

**Idaho
National
Engineering
Laboratory**

*Managed
by the U.S.
Department
of Energy*

**EGG-FSP-10960
SEPTEMBER 1993**

**RECEIVED
MAY 09 1994
OSTI**

**SIMPLIFIED TRITIUM PERMEATION
MODEL**

G. R. Longhurst



*Work performed under
DOE Contract
No. DE-AC07-76ID01570*

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

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

SIMPLIFIED TRITIUM PERMEATION MODEL

Glen R. Longhurst
Idaho National Engineering Laboratory
Idaho Falls, Idaho, 83415

September 17, 1993

Prepared for the
U.S. Department of Energy
Office of Energy Research
under DOE Idaho Field Office
Contract DE-AC07-76ID01570

MASTER

INTRODUCTION

In this model I seek to provide a simplified approach to solving permeation problems addressed by ΓMAP4.^{1,2} I will assume that there are m one-dimensional segments with thickness L_i , $i = 1, 2, \dots, m$, joined in series with an implantation flux, J_i , implanting at the single depth, δ , in the first segment. From material properties and heat transfer considerations, I calculate temperatures at each face of each segment, and from those temperatures I find local diffusivities and solubilities. I assume recombination coefficients K_{r_1} and K_{r_2} are known at the upstream and downstream faces, respectively, but the model will generate Baskes³ recombination coefficient values on demand. Here I first develop the steady-state concentration equations and then show how trapping considerations can lead to good estimates of permeation transient times.

BASIC TRANSPORT MODEL

The physical situation addressed by this model is shown in Figure 1. The fundamental equation governing tritium transport in a material is

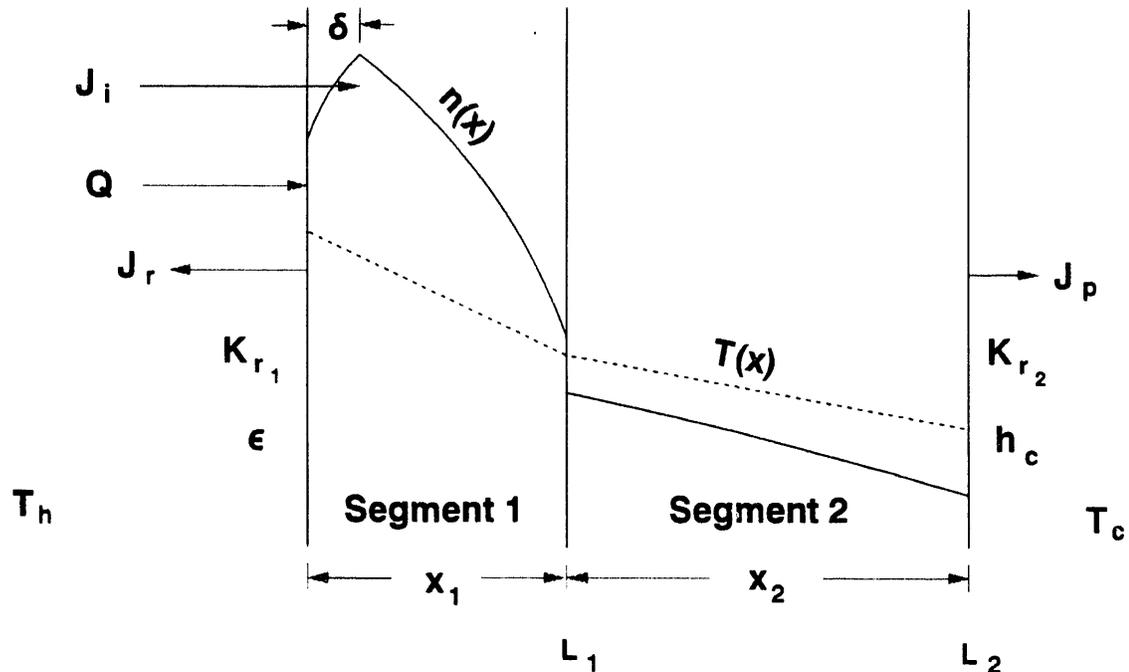


Figure 1. Conceptualization of quantities and relationships used in the present model.

$$N \frac{\partial}{\partial t} (n + n_t) = \nabla \cdot \mathbf{J} \quad (1)$$

where n represents the mobile tritium concentration in the material, and n_t is the local trapped atom concentration. \mathbf{J} is the flux of tritium atoms through the material given by

$$\mathbf{J} = -DN \left[\nabla n + n \frac{Q^*}{RT^2} \nabla T \right] \quad (2)$$

Here D is the tritium diffusivity in the material, given by the standard Arrhenius expression, n is the tritium atom fraction in the material, and Q^* is the heat of transport. N is the number density of the host lattice atoms.

For the steady-state solution, the fundamental relation is

$$\nabla \cdot \mathbf{J} = 0 \quad (3)$$

Keeping in mind that D is temperature dependent and that on any segment ∇T will be a constant, Eqs.(2) and (3) may be combined to give

$$\nabla^2 n + \frac{E_d + Q^*}{RT^2} \nabla T \nabla n + \frac{E_d - 2RT}{(RT^2)^2} Q^* (\nabla T)^2 n = 0 \quad (4)$$

Now with the definitions

$$\beta = \frac{E_d + Q^*}{2R\langle T \rangle^2} \nabla T \quad (5)$$

and

$$\gamma = \frac{E_d - 2R\langle T \rangle}{(R\langle T \rangle^2)^2} Q^* (\nabla T)^2 \quad (6)$$

Eq.(4) becomes

$$\nabla^2 n + 2\beta \nabla n + \gamma n = 0 \quad (7)$$

which, if the coefficients can be considered constants, is easily solved as

$$n(x) = C_1 \exp(\lambda_1 x) + C_2 \exp(\lambda_2 x) \quad (8)$$

$$n(x) = \exp(-\beta x) [C_1 \cos(qx) + C_2 \sin(qx)] \quad (9)$$

or

$$n(x) = (C_1 + C_2 x) \exp(-\beta x) \quad (10)$$

depending on whether the λ_i are real and distinct (Type I), complex conjugates (Type II), or real and equal (TYPE III), respectively. For sake of simplification, I adopt constant values of the material property coefficients, β and γ , based on the average temperature in each segment,

$$\langle T \rangle = \sqrt{T_1 T_2} \quad (11)$$

where T_1 and T_2 are the upstream and downstream face temperatures for that segment, respectively. In the above equations C_1 and C_2 are constants that depend on the boundary conditions, and

$$\lambda_1 = -\beta + \sqrt{\beta^2 - \gamma} \quad (12)$$

$$\lambda_2 = -\beta - \sqrt{\beta^2 - \gamma} \quad (13)$$

and

$$q = \sqrt{\gamma - \beta^2} \quad (14)$$

BOUNDARY CONDITIONS

There are really $m+1$ regions requiring boundary conditions. The first segment is divided into two regions, one upstream and one downstream of $x = \delta$. I will refer to the constants on the short upstream segment as \hat{C}_1 and \hat{C}_2 . At the upstream face ($x = 0$), recombination is assumed to be second order, and a constraint is that

$$\mu K_r n_1^2 N_1^2 = D_1 N_1 \left[\nabla n|_{x=0} + \frac{n_1 Q^*}{R \langle T \rangle^2} \nabla T \right] \quad (15)$$

where

$$\mu \equiv 2 + \sqrt{\frac{6}{5} \frac{1 - \phi}{\phi}} \quad (16)$$

accounts for DT molecule formation as well as T_2 for tritium fraction, ϕ , and D_1 is the diffusivity evaluated at $x_1 = 0$. Note that this assumes Sieverts' law solution and, therefore, will generally not apply to oxide films. For convenience, define

$$\eta_i \equiv \frac{Q_i^*}{R \langle T \rangle_i^2} \nabla T_i \quad (17)$$

Then, when the functional forms of n and ∇n from Eqs.(8) - (10) are put into Eq.(15) and $x \rightarrow 0$, then there results for Type I, II, or III solutions, respectively

$$\frac{N_1 \mu K_r}{D_1} (\hat{C}_1 + \hat{C}_2)^2 - (\lambda_1 + \eta_1) \hat{C}_1 - (\lambda_2 + \eta_1) \hat{C}_2 = 0 \quad (18)$$

$$\frac{N_1 \mu K_r}{D_1} \hat{C}_1^2 + (\beta_1 - \eta_1) \hat{C}_1 - q \hat{C}_2 = 0 \quad (19)$$

or

$$\frac{N_1 \mu K_{r_1} \hat{C}_1^2}{D_1} + (\beta_1 - \eta_1) \hat{C}_1 - \hat{C}_2 = 0 \quad (20)$$

At the $x = \delta$ interface in segment 1, the chemical potentials are equal. This is expressed by

$$n(\delta^-) = n(\delta^+) \quad (21)$$

where δ^- and δ^+ refer to the approaches to the $x = \delta$ surface from the upstream and downstream directions, respectively. There must also be a flux continuity at $x = \delta$ such that $J(\delta^+) - J(\delta^-) = \phi J_i$.

After some rearrangement I get for this implantation-depth interface boundary condition

$$\frac{\phi J_i}{N_1 D_1} + (C_1 - \hat{C}_1) \lambda_1 \exp(\lambda_1 \delta) + (C_2 - \hat{C}_2) \lambda_2 \exp(\lambda_2 \delta) = 0 \quad (23)$$

$$\begin{aligned} \frac{\phi J_i}{N_1 D_1} \exp(\beta_1 \delta) + \cos(q_1 \delta) [q(C_2 - \hat{C}_2) - \beta(C_1 - \hat{C}_1)] \\ + \sin(q_1 \delta) [q(\hat{C}_1 - C_1) + \beta(\hat{C}_2 - C_2)] = 0 \end{aligned} \quad (24)$$

and

$$\frac{\phi J_i}{N_1 D_1} \exp(\beta_1 \delta) + C_2 - \hat{C}_2 + \beta_1 [\hat{C}_1 - C_1 + \delta(\hat{C}_2 - C_2)] = 0 \quad (25)$$

for the various solution types, respectively.

Now at the interfaces between the segments, the same basic equations for continuity of chemical potential and flux exist. Because the solution may be of different type from one segment to another, these relations are best found for each type in segment $i+1$ as functions of the corresponding concentrations or gradients in the i th segment. For concentration

$$n_{i+1} = n_i \frac{S_{1,i+1}}{S_{2,i}} \quad (26)$$

and flux continuity requires

$$N_1 D_2 (\nabla n + \eta n)_i = N_{i+1} D_{1,i+1} (\nabla n + \eta n)_{i+1} \quad (27)$$

Note that x is reset to zero at the upstream face of each segment.

Specifically, for Type I segments, these conditions are reflected by and

$$C_{2,i} = \frac{N_i D_2 (\nabla n_i + \eta_i n_i) - n_i \frac{S_{1,i}}{S_2} (\eta_{i+1} + \lambda_{1,i})}{\lambda_{1,i} - \lambda_{2,i}} \quad (28)$$

$$C_{1,i} = n_i \frac{S_{1,i}}{S_2} - C_{2,i} \quad (29)$$

For Type II segments

$$C_{1,i} = n_i \frac{S_{1,i}}{S_2} \quad (30)$$

and

$$C_{2,i} = \frac{N_i D_2 (\nabla n_i + \eta_i n_i) + C_{1,i} (\beta_{i+1} - \eta_{i+1})}{q_{i+1}} \quad (31)$$

For Type III segments

$$C_{1,i} = n_i \frac{S_{1,i}}{S_2} \quad (32)$$

and

$$C_{2,i} = \frac{N_i D_2 (\nabla n_i + \eta_i n_i) + C_{1,i} (\beta_{i+1} - \eta_{i+1})}{N_{i+1} D_{1,i}} \quad (33)$$

At the downstream face, recombination is assumed to be again second order such that the permeation flux is given by

$$J_p = \mu K_r n_2^2 N_m^2 \quad (34)$$

Similarly at the upstream face, the reemission flux is

$$J_r = \mu K_r n_1^2 N_1^2 \quad (35)$$

Then mass conservation requires that

$$\phi J_i - J_p - J_r = 0 \quad (36)$$

SOLUTION ALGORITHM

The approach taken for solution is to estimate the fraction, F , of tritium implanted that permeates at the downstream face. That leads to an estimate of \hat{C}_1 on the basis that $(1-F)$ of the implantation flux returns by diffusion to the upstream face as reemission flux. Then with the additional condition that the diffusive flux of $(1-F)\phi J_i$ to the surface must recombine in a second-order process, it may be shown that for Type I problems

$$\hat{C}_1 = \frac{\frac{(1-F)\phi J_i}{N_1 D_1} - (\lambda_2 + \eta_1) \sqrt{\frac{(1-F)\phi J_i}{\mu K_r N_1^2}}}{\lambda_1 - \lambda_2} \quad (37)$$

The corresponding function for both Types II and III is

$$\hat{C}_1 = \sqrt{\frac{(1-F)\phi J_i}{\mu K_r N_1^2}} \quad (38)$$

Then I use the equations above to estimate in turn \hat{C}_2 , C_1 , C_2 , and for successive segments beyond the first, C_1 and C_2 . That gives a temporary definition for each of the concentration profiles in each segment. The concentration at the downstream edge may then be used with Eq.(34) evaluated at $x = L_m$ to get the permeating flux. Then

$$\Psi \equiv F - \frac{J_p}{\phi J_i} \quad (39)$$

becomes an error function that can be zeroed. Because of the second-order recombination process at the outside surfaces, it is found that using n times its absolute value in evaluating J_p with Eq.(34) makes possible negative values of Ψ for F between 0 and 1 where that zero occurs.

An important part of the process of finding the concentration equation coefficients is the step of obtaining \hat{C}_2 from \hat{C}_1 . This is done with the aid of Eqs.(18), (19), or (20), depending on the solution type. It is straightforward for Type II and Type III cases, but for Type I, Eq.(18) is a quadratic with some interesting implications. If Eq.(18) is solved for \hat{C}_2 and

$$g \equiv \frac{D_1}{2\mu K_r N_1} (\lambda_2 + \eta_1) \quad (40)$$

is defined for convenience, there results

$$\hat{C}_2 = g \left[1 \pm \sqrt{2 \frac{\hat{C}_1 (\lambda_1 - \lambda_2)}{g (\lambda_2 + \eta_1)} + 1} \right] - \hat{C}_1 \quad (41)$$

Since \hat{C}_1 and \hat{C}_2 must be real, a requirement must then be that

$$\hat{C}_1 \geq \frac{-g(\lambda_2 + \eta_1)}{2(\lambda_1 - \lambda_2)} = \frac{-D_1 (\lambda_2 + \eta_1)^2}{4\mu K_r N_1 (\lambda_1 - \lambda_2)} \quad (42)$$

This provides a negative lower bound for values of \hat{C}_1 . The requirement that $\eta_1 \geq 0$ will be met if

$$g \left[1 \pm \sqrt{2 \frac{\hat{C}_1 (\lambda_1 - \lambda_2)}{g (\lambda_2 + \eta_1)} + 1} \right] \geq 0 \quad (43)$$

Depending on the algebraic sign of the heat of transport and its magnitude relative to the diffusion activation energy, E_d , and on the temperature, g may be negative. If $g \leq 0$ then only the negative sign in the solution of Eq.(43) will give a positive value for the left side, and there is a further requirement that

$$2 \frac{\hat{C}_1 (\lambda_1 - \lambda_2)}{g (\lambda_2 + \eta_1)} \geq 0 \quad (44)$$

which will be met if $\hat{C}_1 \geq 0$ (note that g and $(\lambda_2 + \eta_1)$ are always of the same sign). If $g \geq 0$ then the positive sign in Eq.(43) will always give a positive value of n_1 . The negative sign solution will also give a positive value provided that

$$-1 \leq 2 \frac{\hat{C}_1 (\lambda_1 - \lambda_2)}{g (\lambda_2 + \eta_1)} \leq 0 \quad (45)$$

which requires $\hat{C}_1 \leq 0$. Hence there is a range

$$\frac{-D_1 (\lambda_2 + \eta_1)^2}{4\mu K_r N_1 (\lambda_1 - \lambda_2)} \leq \hat{C}_1 \leq 0 \quad (46)$$

for positive g over which a double valued solution may exist. Rejecting the positive sign when the negative-sign solution is available gives good agreement with TMAP4 calculations.

TRANSIENT TIMES

The time evolution of the permeation current, J_p , in the classical one-dimensional problem where a plate with concentrations $n(x,t)$ where $n(x>0,0) = 0$ and $n(0,0) = n_0$, where $n(L,t) = 0$, and where temperature gradient effects are negligible, is given by the equation

$$J_p = \frac{n_0 DA}{L} \left\{ 1 + 2 \sum_{m=1}^{\infty} \left[(-1)^m \exp \left[-m^2 \frac{t}{\tau_D} \right] \right] \right\} \quad (47)$$

where the characteristic diffusion time, τ_D , is given by

$$\tau_D = \frac{L^2}{\pi^2 D} \quad (48)$$

Breakthrough is usually defined to occur at $\tau_D/2$, the time-axis intercept of the steepest tangent.

The permeation rate development is estimated for the above idealized problem as essentially a saturating exponential function with time constant τ_D and offset from the $t = 0$ axis by $\tau_D/2$. Inventory buildup has the same time constant but no delay.

In estimating the times required to get breakthrough for the multiple-segment problem encountered here, I note that for linear problems such as this, one may take the total diffusion transient time as the simple sum of the diffusion times for each of the segments. That is

$$\bar{\tau}_D = \sum_{i=1}^m \frac{L_i}{\pi^2 D_i} \quad (49)$$

It is entirely possible that one segment will have a substantially longer diffusion time than the rest of the segments in which case the total time will be close to that longest one. Likewise, if trapping effects are evident, one may again sum the characteristic times for the individual segments to arrive at the overall time.

Detailed discussion of the effects of trapping on permeation transients is provided elsewhere,⁴ but a brief summary is given here. Two limits must be considered. In the effective diffusivity limit, the traps are present and delay the permeation transient, but their effects are virtually indistinguishable from a reduction in the diffusivity. The effective diffusivity is given by

$$D_{eff} = \frac{D}{\sum_j \left(1 + \frac{1}{\zeta_j}\right)} \quad (50)$$

where the trapping parameter, ζ_j , is

$$\zeta_j = \frac{\lambda^2 \nu}{\rho_j D} \exp \left[-\frac{E_{tj}}{RT} \right] \quad (51)$$

In this equation λ is the material lattice spacing, ν is the Debye frequency (typically 10^{13} s^{-1}), and ρ_j and E_{tj} are the trap concentration fraction and trapping energy for the j th trap type. When the trapping characteristics are known, Eqs.(50) and (51) may be used to find D_{eff} which is then substituted for D in Eq.(48) to find the characteristic permeation time, τ_D . Then breakthrough and permeation transient development may be estimated for PFC components using the standard formula.

The inventory for the i th segment in the effective diffusivity limit is just $1+1/\zeta_i$ times the inventory given by the classical diffusion model.

The other limit is the strong trapping limit. There, the traps so dominate atom movement that effectively none of the diffusing tritium can get past them until they become filled. Hence, a front moves through the plate behind which traps are all filled and the mobile concentration is non-zero and ahead of which neither trapped nor mobile atoms of the diffusing species are present. In that case the breakthrough time is given by

$$\tau_s = \frac{L^2 \rho}{2n_s D} \quad (52)$$

In the strong trapping limit, no permeation occurs until τ_s , and then permeation quickly jumps to its steady state value. The total inventory in this limit is $1 + \rho/n$ times the mobile atom inventory.

The determination as to whether the effective-diffusivity or the strong-trapping limit may apply lies in the comparison of ζ and n/ρ . If $\zeta \gg n/\rho$ then the effective diffusivity limit applies and vice versa. For situations when $\zeta \approx n/\rho$, neither of these limits is satisfactory, and an intermediate value must be found. As an interpolating function, consider again Eq.(1) which may be rewritten as

$$N \frac{\partial}{\partial t} \left[n \left(1 + \frac{n_t}{n} \right) \right] = \nabla \cdot \mathbf{J} \quad (53)$$

or

$$N \frac{\partial}{\partial t} \left[n \left(1 + \frac{1}{\zeta + \frac{n}{\rho}} \right) \right] = \nabla \cdot \mathbf{J} \quad (54)$$

If the ratio of trapped to mobile tritium concentrations, $\frac{n_t}{n}$, could be considered constant, then the term in parentheses on the left side of Eqs.(53) and (54) may be thought of as the scale factor for problem time. If $\zeta \gg n/\rho$, then this is the case, and the effective-diffusivity time scaling results. If ζ is of the order of or less than n/ρ , then the issue is more clouded.

In equilibrium $\frac{n_t}{n}$ may be obtained by equating the trapping rate to the release rate:

$$n \frac{D}{\lambda^2} (\rho - n_t) = n_t \nu \exp \left(- \frac{E_t}{RT} \right) \quad (55)$$

This leads to

$$\frac{n_t}{n} = \frac{1}{\zeta + \frac{n}{\rho}} \quad (56)$$

with ζ as defined in Eq.(51). Note further that τ_s as given by Eq.(52) is related to the simple diffusional breakthrough time by

$$\tau_s = \left(\frac{\pi^2 \rho}{n_s} \right) \frac{\tau_D}{2} \quad (57)$$

If n/ρ in Eq.(54) were replaced by $n_s/\pi^2\rho$, then in the $\zeta \ll n/\rho$ case, the right value would be obtained. For the general segment, n_s is the upstream tritium concentration in steady state. These relationships suggest that the characteristic breakthrough time for a segment may be taken as

$$\tau_B = \frac{\tau_D}{2} \left[1 + \frac{1}{\zeta + \frac{n}{\pi^2 \rho}} \right] \quad (58)$$

This is found to give good agreement with TMAP4 calculations.

INVENTORY AND PERMEATION

Mobile atom inventory is found as the simple integral over the thickness of the concentrations. Hence, for a Type I segment

$$I = N_i \left\{ \frac{C_1}{\lambda_1} [\exp(\lambda_1 L_i) - 1] + \frac{C_2}{\lambda_2} [\exp(\lambda_2 L_i) - 1] \right\} \quad (59)$$

while for Types II and III, respectively

$$I = N_i \left\{ \frac{\exp(-\beta_i L_i) [\sin(q_i L_i) (C_1 q_i - C_2 \beta_i) - \cos(q_i L_i) (C_1 \beta_i + C_2 q_i)] + C_1 \beta_i + C_2 q_i}{\beta^2 + q_i^2} \right\} \quad (60)$$

and

$$I = N_i \left\{ \frac{C_1}{\beta_i} [1 - \exp(-\beta_i L_i)] + \frac{C_2}{\beta_i^2} [1 - \exp(-\beta_i L_i) (1 + \beta_i L_i)] \right\} \quad (61)$$

Trapped atom inventory is simply taken as

$$I_t = I \frac{n_t}{n} = \frac{I}{\zeta + \frac{\langle n \rangle}{\rho}} \quad (62)$$

Tritium permeation is found by evaluating J at the downstream boundary using Eq.(2) or

$$J_F = n_{2a}^2 \mu K_r^2 \quad (63)$$

where n is determined from one of Eqs.(8) to (10) depending on the type of the m th segment.

The solution has been encoded in Microsoft BASICA. A listing of the source code is in the Appendix. It is important to run in full double precision mode (add the /D parameter on starting BASICA). Outputs include screen tabular data on the parameters calculated and a concentration profile plot.

ILLUSTRATIVE PROBLEM

For a comparison problem, I have selected one referenced by Causey et al.⁵ that involves 2 mm of beryllium on a 5-mm thick vanadium substrate. Three segments are used. The first is a 100-nm thick beryllium surface layer with a trap concentration $\rho = 0.01$ and average trap energy of 1.8 eV. The second layer is the remainder of the beryllium, 2-mm thick, with a trap concentration of 0.001, again at 1.8 eV. No trapping is assumed in the vanadium layer. Input parameters are listed in Table 1. Heat fluxes and surrounding temperatures were adjusted to give the same temperature distributions as the same problem solved using TMAP4.

Table I. Input file with the parameters used in the comparison problem.

Test Problem for Be on V	
3	Number of serial segments
"Be Film", "Be", "Vanadium"	Segment materials in order
1	Plate area (m ²)
1.e-7, 0.002, 0.005	Segment thicknesses L(i) (m)
8.542e5	Heat load to surface (W/m ²)
1e21	Implantation flux (D+T/m ² s)
1.0	Tritium fraction of flux
1.5e-8	Implantation depth (m)
"Set Value"	Flag: use "Baskes" for that K _r
1.0000e-30	Upstream recombination coefficient or α_1
1e-10	Downstream recombination coefficient or α_2
602	Coolant temperature (K)
923	Radiant heat sink temperature (K)
5000	Convective coefficient (W/m ² K)
103, 103, 32	Thermal conductivities (W/mK)
0.0581, 0.0581, 1.596e-6	Solubility pre-exponentials (Pa ^{-1/2})
1.0, 1.0, 0.34	Solution enthalpies (eV)
8.0e-9, 8.0e-9, 5.20e-8	Diffusion pre-exponentials (m ² /s)
0.365, 0.365, 0.08	Diffusion activation energies (eV)
1e-6, 1e-6, 1e-6	Heats of transport (eV)
1.23e29, 1.23e29, 8.46e28	Material number densities (m ⁻³)
0.01, 0.001, 1e-6	Trap concentrations (atom fraction)
1.8, 1.8, 1e-6	Trap energies (eV)
0.15	Thermal emissivity upstream
1e-5	Convergence limit
500	Maximum number of iterations allowed

Results of this calculation give good agreement with TMAP4 results. Figure 2 gives a comparison of the concentration profiles predicted by the two models. They are virtually identical. Steady-state mobile atom inventory calculated by the present model is 2.35×10^{22} atom/m² while the trapped atom inventory is 2.43×10^{23} atom/m². Buildup of mobile and trapped inventories in the composite as calculated by TMAP4 are shown in Figure 3 and Figure 4, respectively. The majority of the mobile atom inventory is in the 2-mm thick Be

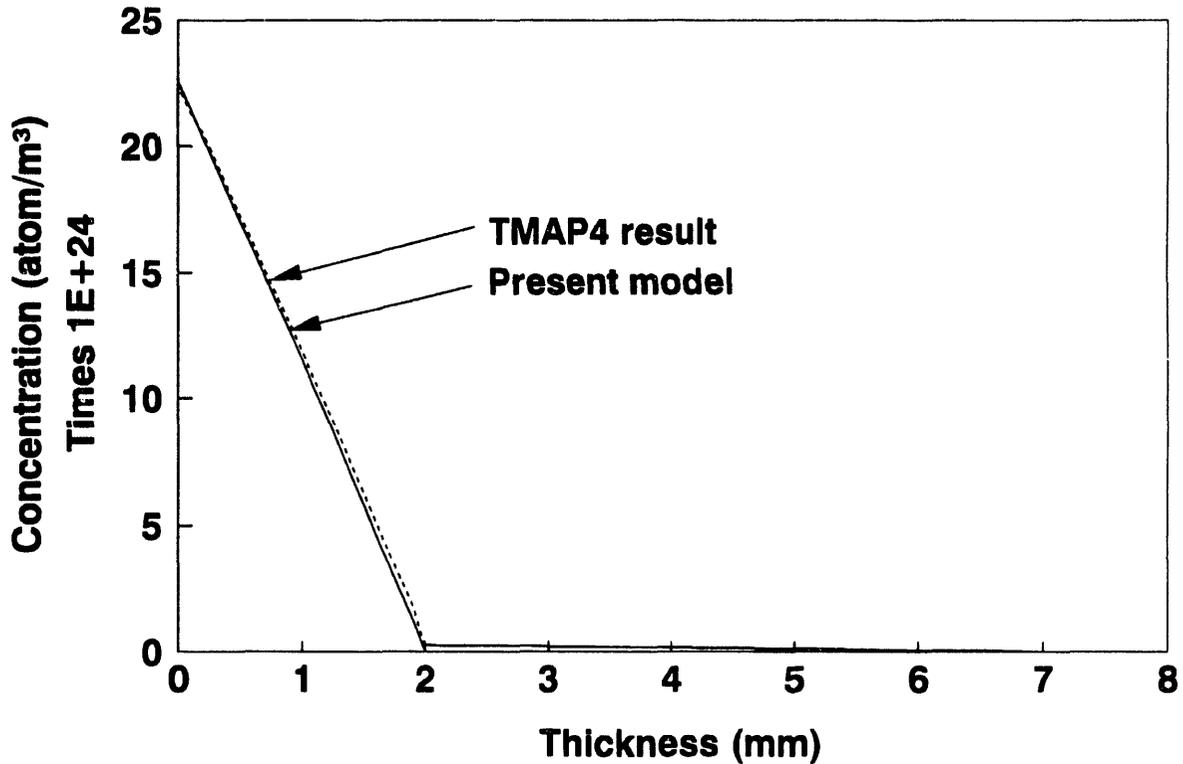


Figure 2. Comparison of the present model with TMAP4 results for the Be on V comparison problem.

layer (segment 2). Virtually all of the trapped inventory is in that segment. Trapping in the thin surface layer is fractionally higher, but it is so thin as to be of little consequence, and no trapping is assumed to take place in the 5-mm thick vanadium layer.

Permeation history is shown in Figure 5. TMAP4 predicts a comparatively abrupt permeation transient after a long lag due to trapping effects. The breakthrough time estimated by the present model is in good agreement, although it is slightly shorter, by approximately $\tau_D/2$ for the second segment.

CONCLUSIONS

This model is easier to use than TMAP4. For many situations it gives results that are useful and in good agreement with results obtained from TMAP4. It is not well suited to the more complex interactive problems addressable with TMAP4.

The general effects of trapping on permeation transients have been described. These effects can range from a retardation that is indistinguishable from slower diffusivity to a long delay followed by abrupt permeation. Times associated with these processes are derived

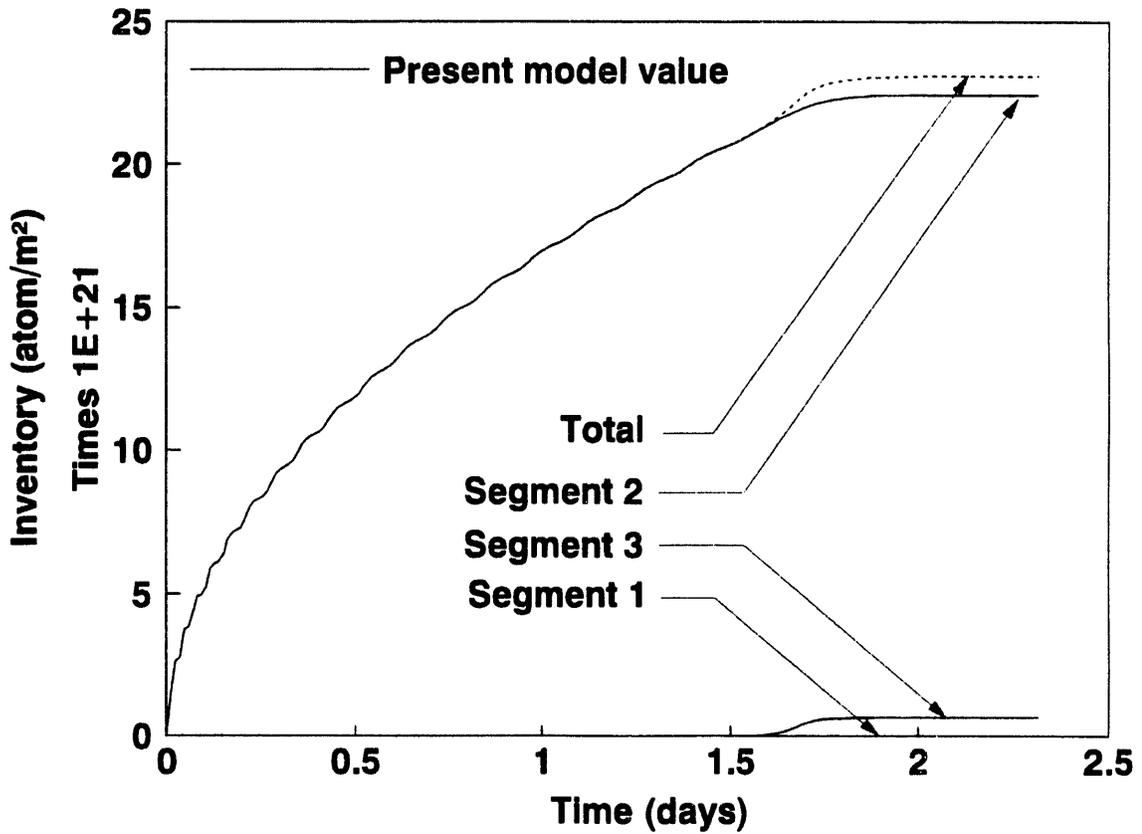


Figure 3. Mobile atom inventory buildup for the Be on V comparison problem as calculated by TMAP4. based on quasi-thermodynamic equilibrium between trap filling and emptying rates.

This model may be readily translated to other high-level languages such as FORTRAN or C++. It should be useful in estimating tritium performance of plasma-facing components in a wide variety of applications.

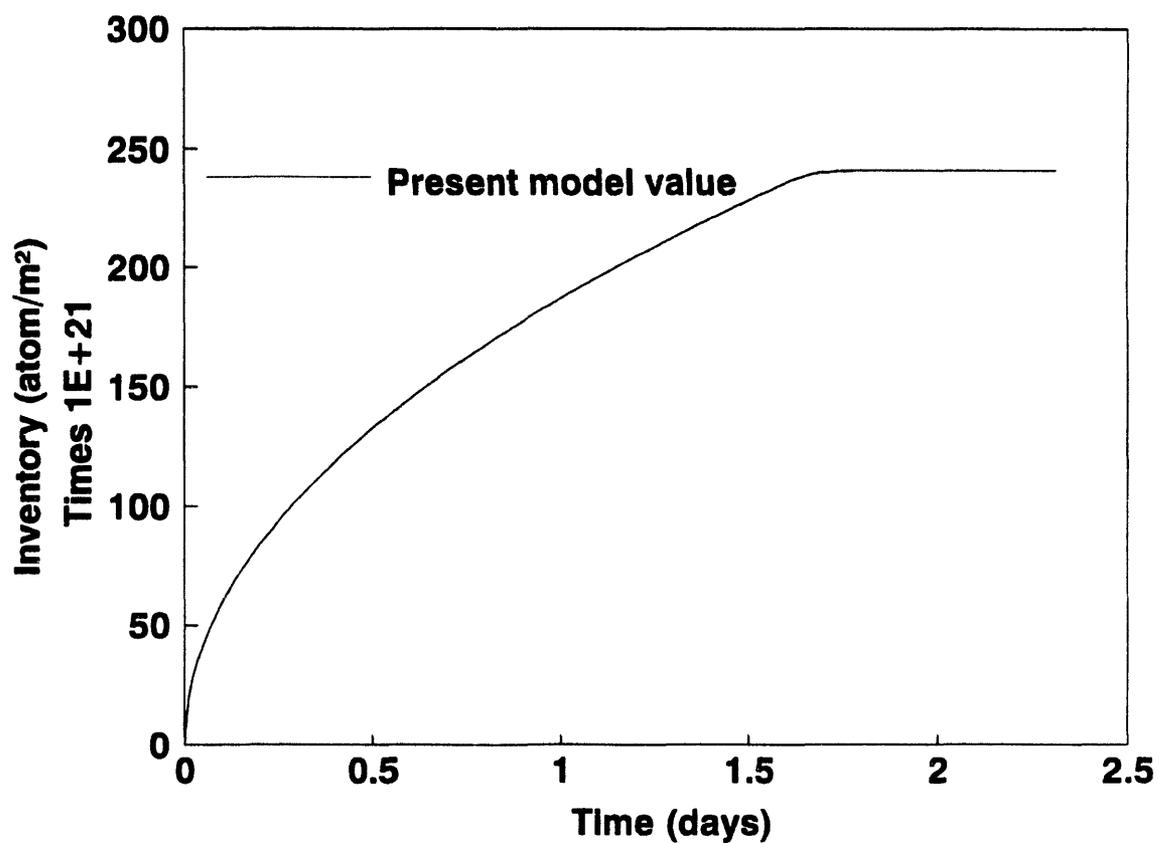


Figure 4. Trapped atom inventory buildup for the Be on V comparison problem as calculated by TMAP4.

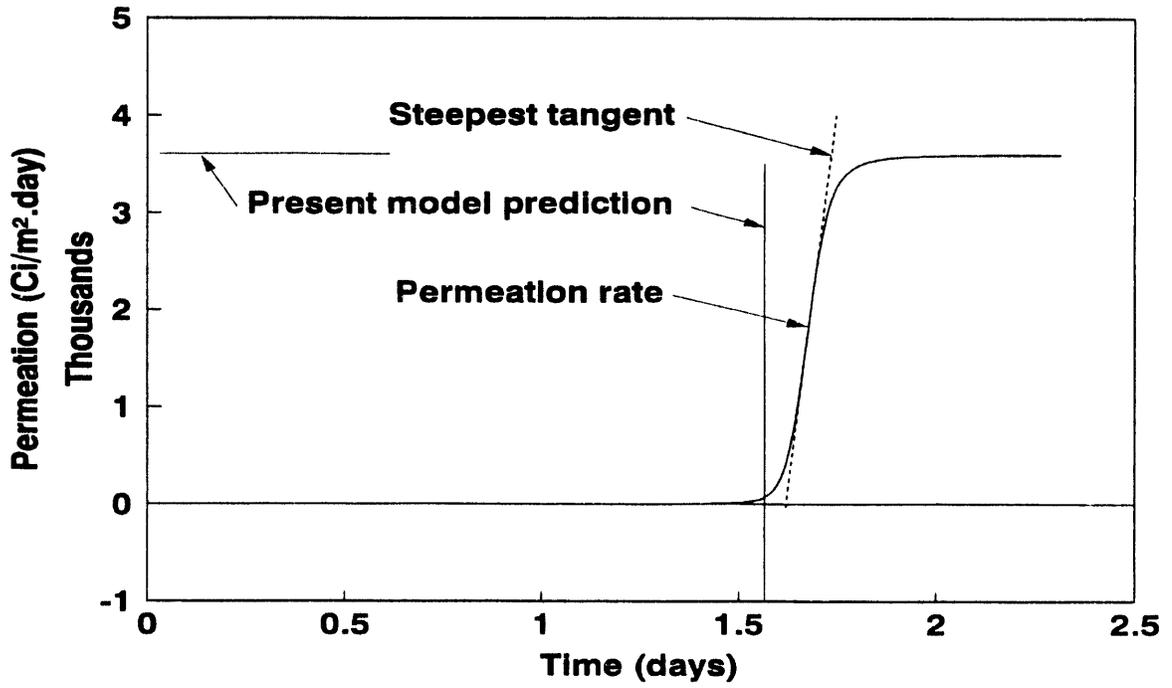


Figure 5. Permeation history as calculated by TMAP4 for the Be on V comparison problem, showing the breakthrough time predicted by the present model.

REFERENCES

1. G. R. Longhurst, D. F. Holland, J. L. Jones, and B. J. Merrill, "TMAP4 User's Manual," EGG-FSP-10315, June 12, 1992.
2. G. R. Longhurst, S. L. Harms, E.S. Marwil, and B. G. Miller, "Verification and Validation of TMAP4," EGG-FSP-10347, July 8, 1992.
3. M. I. Baskes, "A Calculation of the Surface Recombination Rate Constant for Hydrogen Isotopes on Metals," *Journal of Nuclear Materials* 92 (1980) p. 318.
4. G. R. Longhurst, "Trapping Effects on Diffusion Transients," EGG-FSP-9789, July 1991, Idaho National Engineering Laboratory, Idaho Falls, Idaho.
5. R. A. Causey, et al., "Tritium Recycling, Permeation, and Inventory in Beryllium," Sixth International Conference on Fusion Reactor Materials, Stresa, Italy, September 27 - October 1, 1993.

APPENDIX

BASICA Listing of the Code

```

10 /*****
20 /*          PFCMOD2 CODE
30 /*          FOR ESTIMATING TRITIUM PERMEATION TRANSIENTS AND CONCENTRATIONS
40 /*          IN MULTILAYER PLATES SUBJECT TO ZERO PRESSURE ION IMPLANTATION.
50 /*****
60 /
70 /          Begin by reading in problem parameters from a file with the
80 /          default name of INDATA.
90 /
100 SCREEN 9,1,0,0: COLOR 13: DIM X(10,10), CON(10,10)
110 CLS: KEY OFF: DEFDBL A-H, 0-Z, K-M
120 PRINT "
130 PRINT "
140 PRINT "
150 PRINT "
160 /
170 /Function definitions for concentration
180 DEF FNF1(I,X) = C1(I)*EXP(LAM1(I)*X)+C2(I)*EXP(LAM2(I)*X)
190 DEF FNF2(I,X) = EXP(-BETA(I)*X)*(C1(I)*COS(Q(I)*X)+C2(I)*SIN(Q(I)*X))
200 /Function definitions for concentration gradients
210 DEF FNF3(I,X) = (C1(I)+C2(I)*X)*EXP(-BETA(I)*X)
220 DEF FND1(I,X) = C1(I)*LAM1(I)*EXP(LAM1(I)*X)+C2(I)*LAM2(I)*EXP(LAM2(I)*X)
230 DEF FND2(I,X) = EXP(-BETA(I)*X)*(COS(Q(I)*X)*(Q(I)*C2(I)-BETA(I)*C1(I))-
SIN(Q(I)*X)*(Q(I)*C1(I)+BETA(I)*C2(I)))
240 DEF FND3(I,X) = (C2(I)-BETA(I)*(C1(I)+C2(I)*X))*EXP(-BETA(I)*X)
250 /Function definitions for mobile atom inventories
260 DEF FNT1(I,X)=C1(I)/LAM1(I)*(EXP(LAM1(I)*X)-1)+C2(I)/LAM2(I)*
(EXP(LAM2(I)*X)-1)
270 DEF FNT2(I,X)=(EXP(-BETA(I)*X)*(SIN(Q(I)*X)*(C1(I)*Q(I)-C2(I)*BETA(I))-
COS(Q(I)*X)*(BETA(I)*C1(I)+Q(I)*C2(I)))+C1(I)*BETA(I)+C2(I)*Q(I))/(Q(I)^2+BETA
(I)^2)
280 DEF
FNT3(I,X)=C1(I)/BETA(I)*(1-EXP(-BETA(I)*X))+C2(I)/BETA(I)^2*(1-EXP(-BETA(I)*X)
*(1+BETA(I)*X))
290 FIS="INDATA": PRINT "What is the input file name-" FIS: INPUT AS
300 IF AS<>" " THEN FIS=AS: AS=""
310 OPEN FIS FOR INPUT AS #1
320 TRSW = 0: INPUT "Do you wish to keep a TRACE file (Y/[N])":AS
330 IF AS<>"y" AND AS<>"Y" THEN 350
340 TRSW=-1: OPEN "trace" FOR OUTPUT AS #2
350 AS=""
360 INPUT#1, PROBS: 'Read the problem title
370 LOCATE CSRLIN-2: PRINT PROBS TAB(60)
380 INPUT#1, N, Z$: 'Read the number of segments in the problem
390 FOR I=1 TO N: INPUT#1, MATL$(I): NEXT: INPUT#1, Z$: 'Read material names
400 INPUT#1, AREA, Z$: 'Read plate area
410 FOR I=1 TO N: INPUT#1, THK(I): NEXT: INPUT#1, Z$: 'Read segment
thicknesses
420 INPUT#1, HEAT, Z$: 'Read heat load to surface
430 INPUT#1, FLUX, Z$: 'Read ion flux (combined D+T) to surface
440 INPUT#1, PHI, Z$: 'Read fraction of flux that is tritium
450 INPUT#1, DELTA, Z$: 'Read the implantation depth
460 INPUT#1, KR$, Z$: 'Read flag to know if recombination is Baskes or other

```

```

PFC MODEL II
Glen R. Longhurst
"
"
": COLOR 3

```

```

470 INPUT#1, ALPHA1, Z$: IF KR$<>"Baskes" THEN KR1=ALPHA1: 'Read upstream
alpha
480 INPUT#1, ALPHA2, Z$: IF KR$<>"Baskes" THEN KR2=ALPHA2: 'Read downstream
alpha
490 INPUT#1, TCOOL, Z$: 'Read coolant temperature
500 INPUT#1, THOT, Z$: 'Read radiant heat sink temperature
510 INPUT#1, KCONV, Z$: 'Read convective heat transfer coefficient
520 FOR I=1 TO N: INPUT#1, TCOND(I): NEXT: INPUT#1, Z$ 'Read thermal
conductivities
530 FOR I=1 TO N: INPUT#1, SO(I): NEXT: INPUT#1, Z$ 'Read solubility
pre-exponentials
540 FOR I=1 TO N: INPUT#1, ES(I): NEXT: INPUT#1, Z$ 'Read solution enthalpies
550 FOR I=1 TO N: INPUT#1, DO(I): NEXT: INPUT#1, Z$ 'Read diffusion
pre-exponentials
560 FOR I=1 TO N: INPUT#1, ED(I): NEXT: INPUT#1, Z$ 'Read diffusion activation
energies
570 FOR I=1 TO N: INPUT#1, QSTAR(I): NEXT: INPUT#1, Z$ 'Read heats of
transport
580 FOR I=1 TO N: INPUT#1, DENS(I): NEXT: INPUT#1, Z$ 'Read material densities
590 FOR I=1 TO N: INPUT#1, RHO(I): NEXT: INPUT#1, Z$ 'Read trap densities
600 FOR I=1 TO N: INPUT#1, ET(I): NEXT: INPUT#1, Z$ 'Read trap energies
610 INPUT#1, EMS, Z$: 'Read upstream thermal emissivity
620 INPUT#1, EPS, Z$: 'Read convergence limit
630 INPUT#1, MAXIT, Z$: 'Read maximum number of iterations allowed
640 BOLTZ = 5.67E-08: 'Set value of Steffan-Boltzmann constant
650 MU = 2+SQR(6/5)*(1-PHI)/PHI: 'Set value for recombination modifier
660 '
670 ' Calculate temperature distribution in segments
680 '
690 QEFF = HEAT: 'Guess an initial value for Qeff
700 ITCT = 0
710 '
720 '---- Loop to calculate temperatures ----
730 '
740 T2(N) = TCOOL + QEFF/KCONV
750 FOR I=N TO 1 STEP -1
760   T1(I) = T2(I) + QEFF*THK(I)/TCOND(I)
770   T2(I-1) = T1(I)
780 NEXT
790 QEST = HEAT-BOLTZ*EMS*(T1(1)^4 - THOT^4)
800 IF ABS(QEST - QEFF)/QEFF > EPS THEN QEFF = QEST: IF ICT < MAXIT THEN 740
ELSE PRINT "Temperature not converged": STOP
810 PRINT USING "Qeff = ##.##^4";QEFF
820 PRINT "Temperatures (K)
"; LOCATE CSRLIN,20: FOR I=1 TO N: PRINT TAB(20) INT(T1(I)): NEXT: PRINT
TAB(20) INT(T2(N))
830 LINPOS = CSRLIN
840 COLOR 7: PRINT "Segment   Type           Zeta           TauD(s)       TauS(s)
TauB(s)": COLOR 3
850 '
860 ' Calculate temperature sensitive constants
870 '
880 FOR I=1 TO N: TAV(I)=SQR(T1(I))*SQR(T2(I))
890 GRADT(I) = (T2(I)-T1(I))/THK(I)

```

```

900 S1(I) = S0(I)*EXP(-ES(I)*11606/T1(I))
910 S2(I) = S0(I)*EXP(-ES(I)*11606/T2(I))
920 D1(I) = D0(I)*EXP(-ED(I)*11606/T1(I))
930 D2(I) = D0(I)*EXP(-ED(I)*11606/T2(I))
940 DAV(I) = D0(I)*EXP(-ED(I)*11606/TAV(I))
950 BETA(I) = (ED(I)+QSTAR(I))*11606/2/TAV(I)^2*GRADT(I)
960 GAMMA(I) =
(ED(I)*11606/TAV(I)^2-2/TAV(I))*QSTAR(I)*11606/TAV(I)^2*GRADT(I)^2
970 '
980 ' Check to see which form the solution should be in for this segment
990 '
1000 IF BETA(I)^2-GAMMA(I) <= 0 THEN 1050
1010 TYP(I) = 1
1020 LAM1(I) = -BETA(I)+SQR(BETA(I)^2-GAMMA(I))
1030 LAM2(I) = -BETA(I)-SQR(BETA(I)^2-GAMMA(I))
1040 GOTO 1090
1050 IF BETA(I)^2-GAMMA(I) = 0 THEN 1080
1060 TYP(I)=2: Q(I) = SQR(GAMMA(I)-BETA(I)^2)
1070 GOTO 1090
1080 TYP(I) = 3
1090 ETA(I) = QSTAR(I)*11606/TAV(I)^2*GRADT(I)
1100 TAUD(I) = THK(I)^2/9.8696044#/DAV(I)
1110 TAUS(I) = THK(I)^2*RHO(I)/2/DAV(I): 'Note this must still be divided by
HIC(i)
1120 ZETA(I) = 1E+13/DENS(I)^(2/3)/RHO(I)/DAV(I)*EXP(-ET(I)*11606/TAV(I))
1130 PRINT USING "  ##      ##      ##.##^ ^ ^ ^      ##.##^ ^ ^ ^      ##.##^ ^ ^ ^
##.##^ ^ ^ ^";I;TYP(I);ZETA(I);TAUD(I)
1140 NEXT: ZETA(0)=ZETA(1)
1150 '
1160 ' Find recombination coefficients
1170 '
1180 IF KR$<>"Baskes" THEN 1230: ' If not Baskes recombination dont find Krs
1190 EX1 = ES(1)+ED(1): IF EX1 < 0 THEN EX1 = 0
1200 EX2 = ES(N)+ED(N): IF EX2 < 0 THEN EX2 = 0
1210 KR1 =
ALPHA1*EXP(-EX1*11606/T1(1))/9.321E-25/SQR(T1(1))/S1(1)/DENS(1)/S1(1)/DENS(1)
1220 KR2 =
ALPHA2*EXP(-EX2*11606/T2(N))/9.321E-25/SQR(T2(N))/S2(N)/DENS(N)/S2(N)/DENS(N)
1230 PRINT USING "Kr1 = ##.###^ ^ ^ ^";KR1
1240 PRINT USING "Kr2 = ##.###^ ^ ^ ^";KR2
1250 G = D1(1)/2/MU/KR1/DENS(1)*(LAM2(1)+ETA(1))
1260 H = G*(LAM2(1)+ETA(1))/2/(LAM1(1)-LAM2(1))
1270 LOCATE LINPOS+N+3,23: COLOR 14: PRINT "FINDING CONCENTRATION
COEFFICIENTS":'PRINT TAB(32) "Iteration No.";
1280 '
1290 ' Basic loop for finding concentration coefficients and convergence
1300 ' to steady state values of concentration
1310 '
1320 ICT = 0: PT1=0#: PT2=.5#: PT3=1#: 'SW=1: PT10=PT1: PT30=PT3
1330 FGS=PT1: GOSUB 1460: PS1=PSI: 'PS10=PS1
1340 FGS = PT3: GOSUB 1460: PS3=PSI: 'PS0=PS1: PS30=PS3
1350 ICT = ICT+1: FGS = PT2: GOSUB 1460: PS2=PSI: IF TRSW THEN PRINT#2, FGS
PS2

```

```

1360 LOCATE LINPOS+N+4,15: PRINT "Iteration " ICT TAB(30) "Direction " SW
TAB(45) "Psi " CSNG(PS2) " ";
1370 IF ICT>MAXIT THEN PRINT "Not converged": STOP
1380 IF ABS(PS2) < EPS OR ABS((PT3-PT2)/PT2)< EPS THEN 1420: 'Convergence
achieved, go on
1390 IF SGN(PS1)=SGN(PS3) THEN PRINT "Problem finding a root": STOP
1400 IF SGN(PS2)=SGN(PS3) THEN PT3=PT2: PS3=PS2: PT2=(PT2+PT1)/2#: SW=2: GOTO
1350
1410 IF SGN(PS2)=SGN(PS1) THEN PT1=PT2: PS1=PS2: PT2=(PT2+PT3)/2#: SW=1: GOTO
1350
1420 GOTO 1680
1430 '-----
1440 ' This is the subroutine for finding the concentration constants
1450 '-----
1460 ON TYP(1) GOSUB 2350,2400,2400: 'This gives C1HAT
1470 IF C1HAT<-H THEN PRINT "Problem with value of C1HAT: " C1HAT: STOP
1480 MLT=1: IF (G<0 AND C1HAT>=0) OR (G>0 AND C1HAT<0) THEN MLT=-1
1490 ON TYP(1) GOSUB 2450,2460,2470: 'Find corresponding values of C2HAT
1500 C1(0) = C1HAT: C2(0)=C2HAT: 'This is for later evaluation of n1
1510 ON TYP(1) GOSUB 2510,2570,2630: 'This finds C1(1) and C2(1)
1520 FOR I=1 TO N-1: 'Solve the interfacial equations
1530 ON TYP(I) GOSUB 2690,2700,2710: 'Find n(i) = CONC
1540 ON TYP(I) GOSUB 2750,2760,2770: 'Find GRAD n(i) = DCONC
1550 ON TYP(I+1) GOSUB 2810,2870,2930: 'This finds remaining C1(i) and C2(i)
1560 NEXT
1570 '
1580 ' Now find the error in permeation flux
1590 '
1600 I = N: ON TYP(N) GOSUB 2690,2700,2710: 'Find n2
1610 ON TYP(N) GOSUB 2750,2760,2770: 'Find GRAD n2
1620 FPN = MU*KR2*CONC*DENS(N)*ABS(CONC): 'Permeation/N(n)
1630 PSI = FPN/PHI/FLUX*DENS(N)-FGS
1640 RETURN
1650 '
1660 ' Begin output of results
1670 '
1680 CLOSE: LOCATE LINPOS+N+2: PRINT TAB(24) "          CONCENTRATIONS
": PRINT "
": LOCATE CSRLIN-1
1690 PRINT "x/L      Implant "; FOR I=1 TO N: PRINT "      Zone " I;: NEXT
1700 RHO(0) = RHO(1): LAM1(0) = LAM1(1): LAM2(0)=LAM2(1): THK(0)=DELTA: CMAX=0
1710 FOR J=0 TO 10
1720   FOR I=0 TO N
1730     X = J/10*THK(I): ON TYP(I) GOTO 1740, 1760, 1780
1740     CON(J,I) = C1(I)*EXP(LAM1(I)*X)+C2(I)*EXP(LAM2(I)*X)
1750     GOTO 1790
1760     CON(J,I) = EXP(-BETA(I)*X)*(C1(I)*COS(Q(I)*X)+C2(I)*SIN(Q(I)*X))
1770     GOTO 1790
1780     CON(J,I) = (C1(I)+C2(I)*X)*EXP(-BETA(I)*X)
1790     IF CON(J,I)>CMAX THEN CMAX=CON(J,I)
1800     IF CON(J,I)>HIC(I) THEN HIC(I)=CON(J,I)
1810   NEXT I,J: TSUM=0
1820 COLOR 3: FOR I=1 TO N: TAUS(I)=TAUS(I)/HIC(I): LOCATE LINPOS+I,42: PRINT
USING "##.##^^^";TAUS(I)

```

```

1830 TEST(I) = TAUD(I)/2*(1+1/(ZETA(I)+(HIC(I)/RHO(I)/9.8696#)^1!)): TSUM =
TSUM+TEST(I)
1840 LOCATE LINPOS+I,54: PRINT USING "##.##^^^^";TEST(I):NEXT: LOCATE
CSRLIN,26:COLOR 12: PRINT USING "Total breakthrough time (d)
##.##^^^^";TSUM/86400!
1850 /
1860 /   Find permeation rates
1870 /
1880 ON TYP(N) GOTO 1890, 1910, 1930
1890 PERM = -D2(N)*(C1(N)*(LAM1(N)+ETA(N))*EXP(LAM1(N)*THK(N))+C2(N)*(LAM2(N)+
ETA(N))*EXP(LAM2(N)*THK(N)))*4.164E-15*AREA*DENS(N)
1900 GOTO 1940
1910 PERM = -D2(N)*EXP(-BETA(N)*THK(N))*(-C1(N)*Q(N)*SIN(Q(N)*THK(N))+C2(N)*
Q(N)*COS(Q(N)*THK(N)))*4.164E-15*AREA*DENS(N)
1920 GOTO 1940
1930 PERM = -D2(N)*EXP(-BETA(N)*THK(N))*(C1(N)+C2(N)*THK(N))*4.164E-15
*AREA*DENS(N)
1940 LOCATE CSRLIN,26: COLOR 12: PRINT USING "Permeation rate (Ci/d)
##.##^^^^"; PERM
1950 COLOR 7: LOCATE LINPOS-N-2,41: PRINT "Inventories (Ci)";
1960 LOCATE LINPOS-N-1,31: PRINT "Mobile      Trapped      Both";
1970 FOR I=1 TO N: ON TYP(I) GOSUB 2990, 3010, 3030: MOB=MOB+MOBINV(I)
1980 TRAPINV(I)=PHI*MOBINV(I)/(ZETA(I)+CON(5,I)/RHO(I)): TRP=TRP+TRAPINV(I)
1990 COLOR 12: LOCATE LINPOS-N-1+I,30: PRINT USING "##.##^^^^  ##.##^^^^
##.##^^^^"; MOBINV(I), TRAPINV(I), MOBINV(I)+TRAPINV(I);: NEXT: TOT=MOB+TRP
2000 COLOR 7: LOCATE LINPOS-2,66: PRINT "TOTAL";
2010 COLOR 10: LOCATE LINPOS-1,65: PRINT USING "##.##^^^^";TOT;
2020 LOCATE LINPOS+5+N,1
2030 FOR J=0 TO 10 STEP 2: PRINT USING "#.#";J/10;
2040 FOR I=0 TO N: PRINT USING "  ##.##^^^^";CON(J,I);: NEXT: PRINT
2050 NEXT: PRINT "Press any key to toggle contour plot then [Q] to quit";
2060 A$=INKEY$: IF A$="" THEN 2060
2070 /
2080 /   Plot concentration profiles
2090 /
2100 SCREEN 9,0,1,1
2110 XX = 0: FOR I=0 TO N: SG(I)=XX: XX=XX+THK(I)
2120 FOR J=0 TO 10: CON(J,I)=300-CON(J,I)/CMAX*250: NEXT: NEXT
2130 FOR I=0 TO N: FOR J=0 TO 10: X(J,I)=(J/10*THK(I)+SG(I))/XX*400+120: NEXT:
NEXT
2140 CLS: LINE (120,300)-(520,50),7,BF:PRESET(120,300),1: LINE
-(X(0,0),CON(0,0)),14: DRAW "NL10"
2150 LOCATE 1,40-LEN(PROB$)/2:PRINT PROB$
2160 LOCATE 4,6: PRINT USING "##.##^^^^";CMAX: LOCATE 22,14: PRINT "0"
2170 FOR J = 1 TO 13: LOCATE 6+J: PRINT MID$("Concentration",J,1): NEXT
2180 FOR I=0 TO N: COLOR 3+I: IF I>0 THEN LOCATE 23+INT(I/2)-INT((I-1)/2),
17+SG(I)/XX*50: PRINT MATL$(I);: HIC(I)=0
2190 FOR J=0 TO 10: LINE -(X(J,I), CON(J,I)): NEXT: IF I>0 THEN LINE
-(X(10,I),300)
2200 NEXT
2210 LOCATE 1,60
2220 A$=INKEY$: IF A$="" THEN 2220
2230 IF A$="q" OR A$="Q" THEN SCREEN 0,0,0:STOP
2240 SCREEN 9,0,0,0

```

```

2250 A$=INKEY$: IF A$="" THEN 2250
2260 IF A$="q" OR A$="Q" THEN SCREEN 0,0,0:STOP
2270 SCREEN 9,0,1,1: GOTO 2220
2280 /-----
2290 / This section is a collection of subroutines for finding the density
2300 / coefficients on the respective segments depending on the type
2310 /-----
2320 /
2330 / This subroutine finds C1HAT for Type I segments
2340 /
2350 C1HAT = ((1#-FGS)*PHI*FLUX/DENS(1)/D1(1)-(LAM2(1)+ETA(1))*SQR((1#-FGS)*
PHI*FLUX)/SQR(MU*KR1)/DENS(1))/(LAMI(1)-LAM2(1))
2360 RETURN
2370 /
2380 / This subroutine finds C1HAT for Type II and III segments
2390 /
2400 C1HAT = SQR((1#-FGS)*PHI*FLUX/MU/KR1)/DENS(1)
2410 RETURN
2420 /
2430 / These subroutines get C2HAT given C1HAT
2440 /
2450 C2HAT = G*(1+MLT*SQR(C1HAT/H+1))-C1HAT: RETURN: ' Type I
2460 C2HAT = (MU*KR1*DENS(1)/D1(1)*C1HAT*C1HAT+(BETA(1)-ETA(1))*C1HAT)/Q(1):
RETURN: 'Type II
2470 C2HAT = MU*KR1*DENS(1)*C1HAT^2/D1(1)+(BETA(1)-ETA(1))*C1HAT: RETURN:
'Type III
2480 /
2490 / This subroutine finds C1(1) and C2(1) for Type I segments
2500 /
2510 C1(1) = -PHI*FLUX/DENS(1)/D1(1)/(LAMI(1)-LAM2(1))*EXP(-LAM1(1)*DELTA)+
C1HAT
2520 C2(1) = (C1HAT-C1(1))*EXP((LAM1(1)-LAM2(1))*DELTA)+C2HAT
2530 RETURN
2540 /
2550 / This subroutine finds C1(1) and C2(1) for Type II segments
2560 /
2570 C1(1) = C1HAT+PHI*FLUX/DENS(1)/D1(1)/Q(1)*SIN(Q(1)*DELTA)
2580 C2(1) = (C1HAT-C1(1))*COS(Q(1)*DELTA/SIN(Q(1)*DELTA)+C2HAT
2590 RETURN
2600 /
2610 / This subroutine finds C1(1) and C2(1) for Type III segments
2620 /
2630 C1(1) = -DELTA*PHI*FLUX/DENS(1)/D1(1)*EXP(BETA(1)*DELTA)+C1HAT
2640 C2(1) = (C1HAT-C1(1))/TAN(Q(I)*DELTA)+C2HAT
2650 RETURN
2660 /
2670 / These subroutines find n(i) for Type I, II, or III segments
2680 /
2690 CONC = FNF1(I,THK(I)): RETURN
2700 CONC = FNF2(I,THK(I)): RETURN
2710 CONC = FNF3(I,THK(I)): RETURN
2720 /
2730 / These subroutines calculate concentration gradients
2740 /

```

```

2750 DCONC = FND1(I,THK(I)): RETURN
2760 DCONC = FND2(I,THK(I)): RETURN
2770 DCONC = FND3(I,THK(I)): RETURN
2780 '
2790 ' This subroutine finds C1(i+1), C2(i+1) for type I segments
2800 '
2810 C2(I+1) = (D2(I)/D1(I+1)*DENS(I)/DENS(I+1)*(DCONC+ETA(I)*CONC)-CONC/
S2(I)*S1(I+1)*(ETA(I+1)+LAM1(I+1)))/(LAM2(I+1)-LAM1(I+1))
2820 C1(I+1) = CONC/S2(I)*S1(I+1)-C2(I+1)
2830 RETURN
2840 '
2850 ' This subroutine finds C1(i+1), C2(i+1) for type II segments
2860 '
2870 C1(I+1) = CONC*S1(I+1)/S2(I)
2880 C2(I+1) = (D2(I)/D1(I+1)*DENS(I)/DENS(I+1)*(DCONC+ETA(I)*CONC)+C1(I+1)*
(BETA(I+1)-ETA(I+1)))/Q(I+1)
2890 RETURN
2900 '
2910 ' This subroutine finds C1(i+1), C2(i+1) for type III segments
2920 '
2930 C1(I+1) = CONC*S1(I+1)/S2(I)
2940 C2(I+1) = D2(I)/D1(I+1)*DENS(I)/DENS(I+1)*(DCONC+ETA(I)*CONC)+C1(I+1)*
(BETA(I+1)-ETA(I+1))
2950 RETURN
2960 '
2970 ' These subroutines find the mobile atom inventories for the segments
2980 '
2990 MOBINV(I) = FNT1(I,THK(I))*DENS(I)*AREA*4.8194E-20
3000 RETURN
3010 MOBINV(I) = FNT2(I,THK(I))*DENS(I)*AREA*4.8194E-20
3020 RETURN
3030 MOBINV(I) = FNT3(I,THK(I))*DENS(I)*AREA*4.814E-21
3040 RETURN

```

**DATE
FILMED**

5/25/94

END

