For Reference

Not to be taken from this room
An Affirmative Action/Equal Opportunity Employer

Edited by Helen M. Sinoradzki

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. References herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.
EOSMOD: A Subroutine Package for
Calculating Equations of State and Opacities

James M. Hyman
Morris M. Klein
CONTENTS

ABSTRACT .................................................................................................................. 1

I. INTRODUCTION ..................................................................................................... 1

II. EQUATION-OF-STATE AND OPACITY ROUTINES ............................................... 2
    A. Subroutine EOSDRE (input R and E, output P and T) ........................................ 3
    B. Subroutine EOSIPT (input P and T, output R and E) ........................................ 5
    C. Subroutine EOSIRT (input R and T, output P and E) ......................................... 7
    D. Subroutine EOSDRT (input R and T, output P and E) ........................................ 9
    E. Subroutine EOSORT (input R and T, output O) .................................................. 11

III. SCALING THE TABLES .......................................................................................... 13
    A. Density Scaling .................................................................................................. 13
    B. Adding New Units ............................................................................................. 13

IV. USER NOTES ......................................................................................................... 14
    A. Information File .................................................................................................. 14
    B. Graphic Output .................................................................................................. 14
    C. Increasing LCM Allocation ................................................................................ 14
    D. Reducing the EOS Data Range ......................................................................... 14
    E. Error Flags and Messages ................................................................................. 14
    F. Creating an EOS/Opacity Table for Mixtures .................................................... 14
    G. Listing the Available Materials ....................................................................... 15

V. EXAMPLES ............................................................................................................. 15
    A. Simplest Example .............................................................................................. 15
    B. Advanced EOS Example ................................................................................... 16

VI. MATERIALS AVAILABLE ...................................................................................... 18

VII. LOCATION OF COMPUTER FILES AT LOS ALAMOS ........................................ 19

VIII. RELATIONSHIPS BETWEEN UNITS .................................................................. 20

ACKNOWLEDGMENTS .................................................................................................. 21

REFERENCES ............................................................................................................... 22
APPENDIX A: SUBROUTINE LISTING

User Callable Subroutines

- EOSDRE (input R and E, output P and T) ........................................ 24
- EOSDRT (input R and T, output P and E) ........................................ 28
- EOSIPT (input P and T, output R and E) ........................................ 32
- EOSIRT (input R and T, output P and E) ........................................ 35
- EOSORT (input R and T, output O) .............................................. 38

Internal Subroutines

- EOSBEG (initializes all the common block variables) ...................... 43
- EOSCON (defines the table conversion factors) ............................ 46
- EOSDSL (allows the user to scale the density) .............................. 49
- EOSEFD (finds the EOS table) ...................................................... 51
- EOSFAS (assigns the input files) .............................................. 54
- EOSGET (loads the Sesame EOS data files) .................................... 56
- EOSKUT (checks the validity of KPARM) ........................................ 59
- EOSOFD (finds the opacity table) ............................................. 61

T-4 Subroutines used by EOSMOD

- DPACXX (packs real numbers) ...................................................... 64
- GETEOSX (loads the total EOS tables) .......................................... 65
- GETINVX (gets inverted EOS tables) ........................................... 67
- GETRPOX (loads Rosseland/Planck opacity tables) ......................... 69
- INBUFRX (sequential read) ............................................................. 71
- INV301X (inverts a 301 table) .................................................... 72
- ISRCHKX (index search) .............................................................. 74
- MATCHKX (checks if material table is loaded) ............................... 75
- RATFNIX (one-dimensional interpolator) ....................................... 76
- TABRANX (fetches a given table for a given material from a Sesame II library) .......................................................... 78
- T4DATIX (interpolates the inverted tables) ................................... 80
- T4DATX (search/interpolate for atomic data as functions of region, density, and temperature) ........................................... 83
- T4INTPX (interpolates for a function z(x,y) and its derivatives) ........ 86
- T4PTREX (computes R and E from inverted table) ............................. 89
- T4RTPEX (computes P and E from inverted table) ............................. 91

APPENDIX B: CROSS-REFERENCE DIRECTORY OF EOSLIB ......................... 93

APPENDIX C: TEST PROGRAM

- OUTPUT ................................................................. 100
- 110
EOSMOD: A SUBROUTINE PACKAGE FOR CALCULATING EQUATIONS OF STATE AND OPACITIES

by

James M. Hyman and Morris M. Klein

ABSTRACT

The EOSMOD package includes a set of FORTRAN subroutines written to make the SESAME equation-of-state and opacity tables readily available. We have tried to make these routines as accessible as possible for casual users with routine problems and at the same time, allow sufficient flexibility for sophisticated users with complicated situations. This has caused only a slight loss in efficiency (~5%) compared to using the SESAME routines directly.

I. INTRODUCTION

The SESAME library is a collection of data files containing equations of state (EOS), Rosseland mean opacities, and other material properties over a wide range of temperatures, pressures, and densities. Group T-4 at Los Alamos National Laboratory developed a FORTRAN subroutine library to access and analyze these data files. The EOSMOD subroutine package was developed to complement the T-4 routines and simplify the interface between these codes and the user's program. The EOSMOD routines do the bookkeeping, unit's conversion, and large core memory (LCM) initialization; reduce the number of parameters the casual user needs to be aware of; and are written in a structured modular design to allow a sophisticated user to modify and optimize the package easily for a particular problem.

To use the package, just access the SESAME data files as described in Sec. VIII and call the driver subroutine. The package then locates the requested data file, converts it to the units specified by the user, and stores it in LCM. On all subsequent calls, the package remembers the contents and location of the file and interpolates the data at the user's requested values.

If the EOS or opacity of a mixture is needed, then the directory of mixtures currently available (MIXDIR) should be checked. If the specific mixture is not in the library, then a SESAME data file may be generated using the procedure described in Refs. 4, 5, and 6.
II. EQUATION-OF-STATE AND OPACITY ROUTINES

The first time a routine is called for each new material (LMAT), the table is converted to the kind of units (KUNIT) requested by the user and copies into LCM. The location or material table number (IMATE) is returned to indicate the location of the EOS table or to indicate whether an error was encountered in the initialization; for example, if the material was not found in the library.

The EOS tables can be loaded in two different formats. The standard SESAME format is for calculating the pressure $P$ and internal energy $E$ of a material as a function of the density $\rho$ and temperature $T$. Subroutine EOSDRT loads and reads the tables in this format.

The inverted SESAME format is for calculating $P$ and $T$ as a function of $\rho$ and $E$ with subroutine EOSDRE. These inverted tables also can be accessed in other ways using the iterative subroutines EOSIRT and EOSIPT. Subroutine EOSIPT calculates $E$ and $\rho$ as function of $P$ and $T$, and subroutine EOSIRT calculates $P$ and $E$ as a function of $\rho$ and $T$. These iterative subroutines use an iterative scheme that is slow compared to the direct methods used in subroutines EOSDRT and EOSDRE. Their main use is for initial state computations and occasional diagnostics.

To speed up the execution time when using subroutine EOSDRE or EOSDRT, the user can request that the package return only $T$, $P$, or $E$ using the computational mode flag KBR. Computer time can also be saved by using the KFN flag to indicate a bilinear interpolation of the data rather than the more accurate (but slower) rational interpolation. These interpolation procedures are described in more detail in Refs. 1 and 7.
A. Subroutine EOSDRE (input R and E, output P and T)

The calling sequence when \( p \) and \( E \) are independent variables (input) and \( P(p,E) \) and \( T(p,E) \) are dependent variables (output) is

\[
\text{CALL EOSDRE (LMAT,R,E,P,T,KEOS,IMATE).}
\]

The arguments are defined as follows.

**INPUT VARIABLES**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMAT</td>
<td>Material name in an A10 field; for example, LMAT = &quot;HELIUM.&quot; The materials available are listed in Sec. VII and the file MIXDIR. (The material SESAME number can also be used to specify the material by setting LMAT to the SESAME number; for example, LMAT = &quot;5760&quot; for helium.)</td>
</tr>
<tr>
<td>R</td>
<td>Density ( p ).</td>
</tr>
<tr>
<td>E</td>
<td>Internal energy.</td>
</tr>
<tr>
<td>KEOS</td>
<td>Multiple parameter flag to describe how to write and retrieve the data file. KEOS has four decimal digits: KBR, KUNIT, KREPE, and KFN ((KEOS = 1000\times KBR + 100 \times KUNIT + 10 \times KREPE + KFN)).</td>
</tr>
</tbody>
</table>

**KBR** Computational flag to indicate which quantities and their partial derivatives are to be calculated and returned by the package.

- 0 Compute \( P \) and \( T \) and their partial derivatives \( \frac{\partial P}{\partial p}, \frac{\partial P}{\partial E}, \frac{\partial T}{\partial p}, \) and \( \frac{\partial T}{\partial E} \).
- 1 Compute \( P \) and its partial derivatives \( \frac{\partial P}{\partial p} \) and \( \frac{\partial P}{\partial E} \).
- 2 Compute \( T \) and its partial derivatives \( \frac{\partial T}{\partial p} \) and \( \frac{\partial T}{\partial E} \).

**KUNIT** Kind of units for writing the data file. The units are explained fully in Sec. IX.

- 0 SESAME EOS units.
- 1 CGS units.
- 2 Standard International Units (SIU).
- 3 Hydrox EOS units.
- 4 Hydrox opacity units.
- 5 SESAME opacity units.
- 6 LASNEX units.

**KREPE** Computational flag to indicate whether \( E \) is to be represented as energy per unit mass or energy per unit volume.

- 0 Energy in units of energy per unit mass (for example, ergs/g). This is the usual \( E \).
Energy in units of energy per unit volume (for example, ergs/cm³). This is the energy density $\rho E$ commonly computed in hydrodynamic computer codes.

KFN Indicates the form of the function used to interpolate the data tables.

0 Accurate rational function interpolation.
1 Fast bilinear function interpolation.

IMATE Indicates whether to load the data file if it does not exist for LMAT or go directly to a previously loaded file.

0 Check if the data file for LMAT has been loaded. If not, search for the file, convert it to the proper units, and copy it into LCM using the inverted SESAME format.
N>0 Equal to the LMAT table number. This number was returned by the package on a previous call to subroutine EOSDRE, EOSIPT, or EOSIRT and is distinct for each LMAT. This option is faster than IMATE = 0 because it skips searching the directory of previously loaded data files.

OUTPUT VARIABLES

P Array of dimension 3 containing the pressure and its partial derivatives.

P(1) Pressure.
P(2) Density derivative of the pressure ($\partial P/\partial p$).
P(3) Energy derivative of the pressure ($\partial P/\partial E$).

T Array of dimension 3 containing the temperature and its partial derivatives. This array must be dimensional even if the partial derivatives are not computed.

T(1) Temperature.
T(2) Density derivative of the temperature ($\partial T/\partial p$).
T(3) Energy derivative of the temperature ($\partial T/\partial E$).

IMATE Indicates the success or failure of locating and loading the data file for LMAT.

0 LMAT not found.
N>0 LMAT table number (success).
N<0 Insufficient LCM storage. The LCM memory allocation must be increased by at least $|N|$ storage locations by the procedure described in Sec. V.
B. Subroutine EOSIPT (input P and T, output R and E)

The calling sequence when P and T are independent variables (input) and ρ(P,T) and E(P,T) are dependent variables (output) is

CALL EOSIPT (LMAT,P,T,R,E,KEOS,IMATE).

This routine reads the EOS data in the inverted SESAME format and uses an iterative method to interpolate the data. The arguments are defined as follows.

INPUT VARIABLES

LMAT Material name in an ALO field; for example, LMAT = "HELIUM."

The materials available are listed in Sec. VII and the file MIXDIR. (The material SESAME number can also be used to specify the material by setting LMAT to the SESAME number; for example, LMAT = "5760" for helium.)

P Pressure.

T Temperature.

KEOS Multiple parameter flag to describe how to write and retrieve the data file. KEOS has three decimal digits: KUNIT, KREPE, and KFN (KEOS = 100*KUNIT + 10*KREPE + KFN).

KUNIT Kind of units for writing the data file. The units are explained fully in Sec. IX.

0 SESAME EOS units.
1 CGS units.
2 Standard International Units (SIU).
3 Hydrox EOS units.
4 Hydrox opacity units.
5 SESAME opacity units.
6 LASNEX units.

KREPE Computational flag to indicate whether E is to be represented as energy per unit mass or energy per unit volume.

0 Energy in units of energy per unit mass (for example, ergs/g). This is the usual E.
1 Energy in units of energy per unit volume (for example, ergs/cm³). This is the energy density ρE commonly computed in hydrodynamic computer codes.

KFN Indicates the form of the function used to interpolate the data tables.

0 Accurate rational function interpolation.
1 Fast bilinear function interpolation.
IMATE  Indicates whether to load the data file if it does not exist for LMAT or go directly to a previously loaded file.

  0       Check if the data file for LMAT has been loaded. If not, search for the file, convert it to the proper units, and copy it into LCM using the inverted SESAME format.

  N>0     Equal to the LMAT table number. This number was returned by the package on a previous call to subroutine EOSDRE, EOSIPT, or EOSIRT and is distinct for each LMAT. This option is faster than IMATE = 0 because it skips searching the directory of previously loaded data files.

OUTPUT VARIABLES

R   Density.
E   Internal energy.
IMATE Indicates the success or failure of locating and loading the data file for LMAT.

  0       LMAT not found or the iteration failed to converge.

  N>0     LMAT table number (success).

  N<0     Insufficient LCM storage. The LCM memory allocation must be increased by at least |N| storage locations by the procedure described in Sec. V.
C. Subroutine EOSIRT (input R and T, output P and E)

The calling sequence when \( \rho \) and \( T \) are independent variables (input) and \( P(\rho,T) \) and \( E(\rho,T) \) are dependent variables (output) is

\[
\text{CALL EOSIRT (LMAT,R,T,P,E,KEOS,IMATE).}
\]

This routine reads the EOS data in the inverted SESAME format and uses an iterative method to interpolate the data. The arguments are defined as follows.

**INPUT VARIABLES**

- **LMAT**: Material name in an A10 field; for example, LMAT = "HELIUM."
  
  The materials available are listed in Sec. VII and the file MIXDIR. (The material SESAME number can also be used to specify the material by setting LMAT to the SESAME number; for example, LMAT = "5760" for helium.)

- **R**: Density.

- **T**: Temperature.

- **KEOS**: Multiple parameter flag to describe how to write and retain the data file. KEOS has three decimal digits: KUNIT, KREPE, and KFN (KEOS = 100*KUNIT + 10*KREPE + KFN).

- **KUNIT**: Kind of units for writing the data file. The units are explained fully in Sec. IX.
  
  0: SESAME EOS units.
  1: CGS units.
  2: Standard International Units (SIU).
  3: Hydrox EOS units.
  4: Hydrox opacity units.
  5: SESAME opacity units.
  6: LASNEX units.

- **KREPE**: Computational flag to indicate whether \( E \) is to be represented as energy per unit mass or energy per unit volume.
  
  0: Energy in units of energy per unit mass (for example, ergs/g). This is the usual \( E \).
  1: Energy in units of energy per unit volume (for example, ergs/cm\(^3\)). This is the energy density \( \rho E \) commonly computed in hydrodynamic computer codes.

- **KFN**: Indicates the form of the function used to interpolate the data tables.
  
  0: Accurate rational function interpolation.
  1: Fast bilinear function interpolation.
IMATE Indicates whether to load the data file if it does not exist for LMAT or go directly to a previously loaded file.

0 Check if the data file for LMAT has been loaded. If not, search for the file, convert it to the proper units, and copy it into LCM using the inverted SESAME format.

N>0 Equal to the LMAT table number. This number was returned by the package on a previous call to subroutine EOSDRE, EOSIPT, EOSDRT, or EOSIRT and is distinct for each LMAT. This option is faster than IMATE = 0 because it skips searching the directory of previously loaded data files.

OUTPUT VARIABLES

P Pressure.
E Internal energy.
IMATE Indicates the success or failure of locating and loading the data file for LMAT.

0 Material not found or iteration failed to converge.
N>0 LMAT table number (success).
N<0 Insufficient LCM storage. The LCM memory allocation must be increased by at least |N| storage locations by the procedure described in Sec. V.
D. Subroutine EOSDRT (input R and T, output P and E)

The calling sequence when \( p \) and \( T \) are independent variables (input) and \( P(p,T) \) and \( E(p,T) \) are dependent variables (output) is

\[
\text{CALL EOSDRT (LMAT},R,T,P,E,KEOS,IMATE) .
\]

The arguments are defined as follows.

**INPUT VARIABLES**

- **LMAT**: Material name in an A1O field; for example, LMAT = "HELIUM." The materials available are listed in Sec. VII and in the file MIXDIR. (The material SESAME number can also be used to specify the material by setting LMAT to the SESAME number; for example, LMAT = "5760" for helium.)

- **R**: Density (\( \rho \)).

- **T**: Temperature.

- **KEOS**: Multiple parameter flag to describe how to write and retrieve the data file. KEOS has four decimal digits: \( KBR, KUNIT, KREPE, \) and \( KFN \) (\( KEOS = 1000 \times KBR + 100 \times KUNIT + 10 \times KREPE + KFN \)).

- **KBR**: Computational flag to indicate which quantities and their partial derivatives are to be calculated and returned by the package.
  - 0: Compute \( P \) and \( E \) and their partial derivatives \( \partial P/\partial \rho \), \( \partial P/\partial T \), \( \partial E/\partial \rho \), and \( \partial E/\partial T \).
  - 1: Compute \( P \) and its partial derivatives \( \partial P/\partial \rho \) and \( \partial P/\partial T \).
  - 2: Compute \( T \) and its partial derivatives \( \partial E/\partial \rho \) and \( \partial E/\partial T \).

- **KUNIT**: Kind of units for writing the data file. The units are explained fully in Sec. IX.
  - 0: SESAME EOS units.
  - 1: CGS units.
  - 2: Standard International Units (SIU).
  - 3: Hydrox EOS units.
  - 4: Hydrox opacity units.
  - 5: SESAME opacity units.
  - 6: LASNEX units.

- **KREPE**: Computational flag to indicate whether \( E \) is to be represented as energy per unit mass or energy per unit volume.
  - 0: Energy in units of energy per unit mass (for example, ergs/g). This is the usual \( E \).
imates

Energy in units of energy per unit volume (for example, ergs/cm$^3$). This is the energy density $\rho E$ commonly computed in hydrodynamic computer codes.

KFN

Indicates the form of the function used to interpolate the data tables.

0 Accurate rational function interpolation.
1 Fast bilinear function interpolation.

IMATE

Indicates whether to load the data file if it does not exist for LMAT or go directly to a previously loaded file.

0 Check if the data file for LMAT has been loaded. If not, search for the file, convert it to the proper units, and copy it into LCM using the standard SESAME format.
N>0 Equal to the LMAT table number. This number was returned by the package on a previous call to subroutine EOSDRE, EOSIPT, EOSDRT, or EOSIRT and is distinct for each LMAT. This option is faster than IMATE = 0 because it skips searching the directory of previously loaded data files.

OUTPUT VARIABLES

P Array of dimension 3 containing the pressure and its partial derivatives.

P(1) Pressure.
P(2) Density derivative of the pressure ($\partial P/\partial \rho$).
P(3) Temperature derivative of the pressure ($\partial P/\partial T$).

E Array of dimension 3 containing the internal energy and its partial derivatives. This array must be dimensional even if the partial derivatives are not computed.

E(1) Internal energy.
E(2) Density derivative of the internal energy ($\partial E/\partial \rho$).
E(3) Energy derivative of the internal energy ($\partial E/\partial T$).

IMATE Indicates the success or failure of locating and loading the data file for LMAT.

0 LMAT not found.
N>0 LMAT table number (success).
N<0 Insufficient LCM storage. The LCM memory allocation must be increased by at least $|N|$ storage locations by the procedure described in Sec. V.
E. Subroutine EOSORT (input R and T, output O)

The Rosseland mean opacity\(^8\) of the material (LMAT) can be calculated by calling subroutine OPCRT. The density R and temperature T are the independent input variables, and the opacity O is the dependent output variable. When the internal energy, but not the temperature, is known, the user must first call subroutine EOSDRE to get the temperature. The (KUNIT) that the table is to be written in and the type of interpolation function (KFN) to be used must be provided by the user. The material opacity table number (IMATO) is returned by the package to indicate the location of the opacity table in LCM or if an error was encountered by the routine.

The calling sequence when \(\rho\) and T are independent variables (input) and O is the dependent variable (output) is

\[
\text{CALL EOSORT (LMAT,R,T,O,KOPC,IMATO)}
\]

The arguments are defined as follows.

INPUT VARIABLES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMAT</td>
<td>Material name in an A1O field; for example, LMAT = &quot;HELIUM.&quot; The materials available are listed in Sec. VII and in the file MIXDIR. (The material SESAME number can also be used to specify the material by setting LMAT to the SESAME number; LMAT = &quot;15760&quot;).</td>
</tr>
<tr>
<td>R</td>
<td>Density ((\rho)).</td>
</tr>
<tr>
<td>T</td>
<td>Temperature.</td>
</tr>
<tr>
<td>KOPC</td>
<td>Multiple parameter flag to describe how to write and retrieve the data file. KEOS has three decimal digits: KUNIT, KREPO, and KFN (KEOS = 100<em>KUNIT + 10</em>KREPO + KFN).</td>
</tr>
<tr>
<td>KUNIT</td>
<td>Computational flag to indicate the kind of units for writing the data file. The units are explained fully in Sec. IX.</td>
</tr>
<tr>
<td>KREPO</td>
<td>Computational flag to indicate which representation to use for the opacity.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KUNIT</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>SESAME EOS units.</td>
</tr>
<tr>
<td>1</td>
<td>CGS units.</td>
</tr>
<tr>
<td>2</td>
<td>Standard International Units (SIU).</td>
</tr>
<tr>
<td>3</td>
<td>Hydrox EOS units.</td>
</tr>
<tr>
<td>4</td>
<td>Hydrox opacity units.</td>
</tr>
<tr>
<td>5</td>
<td>SESAME opacity units.</td>
</tr>
<tr>
<td>6</td>
<td>LASNEX units.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KREPO</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Opacity represented as (K) in dimensional units of (\text{length}^2/\text{mass}).</td>
</tr>
<tr>
<td>1</td>
<td>Opacity represented as a mean-free path, (\Lambda = 1/(\kappa\rho)), in dimensional units of length.</td>
</tr>
</tbody>
</table>
Indicates the form of the function used to interpolate the data tables.

<table>
<thead>
<tr>
<th>KFN</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Accurate rational function interpolation.</td>
</tr>
<tr>
<td>1</td>
<td>Fast bilinear function interpolation.</td>
</tr>
</tbody>
</table>

**IMATO**

Indicates whether to load the data file if it does not exist for LMAT or go directly to a previously loaded file.

<table>
<thead>
<tr>
<th>IMATO</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Check if the data file for LMAT has been loaded. If not, search for the file, convert it to the proper units, and copy it into LCM.</td>
</tr>
<tr>
<td>N&gt;0</td>
<td>Equal to the LMAT table number. This number was returned by the package on a previous call to subroutine OPCRT and is distinct for each LMAT. This option is faster than IMATE = 0 because it skips searching the directory of previously loaded data files.</td>
</tr>
<tr>
<td>N&lt;0</td>
<td>Insufficient LCM storage. The LCM memory allocation must be increased by at least (</td>
</tr>
</tbody>
</table>

**OUTPUT VARIABLES**

<table>
<thead>
<tr>
<th>IMATEO</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Opacity. LMAT not found.</td>
</tr>
<tr>
<td>N&gt;0</td>
<td>LMAT table number (success).</td>
</tr>
<tr>
<td>N&lt;0</td>
<td>Insufficient LCM storage. The LCM memory allocation must be increased by at least (</td>
</tr>
</tbody>
</table>
III. SCALING THE TABLES

A. Density Scaling

When the atomic mass of the desired material is different from the mass of the SESAME material and the material properties are expected to be similar, then a simple density scale factor can be incorporated into the EOS and opacity tables. This is usually sufficient for isotope mixtures of a specific material.

The density scale factor (DSFAC) for the material LMAT can be incorporated into the tables through the user supplied subroutine EOSSCL(LMAT, DSFAC). This routine will be called by EOSMOD, and the routine should return DSFAC equal to the ratio of the atomic mass of the SESAME material and the desired material; that is,

$$\text{DSFAC} = \frac{\text{atomic mass of SESAME material LMAT}}{\text{atomic mass of desired material}}$$

In example B, Sec. VI, the EOS of a 60% deuterium/40% tritium mixture is approximated from the EOS of deuterium by defining

$$\text{DSFAC} = \frac{2}{0.60 \times 2 + 0.40 \times 3} = 0.833$$

B. Adding New Units

When the EOS and opacity tables are copied from the SESAME data file into LCM, they are converted to the user's specified units by multiplying the data by a conversion factor. The EOS conversion factors for temperature, density, pressure, and internal energy are TFACE, RFACE, PFACE, and EFACE, respectively. The opacity conversion factors for density, temperature, and opacity are RFACO, TFACO, and OFACO, respectively.

These factors are defined in the EOSMOD subroutine EOSCON according to the user's specifications (KUNIT) and communicated to the various EOSMOD routines through the common blocks

```
COMMON /EOSCCE/ TFACE, RFACE, PFACE, EFACE
COMMON /EOSCCO/ TFACO, RFACO, OFACO
```

If the desired units are not automatically available in the package, the user can reset the conversion factors at execution time (not in a data statement) in the user-provided routine EOSSCL (See Sec. V.A). This is done in example 5B.

The scale factor is the constant that the data in the original SESAME EOS table units (KUNIT = 0) or SESAME opacity table units (KUNIT = 5) must be multiplied by to convert the data to the desired units. The conversion factors for KUNIT = 0-6 are easily found in subroutine EOSCON listed in the Appendix A.
IV. USER NOTES

A. Information File

The file EOSINFO contains the latest user notes and information on the EOSMOD package. Users are encouraged to add notes to this file that will be helpful to others using the package.

B. Graphic Output

Several plotting routines exist for displaying EOS and opacity data stored in the SESAME format. The routines are maintained by Los Alamos Group T-4 and are described in the data file S2DHELP. See Ref. 9.

C. Increasing LCM Allocation

On the CDC 7600's at the Laboratory, the EOSMOD default LCM allocation is 12,000 words. This is enough storage to load approximately four EOS tables. The allocation can be increased to LCMX, say 20,000 words, by declaring

```
LEVEL 2, TBLS
COMMON /S2DIR/ LCMX
COMMON /SESDAT/ TBLS (20000)
LCMX=20000
```

in the main program. LCMX must be set to the dimension of TBLS at execution time not in a data statement. Example 2B in Sec. VI does this.

D. Reducing the EOS Data Range

Subroutine WINDOW in the Hydses package can be used to reduce the size of a standard temperature-based SESAME data table when the full density and temperature range are not needed. Because the use of this routine requires knowledge of where and how the tables are stored in LCM, we refer the interested user to the HYDSES report for further information.

E. Error Flags and Messages

All error messages are written into a file called "OUTPUT." The name of this file is defined at compile time in the Hollerith variable LOUT in the common block

```
COMMON/EOSCZ/LOUT.
```

LOUT can be changed to another file name or unit number of the user at execution time before the first call to EOSMOD.

F. Creating an EOS/Opacity Table for Mixtures

The EOS/opacity tables suitable for gas mixtures can be created with the aid of the BCON controller MIXB. These EOS mixture tables currently are prepared under the assumption that the ideal mixing of individual com-
ponent parts occurs. The pressure of the mixture is taken, as in Dalton's law, to be the weighted mole fraction of the partial pressures of the component parts. Opacity mixture tables are created by weighing the opacities of each component in frequency space according to the component's fraction of the total mass and then by integrating the resultant frequency spectrum to obtain the Rossland mean opacity for the mixture. These tables are generated in SESAME format with a simple input deck.

G. Listing the Available Materials
The EOS and opacity materials, and SESAME numbers are listed in the common blocks

```
COMMON/EOSC5/NMAT,LABMAT(40),IDMAT(40)
COMMON/EOSC7/NMATO,LABMO(40),IDMATO(40).
```

There are NMAT (NMATO) EOS (opacity) materials in the common blocks. The labels in LABMAT (LABMO) are in an A10 format and correspond to the SESAME material ID in IDMAT (IDMATO).

To list the EOS tables, execute the code

```
DO 10 I = 1,NMAT
10 PRINT 20, LABMAT(I),IDMAT(I)
20 FORMAT (1X,A10,I10).
```

A complete description of each material can be found using the T-4 SESAME utility LSTX. (See Sec. VIII and Ref. 11.)

V. EXAMPLES

A. Simplest Example
The following program will compute the pressure of helium in microbars at a temperature of 300 K and a density of 0.01 g/cm³.

```
PROGRAM TST(OUTPUT)
DIMENSION P(3),E(3)
LMAT = "HELIUM"
R = 0.01
T = 300.0
KEOS = 110
IMATE = 0
CALL EOSDRT(LMAT,R,T,P,E,KEOS,IMATE)
PRINT 10, P(1)
10 FORMAT(''PRESSURE = '', 1PE12.4,'' MICROBARS'')
CALL EXIT
END
```
The execute line is

FTN (I=TEST, GLIB=EOSLIB, GO)

The output from this program is:

PRESSURE = 6.3498E+07 MICROBARS

B. Advanced EOS Example

In this example we scale the density for a 60% deuterium/40% tritium mixture, add a new set of MKS units, and increase LCM so we can load four EOS tables.

```plaintext
PROGRAM TST(OUTPUT)

DECLARE THE COMMON BLOCKS USED BY EOSMOD(HYDSES) TO STORE THE TABLES
LEVEL 2,TBLS
COMMON /S2DIR/ LCMX
COMMON /SESDAT/ TBLS(20000)

DECLARE THE COMMON BLOCK WITH THE FILE LABELS
COMMON /EOSC2/ LF41,LF42,LF43,LF44,LF45

DIMENSION P(3),E(3)

INCREASE THE LCM STORAGE AVAILABLE TO EOSMOD
THE MAXIMUM LCM STORAGE IN /SESDAT/ IS LCMX WORDS
LCMX = 20000

SET THE DENSITY IN KILOGRAMS
AND THE TEMPERATURE IN DEGREES KELVIN

R = 1.E-5
T = 300.0
KEOS = 110

THE TABLE FOR DEUTERIUM IS CONVERTED BY EOSMOD TO A MIXTURE BY
SCALING THE DENSITY IN SUBROUTINE E oss.L.
LMAT = "DEUTERIUM"
IMATE = 0
CALL EOSDRT(LMAT,R,T,P,E,KEOS,IMATE)
PRINT 10,P(1)
10 FORMAT("40% D + 60% T, PRESSURE =", 1PE12.4, "BARS")

LMAT = "HELIUM"
IMATE = 0
CALL EOSDRT(LMAT,R,T,P,E,KEOS,IMATE)
PRINT 20,LMAT,P(1)
```

16
DECLARE THE NAME OF THE PRIVATE EOS DATA TABLES AND DIRECTORY
LF44 = "MIXLIB"
LF45 = "MIXDIR"

FIND EOS PRESSURE OF A 90% DEUTERIUM 10% NEON MIXTURE.
The EOS is contained in the private tables MIXLIB and MIXDIR
LMAT = "D9ONE10"
IMATE = 0
CALL EOSDRT(LMAT,R,T,P,E,KEOS,IMATE)
PRINT 20,LMAT,P(1)

20 FORMAT(1X,A1O," PRESSURE = ",1PE12.4," "BARS")

CALL EXIT
END

SUBROUTINE EOSSCL (LMAT,DSFAC)
COMMON /EOSCCE/ TFACE,RFACE,PFACE,EFACE,KPE

SET UP A NEW SET OF UNITS TO BE USED BY THE CODE
PFACE AND EFACE ARE THE CONVERSION FACTORS FOR THE MKS SYSTEM
YOU ONLY NEED TO DEFINE THE CONVERSION FACTORS THAT DIFFER
FROM THE KEOS DECLARED UNITS (CGS SYSTEM)
CONVERT THE PRESSURE TO BARS
PFACE = 1.E+6
EFACE = 1.E+6

WE NOW RESCALE THE DENSITY FOR DEUTERIUM TO WHAT IT
SHOULD BE FOR A 60% DEUTERIUM AND 40% TRIDUUM MIXTURE

IF(LMAT.EQ."DEUTERIUM") DSFAC=0.833

RETURN
END

The output from this program is

40% D + 60% T PRESSURE = 6.1923E+04 BARS
HELUM PRESSURE = 5.6817E+04 BARS
D9ONE10 PRESSURE = 7.3926E+05 BARS
## VI. MATERIALS AVAILABLE

The following materials are currently available using EOSMOD.\(^\text{11}\)

<table>
<thead>
<tr>
<th>Material Name (LMAT)</th>
<th>EOS File</th>
<th>Opacity Name</th>
<th>Temperature Maximum (K)</th>
<th>Density Minimum (g/cm(^3))</th>
<th>Density Minimum (g/cm(^3))</th>
<th>SESAME Material Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALLUVIUM (Nevada)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUMINUM</td>
<td>SESAME SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL2O3 (= Al(_2)O(_3))</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARGON</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BERYLLIUM</td>
<td>SESAME SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BORON</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BORON CARB(ide)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BRASS</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CALCIUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CARBON</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH (= polystyrene)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH2 (= polyethylene)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHLORINE</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHROMIUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COPPER</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEUTERIUM</td>
<td>SESAME SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GOLD</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRANITE</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HELIUM</td>
<td>SESAME SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HE (= High Explosive)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IRON</td>
<td>SESAME SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IRON2</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEAD</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LITHIUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6LiD (= 6LiD)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6LiH (= 6LiH)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAGNESIUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MOLYBDENUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NEON</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NICKEL</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NITROGEN</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OXYGEN</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PBX-9502</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PHOSPHORUS</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLATINUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>POLYE (= polyethylene)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Material Name</td>
<td>EOS File Name</td>
<td>Opacity</td>
<td>Temperature Maximum (K)</td>
<td>Density Minimum (g/cm³)</td>
<td>Density Minimum (g/cm³)</td>
<td>SESAME Material Number</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------</td>
<td>---------</td>
<td>-------------------------</td>
<td>-------------------------</td>
<td>-------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>POLYS (= polystyrene)</td>
<td>SESAME</td>
<td></td>
<td>4x10⁸</td>
<td>0.0082</td>
<td>2.1x10⁴</td>
<td>7590</td>
</tr>
<tr>
<td>POTASSIUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2460</td>
</tr>
<tr>
<td>SIO2 (= SiO₂)</td>
<td>SESAME SESAME</td>
<td></td>
<td>4x10⁸</td>
<td>0.017</td>
<td>4.4x10⁴</td>
<td>7380</td>
</tr>
<tr>
<td>SILICON</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3810</td>
</tr>
<tr>
<td>SODIUM</td>
<td>SESAME SESAME</td>
<td></td>
<td>1x10⁴</td>
<td>0.0</td>
<td>1.3</td>
<td>2448</td>
</tr>
<tr>
<td>SS (Steel)</td>
<td>SESAME SESAME</td>
<td></td>
<td>4x10⁸</td>
<td>0.062</td>
<td>1.6x10⁵</td>
<td>4270</td>
</tr>
<tr>
<td>STAINLESS STEEL</td>
<td>SESAME</td>
<td></td>
<td>4x10⁸</td>
<td>0.062</td>
<td>1.6x10⁵</td>
<td>4270</td>
</tr>
<tr>
<td>STEAM</td>
<td>SESAME</td>
<td></td>
<td>1300</td>
<td>0.0</td>
<td>0.9</td>
<td>7151</td>
</tr>
<tr>
<td>SULPHUR</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4010</td>
</tr>
<tr>
<td>TITANIUM</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2960</td>
</tr>
<tr>
<td>TITANIUM N (Nitride)</td>
<td>SESAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6000</td>
</tr>
<tr>
<td>UO2 (= UO₂)</td>
<td>SESAME</td>
<td></td>
<td>3x10⁴</td>
<td>0.0</td>
<td>14.3</td>
<td>7432</td>
</tr>
<tr>
<td>URANIUM</td>
<td>SESAME</td>
<td></td>
<td>4x10⁸</td>
<td>0.15</td>
<td>4.0x10⁵</td>
<td>1540</td>
</tr>
<tr>
<td>URETHANE</td>
<td>SESAME</td>
<td></td>
<td>4x10⁹</td>
<td>0.0099</td>
<td>2.5x10⁴</td>
<td>7560</td>
</tr>
<tr>
<td>VERMICULITE</td>
<td>SESAME</td>
<td></td>
<td>4x10⁸</td>
<td>0.021</td>
<td>5.4x10⁴</td>
<td>7520</td>
</tr>
<tr>
<td>WATER</td>
<td>SESAME SESAME</td>
<td></td>
<td>1.8x10⁸</td>
<td>2.0x10⁻⁶</td>
<td>4.0x10²</td>
<td>7150</td>
</tr>
</tbody>
</table>

On the CRAY-1 use only the first eight characters in the above material names.

VII. LOCATION OF COMPUTER FILES AT LOS ALAMOS

The files needed to execute any of the EOSMOD subroutines are available on the CDC 7600 computers at Los Alamos. For most users, it will be sufficient to attach the EOSMOD library,

MASS GET/EOSMOD/EOSLIB

and load the binary source with their program, that is,

FTN (I = program, GLIB = EOSLIB, ...)

For more advanced users, we list the location of most of the SESAME files which may be useful in complicated situations

<table>
<thead>
<tr>
<th>File Name</th>
<th>Description</th>
<th>CFS File Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>EOSFTN</td>
<td>FORTRAN source of EOSMOD</td>
<td>/EOSMOD/EOSFTN</td>
</tr>
<tr>
<td>EOSINFO</td>
<td>User information</td>
<td>/EOSMOD/EOSINFO</td>
</tr>
<tr>
<td>EOSLIB</td>
<td>Compiled FTN Library file of EOSMOD</td>
<td>/EOSMOD/EOSLIB</td>
</tr>
</tbody>
</table>
### File Names

<table>
<thead>
<tr>
<th>File Name</th>
<th>Description</th>
<th>CFS File Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIXDATA</td>
<td>T-7 mixture EOS and opacity data file</td>
<td>/EOSMOD/MIXLIB</td>
</tr>
<tr>
<td>MIXDIR</td>
<td>directory of MIXLIB</td>
<td>/EOSMOD/MIXDIR</td>
</tr>
<tr>
<td>EOSTST</td>
<td>Test program</td>
<td>/EOSMOD/EOSTST</td>
</tr>
<tr>
<td>MIXB</td>
<td>T-4 procedure to generate mixture</td>
<td>-</td>
</tr>
<tr>
<td>HYDSES</td>
<td>Subroutine package for using SESAME in hydrodynamic codes</td>
<td>-</td>
</tr>
<tr>
<td>SAC</td>
<td>change files in a library</td>
<td>/088077/SES/SAC</td>
</tr>
<tr>
<td>SAX</td>
<td>-</td>
<td>/088077/SES/SAX</td>
</tr>
<tr>
<td>SESAME</td>
<td>unclassified EOS data file</td>
<td>public</td>
</tr>
<tr>
<td>SESAMEA</td>
<td>classified EOS data file</td>
<td>secret</td>
</tr>
<tr>
<td>SESAME</td>
<td>unclassified opacity data file</td>
<td>public</td>
</tr>
<tr>
<td>S2MV2</td>
<td>create, modify, and print EOS data</td>
<td>/SESAME/SEMV2</td>
</tr>
<tr>
<td>S2DV3</td>
<td>plots SESAME data</td>
<td>/SESAME/S2DV3</td>
</tr>
<tr>
<td>S2DHELP</td>
<td>help package for SES2D</td>
<td>/SESAME/S2DHELP</td>
</tr>
<tr>
<td>DSPLX</td>
<td>computes Hugoniots, isentropes and isobars</td>
<td>/SESAME/DSPLX</td>
</tr>
<tr>
<td>S3D</td>
<td>3-D graphics for EOS data</td>
<td>/SESAME/S3D</td>
</tr>
<tr>
<td>LSTX</td>
<td>list of current SESAME materials</td>
<td>/SESAME/LSTX</td>
</tr>
</tbody>
</table>

### VIII. RELATIONSHIPS BETWEEN UNITS

The KUNIT parameter indicates the kind of units the table is to be written in for R, T, P, E, and O. This parameter can have the following integer values and corresponding meanings:

**KUNIT 0** SESAME EOS Units

- R - grams/cm³
- E - megajoules/kilogram (= 10¹⁰ ergs/gram)
- P - gigapascals (= 10¹⁰ dyne/cm²)
- T - degrees Kelvin
- O - cm²/gram

**KUNIT 1** CGS Units

- R - grams/cm³
- E - ergs/gram
- P - microbars (= 1 dyne/cm²)
- T - degrees Kelvin
- O - cm²/gram

**KUNIT 2** Standard International Units (SIU)

- R - kilograms/meter³
- E - joules/kilogram (= 10 ergs/gram)
- P - pascals (= 10 dyne/cm²)
The EOS and opacity tables are scaled according to the numerical value of KUNIT when the tables are copied into LCM. The scaling factors used to convert the tables are defined in subroutine EOSCON listed in the Appendixes. This subroutine can be changed easily by the user to write the tables in units other than those provided automatically by the package.

ACKNOWLEDGMENTS

We wish to thank J. Abdallah, Jr., M. Argo, S. Colgate, W. Huebner, J. D. Johnson, G. Kerley, and S. Lyon for their constructive criticism, advice, and help in developing this package.
REFERENCES


APPENDIX A

SUBROUTINE LISTING

User Callable Routines

EOSDRE (input R and E, output P and T) . . . . . . . . . . . 24
EOSDRT (input R and T, output P and E) . . . . . . . . . . . 28
EOSIPT (input P and T, output R and E) . . . . . . . . . . . 32
EOSIRT (input R and T, output P and E) . . . . . . . . . . . 35
EOSORT (input R and T, output O) . . . . . . . . . . . . . 38
SUBROUTINE EOSORE (LMAT,R,E,P,T,KEOS,IMAT)

******************************************************************************

* PURPOSE -
* GIVEN THE DENSITY (R) AND ENERGY (E) OF A MATERIAL (LMAT)
* THIS SUBROUTINE RETURNS THE PRESSURE (P) AND INTERNAL
* TEMPERATURE (T) USING THE LASL T-4 SESAME EOS Routines

* INPUT VARIABLES -
* LMAT = MATERIAL IN AN A10 FIELD, FOR EXAMPLE- LMAT = "HELIUM"
* THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
* THE MATERIAL BY SETTING LMAT TO THE SESAME
* NUMBER, FOR EXAMPLE- LMAT = "5760" FOR HELIUM

* R = DENSITY (RHO)
* E = INTERNAL ENERGY

* KEOS = MULTIPLE PARAMETER FLAG TO DESCRIBE HOW TO WRITE
* AND RETRIEVE THE DATA FILE. KEOS HAS FOUR DECIMAL DIGITS.

* KEOS = 1000*KBR + 100*KUNIT + 10*KREPE + KFN WHERE

* KBR = COMPUTATIONAL MODE FLAG TO INDICATE WHICH
* QUANTITIES AND THEIR PARTIAL DERIVATIVES ARE TO
* BE CALCULATED AND RETURNED BY THE PACKAGE.
* = 0 COMPUTE PRESSURE AND TEMPERATURE
* = 1 COMPUTE PRESSURE ONLY
* = 2 COMPUTE TEMPERATURE ONLY

* KUNIT= KIND OF UNITS
* 0 (SESAMEE) R-G/CC,T-DEG.K,EV=E-MJ/KG
* 1 (CGS) R-G/CC,T-DEG.K,P-GPA,E-MJ/KG
* 2 (SIU) R-KG/M**3,T-DEG.K,P-GPA,E-MJ/KG
* 3 (HYDROXO) R-G/CC,T-DEG.K,EV=E-MJ/KG
* 4 (HYDROXO) R-G/CC,T-KEV,E-MBR,E-MBRCC/GM,O-M**2/G
* 5 (SESAMED) R-G/CC,T-KEV,O-M**2/G,P-GPA,E-MJ/KG
* 6 (LASNEX) R-G/CC,T-KEV,O-M**2/G,P-JRKS/CC,EJ-JRKS/CC

* LEGEND -
* R = DENSITY
* T = TEMPERATURE
* O = OPACITY
* P = PRESSURE
* E = INTERNAL ENERGY

* CC = CUBIC CENTIMETER
* CM = CENTIMETER
* DEG. K = DEGREES KELVIN
* EV = ELECTRON VOLT
* G = GRAM
* GPA = GIGA PASCALS
* J = JOULES
* MBR = MEGABAR
* M = METER
* PA = PASCAL
* KG = KILOGRAM
KREPE = COMPUTATION FLAG TO INDICATE WHETHER E IS
IS TO BE REPRESENTED AS ENERGY PER UNIT MASS OR ENERGY
PER UNIT VOLUME
O ENERGY IN UNITS OF ENERGY PER UNIT MASS. FOR EXAMPLE-
UNITS OF ERGS/GRAM WHEN KUNIT = 1. THIS IS THE USUAL E.
1 ENERGY IN UNITS OF ENERGY PER UNIT VOLUME. FOR EXAMPLE-
UNITS OF ERGS/CM**3 WHEN KUNIT = 1. THIS IS THE ENERGY
DENSITY RHOE COMMONLY COMPUTED IN HYDRODYNAMIC
COMPUTER CODES.

KFN = KIND OF FUNCTION INTERPOLATION IN THE TABLES
0 RATIONAL APPROXIMATIONS (ACCURATE)
1 BILINEAR APPROXIMATIONS (FAST)

IMATE = INDICATES WHETHER TO LOAD THE DATA FILE IF
IT DOES NOT EXIST FOR LMAT OR GO DIRECTLY TO A
PREVIOUSLY LOADED FILE.

=0 CHECK IF THE DATA FILE FOR LMAT HAS BEEN LOADED.
IF NOT, SEARCH FOR THE FILE, CONVERT IT TO THE PROPER UNITS
AND COPY IT INTO LCM USING THE INVERTED SESAME FORMAT.
>0 EQUAL TO THE LMAT TABLE NUMBER. THIS NUMBER WAS RETURNED
BY THE PACKAGE ON A PREVIOUS CALL TO SUBROUTINE EOSDRE,
EOSIPT OR EOSIRT. THIS OPTION IS FASTER THAN IMATE = 0
BECAUSE IT SKIPS SEARCH IN THE DIRECTORY OF THE

OUTPUT VARIABLES-

P = ARRAY OF DIMENSION 3 CONTAINING THE PRESSURE AND
ITS PARTIAL DERIVATIVES. THIS ARRAY MUST BE DIMENSIONED
EVEN IF THE PARTIAL DERIVATIVES ARE NOT COMPUTED.
P(1) = PRESSURE
P(2) = DENSITY DERIVATIVE OF THE PRESSURE (OP/OR)
P(3) = TEMPERATURE DERIVATIVE OF THE PRESSURE (OP/DE)

T = ARRAY OF DIMENSION 3 CONTAINING THE TEMPERATURE AND
ITS PARTIAL DERIVATIVES. THIS ARRAY MUST BE DIMENSIONED
EVEN IF THE PARTIAL DERIVATIVES ARE NOT COMPUTED
T(1) = INTERNAL TEMPERATURE
T(2) = DENSITY DERIVATIVE OF THE TEMPERATURE (DT/OR)
T(3) = ENERGY DERIVATIVE OF THE TEMPERATURE (DT/DE)

IMATE = INDICATES THE SUCCESS OR FAILURE OF
LOCATING AND LOADING THE DATA FILE FOR LMAT.

= N>0 MATERIAL TABLE NUMBER (SUCCESS)
= 0 MATERIAL (LMAT) NOT IN LIBRARY
-N (N>1) INSUFFICIENT STORAGE
THE LCM STORAGE MUST BE INCREASED BY AT LEAST
N STORAGE LOCATIONS. SEE THE EOSMOD MANUAL

REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

SAMPLE DRIVER PROGRAM-

PROGRAM TST(OUTPUT)
DIMENSION P(3),E(3)
LMAT = "HELIUM"
R = 0.001
E = 1.0
KEOS = 110
IMATE = 0
CALL EOSORE(LMAT,R,E,P,T,KEOS,IMATE)
PRINT 10,P(1)
10 FORMAT(* PRESSURE = *,E10.2,* MICROBARS*)
CALL EXIT

PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

REFERENCE- J. M. HYMAN, M. M. KLEIN
EQUATIONS-OF-STATE AND OPACITIES
LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980

DATE- MARCH 6, 1980

*********
**********

COMMON BLOCKS FOR THE SESAME EOS ROUTINES
LEVEL 2, TBLS
COMMON /S201RX/ LCMX, NRS, LCFW(10,3)
COMMON /SESATX/ TBLS(11000)
COMMON /INTEROX/ KFN
COMMON /SESINX/ IRC, IDT, @HD, ENERGY, KBR, IFL
COMMON /SESOUTX/ PRES(3), TEMP(3)

COMMON BLOCKS FOR THE EOSMOD ROUTINES
COMMON /EDSCZ/ LOUT
COMMON /EDSC3/ INIT, IRDIN, IR(60,3), KUP(60,3)
COMMON /EDSC4/ NTABLE, NTABL0, IFLP, LCNT

DIMENSION T(3), P(3)
DATA KEOS /-99/, KBR, KUNIT, KREPE, KFN, KEOS, KBRS, KFNS, IMATE /0 /

IDT=1 LOCATOR OF DATA TYPE FOR IR, GETINVX(, , IDT,...)

FOR THE INVERTED SESAME FORMAT
IDT=1

CHECK IF THE UNITS HAVE CHANGED SINCE THE LAST CALL
IF(KEOS.SNE.KEDS.OR.LMAT.NE.LMAT) GO TO 5
LMATS=LMAT
KBR=KBRS
KFN=KFNS
GO TO 10

5 CONTINUE

CHECK THE VALIDITY OF THE INPUT PARAMETERS
CALL EOSKUT(KEOS, KBR, KUNIT, KREPE, KFN, KEOS, KBRS, KFNS, IMATE, IDT
IF(IERR.LT.0) GO TO 80
CONTINUE

FIND THE MATERIAL
IF(IMATE.GT.0) GO TO 5
CALL EOSGET(LMAT, KUNIT, KREPE, IMATE, IDT, IERR)
IF(IMATE.LE.0 OR IERR.LT.0) GO TO 80
CONTINUE

TRANSFER INPUT CALL PARAMETERS TO COMMON BLOCK
ENERGY = E
RHO = R
IRC = IR(IMATE, 1)

C
CALCULATE THE EQUATION OF STATE
CALL T4DATIX

* T4DATIX DOES NOT RETURN AN ERROR FLAG FOR DATA OUT OF RANGE
C
RESTORE OUTPUT VARIABLES FOR RETURN

75 CONTINUE
P(1) = PRES(1)
P(2) = PRES(2)
P(3) = PRES(3)
T(1) = TEMP(1)
T(2) = TEMP(2)
T(3) = TEMP(3)

80 CONTINUE
IFLP = IERR
RETURN
END
SUBROUTINE EOSORT (LMAT,R,T,P,E,KEOS,IMATE)

*********************************************************************************************

* PURPOSE -
* GIVEN THE DENSITY (R) AND TEMPERATURE (T) OF A MATERIAL (LMAT)
* THIS SUBROUTINE RETURNS THE PRESSURE (P) AND INTERNAL 
* ENERGY (E) USING THE LASL T-4 SESAME EOS ROUTINES

* INPUT VARIABLES -
* LMAT = MATERIAL IN AN AIO FIELD, FOR EXAMPLE- LMAT = "HELIUM"
* THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
* THE MATERIAL BY SETTING LMAT TO THE SESAME
* NUMBER, FOR EXAMPLE- LMAT = "5760" FOR HELIUM
* R = DENSITY (RHO)
* T = TEMPERATURE
* KEOS = MULTIPLE PARAMETER FLAG TO DESCRIBE HOW TO WRITE
* AND RETRIEVE THE DATA FILE. KEOS HAS FOUR DECIMAL DIGITS.
* KBR = COMPUTATIONAL MODE FLAG TO INDICATE WHICH
* QUANTITIES AND THEIR PARTIAL DERIVATIVES ARE TO
* BE CALCULATED AND RETURNED BY THE PACKAGE.
* = 0 COMPUTE PRESSURE AND TEMPERATURE
* = 1 COMPUTE PRESSURE ONLY
* = 2 COMPUTE TEMPERATURE ONLY

* KUNIT = KIND OF UNITS
0 (SESAME) R-G/CC,T-DEG.K,D-CM**2/G,P-GPA,E-MJ/KG
1 (CGS) R-G/CC,T-DEG.K,P-MA, E-MBR, D=MBR/G, E-MBR/CM**2/G
2 (SI) R-KG/M**3,T-DEG.K,P-PA, E-MBR, D=MBR/CM**2/G
3 (HYDROX) R-G/CC,T-DEG.K,P=MBR, E-MBR/CC/G, D=MBR**2/G
4 (HYDROX) R-G/CC,T-KEV,P=MBR, E-MBR/CC/G, D=MBR**2/G
5 (SESAME) R-G/CC,T-EV,0-CM**2/G,P-GPA, E-MJ/KG
6 (LASNEX) R-G/CC,T-KEV,0-CM**2/G, P-JRKS/CC, E=JRKS/CC

LEGALND -
R = DENSITY
T = TEMPERATURE
D = OPACITY
P = PRESSURE
E = INTERNAL ENERGY
CC = CUBIC CENTIMETER
CM = CENTIMETER
DEG. K = DEGREES KELVIN
EV = ELECTRON VOLT
G = GRAM
GPA = GIGA PASCALS
J = JOULES
MBR = MEGABAR
M = METER
MUBR = MICROBAR
PA = PASCAL

KREPE = COMPUTATION FLAG TO INDICATE WHETHER E IS
IS TO BE REPRESENTED AS ENERGY PER UNIT MASS OR ENERGY
PER UNIT VOLUME
0 ENERGY IN UNITS OF ENERGY PER UNIT MASS. FOR EXAMPLE:
UNITS OF ERGS/GRAM WHEN KUNIT = 1. THIS IS THE USUAL E.
1 ENERGY IN UNITS OF ENERGY PER UNIT VOLUME. FOR EXAMPLE:
UNITS OF ERGS/CM**3 WHEN KUNIT = 1. THIS IS THE ENERGY
DENSITY rho*E COMMONLY COMPUTED IN HYDRODYNAMIC
COMPUTER CODES.

KFN = KIND OF FUNCTION INTERPOLATION IN THE TABLES
0 RATIONAL APPROXIMATIONS (ACCURATE)
1 BILINEAR APPROXIMATIONS (FAST)

IMATE = INDICATES WHETHER TO LOAD THE DATA FILE IF
IT DOES NOT EXIST FOR LMAT OR GO DIRECTLY TO A
PREVIOUSLY LOADED FILE.
0 CHECK IF THE DATA FILE FOR LMAT HAS BEEN LOADED.
IF NOT, SEARCH FOR THE FILE, CONVERT IT TO THE PROPER UNITS
AND COPY IT INTO LCM USING THE STANDARD SESAME FORMAT.
>0 EQUAL TO THE LMAT TABLE NUMBER. THIS NUMBER WAS RETURNED
BY THE PACKAGE ON A PREVIOUS CALL TO SUBROUTINE EOSORE.
EOSIPT OR EOSIRT. THIS OPTION IS FASTER THAN IMATE = 0
BECAUSE IT SKIPS SEARCH IN THE DIRECTOR OF THE

OUTPUT VARIABLES-

P = ARRAY OF DIMENSION 3 CONTAINING THE PRESSURE AND
ITS PARTIAL DERIVATIVES. THIS ARRAY MUST BE DIMENSIONED
EVEN IF THE PARTIAL DERIVATIVES ARE NOT COMPUTED.
P(1) = PRESSURE
P(2) = DENSITY DERIVATIVE OF THE PRESSURE (DP/OR)
P(3) = TEMPERATURE DERIVATIVE OF THE PRESSURE (DP/OT)

E = ARRAY OF DIMENSION 3 CONTAINING THE ENERGY AND
ITS PARTIAL DERIVATIVES. THIS ARRAY MUST BE DIMENSIONED
EVEN IF THE PARTIAL DERIVATIVES ARE NOT COMPUTED
E(1) = INTERNAL ENERGY
E(2) = DENSITY DERIVATIVE OF THE ENERGY (DE/OR)
E(3) = TEMPERATURE DERIVATIVE OF THE ENERGY (DE/OT)

IMATE = INDICATES THE SUCCESS OR FAILURE OF
LOCATING AND LOADING THE DATA FILE FOR LMAT.

LMAT = "HELIUM"
R = 0.001

REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

PROGRAM TST(OUTPUT)
DIMENSION P(3),E(3)
LMAT = "HELIUM"
* T = 300.0
* KEOS = 110
* IMATE = 0
* CALL EOSORT(LMAT,R,T,P,E,KEOS,IMATE)
* PRINT 10,P(1)
* 10 FORMAT(" PRESSURE = "E10.2," MICROBARS")
* CALL EXIT
* END
* REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE
* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL
* REFERENCE- J. M. HYMAN, M. M. KLEIN
* EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
* EQUATIONS-OF-STATE AND OPACITIES
* LOS ALAMOS SCIENTIFIC LABORATORY RPT.,LA-8502-M,1980
* DATE- MARCH 6, 1980
* COMMON BLOCKS FOR THE SESAME EOS ROUTINES
* LEVEL 2. TLS
* COMMON /S20IRX/ LCMX, NRS, LCFW(10,3)
* COMMON /SESDATX/ TLS(11000)
* COMMON /INTORDX/ KFN
* COMMON /SESINX/ IRC, IOT, RHO, TEMP, KBR, IFL
* COMMON /SESOUTX/ PRES(3), ENERGY(3)
* COMMON BLOCKS FOR THE EOSMOO ROUTINES
* COMMON /EOSC2/ LOUT
* COMMON /EOSCCE/ TFAC, RFAC, PFACE, EFACE, KREPE
* COMMON /EOSC1/ LU41, LU42, LU43, LU44, LU45
* COMMON /EOSC3/ INIT, IRDIM, IR(60,3), KUP(60,3)
* COMMON /EOSC4/ NTABLE, NTABLO, IFLP, LCNT
* DIMENSION E(3), P(3)
* DATA KEOS /-99/,KBRS/O/,KFNS/O/,LMATS/1H /
* COMMON")E3,P3"
* IOT=3 LOCATOR OF DATA TYPE FOR IR,GETEOSX(., IOT,....)
* FOR THE STANDARD SESAME FORMAT
* IOT=3
* CHECK IF THE UNITS HAVE CHANGED SINCE THE LAST CALL
* IF(KEOSS.NE.KEOS.OR.LMAT.NE.LMAT) GO TO 5
* LMATS=LMAT
* KBR=KBRS
* KFN=KFNS
* GO TO 10
* 5 CONTINUE
* CHECK THE VALIDITY OF THE INPUT PARAMETERS
* CALL EOSKUT(KEOS,KBR,KUNIT,KREPE,KFN,KEOSS,KBRS,KFNS,IMATE,IOT
* 1,IERR)
* IF(IERR.LT.0) GO TO 80
* 10 CONTINUE
* FIND THE MATERIAL
* IF (IMATE.GT.0) GO TO 60
CALL EOSGET(LMAT,KUNIT,KREPE,IMATE,IOT,IERR)
IF(IMATE.LE.0.OR.IERR.LT.0) GO TO 80
60 CONTINUE

TRANSFER INPUT CALL PARAMETERS TO COMMON BLOCK
TEMP=T
RHO=R
IRC=IR(IMATE,3)

CALCULATE THE EQUATION OF STATE
CALL T4DATX
* T4DATX DOES NOT RETURN AN ERROR FLAG FOR DATA OUT OF RANGE

75 CONTINUE

RESTORE OUTPUT VARIABLES FOR RETURN TO CALL
P(1)=PRES(1)
P(2)=PRES(2)
P(3)=PRES(3)
E(1)=ENERGY(1)
E(2)=ENERGY(2)
E(3)=ENERGY(3)

80 CONTINUE
IFLP=IERR
RETURN
END
SUBROUTINE EOSIPT (LMAT,P,T,E,KEOS,IMATE)

* PURPOSE -
* GIVEN THE PRESSURE (P) AND TEMPERATURE (T) OF A MATERIAL (LMAT)
* THIS SUBROUTINE RETURNS THE PRESSURE (P) AND INTERNAL
* ENERGY (E) USING THE LASL T-4 SESAME EQUATION OF STATE ROUTINES
* AN ITERATIVE METHOD IS USED TO INTERPOLATE THE TABLES

* INPUT VARIABLES -
* LMAT = MATERIAL IN AN AIO FIELD, FOR EXAMPLE - LMAT = "HELIUM"
* THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
* THE MATERIAL BY SETTING LMAT TO THE SESAME
* NUMBER, FOR EXAMPLE - LMAT = "5760" FOR HELIUM

* P = PRESSURE
* T = TEMPERATURE
* KEOS = MULTIPLE PARAMETER FLAG TO DESCRIBE HOW TO WRITE
* AND RETRIEVE THE DATA FILE. KEOS HAS FOUR DECIMAL DIGITS,

* KEDS = 100*KUNIT + 10*KREPE WHERE

* KUNIT = KIND OF UNITS
* 0 (SESAME) R-G/CC,T-DEG.K,D-CM**2/G,P-GPA,E-MJ/KG
* 1 (CGS) R-G/CC,T-DEG.K,D-CM**2/GM,P-MBAR,E-ERGS/GM
* 2 (SIU) R-KG/M**3,T-DEG.K,P-PA,E-J/KG,D-M**2/PA
* 3 (HYDROX) R-G/CC,T-DEG,K,P-MBAR,E-MBAR*CC/GM,D-CM**2/G
* 4 (HYDROX0) R-G/CC,T-KEV,P-MBAR,E-MBAR*CC/GM,D-CM**2/G
* 5 (SESAMEO) R-G/CC,T-KEV,D-CM**2/G,P-GPA,E-MJ/KG
* 6 (LASNEX) R-G/CC,T-KEV,D-CM**2/GM,P-JRKS/CC,ED-JRKS/CC

* LEGEND -
* R = DENSITY
* T = TEMPERATURE
* D = OPACITY
* P = PRESSURE
* E = INTERNAL ENERGY
* CC = CUBIC CENTIMETER
* CM = CENTIMETER
* DEG. K = DEGREES KELVIN
* EV = ELECTRON VOLT
* G = GRAM
* GPA = GIGA PASCALS
* J = JOULES
* JRKS = JERKS
* KEV = KILO ELECTRON VOLTS
* KG = KILOGRAM
* M = METER
* MBAR = MEGABAR
* MUBR = MICROBAR
* PA = PASCAL

* KREPE = COMPUTATION FLAG TO INDICATE WHETHER E IS
* IS TO BE REPRESENTED AS ENERGY PER UNIT MASS OR ENERGY
* PER UNIT VOLUME

* O ENERGY IN UNITS OF ENERGY PER UNIT MASS. FOR EXAMPLE -
UNITS OF ERGS/GRAM WHEN KUNIT = 1. THIS IS THE USUAL ENERGY UNIT.

1 ENERGY IN UNITS OF ENERGY PER UNIT VOLUME. FOR EXAMPLE:

UNITS OF ERGS/CM^3 WHEN KUNIT = 1. THIS IS THE ENERGY DENSITY rho*E COMMONLY COMPUTED IN HYDRODYNAMIC COMPUTER CODES.

IMATE = INDICATES WHETHER TO LOAD THE DATA FILE IF IT DOES NOT EXIST FOR LMAT OR GO DIRECTLY TO A PREVIOUSLY LOADED FILE.

=0 CHECK IF THE DATA FILE FOR LMAT HAS BEEN LOADED.

IF NOT, SEARCH FOR THE FILE, CONVERT IT TO THE PROPER UNITS AND COPY IT INTO LCM USING THE INVERTED SESAME FORMAT.

>0 EQUAL TO THE LMAT TABLE NUMBER. THIS NUMBER WAS RETURNED BY THE PACKAGE ON A PREVIOUS CALL TO SUBROUTINE EOSDRE, EOSIPT OR EOSIRT. THIS OPTION IS FASTER THAN IMATE = 0 BECAUSE IT SKIPS SEARCH IN THE DIRECTORY OF THE INPUT VARIABLES:

R = DENSITY

E = INTERNAL ENERGY

IMATE = INDICATES THE SUCCESS OR FAILURE OF LOCATING AND LOADING THE DATA FILE FOR LMAT.

N>0 MATERIAL TABLE NUMBER (SUCCESS)

O MATERIAL (LMAT) NOT IN LIBRARY

-N (N>1) INSUFFICIENT STORAGE

THE LCM STORAGE MUST BE INCREASED BY AT LEAST N STORAGE LOCATIONS. SEE THE EOSMOD MANUAL.

REMARKS - THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

SAMPLE DRIVER PROGRAM:

PROGRAM TST(OUTPUT)
DIMENSION P(3),E(3)
LMAT = "HELIUM"
R = 0.001
T = 300.0
KEDS = 110
IMATE = 0
CALL EOSIPT(LMAT,P,T,E,KEDS,IMATE)
CALL EXIT
END

REMARKS - THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

PROGRAMMER - J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

REFERENCE - J. M. HYMAN, M. M. KLEIN

EOSMOD - A SUBROUTINE PACKAGE FOR CALCULATING EQUATIONS-OF-STATE AND OPACITIES

ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980

DATE - MARCH 6, 1980

***********************************************
COMMON BLOCKS FOR THE SESAME EOS ROUTINES

LEVEL 2, TBLS
COMMON /S20IRX/ LCMX, NRS, LCFW(10,3)
COMMON /SESDATX/ TBLS(11000)
COMMON /SESINX/ DUM(4), KBR, DUM1
COMMON /INTORDX/ KFN

COMMON BLOCKS FOR THE EOSMOD ROUTINES
COMMON /EOSCZ/ LOUT
COMMON /EOSC3/ INIT, IRDIM, IR(60,3), KUP(60,3)
COMMON /EOSC4/ NTABLE, NTABLO, IFLP, LCNT

DIMENSION R(3), E(3)
DATA KEOS /-99/, KBRS/O/, KFNS/O/, LMATS/ 1/

*******************************************************************************
*******************************************************************************
IDT=1 LOCATOR OF DATA TYPE FOR IR,GETINVX(, , IDT, ...)
FOR THE INVERTED SESAME FORMAT
IDT=1

CHECK IF THE UNITS HAVE CHANGED SINCE THE LAST CALL
IF(KEOS.NE.KEOS.OR.LMAT.NE.LMATS) GO TO 5
LMATS=LMAT
KBR=KBRS
KFN=KFNS
GO TO 10
CONTINUE

5 CONTINUE

CHECK THE VALIDITY OF THE INPUT PARAMETERS
CALL EOSKUT(KEOS, KBR, KUNIT, KREPE, KFN, KEOS, KBRS, KFNS, IMATE, IDT
1 , IERR)
IF(IERR.LT.0) GO TO 75

10 CONTINUE

FIND THE MATERIAL
IF (IMATE.GT.0) GO TO 60
CALL EDSGET(LMAT, KUNIT, KREPE, IMATE, IDT, IERR)
IF(IMATE.LE.O.DR. IERR.LT .0) Go To 75
CONTINUE

CALCULATE THE EQUATION OF STATE
CALL T4PTRLX (IR(IMATE, 1), 1, TBLS, P, T, R, E, IERR)

PRINT AN ERROR MESSAGE IF T4PTRLX FAILS TO CONVERGE
IF (IERR.EQ.0) WRITE(LOUT, 80) LMAT, P, T
IF (IERR.EQ.0) IMATE=O
IFLP=IERR
RETURN

FORMAT(FAILD TO CONVERGE WHEN ITERATING ON THE INVERTED*,
1 ./, " EOS TABLES IN SUBROUTINE T4PTRLX CALLED BY EOSIPT*/,
2 " THE REQUESTED VALUES FOR RHO, PRESSURE, TEMPERATURE AND ENERGY" 
2 ./, " MAY BE OUT OF RANGE OR NEAR THE EDGE OF THE TABLE*/,
3 " CHECK THE EOSMOD WRITEUP FOR THE DATA RANGE*/,
1 " LMAT **,A10,
4 " PRESSURE =", 1PE12.4, " TEMPERATURE =", 1PE12.4)
END
SUBROUTINE EOSIRT (LMAT, R, T, P, E, KEOS, IMATE)

*--------------------------------------------------------------------------*

PURPOSE -
GIVEN THE DENSITY (R) AND TEMPERATURE (T) OF A MATERIAL, THIS SUBROUTINE RETURNS THE PRESSURE (P) AND INTERNAL ENERGY (E) USING THE LASL E-4 SESAME EQUATION OF STATE ROUTINES.
AN ITERATIVE METHOD IS USED TO INTERPOLATE THE TABLES.
SUBROUTINE EOSORT USES A FASTER DIRECT METHOD.

INPUT VARIABLES -
LMAT = MATERIAL IN AN AO FIELD, FOR EXAMPLE- LMAT = "HELIUM"
THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
THE MATERIAL BY SETTING LMAT TO THE SESAME NUMBER, FOR EXAMPLE- LMAT = "5760" FOR HELIUM
R = DENSITY (RHO)
T = TEMPERATURE
KEOS = MULTIPLE PARAMETER FLAG TO DESCRIBE HOW TO WRITE
AND RETRIEVE THE DATA FILE. KEOS HAS FOUR DECIMAL DIGITS,
KEDS = 100*KUNIT + 10*KREPE WHERE
KUNIT = KIND OF UNITS
O (SESAME) R-G/CC, T-DEG.K, O- CM**2/G, P-GPA, E-MJ/KG
1 (CGS) R-G/CC, T-DEG.K, O- CM**2/GM, P-MBAR, E-ERGS/GM
2 (SI) R-KG/M**3, T- DEG.K, P-PA, E-J/KG, O-M**2/2/KG
3 (HYDROX) R-G/CC, T- DEG.K, P- MBR, E-MBR*CC/GM, O- CM**2/2/G
4 (HYDROXO) R-G/CC, T-KEV, P- MBR, E-MBR*CC/GM, O- CM**2/2/G
5 (SESAMEO) R-G/CC, T-EV, O- CM**2/2/G, P-GPA, E-MJ/KG
6 (LASNEX) R-G/CC, T-KEV, O- CM**2/2/GM, P-JRKS/CC, ED-JRKS/CC

LEGEND -
R = DENSITY
T = TEMPERATURE
O = OPACITY
P = PRESSURE
E = INTERNAL ENERGY
CC = CUBIC CENTIMETER
CM = CENTIMETER
DEG. K = DEGREES KELVIN
EV = ELECTRON VOLT
G = GRAM
GPA = GIGA PASCALS
J = JOULES
JRKS = JERKS
KEV = KILO ELECTRON VOLTS
KG = KILOGRAM
M = METER
MBR = MEGABAR
MUBR = MICROBAR
PA = PASCAL

KEPE = COMPUTATION FLAG TO INDICATE WHETHER T IS
IS TO BE REPRESENTED AS ENERGY PER UNIT MASS OR ENERGY
PER UNIT VOLUME.
ENERGY IN UNITS OF ENERGY PER UNIT MASS. FOR EXAMPLE-
UNITS OF ERGS/GRAM WHEN KUNIT = 1. THIS IS THE USUAL T.
ENERGY IN UNITS OF ENERGY PER UNIT VOLUME. FOR EXAMPLE-
UNITS OF ERGS/CM**3 WHEN KUNIT = 1. THIS IS THE ENERGY
DENSITY RHO*T COMMONLY COMPUTED IN HYDRODYNAMIC
COMPUTER CODES.

IMATE = INDICATES WHETHER TO LOAD THE DATA FILE IF
IT DOES NOT EXIST FOR LMAT OR GO DIRECTLY TO A
PREVIOUSLY LOADED FILE.
"O CHECK IF THE DATA FILE FOR LMAT HAS BEEN LOADED.
IF NOT, SEARCH FOR THE FILE, CONVERT IT TO THE PROPER UNITS
AND COPY IT INTO LCM USING THE INVERTED SESAME FORMAT.
EQUAL TO THE LMAT TABLE NUMBER. THIS NUMBER WAS RETURNED
BY THE PACKAGE ON A PREVIOUS CALL TO SUBROUTINE EOSORE.
EOSIPT OR EOSIIRT. THIS OPTION IS FASTER THAN IMATE = 0
BECAUSE IT SKIPS SEARCH IN THE DIRECTOR OF THE

OUTPUT VARIABLES-

P = PRESSURE

E = INTERNAL ENERGY

IMATE = INDICATES THE SUCCESS OR FAILURE OF
LOCATING AND LOADING THE DATA FILE FOR LMAT.

N+O MATERIAL TABLE NUMBER (SUCCESS)
0 MATERIAL (LMAT) NOT IN LIBRARY
-N (N>1) INSUFFICIENT STORAGE
THE LCM STORAGE MUST BE INCREASED BY AT LEAST
N STORAGE LOCATIONS. SEE THE EOSMOD MANUAL

REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

SAMPLE DRIVER PROGRAM-

PROGRAM TST(OUTPUT)
DIMENSION P(3),T(3)
LMAT = "HELIUM"
R = 0.001
T = 300.0
KEDS = 110
IMATE = 0
CALL EOSIPT(LMAT,R,T,P,E,KEDS,IMATE)
PRINT 10,P(1)
CALL EXIT
END

REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL
REFERENCE- J. M. HYMAN, M. M. KLEIN
EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
EQUATIONS-OF-STATE AND OPACITIES
LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980

DATE- MARCH 6, 1980
COMMON BLOCKS FOR THE SESAME EOS ROUTINES

LEVEL 2, TBLS
COMMON /S2DIRX/ LCMX, NRS, LCFW(10,3)
COMMON /SESOATX/ TBL(S(11000)
COMMON /SESINX/ DUM(4), KBR, DUM1
COMMON /INTORDX/ KFN

COMMON BLOCKS FOR THE EOSMOD ROUTINES
COMMON /EDSCZ/ LOUT
COMMON /EDSC3/ INIT, IRDIM, IR(60,3), KUP(60,3)
COMMON /EDSC4/ NTABLE, NTABLO, *IFLP, LCNT

DIMENSION P(3), E(3)
DATA KEDSS /-99/, KBR/0/, KFN/0/, LMA/1/H /

******************************************************************************

IOT=1 LOCATOR OF DATA TYPE FOR IR, GETINVX(., IOT, ...)
FOR THE INVERTED SESAME FORMAT
IOT=1

CHECK IF THE UNITS HAVE CHANGED SINCE THE LAST CALL
IF(KEDSS.NE.KEOS.OR.LMAT,NE.LMATS) GO TO 5
LMATS=LMAT
KBR=KBRS
KFN=KFNS
GO TO 10

5 CONTINUE

CHECK THE VALIDITY OF THE INPUT PARAMETERS
CALL EOSKUT(KEOS, KBR, KUNIT, KREPE, KFN, KEDSS, KEDS, KFNS, IMATE, IOT
1, IERR)
IF(IERR.LT.0) GO TO 75
CONTINUE

FINO THE MATERIAL
IF (IMATE.GT.0) Go To 60
CALL EOSGET(LMAT, KUNIT, KREPE, IMATE, IOT, IERR)
IF(IMATE.LE.0.OR. IERR.LT.0) GO TO '75
CONTINUE

CALCULATE THE EQUATION OF STATE
CALL T4RTPEX (IR(IMATE, 1), 1, TBLS, R, T, P, E, IERR)

PRINT AN ERROR MESSAGE IF T4RTPEX FAILED TO CONVERGE
IF (IERR.EQ.0) WRITE(LOUT, 80) LMAT, R, T
IF (IERR.EQ.0) IMATE=0
CONTINUE

IFLP=IERR
RETURN

80 FORMAT(" FAILED TO CONVERGE WHEN ITERATING ON THE INVERTED".
1 */ " EOS TABLES IN SUBROUTINE T4RTPEX CALLED BY EDSIRT"./,
2 " THE REQUESTED VALUES FOR RHO, PRESSURE, TEMPERATURE AND ENERGY".
2 */ " MAY BE OUT OF RANGE OR NEAR THE EDGE OF THE TABLE"./,
3 " CHECK THE EOSMOD WRITEUP FOR THE DATA RANGE"./,
1 " LMAT =", A10,
4 " DENSITY =", 1PE12.4, " TEMPERATURE =", 1PE12.4)
END
SUBROUTINE EOSDR1 (LMAT,R,T,O,KOPC,IMAT)

***************
PURPOSE -
GIVEN THE DENSITY (R) AND TEMPERATURE (T) OF A MATERIAL (LMAT) THIS ROUTINE RETURNS THE OPACITY (O) USING THE LASL HYDSES (T-4) ROUTINES

INPUT VARIABLES -
LMAT = MATERIAL IN AN A10 FIELD, FOR EXAMPLE - LMAT = "HELIUM"
THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY THE MATERIAL BY SETTING LMAT TO THE SESAME NUMBER, FOR EXAMPLE - LMAT = "5760" FOR HELIUM

R = DENSITY (RHO)
T = TEMPERATURE

KOPC = MULTIPLE PARAMETER FLAG TO DESCRIBE HOW TO WRITE AND RETRIEVE THE DATA FILE. KOPC HAS FOUR DECIMAL DIGITS.

KOPC = 100*KUNIT + 10*KREPC + KFN WHERE
KUNIT= KIND OF UNITS
0 (SESAME) R-G/CC,T-DEG.K,O-CM**2/G,P-GPA,E-MJ/KG
1 (CGS) R-G/CC,T-DEG.K,O-CM**2/GM,P-MBR,E-ERGS/GM
2 (SIU) R-KG/M**3,T-DEG.K,P-PA,E-J/KG,O-M**2/2/KG
3 (HYDROX) R-G/CC,T-DEG.K,P-MBR,E-MBR+CC/GM,O-CM**2/G
4 (HYDROXO) R-G/CC,T-KEV,P-MBR,E-MBR+CC/G,O-CM**2/2/G
5 (SESAMEO) R-G/CC,T-KEV,O-CM**2/2/G,P-GPA,E-MJ/KG
6 (LASNEX) R-G/CC,T-KEV,O-CM**2/2/GM,P-JRKS/CC,ED-JRKS/CC

LEGEND -
R = DENSITY
T = TEMPERATURE
O = OPACITY
P = PRESSURE
E = INTERNAL ENERGY
CC = CUBIC CENTIMETER
CM = CENTIMETER
DEG. K = DEGREES KELVIN
EV = ELECTRON VOLT
G = GRAM
GPA = GIGA PASCALS
J = JOULES
JRKS = JERKS
KEV = KILO ELECTRON VOLTS
KG = KILOGRAM
M = METER
MBR = MEGABAR
MUBR = MICROBAR
PA = PASCAL

KREPO = COMPUTATIONAL FLAG TO INDICATE WHICH REPRESENTATION TO USE FOR THE OPACITY VARIABLE.

0 OPACITY REPRESENTED AS KAPPA IN DIMENSIONAL UNITS OF LENGTH**2/MASS
1 OPACITY REPRESENTED AS A MEAN-FREE PATH, LAMBD = 1/(KAPPA*RHO), IN DIMENSIONAL
UNITS OF LENGTH.

KFN = KIND OF FUNCTION INTERPOLATION IN THE TABLES

= 0 RATIONAL APPROXIMATIONS (ACCURATE)

= 1 BILINEAR APPROXIMATIONS (FAST)

IMATO = INDICATES WHETHER TO LOAD THE DATA FILE IF

IT DOES NOT EXIST FOR LMAT OR GO DIRECTLY TO A

PREVIOUSLY LOADED FILE.

=0 CHECK IF THE DATA FILE FOR LMAT HAS BEEN LOADED.

IF NOT, SEARCH FOR THE FILE, CONVERT IT TO THE PROPER UNITS

AND COPY IT INTO LCM USING THE INVERTED SESAME FORMAT.

>0 EQUAL TO THE LMAT TABLE NUMBER. THIS NUMBER WAS RETURNED

BY THE PACKAGE ON A PREVIOUS CALL TO SUBROUTINE EDSORT.

THIS OPTION IS FASTER THAN IMATO = 0

BECAUSE IT SKIPS SEARCH IN THE DIRECTOR OF THE

OUTPUT VARIABLES-

D = OPACITY

IMATO = INDICATES THE SUCCESS OR FAILURE OF

LOCATING AND LOADING THE DATA FILE FOR LMAT.

= N>0 MATERIAL TABLE NUMBER (SUCCESS)

0 MATERIAL (LMAT) NOT IN LIBRARY

-N (N>1) INSUFFICIENT STORAGE

THE LCM STORAGE MUST BE INCREASED BY AT LEAST

N STORAGE LOCATIONS. SEE THE EOSMOD MANUAL

REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

SAMPLE DRIVER PROGRAM-

PROGRAM TST(OUTPUT)

DIMENSION P(3),E(3)

LMAT = "HELIUM"

R = 0.001

T = 300.0

KDPC = 500

IMATO = 0

CALL EDSORT(LMAT,R,T,O,KDPC,IMATO)

PRINT 10,P(1)

FORMAT(" OPACITY = ",E10.2)

CALL EXIT

END

EXTERNALS AND COMMON BLOCKS-

*LOG10 = QUICK VERSION OF ALOG10

REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7 LASL

REFERENCE- J. M. HYMAN, M. M. KLEIN

EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING

EQUATIONS-OF-STATE AND OPACITIES

LDS ALAMOS SCIENTIFIC LABORATORY RPT.,LA-8502-M,1980

DATE- MARCH 6, 1979

***********************************************************************
COMMON BLOCKS FOR THE SESAME EOS ROUTINES

DIMS T8L, LCMX, NLBUF, LCFW(), ZZ()

LEVEL 2, TBL

COMMON /S2IRX/, LCMX, NRS, LCFW(10, 3)

COMMON /SESINX/, TBL(11000)

COMMON /INTORDX/, KFN

COMMON /SESINX/, IRC, IDT, RHO, TEMP, KBR, IFL

COMMON /SESOATX/, OPACITY(3), PLANKD(3)

COMMON BLOCKS FOR THE ESMOD Routines

COMMON /EDSC/ LOUT

/sEsOATX/ T8Ls(fwo0)

/sEsOATX/ LcMX. NRS. LcFW(~o.3)

/INTORDX/ KFN

/SESouTx/ opacity. pLANKo(3)

8LOCKS FOR THE EOSMOO ROUTINES

/EOSCZ/ LOUT

/EOSCco/ TFACO, RFACO, DFACO, KREPO

/EOSC3/ INIT, IROI, IR(60, 3), KUP(60, 3)

/EOSC3/ INIT, IROI, IR(60, 3), KUP(60, 3)

DATA KOPCS /-99/, KFNS/O/, LMATS/IH /

******************************************************************************

CHECK POSITIVITY OF INPUT PARAMETERS

IF ((T.GT.O.O).AND. (R.GT.O.O)) GO TO 4

fMATO=-1

WRITE(LOUT, 20) R, T

20 FORMAT(" THE OENSITY =", 1E12.4, " OR THE TEMPERATURE ="

1.E12.4, " IS NONPOSITIVE")

GO TO 75

4 CONTINUE

SET INITIAL VARIABLES FOR OPACITY TABLES

IOT=2 LOCATOR OF DATA TYPE FOR IR AND SUB. GETPRX(., IDT, ...)

IDT=2

CHECK IF THE UNITS HAVE CHANGED SINCE THE LAST CALL

IF (KOPCS.NE.KOPC.OR.LMAT .NE.LMATS) GO TO 5

KFN=KFNS

LMATS=LMAT

KOPCS=KOPC

GO TO 10

5 CONTINUE

UNSCRAMBLE MULTIPLE FLAG KOPC

CALL EOSKUT(KOPC, KOMMY, KUNIT, KREPO, KFNS, KOPCS, KFNS, IMATO, IDT

1 ,IERR)

IF (IERR.LT.0) GO TO 75

CONTINUE

10 CONTINUE

IF (IERR.LT.0) GO TO 75

CONTINUE

FIND THE MATERIAL

IF (IMATO.GT.0) GO TO 60

CALL EDSGET(LMAT, KUNIT, KREPO, IMATO, IOT, IERR)

IF (IMATO.LE.0 OR IERR.LT.0) GO TO 75

CONTINUE

10 CONTINUE

TRANSFER INPUT CALL PARAMETERS TO COMMON BLOCK

TEMP=QLOG10(T)

RHO=QLOG10(R)

KBR=1

IRC=IR(IMATO, 2)
CALCULATE THE EQUATION OF STATE
CALL T4OATX
T4OATIX DOES NOT RETURN AN ERROR FLAG
FOR DATA OUT OF BOUNDS
RESTORE OUTPUT VARIABLES FOR RETURN TO CALL
IFLP=IERR
D=10.**OPACITY(1)
CONTINUE
RETURN
END
**INTERNAL SUBROUTINES**

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>EOSBEG (initializes all the common block variables)</td>
<td>43</td>
</tr>
<tr>
<td>EOSCON (defines the table conversion factors)</td>
<td>46</td>
</tr>
<tr>
<td>EOSDSL (allows the user to scale the density)</td>
<td>49</td>
</tr>
<tr>
<td>EOSEFD (finds the EOS table)</td>
<td>51</td>
</tr>
<tr>
<td>EOSFAS (assigns the input files)</td>
<td>54</td>
</tr>
<tr>
<td>EOSGET (loads the Sesame EOS data files)</td>
<td>56</td>
</tr>
<tr>
<td>EOSKUT (checks the validity of KPARM)</td>
<td>59</td>
</tr>
<tr>
<td>EOSOFD (finds the opacity table)</td>
<td>61</td>
</tr>
</tbody>
</table>
SUBROUTINE EOSBEG

*******************************************************************************

* PURPOSE -
* TO INITIALIZE ALL COMMON BLOCKS IN ONE PLACE OF CODE

* INPUT VARIABLES -
* NONE

* OUTPUT VARIABLES -
* ALL OUTPUT IS AT COMPILATION TIME IN THE COMMON BLOCKS.
* THIS ALLOWS THE USER AN EASY WAY TO CHANGE THE VARIABLES
* BY SETTING THEM TO ANY OTHER VALUE AT EXECUTION
* -- - - - - - --- - - - . ------ - - . -- - ---- ------ - ------ - - ----- - - . ---- - - .

* LOCALLY DEFINED SESAME VARIABLES -
* TBLS = ARRAY FOR STORAGE OF THE EOS TABLES
* LCNx = LENGTH OF THE TBLS ARRAY
* NRS = UPPER BOUND ON THE NUMBER OF MAT REGIONS LCFW(NRS.)
* LCFW = ARRAY USED AS A DIRECTORY BY THE SESAME ROUTINES
* IRC = MATERIAL REGION NUMBER
* IRC = IR (DEFINED TO PERMIT SUBROUTINE CALL
* IDS2 = SESAME MATERIAL NUMBER
* TBLS = NAME OF AN ARRAY DESIGNATED FOR THE STORAGE OF TABLES
* LCNT = CURRENT WORD IN THE ARRAY TBLS
* LU41 = UNIT NUMBER ASSIGNED TO THE SESAME INPUT FILE SES2CL
* LU42 = UNIT NUMBER ASSIGNED TO THE SESAME INPUT FILE SESAME
* LU44 = UNIT NUMBER ASSIGNED TO MIXLIB (MIXTURES)
* LU45 = UNIT NUMBER ASSIGNED TO MIXLIB DIRECTORY.
* KFN = 0 RATIONAL APPROXIMATIONS (ACCURATE)
* KFN = 1 BILINEAR APPROXIMATIONS (FAST)
* ZB (OUTPUT FROM GETINVX) AT. CHARGE, AT. CHG**2, MASS
* IOT = DATA TYPE INDICATOR
* MID (ID) MATERIAL ID = 1 INVERSE TABLES = 2 OPACITY

* EXTERNAL FILES AND COMMON BLOCKS -
* SESAME ROUTINES- S2GET,S2EOS
* SESAME ROUTINES MATCHXKX, TABRANX, INBUFRX, OPCXXKX, ISRCKXK.
* T4INTPX, GETINVX, RATFN1X, T40ATIX, INV301X, T4RTPEX
* SESAME COMMON BLOCKS-SZ0IRX, RTBLK2X, SESOAATX, SESINX, SESOUTX, INTORDX
* ESMOD COMMON BLOCKS- EOSC1,-7
* ESMOD COMMON (ALSO INSERTED INTO GETINVX) ESDCCE, EOSCCO

* REMARKS - THIS SUBROUTINE IS PART OF THE ESMOD PACKAGE

* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

* REFERENCE- J. M. HYMAN, M. M. KLEIN

**EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
* EQUATIONS-OF-STATE AND OPACITIES
* LOS ALAMOS SCIENTIFIC LABORATORY RPT...LA-8502-M.1980
**COMMON BLOCKS FOR THE SESAME EOS ROUTINES**

- `DIMS TBLS, LCNX, NLIBUF, LCFW(10,3)`
- `MUST BE WATCHED IF CHANGING DIMENSIONS` **Appears in RTBLK2X, SESDATX, S2OIRX, DATA LCMX...**
- **LEVEL 2, TBLS**

**COMMON /S2OIRX/ LCMX, NRS, LCFW(10,3)**
**COMMON /SESATX/ TBLS(11000)**
**COMMON /SESINX/ DUM(4), KBR, DUM1**
**COMMON /INTORDX/ KFN**

**COMMON BLOCKS FOR THE EDSMOD ROUTINES**

- `COMMON /EDSCZ/ LOUT`**
- `COMMON /EDSC1/ LU41, LU42, LU43, LU44, LU45`**
- `COMMON /EDSC2/ LF41, LF42, LF43, LF44, LF45`**
- `COMMON /EDSC3/ INIT, IRDIM, IR(60,3), KUP(60,3)`**
- `COMMON /EDSC4/ NTABLE, NTABLO, IFLP, LCNT`**
- `COMMON /EDSC5/ NMAT, LABMAT(60), IDMAT(60), IMATEL`**
- `COMMON /EDSC6/ NMCL, LABMCL(60), IDMCL(60)`**
- `COMMON /EDSC7/ NMAT0, LABMAT0(60), IDMAT0(60), IMATO`**

**EDSMOD COMMON BLOCKS USED BY THE MODIFIED SESAME ROUTINES**

- `COMMON /EDSCOE/ TFACE, RFACE, PFACE, EFAC, KREPE`**
- `COMMON /EDSCOO/ TFACO, RFACO, OFACO, KREPO`**

**DESIGNATE THE OUTPUT FILE FOR THE ERROR MESSAGES**

- `DATA LOUT/"OUTPUT"/`**
- `DATA LCMX/11000/. NRS/10/. LCNT/1/, LCFW/30+0/,`**
- `IR/180+0/, NTABLE/1/, NTABLO/0/, INIT/O/, IRDIM/60/, KUP/180*(-1)/`**

**DATA LCMX/11000/, NRS/10/, LCNT/1/, LCFW/30+0/,**
**IR/180+0/, NTABLE/1/, NTABLO/0/, INIT/O/, IRDIM/60/, KUP/180*(-1)/**

**DATA LU41/41/, LU42/42/, LU43/42/, LU44/44/, LU45/45/**
**DATA LF41/6HSES2CL/, LF42/6HSESAME/, LF43/6HSESAME/, LF44/6HMIXLIB/**
**DATA LF45/6HMIXDIR/**

**INITIALIZE THE CONTENTS OF THE EOS TABLE SESAME**

- `DATA NMAT/32/`
- `DATA LABMAT/`**
- **DATA IDMCL/40*(1H )/**
- **DATA IDMAT/0/27/**

**DATA NMCL/7111.3710,7410,2020,7081,**
**4100,3330,5263.2700,7390,**
**2510,8180,2140,2145,3200,7240,7370,**
**3290,5410,3100.3730,7170,7590,**
**4 7380,2448,4270,7151,7432,1540,7560,7520,**
**5 7150/**

**DATA IDMAT/7111,3710,7410,2020,7081,**
**4100,3330,5263.2700,7390,**
**2510,8180,2140,2145,3200,7240,7370,**
**3290,5410,3100.3730,7170,7590,**
**4 7380,2448,4270,7151,7432,1540,7560,7520,**
**5 7150/**

**INITIALIZE THE CONTENTS OF THE CLASSIFIED EOS TABLE SES2CL**

- `DATA NMCL/0/`
- `DATA LABMCL/40*(1H )/`
- `DATA IDMCL/40+0/`

**INITIALIZE THE CONTENTS OF THE OPACITY TABLE SESAME**

- `DATA NMAT/0/27/`
DATA LABNO/"ALUMINUM","ARGON","BERYLLIUM","BORON","CALCIUM"
  1 ,"CARBON","CHLORINE","CHROMIUM","DEUTERIUM","HELIUM"
  2 ,"IRON","LITHIUM","MAGNESIUM","NITROGEN"
  3 ,"OXYGEN","PBX-9502","PHOSPHORUS","POTASSIUM","SIO2"
  4 ,"SILICON","SODIUM","SS","STAINLESS","SULPHUR"
  5 ,"TITANIUM","TITANIUM N","WATER"/

DATA IDMATO/13710,15170,12020,12330,12030
  1 ,12180,15020,13070,15263,15760
  2 ,12140,12290,13080,15000
  3 ,15010,18200,13910,12460,17380
  4 ,13810,12448,14270,14270,14010
  5 ,12960,16000,17150/

IMATEL=NMAT
IMATOL=NMATO

INIT=1
RETURN
END
SUBROUTINE EOSCON(KUNIT,KREP,LMAT)

* PURPOSE -
  TO PROVIDE THE APPROPRIATE EOS SCALE FACTORS FOR THE
  SYSTEM OF UNITS CHOSEN BY KUNIT (SEE BELOW)

* INPUT VARIABLES -
  KUNIT = KIND OF UNITS
  0 (SESAME) R-G/CC,T-DEG.K,O-CM**2/G,P-GPA,E-MJ/KG
  1 (CGS) R-G/CC,T-DEG.K,O-CM**2/GM,P-MUBR,E-ERGS/GM
  2 (SIU) R-KG/M**3,T-DEG.K,P-E/J,KG,O-M-E/G
  3 (HYDROX) R-G/CC,T-DEG.K,P-MBR,E-MBR**CC/GM,O-CM**2/G
  4 (HYDROX) R-G/CC,T-KEV,P-MBR,E-MBR**CC/G,D-CM**2/G
  5 (SESAME) R-G/CC,T-KEV,O-CM**2/G,P-GPA,E-MJ/KG
  6 (LASNEX) R-G/CC,T-KEV,O-CM**2/GM,P-JRKS/CC,E-JRKS/CC

* LEGEND -
  R = DENSITY
  T = TEMPERATURE
  O = OPACITY
  P = PRESSURE
  E = INTERNAL ENERGY
  CC = CUBIC CENTIMETER
  CM = CENTIMETER
  DEG. K = DEGREES KELVIN
  EV = ELECTRON VOLT
  G = GRAM
  GPA = GIGA PASCALS
  M = METER
  MBR = MEGABAR
  MUBR = MICROBAR
  PA = PASCAL
  J = JOULES
  JRKS = JERKS
  KG = KILOGRAM
  KEV = KILO ELECTRON VOLTS
  P = PASCAL

* OUTPUT VARIABLES - IN THE COMMON BLOCKS EOSCCE AND EOSCCO
  TFACE = TEMPERATURE EOS SCALING FACTOR
  RFACE = DENSITY EOS SCALING FACTOR
  PFACE = PRESSURE EOS SCALING FACTOR
  EFACE = ENERGY EOS SCALING FACTOR
  TFACO = TEMPERATURE OPACITY SCALING FACTOR
  RFACO = DENSITY OPACITY SCALING FACTOR
  OFACO = OPACITY SCALING FACTOR

* LOCAL VARIABLES -
  NONE

* EXTERNALS AND COMMON BLOCKS -
  EOSMOD COMMON BLOCKS- EOSCCE,EOSCCO

* REMARKS - THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

* REFERENCE- J. M. HYMAN, M. M. KLEIN

46
**EDSMOD**: A SUBROUTINE PACKAGE FOR CALCULATING EQUATIONS-OF-STATE AND OPACITIES

**LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980**

**DATE**: MARCH 6, 1980

**COMMEN /EOSCCE/ TFACE, RFACE, PFACE, EFACE, KREPE**

**COMMON /EOSCCO/ TFACO, RFACO, OFACO, KREPO**

**KREPE=KREP**

**KREPO=KREP**

**DEFINE THE DEFAULT SESAME VALUES**

**THESE ARE THE UNITS THAT THE SESAME DATA FILES ARE WRITTEN IN**

**TFACE=1.**

**RFACE=1.**

**PFACE=1.**

**EFACE=1.**

**TFACO=4.0646423**

**RFACO=0.**

**OFACO=0.**

**RESET THE SCALE FACTORS THAT ARE DIFFERENT FROM THE DEFAULT**

**KP1=KUNIT+1**

**GO TO (45, 40, 30, 20, 15, 10, 25), KP1**

**LASNEX UNITS**

**TFACE=8.617346719E-8**

**TFACO=-3.**

**GO TO 50**

**SESAME OPACITY UNITS**

**TFACE=8.61703E-5**

**TFACO=0.0**

**GO TO 50**

**HYDROX OPACITY UNITS**

**TFACE=8.61703E-8**

**TFACO=-3.**

**GO TO 50**

**SESAME EOS UNITS**

**TFACE=1.0**

**TFACO=-1.0**

**GO TO 50**

**HYDROX EOS UNITS**

**TFACE=1.0**

**TFACO=-1.0**

**GO TO 50**

**STANDARD INTERNATIONAL UNITS (SIU)**

**TFACE=1.0**

**TFACO=-1.0**

**GO TO 50**

**CGS UNITS**
40  PFACE=1.E+10
41  EFACE=1.E+10
42  GO TO 50
43  C
44  c  SESAME EOS UNITS
45 45 DFACO=2.0
46 50 CONTINUE
47  C
48  C  RESCALE THE TABLES IN THE USER PRESCRIBED SCALE FACTORS
49  DSFAC=1.0
50  CALL EOSDSL(LMAT,DSFAC)
51  EFACE=EFACE*DSFAC
52  RFACE=RFACE/OSFAC
53  RFACO=RFACO/11SFAC
54  C
55 140  RETURN
56 141  END
SUBROUTINE EOSDSL(LMAT, DSFAC)

* PURPOSE-
  TO ALLOW A USER TO RESCALE THE MASS DENSITY IN
  THE EOS TABLES. THIS IS A USEFUL ROUTINE TO APPROXIMATE THE
  EDS AND OPACITIES OF DIFFERENT ISOTOPES AND ISOTOPIC MIXTURES
  OF THE MATERIALS IN THE SESAME LIBRARY

* AN ALTERNATE PURPOSE IS TO ALLOW A USER TO DEFINE NEW
  SCALE FACTORS FOR THE UNITS THE TABLE IS TO BE WRITTEN IN

* INPUT VARIABLE-
  LMAT = MATERIAL IN AN AIO FIELD, FOR EXAMPLE- LMAT = "HELIUM"
  THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
  THE MATERIAL BY SETTING LMAT TO THE SESAME
  NUMBER, FOR EXAMPLE- LMAT = "5760" FOR HELIUM

* OUTPUT VARIABLE-
  DSFAC = DENSITY SCALE FACTOR EQUAL TO THE RATIO OF THE
  ATOMIC MASSES OF THE MATERIALS. THAT IS,
  DSFAC=ATOMIC MASS DENSITY OF THE SESAME MATERIAL)/
  (ATOMIC MASS DENSITY OF THE DESIRED MATERIAL)

* FOR EXAMPLE- THE EOS OF A 60-40 MIXTURE OF DEUTERIUM-TRIDIUM
  CAN BE APPROXIMATED BY DEFINING DSFAC=2/(0.6*2+0.4*3)=0.833
  AND CALLING EOSMOD WITH LMAT="DEUTERIUM"

* INPUT-OUTPUT VARIABLES IN THE COMMON BLOCKS EOSCCE AND EOSCCO
  TFACE = TEMPERATURE EOS SCALING FACTOR
  RFACE = DENSITY EOS SCALING FACTOR
  PFACE = PRESSURE EOS SCALING FACTOR
  EFACE = ENERGY EOS SCALING FACTOR
  TFACO = TEMPERATURE OPACITY SCALING FACTOR
  RFACO = DENSITY OPACITY SCALING FACTOR
  OFACO = OPACITY SCALING FACTOR

* FOR FURTHER INFORMATION ON THESE FACTORS SEE SUBROUTINE EOSCON

* EXTERNALS AND COMMON BLOCKS-
  EOSMOD COMMON BLOCKS- EOSCCE, EOSCCO

* REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

* REFERENCE- J. M. HYMAN, M. M. KLEIN
  EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
  EQUATIONS-OF-STATE AND OPACITIES
  LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980

* DATE- MARCH 6, 1980

* DUMMY SUBROUTINE FOR THE PACKAGE.

* IF A USER SUPPLIES THIS ROUTINE THEN THE FACTORS CAN BE
BE RESET AS DESCRIBED IN THE MANUAL WHEN THE TABLES ARE
WRITTEN, OR A DENSITY SCALE FACTOR CAN BE INCLUDED
TO CHANGE THE DENSITY TABLES BY A CONSTANT FACTOR.
FOR EXAMPLE, FOR A 60-40 MIXTURE OF DEUTERIUM-TRITIUM
THE CODE COULD BE WRITTEN AS-
IF(LMAT.EQ."DEUTERIUM") DSFAC=0.833
RETURN
END
SUBROUTINE EOSEFD (LMAT, IO, IMATE)

* PURPOSE-
* TO LOCATE EOS MATERIAL DEFINED BY HOLLERITH NAME
* IN APPROPRIATE FILE AND ASSIGN IT A SESAME (OR
* PRIVATE ID) EOS NUMBER

* INPUT VARIABLES-
* LMAT = MATERIAL IN AN AIO FIELD, FOR EXAMPLE- LMAT = "HELIUM"
* THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
* THE MATERIAL BY SETTING LMAT TO THE SESAME
* NUMBER, FOR EXAMPLE- LMAT = "5760" FOR HELIUM

* OUTPUT VARIABLES-
* IO = SESAME OR PRIVATE ID NUMBER AS STORED IN LIBRARY
* SESAME (EOS LIBRARY)
* IMATE = LOCATION OF MATERIAL IN ARRAY LABMAT+ MIXDIR
* PROVIDED MATERIAL HAS BEEN LOCATED
* = 0 IF MATERIAL HAS NOT BEEN LOCATED BY ROUTINE
* UPPER BOUND ON IMATE IS 60 (DIM IR(,))

* LOCAL VARIABLES-
* ICFASE = 1 IF THE PUBLIC EOS FILES HAVE BEEN ASSIGNED
* ICFASCL = 1 IF THE CLASSIFIED EOS FILES HAVE BEEN ASSIGNED
* ICFASP = 1 IF THE PRIVATE EOS FILES HAVE BEEN ASSIGNED

* EXTERNALS AND COMMON BLOCKS-
* EOSMOD COMMON BLOCKS- EOSC1.2,3.5.6
* FTN ROUTINES- ENCODE, EOF
* LASL T-4 HYDROSES ROUTINE- EOSFAS

* REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

* REFERENCE- J. M. HYMAN, M. M. KLEIN
* EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
* EQUATIONS-OF-STATE AND OPACITIES
* LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980
* DATE- MARCH 6, 1980

COMMON /EOSCZ/ LOUT
COMMON /EOSC1/ LU41, LU42, LU43, LU44, LU45
COMMON /EOSC2/ LF41, LF42, LF43, LF44, LF45
COMMON /EOSC3/ INIT, IRDIM, IR(60,3), KUP(60,3)
COMMON /EOSC5/ NMAT, LABMAT(60), IOMAT(60), IMATEL
COMMON /EOSC6/ NMCL, LAMBCL(60), LOMCL(60)

DATA ICFASE/0/,ICFASCL/0/,ICFASP/0/,IDCNT/1/

CHECK IF THE MATERIAL IS IN THE STANDARD SESAME LIST
DO 10 IMATE=1,IMATEL
IF (LMAT.EQ.LABMAT(IMATE)) GO TO 40

CONTINUE
CHECK IF THE MATERIAL IS IN THE CLASSIFIED EOS SESAME LIST
DO 12 IMATE=1,NMCL
IF (LMAT.EQ.LABMCL(IMATE)) GO TO 45

CONTINUE

*** ASSIGN EoS PRIVATE FILES TO PROGRAM IF AVAILABLE

IF(LU45.EQ.4HNDNE) GO TO 31

IMATE=NMAT

IF(ICFASP.GT.0) GO TO 15

ICFASP=1

CALL EOSFAS(3)

CONTINUE

THE CURRENT PRIVATE LIST DIRECTORY (MIXDIR) IS ON UNIT LU45

CALL FEXIST(LF45,IFFLAG)

IF(IFFLAG.EQ.0) GO TO 30

IFFLAG = 0 FILE NOT IN LOCAL FILE SPACE

IFFLAG = 1 FILE LOCAL

REWIND LU45

READ (LU45,80) LABEL,ID

IMATE=IMATE+1

IF (LMAT.EQ.LABEL) GO TO 5Q

IF (EOF(LU45)) 30,20

30 CONTINUE

31 CONTINUE

THE CURRENT PRIVATE LIST DIRECTORY (MIXDIR) IS ON UNIT LU45

CALL FEXIST(LF45,IFFLAG)

IF(IFFLAG.EQ.0) GO TO 30

IFFLAG = 0 FILE NOT IN LOCAL FILE SPACE

IFFLAG = 1 FILE LOCAL

REWIND LU45

READ (LU45,80) LABEL,ID

IMATE=IMATE+1

IF (LMAT.EQ.LABEL) GO TO 5Q

IF (EOF(LU45)) 30,20

30 CONTINUE

31 CONTINUE

32 FORMAT(I4)

33 IMATE=IMATE+1

IMATE=IMATE

LABMAT(IMATE)=LMAT

IOMAT(IMATE)=IO

ASSUME THE MATERIAL IS IN THE STANDARD SESAME LIST

IF IT IS NOT, A NONFATAL ERROR WILL OCCUR AT A LATER STEP

GO TO 40

THE MATERIAL WAS NOT FOUND. PRINT AN ERROR MESSAGE

WRITE(LOUT,75) LMAT

GO TO 999

IO=IOMAT(IMATE)

*** ASSIGN EoS FILES TO PROGRAM

IF(ICFASP.GT.0) GO TO 44

ICFASP=1

CALL EOSFAS(1)

CONTINUE
125 GO TO 999
126 C 45 ID=IDMCL(IMATE)
127 C 46 IF(ICFASCL .GT. 0 ) GO TO 46
129 C *** ASSIGN CLASSIFIED EOS FILES TO PROGRAM
130 CALL EOSFAS(4)
131 C 46 CONTINUE
132 C 50 CONTINUE
133 C 60 CONTINUE
134 C 999 CONTINUE
135 C RETURN
136 C 75 FORMAT (" MATERIAL LMAT = ",A10," NOT FOUND")
137 C 80 FORMAT (A10,I3)
138 C 90 FORMAT (" IMATE EXCEEDS UPPER BOUND IN SUBROUTINE EOSEFD ",A10)
139 C END
SUBROUTINE EOSFAS(KTABLE)

* PURPOSE-
  * TO ASSIGN FILE NAMES TO THE EOS AND OPACITY DATA FILES

* INPUT VARIABLES-
  * KTABLE = 1 EOS TABLE
  * 2 OPACITY TABLE
  * 3 PRIVATE TABLES
  * 4 SES2CL TABLES

* OUTPUT VARIABLES-
  * NONE

* LOCAL VARIABLES-
  * INITE, INITECL, INITO AND INITP ARE SET TO 1 AFTER THE EOS, OPACITY
  * AND PRIVATE FILES HAVE BEEN INITIALIZED

* EXTERNALS AND COMMON BLOCKS-
  * EOSMOD COMMON BLOCKS- EOSC1, EOSC2
  * FTN SUBROUTINES- QASSIGN, ASSIGN

* REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

* REFERENCE- J. M. HYMAN, M. M. KLEIN
  * EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
  * EQUATIONS-OF-STATE AND OPACITIES
  * LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980

* DATE- MARCH 6, 1980

COMMON /EOSC1/ LOUT
COMMON /EOSC2/ LU41, LU42, LU43, LU44, LU45
COMMON /EOSC2/ LF41, LF42, LF43, LF44, LF45
DATA INITE/O/, INITO/O/, INITP/O/, INITECL/O/
GO TO (10,20,30,40), KTABLE

10 CONTINUE
IF(INITE.NE.0) GO TO 999
IF((INITO.NE.0).AND.(LF42.EQ.LF43)) GO TO 999
INITE=1
CALL QASSIGN (LU42,LF42,0,0)
GO TO 999

20 CONTINUE
IF(INITE.NE.0) GO TO 999
IF((INITE.NE.0).AND.(LF42.EQ.LF43)) GO TO 999
INITE=1
CALL QASSIGN (LU43,LF43,0,0)
GO TO 999

PRIVATE TABLES
30 CONTINUE
63 IF(INITP.NE.0) GO TO 999
64 INITP=1
65 CALL QASSIGN (LU44,LF44,0,0)
66 CALL ASSIGN (LU45,LF45,40008)
67 GO TO 999
68 C
69 C CLASSIFIED EOS TABLE ASSIGNMENT CODING
70 40 CONTINUE
71 IF(INITECL.NE.0) GO TO 999
72 INITECL=1
73 CALL QASSIGN (LU41,LF41,0,0)
74 GO TO 999
75 C
76 999 RETURN
77 END
SUBROUTINE EOSGET(LMAT, KUNIT, KREP, IMAT, IDT, IERR)

* PURPOSE-* LOAD THE SESAME EOS DATA FILES

* INPUT VARIABLES-*
  * LMAT = MATERIAL IN AN A1O FIELD, FOR EXAMPLE- LMAT = "HELIUM"
  * THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
  * THE MATERIAL BY SETTING LMAT TO THE SESAME
  * NUMBER, FOR EXAMPLE- LMAT = "5760" FOR HELIUM

  * KUNIT= KIND OF UNITS
    * 0 (SESAMEE) R-G/CC,T-DEG.K,P-GPA,E-MJ/KG
    * 1 (CGS) R-G/CC,T-DEG.K,P-MB, E-ERGS/GM
    * 2 (SIU) R-KG/M**3,T-DEG.K,P-J, E-CM**2/GM
    * 3 (HYDROX) R-G/CC,T-DEG,K,P-MBR,E-MBR*CC/GM, D-CM**2/G
    * 4 (HYDROXO) R-G/CC,T-KEV,P-MBR,E-MBR*CC/G, D-CM**2/G
    * 6 (LASNEX) R-G/CC,T-KEV,D-CM**2/GM,P-JRKS/CC, E-JRKS/CC

  * IMAT = INDICATES WHETHER TO LOAD THE DATA FILE IF
    * IT DOES NOT EXIST FOR LMAT OR GO DIRECTLY TO A
    * PREVIOUSLY LOADED'FILE.
    * 0 CHECK IF THE DATA FILE FOR LMAT HAS BEEN LOADED.
    * IF NOT, SEARCH FOR THE FILE, CONVERT IT TO THE PROPER UNITS
    * AND COPY IT INTO LCM .
    * >0 EQUAL TO THE LMAT TABLE NUMBER. THIS NUMBER WAS RETURNED
    * BY THE PACKAGE ON A PREVIOUS CALL TO SUBROUTINE ESOORE,
    * EDSIPT OR EDSIERT. THIS OPTION IS FASTER THAN IMAT = 0
    * BECAUSE IT SKIPS SEARCH IN THE DIRECTOR OF THE

* OUTPUT VARIABLES-*
  * IERR = 0 SUCCESSFULL
  * .NE.0 UNSUCCESSFUL

* LOCAL VARIABLES-*

* EXTERNALS AND COMMON BLOCKS-
  * EOSMOD COMMON BLOCKS- EOSC1,EOSC3

* REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

* REFERENCE- J. M. HYMAN, M. M. KLEIN
  * EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
  * EQUATIONS-OF-STATE AND OPCITIES
  * LOS ALAMOS SCIENTIFIC LABORATORY RPT..LA-8502-M,1980

* DATE- MARCH 6, 1980

* LOAD THE SESAME EOS TABLES IN THE INVERTED FORMAT

COMMON BLOCKS FOR THE SESAME EOS ROUTINES

COMMON /S20IRX/ LCMX, NRS, LCFW(10,3)
COMMON /SESOSNX/ TBLS(11000)
COMMON /SESOSNX/ OUM(4), KBR, OUM1
COMMON /INTRODSX/ KFN
COMMON BLOCKS FOR THE EDSMOO ROUTINES
COMMON /EDOCS/ LOUT
COMMON /EDOCS/ LU41, LU42, LU43, LU44, LU45
COMMON /EDOCS/ INIT, IRDIM, IR(60,3), KUP(60,3)
COMMON /EDOCS/ NTABLE, NTABLD, IFLP, LCNT
DIMENSION ZB(3)
IERR=0
CALL THE FILE ASSIGNMENT ROUTINE TO ASSIGN READ AND WRITE
UNIT NUMBERS TO THE INPUT DATA FILES
IF(IOT.NE.2) CALL EDSEFD (LMAT,ID,IMAT)
IF(IOT.EQ.2) CALL EDSSOFD (LMAT,ID,IMAT)
IF (IMAT.LE.0) IERR=-1
IF (IMAT.LE.0) GO TO 75
CHECK IF THE TABLES HAVE BEEN INITIALIZED
IF (IR(IMAT,IOT).GT.0) GO TO 70
CONVERT TABLES TO APPROPRIATE UNITS
CALL EDSSCON(KUNIT,KREP,LMAT)
MIXTST=ID/1000
GO TO (10,20,30),IOT
LOAD THE EOS TABLES IN THE INVERTED FORMAT (IOT=1)
IF(MIXTST.NE.O)CALL GETINVS (NTABLE,ID,TBLS,LCNT,LU42,IERR,ZB)
IF(MIXTST.EQ.O)CALL GETINVS (NTABLE,ID,TBLS,LCNT,LU44,IERR,ZB)
GO TO 40
LOAD THE OPACITY TABLES (IOT=2)
IF (MIXTST.NE.O) CALL GETRPOX (NTABLE,ID,TBLS,LCNT,LU43,IERR)
IF (MIXTST.EQ.O) CALL GETRPOX (NTABLE,ID,TBLS,LCNT,LU44,IERR)
GO TO 40
LOAD THE EOS TABLES IN THE STANDARD FORMAT (IOT=3)
IF(MIXTST.NE.O)CALL GETEDS (NTABLE,ID,TBLS,LCNT,LU42,IERR,ZB)
IF(MIXTST.EQ.O)CALL GETEDS (NTABLE,ID,TBLS,LCNT,LU44,IERR,ZB)
IERR RETURNS NEGATIVE IF THERE IS AN INITIALIZATION ERROR
IF (IERR.GT.0) GO TO 60
IF (IERR.LT.0) WRITE(LOUT,90) IERR
IF (IERR.EQ.0) WRITE(LOUT,100)LMAT
IERR=-2
GO TO 75
THE TABLES HAS BEEN LOADED, SET THE EDSMOO VARIABLES
IF(IOT.NE.2) GO TO 65
IERR=0
C OPACITY TABLE UPDATE
125 IR(IMAT,IDT)=NTABLE
126 KUP(IMAT,IDT)=10*KUNIT+KREP
127 NTABLE=NTABLE+1
128 GO TO 70
129 C EOS TABLE UPDATE
130 CONTINUE
131 IR(IMAT,IDT)=NTABLE
132 KUP(IMAT,IDT)=10*KUNIT+KREP
133 NTABLE=NTABLE+1
134 70 CONTINUE
135 C CHECK IF THE UNITS ARE VALID
136 IF(10*KUNIT+KREP.NE.KUP(IMAT,IDT)) IERR=-6
137 C IF(IERR.EQ.-6) WRITE(OUT,80)KUNIT,KREP,KUP(IMAT,3)
138 80 FORMAT(" THE UNITS HAVE CHANGED SINCE THE LAST USE OF THE TABLE"
139 2," THE CURRENT VALUES OF KUNIT AND KREP ARE ",2I4
140 3," THE PREVIOUS VALUES OF KUNIT AND KREP WERE ",15)
141 C 75 CONTINUE
142 IF(IERR.LT.0) IMAT=IERR
143 C RETURN
144 90 FORMAT(" INSUFFICIENT STORAGE IN LCM"
145 1," LACK ",I10," WORDS")
146 100 FORMAT(" UNABLE TO LOCATE MATERIAL ",A10/
147 1 " IN SUBROUTINE EOSGET")
148 END
SUBROUTINE EOSKUT(KPARM,KBR,KUNIT,KREP,KFN,KEOSS,KBRS,KFNS,IMAT,
   2 IOT,IERR)

******************************

PURPOSE-

TO CHECK THE VALIDITY OF THE INPUT VALUE FOR KPARM
AND TO SEPARATE OUT THE INTERNAL PARTS

INPUT VARIABLES-

KPARM = MULTIPLE PARAMETER FLAG TO DESCRIBE HOW TO WRITE
AND RETRIEVE THE DATA FILE. KPARM HAS FOUR DECIMAL DIGITS,

KPARM = 1000*KBR + 100*KUNIT + 10*KREP + KFN WHERE

OUTPUT VARIABLES-

KBR = COMPUTATIONAL MODE FLAG TO INDICATE WHICH
QUANTITIES AND THEIR PARTIAL DERIVATIVES ARE TO
BE CALCULATED AND RETURNED BY THE PACKAGE.

KUNIT = KIND OF UNITS
0 (SESAMEE) R-G/CC,T-DEG.K,D-CM**2/G,P-GPA,E-MJ/KG
1 (CGS) R-G/CC,T-DEG.K,P-MA/F,E-MJ/KG
2 (SIU) R-G/CC,T-DEG.K,P-MBR,E-MBR*CC/GM,D-CM**2/G
3 (HYDROX) R-G/CC,T-DEG.K,P-MBR,E-MBR*CC/GM,D-CM**2/G
4 (HYDROXD) R-G/CC,T-KEV,P-MBR,E-MBR*CC/G,D-CM**2/G
5 (SESAMED) R-G/CC,T-KEV,P-MBR*CC/GM,P-MBR/CC
6 (LASNEX) R-G/CC,T-KEV,P-MBR*CC/GM,P-JRKS/CC,ED-JRKS/CC

IF KREP REFERS TO EOS TABLE UNITS
KREP = COMPUTATIONAL FLAG TO INDICATE WHETHER E IS
IS TO BE REPRESENTED AS ENERGY PER UNIT MASS OR ENERGY
PER UNIT VOLUME
0 ENERGY IN UNITS OF ENERGY PER UNIT MASS. FOR EXAMPLE-
1 ENERGY IN UNITS OF ENERGY PER UNIT VOLUME. FOR EXAMPLE-
O ENERGY IN UNITS OF ERGS/GRAM WHEN KUNIT = 1. THIS IS THE USUAL E.
1 ENERGY IN UNITS OF ERGS/CM**3 WHEN KUNIT = 1. THIS IS THE ENERGY
DENSITY RHO*E COMMONLY COMPUTED IN HYDRODYNAMIC
COMPUTER CODES.

IF KREP (KREP0) REFERS TO OPACITY TABLE UNITS
KREP0 = COMPUTATIONAL FLAG TO INDICATE WHICH REPRESENTATION
TO USE FOR THE OPACITY VARIABLE.
0 OPACITY REPRESENTED AS KAPPA IN DIMENSIONAL
1 OPACITY REPRESENTED AS A MEAN-FREE PATH.
LAMDA = 1/(KAPPA*RHO), IN DIMENSIONAL
UNITS OF LENGTH.

KFN = KIND OF FUNCTION INTERPOLATION IN THE TABLES
0 RATIONAL APPROXIMATIONS (ACCURATE)
1 BILINEAR APPROXIMATIONS (FAST)

LOCAL VARIABLES-

EXTERNALS AND COMMON BLOCKS-

REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE

PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL

REFERENCE- J. M. HYMAN, M. M. KLEIN

59
COMMON /EDSCLZ/ LOUT
COMMON /EDSC3/ INIT, IRDIM, IR(60.3), KUP(60,3)

IERR=0
IF(INIT.EQ.O) CALL EOSBEG
UNSCRAMBLE MULTIPLE FLAG KPARM
KBR=KPARM/1000
ITEMP=KPARM-KBR*1000
KUNIT=ITEMP/100
ITEMP=ITEMP-KUNIT*100
KREP=ITEMP/10
KFN=ITEMP-10*KREP
KDESS=KPARM
KFNS=KFN
KBR5=KBR
CHECK IF KPARM IS A VALID INPUT PARAMATER
IF(KPARM.LT.0) IERR=-2
IF(KBR.GT.2) IERR=-2
IF(KUNIT.GT.6) IERR=-2
IF(KREP.GT.1) IERR=-2
IF(KFN.GT.1) IERR=-2
PRINT AN ERROR MESSAGE IF KUNIT IS NOT VALID
IF(IERR.LT.0) WRITE(LOUT,10)KPARM,KBR,KUNIT,KREP,KFN
10 FORMAT( "ERROR DETECTED IN KPARM VALUE IN SUBROUTINE EOSKUT"
          ,/.," KPARM=",KPARM,", KBR=",KBR,", KUNIT=",KUNIT,", KREP=",KREP,", KFN=",KFN)
CHECK IF THE UNITS HAVE CHANGED
IF(IERR.LE.0) WRITE(LOUT,10)KUNIT,KREP,KUP(IMAT,10)
10 FORMAT( "THE UNITS HAVE CHANGED SINCE THE LAST USE OF THE TABLE"
          ,/.," THE CURRENT VALUES OF KUNIT AND KREP ARE ",KUNIT,KREP)
WRITE(LOUT,30)KUNIT,KREP,KUP(IMAT,10)
30 FORMAT( "THE UNITS HAVE CHANGED SINCE THE LAST USE OF THE TABLE"
          ,/.," THE CURRENT VALUES OF KUNIT AND KREP WERE ",KUNIT,KREP)
SUBROUTINE EOSOFD (LMAT, IO, IMAT0)

********************************************************************

* PURPOSE-
* TO OBTAIN MATERIAL NUMBERS FOR OPACITY TABLES

* INPUT VARIABLES-
* LMAT = MATERIAL IN AN A1O FIELD, FOR EXAMPLE- LMAT = "HELIUM"
* THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY
* THE MATERIAL BY SETTING LMAT TO THE SESAME
* NUMBER, FOR EXAMPLE- LMAT = *5760* FOR HELIUM

* OUTPUT VARIABLES-
* ID= SESAME OR MIXTUREID NUMBER AS STORED
* IN SESAME OR MIXDIR(MIXLIB)
* IMAT0 = LOCATION OF MATERIAL IN ARRAY LABMD + MIXDIR
* PROVIDED MATERIAL HAS BEEN LOCATED
* = 0 IF MATERIAL HAS NOT BEEN LOCATED

* LOCAL VARIABLES-
* EOSMOD COMMON BLOCKS- EOSC1, EOSC3
* EXTERNALS AND COMMON BLOCKS-
* PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL
* REFERENCE- J. M. HYMAN, M. M. KLEIN
* EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
* EQUATIONS-OF-STATE AND OPACITIES
* LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980
* DATE- MARCH 6, 1980

*----------------------------------------------------------------------

COMMON /EOSC1/ LU41, LU42, LU43, LU44, LU45
COMMON /EOSC2/ LF41, LF42, LF43, LF44, LF45
COMMON /EOSC3/ INIT, IRDIM, IR(60,3), KUP(60,3)
COMMON /EOSC7/ NMAT0, LABMD(60), IDMAT0(60), IMATOL

ICFASO, ICFASP/2*/0/

***** ASSIGN OPACITY FILES TO PROGRAM
IF(ICFASO .GT. 0) GO TO 5
ICFASO=1
CALL EOSFAS(2)
5 CONTINUE

ICFASP=1
CALL EOSFAS(3)
15 CONTINUE

*** ASSIGN PRIVATE OPACITY FILES TO PROGRAM
IF(ICFASP .GT. 0) GO TO 15
ICFASP=1
CALL EOSFAS(3)
15 CONTINUE

THE CURRENT PRIVATE LIST DIRECTORY (MIXDIR) IS ON UNIT LU45
63 C CHECK IF LF45 EXISTS IN LOCAL FILE SPACE
64 CALL FEXIST(LF45, IFFLAG)
65 IF(IFFLAG .EQ. 0) GO TO 30
66 C IFFLAG = 0 FILE NOT IN LOCAL FILE SPACE
67 C IFFLAG = 1 FILE LOCAL
68 REWIND LU45
69 20 READ (LU45,80) LABEL,ID
70 IMAT=IMAT+1
71 IF (LMAT.EQ.LABEL) GO TO 50
72 IF (EED(LU45)) 30,20
73 30 CONTINUE
74 C
75 C CHECK IF THE LMAT IS A SESAME NUMBER
76 IMAT1=AND(SHIFT(LMAT,6),77B)
77 CRAY CODE IMAT1=AND(SHIFT(LMAT,8),377B)
78 C
79 IF(IMAT1.LT.20B) GO TO 35
80 IF(IMAT1.GT.31B) GO TO 35
81 C
82 OECODE(10,32,LMAT) ID
83 32 FORMAT(I5)
84 C
85 00 33 IMAT1=IMAT1+1
86 IF(ID.EQ.IDMATO(IMAT1)) GO TO 40
87 33 CONTINUE
88 C
89 IMATOL=IMATOL+1
90 IMAT=IMATOL
91 LABMD(IMAT)=LMAT
92 IDMATO(IMAT)=ID
93 C
94 C ASSUME THE MATERIAL IS IN THE STANDARD SESAME LIST
95 C IF IT IS NOT. A NONFATAL ERROR WILL OCCUR AT A LATER STEP
96 GO TO 40
97 C
98 C
99 C THE MATERIAL WAS NOT FOUND. PRINT AN ERROR MESSAGE
100 35 CONTINUE
101 IMAT=0
102 WRITE(LOUT,70) LMAT
103 GO TO 999
104 C
105 40 ID=IDMATO(IMAT)
106 50 CONTINUE
107 C
108 IF (IMAT.LE.IRDIM) GO TO 60
109 IMAT=0
110 WRITE(LOUT,90) LMAT
111 60 CONTINUE
112 C
113 999 CONTINUE
114 RETURN
115 C**************** IMPROVE THESE DIAGNOSTICS*******************
116 70 FORMAT (" MATERIAL LMAT = ",A10," NOT FOUND")
117 80 FORMAT (A10,13)
118 90 FORMAT (" IMAT EXCEEDS UPPER BOUND IN SUBROUTINE EOSOFO ",1./," FOR MATERIAL LMAT=",A10)
119 END
T-4 SUBROUTINES USED BY EOSMOD

DPACKX (packs real numbers) ........................................... 64
GETEOSX (loads the total EOS tables) .............................. 65
GETINVX (gets inverted EOS tables) ................................. 67
GETRPOX (loads Rosseland/Planck opacity tables) .............. 69
INBUFHX (sequential read) ........................................... 71
INV301X (inverts a 301 table) ...................................... 72
ISRCHXX (index search) ................................................ 74
MATCHXX (checks if material table is loaded) ................. 75
RATFNIX (one-dimensional interpolator) ......................... 76
TABRANX (fetches a given table for a given material from a Sesame II library) ........................................... 78
T4DATIX (interpolates the inverted tables) ....................... 80
T4DATX (search/interpolate for atomic data as functions of region, density, and temperature) ....................... 83
T4INTPX (interpolates for a function z(x,y) and its derivatives) ................................................................. 86
T4PTREX (computes R and E from inverted table) ............... 89
T4RTPEX (computes P and E from inverted table) ............... 91
FUNCTION OPACKX(A,B)

FUNCTION OPACKX

PURPOSE TO DOUBLE PACK ARGUMENTS A AND B INTO A SINGLE WORD

REMARKS SYSTEM DEPENDENT SHIFT FUNCTION

PROGRAMMER J.ABDALLAH,JR.

DATE 1 MAY 1979

EQUIVALENCE (I1,X1),(I2,X2)

DATA MASK/7777777770000000000B/

X1=A

X2=B

I1=I1.AND.MASK

I2=I2.AND.MASK

I2=SHIFT(I2,30)

I1=I1.OR.I2

DPACKX*=X1

RETURN

END
SUBROUTINE GETEOSX(IR,MIO,IOT,TBLS,LCNT,LU,IFL,ZB)

C*************************************************************
C SUBROUTINE
C PURPOSE
C TO LOAD THE TOTAL EDS TABLES
C ARGUMENTS
C IR (INPUT) REGION NO.
C MIO (INPUT) MATERIAL ID.
C IOT (INPUT) DATA TYPE INDICATOR
C TBLS (OUTPUT) ARRAY FOR TABLE STORAGE
C LCNT (1/0) POSITION IN ARRAY FOR STORING TABLES
C LU (INPUT) SESAME LIBRARY UNIT NUMBER
C IFL (OUTPUT) ERROR FLAG
C = 2 FOR MATERIAL ALREADY LOADED
C = 1 FOR SUCCESSFUL LOADING
C = 0 FOR DATA NOT FOUND
C = - NO. OF EXTRA WORDS NEEDED FOR STORAGE
C ZB (OUTPUT) ATOMIC CHARGE, CHARGE**2, AND MASS
C = Z
C = Z**2
C = A
C REMARKS
C THIS IS THE LASNEX VERSION OF GETEOS
C PRESSURES AND ENERGIES ARE DOUBLE PACKED
C ENERGY DENSITIES ARE PER UNIT VOLUME (NOT MASS)
C EXTERNALS MATCHKX, TABRANX, DPACKX
C PROGRAMMER R.C. ALBERS, T-4
C DATE 25 APRIL 79
C*************************************************************
LEVEL 2,TBLS
DIMENSION TBLS(1),ZB(3)
REPLACE FOLLOWING LINE BY USER COMMON BLOCKS
COMMON/S201RX/LCMX,NRS,LCFW(10,3)
COMMON/EDSCCE/TFACE,RFACE,PFACE,EFACE,KREPE
CHECK TO SEE IF TABLE HAS BEEN LOADED
CALL MATCHKX(MIO,NRS,LCFW(1,IOT),TBLS(1),IFLG)
IF(IFLG.EQ.0) GO TO 10
IFL=2
RETURN
10 NLEFT = LCMX - LCNT - 1
FETCH THE 201 TABLE
CALL TABRANX(MIO,201.,LU,TBLS(LCNT+2),NLEFT,IFL)
IF(IFL.LE.0) RETURN
ZB(1) = TBLS(LCNT+2)
ZB(2) = ZB(1)*ZB(1)
ZB(3) = TBLS(LCNT+3)
RHO0=TBLS(LCNT+4)
FETCH THE 301 TABLE
CALL TABRANX(MIO,301.,LU,TBLS(LCNT+2),NLEFT,IFL)
IF(IFL.LE.0) RETURN

CONVERT TO LASNEX UNITS AND DOUBLE PACK
NR = TBLS(LCNT+2)
NT = TBLS(LCNT+3)
DO 20 I=1,NT
LOCT = I + (LCNT + NR + 3)
TBLS(LOCT) = TFACE*TBLS(LOCT)
NWOS = NR*NT
DO 30 J=1,NR
RHO = TBLS(J+LCNT+3)*RFACE
TBLS(J+LCNT+3) = RHO
DO 40 I=1,NT
LOCP = (I-1)*NR + J + (NT + NR + LCNT + 3)
LOCE = LOCP + NWOS
PTEM = TBSL(LOCP)*PFACE
ETEM = TBSL(LOCE)*EFACE
IF(KREPE.EQ.1)ETEM = ETEM*RHO
TBLS(LOCP) = DPACKX(PTEM,ETEM)
40 CONTINUE
30 CONTINUE

RESET INPUT PARAMETERS AND END
TBLS(LCNT) = FLOAT(MIO)
TBLS(LCNT+1) = RHOO
LCFW(IR,IOT) = LCNT
LCNT = LCNT + 2 + IFL - NWOS
IFL = 1
RETURN
END
SUBROUTINE GETINVX(IR,MID,IDT,TBLS,LCNT,LU,IFL,ZB)

C --- - -- - -- - ------- ----- ---- -- --- - --- -- - - -- - -- -- --------- ----- - . - -- --- .-
C
C SUBROUTINE
C  PURPOSE
C        TO LOAD INVERTED (ENERGY BASED) SESAME II EOS TABLES
C  ARGUMENTS
C  IR        (INPUT)  REGION NO.
C  MID       (INPUT)  SESAME MATERIAL ID
C  IDT       (INPUT)  DATA TYPE INDICATOR
C  TBLS      (INPUT)  TABLE STORAGE ARRAY
C  LCNT      (IN/OUT) POSITION IN ARRAY FOR STORING TABLES
C  LU        (INPUT)  SESAME LIBRARY UNIT NO.
C  IFL       (OUTPUT) ERROR FLAG
C  2=MATERIAL ALREADY LOADED
C  1=SUCCESSFUL LOADING
C  0=DATA NOT FOUND
C  LT.O FOR - THE NO. OF EXTRA WORDS NEEDED FOR LOADING
C  ZB        (OUTPUT) ATOMIC CHARGE,CHARGE**2,AND MASS
C  ZB(1)=Z
C  ZB(2)=Z**2
C  ZB(3)=A
C  UNITS - ENERGY  MBAR*CC/GM
C  TEMP  DEGREES KELVIN
C  DENSITY  GRAMS/CC
C  PRESSURE  MBAR
C REMARKS
C THIS ROUTINE WAS ORIGINALLY NAMED GETINV BEFORE THE MODIFICATIONS WERE MADE SO IT WOULD INTERFACE WITH EOSMDO
C EXTERNALS MATCHKX,TABRANX,INV3O1X
C PROGRAMMER  J.ABDALLAH,JR.
C DATE  13 JUNE 1979

LEVEL 2,TBLS
COMMON/S2DIRX/LCMX,NRS,LCFW(10.3)
DIMENSION ZB(3),TBLS(1)

OBTAIN THE UNIT CONVERSION FACTORS FROM THE EOSMDO ROUTINES
COMMON/EOSCCE/TFACE.RFACE.PFACE.EFACE,KREPE

CALL MATCHKX(MID,NRS,LCFW(1,1DT),TBLS(3),IFL)
IF(IFL.EQ.0) GO TO 10
LCFW(IR,IDT)=IFL
IFL=2
RETURN

NL=LCMX-LCNT-1
FETCH EOS TABLES
CALL TABRANX(MID,201..LU,TBLS(LCNT+2),NL,IFL)
IF(IFL.LE.O) RETURN
ZB(1)=TBLS(LCNT+2)
ZB(2)=ZB(1)*ZB(1)
ZB(3)=TBLS(LCNT+3)
TBLS(LCNT+1)=TBLS(LCNT+4)
CALL TABRANX(MID,301..LU,TBLS(LCNT+2),NL,IFL)
IF(IFL.LE.0) RETURN
TBLS(LCNT)=FLOAT(MIO)
CALL PERTCB(IR,TBLS(LCNT),ZB(1),ZB(3))
NR=TBLS(LCNT+2)
NT=TBLS(LCNT+3)
NRT=NR+NT
LOCP=LCNT+3+NR+NT

CONVERT TO DESIRED UNITS
DO 30 I=1,NT
TBLS(3+I+LCNT+NR)=TFACE*TBLS(3+I+LCNT+NR)
DO 30 D=1,NR
IF(I.GT.1) GO TO 20
TBLS(3+D+LCNT)=TBLS(3+D+LCNT)*RFACE
RHO=TBLS(3+LI+LCNT)
20 LOCP=LOCP+I
TBLS(LOCP)=PFACE*TBLS(LOCP)
TBLS(LOCP+NRT)=EFACE*TBLS(LOCP+NRT)
IF (KREPE.EQ.1) TBLS(LOCP+NRT)=TBLS(LOCP+NRT)*RHO
CONTINUE
WINDOW TABLES HERE AND RESET VALUES OF NR NT AND NRT
IF(WINDOWING IS NEEDED
INVERT TABLES
NINV=LCNT+3+2*NRT+2*NR+4*NT
IF(NINV.LE.LCMX) GO TO 40
IFL=LCMX-NINV
RETURN
40 RO=TBLS(LCNT+1)
LOC=LCNT+2
CALL INV301X(TBLS,LOC,RO,LODS)
DD PACK DEPENDENT VARIABLES
LOC=LCNT+3+NR+NT+NR
DO 50 I=1,NRT
LOC=LOC+1
PTEM=TBLS(LOC)
TTTEM=TBLS(LOC+NRT)
TBLS(LOC)=OPACKX(PTEM,TTTEM)
CONTINUE
WRAP UP
LCFW(IR,IOT)=LCNT
LCNT=LCNT+2+LODS-NRT
IFL=1
RETURN
END
SUBROUTINE GETRPOX(IR,MID,IDT,TBLS,LCNT,LU,IFL)

PURPOSE
TO LOAD THE ROSSELAND/PLANCK OPACITY TABLE

ARGUMENTS
IR (INPUT) REGION NO.
MID (INPUT) SESAME MATERIAL ID
IDT (INPUT) DATA TYPE INDICATOR
TBLS (OUTPUT) ARRAY FOR TABLE STORAGE
LCNT (I/O) POSITION IN ARRAY FOR STORING TABLES
LU (INPUT) SESAME LIBRARY UNIT NO.
IFL (OUTPUT) ERROR FLAG

EXTERNALS
MATCHKX,TABRANX,DPACKX
COMMON/EOSCCO/ FROM THE EOSMOD PACKAGE

PROGRAMMER J.ABDALLAH, JR.

MODIFIED BY M. KLEIN, GROUP T-7, 11 DECEMBER 1979
DATE 24 APRIL 1979

LEVEL 2, TBLS
DIMENSION TBLS(1), COMMON/S2DIRX/LCMX,NRS,LCFW(10,3)

THE COMMON BLOCK EOSCCO PROVIDES THE UNIT CONVERSION FACTORS FROM THE EOSMOD PACKAGE
COMMON/EOSCCO/ TFACO,RFACO,OFCACO,KREPO

UNITS..TEMP..DEG.K,RHO IN G/CC,OPACITY IN CM**2/G
CHECK TO SEE IF TABLE HAS BEEN LOADED ALREADY
CALL MATCHKX(MID,NRS,LCFW(1,IOT),TBLS(1),IFL)
IF(IFL.EQ.0) GO TO 10
LCFW(IR,IDT)*IFL
IFL=2
RETURN
NLEFT=LCMX-LCNT-1
FETCH THE S02 TABLE
CALL TABRANX(MID,S02..LU,TBLS(LCNT+2),NLEFT,IFL)
IF(IFL.LE.0) RETURN
CONVERT TO DESIRED UNITS
LINES THROUGH STATEMENT 60 MAY BE DELETED IF NO CONVERSION IS REQUIRED
NR=TBLS(LCNT+2)
NT=TBLS(LCNT+3)
IPT = LCNT + 3 + NR + NT

DO 60 K = 1, NT
TBLS(LCNT + 3 + NR + K) = TBLS(LCNT + 3 + NR + K) + TFACO
DO 60 J = 1, NR
IF(K.GT.1) GO TO 50
TBLS(LCNT + 3 + J) = TBLS(LCNT + 3 + J) + RFACO
50 IPT = IPT + 1
RDP = TBLS(IPT)
POP = SHIFT(RDP, 30)
RDP = RDP + TBLS(LCNT + 3 + J) * KREPO + DFACO
POP = POP + TBLS(LCNT + 3 + J) * KREPO + DFACO
RDP = DPACKX(RDP, POP)
TBLS(IPT) = RDP
CONTINUE
TBLS(LCNT) = FLOAT(MIO)
TBLS(LCNT + 1) = FLOAT(IDT)
LCFW(IR, IDT) = LCNT
LCNT = LCNT + IFL + 2
IFL = 1
RETURN
END
SUBROUTINE INBUFRX(LU,Z,NW,IAO,IFLG)

C SUBROUTINE INBUFRX(LU,Z,NW,IAO,IFLG)

C PURPOSE RANDOM I/O READ

C ARGUMENTS
LU (INPUT) UNIT NO.
Z (OUTPUT) STORAGE AREA WHERE DATA IS RETURNED
NW (INPUT) NO. OF WORDS TO BE READ
IAO (INPUT) STARTING DISK ADDRESS OF DATA
IFLG (OUTPUT) 0=NORMAL

1=EOF ENCOUNTERED
-1=ERROR

C REMARKS NONE

C EXTERNALS RDISK

C PROGRAMMER J.ABOALLAH,JR.

C DATE 1 MAY 1979

C-------------------------------

CALL RDISK(LU,Z,NW,IAO)

IF(UNIT(LU)) 10,20,30
10 IFLG=1
RETURN
20 IFLG=0
RETURN
30 IFLG=-1
RETURN
END
SUBROUTINE INV301X(DSTR, LOC, RO, LDS)

C SUBROUTINE:
C PURPOSE:
C ARGUMENTS:
C REMARKS:
C EXTERNALS:
C PROGRAMMER:
C DATE:

LEVEL 2, DSTR
DIMENSION DSTR(1)
COMMON/INTORDX/ IFN
COMMON/RTBLK1X/LOCX, NR, LOCY, KY, JX, NT, INT, ET, Z(2)
INT=I
IFNS=IFN
IFN=O
NR = DSTR(LOC)
NT = DSTR(LOC+1)
LOCT = 2+NR+LOC
LCEC = LOCT+NT
LOCP = LCEC+NR
LOCE = LOCP+NR+NT
LOCN = LOCE+NR+NT
IMAX = 2+NR+NT
DO 1 I=1, IMAX
DSTR(LOCN-1) = DSTR(LOCN-I-NR)
DO 2 I=1, NR
JJ = LOCE+I-1
Q = 1.E-12*ABS(DSTR(JJ))
DSTR(LCEC+I-1) = DSTR(JJ)
DSTR(JJ) = O.
DO 2 J=2, NT
JJ = JJ+NR
DSTR(JJ) = DSTR(JJ)-DSTR(LCEC+I-1)
IF(DSTR(JJ)-DSTR(JJ-NR).LT.O) DSTR(JJ)=DSTR(JJ-NR)+O
CONTINUE
I = ISRCHKX(RO, DSTR(LOC+3), NR-2, 1, 0)+1
DO 3 J=1, NT
DSTR(LOCN+J-1) = DSTR(LOC+J-1)
DSTR(LOC+J-1) = DSTR(LOC+J-1+NR*(J-1))
DO 3 J=1, NT
DD 4 J=1, NT
ET = DSTR(LOC+J-1)
63 \quad JX = ISRCH\#XX(ET, DSTR(LOCX+NR), NT-2, NR, 0)+1
64 \quad LOCY = LOCX+I-1
65 \quad KY = NR
66 \quad CALL RATFN1X
67 \quad OSTR(LOCN+NT+J-1) = Z(1)
68 \quad LOCY = LOCN
69 \quad KY = 1
70 \quad CALL RATFN1X
71 \quad DSTR(LOCN+NT+J-1) = Z(1)
72 \quad DD S J=1, NT
73 \quad OSTR(LOC+I-1+NR*(J-1)) = DSTR(LOCN+NT+J-1)
74 \quad OSTR(LOC+NR*(J-1)) = DSTR(LOCN+NT+J-1)
75 \quad LDS = LOCN-LOC
76 \quad IFN=IFNS
77 \quad RETURN
78 \quad END
FUNCTION ISRCHKX(X,TBLS,N,K,NSFT)

C ---------------------------------- -----------------------------
C FUNCTION: ISRCHKX(X,TBLS,N,K,NSFT)
C PURPOSE: FIND INDEX OF X IN AN ARRAY TBLS. TABLE VALUES
C          NEED NOT BE CONTIGUOUS AND CAN BE IN EITHER
C          ASCENDING OR DESCENDING ORDER.
C
C ARGUMENTS: X  (INPUT) - VALUE TO BE LOCATED
C          TBLS (INPUT) - TABLE TO BE SEARCHED
C          N (INPUT) - NUMBER OF VALUES TO BE SEARCHED
C          K (INPUT) - SPACING BETWEEN VALUES IN TABLE
C          NSFT (INPUT) - NO. OF BITS THE TABLE VALUES ARE
C          TO BE SHIFTED
C
C REMARKS: TBLS CAN BE DECLARED LCM ON THE CCO 7600.
C
C EXTERNALS: SHIFT.
C
C PROGRAMMER: G. I. KERLEY, T-411, J. ABDALLAH, JR.
C
C DATE: 19 NOVEMBER 1976, REVISED 6 JULY 1979
C
C------------------------ '----------------------- ---------------
LEVEL 2 TBLS
DIMENSION TBLS(1)
ISRCMKX = 0
J = N+1
KI = 1-K
S1=TBlS(1)
S=SHIFT(S1,NSFT)
S=SBSLS(KI+K+N)
S=SHIFT(S,NSFT)
S=S-S1
1 IF(J-ISRCMKX,EQ.1) RETURN
J= J+ ISRCMKX
S1=TBlS(KI+K+J)
S1=SHIFT(S1,NSFT)
2 IF(S*(X-S1),LT.0.0) GO TO 2
ISRCMKX = J
GO TO 1
J = JP
GO TO 1
END
SUBROUTINE MATCHKX(MID,NRS,LOC,TBLS,IFLG)

C---------------------- ------------------------------- -----------------
C SUBROUTINE MATCHKX(MID,NRS,LOC,TBLS,IFLG)
C PURPOSE TO CHECK IF A MATERIAL HAS BEEN
C PREVIOUSLY LOADED
C ARGUMENTS MID (INPUT) SESAME MATERIAL ID
NRS (INPUT) NUMBER OF REGIONS
LOC (INPUT) ARRAY OF FIRST WORD LOCATIONS
IN TABLE STORAGE ARRAY FOR
FOR EACH REGION
TBLS (INPUT) TABLE STORAGE ARRAY
IFLG (OUTPUT) =0 MATERIAL NOT PREVIOUSLY LOADED
GT.0 LOCATION OF TABLE IF LOADED
ALREADY
REMARKS NONE
EXTERNALS NONE
PROGRAMMER J. ABDALLAH, JR.
DATE 26 APRIL 1979

LEVEL 2, TBLS
DIMENSION LOC(1), TBLS(1)
IFLG=0
GO 100 J=1,NRS
LC=LOC(J)
IF(LC.LE.0) GO TO 100
ITEST=TBLS(LC)
IF(MID.EQ.ITEST) GO TO 200
CONTINUE
RETURN
GO 200 IFLG=LC
RETURN
END
SUBROUTINE RATFNIX

Purpose: Interpolate for a function \( y(x) \) and its derivative from tables located in array TBLS.

The routine also requires common blocks, COMMON/RTBLK1X/LOCX,KX,LOCY,KY,I,N,IP,X,Y(2)

**Remarks:** Unless linear form is specified, routine uses rational function method with quadratic estimate of derivatives at the mesh points.

TBLS can be declared LCM on the CDC 7600.

**Externals:** None, but a search routine must be called first, to compute index I.

**Programmer:** G. I. Kerley, T-4.

**Date:** 18 July 1979

---

```fortran
SUBROUTINE RATFNIX

REAL LOCX, KX, LOCY, KY, I, N, IP, X(2)

EXTERNAL TBLS

DIMENSION YY(6)

COMMON/INTOROX/ IFN
COMMON/RTBLK1X/LOCX,KX,LOCY,KY,I,N,IP,X,Y(2)

IF(IFN.EQ.1) GO TO 6
IF(IP.EQ.0) GO TO 3

IX = LOCX+KX*(I-1)
IY = LOCY+KY*(I-1)
YY(3) = TBLS(IX)
YY(4) = TBLS(IX+KX)-YY(3)
YY(1) = TBLS(IY)
YY(2) = (TBLS(IY+KY)-YY(1))/YY(4)
IF(I.EQ.0) GO TO 1

SP = (TBLS(IY+KY)-YY(1))/TBLS(IX+KX)
YY(6) = (SP-YY(2))/(TBLS(IX+KX)-YY(3))
IF(I.GT.1) GO TO 3

GO TO 1
```

---

1 SUBROUTINE RATFNIX
2 -----------------------------------------------
3 C SUBROUTINE: RATFNIX
4 C PURPOSE: INTERPOLATE FOR A FUNCTION Y(X) AND ITS DERIVATIVE FROM TABLES LOCATED IN ARRAY TBLS.
5 C THE ROUTINE ALSO REQUIRES COMMON BLOCKS,
6 COMMON/RTBLK1X/LOCX,KX,LOCY,KY,I,N,IP,X,Y(2)
7 C LOCX = LOCATION OF X VECTOR
8 C KX = SPACING OF X VECTOR
9 C LOCY = LOCATION OF Y VECTOR
10 C KY = SPACING OF Y VECTOR
11 C I = INDEX OF X AND Y VECTORS
12 C N = LENGTH OF X AND Y VECTORS
13 C X (INPUT) - INDEPENDENT VARIABLE
14 C Y (OUTPUT) - VECTOR OF LENGTH 2, WHERE
15 C Y(1) = VALUE OF FUNCTION
16 C Y(2) = DERIVATIVE OF FUNCTION
17 C IP (INPUT) - BRANCH PARAMETER
18 C IP.EQ.0. USE INPUT COEFFICIENTS IN YY
19 C IP.NE.0. CALCULATE YY VECTOR FIRST
20 C COMMON/INTOROX/ IFN
21 C IFN (INPUT) - INTERPOLATION TYPE
22 C IFN.NE.1, RATIONAL FUNCTION
23 C IFN.EQ.1, LINEAR
24 C COMMON/SESOATX/TBLS
25 C TBLS (INPUT) - TABLE STORAGE ARRAY
27 C DATE: 18 JULY 1979
28 C----------------------------------------------
29 LEVEL 2,TBLS
30 DIMENSION YY(6)
31 COMMON/SESOATX/TBLS(10000)
32 COMMON/INTOROX/IFN
33 COMMON/RTBLK1X/LOCX,KX,LOCY,KY,I,N,IP,X,Y(2)
34 IF(IFN.EQ.1) GO TO 6
35 IF(IP.EQ.0) GO TO 3
36
37 C CALCULATE COEFFICIENTS FOR RATIONAL FUNCTION INTERPOLATION
38 IX = LOCX+KX*(I-1)
39 IY = LOCY+KY*(I-1)
40 YY(3) = TBLS(IX)
41 YY(4) = TBLS(IX+KX)-YY(3)
42 YY(1) = TBLS(IY)
43 YY(2) = (TBLS(IY+KY)-YY(1))/YY(4)
44 IF(I.EQ.0) GO TO 1
45
46 SP = (TBLS(IY+KY)-YY(1))/TBLS(IX+KX)
47 YY(6) = (SP-YY(2))/(TBLS(IX+KX)-YY(3))
48 IF(I.GT.1) GO TO 3
49```
IF(YY(2)*YY(4)-YY(6)) .LE. 0.) YY(6) = YY(2)/YY(4)
YY(5) = YY(6)
GO TO 2
DM = YY(3)-TBLS(IX-KX)
SM = (YY(1)-TBLS(IY-KY))/DM
YY(5) = (YY(2)-SM)/(YY(4)+DM)
IF(I.EQ.N-1) YY(6)=YY(5)
IF(I.GT.2) GO TO 2
IF(SM*(SM-DM*YY(5)) .LE. 0.) YY(5)=(YY(2)-SM-SM)/YY(4)
IF(YY(6).NE.0.) YY(5)=YY(5)/YY(6)
C EVALUATE RATIONAL FUNCTION FROM PRECALCULATED COEFFICIENTS
Q = X-YY(3)
R = YY(4)-Q
IF(R.NE.0.) GO TO 4
W = 1.
GO TO 5
Q = Q*(1.-YY(5)*Q/R)
Y(1) = YY(1)+Q*(YY(2)-R+F)
Y(2) = YY(2)+(Q-R)*F+YY(4)*W*(F-YY(6))
RETURN
C CALCULATE COEFFICIENTS FOR LINEAR INTERPOLATION
IF(IP.EQ.0) GO TO 7
IX = LOCX+KX*(I-1)
IY = LOCY+KY*(I-1)
YY(3) = TBLS(IX)
YY(1) = TBLS(IY)
YY(2) = (TBLS(IY+KY)-YY(1))/(TBLS(IX+KX)-YY(3))
C CALCULATE LINEAR ESTIMATE FROM PRECALCULATED COEFFICIENTS
Y(1) = YY(1)+YY(2)*(X-YY(3))
Y(2) = YY(2)
RETURN
END
SUBROUTINE TABRANX(MID,TID,LIB,A,LEN,IFLAG)

PURPOSE
TO FETCH A GIVEN TABLE FOR A GIVEN MATERIAL
FROM A SESAME II LIBRARY

ARGUMENTS
MID (INPUT) MATERIAL IO
TID (INPUT) TABLE NO.- IF O.O MATERIAL INDEX IS RETURNED
LIB (INPUT) LIBRARY FILE UNIT NO.
A (OUTPUT) ARRAY FOR TABLE STORAGE
LEN (INPUT) NO. OF WORDS IN A AVAILABLE
IFLAG (OUTPUT) =O IF TABLE COULD NOT BE LOCATED
GT. O-NO. OF ADDITIONAL WORDS NEEDED

REMARKS
A RANDOM I/O TECHNIQUE IS USED TO LOCATE AND LOAD
THE SPECIFIED TABLE FROM THE SESAME II LIBRARY.
THE MATERIAL INDEX AND ITS ADDRESS ARE TO SAVED
TO HASTEN THE FETCHING OF ANOTHER TABLE FOR THE SAME
MATERIAL AND LIBRARY FILE IN SUBSEQUENT CALLS TO
TABFCH.

EXTERNALS INBUFRX

PROGRAMMER J.ABDALLAH.JR.

DATE 24 APRIL 1979

LEVEL 2,A
DIMENSION A(1),HINDEX(50)
DATA HINDEX(1)/O.O/
DATA LIBLST/O/
IFLAG=O

FIND NO. MATERIALS ON LIBRARY
IF(LIB.NE.LIBLST) GO TO 50
IOLAST=HINDEX(1)
IF(IOLAST.NE.MIO) GO TO 50
IF(TID.NE.O.O) GO TO 230
NW=HINDEX(5)
NW=NW+NW+5
IF(LEN.LT.NW) GO TO 999
00 30 0=1.NW
A(d)=HINDEX(U)
CONTINUE
IFLAG=NW
RETURN
LIBLST=LIB
NW=NW+NW+5
IF(LEN.LT.NW) GO TO 999
CALL INBUFRX(LIB,A,NW,3,IER)
A(J)=HINDEX(J)
30 CONTINUE
IFLAG=NW
RETURN
50 LIBLST=LIB
51 NW=1
54 IF(LIB.LT.NW) GO TO 999
55 CALL INBUFRX(LIB,A,1,O,IER)
56 N=A(1)
57 NW=N+N+N
58 IF(LIB.LT.NW) GO TO 999
59 CALL INBUFRX(LIB,A,NW,3,IER)
60 FINOADDRESS OF MATERIAL FILE
61 GO 100 J=1,N
62 ITEST=A(J)
IF(ITEM.NE.MID) GO TO 100
NW=A(J+N)
IAO=A(J+N+N)
GO TO 200
CONTINUE
RETURN
GET MATERIAL INDEX
IF(LEN.LT.NW) GO TO 999
IAOX=IAO
CALL INBUFRX(LIB,A,NW,IAOX,IER)
DO 210 J=1,NW
HINDEX(J)=A(J)
CONTINUE
IF(TID.EQ.0) GO TO 500
DO 300 J=1,N
IF(TID.NE.HINDEX(5+J)) GO TO 300
IAO=HINDEX(5+J+N)
IAO=IAOX+IAO
GO TO 400
CONTINUE
READ REQUESTED TABLE
CALL INBUFRX(LIB,A,NW,IAO,IER)
IFLAG=NW
RETURN
IFLAG=LEN-NW
RETURN
END
SUBROUTINE T4DATIX

PURPOSE: SEARCH/INTERPOLATE FOR PRESSURE AND TEMPERATURE AS FUNCTIONS OF REGION, DENSITY AND ENERGY, USING PACKED SESAME 2 DATA STRING OF TYPE 302

COMMON/SESINX/IR, IOT, R, E, IBR, IFL
COMMON/SESOUTX/P(3), T(3)
IR (INPUT) - MATERIAL REGION NUMBER
IOT (INPUT) - DATA TYPE INDICATOR
R (INPUT) - DENSITY
E (INPUT) - INTERNAL ENERGY
P, T (OUTPUT) - PRESSURE, TEMPERATURE VECTORS
P(1), T(1) = PRESSURE AND TEMPERATURE
P(2), T(2) = DENSITY DERIVATIVES
P(3), T(3) = ENERGY DERIVATIVES
IBR (INPUT) - 0=COMPUTE BOTH P AND T
1=COMPUTE P ONLY
2=COMPUTE T ONLY
COMMON/SESDATX/TBLS
TBLS (INPUT) - TABLE STORAGE ARRAY
COMMON/s201Rx/LCMX, NRS, LCFW(10,3)
COMMON/RTBLKIX/LOCX, IX, N, ISAME, RX1, PX1(2)
COMMON/RTBLK2X/LOCY, IY, NY, LOCZ, NZ, NSFT
/ RX2, ET, PX2(3), INT, IOS, ZZ(96)
COMMON/SESINX/IR, IOT, R, E, IBR, IFL
COMMON/SESOUTX/P(3), T(3)
COMMON/SESDATX/TBLS(10000)
DATA LOCLST, IP, IT/0, 1, 1/
LOC IS POINTER TO START OF DATA STRING FOR REGION IR
LOC = LCFW(IR, IOT)+2
THE FOLLOWING LINES OF CODE (THRU NZ+1) CAN BE
MOVED AFTER THE IF(LOC.EQ.LOCLST) GO TO 5
STATEMENT TO MAKE THE SUBROUTINE QUICKER FOR CODES WHICH
DO NOT ALSO USE TEMPERATURE BASED EOS TABLES.
NX = TBLS(LOC)
NY = TBLS(LOC+1)
N = NX
LOCR = LDC+2


DATE: 2 AUGUST 1978
63  KX = 1
64  LOCX = LOCR
65  LOCY = LOCX+NX
66  LOCE = LOCY+NY
67  KY = 1
68  LOCZ = LOCE+NX
69  NZ = 1
70  C TEST TO SEE IF THE MATERIAL IS THE SAME AS LAST CALL
71  IF(LOC.EQ.LOCST) GO TO 5
72  C THE FOLLOWING OPERATIONS DO NOT NEED TO BE REPEATED
73  C UNLESS A NEW REGION HAS BEEN ENTERED
74  LOCST=LOC
75  IXLAST = 0
76  IYLAST = 0
77  LOCI = LOCX+NX/2-1
78  LOCJ = LOCY+NY/2-1
79  LOCNX=LOCX+NX-2
80  LOCNY=LOCY+NY-2
81  C SEARCH FOR DENSITY INDEX
82   5 IF(R.LT.TBLS(LOCI)) GO TO 15
83   10 IF(R.LT.TBLS(LOCI+1)) GO. TO 20
84   15 IF(LOCI.EQ.LOCNX) GO TO 20
85   20 IX=LOCI-LOCX+1
86   25 C INTERPOLATE FOR ENERGY ON COLD CURVE. IF ISAME = 0,
87       O. DENSITY
88   30 IF(ISAME = 0, O. DENSITY AND TEMPERATURE INOICES ARE
89   40 IY=LOCJ-LOCY+1
90   45 C THE SAME AS IN THE LAST CALL TO THIS ROUTINE
91   50 ISAME = ISAME+IABS(IY-IYLAST)
92   55 IP = MINO(1,IP+ISAME)
93   60 IT = MINO(1,IT+ISAME)
94   65 IXLAST = IX
95   70 IYLAST = IY
96   75 IDS=(IOT-1)*32+1
97   80 IF(IBR.EQ.2) GO TO 50
98   85 C PRESSURE CALCULATION
99   90 NSFT = 0
100  INT=IP
101  CALL T4INTPX
102  P(1)=PX2(1)
103  P(2)=PX2(2)-DECOR*PX2(3)
125  P(3)=PX2(3)
126  IP = 0
127  IF(IBR.EQ.1) RETURN
128  C TEMPERATURE CALCULATION
129  50  NSFT = 30
130  INT=IT
131  IDS=IDS+16
132  CALL T4INTPX
133  T(1)=PX2(1)
134  T(2)=PX2(2)-DECOR*PX2(3)
135  T(3)=PX2(3)
136  IT = 0
137  RETURN
138  END
SUBROUTINE T40ATX

* PURPOSE: SEARCH/INTERPOLATE FOR ATOMIC DATA AS
FUNCTIONS OF REGION, DENSITY AND TEMPERATURE,
USING PACKED SESAME 2 DATA STRING

COMMON/SESINX/IR,IDT,AR,AT,IBR,IFL
IR (INPUT) - MATERIAL REGION NUMBER
IDT (INPUT) - DATA TYPE INDICATOR
AR (INPUT) - DENSITY
AT (INPUT) - TEMPERATURE
IBR (INPUT) - SPECIFIES VARIABLES REQUIRED
IBR = 0, BOTH VARIABLES
IBR = 1, FIRST HALF VARIABLE ONLY
IBR = 2, SECOND HALF VARIABLE ONLY
IFL (NOT USED)

COMMON/SESOUTX/P(3),E(3)
P,E (OUTPUT) - VARIABLES OF FIRST AND SECOND HALF
OF PACKED DATA STRING
P(1),E(1) = VALUE OF THE VARIABLES
P(2),E(2) = DENSITY DERIVATIVES
P(3),E(3) = TEMPERATURE DERIVATIVES

REMARKS: ADAPTED FROM T-4 SESAME 2 ROUTINES S2E0S AND
LA301A. TABLES OF 2 VARIABLES ARE DOUBLE PACKED.
THE SEARCH INDICES AND INTERPOLATION CONSTANTS
ARE SAVED AND REUSED, IF POSSIBLE.

******* SYSTEM DEPENDENT FEATURE. THE CONSTANT NSFT
******* IN STATEMENT 60 SHOULD BE SET TO 1/2 THE BIT
LENGTH. FOR A CDC 7600, NSFT = 30.

EXTERNALS: T4INTPX (RATIONAL FUNCTION AND BI LINEAR INTERPOLATION)
INTERPOLATION COEFFICIENTS FROM A PREVIOUS
CALL TO THE ROUTINE CAN BE REUSED.

PROGRAMMER: G. I. KERLEY, T-4., J. ABOALLAH, T-4

DATE: 11 JULY 1978, REVISED 27 APRIL 1979

LEVEL 2, TBLs
DIMENSION LOCLS(3), IXLAST(3), IYLAST(3),
IPLAST(3), IELAST(3),
COMMON/S2DIRX/LCMX, NRS, LCFW(10,3)
COMMON/RTBLK2X/LDCX, IX, NX, LOCY, IY, NY, LOCZ, NZ, NSFT,
R.T,Z(3), INT, IDS.ZZ(96)
COMMON/SES DatX/TBLS(10000)
COMMON/SESINX/IR,IDT,AR,AT,IBR,IFL
COMMON/SESOUTX/P(3),E(3)
DATA IPLAST/3*0/
DATA IELAST/3*1/
DATA LOCLS/3*0/
LOC IS POINTER TO START OF DATA STRING FOR REGION IR
LOC = LCFW(IR,IDT)+2
NZ = 1
NX*TBLS(LOC)
NY*TBLS(LOC+1)
LDCX = LOC+2
LOCY = LOCX + NX
LOCZ = LOCY + NY
LOCNX = LOCX + NX - 2
LOCNY = LOCY + NY - 2

IF (LOC.EQ. LOCLST(IOT)) GO TO 2
LOCLST(IOT) = LOC
IX = NX / 2
IY = NY / 2
IXLAST(IOT) = 0
IYLAST(IOT) = 0
GO TO 3

2 IX = IXLAST(IOT)
IY = IYLAST(IOT)
GO TO 3

R = AR
T = AT

C SEARCH FOR DENSITY INDEX
LOCI = LOCX + IX - 1
IF (R.LT. TBL(S(LOCI))) GO TO 15
LOCI = LOCI + 1
GO TO 10

15 IF (LOCI.EQ. LOCX) GO TO 20
LOCI = LOCI - 1
IF (R.LT. TBL(S(LOCI))) GO TO 15
IX = LOCI - LOCX + 1
C SEARCH FOR TEMPERATURE INDEX
LOCI = LOCY + IY - 1
IF (T.LT. TBL(S(LOCI))) GO TO 35
LOCI = LOCI + 1
GO TO 30

35 IF (LOCI.EQ. LOCY) GO TO 40
LOCI = LOCI - 1
IF (T.LT. TBL(S(LOCI))) GO TO 35
IY = LOCI - LOCY + 1

C IF ISAME = 0, DENSITY AND TEMPERATURE INDICES ARE THE SAME AS IN THE LAST CALL TO THIS ROUTINE
ISAME = IABS(IX - IXLAST(IOT)) + IABS(IY - IYLAST(IOT))
IXLAST(IOT) = IX
IYLAST(IOT) = IY
IOS = (IOT - 1) * 32 + 1
IPLAST(IOT) = MINO( I, IPLAST(IOT) + ISAME)
IELAST(IOT) = MINO( I, IELAST(IOT) + ISAME)
IF (IBR.EQ. 2) GO TO 50

50 INT = IELAST(IOT)

C IF ISAME = 0, DENSITY AND TEMPERATURE INDICES ARE THE SAME AS IN THE LAST CALL TO THIS ROUTINE

CALL T4INTPX
P(1)Z(1)
P(2)Z(2)
P(3)Z(3)
IPLAST(IOT) = 0
IF (IBR.EQ. 1) RETURN

50 INT = IELAST(IOT)

IPLAST(IOT) = 0

CALL T4INTPX
E(1)Z(1)
E(2)Z(2)
E(3)Z(3)
IELAST(IOT) = 0
RETURN
END
SUBROUTINE T4INTPX

PURPOSE: INTERPOLATE FOR A FUNCTION Z(X,Y) AND ITS DERIVATIVES FROM TABLES LOCATED IN ARRAY TBLS.

THE ROUTINE REQUIRES COMMON BLOCKS, COMMON/RTBLK2X/LOCX,IX,NX,LOCY,IY,NY,LOCZ,NZ,NSFT,

LOCX = LOCATION OF X VECTOR
IX = INDEX OF X VECTOR
NX = LENGTH OF X VECTOR
LOCY = LOCATION OF Y VECTOR
IY = INDEX OF Y VECTOR
NY = LENGTH OF Y VECTOR
LOCZ = LOCATION OF Z(X,Y) ARRAY
NZ = SPACING OF Z ARRAY
NSFT = BIT SHIFT PARAMETER

X,Y (INPUT) - INDEPENDENT VARIABLES
Z (OUTPUT) - VECTOR OF LENGTH 3, WHERE
Z(1) = VALUE OF FUNCTION
Z(2) = X DERIVATIVE OF FUNCTION
Z(3) = Y DERIVATIVE OF FUNCTION

ZZ (IN/OUT) - COEFFICIENT VECTOR OF LENGTH 16
IP (INPUT) - BRANCH PARAMETER
IP.EQ.0, USE INPUT COEFFICIENTS IN ZZ
IP.NE.0, CALCULATE ZZ VECTOR FIRST
IDS (INPUT) - DISPLACEMENT INTO ZZ FOR COEFFS.
TO BE USED

COMMON/INTORDX/IFN.
IFN (INPUT) - INTERPOLATION TYPE
IFN.NE.1, RATIONAL FUNCTION
IFN.EQ.1, BILINEAR
COMMON/SESOATX/TBLS
TBLS IS THE TABLE STORAGE ARRAY

REMARKS: UNLESS BILINEAR FORM IS SPECIFIED, ROUTINE USES RATIONAL FUNCTION METHOD WITH QUADRATIC ESTIMATE OF DERIVATIVES AT THE MESH POINTS.
TBLS CAN BE DECLARED LCM ON THE CDC 7600.

SYSTEM DEPENDENT FEATURE. THE Z-ARRAY CAN BE DOUBLE PACKED. PARAMETER NSFT SPECIFIES THE NUMBER OF BITS TO BE SHIFTED WHEN UNPACKING THE RIGHT HALF OF THE WORD. THIS ROUTINE USES THE LAST SHIFT FUNCTION

EXTERNALS: NONE, BUT A SEARCH ROUTINE MUST BE CALLED FIRST, TO COMPUTE INDECES IX AND IY.


DATE: 01 AUG 1979

COMMON/INTORDX/IFN

LEVEL 2,TBLS
COMMON/RTBLK2X/LOCX,IX,NX,LOCY,IY,NY,LOCZ,NZ,NSFT,X,Y,Z(3),
$ IP,IDS,ZZ(96) *
COMMON/INTORDX/IFN

86
63 COMMON/TBL5(10000)
64 C CALCULATE COEFFICIENTS FOR RATIONAL FUNCTION INTERPOLATION
65 IF(IFN.EQ.1) GO TO 13
66 IF(IP.EQ.0) GO TO 8
67 1 I = LOCX+IX-1
68 2 IZ = LOCZ+NZ*(IX-1+NX*(IY-1))
69 KZ = NZ
70 IBR = IX
71 3 NBR = NX-IX
72 ZZ(IDS+4) = TBL5(I)
73 DO 7 K=1,4
74 4 KI=IDS+K-1
75 IF(K.LT.4) GO TO 1
76 5 IZ = IZ+NZ
77 GO TO 4
78 1 IF(K.LT.3) GO TO 2
79 ZZ(IDS+6) = 0
80 I = LOCY+IY-1
81 6 KZ = KZ*NX
82 IZ = IZ-KZ
83 IBR = IY
84 NBR = NY-IY
85 ZZ(IDS+5) = TBL5(I)
86 GO TO 3
87 2 IF(K.LT.2) GO TO 3
88 IZ = IZ+NX+NZ
89 GO TO 4
90 3 7 O = TBL5(I+1)-TBL5(I)
91 4 ZZ(KI)+SHIFT(TBL5(IZ),NSFT)
92 8 S=SHIFT(TBL5(IZ+KZ),NSFT)
93 9 S = (S-ZZ(KI))/O
94 IF(NBR.EQ.1) GO TO 5
95 SP=SHIFT(TBL5(IZ+KZ),NSFT)
96 10 SP = (SP-O*S-ZZ(KI))/(TBL5(I+2)-TBL5(I+1))
97 G2 = (SP-O)/(TBL5(I+2)-TBL5(I))
98 IF(IBR.GT.1) GO TO 5
99 IF(S*(S-D*G2).LE.0.) G2=S/D
100 11 G1 = G2
101 GO TO 6
102 5 12 DM = TBL5(I)-TBL5(I-1)
103 13 SM=SHIFT(TBL5(IZ-KZ),NSFT)
104 14 SM = (ZZ(KI)-SM)/DM
105 G1 = (S-SM)/(O+DM)
106 IF(NBR.EQ.1) G2=G1
107 IF(IBR.GT.2) GO TO 6
108 IF(SM*(SM-DM*G1).LE.0.) G1=(S-SM-SM)/D
109 15 G1 = G1/G2
110 ZZ(KI+8) = G1
111 7 ZZ(KI+12) = G2
112 ZZ(IDS+7)=O
113 ZZ8=ZZ(IDS+7)
114 ZZ7=ZZ(IDS+6)
115 ZZ(IDS+2)*(ZZ(IDS+1)-ZZ(IDS))/ZZ8
116 ZZ(IDS+1) = (ZZ(IDS+3)-ZZ(IDS))/ZZ7
117 ZZ(IDS+3) = (S-ZZ(IDS+2))/ZZ7
118 ZZ(IDS+12)=ZZ(IDS+12)/ZZ8
119 ZZ(IDS+13)=ZZ(IDS+13)/ZZ8
120 ZZ(IDS+14)=ZZ(IDS+14)/ZZ7
121 ZZ(IDS+15)=ZZ(IDS+15)/ZZ7
122 C EVALUATE RATIONAL FUNCTION FROM PRECALCULATED COEFFICIENTS
123 8 QX = X-ZZ(IDS+4)
124 RX = ZZ(IDS+6)-QX

87
125 \[ QY = Y - ZZ(IDS+5) \]
126 \[ RY = ZZ(IDS+7) - QY \]
127 IF(RX .NE. 0.) GO TO 9
128 W1 = 1.
129 W2 = 1.
130 GO TO 10
131 9 W1 = 1. - 1. / (1. + ABS(ZZ(IDS+8)*OX/RX))
132 W2 = 1. - 1. / (1. + ABS(ZZ(IDS+9)*OX/RX))
133 F1 = ZZ(IDS+12)*(W1+ZZ(IDS+8)*(1.-W1))
134 F2 = ZZ(IDS+13)*(W2+ZZ(IDS+9)*(1.-W2))
135 Z(2) = ZZ(IDS+6)*((RY*(F1-ZZ(IDS+12))*W1+QY*(F2-ZZ(IDS+13))*W2)
136 G1 = RY*F1+QY*F2
137 IF(RY .NE. 0.) GO TO 11
138 W1 = 1.
139 W2 = 1.
140 GO TO 12
141 11 W1 = 1. - 1. / (1. + ABS(ZZ(IDS+10)*QY/RY))
142 W2 = 1. - 1. / (1. + ABS(ZZ(IDS+11)*QY/RY))
143 F3 = ZZ(IDS+14)*(W1+ZZ(IDS+10)*(1.-W1))
144 F4 = ZZ(IDS+15)*(W2+ZZ(IDS+11)*(1.-W2))
145 Z(3) = ZZ(IDS+7)*((RX*(F3-ZZ(IDS+14))*W1+QX*(F4-ZZ(IDS+15))*W2)
146 G2 = RX*F3+QX*F4
147 Z2*ZZ(IDS+1)
148 ZZ3*ZZ(IDS+2)
149 ZZ4*ZZ(IDS+3)
150 Z(1) = ZZ(IDS)+(ZZ2+ZZ4*QY-RX*G1)*OX+(2Z3-RY*G2)*QY
151 Z(2) = Z(2)+ZZ2+QY*(ZZ4+RY*(F3-F4))+(OX-RX)*G1
152 Z(3) = Z(3)+ZZ3+OX*(ZZ4+RX*(F1-F2))+(QY-RY)*G2
153 RETURN
154 C CALCULATE COEFFICIENTS FOR BILINEAR INTERPOLATION
155 13 IF(IP .EQ. 0) GO TO 14
156 I=LOCX+IX
157 IND=IDS+4
158 ZZ(IND)=TBL5(I-1)
159 DX=TBL5(I)-ZZ(IND)
160 J=LOCY+IY
161 IND=IDS+5
162 ZZ(IND)=TBL5(J-1)
163 OY=TBL5(J)-ZZ(IND)
164 IZ=LOCZ+NZ*(IX-1+NX*(IY-1))
165 ZZ(IDS)=SHIFT(TBL5(IZ),NSFT)
166 IND=IDS+6
167 ZZ(IND)=SHIFT(TBL5(IZ+NZ),NSFT)
168 ZZ(IND)=(ZZ(IND)-ZZ(IDS))/OX
169 IZ=1Z+NZ*NX
170 IND=IDS+2
171 ZZ(IND)=SHIFT(TBL5(IZ),NSFT)
172 ZZ(IND)=(ZZ(IND)-ZZ(IDS))/DY
173 IND=IDS+3
174 ZZ(IND)=SHIFT(TBL5(IZ+NZ),NSFT)
175 ZZ(IND)=(ZZ(IND)-ZZ(IDS)-ZZ(IDS+1)+QX-ZZ(IDS+2)*QY)/(QX*QY)
176 C EVALUATE BILINEAR FUNCTION FROM PRECALCULATED COEFFICIENTS
177 14 QX = X-ZZ(IDS+4)
178 QY = Y-ZZ(IDS+5)
179 Z(2) = ZZ(IDS+4)*QY
180 Z(3) = ZZ(IDS+5)*QX
181 Z(1) = ZZ(IDS)+Z(2)*QX+ZZ(IDS+2)*QY
182 RETURN
183 END
SUBROUTINE T4PTREX(IR, IDT, TBLS, P, T, R, E, IFL)

C---------------------------------------------------------------
C SUBROUTINE: T4PTREX(IR, IDT, TBLS, P, T, R, E, IFL)
C PURPOSE:
C
C ARGUMENTS:
C IR (INPUT) - REGION NO.
C IDT (INPUT) - DATA TYPE CORRESPONDING TO E BASED EOS
C TBLS (INPUT) - TABLE STORAGE ARRAY
C P (INPUT) - PRESSURE
C T (INPUT) - TEMPERATURE
C R (OUTPUT) - DENSITY
C E (OUTPUT) - ENERGY
C IFL (OUTPUT) - ERROR FLAG
C
C REMARKS:
C TBLS CAN BE DECLARED LCM ON THE CDC 7600.
C EXTERNALS: T40ATIX, ISRCHKX.
C DATE: 3 MARCH 1978
C---------------------------------------------------------------

LEVEL 2, TBLS
DIMENSION TBLS(1)
COMMON/S2DIRX/LCMX, NRS, LCFW(10,3)
COMMON/SESINX/IRX, IDTX, RX, EX, IBR, IFLX
COMMON/SESOUTX/ZP(3), ZT(3)
IBR=0
IFLX=I
IRX=IR
IOTX=IOT
LOC=LCFW(IR, IDT)+1
NR = TBLS(LOC+1)
NT = TBLS(LOC+2)
ILO = 1
IHI = NR
I = .5*(ILO+IHI)
LOC = LCFW(IR, IDT)+1
J = ISRCHKX(T, TBLS(LOC)+1), NT-2, NR, 30)+1
LOC = LOC+1+NR+NT+I+NR*(J)
IF(IHI-I.EQ.1) GO TO 3
IF(P.LT.TBLS(LOC)+1) GO TO 2
ILO = I
GO TO 1
IHI = I
GO TO 3
RX = TBLS(LOC+2+I)
EX = TBLS(LOC+2+NR+J)+TBLS(LOC+2+NR+NT+I)
K = 0
IFL = 1
IF(K.EQ.50) GO TO 6
CALL T40ATIX
R = RX
E = EX
PTEST = ABS(P-ZP(1)) - 1.E-05*(ABS(P)+1.E-05)
IF(PTEST.GT.0) GO TO 5
TTEST = ABS(T-ZT(1)) - 1.E-05*(ABS(T)+1.E-02)
IF(TTEST.LT.0) RETURN
DNOMR = ZT(3)*ZP(2)-ZP(3)*ZT(2)
IF(DNOMR.EQ.0) GO TO 6
RX = RX+(ZT(3)*(P-ZP(1))-ZP(3)*(T-ZT(1)))/DNOMR
DNOMR = ZT(2)*ZP(3)-ZP(2)*ZT(3)
IF(DNOMR.EQ.0) GO TO 6
EX = EX+(ZT(2)*(P-ZP(1))-ZP(2)*(T-ZT(1)))/DNOMR
GO TO 4
IFL = 0
RETURN
END
SUBROUTINE T4RTPEX(IR, IDT, TBLS, R, T, P, E, IFL)

  PURPOSE  TO FIND PRESSURE AND ENERGY AS FUNCTIONS
  OF DENSITY AND TEMPERATURE FROM A
  SESAME TYPE 302 TABLE USING NEWTON'S METHOD.

  ARGUMENTS
  IR   (INPUT)  REGION NO.
  IDT  (INPUT)  DATA TYPE FOR 302 TABLES
  TBLS (INPUT)  TABLE STORAGE ARRAY
  T    (INPUT)  TEMPERATURE
  P    (OUTPUT) PRESSURE
  E    (OUTPUT) ENERGY
  IFL  (OUTPUT) OUTPUT FLAG
  *1 FOR SUCCESS
  *0 FOR FAILURE

  REMARKS  NONE

  PROGRAMMER  J. ABDALLAH, JR.

  DATE      5 JULY 1979

  COMMON/S20IRX/LCMX,NRS,LCFW(10,3)
  DIMENSION TBLS(I)
  COMMON/SESINX/IRXX,IDTX,RX,EX,IBR,IFLX
  COMMON/SESOUTX/ZP(3),ZT(3)
  IFR=O
  IFLX=1
  RX=R
  IRXX=IR
  IDTX=IDT
  LOC=LCFW(IR, IDT)
  NR=TBLS(LOC+2)
  NE=TBLS(LOC+3)

  GET INITIAL GUESS ON ENERGY
  FIND CLOSEST DENSITY INDEX
  LOCX=LOC+4
  IRX=1
  OELS=ABS(R-TBLS(LOCX))
  IF(NR.EQ.1) GO TO 20
  DO 10 J=2,NR
  10  LDUX=LOCX+1
  OEL=ABS(R-TBLS(LOCX))
  IF(OEL.GT.OELS) GO TO 10
  IRX=J
  OELS=OEL
  CONTINUE

  FIND THE ENERGY INDEX ASSOCIATED WITH THE CLOSEST TEMP
  LOCX=LOC+3+NR+NE+NR+IRX
  DELS=TBLS(LOCX)
  OELS=SHIFT(DELS,30)
  DELS=ABS(T-OELS)
  IEX=1
  IF(NE.EQ.1) GO TO 40
  DO 30 J=2,NE
  30  LOCX=LOCX+NR

91
63 DEL=TBLS(LOCX)
64 DEL=SHIFT(DEL,30)
65 DEL=ABS(T-DEL)
66 IF(DEL.GT.DEI) GO TO 30
67 IEX=J
68 DELS=DEL
69 CONTINUE
70 C.. INITIAL GUESS ON ENERGY
71 40 EX=TBLS(LOC+3+NR+IEX)+TBLS(LOC+3+NR+NE+IRX)
72 C.. ITERATE USING NEWTONS METHOD
73 K=0
74 IFL=1
75 50 K=K+1
76 IF(K.EQ.50) GO TO 90
77 CALL T4DATIX
78 E=EX
79 P=ZP(1)
80 TTEST=ABS(T-ZT(1))-1.0E-05*(ABS(T)+1.0E-02)
81 IF(TTEST.LT.0.) RETURN
82 D=ZT(3)
83 IF(D.EQ.0.0) GO TO 90
84 EX=EX-(T-ZT(1))/D
85 GO TO 50
86 90 IFL=0
87 RETURN
88 END
APPENDIX B

CROSS-REFERENCE DIRECTORY OF EOSLIB
<table>
<thead>
<tr>
<th>ROUTINE</th>
<th>INDEX</th>
<th>LENGTH</th>
<th>DSKAD</th>
<th>EXTERNAL SYMBOLS</th>
<th>COMMON BLOCKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. DPAOX</td>
<td>18</td>
<td>15</td>
<td>017113</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. EDSBEG</td>
<td>5</td>
<td>7</td>
<td>002354</td>
<td>S20IRX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESDATX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESINX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>INTORDX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSCCE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSCCO</td>
<td></td>
</tr>
<tr>
<td>3. EDSCON</td>
<td>1</td>
<td>81</td>
<td>000000</td>
<td>EDSOSL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>GDODER.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSCE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC0</td>
<td></td>
</tr>
<tr>
<td>4. EDSDE</td>
<td>6</td>
<td>81</td>
<td>005404</td>
<td>T4DATIX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDKUT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S20IRX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESDATX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESINX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>INTORDX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td>5. EDSOPT</td>
<td>7</td>
<td>80</td>
<td>006173</td>
<td>T4DATIX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDKUT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S20IRX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESDATX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESINX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>INTORDX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td>6. EDSOSL</td>
<td>2</td>
<td>6</td>
<td>000454</td>
<td>OUTCI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>REWIND.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSFAS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OECODI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>INPCI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FEXIST</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC6</td>
<td></td>
</tr>
<tr>
<td>7. EDSEFD</td>
<td>4</td>
<td>168</td>
<td>001320</td>
<td>ASSIGN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>QASSIGN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC6</td>
<td></td>
</tr>
<tr>
<td>8. EDSFAS</td>
<td>3</td>
<td>74</td>
<td>000644</td>
<td>OUTCI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>REWIND.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSFAS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OECODI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>INPCI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FEXIST</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC6</td>
<td></td>
</tr>
<tr>
<td>9. EDSEGET</td>
<td>11</td>
<td>231</td>
<td>011374</td>
<td>OUTCI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>REWIND.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSFAS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EGETDX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>GDLXPER.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSCE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSOX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSO0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S20IRX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESDATX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESINX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>INTORDX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td>10. EDSIPT</td>
<td>9</td>
<td>128</td>
<td>007722</td>
<td>OUTCI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSGET</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TAPRNX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSKE1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S20IRX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESDATX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SESINX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>INTORDX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EDSC4</td>
<td></td>
</tr>
<tr>
<td>11. EDSIPT</td>
<td>8</td>
<td>128</td>
<td>007020</td>
<td>OUTCI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TAPRNX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S20IRX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>12. EDSKUT</td>
<td>EDSKUT</td>
<td>10</td>
<td>123</td>
<td>010624</td>
<td>EDSGET</td>
</tr>
<tr>
<td>13. EDSKFD</td>
<td>EDSKFD</td>
<td>12</td>
<td>145</td>
<td>012524</td>
<td>OUTCI.</td>
</tr>
<tr>
<td>14. EDSORT</td>
<td>EDSORT</td>
<td>13</td>
<td>106</td>
<td>013464</td>
<td>OUTCI.</td>
</tr>
<tr>
<td>15. GETEDSX</td>
<td>GETEDSX</td>
<td>14</td>
<td>171</td>
<td>014330</td>
<td>OLDG10</td>
</tr>
<tr>
<td>16. GETINVX</td>
<td>GETINVX</td>
<td>21</td>
<td>201</td>
<td>021107</td>
<td>TDQAX</td>
</tr>
<tr>
<td>17. GETRPDX</td>
<td>GETRPDX</td>
<td>26</td>
<td>139</td>
<td>025611</td>
<td>TDQAX</td>
</tr>
<tr>
<td>18. INBUF</td>
<td>INBUF</td>
<td>17</td>
<td>29</td>
<td>01627</td>
<td>TDQAX</td>
</tr>
<tr>
<td>19. INV301X</td>
<td>INV301X</td>
<td>24</td>
<td>166</td>
<td>023711</td>
<td>TDQAX</td>
</tr>
<tr>
<td>20. ISRM</td>
<td>ISRM</td>
<td>19</td>
<td>55</td>
<td>017303</td>
<td>TDQAX</td>
</tr>
<tr>
<td>21. MATCHX</td>
<td>MATCHX</td>
<td>16</td>
<td>22</td>
<td>016437</td>
<td>TDQAX</td>
</tr>
<tr>
<td>22. RATFN1X</td>
<td>RATFN1X</td>
<td>22</td>
<td>128</td>
<td>022237</td>
<td>TDQAX</td>
</tr>
<tr>
<td>23. TD4AT1X</td>
<td>TD4AT1X</td>
<td>23</td>
<td>124</td>
<td>022770</td>
<td>TDINTPX</td>
</tr>
<tr>
<td>24. TD4ATX</td>
<td>TD4ATX</td>
<td>27</td>
<td>130</td>
<td>026551</td>
<td>TDINTPX</td>
</tr>
<tr>
<td>25. TD4INTP</td>
<td>TD4INTP</td>
<td>20</td>
<td>289</td>
<td>017702</td>
<td>TD4AT1X</td>
</tr>
</tbody>
</table>

...
<table>
<thead>
<tr>
<th></th>
<th>T4RTPEX</th>
<th>T4RTPEX</th>
<th></th>
<th>T4RTPEX</th>
<th>T4RTPEX</th>
<th></th>
<th>T4RTPEX</th>
<th>T4RTPEX</th>
<th></th>
<th>T4RTPEX</th>
<th>T4RTPEX</th>
<th></th>
<th>T4RTPEX</th>
<th>T4RTPEX</th>
<th></th>
<th>T4RTPEX</th>
<th>T4RTPEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>26</td>
<td>T4RTPEX</td>
<td></td>
<td>196</td>
<td>014053</td>
<td></td>
<td>T4DATIX</td>
<td>ISRX</td>
<td></td>
<td></td>
<td>S2DIRX</td>
<td>SESOUTX</td>
<td></td>
<td>SESINX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>T4RTPEX</td>
<td></td>
<td>258</td>
<td>024613</td>
<td></td>
<td>T4DATIX</td>
<td></td>
<td></td>
<td></td>
<td>S2DIRX</td>
<td>SESOUTX</td>
<td></td>
<td>SESINX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>TARRAX</td>
<td>TARRAX</td>
<td>278</td>
<td>037666</td>
<td></td>
<td>INBUFEX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO.</td>
<td>ENTRY PT</td>
<td>CALLED BY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>----------</td>
<td>-----------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>EOSCON</td>
<td>EOSGET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>EOSDSL</td>
<td>EOSCON</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>EOSFAS</td>
<td>EOSEFO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>EOSEFO</td>
<td>EOSGET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>EOSBEG</td>
<td>EOSKUT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>EOSDRE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>EOSDRT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>EOSIRT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>EOSIPT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>EOSKUT</td>
<td>EOSORE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>EOSGET</td>
<td>EOSORE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>EOSOFD</td>
<td>EOSGET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>EOSSORT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>GETEOSX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>T4PTREX</td>
<td>EOSIPT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>MATCHKX</td>
<td>GETEOSX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>INBUFRX</td>
<td>GETINVX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>OPACKX</td>
<td>GETEOSX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>ISRCHKX</td>
<td>T4PTREX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>T4INTPX</td>
<td>T4DATIX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>GETINVX</td>
<td>T4DATX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>RATFN1X</td>
<td>EOSGET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
23 **T4DATIX**
   - **EOSDRE**
   - **T4PTREX**
   - **T4RTPEX**
24 **INV301X**
   - **GETINVX**
25 **T4RTPEX**
   - **EOSIRT**
26 **GETRPOX**
   - **EOSGET**
27 **T4DATX**
   - **EOSORT**
   - **EOSORT**
28 **TABRANX**
   - **GETEOSX**
   - **GETINVX**
   - **GETRPOX**
COMMON BLOCK: USED BY:

EOSCC1 EOSCD EOSCLL EOSCB EOSCR EOSCQ GETEOSX GETINVX
EOSCCD EOSCD EOSCLL EOSCB EOSCR EOSCQ GETINVX
EOSCZ EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS
EOSC1 EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS
EOSC2 EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS
EOSC3 EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS
EOSC6 EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS EOSFAS
S201RX EOSBEG EOSDRE EOSDRE GETINVX T4D3ATI T4D3ATI
SESOS X EOSBEG EOSDRE EOSDRE T4D3ATI T4D3ATI T4D3ATI
SESINX EOSBEG EOSDRE EOSDRE T4D3ATI T4D3ATI T4D3ATI
INTORDX EOSBEG EOSDRE EOSDRE T4D3ATI T4D3ATI T4D3ATI
EOSC4 EOSBEG EOSDRE EOSDRE EOSDRE T4D3ATI T4D3ATI
EOSC7 EOSBEG EOSDRE EOSDRE EOSDRE T4D3ATI T4D3ATI
EOSOUTX EOSDRE EOSDRE EOSDRE T4D3ATI T4D3ATI T4D3ATI
RTBLK2X T4D3ATI T4D3ATI T4D3ATI
RTBLK1X T4D3ATI T4D3ATI T4D3ATI
APPENDIX C

TEST PROGRAM
PROGRAM SES (INPUT, OUTPUT, TAPE6=OUTPUT, TAPE59=TTY, TAPE3)

**************************************************************************************

* PURPOSE -
* TO PROVIDE THE APPROPRIATE EOS SCALE FACTORS FOR THE SYSTEM OF UNITS CHOSEN BY KUNIT (SEE BELOW)

* INPUT VARIABLES -
* LMAT = MATERIAL IN AN AI0 FIELD, FOR EXAMPLE - LMAT = "HELIUM"
* THE MATERIAL SESAME NUMBER CAN ALSO BE USED TO SPECIFY THE MATERIAL BY SETTING LMAT TO THE SESAME NUMBER, FOR EXAMPLE - LMAT = "5760" FOR HELIUM

* R = DENSITY (RHO)
* E = INTERNAL EN

* KEDS = MULTIPLE PARAMETER FLAG TO DESCRIBE HOW TO WRITE AND RETRIEVE THE DATA FILE. KEDS HAS FOUR DECIMAL DIGITS.
* KEDS = 1000*KBR + 100*KUNIT + 10*KREPE + KFN WHERE

* KBR = COMPUTATIONAL MODE FLAG TO INDICATE WHICH QUANTITIES AND THEIR PARTIAL DERIVATIVES ARE TO BE CALCULATED AND RETURNED BY THE PACKAGE.

* = 0 COMPUTE PRESSURE AND TEMPERATURE
* = 1 COMPUTE PRESSURE ONLY
* = 2 COMPUTE TEMPERATURE ONLY

* KUNIT = KIND OF UNITS
* 0 (SESAMEE) R-G/CC,T-DEG.K,D-CM**2/G,P-GPA,E-MJ/KG
* 1 (CGS) R-G/CC.T-DEG.K,P-MUBR,E-ERGS/GM
* 2 (SIU) R-KG/M**3.T-DEG.K,P-PA,E-J/KG,D-CM**2/G
* 3 (HYDROX) R-G/CC,T-DEG.K,P-MBR,E-MBR*CC/GM,D-CM**2/G
* 4 (HYDROXO) R-G/CC,T-KEV,P-MBR,E-MBR*CC/GM,D-CM**2/G
* 6 (LASNEX) R-G/CC,T-KEV.D-CM**2/GM,P-JRKS/CC.ED-JRKS/CC

* LEGEND -
* R = DENSITY
* T = TEMPERATURE
* D = OPACITY
* P = PRESSURE
* E = INTERNAL EN
* CC = CUBIC CENTIMETER
* CM = CENTIMETER
* DEG. K = DEGREES KELVIN
* EV = ELECTRON VOLT
* G = GRAM
* GPA = GIGA PASCALS
* J = JULIUS
* JRKS = JERKS
* KEV = KILO ELECTRON VOLTS
* KG = KILOGRAM
* M = METER
* MBR = MEGABAR
* MUBR = MICROBAR
* PA = PASCAL
• KREN = COMPUTATION FLAG TO INDICATE WHETHER E IS
• IS TO BE REPRESENTED AS EN PER UNIT MASS OR EN
• PER UNIT VOLUME
• O EN IN UNITS OF EN PER UNIT MASS. FOR EXAMPLE-
• UNITS OF ERGS/GRAM WHEN KUNIT = 1. THIS IS THE USUAL E.
• 1 EN IN UNITS OF EN PER UNIT VOLUME. FOR EXAMPLE-
• UNITS OF ERGS/CM**3 WHEN KUNIT = 1. THIS IS THE EN
• DENSITY RH*E COMMONLY COMPUTED IN HYDRODYNAMIC
• COMPUTER CODES.
•
• KFN = KIND OF FUNCTION INTERPOLATION IN THE TABLES
• = 0 RATIONAL APPROXIMATIONS (ACCURATE)
• 1 BILINEAR APPROXIMATIONS (FAST)
•
• IMATE = INDICATES WHETHER TO LOAD THE DATA FILE IF
• IT DOES NOT EXIST FOR LMAT OR GO DIRECTLY TO A
• PREVIOUSLY LOADED FILE.
• =0 CHECK IF THE DATA FILE FOR LMAT HAS BEEN LOADED.
• IF NOT, SEARCH FOR THE FILE, CONVERT IT TO THE PROPER UNITS
• AND COPY IT INTO LCM USING THE STANDARD SESAME FORMAT.
• >0 EQUAL TO THE LMAT TABLE NUMBER. THIS NUMBER WAS RETURNED
• BY THE PACKAGE ON A PREVIOUS CALL TO SUBROUTINE EOSDR,
• EOSIPT OR EOSIRT. THIS OPTION IS FASTER THAN IMATE = 0
• BECAUSE IT SKIPS SEARCH IN THE DIRECTOR OF THE
•
• OUTPUT VARIABLES-
•
• P = ARRAY OF DIMENSION 3 CONTAINING THE PRESSURE AND
• ITS PARTIAL DERIVATIVES. THIS ARRAY MUST BE DIMENSIONED
• EVEN IF THE PARTIAL DERIVATIVES ARE NOT COMPUTED.
•
• P(1) = PRESSURE
• P(2) = DENSITY DERIVATIVE OF THE PRESSURE (DP/DR)
• P(3) = TEMPERATURE DERIVATIVE OF THE PRESSURE (DP/DE)
•
• T = ARRAY OF DIMENSION 3 CONTAINING THE TEMPERATURE AND
• ITS PARTIAL DERIVATIVES. THIS ARRAY MUST BE DIMENSIONED
• EVEN IF THE PARTIAL DERIVATIVES ARE NOT COMPUTED
•
• T(1) = INTERNAL TEMPERATURE
• T(2) = DENSITY DERIVATIVE OF THE TEMPERATURE (DT/DR)
• T(3) = EN DERIVATIVE OF THE TEMPERATURE (DT/OE)
•
• IMATE = INDICATES THE SUCCESS OR FAILURE OF
• LOCATING AND LOADING THE DATA FILE FOR LMAT.
•
• N=0 MATERIAL TABLE NUMBER (SUCCESS)
• 0 MATERIAL (LMAT) NOT IN LIBRARY
• -N (N>1) INSUFFICIENT STORAGE
• THE LCM STORAGE MUST BE INCREASED BY AT LEAST
• N STORAGE LOCATIONS. SEE THE EOSMOD MANUAL
•
• REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE
•
• SAMPLE DRIVER PROGRAM-
•
• PROGRAM TST(OUTPUT)
• DIMENSION P(3),E(3)
• LMAT = "HELIUM"
• R = 0.001
• E = 1.0
• KEOS = 110
•
**IMATE = 0**

CALL EDS0RE(LMAT,R.E.P,T.KEOS,IMATE)

PRINT 10,P(1)

1O FORMAT(" PRESSURE = ",E1O.2," MICROBARS")

CALL EXIT

END

**OUTPUT VARIABLES- IN THE COMMON BLOCKS EOSCCE AND EOSCCO**

**TFACE = TEMPERATURE EOS SCALING FACTOR**

**RFACE = DENSITY EOS SCALING FACTOR**

**PFACE = PRESSURE EOS SCALING FACTOR**

**EFACE = EN EOS SCALING FACTOR**

**TFACO = TEMPERATURE OPACITY SCALING FACTOR**

**RFACO = DENSITY OPACITY SCALING FACTOR**

**OFACO = OPACITY SCALING FACTOR**

**TBLS = ARRAY FOR STORAGE OF THE EOS TABLES**

**LCMx = LENGTH OF THE TBLS ARRAY**

**NRS = UPPER BOUND ON THE NUMBER OF MAT REGIONS LCFW(NRS, )**

**LCFW = ARRAY USED AS A DIRECTORY BY THE SESAME ROUTINES**

**IR = MATERIAL REGION NUMBER**

**IRC = IR (DEFINED TO PERMIT SUBROUTINE CALL**

**IOS2 = SESAME MATERIAL NUMBER**

**TMLS = NAME OF AN ARRAY DESIGNATED FOR THE STORAGE OF TABLES**

**LCNT = CURRENT WORD IN THE ARRAY TBLS**

**LU41 = UNIT NUMBER ASSIGNED TO THE SESAME INPUT FILE SES2CL**

**LU42 = UNIT NUMBER ASSIGNED TO THE SESAME INPUT FILE SES2L**

**LU43 = UNIT NUMBER ASSIGNED TO SES2OP/8 FILES**

**LU44 = UNIT NUMBER ASSIGNED TO MIXLIB (MIXTURES)**

**LU45 = UNIT NUMBER ASSIGNED TO MIXLIB DIRECTORY.**

**KFN = 0 RATIONAL APPROXIMATIONS (ACCURATE)**

**KFN = 1 BILINEAR APPROXIMATIONS (FAST)**

**ZB (OUTPUT FROM GETINVX) AT. CHARGE,AT. CHG**2,MASS

**IDT = DATA TYPE INDICATOR**

**MID (ID) MATERIAL 10 =1 INVERSE TABLES =2 OPACITY**

**EXTERNAL FILES TO HANDLE EOS, OPACITIES AND MIXTURES**

**SES2CL - CLASSIFIED SESAME LIBRARY**

**SES2L - CLASSIFIED SESAME LIBRARY**

**SES2OP - OPACITY TABLE FROM T4**

**MIXLIB - PRIVATE (EOS,OPC) TABLES CREATED BY MIXB(OR MIXER)**

**MIXDIR - DIRECTORY OF MIXTURES ON MIXLIB ( NAME.NO (A10,13))**

**EXTERNALS ANO COMMON BLOCKS-**

**SESAME ROUTINES- S2GET,S2EOS**

**SESAME ROUTINES MATC+0X,TABFCHX,INBUF0X,OPACKX,ISRC+0X,**

**T4INTPX,GETINVX,RATFNX,T4DATXX,INV301X,T4RTPEX**

**SESAME COMMON BLOCKS-S20IRX,RBLK2X,SES0ATX,SESINX,SESOUTX,INTO**

**EOSMOD COMMON BLOCKS- EDSCL,-7**

**EOSMOD COMMON (ALSO INSERTED INTO GETINVX) EOSCCE, EOSCDO**

**EXTERNALS AND COMMON BLOCKS-**

**LOCAL VARIABLES-**

**NONE**

**EXTERNALS AND COMMON BLOCKS-**

**EOSMOD COMMON BLOCKS- EDSCE.EOSCCO**

**REMARKS- THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE**

**PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL**
• REFERENCE- J. M. HYMAN, M. M. KLEIN
• EOSMOD- A SUBROUTINE PACKAGE FOR CALCULATING
• EQUATIONS-OF-STATE AND OPACITIES
• LOS ALAMOS SCIENTIFIC LABORATORY RPT., LA-8502-M, 1980
• DATE- MARCH 6, 1980

**********************************************************************************************

LEVEL 2. TBLS

COMMON /S2DIR/ LCMX, NRS, LCFW(10,3)
COMMON /SESOT/ TBLs(1000)
COMMON /SESIN/ IRC, IDT, DUM(2), KBR, DUM1
COMMON /INTORO/ KFN

COMMON BLOCKS FOR THE EOSMOD ROUTINES

COMMON /EDSCL/ LU41, LU42, LU43, LU44, LU45
COMMON /EDSCE/ LF41, LF42, LF43, LF44, LF45
COMMON /EDSCE/ INIT, IROIM, IR(60,3), KUT(60,3)
COMMON /EDSCE/ NTABLE, NTABLED, IFLP, LCNT
COMMON /EDSC5/ NMAT, LABMAT(60), IOMAT(60)
COMMON /EDSC6/ NMCL, LABMCL(60), IOMCL(60)
COMMON /EDSC7/ NMATO, LABMATO(60), IOMATO(60)

EOSMOD COMMON BLOCKS USED BY THE MODIFIED SESAME ROUTINES

COMMON /EDSCC/ TFACE, RFACE, EFACE, EFACE, KREPE
COMMON /EDSCCD/ TFACO, RFACO, OFACO, KREPO

NAMELIST /INP/ KUNIT, KREP, NLMAT, NBR, NFN, IIPT, IIRT, IDRT, IORE, LCMX
1, NRS, DUM, KBR, DUM1, KFN, INIT, IROIM, NTABLE, NTABLED, IFLP, LCNT, NMAT
2, NMC, NMATD, TFACE, RFACE, EFACE, KREPE, TFACO, RFACO, OFACO, KREPO, KBRP
3, TEMP, PR, EN, IRC, IDT, IDRT, NGO

DATA RHO /0.16/, TEMP /5.E4/, PR /19.2/, EN /310.1/
DATA KUNIT /0/, KREP /0/, NLMAT /2/, NBR /1/, NFN /1/
DATA IIPT /1/, IIRT /1/, IDRT /1/, IDRT /1/, NGO /0/
WRITE (59, INP)

WRITE (3, 210)
DO 10 I=1, NMAT
WRITE (3, 220) LABMAT(I), IOMAT(I)
10 CONTINUE
WRITE (3, 230)
DO 20 I=1, NMATD
WRITE (3, 220) LABMAT(I), IOMAT(I)
20 CONTINUE
WRITE (3, 220) LABMATO(I), IOMATO(I)
30 CONTINUE
READ (59, INP)
WRITE (6, 160)
DO 60 LMATP=1, NLMAT
LMAT=64-HELIUM
LMATO=LMAT
IF (LMATP.EQ.2) LMAT=5760
IF (LMATP.EQ.2) LMATO=15760
DO 50 KBRP=1, NBR
KBR=KBRP-1
DO 40 KFNP=1, NFN
KFN = KFNP - 1

KEOS = 1000 * KBR + 100 * KUNIT + 10 * KREP + KFN

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

40 CONTINUE

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

40 CONTINUE

40 CONTINUE

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

40 CONTINUE

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

CHANGE TABLE FORMAT FOR THE ENERGY AND OPACITY REPRESENTATION

80 CONTINUE

KEOS = KEOS + 10

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

80 CONTINUE

KEOS = KEOS + 10

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

80 CONTINUE

KEOS = KEOS - 10

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

80 CONTINUE

NEGATIVE DENSITY TEST

100 CONTINUE

R = RHO

CALL ES (LMAT, LMATO, R, TEMP, PR, EN, KEOS, IIP, IORT, IIRT, IDRE, IORT, I)

100 CONTINUE

DO 120 I = 1, NMAT

CALL ES (LMAT, LMATO, R, TEMP, PR, EN, KEOS, O, 1, 0, 0, 0, 0)

ENCODE (4.190, LMAT) IDMAT(I)

CALL ES (LMAT, LMATO, R, TEMP, PR, EN, KEOS, O, 1, 0, 0, 0, 0)
120 CONTINUE

130 CONTINUE

DO 140 I=1,NMATD

LMAT=LABMO(I)
CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, 0, 0, 0, 0, 1.0)

ENCODE (5, 200, LMATO) IDMATO(I)

CALL ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS, 0.1, 0.0, 0, 0)

140 CONTINUE

150 CONTINUE

CALL EXITA (1)

160 FORMAT (* KEOS IMATE R T P E*)
170 FORMAT (* CONTINUE?*
180 FORMAT (A1)
190 FORMAT (I4)
200 FORMAT (I5)
210 FORMAT (* THE EOS MATERIALS ARE*
220 FORMAT (1X, A10, I10)
230 FORMAT (* THE OPACITY MATERIALS ARE*)
240 END
SUBROUTINE ES (LMAT, LMATO, RHO, TEMP, PR, EN, KEOS.IIPT, IORT, IIRT, IORE
1 , IORT, IWP)

* PURPOSE -
* INPUT VARIABLE -
* INPUT-OUTPUT VARIABLES IN THE COMMON BLOCKS EOSC2 AND EOSCC0
* EXTERNALS AND COMMON BLOCKS -
* REMARKS - THIS SUBROUTINE IS PART OF THE EOSMOD PACKAGE
* PROGRAMMER - J. M. HYMAN AND M. KLEIN, GROUP T-7. LASL
* REFERENCES - J. M. HYMAN AND M. KLEIN
PROGRAMMER- J. M. HYMAN AND M. KLEIN, GROUP T-7, LASL
EOSMOO- A SUBROUTINE PACKAGE FOR CALCULATING
EQUATIONS-OF-STATE AND OPACITIES
LOS ALAMOS SCIENTIFIC LABORATORY RPT. .LA-B502-M, 1980
DATE- MARCH 22, 1981

DIMENSION R(3), P(3), T(3), E(3)
LEVEL 2. TBLS
COMMON /S2DIR/ LCMX, NRS, LCFW(10,3)
COMMON /SESDAT/ TBLS(11000)
COMMON /SESIN/ IRC, IOT, OUM(2), KBR, OUM1
COMMON /INTORD/ KFN
COMMON BLOCKS FOR THE EOSMOO ROUTINES
COMMON /EOSC1/ LU41, LU42, LU43, LU44, LU45
COMMON /EOSC2/ LF41, LF42, LF43, LF44, LF45
COMMON /EOSC3/ INIT, IROIM, IR(60,3), KUT(60,3)
COMMON /EOSC4/ NTABLE, NTABLO, IFLP, LCNT
COMMON /EOSC5/ NMAT, LABMAT(60), IDMAT(60)
COMMON /EOSC6/ NMAC, LABMCL60), IDMCL60)
COMMON /EOSC7/ NMATO, LABMATO(60), IDMATO(60)
COMMON /EOSC8/ TFACE, RFACE, PFACE, EFACE, KREPE
COMMON /EOSC9/ TFACO, RFACO, OFACO, KREPO
COMMON /EOSC0/ TFACO, RFACO, OFACO, KREPO
NAMELIST /INP/ KUNIT, KREP, NLMAT, NBR, NFN, IIPT, IIRT, IORT, IORE, LCMX
1 , NRS, OUM, KBR, OUM1, KFN, INIT, IRODIM, NTABLE, NTABLO, IFLP, LCNT, NMAT
2 , NMAT, NMAC, TFACE, RFACE, EFACE, KREPE, TFACO, RFACO, OFACO, KREPO, RHO
3 , TEMP, PR, EN, IRC, IOT, IORT
R(1)=RHO
T(1)=TEMP
P(1)=PR
E(1)=EN
*************** EOSIPT TEST ***************
IF (IIPT.EQ.O) GO TO 10
WRITE (6.70)
WRITE (3.70)
IMATE=0
R(1)=R(2)=R(3)=0.0
E(1)=E(2)=E(3)=0.0
CALL EOSIPT (LMAT,P,T,R,E,KEDS,IMATE)
WRITE (6,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I).I=1,3)
WRITE (3,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I).I=1,3)
IF (IWNL.NE.0) WRITE (3,INP)
IF (IMATE.LE.0) IMATE=0
CONTINUE

*************** EosIPT TEST ***************
IF (IORT.EQ.0) GO TO 20
WRITE (6,90)
WRITE (3,90)
IMATE=0
P(1)=P(2)=P(3)=0.0
E(1)=E(2)=E(3)=0.0
CALL EOSIPT (LMAT,R,T,P,E,KEDS,IMATE)
WRITE (6,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I).I=1,3)
WRITE (3,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I).I=1,3)
IF (IWNL.NE.0) WRITE (3,INP)
IF (IMATE.LE.0) IMATE=0
CONTINUE

*************** EosIPT TEST ***************
IF (IIRT.EQ.0) GO TO 30
WRITE (6,90)
WRITE (3,90)
IMATE=0
P(1)=P(2)=P(3)=0.0
E(1)=E(2)=E(3)=0.0
CALL EOSIPT (LMAT,R,T,P,E,KEDS,IMATE)
WRITE (6,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I).I=1,3)
WRITE (3,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I).I=1,3)
IF (IWNL.NE.0) WRITE (3,INP)
IF (IMATE.LE.0) IMATE=0
CONTINUE
**EosDre Test**

```fortran
125 C
126 C
127 C
128 IF (IDRE.EQ.0) GO TO 40
129 WRITE (6,110)
130 WRITE (3,110)
131 IMATE=0
132 P(1)=P(2)=P(3)=0.0
133 T(1)=T(2)=T(3)=0.0
134 CALL EOSDRE (LMAT,R.E,P,T,KEOS,IMATE)
135 WRITE (6,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I),I*1.3)
136 WRITE (3,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I),I*1.3)
137 IF (IWNL.NE.0) WRITE (3,INP)
138 IF (IMATE.LE.0) IMATE=0
139 C
140 P(1)=P(2)=P(3)=0.0
141 T(1)=T(2)=T(3)=0.0
142 CALL EOSDRE (LMAT,R.E,P,T,KEOS,IMATE)
143 WRITE (6,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I),I*1.3)
144 WRITE (3,60) LMAT,KEOS,IMATE,(R(I),T(I),P(I),E(I),I*1.3)
145 IF (IWNL.NE.0) WRITE (3,INP)
146 IF (IMATE.LE.0) IMATE=0
147 40 CONTINUE
148 C
149 C
150 C
151 IF (IDRT.EQ.0) GO TO 50
152 KOPC=KEOS
153 WRITE (6,100)
154 WRITE (3,100)
155 IMATD=0
156 OPACITY=0.0
157 CALL EOSORT (LMATD,R.T,OPACITY,KOPC,IMATD)
158 WRITE (6,60) LMATD,OPACITY,KOPC,(R(I),T(I),OPACITY)
159 WRITE (3,60) LMATD,OPACITY,KOPC,(R(I),T(I),OPACITY)
160 IF (IWNL.NE.0) WRITE (3,INP)
161 IF (IMATD.LE.0) IMATD=0
162 C
163 OPACITY=0.0
164 CALL EOSORT (LMATD,R.T,OPACITY,KOPC,IMATD)
165 WRITE (6,60) LMATD,KOPC,IMATD,R(I),T(I),OPACITY
166 WRITE (3,60) LMATD,KOPC,IMATD,R(I),T(I),OPACITY
167 IF (IWNL.NE.0) WRITE (3,INP)
168 IF (IMATD.LE.0) IMATD=0
169 50 CONTINUE
170 C
171 RETURN
172 C
173 60 FORMAT (1X,A10.14,13(1PE9.2))
174 70 FORMAT (* Eosipt* )
175 80 FORMAT (* Eosort* )
176 90 FORMAT (* Eosrt* )
177 100 FORMAT (* Eosert* )
178 110 FORMAT (* Eosde* )
179 END
```
OUTPUT
<table>
<thead>
<tr>
<th>KEOS</th>
<th>INATE</th>
<th>R</th>
<th>T</th>
<th>P</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>EOSIPT</td>
<td>010</td>
<td>1.95E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>HELLM</td>
<td>011</td>
<td>1.99E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>EOSIPT</td>
<td>010</td>
<td>1.95E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>HELLM</td>
<td>011</td>
<td>1.99E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>EOSIPT</td>
<td>010</td>
<td>1.95E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>HELLM</td>
<td>011</td>
<td>1.99E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>EOSIPT</td>
<td>010</td>
<td>1.95E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>HELLM</td>
<td>011</td>
<td>1.99E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>EOSIPT</td>
<td>010</td>
<td>1.95E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>HELLM</td>
<td>011</td>
<td>1.99E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>EOSIPT</td>
<td>010</td>
<td>1.95E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
<tr>
<td>HELLM</td>
<td>011</td>
<td>1.99E-01</td>
<td>5.00E+04</td>
<td>1.92E+01</td>
<td>3.10E+02</td>
</tr>
</tbody>
</table>

**NOTES:**
- The units have changed since the last use of the table.
- The current values of KUNIT and KREP are 1000.
- The previous values of KUNIT and KREP were 0.
- The units have changed since the last use of the table.
- The current values of KUNIT and KREP are 1000.
- The previous values of KUNIT and KREP were 0.
- The units have changed since the last use of the table.
- The current values of KUNIT and KREP are 1000.
- The previous values of KUNIT and KREP were 0.
- The units have changed since the last use of the table.
- The current values of KUNIT and KREP are 1000.
- The previous values of KUNIT and KREP were 0.
UNABLE TO LOCATE MATERIAL 99999
IN SUBROUTINE EOSGET
99999
0 -2 0.
5.00E+04 0.
0. 0. 0. 0. 0. 0. 0. 0. 0.
UNABLE TO LOCATE MATERIAL 99999
IN SUBROUTINE EOSGET
99999
0 -2 0.
5.00E+04 0.
0. 0. 0. 0. 0. 0. 0. 0. 0.
EDSOR
UNABLE TO LOCATE MATERIAL 99999
IN SUBROUTINE EOSGET
99999
0 -2 0.
0. 0. 0. 0. 0. 0. 0. 0. 0.
UNABLE TO LOCATE MATERIAL 99999
IN SUBROUTINE EOSGET
99999
0 -2 0.
0. 0. 0. 0. 0. 0. 0. 0. 0.
EDSOR
THE DENSITY = 0. OR THE TEMPERATURE = 0. IS NONPOSITIVE
99999
0 -1 0.
0. 0.
THE DENSITY = 0. OR THE TEMPERATURE = 0. IS NONPOSITIVE
18760
0 -1 0.
0. 0.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>001-025</td>
<td>$5.00</td>
<td>A02</td>
<td>151-175</td>
<td>$11.00</td>
<td>A08</td>
</tr>
<tr>
<td>026-050</td>
<td>6.00</td>
<td>A03</td>
<td>176-200</td>
<td>12.00</td>
<td>A09</td>
</tr>
<tr>
<td>051-075</td>
<td>7.00</td>
<td>A04</td>
<td>201-225</td>
<td>13.00</td>
<td>A10</td>
</tr>
<tr>
<td>076-100</td>
<td>8.00</td>
<td>A05</td>
<td>226-250</td>
<td>14.00</td>
<td>A11</td>
</tr>
<tr>
<td>101-125</td>
<td>9.00</td>
<td>A06</td>
<td>251-275</td>
<td>15.00</td>
<td>A12</td>
</tr>
<tr>
<td>126-150</td>
<td>10.00</td>
<td>A07</td>
<td>276-300</td>
<td>16.00</td>
<td>A13</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>301-325</td>
<td>$17.00</td>
<td>A14</td>
<td>326-350</td>
<td>18.00</td>
<td>A15</td>
</tr>
<tr>
<td>351-375</td>
<td>19.00</td>
<td>A16</td>
<td>376-400</td>
<td>20.00</td>
<td>A17</td>
</tr>
<tr>
<td>401-425</td>
<td>21.00</td>
<td>A18</td>
<td>426-450</td>
<td>22.00</td>
<td>A19</td>
</tr>
<tr>
<td>426-450</td>
<td>22.00</td>
<td>A19</td>
<td>451-475</td>
<td>$23.00</td>
<td>A20</td>
</tr>
<tr>
<td>476-500</td>
<td>24.00</td>
<td>A21</td>
<td>501-525</td>
<td>25.00</td>
<td>A22</td>
</tr>
<tr>
<td>526-550</td>
<td>26.00</td>
<td>A23</td>
<td>551-575</td>
<td>27.00</td>
<td>A24</td>
</tr>
<tr>
<td>576-600</td>
<td>28.00</td>
<td>A25</td>
<td>601-up</td>
<td>$1.00</td>
<td>A99</td>
</tr>
</tbody>
</table>

*Add $1.00 for each additional 25-page increment or portion thereof from 601 pages up.