PERFECT V-A : Fortran Programs to Compute Various Properties of the Triton Wave Function Using the RSC5 Potential(Annual Report)

著者	Sasakawa T., Ishikawa S., Sawada T.
journal or publication title	The science reports of the Tohoku University. Ser. 8, Physics and astronomy
volume	3
number	3
page range	268-336
year	1982-12-15
URL	http://hdl.handle.net/10097/25504

PERFECT V-A

Fortran Programs to Compute Various Properties of the Triton Wave Function Using the RSC5 Potential

T. Sasakawa, S. Ishikawa and T. Sawada*

Department of Physics, Faculty of Science
Tohoku University, Sendai 980

* Department of Applied Mathematics, Faculty of Engineering Science
Osaka University, Toyonaka 560

(Received November 17, 1982)

We present the detailed description of the auxiliary codes to PERFECT V that compute the one-body charge form factors, the percentage probabilities, the asymptotic normalization constants, the E1 sum rule, and the two-body correlation function of the triton wave function obtained by PERFECT V using the RSC5 potential.

Keywords: Triton bound state. Reid soft core potential. Interacting S, D-states and spectator s-, d-state. One-body charge form factor. The percentage probabilities. The asymptotic normalization constants. The E1 sum rule. The two-body correlation function.

Table of Contents

§1. Introduction

Based on the formulations presented in reference 1, we have described in our previous report in detail the program named PERFECT V to carry out the iterative calculation of the triton bound state[2]. In the present paper, we present the detailed description of various auxiliary codes to compute other physical quantities of interest of the triton wave function. They are named NFORM, ASYM5, SUM5 and CORR5, that carry out the computations of the one-body charge form factors and the percentage probabilities(NFORM), the asymptotic normalization constants(ASYM5), the enhancement factor κ of the E1 sum rule (SUM5), and the two-body correlation function(CORR5), all utilizing the triton wave functions output from PERFECT V, or the extension of them to large x. Necessary formulas to compute these quantities are given in reference 1.

Although in the following sections, we present these codes separately to be used independent of each other, the user will find a strong similarity in parts of the codes that deal with the extension of the wave functions to larger x. In fact, these are the most tedious parts of the computation, but they are essentially the same as the computation done in subroutine RHS of PERFECT V. It would be a simple matter for the user to rearrange these codes in such a way that he prepares a single file in advance on which the wave functions are stored up to the desired large distance x, so that he can compute all the above mentioned quantities all at once.

§2. Computation of One-Body Charge Form Factors and Percentage Probabilities

2-a. One-Body Charge Form Factors

The one-body charge form factors of ^3H and ^3He are computed by using the formulas given in section 4 of I. We have the main program of PERFECT V[2] output on File LFCHI the wave functions - $\Theta_\alpha(q,x)$ and $\phi_\alpha(q,x)$ of Eqs.(II-2.12) and (II-2.16) that correspond to $(\Phi(12,3)+\Phi(23,1)+\Phi(31,2))$ of Eq.(I-2-40a) and $-\Phi(12,3)$ of Eq.(I-2-38a), respectively. As explained at the beginning of section I-4, we must utilize the (12,3) component $\phi_\alpha(q,x)$ to compute the one-body charge form factors rather than the total wave function $\Theta_\alpha(q,x)$.

Necessary formulas for computation of the relevant quantities $\langle F_i^{uv} \rangle$ (i=1 to 4) have already been derived in section 4 of I. Since all these quantities are expressed as integrals involving rapidly oscillating functions, the evaluation of them requires careful treatments, or the resulting charge form factors deteriorate rapidly at large momentum transfers. We describe below how to compute these quantities.

First of all, we need to calculate $\rho_\alpha(x,y)$ of Eq.(I-4-29)

$$\rho_\alpha(x,y) = \int_0^{p_{MALL}} dp \, \frac{\phi_\alpha(q,x)}{x^L} \frac{u_\ell(p,y)}{y^\ell} \tag{2.1}$$

to sufficiently far distances $x \leqq x_G$ and $y \leqq y_G$. To take care accurately the possible oscillatory behavior of $u_\ell(p,y)$, we carry out the p-integration by means of the p-spline as described in section II-5. Thus, we introduce $Y_\ell^{(t)}(y,p_k)$ of Eq.(II-5.9):

$$Y_\ell^{(t)}(y,p_k) = \int_{p_k}^{p_{k+1}} dp' \, \frac{u_\ell(p',y)}{y^\ell} \, S^{(t)}(p'-p_k) \tag{2.2}$$

where $S^{(t)}(z)$ (t=1 to 4) are the spline functions (see Eq.(II-5.3)). After extending $\phi_\alpha(q,x)$ to $x_{max} < x \leqq x_G$ by using the known asymptotic form (see Eq.(II-3.10)), we compute the p-spline coefficients $a_\alpha^{(t)}(x;p_k)$ by (see Eq.(II-5.7))

$$\frac{\phi_\alpha(q,x)}{x^L} \cong \sum_{t=1}^{4} a_\alpha^{(t)}(x;p_k) \; S^{(t)}(p-p_k) \qquad (p_k \leqq p \leqq p_{k+1}) \tag{2.3}$$

Then $\rho_\alpha(x,y)$ of Eq.(2.1) can be approximated by

$$\rho_\alpha(x,y) \cong \sum_{p_k} \sum_{t=1}^{4} a_\alpha^{(t)}(x,p_k) \; Y_\ell^{(t)}(y,p_k) \qquad (x \leqq x_G, \; y \leqq y_G) \tag{2.4}$$

Perhaps we should mention that, unlike $\rho_{\alpha'}^{\beta(m)}(x,y)$ of Eq.(II-5.8), we need not to separate the p-interval in Eq.(2.1) at $p=p_M$ since $\phi_\alpha(q,x)$ is continuous there as a function of p.

The quantity $\langle F_1^{uv} \rangle$ of Eq.(I-4-40) is given by

$$\langle F_1^{uv} \rangle = \delta_{u,v} \sum_\alpha^u F_{\alpha\alpha}^{(1)}(Q) \tag{2.5}$$

with

$$F_{\alpha\alpha}^{(1)}(Q) = \int_0^{y_M} dy \; j_0(\tfrac{2}{3} Qy) \; f_\alpha(y) \tag{2.5a}$$

where

$$f_\alpha(y) = \int_0^{x_M} dx \; x^{2L+2} \; y^{2\ell+2} [\rho_\alpha(x,y)]^2 \tag{2.5b}$$

where x_M and y_M are appropriately chosen upper limits (see Eq.(2.24) below), and Q is the momentum transfer. The value of Q extends as far as 10 fm^{-1}, and therefore the function $j_0(\frac{2}{3} Qy)$ will oscillate many times as y runs from 0 to the upper limit y_M. To evaluate accurately the y-integral with such an oscillating integrand, we utilize the method of quadrature by a spline interpolation. The y-spline coefficients $b_\alpha^{(t)}(y_i)$ (t=1 to 4) of $f_\alpha(y)$ are defined by

$$f_\alpha(y) \cong \sum_{t=1}^{4} b_\alpha^{(t)}(y_i)\ S^{(t)}(y-y_i) \tag{2.6}$$

Introducing

$$I^{(t)}(Q;y_i) = \int_{y_i}^{y_{i+1}} dy\ j_0(\tfrac{2}{3} Qy)\ S^{(t)}(y-y_i) \quad , \tag{2.7}$$

we find for the y-integral in Eq.(2.5a)

$$F_{\alpha\alpha}^{(1)}(Q) \cong \sum_{y_i=y_0=0}^{y_{M-1}} \sum_{t=0}^{4} b_\alpha^{(t)}(y_i)\ I^{(t)}(Q;y_i) \tag{2.8}$$

For $\langle F_2^{uv}\rangle$ of Eq.(I-4-44), we write

$$\langle F_2^{uv}\rangle = \sum_\alpha^u \sum_{\alpha'}^v F_{\alpha\alpha'}^{(2)}(Q) \tag{2.9}$$

with

$$F_{\alpha\alpha'}^{(2)}(Q) = \sum_\lambda N_{\alpha\alpha',\lambda} G_{\alpha\alpha',\lambda}(Q) \tag{2.9a}$$

where

$$G_{\alpha\alpha',\lambda}(Q) = \int_0^{x_M} dx\ j_\lambda(\tfrac{Q}{2} x)\ g_{\alpha\alpha',\lambda}(Q;x) \tag{2.9b}$$

and

$$N_{\alpha\alpha',\lambda} = (-)^{\ell+L'}\ \hat{L}\hat{\ell}\langle L0\lambda0|L'0\rangle\langle \ell0\lambda0|\ell'0\rangle\hat{\lambda}^2$$

$$\times \sum_{L_0S_0} N_\alpha^{(L_0S_0)} N_{\alpha'}^{(L_0S_0)}\ (-)^{L_0} \begin{Bmatrix} L & \lambda & L' \\ \ell' & L_0 & \ell \end{Bmatrix} \tag{2.10}$$

$$g_{\alpha\alpha',\lambda}(Q;x) = \int_0^{y_M} dy\ j_\lambda(\tfrac{Q}{3} y)\ f_{\alpha\alpha'}(x,y) \tag{2.11}$$

with

$$f_{\alpha\alpha'}(x,y) = x^{L+2}\ y^{\ell+2}\ \rho_\alpha(x,y)\ x^{L'}y^{\ell'}\ \rho_{\alpha'}(x,y) \tag{2.12}$$

In Eq.(2.10), the factor $N_\alpha^{(L_0S_0)}$ is given by Eq.(I-3-4). The y-integration in Eq.(2.11) involving $j_\lambda(\frac{Q}{3}y)$ should be done by the spline method. Thus, the y-spline coefficients $c_{\alpha\alpha'}^{(t)}(x;y_i)$ of $f_{\alpha\alpha'}(x,y)$ is determined by using subroutine SPCOEF: for $y_i \leqq y \leqq y_{i+1}$

$$f_{\alpha\alpha'}(x,y) \cong \sum_{t=1}^{4} c_{\alpha\alpha'}^{(t)}(x;y_i)\ S^{(t)}(y-y_i) \tag{2.13}$$

Introducing

$$J_\lambda^{(t)}(Q;y_i) = \int_{y_i}^{y_{i+1}} dy\ j_\lambda(\tfrac{Q}{3}y)\ S^{(t)}(y-y_i) \qquad (\lambda=0,2,4) \tag{2.14}$$

we find

$$g_{\alpha\alpha',\lambda}(Q;x) \cong \sum_{y_i=y_0=0}^{y_{M-1}} \sum_{t=1}^{4} c_{\alpha\alpha'}^{(t)}(x;y_i)\ J_\lambda^{(t)}(Q;y_i) \tag{2.15}$$

Here we need only $\lambda=0,2,4$ due to the factor $N_{\alpha\alpha',\lambda}$ of Eq.(2.10) and the fact that there are even parity states only in RSC5. To carry out the x-integration in Eq.(2.9) involving $j_\lambda(\frac{Q}{2}x)$, we further utilize the method of spline interpolation. Denoting the spline coefficients of $g_{\alpha\alpha',\lambda}(Q;x)$ by $d_{\alpha\alpha',\lambda}^{(t)}(Q;x_j)$, we have

$$g_{\alpha\alpha',\lambda}(Q;x) \cong \sum_{t=1}^{4} d_{\alpha\alpha',\lambda}^{(t)}(Q;x_j)\ S^{(t)}(x-x_j) \qquad (x_j \leqq x \leqq x_{j+1}) \tag{2.16}$$

Further we introduce

$$K_\lambda^{(t)}(Q;x_j) = \int_{x_j}^{x_{j+1}} j_\lambda(\tfrac{Q}{2}x)\ S^{(t)}(x-x_j)\ dx \qquad (\lambda=0,2,4) \tag{2.17}$$

Then we find for the x-integral in Eq.(2.9b)

$$G_{\alpha\alpha',\lambda}(Q) \cong \sum_{x_j=x_0=0}^{x_{M-1}} \sum_{t=1}^{4} d_{\alpha\alpha',\lambda}^{(t)}(Q;x_j)\ K_\lambda^{(t)}(Q;x_j) \tag{2.18}$$

The quantity $\langle F_3^{uv}\rangle$ of Eq.(I-4-49) is given by

$$\langle F_3^{uv}\rangle = \sum_\alpha^u \sum_{\alpha'}^v F_{\alpha\alpha'}^{(3)}(Q) \tag{2.19}$$

with

$$F^{(3)}_{\alpha\alpha'}(Q) = \sum_{L_0 S_0} \bar{N}^{(L_0 S_0)}_{\alpha\alpha'} Y^{L_0}_{\alpha\alpha'}(Q) \tag{2.19a}$$

where

$$Y^{L_0}_{\alpha\alpha'}(Q) = \int_0^{Y_M} dy\; j_0(\tfrac{2}{3} Qy)\; h^{L}_{\alpha\alpha'}(y), \tag{2.19b}$$

$$\bar{N}^{(L_0 S_0)}_{\alpha\alpha'} = N^{(L_0 S_0)}_{\alpha} N^{(L_0 S_0)}_{\alpha'} \langle (S\tfrac{1}{2}) S_0 M_{S_0} (12,3) \,|\, (S'\tfrac{1}{2}) S_0 M_{S_0} (31,2) \rangle$$

$$= N^{(L_0 S_0)}_{\alpha} N^{(L_0 S_0)}_{\alpha'} (-)^{S} \hat{S}\hat{S}' \begin{Bmatrix} \tfrac{1}{2} & \tfrac{1}{2} & S' \\ \tfrac{1}{2} & S_0 & S \end{Bmatrix} \tag{2.20}$$

and

$$h^{L}_{\alpha\alpha'}(y) = \int_0^{x_M} dx\; x^{L+2} y^{\ell+2} \rho_\alpha(x,y)\; U^{L_0}_{\alpha\alpha'}(Q=0;x,y) \tag{2.21}$$

where $U^{L_0}_{\alpha\alpha'}(Q=0;x,y)$ is given by Eq.(I-4-51) with Q=0. From Eq.(I-3-27)

$$U^{L_0}_{\alpha\alpha'}(Q;x,y) = \sum_{\substack{a=0\\(b=L'-a)}}^{L'} \sum_{\substack{c=0\\(d=\ell'-c)}}^{\ell'} \sum_{\gamma} x^{a+c} y^{b+d} K^{\alpha'}_{\gamma}(x,y) R^{ac\gamma}_{(L\ell,L'\ell')L_0} \tag{2.22}$$

with

$$K^{\alpha'}_{\gamma}(x,y) = \frac{1}{2}\int_{-1}^{1} d(\cos\theta_{xy})\; P_\gamma(\cos\theta_{xy}) \rho_{\alpha'}(x'',y'') \tag{2.23}$$

and with the coefficients $R^{ac\gamma}_{(L\ell,L'\ell')L_0}$ of Eq.(I-3-38).(see section 10 of II on Code RAM.) By a double spline interpolation we can compute $\rho_{\alpha'}(x'',y'')$ as in the main program (see Eqs.(II-5.10)∿(II-5.12)). Unlike during iterations of the main program, here we have to compute $U^{L_0}_{\alpha\alpha'}(Q;x,y)$ just once. Therefore there is no merit of splitting the y-integration interval in Eq.(19b) into two parts as we do in Eq.(II-3.13). Furthermore, here we must integrate over x as well as y. Thus, we prepare $\rho_\alpha(x,y)$ at the beginning to sufficiently large distances $x \leqq x_G$ and $y \leqq y_G$, and compute $K^{\alpha'}_{\gamma}(x,y)$ of Eq.(2.23) up to large distances $x \leqq x_M$ and $y \leqq y_M$. The relations between (x_G,y_G) and (x_M,y_M) are (see Eq.(II-3.11))

$$x_G = \frac{1}{2} x_M + y_M , \qquad y_G = \frac{3}{4} x_M + \frac{1}{2} y_M \tag{2.24}$$

The rest of the computation of $U^{L_0}_{\alpha\alpha'}(Q=0;x,y)$ proceeds just as in subroutine RHS (see section 6-b of II). After obtaining $U^{L_0}_{\alpha\alpha'}(Q=0;x,y)$, we perform the x-integration in Eq.(2.21). Since $\rho_\alpha(x,y)$ and $U^{L_0}_{\alpha\alpha'}(Q=0;x,y)$ are slowly varying functions of x, we may use a conventional method of quadrature over x to find $h^{L_0}_{\alpha\alpha'}(y)$. This function, in turn, is approximated by a y-spline interpolation,

$$h^{L_0}_{\alpha\alpha'}(y) \cong \sum_{t=1}^{4} u^{(t)}_{\alpha\alpha' L_0}(y_i)\, S^{(t)}(y-y_i) \qquad (y_i \leq y \leq y_{i+1}) \tag{2.25}$$

and the y-integration in Eq.(2.19b) is carried out by means of the y-spline method:

$$Y^{L}_{\alpha\alpha'}(Q) = \sum_{y_i=y_0}^{y_{M-1}} \sum_{t=1}^{4} u^{(t)}_{\alpha\alpha',L_0}(y_i)\, I^{(t)}(Q;y_i) \tag{2.26}$$

where $I^{(t)}(Q;y_i)$ is given by Eq.(2.7).

Finally, we need to compute $\langle F^{uv}_4\rangle$ of Eq.(I-4-49). This can be written as

$$\langle F^{uv}_4\rangle = \sum_{\alpha}^{u} \sum_{\alpha'}^{v} F^{(4)}_{\alpha\alpha'}(Q) \tag{2.27}$$

with

$$F^{(4)}_{\alpha\alpha'}(Q) = \sum_{L_0 S_0} \bar{N}^{(L_0 S_0)}_{\alpha\alpha'} \int_0^{y_N} dy\, h^{L_0}_{\alpha\alpha'}(Q;y) \tag{2.27a}$$

where

$$h^{L_0}_{\alpha\alpha'}(Q;y) = \int_0^{x_M} dx\, x^{L+2}\, y^{\ell+2}\, \rho_\alpha(x,y)\, U^{L_0}_{\alpha\alpha'}(Q;x,y), \tag{2.28}$$

$$U^{L_0}_{\alpha\alpha'}(Q;x,y) = \sum_{\substack{a=0\\(b=L'-a)}}^{L'} \sum_{\substack{c=0\\(d=\ell'-c)}}^{\ell'} \sum_{\gamma} x^{a+c}\, y^{b+d}\, K^{\alpha'}_{\gamma}(Q;x,y)\, R^{ac\gamma}_{(L\ell,L'\ell')L_0} \tag{2.29}$$

and

$$K^{\alpha'}_{\gamma}(Q;x,y) = \frac{1}{2}\int_{-1}^{1} d(\cos\theta_{xy})\, P_\gamma(\cos\theta_{xy})\, j_0(QR)\, \rho_{\alpha'}(x'',y'') \tag{2.30}$$

with

$$R = \sqrt{\frac{x^2}{4} + \frac{y^2}{9} - \frac{xy}{3}\cos\theta_{xy}} \tag{2.31}$$

Unlike $U^{L_0}_{\alpha\alpha'}(Q=0;x,y)$, $U^{L_0}_{\alpha\alpha'}(Q;x,y)$ of Eq.(2.29) depends on Q through $j_0(QR)$ in $K^{\alpha'}_{\gamma}(Q;x,y)$ of Eq.(2.30). However, due to the $\cos\theta_{xy}$-integration in $K^{\alpha'}_{\gamma}(Q;x,y)$, it is expected that the oscillatory behavior of $j_0(QR)$ is smeared out after the $\cos\theta$-integration (which requires a small $\cos\theta$-mesh size depending on the value of Q) so that the resulting $U^{L0}_{\alpha\alpha'}(Q;x,y)$ is a relatively slowly varying function of x and y even for large values of Q. We should be able to tell whether this expectation is born out or not by simply varying the x-mesh size in Eq.(2.28) and watch the change in the resulting values of $\langle F^{uv}_4 \rangle$.

Having obtained $\langle F^{uv}_i \rangle$, i=1 to 4, the one-body charge form factors of ^3H and ^3He can be calculated by using Eqs.(I-4-19,20,21,25 and 26). In terms of $F^{(i)}_{\alpha\alpha'}(Q)$, i=1 to 4, of Eqs.(2.5a), (2.9a), (2.19a) and (2.27a) we have for the quantities appearing in Eqs.(I-4-25 and 26)

$$\langle F^{SS}_i \rangle = \sum_{k=2}^{5} \sum_{k'=2}^{5} F^{(i)}_{kk'}(Q) \tag{2.32a}$$

$$\langle F^{AA}_i \rangle = F^{(i)}_{11}(Q) \tag{2.32b}$$

$$\langle F^{SA}_i \rangle = \sum_{k=2}^{5} F^{(i)}_{k1}(Q) \tag{2.32c}$$

and

$$\langle F^{AS}_i \rangle = \sum_{k=2}^{5} F^{(i)}_{1k}(Q) \tag{2.32d}$$

One way of checking the accuracy of the resulting $\langle F^{uv}_i \rangle$ is to see if the symmetry properties of Eq.(I-4-27) are satisfied.

2-b. Percentage Probabilities

All the formulas necessary to compute the percentage probabilities of the Derrick-Blatt classification of the ^3H wave function have been derived in section 5 and Appendix of I for the case of RSC5.

We need the following integrals defined by Eqs.(I-5-19) and (I-5-21)(see also Eq.(I-A-3)).

$$I_{kk'} = \delta_{LL'} \delta_{\ell\ell'} N_k N_{k'} \int_0^\infty dx\, x^2 \int_0^\infty dy\, y^2\, x^L y^\ell \rho_\alpha(x,y) \rho_{\alpha'}(x,y) \tag{2.33a}$$

and

$$K_{kk'} = N_k N_{k'} \int_0^\infty dx\, x^2 \int_0^\infty dy\, y^2\, x^L y^\ell \rho_\alpha(x,y)\, U^{L0}_{\alpha\alpha'}(Q=0;x,y) \tag{2.33b}$$

Here, $U^{L0}_{\alpha\alpha'}(Q=0;x,y)$ is the same as in Eq.(2.21), $\rho_\alpha(x,y)$ is given by Eq.(2.1), and $N_k = N^{(L_0 S_0)}_\alpha$ of Eq.(I-3-4). The correspondence between the index k and

$\{\alpha, (L_0 S_0)\}$ is given by Table A-1 of I. Comparing Eq.(2.33a) with Eqs.(2.9b), (2.11) and (2.12) we ses that

$$I_{kk'} = \delta_{LL'} \delta_{\ell\ell'} N_k N_{k'} G_{\alpha\alpha',\lambda=0}(Q=0) \; \delta_{L_0 L_0'} \; \delta_{S_0 S_0'}$$

$$= \delta_{LL'} \delta_{\ell\ell'} N_k N_{k'} \frac{F^{(2)}_{\alpha\alpha'}(Q=0)}{N_{\alpha\alpha',\lambda=0}} \delta_{L_0 L_0'} \; \delta_{S_0 S_0'} \tag{2.34}$$

Similarly, comparing Eq.(2.33b) with Eqs.(2.19b) and (2.21), we find

$$K_{kk'} = N_k N_{k'} \; Y^{L_0}_{\alpha\alpha'}(Q=0) \; \delta_{L_0 L_0'} \; \delta_{S_0 S_0'} \tag{2.35}$$

Therefore $I_{kk'}$ and $K_{kk'}$ have essentially been obtained already. In terms of these quantities, we can compute the percentage probabilities as per Eqs.(I-A-11,13,14,17 and 18).

2-c. General Description of Code NFORM

The calculations of one-body charge form factors and the percentage probabilities of ^3H and ^3He (without the Coulomb interaction) with RSC5 are carried out in the code named NFORM. It consists of the following parts.

MAIN routine

Subroutines and functions FORM1, FORM2, NRHS3, NRHS4, FORM, PARAM, KOEF2, XTABLE, PTABLE, YTAB, BESSEL, SPLINE, SPCOEF, CLEBSH, F6J, NINEJ, FYFORM, and PL

The code requires the following data files.

File LFRAM, File LFCHI.

Of eighteen subroutines and functions listed above, PTABLE, SPLINE, SPCOEF, CLEBSH, F6J, NINEJ, and PL are identical to those used in the main program, and hence no listings of them are given. The user will find Subroutine FYFORM to run parallel to part of Code FYFI5, Subroutines NRHS3 and NRHS4 similar to Subroutine RHS in the main program. For detailed description of these subroutines, we refer the user to reference II.

Since there are ample comment cards provided in various key places within the code, we shall not give detailed description of each subroutine. Together with the description given in the flow chart below, they snould be sufficient to giude the user to read through each subroutine.

2-d. MAIN routine

The function of this routine is best explained by the flow chart below.

Start

#1 Initialization (Input $|E|$, the x-and y- mesh numbers. See Eq.(2.24): $x_{JMX}=x_{max}$, $x_{JMXG}=x_M$, $x_{JMXGG}=x_G$, $y_{JMYM}=y_M$ and $y_{JMYMM}=y_G$)

#2 Call PARAM (Assign values to key-indices)

Call FYFORM(PTABLE,YTAB,BESSEL)
(Set up the same p-table as used in the main program, construct the y-table for $0\leq y\leq y_G$, and compute $Y_\ell^{(t)}(y,p_k)$ of Eq.(2.2))

Read File LFRAM (Read in $R^{ac\gamma}_{(L\ell,L'\ell')L_0}$, a,b,c,d,γ of Eq.(2.22))

Call KOEF2(CLEBSH,F6J,NINEJ)
(Compute $N_\alpha^{(L_0S_0)}$ of Eq.(I-3-4), $N_{\alpha\alpha',\lambda}$ of Eq.(2.10), and $\bar{N}_{\alpha\alpha'}^{(L_0S_0)}$ of Eq.(2.20))

#3 Read File LFCHI (Read in $-\Theta_\alpha(q,x)$ and $\phi_\alpha(q,x)$ of Eqs.(II-2.12) and (II-2.16))

#4 Call XTABLE (Set up the same x-table as used in the main program, and extend it to $x_{max}\leq x\leq x_G$)

#5 (Compute various spline matrices by calling subroutine SPLINE)

#6 Construct $\rho_\alpha(x,y)$ of Eq.(2.1) for all α as follows:
First, extend $\phi_\alpha(q,x)$ out to $x_{max}\leq x\leq x_G$.
Next, construct $\phi_\alpha(q,x)/x^L$ and compute the p-spline coefficients $a_\alpha^{(t)}(x;p_k)$ of Eq.(2.3) by calling subroutine SPCOEF.
Compute $\rho_\alpha(x,y)$ as per Eq.(2.4), extrapolate it to x=0 assuming the form $a+bx^2$. The results are stored as 'RHO(JX,JY,α)', 'JX'=1 and 'JY'=1 corresponding to x=0 and y=0.

- -(Do-loop over the momentum transfer Q)

#7 Construct $I^{(t)}(Q,y_i)$ and $J_\lambda^{(t)}(Q,y_i)$ of Eqs.(2.7) and (2.14).

-(Do-loop over the (grand) y-mesh points, y_i)

(i) Set up fine y-mesh in the interval $y_{i-1} \leq y \leq y_i$ (with $y_0=0.$) These are taken equidistant with the number of intervals 'NY' chosen so that there are approximately 30 points in one period of sin (Qy). If 'NY' is less than 10, it is set to 10.

(ii) Process the Simpson quadratures in the interval $y_{i-1} \leq y \leq y_i$ for $I^{(t)}(Q;y_i)$ and $J_\lambda^{(t)}(Q;y_i)$.

-(End of the do-loop over y_i)

(iii) Multiply the (fine) y- mesh size 'DY' to the results obtained above to finish the Simpson quadratures. This completes the computation of $I^{(t)}(Q;y_i)$ and $J_\lambda^{(t)}(Q;y_i)$ which are stored as 'FI1(y_i,Q,t)' and 'FI2(y_i,Q,t,λ)', respectively.

#8 Construct $K_\lambda^{(t)}(Q;x_j)$ of Eq.(2.17).

-(Do-loop over the (grand) x-mesh points, x_j)

(i) Set up fine x-mesh in the interval $x_{j-1} \leq x \leq x_j$ (with $x_0=0$). The fine x-mesh number 'NX' is chosen with the same principle as for 'NY' in #7 above.

(ii) Process the Simpson quadrature in the interval $x_{j-1} \leq x \leq x_j$ for $K_\lambda^{(t)}(Q;x_j)$.

-(End of the do-loop over x_j)

(iii) Multiply the (fine) x-mesh size 'DX' to the results above to finish the Simpson quadrature. This completes the computation of $K_\lambda^{(t)}(Q,x_j)$ of Eq.(2.17). The results are stored as 'FK(x_j, Q,t,λ)'.

-(End of the do-loop over Q)

#9 Call FORM1(SPCOEF) (Compute $F^{(1)}_{\alpha\alpha'}$ as per Eqs.(2.5) to (2.8), and store the results as 'FF1(Q,α)'.)

Call FORM2(SPCOEF) (Compute $F^{(2)}_{\alpha\alpha'}$ as per Eqs.(2.9) to (2.18), and store the results as 'FF2(Q,α,α')'.

-(Do-loop over the initial state index α')

#10 (Compute the double -spline coefficients of $\rho_{\alpha'}(x,y)$)

| #11 { Compute the y-integrals in Eqs.(2.19b) and (2,27a).
Call NRHS3(PL,SPCOEF) (Compute $Y^{L0}_{\alpha\alpha'}(Q)$ of Eq.(2.19b).)
Call NRHS4(PL,SPCOEF) (Compute the y-integral in Eq.(2.27a).) }

'-(End of the do-loop over α')

#12 Call FORM (Compute $F^{(3)}_{\alpha\alpha'}$ of Eq.(2.19) and $F^{(4)}_{\alpha\alpha'}$ of Eq.(2.27))

#13 { Compute $<F^{uv}_{i}>$ using Eq.(2.32) and construct $F_a(Q)$ and $F_b(Q)$ by Eqs.(I-4-25 and 26). }

#14 { Compute nucleon charge form factors using the formulas given by Janssens et al.(Phys.Rev. 142('66) 992).
Calculate the ^3H and ^3He charge form factors by Eqs.(I-4-19,20 and 21). }

#15 { Obtaine $I_{kk'}$ and $K_{kk'}$ by Eqs.(2.34) and (2.35), and compute percentage probabilities as per Eqs.(I-A-11,13,14,17 and 18). }

Output the overall normalization constant 'FNORM'.

Stop

```
 20.224    ** CODE NFORM ***                                                    LABEL
(09-30-82)    OPTIONS: INTER,OPT=1,NROUND,NDLR,ALC,NASTER,INLINE=0,LSTIN,LNO,NREST

      C**** *** CODE NFORM ***
      C**** TRITON CHARGE FORM FACTOR (WITH RSC5)
      C**** REFERENCES  (SRT=SCIENCE REPORT OF TOHOKU UNIV.)
      C        I: FORMULATIONS       (T.SAWADA+T.SASAKAWA)
      C                               SRT SER.8,VOL.3,NO.1(1982)1
      C       II: PERFECT V          (T.SASAKAWA+S.ISHIKAWA+T.SAWADA)
      C                               SRT(1982)
      C      III: THIS REPORT
            COMMON/AAAAA/XR(60),TR1(60),DT,AH2,JMX,JMXG,JMXGG
            COMMON/BBBBB/YY(47),TY1(47),JMYM,JMYMM,AY2,FY(48,12,4,2)
            COMMON/DDDDD/RHO(61,48,5),FI1(48,40,4),FI2(48,40,4,3),FK(60,40,4,
           &  3)
            COMMON/ZZZZZ/ZFIT(43),EZ(41,3),HZ(42),VZ(42),WZ(42),ZA(42),ZB(42)
           &  ,ZC(42),ZD(42)
            COMMON/UUUUU/UFIT(47),EU(45,3),HU(46),VU(46),WU(46),UA(46),UB(46)
           &  ,UC(46),UD(46)
            COMMON/XXXXX/XFIT(61),EX(59,3),HX(60),VX(60),WX(60),XA(60),XB(60)
           &  ,XC(60),XD(60)
            COMMON/YYYYY/YFIT(48),EY(46,3),HY(47),VY(47),WY(47),YA(47),YB(47)
           &  ,YC(47),YD(47)
            COMMON/CCCCC/DSP(4,4,60,48)
            COMMON/FFFFF/FF1(40,5),FF2(40,5,5),FF3(40,5,5),FF4(40,5,5)
            COMMON/RAMRAM/COEF(8,8),RAM(111),LA(111),LB(111),LC(111),LD(111),
           &  LG(111)
            COMMON/MMMMM/MAL1(5),KST(14),KED(14),IGX(5),IG(5,5),IPX(5),
           &  LLF(19),LF(19),MSF(19),NR(19)
            DIMENSION QQ(40),AQ(12),AK2(12),EXK(60,12,2)
            DIMENSION PFIT(13),EP(11,3),HP(12),VP(12),WP(12),PA(12),PB(12),
           &  PC(12),PD(12)
            DIMENSION PSI(60,12),FIT(61),SSUM(4,3),S(4),SUMX(4,3),AY(61,48,4)
           &  ,FCHHE3(40),FII(8,8),FKK(8,8),PROB(6)
           &  ,TSI(30,5,12),GSI(30,5,12),
           &  FCHH3(40),RSUM3(40,19),RSUM4(40,19),KEY(20),ZSUM(4),CF2(5,5,3),
           &  CF3(8,8),FAC(8)
            DATA IQX/40/,
           &       QQ/  0.0,  1.0,  1.5,  2.0,  2.5,  3.0,  3.25,  3.5,  3.7,
           &            3.8,  3.9,  4.0,  4.1,  4.2,  4.3,  4.4,  4.5,  4.6,
           &            4.75,  5.0,  5.25,  5.5,  5.75,  6.0,  6.25,  6.5,
           &            6.75,  7.0,  7.25,  7.5,  7.75,  8.0,  8.25,  8.5,
           &            8.75,  9.0,  9.25,  9.5,  9.75,  10.0/
            DATA KEY/11,21,51,12,22,52,33,43,83,34,44,84,15,25,55,66,77, 38,
           &  48,88/
      C**** FOR 'KEY(20)', SEE TABLE 3 OF II. THESE ARE THE LOCATIONS OF
      C**** TWENTY NON-ZERO MATRIX ELEMENTS OF THE 8*8 MATRIX IN
      C**** TABLE II-3.
            KOB=5
            WRITE(6,1060) KOB
       1060 FORMAT('1*** CODE NFORM *** TRITON CHARGE FORM FACTOR WITH RSC',I1)
      C-01
            BEI=6.80
            JMX=30
            JMXG=46
            JMXGG=59
            JMYM=42
            SQ3=SQRT(3.)
      C-02
            CALL PARAM
            CALL FYFORM(AQ,MALL)
            LFRAM=21
            REWIND LFRAM
            READ(LFRAM) COEF,RAM,LA,LB,LC,LD,LG
```

```
                CALL KOEF2(CF2,CF3,FAC)
                WRITE(6,1005) BEI,JMX,JMXG,JMXGG,JMYM,JMYMM
          1005  FORMAT('0BEI=',1PE13.5,'  JMX,JMXG,JMXGG,JMYM,JMYMM=',5I5)
         C-03
                LFCHI=27
                REWIND LFCHI
                READ(LFCHI) TSI,GSI
         C      DO 500 MS=1,KOB
         C      WRITE(6,4010) MS
         C4010  FORMAT(///'1*** GSI(MS=',I3,')')
         C      DO 500 IM=1,MALL
         C 500  WRITE(6,4020) IM,(GSI(J,MS,IM),J=1,JMX)
         C4020  FORMAT(' IM=',I3/(2X,1P10E12.4))
                HM=1./41.47
                DO 5 I=1,MALL
     1        5 AK2(I)=SQRT(HM*BEI+0.75*AQ(I)**2)
         C-04
                CALL XTABLE(30,JMXGG,XR,TR1,DT,12.,6.,9.,0.3)
                WRITE(6,1000) (XR(J),J=1,JMXGG)
          1000  FORMAT(///'0XR=',1P10E12.5/(4X,1P10E12.5))
                AH2=2.*DT
                WRITE(6,1010) (YY(J),J=1,JMYMM)
          1010  FORMAT('0YY=',1P10E12.5/(4X,1P10E12.5))
                WRITE(6,1040) IQX,(QQ(I),I=1,IQX)
          1040  FORMAT('0IQX=',I3/' QQ=',1P10E12.5/(4X,1P10E12.5))
         C-05
         C****  VARIOUS SPLINE MATRICES
                PFIT(1)=0.
                DO 10 I=1,MALL
     1       10 PFIT(I+1)=AQ(I)
                CALL SPLINE(PFIT,MALL-1,EP,HP,VP,WP,MALL-1,PA,PB,PC,PD)
                XFIT(1)=0.
                DO 20 I=1,JMXGG
     1       20 XFIT(I+1)=XR(I)
                CALL SPLINE(XFIT,JMXGG-1,EX,HX,VX,WX,59,XA,XB,XC,XD)
                YFIT(1)=0.
                DO 30 I=1,JMYMM
     1       30 YFIT(I+1)=YY(I)
                CALL SPLINE(YFIT,JMYMM-1,EY,HY,VY,WY,46,YA,YB,YC,YD)
                ZFIT(1)=0.
                DO 35 I=1,JMYM
     1       35 ZFIT(I+1)=YY(I)
                CALL SPLINE(ZFIT,JMYM-1,EZ,HZ,VZ,WZ,41,ZA,ZB,ZC,ZD)
                UFIT(1)=0.
                DO 25 I=1,JMXG
     1       25 UFIT(I+1)=XR(I)
                CALL SPLINE(UFIT,JMXG-1,EU,HU,VU,WU,45,UA,UB,UC,UD)
         C-06
         C****  CONSTRUCT THE RHO-FUNCTION OF EQ.(III-2.1).
                X1Q=XR(1)**2
                X2Q=XR(2)**2
                XQ12=X2Q-X1Q
         C****  FIRST, EXTEND THE GSI-FUNCTIONS UP TO XR(JMXGG).
                J1=JMX
                DO 50 I=1,MALL
     1          DO 40 JX=J1,JMXGG
     2          AKX=AK2(I)*XR(JX)
     2          EXK(JX,I,1)=EXP(-AKX)
     2       40 EXK(JX,I,2)=EXK(JX,I,1)*(3./AKX**2+3./AKX+1.)
     1       50 CONTINUE
                DO 160 MA=1,KOB
     1          LL=1
     1          IF(MA.EQ.3.OR.MA.EQ.5) LL=2
     1          DO 120 I=1,MALL
```

```
2          ASYM=GSI(JMX,MA,I)/EXK(JMX,I,LL)*XR(JMX)
2          DO 80 J=1,J1
3       80 PSI(J,I)=GSI(J,MA,I)*XR(J)
2          DO 90 J=J1,JMXGG
3       90 PSI(J,I)=ASYM*EXK(J,I,LL)
2      120 CONTINUE
1 C        WRITE(6,2020) MA
1 C2020 FORMAT('1*** PSI(MA=',I3,')')
1 C        DO 125 I=1,MALL
1 C 125 WRITE(6,2030) I,(PSI(J,I),J=1,JMXGG)
1 C2030 FORMAT('0IM=',I3/(' ',1P10E12.4))
1 C**** P-SPLINE OF 'PSI' DIVIDED BY X**(L+1)
1          LLP=2*(LL/2)+1
1          LP=1
1          IF(MA.GE.4) LP=2
1          DO 150 JX=1,JMXGG
2          FIT(1)=0.
2          DO 130 I=1,MALL
3      130 FIT(I+1)=PSI(JX,I)/XR(JX)**LLP
2          CALL SPCOEF(PFIT,FIT,MALL-1,EP,HP,VP,WP,PA,PB,PC,PD,MALL-1)
2          DO 150 JY=1,JMYMM+1
3          SUMA=0.
3          DO 140 I=1,MALL
4      140 SUMA=SUMA+PA(I)*FY(JY,I,1,LP)+PB(I)*FY(JY,I,2,LP)+PC(I)
3        &         *FY(JY,I,3,LP)+PD(I)*FY(JY,I,4,LP)
3      150 RHO(JX+1,JY,MA)=SUMA
1 C**** EXTRAPOLATE THE 'RHO'-FUNCTION OF EQ.(III-2.1) TO X=0
1 C**** ASSUMING THE FORM    A+B*X**2
1          DO 154 JY=1,JMYMM+1
2      154 RHO(1,JY,MA)=(RHO(2,JY,MA)*X2Q-RHO(3,JY,MA)*X1Q)/XQ12
1 C        WRITE(6,2040) MA
1 C2040 FORMAT('1*** RHO(MA=',I3,')')
1 C        DO 155 JY=1,JMYMM+1
1 C 155 WRITE(6,2050) JY,(RHO(JX,JY,MA),JX=1,JMXGG+1)
1 C2050 FORMAT('0JY=',I3/(' ',1P10E12.4))
1      160 CONTINUE
  C**** CONSTRUCT THE FUNCTIONS OF Q; 'FI1','FI2', AND 'FK'.
           DO 270 IQ=1,IQX
1 C-07
1          Y0=0.
1          DO 210 JY=1,JMYM
2          Y1=YY(JY)
2          NY=(Y1-Y0)*QQ(IQ)*5.
2          NY=2*(NY/2)
2          IF(NY.LT.10) NY=10
2          DY=(Y1-Y0)/NY
2          Y=Y0-DY
2          CCC=1./3.
2          DO 170 N=1,4
3          ZSUM(N)=0.
3          DO 170 K=1,3
4      170 SSUM(N,K)=0.
2          DO 190 I=1,NY+1
3          DDD=CCC
3          IF(I.EQ.1.OR.I.EQ.NY+1) DDD=DDD/2.
3          Y=Y+DY
3          YQ2=QQ(IQ)*Y/3.
3          YQ1=2.*YQ2
3          U=Y-YFIT(JY)
3          FX=EXP(U)
3          EXV=1./FX
3          S(1)=(FX-EXV)/2.
3          S(2)=S(1)+EXV
```

```
3          S(3)=U
3          S(4)=1.
3          IF(YQ1.LT.1.E-10) GO TO 171
3          FJ0=SIN(YQ1)/YQ1
3          GO TO 181
3     171  FJ0=1.
3     181  DO 182 N=1,4
4     182  ZSUM(N)=ZSUM(N)+DDD*FJ0*S(N)
3          IF(YQ2.LT.1.E-10) GO TO 175
3          FJ0=SIN(YQ2)/YQ2
3          FJ1=FJ0/YQ2-COS(YQ2)/YQ2
3          FJ2=3.*FJ1/YQ2-FJ0
3          FJ3=5.*FJ2/YQ2-FJ1
3          FJ4=7.*FJ3/YQ2-FJ2
3          GO TO 185
3     175  FJ0=1.
3          FJ2=0.
3          FJ4=0.
3     185  DO 180 N=1,4
4          TEMP=DDD*S(N)
4          SSUM(N,1)=SSUM(N,1)+TEMP*FJ0
4          SSUM(N,2)=SSUM(N,2)+TEMP*FJ2
4     180  SSUM(N,3)=SSUM(N,3)+TEMP*FJ4
3     190  CCC=1.-CCC
2          DO 200 N=1,4
3          FI1(JY,IQ,N)=2.*DY*ZSUM(N)
3          DO 200 K=1,3
4     200  FI2(JY,IQ,N,K)=2.*DY*SSUM(N,K)
2     210  Y0=Y1
1  C-08
1          X0=0.
1          DO 260 JX=1,JMXG
2          X1=XR(JX)
2          NX=(X1-X0)*QQ(IQ)*5.
2          NX=2*(NX/2)
2          IF(NX.LT.10) NX=10
2          DX=(X1-X0)/NX
2          X=X0-DX
2          CCC=1./3.
2          DO 220 K=1,3
3          DO 220 N=1,4
4     220  SUMX(N,K)=0.
2          DO 240 I=1,NX+1
3          DDD=CCC
3          IF(I.EQ.1.OR.I.EQ.NX+1) DDD=DDD/2.
3          X=X+DX
3          XQ1=QQ(IQ)*X/2.
3          U=X-XFIT(JX)
3          FX=EXP(U)
3          EXV=1./FX
3          S(1)=(FX-EXV)/2.
3          S(2)=S(1)+EXV
3          S(3)=U
3          S(4)=1.
3          IF(XQ1.LT.1.E-10) GO TO 225
3          FJ0=SIN(XQ1)/XQ1
3          FJ1=FJ0/XQ1-COS(XQ1)/XQ1
3          FJ2=3.*FJ1/XQ1-FJ0
3          FJ3=5.*FJ2/XQ1-FJ1
3          FJ4=7.*FJ3/XQ1-FJ2
3          GO TO 230
3     225  FJ0=1.
3          FJ2=0.
3          FJ4=0.
```

```
 3    230 DO 235 N=1,4
 4        TEMP=DDD*S(N)
 4        SUMX(N,1)=SUMX(N,1)+TEMP*FJ0
 4        SUMX(N,2)=SUMX(N,2)+TEMP*FJ2
 4    235 SUMX(N,3)=SUMX(N,3)+TEMP*FJ4
 3    240 CCC=1.-CCC
 2        DO 250 N=1,4
 3        DO 250 K=1,3
 4    250 FK(JX,IQ,N,K)=2.*DX*SUMX(N,K)
 2    260 X0=X1
 1    270 CONTINUE
    C**** NOW, WE ARE READY TO COMPUTE 'FF1' AND 'FF2'.
    C-09
          CALL FORM1(IQX,QQ,KOB)
          CALL FORM2(IQX,QQ,CF2,KOB)
          DO 285 IQ=1,IQX
 1        DO 285 M=1,19
 2        RSUM3(IQ,M)=0.
 2    285 RSUM4(IQ,M)=0.
    C**** DO-LOOP OVER THE INITIAL STATE ALPHA'.
          DO 330 NA=1,KOB
 1  C**** COMPUTE THE DOUBLE-SPLINE COEFFICIENTS OF 'RHO'.
 1  C-10
 1        DO 300 JX=1,JMXGG+1
 2        DO 280 JY=1,JMYMM+1
 3    280 FIT(JY)=RHO(JX,JY,NA)
 2        CALL SPCOEF(YFIT,FIT,JMYMM-1,EY,HY,VY,WY,YA,YB,YC,YD,46)
 2        DO 290 JY=1,JMYMM
 3        AY(JX,JY,1)=YA(JY)
 3        AY(JX,JY,2)=YB(JY)
 3        AY(JX,JY,3)=YC(JY)
 3    290 AY(JX,JY,4)=YD(JY)
 2    300 CONTINUE
 1        DO 320 L=1,4
 2        DO 320 JY=1,JMYMM
 3        DO 310 JX=1,JMXGG+1
 4    310 FIT(JX)=AY(JX,JY,L)
 3        CALL SPCOEF(XFIT,FIT,JMXGG-1,EX,HX,VX,WX,XA,XB,XC,XD,59)
 3        DO 320 JX=1,JMXGG
 4        DSP(1,L,JX,JY)=XA(JX)
 4        DSP(2,L,JX,JY)=XB(JX)
 4        DSP(3,L,JX,JY)=XC(JX)
 4    320 DSP(4,L,JX,JY)=XD(JX)
 1  C**** COMPUTE THE Y-INTEGRALS IN EQS.(III-2.19B) AND (III-2.27A)
 1  C-11
 1        NAR=NA
 1        CALL NRHS3(IQX,QQ,RSUM3,NAR)
 1        CALL NRHS4(IQX,QQ,RSUM4,NAR)
 1    330 CONTINUE
    C**** COMPUTE 'FF3' AND 'FF4' OF EQS.(III-2.19) AND (III-2.27).
    C-12
          CALL FORM(CF3,RSUM3,IQX,FF3,KEY)
          CALL FORM(CF3,RSUM4,IQX,FF4,KEY)
          WRITE(6,1110)
     1110 FORMAT('0*** FF3 ***')
          DO 332 IQ=1,IQX
 1    332 WRITE(6,1115) QQ(IQ),((FF3(IQ,N,M),M=1,5),N=1,5)
     1115 FORMAT(' ',1P6E15.5/(' ',15X,1P5E15.5))
          WRITE(6,1120)
     1120 FORMAT('0*** FF4 ***')
          DO 334 IQ=1,IQX
 1    334 WRITE(6,1115) QQ(IQ),((FF4(IQ,N,M),M=1,5),N=1,5)
    C**** DO-LOOP OVER THE MOMENTUM TRANSFER
```

```
                 WRITE(6,2000)
            2000 FORMAT('1',7X,'QQ',9X,'FCHP',8X,'FCHN',9X,'FA',10X,'FB', 7X,
                &  'FCHHE3',7X,'FCHH3',9X,'SS',10X,'AA',10X,'AS',10X,'SA')
                 DO 370 IQ=1,IQX
      1 C*** COMPUTE THE AVERAGE OVER F-SUB-UV USING EQS.(III-2.32A,B,C,D)
      1 C-13
      1          F1SS=FF1(IQ,2)+FF1(IQ,3)+FF1(IQ,4)+FF1(IQ,5)
      1          F1AA=FF1(IQ,1)
      1          F2SS=0.
      1          F3SS=0.
      1          F4SS=0.
      1          DO 340 NA=2,KOB
      2          DO 340 NB=2,KOB
      3          F2SS=F2SS+FF2(IQ,NA,NB)
      3          F3SS=F3SS+FF3(IQ,NA,NB)
      3      340 F4SS=F4SS+FF4(IQ,NA,NB)
      1          F2AA=FF2(IQ,1,1)
      1          F3AA=FF3(IQ,1,1)
      1          F4AA=FF4(IQ,1,1)
      1          F3AS=0.
      1          F3SA=0.
      1          F4AS=0.
      1          F4SA=0.
      1          DO 350 NA=2,KOB
      2          F3AS=F3AS+FF3(IQ,1,NA)
      2          F3SA=F3SA+FF3(IQ,NA,1)
      2          F4AS=F4AS+FF4(IQ,1,NA)
      2      350 F4SA=F4SA+FF4(IQ,NA,1)
      1 C**** COMPUTE 'FA' AND 'FB' AS PER EQS.(I-4.25,26).
      1          FA=F1SS+0.5*F2SS+1.5*F2AA-2.*F3SS+2.*SQ3*F3SA +0.5*F4SS+
      1         &      SQ3*F4SA-1.5*F4AA
      1          FB=F1AA+1.5*F2SS+0.5*F2AA-2.*F3AA-2.*SQ3*F3AS -1.5*F4SS+
      1         &      SQ3*F4SA+0.5*F4AA
      1 C-14
      1 C**** NUCLEON CHARGE FORM FACTOR (JANSSENS ET AL, PHYS.REV.142('66) 992)
      1          QS=QQ(IQ)**2
      1          GES=0.5*(2.5/(1.+QS/15.7)-1.6/(1.+QS/26.7)+0.1)
      1          GEV=0.5*(1.16/(1.+QS/8.19)-0.16)
      1          FCHP=GES+GEV
      1          FCHN=GES-GEV
      1 C**** HE-3 CHARGE FORM FACTOR (TIMES 2)
      1          FL=0.5*(3.*FA+FB)
      1          FQ=2.*FB
      1          FCHHE3(IQ)=2.*FCHP*FL+FCHN*FQ
      1          FCHH3(IQ) =2.*FCHN*FL+FCHP*FQ
      1          IF(IQ.EQ.1) GO TO 335
      1          FCHHE3(IQ)=FCHHE3(IQ)/FCHHE3(1)
      1          FCHH3(IQ)=FCHH3(IQ)/FCHH3(1)
      1      335 WRITE(6,2010) QQ(IQ),FCHP,FCHN,FA,FB,FCHHE3(IQ),FCHH3(IQ),
      1         &      F1SS,F1AA
      1     2010 FORMAT(' ',1P9E12.4)
      1          WRITE(6,2011) F2SS,F2AA,F3SS,F3AA,F3AS,F3SA,F4SS,F4AA,F4AS,
      1         &      F4SA
      1     2011 FORMAT(' ',84X,1P2E12.4,2(/' ',84X,1P4E12.4))
      1      370 CONTINUE
        C**** PERCENTAGE PROBABILITIES
        C-15
        C**** FIRST, COMPUTE I-SUB-KK' AND K-SUB-KK' OF EQS.(III-2.34) AND
        C**** (III-2.35) (SEE ALSO TABLE 3 OF II FOR K-SUB-KK').
                 DO 375 K=1,8
      1          DO 375 KP=1,8
      2          FII(K,KP)=0.
      2      375 FKK(K,KP)=0.
                 DO 355 K=1,4
```

```
1   355 FII(K,K)=FAC(K)**2*FF2(1,K,K)/CF2(K,K,1)
        TEMP=FF2(1,5,5)/CF2(5,5,1)
        DO 365 K=5,8
1   365 FII(K,K)=FAC(K)**2*TEMP
        FII(1,2)=FAC(1)*FAC(2)*FF2(1,1,2)/CF2(1,2,1)
        FII(2,1)=FAC(2)*FAC(1)*FF2(1,2,1)/CF2(2,1,1)
        DO 385 MM=1,20
1       M=MM
1       IF(MM.GE.17) M=MM-1
1       KA=KEY(MM)/10
1       KB=KEY(MM)-10*KA
1   385 FKK(KA,KB)=FAC(KA)*FAC(KB)*RSUM3(1,M)
        WRITE(6,3000) ((FII(KA,KB),KB=1,8),KA=1,8)
   3000 FORMAT('0FII'/(5X,1P8E13.5))
        WRITE(6,3010) ((FKK(KA,KB),KB=1,8),KA=1,8)
   3010 FORMAT('0FKK'/(5X,1P8E13.5))
  C**** NOW, COMPUTE PERCENTAGE PROBABILITIES AS PER EQS.(I-A-11,13,14,17,
  C****    AND 18)
        PROB(1)=1.5*(FII(1,1)+FII(2,2)+FII(5,5)-2.*FII(1,2)) +3.*(FKK(1,1)
      &  +FKK(2,2)+FKK(5,5)+FKK(2,5)+FKK(5,2) -FKK(1,2)-FKK(1,5)-FKK(2,1)
      &  -FKK(5,1))
        PROB(2)=1.5*FII(6,6)+3.*FKK(6,6)
        PROB(3)=0.75*(FII(1,1)+FII(2,2)+FII(5,5)+2.*FII(1,2) -FKK(1,1)-
      &  FKK(2,2)-FKK(5,5)-FKK(2,5)-FKK(5,2) -FKK(1,2)-FKK(1,5)-FKK(2,1)-
      &  FKK(5,1))
        PROB(4)=0.75*(FII(6,6)-FKK(6,6))
        PROB(5)=1.5*(FII(7,7)-FKK(7,7))
        PROB(6)=1.5*(FII(3,3)+FII(4,4)+FII(8,8)-FKK(3,3)-FKK(4,4) -FKK(8,
      &  8)-FKK(3,4)-FKK(3,8)-FKK(4,3)-FKK(4,8)-FKK(8,3) -FKK(8,4))
        DO 390 K=3,6
1   390 PROB(K)=2.*PROB(K)
        FNORM=0.
        DO 400 K=1,6
1   400 FNORM=FNORM+PROB(K)
        DO 410 K=1,6
1   410 PROB(K)=PROB(K)/FNORM
        WRITE(6,3020) FNORM,(PROB(K),K=1,6)
   3020 FORMAT('0FNORM=',1PE13.5/'  PROB=',1P6E13.5)
        STOP
        END
        SUBROUTINE FYFORM(AQ,MALL)
  C**** THIS COMPUTE (I) THE P-TABLE, (II) THE Y-TABLE, AND
  C**** (III) THE Y-FUNCTION OF EQ.(III-2.2).(SEE 'CODE FYFI5'
  C**** FOR DETAILES)
        COMMON/BBBBB/YY(47),TY1(47),JMYM,JMYMM,AY2,FY(48,12,4,2)
        DIMENSION AQ(*),AQ1(12),U(48,2),S(4),SUM(48,4,2)
  C-01
        WRITE(6,1000)
   1000 FORMAT('0*** FYFORM ***')
        MALL=12
        IMX=6
        PIMX=0.750428
        CALL PTABLE(AQ,AQ1,DK,PIMX,IMX,MALL)
        JMY=47
        JMYMM=JMY
        CALL YTAB(YY,TY1,JMY,2.,30.,18.,0.3,DTY)
        AY2=2.*DTY
  C**** DO-LOOP OVER THE GRAND P-MESH.
        P0=0.
        DO 150 JM=1,MALL
1       JMR=JM
1       P1=AQ(JM)
1 C-02
1 C**** DECIDE THE FINER P'-MESH FOR THE P' INTEGRATION.
```

```
1          NP=INT((P1-P0)*YY(JMY)/0.3)
1          NP=2*(NP/2)
1          IF(NP.LT.4) NP=4
1          NP1=NP+1
1          DP=(P1-P0)/NP
1  C-03
1  C****   INITIALIZE THE P' INTEGRATION.
1          IF(JM.EQ.1) GO TO 30
1          DO 20 LL=1,2
2          DO 20 JY=1,JMY+1
3          SUM(JY,1,LL)=0.
3          SUM(JY,2,LL)=U(JY,LL)/6.
3          SUM(JY,3,LL)=0.
3          SUM(JY,4,LL)=SUM(JY,2,LL)
3       20 CONTINUE
1          GO TO 50
1       30 DO 40 JY=1,JMY+1
2          DO 40 LL=1,2
3          DO 40 N=1,4
4          SUM(JY,N,LL)=0.
4       40 CONTINUE
1       50 P=P0
1  C-04
1          CCC=2./3.
1  C****   DO LOOP OVER THE FINER P-MESH(I.E.,P')
1          DO 70 JP=2,NP1
2          DDD=CCC
2          IF(JP.EQ.NP1) DDD=DDD/2.
2          P=P+DP
2  C****   CALCULATE THE SPECTATOR FUNCTION 'U' AT P' AND AT EVERY 'Y'.
2  C****   U(Y)=F(P,Y)/Y**L, WHERE F(P,Y) IS THE NORMALIZED SPHERICAL
2  C****   BESSEL FUNCTION.
2          CALL BESSEL(P,U,YY,JMY)
2  C****   FIND THE SPLINE FUNCTION 'S' AND CARRY OUT THE SIMPSON QUADRATOR
2  C****   FOR THE P' INTEGRATION.
2          EX=EXP(P-P0)
2          S(1)=(EX-1./EX)/2.
2          S(2)=S(1)+1./EX
2          S(3)=P-P0
2          S(4)=1.
2          DO 60 LL=1,2
3          DO 60 JY=1,JMY+1
4          DO 60 N=1,4
5          SUM(JY,N,LL)=SUM(JY,N,LL)+DDD*U(JY,LL)*S(N)
5       60 CONTINUE
2          CCC=1.-CCC
2       70 CONTINUE
1  C-05
1          DO 80 LL=1,2
2          DO 80 JY=1,JMY+1
3          DO 80 N=1,4
4          FY(JY,JM,N,LL)=2.*DP*SUM(JY,N,LL)
4       80 CONTINUE
1          P0=P1
1      150 CONTINUE
   C       DO 400 LL=1,2
   C       DO 400 JM=3,MALL,3
   C       WRITE(6,4001) JM,JMYMM
   C4001   FORMAT('0*** JM=' I3,' ***  JMYMM=',I3)
   C       DO 410 M=1,4
   C 410   WRITE(6,4002) M,(FY(JY,JM,M,LL),JY=1,JMY+1)
   C4002   FORMAT('0FY(M=',I2,')'/(' ',1P10E12.4))
   C 400   CONTINUE
           RETURN
           END
```

```
              SUBROUTINE YTAB(YY,YR1,JMY,YEM,YM,YTM,TYC,DTY)
        C**** Y-TABLE.  SEE EQS.(II-8.2) AND (II-8.3).
              DIMENSION YY(*),YR1(*)
              SYO=(YTM-TYC*YM)*YM/(YEM*YM-YTM)
              TYO=YEM*SYO/TYC
              DTY=YTM/JMY
              CTO=TYC*TYO
              CSO2=2.*TYC*SYO
              CSO4=2.*CSO2
              DTDYO=TYC*TYO/SYO
              T=0.
              DO 40 I=1,JMY
  1           T=T+DTY
  1           A=CTO-T
  1           T1=SQRT(A*A+CSO4*T)
  1           YY(I)=(-A+T1)/(2.*TYC)
  1           DTDY=TYC*(YY(I)**2+2.*SYO*YY(I)+TYO*SYO)/(YY(I)+SYO)**2
  1 C         DTDY2=TYC*2.*SYO*(SYO-TYO)/(YY(I)+SYO)**3
  1           YR1(I)=1./DTDY
  1 C         YR2(I)=YR1(I)**2
  1 C         YR5(I)=-DTDY2*YR2(I)
  1        40 CONTINUE
              WRITE(6,1000) JMY,YEM,YM,YTM,TYC,DTY,SYO,TYO
         1000 FORMAT('OJMY=',I5,'  YEM,YM,YTM,TYC=',1P4E13.5/'ODTY,SYO,TYO=',
             &    1P3E13.5)
        C       WRITE(6,1010) (YY(I),I=1,JMY)
        C1010   FORMAT('OYY=',1P10E13.5/(4X,1P10E13.5))
              RETURN
              END
              SUBROUTINE BESSEL(P,U,YR,JMYM)
        C**** COMPUTE THE NORMALIZED BESSEL FUNCTION FJL(P,Y) DIVIDED BY Y**L
              DIMENSION U(48,2),YR(*)
              DATA PI2/7.9788456E-01/
              DO 20 J=1,JMYM
  1           I=J+1
  1           Z=P*YR(J)
  1           IF(Z.GT.1.E-5) GO TO 10
  1           U(I,1)=1.-Z**2/6.
  1           U(I,2)=(1.-Z**2/14.)/15.
  1           GO TO 20
  1        10 SN=SIN(Z)
  1           CO=COS(Z)
  1           U(I,1)=SN/Z
  1           U(I,2)=((3./Z**2-1.)*SN/Z-(3./Z**2)*CO)/Z**2
  1        20 CONTINUE
              U(1,1)=1.
              U(1,2)=1./15.
              FAC=P*PI2
              FAQ=FAC*P**2
              DO 30 J=1,JMYM
  1           I=J+1
  1           U(I,1)=FAC*U(I,1)
  1        30 U(I,2)=FAQ*U(I,2)
              U(1,1)=FAC*U(1,1)
              U(1,2)=FAQ*U(1,2)
              RETURN
              END
              SUBROUTINE KOEF2(CF2,CF3,FAC)
        C**** 'FAC(I)'(I=1 TO 8) = THE N COEFFICIENTS OF EQ.(II-3.3).
        C****                        (SEE TABLE 1 OF II FOR THE INDEX I).
        C**** 'CF2(N,M,L)'(L=1 TO 3) = THE N COEFFICIENTS OF EQ.(III-2.10) FOR
        C****                                STATES(N,M). L=1,2,3 FOR LAMBDA=0,2,4.
        C**** 'CF3(I,I')'(I,I'=1 TO 8) = THE N-BAR COEFFICIENTS OF EQ.(III-2.20).
        C****                                   (SEE TABLE 3 OF II FOR THE INDICES I
```

```
         C****                                          AND I'.)
                    DIMENSION CF2(5,5,3),CF3(8,8),FAC(8),FACLOG(100),LL(5),L(5),JJ(5)
                  &  ,J(5),IS(5),LLL(8),KSS(8),NLS(5),M(9),LAMBDA(3),SUMLS(3),ILS(5)
                    DATA LL/0,0,2,0,2/, L/0,0,0,2,2/, JJ/0,1,1,1,1/, J/1,1,1,3,3/,
                  &  IS/0,1,1,1,1/, LLL/0,0,2,2,0,1,1,2/, KSS/1,1,3,3,1,1,3,3/,
                  &  NLS/1,2,3,4,8/, ILS/1,2,3,4,5/, LAMBDA/0,2,4/
                    DO 10 NA=1,5
     1              DO 10 NB=1,5
     2              DO 10 NL=1,3
     3           10 CF2(NA,NB,NL)=0.
         C****  FIRST, COMPUTE THE N FACTOR OF EQ.(II-3.3)
                    FACLOG(1)=0.
                    FACLOG(2)=0.
                    F1=1.
                    DO 15 I=3,100
     1              F1=F1+1.
     1           15 FACLOG(I)=FACLOG(I-1)+ALOG(F1)
                    DO 20 NA=1,5
     1              JJA=2*JJ(NA)
     1              JA=J(NA)
     1              M(1)=2*LL(NA)
     1              M(2)=2*L(NA)
     1              M(3)=2*IS(NA)
     1              M(4)=1
     1              M(7)=JJA
     1              M(8)=JA
     1              M(9)=1
     1              NLSA=NLS(NA)
     1              ILSA=ILS(NA)
     1              DO 20 K=ILSA,NLSA
     2              LLLA=2*LLL(K)
     2              KSSA=KSS(K)
     2              FAC(K)=SQRT((JJA+1.)*(JA+1.)*(LLLA+1.)*(KSSA+1.))
     2              M(5)=LLLA
     2              M(6)=KSSA
     2              CALL NINEJ(FACLOG, M,UNINE)
     2           20 FAC(K)=UNINE*FAC(K)
                    WRITE(6,1000) (FAC(K),K=1,8)
               1000 FORMAT('0*** KOEF2 ***'/' FAC=',1P8E13.5)
         C****  NEXT, COMPUTE THE N FACTOR OF EQ.(III-2.10)
                    DO 70 NA=1,5
     1              LLA=2*LL(NA)
     1              LA=2*L(NA)
     1              NLSA=NLS(NA)
     1              ILSA=ILS(NA)
     1              DO 70 NB=1,5
     2              LLB=2*LL(NB)
     2              LB=2*L(NB)
     2              NLSB=NLS(NB)
     2              ILSB=ILS(NB)
     2              DO 30 NL=1,3
     3           30 SUMLS(NL)=0.
     2              DO 50 KA=ILSA,NLSA
     3              DO 50 KB=ILSB,NLSB
     4              IF(LLL(KA).NE.LLL(KB)) GO TO 50
     4              IF(KSS(KA).NE.KSS(KB)) GO TO 50
     4              DO 40 NL=1,3
     5              LAM=2*LAMBDA(NL)
     5              CALL F6J(FACLOG,LLA,LAM,2*LLL(KA),LB,LLB,LA,RAC)
     5           40 SUMLS(NL)=SUMLS(NL)+FAC(KA)*FAC(KB)*(-1.)**LLL(KA)*RAC
     4           50 CONTINUE
     2              FAK=(-1.)**((LA+LLB)/2)*SQRT((LA+1.)*(LLA+1.))
     2              DO 60 NL=1,3
     3              LAM=2*LAMBDA(NL)
```

```
3        CALL CLEBSH(LA,LAM,LB,0,0,0,FACLOG,CLEBA)
3        CALL CLEBSH(LLA,LAM,LLB,0,0,0,FACLOG,CLEBB)
3     60 CF2(NA,NB,NL)=FAK*CLEBA*CLEBB*SUMLS(NL)*(LAM+1.)
2     70 CONTINUE
         DO 80 NL=1,3
1        WRITE(6,1010) NL
1   1010 FORMAT('OCF2(NL=',I2,')')
1     80 WRITE(6,1020) ((CF2(NA,NB,NL),NB=1,5),NA=1,5)
    1020 FORMAT((5X,1P5E15.5))
   C**** FINALLY, COMPUTE THE N-BAR OF EQ.(III-2.20)
         DO 85 KA=1,8
1        DO 85 KB=1,8
2     85 CF3(KA,KB)=0.
         DO 100 NA=1,5
1        ISA=2*IS(NA)
1        NLSA=NLS(NA)
1        ILSA=ILS(NA)
1        DO 100 NB=1,5
2        ISB=2*IS(NB)
2        NLSB=NLS(NB)
2        ILSB=ILS(NB)
2        DO 90 KA=ILSA,NLSA
3        DO 90 KB=ILSB,NLSB
4        IF(LLL(KA).NE.LLL(KB)) GO TO 90
4        IF(KSS(KA).NE.KSS(KB)) GO TO 90
4        CALL F6J(FACLOG,1,1,KSS(KA),1,ISB,ISA,RAC)
4        CF3(KA,KB)=(-1.)**IS(NA)*SQRT((ISA+1.)*(ISB+1.))
4       &                   *RAC *FAC(KA)*FAC(KB)
4     90 CONTINUE
2    100 CONTINUE
         WRITE(6,1030) ((CF3(KA,KB),KB=1,8),KA=1,8)
    1030 FORMAT('OCF3'/(5X,1P8E15.5))
         RETURN
         END
         SUBROUTINE PARAM
   C**** ASSIGN VALUES TO KEY-INDICES
         COMMON/MMMMM/MAL1(5),KST(14),KED(14),IGX(5),IG(5,5),IPX(5),LLF(19)
        &  , LF(19),MSF(19),NR(19)
         DIMENSION JAL1(5),JST(14),JED(14),JGX(5),JG(5,5),JPX(5), JLF(19),
        &  JF(19),JSF(19),JR(19)
         EQUIVALENCE (MAL1(1),JAL1(1)),(KST(1),JST(1)),(KED(1),JED(1)),
        &  (IGX(1),JGX(1)),(IG(1,1),JG(1,1)),(IPX(1),JPX(1)), (LLF(1),
        &  JLF(1)),(LF(1),JF(1)),(MSF(1),JSF(1)),(NR(1),JR(1))
   C**** FOR 'IGX,IG,IPX,MAL1,KST AND KED', SEE THE COMMENT CARDS
   C**** AT THE BEGINNING OF SUBROUTINE RHS OF THE MAIN PROGRAM.
   C**** FOR THE QUANTITIES BELOW, SEE TABLE 2 OF II.
   C**** NR(19) = N(M)      :THE KEY-INDEX N
   C**** LF(19) = THE FINAL L (SPECTATOR)
   C**** LLF(19) = THE FINAL L (INTERACTING PAIR)
   C**** MSF(19) = THE FINAL STATE INDEX (ALPHA)
         DATA JGX/2,2,4,4,5/,IG/0,2,3*0,0,2,3*0,0,1,2,3,0,0,1,2,3,0,0,1,2,
        &  3,4/, JPX/3,3,3,3,7/, JAL1/1,4,7,10,13/, JST/1,2,3,6,9,13,16,
        &  19,23,33,52,69,81,93/, JED/1,2,5,8,12,15,18,22,32,51,68,80,92,
        &  111/, JR/1,1,2,1,1,2,3,4,5,6,7,8,9,9,10,11,12,13,14/, JF/0,0,2,
        &  0,0,2,0,2,2,0,2,2,0,0,2,2,0,2,2/ JLF/0,0,2,0,0,2,2,0,2,2,0,2,0,
        &  0,2,2,2,0,2/, JSF/1,2,5,1,2,5,3,4,5,3,4,5,1,2,5,5,3,4,5/
         RETURN
         END
         SUBROUTINE FORM1(IQX,QQ,KOB)
   C**** COMPUTE 'FF1(Q,N)'(N=1 TO 5) OF EQ.(III-2.5A)
         COMMON/AAAAA/XR(60),TR1(60),DT,AH2,JMX,JMXG,JMXGG
         COMMON/BBBBB/YY(47),TY1(47),JMYM,JMYMM,AY2,FY(48,12,4,2)
         COMMON/DDDDD/RHO(61,48,5),FI1(48,40,4),FI2(48,40,4,3),FK(60,40,4,
        &  3)
```

```
            COMMON/ZZZZZ/ZFIT(43),EZ(41,3),HZ(42),VZ(42),WZ(42),ZA(42),ZB(42)
           &  ,ZC(42),ZD(42)
            COMMON/FFFFF/FF1(40,5),FF2(40,5,5),FF3(40,5,5),FF4(40,5,5)
            DIMENSION FIT(61),QQ(*),LL(5),L(5)
            DATA LL/0,0,2,0,2/, L/0,0,0,2,2/
      C**** DO-LOOP OVER THE STATE
            DO 50 NA=1,KOB
 1    C**** INTEGRATION OVER X OF EQ.(III-2.5B)
 1          KX=2*LL(NA)+2
 1          KY=2*L(NA)+2
 1          DO 20 JY=1,JMYM
 2          SUM=0.
 2          CCC=2./3.
 2          DO 10 JX=1,JMXG
 3          SUM=SUM+TR1(JX)*RHO(JX+1,JY+1,NA)**2*CCC*XR(JX)**KX
 3       10 CCC=1.-CCC
 2       20 FIT(JY+1)=SUM*AH2*YY(JY)**KY
 1          FIT(1)=0.
 1    C**** Y-SPLINE COEFFICIENTS OF FIT.
 1          CALL SPCOEF(ZFIT,FIT,JMYM-1,EZ,HZ,VZ,WZ,ZA,ZB,ZC,ZD,41)
 1    C**** CARRY OUT THE Y-INTEGRATION BY MEANS OF SPLINE. (SEE EQ.
 1    C**** (III-2.8))
 1          DO 40 IQ=1,IQX
 2          SUM=0.
 2          DO 30 JY=1,JMYM
 3          SUM=SUM+ZA(JY)*FI1(JY,IQ,1)+ZB(JY)*FI1(JY,IQ,2)+ZC(JY)
 3         &          *FI1(JY,IQ,3 )+ZD(JY)*FI1(JY,IQ,4)
 3       30 CONTINUE
 2          FF1(IQ,NA)=SUM
 2       40 CONTINUE
 1       50 CONTINUE
            WRITE(6,1000)
       1000 FORMAT('0*** FF1 ***')
            DO 60 IQ=1,IQX
 1       60 WRITE(6,1010) QQ(IQ),(FF1(IQ,NA),NA=1,KOB)
       1010 FORMAT(' ',1P6E15.5)
            RETURN
            END
            SUBROUTINE FORM2(IQX,QQ,CF2,KOB)
      C**** COMPUTE 'FF2(Q,N,M)'(N,M=1 TO 5) OF EQ.(III-2.9A).
            COMMON/AAAAA/XR(60),TR1(60),DT,AH2,JMX,JMXG,JMXGG
            COMMON/BBBBB/YY(47),TY1(47),JMYM,JMYMM,AY2,FY(48,12,4,2)
            COMMON/DDDDD/RHO(61,48,5),FI1(48,40,4),FI2(48,40,4,3),FK(60,40,4,
           &  3)
            COMMON/ZZZZZ/ZFIT(43),EZ(41,3),HZ(42),VZ(42),WZ(42),ZA(42),ZB(42)
           &  ,ZC(42),ZD(42)
            COMMON/UUUUU/UFIT(47),EU(45,3),HU(46),VU(46),WU(46),UA(46),UB(46)
           &  ,UC(46),UD(46)
            COMMON/FFFFF/FF1(40,5),FF2(40,5,5),FF3(40,5,5),FF4(40,5,5)
            COMMON/RAMRAM/COEF(8,8),RAM(111),LA(111),LB(111),LC(111),LD(111),
           &  LG(111)
            DIMENSION QQ(*),CF2(5,5,3),FIT(61),AZ(48,4),SUML(3),G(40,60,3),
           &  LL(5),L(5),IS(5)
            DATA LL/0,0,2,0,2/, L/0,0,0,2,2/, IS/0,1,1,1,1/
      C**** DO-LOOP OVER THE STATE INDICES
            DO 100 NA=1,KOB
 1          DO 90 NB=1,KOB
 2          DO 5 IQ=1,IQX
 3        5 FF2(IQ,NA,NB)=0.
 2    C**** Y-SPLINE OF THE F-FUNCTION OF EQ.(III-2.12)
 2          IP=LL(NA)+2+LL(NB)
 2          JP=L(NA)+2+L(NB)
 2          DO 30 JX=1,JMXG
```

```
3           XIP=XR(JX)**IP
3           DO 10 JY=1,JMYM
4        10 FIT(JY+1)=RHO(JX+1,JY+1,NA)*RHO(JX+1,JY+1,NB)*XIP
3         &                     *YY(JY)**JP
3           FIT(1)=0.
3           CALL SPCOEF(ZFIT,FIT,JMYM-1,EZ,HZ,VZ,WZ,ZA,ZB,ZC,ZD,41)
3           DO 20 JY=1,JMYM
4           AZ(JY,1)=ZA(JY)
4           AZ(JY,2)=ZB(JY)
4           AZ(JY,3)=ZC(JY)
4        20 AZ(JY,4)=ZD(JY)
3 C**** FIND THE G-FUNCTION BY EQ(III-2.15)
3           DO 25 IQ=1,IQX
4           DO 25 NL=1,3
5           G(IQ,JX,NL)=0.
5           DO 25 JY=1,JMYM
6           DO 25 K=1,4
7        25 G(IQ,JX,NL)=G(IQ,JX,NL)+AZ(JY,K)*FI2(JY,IQ,K,NL)
3        30 CONTINUE
2 C**** X-SPLINE OF THE G-FUNCTION (SEE EQ.(III-2.16))
2           DO 80 IQ=1,IQX
3           DO 60 NL=1,3
4           FIT(1)=0.
4           DO 40 JX=1,JMXG
5        40 FIT(JX+1)=G(IQ,JX,NL)
4           CALL SPCOEF(UFIT,FIT,JMXG-1,EU,HU,VU,WU,UA,UB,UC,UD,45)
4 C**** X-SPLINE INTEGRATION BY EQ.(III-2.18)
4           SUMX=0.
4           DO 50 JX=1,JMXG
5        50 SUMX=SUMX+UA(JX)*FK(JX,IQ,1,NL)+UB(JX)*FK(JX,IQ,2,NL)
4         &                 +UC(JX)*FK(JX,IQ,3,NL)+UD(JX)*FK(JX,IQ,4,NL)
4           SUML(NL)=SUMX
4        60 CONTINUE
3 C**** SUMMATION OVER LAMBDA IN EQ.(III-2.9A)
3           SUM=0.
3           IF(IS(NA).NE.IS(NB)) GO TO 80
3           T=MAX(ABS(CF2(NA,NB,1)),ABS(CF2(NA,NB,2)),ABS(CF2(NA,NB,3)))
3           IF(T.LT.1.E-10) GO TO 80
3           DO 70 NL=1,3
4        70 SUM=SUM+SUML(NL)*CF2(NA,NB,NL)
3        80 FF2(IQ,NA,NB)=SUM
2        90 CONTINUE
1       100 CONTINUE
            WRITE(6,1000)
       1000 FORMAT('0*** FF2 ***')
            DO 110 IQ=1,IQX
1       110 WRITE(6,1010) QQ(IQ),((FF2(IQ,NA,NB),NB=1,KOB),NA=1,KOB)
       1010 FORMAT(' ', 1P6E15.5/(' ',15X,1P5E15.5))
            RETURN
            END
            SUBROUTINE NRHS3(IQX,QQ,RSUM,MS)
      C**** COMPUTE THE Y-FUNCTION OF EQ.(III-2.19B) (SEE SUBROUTINE
      C**** 'RHS' FOR DETAILES.)
            COMMON/AAAAA/XR(60),TR1(60),DT,AH2,JMX,JMXG,JMXGG
            COMMON/BBBBB/YY(47),TY1(47),JMYM,JMYMM,AY2,FY(48,12,4,2)
            COMMON/DDDDD/RHO(61,48,5),FI1(48,40,4),FI2(48,40,4,3),FK(60,40,4,
           &  3)
            COMMON/ZZZZZ/ZFIT(43),EZ(41,3),HZ(42),VZ(42),WZ(42),ZA(42),ZB(42)
           &  ,ZC(42),ZD(42)
            COMMON/XXXXX/XFIT(61),EX(59,3),HX(60),VX(60),WX(60),XA(60),XB(60)
           &  ,XC(60),XD(60)
            COMMON/YYYYY/YFIT(48),EY(46,3),HY(47),VY(47),WY(47),YA(47),YB(47)
           &  ,YC(47),YD(47)
```

```
                COMMON/CCCCC/DSP(4,4,60,48)
                COMMON/RAMRAM/COEF(8,8),RAM(111),LA(111),LB(111),LC(111),LD(111),
               &  LG(111)
                COMMON/MMMMM/MAL1(5),KST(14),KED(14),IGX(5),IG(5,5),IPX(5),
               &  LLF(19),LF(19),MSF(19),NR(19)
                DIMENSION SY(4),SX(4),FIT(43),AS(43,7,4),RSUM(40,19),GUM(5),
               &  SKY(7,43),QQ(*),YR(47)
                EQUIVALENCE (YR(1),YY(1))
                KY=42
          C**** DO-LOOP OVER Y
                DO 258 IY=1,JMYM
      1         Y=YR(IY)
      1   C**** PREPARE QUANTITIES IN EQS.(II-3.11),(II-3.19)
      1         YQ=Y*Y
      1         DO 60 K=1,7
      2      60 SKY(K,IY)=0.
      1         XXX=2./3.
      1   C**** DO-LOOP OVER X
      1         DO 255 IX=1,JMXG
      2         X=XR(IX)
      2         XH=0.5*X
      2         XQ=0.25*X*X
      2         XT=1.5*XH
      2         XU=XT*XT
      2         RXQ=XQ+YQ
      2         RYQ=XU+0.25*YQ
      2         RXY=X*Y
      2         XPMIN=ABS(XH-Y)
      2         XPMAX=XH+Y
      2   C**** DECIDE THE COS-THETA MESH SIZE FOR EQ.(III-2.23) (IS THIS
      2   C**** ENOUGH ?)
      2         MCO=INT((XPMAX-XPMIN)/0.2)+1
      2         MCO=2*MCO
      2         IF(MCO.LT.20) MCO=20
      2   C**** COMPUTE THE K-FUNCTION OF EQ.(III-2.23) BY THE SIMPSON
      2   C**** QUADRATURE.
      2         DO 80 N=1,5
      3      80 GUM(N)=0.
      2         DCO=2./MCO
      2         CO=-1.-DCO
      2         CCC=1./3.
      2         DO 220 NCO=1,MCO+1
      3         DDD=CCC
      3         IF(NCO.EQ.1.OR.NCO.EQ.MCO+1) DDD=DDD/2.
      3         CO=CO+DCO
      3   C**** FOR RXYCO,XP,YP, SEE EQ.(II-3.19).
      3         RXYCO=RXY*CO
      3         XP=SQRT(ABS(RXQ-RXYCO))
      3         YP=SQRT(ABS(RYQ+0.75*RXYCO))
      3   C**** PREPARE FOR THE Y-SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
      3         SY(4)=1.
      3         DO 150 JY=1,JMYMM
      4         MY=JY
      4     150 IF(YP.LT.YR(JY)) GO TO 160
      3     160 U=YP-YFIT(MY)
      3         FX=EXP(U)
      3         EXV=1./FX
      3         SY(1)=(FX-EXV)/2.
      3         SY(2)=SY(1)+EXV
      3         SY(3)=U
      3   C**** PREPARE FOR THE X-SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
      3         SX(4)=1.
      3         DO 170 I=1,JMXGG
```

```
4         II=I
4     170 IF(XP.LT.XR(I)) GO TO 180
3     180 U=XP-XFIT(II)
3         FX=EXP(U)
3         EXV=1./FX
3         SX(1)=(FX-EXV)/2.
3         SX(2)=SX(1)+EXV
3         SX(3)=U
3         KGX=IGX(MS)
3 C**** DOUBLE SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
3         FPI=0.
3         DO 190 M=1,4
4         DO 190 N=1,4
5     190 FPI=FPI+DSP(N,M,II,MY)*SX(N)*SY(M)
3 C**** SIMPSON QUADRATURE OF EQ.(III-2.23)
3         DO 200 NG=1,KGX
4         L=IG(NG,MS)+1
4     200 GUM(L)=GUM(L)+DDD*FPI*PL(CO,L-1)
3     220 CCC=1.-CCC
2 C**** COMPLETE THE SIMPSON QUADRATURE OF EQ.(III-2.23) FOR ALL
2 C**** POSSIBLE GAMMA.
2         DO 230 NG=1,KGX
3         LL=IG(NG,MS)+1
3     230 GUM(LL)=DCO*GUM(LL)
2 C**** 'GUM' IS THE K-FUNCTION OF EQ.(III-2.23).
2 C**** COMPUTE THE U-FUNCTION OF EQ.(III-2.22).
2         MPX=IPX(MS)
2         M1=MAL1(MS)
2         DO 250 MP=1,MPX
3         M=MP+M1-1
3         N=NR(M)
3         MA=MSF(M)
3         K1=KST(N)
3         K2=KED(N)
3 C**** SUM OVER A,C,AND GAMMA (I.E., SUM OVER K) IN EQ.(III-2.22).
3         USUM=0.
3         DO 240 K=K1,K2
4         KA=LA(K)+LC(K)
4         KB=LB(K)+LD(K)
4         KG=LG(K)+1
4     240 USUM=USUM+X**KA*Y**KB*GUM(KG)*RAM(K)
3 C**** 'USUM' IS THE U-FUNCTION OF EQ.(III-2.22).
3 C**** NOW, CARRY OUT THE X-INTEGRATION IN EQ.(III-2.21).
3         SKY(MP,IY)=SKY(MP,IY)+X**(LLF(M)+2)*Y**(LF(M)+2)
3        &         *RHO(IX+1,IY+1,MA)*USUM*XXX*TR1(IX)*AH2
3     250 CONTINUE
2     255 XXX=1.-XXX
1     258 CONTINUE
  C**** 'SKY(MP,Y)' IS THE H-FUNCTION OF EQ.(III-2.21).
  C**** OBTAIN THE Y-SPLINE COEFFICIENTS OF 'SKY'(SEE EQ.(III-2.25))
          FIT(1)=0.
          DO 280 MP=1,MPX
1         DO 260 JY=1,JMYM
2     260 FIT(JY+1)=SKY(MP,JY)
1         CALL SPCOEF(ZFIT,FIT,JMYM-1,EZ,HZ,VZ,WZ,ZA,ZB,ZC,ZD,KY-1)
1         DO 270 JY=1,JMYM
2         AS(JY,MP,1)=ZA(JY)
2         AS(JY,MP,2)=ZB(JY)
2         AS(JY,MP,3)=ZC(JY)
2     270 AS(JY,MP,4)=ZD(JY)
1     280 CONTINUE
  C**** CARRY OUT THE Y-INTEGRATION IN EQ.(III-2.19B) BY MEANS OF
  C**** SPLINE INTERPOLATION (SEE EQ.(III-2.26)).
```

```
                DO 320 IQ=1,IQX
    1           DO 310 MP=1,MPX
    2           M=MP+M1-1
    2           RTEMP=0.
    2           DO 300 JY=1,JMYM
    3           DO 300 L=1,4
    4       300 RTEMP=RTEMP+AS(JY,MP,L)*FI1(JY,IQ,L)
    2       310 RSUM(IQ,M)=RTEMP
    1 C**** 'RSUM' IS THE Y-FUNCTION OF EQS.(III-2.19B) OR (III-2.26).
    1       320 CONTINUE
                RETURN
                END
                SUBROUTINE NRHS4(IQX,QQ,RSUM,MS)
      C**** COMPUTE THE INTEGRAL OVER Y OF THE H-FUNCTION IN EQ.(III-2.27A).
      C**** (SEE SUBROUTINE 'RHS' FOR DETAILS.)
                COMMON/AAAAA/XR(60),TR1(60),DT,AH2,JMX,JMXG,JMXGG
                COMMON/BBBBB/YY(47),TY1(47),JMYM,JMYMM,AY2,FY(48,12,4,2)
                COMMON/DDDDD/RHO(61,48,5),FI1(48,40,4),FI2(48,40,4,3),FK(60,40,4,
              &  3)
                COMMON/ZZZZZ/ZFIT(43),EZ(41,3),HZ(42),VZ(42),WZ(42),ZA(42),ZB(42)
              &  ,ZC(42),ZD(42)
                COMMON/XXXXX/XFIT(61),EX(59,3),HX(60),VX(60),WX(60),XA(60),XB(60)
              &  ,XC(60),XD(60)
                COMMON/YYYYY/YFIT(48),EY(46,3),HY(47),VY(47),WY(47),YA(47),YB(47)
              &  ,YC(47),YD(47)
                COMMON/CCCCC/DSP(4,4,60,48)
                COMMON/RAMRAM/COEF(8,8),RAM(111),LA(111),LB(111),LC(111),LD(111),
              &  LG(111)
                COMMON/MMMMM/MAL1(5),KST(14),KED(14),IGX(5),IG(5,5),IPX(5),
              &  LLF(19),LF(19),MSF(19),NR(19)
                DIMENSION SY(4),SX(4),RSUM(40,19),GUM(5,40),SKY(7,43),QQ(*),
              &  SUMX(40),YR(47)
                EQUIVALENCE (YR(1),YY(1))
                KY=42
                YYY=2./3.
      C**** DO-LOOP OVER Y
                DO 258 IY=1,JMYM
    1           Y=YR(IY)
    1           YQ=Y*Y
    1           DO 60 K=1,7
    2           DO 60 IQ=1,40
    3        60 SKY(K,IQ)=0.
    1           XXX=2./3.
    1 C**** DO-LOOP OVER X
    1           DO 255 IX=1,JMXG
    2           X=XR(IX)
    2           XH=0.5*X
    2           XQ=XH*XH
    2           XT=1.5*XH
    2           XU=XT*XT
    2           RXQ=XQ+YQ
    2           RYQ=XU+0.25*YQ
    2           RXY=X*Y
    2           XPMIN=ABS(XH-Y)
    2           XPMAX=XH+Y
    2 C**** FOR 'RMAX' AND 'RMIN' BELOW, SEE EQ.(III-2.31).
    2           RMAX=XH+Y/3.
    2           RMIN=ABS(XH-Y/3.)
    2           RDIF=RMAX-RMIN
    2 C**** COMPUTE THE K-FUNCTION OF EQ.(III-2.30) BY THE SIMPSON
    2 C**** QUADRATURE.
    2           RQ=XQ+YQ/9.
    2           DO 80 IQ=1,IQX
```

```
3        DO 80 N=1,5
4     80 GUM(N,IQ)=0.
2        DO 230 IQ=1,IQX
3        MCO=INT(QQ(IQ)*RDIF)
3        MCO=2*MCO
3        IF(MCO.LT.20) MCO=20
3        DCO=2./MCO
3        CO=-1.-DCO
3        CCC=1./3.
3        DO 220 NCO=1,MCO+1
4        DDD=CCC
4        IF(NCO.EQ.1.OR.NCO.EQ.MCO+1) DDD=DDD/2.
4        CO=CO+DCO
4        RXYCO=RXY*CO
4        RR=SQRT(ABS(RQ-RXYCO/3.))*QQ(IQ)
4        IF(RR.LT.1.E-5) GO TO 90
4        TSIN=SIN(RR)/RR
4        GO TO 100
4     90 TSIN=1.-RR**2/6.
4    100 XP=SQRT(ABS(RXQ-RXYCO))
4        YP=SQRT(ABS(RYQ+0.75*RXYCO))
4 C**** PREPARE FOR THE Y-SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1).
4        SY(4)=1.
4        DO 150 JY=1,JMYMM
5        MY=JY
5    150 IF(YP.LT.YR(JY)) GO TO 160
4    160 U=YP-YFIT(MY)
4        FX=EXP(U)
4        EXV=1./FX
4        SY(1)=(FX-EXV)/2.
4        SY(2)=SY(1)+EXV
4        SY(3)=U
4 C**** PREPARE FOR THE X-SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
4        SX(4)=1.
4        DO 170 I=1,JMXGG
5        II=I
5    170 IF(XP.LT.XR(I)) GO TO 180
4    180 U=XP-XFIT(II)
4        FX=EXP(U)
4        EXV=1./FX
4        SX(1)=(FX-EXV)/2.
4        SX(2)=SX(1)+EXV
4        SX(3)=U
4        KGX=IGX(MS)
4 C**** DOUBLE SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
4        FPI=0.
4        DO 190 M=1,4
5        DO 190 N=1,4
6    190 FPI=FPI+DSP(N,M,II,MY)*SX(N)*SY(M)
4 C**** SIMPSON QUADRATURE OF EQ.(III-2.30)
4        DO 200 NG=1,KGX
5        L=IG(NG,MS)+1
5    200 GUM(L,IQ)=GUM(L,IQ)+DDD*FPI*PL(CO,L-1)*TSIN
4    220 CCC=1.-CCC
3 C**** COMPLETE THE SIMPSON QUADRATURE OF EQ.(III-2.23) FOR ALL
3 C**** POSSIBLE GAMMA.
3        DO 230 NG=1,KGX
4        LL=IG(NG,MS)+1
4    230 GUM(LL,IQ)=DCO*GUM(LL,IQ)
2 C**** 'GUM' IS THE K-FUNCTION OF EQ.(III-2.30).
2 C**** COMPUTE THE U-FUNCTION OF EQ.(III-2.29).
2        MPX=IPX(MS)
2        M1=MAL1(MS)
```

```
 2          DO 250 MP=1,MPX
 3          M=MP+M1-1
 3          N=NR(M)
 3          MA=MSF(M)
 3          K1=KST(N)
 3          K2=KED(N)
 3          DO 242 IQ=1,IQX
 4     242  SUMX(IQ)=0.
 3          DO 240 K=K1,K2
 4          KA=LA(K)+LC(K)
 4          KB=LB(K)+LD(K)
 4          KG=LG(K)+1
 4          T=X**KA*Y**KB
 4          DO 240 IQ=1,IQX
 5     240  SUMX(IQ)=SUMX(IQ)+T*GUM(KG,IQ)*RAM(K)
 3  C****  'SUMX' IS THE U-FUNCTION OF EQ.(III-2.29).
 3  C****  NOW, CARRY OUT THE X-INTEGRATION IN EQ.(III-2.28).
 3          T=X**(LLF(M)+2)*Y**(LF(M)+2)*RHO(IX+1,IY+1,MA)
 3        &          *XXX*TR1(IX)*AH2
 3          DO 250 IQ=1,IQX
 4     250  SKY(MP,IQ)=SKY(MP,IQ)+T*SUMX(IQ)
 2     255  XXX=1.-XXX
 1  C****  'SKY' IS THE H-FUNCTION OF EQ.(III-2.28).
 1  C****  FINALLY, PERFORM THE Y-INTEGRATION IN EQ.(III-2.27A).
 1          DO 310 MP=1,MPX
 2          M=MP+M1-1
 2          DO 300 IQ=1,IQX
 3     300  RSUM(IQ,M)=RSUM(IQ,M)+SKY(MP,IQ)*YYY*TY1(IY)*AY2
 2     310  CONTINUE
 1     258  YYY=1.-YYY
    C****  'RSUM' IS THE Y-INTEGRAL OF H IN EQ.(III-2.27A).
            RETURN
            END
            SUBROUTINE FORM(CF3,RSUM,IQX,FF,KEY)
    C****  THIS SUBROUTINE PERFORMS THE LAST STEPS TO OBTAIN 'FF3'
    C****  AND 'FF4' OF EQS.(III-2.19) AND (III-2.27A). THE STEPS
    C****  TAKEN HERE ARE THE MULTIPLICATION OF N-BAR AND SUMMATION
    C****  OVER GRAND-L AND GRAND-S.
            DIMENSION CF3(8,8),RSUM(40,19),FF(40,5,5),KEY(20),LOCK(20)
            DATA LOCK/11,21,51,12,22,52,33,43,53,34,44,54,15,25,55,55,55,35,
          &  45,55/
    C****  'LOCK(20)' CORRESPOND TO 'KEY(20)', GIVING LOCATIONS
    C****  OF TWENTY NON-ZERO MATRIX ELEMENTS OF TABLE II-3, BUT
    C****  AS A 5*5 MATRIX.
            DO 10 IQ=1,IQX
 1          DO 10 NA=1,5
 2          DO 10 NB=1,5
 3      10  FF(IQ,NA,NB)=0.
            DO 100 IQ=1,IQX
 1          DO 20 MM=1,20
 2          M=MM
 2          IF(MM.GE.17) M=MM-1
 2          NA=LOCK(MM)/10
 2          NB=LOCK(MM)-10*NA
 2          KA=KEY(MM)/10
 2          KB=KEY(MM)-10*KA
 2      20  FF(IQ,NA,NB)=FF(IQ,NA,NB)+CF3(KA,KB)*RSUM(IQ,M)
 1     100  CONTINUE
            RETURN
            END
```

```
              SUBROUTINE XTABLE(JMX,JMXG,XR,TR1,DT,EM,XM,TM,TXC)
      C****    T(X)=C*(X+TO)*X/(X+SO)
              DIMENSION XR(*),TR1(*)
              JMXS=JMXG
              SXO=(TM-TXC*XM)*XM/(EM*XM-TM)
              TXO=EM*SXO/TXC
              DT=TM/JMX
              DTQ=DT**2
              CTO=TXC*TXO
              CSO2=2.*TXC*SXO
              CSO4=2.*CSO2
              TXC2=2.*TXC
              T=0.
              DO 10 I=1,JMXS
  1           T=T+DT
  1           A=CTO-T
  1           XR(I)=(-A+SQRT(A*A+CSO4*T))/TXC2
  1           DTDX=TXC*(XR(I)**2+2.*SXO*XR(I)+TXO*SXO)/(XR(I)+SXO)**2
  1           TR1(I)=1./DTDX
  1        10 CONTINUE
              RETURN
              END
```

§3. Asymptotic Normalization Constants

3-a. General description of Code ASYM5

The asymptotic normalization constants C_ℓ^j can be calculated by using Eqs. (I-6-20) and (I-6-23);

$$C_\ell^j = I_{00}^{\ell j} + I_{20}^{\ell j} + I_{02}^{\ell j} + I_{22}^{\ell j} \tag{3.1}$$

with

$$I_{L''L}^{\ell j} = \sum_{\alpha'} \frac{2}{\sqrt{|\beta|}} \int_0^\infty y^2\, dy\, j_\ell(i|\beta|y) \quad Z_{\alpha\alpha'}^{L''}(y), \tag{3.2}$$

$$Z_{\alpha\alpha'}^{L''}(y) = \int_0^\infty x^2 dx\, w_{L''}(x) \quad ({}^3L''_1|\frac{4M}{3\hbar^2}V|{}^3L_1)\, h_{\alpha\alpha'}(x,y), \tag{3.3}$$

and

$$h_{\alpha\alpha'}(x,y) = \delta_{L_0L_0'}\, \delta_{S_0S_0'} \sum_{L_0S_0} C_{\alpha\alpha'}^{(L_0S_0)}\, U_{\alpha\alpha'}^{L_0}(Q=0;x,y). \tag{3.4}$$

In Eq.(3.1),

$$|\beta| = \sqrt{\frac{4M}{3\hbar^2}\,(|E_t| - |E_d|)} \tag{3.5}$$

In Eq.(3.3), $w_{L''}(x)$ is the deuteron spatial wave function for the ${}^3S_1$-component ($L''=0$) or the ${}^3D_1$ -component ($L''=2$), and $({}^3L''_1|V|{}^3L_1)$ is an element of the 2×2 matrix of the two nucleon potential in the ${}^3S_1+{}^3D_1$ state. The summation over α' in Eq.(3.2) extends over all states α' that can couple to the state α with the same (L_0,S_0). The coefficients $C_{\alpha\alpha'}^{(L_0S_0)}$ are those of Eq.(II-3.2) and are equal to twice $N_{\alpha\alpha'}^{(L_0S_0)}$ of Eq.(I-3-6). The function $U_{\alpha\alpha'}^{L_0}(Q=0;x,y)$ is defined by Eq.(I-3-27). We have dealt with this function many times before. Specifically, the U-function we have here is precisely the one given by Eq. (2.22) in the previous section. Here, we must properly normalize the wave functions obtained by the main program by using the normalization factor computed at the end of Code NFORM. The computation of $U_{\alpha\alpha'}^{L_0}(Q=0;x,y)$ can be done as in subroutine RHS of the main program (see section 6-b of II). However, since we need to consider the ${}^3S_1$ and the ${}^3D_1$ states only for the final states, referring to Tables 1 and 2 of II, the combinations $\{\alpha,L_0,\alpha'\}$ with $\alpha=1$, i.e., with the key-index m=1,4 and 13 must be excluded. Further changes from subroutine RHS are required since we need to consider a greatly extended region of x" and y" as explained below.

The functions $j_\ell(i|\beta|y)$ for $\ell=0$ and $\ell=2$ are explicitly given by

$$j_0(iz) = \frac{e^z - e^{-z}}{2z} \tag{3.6}$$

and

$$j_2(iz) = \left(-\frac{3}{z^2} + \frac{3}{z} - 1\right)\frac{e^z}{2z} + \left(\frac{3}{z^2} + \frac{3}{z} + 1\right)\frac{e^{-z}}{2z} \tag{3.7}$$

with $z=|\beta|y$. The exponential increase of these functions in the intagrand of Eq.(3.2) as $y\to\infty$ must be damped by the counter decrease of $z^{L''}(y)$, and the integrand as a whole will eventually tend to zero as $y\to\infty$. However, its decrease will be extremely slow and we have to extend the upper limit y_M of the y-integration to a very far distance. This necessitates the computation of $\rho_{\alpha'}(x'',y'')$ in Eq.(2.23) up to

$$x'' \leqq x_G = \frac{1}{2} x_{max} + y_M \quad , \tag{3.8a}$$

$$y'' \leqq y_G = \frac{3}{4} x_{max} + \frac{1}{2} y_M \quad . \tag{3.8b}$$

(See Eq.(II-3.11)). Yet we may still need to correct for the cutoff at y_M. We take into account the contribution from beyond $y=y_M$ by assuming that the integrand of Eq.(3.2) decreases exponentially.

The code to perform the above calculations is named ASYM5. It consists of the following parts.

MAIN routien

Subroutines and functions

FYASYM, PTABLE, YTAB, BESSEL, ARHS, SPLINE, SPCOEF, PARAM and PL.

It requires the following preparatory code and data files:

Code DEUTRN

File LFCHI, File LFDEUT and File LFRAM

The preparatory code DEUTRN computes the deuteron wave function and the binding energy. For detailes, see section 3-e. Subroutines PTABLE, YTAB, BESSEL, SPLINE, SPCOEF and PARAM and function PL are identical to those used in Code NFORM (see the previous section). Therefore no listings of them will be given. In the following sections we describe MAIN routine, subroutines FYASYM and ARHS, and Code DEUTRN.

3-b. MAIN routine

The following flow chart should suffice to explain the structure of this routine.

Start

#1 Initialization (Assign values to key-indices by calling subroutine PARAM.
Assign file numbers and x- and y-mesh numbers:
$x_{JMX}=x_{max}$ (same as in the main program), $x_{JMXG}=x_G$,
$y_{JMYM}=y_M$, and $y_{JMYG}=y_G$ of Eq.(3.8).
Also input the binding energy $|E_t|$ and the normalization constant 'FNORM' taken from the output of Code NFORM.)

#2 Read File LFCHI (read in $-\Theta_\alpha(q,x)$ and $\phi_\alpha(q,x)$ that are obtained at the end of the main program)

(Normalize $\phi_\alpha(q,x)$ using the normalization constant 'FNORM')

#3 Read File LFRAM (read in $C_{\alpha\alpha'}^{(L_0S_0)}$, $R_{(L\ell,L'\ell')L_0}^{ac\gamma}$, a,b,c,d, and γ of Eqs.(II-3.2) and (III-2.22).)

#4 Call FYASYM (PTABLE,YTAB,BESSEL)
((i) Set up the same p-table as used in the main program.
(ii) Set up the y-table y(t) and dy/dt as per Eq.(II-8.2) for $0\leq y\leq y_M$,
and (iii) construct $Y_\ell^{(t)}(y,p_k)$ of Eq.(III-2.2) up to y_M.)

#5 Read File LFDEUT
((i) Read in the same x-table x(t) and dx/dt as used in the main program, but extended to x_G ,
(ii) read in the RSC potential for the $^3S_1+^3D_1$ state,
and (iii) read in the wave function and the binding energy of the deuteron.)

#6 (Compute $|\beta|$ and $\sqrt{|\beta|}$ of Eq.(I-6-4).
Compute the (negative energy) regular function $j_\ell(i|\beta|y)$ that enters into Eq.(3.2) as given by Eqs.(3.6) and (3.7).)

#7 (Prepare various spline matrices.)

#8 (Construct $\rho_\alpha(x,y)$ of Eq.(III-2.1).)

```
|-- (Do-loop over the summation index α'(1 to 5))
|
|   #9    (Compute the double-spline coefficients of ρ_α(x,y) of Eq.(II-2.1)
|          for 0≤x≤x_G and 0≤y≤y_G .)
|
|   #10   Call ARHS (SPLINE,SPCOEF,PL)
|                    (Compute the y-integrals in Eq.(3.2) and output them as
|                     'YRHS(L",α)')
|         (Compute I^j_L"L of Eq.(III-3.2) by taking the summation over α' of
|          'YRHS(L",α)' and obtain C^j_ℓ as per Eq.(III-3.1).)
|
|-- (End of the do-loop over α').

          Stop
```

3-c. Subroutine FYASYM

This subroutine sets up the p-table (as used in the main program) and the y-table y(t) and dy/dt using the formula Eq.(II-8.2) for $0\leq y\leq y_M$. Also, this code constructs $Y_\ell^{(t)}(y,p_k)$ of Eq.(III-2.2) up to y_M.

The coding of this subroutine is almost identical to subroutine FYFORM in Code NFORM of the previous section, except for minor changes in dimensionalities of variables relating to y-mesh. For detailes, see Code FYFI5 in the main program.

3-d. Subroutine ARHS (YRHS,NB,BETA)

This subroutine computes the y-integrals (divided by $\sqrt{|\beta|}$) in Eq.(III-3.2) for α'='NB' for all α and L", and outputs them as 'YRHS(L",α)'. The computation of $U^{L0}_{\alpha\alpha'}(Q=0;x,y)$ proceeds as in subroutine RHS of the main program. The following flow chart will be useful to understand the code. We should remember that this subroutine is called within the do-loop over α' in MAIN routine.

```
          Enter

|--- (Do-loop over y)
|
|  |--(Do-loop over x)
|  |
|  |
```

#1 Prepare quantities to compute x" and y" (see Eqs.(3.11) and (3.19) of II.)
Decide the cos θ-mesh size and initiate the cos θ-integration in Eq.(2.32).

#2 Compute $K_{\gamma}^{\alpha'}(x,y)$ of Eq.(2.32) as follows:

(Do-loop over cos θ)

Compute x" and y" as per Eq.(II-3.19)
Construct $\rho_{\alpha'}(x",y")$ by the double-spline over x and y.
Process the Simpson quadrature over cos θ to find $K_{\gamma}^{\alpha'}(x,y)$ of Eq. (2.23).

(End of the do-loop over cos θ)

Multiply the cos θ-mesh size to complete the Simpson quadrature. The resulting $K_{\gamma}^{\alpha'}(x,y)$ is stored as 'GUM(γ)'.

#3 Compute $U_{\alpha\alpha'}^{L_0}(Q=0;x,y)$ of Eq.(2.22). Here we do not need the case with α=1. Thus, we skip the computation for m=1,4 and 13, where m is the key-index in Table 2 of II. For various other indices, see subroutine RHS of the main program.

#4 Multiply $C_{\alpha\alpha'}^{(L_0 S_0)}$ to $U_{\alpha\alpha'}^{L_0}(Q=0;x,y)$ and sum over (L_0,S_0) to find $h_{\alpha\alpha'}(x,y)$ of Eq.(3.4). (For the indices 'LOCK' and "KEY', see the comment cards at the beginning). The results are stored as 'HRHS (α)'.

#5 Process the Simpson quadrature over x in Eq.(3.3). The results are stored as 'ZRHS(L",α)'.

-- (End of the do-loop voer x)

#6 Construct the integrand of the y-integral in Eq.(3.2). The results are 'URHS(L",α)'.

#7 Process the Simpson quadrature over y in Eq.(3.2). The results are devided by $\sqrt{|\beta|}$ and stored as 'YRHS(L",α)'.

----(End of the do-loop over y)

#8 Add the contribution from beyond y_M to the y-integral of Eq.(3.2) by

assuming the integrand goes to zero exponentially.
This bit of contribution is called 'SRHS(L",α)'.

Return

3-e. Code DEUTRN

This code computes the deuteron binding energy and the normalized wave functions for the RSC potential. The method used here is to search for the energy at which the eigenvalue λ of the Sturm-Liouville function for the $^3S_1$ $+^3D_1$ state becomes 1.

The computation of the Sturm-Liouville function is identical to Code STURM of PERFECT V. Since we need to calculate the $^3S_1+^3D_1$ state only, the part relating to the $^1S_0$ state may be deleted. For details see Code STURM of PERFECT V. Unlike in PERFECT V, however, here we must normalize the wave function in an ordinary manner.

The search for the binding energy is performed by Newton-Raphson's method, which proceeds as follows. Writing $q=\sqrt{M|E_d|/\hbar^2}$ and defining

$$f(q) = \lambda(q) - 1 \tag{3.9}$$

we start from a trial value q_1, and compute successive values of q_i (i=1,2,3,...) by

$$q_i = q_{i-1} - f(q_{i-1})/f'(q_{i-1}) \tag{3.10}$$

until the sequence $\{q_1,q_2,...\}$ converges to a value q_0. This value q_0 satisfies $\lambda(q_0)=1$ and yields the binding energy $|E_d|=\frac{\hbar^2}{M}q_0^2$. To utilize this method, we need to calculate $d\lambda(q)/dq$. Instead of computing this quantity numerically, we use the following analytic expression:

$$\frac{d\lambda(q)}{dq} = 2q\lambda(q)^2 \int_0^\infty (u_\lambda(q,x))^2 dx \Big/ \int_0^\infty u_\lambda(q,x)\ \frac{M}{\hbar^2}\ V(x)u_\lambda(q,x)dx \tag{3.11}$$

where $u_\lambda(q,x)$ is the Sturm-Liouville function satisfying

$$\left[\frac{d^2}{dx^2} - \frac{L(L+1)}{x^2} - \frac{1}{\lambda}\frac{M}{\hbar^2}V(x) - q^2\right] u_\lambda(q,x) = 0\ . \tag{3.12}$$

Equation (3.11) can be derived by a simple manipulation of Eq.(3.12). It is fairly general and is applicable even for a velocity dependent potential.

The code consists of the following parts:

MAIN routine

Subroutines and functions
STURMD,POTD,RES1,REXS,COW,BULL and YTABLE

All these subroutines and functions are the same as in Code STURM in Perfect V, except for subroutine YTABLE which utilizes Eq.(II-6.2) to compute the same x-table as used in the main program. (See code STURM for subroutine YTABLE.)

The code requires a data file named LFDEUT.

Perhaps no explanation of MAIN routine is necessary since it is simple and contains enough comment cards in it.

```
 19.812    ** CODE ASYM5 ***                                                     LABEL
(09-30-82)    OPTIONS: INTER,OPT=1,NROUND,NDLR,ALC,NASTER,INLINE=0,LSTIN,LNO,NREST

        C**** *** CODE ASYM5 ***
        C**** TRITON(NEUTRON-DEUTERN) ASYMPTOTIC NORMALIZATION CONSTANTS
        C**** REFERENCES  (SRT=SCIENCE REPORT OF TOHOKU UNIV.)
        C        I: FORMULATIONS  (T.SAWADA+T.SASAKAWA)
        C                          SRT SER.8,VOL.3,NO.1(1982)1
        C       II: PERFECT V     (SASAKAWA+ISHIKAWA+SAWADA)
        C                          SRT (1982)
        C      III: THIS REPORT
              COMMON/AAAAA/XR(80),TR1(80),AH2,RSC(80,3),UU(80),WW(80),JMX,JMXG
              COMMON/BBBBB/YY(60),TY1(60),JMYM,JMYG,AY2,FY(61,12,4,2),SSH(60,2),
             & DSP(4,4,71,46)
              COMMON/RAMRAM/COEF(8,8),RAM(111),LA(111),LB(111),LC(111),LD(111),
             & LG(111)
              COMMON/MMMMM/MAL1(5),KST(14),KED(14),IGX(5),IG(5,5),IPX(5),
             & LLF(19),LF(19),MSF(19),NR(19)
              DIMENSION PFIT(13),EP(11,3),HP(12),VP(12),WP(12),PA(12),PB(12),
             & PC(12),PD(12)
              COMMON/XXXXX/XFIT(71),EX(69,3),HX(70),VX(70),WX(70),XA(70),XB(70),
             & XC(70),XD(70)
              COMMON/YYYYY/YFIT(46),EY(44,3),HY(45),VY(45),WY(45),YA(45),YB(45),
             & YC(45),YD(45)
              DIMENSION TSI(30,5,12),GSI(30,5,12),AQ(12),C(2,5),AK2(12),
             & EXK(70,12,2),PSI(70,12),FIT(71),RHO(71,46,5),
             & AY(71,45,4),YRHS(2,5)
        C-01
        C**** ASSIGN VALUES TO KEY-INDICES.
              CALL PARAM
        C**** ASSIGN FILE NUMBERS, X- AND Y- MESH NUMBERS.
              LFDEUT=28
              LFCHI=27
              LFRAM=21
              JMX=30
              JMXG=70
              JMYM=60
              JMYG=45
        C**** INPUT THE TRITON BINDING ENERGY, AND THE NORMALIZATION CONSTANT.
              BEI=6.60
              FNORM=2.6
              HM=1./41.47
              MALL=12
        C-02
        C**** INPUT THE TRITON WAVE FUNCTIONS.
              REWIND LFCHI
              READ(LFCHI) TSI,GSI
              WRITE(6,1000) BEI,FNORM
         1000 FORMAT('1*** TRITON ASYMPTOTIC NORMALIZATION (ASYM5) ***'/
            & '0BEI=',1PE13.5,'   FNORM=',1PE13.5)
        C**** NORMALIZE THE WAVE FUNCTION
              CONST=SQRT(FNORM)
              DO 20 MS=1,5
    1 *       WRITE(6,1010) MS
    1 *1010 FORMAT('0*** NORMALIZED GSI(MS=',I2,') ***')
    1         DO 20 IM=1,MALL
    2         DO 10 J=1,JMX
    3      10 GSI(J,IM,MS)=GSI(J,IM,MS)/CONST
    2 *       WRITE(6,1020) IM,(GSI(J,IM,MS),J=1,JMX)
    2 *1020 FORMAT(' IM=',I3/(' ',1P10E12.4))
    2      20 CONTINUE
        C-03
```

```
      C**** READ IN THE C-COEFFICIENTS OF EQ.(II-3.2) AND THE R-COEFFICIENTS,
      C**** A,B,C,D AND G OF EQ.(III-2.22).
            REWIND LFRAM
            READ(LFRAM) COEF,RAM,LA,LB,LC,LD,LG
      C-04
      C**** SET UP THE P- AND Y-TABLE, AND CONSTRUCT THE Y-FUNCTION OF EQ.
      C**** (III-2.2).
            CALL FYASYM(AQ,MALL)
            DO 30 I=1,MALL
  1      30 AK2(I)=SQRT(HM*BEI+0.75*AQ(I)**2)
      C-05
      C**** INPUT THE X-TABLE, THE RSC(3S1+3D1) POTENTIAL, AND THE
      C**** DEUTERON BINDING ENERGY AND WAVE FUNCTION.
            REWIND LFDEUT
            READ(LFDEUT) ALPHA,BED,CUTOFF,UU,WW,RSC,DT,XR,TR1
            WRITE(6,1030) ALPHA,BED,CUTOFF
       1030 FORMAT('0DEUTERON WAVE FUNCTION'/' ALPHA=',1PE13.5,'  BED=',
           & 1PE13.5,'  CUTOFF=',1PE13.5)
            AH2=2.*DT
            WRITE(6,1040) (XR(J),J=1,JMX)
       1040 FORMAT('0XR'/(' ',1P10E12.4))
            WRITE(6,1050) (UU(J),J=1,JMX)
       1050 FORMAT('0UU'/(' ',1P10E12.4))
            WRITE(6,1060) (WW(J),J=1,JMX)
       1060 FORMAT('0WW'/(' ',1P10E12.4))
            WRITE(6,1070) (RSC(J,1),J=1,JMX)
       1070 FORMAT('0V11'/(' ',1P10E12.4))
            WRITE(6,1080) (RSC(J,2),J=1,JMX)
       1080 FORMAT('0V12'/(' ',1P10E12.4))
            WRITE(6,1090) (RSC(J,3),J=1,JMX)
       1090 FORMAT('0V22'/(' ',1P10E12.4))
            DO 40 IR=1,3
  1         DO 40 J=1,80
  2      40 RSC(J,IR)=RSC(J,IR)/0.75
      C-06
      C**** FOR BETA, SEE EQ.(I-6-4)
            BETA=SQRT(HM*(BEI-BED)/0.75)
      C**** (NEGATIVE ENERGY) SOLUTIONS (SEE EQS.(III-3.6) AND (III-3.7))
            DO 50 JY=1,JMYM
  1         Z=BETA*YY(JY)
  1         EXZ=EXP(Z)
  1         T1=EXZ/(2.*Z)
  1         T2=1./(EXZ*2.*Z)
  1         SSH(JY,1)=YY(JY)*(T1-T2)
  1      50 SSH(JY,2)=YY(JY)*((-3./Z**2+3./Z-1.)*T1
           &                    +(3./Z**2+3./Z+1.)*T2)
      *     DO 60 K=1,2
      *  60 WRITE(6,1100) K,(SSH(J,K),J=1,JMYM)
      *1100 FORMAT('0SSH(J,K=',I2,')'/(' ',1P10E12.4))
            DO 70 K=1,2
  1         DO 70 NA=1,5
  2      70 C(K,NA)=0.
      C-07
      C**** VARIOUS SPLINE MATRICES
            PFIT(1)=0.
            DO 80 I=1,MALL
  1      80 PFIT(I+1)=AQ(I)
            CALL SPLINE(PFIT,MALL-1,EP,HP,VP,WP,MALL-1,PA,PB,PC,PD)
            XFIT(1)=0.
            DO 90 I=1,JMXG
  1      90 XFIT(I+1)=XR(I)
            CALL SPLINE(XFIT,JMXG-1,EX,HX,VX,WX,69,XA,XB,XC,XD)
            YFIT(1)=0.
            DO 100 I=1,JMYG
```

```
1    100 YFIT(I+1)=YY(I)
         CALL SPLINE(YFIT,JMYG-1,EY,HY,VY,WY,44,YA,YB,YC,YD)
    C-08
    C**** CONSTRUCT THE RHO-FUNCTION OF EQ.(III-2.1).
         X1Q=XR(1)**2
         X2Q=XR(2)**2
         XQ12=X2Q-X1Q
    C**** FIRST, EXTEND THE GSI-FUNCTIONS UP TO XR(JMXG).
         J1=JMX
         DO 120 I=1,MALL
1        DO 110 JX=J1,JMXG
2        AKX=AK2(I)*XR(JX)
2        EXK(JX,I,1)=EXP(-AKX)
2    110 EXK(JX,I,2)=EXK(JX,I,1)*(3./AKX**2+3./AKX+1.)
1    120 CONTINUE
         DO 220 MA=1,5
1        LL=1
1        IF(MA.EQ.3.OR.MA.EQ.5) LL=2
1        DO 150 I=1,MALL
2        ASYM=GSI(JMX,MA,I)/EXK(JMX,I,LL)*XR(JMX)
2        DO 130 J=1,J1
3    130 PSI(J,I)=GSI(J,MA,I)*XR(J)
2        DO 140 J=J1,JMXG
3    140 PSI(J,I)=ASYM*EXK(J,I,LL)
2    150 CONTINUE
1 *      WRITE(6,1110) MA
1 *1110 FORMAT('1*** PSI(MA=',I3,')')
1 *      DO 160 I=1,MALL
1 * 160 WRITE(6,1120) I,(PSI(J,I),J=1,JMXG)
1 *1120 FORMAT('0IM=',I3/(' ',1P10E12.4))
1 C *** P-SPLINE OF 'PSI' DIVIDED BY X**(L+1)
1        LLP=2*(LL/2)+1
1        LP=1
1        IF(MA.GE.4) LP=2
1        DO 190 JX=1,JMXG
2        FIT(1)=0.
2        DO 170 I=1,MALL
3    170 FIT(I+1)=PSI(JX,I)/XR(JX)**LLP
2        CALL SPCOEF(PFIT,FIT,MALL-1,EP,HP,VP,WP,PA,PB,PC,PD,MALL-1)
2        DO 190 JY=1,JMYG+1
3        SUMA=0.
3        DO 180 I=1,MALL
4    180 SUMA=SUMA+PA(I)*FY(JY,I,1,LP)+PB(I)*FY(JY,I,2,LP)+
3      &           PC(I)*FY(JY,I,3,LP)+PD(I)*FY(JY,I,4,LP)
3    190 RHO(JX+1,JY,MA)=SUMA
1 C**** EXTRAPOLATE THE 'RHO'-FUNCTION OF EQ.(III-2.1) TO X=0 ASSUMING
1 C**** THE FORM     A+B*X**2
1        DO 200 JY=1,JMYG+1
2    200 RHO(1,JY,MA)=(RHO(2,JY,MA)*X2Q-RHO(3,JY,MA)*X1Q)/XQ12
1 *      WRITE(6,1130) MA
1 *1130 FORMAT('1*** RHO(MA=',I3,')')
1 *      DO 210 JY=1,JMYG+1
1 * 210 WRITE(6,1140) JY,(RHO(JX,JY,MA),JX=1,JMXG+1)
1 *1140 FORMAT('0JY=',I3/(' ',1P10E12.5))
1    220 CONTINUE
    C**** DO-LOOP OVER THE SUMMATION INDEX ALPHA'.
         DO 290 NB=1,5
1 C-09
1 C**** COMPUTE THE DOUBLE-SPLINE COEFFICIENTS OF 'RHO'.
1        DO 250 JX=1,JMXG+1
2        DO 230 JY=1,JMYG+1
3    230 FIT(JY)=RHO(JX,JY,NB)
2        CALL SPCOEF(YFIT,FIT,JMYG-1,EY,HY,VY,WY,YA,YB,YC,YD,44)
```

```
2          DO 240 JY=1,JMYG
3          AY(JX,JY,1)=YA(JY)
3          AY(JX,JY,2)=YB(JY)
3          AY(JX,JY,3)=YC(JY)
3      240 AY(JX,JY,4)=YD(JY)
2      250 CONTINUE
1          DO 270 L=1,4
2          DO 270 JY=1,JMYG
3          DO 260 JX=1,JMXG+1
4      260 FIT(JX)=AY(JX,JY,L)
3          CALL SPCOEF(XFIT,FIT,JMXG-1,EX,HX,VX,WX,XA,XB,XC,XD,69)
3          DO 270 JX=1,JMXG
4          DSP(1,L,JX,JY)=XA(JX)
4          DSP(2,L,JX,JY)=XB(JX)
4          DSP(3,L,JX,JY)=XC(JX)
4      270 DSP(4,L,JX,JY)=XD(JX)
1 C-10
1 C**** COMPUTE THE Y-INTEGRALS IN EQ.(III-3.2). THE RESULTS ARE 'YRHS'.
1          NBR=NB
1          CALL ARHS(YRHS,NBR,BETA)
1 C**** SUM OVER ALPHA-PRIME(=NB).
1          DO 280 K=1,2
2          DO 280 NA=2,5
3      280 C(K,NA)=C(K,NA)+YRHS(K,NA)
1      290 CONTINUE
  C**** 'C' IS THE I-FUNCTION OF EQ.(III-3.2).
           WRITE(6,1150) ((K,NA,C(K,NA),NA=2,5),K=1,2)
      1150 FORMAT('OTHE ASYMPTOTIC NORMALIZATION CONSTANTS C(K,NA)'/
         & 2(5X,4(3X,'C(',I1,',',I1,')=',1PE13.5)/))
           STOP
           END
           SUBROUTINE ARHS(YRHS,NB,BETA)
  C**** COMPUTE THE Y-INTEGRALS IN EQ.(III-3.2) FOR ALPHA'= NB
  C**** AND FOR ALL ALPHA AND L''.
           COMMON/AAAAA/XR(80),TR1(80),AH2,RSC(80,3),UU(80),WW(80),
         & JMX,JMXG
           COMMON/BBBBB/YY(60),TY1(60),JMYM,JMYG,AY2,FY(61,12,4,2),
         & SSH(60,2),DSP(4,4,71,46)
           COMMON/RAMRAM/COEF(8,8),RAM(111),LA(111),LB(111),LC(111),LD(111),
         & LG(111)
           COMMON/MMMMM/MAL1(5),KST(14),KED(14),IGX(5),IG(5,5),IPX(5),
         & LLF(19),LF(19),MSF(19),NR(19)
           COMMON/XXXXX/XFIT(71),EX(69,3),HX(70),VX(70),WX(70),XA(70),XB(70),
         & XC(70),XD(70)
           COMMON/YYYYY/YFIT(46),EY(44,3),HY(45),VY(45),WY(45),YA(45),YB(45),
         & YC(45),YD(45)
           DIMENSION SY(4),SX(4),YR(61),GUM(5),HRHS(5),SRHS(2,5),URHS(2,5),
         & YRHS(2,5),ZRHS(2,5),LOCK(19),KEY(19),PRURHS(2,5)
           EQUIVALENCE (YR(1),YY(1))
           DATA LOCK/11,21,51,12,22,52,33,43,53,34,44,54,15,25,55,55,35,45,
         &          55/,
         &       KEY/11,21,51,12,22,52,33,43,83,34,44,84,15,25,55,66,38,48,
         &          88/
  C**** 'LOCK(M)' GIVES THE INDICES (ALPHA,ALPHA') OF TABLE 2 OF II,
  C**** 'KEY(M)' THE INDICES (I,I') IN TABLE 3 OF II, BOTH CORRESPONDING
  C**** TO THE KEY-INDIX M.
  C**** FOR KEY(M=16), WE USE 66. THE POSSIBILITY OF KEY(M=16) BEING
  C**** 77 IS TAKEN CASE AT C-04.
           BETAQ=SQRT(BETA)
           MS=NB
           WRITE(6,1000) NB,((K,NA,NA=2,5),K=1,2)
      1000 FORMAT('OTHE INTEGRAND OF THE Y-INTEGRATION IN THE I-FUNCTION ',
         & 'FOR NB=',I2/7X,'Y',6X,2(3X,4(4X,'K=',I1,',NA=',I1,' ')))
           DO 10 K=1,2
```

```
  1         DO 10 NA=1,5
  2      10 YRHS(K,NA)=0.
            YYY=2./3.
    C****   DO-LOOP OVER Y
            DO 220 IY=1,JMYM
  1         IF(IY.EQ.JMYM) YYY=YYY/2.
  1         Y=YR(IY)
  1         YQ=Y*Y
  1         DO 20 K=1,2
  2         DO 20 NA=1,5
  3      20 ZRHS(K,NA)=0.
  1         XXX=2./3.
  1 C****   DO-LOOP OVER X
  1         DO 170 IX=1,JMX
  2         X=XR(IX)
  2 C****   PREPARE QUANTITIES IN EQS.(II-3.11),(II-3.19).
  2 C-01
  2         XH=0.5*X
  2         XQ=XH*XH
  2         XT=1.5*XH
  2         XU=XT*XT
  2         RXQ=XQ+YQ
  2         RYQ=XU+0.25*YQ
  2         RXY=X*Y
  2         XPMIN=ABS(XH-Y)
  2         XPMAX=XH+Y
  2 C****   DECIDE THE COS-THETA MESH SIZE FOR EQ.(III-2.23) (IS THIS
  2 C****   ENOUGH ?)
  2         MCO=INT((XPMAX-XPMIN)/0.2)+1
  2         MCO=2*MCO
  2         IF(MCO.LT.20) MCO=20
  2 C****   COMPUTE THE K-FUNCTION OF EQ.(III-2.23) BY THE SIMPSON
  2 C****   QUADRATURE.
  2         DO 30 N=1,5
  3      30 GUM(N)=0.
  2         DCO=2./MCO
  2         CO=-1.-DCO
  2         CCC=1./3.
  2 C-02
  2         DO 100 NCO=1,MCO+1
  3         DDD=CCC
  3         IF(NCO.EQ.1.OR.NCO.EQ.MCO+1) DDD=DDD/2.
  3         CO=CO+DCO
  3 C****   FOR RXYCO,XP,YP, SEE EQ.(II-3.19).
  3         RXYCO=RXY*CO
  3         XP=SQRT(ABS(RXQ-RXYCO))
  3         YP=SQRT(ABS(RYQ+0.75*RXYCO))
  3 C****   PREPARE FOR THE Y-SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
  3         SY(4)=1.
  3         DO 40 JY=1,JMYG
  4         MY=JY
  4      40 IF(YP.LT.YR(JY)) GO TO 50
  3      50 U=YP-YFIT(MY)
  3         FX=EXP(U)
  3         EXV=1./FX
  3         SY(1)=(FX-EXV)/2.
  3         SY(2)=SY(1)+EXV
  3         SY(3)=U
  3 C****   PREPARE FOR THE X-SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
  3         SX(4)=1.
  3         DO 60 I=1,JMXG
  4         II=I
  4      60 IF(XP.LT.XR(I)) GO TO 70
  3      70 U=XP-XFIT(II)
```

```
3          FX=EXP(U)
3          EXV=1./FX
3          SX(1)=(FX-EXV)/2.
3          SX(2)=SX(1)+EXV
3          SX(3)=U
3          KGX=IGX(MS)
3  C****   DOUBLE SPLINE OF THE RHO-FUNCTION OF EQ.(III-2.1)
3          FPI=0.
3          DO 80 M=1,4
4          DO 80 N=1,4
5       80 FPI=FPI+DSP(N,M,II,MY)*SX(N)*SY(M)
3  C****   SIMPSON QUADRATURE OF EQ.(III-2.23)
3          DO 90 NG=1,KGX
4          L=IG(NG,MS)+1
4       90 GUM(L)=GUM(L)+DDD*FPI*PL(CO,L-1)
3      100 CCC=1.-CCC
2  C****   COMPLETE THE SIMPSON QUADRATURE OF EQ.(III-2.23) FOR ALL
2  C****   POSSIBLE GAMMA.
2          DO 110 NG=1,KGX
3          LL=IG(NG,MS)+1
3      110 GUM(LL)=DCO*GUM(LL)
2  C****   'GUM' IS THE K-FUNCTION OF EQ.(III-2.23).
2  C-03
2          DO 120 NA=1,5
3      120 HRHS(NA)=0.
2  C****   COMPUTE THE U-FUNCTION OF EQ.(III-2.22).
2          MPX=IPX(MS)
2          M1=MAL1(MS)
2          DO 140 MP=1,MPX
3          M=MP+M1-1
3          IF(M.EQ.1.OR.M.EQ.4.OR.M.EQ.13) GO TO 140
3          N=NR(M)
3          MA=MSF(M)
3          K1=KST(N)
3          K2=KED(N)
3  C****   SUM OVER A,C,AND GAMMA (I.E., SUM OVER K) IN EQ.(III-2.22)
3          USUM=0.
3          DO 130 K=K1,K2
4          KA=LA(K)+LC(K)
4          KB=LB(K)+LD(K)
4          KG=LG(K)+1
4      130 USUM=USUM+X**KA*Y**KB*GUM(KG)*RAM(K)
3  C****   'USUM' IS THE U-FUNCTION OF EQ.(III-2.22).
3  C-04
3  C****   COMPUTE THE H-FUNCTION OF EQ.(III-3.4) BY MULTIPLYING
3  C****   'USUM' BY THE C-COEFFICIENTS AND SUM OVER THE GRAND-L AND
3  C****   GRAND-S.
3          NA=LOCK(M)/10
3          KA=KEY(M)/10
3          KB=KEY(M)-10*KA
3          HRHS(NA)=HRHS(NA)+COEF(KA,KB)*USUM
3          IF(M.EQ.16) HRHS(NA)=HRHS(NA)+COEF(KA+1,KB+1)*USUM
3  C****   'HRHS(NA)' IS THE H-FUNCTION OF EQ.(III-3.4).
3      140 CONTINUE
2  C-05
2  C****   PROCESS THE X-INTEGRATION OF EQ.(III-3.3).
2          XTEMP=X*TR1(IX)*XXX
2          DO 150 NA=2,4,2
3          ZRHS(1,NA)=ZRHS(1,NA)+XTEMP*UU(IX)*RSC(IX,1)*HRHS(NA)
3      150 ZRHS(2,NA)=ZRHS(2,NA)+XTEMP*WW(IX)*RSC(IX,2)*HRHS(NA)
2          DO 160 NA=3,5,2
3          ZRHS(1,NA)=ZRHS(1,NA)+XTEMP*UU(IX)*RSC(IX,2)*HRHS(NA)
3      160 ZRHS(2,NA)=ZRHS(2,NA)+XTEMP*WW(IX)*RSC(IX,3)*HRHS(NA)
2      170 XXX=1.-XXX
```

```
1            DO 180 K=1,2
2            DO 180 NA=2,5
3       180  ZRHS(K,NA)=AH2*ZRHS(K,NA)
1  C****     'ZRHS' IS THE Z-FUNCTION OF EQ.(III-3.3).
1  C-06
1  C****     CONSTRUCT THE INTEGRAND OF THE Y-INTEGRATION IN EQ.(III-3.2).
1            DO 200 K=1,2
2            DO 190 NA=2,3
3       190  URHS(K,NA)=Y*SSH(IY,1)*ZRHS(K,NA)
2            DO 200 NA=4,5
3       200  URHS(K,NA)=Y*SSH(IY,2)*ZRHS(K,NA)
1  *         WRITE(6,1010) Y,((URHS(K,NA),NA=2,5),K=1,2)
1  *1010     FORMAT(' ',1PE13.5,2(3X,1P4E13.5))
1  C-07
1  C****     PROCESS THE SIMPSON QUADRATURE OF EQ.(III-3.2).
1            DO 210 K=1,2
2            DO 210 NA=2,5
3            YRHS(K,NA)=YRHS(K,NA)+YYY*TY1(IY)*URHS(K,NA)
3       210  IF(IY.EQ.JMYM-5) PRURHS(K,NA)=URHS(K,NA)
1            YYY=1.-YYY
1       220  CONTINUE
             DO 230 K=1,2
1            DO 230 NA=2,5
2       230  YRHS(K,NA)=AY2*YRHS(K,NA)/BETAQ
   C****     'YRHS' IS THE CONTRIBUTION TO EQ.(III-3.2) FROM Y=0 TO Y(JMYM).
   C-08
   C****     NOW, ADD THE CONTRIBUTION FROM Y BEYOND Y(JMYM) BY ASSUMING
   C****     THE INTEGRAND 'URHS' TO GO TO ZERO EXPONENTIALLY.
             DYM=YY(JMYM)-YY(JMYM-5)
             DO 240 K=1,2
1            DO 240 NA=2,5
2            SRHS(K,NA)=0.
2            IF(ABS(URHS(K,NA)).LT.1.E-15) GO TO 240
2            AL=ALOG(PRURHS(K,NA)/URHS(K,NA))/DYM
2            SRHS(K,NA)=URHS(K,NA)/AL/BETAQ
2       240  YRHS(K,NA)=2.*(YRHS(K,NA)+SRHS(K,NA))
             WRITE(6,1020) MS,((K,NA,NA=2,5),K=1,2)
       1020  FORMAT("0ALPHA' =",I2,2(3X,4(4X,"K=",I1,"NA=",I1," ")))
             WRITE(6,1030) ((YRHS(K,NA),NA=2,5),K=1,2)
       1030  FORMAT(5X,'YRHS=',2(3X,1P4E13.5))
             WRITE(6,1040) ((SRHS(K,NA),NA=2,5),K=1,2)
       1040  FORMAT(' CONTRIBUTION FROM BEYOND Y(JMYM)'/10X,2(3X,1P4E13.5))
             RETURN
             END
             SUBROUTINE FYASYM(AQ,MALL)
   C****     THIS COMPUTE (I) THE P-TABLE, (II) THE Y-TABLE, AND
   C****     (III) THE Y-FUNCTION OF EQ.(III-2.2).(SEE 'CODE FYFI5'
   C****     FOR DETAILES)
             COMMON/BBBBB/YY(60),TY1(60),JMYM,JMYG,AY2,FY(61,12,4,2)
            & ,SSH(60,2),DSP(4,4,71,46)
             DIMENSION AQ(*),AQ1(12),U(61,2),S(4),SUM(61,4,2)
   C-01
   *         WRITE(6,1000)
   *1000     FORMAT('0*** FYASYM ***')
             MALL=12
             IMX=6
             PIMX=0.750428
             CALL PTABLE(AQ,AQ1,DK,PIMX,IMX,MALL)
             JMY=JMYM
             CALL YTAB(YY,TY1,JMY,2.,30.,18.,0.3,DTY)
             AY2=2.*DTY
   C****     DO-LOOP OVER THE GRAND P-MESH.
             P0=0.
```

```
                DO 80 JM=1,MALL
1               JMR=JM
1               P1=AQ(JM)
1    C-02
1    C****      DECIDE THE FINER P'-MESH FOR THE P' INTEGRATION.
1               NP=INT((P1-P0)*YY(JMY)/0.3)
1               NP=2*(NP/2)
1               IF(NP.LT.4) NP=4
1               NP1=NP+1
1               DP=(P1-P0)/NP
1    C-03
1    C****      INITIALIZE THE P' INTEGRATION.
1               IF(JM.EQ.1) GO TO 20
1               DO 10 LL=1,2
2               DO 10 JY=1,JMY+1
3               SUM(JY,1,LL)=0.
3               SUM(JY,2,LL)=U(JY,LL)/6.
3               SUM(JY,3,LL)=0.
3               SUM(JY,4,LL)=SUM(JY,2,LL)
3            10 CONTINUE
1               GO TO 40
1            20 DO 30 JY=1,JMY+1
2               DO 30 LL=1,2
3               DO 30 N=1,4
4               SUM(JY,N,LL)=0.
4            30 CONTINUE
1            40 P=P0
1    C-04
1               CCC=2./3.
1    C****      DO LOOP OVER THE FINER P-MESH(I.E.,P')
1               DO 60 JP=2,NP1
2               DDD=CCC
2               IF(JP.EQ.NP1) DDD=DDD/2.
2               P=P+DP
2    C****      CALCULATE THE SPECTATOR FUNCTION 'U' AT P' AND AT EVERY 'Y'.
2    C****      U(Y)=F(P,Y)/Y**L, WHERE F(P,Y) IS THE NORMALIZED SPHERICAL
2    C****      BESSEL FUNCTION.
2               CALL BESSEL(P,U,YY,JMY)
2    C****      FIND THE SPLINE FUNCTION 'S' AND CARRY OUT THE SIMPSON QUADRATOR
2    C****      FOR THE P' INTEGRATION.
2               UP=P-P0
2               EX=EXP(UP)
2               S(1)=(EX-1./EX)/2.
2               S(2)=S(1)+1./EX
2               S(3)=UP
2               S(4)=1.
2               DO 50 LL=1,2
3               DO 50 JY=1,JMY+1
4               DO 50 N=1,4
5               SUM(JY,N,LL)=SUM(JY,N,LL)+DDD*U(JY,LL)*S(N)
5            50 CONTINUE
2               CCC=1.-CCC
2            60 CONTINUE
1    C-05
1               DO 70 LL=1,2
2               DO 70 JY=1,JMY+1
3               DO 70 N=1,4
4               FY(JY,JM,N,LL)=2.*DP*SUM(JY,N,LL)
4            70 CONTINUE
1               P0=P1
1            80 CONTINUE
       *        DO 110 LL=1,2
       *        DO 110 JM=3,MALL,3
```

```
      *       WRITE(6,1010) JM,JMYM
      *1010 FORMAT('0*** JM=' I3,' ***  JMYM=',I3)
      *       DO 100 M=1,4
      * 100 WRITE(6,1020) M,(FY(JY,JM,M,LL),JY=1,JMY+1)
      *1020 FORMAT('OFY(M=',I2,')'/(' ',1P10E12.4))
      * 110 CONTINUE
            RETURN
            END
            SUBROUTINE XTABLE(JMX,JMXGG,XR,TR1,DT,EM,XM,TM,TXC)
      C****  T(X)=C*(X+TO)*X/(X+SO)
            DIMENSION XR(*),TR1(*)
            SXO=(TM-TXC*XM)*XM/(EM*XM-TM)
            TXO=EM*SXO/TXC
            DT=TM/JMX
            DTQ=DT**2
            CTO=TXC*TXO
            CSO2=2.*TXC*SXO
            CSO4=2.*CSO2
            TXC2=2.*TXC
            T=0.
            DO 10 I=1,JMXGG
  1         T=T+DT
  1         A=CTO-T
  1         XR(I)=(-A+SQRT(A*A+CSO4*T))/TXC2
  1         DTDX=TXC*(XR(I)**2+2.*SXO*XR(I)+TXO*SXO)/(XR(I)+SXO)**2
  1         TR1(I)=1./DTDX
  1      10 CONTINUE
            WRITE(6,1000) TXC,TXO,SXO
       1000 FORMAT('OXTABLE WITH T(X)=C*(X+TO)*X/(X+SO)'/5X,'C=',1PE13.5,
          &  '  TO=',1PE13.5,'  SO=',1PE13.5)
            RETURN
            END
```

```
 20.084    ** CODE DEUTRN ***                                                  LABEL
(09-30-82)   OPTIONS: INTER,OPT=1,NROUND,NDLR,ALC,NASTER,INLINE=0,LSTIN,LNO,NREST

        C**** *** CODE DEUTRN ***
        C**** *** COMPUTE THE DEUTERON WAVE FUNCTION USING THE STURM-FUNCTION
        C**** CODE. SEARCH FOR THE RIGHT VALUE OF Q AT WHICH ALAM=1, BY MEANS
        C**** OF NEWTON-RAPHSON. NORMALIZE THE WAVE FUNCTION IN ORDINARY SENSE.
              COMMON/ZZZZZ/TR(80),XR(80),TR1(80),TR2(80),TR3(80),TR4(80),TR5(80)
             &  ,DT,TMATCH,A,A2,B,CUTOFF,JMX
              COMMON/TTTTT/V11(80),V22(80),V12(80),FL2(80)
              DIMENSION UY(80),WY(80)
              LFDEUT=28
              JMX=30
              JMXS=80
              HM=1./41.47
              CUT=50.
              CALL YTABLE(JMX,JMXS)
              DO 10 J=1,JMX
   1          FL2(J)=6./XR(J)**2
   1          CALL POTD(XR(J),V11(J),V22(J),V12(J),HM,CUT)
   1       10 CONTINUE
        C**** INITIAL GUESS FOR 'Q'.
              Q0=0.230
              Q=Q0
        C**** INITIAL GUESS FOR 'ALAM' AT 'Q' = 'Q0'.
              AM=1.00
              DLAM0=0.05
              WRITE(6,1000)
         1000 FORMAT('0',8X,'Q',18X,'ALAM',16X,'DIFF',7X,'NODE',2X,'I',7X,
             & 'ANO')
              ITIME=0
           20 CALL STURMD(Q,AM,DLAM0,1,UY,WY,ANO)
        C**** NORMALIZATION FACTOR IN ORDINARY SENSE. THE REGION BEYOND XR(JMX)
        C**** IS INTEGRATED ANALYTICALLY.
              SUM=0.
              CCC=2./3.
              DO 30 J=1,JMX
   1          SUM=SUM+(UY(J)**2+WY(J)**2)*CCC*TR1(J)
   1       30 CCC=1.-CCC
              SUM=SUM*2.*DT
              QX=Q*XR(JMX)
              H0=EXP(-QX)
              H2=H0*(3./QX**2+3./QX+1.)
              ASYU=UY(JMX)/H0
              ASYW=WY(JMX)/H2
              HQ=H0**2/(2.*Q)
              HQ2=HQ*(6./QX**3+12./QX**2+6./QX+1.)
              SUM=SUM+ASYU**2*HQ+ASYW**2*HQ2
              IF(DABS(AM-1.0).LT.1.E-5) GO TO 40
              ITIME=ITIME+1
              IF(ITIME.LT.50) GO TO 25
              WRITE(6,1030)
         1030 FORMAT('0NEWTON-RAPHSON DOES NOT CONVERGE.')
              STOP
           25 CONTINUE
        C**** NOW, WE CAN CALCULATE THE DERIVATIVE OF 'ALAM'. (EQ.(III-
        C**** 3.11))
              DAM=2.*Q*AM**2*SUM*ANO
        C**** NEWTON-RAPHSON (SEE EQ.(III-3.10))
              Q=Q-(AM-1.)/DAM
              GO TO 20
        C**** NORMALIZE THE WAVE FUNCTION IN ORDINARY SENSE.
```

```
          40 FNORM=SQRT(SUM)
             DO 50 J=1,JMX
    1           UY(J)=UY(J)/FNORM
    1     50 WY(J)=WY(J)/FNORM
             BED=Q**2/HM
             WRITE(6,1005) Q,BED,AM,FNORM
        1005 FORMAT('0Q=',1PE13.5,'  BED=',1PE13.5,'  AM=',1PE13.5,
           & '  NORM=',1PE13.5)
             WRITE(6,1010) (UY(J),J=1,JMX)
        1010 FORMAT('0UY'/(' ',1P10E12.4))
             WRITE(6,1020) (WY(J),J=1,JMX)
        1020 FORMAT('0WY'/(' ',1P10E12.4))
              REWIND LFDEUT
              WRITE(LFDEUT) Q,BED,CUT,UY,WY,V11,V12,V22,DT,XR,TR1
              ENDFILE LFDEUT
             STOP
             END
             SUBROUTINE POTD(R,A11,A22,A12,HM,CUT)
       C**** COMPUTE  THE 3S1 +3D1 RSC POTENTIAL
             X=0.7*R
             EX1=REX1(1.,CUT,X)
             EX2=REX1(2.,CUT,X)
             EX4=REX1(4.,CUT,X)
             EX6=REX1(6.,CUT,X)
             EXT=REXS(CUT,X)
             EXS=(REXS(CUT/4.,4.*X)-REX1(1.,CUT/4.,4.*X))*64.
             VT=-10.463*(EXT-EXS)
             VC= -10.463*EX1+105.468*EX2-3187.8*EX4+9924.3*EX6
             VT=VT+ 351.77*EX4-1673.5*EX6
             VLS= 708.91*EX4-2713.1*EX6
             A11=HM*VC
             A22=HM*(VC-2.*VT-3.*VLS)
             A12=HM*SQRT(8.)*VT
             RETURN
             END
             FUNCTION REX1(A,C,X)
             P=C*X
             IF(P.LT.50.) GO TO 10
             REX1=EXP(-A*X)/X
             RETURN
          10 REX1=(EXP(-A*X)-EXP(-P))/X
             RETURN
             END
             FUNCTION REXS(C,X)
             P=C*X
             IF(P.LT.50.) GO TO 10
             REXS=(1.+3./X+3./X**2)*EXP(-X)/X
             RETURN
          10 T=(3.*C**2-1.)/2.+3.*C/X+3./X**2
             REXS=((1.+3./X+3./X**2)*EXP(-X)-T*EXP(-P))/X
             RETURN
             END
             SUBROUTINE COW(J,AKQ,DTQ,ALAM)
       C**** COMPUTE
       C****    H**2*A,   AND  1./(1-H**2*A/12)
       C**** IN THE NUMEROV INTEGRATION.
             COMMON/ZZZZZ/TR(80),XR(80),TR1(80),TR2(80),TR3(80),TR4(80),TR5(80)
           &  ,DT,TMATCH,ALPHA,ALPHA2,BETA,CUT,KMX
             COMMON/TTTTT/V11(80),V22(80),V12(80),FL2(80)
             COMMON/CCCCC/A11,A12,A22,B11,B12,B22,C11,C12,C22,UZ1(2),UZ2(2),
           &  WZ1(2),WZ2(2),UZ3(2),WZ3(2)
             A11=(TR2(J)*(V11(J)/ALAM+AKQ)+TR4(J))*DTQ
             A12=TR2(J)*(V12(J)/ALAM)*DTQ
```

```
                A22=(TR2(J)*(V22(J)/ALAM+AKQ+FL2(J))+TR4(J))*DTQ
                B11=1.-A11/12.
                B12=-A12/12.
                B22=1.-A22/12.
                DENOM=B11*B22-B12**2
                C11=B22/DENOM
                C12=-B12/DENOM
                C22=B11/DENOM
                RETURN
                END
                SUBROUTINE BULL(J,AKQ,DXQ,ALAM,UZ,WZ)
          C**** PROCESS   THE NUMEROV ALGORITHM.
                COMMON/TTTTT/V11(80),V22(80),V12(80),FL2(80)
                COMMON/CCCCC/A11,A12,A22,B11,B12,B22,C11,C12,C22,UZ1(2),UZ2(2),
               &  WZ1(2),WZ2(2),UZ3(2),WZ3(2)
                DIMENSION UZ(*),WZ(*)
                D11=2.+A11*C11+A12*C12
                D12=A11*C12+A12*C22
                D22=2.+A12*C12+A22*C22
                DO 11  I=1,2
   1            UZ3(I)=D11*UZ2(I)+D12*WZ2(I)-UZ1(I)
   1            WZ3(I)=D12*UZ2(I)+D22*WZ2(I)-WZ1(I)
   1            UZ1(I)=UZ2(I)
   1            WZ1(I)=WZ2(I)
   1            UZ2(I)=UZ3(I)
   1         11 WZ2(I)=WZ3(I)
                CALL COW(J,AKQ,DXQ,ALAM)
                DO 12  I=1,2
   1            UZ(I )=C11*UZ3(I)+C12*WZ3(I)
   1         12 WZ(I )=C12*UZ3(I)+C22*WZ3(I)
                RETURN
                END
                SUBROUTINE STURMD(AK,ALAM,DLAMO,ND,UY,WY,ANO)
          C**** STURM FUNCTION (NEGATIVE ENERGY) FOR RSC(3S1 + 3D1)
                COMMON/ZZZZZ/TR(80),XR(80),TR1(80),TR2(80),TR3(80),TR4(80),TR5(80)
               &  ,DT,TMATCH,ALPHA,ALPHA2,BETA,CUT,KMX
                COMMON/TTTTT/V11(80),V22(80),V12(80),FL2(80)
                COMMON/CCCCC/A11,A12,A22,B11,B12,B22,C11,C12,C22,UZ1(2),UZ2(2),
               &  WZ1(2),WZ2(2),UZ3(2),WZ3(2)
                DIMENSION UY(*),WY(*),UZ(2, 80),WZ(2, 80),UX(2, 80),WX(2, 80),
               &  AA(4,4),UT(2),WT(2),RR(80)
                EQUIVALENCE (RR(1),XR(1)),(JMX,KMX),(DX,DT)
                XK=AK*RR(JMX)
                HO=EXP(-XK)
                HO=HO/TR3(JMX)
                H2=-(3./XK**2+3./XK+1.)*HO
                H2=-H2
                XK=AK*RR(JMX-1)
                HOM=EXP(-XK)
                HOM=HOM/TR3(JMX-1)
                H2M=-(3./XK**2+3./XK+1.)*HOM
                H2M=-H2M
                MESH=JMX
                DXQ=DX**2
                MESH1=MESH-1
                MESH2=MESH-2
                MATCHO=TMATCH/DT
                MATCH=MATCHO
                IN=1
                NODEO=ND-1
                INODE=20
                AKQ=AK**2
                DLAM=DLAMO
```

```
              JMATCH=MATCH+1
              MATCH1=MATCH+3
              MATCH2=MATCH-3
              M=1
          100 NODE=0
        C**** FIRST, SOLVE FROM INSIDE OUTWARD TO MATCH.
              J=1
              UZ(1,1)=1.E-3
              WZ(1,1)=0.
              UZ(2,1)=0.
              WZ(2,1)=1.E-3
              DO 5 I=1,2
     1        UZ1(I)=0.
     1      5 WZ1(I)=0.
              CALL COW(J,AKQ,DXQ,ALAM)
              DO 10 I=1,2
     1        UZ2(I)=B11*UZ(I,1)+B12*WZ(I,1)
     1     10 WZ2(I)=B12*UZ(I,1)+B22*WZ(I,1)
              DO 50 J=2,MATCH1
     1        K=J
     1        CALL BULL(K,AKQ,DXQ,ALAM,UT,WT)
     1        DO 50 I=1,2
     2        UZ(I,J)=UT(I)
     2        WZ(I,J)=WT(I)
     2     50 CONTINUE
        C**** CHECK THE NUMBER OF NODES INSIDE MATCH.
              DO 60 J=2,MATCH1,2
     1        IF(ABS(UY(J)).LT.1.E-10.OR.ABS(UY(J+2)).LT.1.E-10) GO TO 60
     1        IF(UY(J)/UY(J+2).LT.0.) NODE=NODE+1
     1     60 CONTINUE
              IF(NODE.LE.NODE0) GO TO 70
           95 ALAM=ALAM*(1.+DLAM/5.)
              IN=IN+1
              IF(IN.GT.INODE ) GO TO 220
              GO TO 100
           70 CONTINUE
        C**** NOW, SOLVE FROM OUTSIDE INWARD TO MATCH.
              J=MESH
              UX(1,J)=H0
              WX(1,J)=0.
              UX(2,J)=0.
              WX(2,J)=H2
              UX(1,J-1)=H0M
              WX(1,J-1)=0.
              UX(2,J-1)=0.
              WX(2,J-1)=H2M
              CALL COW(J,AKQ,DXQ,ALAM)
              DO 71 I=1,2
     1        UZ1(I)=B11*UX(I,J)+B12*WX(I,J)
     1     71 WZ1(I)=B12*UX(I,J)+B22*WX(I,J)
              J=J-1
              CALL COW(J,AKQ,DXQ,ALAM)
              DO 72 I=1,2
     1        UZ2(I)=B11*UX(I,J)+B12*WX(I,J)
     1     72 WZ2(I)=B12*UX(I,J)+B22*WX(I,J)
              DO 40 JV=MATCH2,MESH2
     1        J=MESH2-JV+MATCH2
     1        CALL BULL(J,AKQ,DXQ,ALAM,UT,WT)
     1        DO 40 I=1,2
     2        UX(I,J)=UT(I)
     2        WX(I,J)=WT(I)
     2     40 CONTINUE
        C**** JOIN SMOOTHLY THE INNER SOLUTION AND THE OUTER SOLUTION AT MATCH
```

```
              DO 111 J=1,2
   1          K=J
   1          AA(1,K)=UZ(J,MATCH)
   1          AA(2,K)=WZ(J,MATCH)
   1          AA(3,K)=(45.*(UZ(J,MATCH+1)-UZ(J,MATCH-1))-9.*(UZ(J,MATCH+2)
   1         &     -UZ(J,MATCH-2))+UZ(J,MATCH+3)-UZ(J,MATCH-3))/(60.*DX)
   1          AA(4,K)=(45.*(WZ(J,MATCH+1)-WZ(J,MATCH-1))-9.*(WZ(J,MATCH+2)
   1         &     -WZ(J,MATCH-2))+WZ(J,MATCH+3)-WZ(J,MATCH-3))/(60.*DX)
   1      111 CONTINUE
              DO 112 I=1,2
   1          K=5-I
   1          AA(1,K)=UX(I,MATCH)
   1          AA(2,K)=WX(I,MATCH)
   1          AA(3,K)=(45.*(UX(I,MATCH+1)-UX(I,MATCH-1))-9.*(UX(I,MATCH+2)
   1         &     - UX(I,MATCH-2))+UX(I,MATCH+3)-UX(I,MATCH-3))/(60.*DX)
   1          AA(4,K)=(45.*(WX(I,MATCH+1)-WX(I,MATCH-1))-9.*(WX(I,MATCH+2)
   1         &     -WX(I,MATCH-2))+WX(I,MATCH+3)-WX(I,MATCH-3))/(60.*DX)
   1      112 CONTINUE
              DO 113 I=1,4
   1      113 AA(I,4)=-AA(I,4)
              DET=AA(1,1)*(AA(2,2)*AA(3,3)-AA(2,3)*AA(3,2))+AA(2,1)*(AA(3,2)
             &  * AA(1,3)-AA(1,2)*AA(3,3))+AA(3,1)*(AA(1,2)*AA(2,3)-AA(2,2)
             &  * AA(1,3))
              AB =(AA(1,4)*(AA(2,2)*AA(3,3)-AA(2,3)*AA(3,2))+AA(2,4)*(AA(3,2)
             &  * AA(1,3)-AA(1,2)*AA(3,3))+AA(3,4)*(AA(1,2)*AA(2,3)-AA(2,2)
             &  * AA(1,3)))/DET
              BB=(AA(1,1)*(AA(2,4)*AA(3,3)-AA(3,4)*AA(2,3))+AA(2,1)*(AA(3,4)
             &  * AA(1,3)-AA(1,4)*AA(3,3))+AA(3,1)*(AA(1,4)*AA(2,3)-AA(2,4)
             &  * AA(1,3)))/DET
              TB =(AA(1,1)*(AA(2,2)*AA(3,4)-AA(3,2)*AA(2,4))+AA(2,1)*(AA(3,2)
             &  * AA(1,4)-AA(1,2)*AA(3,4))+AA(3,1)*(AA(1,2)*AA(2,4)-AA(2,2)
             &  * AA(1,4)))/DET
        C**** CHECK IF THE LAST EQUATION ALSO HOLDS.
              DIFF=AA(4,1)*AB +AA(4,2)*BB+AA(4,3)*TB -AA(4,4)
        C**** JOIN THE INNER AND THE OUTER SOLUTIONS.
              DO 211 J=1,MATCH
   1          UY(J)=UZ(1,J)*AB+UZ(2,J)*BB
   1      211 WY(J)=WZ(1,J)*AB+WZ(2,J)*BB
              DO 212 J=JMATCH,MESH
   1          UY(J)=-UX(1,J)-UX(2,J)*TB
   1      212 WY(J)=-WX(1,J)-WX(2,J)*TB
        C**** CHECK THE NUMBER OF NODES ALTOGETHER.
              DO 120 J=2,MESH1,2
   1          IF(ABS(UY(J)).LT.1.E-10.OR.ABS(UY(J+2)).LT.1.E-10) GO TO 120
   1          IF(UY(J)/UY(J+2).LT.0.) NODE=NODE+1
   1      120 CONTINUE
              IF(NODE-NODEO) 90,110,95
           90 ALAM=ALAM*(1.-DLAMO)
              IN=IN+1
              IF(IN.GT.INODE) GO TO 220
              GO TO 100
          110 CONTINUE
              IF(ABS(DIFF).LT.1.E-5) GO TO 210
              IF(ABS(DIFF).LT.10.) GO TO 121
              WRITE(6,1011) DIFF
         1011 FORMAT('ODIFF=',1PE13.5)
              STOP
          121 M=M+1
              IF(M.GT.99) GO TO 230
              IF(M.GE.3) GO TO 130
              DIFF1=DIFF
              ALAM1=ALAM
              ALAM=ALAM*(1.-DLAM/3.)
              GO TO 100
```

```
          130 DIFF2=DIFF
              ALAM2=ALAM
              IF(DIFF2/DIFF1.LT.0.) GO TO 150
              IF(ABS(DIFF2).GT.ABS(DIFF1)) GO TO 140
              ALAM=ALAM2*(1.-DLAM/3.)
              ALAM1=ALAM2
              DIFF1=DIFF2
              GO TO 100
          140 IF(M.GE.4) GO TO 150
              DLAM=-DLAM
              GO TO 160
          150 DLAM=DLAM/3.
          160 ALAM=ALAM1*(1.-DLAM/3.)
              GO TO 100
          210 DO 205 J=1,MESH
 1            UY(J)=TR3(J)*UY(J)
 1        205 WY(J)=TR3(J)*WY(J)
        C**** CALCULATE THE NORMALIZATION FACTOR 'ANO' BY EQ.(57).
              SUM=0.
              CCC=2./3.
              DO 213 J=1,MESH
 1            SUM=SUM+(UY(J)*(V11(J)*UY(J)+V12(J)*WY(J))+WY(J)*(V12(J)*UY(J)+
 1           &     V22(J)*WY(J)))*CCC*TR1(J)
 1            CCC=1.-CCC
 1        213 CONTINUE
              SUM=SUM*DX*2.
              ANO=1./SUM
              WRITE(6,1006) AK,ALAM,DIFF,NODE,M, ANO
         1006 FORMAT(' ',1PE15.4,1PD25.13,1PE15.4,2I5,1PE15.5)
              GO TO 200
          220 WRITE(6,1003)
         1003 FORMAT(' NODE EXCEEDS NODEO')
              GO TO 200
          230 WRITE(6,1004)
              GO TO 210
          200 RETURN
         1004 FORMAT('OCONVERGENCE TOO SLOW')
              END
              SUBROUTINE YTABLE(JMX,JMXS)
        C****    T(X) = C*(X+TO)*X/(X+SO)
              COMMON/ZZZZZ/TR(80),XR(80),TR1(80),TR2(80),TR3(80),TR4(80),TR5(80)
             &  ,DT,TMATCH,A,A2,B,CUT,M
              DATA EM,XM,TM,TXC/  12.0,   6.0,   9.0,   0.3/
              SXO=(TM-TXC*XM)*XM/(EM*XM-TM)
              TXO=EM*SXO/TXC
              DT=TM/JMX
              DTQ=DT**2
              CTO=TXC*TXO
              CSO2=2.*TXC*SXO
              CSO4=2.*CSO2
              TXC2=2.*TXC
              T=0.
              DO 10 I=1,JMXS
 1            T=T+DT
 1            A=CTO-T
 1            TR(I)=T
 1            XR(I)=(-A+SQRT(A*A+CSO4*T))/TXC2
 1            DTDX=TXC*(XR(I)**2+2.*SXO*XR(I)+TXO*SXO)/(XR(I)+SXO)**2
 1            DTDX2=TXC*2.*SXO*(SXO-TXO)/(XR(I)+SXO)**3
 1            DTDX3=-(3./(XR(I)+SXO))*DTDX2
 1            TR1(I)=1./DTDX
 1            TR2(I)=TR1(I)**2
 1            TR3(I)=SQRT(TR1(I))
```

```
1          FPP=-DTDX2/(DTDX**3)
1          FPPP=-(DTDX3*DTDX-3.*DTDX2**2)/(DTDX**5)
1          TR4(I)=0.75*(FPP/TR1(I))**2-FPPP/(2.*TR1(I))
1          TR5(I)=TR1(I)*TR3(I)
1       10 CONTINUE
           WRITE(6,1000) TXC,TXO,SXO
      1000 FORMAT('OXTABLE WITH  T(X)=C*(X+TO)*X/(X+SO)'/'OC=',1PE13.5,
         &  '  TO=',1PE13.5,'  SO=',1PE13.5/9X,'TR',13X,'XR',13X,'TR1',12X,
         &  'TR2',12X,'TR3',12X,'TR4',12X,'TR5')
    C       DO 20 J=1,JMXS
    C    20 WRITE(6,1001) TR(J),XR(J),TR1(J),TR2(J),TR3(J),TR4(J),TR5(J)
    C 1001 FORMAT(' ',1P7E15.5)
           DO 30 I=1,JMXS
1       30 IF(XR(I).GE.1.) GO TO 40
        40 TMATCH=TR(I)
           RETURN
           END
```

§4. E1 Sum Rule

4-a. General description of Code SUM5

The enhancement factor κ of the integrated photo-nuclear absorption cross section with the RSC potential can be calculated by Eqs.(I-7-5), (I-7-11), (I-7-18) to (I-7-22). We rearrange these formulas so that the contribution to κ from each component is apparent.

$$\kappa = \sum_{\alpha=1}^{5} \sum_{\alpha' \geqq \alpha}^{5} \kappa_{\alpha\alpha'} \tag{4.1}$$

with

$$\kappa_{\alpha\alpha'} = 3 \frac{M}{\hbar^2} \int_0^\infty dp \int_0^\infty dx\ x^2\ \Theta_\alpha(q,x)\Theta_{\alpha'}(q,x)\ x^2\ G_{\alpha\alpha'}(x) \tag{4.2}$$

where there are only seven non-zero elements $G_{\alpha\alpha'}(x)$ with $\alpha' \geqq \alpha$, and they are

$$G_{11}(x) = \frac{1}{6} \{-V(^1S_0)+V(^1P_1)\} \tag{4.3a}$$

$$G_{22}(x) = G_{44}(x) = \frac{1}{2} \{-V(^3S_1)+\sum_{J=0}^{2} C^2(^3S_1, {}^3P_J)\ V(^3P_J)\} \tag{4.3b}$$

$$G_{33}(x) = G_{55}(x) = \frac{1}{2} \{-V(^3D_1)+\sum_{J=0}^{2} C^2(^3D_1, {}^3P_J)\ V(^3P_J)$$
$$+ C^2(^3D_1, {}^3F_2)\ V(^3F_2)+2\ C(^3D_1, {}^3P_2)C(^3D_1, {}^3F_2)V(^3P_2-{}^3F_2)\} \tag{4.3c}$$

$$G_{23}(x) = G_{45}(x) = -V(^3S_1-{}^3D_1)+\sum_{J=0}^{2} C(^3S_1, {}^3P_J)C(^3D_1, {}^3P_J)V(^3P_J)$$
$$+C(^3S_1, {}^3P_2)C(^3D_1, {}^3F_2)V(^3P_2-{}^3F_2) \tag{4.3d}$$

where for simplicity we have written $V(^{2S+1}\lambda_\xi)$ for $\langle ^{2S+1}\lambda_\xi|V|^{2S+1}\lambda_\xi\rangle$ and $V(^{2S+1}\lambda_\xi - {}^{2S+1}\lambda'_\xi)$ for $\langle ^{2S+1}\lambda_\xi|V|^{2S+1}\lambda'_\xi\rangle$. The coefficients $C(^{2S+1}L_J, {}^{2S+1}\lambda_\xi)$ are given by Eq.(I-7-22):

$$C(^{2S+1}L_J,\ {}^{2S+1}\lambda_\xi) = \hat{L}\hat{\lambda}\hat{\xi} \begin{pmatrix} 1 & L & \lambda \\ 0 & 0 & 0 \end{pmatrix} \begin{Bmatrix} \xi & J & 1 \\ L & \lambda & S \end{Bmatrix} \tag{4.4}$$

There appear ten potential terms in Eq.(4.3) which we number according to Table 1. There are altogether seventeen terms in Eq.(4.3). We denote these seventeen terms by $T_i(x)$ where the numbering i is given by Table 2, so that regularization given here is slightly simpler than that given in PERFECT since here we use one cut-off mass κ whereas in Eqs.(63)-(65) of PERFECT we use stronger cut-off with two regularization masses κ and Λ in the limit of $\Lambda \to \kappa$. Functions REX1(n,κ,x) and REXS(κ,a,x) compute $Y_n(\kappa;x)$ of Eq.(4.11) and $Z(\kappa,a;x)$ of Eq.(4.12), respectively.

Table 1 The numbering of the potential terms

	$^1S_0$	$^3S_1$-$^3S_1$	$^3D_1$-$^3D_1$	$^3S_1$-$^3D_1$	$^1P_1$	$^3P_0$	$^3P_1$	$^3P_2$-$^3P_2$	$^3F_2$-$^3F_2$	$^3P_2$-$^3F_2$
n	1	2	3	4	5	6	7	8	9	10

Table 2 Classification of terms in Eq.(4.3)

The index n corresponds to the numbering of potential by Table 1. For $i \geq 6$, each term is assigned a set of quantum numbers $(L,L',S,J,\lambda,\lambda',\xi)$ according to the form $C(^{2S+1}L_J,{}^{2S+1}\lambda_\xi)C(^{2S+1}L'_J,{}^{2S+1}\lambda'_\xi)\times V(^{2S+1}\lambda_\xi-{}^{2S+1}\lambda'_\xi)$.

i	$T_i(x)$	L L' S J	n	λ λ' ξ
1	$V(^1S_0)$		1	
2	$V(^3S_1)$		2	
3	$V(^3D_1)$		3	
4	$V(^3S_1-{}^3D_1)$		4	
5	$V(^1P_1)$		5	
6	$C^2(^3S_1,{}^3P_0)V(^3P_0)$	0 0 1 1	6	1 1 0
7	$C^2(^3S_1,{}^3P_1)V(^3P_1)$	0 0 1 1	7	1 1 1
8	$C^2(^3S_1,{}^3P_2)V(^3P_2)$	0 0 1 1	8	1 1 2
9	$C^2(^3D_1,{}^3P_0)V(^3P_0)$	2 2 1 1	6	1 1 0
10	$C^2(^3D_1,{}^3P_1)V(^3P_1)$	2 2 1 1	7	1 1 1
11	$C^2(^3D_1,{}^3P_2)V(^3P_2)$	2 2 1 1	8	1 1 2
12	$C^2(^3D_1,{}^3F_2)V(^3F_2)$	2 2 1 1	9	3 3 2
13	$C(^3D_1,{}^3P_2)C(^3D_1,{}^3F_2)V(^3P_2-{}^3F_2)$	2 2 1 1	10	1 3 2
14	$C(^3S_1,{}^3P_0)C(^3D_1,{}^3P_0)V(^3P_0)$	0 2 1 1	6	1 1 0
15	$C(^3S_1,{}^3P_1)C(^3D_1,{}^3P_1)V(^3P_1)$	0 2 1 1	7	1 1 1
16	$C(^3S_1,{}^3P_2)C(^3D_1,{}^3P_2)V(^3P_2)$	0 2 1 1	8	1 1 2
17	$C(^3S_1,{}^3P_2)C(^3D_1,{}^3F_2)V(^3P_2-{}^3F_2)$	0 2 1 1	10	1 3 2

Eq.(4.3) is expressed as

$$G_{11}(x) = \frac{1}{6}(-T_1 + T_5) \tag{4.5a}$$

$$G_{22}(x) = G_{44}(x) = \frac{1}{2}(-T_2 + \sum_{i=6}^{8} T_i) \tag{4.5b}$$

$$G_{33}(x) = G_{55}(x) = \frac{1}{2}(-T_3 + \sum_{i=9}^{12} T_i + 2T_{13}) \tag{4.5c}$$

$$G_{23}(x) = G_{45}(x) = -T_4 + \sum_{i=14}^{17} T_i \tag{4.5d}$$

The code to carry out the above computation is named SUM5. It consists of the following parts:

MAIN routine

Subroutines and functions
PGEN,PEX1,PEXS,XTABLE,PTABLE,CLEBSH, and F6J.

The code reauires a data file named LFCHI which is an output of the main program.

Subroutines PTABLE, CLEBSH and F6J are identical to , and XTABLE is a simplified version of , the ones in the main program. MAIN routine also is simple and self-explanatory with its comment cards. Thus no further explanations are necessary. In the next section, we explain subroutine PGEN and functions REX1 and REXS.

4-b. Subroutine PGEN and functions REX1 and REXS.

These compute ten components of the RSC potential in Table 1. They are given in the notation of reference 3 by the following forms.(In these formulas h=10.463 MeV, and x is in units of μ^{-1}=(1/0.7) fm).

$$V(^1S) = -he^{-x}/x - 1650.6\ e^{-4x}/x + 6484.2\ e^{-7x}/x \tag{4.6a}$$

$$V(^1D) = -he^{-x}/x - 12.322\ e^{-2x}/x - 1112.6\ e^{-4x}/x + 6484.2\ e^{-7x}/x \tag{4.6b}$$

$$V(^3P_0) = -h[(1+4/x+4/x^2)e^{-x} - (16/x+4/x^2)e^{-4x}]/x + 27.133\ e^{-2x}/x - 790.74\ e^{-4x}/x + 20662\ e^{-7x}/x \tag{4.6c}$$

$$V(^3P_1) = h[(1+2/x+2/x^2)\ e^{-x} - (8/x+2/x^2)\ e^{-4x}]/x -135.25\ e^{-2x}/x + 472.81\ e^{-3x}/x \tag{4.6d}$$

$$V(^3P_2-{}^3F_2) = V_C + V_T S_{12} + V_{LS}\ \vec{L}\cdot\vec{S} \tag{4.7}$$

$$V_C = \frac{h}{3}\ e^{-x}/x - 933.48\ e^{-4x}/x + 4152.1\ e^{-6x}/x \tag{4.7a}$$

$$V_T = h[(1/3 + 1/x + 1/x^2)\ e^{-x} - (4/x + 1/x^2)\ e^{-4x}]/x - 34.925\ e^{-3x}/x \tag{4.7b}$$

$$V_{LS} = -2074.1\ e^{-6x}/x \tag{4.7c}$$

and for I=0 potentials

$$V(^1P_1) = 3\ h\ e^{-x}/x - 634.39\ e^{-2x}/x + 2163.4\ e^{-3x}/x \tag{4.8a}$$

$$V(^3D_2) = -3\ h\ [(1+2/x + 2/x^2)\ e^{-x} - (8/x + 2/x^2)\ e^{-4x}]/x -220.12\ e^{-2x}/x + 871\ e^{-3x}/x \tag{4.8b}$$

$$V(^3S_1 - {}^3D_1) = V_C + V_T\, S_{12} + V_{LS}\, \vec{L}\cdot\vec{S} \tag{4.9}$$

$$V_C = -h\, e^{-x}/x + 105.468\, e^{-2x}/x - 3187.8\, e^{-4x}/x + 9924.3\, e^{-6x}/x \tag{4.9a}$$

$$V_T = -h[(1 + 3/x + 3/x^2)\, e^{-x} - (12/x + 3/x^2)\, e^{-4x}]/x$$
$$+351.77\, e^{-4x}/x + 1673.5\, e^{-6x}/x \tag{4.9b}$$

$$V_{LS} = 708.91\, e^{-4x}/x - 2713.1\, e^{-6x}/x \tag{4.9c}$$

The matrix elements of $\vec{L}\cdot\vec{S}$ are

$$\begin{aligned} (\vec{L}\cdot\vec{S}) &= J - 1 && \text{for} \quad L=J-1 \\ &= -1 && \text{for} \quad L=J \\ &= -J - 2 && \text{for} \quad L=J+1 \end{aligned} \tag{4.10}$$

The matrix elements of S_{12} are given by Table 3.

Since these potentials are all singular at the origin, we regularize them in a manner consistent with the regularization adopted in the main program (see subroutine POTD in Code STURM):

L \ L'	J-1	J	J+1
J-1	$-2(J-1)/(2J+1)$	0	$6\sqrt{J(J+1)}/(2J+1)$
J	0	2	0
J+1	$6\sqrt{J(J+1)}/(2J+1)$	0	$-2(J+2)/(2J+1)$

Table 3. Matrix elements of $\langle (LS)J|S_{12}|(L'S)J\rangle$

(i) Terms of the form e^{-nx}/x are replaced by

$$Y_n(\kappa;x) = (e^{-nx} - e^{-\kappa x})/x\ , \tag{4.11}$$

(ii) terms of the form $(1+a/x + a/x^2)\, e^{-x}/x$ are regularized to

$$Z(\kappa,a;x) = (1 + a/x + a/x^2)\, e^{-x}/x$$
$$-[\frac{2-a+\kappa^2 a}{2} + \frac{\kappa a}{x} + \frac{a}{x^2}]\ \frac{e^{-\kappa x}}{x} \tag{4.12}$$

and (iii) terms of the form $(\frac{12}{x} + \frac{3}{x^2})\ \frac{e^{-4x}}{x}$ are converted to

$$W(\kappa,x) = 64\ [Z(\frac{\kappa}{4}\ ,\ 3;4x) - Y_1(\frac{\kappa}{4};4x)] \tag{4.13}$$

In the above expressions, all x are given in units of μ^{-1} (μ=0.7 fm^{-1}). The

```
 10.451    ** CODE SUM5 ***                                                      LABEL
(09-30-82)   OPTIONS: INTER,OPT=1,NROUND,NDLR,ALC,NASTER,INLINE=0,LSTIN,LNO,NREST

      C**** *** CODE SUM5 ***
      C**** COMPUTATION OF THE ENHANCEMENT FACTOR KAPPA OF EQ.(III-4.1)
      C**** FOR THE RSC5 POTENTIAL.
            DIMENSION XR(30),TR1(30),POT(30,10),AQ(12),AQ1(12),TSI(30,5,12),
           & FAC(12),SUMP(7),SUMX(7),VK(7),COEF(17),TPX(17),G(5)
            DIMENSION NSTATE(7),NLA(17),NLB(17),NSS(17),NJJ(17),
           & NL1(17),NL2(17),NPT(17),NJ2(17),GSI(30,5,12),FACLOG(100)
      C**** THE FOLLOWING INDICES REFER TO TABLES 1 AND 2 EXCEPT FOR 'NSTATE'.
      C**** 'NLA'=L, 'NLB'=L', 'NSS'=S, 'NJJ'=J, 'NL1'=LAMBDA, 'NL2'=LAMBDA',
      C**** 'NJ2'=XI, AND 'NPT'=N.
      C**** 'NSTATE' IS THE KEY TO STATE INDICES OF SEVEN G-FUNCTIONS OF
      C**** EQ.(III-4.3).
            DATA NSTATE/11,22,44,33,55,23,45/,
           &        NLA/0,0,2,0,0,0,0,0,2,2,2,2,2,0,0,0,0/,
           &        NLB/0,0,2,2,0,0,0,0,2,2,2,2,2,2,2,2,2/,
           &        NSS/0,1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1/,
           &        NJJ/0,1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1/,
           &        NL1/0,0,2,0,1,1,1,1,1,1,1,3,1,1,1,1,1/,
           &        NL2/0,0,2,2,1,1,1,1,1,1,1,3,3,1,1,1,3/,
           &        NPT/1,2,3,4,5,6,7,8,6,7,8,9,10,6,7,8,10/,
           &        NJ2/0,1,1,1,1,0,1,2,0,1,2,2,2,0,1,2,2/
            DATA FNORM/2.6/
      C**** 'FNORM' IS THE NORMALIZATION FACTOR COMPUTED BY CODE NFORM.
            WRITE(6,1010)
       1010 FORMAT('0*** CODE SUM5 ***')
      C-01
            LFCHI=27
            REWIND LFCHI
            READ(LFCHI) TSI,GSI
            JMX=30
            CALL XTABLE(30,JMX,XR,TR1,DT,12.,6.,9.,0.3)
            AH2=2.*DT
            WRITE(6,2000) (XR(J),J=1,JMX)
       2000 FORMAT('0 XR=',1P10E12.5/(5X,1P10E12.5))
            IMX=6
            MALL=12
            PIMX=0.750428
            CALL PTABLE(AQ,AQ1,DK,PIMX,IMX,MALL)
            AKH2=2.*PIMX/IMX
            CCC=2./3.
            DO 10 I=1,IMX
 1          FAC(I)=CCC*AKH2*AQ1(I)
 1          IF(I.EQ.IMX) FAC(I)=FAC(I)/2.
 1       10 CCC=1.-CCC
            AKH2=2.*DK
            FAC(IMX)=FAC(IMX)+AKH2*AQ1(IMX)/6.
            IMX1=IMX+1
            CCC=2./3.
            DO 20 I=IMX1,MALL
 1          FAC(I)=CCC*AKH2*AQ1(I)
 1          IF(I.EQ.MALL) FAC(I)=FAC(I)/2.
 1       20 CCC=1.-CCC
            DO 40 MS=1,5
            WRITE(6,1040) MS
       1040 FORMAT('0*** TSI(MS=',I3,') ***')
            DO 30 IM=1,MALL
         30 WRITE(6,1060) IM,(TSI(J,MS,IM),J=1,JMX)
       1060 FORMAT(' IM=',I3/(' ',1P10E12.5))
         40 CONTINUE
```

```
                WRITE(6,1050) FNORM
           1050 FORMAT('0FNORM=',1PE13.5)
          C-02
          C**** COMPUTE TEN POTENTIAL COMPONENTS OF TABLE III-1.
                CALL PGEN(XR,JMX,POT)
                WRITE(6,1030)
           1030 FORMAT('0*** POT ***')
                DO 50 JX=1,JMX
             50 WRITE(6,1045) JX,(POT(JX,K),K=1,10)
           1045 FORMAT(I4,1P10E12.4)
                HM=1./41.47
                DO 60 K=1,10
      1         DO 60 JX=1,JMX
      2      60 POT(JX,K)=HM*POT(JX,K)
          C-03
          C**** 'COEF' ARE THE COEFFICIENTS C*C IN TABLE III-2.
                DO 70 I=1,5
      1      70 COEF(I)=1.
                FACLOG(1)=0.
                FACLOG(2)=0.
                F1=1.
                DO 80 I=3,100
      1         F1=F1+1.
      1         FACLOG(I)=FACLOG(I-1)+ALOG(F1)
      1      80 CONTINUE
                DO 90 K=6,17
      1         LA=2*NLA(K)
      1         LB=2*NLB(K)
      1         L1A=2*NL1(K)
      1         L1B=2*NL2(K)
      1         JJ=2*NJJ(K)
      1         JS=2*NSS(K)
      1         J2=2*NJ2(K)
      1         FACTOR=(J2+1.)*SQRT((L1A+1.)*(L1B+1.))
      1         CALL CLEBSH(L1A,2,LA,0,0,0,FACLOG,CLEBA)
      1         CALL CLEBSH(L1B,2,LB,0,0,0,FACLOG,CLEBB)
      1         CALL F6J(FACLOG,L1A,J2,LA,JJ,JS,2,RACA)
      1         CALL F6J(FACLOG,L1B,J2,LB,JJ,JS,2,RACB)
      1      90 COEF(K)=FACTOR*CLEBA*CLEBB*RACA*RACB
                WRITE(6,1065) (COEF(K),K=6,17)
           1065 FORMAT('0COEF=',1P10E12.4/(6X,1P10E12.4))
          C-04
          C**** CARRY OUT THE P- AND X-INTEGRATIONS OF EQ.(III-4.2).
                DO 100 K=1,7
      1     100 SUMP(K)=0.
                DO 160 IM=1,MALL
      1         DO 110 K=1,7
      2     110 SUMX(K)=0.
      1         CCC=2./3.
      1         DO 140 JX=1,JMX
      2         XQ=XR(JX)**4*TR1(JX)*AH2*CCC
      2   C**** COMPUTE THE SEVENTEEN T-FUNCTIONS IN TABLE III-2 AND EQ.(III-4.5).
      2         DO 120 I=1,17
      3         KPT=NPT(I)
      3     120 TPX(I)=COEF(I)*POT(JX,KPT)
      2   C**** CONSTRUCT THE G-FUNCTIONS OF EQ.(III-4.5).
      2         G(1)=(-TPX(1)+TPX(5))/6.
      2         G(2)=(-TPX(2)+TPX(6)+TPX(7)+TPX(8))/2.
      2         G(3)=G(2)
      2         G(4)=(-TPX(3)+TPX(9)+TPX(10)+TPX(11)+TPX(12)+2.*TPX(13))/2.
      2         G(5)=G(4)
      2         G(6)=-TPX(4)+TPX(14)+TPX(15)+TPX(16)+TPX(17)
      2         G(7)=G(6)
      2   C**** NOW, CARRY OUT THE SIMPSON QUADRATURE OVER X OF EQ.(III-4.2).
```

```
2          DO 130 K=1,7
3          NA=NSTATE(K)/10
3          NB=NSTATE(K)-10*NA
3      130 SUMX(K)=SUMX(K)+TSI(JX,NA,IM)*G(K)*TSI(JX,NB,IM)*XQ
2      140 CCC=1.-CCC
1          DO 150 K=1,7
2      150 SUMP(K)=SUMP(K)+SUMX(K)*FAC(IM)
1      160 CONTINUE
     C**** 'SUMP' IS THE RESULTS OF THE P- AND X-INTEGRATION OF EQ.(III-4.2).
     C**** NOW, MULTIPLY 3 AND DIVID BY THE NORMALIZATION FACTOR TO
     C**** FIND THE SEVEN KAPPAS OF EQ.(III-4.2).
           DO 170 K=1,7
1      170 VK(K)=3.*SUMP(K)/FNORM
     C**** SUM OVER ALL KAPPAS AS IN EQ.(III-4.1).
           FKAPPA=0.
           DO 180 K=1,7
1      180 FKAPPA=FKAPPA+VK(K)
           WRITE(6,1020) FKAPPA,(NSTATE(K),VK(K),K=1,7)
      1020 FORMAT('OKAPPA=',1PE13.5/' CONTRIBUTION TO KAPPA FROM COMPONENT'/
          & (5X,' K(',I2,')=',1PE13.5))
           STOP
           END
           SUBROUTINE PGEN(XR,JMX,POT)
           DIMENSION XR(*),POT(30,10)
           CUT=50
           H=10.463
           DO 10 JX=1,JMX
1          X=0.7*XR(JX)
1          EX1=REX1(1.,CUT,X)
1          EX2=REX1(2.,CUT,X)
1          EX3=REX1(3.,CUT,X)
1          EX4=REX1(4.,CUT,X)
1          EX6=REX1(6.,CUT,X)
1          EX7=REX1(7.,CUT,X)
1          TX2=REXS(CUT,2.,X)
1          TX3=REXS(CUT,3.,X)
1          TX4=REXS(CUT,4.,X)
1          WX=64.*(REXS(CUT/4.,3.,4.*X)-REX1(1.,CUT/4.,4.*X))
1          POT(JX,1)=-H*EX1-1650.6*EX4+6484.2*EX7
1          VC=-H*EX1+105.468*EX2-3187.8*EX4+9924.3*EX6
1          VT=-H*(TX3-WX)+351.77*EX4-1673.5*EX6
1          VLS=708.91*EX4-2713.1*EX6
1          POT(JX,2)=VC
1          POT(JX,3)=VC-2.*VT-3.*VLS
1          POT(JX,4)=2.828427125*VT
1          POT(JX,5)=3.*H*EX1-634.39*EX2+2163.4*EX3
1          POT(JX,6)=-H*(TX4-WX/0.75)+27.133*EX2-790.74*EX4+20662.*EX7
1          POT(JX,7)=H*(TX2-WX/1.5)-135.25*EX2+472.81*EX3
1          VC=H*EX1/3.-933.48*EX4+4152.1*EX6
1          VT=H*(TX3-WX)/3.-34.925*EX3
1          VLS=-2074.1*EX6
1          POT(JX,8)=VC-VT/2.5+VLS
1          POT(JX,9)=VC-1.6*VT-4.*VLS
1          POT(JX,10)=2.939387692*VT
1       10 CONTINUE
           RETURN
           END
           FUNCTION REX1(A,C,X)
           P=C*X
           IF(P.LT.50.) GO TO 10
           REX1=EXP(-A*X)/X
           RETURN
        10 REX1=(EXP(-A*X)-EXP(-P))/X
```

```
              RETURN
              END
              FUNCTION REXS(C,F,X)
              P=C*X
              IF(P.LT.50.) GO TO 10
              REXS=(1.+F/X+F/X**2)*EXP(-X)/X
              RETURN
           10 T=(2.-F+F*C**2)/2.+F*C/X+F/X**2
              REXS=((1.+F/X+F/X**2)*EXP(-X)-T*EXP(-P))/X
              RETURN
              END
              SUBROUTINE XTABLE(JMX,JMXG,XR,TR1,DT,EM,XM,TM,TXC)
        C****  T(X)=C*(X+TO)*X/(X+SO)
              DIMENSION XR(*),TR1(*)
              JMXS=JMXG
              SXO=(TM-TXC*XM)*XM/(EM*XM-TM)
              TXO=EM*SXO/TXC
              DT=TM/JMX
              DTQ=DT**2
              CTO=TXC*TXO
              CSO2=2.*TXC*SXO
              CSO4=2.*CSO2
              TXC2=2.*TXC
              T=0.
              DO 10 I=1,JMXS
   1          T=T+DT
   1          A=CTO-T
   1          XR(I)=(-A+SQRT(A*A+CSO4*T))/TXC2
   1          DTDX=TXC*(XR(I)**2+2.*SXO*XR(I)+TXO*SXO)/(XR(I)+SXO)**2
   1          TR1(I)=1./DTDX
   1       10 CONTINUE
              RETURN
              END
```

§5. Two-body Correlation Function

For the computation of the two-body correlation function

$$\rho(x) = \int d\hat{x}\, d^3y\, |\Psi|^2 \tag{5.1}$$

we express the total wave function

$$\Psi = \Phi(12,3) + \Phi(23,1) + \Phi(31,2) \tag{5.2}$$

in terms of a single partition, for example,

$$\Psi = - \underset{\underset{\sim}{\alpha}}{S}\, |F_{\underset{\sim}{\alpha}}(12,3))\ \Theta_{\underset{\sim}{\alpha}}(x) \tag{5.3}$$

(See Eq.(I-2-40)). Then due to the ortho-normality of $|F_{\underset{\sim}{\alpha}}(12,3))$, tne function $\rho(x)$ is simply given by

$$\rho(x) = \sum_{\alpha} \int_0^{\infty} dp\ [\Theta_{\alpha}(q,x)]^2 \tag{5.4}$$

where

$$\Theta_{\alpha}(q,x) = \phi_{\alpha}(q,x) + \chi_{\alpha}(q,x) \tag{5.5}$$

The wave function $\chi_{\alpha}(q,x)$ is the radial part of the component $\Phi(23,1)+\Phi(31,2)$, and $\phi_{\alpha}(q,x)$ is that of the component $\Phi(12,3)$. (See Eqs.(I-2-39a) and (I-2-38a)). In the main program, we compute and output $-\Theta_{\alpha}(q,x)$ and $\phi_{\alpha}(q,x)$ for $0 \leqq x \leqq x_{max}$ where x_{max} is the force range. To compute $\rho(x)$ for $x \leqq x_{max}$, therefore, is straightforward. We should not forget to normalize the wave function. The normalization factor is output from Code NFORM (see section 2-d).

However, as mentioned in section 8 of I, $\chi_{\alpha}(q,x)$ is considerably long ranged compared to $\phi_{\alpha}(q,x)$. When it is necessarty to compute $\rho(x)$ for $x > x_{max}$, say for $x \leqq x_M$ where x_M is much larger than x_{max} , we have to extend the computation of $\chi_{\alpha}(q,x)$ to $x_{max} < x \leqq x_M$ as explained in section 3 of II.

5-a. General description of Code CORR5

The code to carry out the computation of $\rho(x)$ of Eq.(5.1) is named CORR5. It consists of the parts

MAIN routine, and

Subroutines XTABLE and PTABLE

It requires the data file LFCHI.

In presenting the listing of the code, we assume that we already have the data file LFCHI on which $-\Theta_\alpha(q,x)$ are stored, so that we can start from there.

If we are satisfied with calculating $\rho(x)$ for $x \leqq x_{max}$, then we can use the data file LFCHI output from the main program. If we wish to calculate $\rho(x)$ for $x \leqq x_M$ $(x_M > x_{max})$, then, before entering the computation of $\rho(x)$ using Code CORR5, we must extend $\chi_\alpha(q,x)$ to $x_{max} < x \leqq x_M$ and re-store $-\Theta_\alpha(q,x)$ on File LFCHI for $x \leqq x_M$. The dimensionality of quanitities relating to x and y must be modified accordingly. This preparatory part of the calculation to extend $\chi_\alpha(q,x)$ to $x_{max} < x \leqq x_M$ is very tedious. Since it has to be done only once, perhaps the simplest way is to follow the procedure explained in section 3 of II in exactly the same way, except now we use x_M in place of x_{max}. We shall not present the listing of this preparatory code but merely indicate how to construct it. Thus, first we note that Y(x) corresponding to Eq.(II-3.12) is to be chosen as

$$Y(x) \leqq \frac{1}{2} x + x_M \leqq \frac{3}{2} x_M \qquad (\text{for } x \leqq x_M) \tag{5.6}$$

By Code OVER5 with appropriately modified dimensionalities, we prepare $Q^{L0}_{\alpha\alpha'}(p,p';x)$ of Eq.(II-3.17) for $0 \leqq x \leqq x_M$. This allows us to find ${}^2H^{L0}_{\alpha\alpha'}(x)$ of Eq.(II-3.16). From Eq.(II-3.11), we have the upper limits of x" and y" given by

$$x" \leqq \frac{1}{2} x + y \leqq 2\, x_M \quad , \qquad y" \leqq \frac{3}{4} x + \frac{1}{2} y \leqq \frac{3}{2} x_M \tag{5.7}$$

Therefore, by Code FYFI5 with appropriately modified dimensionalities, we prepare $Y^{(t)}_\ell(y,p_k)$ and $I^{(s)}_\ell(p,y_i)$ of Eqs.(II-5.9) and (II-5.6) for $0 \leqq y \leqq \frac{3}{2} x_M$. Next, we read off from File LFCHI the functions $\phi_\alpha(q,x)$ for $x \leqq x_{max}$ (which are obtained in the main program). Then we extend $\phi_\alpha(q,x)$ to the region $x_{max} < x \leqq 2\, x_M$ using the asymptotic form Eq.(II-3.10).
Now, we are ready to construct $\rho_\alpha(x",y")$ in Eq.(II-3.8) for $0 \leqq x" \leqq 2X_M$ and $0 \leqq y" \leqq \frac{3}{2} x_M$ with the help of the p-spline using $Y^{(t)}_\ell(y,p_k)$ as in Eqs.(II-5.7) and (II-5.8). The computation of $K^{\alpha'}_\gamma(x,y)$ of Eq.(II-3.8) and $U^{L0}_{\alpha\alpha'}(Q=0;x,y)$ of Eq.(II-3.7) and the subsequent calculation of ${}^1H^{L0}_{\alpha\alpha'}(x)$ of Eq.(II-3.14) by means of the y-spline using $I^{(s)}_\ell(p,y_i)$ proceed as in subroutine RHS of the main program. Thus we obtain $H^{L0}_{\alpha\alpha'}(x) = {}^1H^{L0}_{\alpha\alpha'} + {}^2H^{L0}_{\alpha\alpha'}$, which yields the desired $\chi_\alpha(x)$ as per Eq.(II-3.1). All these calculations are entirely parallel to the steps No.25 to No.28 in the main program (see section 6-a of II), and require no further explanations. After calculating $\chi_\alpha(q,x)$ we can construct

$$\Theta_\alpha(q,x) = \phi_\alpha(q,x) + \chi_\alpha(q,x) \tag{5.8}$$

for $0 \leqq x \leqq x_M$ which is to be stored on File LFCHI to be used in MAIN routine.

```
 19.399     ** CODE CORR5 ****                                          LABEL
(09-30-82)    OPTIONS: INTER,OPT=1,NROUND,NDLR,ALC,NASTER,INLINE=0,LSTIN,LNO,NREST

          C**** *** CODE CORR5 ****
          C**** COMPUTATION OF THE TWO-BODY CORRELATION FUNCTION OF THE TRITON
          C**** WITH RSC5.
                DIMENSION XR(30),TR1(30),TSI(30,5,12),GSI(30,5,12),
               & CORR(30,5),CORSUM(30),AQ(12),AQ1(12),FAC(12)
          C**** INPUT THE NORMALIZATION CONSTANT OBTAINED IN CODE FNORM.
                DATA FNORM/2.6/
                WRITE(6,1000)
           1000 FORMAT('1*** TRITON TWO-BODY CORRELATION FUNCTION ***')
          C-01
          C**** SET UP THE X-TABLE. THIS MUST BE THE SAME AS IN THE MAIN
          C**** PROGRAM, OR THE EXTENSION OF IT.
                JMXG=30
                CALL XTABLE(XR,TR1,JMXG,DT)
          C**** SET UP THE SAME P-TABLE AS USED IN THE MAIN PROGRAM, AND
          C**** CONSTRUCT THE SIMPSON-WEIGHTS 'FAC' FOR THE P-INTEGRATION.
                PIMX=0.750428
                IMX=6
                MALL=12
                CALL PTABLE(AQ,AQ1,DP,PIMX,IMX,MALL)
                C=2./3.
                DK=2.*PIMX/IMX
                DO 10 I=1,IMX
      1         FAC(I)=CCC*DK
      1         IF(I.EQ.IMX) FAC(I)=FAC(I)/2.
      1      10 CCC=1.-CCC
                DK=0.4
                FAC(IMX)=FAC(IMX)+DK/6.
                IMX1=IMX+1
                CCC=2./3.
                DO 20 I=IMX1,MALL
      1         FAC(I)=CCC*DK
      1         IF(I.EQ.MALL) FAC(I)=FAC(I)/2.
      1      20 CCC=1.-CCC
          C-02
          C**** INPUT THE TOTAL WAVE FUNCTION 'TSI'.
                LFCHI=27
                REWIND LFCHI
                READ(LFCHI) TSI,GSI
                DO 80 JS=1,5
      1   C     WRITE(6,1010) JS
      1   C1010 FORMAT('OTSI(JS=',I2,')')
      1   C     DO 30 IM=1,MALL
      1   C  30 WRITE(6,1020) IM,(TSI(JX,JS,IM),JX=1,JMXG)
      1   C1020 FORMAT('OIM=',I2,/(5X,1P10E12.4))
      1   C-03
      1   C**** CORRELATION FUNCTION OF EACH STATE.
      1         DO 50 JX=1,JMXG
      2         SUM=0.
      2         DO 40 I=1,MALL
      3      40 SUM=SUM+FAC(I)*(TSI(JX,JS,I)*XR(JX))**2
      2      50 CORR(JX,JS)=SUM
      1         WRITE(6,1030) JS,(CORR(JX,JS),JX=1,JMXG)
      1    1030 FORMAT('OCORR(JS=',I2,')'/(5X,1P10E12.4))
      1   C**** NORMALIZE STATE-BY-STATE.
      1         CCC=2./3.
      1         CNORM=0.
      1         DO 60 JX=1,JMXG
      2         CNORM=CNORM+CCC*TR1(JX)*CORR(JX,JS)
```

```
2      60 CCC=1.-CCC
1         CNORM=2.*DT*CNORM
1         DO 70 JX=1,JMXG
2      70 CORSUM(JX)=CORR(JX,JS)/CNORM
1         WRITE(6,1040) CNORM,(CORSUM(JX),JX=1,JMXG)
1    1040 FORMAT('0AFTER NORMALIZATION WITH CNORM=',1PE13.5/
1        &             (5X,1P10E12.4))
1      80 CONTINUE
     C-04
     C**** TOTAL CORRELATION FUNCTION
          DO 100 JX=1,JMXG
1         SUM=0.
1         DO 90 JS=1,5
2      90 SUM=SUM+CORR(JX,JS)
1     100 CORSUM(JX)=SUM
     C**** COMPUTE THE (TOTAL) NORMALIZATION COMSTANT 'CNORM'(TO BE
     C**** COMPARED WITH THE TRUE VALUE 'FNORM')
          CCC=2./3.
          SUM=0.
          DO 110 JX=1,JMXG
1         SUM=SUM+CCC*TR1(JX)*CORSUM(JX)
1     110 CCC=1.-CCC
          CNORM=2.*DT*SUM
     C**** NORMALIZE THE TOTAL CORRELATION FUNCTION WITH THE
     C**** TRUE NORMALIZATION CONSTANT 'FNORM'.
          DO 120 JX=1,JMXG
1     120 CORSUM(JX)=CORSUM(JX)/FNORM
          WRITE(6,1050) FNORM,CNORM,(CORSUM(JX),JX=1,JMXG)
     1050 FORMAT('0THE TRUE NORMALIZATION CONSTANT=',1PE13.5/
         & ' THE NORMALIZATION CONSTANT FOR X.LE.XR(JMXG)=',1PE13.5/
         & '0THE TWO-BODY CORRELATION FUNCTION'/(5X,1P10E12.4))
          DO 130 JX=1,JMXG
1     130 CORSUM(JX)=CORSUM(JX)/XR(JX)**2
          WRITE(6,1060) (CORSUM(JX),JX=1,JMXG)
     1060 FORMAT('0AFTER DIVISION BY X**2'/(5X,1P10E12.4))
          STOP
          END
          SUBROUTINE XTABLE(XR,TR,JMXG,DT)
     C****  T(X)=C*(X+T0)*X/(X+S0)
          DIMENSION XR(*),TR(*)
          DATA EM,XM,TM,ZTXC/  12.0,   6.0,   9.0,   0.3/
          JMX=30
          JMXS=JMXG
          TXC=ZTXC
          SX0=(TM-TXC*XM)*XM/(EM*XM-TM)
          TX0=EM*SX0/TXC
          DT=TM/JMX
          CT0=TXC*TX0
          CS02=2.*TXC*SX0
          CS04=2.*CS02
          TXC2=2.*TXC
          T=0.
          DO 10 I=1,JMXS
1         T=T+DT
1         A=CT0-T
1         XR(I)=(-A+SQRT(A*A+CS04*T))/TXC2
1         DTDX=TXC*(XR(I)**2+2.*SX0*XR(I)+TX0*SX0)/(XR(I)+SX0)**2
1         TR(I)=1./DTDX
1      10 CONTINUE
          WRITE(6,1000) TXC,TX0,SX0
     1000 FORMAT('0XTABLE WITH T(X)=C*(X+T0)*X/(X+S0)'/5X,'C=',1PE13.5,
         &  '   T0=',1PE13.5,'   S0=',1PE13.5)
          WRITE(6,1002) (XR(I),I=1,JMXS)
     1002 FORMAT(' XR=',1P10E12.5/(4X,1P10E12.5))
```

```
              WRITE(6,1001) (TR(I),I=1,JMXS)
         1001 FORMAT(' TR=',1P10E12.5/(4X,1P10E12.5))
              RETURN
              END
              SUBROUTINE PTABLE(AQ,AQ1,DP,PIMX,IMX,M)
        C**** CONSTRUCTS THE P-TABLE.
              DIMENSION AQ(*),AQ1(*)
              DP=PIMX/IMX
              DO 10 I=1,IMX
  1        10 AQ(I)=I*DP
              IMX1=IMX+1
              DP2=0.2
              DO 20 I=IMX1,M
  1        20 AQ(I)=AQ(I-1)+DP2
              DO 30 I=1,M
  1        30 AQ1(I)=1.
              WRITE(6,1000) (AQ(I),I=1,M)
         1000 FORMAT('0 AQ=',1P10E12.5/(5X,1P10E12.5))
              RETURN
              END
```

References

1) T. Sawada and T. Sasakawa; Sci.Rep. Tohoku Univ. Series 8, III(1982) 1. (Referred to as I)

2) T. Sasakawa, S. Ishikawa and T. Sawada; Sci.Rep. Tohoku Univ. Preceding article. (This is the program named PERFECT V. We refer this as II).

3) V. Reid; Ann. Phys. (N.Y.) 50(1968) 411.

4) T. Sasakawa, H. Okuno and T. Sawada, Phys. Rev. 23C(1981) 905.