5) Finally the derivatives of the coulomb functions for $\ell = 0$ to $\ell_{\max}$ are formed as per equation (123).

RMXINC (Rho max increase)

The purpose of this subroutine is to extend the table of RHO(I) and DRHO(I) by increments of the last value DRHOIN until the final value of RHO(I) equals RHOMAX which may have been increased by the subroutine *COULFN*.

PGEN4 (Potential generator 4)

The purpose of this subroutine is to form tables of the ℓ-independent parts of the potential corresponding to the RHO(I) tables and suitable for using in the numerical integrations.

These include:

UCRB(I), UCIB(I), USRB(I), USIB(I) for I = 1 **to** ILAST and corresponding to the values at the beginning of an interval of integration; a corresponding table of form factors is also formed:

FFCR(I), FFCI(I), FFSR(I), FFSI(I),
and
UCRM(I), UCIM(I), USRM(I), USIM(I),
and
FFCRM(I), FFCIM(I), FFSRM(I), FFSIM(I) for I = 1 **to** ILAST − 1 corresponding to the values in the middle of an interval of integration.

The original and tightest part of the subroutine corresponds to a standard form factor; modifications have been added to permit use of a variety of form factors briefly described earlier.

The subroutine operates as follows: The UCR–'s are calculated as per equation (98), the UCI–'s as per equation (99), the USR–'s as per equation (100) and the USI–'s as per equation (101), wherein:

(i) KTRL(I) = 0: VOLUME ABSORPTION OR SPECIAL NUCLEAR FORM FACTOR:

If KTRL(7) = 0, f_{CR} is computed as per equation (80); [FFCR][8] = f_{CR}
= 1, f_{CR} is computed as per equation (86); [FFCR] = f_{CR}
= 2, f_{CR} is computed as per equation (87); [FFCR] = f_{CR}

If KTRL(8) = 0, f_{CI} is computed as per equation (80); [FFCI] = f_{CI}
= 1, f_{CI} is computed as per equation (86); [FFCI] = f_{CI}
= 2, f_{CI} is computed as per equation (87); [FFCI] = f_{CI}

If KTRL(9) = 0, f_{SR} is computed as per equation (81); [FFSR] = f_{SR}
= 1, f_{SR} is computed as per equation (91); [FFSR] = f_{SR}/ka
= 2, f_{SR} is computed as per equation (94); [FFSR] = $f_{SR}/2$

If KTRL(10) = 0, f_{SI} is computed as per equation (81); [FFSI] = f_{SI}
= 1, f_{SR} is computed as per equation (91); [FFSI] = f_{SI}/ka
= 2, f_{SR} is computed as per equation (94); [FFSI] = $f_{SI}/2$

(ii) KTRL(1) = 1: GAUSSIAN ABSORPTION

f_{CR} is computed as per equation (80); [FFCR] = f_{CR}
f_{CI} is computed as per equation (82); [FFCI] = f_{CI}
f_{SR} is computed as per equation (81); [FFSR] = f_{SR}
f_{SI} is computed as per equation (81); [FFSI] = f_{SI}

(iii) KTRL(1) = 2: SQUARE WELL

f_{CR} is computed as per equation (84); [FFCR] = f_{CR}
f_{CI} is computed as per equation (84); [FFCI] = f_{CI}
f_{SR} and f_{CI} are taken to be zero.

Furthermore,

If KTRL(11) = 1, USR– are computed as per equation (100) *including* the coulomb spin-orbit term.

If KTRL(11) = 0, USR– are computed as per equation (100) *excluding* the coulomb spin-orbit term, i.e, the second term on the right hand side. KTRL(7) to KTRL(11) can of course be given any combination of permitted values.

INTCTR (Integration Control)

For each value of L = 1 **to** LMAX this subroutine carries out the following steps:

1) Sets up starting values for the numerical integration as per equation (113). The quantities IIN(L) are not especially useful at the present time, but they have been included in order to permit start of the numerical integration at various values of ρ depending on ℓ and thus permitting considerable time saving by foreshortening the numerical integrations. A study of this method is presently under way.

2) Calls RKINT which performs the numerical integration.

[8] FFCR refers to the symbolic variables FFCR(I) and FFCRM(I) appearing in the program (see glossary of symbols), similarly for FFCI, FFSR, and FFSI.

3) Stores the final values of the functions and their derivatives at the completion of each integration.

RKINT (Runge-Kutta integration)

This is the most crucial subroutine in the program as most of the time is spent in numerical integration. Special efforts have therefore been made to produce a rapid program.

The subroutine integrates numerically as per equations (102) to (112) the differential equations (96) operating simultaneously on the two sets corresponding to $\vec{\sigma} \cdot \vec{\ell} = \ell$ and $-\ell - 1$.

Special provisions have been made to avoid overflow; this is accomplished by dividing all the functions and their derivatives by the largest of these at every step (RENORM); whenever such renormalization is carried out it is accompanied by the following printout:
RENORMALIZATION FACTOR = (value of RENORM) IN RKINT FOR CODED
L =(value of $\ell + 1$) and RHO =(value of ρ at which renormalization took place).

CSUBL

This subroutine computes $C_\ell^\pm$ as per equation (125) for $\ell = 0$ to $\ell_{\max}$.

AB

This subroutine computes A(J) and B(J) for J = 1 **to** JMAX i.e., for the various angles θ required, as per equation (60).

SQSGCP (Sigma, sigma-coulomb, polarization)

This subroutine computes $\sigma(\theta)$, $P(\theta)$, $\sigma_c(\theta)$, as per equations (34), (35); (136) and finally $\sigma(\theta)/\sigma_c(\theta)$ for the various angles required.

SIGMAR

This subroutine computes σ_R as per equation (132).

***CHISQ* (Chi Square)**

This subroutine computes $\chi^2_\sigma(\theta)$, χ^2_σ, $\chi^2_P(\theta)$, χ^2_P, χ^2_T as per equations (137), (138) and (139).

Note: The quantities $\Delta\sigma^{\text{ex}}(\theta)$ and $\Delta P^{\text{ex}}(\theta)$ are always assumed to be non-zero. Thus to avoid including an unknown experimental quantity, the corresponding standard deviation must be taken as very large.

OUTPT4 (Output 4)

Several output formats are available:

(1) **Minimum output** (KTRL(6) = 1).

(a) Basic quantities
NUMPRG
KTRL(I) for I = 1 **to** 13
FMI, FMB, ELAB, ZZ, V, W, A, RO, VS, WS, RC, BG, RG RHOBN, RHOBC, RHOBNG, ECM,ETA, FKAY, FKAYA, FKAYB
and, if either KTRL(7), (8), (9), or (10) is not zero,
HA, RMA, FN1A, FN2A, PNA, HB, RMB, FN1B, FN2B, PMB,
then RHOMAX, LMAXM, NMAX, RHOIN(I) for I = 1 **to** NMAX,

DRHOIN(I) for I = 1 **to** NMAX−1, SGMRTH
and, if KTRL(2) = 1, *CHI2ST*, *CHI2PT*, *CHI2T*.

(b) Basic Table
THETAD(I), SGMATH(I), SRATIO(I), POLTH(I),
and, if KTRL(2) = 1, SGMAEX(I), POLEX(I), for I = 1 **to** JMAX.

(2) **Normal output** (KTRL(6) = 0)

(a) Basic quantities
(See above)

(b) Basic Table
(See above)

(c) Form factor table (output only if KTRL(12) = 1)
RHO(I), FFCR(I), FFCI(I), FFSR(I), FFSI(I),
for I = 1 **to** ILAST.

(d) Fitting table (output only if KTRL(2)=1)
THETAD(I), DSGMEX(I), DPOLEX(I), *CHI2S(I)*,
CHI2P(I), *CHI2(I)* for I = 1 **to** JMAX.

(e) L table
L, *CR1(L)*, *CI1(L)*, *CR2(L)*, *CI2(L)* for L = 1 **to** LMAXM (corresponding to $\ell = 0$ to $\ell_{\max}$).

This output is made for *every* run, and maybe preceded by underflow descriptions which may be ignored, and by other comments referring to an increase in $\rho_{\max}$, $\ell_{\max}$, renormalization, etc.

Every page of output is headed by the run number on the left and the page number on the right. The number of lines per page is held to be less than 50, otherwise the subroutine calls subroutine SKIP which starts a new page.

SKIP

This subroutine increases the page number, resets K, the line counter, and outputs the run and page number. Note that arguments giving the number of lines, page and run numbers are required.

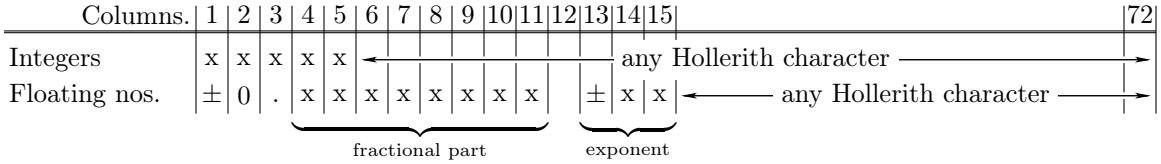
LEAVE

This subroutine is called whenever a run gets into difficulty because overflow, or division by zero occur. The subroutine calls PDUMP to give a partial core dump.

This subroutine was included so as to allow for various possible requirements upon overflow and division by zero without having to change every command where the difficulty might occur.

IV. Description of Input Data

All data is input from tape 7. The input data tape is prepared from IBM cards which contain one piece of input data per card in either of the two following formats:



Note: Any floating point format which uses 15 columns or less and is acceptable to FORTRAN may be used in place of the above.

(1) The following identification data is input first:

NUMRUN(1)	:	month
NUMRUN(2)	:	day
NUMRUN(3)	:	year
NUMRUN(4)	:	set number (put in 0 to start with 1)
NUMRUN(5)	:	run number (put in 0 to start with 1)
NUMPRG	:	program number (we use 4).

Note: The identification which consists of the five quantities NUMRUN(I), I = 1 **to** 5, is printed at the top left of every output sheet. NUMRUN(4) is advanced every time a new set of data is input, NUMRUN(5) is advanced every time a run is made with a new set of parameters.

(2) Then, for every set of run, i.e., for every set of input data:

(a) **Controls**

KTRL(1) = 0	:	Standard potential (possibly with generalized form factors)
= 1	:	Gaussian absorption
= 2	:	Square well[9]
KTRL(2) = 0	:	no χ^2 required
= 1	:	χ^2 required

[9]The quantity A is eventually discarded but it must still be input as 1/2 to avoid overflow in the early part of the program.

KTRL(3) = 0 : same experimental values as in last set
= 1 : new experimental values coming[10]
KTRL(4) : not used in present program
KTRL(5) = 0 : same angles as in last set
= 1 : new angles coming
KTRL(6) = 0 : normal output
= 1 : minimum output
KTRL(7) = 0 : UCR – Standard form
= 1 : UCR – form A
= 2 : UCR – form B
KTRL(8) = 0 : UCI – Standard form
= 1 : UCI – form A
= 2 : UCI – form B
KTRL(9) = 0 : USR – derivative standard form
= 1 : USR – derivative form A
= 2 : USR – form B
KTRL(10)= 0 : USI – derivative standard form
= 1 : USI – derivative form A
= 2 : USI – form B
KTRL(11)= 0 : do not include coulomb spin-orbit
= 1 : do include coulomb spin-orbit
KTRL(12)= 0 : do not print out form factors
= 1 : do print out form factors
KTRL(13)= 1 : check $\rho_{\max}$ and $\ell_{\max}$
= 2 : check $\rho_{\max}$ only
= 3 : check $\ell_{\max}$ only
= 4 : do not check $\rho_{\max}$ nor $\ell_{\max}$.

(b) **Basic data**
FMI, FMB, ELAB, ZZ, RC, V, W, RO, A, VS, WS, RG, BG, DV, DW, DA, DVS, DWS, DBG, HA, PMA, FN1A, FN2A, HB, PMB, FN1B, FN2B, NVMAX, NWMAX, NAMAX, NVSMAX, NWSMAX, NBGMAX.

(c) **Integration data**
NMAX, RHOIN(I) for I = 1 **to** NMAX, DRHOIN(I) for I = 1 **to** NMAX – 1,

(d) LMAXM

(e) **Angles**:
if KTRL(5) = 1 input: JMAX, THETAD(I) for I = 1 **to** JMAX

(f) **Experimental data**:

[10] KTRL(3) = 1 also requires KTRL(2) = 1 for proper operation.

if KTRL(2) = 1 and KTRL(3) = 1 input:

SGMAEX(I) for I = 1 **to** JMAX

DSGMEX(I) for I = 1 **to** JMAX

POLEX(I) for I = 1 **to** JMAX

DPOLEX(I) for I = 1 **to** JMAX

(3) Final card:
KTRL(l) = 100.

V. Glossary and Description of Symbolic Variables Appearing in Common and Dimension Statements

FORTRAN Symbol	Math. Symbol	Description
A	a	Rounding parameter appearing in standard potential, see eq. (62)
AR(I), AI(I) I = 1 **to** 75	$\mathrm{Re}\{a_i\}$, $\mathrm{Im}\{a_i\}$	1) Real and imaginary parts of the terms of the auxiliary series used to calculate asymptotically the coulomb functions, see eq. (116)
	$\mathrm{Re}\{A(\theta_i)\}$, $\mathrm{Im}\{A(\theta_i)\}$	2) See eq. (60) for definition
BR(I), BI(I) I = 1 **to** 75	$\mathrm{Re}\{b_i\}$, $\mathrm{Im}\{b_i\}$ $\mathrm{Re}\{B(\theta_i)\}$, $\mathrm{Im}\{B(\theta_i)\}$	1) Ibid, see eq. (116) 2) See eq. (60) for definition
BG	b	Width parameter in Gaussian absorption see eq. (82)
CHI2(I) I = 1 **to** 75	$\chi^2(\theta_i)$	$= \chi^2_\sigma(\theta_i) + \chi^2_P(\theta_i)$
CHI2P(I) I = 1 **to** 75	$\chi^2_P(\theta_i)$	See eq. (139)
CHI2PT	χ^2_P	See eq. (139)
CHI2S(I) I = 1 **to** 75	$\chi^2_\sigma(\theta_i)$	See eq. (138)
CHI2ST	χ^2_σ	See eq. (138)
CHI2T	χ^2	$= \chi^2_\sigma + \chi^2_P$
CR1(L), *CI1(L)* for L = 1 **to** 51	$\mathrm{Re}(C^+_\ell)$, $\mathrm{Im}(C^+_\ell)$	See eqs. (57) and (125)
CR2(L), *CI2(L)*	$\mathrm{Re}(C^-_\ell)$, $\mathrm{Im}(C^-_\ell)$	See eqs. (57) and (125)
DA, DV, DW, DVS, DWS, DBG		Amount by which A, V, W, VS, WS, BG must be incremented for succeeding runs (these increments may be input as positive, zero or negative).
DPOLEX(I) for I = 1 **to** 75	$\Delta P^{\mathrm{ex}}(\theta_i)$	Standard deviation in the experimental polarization (must *never* be input as 0)
DRHO(I) for I = 1 **to** 250	$\Delta\rho_i$	Interval of numerical integration (see description of subroutine RHOTB)
DRHOL		Last interval to be used in the numerical integration

FORTRAN Symbol	Math. Symbol	Description
DRHOIN(I) I = 1 **to** 250		Interval of numerical integration specified by input for RHOIN(I) $< \rho \leq$ RHOIN(I+1) (See description of subroutine RHOTB)
DSGMEX(I) I = 1 **to** 75	$\Delta\sigma^{\rm ex}(\theta_i)$	Standard deviation in the experimental differential elastic scattering cross section in square fermis/sterad, (must *never* be input as 0)
ECM	E	Incident energy in center-of-mass system (MeV)
ELAB	$E_{\rm LAB}$	Incident energy in laboratory system (MeV)
EPS1, EPS2, EPS3	ϵ_1, ϵ_2, ϵ_3	Error thresholds appearing in various parts of the calculation of the coulomb functions. See eqs. (155) to (158)
EPS4	ϵ_4	Error threshold used in POT1CH subroutine, see eqs. (144) to (150)
ETA	η	See eq. (43)
ETA2	η^2	
EXSGMR(L), EXSGMI(L) L = 1 **to** 51	Re$\{e^{2i\sigma_\ell}\}$, Im$\{e^{2i\sigma_\ell}\}$	See eq. (133)
F(L), L = 1 **to** 52	F_ℓ	See eq. (114) and (122)
FBAR(L), L = 1 **to** 91	$F_\ell^{(n)}$	See eq. (120)
FCR(I), FCI(I) I = 1 **to** 75	Re$\{f_c(\theta_i)\}$, Im$\{f_c(\theta_i)\}$	See eq. (47)
FFCR(I), FFCRM(I) I = 1 **to** 250	$f_{\rm CR}(\rho_i)$ $f_{\rm CR}(\rho_i + \frac{\Delta\rho_i}{2})$	Form factors for the real central part of the potential at the beginning and middle of an integration interval (See eqs. (80), (84), (86), (87) and description of subroutine PGEN4)
FFCI(I), FFCIM(I) I = 1 **to** 250	$f_{\rm CI}(\rho_i)$ $f_{\rm CR}(\rho_i + \frac{\Delta\rho_i}{2})$	As above for the imaginary central part of the potential (See eqs. (80), (82), (84), (86), (87), and description of subroutine PGEN4)
FFSR(I), FFSRM(I) I = 1 **to** 250	$f_{\rm SR}(\rho_i)$ $f_{\rm SR}(\rho_i + \frac{\Delta\rho_i}{2})$	As above for the real spin-orbit part of the potential (See eqs. (81), (85), (91), (94) and description of subroutine PGEN4)

FORTRAN Symbol	Math. Symbol	Description
FFSI(I), FFSIM(I) I = 1 **to** 250	$f_{\rm SI}(\rho_i)$ $f_{\rm SI}(\rho_i + \frac{\Delta\rho_i}{2})$	As above for the imaginary spin-orbit part of the potential (See eqs. (81), (85), (91), (94), and description of subroutine PGEN4)
FKAY	k	See eq. (8) (inverse fermis)
FKAYA	ka	
FKAYB	kb	
FMB	m_b	Mass number of target nucleus (atomic units)
FMI	m_i	Mass number of incident particle (atomic units)
FMU	μ	Reduced mass of incident particle (atomic units (see eq. (5))
FN1A, FN2A	nA_1, nA_2	See eq. (86) and following description
FN1B, FN2B	nB_1, nB_2	See eq. (87) and following description
FF(L), L = 1 **to** 51	F'_ℓ	See eq. (123)
G(L), L = 1 **to** 52	G_ℓ	See eq. (114) and (119)
GP(L), L = 1 **to** 51	G'_ℓ	See eq. (123)
HA, HB	h_{0A}, h_{0B}	See eq. (88)
IDATA		Number of sets of data to be processed after making use of subroutine **SAVE**
IFIRST		Initial value of I, the subscript appearing in RHO(I)
ILAST		Final value of I, the subscript appearing in RHO(I)
IIN(L), L = 1 **to** 51		Originally designed to allow input of any desired value of IFIRST for various L's in order to speed up the numerical integration. In the present program the IIN(L) are all set equal to 1 by subroutine INPT4
ISPILL, JSPILL		Underflow and overflow indicators used in conjunction with subroutine SPILL
JMAX		Total number of angles input (JMAX$\leq$75)
JMAXT		Temporary storage for JMAX used after calling subroutine **SAVE**

FORTRAN Symbol	Math. Symbol	Description
KTRL(I) I = 1 **to** 13		Controls used throughout the program to specify the potential, input and output type (see description of input data)
KTRLT(I) I = 1 **to** 13		Temporary storage for KTRL(I) used after calling subroutine **SAVE**
L	$\ell + 1$	
LMAX	$\ell_{\max} + 1$	
LMAXM	$\ell_{\max}$	
NA, NV, NW, NVS, NWS, NBG		**DO** loop variables used in subroutine *CTRL4* to specify the number of times the parameters have been incremented
NAMAX, NVMAX, NWMAX, NVSMAX, NWSMAX, NBGMAX		Total number of incrementations of the parameters specified as input data ($\geq$ 1)
NINPUT		**DO** loop variable used after calling subroutine **SAVE** in order to count the number of sets of processed input data
NMAX		Total number of input values of RHOIN(I) specified in input
NMAXT		Temporary storage for NMAX used after calling subroutine **SAVE**
NMAXP		= NMAX − 1
NUMPRG		Program number (see description of input data)
NUMRUN(I) I = 1 **to** 5		Identification (see description of input data)
POLEX(I) I = 1 **to** 75	$P^{\mathrm{ex}}(\theta_i)$	Experimental value of the polarization
POLTH(I) I = 1 **to** 75	$P^{\mathrm{th}}(\theta_i)$	Calculated value of the polarization See eq. (35)
P(L,J) L = 1 **to** 51 J = 1 **to** 75	$P_\ell(\theta_j)$	Legendre polynomial, see eq. (134)
PP(L,J) L = 1 **to** 50 J = 1 **to** 75	$P_\ell^{(l)}(\theta_j)$	Associated Legendre polynomial, see eq. (135)
PMA, PMB	$\rho_{m_A}/\bar{\rho_N}$ and $\rho_{m_B}/\bar{\rho_N}$	These are the quantities specified by the input as they are more convenient than RMA and RMB.

FORTRAN Symbol	Math. Symbol	Description
RO	R_{ON}	Nuclear radius constant (fermis), see eq. (63)
RC	R_{OC}	Charge radius constant (fermis) see eq. (66)
RG	R_{OG}	Gaussian radius constant (fermis) see eq. (83)
RHOBC	$\bar{\rho}_C$	Value of ρ at which the uniform charge density ends, see eq. (74)
RHOBN	$\bar{\rho}_N$	Value of ρ at which the standard potential falls to half of its initial value, see eq. (73)
RHOBNG	$\bar{\rho}_G$	Value of ρ at which the Gaussian absorption is centered
RHOIN(I) I = 1 **to** 250		Input values of ρ for which the integration interval must change from DRHOIN(I–1) to DRHOIN(I). See description of subroutine RHOTB)
ROMAX		Final value of ρ in the numerical integration
RHO(I) I = 1 **to** 250	ρ_i	Value of ρ at the i-th interval of integration, see eq. (14)
RMA, RMB	ρ_{m_A}, ρ_{m_B}	Values of ρ at which special form factors are matched to standard form factors, see eqs. (86) and (87)
SGMAC(I) I = 1 **to** 75	$\sigma_c(\theta_i)$	See eq. (136) (square fermis/sterad)
SGMAEX(I) I = 1 **to** 75	$\sigma^{\mathrm{ex}}(\theta_i)$	Experimental values of the differential elastic scattering cross section (square fermis/sterad)
SGMATH(I) I = 1 **to** 75	$\sigma^{\mathrm{th}}(\theta_1)$	Calculated values of the differential elastic scattering cross section (square fermis/sterad), see eq. (34)
SGMRTH	σ_R	Calculated value of the reaction cross section (square fermis) see eq. (132)
SIGMA0	σ_0	See eqs. (117) and (118)
SIGMA1	σ_1	See eq. (117)
SRATIO(I) I = 1 **to** 75	$\sigma(\theta_i)/\sigma_c(\theta_i)$	Ratio of calculated to Rutherford cross section
TA, TV, TW, TVS, TWS, TBG,		Storage for initial values input for the parameters

FORTRAN Symbol	Math. Symbol	Description
THETAD(I) I = 1 **to** 75	θ_i	Scattering angle in center-of-mass system (degrees)
THETA(I) I = 1 **to** 75	θ_i	As above (radians)
UCRB(I), UCRM(I) I = 1 **to** 250	$U_{\rm CR}(\rho_i)$ $U_{\rm CR}(\rho_i + \frac{\Delta\rho_i}{2})$	L-independent part of the real central potential at the beginning and in the middle of the i-th interval of integration, see eq. (98)
UCIB(I), UCIM(I) I = 1 **to** 250	$U_{\rm CI}(\rho_i)$ $U_{\rm CI}(\rho_i + \frac{\Delta\rho_i}{2})$	As above for the imaginary central potential, see eq. (99)
USRB(I), USRM(I) I = 1 **to** 250	$U_{\rm SR}(\rho_i)$ $U_{\rm SR}(\rho_i + \frac{\Delta\rho_i}{2})$	As above for the real spin-orbit potential, see eq. (100)
USIB(I), USIM(I) I = 1 **to** 250	$U_{\rm SI}(\rho_i)$ $U_{\rm SI}(\rho_i + \frac{\Delta\rho_i}{2})$	As above for the imaginary spin-orbit potential, see eq. (101)
V	V	Depth of real central potential (MeV)
W	W	Depth of imaginary central potential (MeV)
VS	V_S	Real part of spin-orbit potential depth (MeV)
WS	W_S	Imaginary part of spin-orbit potential depth (MeV)
XC1, XCP1	$x_\ell^+(\rho)$, $\dot{x}_\ell^+(\rho)$	Real part of the radial (unnormalized) wave function and its first derivative for the case $L + 1/2$
YC1, YCP1	$y_\ell^+(\rho)$, $\dot{y}_\ell^+(\rho)$	As above for the imaginary part and the case $L + 1/2$
XD1, XDP1	$x_\ell^-(\rho)$, $\dot{x}_\ell^-(\rho)$	As above for the real part and the case $L - 1/2$
YD1, YDP1	$y_\ell^-(\rho)$, $\dot{y}_\ell^-(\rho)$	As above for the imaginary part and the case $L - 1/2$
X1(L), X1P(L) L = 1 **to** 51	$x_\ell^+(\rho_{\rm max})$, $\dot{x}_\ell^+(\rho_{\rm max})$	Real part of the radial (unnormalized) wave function and its first derivative for the case $L+1/2$ at the end of a numerical integration
Y1(L), Y1P(L) L = 1 **to** 51	$y_\ell^+(\rho_{\rm max})$, $\dot{y}_\ell^+(\rho_{\rm max})$	As above for the imaginary part and the case $L + 1/2$
X2(L), X2P(L) L = 1 **to** 51	$x_\ell^-(\rho_{\rm max})$, $\dot{x}_\ell^-(\rho_{\rm max})$	As above for the real part and the case $L - 1/2$
Y2(L), Y2P(L) L = 1 **to** 51	$y_\ell^-(\rho_{\rm max})$, $\dot{y}_\ell^-(\rho_{\rm max})$	As above for the imaginary part and the case $L - 1/2$

FORTRAN Symbol	Math. Symbol	Description
ZZ	ZZ'	Product of the atomic numbers of the target nucleus and the incident particle.

VI. Symbolic Listing of the Program

```
 MAIN ROUTINE - SCAT 4
 COMMON A,AR,AI,
1BR,BI,BG,
2CHI2,CHI2P,CHI2PT,CHI2S,CHI2ST,CHI2T,CR1,CI1,CR2,CI2,
3DPOLEX,DSGMEX,DRHO,DRHOIN,DRHOL,DV,DW,DA,DVS,DWS,DBG,
4ECM,ELAB,EPS1,EPS2,EPS3,EPS4,ETA,ETA2,EXSGMR,EXSGMI,
5F,FBAR,FCR,FCI,FFCR,FFCI,FFCRM,FFCIM,FFSR,FFSI,FFSRM,FFSIM,
6FKAY,FMB,FMI,FMU,FN1A,FN2A,FN1B,FN2B,FP,FKAYA,FKAYB,
7G,GP,
8HA,HB,
9IDATA,IFIRST,IIN,ILAST,ISPILL
 COMMON JMAX,JMAXT,JSPILL,
1KTRL,KTRLT,
2L,LMAX,LMAXM,
3NMAX,NMAXP,NMAXT,NINPUT,NUMRUN,NUMPRG,NVMAX,NWMAX,NAMAX,NVSMAX,
4NWSMAX,NV,NW,NA,NVS,NWS,NBGMAX,NBG,
5P,PP,POLEX,POLTH,PMA,PMB,
6RC,RO,RHO,RHOBC,RHOBN,RHOIN,RHOMAX,RMA,RMB,RG,RHOBNG,
7SGMAC,SGMAEX,SGMATH,SGMRTH,SIGMA0,SIGMA1,SRATIO,
8THETA,THETAD,TV,TW,TA,TVS,TWS,TBG,
9UCRB,UCIB,UCRM,UCIM,USRB,USIB,USRM,USIM
 COMMON V,VS,
1W,WS,
2X1,X2,X1P,X2P,XC1,XCP1,XD1,XDP1,
3Y1,Y2,Y1P,Y2P,YC1,YCP1,YD1,YDP1,
4ZZ
 DIMENSION AR(75),AI(75),
1BR(75),BI(75),
2CHI2(75),CHI2P(75),CHI2S(75),CR1(51),CI1(51),CR2(51),CI2(51),
3DPOLEX(75),DSGMEX(75),DRHO(250),DRHOIN(250),
4EXSGMR(51),EXSGMI(51),
5F(52),FBAR(91),FCR(75),FCI(75),FFCR(250),FFCI(250),FFCRM(250),
6FFCIM(250),FFSR(250),FFSI(250),FFSRM(250),FFSIM(250),FP(51),
7G(52),GP(51),
8IIN(51),
9KTRL(13),KTRLT(13)
 DIMENSION NUMRUN(5),
1P(51,75),PP(50,75),POLEX(75),POLTH(75),
2RHO(250),RHOIN(250),
3SGMAC(75),SGMAEX(75),SGMATH(75),SRATIO(75),
4THETA(75),THETAD(75),
5UCRB(250),UCIB(250),UCRM(250),UCIM(250),USRB(250),USIB(250),
6USRM(250),USIM(250),
7X1(51),X2(51),X1P(51),X2P(51),
8Y1(51),Y2(51),Y1P(51),Y2P(51)
 CALL SPILL(JSPILL,ISPILL,0.,0.)
 EPS1= 0.00001
 EPS2= 0.00001
 EPS3= 0.00001
 EPS4=0.001
```

```
      READ INPUT TAPE 7,10,(NUMRUN(I)),I=1,5)
      READ INPUT TAPE 7,10,NUMPRG
10    FORMAT(I5)
      CALL CTRL4
      GO
```

```
      SUBROUTINE CTRL4
  3   NUMRUN(4)=NUMRUN(4)+1
      NUMRUN(5)=0
      CALL INPT4
      CALL POT1CH
 35   IF(KTRL(5)) 80,81,80
 80   CALL POP1
 81   CALL SIGZRO
      CALL FSUBC
      CALL EXSGML
      DO 20 NV=1,NVMAX
      IF (NV-1) 102,101,102
 101  V=TV
      GO TO 103
 102  V=V+DV
 103  DO 20 NW=1,NWMAX
      IF (NW-1) 105,104,105
 104  W=TW
      GO TO 109
 105  W=W+DW
 109  DO 20 NA=1,NAMAX
      IF (NA-1) 111,110,111
 110  A=TA
      GO TO 112
 111  A=A+DA
 112  DO 20 NVS=1,NVSMAX
      IF (NVS-1) 114,113,114
 113  VS=TVS
      GO TO 115
 114  VS=VS+DVS
 115  DO 20 NWS=1,NWSMAX
      IF (NWS-1) 117,116,117
 116  WS=TWS
      GO TO 118
 117  WS=WS+DWS
 118  DO 20 NBG=1,NBGMAX
      IF(NBG-1) 120,119,120
 119  BG=TBG
      GO TO 121
 120  BG=BG+DBG
 121  IF (SENSE SWITCH 1) 26,27
  26  REWIND 7
      CALL SAVE(8)
      READ INPUT TAPE 7,50,(LGAR,I=1,6)
      IDATA= NUMRUN(4)
      DO 66 NINPUT=1, IDATA
      READ INPUT TAPE 7,50,(KTRLT(I),I=1,13)
 50   FORMAT (I5)
 51   FORMAT (E15.9)
      READ INPUT TAPE 7,51,(GAR,I=1,27)
      READ INPUT TAPE 7,50,(LGAR,I=1,6), NMAXT
      NT=2*NMAXT-1
      READ INPUT TAPE 7,51,(GAR,I=1,NT)
      READ INPUT TAPE 7,51,LGAR
```

```
      IF  (KTRLT(5))  71,70,71
71    READ INPUT TAPE 7,50, JMAXT
      READ INPUT TAPE 7,51,(GAR,I=1,JMAXT)
70    IF  (KTRLT(2))  61,66,61
61    IF(KTRLT(3))  63,66,63
 63   NT=4*JMAXT
      READ INPUT TAPE 7,51,(GAR,I=1,NT)
66    CONTINUE
27    NUMRUN(5)= NUMRUN(5)+1
      CALL RHOTB
      CALL COULFN
      CALL RMXINC
      CALL PGEN4
      CALL INTCTR
      CALL CSUBL
      CALL AB
      CALL SGSGCP
      CALL SIGMAR
      IF  (KTRL(2))  33,100,33
 33   CALL CHISQ
 100  CALL OUTPT4
 20   CONTINUE
      GO TO 3
```

```
      SUBROUTINE INPT4
      IF DIVIDE CHECK 100,110
100   WRITE OUTPUT TAPE 6,101
101   FORMAT(59H DIVIDE CHECK TRIGGER FOUND ON AT START OF INPT4 SUBROUT
     1INE)
      CALL LEAVE
      STOP
110   ISPILL=0
      JSPILL=0
      READ INPUT TAPE 7,10,KTRL(1)
      IF (KTRL(1)-100) 150,151,151
 151  CALL EXIT
      STOP
 150  READ INPUT TAPE 7,10,(KTRL(I),I=2,13)
10    FORMAT (I5)
      READ INPUT TAPE 7,12,FMI,FMB,ELAB,ZZ,RC,V,W,RO,A,VS,WS,RG,BG,
     1DV,DW,DA,DVS,DWS,DBG
      READ INPUT TAPE 7,12,HA,PMA,FN1A,FN2A,HB,PMB,FN1B,FN2B
      READ INPUT TAPE 7,10,NVMAX,NWMAX,NAMAX,NVSMAX,NWSMAX,NBGMAX
12    FORMAT (E15.9)
      TV= V
      TW=W
      TA=A
      TVS=VS
      TWS=WS
      TBG=BG
      READ INPUT TAPE 7,10,NMAX
      NMAXP=NMAX-1
      READ INPUT TAPE 7,12,(RHOIN(I),I=1,NMAX),(DRHOIN(I),I=1,NMAXP)
      CO2=FMI+FMB
      FMU=(FMI*FMB)/CO2
      ECM=ELAB*(FMB/CO2)
      FKAY= .2195376*SQRTF(FMU*ECM)
      T=FKAY*(FMB**.333333333)
      RHOBN= T*RO
      RHOBNG=T*RG
      RMA=PMA*RHOBN
      RMB=PMB*RHOBN
      RHOBC= T*RC
      ETA= .15805086*ZZ*SQRTF(FMI/ELAB)
      IF DIVIDE CHECK 200,47
200   WRITE OUTPUT TAPE 6,201
201   FORMAT(43H INPUT DIVISOR WAS ZERO IN INPT4 SUBROUTINE)
      CALL LEAVE
      STOP
47    READ INPUT TAPE 7,10,LMAXM
      LMAX=LMAXM+1
      DO 147 J=1,LMAX
147   IIN(J)=1
      IF (KTRL(5)) 48,50,48
48    READ INPUT TAPE 7,10,JMAX
      READ INPUT TAPE 7,12,(THETAD(I),I=1,JMAX)
      DO 49 I=1,JMAX
49    THETA(I)= 0.01745329252*THETAD(I)
```

```
50    IF(KTRL(2))  51,207,51
51    IF(KTRL(3))  53,207,53
53    READ INPUT TAPE 7,12,(SGMAEX(I),I=1,JMAX),(DSGMEX(I),I=1,JMAX),
     1(POLEX(I),I=1,JMAX),(DPOLEX(I),I=1,JMAX)
207   IF(ISPILL)202,204,202
202   WRITE OUTPUT TAPE 6,203,ISPILL
 203 FORMAT(23H UNDERFLOW OCCURRED AT I5,20H IN INPT4 SUBROUTINE)
204   IF(JSPILL)205,210,205
205   WRITE OUTPUT TAPE 6,206,JSPILL
 206 FORMAT(22H OVERFLOW OCCURRED AT I5,20H IN INPT4 SUBROUTINE)
      CALL LEAVE
      STOP
210   RETURN
```

```
      SUBROUTINE POT1CH
      IF DIVIDE CHECK 30,31
 30   WRITE OUTPUT TAPE 6,130
 130  FORMAT (60H DIVIDE CHECK TRIGGER FOUND ON AT START OF POT1CH SUBRO
     1UTINE)
      CALL LEAVE
      STOP
 31   ISPILL=0
      JSPILL=0
      IKTRL=KTRL(13)
      NMAX=NMAX
      NMAXP= NMAX-1
      AMAX=NAMAX-1
      TTA=MAX1F(A,((AMAX*DA)+A))
      VMAX=NVMAX-1
      TTV=MAX1F(V,((VMAX*DV)+V))
      WMAX=NWMAX-1
      TTW=MAX1F(W,((WMAX*DW)+W))
      VSWAX=NVSMAX-1
      TTVS=MAX1F(VS,((VSMAX*DVS)+VS))
      WSMAX=NWSMAX-1
      TTWS=MAX1F(WS,((WSMAX*DWS)+WS))
      BGMAX=NBGMAX-1
      TTBG=MAX1F(BG,((BGMAX*DBG)+BG))
      FKAYA=FKAY*TTA
      FKAYB=FKAY*TTBG
      T2=SQRTF(TTV**2+TTW**2)/ECM
      T7=TTV/ECM
      T8=TTW/ECM
      IF DIVIDE CHECK 60,61
 60   WRITE OUTPUT TAPE 6,160
 160  FORMAT(26H ECM IS ZERO IN POT1CH SUB)
      CALL LEAVE
      STOP
 61   GO TO (3,3,111,15),IKTRL
 3    IF(KTRL(1)-2) 24,25,24
 25   IF(RHOIN(NMAX)-RHOBN) 10,10,8
 24   T1=1./(1.+EXPF((RHOIN(NMAX)-RHOBN)/FKAYA))
      IF DIVIDE CHECK 50,28
 50   WRITE OUTPUT TAPE 6,150
 150  FORMAT(28H FKAYA IS ZERO IN POT1CH SUB)
      CALL LEAVE
      STOP
 28   IF(KTRL(1)-1) 40,41,40
 40   T3= T2*T1
      GO TO 43
 41   T3=T7*T1
 43   IF(T3-EPS4) 42,42,10
 10   WRITE OUTPUT TAPE 6,100, RHOIN(NMAX),DRHOIN(NMAXP)
 100  FORMAT(13H RHOIN(NMAX)=E16.9,2H+ E16.9,46H RHOIN(NMAX) IS TOO SMAL
     1L IN NUCLEAR POTENTIAL)
      RHOIN(NMAX)= RHOIN(NMAX)+DRHOIN(NMAXP)
      GO TO 3
 42   IF(KTRL(1)-1) 8,6,8
```

```
6       T11= EXPF(-((RHOIN(NMAX)-RHOBNG)/FKAYB)**2)
        IF((T8*T11)-EPS4) 8,8,7
7       WRITE OUTPUT TAPE 6,103,RHOIN(NMAX),DRHOIN(MMAXP)
103     FORMAT(13H RHOIN(NMAX)=E16.9,2H+ E16.9,46H RHOIN(NMAX) IS TOO SMAL
       1L IN NUCLEAR POTENTIAL)
        RHOIN(NMAX)= RHOIN(NMAX)+DRHOIN(NMAXP)
        GO TO 6
8       GO TO(111,15),IKTRL
111     FLMAX=LMAXM
        IF(KTRL(1)-2) 29,300,29
300     IF(FLMAX-(RHOBN+3.)) 12,12,15
29      T4=1./(1.+EXPF((FLMAX-RHOBN)/FKAYA))
        IF(KTRL(1)-1) 33,32,33
33      T5= T2*T4
        GO TO 310
32      T5=T7*T4
310     IF(T5-EPS4)13,13,12
  12    WRITE OUTPUT TAPE 6,101,LMAXM
 101    FORMAT (7H LMAXM=I5,3H +1,45H LMAXM TOO SMALL BECAUSE OF CENTRAL P
       1OTENTIAL)
        LMAX= LMAX+1
        LMAXM= LMAXM+1
        IIN(LMAX)=1
        GO TO 111
13      IF(KTRL(1)-1) 17,19,17
19      T4=EXPF(-((FLMAX-RHOBNG)/FKAYB)**2)
        IF((T8*T4)-EPS4) 17,17,20
20      WRITE OUTPUT TAPE 6,200,LMAXM
200     FORMAT (7H LMAXM=I5,3H +1,45H LMAXM TOO SMALL BECAUSE OF CENTRAL P
       1OTENTIAL)
        LMAX=LMAX+1
        LMAXM=LMAXM+1
        IIN(LMAX)=1
        GO TO 19
17      T2=SQRTF(TTVS**2+TTWS**2)/ECM
18      FLMAX=LMAXM
        T4=1./(1.+EXPF((FLMAX-RHOBN)/FKAYA))
38      T6=2.*T2*T4*(FKAYW**2)
        IF(T6-EPS4) 15,15,14
 14     WRITE OUTPUT TAPE 6,102, LMAXM
 102    FORMAT (7H LMAXM=I5,3H +1,48H LMAXM TOO SMALL BECAUSE OF SPIN ORB
       1IT POTENTIAL)
        LMAX= LMAX+1
        LMAXM= LMAXM+1
        IIN(LMAX)=1
        GO TO 18
15      IF(ISPILL)202,204,202
202     WRITE OUTPUT TAPE 6,203,ISPILL
 203    FORMAT(23H UNDERFLOW OCCURRED AT I5,14H IN POT1CH SUB)
204     IF(JSPILL)205,210,205
205     WRITE OUTPUT TAPE 6,206,JSPILL
 206    FORMAT(22H OVERFLOW OCCURRED AT I5,14H IN POT1CH SUB)
        CALL LEAVE
        STOP
```

```
210   RETURN
```

```
      SUBROUTINE POP1
      IF DIVIDE CHECK 1,2
 1    WRITE OUTPUT TAPE 6,101
101   FORMAT (58H DIVIDE CHECK TRIGGER FOUND ON AT START OF POP1 SUBROUT
     1INE)
      CALL LEAVE
      STOP
 2    ISPILL=0
      JSPILL=0
      LMAXP=LWAX+1
      DO 20 J=1,JMAX
      SI2=1./SINF(THETA(J))
      IF DIVIDE CHECK 3,4
 3    WRITE OUTPUT TAPE 6,103, J
103   FORMAT (71H DIVISOR SINF THETA IS ZERO IN FIRST DIVISION OF POP1 S
     1UBROUTINE FOR J=I3)
      CALL LEAVE
      STOP
 4    CO=COSF(THETA(J))
      P(1,J)=1.0
      P(2,J)=CO
      PP(1,J)=0.0
      TWOLP1=3.
      FL=1.
      DO 20 L=1,LMAXP
      TL=FL+1.
      P(L+2,J)=(TWOLP1*CO*P(L+1,J)-FL*P(L,J))/TL
      PP(L+1,J)=TL*SI2*(CO*P(L+1,J)-P(L+2,J))
      TWOLP1=TWOLP1+2.
 20   FL=TL
      IF (ISPILL) 30,31,30
 30   WRITE OUTPUT TAPE 6,130, ISPILL
 130  FORMAT(23H UNDERFLOW OCCURRED AT I6,19H IN POP1 SUBROUTINE)
 31   IF (JSPILL) 32,33,32
 32   WRITE OUTPUT TAPE 6,132, JSPILL
 132  FORMAT (22H OVERFLOW OCCURRED AT I6,19H IN POP1 SUBROUTINE)
      CALL LEAVE
      STOP
 33   RETURN
```

```
      SUBROUTINE SIGZRO
      IF DIVIDE CHECK 5,6
5     WRITE OUTPUT TAPE 6,105
105   FORMAT (60H DIVIDE CHECK TRIGGER FOUND ON AT START OF SIGZRO SUBRO
     1UTINE)
      CALL LEAVE
      STOP
6     ISPILL = 0
      JSPILL = 0
      SIGMA0=-(ETA/(12.*(ETA**2+16.)))*(1.+(ETA**2-48.)/(30.*((ETA**2+16
     1.)**2))+(ETA**4-160.*(ETA**2)+1280.)/(((16.+ETA**2)**4)*105.))
      SIGMA0=SIGMA0-ETA+(ETA/2.)*LOGF(ETA**2+16.)+((7./2.)*ATANF(ETA/4.)
     1)-(ATANF(ETA)+ATANF(ETA/2.)+ATANF(ETA/3.))
      SIGMA1=SIGMA0+ATANF(ETA)
15    IF (ISPILL) 30,31,30
 30   WRITE OUTPUT TAPE 6,130,ISPILL
130   FORMAT (23H UNDERFLOW OCCURRED AT I6,21H IN SIGZRO SUBROUTINE)
31    IF (JSPILL) 32,11,32
32    WRITE OUTPUT TAPE 6,132,JSPILL
132   FORMAT (22H OVERFLOW OCCURRED AT I6,21H IN SIGZRO SUBROUTINE)
      CALL LEAVE
      STOP
11    RETURN
```

```
      SUBROUTINE FSUBC
      IF DIVIDE CHECK 20,21
   20 WRITE OUTPUT TAPE 6,120
  120 FORMAT (53H DIVIDE TRIGGER FOUND ON AT START OF FSUBC SUBROUTINE)
      CALL LEAVE
      STOP
   21 ISPILL=0
      JSPILL=0
      DO 10 J=1,JMAX
      SN=(SINF(THETA(J)/2.0))**2
      FLN=ETA*(LOGF(SN))-2.0*SIGMA0
      FNO=ETA/(2.0*FKAY*(SN))
      IF DIVIDE CHECK 22,23
   22 WRITE OUTPUT TAPE 6,122,J
  122 FORMAT (23H DIVISOR IS ZERO FOR J=I3,20H IN FSUBC SUBROUTINE)
      CALL LEAVE
      STOP
   23 FCR(J)=(-FNO*COSF(FLN))
   10 FCI(J)=(FNO*SINF(FLN))
      IF (ISPILL) 24,25,24
   24 WRITE OUTPUT TAPE 6,124, ISPILL
  124 FORMAT (23H UNDERFLOW OCCURRED AT I6,20H IN FSUBC SUBROUTINE)
   25 IF (JSPILL) 26,27,26
   26 WRITE OUTPUT TAPE 6,126, JSPILL
  126 FORMAT (22H OVERFLOW OCCURRED AT I6,20H IN FSUBC SUBROUTINE)
      CALL LEAVE
      STOP
   27 RETURN
```

```
      SUBROUTINE EXSGML
      IF DIVIDE CHECK 10,11
 10   WRITE OUTPUT TAPE 6,110
 110  FORMAT (60H DIVIDE CHECK TRIGGER FOUND ON AT START OF EXSGML SUBRO
     1UTINE)
      CALL LEAVE
      STOP
 11   ISPILL=0
      JSPILL=0
 1    FL=O.
      EXSGMR(1)=COSF(2.0*SIGMA0)
      EXSGMI(1)=SINF(2.0*SIGMA0)
      ETA2=ETA**2
      ETA2A=2.0*ETA
      DO 20 L=2,LMAX
      FL=FL+1.0
      TER0=FL**2
      TER1=TER0+ETA2
      TER2=(TER0–ETA2)/TER1
      TER3=(ETA2A*FL)/TER1
      IF DIVIDE CHECK 12,13
 12   WRITE OUTPUT TAPE 6,112,L
 112  FORMAT (44H DIVISOR IS ZERO IN EXSGML SUBROUTINE FOR L=I3)
      CALL LEAVE
      STOP
 13   EXSGMR(L)=(TER2*EXSGMR(L-1))-(TER3*EXSGMI(L-1))
 20   EXSGMI(L)=(TER2*EXSGMI(L-1))+(TER3*EXSGMR(L-1))
      IF (ISPILL) 14,15,14
 14   WRITE OUTPUT TAPE 6,114, ISPILL
 114  FORMAT(23H UNDERFLOW OCCURRED AT I6,21H IN EXSGML SUBROUTINE)
 15   IF (JSPILL) 16,17,16
 16   WRITE OUTPUT TAPE 6,116,JSPILL
 116  FORMAT(22H OVERFLOW OCCURRED AT I6,21H IN EXSGML SUBROUTINE)
      CALL LEAVE
      STOP
 17   RETURN
```

```
      SUBROUTINE RHOTB
      DRHO(1)=DRHOIN(1)
      RHO(1)=RHOIN(1)
      N=1
      I=1
 20   RHO(I+1)=RHO(I)+DRHOIN(N)
      IF  (RHO(I+1)-RHOIN(NMAX))30,50,70
30    IF(ABSF(RHO(I+1)-RHOIN(N+1))-.5*DRHOIN(N))  35,35,40
 35       N=XMINOF(N+1,NMAX-1)
 40   DRHO(I+1)=DRHOIN(N)
      I=I+1
      GO TO 20
 50   ILAST=I+1
60    RHO(ILAST)=RHOIN(NMAX)
      DRHO(ILAST-1)=RHO(ILAST)-RHO(ILAST-1)
      RHOMAX=RHOIN(NMAX)
      DRHOL=DRHOIN(NMAX-1)
      IF(ISPILL)  80,81,80
 80   WRITE OUTPUT TAPE 6,180,ISPILL
180   FORMAT(23H UNDERFLOW OCCURRED AT I6,21H IN RHOTB SUBROUTINE)
81    IF(JSPILL)82,83,82
 82   WRITE OUTPUT TAPE 6,182,JSPILL
182   FORMAT(22H OVERFLOW OCCURRED AT I6,21H IN RHOTB SUBROUTINE)
      CALL LEAVE
      STOP
 83   RETURN
 70   IF((RHO(I+1)-RHOIN(NMAX))-.5*DRHOIN(N))50,50,75
 75   ILAST=I
      GO TO 60
```

```
      SUBROUTINE COULFN
      IF DIVIDE CHECK 50,51
 50   WRITE OUTPUT TAPE 6,150
150   FORMAT (60H DIVIDE CHECK TRIGGER FOUND ON AT START OF COULFN SUBRO
     1UTINE)
      CALL LEAVE
      STOP
 51   ISPILL=0
      JSPILL=0
      IKTRL=KTRL(13)
      LMAX=LMAXM+1
      ETA2=ETA**2
      SQ=SQRTF(1.+ETA2)
 1    IJ = 1
      AR(1)=-ETA
      AI(1)=0.
      AR(2)=-.5*ETA2
      AI(2)=.5*ETA
 2    SI=0.
      SR=0.
      PR= RHOMAX
      DO 10  K=2,49
      T= PR*FLOATF(1-K)
      TR=AR(K)/T
      TI=AI(K)/T
      IF DIVIDE CHECK 52,53
 52   WRITE OUTPUT TAPE 6,152
 152  FORMAT(57H DIVISOR T IS ZERO IN FIRST DIVISION OF COULFN SUBROUTIN
     1E)
      CALL LEAVE
      STOP
 53   SQN=TR**2+TI**2
      IF (K-2) 4,4,3
 3    IF (SQN-SQO) 4,4,11
 4    TR=SR+TR
      TI=SI+TI
      IF (TR-SR) 6,5,6
 5    IF (TI-SI) 6,13,6
 6    SR=TR
      SI=TI
      AR(K+1)=0.
      AI(K+1)=0.
      KP=K/2
      DO 7 M=1,KP
      KM=K+1-M
      AR(K+1)=AR(K+1)-AR(M)*AR(KM)+AI(W)*AI(KM)
      AI(K+1)=AI(K+1)-AI(KM)*AR(M)-AI(M)*AR(KM)
      IF (K-2*KP) 8,9,8
      AR(K+1)=AR(K+1)-.5*(AR(KP+1)**2-AI(KP+1)**2)
      AI(K+1)=AI(K+1)-AR(KP+1)*AI(KP+1)
 9    FK=.5*FLOATF(K)
      AI(K+1)=AI(K+1)-FK*AR(K)
      AR(K+1)=AR(K+1)+FK*AI(K)
      PR= PR*RHOMAX
```

```
10    SQO=SQN
      GO TO 101
 11   T=SR**2+SI**2
      IF(T) 105,105,12
 12   IF(ABSF(SQO/T)-EPS3) 13,13,106
 13   GO TO (14,15),IJ
 14   PAR=RHOMAX-ETA*LOGF(2.*RHOMAX)
      PHI0R=PAR+SIGMA0+SR
      PHI0I=SI
      AR(2)=-1.+AR(2)
      IJ=2
      GO TO 2
 15   PHI1R=PAR+SIGMA1-1.570796325+SR
      PHI1I=SI
 25   T1=EXPF(-PHI0I)
      T2=EXPF(-PHI1I)
      G(1)=T1*COSF(PHI0R)
      G(2)=T2*COSF(PHI1R)
      F1=T1*SINF(PHI0R)
      F2=T2*SINF(PHI1R)
      IF(ABSF(F1*G(2)-F2*G(1)-1./SQ)-EPS1)  31,31,102
 31   IDEC=11
 32   I=LMAX+IDEC
      FBAR(I)=.1
      FBAR(I+1)=0.
      LIMIT=LMAXM+IDEC
      FL=LMAX+11
      T1=SQRTF((FL+1.)**2+ETA2)
      IF(JSPILL) 139,133,139
139   WRITE OUTPUT TAPE 6,1390,JSPILL
1390  FORMAT(23H OVERFLOW2 OCCURRED AT I6,21H IN COULFN SUBROUTINE)
      CALL LEAVE
      STOP
133   DO 33 I=1,LIMIT
      L=LMAX+IDEC-I
      FL=L
      T2=SQRTF(FL**2+ETA2)
      FBAR(L)=((2.*FL+1.)*(ETA+FL*(FL+1.)/RHOMAX)*FBAR(L+1)-FL*T1*FBAR(L
     1+2))/((FL+1.)*T2)
      IF DIVIDE CHECK 54,600
 54   WRITE OUTPUT TAPE 6,154
 154  FORMAT(56H DIVISOR IS ZERO IN SECOND DIVISION OF COULFN SUBROUTINE
     1)
      CALL LEAVE
      STOP
600   IF(JSPILL) 601,33,601
601   WRITE OUTPUT TAPE 6,1601,JSPILL
1601  FORMAT(22H OVERFLOW OCCURRED AT I6,21H IN COULFN SUBROUTINE,24H MU
     1LTIPLY FBAR(I) BY 0.1)
      K=LMAX+IDEC
      FBAR(K)=FBAR(K)*0.1
      JSPILL=0
      GO TO 133
 33   T1=T2
```

```
      ALPHA=1./((FBAR(1)*G(2)-FBAR(2)*G(1))*SQ)
      IF DIVIDE CHECK 55,43
 55   WRITE OUTPUT TAPE 6,155
 155  FORMAT (55H DIVISOR IS ZERO IN THIRD DIVISION OF COULFN SUBROUTINE
     1)
      CALL LEAVE
      STOP
 43   LMAXP=LMAX+1
      DO 34 I=1,LMAXP
 34   FBAR(I)=ALPHA*FBAR(I)
      IF(IDEC-11) 371,35,371
 371  IF(ABSF(F1/FBAR(1)-1.)-EPS2) 37,37,35
 35   DO 36 I=1,LMAXP
 36   F(I)=FBAR(I)
      IDEC=IDEC+5
      IF (IDEC-40) 32,32,103
37    DO 38 I=1,LMAXP
      IF(ABSF(F(I)/FBAR(I)-1.)-EPS2) 44,44,35
 44   IF DIVIDE CHECK 56,38
 56   WRITE OUTPUT TAPE 6,156,L,I
156   FORMAT(74H DIVISOR FBAR(I)-1. IS ZERO IN FOURTH DIVISION OF COULFN
     1 SUBROUTINE FOR  L=I3,7H AND I=I3)
      CALL LEAVE
      STOP
38    CONTINUE
      DO 381 I=1,MAXP
381   F(I)=FBAR(I)
382   T1=SQ
      DO 40 L=1,LMAX
      FL=L
      T2=SQRTF((FL+1.)**2+ETA2)
      G(L+2)=((2.*FL+1.)*(ETA+FL*(FL+1.)/RHOWAX)*G(L+1)-(FL+1.)*T1*G(L))
     1/(FL*T2)
      TS=FL/T1
      IF DIVIDE CHECK 57,45
57    WRITE OUTPUT TAPE 6,157
157   FORMAT(58H DIVISOR T1 IS ZERO IN FIFTH DIVISION OF COULFN SUBROUTI
     1NE)
      CALL LEAVE
      STOP
 45   IF(ABSF(F(L)*G(L+1)-F(L+1)*G(L)-TS)-EPS1) 40,40,104
 40   T1=T2
 41   DO 42 L=1,LMAX
      FL=L
      T=FL**2
      T1=T/RHOMAX+ETA
      IF DIVIDE CHECK 58,46
 58   WRITE OUTPUT TAPE 6,158
 158  FORMAT (62H DIVISOR RHOMAX IS ZERO IN SIXTH DIVISION OF COULFN SUB
     1ROUTINE)
      CALL LEAVE
      STOP
 46   T2=SQRTF(T+ETA2)
      FP(L)=(T1*F(L)-T2*F(L+1))/FL
```

```
 42   GP(L)=(T1*G(L)-T2*G(L+1))/FL
      IF DIVIDE CHECK 59,47
 59   WRITE OUTPUT TAPE 6,159
 159  FORMAT(60H DIVISOR FL IS ZERO IN SEVENTH DIVISION OF COULFN SUBROU
     1TINE)
      CALL LEAVE
      STOP
 47   IF(ISPILL) 60,61,60
 60   WRITE OUTPUT TAPE 6,160,ISPILL
 160  FORMAT(23H UNDERFLOW OCCURRED AT I6,21H IN COULFN SUBROUTINE)
 61   IF(JSPILL) 62,63,62
 62   WRITE OUTPUT TAPE 6,162,JSPILL
 162  FORMAT(22H OVERFLOW OCCURRED AT I6,21H IN COULFN SUBROUTINE)
      CALL LEAVE
      STOP
 63   RETURN
 101  WRITE OUTPUT TAPE 6,121,RHOMAX,DRHOL
      GO TO (110,110,109,109),IKTRL
 109  WRITE OUTPUT TAPE 6,114
      GO TO 13
 102  WRITE OUTPUT TAPE 6,122,RHOMAX,DRHOL
      GO TO(110,110,111,111),IKTRL
 111  WRITE OUTPUT TAPE 6,114
      GO TO 31
 103  WRITE OUTPUT TAPE 6,123,RHOMAX,DRHOL
      GO TO (110,110,112,112),IKTRL
 112  WRITE OUTPUT TAPE 6,114
      GO TO 382
 104  WRITE OUTPUT TAPE 6,124,RHOMAX,DRHOL ,L
      GO TO (110,110,113,113),IKTRL
 113  WRITE OUTPUT TAPE 6,114
      GO TO 40
 105  WRITE OUTPUT TAPE 6,125,RHOMAX,DRHOL
      GO TO (110,110,115,115),IKTRL
 115  WRITE OUTPUT TAPE 6,114
      GO TO 12
 106  WRITE OUTPUT TAPE 6,126,RHOMAX,DRHOL
      GO TO (110,110,116,116),IKTRL
 116  WRITE OUTPUT TAPE 6,114
      GO TO 13
 110  RHOMAX=RHOMAX+DRHOL
      GO TO 1
 121  FORMAT(18H INCREASE RHO MAX=E11.4,2H+ E11.4,35H A OR B SERIES CONV
     1ERGES TOO SLOWLY)
 122  FORMAT(18H INCREASE RHO MAX=E11.4,2H+ E11.4,22H BAD INITIAL WRONSK
     1IAN)
 123  FORMAT(18H INCREASE RHO MAX=E11.4,2H+ E11.4,24H L TOO LARGE IN FBA
     1R (L))
 124  FORMAT(18H INCREASE RHO MAX=E11.4,2H+ E11.4,21H BAD WRONSKIAN FOR
     1L=I3)
 125  FORMAT(67H SERIES IN PHI0 OR PHI1 IS ZERO, CHECK DATA, IF OK INCRE
     1ASE RHOMAX=E11.4,2H+ E11.4)
 126  FORMAT(52H A OR B SERIES DIVERGES TOO QUICKLY INCREASE RHOMAX=E11.
     14,2H+ E11.4)
```

```
114   FORMAT(42H RHOMAX INCREASE NOT PERMITTED BY KTRL(13))



      SUBROUTINE RMXINC
3     IF (RHOMAX-RHO(ILAST)) 1,2,1
1     ILAST=ILAST+1
      RHO(ILAST)=RHO(ILAST-1)+DRHOL
      DRHO(ILAST-1)=DRHOL
      GO TO 3
2     RETURN
```

```
      SUBROUTINE PGEN4
      IF DIVIDE CHECK 60,61
 60   WRITE OUTPUT TAPE 6,160
 160  FORMAT (59H DIVIDE CHECK TRIGGER FOUND ON AT START OF PGEN4 SUBROU
     1TINE)
      CALL LEAVE
      STOP
 61   ISPILL=0
      JSPILL=0
      IF(KTRL(1)) 3,4,3
3     KTRL(7)=0
      KTRL(8)=0
      KTRL(9)=0
      KTRL(10)=0
4     T1=V/ECM
      T2=W/ECM
      T10=VS/ECM
      T11=WS/ECM
      T12=FKAY*BG
      T3=2.*FKAY/A
      IF DIVIDE CHECK 62,65
 62   WRITE OUTPUT TAPE 6,162
 162  FORMAT (65H DIVISORS ECM OR A WERE WRONGLY INPUT AS ZERO IN PGEN4
     1SUBROUTINE)
      CALL LEAVE
      STOP
 65   T4=T10*T3
      T5=T11*T3
      T6=FKAY*A
      T7=ETA/RHOBC
      IF DIVIDE CHECK 63,64
 63   WRITE OUTPUT TAPE 6,163
 163  FORMAT(61H DIVISOR RHOBC IS ZERO IN SECOND DIVISION OF PGEN4 SUBRO
     1UTINE)
      CALL LEAVE
      STOP
 64   T8=RHOBC**2
      T9=ETA*2.
      I=1
 40   EX=EXPF((RHO(I)-RHOBN)/T6)
      IF DIVIDE CHECK 80,66
 80   WRITE OUTPUT TAPE 6,165
 165  FORMAT (58H QUANTITY T6 IS ZERO IN THIRD DIVISION OF PGEN4 SUBROUT
     1INE)
      CALL LEAVE
      STOP
 66   K=1
 41   IF(I-1) 42,43,42
 42   IF(DRHO(I)-DRHO(I-1)) 43,44,43
 43   HDRHO=DRHO(I)*.5
      DEX=EXPF(HDRHO/T6)
44    IF(KTRL(1)-2)53,52,53
52    IF(RHO(I)-RHOBN) 54,55,55
54    S1=1.0
```

```
      GO TO 68
55    S1=0.0
      GO TO 68
53    S1=1./(1.+EX)
      IF DIVIDE CHECK 67,68
 67   WRITE OUTPUT TAPE 6,167
167   FORMAT(60H DIVISOR 1.+EX IS ZERO IN FOURTH DIVISION OF PGEN4 SUBRO
     1UTINE)
      CALL LEAVE
      STOP
 68   S2=EX*(S1**2)
      S4=S2/RHO(I)
      IF DIVIDE CHECK 69,70
 69   WRITE OUTPUT TAPE 6,169,I
169   FORMAT(58H DIVISOR RHO IS ZERO IN FIFTH DIVISION OF PGEN4 SUBROUTI
     1NE)
      CALL LEAVE
      STOP
 70   IF (RHO(I)-RHOBC) 9,9,10
 9    S3=T7*(3.-(RHO(I)**2)/T8)
      GO TO 11
 10   S3=T9/RHO(I)
11    IF (KTRL(7)) 350,300,350
 300  UCRB(I)=-1.-T1*S1+S3
      FFCR(I)=S1
 301  IF (KTRL(8)) 355,302,355
302   IF(KTRL(1)-1) 309,308,309
308   S1=EXPF(-((RHO(I)-RHOBNG)/T12)**2)
      IF DIVIDE CHECK 82,309
82    WRITE OUTPUT TAPE 6,182
182   FORMAT(22H BG IS ZERO IN PGEN SR)
      CALL LEAVE
      STOP
309   UCIB(I)=-T2*S1
      FFCI(I)=S1
 303  IF (KTRL(9)) 360,304,360
 304  USRB(I)=T4*S4
      FFSR(I)=S4
 305  IF (KTRL(11)) 501,500,501
 500  IF (KTRL(10))365,306,365
 306  USIB(I)=T5*S4
      FFSI(I)=S4
 307  IF (I-ILAST) 50,200,200
 350  ITT=1
      GO TO 340
 355  ITT=2
      GO TO 340
 340  ITQ=1
      IF(ITT-1) 380,380,381
 380  IF (KTRL(7)-1) 352,351,352
 351  TW=HA
      TRM=RMA
      TN1=FN1A
      TN2=FN2A
```

```
      GO TO 400
352   TH=HB
      TRM=RMB
      TN1=FN1B
      TN2=FN2B
      GO TO 400
 381  IF (KTRL(8)-1) 352,351,352
 400  IF (RHO(I)-RHOBN) 410,410,411
 410  TTN=TN1
      GO TO 412
 411  TTN=TN2
412   T20=RHO(I)/RHOBN
      IF (TTN*LOGF(T20)-80.) 403,403,409
 403  TQ=(T20**TTN-1.)*RHOBN/(TTN*FKAY*A)
      IF DIVIDE CHECK 405,406
405   TG=T20**(RHOBN/(FKAY*A))
      GO TO 407
 406  IF (TQ-80.) 408,408,409
 408  TG=EXPF(TQ)
      GO TO 407
409   TF=0.
      GO TO 422
 407  TFN=1./(1.+TG)
      IF (RHO(I)-TRM) 420,420,419
 419  TF=TFN
      GO TO 418
 420  T21=RHO(I)/TRM
      THH=TH*(1.+(2.*T21))*((1.-T21)**2)
      TF=TFN*(1.+THH)
 418  TFF=TF
 421  GO TO (422,423),ITQ
 422  GO TO (425,426,427,428),ITT
 425  FFCR(I)=TF
      UCRB(I)=-1.-T1*FFCR(I)+S3
      GO TO 301
 426  FFCI(I)=TF
      UCIB(I)=-T2*FFCI(I)
      GO TO 303
427   FFSR(I)=TF
      IF (ITQ-1) 470,470,471
 471  USRB(I)=FKAY*A*T4*FFSR(I)
      GO TO 305
470   USRB(I)=(T4/2.)*FFSR(I)
      GO TO 305
 428  FFSI(I)=TF
      IF(ITQ-1) 472,472,473
 473  USIB(I)=FKAY*A*T5*FFSI(I)
      GO TO 307
 360  ITT=3
      IF (KTRL(9)-1) 431,431,430
 430  ITQ=1
      GO TO 352
 365  ITT=4
      IF (KTRL(10)-1)  431,431,430
```

```
472   USIB(I)=(T5/2.)*FFSI(I)
      GO TO 307
431   ITQ=2
      GO TO 351
 423  T23=(RHOBN/(FKAY*A))*(T20**TTN)*TG*((TFN/RHO(I))**2)
      T25=T23
      IF(RHO(I)-TRM) 460,460,461
460   T24=6.*TH*(1.-T21)/(TRM**2)
      T25=(T24*TFN)+((1.+THH)*T23)
461   TF=T25
      IF(ITT-3) 427,427,428
501   T30=0.004927*ETA*ECM
      IF(RHO(I)-RHOBC) 502,502,503
 502  SOCOUL=T30/(RHOBC**3)
      GO TO 504
 503  SOCOUL=T30/(RHO(I)**3)
 504  USRB(I)=USRB(I)+SOCOUL
      GO TO 500
 50   I=I+1
      EX=EX*DEX
      RHOM=RHO(I-1)+HDRHO
      IF(KTRL(1)-2) 153,152,153
152   IF(RHOM-RHOBN)34,35,35
34    S1=1.0
      GO TO 72
35    S1=0.0
      GO TO 72
153   S1=1./(1.+EX)
      IF DIVIDE CHECK 71,72
 71   WRITE OUTPUT TAPE 6,171
171   FORMAT(54H DIVISOR 15 ZERO IN SIXTH DIVISION OF PGEN4 SUBROUTINE)
      CALL LEAVE
      STOP
 72   S2=EX*(S1**2)
      S4=S2/RHOM
      IF DIVIDE CHECK 73,74
 73   WRITE OUTPUT TAPE 6,173
 173  FORMAT (62H QUANTITY RHOM IS ZERO IN SEVENTH DIVISION OF PGEN4 SUB
     1ROUTINE)
      CALL LEAVE
      STOP
 74   IF(RHOM-RHOBC) 21,21,22
 21   S3=T7*(3.-(RHOM**2)/T8)
      GO TO 23
 22   S3=T9/RHOM
 23   IF (KTRL(7))1350,1300,1350
1300  UCRM(I-1)=-1.-T1*S1+S3
      FFCRM(I-1)=51
1301  IF (KTRL(8)) 1355,1302,1355
1302  IF(KTRL(1)-1) 1309,1308,1309
1308  S1=EXPF(-((RHOM-RHOBNG)/T12)**2)
1309  UCIM(I-1)=-T2*S1
      FFCIM(I-1)=S1
1303  IF (KTRL(9)) 1360,1304,1360
```

```
1304 USRM(I-1)=T4*S4
     FFSRM(I-1)=S4
1305 IF (KTRL(I1)) 1501,1500,1501
1500 IF (KTRL(I0))1365,1306,1365
1306 USIM(I-1)=T5*S4
     FFSIM(I-1)=S4
1307 IF (K-10) 24,40,40
1350 ITT=1
     GO TO 1340
1355 ITT=2
     GO TO 1340
1340 ITQ=1
     IF (ITT-1)1380,1380,1381
1380 IF (KTRL(7)-1) 1352,1351,1352
1351 TH=HA
     TRM=RMA
     TN1=FN1A
     TN2=FN2A
     GO TO 1400
1352 TH=HB
     TRM=RMB
     TN1=FN1B
     TN2=FN2B
     GO TO 1400
1381 IF (KTRL(8)-1) 1352,1351,1352
1400 IF (RHOM-RHOBN) 1410,1410,1411
1410 TTN=TN1
     GO TO 1412
1411 TTN=TN2
1412 T20=RHOM/RHOBN
     IF (TTN*LOGF(T20)-80.) 1403,1403,1409
1403 TQ=(T20**TTN-1.)*RHOBN/(TTN*FKAY*A)
     IF DIVIDE CHECK 1405,1406
1405 TG=T20**(RHOBN/(FKAY*A))
     GO TO 1407
1406 IF (TQ-80.) 1408,1408,1409
1408 TG=EXPF(TQ)
     GO TO 1407
1409 TF=0.
     GO TO 1422
1407 TFN=1./(1.+TG)
     IF (RHOM-TRM) 1420,1420,1419
1419 TF=TFN
     GO TO 1418
1420 T21=RHOM/TRM
     TRH=TH*(1.+(2.*T21))*((1.-T21)**2)
     TF=TFN*(1.+THH)
1418 TFF=TF
1421 GO TO (1422,1423),ITQ
1422 GO TO (1425,1426,1427,1428),ITT
1425 FFCRM(I-1)=TF
     UCRM(I-1)=-1.-T1*FFCRM(I-1)+S3
     GO TO 1301
1426 FFCIM(I-1)=TF
```

```
      UCIM(I-1)=-T2*FFCIM(I-1)
      GO TO 1303
 1427 FFSRM(I-1)=TF
      IF (ITQ-1) 1470,1470,1471
 1471 USRM(I-1)=FKAY*A*T4*FFSRM(I-1)
      GO TO 1305
 1470 USRM(I-1)=(T4/2.)*FFSRM(I-1)
      GO TO 1305
 1428 FFSIM(I-1)=TF
      IF (ITQ-1) 1472,1472,1473
 1473 USIM(I-1)=FKAY*A*T5*FFSIM(I-1)
      GO TO 1307
 1360 ITT=3
      IF (KTRL(9)-1) 1431,1431,1430
 1430 ITQ=1
      GO TO 1352
 1365 IIT=4
      IF (KTRL(10)-1) 1431,1431,1430
 1472 USIM(I-1)=(T5/2.)*FFSIM(I-1)
      GO TO 1307
 1431 ITQ=2
      GO TO 1351
 1423 T23=(RHOBN/(FKAY*A))*(T20**TTN)*TG*((TFN/RHOM)**2)
      T25=T23
      IF(RHOM-TRM) 1460,1460,1461
 1460 T24=6.*TH*(1.-T21)/(TRM**2)
      T25=(T24*TFN)+((1.+THH)*T23)
 1461 TF=T25
      IF (ITT-3) 1427,1427,1428
 1501 T30=0.004927*ETA*ECM
      IF (RHOM-RHOBC) 1502,1502,1503
 1502 SOCOUL=T30/(RHOBC**3)
      GO TO 1504
 1503 SOCOUL=T30/(RHOM**3)
 1504 USRM(I-1)=USRM(I-1)+SOCOUL
      GO TO 1500
   24 K=K+1
      EX=EX*DEX
      GO TO 42
  200 IF(ISPILL) 75,76,75
   75 WRITE OUTPUT TAPE 6,175,ISPILL
  175 FORMAT(23H UNDERFLOW OCCURRED AT I6,20H IN PGEN4 SUBROUTINE)
   76 IF (JSPILL) 77,51,77
   77 WRITE OUTPUT TAPE 6,177, JSPILL
  177 FORMAT(22H OVERFLOW OCCURRED AT I6,20H IN PGEN4 SUBROUTINE)
      CALL LEAVE
      STOP
   51 RETURN
```

```
      SUBROUTINE INTCTR
      DO1 L=1,LMAX
      IFIRST=IIN(L)
      T=RHO(IFIRST)**(L-1)
      XC1=T*RHO(IFIRST)
      XD1=XC1
      FL=L
      XCP1=FL*T
      XDP1=XCP1
      YC1=0.
      YD1=0.
      YCP1=0.
      YDP1=0.
      CALL RKINT
      X1(L)=XC1
      X2(L)=XD1
      Y1(L)=YC1
      Y2(L)=YD1
      X1P(L)=XCP1
      X2P(L)=XDP1
      Y1P(L)=YCP1
    1 Y2P(L)=YDP1
      RETURN
```

```
      SUBROUTINE RKINT
      IF DIVIDE CHECK 10,11
10    WRITE OUTPUT TAPE 6,110,L,I
110   FORMAT(66H DIVIDE CHECK TRIGGER FOUND ON AT START OF RKINT SUBROUT
     1INE FOR L=I3,7H AND I=I3)
      CALL LEAVE
      STOP
11    ISPILL=0
      JSPILL=0
1     FL=L-1
      F2L=-1.-FL
      F3L=FL*(FL+1.)
      TB=UCRB(IFIRST)+F3L/(RHO(IFIRST)**2)
      IF DIVIDE CHECK 12,13
12    WRITE OUTPUT TAPE 6,112,L,I
112   FORMAT(76H DIVISOR RHO(IFIRST)**2 IS ZERO IN FIRST DIVISION OF RKI
     1NT SUBROUTINE FOR L=I3,7H AND I=I3)
      CALL LEAVE
      STOP
13    PCB=TB+USRB(IFIRST)*FL
      PDB=TB+USRB(IFIRST)*F2L
      QCB=UCIB(IFIRST)+USIB(IFIRST)*FL
      QDB=UCIB(IFIRST)+USIB(IFIRST)*F2L
      IK=ILAST-1
      DO 6 I=IFIRST,IK
2     HDRHO=.5*DRHO(I)
      DRHO2=(DRHO(I)**2)*.5
      RHOM=RHO(I)+HDRHO
      TM=UCRM(I)+F3L/(RHOM**2)
      IF DIVIDE CHECK 14,15
14    WRITE OUTPUT TAPE 6,114,L,I
114   FORMAT(70H DIVISOR RHOM**2 IS ZERO IN SECOND DIVISION OF RKINT SUB
     1ROUTINE FOR L=I3,7H AND I=I3)
      CALL LEAVE
      STOP
15    PCM=TM+USRM(I)*FL
      PDM=TM+USRM(I)*F2L
      QCM=UCIM(I)+USIM(I)*FL
      QDM=UCIM(I)+USIM(I)*F2L
      XCPP1=PCB*XC1-QCB*YC1
      YCPP1=QCB*XC1+PCB*YC1
      XDPP1=PDB*XD1-QDB*YD1
      YDPP1=QDB*XD1+PDB*YD1
      XC2=XC1+XCP1*HDRHO
      YC2=YC1+YCP1*HDRHO
      XD2=XD1+XDP1*HDRHO
      YD2=YD1+YDP1*HDRHO
      XCPP2=PCM*XC2-QCM*YC2
      YCPP2=QCM*XC2+PCM*YC2
      XDPP2=PDM*XD2-QDM*YD2
      YDPP2=QDM*XD2+PDM*YD2
      DRHO4=.5*DRHO2
      SDRHO=.33333333*HDRHO
      XC3=XC2+XCPP1*DRHO4
```

```
      YC3=YC2+YCPP1*DRHO4
      XD3=XD2+XDPP1*DRHO4
      YD3=YD2+YDPP1*DRHO4
      XCPP3=PCM*XC3-QCW*YC3
      YCPP3=QCM*XC3+PCM*YC3
      XDPP3=PDM*XD3-QDM*YD3
      YDPP3=QDM*XD3+PDW*YD3
      XC4=XC2+XCPP2*DRHO2+XCP1*HDRHO
      YC4=YC2+YCPP2*DRHO2+YCP1*HDRHO
      XD4=XD2+XDPP2*DRHO2+XDP1*HDRHO
      YD4=YD2+YDPP2*DRHO2+YDP1*HDRHO
      TB=UCRB(I+1)+F3L/(RHO(I+1)**2)
      IF DIVIDE CHECK 16,17
   16 WRITE OUTPUT TAPE 6,116,L,I
  116 FORMAT(74H DIVISOR RHO(I+1)**2 IS ZERO IN THIRD DIVISION FOR RKINT
     1 SUBROUTINE FOR L=I3,7H AND I=I3)
      CALL LEAVE
      STOP
   17 PCB=TB+USRB(I+1)*FL
      PDB=TB+USRB(I+1)*F2L
      QCB=UCIB(I+1)+USIB(I+1)*FL
      QDB=UCI8(I+1)+USIB(I+1)*F2L
      XCPP4=PCB*XC4-QCB*YC4
      YCPP4=QCB*XC4+PCB*YC4
      XDPP4=PDB*XD4-QDB*YD4
      YDPP4=QDB*XD4+PDB*YD4
      SXC=XCPP2+XCPP3
      SYC=YCPP2+YCPP3
      SXD=XDPP2+XDPP3
      SYD=YDPP2+YDPP3
      TXC=SXC+XCPP1
      TYC=SYC+YCPP1
      TXD=SXD+XDPP1
      TYD=SYD+YDPP1
      TXC1=XC1+DRHO(I)*(XCP1+SDRHO*TXC)
      TYC1=YC1+DRHO(I)*(YCP1+SDRHO*TYC)
      TXD1=XD1+DRHO(I)*(XDP1+SDRHO*TXD)
      TYD1=YD1+DRWO(I)*(YDP1+SDRHO*TYD)
      TXCP1=XCP1+SDRHO*(TXC+SXC+XCPP4)
      TYCP1=YCP1+SDRHO*(TYC+SYC+YCPP4)
      TXDP1=XDP1+SDRHO*(TXD+SXD+XDPP4)
      TYDP1=YDP1+SDRHO*(TYD+SYD+YDPP4)
      IF (JSPILL) 20,21,20
   20 RENORM=MAX1F(ABSF(XC1),ABSF(YC1),ABSF(XCP1),ABSF(YCP1),ABSF(XD1),
     1ABSF(YD1),ABSF(XDP1),ABSF(YDP1))
      XC1=XC1/RENORM
      YC1=YC1/RENORM
      XCP1=XCP1/RENORM
      YCP1=YCP1/RENORM
      XD1=XD1/RENORM
      YD1=YD1/RENORM
      XDP1=XDP1/RENORM
      YDP1=YDP1/RENORM
      WRITE OUTPUT TAPE 6,200,RENORM,L,RHO(I)
```

```
200 FORMAT(24H RENORMALIZATION FACTOR=E16.9,22H IN RKINT FOR CODED L=I
    13,9H AND RHO=E16.9)
    JSPILL=0
    GO TO2
21  XC1=TXC1
    YC1=TYC1
    XD1=TXD1
    YD1=TYD1
    XCP1=TXCP1
    YCP1=TYCP1
    XDP1=TXDP1
    YDP1=TYDP1
6   CONTINUE
    IF (ISPILL) 30,31,30
30  WRITE OUTPUT TAPE 6,130, ISPILL,L,I
130 FORMAT(23H UNDERFLOW OCCURRED AT I6,27H IN RKINT SUBROUTINE FOR L=
    1I3.7H AND I=I3)
31  IF (JSPILL) 32,4,32
32  WRITE OUTPUT TAPE 6,132, JSPILL,L,I
132 FORMAT(22H OVERFLOW OCCURRED AT I6,27H IN RKINT SUBROUTINE FOR L=I
    13,7H AND I=I3)
    CALL LEAVE
    STOP
4   RETURN
```

```
      SUBROUTINE CSUBL
      IF DIVIDE CHECK 50,51
   50 WRITE OUTPUT TAPE 6,150
  150 FORMAT (59H DIVIDE CHECK TRIGGER FOUND ON AT START OF CSUBL SUBROU
     1TINE)
      CALL LEAVE
      STOP
   51 ISPILL=0
      JSPILL=0
      DO 40 L=1,LMAX
      XNORM1=MAX1F(ABSF(X1(L))*ABSF(Y1(L)),ABSF(X1P(L)),ABSF(Y1P(L)))
      TX1L=N1(L)/XNORM1
      TY1L=Y1(L)/XNORM1
      TX1PL=N1P(L)/XNORM1
      TY1PL=Y1P(L)/XNORM1
      FNORM=MAX1F(F(L),G(L),FP(L),GP(L))
      TFL=F(L)/FNORM
      TGL=G(L)/FNORM
      TFPL=FP(L)/FNORM
      TGPL=GP(L)/FNORM
      CO1=TFL*TY1PL-TFPL*TY1L
      CO2=TFPL*TX1L-TFL*TX1PL
      CO3=TY1L*TGPL-TY1PL*TGL+TX1L*TFPL-TX1PL*TFL
      CO4=TX1PL*TGL-TX1L*TGPL+TY1L*TFPL-TY1PL*TFL
      CO7=1.0/(CO3**2+CO4**2)
      IF DIVIDE CHECK 52,53
   52 WRITE OUTPUT TAPE 6,152
  152 FORMAT(54H DIVISOR IS ZERO IN FIRST DIVISION OF CSUBL SUBROUTINE)
      CALL LEAVE
      STOP
   53 CR1(L)=(CO1*CO3+CO2*CO4)*CO7
      CI1(L)=(CO2*CO3-CO1*CO4)*CO7
      XNORM2=MAX1F(ABSF(X2(L)),ABSF(Y2(L)),ABSF(X2P(L)),ABSF(Y2P(L)))
      TX2L=N2(L)/XNORM2
      TY2L=Y2(L)/XNORM2
      TX2PL=N2P(L)/XNORM2
      TY2PL=Y2P(L)/XNORM2
      CO1=TFL*TY2PL-TFPL*TY2L
      CO2=TFPL*TX2L-TFL*TX2PL
      CO3=TY2L*TGPL-TY2PL*TGL+TX2L*TFPL-TX2PL*TFL
      CO4=TX2PL*TGL-TX2L*TGPL+TY2L*TFPL-TY2PL*TFL
      CO7=1.0/(CO3**2+CO4**2)
      IF DIVIDE CHECK 54,55
   54 WRITE OUTPUT TAPE 6,154
  154 FORMAT (55H DIVISOR IS ZERO IN SECOND DIVISION OF CSUBL SUBROUTINE
     1)
      CALL LEAVE
      STOP
   55 CR2(L)=(CO1*CO3+CO2*CO4)*CO7
   40 CI2(L)=(CO2*CO3-CO1*CO4)*CO7
      IF (ISPILL) 56,57,56
   56 WRITE OUTPUT TAPE 6,156,ISPILL,L
  156 FORMAT (23H UNDERFLOW OCCURRED AT I6,27H IN CSUBL SUBROUTINE FOR L
     1=I3)
```

```
 57   IF (JSPILL) 58,59,58
 58   WRITE OUTPUT TAPE 6,158, JSPILL, L
158   FORMAT (22H OVERFLOW OCCURRED AT I6,27H IN CSUBL SUBROUTINE FOR L=
     1I3)
      CALL LEAVE
      STOP
 59   RETURN
```

```
      SUBROUTINE AB
      IF DIVIDE CHECK 1,2
 1    WRITE OUTPUT TAPE 6,101
101   FORMAT (56H DIVIDE CHECK TRIGGER FOUND ON AT START OF AB SUBROUTIN
     1E)
      CALL LEAVE
      STOP
 2    ISPILL=0
      JSPILL=0
      FKAYD=1./FKAY
      IF DIVIDE CHECK 3,4
 3    WRITE OUTPUT TAPE 6,103
 103  FORMAT(38H DIVISOR FKAY IS ZERO IN AB SUBROUTINE)
      CALL LEAVE
      STOP
 4    DO 20 J=1,JMAX
      ASUMR=0.
      ASUMI=0.
      BSUMR=0.
      BSUMI=0.
      DO 10 L=1,LMAX
      FL=L
      ATR1=FL*CR1(L)+(FL-1.)*CR2(L)
      ATI1=FL*CI1(L)+(FL-1.)*CI2(L)
      BTR1=CR1(L)-CR2(L)
      BTI1=CI1(L)-CI2(L)
      ATR2=ATR1*EXSGMR(L)-(ATI1*EXSGMI(L))
      ATI2=ATR1*EXSGMI(L)+(ATI1*EXSGMR(L))
      BTR2=BTR1*EXSGMR(L)-(BTI1*EXSGMI(L))
      BTI2=BTR1*EXSGMI(L)+(BTI1*EXSGMR(L))
      ASUMR=ASUMR+(ATR2*P(L,J))
      ASUMI=ASUMI+(ATI2*P(L,J))
      BSUMR=BSUMR+(BTR2*PP(L,J))
 10   BSUMI=BSUMI+(BTI2*PP(L,J))
      AR(J)= FCR(J)+(FKAYD*ASUMR)
      AI(J)=FCI(J)+(FKAYD*ASUMI)
      BR(J)= FKAYD*BSUMI
 20   BI(J)= -FKAYD*BSUMR
      IF (ISPILL) 30,31,30
 30   WRITE OUTPUT TAPE 6,130, ISPILL
130   FORMAT(23H UNDERFLOW OCCURRED AT I6,17H IN AB SUBROUTINE)
31    IF (JSPILL) 32,33,32
 32   WRITE OUTPUT TAPE 6,132,JSPILL
132   FORMAT (22H OVERFLOW OCCURRED AT I6,17H IN AB SUBROUTINE)
      CALL LEAVE
      STOP
 33   RETURN
```

```
      SUBROUTINE SGSGCP
      IF DIVIDE CHECK 10,11
   10 WRITE OUTPUT TAPE 6,110
  110 FORMAT (60H DIVIDE CHECK TRIGGER FOUND ON AT START OF SGSGCP SUBRO
     1UTINE)
      CALL LEAVE
      STOP
   11  ISPILL=0
       JSPILL=0
      DO 5 J=1,JMAX
      SGMATH(J)=AR(J)**2.+AI(J)**2.+BR(J)**2.+BI(J)**2.
      POLTH(J)= (2.*(AR(J)*BR(J)+AI(J)*BI(J)))/SGMATH(J)
      IF DIVIDE CHECK 12,13
   12 WRITE OUTPUT TAPE 6,112,J
  112 FORMAT(30H DIVISOR SGMATH IS ZERO FOR J=13,21H IN SGSGCP SUBROUTIN
     1E)
      CALL LEAVE
      STOP
   13 SGMAC(J)=FCR(J)**2.+FCI(J)**2.
      IF(ETA) 7,7,8
    8 SRATIO(J)=SGMATH(J)/SGMAC(J)
      IF DIVIDE CHECK 14,15
   14 WRITE OUTPUT TAPE 6,114,J
  114 FORMAT(29H DIVISOR SGMAC IS ZERO FOR J=13,21H IN SGSGCP SUBROUTINE
     1)
      CALL LEAVE
      STOP
   15 GO TO 5
    7  SRATIO(J)=0.
    5  CONTINUE
      IF (ISPILL) 16,17,16
   16 WRITE OUTPUT TAPE 6,116,ISPILL
  116 FORMAT (23H UNDERFLOW OCCURRED AT 16,21H IN SGSGCP SUBROUTINE)
   17 IF (JSPILL) 18,19,18
   18 WRITE OUTPUT TAPE 6,118,JSPILL
  118 FORMAT(22H OVERFLOW OCCURRED AT 16,21H IN SGSGCP SUBROUTINE)
      CALL LEAVE
      STOP
   19 RETURN
```

```
      SUBROUTINE SIGMAR
      ISPILL=0
      JSPILL=0
      FL=0.
      SGMRTH=0.
      CPI=(12.56637060)/(FKAY**2)
      DO 20 L=I,LMAX
      SGMRTH=SGMRTH+FL*(C12(L)-(C12(L))**2-(CR2(L))**2)
      FL=FL+1.0
20    SGMRTH=SGMRTH+FL*(CI1(L)-(CI1(L))**2-(CR1(L))**2)
      SGMRTH=CPI*SGMRTH
      IF(ISPILL) 10,11,10
10    WRITE OUTPUT TAPE 6,110,ISPILL
110   FORMAT(23H UNDERFLOW OCCURRED AT 16,21H IN SIGMAR SUBROUTINE)
11    IF(JSPILL) 12,13,12
12    WRITE OUTPUT TAPE 6,112,JSPILL
112   FORMAT(22H OVERFLOW OCCURRED AT 16,21H IN SIGMAR SUBROUTINE)
      CALL LEAVE
      STOP
13    RETURN
```

```
      SUBROUTINE CHISQ
      IF DIVIDE CHECK 10,11
   10 WRITE OUTPUT TAPE 6,110
  110 FORMAT(59H DIVIDE CHECK TRIGGER FOUND ON AT START OF CHISQ SUBROUT
     1INE)
      CALL LEAVE
      STOP
   11 ISPILL=0
      JSPILL=0
      CHI2ST=0
      CHI2PT=0
      DO 20 J=1,JMAX
      CHI2S(J)= ((SGMATH(J)-SGMAEX(J))/DSGMEX(J))**2.
      CHI2P(J)= ((POLTH(J)-POLEX(J))/DPOLEX(J))**2.
      IF DIVIDE CHECK 14,15
   14 WRITE OUTPUT TAPE 6,114,J
  114 FORMAT(40H DIVISOR DSGMEX OR DPOLEX IS ZERO FOR J=13,20H IN CHISQ
     1SUBROUTINE)
      CALL LEAVE
      STOP
   15 CHI2ST=CHI2ST + CHI2S(J)
      CHI2(J)=CHI25(J)+CHI2P(J)
   20 CHI2PT=CHI2PT+CHI2P(J)
      CHI2T=CHI2ST+CHI2PT
      IF (ISPILL) 16,17,16
   16 WRITE OUTPUT TAPE 6,116, ISPILL
  116 FORMAT(23H UNDERFLOW OCCURRED AT 16,20H IN CHISQ SUBROUTINE)
   17 IF(ISPILL) 18,19,18
   18 WRITE OUTPUT TAPE 6,118,JSPILL
  118 FORMAT(22H OVERFLOW OCCURRED AT 16,20H IN CHISQ SUBROUTINE)
      CALL LEAVE
      STOP
   19 RETURN
```

```
      SUBROUTINE OUTPT4
      NPGS=0
      CALL SKIP(K,NPGS,NUMRUN)
      WRITE OUTPUT TAPE 6,245,NUMPRG
  245 FORMAT (16H0PROGRAM NUMBER, I5)
      DO 8 I=1,13
      WRITE OUTPUT TAPE 6,250,I,(KTRL(I))
  250 FORMAT (6H KTRL(I2,2H)=I2)
    8 CONTINUE
      WRITE OUTPUT TAPE 6,12
   12 FORMAT (11H0BASIC DATA)
      FKAYA=FKAY*A
      FKAYB=FKAY*BG
      WRITE OUTPUT TAPE 6,14,FMI,FMB,ELAB,ZZ,V,W,A,RO,VS,WS,RC,BG,RG,
   14 FORMAT(7H0MSUBI=E16.9,10H     MSUBB=E16.9,10H     ELAB=E16.9,10H
     1    ZZP=E16.9/7H0     V=E16.9,10H          W=E16.9,10H          A=E16.9,
     210H         RO=E16.9/7H0    VS=E16.9,10H         WS=E16.9,36H
     3                          RC=E16.9/59H0
     4                            BG=E16.9,10H         RG=E16.9)
      WRITE OUTPUT TAPE 6,16,RHOBN,RHOBC,RHOBNG,ECM,ETA,FKAY,FKAYA,FKAYB
   16 FORMAT(7H0RHOBN=E16.9,10H     RHOBC=E16.9,10H    RHOBNG=E16.9,10H
     1    ECM=E16.9/7H0  ETA=E16.9,10H          K=E16.9/10H          KA=E16.9,
     210H         KB=E16.9)
      KT=KTRL(7)+KTRL(8)+KTRL(9)+KTRL(10)
      IF (KT) 13,1818,13
   13 WRITE OUTPUT TAPE 6,150,HA,RMA,FN1A,FN2A,PMA,HB,RMB,FN1B,FN2B,PMB
  150 FORMAT(7H0    HA=E16.9,7H    RMA=E16.9,7H    N1A=E16.9,7H    N2A=E16.9
     1,7H    PMA=E16.9/7H    HB=E16.9,7H    RMB=E16.9,7H    N1B=E16.9,7H
     2N2B=E16.9,7H    PMB=E16.9)
 1818 WRITE OUTPUT TAPE 6,18,RHOMAX,LMAXM
   18 FORMAT (17H0INTEGRATION DATA/8H0RHOMAX=E16.9,10H     LMAXM=I5)
      WRITE OUTPUT TAPE 6,220,NMAX
  220 FORMAT (6H0NMAX=I5)
      WRITE OUTPUT TAPE 6,24
   24 FORMAT (6H0RHOIN)
      NOLINE=50
      K=20
      DO 40 I=1, NMAX,6
      IF(K-NOLINE) 30,29,29
   29 CALL SKIP(K,NPGS,NUMRUN)
   30 M=XMINOF(I+5,NMAX)
      K=K+1
      WRITE OUTPUT TAPE 6,32,(RHOIN(J),J=I,M)
   32 FORMAT(1H E19.9,5E20.9)
   40 CONTINUE
      WRITE OUTPUT TAPE 6,41
   41 FORMAT (7H0DRHOIN)
      DO 60 I=1,NMAX,6
      IF(K-NOLINE) 45,43,43
   43 CALL SKIP(K,NPGS,NUMRUN)
   45 M=XMINOF(I+5,NMAX-1)
      K=K+1
      WRITE OUTPUT TAPE 6,32,(DRHOIN(J),J=I,M)
   60 CONTINUE
```

```
      WRITE OUTPUT TAPE 6,118,SGMRTH
118   FORMAT(12H0SIGMAR(TH)=E16.9)
15    IF(KTRL(2)-1) 1900,20,1900
20    WRITE OUTPUT TAPE 6,119,CHI2ST,CHI2PT,CHI2T
 119  FORMAT (25H0SUM OF CHI SQUARE SIGMA=E16.9/23H0SUM OF CHI SQUARE PO
     1L=E16.9/25H0SUM OF CHI SQUARE TOTAL=E16.9)
21    CALL SKIP(K,NPGS,NUMRUN)
      WRITE OUTPUT TAPE 6,200
200   FORMAT (113H          THETA                SIGMATH                SIG-SIGC
     1               POL TH               SIGMA EX               POL EX)
      DO 90 I=1,JMAX
      IF(K-NOLINE)75,70,70
 70   CALL SKIP(K,NPGS,NUMRUN)
 75   K=K+1
      WRITE OUTPUT TAPE 6,32,THETAD(I),SGMATH(I),SRATIO(I),POLTH(I),
     1SGMAEX(I),POLEX(I)
 90   CONTINUE
      GO TO 299
1900  CALL SKIP (K,NPGS,NUMRUN)
      WRITE OUTPUT TAPE 6,1905
1905  FORMAT (120H                THETA                              SIGMATH
     1                    SIG-SIGC                              POL TH
     2)
      DO 1920 I=1,JMAX
      IF (K-NOLINE) 1910,1908,1908
1908  CALL SKIP (K,NPGS,NUMRUN)
1910  K=K+1
      WRITE OUTPUT TAPE 6,1919,THETAD(I),SGMATH(I),SRATIO(I),POLTH(I)
1919  FORMAT (1H E20.9,3E30.9)
1920  CONTINUE
299   IF(KTRL(6)-1) 300,121,300
300   IF(KTRL(12)-1) 25,1700,25
1700  CALL SKIP(K,NPGS,NUMRUN)
      WRITE OUTPUT TAPE 6,1701
1701  FORMAT (92H          RHO(I)                FFCR                FFCI
     1                FFSR                FFSI)
      DO 1709 I=1,ILAST
      IF (K-NOLINE) 1703,1702,1702
1702  CALL SKIP (K,NPGS,NUMRUN)
1703  WRITE OUTPUT TAPE 6,158,RHO(I),FFCR(I),FFCI(I),FFSR(I),FFSI(I)
 158  FORMAT(1H 5E20.9)
1709  CONTINUE
 25   IF(KTRL(2)-1) 23,22,23
 22   CALL SKIP (K,NPGS,NUMRUN)
      WRITE OUTPUT TAPE 6,95
 95   FORMAT(120H          THETA                DSIGMA EX                DPOL EX
     1         CHI SQUARE SIGMA         CHI SQUARE POL         CHI SQUARE TOTAL   )
      DO 120 J=1,JMAX
      IF(K-NOLINE) 97,96,96
 96   CALL SKIP(K,NPGS,NUMRUN)
 97   K=K+1
      WRITE OUTPUT TAPE 6,32,THETAD(J),DSGMEX(J),DPOLEX(J)*CHI2S(J),
     1CHI2P(J))CHI2(J)
120   CONTINUE
```

```
23     CALL SKIP(K,NPGS,NUMRUN)
1623   WRITE OUTPUT TAPE 6,1150
1150   FORMAT (120H                L                    REAL C(L+1/2)                IMA
      1G C(L+1/2)                  REAL C(L-1/2)              IMAG C(L-1/2)
      2)
       DO 160 L=1,LMAX
       IF (K-NOLINE) 155,153,153
153    CALL SKIP (K,NPGS,NUMRUN)
 155   K=K+1
       L1=L-1
       WRITE OUTPUT TAPE 6,1156,L1,CR1(L),CI1(L),CR2(L),CI2(L)
1156   FORMAT (1H I11,E30.9,3E25.9)
 160   CONTINUE
 121   RETURN
```

```
      SUBROUTINE SKIP(K,NPGS,NUMRUN)
      NPGS=NPGS+1
      WRITE OUTPUT TAPE 6,1510, (NUMRUN(I),I=1,5),NPGS
1510  FORMAT(12H1RUN NUMBER=I2,1H–I2,1H–I4,3H   –I3,3H   –I3,79H
     1                                                                PA
     2GE 15/)
      K=0
      RETURN




      SUBROUTINE LEAVE
      CALL PDUMP(A,ZZ)
      CALL CTRL4
      RETURN
```

```
*       CARDS   COLUMN
*       FAP
        COUNT   43
*SPILL SUBROUTINE
        ENTRY   SPILL
 SPILL  STZ*    1,4          STORE ZERO IN JSPILL
        STZ*    2,4            STORE ZERO IN ISPILL
        STZ     0             STORE ZERO IN LOCATION 00000
        CAL     1,4
        STA     AA41                SET ADDRESS AA41,
        STA     AA36                  AA36 TO JSPILL
        CAL     2,4                     SET ADDRESS AA31
        STA     AA31                    TO ISPILL
        CLA*    3,4                       SET COMMON STORAGE
        STO     AA45
        CLA*    4,4                       SET COMMON STORAGE
        STO     AA46
        CAL     AA47                  PLACE TRANSFER
        SLW     8                   INSTRUCTION IN LOCATION 8
        TRA     5,4                       EXIT TO MAIN PROGRAM
 AA16   LDI     0                   ENTRY IN CASE OF OVER-OR UNDERFLOW
        LFT     4                   TEST FOR OVERFLOW
        TRA     AA36              TRANSFER IN CASE OF OVERFLOW
        LFT     16
        TRA     AA24                    TRANSFER IN CASE OF UNDERFLOW
        TRA*    0                       TRANSFER TO MAIN PROGRAM, NO UFLOW
 AA24   LNT     1                       TEST FOR UNDERFLOW
        TRA*    0                   UNDERFLOW IN AC ONLY
        CAL     0                   PLACE LOCATION AT WHICH
        SUB     AA35                UNDERFLOW OCCURRED IN AC
        LLS     18                SHIFT LEFT 18
 AA31   STD     AA31           STORE IN ISPILL
        CLA     AA46           SET AC, MQ WITH
        LDQ     AA46                SPECIFIED CONSTANTS
        TRA*    0                 EXIT TO MAIN PROGRAM
 AA35   HTR     1                   CONSTANT
 AA36   CLA     AA36                    TEST IF JSPILL ZERO
        TNZ     AA42              TRANSFER IN CASE JSPILL NON-ZERO
        CAL     0               PLACE LOCATION AT WHICH OVERFLOW OCCURRED
        SUB     AA35                        IN AC
        LLS     18                      SHIFT LEFT 18
 AA41   STD     AA41                  STORE IN JSPILL
 AA42   CLA     AA45                  SET AC,MQ WITH SPECIFIED CONSTANTS
        LDQ     AA45
        TRA*    0                       EXIT TO MAIN PROGRAM
 AA45   HTR     0                       COMMON STORAGE
 AA46   HTR     0                       COMMON STORAGE
 AA47   TRA     AA16                INSTRUCTION TO BE INSERTED AT LOC.  8
        END
```

VII. Typical Input and Output

A. Input Data for Protons against Copper at 9.75 MeV

```
     3
    22
  1960
     0
     0
     4
     0
     1
     1
     0
     1
     0
     0
     0
     0
     0
     0
     1
+0.10000000 +01
+0.64000000 +02
+0.97500000 +01
+0.29000000 +02
+0.12000000 +01
+0.62000000 +02
+0.85000000 +01
+0.12000000 +01
+0.52000000 +00
-0.40000000 +01
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
     1
     1
     1
     1
     1
     1
     3
+0.62500000 -01
+0.50000000 +00
+0.10000000 +02
+0.62500000 -01
+0.25000000 +00
    10
    32
+0.15200000 +02
+0.20300000 +02
+0.25400000 +02
+0.28000000 +02
+0.30400000 +02
+0.33000000 +02
+0.35500000 +02
+0.39000000 +02
+0.40600000 +02
+0.43000000 +02
+0.45600000 +02
+0.47000000 +02
+0.507000000+02
+0.51500000 +02
+0.54000000 +02
+0.55700000 +02
+0.57000000 +02
+0.60000000 +02
+0.60800000 +02
+0.65500000 +02
+0.65800000 +02
+0.69000000 +02
+0.70800000 +02
+0.75500000 +02
+0.75900000 +02
+0.80900000 +02
+0.85900000 +02
+0.86000000 +02
+0.90900000 +02
+0.95500000 +02
+0.95900000 +02
+0.10000000 +03
+0.38650000 +04
+0.97340000 +03
+0.42470000 +03
+0.00000000 +00
+0.22690000 +03
+0.00000000 +00
+0.13460000 +03
+0.00000000 +00
+0.82920000 +02
+0.00000000 +00
+0.47660000 +02
+0.00000000 +00
+0.22870000 +02
+0.00000000 +00
+0.00000000 +00
+0.12410000 +02
+0.00000000 +00
+0.00000000 +00
+0.64560000 +01
+0.00000000 +00
+0.40750000 +01
+0.00000000 +00
+0.33390000 +01
+0.00000000 +00
+0.33560000 +01
+0.37570000 +01
+0.38570000 +01
+0.00000000 +00
+0.38460000 +01
+0.00000000 +00
+0.37570000 +01
+0.00000000 +00
+0.39800000 +03
+0.35500000 +02
+0.16700000 +02
+0.10000000 +30
+0.90800000 +01
+0.10000000 +30
+0.53800000 +01
+0.10000000 +30
+0.37300000 +01
+0.10000000 +30
+0.19100000 +01
+0.10000000 +30
+0.91500000 +00
+0.10000000 +30
+0.10000000 +30
+0.49600000 +00
+0.10000000 +30
+0.10000000 +30
+0.25800000 +00
+0.10000000 +30
+0.16300000 +00
+0.10000000 +30
+0.13400000 +00
+0.10000000 +30
+0.13400000 +00
+0.15000000 +00
+0.15400000 +00
+0.10000000 +30
+0.15400000 +00
+0.10000000 +30
+0.15000000 +00
+0.10000000 +30
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
-0.20000000 -01
+0.00000000 +00
+0.10000000 -01
+0.00000000 +00
-0.30000000 -01
+0.00000000 +00
-0.60000000 -01
+0.00000000 +00
-0.10000000 +00
+0.00000000 +00
-0.16000000 +00
-0.20000000 +00
+0.00000000 +00
-0.17000000 +00
-0.17000000 +00
+0.00000000 +00
-0.10000000 +00
+0.00000000 +00
+0.10000000 -01
+0.00000000 +00
+0.20000000 +00
+0.00000000 +00
+0.00000000 +00
+0.00000000 +00
+0.13000000 +00
+0.00000000 +00
+0.70000000 -01
+0.00000000 +00
-0.20000000 -01
+0.10000000 +30
+0.10000000 +30
+0.10000000 +30
+0.30000000 -01
+0.10000000 +30
+0.40000000 -01
+0.10000000 +30
+0.30000000 -01
+0.10000000 +30
+0.30000000 -01
+0.10000000 +30
+0.40000000 -01
+0.40000000 -01
+0.10000000 +30
+0.40000000 -01
+0.30000000 -01
+0.10000000 +30
+0.50000000 -01
+0.10000000 +30
+0.40000000 -01
+0.10000000 +30
+0.60000000 -01
+0.10000000 +30
+0.10000000 +30
+0.10000000 +30
+0.60000000 -01
+0.10000000 +30
+0.50000000 -01
+0.10000000 +30
+0.60000000 -01
   100
```

B. Output Listing

RUN NUMBER= 2−40−1961 − 1 − 1
PAGE 1

PROGRAM NUMBER 4

KTRL(1)=0
KTRL(2)=1
KTRL(3)=1
KTRL(4)=0
KTRL(5)=1
KTRL(6)=0
KTRL(7)=0
KTRL(8)=0
KTRL(9)=0
KTRL(10)=0
KTRL(11)=0
KTRL(12)=0
KTRL(13)=1

BASIC DATA

MSUB1= 0.099999994E 01 MSUBB= 0.639999993E 02 ELAB= 0.974999994E 01 ZZP= 0.289999999E 02

V= 0.619999997E 02 W= 0.849999994E 01 A= 0.519999996E 00 RO= 0.119999997E 01

VS=−0.399999999E 01 WS= 0. RC= 0.119999997E 01

9 BG= 0. RG= 0.

RHOBN= 0.393980615E 01 RHOBC= 0.323980615E 01 RHOBNG= 0. ECM= 0.959999986E 01

ETA= 0.146788672E 01 K= 0.674959674E 00 KA= 0.350979023E−00 KB= 0.

INTEGRATION DATA

RHOMAX= 0.099999994E 02 LMAXM= 10

NMAX= 3

RHOIN

0.625000000E−01 0.500000000E 00 0.099999994E 02

DRHOIN

0.625000000E−01 0.250000000E−00

SIGMAR(TH)= 0.668857820E 02

SUM OF CHI SQUARE SIGMA= 0.587550342E 02

SUM OF CHI SQUARE POL= 0.999665476E 02

SUM OF CHI SQUARE TOTAL= 0.158721581E 03

THETA	SIGMATH	SIG−SIGC	POL TH	SIGMA EX	POL EX
0.151999995E 02	0.366688885E 04	0.948844409E 00	0.624454483E−03	0.386499993E 04	0.
0.202999994E 02	0.107591100E 04	0.877576292E 00	−0.488765538E−02	0.973399989E 03	0.
0.253999993E 02	0.437771246E 03	0.864875652E 00	−0.126092605E−01	0.424699992E 03	0.
0.279999994E 02	0.302826010E 03	0.877252147E 00	−0.152099080E−01	0.	0.199999996E−01
0.303999998E 02	0.223364875E 03	0.892697871E 00	−0.161199562E−01	0.226899996E 03	0.
0.329999998E 02	0.164800696E 03	0.906894624E 00	−0.153716959E−01	0.	−0.999999985E−02
0.354999997E 02	0.124848992E 03	0.912098765E 00	−0.130572930E−01	0.134599991E 03	0.
0.389999993E 02	0.854579188E 02	0.897354133E 00	−0.757811405E−02	0.	0.299999997E−01
0.405999996E 02	0.718648233E 02	0.880526960E 00	−0.435241245E−02	0.829199985E 02	0.
0.430000006E 02	0.552558191E 02	0.843160637E 00	0.115248807E−02	0.	0.599999994E−01
0.455999993E 02	0.412997656E 02	0.787652783E 00	0.776244439E−02	0.476599991E 02	0.
0.469999999E 02	0.351879030E 02	0.752354726E 00	0.114611000E−01	0.	0.999999993E−01
0.506999992E 02	0.227685094E 02	0.647046342E 00	0.208112434E−01	0.228699997E 02	0.
0.514999993E 02	0.206800543E 02	0.623049393E 00	0.225530863E−01	0.	0.159999996E−00
0.539999999E 02	0.152632877E 02	0.548359923E 00	0.265358075E−01	0.	0.199999996E−00
0.556999996E 02	0.124075061E 02	0.499775782E−00	0.272366613E−01	0.124100000E 02	0.
0.569999993E 02	0.106054634E 02	0.464955918E−00	0.261134841E−01	0.	0.169999994E−00
0.599999994E 02	0.749187976E 01	0.396005623E−00	0.156014524E−01	0.	0.169999994E−00
0.607999995E 02	0.686953478E 01	0.380956881E−00	0.104624555E−01	0.645599991E 01	0.
0.654999994E 02	0.452632517E 01	0.327856168E−00	−0.413961068E−01	0.	0.999999993E−01
0.657999992E 02	0.443769753E 01	0.326697305E−00	−0.457080074E−01	0.407499999E 01	0.
0.689999998E 02	0.381219082E 01	0.331831254E−00	−0.927333571E−01	0.	−0.999999985E−02
0.707999989E 02	0.366038024E 01	0.348589554E−00	−0.115995258E−00	0.333899997E 01	0.
0.754999995E 02	0.365827605E 01	0.434639670E−00	−0.151801050E−00	0.	−0.199999996E−00
0.758999996E 02	0.367384672E 01	0.444403417E−00	−0.153142102E−00	0.335599996E 01	0.
0.808999993E 02	0.392783776E 01	0.588552453E 00	−0.153684869E−00	0.375699997E 01	0.
0.858999990E 02	0.411561452E 01	0.750190347E 00	−0.136236615E−00	0.385699995E 01	0.
0.859999999E 02	0.411754631E 01	0.753359631E 00	−0.135786600E−00	0.	−0.129999995E−00
0.908999994E 02	0.410156414E 01	0.894658349E 00	−0.110665120E−00	0.384599999E 01	0.
0.954999998E 02	0.388636135E 01	0.986762404E 00	−0.826589502E−01	0.	−0.699999988E−01
0.958999991E 02	0.385919407E 01	0.992326975E 00	−0.800342456E−01	0.375699997E 01	0.
0.099999994E 03	0.351563454E 01	0.102388248E 01	−0.512336008E−01	0.	0.199999996E−01

THETA	DSIGMA EX	DPOL EX	CHI SQUARE SIGMA	CHI SQUARE POL	CHI SQUARE TOTAL
0.151999995E 02	0.397999994E 03	0.099999994E 30	0.247771524E−00	0.	0.247771524E−00
0.202999994E 02	0.354999997E 02	0.099999994E 30	0.833843596E 01	0.	0.833843596E 01
0.253999993E 02	0.166999996E 02	0.099999994E 30	0.612634748E 00	0.	0.612634748E 00
0.279999994E 02	0.099999994E 30	0.299999997E−01	0.	0.137748629E 01	0.137748629E 01
0.303999998E 02	0.907999992E 01	0.099999994E 30	0.151578002E−00	0.	0.151578002E−00
0.329999998E 02	0.099999994E 30	0.399999991E−01	0.	0.180344537E−01	0.180344537E−01
0.354999997E 02	0.537999995E 01	0.099999994E 30	0.328498974E 01	0.	0.328498974E 01
0.389999993E 02	0.099999994E 30	0.299999997E−01	0.	0.156901620E 01	0.156901620E 01
0.405999996E 02	0.372999996E 01	0.099999994E 30	0.878443092E 01	0.	0.878443092E 01
0.430000000E 02	0.099999994E 30	0.299999997E−01	0.	0.384781063E 01	0.384781063E 01
0.455999993E 02	0.191000000E 01	0.099999994E 30	0.110886693E 02	0.	0.110886693E 02
0.469999999E 02	0.099999994E 30	0.299999997E−01	0.	0.871015161E 01	0.871015161E 01
0.506999992E 02	0.914999999E 00	0.099999994E 30	0.123029307E−01	0.	0.123029307E−01
0.514999993E 02	0.099999994E 30	0.399999991E−01	0.	0.118072823E 02	0.118072823E 02
0.539999999E 02	0.099999994E 30	0.399999991E−01	0.	0.188061401E 02	0.188061401E 02
0.556999996E 02	0.495999999E−00	0.099999994E 30	0.232838377E−04	0.	0.232838377E−04
0.569999993E 02	0.099999994E 30	0.399999991E−01	0.	0.129395790E 02	0.129395790E 02
0.599999994E 02	0.099999994E 30	0.299999997E−01	0.	0.264876761E 02	0.264876761E 02
0.607999995E 02	0.257999994E−00	0.099999994E 30	0.256912217E 01	0.	0.256912217E 01
0.654999994E 02	0.099999994E 30	0.499999993E−01	0.	0.799714297E 01	0.799714297E 01
0.657999992E 02	0.162999995E−00	0.099999994E 30	0.495124198E 01	0.	0.495124198E 01
0.689999998E 02	0.099999994E 30	0.399999991E−01	0.	0.427800477E 01	0.427800477E 01
0.707999989E 02	0.133999996E−00	0.099999994E 30	0.575213231E 01	0.	0.575213231E 01
0.754999995E 02	0.099999994E 30	0.599999994E−01	0.	0.645316236E 00	0.645316236E 00
0.758999996E 02	0.133999996E−00	0.099999994E 30	0.562634163E 01	0.	0.562634163E 01
0.808999993E 02	0.149999999E−00	0.099999994E 30	0.129713513E 01	0.	0.129713513E 01
0.858999990E 02	0.153999999E−00	0.099999994E 30	0.282009937E 01	0.	0.282009937E 01
0.859999999E 02	0.099999994E 30	0.599999994E−01	0.	0.930132821E−02	0.930132821E−02
0.908999994E 02	0.153999999E−00	0.099999994E 30	0.275396556E 01	0.	0.275396556E 01
0.954999998E 02	0.099999994E 30	0.499999993E−01	0.	0.640996160E−01	0.640996180E−01
0.958999991E 02	0.149999999E−00	0.099999994E 30	0.464161523E−00	0.	0.464161523E−00
0.099999994E 03	0.099999994E 30	0.599999994E−01	0.	0.140950717E 01	0.140950717E 01

L	REAL C(L+1/2)	IMAG C(L+1/2)	REAL C(L−1/2)	IMAG C(L−1/2)
0	−0.149473831E−00	0.621800341E 00	−0.139704145E−00	0.618552327E 00
1	−0.890974633E−01	0.280818105E−00	−0.112762213E−00	0.266789824E−00
2	−0.187045686E−00	0.241325634E−00	−0.174136B25E−00	0.274325125E−00
3	0.670826085E−01	0.135792047E−00	0.546362303E−01	0.848520368E−01
4	0.756759964E−02	0.149855547E−01	0.229124613E−02	0.170210496E−01
5	0.503797509E−02	0.559476413E−02	0.384988777E−02	0.176129699E−02
6	0.628714196E−03	0.123979807E−03	0.362633042E−03	0.807685591E−04
7	0.627130263E−04	0.826003924E−05	0.370286375E−04	0.729130283E−05
8	0.713366367E−05	0.780125931E−06	0.441690803E−05	0.750615060E−06
9	0.130098701E−05	0.808378339E−07	0.101125993E−05	0.797655620E−07
10	0.413292557E−06	0.833891876E−08	0.381933421E−06	0.844628319E−08

VIII. Further Subroutines and Programs in Preparation

The following subroutines are presently being prepared at UCLA:

Subroutine TV

This subroutine is designed to output on CRT and on film various required curves such as $\sigma(\theta)$ vs θ, $\sigma(\theta)/\sigma_c(\theta)$ vs θ, $P(\theta)$ vs θ.

Subroutine RHOBEG

This subroutine will make use of the quantities IIN(L) to allow the numerical integrations to start at different values of ρ depending upon ℓ in order to speed up the numerical integration.

Subroutine FLUX

This subroutine will if desired compute the normalized total wave functions, the scattered flux $\vec{j}$, the divergence and the curl of $\vec{j}$ at specified values of ρ and θ.

All the above subroutines will of course require some modification of the basic program.

The following programs are presently being prepared at UCLA:

Program `SCAT 3`

This program will be similar to program `SCAT 4` except that it will treat incident and target particles of zero spin, thus speeding up the calculation for that case.

Program `SCAT 5`

This is a modified version of program `SCAT 4` offering a simplified input and using only as many ℓ's as may be significant in the C_ℓ's calculations.

Program `SCAT K`

This is a modified version of program `SCAT 4` designed to analyze the scattering of K-mesons against complex nuclei, including the use of an approximate Klein-Gordon equation, relativistic kinematic corrections, and averaging of the cross sections over angles, energies, and representative nuclei.

Program `SCAT 6`

This is a modified version of program `SCAT 4` designed to calculate cross sections and polarization of spin 1 particles scattered by 0 spin targets.

Program `SEEK 4`

This is a program designed to search automatically the parameter space so as to minimize χ^2.

BIBLIOGRAPHY OF DIFFUSE SURFACE OPTICAL MODEL ANALYSES BY MACHINE CALCULATIONS

R. D. Albert, (p, n) Cross Sections and Proton Optical-Model Parameters in 4- to 5.5-MeV Energy Region, UCRL-5488 (1959).

H. J. Amster, Optical Model Evidence for Surface Absorption of Neutrons, Phys. Rev. **113**, 911 (1959).

H. J. Amster and L. M. Culpepper, Surface Modified Nuclear Optical Model: Description of the SUMNUM Code for the NORC Computer, WAPD-TM-87, Bettis Plant of the AEC, Pittsburgh, Pa.

Baker, Byfield, and Rainwater, Theoretical Calculations of the Scattering of π-Mesons by Complex Nuclei, Phys. Rev. **112**, 1773 (1958).

H. R. Beyster, Predictions of Fast Neutrons Scattering Data with a Diffuse Surface Potential Well, LA-2099 (1956).

Beyster, Walt and Salmi, Interaction of 1.0-, 2.5-, 3.25-, and 7.0-MeV Neutrons with Nuclei Phys. Rev. **104**, 1319 (1956).

Bjorklund, Blandford and Fernbach, Analysis of Elastic Scattering and Polarization of 300-MeV Protons, Phys. Rev. **108**, 795 (1957).

F. Bjorklund and S. Fernbach, Elastic Scattering of 7-MeV Neutrons (Theoretical Curves), UCRL-4927-T (1957).

F. Bjorklund and S. Fernbach, Optical-Model Analysis of Scattering of 4.1-, 7-, and 14-MeV Neutrons by Complex Nuclei, Phys. Rev. **109**. 1295 (1958).

F. Bjorklund and S. Fernbach, Exact Phase-Shift Calculation for Nucleon-Nuclear Scattering, UCRL-5028 (1958).

Bjorklund, Fernbach and Sherman, Optical Model of Nucleus with Absorbing Surface, Phys. Rev. **101**, 1832 (1956).

W. B. Cheston and A. E. Glassgold, Elastic Scattering of Alpha-Particles with the Optical Model, Phys. Rev. **106**, 1215 (1957).

Culler, Fernbach and Sherman, Optical Model Analysis of Scattering of 14-MeV Neutrons, Phys. Rev. **101**, 1047 (1956).

Eisberg, Gugelot and Porter, Conference on the Statistical Aspects of the Nucleus, Brookhaven National Laboratory (1955).

W. S. Emmerich, Cross Section Calculations for Fast Neutron Scattering, Westinghouse Research Report 60-94511-6-R17 (1957).

W. S. Emmerich, Optical Model Theory of Neutron Scattering and Reactions, Westinghouse Research Report 6-94511-6-R20 (1958).

Franklin, Margolis and Oberthal, Scattering of μ-Mesons by Nuclei, Phys. Rev. **111**, 296 (1958).

Glassgold, Cheston, Stein, Schuld and Erickson, Analysis of Proton-Nucleus Scattering at 9.8 MeV, Phys. Rev. **106**, 1207 (1958).

A. E. Glassgold, Interaction of Antiprotons with Complex Nuclei, Phys. Rev. **110**, 220 (1958).

A. E. Glassgold and P. J. Kellogg, Proton-Nucleus Scattering at 17 MeV, Phys. Rev. **107**, 1372 (1957).

A. E. Glassgold and P. J. Kellogg, Nuclear Scattering of 40- and 95-MeV Protons, Phys. Rev. **109**, 1291 (1958).

Green, Porter and Saxon, Proceedings of the International Conference on the Nuclear Optical Model, Florida State University, Tallahassee (1959).

G. Igo, Optical-Model Analysis of the Elastic Scattering of Alpha Particles, Phys. Rev. **106**, 126 (1957).

G. Igo, Optical Model Potential at the Nuclear Surface for the Elastic Scattering of Alpha Particles, Phys. Rev. Let. **1**, 72 (1958).

G. Igo, Optical Model Analysis of the Scattering of Alpha Particles from Helium (in press).

G. Igo, Optical-Model Analysis of Excitation Function Data and Theoretical Reaction Cross Sections for Alpha Particles, Phys. Rev. **115**, 1665 (1959).

Igo, Ravenhall, Tiemann, Chupp, Goldhaber, Goldhaber, Lanutti and Thaler, The Scattering of K^+-Mesons in Emulsion, Phys. Rev. **109**, 2133 (1958).

R. Jastrow and I. Harris, Nuclear Cross Sections for the Scattering of Neutrons and Protons, Proceedings of the ONR Decennial Symposium (1957).

Lukyanov, Orlov and Turovstev, Optical Model of the Interaction between Intermediate Energy Neutrons and Nuclei, Nucl. Phys. **8**, 325 (1958).

I. E. McCarthy, Flux of Particles in the Optical Model, Nucl. Phys. **10**, 583 (1959).

Melkanoff, Moszkowski, Nodvik and Saxon, Energy Dependence of the Optical Model Parameters, Phys. Rev. **101**, 507 (1956).

Melkanoff, Nodvik and Saxon, Diffuse-Surface Optical Model Analysis of Elastic Scattering of 17- and 31.5-MeV Protons, Phys. Rev. **106**, 793 (1957).

Melkanoff, Price, Stork and Ticho, Optical Model Analysis of Elastic Scattering of 125-MeV K^+-Mesons in Nuclear Emulsions, Phys. Rev. **113**, 1303 (1959).

University of Minnesota Annual Progress Report 1956-1957, 1957-1958, 1958, University of Minnesota Linear Accelerator Laboratory, Minneapolis, Minn.

J. S. Nodvik and D. S. Saxon, Analysis of Elastic Cross Sections and Polarization of 10 MeV Protons (in press).

C. E. Porter, Nitrogen-Nitrogen Elastic Scattering, Phys. Rev. **112**, 1722 (1958).

H. M. Shey, Scattering of Neutrons by Non-spherical Nuclei, Phys. Rev. **113**, 900 (1959).

R. D. Woods and D. S. Saxon, Diffuse Surface Optical Model for Nucleon-Nuclei Scattering, Phys. Rev. **95**, 577 (1954).

1.E.9.

1.E.3. If an individual Project Gutenberg-tm electronic work is posted with the permission of the copyright holder, your use and distribution must comply with both paragraphs 1.E.1 through 1.E.7 and any additional terms imposed by the copyright holder. Additional terms will be linked to the Project Gutenberg-tm License for all works posted with the permission of the copyright holder found at the beginning of this work.

1.E.4. Do not unlink or detach or remove the full Project Gutenberg-tm License terms from this work, or any files containing a part of this work or any other work associated with Project Gutenberg-tm.

1.E.5. Do not copy, display, perform, distribute or redistribute this electronic work, or any part of this electronic work, without prominently displaying the sentence set forth in paragraph 1.E.1 with active links or immediate access to the full terms of the Project Gutenberg-tm License.

1.E.6. You may convert to and distribute this work in any binary, compressed, marked up, nonproprietary or proprietary form, including any word processing or hypertext form. However, if you provide access to or distribute copies of a Project Gutenberg-tm work in a format other than "Plain Vanilla ASCII" or other format used in the official version posted on the official Project Gutenberg-tm web site (www.gutenberg.org), you must, at no additional cost, fee or expense to the user, provide a copy, a means of exporting a copy, or a means of obtaining a copy upon request, of the work in its original "Plain Vanilla ASCII" or other form. Any alternate format must include the full Project Gutenberg-tm License as specified in paragraph 1.E.1.

1.E.7. Do not charge a fee for access to, viewing, displaying, performing, copying or distributing any Project Gutenberg-tm works unless you comply with paragraph 1.E.8 or 1.E.9.

1.E.8. You may charge a reasonable fee for copies of or providing access to or distributing Project Gutenberg-tm electronic works provided that

- You pay a royalty fee of 20% of the gross profits you derive from the use of Project Gutenberg-tm works calculated using the method you already use to calculate your applicable taxes. The fee is owed to the owner of the Project Gutenberg-tm trademark, but he has agreed to donate royalties under this paragraph to the Project Gutenberg Literary Archive Foundation. Royalty payments must be paid within 60 days following each date on which you prepare (or are legally required to prepare) your periodic tax returns. Royalty payments should be clearly marked as such and sent to the Project Gutenberg Literary Archive Foundation at the address specified in Section 4, "Information about donations to the Project Gutenberg Literary Archive Foundation."

- You provide a full refund of any money paid by a user who notifies you in writing (or by e-mail) within 30 days of receipt that s/he does not agree to the terms of the full Project Gutenberg-tm License. You must require such a user to return or destroy all copies of the works possessed in a physical medium and discontinue all use of and all access to other copies of Project Gutenberg-tm works.

- You provide, in accordance with paragraph 1.F.3, a full refund of any

money paid for a work or a replacement copy, if a defect in the electronic work is discovered and reported to you within 90 days of receipt of the work.

- You comply with all other terms of this agreement for free distribution of Project Gutenberg-tm works.

1.E.9. If you wish to charge a fee or distribute a Project Gutenberg-tm electronic work or group of works on different terms than are set forth in this agreement, you must obtain permission in writing from both the Project Gutenberg Literary Archive Foundation and Michael Hart, the owner of the Project Gutenberg-tm trademark. Contact the Foundation as set forth in Section 3 below.

1.F.

1.F.1. Project Gutenberg volunteers and employees expend considerable effort to identify, do copyright research on, transcribe and proofread public domain works in creating the Project Gutenberg-tm collection. Despite these efforts, Project Gutenberg-tm electronic works, and the medium on which they may be stored, may contain "Defects," such as, but not limited to, incomplete, inaccurate or corrupt data, transcription errors, a copyright or other intellectual property infringement, a defective or damaged disk or other medium, a computer virus, or computer codes that damage or cannot be read by your equipment.

1.F.2. LIMITED WARRANTY, DISCLAIMER OF DAMAGES - Except for the "Right of Replacement or Refund" described in paragraph 1.F.3, the Project Gutenberg Literary Archive Foundation, the owner of the Project Gutenberg-tm trademark, and any other party distributing a Project Gutenberg-tm electronic work under this agreement, disclaim all liability to you for damages, costs and expenses, including legal fees. YOU AGREE THAT YOU HAVE NO REMEDIES FOR NEGLIGENCE, STRICT LIABILITY, BREACH OF WARRANTY OR BREACH OF CONTRACT EXCEPT THOSE PROVIDED IN PARAGRAPH F3. YOU AGREE THAT THE FOUNDATION, THE TRADEMARK OWNER, AND ANY DISTRIBUTOR UNDER THIS AGREEMENT WILL NOT BE LIABLE TO YOU FOR ACTUAL, DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE OR INCIDENTAL DAMAGES EVEN IF YOU GIVE NOTICE OF THE POSSIBILITY OF SUCH DAMAGE.

1.F.3. LIMITED RIGHT OF REPLACEMENT OR REFUND - If you discover a defect in this electronic work within 90 days of receiving it, you can receive a refund of the money (if any) you paid for it by sending a written explanation to the person you received the work from. If you received the work on a physical medium, you must return the medium with your written explanation. The person or entity that provided you with the defective work may elect to provide a replacement copy in lieu of a refund. If you received the work electronically, the person or entity providing it to you may choose to give you a second opportunity to receive the work electronically in lieu of a refund. If the second copy is also defective, you may demand a refund in writing without further opportunities to fix the problem.

1.F.4. Except for the limited right of replacement or refund set forth in paragraph 1.F.3, this work is provided to you 'AS-IS' WITH NO OTHER WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTIBILITY OR FITNESS FOR ANY PURPOSE.

1.F.5. Some states do not allow disclaimers of certain implied warranties or the exclusion or limitation of certain types of damages.

Section 2. Information about the Mission of Project Gutenberg-tm

Project Gutenberg-tm is synonymous with the free distribution of electronic works in formats readable by the widest variety of computers including obsolete, old, middle-aged and new computers. It exists because of the efforts of hundreds of volunteers and donations from people in all walks of life.

Volunteers and financial support to provide volunteers with the assistance they need, are critical to reaching Project Gutenberg-tm's goals and ensuring that the Project Gutenberg-tm collection will remain freely available for generations to come. In 2001, the Project Gutenberg Literary Archive Foundation was created to provide a secure and permanent future for Project Gutenberg-tm and future generations. To learn more about the Project Gutenberg Literary Archive Foundation and how your efforts and donations can help, see Sections 3 and 4 and the Foundation web page at http://www.pglaf.org.

Section 3. Information about the Project Gutenberg Literary Archive Foundation

The Project Gutenberg Literary Archive Foundation is a non profit 501(c)(3) educational corporation organized under the laws of the state of Mississippi and granted tax exempt status by the Internal Revenue Service. The Foundation's EIN or federal tax identification number is 64-6221541. Its 501(c)(3) letter is posted at http://pglaf.org/fundraising. Contributions to the Project Gutenberg Literary Archive Foundation are tax deductible to the full extent permitted by U.S. federal laws and your state's laws.

The Foundation's principal office is located at 4557 Melan Dr. S. Fairbanks, AK, 99712., but its volunteers and employees are scattered throughout numerous locations. Its business office is located at 809 North 1500 West, Salt Lake City, UT 84116, (801) 596-1887, email business@pglaf.org. Email contact links and up to date contact information can be found at the Foundation's web site and official page at http://pglaf.org

For additional contact information:

Dr. Gregory B. Newby
Chief Executive and Director
gbnewby@pglaf.org

Section 4. Information about Donations to the Project Gutenberg Literary Archive Foundation

Project Gutenberg-tm depends upon and cannot survive without wide spread public support and donations to carry out its mission of increasing the number of public domain and licensed works that can be freely distributed in machine readable form accessible by the widest array of equipment including outdated equipment. Many small donations ($1 to $5,000) are particularly important to maintaining tax exempt status with the IRS.

The Foundation is committed to complying with the laws regulating charities and charitable donations in all 50 states of the United States. Compliance requirements are not uniform and it takes a considerable effort, much paperwork and many fees to meet and keep up with these requirements. We do not solicit donations in locations where we have not received written confirmation of compliance. To SEND DONATIONS or determine the status of compliance for any particular state visit http://pglaf.org

While we cannot and do not solicit contributions from states where we have not met the solicitation requirements, we know of no prohibition against accepting unsolicited donations from donors in such states who approach us with offers to donate.

International donations are gratefully accepted, but we cannot make any statements concerning tax treatment of donations received from outside the United States. U.S. laws alone swamp our small staff.

Please check the Project Gutenberg Web pages for current donation methods and addresses. Donations are accepted in a number of other ways including checks, online payments and credit card donations. To donate, please visit: http://pglaf.org/donate

Section 5. General Information About Project Gutenberg-tm electronic works.

Professor Michael S. Hart is the originator of the Project Gutenberg-tm concept of a library of electronic works that could be freely shared with anyone. For thirty years, he produced and distributed Project Gutenberg-tm eBooks with only a loose network of volunteer support.

Project Gutenberg-tm eBooks are often created from several printed editions, all of which are confirmed as Public Domain in the U.S. unless a copyright notice is included. Thus, we do not necessarily keep eBooks in compliance with any particular paper edition.

Most people start at our Web site which has the main PG search facility:

http://www.gutenberg.org

This Web site includes information about Project Gutenberg-tm, including how to make donations to the Project Gutenberg Literary Archive Foundation, how to help produce our new eBooks, and how to subscribe to our email newsletter to hear about new eBooks.