OAK RIDGE NATIONAL LABORATORY LIBRARIES
3 4456 0547010 2

4

ORNL-4152
UC-32 – Mathematics and Computers

COMPUTER PROGRAM JUPITOR-1 FOR COUPLED-CHANNEL CALCULATIONS

Taro Tamura

OAK RIDGE NATIONAL LABORATORY
CENTRAL RESEARCH LIBRARY
CIRCULATION SECTION
4500N ROOM 175
LIBRARY LOAN COPY
DO NOT TRANSFER TO ANOTHER PERSON.
If you wish someone else to see this report, send in name with report and the library will arrange a loan.
ORNL-118 (6-97)

OAK RIDGE NATIONAL LABORATORY
operated by
UNION CARBIDE CORPORATION
for the
U.S. ATOMIC ENERGY COMMISSION

Printed in the United States of America. Available from Clearinghouse for Federal Scientific and Technical Information, National Bureau of Standards, U.S. Department of Commerce, Springfield, Virginia 22151
Price: Printed Copy $3.00; Microfiche $0.65

LEGAL NOTICE

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

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

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

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

ORNL-4152

Contract No. W-7405-eng-26

PHYSICS DIVISION

COMPUTER PROGRAM JUPITOR-1 FOR COUPLED-CHANNEL CALCULATIONS

Taro Tamura

AUGUST 1967

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

OAK RIDGE NATIONAL LABORATORY LIBRARIES

3 4456 0547010 2

TABLE OF CONTENTS

		Page
I.	INTRODUCTION	2
II.	CODING OF THE PROGRAM	6
	A. General Remarks Concerning the Coding of the Program	6
	B. Names and Roles of the Subroutines	7
	C. Detailed Explanation of Each Subroutine	9
	1) JP1MAIN	11
	2) CCCTRL	36
	2-1) NLJJJK	38
	3) FLGLCH	39
	3-1) FLGLNG	39
	4) POTENT	39
	5) COUPLE	40
	5-1) BMATRX	40
	5-2) AMATRX	40
	6) SMATRX	45
	6-1) SIMLEQ	45
	7) XSEC	45
	7-1) ACCSAB	45
	8) CROSPL	45
	8-1) CMMMFC	45
	8-2) BFCTOR	45
	9) OUTPUT	50
	9-1) PLOTER	50
	10) RAC7	50

TABLE OF CONTENTS (Cont'd.)

		Page
	10) CLEB	50
	APPENDIX A	51
	APPENDIX B	53
	APPENDIX C	57
III.	SUGGESTIONS FOR POSSIBLE MODIFICATIONS TO THE PRESENT PROGRAM	61
IV.	EXAMPLES OF THE INPUT-DATA CARDS	64
V.	EXAMPLES OF MACHINE OUTPUTS	68
VI.	FORTRAN LISTING OF JUPITOR-1	76

ABSTRACT

This report consists of the FORTRAN listing and the corresponding description of a computer program named JUPITOR-1 (abbreviated JP1). This program can be used to perform coupled-channel calculations to evaluate the cross sections for the scattering of nuclear particles by nuclei. Preparations of the input data cards are explained in great detail with illustrations. Ways of possible modification or expansion of the present program are also suggested.

I. Introduction

This report is the description of a computer program named JUPITOR-1 and abbreviated JP1. This is a program for coupled-channel calculations that calculate the scattering cross sections of nuclear particles by various (collective) nuclei.

This program has a vast flexibility as can be seen below. To allow for this flexibility, including the possibility of handling very large sized coupled differential equations, it has become inevitable that the speed of computation when only coupled equations of smaller size are handled is relatively slow. In order to handle with higher speed the problems in which only such smaller sized sets of equations appear, another program named JUPITOR-2 was written and is used. The description for JUPITOR-2 is under preparation and will appear as another ORNL report sometime later.

The writing of the program JP1 was started at the end of 1962. Afterwards various possibilities were added one by one, and correspondingly various calculations were performed with this program. A review of the results of such calculations was published in 1965, which also contained details of the formulation of the coupled channel calculations.[1] Some additional modifications were made in the program after the above review article was written, and the results of calculations made with such new possibilities were also published.[2-5]

The present report is written assuming that the reader is familiar with these publications, particularly with T[1]. Therefore, reference to the notations and equations of T will be made frequently without renewed explanations of them. The program contains, however, also some other

possibilities which have not yet been used for production runs and the results of which thus have not been published anywhere so far. Their formulations are included in Appendix A of the present report.*

Section II of this report is devoted to the detailed explanation of all the routines (in CDC-FORTRAN-63) that constitute the source deck of the present program. In Sec. II-A remarks on the general structure of the coding of the present program are given. A list and rough explanations of the functions of the various routines are given in Sec. II-B, together with a very simplified flow diagram of the whole program. Detailed explanation of each routine is given in Sec. II-C, which is divided into subsections devoted to the explanation of each subroutine. Among these the explanations of the main routine (JP1MAIN) exhausts the explanation of how the input-data cards are to be prepared. Therefore, to the readers who want to use this program just as it is, it is sufficient to read only this subsection.

In Sec. III suggestions on how the present program may possibly be modified or enlarged are given. Sections IV and V give, respectively, examples of input-data cards and some of the machine outputs. The FORTRAN listing† of the whole program is given in Sec. VI.

In the following, several abbreviations are used. They are as follows: (See T for more detail.)

CC = coupled-channel calculations

ACC = adiabatic coupled-channel calculations

NACC = non-adiabatic coupled-channel calculations

---

*Appendices A, B and C are found at the end of Sec. II.

†As of May, 1967.

RFF = real form factor

CFF = complex form factor.

We shall close this section by giving a list of possibilities that can be handled with the present program.

(i) Spin s of the projectile can be either 0, $\frac{1}{2}$, or 1 (s = 0 or $\frac{1}{2}$ for ACC).

(ii) Targets can be anything, though those with some collectivity are of primary interest. They can be either vibrational (spherical) or rotational (permanently deformed), and of either even or odd A.

(iii) When the target is deformed, either NACC or ACC can be made.

(iv) When the target is deformed, excitation of states belonging to higher (vibrational) bands can be considered. (Of course, NACC is to be used in this case.)

(v) Coulomb excitation can be included.

(vii) The form factor can be either real (RFF) or complex (CFF).

(viii) Up to six states can be coupled at one time.

(ix) Up to thirty partial waves can be coupled at one time: (Up to twenty-five partial waves, if the projectile energy becomes negative in some excited channels.)

(x) The maximum value of the orbital angular momentum, ℓ_{max}, is 69, though it is not difficult to modify the program so as to make ℓ_{max} larger. (See Sec. III).

(xi) Can compute the differential cross sections for up to 100 angles for any number of states (Up to 35 angles if a

polarized beam or target is considered.) Total and reaction cross sections and s- and p-wave strength functions can also be computed (Only differential and total cross sections for ACC.)

(xii) If the projectile is a neutron, its energy in some excited channels can be negative.

(xiii) Automatic plot of the theoretical and experimental differential cross sections and polarizations can be made. (Up to six states for cross sections and up to two states for polarizations.)

The original version of the present program had the possibility to compute also the Hauser-Feshbach (HF) cross sections. In the present version, however, this possibility was deleted, because of the following reason: The evaluation of HF cross sections was made previously[6] and it was found that the result (see (T-57)) obtained by using the C-matrices (or in turn the transmission coefficients) obtained by solving the coupled-equations is very similar to that obtained by using the transmission coefficients derived from the uncoupled optical model calculation. The latter calculation is certainly much easier than ours. It is also known[7] that the HF theory is only approximately correct in evaluating the compound-process contribution to the differential cross sections. Therefore, it does not seem meaningful to retain that possibility in our program.

II. Coding of the Program

A. General Remarks Concerning the Coding of the Program

The present program has been coded aiming at satisfying at the same time the following three requirements as much as possible.

(i) To be general; (i.e., can handle various projectiles and targets with one program deck).

(ii) To be fast in computation.

(iii) To be easy to understand; (i.e., easy to read and use the program).

The requirements (i) and (ii) conflict with each other, unless one has access to a computer with a very large core storage. When such a conflict is unavoidable the requirement (i) has been given the priority. Namely, the coding was made to allow for a variety of possibilities, with the possible expense of the running speed. Thus, in the present program two scratch tapes are used very frequently, making the running speed slower and the understanding of the program harder than otherwise.

Concerning the above item (iii), a policy was taken of having the statement numbers that are used in each routine appear always in an increasing order. This makes it easy to follow the flow of the logic of the program. (An exception to this rule is made for statement numbers 4000 ~ 4999. They are attached to the statements which handle the intermediary output (See Sec. IIC-1). Another policy taken was to keep all the routines in approximately the same size. Also the quantities that are communicated from one routine to the other are all placed in the COMMON field. No subroutine exists which has arguments.

The logical numbers of tapes used by this program are the following:

TAPE 5 = Standard Input Tape,

TAPE 6 = Standard Output Tape,

TAPE 7 and TAPE 8 = Scratch Tapes.

The present program was coded to be run on a CDC-1604 computer, and the language used is that of FORTRAN 63. It was not tested, but it is believed that an IBM machine for which a FORTRAN-IV compiler is available could also be used to run this program with very little modification, if any. (See also remarks in Sec. IIC-1, particularly concerning the use of KTRL(27) = 1.)

A big difference between FORTRAN 63 and FORTRAN-IV is that the maximum number of alphanumeric characters that can be used to describe the variable names is 8 in the former, but is 6 in the latter. All the names of variables that appear in the present program are described with less than 6 characters, except JP1MAIN which is the name of the main routine.

B. Names and Roles of the Subroutines

We first give the list of the subroutines that constitute the source deck of our program. The number that is given in parenthesis that follows each name is the serial number of each subroutine, and its use will be explained later (cf. Sec. IIC-1).

JP1MAIN (1),	CCCTRL (2),	NLJJJK (2-1),	FLGLCH (3),
FLGLNG (3-1),	POTENT (4),	COUPLE (5),	BMATRX (5-1),
AMATRX (5-2),	SMATRX (6),	SIMLEQ (6-1),	XSEC (7),
ACCSAB (7-2),	CROSPL (8),	CMMMFC (8-1),	BFACTR (8-2),
OUTPUT (9),	PLOTER (9-1),	RAC7 (10),	CLEB (10-1).

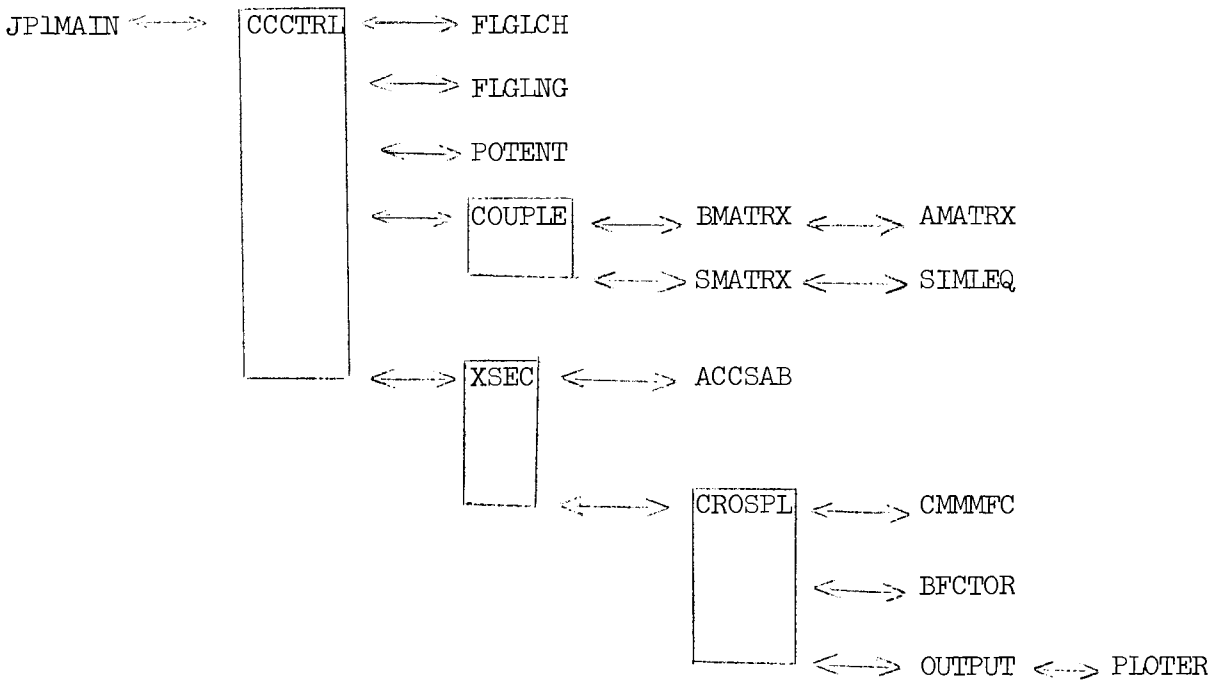
The functions which these routines play are very roughly as follows: First, the main routine JP1MAIN reads all the input data cards, and then outputs them immediately after reading. This routine then calls CCCTRL, the name meaning control of the coupled-channel calculation. CCCTRL first fixes various basic quantities, like reduced mass, center of mass energy, matching radius, etc. It then calls FLGLCH or FLGLNG to compute the Coulomb wave functions for positive or negative energy projectiles at the matching radius, and then calls POTENT to compute the potentials that appear in the coupled equations. From CCCTRL also the routine COUPLE is called, which itself first calls AMATRX and BMATRX to establish the geometrical part of the coupled equations, and then solves that equation. The solutions are used in SMATRX, which is also called from COUPLE, to construct the matching equation. This equation is solved in SIMLEQ that is called by SMATRX, and the C-matrix elements are obtained.

After all the needed C-matrix elements are obtained, CCCTRL calls XSEC, which performs the computation of the cross sections. This routine performs, when NACC is made, partial summations that appear in the expression of the cross sections, while if ACC is made it calls ACCSAB to perform the corresponding partial summations. XSEC then calls CROSPL to perform the rest of the summation. In this process CMMMFC and BFCTOR are called from CROSPL. After the cross sections are computed, CROSPL further calls OUTPUT to output the obtained results. OUTPUT may also call PLOTER to plot the experimental and theoretical cross sections and polarizations.

The small, but important routines NLJJJK, RAC7 and CLEB are called

by various routines. Among them RAC7 and CLEB compute, respectively, the Racah and Clebsch-Gordon coefficients,[8] while NLJJJK sets up the quantum numbers N, L and J of the partial waves that are to be coupled together for a given value of JJ and K. (The meaning of the quantities N, L, J, JJ and K will be explained later.)

Summarizing, the crude flow diagram of the present routine is given as follows:



C. Detailed Explanation of Each Subroutine

In this subsection we give explanations of how each subroutine was coded. If the reader wants to use the present program just as it is, and thus has no intention to modify it, it is sufficient to read only the explanation of the routine JP1MAIN. The reason is that JP1MAIN is the sole routine into which all the input data are read, and thus by

understanding this routine the reader can know how to prepare the input data cards.

As we have emphasized in T, the coupled channel calculations can be formulated in a form quite common for various kinds of projectiles and the targets. It was also emphasized in Sec. III of T that differences in the nature of the target are reflected only through differences in the matrix, called B-matrix there, whose elements are the quantities $\langle I \| Q_\lambda^{(t)} \| I' \rangle$ introduced in (T-27). In the present program this B-matrix is computed in the subroutine BMATRX. Therefore, if the reader wants to extend the present program to use it for calculations which are not included in its present form, he must add or modify an appropriate part in BMATRX. In other words, the understanding of BMATRX is the most important next to that of JP1MAIN. A comparatively detailed explanation is thus given of BMATRX, too.

However, in order to know exactly how the coupled equations are constructed, the reader has to know further how the B-matrix elements are combined with the geometrical factor $A(\ell jI, \ell' j' I', \lambda Js)$ of (T-27), and then with the form factors $v_\lambda^{(t)}(r)$, also introduced in (T-27). How to do this is known by understanding the subroutines AMATRX and COUPLE; therefore some detailed explanations are also given to these two. Explanations of other routines are also given, but with less detail.

In the following an abbreviation SN is used to mean statement number. Therefore, e.g., SN-100 means the card to which the statement number 100 is attached. When no SN is attached to a particular card, we refer to it by saying, e.g., SN-100 + 3 (or SN-100-2) which means that the card we refer to is the third card below (or the second card above)

the one to which SN-100 is attached.

(1) JP1MAIN

This is the main routine. As we remarked above, all the input data are read in from this routine and nowhere else.

In the beginning of this routine (below SN-4100), FACLOG(N), N = 1 ~ 500 are computed. These are log [(N-1)!] and are used mostly in RAC7 and CLEB to compute Racah and Clebsch-Gordan coefficients. The explanation of these two routines were made in a previous ORNL report[8] and thus will not be discussed here again.

At SN-100 the first READ statement appears. It reads 5-cards. All use FORMAT 701 (i.e., 14I5).*

CARD-1 and CARD-2 read in KTRL(N), N = 1 ~ 28. KTRL stands for control and KTRL(N) are used to specify different uses of the present program. The CARD-1 and CARD-2 are completely blank if the present routine is used in its simplest version. Namely, when the target is even-A, spherical nucleus, real form-factor (RFF) is used, no Coulomb excitation is considered, etc. The more sophisticated the calculation gets, the larger number of non-zero values appear in these cards.

When a KTRL(N) is non-zero, its value should be 1, except that KTRL(8) can take values 2 and 3, and KTRL(3) and KTRL(9) can take a value 2. Under what conditions KTRL(N) should be given non-zero values is explained in the COMMENT cards given in the beginning of JP1MAIN. Some of them are clear, but the others may require more detailed

---

*All the cards that read in integer values exclusively have this format, 14I5. On the other hand all the cards that read in floating point values exclusively have the format 10F7.2 (cf. SN-702).

explanation. We shall list here, for completeness, all the KTRL(N) that can take non-zero values. Those which do not appear below are not used and thus can be zero (blank) all the time.

If KTRL(1) = 1, the machine understands that the target has odd-A.

If KTRL(3) = 1, the total and reaction cross sections and the s- and p-wave strength functions (for ACC total cross sections only) are computed, together with the differential cross sections. If KTRL(3) = 2, however, the calculation of the differential cross sections is skipped.

If KTRL(4) = 1 the terms given in (T-39) are included in the coupled equations (which occurs only when the target nucleus is vibrational). In most of the calculations so far made with the present code, however, these terms were omitted. The reason is the following. If these terms are added, it amounts to the modifications of the depth and the radius of the optical potential and thus it affects particularly the elastic scattering cross sections. Therefore the optical model parameters that were fixed through the fit to experimental elastic cross section by the usual optical model analysis (uncoupled calculation) cannot be used anymore. However, in the usual optical model analysis the effect of the terms such as (T-39) must already have been included effectively. Therefore, if we want to use parameters found by such optical model analysis, to add (T-39) anew is redundant.

If KTRL(7) = 1, a deformed nucleus is taken as target and ACC is used.

If KTRL(8) = 0, neither projectile nor target is initially polarized. If both the projectile and the targets are initially polarized (or oriented), make KTRL(8) = 1. If only the projectile (target) is polarized, make KTRL(8) = 2 (= 3).

If KTRL(9) $\neq$ 0, some restrictive calculations are made, but for its full explanation, some preparatory remarks are necessary. If we do NACC, as can be seen in Sec. III of T, a set of coupled equations is given for each set of values of J and Π; cf., (T-17), and the C-matrix elements are designated as $C^{J}_{\ell j;n\ell' j'}$ as in (T-46). On the other hand, if ACC is made, a set of coupled equations is given for each set of values of $\bar{m}_j$ and Π, and the C-matrix elements are designated as $C^{\bar{m}_j}_{\ell j;\ell' j'}$ as in (T-68). (Here $\bar{m}_j$ was introduced in (T-66) while Π equals π times $(-)^{\ell}$, ℓ being one of the ℓ's in (T-66) and π the parity of the intrinsic state.) It is therefore easy to see that the part of the program which performs the evaluation of the C-matrices and another part which evaluates the cross sections by using these C-matrices should have two-fold DO-loops, in one of which the value of J (or $\bar{m}_j$) is varied within some range and in the other the value of Π is varied from +1 to -1 (cf., e.g., SN-605, 605 + 2 in CCCTRL).

In the present program, the notations JJ and K are used as the variables of these two DO-loops. Among them K = 1 and 2 correspond to π = + and -. On the other hand JJ ranges from 1 to JJJMAX, where the definition of JJJMAX and the relation of JJ to J or $\bar{m}_j$ are as follows:

If KTRL(7) = 0

JJ = J+1 if J is integer

JJ = J+1/2 if J is half integer

JJJMAX = the smallest integer that equals or exceeds

$$(\ell_{max} + I_1 + s + 1).$$

Here s is the projectile spin, I_1 is the spin of the ground state of the target and ℓ_{max} is the maximum value of the orbital angular momentum to be considered in the calculation.

If KTRL(7) = 1

$$JJ = \bar{m}_j + 1 \text{ if } \bar{m}_j \text{ is integer}$$

$$JJ = \bar{m}_j + 1/2 \text{ if } \bar{m}_j \text{ is half integer}$$

JJJMAX = the smallest integer that equals or exceeds

$$(\ell_{max} + s + 1).$$

In the following, these two DO-loops are referred to as JJ-DO-loop and K-DO-loop, or collectively as JJ-K-double-DO-loops.

After defining these double DO-loops, the explanation of the use of KTRL(9) $\neq$ 0 is easy. If KTRL(9) = 1 or 2, the value of JJ ranges from KEXCOM(5) to KEXCOM(6), instead of from 1 to JJJMAX, where KEXCOM(5,6) are read in from CARD-3 (see below). Further, if KTRL(9) = 1, the value of K which differs from KEXCOM(7), which is also read in from CARD-3, is omitted from the K-DO-loop. If KTRL(9) = 2 no restriction to the K-DO-loop is imposed. (For the complete understanding of the above statements, see SN-601-1 ~ SN-606 of CCCTRL.)

The use of KTRL(9) $\neq$ 0 may be useful in some test running, because it is often sufficient for test running to be made for a restricted set of values of J and Π.

Another use of KTRL(9) $\neq$ 0 may be made when a resonance is expected to appear for a particular set of values of J and Π. In the preliminary running to investigate the nature of that resonance, the use of KTRL(9) $\neq$ 0 will save machine time. It is to be noted that if KTRL(9) $\neq$ 0, subroutines that compute cross sections are not called. (Cf. CCCTRL, SN-800+1).

If KTRL(11) = 0, the coupling potential (or the form factor) is calculated as given in (T-13). If KTRL(11) = 1, it is calculated

according to (T-15) and (T-16). When the target is deformed, use of KTRL(11) = 1 is strongly recommended.

If KTRL(12) = 0, only the real part of (T-13) or (T-15) is taken; i.e., the real form factor (RFF) is used. If KTRL(12) = 1, the whole of (T-13) or (T-15) is taken; i.e., the complex form factor (CFF) is used. Note that with CFF the time needed to solve the coupled equations is about doubled, compared to that with RFF.

If KTRL(13) = 1, the Coulomb excitation is added. Be aware of the tremendous increase of machine time in using this version, since the matching radius and ℓ_{max} must have larger values than otherwise (see below). Note that for INTYPE (explained in CARD-6 below) equals 6, Coulomb excitation cannot be included.

If KTRL(17) = 1, theoretical elastic differential cross sections are divided by the Rutherford cross section.

If KTRL(19) = 1, the $(\sigma \cdot I)$ type interaction[6] is added. In the present form of the program this possibility exists only for deformed nuclei and with RFF.

The use of KTRL(27) = 1 is made when the present program is run on a computer smaller than the CDC-1604 (e.g., IBM-7090 or 7094)*, and thus the whole program cannot be loaded in the core at one time. In that case, first set up two DECKS, each consisting of the following routines. (The numbers given are the serial numbers of the subroutines defined in Sec. II-B.)

---

*Although the CDC-1604 and the IBM-7094 are both 32 K machines, the length of one word is 48 bits in the former, but is 36 bits in the latter, and 2 commands can be carried in one word in the former, but only 1 command in the latter. This, as well as other features like the smaller number of index registers, effectively makes the IBM-7094 a smaller machine than the CDC-1604, as far as the loading of the program part is concerned.

DECK-1: 1,2,2-1,3,3-1,4,5,5-1,5-2,6,6-1,10,10-1 and the Dummy of 7.

DECK-2: 1,2,2-1,7,7-1,8,8-1,8-2,9,9-1,10,10-1 and the Dummies of 3,3-1,4,5.

Here Dummy of 7, e.g., means a Dummy routine of XSEC and in FØRTRAN is one card which looks like

```
SUBROUTINE    XSEC    $    RETURN $    END
```

DECK-1 and DECK-2 expect to have the same set of data cards, except that KTRL(27) = 0 in the former, while KTRL(27) = 1 in the latter. By using the DECK-1 all the needed C-matrix elements are computed and are saved in the scratch TAPE 7. In the DECK-2 they are read in from that TAPE and are used to compute the cross sections. (See CCCTRL, SN-560 for more detail.)

CARD-3 (SN-100) reads in KEXCOM(N), N = 1 ~ 14. Not all of them are used. For those that are used, brief accounts of their use are given in the comment cards in the beginning of JP1MAIN.* We shall give here somewhat more detailed explanations.

First KEXCOM(2) and KEXCOM(4) are used to fix the matching radius R_m. In order to see how they are used (cf. CCCTRL, SN-181 ~ 183) we first note that there are four sets of the radius and diffuseness parameters, r_o and a, i.e., for the real and imaginary parts of the Saxon potential, derivative-type imaginary potential and the spin-orbit potential. Using these four sets of r_o and a, four values of $R_o = r_o A^{1/3} + 10 \times a$ are first computed (CCCTRL, SN-167+1 ff). Their maximum is

*These comment cards include also explanations of KEXCOM(N) with N > 14.

then taken and is called r_{cpl} (= XCPLE in CCCTRL). In the program this XCPLE is redefined as the product of NXCPLE times DX, where NXCPLE is the truncated value of the ratio XCPLE/DX, while DX is the mesh size chosen in solving the coupled equations. If KTRL(13) = 0 the matching radius R_m (called XMAX in the program) is then set equal to XCPLE and the ratio NXMAX = XMAX/DX is equated to NXCPLE. In some cases, however, it is preferred to give NXMAX = NSCPLE as an input number. In that case make KEXCOM(2) $\neq$ 0. Then NXMAX and NXCPLE are equated to KEXCOM(2).

If KTRL(13) = 1, R_m must be taken as[9]

$$R_m = 2\eta_1/k_1\theta_{min}. \tag{2-1}$$

Here η_1 and k_1 are the Coulomb parameter and the wave number, respectively, in the ground channel, while θ_{min} is the smallest angle (in radian) up to which the effect of the Coulomb excitation is to be evaluated accurately. Usually R_m of (2-1) is much larger than r_{cpl} defined above. KEXCOM(4) is then used to redefine NXMAX as the sum of previous NXMAX plus KEXCOM(4), (cf. CCCTRL, SN-183). (Between this new R_m and r_{cpl} no nuclear interaction is considered (cf. COUPLE for more detail.)

How to use KEXCOM(5 $\sim$ 7) was already explained above in connection with the use of KTRL(9) = 1 or 2.

When the target is an odd-A vibrational nucleus and (with ground state spin 1/2) if excitations of states in which the even-core is excited to one of the two-phonon I = 0, 2 or 4 states is considered, the quantity KEXCOM(12) should have one of the values of I = 0, 2 and 4.

When the target is an even-A vibrational nucleus and if the excitation of the three phonon states is to be included in the calculation,

give the number of such states to KEXCOM(13). In the present shape of the program, however, only KEXCOM(13) = 1 (and 0) is allowed.

CARD-3 finally reads in KEXCOM(14) which equals $\ell_{max}+1$. Since JJJMAX increases with KEXCOM(14), and since it is desired to keep JJJMAX as small as possible to save machine time, the smallest possible value is to be chosen for KEXCOM(14). When KTRL(13) = 1 to take $\ell_{max} \geq 2\eta_1/\theta_{min}$ is requested.[9]

CARD-4 and CARD-5 (in SN-100) read in KTLOUT(N), N = 1 ~ 28. Here KTLOUT stands for output control and is used to control the intermediary outputs. If these two cards are blank, there will be no intermediary output. The more non-vanishing values (usually = 1) are given to KTLOUT(N), the larger number of intermediary output is given.

What quantities can be output is known precisely only after understanding all the subroutines. Here we note the following general rule in using KTLOUT(N).

In Sec. II-B above, we gave a serial number to each routine, which ranged from 1 to 10. If one wants to get an output from a routine whose serial number is n, then make KTLOUT(2n-1) and/or KTLOUT(2n) equal to 1. Of these two, KTLOUT(2n-1) outputs quantities of smaller amount, but of more physical or computational importance, than KTLOUT(2n) does. For example, take COUPLE, BMATRX and AMATRX, which are grouped to have n = 5. As is seen in the end of AMATRX, the B-matrices are output if KTLOUT(9) = 1, while the A-matrices are output if KTLOUT(10) = 1. The B-matrices which contain all the physics involved in the coupling scheme are more crucial than the A-matrices, the latter being obtained by multiplying the

B-matrices with the well defined geometrical factors.* Further, the amount of the output of the B-matrices is much smaller than that of the A-matrices.

CARD-6 reads in EXTCOM(N); N = 1 ~ 10. In the present shape of our program none of them are used. Therefore, this card can be blank. It may be that the modification of JP1MAIN is to be made so as to drop the card SN-100+1 altogether, but we prefer to retain it. The reason is the following.

The name EXTCOM means extra-common. Many of them are unused at present, but they are useful when a small modification of the program is to be made in which communication of a quantity from a routine to another is needed, since EXTCOM lies in the COMMON field (occupying 50 core storages altogether). EXTCOM(N) in the CARD-6 is useful when the quantity to be communicated is to be given as input data.

As was seen in CARD-3, there are quantities which are not used in KEXCOM(N), N = 1 ~ 14. They can be used to read in some new integer quantities. KEXCOM(N), N = 15 ~ 50 can be used to communicate new integer quantities between subroutines.

In the comment cards given in the beginning of JP1MAIN, there is given a list of KEXCOM(N) and EXTCOM(N) that are already used in the present shape of our program.

CARD-7 reads in ten integer quantities. Their meaning and use are as follows:

---

*It should be noted that the A-matrices in the present program are the product of the B- and A-matrices of T, and not the A-matrices of T themselves. It should also be noted that some non-vanishing values must be given to KEXCOM(5), (6) and (7) to get output of A- and B-matrices. (Cf. AMATRX, SN-4210 ~ 4214.)

(i) IICPLE equals the number of coupled states in the target and thus equals N_c of (T-19). (If KTRL(7) = 1, IICPLE = 1 always.)

(ii-iii) INTYPE and INTMAX specify the type of the coupling and the number of coupling terms, respectively, and for their complete understanding the reader has to understand the subroutines COUPLE, BMATRX and AMATRX. A brief explanation of these quantities will, nevertheless, be given in the explanation of the CARD-19 below.

(iv) ISTRTW is twice the value of the projectile spin s.

(v) NANGLR is the number of angles for which the differential cross sections (and polarizations) are to be computed.

(vi) IIXCAL is the number of states for which the differential scattering cross sections are to be computed. For NACC, usually IIXCAL equals IICPLE, but there are cases, for example, in which the energy of the incident particle is so low that the projectile gets a negative energy when the target is in an excited state. In such a case IIXCAL is smaller than IICPLE. For ACC usually IIXCAL $>$ IICPLE = 1.

(vii) IIXPLT is the number of states for which the theoretical and experimental differential cross sections are to be machine-plotted. Clearly IIXPLT $\leq$ IIXCAL. Note that IIXPLT must be ≤ 6.

(viii-ix) IIPCAL and IIPPLT serve the same role for polarizations as IIXCAL and IIXPLT serve for differential cross sections. Note that IIPPLT must be ≤ 2.

(x) KANGRD equals either 0 or 1. How to use these two values will be seen below in the explanation of CARD-10.

CARD-7-1 (SN-102) is needed to be in the data deck only when INTYPE = 6. If INTYPE = 6 this card reads in KEXCOM(N), N = 21 ~ 27. Note that when INTYPE = 6 the target is a deformed even nucleus and excitation of vibrational states are also considered. KEXCOM(N), N = 21 ~ 27, respectively, give the number of states considered in the coupling in each of the following seven bands: (i) ground band; (ii) $K = 0^+$ (β-vibrational) band; (iii) $K = 0^-$ band; (iv) $K = 2^+$ (γ-vibrational) band; and (v-vii) $K = 1^-$, 2^-, and 3^- bands. See the explanation of CARD-19 for more detail.

CARD-8 (SN-103) reads in the quantities that specify the nature of the target states. (FORMAT is 3(2I5, F10.3)). This card reads in three quantities for each state. (i) IIREAD(N) = I_N if KTRL(1) = 0; IIREAD(N) = $I_N + \frac{1}{2}$ if KTRL(1) = 1. (ii) KPRITR(N) = 1 or 2, depending on whether the parity of the N-th states is + or -. (iii) QVALUE(N) is the excitation energy (thus the negative of the usual Q-value) of the N-th state above the ground state. Therefore QVALUE(1) is zero always and QVALUE(N) > 0 for N $\geq$ 2. If IICPLE $\geq$ 4 for NACC or IIXCAL $\geq$ 4 for ACC more than two cards of type CARD-8 are needed.

The order with which these states appear in CARD-8 must satisfy certain rules. These rules are shown in the explanation of CARD-19.

CARD-9 (SN-105) reads in seven floating point quantities that follow:

(i) ELAB = E_{lab} of the projectile in MeV.

(ii) PMAS = Mass of the projectile in amu or pmu depending on whether AMUPMU = 0 or 1; see (vii) below. However, the use of an integer value may be made.

(iii) TMAS = Mass of the target. Remarks concerning the unit is the

same as in (ii).

(iv) CHARGE = ZZ′ of (T-7), i.e., the product of the charge numbers of the projectile and the target. Thus if the projectile is a neutron, CHARGE = 0.0.

(v-vi) XMES1 and XMES2 are the mesh sizes used in the differential equation integration. They are equal to h_1 and h of Appendix B, respectively. The use of XMES1 = 0.0125 and XMES2 = 0.1 is usually recommended.

(vii) AMUPMU = 0.0 (= 1.0) if atomic (proton) mass unit is to be used.

CARD-10 reads in ANGLER(N), i.e., the center of mass angles θ_{CM} in degree. If KANGRD, which was read in from CARD-7 were zero, SN-111 of JP1MAIN is executed and thus the cards should contain values of ANGLER(N) for N = 1 ~ NANGLR. Therefore the needed number of the cards, N_{10} say, of the type CARD-10 is given by N_{10} = ((NANGLR-1)/10)+1. On the other hand, if KNAGRD = 1, which is used when all the θ_{CM} are equally spaced, SN-112 is executed and CARD-10 needs to carry only two quantities, the smallest value of θ_{CM} and the increment $\Delta\theta_{CM}$.

CARDs-10-1 (SN-127). These cards read in the experimental differential cross sections SGMEXP(N,I) (in mb/sterad) and the experimental polarization POLEXP(N,I). They are needed in the card deck only when either or both of IIXPLT and IIPPLT are non-zero. The number of cards needed equals N_{10} times (IIXPLT + IIPPLT).

As is seen in PLOTER, the plotting of $\sigma_{th}(\theta)$ and $\sigma_{exp}(\theta)$, and $P_{th}(\theta)$ and $P_{exp}(\theta)$ (the theoretical and experimental cross sections and polarizations) is made on a sheet of paper taking a fixed spacing between the

neighboring angle, irrespective of whether the actual difference of the neighboring θ is a constant or not. Therefore, it is recommended to use θ of an equal spacing when the machine plot is to be made. Thus, when original σ_{exp} and P_{exp} are not given with equally spaced θ, plot σ_{exp} and P_{exp} on a sheet of paper, draw a line through, read off the values at equally spaced angles θ and use them as $\sigma_{exp}(\theta)$ and $P_{exp}(\theta)$, i.e., as SGMEXP(N,I) and POLEXP(N,I).

It should also be noted that the number of angles NANGLR for which the cross sections are to be calculated and then plotted is commonly used for all the states. In many practical cases, however, it happens that the range of the angles for which experimental cross sections are supplied differs from a state to another. In such a case fix NANGLR so that the widest range of angles are covered. For the angles (in other states) where no experimental data exist <u>never</u> make SGMEXP(N,I) equal to zero. (The computer is requested to take a logarithm of zero and may stop.) It is recommended to give to such SGMEXP(N,I) the following values. Suppose there are experimental cross sections for $N = N_{min} \sim N_{max}$. Make SGMEXP(N,I) = SGMEXP(N_{min},I) for $N \leq N_{min}$ and SGMEXP(N,I) = SGMEXP(N_{max},I) for $N \geq N_{max}$. The same remark applies to POLEXP(N,I), too.

<u>CARD-11</u> (SN-132). This card with FORMAT (I5,4F7.2) is needed in the card deck only when KTRL(8) $\neq$ 0. The first quantity NFAI is the number of different azimuthal angles φ for which the differential cross sections $\sigma(\theta,\varphi)$ are to be computed. Up to two values of φ can be considered at present. FAI(N), $N = 1 \sim 2$ are such φ values in degree. Note that for KTRL(8) $\neq$ 0 there is an initial polarization and thus the

differential cross sections are usually functions of φ. It is clear that SN-135 changes FAI(N) from degree to radian, while SN-138 gives to THETA(N) in radian the values of θ_N, which was given to ANGLER(N) in degree.

Next comes a set of cards that starts at SN-138+1 and ends at SN-290. This part is used to evaluate the values of the amplitudes $a_{m_s}^{(i)}$ and $b_{M_1}^{(i')}$, defined in Sec. IV of T, and store them in AMS(M1,M2,N1) and BMTR,I(M1,M2,N1), respectively. Here AMS is real but BMTR,I is in general complex. The reason why AMS can always be real is that it is always possible to choose the plane defined by the direction of the motion of the projectile and the direction of the polarization of the projectile as the x-z plane. On the other hand, b_{M_1} may take complex values.

The meaning of the arguments M1, M2 and N1 in AMS and BMTR,I are as follows: N1 = 1 ~ NPOLST (NPOLST $\leq$ 3) is used to specify different initial polarizations. For example, suppose we have a target of known initial polarization, and want to compute the cross sections. Let us say that the target is in the N1 = 1 polarized state and assume that BMTR,I are defined accordingly. We may further want to compute the cross section for the case in which the target is unpolarized but everything else (i.e., parameters, coupling-scheme, etc.) is the same as in N1 = 1 case. Let us say that the target is in N1 = 2 polarized state, and assume that BMTR,I are again defined accordingly. If no other states are considered, put NPOLST = 2. The reason to allow the use of NPOLST > 1 is the following: Most of the machine time in performing CC is taken in evaluating the C-matrices. Once they are obtained, the time

needed to evaluate the cross sections with two or more sets of BMTR,I is not very long compared with the time needed for evaluating the cross sections for one set of BMTR,I. We thus want to evaluate the cross sections for as many polarized states as possible at one time. (The same remark applies also to the possibility of NFAI > 1.) When there is no initial polarization, it is clear that one can put NPOLST = 1 (which is automatically made by the computer, SN-138+1).

The argument M2 in AMS is essentially m_s of a_{m_s}. Therefore M2 ranges from 1 to 2s + 1. If s = 0, M2 = 1 means $m_s = 0$. If s = 1/2, M2 = 1 and 2 means $m_s = 1/2$ and -1/2. If s = 1, M2 = 1, 2 and 3 means $m_s = 1$, 0 and -1. M2 in BMTR,I is essentially M_1 of b_{M_1}. M2 ranges from 1 to MULT1Z = $2I_1 + 1$. The relation between M2 and the actual value of M_1 is given by M_1 = M2-1 = 0 if $I_1 = 0$, while if I_1 is half integer (i.e., if KTRL(1) = 1), M2 = 1, 2, ... MULT1Z means that $M_1 = I_1, I_1-1, \ldots -I_1$.

The argument M1 in AMS and BMTR,I specifies different ensembles in a given polarized state. Therefore M1 = i or i′ of $a^{(i)}_{m_s}$ or $b^{(i')}_{M_1}$. The range of i and i′ which are called NENSBP(N1) and NENSBT(N1) are generally $1 \sim 2s + 1$ and $1 \sim 2I_1 + 1$, but can be narrower.

When there is no initial polarkzation, i.e., if KTRL(8) = 0, AMS(M1, M2, N1) = $\delta_{M1,M2}/\sqrt{2s + 1}$ and BMTR,I(M1, M2, N1) = $\delta_{M1,M2}/\sqrt{2I_1 + 1}$ See (T-62) and (T-61). In JP1MAIN these evaluations of AMS and BMTR,I are made at SN-155 and SN-160. Thus none of the CARDs-12 ~ 16 which describe the initial polarization need be in the deck if KTRL(8) = 0. If KTRL(8) ≠ 0 part or all of the CARDs-12 ~ 16 must be in the deck.

<u>CARD-12</u> (SN-205) reads in NPOLST, and NENSBP(N1) and NENSBT(N1) for

N1 = 1 ~ 3. The latter quantities which correspond to N1 > NPOLST can be zero (or any value).

At SN-205+1 there starts a DO-loop with NSP = 1 ~ NPOLST. A set of CARDs-13 ~ 16, or part of it is read in each time this DO-loop is executed.

CARD-13 (SN-195+1) must be in the deck if KTRL(8) = 1 or 2 and reads in AMS(M1, M2, NPS). The way how to prepare this card is easy to see from (T-62) or (T-63) and SN-195+ 1. (Note that MULSPN means 2s + 1.)

If KTRL(8) = 1 or 3, there must be CARDs-14 ~ 16 in the deck.

CARD-14 (SN-215+2) reads in MENSBT(N1,NPS) for N1 = 1 ~ NENSBT(NPS). MENSBT equsls $N_1 + \frac{1}{2}$, where N_1 is defined in (T-59) and/or (T-60).

CARD-15 (SN-215+3) reads in OCCUPA(N1,NPS) for N1 = 1 ~ NENSBT(NPS). OCCUPA(N1,NPS) = P_{N1}, where P_{N1} was defined in (T-58).

CARD-16 (SN-215+4) reads in EULER (N1,NPS), N1 = 1 ~ 3. They are just the Euler angles θ_1, θ_2 and θ_3 (in degree) that are defined in (T-60).

Using the quantities that are read in from CARDs-14 ~ 16, BMTR,I is computed in the DO-loop that starts at SN-225+2 and ends at SN-260. The formula used is as follows: (See (T-60) and e.g., page 22 of ref. 10).

$$b_{M_1}^{(N_1)} = D_{N_1M_1}^{(I_1)}(\alpha,\beta,\gamma)\sqrt{P(N_1)}^{\frac{1}{2}}$$

$$D_{N_1M_1}^{(I_1)} = \exp[-i(\alpha N_1 + \gamma M_1)]\, d_{M_1N_1}^{I_1}(\beta)$$

$$d_{mn}^{j}(\beta) = [(j+m)!(j-m)!(j+n)!(j-n)!]^{\frac{1}{2}}$$

$$\times \sum_t [(j+m-t)!(j-n-t)!t!(t+n-m)!]^{-1} \cos(\beta/2)^{2j+m-n-2t} (\sin(\beta/2))^{2t+n-m}.$$

CARD-17 (SN-400) reads in the strength parameters of the optical potentials. To make easy the explanation of this card and the next card (CARD-18), we shall call the real and imaginary parts of the Saxon potential, surface type imaginary potential and the spin-orbit potential, as No. 1 ~ No. 4 potentials, respectively.

VSX = depth of No. 1 potential = V of (T-1)

WSX = depth of No. 2 potential = W of (T-1)

WSF = depth of No. 3 potential = W_D of (T-1)

VSO = depth of No. 4 potential = V_{so} of (T-1)

CARD-17 also reads in WC(N), N = 1 ~ 6. They are the "fudge" factors for the strength of the No. 2 and/or No. 3 potentials. See Sec. IV-(ii) and IV-(iv) of T. See also ref. 2. WC(N) for N > IICPLE is of no meaning (and not used). (Note that the actual strength of imaginary potentials in channel N equals WSX or WSF times WC(N). Thus if no fudging is to be done, WC(N) = 1.0, not = 0.)

CARD-18 (SN-400+1) reads in the geometrical parameters of the optical potentials. (DFN,RZERO), (DFNW,RZEROW), (DFNS,RZEROS) and (DFNSP,RZROSP) are the diffuseness parameter (a or $\bar{a}$ of (T-1)) and the radius parameter (r_o or $\bar{r}_o$ in (T-3)) for the potentials No. 1 through No. 4. RZEROC is the radius parameter for the Coulomb potential and equals R_c of (T-8) divided by $A^{1/3}$. (Order of reading is that four a's are read first and then five r_o's.)

CARD-19 (SN-400+2) reads in VCOULE(N), N = 1 ~ 10 and an integer quantity KREAD1.

VCOUPL is essentially the value of β_λ introduced in (T-2) and (T-32)

or in (T-3). Since the use of the β-values in the present program is somewhat more complicated than was explained in T, and the β-values are the most important quantities that one wants to obtain from the fit of the theoretical cross sections of CC to experiment, we shall give here a detailed explanation of VCOUPL. We noted above that the complete understanding of the CARDs-7 and 8 awaits the knowledge of VCOUPL. We shall thus give here the complete explanation of these two cards, too.

The quantities in CARD-7 for which a detailed explanation has to be made here are INTYPE and INTMAX. The former, i.e., INTYPE stands for "interaction type" and by giving one of the values from 2 to 6 to it, a particular type of coupling scheme is chosen. On the other hand INTMAX roughly corresponds to the number of the coupling terms that appear in (T-27). What kind of coupling scheme is considered for a given INTYPE, and correspondingly what value is to be given to INTMAX will be explained below separately for INTYPE = $2 \sim 6$.

<u>INTYPE = 2</u> is used when the target is an even-spherical nucleus. If further INTMAX = 1, a very simple 0^+ - I^π coupling is considered, where 0^+ is, of course, the ground state, while I^π is any one-phonon type (or two-quasiparticle type) state of natural parity: ($I^\pi = 0^+, 1^-, 2^+, 3^-, 4^+, 5^-, \ldots\ldots$). In CARD-8, these two states should appear in the order of (i) 0^+ and (ii) I^π. The value of β_λ of (T-36) should appear in CARD-19 as VCOUPL(1).

If INTMAX = 2, we can consider the coupling 0^+-2^+-3^-, the states 2^+ and 3^- being, respectively, the quadrupole and octupole one-phonon states. The order of appearance of these states in CARD-8 must be of the order (i) 0^+, (ii) 2^+ and (iii) 3^-. The values of β_2 and β_3 of (T-36.1) should be given as VCOUPL(1) and VCOUPL(2) in CARD-19.

In both cases of INTMAX = 1 and 2, the coupling terms considered are only the first terms of (T-13), and thus only the first derivative of the original optical potentials appears as the form factor. If INTMAX = 5, and if still 0^+-2^+-3^- coupling is considered, the contribution of the second term of (T-13) is included. In this case the necessary B-matrix elements are evaluated by using (T-37.4). In (T-37.4) the same notation β_2 and β_3, as introduced in (T-36.1) appear, but the present program allows to use different values to them. Thus in full, there should be given four non-zero values in CARD-19. Among them VCOUPL(1) and (2) are the same as in the case of INTMAX = 2. On the other hand VCOUPL(3) and (4) are, respectively, the values of β_2 and β_3 that appear in (T-37.4).

In the above 0^+-2^+-3^- coupling, IICPLE of CARD-7 was, of course, equal to 3. If INTMAX = 5 is still taken, but if IICPLE is put equal to 4, one can consider the excitation of one of the I^- states that are the quadrupole-octupole-two-phonon states, ($I = 1^-, 2^-, 3^-, 4^-$ or 5^-).[11] In this case this state I^- should appear as the fourth state, (iv) I^-, in CARD-8. In the present shape of our program, the same values of VCOUPL(3) and VCOUPL(4) are used to giving values of β_2 and β_3, respectively, in all the matrix elements of (36.4), (36.5), and (37.3) as well as of (37.4). The removal of this restriction is easily made by a slight modification of BMATRX.

Finally it should be noted that if IICPLE = 1, INTYPE = 2 and INTMAX = 1 are taken, our program is used as an usual optical model program (though it is slow to be used in this sense; cf., the remark made in the Introduction).

<u>INTYPE = 3</u> is used also for even-vibrational nuclei, but here we

consider the excitation of quadrupole multi-phonon states. In the present form of our program the most extensive coupling that can be considered is 0_o^+-2_1^+-0_2^+-2_2^+-4_2^+-I_3^+ coupling, where the suffices indicate the number of the quadrupole-phonons in each state, while I_3 is one of the values of 0, 2, 3, 4, 6. INTYPE = 3 allows for a variety of smaller sized coupling schemes between the above six states. Namely we have the following variety of coupling schemes:

(a) IICPLE = 3: 0_o^+-2_1^+-I_2^+($I_2^+ = 0_2^+$, 2_2^+ or 4_2^+).

(b) IICPLE = 4: 0_o^+-2_1^+-I_2^+-$I_2'^+$ (I_2^+ and $I_2'^+$ are any combinations of two of the 0_2^+, 2_2^+ and 4_2^+), or 0_o^+-2_1^+-I_2^+-I_3^+.

(c) IICPLE = 5: 0_o^+-2_1^+-0_2^+-2_2^+-4_2^+, or 0_o^+-2_1^+-I_2^+-$I_2'^+$-I_3^+. (I_2 and I_2' the same as in (b)).

(d) IICPLE = 6: 0_o^+-2_1^+-0_2^+-2_2^+-4_2^+-I_3^+.

The order with which these states appear in CARD-8 must obey the following rule: the first and the second states should always be 0_o^+ and 2_1^+, respectively, while if I_3^+ state is included in the coupling it must be the last state in CARDs-8. As the third through IICPLE-th or (IICPLE-1)-th, states there appear two-phonon-state(s) and if more than two of such two-phonon states appear, their order of appearance among themselves in CARDs-8 is immaterial, (e.g., the orders 0_o^+-2_1^+-2_2^+-4_2^+ and 0_o^+-2_1^+-4_2^+-2_2^+ are effectively the same).

In order to explain how to prepare CARD-19, we introduce here the following parameters: β_{02}, β_{2I}, β_{0I}' and β_{0I}'' (I is the spin of one of the two phonon states.) Detailed explanations of these parameters have

been given previously, and the reader is referred to these papers.[1,4] They all describe the strength of the quadrupole vibrations (excepting β''_{0I} with $I \neq 2$), and thus were denoted in T simply as β_2. The present program allows to assign different values to β_2, depending on between which states they appear and that is the reason why this large number of parameters is to be used. β_{02} is just the β_2 that appeared in (T-36.1), while β_{2I} is the β_2 that appeared in (T-36.2) and is now dependent on I. β'_{0I} is the β_2 that appeared in (T-37.1), while β''_{0I} is essentially the β_λ that appeared in (T-36.1), but it appears in the B-matrix element that connects the ground and the two-phonon states. Thus $\beta''_{0I} \neq 0$ means that it is assumed that a non-vanishing amplitude of the one-phonon type state is admixed into that of the two-phonon state.

Irrespective of what are the states that appear in CARDs-8, the VCOUPL(N), N = 1 ~ 10 in CARD-19 should always have the values of β_{02}, β_{20}, β'_{00}, β_{22}, β'_{02}, β_{24}, β'_{04}, β''_{00}, β''_{02}, β''_{04} in this order. It should be noted that the sign of β''_{0I} is quite important in data fitting and also of physical significance.[4]

When one of the states coupled (the last state in CARD-8) is of the three-phonon type, KEXCOM(13) is put to 1, as mentioned above (cf. CARD-3). In this case another card of the type CARD-19 must follow CARD-19, and the first two quantities that are read in from this card, i.e., VCOUPL(11) and VCOUPL(12) give, respectively, the β_2 values that were defined in (T-36.3) and (T-37.2).

<u>INTYPE = 4</u> is used exclusively for deformed nuclei as a target. There are two different ways of this use. Note that when INTYPE = 4, always KTRL(11) = 1 must be taken. (The program works even if KTRL(11) = 0 is

taken, but the obtained results may have little physical reality.)

(i) When ACC is used (i.e., when KTRL(7) = 1), always make INTYPE = 4. This is true both for even- and odd-A targets. The present shape of the program allows to take either INTMAX = 1 or 2. If INTMAX = 1 is taken, only one term of (T-15) with $\lambda = 2$ is included in the coupling. If INTMAX = 2, two terms of (T-15) with $\lambda = 2$ and 4 are included. In either case β_2 and β_4 are to be read in as VCOUPL(1) and VCOUPL(2) from CARD-19. If a pure quadrupole type deformation is to be considered, do not forget to make VCOUPL(2) = 0. (Note that β_2 and β_4 are $\beta_{\lambda'}$ appearing on the righthand side of (T-16). This λ' is quite different from λ appearing just above.)

The target states that are to be defined in CARDs-8 can appear in any order, except that the lowest and the highest spin states should appear as the first and the last states in that CARD(s). (If K = 1/2 and the ground state spin is not 1/2, a few modifications of the present program are needed.)

(ii) When the excitation only of states in the ground band is considered, just as it was in (i), but the incident energy is too low to allow the use of ACC, still INTYPE = 4 can be used. Again the target can be either of even- or odd-A, and the way to use INTMAX, VCOUPL(1) and VCOUPL(2) is exactly the same as in (i).

There is, however, a special use of INTYPE = 4, when the target is an odd-A nucleus, and when the spin-spin interaction is considered: (Thus KTRL(19) = 1.) As in ref. 6 this spin-spin interaction is taken as $(\sigma \cdot I)V_{ss}$ times the Saxon form factor.

When this term is considered, the use of VCOUPL(1) and VCOUPL(2) is

still the same as above, but one has to make INTMAX = 3. Also one has to give the value of V_{ss} in MeV in CARD-19 as VCOUPL(3). If KTRL(19) = 1, do not make KTRL(12) = 1.

The rule for the order of appearance of the target states in CARDs-8 is also the same as in (i), irrespective of whether KTRL(19) = 0 or 1.

INTYPE = 5 is used when the target is an odd-A, vibrational (spherical) nucleus. If the core excitation is restricted to (either quadrupole or octupole) one-phonon excitation, cases with $I_1 \leq 9/2$ can be handled. (The computer knows which of the quadrupole or octupole vibrations is excited by checking whether |KPRITR(2)-KPRITR(1)| = 0 or 1 (see BMATRX, SN-515). Therefore, no input data is needed to specify this.) In this case INTMAX = 1.

In CARD-8 the ground state should appear as the first state. The order of appearance in CARDs-8 of the excited states is immaterial. However, the β_2 values for the coupling between the ground state and the N-th states is to be given as VCOUPL(N-1) in CARD-19. See BMATRX, SN-520, and note that the B-matrix is given by:

$$\langle I_n \| Q_2^{(1)} \| gr_o \rangle = (-)^{I_n - I_1} \beta_2 I_n / \sqrt{5}.$$

When $I_1 = 1/2$, the present program allows one to consider excitation of two-quadrupole-phonon core states, as well as the one-quadrupole-phonon core states. The following explanation is for this case. For the details of the formulation, see ref. 5 .

When the core is excited to one-phonon state with core spin equal to 2, there can be two states with the total spin of 3/2 or 5/2. When two-phonons are excited in the core, its spin can be either I_c = 0, 2 or

4, and the total spin can be either I = 1/2, 3/2, 5/2, 7/2 or 9/2. Note that if once the value of I is given, the value of I_c is fixed uniquely. The present program allows one to consider the excitation of only one state with a two-phonon core at one time, together with the three states with zero and one-phonon cores.

Thus with IICPLE = 4, we consider the 1/2-3/2-5/2-I coupling, where I is one of the above five states with a two-phonon core. In this case the value of the corresponding I_c must be given as KEXCOM(12) in CARD-3. Next make INTMAX = 2. The value of β_2 which is common to one-phonon-cored 3/2 and 5/2 states is read in as VCOUPL(1). The values of the parameters that correspond to β_{2I_c} and β'_{0I_c}, defined in the explanation of INTYPE = 3, are to be read in as VCOUPL(2) and VCOUPL(3), respectively. For the states with total spin I = 3/2 or 5/2, the program allows one to consider them to be a mixture of the states with one- and two-phonon cores with I_c = 2. Thus, let the wave functions of the first and second 3/2 states be written as ψ_1 and ψ_2, respectively. Correspondingly, let the wave functions of 3/2 states with pure one- and two-phonon cores be written as $|1\rangle$ and $|2\rangle$, respectively. Then one may be able to write ψ_1 and ψ_2 as

$$\psi_1 = a|1\rangle \pm c\sqrt{1-a^2}\,|2\rangle + \ldots\ldots,$$

$$\psi_2 = -a|2\rangle \pm c\sqrt{1-a^2}\,|1\rangle + \ldots\ldots.$$

where the coefficient c with $|c| \leq 1.0$ is inserted to take into account the possibility that $|1\rangle$ and $|2\rangle$ do not exhaust ψ_1 and ψ_2. This c is read in as VCOUPL(4). The value of a is read in as VCOUPL(7), while one of the double signs in the above expressions is read in as VCOUPL(8) = ± 1.

When the total spin under consideration is 5/2, the corresponding quantities are read in from VCOUPL(9) and VCOUPL(10), respectively. Note that VCOUPL(7) and VCOUPL(9) should never be less than $\sqrt{1/2}$ and never exceed 1.

The order of appearance of states in CARD(s)-8 should be of the order of increasing energy.

INTYPE = 6 is used when the target is an even-A deformed nucleus, as with INTYPE = 4, but the excitations of vibrational bands are to be considered. The list of the seven bands that can be considered in the present program is found in the explanation of CARD-7-1. As was stated there, the numbers of states in these bands are read in from CARD-7-1 as KEXCOM(21) through KEXCOM(27). The order with which these states appear in CARD(s)-8 is to follow the same order of the numbering made in the explanation of CARD-7-1. Namely there should come states in the ground band first, then those in the β-band (if any), and then those in the $K = 0^-$ band (if any), and so on. Clearly KEXCOM(21) must be non-zero always and $\sum_{N=21}^{27}$ KEXCOM(N) $\leqq 6$. The value of INTMAX does not need to be given or can be anything, since it is set automatically by the computer.

The value of β_2 of Eq. (A-2) in Appendix A of this report is to be read in as VCOUPL(1) from CARD-19. The parameters that describe the strength of the excitations of the above six excited bands, namely η_i ($i = 1 \sim 6$) of Eq. (A-8) are to be read in as VCOUPL(3) through VCOUPL(8), respectively, from CARD-19.

In column 72 of the CARD-19 a quantity called KREAD1 = 1, 2 or 3 is read in. The role of this quantity is clear from SN-690 + 2. Namely this quantity specifies, when more than one set of calculations are

intended to be made at one time, what sort of CARD should follow the CARD-19 of the preceding calculation. For example, if KREAD1 = 3, the CARD-17 ~ 19 are the cards that follow the previous CARD-19. In this case two (or more) calculations can be made, with the same coupling scheme, but with different optical model and/or coupling parameters. On the other hand, if KREAD1 = 1, a calculation which is completely independent from the preceding run can be made.

This concludes the explanation of all the input cards. Before leaving this subsection, however, we remark once again that, if the reader wants to get complete understanding of how the quantities read in from CARDS 7, 8 and 19 are indeed used in the program, he must read the listings of the subroutines BMATRX, AMATRX and COUPLE, in particular the first one. As is seen in the very beginning (SN-110-1) of BMATRX, there is a computed GO TO statement, which leads the flow to different groups of statements that compute appropriate B-matrices depending on the value of INTYPE. For a given value of INTYPE, it is sufficient to read the part of BMATRX which is headed by a card with a statement number that equals 100 x INTYPE + 10, and ends with a card that says GO TO 1000.

SN-500 ~ 690 handles the output of the quantities thus far read in. The contents are quite straightforward and no detailed explanation seems needed.

The last role of JP1MAIN is to call CCCTRL (SN-690+1), and the coupled-channel calculation for a given set of parameters now starts.

2) CCCTRL

The name CCCTRL means CC-control, and the purpose of this

subroutine is to prepare various basic quantities that are used in the CC-calculations that follow.

In the very beginning (up to SN-150) IIMULT(N), ECM(M), CE(N), WN(N), WNINI(N) and SGMAZZ(N) are evaluated for N = 1 ~ IICPLE (1 ~ IIXCAL for ACC) channels. They are, respectively, $2I_N + 1$, c.m. energy, η_N (Coulomb parameters), k_N, k_{iN} (the wave number at $r \approx 0$) and $\sigma_o(N)$ (Coulomb phase shift for $\ell = 0$).

At SN-150+1 ~ 167, XMES1 is re-evaluated so that XMES1 = XMES2/2^N with N an integer. (Thus XMES1 in CARD-9 was not necessary to satisfy the relation XMES1 = XMES2/2^N exactly.) This part also fixes the value of NDFMES (meaning number of different meshes). See Appendix B to understand the meaning of these quantities.

Next at SN-167+1 ~ SN-185+2 XMAX, NXMAX, NXCPLE and JJMAX are evaluated, just in the way as was explained in the explanation of CARD-3.

At SN-273+2 ~ SN-290 a quantity called KBOUND is evaluated. KBOUND = 0 if all ECM(N) > 0, but KBOUND = 1 if at least one of ECM(N) < 0. The quantity NFGMX, defined at SN-290+1, equals 1 or 5, depending on KBOUND = 0 or 1.

At SN-290+2 ~ SN-390, the Coulomb wave functions are computed. When KBOUND = 0, where all ECM(N) > 0, the subroutine FLGLCH is called (SN-341) and the functions F_ℓ, G_ℓ, F'_ℓ and G'_ℓ are evaluated for RHOMX = $\rho = k_N R_m$. The results are stored in FC, GC, FDC and GDC (SN-355).

On the other hand, when KBOUND = 1, again FLGLCH is called if ECM(N) > 0, but FLGLNG is called if ECM(N) < 0. The evaluation of the Coulomb wave functions are made at five mesh points, i.e., at $X = R_m +$ n x XMES2 with n = 2, 1, 0, -1 and -2. The reason why this has to be

made is understood from Appendices B and C, where the techniques to solve the differential equations and to evaluate the scattering amplitudes are explained. The Coulomb functions obtained are stored as FCRA, GCRA, FDCRA and GDCRA as is seen at SN-365. Note for ECM(N) < 0, what are computed are, in fact, the spherical Hankel functions and not the Whittaker functions, since we consider the case ECM(N) < 0 only when the projectile is a neutron. Note also that in FLGLNG these spherical Hankel functions are stored as G and GD; F and FD are set equal to zero.

The quantities $\exp(i\sigma_\ell^{(n)})$ are evaluated and are stored in EXSGRI; (SN-273+1 ~ SN-510). At SN-580 POTENT is called to evaluate the radial part of various potentials and then at SN-605 the double DO-loops on JJ and K begin. The meaning of these double DO-loops were already explained (cf. CARD-3). In this DO-loop two subroutines NLJJJK and COUPLE are called. The explanation of the former is given in the next subsection. COUPLE is explained somewhat later.

2-1) NLJJJK

This is a tiny but important routine. As is seen in T, if KTRL(7) = 0, the size of the coupled equation N_c of (T-20) and the quantum numbers $(n\ell j)$ of (T-25) can be fixed if once the set of quantities $I_1, \ldots, I_N$, and J and Π of (T-17) are given. The routine NLJJJK is used just to evaluate these quantities. As is seen J and Π are communicated as JJ and K, respectively, to NLJJJK through KEXCOM(45) and (46), and then the evaluation is straightforward. The storage of the results is made as follows:

$$\text{MXROW} = N_c$$

$$\text{NNROW}(N) = n,\ \text{LLROW}(N) = 2\ell,\ \text{JJROW}(N) = 2j \text{ for } N = 1 \sim N_c.$$

Note that ℓ that exceeds KEXCOM(14)-1 is omitted (SN-410).

How this routine works when KTRL(7) = 1 will be almost evident by knowing the definition of JJ and K, as made in the explanation of KTRL(9), and will require no further explanation.

3) FLGLCH

This subroutine computes for ECM(N) > 0 the Coulomb (or spherical Bessel and Neuman) functions and their derivatives for a given value of $\rho = k_n r_m$ (= RHOMX in the program). The coding follows very closely the method given by Buck et al.[12] and by reading this reference the understanding of FLGLCH is quite straightforward. F_ℓ, G_ℓ, F'_ℓ and G'_ℓ are stored in F(L), G(L), FD(L) and GD(L) with $L = \ell + 1$ for $L = 1 \sim$ LMAX.

3-1) FLGLNG

As was explained in CCCTRL, this subroutine evaluates the spherical Hankel functions. This subroutine is very short and the content is simple. Therefore no further explanation will be necessary.

4) POTENT

This routine prepares the radial parts (form factors) of the optical potentials and the coupling potentials. The coding is quite straightforward.

At the very beginning of this routine evaluation of several variables is made. Their meaning is clear so as not to warrant any explanation except that ALFCλ, with λ = 0, 2, 4 and 6 equals $[4\pi(2\lambda + 1)]^{\frac{1}{2}}$.

At SN-258 $\sim$ 264 evaluation is made of how many mesh points are to be considered for each of the given mesh values (= DX), cf. Appendix

B for the explanation. After that the evaluation of the form factors at each mesh point is made in SN-265 ~ 400.

If KTRL(11) = 0, all the needed form factors are evaluated in SN-265+2 ~ SN-285-1. If KTRL(11) = 1, the re-evaluation of them are made at SN-285-400. Note that VSPIN(NX) and VCOULM(NX) are not re-evaluated here; therefore, SN-265+2 ~ SN-285-1 should not be skipped for KTRL(11) = 1.

The meaning of the evaluated form factors is quite easy to understand. Thus for KTRL(11) = 0, VCENTR(NX), VCENTI(NX), VSPIN(NX) and VCOULM(NX) added together is exactly equal to V_{diag} defined in (T-12). On the other hand VCPLnR(NX) + i VCPLnI(NX) with n = 1 and 2 equals the first two lines of the right hand side of (T-13.1) and (T-13.2), respectively. (The Coulomb part of (T-13.1) and (T-13.2) are evaluated, if KTRL(13) = 1, in COUPLE at the process of integrating the differential equations; therefore, they don't need to be stored.)

For KTRL(11) = 1, VCENTR(NX) + i VCENTI(NX) and VCPLnR(NX) + i VCPLnI(NX) (with n = 1 and 2) are equal to $v_{cp}^{(\lambda)(r)}(r)$, defined in (T-16), with λ = 0, 2 and 4, respectively. On the other hand VCPDnR(NX) + i VCPDnI(NX) with n = 1, 2 and 3, respectively equals $v_{cp}^{\prime(\bar{\lambda})(r)}(r)$ (defined in Appendix A), with $\bar{\lambda}$ = 0, 2 and 4.

5) COUPLE, BMATRX and AMATRX

Since these three routines are intimately related with each other, it will be better to explain them together, rather than separately.

As we have seen in the explanations of CARDs-7, 8 and 19, there are various ways to use these routines, depending on the value of INTYPE.

Since, if one understands their use, however, for a given value of INTYPE, the understanding for the use of other values of INTYPE is quite easy. Therefore, we shall give here only the explanation for the case of INTYPE = 2.

First of all, in COUPLE the parts SN-12-3 ~ SN-815 are executed irrespective of the value of INTYPE. The meaning of KBOUND (same as in CCCTRL), NBOUND and MXRØWI is clear. NXRM (SN-20+1) defines the matching radius for KBOUND = 1. The coefficients PRE41, ..., PRE45 are just those appearing in Eq. (B.3) of Appendix B. KCTY2 = 0 or 1; its use will be found at SN-1412 and SN-1512. The part SN-70 ~ SN-670 is executed only when KTRL(13) = 1, and prepares for the Coulomb excitation calculation. SPFACT(M1) evaluated at SN-815 is just the expectation value of the operator $(\vec{\sigma}\cdot\vec{\ell})$ for each partial wave channel M1.

At SN-815+1 the subroutine BMATRX is called. We shall therefore look at this subroutine.

In the beginning of BMATRX a quantity JJTRTW is evaluated. It is just 2J where J was defined in (T-17). SN-50-1 ~ SN-60 simply clears all the B and A matrices. The B-matrix, which is simply called as B here, has three dimensions, B = B(N1, N2, K1). N1 and N2 specify the row and column of this matrix; thus they correspond to I and I′ of $\langle I\|Q_\lambda^{(t)}\|I'\rangle$ of T-27 . On the other hand, K1 specifies different sets of λ and t.

COMFAC that appear at SN - 70 equals $1/\sqrt{4\pi}$. (Note that the B-matrix defined in BMATRX is $1/\sqrt{4\pi}$ times the B-matrix of T.) At SN-70 KII(1) is set equal to IIREAD(2). KII(1) stands for λ of (T-36.1) and thus the relation KII(1) = IIREAD(2) appears because of the factor $\delta_{\lambda I}$

in (T-36.1). SN-70+1 is nothing but the equation (T-36.1).

Since we have INTYPE = 2, we next go to SN-210. SN-210 and SN-210+1 simply prepares for considering 0^+-2^+-3^- coupling if IICPLE = 3. Now suppose we have INTMAX = 1 or 2. Then we go to SN-1000 and AMATRX is called.

In SN-570 ~ SN-900 of AMATRX the A-matrices (which is the product of the A- and B-matrices in the notation of T) are evaluated. Let us assume that we have INTMAX = 1 (with INTYPE = 2). Then we first go to SN-713. Since we have B(1,2,1) $\neq$ 0 (unless VCOUPL(1) = 0) we then go to SN-720 and have KI = λ; (λ is of (T-36.1)). At SN-725 KISELC is set equal to 1, and we go to SN-750, where the evaluation of the A-matrix of (T-25) is started and it is multiplied into B(1,2,1) at SN-752-2, the result being T1. (Note the use of KII(K1) in the process of the evaluation of the A-matrix.) Since KISELC = 1 at SN-752, we next go to SN-761 and the above T1 is stored as one of the elements of the matrix AMAT1 (M1,M2). This concludes the evaluation of the A-matrix for 0^+-I^π (I^π = 1^-, 2^+, 3^- ...) coupling.

Next suppose we have 0^+-2^+-3^- coupling, and thus had INTMAX = 2. As we have seen at SN-210+1 of BMATRX, B(1,3,1) $\neq$ 0 unless VCOUPL(2) = 0. Since INTMAX = 2, SN-720 of AMATRX is passed twice, once with K1 = 1, then with K1 = 2. In both cases, however, KISELC = 1, as is seen from SN-725 and SN-725+1, and thus the result is again stored in AMAT1 (M1,M2) at SN-761.

Thus in this case the coupling term (T-27) can be rewritten as

$$\langle \ell j\ I | V_{coupl} | \ell j' I' \rangle = v^{(1)}(r) \sum_{\lambda=2,3} \langle I \| Q_\lambda^{(1)} \| I' \rangle\ A(\ell j\ I,\ \ell' j' I'; \lambda J_s)$$

$$= v^{(1)}(r)\ \ \text{AMAT1 (M1,M2)}.$$

The merit of grouping several matrices, which are to be multiplied in (T-27) onto a same form factor, into one matrix is understood by looking at Appendix B. As is seen there, the integration of the coupled differential equations is nothing but a repetition of the multiplication of matrices onto a vector. This calculation can certainly be performed faster for a smaller number of independent matrices. As is seen in SN-752 of AMATRX, KISELC is used in performing this grouping of the matrices.

Let us now consider the case of INTMAX = 5, still with INTYPE = 2. In this case the coupling considered is either 0^+-2^+-3^- or 0^+-2^+-3^--I^- coupling. In SN-232 ~ 240 the B-matrix elements of (T-36.4), (T-36.5), (T-37.3) and (T-37.4) are computed and stored as B(N1,N2,K1) with K1 = 1 ~ 5. Among them, those with K1 = 1 and 2 are multiplied in (T-27) by $v^{(1)}(r)$ of (T-13.1), and as is seen in AMATRX, the corresponding A-matrix elements are stored in AMAT1 (M1,M2). On ther other hand those with K1 = 3, 4 and 5 are to be multiplied in (T-27) by $v^{(2)}(r)$ of (T-13.1). As is seen in SN-726, KISELC = 2 for these cases and the corresponding A-matrix elements are stored in AMAT2 (M1,M2). (Cf. SN-752 and SN-766 of AMATRX.) Thus in this case we have two A-matrices.

The evaluation of the A-matrices differs depending on the values of INTYPE and INTMAX, but the above illustration will be sufficient to get

a general idea how it is made. Anyway, when the command is returned from BMATRX to COUPLE at SN-815+1, it is understood that all the necessary A-matrices were evaluated, and thus we can proceed to solving the coupled equations.

At SN-815+3 a DO-loop starts with a variable INEX. INEX = 1 and 2 corresponds to the outward and inward solutions of Appendix C. The variable M3 for the DO-loop at SN-815+5 discriminates various initial conditions, and in SN-815+7 ~ SN-1225-1 detailed preparations of the initial conditions are set. Actual integration of the differential equations starts at SN-1225 and the process for one step of mesh ends at SN-1800.

In between SN-1225 and SN-1800 various matrices are multiplied onto a vector (whose elements are called UCR2(M1) + i UCI2(M1)) and the results are summed up as BR(M1) + i BI(M1). Since we have prepared various A-matrices, and know with which form factors they are to be multiplied, the coding in this part is to be made according to this knowledge.

We thus consider once again the INTYPE = 2 case. We know that if INTMAX $\leq$ 2 only one A-matrix is to be considered, while two A-matrices have to be considered if INTMAX $\geq$ 3. As is seen at SN-1415, these two kinds of multiplications are made either at SN-1470 ~ 1473 or at SN-1475 ~ 1477.

Irrespective of the values of INTYPE and INTMAX, SN-1410 ~ 1492 performs the integration for RFF, while SN-1510-1592 does it for CFF. SN-1610 ~ 1700 is executed if KTRL(13) = 1.

The part of SN-4190+1 ~ 1800 is to perform the calculation of (B-3) in Appendix B. At SN-1805 ~ 1850 the derivatives of the solutions at the matching radius are evaluated and are stored as UCR2(M1) + i UCI2(M1).

They are written on TAPE-8 together with the values of the solutions UCR1(M1) + i UCI1(M1) at the matching radius (see SN-1897 for KBOUND = 0, or SN-1892 ~ 1895+1 for KBOUND = 1).

6) SMATRX

If once the content of Appendix C is understood, the understanding of this subroutine is quite simple. We simply say that the simultaneous equation (C-6) for KBOUND = 0 or (C-10) for KBOUND = 1 is prepared at SN-119+1 ~ 179+2. This is solved by calling SIMLEQ at SN-300.

For KBOUND = 1, the C-matrix elements are obtained already at this stage as the solution of this simultaneous equation. On the other hand for KBOUND = 0, the C-matrix elements have to be calculated by using Eq. (C-7). This is performed at SN-340 ~ 455.

In both cases the obtained C-matrices are saved on TAPE-7 at SN-490.

6-1) SIMLEQ

This is a subroutine to solve a set of simultaneous equations. A standard technique to bring the coefficient matrix into a triangular form is used. Complex arithmetic is used throughout.

7-8) XSEC, ACCSAB, CROSPL, CMMMFC and BFCTOR

These five subroutines are used in evaluating various cross sections after the C-matrix elements were obtained. Since these routines are correlated very intimately with each other, it is more convenient to give their explanations together, rather than do it separately.

Usually the coding for the evaluation of the cross sections is

simpler and easier than that of the C-matrix elements, because it is essentially a matter of bookkeeping. In the present program, however, cases in which a huge number of the C-matrix elements appear are allowed and thus these elements have to be saved in TAPE, rather than in the core. This fact makes the performance of the many-fold summations that appear in the expressions of the cross sections rather complicated.

We start the explanation with that of the differential scattering cross sections $\sigma_n^{(s)}(\theta,\varphi)$ for NACC, i.e., with KTRL(7) = 0. Its analytic expression is found in Eqs. (T-48) and (T-49), and we note that the hardest part of its evaluation is that of the second term of (T-48), which we shall call $\bar{X}^{(NACC)}_{m_s M_1;m_s' M_n}(\theta,\varphi)$.

Let us first define quantities

$$C^{JM}_{\ell j;n\ell' j';m_s M_1} = (\ell s o m_s|jm_s)(jI_1 m_s M_1|JM)\, C^{J}_{\ell j;n\ell' j'}\, \delta_{M,m_s+M_1} \tag{7.1}$$

Then we see immediately that

$$\bar{X}^{(NACC)}_{m_s M_1;m_s' M_n}(\theta,\varphi) = \sum_{\ell j \ell' j' J m_\ell' m_j' M} (4\pi/k_n^2)^{\frac{1}{2}}\, \hat{\ell}'\, \exp(2i\sigma_{\ell'}^{(n)})\, C^{JM}_{\ell j;n\ell' j';m_s M_1}$$
$$\times\, (\ell' s' m_s' m_\ell'|j' m_j')(j' I_n\, m_j' M_n|JM)\, Y_{\ell' m_\ell'}(\theta,\varphi). \tag{7.2}$$

We next note the relation

$$Y_{\ell' m_\ell'}(\theta,\varphi) = \left[\frac{2\ell'+1}{4\pi}\right]^{\frac{1}{2}} (-)^{\frac{m_\ell'+|m_\ell'|}{2}} \left[\frac{(\ell'-|m_\ell'|)!}{(\ell'+|m_\ell'|)!}\right]^{\frac{1}{2}} P_{\ell'|m_\ell'|}(\cos\theta)\exp[im_\ell'\varphi] \tag{7.3}$$

and define

$$C^{\ell'(m'_\ell)}_{m_s M_1;m'_s M_n} = \sum_{\ell j j' J m' j M} \frac{1}{k_n} (2\ell'+1)\exp(2i\sigma_{\ell'}^{(n)}) C^{JM}_{\ell j;n\ell j;m_s M_1}$$

$$\times (-)^{\frac{m'_\ell+|m'_\ell|}{2}} \left[\frac{(\ell'-|m'_\ell|)!}{(\ell'+|m'_\ell|)!}\right]^{\frac{1}{2}} (j' I_n m'_j M_n | JM)(\ell' s' m'_\ell m'_s | j' m'_j)\, \delta_{m'_\ell;m_s+M_1-m'_s-M_n}. \tag{7.4}$$

Then clearly

$$\bar{X}^{(NACC)}_{m_s M_1;m'_s M_n}(\theta,\varphi) = \sum_{\ell'(m'_\ell)} C^{\ell'(m'_\ell)}_{m_s M_1;m'_s M_n} \exp(im'_\ell\varphi)\, P_{\ell'|m'_\ell|}(\cos\theta). \tag{7.5}$$

In the subroutine XSEC, $C^{J}_{\ell j;n\ell' j'}$ is read in from TAPE 7 (SN-110-1), converted into $C^{JM}_{\ell j;n\ell' j';m_s M_1}$ in SN-160 ~ 240 and the latter is stored on TAPE 8 (SN-250). Then (if KTRL(3) = 0) XSEC calls CROSPL (SN-955), and the rest of the calculations are performed.

CROSPL first calls CMMMFC for each given value of n, (SN-4190 + 5) and in the latter $C^{JM}_{\ell j;n\ell' j';m_s M_1}$ are first read in from TAPE 8 (SN-425 + 4). They are multiplied by $\exp(2i\sigma_{\ell'}^{(n)})$ and the results are stored as CMXR + iCMXI (SN-441 ~ 443). The multiplication of other factors in (7.4) and the summations appearing there are performed in SN-610 ~ 1010, resulting in $C^{\ell'(m_\ell)}_{m_s M_1;m'_s M_n}$, which are written as CMMMKR + iCMMMKI (K = 1 ~ 5), and are stored in TAPE 7 at SN-1110 ~ 1118. It should be noted that the calculation so far has been independent on the angles θ and φ.

CROSPL then divides the angles into groups of not more than seven angles, and for each set of these angles it calls BFCTOR (SN-4190 + 10). In BFCTOR first $P_{\ell'|m'_\ell|}(\cos\theta)$ are computed (SN-1410 + 1 ~ 1450), and then $C^{\ell'(m'_\ell)}_{m_s M_1;m'_s M_n}$ are read in from TAPE 8. Results of the multiplication

are saved as BFACKR + iBFACKI (SN-1471 ~ 1490).

The command is then returned to CROSPL and there the multiplication of $C^{\ell'(m'_\ell)}_{m_s M_1; m'_s M_n}$ by $\exp(im'_\ell\varphi)$, $a^{(i)}_{m_s}$ and $b^{(i')}_{M_1}$ are performed in SN-1045 ~ 1350; i.e., XAMPKR + XAMPKI (K = 1 ~ 3) of SN-1161 ~ 1165 are $\bar{X}^{(NACC)}_{m_s M_1; m'_s M_n}(\theta,\varphi) a^{(i)}_{m_s} b^{(i')}_{M_1}$. The addition of the contribution to these quantities from the first term of (T-48) is made at SN-1390 ~ 1405. The rest of CROSPL is devoted to the evaluation of $\sigma_n^{(s)}(\theta,\varphi)$ and the polarization $P_n(\theta,\varphi)$ of (T-50). The understanding of this part is very easy.

We shall now turn to the evaluation of the differential cross section and the polarization in the case of ACC, i.e., for KTRL(7) = 1, which requires the evaluation of (T-71). In evaluating $X^{(ACC)}_{m_s M_1; m'_s M_n}(\theta,\varphi)$ of (T-71), we first define the quantities

$$C^{J\ell' j'}_{m_s m'_j M_J} = \sum_{\ell j} \sum_{\bar{m}_j \geq 0} (-)^{\bar{m}_j - m'_j} (\ell s o m_s | j m_s)(j j' \bar{m}_j - \bar{m}_j | J 0)(j j' m_s - m'_j | J M_J)\, C^{\bar{m}_j}_{\ell j; \ell' j'} \times \frac{1+(-)^J}{2(1+\delta_{\bar{m}_j, 0})} . \tag{7.6}$$

Then the second term of (T-71), which we call $\bar{X}^{(ACC)}_{m_s M_1; m'_s M_n}(\theta,\varphi)$ is rewritten as

$$\bar{X}^{(ACC)}_{m_s M_1; m'_s M_n}(\theta,\varphi) = \sum_{\ell' j' m'_\ell m'_j J M_J} 2(4\pi)^{\frac{1}{2}} k_1^{-1} \hat{\ell}' \hat{I}_1 \hat{I}_n^{-1} \exp[2i\sigma^{(1)}_{\ell'}] C^{J\ell' j'}_{m_s m'_j M_J} \times (J I_1 O K | I_n K)(J I_1 M_J M_1 | I_n M_n)(\ell' s m'_\ell m'_s | j' m'_j)\, Y_{\ell' m'_\ell}(\theta,\varphi) . \tag{7.7}$$

We notice the relation (7.3), and then define the quantities

$$C^{I_1 I_N \ell'(m'_\ell)}_{m_s M_1; m'_s M_n} = \sum_{J M_J j' m'_j} \left\{ \frac{2}{k_1} \frac{\hat{I}'_1}{\hat{I}'_N} \exp[2i\sigma^{(1)}_{\ell'}] C^{J\ell' j'}_{m_s m'_j M_J} (J I_1 0 K | I_n K)(J I_1 M_J M_1 | I_n M_n) \right\}$$

$$(2\ell'+1)(\ell' s m'_\ell m'_s | j' m'_j) \; (-)^{\frac{m'_\ell + |m'_\ell|}{2}} \left[\frac{(\ell' - |m'_\ell|)!}{(\ell' + |m'_\ell|)!} \right]^{\frac{1}{2}} \tag{7.8}$$

Then (7.7) is rewritten into the following very compact form

$$\bar{X}^{(ACC)}_{m_s M_1; m'_s M_n}(\theta, \varphi) = \sum_{\ell'} C^{I_1 I_n \ell'(m'_\ell)}_{m_s M_1; m'_s M_n} \exp[i m'_\ell \varphi] P_{\ell' |m'_\ell|}(\cos\theta) \tag{7.9}$$

Equation (7.9) has the same structure as that of Eq. (7.5).

When KTRL(7) = 1, the subroutine XSEC does nothing but to call ACCSAB. In ACCSAB, $C^{\bar{m}_j}_{\ell j; \ell' j'}$ are read in from TAPE 7 as CMXADR + iCMXADI (SN-1175), and then the summation of Eq. (7.6) is performed between SN-1173 and SN-1280. The quantities $C^{J\ell' j'}_{m_s m_j M_J}$ are temporarily stored as CMSXJR + iCMXSJI (cf. SN-1280-1 and -1), and then saved in TAPE 8 at SN-1390+1.

XSEC than calls CROSPL, which for each given value of n calls subroutine CMMMFC at SN-4190 + 5 just the same way as was the case with KTRL(7) = 0. CMMMFC then reads from TAPE 8 the quantities CMXSJR + iCMXSJI (SN-515 + 1) and then starts to perform the summations that appear in Eq. (7.8). The summation is completed at SN-1010, and the resulting quantities $C^{I_1 I_N \ell'}_{m_s M_1; m'_s M_n, m'_\ell}$ which are called CMMMKR + iCMMMKI (K = 1 ~ 5) are written on TAPE 7 as is seen in SN-1110 ~ SN-1118.

After this, the evaluation of the cross sections follows exactly the same procedure as with KTRL(7) = 0, and thus its explanation can be avoided.

The evaluation of the total and reaction cross sections and the s- and p-strength functions are made in XSEC if KTRL(7) = 0, and in ACCSAB if KTRL(7) = 1. How the evaluation is made can be seen by looking at the statements that are executed when KTRL(3) $\neq$ 0, in these two subroutines.

Since the meaning of various quantities that appear in these routines have been explained above, and since the necessary formulas are found in previous publications ((T-53), (T-54) and Eq. (7) of ref. 13), the reader will find no difficulty in understanding how the coding was made.

9) OUTPUT, PLOTER

These two routines are very simple; no explanation is needed, except to note that in PLOTER the cross sections are plotted with logarithmic scale, while the polarizations are plotted with linear scale.

10) RAC7, CLEB

These two routines have been explained in detail.[8]

Appendix A

In this appendix the coupling terms in the coupled-channel equations that appear when the excitation of states that belong to the vibrational bands of even, deformed nuclei are considered.

For this purpose we recall that when only the excitation of states in the ground band was considered, the radius of the nucleus was simply described as

$$R = R_o(1 + \sum_\lambda \beta_\lambda \; Y'_{\lambda o}), \tag{A-1}$$

as is seen in (T-3), (cf. also Sec. II (ii) of T).

We now extend (A-1) as

$$R = R_o(1 + \sum_\lambda \beta_\lambda \; Y'_{\lambda o} + \sum_{i=1}^{2} \epsilon_{\lambda_i o} Y'_{\lambda_i o} + \sum_{i=3}^{6} \frac{1}{\sqrt{2}} (\epsilon_{\lambda_i K_i} Y'_{\lambda_i K_i} + \epsilon_{\lambda_i -K_i} Y'_{\lambda_i -K_i}) \tag{A-2}$$

where $i = 1 \sim 6$ stands for $K = 0^+$, 0^-, 2^+, 1^-, 2^- and 3^- vibrations, that were defined in the explanation of the CARD-7-1. The $\epsilon_{\lambda_i K_i}$ are operators which are equal to some constant times a linear combination of creation and annihilation operators of the phonons corresponding to the various vibrations. Thus, e.g., for $K = 0^+$, $\epsilon_{20} = \beta - \beta_2$, while for $K = 2^+$, $\epsilon_{22} = \gamma \; \beta_2$, etc.

We now insert (A-2) into the optical potential (T-1), expand the latter in powers of the last term of (A-2) (up to first order) and then expand the coefficients in Legendre polynomials, in a manner similar to what was done in Sec. II (ii) of T.

The resulting expression may be written as

$$V(r,\theta,\varphi) = V_{diag} + V_{coupl}^{(r)} + V_{coupl}^{(r)'}, \tag{A-3}$$

where the first two terms are the same as the two terms on the right hand side of (T-14). On the other hand the last term of (A-3) may be written as

$$V_{coupl}^{(r)'} = \sum_{\lambda}\sum_{\bar{\lambda}} v_{cp}'^{(\bar{\lambda})(r)}(r) \left\{ \sum_{i=1}^{2} \epsilon_{\lambda_i o} C_{\lambda\bar{\lambda}}^{(i)} D_{\mu o}^{(\lambda)} + \sum_{i=3}^{6} \frac{1}{\sqrt{2}} (\epsilon_{\lambda_i K_i} C_{\lambda\bar{\lambda}_{K_i}}^{(i)} D_{\mu K_i}^{(\lambda)} + \epsilon_{\lambda_i -K_i} C_{\lambda\bar{\lambda}-K_i}^{(i)} D_{\mu -K_i}^{(\lambda)}) \right\} Y_{\lambda\mu}(\theta,\varphi) \tag{A-4}$$

with

$$C_{\lambda\bar{\lambda}}^{(i)} = \left[\frac{(2\lambda i+1)(2\bar{\lambda}+1)}{4\pi(2\lambda+1)}\right]^{\frac{1}{2}} (\lambda_i\bar{\lambda}oo|\lambda o)^2; \; i = 1,2,$$

$$C_{\lambda\bar{\lambda}\pm K_i}^{(i)} = \left[\frac{(2\lambda i+1)(2\bar{\lambda}+1)}{4\pi(2\lambda+1)}\right]^{\frac{1}{2}} (\lambda_i\bar{\lambda}oo|\lambda o)(\lambda_i\bar{\lambda} \pm K_i 0|\lambda \pm K_i); \; i = 3, 6. \tag{A-5}$$

In (A-4) the form factor $v_{cp}'^{(\bar{\lambda})(r)}(r)$ is obtained from (T-16) by replacing λ there by $\bar{\lambda}$, and then replacing the expression in the curly bracket there by the first term of (T-13.1) with the radius defined by (T-3), and by finally multiplying with R_o/a.

The contribution of $V_{coupl}^{(r)}$ of (A-4) to the coupling term (T-27) was already given in (T-40). The contribution of (A-4) can be described as

$$\sum_{\lambda}\left[\sum_{\bar{\lambda}} v_{cp}'^{(\bar{\lambda})(r)}(r) \sum_{i} B^{(i)}(I,I';\lambda\bar{\lambda})\right] A(\ell jI;\ell' j' I';\lambda Js) \tag{A-6}$$

with

$$B^{i}(II';\lambda\bar{\lambda}) = \left[\frac{(2\lambda_i+1)(2\bar{\lambda}+1)}{4\pi(2\lambda+1)}\right]^{\frac{1}{2}} (\lambda_i\bar{\lambda}oo|\lambda o)(\lambda_i\bar{\lambda}\; K_i o|\lambda\; K_i)\; \hat{I}'(I'\lambda\; K_i-K_i|IO)\eta_i$$
$$(i = 1 \sim 6) \tag{A-7}$$

with $\eta_i = \langle gr|\epsilon_{\lambda_i o}|vib\rangle$ for i = 1, 2 and $\eta_i = \sqrt{2}\ \langle gr|\epsilon_{\lambda_i K_i}|vib\rangle$ for i = 3 ~ 6. In (A-7) it is assumed that the state with spin I is a member of the ground band, while the state with spin I' is a member of the i-th vibrational band.

Appendix B

Method of differential equation integration

The method used in the present program in solving the differential equations is the prediction-correction method due to Stormer.[14] This method is fast and accurate and is particularly suited when the coupled differential equations are to be solved. Stormer's method is, in short, an approximate way to obtain the solution at a mesh point x when the solution at the n preceding mesh points x-h, x-2h, ..., x-nh are known, h being the mesh size. The accuracy of the approximation is increased as the number n increases, the error being $O(h^4)$ for n = 2 and $O(h^{n+1})$ for $n \geq 4$. (No case exists with n = 3.)

In the present code a combination of n = 2 and n = 5 are used. The use of n = 2 was forced, since at the origin one cannot know the values of the solution at more than two points, if it is avoided to assume any specific shape of the solution. We know that all the solutions of our equations satisfy u(o) = o. We can then give any value to u(h), the different values given to u(h) simply meaning different normalization which is readjusted through the matching condition at a matching radius. In other words, so long as n = 2 is used, no ad hoc assumption was made, except that we assumed that there is no potential between r = o and r = h. Because of this last reason, and also of the fact that with n = 2

the error is comparatively large for a given h we should choose h fairly small near origin. Once away from origin to some extent, however, we can switch to n = 5 method. Then we can begin to use large values for h.

We now show explicitly the Stormer method for the cases of n = 2 and 5. For that purpose we note that our differential equations can be written in the following form:

$$U''(x) = A(x)U(x). \tag{B-1}$$

When (B-1) is a set of N coupled equations, U is a column vector of dimension N, while A is a N x N matrix. In the n = 2 approximation, the solution is obtained by the following relation: $(v(x) \equiv h^2A(x)u(x))$

$$U(x) = 2u(x-h)-u(x-2h) + v(x-h) \tag{B-2}$$

In the n = 5 approximation, the corresponding relation is

$$\begin{aligned} u(x) = {} & 2u(x-h) - u(x-2h) + \frac{299}{240}\, v(x-h) \\ & - \frac{11}{15}\, v(x-2h) + \frac{97}{120}\, v(x-3h) \\ & - \frac{2}{5}\, v(x-4h) + \frac{19}{240}\, v(x-5h). \end{aligned} \tag{B-3}$$

By Taylor expanding both sides of (B-2) in powers of h, the reader will easily find that the error is indeed $O(h^4)$. Similarly, Eq. (B-3) is easily shown to have an error of $O(h^6)$.

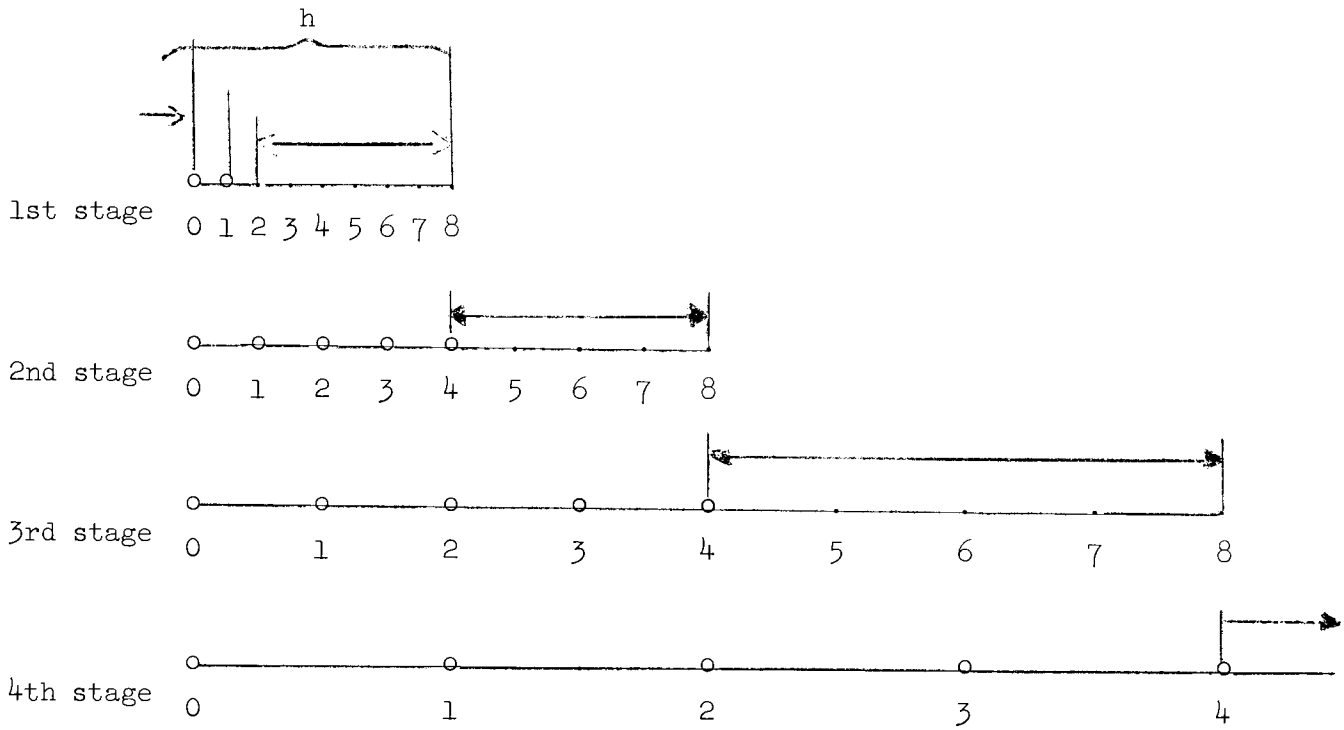
The merit of the use of the Stormer method is the following. As is seen in (B-2), there appears a multiplication of the matrix A onto the vector u at each mesh point. Thus N^2 multiplications are needed at each mesh point and this is the part which takes the most of the machine time

in our calculations. (Still this method is much faster than, e.g., the Runge-Kutta method.)

In Eq. (B-3) there are five such multiplications and thus it may seem that the use of Eq. (B-3) is five times slower than that of Eq. (B-2). Obviously it is not true, since four of the multiplications had been made prior to reaching the present mesh point x. In other words, if the vectors $v(x')$ had been saved for $x' = x-2h$, $x-3h$, $x-4h$ and $x-5h$, what has to be made in Eq. (B-3) is the 4N multiplication and 4N summations, in addition to the above N^2 multiplications. Therefore, when N is large, the use of Eq. (B-3) does not slow down the calculation very much compared with the use of Eq. (B-2). Since the former allows the use of a larger h than the latter, the calculation is, in fact, sped up.

The combined use of Eqs. (B-2) and (B-3) is made in the following way. First we choose h ($\approx$ 0.1 fm). Then we define another mesh size h_1 which is $h_1 = h/2^k$ (k = integer). We use Eq. (B-2) for the first eight steps. Then we increase h_1 by a factor of two and then switch to Eq. (B-3). With this new mesh $2h_1$, we already have u's at five preceding points, and thus Eq. (B-3) can indeed be used. With mesh $2h_1$, we proceed four more steps, and then increase the mesh to $4h_1$. We again have five preceding u's in terms of this new mesh, and we proceed four more steps. This procedure is repeated until the mesh equals h. We then keep integrating up to the matching radius.

The change of the mesh size is illustrated in the following figure for k = 3, where the arrows show the range of x between which the integration proceeds in each stage. The circles show the points of x where the values of u are already known at the beginning of each stage.



In most of the nuclear problems the use of h = 0.1 fm and k = 3 or 4 seems to give sufficiently high accuracy to the solution.

Appendix C

Method of solving the coupled equations and obtaining the C-matrices

We have a set of N coupled equations as given in (B-1) of Appendix B. We first solve these equations under the condition that the p-th component of the vector U is non-zero at the origin, but the other components are all equal to zero. We write the solution thus obtained at $r = r_m$ as U_{pq} $(q = 1 \sim N)$, where r_m is the matching radius and q specifies N different components. We repeat this procedure for $p = 1 \sim N$ and it is clear that the solutions obtained with different p are (linearly) independent with each other. We then define functions u_q at $r = r_m$ by

$$u_q = \sum_{p=1}^{N} a_p u_{pq} \quad (j = 1 \sim N) \tag{C-1}$$

which is to be matched to the asymptotic solution, defined also at $r = r_m$. The condition of smooth matching fixes the unknown coefficients a_p, as well as the scattered amplitude, or the C-matrix elements.

The asymptotic form (T-42) or (T-48) may be written, in general, as

$$d_q F_q \delta_{kq} + e_{kq} C_{kq} (G_q + iF_q) \tag{C-2}$$

where k is a fixed number among $1 \sim N$ that specifies a channel in which there is an incoming wave, while d_q and e_{kq} are known constants. Equating (C-1) and (C-2) (and their corresponding derivatives) we get the following matching equations:

$$\begin{cases} \sum_{p=1}^{N} a_p u_{pq} = d_q \delta_{kq} F_q + e_{kq} C_{kq} (G_q + iF_q) & (q = 1 \sim N) \quad \text{(C-3)} \\ \sum_{p=1}^{N} a_p u'_{pq} = d_q \delta_{kq} F'_q + e_{kq} C_{kq} (G'_q + iF'_q) & (q = 1 \sim N) \quad \text{(C-4)} \end{cases}$$

This is a set of 2N simultaneous equations for 2N unknowns, a_p and C_{kq}. However, it is not necessary to solve these 2N equations as they are. Rather, we can reduce them into a set of N silultaneous equations, as is shown now.

Multiply (C-3) and (C-4), respectively, by $(G'_q + iF'_q)$ and $(G_q + iF_q)$ and take the difference of the resulting equations. Then noting that

$$F'_q G_q - F_q G'_q = 1 \tag{C-5}$$

we easily get

$$\sum_{p=1}^{N} a_p \left\{ u_{pq}(G_q + iF_q) - u_{pq}(G_q + iF_q) \right\} = -d_q \delta_{kq} F_q \quad (q = 1 \sim N) \tag{C-6}$$

This is a set of N simultaneous equations for N unknowns a_p, and we can solve it easily.

Next multiply (C-3) and (C-4), respectively, by F'_q and F_q and take the difference. Again using (C-5) we get

$$\sum_{p=1}^{N} a_p \left\{ u'_{pq} F_q - u_{pq} F'_q \right\} = e_{kq} C_{kq} \tag{C-7}$$

Since the left hand side is known, (C-7) is nothing but the equation to express C_{kq} in terms of the known quantities, and thus our task is finished.

When the (asymptotic) energy of the projectile in some of the coupled channels is negative, the preceding matching equations cannot be used anymore, because we should not keep integrating the differential equations beyond the nuclear radius. (Note that there the integration becomes unstable, in the sense that the exponentially increasing components are mixed into the negative energy components of the solutions which are supposed to be exponentially decreasing.) The procedure to be taken then is the following.

First choose the matching radius R_m equal to the nuclear radius R_o, (say, the half-fall-off radius of the real part of the Saxon potential). The coupled equations are then integrated from $r = 0$ to $r = R_m$ in exactly the same fashion as was made above (outward solutions). Let us call the solution at $r = R_m$ as $u_{pq}^{(i)}$, where the superscript i stands for the internal solution.

Next take a radius $R_a \gg R_o$, so that at $r = R_a$ all the nuclear interactions are effectively zero. The coupled equations are then integrated N + 1 times with the following boundary conditions (inward solutions).

(i) Pick up a partial channel k among the ground state channels, cf. Eq. (C-2). Make the amplitude at $r = R_m$ in channel k equal to $d_k F_k$, and those in other channels equal to zero. The solution obtained at $r = R_m$ with this boundary condition is called $u_{kq}^{(e)}$, where the superscript e stands for the external solutions.

(ii) Pick up any channel p and make the amplitude at $r = R_m$ in this channel equal to $e_{kp}(G_p + iF_p)$, if the energy in this channel is positive, or equal to $e_{kp}H_p$, if the energy is negative. Here H_p is either the spherical Hankel function or the Whittaker function, depending on whether the projectile is charged or uncharged. Putting the amplitude in all the other channels equal to zero, integrate the equation and call the solution at $r = R_m$ as $u_{pq}^{(e)}$.

The internal solution $u_q^{(i)}$ may be written as

$$u_q^{(i)} = \sum_{p=1}^{N} a_p \, u_{pq}^{(i)} \tag{C-8}$$

just as in (C-1), while the external solution $u_q^{(e)}$ may be written as

$$u_q^{(i)} = u_{kq}^{(e)} + \sum_{p=1}^{N} C_{kp} \, u_{pq}^{(e)} \tag{C-9}$$

Equating $u_q^{(i)}$ and its derivative $u_q^{(i)'}$, respectively to $u_q^{(e)}$, and its derivative $u_q^{(e)'}$, the following matching equations are obtained

$$\begin{aligned} \sum_{p=1}^{N} a_p \, u_{pq}^{(i)} - \sum_{p=1}^{N} C_{kp} \, u_{pq}^{(e)} &= u_{kq}^{(e)}, \\ \sum_{p=1}^{N} a_p \, u_{pq}^{(i)'} - \sum_{p=1}^{N} C_{kp} \, u_{pq}^{(e)'} &= u_{kq}^{(e)'}. \end{aligned} \tag{C-10}$$

Equation (C-10) is a set of 2N simultaneous equations for the 2N unknowns a_p and C_{kp}. (Note that it cannot be reduced to a set of N simultaneous

equations, as was the case previously.) (Note also that only the solutions C_{kp} that correspond to p for which the projectile energy is positive have the meaning as the scattering amplitude, and thus are needed in evaluating the scattering cross sections.)

III. Suggestions for Possible Modifications to the Present Program

Since detailed explanations were given in the preceding sections of the present form of our program, it will be quite easy for the readers to find out particular places to modify, in case he wants to add new features or make any improvements. We shall give here a few comments which may be of help in making such modifications; they may also serve as a summary of what was presented in the preceding sections.

As we said in the Introduction, this program was coded for the CDC-1604 computer, and thus was designed so that the whole program can be loaded within the 32 K core. As was also said in the Introduction, this made some of the calculations comparatively slow. Thus if one has access to a larger computer, it may be desirable to rewrite some of the routines for increased speed. Perhaps the part for which this is the most desirable would be the routines that evaluate the cross sections; there, scratch tapes are used frequently and slow down the calculations. (In JP2 this is avoided.) It should be noted, however, that in its present shape our program has essentially no restriction on the value of ℓ_{max}, as far as the storing of various quantities are concerned. (The only source of restriction comes from the limited size of the cores allocated to store the quantities F_ℓ, G_ℓ, F'_ℓ, G'_ℓ, $\exp(i\sigma_\ell^{(n)})$ and $\log(\ell!)$. To enlarge their capacity does not require so many cores.) This unlimited

value of ℓ_{max} was brought about because of the frequent use of the tapes.

The reader might have noticed in the preceding sections that there are various restrictions on the possible numbers of coupled channels (target states and the partial waves), total number of mesh points, etc. Clearly most of these restrictions can be removed easily by simply enlarging the size of core storages allocated to various variables. However, for most of the practical calculations, the limitations incurred by the present program are not so severe.

So far as the mathematical technique is concerned, there is room to modify the subroutine COUPLE for speeding up the solution of the coupled differential equations. A suggested method to be used is a sort of relaxation method. When the number of coupled equations is large, perhaps this method is faster than the straightforward method used in the present program (still maintaining the same accuracy).

As far as the physics is concerned, a large modification that is suggested would be to introduce the microscopic description of the wave functions of the target nucleus, in place of the phenomenological description used here (and in T). Even with this modification, however, most parts of the present program may be used just as they are. The modification required is to replace POTENT by a new subroutine, so that the form factors are evaluated from the microscopic model assumed for the target. (Or modify POTENT so that these form factors are read in from cards or tapes.) This procedure may make the form factors different for the coupling between, e.g., the 0^+ and 2^+ states and between the 0^+ and 3^- states in vibrational nuclei, which at present are both taken as the first derivative of the optical potential. This will, however,

make it impossible to group together the A-matrices, a procedure explained in detail concerning the subroutines BMATRX and AMATRX. This fact then certainly requires some modifications in these two matrices as well as in COUPLE. However, no other routines, particularly cross section routines, seem to require any major modifications.

In any event, if one made any modifications, it is strongly suggested that one output the quantities obtained by such new possibilities. For such a purpose, efficient use of KTLOUT is recommended.

In coding the present program, every care was exercised to avoid making any possible errors. Nevertheless, it is not impossible that the program still has some errors. In the past few months, fairly large modifications were made throughout the program in order to bring it in final shape appropriate for publication. It is believed that the program has become quite comprehensive this way, but some errors might have been committed in this procedure.

It would therefore be highly appreciated if the reader would notify the author of any errors he may find. These would be transmitted to the other users of the present program.

IV. Examples of Input Data Cards

In this section we show five examples of the input data, one for each value of INTYPE = 2, 3, 4, 5 and 6.

EXAMPLE-1 is for the scattering of a 6.5-MeV proton by ^{92}Zr, with 0^+-2^+ coupling considered. Therefore INTYPE = 2 and INTMAX = 1. β_2 = 0.110 is taken. Differential cross sections to the 0^+ (ground) and 2^+ (assumed to be at 1.0 MeV excitation) states are calculated. Polarizations are not calculated.

EXAMPLE-2 is for the scattering of a 13-MeV proton by ^{112}Cd. 0^+ (ground)-2^+ (0.617 MeV)-2^+ (1.311 MeV) coupling is considered. Therefore INTYPE = 3. CFF is taken. Automatic plot of the cross sections for the three states are made. After a set of calculations with β''_{02} = 0.0415 is done, another calculation with β''_{02} = 0.0 is made.

EXAMPLE -3 is for the scattering of a 14-MeV neutron by ^{165}Ho. Thus INTYPE = 4. ACC is taken (KTRL(7) = 1), and Legendre polynomial expansion of the potential is made (KTRL(11) = 1). Initial polarization of the target is considered. It includes two cases; the first one is for no polarization; the second case is for a 100% polarization. Total and reaction cross sections are computed, but differential cross sections are not. Spin s of the neutron is set to zero to save machine time.

EXAMPLE-4 is for the scattering of a 12.94-MeV proton by ^{107}Ag. Thus INTYPE = 5. $1/2^-$-$3/2^-$-$5/2^-$-$3/2^-$ coupling is considered. Parameters, particularly those in CARD-19, are those used in obtaining the result of ref. 5.

EXAMPLE-5 is for the scattering of a 14-MeV neutron by ^{12}C. 0^+-2^+-0^+ coupling is considered assuming that the second 0^+ state can be interpreted as a β-vibrational state. There INTYPE = 6 and KTRL(11) = 1, but KTRL(7) = 0.

********** EXAMPLE-1 **********

```
                1                                                       KTRL    1
                                                                        KTRL    2
                        1     1     1                              6    KEXCOM  3
                1       1           1           1     1     1           KTLOUT  4
                                                                        KTLOUT  5
                                                                        EXTCOM  6
  2     2     1     1    35     2     0     2     0     1              IICPLE  7
  0     1      0.000     2     1      1.000                             IIREAD  8
6.500  1.000  92.00  40.00  .0125  0.100                               ELAB    9
5.000  5.000                                                            ANGMIN10
53.40  0.000  8.000  7.500  1.000  1.200  1.000  1.000  1.000  1.000   VSX    17
0.650  0.650  0.470  0.650  1.250  1.250  1.250  1.250  1.250          DFN    18
0.110                                                                 1 VCPL   19
```

********** EXAMPLE-2 **********

```
                                                        1               KTRL    1
                1                                                       KTRL    2
                        1     1     1                              10   KEXCOM  3
                1                                       1               KTLOUT  4
                                                                        KTLOUT  5
                                                                        EXTCOM  6
  3     3     7     1    29     3     3     3     0     1              IICPLE  7
  0     1      0.000     2     1      0.617     2     1      1.311     IIREAD  8
13.00  1.000  112.0  48.00  .0125  0.100                               ELAB    9
20.00  5.000                                                            ANGMIN10
0.940  0.970  0.940  0.820  0.640  0.500  0.450  0.480  0.580  0.670  CD112-011
0.680  0.610  0.500  0. 70  0.260  0.220  0.230  0.300  0.380  0.440  CD112-012
0.470  0.460  0.410  0.350  0.280  0.230  0.215  0.215  0.240         CD112-013
6.800  5.600  4.600  4.100  3.500  2.900  2.600  2.700  2.700  2.700  CD112-211
2.700  2.700  2.500  2.000  1.550  1.150  0.850  0.800  0.830  0.860  CD112-212
0.920  0.970  1.000  0.980  0.900  0.760  0.650  0.520  0.400         CD112-213
0.600  0.560  0.490  0.400  0.370  0.280  0.215  0.215  0.215  0.215  CD112-221
0.220  0.230  0.220  0.170  0.110  0.080  0.040  0.028  0.026  0.026  CD112-222
0.040  0.047  0.050  0.052  0.049  0.041  0.033  0.033  0.033         CD112-223
```

```
 55.70  0.000  7.600  6.000  1.000  1.200  1.000  1.000  1.000  1.000  VSX   17
 0.700  0.700  0.700  0.700  1.200  1.200  1.250  1.200  1.250         DFN   18
 0.200                0.129  0.160                       .0415        3 VCPL 19
 55.70  0.000  7.600  6.000  1.000  1.200  1.000  1.000  1.000  1.000  VSX   17
 0.700  0.700  0.700  0.700  1.200  1.200  1.250  1.200  1.250         DFN   18
 0.200                0.129  0.160                                    1 VCPL 19
```

```
        **************          EXAMPLE-3          ***************

  1        2                 1    3              1                  KTRL   1
                                                                     KTRL   2
                                                                 20  KEXCOM 3
           1                                     1                  KTLOUT 4
                                                                     KTLOUT 5
                                                                     EXTCOM 6
  1    4    1    0   30    0    0    0    0    1                    IICPLE 7
  4    1      0.000    5    1      0.198                            IIREAD 8
14.00  1.000  165.0  0.000  .0125  0.100                            ELAB   9
25.00  5.000                                                         ANGMIN10
  1  0.000                                                           NFAI  11
  2    1    8    1    1                                             NPOLST12
  4    3    2    1    0   -1   -2   -3                              MENB2 14
0.125  0.125  0.125  0.125  0.125  0.125  0.125  0.125               OCCUP 15
0.000  0.000  0.000                                                  EULER 16
  4                                                                  MENB2 14
1.000                                                                OCCUP 15
0.000  90.00  0.000                                                  EULER 16
45.50  0.000  1.000  6.500  1.000  1.000  1.000  1.000  1.000  1.000 VSX   17
0.600  0.600  0.600  0.600  1.350  1.350  1.350  1.350  1.350         DFN   18
0.330                                                               1 VCPL 19
```

```
        **************          EXAMPLE-4          ***************

  1                                                                  KTRL   1
                                                                     KTRL   2
                     1    1    1                        2       11   KEXCOM 3
```

```
          1                                   1                KTLOUT 4
                                                               KTLOUT 5
                                                               EXTCOM 6
  3    5    2    1   35    3    0    3    0    1              IICPLE 7
  1    2    0.000    2    2    0.324    3    2    0.423       IIREAD81
  3    2    0.780                                             IIREAD82
12.94  1.000  107.0  47.00  .0125  0.100                      ELAB   9
5.000  5.000                                                  ANGMIN10
54.30  0.000  7.440  6.000  1.000  1.000  1.000  1.000  1.000  1.000  VSX   17
0.680  0.680  0.710  0.680  1.210  1.210  1.240  1.250  1.250        DFN   18
0.230  0.120  0.166  1.000                0.997  1.000              1 VCPL 19
```

********** EXAMPLE-5 **********

```
                                                 1              KTRL   1
                                                                KTRL   2
                    1    1    1                              8  KEXCOM 3
          1                         1    1    1                KTLOUT 4
                                                                KTLOUT 5
                                                                EXTCOM 6
  3    6    7    1   30    3    0    3    0    1                IICPLE 7
  2    1                                                       KEX21 71
  0    1    0.000    2    1    4.430    0    1    7.66         IIREAD81
14.10  1.000  12.00  0.000  .0125  0.100                      ELAB   9
10.00  5.000                                                  ANGMIN10
47.00  0.000  2.000  5.000  1.000  1.000  1.000  1.000  1.000  1.000  VSX   17
0.350  0.650  0.200  0.200  1.250  1.250  1.150  1.150  1.250        DFN   18
-.600  0.000  0.200  0.000  0.000  0.000  0.000                    3 VCPL 19
```

V. Examples of Machine Outputs

Since the machine output is usually voluminous, we shall give only one example here. This is the result of a running made with the input data presented as EXAMPLE-1 in the last section.

Although this is one of the simplest cases to be run on JP1 since it uses INTYPE = 2, still it uses the most important parts of this program, and it is believed to be useful for a test running that the reader would do after he has converted (with varying amount) the present program to make it compatible to the computer available to him.

This example contains some amount of intermediate outputs. The first page is just the reproduction of the input data. The second page gives first the quantities basic for the computation and then the Coulomb wave functions. Next, the form factors are output and then at the end of the fourth page, B- and A-matrices are given for JJ = K = 1. C-matrix elements and the quantum numbers n, ℓ and J are given for all values of JJ and K. Finally, various cross sections are output.

```
**********        COUPLED CHANNEL CALCULATION        **********
                    (ON PROGRAM JUPITOR-1)

ELAB= 6.500,  CHARGE= 40.0,  TMAS= 92.000,  PMAS= 1.000

VSX,WSX,WSF,VSO                    =  53.400      0   8.000   7.500
DFN,DFNW,DFNS,DFNSP                =    .650    .650    .470    .650
RZERO,RZEROW,RZEROS,RZROSP,RZEROC=    1.250   1.250   1.250   1.250   1.250
WC                                 =   1.000   1.200   1.000   1.000   1.000   1.000

PROJECTILE SPIN=1/2

IIRMAX=2,INTYPE=2,INTMAX=1,BETA=  .110    -0     -0     -0     -0     -0     -0     -0     -0     -0

TARGET STATES,  0  (+) AT      0 MEV,  2  (+) AT  1.000 MEV,

KTRL     -0   -0    1   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0
KEXCOM   -0   -0   -0   -0    1    1    1   -0   -0   -0   -0   -0   -0    6
KTLOUT   -0   -0    1   -0    1   -0    1   -0    1    1    1   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0
EXTCOM        -0        -0        -0        -0        -0        -0        -0        -0        -0        -0
XMES1= .01250    XMES2= .10000    UNIT=AMU
```

```
ECM        6.43011E+00    5.43011E+00
WN         5.51599E-01    5.06896E-01
WNINI      1.68257E+00    1.66845E+00
ETA        2.47034E+00    2.68820E+00
SIGMO      5.15198E-01    7.24336E-01

JJJMAX,KEX14,NXCPLE,NXMAX=    6    6  121  T21 XMAX,XBAR=  1.21000000E+01  5.6429468E+00

IN FLGLCH
ETA,SIGMAZ,RHOMX,RHOMXG,RD=  2.470342E+00  5.151980E-01  6.674348E+00  1.108714E+01  5.515990E-02
LMAX,NINC,NMAX=    6    3   80
F   1.3095854E+00  1.2527150E+00  1.0790706E+00  8.0200267E-01  5.0965559E-01  2.7898862E-01  1.331929E-01
G   3.4413082E-01  6.1570842E-01  1.0326603E+00  1.4829036E+00  2.0032247E+00  2.8556955E+00  4.6777889E+00
FD  9.2764469E-02  2.0780934E-01  3.3466510E-01  3.7693706E-01  3.2242430E-01  2.2102662E-01
GD -7.3922392E-01 -6.9612805E-01 -6.0645210E-01 -5.4992171E-01 -6.9480585E-01 -1.3158309E+00
WR  1.4551915E-11  5.8207661E-11  4.3655746E-11  1.8917490E-10  1.8917490E-10  1.4551915E-10

IN FLGLCH
ETA,SIGMAZ,RHOMX,RHOMXG,RD=  2.688202E+00  7.243364E-01  6.133437E+00  1.018860E+01  5.068956E-02
LMAX,NINC,NMAX=    6    3   80
F   1.0945666E+00  9.7350442E-01  7.5343141E-01  4.9980667E-01  2.8428474E-01  1.3991685E-01  6.0302800E-02
G   1.0305903E+00  1.2351354E+00  1.5690737E+00  2.0293582E+00  2.8148838E+00  4.4836121E+00  8.4547799E+00
FD  3.2869353E-01  3.6371145E-01  3.7253749E-01  3.1933160E-01  2.2573600E-01  1.3348171E-01
GD -6.0412192E-01 -5.6575718E-01 -5.5142541E-01 -7.0419588E-01 -1.2824445E+00 -2.8697029E+00
WR  3.6379788E-11  2.9103830E-11  1.0186341E-10  2.1827873E-10  2.6193447E-10  2.6193447E-10
X
 1.2500E+02  2.5000E+02  3.7500E+02  5.0000E+02  6.2500E+02  7.5000E+02  8.7500E+02  1.0000E+01  1.2500E+01  1.5000E+01
 1.7500E+01  2.0000E+01  2.5000E+01  3.0000E+01  3.5000E+01  4.0000E+01  5.0000E+01  6.0000E+01  7.0000E+01  8.0000E+01
 9.0000E+01  1.0000E+00  1.1000E+00  1.2000E+00  1.3000E+00  1.4000E+00  1.5000E+00  1.6000E+00  1.7000E+00  1.8000E+00
 1.9000E+00  2.0000E+00  2.1000E+00  2.2000E+00  2.3000E+00  2.4000E+00  2.5000E+00  2.6000E+00  2.7000E+00  2.8000E+00
 2.9000E+00  3.0000E+00  3.1000E+00  3.2000E+00  3.3000E+00  3.4000E+00  3.5000E+00  3.6000E+00  3.7000E+00  3.8000E+00
 3.9000E+00  4.0000E+00  4.1000E+00  4.2000E+00  4.3000E+00  4.4000E+00  4.5000E+00  4.6000E+00  4.7000E+00  4.8000E+00
 4.9000E+00  5.0000E+00  5.1000E+00  5.2000E+00  5.3000E+00  5.4000E+00  5.5000E+00  5.6000E+00  5.7000E+00  5.8000E+00
 5.9000E+00  6.0000E+00  6.1000E+00  6.2000E+00  6.3000E+00  6.4000E+00  6.5000E+00  6.6000E+00  6.7000E+00  6.8000E+00
 6.9000E+00  7.0000E+00  7.1000E+00  7.2000E+00  7.3000E+00  7.4000E+00  7.5000E+00  7.6000E+00  7.7000E+00  7.8000E+00
 7.9000E+00  8.0000E+00  8.1000E+00  8.2000E+00  8.3000E+00  8.4000E+00  8.5000E+00  8.6000E+00  8.7000E+00  8.8000E+00
 8.9000E+00  9.0000E+00  9.1000E+00  9.2000E+00  9.3000E+00  9.4000E+00  9.5000E+00  9.6000E+00  9.7000E+00  9.8000E+00
 9.9000E+00  1.0000E+01  1.0100E+01  1.0200E+01  1.0300E+01  1.0400E+01  1.0500E+01  1.0600E+01  1.0700E+01  1.0800E+01
 1.0900E+01  1.1000E+01  1.1100E+01  1.1200E+01  1.1300E+01  1.1400E+01  1.1500E+01  1.1600E+01  1.1700E+01  1.1800E+01
 1.1900E+01  1.2000E+01  1.2100E+01  1.2200E+01  1.2300E+01
VCENTR
-5.3391E+01 -5.3391E+01 -5.3390E+01 -5.3390E+01 -5.3390E+01 -5.3390E+01 -5.3390E+01 -5.3389E+01 -5.3389E+01 -5.3389E+01
-5.3388E+01 -5.3388E+01 -5.3387E+01 -5.3386E+01 -5.3384E+01 -5.3383E+01 -5.3380E+01 -5.3377E+01 -5.3373E+01 -5.3369E+01
-5.3364E+01 -5.3358E+01 -5.3351E+01 -5.3343E+01 -5.3333E+01 -5.3322E+01 -5.3309E+01 -5.3294E+01 -5.3276E+01 -5.3256E+01
-5.3232E+01 -5.3204E+01 -5.3172E+01 -5.3134E+01 -5.3090E+01 -5.3039E+01 -5.2979E+01 -5.2910E+01 -5.2829E+01 -5.2735E+01
-5.2626E+01 -5.2500E+01 -5.2353E+01 -5.2183E+01 -5.1986E+01 -5.1758E+01 -5.1495E+01 -5.1191E+01 -5.0841E+01 -5.0439E+01
-4.9978E+01 -4.9451E+01 -4.8851E+01 -4.8168E+01 -4.7396E+01 -4.6526E+01 -4.5551E+01 -4.4464E+01 -4.3260E+01 -4.1935E+01
-4.0489E+01 -3.8924E+01 -3.7245E+01 -3.5461E+01 -3.3585E+01 -3.1632E+01 -2.9624E+01 -2.7582E+01 -2.5529E+01 -2.3490E+01
-2.1488E+01 -1.9546E+01 -1.7681E+01 -1.5911E+01 -1.4248E+01 -1.2699E+01 -1.1271E+01 -9.9634E+00 -8.7761E+00 -7.7052E+00
-6.7453E+00 -5.8895E+00 -5.1304E+00 -4.4599E+00 -3.8700E+00 -3.3528E+00 -2.9007E+00 -2.5065E+00 -2.1636E+00 -1.8658E+00
-1.6078E+00 -1.3845E+00 -1.1914E+00 -1.0248E+00 -8.8109E-01 -7.5723E-01 -6.5057E-01 -5.5877E-01 -4.7981E-01 -4.1192E-01
-3.5357E-01 -3.0344E-01 -2.6038E-01 -2.2340E-01 -1.9166E-01 -1.6442E-01 -1.4103E-01 -1.2097E-01 -1.0375E-01 -8.8982E-02
-7.6311E-02 -6.5443E-02 -5.6121E-02 -4.8126E-02 -4.1268E-02 -3.5388E-02 -3.0344E-02 -2.6019E-02 -2.2311E-02 -1.9130E-02
-1.6403E-02 -1.4065E-02 -1.2060E-02 -1.0340E-02 -8.8662E-03 -7.6021E-03 -6.5182E-03 -5.5888E-03 -4.7919E-03 -4.1087E-03
-3.5228E-03 -3.0205E-03 -2.5898E-03 -2.2205E-03 -1.9039E-03
VCENTI
-2.0065E-04 -2.0006E-04 -2.1161E-04 -2.1731E-04 -2.2317E-04 -2.2919E-04 -2.3536E-04 -2.4171E-04 -2.5491E-04 -2.6884E-04
-2.8352E-04 -2.9901E-04 -3.3258E-04 -3.6991E-04 -4.1143E-04 -4.5761E-04 -5.6610E-04 -7.0031E-04 -8.6635E-04 -1.0717E-03
-1.3258E-03 -1.6401E-03 -2.0290E-03 -2.5099E-03 -3.1049E-03 -3.8409E-03 -4.7513E-03 -5.8774E-03 -7.2702E-03 -8.9930E-03
```

-1.1124E-02	-1.3759E-02	-1.7017E-02	-2.1047E-02	-2.6029E-02	-3.2188E-02	-3.9800E-02	-4.9208E-02	-6.0830E-02	-7.5185E-02
-9.2907E-02	-1.1478E-01	-1.4175E-01	-1.7499E-01	-2.1591E-01	-2.6624E-01	-3.2806E-01	-4.0386E-01	-4.9660E-01	-6.0977E-01
-7.4744E-01	-9.1425E-01	-1.1154E+00	-1.3565E+00	-1.6433E+00	-1.9816E+00	-2.3761E+00	-2.8301E+00	-3.3438E+00	-3.9135E+00
-4.5298E+00	-5.1763E+00	-5.8291E+00	-6.4569E+00	-7.0230E+00	-7.4885E+00	-7.8178E+00	-7.9833E+00	-7.9706E+00	-7.7808E+00
-7.4304E+00	-6.9483E+00	-6.3712E+00	-5.7378E+00	-5.0841E+00	-4.4406E+00	-3.8300E+00	-3.2678E+00	-2.7624E+00	-2.3169E+00
-1.9306E+00	-1.5999E+00	-1.3198E+00	-1.0847E+00	-8.8876E-01	-7.2637E-01	-5.9242E-01	-4.8236E-01	-3.9221E-01	-3.1856E-01
-2.5850E-01	-2.0961E-01	-1.6987E-01	-1.3759E-01	-1.1141E-01	-9.0176E-02	-7.2972E-02	-5.9038E-02	-4.7757E-02	-3.8626E-02
-3.1238E-02	-2.5261E-02	-2.0425E-02	-1.6515E-02	-1.3352E-02	-1.0795E-02	-8.7272E-03	-7.0554E-03	-5.7037E-03	-4.6108E-03
-3.7274E-03	-3.0131E-03	-2.4357E-03	-1.9690E-03	-1.5916E-03	-1.2866E-03	-1.0401E-03	-8.4073E-04	-6.7961E-04	-5.4936E-04
-4.4408E-04	-3.5897E-04	-2.9017E-04	-2.3456E-04	-1.8961E-04	-1.5327E-04	-1.2389E-04	-1.0015E-04	-8.0955E-05	-6.5440E-05
-5.2898E-05	-4.2760E-05	-3.4565E-05	-2.7940E-05	-2.2585E-05					

VSPIN

3.1927E-01	1.6273E-01	1.1060E-01	8.4557E-02	6.8958E-02	5.8581E-02	5.1186E-02	4.5657E-02	3.7958E-02	3.2871E-02
2.9280E-02	2.6624E-02	2.3001E-02	2.0699E-02	1.9160E-02	1.8105E-02	1.6891E-02	1.6415E-02	1.6407E-02	1.6741E-02
1.7353E-02	1.8211E-02	1.9304E-02	2.0631E-02	2.2204E-02	2.4037E-02	2.6153E-02	2.8580E-02	3.1351E-02	3.4507E-02
3.8094E-02	4.2164E-02	4.6777E-02	5.2003E-02	5.7919E-02	6.4612E-02	7.2180E-02	8.0735E-02	9.0399E-02	1.0131E-01
1.1361E-01	1.2747E-01	1.4307E-01	1.6061E-01	1.8027E-01	2.0228E-01	2.2686E-01	2.5421E-01	2.8455E-01	3.1805E-01
3.5486E-01	3.9506E-01	4.3867E-01	4.8559E-01	5.3558E-01	5.8825E-01	6.4301E-01	6.9905E-01	7.5533E-01	8.1059E-01
8.6335E-01	9.1198E-01	9.5477E-01	9.9001E-01	1.0162E+00	1.0319E+00	1.0364E+00	1.0291E+00	1.0102E+00	9.8032E-01
9.4058E-01	8.9250E-01	8.3787E-01	7.7859E-01	7.1657E-01	6.5358E-01	5.9117E-01	5.3066E-01	4.7303E-01	4.1902E-01
3.6910E-01	3.2349E-01	2.8227E-01	2.4533E-01	2.1250E-01	1.8351E-01	1.5806E-01	1.3583E-01	1.1651E-01	9.9762E-02
8.5302E-02	7.2848E-02	6.2148E-02	5.2972E-02	4.5118E-02	3.8404E-02	3.2673E-02	2.7784E-02	2.3619E-02	2.0072E-02
1.7054E-02	1.4487E-02	1.2305E-02	1.0450E-02	8.8742E-03	7.5355E-03	6.3986E-03	5.4331E-03	4.6133E-03	3.9173E-03
3.3264E-03	2.8247E-03	2.3987E-03	2.0371E-03	1.7301E-03	1.4695E-03	1.2482E-03	1.0603E-03	9.0071E-04	7.6521E-04
6.5014E-04	5.5242E-04	4.6941E-04	3.9891E-04	3.3902E-04	2.8814E-04	2.4491E-04	2.0819E-04	1.7698E-04	1.5046E-04
1.2792E-04	1.0877E-04	9.2491E-05	7.8653E-05	6.6890E-05					

VCOULM

1.5310E+01	1.5310E+01	1.5309E+01	1.5309E+01	1.5309E+01	1.5309E+01	1.5308E+01	1.5308E+01	1.5307E+01	1.5306E+01
1.5305E+01	1.5303E+01	1.5300E+01	1.5295E+01	1.5290E+01	1.5284E+01	1.5270E+01	1.5252E+01	1.5231E+01	1.5207E+01
1.5180E+01	1.5149E+01	1.5116E+01	1.5079E+01	1.5039E+01	1.4996E+01	1.4949E+01	1.4899E+01	1.4847E+01	1.4790E+01
1.4731E+01	1.4669E+01	1.4603E+01	1.4534E+01	1.4462E+01	1.4387E+01	1.4308E+01	1.4226E+01	1.4141E+01	1.4053E+01
1.3962E+01	1.3867E+01	1.3770E+01	1.3669E+01	1.3564E+01	1.3457E+01	1.3346E+01	1.3233E+01	1.3116E+01	1.2996E+01
1.2872E+01	1.2746E+01	1.2616E+01	1.2483E+01	1.2346E+01	1.2207E+01	1.2064E+01	1.1919E+01	1.1769E+01	1.1617E+01
1.1462E+01	1.1303E+01	1.1141E+01	1.0976E+01	1.0808E+01	1.0636E+01	1.0462E+01	1.0284E+01	1.0104E+01	9.9301E+00
9.7618E+00	9.5991E+00	9.4417E+00	9.2895E+00	9.1420E+00	8.9992E+00	8.8607E+00	8.7265E+00	8.5962E+00	8.4698E+00
8.3470E+00	8.2278E+00	8.1119E+00	7.9993E+00	7.8897E+00	7.7831E+00	7.6793E+00	7.5782E+00	7.4798E+00	7.3839E+00
7.2905E+00	7.1993E+00	7.1104E+00	7.0237E+00	6.9391E+00	6.8565E+00	6.7758E+00	6.6970E+00	6.6201E+00	6.5448E+00
6.4713E+00	6.3994E+00	6.3291E+00	6.2603E+00	6.1930E+00	6.1271E+00	6.0626E+00	5.9994E+00	5.9376E+00	5.8770E+00
5.8176E+00	5.7595E+00	5.7024E+00	5.6465E+00	5.5917E+00	5.5379E+00	5.4852E+00	5.4335E+00	5.3827E+00	5.3328E+00
5.2839E+00	5.2359E+00	5.1887E+00	5.1424E+00	5.0969E+00	5.0522E+00	5.0082E+00	4.9651E+00	4.9226E+00	4.8809E+00
4.8399E+00	4.7995E+00	4.7599E+00	4.7209E+00	4.6825E+00					

VCPL|R

-8.0173E-02	-8.1729E-02	-8.3358E-02	-8.4932E-02	-8.6581E-02	-8.8261E-02	-8.9974E-02	-9.1721E-02	-9.3316E-02	-9.9052E-02
-1.0293E-01	-1.0697E-01	-1.1552E-01	-1.2475E-01	-1.3472E-01	-1.4548E-01	-1.6966E-01	-1.9785E-01	-2.3072E-01	-2.6905E-01
-3.1374E-01	-3.6583E-01	-4.2656E-01	-4.9735E-01	-5.7986E-01	-6.7602E-01	-7.8807E-01	-9.1861E-01	-1.0707E+00	-1.2478E+00
-1.4540E+00	-1.6940E+00	-1.9734E+00	-2.2983E+00	-2.6761E+00	-3.1151E+00	-3.6251E+00	-4.2169E+00	-4.9032E+00	-5.6984E+00
-6.6187E+00	-7.6824E+00	-8.9100E+00	-1.0324E+01	-1.1951E+01	-1.3816E+01	-1.5951E+01	-1.8385E+01	-2.1150E+01	-2.4279E+01
-2.7802E+01	-3.1745E+01	-3.6131E+01	-4.0971E+01	-4.6264E+01	-5.1996E+01	-5.8128E+01	-6.4598E+01	-7.1317E+01	-7.8162E+01
-8.4984E+01	-9.1603E+01	-9.7819E+01	-1.0342E+02	-1.0819E+02	-1.1194E+02	-1.1451E+02	-1.1577E+02	-1.1567E+02	-1.1422E+02
-1.1148E+02	-1.0758E+02	-1.0267E+02	-9.6975E+01	-9.0689E+01	-8.4030E+01	-7.7194E+01	-7.0358E+01	-6.3668E+01	-5.7241E+01
-5.1162E+01	-4.5491E+01	-4.0260E+01	-3.5485E+01	-3.1163E+01	-2.7280E+01	-2.3814E+01	-2.0739E+01	-1.8022E+01	-1.5632E+01
-1.3538E+01	-1.1707E+01	-1.0113E+01	-8.7261E+00	-7.5229E+00	-6.4806E+00	-5.5790E+00	-4.8002E+00	-4.1280E+00	-3.5484E+00
-3.0491E+00	-2.6193E+00	-2.2494E+00	-1.9314E+00	-1.6579E+00	-1.4230E+00	-1.2211E+00	-1.0478E+00	-8.9897E-01	-7.7120E-01
-6.6155E-01	-5.6745E-01	-4.8670E-01	-4.1742E-01	-3.5799E-01	-3.0701E-01	-2.6328E-01	-2.2578E-01	-1.9361E-01	-1.6602E-01
-1.4236E-01	-1.2207E-01	-1.0467E-01	-8.9752E-02	-7.6958E-02	-6.5988E-02	-5.6580E-02	-4.8514E-02	-4.1597E-02	-3.5667E-02
-3.0581E-02	-2.6221E-02	-2.2482E-02	-1.9277E-02	-1.6528E-02					

VCPL|I

2.4090E-03	2.4740E-03	2.5406E-03	2.6091E-03	2.6794E-03	2.7516E-03	2.8258E-03	2.9020E-03	3.0605E-03	3.2277E-03
3.4040E-03	3.5900E-03	3.9929E-03	4.4411E-03	4.9396E-03	5.4940E-03	6.7965E-03	8.4078E-03	1.0401E-02	1.2867E-02
1.5917E-02	1.9690E-02	2.4357E-02	3.0130E-02	3.7271E-02	4.6104E-02	5.7028E-02	7.0539E-02	8.7249E-02	1.0791E-01

1.3346E-01	1.6505E-01	2.0410E-01	2.5236E-01	3.1200E-01	3.8508E-01	4.7666E-01	5.8898E-01	7.2756E-01	8.9844E-01
1.1090E+00	1.3681E+00	1.6867E+00	2.0778E+00	2.5571E+00	3.1430E+00	3.8572E+00	4.7249E+00	5.7742E+00	7.0365E+00
8.5444E+00	1.0330E+01	1.2423E+01	1.4841E+01	1.7588E+01	2.0636E+01	2.3919E+01	2.7315E+01	3.0628E+01	3.3582E+01
3.5819E+01	3.6923E+01	3.6457E+01	3.4047E+01	2.9467E+01	2.2734E+01	1.4165E+01	4.3762E+00	-5.8012E+00	-1.5464E+01
-2.3805E+01	-3.0247E+01	-3.4516E+01	-3.6633E+01	-3.6852E+01	-3.5562E+01	-3.3199E+01	-3.0175E+01	-2.6836E+01	-2.3446E+01
-2.0190E+01	-1.7181E+01	-1.4480E+01	-1.2108E+01	-1.0061E+01	-8.3156E+00	-6.8443E+00	-5.6141E+00	-4.5921E+00	-3.7478E+00
-3.0531E+00	-2.4835E+00	-2.0177E+00	-1.6377E+00	-1.3282E+00	-1.0766E+00	-8.7212E+01	-7.0621E+01	-5.7167E+01	-4.6264E+01
-3.7432E+01	-3.0281E+01	-2.4492E+01	-1.9808E+01	-1.6018E+01	-1.2952E+01	-1.0472E+01	-8.4671E+02	-6.8455E+02	-5.5343E+02
-4.4741E+02	-3.6170E+02	-2.9240E+02	-2.3637E+02	-1.9108E+02	-1.5446E+02	-1.2486E+02	-1.0094E+02	-8.1592E+03	-6.5956E+03
-5.3316E+03	-4.3098E+03	-3.4838E+03	-2.8162E+03	-2.2764E+03	-1.8402E+03	-1.4875E+03	-1.2024E+03	-9.7196E+04	-7.8568E+04
-6.3510E+04	-5.1338E+04	-4.1499E+04	-3.3546E+04	-2.7117E+04					

VCPL2R

3.4789E+01	3.5464E+01	3.6152E+01	3.6853E+01	3.7568E+01	3.8297E+01	3.9040E+01	3.9798E+01	4.1357E+01	4.2977E+01
4.4661E+01	4.6410E+01	5.0118E+01	5.4121E+01	5.8443E+01	6.3110E+01	7.3591E+01	8.5809E+01	1.0005E+00	1.1665E+00
1.3600E+00	1.5855E+00	1.8482E+00	2.1542E+00	2.5107E+00	2.9258E+00	3.4091E+00	3.9716E+00	4.6260E+00	5.3871E+00
6.2717E+00	7.2994E+00	8.4926E+00	9.8769E+00	1.1481E+01	1.3339E+01	1.5487E+01	1.7968E+01	2.0828E+01	2.4119E+01
2.7897E+01	3.2223E+01	3.7160E+01	4.2773E+01	4.9128E+01	5.6285E+01	6.4296E+01	7.3200E+01	8.3008E+01	9.3702E+01
1.0521E+02	1.1742E+02	1.3011E+02	1.4299E+02	1.5566E+02	1.6759E+02	1.7814E+02	1.8655E+02	1.9200E+02	1.9360E+02
1.9052E+02	1.8205E+02	1.6770E+02	1.4730E+02	1.2109E+02	8.9765E+01	5.4435E+01	1.6596E+01	-2.2022E+01	-5.9609E+01
-9.4457E+01	-1.2512E+02	-1.5054E+02	-1.7009E+02	-1.8359E+02	-1.9126E+02	-1.9364E+02	-1.9144E+02	-1.8553E+02	-1.7676E+02
-1.6598E+02	-1.5391E+02	-1.4118E+02	-1.2830E+02	-1.1566E+02	-1.0354E+02	-9.2142E+01	-8.1570E+01	-7.1889E+01	-6.3113E+01
-5.5224E+01	-4.8184E+01	-4.1938E+01	-3.6424E+01	-3.1577E+01	-2.7333E+01	-2.3627E+01	-2.0400E+01	-1.7596E+01	-1.5165E+01
-1.3060E+01	-1.1240E+01	-9.6690E+00	-8.3134E+00	-7.1450E+00	-6.1387E+00	-5.2726E+00	-4.5276E+00	-3.8870E+00	-3.3364E+00
-2.8634E+00	-2.4571E+00	-2.1082E+00	-1.8087E+00	-1.5515E+00	-1.3309E+00	-1.1415E+00	-9.7908E+01	-8.3970E+01	-7.2013E+01
-6.1757E+01	-5.2960E+01	-4.5415E+01	-3.8944E+01	-3.3394E+01	-2.8635E+01	-2.4554E+01	-2.1054E+01	-1.8053E+01	-1.5479E+01
-1.3273E+01	-1.1381E+01	-9.7580E+02	-8.3668E+02	-7.1739E+02					

VCPL2I

-1.4461E+02	-1.4851E+02	-1.5251E+02	-1.5662E+02	-1.6084E+02	-1.6518E+02	-1.6963E+02	-1.7420E+02	-1.8372E+02	-1.9376E+02
-2.0434E+02	-2.1550E+02	-2.3969E+02	-2.6659E+02	-2.9651E+02	-3.2979E+02	-4.0797E+02	-5.0469E+02	-6.2432E+02	-7.7230E+02
-9.5535E+02	-1.1818E+01	-1.4618E+01	-1.8082E+01	-2.2366E+01	-2.7663E+01	-3.4214E+01	-4.2315E+01	-5.2329E+01	-6.4708E+01
-8.0007E+01	-9.8911E+01	-1.2226E+00	-1.5110E+00	-1.8669E+00	-2.3059E+00	-2.8472E+00	-3.5139E+00	-4.3344E+00	-5.3426E+00
-6.5797E+00	-8.0945E+00	-9.9450E+00	-1.2198E+01	-1.4932E+01	-1.8232E+01	-2.2191E+01	-2.6904E+01	-3.2460E+01	-3.8924E+01
-4.6322E+01	-5.4599E+01	-6.3579E+01	-7.2902E+01	-8.1949E+01	-8.9758E+01	-9.4960E+01	-9.5740E+01	-8.9904E+01	-7.5092E+01
-4.9190E+01	-1.0984E+01	3.9057E+01	9.8045E+01	1.6036E+02	2.1811E+02	2.6249E+02	2.8590E+02	2.8407E+02	2.5735E+02
2.1057E+02	1.5165E+02	8.9364E+01	3.1360E+01	-1.7124E+01	-5.3576E+01	-7.7811E+01	-9.1217E+01	-9.5977E+01	-9.4447E+01
-8.8778E+01	-8.0721E+01	-7.1586E+01	-6.2281E+01	-5.3383E+01	-4.5223E+01	-3.7956E+01	-3.1622E+01	-2.6190E+01	-2.1589E+01
-1.7728E+01	-1.4514E+01	-1.1853E+01	-9.6612E+00	-7.8619E+00	-6.3895E+00	-5.1875E+00	-4.2081E+00	-3.4113E+00	-2.7639E+00
-2.2383E+00	-1.8120E+00	-1.4665E+00	-1.1866E+00	-9.5996E+01	-7.7648E+01	-6.2799E+01	-5.0784E+01	-4.1065E+01	-3.3204E+01
-2.6846E+01	-2.1705E+01	-1.7548E+01	-1.4186E+01	-1.1468E+01	-9.2711E+02	-7.4947E+02	-6.0586E+02	-4.8977E+02	-3.9591E+02
-3.2004E+02	-2.5871E+02	-2.0938E+02	-1.6905E+02	-1.3665E+02	-1.1047E+02	-8.9295E+03	-7.2181E+03	-5.8348E+03	-4.7165E+03
-3.8126E+03	-3.0819E+03	-2.4912E+03	-2.0138E+03	-1.6278E+03					

BMATRX FOR INTYPE=2 INTMAX=1 LAMDA= 2

0	3.10304E-02	
0	0	

*** AMATRIX

0	-2.40361E-02	-1.96254E-02
-2.40361E-02	0	0
-1.96254E-02	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

```
                0                0                0
                0                0                0
                0                0                0
                0                0                0
                0                0                0

**CMATRIX FOR JJ,K,NLJ=  1  1  1
(-1.401410E-01  1.349926E-01) ( 2.750237E-03 -1.559105E-03) ( 1.199652E-03 -2.946383E-03) (

N=    1    2    2
L=    0    2    2
J=   1/2  5/2  3/2

**CMATRIX FOR JJ,K,NLJ=  1  2  1
( 4.951707E-02  6.136989E-02) (-8.760143E-04  3.246638E-03) ( 1.536092E-02  7.742887E-03) (

N=    1    2    2
L=    1    3    1
J=   1/2  5/2  3/2

**CMATRIX FOR JJ,K,NLJ=  2  1  1
(-2.496716E-02  4.168742E-02) ( 1.929499E-04  1.376254E-04) (-2.971716E-05 -1.663388E-03) ( 2.015055E-03  2.104988E-03)
( 1.343257E-02  1.263805E-04) (

N=    1    2    2    2    2
L=    2    4    2    2    0
J=   3/2  7/2  5/2  3/2  1/2

**CMATRIX FOR JJ,K,NLJ=  2  2  1
( 7.361576E-02  9.462940E-02) (-2.701566E-03  3.713664E-03) (-5.833851E-04  1.349420E-03) (-1.145323E-02 -8.058260E-03)
(-1.058809E-02 -6.215575E-03) (

N=    1    2    2    2    2
L=    1    3    3    1    1
J=   3/2  7/2  5/2  3/2  1/2

**CMATRIX FOR JJ,K,NLJ=  3  1  1
(-2.022577E-02  2.574969E-02) (-2.923965E-05  4.229065E-04) ( 6.056132E-05  9.471165E-05) (-1.470913E-03  2.401925E-03)
( 4.670248E-05  1.306536E-03) (-8.200525E-03 -6.373828E-03) (

N=    1    2    2    2    2    2
L=    2    4    4    2    2    0
J=   5/2  9/2  7/2  5/2  3/2  1/2

**CMATRIX FOR JJ,K,NLJ=  3  2  1
( 1.071448E-02  7.126099E-03) (-8.245570E-05  1.290666E-04) ( 5.706914E-04  2.672418E-04) (-1.169138E-03 -2.435456E-04)
(-1.208534E-03  2.915264E-03) ( 1.587669E-03 -4.892662E-03) (

N=    1    2    2    2    2    2
L=    3    5    3    3    1    1
J=   5/2  9/2  7/2  5/2  3/2  1/2

**CMATRIX FOR JJ,K,NLJ=  4  1  1
( 6.071070E-04  8.286415E-04) ( 4.105459E-05 -1.755220E-05) (-8.866556E-05  8.192949E-05) ( 1.737588E-04  1.115754E-04)
(-3.157915E-04 -1.430322E-04) (

N=    1    2    2    2    2
L=    4    4    4    2    2
J=   7/2  9/2  7/2  5/2  3/2
```

```
**CMATRIX FOR JJ,K,NLJ=  4  2  1
( 1.712140E-02  1.873968E-02) (-1.709052E-04  2.259845E-05) (-3.222395E-05  2.549726E-05) (-1.480708E-03 -1.590485E-03)
(-4.772929E-04 -2.945644E-04) ( 5.751075E-03 -6.508066E-03) (

N=    1    2    2    2    2    2
L=    3    5    5    3    3    1
J=   7/2  11/2  9/2  7/2  5/2  3/2

**CMATRIX FOR JJ,K,NLJ=  5  1  1
( 2.081288E-03  6.825616E-04) (-1.649916E-04  4.122186E-05) (-3.644868E-05  1.506874E-05) (-2.393571E-04 -6.518487E-04)

N=    1    2    2    2
L=    4    4    4    2
J=   9/2  9/2  7/2  5/2

**CMATRIX FOR JJ,K,NLJ=  5  2  1
( 2.597917E-04  1.115348E-04) ( 3.847026E-06  1.802680E-06) (-2.114313E-05  9.162737E-07) ( 1.881251E-05  6.081680E-05)
(-1.047567E-04 -1.824298E-04) (

N=    1    2    2    2    2
L=    5    5    5    3    3
J=   9/2  11/2  9/2  7/2  5/2

**CMATRIX FOR JJ,K,NLJ=  6  2  1
( 2.805794E-04  2.893333E-04) ( 3.525929E-06 -3.243179E-06) (-2.873118E-06 -1.465811E-06) ( 5.113069E-05 -2.002156E-04)

N=    1    2    2    2
L=    5    5    5    3
J=  11/2  11/2  9/2  7/2


          SGTOTL=   2.691202E+02  REACCR=   2.344804E+02

          STRGFS=   2.425366E+01  STRGFP=   1.980327E+01    FOR NPS=1
```

***** SCATTERING TO 0 (+) STATE AT 0-MEV ***** (FAI= .00-DEGREE) NPS=1

ANGLE	SGMTH	POLTH	ANGLE	SGMTH	POLTH	ANGLE	SGMTH	POLTH
5.0	1.38452E+07	-.000	65.0	4.97364E+02	.003	125.0	6.53592E+01	-.011
10.0	8.70823E+05	.000	70.0	3.73020E+02	.018	130.0	5.74701E+01	-.015
15.0	1.72483E+05	-.000	75.0	2.93977E+02	.033	135.0	5.11413E+01	-.016
20.0	5.45998E+04	-.000	80.0	2.41553E+02	.045	140.0	4.61728E+01	-.014
25.0	2.27811E+04	.001	85.0	2.04440E+02	.052	145.0	4.23542E+01	-.010
30.0	1.14040E+04	.002	90.0	1.76062E+02	.052	150.0	3.94811E+01	-.005
35.0	6.40356E+03	.001	95.0	1.52805E+02	.047	155.0	3.73674E+01	-.000
40.0	3.84675E+03	-.003	100.0	1.32830E+02	.039	160.0	3.58528E+01	.003
45.0	2.40433E+03	-.007	105.0	1.15298E+02	.028	165.0	3.48067E+01	.004
50.0	1.54507E+03	-.011	110.0	9.98792E+01	.016	170.0	3.41290E+01	.004
55.0	1.02033E+03	-.012	115.0	8.64596E+01	.005	175.0	3.37499E+01	.002
60.0	6.97105E+02	-.008	120.0	7.49799E+01	-.004	-0	-6.47658-261	-0

RUTHERFORD CROSS SECTION

5.0	1.38512E+07	50.0	1.57186E+03	95.0	1.69701E+02	140.0	6.43078E+01
10.0	8.69007E+05	55.0	1.10302E+03	100.0	1.45610E+02	145.0	6.06079E+01
15.0	1.72749E+05	60.0	8.02282E+02	105.0	1.26573E+02	150.0	5.76013E+01
20.0	5.51475E+04	65.0	6.01646E+02	110.0	1.11365E+02	155.0	5.51926E+01
25.0	2.28486E+04	70.0	4.63278E+02	115.0	9.91038E+01	160.0	5.33091E+01
30.0	1.11743E+04	75.0	3.65106E+02	120.0	8.91425E+01	165.0	5.18959E+01
35.0	6.13253E+03	80.0	2.93722E+02	125.0	8.10012E+01	170.0	5.09132E+01
40.0	3.66438E+03	85.0	2.40699E+02	130.0	7.43198E+01	175.0	5.03340E+01
45.0	2.33802E+03	90.0	2.00571E+02	135.0	6.88249E+01	-0	-6.47658-261

***** SCATTERING TO 2 (+) STATE AT 1.000-MEV ***** (FAI= .00-DEGREE) NPS=1

ANGLE	SGMTH	POLTH	ANGLE	SGMTH	POLTH	ANGLE	SGMTH	POLTH
5.0	1.58622E+01	-.003	65.0	7.41031E-02	-.121	125.0	2.84916E-02	-.384
10.0	1.55127E+01	-.006	70.0	7.00756E-02	-.151	130.0	2.86762E-02	-.302
15.0	1.49575E+01	-.010	75.0	6.63397E-02	-.186	135.0	3.01941E-02	-.214
20.0	1.42342E+01	-.014	80.0	6.26057E-02	-.225	140.0	3.28665E-02	-.137
25.0	1.33900E+01	-.018	85.0	5.86451E-02	-.267	145.0	3.64123E-02	-.078
30.0	1.24766E+01	-.023	90.0	5.43354E-02	-.312	150.0	4.04850E-02	-.039
35.0	1.15456E+01	-.029	95.0	4.96885E-02	-.357	155.0	4.47119E-02	-.015
40.0	1.06432E+01	-.036	100.0	4.48539E-02	-.401	160.0	4.87301E-02	-.002
45.0	9.80663E-02	-.046	105.0	4.00970E-02	-.440	165.0	5.22151E-02	.004
50.0	9.06078E-02	-.059	110.0	3.57557E-02	-.465	170.0	5.49007E-02	.005
55.0	8.41681E-02	-.075	115.0	3.21835E-02	-.469	175.0	5.65915E-02	.003
60.0	7.87179E-02	-.096	120.0	2.96902E-02	-.443	-0	-6.47658-261	-0

VI. FORTRAM Listing of JUPITOR-1

The following is the listing of the FORTRAN program JUPITOR-1, as of May, 1967.

The FORTRAN cards of this program (including the data cards of Sec. IV) are available upon request of the author. It is hoped, however, that such a request is discriminated, since the duplication and shipping of these cards involves labor and expense. When the reader has a copy and is requested from somebody else to make a copy, he is quite welcome to do so.

```
      PROGRAM JP1MAIN
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      ******       COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES    *****
      COMMON       EXTRA4(12190)
      COMMON       AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1             NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2             NPOLST
CCCCCC      ******       DIMENSION FIELDS USED ONLY IN THIS ROUTINE       ********
      DIMENSION    MENSBT(20,3),OCCUPA(20,3),EULER(3,3)
      EQUIVALENCE (EXTRA4(    1),MENSBT),(EXTRA4( 61),OCCUPA),
     1             (EXTRA4( 121),EULER)
      DIMENSION    IIOT(6),IIBB(6),KPBB(6)
      TYPE INTEGER ISBB,IIBB,KPBB,AMPMWR
      EQUIVALENCE (AR,ISBB),(AR(2),IIBB),(AR(8),KPBB),(AR(14),IIOT)
CCCCCC      IF KTRL(1)=0 TARGET IS OF EVEN A, IF=1 TARGET IS OF ODD A.
CCCCCC      IF KTRL(3)=1 REACTION AND TOTAL CROSS SECTIONS ARE COMPUTED.
CCCCCC      IF KTRL(3)=2 DIFFERENTIAL CROSS SECTIONS ARE NOT COMPUTED.
CCCCCC      IF KTRL(4)=1 DIAGONAL ELEMENTS OF ALPHA*ALPHA TERM ARE ADDED.
CCCCCC      IF KTRL(7)=1 ADIABATIC ASSUMPTION IS MADE.
CCCCCC      IF KTRL(8)=1 BOTH PROJECTILE AND TARGET ARE INITIALLY POLARIZED.
CCCCCC      IF KTRL(8)=2 ONLY THE PROJECTILE IS INITIALLY POLARIZED.
CCCCCC      IF KTRL(8)=3 ONLY THE TARGET IS INITIALLY POLARIZED.
CCCCCC      IF KTRL(9)=1 RANGES OF JJJ AND K ARE RESTRICTED BY KEXCOM(5,6,7).
CCCCCC      IF KTRL(9)=2 RANGES OF JJJ AND K ARE RESTRICTED BY KEXCOM(5,6).
CCCCCC      IF KTRL(11)=1 EXPANSION OF V IN LEGENDRE POLINOMIAL IS MADE.
CCCCCC      IF KTRL(12)=1 COMPLEX COUPLING IS TAKEN.
CCCCCC      IF KTRL(13)=1 COULOMB EXCITATION IS ADDED.
CCCCCC      IF KTRL(17)=1 ELASTIC XSEC IS DIVIDED BY RUTHERFORD XSEC.
CCCCCC      IF KTRL(19)=1 SPIN-SPIN INTERACTION IS CONSIDERED.
CCCCCC      IF KTRL(27)=1 IT IS ASSUMED THAT C-MARICES WERE ALREADY COMPUTED.
CCCCCC      KEXCOM(2)=NXMAX=NXCPLE IF NON-ZERO (CCCTRL,SN-181)
CCCCCC      KEXCOM(4)=INCREMENT OF NXMAX. (CCCTRL).
CCCCCC      KEXCOM(5,6,7)=JMIN,JMAX,KPARITY USED WHEN KTRL(9) IS NON-ZERO.
CCCCCC      KEXCOM(12)=SPIN OF TWO-PHONON CORE STATE (INTYPE=5).
CCCCCC      KEXCOM(13)=NUMBER OF THREE-PHONON STATES (INTYPE=3).
CCCCCC      KEXCOM(14)=LLMAX IN NLJJJK.
CCCCCC      KEXCOM(21-27) GIVE NUMBER OF STATES IN THE GROUND BAND,AND
CCCCCC      K=0+,0-,2+,1-,2-,3- BANDS.
CCCCCC      KEXCOM(28)=KIREPT (BMATRX,AMATRX)
CCCCCC      KEXCOM(41)=KBOUND  (C.F.COUPLE)
CCCCCC      KEXCOM(42)=NBOUND  (C.F.COUPLE)
CCCCCC      KEXCOM(43)=MXROWI  (C.F.COUPLE)
CCCCCC      KEXCOM(45,46)=JJ,K IN NLJJJK.
      TTR=(1.0,0.0)  $  TTI=(0.0,1.0)  $  ZERO=(0.0,0.0)
  701 FORMAT(14I5)
```

```
  702 FORMAT(10F7.2)
  703 FORMAT(3(2I5,F10.3))
  704 FORMAT(I5,4F7.2)
  705 FORMAT(10F7.3,I2)
 4000 FORMAT(1H1)
 4100 FORMAT(1H0)
      FACLOG(1)=FACLOG(2)=0.0  $  FN=1.0
      DO 10 N=3,500  $  FN=FN+1.0
   10 FACLOG(N)=FACLOG(N-1)+LOGF(FN)
  100 READ(5,701),(KTRL(N),N=1,28),(KEXCOM(N),N=1,14),(KTLOUT(N),N=1,28)CARD 1-5
      READ(5,702),(EXTCOM(N),N=1,10)                                    CARD   6
      READ(5,701),IICPLE,INTYPE,INTMAX,ISTRTW,NANGLR,IIXCAL,IIXPLT,     CARD   7
     1            IIPCAL,IIPPLT,KANGRD
      IF(INTYPE-6) 103,102,103
  102 READ(5,701),(KEXCOM(N),N=21,27)                                   CARD 7.1
  103 IIRMAX=IICPLE*(1-KTRL(7))+IIXCAL*KTRL(7)
      READ(5,703),(IIREAD(N),KPRITR(N),QVALUE(N),N=1,IIRMAX)            CARD   8
  105 READ(5,702),ELAB,PMAS,TMAS,CHARGE,XMES1,XMES2,AMUPMU              CARD   9
      IF(KANGRD) 112,111,112
  111 READ(5,702),(ANGLER(N),N=1,NANGLR)  $  GO TO 115                  CARD  10
  112 READ(5,702),ANGMIN,ANGDEL  $  ANG=ANGMIN                          CARD  10
      DO 113 N=1,NANGLR  $  ANGLER(N)=ANG
  113 ANG=ANG+ANGDEL
  115 DO 117 I1=1,IIXPLT
  117 READ(5,702),(SGMEXP(N,I1),N=1,NANGLR)                             CARD10.1
      DO 119 I1=1,IIPPLT
  119 READ(5,702),(POLEXP(N,I1),N=1,NANGLR)                             CARD10.1
  128 IF(KTRL(8)) 132,131,132
  131 NFAI=1  $  FAI(1)=0.0  $  GO TO 137
  132 READ(5,704),NFAI,(FAI(N),N=1,2)                                   CARD  11
      DO 135 N=1,2
  135 FAI(N)=FAI(N)*0.0174532925
  137 DO 138 N=1,NANGLR
  138 THETA(N)=ANGLER(N)*0.0174532925
      NPOLST=1
      MULSPN=ISTRTW+1  $  FMLSPN=MULSPN  $  SQRTSP=1.0/SQRTF(FMLSPN)
      MULT1Z=IIREAD(1)+IIREAD(1)+1-KTRL(1)
      FSQ=MULT1Z  $  FSQ=1.0/FSQ  $  FSQINV=SQRTF(FSQ)
      DO 155 N1=1,3  $  DO 155 M1=1,3  $  DO 145 M2=1,3
  145 AMS(M1,M2,N1)=0.0
  155 AMS(M1,M1,N1)=SQRTSP
      DO 160 N1=1,3  $  DO 160 M1=1,MULT1Z  $  DO 157 M2=1,MULT1Z
  157 BMTR(M1,M2,N1)=BMTI(M1,M2,N1)=0.0
  160 BMTR (M1,M1,N1)=FSQINV
      DO 180 N=1,3  $  NENSBP(N)=MULSPN  $  NENSBT(N)=MULT1Z
      DO 180 NE2R=1,MULT1Z
      MENSBT(NE2R,N)=2*(-IIREAD(1)+NE2R)-1
  180 OCCUPA(NE2R,N)=FSQ
      IF(KTRL(8)) 190,400,190
  190 READ(5,701),NPOLST,(NENSBP(N1),NENSBT(N1),N1=1,3)                 CARD  12
      IF(KTRL(8)-3) 195,210,195
  195 DO 197 N3=1,NPOLST
      READ(5,702),((AMS(N1,N2,N3),N2=1,MULSPN),N1=1,MULSPN)             CARD  13
  197 CONTINUE
  210 IF(KTRL(8)-2) 215,400,215
  215 DO 290 NPS=1,NPOLST
      N1MAX=NENSBT(NPS)  $  N2MAX=MULT1Z
      READ(5,701),(MENSBT(N1,NPS),N1=1,N1MAX)                           CARD  14
      READ(5,702),(OCCUPA(N1,NPS),N1=1,N1MAX)                           CARD  15
      READ(5,702),(EULER (N1,NPS),N1=1,3)                               CARD  16
      DO 225 N1=1,N1MAX
  225 OCCUPA(N1,NPS)=SQRTF(OCCUPA(N1,NPS))
```

```
      J1=IIREAD(1)
      DO 260 NE2R=1,N1MAX  $  M1=MENSBT(NE2R,NPS)
      EUL1=EULER (1,NPS)*0.0087266463
      EUL2=EULER (2,NPS)*0.0087266463
      EUL3=EULER (3,NPS)*0.0087266463
      COBETA=COSF(EUL2)  $  SIBETA=SINF(EUL2)
      DO 255 N1D=1,MULT1Z
      N1=-IIREAD(1)+N1D  $  FN1=N1  $  FM1=M1
      ARG13A=EUL1*(FM1-0.5)+EUL3*(FN1-0.5)
      COS13A=COSF(ARG13A) $ SIN13A=SINF(-ARG13A) $ JPM=J1+M1 $ JPN=J1+N1
      JMM=J1-M1+1  $  JMN=J1-N1+1  $  MMN=M1-N1+1 $ NTMIN=XMAXOF(1,MMN)
      NTMAX=XMINOF(JPM,JMN)  $  BMTFC=0.0  $  IF(NTMIN-NTMAX)230,230,247
  230 NCOPOW=J1+J1+MMN  $  NSIPOW=MMN+1
      DCOMFC=EXPF(0.5*(FACLOG(JPM)+FACLOG(JPN)+FACLOG(JMM)+FACLOG(JMN)))
      SUM=0.0
      DO 245 NT=NTMIN,NTMAX
      NTTRUE=NT-1 $ NTTW=NT+NT $ K1=JPM-NTTRUE $ K2=JMN-NTTRUE
      K3=NT  $  K4=NT-MMN+1
      NCPW=NCOPOW-NTTW  $  NSPW=NTTW-NSIPOW
      COFC=COBETA**NCPW  $  SIFC=SIBETA**NSPW
      IF(COBETA) 234,232,234
  232 IF(NCPW) 234,233,234
  233 COFC=1.0
  234 IF(SIBETA) 237,235,237
  235 IF(NSPW) 237,236,237
  236 SIFC=1.0
  237 TERM=((-1.)**NTTRUE)*COFC*SIFC/
     1          EXPF(FACLOG(K1)+FACLOG(K2)+FACLOG(K3)+FACLOG(K4))
  245 SUM=SUM+TERM
      BMTFC=DCOMFC*SUM*OCCUPA(NE2R,NPS)
  247 BMTR(NE2R,N1D,NPS)=BMTFC*COS13A
      BMTI(NE2R,N1D,NPS)=BMTFC*SIN13A
  255 CONTINUE
  260 CONTINUE
      DO 262 N1=1,N1MAX
      OCCUPA(N1,NPS)=OCCUPA(N1,NPS)**2
  262 MENSBT(N1,NPS)=MENSBT(N1,NPS)*2-1
  290 CONTINUE
  400 READ(5,702),VSX,WSX,WSF,VSO,(WC(N),N=1,6)                          CARD  17
      READ(5,702),DFN,DFNW,DFNS,DFNSP,RZERO,RZEROW,RZEROS,RZROSP,RZEROC CARD  18
      READ(5,705),(VCOUPL(N),N=1,10),KREAD1                              CARD  19
      IF(INTYPE-3) 500,410,500
  410 IF(KEXCOM(13)) 420,500,420
  420 READ(5,702),(VCOUPL(N),N=11,20)                                    CARD19.1
  500 WRITE(6,502)
  502 FORMAT(1H1,//////////)
      WRITE(6,503)
  503 FORMAT(31X,    10H**********,6X,27HCOUPLED CHANNEL CALCULATION,5X,
     110H**********)
      WRITE(6,505)
  505 FORMAT(1H0,48X,22H(ON PROGRAM JUPITOR-1))
      WRITE(6,510),ELAB,CHARGE,TMAS,PMAS
  510 FORMAT(////21X,5HELAB=F6.3,10H,  CHARGE=F5.1,8H,  TMAS=F7.3,8H,  P
     1MAS=F6.3)
      WRITE(6,513),VSX,WSX,WSF,VSO,DFN,DFNW,DFNS,DFNSP,
     1              RZERO,RZEROW,RZEROS,RZROSP,RZEROC,(WC(N),N=1,6)
  513 FORMAT(///  21X,15HVSX,WSX,WSF,VSO,18X,1H=,4F8.3/21X,19HDFN,DFNW,DF
     1NS,DFNSP,14X,1H=4F8.3/21X,34HRZERO,RZEROW,RZEROS,RZROSP,RZEROC=5F8
     2.3/21X,2HWC,31X,1H=6F8.3)
      ISOE=ISTRTW-2*(ISTRTW/2)  $  IF(ISOE) 516,514,516
  514 ISTROT=ISTRTW/2  $  ISBB=2H    $  GO TO 517
  516 ISTROT=ISTRTW    $  ISBB=2H/2
```

```
517 WRITE(6,518),ISTROT,ISBB
518 FORMAT(1H0,20X,16HPROJECTILE SPIN=I1,A2)
    WRITE(6,520),IIRMAX,INTYPE,INTMAX,(VCOUPL(N),N=1,10)
520 FORMAT(1H0,20X,7HIIRMAX=I1,8H,INTYPE=I1,8H,INTMAX=I1,6H,BETA=10F6.
   13)
    IF(INTYPE-4) 530,523,530
523 IF(KEXCOM(13)) 524,530,524
524 WRITE(6,527),(VCOUPL(N),N=11,20)
527 FORMAT(48X,5HBETA=10F6.3)
530 DO 538 I1=1,IIRMAX  $  IF(KTRL(1)) 533,532,533
532 IIOT(I1)=IIREAD(I1)      $  IIBB(I1)=2H    $  GO TO 534
533 IIOT(I1)=IIREAD(I1)*2-1  $  IIBB(I1)=2H/2
534 IF(KPRITR(I1)-1) 536,535,536
535 KPBB(I1)=3H(+)  $  GO TO 538
536 KPBB(I1)=3H(-)
538 CONTINUE
    WRITE(6,540),(IIOT(I),IIBB(I),KPBB(I),QVALUE(I),I=1,IIRMAX)
540 FORMAT(1H0,20X,14HTARGET STATES,3(2X,I1,A2,A3,4H AT F6.3,5H MEV,))
    IF(INTYPE-6) 555,550,555
550 WRITE(6,561),(KEXCOM(N),N=21,27)
555 WRITE(6,4100)  $  WRITE(6,562),(KTRL(N),N=1,28)
    WRITE(6,563),(KEXCOM(N),N=1,14) $ WRITE(6,564),(KTLOUT(N),N=1,28)
    WRITE(6,565),(EXTCOM(N),N=1,10)
    AMPMWR=3HAMU  $  IF(AMUPMU) 557,558,557
557 AMPMWR=3HPMU
558 WRITE(6,566),XMES1,XMES2,AMPMWR
561 FORMAT(/21X,31HNO. OF STATES IN VARIOUS BANDS=7I5)
562 FORMAT(///7H KTRL  ,28I4)
563 FORMAT(7H KEXCOM,28I4)
564 FORMAT(7H KTLOUT,28I4)
565 FORMAT(7H EXTCOM,10F10.2)
566 FORMAT(7H XMES1=F7.5,3X,7H XMES2=F7.5,3X,6H UNIT=A3)
    IF(KTRL(8)) 605,601,605
601 IF(KTLOUT(1)) 605,690,605
605 WRITE(6,610)
610 FORMAT(1H1,20X,5H*****,5X,21H INITIAL POLARIZATION,5X,5H*****)
    DO 680 NPS=1,NPOLST
    WRITE(6,620),NPS
620 FORMAT(///20X,9H AMS FOR I1,22H-TH POLARIZATION STATE//)
    NAMAX=NENSBP(NPS)  $  N1MAX=NENSBT(NPS)
    DO 640 N1=1,NAMAX
    WRITE(6,635),N1,(AMS(N1,N2,NPS),N2=1,MULSPN)
635 FORMAT(10X,5H NE1=I1,5X,3F10.5)
640 CONTINUE
645 WRITE(6,650),(EULER (N1,NPS),N1=1,3)
650 FORMAT(///12X,2HMN,3X,6HWEIGHT,10X,3HBMT,5X,3H***,2X,15H EULER ANG
   1LES =3F7.2//)
    DO 670 N1=1,N1MAX
    DO 663 N2=1,MULT1Z
    BR(N2)=BMTR(N1,N2,NPS)
663 BI(N2)=BMTI(N1,N2,NPS)
    WRITE(6,665),N1,MENSBT(N1,NPS),OCCUPA(N1,NPS),(BR(N2),N2=1,MULT1Z)
665 FORMAT(5H NE2=I2,2X,I3,2H/2,F9.4,2X,10F9.4)
    WRITE(6,667),(BI(N2),N2=1,MULT1Z)
667 FORMAT(25X,10F9.4)
670 CONTINUE
680 CONTINUE
690 WRITE(6,4000)
    CALL CCCTRL
    GO TO (100,105,400),KREAD1
    END JP1MAIN
```

```
      SUBROUTINE CCCTRL
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      *****     COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON      FC(70,6),FDC(70,6),GC(70,6),GDC(70,6)
      DIMENSION   FCRA(14,5,6),FDCRA(14,5,6),GCRA(14,5,6),GDCRA(14,5,6)
      EQUIVALENCE (FC,FCRA),(FDC,FDCRA),(GC,GCRA),(GDC,GDCRA)
      DX=XMES2
      DO 130 II=1,IIXCAL
  130 IIMULT(II)=2*IIREAD(II)+1-KTRL(1)
      RMAS=PMAS*TMAS/(PMAS+TMAS)  $  ECM(1)=ELAB*RMAS/PMAS
      ETUNIT=0.15745400*(1.0-AMUPMU)+0.15805086*AMUPMU
      WNUNIT=0.21870660*(1.0-AMUPMU)+0.21953760*AMUPMU
      CFUNIT=1.0/(RMAS*WNUNIT*WNUNIT)
      DO 150 II=1,IICPLE
      ECM(II)=ECM(1)-QVALUE(II)  $  E1=ABSF(ECM(II))
      CE   (II)=ETUNIT*CHARGE*SQRTF(RMAS/E1)
      WN   (II)=WNUNIT*SQRTF(RMAS*E1)
      WNINI(II)=WNUNIT*SQRTF(RMAS*(ECM(II)+VSX))
      ETA=CE(II)
      IF(ETA-10.0) 131,133,133
  131 ETA2=ETA*ETA  $  ETA2A=2.0*ETA  $  ETA6=ETA2+16.0
      SIGMA0=-(ETA/(12.*ETA6))*(1.+(ETA2-48.)/(30.*(ETA6**2))+((ETA2
     1-160.)*ETA2+1280.)/(105.*(ETA6**4)))-ETA+(ETA/2.)*LOGF(ETA6)
     2+3.5*ATANF(0.25*ETA)-(ATANF(ETA)+ATANF(0.5*ETA)+ATANF(ETA/3.))
      GO TO 135
  133 EINV1=1.0/ETA  $  EINV2=EINV1*EINV1  $  EINV3=EINV1*EINV2
      EINV5=EINV3*EINV2  $  EINV7=EINV5*EINV2  $  EINV9=EINV7*EINV2
      SIGMA0=0.7853981634+ETA*LOGF(ETA)-ETA
     1      -(0.08333333333*EINV1+0.00277777777*EINV3
     2        +0.00079365079*EINV5+0.00059523810*EINV7
     3        +0.00084175084*EINV9)
  135 MODTPI=SIGMA0/6.2831853072
      FMODTP=MODTPI
      SGMAZZ(II)=SIGMA0-6.2831853072*FMODTP
  150 CONTINUE
      NDFMES=1  $  TX=2.0*XMES2
      DO 165 N=1,10
      TX=0.5*TX
      IF(TX-0.00001-XMES1) 167,167,165
  165 NDFMES=NDFMES+1
  167 XMES1=TX
      A1=TMAS**0.333333333333  $  XBAR=RZERO*A1
      XCPLR =XBAR+10.*DFN  $  XCPLW=RZEROW*A1+10.*DFNW
      XCPLS =RZEROS*A1+10.*DFNS  $  XCPLSP=RZROSP*A1+10.*DFNSP
```

```
      XCPLE=MAX1F(XCPLR,XCPLW,XCPLS,XCPLSP)
      NXMAX=XCPLE/DX
      IF(KEXCOM(2)) 181,182,181
  181 NXMAX=KEXCOM(2)
  182 NXCPLE=NXMAX
      IF(KTRL(13)) 183,185,183
  183 NXMAX=NXMAX+KEXCOM(4)
  185 FNXMAX=NXMAX  $  XMAX=DX*FNXMAX  $  KEX14=KEXCOM(14)
      JJJMAX=(1-KTRL(7))*(KEX14+(ISTRTW+2*IIREAD(1)-KTRL(1))/2)
     1       +KTRL(7)*(KEX14+(ISTRTW/2))
      WRITE(6,263),(ECM(I1),I1=1,IICPLE)
      WRITE(6,265),(WN(I1),I1=1,IICPLE)
      WRITE(6,267),(WNINI(I1),I1=1,IICPLE)
      WRITE(6,269),(CE(I1),I1=1,IICPLE)
      WRITE(6,271),(SGMAZZ(I1),I1=1,IICPLE)
      WRITE(6,273),JJJMAX,KEXCOM(14),NXCPLE,NXMAX,XMAX,XBAR
  263 FORMAT(7H ECM    6E15.5)
  265 FORMAT(7H WN     6E15.5)
  267 FORMAT(7H WNINI 6E15.5)
  269 FORMAT(7H ETA    6E15.5)
  271 FORMAT(7H SIGMO 6E15.5)
  273 FORMAT(27HOJJJMAX,KEX14,NXCPLE,NXMAX=4I5,11H XMAX,XBAR=2E15.7)
      KEX14=LMAX=KEXCOM(14)  $  LMAXM1=LMAX-1
      KBOUND=0
      DO 290 I1=1,IICPLE
      IF(ECM(I1)) 282,290,290
  282 KBOUND=1
  290 CONTINUE
      NFGMX=4*KBOUND+1  $  X=XMAX+2*KBOUND*XMES2
      DO 390 NFG=1,NFGMX
      DO 380 I1=1,IICPLE
      ETA=CE(I1)  $  SIGMAZ=SGMAZZ(I1)
      RHOMX=X*WN(I1)  $  RD=DX*WN(I1)  $  IF(ECM(I1)) 343,341,341
  341 CALL FLGLCH  $  GO TO 345
  343 CALL FLGLNG
  345 IF(KBOUND) 361,351,361
  351 DO 355 L=1,LMAX
      FC(L,I1)=F(L)  $  GC(L,I1)=G(L)  $  FDC(L,I1)=FD(L)
  355 GDC(L,I1)=GD(L)
      GO TO 380
  361 DO 365 L=1,LMAX
      GCRA(L,NFG,I1)=G(L)  $  GDCRA(L,NFG,I1)=GD(L)
  365 FCRA(L,NFG,I1)=F(L)  $  FDCRA(L,NFG,I1)=FD(L)
  380 CONTINUE
      X=X-XMES2
  390 CONTINUE
      DO 510 I1=1,IICPLE  $  ETA=CE(I1)  $  ETASQ=ETA*ETA
      SG=SGMAZZ(I1)  $  EXSGRI(1,I1)=COSF(SG)+TTI*SINF(SG)  $  FL=1.0
      DO 510 L=1,LMAXM1
      DENOM=SQRTF(1.0/(ETASQ+FL*FL))
      EXSGRI(L+1,I1)=(FL+TTI*ETA)*EXSGRI(L,I1)*DENOM
  510 FL=FL+1.0
 4060 FORMAT(4(1X,1H(,C(E13.6,E14.6),1H)))
      IF(KTLOUT(6)) 4110,4190,4110
 4110 DO 4120 I1=1,IICPLE  $  WRITE(6,4115),I1
 4115 FORMAT(15HOEXSGRI FOR I1=I1)
      WRITE(6,4060),(EXSGRI(L,I1),L=1,LMAX)
 4120 CONTINUE
 4190 CONTINUE
      IF(KTRL(27)) 560,580,560
  560 REWIND 7  $  REWIND 8  $  GO TO 900
  580 CALL POTENT
```

```
      REWIND 7  $  JMIN=1  $  JMAX=JJJMAX  $  IF(KTRL(9)) 601,605,601
  601 JMIN=KEXCOM(5)  $  JMAX=KEXCOM(6)
  605 DO 800 JJ=JMIN,JMAX
      KEXCOM(45)=JJ
      DO 790 K=1,2  $  IF(KTRL(9)-1) 608,606,608
  606 IF(K-KEXCOM(7)) 790,608,790
  608 KEXCOM(46)=K  $  CALL NLJJJK  $  IF(MXROW) 790,790,720
  720 REWIND 8
      CALL COUPLE
      IF(KTLOUT(3)) 725,790,725
  725 J1CHEK=JJROW(1)-2*(JJROW(1)/2)  $  J1CKDV=2-J1CHEK
      DO 750 N=1,MXROW  $  LLROW(N)=LLROW(N)/2
  750 JJROW(N)=JJROW(N)/J1CKDV
      WRITE(6,761),(NNROW(N),N=1,MXROW)
      WRITE(6,762),(LLROW(N),N=1,MXROW)
      IF(J1CHEK) 756,754,756
  754 WRITE(6,763),(JJROW(N),N=1,MXROW)   $  GO TO 770
  756 WRITE(6,765),(JJROW(N),N=1,MXROW)
  761 FORMAT(3HON=,23I5)
  762 FORMAT(3H L=,23I5)
  763 FORMAT(3H J=,23I5)
  765 FORMAT(4H J= ,23(I3,2H/2))
  770 DO 775 N=1,MXROW  $  LLROW(N)=LLROW(N)*2
  775 JJROW(N)=JJROW(N)*J1CKDV
  790 CONTINUE
  800 CONTINUE
      IF(KTRL(9)) 1000,900,1000
  900 CALL XSEC
 1000 RETURN
      END
```

```
      SUBROUTINE NLJJJK
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON      ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1            ECM(10),
     2            IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3            NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4            WN(10),WNINI(10),WC(10)
      COMMON      ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1            IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2            AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3            PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4            RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5            XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
      KTRL1T=KTRL(1)*(1-KTRL(7))  $  JJ=KEXCOM(45)  $  K=KEXCOM(46)
      KT1PIS=KTRL1T+ISTRTW  $  KTISCK=KT1PIS-2*(KT1PIS/2)
      JJTRTW=2*JJ-2+KTISCK  $  NROW=0
      DO 600 I1=1,IICPLE  $  IF(KTRL(7)) 210,230,210
  210 K1=K+1  $  J1TWMI=JJTRTW+2  $  J1TWMX=2*JJJMAX+2-KTISCK $GO TO 250
  230 K1=K+KPRITR(I1)  $  IITRTW=2*IIREAD(I1)-KTRL(1)
      J1TWMI=XABSF(IITRTW-JJTRTW)+2  $  J1TWMX=IITRTW+JJTRTW+2
  250 KPCHEK=K1-2*(K1/2)
      DO 600 J1TW=J1TWMI,J1TWMX,2 $ J1TRTW=J1TWMI+J1TWMX-J1TW-2
      L1TWMI=XABSF(J1TRTW-ISTRTW)+2  $  L1TWMX=J1TRTW+ISTRTW+2
      DO 500 L1TW=L1TWMI,L1TWMX,2
      L1TRTW=L1TWMI+L1TWMX-L1TW-2  $  L1TR=L1TRTW/2
      L1PRCK=L1TR-2*(L1TR/2)  $  IF(L1PRCK-KPCHEK) 500,410,500
  410 IF(L1TR-(KEXCOM(14)-1)) 420,420,500
  420 NROW=NROW+1 $ NNROW(NROW)=I1 $ LLROW(NROW)=L1TRTW
      JJROW(NROW)=J1TRTW
  500 CONTINUE
  600 CONTINUE
      MXROW=NROW
      IF(NNROW(1)-1) 610,620,610
  610 MXROW=0
  620 RETURN
      END
```

```
      SUBROUTINE FLGLCH
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES   ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
CCCCCC      ******      DIMENSION FIELD USED ONLY IN THIS ROUTINE      *********
      COMMON EXTRA3(6070),FBAR(200),GT(75,2),T1MEMO(100),WRONSK(75)
CCCCCC      ******      CALCULATION OF G      ******
      ETATW=ETA+ETA  $  ETASQ=ETA*ETA  $  LMAX1=LMAX+1  $  KTRL(15)=0
      DO 200 L=1,LMAX1
  200 WRONSK(L)=0.0
      EEPS1=0.000001  $  EEPS2=0.000001  $  EEPS3=0.00002
      COREC1=1.0/(10.0**10)  $  SQ=SQRTF(1.0+ETASQ)
      M=1  $  RHOMXG=RHOMX+20.0*RD  $  NINC=0
  250 SL1=1.0 $ TL1=0.0 $ SC1=0.0 $ TC1=1.0-ETA/RHOMXG
      SL=SL1 $  TL=TL1  $  SC=SC1  $  TC=TC1
      IF(ABSF(ETA)-EEPS1) 263,263,255
  255 DO 260 N=1,25  $ T1=N $ T2=2.0*T1-1.0 $ T3=T1*(T1-1.0)
      DENOM=2.0*RHOMXG*T1 $ C1=(ETA*T2)/DENOM $ C2=(ETASQ-T3)/DENOM
      SL2=C1*SL1-C2*TL1  $  TL2=C1*TL1+C2*SL1
      SC2=C1*SC1-C2*TC1-SL2/RHOMXG  $  TC2=C1*TC1+C2*SC1-TL2/RHOMXG
      SL=SL+SL2  $  TL=TL+TL2  $  SC=SC+SC2  $  TC=TC+TC2
      SL1=SL2  $  TL1=TL2  $  SC1=SC2  $  TC1=TC2
  260 CONTINUE
  263 IF(M-1) 265,265,310
  265 IF(ABSF(SL*TC-SC*TL-1.0)-EEPS1)310,270,270
  270 NINC=NINC+1  $  IF(NINC-50) 275,280,280
  275 RHOMXG=RHOMXG+20.0*RD  $  GO TO 250
  280 KTRL(15)=1  $  GO TO 900
  310 ARG=SIGMAZ+RHOMXG-ETA*LOGF(2.0*RHOMXG)
      SINE=SINF(ARG)  $  COSI=COSF(ARG)
      GT(1,M)=SL*COSI-TL*SINE $ GDT=SC*COSI-TC*SINE $ T1=SQ
      GT(2,M)=((ETA+1.0/RHOMXG)*GT(1,M)-GDT)/T1
      IF(M-1) 320,320,330
  320 M=M+1  $  RHOMXG=RHOMXG+RD  $  GO TO 250
CCCCCC      ******      INWARD SOLUTION OF DIFFERENTIAL EQUATION      ******
  330 RHOMXG=RHOMXG-RD
      IF(ABSF(RHOMXG-RHOMX)-0.0001) 335,335,340
  335 G(1)=GT(1,1)  $  G(2)=GT(2,1)  $  GO TO 360
  340 RDSQ=RD*RD $ RDSQ56=0.83333333*RDSQ $ RDSQ12=0.1*RDSQ56
      NMAX=(RHOMXG-RHOMX)/RD+0.1
      DO 350 L=1,2
      RHOTM=RHOMXG  $  FL=L*(L-1)  $  PSI1=GT(L,2)  $  PSI2=GT(L,1)
      R1=RHOTM+RD  $  R2=RHOTM  $  R3=RHOTM-RD
      DO 345 N=1,NMAX
      DENOM=1.0-RDSQ12*(FL/(R3*R3)+ETATW/R3-1.0)
      FAC1=(1.0-RDSQ12*(FL/(R1*R1)+ETATW/R1-1.0))/DENOM
      FAC2=(2.0+RDSQ56*(FL/(R2*R2)+ETATW/R2-1.0))/DENOM
      PSI3=FAC2*PSI2-FAC1*PSI1 $  PSI1=PSI2  $  PSI2=PSI3
      R1=R2  $  R2=R3  $  R3=R3-RD
  345 CONTINUE
      G(L)=PSI3
  350 CONTINUE
  360 T1=SQ
      DO 370 L=1,LMAX1  $  FL=L  $  FL1=FL+1.0
      T2=SQRTF(ETASQ+FL1*FL1)  $  FAC1=1.0/(FL*T2) $ FAC2=(FL+FL1)*ETA
      FAC3=(FL+FL1)*FL*FL1  $  FAC4=FL1*T1
      G(L+2)=FAC1*((FAC2+(FAC3/RHOMX))*G(L+1)-FAC4*G(L))  $  T1=T2
  370 CONTINUE
CCCCCC      ******      CALCULATION OF F      ******
```

```
      IDEC=31
  420 I=LMAX+IDEC   $   FBAR(I)=1.0/(10.0**25)   $   FBAR(I+1)=0.0
      IMAX=I-1   $   FL=I   $   FLSQ=FL*FL   $   T1=SQRTF(FLSQ+ETASQ)
  430 DO 500 I=1,IMAX    $   L=IMAX+1-I   $   FL=L   $   FL1=FL+1.0
      FLSQ=FL*FL   $   T2=SQRTF(FLSQ+ETASQ)
      FBAR(L)=((FL+FL1)*(ETA+FL*FL1/RHOMX)*FBAR(L+1)-FL*T1*FBAR(L+2))
     1          /(FL1*T2)
      FBARC1=FBAR(L) $ FBARC2=ABSF(FBARC1) $ IF(L-LMAX) 460,440,435
  435 IF(FBARC2-10.0**15) 460,445,445
  440 IF(FBARC2-COREC1) 460,445,445
  445 COREC2=COREC1/FBARC1   $   FBAR(L)=COREC2*FBAR(L)
      FBAR(L+1)=COREC2*FBAR(L+1)
  460 T1=T2
  500 CONTINUE
      ALPHAI=(FBAR(1)*G(2)-FBAR(2)*G(1))*SQ
      IF(ALPHAI) 520,510,520
  510 KTRL(15)=4   $   GO TO 900
  520 ALPHA=1.0/ALPHAI
      DO 530 L=1,LMAX1
  530 FBAR(L)=ALPHA*FBAR(L)   $   IF(IDEC-31) 535,540,535
  535 IF(FBAR(1)) 538,536,538
  536 IF(ABSF(F(1))-0.000001) 560,560,540
  538 IF(ABSF(F(1)/FBAR(1)-1.0)-EEPS2)560,560,540
  540 DO 545 L=1,LMAX1
  545 F(L)=FBAR(L)
      IDCINC=1
  550 IDEC=IDEC+9
      IF(IDEC+LMAX1-100) 420,420,555
  555 KTRL(15)=5   $   GO TO 900
  560 DO 561 L=1,LMAX1
  561 T1MEMO(L)=0.0
      DO 570 L=1,LMAX1   $   L1=L   $   IF(FBAR(L)) 565,562,565
  562 IF(ABSF(F(L))-0.00001) 570,570,564
  564 KTRL(15)=6   $   GO TO 900
  565 T1=ABSF(F(L)/FBAR(L)-1.0)   $   T1MEMO(L)=T1
      IF(T1-EEPS2) 570,570,566
  566 L1=0   $   GO TO 572
  570 CONTINUE
  572 CONTINUE
      IF(L1) 574,573,574
  573 IDCINC=2   $   GO TO 550
  574 T1=SQ
      DO 590 L=1,LMAX   $   FL=L   $   FL1=FL+1.0   $ T2=SQRTF(FL1*FL1+ETASQ)
      TS=FL/T1
      TEST=ABSF(F(L)*G(L+1)-F(L+1)*G(L)-TS)
      WRONSK(L)=TEST   $   IF(TEST-EEPS3) 580,580,577
  577 KTRL(15)=6   $   GO TO 900
  580 T1=T2
  590 CONTINUE
      DO 600 L=1,LMAX    $   FL=L   $   FLSQ=FL*FL
      FAC1=ETA/FL+FL/RHOMX   $   FAC2=SQRTF(ETASQ+FLSQ)/FL
      FD(L)=FAC1*F(L)-FAC2*F(L+1)   $   GD(L)=FAC1*G(L)-FAC2*G(L+1)
  600 CONTINUE
      IF(KTLOUT(5)) 760,1000,760
  760 WRITE(6,765)
  765 FORMAT(10HOIN FLGLCH)
      WRITE(6,770),ETA,SIGMAZ,RHOMX,RHOMXG,RD,LMAX,NINC,NMAX
  770 FORMAT(28H ETA,SIGMAZ,RHOMX,RHOMXG,RD=5E14.6/16H LMAX,NINC,NMAX=3I
     15)
      WRITE(6,781),(F (L),L=1,LMAX1)   $   WRITE(6,782),(G (L),L=1,LMAX1)
      WRITE(6,783),(FD(L),L=1,LMAX )   $   WRITE(6,784),(GD(L),L=1,LMAX )
      WRITE(6,785),(WRONSK(L),L=1,LMAX)
```

```
  781 FORMAT(3H F ,7E15.7)
  782 FORMAT(3H G ,7E15.7)
  783 FORMAT(3H FD,7E15.7)
  784 FORMAT(3H GD,7E15.7)
  785 FORMAT(3H WR,7E15.7)
      GO TO 1000
  900 WRITE(6,920),KTRL(15)
  920 FORMAT(18HOIN FLGL KTRL(15)=I2)
 1000 RETURN
      END
```

```
      SUBROUTINE FLGLNG
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      LMX=LMAX+2
      R=RHOMX  $  RI=1.0/R  $  EX=EXPF(-R)
      G(1)=EX*RI  $  G(2)=EX*(RI+RI*RI)
      FL=1.0
      DO 50 L=1,LMX
      TWELP1=2.0*FL+1.0
      G(L+2)=TWELP1*RI*G(L+1)+G(L)
      GD(L)=-G(L+1)+RI*G(L)
      F(L)=FD(L)=0.0
   50 FL=FL+1.0
      GD(1)=-G(2)
      IF(KTLOUT(5)) 4100,4190,4100
 4100 WRITE(6,4110),R,RI,EX
 4110 FORMAT(19H0IN FLGLNG R,RI,EX=3E15.7)
      WRITE(6,4120),(G (L),L=1,LMX)
      WRITE(6,4130),(GD(L),L=1,LMX)
 4120 FORMAT(4H G =,7E15.7)
 4130 FORMAT(4H GD=,7E15.7)
 4190 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE POTENT
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      *****      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON      FC(70,6),FDC(70,6),GC(70,6),GDC(70,6)
      COMMON       VCENTR(200),VCENTI(200),VSPIN (200),VCOULM(200),
     1             VCPL1R(200),VCPL1I(200),VCPL2R(200),VCPL2I(200),
     2             VCPD1R(200),VCPD1I(200),VCPD2R(200),VCPD2I(200),
     3             VCPD3R(200),VCPD3I(200)
CCCCCC      ******      DIMENSION FIELDS USED ONLY IN THIS ROUTINE      **********
      COMMON     PFORM1(4),PFORM2(4),PFORM3(4),PFORM4(4),PFORM5(4),
     1           VLAMD1(3,3),VLAMD2(3,3),VTERM1(3,3),VTERM2(3,3)
     2          ,XMEM(300)
      TYPE INTEGER VWRITE
      NTTLMS=NXCPLE+4*(NDFMES-1)
      VSPFC=2.0*VSO/DFNS
      XBFAC=TMAS**0.333333333  $  XBARW=RZEROW*XBFAC
      XBARS=RZEROS*XBFAC  $  XBARC=RZEROC*XBFAC
      XBARSP=RZROSP*XBFAC
      VCLFC2=1.4398650*CHARGE  $  VCLFC1=VCLFC2*0.5/XBARC
      VFAC1=-VSX*XBAR/DFN   $  WFAC1=-WSX*XBARW/DFNW
      WDFC1=-4.0*WSF*XBARS/DFNS
      VFAC2=VFAC1*XBAR*0.5/DFN  $  WFAC2=WFAC1*XBARW*0.5/DFNW
      WDFC2=WDFC1*XBARS*0.5/DFNS
      IF(KTRL(11)) 257,258,257
  257 MESIZE=31  $  LAMAX=3  $  FN=30.0
      DCOSI=1.0/FN   $  DCOSI3=DCOSI/3.0
      BETA5=0.6307831*VCOUPL(1)  $  BETA9=0.846284367*VCOUPL(2)
      ALFC0= 3.5449078*DCOSI3  $  ALFC2= 7.9266549*DCOSI3
      ALFC4=10.6347234*DCOSI3  $  ALFC6=12.7813476*DCOSI3
  258 DO 410 ND=1,NDFMES
      FND=2.0**(ND-1)
      DX=XMES1*FND
      IF(ND-1) 263,259,263
  259 X=0.0  $  NXIMIN=1  $  NXIMAX=8
      IF(NDFMES-1) 265,260,265
  260 NXIMAX=NXCPLE+2  $  GO TO 265
  263 NXIMIN=NXIMAX+1  $  NXIMAX=NXIMAX+4
      IF(ND-NDFMES) 265,264,265
  264 NXIMAX=NXMAX2=NTTLMS+2
  265 DO 400 NX=NXIMIN,NXIMAX
      X=X+DX  $  XMEM(NX)=X
      PFORM1(1)=EXPF((X-XBAR )/DFN )
      PFORM1(2)=EXPF((X-XBARW)/DFNW)
      PFORM1(3)=EXPF((X-XBARS)/DFNS)
```

```
      PFORM1(4)=EXPF((X-XBARSP)/DFNSP)
      DO 267 N=1,4
      PFORM2(N)=1.0/(1.0+PFORM1(N))
      PFORM3(N)=PFORM1(N)*PFORM2(N)*PFORM2(N)
      PFORM4(N)=(PFORM1(N)-1.0)*PFORM2(N)*PFORM3(N)
      PFORM5(N)=(1.0-PFORM1(N)*(4.0-PFORM1(N)))*PFORM2(N)
     1          *PFORM2(N)*PFORM3(N)
  267 CONTINUE
      VCENTR(NX)=-VSX*PFORM2(1)
      VCENTI(NX)=-WSX*PFORM2(2)-4.0*WSF*PFORM3(3)
      VSPIN(NX)=(VSPFC*PFORM3(4))/X
      IF(X-XBARC)271,271,272
  271 VCOULM(NX)=VCLFC1*(3.0-((X/XBARC)**2))
      GO TO 275
  272 VCOULM(NX)=VCLFC2/X
  275 VCPL1R(NX)=VFAC1*PFORM3(1)  $  VCPL2R(NX)=VFAC2*PFORM4(1)
      VCPL1I(NX)=WFAC1*PFORM3(2)+WDFC1*PFORM4(3)
      VCPL2I(NX)=WFAC2*PFORM4(2)+WDFC2*PFORM5(3)
      IF(KTRL(11)) 285,400,285
  285 DO 287 LA=1,LAMAX  $  DO 287 N=1,3
      VLAMD1(N,LA)=0.0  $  VLAMD2(N,LA)=0.0
  287 CONTINUE
      IIPA=1  $  COSI=-DCOSI
      DO 300 ME=1,MESIZE  $  COSI=COSI+DCOSI
      COSI2=COSI*COSI $ COSI4=COSI2*COSI2 $ COSI6=COSI2*COSI4
      BR (1)=1.0  $  BR (2)=1.5*COSI2-0.5
      BR (3)=4.375*COSI4-3.75*COSI2+0.375
      BR (4)=14.4375*COSI6-19.6875*COSI4+6.5625*COSI2-0.3125
      BETAFC=1.0+BETA5*BR(2)+BETA9*BR(4)
      PFORM1(1)=EXPF((X-XBAR *BETAFC)/DFN )
      PFORM1(2)=EXPF((X-XBARW*BETAFC)/DFNW)
      PFORM1(3)=EXPF((X-XBARS*BETAFC)/DFNS)
      DO 288 N=1,3
      PFORM2(N)=1.0/(1.0+PFORM1(N))
      PFORM3(N)=PFORM1(N)*PFORM2(N)*PFORM2(N)
      PFORM4(N)=(PFORM1(N)-1.0)*PFORM2(N)*PFORM3(N)
  288 CONTINUE
      DO 289 LA=1,LAMAX
      VTERM1(1,LA)=PFORM2(1)*BR(LA)  $  VTERM2(1,LA)=PFORM3(1)*BR(LA)
      VTERM1(2,LA)=PFORM2(2)*BR(LA)  $  VTERM2(2,LA)=PFORM3(2)*BR(LA)
      VTERM1(3,LA)=PFORM3(3)*BR(LA)  $  VTERM2(3,LA)=PFORM4(3)*BR(LA)
  289 CONTINUE
      IF(ME-1) 291,295,291
  291 IF(ME-MESIZE) 292,295,292
  292 IF(IIPA) 293,294,293
  293 TERMFC=4.0  $  IIPA=0  $  GO TO 296
  294 TERMFC=2.0  $  IIPA=1  $  GO TO 296
  295 TERMFC=1.0
  296 DO 297 LA=1,LAMAX  $  DO 297 N=1,3
      VLAMD1(N,LA)=VLAMD1(N,LA)+VTERM1(N,LA)*TERMFC
      VLAMD2(N,LA)=VLAMD2(N,LA)+VTERM2(N,LA)*TERMFC
  297 CONTINUE
  300 CONTINUE
      VCENTR(NX)= -VSX*VLAMD1(1,1)*DCOSI3
      VCENTI(NX)=-(WSX*VLAMD1(2,1)+4.0*WSF*VLAMD1(3,1))*DCOSI3
      VCPL1R(NX)=-VSX*VLAMD1(1,2)*ALFC2
      VCPL2R(NX)=-VSX*VLAMD1(1,3)*ALFC4
      VCPL1I(NX)=(-WSX*VLAMD1(2,2)-4.0*WSF*VLAMD1(3,2))*ALFC2
      VCPL2I(NX)=(-WSX*VLAMD1(2,3)-4.0*WSF*VLAMD1(3,3))*ALFC4
      IF(INTYPE-6) 400,310,400
  310 VCPD1R(NX)= VFAC1*VLAMD2(1,1)*ALFC0
      VCPD2R(NX)= VFAC1*VLAMD2(1,2)*ALFC2
```

```
      VCPD3R(NX)= VFAC1*VLAMD2(1,3)*ALFC4
      VCPD1I(NX)=(WFAC1*VLAMD2(2,1)+ WDFC1*VLAMD2(3,1))*ALFC0
      VCPD2I(NX)=(WFAC1*VLAMD2(2,2)+ WDFC1*VLAMD2(3,2))*ALFC2
      VCPD3I(NX)=(WFAC1*VLAMD2(2,3)+ WDFC1*VLAMD2(3,3))*ALFC4
  400 CONTINUE
  410 CONTINUE
      IF(KTLOUT(7)) 4510,4590,4510
 4510 VWRITE=6HX          $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(XMEM  (NX),NX=1,NXMAX2)
      VWRITE=6HVCENTR   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCENTR(NX),NX=1,NXMAX2)
      VWRITE=6HVCENTI   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCENTI(NX),NX=1,NXMAX2)
      VWRITE=6HVSPIN    $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VSPIN (NX),NX=1,NXMAX2)
      VWRITE=6HVCOULM   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCOULM(NX),NX=1,NXMAX2)
      VWRITE=6HVCPL1R   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPL1R(NX),NX=1,NXMAX2)
      VWRITE=6HVCPL1I   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPL1I(NX),NX=1,NXMAX2)
      VWRITE=6HVCPL2R   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPL2R(NX),NX=1,NXMAX2)
      VWRITE=6HVCPL2I   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPL2I(NX),NX=1,NXMAX2)
      IF(INTYPE-6) 4590,4515,4590
 4515 VWRITE=6HVCPD1R   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPD1R(NX),NX=1,NXMAX2)
      VWRITE=6HVCPD1I   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPD1I(NX),NX=1,NXMAX2)
      VWRITE=6HVCPD2R   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPD2R(NX),NX=1,NXMAX2)
      VWRITE=6HVCPD2I   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPD2I(NX),NX=1,NXMAX2)
      VWRITE=6HVCPD3R   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPD3R(NX),NX=1,NXMAX2)
      VWRITE=6HVCPD3I   $  WRITE(6,4520),VWRITE
      WRITE(6,4525),(VCPD3I(NX),NX=1,NXMAX2)
 4520 FORMAT(1X,A6)
 4525 FORMAT(10E12.4)
 4590 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE COUPLE
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      *****      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON      FC(70,6),FDC(70,6),GC(70,6),GDC(70,6)
      COMMON       VCENTR(200),VCENTI(200),VSPIN (200),VCOULM(200),
     1             VCPL1R(200),VCPL1I(200),VCPL2R(200),VCPL2I(200),
     2             VCPD1R(200),VCPD1I(200),VCPD2R(200),VCPD2I(200),
     3             VCPD3R(200),VCPD3I(200)
      COMMON      AMAT1 (30,30),AMAT2 (30,30),AMATCP(60    ),
     1            AMAT1C(30,30),AMAT2C(30,30),AMATSS(30,30)
      COMMON      UCR1(30),UCI1(30),UCR2(30),UCI2(30),
     1            URFORD(30,5),UIFORD(30,5),SPFACT(30),
     2            FPRER(5,30),FPREI(5,30),FPRERM(4,30),FPREIM(4,30),
     3            UCRIM(30),UCIIM(30)
      DIMENSION   FCRA(14,5,6),FDCRA(14,5,6),GCRA(14,5,6),GDCRA(14,5,6),
     1            UCRI(30),UDRI(30)
      EQUIVALENCE (FC,FCRA),(FDC,FDCRA),(GC,GCRA),(GDC,GDCRA),
     1             (URFORD,UCRI),(UIFORD,UDRI)
      TYPE COMPLEX UCRI,UDRI,EX
      NBOUND=KBOUND=MXROWI=0
      DO 20 M1=1,MXROW
      N1=NNROW(M1)    $  IF(ECM(N1)) 12,14,14
   12 NBOUND=NBOUND+1 $  KBOUND=1
   14 IF(N1-1) 20,17,20
   17 MXROWI=MXROWI+1
   20 CONTINUE
      NXRM=XBAR/XMES2
      NXRMM=NXMAX*(1-KBOUND)+NXRM*KBOUND
      KEXCOM(41)=KBOUND  $  KEXCOM(42)=NBOUND  $  KEXCOM(43)=MXROWI
      PRE41=19./240.  $  PRE42=-2./5.  $  PRE43=97./120.
      PRE44=-11./15.  $  PRE45=299./240.
      NQPR=NQPRED=4  $  FNQPR=4.0
      KCTY2=0  $  IF(ISTRTW-2) 43,55,43
   43 IF(INTYPE-2) 55,45,55
   45 IF(IICPLE-3) 50,50,55
   50 IF(INTMAX-3) 52,55,55
   52 KCTY2=1
   55 IF(KTRL(13)) 70,810,70
   70 XBAR2=XBAR*XBAR  $  XBAR3=XBAR*XBAR2  $  XBAR4=XBAR*XBAR3
      VCOUVX=4.319595*CHARGE
      VCX2  =VCOUVX*XBAR2/5.0  $  VCX3  =VCOUVX*XBAR3/7.0
      VCXI2 =VCOUVX/(XBAR3*5.0)  $  VCXI3 =VCOUVX/(XBAR4*7.0)
      VCOUVP=VCOUVX*0.8*XBAR*DFN  $  VCX2P=VCOUVP
      VCXI2P=VCOUVP*(-0.25)/(XBAR2*XBAR3)
```

```
      MIROW2=0  $  K2CHK=0  $  MIROW3=0  $  K3CHK=0  $  MIRW2P=0
      K2CHKP=0
      DO 670 M1=1,MXROW  $  N1=NNROW(M1)  $  I1=IIREAD(N1)
      J1=JJROW(M1)  $  L1=LLROW(M1)  $  KP=KPRITR(N1)
      IF(KP-2) 624,621,624
  621 IF(K3CHK) 623,622,623
  622 K3CHK=1  $  MIROW3=M1  $  GO TO 624
  623 MXROW3=M1
  624 IF(KCTY2) 650,660,650
  650 IF(N1-2) 660,652,652
  652 IF(KP-1) 660,654,660
  654 IF(K2CHK) 658,656,658
  656 K2CHK=1  $  MIROW2=M1  $  GO TO 660
  658 MXROW2=M1
  660 IF(INTYPE-3) 670,661,670
  661 IF(N1-3) 670,663,663
  663 IF(I1-2) 670,665,670
  665 IF(K2CHKP) 669,667,669
  667 K2CHKP=1  $  MIRW2P=M1  $  GO TO 670
  669 MXRW2P=M1
  670 CONTINUE
CCCCCC      MIROW3,MXROW3=MIN,MAX OF M1 FOR WHICH STATE IS 3(-).
CCCCCC      MIROW2,MXROW2=MIN,MAX OF M1 FOR WHICH STATE IS 2(+)-1.
CCCCCC      MIRW2P,MXRW2P=MIN,MAX OF M1 FOR WHICH STATE IS 2(+)-2.
  810 REWIND 8 $  CALL BMATRX
      DO 815 M1=1,MXROW
      J1=JJROW(M1)  $  L1=LLROW(M1)
      SPFC=J1*(J1+2)-L1*(L1+2)-ISTRTW*(ISTRTW+2)
  815 SPFACT(M1)=0.250*SPFC
      INEXMX=1+KBOUND
      DO 1950 INEX=1,INEXMX
      FINEX=INEX  $  M3MAX=MXROW+MXROWI*(INEX-1)
      DO 1900 M3=1,M3MAX
      NDMAX=NDFMES*(2-INEX)+(INEX-1)
      DO 1890 ND=1,NDMAX
      FND=2.0**(ND-1)  $  DX=XMES1*FND*(2.0-FINEX)-XMES2*(FINEX-1.0)
      DR=DX*WN(1)  $  DRSQ=(DR*DR)/ECM(1)  $  K4COR2=4*ND-5
      IF(ND-1) 860,822,860
  822 DO 825 M1=1,MXROW
      UCR1(M1)=UCI1(M1)=UCR2(M1)=UCI2(M1)=0.0
      DO 825 N=1,5
      FPRER(N,M1)=FPREI(N,M1)=FPRERM(N,M1)=FPREIM(N,M1)=0.0
  825 CONTINUE
      IF(INEX-1) 925,830,925
  830 N1=NNROW(M3)  $  J1=JJROW(M3)  $  L1=LLROW(M3)
      LL=(L1/2)+1  $  WNI=WNINI(N1)
      UCR2(M3)=0.5*((2.0*DX*WNI)**LL)*EXPF(FACLOG(LL)-FACLOG(2*LL))
      IF(LL-2) 840,834,840
  834 FPRER(1,M3)=FPRERM(1,M3)=CFUNIT*0.6666666666*WNI*WNI*DRSQ
  840 X=0.0  $  NXIMIN=2  $  NXIMAX=8  $  NXPRCH=3  $  NNX=2
      IF(NDFMES-1) 1225,857,1225
  857 NXIMAX=NXRMM+2  $  NXMAX3=NXIMAX-5  $  GO TO 1225
  860 NXIMIN=5  $  X=3.0*DX
      IF(ND-NDFMES) 862,863,862
  862 NXIMAX=8  $  GO TO 865
  863 NXIMAX=NXRMM+2  $  NXMAX3=NXIMAX-5
  865 DO 875 M1=1,MXROW
      DO 870 NQ=1,NQPR
      FPRER(NQ,M1)=FPRERM(NQ,M1)*4.0
  870 FPREI(NQ,M1)=FPREIM(NQ,M1)*4.0
      FPRERM(1,M1)=FPRER(1,M1)  $  FPRERM(2,M1)=FPRER(3,M1)
      FPREIM(1,M1)=FPREI(1,M1)  $  FPREIM(2,M1)=FPREI(3,M1)
```

```
      UCR1(M1)=UCR1M(M1)  $  UCI1(M1)=UCI1M(M1)
  875 CONTINUE
      NXPRCH=5  $  NNX=3
      GO TO 1225
  925 GR1=FI1=1.0  $  GI1=FR1=0.0  $  M4=M3
      IF(M3-MXROW) 935,935,930
  930 GR1=GI1=FI1=0.0  $  FR1=1.0  $  M4=M3-MXROW
  935 N1=NNROW(M4)  $  L1=LLROW(M4)  $  J1=JJROW(M4)  $  LL=(L1/2)+1
      X=(NXMAX+3)*XMES2  $  CFFCLL=LL*(LL-1)*CFUNIT
      FAC=L1+1  $  FAC=SQRTF(FAC*((WN(1)/WN(N1))**3))
      DO 950 NQ=1,5
      X=X-XMES2
      V1=(1.4398650*CHARGE/X+CFFCLL/(X*X)-ECM(N1))*DRSQ
      FRR   =GR1*GCRA(LL,NQ,N1)+FR1*FCRA(LL,NQ,N1)
      FII   =GI1*GCRA(LL,NQ,N1)+FI1*FCRA(LL,NQ,N1)
      EX=(FRR+TTI*FII)*FAC*EXSGRT(LL,N1)
      FPRERR=EX  $  FPREII=CIMAG(EX)
      FPRER(NQ,M4)=FPRERR*V1  $  FPREI(NQ,M4)=FPREII*V1
      IF(NQ-4) 950,945,947
  945 UCR1(M4)=FPRERR  $  UCI1(M4)=FPREII  $  GO TO 950
  947 UCR2(M4)=FPRERR  $  UCI2(M4)=FPREII
  950 CONTINUE
      NXIMIN=1  $  NXIMAX=NXMAX-NXRM  $  NXMAX3=NXIMAX-5
      X=(NXMAX-1)*XMES2  $  K4COR2=NXMAX+4*NDFMES-5
 1225 DO 1800 NX=NXIMIN,NXIMAX
      N3=(NX+K4COR2)*(2-INEX)+(K4COR2-NX)*(INEX-1)
      X=X+DX  $  XX2=X*X  $  XX2INV=1.0/XX2
      IF(KTRL(I3)) 1235,1245,1235
 1235 XX3=X*XX2  $  XX4=X*XX3  $  XX3INV=1.0/XX3  $  XX4INV=1.0/XX4
 1240 IF(X-XBAR) 1241,1241,1243
 1241 VCX2X=VCXI2*XX2 $ VCX2XP=VCXI2P*XX2 $ VCX3X=VCXI3*XX3 $ GO TO 1245
 1243 VCX2X=VCX2*XX3INV  $  VCX2XP=VCX2P*XX3INV  $  VCX3X=VCX3*XX4INV
 1245 DO 1700 M1=1,MXROW
      L1=LLROW(M1) $ LL=(L1/2)+1 $ CFFCLL=(LL-1)*LL*CFUNIT $N1=NNROW(M1)
      WCOREC=WC(N1)  $  IF(NX-(NXCPLE+2)) 1310,1310,1315
 1310 AR2=VCENTR(N3)+VCOULM(N3)+CFFCLL*XX2INV
     1   -SPFACT(M1)*VSPIN(N3)-ECM(N1)
      AI2=VCENTI(N3)*WCOREC  $  GO TO 1320
 1315 AR2=CFFCLL*XX2INV+1.4398650*CHARGE/X-ECM(N1)  $  AI2=0.0
 1320 BR2=AR2*UCR2(M1)-AI2*UCI2(M1)
      BI2=AR2*UCI2(M1)+AI2*UCR2(M1)
      BR(M1)=BR2*DRSQ  $  BI(M1)=BI2*DRSQ
CCCCCC      ******      REAL COUPLING      ************************************
      IF(NX-NXCPLE+2) 1330,1330,1610
 1330 AFAC1R=DRSQ       *VCPL1R(N3)  $  AFAC2R=DRSQ       *VCPL2R(N3)
      IF(KTRL(I2)) 1510,1410,1510
 1410 M2MAX=MXROW
      IF(INTYPE-3) 1412,1475,1416
 1412 IF(KCTY2) 1413,1415,1413
 1413 IF(M1-1) 1414,1470,1414
 1414 M2MAX=1  $  GO TO 1470
 1415 IF(INTMAX-2) 1470,1470,1475
 1416 IF(INTYPE-4) 1431,1418,1431
 1418 IF(KTRL(I9)) 1485,1420,1485
 1420 IF(INTMAX-2) 1470,1475,1475
 1431 IF(INTYPE-5) 1490,1420,1490
 1470 DO 1473 M2=1,M2MAX
      AFACTR=AMATI(M1,M2)*AFAC1R
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)
      BI(M1)=BI(M1)+AFACTR*UCI2(M2)
 1473 CONTINUE  $  GO TO 1610
 1475 DO 1477 M2=1,MXROW
```

```
      AFACTR=AMAT1(M1,M2)*AFAC1R+AMAT2(M1,M2)*AFAC2R
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)
      BI(M1)=BI(M1)+AFACTR*UCI2(M2)
 1477 CONTINUE  $  GO TO 1610
 1485 AFAC3R=DRSQ       *( VCOUPL(9)/VSX)*VCENTR(N3)
      DO 1487 M2=1,MXROW
      AFACTR=AMAT1(M1,M2)*AFAC1R+AMAT2(M1,M2)*AFAC2R
     1                          +AMATSS(M1,M2)*AFAC3R
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)  $  BI(M1)=BI(M1)+AFACTR*UCI2(M2)
 1487 CONTINUE   $  GO TO 1610
 1490 AFAC3R=DRSQ       *VCPD1R(N3)
      AFAC4R=DRSQ       *VCPD2R(N3)  $  AFAC5R=DRSQ       *VCPD3R(N3)
      DO 1492 M2=1,MXROW
      AFACTR=AMAT1 (M1,M2)*AFAC1R+AMAT2 (M1,M2)*AFAC2R
     1       +AMAT1C(M1,M2)*AFAC3R+AMAT2C(M1,M2)*AFAC4R
     2       +AMATSS(M1,M2)*AFAC5R
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)  $  BI(M1)=BI(M1)+AFACTR*UCI2(M2)
 1492 CONTINUE  $  GO TO 1610
CCCCCC       ******       COMPLEX COUPLING.              ***************************
 1510 AFAC1I=DRSQ       *VCPL1I(N3)  $  AFAC2I=DRSQ       *VCPL2I(N3)
      M2MAX=MXROW
      IF(INTYPE-3) 1512,1575,1516
 1512 IF(KCTY2) 1513,1515,1513
 1513 IF(M1-1) 1514,1570,1514
 1514 M2MAX=1  $  GO TO 1570
 1515 IF(INTMAX-2) 1570,1570,1575
 1516 IF(INTYPE-4) 1531,1520,1531
 1520 IF(INTMAX-2) 1570,1575,1575
 1531 IF(INTYPE-5) 1590,1520,1590
 1570 DO 1573 M2=1,M2MAX
      AFACTR=AMAT1(M1,M2)*AFAC1R  $  AFACTI=AMAT1(M1,M2)*AFAC1I
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)-AFACTI*UCI2(M2)
      BI(M1)=BI(M1)+AFACTR*UCI2(M2)+AFACTI*UCR2(M2)
 1573 CONTINUE   $  GO TO 1610
 1575 DO 1577 M2=1,MXROW
      AFACTR=AMAT1(M1,M2)*AFAC1R+AMAT2(M1,M2)*AFAC2R
      AFACTI=AMAT1(M1,M2)*AFAC1I+AMAT2(M1,M2)*AFAC2I
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)-AFACTI*UCI2(M2)
      BI(M1)=BI(M1)+AFACTR*UCI2(M2)+AFACTI*UCR2(M2)
 1577 CONTINUE  $  GO TO 1610
 1590 AFAC3R=DRSQ       *VCPD1R(N3)
      AFAC4R=DRSQ       *VCPD2R(N3)
      AFAC5R=DRSQ       *VCPD3R(N3)
      AFAC3I=DRSQ       *VCPD1I(N3)
      AFAC4I=DRSQ       *VCPD2I(N3)
      AFAC5I=DRSQ       *VCPD3I(N3)
      DO 1592 M2=1,MXROW
      AFACTR=AMAT1 (M1,M2)*AFAC1R+AMAT2 (M1,M2)*AFAC2R
     1       +AMAT1C(M1,M2)*AFAC3R+AMAT2C(M1,M2)*AFAC4R
     2       +AMATSS(M1,M2)*AFAC5R
      AFACTI=AMAT1 (M1,M2)*AFAC1I+AMAT2 (M1,M2)*AFAC2I
     1       +AMAT1C(M1,M2)*AFAC3I+AMAT2C(M1,M2)*AFAC4I
     2       +AMATSS(M1,M2)*AFAC5I
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)-AFACTI*UCI2(M2)
      BI(M1)=BI(M1)+AFACTR*UCI2(M2)+AFACTI*UCR2(M2)
 1592 CONTINUE
CCCCCC       ******       COULOMB EXCITATION.        ********************************
 1610 IF(KTRL(13)) 1620,1700,1620
 1620 IF(KCTY2) 1622,1637,1622
 1622 IF(MIROW2) 1624,1645,1624
 1624 IF(M1-1) 1628,1626,1628
 1626 M2MIN=MIROW2  $  M2MAX=MXROW2  $  GO TO 1640
```

```
 1628 IF(M1-MIROW2) 1645,1633,1633
 1633 IF(M1-MXROW2) 1635,1635,1645
 1635 M2MIN=1 $  M2MAX=1  $  GO TO 1640
 1637 M2MIN=1  $  M2MAX=MXROW
CCCCCC       ******       L=2 MULTIPLE COULOMB COUPLING.      *******
 1640 AFAC1R=DRSQ       *VCX2X
      DO 1643 M2=M2MIN,M2MAX
      AFACTR=AMATIC(M1,M2)*AFAC1R
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)
      BI(M1)=BI(M1)+AFACTR*UCI2(M2)
 1643 CONTINUE
CCCCCC       ******       L=2 DIRECT COULOMB COUPLING.       ******
 1645 IF(MIRW2P) 1650,1665,1650
 1650 IF(M1-1) 1655,1651,1655
 1651 AFAC1R=DRSQ      *VCX2XP
      DO 1653 M2=MIRW2P,MXRW2P
      AFACTR=AMATCP(M2)*AFAC1R
      BR( 1)=BR( 1)+AFACTR*UCR2(M2)
 1653 BI( 1)=BI( 1)+AFACTR*UCI2(M2)
      GO TO 1665
 1655 IF(M1-MIRW2P) 1665,1657,1657
 1657 IF(M1-MXRW2P) 1659,1659,1665
 1659 AFAC1R=DRSQ       *VCX2XP
      AFACTR=AMATCP(M1)*AFAC1R
      BR(M1)=BR(M1)+AFACTR*UCR2(1)
      BI(M1)=BI(M1)+AFACTR*UCI2(1)
 1665 I1=I1
CCCCCC       ******       L=3 COULOMB COUPLING.       ************
      IF(MIROW3) 1670,1700,1670
 1670 IF(M1-1) 1674,1672,1674
 1672 M2MIN=MIROW3  $  M2MAX=MXROW3  $  GO TO 1680
 1674 IF(M1-MIROW3) 1700,1676,1676
 1676 IF(M1-MXROW3) 1678,1678,1700
 1678 M2MIN=1  $  M2MAX=1
 1680 AFAC1R=DRSQ       *VCX3X
      DO 1685 M2=M2MIN,M2MAX
      AFACTR=AMAT2C(M1,M2)*AFAC1R
      BR(M1)=BR(M1)+AFACTR*UCR2(M2)
      BI(M1)=BI(M1)+AFACTR*UCI2(M2)
 1685 CONTINUE
 1700 CONTINUE
      IF(KTLOUT(IO)-2)4190,4110,4190
 4110 IF(LLROW(1)-2) 4120,4120,4190
 4120 WRITE(6,4125) NX,X,(UCR2(M),UCI2(M),UCR1(M),UCI1(M),M=1,2)
 4125 FORMAT(4H NX=I3,3H X=F8.4,8E12.4)
 4190 CONTINUE
      DO 1740 M1=1,MXROW
      IF(INEX-1) 1714,1711,1714
 1711 IF(ND-1) 1714,1712,1714
 1712 IF(NX-4) 1713,1713,1714
 1713 FPRER(NX,M1)=TERMR=BR(M1)  $  FPREI(NX,M1)=TERMI=BI(M1)
      GO TO 1715
 1714 FPRER(5,M1)=BR(M1)  $  FPREI(5,M1)=BI(M1)
      TERMR =PRE41*FPRER (1,M1)+PRE42*FPRER (2,M1)+PRE43*FPRER (3,M1)
     1       +PRE44*FPRER (4,M1)+PRE45*FPRER (5,M1)
      TERMI =PRE41*FPREI (1,M1)+PRE42*FPREI (2,M1)+PRE43*FPREI (3,M1)
     1       +PRE44*FPREI (4,M1)+PRE45*FPREI (5,M1)
 1715 ARR=2.0*UCR2(M1)-UCR1(M1)+TERMR
      AII=2.0*UCI2(M1)-UCI1(M1)+TERMI
      UCR1(M1)=UCR2(M1)  $  UCR2(M1)=ARR
      UCI1(M1)=UCI2(M1)  $  UCI2(M1)=AII
      IF(INEX-1) 1735,1717,1735
```

```
 1717 IF(ND-NDFMES) 1718,1729,1718
 1718 IF(NX-NXPRCH) 1729,1719,1729
 1719 FPRERM(NNX,M1)=BR(M1)  $  FPREIM(NNX,M1)=BI(M1)
      IF(M1-MXROW) 1725,1723,1725
 1723 NXPRCH=NXPRCH+2  $  NNX=NNX+1
 1725 IF(NX-(NXIMAX-1)) 1729,1728,1729
 1728 UCR1M(M1)=UCR1(M1)  $  UCI1M(M1)=UCI1(M1)
 1729 IF(ND-1) 1735,1730,1735
 1730 IF(NX-4) 1740,1740,1735
 1735 DO 1738 NQ=1,NQPR
      FPRER(NQ,M1)=FPRER(NQ+1,M1)
 1738 FPREI(NQ,M1)=FPREI(NQ+1,M1)
 1740 CONTINUE
      IF(ND-NDMAX ) 1800,1760,1800
 1760 NXCH2=NX-NXMAX3  $  IF(NXCH2)1800,1800,1765
 1765 DO 1770 M1=1,MXROW
      URFORD(M1,NXCH2)=UCR2(M1)  $  UIFORD(M1,NXCH2)=UCI2(M1)
 1770 CONTINUE
 1800 CONTINUE
CCCCCC      ******      CALCULATION OF THE DERIVATIVES      ******
      IF(ND-NDMAX ) 1890,1805,1890
 1805 DO 1850 M1=1,MXROW
      UCR1(M1)=URFORD(M1,3)  $  UCI1(M1)=UIFORD(M1,3)
      N1=NNROW(M1)  $  DENOM=1.0/(12.0*DX*WN(N1))
      UCR2(M1)   =(8.0*(URFORD(M1,4)-URFORD(M1,2))
     1                -(URFORD(M1,5)-URFORD(M1,1)))*DENOM
      UCI2(M1)   =(8.0*(UIFORD(M1,4)-UIFORD(M1,2))
     1                -(UIFORD(M1,5)-UIFORD(M1,1)))*DENOM
 1850 CONTINUE
 1890 CONTINUE
      IF(KBOUND) 1892,1897,1892
 1892 DO 1895 M1=1,MXROW
      UCRI(M1)=UCR1(M1)+TTI*UCI1(M1)
 1895 UDRI(M1)=UCR2(M1)+TTI*UCI2(M1)
      WRITE TAPE 8,(UCRI(M1),UDRI(M1),M1=1,MXROW)
      GO TO 1900
 1897 WRITE TAPE 8,(UCR1(M1),UCI1(M1),UCR2(M1),UCI2(M1),M1=1,MXROW)
 1900 CONTINUE
 1950 CONTINUE
      CALL SMATRX
 2000 RETURN
      END
```

```
      SUBROUTINE BMATRX
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      *****      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON      FC(70,6),FDC(70,6),GC(70,6),GDC(70,6)
      COMMON       VCENTR(200),VCENTI(200),VSPIN (200),VCOULM(200),
     1             VCPL1R(200),VCPL1I(200),VCPL2R(200),VCPL2I(200),
     2             VCPD1R(200),VCPD1I(200),VCPD2R(200),VCPD2I(200),
     3             VCPD3R(200),VCPD3I(200)
      COMMON      AMAT1 (30,30),AMAT2 (30,30),AMATCP(60    ),
     1            AMAT1C(30,30),AMAT2C(30,30),AMATSS(30,30),
     2            B(10,10,10),KII(24),ALP23P(7,3),ALSQ3P(7,3),
     3            IITYPE(10),KQNOTW(7)
      JJ=KEXCOM(45)  $ KTRL1T=KTRL(1)*(1-KTRL(7))
      KTIPIS=KTRL1T+ISTRTW  $  KTISCK=KTIPIS-2*(KTIPIS/2)
      JJTRTW=2*JJ-2+KTISCK
      DO 50 N1=1,6  $  DO 50 N2=1,6  $  DO 50 K1=1,10
   50 B(N1,N2,K1)=0.0
      DO 60 M=1,MXROW
      DO 55 N=1,MXROW
   55 AMAT1(M,N)=AMAT2(M,N)=AMAT1C(M,N)=AMAT2C(M,N)=AMATSS(M,N)=0.0
   60 AMATCP(M)=0.0
      IF(KTRL(7)) 1000,70,1000
   70 COMFAC=0.2820948 $  KII(1)=IIREAD(2)
      B(1,2,1)=VCOUPL(1)*COMFAC
      GO TO (1000,210,310,410,510,610), INTYPE
  210 KII(2)=3
      B(1,3,2)=VCOUPL(2)*COMFAC  $  IF(INTMAX-3) 1000,232,232
  232 HATISQ=IIREAD(4)*2+1  $  HATI=SQRTF(HATISQ)
      B(3,4,1)=VCOUPL(3)*COMFAC*HATI*0.4472136
      B(2,4,2)=VCOUPL(4)*COMFAC*HATI*0.3779645*((-1.0)**IIREAD(4))
      VCPL23=VCOUPL(3)*VCOUPL(4)*COMFAC*0.2820948
      KII(3)=1  $  KII(4)=3  $  KII(5)=5  $  IA=4  $  IB=6  $  ID=0
      IE=0  $  IF=0
      DO 240 K1=3,5
      IC=KII(K1)*2  $  CALL CLEB  $  B(2,3,K1)=VCPL23*RAC
      IF(KII(K1)-IIREAD(4)) 240,235,240
  235 B(1,4,K1)=VCPL23*RAC
  240 CONTINUE
      GO TO 1000
  310 COMFC2=COMFAC/2.5066282 $  IIRMX1=IICPLE-KEXCOM(13)
      NTPHST=IIRMX1-2  $  KII(2)=0  $  KII(3)=2  $  KII(4)=4
      KII(5)=0  $  KII(6)=2  $  KII(7)=4
      DO 340 N1=3,IIRMX1
      II=IIREAD(N1) $ N2=II+2 $ N3=N2+1 $ N4=(II/2)+2 $ HAT=II+II+1
```

```
      HAT=SQRTF(HAT)
      B(2,N1,I)=COMFAC*VCOUPL(N2)*HAT*0.6324555
      N22=(I1/2)+8 $  N44=N22-3
      B(1,N1,N44)=COMFAC*VCOUPL(N22)
      I1M2=I1-2  $  IF(I1M2)331,333,335
  331 CLEB1=0.4472136  $  GO TO 337
  333 CLEB1=-0.53452243  $  GO TO 337
  335 CLEB1=0.71713714
  337 VC12=VCOUPL(N3)
      B(1,N1,N4)=(VC12**2)*COMFC2*CLEB1
  340 CONTINUE
      INTMAX=7 $  IF(KTRL(4)) 351,375,351
  351 ALFADG=VCOUPL(1)**2  $  ALFADG=ALFADG*COMFC2
      B(1,1,2)=ALFADG*0.7070107  $  B(2,2,2)=ALFADG*2.2135946
      B(2,2,3)=ALFADG*0.7559290  $  B(2,2,4)=ALFADG*1.01418507
      DO 370 K1=2,4
      K1M1=K1-1  $  GO TO (353,355,357),K1M1
  353 FKFAC=1.118034  $  FKHAT=1.0  $  IF=0  $  GO TO 360
  355 FKFAC=0.59761437  $  FKHAT=2.236068  $  IF=4  $  GO TO 360
  357 FKFAC=0.59761437  $  FKHAT=3.0  $  IF=8
  360 IA=4  $  IC=4  $  IE=4
      DO  370 N1=1,NTPHST  $  I1=IIREAD(N1+2)  $  IB=I1+I1
      HAT1SQ=IB+1 $  HAT1=SQRTF(0.2*HAT1SQ)
      DO  370 N2=N1,NTPHST  $  IF(N1-N2) 369,365,369
  365 IF(K1-2) 369,367,369
  367 B(N1+2,N2+2,2)=HAT1
  369 I2=IIREAD(N2+2) $ ID=I2+I2 $ HAT2SQ=ID+1 $ HAT2=SQRTF(3.2*HAT2SQ)
      CALL RAC7  $  SNDTRM=HAT1*HAT2*FKHAT*RAC
      B(N1+2,N2+2,K1)=FKFAC*VCOUPL(2*N1+1)*VCOUPL(2*N2+1)
     1         *(B(N1+2,N2+2,K1)+SNDTRM)
  370 CONTINUE
  375 IF(KEXCOM(13)) 377,1000,377
  377 SQRT10=0.31622773  $  SQRT2 =0.70710656
      C1FAC=COMFAC*SQRT10*VCOUPL(I1)$ C2FAC=COMFC2*SQRT2*(VCOUPL(I2)**2)
      DO 379 K1=1,3  $  DO 379 J1=1,7
      ALP23P(J1,K1)=0.0  $  ALSQ3P(J1,K1)=0.0
  379 CONTINUE
      ALP23P(1,2)= 2.449490
      ALP23P(3,1)= 3.741657
      ALP23P(3,2)= 2.3904574
      ALP23P(3,3)= 3.2071351
      ALP23P(4,2)=-5.477226
      ALP23P(4,3)= 3.464102
      ALP23P(5,2)= 5.318433
      ALP23P(5,3)= 5.070927
      ALP23P(7,3)= 8.8311761
      ALSQ3P(1,2)=-0.58554009
      ALSQ3P(3,1)= 1.67331986
      ALSQ3P(3,2)= 0.57142857
      ALSQ3P(3,3)= 0.76665167
      ALSQ3P(4,2)= 0.43643575
      ALSQ3P(4,3)= 0.27602623
      ALSQ3P(5,2)=-1.2713481
      ALSQ3P(5,3)= 1.2121830
      ALSQ3P(7,3)= 2.1111953
      N3P=IIRMX1+1  $  I3PP1=IIREAD(N3P)+1
      DO 380 N1=1,NTPHST $ N2P=N1+2 $ I2PP=(IIREAD(N2P)/2)+1
      B(N2P,N3P,1)=C1FAC*ALP23P(I3PP1,I2PP)
  380 CONTINUE
      DO 385 K1=1,3
      B(2,N3P,K1+1)=C2FAC*ALSQ3P(I3PP1,K1)
  385 CONTINUE
```

```
      GO TO 1000
  410 ROTFAC=0.2820948  $  KII(1)=2  $  KII(2)=4
      ID=IF=2*IIREAD(1)-KTRL(1)  $  IE=0
      DO 450 N3=1,INTMAX
      IF(KTRL(19)) 415,420,415
  415 IF(N3-INTMAX) 420,435,420
  420 IB=2*KII(N3)
      DO 430 N1= 1,IICPLE  $  I1=IIREAD(N1)  $  IC=2*I1-KTRL(1)
      DO 430 N2=N1,IICPLE  $  I2=IIREAD(N2)  $  IA=2*I2-KTRL(1)
      HATA=IA+1  $  HATA=SQRTF(HATA)
      CALL CLEB
  430 B(N1,N2,N3)=ROTFAC*HATA*RAC
      GO TO 450
  435 DO 440 N1=1,IICPLE
      ITW=IIREAD(N1)*2-1  $  ELSQ=ITW*(ITW+1)*(ITW+2)
      ELSQ=SQRTF(ELSQ)
  440 B(N1,N2,N3)=0.5*ELSQ*VCOUPL(3)
  450 CONTINUE
      GO TO 1000
  510 IF(INTMAX-1) 555,515,555
  515 KII(1)=2+XABSF(KPRITR(2)-KPRITR(1))  $  I1=IIREAD(1)
      DO 520 N2=2,IICPLE
      I2=IIREAD(N2)  $  HATI2=2*I2
      HATFC=SQRTF(HATI2/5.0)*((-1.0)**(I2-I1))
  520 B(1,N2,1)=COMFAC*HATFC*VCOUPL(N2-1)
      GO TO 1000
  555 N1PHST=IICPLE-1  $  IIGRND=IIREAD(1)
      IRTRTW=KEXCOM(I2)*2  $  HATR=IRTRTW+1
      KII(1)=2  $  KII(2)=KEXCOM(12)
      A3=VCOUPL(7)  $  B3=VCOUPL(8)*SQRTF(1.0-A3*A3)
      A5=VCOUPL(9)  $ B5=VCOUPL(10)*SQRTF(1.0-A5*A5)
      COMFC2=COMFAC/2.5066282
      DO 585 N1=2,IICPLE  $  II1=IIREAD(N1)  $  HAT1=II1*2
      HATI=(III/2)*2*2+1
      IF(N1-N1PHST) 561,561,564
  561 IF(II1-2) 563,562,563
  562 A1=A3  $  B1=B3  $  GO TO 569
  563 A1=A5  $  B1=B5  $  GO TO 569
  564 IF(KEXCOM(12)/2-1) 565,566,565
  565 A1=0.  $  B1=1.  $  GO TO 569
  566 IF(II1-2) 568,567,568
  567 A1=-B3*VCOUPL(4)  $  B1=A3  $  GO TO 569
  568 A1=-B5*VCOUPL(4)  $  B1=A5
  569 S1=(-1.0)**(III-IIGRND)
      B(1,N1,1)=COMFAC*S1*SQRTF(HAT1/HATI)*VCOUPL(1)*A1
      IA=IB=4  $  IC=(III/2)*2*2  $  ID=IE=IF=0  $  CALL CLEB  $ C1=RAC
      B(1,N1,2)=COMFC2*S1*C1*SQRTF(HAT1/HATI)*(VCOUPL(3)**2)*B1
      DO 585 N2=N1,IICPLE  $  II2=IIREAD(N2)  $  HAT2=II2*2
      HATII=(II2/2)*2*2+1
      IF(N2-N1PHST) 571,571,574
  571 IF(II2-2) 573,572,573
  572 A2=A3  $  B2=B3  $  GO TO 579
  573 A2=A5  $  B2=B5  $  GO TO 579
  574 IF(KEXCOM(I2)/2-1) 575,576,575
  575 A2=0.  $  B2=1.  $  GO TO 579
  576 IF(II2-2) 578,577,578
  577 A2=-B3*VCOUPL(4)  $  B2=A3  $  GO TO 579
  578 A2=-B5*VCOUPL(4)  $  B2=A5
  579 IA=4  $  IB=(III/2)*2*2  $  IC=II2*2-1
      ID=IIGRND*2-1  $  IE=(II2/2)*2*2  $  IF=III*2-1
      CALL RAC7  $  R1=RAC
      GEOFAC=SQRTF(0.4*HATII*HAT1*HAT2)*((-1.0)**(III-II2))*VCOUPL(2)
```

```
      B(N1,N2,1)=COMFAC*GEOFAC*(A2*B1+A1*B2)*R1
  585 CONTINUE
      GO TO 1000
  610 ROTFAC=0.2820948  $  ROTFC2=ROTFAC*ROTFAC
      KQNOTW(1)=0  $  KQNOTW(2)=0  $  KQNOTW(3)=0  $  KQNOTW(4)=4
      KQNOTW(5)=2  $  KQNOTW(6)=4  $  KQNOTW(7)=6
      SQRT5=SQRTF(5.0)  $  SQRT7=SQRTF(7.0)
      NB=0
      DO 620 N=1,7
      IF(KEXCOM(N+20)) 615,620,615
  615 ITRANG=KEXCOM(N+20)
      DO 617 IT=1,ITRANG  $  NB=NB+1
  617 IITYPE(NB)=N
  620 CONTINUE
      DO 690 KIREPT=1,4  $  KEXCOM(28)=KIREPT
      GO TO (631,632,633,634),KIREPT
  631 INTMAX=2
      KII(1)=2  $  KII(2)=4  $  GO TO 640
  632 INTMAX=2  $  LAMDAP=0
      KII(1)=2  $  KII(2)=3  $  GO TO 640
  633 INTMAX=6  $  LAMDAP=2
      KII(1)=0  $  KII(2)=2  $  KII(3)=4
      KII(4)=1  $  KII(5)=3  $  KII(6)=5  $  GO TO 640
  634 INTMAX=7  $  LAMDAP=4
      KII(1)=2  $  KII(2)=4  $  KII(3)=6
      KII(4)=1  $  KII(5)=3  $  KII(6)=5  $  KII(7)=7
  640 DO 641 N1=1,IICPLE  $  DO 641 N2=1,IICPLE  $  DO 641 N3=1,INTMAX
  641 B(N1,N2,N3)=0.0
      IF(KIREPT-1) 643,645,643
  643 LMDPTW=LAMDAP*2  $  HATLDP=LMDPTW+1  $  HATLDP=SQRTF(HATLDP)
  645 DO 680 K1=1,INTMAX
      LAMDA=KII(K1)  $  LAMDTW=LAMDA*2
      HATLMD=LAMDTW+1  $  HATLMD=SQRTF(HATLMD)
      DO 680 N1=1,IICPLE
      I1=IIREAD(N1)  $  I1TW=I1+I1-KTRL(1)  $  IT1=IITYPE(N1)
      DO 675 N2=N1,IICPLE
      I2=IIREAD(N2)  $  I2TW=2*I2-KTRL(1)
      HATI2=I2TW+1  $  HATI2=SQRTF(HATI2)  $  IT2=IITYPE(N2)
      IF(KIREPT-1) 655,651,655
  651 IF(IT1-IT2) 675,652,675
  652 IA=I2TW  $  IB=2*KII(K1)  $  IC=I1TW
      ID=IF=KQNOTW(IT1)  $  IE=0  $  CALL CLEB
      B(N1,N2,K1)=ROTFAC*HATI2*RAC  $  GO TO 675
  655 IF(IT1-1) 675,656,675
  656 IF(IT2-1) 657,675,657
  657 IT2M1=IT2-1
      GO TO (661,662,661,662,662,662),IT2M1
  661 IA=4  $  C1=SQRT5  $  GO TO 663
  662 IA=6  $  C1=SQRT7
  663 IB=LMDPTW  $  IC=LAMDTW  $  ID=IE=IF=0  $  CALL CLEB  $  R1=RAC
      GO TO (666,666,665,665,665,665),IT2M1
  665 ID=IF=KQNOTW(IT2)  $  CALL CLEB
  666 R2=RAC
      IE=-ID  $  IF=0
      IA=I2TW  $  IB=LAMDTW  $  IC=I1TW  $  CALL CLEB  $  R3=RAC
      RCFAC=R1*R2*R3*C1*VCOUPL(IT2+1)
      ROTFC3=ROTFC2*HATI2*HATLDP/HATLMD
      B(N1,N2,K1)=ROTFC3*RCFAC
  675 CONTINUE
  680 CONTINUE
      CALL AMATRX
  690 CONTINUE
```

```
      GO TO 2000
 1000 CALL AMATRX
 2000 RETURN
      END
```

```
      SUBROUTINE AMATRX
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON      ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1            ECM(10),
     2            IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3            NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4            WN(10),WNINI(10),WC(10)
      COMMON      ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1            IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2            AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3            PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4            RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5            XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC     *****     COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON     EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON     FC(70,6),FDC(70,6),GC(70,6),GDC(70,6)
      COMMON      VCENTR(200),VCENTI(200),VSPIN (200),VCOULM(200),
     1            VCPL1R(200),VCPL1I(200),VCPL2R(200),VCPL2I(200),
     2            VCPD1R(200),VCPD1I(200),VCPD2R(200),VCPD2I(200),
     3            VCPD3R(200),VCPD3I(200)
      COMMON     AMAT1 (30,30),AMAT2 (30,30),AMATCP(60    ),
     1           AMATIC(30,30),AMAT2C(30,30),AMATSS(30,30),
     2           B(10,10,10),KII(24)
      KIREPT=KEXCOM(28)  $  JJ=KEXCOM(45)
      KTRL1T=KTRL(1)*(1-KTRL(7))  $  KT1PIS=KTRL1T+ISTRTW
      KTISCK=KT1PIS-2*(KT1PIS/2)  $  JJTRTW=2*JJ-2+KTISCK
      MJBRTW=-JJTRTW  $  IF(KTRL(7)) 910,570,910
  570 DO 900 M1=1,MXROW
      N1=NNROW(M1) $ I1=IIREAD(N1) $ J1=JJROW(M1) $ L1=LLROW(M1)
      DO 900 M2=M1,MXROW
      N2=NNROW(M2) $ I2=IIREAD(N2) $ J2=JJROW(M2) $ L2=LLROW(M2)
      KISELC=1
      DO 890 K1=1,INTMAX
      KASPSP=0  $  IF(KTRL(19)) 711,713,711
  711 IF(K1-3) 713,712,713
  712 KASPSP=1
  713 B1=B(N1,N2,K1)  $  IF(B1) 720,890,720
  720 KI=KII(K1)
      IF(INTYPE-2) 730,725,730
  725 KISELC=1
      IF(K1-3) 750,726,726
  726 KISELC=2  $  GO TO 750
  730 IF(INTYPE-3) 740,735,740
  735 KISELC=1
      GO TO (750,726,726,726,750,750,750),K1
  740 IF(INTYPE-4) 743,745,743
  743 IF(INTYPE-5) 750,745,750
  745 KISELC=K1
  750 KITRTW=KI+KI  $ IA=I1+I1-KTRL(1) $ IB=J1 $ IC=I2+I2-KTRL(1)
      ID=J2 $ IE=JJTRTW $ IF=KITRTW $ CALL RAC7 $ RAC1=RAC
      IF(KASPSP) 785,751,785
  751 IA=L1 $ IB=L2 $ IC=KITRTW $ ID=0 $ IE=0 $ IF=0 $ CALL CLEB
      RAC2=RAC
      IA=L1 $ IB=J1 $ IC=L2 $ ID=J2 $ IE=ISTRTW $ IF=KITRTW
      CALL RAC7 $  RAC3=RAC
      HAT=(L1+1)*(L2+1)*(J1+1)*(J2+1)  $  HAT=SQRTF(HAT)
```

```
      S1=(-1.0)**((JJTRTW-ISTRTW-I2*2+KTRL(1)+L1+L2)/2
     1          +((L1-L2)/2+KPRITR(NI)-KPRITR(N2))/2)
      T2=S1*HAT*RAC1*RAC2*RAC3    $  T1=T2*B1
      IF(INTYPE-6) 752,754,752
  752 GO TO ( 761,766,771,773),KISELC
  754 IF(KIREPT-1) 756,755,756
  755 IF(K1-1) 766,761,766
  756 GO TO (755,763,765,790),KIREPT
  761 AMAT1(M1,M2)=AMAT1(M1,M2)+T1
      IF(KTRL(13)) 762,890,762
  762 IF(KI-2) 890,763,764
  763 AMAT1C(M1,M2)=AMAT1C(M1,M2)+T1  $  GO TO 890
  764 IF(KI-3) 890,765,890
  765 AMAT2C(M1,M2)=AMAT2C(M1,M2)+T1  $  GO TO 890
  766 AMAT2(M1,M2)=AMAT2(M1,M2)+T1
      IF(INTYPE-3) 890,767,890
  767 IF(I2-2) 890,768,890
  768 IF(M1-1) 890,769,890
  769 AMATCP(M2)=AMATCP(M2)+T1 $  GO TO 890
  771 AMAT1C(M1,M2)=AMAT1C(M1,M2)+T1  $  GO TO 890
  773 AMAT2C(M1,M2)=AMAT2C(M1,M2)+T1  $  GO TO 890
  785 IF(L1-L2) 890,786,890
  786 IA=1 $ IC=1 $ IE=L1 $ IF=2 $ CALL RAC7  $  RAC2=RAC
      C1=(-1.0)**((JJTRTW+J1+J2+L1-ITMULT(N1))/2+1)
      HAT=6*(IB+1)*(ID+1) $ HAT=SQRTF(HAT) $ T2=RAC1*RAC2*C1*HAT
      T1=T2*B1
  790 AMATSS(M1,M2)=AMATSS(M1,M2)+T1
  890 CONTINUE
      AMAT1 (M2,M1)=AMAT1 (M1,M2)  $  AMAT2 (M2,M1)=AMAT2 (M1,M2)
      AMATSS(M2,M1)=AMATSS(M1,M2)  $  AMAT1C(M2,M1)=AMAT1C(M1,M2)
      AMAT2C(M2,M1)=AMAT2C(M1,M2)
  900 CONTINUE
      GO TO 990
  910 ROTFAC=0.2820948
      DO 930 M1=1,MXROW
      L1=LLROW(M1)  $  J1=JJROW(M1)  $  HATSQ1=(L1+1)*(J1+1)
      DO 930 M2=M1,MXROW
      L2=LLROW(M2)  $  J2=JJROW(M2)  $ HATSQ2=(L2+1)*(J2+1)
      DO 925 K1=1,INTMAX
      LAMDA=2*K1 $ LAMDTW=LAMDA*2 $ HATLMD=LAMDTW+1
      HATFAC=SQRTF(HATSQ1*HATSQ2/HATLMD)
      S1=(-1.0)**((ISTRTW+MJBRTW)/2+(L1-L2)/4)
      IA=L1 $ IB=L2 $ IC=LAMDTW $ ID=0 $ IE=0 $ IF=0 $ CALL CLEB $C1=RAC
      IC=J1 $ ID=J2 $ IE=LAMDTW $ IF=ISTRTW $ CALL RAC7 $ R1=RAC
      IA=J1 $ IB=J2 $ IC=LAMDTW $ IE=MJBRTW $ ID=-IE $ IF=0
      CALL CLEB   $  C2=RAC
      T1=ROTFAC*S1*HATFAC*C1*C2*R1  $  GO TO (921,922),K1
  921 AMAT1(M1,M2)=AMAT1(M1,M2)+T1  $  GO TO 925
  922 AMAT2(M1,M2)=AMAT2(M1,M2)+T1
  925 CONTINUE
      AMAT1(M2,M1)=AMAT1(M1,M2)  $  AMAT2(M2,M1)=AMAT2(M1,M2)
  930 CONTINUE
  990 I1=I1
 4050 FORMAT(8E15.5)
 4060 FORMAT(1H0)
      IF(KTLOUT(9)) 4210,4250,4210
 4210 IF(JJ-KEXCOM(5)) 4250,4212,4212
 4212 IF(JJ-KEXCOM(6)) 4214,4214,4250
 4214 IF(KEXCOM(46)-KEXCOM(7)) 4250,4220,4250
 4220 WRITE(6,4230),INTYPE,INTMAX,(KII(NI),NI=1,INTMAX)
 4230 FORMAT(21H0  BMATRX.FOR INTYPE=I1,8H INTMAX=I1,7H LAMDA=10I3)
      DO 4232 KI=1,INTMAX
```

```
      WRITE(6,4060)
      DO 4232 N1=1,IICPLE
      WRITE(6,4050),(B(N1,N2,K1),N2=1,IICPLE)
 4232 CONTINUE
      IF(KTLOUT(10)) 4233,4250,4233
 4233 WRITE(6,4234)
 4234 FORMAT(12H0*** AMATRIX)
      DO 4240 M1=1,MXROW
 4240 WRITE(6,4050),(AMAT1 (M1,M2),M2=1,MXROW)
      DO 4245 M1=1,MXROW
 4245 WRITE(6,4050),(AMAT2 (M1,M2),M2=1,MXROW)
      DO 4246 M1=1,MXROW
 4246 WRITE(6,4050),(AMATSS(M1,M2),M2=1,MXROW)
      DO 4247 M1=1,MXROW
 4247 WRITE(6,4050),(AMAT1C(M1,M2),M2=1,MXROW)
      DO 4249 M1=1,MXROW
 4249 WRITE(6,4050),(AMAT2C(M1,M2),M2=1,MXROW)
      WRITE(6,4050),(AMATCP(   M2),M2=1,MXROW)
 4250 I1=I1
      RETURN
      END
```

```
      SUBROUTINE SMATRX
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES   ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON      ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1            ECM(10),
     2            IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3            NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4            WN(10),WNINI(10),WC(10)
      COMMON      ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1            IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2            AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3            PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4            RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5            XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      *****      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON      FC(70,6),FDC(70,6),GC(70,6),GDC(70,6)
      COMMON      VCENTR(200),VCENTI(200),VSPIN (200),VCOULM(200),
     1            VCPL1R(200),VCPL1I(200),VCPL2R(200),VCPL2I(200),
     2            VCPD1R(200),VCPD1I(200),VCPD2R(200),VCPD2I(200),
     3            VCPD3R(200),VCPD3I(200)
      COMMON      H(6710)
      DIMENSION   URC(30,30),UIC(30,30),UDRC(30,30),UDIC(30,30),
     1            WNRATO(10),Q(50,51)
      EQUIVALENCE (H(3101),URC),(H(4001),UIC),(H(4901),UDRC),
     1            (H(5801),UDIC),(H(6701),WNRATO),(H,Q)
      DIMENSION ARI(70),BRI(70) $ EQUIVALENCE (AR,ARI),(BR,BRI)
      TYPE COMPLEX Q,ARI,BRI,EX,DIVQ,SUMRI
      KBOUND=KEXCOM(41)  $  NBOUND=KEXCOM(42)  $  MXROWI=KEXCOM(43)
      JJ=KEXCOM(45)  $  K=KEXCOM(46)
      MXROW1=MXROW+1  $  MXRBRI=MXROW+MXROW*KBOUND
      DO 105 I1=1,IICPLE
      WNR=WN(I1)/WN(1)  $  WNRATO(I1)=WNR*SQRTF(WNR)
  105 CONTINUE
      KWRIT2=0
      IF(KTLOUT(12)) 113,119,113
  113 IF(KTRL(9)) 118,114,118
  114 IF(JJ-3) 118,118,119
  118 KWRIT2=1
  119 NMATCH=(1-KBOUND)+KBOUND*MXROWI
      DO 700 NMAT=1,NMATCH
      REWIND 8
      IF(KBOUND) 125,145,125
  125 MXROW2=2*MXROW  $  IM=MXROW2+1
      DO 127 M3=1,MXROW2
  127 READ TAPE 8,(Q(M1,M3),M1=1,MXROW2)
      DO 129 M3=1,NMAT
  129 READ TAPE 8,(Q(M1,IM),M1=1,MXROW2)
      DO 130 M3=MXROW1,MXROW2
      DO 130 M1=1,MXROW2
  130 Q(M1,M3)=-Q(M1,M3)
      NLJMAX=1
      IF(KWRIT2) 4610,4690,4610
 4610 DO 4620 M1=1,MXROW2
 4620 WRITE(6,4060),(Q(M1,M3),M3=1,IM)
 4690 CONTINUE
      GO TO 170
```

```
  145 DO 150 M3=1,MXROW
      READ TAPE8,(URC(M1,M3),UIC(M1,M3),UDRC(M1,M3),UDIC(M1,M3),
     1            M1=1,MXROW)
  150 CONTINUE
      IM=MXROW1
 4050 FORMAT(8E15.6)
 4060 FORMAT(4(1X,1H(,C(E13.6,E14.6),1H)))
 4110 IF(KWRIT2) 4120,4190,4120
 4120 WRITE(6,4150)
 4150 FORMAT(8HOURC ETC)
      DO 4160 M3=1,MXROW
      WRITE(6,4050),(URC(M1,M3),UIC(M1,M3),UDRC(M1,M3),UDIC(M1,M3),
     1              M1=1,MXROW)
 4160 CONTINUE
 4190 CONTINUE
      IF(KTRL(7)) 162,165,162
  162 NLJMAX=MXROW  $  GO TO 170
  165 NLJMAX=MXROWI
  170 DO 600 NLJ=1,NLJMAX
      IF(KBOUND) 300,175,300
  175 DO 179 N=1,MXROW
      Q(N,IM)=ZERO  $  NS=NNROW(N)  $  LT=(LLROW(N)/2)+1
      FT=FC(LT,NS) $ GT=GC(LT,NS) $ FDT=FDC(LT,NS) $ GDT=GDC(LT,NS)
      DO 179 M=1,MXROW
      UR=URC(N,M) $  UI=UIC(N,M)  $  UDR=UDRC(N,M)  $  UDI=UDIC(N,M)
      QR=UR*GDT-UI*FDT-UDR*GT+UDI*FT $ QI=UR*FDT+UI*GDT-UDR*FT-UDI*GT
      Q(N,M)=QR+TTI*QI
  179 CONTINUE
      LT=(LLROW(NLJ)/2)+1  $  HAT=2*LT-1  $  FAC=-SQRTF(HAT)
      Q(NLJ,IM)=FAC*EXSGRI(LT,1)
 4210 IF(KWRIT2) 4220,4290,4220
 4220 WRITE(6,4230),NLJ,NLJMAX
 4230 FORMAT(22H0***QMATRIX.NLJ,NLJMAX,2I3)
      WRITE(6,4060),(Q(M1,IM),M1=1,MXROW)
      IF(NLJ-1) 4290,4235,4290
 4235 DO 4240 M1=1,MXROW
 4240 WRITE(6,4060),(Q(M1,M2),M2=1,MXROW)
 4290 CONTINUE
  300 CALL SIMLEQ
 4310 IF(KWRIT2) 4320,4390,4320
 4320 WRITE(6,4330)
 4330 FORMAT(7H0***AMP)
      WRITE(6,4060),(BRI(N),N=1,MXRBRI)
 4390 CONTINUE
      IF(KBOUND) 330,340,330
  330 DO 335 M1=1,MXROW
      M2=M1+MXROW
  335 BRI(M1)=BRI(M2)
      GO TO 490
  340 DO  430 N=1,MXROW
      NS=NNROW(N) $ LT=(LLROW(N)/2)+1 $ FT=FC(LT,NS) $ FDT=FDC(LT,NS)
      DO 430 M=1,MXROW
      UR=URC(N,M)  $  UI=UIC(N,M)  $  UDR=UDRC(N,M)  $  UDI=UDIC(N,M)
      QR=UR*FDT-UDR*FT  $  QI=UI*FDT-UDI*FT  $  Q(N,M)=QR+TTI*QI
  430 CONTINUE
      DO 445 N1=1,MXROW
      ARI(N1)=ZERO
  445 CONTINUE
      DO 450 N1=1,MXROW
      DO 447 M1=1,MXROW
  447 ARI(N1)=ARI(N1)+Q(N1,M1)*BRI(M1)
  450 CONTINUE
```

```
      DO 455 N1=1,MXROW
      NT=NNROW(N1)  $  LT=(LLROW(N1)/2)+1  $  HAT=LT+LT-1
      FAC=WNRATO(NT)/SQRTF(HAT)
      SUMRI=FAC/EXSGRI(LT,NT)
  455 BRI(N1)=ARI(N1)*SUMRI
  490 WRITE TAPE 7,(BRI(N),N=1,MXROW)
 4505 IF(KTLOUT(11)) 4510,4590,4510
 4510 WRITE(6,4520),JJ,K,NLJ
 4520 FORMAT(24H0**CMATRIX FOR JJ,K,NLJ=3I3)
      WRITE(6,4060),(BRI(NI),NI=1,MXROW)
 4590 CONTINUE
  600 CONTINUE
  700 CONTINUE
 1000 RETURN
      END
```

```
      SUBROUTINE SIMLEQ
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON      ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1            ECM(10),
     2            IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3            NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4            WN(10),WNINI(10),WC(10)
      COMMON      ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1            IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2            AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3            PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4            RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5            XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      *****      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON      FC(70,6),FDC(70,6),GC(70,6),GDC(70,6)
      COMMON       VCENTR(200),VCENTI(200),VSPIN (200),VCOULM(200),
     1             VCPL1R(200),VCPL1I(200),VCPL2R(200),VCPL2I(200),
     2             VCPD1R(200),VCPD1I(200),VCPD2R(200),VCPD2I(200),
     3             VCPD3R(200),VCPD3I(200)
      COMMON      Q(50,51)
      DIMENSION ARI(70),BRI(70) $ EQUIVALENCE (AR,ARI),(BR,BRI)
      TYPE COMPLEX Q,ARI,BRI,EX,DIVQ,SUMRI
      MXROWM=MXROW  $  KBOUND=KEXCOM(41)  $  MXROW=MXROW*(1+KBOUND)
      IM=MXROW+1
      DO 870 I=1,MXROW
      BIGQ=CABS(Q(I,I))  $  MXI=I
      DO 816 I1=I,MXROW
      B1=CABS(Q(I1,I))  $  IF(BIGQ-B1) 815,816,816
  815 BIGQ=B1  $  MXI=I1
  816 CONTINUE
      IF(MXI-I) 818,822,818
  818 DO 820 J=I,IM
      EX=Q(I,J)  $  Q(I,J)=Q(MXI,J)
  820 Q(MXI,J)=EX
  822 DIVQ=Q(I,I)  $  DIVQAB=CABS(DIVQ)  $  IF(DIVQAB) 840,830,840
  830 WRITE(6,835)
  835 FORMAT(17HODIVQ=0 IN SIMLEQ)
      GO TO 1000
  840 DIVQ=1.0/DIVQ
      DO 850 J=I,IM
  850 Q(I,J)=Q(I,J)*DIVQ
      I1=I+1
      IF(I1-IM) 860,880,880
  860 DO 870 K1=I1,MXROW
      EX=Q(K1,I)
      DO 870 K2=I,IM
      Q(K1,K2)=Q(K1,K2)-EX*Q(I,K2)
  870 CONTINUE
  880 BRI(MXROW)=Q(MXROW,IM)
  890 I=MXROW
      IF(MXROW-1) 892,910,892
  892 SUMRI=ZERO
      DO 894 K2=I,MXROW
      SUMRI=SUMRI+BRI(K2)*Q(I-1,K2)
  894 CONTINUE
```

```
      I=I-1
      BRI(I)=Q(I,IM)-SUMRI
      IF(I-1) 910,910,892
  910 I1=I1
 1000 MXROW=MXROWM
      RETURN
      END
```

```
      SUBROUTINE XSEC
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON      ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1            ECM(10),
     2            IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3            NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4            WN(10),WNINI(10),WC(10)
      COMMON      ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1            IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2            AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3            PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4            RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5            XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      ******     DIMENSION FIELDS USED ONLY IN THIS ROUTINE     **********
      COMMON      EXSGRI(840)
      COMMON      CMXR(10,30),CMXI(10,30),
     1            ZBART (10,10,3),ZBARS (10,10,3),ZBARP (10,10,3),
     2            ZBART1(10,10,3),ZBARS1(10,10,3),ZBARP1(10,10,3),
     3            CBAR1T(3,10,10),CBAR2T(3,10,10),CBAR3T(3,10,10),
     4            CBAR1S(3,10,10),CBAR2S(3,10,10),CBAR3S(3,10,10),
     5            CBAR1P(3,10,10),CBAR2P(3,10,10),CBAR3P(3,10,10),
     6            REACCR(3),TOTLXC(3),STRGFN(3),STRGFS(3),STRGFP(3),
     7            TOTLXS(3),TOTLXP(3)
      COMMON      EXTRA4(1729)
      TYPE COMPLEX ZBART,ZBARS,ZBARP,CBAR1T,CBAR2T,CBAR3T,CBAR1S,CBAR2S,
     1             CBAR3S,CBAR1P,CBAR2P,CBAR3P,EX,EXS,EX1,
     2             ZBART1,ZBARS1,ZBARP1
CCCCCC      *****     COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1            NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2            NPOLST
      REWIND 7  $  REWIND 8
      IF(KTRL(7)) 30,45,30
   30 CALL ACCSAB  $  GO TO 990
   45 MULT1Z=IIMULT(1)  $  ISMULT=ISTRTW+1
   60 MSREPT=(ISTRTW/2)+1  $  IF(KTRL(3)) 61,65,61
   61 DO 63 M1=1,MULT1Z  $  DO 63 M2=1,MULT1Z  $  DO 63 M3=1,ISMULT
      ZBART1(M1,M2,M3)=ZBARS1(M1,M2,M3)=ZBARP1(M1,M2,M3)=ZERO
   63 ZBART(M1,M2,M3)=ZBARS(M1,M2,M3)=ZBARP(M1,M2,M3)=ZERO
      DO 64 NPS=1,NPOLST
      TOTLXS(NPS)=TOTLXP(NPS)=0.0
   64 REACCR(NPS)=TOTLXC(NPS)=STRGFN(NPS)=STRGFS(NPS)=STRGFP(NPS)=0.0
      STGFCS=50.6606*WN(1)*WN(1)*SQRTF(0.000001/ELAB)
      PAISQ=9.8696044011  $  XSQ=(WN(1)*XBAR)**2
      C1=6.0*PAISQ*XSQ/(1.+XSQ)
      STGFCP=1000.*WN(1)*WN(1)/(C1*SQRTF(1000000.0*ELAB))
   65 DO 600 JJ=1,JJJMAX
      KTIPIS=KTRL(1)+ISTRTW  $  KTISCK=KTIPIS-2*(KTIPIS/2)
      JJTRTW=2*JJ-2+KTISCK  $  KEXCOM(45)=JJ
      DO 595 K=1,2
      KEXCOM(46)=K  $  CALL NLJJJK
      IF(MXROW) 595,595,70
   70 NRT7MX=0
      DO 75 N1=1,MXROW
      IF(NNROW(N1)-1) 75,73,80
   73 NRT7MX=NRT7MX+1
```

```
 75 CONTINUE
 80 DO 110 NRT7=1,NRT7MX
110 READ TAPE 7,(CMXR(NRT7,N1),CMXI(NRT7,N1),N1=1,MXROW)
    DO 127 NE1=1,3  $  DO 127 NE2=1,10 $  DO 127 N1=1,NRT7MX
    CBAR1T(NE1,NE2,N1)=CBAR2T(NE1,NE2,N1)=CBAR3T(NE1,NE2,N1)=ZERO
    CBARIS(NE1,NE2,N1)=CBAR2S(NE1,NE2,N1)=CBAR3S(NE1,NE2,N1)=ZERO
127 CBAR1P(NE1,NE2,N1)=CBAR2P(NE1,NE2,N1)=CBAR3P(NE1,NE2,N1)=ZERO
    DO 405 MSRE=1,MSREPT  $  MSTRTW=ISTRTW*(2-MSRE)
    DO 400 MTII=1,MULT1Z  $  MTIINV=MULT1Z+1-MTII
    MTITTW=2*MTII-1-MULT1Z  $  MJTRTW=MSTRTW+MTITTW
    DO 130 N1=1,MXROW
130 BR(N1)=BI(N1)=0.0
    IF(JJTRTW-XABSF(MJTRTW)) 250,160,160
160 DO 240 NRT7=1,NRT7MX  $  CG1=1.0  $  IF(ISTRTW) 210,212,210
210 JT=JJROW(NRT7)   $  LT=LLROW(NRT7)
    IA=LT  $  IB=ISTRTW  $  IC=JT  $  ID=0  $  IE=IF=MSTRTW
    CALL CLEB  $  CG1=RAC
212 CG2=1.0
    IF(KTRL(1)) 214,216,214
214 IA=JT  $  IB=MULT1Z-1  $  IC=JJTRTW  $  ID=MSTRTW  $  IE=MTITTW
    IF=MJTRTW  $  CALL CLEB  $  CG2=RAC
216 CG12=CG1*CG2
    DO 230 N1=1,MXROW
    BR(N1)=BR(N1)+CG12*CMXR(NRT7,N1)
230 BI(N1)=BI(N1)+CG12*CMXI(NRT7,N1)
240 CONTINUE
250 WRITE TAPE 8,(BR(N1),BI(N1),N1=1,MXROW)
    IF(KTRL(3)) 252,400,252
252 DO 395 NLJ=1,NRT7MX
    J2TRTW=JJROW(NLJ)  $  L2TRTW=LLROW(NLJ)
    TWL2P1=L2TRTW+1  $  SQRTLB=SQRTF(TWL2P1) $  SIGNIV=0.0
    IF(ISTRTW-1) 262,261,254
254 IF(MSRE-1) 262,261,262
261 SIGNIV=(-1.0)**((L2TRTW+ISTRTW-MULT1Z+1-JJTRTW)/2)
262 DO 295 MTFF=1,MULT1Z  $  MTFTTW=2*MTFF-1-MULT1Z
    MSFTTW=MJTRTW-MTFTTW  $  IF(ISTRTW-XABSF(MSFTTW)) 295,264,264
264 MSFF=(ISMULT+1-MSFTTW)/2  $  CG1=1.0  $  IF(KTRL(1)) 266,267,266
266 IA=J2TRTW  $  IB=MULT1Z-1  $  IC=JJTRTW  $  ID=MSFTTW
    IE=MTFTTW  $  IF=MJTRTW  $  CALL CLEB  $  CG1=RAC
267 CG2=1.0  $  IF(ISTRTW) 268,269,268
268 IA=L2TRTW  $  IB=ISTRTW  $  IC=J2TRTW  $  ID=0  $  IE=IF=MSFTTW
    CALL CLEB  $  CG2=RAC
269 T1=CG1*CG2*TWL2P1  $  TR1=T1*BR(NLJ)  $  TI1=T1*BI(NLJ)
    EX=TR1+TTI*TI1  $  EX1=ZERO
    IF(MSRE-1) 271,273,271
271 EX1=EX  $  EX=ZERO
273 ZBART (MTII,MTFF,MSFF)=ZBART (MTII,MTFF,MSFF)+EX
    ZBARTI(MTII,MTFF,MSFF)=ZBARTI(MTII,MTFF,MSFF)+EX1
    IF(L2TRTW-2) 276,277,295
276 ZBARS (MTII,MTFF,MSFF)=ZBARS (MTII,MTFF,MSFF)+EX
    ZBARS1(MTII,MTFF,MSFF)=ZBARS1(MTII,MTFF,MSFF)+EX1   $  GO TO 295
277 ZBARP (MTII,MTFF,MSFF)=ZBARP (MTII,MTFF,MSFF)+EX
    ZBARP1(MTII,MTFF,MSFF)=ZBARP1(MTII,MTFF,MSFF)+EX1
295 CONTINUE
    DO 370 NPS=1,NPOLST
    NENS1=NENSBP(NPS)  $  NENS2=NENSBT(NPS)
    DO 370 NE1=1,NENS1
    A2=AMS(NE1,MSRE,NPS)  $  A22=AMS(NE1,ISMULT,NPS)*SIGNIV
    DO 365 NE2=1,NENS2
    A2B2R=A2*BMTR(NE2,MTII,NPS)+A22*BMTR(NE2,MTIINV,NPS)
    A2B2I=A2*BMTI(NE2,MTII,NPS)+A22*BMTI(NE2,MTIINV,NPS)
    TR1=(A2B2R*BR(NLJ)-A2B2I*BI(NLJ))*SQRTLB
```

```
      TI1=(A2B2R*BI(NLJ)+A2B2I*BR(NLJ))*SQRTLB
      EX=TRI+TTI*TI1
      GO TO (321,322,323),NPS
  321 CBAR1T(NE1,NE2,NLJ)=CBAR1T(NE1,NE2,NLJ)+EX   $  GO TO 330
  322 CBAR2T(NE1,NE2,NLJ)=CBAR2T(NE1,NE2,NLJ)+EX   $  GO TO 330
  323 CBAR3T(NE1,NE2,NLJ)=CBAR3T(NE1,NE2,NLJ)+EX
  330 IF(L2TRTW-2) 341,346,365
  341 GO TO (342,343,344),NPS
  342 CBAR1S(NE1,NE2,NLJ)=CBAR1S(NE1,NE2,NLJ)+EX   $  GO TO 365
  343 CBAR2S(NE1,NE2,NLJ)=CBAR2S(NE1,NE2,NLJ)+EX   $  GO TO 365
  344 CBAR3S(NE1,NE2,NLJ)=CBAR3S(NE1,NE2,NLJ)+EX   $  GO TO 365
  346 GO TO (347,348,349),NPS
  347 CBAR1P(NE1,NE2,NLJ)=CBAR1P(NE1,NE2,NLJ)+EX   $  GO TO 365
  348 CBAR2P(NE1,NE2,NLJ)=CBAR2P(NE1,NE2,NLJ)+EX   $  GO TO 365
  349 CBAR3P(NE1,NE2,NLJ)=CBAR3P(NE1,NE2,NLJ)+EX
  365 CONTINUE
  370 CONTINUE
  395 CONTINUE
  400 CONTINUE
  405 CONTINUE
      IF(KTRL(3)) 505,595,505
  505 DO 520 NPS=1,NPOLST
      NENS1=NENSBP(NPS)   $   NENS2=NENSBT(NPS)
      DO 520 NE1=1,NENS1
      DO 520 NE2=1,NENS2
      DO 520 NLJ=1,NRT7MX
      GO TO (510,511,512),NPS
  510 EX =CBAR1T(NE1,NE2,NLJ)   $   EXS=CBAR1S(NE1,NE2,NLJ)
      EX1=CBAR1P(NE1,NE2,NLJ)   $   GO TO 515
  511 EX =CBAR2T(NE1,NE2,NLJ)   $   EXS=CBAR2S(NE1,NE2,NLJ)
      EX1=CBAR2P(NE1,NE2,NLJ)   $   GO TO 515
  512 EX =CBAR3T(NE1,NE2,NLJ)   $   EXS=CBAR3S(NE1,NE2,NLJ)
      EX1=CBAR3P(NE1,NE2,NLJ)
  515 STRGFN(NPS)=STRGFN(NPS)+EX *CONJG(EX )
      STRGFS(NPS)=STRGFS(NPS)+EXS*CONJG(EXS)
      STRGFP(NPS)=STRGFP(NPS)+EX1*CONJG(EX1)
  520 CONTINUE
  595 CONTINUE
  600 CONTINUE
      IF(KTRL(3)) 610,990,610
  610 SGABS=125.66371/(WN(1)*WN(1))
      DO 675 NPS=1,NPOLST   $   NENS1=NENSBP(NPS)   $   NENS2=NENSBT(NPS)
      DO 670 NE1=1,NENS1   $   DO 670 NE2=1,NENS2   $   DO 670 MTFF=1,MULT1Z
      MTFTTW=2*MTFF-MULT1Z-1   $   MTFINV=MULT1Z+1-MTFF
      B2R=BMTR(NE2,MTFF,NPS)   $   B2I=BMTI(NE2,MTFF,NPS)
      DO 670 MSFF=1,ISMULT
      MSFTTW=ISTRTW+2-2*MSFF   $   MSFINV=ISMULT+1-MSFF
      A2=AMS(NE1,MSFF,NPS)   $   A2B2R=A2*B2R   $   A2B2I=A2*B2I
      DO 670 MTII=1,MULT1Z
      MTITTW=2*MTII-MULT1Z-1   $   MTIINV=MULT1Z+1-MTII
      B1R=BMTR(NE2,MTII,NPS)   $   B1I=BMTI(NE2,MTII,NPS)
      DO 660 MSII=1,ISMULT
      MSITTW=ISTRTW+2-2*MSII
      IF(MSITTW+MTITTW-MSFTTW-MTFTTW) 660,615,660
  615 A1=AMS(NE1,MSII,NPS)   $   A1B1R=A1*B1R   $   A1B1I=A1*B1I
      ABFACR=A1B1R*A2B2R+A1B1I*A2B2I   $   ABFACI=A1B1I*A2B2R-A1B1R*A2B2I
      IF(ISTRTW-1) 621,617,618
  617 IF(MSII-1) 622,621,622
  618 IF(MSII-2) 621,621,622
  621 MTIIT=MTII $   MTFFT=MTFF   $   MSFFT=MSFF   $   GO TO 623
  622 MTIIT=MTIINV   $   MTFFT=MTFINV   $   MSFFT=MSFINV
  623 IF(ISTRTW-1) 631,631,624
```

```
  624 IF(MSII-2) 631,633,631
  631 EX =ZBART (MTIIT,MTFFT,MSFFT)
      EXS=ZBARS (MTIIT,MTFFT,MSFFT)
      EX1=ZBARP (MTIIT,MTFFT,MSFFT)  $  GO TO 635
  633 EX =ZBART1(MTIIT,MTFFT,MSFFT)
      EXS=ZBARS1(MTIIT,MTFFT,MSFFT)
      EX1=ZBARP1(MTIIT,MTFFT,MSFFT)
  635 TSGT=CIMAG((ABFACR+TTI*ABFACI)*EX )
      TSGS=CIMAG((ABFACR+TTI*ABFACI)*EXS)
      TSGP=CIMAG((ABFACR+TTI*ABFACI)*EX1)
      TOTLXC(NPS)=TOTLXC(NPS)+TSGT
      TOTLXS(NPS)=TOTLXS(NPS)+TSGS
      TOTLXP(NPS)=TOTLXP(NPS)+TSGP
  660 CONTINUE
  670 CONTINUE
      TOTLXC(NPS)=SGABS*TOTLXC(NPS)
      TOTLXS(NPS)=SGABS*TOTLXS(NPS)
      TOTLXP(NPS)=SGABS*TOTLXP(NPS)
      REACCR(NPS)=TOTLXC(NPS)-SGABS*STRGFN(NPS)
      REACCS     =TOTLXS(NPS)-SGABS*STRGFS(NPS)
      REACCP     =TOTLXP(NPS)-SGABS*STRGFP(NPS)
      STRGFS(NPS)=REACCS*STGFCS
      STRGFP(NPS)=REACCP*STGFCP
  675 CONTINUE
      DO 980 NPS=1,NPOLST
      WRITE(6,978),TOTLXC(NPS),REACCR(NPS),STRGFS(NPS),STRGFP(NPS),NPS
  978 FORMAT(///12X,7HSGTOTL=E15.6,9H  REACCR=E15.6//
     1  12X,7HSTRGFS=E15.6,9H  STRGFP=E15.6,4X,8HFOR NPS=I1)
  980 CONTINUE
  990 IF(KTRL(3)-2) 995,2000,995
  995 CALL CROSPL
 2000 RETURN
      END
```

```
      SUBROUTINE ACCSAB
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON      FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON      KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON      F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON      ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1            ECM(10),
     2            IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3            NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4            WN(10),WNINI(10),WC(10)
      COMMON      ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1            IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2            AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3            PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4            RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5            XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      ******     DIMENSION FIELDS USED ONLY IN THIS ROUTINE     **********
      COMMON      EXSGRI(840)
      COMMON      CMXADR(30,30),CMXADI(30,30),CMXSJR(3,49,30),
     1            CMXSJI(3,49,30),ZBAR1R(8,8,2),ZBAR1I(8,8,2),
     2            ZBAR2R(8,8,2),ZBAR2I(8,8,2),REACCR(3),TOTLXC(3),
     3            STRGFN(3),STRGFS(3),STRGFP(3)
      COMMON      EXTRA4(203)
CCCCCC     *****     COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES     *******
      COMMON      AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1            NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2            NPOLST
      MULT1Z=IIMULT(1)  $  ISMULT=ISTRTW+1  $  KQNOTW=MULT1Z-1
      MSREPT=ISTRTW+1
      JSMAX=(IIREAD(1)+IIREAD(IIXCAL)-KTRL(1))/2+1
      JSIMAX=(2*IIREAD(1)-KTRL(1))/2+1
      JMJRPT=0
      DO 1160 JS=1,JSMAX
 1160 JMJRPT=JMJRPT+(JS*2-1)
      DO 1165 M1=1,MULT1Z  $  DO 1165 M2=1,MULT1Z  $  DO 1165 MSP=1,2
      ZBAR1R(M1,M2,MSP)=ZBAR1I(M1,M2,MSP)=0.0
      ZBAR2R(M1,M2,MSP)=ZBAR2I(M1,M2,MSP)=0.0
 1165 CONTINUE
      DO 1167 NSP=1,NPOLST  $  TOTLXC(NSP)=0.0
 1167 CONTINUE
      REWIND 8
      DO 1500 KK=1,2
      DO 1170 MS=1,MSREPT  $  DO 1170 JS=1,JMJRPT  $  DO 1170 NR=1,30
      CMXSJR(MS,JS,NR)=0.0
 1170 CMXSJI(MS,JS,NR)=0.0
      REWIND 7
      DO 1390 JJ=1,JJJMAX  $  KTISCK=ISTRTW-2*(ISTRTW/2)
      MBARTW=-(2*JJ-2+KTISCK)  $  CRONEC=1.0
      IF(MBARTW) 1173,1172,1173
 1172 CRONEC=0.5
 1173 DO 1380 K=1,2
      KEXCOM(45)=JJ $ KEXCOM(46)=K  $  CALL NLJJJK
      IF(MXROW) 1380,1380,1175
 1175 DO 1177 N1=1,MXROW
      READ TAPE 7,(CMXADR(N1,N2),CMXADI(N1,N2),N2=1,MXROW)
 1177 CONTINUE
      IF(KK-K) 1380,1180,1380
 1180 IF(JJ-1) 1190,1185,1190
 1185 MXROWM=MXROW
```

```
1190 DO 1375 JMJ=1,JMJRPT  $  JSSUM=0
     DO 1230 JS=1,JSMAX  $  JSADD=2*JS-1  $  JSSUM=JSSUM+JSADD
     IF(JSSUM-JMJ) 1230,1235,1235
1230 CONTINUE
1235 JSTRTW=4*JS-4  $  MJSTTW=2*(JMJ+JSADD-JSSUM-1)
     DO 1370 MSII=1,MSREPT  $  MSITTW=ISTRTW+2-2*MSII
     DO 1365 N2=1,MXROW
     L2TW=LLROW(N2)  $  J2TW=JJROW(N2)  $  CMLJPR=CMLJPI=0.0
     DO 1280 N1=1,MXROW
     L1TW=LLROW(N1)  $  J1TW=JJROW(N1)  $  C1=1.0
     IF(ISTRTW) 1255,1260,1255
1255 IA=L1TW  $  IB=ISTRTW  $  IC=J1TW  $  ID=0  $  IE=IF=MSITTW
     CALL CLEB  $  C1=RAC
1260 IA=J1TW  $  IB=J2TW  $  IC=JSTRTW  $  ID=MSITTW  $  IF=MJSTTW
     IE=IF-ID  $  CALL CLEB  $  C2=RAC  $  MSFTTW=-IE
     ID=MBARTW  $  IE=-MBARTW  $  IF=0  $  CALL CLEB  $  C3=RAC
     S1=(-1.0)**((MBARTW-MSFTTW)/2)
     GEOFAC=S1*C1*C2*C3*CRONEC
     TERMR=GEOFAC*CMXADR(N1,N2)  $  TERMI=GEOFAC*CMXADI(N1,N2)
     CMLJPR=CMLJPR+TERMR  $  CMLJPI=CMLJPI+TERMI
     CMXSJR(MSII,JMJ,N2)=CMXSJR(MSII,JMJ,N2)+TERMR
     CMXSJI(MSII,JMJ,N2)=CMXSJI(MSII,JMJ,N2)+TERMI
1280 CONTINUE
     IF(KTRL(3)) 1310,1365,1310
1310 IF(JS-JS1MAX) 1312,1312,1365
1312 IF(XABSF(MSFTTW)-ISTRTW) 1314,1314,1365
1314 MSFF=(ISTRTW-MSFTTW)/2+1
     F1=2*(L2TW+1)  $  C1=1.0
     IF(ISTRTW) 1316,1318,1316
1316 IA=L2TW  $  IB=ISTRTW  $  IC=J2TW  $  ID=0  $  IE=IF=MSFTTW
     CALL CLEB  $  C1=RAC
1318 C2=1.0
     IF(KQNOTW) 1330,1332,1330
1330 IA=IC=ID=IF=KQNOTW  $  IB=JSTRTW  $  IE=0  $  CALL CLEB  $  C2=RAC
1332 FC12=F1*C1*C2
     DO 1360 MTII=1,MULT1Z
     C3=1.0  $  MTFF=MTII
     IF(KQNOTW) 1334,1338,1334
1334 MTITTW=2*MTII-1-MULT1Z
     MTFTTW=MTITTW+MJSTTW  $  MTFF=(MTFTTW+1+MULT1Z)/2
     IF(XABSF(MTFTTW)-KQNOTW) 1336,1336,1360
1336 ID=MTITTW  $  IE=MJSTTW  $  IF=MTFTTW  $  CALL CLEB  $  C3=RAC
1338 GEO=FC12*C3  $  TER=GEO*CMLJPR  $  TEI=GEO*CMLJPI
     IF(MSII-1) 1352,1351,1352
1351 ZBAR1R(MTII,MTFF,MSFF)=ZBAR1R(MTII,MTFF,MSFF)+TER
     ZBAR1I(MTII,MTFF,MSFF)=ZBAR1I(MTII,MTFF,MSFF)+TEI  $  GO TO 1360
1352 ZBAR2R(MTII,MTFF,MSFF)=ZBAR2R(MTII,MTFF,MSFF)+TER
     ZBAR2I(MTII,MTFF,MSFF)=ZBAR2I(MTII,MTFF,MSFF)+TEI
1360 CONTINUE
1365 CONTINUE
1370 CONTINUE
1375 CONTINUE
1380 CONTINUE
1390 CONTINUE
     DO 1400 N2=1,MXROWM
     WRITE TAPE8,((CMXSJR(MSRE,JMJ,N2),CMXSJI(MSRE,JMJ,N2),
    1              JMJ=1,JMJRPT),MSRE=1,MSREPT)
1400 CONTINUE
1500 CONTINUE
     IF(KTRL(3)) 1610,2000,1610
1610 SGABS=125.66371/(WN(1)*WN(1))
     DO1655 NPS=1,NPOLST  $  NENSI=NENSBP(NPS)  $  NENS2=NENSBT(NPS)
```

```
      DO1650 NE1=1,NENS1 $ DO1650 NE2=1,NENS2 $ DO1650 MTFF=1,MULT1Z
      MTFTTW=2*MTFF-MULT1Z-1  $  MTFINV=MULT1Z+1-MTFF
      B2R=BMTR(NE2,MTFF,NPS)  $  B2I=BMTI(NE2,MTFF,NPS)
      DO1650 MSFF=1,ISMULT
      MSFTTW=ISTRTW+2-2*MSFF  $  MSFINV=ISMULT+1-MSFF
      A2=AMS(NE1,MSFF,NPS)  $  A2B2R=A2*B2R
      DO1650 MTII=1,MULT1Z
      MTITTW=2*MTII-MULT1Z-1  $  MTIINV=MULT1Z+1-MTII
      B1R=BMTR(NE2,MTII,NPS)  $  B1I=BMTI(NE2,MTII,NPS)
      DO1630 MSII=1,ISMULT
      MSITTW=ISTRTW+2-2*MSII  $  MSIINV=ISMULT+1-MSII
      IF(MSITTW+MTITTW-MSFTTW-MTFTTW) 1630,1615,1630
 1615 A1=AMS(NE1,MSII,NPS)  $  A1B1R=A1*B1R  $  A1B1I=A1*B1I
      ABFACR=A1B1R*A2B2R+A1B1I*A2B2I  $  ABFACI=A1B1I*A2B2R-A1B1R*A2B2I
      A2B2I=A2*B2I
      IF(MTFTTW-MTITTW) 1618,1617,1617
 1617 MTIIT=MTII     $  MTFFT=MTFF     $  MSIIT=MSII     $  MSFFT=MSFF
      GO TO 1619
 1618 MTIIT=MTIINV   $  MTFFT=MTFINV   $  MSIIT=MSIINV   $  MSFFT=MSFINV
 1619 IF(MSIIT-1) 1622,1621,1622
 1621 ZBR =ZBAR1R(MTIIT,MTFFT,MSFFT)  $  ZBI=ZBAR1I(MTIIT,MTFFT,MSFFT)
      GO TO 1623
 1622 ZBR =ZBAR2R(MTIIT,MTFFT,MSFFT)  $  ZBI=ZBAR2I(MTIIT,MTFFT,MSFFT)
 1623 TSG=ABFACR*ZBI+ABFACI*ZBR
      TOTLXC(NPS)=TOTLXC(NPS)+TSG
 1630 CONTINUE
 1650 CONTINUE
      TOTLXC(NPS)=SGABS*TOTLXC(NPS)
 1655 CONTINUE
      DO 1730 NPS=1,NPOLST
      WRITE(6,1710),TOTLXC(NPS),NPS
 1710 FORMAT(///10X,7HTOTLXC=E15.7,9H FOR NPS=I1)
 1730 CONTINUE
 2000 RETURN
      END
```

```
      SUBROUTINE CROSPL
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      *****    COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES    *******
      COMMON       EXSGRI(840),EXTRA4(10812),EXTRA5(525)
      COMMON       MULTIZ,MULTPO,MULTP1,MULTP2,NANGMI,NANGMX,N5INC,
     1             NAGR12,I1OUT,KQNOTW,CLEBCH,FMULFC,WNFAC
      COMMON       AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1             NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2             NPOLST
CCCCCC      ******     DIMENSIONS CHARACTERISTIC TO THE PRESENT ROUTINE     *****
      DIMENSION    BFAC1R(8,16,7),BFAC2R(8,16,7),BFAC3R(8,16,7),
     1             BFAC4R(8,16,7),BFAC5R(8,16,7),BFAC1I(8,16,7),
     2             BFAC2I(8,16,7),BFAC3I(8,16,7),BFAC4I(8,16,7),
     3             BFAC5I(8,16,7)
     4             XAMP1R(16,2,7),XAMP2R(16,2,7),XAMP3R(16,2,7),
     5             XAMP1I(16,2,7),XAMP2I(16,2,7),XAMP3I(16,2,7)
      EQUIVALENCE  (EXTRA4(     1),BFAC1R),(EXTRA4( 897),BFAC1I),
     1             (EXTRA4( 1793),BFAC2R),(EXTRA4(2689),BFAC2I),
     2             (EXTRA4( 3585),BFAC3R),(EXTRA4(4481),BFAC3I),
     3             (EXTRA4( 5377),BFAC4R),(EXTRA4(6273),BFAC4I),
     4             (EXTRA4( 7169),BFAC5R),(EXTRA4(8065),BFAC5I),
     5             (EXTRA4( 8961),XAMP1R),(EXTRA4( 9185),XAMP1I),
     6             (EXTRA4( 9409),XAMP2R),(EXTRA4( 9633),XAMP2I),
     7             (EXTRA4( 9857),XAMP3R),(EXTRA4(10081),XAMP3I)
      DIMENSION    FCR(35),FCI(35),SGMAC(35),POLTHN(35,2,3),
     1             SGMTHN(35,2,3),FCR5(100),FCI5(100),SGMAC5(100),
     2             POLTH5(100),SGMTH5(100)
      EQUIVALENCE  (EXTRA5(     1),FCR    ),(EXTRA5(  36),FCI    ),
     1             (EXTRA5(    71),SGMAC  ),(EXTRA5( 106),POLTHN),
     2             (EXTRA5(   316),SGMTHN),
     3             (EXTRA5(     1),FCR5   ),(EXTRA5( 101),FCI5   ),
     4             (EXTRA5(   201),SGMAC5),(EXTRA5( 301),POLTH5),
     5             (EXTRA5(   401),SGMTH5)
   55 KQNOTW=IIREAD(1)*2-KTRL(1)
   60 CLEBCH=0.0000001  $  SQRT10=SQRTF(10.)  $  ISMULT=ISTRTW+1
      LINMAX=ISTRTW*ISTRTW+1  $  MULT1Z=IIMULT(1)
      IF(CHARGE) 264,252,264
  252 IF(KTRL(8)) 254,256,254
  254 DO 255 N=1,NANGLR  $  FCR(N)=0.0
  255 FCI(N)=0.0  $  GO TO 281
  256 DO 257 N=1,NANGLR  $  FCR5(N)=0.0
  257 FCI5(N)=0.0  $  GO TO 281
  264 ETA=CE(1)
      DO 270 N=1,NANGLR  $  SN=SINF(THETA(N)*0.5)  $  SN=SN*SN
      FLN=ETA*(LOGF(SN))-2.0*SGMAZZ(1)  $  FNO=ETA/(2.0*WN(1)*SN)
```

```
      FCRT=-FNO*COSF(FLN)*SQRT10  $  FCIT= FNO*SINF(FLN)*SQRT10
      IF(KTRL(8)) 265,266,265
  265 FCR(N)=FCRT  $  FCI(N)=FCIT  $  GO TO 270
  266 FCR5(N)=FCRT  $  FCI5(N)=FCIT
  270 CONTINUE
      DO 280 N=1,NANGLR
      IF(KTRL(8)) 273,275,273
  273 SGMAC (N)=FCR (N)**2+FCI (N)**2  $  GO TO 280
  275 SGMAC5(N)=FCR5(N)**2+FCI5(N)**2
  280 CONTINUE
  281 NAGREP=((NANGLR-1)/7)+1
      IF(KTLOUT(15)) 4110,4190,4110
 4050 FORMAT(8E15.6)
 4110 WRITE(6,4120)
 4120 FORMAT(15HORUTHERFORD-AMP)
      IF(KTRL(8)) 4125,4135,4125
 4125 WRITE(6,4050),(FCR(N),N=1,NANGLR)
      WRITE(6,4050),(FCI(N),N=1,NANGLR)
      GO TO 4190
 4135 WRITE(6,4050),(FCR5(N),N=1,NANGLR)
      WRITE(6,4050),(FCI5(N),N=1,NANGLR)
 4190 CONTINUE
CCCCCC      ******      DO LOOP FOR DIFFERENT STATES      ******
      DO 2000 I1=1,IIXCAL
      MULTPO=IIMULT(I1) $ MULTP1=MULTPO+1 $ MULTP2=MULTPO+2
      WNFAC=SQRT10/WN(I1)
      I1OUT=I1
      NAGR12=0  $  CALL CMMMFC
CCCCCC      ******      DO LOOP FOR DIFFERENT BLOCKS OF ANGLES      ******
      DO 1900 NAGR12=1,NAGREP
      N5INC=7*(NAGR12-1)  $  NANGMI=1  $  NAGXCH=7+N5INC
      NANGMX=XMINOF( NAGXCH,NANGLR)-N5INC
      REWIND 7  $  REWIND 8
      CALL BFCTOR
      KDIF=KPRITR(I1)-KPRITR(1)+IIREAD(I1)-IIREAD(1)
      DO 1600 NPS=1,NPOLST
      NENS1=NENSBP(NPS)  $  NENS2=NENSBT(NPS)  $  I2=NPS
      DO 1030 N3=NANGMI,NANGMX  $  N5=N3+N5INC
      IF(KTRL(8)) 1010,1005,1010
 1005 SGMTH5(N5)=0.0 $ POLTH5(N5)=0.0  $  GO TO 1030
 1010 DO 1020 N4=1,NFAI
      SGMTHN(N5,N4,I2)=0.0  $  POLTHN(N5,N4,I2)=0.0
 1020 CONTINUE
 1030 CONTINUE
      DO 1500 NE1=1,NENS1  $  DO 1500 NE2=1,NENS2
      DO 1050 MN=1,MULTPO  $  DO 1050 N4=1,NFAI
      DO 1050 N3=NANGMI,NANGMX
      XAMP1R(MN,N4,N3)=0.0  $  XAMP1I(MN,N4,N3)=0.0
      XAMP2R(MN,N4,N3)=0.0  $  XAMP2I(MN,N4,N3)=0.0
      XAMP3R(MN,N4,N3)=0.0  $  XAMP3I(MN,N4,N3)=0.0
 1050 CONTINUE
      DO 1350 MSII=1,ISMULT
      MSITTW=ISTRTW+2-2*MSII  $  WEITSS=AMS(NE1,MSII  ,NPS)
      DO 1350 MSFF=1,ISMULT
      MSFTTW=ISTRTW+2-2*MSFF  $  LINT=MSII+ISMULT*(MSFF-1)
      DO 1350 MTII=1,MULT1Z
      MTITTW=2*MTII-IIMULT(1)-1  $  MTIINV=MULT1Z+1-MTII
      WEITTS=BMTR(NE2,MTII  ,NPS)
      DO 1340 MTFF=1,MULTPO
      MTFTTW=2*MTFF-IIMULT(I1)-1  $  MTFINV=MULTPO+1-MTFF
      MEL=(MSITTW+MTITTW-MSFTTW-MTFTTW)/2
      WEIT1=WEITSS*WEITTS $  FMEL=MEL $  IF(LINT-LINMAX) 1075,1075,1077
```

```
1075 LIN=LINT   $  MTIT=MTII   $  MTFT=MTFF   $  GO TO 1080
1077 LIN=ISMULT*ISMULT+1-LINT   $  SIGNKM=(-1.0)**(KDIF+MELT
     WEIT1=WEIT1*SIGNKM   $  MTIT=MTIINV   $  MTFT=MTFINV
1080 DO 1300 N3=NANGMI,NANGMX
     GO TO (1111,1113,1115,1117,1119),LIN
1111 AZMF1R=BFAC1R(MTIT,MTFT,N3)
     AZMF1I=BFAC1I(MTIT,MTFT,N3)  $  GO TO 1130
1113 AZMF1R=BFAC2R(MTIT,MTFT,N3)
     AZMF1I=BFAC2I(MTIT,MTFT,N3) $  GO TO 1130
1115 AZMF1R=BFAC3R(MTIT,MTFT,N3)
     AZMF1I=BFAC3I(MTIT,MTFT,N3)  $  GO TO 1130
1117 AZMF1R=BFAC4R(MTIT,MTFT,N3)
     AZMF1I=BFAC4I(MTIT,MTFT,N3)  $  GO TO 1130
1119 AZMF1R=BFAC5R(MTIT,MTFT,N3)
     AZMF1I=BFAC5I(MTIT,MTFT,N3)
1130 DO 1300 N4=1,NFAI
     ARG=FMEL*FAI(N4)   $  EXPMLR=COSF(ARG)   $  EXPMLI=SINF(ARG)
     TERM1R=(EXPMLR*AZMF1R-EXPMLI*AZMF1I)*WEIT1
     TERM1I=(EXPMLR*AZMF1I+EXPMLI*AZMF1R)*WEIT1
     GO TO (1161,1163,1165),MSFF
1161 XAMP1R(MTFF,N4,N3)=XAMP1R(MTFF,N4,N3)+TERM1R
     XAMP1I(MTFF,N4,N3)=XAMP1I(MTFF,N4,N3)+TERM1I   $  GO TO 1300
1163 XAMP2R(MTFF,N4,N3)=XAMP2R(MTFF,N4,N3)+TERM1R
     XAMP2I(MTFF,N4,N3)=XAMP2I(MTFF,N4,N3)+TERM1I   $  GO TO 1300
1165 XAMP3R(MTFF,N4,N3)=XAMP3R(MTFF,N4,N3)+TERM1R
     XAMP3I(MTFF,N4,N3)=XAMP3I(MTFF,N4,N3)+TERM1I
1300 CONTINUE
1340 CONTINUE
1350 CONTINUE
     IF(I1-1) 1407,1361,1407
1361 IF(CHARGE) 1362,1407,1362
1362 AMS1=AMS(NE1   ,1,NPS)
     AMS2=AMS(NE1   ,2,NPS)
     AMS3=AMS(NE1   ,3,NPS)
     DO 1405 N3=NANGMI,NANGMX   $  N5=N3+N5INC
     IF(KTRL(8)) 1380,1385,1380
1380 FCRR=FCR (N5)   $  FCII=FCI (N5)   $  GO TO 1390
1385 FCRR=FCR5(N5)   $  FCII=FCI5(N5)
1390 DO 1405 N4=1,NFAI
     DO 1405 MTFF=1,MULTPO
     BMT1=BMTR(NE2 ,MTFF,NPS)
     AMBM1=AMS1*BMT1   $  AMBM2=AMS2*BMT1   $  AMBM3=AMS3*BMT1
     XAMP1R(MTFF,N4,N3)=XAMP1R(MTFF,N4,N3)+FCRR*AMBM1
     XAMP1I(MTFF,N4,N3)=XAMP1I(MTFF,N4,N3)+FCII*AMBM1
     XAMP2R(MTFF,N4,N3)=XAMP2R(MTFF,N4,N3)+FCRR*AMBM2
     XAMP2I(MTFF,N4,N3)=XAMP2I(MTFF,N4,N3)+FCII*AMBM2
     XAMP3R(MTFF,N4,N3)=XAMP3R(MTFF,N4,N3)+FCRR*AMBM3
     XAMP3I(MTFF,N4,N3)=XAMP3I(MTFF,N4,N3)+FCII*AMBM3
1405 CONTINUE
1407 I2=NPS
     DO 1450 N3=NANGMI,NANGMX   $  N5=N3+N5INC
     DO 1450 N4=1,NFAI   $  CO=COSF(FAI(N4))   $ SI=SINF(FAI(N4))
     DO 1430 MTFF=1,MULTPO
     XP1R=XAMP1R(MTFF,N4,N3)   $  XP1I=XAMP1I(MTFF,N4,N3)
     XP2R=XAMP2R(MTFF,N4,N3)   $  XP2I=XAMP2I(MTFF,N4,N3)
     XP3R=XAMP3R(MTFF,N4,N3)   $  XP3I=XAMP3I(MTFF,N4,N3)
     GO TO (1421,1423,1425), ISMULT
1421 TERMSG=XP1R*XP1R+XP1I*XP1I   $  TERMPL=0.0   $  GO TO 1427
1423 TERMSG=XP1R*XP1R+XP1I*XP1I+XP2R*XP2R+XP2I*XP2I
     TERMPL=CO*(XP1R*XP2I-XP1I*XP2R)
    1       -SI*(XP1R*XP2R+XP1I*XP2I)
     GO TO 1427
```

```
1425 TERMSG=XP1R*XP1R+XP1I*XP1I+XP2R*XP2R+XP2I*XP2I+XP3R*XP3R+XP3I*XP3I
     TERMPL=0.0
1427 IF(KTRL(8)) 1428,1429,1428
1428 SGMTHN(N5,N4,I2)=SGMTHN(N5,N4,I2)+TERMSG
     POLTHN(N5,N4,I2)=POLTHN(N5,N4,I2)+TERMPL
     GO TO 1430
1429 SGMTH5(N5)=SGMTH5(N5)+TERMSG
     POLTH5(N5)=POLTH5(N5)+TERMPL
1430 CONTINUE
1450 CONTINUE
1500 CONTINUE
     DO 1560 N3=NANGMI,NANGMX   $  N5=N3+N5INC
     IF(KTRL(8)) 1547,1545,1547
1545 POLTH5(N5)=2.0*POLTH5(N5)/SGMTH5(N5)   $  GO TO 1560
1547 DO 1550 N4=1,NFAI
     POLTHN(N5,N4,I2)=2.0*POLTHN(N5,N4,I2)/SGMTHN(N5,N4,I2)
1550 CONTINUE
1560 CONTINUE
1600 CONTINUE
1900 CONTINUE
     IF(KTRL(17)) 1905,1930,1905
1905 IF(KTRL(8)) 1930,1910,1930
1910 IF(CHARGE) 1912,1930,1912
1912 IF(I1-1) 1930,1915,1930
1915 DO 1920 NA=1,NANGLR
1920 SGMTH5(NA)=SGMTH5(NA)/SGMAC5(NA)
1930 CALL OUTPUT
2000 CONTINUE
     RETURN
     END
```

```
      SUBROUTINE CMMMFC
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      ******      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES    *****
      COMMON       EXSGRI(70,6)  $  TYPE COMPLEX EXSGRI
      COMMON       EXTRA4(10812),EXTRA5(525)
      COMMON       MULT1Z,MULTPO,MULTP1,MULTP2,NANGMI,NANGMX,N5INC,
     1             NAGR12,I1OUT,KQNOTW,CLEBCH,FMULFC,WNFAC
      COMMON       AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1             NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2             NPOLST
CCCCCC      ******     DIMENSIONS CHARACTERISTIC TO THE PRESENT ROUTINE     *****
      DIMENSION  CMMM1R(8,16,8),CMMM2R(8,16,8),CMMM3R(8,16,8),
     1           CMMM4R(8,16,8),CMMM5R(8,16,8),CMMM1I(8,16,8),
     2           CMMM2I(8,16,8),CMMM3I(8,16,8),CMMM4I(8,16,8),
     3           CMMM5I(8,16,8)
      DIMENSION  CMXR(3,8,13),CMXI(3,8,13)
      DIMENSION  CMXSJR(3,49),CMXSJI(3,49),CGJSME(7)
      EQUIVALENCE  (EXTRA4(    1),CMMM1R),(EXTRA4(1025),CMMM1I),
     1             (EXTRA4(2049),CMMM2R),(EXTRA4(3073),CMMM2I),
     2             (EXTRA4(4097),CMMM3R),(EXTRA4(5121),CMMM3I),
     3             (EXTRA4(6145),CMMM4R),(EXTRA4(7169),CMMM4I),
     4             (EXTRA4(8193),CMMM5R),(EXTRA4(9217),CMMM5I)
      EQUIVALENCE  (EXTRA4(10241),CMXR),(EXTRA4(10553),CMXI)
      EQUIVALENCE  (EXTRA4(10241),CMXSJR),(EXTRA4(10388),CMXSJI),
     1             (EXTRA4(10535),CGJSME)
      DIMENSION  NROWST(6)
      I1=I1OUT
      SQRT10=SQRTF(10.0)   $  LINMAX=ISTRTW*ISTRTW+1   $  ISMULT=ISTRTW+1
      MSREPT=(ISTRTW/2)+1
      JSMAX=(IIREAD(1)+IIREAD(IIXCAL)-KTRL(1))/2+1   $  JMJRPT=0
      DO 280 JS=1,JSMAX
  280 JMJRPT=JMJRPT+(JS*2-1)
      JSMAXI=(IIREAD(I1)+IIREAD(I1)-KTRL(1))/2+1   $  JMJRPI=0
      DO 285 JS=1,JSMAXI
  285 JMJRPI=JMJRPI+(JS*2-1)
      K14=KEXCOM(14)-1   $  K14RPT=(K14/8)+1
      REWIND 7
  300 DO 1200 K14R=1,K14RPT
      REWIND 8
      LTRUMI=8*(K14R-1)   $  LTRUMX=LTRUMI+7
      LTRUMX=XMINOF(K14,LTRUMX) $  LWRTMX=LTRUMX-LTRUMI+1
      DO 310 MTII=1,MULT1Z
      DO 310 MTFF=1,MULTPO
      DO 310 LWRT=1,LWRTMX
      CMMM1R(MTII,MTFF,LWRT)=0.0   $  CMMM2R(MTII,MTFF,LWRT)=0.0
```

```
      CMMM3R(MTII,MTFF,LWRT)=0.0  $  CMMM4R(MTII,MTFF,LWRT)=0.0
      CMMM5R(MTII,MTFF,LWRT)=0.0
      CMMM1I(MTII,MTFF,LWRT)=0.0  $  CMMM2I(MTII,MTFF,LWRT)=0.0
      CMMM3I(MTII,MTFF,LWRT)=0.0  $  CMMM4I(MTII,MTFF,LWRT)=0.0
      CMMM5I(MTII,MTFF,LWRT)=0.0
  310 CONTINUE
      IF(KTRL(7)) 332,330,332
  330 JJMAX=JJJMAX  $  KMAX=2  $  GO TO 350
  332 WNFCAD=SQRT10/WN(1)  $  KEXCOM(45)=1  $  KEXCOM(46)=1
      CALL NLJJJK  $  MXROW1=MXROW
      KEXCOM(46)=2  $  CALL NLJJJK  $  MXROW2=MXROW
      JJMAX=MXROW1+MXROW2 $ KMAX=1 $ HAT1=IIMULT(1) $ HATN=IIMULT(I1)
      HATFAC=SQRTF(HAT1/HATN)*2.0*WNFCAD  $  ID=0  $  IE=KQNOTW
      IF=KQNOTW  $  IB=IIMULT(I)-1  $  IC=IIMULT(II)-1
      DO 340 JS=1,JSMAX
      JSTRTW=4*JS-4  $  IA=JSTRTW  $  CALL CLEB
      CGJSME(JS)=RAC*HATFAC
  340 CONTINUE
  350 DO 1010 JJ=1,JJMAX
      IF(KTRL(7)) 353,351,353
  351 KS=KTRL(1)+ISTRTW $ KTISCK=KS-2*(KS/2) $ JJTRTW=2*JJ-2+KTISCK
      GO TO 360
  353 IF(JJ-1) 357,355,357
  355 KEXCOM(45)=1  $  KEXCOM(46)=1  $  CALL NLJJJK  $  MXROWM=MXROW
  357 IF(JJ-MXROW1-1) 360,359,360
  359 KEXCOM(45)=1  $  KEXCOM(46)=2  $  CALL NLJJJK  $  MXROWM=MXROW
  360 DO 1005 K=1,KMAX
      IF(KTRL(7)) 511,411,511
  411 SIGN7=(-1.0)**((ISTRTW+IIMULT(1)-1-JJTRTW)/2+K+1)
      KEXCOM(45)=JJ  $  KEXCOM(46)=K  $  CALL NLJJJK
      IF(MXROW) 1005,1005,412
  412 DO 420 I2=1,I1  $  NROWST(I2)=0
      DO 415 N1=1,MXROW  $  IF(NNROW(N1)-I2) 415,413,420
  413 NROWST(I2)=NROWST(I2)+1
  415 CONTINUE
  420 CONTINUE
      I3=I1-1  $  NROWMI=1  $ IF(I3) 422,425,422
  422 DO 423 I2=1,I3
  423 NROWMI=NROWMI+NROWST(I2)
  425 NROWNS=NROWST(I1)
      DO 450 MSRE=1,MSREPT
      DO 450 MTII=1,MULTIZ
      MTIINV=MULTIZ+1-MTII
      READ TAPE 8,(BR(N1),BI(N1),N1=1,MXROW)
      DO 448 NRW=1,NROWNS
      N1=NRW+NROWMI-1  $  CR=BR(N1)*WNFAC  $  CI=BI(N1)*WNFAC
      L1=(LLROW(N1)/2)+1 $ ER1=EXSGRI(L1,I1) $ EI1=CIMAG(EXSGRI(L1,I1))
      ER=ER1*ER1-EI1*EI1 $ EI=2.0*ER1*EI1 $ CMR=CR*ER-CI*EI
      CMI=CR*EI+CI*ER
      IF(MSRE-1) 443,441,443
  441 CMXR(1,MTII,NRW)=CMR  $  CMXI(1,MTII,NRW)=CMI
      IF(ISTRTW-1) 448,442,442
  442 ISR=ISTRTW+1
      CMXR(ISR,MTIINV,NRW)=CMR*SIGN7
      CMXI(ISR,MTIINV,NRW)=CMI*SIGN7  $  GO TO 448
  443 CMXR(2,MTII,NRW)=CMR  $  CMXI(2,MTII,NRW)=CMI
  448 CONTINUE
  450 CONTINUE
      GO TO 530
  511 IF(JJ-MXROW1) 512,512,514
  512 N1=JJ  $  GO TO 515
  514 N1=JJ-MXROW1
```

```
515 L1TRTW=LLROW(N1)   $  L1TR=L1TRTW/2
    READ TAPE8,((CMXSJR(MSRE,JMJ),CMXSJI(MSRE,JMJ),
   1            JMJ=1,JMJRPT),MSRE=1,ISMULT)
    IF(L1TR-LTRUMI) 1010,516,516
516 IF(L1TR-LTRUMX) 517,517,1010
517 J1TRTW=JJROW(N1) $ L1=L1TR+1
    ER1=EXSGRI(L1,1)   $  EI1=CIMAG(EXSGRI(L1,1))
    ER=ER1*ER1-EI1*EI1   $ EI=2.0*ER1*EI1
    DO 520 JMJ=1,JMJRPI
    DO 520 MSRE=1,ISMULT
    CR=CMXSJR(MSRE,JMJ)   $  CI=CMXSJI(MSRE,JMJ)
    CMXSJR(MSRE,JMJ)=CR*ER-CI*EI $  CMXSJI(MSRE,JMJ)=CR*EI+CI*ER
520 CONTINUE
530 NRWMAX=NROWNS*(1-KTRL(7))+KTRL(7)
    DO 1002 NRW=1,NRWMAX
    IF(KTRL(7)) 650,610,650
610 NRWT=NROWMI+NRW-1   $  L1TRTW=LLROW(NRWT)   $  L1TR=L1TRTW/2
    IF(L1TR-LTRUMI) 1002,612,612
612 IF(L1TR-LTRUMX) 615,615,1002
615 J1TRTW=JJROW(NRWT)
650 DO 1000 MSII=1,ISMULT
    MSITTW=ISTRTW+2-2*MSII $ MSIINV=ISMULT+1-MSII
    DO 1000 MTII=1,MULT1Z
    MTITTW=2*MTII-IIMULT(I)-1   $  MSMITW=MSITTW+MTITTW
    DO 990 MTFF=1,MULTPO
    MTFTTW=2*MTFF-IIMULT(I1)-1
    DO 980 MSFF=1,MSREPT
    MSFTTW=ISTRTW+2-2*MSFF   $  LINEAR=MSII+ISMULT*(MSFF-1)
    IF(LINEAR-LINMAX) 660,660,980
660 IF(KTRL(7)) 710,670,710
670 IA=J1TRTW $ IB=MULTPO-1 $ IC=JJTRTW $ IE=MTFTTW $ IF=MSMITW
    ID=IF-IE   $  CALL CLEB   $  CLEBS1=RAC
    PLFACR=CMXR(MSII,MTII,NRW)*RAC
    PLFACI=CMXI(MSII,MTII,NRW)*RAC   $ GO TO 750
710 SUMJJR=0   $  SUMJJI=0   $  MJTRTW=MTFTTW-MTITTW
    IB=IIMULT(I)-1 $ IC=IIMULT(I1)-1 $ ID=MJTRTW $ IE=MTITTW
    IF=MTFTTW
    DO 720 JS=1,JSMAXI
    JSTRTW=4*JS-4
    IF(JSTRTW-XABSF(MJTRTW)) 720,712,712
712 IA=JSTRTW   $  CALL CLEB   $  T1=RAC*CGJSME(JS)
    MI=XABSF(MJTRTW)/2   $  JMJ=0   $  JSM1=JS-1
    DO 715 JS1=1,JSM1
715 JMJ=JMJ+(2*JS1-1)   $  JMJ=JMJ+M1+1
    IF(MJTRTW) 718,717,717
717 SUMJJR=SUMJJR+T1*CMXSJR(MSII,JMJ)
    SUMJJI=SUMJJI+T1*CMXSJI(MSII,JMJ)   $  GO TO 720
718 T1=T1*((-1.0)**((L1TRTW+J1TRTW-ISTRTW-JSTRTW)/2))
    SUMJJR=SUMJJR+T1*CMXSJR(MSIINV,JMJ)
    SUMJJI=SUMJJI+T1*CMXSJI(MSIINV,JMJ)
720 CONTINUE
    PLFACR=SUMJJR   $  PLFACI=SUMJJI
750 I1=I1
    IA=L1TRTW $ IB=ISTRTW $ IC=J1TRTW $ IE=MSFTTW $ IF=MSMITW-MTFTTW
    ID=IF-IE $ CALL CLEB $ CLEBS2=RAC $ MEL=ID/2 $ MELABS=XABSF(MEL)
    FL2=L1TRTW+1 $ K1=L1TR+MELABS+1 $ K2=L1TR-MELABS+1
    IF(K2) 980,980,760
760 FCC=FACLOG(K1)-FACLOG(K2)   $  FCC=EXPF(-0.5*FCC)
    GEOFAC=FL2*FCC*CLEBS2*((-1.0)**((MEL+MELABS)/2))
    PLFACR=GEOFAC*PLFACR   $  PLFACI=GEOFAC*PLFACI
    N1=L1TR+1-LTRUMI
    GO TO ( 831,833,835,837,839 ),LINEAR
```

```
  831 CMMM1R(MTII,MTFF,N1)=CMMM1R(MTII,MTFF,N1)+PLFACR
      CMMM1I(MTII,MTFF,N1)=CMMM1I(MTII,MTFF,N1)+PLFACI   $  GO TO 840
  833 CMMM2R(MTII,MTFF,N1)=CMMM2R(MTII,MTFF,N1)+PLFACR
      CMMM2I(MTII,MTFF,N1)=CMMM2I(MTII,MTFF,N1)+PLFACI   $  GO TO 840
  835 CMMM3R(MTII,MTFF,N1)=CMMM3R(MTII,MTFF,N1)+PLFACR
      CMMM3I(MTII,MTFF,N1)=CMMM3I(MTII,MTFF,N1)+PLFACI   $  GO TO 840
  837 CMMM4R(MTII,MTFF,N1)=CMMM4R(MTII,MTFF,N1)+PLFACR
      CMMM4I(MTII,MTFF,N1)=CMMM4I(MTII,MTFF,N1)+PLFACI   $  GO TO 840
  839 CMMM5R(MTII,MTFF,N1)=CMMM5R(MTII,MTFF,N1)+PLFACR
      CMMM5I(MTII,MTFF,N1)=CMMM5I(MTII,MTFF,N1)+PLFACI
  840 I1=I1
  980 CONTINUE
  990 CONTINUE
 1000 CONTINUE
 1002 CONTINUE
 1005 CONTINUE
 1010 CONTINUE
      DO 1130 N1=1,LWRTMX
      DO 1120 LIN=1,LINMAX
      GO TO (1110,1112,1114,1116,1118 ),LIN
 1110 WRITE TAPE 7,((CMMM1R(MTII,MTFF,N1),CMMM1I(MTII,MTFF,N1),
     1            MTFF=1,MULTPO),MTII=1,MULTIZ)    $  GO TO 1120
 1112 WRITE TAPE 7,((CMMM2R(MTII,MTFF,N1),CMMM2I(MTII,MTFF,N1),
     1            MTFF=1,MULTPO),MTII=1,MULTIZ)    $  GO TO 1120
 1114 WRITE TAPE 7,((CMMM3R(MTII,MTFF,N1),CMMM3I(MTII,MTFF,N1),
     1            MTFF=1,MULTPO),MTII=1,MULTIZ)    $  GO TO 1120
 1116 WRITE TAPE 7,((CMMM4R(MTII,MTFF,N1),CMMM4I(MTII,MTFF,N1),
     1            MTFF=1,MULTPO),MTII=1,MULTIZ)    $  GO TO 1120
 1118 WRITE TAPE 7,((CMMM5R(MTII,MTFF,N1),CMMM5I(MTII,MTFF,N1),
     1            MTFF=1,MULTPO),MTII=1,MULTIZ)
 1120 CONTINUE
 1130 CONTINUE
 1200 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BFCTOR
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      ******      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES    *****
      COMMON       EXSGRI(840),EXTRA4(10812),EXTRA5(525)
      COMMON       MULTIZ,MULTPO,MULTPI,MULTP2,NANGMI,NANGMX,N5INC,
     1             NAGR12,I1OUT,KQNOTW,CLEBCH,FMULFC,WNFAC
      COMMON       AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1             NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2             NPOLST
CCCCCC      ******     DIMENSIONS CHARACTERISTIC TO THE PRESENT ROUTINE     *****
      DIMENSION  BFAC1R(8,16,7),BFAC2R(8,16,7),BFAC3R(8,16,7),
     1           BFAC4R(8,16,7),BFAC5R(8,16,7),BFAC1I(8,16,7),
     2           BFAC2I(8,16,7),BFAC3I(8,16,7),BFAC4I(8,16,7),
     3           BFAC5I(8,16,7)
      DIMENSION  P(9,14,7),CMMMR(8,16),CMMMI(8,16)
      EQUIVALENCE  (EXTRA4(    1),BFAC1R),(EXTRA4( 897),BFAC1I),
     1             (EXTRA4(1793),BFAC2R),(EXTRA4(2689),BFAC2I),
     2             (EXTRA4(3585),BFAC3R),(EXTRA4(4481),BFAC3I),
     3             (EXTRA4(5377),BFAC4R),(EXTRA4(6273),BFAC4I),
     4             (EXTRA4(7169),BFAC5R),(EXTRA4(8065),BFAC5I)
      EQUIVALENCE  (EXTRA4(8961),P),(EXTRA4(9843),CMMMR),
     1                              (EXTRA4(9971),CMMMI)
      I1=I1OUT
      SQRT10=SQRTF(10.0)    $  LINMAX=ISTRTW*ISTRTW+1   $  ISMULT=ISTRTW+1
      MSREPT=(ISTRTW/2)+1
      JSMAX=(IIREAD(1)+IIREAD(IIXCAL)-KTRL(1))/2+1   $  JMJRPT=0
      DO 280 JS=1,JSMAX
  280 JMJRPT=JMJRPT+(JS*2-1)
      JSMAXI=(IIREAD(1)+IIREAD(I1)-KTRL(1))/2+1   $  JMJRPI=0
      DO 285 JS=1,JSMAXI
  285 JMJRPI=JMJRPI+(JS*2-1)
      K14=KEXCOM(14)-1   $  K14RPT=(K14/8)+1
      REWIND 7
 1400 DO 1410 MTII=1,MULT1Z
      DO 1410 MTFF=1,MULTPO
      DO 1410 N3=NANGMI,NANGMX
      BFAC1R(MTII,MTFF,N3)=0.0 $ BFAC2R(MTII,MTFF,N3)=0.0
      BFAC3R(MTII,MTFF,N3)=0.0 $ BFAC4R(MTII,MTFF,N3)=0.0
      BFAC5R(MTII,MTFF,N3)=0.0
      BFAC1I(MTII,MTFF,N3)=0.0 $ BFAC2I(MTII,MTFF,N3)=0.0
      BFAC3I(MTII,MTFF,N3)=0.0 $ BFAC4I(MTII,MTFF,N3)=0.0
      BFAC5I(MTII,MTFF,N3)=0.0
 1410 CONTINUE
      DO 1700 K14R=1,K14RPT
      LTRUMI=8*(K14R-1)   $  LTRUMX=LTRUMI+7 $ LTRUMX=XMINOF(K14,LTRUMX)
```

```
      LWRTMX=LTRUMX-LTRUMI+1
      MMCH2=ISTRTW+IIREAD(I)+IIREAD(I1)+1-KTRL(1)
      DO 1450 N3=NANGMI,NANGMX
      N5=N3+N5INC  $  TH=THETA(N5)  $  CO=COSF(TH)  $  SI=1.0/SINF(TH)
      CT=2.0*CO*SI
      IF(KI4R-I) 1427,1425,1427
 1425 P(1,1,N3)=1.0 $ P(2,1,N3)=CO $ P(1,2,N3)=0.0 $ FL=0.0 $ GO TO 1430
 1427 FL=LTRUMI $  FLP1=FL+1.0  $  TWLP1=FL+FLP1
      DO 1428 M=1,MMCH2
 1428 P(1,M,N3)=P(9,M,N3)
      P(2,1,N3)=(TWLP1*CO*P(1,1,N3)-FL*P(8,1,N3))/FLP1
 1430 NMAX=LWRTMX+2
      DO 1445 N=3,NMAX
      FL=FL+1.0  $  FLP1=FL+1.0  $  TWLP1=FL+FLP1  $  NM1=N-1 $ NM2=N-2
      PN1         =(TWLP1*CO*P(NM1,1,N3)-FL*P(NM2,1,N3))/FLP1
      IF(N-NMAX) 1432,1431,1432
 1431 P(NM1,2,N3)= FLP1*SI*(CO*P(NM1,1,N3)-PN1)  $  GO TO 1433
 1432 P(N,1,N3)=PN1
      P(NM1,2,N3)= FLP1*SI*(CO*P(NM1,1,N3)-P(N,1,N3))
 1433 MMCH1=NM1+LTRUMI  $  MMAX=XMINOF(MMCH1,MMCH2)  $  FM=0.0
      DO 1435 M=3,MMAX  $  FMP1=FM+1.0
      P(NM1,M,N3)= CT*FMP1*P(NM1,M-1,N3)-(FL-FM)*(FL+FMP1)*P(NM1,M-2,N3)
 1435 FM=FM+1.0
 1445 CONTINUE
 1450 CONTINUE
      DO 1600 LI=1,LWRTMX  $ LITRP1=LTRUMI+LI
      DO 1500 LIN=1,LINMAX
      MSFF=((LIN-I)/ISMULT)+1  $  MSII=LIN-ISMULT*(MSFF-1)
      MSITTW=ISTRTW+2-2*MSII  $  MSFTTW=ISTRTW+2-2*MSFF
      READ TAPE 7,((CMMMR(MTII,MTFF),CMMMI(MTII,MTFF),
     1              MTFF=1,MULTPO),MTII=1,MULTIZ)
      DO 1500 MTII=1,MULTIZ  $  MTITTW=2*MTII-IIMULT(1)-1
      DO 1495 MTFF=1,MULTPO
      MTFTTW=2*MTFF-IIMULT(II)-1 $ MELTW=MSITTW+MTITTW-MSFTTW-MTFTTW
      MELP1=(XABSF(MELTW)/2)+1  $  IF(LITRP1-MELP1) 1495,1470,1470
 1470 DO 1490 N3=NANGMI,NANGMX  $  PFACT=P(LI,MELP1,N3)
      TR=PFACT*CMMMR(MTII,MTFF)  $  TI=PFACT*CMMMI(MTII,MTFF)
      GO TO ( 1471,1473,1475,1477,1479 ),LIN
 1471 BFAC1R(MTII,MTFF,N3)=BFAC1R(MTII,MTFF,N3)+TR
      BFAC1I(MTII,MTFF,N3)=BFAC1I(MTII,MTFF,N3)+TI  $  GO TO 1490
 1473 BFAC2R(MTII,MTFF,N3)=BFAC2R(MTII,MTFF,N3)+TR
      BFAC2I(MTII,MTFF,N3)=BFAC2I(MTII,MTFF,N3)+TI  $ GO TO 1490
 1475 BFAC3R(MTII,MTFF,N3)=BFAC3R(MTII,MTFF,N3)+TR
      BFAC3I(MTII,MTFF,N3)=BFAC3I(MTII,MTFF,N3)+TI  $  GO TO 1490
 1477 BFAC4R(MTII,MTFF,N3)=BFAC4R(MTII,MTFF,N3)+TR
      BFAC4I(MTII,MTFF,N3)=BFAC4I(MTII,MTFF,N3)+TI  $  GO TO 1490
 1479 BFAC5R(MTII,MTFF,N3)=BFAC5R(MTII,MTFF,N3)+TR
      BFAC5I(MTII,MTFF,N3)=BFAC5I(MTII,MTFF,N3)+TI
 1490 CONTINUE
 1495 CONTINUE
 1500 CONTINUE
 1600 CONTINUE
 1700 CONTINUE
 2000 RETURN
      END
```

```
      SUBROUTINE OUTPUT
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      ******      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES    *****
      COMMON       EXTRA4(11652),EXTRA5(525)
      COMMON       MULTIZ,MULTPO,MULTPI,MULTPZ,NANGMI,NANGMX,N5INC,
     1             NAGR12,I1OUT,KQNOTW,CLEBCH,FMULFC,WNFAC
      COMMON       AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1             NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2             NPOLST
      DIMENSION    FCR(35),FCI(35),SGMAC(35),POLTHN(35,2,3),
     1             SGMTHN(35,2,3),FCR5(100),FCI5(100),SGMAC5(100),
     2             POLTH5(100),SGMTH5(100)
      EQUIVALENCE  (EXTRA5(     1),FCR    ),(EXTRA5(  36),FCI    ),
     1             (EXTRA5(    71),SGMAC  ),(EXTRA5( 106),POLTHN),
     2             (EXTRA5(   316),SGMTHN),
     3             (EXTRA5(     1),FCR5   ),(EXTRA5( 101),FCI5   ),
     4             (EXTRA5(   201),SGMAC5),(EXTRA5( 301),POLTH5),
     5             (EXTRA5(   401),SGMTH5)
      TYPE INTEGER PARIK1
 4000 FORMAT(1H1)
      I1=I1OUT  $  I2=I1-XMAXOF(IIXPLT,IIPPLT)  $  IF(I2) 510,510,505
  505 IF(NFAI+NPOLST-2) 510,507,510
  507 IF(I2-2*(I2/2)) 510,520,510
  510 WRITE(6,4000)
  520 DO 1000 NPS=1,NPOLST
      DO 900 N4=1,NFAI
      FAIDEG=FAI(N4)*57.296+0.001  $  III=IIREAD(I1)
      QV=QVALUE(I1)  $  KP=KPRITR(I1)  $  IF(KTRL(1)) 605,601,605
  601 IIW=III  $  IF(KP-1) 603,602,603
  602 PARIK1=5H (+)  $  GO TO 610
  603 PARIK1=5H (-)  $  GO TO 610
  605 IIW=III+III-1  $  IF(KP-1) 607,606,607
  606 PARIK1=5H/2(+)  $  GO TO 610
  607 PARIK1=5H/2(-)
  610 WRITE(6,620),IIW,PARIK1,QV,FAIDEG,NPS
  620 FORMAT(1H0//20X,5H*****,5X,15H SCATTERING TO I2,A5,10H STATE AT ,
     1   F6.3,4H-MEV,5X,5H*****,3X,5H(FAI=F6.2,8H-DEGREE),5H NPS=I1/)
      NA3RD=(NANGLR+2)/3
      WRITE(6,740)
  740 FORMAT(3(5X,5HANGLE,6X,5HSGMTH,9X,5HPOLTH,3X)/)
      DO 760 N5=1,NA3RD  $  N6=N5+NA3RD  $  N7=N6+NA3RD
      IF(KTRL(8)) 745,747,745
  745 WRITE(6,750),ANGLER(N5),SGMTHN(N5,N4,NPS),POLTHN(N5,N4,NPS)
     1             ,ANGLER(N6),SGMTHN(N6,N4,NPS),POLTHN(N6,N4,NPS)
     2             ,ANGLER(N7),SGMTHN(N7,N4,NPS),POLTHN(N7,N4,NPS)
```

```
      GO TO 760
  747 WRITE(6,750),ANGLER(N5),SGMTH5(N5),POLTH5(N5)
     1            ,ANGLER(N6),SGMTH5(N6),POLTH5(N6)
     2            ,ANGLER(N7),SGMTH5(N7),POLTH5(N7)
  750 FORMAT(3(F10.1,E15.5,F10.3,3X))
  760 CONTINUE
      IF(I1+NPS+NFAI-3) 890,770,890
  770 IF(CHARGE) 772,890,772
  772 WRITE(6,775)
  775 FORMAT(//48X,24HRUTHERFORD CROSS SECTION/)
      NA4H=(NANGLR+3)/4
      DO 785 N5=1,NA4H  $  N6=N5+NA4H  $  N7=N6+NA4H  $  N8=N7+NA4H
      IF(KTRL(8)) 777,779,777
  777 WRITE(6,780),ANGLER(N5),SGMAC (N5),ANGLER(N6),SGMAC (N6)
     1            ,ANGLER(N7),SGMAC (N7),ANGLER(N8),SGMAC (N8)
      GO TO 785
  779 WRITE(6,780),ANGLER(N5),SGMAC5(N5),ANGLER(N6),SGMAC5(N6)
     1            ,ANGLER(N7),SGMAC5(N7),ANGLER(N8),SGMAC5(N8)
  780 FORMAT(F10.1,E14.5,3(F16.1,E14.5))
  785 CONTINUE
  890 CALL PLOTER
  900 CONTINUE
 1000 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PLOTER
CCCCCC      ******   COMMON STATEMENTS THAT ARE COMMON TO ALL THE ROUTINES    ***
      COMMON       FACLOG(500),RAC,IA,IB,IC,ID,IE,IF,L9(9),U9
      COMMON       KTRL(30),KEXCOM(50),EXTCOM(50),KTLOUT(30)
      COMMON       F(71),G(71),FD(70),GD(70),ETA,SIGMAZ,RHOMX,RD,LMAX
      COMMON       ANGLER(100),AR(70),AI(70),BR(70),BI(70),CE(10),DR(10),
     1             ECM(10),
     2             IIMULT(10),IIREAD(10),KPRITR(10),JJROW(30),LLROW(30),
     3             NNROW(30),QVALUE(10),SGMAZZ(10),THETA(100),VCOUPL(20),
     4             WN(10),WNINI(10),WC(10)
      COMMON       ISTRTW,IICPLE,INTYPE,INTMAX,IIXCAL,IIXPLT,IIPCAL,
     1             IIPPLT,JJJMAX,MXROW,NXMAX,NXCPLE,NANGLR,NDFMES,
     2             AMUPMU,CHARGE,CFUNIT,DFN,DFNS,DFNW,DFNSP,ELAB,ETUNIT,
     3             PMAS,RMAS,RZERO,RZEROC,RZEROS,RZEROW,RZROSP,
     4             RMAX,RBAR,SGMAR,TMAS,VSX,VSF,VSO,WSX,WSF,XMAX,XBAR,
     5             XMES1,XMES2,WNUNIT,TTR,TTI,ZERO
      TYPE COMPLEX TTR,TTI,ZERO
CCCCCC      ******      COMMON FIELDS USED IN COMMON BY SEVERAL ROUTINES    *****
      COMMON       EXTRA4(11652),EXTRA5(525)
      COMMON       MULTIZ,MULTP0,MULTP1,MULTP2,NANGMI,NANGMX,N5INC,
     1             NAGR12,I1OUT,KQNOTW,CLEBCH,FMULFC,WNFAC
      COMMON       AMS(3,3,3),BMTR(10,10,3),BMTI(10,10,3),NENSBP(3),
     1             NENSBT(3),SGMEXP(100,6),POLEXP(100,2),FAI(4),NFAI,
     2             NPOLST
      DIMENSION    FCR(35),FCI(35),SGMAC(35),POLTHN(35,2,3),
     1             SGMTHN(35,2,3),FCR5(100),FCI5(100),SGMAC5(100),
     2             POLTH5(100),SGMTH5(100)
      EQUIVALENCE (EXTRA5(     1),FCR    ),(EXTRA5(  36),FCI    ),
     1             (EXTRA5(    71),SGMAC ),(EXTRA5( 106),POLTHN),
     2             (EXTRA5(   316),SGMTHN),
     3             (EXTRA5(     1),FCR5   ),(EXTRA5( 101),FCI5   ),
     4             (EXTRA5(   201),SGMAC5),(EXTRA5( 301),POLTH5),
     5             (EXTRA5(   401),SGMTH5)
CCCCCC      ******      DIMENSIONS CHARACTERISTIC TO THE PRESENT ROUTINE     *****
      DIMENSION    PLOT(103),POINT1(100),POINT2(100)
      EQUIVALENCE (EXTRA4(6051),PLOT),
     1             (EXTRA4(6154),POINT1),(EXTRA4(6254),POINT2)
      TYPE INTEGER PLOT
  801 FORMAT(1H )
  850 FORMAT(11X,103A1)
  851 FORMAT(1H ,F6.2, 4H DEG,103A1)
      NTMAS=TMAS+0.1   $  NPMAS=PMAS+0.1   $  NZZ  =CHARGE+0.1
      DO 500 IPL=1,2   $  IPLCHK=IIXPLT*(2-IPL)+IIPPLT*(IPL-1)
      IF(I1OUT-IPLCHK) 155,155,500
  155 DO 170 NA=1,NANGLR   $  IF(IPL-1) 165,160,165
  160 POINT1(NA)=LOGF(SGMTH5(NA))*0.43429448
      POINT2(NA)=LOGF(SGMEXP(NA,I1OUT))*0.43429448   $  GO TO 170
  165 POINT1(NA)=POLTH5(NA)*10.0
      POINT2(NA)=POLEXP(NA,I1OUT)*10.0
  170 CONTINUE
      CO1=MAX1F(POINT1(1),POINT2(1))
      CO2=MIN1F(POINT1(1),POINT2(1))
      DO 230 NA=1,NANGLR
      CO1=MAX1F(CO1,POINT1(NA),POINT2(NA))
      CO2=MIN1F(CO2,POINT1(NA),POINT2(NA))
  230 CONTINUE
      NCO1=CO1-(1.0-(CO1/ABSF(CO1)))*0.5
      NCO1=NCO1+1
      NCO2=CO2-(1.0-(CO2/ABSF(CO2)))*0.5
      NCYCLE=NCO1-NCO2     $  NSPACE=100/NCYCLE
```

```
      IF(IPL-1) 310,305,310
  305 WRITE(6,307),NCO2,NCO1
  307 FORMAT(47H1PLOT OF XSEC. LOWEST AND HIGHEST * ARE FOR 10E12,9H  AN
     1D 10E12)
      GO TO 320
  310 FNCO2=NCO2  $  FNCO1=NCO1  $  FNCO2=FNCO2*0.1  $  FNCO1=FNCO1*0.1
      WRITE(6,312),FNCO2,FNCO1
  312 FORMAT(52H1PLOT OF POLARIZATION. LOWEST AND HIGHEST * ARE FOR F4.1
     1,6H  AND F4.1)
  320 WRITE(6,330),NTMAS,NPMAS,NZZ,ELAB,ISTRTW,I1OUT,I1READ(I1OUT),
     1              KPRITR(I1OUT)
  330 FORMAT(3H A=I3,3H P=I2,4H ZZ=I3,6H ELAB=F6.2,4H 2S=I1,4H II=I1,4H
     1IR=I1,3H K=I1)
      WRITE(6,331),VSX,WSX,WSF,VSO,DFN,DFNW,DFNS,DFNSP,RZERO,RZEROW,
     1              RZEROS,RZROSP,RZEROC,(WC(N),N=1,6),(VCOUPL(N),N=1,10)
  331 FORMAT(8H VSX ETC,13F8.3/4H WC=6F6.3,8H VCOUPL=10F6.3)
      NSTAR=0
  340 WRITE(6,801)
      DO 350 I=1,103
  350 PLOT(I)=1H
      DO 355 I=2,102,NSPACE
  355 PLOT(I)=1H*
      WRITE(6,850),PLOT
      IF(NSTAR) 500,357,500
  357 DO 365 I=1,103
  365 PLOT(I)=1H
      WRITE(6,801)  $  FNPOWR=NCO2  $  FSPACE=NSPACE
      DO 400 NA=1,NANGLR
      NCOW1=(POINT1(NA)-FNPOWR)*FSPACE+2.1
      NCOW2=(POINT2(NA)-FNPOWR)*FSPACE+2.1
      PLOT(NCOW1)=1HT  $  PLOT(NCOW2)=1HE
      IF(NANGLR-20) 371,371,375
  371 WRITE(6,801)
  375 WRITE(6,851),ANGLER(NA),PLOT
      PLOT(NCOW1)=1H  $  PLOT(NCOW2)=1H
  400 CONTINUE
      NSTAR=1  $  GO TO 340
  500 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CLEB
      COMMON FACLOG(500),RAC,IA,IB,IC,ID,IE,IF
      RAC=0.0
      IF(ID+IE-IF) 1000,105,1000
  105 K1=IA+IB+IC  $  IF(K1-2*(K1/2)) 1000,110,1000
  110 K1=IA+IB-IC  $  K2=IC-XABSF(IB-IA)  $  K3=XMINOF(K1,K2)
      IF(K3) 1000,130,130
  130 IF((-1)**(IB+IE)) 1000,1000,140
  140 IF((-1)**(IC+IF)) 1000,1000,150
  150 IF(IA-XABSF(ID)) 1000,152,152
  152 IF(IB-XABSF(IE)) 1000,154,154
  154 IF(IC-XABSF(IF)) 1000,160,160
  160 IF(IA) 1000,175,165
  165 IF(IB) 1000,175,170
  170 IF(IC) 1000,180,250
  175 RAC=1.0  $  GO TO 1000
  180 FB=IB+1  $  RAC=((-1.0)**((IA-ID)/2))/SQRTF(FB)  $  GO TO 1000
  250 FC2=IC+1  $  IABCP=(IA+IB+IC)/2+1
      IABC=IABCP-IC  $  ICAB=IABCP-IB  $  IBCA=IABCP-IA
      IAPD=(IA+ID)/2+1 $ IAMD=IAPD-ID $ IBPE=(IB+IE)/2+1 $ IBME=IBPE-IE
      ICPF=(IC+IF)/2+1 $ ICMF=ICPF-IF
      SQFCLG=0.5*(LOGF(FC2)-FACLOG(IABCP+1)
     1       +FACLOG(IABC)+FACLOG(ICAB)+FACLOG(IBCA)
     2       +FACLOG(IAPD)+FACLOG(IAMD)+FACLOG(IBPE)
     3       +FACLOG(IBME)+FACLOG(ICPF)+FACLOG(ICMF))
      NZMIC2=(IB-IC-ID)/2  $  NZMIC3=(IA-IC+IE)/2
      NZMI=XMAXOF(0,NZMIC2,NZMIC3)+1  $  NZMX=XMINOF(IABC,IAMD,IBPE)
      S1=(-1.0)**(NZMI-1)
      DO 400 NZ=NZMI,NZMX
      NZM1=NZ-1  $  NZT1=IABC-NZM1  $  NZT2=IAMD-NZM1
      NZT3=IBPE-NZM1  $  NZT4=NZ-NZMIC2  $  NZT5=NZ-NZMIC3
      TERMLG=SQFCLG-FACLOG(NZ)-FACLOG(NZT1)-FACLOG(NZT2)
     1            -FACLOG(NZT3)-FACLOG(NZT4)-FACLOG(NZT5)
      SSTERM=S1*EXPF(TERMLG)  $  RAC=RAC+SSTERM
  400 S1=-S1
 1000 RETURN
      END
```

```
      SUBROUTINE RAC7
      COMMON FACLOG(500),RAC,IA,IB,IC,ID,IE,IF $  DIMENSION LT(6)
      K1=IA+IB-IE  $  K3=IC+ID-IE  $  K5=IA+IC-IF  $  K7=IB+ID-IF
      K2=IE-XABSF(IA-IB)  $  K4=IE-XABSF(IC-ID)  $  K6=IF-XABSF(IA-IC)
      K8=IF-XABSF(IB-ID)  $  K9=XMINOF(K1,K2,K3,K4,K5,K6,K7,K8)
      RAC=0.0  $  IF (K9) 4000,20,20
   20 K2=K1-2*(K1/2)  $  K4=K3-2*(K3/2)  $  K6=K5-2*(K5/2)
      K8=K7-2*(K7/2)  $  IF(XMAXOF(K2,K4,K6,K8)) 4000,25,4000
   25 LTMIN=XMINOF(IA,IB,IC,ID,IE,IF)  $  IF(LTMIN) 4000,30,150
   30 LT(1)=IA $ LT(2)=IB $ LT(3)=IC $ LT(4)=ID $ LT(5)=IE $ LT(6)=IF
      LTMIN=LT(1)  $  KMIN=1  $  DO 40 N=2,6  $  IF(LT(N)-LTMIN)35,40,40
   35 LTMIN=LT(N)  $  KMIN=N
   40 CONTINUE
      S1=1.0  $  F1=IE  $  F2=IF  $  GO TO (55,55,55,55,45,50),KMIN
   45 F1=IA  $  F2=IC  $  S1=(-1.0)**(K5/2)  $  GO TO 55
   50 F1=IA  $  F2=IB  $  S1=(-1.0)**(K1/2)
   55 RAC=S1/SQRTF((F1+1.)*(F2+1.))  $  GO TO 4000
  150 IABEP=(IA+IB+IE)/2+1  $  ICDEP=(IC+ID+IE)/2+1
      IACFP=(IA+IC+IF)/2+1  $  IBDFP=(IB+ID+IF)/2+1
      IABE=IABEP-IE  $  IEAB=IABEP-IB  $  IBEA=IABEP-IA
      ICDE=ICDEP-IE  $  IECD=ICDEP-ID  $  IDEC=ICDEP-IC
      IACF=IACFP-IF  $  IFAC=IACFP-IC  $  ICFA=IACFP-IA
      IBDF=IBDFP-IF  $  IFBD=IBDFP-ID  $  IDFB=IBDFP-IB
      NZMAX=XMINOF(IABE,ICDE,IACF,IBDF)  $  IABCDI=(IA+IB+IC+ID+4)/2
      IEFMAD=(IE+IF-IA-ID)/2  $  IEFMBC=(IE+IF-IB-IC)/2
      NZMI1=-IEFMAD  $  NZMI2=-IEFMBC  $  NZMIN=XMAXOF(0,NZMI1,NZMI2)+1
      SQLOG=0.5*(FACLOG(IABE)+FACLOG(IEAB)+FACLOG(IBEA)+FACLOG(ICDE)
     1          +FACLOG(IECD)+FACLOG(IDEC)+FACLOG(IACF)+FACLOG(IFAC)
     2          +FACLOG(ICFA)+FACLOG(IBDF)+FACLOG(IFBD)+FACLOG(IDFB)
     3 -FACLOG(IABEP+1)-FACLOG(ICDEP+1)-FACLOG(IACFP+1)-FACLOG(IBDFP+1))
      DO 200 NZ=NZMIN,NZMAX  $  NZM1=NZ-1
      K1=IABCDI-NZM1  $  K2=IABE-NZM1  $  K3=ICDE-NZM1  $  K4=IACF-NZM1
      K5=IBDF-NZM1  $  K6=NZ  $  K7=IEFMAD+NZ  $  K8=IEFMBC+NZ
      SSLOG=SQLOG+FACLOG(K1)-FACLOG(K2)-FACLOG(K3)-FACLOG(K4)
     1          -FACLOG(K5)-FACLOG(K6)-FACLOG(K7)-FACLOG(K8)
      SSTERM=((-1.0)**NZM1)*EXPF(SSLOG)  $  RAC=RAC+SSTERM
  200 CONTINUE
 4000 RETURN
      END
```

References

1) T. Tamura, Revs. Mod. Phys. 37, 679 (1965). This paper will be referred to as T in the text. Equation (25), say, in T will be referred to as Eq. (T-25).

2) T. Tamura, Nucl. Phys. 73, 81 (1965).

3) T. Tamura, Nucl. Phys. 73, 241 (1965).

4) T. Tamura, Prog. Theor. Phys. Supplement No. 37-38, 383 (1966).

5) J. L. C. Ford, Jr., C. Y. Wong, T. Tamura, R. L. Robinson, and P. H. Stelson, Phys. Rev. (to be published).

6) R. Wagner, P. D. Miller, T. Tamura, and H. Marshak, Phys. Rev. 139B, 29 (1965).

7) G. R. Satchler, Phys. Letters 7, 55 (1963).

8) T. Tamura, ORNL-3877, revised (1966).

9) R. H. Bassel, R. M. Drisko, and G. R. Satchler, ORNL-3240 (1962).

10) D. M. Brink and G. R. Satchler, "Angular Momentum" (Clarendon Press, Oxford, 1962).

11) M. Sakai and T. Tamura, Phys. Letters 10, 323 (1964).

12) B. Buck, R. N. Madison, and P. E. Hodgson, Phil. Mag. 5, 118 (1960).

13) H. Marshak, A. C. B. Richardson, and T. Tamura, Phys. Rev. 150, 996 (1966).

14) See, e.g., P. Henrici, "Discrete Variable Methods in Ordinary Differential Equations", Chapter 6 (John Wiley and Sons, New York, 1962).

ORNL-4152
UC-32 — Mathematics and Computers

INTERNAL DISTRIBUTION

1. Biology Library
2-4. Central Research Library
5-6. ORNL — Y-12 Technical Library
Document Reference Section
7-26. Laboratory Records Department
27. Laboratory Records, ORNL R.C.
28. J. B. Ball
29. J. A. Biggerstaff
30. J. L. Fowler
31. C. B. Fulmer
32. C. D. Goodman
33. R. L. Hahn
34. C. E. Larson
35. H. G. MacPherson
36. F. K. McGowan
37. A. Meyer
38. J. K. Poggenburg
39. Russell Robinson
40. G. R. Satchler
41. M. J. Skinner
42. A. H. Snell
43-142. Taro Tamura
143. A. M. Weinberg
144. D. R. Inglis (consultant)
145. T. Lauritsen (consultant)
146. H. W. Newson (consultant)
147. L. Smith (consultant)

EXTERNAL DISTRIBUTION

148. J. A. Swartout, Union Carbide Corporation, New York
149. J. H. Barrett, Mathematics Department, University of Tennessee
150. Eva Sobocinski, University of Notre Dame, Notre Dame, Indiana 46556
151. Research and Development Division, AEC, ORO
152-443. Given distribution as shown in TID-4500 under Mathematics and Computers category (25 copies — CFSTI)