TRIDYN -
Binary Collision Simulation of Atomic Collisions
Dynamic Composition Changes in Solids

Wolfhard Möller and Wolfgang Eckstein

IPP 9/64

May 1988

MAX-PLANCK-INSTITUT FÜR PLASMAPHYSIK

8046 GARCHING BEI MÜNCHEN

MAX-PLANCK-INSTITUT FÜR PLASMAPHYSIK

GARCHING BEI MÜNCHEN

TRIDYN -
Binary Collision Simulation of Atomic Collisions
Dynamic Composition Changes in Solids

Wolfhard Möller and Wolfgang Eckstein

IPP 9/64 May 1988

Die nachstehende Arbeit wurde im Rahmen des Vertrages zwischen dem Max-Planck-Institut für Plasmaphysik und der Europäischen Atomgemeinschaft über die Zusammenarbeit auf dem Gebiete der Plasmaphysik durchgeführt.

TRIDYN -

Binary Collision Simulation of Atomic Collisions and Dynamic Composition Changes in Solids

W. Möller and W. Eckstein

Max-Planck-Institut für Plasmaphysik, EURATOM Association,

D-8046 Garching/München, FRG

Program Summary

Title of program:	TRIDYN (Version 3.1)
Computer:	CRAY-XMP
Installation:	Max-Planck-Institut für Plasmaphysik
Operating system:	COS 1.16
Programming language:	FORTRAN 77
High speed storage used:	91 kwords
Number of bits in a word:	64
Peripherals used:	Line printer

Number of lines in combined program and test deck: 1661

Keywords: Atomic collisions, collision cascades, binary collision approximation, ion ranges, ion reflection, (preferential) sputtering, ion mixing, ion implantation

Nature of the physical problem:

A beam of fast ions (energy range app. 1 eV/amu to 100 keV/amu) entering a solid substance is slowed down and scattered due to electronic interaction and nuclear collisions. Along its path, an individual projectile may create fast recoil atoms which in turn may initiate collision cascades of moving target atoms. These may either leave the surface (be sputtered) or be deposited at a site different from their original one. Together with the projectiles being deposited in the substance, this results in local composition changes. In the case of large implantation fluences, these phenomena will cause collisional mixing in layered substances, changes of the surface composition due to preferential sputtering, and the establishment of a stationary range profile of the implanted ions.

Method of solution:

The paths of the individual moving particles and their collisions are modelled by means of the binary collison approximation for an amorphous substance, using a screened Coulomb potential for nuclear collisions and local or nonlocal free-electron-gas approximations for the electronic energy loss. For each nuclear collision, the impact parameter, the azimuthal deflection angle and the species of the collision partner are determined from random numbers. A proper scaling is chosen so that each incident projectile ('pseudoprojectile') represents an interval of implantation fluence. Subsequent to the termination of each pseudo-projectile and its associated collision cascades, the local partial densities of the constituents are rearranged according to their atomic

volumes. In order to make advantage of vector processing, the time-consuming sections of the code have been written in vectorized form, where possible.

Typical running time:

The running time depends strongly on the problem chosen and is mainly influenced by the number of pseudoprojectiles, their energy and their atomic species. For the test example of 1 keV Ne atoms incident on Ta_2O_5, a calculation with $6 \cdot 10^4$ pseudoprojectiles corresponding to a fluence of $6 \cdot 10^{16}$ Ne/cm^2 requires 18 min on the CRAY-XMP computer.

Long Write-Up

1. Introduction

Computer simulations of atomic collisions based on the binary collision approximation have proven to be a powerful tool for about 30 years (earlier and recent reviews are found in refs. /1-4/). A number of different codes have been developed being applicable to substances of different structure, or, for example, differing in the choice of the interaction potentials. Among these programs, the TRIM code originally written by Biersack and Haggmark /5/ provides reasonably fast simulations of collisions in amorphous substances, i.e. using fixed free pathlength between subsequent nuclear collisions. Its main advantage is a fast approximative solution of the scattering integral, which derives the polar deflection angle from the randomly chosen impact parameter of the collision. In the mean time, the TRIM code became the most wide-spread one being accessible to many laboratories around the world.

Originally, TRIM had been set up for the slowing down of fast projectiles only, disregarding any target recoil atoms. (A recent version of this program is described in ref. /6/.) Later, TRIM was extended to include recoil atoms, so that simulations of sputtering could be performed with TRIM.SP /7/. A further version TRSP2C /8,9/ allowed simulations of sputtering and preferential sputtering with two-component target substances. Recently /10/, the speed of TRIM sputtering simulations could be increased considerably by means of the vectorized version TRSPV1C to be used with vector processors.

All of the TRIM simulations mentioned above are performed with a static target substance, i.e. being strictly valid only in the limit of low implantation fluences. In reality, the ion beam induces changes of the substance both due to the deposition of the projectiles and due to collisional transport, including sputtering, in polyatomic media. In order to simulate the resulting phenomena of fluence-dependent ion deposition, preferential sputtering, and atomic mixing, a 'dynamic' version of the TRIM code has been developed /11,12/ which was called TRIDYN. Its earlier, non-vectorized versions were used for preferential sputtering calculations with surprisingly good agreement with experimental findings /9,13/, proving the dominance of collisional transport in some non-metallic compounds exposed to ion-bombardment.

The present paper describes a recent vectorized version (version 3.1) of the TRIDYN program, which typically reduces the computing time by a factor of 1.5 to 2 compared to the earlier versions. It should be noted that simulations of the present type cannot be vectorized very effectively, as the calculations have to be interrupted by many checks on, e.g., particle energies and positions. Nevertheless, the above reduction is believed to be important in view of computing times of the order of one or several hours.

2. Computational Procedure

2.1 Atomic Transport

The transport of moving atoms is described in the binary collision approximation by a sequence of elastic binary collisions with the atoms of

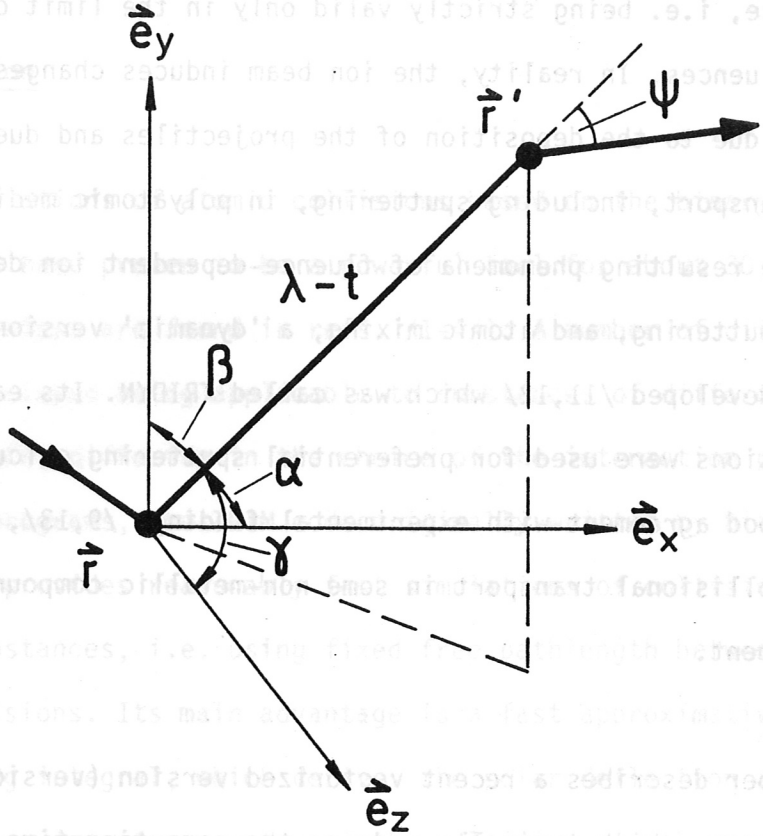

Fig. 1 Path of a moving atom with two collisions. Its direction is defined by the directional angles with respect to the unit vectors e_x, e_y, and e_z. By definition, e_x points in inward direction normal to the surface. The azimuthal scattering angle ϕ has been omitted for clarity.

the substance. In the present model for an amorphous substance, straight mean free paths are assumed between the collisions, the length λ of which is fixed for a given atomic density of the substance (see Fig. 1). Right after a collision, the state of an atom is given by its position $r = (x,y,z)$, its directional angles α, β, and γ with respect to a fixed Cartesian system, and its energy E. The locus after the subsequent collision is then given by

$$\vec{r}' = \vec{r} + (\lambda - t)\begin{pmatrix}\cos\alpha\\ \cos\beta\\ \cos\gamma\end{pmatrix} \tag{1}$$

where t denotes the distance of the asymptotic deflection point from the plane which is defined by the original position of the target atom (see Fig. 2). Each collision is characterized by a polar deflection angle, ψ, and an azimuthal one, ϕ. Both are determined randomly which will be described below. The new direction of the particle is then given by

$$\begin{aligned}
\cos\alpha' &= \cos\psi\cos\alpha + \sin\psi\cos\phi\sin\alpha\\
\cos\beta' &= \cos\psi\cos\beta - \frac{\sin\psi}{\sin\alpha}(\cos\phi\cos\alpha\cos\beta - \sin\phi\cos\gamma)\\
\cos\gamma' &= \cos\psi\cos\gamma - \frac{\sin\psi}{\sin\alpha}(\cos\phi\cos\alpha\cos\gamma + \sin\phi\cos\beta)
\end{aligned} \tag{2}$$

and its energy by

$$E' = E - T - \Delta E_l - \Delta E_{nl} \quad , \tag{3}$$

where T denotes the elastic energy transfer to the target atom, and ΔE the electronic energy losses, which may occur locally during the collisions or nonlocally along the straight paths between the collisions.

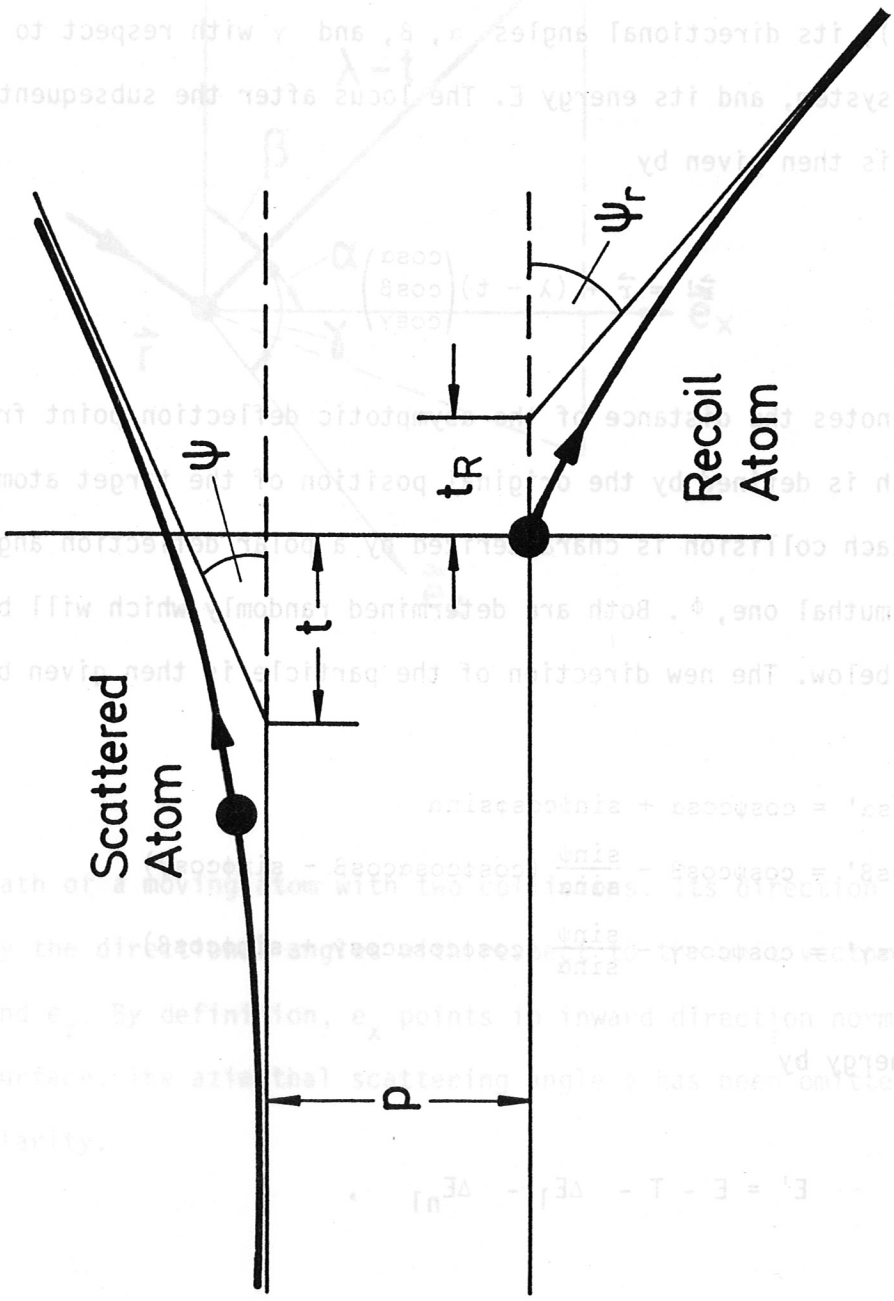

Fig. 2 Scattering geometry for an elastic two-body interaction with the target atom being initially at rest.

In eq. (1), λ is chosen from the local atomic density, n, of the substance:

$$\lambda = n^{-1/3} \quad . \tag{4}$$

In a multicomponent substance, which is composed from N_c constituents denoted by $j = 1, \ldots, N_c$, n depends on the fractional compositions of the different atomic species, q_j:

$$n^{-1} = \sum_j q_j n_{o,j}^{-1} \quad . \tag{5}$$

$n_{o,j}$ denote the atomic densities of the pure components; they might, however, also be chosen to fit the correct atomic density of a given compound. By definition, the fractional composition add up to unity:

$$\sum_j q_j = 1 \tag{6}$$

Whereas ΔE_{nl} depends on the local composition of the substance, ψ, T, and ΔE_l depend on the atomic mass and charge of the target atom. Therefore, its species j_b is determined before each collision randomly according to the local composition. For a given random number r_s, j_b results implicitly as

$$j_b = \min \left(j : \sum_{i=1}^{j} q_i \geq r_s \right) \quad . \tag{7}$$

2.2 Elastic Collisions

The target atom for each atomic collision is chosen within a disc of radius p_{max} (Fig. 3). One collision takes place per atomic volume provided

$$p_{max} = (\lambda \pi n)^{-1/2} , \tag{8}$$

i.e. the mean atomic volume is represented by a cylinder of length λ and radius p_{max}. The actual impact parameter p is chosen from a random number r_p according to

$$p = p_{max} \sqrt{r_p} . \tag{9}$$

The azimuthal position of the target atom within the disc is given by the azimuthal deflection angle ϕ, which is determined from its random number r_ϕ according to

$$\phi = 2\pi r_\phi \tag{10}$$

The relation between the polar scattering angle, θ, and the impact parameter is calculated in TRIM by a fast approximation procedure /5/. According to Lindhard et al. /14/, a reduced energy is introduced by

$$\varepsilon = \frac{4\pi\varepsilon_o \, a \, m_b E}{(m_a+m_b) Z_a Z_b e^2} \tag{11}$$

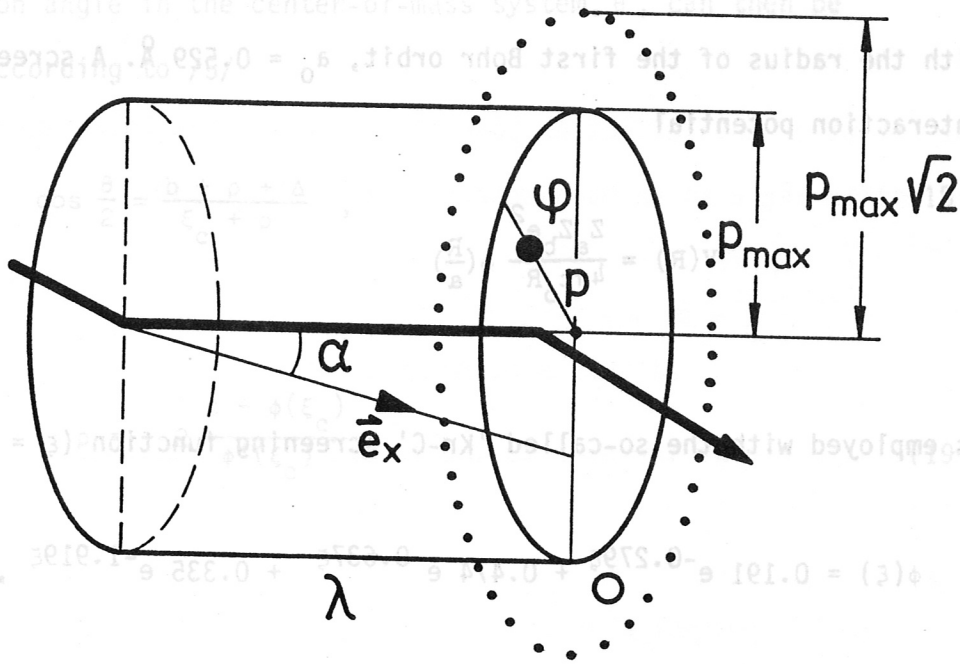

Fig. 3 Definition of the target atom position and the corresponding impact parameter p and azimuthal deflection angle, ϕ. The choice for an additional soft collision of first order is indicated by the dotted circle and the open symbol.

where (m_a, Z_a) and (m_b, Z_b) denote the atomic masses and charges of the scattered atom and target atom, respectively, and a the screening distance according to Firsov /15/

$$a = \frac{0.8853\ a_o}{(Z_a^{1/2}+Z_b^{1/2})^{2/3}} \tag{12}$$

with the radius of the first Bohr orbit, $a_o = 0.529$ Å. A screened Coulomb interaction potential

$$V(R) = \frac{Z_a Z_b e^2}{4\pi\varepsilon_o R}\,\phi(\frac{R}{a}) \tag{13}$$

is employed with the so-called 'Kr-C' screening function ($\xi = R/a$)

$$\phi(\xi) = 0.191\ e^{-0.279\xi} + 0.474\ e^{-0.637\xi} + 0.335\ e^{-1.919\xi}\ , \tag{14}$$

which represents a mean fit to individual interatomic potentials calculated for different combinations of scattering pairs by means of Hartree-Fock calculations /16/. The distance of closest approach, R_c, is calculated from

$$1 - \frac{V(R_c)}{E_c} - (\frac{p}{R_c})^2 = 0 \tag{15}$$

with the center-of-mass energy

$$E_c = \frac{m_b}{m_a+m_b}\,E\ . \tag{16}$$

In reduced form, eq. (15) can be written as ($\xi_c = R_c/a$, $b = p/a$)

$$\frac{b^2}{\xi_c} + \frac{\phi(\xi_c)}{\varepsilon} - \xi_c = 0 \ , \tag{17}$$

which is solved by Newton's method.

The deflection angle in the center-of-mass system, θ , can then be calculated according to /5/

$$\cos\frac{\theta}{2} = \frac{b + \rho + \Delta}{\xi_c + \rho} \ , \tag{18}$$

where

$$\rho = -2\,\frac{\varepsilon - \phi(\xi_c)}{\phi'(\xi_c)} \tag{19}$$

and

$$\Delta = \frac{A \cdot F}{1 + F}\,(\xi_c - b) \ , \tag{20}$$

with

$$A = 2\,(1 + \frac{C_1}{\sqrt{\varepsilon}})\varepsilon b^{(C_2+\sqrt{\varepsilon})/(C_3+\sqrt{\varepsilon})} \tag{21}$$

and

$$F = \frac{C_5+\varepsilon}{C_4+\varepsilon}\,(\sqrt{1+A^2} - A) \ . \tag{22}$$

C_1, ..., C_5 denote fitting parameters given for a specific interaction potential.

Finally, the laboratory deflection angle is determined by

$$\tan\psi = \frac{\sin\theta}{\frac{m_a}{m_b} + \cos\theta} \tag{23}$$

The offset of the deflection point (see eq. (1)) is calculated from the deflection angle in an hard-sphere approximation yielding

$$t = R_c \sin\frac{\theta}{2} \quad . \tag{24}$$

The elastic energy transfer to the target atom is given by

$$T = 4\,\frac{m_a\, m_b\, E}{(m_a+m_b)^2}\,\sin^2\frac{\theta}{2} \quad . \tag{25}$$

The above treatment allows only one collision per mean atomic distance of the substance. However, 'weak' collisions might occur with more distant atoms which might contribute to energy loss and angular deflection. The present version allows up to three additional weak collisions with impact parameters larger than p_{max}, each of them representing one additional atomic volume. In this case, eq. (9) is replaced by (k = 1,2,3):

$$p_k^{weak} = p_{max}\sqrt{k + r_p} \tag{26}$$

2.3 Inelastic Energy Loss

For low ion energies ($E \lesssim 25$ keV/amu), the electronic stopping cross section can be written according to Lindhard and Scharff /27/ as

$$S_{el} = 1.212 \frac{Z_a^{7/6} Z_b}{(Z_a^{2/3}+Z_b^{2/3})^{3/2}} \sqrt{\frac{E}{eV}} \text{ eV Å}^2 \tag{27}$$

From this, the nonlocal inelastic energy loss is calculated by

$$\Delta E_{nl} = (\lambda - t) n S_{el} , \tag{28}$$

and the local one according to Oen and Robinson /18/ by

$$\Delta E = \frac{C_6^2}{2} \frac{S_{el}}{\pi a^2} \exp(-C_6 \xi_c) \tag{29}$$

with a constant C_6 being valid for the interaction potential chosen.

2.4 Initial Conditions - Projectiles

The projectiles which initiate the collision cascades are indexed as the constituent number 1. They may start internally ($x_0 > 0$) or externally ($x_0 \leq 0$).

They might be chosen as monoenergetic (energy E_0) with a fixed angle of incidence α_0, with a two-dimensional cosine angular distribution according to

$$\cos\alpha_o = \sqrt{r_1}$$

$$\cos\beta_o = \sin\alpha_o \cos(2\pi r_2) \qquad (30)$$

$$\cos\gamma_o = \sin\alpha_o \sin(2\pi r_2) \quad ,$$

or with a random angular distribution given by

$$\cos\alpha_o = \begin{cases} 1 - r_1 \, , & x_o \leq 0 \\ 1 - 2r_1 \, , & x_o > 0 \end{cases}$$

$$\cos\beta_o = \sin\alpha_o \cos(2\pi r_2) \qquad (31)$$

$$\cos\gamma_o = \sin\alpha_o \sin(2\pi r_2)$$

from a pair of random numbers r_1, r_2. Alternatively, a Maxwellian energy-angle distribution might be chosen with an ion temperature kT_i and an additional sheath potential V_s. In this case, the squares of the velocity components read

$$v_{ox}^2 = \frac{2}{m_1}(V_s - kT_i \log r_1)$$

$$v_{oy}^2 = -\frac{2}{m_1} kT_i \cos^2(2\pi r_2)\log r_1 \qquad (32)$$

$$v_{oz}^2 = -\frac{2}{m_1} kT_i \sin^2(2\pi r_2)\log r_2 \quad ,$$

and with

$$v_o = \sqrt{v_{ox}^2 + v_{oy}^2 + v_{oz}^2} \qquad (33)$$

the directional angles and the energy result as

$$\cos\alpha_o = \frac{v_{ox}}{v_o}$$

$$\cos\beta_o = \frac{v_{oy}}{v_o} \qquad (34)$$

$$\cos\gamma_o = \frac{v_{oy}}{v_o}$$

and

$$E_o = \frac{m_1}{2} v_o^2 \; . \qquad (35)$$

In the case of an external start ($x_o \geq 0$), the actual start position is chosen at

$$x_s = -\frac{2}{\sqrt{\pi}} n_s^{-1/3} \; , \qquad (36)$$

where n_s denotes the atomic density at the surface. Eqs. (4) and (8) show x_s to be twice as large as p_{max}. Furthermore, a binding energy of the projectile to the surface can be taken into account. The projectile start energy and directions are then modified according to the surface binding energy of the projectile, $E_{s,1}$:

$$E_o' = E_o + E_{s,1} \qquad (37)$$

and

$$\cos\alpha_o' = \left(\frac{E_o \cos^2\alpha_o + E_{s,1}}{E_o + E_{s,1}} \right)^{1/2}$$

$$\cos\beta_o' = \cos\beta_o \frac{\sin\alpha_o'}{\sin\alpha_o} \qquad (38)$$

$$\cos\gamma_o' = \cos\gamma_o \frac{\sin\alpha_o'}{\sin\alpha_o} \quad .$$

Any projectile starting externally is assumed to enter an 'atomically rough' surface. Accordingly, its first position of collision is determined from a random number r by

$$x_1 = x_c + \lambda r \cos\alpha_o'$$

$$y_1 = \lambda\, r \cos\beta_o' \qquad (39)$$

$$z_1 = \lambda\, r \cos\gamma_o'$$

2.5 Recoil Generation

By any hard collision of a projectile or recoil atom with a target atom of species j_b, a new recoil may be generated provided the elastic energy transfer T is larger than the bulk binding energy, E_{b,j_b}. Its initial energy is then

$$E_{bo} = T - E_{b,jb}. \qquad (40)$$

The start position is given by the locus of the collision, as the asymptotic offset t_R (see Fig. 2) vanishes in case of the hard sphere approximation. Its directional angles with respect to the original direction of the scattered particle are given by

$$\tan \psi_r = \frac{\sin\theta}{1-\cos\theta} \tag{41}$$

and

$$\phi_r = \pi - \phi \quad . \tag{42}$$

2.6 Boundary Conditions

Any particle of species j moving in front of the surface between x = 0 and $x = x_s$, is still allowed to interact with the substance in order to include weak external deflections. However, an emerging particle exceeding x_s will be subject to boundary conditions imposed by the model of a planar surface threshold. Provided the normal fraction of its energy, $E\cos^2\alpha$, is smaller than the surface binding energy, $E_{s,j}$, the atom will reenter the surface with the new direction

$$\cos\alpha' = -\cos\alpha \quad . \tag{43}$$

Alternatively, the atom is regarded as reflected (in case of an incident projectile) or sputtered (in case of a recoil atom). Then, the emerging energy is modified according to

$$E' = E - E_{s,j} \quad . \tag{44}$$

If statistics on the emerging directions are desired, the final directions may be obtained in a way similar to eq. (38). This is not included in the present version of TRIDYN. It does neither include a corresponding treatment of transmitted particles in case of thin films.

2.7 Particle Termination

The history of a moving particle of species j is terminated when it has been slowed down to an energy below a predefined cutoff energy, $E_{f,j}$. In order to obtain reflection and sputtering yields correctly, $E_{f,i}$ ($i = 1, \ldots, N_C$) have to be chosen equal to or smaller than the surface binding energies, $E_{s,i}$.

The final position of an incident projectile is taken equal to its last recorded depth. However, in case of a recoil atom, one may decide to treat it as 'free' or 'bound'. In the latter case, it will be put back to its depth of origin, provided its starting energy had been lower than its displacement threshold, $E_{d,j}$. In this way, the program may treat energy transport without mass transport.

2.8 Radiation Damage

The present program calculates a mean number of Frenkel pairs for each primary collision between a projectile and a target atom of type j according to the modified Kinchin-Pease model /19/

$$N_F(T) = \begin{cases} 0 & T \leq \overline{E_d} \\ 1 & E_d \leq T \leq 2.5\,\overline{E_d} \\ \dfrac{0.8\,\nu(T)}{2\overline{E_d}} & T \geq 2.5\,\overline{E_d} \end{cases} \qquad (45)$$

where $\nu(T) \simeq 0.9\ T$ denotes the fraction of the primary recoil energy which is not dissipated into electronic losses. The mean displacement threshold energy is calculated by

$$E_d^{-1} = \sum_i q_i\ E_{d,i}^{-1} \quad . \tag{46}$$

In addition, the Frenkel pairs of each component generated both by primary events and in the collision cascades are recorded, being represented by the corresponding number of recoil atoms set in motion with an energy above the displacement threshold.

2.9 Composition of the Substance

Many of the equations in sects. 2.1 to 2.8 depend on the local composition of the substance. An inhomogeneous substance can be treated by subdividing the substance into N slabs of initially constant thickness, Δx_o. As it will be seen below, the depth intervals might be changed to nonequidistant ones during the calculation. However, outputs of dynamic composition profiles are calculated for equidistant intervals by interpolation.

2.10 Dynamic Relaxation

The principle of the dynamic relaxation of the target substance is depicted in Fig. 4. Each projectile or moving particle in the computer simulation ('pseudoparticle') represents an interval of fluence. If an implantation procedure with a total fluence Φ_{tot} is simulated by a computer simulation

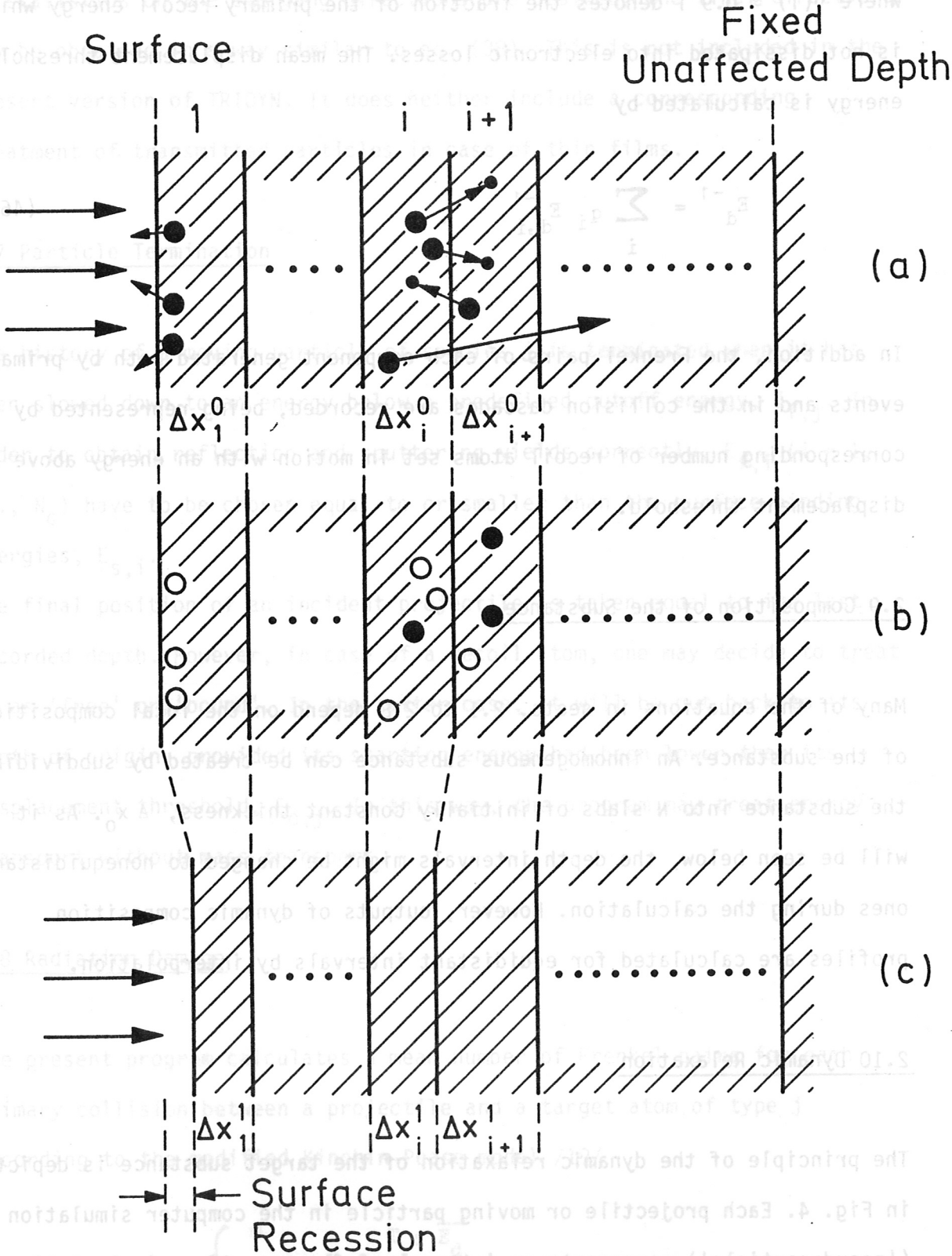

Fig. 4 Schematic representation of collisional transport and dynamic relaxation: Relocations and sputtering caused by bombardment (a) produce vacancies and additional atoms (b), which are allowed to relax (c).

employing N_H incident pseudoprojectile histories, each pseudoparticle represents a differential fluence of

$$\Delta\Phi = \frac{\Phi_{tot}}{N_H} \tag{47}$$

TRIDYN performs the following procedure subsequent to the termination of each pseudoprojectile history. The collision cascades, and the slowing down of the projectile may cause the removal or the deposition of pseudoparticles in different depth layers due to sputtering, atomic relocation, and implantation. Denoting the change of the number of pseudoparticles of type j in each layer i (i = 1, ..., N) by ΔN_{ij}, the new areal densities of real atoms in that layer are given by

$$A_{ij} = q_j n_i \Delta x_i + \Delta N_{ij} \Delta\Phi \quad . \tag{48}$$

For j = 1 (the incident projectile species), a maximum allowed atomic fraction q_1^{max} can be defined in order to simulate local saturation phenomena. The maximum areal density of constituent 1 is then

$$A_{i1}^{max} = \frac{q_1^{max}}{1-q_1^{max}} \sum_{j=2}^{N_c} A_{ij} \quad . \tag{49}$$

If $A_{i1} > A_{i1}^{max}$, an incremental flux of reemitted projectiles is defined by

$$\Delta A_{i1}^{reem} = A_{i1} - A_{i1}^{max} \quad . \tag{50}$$

and eq. (48) (for j = 1) is replaced by

$$A_{i1} = A_{i1}^{max} \quad . \tag{51}$$

From the above procedure, there arise local excess densities or depletions. These are allowed to relax by adjustment of the interval thicknesses:

$$\Delta x_i = \sum_{j=1}^{N_c} A_{ij} \, n_{o,j}^{-1} \tag{52}$$

The incremental surface recession per pseudoprojectile results naturally as the difference of the integrated slab thicknesses,

$$d_{tot} = \sum_{i=1}^{N} \Delta x_i \quad , \tag{53}$$

before and after that pseudoprojectile passage.

The thickness of the individual slabs is always held between $0.5 \cdot \Delta x_o$ and $1.5 \cdot \Delta x_o$. A too narrow interval is combined with its next neighbour; a new interval N is then fetched into the calculation with the atomic fractions

of the last interval at the beginning of the simulation. A too large interval is split into two smaller ones, and the last interval is discarded.

Finally, the new atomic fractions are calculated according to

$$q_{ij} = \frac{A_{ij}}{\sum_{k} A_{ik}} \quad , \tag{54}$$

and the new local atomic densities according to eq. (5).

The statistical quality of the dynamic relaxation procedure depends on the choice of the number of pseudoprojectiles for a given total fluence of projectiles. As a figure of experience, the maximum relative change of the areal density in any layer

$$\left(\frac{\Delta A}{A}\right)_{max} = \max\left(\frac{\Delta\Phi \sum_{j=1}^{N_c} \Delta N_{ij}}{n_i \Delta x_i} \quad , i=1,\ldots,N \right) \tag{55}$$

should not exceed 5% during the complete simulation.

3. Program Description

3.1 Numerical Procedure

The TRIDYN program consists mainly of one module only in order to save computer time which otherwise would be needed for subprogram calls. Where useful and possible, the equations have been written in vectorized form. As the dynamic relaxation procedure is performed subsequent to each pseudo-projectile history, parallel computing is not possible for the incident projectiles.

The main blocks of the program are shown in Fig. 5. Initially, constants are preset from the input data, and the projectile history loop is entered to be run N_H times. Details of the projectile history and the projectile loops are given in Figs. 6 and 7, respectively. The projectile loop is entered with the locus of the first collision. After determining the actual depth interval and the corresponding local quantities, the weak collision loop is entered which finally defines new directions after each of the simultaneous collisions. (Actually, the last passage of the weak collision loop represents the hard collision.) A primary recoil may be generated and stored for each hard collision. The projectile loop is left after termination of the projectiles' history with the primary recoil atoms having been generated stored in list 1.

The collision cascade loop is essentially of the same structure as the projectile loop. Its vectorized equations, however, allow a simultaneous treatment of recoil atoms over large fractions of the loop. The loop is

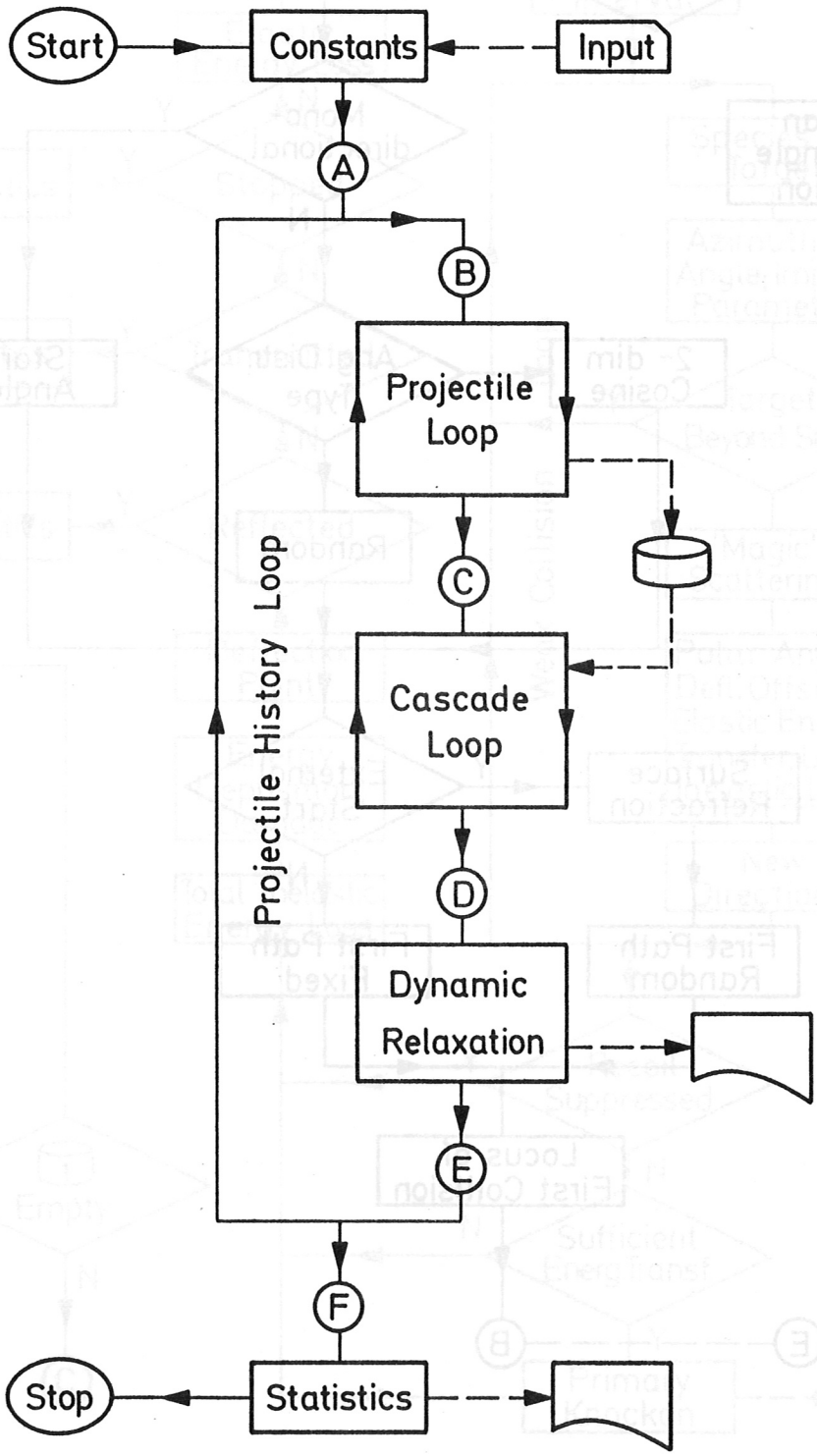

Fig. 5 Block flow chart of TRIDYN. (A) to (F) denote the connection points of the different units.

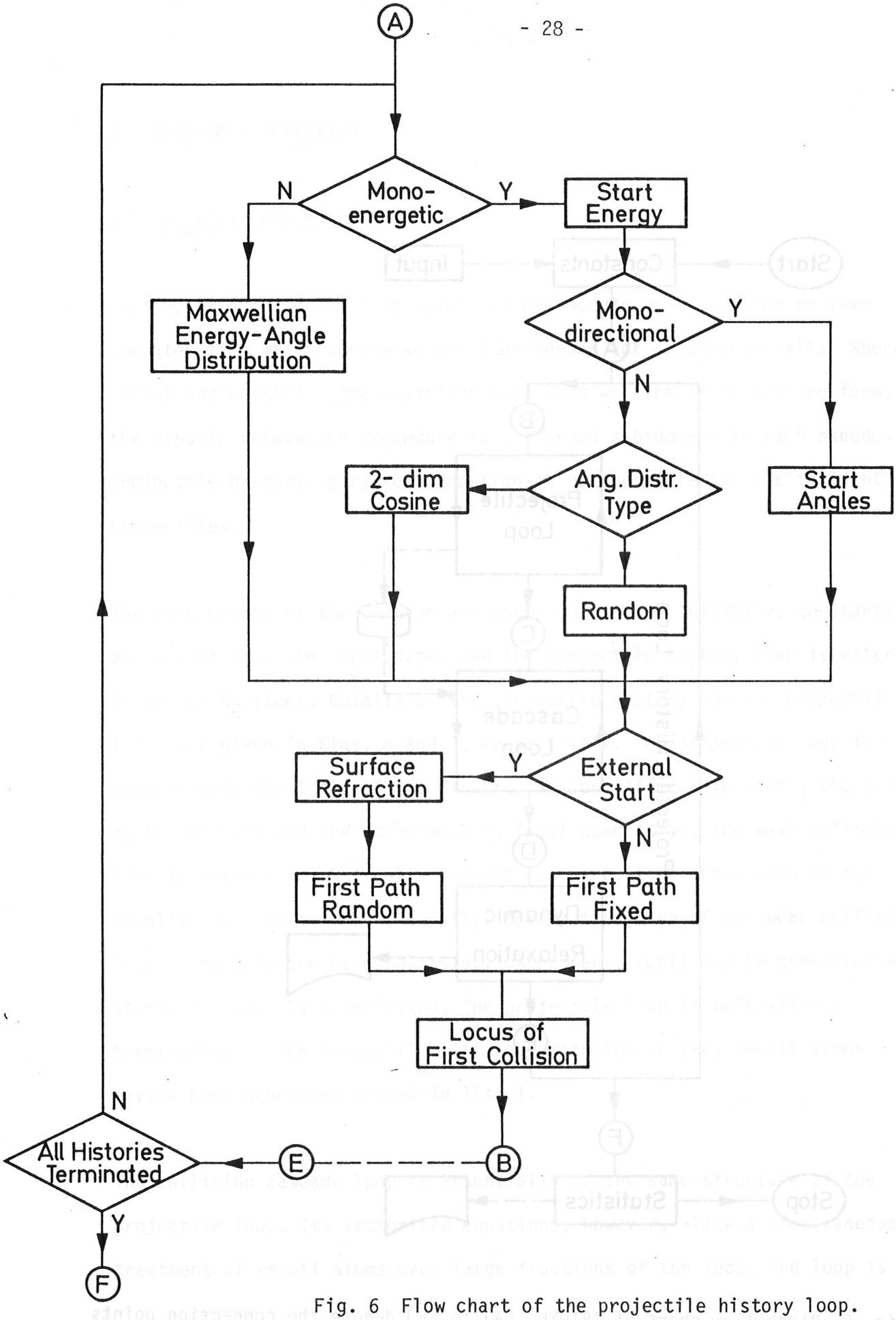

Fig. 6 Flow chart of the projectile history loop.

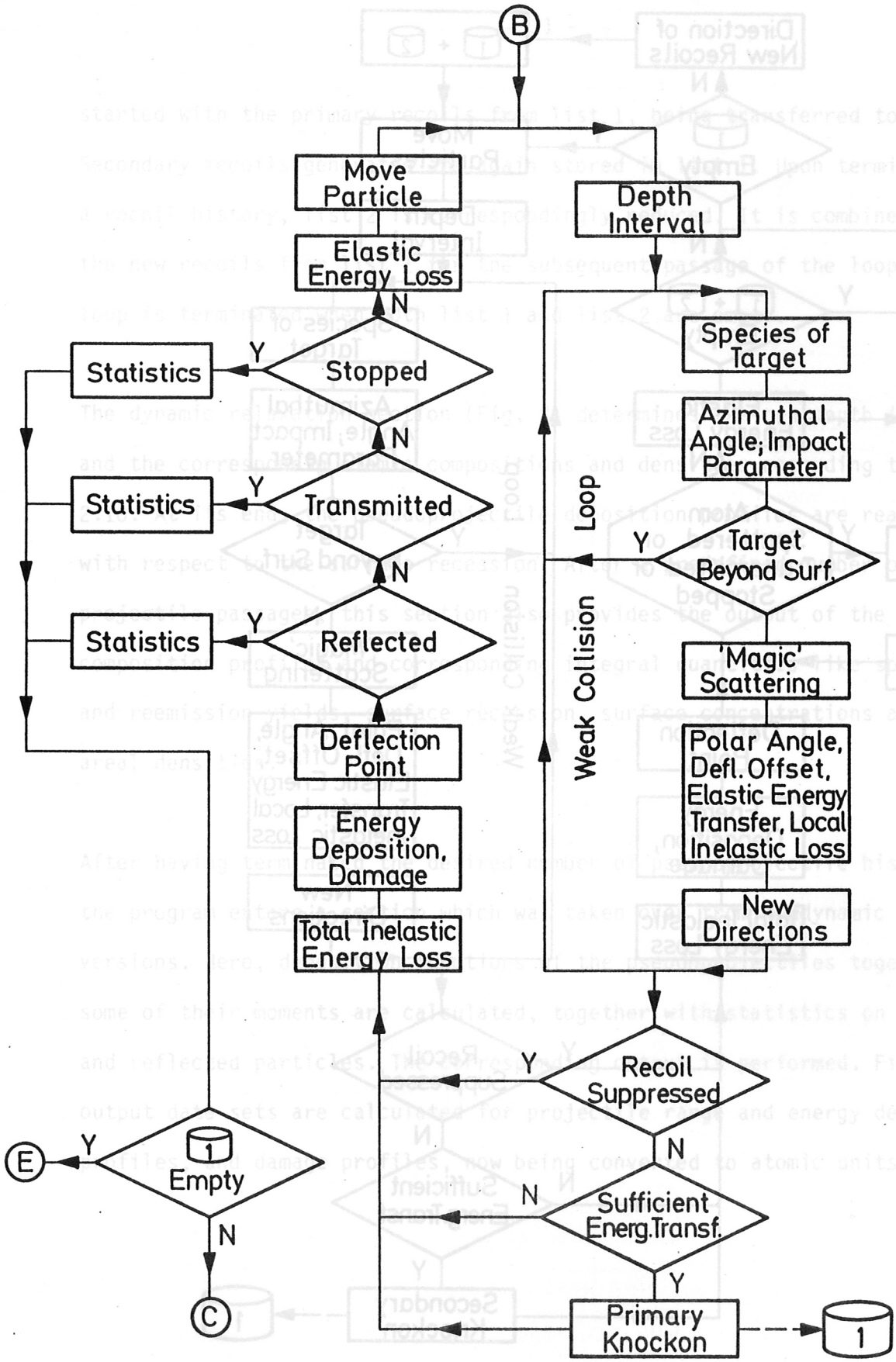

Fig. 7 Flow chart of the projectile loop. Primary knockon atoms generated during the projectile history are stored in list 1.

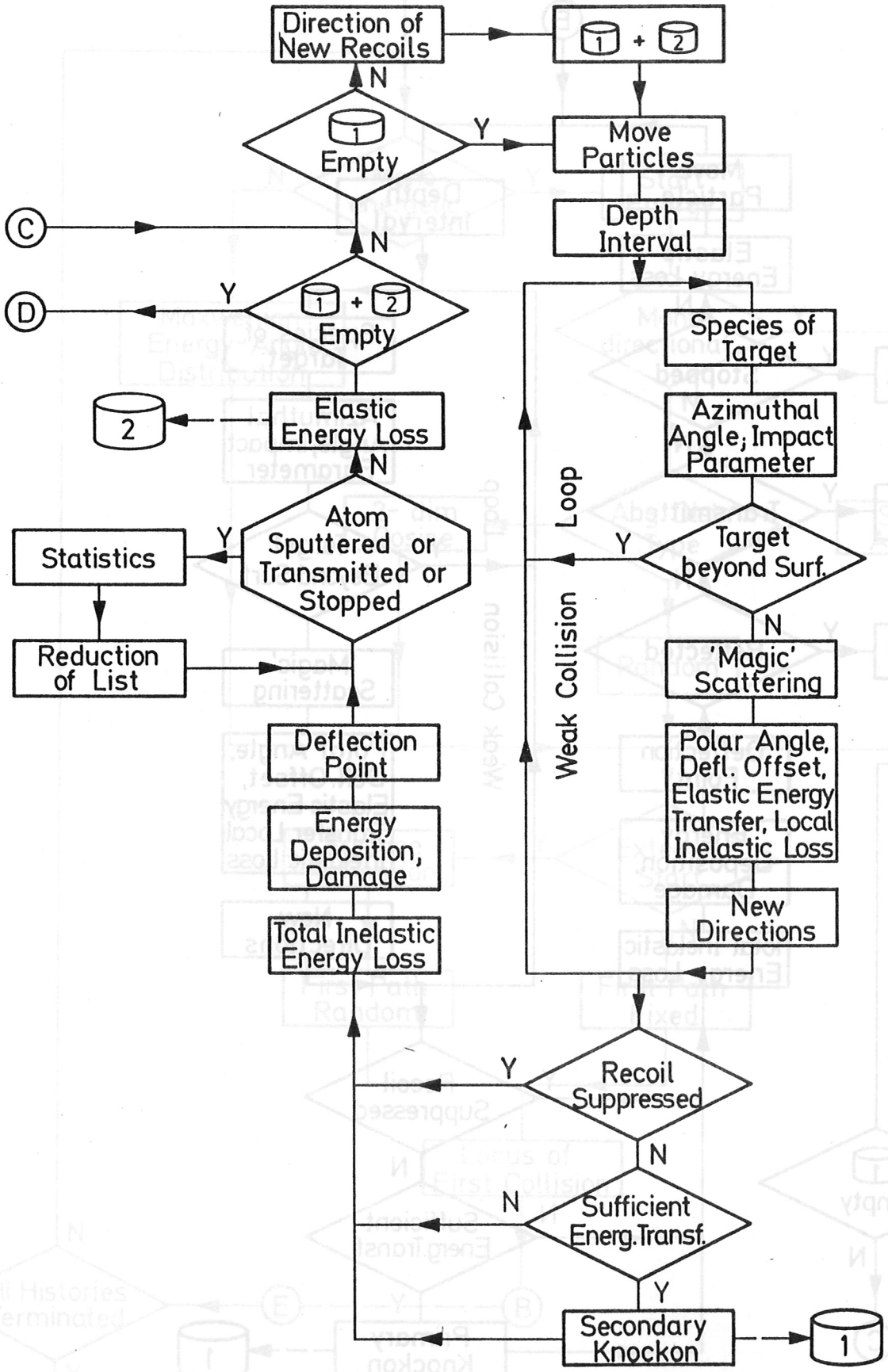

Fig. 8 Flow chart of the vectorized collision cascade loop. The loop starts with the list 1 primary recoils generated in the projectile loop.

started with the primary recoils from list 1, being transferred to list 2. Secondary recoils generated are again stored in list 1. Upon termination of a recoil history, list 2 is correspondingly reduced. It is combined with the new recoils from list 1 for the subsequent passage of the loop. The loop is terminated when both list 1 and list 2 are empty.

The dynamic relaxation section (Fig. 9) determines the new depth intervals and the correspondig atomic compositions and densities according to sect. 2.10. At its end, the pseudoprojectile deposition profiles are readjusted with respect to the surface recession. After a predefined number of pseudo-projectile passages, this section also provides the output of the dynamic composition profiles and corresponding integral quantities like sputtering and reemission yields, surface recession, surface concentrations and total areal densities.

After having terminated the desired number of pseudoprojectile histories, the program enters a section which was taken over from nondynamic TRIM versions. Here, deposition functions of the pseudoprojectiles together with some of their moments are calculated, together with statistics on sputtered and reflected particles. The corresponding output is performed. Finally, output data sets are calculated for projectile range and energy deposition profiles, and damage profiles, now being converted to atomic units.

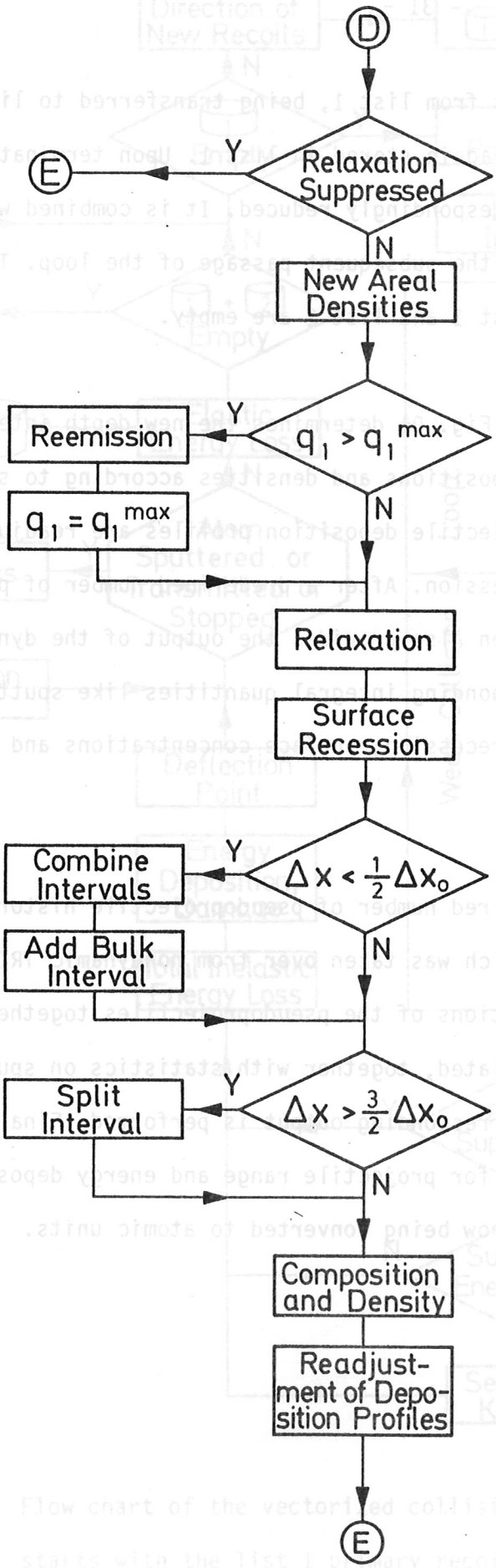

Fig. 9 Flow chart of the dynamic relaxation section.

3.2 Important and Frequently Occuring Variables

In the following, variables are listed which denote the important physical and computational variables. Some of them occur both as scalars in the projectile loop and as vectors in the recoil loop; the latter are denoted by an index IV and may occur both in list 1 and in list 2 as described above (index IL). Depth-dependent arrays are denoted by the index ID, the different constituents of the substance by IC. Note that the length unit is 10^{-10} m, and the energy unit eV.

General Constants

NH	- total number of projectile histories
FLC	- total projectile fluence
TT	- thickness of target substance
TTDYN	- max. depth of calculation
NQX	- number of depth intervals
DQX	- initial depth spacing
NCP	- number of different constituents

Properties of Constituents

ZZ(IC)	- atomic number
M(IC)	- atomic mass
SBE(IC)	- surface binding energy
BE(IC)	- bulk binding energy
ED(IC)	- displacement threshold energy
EF(IC)	- cutoff energy
CK(IC)	- correction factor for projectile electronic stopping
DNSO(IC)	- inverse of atomic volume

QU1MX - maximum atomic fraction of component 1 (projectile)

Depth-Dependent Arrays

XNO(ID) - depth grid

DNS(ID) - total atomic density

QUX(ID,IC) - atomic fraction

DEL(ID,IC) - change of number of pseudoparticles per layer; also areal density

Collisional Constants

$MU(IC_a,IC_b)$ - mass ratio

$EC(IC_a,IC_b)$ - elastic energy transfer factor

$A(IC_a,IC_b)$ - screening distance

$F(IC_a,IC_b)$ - energy scaling factor

$KL(IC_a,IC_b)$ - nonlocal electronic stopping factor

$KOR(IC_a,IC_b)$ - local electronic stopping factor

Collisional Variables

LM,LMV(IV,IL) - free path length

PMAX,PMAXV(IV) - maximum impact parameter

P,PR(IV) - impact parameter

B,BR(IV) - reduced impact parameter

EPS,EPSR(IV) - reduced energy

R,RRV(IV) - reduced distance of closest approach

PHI,PHIR(IV) - azimuthal deflection angle

C2,C2R(IV) - $\cos^2\frac{\theta}{2}$, θ polar deflection angle (center-of-mass system)

S2,S2R(IV) - $\sin^2\frac{\theta}{2}$

PSI,PSIR(IV,IL) - polar deflection angle (laboratory system)

DEN,T(IV) - elastic energy transfer

DENS,TS(IV) - weak and hard collision sum of elastic energy transfers

TAU,TAUR(IV) - deflection point offset

DEEOR,DEEROR - local inelastic energy transfer

DEES,DEERS(IV) - sum of local inelastic energy transfers

Moving Atom State and Flags

E,ER(IV,IL) - energy

COSX,CSXR(IV)
COSY,CSYR(IV)
COSZ,CSZR(IV) } - directional cosines

SINE,SNXR(IV) - directional sine, x-direction

X,XR(IV,IL)
Y,YR(IV,IL)
Z,ZR(IV,IL) - position (x normal to surface)

IXST(IV,IL) - depth interval of origin

EORC(IV,IL) - recoil start energy

INOUT(IV,IL) - flag for in-outgoing projectile at primary recoil generation

KO(IV,IL) - primary or secondary knockon flag

LABEL(IV,IL) - atomic species

L(IV,IL) - depth of origin

Deposition Profiles and Statistics

IRP(ID)	- range of pseudoprojectiles
IPL(ID)	- pathlength of pseudoprojectiles
NCLR(ID)	- nuclear energy deposition of pseudoprojectiles
PHON(ID)	- nuclear energy deposition of pseudoprojectiles below displacement threshold
DMGN(ID)	- nuclear energy deposition of pseudoprojectiles above displacement threshold
ION(ID)	- electronic energy deposition of pseudoprojectiles
FPKP(ID)	- number of Frenkel pairs generated by pseudoprojectiles (Kinchin-Pease model)
BBR(ID,IC)	- number of Frenkel pairs generated in all collisions (including cascades)
BBRTOT(IC)	- total number of Frenkel pairs of each component
IB	- total number of reflected pseudoprojectiles
EB	- total energy of reflected pseudoprojectiles
IBSP(IC)	- total number of sputtered pseudoatoms
EBSP(IC)	- total energy of sputtered pseudoatoms
IPREC	- total number of pseudoprojectiles being removed from range profile due to surface recession

3.3 Subprograms

The following subroutines are employed (those denoted by '(L)' are CRAY library routines):

CVMGT(A1,A2,LOG) - Function assigning A1 if LOG = .TRUE. or A2 if LOG = .FALSE. (L)

DIRCOS(COSX,COSY,COSZ,SINE,PSI,PHI) - Subroutine determining new directions from old ones and deflection angles

DIRCOSV(COSX,COSY,COSZ,SINE,PSI,PHI,N) - same as DIRCOS but for a vector of length N

ILLZ(N,TEST,1) - Function returning the number of leading .FALSE. components of a logical array TEST of length N (L)

ILSUM(N,TEST,1) - Function returning the total number of .TRUE. components of a logical array TEST of length N (L)

ISMAX(N,A,1) - Function returning the index of the maximum component of an array A of length N (L)

ISCHRFGE(N,A(1),1,X) - Function returning the first index where $A(I) \geq X$ for an array A of length N (L)

MXVELO(E,COSX,COSY,COSZ,SINE) - Subroutine for random generation of energy and directions from a Maxwellian distribution

RANF( ) - Random number generator (L)

RANSET(IRAND) - Random number initialization (L)

TRUNC(A,F,N,M) - Truncation of an array A of length N for trailing zeroes and multiplying it by the elements of an array F of length N. The new dimension is M.

4. Input and Output

Channel 5 is used for the program input data. A detailed explanation of the input data and their formats is given in the program listing.

A lineprinter output is assigned to channel 6. It contains a reproduction of the input data, followed by a list of collisional constants. In the subsequent lines, the following quantities are given:

SIMREC - the maximum number of recoils treated simultaneously (maximum length of list 2)

TOTREC - the total number of recoils treated

MAXCHA - the maximum relative change of areal density in any interval of depth (eq. (55))

Furthermore, the final surface recession and the areal density of reemitted projectiles are printed. The remaining printer output concerns pseudeparticle statistics, with numbers not having been converted to areal densities or local concentrations. It lists the total numbers of reflected (backscattered) pseudoprojectiles, transmitted ones and those which have been removed from the range profile due to surface recession. Sputtered pseudoparticles are classified with respect to their species and their history of origin (generated as primary or secondary knockon atoms by in- or outgoing projectiles). For the pseudoprojectiles, four moments of their range profile and two moments of their pathlength distribution are given, as well as integral energy transfer and damage quantities. These data are not corrected for surface recession. Finally, the corrected range, pathlength and energy deposition profiles of pseudoprojectiles are listed.

An additional output file on channel 17 contains the projectile deposition profiles and damage profiles in atomic units, normalized to the predefined total fluence. The output format is described in the program listing.

The result of the dynamic composition simulation is given on channel 7. At the beginning of the run and after termination of IDOUT pseudoprojectiles each, a four-record listing of integral quantities is provided. In addition, the composition and local density profiles are listed after termination of IQOUT pseudoprojectiles each. A detailed explanation is given in the program listing. The first two lines of the file contain a headline and the quantities NH, IDOUT, IQOUT, NCP, NQX, and the program name.

5. Test Run

As the test problem, the bombardment of Ta_2O_5 with 1 keV Ne ions at a fluence of $6 \cdot 10^{16}$ Ne/cm^2 has been chosen. The maximum Ne atomic fraction is arbitrarily set to 0.2. The runtime is about 18 min on the CRAY-XMP.

Fig. 10 shows the evolution of the composition profiles with the buildup of the Ne profile to saturation and the change of Ta profile due to preferential sputtering and collisional relocation. Correspondingly, the host surface composition and the amount of retained projectiles are displayed in Fig. 11 as functions of the implantation fluence. Finally, the partial sputtering yields of all components are given in Fig. 12.

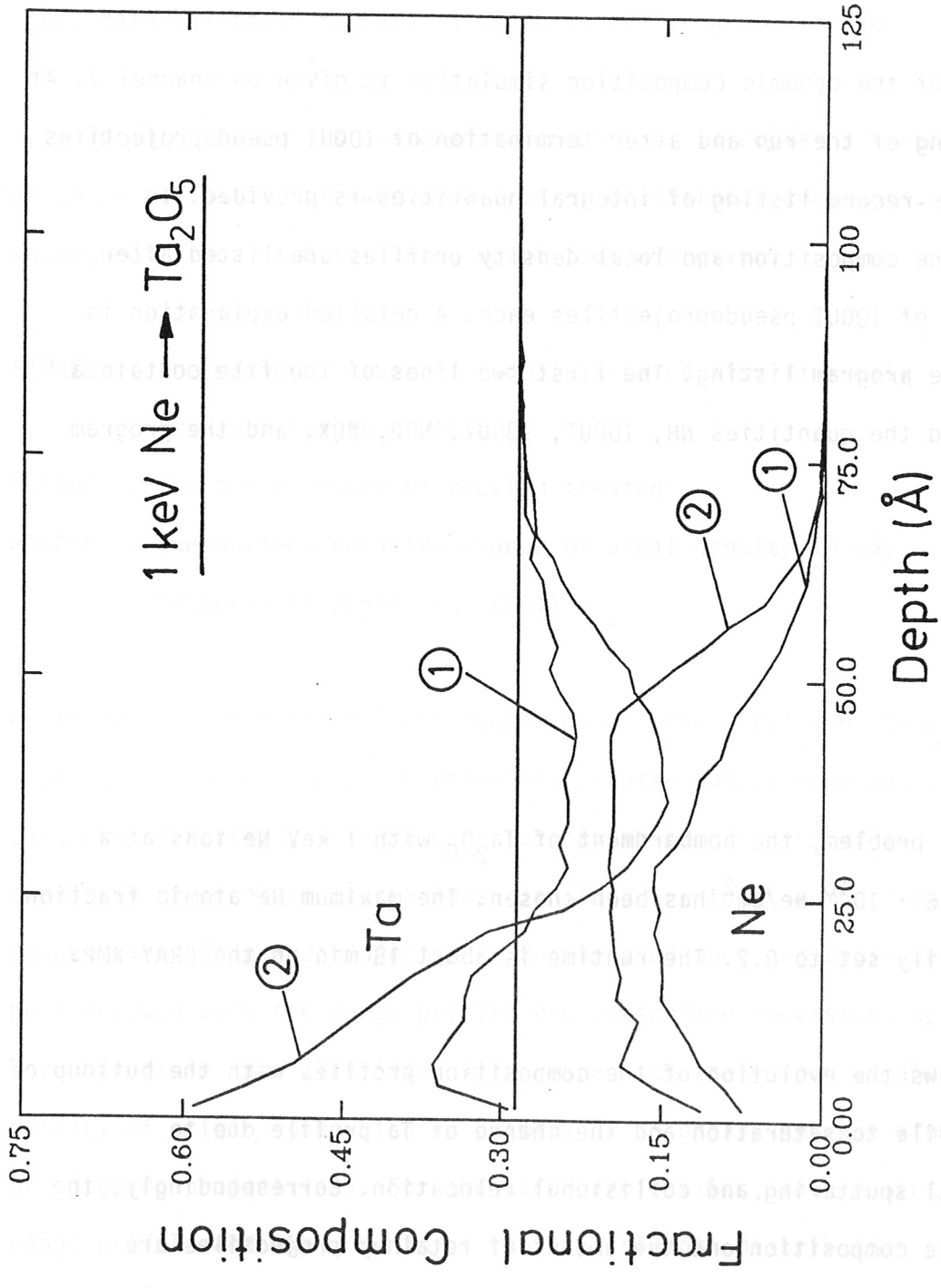

Fig.10 Initial, intermediate and stationary composition profile from the test run. The fluences are $6 \cdot 10^{15}$ Ne/cm^2 (denoted by ①), , and $6 \cdot 10^{16}$ Ne/cm^2 (②).

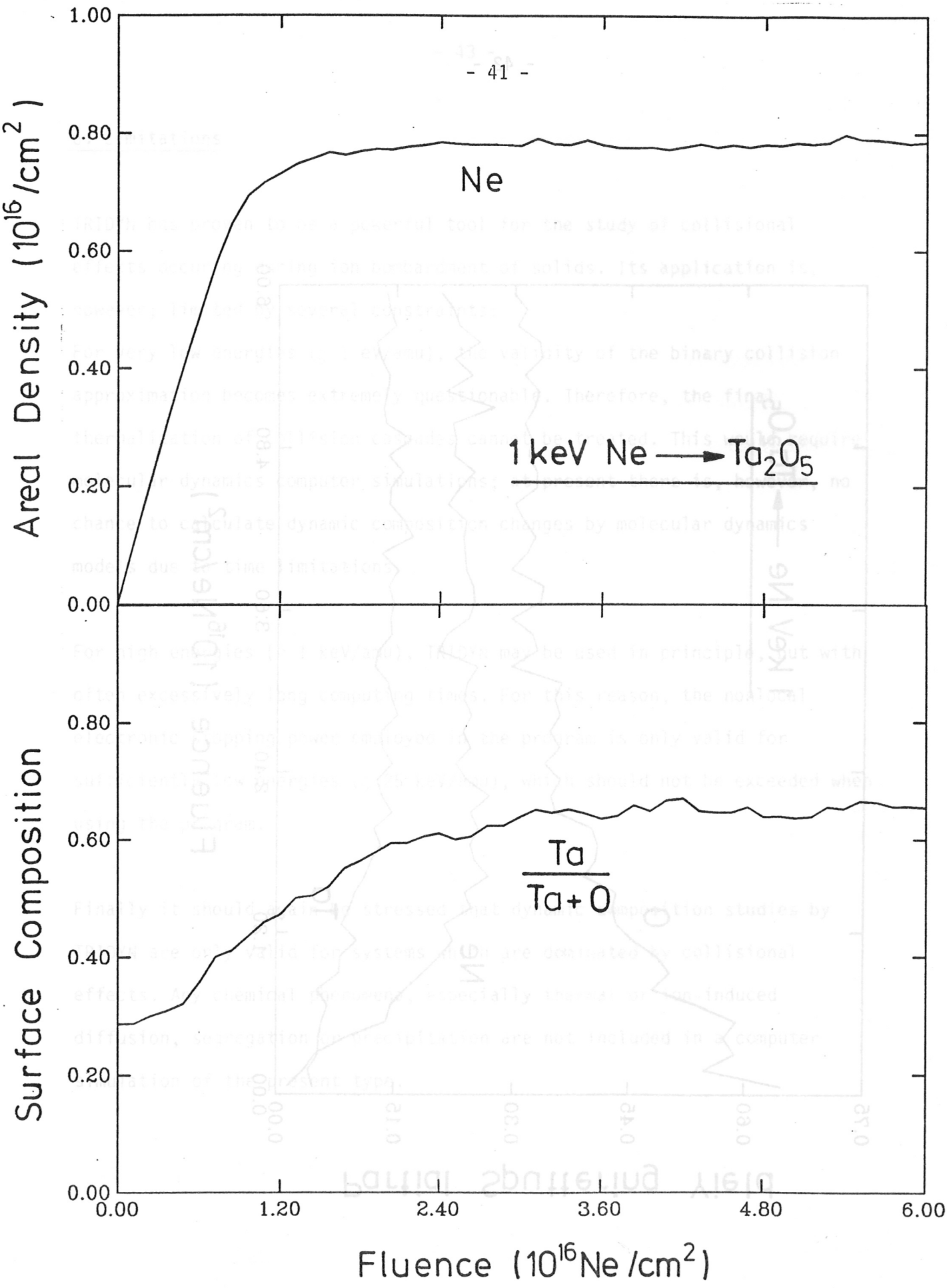

Fig.11 Surface composition of the host elements (integrated over 5 Å) and retained amount of implanted projectiles as function of fluence from the test run.

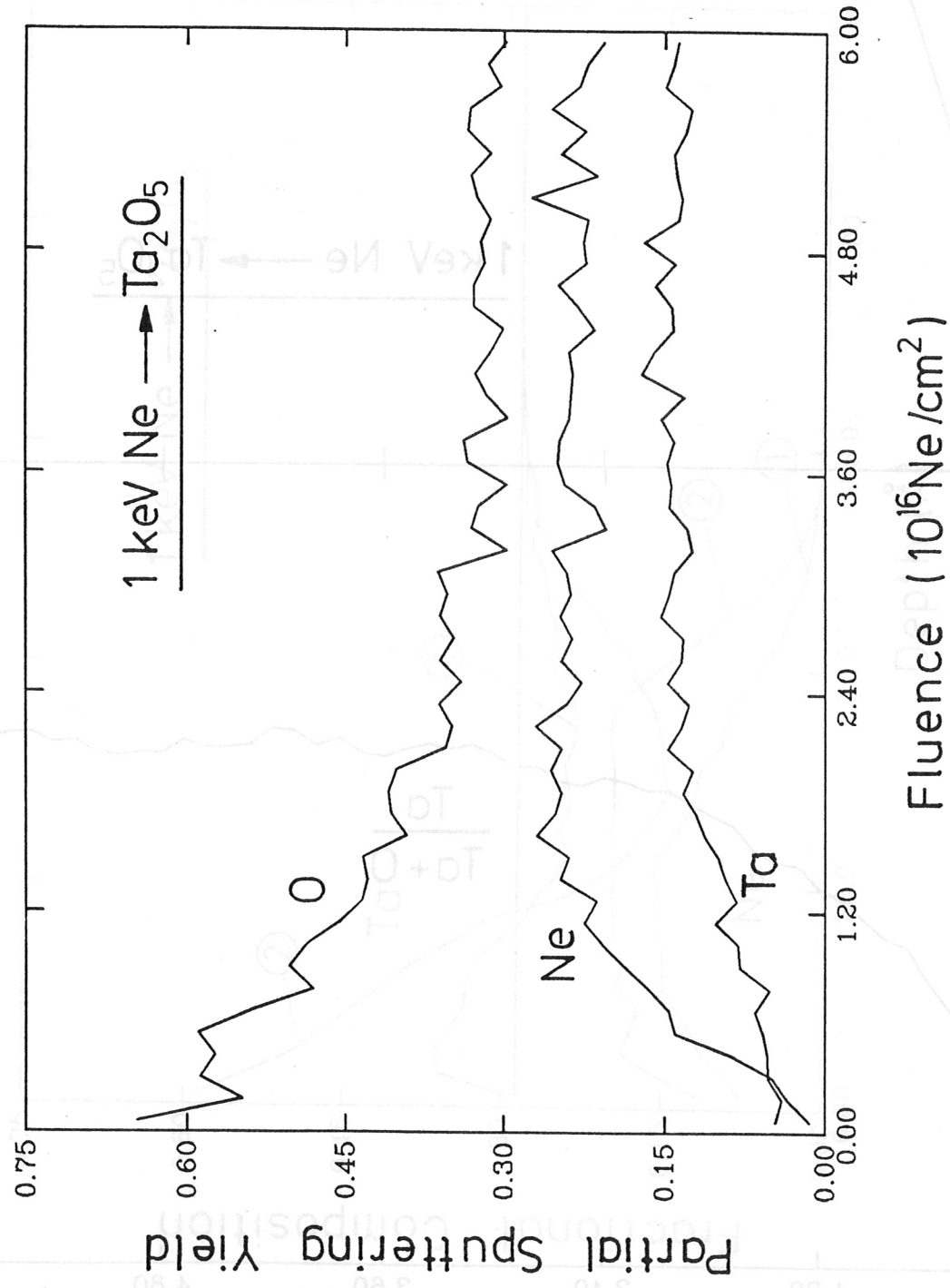

Fig.12 Partial sputtering yields as function of fluence from the test run.

6. Limitations

TRIDYN has proven to be a powerful tool for the study of collisional effects occuring during ion bombardment of solids. Its application is, however, limited by several constraints:
For very low energies ($\lesssim$ 1 eV/amu), the validity of the binary collision approximation becomes extremely questionable. Therefore, the final thermalization of collision cascades cannot be treated. This would require molecular dynamics computer simulations; at present there is, however, no chance to calculate dynamic composition changes by molecular dynamics models due to time limitations.

For high energies ($>$ 1 keV/amu), TRIDYN may be used in principle, but with often excessively long computing times. For this reason, the nonlocal electronic stopping power employed in the program is only valid for sufficiently low energies ($\lesssim$ 25 keV/amu), which should not be exceeded when using the program.

Finally it should again be stressed that dynamic composition studies by TRIDYN are only valid for systems which are dominated by collisional effects. Any chemical phenomena, especially thermal or ion-induced diffusion, segregation or precipitation are not included in a computer simulation of the present type.

References

/1/ D.P. Jackson and D.V. Morgan, Contemp. Phys. 14 (1974) 25

/2/ M.T. Robinson in: Sputtering by Particle Bombardment I, ed. R. Behrisch (Springer, Berlin-Heidelberg-New York 1981), p. 74

/3/ H.H. Andersen, Nucl. Instrum. Meth. B18 (1987) 321

/4/ W. Möller, Proc. of the NATO Adv. Study Institute "Materials Modification by High-Fluence Ion Beams", Viana do Castelo, Portugal 1987, ed. R. Kelly and F. da Silva, in press

/5/ J.P. Biersack and LG. Haggmark, Nucl. Instrum. Meth. 174 (1980) 257

/6/ J.F. Ziegler, J.P. Biersack, and U. Littmark, Stopping Power and Ranges of Ions in Matter, Vol. 1, ed. J.F. Ziegler (Pergamon, New York 1985)

/7/ J.P. Biersack and W. Eckstein, Appl. Phys. A34 (1974) 73

/8/ W. Eckstein and J.P. Biersack, Appl. Phys. A37 (1985) 95

/9/ W. Eckstein and W. Möller, Nucl. Instrum. Meth. B7/8 (1985) 727

/10/ W. Eckstein, unpublished results

/11/ W. Möller and W. Eckstein, Nucl. Instrum. Meth. B2 (1984) 814

/12/ W. Möller and W. Eckstein, Nucl. Instrum. Meth. B7/8 (1985) 645

/13/ B. Baretzky, W. Möller, and E. Taglauer, Nucl. Instrum. Meth. B18 (1987) 496

/14/ J. Lindhard, V. Nielsen, and M. Scharff, Kgl. Dan. Vid. Selsk. Mat.-Fys. Medd. 36, 10 (1968)

/15/ O.B. Firsov, Sov. Phys. JETP 6 (1958) 534

/16/ W.D. Wilson, L.G. Haggmark, and J.P. Biersack, Phys. Rev. B15 (1977) 2458

/17/ J. Lindhard and M. Scharff, Phys. Rev. 124 (1961) 128

/18/ O.S.Oen and M.T. Robinson, Nucl. Instrum. Meth. 132 (1976) 647

/19/ K.L. Merkle in: Radiation Damage in Metals, eds. N.L. Petersen and S.D. Harkness (American Soc. for Metals, Metal Parks, Ohio, 1976) p. 58

Appendix 1

Listing of the Program

```
C                                                                       0001
C                                                                       0002
C                                                                       0003
C     PROGRAM TRIDYN    VERSION 3.1                                     0004
C                                                                       0005
C     DYNAMIC COMPOSITION TRIM PROGRAM                                  0006
C                                                                       0007
C     W.MOELLER    IPP/OP    JAN.1988    CRAY-XMP                       0008
C                                                                       0009
C     VECTOR COMPUTER VERSION                                           0010
C                                                                       0011
C     BASED ON 1-COMPONENT SPUTTER-TRIM TRSPTC1 (J.P.BIERSACK AND       0012
C     W.ECKSTEIN), VECTORIZED SPUTTER-TRIM TRSPV1CT (W.ECKSTEIN),       0013
C     AND 5-COMPONENT DYNAMIC COMPOSITION TRIM PROGRAM TRIDYN,          0014
C     VERSION 2.3 (W.MOELLER)                                           0015
C                                                                       0016
C                                                                       0017
C                                                                       0018
C                                                                       0019
C     INPUT DATA (CHANNEL 5)                                            0020
C                                                                       0021
C     TITLE RECORD (FORMAT 16A4):                                       0022
C                                                                       0023
C     64-CHARACTER TITLE OF CALCULATION                                 0024
C                                                                       0025
C     1.RECORD (FORMAT I10,11I5):                                       0026
C                                                                       0027
C     NH          NUMBER OF PROJECTILE HISTORIES                        0028
C     IDOUT>0     DATA OUTPUT AFTER EACH IDOUT'TH HISTORY               0029
C          <=0     OUTPUT SUPPRESSED                                    0030
C     IQOUT       ADDITIONAL PROFILE OUTPUT AFTER EACH IQOUT'TH HISTORY 0031
C     NCP         NUMBER OF TARGET COMPONENTS (INCLUDING PROJECTILE)    0032
C     IDREL>0     SUPPRESSION OF DYNAMIC RELAXATION                     0033
C          <0     SUPPRESSION OF DYNAMIC RELAXATION AND CASCADES        0034
C     IQ0<0       INITIAL COMPOSITION VARIABLE ACCORDING TO VECTOR INPUT 0035
C       =0        INITIAL COMPOSITION HOMOGENEOUS                       0036
C       >0        INITIAL COMPOSITION ACCORDING TO LAYER INPUT          0037
C     IRC0<0      SUBTHRESHOLD RECOIL ATOMS FREE                        0038
C       >=0       SUBTHRESHOLD RECOIL ATOMS BOUND                       0039
C     IRAND       INITIAL RANDOM NUMBER                                 0040
C     JSP1,JSP2   SUPPRESSION OF RECOILS OF TYPE JSP1,...,JSP2          0041
C       JSP1=0    ALL RECOILS TRACED                                    0042
C     JFRP        FRENKEL PAIR GENERATION FOR COMPONENTS JFRP,...,NCP    0043
C                 (DEFAULT: NCP)                                        0044
C     JNRM        NORMALIZATION OF PUNCH1 OUTPUT TO PARTIAL DENSITY OF  0045
C                 COMPONENTS JNRM,...,NCP (DEFAULT: NCP)                0046
C                                                                       0047
C     2.RECORD (FORMAT E10.3,3F10.2,4I5,F10.2):                         0048
```

```
C                                                                       0049
C     FLC         IMPLANTED FLUENCE (1E16/CM2) FOR COMPLETE RUN         0050
C     E0          PROJECTILE ENERGY (EV)                                0051
C       <0:       MAXWELLIAN ENERGY-ANGLE DISTRIBUTION WITH KT=-E0      0052
C                 (ALPHA IS IGNORED IN THIS CASE)                       0053
C     X0          PROJECTILE START DEPTH (A)                            0054
C     ALPHA       ANGLE OF INCIDENCE (DEG) WITH RESPECT TO NORMAL       0055
C        = -1:    RANDOM ANGLE DISTRIBUTION                             0056
C        = -2:    2-DIMENSIONAL COSINE DISTRIBUTION                     0057
C     INEL =1     INELASTIC PROJECTILE-TARGET INTERACTION NONLOCAL      0058
C          2                                              LOCAL         0059
C          3                                              EQUIPART.     0060
C     INELR       INELASTIC TARGET-TARGET INTERACTION (SEE INEL)        0061
C     IWC         MAX. ORDER OF WEAK PROJECTILE-TARGET COLLISIONS       0062
C     IWCR        MAX. ORDER OF WEAK TARGET-TARGET COLLISIONS           0063
C     SHTH        SHEATH POTENTIAL (EV) IN CASE OF MAXWELLIAN           0064
C                 ENERGY DISTRIBUTION                                   0065
C                                                                       0066
C     3.RECORD (FORMAT 2E10.3,I5,E10.3,3I5):                            0067
C                                                                       0068
C     TT          TARGET THICKNESS (A)                                  0069
C     TTDYN       DEPTH RANGE (A) FOR DYNAMIC RELAXATION                0070
C     NQX         NR. OF DEPTH INTERVALS WITHIN TTDYN                   0071
C     DSF         AVERAGING DEPTH (A) FOR SURFACE COMPOSITION           0072
C     IQXN,IQXX   LIMITING DEPTH INTERVALS FOR PROFILE OUTPUT           0073
C                 (IQXN = 0: ALL INTERVALS)                             0074
C     IMCP        COMPONENT FOR WHICH MOMENTS SHALL BE CALCULATED       0075
C                 (= 0: NO MOMENT CALCULATION)                          0076
C                                                                       0077
C     FOLLOWING RECORDS (FORMAT 6F7.2,F7.3,E10.3,F5.2)                  0078
C     (ONE FOR EACH COMPONENT, FIRST FOR PROJECTILE):                   0079
C                                                                       0080
C     ZZ          ATOMIC NUMBER                                         0081
C     M           ATOMIC MASS                                           0082
C     SBE         SURFACE BINDING ENERGY (EV)                           0083
C     BE          BINDING ENERGY (EV)                                   0084
C     ED          DISPLACEMENT ENERGY (EV)                              0085
C     EF          CUTOFF ENERGY (EV)                                    0086
C     QU          MAXIMUM ALLOWABLE ATOMIC FRACTION (PROJECTILE)        0087
C                 INITIAL ATOMIC FRACTION (TARGET;IQ0=0)                0088
C     DNS0        ATOMIC DENSITY OF PURE COMPONENT (A-3)                0089
C     CK          PROJECTILE ELECTRONIC STOPPING CORRECTION FACTOR      0090
C                                                                       0091
C     ADDITIONAL RECORDS IN CASE OF IQ0<0 (FORMAT 10F6.3)               0092
C     (VECTOR INPUT):                                                   0093
C                                                                       0094
C     ONE SET OF NQX LOCAL ATOMIC FRACTIONS FOR EACH TARGET             0095
C     COMPONENT (EXCLUDING PROJECTILE)                                  0096
```

```
C                                                                       0097
C     ADDITIONAL RECORDS IN CASE OF IQO>0 (FORMAT E10.3,4F6.3)          0098
C     (LAYER INPUT):                                                    0099
C                                                                       0100
C     ARBITRARY NR. OF RECORDS CONTAINING EACH                          0101
C     DMAX          MAX.DEPTH (A)                                       0102
C     QU(2...NCP)   ATOMIC FRACTIONS (TARGET)                           0103
C                                                                       0104
C                                                                       0105
C                                                                       0106
C     OUTPUT ON CHANNEL 7:                                              0107
C                                                                       0108
C                                                                       0109
C     QUANTITIES AS FUNCTION OF FLUENCE:                                0110
C                                                                       0111
C     1.RECORD: (FORMAT 7E11.4)                                         0112
C     FLUC      PROJECTILE FLUENCE (A-2)                                0113
C     SRRC      SURFACE RECESSION (A)                                   0114
C     REEM1     REEMITTED PROJECTILE FLUENCE (A-2)                      0115
C     CSF       AVERAGED ATOMIC SURFACE CONCENTRATIONS (2...5)          0116
C                                                                       0117
C     2.RECORD: (FORMAT 5E11.4)                                         0118
C     FLIB      MOMENTS OF DEPTH DISTRIBUTION OF COMPONENT IMCP         0119
C               (INTEGRAL(A-2), MEAN DEPTH(A), STD.DEV.(A),             0120
C               SKEWNESS, KURTOSIS)                                     0121
C                                                                       0122
C     3.RECORD: (FORMAT 5E11.4)                                         0123
C     SPYD      SPUTTERING YIELDS (1...5)                               0124
C                                                                       0125
C     4.RECORD: (FORMAT 5E11.4)                                         0126
C     ARD       AREAL DENSITIES (1...5)                                 0127
C                                                                       0128
C                                                                       0129
C     PROFILES:                                                         0130
C                                                                       0131
C     ONE RECORD FOR EACH DEPTH INTERVAL: (FORMAT 2X,2E11.4,5F6.3)      0132
C     XXX       CENTRAL DEPTH (A)                                       0133
C     DNS       TOTAL ATOMIC DENSITY (A-3)                              0134
C     QUX       RELATIVE ATOMIC FRACTIONS (1...NCP)                     0135
C                                                                       0136
C                                                                       0137
C     OUTPUT ON CHANNEL 17:                                             0138
C                                                                       0139
C     DEPOSITION PROFILES, NORMALIZED TO TARGET ATOM DENSITY            0140
C     (CPTS.#JNRM...NCP); IN CASE OF DYNAMIC MODE, THE FINAL            0141
C     DENSITY IS TAKEN                                                  0142
C                                                                       0143
C     HEADLINE AND DATA PAIRS (FORMAT 2E15.6) IN CONSECUTIVE ORDER FOR  0144
```

```
4915  FORMAT(E10.3,4F6.3)                                              0241
      GOTO 4940                                                        0242
4930  DO 4920 JP=2,NCP                                                 0243
4920  QUX(MM,JP) = QU(JP)                                              0244
      GOTO 4040                                                        0245
4020  DO 4050 JP=2,NCP                                                 0246
      QUINP(JP) = QU(JP)                                               0247
      DO 4050 MM=1,NQX                                                 0248
4050  QUX(MM,JP) = QU(JP)                                              0249
4040  QUINP(1) = QU(1)                                                 0250
      IF(IQ0.EQ.0) GOTO 5050                                           0251
      DO 5060 JP=2,NCP                                                 0252
      DO 5060 MM=1,NQX                                                 0253
5060  QUX0(MM,JP) = QUX(MM,JP)                                         0254
5050  QU1MX = QU(1)                                                    0255
      DO 4100 MM=1,NQX                                                 0256
      XXX(MM) = (MM-.5)*DQX                                            0257
      XNO(MM) = (MM-1)*DQX                                             0258
      SUM = 0.                                                         0259
      DNS(MM) = 0.                                                     0260
      DO 4070 JP=2,NCP                                                 0261
      SUM = SUM+QUX(MM,JP)                                             0262
4070  DNS(MM) = DNS(MM)+QUX(MM,JP)/DNS0(JP)                            0263
      DNS(MM) = 1./(DNS(MM)+(1.-SUM)/DNS0(1))                          0264
4100  QUX(MM,1) = 1.-SUM                                               0265
      XNO(NQX+1) = TTDYN                                               0266
      SUM = 0.                                                         0267
      DO 4230 JP=2,NCP                                                 0268
      QUXB(JP) = QUX(NQX,JP)                                           0269
4230  SUM = SUM+QUXB(JP)                                               0270
      QUXB(1) = 1.-SUM                                                 0271
      DNSB = DNS(NQX)                                                  0272
C                                                                      0273
C     INITIAL OUTPUT OF INTEGRAL DATA                                  0274
C                                                                      0275
      IF(IQXN.LE.0) GOTO 4870                                          0276
      NQOUT = IQXX-IQXN+1                                              0277
      GOTO 4880                                                        0278
4870  IQXN = 1                                                         0279
      IQXX = NQX                                                       0280
      NQOUT = NQX                                                      0281
4880  SRRC = 0.                                                        0282
      FLUC = 0.                                                        0283
      IF(IDOUT.LE.0.OR.IDREL.NE.0) GOTO 2220                           0284
      AMM = 0.                                                         0285
      DO 4060 JP=1,NCP                                                 0286
      CSF(JP) = 0.                                                     0287
4060  ARD(JP) = 0.                                                     0288
```

```
      DO 4730 MM=1,NQX                                                  0289
      SUM = 0.                                                          0290
      DO 4740 JP=2,NCP                                                  0291
4740  SUM = SUM+QUX(MM,JP)                                              0292
      DO 4080 JP=1,NCP                                                  0293
4080  ARD(JP) = ARD(JP)+QUX(MM,JP)*DQX*DNS(MM)                          0294
      IF(XNO(MM).GE.DSF) GOTO 4730                                      0295
      DAM = 1.                                                          0296
      IF(XNO(MM+1).LE.DSF) GOTO 4750                                    0297
      DAM = (DSF-XNO(MM))/DQX                                           0298
4750  AMM = AMM+DAM                                                     0299
      DO 4110 JP = 2,NCP                                                0300
4110  CSF(JP) = CSF(JP)+QUX(MM,JP)/SUM*DAM                             0301
4730  CONTINUE                                                          0302
      IF(AMM.LE.0) GOTO 4440                                            0303
      DO 4450 JP=2,NCP                                                  0304
4450  CSF(JP) = CSF(JP)/AMM                                             0305
4440  IF(IMCP.LE.0) GOTO 4960                                           0306
      DO 4970 MM=1,NQX                                                  0307
      FLIB(1) = FLIB(1)+QUX(MM,IMCP)*DQX*DNS(MM)                        0308
4970  FLIB(2) = FLIB(2)+QUX(MM,IMCP)*XXX(MM)*DQX*DNS(MM)                0309
      IF(FLIB(1).LE.0.) GOTO 4960                                       0310
      FLIB(2) = FLIB(2)/FLIB(1)                                         0311
      DO 4980 MM=1,NQX                                                  0312
      SUM = QUX(MM,IMCP)*DQX*DNS(MM)*(XXX(MM)-FLIB(2))**2               0313
      FLIB(3) = FLIB(3)+SUM                                             0314
      FLIB(4) = FLIB(4)+SUM*(XXX(MM)-FLIB(2))                           0315
4980  FLIB(5) = FLIB(5)+SUM*(XXX(MM)-FLIB(2))**2                        0316
      IF(FLIB(3).LT.1.E-10*FLIB(2)) GOTO 5030                           0317
      FLIB(3) = SQRT(FLIB(3)/FLIB(1))                                   0318
      FLIB(4) = FLIB(4)/FLIB(3)**3                                      0319
      FLIB(5) = FLIB(5)/FLIB(3)**4-3.                                   0320
      GOTO 4960                                                         0321
5030  FLIB(3) = 0.                                                      0322
      FLIB(4) = 0.                                                      0323
      FLIB(5) = 0.                                                      0324
4960  WRITE(IO1,2025)                                                   0325
2025  FORMAT('DYNAMIC COMPOSITION TRIM PROGRAM')                        0326
      WRITE(IO1,2055)NH,IDOUT,IQOUT,NCP,NQOUT                           0327
2055  FORMAT(5I10,5X,'TRIDYN31')                                        0328
      WRITE(IO1,2035)FLUC,SRRC,FLUC,(CSF(JP),JP=2,NCPM),FLIB,SPYD,ARD   0329
2035  FORMAT(7E11.4,3(/5E11.4))                                         0330
C                                                                       0331
C     INITIAL OUTPUT OF PROFILES                                        0332
C                                                                       0333
      IF(IQOUT.LE.0) GOTO 2220                                          0334
      DO 4090 MM=IQXN,IQXX                                              0335
4090  WRITE(IO1,2045)XXX(MM),DNS(MM),(QUX(MM,JP),JP=1,NCP)              0336
```

```
2045  FORMAT(2X,2E11.4,5F6.3)                                          0337
C                                                                      0338
C     SET UNIVERSAL AND INITIAL QUANTITIES                             0339
C                                                                      0340
2220  CHAMAX = 0.                                                      0341
      SRRCN = 0.                                                       0342
      REEM1 = 0.                                                       0343
      IF(QU1MX.LT.0..OR.QU1MX.GT.1.) QU1MX = 1.                        0344
      IF(JNRM.LE.0.OR.JNRM.GT.NCP) JNRM = NCP                          0345
      IF(JFRP.LE.0.OR.JFRP.GT.NCP) JFRP = NCP                          0346
      JSP1I = JSP1                                                     0347
      IF(JSP1.LE.0) JSP2 = 0                                           0348
      IF(JSP1.LE.0) JSP1 = NCP+1                                       0349
      IPREC = 0                                                        0350
      KK1 = 4-IWC                                                      0351
      KK2 = 4-IWCR                                                     0352
      ALPHA = CVMGT(.001,ALPHA,ALPHA.EQ.0.)                            0353
      ALPHA = CVMGT(179.999,ALPHA,ALPHA.EQ.180.)                       0354
      ALFA=ALPHA/57.295779                                             0355
      DO 102 I=1,NCP                                                   0356
      DO 102 J=1,NCP                                                   0357
      MU(I,J)=M(I)/M(J)                                                0358
      EC(I,J)=4.*MU(I,J)/(1.+MU(I,J))**2                               0359
      A(I,J)=.529*.8853/(ZZ(I)**.5+ZZ(J)**.5)**(2./3.)                 0360
      F(I,J)=A(I,J)*M(J)/(ZZ(I)*ZZ(J)*14.4*(M(I)+M(J)))                0361
      KL(I,J)=1.212*ZZ(I)**(7./6.)*ZZ(J)/((ZZ(I)**(2./3.)+ZZ(J)**(2./3.  0362
     ?))**1.5*SQRT(M(I)))                                              0363
      KOR(I,J)=.03892*KL(I,J)/(PI*A(I,J)**2)                           0364
  102 CONTINUE                                                         0365
      Z1=ZZ(1)                                                         0366
      M1=M(1)                                                          0367
      DO 104 J=1,NCP                                                   0368
      K(J)=CK(J)*KL(1,J)                                               0369
  104 EPS0(J)=ABS(E0)*F(1,J)                                           0370
      ANFANG = RANSET(IRAND)                                           0371
      IF(E0.GE.0.) GOTO 105                                            0372
      TI = -E0                                                         0373
      ZARG = SQRT(TI/2./M1)                                            0374
      VELC = SHTH/M1                                                   0375
C                                                                      0376
C     PROJECTILE HISTORY LOOP                                          0377
C                                                                      0378
105   IH = 0                                                           0379
10    IH = IH+1                                                        0380
      IF(IH.GT.NH) GOTO 18                                             0381
C                                                                      0382
C     PROJECTILE START                                                 0383
C                                                                      0384
```

```
      DO 2070 JP=1,NCP                                                  0385
      DO 2070 MM=1,NQX                                                  0386
2070  DEL(MM,JP) = 0.                                                   0387
      SU = DNS(1)**(-1./3.)/SQ2PI*2.                                    0388
      XC = CVMGT(X0,-SU,X0.GT.0.)                                       0389
      IC=0                                                              0390
      IC1=0                                                             0391
      PL = 0.                                                           0392
      NREC1=1                                                           0393
      IF(E0.GE.0.) GOTO 106                                             0394
C                                                                       0395
C     MAXWELLIAN ENERGY-ANGLE DISTRIBUTION                              0396
C                                                                       0397
      CALL MXVELO(E,COSX,COSY,COSZ,SINE)                                0398
      GOTO 107                                                          0399
106   E = E0                                                            0400
      IF(ALPHA.LT.0.) GOTO 108                                          0401
      COSX = COS(ALFA)                                                  0402
      COSY = SIN(ALFA)                                                  0403
      COSZ = 0.                                                         0404
      SINE = COSY                                                       0405
      GOTO 107                                                          0406
108   IF(ALPHA.NE.-2.) GOTO 109                                         0407
C                                                                       0408
C     2-DIM COSINE ANGLE DISTRIBUTION                                   0409
C                                                                       0410
      RPHI = RANF()                                                     0411
      RTHETA = RANF()                                                   0412
      COSX = SQRT(RTHETA)                                               0413
      SINE = SQRT(1.-RTHETA)                                            0414
      COSY = SINE*COS(2.*PI*RPHI)                                       0415
      COSZ = SINE*SIN(2.*PI*RPHI)                                       0416
      GOTO 107                                                          0417
C                                                                       0418
C     RANDOM ANGLE DISTRIBUTION                                         0419
C                                                                       0420
109   RPHI = RANF()                                                     0421
      RTHETA = RANF()                                                   0422
      COSX = 1.-RTHETA                                                  0423
      IF(X0.GT.0.) COSX = COSX-RTHETA                                   0424
      SINE = SQRT(1.-COSX**2)                                           0425
      COSY = SINE*COS(2.*PI*RPHI)                                       0426
      COSZ = SINE*SIN(2.*PI*RPHI)                                       0427
C                                                                       0428
C     EXTERNAL START                                                    0429
C                                                                       0430
107   IF(X0.GT.0.) GOTO 110                                             0431
      SINA = SINE                                                       0432
```

```
      COSX = SQRT((E*COSX**2+SBE(1))/(E+SBE(1)))                        0433
      SINE = SQRT(1.-COSX**2)                                           0434
      COSY = COSY*SINE/SINA                                             0435
      COSZ = COSZ*SINE/SINA                                             0436
      E = E+SBE(1)                                                      0437
C                                                                       0438
C     PROJECTILE START INTERVAL                                         0439
C                                                                       0440
110   IX = ISRCHFGE(NQX+1,XNO(1),1,XC)-1                                0441
      IX = MAXO(IX,1)                                                   0442
      IF(IX.LE.NQX) GOTO 2130                                           0443
2140  WRITE(IO,2075)IH,IC,X,E,IERR,XR(IERR,2),ER(IERR,2)                0444
2075  FORMAT(5X,'IH=',I7,'   IC=',I5,'    X=',E11.4,'    E=',E11.4/    0445
     C18X,'IR=',I5,'   XR=',E11.4,'   ER=',E11.4)                       0446
      WRITE(IO,2095)                                                    0447
2095  FORMAT(5X,'PARTICLE RANGE EXCEEDS DYNAMIC COMPOSITION RANGE'/     0448
     C5X,'----- INCREASE TTDYN')                                        0449
      STOP                                                              0450
2130  LM = DNS(IX)**(-1./3.)                                            0451
C                                                                       0452
C     LOCUS OF FIRST COLLISION                                          0453
C                                                                       0454
      RA = 1.                                                           0455
      IF(XO.LE.O.) RA = RANF()                                          0456
      X = XC+LM*COSX*RA                                                 0457
      Y = LM*COSY*RA                                                    0458
      Z = LM*COSZ*RA                                                    0459
      GOTO 2                                                            0460
C                                                                       0461
C     COLLISION LOOP                                                    0462
C                                                                       0463
C     MOVE PROJECTILES                                                  0464
C                                                                       0465
    1 X=X+LM*COSX                                                       0466
      Y=Y+LM*COSY                                                       0467
      Z=Z+LM*COSZ                                                       0468
C                                                                       0469
C     STORE DIRECTIONS                                                  0470
C                                                                       0471
2     CX = COSX                                                         0472
      CY = COSY                                                         0473
      CZ = COSZ                                                         0474
      SX = SINE                                                         0475
      EX1 = 0.                                                          0476
      DEES = 0.                                                         0477
      DENS = 0.                                                         0478
C                                                                       0479
C     DEPTH INTERVAL OF COLLISION                                       0480
```

```
C                                                                       0481
      IX = ISRCHFGE(NQX+1,XNO(1),1,X)-1                                 0482
      IX = MAXO(IX,1)                                                   0483
      IF(IX.GT.NQX) GOTO 2140                                           0484
C                                                                       0485
C     COLLISION PARAMETERS                                              0486
C                                                                       0487
      LM = DNS(IX)**(-1./3.)                                            0488
      ASIG = LM*DNS(IX)                                                 0489
      PMAX = LM/SQ2PI                                                   0490
      K23 = 0.                                                          0491
      DO 4120 JP=1,NCP                                                  0492
4120  K23 = K23+K(JP)*QUX(IX,JP)                                        0493
C                                                                       0494
C     WEAK COLLISION LOOP                                               0495
C                                                                       0496
      DO 245 KK=KK1,4                                                   0497
      TAU = 0.                                                          0498
      DEN = 0.                                                          0499
C                                                                       0500
C     SPECIES OF COLLISION PARTNER                                      0501
C                                                                       0502
      RR = RANF()                                                       0503
      SUM = 1.-QUX(IX,1)                                                0504
      DO 4130 J=1,NCP-1                                                 0505
      IF(RR.GT.SUM) GOTO 4140                                           0506
4130  SUM = SUM-QUX(IX,J+1)                                             0507
      J = NCP                                                           0508
4140  LABEL(NREC1,1)=J                                                  0509
      JPKO = J                                                          0510
C                                                                       0511
C     AZIMUTHAL ANGLE, IMPACT PARAMETER, REDUCED ENERGY                 0512
C                                                                       0513
      PHI=2.*PI*RANF()                                                  0514
      P=PMAX*SQRT(RANF()+4.-KK)                                         0515
      EPS=E*F(1,J)                                                      0516
      IF(KK.LT.4.AND.EPS.GT.1.E-2) GOTO 245                             0517
C                                                                       0518
C     POSITION OF TARGET ATOM                                           0519
C                                                                       0520
      X1=X-P*COS(PHI)*SX                                                0521
      IF(X1.LT.0.) GOTO 245                                             0522
C                                                                       0523
C      MAGIC (DETERMINATION OF SCATTERING ANGLE : ION - TARGET ATOM :   0524
C      KR-C POTENTIAL)                                                  0525
C                                                                       0526
      B = P/A(1,J)                                                      0527
      R=B                                                               0528
```

```
      RR=ALOG(.35/(EPS*B))*3.33333333333                                0529
      IF (RR.LT.B) GO TO 121                                            0530
      RR=ALOG(.35/(EPS*RR))*3.3333333333                                0531
      IF (RR.LT.B) GO TO 121                                            0532
      R=RR                                                              0533
  121 EX1=EXP(-.279*R)                                                  0534
      EX2=EXP(-.637*R)                                                  0535
      IF(EX2-1.E-10)122,123,123                                         0536
  123 CONTINUE                                                          0537
      EX3=EXP(-1.919*R)                                                 0538
      GO TO 124                                                         0539
  122 EX3=0.0                                                           0540
  124 CONTINUE                                                          0541
      RRR1 = 1.0/R                                                      0542
      V=(.191*EX1+.474*EX2+.335*EX3)*RRR1                               0543
      FR=B*B*RRR1+V*R/EPS-R                                             0544
      V1=-(V+.0531865*EX1+.30181*EX2+.6437*EX3)*RRR1                    0545
      FR1=-B*B*RRR1**2+(V+V1*R)/EPS-1.                                  0546
      Q=FR/FR1                                                          0547
      R=R-Q                                                             0548
      IF (ABS(Q/R).GT..001) GO TO 121                                   0549
      ROC=-2.*(EPS-V)/V1                                                0550
      SQE=SQRT(EPS)                                                     0551
      CC=(.235800+SQE)/(.126000+SQE)                                    0552
      AA=2.*EPS*(1.+(1.0144/SQE))*B**CC                                 0553
      FF=(SQRT(AA**2+1.)-AA)*((69350.+EPS)/(83550.+EPS))                0554
      DELTA=(R-B)*AA*FF/(FF+1.)                                         0555
      C=(B+DELTA+ROC)/(R+ROC)                                           0556
      C2=AMIN1(1.0,C*C)                                                 0557
      S2=1.-C2                                                          0558
C                                                                       0559
C     ENERGY TRANSFER, LOCAL INELASTIC ENERGY LOSS, SCATTERING ANGLE    0560
C                                                                       0561
      DEN = EC(1,J)*S2*E                                                0562
      DENS = DENS+DEN                                                   0563
      IF(DEN.GT.ED(J)) IC1 = IC1+1                                      0564
      DEEOR = CVMGT(0.,CK(J)*KOR(1,J)*SQRT(E)*EX1,INEL.EQ.1)            0565
      DEES = DEES+DEEOR                                                 0566
      TAU = R*A(1,J)*SQRT(S2)                                           0567
      CT=2.*C2-1.                                                       0568
      ST=SQRT(1.-CT*CT)                                                 0569
      CU = CT+MU(1,J)                                                   0570
      IF(ABS(CU).LT.1.E-8) CU = 1.E-8                                   0571
      PSI = ATAN2(ST,CU)                                                0572
      IF(PSI.LT.0.) PSI = PSI+PI                                        0573
C                                                                       0574
C     NEW DIRECTIONS                                                    0575
C                                                                       0576
```

```
      CALL DIRCOS(COSX,COSY,COSZ,SINE,PSI,PHI)                          0577
C                                                                       0578
C     END OF WEAK COLLISION LOOP                                        0579
C                                                                       0580
245   CONTINUE                                                          0581
C                                                                       0582
C     BOND BREAKING                                                     0583
C                                                                       0584
      IF(DEN.LE.ED(J)) GOTO 128                                         0585
      BBR(IX,J) = BBR(IX,J)+1.                                          0586
      BBRTOT(J) = BBRTOT(J)+1.                                          0587
C                                                                       0588
C     GENERATION OF PRIMARY KNOCKON                                     0589
C                                                                       0590
128   IF(IDREL.LT.0) GOTO 127                                           0591
      IF(DEN.LE.BE(J)) GOTO 127                                         0592
      IF(J.GE.JSP1.AND.J.LE.JSP2) GOTO 127                              0593
      ER(NREC1,1)=DEN-BE(J)                                             0594
      IF(ER(NREC1,1).LE.EF(J)) GOTO 127                                 0595
      LMV(NREC1,1)=LM                                                   0596
      CPHI=COS(PHI)                                                     0597
      SPHI=SIN(PHI)                                                     0598
      XR(NREC1,1)=X1                                                    0599
      YR(NREC1,1)=Y-P*(SPHI*CZ-CPHI*CY*CX)/SX                           0600
      ZR(NREC1,1)=Z+P*(SPHI*CY+CPHI*CZ*CX)/SX                           0601
      CSXR(NREC1,1)=CX                                                  0602
      CSYR(NREC1,1)=CY                                                  0603
      CSZR(NREC1,1)=CZ                                                  0604
      SNXR(NREC1,1)=SX                                                  0605
      PHIR(NREC1,1)=PHI-PI                                              0606
      CT = AMIN1(CT,.99999999)                                          0607
      PSIR(NREC1,1)=ATAN2(ST,1.-CT)                                     0608
      IXST(NREC1,1) = IX                                                0609
      L(NREC1,1) = X1+1.                                                0610
      KO(NREC1,J,1)=1                                                   0611
      EORC(NREC1,1) = ER(NREC1,1)                                       0612
      INOUT(NREC1,1)=SIGN(1.,CX)                                        0613
      DEL(IX,J) = DEL(IX,J)-1.                                          0614
      NREC1=NREC1+1                                                     0615
      IF(NREC1.LE.MAXMAX) GOTO 127                                      0616
75    WRITE(IO,130)                                                     0617
130   FORMAT(5X,'RECOIL STORAGE CAPACITY EXCEEDED')                     0618
      STOP                                                              0619
C                                                                       0620
C     TOTAL INELASTIC ENERGY LOSS                                       0621
C                                                                       0622
127   ASIGT = ASIG-TAU*DNS(IX)                                          0623
      GOTO(15,16,17)INEL                                                0624
```

```
15    DEE = CVMGT(0.,K23*ASIGT*SQRT(E),X.LT.0.)                          0625
      GOTO 7                                                             0626
16    DEE = DEES                                                         0627
      GOTO 7                                                             0628
17    DEE = CVMGT(DEES,.5*(K23*ASIGT*SQRT(E)+DEES),X.LT.0.)              0629
C                                                                        0630
C     COLLISION COUNTER                                                  0631
C                                                                        0632
    7 IF(DENS.GT.0.) IC = IC+1                                           0633
C                                                                        0634
C     PROJECTILE ENERGY AND ENERGY DEPOSITION FUNCTIONS                  0635
C                                                                        0636
      IXI = INT(X/DQX)+1                                                 0637
      IF(IXI.LE.0) IXI = 1                                               0638
      ION(IXI) = ION(IXI)+DEE                                            0639
      IONTOT = IONTOT+DEE                                                0640
      NCLR(IXI) = NCLR(IXI)+DENS                                         0641
      NCLTOT = NCLTOT+DENS                                               0642
      PHON(IXI) = PHON(IXI)+DENS-DEN                                     0643
      PHOTOT = PHOTOT+DENS-DEN                                           0644
      IF(DEN.LE.ED(JPKO)) GOTO 28                                        0645
      DMGN(IXI) = DMGN(IXI)+DEN                                          0646
      DMGTOT = DMGTOT+DEN                                                0647
      GOTO 4860                                                          0648
28    PHON(IXI) = PHON(IXI)+DEN                                          0649
      PHOTOT = PHOTOT+DEN                                                0650
4860  IF(JPKO.LT.JFRP) GOTO 29                                           0651
      EDM = 0.                                                           0652
      SUM = 0.                                                           0653
      DO 4850 JP = JFRP,NCP                                              0654
      SUM = SUM+QUX(IXI,JP)                                              0655
4850  EDM = EDM+QUX(IXI,JP)/ED(JP)                                       0656
      EDM = SUM/EDM                                                      0657
      IF(DEN.LE.EDM) GOTO 29                                             0658
      IF(DEN.GT.2.5*EDM) GOTO 4840                                       0659
      FPKP(IXI) = FPKP(IXI)+1.                                           0660
      FPKTOT = FPKTOT+1.                                                 0661
      GOTO 29                                                            0662
4840  FPKP(IXI) = FPKP(IXI)+.35*DEN/EDM                                  0663
      FPKTOT = FPKTOT+.35*DEN/EDM                                        0664
C                                                                        0665
C     ENERGY AND DEFLECTION POINT OF PROJECTILE                          0666
C                                                                        0667
29    E = E-DEE                                                          0668
      X=X-TAU*CX                                                         0669
      Y=Y-TAU*CY                                                         0670
      Z=Z-TAU*CZ                                                         0671
      PL=PL+LM-TAU                                                       0672
```

```
C                                                                       0673
C     PARTICLE STOPPED, REFLECTED OR TRANSMITTED                        0674
C                                                                       0675
      IF(X.LT.-SU) GOTO 8                                               0676
      IF(X.GT.TT) GOTO 9                                                0677
      IF(E-DENS.LT.EF(1)) GOTO 4                                        0678
      GOTO 5                                                            0679
    8 ENO=E*COSX**2                                                     0680
      IF(ENO.GT.SBE(1)) GOTO 24                                         0681
C                                                                       0682
C     BACKREFLECTED PROJECTILES                                         0683
C                                                                       0684
      X = -SU                                                           0685
      COSX=-COSX                                                        0686
      GOTO 5                                                            0687
C                                                                       0688
C     BACKSCATTERED PROJECTILES                                         0689
C                                                                       0690
   24 ES = E-SBE(1)                                                     0691
      IB=IB+1                                                           0692
      EB = EB+ES                                                        0693
      GOTO 6                                                            0694
C                                                                       0695
C     TRANSMITTED PROJECTILES                                           0696
C                                                                       0697
    9 IT=IT+1                                                           0698
      ET=ET+E                                                           0699
      GOTO 6                                                            0700
C                                                                       0701
C     PROJECTILE STOPPED                                                0702
C                                                                       0703
    4 DEL(IX,1) = DEL(IX,1)+1.                                          0704
      IXI = INT(X/DQX)+1                                                0705
      IF(IXI.LE.0) IXI = 1                                              0706
      IRP(IXI) = IRP(IXI)+1                                             0707
      IP=INT(PL/DQX+1.)                                                 0708
      IF(IP.GT.NQX) IP=NQX                                              0709
      IPL(IP)=IPL(IP)+1                                                 0710
      IF(X.LT.0.) X = 0.                                                0711
      XSUM = XSUM+X                                                     0712
      X2SUM = X2SUM+X*X                                                 0713
      X3SUM = X3SUM+X**3                                                0714
      X4SUM = X4SUM+X**4                                                0715
      PLSUM=PLSUM+PL                                                    0716
      PL2SUM=PL2SUM+PL**2                                               0717
      ICSUM=ICSUM+IC                                                    0718
      ICSUM1=ICSUM1+IC1                                                 0719
      GOTO 6                                                            0720
```

```
C                                                                       0721
C     ELEASTIC ENERGY LOSS                                              0722
C                                                                       0723
5     E = E-DENS                                                        0724
      GOTO 1                                                            0725
C                                                                       0726
C     END OF PROJECTILE SECTION                                         0727
C                                                                       0728
6     NREC1=NREC1-1                                                     0729
      IF(NREC1.LE.0) GOTO 10                                            0730
C                                                                       0731
C     RECOIL ATOM SECTION                                               0732
C                                                                       0733
      NREC2=0                                                           0734
73    IF(NREC1.LE.0) GOTO 74                                            0735
C                                                                       0736
C     DIRECTIONS OF NEW RECOILS (LIST 1)                                0737
C                                                                       0738
      CALL DIRCOSV(CSXR(1,1),CSYR(1,1),CSZR(1,1),SNXR(1,1),             0739
     CPSIR(1,1),PHIR(1,1),NREC1)                                        0740
C                                                                       0741
C     MOVE NEW RECOILS (LIST 1) TO LIST 2                               0742
C                                                                       0743
      DO 91 IREC1=1,NREC1                                               0744
      IREC=IREC1+NREC2                                                  0745
      ER(IREC,2)=ER(IREC1,1)                                            0746
      EORC(IREC,2) = EORC(IREC1,1)                                      0747
      XR(IREC,2)=XR(IREC1,1)                                            0748
      YR(IREC,2)=YR(IREC1,1)                                            0749
      ZR(IREC,2)=ZR(IREC1,1)                                            0750
      CSXR(IREC,2)=CSXR(IREC1,1)                                        0751
      CSYR(IREC,2)=CSYR(IREC1,1)                                        0752
      CSZR(IREC,2)=CSZR(IREC1,1)                                        0753
      SNXR(IREC,2)=SNXR(IREC1,1)                                        0754
      L(IREC,2)=L(IREC1,1)                                              0755
      LMV(IREC,2)=LMV(IREC1,1)                                          0756
      LABEL(IREC,2)=LABEL(IREC1,1)                                      0757
      IXST(IREC,2)=IXST(IREC1,1)                                        0758
      KO(IREC,LABEL(IREC1,1),2)=KO(IREC1,LABEL(IREC1,1),1)              0759
      INOUT(IREC,2)=INOUT(IREC1,1)                                      0760
91    CONTINUE                                                          0761
      NREC2=NREC2+NREC1                                                 0762
      MAXA=MAX0(MAXA,NREC2)                                             0763
      NREC1=0                                                           0764
74    NALL=NALL+NREC2                                                   0765
C                                                                       0766
C     PROCESS THE PARTICLES IN LIST 2                                   0767
C                                                                       0768
```

```
C     MOVE PARTICLES                                                    0769
C                                                                       0770
      DO 60 IREC1=1,NREC2                                               0771
      XR(IREC1,2)=XR(IREC1,2)+LMV(IREC1,2)*CSXR(IREC1,2)                0772
      YR(IREC1,2)=YR(IREC1,2)+LMV(IREC1,2)*CSYR(IREC1,2)                0773
      ZR(IREC1,2)=ZR(IREC1,2)+LMV(IREC1,2)*CSZR(IREC1,2)                0774
   60 CONTINUE                                                          0775
C                                                                       0776
C     STORE DIRECTIONS                                                  0777
C                                                                       0778
      DO 81 IREC1=1,NREC2                                               0779
      CXR(IREC1)=CSXR(IREC1,2)                                          0780
      CYR(IREC1)=CSYR(IREC1,2)                                          0781
      CZR(IREC1)=CSZR(IREC1,2)                                          0782
      SXR(IREC1)=SNXR(IREC1,2)                                          0783
      DEERS(IREC1)=0.                                                   0784
      TS(IREC1)=0.                                                      0785
   81 CONTINUE                                                          0786
C                                                                       0787
C     DEPTH INTERVAL OF COLLISION                                       0788
C                                                                       0789
      DO 84 IREC1=1,NREC2                                               0790
      IXRV(IREC1) = ISRCHFGE(NQX+1,XNO(1),1,XR(IREC1,2))-1              0791
      IXRV(IREC1) = MAXO(IXRV(IREC1),1)                                 0792
84    TEST(IREC1) = IXRV(IREC1).GT.NQX                                  0793
      IERR = ILLZ(NREC2,TEST,1)                                         0794
      IF(IERR.EQ.NREC2) GOTO 80                                         0795
      IERR = IERR+1                                                     0796
      GOTO 2140                                                         0797
C                                                                       0798
C     DEPTH-DEPENDENT PARAMETERS                                        0799
C                                                                       0800
80    DO 85 IREC1=1,NREC2                                               0801
      LMV(IREC1,2) = DNS(IXRV(IREC1))**(-1./3.)                         0802
      PMAXV(IREC1) = LMV(IREC1,2)/SQ2PI                                 0803
85    KL2(IREC1) = 0.                                                   0804
      DO 4150 JP=1,NCP                                                  0805
      DO 4150 IREC1=1,NREC2                                             0806
      J = LABEL(IREC1,2)                                                0807
      IXRC=IXRV(IREC1)                                                  0808
      KL2(IREC1)=KL2(IREC1)+KL(J,JP)*QUX(IXRC,JP)                       0809
4150  CONTINUE                                                          0810
C                                                                       0811
C     WEAK COLLISION LOOP                                               0812
C                                                                       0813
      DO 235 KK=KK2,4                                                   0814
C                                                                       0815
C     SPECIES OF COLLISION PARTNER                                      0816
```

```
C                                                                       0817
      IF(NCP.GT.1) GOTO 82                                              0818
      DO 83 IREC1=1,NREC2                                               0819
   83 LABEL(IREC1,1) = 1                                                0820
      GOTO 90                                                           0821
   82 DO 86 IREC1=1,NREC2                                               0822
      IXRC = IXRV(IREC1)                                                0823
   86 SQUXV(1,IREC1) = QUX(IXRC,1)                                      0824
      IF(NCP.LE.2) GOTO 87                                              0825
      DO 88 JP=2,NCP-1                                                  0826
      DO 88 IREC1=1,NREC2                                               0827
      IXRC = IXRV(IREC1)                                                0828
   88 SQUXV(JP,IREC1) = SQUXV(JP-1,IREC1)+QUX(IXRC,JP)                  0829
   87 DO 89 IREC1=1,NREC2                                               0830
      SQUXV(NCP,IREC1) = 1.                                             0831
   89 LABEL(IREC1,1) = ISRCHFGE(NCP,SQUXV(1,IREC1),1,RANF())            0832
C                                                                       0833
C     AZIMUTHAL ANGLE, IMPACT PARAMETER, REDUCED ENERGY                 0834
C                                                                       0835
   90 DO 236 IREC1=1,NREC2                                              0836
      PHIR(IREC1,2)=2.*PI*RANF()                                        0837
      PR(IREC1)=PMAXV(IREC1)*SQRT(RANF()+4.-KK)                         0838
      X2(IREC1)=XR(IREC1,2)-PR(IREC1)*COS(PHIR(IREC1,2))*SXR(IREC1)     0839
      PR(IREC1)=CVMGT(1.E10,PR(IREC1),X2(IREC1).LT.0.)                  0840
      BR(IREC1)=PR(IREC1)/A(LABEL(IREC1,2),LABEL(IREC1,1))              0841
      EPSR(IREC1) = ER(IREC1,2)*F(LABEL(IREC1,2),LABEL(IREC1,1))        0842
  236 CONTINUE                                                          0843
C                                                                       0844
C     'MAGIC' CALCULATION OF POLAR DEFLECTION                           0845
C                                                                       0846
      CALL SCOPY(NREC2,BR,1,RRV,1)                                      0847
      IVMIN=1                                                           0848
      IVMAX=NREC2                                                       0849
  205 DO 206 IV=IVMIN,IVMAX                                             0850
      EX1R(IV)=EXP(-.279*RRV(IV))                                       0851
      EX2R(IV)=EXP(-.637*RRV(IV))                                       0852
      EX3R(IV)=EXP(-1.919*RRV(IV))                                      0853
      RRR1=1./RRV(IV)                                                   0854
      VR(IV)=(.191*EX1R(IV)+.474*EX2R(IV)+.335*EX3R(IV))*RRR1           0855
      FR=BR(IV)*BR(IV)*RRR1+VR(IV)*RRV(IV)/EPSR(IV)-RRV(IV)             0856
      V1R(IV)=-(VR(IV)+.0531865*EX1R(IV)+.30181*EX2R(IV)+               0857
     1     .6437*EX3R(IV))*RRR1                                         0858
      FR1=-BR(IV)*BR(IV)*RRR1*RRR1+(VR(IV)+V1R(IV)*RRV(IV))/            0859
     1    EPSR(IV)-1.                                                   0860
      Q=FR/FR1                                                          0861
      RRV(IV)=RRV(IV)-Q                                                 0862
      TEST(IV)=ABS(Q/RRV(IV)).GT.0.001                                  0863
  206 CONTINUE                                                          0864
```

```
C     GET MAX AND MIN INDEX OF TEST FAILURES                          0865
      IVMIN=IVMIN+ILLZ(IVMAX-IVMIN+1,TEST(IVMIN),1)                   0866
      IF(IVMIN.GT.IVMAX) GO TO 207                                    0867
      IVMAX=IVMAX-ILLZ(IVMAX-IVMIN+1,TEST(IVMIN),-1)                  0868
      IF(IVMIN.GT.IVMAX) GO TO 207                                    0869
      GO TO 205                                                       0870
  207 DO 208 IV=1,NREC2                                               0871
      ROCINV=-.5*V1R(IV)/(EPSR(IV)-VR(IV))                            0872
      SQE=SQRT(EPSR(IV))                                              0873
      CC=(.235800+SQE)/(.126000+SQE)                                  0874
      AA=2.*EPSR(IV)*(1.+(1.0144/SQE))*BR(IV)**CC                     0875
      FF=(SQRT(AA*AA+1.)-AA)*((69350.+EPSR(IV))/(83550.+EPSR(IV)))    0876
      DELTA=(RRV(IV)-BR(IV))*AA*FF/(FF+1.)                            0877
      C=(ROCINV*(BR(IV)+DELTA)+1.)/(ROCINV*RRV(IV)+1.)                0878
      C2R(IV)=AMIN1(1.0,C*C)                                          0879
  208 S2R(IV)=1.-C2R(IV)                                              0880
C                                                                     0881
C     ENERGY TRANSFER, LOCAL ELECTRONIC ENERGY LOSS, SCATTERING ANGLE 0882
C                                                                     0883
      DO 237 IREC1=1,NREC2                                            0884
      T(IREC1)=ER(IREC1,2)*S2R(IREC1)*EC(LABEL(IREC1,2),LABEL(IREC1,1)) 0885
      TS(IREC1)=TS(IREC1)+T(IREC1)                                    0886
      DEEROR=CVMGT(0.,KOR(LABEL(IREC1,2),LABEL(IREC1,1))              0887
     C*SQRT(ER(IREC1,2))*EX1R(IREC1),INELR.EQ.1)                      0888
      DEERS(IREC1)=DEERS(IREC1)+DEEROR                                0889
      TAUR(IREC1)=RRV(IREC1)*A(LABEL(IREC1,2),LABEL(IREC1,1))         0890
     C*SQRT(S2R(IREC1))                                               0891
      CTR(IREC1)=C2R(IREC1)+C2R(IREC1)-1.                             0892
      STR(IREC1)=SQRT(1.-CTR(IREC1)*CTR(IREC1))                       0893
      CUR(IREC1) = CTR(IREC1)+MU(LABEL(IREC1,2),LABEL(IREC1,1))       0894
      CUR(IREC1) = CVMGT(CUR(IREC1),1.E-8,ABS(CUR(IREC1)).GE.1.E-8)   0895
      PSIR(IREC1,2) = ATAN2(STR(IREC1),CUR(IREC1))                    0896
      PSIR(IREC1,2) = CVMGT(PSIR(IREC1,2),PSIR(IREC1,2)+PI,PSIR(IREC1,2) 0897
     C.GE.0.)                                                         0898
  237 CONTINUE                                                        0899
C                                                                     0900
C     NEW DIRECTIONS OF OLD RECOIL                                    0901
C                                                                     0902
      CALL DIRCOSV(CSXR(1,2),CSYR(1,2),CSZR(1,2),SNXR(1,2),           0903
     1             PSIR(1,2),PHIR(1,2),NREC2)                         0904
  235 CONTINUE                                                        0905
C                                                                     0906
C     BOND BREAKING                                                   0907
C                                                                     0908
      DO 240 IREC1=1,NREC2                                            0909
      IF(T(IREC1).LE.ED(LABEL(IREC1,1))) GO TO 240                    0910
      J = LABEL(IREC1,1)                                              0911
      IXRC = IXRV(IREC1)                                              0912
```

```
      BBR(IXRC,J) = BBR(IXRC,J)+1.                                      0913
      BBRTOT(J) = BBRTOT(J)+1.                                          0914
  240 CONTINUE                                                          0915
C                                                                       0916
C     CREATE SECONDARY KNOCK-ON ATOMS (LIST 1)                          0917
C                                                                       0918
      DO 246 IREC1=1,NREC2                                              0919
      IF(T(IREC1).LE.BE(LABEL(IREC1,1))) GO TO 246                      0920
      IF(T(IREC1)-BE(LABEL(IREC1,1)).LE.EF(LABEL(IREC1,1))) GO TO 246   0921
      IF(LABEL(IREC1,1).GE.JSP1.AND.LABEL(IREC1,1).LE.JSP2) GOTO 246    0922
      NREC1=NREC1+1                                                     0923
      LMV(NREC1,1)=LMV(IREC1,2)                                         0924
      ER(NREC1,1)=T(IREC1)-BE(LABEL(IREC1,1))                           0925
      CPR=COS(PHIR(IREC1,2))                                            0926
      SPR=SIN(PHIR(IREC1,2))                                            0927
      XR(NREC1,1)=X2(IREC1)                                             0928
      YR(NREC1,1)=YR(IREC1,2)-PR(IREC1)*(SPR*CZR(IREC1)-                0929
     1          CPR*CYR(IREC1)*CXR(IREC1))/SXR(IREC1)                   0930
      ZR(NREC1,1)=ZR(IREC1,2)+PR(IREC1)*(SPR*CYR(IREC1)+                0931
     1          CPR*CXR(IREC1)*CZR(IREC1))/SXR(IREC1)                   0932
      CSXR(NREC1,1)=CXR(IREC1)                                          0933
      CSYR(NREC1,1)=CYR(IREC1)                                          0934
      CSZR(NREC1,1)=CZR(IREC1)                                          0935
      SNXR(NREC1,1)=SXR(IREC1)                                          0936
      PHIR(NREC1,1)=PHIR(IREC1,2)-PI                                    0937
      CTR(NREC1)=AMIN1(CTR(IREC1),.99999999)                            0938
      PSIR(NREC1,1)=ATAN2(STR(IREC1),1.-CTR(NREC1))                     0939
      IXST(NREC1,1)=IXRV(IREC1)                                         0940
      L(NREC1,1)=XR(NREC1,1)+1.                                         0941
      LABEL(NREC1,1) = LABEL(IREC1,1)                                   0942
      KO(NREC1,LABEL(IREC1,1),1)=0                                      0943
      EORC(NREC1,1) = ER(NREC1,1)                                       0944
      INOUT(NREC1,1)=INOUT(IREC1,2)                                     0945
      DEL(IXRV(IREC1),LABEL(IREC1,1))=DEL(IXRV(IREC1),LABEL(IREC1,1))-1.  0946
  246 CONTINUE                                                          0947
C                                                                       0948
C     INELASTIC ENERGY LOSS                                             0949
C                                                                       0950
      DO 238 IREC1=1,NREC2                                              0951
      IXRC=IXRV(IREC1)                                                  0952
      ASIGTR(IREC1)=(LMV(IREC1,2)-TAUR(IREC1))*DNS(IXRC)                0953
  238 CONTINUE                                                          0954
      GO TO(115,116,117),INELR                                          0955
  115 DO 241 IREC1=1,NREC2                                              0956
      DEER(IREC1)=CVMGT(0.,KL2(IREC1)*ASIGTR(IREC1)*SQRT(ER(IREC1,2)),  0957
     1                  XR(IREC1,2).LT.0.)                              0958
  241 CONTINUE                                                          0959
      GO TO 242                                                         0960
```

```
  116 DO 243 IREC1=1,NREC2                                              0961
      DEER(IREC1)=DEERS(IREC1)                                          0962
  243 CONTINUE                                                          0963
      GO TO 242                                                         0964
  117 DO 244 IREC1=1,NREC2                                              0965
      DEER(IREC1)=CVMGT(DEERS(IREC1),.5*(KL2(IREC1)                     0966
     1           *ASIGTR(IREC1)*SQRT(ER(IREC1,2))+DEERS(IREC1)),        0967
     2           XR(IREC1,2).LT.0.)                                     0968
  244 CONTINUE                                                          0969
  242 CONTINUE                                                          0970
C                                                                       0971
C     FINAL ENERGY AND POSITION OF OLD RECOIL                           0972
C                                                                       0973
      DO 253 IREC1=1,NREC2                                              0974
      ER(IREC1,2)=ER(IREC1,2)-DEER(IREC1)                               0975
      XR(IREC1,2)=XR(IREC1,2)-TAUR(IREC1)*CXR(IREC1)                    0976
      YR(IREC1,2)=YR(IREC1,2)-TAUR(IREC1)*CYR(IREC1)                    0977
      ZR(IREC1,2)=ZR(IREC1,2)-TAUR(IREC1)*CZR(IREC1)                    0978
      TEST(IREC1)=ER(IREC1,2)-TS(IREC1).LE.EF(LABEL(IREC1,2))           0979
     C.OR.XR(IREC1,2).LT.-SU.OR.XR(IREC1,2).GT.TT                       0980
  253 CONTINUE                                                          0981
C                                                                       0982
C     ANY RECOIL ATOM SLOWED DOWN, TRANSMITTED OR SPUTTERED ?           0983
C                                                                       0984
      IVMIN=1+ILLZ(NREC2,TEST,1)                                        0985
      IF(IVMIN.GT.NREC2) GO TO 247                                      0986
      IVMAX=NREC2-ILLZ(NREC2,TEST,-1)                                   0987
      IREC1=IVMIN                                                       0988
  248 IF(IREC1.GT.IVMAX.OR.IREC1.GT.NREC2) GOTO 247                     0989
      IF(.NOT.TEST(IREC1)) GOTO 249                                     0990
      IF(XR(IREC1,2).LT.(-SU)) GO TO 251                                0991
      IF(XR(IREC1,2).GT.TT) GO TO 255                                   0992
      IF(ER(IREC1,2)-TS(IREC1).LE.EF(LABEL(IREC1,2))) GO TO 252         0993
  249 IREC1 = IREC1+1                                                   0994
      GOTO 248                                                          0995
  251 ENOR=ER(IREC1,2)*CXR(IREC1)*CXR(IREC1)                            0996
      IF(ENOR.GT.SBE(LABEL(IREC1,2))) GO TO 254                         0997
C                                                                       0998
C     BACKREFLECTED RECOIL                                              0999
C                                                                       1000
      XR(IREC1,2)=-SU                                                   1001
      CSXR(IREC1,2)=-CSXR(IREC1,2)                                      1002
      GO TO 248                                                         1003
C                                                                       1004
C     SPUTTERED RECOIL                                                  1005
C                                                                       1006
  254 ESP=ER(IREC1,2)-SBE(LABEL(IREC1,2))                               1007
C                                                                       1008
```

```
C     TOTAL NUMBER AND ENERGY OF SPUTTERED PARTICLES                    1009
C                                                                       1010
      IBSP(LABEL(IREC1,2))=IBSP(LABEL(IREC1,2))+1                       1011
      EBSP(LABEL(IREC1,2))=EBSP(LABEL(IREC1,2))+ESP                     1012
C                                                                       1013
C     4 GROUPS:ION IN , PKA ;ION IN , SKA ;ION OUT, PKA ;ION OUT, SKA   1014
C                                                                       1015
      KOI=KO(IREC1,LABEL(IREC1,2),2)                                    1016
      IF(INOUT(IREC1,2).EQ.-1) GO TO 61                                 1017
      IF(KOI.EQ.0) GO TO 62                                             1018
      ISPINP(LABEL(IREC1,2))=ISPINP(LABEL(IREC1,2))+1                   1019
      ESPINP(LABEL(IREC1,2))=ESPINP(LABEL(IREC1,2))+ESP                 1020
      GO TO 250                                                         1021
   62 ISPINS(LABEL(IREC1,2))=ISPINS(LABEL(IREC1,2))+1                   1022
      ESPINS(LABEL(IREC1,2))=ESPINS(LABEL(IREC1,2))+ESP                 1023
      GO TO 250                                                         1024
   61 IF(KOI.EQ.0) GO TO 163                                            1025
      ISPOP(LABEL(IREC1,2))=ISPOP(LABEL(IREC1,2))+1                     1026
      ESPOP(LABEL(IREC1,2))=ESPOP(LABEL(IREC1,2))+ESP                   1027
      GO TO 250                                                         1028
  163 ISPOS(LABEL(IREC1,2))=ISPOS(LABEL(IREC1,2))+1                     1029
      ESPOS(LABEL(IREC1,2))=ESPOS(LABEL(IREC1,2))+ESP                   1030
      GOTO 250                                                          1031
C                                                                       1032
C     RECOIL EXCEEDING TT                                               1033
C                                                                       1034
  255 DEL(IXRV(IREC1),LABEL(IREC1,2))=DEL(IXRV(IREC1),LABEL(IREC1,2))+1. 1035
      GOTO 250                                                          1036
C                                                                       1037
C     RECOIL SLOWED DOWN                                                1038
C                                                                       1039
  252 IXRC = IXRV(IREC1)                                                1040
      IF(EORC(IREC1,2).GT.ED(LABEL(IREC1,2)).OR.IRC0.LT.0) GOTO 257     1041
      IXRC = IXST(IREC1,2)                                              1042
  257 DEL(IXRC,LABEL(IREC1,2)) = DEL(IXRC,LABEL(IREC1,2))+1.            1043
C                                                                       1044
C     REARRANGEMENT OF PARTICLES IN LIST 2                              1045
C                                                                       1046
  250 IF(IREC1.GE.NREC2) GOTO 258                                       1047
      ER(IREC1,2)=ER(NREC2,2)                                           1048
      TS(IREC1)=TS(NREC2)                                               1049
      XR(IREC1,2)=XR(NREC2,2)                                           1050
      YR(IREC1,2)=YR(NREC2,2)                                           1051
      ZR(IREC1,2)=ZR(NREC2,2)                                           1052
      CSXR(IREC1,2)=CSXR(NREC2,2)                                       1053
      CSYR(IREC1,2)=CSYR(NREC2,2)                                       1054
      CSZR(IREC1,2)=CSZR(NREC2,2)                                       1055
      SNXR(IREC1,2)=SNXR(NREC2,2)                                       1056
```

```
      PHIR(IREC1,2)=PHIR(NREC2,2)                                       1057
      PSIR(IREC1,2)=PSIR(NREC2,2)                                       1058
      L(IREC1,2)=L(NREC2,2)                                             1059
      LABEL(IREC1,2)=LABEL(NREC2,2)                                     1060
      IXST(IREC1,2)=IXST(NREC2,2)                                       1061
      EORC(IREC1,2)=EORC(NREC2,2)                                       1062
      LMV(IREC1,2)=LMV(NREC2,2)                                         1063
      KO(IREC1,LABEL(IREC1,2),2)=KO(NREC2,LABEL(NREC2,2),2)             1064
      INOUT(IREC1,2)=INOUT(NREC2,2)                                     1065
      IXRV(IREC1) = IXRV(NREC2)                                         1066
      TEST(IREC1) = TEST(NREC2)                                         1067
  258 NREC2=NREC2-1                                                     1068
      IF(NREC2+1.GT.IVMAX) GO TO 249                                    1069
      GOTO 248                                                          1070
C                                                                       1071
C     ELASTIC ENERGY LOSS                                               1072
C                                                                       1073
  247 DO 256 IREC1=1,NREC2                                              1074
      ER(IREC1,2)=ER(IREC1,2)-TS(IREC1)                                 1075
  256 CONTINUE                                                          1076
C                                                                       1077
C     END OF RECOIL ATOM SECTION                                        1078
C                                                                       1079
      IF(NREC1+NREC2.GT.MAXMAX) GOTO 75                                 1080
      IF(NREC1+NREC2.GT.0) GOTO 73                                      1081
C                                                                       1082
C     DYNAMIC RELAXATION SECTION                                        1083
C                                                                       1084
      IF(IDREL.NE.0) GOTO 10                                            1085
C                                                                       1086
C     NEW AREAL DENSITIES                                               1087
C                                                                       1088
      DO 4180 JP=1,NCP                                                  1089
      DO 4180 MM=1,NQX                                                  1090
 4180 DEL(MM,JP) = DEL(MM,JP)*FLC/NH                                    1091
      DO 4182 MM=1,NQX                                                  1092
      DXV(MM) = XNO(MM+1)-XNO(MM)                                       1093
      DNSOV(MM) = DNS(MM)                                               1094
 4182 SUMV(MM) = 0.                                                     1095
      DO 4184 JP=1,NCP                                                  1096
      DO 4184 MM=1,NQX                                                  1097
 4184 SUMV(MM) = SUMV(MM)+DEL(MM,JP)                                    1098
      DO 4186 MM=1,NQX                                                  1099
 4186 CHAV(MM) = ABS(SUMV(MM))/DNSOV(MM)/DXV(MM)                        1100
      MM = ISMAX(NQX,CHAV,1)                                            1101
      IF(CHAV(MM).GT.CHAMAX) CHAMAX = CHAV(MM)                          1102
      DO 4190 JP=1,NCP                                                  1103
      DO 4190 MM=1,NQX                                                  1104
```

```
4190  DEL(MM,JP) = DEL(MM,JP)+QUX(MM,JP)*DNSOV(MM)*DXV(MM)                    1105
C                                                                            1106
C     PROJECTILE REEMISSION                                                  1107
C                                                                            1108
      IF(QU1MX.GE.1.) GOTO 2350                                              1109
      DO 4192 MM=1,NQX                                                       1110
4192  SUMV(MM) = 0.                                                          1111
      DO 4200 JP=2,NCP                                                       1112
      DO 4200 MM=1,NQX                                                       1113
4200  SUMV(MM) = SUMV(MM)+DEL(MM,JP)                                         1114
      DO 4202 MM=1,NQX                                                       1115
      D1MXV(MM) = QU1MX*SUMV(MM)/(1.-QU1MX)                                  1116
4202  TEST(MM) = DEL(MM,1).LE.D1MXV(MM)                                      1117
      DO 4204 MM=1,NQX                                                       1118
      IF(TEST(MM)) GOTO 4204                                                 1119
      REEM1 = REEM1+DEL(MM,1)-D1MXV(MM)                                      1120
      DEL(MM,1) = D1MXV(MM)                                                  1121
4204  CONTINUE                                                               1122
C                                                                            1123
C     RELAXATION TO NEW INTERVAL THICKNESSES                                 1124
C                                                                            1125
2350  DO 2352 MM=1,NQX                                                       1126
2352  XXX(MM) = 0.                                                           1127
      DO 4210 JP=1,NCP                                                       1128
      DO 4210 MM=1,NQX                                                       1129
      XXX(MM) = XXX(MM)+DEL(MM,JP)/DNSO(JP)                                  1130
4210  TEST(MM) = XXX(MM).LE.0.                                               1131
      MM = ILSUM(NQX,TEST,1)                                                 1132
      IF(MM.LE.0) GOTO 2050                                                  1133
      WRITE(IO,2105)IH,MM,XXX(MM),NCPM,(DEL(MM,JP),JP=1,NCPM)                1134
2105  FORMAT(5X,'IH=',I5,5X,'MM=',I5,5X,'XXX=',E12.4/                        1135
     C17X,'DEL1..',I1,'=',4(E12.4,1H,),E12.4/                                1136
     C5X,'DEPTH INTERVAL COMPLETELY DEPLETED'/                               1137
     C5X,'----- DECREASE FLC OR INCREASE NH -----')                          1138
      STOP                                                                   1139
2050  SUMM = 0.                                                              1140
      DO 2052 MM=1,NQX                                                       1141
2052  SUMM = SUMM+XXX(MM)                                                    1142
C                                                                            1143
C     SURFACE RECESSION                                                      1144
C                                                                            1145
      DSRRC = XNO(NQX+1)-SUMM                                                1146
      SRRCN = SRRCN+DSRRC                                                    1147
      SRRC = SRRC+DSRRC                                                      1148
C                                                                            1149
C     COMBINATION OF TOO SMALL INTERVALS                                     1150
C     SPLITTING OF TOO LARGE INTERVALS                                       1151
C                                                                            1152
```

```
      DO 4560 MM=1,NQX-2                                                1153
4560  TEST(MM) = XXX(MM).LT..5*DQX.OR.XXX(MM).GT.1.5*DQX                1154
      MM = ILSUM(NQX-2,TEST,1)                                          1155
      IF(MM.LE.0) GOTO 4620                                             1156
      DO 4562 MM=1,NQX-2                                                1157
      IF(XXX(MM).GE..5*DQX) GOTO 4570                                   1158
      XXX(MM) = XXX(MM)+XXX(MM+1)                                       1159
      DO 4580 JP=1,NCP                                                  1160
4580  DEL(MM,JP) = DEL(MM,JP)+DEL(MM+1,JP)                             1161
      DO 4590 MMM=MM+1,NQX-1                                            1162
4590  XXX(MMM) = XXX(MMM+1)                                             1163
      DO 4600 JP=1,NCP                                                  1164
      DO 4600 MMM=MM+1,NQX-1                                            1165
4600  DEL(MMM,JP) = DEL(MMM+1,JP)                                       1166
      XXX(NQX) = DQX                                                    1167
      DO 4610 JP=1,NCP                                                  1168
4610  DEL(NQX,JP) = QUXB(JP)                                            1169
4570  IF(XXX(MM).LE.1.5*DQX) GOTO 4562                                  1170
      DO 4630 MMM=NQX-1,MM+1,-1                                         1171
4630  XXX(MMM+1) = XXX(MMM)                                             1172
      DO 4640 JP=1,NCP                                                  1173
      DO 4640 MMM=NQX-1,MM+1,-1                                         1174
4640  DEL(MMM+1,JP) = DEL(MMM,JP)                                       1175
      XXX(MM) = XXX(MM)/2.                                              1176
      XXX(MM+1) = XXX(MM)                                               1177
      DO 4650 JP=1,NCP                                                  1178
      DEL(MM,JP) = DEL(MM,JP)/2.                                        1179
4650  DEL(MM+1,JP) = DEL(MM,JP)                                         1180
4562  CONTINUE                                                          1181
C                                                                       1182
C     ATOMIC FRACTIONS AND DENSITIES                                    1183
C                                                                       1184
4620  DO 4622 MM=1,NQX                                                  1185
      XNO(MM+1) = XNO(MM)+XXX(MM)                                       1186
      DNS(MM) = 0.                                                      1187
4622  SUMV(MM) = 0.                                                     1188
      DO 4660 JP=1,NCP                                                  1189
      DO 4660 MM=1,NQX                                                  1190
4660  SUMV(MM) = SUMV(MM)+DEL(MM,JP)                                    1191
      DO 4670 JP=1,NCP                                                  1192
      DO 4670 MM=1,NQX                                                  1193
      QUX(MM,JP) = DEL(MM,JP)/SUMV(MM)                                  1194
4670  DNS(MM) = DNS(MM)+QUX(MM,JP)/DNSO(JP)                             1195
      DO 4672 MM=1,NQX                                                  1196
4672  DNS(MM) = 1./DNS(MM)                                              1197
C                                                                       1198
C     PROJECTILE REMOVAL AND READJUSTMENT OF DEPOSITION FUNCTIONS       1199
C     DUE TO SURFACE RECESSION                                          1200
```

```
C                                                                       1201
      IF(SRRCN.LT.DQX/2.) GOTO 2270                                     1202
      J = (INT(2.*SRRCN/DQX)-1)/2+1                                     1203
      SRRCN = SRRCN-J*DQX                                               1204
      DO 2280 I=1,J                                                     1205
2280  IPREC = IPREC+IRP(I)                                              1206
      DO 2290 I=1,NQX-J                                                 1207
      FPKP(I) = FPKP(I+J)                                               1208
      IRP(I) = IRP(I+J)                                                 1209
      ION(I) = ION(I+J)                                                 1210
      NCLR(I) = NCLR(I+J)                                               1211
      DMGN(I) = DMGN(I+J)                                               1212
2290  PHON(I) = PHON(I+J)                                               1213
      DO 2300 I=NQX-J+1,NQX                                             1214
      FPKP(I) = 0.                                                      1215
      IRP(I) = 0                                                        1216
      ION(I) = 0.                                                       1217
      NCLR(I) = 0.                                                      1218
      DMGN(I) = 0.                                                      1219
2300  PHON(I) = 0.                                                      1220
C                                                                       1221
C     CALCULATION AND OUTPUT OF INTEGRAL QUANTITIES                     1222
C                                                                       1223
2270  IF(IDOUT.LE.0) GOTO 10                                            1224
      IF(MOD(IH,IDOUT).NE.0) GOTO 10                                    1225
      AMM = 0                                                           1226
      DO 4300 JP=1,NCP                                                  1227
      ARD(JP) = 0.                                                      1228
4300  CSF(JP) = 0.                                                      1229
      DO 4310 MM=1,NQX                                                  1230
      SUM = 0.                                                          1231
      DO 4320 JP=2,NCP                                                  1232
4320  SUM = SUM+QUX(MM,JP)                                              1233
      DO 4330 JP=1,NCP                                                  1234
4330  ARD(JP) = ARD(JP)+QUX(MM,JP)*XXX(MM)*DNS(MM)                      1235
      IF(XNO(MM).GE.DSF) GOTO 4310                                      1236
      DAM = 1.                                                          1237
      IF(XNO(MM+1).LE.DSF) GOTO 4720                                    1238
      DAM = (DSF-XNO(MM))/XXX(MM)                                       1239
4720  AMM = AMM+DAM                                                     1240
      DO 4340 JP=2,NCP                                                  1241
4340  CSF(JP) = CSF(JP)+QUX(MM,JP)/SUM*DAM                              1242
4310  CONTINUE                                                          1243
      IF(AMM.LE.0) GOTO 4350                                            1244
      DO 4360 JP=2,NCP                                                  1245
4360  CSF(JP) = CSF(JP)/AMM                                             1246
4350  DO 5000 JP=1,5                                                    1247
5000  FLIB(JP) = 0.                                                     1248
```

```
      IF(IMCP.LE.0) GOTO 4990                                          1249
      DO 5010 MM=1,NQX                                                 1250
      SUM = XNO(MM)+XXX(MM)/2.                                         1251
      FLIB(1) = FLIB(1)+QUX(MM,IMCP)*XXX(MM)*DNS(MM)                   1252
5010  FLIB(2) = FLIB(2)+QUX(MM,IMCP)*XXX(MM)*SUM*DNS(MM)               1253
      IF(FLIB(1).LE.0.) GOTO 4990                                      1254
      FLIB(2) = FLIB(2)/FLIB(1)                                        1255
      DO 5020 MM=1,NQX                                                 1256
      SUM = XNO(MM)+XXX(MM)/2.                                         1257
      SUMM = QUX(MM,IMCP)*XXX(MM)*DNS(MM)*(SUM-FLIB(2))**2             1258
      FLIB(3) = FLIB(3)+SUMM                                           1259
      FLIB(4) = FLIB(4)+SUMM*(SUM-FLIB(2))                             1260
5020  FLIB(5) = FLIB(5)+SUMM*(SUM-FLIB(2))**2                          1261
      IF(FLIB(3).LE.1.E-10*FLIB(2)) GOTO 5040                          1262
      FLIB(3) = SQRT(FLIB(3)/FLIB(1))                                  1263
      FLIB(4) = FLIB(4)/FLIB(3)**3                                     1264
      FLIB(5) = FLIB(5)/FLIB(3)**4-3.                                  1265
      GOTO 4990                                                        1266
5040  FLIB(3) = 0.                                                     1267
      FLIB(4) = 0.                                                     1268
      FLIB(5) = 0.                                                     1269
4990  FLUC = FLC*IH/NH                                                 1270
      DO 4370 JP=1,NCP                                                 1271
      SPYD(JP) = (IBSP(JP)-IBSPO(JP))/FLOAT(IDOUT)                     1272
4370  IBSPO(JP) = IBSP(JP)                                             1273
      WRITE(IO1,2035)FLUC,SRRC,REEM1,(CSF(JP),JP=2,NCPM),FLIB,SPYD,ARD 1274
C                                                                      1275
C     INTERPOLATION TO EQUIDISTANT INTERVALS AND OUTPUT OF PROFILES    1276
C                                                                      1277
      IF(IQOUT.LE.0) GOTO 10                                           1278
      IF(MOD(IH,IQOUT).NE.0) GOTO 10                                   1279
      DO 2340 MM=IQXN,IQXX                                             1280
      XX0 = (MM-1)*DQX                                                 1281
      XX1 = MM*DQX                                                     1282
      DO 4220 JP=1,NCP                                                 1283
4220  SSUM(JP) = 0.                                                    1284
      XXX0 = 0.                                                        1285
      DO 2160 MMM=1,NQX                                                1286
      XXX0 = XXX0+XXX(MMM)                                             1287
2160  IF(XXX0.GT.XX0) GOTO 2170                                        1288
      DO 4240 JP=1,NCP                                                 1289
4240  SSUM(JP) = QUXB(JP)*DQX*DNSB                                     1290
      GOTO 2090                                                        1291
2170  DO 4250 JP=1,NCP                                                 1292
4250  SSUM(JP) = SSUM(JP)+(XXX0-XX0)/XXX(MMM)*DEL(MMM,JP)              1293
      IF(XXX0-XX1)2180,2090,2200                                       1294
2180  MMM = MMM+1                                                      1295
      IF(MMM.LE.NQX) GOTO 2210                                         1296
```

```
      DO 4260 JP=1,NCP                                                  1297
4260  SSUM(JP) = SSUM(JP)+(XX1-XXX0)*QUXB(JP)*DNSB                      1298
      GOTO 2090                                                         1299
2210  XX0 = XXX0                                                        1300
      XXX0 = XXX0+XXX(MMM)                                              1301
      GOTO 2170                                                         1302
2200  DO 4270 JP=1,NCP                                                  1303
4270  SSUM(JP) = SSUM(JP)-(XXX0-XX1)/XXX(MMM)*DEL(MMM,JP)               1304
2090  SUM = 0.                                                          1305
      DO 4280 JP=1,NCP                                                  1306
4280  SUM = SUM+SSUM(JP)                                                1307
      DNSI = 0.                                                         1308
      DO 4290 JP=1,NCP                                                  1309
      QUXI(JP) = SSUM(JP)/SUM                                           1310
4290  DNSI = DNSI+QUXI(JP)/DNS0(JP)                                     1311
      DNSI = 1./DNSI                                                    1312
      XXXI = (MM-.5)*DQX                                                1313
2340  WRITE(IO1,2045)XXXI,DNSI,(QUXI(JP),JP=1,NCP)                      1314
C                                                                       1315
C     END OF DYNAMIC RELAXATION SECTION                                 1316
C                                                                       1317
      GOTO 10                                                           1318
C                                                                       1319
C     PRINTED OUTPUT                                                    1320
C                                                                       1321
18    WRITE(IO,4055)AVS0,TITLE                                          1322
4055  FORMAT(' DYNAMIC COMPOSITION TRIM PROGRAM     TRIDYN',            1323
     C'     VERSION ',A3//1X,16A4//6X,                                  1324
     C'NH   NCP  IDOUT IQOUT IDREL IQ0 IRC0 IRAND JSP1 JSP2 JFRP JNRM')  1325
      WRITE(IO,4065)NH,NCP,IDOUT,IQOUT,IDREL,IQ0,IRC0,IRAND,JSP1I,JSP2,  1326
     CJFRP,JNRM                                                         1327
4065  FORMAT(1X,I9,I3,I8,I6,I5,I5,I4,I7,I4,3I5)                         1328
      WRITE(IO,4075)                                                    1329
4075  FORMAT(/4X,'FLC',9X,'E0',8X,'X0',4X,'ALPHA INEL INELR',           1330
     C' IWC IWCR',3X,'SHTH')                                            1331
      WRITE(IO,4085)FLC,E0,X0,ALPHA,INEL,INELR,IWC,IWCR,SHTH            1332
4085  FORMAT(2E11.4,F7.2,F8.2,I4,3I5,F9.2)                              1333
      WRITE(IO,4095)                                                    1334
4095  FORMAT(/5X,'TT',8X,'TTDYN   NQX    DSF    IQXN IQXX IMCP')       1335
      WRITE(IO,4105)TT,TTDYN,NQX,DSF,IQXN,IQXX,IMCP                     1336
4105  FORMAT(2E11.4,I4,F8.2,1X,3I5)                                     1337
      WRITE(IO,4115)                                                    1338
4115  FORMAT(/' CPT.    Z     M    SBE    BE    ED  ',                 1339
     C' EF    DNS0    CK    Q')                                        1340
      SUM = '(MAX'                                                      1341
      SUMM = '.) '                                                      1342
      DO 4390 JP=1,NCP                                                  1343
      WRITE(IO,4125)JP,ZZ(JP),M(JP),SBE(JP),BE(JP),ED(JP),EF(JP),       1344
```

```
     CDNSO(JP),CK(JP),QUINP(JP),SUM,SUMM                                1345
4125  FORMAT(1X,I2,F8.0,F7.2,4F6.2,E11.4,F5.2,F6.3,2A4)                 1346
      IF(JP.GT.1) GOTO 4390                                             1347
      IF(IQ0.NE.0) GOTO 4400                                            1348
      SUM = '(INI'                                                      1349
      SUMM = 'T.) '                                                     1350
      GOTO 4390                                                         1351
4400  SUM = '(VAR'                                                      1352
      SUMM = 'BL.)'                                                     1353
4390  CONTINUE                                                          1354
      IF(IQ0.EQ.0) GOTO 5070                                            1355
      WRITE(IO,4925)                                                    1356
4925  FORMAT(///' VARIABLE INITIAL COMPOSITION:')                       1357
      DO 5080 JP=2,NCP                                                  1358
5080  WRITE(IO,4935)JP,(QUX0(MM,JP),MM=1,NQX)                           1359
4935  FORMAT(/'  COMPONENT #',I1,':'/25(1X,20F6.3/))                    1360
5070  WRITE(IO,4135)                                                    1361
4135  FORMAT(///'  I',5X,'EPS0  A(I,J) KL(I,J) KOR(I,J) EC(I,J)'      1362
     C,'  F(I,J)')                                                     1363
      DO 4410 JP=1,NCP                                                  1364
      WRITE(IO,4145)JP,EPS0(JP),A(JP,1),KL(JP,1),KOR(JP,1),EC(JP,1),    1365
     CF(JP,1)                                                           1366
4145  FORMAT(1X,I2,F10.4,F7.4,2F8.4,F9.4,E11.4)                         1367
      DO 4410 MM=2,NCP                                                  1368
4410  WRITE(IO,4155)A(JP,MM),KL(JP,MM),KOR(JP,MM),EC(JP,MM),F(JP,MM)    1369
4155  FORMAT(12X,3F8.4,F9.4,E11.4)                                      1370
      WRITE(IO,483)MAXA,NALL,CHAMAX                                     1371
483   FORMAT(//6X,'SIMREC =',I4,5X,'TOTREC =',I9,5X,'MAXCHA = ',F5.3)   1372
      IF(IDREL.GE.0) WRITE(IO,2135)SRRC                                 1373
2135  FORMAT(//6X,'FINAL SURFACE RECESSION =',F8.3,' A')                1374
      IF(QU1MX.LT.1.) WRITE(IO,2145)REEM1                               1375
2145  FORMAT(/6X,'REEMITTED PROJECTILES   =',F8.3,' A-2')               1376
      WRITE(IO,500)IB,EB,IT,ET,IPREC                                    1377
 500 FORMAT(//6X,26HBACKSCATTERED PROJECTILES=,I8,5X,7HENERGY=,E10.4,   1378
     1 3H EV/8X,24HTRANSMITTED PROJECTILES=,I8,5X,7HENERGY=,E10.4,3H EV/ 1379
     512X,20HREMOVED PROJECTILES=,I8)                                   1380
      IBSPA = 0.                                                        1381
      EBSPA = 0.                                                        1382
      DO 4420 JP = 1,NCP                                                1383
      IBSPA = IBSPA+IBSP(JP)                                            1384
      EBSPA = EBSPA+EBSP(JP)                                            1385
4420  WRITE(IO,4165)JP,IBSP(JP),EBSP(JP)                                1386
4165  FORMAT(9X,'SPUTTERED PARTICLES(',I1,')=',I8,5X,                   1387
     C'ENERGY=',E10.4,' EV')                                            1388
      WRITE(IO,4175)IBSPA,EBSPA                                         1389
4175  FORMAT(8X,24HALL SPUTTERED PARTICLES=,I8,5X,7HENERGY=,E10.4,3H EV) 1390
       ISPINPA = 0                                                      1391
      ESPINPA = 0.                                                      1392
```

```
      ISPINSA = 0                                                       1393
      ESPINSA = 0.                                                      1394
      ISPOPA = 0                                                        1395
      ESPOPA = 0.                                                       1396
      ISPOSA = 0                                                        1397
      ESPOSA = 0.                                                       1398
      DO 4430 JP=1,NCP                                                  1399
      ISPINPA = ISPINPA+ISPINP(JP)                                      1400
      ESPINPA = ESPINPA+ESPINP(JP)                                      1401
      ISPINSA = ISPINSA+ISPINS(JP)                                      1402
      ESPINSA= ESPINSA+ESPINS(JP)                                       1403
      ISPOPA = ISPOPA+ISPOP(JP)                                         1404
      ESPOPA = ESPOPA+ESPOP(JP)                                         1405
      ISPOSA = ISPOSA+ISPOS(JP)                                         1406
      ESPOSA = ESPOSA+ESPOS(JP)                                         1407
 4430 WRITE(IO,4185)JP,ISPINP(JP),ESPINP(JP),JP,ISPINS(JP),ESPINS(JP),  1408
     CJP,ISPOP(JP),ESPOP(JP),JP,ISPOS(JP),ESPOS(JP)                     1409
 4185 FORMAT(/5X,'    ION IN , PRIMARY KO(',I1,')=',I8,5X,'ENERGY=',     1410
     CE10.4,' EV'/9X,'ION IN , SECOND. KO(',I1,')=',I8,5X,'ENERGY=',    1411
     CE10.4,' EV'/9X,'ION OUT, PRIMARY KO(',I1,')=',I8,5X,'ENERGY=',    1412
     CE10.4,' EV'/9X,'ION OUT, SECOND. KO(',I1,')=',I8,5X,'ENERGY=',    1413
     CE10.4,' EV')                                                      1414
       WRITE(IO,1320)ISPINPA,ESPINPA,ISPINSA,ESPINSA,ISPOPA,ESPOPA,ISPOS 1415
     1 A,ESPOSA                                                         1416
 1320 FORMAT(/3X,29H    ION IN , PRIMARY KO(ALL)=,I8,5X,7HENERGY=,E10.4 1417
     1 ,3H EV/7X,25HION IN , SECOND. KO(ALL)=,I8,5X7HENERGY=,E10.4,3H EV 1418
     2 /6X,26HION OUT , PRIMARY KO(ALL)=,I8,5X,7HENERGY=,E10.4,3H EV/   1419
     3 6X,26HION OUT , SECOND. KO(ALL)=,I8,5X,7HENERGY=,E10.4,3H EV)    1420
C                                                                       1421
C     RANGE AND PATHLENGTH PARAMETERS                                   1422
C                                                                       1423
      YH=NH-IB-IT                                                       1424
      IF(YH.LE.0.) GOTO 12                                              1425
      AVEX = XSUM/YH                                                    1426
      VARI = X2SUM/YH-AVEX**2                                           1427
      SIGMAX= SQRT(VARI)                                                1428
      AVEPL=PLSUM/YH                                                    1429
      AVECOL=FLOAT(ICSUM)/YH                                            1430
      AVCOL=FLOAT(ICSUM1)/YH                                            1431
      SIGMPL= SQRT(PL2SUM/YH-AVEPL**2)                                  1432
      IF(SIGMAX.EQ.0.) GOTO 12                                          1433
      U=AVEX/SIGMAX                                                     1434
      U2=U**2                                                           1435
      GAMMA = X3SUM/YH/SIGMAX**3-U*(3.+U2)                              1436
      BETA = X4SUM/YH/VARI**2-4.*U*GAMMA-U2*(6.+U2)                     1437
   12 WRITE(IO,800)AVEX,AVEPL,AVECOL,SIGMAX,SIGMPL,AVCOL,GAMMA,BETA     1438
  800 FORMAT(//5X'  AVERAGE  DEPTH  ='F8.2,5X'AVERAGE PATHLENGTH='F8.2, 1439
     15X'AVER. NUMBER COLL.='F8.1/5X'STANDARD DEVIATION='F8.2,5X'STANDAR 1440
```

```
     2D DEVIATION='F8.2,5X'AV. NR. COLL.(ED) ='F8.1/5X,' SKEWNESS GAMMA   1441
     3  ='F8.2/5X'   KURTOSIS BETA  ='F8.2)                               1442
C                                                                         1443
C     REFLECTION COEFFICIENTS AND SPUTTERING YIELDS                       1444
C                                                                         1445
      EMEAN = 0.0                                                         1446
      RN = FLOAT(IB)/FLOAT(NH)                                            1447
      RE = EB/NH/ABS(E0)                                                  1448
      WRITE(IO,810) RN,RE                                                 1449
  810 FORMAT(//5X,'REFL.COEFF.=',F8.5,'   ENERGY REFL.COEFF.=',F8.5)      1450
      SPY=FLOAT(IBSPA)/FLOAT(NH)                                          1451
      SPE=EBSPA/NH/ABS(E0)                                                1452
      WRITE(IO,830) SPY,SPE                                               1453
  830 FORMAT(/5X,'SPUTTERING YIELD(ALL) = ',1PE9.3,                       1454
     C'  SPUTTERED ENERGY(ALL) = ',E9.3)                                  1455
      WRITE(IO,834)FPKTOT                                                 1456
834   FORMAT(/5X,'FRENKEL PAIRS (K/P) =',E11.4)                           1457
      WRITE(IO,833)(I,BBRTOT(I),I=1,NCP)                                  1458
833   FORMAT(5X,'FRENKEL PAIRS (',I1,')  =',E11.4)                        1459
      WRITE(IO,832)IONTOT,NCLTOT,DMGTOT,PHOTOT                            1460
832   FORMAT(/8X,'INELASTIC ENERGY LOSS =',E11.4,' EV',                   1461
     C10X,'ELASTIC ENERGY LOSS =',E11.4,' EV'/6X,'ENERGY LOSS INTO',      1462
     C' DAMAGE =',E11.4,' EV',5X,'ENERGY LOSS INTO PHONONS =',E11.4,      1463
     C' EV'/)                                                             1464
C                                                                         1465
C     DEPOSITION PROFILES                                                 1466
C                                                                         1467
485   FORMAT(/////10X,'PSEUDOPROJECTILE DEPOSITION PROFILES:')            1468
      WRITE(IO,600)                                                       1469
  600 FORMAT(///4X,8HDEPTH(A),2X,9HPARTICLES,1X,6HPATHL.,                 1470
     12X,15HFRENKEL P.(K/P),2X,10HELECTR(EV),4X,11HNUCLEAR(EV),5X,        1471
     C10HDAMAGE(EV),4X,11HPHONONS(EV)/)                                   1472
      D1=0.                                                               1473
      D2=DQX                                                              1474
      DO 19 MM=NQX,1,-1                                                   1475
19    IF(IPL(MM).NE.0) GOTO 20                                            1476
      MM = 1                                                              1477
20    MM = MM+2                                                           1478
      DO 11 I=1,MM                                                        1479
      WRITE(IO,700) D1,D2,IRP(I),IPL(I),FPKP(I),ION(I),NCLR(I),DMGN(I),   1480
     CPHON(I)                                                             1481
  700 FORMAT(1X,F6.0,1H-,F5.0,2I8,5E15.4)                                 1482
      D1=D2                                                               1483
   11 D2=D2+DQX                                                           1484
C                                                                         1485
C     INTERPOLATION AND OUTPUT OF DEPOSITION PROFILES                     1486
C                                                                         1487
      DO 4470 MM=1,NQX                                                    1488
```

```
      IF(IDREL.NE.0) GOTO 4472                                          1489
      XX0 = (MM-1)*DQX                                                  1490
      XX1 = MM*DQX                                                      1491
      DO 4222 JP=1,NCP                                                  1492
 4222 SSUM(JP) = 0.                                                     1493
      XXX0 = 0.                                                         1494
      DO 2162 MMM=1,NQX                                                 1495
      XXX0 = XXX0+XXX(MMM)                                              1496
 2162 IF(XXX0.GT.XX0) GOTO 2172                                         1497
      DO 4242 JP=1,NCP                                                  1498
 4242 SSUM(JP) = QUXB(JP)*DQX*DNSB                                      1499
      GOTO 2092                                                         1500
 2172 DO 4252 JP=1,NCP                                                  1501
 4252 SSUM(JP) = SSUM(JP)+(XXX0-XX0)/XXX(MMM)*DEL(MMM,JP)               1502
      IF(XXX0-XX1)2182,2092,2202                                        1503
 2182 MMM = MMM+1                                                       1504
      IF(MMM.LE.NQX) GOTO 2212                                          1505
      DO 4262 JP=1,NCP                                                  1506
 4262 SSUM(JP) = SSUM(JP)+(XX1-XXX0)*QUXB(JP)*DNSB                      1507
      GOTO 2092                                                         1508
 2212 XX0 = XXX0                                                        1509
      XXX0 = XXX0+XXX(MMM)                                              1510
      GOTO 2172                                                         1511
 2202 DO 4272 JP=1,NCP                                                  1512
 4272 SSUM(JP) = SSUM(JP)-(XXX0-XX1)/XXX(MMM)*DEL(MMM,JP)               1513
 2092 SUM = 0.                                                          1514
      DO 4282 JP=1,NCP                                                  1515
 4282 SUM = SUM+SSUM(JP)                                                1516
      DO 4292 JP=1,NCP                                                  1517
 4292 QUX(MM,JP) = SSUM(JP)/SUM                                         1518
 4472 DNS(MM) = 0.                                                      1519
      SUM = 0.                                                          1520
      DO 4294 JP=JNRM,NCP                                               1521
      DNS(MM) = DNS(MM)+QUX(MM,JP)/DNS0(JP)                             1522
 4294 SUM = SUM+QUX(MM,JP)                                              1523
      DNS(MM) = SUM/DNS(MM)                                             1524
 4470 FACT(MM) = FLC/NH/DNS(MM)/DQX                                     1525
      DO 4770 MM=1,NQX                                                  1526
      RION(MM) = IRP(MM)                                                1527
 4770 XXX(MM) = (MM-.5)*DQX*1.E-8                                       1528
      WRITE(IO2,4775)TITLE                                              1529
 4775 FORMAT(' Range Distrib. ',16A4)                                   1530
      CALL TRUNC(RION,FACT,NQX,N)                                       1531
      WRITE(IO2,4205)(XXX(MM),RION(MM),MM=1,N)                          1532
      DO 4780 MM=1,NQX                                                  1533
 4780 RION(MM) = IPL(MM)                                                1534
      WRITE(IO2,4785)TITLE                                              1535
 4785 FORMAT(' Pathlength D.  ',16A4)                                   1536
```

```
      CALL TRUNC(RION,FACT,NQX,N)                                       1537
      WRITE(IO2,4205)(XXX(MM),RION(MM),MM=1,N)                          1538
      WRITE(IO2,4795)TITLE                                              1539
 4795 FORMAT(' Frenkel P.(K/P)',16A4)                                   1540
      CALL TRUNC(FPKP,FACT,NQX,N)                                       1541
      WRITE(IO2,4205)(XXX(MM),FPKP(MM),MM=1,N)                          1542
      WRITE(IO2,4805)TITLE                                              1543
 4805 FORMAT(' Electr. Losses ',16A4)                                   1544
      CALL TRUNC(ION,FACT,NQX,N)                                        1545
      WRITE(IO2,4205)(XXX(MM),ION(MM),MM=1,N)                           1546
      WRITE(IO2,4815)TITLE                                              1547
 4815 FORMAT(' Nuclear Losses ',16A4)                                   1548
      CALL TRUNC(NCLR,FACT,NQX,N)                                       1549
      WRITE(IO2,4205)(XXX(MM),NCLR(MM),MM=1,N)                          1550
      WRITE(IO2,4825)TITLE                                              1551
 4825 FORMAT(' Damage Losses ',16A4)                                    1552
      CALL TRUNC(DMGN,FACT,NQX,N)                                       1553
      WRITE(IO2,4205)(XXX(MM),DMGN(MM),MM=1,N)                          1554
      WRITE(IO2,4835)TITLE                                              1555
 4835 FORMAT(' Phonon Losses  ',16A4)                                   1556
      CALL TRUNC(PHON,FACT,NQX,N)                                       1557
      WRITE(IO2,4205)(XXX(MM),PHON(MM),MM=1,N)                          1558
      DO 4830 JP=1,NCP                                                  1559
      WRITE(IO2,4845)JP,TITLE                                           1560
 4845 FORMAT(' Cpt',I1,' Frenkel P.',16A4)                              1561
      CALL TRUNC(BBR(1,JP),FACT,NQX,N)                                  1562
 4830 WRITE(IO2,4205)(XXX(MM),BBR(MM,JP),MM=1,N)                        1563
 4205 FORMAT(2E15.6)                                                    1564
      STOP                                                              1565
      END                                                               1566
C                                                                       1567
C                                                                       1568
C                                                                       1569
      SUBROUTINE TRUNC(A,B,NO,N)                                        1570
      DIMENSION A(1),B(1)                                               1571
      DO 10 I=NO,3,-1                                                   1572
   10 IF(A(I).NE.0.) GOTO 20                                            1573
      I = 3                                                             1574
   20 N = MIN0(I+2,NO)                                                  1575
      DO 30 I=1,N                                                       1576
   30 A(I) = A(I)*B(I)                                                  1577
      RETURN                                                            1578
      END                                                               1579
C                                                                       1580
C                                                                       1581
C                                                                       1582
      SUBROUTINE DIRCOS(COSX,COSY,COSZ,SINE,PSI,PHI)                    1583
      CPHI=COS(PHI)                                                     1584
```

```
      SPHI=SIN(PHI)                                                     1585
      CPSI=COS(PSI)                                                     1586
      SPSI=SIN(PSI)                                                     1587
      SRAT=SPSI/SINE                                                    1588
      CX2=CPSI*COSX+SPSI*SINE*CPHI                                      1589
      CY2=CPSI*COSY-SRAT*(COSY*COSX*CPHI-COSZ*SPHI)                     1590
      CZ2=CPSI*COSZ-SRAT*(COSZ*COSX*CPHI+COSY*SPHI)                     1591
      UNIT = 1.0/SQRT(CX2**2+CY2**2+CZ2**2)                             1592
      COSX=CX2*UNIT                                                     1593
      COSY=CY2*UNIT                                                     1594
      COSZ=SIGN(ABS(CZ2*UNIT)+1.E-12,CZ2)                               1595
      SINE=SQRT(COSY*COSY+COSZ*COSZ)                                    1596
      RETURN                                                            1597
      END                                                               1598
C                                                                       1599
C                                                                       1600
C                                                                       1601
      SUBROUTINE DIRCOSV(COSX,COSY,COSZ,SINE,PSI,PHI,N)                 1602
      DIMENSION PHI(N),PSI(N),COSX(N),COSY(N),COSZ(N),SINE(N)           1603
      DO 1 IV=1,N                                                       1604
      CPHI=COS(PHI(IV))                                                 1605
      SPHI=SIN(PHI(IV))                                                 1606
      CPSI=COS(PSI(IV))                                                 1607
      SPSI=SIN(PSI(IV))                                                 1608
      SRAT=SPSI/SINE(IV)                                                1609
      CX2=CPSI*COSX(IV)+SPSI*SINE(IV)*CPHI                              1610
      CY2=CPSI*COSY(IV)-SRAT*(COSY(IV)*COSX(IV)*CPHI-COSZ(IV)*SPHI)     1611
      CZ2=CPSI*COSZ(IV)-SRAT*(COSZ(IV)*COSX(IV)*CPHI+COSY(IV)*SPHI)     1612
      UNIT = 1.0/SQRT(CX2**2+CY2**2+CZ2**2)                             1613
      COSX(IV)=CX2*UNIT                                                 1614
      COSY(IV)=CY2*UNIT                                                 1615
      COSZ(IV)=SIGN(ABS(CZ2*UNIT)+1.E-12,CZ2)                           1616
      SINE(IV)=SQRT(COSY(IV)*COSY(IV)+COSZ(IV)*COSZ(IV))                1617
    1 CONTINUE                                                          1618
      RETURN                                                            1619
      END                                                               1620
C                                                                       1621
C                                                                       1622
C                                                                       1623
      SUBROUTINE MXVELO(E,COSX,COSY,COSZ,SINE)                          1624
C                                                                       1625
C     FETCH ENERGY AND VELOCITY FROM A MAXWELLIAN FLUX AT A SURFACE     1626
C                                                                       1627
      REAL M1                                                           1628
      COMMON/A/ M1,VELC,ZARG                                            1629
      DATA PI2/6.283185/                                                1630
C                                                                       1631
      AR = ALOG(RANF())                                                 1632
```

```
      VELX=SQRT(-2.*AR)*ZARG                                          1633
      ZZ = PI2*RANF()                                                 1634
      ZSIN = SIN(ZZ)                                                  1635
      ZCOS = COS(ZZ)                                                  1636
      AR = ALOG(RANF())                                               1637
      ZT = SQRT(-2.*AR)                                               1638
      VELY = ZT*ZCOS*ZARG                                             1639
      VELZ = ZT*ZSIN*ZARG                                             1640
      IF(VELC.GT.0.) VELX = SQRT(VELC+VELX**2)                        1641
      VELQ=VELX*VELX+VELY*VELY+VELZ*VELZ                              1642
      VEL=SQRT(VELQ)                                                  1643
      COSX=VELX/VEL                                                   1644
      COSY=VELY/VEL                                                   1645
      COSZ=VELZ/VEL                                                   1646
      COSZ = SIGN(ABS(COSZ)+1.E-12,COSZ)                              1647
      SINE=SQRT(COSY*COSY+COSZ*COSZ)                                  1648
      E=M1*VELQ                                                       1649
      RETURN                                                          1650
      END                                                             1651
C                                                                     1652
C                                                                     1653
C                                                                     1654
```

Appendix 2

Test Run Input

(Channel 5)

```
1 keV Ne->Ta205      60000 Hist. 6e16/cm2                                    1655
     60000 1200 6000     3     0     0    -1 2003     0     1     2     2    1656
  0.600E+1    1000.00        0.00        0.00     1     3     2     2         1657
  0.100E+5  0.200E+3    80  0.500E+1      0  100     1                       1658
  10.00  20.18   0.00   0.00   0.00   5.00  0.200  0.100E+0 1.00             1659
  73.00 180.95   8.10   0.00  50.00   8.00  0.286  0.554E-1 1.00             1660
   8.00  16.01   6.70   0.00  50.00   6.50  0.714  0.809E-1 1.00             1661
```

Appendix 3

Test Run Output

(Channel 6)

1 keV Ne->Ta205 60000 Hist. 6e16/cm2

NH	NCP	IDOUT	IQOUT	IDREL	IQ0	IRC0	IRAND	JSP1	JSP2	JFRP	JNRM
60000	3	1200	6000	0	0	-1	2003	0	0	2	2

FLC	E0	X0	ALPHA	INEL	INELR	IWC	IWCR	SHTH
0.6000E+01	0.1000E+04	0.00	0.00	1	3	2	2	0.00

TT	TTDYN	NQX	DSF	IQXN	IQXX	IMCP
0.1000E+05	0.2000E+03	80	5.00	1	80	1

CPT.	Z	M	SBE	BE	ED	EF	DNS0	CK	Q
1	10.	20.18	0.00	0.00	0.00	5.00	0.1000E+00	1.00	0.200(MAX.)
2	73.	180.95	8.10	0.00	50.00	8.00	0.5540E-01	1.00	0.286(INIT.)
3	8.	16.01	6.70	0.00	50.00	6.50	0.8090E-01	1.00	0.714(INIT.)

I	EPS0	A(I,J)	KL(I,J)	KOR(I,J)	EC(I,J)	F(I,J)
1	0.0475	0.1369	1.4001	0.9250	1.0000	0.4755E-04
		0.0908	2.7810	4.1753	0.3611	0.7774E-05
		0.1420	1.2471	0.7664	0.9867	0.5452E-04
2	0.0078	0.0908	1.2935	1.9420	0.3611	0.8670E-06
		0.0706	4.7540	11.8188	1.0000	0.4600E-06
		0.0926	1.0815	1.5623	0.2987	0.8951E-06
3	0.0545	0.1420	1.3490	0.8291	0.9867	0.6872E-04
		0.0926	2.5153	3.6334	0.2987	0.1012E-04
		0.1475	1.2116	0.6898	1.0000	0.8003E-04

SIMREC = 35 TOTREC = 8390463 MAXCHA = 0.015

FINAL SURFACE RECESSION = 33.726 A

REEMITTED PROJECTILES = 2.959 A-2

BACKSCATTERED PROJECTILES=	9639	ENERGY=0.3720E+07 EV
TRANSMITTED PROJECTILES=	0	ENERGY=0.0000E+00 EV
REMOVED PROJECTILES=	17584	
SPUTTERED PARTICLES(1)=	12909	ENERGY=0.4959E+06 EV
SPUTTERED PARTICLES(2)=	7310	ENERGY=0.1805E+06 EV
SPUTTERED PARTICLES(3)=	22968	ENERGY=0.1186E+07 EV
ALL SPUTTERED PARTICLES=	43187	ENERGY=0.1863E+07 EV
ION IN , PRIMARY KO(1)=	2666	ENERGY=0.1735E+06 EV
ION IN , SECOND. KO(1)=	5554	ENERGY=0.1073E+06 EV
ION OUT, PRIMARY KO(1)=	2205	ENERGY=0.1668E+06 EV
ION OUT, SECOND. KO(1)=	2484	ENERGY=0.4835E+05 EV
ION IN , PRIMARY KO(2)=	537	ENERGY=0.1761E+05 EV
ION IN , SECOND. KO(2)=	2584	ENERGY=0.3613E+05 EV
ION OUT, PRIMARY KO(2)=	2153	ENERGY=0.8932E+05 EV
ION OUT, SECOND. KO(2)=	2036	ENERGY=0.3739E+05 EV
ION IN , PRIMARY KO(3)=	6367	ENERGY=0.4863E+06 EV
ION IN , SECOND. KO(3)=	8203	ENERGY=0.1992E+06 EV
ION OUT, PRIMARY KO(3)=	4866	ENERGY=0.4144E+06 EV
ION OUT, SECOND. KO(3)=	3532	ENERGY=0.8656E+05 EV

ION IN , PRIMARY KO(ALL)= 9570 ENERGY=0.6775E+06 EV
ION IN , SECOND. KO(ALL)= 16341 ENERGY=0.3426E+06 EV
ION OUT , PRIMARY KO(ALL)= 9224 ENERGY=0.6705E+06 EV
ION OUT , SECOND. KO(ALL)= 8052 ENERGY=0.1723E+06 EV

AVERAGE DEPTH = 26.26 AVERAGE PATHLENGTH= 54.08 AVER. NUMBER COLL.= 25.0
STANDARD DEVIATION= 14.27 STANDARD DEVIATION= 17.02 AV. NR. COLL.(ED) = 10.0
SKEWNESS GAMMA = 0.57
KURTOSIS BETA = 3.09

REFL.COEFF.= 0.16065 ENERGY REFL.COEFF.= 0.06200

SPUTTERING YIELD(ALL) = 7.198E-01 SPUTTERED ENERGY(ALL) = 3.105E-02

FRENKEL PAIRS (K/P) = 0.2416E+06
FRENKEL PAIRS (1) = 0.1622E+07
FRENKEL PAIRS (2) = 0.8877E+05
FRENKEL PAIRS (3) = 0.2158E+06

INELASTIC ENERGY LOSS = 0.8062E+07 EV ELASTIC ENERGY LOSS = 0.4813E+08 EV
ENERGY LOSS INTO DAMAGE = 0.3769E+08 EV ENERGY LOSS INTO PHONONS = 0.1044E+08 EV

DEPTH(A)	PARTICLES	PATHL.	FRENKEL P.(K/P)	ELECTR(EV)	NUCLEAR(EV)	DAMAGE(EV)	PHONONS(EV)
0.- 2.	2575	0	0.1417E+05	0.5137E+06	0.2839E+07	0.2246E+07	0.5934E+06
2.- 5.	2701	6	0.1353E+05	0.4648E+06	0.2742E+07	0.2166E+07	0.5756E+06
5.- 7.	2759	25	0.1199E+05	0.4083E+06	0.2477E+07	0.1933E+07	0.5437E+06
7.- 10.	2618	72	0.1082E+05	0.3541E+06	0.2227E+07	0.1730E+07	0.4962E+06
10.- 12.	2537	103	0.9510E+04	0.3018E+06	0.1981E+07	0.1523E+07	0.4583E+06
12.- 15.	2400	166	0.8362E+04	0.2543E+06	0.1744E+07	0.1330E+07	0.4142E+06
15.- 17.	2341	261	0.7292E+04	0.2110E+06	0.1529E+07	0.1155E+07	0.3739E+06
17.- 20.	2158	367	0.6320E+04	0.1718E+06	0.1313E+07	0.9814E+06	0.3312E+06
20.- 22.	1909	497	0.5281E+04	0.1383E+06	0.1102E+07	0.8161E+06	0.2861E+06
22.- 25.	1791	623	0.4494E+04	0.1099E+06	0.9320E+06	0.6875E+06	0.2445E+06
25.- 27.	1593	799	0.3646E+04	0.8700E+05	0.7749E+06	0.5627E+06	0.2122E+06
27.- 30.	1372	1008	0.2932E+04	0.6734E+05	0.6172E+06	0.4427E+06	0.1745E+06
30.- 32.	1202	1233	0.2260E+04	0.5292E+05	0.4860E+06	0.3406E+06	0.1454E+06
32.- 35.	1002	1503	0.1758E+04	0.4166E+05	0.3872E+06	0.2641E+06	0.1231E+06
35.- 37.	847	1730	0.1399E+04	0.3158E+05	0.3020E+06	0.2028E+06	0.9914E+05
37.- 40.	669	2013	0.1058E+04	0.2443E+05	0.2303E+06	0.1514E+06	0.7887E+05
40.- 42.	551	2276	0.7212E+03	0.1811E+05	0.1667E+06	0.1043E+06	0.6244E+05
42.- 45.	432	2562	0.6484E+03	0.1346E+05	0.1335E+06	0.8666E+05	0.4685E+05
45.- 47.	368	2653	0.4481E+03	0.1010E+05	0.9719E+05	0.5906E+05	0.3814E+05
47.- 50.	262	2829	0.3298E+03	0.7174E+04	0.6831E+05	0.4140E+05	0.2692E+05
50.- 52.	180	2841	0.2123E+03	0.5021E+04	0.4659E+05	0.2533E+05	0.2127E+05
52.- 55.	158	2911	0.1527E+03	0.3452E+04	0.3309E+05	0.1833E+05	0.1475E+05
55.- 57.	106	2930	0.1132E+03	0.2403E+04	0.2342E+05	0.1225E+05	0.1116E+05
57.- 60.	69	2847	0.8421E+02	0.1627E+04	0.1555E+05	0.7833E+04	0.7718E+04
60.- 62.	62	2673	0.4852E+02	0.1208E+04	0.1129E+05	0.5357E+04	0.5933E+04
62.- 65.	38	2516	0.2801E+02	0.6692E+03	0.6366E+04	0.2704E+04	0.3663E+04
65.- 67.	25	2226	0.1830E+02	0.4799E+03	0.3858E+04	0.1680E+04	0.2179E+04
67.- 70.	16	1961	0.9224E+01	0.3037E+03	0.2340E+04	0.8970E+03	0.1443E+04
70.- 72.	12	1747	0.1206E+02	0.2444E+03	0.2178E+04	0.9430E+03	0.1235E+04
72.- 75.	5	1437	0.2000E+01	0.1774E+03	0.1023E+04	0.1305E+03	0.8926E+03
75.- 77.	7	1196	0.3000E+01	0.9537E+02	0.7024E+03	0.2322E+03	0.4703E+03
77.- 80.	2	994	0.1000E+01	0.7566E+02	0.3595E+03	0.6211E+02	0.2974E+03
80.- 82.	1	826	0.2000E+01	0.5345E+02	0.4852E+03	0.1718E+03	0.3133E+03

82.- 85.	5	651	0.4118E+01	0.3697E+02	0.5537E+03	0.3647E+03	0.1890E+03
85.- 87.	2	479	0.0000E+00	0.3828E+02	0.3247E+03	0.0000E+00	0.3247E+03
87.- 90.	0	389	0.0000E+00	0.8001E+01	0.1613E+02	0.0000E+00	0.1613E+02
90.- 92.	0	291	0.0000E+00	0.9183E+01	0.5141E+02	0.0000E+00	0.5141E+02
92.- 95.	1	197	0.9025E+00	0.6527E+01	0.1515E+03	0.1289E+03	0.2253E+02
95.- 97.	1	149	0.0000E+00	0.7312E+01	0.9985E+02	0.0000E+00	0.9985E+02
97.- 100.	0	112	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
100.- 102.	0	84	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
102.- 105.	0	58	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
105.- 107.	0	37	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
107.- 110.	0	28	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
110.- 112.	0	18	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
112.- 115.	0	12	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
115.- 117.	0	10	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
117.- 120.	0	7	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
120.- 122.	0	5	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
122.- 125.	0	1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
125.- 127.	0	1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
127.- 130.	0	1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
130.- 132.	0	0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
132.- 135.	0	0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00

/EOF