Program SCATMAN: A Code Designed to Calculate
Photon Coherent Scattering Anomalous Scattering Factors
and Cross Sections

Dermott E. Cullen

Lawrence Livermore National Laboratory
P.O. Box 808, L-298
Livermore, CA 94550
November 1, 1989

## Introduction

The computer code SCATMAN starts from photon interaction cross sections in the ENDL character (as opposed to binary) format and calculates photon coherent scattering anomalous scattering factors and cross sections. This code is designed to be both a research code to be used to investigate the effect of various assumptions and approximations and as a production code to be used to produce data for inclusion in the 1989 Evaluated Photon Data Library (EPDL).

## Code documentation

This code is one of a series of codes which are used to create a set of coupled electron-photon data bases which can be used in a wide variety of applications.

These codes are designed to be self documenting, in the sense that the latest documentation for each code is included as comment lines at the beginning of each code. Printed documentation, such as this document, are periodically published and consists mostly of a copy of the comment lines from the beginning of each code. The user should be aware that the comment lines within the codes are continually updated to reflect the most recent status of the codes and these comments should always be considered to be the most recent documentation for the codes and may supercede published documentation, such as this document. Therefore, the user is advised to always read the documentation within the actual code which is being used.

The remainder of this report consists of a listing of the documentation which appears at the beginning of program SCATMAN and example output. See also, "A Consistent Definition of Anomalous Scattering Factors for use in Applications with Atomic Coherent Scattering Form Factors" by D.E. Cullen (to be published) for a detailed discussion of the effect of anomalous scattering on the coherent scattering angular distribution and cross sections.

```
PROGRAM SCATMAN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE10,
1 TAPE11)
```

PROGRAM SCATMAN
VERSION 90-1 (FEBRUARY 1990)

WRITTEN BY DERMOTT E. CULLEN
          L-298
          LAWRENCE LIVERMORE NATIONAL LABORATORY
          P.O. BOX 808
          LIVERMORE CA, 94550
          U.S.A.
TELEPHONE  415-423-7359

PURPOSE
========================================================================
CALCULATE ANOMALOUS SCATTERING FACTORS AND/OR COHERENT CROSS
SECTIONS.

INPUT OPTIONS
========================================================================
THIS CODE HAS BEEN WRITTEN TO INCLUDE A NUMBER OF INPUT OPTIONS
IN ORDER TO ALLOW THE USER TO EVALUATE THE IMPORTANCE OF A
VARIETY OF POSSIBLE DEFINITIONS OF SCATTERING FACTORS, DESCRIBED
BELOW.

THIS CODE HAS ALREADY BEEN USED TO CALCULATE ANOMALOUS SCATTERING
FACTORS FOR THE ENTIRE LIVERMORE EVALUATED PHOTON DATA LIBRARY
(EPDL). THEREFORE IN ORDER TO OBTAIN STANDARD RESULTS YOU NEED
NOT RUN THIS CODE, YOU NEED MERELY CONTACT THE AUTHOR FOR A COPY
OF RESULTS.

HOWEVER, IF YOU WISH TO USE THIS CODE TO EVALUATE THE IMPORTANCE
OF DEFINITIONS, PLEASE USE CARE TO READ ALL OF THIS DOCUMENTATION
IN ORDER TO UNDERSTAND HOW TO PREPARE INPUT PARAMETERS AND HOW
TO INTERPRET THE RESULTS CALCULATED BY THIS CODE.

THE PRIMARY PURPOSE OF THIS CODE IS TO CALCULATE ELEMENTAL
ANOMALOUS SCATTERING FACTORS. THIS CODE CAN BE USED FOR A VARIETY
OF OTHER PURPOSES (SEE, THE DESCRIPTION OF INPUT OPTIONS), BUT
GENERALLY IT IS SIMPLER TO USE THIS CODE TO ESTABLISH A DATA
BASE OF ANOMALOUS SCATTERING FACTORS (WHICH REQUIRES CONSIDERABLE
COMPUTER TIME) AND TO THEN USE THIS DATA BASE IN COMBINATION WITH
A VARIETY OF DIFFERENT MODELS AND ASSUMPTIONS TO CALCULATE THE
PHOTON SCATTERING ANGULAR DISTRIBUTIONS AND CROSS SECTIONS. FOR
THIS PURPOSE 2 OTHER CODES MAY BE USED,

COHERT   = FOR COHERENT SCATTERING
INCOHERT = FOR INCOHERENT SCATTERING

THESE CODES ARE AVAILABLE FROM THE AUTHOR.

OPERATING MODES
================================================================
THIS PROGRAM MAY BE USED IN ONE OF THREE OPERATING MODES
(1) CALCULATE COHERENT CROSS SECTION WITHOUT ANOMALOUS SCATTERING
    - ONLY READ FORM FACTORS AND CALCULATE CROSS SECTIONS
(2) CALCULATE ANOMALOUS SCATTERING FACTORS
    - READ CROSS SECTIONS AND CALCULATE SCATTERING FACTORS
    - THIS MODE MAY ALSO BE USED TO THEN CALCULATE CROSS SECTIONS
(3) CALCULATE CROSS SECTIONS WITH ANOMALOUS SCATTERING
    - READ ANOMALOUS SCATTERING FACTORS AND CALCULATE CROSS
      SECTIONS.

THE OPERATING MODE IS SELECTED BY THE PROGRAM BASED ON INPUT
(1) THIS MODE IS USED IF NO ANOMALOUS SCATTERING DEPENDENT
    OUTPUT IS REQUESTED.
(2) THIS MODE IS USED IF THE ENDL FORMATTED DATA IS CROSS SECTIONS
(3) THIS MODE IS USED IF THE ENDL FORMATTED DATA IS ANOMALOUS
    SCATTERING FACTORS.

THE RECOMMENDED PROCEDURE IS TO USE THIS CODE TO ESTABLISH A DATA
BASE OF SUBSHELL ANOMALOUS SCATTERING FACTORS - THIS DATA BASE
IS EXPENSIVE TO GENERATE, IN TERMS OF COMPUTER TIME, BUT NEED
ONLY BE CREATED ONCE - ONCE IT IS ESTABLISHED IT CAN BE USED
WITH THIS PROGRAM TO ECONOMICALLY DETERMINE THE EFFECT OF ALL
THE MODELS DESCRIBED BELOW.

CONVENTIONS AND DEFINITIONS
================================================================
THIS PROGRAM USES THE CONVENTIONS AND DEFINITIONS OF THE REAL
AND IMAGINARY SCATTERING AMPLITUDES AND SCATTERING FACTORS AS
USED AS GIVEN BY P.P. KANE, L. KISSEL, R.H. PRATT AND S.C. ROY
IN 'ELASTIC SCATTERING OF GAMMA-RAYS AND X-RAYS BY ATOMS',
PHYSICS REPORTS (REVIEW SECTION OF PHYSICS LETTERS) 140, NO. 2
(1986) 75-159, NORTH-HOLLAND, AMSTERDAM.

DEFINITION OF ANGULAR DISTRIBUTIONS AND CROSS SECTIONS
================================================================
THE ANGULAR DISTRIBUTION OF COHERENTLY SCATTERED PHOTONS IS
DEFINED TO BE,

SIG(E,COS) = 3*THOMSON/8*T(E,COS)*((FF+F'(E))**2 + (F''(E))**2)

SIG(E,COS) = ANGULAR DISTRIBUTION IN BARNS PER UNIT COSINE
T(E,COS)   = (1 + COS*COS) (CLASSICAL THOMSON SCATTERING)
THOMSON    = 6.652448D-25 CM*CM = 0.6652448 BARNS
FF         = THE FORM FACTOR
F'(E)      = THE REAL ANOMALOUS SCATTERING FACTOR
F''(E)     = THE IMAGINARY SCATTERING FACTOR
M          = MOMENTUM TRANSFER SIN(THETA/2)/LAMBDA
THETA      = SCATTERING ANGLE
LAMBDA     = WAVELENGTH
COS        = SCATTERING COSINE

THE COHERENT SCATTERING CROSS SECTION IS DEFINED BY INTEGRATING
THE ANGULAR DISTRIBUTION OVER COS.

ANOMALOUS SCATTERING
================================================================
WE EXPECT ANOMALOUS SCATTERING TO BE IMPORTANT AT LOW ENERGY AND
THAT ITS EFFECT DECREASES WITH INCREASING ENERGY SO THAT AT HIGH
ENERGY THE ANGULAR DISTRIBUTION AND CROSS SECTION SMOOTHLY
APPROACH THAT PREDICTED ONLY USING FORM FACTORS, I.E., AT HIGH
ENERGY F'(E) AND F''(E) APPROACH ZERO.

IN ORDER TO ASSESS THE IMPORTANCE OF THE FORM FACTOR AND THE
ANOMALOUS SCATTERING THIS PROGRAM CAN BE USED TO CALCULATE CROSS
CROSS SECTIONS USING 3 DIFFERENT MODELS,
(1) INCLUDING THE FORM FACTOR AND ANOMALOUS SCATTERING.
(2) INCLUDING ANOMALOUS SCATTERING AND ASSUMING THE FORM FACTOR
    IS EQUAL TO Z (THE LOW ENERGY LIMIT OF THE FORM FACTOR).
(3) INCLUDING ONLY THE FORM FACTOR (NO ANOMALOUS SCATTERING)

WE EXPECT (1) AND (2) TO BECOME EQUAL AT LOW ENERGY WHERE THE
FORM FACTOR IS APPROACHING Z.

WE EXPECT (1) AND (3) TO BECOME EQUAL AT HIGH ENERGY WHERE THE
THE ANOMALOUS SCATERING IS APPROACHING ZERO.

DEFINITION OF FORWARD ANGLE SCATTERING AMPLITUDES
================================================================
THE FORWARD DIRECTION ANGULAR DISTRIBUTION CAN BE WRITTEN IN THE
FORM,

SIG(E,1) = ((R A(E,1))**2 + (I A(E,1))**2)

R A(E,1) = FORWARD ANGLE REAL SCATTERING FACTOR
I A(E,1) = FORWARD ANGLE IMAGINARY SCATTERING AMPLITUDE
COS      = 1

DEFINITION OF FORWARD ANGLE ANOMALOUS SCATTERING FACTORS
================================================================
FROM OUR INITIAL DEFINITION OF THE ANGULAR DISTRIBUTION WE CAN
DEFINE THE FORWARD ANGLE ANOMALOUS SCATTERING FACTORS, WHICH
USING THE SIGN CONVENTION OF KISSEL, CAN BE WRITTEN IN THE FORM,

R A(E,1) = -R0*(Z + F'(E))
I A(E,1) =  R0*(F''(E))

Z        = FORWARD ANGLE VALUE OF THE FORM FACTOR
R0       = 2.82D-13 CM = 0.282 SQRT(BARNS)

OR IF WE KNOW THE SCATTERING AMPLITUDES,

F'(E)    = -(R A(E,1)/R0 + Z)
F''(E)   =   I A(E,1)/R0

IN THIS CODE WE WILL USE DISPERSION RELATIONSHIPS TO DEFINE THE
AMPLITUDES AND THEN USE THE ABOVE RELATIONSHIPS TO DEFINE THE
SCATTERING FACTORS.

4

FORWARD ANGLE SCATTERING AMPLITUDES IN TERM OF CROSS SECTION
================================================================
DISPERSION RELATIONSHIPS WILL BE USED TO CALCULATE ANOMALOUS
SCATTERING AMPLITUDES. THE FORWARD ANGLE IMAGINARY AND REAL
SCATTERING AMPLITUDE CAN BE DEFINED IN TERMS OF ONE ANOTHER
USING THE TWO DISPERSION RELATIONSHIPS,

$$R A(E,1) = C1*E*E*INTEGRAL \frac{I A(E',0)*DE'}{E'*(E'*E' - E*E)}$$

$$I A(E,1) = -C1*E*INTEGRAL \frac{R A(E',0)*DE'}{(E'*E' - E*E)}$$

WHERE THE INTEGRALS EXTEND FROM E' = 0 TO INFINITY.

IN ADDITION THE IMAGINARY SCATTERING AMPLITUDES CAN BE DEFINED
DIRECTLY IN TERMS OF THE PHOTOELECTRIC CROSS SECTION,

I A(E,1) = C2*E*SIGPE(E)

R A(E,1) = FORWARD ANGLE REAL ANOMALOUS SCATTERING AMPLITUDE
I A(E,1) = FORWARD ANGLE IMAGINARY ANOMALOUS SCATTERING AMPLITUDE
SIGPE(E) = PHOTOELECTRIC CROSS SECTION
C1       = 2/PI
C2       = 1/(4*PI*C)

WE WILL USE THE THIRD EQUATION ABOVE TO DIRECTLY DEFINE THE
IMAGINARY AMPLITUDE IN TERMS OF THE PHOTOELECTRIC CROSS SECTION
AND SUBSTITUTE THIS RELATIONSHIP INTO THE FIRST EQUATION TO
DEFINE THE REAL AMPLITUDE IN TERMS OF AN INTEGRAL OVER THE
PHOTOELECTRIC CROSS SECTION.

$$R A(E,1) = C3*E*E*INTEGRAL \frac{SIGPE(E')*DE'}{(E'*E' - E*E)}$$

C3       = C1*C2 = 1/(2*PI*PI*C)

# CONSIDERATION OF LIMITING VALUES
=================================
## LOW ENERGY
==========
AT LOW ENERGY THE FORM FACTOR AND ANOMALOUS SCATTERING FACTORS
BECOME ISOTROPIC. THEREFORE THE ANGULAR DISTRIBUTION BECOMES
PROPORTIONAL TO THOMSON SCATTERING, 1 +COS**2, AND THE FORWARD
ANGLE SCATTERING AMPLITUDES CAN BE USED TO DEFINE THE ANGULAR
DISTRIBUTION AT ALL ANGLES AND IN TURN THE CROSS SECTION.

FROM THE DEFINITION OF THE SCATTERING AMPLITUDES IN TERMS OF THE
PHOTOELECTRIC CROSS SECTION WE CAN SEE THAT BELOW THE LOWEST
ENERGY EDGE I A(E,1) IS ZERO AND AS LONG AS THE INTEGRAL DEFINING
R A(E,1) CONVERGES AS ENERGY APPROACHES ZERO (AS IT ALWAYS DOES)
R A(E,1) WILL APPROACH ZERO AS E**2. THEREFORE THE ANGULAR
DISTRIBUTION AND CROSS SECTION WILL APPROACH ZERO AS E**4. THIS
IS IN MARKED CONTRAST TO THE RESULTS OBTAINED USING ONLY FORM
FACTORS, WHICH APPROACH A CONSTANT VALUE PROPORTIONAL TO Z**2
AT LOW ENERGY (Z = ATOMIC NUMBER OF AN ELEMENT).

SINCE THE REAL SCATTERING AMPLITUDE APPROACHES ZERO AT LOW ENERGY
WE CAN SEE FROM OUR DEFINITION OF THE REAL SCATTERING FACTOR THAT
IT MUST APPROACH -Z, EXACTLY CANCELLING THE CONTRIBUTION OF THE
FORM FACTOR, IN ORDER FOR THE CROSS SECTION TO APPROACH ZERO.

AS WILL BE POINTED OUT BELOW, DEPENDING ON THE MODEL USED TO
ACTUALLY DEFINE SCATTERING FACTORS, THIS LOW ENERGY LIMIT OF A
ZERO CROSS SECTION MAY OR MAY NOT BE ATTAINED.

## HIGH ENERGY
===========
TWO ADDITIONAL RELATIONSHIPS WILL BE USED TO DEFINE THE HIGH
ENERGY LIMIT AS ENERGY APPROACHES INFINITY,

I A(E,1) --------> 0

R A(E,1) --------> -C3*INTEGRAL SIGPE(E')*DE'

IN ORDER TO SATISFY THESE RELATIONSHIPS, FROM THE DEFINITION OF
I A(E,1) IN TERMS OF THE PHOTOELECTRIC CROSS SECTION WE SEE
THAT WE REQUIRE THAT,

E*SIGPE(E) -------> 0, AS E APPROACHES INFINITY

AND THAT THE INTEGRAL OF SIGPE(E) BE BOUNDED.

AT HIGH ENERGY THE PHOTOELECTRIC CROSS VARIES AS 1/E AND AS
SUCH WILL NOT SATISFY EITHER OF THESE RELATIONSHIPS. FOR A
1/E CROSS SECTION I A(E,1) WILL APPROACH A CONSTANT AND
THE INTEGRAL DEFINING R A(E,1) WILL DIVERSE TO -INFINITY.

IN ORDER TO SATISFY THESE RELATIONSHIPS WE MUST INCLUDE THE
EFFECT OF BOUND PAIR PRODUCTION AT HIGH ENERGY (SEE: REF. 1).
THE BOUND PAIR PRODUCTION CROSS SECTION IS NOT THE SAME AS
THE PAIR PRODUCTION. THE BOUND PAIR PRODUCTION WILL BE
ZERO AT LOW ENERGY. IN THE 1 TO 10 MEV ENERGY RANGE IT WILL
RAPIDLY INCREASE AND BECOME EQUAL TO THE PHOTOELECTRIC
CROSS SECTION.

THE CORRECT CROSS SECTION TO USE TO DEFINE THE SCATTERING
AMPLITUDES IS THE PHOTOELECTRIC MINUS THE BOUND PAIR
PRODUCTION CROSS SECTION.

THEREFORE THE CORRECT CROSS SECTION TO USE SHOULD BE THE
PHOTOELECTRIC UP TO 1 MEV. ABOVE THIS ENERGY THE CROSS
SECTION SHOULD SMOOTHLY APPROACH ZERO AT 10 MEV AND BE
ZERO FOR ALL HIGHER ENERGIES.

BY 1 MEV THE PHOTOELECTRIC CROSS SECTION IS GENERALLY QUITE
SMALL AND THE DETAILS OF HOW THE CROSS SECTION APPROACHES
ZERO IN THE 1 TO 10 MEV ENERGY RANGE HAS LITTLE EFFECT ON
R A(E,1). THE ONLY IMPORTANT POINT IS THAT IT NOT BE DEFINED
AS EXTENDING AS 1/E TO INFINITY. IN THIS CODE ABOVE 1 MEV THE
CROSS SECTION WILL BE DEFINED TO BE CONTINUOUS AT 1 MEV AND
VARIE BETWEEN 1 AND 10 MEV AS PROPORTIONAL TO (10 - E)/E**3
AND TO BE ZERO ABOVE 10 MEV.

WE CAN SEE THAT IN ORDER TO OBTAIN CONSISTENT RESULTS WHICH
SMOOTHLY APPROACH THE RESULTS OBTAINED USING ONLY FORM FACTORS,
FROM THE DEFINITION OF OUR SCATTERING AMPLITUDES AND FACTORS WE
MUST HAVE,

R A(E,1) --------> -C3*INTEGRAL SIGPE(E')*DE' = -R0*Z

SINCE ONLY IN THIS CASE WILL F'(E) APROACH ZERO AT HIGH ENERGY.
IT WILL BE POINTED OUT BELOW THAT IN ALMOST NO REAL CASE IS THIS
LIMITING VALUE OBTAINED AND UNLESS ADDITIONAL ASSUMPTIONS OR
APPROXIMATIONS ARE INTRODUCED, THE RESULTS WILL BE INCONSISTENT.

EXPLICIT AND IMPLICIT EDGES
========================================================================
IT MUST BE REALIZED THAT THE INTEGRAL USED TO DEFINE THE REAL
SCATTERING AMPLITUDE EXTENDS FROM ZERO TO INFINITE ENERGY. NEAR
PHOTOELECTRIC EDGES, WHERE THERE IS AN ABRUPT CHANGE IN THE
CROSS SECTION WE WILL OBTAIN EXPLICIT EDGE EFFECTS, WHERE THE
REAL SCATTERING AMPLITUDE CHANGES VERY RAPIDLY.

IF CROSS SECTIONS ARE ONLY AVAILABLE OVER A SMALLER ENERGY RANGE
AND THESE ARE USED TO CALCULATE SCATTERING AMPLITUDES, ONE MUST
REALIZE THAT IF THE INTEGRAL IS TRUNCATED TO THE ENERGY RANGE OF
THE CROSS SECTION YOU ARE IMPLICITLY SAYING THERE IS AN EDGE AT
THE END OF THE RANGE OF INTEGRATION AND YOU SHOULD EXPECT TO
OBTAIN RESULTS INCLUDING IMPLICIT EDGE EFFECTS NEAR THE END OF
OF THE ENERGY RANGE OF INTEGRATION. FOR EXAMPLE, MATHEMATICALLY
THERE IS ABSOLUTELY NO DIFFERENCE BETWEEN YOU SPECIFYING THAT
THERE IS AN EDGE AT 10 EV AND THE CROSS SECTION IS ZERO AT ALL
LOWER ENERGY - AND YOU USING CROSS SECTIONS THAT ONLY EXTEND DOWN
TO 10 EV AND THEN SIMPLY TRUNCATE THE INTEGRAL AT THIS POINT -
IN EITHER CASE THE RESULTS WILL BE THE SAME AND WILL INCLUDE
TYPICAL EDGE EFFECTS - THE SAME IS TRUE AT THE UPPER ENERGY
LIMIT OF THE RANGE OF INTEGRATION.

DATA DOWN TO EDGE AND UP TO INFINITY
================================================================
THE TABULATED CROSS SECTIONS USED BY THIS PROGRAM ARE GIVEN OVER
THE ENERGY RANGE 10 EV TO 100 GEV. THE DISPERSION INTEGRAL IS
OVER 0 TO INFINITE ENERGY. IF THE EDGE FOR A SUBSHELL IS BELOW
10 EV THE CROSS SECTION CAN BE EXTENDED FROM 10 EV DOWN TO THE
EDGE AS $1/E^{**}3$ WITH CONTINUITY AT 10 EV. AS DESCRIBED ABOVE THE
CROSS SECTION WILL BE DEFINED TO SMOOTHLY APPROACH ZERO AT 10 MEV
AND TO BE ZERO AT ALL HIGHER ENERGIES.

WITH THIS APPROACH THE CROSS SECTIONS ARE DEFINED AT ALL ENERGIES
AND IMPLICIT EDGE EFFECTS CAN BE AVOIDED.

HOWEVER, IF YOU WISH TO EVALUATE THE EFFECT OF IMPLICIT EDGES YOU
HAVE THE INPUT OPTION NOT TO EXTEND CROSS SECTIONS BELOW 10 EV.
IN THIS CASE YOU EXPECT TO SEE IMPLICIT EDGE EFFECTS NEAR 10 EV.

RESOLUTION
==========
IT CAN BE SHOWN THAT AS LONG AS THE CROSS SECTION USED TO DEFINE
THE ANOMALOUS SCATTERING AMPLITUDES IS CONTINUOUS ONE CAN OBTAIN
CONSISTENT, BOUNDED ANOMALOUS SCATTERING AMPLITUDES. OBVIOUSLY
IF WE TREAT THE PHOTOELECTRIC CROSS SECTION AS HAVING INFINITELY
NARROW EDGES WE WILL HAVE A DISCONTINUITY IN THE CROSS SECTION
WHICH WILL PREVENT THE RESULTS FROM BEING CONSISTENT AND BOUNDED,
E.G., THE REAL AMPLITUDE WILL HAVE A LOG POLE AT THE EDGES.

FOR USE IN APPLICATIONS WE REQUIRE THAT THE AMPLITUDES BE
CONSISTENT AND BOUNDED AND AS SUCH WE MUST DEFINE CONTINUOUS
CROSS SECTIONS FOR USE IN EVALUATING THE DISPERSION INTEGRALS.

BY CONSISTENT WE MEAN THAT THE RESULTS ARE CONSISTENT WITH THE
TWO DISPERSION RELATIONSHIPS DEFINING THE REAL AND IMAGINARY
AMPLITUDES IN TERMS OF ONE ANOTHER AND THE DIRECT RELATIONSHIP
USED TO DEFINE THE IMAGINARY AMPLITUDE IN TERMS OF THE CROSS
CROSS. BY COMBINING THESE THREE RELATIONSHIPS WE CAN OBTAIN A
HOMOGENEOUS INTEGRAL EQUATION ONLY INVOLVING THE CROSS SECTION.
AS SUCH WE CANNOT EXPECT TO START FROM MERELY ANY CROSS SECTION
AND OBTAIN RESULTS WHICH ARE CONSISTENT WITH ALL THREE OF OUR
RELATIONSHIPS - WE MUST PLACE RESTRAINTS ON THE CROSS SECTION
IF WE ARE TO OBTAIN CONSISTENT RESULTS - THE MOST OBVIOUS
CONSTRAINT THAT WE MUST PLACE ON THE CROSS SECTION IS THAT IT BE
CONTINUOUS.

ONE CAN ARGUE THAT PHOTOELECTRIC EDGES CANNOT BE INFINITELY NARROW
DUE TO FINITE LIFETIMES, ETC. AND USE THIS ARGUMENT TO VARY THE
CROSS CONTINUOUSLY ACROSS THE EDGE. WE WILL TAKE A MORE PRAGMATIC
APPROACH AND MERELY STATE THAT WE WISH TO DEFINE ANOMALOUS
SCATTERING AMPLITUDES FOR SUBSEQUENT USE IN APPLICATIONS, IN
PARTICULAR WE WISH TO SIMULATE EXPERIMENTAL CONDITIONS IN ORDER TO
BE ABLE TO REPRODUCE EXPERIMENTAL MEASUREMENTS. SINCE MEASUREMENTS
HAVE A FINITE ENERGY RESOLUTION WE WILL DEFINE OUR ANOMALOUS
SCATTERING AMPLITUDES IN ORDER TO SIMULATE THIS RESOLUTION.

WE WILL SIMULATE RESOLUTION BROADENING BY STARTING FROM THE
PHOTOELECTRIC CROSS SECTION AND DEFINING A RESOLUTION BROADENED
CROSS SECTION,

$$SIGB(E,A) = \frac{INT\ SIGPE(E')*EXP[-A*(E'-E)**2]*DE'}{INT\ EXP[-A*(E'-E)**2]*DE'}$$

EVEN IN THE CASE WHERE THE PHOTOELECTRIC CROSS SECTION IS TREATED
AS HAVING INFINITELY NARROW EDGES, THE BROADENED CROSS SECTION
WILL BE SMOOTH, NOT HAVE ANY DISCONTINUITIES AND CAN YIELD RESULTS
WHICH ARE CONSISTENT WITH BOTH DISPERSION RELATIONSHIPS AND THE
DIRECT DEFINITION OF THE IMAGINARY AMPLITUDE IN TERMS OF THE
PHOTOELECTRIC CROSS SECTION.

THE WIDTH OF THE GAUSSIAN BROADENING KERNEL (A) WILL BE DEFINED
TO PRODUCE A RESOLUTION WHICH IS A FRACTION OF THE ENERGY. THE
WIDTH OF THE GAUSSIAN AT ITS HALF HEIGHT IS,

1/2      = EXP[-A*(E'-E)**2]

FOR E'-E TO BE A FRACTION OF THE ENERGY WE DEFINE,

E'-E     = DE*E

1/2      = EXP[-A*(DE*E)**2]

LOG(1/2) = -A*(DE*E)**2
LOG(2)   =  A*(DE*E)**2

A        =LOG(2)/[(DE*E)**2]

DE IS ONE OF THE INPUT PARAMETERS TO THIS PROGRAM. IF YOU WISH TO
EVALUATE THE EFFECT OF RESOLUTION ON THE RESULTS YOU MAY VARY
THIS PARAMETER. FOR GENERAL USE IN ORDER TO SIMULATE CURRENT
EXPERIMENTAL RESOLUTION, 1 PER-CENT IS RECOMMENDED AS AN INPUT
PARAMETER.

TABULATED LINEARLY INTERPOLABLE CROSS SECTIONS
==================================================
THE CROSS SECTIONS IN THE LIVERMORE EVALUATED PHOTON DATA LIBRARY
ARE DEFINED IN TERMS OF A TABLE OF (ENERGY, CROSS SECTION) PAIRS
WITH LOG-LOG INTERPOLATION BETWEEN TABULATED VALUES. THIS DATA CAN
EASILY BE REDUCED TO WITHIN ANY REQUIRED ACCURACY TO A TABLE OF
(ENERGY, CROSS SECTION) PAIRS WITH LINEAR-LINEAR INTERPOLATION
BETWEEN TABULATED VALUES. THIS PROCEDURE IS USED IN THIS CODE IN
ORDER TO PRODUCE A TABLE OF LINEARLY INTERPOLATED CROSS SECTIONS
TO BE USED TO CALCULATE THE RESOLUTION BROADENED CROSS SECTIONS.

THE INTERVAL HALVING TECHNIQUE IS THEN USED TO CALCULATE THE
RESOLUTION BROADENED CROSS SECTION TO PRODUCE TABULATED RESULTS
WHERE LINEAR INTERPOLATION CAN BE USED TO DEFINE THE CROSS
SECTION BETWEEN TABULATED VALUES. THIS RESOLUTION BROADENED,
TABULATED LINEARLY INTERPOLABLE CROSS SECTION IS THEN USED TO
DEFINE THE SCATTERING AMPLITUDES.

STARTING FROM THIS DATA THE IMAGINARY SCATTERING AMPLITUDE CAN BE
DEFINED DIRECTLY IN TERMS OF THE CROSS SECTION AND THE REAL
SCATTERING AMPLITUDE CAN BE DEFINED IN TERMS OF THE INTEGRAL
OVER A NUMBER OF ENERGY INTERVALS, IN EACH OF WHICH THE CROSS
SECTION IS LINEARLY INTERPOLABLE,

# CALCULATION OF FORWARD ANGLE SCATTERING AMPLITUDES
==========================================================

STARTING FROM TABULATED DATA WE CAN DEFINE THE INTEGRAL AS A SUM
OVER ENERGY INTERVALS IN THE TABLUATED DATA,

$$R \; A(E,1) = C3*E*E* \; \text{SUM (K) INTEGRAL (EK - EK+1)} \; \frac{SIGB(E')*DE'}{(E'*E' - E*E)}$$

THEREFORE IN ORDER TO DEFINE R A(E,1) WE NEED ONLY BE CONCERNED
WITH DEFINING THIS INTEGRAL FOR ONE SUCH INTERVAL AND THEN SUMMING
OVER ALL CONTRIBUTING INTERVALS. ALTERNATIVELY WE MAY EXPRESS THIS
INTEGRAL AS THE SUM OVER CONTRIBUTIONS FROM EACH TABULATED CROSS
SECTION POINT, WHICH IS WHAT WE WILL ACTUALLY DO.

# CALCULATION OF INTEGRALS
===========================

IF WE START WITH A TABLE OF CROSS SECTIONS WHICH ARE LINEARLY
INTERPOLABLE THEN THE CROSS SECTION MAY BE EXPRESSED IN A FORM IN
WHICH A CROSS SECTION WHICH IS TABULATED AT ENERGY EK CONTRIBUTES
TO THE CROSS SECTION IN A LINEAR FASHION ONLY IN THE ENERGY
INTERVALS EK-1 TO EK AND EK TO EK+1 WHERE THE CONTRIBUTION OF
THE POINT IS,

```
SIGB(EK)*(E - EK-1)/(EK - EK-1)  : E FROM EK-1 TO EK
SIGB(EK)*(EK+1 - E)/(EK+1 - EK)  : E FROM EK TO EK+1
0                                : E LESS THAN EK-1
                                   OR GREATER THAN EK+1
```

NOTE, AT EK THE CROSS SECTION IS CONTINUOUS AND REDUCES TO THE
TABULATED VALUE SIGB(EK). BETWEEN EK-1 AND EK THE CONTRIBUTION
VARIES FROM ZERO AT EK-1 TO SIGB(EK) AT EK. BETWEEN EK AND EK+1
THE CONTRIBUTION VARIES FROM SIGB(EK) AT EK TO ZERO AT EK+1. FOR
ENERGIES LESS THAN EK-1 OR GREATER THAN EK+1 THERE IS NO
CONTRIBUTION FROM THIS TABULATED POINT.

CONSISTENT WITH THE ABOVE DEFINITIONS WE WILL ASSUME THAT THE
TABULATED CROSS SECTIONS STARTS WITH A ZERO CROSS SECTION (AT
AN ENERGY BELOW THE LOWEST LYING EDGE WHERE THE RESOLUTION
BROADENING WILL SMOOTHLY REDUCE IT TO ZERO) AND ENDS WITH A
ZERO CROSS SECTION (AT 10 MEV WHERE THE COMBINATION OF THE
PHOTOELECTRIC AND BOUND PAIR PRODUCTION SMOOTHLY APPROACH
ZERO).

WITH THIS DEFINITION WE NEED ONLY BE CONCERNED WITH THE
CONTRIBUTION OF ONE TABULATED DATA POINT TO THE INTEGRAL TO
DEFINE THE REAL SCATTERING AMPLITUDE. THE INTEGRAL WILL THEN
MERELY BE A SUM OVER ALL TABULATED POINTS.

EFFECT OF ONE DATA POINT
=========================
SEPARATION OF DISPERSION RELATIONSHIP INTO 2 SIMPLE POLES
===========================================================

$$R \; A(E,1) = C1*E*E*INTEGRAL \; \frac{SIG(E')*DE'}{(E'*E' - E*E)}$$

$$= C*E*E*INTEGRAL \; \frac{SIG(E')*[(E' + E) - (E' - E)]*DE'/(2*E)}{(E' - E)*(E' + E)}$$

$$= C*E/2*INTEGRAL \; [SIG(E')*DE'/(E' - E) \\ - SIG(E')*DE'/(E' + E)]$$

$$= C*E/2*[F(E) - F(-E)]$$

$$F(E) \quad = INTEGRAL \; [SIG(E')*DE'/(E' - E)]$$

LINEARLY INTERPOLABLE TABULATED CROSS SECTIONS
================================================
A CROSS SECTION TABULATED AT E1 WILL BE LINEARLY INTERPOLABLE
BETWEEN E0 AND E1, WILL BE ZERO AT E0 AND SIG(E1) AT E1. SIMILARLY
IT WILL BE LINEARLY INTERPOLABLE BETWEEN E1 AND E2, WILL BE
SIG(E1) AT E1 AND ZERO AT E2,

```
SIG(E)    = SIG(E1)*(E - E0)/(E1 - E0)  : E = E0 TO E1
          = SIG(E1)*(E2 - E)/(E2 - E1)  : E = E1 TO E2
          = 0                            : E < E0 OR E > E2
```

CALCULATION OF F(E)
=====================

```
F(E)      = SIG(E1)*[
            INTEGRAL{(E' - E0)*DE'/(E' - E)/(E1 - E0)}
           +INTEGRAL{(E2 - E')*DE'/(E' - E)/(E2 - E1)}]

F(E)      = SIG(E1)*G(E)

G(E)      = INTEGRAL{(E' - E0)*DE'/(E' - E)/(E1 - E0)}
           +INTEGRAL{(E2 - E')*DE'/(E' - E)/(E2 - E1)}
```

WHERE THE FIRST INTEGRAL IS FROM E0 TO E1 AND THE SECOND INTEGRAL
FROM E1 TO E2.

NOTE THAT G(E) ONLY DEPENDS ON ENERGY, E, E0, E1 AND E2 AND IS
THE RESPONSE THAT ONE OBTAINS BY PLACING 1 BARN OF CROSS SECTION
AT EK.

ADDING AND SUBTRACTING E,

```
G(E)       = INTEGRAL{(E' - E + E - E0)*DE'/(E' - E)/(E1 - E0)}
            +INTEGRAL{(E2 - E + E - E')*DE'/(E' - E)/(E2 - E1)}


G(E)       = INTEGRAL{DE' + (E - E0)*DE'/(E' - E)}/(E1 - E0)
            +INTEGRAL{(E2 - E)/(E' - E)*DE' - DE'}/(E2 - E1)
```

SINCE,
```
          INTEGRAL{DE'/(E1 - E0)} = 1
          INTEGRAL{DE'/(E2 - E1)} = 1
```

```
G(E)       = 1 + (E - E0)/(E1 - E0)*INTEGRAL{DE'/(E' - E)}
            -1 + (E2 - E)/(E2 - E1)*INTEGRAL{DE'/(E' - E)}


G(E)         (E - E0)/(E1 - E0)*INTEGRAL{DE'/(E' - E)}
            +(E2 - E)/(E2 - E1)*INTEGRAL{DE'/(E' - E)}
```

IN THIS FORM WE CAN IMMEDIATELY INTEGRATE TO FIND,

```
G(E)       = (E - E0)/(E1 - E0)*LOG[|E - E1|/|E - E0|]
            +(E2 - E)/(E2 - E1)*LOG[|E - E2|/|E - E1|]
```

IN THIS FORM IT APPEARS THAT WE HAVE LOGARITHMIC POLES AT E0, E1
AND E2.

## A NUMERICALLY STABLE DEFINITION OF F(E)
==========================================
IN  THE ABOVE FORM IT IS NOT IMMEDIATELY OBVIOUS THAT THIS
DEFINITION IS NUMERICALLY STABLE, IN THE SENSE OF BEING BOUNDED
FOR ALL VALUES OF E. IN ORDER TO DEMONSTRATE THAT IT IS BOUNDED
WE WILL CONVERT THE DEFINITION TO A SUM OF TERMS OF THE FORM,

X*LOG[|X|]

AND NOTE THAT,

LIM X*LOG[|X|] -----> 0 AS X -----> 0

WE CAN USE THE RELATIONSHIP,

LOG[|X|/|Y|] = LOG[|X|/|Z|] - LOG[|Y|/|Z|]

TO REWRITE G(E) IN THE FORM,

```
G(E)       =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
            +(E - E0)/(E1 - E0)*LOG[|E - E1|/|E1 - E0|]
            -(E - E2)/(E1 - E2)*LOG[|E - E1|/|E2 - E1|]
            +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E2 - E1|]
```

IN THIS FORM THE FIRST AND LAST TERMS ARE ALREADY IN THE REQUIRED
FORM OF X*LOG[|X|] AND WE NEED ONLY CONSIDER THE SECOND AND
THIRD TERMS. ADDING AND SUBTRACTING TERMS,

```
G(E)      =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
           +(E - E1 + E1 - E0)/(E1 - E0)*LOG[|E - E1|/|E1 - E0|]
           -(E - E1 + E1 - E2)/(E1 - E2)*LOG[|E - E1|/|E2 - E1|]
           +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E2 - E1|]

G(E)      =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
           +(E - E1)/(E1 - E0)*LOG[|E - E1|/|E1 - E0|]+LOG[|E - E1|
           -(E - E1)/(E1 - E2)*LOG[|E - E1|/|E2 - E1|]-LOG[|E - E1|
           +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E2 - E1|]
```

IN THIS FORM WE CAN SEE THAT THE TWO PURE LOG[|E - E1|] TERMS
EXACTLY CANCEL AND WE FIND,

```
G(E)      =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
           -(E - E1)/(E0 - E1)*LOG[|E - E1|/|E0 - E1|]
           +(E - E1)/(E2 - E1)*LOG[|E - E1|/|E2 - E1|]
           +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E1 - E2|]
           +LOG[|E2 - E1|/|E1 - E0|]
```

UPON DEFINING,

```
X0      = (E - E0)/(E1 - E0)
X1      = (E - E1)/(E0 - E1)
X2      = (E - E1)/(E2 - E1)
X3      = (E - E2)/(E1 - E2)
```

THE SOLUTION CAN BE WRITTEN IN THE FORM,

```
G(E)      =- X0*LOG[|X0|] - X1*LOG[|X1|] + X2*LOG[|X2|]
           + X3*LOG[|X3|] + LOG[|E2 - E1|/|E1 - E0|]
```

SIMILARLY IN ORDER TO DEFINE G(-E) WE NEED ONLY CHANGE E TO -E AND
UPON DEFINING,

```
X4      =-(E + E0)/(E1 - E0)
X5      =-(E + E1)/(E0 - E1)
X6      =-(E + E1)/(E2 - E1)
X7      =-(E + E2)/(E1 - E2)
```

AND THE SOLUTION CAN BE WRITTEN IN THE FORM,

```
G(-E)     =- X4*LOG[|X4|] - X5*LOG[|X5|] + X6*LOG[|X6|]
           + X7*LOG[|X7|] + LOG[|E2 - E1|/|E1 - E0|]
```

COMBINING THESE TWO DEFINITIONS AND NOTING LOG[|E2-E1|/|E1-E0|]
TERMS EXACTLY CANCEL, THE SOLUTION CAN FINALLY BE WRITTEN IN THE
FORM,

```
G(E)-G(-E) = - X0*LOG[|X0|] - X1*LOG[|X1|] + X2*LOG[|X2|]
             + X3*LOG[|X3|]
             + X4*LOG[|X4|] + X5*LOG[|X5|] - X6*LOG[|X6|]
             - X7*LOG[|X7|]
```

AS NOTED ABOVE, SINCE,

LIM X*LOG(X) -----> 0 AS X APPROACHES 0

IN THIS FORM THE DEFINITION IS BOUNDED AND CONTINUOUS AS THE
ARGUMENT ANY OF THE LOG TERMS APPROACHES ZERO.

SOLUTION ALGORITHM
==================
FOR E MUCH LESS THAN E0 OR MUCH GREATER THAN E2 THE FIRST
DEFINITION OF G(E) CAN BE ECONOMICALLY USED - THIS RELATIONSHIP
FOR G(E) ONLY INVOLVES 2 LOGS.

G(E)      = (E - E0)/(E1 - E0)*LOG[|E - E1|/|E - E0|]
            +(E2 - E)/(E2 - E1)*LOG[|E - E2|/|E - E1|]


G(-E)     =-(E + E0)/(E1 - E0)*LOG[|E + E1|/|E + E0|]
            +(E2 + E)/(E2 - E1)*LOG[|E + E2|/|E + E1|]

FOR E NEAR THE INTERVAL E0 TO E2 WE WILL USE THE MORE COMPLICATED
BUT COMPLETELY NUMERICALLY STABLE DEFINITION,

G(E)-G(-E) = - X0*LOG[|X0|] - X1*LOG[|X1|] + X2*LOG[|X2|]
              + X3*LOG[|X3|]
              + X4*LOG[|X4|] + X5*LOG[|X5|] - X6*LOG[|X6|]
              - X7*LOG[|X7|]

THIS ALGORITHM CAN BE USED TO CALCULATE RESULTS AT ANY GIVEN
ENERGY E.

CONSISTENT LOW AND HIGH ENERGY RESULTS
======================================================================
AS POINTED OUT ABOVE THE RESULTS CALCULATED BY THIS CODE WILL BE
CONSISTENT ONLY IF THE INTEGRAL OF THE CROSS SECTION PROPERLY
NORMALIZED APPROACHES Z AT HIGH ENERGY. ONLY IN THIS CASE WILL
THE RESULTS APPROACH THOSE USING ONLY FORM FACTORS AT HIGH ENERGY.

SINCE OUR CROSS SECTIONS HAVE AN UNCERTAINTY ASSOCIATED WITH THEM
AND SINCE CERTAIN DATA IS MISSING, E.G., WE ONLY HAVE IONIZATION
DATA, BUT NOT EXCITATION DATA, WE SHOULD NOT BE SURPRISED TO FIND
THAT IN VIRTUALLY ALL REAL CASES OF INTEREST WE WILL NOT BE ABLE
TO CALCULATE RESULTS WITH EXACTLY THE REQUIRED HIGH ENERGY LIMIT.

THIS CODE WILL ALLOW YOU TO CALCULATE RESULTS USING ONE OF THREE
CONVENTIONS, IN AN ATTEMPT TO OBTAIN CONSISTENT RESULTS,

CONSERVE R A(,0)
================
USE THE RESULTS EXACTLY AS CALCULATED WITHOUT ANY CORRECTION
FACTORS. IN THIS CASE THE REAL SCATTERING FACTOR IS DEFINED TO
BE,
F'(E)     = -(R A(E,1)/R0 + Z)

THESE RESULTS WILL APPROACH THE CORRECT LOW ENERGY LIMIT OF A
ZERO CROSS SECTION, BUT GENERALLY WILL NOT APPROACH THE CORRECT
HIGH ENERGY LIMIT, SINCE IN ALMOST ALL CASES THE INTEGRAL OF
THE CROSS SECTION DOES NOT APPROACH -Z.

SHIFT TO -Z
===========
SHIFT THE RESULTS TO FORCE THEM TO APPROACH THE CORRECT HIGH
ENERGY LIMIT. IN THIS CASE THE REAL SCATTERING FACTOR IS DEFINED
TO BE,
F'(E)     = -(R A(E,1)/R0 - R A(INFINITY,0))
WHICH BY DEFINITION APPROACHES ZERO AS E APPROACHES INFINITY.
THESE RESULTS WILL APPROACH THE CORRECT HIGH ENERGY LIMIT, BUT
GENERALLY WILL NOT APPROACH THE CORRECT LOW ENERGY LIMIT OF A
ZERO CROSS SECTION.

NORMALIZE TO -Z
=================
NORMALIZE THE RESULTS TO FORCE THEM TO APPROACH THE CORRECT
HIGH ENERGY LIMIT. IN THIS CASE THE REAL SCATTERING IS DEFINED
TO BE,
F'(E)     = -(X*R A(E,1)/R0 + Z)
WHERE,
X         = -Z/R A(INFINITY,0)
SINCE NORMALIZATION IS USED THIS METHOD WILL ALSO APPROACH THE
CORRECT LOW ENERGY LIMIT.

THIS INPUT OPTION MAY BE USED TO ASSESS THE EFFECT THAT USING ONE
OR THE OTHER OF THESE DEFINITIONS MAY HAVE ON THE RESULTS. SINCE
ONLY NORMALIZING RETURNS WILL PRODUCE THE CORRECT LOW AND HIGH
ENERGY LIMITS, THIS IS THE RECOMMENDED OPTION.

THE ANGULAR DEPENDENCE OF THE SCATTERING FACTORS
=====================================================================
THE NORMAL PROCEDURE USED (SEE: REF. 1) IS TO ASSUME THAT THE
SCATTERING FACTORS ARE ISOTROPIC. WITH THIS ASSUMPTION ONCE WE
HAVE CALCULATED THE FORWARD DIRECTION SCATTERING AMPLITUDES AND
IN TURN USED THESE TO DEFINE OUR SCATTERING FACTORS, WE HAVE ALL
THE INFORMATION THAT WE REQUIRE TO CALCULATE ANGULAR DISTRIBUTIONS
AND CROSS SECTIONS. HOWEVER, ASSUMING ISOTROPIC SCATTERING FACTORS
LEADS TO INCONSISTENT RESULTS.

IN ORDER TO HAVE THE ANOMALOUS EFFECT DECREASE AND HAVE THE
CROSS SECTIONS APPROACH THE FORM FACTOR LIMIT AT HIGH ENERGY
THE SCATTERING FACTORS MUST APPROACH ZERO FASTER THAN THE
FORM FACTOR, OTHERWISE THE EFFECT OF COMBINING THEM WILL
ALWAYS HAVE A SIGNIFICANT CONTRIBUTION FROM THE ANOMALOUS
SCATTERING.

WITH THE NORMAL DEFINITION OF ANOMALOUS SCATTERING FACTORS, WHERE
THEY ARE ASSUMED TO BE ISOTROPIC, ONE DOES NOT OBTAIN CONVERGENT
RESULTS AT HIGH ENERGY. INDEED AT HIGH ENERGY THE CONTRIBUTION
OF THE ANOMALOUS SCATTERING FACTORS DOMINATE THAT OF THE FORM
FACTOR, FOR LARGE ANGLE SCATTERING. THIS IS TRUE REGARDLESS OF
WHICH OF THE THREE POSSIBLE DEFINITION DESCRIBED ABOVE IS USED
(CONSERVE, SHIFT, NORMALIZE).

IN ORDER TO OBTAIN CONVERGENT RESULTS THE SCATTERING FACTORS MUST
BE CONSIDERED TO BE ANISOTROPIC. THIS CODE WILL ALLOW THE USER TO
SPECIFY BY INPUT PARAMETERS EITHER ISOTROPIC OR ANISOTROPIC
SCATTERING FACTORS. IF ANISOTROPIC FACTORS ARE SPECIFIED THIS
CODE WILL USE THE ONLY INFORMATION THAT WE HAVE CONCERNING THE
ANGULAR DISTRIBUTION, NAMELY THE FORM FACTOR. THE FORM FACTOR
CAN BE WRITTEN IN THE FORM,

FF(M)  = Z*Q(M)

FF(M)  = FORM FACTOR
Z      = ATOMIC NUMBER OF AN ELEMENT
Q(M)   = THE ANGULAR DEPENDENCE NORMALIZED TO 1.0 AT M = 0.0
M      = MOMENTUM TRANSFER

15

WE WILL ASSUME THAT THE ANGULAR DISTRIBUTION OF THE ANOMALOUS
SCATTERING FACTORS IS,

F(E,M) = F(E,0)*[Q(M)]**POWER

F(E,0) = FORWARD ANGLE REAL OR IMAGINARY SCATTERING FACTOR
F(E,M) = ANGULAR DEPENDENT REAL OR IMAGINARY SCATTERING FACTOR
POWER  = THE POWER OF Q(M) USED

IT HAS BEEN EMPIRICALLY DETERMINED BY EXAMINING THE RESULTS FOR
ALL Z = 1 TO 100 THAT POWER = 1/4 OR MORE WILL YIELD RESULTS
WHICH SMOOTHLY REDUCE TO THOSE USING ONLY FORM FACTORS AT HIGH
ENERGY. IT IS RECOMMENDED THAT THE MINIMUM ANISOTROPY REQUIRED
TO PRODUCE CONVERGENT RESULTS BE USED, I.E., USE POWER = 1/4.

BELOW ABOUT 1 KEV THE FORM FACTOR IS ALMOST ISOTROPIC, SO THAT
INTRODUCING ANISOTROPY INTO THE SCATTERING FACTORS WILL NOT HAVE
ANY EFFECT ON THE ANGULAR DISTRIBUTION AND CROSS SECTION.

AT PROGRESSIVELY HIGHER ENERGIES INTRODUCING MORE ANISOTROPY,
I.E., USING A LARGER POWER, WILL REDUCE THE ANOMALOUS EFFECT AND
CAUSE THE ANGULAR DISTRIBUTION AND CROSS SECTION TO BE CLOSER TO
THE RESULTS OBTAINED USING ONLY FORM FACTORS.

THIS INPUT OPTION MAY BE USED TO ASSESS THE EFFECT OF ANISOTROPY
ON THE RESULTS.

DATA UNITS
================================================================
IN THE FORM IN WHICH THEY WILL BE CALCULATED AND USED,

IM A(W,0) = W*SIG(W)/(4*PI*C)
R  A(W,0) = W*W INTERVALS SIG(W')*DW'/(W'*W' - W*W)/(4*PI*PI*C)

WHERE W IS IN UNITS OF 1/SECONDS AND C IS IN UNITS OF CM/SECONDS,
SO THAT BOTH NORMALIZATION FACTORS ARE IN UNITS OF 1/CM AND THE
RESULTS ARE IN CM. HOWEVER, RATHER THEN USE FREQUENCY, W, AS THE
VARIABLE OF INTEGRATION WE WILL USE ENERGY, E, WHERE,
E = HBAR*W (HBAR = PLANCK'S CONSTANT)
IN WHICH CASE THE NORMALIZATION FACTORS BECOME,
E/(4*PI*H*C)
E/(4*PI*PI*H*C)
C    = 2.99792D+10 CM/SECOND
HBAR = 6.58217D-22 MEV*SECONDS
USING THESE CONSTANTS RESULTS ARE IN CM. HOWEVER SIG(W) AS USED BY
THIS CODE IS GIVEN IN BARNS (1.0D-24 CM*CM) AND WE REQUIRE RESULTS
IN SQRT(BARNS), SO THAT WHEN THE SCATTERING AMPLITUDES ARE SQUARED
THE RESULTS ARE IN BARNS. ALLOWING FOR SIG(W) IN BARNS IN ORDER TO
OBTAIN THE RESULTS IN SQRT(BARNS), INSTEAD OF CM, WE MUST MULTIPLY
THE RESULTS BY 1.0D-12 SQRT(BARNS)/CM. NOTE, C*HBAR WILL YIELD AN
EXPONENT OF 1.0D-12 IN THE DENOMINATOR WHICH WILL EXACTLY CANCEL
THE 1.0D-12 SQRT(BARNS)/CM IN THE NUMERATOR. AS SUCH C AND HBAR
CAN SIMPLY BE TREATED AS EQUAL TO,
C    = 2.99792
HBAR = 6.58217

SIMILARLY,
R0      = 2.82D-13 CM      = 0.282 SQRT(BARNS)
THOMSON = 6.652448D-25 CM*CM = 0.6652448 BARNS

DATA
=============================================================================
THE FOLLOWING DATA ARE USED BY THE PROGRAM,
(1) PHOTOELECTRIC SUBSHELL CROSS SECTIONS FROM JIM SCOFIELD
    TABULATED BETWEEN EITHER 10 EV OR THE EDGE ENERGY, WHICHEVER
    IS HIGHER AND 100 GEV. THIS DATA IS USED TO CALCULATE THE
    SCATTERING AMPLITUDES R A(E,1) AND IM A(E,1). FOR CALCULATION
    OF R A(E,1) USING THE DISPERSION RELATIONSHIP, DEFINED ABOVE,
    IF THE EDGE IS BELOW 10 EV (IN WHICH CASE CROSS SECTIONS ONLY
    EXTEND DOWN TO 10 EV) THE CROSS SECTION MAY BE CONTINUED FROM
    10 EV DOWN TO THE EDGE AS 1/E**3 (BASED ON JIM SCOFIELD'S
    SUBSHELL EDGE DATA) - THIS IS AN INPUT OPTION.
(2) SUBSHELL EDGE ENERGIES FROM JIM SCOFIELD. IF THE EDGE ENERGY
    IS BELOW 10 EV THE EDGE ENERGY IS USED TO DEFINE WHERE TO
    EXTEND THE CROSS SECTION DOWN TO (1/E**3 BETWEEN THE EDGE
    AND 10 EV).
(3) ELECTRONS PER SUBSHELL FROM JIM SCOFIELD. THIS IS USED TO
    CHECK THE EXPECTED HIGH ENERGY LIMIT OF R A(E,1), WHICH
    SHOULD BE EQUAL TO THE NUMBER OF ELECTRONS PER SUBSHELL.
(4) FORM FACTORS FROM JOHN HUBBELL. THIS IS USED TO INTEGRATE
    THE ANGULAR DISTRIBUTIONS TO DEFINE THE COHERENT CROSS
    SECTION.

CROSS SECTIONS AND FORM FACTORS MUST BE IN THE LIVERMORE
EVALUATED NUCLEAR DATA LIBRARY (ENDL) FORMAT.

ALL DATA ARE IN UNITS OF MEV AND BARNS.

DEFINING AN ENERGY GRID
=============================================================================
USING THE ABOVE ALGORITHMS WE CAN CALCULATE THE SCATTERING
AMPLITUDES AND FACTORS AT ANY GIVEN ENERGY. THIS PROGRAM WILL
START FROM THE ENERGY POINTS AT WHICH THE LINEARIZED CROSS
SECTIONS ARE TABULATED, ADD ADDITIONAL POINTS DOWN TO 10 EV IF
NEEDED AND ITERATE IN EACH ENERGY INTERVAL TO PRODUCE TABULATED,
LINEARLY INTERPOLABLE SCATTERING FACTORS.

ENERGY RANGES
==============
BY INPUT THE USER SPECIFIES THE ENERGY RANGE OVER WHICH ANOMALOUS
SCATTERING FACTORS SHOULD BE CALCULATED. FOR ANY GIVEN ELEMENT
THE CALCULATIONS SHOULD BE EXTENDED TO WELL ABOVE ALL PHOTO-
ELECTRIC EDGES. FOR HIGH Z ELEMENTS THE HIGHEST ENERGY EDGE IS
MORE THEN 100 KEV. FOR USE IN OUR EVALUATED PHOTON DATA LIBRARY
(EPDL) WE HAVE ADOPTED THE ENERGY RANGE 10 EV TO 1 MEV AS THE
ENERGY RANGE OF NON-ZERO ANOMALOUS SCATTERING FACTORS AND WE
TREAT THEM AS EXACTLY ZERO ABOVE 1 MEV - THIS IS THE RECOMMENDED
ENERGY RANGE TO SPECIFY AS INPUT.

IN ADDITION TO THE ENERGY OF THE SCATTERING FACTORS THE OTHER
ENERGY RANGE OF INTEREST IS THE ENERGY RANGE OF THE COHERENT
CROSS SECTION. BY AN INPUT OPTION THE USER MAY SPECIFY EITHER
THAT THIS BE THE SAME ENERGY RANGE AS THE ANOMALOUS SCATTERING
FACTORS OR THAT THE CALCULATION OF CROSS SECTION BE EXTENDED
UP TO THE UPPER LIMIT OF THE PHOTOELECTRIC CROSS SECTION - E.G.,
FOR EPDL CROSS SECTIONS, UP TO 100 GEV. WHEN ANOMALOUS DATA,
RATHER THAN PHOTOELECTRIC CROSS SECTIONS, ARE USED AS THE STARTING
POINT BY THIS CODE (SEE, THE DESCRIPTION OF OPERATING MODES ABOVE)
IF EXTENSION IS INDICATED BY THE INPUT OPTION, THE CROSS SECTIONS
WILL BE EXTENDED UP TO 100 GEV, REGARDLESS OF THE ENERGY RANGE OF
THE SCATTERING FACTORS.

## ALLOWABLE UNCERTAINTY

INPUT OPTIONS ARE USED TO SPECIFY ALLOWABLE UNCERTAINTY IN ORDER
TO LINEARIZE CROSS SECTIONS, CALCULATE SCATTERING FACTORS, ETC.
BY ALLOWABLE UNCERTAINTY WE MEAN THAT AT THE TABULATED VALUES
RESULTS WILL BE MORE OR LESS EXACT. BETWEEN TABULATED VALUES
LINEAR INTERPOLATION MAY BE USED TO DEFINE RESULTS TO WITHIN THE
ALLOWABLE UNCERTAINTY.

AN INPUT OPTION IS USED TO DEFINE HOW ACCURATELY TO LINEARIZE
CROSS SECTIONS AND TO CALCULATE RESULTS. THE RESULTS ARE FAIRLY
INSENSITIVE TO THE ACCURACY TO WHICH CROSS SECTIONS ARE LINEARIZED
AND RESULTS SHOULD NOT BE CALCULATED TO MUCH HIGHER PRECISION THEN
WE KNOW OUR CROSS SECTIONS. THEREFORE IT IS RECOMMENDED THAT A
PRECISION IN THE RANGE 0.1 TO 1.0 PER-CENT BE USED FOR THIS INPUT
OPTION - THIS IS SIGNIFICANTLY LESS THEN THE UNCERTAINTY OF THE
CROSS SECTIONS AND AS SUCH WILL NOT INTRODUCE ANY ADDITIONAL
ERROR.

## HIGH ENERGY ASYMPTOTIC FORM

BOTH THE REAL AND IMAGINARY SCATTERING FACTORS APPROACH ZERO AT
HIGH ENERGY. THE USER OF THIS CODE HAS THE OPTION TO CUTOFF THE
RESULTS AT AN ENERGY WHERE THE RESULTS BECOME SMALL. IN ORDER TO
DEFINE SMALL THIS PROGRAM WILL USE THE CHARACTERISTIC SHAPE OF
THE REAL SCATTERING FACTOR. NEAR AN EDGE THE REAL SCATTERING
WILL HAVE A MAXIMUM. AT HIGHER ENERGIES IT WILL DECREASE, BECOME
NEGATIVE, PASS THROUGH A MINIMUM AND THEN BEGIN TO ASSYMPTOTICALLY
APPROACH ZERO. IF INPUT PARAMETERS INDICATE THE CALCULATION SHOULD
BE CUTOFF AT ENERGIES WELL ABOVE EDGES, THE PROGRAM WILL DO 2
THINGS,

(1) CHECK FOR R A(E,1) PASSING THROUGH A NEGATIVE MINIMUM VALUE
     ABOVE THE EDGE. AFTER R A(E,1) HAS PASSED THROUGH A MINIMUM
     CALCULATIONS WILL CONTINUE UP TO 10 TIMES THIS ENERGY AND THEN
     THE PROGRAM WILL TEST FOR APPROACH TO ASYMPTOTIC FORM, STEP
     (2), DESCRIBED BELOW.

(2) CHECK FOR APPROACH TO THE ASYMPTOTIC FORM,
     R A(E,1) = R A(INFINITY,0)*[1 + Z**2 + Z**4/2 + Z**6/3...
     WHERE Z = (EDGE ENERGY)/E
     THE COMPUTED VALUE OF R A(E,1) WILL BE COMPARED TO THIS
     ASYMPTOTIC FORM AND WHEN IT IS WITHIN THE ALLOWABLE
     UNCERTAINTY (AN INPUT PARAMETER) THE ASYMPTOTIC FORM WILL
     BE USED TO DEFINE R A(E,1) AT ALL HIGHER ENERGIES. FOR
     SUBSHELLS WITH LOW EDGE ENERGIES (HYDROGEN ABOUT 13.6 EV)
     THIS METHOD WILL GREATLY REDUCE RUNNING TIME WITHOUT
     INTRODUCING ANY APPRECIABLE ERROR, E.G, WITHOUT USING
     THIS METHOD IF WE WISH TO OBTAIN RESULTS FOR 10 EV UP TO
     1 MEV, BY 1 MEV WE WILL BE AT ALMOST 100,000 THE EDGE
     ENERGY AND THE RESULTS WILL TO ESSENTIALLY EQUAL TO ZERO
     FROM MUCH LOWER ENERGIES UP TO 1 MEV.

IT IS RECOMMENDED THAT THE CALCULATION BE CUTOFF WHEN THE APPROACH
TO THE ASYMPTOTIC FORM IS ABOUT 1/10 IS THE ALLOWABLE UNCERTAINTY
IN THE CROSS SECTION, E.G., IF YOU SPECIFY AN ALLOWABLE ERROR OF
1 PER-CENT FOR THE CROSS SECTION, SPECIFY 0.1 PER-CENT AS THE
CUTOFF PARAMETER - USING A SMALLER CUTOFF ONLY CAUSE THE PROGRAM
TO RUN ALONG AND WILL NOT SIGNIFICANTLY IMPROVE THE ACCURACY OF
THE RESULTS - USING A MUCH LARGER CUTOFF CAN LEAD TO INACCURATE
RESULTS AT HIGH ENERGY.

CROSS SECTION USED TO CALCULATE ANOMALOUS SCATTERING FACTORS
=================================================================
NORMALLY THE PHOTOELECTRIC CROSS SECTION WILL BE USED TO CALCULATE
ANOMALOUS SCATTERING FACTORS. HOWEVER, THIS CODE HAS BEEN DESIGNED
TO ALLOW ANY CROSS SECTION TO BE USED TO CALCULATE THE ANOMALOUS
SCATTERING FACTORS. THIS OPTION IS INTENDED ONLY BE ALLOW THE
USER TO DETERMINE THE EFFECT OF USING ANOTHER CROSS SECTION, E.G.,
INCOHERENT, PAIR PRODUCTION, ETC. IN THE BELOW INPUT COLUMNS
11-20 OF THE FIRST INPUT LINE REFER TO THE CROSS SECTION WHICH
SHOULD BE USED TO CALCULATE THE ANOMALOUS SCATTERING FACTORS.
TO USE PHOTOELECTRIC CROSS SECTIONS THE MINIMUM AND MAXIMUM C
(ENDL REACTION DESIGNATOR) SHOULD BE 73.

INPUT PARAMETERS
=================================================================
3 LINES OF INPUT ARE REQUIRED FOR EACH CALCULATION. THE 3 LINES
MAY BE REPEATED ANY NUMBER OF TIMES. REQUESTED Z RANGES MAY BE
IN ANY ORDER, BUT PROGRAM EXECUTION TIME MAY BE MINIMIZED BY
SPECIFYING REQUESTED Z IN ASCENDING ORDER (E.G., IF NOT THE
PROGRAM WILL HAVE TO REWIND THE DATA FILES AND SEARCH FROM THE
BEGINNING).

IF ANY INPUT FIELD IS ZERO IN WILL BE ASSIGNED THE DEFAULT VALUE
INDICATED BELOW.

| LINE | COLUMNS | FORMAT | DEFAULT | DEFINITION |
|------|---------|--------|---------|------------|
| 1 | 1- 5 | I5 | 1 | MINIMUM Z |
| | 6-10 | I5 | 100 | MAXIMUM Z |
| | 11-15 | I5 | 73 | MINIMUM C   (CROSS SECTION USED TO |
| | 16-20 | I5 | 73 | MAXIMUM C    CALCULATE ANOMALOUS) |
| | | | | C = 70 - TOTAL |
| | | | | = 71 - COHERENT |
| | | | | = 72 - INCOHERENT |
| | | | | = 73 - PHOTOELECTRIC |
| | | | | = 74 - PAIR PRODUCTION |
| | | | | = 75 - TRIPLET PRODUCTION |
| | | | | = 93 - ANOMALOUS SCATTERING FACTORS |
| | 21-25 | I5 | 1 | MINIMUM SUBSHELL DESIGNATOR |
| | 26-30 | I5 | 100 | MAXIMUM SUBSHELL DESIGNATOR SEE, SUBROUTINE SUBHOL FOR A DEFINITION OF ALL SUBSHELL DESIGNATORS. |
| | 31-35 | I5 | 0 | USE SCOFIELD EDGE ENERGIES TO DEFINE SUBSHELL EDGE ENERGY OR LOWEST EDGE ENERGY FOR TOTAL PHOTOELECTRIC = 0 - NO, USE DATA AS READ = 1 - YES, MOVE SUBSHELL EDGE TO SCOFIELD ENERGY OR EXTEND TOTAL PHOTOELECTRIC DOWN TO LOWEST EDGE. |
| | 36-40 | I5 | 0 | CONSERVE, NORMALIZE OR SHIFT = 0 - CONSERVE R A(E,1) = 1 - NORMALIZE TO -Z = 2 - SHIFT TO -Z |
| | 41-45 | I5 | 0 | EXTEND CROSS SECTION CALCULATION TO UPPER ENERGY LIMIT OF TABULATED PHOTOELECTRIC CROSS SECTIONS = 0 - NO = 1 - YES |

19

```
2      1-10    E10.4    10 EV      MINIMUM ENERGY
      11-20    E10.4    1 MEV      MAXIMUM ENERGY
      21-30    E10.4    0.001      EDGE WIDTH
      31-40    E10.4    0.01       ALLOWABLE UNCERTAINTY
                                   (TO LINEARIZE CROSS SECTION AND
                                    PRODUCE LINEARLY INTERPOLABLE
                                    TABULATED RESULTS)
      41-50    E10.4    0.0        HIGH ENERGY CUTOFF CRITERIA
      51-60    E10.4    0.0        ANISOTROPY OF SCATTERING FACTORS
                                   (POWER OF THE NORMALIZED FORM
                                    FACTOR)
```

THE THIRD INPUT LINE DEFINES WHAT TO CALCULATE AND OUTPUT

```
3      1- 5    I5       0          REPORT ON LINE OR TO FILE
                                   = 0 - ON LINE
                                   = 1 - TO FILE
       6-10    I5       0          SUBSHELL CROSS SECTIONS
      11-15    I5       0          F'' SUBSHELL
      16-20    I5       0          REAL SUBSHELL AMPLITUDE
      21-25    I5       0          F' SUBSHELL
      26-30    I5       0          (NO LONGER USED)
      31-35    I5       0          F'' TOTAL (FOR EACH Z)
      36-40    I5       0          REAL TOTAL AMPLITUDE
      41-45    I5       0          F' TOTAL (FOR EACH Z)
      46-50    I5       0          COHERENT CROSS SECTION
      51-55    I5       0          COHERENT (FORM FACTOR = Z)
      56-60    I5       0          COHERENT (NO ANOMALOUS)
      61-65    I5       0          FORM FACTOR MAX/MIN LIMITS
      66-70    I5       0          F' HIGH ENERGY LIMIT VS. Z
      71-75    I5       0          PRINT EACH ENERGY POINT

                                   FOR OUTPUT,
                                   0 = NONE
                                   1 = ENDL FORMAT
                                   2 = PLOTTAB FORMAT
                                   3 = ENDL AND PLOTTAB FORMAT
```

IF YOU DO NOT SPECIFY ANY OUTPUT THE PROGRAM WILL NOT PERFORM ANY
CALCULATIONS AND WILL PROCEED TO READ THE NEXT 3 INPUT LINES, FOR
THE NEXT REQUESTED CALCULATION.

```
EXAMPLE INPUT
=============
CALCULATE AND OUTPUT REAL AND IMAGINARY SCATTERING FACTORS
FROM EACH Z = 1 TO 100 FROM PHOTOELECTRIC SUBSHELL CROSS SECTIONS.
IF THE EDGE OF ANY SUBSHELL IS BELOW 10 EV USE SCOFIELD'S EDGES
TO EXTEND THE CROSS SECTIONS BELOW 10 EV DOWN TO THE EDGE.
NORMALIZE RESULTS TO -Z TO OBTAIN THE CORRECT LOW AND HIGH ENERGY
LIMITS. RESULTS WILL BE CALCULATED FOR THE ENERGY RANGE 10 EV UP
TO 1 MEV. 1 PER-CENT ENERGY RESOLUTION WILL BE USED. CROSS
SECTIONS WILL BE LINEARIZED AND TABULATED RESULTS PRODUCED TO
WITHIN 1 PER-CENT. THE APPROACH TO THE HIGH ENERGY ASYMPTOTIC
FORM WILL START AFTER F'(E) IS LESS THEN 0.1 PER-CENT OF ITS
MAXIMUM NEGATIVE VALUE. OUTPUT ONLY F'(E) AND F''(E) FOR EACH Z.
THE OUTPUT REPORT WILL BE WRITTEN TO A FILE (INSTEAD OF ON LINE).
THE FOLLOWING 3 INPUT LINES ARE REQUIRED,

  1  100    73    73    1  100     1     1     0
1.0000-05 1.0000+00 1.0000-02 1.0000-02 1.0000-03 0.0000+00
  1    0    0     0     0     0     1     0     1     0     0     0      0


NOTE - SINCE COHERENT CROSS SECTIONS WILL NOT BE CALCULATED IT
IS NOT NECESSARY TO SPECIFY ANISOTROPY - ON THE ABOVE LINES FOR
COMPLETENESS OF ALL INPUT FIELDS, THIS FIELD HAS BEEN DEFINED TO
BE ZERO.

THESE 3 LINES MAY BE FOLLOWED BY ADDITIONAL SETS OF 3 LINES
SPECIFYING THE NEXT CALCULATION. CALCULATIONS WILL CONTINUE UNTIL
ALL SETS OF INPUT REQUESTS HAVE BEEN PROCESSED (I.E., UNTIL AN
END OF FILE IS READ ON THE INPUT PARAMETER FILE).

I/O FILE DEFINITIONS
================================================================
UNIT     FILENAME     DEFINITION
====     ========     ==========
   5     INSCAT       INPUT PARAMETERS
   6     REPORT       OUTPUT REPORT
  10     PHOTO        PHOTON CROSS SECTIONS (ENDL FORMAT)
  11     PLOTTAB      PLOTTAB FORMATTED OUTPUT
  14     ATOMIN       SUBSHELL PARAMETERS
  15     FFACTORS     FORM FACTORS (ENDL FORMAT)
  16     ENDL         ENDL FORMATTED OUTPUT
```

***** COMPUTER DEPENDENT CODING ****************************************

THIS PROGRAM HAS BEEN DESIGNED TO WORK ON VIRTUALLY ANY COMPUTER.
IN ORDER TO ACCOMPLISH THIS THE PROGRAM CONTAINED REDUNTANT
CODING WHICH CAN BE AUOTMATICALLY TURNED ON OR OFF IN ORDER TO
IMPLEMENT THIS CODE ON ANY GIVEN COMPUTER. THE COMPUTER DEPENDENT
CODING IN THIS PROGRAM INCLUDES,

(1) CHARACTER/INTEGER
    TREAT CHARACTER ARRAYS EITHER AS CHARACTER (FORTRAN-77
    CONVENTION) OR INTEGER (FORTRAN-H CONVENTION).

(2) DOUBLE/SINGLE
    TREAT ENERGY DEPENDENT VARIABLES IN SINGLE OR DOUBLE
    PRECISION. ON LONG WORD LENGTH COMPUTERS (E.G., CRAY)
    THIS PROGRAM CAN BE OPTIMIZED BY USING SINGLE PRECISION.
    ON SHORT WORD LENGTH COMPUTERS (E.G., IBM) DOUBLE
    PRECISION MUST BE USED.

(3) NAMES
    OPTION TO USE STANDARD FILE NAMES

(4) CIVIC
    LIVERMORE CIVIC COMPILER CONVENTIONS

(5) STOP/EXIT
    USE EITHER STOP OR CALL EXIT TO TERMINATE EXECUTION

(6) SAVE
    FORTRAN OPTION TO SAVE ALL VARIABLES BETWEEN SUBROUTINE
    CALLS - REQUIRED WITH SOME NEWER FORTRAN COMPILERS.

PROGRAM CONVERT MAY BE USED TO AUTOMATICALLY CONVERT THIS PROGRAM
BACK AND FORTH USING ANY COMBINATION OF THESE COMPUTER DEPENDENT
OPTIONS TO BEST MEET YOUR NEEDS (FOR DETAILS OF PROGRAM CONVERT
CONTACT D. E. CULLEN AT THE ABOVE ADDRESS).

***** COMPUTER DEPENDENT CODING ****************************************

## Example Input Parameters and Output Report

After reading the 3 lines of input parameters the code will extensively check the validity of all parameters, define standard input options where required and then print a summary of its interpretation of all input parameters. This summary will include an interpretation of all input parameters exactly in the order in which they are read. For example, the following 3 lines of input parameters (exactly the example shown in the above documentation), will produce the following summary report,

```
      1  100     73    73     1  100     1     1     0
   1.0000-05 1.0000+00 1.0000-02 1.0000-02 1.0000-03 0.0000+00
      1     0     0     0     0     0     1     0     1     0     0     0     0
```

COHERENT SCATTERING FACTORS AND CROSS SECTIONS (SCATMAN: VERSION 90-1)

============================================================================

PARAMETERS

============================================================================

```
Z RANGE...............................         1 TO            100
REACTION NUMBER RANGE.................        73 TO             73
SUBSHELL DESIGNATOR RANGE............         1 TO            100
LOWEST EDGE ENERGY..................LOWEST SCOFIELD EDGE
NORMALIZATION.......................RE-NORMALIZE TO -Z LIMIT
EXTEND CROSS SECTION CALCULATION....NO
ENERGY RANGE........................ 1.00000- 5 TO  1.00000+ 0 MEV
WIDTH OF EDGES...................... 1.00000- 2 TIMES ENERGY
LINEAR CROSS SECTIONS TO WITHIN.....      1.0000 %
HIGH ENERGY CUTOFF CRITERIA.........      0.1000 %
ANISOTROPY (POWER OF FORM FACTOR)... 0.00000+ 0
```

============================================================================

OUTPUT

============================================================================

```
OUTPUT..............................TO FILE
SUBSHELL CROSS SECTION.............NO
F2 SUBSHELL (IMAGINARY)...........NO
REAL SUBSHELL AMPLITUDE/R0.........NO
F1 SUBSHELL (REAL)................NO
TOTAL PHOTOELECTRIC CROSS SECTION..NO
F2 TOTAL (IMAGINARY)...............YES
REAL TOTAL AMPLITUDE/R0............NO
F1 TOTAL (REAL)....................YES
COHERENT..........................NO
COHERENT (FORM FACTOR = Z)........NO
COHERENT (NO ANOMALOUS)...........NO
LIMITS OF FORM FACTOR.............NO
HIGH ENERGY LIMIT VS. Z...........NO
PRINT EACH ENERGY POINT...........NO
```

============================================================================

Following the summary of the input parameters the code will produce a one line summary of each section of ENDL formatted data as it has read. This one line summary will include,

(1)  Identification of the element, e.g., 92-U.

(2)  Identification of the type of data read, e.g., total photoelectric or an identification for a specific subshell.

(3)  Lower edge energy - this will be either the lowest edge energy for an element (if the total photoelectric is used) or the edge energy for a subshell, if cross section extension is specified by input, or the first tabulated energy of the cross section, if cross section extension is not specified by input. The two can differ only if the actual lowest edge is less than 10 eV.

(4)  Electrons - the atomic number of an element (if the total photoelectric is used) or the number of electrons in the subshell (if subshell data is used).

(5)  High E limit - the normalized integral of the cross section up to maximum energy up to which anomalous scattering factors are to be calculated - if the shift to -Z or normalize to -Z input option is specified by input, this is the value which will be used to force the real scattering factor to become zero at the maximum energy of the scattering factors.

(6)  Infinity limit - the integral of the entire cross section - if this differs significantly from the high E limit it is an indication that the real amplitude has not reached its asymptotic value by the maximum energy to which scattering factors were calculated. Note, since the dispersion integral does not vary monotonically with increasing energy one cannot assume that the infinity limit is in all cases greater than the high energy limit.

(7)  Points - the number of energy points which have been output by the code - the number of points will be the same for the real and imaginary scattering factors.

The following illustrates a portion of the output report corresonding to the above input parameters.

| ELEMENT | SUBSHELL OR REACTION | LOWEST E EDGE (MEV) | ELECTRONS | HIGH E LIMIT | INFINITY LIMIT | POINTS |
|---|---|---|---|---|---|---|
| 1-H | K (1s1/2) | 1.36100- 5 | 1.0000 | 0.4351 | 0.4351 | 253 |
| | | | 1.0000 | 0.4351 | 0.4351 | 254 |
| 2-He | K (1s1/2) | 2.34200- 5 | 2.0000 | 1.5084 | 1.5084 | 244 |
| | | | 2.0000 | 1.5084 | 1.5084 | 245 |
| 3-Li | K (1s1/2) | 5.98500- 5 | 2.0000 | 1.8396 | 1.8396 | 237 |
| 3-Li | L1 (2s1/2) | 5.50000- 6 | 1.0000 | 0.3013 | 0.3013 | 144 |
| | | | 3.0000 | 2.1409 | 2.1409 | 186 |
| 4-Be | K (1s1/2) | 1.18360- 4 | 2.0000 | 1.8009 | 1.8009 | 235 |
| 4-Be | L1 (2s1/2) | 8.18000- 6 | 2.0000 | 0.6503 | 0.6503 | 151 |
| | | | 4.0000 | 2.4513 | 2.4513 | 182 |

## Example Results

The following figures illustrate the results for four representiative elements. The results for all elements, Z = 1 through 100, are presented in "Tables and Graphs of Photon Interaction Cross Sections from 10 eV to 100 GeV derived from the LLNL Evaluated Photon Data Library (EPDL)," UCRL-40500, Vol. 6, Rev. 4 (1989), by. D.E. Cullen, M.H.Chen, J.H. Hubbell, S.T. Perkins, E.F. Plechaty, J.A. Rathkopf and J.H. Scofield.

Fig. 1 presents the real and imaginary scattering factors. The imaginary scattering factor is merely proportional to energy times the photoelectric cross section modified to include the effect of bound pair production, so that it smoothly approaches zero at 10 MeV. This figure illustrates the general characteristics of the real scattering factor which asymptotically approaches -Z at low energy, cancelling the contribution of the form factor and leading to a coherent cross section which approaches zero at low energy. At higher energies the real scattering factor decreases in magnitude and approaches zero at high energy. Near photoelectric edges it increases in magnitude tending to cancel a portion of the contribution of the form factor and leading to a minimum in the coherent cross section.

Fig. 2 illustrates the coherent cross section with and without the effect of anomalous scattering. Without anomalous scattering at low energy the cross section approaches a constant proportional to Z squared. In contrast, with anomalous scattering at low energy

the cross section approaches zero. Below the highest energy photoelectric edge as energy decreases the results with anomalous scattering are progressively smaller then those without anomalous scattering. The actual monotonic decrease toward zero at low energy will only be seen below the lowest energy edge. This effect can be seen in the case of 8-O, since all of its edges lie above 10 eV. In the other three cases we can see the general decrease in the cross section, but since all have edges below 10 eV the cross section has not yet reached the point of monotonic variation toward zero. Near photoelectric edges the anomalous effect produces minima in the cross section. At progressively higher energies, well above all edges, the two sets of results smoothly converge. Since the highest energy of the edges increases with the atomic number of the element, Z, we can see that for progressively higher Z elements the effect of anomalous scattering extends to higher and higher energies.

The results shown on this figure are the most important results presented here, since they clearly illustrate the importance of including the anomalous effect in order to calculate the correct coherent scattering. For example, in the case of 92-U we can see that at low energy, near 10 eV, the cross section including anomalous scattering is of the order of a factor of 1,000 smaller then the results obtained only using form factors. Similarly we can see that even at higher energies where the two results are converging together, near photoelectric edges the cross section including anomalous scattering is roughly half that predicted using only form factors; since we expect fluorescence x-rays to be emitted just below the photoelectric edges the combination of a minimum in the photoelectric and coherent cross sections can make these x-rays quite penetrating.

In Fig. 3 and 4 angular distributions are shown at fixed scattering angles versus energy, with and without anomalous scattering. The angular distribution may be written in the form,

$$\frac{d\sigma}{d\mu}(E,\mu) = \left(1 + \mu^2\right)G(E,\mu)$$

(1)

where G(E,m) is the perpendicular-polarized coherent scatter. On these two figures we present G(E,m) rather than the entire angular distribution in order to more clearly illustrate the behavior of the angular distribution. From this figure we can see that at low energy, below about 1 keV, G(E,m) becomes isotropic; the angular distribution becomes $1+\mu^2$. With increasing energy the angular distributions become progressively more forward peaked. Near and below photoelectric edges inclusion of anomalous scattering further suppresses the large angle scattering and the angular distribution becomes even more forward peaked. In contrast the results without anomalous scattering, based solely on form factors, show no effect of photoelectric edges. These results were calculated using Program COHERT, since SCATMAN does not calculate and output angular distributions.

Fig. 5 illustrates the results obtained based on the three possible models used to derive the real scattering factor from the real scattering amplitude, namely,

1) Conserve the real amplitude
2) Normalize to obtain the required -Z limit
3) Shift to obtain the required -Z limit

From this figure we can clearly see that when we conserve the real amplitude we do not obtain the expected high energy cross section. In all cases we over estimate the cross section with the difference increasing for higher Z elements. Similarly we can see that when we shift to obtain the required -Z limit we do not obtain the expected low energy cross

section. For 8-O the cross section actually becomes equal to zero and then rises below the lowest energy edge. For the other three elements the shifted results diverge from the other results at low energy and become an order of magnitude or more larger than the other results. From the normalized results we can see that these are somewhat of a compromise between the other two results, which at high energy approach the shifted results and at low energy approach the conserve results, thereby reproducing both the expected low and high energy limits.

The above results are based on scattering factors which include anisotropy (1/4 power of the normalized form factor). Fig. 6 illustrates the effect of anisotropy. From this figure we can see that at low energy, where the form factor is essentially isotropic, anisotropy has no effect. With increasing energy, as the form factor becomes progressively more anisotropic, the results with anisotopy always lie closer to the results using only form factors, then do the isotropic results. From this figure it is not obvious that anistropy is required in order to obtain convergence toward the form factor limit at high energy.

In Fig. 7 we present the first moment of the angular distribution,

$$\langle 1-\mu \rangle = \frac{\int (1-\mu)\dfrac{d\sigma}{d\mu}(E,\mu)d\mu}{\int \dfrac{d\sigma}{d\mu}(E,\mu)d\mu}$$

(2)

At low energy the angular distribution becomes proportional to Thomson scattering, $1+\mu^2$, so that in all case $<1-\mu>$ approaches 1.0. At progressively higher energies the angular distribution becomes more and more forward peaked and $<1-\mu>$ decreases. In this case we can see that even when we consider only the first moment of the angular distribution we do not obtain convergent results at high energy. This effect becomes more and more obvious for higher Z elements. In the case of 92-U we can see that by 1 MeV the results are diverging and only eventually converge again by 10 MeV because we have forced the cross section to approach zero at 10 MeV. This divergence is because without anisotropy the anomalous contribution to the angular distribution dominates the form factor contribution for large angle scattering and rather then decreasing and smoothly disappearing at high energy, the effect is actually becoming stronger.

8-O Real and Imaginary
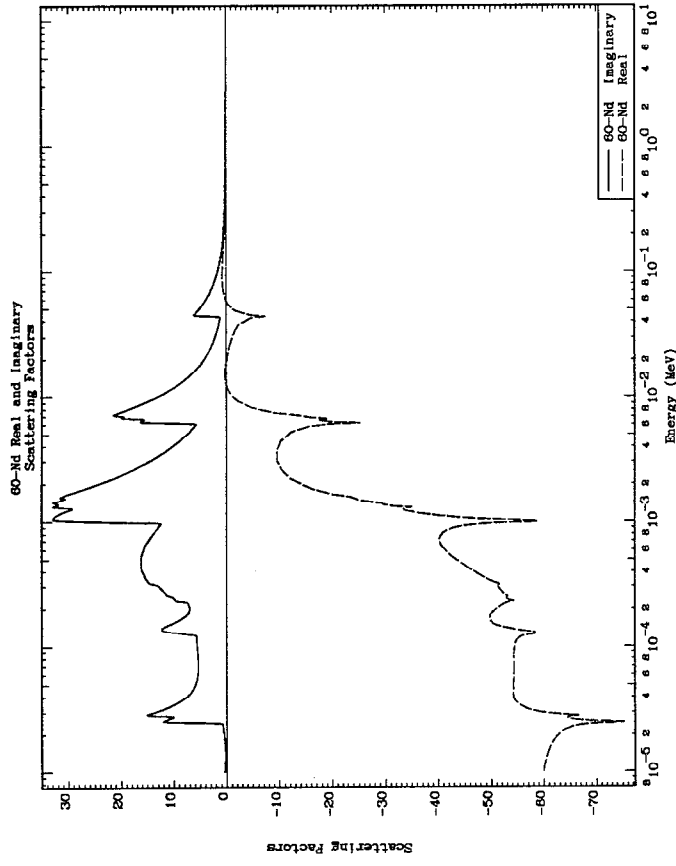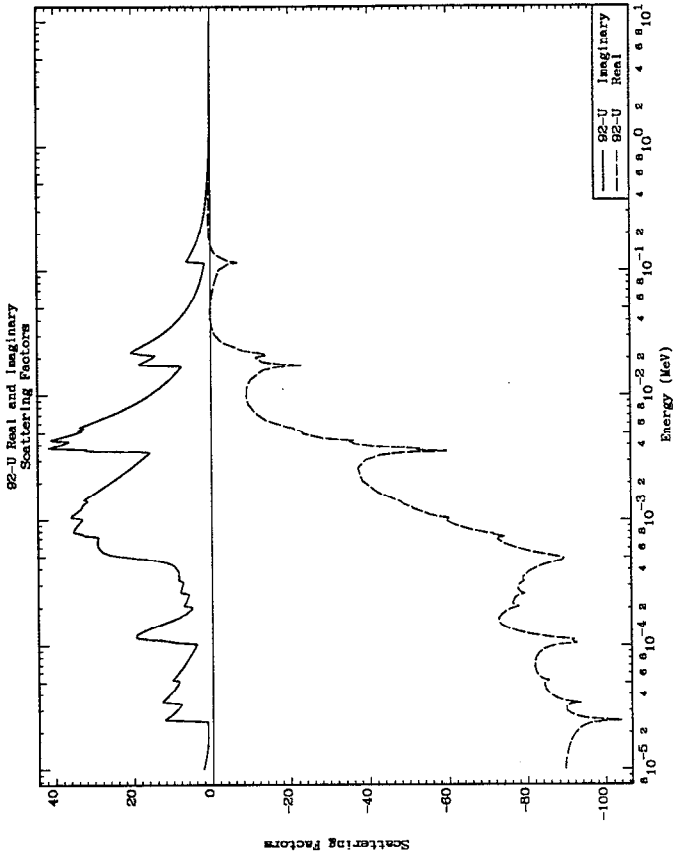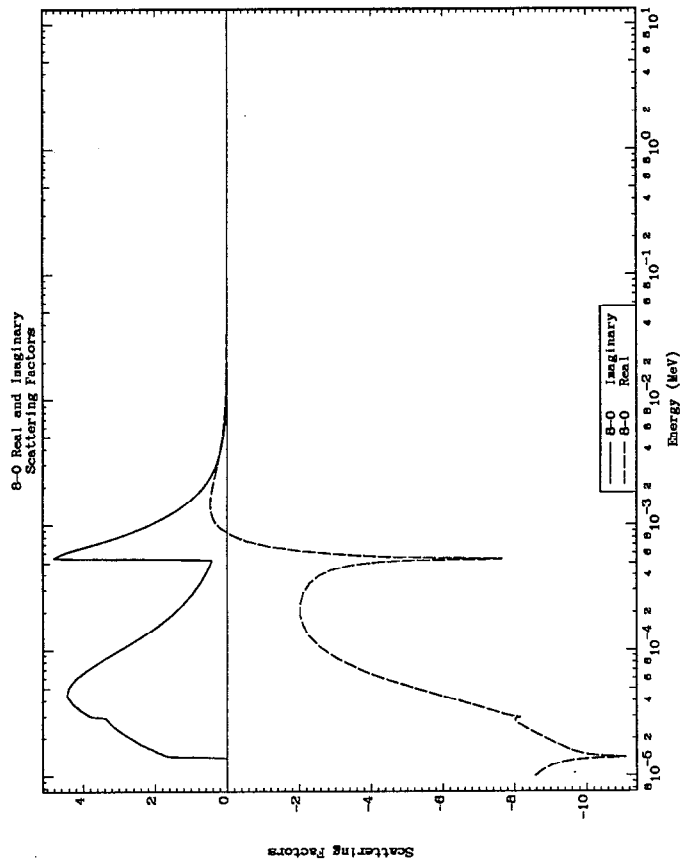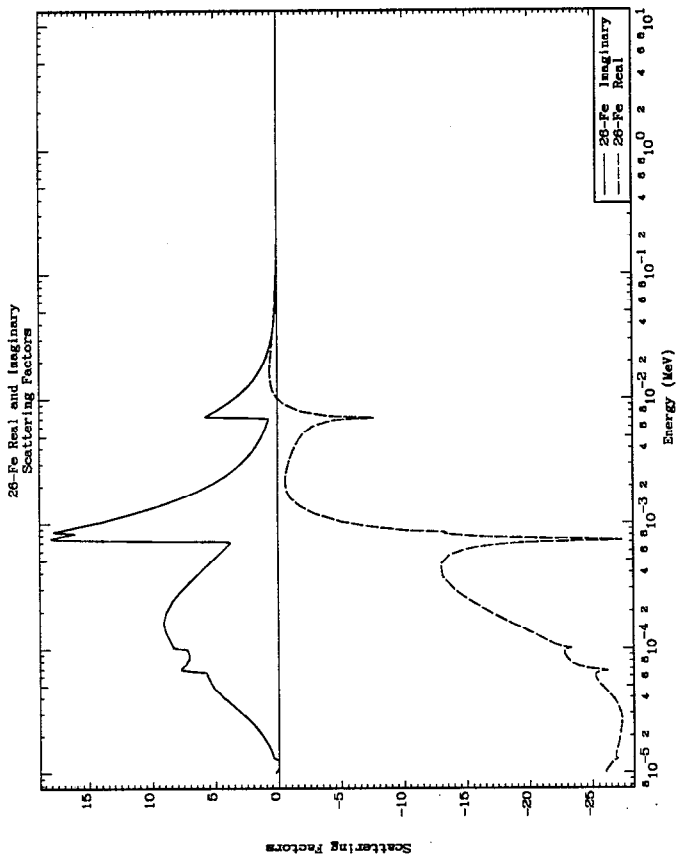Scattering Factors

26-Fe Real and Imaginary
Scattering Factors

60-Nd Real and Imaginary
Scattering Factors

92-U Real and Imaginary
Scattering Factors

Figure 1

28

Figure 2

29

Figure 3

30

Figure 4

31

Figure 5

32

Figure 6

33

Figure 7

34
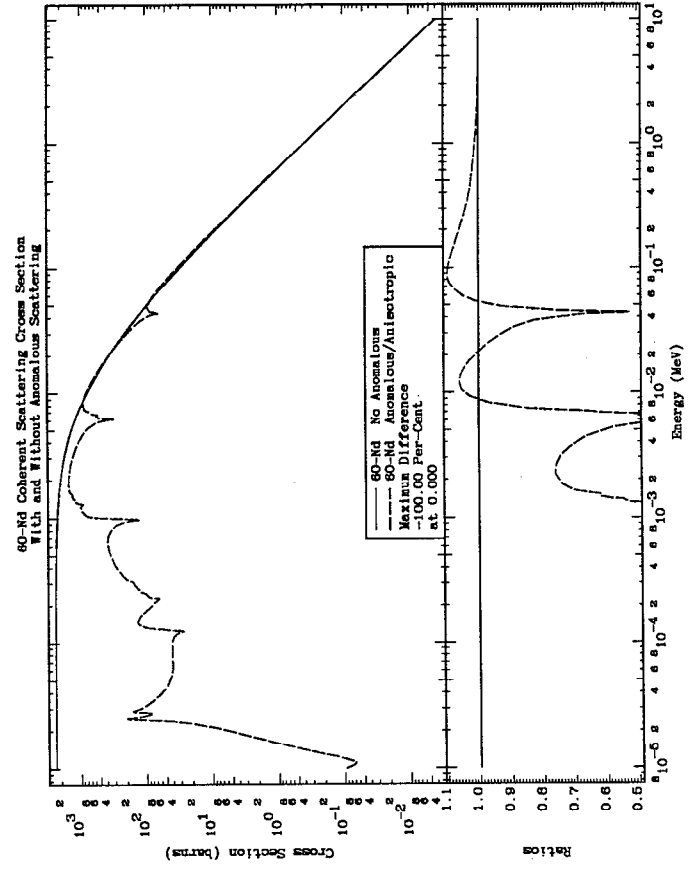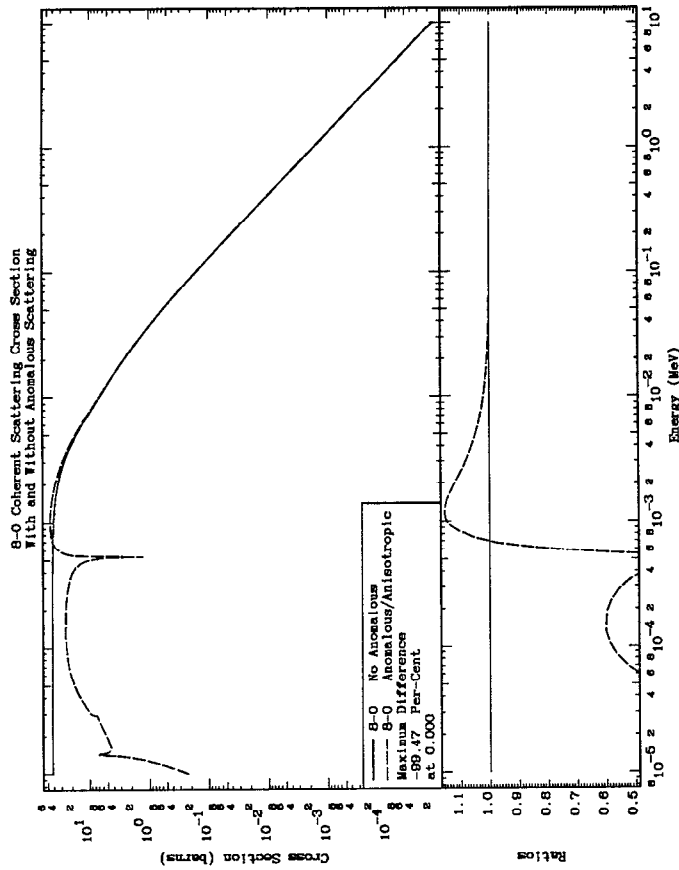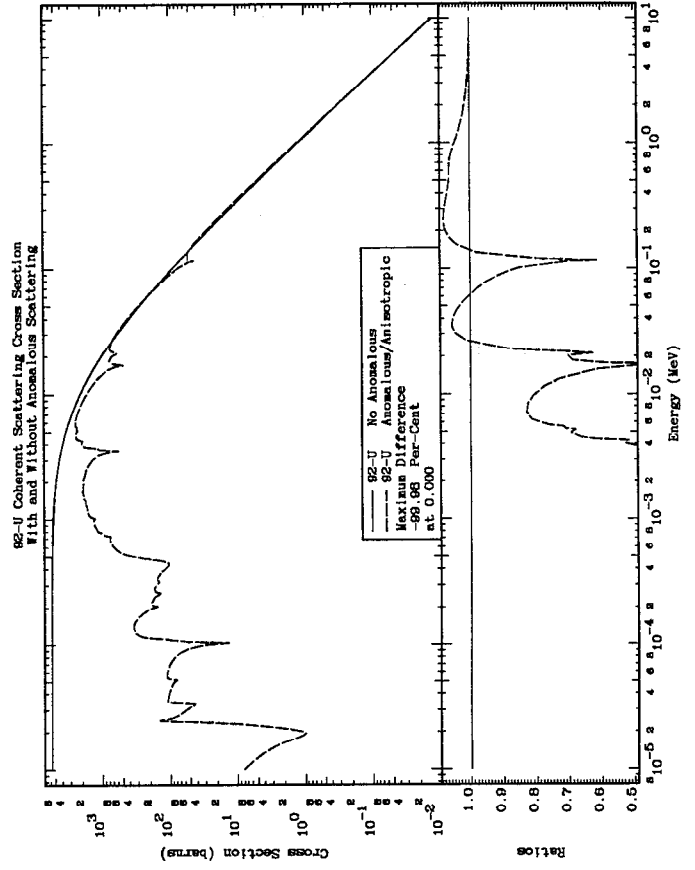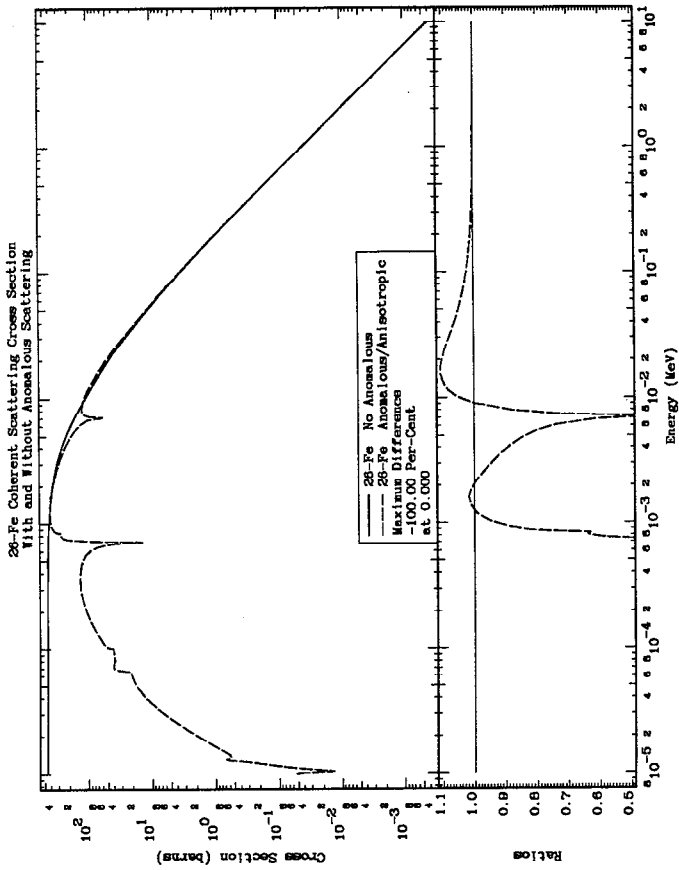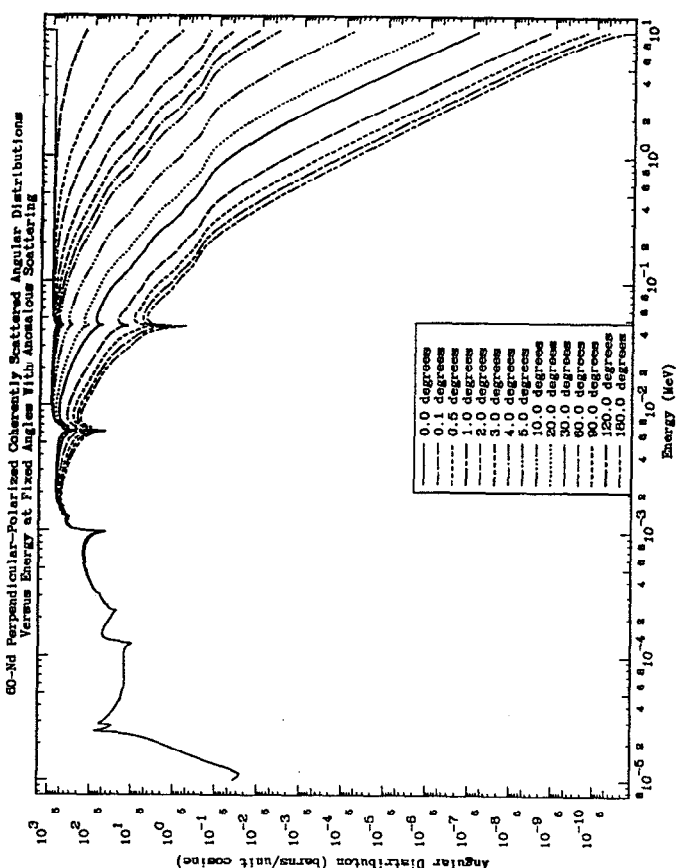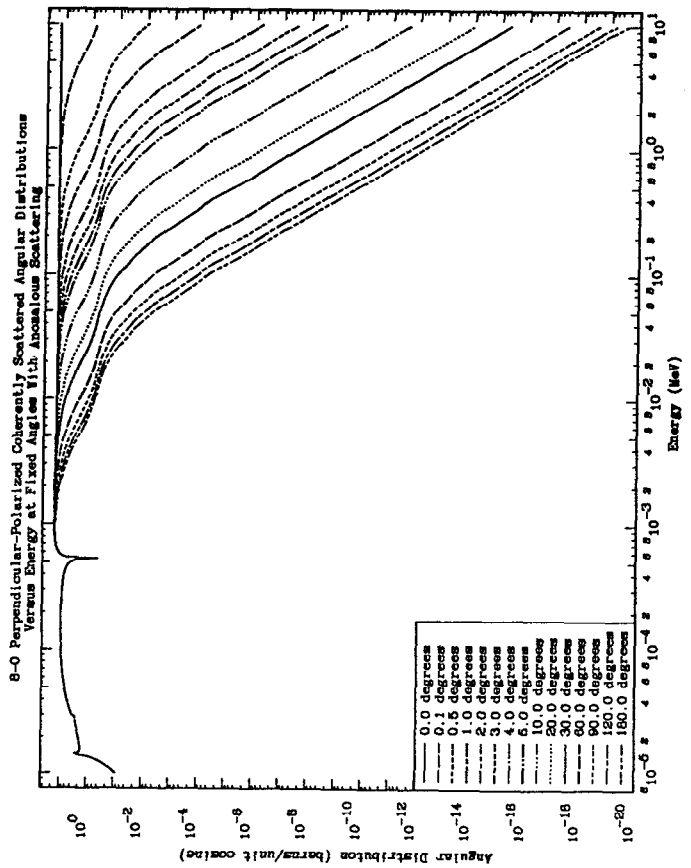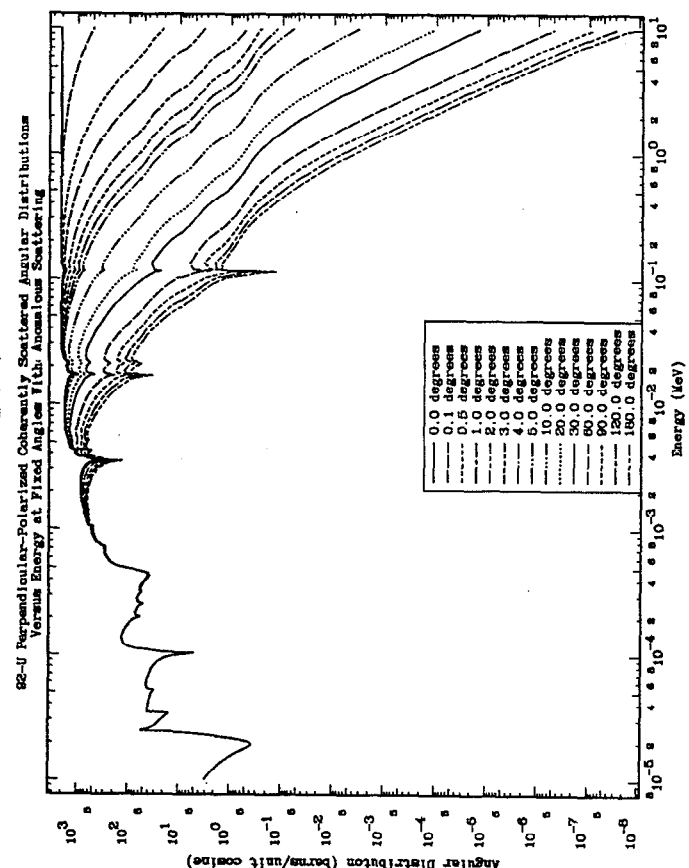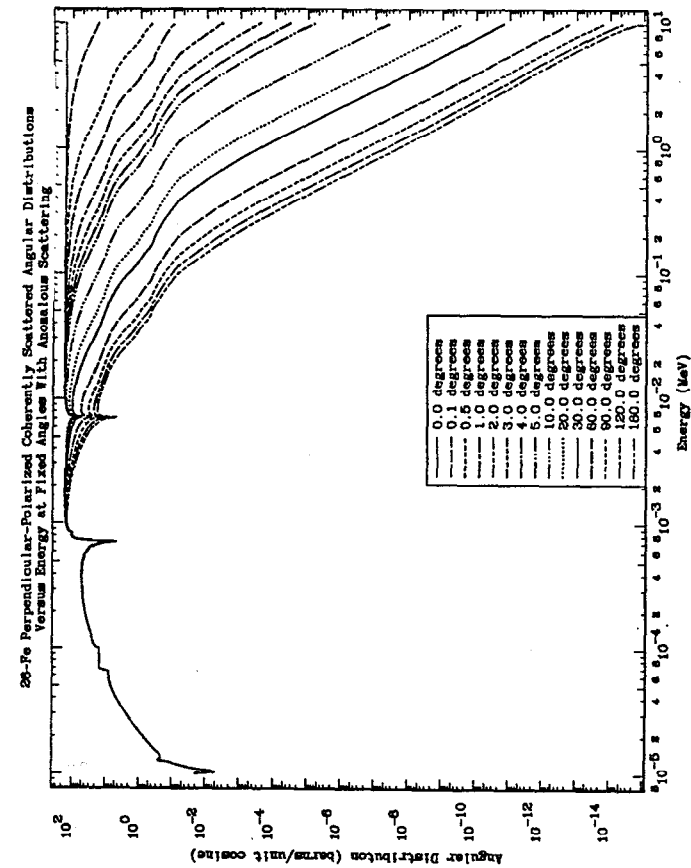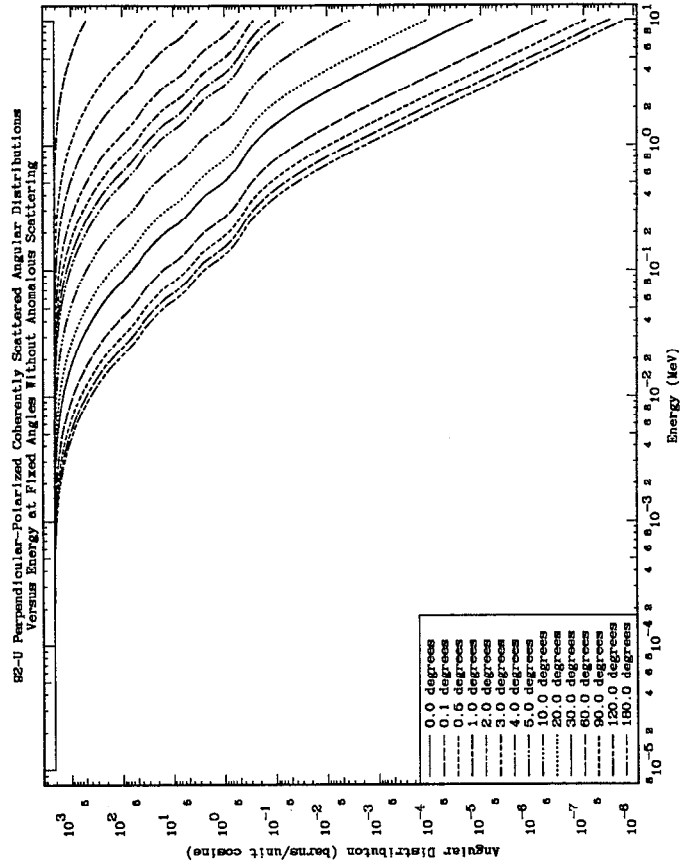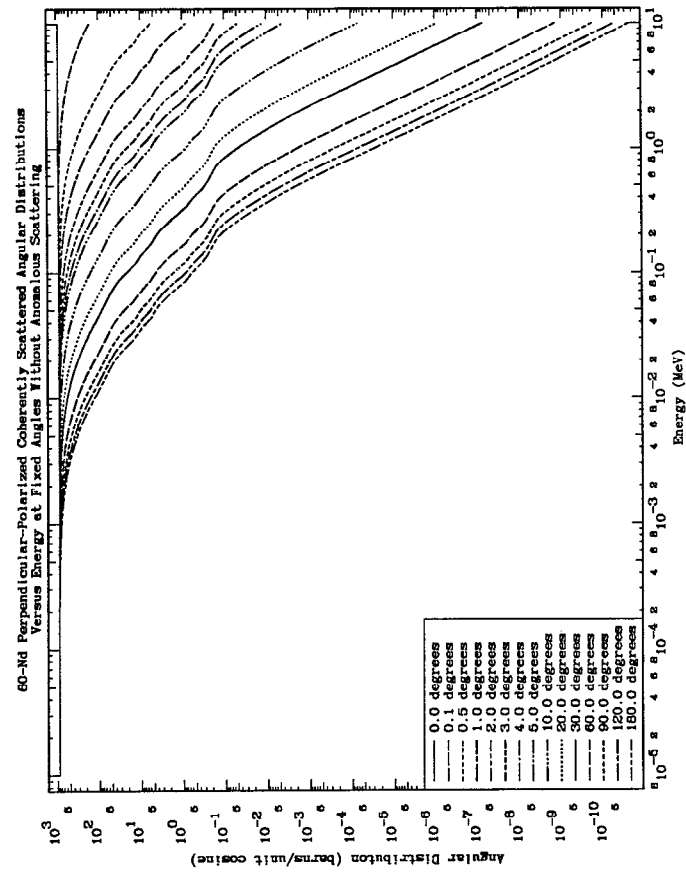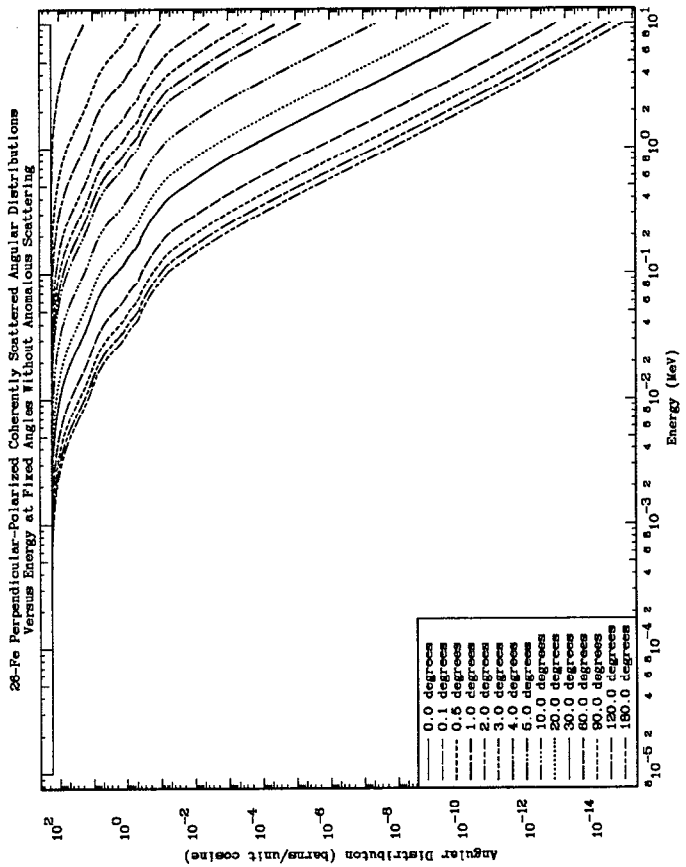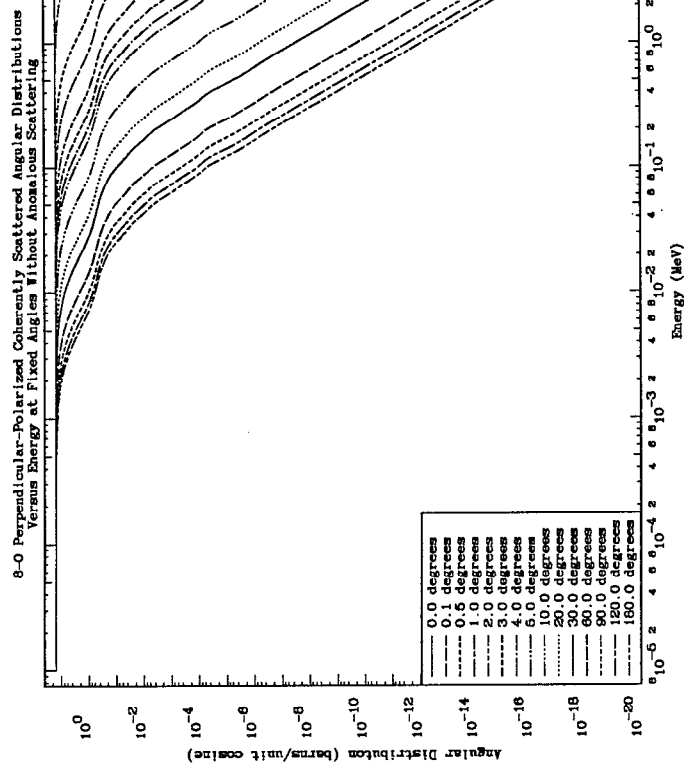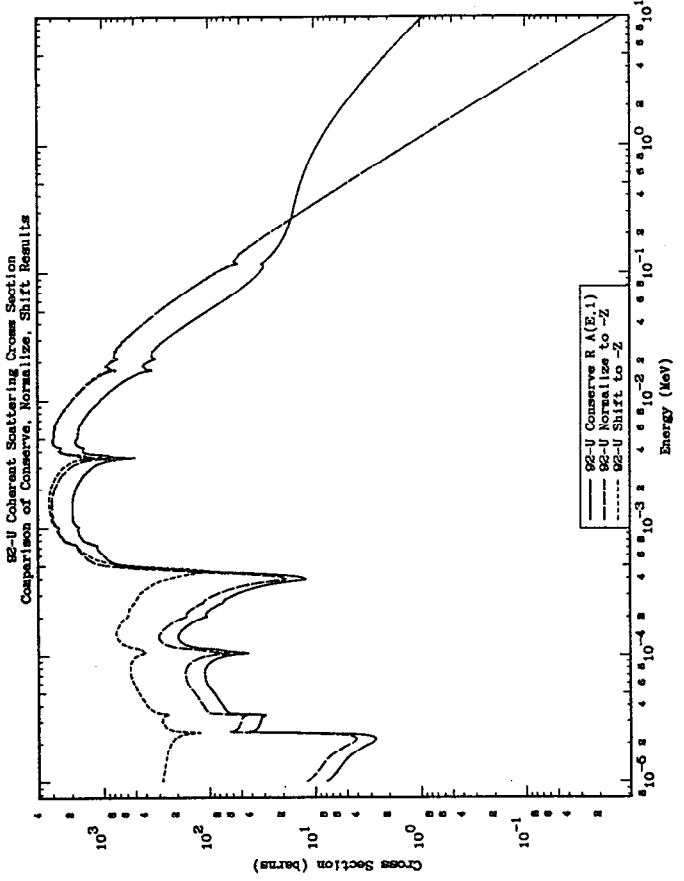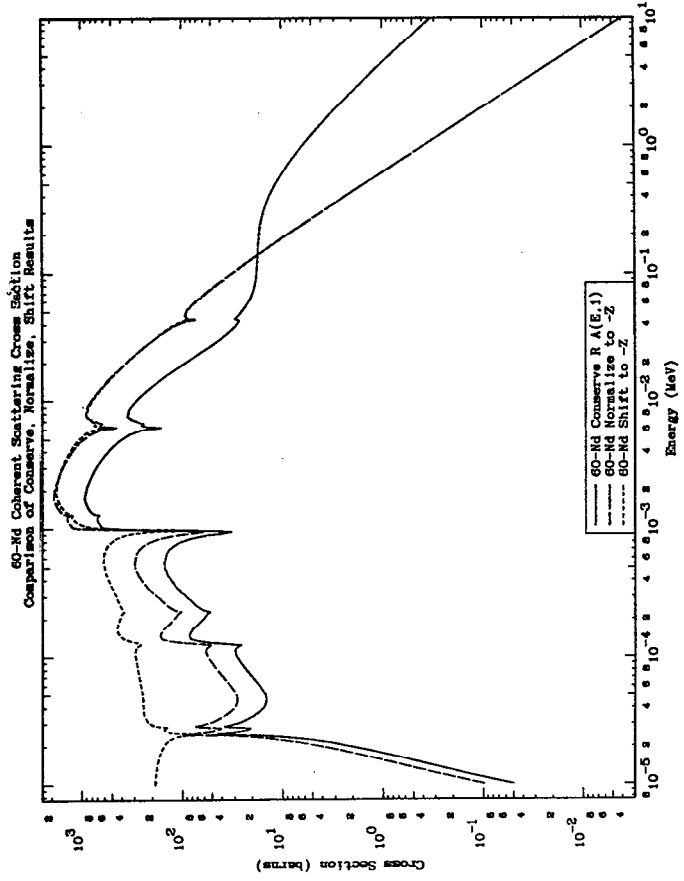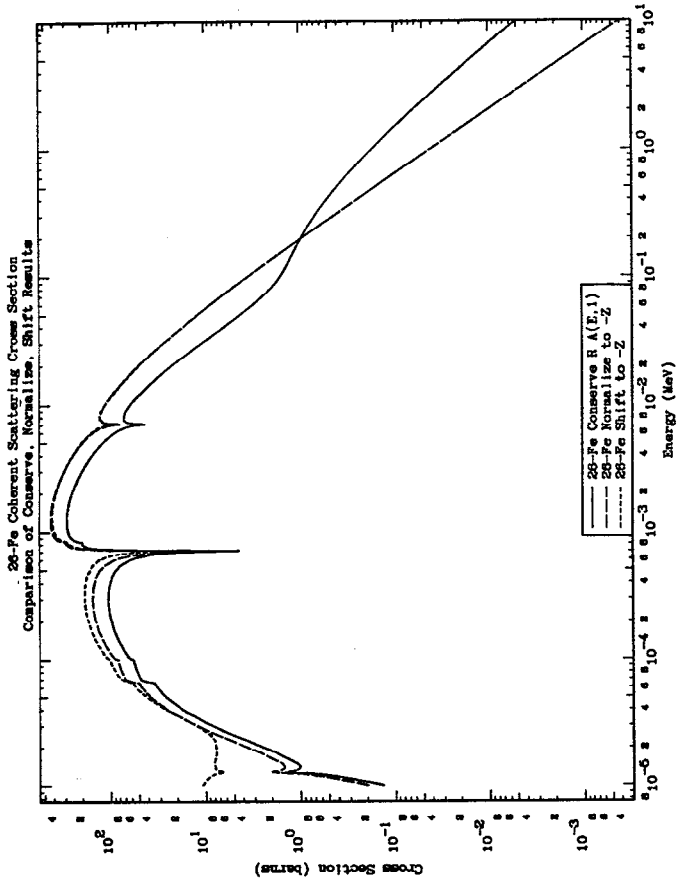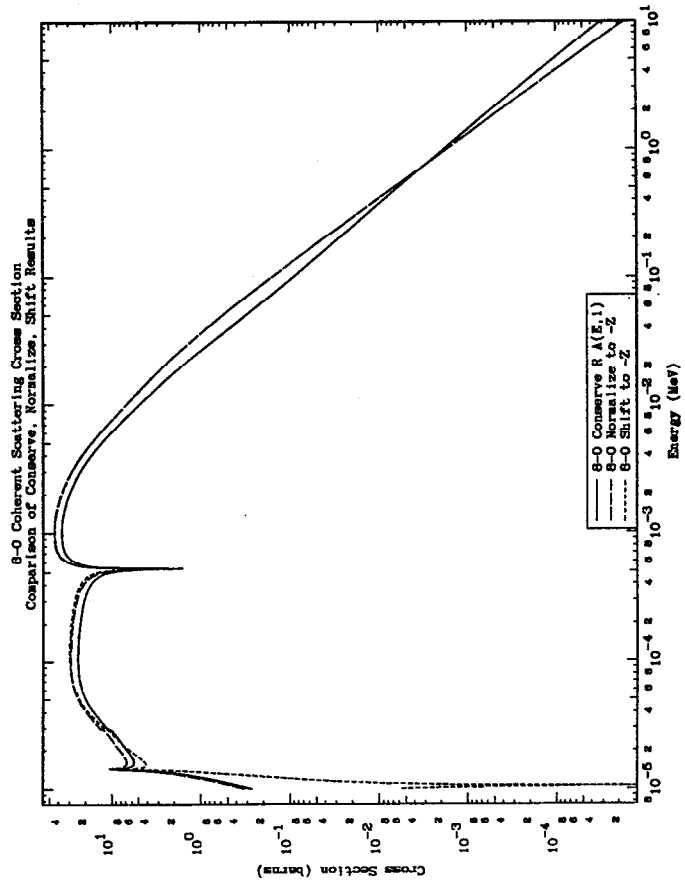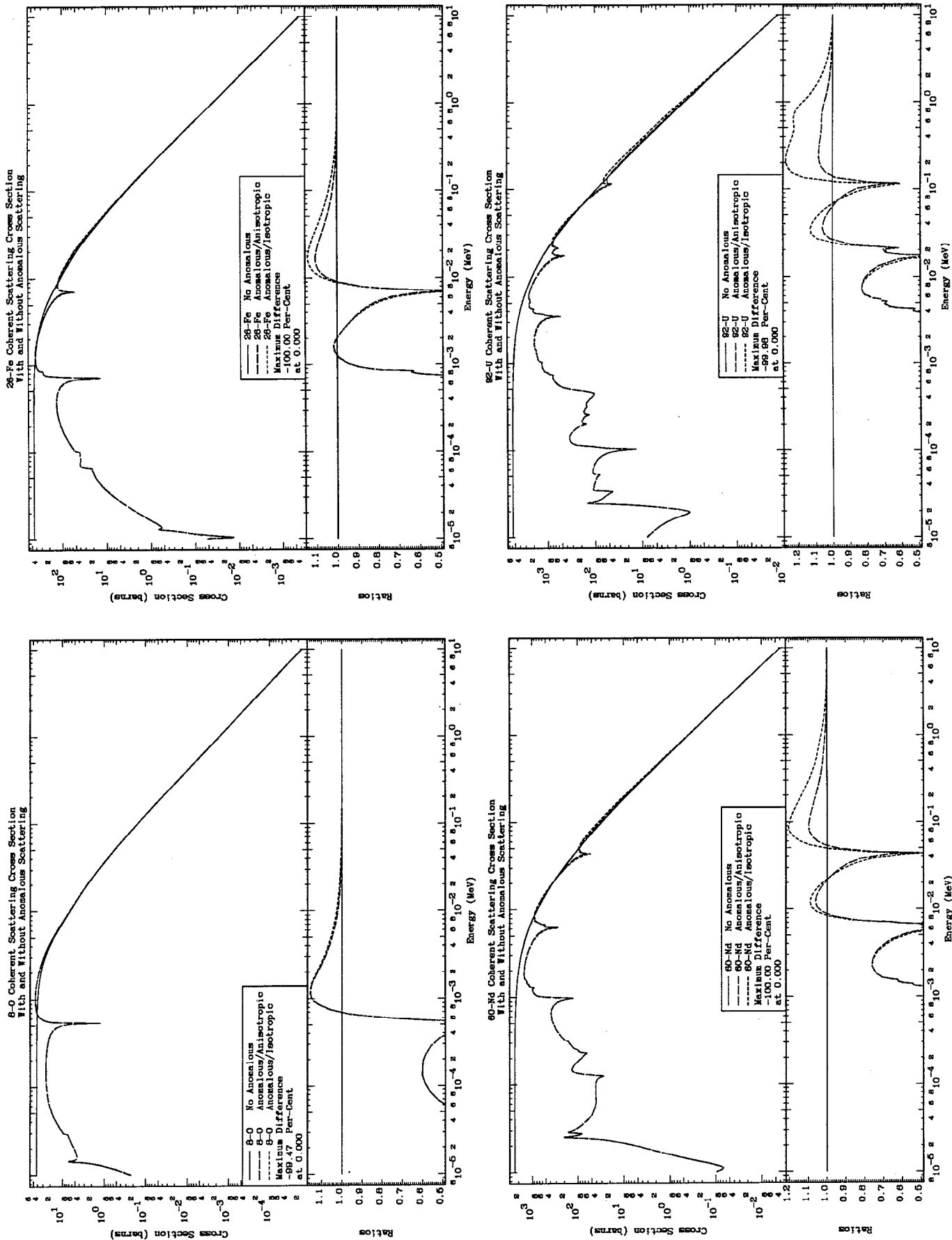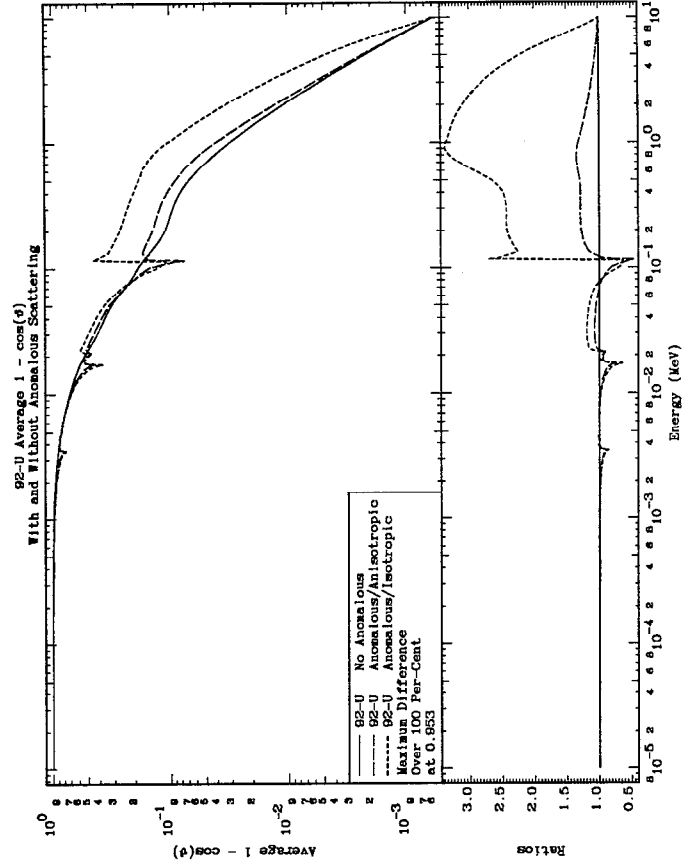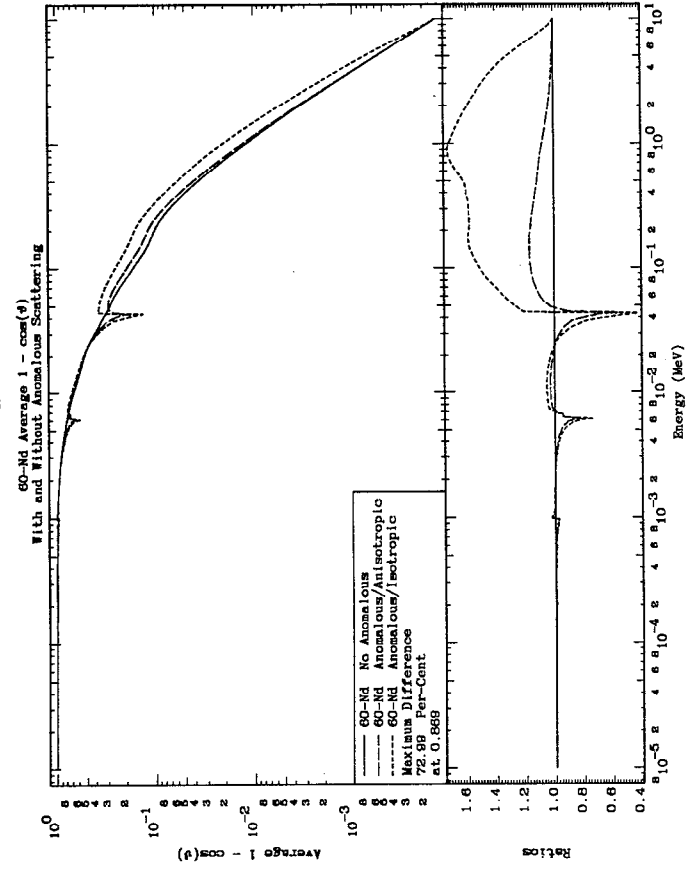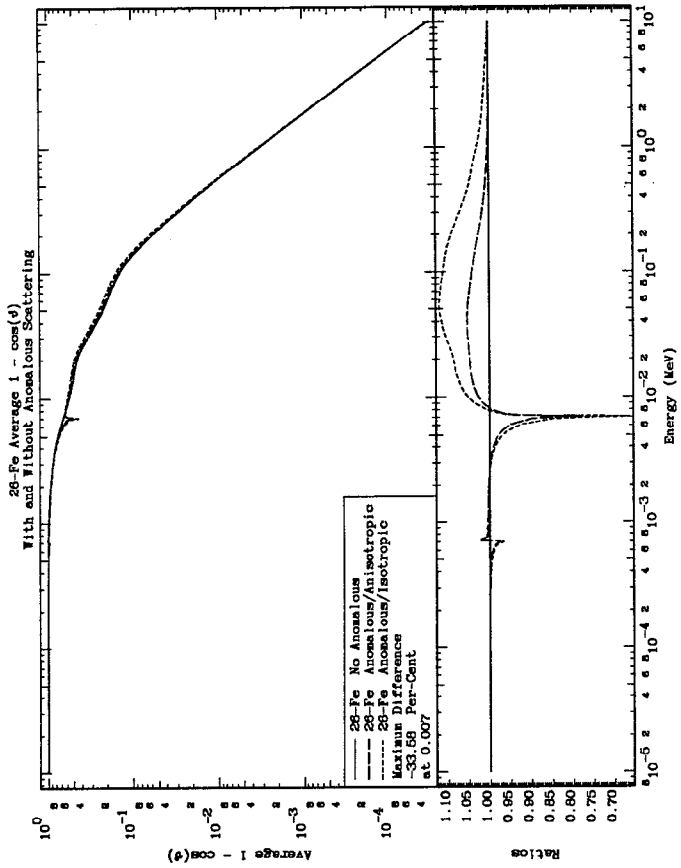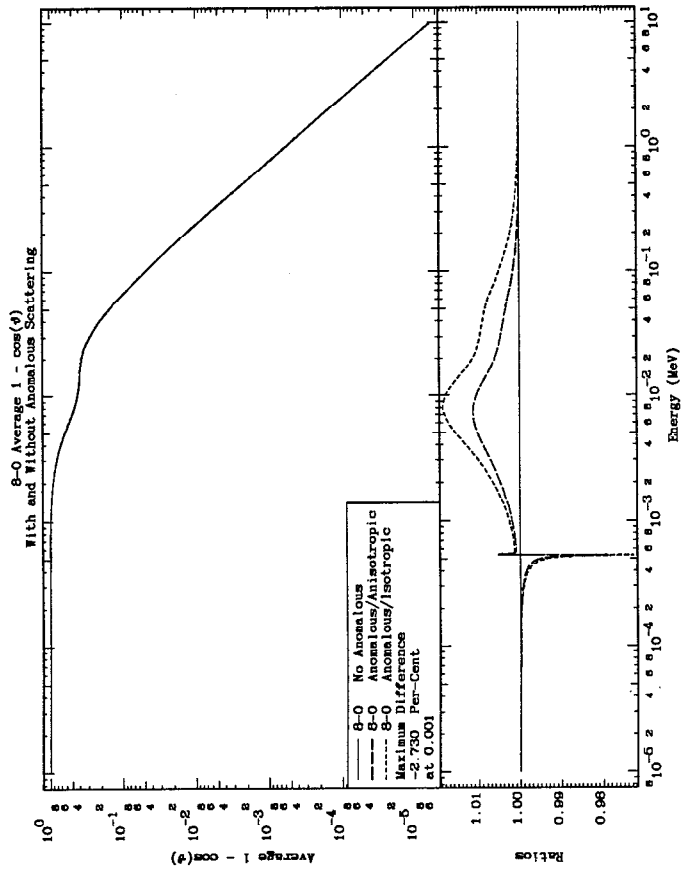
```
C      PROGRAM SCATMAN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE10,
C     1 TAPE11)
C
C      PROGRAM SCATMAN
C      VERSION 90-1 (FEBRUARY 1990)
C
C      WRITTEN BY DERMOTT E. CULLEN
C                 L-298
C                 LAWRENCE LIVERMORE NATIONAL LABORATORY
C                 P.O. BOX 808
C                 LIVERMORE CA, 94550
C                 U.S.A.
C      TELEPHONE  415-423-7359
C
C      PURPOSE
C
======================================================================
C      CALCULATE ANOMALOUS SCATTERING FACTORS AND/OR COHERENT CROSS
C      SECTIONS.
C
C      INPUT OPTIONS
C
======================================================================
C      THIS CODE HAS BEEN WRITTEN TO INCLUDE A NUMBER OF INPUT OPTIONS
C      IN ORDER TO ALLOW THE USER TO EVALUATE THE IMPORTANCE OF A
C      VARIETY OF POSSIBLE DEFINITIONS OF SCATTERING FACTORS, DESCRIBED
C      BELOW.
C
C      THIS CODE HAS ALREADY BEEN USED TO CALCULATE ANOMALOUS SCATTERING
C      FACTORS FOR THE ENTIRE LIVERMORE EVALUATED PHOTON DATA LIBRARY
C      (EPDL). THEREFORE IN ORDER TO OBTAIN STANDARD RESULTS YOU NEED
C      NOT RUN THIS CODE, YOU NEED MERELY CONTACT THE AUTHOR FOR A COPY
C      OF RESULTS.
C
C      HOWEVER, IF YOU WISH TO USE THIS CODE TO EVALUATE THE IMPORTANCE
C      OF DEFINITIONS, PLEASE USE CARE TO READ ALL OF THIS DOCUMENTATION
C      IN ORDER TO UNDERSTAND HOW TO PREPARE INPUT PARAMETERS AND HOW
C      TO INTERPRET THE RESULTS CALCULATED BY THIS CODE.
C
C      THE PRIMARY PURPOSE OF THIS CODE IS TO CALCULATE ELEMENTAL
C      ANOMALOUS SCATTERING FACTORS. THIS CODE CAN BE USED FOR A VARIETY
C      OF OTHER PURPOSES (SEE, THE DESCRIPTION OF INPUT OPTIONS), BUT
C      GENERALLY IT IS SIMPLER TO USE THIS CODE TO ESTABLISH A DATA
C      BASE OF ANOMALOUS SCATTERING FACTORS (WHICH REQUIRES CONSIDERABLE
C      COMPUTER TIME) AND TO THEN USE THIS DATA BASE IN COMBINATION WITH
C      A VARIETY OF DIFFERENT MODELS AND ASSUMPTIONS TO CALCULATE THE
C      PHOTON SCATTERING ANGULAR DISTRIBUTIONS AND CROSS SECTIONS. FOR
C      THIS PURPOSE 2 OTHER CODES MAY BE USED,
C
C      COHERT   = FOR COHERENT SCATTERING
C      INCOHERT = FOR INCOHERENT SCATTERING
C
```

```
C     THESE CODES ARE AVAILABLE FROM THE AUTHOR.
C
C     OPERATING MODES
C
======================================================================
C     THIS PROGRAM MAY BE USED IN ONE OF THREE OPERATING MODES
C     (1) CALCULATE COHERENT CROSS SECTION WITHOUT ANOMALOUS SCATTERING
C          - ONLY READ FORM FACTORS AND CALCULATE CROSS SECTIONS
C     (2) CALCULATE ANOMALOUS SCATTERING FACTORS
C          - READ CROSS SECTIONS AND CALCULATE SCATTERING FACTORS
C          - THIS MODE MAY ALSO BE USED TO THEN CALCULATE CROSS SECTIONS
C     (3) CALCULATE CROSS SECTIONS WITH ANOMALOUS SCATTERING
C          - READ ANOMALOUS SCATTERING FACTORS AND CALCULATE CROSS
C            SECTIONS.
C
C     THE OPERATING MODE IS SELECTED BY THE PROGRAM BASED ON INPUT
C     (1) THIS MODE IS USED IF NO ANOMALOUS SCATTERING DEPENDENT
C          OUTPUT IS REQUESTED.
C     (2) THIS MODE IS USED IF THE ENDL FORMATTED DATA IS CROSS SECTIONS
C     (3) THIS MODE IS USED IF THE ENDL FORMATTED DATA IS ANOMALOUS
C          SCATTERING FACTORS.
C
C     THE RECOMMENDED PROCEDURE IS TO USE THIS CODE TO ESTABLISH A DATA
C     BASE OF SUBSHELL ANOMALOUS SCATTERING FACTORS - THIS DATA BASE
C     IS EXPENSIVE TO GENERATE, IN TERMS OF COMPUTER TIME, BUT NEED
C     ONLY BE CREATED ONCE - ONCE IT IS ESTABLISHED IT CAN BE USED
C     WITH THIS PROGRAM TO ECONOMICALLY DETERMINE THE EFFECT OF ALL
C     THE MODELS DESCRIBED BELOW.
C
C     CONVENTIONS AND DEFINITIONS
C
======================================================================
C     THIS PROGRAM USES THE CONVENTIONS AND DEFINITIONS OF THE REAL
C     AND IMAGINARY SCATTERING AMPLITUDES AND SCATTERING FACTORS AS
C     USED AS GIVEN BY P.P. KANE, L. KISSEL, R.H. PRATT AND S.C. ROY
C     IN 'ELASTIC SCATTERING OF GAMMA-RAYS AND X-RAYS BY ATOMS',
C     PHYSICS REPORTS (REVIEW SECTION OF PHYSICS LETTERS) 140, NO. 2
C     (1986) 75-159, NORTH-HOLLAND, AMSTERDAM.
C
C     DEFINITION OF ANGULAR DISTRIBUTIONS AND CROSS SECTIONS
C
======================================================================
C     THE ANGULAR DISTRIBUTION OF COHERENTLY SCATTERED PHOTONS IS
C     DEFINED TO BE,
C
C     SIG(E,COS) = 3*THOMSON/8*T(E,COS)*((FF+F'(E))**2 + (F''(E))**2)
C
C     SIG(E,COS) = ANGULAR DISTRIBUTION IN BARNS PER UNIT COSINE
C     T(E,COS)   = (1 + COS*COS) (CLASSICAL THOMSON SCATTERING)
C     THOMSON    = 6.652448D-25 CM*CM = 0.6652448 BARNS
C     FF         = THE FORM FACTOR
C     F'(E)      = THE REAL ANOMALOUS SCATTERING FACTOR
```

```
C       F''(E)      = THE IMAGINARY SCATTERING FACTOR
C       M           = MOMENTUM TRANSFER SIN(THETA/2)/LAMBDA
C       THETA       = SCATTERING ANGLE
C       LAMBDA      = WAVELENGTH
C       COS         = SCATTERING COSINE
C
C       THE COHERENT SCATTERING CROSS SECTION IS DEFINED BY INTEGRATING
C       THE ANGULAR DISTRIBUTION OVER COS.
C
C       ANOMALOUS SCATTERING
C
=======================================================================
C       WE EXPECT ANOMALOUS SCATTERING TO BE IMPORTANT AT LOW ENERGY AND
C       THAT ITS EFFECT DECREASES WITH INCREASING ENERGY SO THAT AT HIGH
C       ENERGY THE ANGULAR DISTRIBUTION AND CROSS SECTION SMOOTHLY
C       APPROACH THAT PREDICTED ONLY USING FORM FACTORS, I.E., AT HIGH
C       ENERGY F'(E) AND F''(E) APPROACH ZERO.
C
C       IN ORDER TO ASSESS THE IMPORTANCE OF THE FORM FACTOR AND THE
C       ANOMALOUS SCATTERING THIS PROGRAM CAN BE USED TO CALCULATE CROSS
C       CROSS SECTIONS USING 3 DIFFERENT MODELS,
C       (1) INCLUDING THE FORM FACTOR AND ANOMALOUS SCATTERING.
C       (2) INCLUDING ANOMALOUS SCATTERING AND ASSUMING THE FORM FACTOR
C           IS EQUAL TO Z (THE LOW ENERGY LIMIT OF THE FORM FACTOR).
C       (3) INCLUDING ONLY THE FORM FACTOR (NO ANOMALOUS SCATTERING)
C
C       WE EXPECT (1) AND (2) TO BECOME EQUAL AT LOW ENERGY WHERE THE
C       FORM FACTOR IS APPROACHING Z.
C
C       WE EXPECT (1) AND (3) TO BECOME EQUAL AT HIGH ENERGY WHERE THE
C       THE ANOMALOUS SCATERING IS APPROACHING ZERO.
C
C       DEFINITION OF FORWARD ANGLE SCATTERING AMPLITUDES
C
=======================================================================
C       THE FORWARD DIRECTION ANGULAR DISTRIBUTION CAN BE WRITTEN IN THE
C       FORM,
C
C       SIG(E,1) = ((R A(E,1))**2 + (I A(E,1))**2)
C
C       R A(E,1) = FORWARD ANGLE REAL SCATTERING FACTOR
C       I A(E,1) = FORWARD ANGLE IMAGINARY SCATTERING AMPLITUDE
C       COS      = 1
C
C       DEFINITION OF FORWARD ANGLE ANOMALOUS SCATTERING FACTORS
C
=======================================================================
C       FROM OUR INITIAL DEFINITION OF THE ANGULAR DISTRIBUTION WE CAN
C       DEFINE THE FORWARD ANGLE ANOMALOUS SCATTERING FACTORS, WHICH
C       USING THE SIGN CONVENTION OF KISSEL, CAN BE WRITTEN IN THE FORM,
C
C       R A(E,1) = -R0*(Z + F'(E))
```

```
C     I A(E,1) =  R0*(F''(E))
C
C     Z         = FORWARD ANGLE VALUE OF THE FORM FACTOR
C     R0        = 2.82D-13 CM = 0.282 SQRT(BARNS)
C
C     OR IF WE KNOW THE SCATTERING AMPLITUDES,
C
C     F'(E)    = -(R A(E,1)/R0 + Z)
C     F''(E)   =   I A(E,1)/R0
C
C     IN THIS CODE WE WILL USE DISPERSION RELATIONSHIPS TO DEFINE THE
C     AMPLITUDES AND THEN USE THE ABOVE RELATIONSHIPS TO DEFINE THE
C     SCATTERING FACTORS.
C
C     FORWARD ANGLE SCATTERING AMPLITUDES IN TERM OF CROSS SECTION
C     ===========================================================
C     DISPERSION RELATIONSHIPS WILL BE USED TO CALCULATE ANOMALOUS
C     SCATTERING AMPLITUDES. THE FORWARD ANGLE IMAGINARY AND REAL
C     SCATTERING AMPLITUDE CAN BE DEFINED IN TERMS OF ONE ANOTHER
C     USING THE TWO DISPERSION RELATIONSHIPS,
C
C                                  I A(E',0)*DE'
C     R A(E,1) = C1*E*E*INTEGRAL ----------------
C                                  E'*(E'*E' - E*E)
C
C
C                                  R A(E',0)*DE'
C     I A(E,1) =  -C1*E*INTEGRAL ----------------
C                                  (E'*E' - E*E)
C
C
C     WHERE THE INTEGRALS EXTEND FROM E' = 0 TO INFINITY.
C
C     IN ADDITION THE IMAGINARY SCATTERING AMPLITUDES CAN BE DEFINED
C     DIRECTLY IN TERMS OF THE PHOTOELCTRIC CROSS SECTION,
C
C     I A(E,1) = C2*E*SIGPE(E)
C
C     R A(E,1) = FORWARD ANGLE REAL ANOMALOUS SCATTERING AMPLITUDE
C     I A(E,1) = FORWARD ANGLE IMAGINARY ANOMALOUS SCATTERING AMPLITUDE
C     SIGPE(E) = PHOTOELECTRIC CROSS SECTION
C     C1       = 2/PI
C     C2       = 1/(4*PI*C)
C
C     WE WILL USE THE THIRD EQUATION ABOVE TO DIRECTLY DEFINE THE
C     IMAGINARY AMPLITUDE IN TERMS OF THE PHOTOELECTRIC CROSS SECTION
C     AND SUBSTITUTE THIS RELATIONSHIP INTO THE FIRST EQUATION TO
C     DEFINE THE REAL AMPLITUDE IN TERMS OF AN INTEGRAL OVER THE
C     PHOTOELECTRIC CROSS SECTION.
C
C                                  SIGPE(E')*DE'
C     R A(E,1) = C3*E*E*INTEGRAL ----------------
```

```
C                                         (E'*E' - E*E)
C
C       C3        = C1*C2 = 1/(2*PI*PI*C)
C
C       CONSIDERATION OF LIMITING VALUES
C       ================================
C       LOW ENERGY
C       ==========
C       AT LOW ENERGY THE FORM FACTOR AND ANOMALOUS SCATTERING FACTORS
C       BECOME ISOTROPIC. THEREFORE THE ANGULAR DISTRIBUTION BECOMES
C       PROPORTIONAL TO THOMSON SCATTERING, 1 +COS**2, AND THE FORWARD
C       ANGLE SCATTERING AMPLITUDES CAN BE USED TO DEFINE THE ANGULAR
C       DISTRIBUTION AT ALL ANGLES AND IN TURN THE CROSS SECTION.
C
C       FROM THE DEFINITION OF THE SCATTERING AMPLITUDES IN TERMS OF THE
C       PHOTOELECTRIC CROSS SECTION WE CAN SEE THAT BELOW THE LOWEST
C       ENERGY EDGE I A(E,1) IS ZERO AND AS LONG AS THE INTEGRAL DEFINING
C       R A(E,1) CONVERGES AS ENERGY APPROACHES ZERO (AS IT ALWAYS DOES)
C       R A(E,1) WILL APPROACH ZERO AS E**2. THEREFORE THE ANGULAR
C       DISTRIBUTION AND CROSS SECTION WILL APPROACH ZERO AS E**4. THIS
C       IS IN MARKED CONTRAST TO THE RESULTS OBTAINED USING ONLY FORM
C       FACTORS, WHICH APPROACH A CONSTANT VALUE PROPORTIONAL TO Z**2
C       AT LOW ENERGY (Z = ATOMIC NUMBER OF AN ELEMENT).
C
C       SINCE THE REAL SCATTERING AMPLITUDE APPROACHES ZERO AT LOW ENERGY
C       WE CAN SEE FROM OUR DEFINITION OF THE REAL SCATTERING FACTOR THAT
C       IT MUST APPROACH -Z, EXACTLY CANCELLING THE CONTRIBUTION OF THE
C       FORM FACTOR, IN ORDER FOR THE CROSS SECTION TO APPROACH ZERO.
C
C       AS WILL BE POINTED OUT BELOW, DEPENDING ON THE MODEL USED TO
C       ACTUALLY DEFINE SCATTERING FACTORS, THIS LOW ENERGY LIMIT OF A
C       ZERO CROSS SECTION MAY OR MAY NOT BE ATTAINED.
C
C       HIGH ENERGY
C       ===========
C       TWO ADDITIONAL RELATIONSHIPS WILL BE USED TO DEFINE THE HIGH
C       ENERGY LIMIT AS ENERGY APPROACHES INFINITY,
C
C       I A(E,1) --------> 0
C
C       R A(E,1) --------> -C3*INTEGRAL SIGPE(E')*DE'
C
C       IN ORDER TO SATISFY THESE RELATIONSHIPS, FROM THE DEFINITION OF
C       I A(E,1) IN TERMS OF THE PHOTOELECTRIC CROSS SECTION WE SEE
C       THAT WE REQUIRE THAT,
C
C       E*SIGPE(E) -------> 0, AS E APPROACHES INFINITY
C
C       AND THAT THE INTEGRAL OF SIGPE(E) BE BOUNDED.
C
C       AT HIGH ENERGY THE PHOTOELECTRIC CROSS VARIES AS 1/E AND AS
C       SUCH WILL NOT SATISFY EITHER OF THESE RELATIONSHIPS. FOR A
```

```
C      1/E CROSS SECTION I A(E,1) WILL APPROACH A CONSTANT AND
C      THE INTEGRAL DEFINING R A(E,1) WILL DIVERSE TO -INFINITY.
C
C      IN ORDER TO SATISFY THESE RELATIONSHIPS WE MUST INCLUDE THE
C      EFFECT OF BOUND PAIR PRODUCTION AT HIGH ENERGY (SEE: REF. 1).
C      THE BOUND PAIR PRODUCTION CROSS SECTION IS NOT THE SAME AS
C      THE PAIR PRODUCTION. THE BOUND PAIR PRODUCTION WILL BE
C      ZERO AT LOW ENERGY. IN THE 1 TO 10 MEV ENERGY RANGE IT WILL
C      RAPIDLY INCREASE AND BECOME EQUAL TO THE PHOTOELECTRIC
C      CROSS SECTION.
C
C      THE CORRECT CROSS SECTION TO USE TO DEFINE THE SCATTERING
C      AMPLITUDES IS THE PHOTOELECTRIC MINUS THE BOUND PAIR
C      PRODUCTION CROSS SECTION.
C
C      THEREFORE THE CORRECT CROSS SECTION TO USE SHOULD BE THE
C      PHOTOELECTRIC UP TO 1 MEV. ABOVE THIS ENERGY THE CROSS
C      SECTION SHOULD SMOOTHLY APPROACH ZERO AT 10 MEV AND BE
C      ZERO FOR ALL HIGHER ENERGIES.
C
C      BY 1 MEV THE PHOTOELECTRIC CROSS SECTION IS GENERALLY QUITE
C      SMALL AND THE DETAILS OF HOW THE CROSS SECTION APPROACHES
C      ZERO IN THE 1 TO 10 MEV ENERGY RANGE HAS LITTLE EFFECT ON
C      R A(E,1). THE ONLY IMPORTANT POINT IS THAT IT NOT BE DEFINED
C      AS EXTENDING AS 1/E TO INFINITY. IN THIS CODE ABOVE 1 MEV THE
C      CROSS SECTION WILL BE DEFINED TO BE CONTINUOUS AT 1 MEV AND
C      VARIE BETWEEN 1 AND 10 MEV AS PROPORTIONAL TO (10 - E)/E**3
C      AND TO BE ZERO ABOVE 10 MEV.
C
C      WE CAN SEE THAT IN ORDER TO OBTAIN CONSISTENT RESULTS WHICH
C      SMOOTHLY APPROACH THE RESULTS OBTAINED USING ONLY FORM FACTORS,
C      FROM THE DEFINITION OF OUR SCATTERING AMPLITUDES AND FACTORS WE
C      MUST HAVE,
C
C      R A(E,1) --------> -C3*INTEGRAL SIGPE(E')*DE' = -R0*Z
C
C      SINCE ONLY IN THIS CASE WILL F'(E) APROACH ZERO AT HIGH ENERGY.
C      IT WILL BE POINTED OUT BELOW THAT IN ALMOST NO REAL CASE IS THIS
C      LIMITING VALUE OBTAINED AND UNLESS ADDITIONAL ASSUMPTIONS OR
C      APPROXIMATIONS ARE INTRODUCED, THE RESULTS WILL BE INCONSISTENT.
C
C      EXPLICIT AND IMPLICIT EDGES
C
=======================================================================
C      IT MUST BE REALIZED THAT THE INTEGRAL USED TO DEFINE THE REAL
C      SCATTERING AMPLITUDE EXTENDS FROM ZERO TO INFINITE ENERGY. NEAR
C      PHOTOELECTRIC EDGES, WHERE THERE IS AN ABRUPT CHANGE IN THE
C      CROSS SECTION WE WILL OBTAIN EXPLICIT EDGE EFFECTS, WHERE THE
C      REAL SCATTERING AMPLITUDE CHANGES VERY RAPIDLY.
C
C      IF CROSS SECTIONS ARE ONLY AVAILABLE OVER A SMALLER ENERGY RANGE
C      AND THESE ARE USED TO CALCULATE SCATTERING AMPLITUDES, ONE MUST
```

```
C      REALIZE THAT IF THE INTEGRAL IS TRUNCATED TO THE ENERGY RANGE OF
C      THE CROSS SECTION YOU ARE IMPLICITLY SAYING THERE IS AN EDGE AT
C      THE END OF THE RANGE OF INTEGRATION AND YOU SHOULD EXPECT TO
C      OBTAIN RESULTS INCLUDING IMPLICIT EDGE EFFECTS NEAR THE END OF
C      OF THE ENERGY RANGE OF INTEGRATION. FOR EXAMPLE, MATHEMATICALLY
C      THERE IS ABSOLUTELY NO DIFFERENCE BETWEEN YOU SPECIFYING THAT
C      THERE IS AN EDGE AT 10 EV AND THE CROSS SECTION IS ZERO AT ALL
C      LOWER ENERGY - AND YOU USING CROSS SECTIONS THAT ONLY EXTEND DOWN
C      TO 10 EV AND THEN SIMPLY TRUNCATE THE INTEGRAL AT THIS POINT -
C      IN EITHER CASE THE RESULTS WILL BE THE SAME AND WILL INCLUDE
C      TYPICAL EDGE EFFECTS - THE SAME IS TRUE AT THE UPPER ENERGY
C      LIMIT OF THE RANGE OF INTEGRATION.
C
C      DATA DOWN TO EDGE AND UP TO INFINITY
C
=====================================================================
C      THE TABULATED CROSS SECTIONS USED BY THIS PROGRAM ARE GIVEN OVER
C      THE ENERGY RANGE 10 EV TO 100 GEV. THE DISPERSION INTEGRAL IS
C      OVER 0 TO INFINITE ENERGY. IF THE EDGE FOR A SUBSHELL IS BELOW
C      10 EV THE CROSS SECTION CAN BE EXTENDED FROM 10 EV DOWN TO THE
C      EDGE AS 1/E**3 WITH CONTINUITY AT 10 EV. AS DESCRIBED ABOVE THE
C      CROSS SECTION WILL BE DEFINED TO SMOOTHLY APPROACH ZERO AT 10 MEV
C      AND TO BE ZERO AT ALL HIGHER ENERGIES.
C
C      WITH THIS APPROACH THE CROSS SECTIONS ARE DEFINED AT ALL ENERGIES
C      AND IMPLICIT EDGE EFFECTS CAN BE AVOIDED.
C
C      HOWEVER, IF YOU WISH TO EVALUATE THE EFFECT OF IMPLICIT EDGES YOU
C      HAVE THE INPUT OPTION NOT TO EXTEND CROSS SECTIONS BELOW 10 EV.
C      IN THIS CASE YOU EXPECT TO SEE IMPLICIT EDGE EFFECTS NEAR 10 EV.
C
C      RESOLUTION
C      ==========
C      IT CAN BE SHOWN THAT AS LONG AS THE CROSS SECTION USED TO DEFINE
C      THE ANOMALOUS SCATTERING AMPLITUDES IS CONTINUOUS ONE CAN OBTAIN
C      CONSISTENT, BOUNDED ANOMALOUS SCATTERING AMPLITUDES. OBVIOUSLY
C      IF WE TREAT THE PHOTOELECTRIC CROSS SECTION AS HAVING INFINITELY
C      NARROW EDGES WE WILL HAVE A DISCONTINUITY IN THE CROSS SECTION
C      WHICH WILL PREVENT THE RESULTS FROM BEING CONSISTENT AND BOUNDED,
C      E.G., THE REAL AMPLITUDE WILL HAVE A LOG POLE AT THE EDGES.
C
C      FOR USE IN APPLICATIONS WE REQUIRE THAT THE AMPLITUDES BE
C      CONSISTENT AND BOUNDED AND AS SUCH WE MUST DEFINE CONTINUOUS
C      CROSS SECTIONS FOR USE IN EVALUATING THE DISPERSION INTEGRALS.
C
C      BY CONSISTENT WE MEAN THAT THE RESULTS ARE CONSISTENT WITH THE
C      TWO DISPERSION RELATIONSHIPS DEFINING THE REAL AND IMAGINARY
C      AMPLITUDES IN TERMS OF ONE ANOTHER AND THE DIRECT RELATIONSHIP
C      USED TO DEFINE THE IMAGINARY AMPLITUDE IN TERMS OF THE CROSS
C      CROSS. BY COMBINING THESE THREE RELATIONSHIPS WE CAN OBTAIN A
C      HOMOGENEOUS INTEGRAL EQUATION ONLY INVOLVING THE CROSS SECTION.
C      AS SUCH WE CANNOT EXPECT TO START FROM MERELY ANY CROSS SECTION
```

```
C      AND OBTAIN RESULTS WHICH ARE CONSISTENT WITH ALL THREE OF OUR
C      RELATIONSHIPS - WE MUST PLACE RESTRAINTS ON THE CROSS SECTION
C      IF WE ARE TO OBTAIN CONSISTENT RESULTS - THE MOST OBVIOUS
C      CONSTRAINT THAT WE MUST PLACE ON THE CROSS SECTION IS THAT IT BE
C      CONTINUOUS.
C
C      ONE CAN ARGUE THAT PHOTOELECTRIC EDGES CANNOT BE INFINITELY NARROW
C      DUE TO FINITE LIFETIMES, ETC. AND USE THIS ARGUMENT TO VARY THE
C      CROSS CONTINUOUSLY ACROSS THE EDGE. WE WILL TAKE A MORE PRAGMATIC
C      APPROACH AND MERELY STATE THAT WE WISH TO DEFINE ANOMALOUS
C      SCATTERING AMPLITUDES FOR SUBSEQUENT USE IN APPLICATIONS, IN
C      PARTICULAR WE WISH TO SIMULATE EXPERIMENTAL CONDITIONS IN ORDER TO
C      BE ABLE TO REPRODUCE EXPERIMENTAL MEASUREMENTS. SINCE MEASUREMENTS
C      HAVE A FINITE ENERGY RESOLUTION WE WILL DEFINE OUR ANOMALOUS
C      SCATTERING AMPLITUDES IN ORDER TO SIMULATE THIS RESOLUTION.
C
C      WE WILL SIMULATE RESOLUTION BROADENING BY STARTING FROM THE
C      PHOTOELECTRIC CROSS SECTION AND DEFINING A RESOLUTION BROADENED
C      CROSS SECTION,
C
C                    INT SIGPE(E')*EXP[-A*(E'-E)**2]*DE'
C      SIGB(E,A)  = -----------------------------------
C                        INT EXP[-A*(E'-E)**2]*DE'
C
C      EVEN IN THE CASE WHERE THE PHOTOELECTRIC CROSS SECTION IS TREATED
C      AS HAVING INFINITELY NARROW EDGES, THE BROADENED CROSS SECTION
C      WILL BE SMOOTH, NOT HAVE ANY DISCONTINUITIES AND CAN YIELD RESULTS
C      WHICH ARE CONSISTENT WITH BOTH DISPERSION RELATIONSHIPS AND THE
C      DIRECT DEFINITION OF THE IMAGINARY AMPLITUDE IN TERMS OF THE
C      PHOTOELECTRIC CROSS SECTION.
C
C      THE WIDTH OF THE GAUSSIAN BROADENING KERNEL (A) WILL BE DEFINED
C      TO PRODUCE A RESOLUTION WHICH IS A FRACTION OF THE ENERGY. THE
C      WIDTH OF THE GAUSSIAN AT ITS HALF HEIGHT IS,
C
C      1/2      = EXP[-A*(E'-E)**2]
C
C      FOR E'-E TO BE A FRACTION OF THE ENERGY WE DEFINE,
C
C      E'-E     = DE*E
C
C      1/2      = EXP[-A*(DE*E)**2]
C
C      LOG(1/2) = -A*(DE*E)**2
C      LOG(2)   =  A*(DE*E)**2
C
C      A        =LOG(2)/[(DE*E)**2]
C
C      DE IS ONE OF THE INPUT PARAMETERS TO THIS PROGRAM. IF YOU WISH TO
C      EVALUATE THE EFFECT OF RESOLUTION ON THE RESULTS YOU MAY VARY
C      THIS PARAMETER. FOR GENERAL USE IN ORDER TO SIMULATE CURRENT
C      EXPERIMENTAL RESOLUTION, 1 PER-CENT IS RECOMMENDED AS AN INPUT
```

```
C     PARAMETER.
C
C     TABULATED LINEARLY INTERPOLABLE CROSS SECTIONS
C     ==============================================
C     THE CROSS SECTIONS IN THE LIVERMORE EVALUATED PHOTON DATA LIBRARY
C     ARE DEFINED IN TERMS OF A TABLE OF (ENERGY, CROSS SECTION) PAIRS
C     WITH LOG-LOG INTERPOLATION BETWEEN TABULATED VALUES. THIS DATA CAN
C     EASILY BE REDUCED TO WITHIN ANY REQUIRED ACCURACY TO A TABLE OF
C     (ENERGY, CROSS SECTION) PAIRS WITH LINEAR-LINEAR INTERPOLATION
C     BETWEEN TABULATED VALUES. THIS PROCEDURE IS USED IN THIS CODE IN
C     ORDER TO PRODUCE A TABLE OF LINEARLY INTERPOLATED CROSS SECTIONS
C     TO BE USED TO CALCULATE THE RESOLUTION BROADENED CROSS SECTIONS.
C
C     THE INTERVAL HALVING TECHNIQUE IS THEN USED TO CALCULATE THE
C     RESOLUTION BROADENED CROSS SECTION TO PRODUCE TABULATED RESULTS
C     WHERE LINEAR INTERPOLATION CAN BE USED TO DEFINE THE CROSS
C     SECTION BETWEEN TABULATED VALUES. THIS RESOLUTION BROADENED,
C     TABULATED LINEARLY INTERPOLABLE CROSS SECTION IS THEN USED TO
C     DEFINE THE SCATTERING AMPLITUDES.
C
C     STARTING FROM THIS DATA THE IMAGINARY SCATTERING AMPLITUDE CAN BE
C     DEFINED DIRECTLY IN TERMS OF THE CROSS SECTION AND THE REAL
C     SCATTERING AMPLITUDE CAN BE DEFINED IN TERMS OF THE INTEGRAL
C     OVER A NUMBER OF ENERGY INTERVALS, IN EACH OF WHICH THE CROSS
C     SECTION IS LINEARLY INTERPOLABLE,
C
C     CALCULATION OF FORWARD ANGLE SCATTERING AMPLITUDES
C     ==================================================
C     STARTING FROM TABULATED DATA WE CAN DEFINE THE INTEGRAL AS A SUM
C     OVER ENERGY INTERVALS IN THE TABLUATED DATA,
C
C
C                                                   SIGB(E')*DE'
C     R A(E,1) = C3*E*E* SUM (K) INTEGRAL (EK - EK+1) ----------------
C                                                     (E'*E' - E*E)
C
C     THEREFORE IN ORDER TO DEFINE R A(E,1) WE NEED ONLY BE CONCERNED
C     WITH DEFINING THIS INTEGRAL FOR ONE SUCH INTERVAL AND THEN SUMMING
C     OVER ALL CONTRIBUTING INTERVALS. ALTERNATIVELY WE MAY EXPRESS THIS
C     INTEGRAL AS THE SUM OVER CONTRIBUTIONS FROM EACH TABULATED CROSS
C     SECTION POINT, WHICH IS WHAT WE WILL ACTUALLY DO.
C
C     CALCULATION OF INTEGRALS
C     ========================
C     IF WE START WITH A TABLE OF CROSS SECTIONS WHICH ARE LINEARLY
C     INTERPOLABLE THEN THE CROSS SECTION MAY BE EXPRESSED IN A FORM IN
C     WHICH A CROSS SECTION WHICH IS TABULATED AT ENERGY EK CONTRIBUTES
C     TO THE CROSS SECTION IN A LINEAR FASHION ONLY IN THE ENERGY
C     INTERVALS EK-1 TO EK AND EK TO EK+1 WHERE THE CONTRIBUTION OF
C     THE POINT IS,
C
C     SIGB(EK)*(E - EK-1)/(EK - EK-1) : E FROM EK-1 TO EK
C     SIGB(EK)*(EK+1 - E)/(EK+1 - EK) : E FROM EK TO EK+1
```

```
C     0                                      : E LESS THAN EK-1
C                                              OR GREATER THAN EK+1
C
C     NOTE, AT EK THE CROSS SECTION IS CONTINUOUS AND REDUCES TO THE
C     TABULATED VALUE SIGB(EK). BETWEEN EK-1 AND EK THE CONTRIBUTION
C     VARIES FROM ZERO AT EK-1 TO SIGB(EK) AT EK. BETWEEN EK AND EK+1
C     THE CONTRIBUTION VARIES FROM SIGB(EK) AT EK TO ZERO AT EK+1. FOR
C     ENERGIES LESS THAN EK-1 OR GREATER THAN EK+1 THERE IS NO
C     CONTRIBUTION FROM THIS TABULATED POINT.
C
C     CONSISTENT WITH THE ABOVE DEFINITIONS WE WILL ASSUME THAT THE
C     TABULATED CROSS SECTIONS STARTS WITH A ZERO CROSS SECTION (AT
C     AN ENERGY BELOW THE LOWEST LYING EDGE WHERE THE RESOLUTION
C     BROADENING WILL SMOOTHLY REDUCE IT TO ZERO) AND ENDS WITH A
C     ZERO CROSS SECTION (AT 10 MEV WHERE THE COMBINATION OF THE
C     PHOTOELECTRIC AND BOUND PAIR PRODUCTION SMOOTHLY APPROACH
C     ZERO).
C
C     WITH THIS DEFINITION WE NEED ONLY BE CONCERNED WITH THE
C     CONTRIBUTION OF ONE TABULATED DATA POINT TO THE INTEGRAL TO
C     DEFINE THE REAL SCATTERING AMPLITUDE. THE INTEGRAL WILL THEN
C     MERELY BE A SUM OVER ALL TABULATED POINTS.
C
C     EFFECT OF ONE DATA POINT
C     ========================
C     SEPARATION OF DISPERSION RELATIONSHIP INTO 2 SIMPLE POLES
C     ========================================================
C                              SIG(E')*DE'
C     R A(E,1) = C1*E*E*INTEGRAL -------------
C                              (E'*E' - E*E)
C
C             = C*E*E*INTEGRAL  SIG(E')*[(E' + E) - (E' - E)]*DE'/(2*E)
C
-----------------------------------------
C                                    (E' - E)*(E' + E)
C
C             = C*E/2*INTEGRAL [SIG(E')*DE'/(E' - E)
C                            - SIG(E')*DE'/(E' + E)]
C
C             = C*E/2*[F(E) - F(-E)]
C
C     F(E)     = INTEGRAL [SIG(E')*DE'/(E' - E)]
C
C     LINEARLY INTERPOLABLE TABULATED CROSS SECTIONS
C     ==============================================
C     A CROSS SECTION TABULATED AT E1 WILL BE LINEARLY INTERPOLABLE
C     BETWEEN E0 AND E1, WILL BE ZERO AT E0 AND SIG(E1) AT E1. SIMILARLY
C     IT WILL BE LINEARLY INTERPOLABLE BETWEEN E1 AND E2, WILL BE
C     SIG(E1) AT E1 AND ZERO AT E2,
C
C     SIG(E)   = SIG(E1)*(E - E0)/(E1 - E0) : E = E0 TO E1
C              = SIG(E1)*(E2 - E)/(E2 - E1) : E = E1 TO E2
```

```
C                 = 0                              : E < E0 OR E > E2
C
C       CALCULATION OF F(E)
C       ===================
C       F(E)      = SIG(E1)*[
C                    INTEGRAL{(E' - E0)*DE'/(E' - E)/(E1 - E0)}
C                   +INTEGRAL{(E2 - E')*DE'/(E' - E)/(E2 - E1)}]
C
C       F(E)      = SIG(E1)*G(E)
C
C       G(E)      = INTEGRAL{(E' - E0)*DE'/(E' - E)/(E1 - E0)}
C                   +INTEGRAL{(E2 - E')*DE'/(E' - E)/(E2 - E1)}
C
C       WHERE THE FIRST INTEGRAL IS FROM E0 TO E1 AND THE SECOND INTEGRAL
C       FROM E1 TO E2.
C
C       NOTE THAT G(E) ONLY DEPENDS ON ENERGY, E, E0, E1 AND E2 AND IS
C       THE RESPONSE THAT ONE OBTAINS BY PLACING 1 BARN OF CROSS SECTION
C       AT EK.
C
C       ADDING AND SUBTRACTING E,
C
C       G(E)      = INTEGRAL{(E' - E + E - E0)*DE'/(E' - E)/(E1 - E0)}
C                   +INTEGRAL{(E2 - E + E - E')*DE'/(E' - E)/(E2 - E1)}
C
C       G(E)      = INTEGRAL{DE' + (E - E0)*DE'/(E' - E)}/(E1 - E0)
C                   +INTEGRAL{(E2 - E)/(E' - E)*DE' - DE'}/(E2 - E1)
C
C       SINCE,
C                 INTEGRAL{DE'/(E1 - E0)} = 1
C                 INTEGRAL{DE'/(E2 - E1)} = 1
C
C       G(E)      = 1 + (E - E0)/(E1 - E0)*INTEGRAL{DE'/(E' - E)}
C                   -1 + (E2 - E)/(E2 - E1)*INTEGRAL{DE'/(E' - E)}
C
C       G(E)        (E - E0)/(E1 - E0)*INTEGRAL{DE'/(E' - E)}
C                   +(E2 - E)/(E2 - E1)*INTEGRAL{DE'/(E' - E)}
C
C       IN THIS FORM WE CAN IMMEDIATELY INTEGRATE TO FIND,
C
C       G(E)      = (E - E0)/(E1 - E0)*LOG[|E - E1|/|E - E0|]
C                   +(E2 - E)/(E2 - E1)*LOG[|E - E2|/|E - E1|]
C
C       IN THIS FORM IT APPEARS THAT WE HAVE LOGARITHMIC POLES AT E0, E1
C       AND E2.
C
C       A NUMERICALLY STABLE DEFINITION OF F(E)
C       =======================================
C       IN  THE ABOVE FORM IT IS NOT IMMEDIATELY OBVIOUS THAT THIS
C       DEFINITION IS NUMERICALLY STABLE, IN THE SENSE OF BEING BOUNDED
C       FOR ALL VALUES OF E. IN ORDER TO DEMONSTRATE THAT IT IS BOUNDED
C       WE WILL CONVERT THE DEFINITION TO A SUM OF TERMS OF THE FORM,
```

```
C
C       X*LOG[|X|]
C
C       AND NOTE THAT,
C
C       LIM X*LOG[|X|] -----> 0 AS X -----> 0
C
C       WE CAN USE THE RELATIONSHIP,
C
C       LOG[|X|/|Y|] = LOG[|X|/|Z|] - LOG[|Y|/|Z|]
C
C       TO REWRITE G(E) IN THE FORM,
C
C       G(E)       =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
C                   +(E - E0)/(E1 - E0)*LOG[|E - E1|/|E1 - E0|]
C                   -(E - E2)/(E1 - E2)*LOG[|E - E1|/|E2 - E1|]
C                   +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E2 - E1|]
C
C       IN THIS FORM THE FIRST AND LAST TERMS ARE ALREADY IN THE REQUIRED
C       FORM OF X*LOG[|X|] AND WE NEED ONLY CONSIDER THE SECOND AND
C       THIRD TERMS. ADDING AND SUBTRACTING TERMS,
C
C       G(E)       =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
C                   +(E - E1 + E1 - E0)/(E1 - E0)*LOG[|E - E1|/|E1 - E0|]
C                   -(E - E1 + E1 - E2)/(E1 - E2)*LOG[|E - E1|/|E2 - E1|]
C                   +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E2 - E1|]
C
C       G(E)       =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
C                   +(E - E1)/(E1 - E0)*LOG[|E - E1|/|E1 - E0|]+LOG[|E - E1|
C                   -(E - E1)/(E1 - E2)*LOG[|E - E1|/|E2 - E1|]-LOG[|E - E1|
C                   +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E2 - E1|]
C
C       IN THIS FORM WE CAN SEE THAT THE TWO PURE LOG[|E - E1|] TERMS
C       EXACTLY CANCEL AND WE FIND,
C
C       G(E)       =-(E - E0)/(E1 - E0)*LOG[|E - E0|/|E1 - E0|]
C                   -(E - E1)/(E0 - E1)*LOG[|E - E1|/|E0 - E1|]
C                   +(E - E1)/(E2 - E1)*LOG[|E - E1|/|E2 - E1|]
C                   +(E - E2)/(E1 - E2)*LOG[|E - E2|/|E1 - E2|]
C                   +LOG[|E2 - E1|/|E1 - E0|]
C
C       UPON DEFINING,
C
C       X0        = (E - E0)/(E1 - E0)
C       X1        = (E - E1)/(E0 - E1)
C       X2        = (E - E1)/(E2 - E1)
C       X3        = (E - E2)/(E1 - E2)
C
C       THE SOLUTION CAN BE WRITTEN IN THE FORM,
C
C       G(E)       =- X0*LOG[|X0|] - X1*LOG[|X1|] + X2*LOG[|X2|]
C                   + X3*LOG[|X3|] + LOG[|E2 - E1|/|E1 - E0|]
```

```
C
C       SIMILARLY IN ORDER TO DEFINE G(-E) WE NEED ONLY CHANGE E TO -E AND
C       UPON DEFINING,
C
C       X4        =-(E + E0)/(E1 - E0)
C       X5        =-(E + E1)/(E0 - E1)
C       X6        =-(E + E1)/(E2 - E1)
C       X7        =-(E + E2)/(E1 - E2)
C
C       AND THE SOLUTION CAN BE WRITTEN IN THE FORM,
C
C       G(-E)     =- X4*LOG[|X4|] - X5*LOG[|X5|] + X6*LOG[|X6|]
C                   + X7*LOG[|X7|] + LOG[|E2 - E1|/|E1 - E0|]
C
C       COMBINING THESE TWO DEFINITIONS AND NOTING LOG[|E2-E1|/|E1-E0|]
C       TERMS EXACTLY CANCEL, THE SOLUTION CAN FINALLY BE WRITTEN IN THE
C       FORM,
C
C       G(E)-G(-E) = - X0*LOG[|X0|] - X1*LOG[|X1|] + X2*LOG[|X2|]
C                      + X3*LOG[|X3|]
C                      + X4*LOG[|X4|] + X5*LOG[|X5|] - X6*LOG[|X6|]
C                      - X7*LOG[|X7|]
C
C       AS NOTED ABOVE, SINCE,
C
C       LIM X*LOG(X) -----> 0 AS X APPROACHES 0
C
C       IN THIS FORM THE DEFINITION IS BOUNDED AND CONTINUOUS AS THE
C       ARGUMENT ANY OF THE LOG TERMS APPROACHES ZERO.
C
C       SOLUTION ALGORITHM
C       ==================
C       FOR E MUCH LESS THAN E0 OR MUCH GREATER THAN E2 THE FIRST
C       DEFINITION OF G(E) CAN BE ECONOMICALLY USED - THIS RELATIONSHIP
C       FOR G(E) ONLY INVOLVES 2 LOGS.
C
C       G(E)      = (E - E0)/(E1 - E0)*LOG[|E - E1|/|E - E0|]
C                  +(E2 - E)/(E2 - E1)*LOG[|E - E2|/|E - E1|]
C
C
C       G(-E)     =-(E + E0)/(E1 - E0)*LOG[|E + E1|/|E + E0|]
C                  +(E2 + E)/(E2 - E1)*LOG[|E + E2|/|E + E1|]
C
C       FOR E NEAR THE INTERVAL E0 TO E2 WE WILL USE THE MORE COMPLICATED
C       BUT COMPLETELY NUMERICALLY STABLE DEFINITION,
C
C       G(E)-G(-E) = - X0*LOG[|X0|] - X1*LOG[|X1|] + X2*LOG[|X2|]
C                      + X3*LOG[|X3|]
C                      + X4*LOG[|X4|] + X5*LOG[|X5|] - X6*LOG[|X6|]
C                      - X7*LOG[|X7|]
C
C       THIS ALGORITHM CAN BE USED TO CALCULATE RESULTS AT ANY GIVEN
```

```
C     ENERGY E.
C
C     CONSISTENT LOW AND HIGH ENERGY RESULTS
C
=======================================================================
C     AS POINTED OUT ABOVE THE RESULTS CALCULATED BY THIS CODE WILL BE
C     CONSISTENT ONLY IF THE INTEGRAL OF THE CROSS SECTION PROPERLY
C     NORMALIZED APPROACHES Z AT HIGH ENERGY. ONLY IN THIS CASE WILL
C     THE RESULTS APPROACH THOSE USING ONLY FORM FACTORS AT HIGH ENERGY.
C
C     SINCE OUR CROSS SECTIONS HAVE AN UNCERTAINTY ASSOCIATED WITH THEM
C     AND SINCE CERTAIN DATA IS MISSING, E.G., WE ONLY HAVE IONIZATION
C     DATA, BUT NOT EXCITATION DATA, WE SHOULD NOT BE SURPRISED TO FIND
C     THAT IN VIRTUALLY ALL REAL CASES OF INTEREST WE WILL NOT BE ABLE
C     TO CALCULATE RESULTS WITH EXACTLY THE REQUIRED HIGH ENERGY LIMIT.
C
C     THIS CODE WILL ALLOW YOU TO CALCULATE RESULTS USING ONE OF THREE
C     CONVENTIONS, IN AN ATTEMPT TO OBTAIN CONSISTENT RESULTS,
C
C     CONSERVE R A(,0)
C     ===============
C     USE THE RESULTS EXACTLY AS CALCULATED WITHOUT ANY CORRECTION
C     FACTORS. IN THIS CASE THE REAL SCATTERING FACTOR IS DEFINED TO
C     BE,
C     F'(E)    = -(R A(E,1)/R0 + Z)
C
C     THESE RESULTS WILL APPROACH THE CORRECT LOW ENERGY LIMIT OF A
C     ZERO CROSS SECTION, BUT GENERALLY WILL NOT APPROACH THE CORRECT
C     HIGH ENERGY LIMIT, SINCE IN ALMOST ALL CASES THE INTEGRAL OF
C     THE CROSS SECTION DOES NOT APPROACH -Z.
C
C     SHIFT TO -Z
C     ===========
C     SHIFT THE RESULTS TO FORCE THEM TO APPROACH THE CORRECT HIGH
C     ENERGY LIMIT. IN THIS CASE THE REAL SCATTERING FACTOR IS DEFINED
C     TO BE,
C     F'(E)    = -(R A(E,1)/R0 - R A(INFINITY,0))
C     WHICH BY DEFINITION APPROACHES ZERO AS E APPROACHES INFINITY.
C     THESE RESULTS WILL APPROACH THE CORRECT HIGH ENERGY LIMIT, BUT
C     GENERALLY WILL NOT APPROACH THE CORRECT LOW ENERGY LIMIT OF A
C     ZERO CROSS SECTION.
C
C     NORMALIZE TO -Z
C     ===============
C     NORMALIZE THE RESULTS TO FORCE THEM TO APPROACH THE CORRECT
C     HIGH ENERGY LIMIT. IN THIS CASE THE REAL SCATTERING IS DEFINED
C     TO BE,
C     F'(E)    = -(X*R A(E,1)/R0 + Z)
C     WHERE,
C     X        = -Z/R A(INFINITY,0)
C     SINCE NORMALIZATION IS USED THIS METHOD WILL ALSO APPROACH THE
C     CORRECT LOW ENERGY LIMIT.
```

```
C
C      THIS INPUT OPTION MAY BE USED TO ASSESS THE EFFECT THAT USING ONE
C      OR THE OTHER OF THESE DEFINITIONS MAY HAVE ON THE RESULTS. SINCE
C      ONLY NORMALIZING RETURNS WILL PRODUCE THE CORRECT LOW AND HIGH
C      ENERGY LIMITS, THIS IS THE RECOMMENDED OPTION.
C
C      THE ANGULAR DEPENDENCE OF THE SCATTERING FACTORS
C
=======================================================================
C      THE NORMAL PROCEDURE USED (SEE: REF. 1) IS TO ASSUME THAT THE
C      SCATTERING FACTORS ARE ISOTROPIC. WITH THIS ASSUMPTION ONCE WE
C      HAVE CALCULATED THE FORWARD DIRECTION SCATTERING AMPLITUDES AND
C      IN TURN USED THESE TO DEFINE OUR SCATTERING FACTORS, WE HAVE ALL
C      THE INFORMATION THAT WE REQUIRE TO CALCULATE ANGULAR DISTRIBUTIONS
C      AND CROSS SECTIONS. HOWEVER, ASSUMING ISOTROPIC SCATTERING FACTORS
C      LEADS TO INCONSISTENT RESULTS.
C
C      IN ORDER TO HAVE THE ANOMALOUS EFFECT DECREASE AND HAVE THE
C      CROSS SECTIONS APPROACH THE FORM FACTOR LIMIT AT HIGH ENERGY
C      THE SCATTERING FACTORS MUST APPROACH ZERO FASTER THAN THE
C      FORM FACTOR, OTHERWISE THE EFFECT OF COMBINING THEM WILL
C      ALWAYS HAVE A SIGNIFICANT CONTRIBUTION FROM THE ANOMALOUS
C      SCATTERING.
C
C      WITH THE NORMAL DEFINITION OF ANOMALOUS SCATTERING FACTORS, WHERE
C      THEY ARE ASSUMED TO BE ISOTROPIC, ONE DOES NOT OBTAIN CONVERGENT
C      RESULTS AT HIGH ENERGY. INDEED AT HIGH ENERGY THE CONTRIBUTION
C      OF THE ANOMALOUS SCATTERING FACTORS DOMINATE THAT OF THE FORM
C      FACTOR, FOR LARGE ANGLE SCATTERING. THIS IS TRUE REGARDLESS OF
C      WHICH OF THE THREE POSSIBLE DEFINITION DESCRIBED ABOVE IS USED
C      (CONSERVE, SHIFT, NORMALIZE).
C
C      IN ORDER TO OBTAIN CONVERGENT RESULTS THE SCATTERING FACTORS MUST
C      BE CONSIDERED TO BE ANISOTROPIC. THIS CODE WILL ALLOW THE USER TO
C      SPECIFY BY INPUT PARAMETERS EITHER ISOTROPIC OR ANISOTROPIC
C      SCATTERING FACTORS. IF ANISOTROPIC FACTORS ARE SPECIFIED THIS
C      CODE WILL USE THE ONLY INFORMATION THAT WE HAVE CONCERNING THE
C      ANGULAR DISTRIBUTION, NAMELY THE FORM FACTOR. THE FORM FACTOR
C      CAN BE WRITTEN IN THE FORM,
C
C      FF(M) = Z*Q(M)
C
C      FF(M) = FORM FACTOR
C      Z     = ATOMIC NUMBER OF AN ELEMENT
C      Q(M)  = THE ANGULAR DEPENDENCE NORMALIZED TO 1.0 AT M = 0.0
C      M     = MOMENTUM TRANSFER
C
C      WE WILL ASSUME THAT THE ANGULAR DISTRIBUTION OF THE ANOMALOUS
C      SCATTERING FACTORS IS,
C
C      F(E,M) = F(E,0)*[Q(M)]**POWER
C
```

```
C        F(E,0) = FORWARD ANGLE REAL OR IMAGINARY SCATTERING FACTOR
C        F(E,M) = ANGULAR DEPENDENT REAL OR IMAGINARY SCATTERING FACTOR
C        POWER  = THE POWER OF Q(M) USED
C
C        IT HAS BEEN EMPIRICALLY DETERMINED BY EXAMINING THE RESULTS FOR
C        ALL Z = 1 TO 100 THAT POWER = 1/4 OR MORE WILL YIELD RESULTS
C        WHICH SMOOTHLY REDUCE TO THOSE USING ONLY FORM FACTORS AT HIGH
C        ENERGY. IT IS RECOMMENDED THAT THE MINIMUM ANISOTROPY REQUIRED
C        TO PRODUCE CONVERGENT RESULTS BE USED, I.E., USE POWER = 1/4.
C
C        BELOW ABOUT 1 KEV THE FORM FACTOR IS ALMOST ISOTROPIC, SO THAT
C        INTRODUCING ANISOTROPY INTO THE SCATTERING FACTORS WILL NOT HAVE
C        ANY EFFECT ON THE ANGULAR DISTRIBUTION AND CROSS SECTION.
C
C        AT PROGRESSIVELY HIGHER ENERGIES INTRODUCING MORE ANISOTROPY,
C        I.E., USING A LARGER POWER, WILL REDUCE THE ANOMALOUS EFFECT AND
C        CAUSE THE ANGULAR DISTRIBUTION AND CROSS SECTION TO BE CLOSER TO
C        THE RESULTS OBTAINED USING ONLY FORM FACTORS.
C
C        THIS INPUT OPTION MAY BE USED TO ASSESS THE EFFECT OF ANISOTROPY
C        ON THE RESULTS.
C
C        DATA UNITS
C
======================================================================
C        IN THE FORM IN WHICH THEY WILL BE CALCULATED AND USED,
C
C        IM A(W,0) = W*SIG(W)/(4*PI*C)
C        R  A(W,0) = W*W INTERVALS SIG(W')*DW'/(W'*W' - W*W)/(4*PI*PI*C)
C
C        WHERE W IS IN UNITS OF 1/SECONDS AND C IS IN UNITS OF CM/SECONDS,
C        SO THAT BOTH NORMALIZATION FACTORS ARE IN UNITS OF 1/CM AND THE
C        RESULTS ARE IN CM. HOWEVER, RATHER THEN USE FREQUENCY, W, AS THE
C        VARIABLE OF INTEGRATION WE WILL USE ENERGY, E, WHERE,
C        E = HBAR*W (HBAR = PLANCK'S CONSTANT)
C        IN WHICH CASE THE NORMALIZATION FACTORS BECOME,
C        E/(4*PI*H*C)
C        E/(4*PI*PI*H*C)
C        C    = 2.99792D+10 CM/SECOND
C        HBAR = 6.58217D-22 MEV*SECONDS
C        USING THESE CONSTANTS RESULTS ARE IN CM. HOWEVER SIG(W) AS USED BY
C        THIS CODE IS GIVEN IN BARNS (1.0D-24 CM*CM) AND WE REQUIRE RESULTS
C        IN SQRT(BARNS), SO THAT WHEN THE SCATTERING AMPLITUDES ARE SQUARED
C        THE RESULTS ARE IN BARNS. ALLOWING FOR SIG(W) IN BARNS IN ORDER TO
C        OBTAIN THE RESULTS IN SQRT(BARNS), INSTEAD OF CM, WE MUST MULTIPLY
C        THE RESULTS BY 1.0D-12 SQRT(BARNS)/CM. NOTE, C*HBAR WILL YIELD AN
C        EXPONENT OF 1.0D-12 IN THE DENOMINATOR WHICH WILL EXACTLY CANCEL
C        THE 1.0D-12 SQRT(BARNS)/CM IN THE NUMERATOR. AS SUCH C AND HBAR
C        CAN SIMPLY BE TREATED AS EQUAL TO,
C        C      = 2.99792
C        HBAR   = 6.58217
C
```

```
C     SIMILARLY,
C     R0       = 2.82D-13 CM        = 0.282 SQRT(BARNS)
C     THOMSON = 6.652448D-25 CM*CM = 0.6652448 BARNS
C
C     DATA
C
======================================================================
C     THE FOLLOWING DATA ARE USED BY THE PROGRAM,
C     (1) PHOTOELECTRIC SUBSHELL CROSS SECTIONS FROM JIM SCOFIELD
C         TABULATED BETWEEN EITHER 10 EV OR THE EDGE ENERGY, WHICHEVER
C         IS HIGHER AND 100 GEV. THIS DATA IS USED TO CALCULATE THE
C         SCATTERING AMPLITUDES R A(E,1) AND IM A(E,1). FOR CALCULATION
C         OF R A(E,1) USING THE DISPERSION RELATIONSHIP, DEFINED ABOVE,
C         IF THE EDGE IS BELOW 10 EV (IN WHICH CASE CROSS SECTIONS ONLY
C         EXTEND DOWN TO 10 EV) THE CROSS SECTION MAY BE CONTINUED FROM
C         10 EV DOWN TO THE EDGE AS 1/E**3 (BASED ON JIM SCOFIELD'S
C         SUBSHELL EDGE DATA) - THIS IS AN INPUT OPTION.
C     (2) SUBSHELL EDGE ENERGIES FROM JIM SCOFIELD. IF THE EDGE ENERGY
C         IS BELOW 10 EV THE EDGE ENERGY IS USED TO DEFINE WHERE TO
C         EXTEND THE CROSS SECTION DOWN TO (1/E**3 BETWEEN THE EDGE
C         AND 10 EV).
C     (3) ELECTRONS PER SUBSHELL FROM JIM SCOFIELD. THIS IS USED TO
C         CHECK THE EXPECTED HIGH ENERGY LIMIT OF R A(E,1), WHICH
C         SHOULD BE EQUAL TO THE NUMBER OF ELECTRONS PER SUBSHELL.
C     (4) FORM FACTORS FROM JOHN HUBBELL. THIS IS USED TO INTEGRATE
C         THE ANGULAR DISTRIBUTIONS TO DEFINE THE COHERENT CROSS
C         SECTION.
C
C     CROSS SECTIONS AND FORM FACTORS MUST BE IN THE LIVERMORE
C     EVALUATED NUCLEAR DATA LIBRARY (ENDL) FORMAT.
C
C     ALL DATA ARE IN UNITS OF MEV AND BARNS.
C
C     DEFINING AN ENERGY GRID
C
======================================================================
C     USING THE ABOVE ALGORITHMS WE CAN CALCULATE THE SCATTERING
C     AMPLITUDES AND FACTORS AT ANY GIVEN ENERGY. THIS PROGRAM WILL
C     START FROM THE ENERGY POINTS AT WHICH THE LINEARIZED CROSS
C     SECTIONS ARE TABULATED, ADD ADDITIONAL POINTS DOWN TO 10 EV IF
C     NEEDED AND ITERATE IN EACH ENERGY INTERVAL TO PRODUCE TABULATED,
C     LINEARLY INTERPOLABLE SCATTERING FACTORS.
C
C     ENERGY RANGES
C     =============
C     BY INPUT THE USER SPECIFIES THE ENERGY RANGE OVER WHICH ANOMALOUS
C     SCATTERING FACTORS SHOULD BE CALCULATED. FOR ANY GIVEN ELEMENT
C     THE CALCULATIONS SHOULD BE EXTENDED TO WELL ABOVE ALL PHOTO-
C     ELECTRIC EDGES. FOR HIGH Z ELEMENTS THE HIGHEST ENERGY EDGE IS
C     MORE THEN 100 KEV. FOR USE IN OUR EVALUATED PHOTON DATA LIBRARY
C     (EPDL) WE HAVE ADOPTED THE ENERGY RANGE 10 EV TO 1 MEV AS THE
C     ENERGY RANGE OF NON-ZERO ANOMALOUS SCATTERING FACTORS AND WE
```

```
C      TREAT THEM AS EXACTLY ZERO ABOVE 1 MEV - THIS IS THE RECOMMENDED
C      ENERGY RANGE TO SPECIFY AS INPUT.
C
C      IN ADDITION TO THE ENERGY OF THE SCATTERING FACTORS THE OTHER
C      ENERGY RANGE OF INTEREST IS THE ENERGY RANGE OF THE COHERENT
C      CROSS SECTION. BY AN INPUT OPTION THE USER MAY SPECIFY EITHER
C      THAT THIS BE THE SAME ENERGY RANGE AS THE ANOMALOUS SCATTERING
C      FACTORS OR THAT THE CALCULATION OF CROSS SECTION BE EXTENDED
C      UP TO THE UPPER LIMIT OF THE PHOTOELECTRIC CROSS SECTION - E.G.,
C      FOR EPDL CROSS SECTIONS, UP TO 100 GEV. WHEN ANOMALOUS DATA,
C      RATHER THAN PHOTOELECTRIC CROSS SECTIONS, ARE USED AS THE STARTING
C      POINT BY THIS CODE (SEE, THE DESCRIPTION OF OPERATING MODES ABOVE)
C      IF EXTENSION IS INDICATED BY THE INPUT OPTION, THE CROSS SECTIONS
C      WILL BE EXTENDED UP TO 100 GEV, REGARDLESS OF THE ENERGY RANGE OF
C      THE SCATTERING FACTORS.
C
C      ALLOWABLE UNCERTAINTY
C      =====================
C      INPUT OPTIONS ARE USED TO SPECIFY ALLOWABLE UNCERTAINTY IN ORDER
C      TO LINEARIZE CROSS SECTIONS, CALCULATE SCATTERING FACTORS, ETC.
C      BY ALLOWABLE UNCERTAINTY WE MEAN THAT AT THE TABULATED VALUES
C      RESULTS WILL BE MORE OR LESS EXACT. BETWEEN TABULATED VALUES
C      LINEAR INTERPOLATION MAY BE USED TO DEFINE RESULTS TO WITHIN THE
C      ALLOWABLE UNCERTAINTY.
C
C      AN INPUT OPTION IS USED TO DEFINE HOW ACCURATELY TO LINEARIZE
C      CROSS SECTIONS AND TO CALCULATE RESULTS. THE RESULTS ARE FAIRLY
C      INSENSITIVE TO THE ACCURACY TO WHICH CROSS SECTIONS ARE LINEARIZED
C      AND RESULTS SHOULD NOT BE CALCULATED TO MUCH HIGHER PRECISION THEN
C      WE KNOW OUR CROSS SECTIONS. THEREFORE IT IS RECOMMENDED THAT A
C      PRECISION IN THE RANGE 0.1 TO 1.0 PER-CENT BE USED FOR THIS INPUT
C      OPTION - THIS IS SIGNIFICANTLY LESS THEN THE UNCERTAINTY OF THE
C      CROSS SECTIONS AND AS SUCH WILL NOT INTRODUCE ANY ADDITIONAL
C      ERROR.
C
C      HIGH ENERGY ASYMPTOTIC FORM
C
======================================================================
C      BOTH THE REAL AND IMAGINARY SCATTERING FACTORS APPROACH ZERO AT
C      HIGH ENERGY. THE USER OF THIS CODE HAS THE OPTION TO CUTOFF THE
C      RESULTS AT AN ENERGY WHERE THE RESULTS BECOME SMALL. IN ORDER TO
C      DEFINE SMALL THIS PROGRAM WILL USE THE CHARACTERISTIC SHAPE OF
C      THE REAL SCATTERING FACTOR. NEAR AN EDGE THE REAL SCATTERING
C      WILL HAVE A MAXIMUM. AT HIGHER ENERGIES IT WILL DECREASE, BECOME
C      NEGATIVE, PASS THROUGH A MINIMUM AND THEN BEGIN TO ASSYMPTOTICALLY
C      APPROACH ZERO. IF INPUT PARAMETERS INDICATE THE CALCULATION SHOULD
C      BE CUTOFF AT ENERGIES WELL ABOVE EDGES, THE PROGRAM WILL DO 2
C      THINGS,
C
C      (1) CHECK FOR R A(E,1) PASSING THROUGH A NEGATIVE MINIMUM VALUE
C          ABOVE THE EDGE. AFTER R A(E,1) HAS PASSED THROUGH A MINIMUM
C          CALCULATIONS WILL CONTINUE UP TO 10 TIMES THIS ENERGY AND THEN
```

```
C          THE PROGRAM WILL TEST FOR APPROACH TO ASYMPTOTIC FORM, STEP
C          (2), DESCRIBED BELOW.
C
C     (2) CHECK FOR APPROACH TO THE ASYMPTOTIC FORM,
C          R A(E,1) = R A(INFINITY,0)*[1 + Z**2 + Z**4/2 + Z**6/3...
C          WHERE Z = (EDGE ENERGY)/E
C          THE COMPUTED VALUE OF R A(E,1) WILL BE COMPARED TO THIS
C          ASYMPTOTIC FORM AND WHEN IT IS WITHIN THE ALLOWABLE
C          UNCERTAINTY (AN INPUT PARAMETER) THE ASYMPTOTIC FORM WILL
C          BE USED TO DEFINE R A(E,1) AT ALL HIGHER ENERGIES. FOR
C          SUBSHELLS WITH LOW EDGE ENERGIES (HYDROGEN ABOUT 13.6 EV)
C          THIS METHOD WILL GREATLY REDUCE RUNNING TIME WITHOUT
C          INTRODUCING ANY APPRECIABLE ERROR, E.G, WITHOUT USING
C          THIS METHOD IF WE WISH TO OBTAIN RESULTS FOR 10 EV UP TO
C          1 MEV, BY 1 MEV WE WILL BE AT ALMOST 100,000 THE EDGE
C          ENERGY AND THE RESULTS WILL TO ESSENTIALLY EQUAL TO ZERO
C          FROM MUCH LOWER ENERGIES UP TO 1 MEV.
C
C     IT IS RECOMMENDED THAT THE CALCULATION BE CUTOFF WHEN THE APPROACH
C     TO THE ASYMPTOTIC FORM IS ABOUT 1/10 IS THE ALLOWABLE UNCERTAINTY
C     IN THE CROSS SECTION, E.G., IF YOU SPECIFY AN ALLOWABLE ERROR OF
C     1 PER-CENT FOR THE CROSS SECTION, SPECIFY 0.1 PER-CENT AS THE
C     CUTOFF PARAMETER - USING A SMALLER CUTOFF ONLY CAUSE THE PROGRAM
C     TO RUN ALONG AND WILL NOT SIGNIFICANTLY IMPROVE THE ACCURACY OF
C     THE RESULTS - USING A MUCH LARGER CUTOFF CAN LEAD TO INACCURATE
C     RESULTS AT HIGH ENERGY.
C
C     CROSS SECTION USED TO CALCULATE ANOMALOUS SCATTERING FACTORS
C     ===========================================================
C     NORMALLY THE PHOTOELECTRIC CROSS SECTION WILL BE USED TO CALCULATE
C     ANOMALOUS SCATTERING FACTORS. HOWEVER, THIS CODE HAS BEEN DESIGNED
C     TO ALLOW ANY CROSS SECTION TO BE USED TO CALCULATE THE ANOMALOUS
C     SCATTERING FACTORS. THIS OPTION IS INTENDED ONLY BE ALLOW THE
C     USER TO DETERMINE THE EFFECT OF USING ANOTHER CROSS SECTION, E.G.,
C     INCOHERENT, PAIR PRODUCTION, ETC. IN THE BELOW INPUT COLUMNS
C     11-20 OF THE FIRST INPUT LINE REFER TO THE CROSS SECTION WHICH
C     SHOULD BE USED TO CALCULATE THE ANOMALOUS SCATTERING FACTORS.
C     TO USE PHOTOELECTRIC CROSS SECTIONS THE MINIMUM AND MAXIMUM C
C     (ENDL REACTION DESIGNATOR) SHOULD BE 73.
C
C     INPUT PARAMETERS
C
=======================================================================
C     3 LINES OF INPUT ARE REQUIRED FOR EACH CALCULATION. THE 3 LINES
C     MAY BE REPEATED ANY NUMBER OF TIMES. REQUESTED Z RANGES MAY BE
C     IN ANY ORDER, BUT PROGRAM EXECUTION TIME MAY BE MINIMIZED BY
C     SPECIFYING REQUESTED Z IN ASCENDING ORDER (E.G., IF NOT THE
C     PROGRAM WILL HAVE TO REWIND THE DATA FILES AND SEARCH FROM THE
C     BEGINNING).
C
C     IF ANY INPUT FIELD IS ZERO IN WILL BE ASSIGNED THE DEFAULT VALUE
C     INDICATED BELOW.
```

```
C
C     LINE   COLUMNS   FORMAT   DEFAULT   DEFINITION
C     ====   =======   ======   =======   ==========
C      1     1- 5      I5        1        MINIMUM Z
C            6-10      I5       100       MAXIMUM Z
C           11-15      I5        73       MINIMUM C  (CROSS SECTION USED TO
C           16-20      I5        73       MAXIMUM C   CALCULATE ANOMALOUS)
C                                         C = 70 - TOTAL
C                                           = 71 - COHERENT
C                                           = 72 - INCOHERENT
C                                           = 73 - PHOTOELECTRIC
C                                           = 74 - PAIR PRODUCTION
C                                           = 75 - TRIPLET PRODUCTION
C                                           = 93 - ANOMALOUS SCATTERING
C                                                  FACTORS
C           21-25      I5        1        MINIMUM SUBSHELL DESIGNATOR
C           26-30      I5       100       MAXIMUM SUBSHELL DESIGNATOR
C                                         SEE, SUBROUTINE SUBHOL FOR A
C                                         DEFINITION OF ALL SUBSHELL
C                                         DESIGNATORS.
C           31-35      I5        0        USE SCOFIELD EDGE ENERGIES TO
C                                         DEFINE SUBSHELL EDGE ENERGY OR
C                                         LOWEST EDGE ENERGY FOR TOTAL
C                                         PHOTOELECTRIC
C                                         = 0 - NO, USE DATA AS READ
C                                         = 1 - YES, MOVE SUBSHELL EDGE
C                                               TO SCOFIELD ENERGY OR
C                                               EXTEND TOTAL PHOTOELECTRIC
C                                               DOWN TO LOWEST EDGE.
C           36-40      I5        0        CONSERVE, NORMALIZE OR SHIFT
C                                         = 0 - CONSERVE R A(E,1)
C                                         = 1 - NORMALIZE TO -Z
C                                         = 2 - SHIFT TO -Z
C           41-45      I5        0        EXTEND CROSS SECTION CALCULATION
C                                         TO UPPER ENERGY LIMIT OF TABULATED
C                                         PHOTOELECTRIC CROSS SECTIONS
C                                         = 0 - NO
C                                         = 1 - YES
C      2     1-10      E10.4    10 EV     MINIMUM ENERGY
C           11-20      E10.4    1 MEV     MAXIMUM ENERGY
C           21-30      E10.4    0.001     EDGE WIDTH
C           31-40      E10.4    0.01      ALLOWABLE UNCERTAINTY
C                                         (TO LINEARIZE CROSS SECTION AND
C                                          PRODUCE LINEARLY INTERPOLABLE
C                                          TABULATED RESULTS)
C           41-50      E10.4    0.0       HIGH ENERGY CUTOFF CRITERIA
C           51-60      E10.4    0.0       ANISOTROPY OF SCATTERING FACTORS
C                                         (POWER OF THE NORMALIZED FORM
C                                          FACTOR)
C
C     THE THIRD INPUT LINE DEFINES WHAT TO CALCULATE AND OUTPUT
C
```

```
C        3      1- 5     I5        0      REPORT ON LINE OR TO FILE
C                                         = 0 - ON LINE
C                                         = 1 - TO FILE
C               6-10     I5        0      SUBSHELL CROSS SECTIONS
C              11-15     I5        0      F'' SUBSHELL
C              16-20     I5        0      REAL SUBSHELL AMPLITUDE
C              21-25     I5        0      F' SUBSHELL
C              26-30     I5        0      (NO LONGER USED)
C              31-35     I5        0      F'' TOTAL (FOR EACH Z)
C              36-40     I5        0      REAL TOTAL AMPLITUDE
C              41-45     I5        0      F' TOTAL (FOR EACH Z)
C              46-50     I5        0      COHERENT CROSS SECTION
C              51-55     I5        0      COHERENT (FORM FACTOR = Z)
C              56-60     I5        0      COHERENT (NO ANOMALOUS)
C              61-65     I5        0      FORM FACTOR MAX/MIN LIMITS
C              66-70     I5        0      F' HIGH ENERGY LIMIT VS. Z
C              71-75     I5        0      PRINT EACH ENERGY POINT
C
C                                         FOR OUTPUT,
C                                         0 = NONE
C                                         1 = ENDL FORMAT
C                                         2 = PLOTTAB FORMAT
C                                         3 = ENDL AND PLOTTAB FORMAT
C
C     IF YOU DO NOT SPECIFY ANY OUTPUT THE PROGRAM WILL NOT PERFORM ANY
C     CALCULATIONS AND WILL PROCEED TO READ THE NEXT 3 INPUT LINES, FOR
C     THE NEXT REQUESTED CALCULATION.
C
C     EXAMPLE INPUT
C     =============
C     CALCULATE AND OUTPUT REAL AND IMAGINARY SCATTERING FACTORS
C     FROM EACH Z = 1 TO 100 FROM PHOTOELECTRIC SUBSHELL CROSS SECTIONS.
C     IF THE EDGE OF ANY SUBSHELL IS BELOW 10 EV USE SCOFIELD'S EDGES
C     TO EXTEND THE CROSS SECTIONS BELOW 10 EV DOWN TO THE EDGE.
C     NORMALIZE RESULTS TO -Z TO OBTAIN THE CORRECT LOW AND HIGH ENERGY
C     LIMITS. RESULTS WILL BE CALCULATED FOR THE ENERGY RANGE 10 EV UP
C     TO 1 MEV. 1 PER-CENT ENERGY RESOLUTION WILL BE USED. CROSS
C     SECTIONS WILL BE LINEARIZED AND TABULATED RESULTS PRODUCED TO
C     WITHIN 1 PER-CENT. THE APPROACH TO THE HIGH ENERGY ASYMPTOTIC
C     FORM WILL START AFTER F'(E) IS LESS THEN 0.1 PER-CENT OF ITS
C     MAXIMUM NEGATIVE VALUE. OUTPUT ONLY F'(E) AND F''(E) FOR EACH Z.
C     THE OUTPUT REPORT WILL BE WRITTEN TO A FILE (INSTEAD OF ON LINE).
C     THE FOLLOWING 3 INPUT LINES ARE REQUIRED,
C
C     1  100    73    73    1  100     1     1     0
C  1.0000-05 1.0000+00 1.0000-02 1.0000-02 1.0000-03 0.0000+00
C     1    0    0    0    0    0    1    0    1    0    0    0    0
C
C     NOTE - SINCE COHERENT CROSS SECTIONS WILL NOT BE CALCULATED IT
C     IS NOT NECESSARY TO SPECIFY ANISOTROPY - ON THE ABOVE LINES FOR
C     COMPLETENESS OF ALL INPUT FIELDS, THIS FIELD HAS BEEN DEFINED TO
C     BE ZERO.
```

```
C
C       THESE 3 LINES MAY BE FOLLOWED BY ADDITIONAL SETS OF 3 LINES
C       SPECIFYING THE NEXT CALCULATION. CALCULATIONS WILL CONTINUE UNTIL
C       ALL SETS OF INPUT REQUESTS HAVE BEEN PROCESSED (I.E., UNTIL AN
C       END OF FILE IS READ ON THE INPUT PARAMETER FILE).
C
C       I/O FILE DEFINITIONS
C
======================================================================
C       UNIT    FILENAME    DEFINITION
C       ====    ========    ==========
C        5      INSCAT      INPUT PARAMETERS
C        6      REPORT      OUTPUT REPORT
C       10      PHOTO       PHOTON CROSS SECTIONS (ENDL FORMAT)
C       11      PLOTTAB     PLOTTAB FORMATTED OUTPUT
C       14      ATOMIN      SUBSHELL PARAMETERS
C       15      FFACTORS    FORM FACTORS (ENDL FORMAT)
C       16      ENDL        ENDL FORMATTED OUTPUT
C
C***** COMPUTER DEPENDENT CODING ***************************************
C
C       THIS PROGRAM HAS BEEN DESIGNED TO WORK ON VIRTUALLY ANY COMPUTER.
C       IN ORDER TO ACCOMPLISH THIS THE PROGRAM CONTAINED REDUNTANT
C       CODING WHICH CAN BE AUOTMATICALLY TURNED ON OR OFF IN ORDER TO
C       IMPLEMENT THIS CODE ON ANY GIVEN COMPUTER. THE COMPUTER DEPENDENT
C       CODING IN THIS PROGRAM INCLUDES,
C
C       (1) CHARACTER/INTEGER
C           TREAT CHARACTER ARRAYS EITHER AS CHARACTER (FORTRAN-77
C           CONVENTION) OR INTEGER (FORTRAN-H CONVENTION).
C
C       (2) DOUBLE/SINGLE
C           TREAT ENERGY DEPENDENT VARIABLES IN SINGLE OR DOUBLE
C           PRECISION. ON LONG WORD LENGTH COMPUTERS (E.G., CRAY)
C           THIS PROGRAM CAN BE OPTIMIZED BY USING SINGLE PRECISION.
C           ON SHORT WORD LENGTH COMPUTERS (E.G., IBM) DOUBLE
C           PRECISION MUST BE USED.
C
C       (3) NAMES
C           OPTION TO USE STANDARD FILE NAMES
C
C       (4) STOP/EXIT
C           USE EITHER STOP OR CALL EXIT TO TERMINATE EXECUTION
C
C       (5) SAVE
C           FORTRAN OPTION TO SAVE ALL VARIABLES BETWEEN SUBROUTINE
C           CALLS - REQUIRED WITH SOME NEWER FORTRAN COMPILERS.
C
C       PROGRAM CONVERT MAY BE USED TO AUTOMATICALLY CONVERT THIS PROGRAM
C       BACK AND FORTH USING ANY COMBINATION OF THESE COMPUTER DEPENDENT
C       OPTIONS TO BEST MEET YOUR NEEDS (FOR DETAILS OF PROGRAM CONVERT
C       CONTACT D. E. CULLEN AT THE ABOVE ADDRESS).
```

```
C
C***** COMPUTER DEPENDENT CODING *************************************
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
C***** CHARACTER *****
      CHARACTER*1 KSIGN
      CHARACTER*4 ZBCD,SHBCD,TYPES,ANSWER,YES,NO,BLANK,MYPATH
C***** CHARACTER *****
C***** INTEGER *****
C     INTEGER ZBCD,SHBCD,TYPES,ANSWER,YES,NO,BLANK,MYPATH
C***** INTEGER *****
      COMMON/PHOTO1/X(10000),XT(10000),Y(10000),YT(10000),IPT,NPT
      COMMON/SHELL/XFS(10000),F1S(10000),F2S(10000),NSCATS
      COMMON/SCAT/XF(10000),F1(10000),F2(10000),NSCAT
      COMMON/NORMAL/XNORMR,XNORMI,F1SNRM,F2SNRM
      COMMON/ZID/ZBCD
      COMMON/ATOM/IZLAST,ICNOW,ISUB,NSUB

COMMON/CONJOB/R0,FOURPI,TOMSON,TOM38,XHIGH,EDGES,DEDGE,DEDGE2,UP,
     1 DOWN,DEDG10
      COMMON/OUTER/IMOUT(13)
      COMMON/HIGHE/ELIMIT(100),TOTHI,F1SUB,F1TOT,F1SLIM,RATN
      COMMON/OKERR/ERROK,ERRFF
      COMMON/ZRANGE/IZMIN,IZMAX
      COMMON/ERANGE/XMIN,XMAX
      COMMON/NORMZF/XNORM(6),KEXP(6)
      COMMON/NORMZC/KSIGN(6)
      COMMON/WATCHR/IWATCH
      COMMON/POWERS/POWER
      COMMON/XCTEND/XETEND,IXTEND,MYANOM,IMANOM(3,2)
      DIMENSION SHBCD(4),TYPES(4,7),ANSWER(14),MYPATH(2,2),XIOUT(4)
C
C     DEFINE CONSTANTS.
C
      DATA YES/'YES '/
      DATA NO/'NO  '/
      DATA BLANK/'    '/
      DATA MYPATH/
     1 'ON L','INE ',
     2 'TO F','ILE '/
      DATA TYPES/
     1 'Tota','l   ','    ','    ',
     1 'Cohe','rent','    ','    ',
     2 'Inco','here','nt  ','    ',
     3 'Phot','oele','ctri','c   ',
     4 'Pair','    ','    ','    ',
     5 'Trip','let ','    ','    ',
     6 'Anom','alou','s   ','    '/
```

```
        DATA IMOPEN/0/
C***** DOUBLE *****
        DATA ZERO/0.0D+00/
        DATA ONE/1.0D+00/
        DATA TWO/2.0D+00/
        DATA THREE/3.0D+00/
        DATA FOUR/4.0D+00/
        DATA EIGHT/8.0D+00/
        DATA TEN/1.0D+01/
        DATA FIFTY/5.0D+01/
        DATA EV10/1.0D-05/
        DATA GEV100/1.0D+05/
        DATA HBAR/6.58217D+00/
        DATA SPEEDC/2.99792D+00/
        DATA R0X/0.282D+00/
        DATA TOMX/0.6652448D+00/
        DATA TEDGE1/1.0D+01/
        DATA TEDGE2/5.0D+02/
        DATA ERRMIN/1.0D-02/
        DATA EDGMIN/1.0D-03/
        DATA XEND1/5.0D-01/
        DATA XEND2/1.0D+00/
        DATA XEND3/1.0D+01/
        DATA DXEND/1.0D-01/
        DATA ATWT/0.0D+00/
        DATA ELEVL/0.0D+00/
        DATA HAFLYF/0.0D+00/
        DATA TMP/0.0D+00/
        DATA QVALUE/0.0D+00/
        DATA XIOUT/0.0D+00,0.0D+00,0.0D+00,0.0D+00/
        QACOS(Z)=DACOS(Z)
        QABS(Z)=DABS(Z)
        QEXP(Z)=DEXP(Z)
        QLOG(Z)=DLOG(Z)
C***** DOUBLE *****
C***** SINGLE *****
C       DATA ZERO/0.0E+00/
C       DATA ONE/1.0E+00/
C       DATA TWO/2.0E+00/
C       DATA THREE/3.0E+00/
C       DATA FOUR/4.0E+00/
C       DATA EIGHT/8.0E+00/
C       DATA TEN/1.0E+01/
C       DATA FIFTY/5.0E+01/
C       DATA EV10/1.0E-05/
C       DATA GEV100/1.0E+05/
C       DATA HBAR/6.58217E+00/
C       DATA SPEEDC/2.99792E+00/
C       DATA R0X/0.282E+00/
C       DATA TOMX/0.6652448E+00/
C       DATA TEDGE1/5.0D+01/
C       DATA TEDGE2/1.0D+03/
```

```
C       DATA ERRMIN/1.0E-02/
C       DATA EDGMIN/1.0E-03/
C       DATA XEND1/5.0E-01/
C       DATA XEND2/1.0E+00/
C       DATA XEND3/1.0E+01/
C       DATA DXEND/1.0E-01/
C       DATA ATWT/0.0E+00/
C       DATA ELEVL/0.0E+00/
C       DATA HAFLYF/0.0E+00/
C       DATA TMP/0.0E+00/
C       DATA QVALUE/0.0E+00/
C       DATA XIOUT/0.0E+00,0.0E+00,0.0E+00,0.0E+00/
C       QACOS(Z)=ACOS(Z)
C       QABS(Z)=ABS(Z)
C       QEXP(Z)=EXP(Z)
C       QLOG(Z)=ALOG(Z)
C***** SINGLE *****
C
C       OPEN FILES
C
C***** NAMES *****
        OPEN(5,FILE='INSCAT')
        OPEN(10,FILE='ENDLIN')
        OPEN(11,FILE='PLOTTAB')
        OPEN(14,FILE='ATOMIN')
        OPEN(15,FILE='FFACTORS')
        OPEN(16,FILE='ENDLOUT')
C***** NAMES *****
C-----SKIP TITLES IN SUBSHELL PARAMETER FILE AND READ FIRST LINE TO
C-----INITIALIZE.
        READ(14,1400) ZBCD
        READ(14,1400) ZBCD
        READ(14,1410) IZATOM,ISATOM,EDGEIN,ELECTR
C
C       DEFINE CONSTANTS AND INITIALIZE VARIABLES.
C
        PI=QACOS(-ONE)
        FOURPI=FOUR*PI
        R0=R0X
        TOMSON=TOMX
        TOM38=THREE*TOMSON/EIGHT
        XNORMI=ONE/(FOUR*PI*HBAR*SPEEDC)
        XNORMR=XNORMI/PI
        IZLAST=0
        NSCAT=0
        TOTHI=ZERO
C
C       READ RUN PARAMETERS AND IF NECESSARY DEFINE STANDARD OPTIONS.
C
   10 IGOOF=0
        READ(5,5000,END=690,ERR=690) IZMIN,IZMAX,ICMIN,ICMAX,ISHMIN,
       1 ISHMAX,MYEDGE,IMNORM,IXTEND
```

```
      READ(5,5005,END=690,ERR=690) XMIN,XMAX,DEDGE,ERROK,ERRCUT,POWER
      READ(5,5010,END=690,ERR=690) MYWAY,IMOUT,IWATCH
      IF(IZMIN.LE.0) IZMIN=1
      IF(IZMAX.LE.0) IZMAX=100
      IF(ICMIN.LE.0) ICMIN=73
      IF(ICMAX.LE.0) ICMAX=73
      IF(ISHMIN.LT.0) ISHMIN=0
      IF(ISHMAX.LE.0) ISHMAX=100
      IF(MYEDGE.NE.0) MYEDGE=1
      IF(IMNORM.LT.0) IMNORM=0
      IF(IMNORM.GT.2) IMNORM=2
      IF(IXTEND.NE.0) IXTEND=1
      IMLOW=0
      IF(XMIN.LE.ZERO) IMLOW=1
      IF(XMAX.LE.ZERO) XMAX=ONE
      IF(DEDGE.LE.ZERO) DEDGE=EDGMIN
      DEDGE2=DEDGE/TWO
      DEDG10=DEDGE/TEN
      UP=ONE+DEDGE2
      DOWN=ONE-DEDGE2
      IF(ERROK.LE.ZERO) ERROK=ERRMIN
      ERRFF=0.1*ERROK
      IF(ERRCUT.LE.ZERO) ERRCUT=ZERO
      IF(POWER.LE.ZERO) POWER=ZERO
      IF(MYWAY.NE.0) MYWAY=1
C-----TURN OFF TOTAL CROSS SECTION OUTPUT. NO LONGER USED.
      IMOUT(5)=0
      DO 20 I=1,13
      IF(IMOUT(I).NE.0) IMOUT(I)=1
      ANSWER(I)=NO
      IF(IMOUT(I).NE.0) ANSWER(I)=YES
   20 CONTINUE
      IF(IWATCH.NE.0) IWATCH=1
      ANSWER(14)=NO
      IF(IWATCH.NE.0) ANSWER(14)=YES
C
C     CHECK TO SEE IF ANY ANOMALOUS DEPENDENT OUTPUT
C
      MYANOM=0
      DO 25 I=1,3
      IMANOM(I,1)=0
   25 IMANOM(I,2)=0
      DO 30 I=2,10
      IF(IMOUT(I).NE.0) MYANOM=1
   30 CONTINUE
      IF(IMOUT(13).NE.0) MYANOM=1
C
C     IF NO ANOMALOUS OUTPUT SET UP STANDARD ENERGY GRID TO CALCULATE
C     COHERENT CROSS SECTIONS WITHOUT ANOMALOUS.
C
      IF(MYANOM.GT.0) GO TO 60
      DX=QEXP(QLOG(TEN)/TEN)
```

```
       XF(1)=EV10
       F1(1)=ZERO
       F2(1)=ZERO
       DO 40 NSCAT=2,10000
       XF(NSCAT)=DX*XF(NSCAT-1)
       F1(NSCAT)=ZERO
       F2(NSCAT)=ZERO
       IF(XF(NSCAT).GE.GEV100) GO TO 50
   40 CONTINUE
       NSCAT=10000
   50 XF(NSCAT)=GEV100
C
C      IF REQUESTED OPEN OUTPUT REPORT FILE (ONLY ONCE).
C
   60 IF(MYWAY.EQ.0.OR.IMOPEN.NE.0) GO TO 70
C***** NAMES *****
       OPEN(6,FILE='REPORT')
C***** NAMES *****
       IMOPEN=1
C
C      IDENTIFY PROGRAM AND PRINT ALL INPUT PARAMETERS.
C
   70 PCERR=100.0*ERROK
       PCCUT=100.0*ERRCUT
       WRITE(6,6020) IZMIN,IZMAX,ICMIN,ICMAX,ISHMIN,ISHMAX
       IF(MYEDGE.EQ.0) WRITE(6,6021)
       IF(MYEDGE.NE.0) WRITE(6,6022)
       IF(IMNORM.EQ.0) WRITE(6,6023)
       IF(IMNORM.EQ.1) WRITE(6,6024)
       IF(IMNORM.EQ.2) WRITE(6,6025)
       IF(IXTEND.EQ.0) WRITE(6,6027)
       IF(IXTEND.NE.0) WRITE(6,6028)
       CALL NORMX(XMIN,XNORM(1),KSIGN(1),KEXP(1))
       CALL NORMX(XMAX,XNORM(2),KSIGN(2),KEXP(2))
       CALL NORMX(DEDGE,XNORM(3),KSIGN(3),KEXP(3))
       CALL NORMX(POWER,XNORM(4),KSIGN(4),KEXP(4))
       WRITE(6,6026) (XNORM(I),KSIGN(I),KEXP(I),I=1,3),PCERR,PCCUT,
      1 XNORM(4),KSIGN(4),KEXP(4)
       WRITE(6,6080) MYPATH(1,MYWAY+1),MYPATH(2,MYWAY+1),
      1 (ANSWER(I),I=1,8)
       WRITE(6,6090) (ANSWER(I),I=9,14)
C
C      IF NO OUTPUT REQUEST, DO NOT PERFORM ANY CALCULATIONS - PROCEED
C      TO READ NEXT SET OF INPUT PARAMETERS.
C
       DO 77 I=1,13
       IF(IMOUT(I).NE.0) GO TO 79
   77 CONTINUE
       WRITE(6,6092)
       GO TO 10
C
C      IF NO ANOMALOUS DEPENDENT OUTPUT, PROCESS ALL REQUESTED Z VALUES.
```

```
C
   79 WRITE(6,6095)
      IF(MYANOM.NE.0) GO TO 90
      DO 80 IZLAST=IZMIN,IZMAX
      CALL ZHOL(IZLAST,ZBCD)
      WRITE(6,6010) IZLAST,ZBCD
   80 CALL FINALE
      IZLAST=IZMAX+1
      GO TO 10
C
C     IF NECESSARY REWIND FILES.
C
   90 IF(IZLAST.LT.IZMIN) GO TO 100
      IZLAST=0
      REWIND 10
      REWIND 14
      REWIND 15
C-----SKIP TITLES IN SUBSHELL PARAMETER FILE AND READ FIRST LINE TO
C-----INITIALIZE.
      READ(14,1400) ZBCD
      READ(14,1400) ZBCD
      READ(14,1410) IZATOM,ISATOM,EDGEIN,ELECTR
C
C     IF IZLAST IS GREATER THAN ZERO FIRST LINE OF NEXT SECTION
C     IS ALREADY IN CORE. RESET LAST Z TO INDICATE NO RESULTS IN
C     CORE.
C
  100 IF(IZLAST.LE.0) GO TO 110
      IZLAST=0
      GO TO 120
C
C     READ NEXT SECTION OF PHOTO SUBSHELL DATA.
C
  110 READ(10,1000,END=640,ERR=640) IZ
C-----INITIALIZE POINT COUNT.
  120 NSCATS=0
C
C     OUTPUT RESULTS AT THE END OF EACH Z (MUST ASSUME CURRENT DATA IS
C     SUBSHELL OF SAME Z UNTIL DATA FOR A NEW Z IS FOUND).
C
      IF(IZ.NE.IZLAST) CALL FINALE
      IZLAST=IZ
C
C     END OF RUN IF Z IS GREATER THAN MAXIMUM REQUESTED.
C
      IF(IZ.GT.IZMAX) GO TO 670
C
C     SKIP IF Z IS LESS THAN MINIMUM REQUESTED
C
      IF(IZ.GE.IZMIN) GO TO 140
  130 READ(10,1025,END=640,ERR=640) IFLAG
      IF(IFLAG) 130,130,110
```

```
C
C      READ NEXT SET OF POINTS. SKIP IF NOT IN REQUESTED C AND SUBSHELL
C      RANGE.
C
  140 READ(10,1010,END=640,ERR=640) ICNOW,IINOW,XI,XI2,XI3
      ISUB=XI
C-----SKIP IF NOT IN REQUESTED C AND SUBSHELL INDEX RANGE.
      IF(ICNOW.LT.ICMIN.OR.ICNOW.GT.ICMAX) GO TO 130
      IF(ISUB.LT.ISHMIN.OR.ISUB.GT.ISHMAX) GO TO 130
C
C      CHECK FOR CROSS SECTIONS OR ANOMALOUS SCATTERING FACTORS
C
C-----CROSS SECTION.
      MYTYPE=0
      IF(IINOW.EQ.0) MYTYPE=-1
      IF(IINOW.EQ.943) MYTYPE=1
      IF(IINOW.EQ.944) MYTYPE=2
      IF(MYTYPE.EQ.0) GO TO 130
C
C      DATA HAS BEEN REQUESTED. READ DATA POINTS.
C
      DO 160 IPT=1,10000
  150 READ(10,1020) X(IPT),Y(IPT),IFLAG
      IF(IFLAG.NE.0) GO TO 170
C-----ENERGY POINTS MUST BE IN ASCENDING ORDER
      IF(IPT.EQ.1) GO TO 160
      IF(X(IPT).GE.X(IPT-1)) GO TO 160
      WRITE(6,6050) IPT,X(IPT-1),X(IPT)
      GO TO 150
  160 CONTINUE
C-----TOO MANY POINTS.
      GO TO 660
C
C      DEFINE OUTPUT I.D. INFORMATION.
C
  170 IPT=IPT-1
C-----DEFINE ELEMENT.
      CALL ZHOL(IZ,ZBCD)
C-----DEFINE EITHER REACTION OR SUBSHELL.
      IF(ISUB.NE.0) GO TO 174
      INDEX=ICNOW-69
      IF(INDEX.LT.1.OR.INDEX.GT.6) INDEX=7
      DO 172 I=1,4
  172 SHBCD(I)=TYPES(I,INDEX)
      GO TO 176
  174 CALL SUBHOL(ISUB,SHBCD)
      SHBCD(4)=BLANK
C
C      IF CROSS SECTIONS PROCESS THEM. IF ANOMALOUS READ BOTH IMAGINARY
C      AND REAL.
C
  176 IF(MYTYPE.LT.0) GO TO 200
```

```
      IMANOM(1,MTYPE)=IZ
      IMANOM(2,MTYPE)=ISUB
      IMANOM(3,MTYPE)=IPT
C-----SAVE SCATTERING FACTORS.
      DO 177 I=1,IPT
      XFS(I)=X(I)
      IF(MYTYPE.EQ.2) F1S(I)=Y(I)
      IF(MYTYPE.EQ.1) F2S(I)=Y(I)
  177 CONTINUE
      IF(MYTYPE.EQ.1) GO TO 110
C-----INSURE SAME Z, SUBSHELL, POINT COUNT FOR REAL AND IMAGINARY
      DO 178 I=1,3
      IF(IMANOM(I,1).NE.IMANOM(I,2)) GO TO 182
  178 CONTINUE
      GO TO 192
  182 DO 188 I=1,3
      IMANOM(I,1)=0
  188 IMANOM(I,2)=0
      GO TO 110
C-----BOTH SCATTERING FACTORS READ. DEFINE POINT COUNT AND SET UPPER
C-----ENERGY FOR CROSS SECTION EXTENSION TO 100 GEV.
  192 NSCATS=IPT
      XETEND=GEV100
C
C     DEFINE EDGE ENERGY, TOTAL KINETIC ENERGY AND NUMBER OF
C     ELECTRONS PER SUBSHELL.
C
C-----INITIALIZE LOWEST EDGE ENERGY TO FIRST TABULATED DATA POINT.
  200 EDGLOW=X(1)
C-----IF DATA IS NOT PHOTOELECTRIC OR SCATTERING FACTORS IGNORE EDGES.
      IF(ICNOW.NE.73.AND.ICNOW.NE.93) GO TO 260
C-----SELECT LOWEST EDGE ENERGY IN ORDER TO EXTEND CROSS SECTIONS BELOW
C-----10 EV.
      IF(MYEDGE.NE.0) EDGLOW=ZERO
      GO TO 220
  210 READ(14,1410) IZATOM,ISATOM,EDGEIN,ELECTR
C-----FIND Z.
  220 IF(IZATOM-IZ) 210,230,250
C-----Z FOUND. KEEP TRACK OF LOWEST EDGE ENERGY.
  230 IF(MYEDGE.LE.0) GO TO 240
      IF(EDGLOW.EQ.ZERO) EDGLOW=EDGEIN
      IF(EDGEIN.LT.EDGLOW) EDGLOW=EDGEIN
  240 IF(ISUB.LE.0) GO TO 210
C-----SELECT SUBSHELL.
      IF(ISATOM-ISUB) 210,270,650
C-----ABOVE Z. O.K. IF TOTAL. ERROR IF SUBSHELL.
  250 IF(ISUB) 260,260,650
C
C     NOT USING SUBSHELL DATA. DEFINE EDGE ENERGY AS ENERGY OF
C     LOWEST EDGE AND ELECTRONS AS Z.
C
  260 EDGES=EDGLOW
```

```
      ELECTR=IZ
      GO TO 271
C
C     USING SUBSHELL DATA. DEFINE EDGE ENERGY.
C
  270 EDGES=EDGEIN
C-----IF NOT USING SCOFIELD EDGES USE FIRST DATA POINT.
      IF(MYEDGE.EQ.0) EDGES=X(1)
C
C     IS DATA CROSS SECTIONS OR SCATTERING FACTORS
C
  271 IF(MYTYPE.LT.0) GO TO 280
C
C     SCATTERING FACTORS...CONVERT REAL SCATTERING FACTOR TO REAL
C     AMPLITUDE OVER R0.
C
      DO 272 I=1,NSCATS
  272 F1S(I)=-(ELECTR+F1S(I))
      WRITE(6,6011) IZ,ZBCD,SHBCD,ELECTR,NSCATS
      GO TO 630
C
C     INSURE CROSS SECTIONS START AT EDGE ENERGY
C
C-----SAVE HIGHEST ENERGY
  280 XETEND=X(IPT)
C-----IF LOWER ENERGY LIMIT IS NOT DEFINED DEFINE IT NOW.
      IF(IMLOW.NE.0) XMIN=EDGES/TEN
      CALL EDGEIT
C
C     DEFINE LINEARLY INTERPOLABLE CROSS SECTIONS VS. ENERGY.
C
      CALL LINEAR(X,Y,IPT,XT,YT,NPT,ERROK)
C
C     SUBSHELL CROSS SECTION...PLOTTAB OUTPUT.
C
      IF(IMOUT(1).LE.0) GO TO 300
      WRITE(11,1100) IZ,ZBCD,SHBCD
      CALL PLTTAB(11,XT,YT,NPT)
C-----NOTHING ELSE TO DO IF ONLY CROSS SECTION OUTPUT.
      DO 290 I=2,13
      IF(IMOUT(I).GT.0) GO TO 300
  290 CONTINUE
      GO TO 110
C
C     INTRODUCE RESOLUTION.
C
  300 CALL FUZZY(XT,YT,NPT,XFS,F1S,NPTF,DEDGE,ERROK,XMAX)
C
C     COPY BACK TO ACTIVE ARRAYS. HAVE CROSS SECTION SMOOTHLY APPROACH
C     ZERO BY 10 MEV.
C
C     COPY UP TO 1/2 MEV.
```

```
C
      DO 320 I=1,NPTF
      IF(XFS(I)-XEND1) 310,340,330
  310 XT(I)=XFS(I)
  320 YT(I)=F1S(I)
C-----CROSS SECTIONS STOP BELOW 1/2 MEV.
      NPT=NPTF
      GO TO 379
C-----POINT IS ABOVE 1/2 MEV. INTERPOLATE TO 1/2 MEV.
  330 WT1=(XEND1-XFS(I-1))/(XFS(I)-XFS(I-1))
      WT2=ONE-WT1
      XFS(I)=XEND1
      F1S(I)=WT1*F1S(I)+WT2*F1S(I-1)
C-----DEFINE POINT AT 1/2 MEV.
  340 XT(I)=XFS(I)
      YT(I)=F1S(I)
C
C     SMMOTHLY CHANGE SLOPE FROM VALUE AT 1/2 MEV TO -3 AT 1 MEV
C     (I.E., SMOOTHLY CHANGE VARIATION FROM 1/2 MEV FORM TO 1/E**3)
C
C-----DEFINE LOCAL SLOPE.
      ALPHA1=QLOG(YT(I)/YT(I-1))/QLOG(XT(I-1)/XT(I))
  350 YEND=YT(I)
      ALPHA1=ALPHA1/QLOG(XEND2/XEND1)
      ALPHA2=THREE/QLOG(XEND2/XEND1)
      K=I+1
C-----USE 50 POINTS PER ENERGY DECADE.
      DXEND=QEXP(QLOG(TEN)/FIFTY)
      DO 360 I=K,10000
      XT(I)=XT(I-1)*DXEND
      IF(XT(I).GE.XEND2) GO TO 370
      RAT=XEND1/XT(I)
      ALPHA=ALPHA1*QLOG(XEND2/XT(I))+ALPHA2*QLOG(XT(I)/XEND1)
      YT(I)=YEND*(RAT**ALPHA)
  360 CONTINUE
      STOP
  370 XT(I)=XEND2
      RAT=XEND1/XT(I)
      YT(I)=YEND*(RAT**THREE)
C
C     DEFINE EXTENSION TO ZERO AT 10 MEV...THE EXTENSION IS 1/E**3
C     TIMES LINEAR FROM THE 1 MEV VALUE TO ZERO AT 10 MEV...THIS WILL
C     FORCE THE IMAGINARY SCATTERING FACTOR TO APPROACH ZERO AS 1/E**2
C     TIMES LINEAR.
C
      K=I+1
      YEND=YT(I)*(XT(I)**3)/QLOG(XEND3/XEND2)
      DO 376 I=K,10000
      XT(I)=XT(I-1)*DXEND
      IF(XT(I).GE.XEND3) GO TO 378
  376 YT(I)=YEND*QLOG(XEND3/XT(I))/(XT(I)**3)
      STOP
```

```
   378 XT(I)=XEND3
       YT(I)=ZERO
       NPT=I
C
C      DEFINE INFINITY LIMIT BY INTEGRATING CROSS SECTIONS.
C
   379 CALL HIGHW(SUBHI)
       TOTHI=TOTHI+SUBHI
C
C      PERFORM DISPERSION INTEGRAL CALCULATION
C
       IF(IWATCH.NE.0) WRITE(6,6040)
C
C      DEFINE LIMITING VALUE AT HIGH ENERGY AS WELL AS WHERE TO START
C      CHECKING FOR CONVERGENCE TOWARD LIMITING VALUE (CAN ONLY CHECK
C      FOR CONVERGENCE IF USING SUBSHELL DATA).
C
C-----DEFINE LIMIT AT THIS ENERGY.
       RATN=ONE
       F1SLIM=ZERO
       F1SNRM=XNORMR/R0
       CALL DEPOSE(XMAX,F1SUB)
       F1SUB=-F1SUB
C
C      DEFINE SUBSHELL AND ELEMENT ACTUAL LIMITS.
C
       F1TOT=F1TOT+F1SUB
C
C      DEFINE SHIFT AND NORMALIZATION.
C      0 = AS CALCULATED
C      1 = RE-NORMALIZE TO ELECTRON
C      2 = SHIFT TO ELECTRONS
C
       IF(IMNORM-1) 380,390,400
C
C      F1SLIM = SHIFT IN CALCULATED R A(E,1) TO DEFINE SHIFTED R A(E,1)
C               IN THE FORM,
C               SHIFTED R A(E,1) = CALCULATED R A(E,1) + ASHIFT
C             =       0          = CONSERVE R A(E,1)
C             =       0          = NORMALIZE TO -Z
C             = -(ELECTR-F1SUB) = SHIFT TO -Z
C      THE COMBINATION OF THE CALCULATED R A(E,1) AND THE SHIFT, F1SLIM,
C      IS TO MAKE THE HIGH ENERGY LIMIT OF THE SHIFTED R A(E,1),
C      AEND   = -F1SUB           = CONSERVE R A(E,1)
C             = -ELECTR          = NORMALIZE TO -Z
C             = -ELECTR          = SHIFT TO -Z
C      AND THE LOW ENERGY LIMIT OF THE SHIFTED R A(E,1) IS F1SLIM
C
C      F'(E) CAN THEN IN ALL CASES BE DEFINED FROM THE SHIFTED R A(E,1)
C      FOR CONSISTENT LATER USE WITH FORM FACTORS USING THE DEFINITION,
C      F'(E) = -(ELECTR + SHIFTED R A(E,1))
C
```

```
C-----AS CALCULATED...USE UPPER LIMIT OF Z TO DERIVE F'(E) FROM
C-----R A(E,1). FINAL VALUE AT XMAX WILL BE (F1SUB-ELECTR).
  380 F1SLIM=ZERO
      AEND=-F1SUB
      GO TO 410
C-----MULTIPLY TO RE-NORMALIZE TO ELECTRONS...THIS ALSO CHANGES
C-----UPPER LIMIT TO -Z. FINAL VALUE AT XMAX WILL BE ZERO
  390 RATN=ELECTR/F1SUB
      F1SLIM=ZERO
      AEND=-ELECTR
      GO TO 410
C-----SHIFT TO ELECTRONS, NO RE-NORMALIZATION...USE UPPER LIMIT
C-----OF R A(INFINITY,0) TO DERIVE F'(E) FROM R A(E,1). FINAL
C-----VALUE AT XMAX WILL BE ZERO
  400 F1SLIM=-(ELECTR-F1SUB)
      AEND=-ELECTR
C
C     DEFINE NORMALIZATION FOR F1 AND F2 AND SHIFT FOR ELEMENT.
C
  410 F1SNRM=RATN*XNORMR/R0
      F2SNRM=RATN*XNORMI/R0
C
C     DEFINE PARAMETERS FOR HIGH ENERGY FIT.
C
      XCHEK1=TEDGE1*EDGES
      XCHEK2=TEDGE2*EDGES
      IF(ISUB.LE.0) XCHEK1=TEDGE1*XMAX
      IF(ISUB.LE.0) XCHEK2=TEDGE2*XMAX
      F1SMIN=ZERO
C
C     FIRST POINT.
C
C-----DEFINE F1S AT MINIMUM ENERGY.
      NSCATS=1
      XFS(NSCATS)=XMIN
      XLAST=XMIN
      CALL DEPOSE(XLAST,F1SLST)
      F1S(NSCATS)=F1SLST
C-----DEFINE F2S BY INTERPOLATING CROSS SECTION.
      DO 420 NPTMIN=1,NPT
      IF(XT(NPTMIN).GT.XMIN) GO TO 430
  420 CONTINUE
      GO TO 110
  430 IF(NPTMIN.EQ.1) GO TO 440
C-----ABOVE EDGE. INTERPOLATE.
      WT1=(XMIN-XT(NPTMIN-1))/(XT(NPTMIN)-XT(NPTMIN-1))
      WT2=ONE-WT1
      YLAST=WT1*YT(NPTMIN)+WT2*YT(NPTMIN-1)
      F2S(NSCATS)=F2SNRM*XMIN*YLAST
      GO TO 450
C-----BELOW EDGE. F2S IS ZERO.
  440 YLAST=ZERO
```

```
      F2S(NSCATS)=ZERO
  450 CALL WATCH
C
C     ALL OTHER POINTS
C
      DO 530 INOW=NPTMIN,NPT
  460 XNOW=XT(INOW)
      YNOW=YT(INOW)
C-----DO NOT ALLOW LARGE STEPS IN ENERGY.
      XUSE=TWO*XLAST
      IF(XNOW.LE.XUSE) GO TO 470
      WT1=(XUSE-XLAST)/(XNOW-XLAST)
      WT2=ONE-WT1
      XNOW=XUSE
      YNOW=WT1*YNOW+WT2*YLAST
  470 LOOPX=0
C-----CHECK FOR MAXIMUM ENERGY.
      IF(XNOW.GE.XMAX) GO TO 560
C-----TOP OF INTERVAL. DEFINE AND SAVE VALUES.
      CALL DEPOSE(XNOW,F1SNOW)
      F2SNOW=F2SNRM*XNOW*YNOW
C-----SAVE MINIMUM F1S.
      IF(F1SNOW.GE.ZERO) GO TO 480
      IF(F1SNOW.LT.F1SMIN) F1SMIN=F1SNOW
C
C     CHECK FOR CONVERGENCE TO ASSYMPTOTIC FORM ABOVE CHECKING ENERGY
C     IF F1S HAS PASSED THROUGH MAXIMUM.
C
      IF(XNOW.GE.XCHEK2) GO TO 540
      IF(XNOW.LE.XCHEK1) GO TO 480
      IF(F1SNOW.LE.F1SMIN) GO TO 480
      IF(QABS(F1SNOW-AEND).LE.ERRCUT*QABS(AEND)) GO TO 540
C
C     INTERVAL HALVING TO CONVERGENCE.
C
C-----DO NOT ITERATE TOWARD ZERO. IF BOTH ENDS OF INTERVAL ARE SMALL
C-----END INTERVAL.
  480 IF(QABS(F1SNOW).LE.ERROK.AND.QABS(F1SLST).LE.ERROK) GO TO 520
C-----INTERVAL HALVING LOOP
      DO 500 LOOPI=1,10
C-----DEFINE EXACT VALUE AT MIDPOINT
      XMID=(XNOW+XLAST)/TWO
      YMID=(YNOW+YLAST)/TWO
      CALL DEPOSE(XMID,F1SMID)
      F2SMID=F2SNRM*XMID*YMID
C-----DEFINE LINEARLY INTERPOLATED VALUE AT MIDPOINT
      F1SAV=(F1SNOW+F1SLST)/TWO
C-----CHECK FOR CONVERGENCE.
      IF(QABS(F1SMID-F1SAV).LE.QABS(ERROK*F1SMID)) GO TO 520
C-----DO NOT ITERATE TOWARD ZERO (IF THE MIDPOINT AND AT LEAST ONE
C-----END OF THE INTERVAL IS SMALL THEN PROGRAM IS ITERATING TOWARD
C-----ZERO).
```

```
      IF(QABS(F1SMID).GT.ERROK) GO TO 490
      IF(QABS(F1SNOW).LE.ERROK) GO TO 510
      IF(QABS(F1SLST).LE.ERROK) GO TO 510
C-----NO CONVERGENCE. INTERVAL HALF BY REPLACING CURRENT END POINT
C-----BY MIDPOINT
  490 XNOW=XMID
      YNOW=YMID
      F1SNOW=F1SMID
      F2SNOW=F2SMID
  500 CONTINUE
C-----END OF INTERVAL HALVING LOOP
  510 LOOPI=10
C
C     END OF INTERATION. KEEP CURRENT POINT
C
  520 NSCATS=NSCATS+1
      XFS(NSCATS)=XNOW
      F1S(NSCATS)=F1SNOW
      F2S(NSCATS)=F2SNOW
      CALL WATCH
C-----SAVE END OF INTERVAL AS BEGINNING OF NEXT INTERVAL.
      XLAST=XNOW
      YLAST=YNOW
      F1SLST=F1SNOW
C-----INTERVAL IS FINISHED IF CONVERGENCE DURING FIRST PASS OF
C-----INTERVAL HALVING LOOP
      IF(XNOW.GE.XT(INOW)) GO TO 530
      LOOPX=LOOPX+LOOPI
      IF(LOOPX.LT.500) GO TO 460
C-----END OF INTERVAL.
  530 CONTINUE
C
C     HAVE CALCULATED TO THE END OF TABULATED DATA
C
      GO TO 560
C
C     CONVERGENCE. DEFINE REMAINING POINTS BY INTERPOLATING.
C
C-----DEFINE AND PRINT POINT AT END OF INTERVAL.
  540 NSCATS=NSCATS+1
      XFS(NSCATS)=XNOW
      F1S(NSCATS)=F1SNOW
      F2S(NSCATS)=F2SNOW
      CALL WATCH
      IF(IWATCH.NE.0) WRITE(6,6070)
C-----NORMALIZE TO CURRENT VALUE.
      Z=EDGES/XNOW
      RATNOW=Z*Z
      Z=EDGES/XMAX
      RATMAX=Z*Z
      F1STOP=(F1S(NSCATS)-AEND)/
     1 ((RATNOW-RATMAX)*(ONE+(RATNOW+RATMAX)/TWO))
```

```
C-----INTERPOLATE REMAINING POINTS.
      INOW=INOW+1
      IF(INOW.GT.NPT) GO TO 560
      DO 550 K=INOW,NPT
      IF(XT(K).GE.XMAX) GO TO 560
      Z=EDGES/XT(K)
      RATNOW=Z*Z
      NSCATS=NSCATS+1
      XFS(NSCATS)=XT(K)
      F1S(NSCATS)=AEND+
     1 F1STOP*(RATNOW-RATMAX)*(ONE+(RATNOW+RATMAX)/TWO)
      F2S(NSCATS)=F2SNRM*XT(K)*YT(K)
      CALL WATCH
  550 CONTINUE
C
C     SET LAST VALUES TO LIMITING VALUES
C
  560 IF(XFS(NSCATS).LT.XMAX) NSCATS=NSCATS+1
      XFS(NSCATS)=XMAX
      F1S(NSCATS)=AEND
      F2S(NSCATS)=ZERO
      CALL WATCH
C
C     THIN RESULTS (WARNING...THE FOLLOWING CALL ASSUMES THAT F1S AND
C     F2S ARE EACH OF DIMENSION 10000 AND F2S IMMEDIATELY FOLLOWS F1S
C     IN CORE.
C
      CALL THIN3L(XFS,F1S,NSCATS,10000,2,ERROK)
C
C     OUTPUT SUBSHELL RESULTS.
C
C-----IMAGINARY SCATTERING FACTOR.
      IF(IMOUT(2).LE.0) GO TO 570
C
C     ENDL OUTPUT
C
      IZA=1000*IZ
      XIOUT(1)=ISUB
      CALL OUTL1(16,IZA,7,0,ATWT,891031,2,2,ELEVL,HAFLYF,TMP)
      CALL OUTL2(16,93,943,91,QVALUE,XIOUT)
      CALL OUTAB2(16,XFS,F2S,NSCATS)
C
C     PLOTTAB OUTPUT.
C
      WRITE(11,1140) IZ,ZBCD,SHBCD
      CALL PLTTAB(11,XFS,F2S,NSCATS)
C-----REAL SCATTERING AMPLITUDE/R0
  570 IF(IMOUT(3).LE.0) GO TO 580
      WRITE(11,1130) IZ,ZBCD,SHBCD
      CALL PLTTAB(11,XFS,F1S,NSCATS)
C-----REAL SCATTERING FACTOR.
  580 IF(IMOUT(4).LE.0) GO TO 610
```

```
C
C       ENDL OUTPUT
C
        DO 590 I=1,NSCATS
  590 F1S(I)=-(ELECTR+F1S(I))
        IZA=1000*IZ
        XIOUT(1)=ISUB
        CALL OUTL1(16,IZA,7,0,ATWT,891031,2,2,ELEVL,HAFLYF,TMP)
        CALL OUTL2(16,93,944,91,QVALUE,XIOUT)
        CALL OUTAB2(16,XFS,F1S,NSCATS)
C
C       PLOTTAB OUTPUT.
C
        WRITE(11,1130) IZ,ZBCD,SHBCD
        CALL PLTTAB(11,XFS,F1S,NSCATS)
        DO 600 I=1,NSCATS
  600 F1S(I)=-(ELECTR+F1S(I))
C
C       PRINT SUBSHELL REPORT.
C
  610 CALL NORMX(EDGES,XNORM(1),KSIGN(1),KEXP(1))
        IF(ICNOW.EQ.73)
     1 WRITE(6,6010) IZ,ZBCD,SHBCD,XNORM(1),KSIGN(1),KEXP(1),ELECTR,
     2 F1SUB,SUBHI,NSCATS
        IF(ICNOW.NE.73)
     1 WRITE(6,6015) IZ,ZBCD,SHBCD,F1SUB,SUBHI,NSCATS
C-----NOTHING ELSE TO DO IF NO COMBINED OR COHERENT OUTPUT.
        DO 620 I=6,12
        IF(IMOUT(I).NE.0) GO TO 630
  620 CONTINUE
        GO TO 110
C
C       ADD SUBSHELL CONTRIBUTION TO Z TOTAL.
C
  630 CALL SUMMER
C-----THIN MERGED DATA IF TABLE IS OVER 1/2 FULL.
        IF(NSCAT.GT.5000) CALL THIN3L(XF,F1,NSCAT,10000,2,ERROK)
C-----READY FOR NEXT SUBSHELL.
        GO TO 110
C
C       END OF ONE OF THE DATA FILES.
C
C-----END OF SUBSHELL DATA IS NOT AN ERROR.
  640 IGOOF=1
        GO TO 670
C-----END OF EDGE ENERGIES.
  650 WRITE(6,6002)
        IGOOF=1
        GO TO 670
C
C       TOO MANY DATA POINTS.
C
```

```
C-----TOO MANY SUBSHELL CROSS SECTIONS.
  660 WRITE(6,6004)
      IGOOF=1
C
C     END OF RUN. PROCESS LAST Z, IF ANY.
C
  670 CALL FINALE
C
C     HIGH ENERGY LIMIT VS. Z...PLOTTAB OUTPUT.
C
      IF(IMOUT(13).LE.0) GO TO 700
      DO 680 I=IZMIN,IZMAX
      X(I)=I
  680 Y(I)=I
      IZD=IZMAX-IZMIN+1
      WRITE(11,1170)
      CALL PLTTAB(11,X(IZMIN),Y(IZMIN),IZD)
      WRITE(11,1180)
      CALL PLTTAB(11,X(IZMIN),ELIMIT(IZMIN),IZD)
      GO TO 700
C
C     END OF RUN.
C
  690 WRITE(6,6030)
      CALL EXIT
C
C     BACK TO READ MORE INPUT PARAMETERS IF NO ERRORS.
C
  700 IF(IGOOF) 10,10,690
 1000 FORMAT(I3)
C***** DOUBLE *****
 1010 FORMAT(I2,I3,16X,3D12.4)
 1020 FORMAT(2D11.4,49X,I1)
C***** DOUBLE *****
C***** SINGLE *****
C1010 FORMAT(I2,I3,16X,3E12.4)
C1020 FORMAT(2E11.4,49X,I1)
C***** SINGLE *****
 1025 FORMAT(71X,I1)
 1100 FORMAT(I4,4A4,A1,' Cross Section')
 1130 FORMAT(I4,4A4,A1,' Real')
 1140 FORMAT(I4,4A4,A1,' Imaginary')
 1170 FORMAT(' Z')
 1180 FORMAT(' High E Limit')
 1400 FORMAT(A1)
C***** DOUBLE *****
 1410 FORMAT(I5,I3,14X,D11.4,11X,D11.4)
C***** DOUBLE *****
C***** SINGLE *****
C1410 FORMAT(I5,I3,14X,E11.4,11X,E11.4)
C***** SINGLE *****
 5000 FORMAT(9I5)
```

```
C***** DOUBLE *****
 5005 FORMAT(6D10.3)
C***** DOUBLE *****
C***** SINGLE *****
C5005 FORMAT(6E10.3)
C***** SINGLE *****
 5010 FORMAT(15I5)
 6002 FORMAT(1X,72('=')/' END OF EDGE ENERGIES'/1X,72('='))
 6004 FORMAT(1X,72('=')/' TOO MANY SUBSHELL CROSS SECTIONS'/1X,72('='))
 6010 FORMAT(I4,A4,1X,3A4,A1,1X,F8.5,A1,I2,0PF9.4,0PF11.4,0PF11.4,I8)
 6011 FORMAT(I4,A4,1X,3A4,A1,12X,0PF9.4,22X,I8)
 6015 FORMAT(I4,A4,1X,3A4,A1,21X,0PF11.4,0PF11.4,I8)
 6020 FORMAT(' COHERENT SCATTERING FACTORS AND CROSS SECTIONS',
     1 ' (SCATMAN: VERSION 90-1)'/1X,72('=')/' PARAMETERS'/1X,72('=')/
     2 ' Z RANGE..............................',I12,' TO',I12/
     3 ' REACTION NUMBER RANGE...............',I12,' TO',I12/
     3 ' SUBSHELL DESIGNATOR RANGE...........',I12,' TO',I12)
 6021 FORMAT(
     1 ' LOWEST EDGE ENERGY..................','DEFINED BY DATA READ')
 6022 FORMAT(
     1 ' LOWEST EDGE ENERGY..................','LOWEST SCOFIELD EDGE')
 6023 FORMAT(
     1 ' NORMALIZATION.......................','CONSERVE REAL ',
     2 ' AMPLITUDE')
 6024 FORMAT(
     1 ' NORMALIZATION.......................','RE-NORMALIZE TO -Z',
     2 ' LIMIT')
 6025 FORMAT(
     1 ' NORMALIZATION.......................','SHIFT TO -Z LIMIT')
 6026 FORMAT(
     1 ' ENERGY RANGE........................',1X,F8.5,A1,I2,' TO',
     2 1X,F8.5,A1,I2, ' MEV'/
     3 ' WIDTH OF EDGES......................',1X,F8.5,A1,I2,
     4 ' TIMES ENERGY'/
     4 ' LINEAR CROSS SECTIONS TO WITHIN.....',0PF12.4,' %'/
     5 ' HIGH ENERGY CUTOFF CRITERIA.........',0PF12.4,' %'/
     5 ' ANISOTROPY (POWER OF FORM FACTOR)...',F8.5,A1,I2/1X,72('='))
 6027 FORMAT(
     1 ' EXTEND CROSS SECTION CALCULATION....NO')
 6028 FORMAT(
     1 ' EXTEND CROSS SECTION CALCULATION....YES')
 6030 FORMAT(1X,72('='))
 6040 FORMAT(' POINT  ENERGY-MEV  R A(E,1)/R0 I A(E,1)/R0'/
     1 1X,72('='))
 6050 FORMAT(I6,1PD12.4,1PD12.4,' ENERGY ORDER ERROR...POINT SKIPPED')
 6070 FORMAT(1X,20('*'),' LIMIT ',19('*'))
 6080 FORMAT(' OUTPUT'/1X,72('=')/
     1 ' OUTPUT..............................',2A4/
     2 ' SUBSHELL CROSS SECTION..............',A4/
     3 ' F2 SUBSHELL (IMAGINARY).............',A4/
     4 ' REAL SUBSHELL AMPLITUDE/R0..........',A4/
     5 ' F1 SUBSHELL (REAL)..................',A4/
```

```
      6 ' TOTAL PHOTOELECTRIC CROSS SECTION..',A4/
      7 ' F2 TOTAL (IMAGINARY)................',A4/
      8 ' REAL TOTAL AMPLITUDE/R0.............',A4/
      9 ' F1 TOTAL (REAL)....................',A4)
 6090 FORMAT(
      1 ' COHERENT...........................',A4/
      2 ' COHERENT (FORM FACTOR = Z).........',A4/
      3 ' COHERENT (NO ANOMALOUS)............',A4/
      4 ' LIMITS OF FORM FACTOR..............',A4/
      5 ' HIGH ENERGY LIMIT VS. Z............',A4/
      6 ' PRINT EACH ENERGY POINT............',A4)
 6092 FORMAT(1X,72('=')/
      1 ' NO OUTPUT REQUESTED - NO CALCULATIONS PERFORMED'/1X,72('='))
 6095 FORMAT(1X,72('=')/
      1 ' ELEMENT SUBSHELL OR    LOWEST E   ELECTRONS   HIGH E',
      2 '    INFINITY POINTS'/
      3 '          REACTION       EDGE (MEV)            LIMIT ',
      4 '    LIMIT'/1X,72('='))
      END
      SUBROUTINE WATCH
C
C     PRINT CURRENT F1S AND F2S
C
C     ARGUMENTS
C     =========
C     XNOW   = CURRENT ENERGY
C     YNOW   = CURRENT CROSS SECTIONS
C
C     XNOW AND YNOW ARE USED TO DEFINE F2S.
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
C***** CHARACTER *****
      CHARACTER*1 KSIGN
C***** CHARACTER *****
      COMMON/SHELL/XFS(10000),F1S(10000),F2S(10000),NSCATS
      COMMON/NORMZF/XNORM(6),KEXP(6)
      COMMON/NORMZC/KSIGN(6)
      COMMON/WATCHR/IWATCH
C-----IF REQUESTED PRINT POINT.
      IF(IWATCH.LE.0) GO TO 10
      CALL NORMX(XFS(NSCATS),XNORM(1),KSIGN(1),KEXP(1))
      CALL NORMX(F1S(NSCATS),XNORM(2),KSIGN(2),KEXP(2))
      CALL NORMX(F2S(NSCATS),XNORM(3),KSIGN(3),KEXP(3))
      WRITE(6,6000) NSCATS,(XNORM(I),KSIGN(I),KEXP(I),I=1,3)
C-----CHECK FOR MEMORY OVERFLOW.
   10 IF(NSCATS.LT.10000) GO TO 20
      WRITE(6,6010)
```

```
         CALL EXIT
   20 RETURN
 6000 FORMAT(I6,3(1X,F8.5,A1,I2))
 6010 FORMAT(' MEMORY FOR F1S AND F2S EXCEEDED'/
     1 ' EXECUTION TERMINATED')
         END
         SUBROUTINE HIGHW(SUBHI)
C
C     DEFINE THE HIGH ENERGY LIMIT OF THE DISPERSION INTEGRAL,
C     W*W/(2*PI*PI*C) INTEGRAL SIG(W')*DW'/(W'*W'-W*W)
C     AT HIGH ENERGIES IF PRINCIPLE THIS APPROACHES,
C     1/(2*PI*PI*C) INTEGRAL SIG(W')*DW'
C
C     NOTE, SINCE THE CROSS SECTIONS HAVE BEEN LINEARIZED INTEGRATION
C     WILL BE EXACT.
C
C     ARGUMENTS
C     =========
C     SUBHI   = HIGH ENERGY LIMIT
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
         IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
         COMMON/PHOTO1/X(10000),XT(10000),Y(10000),YT(10000),IPT,NPT
         COMMON/NORMAL/XNORMR,XNORMI,F1SNRM,F2SNRM

COMMON/CONJOB/R0,FOURPI,TOMSON,TOM38,XHIGH,EDGES,DEDGE,DEDGE2,UP,
     1 DOWN,DEDG10
C***** DOUBLE *****
         DATA ZERO/0.0D+00/
C***** DOUBLE *****
C***** SINGLE *****
C     DATA ZERO/0.0E+00/
C***** SINGLE *****
         SUM=ZERO
         XUP=XT(1)
         YUP=YT(1)
         DO 10 I=2,NPT
         XDOWN=XUP
         YDOWN=YUP
         XUP=XT(I)
         YUP=YT(I)
   10 SUM=SUM+(XUP-XDOWN)*(YUP+YDOWN)
         SUBHI=XNORMR*SUM/R0
         RETURN
         END
         SUBROUTINE DEPOSE(E,SUM)
C
C     PURPOSE
```

```
C      =======
C      CALCULATE DISPERSION INTEGRAL,
C      C1*E*E*INTEGRAL SIG(E')*DE'/(E'*E'-E*E)
C      TO DEFINE REAL SCATTERING AMPLITUDE
C      ARGUMENTS
C      =========
C      E      = ENERGY
C      SUM    = RESULTING INTEGRAL
C
C      CROSS SECTIONS
C      ==============
C      ARE STORED IN XT, YT AND THERE ARE NPT TABULATED POINTS.
C
C      METHOD
C      ======
C      SEE COMMENT AT THE BEGINNING OF THIS PROGRAM.
C
C***** SAVE ******
C      SAVE
C***** SAVE ******
C***** DOUBLE *****
       IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
       COMMON/PHOTO1/X(10000),XT(10000),Y(10000),YT(10000),IPT,NPT
       COMMON/NORMAL/XNORMR,XNORMI,F1SNRM,F2SNRM
       COMMON/HIGHE/ELIMIT(100),TOTHI,F1SUB,F1TOT,F1SLIM,RATN
C-----DEFINE CONSTANTS.
C***** DOUBLE *****
       DATA ZERO/0.0D+00/
       DATA EPOLE/-2.0D+00/
       QLOG(Q)=DLOG(Q)
       QABS(Q)=DABS(Q)
C***** DOUBLE *****
C***** SINGLE *****
C      DATA ZERO/0.0E+00/
C      DATA EPOLE/-2.0E+00/
C      QLOG(Q)=ALOG(Q)
C      QABS(Q)=ABS(Q)
C***** SINGLE *****
C-----INITIALIZE SUM.
       SUM=ZERO
C-----LOOP OVER TABULATED INTERVALS (ASSUMING FIRST AND LAST TABULATED
C-----VALUES ARE ZERO).
       NPTM1=NPT-1
       DO 30 K=2,NPTM1
C-----DEFINE ENERGY INTERVALS WHICH NEXT TABULATED CROSS SECTION POINT
C-----CONTRIBUTES TO.
       E0=XT(K-1)
       E1=XT(K)
       E2=XT(K+1)
       D01=E1-E0
       D12=E2-E1
```

```
C-----DEFINE DISTANCE FROM INTERVAL CONTAINING THE POLE.
      X0=(E-E0)/D01
      X3=(E2-E)/D12
C
C     POSITIVE ENERGY POLE.
C
C-----SPECIAL TREATMENT NEAR THE POLE.
      IF(X0.LT.EPOLE.OR.X3.LT.EPOLE) GO TO 10
C-----NEAR POLE.
      X1=(E1-E)/D01
      X2=(E-E1)/D12
      TERM1=QLOG(D12/D01)
      IF(X0.NE.ZERO) TERM1=TERM1-X0*QLOG(QABS(X0))
      IF(X1.NE.ZERO) TERM1=TERM1-X1*QLOG(QABS(X1))
      IF(X2.NE.ZERO) TERM1=TERM1+X2*QLOG(QABS(X2))
      IF(X3.NE.ZERO) TERM1=TERM1+X3*QLOG(QABS(X3))
      GO TO 20
C-----AWAY FROM POLE.
   10 ARG1=SLOG(E,E1,E0)
      ARG2=SLOG(E,E2,E1)
      TERM1=E*(ARG1-ARG2)+(E2*ARG2-E0*ARG1)
C
C     NEGATIVE ENERGY POLE.
C
C-----ALGORITHM IS ALWAYS STABLE FOR POSITIVE E.
   20 ARG1=-SLOG(E,-E1,-E0)
      ARG2=-SLOG(E,-E2,-E1)
      TERM2=E*(ARG2-ARG1)+(E2*ARG2-E0*ARG1)
C
C     SUM OF CONTRIBUTIONS FROM BOTH POLES.
C
      SUM=SUM+YT(K)*(TERM1-TERM2)
   30 CONTINUE
C
C     NORMALIZE RESULT TO DEFINE REAL SCATTERING AMPLITUDE.
C
      SUM=F1SNRM*E*SUM+F1SLIM
      RETURN
C***** SINGLE *****
C     END
C     FUNCTION SLOG(E,E1,E2)
C***** SINGLE *****
C***** DOUBLE *****
      END
      DOUBLE PRECISION FUNCTION SLOG(E,E1,E2)
C***** DOUBLE *****
C
C     PURPOSE
C     =======
C     DEFINE LOG(ABS((E-E2)/(E-E1)))/(E2-E1)
C
C     THE ARGUMENTS CAN BE WRITTEN AS,
```

```
C      A/2 - B/2   = E - E2
C      A/2 + B/2   = E - E1
C      A           = 2*E - (E1+E2)
C      B           = E2-E1
C
C      NOTE, A = B, IMPLIES E = E1
C            A =-B, IMPLIES E = E2
C      IN EITHER OF THESE CASES THE LOG IS INFINITE. IN ALL OTHER
C      CASES THE LOG IS STABLE,
C
C      WRITE THE EXPRESSION IN THE FORM,
C      LOG((ABS((A/2-B/2)/(A/2+B/2)))/B
C  =   LOG((ABS((A-B)/(A+B)))/B
C
C      FOR A LARGER THAN B DEFINE RAT=B/A,
C      LOG(ABS(1-RAT)/(1+RAT))/B
C
C      FOR B LARGER THAN A DEFINE RAT=A/B,
C      LOG(ABS((RAT-1)/(RAT+1)))/B
C  = -LOG(ABS(1-RAT)/(1+RAT))/B
C
C      FOR LARGE RAT THE LOG CAN BE DEFINED DIRECTLY IN THIS FORM.
C      FOR SMALL RAT WE CAN USE THE EXPANSION,
C
C      LOG((1-X)/(1+X)) =-2*(X + X**3/3 + X**5/5 + X**7/7 + X**9/9 +
C                       =-2*X*(1 + X**2/3 + X**4/5 + X**6/7 + X**9/9 +
C
C      TRUNCATING THIS EXPRESSION AFTER X**9/9 WILL YIELDS RESULTS
C      WHICH ARE ACCURATE TO ORDER X**11/9. FOR ABS(X) LESS THAN
C      0.01 THIS WILL YIELD RESULT TO AN ACCURACY OF OVER 16 DIGITS.
C
C      FOR A LARGE THAN B, X=B/A,
C      LOG(ABS(1-X)/(1+X))/B
C  =   LOG(ABS(1-X)/(1+X))/(A*(B/A))
C  =   LOG(ABS(1-X)/(1+X))/(A*X)
C  = -2*(1 + X**2/3 + X**4/5 + X**6/7 + X**8/9)/A
C
C      FOR B LARGE THAN A, X=A/B,
C      LOG(ABS((X-1)/(X+1)))/B
C  = -LOG(ABS((1-X)/(1+X)))/B
C  =  2*X*(1 + X**4/3 + X**6/5 + X**8/7 + X**10/9)/B
C
C      NOTE, IN THIS FORM THE TERM IN THE DENOMINATOR (A OR B) IS ALWAYS
C      THE LARGER OF THE TWO, I.E., A STABLE DEFINITION.
C
C***** SAVE ******
C      SAVE
C***** SAVE ******
C***** DOUBLE *****
       IMPLICIT REAL*8 (A-H,O-Z)
       REAL*8 NINE
       DATA IPASS/0/
```

```
       DATA RATMIN/1.0D-02/
       DATA ONE/1.0D+00/
       DATA TWO/2.0D+00/
       DATA THREE/3.0D+00/
       DATA FIVE/5.0D+00/
       DATA SEVEN/7.0D+00/
       DATA NINE/9.0D+00/
       QABS(Q)=DABS(Q)
       QLOG(Q)=DLOG(Q)
C***** DOUBLE *****
C***** SINGLE *****
C      REAL NINE
C      DATA IPASS/0/
C      DATA RATMIN/1.0E-02/
C      DATA ONE/1.0E+00/
C      DATA TWO/2.0E+00/
C      DATA THREE/3.0D+00/
C      DATA FIVE/5.0D+00/
C      DATA SEVEN/7.0D+00/
C      DATA NINE/9.0D+00/
C      QABS(Q)=ABS(Q)
C      QLOG(Q)=ALOG(Q)
C***** SINGLE *****
C
C      DEFINE CONSTANTS FOR EXPANSION OF LOG THE FIRST TIME THIS
C      ROUTINE IS CALLED.
C
       IF(IPASS.NE.0) GO TO 10
       IPASS=1
       C1=TWO
       C3=TWO/THREE
       C5=TWO/FIVE
       C7=TWO/SEVEN
       C9=TWO/NINE
C
C      DEFINE A AND B AND RATIO TO LARGER
C
   10 A=TWO*E-(E1+E2)
       B=E2-E1
       IF(QABS(A)-QABS(B)) 40,60,20
C
C      USE B/A AND EXPANSION OR ANALYTICAL EXPRESSION.
C
   20 RAT=B/A
       IF(QABS(RAT).GE.RATMIN) GO TO 30
C-----EXPAND.
       RAT2=RAT*RAT
       SLOG=-((((C9*RAT2+C7)*RAT2+C5)*RAT2+C3)*RAT2+C1)/A
       RETURN
C-----ANALYTICAL.
   30 SLOG=QLOG(QABS((ONE-RAT)/(ONE+RAT)))/B
       RETURN
```

```
C
C     USE A/B AND EXPANSION OR ANALYTICAL EXPRESSION.
C
   40 RAT=A/B
      IF(QABS(RAT).GE.RATMIN) GO TO 50
C-----EXPAND.
      RAT2=RAT*RAT
      SLOG=RAT*((((C9*RAT2+C7)*RAT2+C5)*RAT2+C3)*RAT2+C1)/B
      RETURN
C-----ANALYTICAL.
   50 SLOG=QLOG(QABS((RAT-ONE)/(RAT+ONE)))/B
      RETURN
C
C     ILLEGAL ARGUMENTS.
C
   60 WRITE(6,6000) E,E1,E2
      STOP
C***** DOUBLE *****
 6000 FORMAT(' ILLEGAL ARGUMENTS FOR FUNCTION SLOG'/
     1 3(1PD16.8))
C***** DOUBLE *****
C***** SINGLE *****
C6000 FORMAT(' ILLEGAL ARGUMENTS FOR FUNCTION SLOG'/
C    1 3(1PE16.8))
C***** SINGLE *****
      END
      SUBROUTINE ZHOL(Z,ZBCD)
C
C     DEFINE THE CHEMICAL SYMBOL CORRESPONDING TO ANY Z.
C
C     Z MUST BE BETWEEN 0 (NEUTRON) AND 104.
C
C     ARGUMENTS
C     =========
C     Z      = ATOMIC NUMBER
C     ZBCD   = HOLLERITH EQUIVALENT IN A4 FORMAT
C
C     EXAMPLE, FOR Z = 6, ZBCD IS RETURNED AS '-C  ', WITHOUT THE
C     ENCLOSING QUOTES.
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
      INTEGER Z
C***** CHARACTER *****
      CHARACTER*4 DUM1,DUM2,ZBCD,ZTAB
C***** CHARACTER *****
C***** INTEGER *****
C     INTEGER DUM1,DUM2,ZBCD,ZTAB
C***** INTEGER *****
      DIMENSION ZTAB(111),DUM1(55),DUM2(56)
      EQUIVALENCE (ZTAB(1),DUM1(1)),(ZTAB(56),DUM2(1))
```

```
      DATA DUM1/ '-Nu ',
     1 '-H  ','-He ','-Li ','-Be ','-B  ','-C  ',
     2 '-N  ','-O  ','-F  ','-Ne ','-Na ','-Mg ',
     3 '-Al ','-Si ','-P  ','-S  ','-Cl ','-Ar ',
     4 '-K  ','-Ca ','-Sc ','-Ti ','-V  ','-Cr ',
     5 '-Mn ','-Fe ','-Co ','-Ni ','-Cu ','-Zn ',
     6 '-Ga ','-Ge ','-As ','-Se ','-Br ','-Kr ',
     7 '-Rb ','-Sr ','-Y  ','-Zr ','-Nb ','-Mo ',
     8 '-Tc ','-Ru ','-Rh ','-Pd ','-Ag ','-Cd ',
     9 '-In ','-Sn ','-Sb ','-Te ','-I  ','-Xe '/
      DATA DUM2/
     1 '-Cs ','-Ba ','-La ','-Ce ','-Pr ','-Nd ',
     2 '-Pm ','-Sm ','-Eu ','-Gd ','-Tb ','-Dy ',
     3 '-Ho ','-Er ','-Tm ','-Yb ','-Lu ','-Hf ',
     4 '-Ta ','-W  ','-Re ','-Os ','-Ir ','-Pt ',
     5 '-Au ','-Hg ','-Tl ','-Pb ','-Bi ','-Po ',
     6 '-At ','-Rn ','-Fr ','-Ra ','-Ac ','-Th ',
     7 '-Pa ','-U  ','-Np ','-Pu ','-Am ','-Cm ',
     8 '-Bk ','-Cf ','-Es ','-Fm ','-Md ','-No ',
     9 '-Lr ','  ','  ','  ','  ','  ','  ','Err '/
C-----Z MUST BE BETWEEN 0 (NEUTRON) AND 104.
      IZ=Z
      IF(IZ.LT.0.OR.IZ.GT.104) IZ=110
      ZBCD=ZTAB(IZ+1)
      RETURN
      END
      SUBROUTINE SUBHOL(ISHELL,SHBCD)
C
C     USE ENDL SUBSHELL DESIGNATOR TO DEFINE SUBSHELL
C
C     ARGUMENTS
C     =========
C     ISHELL  = ENDL SUBSHELL DESIGNATOR
C     SHBCD   = HOLLERITH EQUIVALENT RETURNED IN 3A4 FORMAT
C
C     FOR EXAMPLE, FOR ISHELL=1, SHBCD IS RETURNED AS 'K (1S1/2)   '
C     WITHOUT THE ENCLOSING QUOTES.
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** CHARACTER *****
      CHARACTER*4 SHTAB,SHTAB1,SHTAB2,SHTAB3,SHBCD,TOTAL
C***** CHARACTER *****
C***** INTEGER *******
C     INTEGER SHTAB,SHTAB1,SHTAB2,SHTAB3,SHBCD,TOTAL
C***** INTEGER *******
      DIMENSION SHTAB(3,62),SHTAB1(3,27),SHTAB2(3,27),SHTAB3(3,8),
     1 SHBCD(3),TOTAL(3)
      EQUIVALENCE (SHTAB(1,1),SHTAB1(1,1)),(SHTAB(1,28),SHTAB2(1,1)),
     1 (SHTAB(1,55),SHTAB3(1,1))
      DATA SHTAB1/
```

```fortran
     1 'K (1','s1/2',')   ','L (2',')   ',' ','L1 (','2s1/','2) ',
     2 'L23 ','(2p)',' ','L2 (','2p1/','2) ','L3 (','2p3/','2) ',
     3 'M (3',')   ',' ','M1 (','3s1/','2) ','M23 ','(3p)',' ',
     4 'M2 (','3p1/','2) ','M3 (','3p3/','2) ','M45 ','(3d)',' ',
     5 'M4 (','3d3/','2) ','M5 (','3d5/','2) ','N (4',')   ',' ',
     6 'N1 (','4s1/','2) ','N23 ','(4p)',' ','N2 (','4p1/','2) ',
     7 'N3 (','4p3/','2) ','N45 ','(4d)',' ','N4 (','4d3/','2) ',
     8 'N5 (','4d5/','2) ','N67 ','(4f)',' ','N6 (','4f5/','2) ',
     9 'N7 (','4f7/','2) ','O (5',')   ',' ','O1 (','5s1/','2) '/
      DATA SHTAB2/
     1 'O23 ','(5p)',' ','O2 (','5p1/','2) ','O3 (','5p3/','2) ',
     2 'O45 ','(5d)',' ','O4 (','5d3/','2) ','O5 (','5d5/','2) ',
     3 'O67 ','(5f)',' ','O6 (','5f5/','2) ','O7 (','5f7/','2) ',
     4 'O89 ','(5g)',' ','O8 (','5g7/','2) ','O9 (','5g9/','2) ',
     5 'P (6',')   ',' ','P1 (','6s1/','2) ','P23 ','(6p)',' ',
     6 'P2 (','6p1/','2) ','P3 (','6p3/','2) ','P45 ','(6d)',' ',
     7 'P4 (','6d3/','2) ','P5 (','6d5/','2) ','P67 ','(6f)',' ',
     8 'P6 (','6f5/','2) ','P7 (','6f7/','2) ','P89 ','(6g)',' ',
     9 'P8 (','6g7/','2) ','P9 (','6g9/','2) ','P101','1 (6','h)  '/
      DATA SHTAB3/
     1 'P10 ','(6h9','/2) ','P11 ','(6h1','1/2)','Q (7',')   ',' ',
     2 'Q1 (','7s1/','2) ','Q23 ','(7p)',' ','Q2 (','7p1/','2) ',
     3 'Q3 (','7p3/','2) ','ERRO','R   ',' '/
      DATA TOTAL/'Tota','l   ',' '/
C-----INSURE SUBSHELL INDEX IS IN LEGAL RANGE (1 TO 61).
      KSHELL=ISHELL
      IF(KSHELL.NE.0) GO TO 20
      DO 10 I=1,3
   10 SHBCD(I)=TOTAL(I)
      RETURN
   20 IF(KSHELL.LT.1.OR.KSHELL.GT.61) KSHELL=62
C-----DEFINE SUBSHELL.
      DO 30 I=1,3
   30 SHBCD(I)=SHTAB(I,KSHELL)
      RETURN
      END
      SUBROUTINE SUMMER
C
C     ADD SUBSHELL F1S ANS F2S TO ELEMENT TOTAL F1 AND F2.
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
      COMMON/PHOTO1/X(10000),XT(10000),Y(10000),YT(10000),IPT,NPT
      COMMON/SHELL/XFS(10000),F1S(10000),F2S(10000),NSCATS
      COMMON/SCAT/XF(10000),F1(10000),F2(10000),NSCAT
      COMMON/ATOM/IZLAST,ICNOW,ISUB,NSUB
      COMMON/XCTEND/XETEND,IXTEND,MYANOM,IMANOM(3,2)
C
```

```
C       MERGE ENERGIES (WARNING...THE FOLLOWING CALL ASSUMES THAT F1S,
C       F2S, F1, F2, Y AND YT ARE EACH DIMENSIONED 10000 AND THAT F1S
C       IMMEDIATELY PRECEEDS F2S, F1 IMMEDIATELY PRECEEDS F2, X
C       IMMEDIATELY PRECEEDS XT AND Y IMMEDIATELY PRECEEDS YT IN COMMON)
C
        CALL EMERGN(XFS,F1S,NSCATS,XF,F1,NSCAT,X,Y,2,10000,2)
C-----COUNT SUBSHELLS.
        NSUB=NSUB+1
C-----RE-INITIALIZE SUBSHELL POINT COUNT
        NSCATS=0
        DO 825 I=1,3
        IMANOM(I,1)=0
  825   IMANOM(I,2)=0
        RETURN
        END
        SUBROUTINE FINALE
C
C       END OF Z. OUTPUT SCATTERING FACTORS. CALCULATE COHERENT CROSS
C       SECTIONS.
C
C***** SAVE ******
C       SAVE
C***** SAVE ******
C***** DOUBLE *****
        IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
C***** CHARACTER *****
        CHARACTER*4 ZBCD
C***** CHARACTER *****
C***** INTEGER *****
C       INTEGER ZBCD
C***** INTEGER *****
        COMMON/ATOM/IZLAST,ICNOW,ISUB,NSUB
        COMMON/ZID/ZBCD
        COMMON/SCAT/XF(10000),F1(10000),F2(10000),NSCAT
        COMMON/PHOTO1/X(10000),XT(10000),Y(10000),YT(10000),IPT,NPT
        COMMON/SHELL/XFS(10000),F1S(10000),F2S(10000),NSCATS
        COMMON/FFLIMS/FFLIM1(10000),FFLIM2(10000)

COMMON/CONJOB/R0,FOURPI,TOMSON,TOM38,XHIGH,EDGES,DEDGE,DEDGE2,UP,
     1 DOWN,DEDG10
        COMMON/OUTER/IMOUT(13)
        COMMON/HIGHE/ELIMIT(100),TOTHI,F1SUB,F1TOT,F1SLIM,RATN
        COMMON/ERANGE/XMIN,XMAX
        COMMON/OKERR/ERROK,ERRFF
        COMMON/SIZNFF/NFF
        COMMON/ZRANGE/IZMIN,IZMAX
        COMMON/XCTEND/XETEND,IXTEND,MYANOM,IMANOM(3,2)
        DIMENSION XFF(5000),YFF(5000),XFFUSE(5000),YFFUSE(5000),
     1 YCOH(10000),XIOUT(4)
        EQUIVALENCE (X(1),XFF(1)),(Y(1),YFF(1)),(X(5001),XFFUSE(1)),
     1 (Y(5001),YFFUSE(1)),(F2S(1),YCOH(1))
```

```
C-----DEFINE CONVERSION FACTOR FROM ENERGY IN MEV TO WAVELEGTH IN
C-----ARGSTROMS (FOR USE WITH FORM FACTORS).
C***** DOUBLE *****
      DATA E2WAVE/1.239852D-10/
      DATA ZERO/0.0D+00/
      DATA TWO/2.0D+00/
      DATA ONE/1.0D+00/
      DATA TEN/1.0D+01/
      DATA ATWT/0.0D+00/
      DATA ELEVL/0.0D+00/
      DATA HAFLYF/0.0D+00/
      DATA TMP/0.0D+00/
      DATA QVALUE/0.0D+00/
      DATA XIOUT/0.0D+00,0.0D+00,0.0D+00,0.0D+00/
      QLOG(Z)=DLOG(Z)
      QEXP(Z)=DEXP(Z)
C***** DOUBLE *****
C***** SINGLE *****
C     DATA E2WAVE/1.239852E-10/
C     DATA ZERO/0.0E+00/
C     DATA ONE/1.0E+00/
C     DATA TWO/2.0E+00/
C     DATA TEN/1.0E+01/
C     DATA ATWT/0.0E+00/
C     DATA ELEVL/0.0E+00/
C     DATA HAFLYF/0.0E+00/
C     DATA TMP/0.0E+00/
C     DATA QVALUE/0.0E+00/
C     DATA XIOUT/0.0E+00,0.0E+00,0.0E+00,0.0E+00/
C     QLOG(Z)=ALOG(Z)
C     QEXP(Z)=EXP(Z)
C***** SINGLE *****
C
C     IGNORE IF NOT IN REQUESTED Z RANGE.
C
      IF(IZLAST.LT.IZMIN.OR.IZLAST.GT.IZMAX) GO TO 310
      ZNOW=IZLAST
C
C     IF NO ANOMALOUS DEPENDENT OUTPUT SKIP TO COHERENT WITHOUT
C     ANOMALOUS.
C
      IF(MYANOM.LE.0) GO TO 110
C-----SAVE HIGH ENERGY LIMIT VS. Z.
      ELIMIT(IZLAST)=F1TOT
C-----PRINT THEORETICAL LIMIT (Z) AND WHAT WAS FOUND.
      IF(ISUB.GT.0.AND.NSCAT.LE.0) WRITE(6,6005) ZNOW,F1TOT,TOTHI
C-----NOTHING ELSE TO DO IF NO COMBINED POINTS.
      IF(NSCAT.LE.0) GO TO 310
C
C     THIN RESULTS (WARNING...THE FOLLOWING CALL ASSUMES THAT F1 AND
C     F2 ARE EACH OF DIMENSION 10000 AND F2 IMMEDIATELY FOLLOWS F1
C     IN CORE.
```

```
C
C     ONLY THIN RESULTS IF MORE THAN 1 SUBSHELL (IF ONLY 1 SUBSHELL
C     IT HAS ALREADY BEEN THINNED).
C
      IF(NSUB.GT.1) CALL THIN3L(XF,F1,NSCAT,10000,2,ERROK)
C
C     INSERT POINTS WHERE F21 IS EQUAL TO 0 (I.E., WILL EXACTLY
C     CANCEL CONTRIBUTION OF THE FORM FACTOR).
C
C-----INITIALIZE INDEX.
      I=2
      ZNOWM=ZERO
C-----COPY IF EQUAL TO Z.
   10 IF(F1(I).EQ.ZNOWM) GO TO 40
C-----DOES F1 CROSS Z BETWEEN POINTS.
      IF(F1(I).GT.ZNOWM.AND.F1(I-1).LT.ZNOWM) GO TO 20
      IF(F1(I).LT.ZNOWM.AND.F1(I-1).GT.ZNOWM) GO TO 20
C-----NO. COPY POINT.
      GO TO 40
C-----YES. MOVE ALL FOLLOWING POINTS BACK ONE LOCATION.
   20 II=NSCAT
      DO 30 K=I,NSCAT
      XF(II+1)=XF(II)
      F1(II+1)=F1(II)
      F2(II+1)=F2(II)
   30 II=II-1
      NSCAT=NSCAT+1
C-----INSERT POINT USING LINEAR INTERPOLATION (NOTE, THE POINTS TO
C-----NOW USE ARE I-1 AND I+1, SINCE ALL POINTS ABOVE I HAVE BEEN
C-----SHIFTED BACK ONE LOCATION)
      WT1=(ZNOWM-F1(I-1))/(F1(I+1)-F1(I-1))
      WT2=ONE-WT1
      XF(I)=WT1*XF(I+1)+WT2*XF(I-1)
      F1(I)=ZNOWM
      WT1=(XF(I)-XF(I-1))/(XF(I+1)-XF(I-1))
      WT2=ONE-WT1
      F2(I)=WT1*F2(I+1)+WT2*F2(I-1)
C-----CONTINUE SEARCH AFTER I+1 (SINCE POINTS HAVE BEEN SHIFTED
C-----BACK ONE LOCATION)
      I=I+1
C-----INCREMENT INDEX AND CONTINUE TO END OF TABLE.
   40 I=I+1
      IF(I.LE.NSCAT) GO TO 10
C-----PRINT THEORETICAL LIMIT (Z) AND WHAT WAS FOUND.
      IF(ISUB.GT.0) WRITE(6,6000) ZNOW,F1TOT,TOTHI,NSCAT
C
C     OUTPUT SCATTERING FACTORS
C
C-----IMAGINARY SCATTERING FACTOR.
      IF(IMOUT(6).LE.0) GO TO 50
C
C     ENDL OUTPUT
```

```
C
      IZA=1000*IZLAST
      CALL OUTL1(16,IZA,7,0,ATWT,890714,2,2,ELEVL,HAFLYF,TMP)
      CALL OUTL2(16,93,943,0,QVALUE,XIOUT)
      CALL OUTAB2(16,XF,F2,NSCAT)
C
C     PLOTTAB OUTPUT.
C
      WRITE(11,1140) IZLAST,ZBCD
      CALL PLTTAB(11,XF,F2,NSCAT)
C-----REAL SCATTERING AMPLITUDE
   50 IF(IMOUT(7).LE.0) GO TO 60
      WRITE(11,1130) IZLAST,ZBCD
      CALL PLTTAB(11,XF,F1,NSCAT)
C
C     CHANGE FROM REAL AMPLITUDE TO REAL SCATTERING FACTOR FROM THIS
C     POINT ON.
C
   60 DO 70 I=1,NSCAT
   70 F1(I)=-(ZNOW+F1(I))
C-----REAL SCATTERING FACTOR.
      IF(IMOUT(8).LE.0) GO TO 80
C
C     ENDL OUTPUT
C
      IZA=1000*IZLAST
      CALL OUTL1(16,IZA,7,0,ATWT,890714,2,2,ELEVL,HAFLYF,TMP)
      CALL OUTL2(16,93,944,0,QVALUE,XIOUT)
      CALL OUTAB2(16,XF,F1,NSCAT)
C
C     PLOTTAB OUTPUT.
C
      WRITE(11,1130) IZLAST,ZBCD
      CALL PLTTAB(11,XF,F1,NSCAT)
C
C     COHERENT CROSS SECTION...PLOTTAB OUTPUT.
C
   80 IF(IMOUT(9).LE.0.AND.IMOUT(10).LE.0.AND.IMOUT(11).LE.0.AND.
     1   IMOUT(12).LE.0) GO TO 310
C
C     AT LEAST ONE TYPE OF COHERENT CROSS SECTION WILL BE OUTPUT. IF
C     THE SCATTERING FACTORS HAVE BEEN TRUNCATED AT AN ENERGY LESS THAN
C     XMAX EXTEND THEM UP TO XMAX, USING 10 POINTS PER ENERGY DECADE
C     IN ORDER TO ALLOW THE COHERENT CROSS SECTIONS TO BE CALCULATED
C     UP TO XMAX.
C
C     IF CROSS SECTION EXTENSION IS REQUESTED EXTEND CALCULATION UP TO
C     MAXIMUM ENERGY OF TABULATED PHOTOELECTRIC CROSS SECTIONS.
C
C-----DO NOT EXTEND ENERGY IF NO CROSS SECTION OUTPUT.
      IF(IMOUT(9).LE.0.AND.IMOUT(10).LE.0.AND.IMOUT(11).LE.0) GO TO 110
      IF(IXTEND.LE.0) XETEND=XMAX
```

```
      IF(XF(NSCAT).GE.XETEND) GO TO 110
      DX=QEXP(QLOG(TEN)/TEN)
      N=NSCAT+1
      IF(N.GT.10000) GO TO 110
      DO 90 NSCAT=N,10000
      XF(NSCAT)=DX*XF(NSCAT-1)
      F1(NSCAT)=ZERO
      F2(NSCAT)=ZERO
      IF(XF(NSCAT).GE.XETEND) GO TO 100
   90 CONTINUE
      NSCAT=10000
  100 XF(NSCAT)=XETEND
C
C     WILL FORM FACTORS BE USED.
C
  110 IF(IMOUT(9).LE.0.AND.IMOUT(11).LE.0.AND.IMOUT(12).LE.0) GO TO 290
C
C     READ FORM FACTOR.
C
  120 READ(15,1500,END=320,ERR=320) IZFORM
      READ(15,1510,END=320,ERR=320) IC,II
C
C     SKIP TO CURRENT Z, C = 93, I = 941 (COHERENT FORM FACTOR)
C
      IF(IZFORM-IZLAST) 140,130,320
  130 IF(IC.EQ.93.AND.II.EQ.941) GO TO 150
C-----SKIP SECTION.
  140 READ(15,1520) XFF(1),YFF(1),IFLAG
      IF(IFLAG) 140,140,120
C-----READ SECTION.
  150 DO 160 NFF=1,1000
      READ(15,1520) XFF(NFF),YFF(NFF),IFLAG
      IF(IFLAG.NE.0) GO TO 170
  160 CONTINUE
      GO TO 330
  170 NFF=NFF-1
C
C     FORM FACTOR IS GIVEN AS A AS FUNCTION OF X = SIN(THETA/2)/LAMBDA
C
C     USE THE RELATIONSHIP 1-COS(THETA) = 2*SIN(THETA/2)**2
C     TO OBTAIN DISTRIBUTION VS. 1-COS(THETA). NOTE, IF THE FORM FACTOR
C     IS LOG-LOG INTERPOLABLE IN X IT IS ALSO LOG-LOG INTERPOLABLE IN
C     X**2, SO IT WILL BE LOG-LOG INTERPOLABLE IN 1-COS.
C
      DO 180 K=1,NFF
  180 XFF(K)=TWO*XFF(K)*XFF(K)
C
C     DATA IS LOG-LOG INTERPOLABLE. CONVERT TO LINEAR-LINEAR.
C
C-----MUST ASSUME LINEAR BETWEEN FIRST 2 POINTS (BECAUSE OF THE 0.0
C-----IN SIN(THETA/2) AT THETA=0.0)
      XFFUSE(1)=XFF(1)
```

```
      YFFUSE(1)=YFF(1)
      CALL LINEAR(XFF(2),YFF(2),NFF-1,XFFUSE(2),YFFUSE(2),NFFUSE,ERRFF)
      NFFUSE=NFFUSE+1
C-----COPY BACK TO ORIGINAL ARRAY FOR LATER USE.
      NFF=NFFUSE
      DO 190 K=1,NFFUSE
      XFF(K)=XFFUSE(K)
  190 YFF(K)=YFFUSE(K)
C
C     FORM FACTOR LIMITS
C
      IF(IMOUT(12).LE.0) GO TO 250
C-----FORWARD LIMIT IS CONSTANT.
      DO 200 I=1,NSCAT
  200 FFLIM1(I)=YFF(1)
      WRITE(11,1150) IZLAST,ZBCD
      CALL PLTTAB(11,XF,FFLIM1,NSCAT)
C-----BACKWARD LIMIT IS VARIABLE.
      DO 240 I=1,NSCAT
      WAVE=E2WAVE/XF(I)
      WAVE=WAVE*WAVE
      XLAST=XFF(1)*WAVE
      YLAST=YFF(1)
      DO 220 K=2,NFFUSE
      XNOW=XFF(K)*WAVE
      IF(XNOW-TWO) 210,230,230
  210 XLAST=XNOW
  220 YLAST=YNOW
      GO TO 240
  230 WT1=(TWO-XLAST)/(XNOW-XLAST)
      WT2=ONE-WT1
      YNOW=WT1*YNOW+WT2*YLAST
  240 FFLIM2(I)=YNOW
      WRITE(11,1160) IZLAST,ZBCD
      CALL PLTTAB(11,XF,FFLIM2,NSCAT)
C
C     CALCULATE COHERENT CROSS SECTION AT EACH ENERGY POINT.
C
C     WITH ANOMALOUS
C
  250 IF(IMOUT(9).LE.0) GO TO 270
      DO 260 I=1,NSCAT
  260 CALL COHERT(YCOH(I),XF(I),F1(I),F2(I),1)
      WRITE(11,1100) IZLAST,ZBCD
      CALL PLTTAB(11,XF,YCOH,NSCAT)
C
C     WITHOUT ANOMALOUS
C
  270 IF(IMOUT(11).LE.0) GO TO 290
      DO 280 I=1,NSCAT
  280 CALL COHERT(YCOH(I),XF(I),ZERO,ZERO,2)
      WRITE(11,1108) IZLAST,ZBCD
```

```
      CALL PLTTAB(11,XF,YCOH,NSCAT)
C
C     LOW ENERGY APPROXIMATION COHERENT CROSS SECTION...PLOTTAB OUTPUT.
C
  290 IF(IMOUT(10).LE.0) GO TO 310
      DO 300 I=1,NSCAT
  300 YCOH(I)=TOMSON*((ZNOW-F1(I))**2+F2(I)*F2(I))
      WRITE(11,1105) IZLAST,ZBCD
      CALL PLTTAB(11,XF,YCOH,NSCAT)
C
C     RE-INITIALIZE VARIABLES.
C
  310 IF(MYANOM.NE.0) NSCAT=0
      NSUB=0
      TOTHI=ZERO
      F1TOT=ZERO
      RETURN
C
C     END OF FORM FACTOR DATA.
C
  320 WRITE(6,6010)
      CALL EXIT
C
C     TOO MANY FORM FACTOR POINTS.
C
  330 WRITE(6,6020)
      CALL EXIT
 1100 FORMAT(I3,A4,' Coherent')
 1105 FORMAT(I3,A4,' Coherent (Form Factor=Z)')
 1108 FORMAT(I3,A4,' Coherent (No Anomalous)')
 1130 FORMAT(I3,A4,' Real')
 1140 FORMAT(I3,A4,' Imaginary')
 1150 FORMAT(I3,A4,' Form Factor Max')
 1160 FORMAT(I3,A4,' Form Factor Min')
 1500 FORMAT(I3)
 1510 FORMAT(I2,I3)
C***** DOUBLE *****
 1520 FORMAT(2D11.4,49X,I1)
C***** DOUBLE *****
C***** SINGLE *****
C1520 FORMAT(2E11.4,49X,I1)
C***** SINGLE *****
 6000 FORMAT(1X,72('=')/34X,F9.4,F11.4,F11.4,I8/1X,72('='))
 6005 FORMAT(1X,72('=')/34X,F9.4,F11.4,F11.4/1X,72('='))
 6010 FORMAT(' END OF FORM FACTOR DATA')
 6020 FORMAT(1X,72('=')/' TOO MANY FORM FACTOR POINTS'/1X,72('='))
      END
      SUBROUTINE LINEAR(X,Y,IPT,XL,YL,LPT,ERROK)
C
C     CONVERT LOG-LOG INTERPOLABLE TABLE TO LINEAR-LINEAR INTERPOLABLE
C     FORM TO WITHIN ANY GIVEN ACCURACY.
C
```

```
C     ARGUMENTS
C     =========
C     X       = X VALUES FOR ORIGINAL DATA
C     Y       = Y VALUES FOR ORIGINAL DATA
C     IPT     = NUMBER OF VALUES FOR ORIGINAL DATA
C     XL      = X VALUES FOR LINEARIZED DATA
C     YL      = Y VALUES FOR LINEARIZED DATA
C     LPT     = NUMBER OF VALUES FOR LINEARIZED DATA
C     ERROK   = ALLOWABLE UNCERTAINTY
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
      COMMON/ERANGE/XMIN,XMAX
      DIMENSION X(10000),Y(10000),XL(10000),YL(10000)
C-----DEFINE MAXIMUM ALLOWABLE X STEP.
C***** DOUBLE *****
      DATA DXMAX/1.1D+00/
      QSQRT(Z)=DSQRT(Z)
C***** DOUBLE *****
C***** SINGLE *****
C     DATA DXMAX/1.1E+00/
C     QSQRT(Z)=SQRT(Z)
C***** SINGLE *****
C-----DEFINE FIRST POINT IN LINEARIZED TABLE.
      LPT=1
      XL(1)=X(1)
      YL(1)=Y(1)
      XLAST=X(1)
      YLAST=Y(1)
      XNEXT=DXMAX*XLAST
C-----LOOP OVER ENERGY INTERVALS.
      DO 80 I=2,IPT
C-----START (OR RESTART) AT END OF INTERVAL.
   10 XNOW=X(I)
      YNOW=Y(I)
C-----AT DISCONTINUITY KEEP BOTH POINTS.
      IF(XNOW.GT.XLAST) GO TO 20
      LOOP=1
      GO TO 70
C-----INTERVAL HALVING LOOP (DO NOT ALLOW INTERVAL TO BE HALVED MORE
C-----THAN 20 TIMES, EQUIVALENT TO DIVIDING INTO 2**20 (ABOUT 1 MILLION)
C-----SUBINTERVALS.
   20 DO 50 LOOP=1,20
C-----DEFINE LOG-LOG INTERPOLATED VALUES AT MIDPOINT OF INTERVAL.
      XMID=QSQRT(XNOW*XLAST)
      YMID=QSQRT(YNOW*YLAST)
C-----CHECK FOR MINIMUM X STEP (ONLY UP TO XMAX).
      IF(XMID.LT.XMAX.AND.XMID.GT.XNEXT) GO TO 40
```

```
C-----CANNOT SUBDIVIDE ZERO LENGTH INTERVAL.
      DXNOW=XNOW-XLAST
      IF(DXNOW) 70,70,30
C-----LINEARLY INTERPOLATE FROM ENDS OF INTERVAL TO MIDPOINT.
   30 YLIN=((XMID-XLAST)*YNOW+(XNOW-XMID)*YLAST)/DXNOW
C-----SEE IF LINEARLY INTERPOLATION WILL ALLOW VALUE AT MIDPOINT TO
C-----BE ACCURATELY APPROXIMATED.
      IF(DABS(YMID-YLIN).LE.ERROK*YMID) GO TO 60
C-----NO. HALF INTERVAL AND CONTINUE.
   40 XNOW=XMID
   50 YNOW=YMID
      LOOP=21
C-----YES. KEEP MIDPOINT AND ENDPOINT OF INTERVAL.
   60 LPT=LPT+1
      XL(LPT)=XMID
      YL(LPT)=YMID
   70 LPT=LPT+1
      XL(LPT)=XNOW
      YL(LPT)=YNOW
      XLAST=XNOW
      YLAST=YNOW
      XNEXT=DXMAX*XNOW
C-----CONTINUE WITH CURRENT X INTERVAL UNTIL THE ENTIRE INTERVAL HAS
C-----BEEN COMPLETED.
      IF(LOOP.GT.1) GO TO 10
C-----END OF INTERVAL LOOP.
   80 CONTINUE
C-----THIN RESULTS.
      CALL THIN3L(XL,YL,LPT,10000,1,ERROK)
      RETURN
      END
      SUBROUTINE COHERT(SUM,XF,F1,F2,ITYPE)
C
C     DEFINE THE COHERENT CROSS SECTION WITH AND WITHOUT ANOMALOUS
C     SCATTERING.
C
C     INTEGRATE PRODUCT OF THOMSON CROSS SECTION, FORM FACTORS AND
C     SCATTERING FACTORS TO DEFINE COHERENT CROSS SECTION.
C
C     THE EXPRESSION WITHOUT ANOMALOUS SCATTERING IS IDENTICAL IF
C     F1 AND F2 ARE SIMPLY IGNORED.
C
C     ARGUMENTS
C     =========
C     SUM      = CROSS SECTION
C     XF       = ENERGY AT WHICH TO EVALUATE CROSS SECTIONS
C     F1       = REAL SCATTERING FACTOR AT ENERGY XF
C     F2       = IMAGINARY SCATTERING FACTOR AT ENERGY XF
C     ITYPE    = 1 - WITH ANOMALOUS
C              = 2 - WITHOUT ANOMALOUS
C
C     METHOD
```

```
C       ======
C       THE ANGULAR DISTRIBUTION IS,
C       3*T/8*(1+COS**2)*((FF+F1)**2+(F2**2))*D(COS)
C
C       CALCULATE (FF+F1)**2+(F2**2) IN LINEARLY INTERPOLABLE FORM
C
C       MULTIPLY BY (1+COS**2) AND ANALYTICALLY INTEGRATE
C
C       MULTIPLY BY 3*T/8 ONLY AT THE END
C
C       THE INTEGRAL IS (1+COS**2)*Y(COS)*D(COS)
C       WHERE Y(COS) = (FF+F1)**2+(F2**2) IS CALCULATED TO BE A LINEAR
C       FUNCTION OF COS IN A NUMBER OF COS INTERVALS.
C
C       WE CAN DEFINE THE INTEGRAL AS A SUM OF THE INTERVALS THAT Y(COS)
C       HAS BEEN LINEARIZED IN.
C
C       THIS INTEGRAL IS VERY NUMERICALLY UNSTABLE UNLESS CARE IS USED TO
C       EXPLICITLY CANCEL ALL TERMS BEFIRE EVALUATING THE INTEGRAL. THIS
C       CAN BE DONE BY USING THE UNIT BASE NORM TRANSFORMATION.
C
C       WITHIN EACH INTERVAL, COS1 TO COS2, THAT Y(COS) HAS BEEN LINEARIZE
C       TRANSFORM ALL VARIABLES TO UNIT BASE NORM,
C       AVC   = (COS2+COS1)/2
C       DC    = (COS2-COS1)/2
C       AVY   = (Y2+Y1)/2
C       DY    = (Y2-Y1)/2
C       COS   = AVC+DC*Q
C       Y     = AVY+DY*Q
C       D(COS)= DC*D(Q)
C       WHERE Q IS THE UNIT BASE VARIABLE WHICH VARIES FROM -1 TO +1.
C
C       THE TRANSFORMED INTEGRAL THEN BECOMES,
C       (1+COS**2)*Y(COS)*D(COS) =
C
C       [1+(AVC+DC*Q)**2]*[AVY+DY*Q]*DC*D(Q) =
C
C       [1+AVC*AVC+2*DC*AVC*Q+DC*DC*Q**2]*[AVY+DY*Q]*DC*D(Q) =
C
C       {AVY*[1+AVC*AVC   + 2*DC*AVC*Q    + DC*DC*Q**2]+
C         DY*[Q+AVC*AVC*Q + 2*DC*AVC*Q**2 + DC*DC*Q**3]}*DC*D(Q)
C
C       WHEN WE INTEGRATE FROM -1 TO +1 ONLY EVEN POWERS OF Q WILL MAKE
C       A NON-ZERO CONTRIBUTION TO THE INTEGRAL AND THE RESULT IS,
C
C       2*DC*{AVY*[1+AVC*AVC+DC*DC/3] + DY*[2*DC*AVC/3]}
C
C       THEREFORE THE CONTRIBUTION OF ONE COS INTERVAL IS,
C
C       2*DC*{AVY*[3+3*AVC*AVC+DC*DC] + 2*DY*DC*AVC}/3
C
C       THE FACTOR OF 2/3 NEED ONLY BE CONSIDERED AS A NORMALIZATION
```

```
C       FACTOR AT THE END OF THE INTEGRAL.
C
C***** SAVE ******
C       SAVE
C***** SAVE ******
C***** DOUBLE *****
        IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
        COMMON/PHOTO1/X(10000),XT(10000),Y(10000),YT(10000),IPT,NPT

COMMON/CONJOB/R0,FOURPI,TOMSON,TOM38,XHIGH,EDGES,DEDGE,DEDGE2,UP,
     1 DOWN,DEDG10
        COMMON/OKERR/ERROK,ERRFF
        COMMON/SIZNFF/NFF
        COMMON/POWERS/POWER
        DIMENSION XFF(5000),YFF(5000),XFFUSE(5000),YFFUSE(5000)
        EQUIVALENCE (X(1),XFF(1)),(Y(1),YFF(1)),(X(5001),XFFUSE(1)),
     1 (Y(5001),YFFUSE(1))
C-----DEFINE CONVERSION FACTOR FROM ENERGY IN MEV TO WAVELEGTH IN
C-----ARGSTROMS (FOR USE WITH FORM FACTORS).
C***** DOUBLE *****
        DATA E2WAVE/1.239852D-10/
        DATA ZERO/0.0D+00/
        DATA ONE/1.0D+00/
        DATA TWO/2.0D+00/
        DATA THREE/3.0D+00/
        QABS(Z)=DABS(Z)
C***** DOUBLE *****
C***** SINGLE *****
C       DATA E2WAVE/1.239852E-10/
C       DATA ZERO/0.0E+00/
C       DATA ONE/1.0E+00/
C       DATA TWO/2.0E+00/
C       DATA THREE/3.0E+00/
C       QABS(Z)=ABS(Z)
C***** SINGLE *****
C
C       FORM FACTORS ARE NOW LINEARLY INTERPOLABLE AS A FUNCTION OF
C       X = (1 - COS)/LAMBDA**2
C
C       SCALE FOR CURRENT ENERGY, I.E., CURRENT LAMBDA AND SELECT THAT
C       PORTION WHICH LIES BETWEEN (1 - COS) = 0 AND 2.
C
C-----CONVERT FROM ENERGY TO WAVELENGTH
        WAVE=E2WAVE/XF
        WAVE=WAVE*WAVE
        XFFUSE(1)=XFF(1)*WAVE
        YFFUSE(1)=YFF(1)
        DO 10 KUSE=2,NFF
        XFFUSE(KUSE)=XFF(KUSE)*WAVE
        YFFUSE(KUSE)=YFF(KUSE)
        IF(XFFUSE(KUSE)-TWO) 10,20,20
```

```
      10 CONTINUE
C
C     FORM FACTOR DOES NOT SPAN THE ENERGY OF INTEREST. IN THIS CASE THE
C     FORM FACTOR NEAR COS = -1 WILL BE SO SMALL THAT IT MAY BE
C     CONSIDERED TO BE ZERO AND WE NEED ONLY INTEGRATE OVER THE
C     TABULATED COS RANGE, WHICH IS EXACTLY EQUIVALENT TO SETTING THE
C     FORM FACTOR EQUAL TO ZERO OVER THE REMAINING COS RANGE.
C
      KUSE=NFF
      GO TO 30
C
C     TRUNCATE FORM FACTOR TO 1 - COS = 0 TO 2
C
C-----TRUNCATE DISTRIBUTION AT 2.0 ASSUMING LINEAR INTERPOLATION.
      20 WT1=(TWO-XFFUSE(KUSE-1))/(XFFUSE(KUSE)-XFFUSE(KUSE-1))
         WT2=ONE-WT1
         YFFUSE(KUSE)=WT1*YFFUSE(KUSE)+WT2*YFFUSE(KUSE-1)
         XFFUSE(KUSE)=TWO
C
C     DEFINE LINEARLY INTERPOLABLE ANGULAR DISTRIBUTION AND INTEGRATE.
C
      30 SUMZ=ZERO
         ZNOW=YFFUSE(1)
         XLAST=XFFUSE(1)
         FLAST=YFFUSE(1)
         YLAST=COHER1(FLAST,ZNOW,F1,F2,ITYPE,POWER)
         DO 70 N=2,KUSE
C-----DEFINE VALUE AT END OF INTERVAL AND SAVE.
         XEND=XFFUSE(N)
         FEND=YFFUSE(N)
         YEND=COHER1(FEND,ZNOW,F1,F2,ITYPE,POWER)
C
C     ITERATE TO CONVERGENCE.
C
C-----RE-DEFINE VALUE AT END OF INTERVAL.
      40 XNOW=XEND
         FNOW=FEND
         YNOW=YEND
         DO 50 LOOP=1,10
         XMID=(XNOW+XLAST)/TWO
         FMID=(FNOW+FLAST)/TWO
         YMID=COHER1(FMID,ZNOW,F1,F2,ITYPE,POWER)
C-----USE LINEAR INTERPOLATION AND CHECK FOR CONVERGENCE.
         YLIN=(YNOW+YLAST)/TWO
         IF(QABS(YMID-YLIN).LE.QABS(ERROK*YMID)) GO TO 60
C-----NO CONVERGENCE. SHORTEN INTERVAL AND CONTINUE ITERATING.
         XNOW=XMID
         FNOW=FMID
      50 YNOW=YMID
         LOOP=10
C-----CONVERGENCE. ADD CONTRIBUTION TO INTEGRAL.
      60 CNOW=ONE-XNOW
```

```
          CLAST=ONE-XLAST
          DC=(XNOW-XLAST)/TWO
          AVC=(CNOW+CLAST)/TWO
          DY=(YNOW-YLAST)/TWO
          AVY=(YNOW+YLAST)/TWO
          AVSUM=DC*AVY*(THREE+THREE*AVC*AVC+DC*DC)
          DSUM=TWO*DY*DC*DC*AVC
          SUMZ=SUMZ+(AVSUM+DSUM)
C-----SAVE END OF INTERVAL AS START OF NEXT INTERVAL.
          XLAST=XNOW
          FLAST=FNOW
          YLAST=YNOW
C-----USE SAME INTERVAL IF NOT YET FINISHED.
          IF(LOOP.GT.1) GO TO 40
      70 CONTINUE
C
C      INTEGRAL IS COMPLETED. NORMALIZE RESULT.
C
          SUM=TWO*TOM38*SUMZ/THREE
          RETURN
C***** DOUBLE *****
          END
          DOUBLE PRECISION FUNCTION COHER1(YNOW,ZNOW,F1,F2,ITYPE,POWER)
C***** DOUBLE *****
C***** SINGLE *****
C      END
C      FUNCTION COHER1(YNOW,ZNOW,F1,F2,ITYPE,POWER)
C***** SINGLE *****
C
C      CALCULATE ANGULAR DISTRIBUTION OF COHERENTLY SCATTERED PHOTON AT
C      ONE ANGLE.
C
C      WARNING THIS ROUTINE DOES NOT MULTIPLY BY 1+COS**2...IT ONLY
C      CALCULATE THE CONTRIBUTION OF THE FORM FACTOR AND SCATTERING
C      FACTORS.
C
C      ARGUMENTS
C      =========
C      YNOW     = FORM FACTOR AT XNOW
C      ZNOW     = FORM FACTOR AT 0
C      F1       = FORWARD ANGLE REAL SCATTERING FACTOR
C      F2       = FORWARD ANGLE IMAGINARY SCATTERING FACTOR
C      ITYPE    = 1 - WITH ANOMALOUS.
C               = 2 - WITHOUT ANOMALOUS
C      POWER    = ANISOTROPY OF SCATTERING FACTORS (POWER OF FORM FACTOR)
C
C***** SAVE ******
C      SAVE
C***** SAVE ******
C***** DOUBLE *****
          IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
```

```
          IF(ITYPE.EQ.1) GO TO 10
C-----WITHOUT ANOMALOUS.
          ANOW=YNOW*YNOW
          GO TO 40
C-----WITH ANOMALOUS - WITH OR WITHOUT ANISOTROPY.
   10 IF(POWER) 30,30,20
C-----WITH ANISOTROPY.
   20 QNOW=YNOW/ZNOW
          QNOW=QNOW**POWER
          ANOW=(YNOW+F1*QNOW)**2+(F2*QNOW)**2
          GO TO 40
C-----WITHOUT ANISOTROPY.
   30 ANOW=(YNOW+F1)**2+(F2)**2
   40 COHER1=ANOW
          RETURN
          END
          SUBROUTINE EDGEIT
C
C     MAKE SURE CROSS SECTION STARTS AT EDGE
C
C     THIS ROUTINE ASSUMES THE CROSS SECTIONS HAVE BEEN READ AND ARE
C     NOW IN X AND Y IN LOG-LOG INTERPOLABLE FORM.
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
          IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
          COMMON/PHOTO1/X(10000),XT(10000),Y(10000),YT(10000),IPT,NPT

COMMON/CONJOB/R0,FOURPI,TOMSON,TOM38,XHIGH,EDGES,DEDGE,DEDGE2,UP,
      1 DOWN,DEDG10
          COMMON/ERANGE/XMIN,XMAX
C***** DOUBLE *****
          DATA ONE/1.0D+00/
          QLOG(X)=DLOG(X)
          QEXP(X)=DEXP(X)
C***** DOUBLE *****
C***** SINGLE *****
C     DATA ONE/1.0E+00/
C     QLOG(X)=ALOG(X)
C     QEXP(X)=EXP(X)
C***** SINGLE *****
C-----COMPARE FIRST POINT TO EDGE.
          IF(X(1)-EDGES) 10,110,70
C
C     TABLE STARTS BELOW EDGE.
C
C-----REMOVE ALL POINTS BELOW EDGE.
   10 II=2
   20 IF(X(II)-EDGES) 30,50,40
```

```fortran
   30 II=II+1
      GO TO 20
C-----DEFINE CROSS SECTION AT EDGE BY LOG-LOG INTERPOLATION FROM
C-----SURROUNDING POINTS.
   40 WT1=QLOG(EDGES/X(II))/QLOG(X(II-1)/X(II))
      WT2=ONE-WT1
      Y(II-1)=QEXP(WT1*QLOG(Y(II-1))+WT2*QLOG(Y(II)))
      X(II-1)=EDGES
      II=II-1
C-----MOVE REMAINING POINTS FORWARD IN TABLE AND DEFINE NEW POINT
C-----COUNT.
   50 IF(II.LE.1) GO TO 110
      KK=0
      DO 60 J=II,IPT
      KK=KK+1
      X(KK)=X(J)
   60 Y(KK)=Y(J)
      IPT=KK
      GO TO 110
C
C     TABLE STARTS ABOVE EDGE.
C
C-----EXTEND DOWN TO EDGE AS 1/E**3
   70 II=IPT
      DO 80 I=1,IPT
      X(II+1)=X(II)
      Y(II+1)=Y(II)
   80 II=II-1
      X(1)=EDGES
C-----1/E**3 EXTENSION.
      RATIOE=X(2)/X(1)
      Y(1)=Y(2)*(RATIOE**3)
      IPT=IPT+1
  110 RETURN
      END
      SUBROUTINE PLTTAB(IUN,X,Y,KPT)
C
C     OUTPUT A TABLE IN THE PLOTTAB FORMAT.
C
C     ARGUMENTS
C     =========
C     IUN     = I/O UNIT NUMBER
C     X       = X VALUES
C     Y       = Y VALUES
C     KPT     = NUMBER OF (X,Y) PAIRS TO OUTPUT
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
```

```
C***** CHARACTER *****
      CHARACTER*1 KSIGN
C***** CHARACTER *****
      COMMON/NORMZF/XNORM(6),KEXP(6)
      COMMON/NORMZC/KSIGN(6)
      DIMENSION X(10000),Y(10000)
C-----OUTPUT POINTS.
      DO 10 K=1,KPT
      CALL NORMX(X(K),XNORM(1),KSIGN(1),KEXP(1))
      CALL NORMX(Y(K),XNORM(2),KSIGN(2),KEXP(2))
   10 WRITE(IUN,1000) (XNORM(L),KSIGN(L),KEXP(L),L=1,2)
C-----OUTPUT END OF CURVE SENTINAL.
      WRITE(IUN,1010)
      RETURN
 1000 FORMAT(6(F8.5,A1,I2))
 1010 FORMAT(30X,'(BLANK LINE TERMINATES CURVE)')
      END
      SUBROUTINE NORMX(X,XNORM,KSIGN,KXPLUS)
C
C     CONVERT NUMBER TO NORMAL FORM FOR OUTPUT.
C
C     OUTPUT WILL BE IN THE FORM X.XXXXX+/-NN, WHICH GIVES
C     AN ADDITIONAL DIGIT OF ACCURACY WHEN COMPARED TO E11.4
C     FORTRAN OUTPUT. IN ADDITION THE OUTPUT WILL BE COMPUTER
C     INDEPENDENT (I.E. SAME ON IBM, CDC, UNIVAC ETC.)
C
C     ARGUMENTS
C     =========
C     X      = FLOATING POINT NUMBER TO PUT IN NORMAL FORM
C     XNORM  = NORMAL MANTISSA - SHOULD BE OUTPUT IN F8.5 FORM
C     KSIGN  = SIGN OF EXPONENT - SHOULD BE OUTPUT IN A1 FORM
C     KEXP   = ABSOLUTE VALUE OF EXPONENT - SHOULD BE OUTPUT IN
C               I2 FORM
C
C     FOR EXAMPLE, IF X=1.2345678D-04, THEN XNORM, KSIGN AND KEXP
C     WILL BE RETURNED AS 1.23457, '-' AND 4, RESPECTIVELY. IF OUTPUT
C     USING F8.5, A1,I2 FORMATS THIS WILL APPEAR AS 1.23457- 4
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE ******
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE ******
C***** CHARACTER *******
      CHARACTER*1 PLUS,MINUS,KSIGN
C***** CHARACTER *******
C***** INTEGER ********
C     INTEGER PLUS
C***** INTEGER ********
C-----DEFINE TWO POSSIBLE SIGNS FOR EXPONENT.
      DATA PLUS/'+'/
```

```
      DATA MINUS/'-'/
C***** DOUBLE ******
      DATA ZERO/0.0D+00/
      DATA ONE/1.0D+00/
      DATA TEN/1.0D+01/
      DATA XLOW/1.0D+00/
      DATA XTOP/9.999995D+00/
      QABS(ZD)=DABS(ZD)
      QLOG10(ZD)=DLOG10(ZD)
C***** DOUBLE ******
C***** SINGLE ******
C     DATA ZERO/0.0E+00/
C     DATA ONE/1.0E+00/
C     DATA TEN/1.0E+01/
C     DATA XLOW/1.0E+00/
C     DATA XTOP/9.999995E+00/
C     QABS(ZD)=ABS(ZD)
C     QLOG10(ZD)=ALOG10(ZD)
C***** SINGLE ******
C-----COMPUTE SIGNED EXPONENT FROM ABSOLUTE VALUE.
      XABS=QABS(X)
C-----IF NUMBER IS 0.0 DEFINE 0.00000+ 0
      IF(XABS) 50,50,10
C-----DEFINE EXPONENT TO NORMALIZE MANTISSA.
   10 KXREAL=QLOG10(XABS)
      IF(XABS.LT.ONE) KXREAL=KXREAL-1
      SHIFT=TEN**KXREAL
      XNORM=XABS/SHIFT
      IF(XNORM.GE.XLOW) GO TO 20
      XNORM=ONE
      GO TO 30
   20 IF(XNORM.LT.XTOP) GO TO 30
      KXREAL=KXREAL+1
      XNORM=ONE
C-----COMPUTE SIGNED MANTISSA.
   30 IF(X.LT.ZERO) XNORM=-XNORM
C-----DEFINE SIGN OF EXPONENT AND INSURE EXPONENT IS POSITIVE.
      IF(KXREAL) 40,60,70
   40 KSIGN=MINUS
      KXPLUS=-KXREAL
      RETURN
C-----X IS 0.0. DEFINE AS 0.00000+ 0
   50 XNORM=ZERO
   60 KXREAL=0
C-----EXPONENT IS ZERO OR POSITIVE. THEREFORE SIGN IS POSITIVE.
   70 KSIGN=PLUS
      KXPLUS=KXREAL
      RETURN
      END
      SUBROUTINE THIN3L(X,Y,NPT,NDIM,KDIM,ERROK)
C
C     THIN TABLE OF MULTIPLE CURVES
```

```
C      ==============================
C
C      A NUMBER OF CURVES (KDIM), EACH DEFINED BY THE SAME NUMBER OF
C      POINTS (NDIM), ALL CURVES GIVEN AT THE SAME X VALUES. ASSUMING
C      THE DATA IS LINEARLY INTERPOLABLE ALONG EACH CURVE REMOVE ALL
C      DATA POINTS THAT ARE NOT REQUIRED TO ACCURATELY INTERPOLATE
C      FROM SURROUNDING POINTS.
C
C      SEE, SUBROUTINE THEN3G TO THIN NON-LINEARLY INTERPOLABLE DATA.
C
C      ARGUMENTS
C      =========
C      X      - X VALUES
C      Y      - Y VALUES
C      NPT    - ORIGINAL AND FINAL NUMBER OF POINTS
C      NDIM   - FIRST DIMENSION (CONTAINS EACH CURVE)
C      KDIM   - SECOND DIMENSION (NUMBER CURVES)
C      ERROK  - ALLOWABLE FRACTIONAL THINNING CRITERIA
C
C***** SAVE ******
C      SAVE
C***** SAVE ******
C***** DOUBLE *****
       IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
       DIMENSION X(NDIM),Y(NDIM,KDIM)
C***** DOUBLE *****
       DATA ONE/1.0D+00/
       DATA ZERO/0.0D+00/
       QABS(Z)=DABS(Z)
C***** DOUBLE *****
C***** SINGLE *****
C      DATA ZERO/0.0D+00/
C      DATA ONE/1.0E+00/
C      QABS(Z)=ABS(Z)
C***** SINGLE *****
C-----CANNOT THIN IF LESS THAN 3 POINTS OR THINNING CRITERIA IS
C-----NOT POSITIVE.
       IF(NPT.LT.3.OR.ERROK.LE.ZERO) RETURN
C
C      REMOVE DUPLICATE POINTS.
C
       KPT=1
       DO 50 I=2,NPT
       IF(X(I)-X(KPT)) 50,10,30
C-----DUPLICATE X. KEEP POINT IF DIFFERENT Y. OTHERWISE IGNORE.
   10 DO 20 K=1,KDIM
       IF(Y(I,K)-Y(KPT,K)) 30,20,30
   20 CONTINUE
       GO TO 50
   30 KPT=KPT+1
       X(KPT)=X(I)
```

```
      DO 40 K=1,KDIM
   40 Y(KPT,K)=Y(I,K)
   50 CONTINUE
      NPT=KPT
      IF(NPT.LT.3) RETURN
C
C     BEGIN THINNING.
C
C-----INITIALIZE THINNING INDICES.
      KPT=1
      IPT1=2
C-----START OR CONVERGENCE IN LAST INTERVAL. RESET INDICES.
   60 IPT2=IPT1
   70 IPT3=IPT2+1
C-----CHECK FOR END OF TABLE.
      IF(IPT3-NPT) 80,80,130
C-----CHECK FOR DISCONTINUITY IN INTERPOLATION INTERVAL.
   80 DXNOW=X(IPT3)-X(KPT)
      IF(DXNOW) 110,110,90
C-----SEE IF ALL POINTS BETWEEN KPT AND IPT3 CAN BE ACCURATELY
C-----APPROXIMATED BY INTERPOLATION.
   90 DO 100 I=IPT1,IPT2
      WT1=(X(I)-X(KPT))/DXNOW
      WT2=ONE-WT1
      DO 100 K=1,KDIM
      YP=WT1*Y(IPT3,K)+WT2*Y(KPT,K)
      IF(QABS(YP-Y(I,K)).GT.QABS(ERROK*Y(I,K))) GO TO 110
  100 CONTINUE
C-----ALL POINTS CAN BE ACCURATELY INTERPOLATED. EXTEND THINNING
C-----INTERVAL BY 1 AND CONTINUE, UNLESS THE END OF THE TABLE
C-----HAS BEEN REACHED.
      IPT2=IPT3
      IF(IPT2-NPT) 70,130,130
C-----CANNOT ACCURATELY INTERPOLATE. SAVE LAST POINT WITHIN
C-----INTERVAL (IPT2). NOTE, INTERPOLATION BETWEEN (KPT) AND (IPT2)
C-----HAS ALREADY BEEN DEMONSTRATED TO ACCURATELY APPROXIMATE ALL OF
C-----THE POINTS BETWEEN THEM (IF ANY), WHEREAS INTERPOLATION BETWEEN
C-----(KPT) AND (IPT3) HAS BEEN DEMONSTRATED NOT TO ACCURATELY
C-----APPROXIMATE THE POINTS BETWEEN THEM. THEREFORE IT IS (IPT2), NOT
C-----(IPT3) WHICH MUST BE SAVED AT THIS POINT..
  110 KPT=KPT+1
      X(KPT)=X(IPT2)
      DO 120 K=1,KDIM
  120 Y(KPT,K)=Y(IPT2,K)
C-----RESET LOWER THINNING INDEX TO START THINNING WITH FIRST POINT
C-----PAST (IPT2) AND RESTART THINNING UNLESS THE END OF THE TABLE
C-----HAS BEEN REACHED.
      IPT1=IPT2+1
      IF(IPT2-NPT) 60,150,150
C-----SAVE LAST POINT.
  130 KPT=KPT+1
      X(KPT)=X(NPT)
```

```
      DO 140 K=1,KDIM
  140 Y(KPT,K)=Y(NPT,K)
C-----DEFINE NUMBER OF THINNED POINTS.
  150 NPT=KPT
      RETURN
      END
      SUBROUTINE OUTL1(IUN,ZA,YI,YO,ATW,DATE,INT,FLAG,ELEVL,HAFLYF,TMP)
C
C     OUTPUT FIRST CARD OF A SECTION
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
C***** CHARACTER *****
      CHARACTER*1 KSIGN
C***** CHARACTER *****
      INTEGER ZA,YI,YO,DATE,FLAG
      COMMON/NORMZF/XNORM(6),KEXP(6)
      COMMON/NORMZC/KSIGN(6)
      CALL NORMX(ATW,XNORM(1),KSIGN(1),KEXP(1))
      CALL NORMX(ELEVL,XNORM(2),KSIGN(2),KEXP(2))
      CALL NORMX(HAFLYF,XNORM(3),KSIGN(3),KEXP(3))
      CALL NORMX(TMP,XNORM(4),KSIGN(4),KEXP(4))
      WRITE(IUN,1000) ZA,YI,YO,XNORM(1),KSIGN(1),KEXP(1),DATE,INT,FLAG,
     1 (XNORM(L),KSIGN(L),KEXP(L),L=2,4)
      RETURN
 1000 FORMAT(I6,2I3,1X,F8.5,A1,I2,I7,I1,I2,3(1X,F8.5,A1,I2))
      END
      SUBROUTINE OUTL2(IUN,C,I,S,Q,XI)
C
C     OUTPUT SECOND LINE OF SECTION
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
C***** CHARACTER *****
      CHARACTER*1 KSIGN
C***** CHARACTER *****
      INTEGER C,S
      COMMON/NORMZF/XNORM(6),KEXP(6)
      COMMON/NORMZC/KSIGN(6)
      DIMENSION XI(4)
      CALL NORMX(Q,XNORM(1),KSIGN(1),KEXP(1))
      DO 10 L=2,5
   10 CALL NORMX(XI(L-1),XNORM(L),KSIGN(L),KEXP(L))
      WRITE(IUN,1000) C,I,S,(XNORM(L),KSIGN(L),KEXP(L),L=1,5)
```

```
         RETURN
  1000 FORMAT(I2,2I3,5(1X,F8.5,A1,I2))
         END
         SUBROUTINE OUTAB2(IUN,X1,X2,KPT)
C
C      OUTPUT A TABLE OF TWO FLOATING POINT WORDS PER CARD.
C      NOTE...THE TABLE IS FOLLOWED BY AN END OF SECTION FLAG.
C
C***** SAVE ******
C      SAVE
C***** SAVE ******
C***** DOUBLE *****
         IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
C***** CHARACTER *****
         CHARACTER*1 KSIGN
C***** CHARACTER *****
         COMMON/NORMZF/XNORM(6),KEXP(6)
         COMMON/NORMZC/KSIGN(6)
         DIMENSION X1(10000),X2(10000)
         DO 10 K=1,KPT
         CALL NORMX(X1(K),XNORM(1),KSIGN(1),KEXP(1))
         CALL NORMX(X2(K),XNORM(2),KSIGN(2),KEXP(2))
     10 WRITE(IUN,1000) (XNORM(L),KSIGN(L),KEXP(L),L=1,2)
         WRITE(IUN,1010)
         RETURN
  1000 FORMAT(2(F8.5,A1,I2))
  1010 FORMAT(71X,'1')
         END
         SUBROUTINE FUZZY(XTAB,YTAB,NPT,XFTAB,YFTAB,IFUZZ,DEDGE,ERROK,
        1 XMAX)
C
C      PURPOSE
C      =======
C      RESOLUTION SMOOTH CROSS SECTION FROM EDGE UP TO XMAX.
C
C      ARGUMENTS
C      =========
C      XTAB      = ORIGINAL ENERGIES
C      YTAB      = ORIGINAL CROSS SECTIONS
C      NPT       = NUMBER OF ORIGINAL POINTS
C      XFTAB     = ENERGIES FOR BROADENED DATA
C      YFTAB     = CROSS SECTIONS FOR BROADENED DATA
C      IFUZZ     = NUMBER OF BROADENED POINTS
C      DEDGE     = ENERGY RESOLUTION - FRACTION OF ENERGY
C      ERROK     = ALLOWABLE UNCERTAINTY IN LINEARLY INTERPOLABLE
C                  RESULTS
C      XMAX      = MAXIMUM ENERGY TO BROADEN UP TO - COPY ALL POINTS
C                  ABOVE THIS ENERGY
C
C      METHOD
C      ======
```

```
C     SEE COMMENTS AT BEGINNING OF THIS PROGRAM
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
      DIMENSION XTAB(1000),YTAB(1000),XFTAB(5000),YFTAB(5000)
C***** DOUBLE *****
      DATA ZERO/0.0D+00/
      DATA TENTH/1.0D-01/
      DATA ONE/1.0D+00/
      DATA TWO/2.0D+00/
      DATA WIDEND/4.0D+00/
      QABS(Q)=DABS(Q)
C***** DOUBLE *****
C***** SINGLE *****
C     DATA ZERO/0.0E+00/
C     DATA TENTH/1.0E-01/
C     DATA ONE/1.0E+00/
C     DATA TWO/2.0E+00/
C     DATA WIDEND/4.0E+00/
C     QABS(Q)=ABS(Q)
C***** SINGLE *****
C
C     DEFINE OVERALL WIDTH AND STEP FOR FIRST EDGE
C
      DWIDE=WIDEND*DEDGE*XTAB(1)
      XLOW=XTAB(1)-DWIDE
C
C     CROSS SECTION IS ZERO AT FIRST POINT
C
      IFUZZ=1
      XLAST=XLOW
      YLAST=ZERO
      XFTAB(IFUZZ)=XLAST
      YFTAB(IFUZZ)=YLAST
C
C     INTERVAL HALF TO CONVERGENCE UP TO XMAX. COPY ABOVE XMAX
C
      DO 40 KPT=1,NPT
C
C     UPPER END OF INTERVAL
C
      XEND=XTAB(KPT)
      CALL FUZZIT(XEND,YEND,DEDGE,XTAB,YTAB,NPT)
C
C     OUTER ITERATION LOOP
C
   10 XNOW=XEND
      YNOW=YEND
```

```
C
C      INNER ITERATION LOOP
C
       DO 20 LOOPI=1,10
       XMID=(XNOW+XLAST)/TWO
       YLIN=(YNOW+YLAST)/TWO
       CALL FUZZIT(XMID,YMID,DEDGE,XTAB,YTAB,NPT)
       IF(QABS(YMID-YLIN).LE.QABS(ERROK*YMID)) GO TO 30
       XNOW=XMID
    20 YNOW=YMID
       LOOPI=11
C
C      CONVERGENCE. SAVE MIDPOINT (TO ALLOW LATER THINNING)
C
    30 IFUZZ=IFUZZ+1
       IF(IFUZZ.GT.10000) GO TO 140
       XFTAB(IFUZZ)=XMID
       YFTAB(IFUZZ)=YMID
C
C      IF NOT YET END OF INTERVAL CONTINUE.
C
       XLAST=XMID
       YLAST=YMID
       IF(LOOPI.GT.1) GO TO 10
C
C      END OF INTERVAL
C
       IFUZZ=IFUZZ+1
       IF(IFUZZ.GT.10000) GO TO 140
       XFTAB(IFUZZ)=XEND
       YFTAB(IFUZZ)=YEND
       XLAST=XEND
       YLAST=YEND
       IF(XEND.GT.XMAX) GO TO 100
    40 CONTINUE
C
C      DATA DOES NOT EXTEND UP TO XMAX. EXTEND IT.
C
C-----EXTEND TO XHIGH AS BROADENED.
       DWIDE=WIDEND*DEDGE*XTAB(NPT)
       XHIGH=XTAB(NPT)+DWIDE
       IF(XHIGH.GT.XMAX) XHIGH=XMAX
       XNEXT=ONE+TENTH*DEDGE
    50 XEND=XFTAB(IFUZZ)*XNEXT
       CALL FUZZIT(XEND,YEND,DEDGE,XTAB,YTAB,NPT)
    60 XNOW=XEND
       YNOW=YEND
       DO 70 LOOPI=1,10
       XMID=(XNOW+XLAST)/TWO
       YLIN=(YNOW+YLAST)/TWO
       CALL FUZZIT(XMID,YMID,DEDGE,XTAB,YTAB,NPT)
       IF(QABS(YMID-YLIN).LE.QABS(ERROK*YMID)) GO TO 80
```

```
            XNOW=XMID
      70 YNOW=YMID
            LOOPI=11
      80 IFUZZ=IFUZZ+1
            IF(IFUZZ.GT.10000) GO TO 140
            XFTAB(IFUZZ)=XMID
            YFTAB(IFUZZ)=YMID
            XLAST=XMID
            YLAST=YMID
            IF(LOOPI.GT.1) GO TO 60
            IFUZZ=IFUZZ+1
            IF(IFUZZ.GT.10000) GO TO 140
            XFTAB(IFUZZ)=XEND
            YFTAB(IFUZZ)=YEND
            XLAST=XEND
            YLAST=YEND
            IF(XEND.LT.XHIGH) GO TO 50
C-----ADD ZERO POINT AND THIN.
            XEND=XFTAB(IFUZZ)*XNEXT
            IFUZZ=IFUZZ+1
            IF(IFUZZ.GT.10000) GO TO 140
            XFTAB(IFUZZ)=XEND
            YFTAB(IFUZZ)=ZERO
            CALL THIN3L(XFTAB,YFTAB,IFUZZ,10000,1,ERROK)
C-----EXTEND TO XMAX AS ZERO.
      90 XEND=TWO*XEND
            IFUZZ=IFUZZ+1
            IF(IFUZZ.GT.10000) GO TO 140
            XFTAB(IFUZZ)=XEND
            YFTAB(IFUZZ)=ZERO
            IF(XEND.LT.XMAX) GO TO 90
            RETURN
C
C     BROADENED UP TO XMAX. THIN.
C
     100 CALL THIN3L(XFTAB,YFTAB,IFUZZ,10000,1,ERROK)
C
C     COPY REMAINING POINTS
C
            DO 110 I=1,NPT
            IF(XTAB(I).GT.XFTAB(IFUZZ)) GO TO 120
     110 CONTINUE
            RETURN
     120 XSPACE=ONE+DEDGE
            DO 130 K=I,NPT
            XNOW=XTAB(K)
C-----SKIP POINTS WHICH ARE CLOSER THAN ENERGY RESOLUTION
            IF(XNOW.LE.XSPACE*XFTAB(IFUZZ)) GO TO 130
            IFUZZ=IFUZZ+1
            IF(IFUZZ.GT.10000) GO TO 140
            XFTAB(IFUZZ)=XNOW
            YFTAB(IFUZZ)=YTAB(K)
```

```
   130 CONTINUE
       RETURN
C-----MEMORY EXCEEDED.
   140 WRITE(6,6000)
       STOP
  6000 FORMAT(' MEMORY FOR RESOLUTION SMOOTHING EXCEEDED'/
      1 ' EXECUTION TERMINATED')
       END
       SUBROUTINE FUZZIT(E,Y,DELTA,ETAB,YTAB,NPT)
C
C      PURPOSE
C      =======
C      INTEGRATION TO SMOOTH CROSS SECTIONS AT ENERGY E
C
C      ARGUMENTS
C      =========
C      E       = ENERGY AT WHICH TO DEFINE SMOOTH CROSS SECTION
C      Y       = SMOOTH CROSS SECTION
C      DELTA   = RESOLUTION - FRACTION OF ENERGY
C      ETAB    = ORIGINAL ENERGIES
C      YTAB    = ORIGINAL CROSS SECTIONS
C      NPT     = NUMBER OF ORIGINAL POINTS
C
C      METHOD
C      ======
C      CALCULATE THE INTEGRAL,
C
C      INTEGRAL SIGMA(E')*EXP(-ALPHA*(E'-E)**2)*DE'
C      --------------------------------------------
C      INTEGRAL          EXP(-ALPHA*(E'-E)**2)*DE'
C
C      WHERE ALPHA IS DEFINED BASED ON THE ENERGY RESOLUTION
C
C      EXP(-ALPHA*(E'-E)**2) = 1/2, FOR E'-E = DELTA*E
C
C      ALPHA*(DELTA*E)**2 = LOG(2)
C
C      ALPHA = LOG(2)/(DELTA*E)**2
C
C      AND SIGMA(E') IS GIVEN IN TABULATED LINEARLY INTERPOLABLE FORM,
C
C      SIGMA(E') = DYDX*(E'-<E>) + <SIGMA>,     E' BETWEEN E1 AND E2
C
C      DYDX     = (SIGMA(E2) - SIGMA(E1))/(E2 - E1)
C      <E>      = (E2 + E1)/2
C      <SIGMA>  = (SIGMA(E2) + SIGMA(E1))/2
C
C      (DYDX*(E'-<E>) + <SIGMA>)*EXP(-ALPHA*(E'-E)**2)*DE'
C      ---------------------------------------------------
C               EXP(-ALPHA*(E'-E)**2)*DE'
C
C      DYDX*((E'-E) + (E-<E>)) + <SIGMA>)*EXP(-ALPHA*(E'-E)**2)*DE'
```

```
C      ----------------------------------------------------------
C               EXP(-ALPHA*(E'-E)**2)*DE'
C
C      X'  = SQRT(ALPHA)*(E'-E)
C      DX' = SQRT(ALPHA)*DE'
C
C      DYDX*(X'/SQRT(ALPHA) + (E-<E>)) + <SIGMA>)*EXP(-X'**2)*DX'
C      ----------------------------------------------------------
C                    EXP(-X'**2)*DX'
C
C      FA = INTEGRAL EXP(-X'**2)*DX'    = SQRT(PI)*(ERF(X1) - ERF(X2))/2
C      FB = INTEGRAL X'*EXP(-X'**2)*DX' = (EXP(-X1**2) - EXP(-X2**2))/2
C
C      <SIGMA>*FA + DYDX*(FB/SQRT(ALPHA) + (E-<E>)*FA)
C      ----------------------------------------------
C          SQRT(PI)*(ERF(-XFUZZ)-ERF(XFUZZ))
C
C      THE INTEGRAL IS TRUNCATED TO THE RANGE XFUZZ WHERE,
C      XFUZZ = +4 = SQRT(ALPHA)*(EMAX-E)
C
C      EMIN = E - 4/SQRT(ALPHA)
C      EMAX = E + 4/SQRT(ALPHA)
C
C      ALPHA = LOG(2)/(DELTA*E)**2
C
C***** SAVE ******
C      SAVE
C***** SAVE ******
C***** DOUBLE *****
       IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
       DIMENSION ETAB(NPT),YTAB(NPT)
C***** DOUBLE *****
       DATA ZERO/0.0D+00/
       DATA ONE/1.0D+00/
       DATA TWO/2.0D+00/
       DATA XFUZZ/4.0D+00/
       DATA P/0.3275911D+00/
       DATA A1/0.254829592D+00/
       DATA A2/-0.284496736D+00/
       DATA A3/1.421413741D+00/
       DATA A4/-1.453152027D+00/
       DATA A5/1.061405429D+00/
       QABS(Q)=DABS(Q)
       QLOG(Q)=DLOG(Q)
       QSQRT(Q)=DSQRT(Q)
       QACOS(Q)=DACOS(Q)
C***** DOUBLE *****
C***** SINGLE *****
C      DATA ZERO/0.0E+00/
C      DATA ONE/1.0E+00/
C      DATA TWO/2.0E+00/
```

```
C       DATA XFUZZ/4.0E+00/
C       DATA P/0.3275911E+00/
C       DATA A1/0.254829592E+00/
C       DATA A2/-0.284496736E+00/
C       DATA A3/1.421413741E+00/
C       DATA A4/-1.453152027E+00/
C       DATA A5/1.061405429E+00/
C       QABS(Q)=ABS(Q)
C       QLOG(Q)=ALOG(Q)
C       QSQRT(Q)=SQRT(Q)
C       QACOS(Q)=ACOS(Q)
C***** SINGLE *****
C-----DEFINE CONSTANTS
        PI=QACOS(-ONE)
        SQRTAL=QSQRT(QLOG(TWO))/(DELTA*E)
        PISQRT=DSQRT(PI)
        EMIN=E-XFUZZ/SQRTAL
        EMAX=E+XFUZZ/SQRTAL
C-----DEFINE NORMALIZATION FACTOR
        EXPMAX=DEXP(-XFUZZ*XFUZZ)
        T=ONE/(ONE+P*XFUZZ)
        SUM0=TWO*PISQRT*(ONE-((((A5*T+A4)*T+A3)*T+A2)*T+A1)*T*EXPMAX)
C-----INITIALIZE INTEGRAL
        SUM1=ZERO
        SUM2=ZERO
C
C       DEFINE START OF INTEGRAL
C
        IF(ETAB(1)-EMIN) 20,20,10
C-----TABULATED DATA STARTS ABOVE EMIN. START INTEGRAL AT FIRST
C-----TABULATED POINT.
   10 ELAST=ETAB(1)
        YLAST=YTAB(1)
        N=2
        GO TO 70
C-----TABULATED DATA STARTS AT OR BELOW EMIN. USE TABLE TO DEFINE
C-----CROSS SECTION AT EMIN.
   20 DO 30 N=1,NPT
        IF(ETAB(N)-EMIN) 30,50,40
   30 CONTINUE
        STOP
C-----INTERPOLATE TO EMIN.
   40 WT1=(EMIN-ETAB(N-1))/(ETAB(N)-ETAB(N-1))
        WT2=ONE-WT1
        YLAST=WT1*YTAB(N)+WT2*YTAB(N-1)
        GO TO 60
C-----USE TABULATED VALUE
   50 YLAST=YTAB(N)
   60 ELAST=EMIN
C-----DEFINE EXP AND ERF AT START
   70 XLAST=SQRTAL*(ELAST-E)
        XA=QABS(XLAST)
```

```
      EXPLST=DEXP(-XA*XA)
      T=ONE/(ONE+P*XA)
      XAERF=ONE-((((A5*T+A4)*T+A3)*T+A2)*T+A1)*T*EXPLST
      IF(XLAST.LT.ZERO) XAERF=-XAERF
      ERFLST=XAERF
C
C     ADD THE CONTRIBUTION OF ALL INTERVALS
C
      IEND=0
      DO 140 K=N,NPT
      IF(ETAB(K)-EMAX) 80,100,90
C-----STILL IN RANGE OF INTEGRATION. USE ENTIRE INTERVAL.
   80 ENOW=ETAB(K)
      YNOW=YTAB(K)
      GO TO 110
C-----INTERPOLATE TO UPPER LIMIT OF INTEGRATION
   90 WT1=(EMAX-ETAB(K-1))/(ETAB(K)-ETAB(K-1))
      WT2=ONE-WT1
      ENOW=EMAX
      YNOW=WT1*YTAB(K)+WT2*YTAB(K-1)
      IEND=1
      GO TO 110
C-----AT UPPER LIMIT OF INTEGRATION
  100 ENOW=EMAX
      YNOW=YTAB(K)
      IEND=1
C
C     UPPER END OF INTERVAL IS NOW DEFINED.
C
C-----SKIP ZERO LENGTH INTERVALS
  110 DE=ENOW-ELAST
      IF(DE.LE.ZERO) GO TO 130
      AVE=(ENOW+ELAST)/TWO
      AVY=(YNOW+YLAST)/TWO
      DYDX=(YNOW-YLAST)/DE
C-----DEFINE EXP AND ERF NOW
      XNOW=SQRTAL*(ENOW-E)
      XA=QABS(XNOW)
      EXPNOW=DEXP(-XA*XA)
      T=ONE/(ONE+P*XA)
      XAERF=ONE-((((A5*T+A4)*T+A3)*T+A2)*T+A1)*T*EXPNOW
      IF(XNOW.LT.ZERO) XAERF=-XAERF
      ERFNOW=XAERF
C-----ADD CONTRIBUTION TO INTEGRAL.
      FA=PISQRT*(ERFNOW-ERFLST)
      FB=EXPLST-EXPNOW
      ADD1=AVY*FA
      ADD2=DYDX*(FB/SQRTAL+(E-AVE)*FA)
C-----CHECK LIMITS
      IF(ADD1.LE.ZERO) GO TO 120
      IF(ADD2.GT.ADD1) ADD2=ADD1
      IF(ADD2.LT.-ADD1) ADD2=-ADD1
```

```
          SUM1=SUM1+ADD1
          SUM2=SUM2+ADD2
C-----EITHER END OF INTEGRAL OR NEXT INTERVAL
   120 IF(IEND.NE.0) GO TO 150
C-----END OF INTERVAL. SAVE END OF CURRENT INTERVAL AS START OF NEXT
C-----INTERVAL.
          EXPLST=EXPNOW
          ERFLST=ERFNOW
          XLAST=XNOW
          ELAST=ENOW
   130 YLAST=YNOW
   140 CONTINUE
C-----DEFINE RESULT AS RATIO
   150 Y=(SUM1+SUM2)/SUM0
          RETURN
          END
          SUBROUTINE EMERGN(X,Y,KPT,XTOT,YTOT,NTOT,XUSE,YUSE,NDIM,NMAX,
      1 INTERP)
C
C       PURPOSE
C       =======
C       GIVEN A NUMBER OF VECTORS, Y, ALL DEFINED AT THE SAME X VALUES
C       MERGE THESE VECTORS INTO THE VECTORS YTOT ALL DEFINED AT THE
C       SAME XTOT VALUES, E.G., DEFINE THE TOTAL CROSS SECTION BY SUMMING
C       ITS PARTS BY CALLING THIS ROUTINE WITH ONE REACTION (ONE VECTOR)
C       AT A TIME AND REPEATEDLY CALLING THIS ROUTINE. E.G., MERGE SUBSHEL
C       CONTRIBUTIONS TO THE REAL AND IMAGINARY SCATTERING FACTORS IN ORDE
C       TO DEFINE THESE QUANTITIES FOR AN ELEMENT - IN THIS CASE THE ROUTI
C       IS CALLED WITH 2 VECTORS (1=REAL, 2=IMAGINARY SCATTERING FACTORS)
C       BOTH GIVEN ON THE SAME ENERGY GRID - THIS ROUTINE WILL BE CALLED
C       ONCE FOR ALL SUBSHELLS UNTIL ALL CONTRIBUTING SUBSHELLS HAVE BEEN
C       INCLUDED.
C
C       USE
C       ===
C       (1) BEFORE CALLING THIS ROUTINE THE FIRST TIME SET NTOT=0
C       (2) EACH TIME THIS ROUTINE IS CALLED IT WILL MERGE (X,Y)
C            INTO (XTOT,YTOT) AT THE UNION OF ALL X VALUES THAT
C            APPEAR IN EITHER.
C       (3) BY REPEATEDLY CALLING THIS ROUTINE IT IS POSSIBLE TO
C            MERGE TOGETHER ANY NUMBER OF TABLES OF (X,Y) PAIRS,
C            E.G., DEFINE THE TOTAL CROSS SECTION AS THE SUM OF
C            ITS PARTS - ANY NUMBER OF PARTS.
C       (4) WHEN YOU HAVE MERGED TOGETHER ALL THE DATA THAT YOU
C            WANT THE RESULTS WILL BE IN (XTOT,YTOT) AND THE NUMBER
C            OF POINTS IS NTOT.
C       (5) YOU MAY THEN RESTART THE MERGE CYCLE BY AGAIN SETTING
C            NTOT=0 AND THEN CALLING THIS ROUTINE ANY NUMBER OF
C            TIMES.
C
C       OPTIMIZING USE
C       ==============
```

```
C        IF YOU ARE MERGING TOGETHER A NUMBER OF TABLES YOU CAN
C        OPTIMIZE THE RUNNING TIME OF THIS ROUTINE BY MERGING THE
C        SMALLEST TABLES FIRST, E.G., TO DEFINE THE TOTAL CROSS
C        SECTION MERGE THE ELASTIC, CAPTURE AND FISSION LAST.
C
C        ARGUMENTS
C        =========
C        X      = X VALUES TO MERGE
C        Y      = Y VALUES TO MERGE
C        KPT    = NUMBER OF (X,Y) PAIRS TO MERGE
C        XTOT   = MERGED X VALUES
C        YTOT   = MERGED Y VALUES
C        NTOT   = NUMBER OF MERGED (XTOT,YTOT) PAIRS
C        XUSE   = SCRATCH X ARRAY
C        YUSE   = SCRATCH Y ARRAY
C        NDIM   = NUMBER OF Y, YTOT AND YUSE VECTORS, I.E., SECOND
C                 DIMENSION OF X, Y, XTOT, YTOT, XUSE, YUSE.
C        NMAX   = MAXIMUM ALLOWABLE MERGED TABLE SIZE, I.E., FIRST
C                 DIMENSION OF X, Y, XTOT, YTOT, XUSE, YUSE.
C        INTERP = INTERPOLATION LAW
C               = 1 - HISTOGRAM
C               = 2 - LIN X LIN Y
C               = 3 - LIN X LOG X
C               = 4 - LOG X LIN Y
C               = 5 - LOG X LOG Y
C
C        LIMITATIONS
C        ===========
C        (1) ONLY LIN X LIN Y INTERPOLATION IS RECOMMENDED. FOR
C            ANY OF THE ABOVE MENTIONED INTERPOLATION LAWS THE
C            MERGED DATA WILL BE EXACTLY EQUAL TO THE SUM OF THE
C            TWO SETS OF DATA AT THE TABULATED POINTS. HOWEVER,
C            ONLY FOR LIN X LIN Y WILL IT ALSO BE EXACTLY EQUAL
C            TO THIS SUM BETWEEN TABULATED POINTS. THE ERRORS
C            INTRODUCED IN USING ANY OTHER INTERPOLATION SCHEME
C            CAN BE QUITE LARGE UNLESS THE TABULATED POINTS ARE
C            VERY CLOSE TOGETHER - IF THEY ARE CLOSE TOGETHER
C            THE RESULTS WILL BE FAIRLY INSENSITIVE TO HOW THE
C            DATA ARE INTERPOLATED - HOWEVER, IN THIS CASE THE
C            ADVANTAGE OF NON-LINEAR INTERPOLATION IS LOST.
C        (2) X VALUES WHICH ARE THE SAME TO WITHIN ABOUT 8 DIGITS
C            ARE CONSIDERED TO BE EQUAL - THIS CONVENTION AVOIDS
C            REPEATED POINTS WHICH ARE ESSENTIALLY EQUAL.
C        (3) WHERE THERE MAY BE A DISCONTINUITY IN THE SUM, IT
C            WILL BE IGNORED IF THE Y VALUES ON EITHER SIDE OF
C            THE DISCONTINUITY ARE THE SAME TO ABOUT 8 DIGITS -
C            THIS CONVENTION AVOIDS REPEATED POINTS WHICH ARE
C            ESSENTIALLY EQUAL.
C        (4) FOR NON-LINEAR INTERPOLATION IT IS THE USERS
C            RESPONSIBILITY TO INSURE THAT ALL OF THE VALUES
C            ARE POSITIVE, X AND/OR Y, DEPENDING ON WHICH
C            REQUIRES LOG INTERPOLATION.
```

```
C      (5) IF THE MAXIMUM ALLOWABLE TABLE SIZE IS EXCEEDED
C           AN ERROR MESSAGE WILL BE PRINTED AND THE PROGRAM
C           WILL STOP.
C
C***** SAVE ******
C      SAVE
C***** SAVE ******
C***** DOUBLE *****
       IMPLICIT REAL*8 (A-H,O-Z)
C***** DOUBLE *****
       DIMENSION X(NMAX),Y(NMAX,NDIM),XTOT(NMAX),YTOT(NMAX,NDIM),
      1 XUSE(NMAX),YUSE(NMAX,NDIM)
C***** DOUBLE *****
       DATA OKDIFF/1.0D-08/
       QABS(Z)=DABS(Z)
C***** DOUBLE *****
C***** SINGLE *****
C      DATA OKDIFF/1.0E-08/
C      QABS(Z)=ABS(Z)
C***** SINGLE *****
C
C      RETURN IF NO VECTORS OR NOTHING TO MERGE.
C
       IF(NDIM.LT.0.OR.KPT.LE.0) RETURN
       IF(NTOT.GT.NMAX) GO TO 600
C
C      JUST COPY FIRST FIRST SET OF DATA.
C
       IF(NTOT.GT.0) GO TO 20
       DO 10 I=1,KPT
       XTOT(I)=X(I)
       DO 10 K=1,NDIM
   10 YTOT(I,K)=Y(I,K)
       NTOT=KPT
       RETURN
C
C      MERGE.
C
C-----INITIALIZE INDICES.
   20 IX=1
       IT=1
       I3=0
C-----SELECT LOWER ENERGY.
   30 IF(QABS(X(IX)-XTOT(IT)).LE.QABS(OKDIFF*X(IX))) GO TO 240
       IF(X(IX)-XTOT(IT)) 40,240,140
C
C      X IS LESS.
C
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
   40 IF(I3.GE.NMAX) GO TO 600
       I3=I3+1
       XUSE(I3)=X(IX)
```

```
C-----IF LESS THAN XTOT(1) JUST USE IT. OTHERWISE MERGE.
      IF(IT-1) 50,50,70
C-----BELOW FIRST POINT OF TOTAL.
   50 DO 60 K=1,NDIM
   60 YUSE(I3,K)=Y(IX,K)
      IX=IX+1
      GO TO 370
C-----ABOVE FIRST POINT OF TOTAL. INTERPOLATE TO X(IX).
   70 ITM1=IT-1
C-----IF FIRST POINT OF X ALLOW FOR DISCONTINUITY.
      IF(IX.LE.1) GO TO 90
      DO 80 K=1,NDIM
      YTERP=TERPG(X(IX),XTOT(IT),XTOT(ITM1),YTOT(IT,K),YTOT(ITM1,K),
     1 INTERP)
   80 YUSE(I3,K)=Y(IX,K)+YTERP
      IX=IX+1
      GO TO 370
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
   90 IF(I3.GE.NMAX) GO TO 600
      I3P1=I3+1
      XUSE(I3P1)=X(IX)
      DO 100 K=1,NDIM
      YTERP=TERPG(X(IX),XTOT(IT),XTOT(ITM1),YTOT(IT,K),YTOT(ITM1,K),
     1 INTERP)
      YUSE(I3,K)=YTERP
  100 YUSE(I3P1,K)=Y(IX,K)+YTERP
C-----IGNORE REPEATED POINTS (Y=0 AT FIRST POINT).
      DO 110 K=1,NDIM
      IF(QABS(YUSE(I3,K)-YUSE(I3P1,K)).GT.QABS(OKDIFF*YUSE(I3,K)))
     1 GO TO 120
  110 CONTINUE
      GO TO 130
  120 I3=I3P1
  130 IX=IX+1
      GO TO 370
C
C     XTOT IS LESS.
C
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
  140 IF(I3.GE.NMAX) GO TO 600
      I3=I3+1
      XUSE(I3)=XTOT(IT)
C-----IF LESS THAN X(1) JUST USE IT. OTHERWISE MERGE.
      IF(IX-1) 150,150,170
C-----BELOW FIRST POINT OF OUT OR BEYOND END OF TOTAL.
  150 DO 160 K=1,NDIM
  160 YUSE(I3,K)=YTOT(IT,K)
      IT=IT+1
      GO TO 370
C-----ABOVE FIRST POINT OF X. INTERPOLATE TO XTOT(IT).
  170 IXM1=IX-1
C-----IF FIRST POINT OF TOTAL ALLOW FOR DISCONTINUITY.
```

```
      IF(IT.LE.1) GO TO 190
      DO 180 K=1,NDIM
      YTERP=TERPG(XTOT(IT),X(IX),X(IXM1),Y(IX,K),Y(IXM1,K),INTERP)
  180 YUSE(I3,K)=YTOT(IT,K)+YTERP
      IT=IT+1
      GO TO 370
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
  190 IF(I3.GE.NMAX) GO TO 600
      I3P1=I3+1
      XUSE(I3P1)=XTOT(IT)
      DO 200 K=1,NDIM
      YTERP=TERPG(XTOT(IT),X(IX),X(IXM1),Y(IX,K),Y(IXM1,K),INTERP)
      YUSE(I3,K)=YTERP
  200 YUSE(I3P1,K)=YTOT(IT,K)+YTERP
C-----IGNORE REPEATED POINTS (YTOT=0 AT FIRST POINT).
      DO 210 K=1,NDIM
      IF(QABS(YUSE(I3,K)-YUSE(I3P1,K)).GT.QABS(OKDIFF*YUSE(I3,K)))
     1 GO TO 220
  210 CONTINUE
      GO TO 230
  220 I3=I3P1
  230 IT=IT+1
      GO TO 370
C
C     X = XTOT. IF FIRST POINT OF BOTH JUST ADD. IF FIRST POINT OF
C     EITHER ALLOW FOR DISCONTINUITY. OTHERWISE JUST ADD.
C
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
  240 IF(I3.GE.NMAX) GO TO 600
      I3=I3+1
      XLOW=X(IX)
      IF(XTOT(IT).LT.XLOW) XLOW=XTOT(IT)
      XUSE(I3)=XLOW
C-----IF FIRST POINT OF BOTH JUST ADD TOGETHER.
      IF(IX.EQ.1.AND.IT.EQ.1) GO TO 340
      IF(IX.GT.1) GO TO 290
C-----START OF X. ALLOW FOR DISCONTINUITY.
      DO 250 K=1,NDIM
  250 YUSE(I3,K)=YTOT(IT,K)
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
      IF(I3.GE.NMAX) GO TO 600
      I3P1=I3+1
      XUSE(I3P1)=XLOW
      DO 260 K=1,NDIM
  260 YUSE(I3P1,K)=YTOT(IT,K)+Y(IX,K)
C-----IGNORE REPEATED POINTS (Y=0)
      DO 270 K=1,NDIM
      IF(QABS(YUSE(I3,K)-YUSE(I3P1,K)).GT.QABS(OKDIFF*YUSE(I3,K)))
     1 GO TO 280
  270 CONTINUE
      GO TO 360
  280 I3=I3P1
```

```
      GO TO 360
  290 IF(IT.GT.1) GO TO 340
C-----START OF TOTAL. ALLOW FOR DISCONTINUITY.
      DO 300 K=1,NDIM
  300 YUSE(I3,K)=Y(IX,K)
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
      IF(I3.GE.NMAX) GO TO 600
      I3P1=I3+1
      XUSE(I3P1)=X(IX)
      DO 310 K=1,NDIM
  310 YUSE(I3P1,K)=YTOT(IT,K)+Y(IX,K)
C-----IGNORE REPEATED POINTS (YTOT=0)
      DO 320 K=1,NDIM
      IF(QABS(YUSE(I3,K)-YUSE(I3P1,K)).GT.QABS(OKDIFF*YUSE(I3,K)))
     1 GO TO 330
  320 CONTINUE
      GO TO 360
  330 I3=I3P1
      GO TO 360
C-----CONTINUOUS.
  340 DO 350 K=1,NDIM
  350 YUSE(I3,K)=YTOT(IT,K)+Y(IX,K)
  360 IX=IX+1
      IT=IT+1
C
C     POINT(S) DEFINE. ARE THERE ANY POINTS LEFT TO MERGE.
C
C-----ANY X POINTS.
  370 IF(IX-KPT) 380,380,390
C-----YES. ANY TOTAL POINTS.
  380 IF(IT-NTOT) 30,30,400
C-----NO. ANY TOTAL POINTS.
  390 IF(IT-NTOT) 420,420,440
C
C     ONLY X REMAINS. COPY IT.
C
  400 DO 410 I=IX,KPT
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
      IF(I3.GE.NMAX) GO TO 600
      I3=I3+1
      XUSE(I3)=X(I)
      DO 410 K=1,NDIM
  410 YUSE(I3,K)=Y(I,K)
      GO TO 440
C
C     ONLY TOTAL REMAINS. COPY IT.
C
  420 DO 430 I=IT,NTOT
C-----STOP MERGE IF TABLE SIZE WILL BE EXCEEDED.
      IF(I3.GE.NMAX) GO TO 600
      I3=I3+1
      XUSE(I3)=XTOT(I)
```

```
      DO 430 K=1,NDIM
  430 YUSE(I3,K)=YTOT(I,K)
C
C     END OF MERGE. COPY BACK TO TOTAL AND DEFINE POINT COUNT.
C
  440 DO 450 I=1,I3
      XTOT(I)=XUSE(I)
      DO 450 K=1,NDIM
  450 YTOT(I,K)=YUSE(I,K)
      NTOT=I3
      RETURN
C
C     CORE EXCEEDED.
C
  600 WRITE(6,6000) NMAX
      CALL EXIT
 6000 FORMAT(' SUBROUTINE EMERGN..OVER',I8,' MERGED POINTS.'/
     1        ' CORE EXCEEDED...EXECUTION TERMINATED')
C***** DOUBLE *****
      END
      DOUBLE PRECISION FUNCTION TERPG(X,X1,X2,Y1,Y2,INTERP)
C***** DOUBLE *****
C***** SINGLE *****
C     END
C     FUNCTION TERPG(X,X1,X2,Y1,Y2,INTERP)
C***** SINGLE *****
C
C     INTERPOLATION.
C     ============================================================
C     GIVEN TABULATED VALUES AT (X1,Y1) AND (X2,Y2) INTERPOLATE
C     TO X TO DEFINE Y ASSUME ONE OF 4 POSSIBLE INTERPOLATION
C     LAWS TO DESCRIBE THE VARIATION BETWEEN (X1,Y1) AND (X2,Y2)
C     ============================================================
C     INTERP - INTERPOLATION LAW,
C            = 1 - HISTOGRAM.
C            = 2 - LIN X VS. LIN Y.
C            = 3 - LIN X VS. LOG Y.
C            = 4 - LOG X VS. LIN Y.
C            = 5 - LOG X VS. LOG Y.
C
C     IF THE REQUESTED INTERPOLATION LAW REQUIRES LOGS AND THE
C     X AND/OR Y VALUES REQUIRED FOR THE LOGS ARE NOT POSITIVE THIS
C     ROUTINE WILL SWITCH TO LINEAR INTERPOLATION AND PRINT A WARNING
C     MESSAGE.
C
C***** SAVE ******
C     SAVE
C***** SAVE ******
C***** DOUBLE *****
      IMPLICIT REAL*8 (A-H,O-Z)
      ZLOG(Z)=DLOG(Z)
      ZEXP(Z)=DEXP(Z)
```

```
C***** DOUBLE *****
C***** SINGLE *****
C      ZLOG(Z)=ALOG(Z)
C      ZEXP(Z)=EXP(Z)
C***** SINGLE *****
C-----NO REASON TO INTERPOLATE IF Y1=Y2 (ANSWER MUST BE Y1)
       IF(Y1.NE.Y2) GO TO 10
       TERPG=Y1
       RETURN
C-----IN ALL CASES IT IS AN ERROR IF NO X INTERVAL.
    10 IF(X1.EQ.X2) GO TO 120
C-----SELECT INTERPOLATION LAW.
       IF(INTERP.LT.1.OR.INTERP.GT.5) GO TO 130
       GO TO (20,30,40,60,80),INTERP
C
C      HISTOGRAM.
C
    20 TERPG=Y1
       RETURN
C
C      LIN X VS. LIN Y.
C
    30 TERPG=((X-X1)*Y2+(X2-X)*Y1)/(X2-X1)
       RETURN
C
C      LIN X VS. LOG Y.
C
    40 IF(Y1.GT.0.0.AND.Y2.GT.0.0) GO TO 50
C-----CANNOT USE LOG Y. SWITCH TO LIN Y.
       WRITE(6,6000) Y1,Y2
       GO TO 30
    50 Q1=ZLOG(Y1)
       Q2=ZLOG(Y2)
       TERPG=ZEXP(((X-X1)*Q2+(X2-X)*Q1)/(X2-X1))
       RETURN
C
C      LOG X VS. LIN Y.
C
    60 IF(X.GT.0.0.AND.X1.GT.0.0.AND.X2.GT.0.0) GO TO 70
C-----CANNOT USE LOG X. SWITCH TO LIN X.
       WRITE(6,6010) X,X1,X2
       GO TO 30
    70 Z=ZLOG(X)
       Z1=ZLOG(X1)
       Z2=ZLOG(X2)
       TERPG=((Z-Z1)*Y2+(Z2-Z)*Y1)/(Z2-Z1)
       RETURN
C
C      LOG X VS. LOG Y.
C
    80 IF(X.GT.0.0.AND.X1.GT.0.0.AND.X2.GT.0.0) GO TO 100
       IF(Y1.GT.0.0.AND.Y2.GT.0.0) GO TO 90
```

```
C-----CANNOT USE LOG X OR Y. SWITCH TO LIN X AND Y.
      WRITE(6,6020) X,X1,X2,Y1,Y2
      GO TO 30
C-----CANNOT USE LOG X. SWITCH TO LIN X.
   90 WRITE(6,6030) X,X1,X2
      GO TO 50
  100 IF(Y1.GT.0.0.AND.Y2.GT.0.0) GO TO 110
C-----CANNOT USE LOG Y. SWITCH TO LIN Y.
      WRITE(6,6040) Y1,Y2
      GO TO 70
  110 Z=ZLOG(X)
      Z1=ZLOG(X1)
      Z2=ZLOG(X2)
      Q1=ZLOG(Y1)
      Q2=ZLOG(Y2)
      TERPG=ZEXP(((Z-Z1)*Q2+(Z2-Z)*Q1)/(Z2-Z1))
      RETURN
C-----NO X INTERVAL.
  120 WRITE(6,6050) X1,X2
      TERPG=0.5*(Y1+Y2)
      RETURN
C-----ILLEGAL INTERPOLATION LAW.
  130 WRITE(6,6060) INTERP
      TERPG=0.0
      RETURN
 6000 FORMAT(' INTERPOLATION ERROR'/
     1 ' Y1/Y2=',2(1PD12.4)/
     2 ' LIN X VS. LOG Y INTERPOLATION IMPOSSIBLE'/
     3 ' WILL USE LIN X VS. LIN Y INTERPOLATION')
 6010 FORMAT(' INTERPOLATION ERROR'/
     1 ' X/X1/X2=',3(1PD12.4)/
     2 ' LOG X VS. LIN Y INTERPOLATION IMPOSSIBLE'/
     3 ' WILL USE LIN X VS. LIN Y INTERPOLATION')
 6020 FORMAT(' INTERPOLATION ERROR'/
     1 ' X/X1/X2/Y1/Y2=',5(1PD12.4)/
     2 ' LOG X VS. LOG Y INTERPOLATION IMPOSSIBLE'/
     3 ' WILL USE LIN X VS. LIN Y INTERPOLATION')
 6030 FORMAT(' INTERPOLATION ERROR'/
     1 ' X/X1/X2=',3(1PD12.4)/
     2 ' LOG X VS. LOG Y INTERPOLATION IMPOSSIBLE'/
     3 ' WILL USE LIN X VS. LOG Y INTERPOLATION')
 6040 FORMAT(' INTERPOLATION ERROR'/
     1 ' Y1/Y2=',2(1PD12.4)/
     2 ' LOG X VS. LOG Y INTERPOLATION IMPOSSIBLE'/
     3 ' WILL USE LOG X VS. LIN Y INTERPOLATION')
 6050 FORMAT(' INTERPOLATION ERROR'/
     1 ' X1/X2=',2(1PD12.4)/
     2 ' NO X INTERVAL...WILL RETURN (Y1+Y2)/2')
 6060 FORMAT(' INTERPOLATION ERROR'/
     1 ' INTERP=',I5,' (MUST BE 1 TO 5)'/
     2 ' WILL RETURN 0.0')
C***** STOP *****
```

```
      END
      SUBROUTINE EXIT
      STOP
C***** STOP *****
      END
```