ORNL/CSD/TM-17

Timed: A Computer Program for Calculating Cumulated Activity of a Radionuclide in the Organs of the Human Body at a Given Time, t, After Deposition

S. B. Watson
W. S. Snyder
M. R. Ford

BLANK PAGE

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22161
Price: Printed Copy $6.75; Microfiche $3.00

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the Energy Research and Development Administration United States Nuclear Regulatory Commission, nor any of their employees, nor any of their contractors, subcontractors, or 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.

ORNL/CSD/TM-17

Contract No. W-7405-eng-26

COMPUTER SCIENCES DIVISION

TIMED: A COMPUTER PROGRAM FOR CALCULATING CUMULATED ACTIVITY OF A RADIONUCLIDE IN THE ORGANS OF THE HUMAN BODY AT A GIVEN TIME, t, AFTER DEPOSITION

S. B. Watson
Computer Sciences Division

W. S. Snyder and M. R. Ford
Health Physics Division

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or 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.

Date Published: December 1976

NOTICE

This document contains information of a preliminary nature. It is subject to revision or correction and therefore does not represent a final report.

UNION CARBIDE CORPORATION, NUCLEAR DIVISION
operating the
Oak Ridge Gaseous Diffusion Plant · Oak Ridge National Laboratory
Oak Ridge Y-12 Plant · Paducah Gaseous Diffusion Plant
for the
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

The authors wish to express their thanks to Dr. D. E. Arnurius for his assistance in modifying the GEAR computer package for use with our particular problem.

B. D. McCarter deserves a special thanks for the preparation of the manuscript.

TIMED: A COMPUTER PROGRAM FOR CALCULATING CUMULATED ACTIVITY OF A RADIONUCLIDE IN THE ORGANS OF THE HUMAN BODY AT A GIVEN TIME, *t*, AFTER DEPOSITION

S. B. Watson, W. S. Snyder,
and M. R. Ford

ABSTRACT

TIMED is a computer program designed to calculate cumulated activity, $U_{Y_i}(t)$, in the various source organs, Y_i, at some time *t* after deposition. TIMED embodies a system of differential equations which describes activity transfer in the lungs, gastrointestinal tract, and other organs of the body. This system accounts for delay of transfer of activity between compartments of the body and radioactive daughters.

The computer program contains routines which are written in either IBM System/360 or System/370 FORTRAN or IBM System/360 Assembler language. The code is executable on the IBM System/360 or System/370 machines and requires a minimum of 310 K core storage for execution.

I. INTRODUCTION

Average dose equivalent in an organ *T* of the body from an internally deposited emitter is given by

$$DE(T,t) = \sum_i U_{Y_i}(t) * S(T \leftarrow Y_i) \qquad (1.1)$$

BLANK PAGE

where $U_{Y_i}(t)$ represents the cumulated activity measured in microcurie-days (μCi-days) in the various source organs, Y_i, at some time t after deposition, and $S(T \leftarrow Y_i)$ represents dose to a target organ T due to 1 μCi-day of the emitter in the source.[1] S has been tabulated for about 160 radionuclides[2,3] and for various source and target organs, but with the proviso that the activity is always considered to be uniformly distributed in the source organ. This paper discusses the computer code, TIMED, which calculates the cumulated activity, $U_{Y_i}(t)$, per microcurie intake at some time t after intake.

In pri:nciple, this calculation is simple in that if $f_{Y_i}(t)$ represents the fraction of activity in microcuries (μCi), which entered the body at time 0 that is present in an organ Y_i at time t, the cumulated activity in that organ from time 0 to time t_1, is given by

$$U_{Y_i}(t_1) = \int_0^{t_1} f_{Y_i}(t)dt \qquad (\mu\text{Ci-days}) \ . \qquad (1.2)$$

However, the calculation of $U_{Y_i}(t)$ is somewhat difficult due to the complex nature of the transfer rates between sections of the body and the possible presence of daughter radionuclides.

This paper discusses the model of activity transfer and the computational procedures that are used in the code, TIMED. Also presented are a description of the program and detailed user information.

II. DESCRIPTION OF THE MODEL

The model of activity transfer in the body is based on the concept of a system of compartments with constant rates of transfer to other compartments or out of the body. These compartments may be an organ or some subsystem of the body (lungs, gastrointestinal tract, blood, etc.). The system must be general enough to allow for delay of transfer of some of the activity and must also account for radioactive daughters. This system has been adequately represented by a set of differential equations which are described below for the lungs, gastrointestinal tract, and other organs. The following is taken in part from Ref. 1 which should be consulted for the detailed explanation. Many of the errors present in Ref. 1 are corrected in this discussion.

Lungs

The lung model, as described in the report of the ICRP Task Group on Lung Dynamics,[4] and as revised in ICRP Publication 19,[5] consists of a nasal-pharyngeal region (N-P), a tracheo-bronchial region (T-B), a pulmonary region (P), and the lymph nodes (L). Deposition is governed by the activity median aerodynamic diameter (AMAD) of the aerosol (see Ref. 4). The pathways of transfer and the deposition and transfer constants are shown in Fig. 2.1.

The differential equations appropriate for this model are given below. Taking X_a^n, X_b^n, X_c^n, X_d^n, X_e^n, X_f^n, X_g^n, X_h^n, X_i^n, X_j^n, X_k^n, X_l^n as the activities of the parent (n=1) and daughters (n=2, 3, . . .) in a compartment, moving by the pathway with the appropriate subscript, and

COMPARTMENT		CLASS					
		D		W		Y	
		T	F	T	F	T	F
N - P	a	0.01	0.5	0.01	0.1	0.01	0.01
(D_{N-P}=0.30)	b	0.01	0.5	0.40	0.9	0.40	0.99
T - B	c	0.01	0.95	0.01	0.5	0.01	0.01
(D_{T-B}=0.08)	d	0.2	0.05	0.2	0.5	0.2	0.99
	e	0.5	0.8	50	0.15	500	0.05
P	f	n.a.	n.a.	1.0	0.4	1.0	0.4
(D_P=0.25)	g	n.a.	n.a.	50	0.4	500	0.4
	h	0.5	0.2	50	0.05	500	0.15
L	i	0.5	1.0	50	1.0	1000	0.9

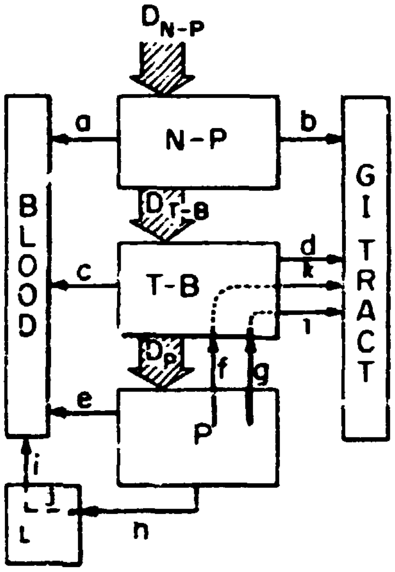


Fig. 2.1. Values for the Removal Half Times, T_a Through T_i, and Compartmental Fractions, F_a Through F_i, Are Given for Each of the Three Classes of Retained Materials. The values given for D_{N-P}, D_{T-B}, and D_P (left column) are based on an aerosol with an AMAD of 1 µm.[4] The schematic drawing identifies the various clearance pathways in the model, *a* through *i*, in relation to the initial depositions D_{N-P}, D_{T-B}, and D_P and the three respiratory regions, *N-P*, *T-B*, and *P*. The lymphatic clearance for class *Y* compounds indicates that a 90% regional fraction follows a 1000-day biological half time. The remaining 10% is presumed to be permanently retained in the nodes and is subject only to radioactive decay.

using $\dot{X}$ to represent the derivative with respect to time $\left(X \equiv X(t)\right.$ and $\left.\dot{X} \equiv \dot{X}(t) \equiv \frac{dX(t)}{dt}\right)$, the equations for the lungs are the following:

$$
\begin{aligned}
\dot{X}_a^{n+1} &= -X_a^{n+1}(\lambda_a^{n+1} + \lambda_r^{n+1}) + (X_a^n + X_b^n)\lambda_r^{n+1}F_a^{n+1} \,, \\
\dot{X}_b^{n+1} &= -X_b^{n+1}(\lambda_b^{n+1} + \lambda_r^{n+1}) + (X_a^n + X_b^n)\lambda_r^{n+1}F_b^{n+1} \,, \\
\dot{X}_c^{n+1} &= -X_c^{n+1}(\lambda_c^{n+1} + \lambda_r^{n+1}) + (X_c^n + X_d^n)\lambda_r^{n+1}F_c^{n+1} \,, \\
\dot{X}_d^{n+1} &= -X_d^{n+1}(\lambda_d^{n+1} + \lambda_r^{n+1}) + (X_c^n + X_d^n)\lambda_r^{n+1}F_d^{n+1} \,, \\
\dot{X}_e^{n+1} &= -X_e^{n+1}(\lambda_e^{n+1} + \lambda_r^{n+1}) + (X_e^n + X_f^n + X_g^n + X_h^n)\lambda_r^{n+1}F_e^{n+1} \,, \\
\dot{X}_f^{n+1} &= -X_f^{n+1}(\lambda_f^{n+1} + \lambda_r^{n+1}) + (X_e^n + X_f^n + X_g^n + X_h^n)\lambda_r^{n+1}F_f^{n+1} \,, \\
\dot{X}_g^{n+1} &= -X_g^{n+1}(\lambda_g^{n+1} + \lambda_r^{n+1}) + (X_e^n + X_f^n + X_g^n + X_h^n)\lambda_r^{n+1}F_g^{n+1} \,, \\
\dot{X}_h^{n+1} &= -X_h^{n+1}(\lambda_h^{n+1} + \lambda_r^{n+1}) + (X_e^n + X_f^n + X_g^n + X_h^n)\lambda_r^{n+1}F_h^{n+1} \,, \\
\dot{X}_i^{n+1} &= -X_i^{n+1}(\lambda_i^{n+1} + \lambda_r^{n+1}) + X_h^{n+1}\lambda_h^{n+1}F_i^{n+1} + (X_i^n + X_j^n)\lambda_r^{n+1}F_i^{n+1} \,, \\
\dot{X}_j^{n+1} &= -X_j^{n+1}\lambda_r^{n+1} + X_h^{n+1}\lambda_h^{n+1}(1 - F_i^{n+1}) + (X_i^n + X_j^n)\lambda_r^{n+1}(1 - F_i^{n+1}) \,, \\
\dot{X}_k^{n+1} &= -X_k^{n+1}(\lambda_d^{n+1} + \lambda_r^{n+1}) + X_k^n\,\lambda_r^{n+1} + X_f^{n+1}\,\lambda_f^{n+1} \,, \\
\dot{X}_l^{n+1} &= -X_l^{n+1}(\lambda_d^{n+1} + \lambda_r^{n+1}) + X_l^n\,\lambda_r^{n+1} + X_g^{n+1}\lambda_g^{n+1} \,.
\end{aligned}
\tag{2.1}
$$

F_ξ^{n+1} is the compartmental fraction from Fig. 2.1 for the n+1st radionuclide of the chain for pathway ξ. The quantity λ_ξ^{n+1} is $ln2/T_\xi^{n+1}$

where T_{ξ}^{n+1} is the removal half time in days taken from Fig. 2.1 for the n+1st radionuclide of the chain for pathway ξ. Similarly, λ_{r}^{n+1} is $ln2/T_{r}^{n+1}$ where T_{r}^{n+1} is the radioactive half life of the n+1st daughter.

The terms with negative signs correspond to elimination by biological transfer or by radioactive decay. The terms involving a superscript n refer to activity in microcuries of the preceding radionuclide of the chain which produces λ_{r}^{n+1} µCi of the following radionuclide per unit time, and these are redistributed according to the values of F for the radionuclide.

For each chain of radionuclides there are 12 differential equations corresponding to the lung model for each radionuclide. The equations are valid for the parent radionuclide, that is n=0, if $X_{\xi}^{0}\equiv 0$ for all values of t and all subscripts ξ, thus greatly simplifying the differential equations for the parent.

The initial conditions used are based on the assumption that 1 µCi of an aerosol of AMAD=1 has been inhaled, although other starting values of $D_3 \equiv D_{N-P}$, $D_4 \equiv D_{T-B}$, and $D_5 \equiv D_P$ may be used (see Ref. 4).

$$\begin{aligned}
X_a^1(0) &= D_3 F_a^1, & X_e^1(0) &= D_5 F_e^1, \\
X_b^1(0) &= D_3 F_b^1, & X_f^1(0) &= D_5 F_f^1, \\
X_c^1(0) &= D_4 F_c^1, & X_g^1(0) &= D_5 F_g^1, \\
X_d^1(0) &= D_4 F_d^1, & X_h^1(0) &= D_5 F_h^1,
\end{aligned} \qquad (2.2)$$

$$X_i^1(0) = X_j^1(0) = X_k^1(0) = X_l^1(0) = 0,$$

$$X_{\xi}^{n}(0) = 0 \text{ for } n > 1 \text{ and for all subscripts } \xi.$$

Upon solution of the differential equations for the X's which are given as activity in μCi's, one calculates the cumulated activity (μCi-days) in each lung compartment during the time interval 0 by t_1 by

$$Y_\xi^n = \int_0^{t_1} X_\xi^n \, dt \tag{2.3}$$

where Y_ξ^n represents $Y_\xi^n(t_1)$.

The cumulated activity for each region of the respiratory system is given by

$$\begin{aligned}
U_{NP}^n &= Y_a^n + Y_b^n \ , \\
U_{TB}^n &= Y_c^n + Y_d^n + Y_k^n + Y_l^n , \\
U_P^n &= Y_e^n + Y_f^n + Y_g^n + Y_h^n , \\
U_L^n &= Y_i^n + Y_j^n \ , \\
B^n &= Y_a^n\lambda_a^n + Y_c^n\lambda_c^n + Y_e^n\lambda_e^n + Y_i^n\lambda_i^n \ , \\
G^n &= Y_b^n\lambda_b^n + Y_d^n\lambda_d^n + Y_k^n\lambda_d^n + Y_l^n\lambda_d^n \ .
\end{aligned} \tag{2.4}$$

Here the U_ξ^n are the cumulated activity (μCi-days) in the region ξ of the respiratory system, while B^n and G^n represent total activity (μCi) to blood and gastrointestinal tract. In the code, TIMED, the values of Y_ξ^n will be computed, but it may be more convenient for the reader to consider the instantaneous state of the system at time t which is represented by the differential equations given in terms of the X's.

Gastrointestinal Tract

Any exposure by inhalation also leads to some activity entering the gastrointestinal tract (GIT). The dosimetric model for the GIT is essentially due to Eve[6] so far as the subdivisions of the tract and the transit times through the sections are concerned. Four subdivisions of the tract are defined: the stomach, the small intestine, the upper large intestine, and the lower large intestine. The estimates of dose are considered to be averaged over these sections. Table 2.1 provides data on average masses and times food remains in each of these sections and is quoted from ICRP Publication 23.[7]

Let X_S^n, X_{SI}^n, X_{UL}^n, and X_{LL}^n be the activities present at time t in the contents of the stomach (S), the contents of the small intestine (SI), the contents of the upper large intestine (UL), and the contents of the lower large intestine (LL), respectively, then

$$
\begin{aligned}
\dot{X}_S^{n+1} &= -X_S^{n+1}(\lambda_S^{n+1} + \lambda_r^{n+1}) + X_b^{n+1}\lambda_b^{n+1} + X_d^{n+1}\lambda_d^{n+1} + X_k^{n+1}\lambda_d^{n+1} \\
&\quad + X_l^{n+1}\lambda_d^{n+1} + X_S^{n}\lambda_r^{n+1} \,, \\
\dot{X}_{SI}^{n+1} &= -X_{SI}^{n+1}(\lambda_{SI}^{n+1} + \lambda_r^{n+1} + \lambda_{ab}^{n+1}) + X_S^{n+1}\lambda_S^{n+1} + X_{SI}^{n}\lambda_r^{n+1} \,, \qquad (2.5) \\
\dot{X}_{UL}^{n+1} &= -X_{UL}^{n+1}(\lambda_{UL}^{n+1} + \lambda_r^{n+1}) + X_{SI}^{n+1}\lambda_{SI}^{n+1} + X_{UL}^{n}\lambda_r^{n+1} \,, \\
\dot{X}_{LL}^{n+1} &= -X_{LL}^{n+1}(\lambda_{LL}^{n+1} + \lambda_r^{n+1}) + X_{UL}^{n+1}\lambda_{UL}^{n+1} + X_{LL}^{n}\lambda_r^{n+1} \,.
\end{aligned}
$$

From Table 2.1 $\lambda_S = 24$ days^{-1}, $\lambda_{SI} = 6$ days^{-1}, $\lambda_{UL} = \frac{24}{13}$ days^{-1}, and $\lambda_{LL} = 1$ day^{-1}. $\lambda_{ab} = \frac{6 \cdot f_1}{1-f_1}$ days^{-1} is the fraction of activity present in

Table 2.1. Gastrointestinal Tract Model of Reference Man

Portion of GIT	Mass of Wall (g)	Mass of Contents (g)	Average Time Food Remains (Days)
Stomach	150	250	1/24
Small Intestine	640	400	4/24
Upper Large Intestine	210	220	13/24
Lower Large Intestine	160	135	24/24

the *SI* which is absorbed to blood per unit time, and f_1 is the fraction of the material ingested which is absorbed to blood.

There are four differential equations corresponding to the GIT for each radionuclide in the chain. For n=0, $X^n_\xi \equiv 0$ which simplifies the differential equations for the parent. Initial conditions for the GIT following intake by inhalation are $X^n_\xi(0) = 0$ for all n and all subscripts ξ. This is easily adapted for oral intake.

The cumulated activity (μCi-days) from time 0 to time t_1 for each section for the GIT is computed by

$$U^n_\xi = Y^n_\xi = \int_0^{t_1} X^n_\xi \, dt \; . \qquad (2.6)$$

Blood and Other Organs

The activity transferred from the lungs or from GIT to blood will probably deposit in some other organs. For intake of 1 μCi to blood

$$R^n_j(t) = \sum_{s=1}^{q^n_j} a^n_{js} e^{-(\lambda^n_{js} + \lambda^n_r)t} \; , \qquad j = 1, 2, \ldots, k, \quad k \equiv \text{the total number of organs} \qquad (2.7)$$

represents retention of the j^{th} organ for the n^{th} radionuclide. The coefficients a^n_{js} and the exponential constants λ^n_{js} and λ^n_r are supposedly known. The retention functions $R^n_j(t)$ may be the results of fitting sums of exponentials to retention data on man or on experimental animals. They may also be the results of a compartmental analysis of the problem.

Let X^n_{js} represent retention of the n^{th} radionuclide in the compartment which corresponds to the s^{th} term in the sum for the j^{th} organ. Then

$$
\begin{aligned}
\dot{X}_{js}^{n+1} = {} & -X_{js}^{n+1}(\lambda_{js}^{n+1} + \lambda_{r}^{n+1}) \\
& + a_{js}^{n+1}(X_{a}^{n+1}\lambda_{a}^{n+1} + X_{c}^{n+1}\lambda_{c}^{n+1} + X_{e}^{n+1}\lambda_{e}^{n+1} + X_{i}^{n+1}\lambda_{i}^{n+1} + X_{SI}^{n+1}\lambda_{ab}^{n+1}) \\
& + a_{js}^{n+1}\sum_{l=1}^{k}\sum_{m=1}^{q_l^n} X_{lm}^{n}\lambda_{r}^{n+1}I_{lm}^{n+1} \qquad (2.8) \\
& + \left[\sum_{m=1}^{q_j^n} X_{jm}^{n}\,\lambda_{r}^{n+1}(1 - I_{jm}^{n+1})\right] a_{js}^{n} \Big/ \sum_{p=1}^{q_j^{n+1}} a_{jp}^{n+1}
\end{aligned}
$$

where I_{lm}^{n+1} is the fraction of activity of the n+1st daughter that recirculates to blood, and

$$\sum_{j=1}^{k}\sum_{s=1}^{q_j^n} a_{js}^{n} = 1 \ .$$

There are $\sum_{i=1}^{r}\sum_{j=1}^{k} q_j^i$ differential equations for other organs for each radionculide in the chain. For the parent, $X_{js}^{0} \equiv 0$ which simplifies the differential equations for the parent. Initial conditions for other organs following intake by inhalation or ingestion are $X_{js}^{n}(0) = 0$ for all n and subscripts j,s. This is easily adapted for intake by injection.

The cumulated activity from time 0 to time t_1 for some compartment j,s of the body is given by

$$Y_{js}^{n} = \int_0^{t_1} X_{js}^{n}\, dt \ . \qquad (2.9)$$

For a given j these are summed to form

$$U_j^n = \sum_{s=1}^{J_j^n} V_{js}^n \; . \tag{2.10}$$

In many cases equation (2.7) is given for k organs, one of which is termed "other tissues". When this occurs, the following transformation is performed on the values (2.10). Assuming the kth organ is "other tissues", $U_k^n = U_{other}^n$, "other tissues" is transformed to total body as follows:

$$U_{TB}^n \leftarrow U_{other}^n \cdot \frac{M_{TB}}{M_{TB} - \sum_{j=1}^{k-1} M_j} \tag{2.11}$$

where M_{TB} is the mass of the total body and M_j is the mass of organ j. All other organs $i = 1, \ldots k-1$ are transformed as follows:

$$U_i^n \leftarrow U_i^n - U_{other}^n \cdot \frac{M_i}{M_{TB} - \sum_{j=1}^{k-1} M_j} \; . \tag{2.12}$$

III. COMPUTATIONAL CONSIDERATIONS

Suppose we wish to determine the activity (μCi) from inhalation of 1 μCi of a radionuclide which decays to a chain of daughter radionuclides having total length n. The problem may be expressed in the form

$$\dot{X} = AX \tag{3.1}$$

given the initial condition $X(t_0) = X(0)$ where the vectors $\dot{X}$ and X may be partitioned as

$$\dot{X} = \begin{bmatrix} \dot{X}^1 \\ \dot{X}^2 \\ \vdots \\ \dot{X}^n \end{bmatrix}, \quad X = \begin{bmatrix} X^1 \\ X^2 \\ \vdots \\ X^n \end{bmatrix}, \tag{3.2}$$

and each vector $\dot{X}^i$ and X^i may be written as

$$\dot{X}^i = \begin{bmatrix} \dot{X}^i_a \\ \vdots \\ \dot{X}^i_l \\ \dot{X}^i_S \\ \vdots \\ \dot{X}^i_{LL} \\ \dot{X}^i_{11} \\ \vdots \\ \dot{X}^i_{kq^n_k} \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \text{Lungs} \\ \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \text{GIT} \\ \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \begin{matrix}\text{Other} \\ \text{Organs}\end{matrix} \end{matrix} , \quad X^i = \begin{bmatrix} X^i_a \\ \vdots \\ X^i_l \\ X^i_S \\ \vdots \\ X^i_{LL} \\ X^i_{11} \\ \vdots \\ X^i_{kq^n_k} \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \text{Lungs} \\ \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \text{GIT} \\ \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \begin{matrix}\text{Other} \\ \text{Organs}\end{matrix} \end{matrix} \tag{3.3}$$

where k is the number of organs, other than lungs and GIT, for which retention information is available. Thus, the differential equations will be ordered: lungs, GIT, other organs - parent; lungs, GIT, other organs - 1st daughter, etc. Then the matrix A may be partitioned in correspondence with $\dot{X}$ and X as expressed in (3.2) as

$$A = \begin{bmatrix} A_{11} & & & & \\ A_{21} & A_{22} & & 0 & \\ & A_{32} & A_{33} & & \\ & & \ddots & \ddots & \\ & 0 & & A_{n,n-1} & A_{n,n} \end{bmatrix} \tag{3.4}$$

where A_{ii} is a lower triangular matrix with negative diagonal elements. $A_{i,i-1}$ is rectangular having the same number of rows as A_{ii} and the same number of columns as $A_{i-1,i-1}$.

In order to better visualize the form of A, consider an example, for inhalation, with a chain of two radionuclides and two organs where

$$R_1^1(t) = a_{11}^1 \, e^{- \lambda_{11}^1 + \lambda_r^1)t} ,$$

$$R_2^1(t) = a_{21}^1 \, e^{-(\lambda_{21}^1 + \lambda_r^1)t} ,$$

and (3.5)

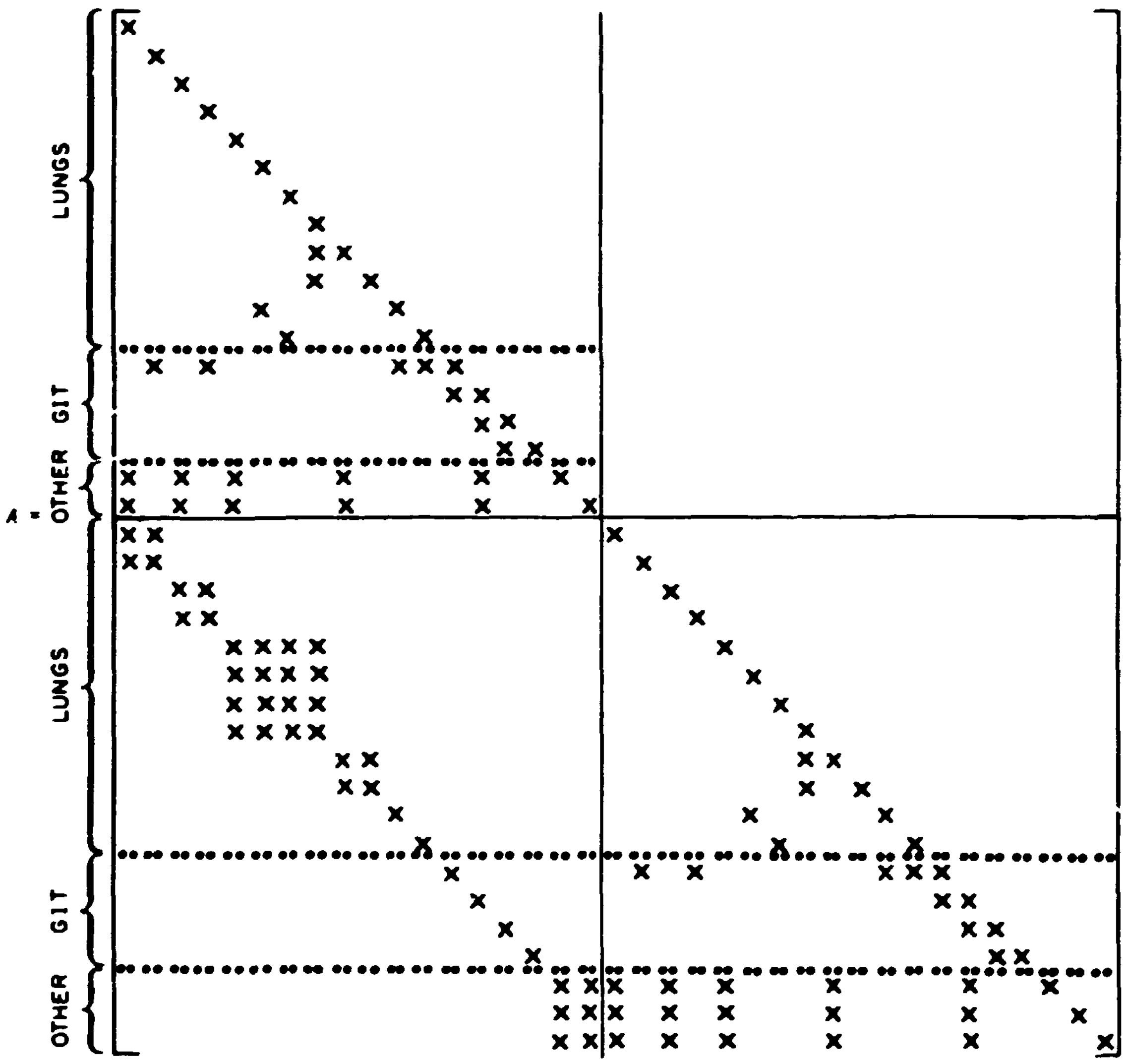
$$R_1^2(t) = a_{11}^2 \, e^{-(\lambda_{11}^2 + \lambda_r^2)t} + a_{12}^2 \, e^{-(\lambda_{12}^2 + \lambda_r^2)t} ,$$

$$R_2^2(t) = a_{21}^2 \, e^{-(\lambda_{21}^2 + \lambda_r^2)t} .$$

describe retention R_j^i for nuclide i organ j.

There will be 37 (2•12 + 2•4 + 2 + 3) differential equations and matrix A will have the form

ORNL-DWG 76-12737



(3.6)

The x's indicate nonzero entries in the matrix; the dotted lines are present to help the reader distinguish visually the coefficients of the terms in the differential equations for the lungs, GIT, and other tissues for both the parent and the daughter; the solid lines indicate where the matrix should be partitioned in order to be of the form

$$A = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix} .$$

Computer Storage of the Coefficient Matrix

It is evident from Eq. (3.6) that the coefficient matrix A is a large, sparse (i.e., few nonzero entries) matrix. For computational purposes the matrix is stored in the computer as three vectors. The first vector contains the nonzero elements of A stored by rows. The second and third vectors hold the row and column index, respectively, for each of the nonzero elements of A. The method of storage is required due to the fact that even for a short chain of radionuclides the coefficient matrix becomes too large to store reasonably the entire matrix in the computer.

Other Modes of Intake

Ingestion or injection of 1 μCi of a radionuclide may be achieved by a slight modification of the system of differential equations. For ingestion, remove from the system those equations for the lungs and modify the initial conditions so that 1 μCi is deposited in the stomach.

$$\begin{aligned} X^1_S(0) &= 1.0 \quad , \\ X^1_{SI}(0) &= X^1_{UL}(0) = X^1_{LL}(0) = 0.0 \ . \end{aligned} \tag{3.7}$$

For injection, remove from the system those equations describing lung and GIT retention, and alter the initial condition vector so that

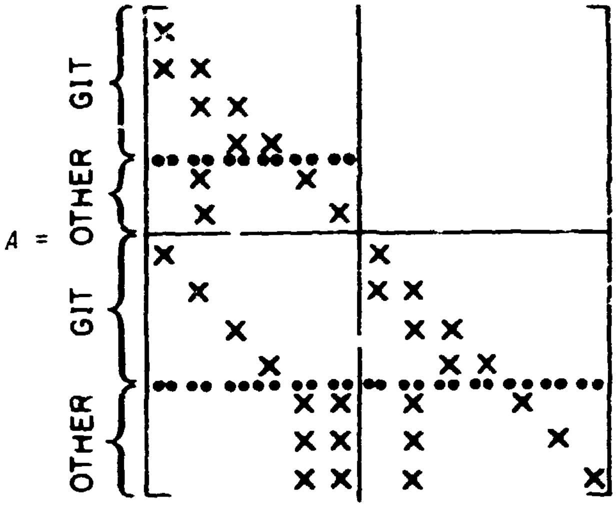
$$X^1_{js}(0) = a^1_{js} \Big/ \sum_{i=1}^{k} \sum_{t=1}^{q^1_i} a^1_{it} \tag{3.8}$$

where k is the number of organs and q^1_i is the number of terms in the retention function corresponding to the ith organ. Thus for ingestion and injection (3.3) reduces to (3.9) and (3.10), respectively,

$$\text{for ingestion} - \quad \dot{X}^i = \begin{bmatrix} \dot{X}^i_S \\ \vdots \\ \dot{X}^i_{LL} \\ \dot{X}^i_{11} \\ \vdots \\ \dot{X}^i_{kq^n_k} \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \text{GIT} \\ \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \begin{matrix} \text{Other} \\ \text{Organs} \end{matrix} \end{matrix} , \quad X^i = \begin{bmatrix} X^i_S \\ \vdots \\ X^i_{LL} \\ X^i \\ \vdots \\ X^i_{kq^n_k} \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \text{GIT} \\ \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \begin{matrix} \text{Other} \\ \text{Organs} \end{matrix} \end{matrix} \tag{3.9}$$

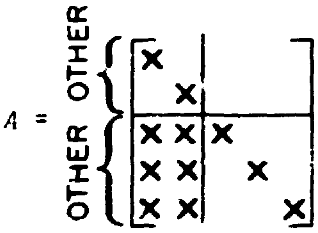
$$\text{for injection} - \quad \dot{X}^i = \begin{bmatrix} \dot{X}^i_{11} \\ \vdots \\ \dot{X}^i_{kq^n_k} \end{bmatrix} \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \begin{matrix} \text{Other} \\ \text{Organs} \end{matrix} , \quad X^i = \begin{bmatrix} X^i_{11} \\ \vdots \\ X^i_{kq^n_k} \end{bmatrix} \left.\vphantom{\begin{matrix} a \\ b \\ c \end{matrix}}\right\} \begin{matrix} \text{Other} \\ \text{Organs} \end{matrix} \tag{3.10}$$

For the example shown in Eq. (3.6) the matrix A would be reduced to



(3.11)

for ingestion, and



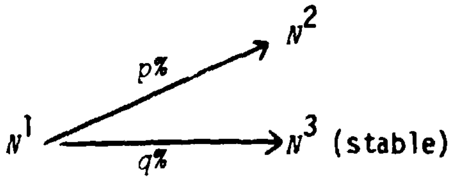
(3.12)

for injection.

Branching in the Chain of Radionuclides

When the decay chain involves branching, the coefficient matrix takes a different form than that shown in Eq.(3.4). The authors have considered three forms of branching. By numbering the nuclides as they will be ordered in Eq. (3.2), we have

(1)



Then the partitioned matrix A takes the form

$$A = \begin{bmatrix} A_{11} & 0 \\ \frac{p}{100} \cdot A_{21} & A_{22} \end{bmatrix} \tag{3.13}$$

where $\frac{p}{100} \cdot A_{21}$ indicates that every element in the matrix A_{21} is multiplied by $\frac{p}{100}$. *If a chain has the form*

(2)

$$\begin{tikzcd} & N^2 \arrow[dr] & & \\ N^1 \arrow[ur, "p\%"] \arrow[rr, "q\%"'] & & N^3 \arrow[r] & N^4 \text{ (stable)} \end{tikzcd} ,$$

then

$$A = \begin{bmatrix} A_{11} & 0 & 0 \\ \frac{p}{100} \cdot A_{21} & A_{22} & 0 \\ \frac{q}{100} \cdot A_{31} & A_{32} & A_{33} \end{bmatrix} . \tag{3.14}$$

The chain

(3)

$$\begin{tikzcd} & N^2 \arrow[dr] & & \\ N^1 \arrow[ur, "p\%"] \arrow[dr, "q\%"'] & & N^4 \arrow[r] & N^5 \text{ (stable)} \\ & N^3 \arrow[ur] & & \end{tikzcd}$$

has the corresponding coefficient matrix:

$$A = \begin{bmatrix} A_{11} & 0 & 0 & 0 \\ \frac{p}{100}\cdot A_{21} & A_{22} & 0 & 0 \\ \frac{q}{100}\cdot A_{31} & 0 & A_{33} & 0 \\ 0 & A_{42} & A_{43} & A_{44} \end{bmatrix} . \qquad (3.15)$$

Certainly these branching schemes do not stand alone or occur only at the beginning of a chain. Thus, the coefficient matrix A will, in general, be some combination of (3.4) and (3.13), (3.14), or (3.15). These examples have sufficed to the present, but it is possible other forms may occur. However, the general pattern seems clear from these examples.

Calculation of Cumulated Activity

It should be noted that, in general, we do not wish to obtain the activity (μCi) as the final result but cumulated activity (μCi-days) which represents the time integral of the solution to (3.1). This is done in a very straightforward manner.

Given $\dot{X}(t) = AX(t)$, Eq. (3.1), with initial condition $X(0) = X_0$ and integrating both sides we have

$$\int_0^{t_1} \dot{X}(t)dt = \int_0^{t_1} AX(t)dt \quad ,$$

$$X(t_1) - X_0 = A\int_0^{t_1} X(t)dt \quad ,$$

$$X(t_1) = A\int_0^{t_1} X(t)dt + X_0 \; .$$

By letting $Y(t_1) = \int_0^{t_1} X(t)dt$ where $Y(0) = 0$,

$$\dot{Y}(t_1) = \left[\int_0^{t_1} X(t)dt\right]^{\cdot} = X(t_1) \ .$$

Thus we may reformulate (3.1) as

$$\dot{Y} = AY + X_0 \quad \text{where } Y(0) = 0 \ , \tag{3.16}$$

and (3.16) will be solved for Y to determine the cumulated activity (μCi-days) in that organ from 0 to t_1. It is Eq. (3.16), not Eq. (3.1) to which TIMED calculates the solution.

Solution of the Differential Equations

The solution of (3.16) is calculated by using a FØRTRAN subroutine package written by A. C. Hindmarsh,[8] based on a program written by C. W. Gear,[9] for the solution of the initial value problem for system of ordinary differential equations (ØDE's). In general, such a system has the form

$$\dot{y} = f(y,t) \tag{3.17}$$

or more specifically

$$\frac{dy_i(t)}{dt} = f_i(y_1(t),\ldots,y_N(t),t) \tag{3.18}$$

where y, $\dot{y}$, and f are vectors of length $N \geq 1$. Given an initial value of the vector

$$y(t_0) = y_0,$$

and a subroutine for the calculation of f, the GEAR package can be used to compute a numerical solution to (3.18) at values of the independent variable t in some interval (t_0,T), as desireo by the user.

The basic methods used for the solution are of implicit linear multistep type. There are classes of such methods available to the user. The first is the implicit Adams methods (up to order 12), and the second is the backward differentiation formula (BDF) methods (up to order 5), also called Gear's stiff methods.

A prime feature of GEAR is the ability to solve stiff ØDE problems. Roughly speaking, an ØDE system is called stiff if it involves both very rapidly changing terms and very slowly changing terms, all of a decaying nature. More precisely, we consider the eigenvalues ν_i of the $N \times N$ Jacobian matrix

$$J = \frac{\partial f}{\partial y} = \left(\frac{\partial f_i}{\partial y_j}\right)_{i,j=1}^{N} \tag{3.19}$$

and suppose that the ν_i all have negative real parts. The "time constants" of the problem are then $\tau_i = 1/|\mathrm{Re}(\nu_i)|$, and the decaying nature (locally) of the solution is given by the exponentials e^{-t/τ_i}. If the N time constants τ_i are widely spread, and those terms with the smaller τ_i have already decayed to an insignificant level, then the system is stiff.

upon considering the properties of Eq. (3.16), it can be seen that the Jacobian of the matrix A is itself (that is, $J(A) = A$), and, A being a nonsingular, lower triangular matrix, the eigenvalues of A are the diagonal elements of the matrix, which are $-(\lambda_\xi^j + \lambda_r^j)$. Thus the time constants τ_i are

$$\tau_i = \frac{1}{\lambda_\xi^j + \lambda_r^j} \text{ , } i = 1,\ldots,N$$

where ξ indicates the compartment and j indicates the member of the chain. The τ_i are of greatly differing orders of magnitude, due to the differing values of the radiological decay constants, λ_r^j, for each nuclide j, and also the differing biological decay constants, λ_ξ^j, for various compartments ξ. This implies the system is stiff.

The difficulty with stiff problems is that most conventional methods for solving ØDE's require incremental values of t commensurate with min τ_i, while the size $|T-t_0|$ of the problem range is commensurate with max τ_i. As a result, the problem cannot be run to completion in a reasonable number of steps. With Gear's methods, however, the time increment h is restricted to small values, by the requirement of accuracy, only where the solution is relatively active. By definition, the problem is not stiff in such regions, and accuracy is achieved at minimum computational cost by allowing both h and the order of the method to vary. Then in regions of stiffness, where the solution is inactive, Gear's methods have the property of "stiff stability", which assures that h is no longer restricted by small time constants, unless or until the corresponding, rapidly decaying terms become active again.

The GEAR package also contains, as an option, a method well suited for nonstiff problems, namely the implicit Adams methods with fixpoint corrector iteration, also called the Adams-Bashforth-Moulton methods. Both the stiff and nonstiff methods are implemented in a manner which allows both the step size and the order to vary in a dynamic way throughout the problem. This variability is now widely recognized as highly desirable for efficiency in using linear multistep methods.

The methods used in the GEAR package are documented in considerable detail elsewhere.[9,10,11] Hence, only a brief summary of them will be given here.

The basic methods involved are linear multipoint methods of the form

$$y_n = \sum_{j=1}^{K_1} \alpha_j y_{n-j} + h \sum_{j=0}^{K_2} \beta_j \dot{y}_{n-j} , \tag{3.20}$$

where y_k is an approximation to $y(t_k)$, $\dot{y}_k = f(y_k,t_k)$ is an approximation to $\dot{y}(t_k)$, and h is a constant step size: $h = t_{k+1} - t_k$. In the case of the Adams method of order q, we have $K_1 = 1$ and $K_2 = q - 1$. In the case of the backward differentiation formula (BDF) of order q, we have $K_1 = q$ and $K_2 = 0$. The BDF's are so called because, on dividing through by $h\beta_0$, they can be regarded as approximation formulas for $\dot{y}_n$ in terms of $y_n, y_{n-1},\ldots,y_{n-q}$. In either case, the α_j and β_j are constants associated with the method and $\beta_0 > 0$. The latter means that Eq. (3.20) is an implicit equation for y_n and is, in general, a nonlinear algebraic system that must be solved at every step. The fact that the order of a given method is q means that, if Eq. (3.20) is solved for y_n with all past values being exact, then y_n will differ from the correct solution of the ØDE by a local truncation error that is of order h^{q+1}, commonly expressed as $O(h^{q+1})$, for small h.

If Eq. (3.20) is written in the form

$$g(y_n) \equiv y_n - h\beta_0 f(y_n, t_n) - \sum_1^{K_1} \alpha_j y_{n-j} - h \sum_1^{K_2} \beta_j \dot{y}_{n-j} = 0, \qquad (3.21)$$

then the nonlinear system $g(y_n) = 0$ can be solved, for example, by Newton's method:

$$\begin{aligned} y_{n(m+1)} &= y_{n(m)} - P^{-1}_{n(m)} g(y_{n(m)}) \, , \\ P_n(m) &= \left.\frac{\partial g}{\partial y}\right|_{y_n(m)} = I - h\beta_0 \left.\frac{\partial f}{\partial y}\right|_{y_n(m)} \, . \end{aligned} \qquad (3.22)$$

The manner in which past values are saved is rather unusual by comparison with most other ØDE programs and was invented by A. Nordsiech.[12] While the conventional choice would be to store an array of $L = K_1 + K_2 + 1 = q + 1$ current and past values of y_k and $\dot{y}_k$, Nordsiech's history is a linear transform of this one and has the form

$$\underline{z}_n = (y_n, h\dot{y}_n, h^2\ddot{y}_n/2, \ldots, h^q y_n^{(q)}/q!) \; . \qquad (3.23)$$

This is an $N \times L$ array of approximate "scaled derivatives" of y at t_n up to order q and is the Y array of the program.

With this choice of history array, the prediction of y_n (and in fact all of $\underline{z}_n$) from $\underline{z}_{n-1}$ becomes quite simple. It is given by

$$\underline{z}_{n(0)} = \underline{z}_{n-1}A, \quad A = \left(\binom{i}{j}\diagdown^{0}\right), \text{ that is } \begin{cases} a_{ij} = \dfrac{i!}{j!(i-j)!} & i \geq j \\ a_{ij} = 0 & i < j \end{cases} \tag{3.24}$$

A is the $L \times L$ Pascal triangle matrix. As Gear observed, the multiplication $\underline{z}A$, for a row vector $\underline{z} = (z_i)_0^q$ can be accomplished with additions only, as follows:

$$\underline{z}_{j-1} \leftarrow \underline{z}_{j-1} + \underline{z}_j \tag{3.25}$$

where $j = q, q-1, \ldots, i-1$ and $i = 0, 1, \ldots, q-1$.

The complete algorithm for the prediction and corrections at step n can then be written as follows:[10]

$$\underline{z}_{n(0)} = \underline{z}_{n-1}A ,$$

$$\left.\begin{aligned} y_{n(m+1)} &= y_{n(m)} + \ell_0 P_n^{-1} F_m , \\ h\dot{y}_{n(m+1)} &= h\dot{y}_{n(m)} + P_n^{-1} F_m , \\ F_{(m)} &\equiv hf(y_{n(m)}, t_n) - h\dot{y}_{n(m)} , \end{aligned}\right\} (m=0,1,\ldots,M-1) \tag{3.26}$$

$$y_n = y_{n(M)} = y_{n(0)} + \ell_0 E_n ,$$

$$E_n = \sum_{m=0}^{M-1} P_n^{-1} F_m ,$$

$$\underline{z}_n = \underline{z}_{n(0)} + E_n \underline{\ell} .$$

Here M is the number of iterations performed, and $\underline{\ell} = (\ell_i)_0^q$ is a row vector of constants determined by the basic method, Eq. (3.20), and satisfying $\ell_0 = \beta_0$ and $\ell_1 = 1$. The vector E_n is saved, as it is proportional to the estimated truncation error committed on the step.[11]

Following a step of size h at order q, the GEAR package, at intervals of $q + 2$ steps, is programmed to choose a larger step size by estimating the local truncation errors at orders q-1, q, and q+1. The largest value h' of the three step sizes obtained is then chosen, and the order reset accordingly. Also the Nordsiech array must be rescaled by powers of h'/h. The data used to take the subsequent steps of size h' is in effect obtained by interpolating with the data at a spacing of h.

BLANK PAGE

IV. PROGRAM DESCRIPTION

The program TIMED consists of sixteen routines that are programmed in a fairly well-structured manner and are heavily augmented with program comment cards. Hence, to anyone who is familiar with the methods used in this report and in Refs. 8, 9, 10, and 11, the details are self-explanatory. However, to supplement this material, we include the diagram of program structure and a description of each routine.

Fig. 4.1 shows the overall structure of the TIMED program. Arrows indicate the calling sequence of the routines. Routines DRIVES, STIFF, YØUT, YØUTD, YNA, and TESØC have been taken from the GEAR package[8] and have been specialized for use in TIMED by D. E. Arnurius at Oak Ridge National Laboratory.[13] MAIN, YPAY, and AXEB have been written by him for use on this particular problem with our computer, the IBM System/360. The remainder of the routines have been written by one author (S. B. Watson) to implement the model described in Section II.

Except for AXEB and YPAY, all routines are written in the IBM System/360 and System/370 FØRTRAN IV language. AXEB and YPAY are IBM System/360 Assembler language routines. A listing of TIMED routines is given in Appendix A.

The MAIN routine functions as follows:

1. Read the values NØ, TLAST, H1, EPS.

   NØ - the number of first order differential equations.

   TLAST - the final value of T, i.e. integration proceeds from T = TØ to T = TLAST.

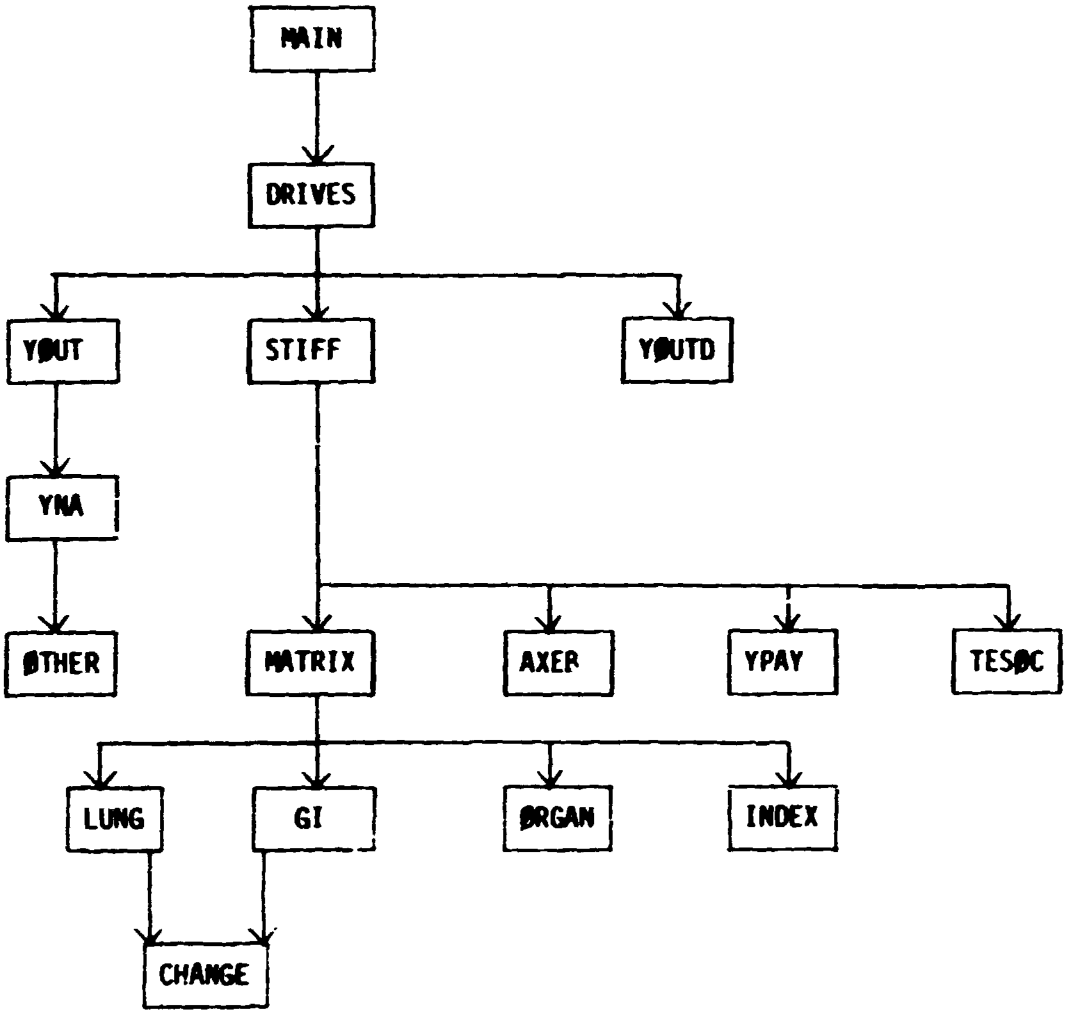


Fig. 4.1. Overall Structure of TIMED.

H1 - initial step size for the integration procedure.

EPS - the local error tolerance parameter. The estimated errors δ_i in y_i are compared to YMAX(i), which is roughly the largest value of $|y_i|$ seen so far, for error control purposes this is done by comparing the root-mean-square (RMS) norm (i.e. Euclidean norm divided by $\sqrt{N}$ of the vector $[\delta_i/\mathrm{YMAX}(i)]_{i=1}^{N}$) to EPS. This ratio is kept less than EPS.

2. Read the number of print intervals, NPRT, and the arrays defining the print times, TPRL, DTPR, and NARPRT by

```
      DØ 10 I = 1,NPRT
   10 TPRINT  = TPRL(I) + Q*DTPR(I)
```

   where Q = 0,...,NARPRT(I) .

3. Check to see if NPRT $\leq$ 10.
4. Check to see if TPRL(NPRT) + NARPRT(NPRT)*DTPR(NPRT) = TLAST.
5. Check to see that consecutive time intervals do not overlap.
6. Set TPRINT to TPRL(1,.
7. Check to see that 13*NØ $\leq$ 2400 for Adams procedure (i.e. NØ $\leq$ 126) or 6*NØ $\leq$ 2400 for stiff procedure (i.e. NØ $\leq$ 400).
8. Call DRIVES.
9. STØP.

The routine DRIVES is an interface between the user and the rest of the GEAR package. It oversees the integration of the ØDE over the interval between two of the user's output points. DRIVES proceeds as follows:

1. Test input parameters for correctness and initialize variables.

2. Call STIFF.

3. KFLAG set in STIFF and returned to DRIVES as one of the following values:

   0 problem was completed successfully.

   -1 the integration was halted after failing to pass the error test even after reducing H by a factor of 10^{10} from its initial value.

   -2 after some initial success, the integration was halted when a reduction in H by a factor of more than 10^4 was indicated in order to pass the error test.

   -3 the integration was halted after failing to achieve corrector convergence even after reducing H by a factor of 10^{10} from its initial value.

   -4 immediate halt because of illegal values of input parameter.

   -5 H is such that T + H = T.

4. If KFLAG = 0 or -4, call YØUT for normal output of Y at time TPRINT; go to 2.

5. If KFLAG = -1, print message; try 10 more reductions of H; call YØUTD if continues to fail; return.

6. If KFLAG = -2, -3 -5, print error message; call YØUTD; return.

YØUT computes interpolated values of the dependent variable Y, and calls YNA to print these values. YNA prints the solution vector Y, Eq. (3.16), and also computes Eqs. (2.4), (2.6), and (2.10) and prints these quantities. YNA calls the routine ØTHER to perform the "other tissues" transformation described in (2.11) and (2.12), should it be required. YØUTD is a dummy routine which could be expanded by the user to print diagnostic information.

STIFF performs one step of the integration of an initial value problem for a system of ordinary differential equations. Fig. 4.2 shows the general flow structure of the routine.

Recalling the unusual structure of the coefficient matrix, A, in (3.1) as discussed in Section III, A is stored as a vector which is limited in length by STIFF to a maximum of 6000 elements. This means that the user is limited to 6000 nonzero elements in the coefficient matrix. A check is made to see that sufficient storage is available for A. STIFF also counts the number of nonzero elements in each row of A and writes an error message if this number exceeds 52. It should be noted that some of the methods for storing elements in certain arrays are both compiler- and machine-dependent. This may cause problems on machines other than IBM System/360 or System/370.

AXEB is an IBM 360 Assembler language routine called by STIFF to solve the matrix equation $AX = b$ where A is an N by N lower triangular matrix, and A is stored as a vector consisting of the nonzero entries in A stored sequentially by rows. It is assumed that the maximum number of nonzero entries in a row of A is less than or equal to 52.

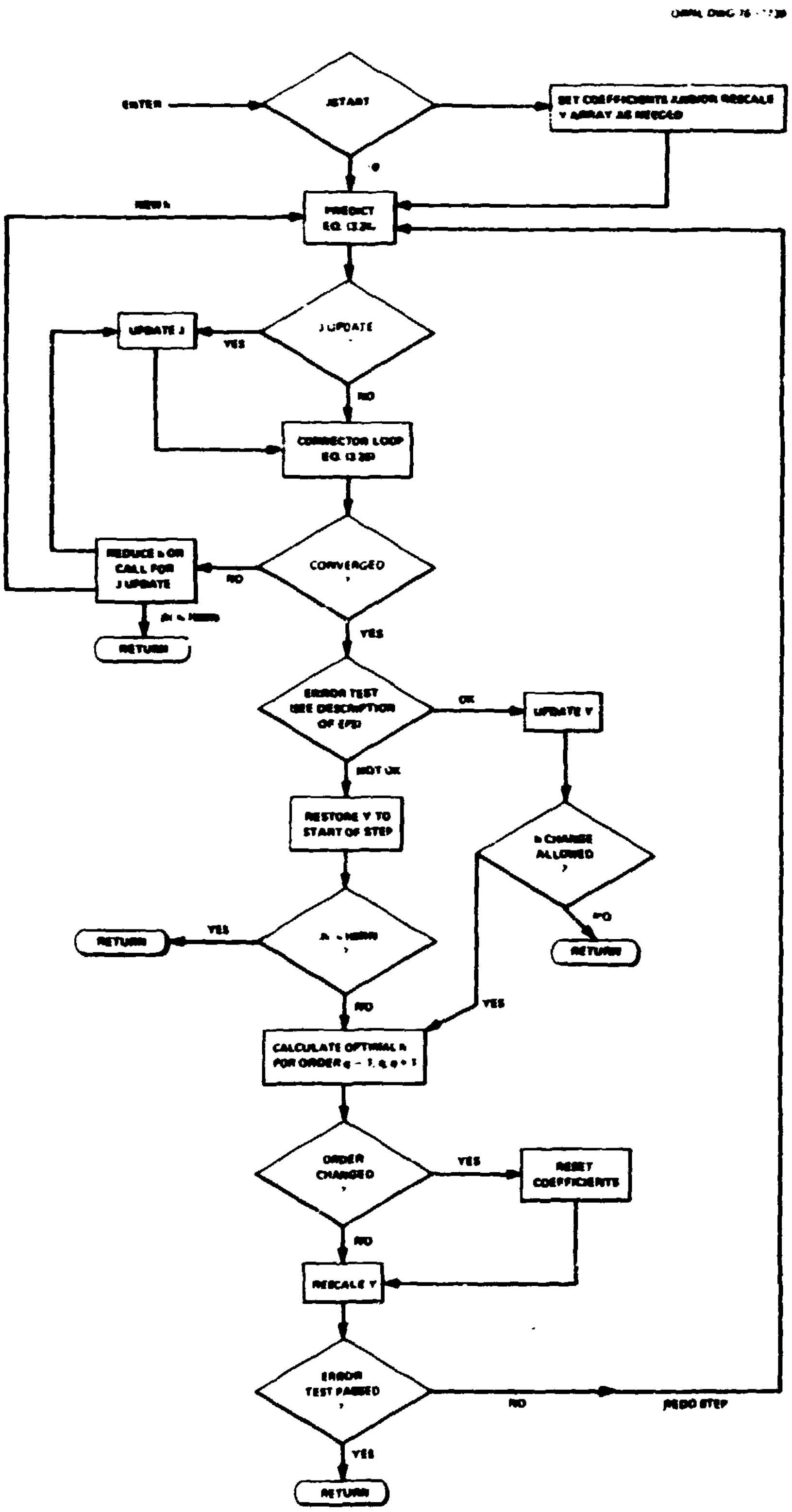


Fig. 4.2. Macroscopic Flow Chart of STIFF (Taken from Ref. 8).

YFAY is also an Assembler language routine. It is called by STIFF to form $\dot{Y} = AY + X_0$. The matrix A is stored as in AXEB, and this routine assumes a maximum of 52 nonzero entries.

AXEB and YPAY were written so as to most efficiently make use of the properties of the matrix A and of the computer storage.

TESØC is called by STIFF and sets the coefficients that are used there, both for the basic integration step and for error control. TESØC also sets MAXDER, the maximum order of the method available. Currently it is 12 for Adam's method and 5 for the GEAR method.

MATRIX directs the formation of the coefficient matrix, A, and may be outlined as follows:

1. Read descriptive information about the chain of radionuclides.

   NISØ - number of radionuclides in the chain

   D3, D4, D5 - regional deposition values[4]

   NØRG - number of organs (excluding lung and GIT)

   IØPT - mode of intake (0 - inhalation, 1 - ingestion,

   2 - injection)

2. Call LUNG to assign regional deposition for each pathway a,...,h (see Fig. 2.1 for explanation of lung pathways).

3. Read for each radionuclide

   TAG - atomic number and symbol

   TR - half life

   ICLASS - inhalation class

   ILAB - (0 - absorption in SI only, 1 - absorption in

   other regions of GIT

4. Set LAMR.

   LAMR = $ln2/TR$

5. Call LUNG1 to set LAMX and FRAC according to inhalation class.

   LAMX = $\lambda_\xi = ln2/T_\xi$, T_ξ is the removal half time described in Fig. 2.1

   FRAC = F_ξ is the compartmental fraction from Fig. 2.1

6. Initialize row and column indices.

7. For each nuclide:

   (a) If intake by inhalation, call LUNG2 to set up portion of matrix corresponding to lung differential equations.

   (b) If intake is by inhalation or ingestion, call routine GI to set up portion of matrix corresponding to GIT differential equations.

   (c) Call ØRGAN to set up coefficient matrix for differential equations corresponding to organs other than lungs and GIT for which we have retention mode's.

8. Print coefficient matrix.

9. Print nonzero elements of the coefficient matrix and the row and column in which each one appears. Recall that, due to the sparse nature of the coefficient matrix, it is stored as three vectors - A, RINDEX, and CINDEX. A(I) is a nonzero element of the coefficient matrix, RINDEX(I) contains the row of A(I), and CINDEX(I) is the column index of A(I). The coefficient matrix is filled from left to right, one row at

a time, proceeding from top to bottom of the matrix, so that the vector A is formed correctly. Note that the components of the vector RINDEX form an increasing sequence, i.e., RINDEX(I) $\leq$ RINDEX(I+1) while the components of CINDEX do not.

Routine LUNG has three entry points which are called from MATRIX and function as follows:

Entry LUNG - assigns the regional deposition values for each pathway (these are used in setting the initial values for inhalation, see Eq. (2.2)).

Entry LUNG1 - determines inhalation class and sets λ_ξ and F_ξ from Fig. 2.1 which appear in Eq. (2.1). LUNG1 also sets the initial values, Eq. (2.2).

Entry LUNG2 - forms the coefficient matrix for Eq. (2.1).

Routine GI is called from MATRIX and determines the entries of the coefficient matrix corresponding to the compartments of the GIT, Eq. (2.5). If intake is by ingestion, initial conditions are set as in Eq. (3.7).

ØRGAN functions as follows:

1. For each organ, read organ number (ISØRS), the organ name (ØRGNM), and the number of compartments (ICOM); i.e., the number of terms in the retention function (see Eq. (2.7)), and the retention function.
2. Form the entries of the coefficient matrix corresponding to Eq. (2.8) for each compartment.

3. If intake is by injection, set the initial values according to Eq. (3.8).

INDEX is called by MATRIX and determines the first column of the matrix entries for each daughter. (Recall the way the matrix entries are filled — from left to right and top to bottom.) This is especially important when branching is considered. Whereas daughter entries begin in a regularly indented fashion for nonbranching radionuclides, Eq. (3.4), the daughter entries have a somewhat irregular pattern in branching chains, see Eqs. (3.13), (3.14), and (3.15).

CHANGE is called by LUNG and GI and determines the initial column for each row of the coefficient matrix for the equations corresponding to N^3, Eq. (3.14), and the equations corresponding to N^4, Eq. (3.15).

V. USER INFORMATION

TIMED is written in the IBM System/360 and System/370 FØRTRAN IV language with the exception of two routines (AXEB and YPAY which are discussed in Section IV) that are written in IBM System/360 Assembler language. The FØRTRAN routines have been compiled using the FØRTRAN H-level compiler; the Assembler routines have been compiled with the Assembler F-level compiler. TIMED has been executed on both the IBM 360/91 and 360/75 at ORNL. The program requires about 310 K (1K = 1024 bytes; 1 byte = 8 bits) of core storage for execution. The FØRTRAN routines require about 7.5 seconds to compile; the Assembler routines require 0.7 seconds for compilation. Execution time depends upon many factors: the number of differential equations, the number of nonzero entries in the coefficient matrix, the length of time period (t_0, t_{final}) for integration, etc. The execution time for the second example in Appendix B, a complicated chain of four radionuclides involving 116 differential equations, is about 5.8 seconds.

The remainder of this section describes the content of the card input which must be prepared in order to execute the program and the output produced by the program. Three examples of sample input and the corresponding output are presented in Appendix B.

Input

The card input data are to be prepared in the order shown in Table 5.1. Referring to Table 5.1, "Subroutine" indicates the subroutine in

Table 5.1. Card Input to TIMED

Subroutine	Card Number	Columns	Remarks, Variables, Etc.
Information Concerning the Differential Equations			
MAIN	1		FØRMAT (I5, 7E10.2)
		1-5	NØ - order of lower triangular matrix A, that is, the number of differential equations.
		6-15	TLAST - integrate $Y'(t) = AY(t) + X_0$ from $t = 0$ to $t =$ TLAST. TLAST given in units of days.
		16-25	H1 - initial step size for the integration procedure. If left blank, then default is H1 = 10^{-7}.
		26-35	EPS - controls the local truncation error during the integration procedure. If left blank, then default is EPS = 10^{-6}.
	2		FØRMAT (I5)
		1-5	NPRT - number of print intervals, NPRT $\le$ 10.
	2+I, I=1,...,NPRT		FØRMAT (2E10.2, I5)
		1-10	TPRL(I) - first print time in this print interval.
		11-20	DTPR(I) - print time step for this interval.
		21-25	NARPRT(I) - number of time steps to use in the current interval.

Table 5.1. (cont'd)

<table>
<tr><th>Subroutine</th><th>Card Number</th><th>Columns</th><th>Remarks, Variables, Etc.</th></tr>
<tr><td colspan="4">TPRL, DTPR, and NARPRT are defined by:
DØ 10 I = 1, NPRT
10 TPRINT = TPRL(I) + Q*DTPR(I)   Q = 0,...,NARPRT(I)
The values assumed by TPRINT must form an increasing sequence.
<u>General Information for Chain of Radionuclides</u></td></tr>
<tr><td>MATRIX</td><td>1</td><td></td><td>FØRMAT (I10, 3F10.3, 3I10)</td></tr>
<tr><td></td><td></td><td>1-10</td><td>NISØ - number of isotopes in the chain, NISØ $\le$ 10.</td></tr>
<tr><td></td><td></td><td>11-20</td><td>D3 - fraction of intake deposited in the NP region.</td></tr>
<tr><td></td><td></td><td>21-30</td><td>D4 - fraction of intake deposited in the TB region.</td></tr>
<tr><td></td><td></td><td>31-40</td><td>D5 - fraction of intake deposited in the P region.</td></tr>
<tr><td></td><td></td><td>41-50</td><td>NØRG - number of organs (excluding lungs and GIT), NØRG $\le$ 10.</td></tr>
<tr><td></td><td></td><td>51-60</td><td>IØPT - mode of intake.
0 - inhalation
1 - ingestion
2 - injection</td></tr>
<tr><td></td><td></td><td>61-70</td><td>NØRAL - number of ingestion cases to be run (used only with ICRP work). Leave blank if IØPT ≠ 1.</td></tr>
<tr><td></td><td>1+I,
I=1,...,NISØ</td><td></td><td>FØRMAT (A8, 2X, F10.0, 8X, A2, 2F10.0, 2I2, 2X, I2)</td></tr>
<tr><td></td><td></td><td>1-8</td><td>TAG(I) - Atomic symbol and atomic number of ITH member of the chain.</td></tr>
</table>

Table 5.1. (cont'd)

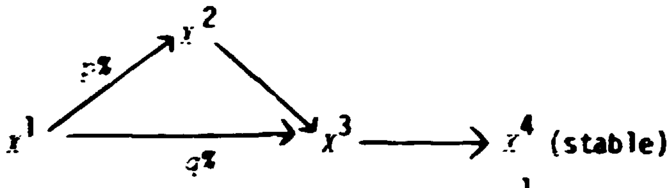
Subroutine	Card Number	Columns	Remarks, Variables, Etc.
		11-20	TR(I) - half life of ITH member of chain in days.
		29-30	ICLASS(I) - inhalation class of ITH member of chain. Should be one of the characters D, W, or Y right justified in the field. Leave blank if IØPT ≠ 0.
		31-40	LAMAB(I,2) - absorption coefficient in the small intestine (SI) $\lambda_{ab} = \frac{6 \cdot f_1}{1-f_1}$.
		41-50	P(I) - fraction of decay to radionuclide I.
		51-52	NBRNCH(I) - type of branching (1, 2, 3).
		53-54	IBRNCH(I) - position in the branching scheme (1, 2, 3).

Three different types of branching cases have been programmed, and the type of branching and position in the branching scheme are defined by the ordered pair (NBRNCH(1), IBRNCH(I)). The ordered pairs are described in the diagrams below.

$$\begin{tikzcd} & X^2 \\ X^1 \arrow[ur, "p\%"] \arrow[r] & X^3 \text{ (stable)} \end{tikzcd}$$

X^1	(0,0) - may leave NBRNCH(I) and IBRNCH(I) blank.
X^2	(1,1)

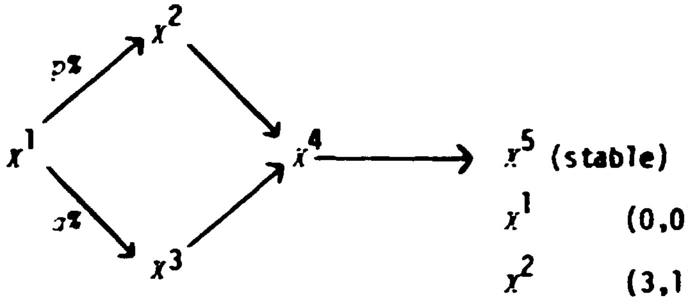
Table 5.1. (cont'd)



x^1 (0,0)

x^2 (2,1)

x^3 (2,2)



x^1 (0,0)

x^2 (3,1)

x^3 (3,2)

x^4 (3,3)

Subroutine	Card Number	Columns	Remarks, Variables, Etc.
		57-58	ILAB - switch to indicate if there is absorption in some section of the GIT tract other than the small intestine (SI) 0 - absorption only in small intestine. 1 - absorption in stomach(S), upper large intestine (UL), or lower large intestine (LL) in addition to small intestine absorption.

Table 5.1. (cont'd)

Subroutine	Card Number	Columns	Remarks, Variables, Etc.
	2·I+i I=1,...,NISØ		FØRMAT (3F10.0)
Card required only if ILAB = 1.			
		1-10	LAMAB(I,1) - adsorption coefficient for S.
		11-20	LAMAB(I,3) - adsorption coefficient for UL.
		21-30	LAMAB(I,4) - adsorption coefficient for LL.
Description of Retention			
This set of cards describes retention in those organs other than the lungs and GIT. These cards must be included for each radionuclide in the chain.			
ØRGAN	1		FØRMAT (10I2)
This card should be omitted for the first radionuclide in the chain.			
		1-2	IS(1) - fraction of daughter radionuclide which returns to blood for the 1st organ. (1-IS(1)) remains in the organ.
		3-4	IS(2) - same fraction for 2nd organ.
		⋮	⋮
Only NØRG numbers should be placed on the card.			

Table 5.1. (cont'd)

<table>
<tr><th>Subroutine</th><th>Card Number</th><th>Columns</th><th>Remarks, Variables, Etc.</th></tr>
<tr><td colspan="4">Include cards 2 and 3 for each organ.</td></tr>
<tr><td></td><td>2</td><td></td><td>FØRMAT (I2, A8, I2)</td></tr>
<tr><td></td><td></td><td>1-2</td><td>ISØRS(I) - index of organ I taken from the following list. ISØRS(I) is used for indexing purposes when calculating dose.</td></tr>
<tr><td colspan="2">1. bladder contents
2. stomach contents
3. SI + content
4. UL
5. LL
6. kidneys
7. liver
8. salivary gland
9. TB region
10. P region
11. lymph
12. muscle</td><td colspan="2">13. ovaries
14. pancreas
15. cortical bone (trabecular bone)
16. cancellous bone
17. red marrow
18. yellow marrow
19. skin
20. spleen
21. testes
22. thyroid
23. total body (used for other tissues, also)
25. adrenals</td></tr>
<tr><td colspan="4">(24. cloud should not be used as a source organ in this code.)</td></tr>
<tr><td></td><td></td><td>3-10</td><td>ØRGNM(I) - name of organ I (at most 8 characters).</td></tr>
<tr><td></td><td></td><td>11-12</td><td>ICØM(K,I) - number of compartments for organ I, radionuclide K (number of terms in the retention function), ICØM(K,I) $\le$ 10.</td></tr>
<tr><td></td><td>3</td><td></td><td>FØRMAT (8F10.0)</td></tr>
<tr><td colspan="4">This card describes the retention function and contains the coefficients of the exponential, AS(JJ,I),JJ=1,...,ICØM(K,I), and the exponential constants, TBS(JJ,I),JJ=1,...,ICØM(K,I), respectively. Obviously, for more than four compartments, i.e., terms in the retention function, another card is needed.)</td></tr>
</table>

Table 5.1. (cont'd)

<table>
<tr><th>Subroutine</th><th>Card Number</th><th>Columns</th><th>Remarks, Variables, Etc.</th></tr>
<tr><td colspan="4">In reference to cards 2 and 3, cards describing total body or "other tissues" should be placed last.
For each nuclide the organs and associated information must appear in the same order.
<u>Other Tissues Transformation</u>
If one of the source organs is "other tissues", the cumulated activity for these source organs must be transformed as described in Section II. To invoke the procedure, a card must be included for each print interval of the form:</td></tr>
<tr><td>ØTHER</td><td>1</td><td></td><td>FØRMAT (10I2)</td></tr>
<tr><td></td><td></td><td>1-2</td><td>NØTØP - nonzero integer</td></tr>
<tr><td></td><td></td><td>3-4</td><td>IØTHER(1) - ISØRS(1)</td></tr>
<tr><td></td><td></td><td>5-6</td><td>IØTHER(2) - ISØRS(2)</td></tr>
<tr><td></td><td></td><td>⋮</td><td>⋮</td></tr>
<tr><td colspan="4">Do not include 23 as a source number on this card.
If do not wish this option, include a blank card for each print interval.</td></tr>
</table>

which the card is read. "Card number" denotes the order within the group (Input Concerning the Differential Equations, General Information for Chain of Radioncelides, etc.) or quantity of cards for each group to be input. "Columns" refers to the actual columns on the card in which the data must be punched. "Remarks, Variables, Etc." gives an explanation as to the variable name, definition, and the form of the data to be punched on the cards. The FØRMAT descriptor indicates the content and length of the fields on the card. In the FØRMAT descriptor the letter "A" denotes alphabetic data; "I" refers to integer data (no decimal point) which must always be right-justified in the field; "F" or "E" indicates a real number with or without a signed exponent (decimal point should be punched). The variable names are those used in the FØRTRAN program. Several examples of input are included in Appendix B.

Output

Output may be grouped in four categories: (1) line printer output on unit 6, (2) line printer output on unit 10, (3) disk output on unit 16, and (4) punched output on unit 7. An explanation of each is given below.

1. Printout on unit 6 consists of the following:
   a. Summary printout of input - printed in MAIN.
   b. Error messages concerning incorrect input - MAIN.
   c. Coefficient matrix printed by rows - MATRIX.
   d. The nonzero elements of the coefficient matrix and the corresponding row and column index for each nonzero entry - MATRIX.

e. Output for each print interval consisting of possible error messages from DRIVES and value of T and the vector Y printed in YNA.

2. Printout on unit 10 consists of the following:

a. Summary of percent deposition - printed in MAIN.

b. Listing of the isotope, half life, class, absorption coefficient and branching information for each nuclide - MAIN.

c. Retention information for each organ and for each member of the chain - ØRGAN.

d. For each time period YNA prints the value of T, the cumulated activity (μCi-days), and activity (μCi) entering blood and GIT for each organ and for each member of the chain. ØTHER prints the result of the "other tissues" transformation if it is requested.

Printout is divided into two output units so that either unit 6 or unit 10 output may be suppressed. We commonly suppress unit 6 output unless the program abnormally terminates after which unit 6 output should be printed in an additional run for diagnostic purposes.

3. Disk output on unit 16 may be passed to another job step for use in dose calculations as in Eq. (1.1). The unit 16 data set is written unformatted on a 2314 disk pack, with a record format of variable blocked spanned, a logical record length of 150, and a blocksize of 1504. The temporary data set is given the name &&UCIDAY and

is created with the disposition of (NEW,PASS). See Appendix C for the summary of Job Control Language (JCL) needed to execute the code. Output on 16 is explained in Table 5.2. In Table 5.2, "Subroutine" refers to the subroutine in which the record is produced. "Record Number" denotes the order in which the records are created. "Remarks, Variables, Etc." gives an explanation of the variable name. "Internal Form of Data" gives the form of the internal data storage. "Alphameric", "integer", and "real" refer to internal computer representation of data.

4. Punched output to unit 7 is performed in routine YNA. For each nuclide in the chain one or more cards are punched which contains the cumulated activity (μCi-days) in the order stated in ISØRS (see Table 5.2). These cards are punched with the format (1P8E10.3).

Table 5.2. Form of Data Set Written on Unit 16

Subroutine	Record Number	Remarks, Variables, Etc.	Internal Form of Data
MATRIX	1	NISØ - number of radionuclides in the chain.	integer
		NØRG - number of organs (excluding lung and GIT).	integer
		IØPT - mode of intake. 0 - inhalation 1 - ingestion 2 - injection	integer
If IØPT=1, read record 2.			
MAIN	2	NØRAL - number of oral intakes (for ICRP tables only).	integer
MAIN	3	TAG(ID), ID=1,...,NISØ - atomic symbol and atomic number for each member of the chain.	alphameric
MAIN	4	ISØRS(ISØ), ISØ=1,...,total number of organs - indices for organs in the order NP, TB, P, L, S, SI, UL, LL, other organs in their order of input.	integer
For each time period read:			
YNA	1	TIME(ITM) - right end point of the time interval.	real
For each radionuclide read:			
YNA	1	VECTØR(ISØ), ISØ=1,...,total number of organs - cumulated activity given as microcurie-days for each of the organs in the order stated in ISØRS.	real
If IØPT=0 read:			
YNA	2	F(I), I=1,2,3. F(1) - microcuries entering blood from lungs. F(2) - microcuries entering GIT from lungs. F(3) - microcuries entering blood from GIT.	real

BIBLIOGRAPHY

1. W. S. Snyder, M. R. Ford, and S. B. Watson, "A Computer Program for Estimating Microcurie-Days of a Radionuclide in the Organs of the Human Body", *Health Physics Division Annual Progress Report for Period Ending June 30, 1975*, Oak Ridge National Laboratory Report ORNL-5046 (October 1975).

2. W. S. Snyder, M. R. Ford, G. G. Warner, and S. B. Watson, *A Tabulation of Dose Equivalent Per Microcurie-Day for Source and Target Organs of an Adult for Various Radionuclides*, Oak Ridge National Laboratory Report ORNL-5000 (November 1974).

3. W. S. Snyder, M. R. Ford, G. G. Warner, and S. B. Watson, *A Tabulation of Dose Equivalent Per Microcurie-Day for Source and Target Organs of an Adult for Various Radionuclides*, Oak Ridge National Laboratory Report ORNL-5000, Part 2 (November 1975).

4. ICRP Task Group on Lung Dynamics, "Deposition and Retention Models for Internal Dosimetry of the Human Respiratory Tract", *Health Phys.*, *12*, 173 (1966).

5. International Commission on Radiological Protection, *The Metabolism of Compounds of Plutonium and the Other Actinides*, ICRP Publication 19 (1972).

6. I. S. Eve, "A Review of the Physiology of the Gastrointestinal Tract in Relation to Radiation Doses from Radioactive Materials", *Health Phys.*, *12*, 131 (1966).

7. International Commission on Radiological Protection, *Report of the Task Group on Reference Man*, ICRP Publication 23 (1975).

8. A. C. Hindmarsh, *GEAR: Ordinary Differential Equation System Solver*, Lawrence Livermore Laboratory Report UCID-30001, Rev. 3 (December 1974).

9. C. W. Gear, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, New Jersey (1971).

10. A. C. Hindmarsh, *Linear Multistep Methods for Ordinary Differential Equations: Method Formulations, Stability, and the Methods of Nordsiech and Gear*, Lawrence Livermore Laboratory Report UCRL-51186, Rev. 1 (March 1972).

11. A. C. Hindmarsh, "The Control of Errors in the GEAR Package for Ordinary Differential Equations", *Construction of Mathematical Software*, F. N. Fritsch (ed.), Lawrence Livermore Laboratory Report UCID-30059, Rev. 1 (in progress).

12. A. Nordsiech, "On Numerical Integration of Ordinary Differential Equations", *Math. of Computations 16*, p. 22 (1962).

13. D. E. Arnurius, private communication (1974).

APPENDIX A

PROGRAM LISTING

(Microfiche)

```
C
C     MAIN DRIVER
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/PRINT/TPRL(10),DTPR(10),TPRINT,IPRT,NPR,NPRT,NABPRT(10)
      COMMON/SC1235/T,H,HMIN,HMAX,EPS1,MF1,KFLAG1,JSTART,N,MINV,MXG,
     1YMAX(400),ERROR(400),PSAVE(800),YPRINT(400)
C *** NEED DIMENSION YMAX(NXO),ERROR(NXO),PSAVE(2*NXO)
      COMMON/NSW/NSW
      DATA NPRTA/10/
      DATA MAXNO/400/
C *** MUST HAVE NO.LE.MAXNO
      DATA NELY/2400/
      DIMENSION Y(2400)
C *** DIMENSION Y(NELY)
C
      NXO=MAXNO
C ***
C ***
C *** SET MF AND T1 INTERNALLY
C *** HAVE DEFAULT VALUES FOR H1 AND EPS
C ***
C ***
C *** DESIRE TO INTEGRATE FROM T1 TO TLAST
C
C *** MF=21 INDICATES A STIFF SYSTEM.
      MF=21
C
C *** INTEGRATE Y'(T)=A*Y(T)+XO FROM T=T1 TO T=TLAST WHERE Y(T1)=0.
C *** Y IS THE TIME INTEGRAL OF ACTIVITY
C *** A IS A CONSTANT LOWER TRIANGULAR MATRIX AND XO IS A CONSTANT VECT.
C *** A,XO ARE OBTAINED FROM SUBS.MATRIX BY A CALL IN SUBR.STIFF
      T1=0.D0
C
 10   READ(5,1001,END=75) NO,TLAST,H1,EPS
 1001 FORMAT(I5,7E10.2)
      NSW=1
      IF (EPS.EQ.0.D0) EPS=1.D-6
      IF(H1.EQ.0.D0) H1=1.D-7
C
C *** SET INITIAL VALUES TO 0.
C
 15   DO 20 I = 1, NO
 20   Y(I)=0.D0
C
C
      READ (5,1002) NPRT
 1002 FORMAT(I5)
      NREAD=MIN0(NPRT,NPRTA)
 25   READ (5,1003) (TPRL(I),DTPR(I),NABPRT(I),I=1,NREAD)
 1003 FORMAT(2E10.2,I5)
      PRINT 1004,NPRT,NREAD,(TPRL(I),DTPR(I),NABPRT(I),I=1,NREAD)
 1004 FORMAT( '1NO. PRINT INTERVALS DESIRED =' I2,3X,'NO. READ =' I2/
     1' POINTS SELECTED IN PRINT INTERVAL = T+Q*DT WHERE Q=0,1,...,N'/
     29X,'T'18X'DT'11X'N'/(1PE19.10,E19.10,I6))
C
```

```
      IF(NREAD.EQ.NPRT) GO TO 30
      PRINT 1005
 1005 FORMAT(///'0NEED MORE STORAGE SPACE FOR INTERVALS IN MAIN.'//)
      GO TO 50
C
 30   CONTINUE
      TMAX=TPRL(NPRT) + NARPRT(NPRT)*DTPR(NPRT)
C
C *** EXPECT TMAX TO BE THE LARGEST T VALUE AT WHICH A 'CALL YOUT' IS
C *** DESIRED.
      IF(DABS(TLAST-TMAX).LE.1.D-14*DABS(TLAST)) TLAST=TMAX
      IF(TLAST.LE.TMAX) GO TO 35
C
      PRINT 1006,TLAST,TMAX
 1006 FORMAT(//'0VALUE OF TLAST HAS BEEN CHANGED FROM'1PE21.13,1X,'TO'
     1E21.13//)
C
      TLAST=TMAX
 35   NT=NPRT-1
      T0 = T1
      NC=0
      IF(NT.LT.1) GO TO 45
      DO 40 I=1,NT
      IF(TPRL(I)+NARPRT(I)*DTPR(I).LT.TPRL(I+1)) GO TO 40
      NC=NC+1
 40   CONTINUE
C
      IF(NC.GT.0) PRINT 1007,NC,NT
 1007 FORMAT('1'I2,1X'OF THE 'I2,1X'CONSECUTIVE INTERVAL PAIRS OVERLAP.'
     1/' PROG. WILL CALL EXIT. SEE MAIN DRIVER.'/)
C
 45   CONTINUE
      IF(T0.LE.TPRL(1)) GO TO 55
      PRINT 1008
 1008 FORMAT('1TPRL(1).LT.T0. CALL EXIT. SEE MAIN DRIVER.')
 50   CALL EXIT
C
 55   IF(NC.GT.0) GO TO 50
      TPRINT=TPRL(1)
      IPRT=1
      NPR=0
      NCASE=0
      N0 = N1
      WRITE (6,1009) N0, NP, T0, TLAST, H0, EPS
 1009 FORMAT('0N0='I4/' NP='I2/' T0='1PE22.15/' TLAST='E22.15/
     1' H0='E22.15/' EPS='E10.4/)
C
C *** FOR STIFF PROCEDURE NEED 6*N0 ELEMENTS IN 'VECTOR' Y.  FOR ADAM'S
C *** (NONSTIFF) NEED 13*N0 ELEMENTS.
C *** NP=10*METH+MITER, METH=1 FOR ADAM'S, =2 FOR STIFF PROCEDURE.
C
      IF(NP.GT.19) GO TO 60
C
C *** ADAM'S (NONSTIFF)
C
      IF(13*N0.LE.NELT) GO TO 65
      PRINT 1010,NELY
 1010 FORMAT('1NEED 13*NO ELEMENTS IN VECTOR Y IN MAIN. HAVE ONLY NELY='
     1I8)
      GO TO 50
C
 60   IF(6*NO.LE.NELY) GO TO 65
C
C *** STIFF PROCEDURE.
C
      PRINT 1011,NELY
 1011 FORMAT('1NEED 6*NO ELEMENTS IN VECTOR Y IN MAIN. HAVE ONLY NELY='
     1I8)
      GO TO 50
C
 65   IF(NO.LE.MAXNO) GO TO 70
      PRINT 1012,NO,MAXNO
 1012 FORMAT('1NO EXCEEDS MAX.NO'/' NO='I8/' MAXNO='I8/' INCREASE DIM. I
     1N COMMON SC1235.')
      GO TO 50
C
 70   CONTINUE
C
C
      CALL DRIVES(NO,TO,TLAST,Y,HO,EPS,MF,KFLAG)
      WRITE (6,1013) KFLAG
 1013 FORMAT(9H0KFLAG = ,I6)
C
      GO TO 10
 75   STOP
C ---------- LAST CARD OF MAIN DRIVER ----------
      END
```

```
      SUBROUTINE DRIVES(NO,TO,TLAST,Y,HO,EPS,MF,KFLAG)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/SC1235/T,H,HMIN,HMAX,EPS1,MF1,KFLAG1,JSTART,N,NINV,NXO,
     1YMAX(400),ERROR(400),FSAVE(800),YPRINT(400)
C *** NEED DIMENSION YMAX(NXO),ERROR(NXO),FSAVE(2*NXO)
      COMMON/PRINT/TPRL(10),DTPR(10),TPRINT,IPRT,NPR,NPRT,NAMPRT(10)
      DIMENSION Y(NO,1)
C***********************************************************************
C* DRIVES IS DRIVER SUBROUTINE FOR STIFF.  SEE WRITEUP FOR FULL DETAILS.
C*
C* THE INPUT PARAMETERS ARE.
C*    NO     =  THE NUMBER OF DIFFERENTIAL EQUATIONS, INITIALLY.
C*    TO     =  THE INITIAL VALUE OF T, THE INDEPENDENT VARIABLE.
C*    TLAST  =  THE FINAL VALUE OF T.
C*    Y      =  AN NO BY LMAX ARRAY FOR THE COMPUTED SOLUTION, THE SAME
```

```
C*             AS IN STIFF.  THE FIRST COLUMN SHOULD CONTAIN THE
C*             INITIAL VALUES OF THE DEPENDENT VARIABLE.
C*             CURRENTLY, LMAX IS 13 FOR THE ADAMS METHODS AND
C*             6 FOR THE GEAR METHODS.  TO CHANGE THIS, SEE
C*             SUBROUTINE TESOC.
C*   H0    =   THE INITIAL VALUE OF H, THE STEP SIZE IN T.
C*   EPS   =   THE RELATIVE ERROR BOUND, THE SAME AS IN STIFF.
C*   MF    =   THE METHOD FLAG, THE SAME AS IN STIFF.
C*             ALLOWED VALUES ARE 10, 11, 12, 13, 20, 21, 22, 23.
C*
C* THE OUTPUT PARAMETERS ARE..
C*   T0    =   VALUE OF T TO WHICH INTEGRATION HAS BEEN COMPLETED.
C*   Y     =   N0 BY LMAX ARRAY OF Y AND ITS SCALED DERIVATIVES.
C*             Y(I) FOR 1 .LE. I .LE. N CONTAINS THE COMPUTED SOLUTION
C*             AT TIME T.
C*   H0    =   THE VALUE OF THE STEP SIZE H USED LAST, BOTH IN THE
C*             INTEGRATION AND IN THE SCALING IN THE Y ARRAY.
C*   KFLAG =   A COMPLETION FLAG WITH THE FOLLOWING VALUES AND MEANINGS..
C*          0   THE PROBLEM WAS COMPLETED SUCCESSFULLY.
C*         -1   THE INTEGRATION WAS HALTED AFTER FAILING TO PASS THE
C*              ERROR TEST EVEN AFTER REDUCING H BY A FACTOR OF
C*              1.0D10 FROM ITS INITIAL VALUE.
C*         -2   AFTER SOME INITIAL SUCCESS, THE INTEGRATION WAS
C*              HALTED WHEN A REDUCTION IN H BY A FACTOR OF MORE
C*              THAN 1.0D4 WAS INDICATED IN ORDER TO PASS THE ERROR TEST.
C*         -3   THE INTEGRATION WAS HALTED AFTER FAILING TO ACHIEVE
C*              CORRECTOR CONVERGENCE EVEN AFTER REDUCING H BY A
C*              FACTOR OF 1.0D10 FROM ITS INITIAL VALUE.
C*         -4   IMMEDIATE HALT BECAUSE OF ILLEGAL VALUES OF INPUT
C*              PARAMETERS.  SEE PRINTED MESSAGE.
C*         -5   H IS SUCH THAT T+H .EQ. T
C*
C* DRIVES CALLS SUBROUTINE YOUT FOR NORMAL OUTPUT AT STATEMENT 20,
C* AND SUBROUTINE YOUTD FOR OUTPUT IN CASE OF FAILURE BELOW 150 AND 200.
C* THE NAMES AND CALL SEQUENCES OF THESE CAN BE ALTERED AS DESIRED.
C*
C* IN THE FOLLOWING, SET..
C*   NMX  =  THE MAXIMUM VALUE OF N FOR WHICH DRIVES IS TO BE USED.
C*   LOUT =  THE LOGICAL UNIT NUMBER FOR THE OUTPUT OF MESSAGES
C*           DURING THE INTEGRATION.
C* THE DIMENSION OF PW ABOVE CAN BE REDUCED IF MITER IS NOT 1 OR 2
C* (SEE DESCRIPTION OF PW IN STIFF).
C*
C* THE ORIGINAL VERSION OF THIS PROGRAM WAS WRITTEN AT LLL BY A. C.
C* HINDMARSH FOR CDC COMPUTERS.  THE CDC VERSION WAS MODIFIED FOR
C* USE ON IBM COMPUTERS AT ARGONNE IN JUNE, 1973.
C*
C* LATEST REVISION  SEPTEMBER, 1973
C*
C*****************************************************************
C*
      DATA LOUT/6/
C*
C*****************************************************************
C*
      IF (EPS .LE. 0.0D0) GO TO 400
      IF (NO .LE. 0) GO TO 430
      IF ((TO-TLAST)*HO .GE. 0.0D0) GO TO 420
      N=NO
      T = TO
      H = HO
      NINV=0
      HMIN = DABS(HO)
      HMAX = DABS(TO-TLAST) * .1D0
      HMIN = DMIN1(HMIN,.1D0*HMAX)
      EPS1 = EPS
      MF1 = MF
      JSTART=0
      KHFLAG = 0
C *** MAY WISH TO PRINT INITIAL VALUES THRU YOUT.
      IF(T.EQ.TPRINT) CALL YOUT(NO,TLAST,Y,1)
   10 CALL STIFF(Y,NO)
      NQ = JSTART
C
      KGO = 1 - KFLAG1
      GO TO ( 20,100,200,300,20,350) , KGO
C  KFLAG1 =    0, -1, -2, -3, -5
C
 20   CONTINUE
      IF(T.GE.TPRINT) CALL YOUT(NO,TLAST,Y,NQ)
      IF ((T-TLAST)*H .LT. 0.0D0) GO TO 10
C*********************************************************************
C* THE PROBLEM IS FINISHED.  HERE CALL YOUT AND/OR OTHER ROUTINES
C* TO OUTPUT DESIRED FINAL RESULTS.
C*********************************************************************
      GO TO 500
C
 100  WRITE (LOUT,105) T
 105  FORMAT (//30H KFLAG = -1 FROM STIFF AT T = ,D25.16/
     1   39H  ERROR TEST FAILED WITH ABS(H) = HMIN/)
 110  IF (KHFLAG .EQ. 10) GO TO 150
      KHFLAG = KHFLAG + 1
      HMIN = HMIN * .1D0
      H = H * .1D0
      WRITE (LOUT,115) H
 115  FORMAT(24H  H HAS BEEN REDUCED TO , D25.16,
     1   26H  AND STEP WILL BE RETRIED//)
      JSTART = -1
      GO TO 10
C
 150  WRITE (LOUT,155)
 155  FORMAT(//44H PROBLEM APPEARS UNSOLVABLE WITH GIVEN INPUT//)
C*********************************************************************
C* HMIN HAS BEEN CUT BY 10 ORDERS OF MAGNITUDE WITH NO SUCCESS.
C* AT THIS POINT, OUTPUT INFORMATION NEEDED FOR DEBUGGING.
C*********************************************************************
      CALL YOUTD(Y,NQ)
      GO TO 500
C
 200  WRITE (LOUT,205) T,H
 205  FORMAT(//30H KFLAG = -2 FROM STIFF AT T = ,D25.16,5H  H =,D25.16/
     1   52H  THE REQUESTED ERROR IS SMALLER THAN CAN BE HANDLED//)
```

```
C* AT THIS POINT, OUTPUT INFORMATION NEEDED FOR DEBUGGING.               D257 159
      CALL YOUTD(Y,NQ)                                                   D257 160
      GO TO 500                                                          D257 161
C                                                                        D257 162
 300  WRITE (LOUT,305) T                                                 D257 163
 305  FORMAT(//30H KFLAG = -3 FROM STIFF AT T = ,D25.16/                 D257 164
     1   45H  CORRECTOR CONVERGENCE COULD NOT BE ACHIEVED/)              D257 165
      GO TO 110                                                          D257 166
 350  WRITE (LOUT,355) T,H                                               D257 167
 355  FORMAT(//30H KFLAG = -5 FROM STIFF AT T = ,D25.16,                 D257 168
     1   5H H = ,D25.16/                                                 D257 169
     2   45H  H IS SUCH THAT T+H .EQ. T ON MACHINE.          /)          D257 170
      CALL YOUTD(Y,NQ)                                                   D257 171
      GO TO 500                                                          D257 172
C                                                                        D257 173
 400  WRITE (LOUT,405)                                                   D257 174
 405  FORMAT(//28H ILLEGAL INPUT.. EPS .LE. 0.//)                        D257 175
      KFLAG = -4                                                         D257 176
      RETURN                                                             D257 177
C                                                                        D257 178
 410  WRITE (LOUT,415)                                                   D257 179
 415  FORMAT(//25H ILLEGAL INPUT.. N .LE. 0//)                           D257 180
      KFLAG = -4                                                         D257 181
      RETURN                                                             D257 182
C                                                                        D257 183
 420  WRITE (LOUT,425)                                                   D257 184
 425  FORMAT(//38H ILLEGAL INPUT.. (T0-TLAST)*H0 .GE. 0.//)              D257 185
      KFLAG = -4                                                         D257 186
      RETURN                                                             D257 187
C                                                                        D257 188
 500  KFLAG = KFLAG1                                                     D257 189
      T0 = T                                                             D257 190
      H0 = H                                                             D257 191
      RETURN                                                             D257 192
      END                                                                D257 194
```

```
      SUBROUTINE STIFF(Y,N0)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/NSW/NSW
      COMMON/SC1235/T,H,HMIN,HMAX,EPS,MF,KFLAG,JSTART,N,NINV,NIC,
     1YMAX(400),ERROR(400),PSAVE(800),YPRINT(400)
C *** NEED DIMENSION YMAX(NX0),ERROR(NX0),PSAVE(2*NX0)
      DIMENSION Y(N0,13),EL(13),TQ(4)                                    D257 206
C***********************************************************************D257 207
C* STIFF PERFORMS ONE STEP OF THE INTEGRATION OF AN INITIAL VALUE        D257 208
C* PROBLEM FOR A SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS.              D257 209
C* COMMUNICATION WITH STIFF IS DONE WITH THE FOLLOWING VARIABLES..       D257 210
```

```
C*
C*   Y       AN N0 BY LMAX ARRAY CONTAINING THE DEPENDENT VARIABLES
C*             AND THEIR SCALED DERIVATIVES.  LMAX IS CURRENTLY 13 FOR
C*             THE ADAMS METHODS AND 6 FOR THE GEAR METHODS.  LMAX - 1
C*             IS MAXDER, THE MAXIMUM ORDER USED. SEE SUBROUTINE TESOC.
C*             Y(I,J+1) CONTAINS THE J-TH DERIVATIVE OF Y(I), SCALED BY
C*             H**J/FACTORIAL(J).  ONLY Y(I), 1 .LE. I .LE. N, NEED BE
C*             SET BY THE CALLING PROGRAM ON THE FIRST ENTRY.
C*               IF IT IS DESIRED TO INTERPOLATE TO NON-MESH POINTS,
C*             THE Y ARRAY CAN BE USED.  IF THE CURRENT STEP SIZE
C*             IS H AND THE VALUE AT T + E IS NEEDED, FORM
C*             S = E/H, AND THEN COMPUTE
C*                                NQ
C*               Y(I)(T+E) =  SUM  Y(I,J+1)*S**J .
C*                               J=0
C*             THE Y ARRAY SHOULD NOT BE ALTERED BY THE CALLING PROGRAM.
C*             WHEN REFERENCING Y AS A 2-DIMENSIONAL ARRAY, USE A
C*             COLUMN LENGTH OF N0, AS THIS IS THE VALUE USED IN STIFF.
C*   N       THE NUMBER OF FIRST ORDER DIFFERENTIAL EQUATIONS.  N
C*             MAY BE DECREASED ON LATER CALLS IF THE NUMBER OF
C*             ACTIVE EQUATIONS REDUCES, BUT IT MUST NOT BE
C*             INCREASED WITHOUT CALLING WITH JSTART = 0.
C*   N0      A CONSTANT INTEGER .GE. N, USED FOR DIMENSIONING PURPOSES.
C*             N0 MUST NOT BE CHANGED WITHOUT SETTING JSTART = 0.
C*   T       THE INDEPENDENT VARIABLE. T IS UPDATED ON EACH STEP TAKEN.
C*   H       THE STEP SIZE TO BE ATTEMPTED ON THE NEXT STEP.
C*             H MAY BE ADJUSTED UP OR DOWN BY THE ROUTINE
C*             IN ORDER TO ACHEIVE AN ECONOMICAL INTEGRATION.
C*             HOWEVER, IF THE H PROVIDED BY THE USER DOES
C*             NOT CAUSE A LARGER ERROR THAN REQUESTED, IT
C*             WILL BE USED.  TO SAVE COMPUTER TIME, THE USER IS
C*             ADVISED TO USE A FAIRLY SMALL STEP FOR THE FIRST
C*             CALL.  IT WILL BE AUTOMATICALLY INCREASED LATER.
C*             H CAN BL EITHER POSITIVE OR NEGATIVE, BUT ITS SIGN
C*             MUST REMAIN CONSTANT THROUGHOUT THE PROBLEM.
C*   HMIN    THE MINIMUM ABSOLUTE VALUE OF THE STEP SIZE THAT WILL BE
C*             USED FOR THE PROBLEM.  ON STARTING THIS MUST BE MUCH
C*             SMALLER THAN THE AVERAGE ABS(H) EXPECTED, SINCE
C*             A FIRST ORDER METHOD IS USED INITIALLY.
C*   HMAX    THE MAXIMUM ABSOLUTE VALUE OF THE STEP SIZE THAT WILL
C*             BE USED FOR THE PROBLEM.
C*   EPS     THE RELATIVE ERROR TEST CONSTANT.  SINGLE STEP ERROR
C*             ESTIMATES DIVIDED BY  YMAX(I)  MUST BE LESS THAN THIS
C*             IN THE EUCLIDEAN NORM. THE STEP AND/OR ORDER IS
C*             ADJUSTED TO ACHEIVE THIS.
C*   MF      THE METHOD FLAG.  MF IS A POSITIVE INTEGER WITH
C*             TWO DECIMAL DIGITS--METH AND MITER (MF=10*METH+MITER).
C*             METH IS THE BASIC METHOD INDICATOR..
C*               METH = 1  MEANS THE ADAMS METHODS.
C*               METH = 2  MEANS THE STIFF SYSTEM METHODS OF GEAR.
C*             MITER IS THE ITERATION METHOD INDICATOR..
C*               MITER = 0 MEANS FUNCTIONAL ITERATION ( NO PARTIAL
C*                         DERIVATIVES NEEDED).
C*               MITER = 1 MEANS CHORD METHOD WITH ANALYTIC JACOBIAN.
C*                         FOR THIS USER SUPPLIES SUBROUTINE
C*                         PEDERV(N,T,Y,PD,N0).  PD IS AN N BY N ARRAY,
C*                          STORED AS AN N0 BY N0 ARRAY.  PD(I,J) IS       D257 267
C*                          TO BE SET TO THE PARTIAL DERIVATIVE OF         D257 268
C*                          YDOT(I) WITH RESPECT TO Y(J).                  D257 269
C*                MITER = 2 MEANS CHORD METHOD WITH JACOBIAN               D257 270
C*                          CALCULATED IN STIFF BY FINITE DIFFERENCES.     D257 271
C*                          PEDERV IS NOT CALLED AND A DUMMY CAN BE USED.D257 272
C*                MITER = 3 MEANS CHORD METHOD WITH JACOBIAN REPLACED      D257 273
C*                          BY A DIAGONAL APPROXIMATION BASED ON A         D257 274
C*                          DIRECTIONAL DERIVATIVE.                        D257 275
C*   YMAX    AN ARRAY OF N LOCATIONS WHICH CONTAINS THE MAXIMUM            D257 276
C*             ABSOLUTE VALUE OF EACH Y SEEN SO FAR.                       D257 277
C*   ERROR   AN ARRAY OF N ELEMENTS PROPORTIONAL TO THE ESTIMATED          D257 278
C*             ONE STEP ERROR IN EACH COMPONENT.                           D257 279
C*   KFLAG   A COMPLETION CODE WITH THE FOLLOWING MEANINGS..               D257 280
C*                        0  THE STEP WAS SUCCESFUL.                       D257 281
C*                       -1  THE REQUESTED ERROR COULD NOT BE ACHIEVED     D257 282
C*                             WITH ABS(H) = HMIN.                         D257 283
C*                       -2  THE REQUESTED ERROR IS SMALLER THAN CAN       D257 284
C*                             BE HANDLED FOR THIS PROBLEM.                D257 285
C*                       -3  CORRECTOR CONVERGENCE COULD NOT BE            D257 286
C*                             ACHIEVED FOR ABS(H) .GT. HMIN.              D257 287
C*                       -5  H WAS SUCH THAT T+H .EQ. T ON MACHINE.        D257 288
C*             ON A RETURN WITH KFLAG NEGATIVE, THE VALUES OF T AND        D257 289
C*             THE Y ARRAY ARE AS OF THE BEGINNING OF THE LAST             D257 290
C*             STEP, AND H IS THE LAST STEP SIZE ATTEMPTED.                D257 291
C*   JSTART  AN INTEGER USED ON INPUT AND OUTPUT.                          D257 292
C*             ON INPUT, IT HAS THE FOLLOWING VALUES AND MEANINGS..        D257 293
C*                        0  PERFORM THE FIRST STEP.  THIS VALUE ENABLES   D257 294
C*                             THE SUBROUTINE TO INITIALIZE ITSELF.        D257 295
C*                    .GT.0  TAKE A NEW STEP CONTINUING FROM THE LAST.     D257 296
C*                             ASSUMES THE LAST STEP WAS SUCCESSFUL AND    D257 297
C*                             USER HAS NOT CHANGED ANY PARAMETERS.        D257 298
C*                    .LT.0  REPEAT THE LAST STEP WITH A NEW VALUE OF      D257 299
C*                             H AND/OR EPS AND/OR MF.  THIS MAY BE        D257 300
C*                             EITHER IN REDOING A STEP THAT FAILED, OR    D257 301
C*                             IN CONTINUING FROM A SUCCESSFUL STEP.       D257 302
C*             ON EXIT, JSTART IS SET TO NQ, THE CURRENT ORDER OF THE      D257 303
C*             METHOD.  THIS IS ALSO THE ORDER OF THE MAXIMUM DERIVATIVED257 304
C*             AVAILABLE IN THE Y ARRAY.  AFTER A SUCCESSFUL STEP,         D257 305
C*             JSTART NEED NOT BE RESET FOR THE NEXT CALL.                 D257 306
C*   PW      A BLOCK OF LOCATIONS USED FOR PARTIAL DERIVATIVES IF          D257 307
C*             MITER IS NOT 0.  IF MITER IS 1 OR 2, ITS LENGTH MUST        D257 308
C*             BE AT LEAST N0*(N0+1), FOR THE JACOBIAN AND PIVOT           D257 309
C*             INFORMATION.  IF MITER = 3, ITS LENGTH MUST ONLY BE         D257 310
C*             AT LEAST N.                                                 D257 311
C*   PSAVE   A BLOCK OF AT LEAST 2*N0 LOCATIONS FOR TEMPORARY STORAGE.    D257 312
C*                                                                         D257 313
C* THE PARAMETERS WHICH MUST BE INPUT BY THE USER ARE..                    D257 314
C*   N, N0, T, Y, H, HMIN, HMAX, EPS, MF, JSTART.                          D257 315
C*                                                                         D257 316
C* ADDITIONAL SUBROUTINES REQUIRED ARE..                                   D257 317
C*   TESOC(METH,NQ,EL,TQ,MAXDER)     SETS METHOD COEFFICIENTS.             D257 318
C*        IT IS PROVIDED IN THE PACKAGE.                                   D257 319
C*   DIFFUN(N,T,Y,YDOT)    COMPUTES DY/DT GIVEN T AND Y.                   D257 320
C*   PEDERV(N,T,Y,PD,N0)    COMPUTES PARTIAL DERIVATIVES.  IT IS USED      D257 321
C*        ONLY IF MITER = 1.  (SEE MF DESCRIPTION.)                        D257 322
```

```
C*   PROCED(NO,N,A,IPS,S,IER,NINV) AND EVLOS(NO,N,A,B,X,IPS)    ARE USED D257 323
C*        FOR THE SOLUTION OF SYSTEMS OF LINEAR ALGEBRAIC EQUATIONS.     D257 324
C*        THEY ARE PROVIDED IN THE PACKAGE, AND USED IF NITER = 1 OR 2.  D257 325
C*                                                                       D257 326
C* THE CALLING PROGRAM MUST CONTAIN THE COMMON DECLARATIONS GIVEN        D257 327
C* ABOVE, IN WHICH THE LENGTHS ARE AS GIVEN BY THE DESCRIPTIONS          D257 328
C* OF THE VARIABLES ABOVE.  THE OPTIONAL ROUTINE  DRIVES  MAY BE         D257 329
C* USED FOR THIS PURPOSE.  THE CONSTANT  ANOISE  IN THE DATA STATEMENT   D257 330
C* BELOW SHOULD BE SET TO THE NOISE LEVEL OF THE MACHINE.                D257 331
C*                                                                       D257 332
C* THE ORIGINAL VERSION OF THIS PROGRAM WAS WRITTEN AT LLL BY A. C.      D257 333
C* HINDMARSH FOR CDC COMPUTERS.  THE CDC VERSION WAS MODIFIED FOR        D257 334
C* USE ON IBM COMPUTERS AT ARGONNE IN JUNE, 1973.                        D257 335
C*                                                                       D257 336
C* LATEST REVISION  SEPTEMBER, 1973                                      D257 337
C*                                                                       D257 338
C***********************************************************************D257 339
      DATA ANOISE / 5.D-16/                                              D257 340
C*                                                                       D257 341
C***********************************************************************D257 342
      DIMENSION A(12001),IDIAG(400),XO(400),PW(6000)
      DIMENSION NBLK(015)
      REAL*8 LR(15)
      EQUIVALENCE (A(1),IND(1))
      DIMENSION IND(1)
      DATA NLR/15/
      DATA NWDAA/12001/
      DATA NWDPW/6000/
C *** NWDPW = NO. WORDS IN VECTOR PW. REQUIRES NWDPW.GE.NELMA
C *** NWDAA = NO. WORDS IN ARRAY A. SHOULD BE ODD
C *** NXO= NO. ELE. IN XO VECTOR = NO. ELE. IN IDIAG VECTOR.
C *** NLR=NO. WORDS IN ARRAY LR.
      IF (NSW.EQ.0) GO TO 1
C ***
      NL=1
C ***
      NSW=0
      NWDAV=NWDAA/2
      INDXA=NWDAV+1
C *** WANT A(1) AND A(INDXA) TO BE LOCATED IN DIFFERENT MEMORY BOXES.
C *** ON THE IBM 360/91.
      IF(16*(NWDAV/16).EQ.NWDAV) INDXA=INDXA+1
C *** XO AND IDIAG WILL EACH HAVE N ELEMENTS IN THEM ON RETURN FROM
C *** MATRIX
      IF (N.LE.NXO) GO TO 3
      PRINT 4,N,NXO
    4 FORMAT('1FORMAT 4  SUBR.STIFF'//' N=' I4/' NXO=MAXNO=' I4//
     1' VECTORS WITH DIM (NXO) NEED TO BE INCREASED TO N AT LEAST.'//
     2' SET MAXNO IN DATA STATEMENT IN ''MAIN'' TO NEW DIM. VALUE OF NXO
     3')
   10 CALL EXIT
    3 CONTINUE
C *** MATRIX RETURNS THE DIAGONAL OF A SPLIT INTO A PART IN THE DIAG.
C *** ELEMENTS IN VECTOR A AND INTO THE LR ARRAY THAT EXPANDS TO
C *** DIAG(LR(1)*I(1),....,LR(NDSUBN)*I(NDSUBN)) WHERE I(J) IS A VECTOR
C *** OF 1'S OF DIMENSION NBLK(J).
```

```
C *** THIS SPLIT IS FOR ANOTHER PROGRAM TO CALC. APPROX.EIGENVECTORS.
C *** THE NO. OF NONZERO OFF-DIAGONAL ELEMENTS IN MATRIX A + N
C *** (N= NUMBER OF DIAG. ENTRIES IN MATRIX A) MUST BE .LE.NWDAV WHERE
C *** NWDAV=MAX. NO. WORDS AVAILABLE FOR THE VECTOR A.
C *** EXPECT SUBR.MATRIX TO CHECK THAT AVAILABLE STORAGE IS SUFFICIENT
C *** FOR CURRENT PROBLEM
      CALL MATRIX(A,NWDAV,A(INDXA),IDIAG,XO,NXO,N,LR,NBLK,NLR,NDSUBM)
C *** DIMENSION IDIAG(NXO),XC(NXO),LR(NLR),NBLK(NLR)
C *** NDSUBM= NO. OF DIAG. SUBMATRICES IN ARRAY A.
      IF(NDSUBM.LE.NLR) GO TO 5
      PRINT 6,NDSUBM,NLR
    6 FORMAT('1SEE SUBR. STIFF'/' NEED MORE SPACE IN LR,NBLK. DEFINE NLR
     1.'/' NEED' I3,1X'ELEMENTS'/' HAVE'I3,1X'ELEMENTS')
      GO TO 10
    5 CONTINUE
      NELEA=IDIAG(N)
      IF(NELEA.LE.NWDPW) GO TO 8
      PRINT 9,NELEA,NWDPW
    9 FORMAT('1SEE SUBR.STIFF'//' NEED NELEA=' I4,1X'ENTRIES IN VECTOR PW
     1'/' HAVE NWDPW=' I4,1X'AVAILABLE.')
      GO TO 10
    8 CONTINUE
      MB=0
      NB=INDXA+INDXA
      DO 11 I=1,N
      MB=MAX0(MB,IND(NB))
   11 NB=NB+4
      MB=MB/(-14)
      IF(MB.LE.052) GO TO 13
      PRINT 12,MB
   12 FORMAT('1ASSEMBLER LANGUAGE ROUTINES YPAY AND AXEB CAN HANDLE AT M
     1OST 052 NONZERO ENTRIES IN A ROW OF THE MATRIX A.'//
     2' MAX.NO. NONZERO ENTRIES IN CURRENT A = MB =' I3)
      GO TO 10
   13 CONTINUE
      ICUT=0
      INBLK=0
      DO 20 I=1,N
      IF(I.LE.ICUT) GO TO 21
      INBLK=INBLK+1
      ICUT=ICUT+NBLK(INBLK)
      RLP=LR(INBLK)
   21 CONTINUE
      ID=IDIAG(I)
   20 A(ID)=A(ID)+RLP
C *** BASIC DIFFERENTIAL EQ. IS Y'=A*Y+XO,  Y(0)=0
C *** INITIAL CONDITIONS SET IN MAIN PROGRAM.
      NJAC=0.
    1 CONTINUE
C*                                                                      D257 343
      KFLAG = 0                                                         D257 344
      TOLD = T                                                          D257 345
C* TEST FOR SMALL  H  ADDED AT ARGONNE IN AUGUST, 1973.                 D257 346
      IF (T+H .EQ. T) GO TO 685                                         D257 347
      IF (JSTART .GT. 0) GO TO 200                                      D257 348
      IF (JSTART .NE. 0) GO TO 120                                      D257 349
```

```
C***********************************************************************
C* ON THE FIRST CALL, THE ORDER IS SET TO 1 AND THE INITIAL
C* DERIVATIVES ARE CALCULATED.  YMAX IS INITIALIZED USING THE INITIAL
C* Y AND YDOT.  IF BOTH ARE INITIALLY ZERO IN ANY COMPONENT, THE DEFAULT
C* VALUE IS 1.  RMAX IS THE MAXIMUM RATIO BY WHICH H CAN BE INCREASED
C* IN A SINGLE STEP.  IT IS INITIALLY 1.E4 TO COMPENSATE FOR THE SMALL
C* INITIAL H, BUT THEN IS NORMALLY EQUAL TO 10.  IF A FAILURE
C* OCCURS (IN CORRECTOR CONVERGENCE OR ERROR TEST), RMAX IS SET AT 2
C* FOR THE NEXT INCREASE.  EPSJ IS USED AS THE RELATIVE INCREMENT
C* TO Y WHEN GETTING PARTIALS BY FINITE DIFFERENCING.
C***********************************************************************
      NSQ = N0 * N0
      NSQ1 = NSQ + 1
      N1 = N0 + 1
      CALL YFAY(PSAVE,A,Y,X0,A(INDXA),N)
      DO 110 I=1,N
      Y(I,2)=PSAVE(I)*H
      AYI=DABS(Y(I,1))
      IF(AYI.EQ.0.D0) AYI=DABS(Y(I,2))
      IF(AYI.EQ.0.D0) AYI=1.D0
      YMAX(I)=AYI
  110 CONTINUE
      NQ = 1
      L = 2
      RMAX = 1.0D4
      EPSJ = DSQRT(ANOISE)
      CRATE = 1.0D0
      OLDL0 = 1.0D0
      FC = 0.0D0
      MFOLD = 0
      METH = 0
      HOLD = H
C* THIS INITIALIZES MITER  SO  STIFF  WILL FUNCTION CORRECTLY WHEN
C* GIVEN THE SAME  MITER  AS WAS GIVEN IN A PREVIOUS PROBLEM.
            MITER=0
C* ADDED AT ARGONNE IN JULY, 1973.
C***********************************************************************
C* IF THE CALLER HAS CHANGED METH, OR IF JSTART = 0, TESOC IS CALLED
C* TO SET THE COEFFICIENTS OF THE METHOD.  IF THE CALLER HAS CHANGED
C* EPS OR METH, THE CONSTANTS E, EDN, EUP, AND BND MUST BE RESET.
C* E IS A COMPARISON FOR ERRORS OF THE CURRENT ORDER NQ.  EUP IS
C* TO TEST FOR INCREASING THE ORDER, EDN FOR DECREASING THE ORDER.
C* BND IS USED TO TEST FOR CONVERGENCE OF THE CORRECTOR ITERATES.
C* IF THE CALLER HAS CHANGED H, Y MUST BE RESCALED.
C* IF H OR METH HAS BEEN CHANGED, IDOUB IS RESET TO L + 1 TO PREVENT
C* FURTHER CHANGES IN H FOR THAT MANY STEPS.  ALSO, RC IS RESET.
C* RC IS THE RATIO OF NEW TO OLD VALUES OF THE COEFFICIENT  L(0)*H.
C* WHEN RC DIFFERS FROM 1 BY MORE THAN 30 PERCENT, OR THE CALLER HAS
C* CHANGED MITER, IEVAL IS SET TO MITER TO FORCE THE PARTIALS TO BE
C* UPDATED, IF PARTIALS ARE USED.
C***********************************************************************
  120 IF (MF .EQ. MFOLD) GO TO 150
      MEO = METH
      MIO = MITER
      METH = MF / 10
      MITER = MF - 10 * METH
      NPOLD = NP                                                        D257 408
      IF (MITER .NE. MIO) IWEVAL = MITER                                D257 409
      IF (METH .EQ. MEO) GO TO 150                                      D257 410
      IDOUB = L + 1                                                     D257 411
      IRET = 1                                                          D257 412
  130 CALL TESOC(METH,NQ,EL,TQ,MAXDER)                                  D257 413
      RC = RC * EL(1) / OLDL0                                           D257 414
      OLDL0 = EL(1)                                                     D257 415
  140 EDN = (TQ(1) * EPS) ** 2                                          D257 416
      E   = (TQ(2) * EPS) ** 2                                          D257 417
      EUP = (TQ(3) * EPS) ** 2                                          D257 418
      BND = (TQ(4) * EPS) ** 2                                          D257 419
      GO TO ( 160 , 170 , 200 ), IRET                                   D257 420
  150 IF (EPS .EQ. EPSOLD) GO TO 160                                    D257 421
      IRET = 1                                                          D257 422
      GO TO 140                                                         D257 423
  160 LMAX = MAXDER + 1                                                 D257 424
      EPSOLD = EPS                                                      D257 425
      IF (H .EQ. HOLD) GO TO 200                                        D257 426
      RH = H / HOLD                                                     D257 427
      H = HOLD                                                          D257 428
  170 RH = DMAX1(RH,HMIN/DABS(H))                                       D257 429
      RH=DMIN1(RH,HMAX/DABS(H),RMAX)
      R1 = 1.0D0                                                        D257 431
C                                                                       D257 432
      DO 180 J = 2, L                                                   D257 433
        R1 = R1 * RH                                                    D257 434
C                                                                       D257 435
        DO 180 I = JL,N                                                 D257 436
          Y(I,J) = Y(I,J) * R1                                          D257 437
  180 CONTINUE                                                          D257 438
C                                                                       D257 439
      H = H * RH                                                        D257 440
      RC = RC * RH                                                      D257 441
      IDOUB = L + 1                                                     D257 442
      IF (T .NE. TOLD) GO TO 690                                        D257 443
C***********************************************************************D257 444
C* THIS SECTION COMPUTES THE PREDICTED VALUES BY EFFECTIVELY            D257 445
C* MULTIPLYING THE Y ARRAY BY THE PASCAL TRIANGLE MATRIX.               D257 446
C***********************************************************************D257 447
  200 IF (DABS(RC-1.0D0) .GT. 0.3D0) IWEVAL = MITER                     D257 448
      T = T + H                                                         D257 449
C                                                                       D257 450
      DO 210 J1 = 1, NQ                                                 D257 451
C                                                                       D257 452
        DO 210 J2 = J1, NQ                                              D257 453
          J = NQ - J2 + J1                                              D257 454
C                                                                       D257 455
          DO 210 I=NL,N                                                 D257 456
            Y(I,J) = Y(I,J) + Y(I,J+1)                                  D257 457
  210 CONTINUE                                                          D257 458
C***********************************************************************D257 459
C* UP TO 3 CORRECTOR ITERATIONS ARE TAKEN.  CONVERGENCE IS TESTED       D257 460
C* BY REQUIRING CHANGES TO BE LESS THAN BND, WHICH IS DEPENDENT ON      D257 461
C* EPS, IN EUCLIDEAN NORM.  THE SUM OF THE CORRECTIONS IS ACCUMULATED   D257 462
C* IN THE VECTOR ERROR(I).  IT IS APPROXIMATELY EQUAL TO THE L-TH       D257 463
```

```
C* DERIVATIVE OF Y MULTIPLIED BY H**L/(FACTORIAL(L-1)*EL(L)), AND IS    D257 464
C* THUS PROPORTIONAL TO THE ACTUAL ERRORS TO THE LOWEST POWER OF         D257 465
C* H PRESENT (H**L).                                                     D257 466
C* THE Y ARRAY IS NOT ALTERED IN THE CORRECTION LOOP.  THE UPDATED       D257 467
C* Y VECTOR IS STORED TEMPORARILY IN PSAVE.  THE NORM OF THE             D257 468
C* ITERATE DIFFERENCE IS STORED IN D.                                    D257 469
C***********************************************************************D257 470
 220  DO 230 I=NL,N                                                      D257 471
 230     ERROR(I) = 0.0D0                                                D257 472
C                                                                        D257 473
      M = 0                                                              D257 474
      CALL YPAY(PSAVE(N1),A,Y,X0,A(INDXA),N)
C***********************************************************************D257 477
C* IF NECESSARY, THE PARTIALS ARE REEVALUATED PRIOR TO STARTING THE      D257 478
C* CORRECTOR ITERATION.  IWEVAL IS THEN SET TO 0 AS AN INDICATOR         D257 479
C* THAT THIS HAS BEEN DONE.                                              D257 480
C* IF MITER = 1 OR 2, THE MATRIX  P = I - L(0)*H*JACOBIAN  IS STORED     D257 481
C* IN PW AND SUBJECTED TO LU DECOMPOSITION, WITH THE RESULTS ALSO        D257 482
C* STORED IN PW.  IF MITER = 3, THE MATRIX USED IS  P = I - L(0)*H*D,    D257 483
C* WHERE D IS A DIAGONAL MATRIX.                                         D257 484
C***********************************************************************D257 485
  240 IF(H.EQ.HJAC) GO TO 360
      HJAC=H
      R=-EL(1)*H
C *** NELEA=NC. NONZERO ELE. IN MATRIX A
      DO 250 I=1,NELEA
  250 PW(I)=R*A(I)
  300 DO 305 I=1,N
      K=IDIAG(I)
  305 PW(K)=1.D0+PW(K)
      IWEVAL = 0                                                         D257 520
      RC = 1.0D0                                                         D257 521
 360  DO 370 I =NL,N                                                     D257 568
 370     PSAVE(I+N0) = PSAVE(I+N0) * H - Y(I,2) - ERROR(I)               D257 569
      CALL AXEB(PW,PSAVE,PSAVE(N1),A(INDXA),N)
 380  D = 0.0D0                                                          D257 572
C                                                                        D257 573
      DO 390 I =NL,N                                                     D257 574
         ERROR(I) = ERROR(I) + PSAVE(I)                                  D257 575
         D = D + (PSAVE(I) / YMAX(I)) ** 2                               D257 576
         PSAVE(I) = Y(I,1) + EL(1) * ERROR(I)                            D257 577
 390  CONTINUE                                                           D257 578
 410  IF (M .NE. 0) CRATE = DMAX1(.9D0*CRATE,D/D1)                       D257 590
      IF ((D*DMIN1(1.0D0,2.0D0*CRATE)) .LE. BND) GO TO 450               D257 591
      D1 = D                                                             D257 592
      M = M + 1                                                          D257 593
      IF (M .EQ. 3) GO TO 420                                            D257 594
      CALL YPAY(PSAVE(N1),A,PSAVE,X0,A(INDXA),N)
      GO TO 360
C***********************************************************************D257 597
C* THE CORRECTOR ITERATION FAILED TO CONVERGE IN 3 TRIES.  IF PARTIALS   D257 598
C* ARE INVOLVED BUT ARE NOT UP TO DATE, THEY ARE REEVALUATED FOR THE     D257 599
C* NEXT TRY.  OTHERWISE THE Y ARRAY IS RETRACTED TO ITS VALUES           D257 600
C* BEFORE PREDICTION, AND H IS REDUCED, IF POSSIBLE.  IF NOT, A          D257 601
C* NO-CONVERGENCE EXIT IS TAKEN.                                         D257 602
C***********************************************************************D257 603
```

```
 420  IF (IWEVAL .EQ. -1) GO TO 440
      T = TOLD
      RMAX = 2.0D0
C
      DO 430 J1 = 1, NQ
C
        DO 430 J2 = J1, NQ
          J = NQ - J2 + J1
C
          DO 430 I =NL,N
            Y(I,J) = Y(I,J) - Y(I,J+1)
 430  CONTINUE
C
      IF (DABS(H) .LE. (HMIN*1.00001D0)) GO TO 680
      RH=.25D0
      GO TO 170
 440  IWEVAL = MITER
      GO TO 220
C***********************************************************************
C* THE CORRECTOR HAS CONVERGED.  IWEVAL IS SET TO -1 IF PARTIAL
C* DERIVATIVES WERE USED, TO SIGNAL THAT THEY MAY NEED UPDATING ON
C* SUBSEQUENT STEPS.  THE ERROR TEST IS MADE AND CONTROL PASSES TO
C* STATEMENT 500 IF IT FAILS.
C***********************************************************************
 450  D = 0.0D0
C
      DO 460 I =NL,N
 460    D = D + (ERROR(I) / YMAX(I)) ** 2
C
      IF (MITER .NE. 0) IWEVAL = -1
      IF (D .GT. E) GO TO 500
C***********************************************************************
C* AFTER A SUCCESSFUL STEP, UPDATE THE Y ARRAY AND YMAX.
C* CONSIDER CHANGING H IF IDOUB = 1.  OTHERWISE DECREASE IDOUB BY 1.
C* IF IDOUB IS THEN 1 AND NQ .LT. MAXDER, THEN ERROR IS SAVED FOR
C* USE IN A POSSIBLE ORDER INCREASE ON THE NEXT STEP.
C* IF A CHANGE IN H IS CONSIDERED, AN INCREASE OR DECREASE IN ORDER
C* BY ONE IS CONSIDERED ALSO.  A CHANGE IN H IS MADE ONLY IF IT IS BY A
C* FACTOR OF AT LEAST 1.1.  IF NOT, IDOUB IS SET TO 10 TO PREVENT
C* TESTING FOR THAT MANY STEPS.
C***********************************************************************
      KFLAG = 0
C
      DO 470 J = 1, L
C
        DO 470 I =NL,N
          Y(I,J) = Y(I,J) + EL(J) * ERROR(I)
 470  CONTINUE
C
      DO 480 I =NL,N
 480    YMAX(I) = DMAX1(YMAX(I),DABS(Y(I,1)))
C
      IF (IDOUB .EQ. 1) GO TO 520
      IDOUB = IDOUB - 1
      IF (IDOUB .GT. 1) GO TO 700
      IF (NQ .EQ. MAXDER) GO TO 700
```

```
C
      DO 490 I = NL,N
  490   Y(I,LMAX) = ERROR(I)
C
      GO TO 700
C************************************************************************
C* THE ERROR TEST FAILED.  KFLAG KEEPS TRACK OF MULTIPLE FAILURES.
C* RESTORE T AND THE Y ARRAY TO THEIR PREVIOUS VALUES, AND PREPARE
C* TO TRY THE STEP AGAIN.  COMPUTE THE OPTIMUM STEP SIZE FOR THIS OR
C* ONE LOWER ORDER.
C************************************************************************
  500 KFLAG = KFLAG - 1
      T = TOLD
C
      DO 510 J1 = 1, NQ
C
        DO 510 J2 = J1, NQ
          J = NQ - J2 + J1
C
          DO 510 I = NL,N
            Y(I,J) = Y(I,J) - Y(I,J+1)
  510 CONTINUE
C
      RMAX = 2.0D0
      IF (DABS(H) .LE. (HMIN*1.00001D0)) GO TO 660
      IF (KFLAG .LE. -3) GO TO 640
      PR3 = 1.0D20
      GO TO 540
C************************************************************************
C* REGARDLESS OF THE SUCCESS OR FAILURE OF THE STEP, FACTORS
C* PR1, PR2, AND PR3 ARE COMPUTED, BY WHICH H COULD BE DIVIDED
C* AT ORDER NQ - 1, ORDER NQ, OR ORDER NQ + 1, RESPECTIVELY.
C* IN THE CASE OF FAILURE, PR3 = 1.0D20 TO AVOID AN ORDER INCREASE.
C* THE SMALLEST OF THESE IS DETERMINED AND THE NEW ORDER CHOSEN
C* ACCORDINGLY.  IF THE ORDER IS TO BE INCREASED, WE COMPUTE ONE
C* ADDITIONAL SCALED DERIVATIVE.
C************************************************************************
  520 PR3 = 1.0D20
      IF (NQ .EQ. MAXDER) GO TO 540
      D1 = 0.0D0
C
      DO 530 I = NL,N
  530   D1 = D1 + ((ERROR(I) - Y(I,LMAX)) / YMAX(I)) ** 2
C
      ENQ3 = 0.5D0 / DFLOAT(L+1)
      PR3 = ((D1 / EUP) ** ENQ3) * 1.4D0 + 1.4D-06
  540 ENQ2 = 0.5D0 / DFLOAT(L)
      PR2 = ((D / E) ** ENQ2) * 1.2D0 + 1.2D-06
      PR1 = 1.0D20
      IF (NQ .EQ. 1) GO TO 560
      D = 0.0D0
C
      DO 550 I = NL,N
  550   D = D + (Y(I,L) / YMAX(I)) ** 2
C
      ENQ1 = 0.5D0 / DFLOAT(NQ)
      PR1 = ((D / EDN) ** ENQ1) * 1.3D0 + 1.3D-06
  560 IF (PR2 .LE. PR3) GO TO 570
      IF (PR3 .LT. PR1) GO TO 590
  570 IF (PR2 .GT. PR1) GO TO 580
      NEWQ = NQ
      RH = 1.0D0 / PR2
      GO TO 620
  580 NEWQ = NQ - 1
C      INEVAL=NITER
      RH = 1.0D0 / PR1
      GO TO 620
  590 NEWQ = L
      RH = 1.0D0 / PR3
      IF (RH .LT. 1.1D0) GO TO 610
C
      DO 600 I =NL,N
  600   Y(I,NEWQ+1) = ERROR(I) * EL(L) / DFLOAT(L)
C
      GO TO 630
  610 IDOUB = 10
      GO TO 700
  620 IF ((KFLAG .EQ. 0).AND.(RH .LT. 1.1D0)) GO TO 610
C***********************************************************************
C* IF THERE IS A CHANGE OF ORDER, RESET NQ, L, AND THE COEFFICIENTS.
C* IN ANY CASE H IS RESET ACCORDING TO RH AND THE Y ARRAY IS RESCALED.
C* THEN EXIT FROM 690 IF THE STEP WAS OK, OR REDO THE STEP OTHERWISE.
C***********************************************************************
      IF (NEWQ .EQ. NQ) GO TO 170
  630 NQ = NEWQ
      L = NQ + 1
      IRET = 2
      GO TO 130
C***********************************************************************
C* CONTROL REACHES THIS SECTION IF 3 OR MORE FAILURES HAVE OCCURED.
C* IT IS ASSUMED THAT THE DERIVATIVES THAT HAVE ACCUMULATED IN THE
C* Y ARRAY HAVE ERRORS OF THE WRONG ORDER.  HENCE THE FIRST
C* DERIVATIVE IS RECOMPUTED, AND THE ORDER IS SET TO 1.  THEN
C* H IS REDUCED BY A FACTOR OF 10, AND THE STEP IS RETRIED.
C* AFTER A TOTAL OF 7 FAILURES, AN EXIT IS TAKEN WITH KFLAG = -2.
C***********************************************************************
  640 IF (KFLAG .EQ. -7) GO TO 670
      RH = .1D0
      RH = DMAX1(HMIN/DABS(H),RH)
      H = H * RH
      CALL YPAY(PSAVE,A,Y,X0,A(INDXA),N)
C
      DO 650 I =NL,N
  650   Y(I,2) = H * PSAVE(I)
C
      INEVAL   NITER
      IDOUB = 10
      IF (NQ .EQ. 1) GO TO 200
      NQ = 1
      L = 2
      IRET = 3
      GO TO 130
```

```
C***********************************************************************
C* ALL RETURNS ARE MADE THROUGH THIS SECTION.  H IS SAVED IN HOLD
C* TO ALLOW THE CALLER TO CHANGE H ON THE NEXT STEP.
C***********************************************************************
 660  KFLAG = -1
      GO TO 700
 670  KFLAG = -2
      GO TO 700
 680  KFLAG = -3
      GO TO 700
 685  KFLAG=-5
      GO TO 700
 690  RMAX = 10.0D0
 700  HOLD = H
      JSTART = NQ
      RETURN
C************** LAST CARD OF STIFF *************************************
      END
```

```
      SUBROUTINE TESOC(METH,NQ,EL,TQ,MAXDER)
C*
C
C   9/ 5/74DP
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PERTST(12,2,3),EL(13),TQ(4)
C***********************************************************************
C* TESOC IS CALLED BY STIFF AND SETS COEFFICIENTS FOR USE THERE.
C* THE VECTOR EL, OF LENGTH NQ + 1, DETERMINES THE BASIC METHOD.
C* THE VECTOR TQ, OF LENGTH 3, IS INVOLVED IN ADJUSTING THE STEP SIZE
C* IN RELATION TO TRUNCATION ERROR.  ITS VALUES ARE GIVEN BY THE
C* PERTST ARRAY.
C* THE VECTORS EL AND TQ DEPEND ON METH AND NQ.
C* TESOC ALSO SETS MAXDER, THE MAXIMUM ORDER OF THE METHOD AVAILABLE.
C* CURRENTLY IT IS 12 FOR THE ADAMS METHODS AND 5 FOR THE GEAR METHODS.
C* LMAX = MAXDER + 1 IS THE NUMBER OF COLUMNS IN THE Y ARRAY.
C* THE MAXIMUM ORDER USED MAY BE REDUCED SIMPLY BY CHANGING THE
C* NUMBERS IN STATEMENTS 1 AND 2 BELOW.
C*
C* THE COEFFICIENTS IN PERTST NEED BE GIVEN TO ONLY ABOUT
C* ONE PERCENT ACCURACY.  THE ORDER IN WHICH THE GROUPS APPEAR BELOW
C* IS..  COEFFICIENTS FOR ORDER NQ - 1, COEFFICIENTS FOR ORDER NQ,
C* COEFFICIENTS FOR ORDER NQ + 1.  WITHIN EACH GROUP ARE THE
C* COEFFICIENTS FOR THE ADAMS METHODS, FOLLOWED BY THOSE FOR THE
C* GEAR METHODS.
C*
C* THE ORIGINAL VERSION OF THIS PROGRAM WAS WRITTEN AT LLL BY A. C.
```

```
C* HINDMARSH FOR CDC COMPUTERS.  THE CDC VERSION WAS MODIFIED FOR       D257 818
C* USE ON IBM COMPUTERS AT ARGONNE IN JUNE, 1973.                       D257 819
C*                                                                      D257 820
C* LATEST REVISION  SEPTEMBER, 1973                                     D257 821
C*                                                                      D257 822
C***********************************************************************D257 823
       DATA PERTST / 1.0D0,1.0D0,2.0D0,1.0D0,.3158D0,.07407D0,.01391D0,  D257 824
     1 .002182D0,.0002945D0,.00003492D0,.000003692D0,.0000003524D0,     D257 825
     1 1.D0,1.D0,.5D0,.1667D0,.04167D0,1.D0,1.D0,1.D0,1.D0,1.D0,1.D0,   D257 826
     1 1.D0,2.D0,12.D0,24.D0,37.89D0,53.33D0,70.08D0,87.97D0,106.9D0,   D257 827
     1 126.7D0,147.4D0,168.8D0,191.0D0,                                 D257 828
     1 2.0D0,4.5D0,7.333D0,10.42D0,13.7D0,1.D0,1.D0,1.D0,1.D0,1.D0,1.D0,D257 829
     1 1.0D0,12.0D0,24.0D0,37.89D0,53.33D0,70.08D0,87.97D0,106.9D0,     D257 830
     1 126.7D0,147.4D0,168.8D0,191.0D0,1.D0,3.0D0,                      D257 831
     1 6.0D0,9.167D0,12.5D0,1.D0,1.D0,1.D0,1.D0,1.D0,1.D0,1.D0,1.0D0/   D257 832
C*                                                                      D257 833
C***********************************************************************D257 834
C*                                                                      D257 835
       EL(2) = 1.0D0                                                    D257 836
       GO TO (1,2), METH                                                D257 837
  1    MAXDER = 12                                                      D257 838
       GO TO (101,102,103,104,105,106,107,108,109,110,111,112), NQ      D257 839
  2    MAXDER = 5                                                       D257 840
       GO TO (201,202,203,204,205), NQ                                  D257 841
C***********************************************************************D257 842
C* THE FOLLOWING COEFFICIENTS SHOULD BE DEFINED TO                      D257 843
C* MACHINE ACCURACY.  FOR EACH ORDER NQ, THEY CAN BE CALCULATED         D257 844
C* FROM THE GENERATING POLYNOMIAL,                                      D257 845
C*      L(T) = EL(1) + EL(2)*T + ... + EL(NQ+1)*T**NQ.                  D257 846
C* FOR THE IMPLICIT ADAMS METHODS, L(T) IS GIVEN BY                     D257 847
C*      DL/DT = (T+1)*(T+2)* ... *(T+NQ-1)/K,     L(-1) = 0,            D257 848
C* WHERE                    K = FACTORIAL(NQ-1).                        D257 849
C* FOR THE GEAR METHODS,                                                D257 850
C*      L(T) = (T+1)*(T+2)* ... *(T+NQ)/K,                              D257 851
C* WHERE            K = FACTORIAL(NQ)*(1 + 1/2 + ... + 1/NQ).           D257 852
C*                                                                      D257 853
C* THE ORDER IN WHICH THE GROUPS APPEAR BELOW IS..                      D257 854
C* IMPLICIT ADAMS METHODS OF ORDERS 1 TO 12,                            D257 855
C* STIFFLY STABLE GEAR METHODS OF ORDERS 1 TO 5.                        D257 856
C***********************************************************************D257 857
 101   EL(1) = 1.0D0                                                    D257 858
       GO TO 900                                                        D257 859
 102   EL(1) = 0.5D0                                                    D257 860
       EL(3) = 0.5D0                                                    D257 861
       GO TO 900                                                        D257 862
 103   EL(1) = 4.1666666666666667D-01                                   D257 863
       EL(3) = 0.75D0                                                   D257 864
       EL(4) = 1.6666666666666667D-01                                   D257 865
       GO TO 900                                                        D257 866
 104   EL(1) = 0.375D0                                                  D257 867
       EL(3) = 9.1666666666666667D-01                                   D257 868
       EL(4) = 3.3333333333333333D-01                                   D257 869
       EL(5) = 4.1666666666666667D-02                                   D257 870
       GO TO 900                                                        D257 871
 105   EL(1) = 3.4861111111111111D-01                                   D257 872
       EL(3) = 1.0416666666666667D0                                     D257 873
```

```
      EL(4)  = 4.8611111111111111D-01                                   D257 874
      EL(5)  = 1.0416666666666667D-01                                   D257 875
      EL(6)  = 9.3333333333333333D-03                                   D257 876
      GO TO 900                                                         D257 877
106   EL(1)  = 3.2986111111111111D-01                                   D257 878
      EL(3)  = 1.1416666666666667D+00                                   D257 879
      EL(4)  = 0.625D+00                                                D257 880
      EL(5)  = 1.7708333333333333D-01                                   D257 881
      EL(6)  = 0.025D+00                                                D257 882
      EL(7)  = 1.3888888888888889D-03                                   D257 883
      GO TO 900                                                         D257 884
107   EL(1)  = 3.1559193121693122D-01                                   D257 885
      EL(3)  = 1.225D+00                                                D257 886
      EL(4)  = 7.5185185185185185D-01                                   D257 887
      EL(5)  = 2.5520833333333333D-01                                   D257 888
      EL(6)  = 4.8611111111111111D-02                                   D257 889
      EL(7)  = 4.8611111111111111D-03                                   D257 890
      EL(8)  = 1.9841269841269841D-04                                   D257 891
      GO TO 900                                                         D257 892
108   EL(1)  = 3.0422453703703704D-01                                   D257 893
      EL(3)  = 1.2464285714285714D+00                                   D257 894
      EL(4)  = 9.6651851851851852D-01                                   D257 895
      EL(5)  = 3.3576388885888889D-01                                   D257 896
      EL(6)  = 7.7777777777777776D-02                                   D257 897
      EL(7)  = 1.0549148148148148D-02                                   D257 898
      EL(8)  = 7.9365079365079365D-04                                   D257 899
      EL(9)  = 2.4801587301587302D-05                                   D257 900
      GO TO 900                                                         D257 901
109   EL(1)  = 2.9486600044091711D-01                                   D257 902
      EL(3)  = 1.3589285714285714D+00                                   D257 903
      EL(4)  = 9.7655423280423280D-01                                   D257 904
      EL(5)  = 0.4171875D+00                                            D257 905
      EL(6)  = 1.1135416666666667D-01                                   D257 906
      EL(7)  = 0.01875D+00                                              D257 907
      EL(8)  = 1.9345238095238095D-03                                   D257 908
      EL(9)  = 1.1160714285714286D-04                                   D257 909
      EL(10)= 2.7557319223985891D-06                                    D257 910
      GO TO 900                                                         D257 911
110   EL(1)  = 2.8697544642857143D-01                                   D257 912
      EL(3)  = 1.4144841269841270D+00                                   D257 913
      EL(4)  = 1.0772156084656085D+00                                   D257 914
      EL(5)  = 4.9856701940035273D-01                                   D257 915
      EL(6)  = 0.1484375D+00                                            D257 916
      EL(7)  = 2.9060570987654321D-02                                   D257 917
      EL(8)  = 3.7202380952380952D-03                                   D257 918
      EL(9)  = 2.9968584656084656D-04                                   D257 919
      EL(10)= 1.3778659611992945D-05                                    D257 920
      EL(11)= 2.7557319223985891D-07                                    D257 921
      GO TO 900                                                         D257 922
111   EL(1)  = 2.8018959644393672D-01                                   D257 923
      EL(3)  = 1.4644841269841270D+00                                   D257 924
      EL(4)  = 1.1715145502645503D+00                                   D257 925
      EL(5)  = 5.7935819003527337D-01                                   D257 926
      EL(6)  = 1.8832286155202822D-01                                   D257 927
      EL(7)  = 4.1430362654320988D-02                                   D257 928
      EL(8)  = 6.2111441798941799D-03                                   D257 929
      EL(9)  = 6.2520667989417989D-04
      EL(10)= 4.0417401523512640D-05
      EL(11)= 1.5156525573192240D-06
      EL(12)= 2.5052108385441719D-08
      GO TO 900
 112  EL(1)  = 2.7426554003159906D-01
      EL(3)  = 1.5039386724386724D+00
      EL(4)  = 1.2602711640211640D+00
      EL(5)  = 6.5923419209876543D-01
      EL(6)  = 2.3045800264550265D-01
      EL(7)  = 5.5697246105232216D-02
      EL(8)  = 9.4394841259941270D-03
      EL(9)  = 1.1192749669312169D-03
      EL(10)= 9.0939153439153439D-05
      EL(11)= 4.8225308641975309D-06
      EL(12)= 1.5031265031265031D-07
      EL(13)= 2.0876756987868099D-09
      GO TO 900
 201  EL(1)  = 1.0D+00
      GO TO 900
 202  EL(1)  = 6.6666666666666667D-01
      EL(3)  = 3.3333333333333333D-01
      GO TO 900
 203  EL(1)  = 5.4545454545454545D-01
      EL(3)  = EL(1)
      EL(4)  = 9.0909090909090909D-02
      GO TO 900
 204  EL(1)  = 0.48D+00
      EL(3)  = 0.7D+00
      EL(4)  = 0.2D+00
      EL(5)  = 0.02D+00
      GO TO 900
 205  EL(1)  = 4.3795620437956204D-01
      EL(3)  = 8.2116788321167883D-01
      EL(4)  = 3.1021897810218978D-01
      EL(5)  = 5.4744525547445255D-02
      EL(6)  = 3.6496350364963504D-03
C*
 900  DO 910 K = 1, 3
 910     TQ(K)  = PERTST(NQ,METH,K)
C*
      TQ(4) =0.5D0*TQ(2)/DFLOAT(NQ+2)
      RETURN
C****************LAST CARD OF TESCO********************
      END
```

```
      SUBROUTINE YOUT(NO,TLAST,Y,NQ)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION Y(NO,13)
      COMMON/SC1235/T,H,HMIN,HMAX,EPS1,MF1,KFLAG1,JSTART,N,NINV,NXO,
     1YMAX(400),ERROR(400),FSAVE(800),YPRINT(100)
C *** NEED DIMENSION YMAX(NXO),ERROR(NXO),FSAVE(2*NXO;
      COMMON/PRINT/TPRL(10),DTPR(10),TPRINT,IPRT,NPR,NPRT,MAXPRT(10)
    5 CONTINUE
      S = (TPRINT - T) / H
      DO 1 I =  1,N
    1 YPRINT(I)  = Y(I,1)
      R = 1.0D0
      L = NQ + 1
      DO 3 J = 2, L
         R = R * S
         DO 2 I =  1,N
    2    YPRINT(I)  = YPRINT(I) + Y(I,J) * R
    3 CONTINUE
      CALL YWA(YPRINT,N)
      NPR=NPR+1
      IF(NPR.LE.MAXPRT(IPRT)) GO TO  100
      IPRT=IPRT+1
      NPR=0
  100 TPRINT=TPRL(IPRT)+NPR*DTPR(IPRT)
      IF (TPRINT.LE.T) GO TO 5
  500 RETURN
      END
```

```
      SUBROUTINE YOUTD
      PRINT  1
    1 FORMAT('0CALLED YOUTD.  SHOULD WE PRINT.')
      RETURN
      END
```

```
      SUBROUTINE MATRIX(A,NWRDA,INDXA,IDIAG,XO,NXO,N,LB,NBLK,NLR,NDSUBM)
C
C  A(NWRDA)
C     CONTAINS NON-ZERO ENTRIES OF THE COEFFICIENT MATRIX
C     A(IDIAG(I)) = DIAGONAL ELEMENT OF ITH ROW OF THE COEFFICIENT MATRIX
```

```
C
C  CINDEX(1510)
C     CINDEX(I) = COLUMN INDEX OF A(I)
C
C  ICLASS(10)
C     ICLASS(I) = CLASS OF THE ITH MEMBER OF CHAIN
C
C  ICOM(10,10)
C     ICOM(I,J) = NO. OF COMPARTMENT FOR DAUGHTER I, ORGAN J
C
C  INDXA( )
C     INDXA(2,2*I-1) = -10*(NO. OF NON-ZERO ELEMENTS IN
C                          ROW I OF A
C     INDXA(2,2*I) = 8*(NO. OF NON-ZERO ELEMENTS IN
C                          ROW I OF A
C     INDXA(1,I) = 8*(J-1) IFF A(I) BELONGS IN COLUMN
C                          J OF THE MATRIX A
C
C  ISORS(20)
C     ISORS(I) = INDEX OF THE ITH SOURCE ORGAN (INDICES FROM SEE CODE)
C
C  LAMAB(10,4)
C     LAMAB(I,1) = LAMBDA OF ABSORPTION IN S FOR DAUGHTER I
C     LAMAB(I,2) = LAMBDA OF ABSORPTION IN SI FOR DAUGHTER I
C     LAMAB(I,3) = LAMBDA OF ABSORPTION IN ULI FOR DAUGHTER I
C     LAMAB(I,4) = LAMBDA OF ABSORPTION IN LLI FOR DAUGHTER I
C
C  LAMR(10)
C     LAMR(I) = LN 2/(HALFLIFE OF DAUGHTER I)
C
C  LAMX(10,10)
C     LAMX(I,J) = LN 2/(HALFIME OF DAUGHTER I IN COMPARTMENT J)
C
C  LP(K) = LAMR(K)
C
C  NBLK(MDSUBM)
C     NBLK(I) = ORDER OF SQUARE DIAGIONAL SUBMATRIX I OF A
C     NBLK(I) = NCOM(I)
C
C  NCOM(10)
C     NCOM(I) = NO. OF COMPARTMENTS FOR DAUGHTER I
C
C  ORGNM(10)
C     ORGNM(I) = NAME OF THE ITH ORGAN
C
C  PRNT(1000)
C     PRNT(I) = ITH ENTRY OF A GIVEN ROW OF THE COEFFICIENT MATRIX
C
C  RINDEX(1510)
C     RINDEX(I) = ROW INDEX OF A(I)
C
C  TAG(10)
C     TAG(I) - ATOMIC SYMBOL AND ATOMIC NUMBER FOR DAUGHTER I
C
C  TR(10)
C     TR(I) = HALFLIFE OF DAUGHTER I
```

```
C
C  XO(NXO) - VECTOR OF INITIAL VALUES
C
C
C
C
C  EXPLANATION OF BRANCHING
C
C     THE THREE TYPES OF BRANCHING AVAILABLE IN
C     THIS CODE WILL BE OF THE FOLLOWING FORMS
C
C     1.   A
C          P*  B    (1,1)
C
C     2.   A
C          P*  B          (2,1)
C          Q*  1*  C      (2,2)
C
C     3.   A
C          P*  B               (3,1)
C          Q*  XX  C           (3,2)            XX-INDICATES BLOCK OF ZEROES
C          XX  1*  1*  D       (3,3)
C
C     THE ORDERED PAIR (X,Y) INDICATES TYPE
C     OF BRANCHING (X=1,2,3) AND POSITION
C     WITHIN THE BRANCHING SCHEME - X=NBRNCH,Y=IBRNCH.
C
C     I. FOR NBRNCH= 0 OR 1, THE ORDER OF
C     EXECUTION IS UNCHANGED; HOWEVER,
C          P(K)=1 IF NBRNCH=0
C          P(K)=1 IF NBRNCH=1
C     P(K) IS READ IN ON CARD.
C
C     II. FOR NBRNCH=2
C          1. THE CASE P* B IS ESSENTIALLY
C             NBRNCH=1 AND WILL BE TREATED AS SUCH.
C          2. Q* 1* C - THE BLOCK Q* MUST BEGIN
C             IN THE SAME COLUMN AS P*, AND 1* MUST
C             BEGIN IN THE SAME COLUMN AS B.  Q MUST
C             BE SET TO 1 WHEN FORMING 1*.
C
C     III. FOR NBRNCH=3
C          1. THE CASE P* B IS ESSENTIALLY
C             NBRNCH=1 AND WILL BE TREATED AS SUCH
C          2. Q* XX C - THE BLOCK Q* MUST BEGIN
C             IN THE SAME COLUMN AS P*.  C WILL BEGIN
C             IN COLUMN NCOM(A)+NCOM(B)+1.
C          3. XX 1* 1* D - THE FIRST FILLED BLOCK 1*
C             WILL BEGIN IN COLUMN NCOM(A)+1. THE
C             SECOND BLOCK 1* WILL BEGIN IN SAME
C             COLUMN AS C. THIS CASE IS SIMILAR
C             TO NBRNCH=2,IBRNCH=2 AND MAY BE
C             TREATED AS SUCH BY ENTERING Q=1 ON
C             INPUT CARD.
C     THUS II.2 AND III.2 ARE THE ONLY TWO
C     DIFFERENT CASES FOR WHICH SPECIAL LOGIC MUST BE EMPLOYED
```

```
C                                                                       MAT 1180
C                                                                       MAT 1190
      IMPLICIT REAL*8 (A-H,O-Z)                                         MAT 1200
      REAL*8 LP(1),LAMX(10,10),LAMB(10),LAMAB(10,4)                     MAT 1210
      INTEGER*2 RINDEX(12001),CINDEX(12001),SUM1                        MAT 1220
      DIMENSION A(1),XO(1),INDXA(2,1),IDIAG(1),ICLASS(10),TR(10),       MAT 1230
     1  PRNT(800), NBLK(1),ISOBS(20),P(10),NBRNCH(10),IBRNCH(10)        MAT 1231
C                                                                       MAT 1240
      COMMON/TABLES/ LAMAB(10,4),LAMX(10,10),ORGNM(10),TAG(10),         MAT 1250
     1  ICON(10,10),IOPT,NCON(10),NISO,NORG                             MAT 1251
      COMMON/SWTCH/ ISWTCH                                              MAT 1260
C                                                                       MAT 1270
      DATA FLN2/.693147180559945D0/                                     MAT 1280
      DATA ISOBS/8,9,10,11,2,3,4,5/                                     MAT 1290
C                                                                       MAT 1300
C                                                                       MAT 1310
C  READ NISO = NO. OF RADIONUCLIDES IN CHAIN                            MAT 1320
C       D3,D4,D5 = REGIONAL DEPOSITION VALUES                           MAT 1330
C       NORG = NO. OF ORGANS (EXCLUDING LUNG AND GI                     MAT 1340
C       IOPT = 0  INTAKE BY INHALATION                                  MAT 1350
C              1  INTAKE BY INGESTION                                   MAT 1360
C              2  INTAKE BY INJECTION                                   MAT 1370
C                                                                       MAT 1380
      READ(50,1001) NISO,D3,D4,D5,NORG,IOPT,NORAL                       MAT 1390
 1001 FORMAT(I10,3F10.3,3I10)                                           MAT 1400
      WRITE(10,1002)                                                    MAT 1410
 1002 FORMAT('1')                                                       MAT 1420
      IF(IOPT.EQ.0) WRITE(10,1003) D3,D4,D5                             MAT 1430
 1003 FORMAT(' PERCENT DEPOSITION'/' ',4X,'N-P',12X,'T-B',              MAT 1440
     1  12X,'P'/' ',F7.2,4(8X,F7.2),/)                                  MAT 1441
      WRITE(16) NISO,NORG,IOPT                                          MAT 1450
      IF(IOPT.EQ.1) WRITE(16) NORAL                                     MAT 1460
C                                                                       MAT 1470
C  ASSIGN REGIONAL DEPOSITION FOR EACH PATHWAY                          MAT 1480
C                                                                       MAT 1490
      IF(IOPT.EQ.0) CALL LUNG(D3,D4,D5)                                 MAT 1500
C                                                                       MAT 1510
C  FOR EACH RADIONUCLIDE READ                                           MAT 1520
C       TAG = ATOMIC NO. AND SYMBOL                                     MAT 1530
C       TR = HALFLIFE                                                   MAT 1540
C       ICLASS = CLASS                                                  MAT 1550
C       ILAB = 0  ABSOPTION IN SI ONLY                                  MAT 1560
C              1  ABSOPTION IN ALL REGIONS OF GI TRACT                  MAT 1570
C                                                                       MAT 1580
      WRITE(10,1004)                                                    MAT 1590
 1004 FORMAT(' ISOTOPE',4X,'HALFLIFE',4X,'CLASS',3X,'LAMAB',            MAT 1600
     1  3X,'BR.RATIO',7X,'NBRNCH',4X,'IBRNCH')                          MAT 1601
      DO 15 I=1,NISO                                                    MAT 1610
      DO 10 J=1,4                                                       MAT 1620
      LAMAB(I,J)=0.                                                     MAT 1630
 10   CONTINUE                                                          MAT 1640
      ISWTCH=0                                                          MAT 1650
      READ(50,1008) TAG(I),TR(I),ICLASS(I),LAMAB(I,2),P(I),             MAT 1660
     1   NBRNCH(I),IBRNCH(I),ILAB                                       MAT 1661
      IF(ILAB.EQ.1) READ(5,1005) LAMAB(I,1),LAMAB(I,3),LAMAB(I,4)       MAT 1670
 1005 FORMAT(3F10.0)                                                    MAT 1680
```

```
      IF(P(I).EQ.0.) P(I)=1.                                            MAT 1690
      WRITE(10,1006) TAG(I),TR(I),ICLASS(I),LAMAB(I,2),P(I),            MAT 1700
     1  NBRNCH(I),IBRNCH(I)                                             MAT 1701
 1006 FORMAT(' ',A8,1PE12.3,4X,A4,2E10.3,2I10)                          MAT 1710
      IF(ILAB.EQ.1) WRITE(10,1007) LAMAB(I,1),LAMAB(I,3),LAMAB(I,4)     MAT 1720
 1007 FORMAT(' ',21X,'LAMAB-S',1PE10.3,/' ',19X,'LAMAB-ULI',E10.3,/     MAT 1730
     1  ' ',19X,'LAMAB-LLI',E10.3)                                      MAT 1731
 15   CONTINUE                                                          MAT 1740
 1008 FORMAT(A8,2X,F10.0,8X,A2,2F10.0,2I2,2X,I2)                        MAT 1750
      WRITE(16) (TAG(I),I=1,NISO)                                       MAT 1760
C                                                                       MAT 1770
C  SET LAMR                                                             MAT 1780
C                                                                       MAT 1790
      DO 20 I=1,NISO                                                    MAT 1800
      LAMR(I)=FLN2/TR(I)                                                MAT 1810
 20   CONTINUE                                                          MAT 1820
C                                                                       MAT 1830
C                                                                       MAT 1840
C  CHECK CLASS;  SET LAMX AND FRAC                                      MAT 1850
C                                                                       MAT 1860
      IF(IOPT.EQ.0) CALL LUNG1(ICLASS,X0,NX0)                           MAT 1870
C                                                                       MAT 1880
C                                                                       MAT 1890
C  CALCULATE ENTRIES OF COEFFICIENT MATRIX                              MAT 1900
C     ENTER NON-ZERO ENTRIES BY ROWS TO FORM VECTOR A                   MAT 1910
C                                                                       MAT 1920
      IROW=0                                                            MAT 1930
      IKOUNT=0                                                          MAT 1940
      ICOL1=0                                                           MAT 1950
      ICOL=0                                                            MAT 1960
      DO 30 K=1,NISO                                                    MAT 1970
C  K - INDEX OF RADIONUCLIDE                                            MAT 1980
C  J - INDEX OF COMPARTMENT                                             MAT 1990
      J=0                                                               MAT 2000
      ICOL=ICOL1                                                        MAT 2010
 25   CONTINUE                                                          MAT 2020
C                                                                       MAT 2030
C  LUNG COMPARTMENTS                                                    MAT 2040
C                                                                       MAT 2050
      IF(IOPT.EQ.0) CALL LUNG2(A,RINDEX,CINDEX,IROW,ICOL,ICOL1,         MAT 2060
     1  IKOUNT,LAMR,P,K,J,N,NBRNCH,IBRNCH)                              MAT 2061
C                                                                       MAT 2070
C                                                                       MAT 2080
C                                                                       MAT 2090
C  GI TRACT                                                             MAT 2100
C                                                                       MAT 2110
      IF(IOPT.EQ.0.OR.IOPT.EQ.1) CALL GI(A,RINDEX,CINDEX,IROW,          MAT 2120
     1  ICOL,ICOL1,IKOUNT,LAMR,P,K,J,N,NBRNCH,IBRNCH,X0,NX0)            MAT 2121
C                                                                       MAT 2130
C                                                                       MAT 2140
C                                                                       MAT 2150
C  OTHER ORGANS                                                         MAT 2160
C                                                                       MAT 2170
      IF(IOPT.EQ.0.OR.IOPT.EQ.1.OR.IOPT.EQ.2) CALL ORGAN(A,RINDEX,      MAT 2180
     1  CINDEX,IROW,ICOL,ICOL1,IKOUNT,LAMR,P,K,J,N,NBRNCH,IBRNCH,       MAT 2181
     2  ISORS,X0,NX0)                                                   MAT 2182
```

```
C
      IF(K.EQ.1) GO TO 30
      CALL INDEX(ICOL1,NCOM,K,NBRNCH(K),IBRNCH(K))
 30   CONTINUE
      IF(IOPT.EQ.0) NORGP8=NORG+8
      IF(IOPT.EQ.1) NORGP8=NORG+8
      IF(IOPT.EQ.2) NORGP8=NORG+8
      IF(IOPT.EQ.0) IORG=1
      IF(IOPT.EQ.1) IORG=1
      IF(IOPT.EQ.2) IORG=1
      WRITE(16) (ISOPS(I),I=IORG,NORGP8)
C
C  DETERMINE TOTAL NUMBER OF COMPARTMENTS (SUM1)
C
      SUM1=0.D0
      DO 35 K=1,NISO
      SUM1=SUM1+NCOM(K)
 35   CONTINUE
C
C
C  PRINT COEFFICIENT MATRIX BY ROWS
C
      WRITE(51,1009)
 1009 FORMAT('1MATRIX A PRINTED BY ROWS')
      II=0
      DO 55 I=1,SUM1
      III=0
      DO 40 J=1,SUM1
      PRNT(J)=0.D0
 40   CONTINUE
      WRITE(51,1010)
 1010 FORMAT('0')
 45   CONTINUE
      II=II+1
      III=III+1
      IF(RINDEX(II).EQ.I) GO TO 50
      II=II-1
      III=III-1
      INDXA(2,2*I-1)=-16*III
      INDXA(2,2*I)=8*III
      IDIAG(I)=II
      WRITE(51,1011) (PRNT(J),J=1,SUM1)
 1011 FORMAT(' ',8D15.6)
      GO TO 55
 50   CONTINUE
      PRNT(CINDEX(II))=A(II)
      INDXA(1,II)=8*(CINDEX(II)-1)
      GO TO 45
 55   CONTINUE
C
      DO 60 I=1,NISO
      LB(I)=-LAMR(I)
      NBLK(I)=NCOM(I)
 60   CONTINUE
      NOSUBM=NISO
      N=SUM1
      NELA=IDIAG(N)
C
C  PRINT NON-ZERO ELEMENTS OF THE COEFFICIENT MATRIX AND
C  THE ROW AND COLUMN IN WHICH EACH ONE APPEARS
C
      WRITE(6,1012)
 1012 FORMAT('1',3X,'ENTRIES OF A',2X,'ROW',2X,'COL')
      DO 65 I=1,NELA
      WRITE(51,1013) A(I),RINDEX(I),CINDEX(I)
 1013 FORMAT(' ',D15.6,2I5)
 65   CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE LUNG(D3,D4,D5)
C
C  THIS SUBROUTINE DETERMINES THE ENTRIES OF THE COEFFICIENT
C  MATRIX CORRESPONDING TO THE LUNG COMPARTMENTS
C
C  AMUT(10)
C     AMUT(I) = LAMR(K)*FRAC(K,I)   FOR PATHWAY I, DAUGHTER K
C
C  D(9) - REGIONAL DEPOSITION VALUES
C     D(1) = D3      D(5) = D5
C     D(2) = D3      D(6) = D5
C     D(3) = D4      D(7) = D5
C     D(4) = D4      D(8) = D5
C
C  F(9, 3)
C     F(X,I) = REGIONAL FRACTION TO PATHWAY
C                    X = A,B,C,...J
C                    I = 1  CLASS D
C                      2  CLASS W
C                      3  CLASS Y
C
C  FRAC(10, 9)
C     FRAC(I,J) = REGIONAL FRACTION IN PATHWAY J OF DAUGHTER I
C
C  IC(10)
C     IC(I) = CLASS INDEX FOR DAUGHTER I
C              1 - CLASS D
C              2 - CLASS W
C              3 - CLASS Y
C
C  LAMBDA(3,9)
C     LAMBDA(J,X) = LN 2/(REMOVAL HALFTIME FOR PATHWAY
C                         I = A,B,C,...I)                                 LUNG 330
C                           J = 1  CLASS D                                LUNG 340
C                               2  CLASS W                                LUNG 350
C                               3  CLASS Y                                LUNG 360
C                                                                         LUNG 370
      IMPLICIT REAL*8 (A-H,O-Z)                                           LUNG 380
      REAL*8 LAMB(1),LAMBDA(3,9),LAMX(10,10),LAMAB(10,4)                  LUNG 390
      INTEGER*2 PINDEX(1),CINDEX(1)                                       LUNG 400
      DIMENSION A(1),FRAC(10,9),F(9,3),ICLASS(1),IC(10),D(9),             LUNG 410
     1  P(1),NBRNCH(1),IBRNCH(1),AMNT(10),XO(1)                           LUNG 411
      INTEGER DAY,WEEK,YEAR                                               LUNG 420
C                                                                         LUNG 430
      COMMON/TABLES/ LAMAB(10,4),LAMX(10,10),ORGNM(10),TAG(10),           LUNG 440
     1  ICOM(10,10),IOPT,NCOM(10),NISO,NORG                               LUNG 441
C                                                                         LUNG 450
      DATA F/.5D0,.5D0,.95D0,.05D0,.8D0,0.D0,0.D0,.2D0,1.D0,              LUNG 460
     1         .1D0,.9D0,.5D0,.5D0,.15D0,.4D0,.4D0,.05D0,1.D0,            LUNG 461
     2         .01D0,.99D0,.01D0,.99D0,.05D0,.4D0,.4D0,.15D0,.9D0/        LUNG 462
C                                                                         LUNG 470
      DATA LAMBDA/3*69.3147180559945D0,                                   LUNG 480
     1  69.3147180559945D0,2*1.7328679513998D0,                           LUNG 481
     2  3*69.3147180559945D0,                                             LUNG 482
     3  3*3.4657359027997 25D0,                                           LUNG 483
     4  1.3862943611198 9D0,.01386294361119 89D0,.00138629436111989D0,    LUNG 484
     5  0.D0,2*.6931471805599 45D0,                                       LUNG 485
     6  0.D0,.01386294361119 89D0,.00138629436111989D0,                   LUNG 486
     7  1.3862943611198 9D0,.01386294361119 89D0,.00138629436111989D0 ,   LUNG 487
     8  1.3862943611198 9D0,.01386294361119 89D0,.000693147180559945D0/   LUNG 488
C                                                                         LUNG 490
      DATA DAY/' D  '/,WEEK/' W  '/,YEAR/' Y  '/                          LUNG 500
C                                                                         LUNG 510
C                                                                         LUNG 520
C  ASSIGN REGIONAL DEPOSITION FOR EACH PATHWAY                            LUNG 530
C                                                                         LUNG 540
      D(1)=D3                                                             LUNG 550
      D(2)=D3                                                             LUNG 560
      D(3)=D4                                                             LUNG 570
      D(4)=D4                                                             LUNG 580
      D(5)=D5                                                             LUNG 590
      D(6)=D5                                                             LUNG 600
      D(7)=D5                                                             LUNG 610
      D(8)=D5                                                             LUNG 620
      RETURN                                                              LUNG 630
C                                                                         LUNG 640
      ENTRY LUNG1(ICLASS,XO,NXO)                                          LUNG 650
C                                                                         LUNG 660
C  CHECK CLASS                                                            LUNG 670
C                                                                         LUNG 680
      DO 45 I=1,NISO                                                      LUNG 690
      IF(ICLASS(I).NE.DAY) GO TO 10                                       LUNG 700
      IC(I)=1                                                             LUNG 710
      GO TO 25                                                            LUNG 720
   10 CONTINUE                                                            LUNG 730
      IF(ICLASS(I).NE.WEEK) GO TO 15                                      LUNG 740
      IC(I)=2                                                             LUNG 750
      GO TO 25                                                            LUNG 760
```

```
 15   CONTINUE
      IF (ICLASS(I).NE.YEAR) GO TO 20
      IC(I) =3
      GO TO 25
 20   CONTINUE
      WRITE(10,1001)
 1001 FORMAT('0',11X,' ERROR.  CLASS DOES NOT COMPARE')
      CALL ERROR
 25   CONTINUE
C
C  SET LAMX AND FRAC
C
      DO 30 J=1,9
      LAMX(I,J) =LAMBDA(IC(I),J)
      FRAC(I,J)=F(J,IC(I))
 30   CONTINUE
      LAMX(I,10)=0.D0
C
C  IF PARENT IS CLASS Y AND DAUGHTER IS CLASS D OR W
C     SET LAMX(I,10) TO LAMX(I,9)
C
      IF(I.EQ.1) GO TO 35
      IF (IC(I-1).EQ.3.AND.(IC(I).EQ.1.OR.IC(2).EQ.2))
     1  LAMX(I,10)=LAMX(I,9)
 35   CONTINUE
      IF (I.GT.1) GO TO 45
C
C  SET INITIAL VALUES FOR ODE
C
      DO 40 K=1,NXO
      XO(K)=0.D0
      IF(K.LE.8) XO(K)=D(K)*FRAC(1,K)
 40   CONTINUE
 45   CONTINUE
C
      RETURN
C
      ENTRY LUNG2(A,RINDEX,CINDEX,IROW,ICOL,ICOL1,
     1  KOUNT,LAMR,P,K,J,N,NBRNCH,IBRNCH)
C
C  LUNG COMPARTMENTS
C    PATHWAYS A-H
C
      IF(K.EQ.1) GO TO 55
      DO 50 I=1,9
      AMNT(I) =LAMR(K)*FRAC(K,I)
 50   CONTINUE
      AMNT(10)=LAMR(K)*(1.D0-FRAC(K,9))
 55   CONTINUE
      J=J+1
      IROW=IROW+1
      IF(K.EQ.1) GO TO 70
      Q=P(K)
 60   CONTINUE
      IF (J.LE.2) N=2
      IF(J.EQ.3.OR.J.EQ.4) N=2
      IF(J.GE.5.AND.J.LE.9)  N=4
      DO 65 I=1,N
      IKOUNT=IKOUNT+1
      ICOL=ICOL+1
      A(IKOUNT)=AMNT(J)*P(K)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
 65   CONTINUE
      IF(NBRNCH(K).EQ.IBRNCH(K).AND.NBRNCH(K).GT.1)
     1  CALL CHANGE(P(K),K,ICOL,ICOL1,NCON,J,N, IOPT,&60)
      P(K)=Q
      II=2*N-J
      IF(J.LE.2)  II=N-J
      ICOL=ICOL+NCON(K-1)-II
      IF(NBRNCH(K).EQ.3.AND.IBRNCH(K).EQ.2)  ICOL=ICOL+NCON(K-2)
      GO TO 75
 70   CONTINUE
      ICOL=ICOL+1
 75   CONTINUE
      IKOUNT=IKOUNT+1
      A(IKOUNT)=-LAMX(K,J)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
      IF(J.EQ.8.AND.K.GT.1)  ICOL=ICOL1+J/N*N
      IF(J.EQ.8) GO TO 80
      IF(K.EQ.1) GO TO 55
      ICOL=ICOL1+J/N*N
      GO TO 55
 80   CONTINUE
C
C   PATHWAYS I AND J
C
      IROW=IROW+1
      J=J+1
      IF(K.EQ.1) GO TO 95
      Q=P(K)
 85   CONTINUE
      DO 90 I=1,2
      ICOL=ICOL+1
      IKOUNT=IKOUNT+1
      A(IKOUNT)=AMNT(J)*P(K)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
 90   CONTINUE
      IF(NBRNCH(K).EQ.IBRNCH(K).AND.NBRNCH(K).GT.1)
     1  CALL CHANGE(P(K),K,ICOL,ICOL1,NCON,J,N, IOPT,&85)
      P(K)=Q
      ICOL=ICOL+NCON(K-1)-2
      IF(NBRNCH(K).EQ.3.AND.IBRNCH(K).EQ.2)  ICOL=ICOL+NCON(K-2)
      GO TO 100
 95   CONTINUE
      ICOL=9
 100  CONTINUE
      IKOUNT=IKOUNT+1
      IF(J.EQ.9)  A(IKOUNT)=LAMX(K,8)*FRAC(K,9)
      IF(J.EQ.10)  A(IKOUNT)=LAMX(K,8)*(1.D0-FRAC(K,9))
      RINDEX(IKOUNT)=IROW                                               LOW 1850
      CINDEX(IKOUNT)=ICOL                                               LOW 1860
      II=1                                                              LOW 1870
      IF(J.EQ.10) II=2                                                  LOW 1880
      ICOL=ICOL+II                                                      LOW 1890
      IKOUNT=IKOUNT+1                                                   LOW 1900
      IF(J.EQ.9) A(IKOUNT)=-LAMX(K,J)                                   LOW 1910
      IF(J.EQ.10) A(IKOUNT)=0.D0                                        LOW 1920
      RINDEX(IKOUNT)=IROW                                               LOW 1930
      CINDEX(IKOUNT)=ICOL                                               LOW 1940
      IF(J.EQ.10.AND.K.GT.1) ICOL=ICOL1+J/2*2                           LOW 1950
      IF(J.EQ.10) GO TO 105                                             LOW 1960
      IF(K.EQ.1) GO TO 80                                               LOW 1970
      ICOL=ICOL1+J/2*2                                                  LOW 1980
      GO TO 80                                                          LOW 1990
  105 CONTINUE                                                          LOW 2000
C                                                                       LOW 2010
C    PATHWAYS K AND L                                                   LOW 2020
C                                                                       LOW 2030
      DO 125 I=1,2                                                      LOW 2040
      IROW=IROW+1                                                       LOW 2050
      J=J+1                                                             LOW 2060
      IF(K.EQ.1) GO TO 115                                              LOW 2070
      Q=P(K)                                                            LOW 2080
  110 CONTINUE                                                          LOW 2090
      ICOL=ICOL+1                                                       LOW 2100
      IKOUNT=IKOUNT+1                                                   LOW 2110
      A(IKOUNT)=LAMB(K)*P(K)                                            LOW 2120
      RINDEX(IKOUNT)=IROW                                               LOW 2130
      CINDEX(IKOUNT)=ICOL                                               LOW 2140
      IF(NBRNCH(K).EQ.IBRNCH(K).AND.NBRNCH(K).GT.1)                     LOW 2150
     1 CALL CHANGE(P(K),K,ICOL,ICOL1,NCON,J,N,IOPT,&110)                LOW 2151
      P(K)=Q                                                            LOW 2160
      ICOL=ICOL+NCON(K-1)-5                                             LOW 2170
      IF(NBRNCH(K).EQ.3.AND.IBRNCH(K).EQ.2) ICOL=ICOL+NCON(K-2)         LOW 2180
      GO TO 120                                                         LOW 2190
  115 CONTINUE                                                          LOW 2200
      ICOL=5+I                                                          LOW 2210
  120 CONTINUE                                                          LOW 2220
      IKOUNT=IKOUNT+1                                                   LOW 2230
      A(IKOUNT)=LAMX(K,5+I)                                             LOW 2240
      RINDEX(IKOUNT)=IROW                                               LOW 2250
      CINDEX(IKOUNT)=ICOL                                               LOW 2260
      IKOUNT=IKOUNT+1                                                   LOW 2270
      ICOL=ICOL+5                                                       LOW 2280
      A(IKOUNT)=-LAMX(K,4)                                              LOW 2290
      RINDEX(IKOUNT)=IROW                                               LOW 2300
      CINDEX(IKOUNT)=ICOL                                               LOW 2310
      IF(K.EQ.1) GO TO 125                                              LOW 2320
      ICOL=ICOL1+J                                                      LOW 2330
  125 CONTINUE                                                          LOW 2340
      RETURN                                                            LOW 2350
      END                                                               LOW 2360
      SUBROUTINE GI(A,RINDEX,CINDEX,IROW,ICOL,ICOL1,IKOUNT,LAMR,
     1  P,K,J,N,NBRNCH,IBRNCH,XO,NXO)
C
C  THIS SUBROUTINE DETERMINES THE ENTRIES OF THE COEFFICIENT
C  MATRIX CORRESPONDING TO THE COMPARTMENTS OF THE GI TRACT
C
C  GILAM(4)
C     GILAM(I) = LAMBDA FOR COMPARTMENT I OF GI TRACT
C
C  LAMAB(10,4)
C     LAMAB(I,1) = LAMBDA OF ABSORPTION IN S FOR DAUGHTER I
C     LAMAB(I,2) = LAMBDA OF ABSORPTION IN SI FOR DAUGHTER I
C     LAMAB(I,3) = LAMBDA OF ABSORPTION IN ULI FOR DAUGHTER I
C     LAMAB(I,4) = LAMBDA OF ABSORPTION IN LLI FOR DAUGHTER I
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 LAMX(10,10),LAMR(1),LAMAB(10,4)
      INTEGER*2 RINDEX(1),CINDEX(1)
      DIMENSION A(1),GILAM(4),P(1),NBRNCH(1),IBRNCH(1),XO(1)
C
      COMMON/TABLES/ LAMAB(10,4),LAMX(10,10),ORGNM(10),TAG(10),
     1  ICON(10,10),IOPT,NCON(10),NISO,NORG
C
      DATA GILAM/24.D0,6.D0,1.846153846D0,1.D0/
C
C
C  SET INITIAL VALUES FOR ODE
C
      IF(K.GT.1) GO TO 15
      IF(IOPT.EQ.0.OR.IOPT.EQ.2) GO TO 15
      DO 10 I=1,NXO
      XO(I)=0.D0
      IF(I.EQ.1) XO(I)=1.D0
 10   CONTINUE
C
 15   CONTINUE
C
C  GI TRACT
C   STOMACH
C
      IROW=IROW+1
      J=J+1
      IF(K.EQ.1) GO TO 25
      Q=P(K)
 20   CONTINUE
      ICOL=ICOL+1
      IKOUNT=IKOUNT+1
      A(IKOUNT)=LAMR(K)*P(K)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
      IF(NBRNCH(K).EQ.IBRNCH(K).AND.NBRNCH(K).GT.1)
     1  CALL CHANGE(P(K),K,ICOL,ICOL1,NCON,J,N,IOPT,&20)
      P(K)=Q
      IF(IOPT.EQ.0)  ICOL=ICOL+NCON(K-1)-1
      IF (IOPT.EQ.1)  ICOL=ICOL+NCON(K-1)
      IF(NBRNCH(K).EQ.3.AND.IBRNCH(K).EQ.2)  ICOL=ICOL+NCON(K-2)
      GO TO 30
 25   CONTINUE
      IF(IOPT.EQ.0)  ICOL=2
      IF (IOPT.EQ.1)  ICOL=1
 30   CONTINUE
      IF (IOPT.EQ.1) GO TO 45
      DO 35 I=2,4,2
      IKOUNT=IKOUNT+1
      A(IKOUNT)=LAMX(K,I)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
      ICOL=ICOL+I
 35   CONTINUE
      ICOL=ICOL+3
      DO 40 I=1,2
      IKOUNT=IKOUNT+1
      A(IKOUNT)=LAMX(K,4)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
      ICOL=ICOL+1
 40   CONTINUE
 45   CONTINUE
      IKOUNT=IKOUNT+1
      A(IKOUNT)=-GILAM(1)
      A(IKOUNT)=A(IKOUNT)-LAMAB(K,1)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
      IF (K.EQ.1) GO TO 50
      ICOL=ICOL1+J
 50   CONTINUE
C
C   SI,ULI,LII
C
      DO 70 I=1,3
      IROW=IROW+1
      J=J+1
      IF (K.EQ.1) GO TO 60
      Q=P(K)
 55   CONTINUE
      ICOL=ICOL+1
      IKOUNT=IKOUNT+1
      A(IKOUNT)=LAMB(K)
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
      A(IKOUNT)=LAMB(K)*P(K)
      IF(NBRNCH(K).EQ.IBRNCH(K).AND.NBRNCH(K).GT.1)
     1  CALL CHANGE(P(K),K,ICOL,ICOL1,NCON,J,N,IOPT,&55)
      P(K)=Q
      ICOL=ICOL+NCON(K-1)-1
      IF(NBRNCH(K).EQ.3.AND.IBRNCH(K).EQ.2) ICOL=ICOL+NCOM(K-2)         GI  1020
      GO TO 65                                                          GI  1030
   60 CONTINUE                                                          GI  1040
      ICOL=J-1                                                          GI  1050
   65 CONTINUE                                                          GI  1060
      IKOUNT=IKOUNT+1                                                   GI  1070
      A(IKOUNT)=GILAM(I)                                                GI  1080
      RINDEX(IKOUNT)=IROW                                               GI  1090
      CINDEX(IKOUNT)=ICOL                                               GI  1100
      ICOL=ICOL+1                                                       GI  1110
      IKOUNT=IKOUNT+1                                                   GI  1120
      A(IKOUNT)=-GILAM(I+1)                                             GI  1130
      A(IKOUNT)=A(IKOUNT)-LAMAB(K,I+1)                                  GI  1140
      RINDEX(IKOUNT)=IROW                                               GI  1150
      CINDEX(IKOUNT)=ICOL                                               GI  1160
      IF(K.EQ.1) GO TO 70                                               GI  1170
      ICOL=ICOL1+J                                                      GI  1180
   70 CONTINUE                                                          GI  1190
C                                                                       GI  1200
      RETURN                                                            GI  1210
      END                                                               GI  1220
```

```
      SUBROUTINE ORGAN(A,RINDEX,CINDEX,IROW,ICOL,ICOL1,IKOUNT,          ORGAN 10
     1   LAMR,P,K,J,N,NBRNCH,IBRNCH,ISORS,X0,NX0)                       ORGAN 11
C                                                                       ORGAN 20
C  THIS SUBROUTINE DETERMINES THE ENTRIES OF THE COEFFICIENT            ORGAN 30
C  MATRIX CORRESPONDING TO THE COMPARTMENTS OF OTHER ORGANS OF          ORGAN 40
C  THE BODY                                                             ORGAN 50
C                                                                       ORGAN 60
C                                                                       ORGAN 70
C  ICOM(10,10)                                                          ORGAN 80
C     ICOM(I,J) = NO. OF COMPARTMENT FOR DAUGHTER I,                    ORGAN 90
C                       ORGAN J                                         ORGA 100
C  AS(10,10),  TBS(10,10)                                               ORGA 110
C     AS(I,J) = COEF. OF EXPONENTIAL OF COMPARTMENT I, ORGAN J          ORGA 120
C     TBS(I,J) =POWER OF E, COMPARTMENT I ORGAN J                       ORGA 130
C                                                                       ORGA 140
C  IS(10)                                                               ORGA 150
C     IS(I) - FRACTION OF THE DAUGHTER RETURNING                        ORGA 160
C     TO BLOOD FOR ORGAN I                                              ORGA 170
C                                                                       ORGA 180
      IMPLICIT REAL*8 (A-H,O-Z)                                         ORGA 190
      REAL*8 LAMX(10,10),LAMR(1),LAMAB(10,4)                            ORGA 200
      INTEGER*2 RINDEX(1),CINDEX(1)                                     ORGA 210
      DIMENSION A(1),AS(10,10),TBS(10,10),IS(10),ISORS(1),P(1),         ORGA 220
     1   NBRNCH(1),IBRNCH(1),X0(1)                                      ORGA 221
C                                                                       ORGA 230
```

```
      COMMON/TABLES/ LAMAB(10,4),LAMI(10,10),ORGNM(10),TAG(10),
     1  ICON(10,10),IOPT,NCON(10),NISO,NORG
      COMMON/FAKE1/ SGM
C
C
C  OTHER ORGANS
C
      IO=0
      IF(NORG.EQ.0) GO TO 80
      IF(K.GT.1) READ(50,1001) (IS(I),I=1,NORG)
 1001 FORMAT(10I2)
      WRITE(10,1002)
 1002 FORMAT('0',/////,' ISOTOPE',4X,'ORGAN',5X,'A''S AND TB''S')
      WRITE(10,1003) TAG(K)
 1003 FORMAT(' ',A8)
      SNORM=0.D0
      DO 65 I=1,NORG
      READ(50,1004) ISORS(I+8),ORGNM(I),ICON(K,I)
 1004 FORMAT(I2,A8,I2)
      ICONKI=ICON(K,I)
      READ(50,1005) (AS(JJ,I),JJ=1,ICONKI),(TBS(JJ,I),JJ=1,ICONKI)
 1005 FORMAT(8F10.0)
      WRITE(10,1006) ORGNM(I),(AS(JJ,I),JJ=1,ICONKI),
     1   (TBS(JJ,I),JJ=1,ICONKI)
 1006 FORMAT(' ',10X,A8,2X,1P8E12.3,/,(' ',20X,8E12.3))
      SUM=0.D0
      DO 10 JJ=1,ICONKI
      SUM=SUM+AS(JJ,I)
 10   CONTINUE
      SNORM=SNORM+SUM
      DO 60 MM=1,ICONKI
      IROW=IROW+1
      J=J+1
      IF(K.EQ.1) GO TO 30
      Q=P(K)
      ISW=1
 15   CONTINUE
      ISW=ISW+1
      DO 25 JJ=1,NORG
      ICONKM=ICON(K-1,JJ)
      IF(IBRNCH(K).GT.1.AND.ISW.EQ.2) ICONKM=ICON(K-2,JJ)
      DO 25 KK=1,ICONKM
      CALL FAKE
      ICOL=ICOL+1
      IKOUNT=IKOUNT+1
      A(IKOUNT)=IS(JJ)*LAMB(K)*AS(MM,I)*P(K)
      IF(IS(JJ).EQ.1) GO TO 20
      IF(JJ.EQ.I) A(IKOUNT)=A(IKOUNT)+(1.D0-IS(JJ))*
     1  LAMB(K)*AS(MM,I)/SUM*P(K)
 20   CONTINUE
      RINDEX(IKOUNT)=IROW
      CINDEX(IKOUNT)=ICOL
 25   CONTINUE
      IF(NBRNCH(K).EQ.IBRNCH(K).AND.NBRNCH(K).GT.1)
     1   CALL CHANGE(P(K),K,ICOL,ICOL1,NCON,J,N,IOPT,&15)
      P(K)=Q
      IF(NBRNCH(K).EQ.3.AND.IBRNCH(K).EQ.2) ICOL=ICOL+NCON(K-1)        ORGA 760
      GO TO 35                                                         ORGA 770
 30   CONTINUE                                                         ORGA 780
      ICOL=0                                                           ORGA 790
 35   CONTINUE                                                         ORGA 800
      ICOL=ICCL+1                                                      ORGA 810
      IF(IOPT.EQ.1) GO TO 45                                           ORGA 820
      IF(IOPT.EQ.2) GO TO 55                                           ORGA 830
      DO 40 JJ=1,5,2                                                   ORGA 840
      IKOUNT=IKOUNT+1                                                  ORGA 850
      A(IKOUNT)=LAMX(K,JJ)*AS(NN,I)                                    ORGA 860
      RINDEX(IKOUNT)=IROW                                              ORGA 870
      CINDEX(IKOUNT)=ICOL                                              ORGA 880
      ICOL=ICOL+2                                                      ORGA 890
 40   CONTINUE                                                         ORGA 900
      ICOL=ICCL+2                                                      ORGA 910
      IKOUNT=IKOUNT+1                                                  ORGA 920
      A(IKOUNT)=LAMX(K,9)*AS(NN,I)                                     ORGA 930
      RINDEX(IKOUNT)=IROW                                              ORGA 940
      CINDEX(IKOUNT)=ICOL                                              ORGA 950
      ICOL=ICOL+4                                                      ORGA 960
 45   CONTINUE                                                         ORGA 970
      DO 50 JJ=1,4                                                     ORGA 980
      IKOUNT=IKOUNT+1                                                  ORGA 990
      A(IKOUNT)=LAMAB(K,JJ)*AS(NN,I)                                   ORG 1000
      RINDEX(IKOUNT)=IROW                                              ORG 1010
      CINDEX(IKOUNT)=ICOL                                              ORG 1020
      ICOL=ICOL+1                                                      ORG 1030
 50   CONTINUE                                                         ORG 1040
 55   CONTINUE                                                         ORG 1050
      IO=IO+1                                                          ORG 1060
      ICOL=ICOL-1+IO                                                   ORG 1070
      IKOUNT=IKOUNT+1                                                  ORG 1080
      A(IKOUNT)=-TBS(NN,I)                                             ORG 1090
      RINDEX(IKOUNT)=IROW                                              ORG 1100
      CINDEX(IKOUNT)=ICOL                                              ORG 1110
      IF(K.EQ.1) GO TO 60                                              ORG 1120
      IF(IOPT.EQ.0) ICOL=ICOL1+16                                      ORG 1130
      IF(IOPT.EQ.1) ICOL=ICOL1+4                                       ORG 1140
      IF(IOPT.EQ.2) ICOL=ICOL1                                         ORG 1150
 60   CONTINUE                                                         ORG 1160
 65   CONTINUE                                                         ORG 1170
C                                                                      ORG 1180
C  SET INITIAL VALUES FOR ODE                                          ORG 1190
C                                                                      ORG 1200
      IF(K.GT.1) GO TO 80                                              ORG 1210
      IF(IOPT.EQ.0.OR.IOPT.EQ.1) GO TO 80                              ORG 1220
      I=0                                                              ORG 1230
      DO 70 INO=1,NORG                                                 ORG 1240
      ICONKI=ICON(K,INO)                                               ORG 1250
      DO 70 ICN=1,ICONKI                                               ORG 1260
      I=I+1                                                            ORG 1270
      Y0(I)=AS(ICN,INO)/SNORM                                          ORG 1280
 70   CONTINUE                                                         ORG 1290
      IOP1=IO+1                                                        ORG 1300
      DO 75 I=IOP1,NXO                                                 ORG 1310
      XO(I)=0.D0                                                 ORG 1320
   75 CONTINUE                                                   ORG 1330
   80 CONTINUE                                                   ORG 1340
      IF (IOPT.EQ.0)  NCOM(K)=IO+16                              ORG 1350
      IF(IOPT.EQ.1)  NCOM(K)=IO+4                                ORG 1360
      IF(IOPT.EQ.2)  NCOM(K)=IO                                  ORG 1370
      RETURN                                                     ORG 1380
      END                                                        ORG 1390
```

```
      SUBROUTINE YNA(Y,N)                                        YNA   10
      IMPLICIT REAL*8 (A-H,O-Z)                                  YNA   20
C                                                                YNA   30
C  THIS SUBROUTINE PRINTS MICROCURIE-DAYS FOR EACH               YNA   40
C    TIME INTERVAL                                               YNA   50
C                                                                YNA   60
C  NORG - NUMBER OF ORGANS (EXCLUDING LUNG AND GI)               YNA   70
C  ICOM(I,J) - NO. OF COMPARTMENTS FOR DAUGHTER I, ORGAN J       YNA   80
C  NCOM(I) - NO. OF COMPARTMENTS FOR DAUGHTER I                  YNA   90
C  NISO - NUMBER OF ELEMENTS IN THE CHAIN                        YNA  100
C  LAMX(I,J) - LN2/(HALFTIME OF DAUGHTER I IN COMPARTMENT J)     YNA  110
C  ORGNM(I) - NAME OF ORGAN I                                    YNA  120
C  TAG(I) - ATOMIC SYMBOL AND ATOMIC NUMBER FOR                  YNA  130
C    DAUGHTER I                                                  YNA  140
C                                                                YNA  150
      REAL*8 TAG,LAMX,X(10,21),ORGNM,LAMAB                       YNA  160
      COMMON/TABLES/ LAMAB(10,4),LAMX(10,10),ORGNM(10),TAG(10),  YNA  170
     1  ICOM(10,10),IOPT,NCOM(10),NISO,NORG                      YNA  171
      COMMON/PRINT/TPRL(10),DTPR(10),TPRINT                      YNA  180
      DIMENSION Y(1)                                             YNA  190
C                                                                YNA  200
C  PRINT SOLUTION VECTOR Y                                       YNA  210
C                                                                YNA  220
      PRINT 1001,TPRINT,(Y(I),I=1,N)                             YNA  230
 1001 FORMAT('0T='1PE16.9,3X'Y FOLLOWS'/(1P6E19.10))             YNA  240
      K=0                                                        YNA  250
C                                                                YNA  260
C  X(I,1) - NP REGION FOR DAUGHTER I                             YNA  270
C  X(I,2) - TB REGION FOR DAUGHTER I                             YNA  280
C  X(I,3) - P REGION FOR DAUGHTER I                              YNA  290
C  X(I,4) - LYMPH FOR DAUGHTER I                                 YNA  300
C  X(I,5) - STOMACH FOR DAUGHTER I                               YNA  310
C  X(I,6) - SI FOR DAUGHTER I                                    YNA  320
C  X(I,7) - ULI FOR DAUGHTER I                                   YNA  330
C  X(I,8) - LLI FOR DAUGHTER I                                   YNA  340
C  X(I,J) - COMPARTMENT J (J=9,10,...18) FOR DAUGHTER I          YNA  350
C  X(I,19) - MICROCURIES ENTERING BLOOD FROM LUNGS FOR DAUGHTER I YNA  360
C  X(I,20) - MICROCURIES ENTERING GI FROM LUNGS FOR DAUGHTER I   YNA  370
```

```
C  X(I,21) - MICROCURIES ENTERING BLOOD FROM GI FOR DAUGHTER I
C
      DO 45 I=1,NISO
      IF(IOPT.EQ.1.OR.IOPT.EQ.2) GO TO 10
      X(I,1)=Y(1+K)+Y(2+K)
      X(I,2)=Y(3+K)+Y(4+K)+Y(11+K)+Y(12+K)
      X(I,3)=Y(5+K)+Y(6+K)+Y(7+K)+Y(8+K)
      X(I,4)=Y(9+K)+Y(10+K)
      GO TO 15
 10   CONTINUE
      X(I,1)=0.
      X(I,2)=0.
      X(I,3)=0.
      X(I,4)=0.
 15   CONTINUE
      IF(IOPT.EQ.0) IORG=12
      IF(IOPT.EQ.1) IORG=0
      IF(IOPT.EQ.2) GO TO 25
      DO 20 J=1,4
      X(I,J+4)=Y(IORG+K+J)
 20   CONTINUE
      GO TO 30
 25   CONTINUE
      X(I,5)=0.
      X(I,6)=0.
      X(I,7)=0.
      X(I,8)=0.
 30   CONTINUE
      IF(NORG.EQ.0) GO TO 40
      ICONTJ=0
      DO 35 J=1,NORG
      II=ICONTJ+1
      ICONTJ=ICON(I,J)+ICONTJ
      X(I,J+8)=0.D0
      IF(IOPT.EQ.0) IORG=16
      IF(IOPT.EQ.1) IORG=4
      IF(IOPT.EQ.2) IORG=0
      DO 35 M=II,ICONTJ
      X(I,J+8)=X(I,J+8)+Y(IORG+K+M)
 35   CONTINUE
 40   CONTINUE
      IF(IOPT.EQ.0) X(I,19)=Y(1+K)*LAMX(I,1)+Y(3+K)*LAMX(I,3)
     1   +Y(5+K)*LAMX(I,5)+Y(9+K)*LAMX(I,9)
      IF(IOPT.EQ.0) X(I,20)=Y(2+K)*LAMX(I,2)+Y(4+K)*LAMX(I,4)
     1   +Y(11+K)*LAMX(I,4)+Y(12+K)*LAMX(I,4)
      IF(IOPT.EQ.0.OR.IOPT.EQ.1) X(I,21)=X(I,5)*LAMAB(I,1)
     1   +X(I,6)*LAMAB(I,2)+X(I,7)*LAMAB(I,3)+X(I,8)*LAMAB(I,4)
      K=K+NCOM(I)
 45   CONTINUE
      WRITE(10,1002) TPRINT
 1002 FORMAT('1T = ',1PE10.4)
      IF(IOPT.EQ.0) WRITE(10,1003) TAG(1)
 1003 FORMAT('OMICROCURIES-DAYS FROM INHALATION OF ',A8)
      IF(IOPT.EQ.1) WRITE(10,1004) TAG(1)
 1004 FORMAT('OMICROCURIE-DAYS FROM INGESTION OF ',A8)
      IF(IOPT.EQ.2) WRITE(10,1005) TAG(1)
```

```
1005 FORMAT('0MICROCURIE-DAYS FROM INJECTION OF ',A8)                    YMA   910
     IF(IOPT.EQ.0) WRITE(10,1006)                                        YMA   920
1006 FORMAT('0LUNGS',59X,'MICROCURIES ENTERING'/                         YMA   930
    1 ' ',67X,'BLOOD',7X,'GI'/' ISOTOPE',6X,'N-P',9X,'T-B',10X,          YMA   931
    2 'P',9X,'LYMPH')                                                    YMA   932
     DO 50 I=1,NISO                                                      YMA   940
     IF(IOPT.EQ.0) WRITE(10,1007) TAG(I),(X(I,J),J=1,4),X(I,19),X(I,20)YMA   950
50   CONTINUE                                                            YMA   960
1007 FORMAT(' ',A8,4(2X,1PE10.4),10X,E10.4,2X,E10.4)                     YMA   970
     IF(IOPT.EQ.0.OR.IOPT.EQ.1) WRITE(10,1008)                           YMA   980
1008 FORMAT('0'//'0GI TRACT',56X,'MICROCURIES ENTERING'/' ',67X,         YMA   990
    1 'BLOOD'/' ISOTOPE',4X,'STOMACH',7X,'S.I.',7X,'U.L.I.',6X,          YMA   991
    2 'L.L.I.')                                                          YMA   992
     DO 55 I=1,NISO                                                      YMA  1000
     IF(IOPT.EQ.0.OR.IOPT.EQ.1) WRITE(10,1007) TAG(I),                   YMA  1010
    1   (X(I,J),J=5,8),X(I,21)                                           YMA  1011
55   CONTINUE                                                            YMA  1020
     NORGP8=NORG+8                                                       YMA  1030
     IF(NORG.EQ.0) GO TO 65                                              YMA  1040
     WRITE(10,1009) (ORGNM(I),I=1,NORG)                                  YMA  1050
1009 FORMAT('0'//'0OTHER ORGANS'/'0ISOTOPE',5X,10(A8,4X))                YMA  1060
     DO 60 I=1,NISO                                                      YMA  1070
     WRITE(10,1010) TAG(I),(X(I,J),J=9,NORGP8)                           YMA  1080
1010 FORMAT(' ',A8,10(1X,1PE11.4))                                       YMA  1090
60   CONTINUE                                                            YMA  1100
     CALL OTHER(X)                                                       YMA  1110
65   CONTINUE                                                            YMA  1120
     WRITE(16) IPRINT                                                    YMA  1130
     IF(IOPT.EQ.0) IORG=1                                                YMA  1140
     IF(IOPT.EQ.1) IORG=1                                                YMA  1150
     IF(IOPT.EQ.2) IORG=1                                                YMA  1160
     DO 70 I=1,NISO                                                      YMA  1170
     WRITE(16) (X(I,J),J=IORG,NORGP8)                                    YMA  1180
     IF(IOPT.EQ.0) WRITE(16) X(I,19),X(I,20),X(I,21)                     YMA  1190
     WRITE(7,1011) (X(I,J),J=IORG,NORGP8)                                YMA  1200
1011 FORMAT(1P8E10.3)                                                    YMA  1210
70   CONTINUE                                                            YMA  1220
     RETURN                                                              YMA  1230
     END                                                                 YMA  1240
```

```
      SUBROUTINE OTHER(X)                                                OTHER 10
C                                                                        OTHER 20
C  PERFORMS OTHER TISSUES TRANSFORMATION                                 OTHER 30
C                                                                        OTHER 40
      REAL*8 TAG,LAMX,X(10,1),ORGNM,TBOD/'T. BODY '/,LAMAB               OTHER 50
      COMMON/TABLES/ LAMAB(10,4),LAMX(10,10),ORGNM(10),TAG(10),          OTHER 60
     1  ICON(10,10),IOPT,NCON(10),NISO,NORG                              OTHER 61
```

```
      REAL TMASS(25)/200.,250.,400.,200.,135.,310.,1800.,85.,0.,          OTHER 70
     1 1000.,0.,28000.,31.,100.,4000.,1000.,1500.,1500.,2600.,180.,35.,OTHER 71
     2 20.,70000.,0.,14./                                                 OTHER 72
      DIMENSION IOTHER(10),OTHTIS(10)                                     OTHER 80
      NORGM1=NORG-1                                                       OTHER 90
      READ(50,1001) NOTOP,(IOTHER(I),I=1,NORGM1)                          OTHE 100
 1001 FORMAT(40I2)                                                        OTHE 110
      IF(NOTOP.EQ.0) RETURN                                               OTHE 120
      SMASS=0.                                                            OTHE 130
      DO 10 I=1,NORGM1                                                    OTHE 140
      SMASS=SMASS+TMASS(IOTHER(I))                                        OTHE 150
 10   CONTINUE                                                            OTHE 160
      DO 15 I=1,NISO                                                      OTHE 170
      OTHTIS(I)=X(I,8+NORG)                                               OTHE 180
      X(I,8+NORG)=OTHTIS(I)/(TMASS(23)-SMASS)*TMASS(23)                   OTHE 190
      DO 15 J=1,NORGM1                                                    OTHE 200
      X(I,8+J)=X(I,8+J)-TMASS(IOTHER(J))*OTHTIS(I)/                       OTHE 210
     1 (TMASS(23)-SMASS)                                                  OTHE 211
 15   CONTINUE                                                            OTHE 220
      WRITE(10,1002)                                                      OTHE 230
 1002 FORMAT('0AFTER TRANSFORMATION')                                     OTHE 240
      WRITE(10,1003) (ORGNM(I),I=1,NORGM1),TMOD                           OTHE 250
 1003 FORMAT(' ',12X,10(A8,4X))                                           OTHE 260
      NORGP8=NORG+8                                                       OTHE 270
      DO 20 I=1,NISO                                                      OTHE 280
      WRITE(10,1004) TAG(I),(X(I,J),J=9,NORGP8)                           OTHE 290
 1004 FORMAT(' ',A8,10(1X,1PE11.4))                                       OTHE 300
 20   CONTINUE                                                            OTHE 310
      RETURN                                                              OTHE 320
      END                                                                 OTHE 330
```

```
      SUBROUTINE INDEX(ICOL1,NCOM,K,IN,II)                                INDEX 10
C                                                                         INDEX 20
C  DETERMINES FIRST COLUMN FOR THE DAUGHTER                               INDEX 30
C                                                                         INDEX 40
      DIMENSION NCOM(1)                                                   INDEX 50
      IF(IN.GE.2) GO TO 10                                                INDEX 60
      ICOL1=ICOL1+NCOM(K-1)                                               INDEX 70
      RETURN                                                              INDEX 80
 10   CONTINUE                                                            INDEX 90
      IF(IN.EQ.II) GO TO 15                                               INDE 100
      IF(II.EQ.1) RETURN                                                  INDE 110
      ICOL1=ICOL1+NCOM(K-2)                                               INDE 120
      RETURN                                                              INDE 130
 15   CONTINUE                                                            INDE 140
      ICOL1=ICOL1+NCOM(K-1)+NCOM(K-2)                                     INDE 150
      RETURN                                                              INDE 160
      END                                                               INDX 170
```

```
      SUBROUTINE CHANGE(P,K,ICOL,ICOL1,NCON,J,N,IOPT,*)                 CHANG 10
C                                                                       CHANG 20
C                                                                       CHANG 30
C  SETS UP THE BRANCHING CASES (2,2) AND (3,3)                          CHANG 40
C                                                                       CHANG 50
      DIMENSION NCON(1)                                                 CHANG 60
      COMMON/SWTCH/ ISWTCH                                              CHANG 70
      ISWTCH=ISWTCH+1                                                   CHANG 80
      P=1.                                                              CHANG 90
      IF(ISWTCH/2*2.EQ.ISWTCH) RETURN                                   CHAN 100
      IF(IOPT.EQ.1) GO TO 10                                            CHAN 110
      IF(IOPT.EQ.2) GO TO 15                                            CHAN 120
      IF(J.LE.8) ICOL=ICOL1+NCON(K-2)*(J-1)/N*N                         CHAN 130
      IF(J.EQ.9.OR.J.EQ.10) ICOL=ICOL1+NCON(K-2)*(J-1)/2*2              CHAN 140
      IF(J.GE.11.AND.J.LE.16) ICOL=ICOL1+NCON(K-2)*(J-1)                CHAN 150
      IF(J.GT.16) ICOL=ICOL1+16*NCON(K-2)                               CHAN 160
      RETURN 1                                                          CHAN 170
 10   CONTINUE                                                          CHAN 180
      IF(J.GE.1.AND.J.LE.4) ICOL=ICOL1+NCON(K-2)*(J-1)                  CHAN 190
      IF(J.GT.4) ICOL=ICOL1+4*NCON(K-2)                                 CHAN 200
      RETURN 1                                                          CHAN 210
 15   CONTINUE                                                          CHAN 220
      ICOL=ICOL1+NCON(K-2)                                              CHAN 230
      RETURN 1                                                          CHAN 240
      END                                                               CHAN 250
```

```
      SUBROUTINE FAKE                                                   FAKE  10
C                                                                       FAKE  20
C  CALLED BY SUBROUTINE ORGAN TO AVOID A ZERO DIVIDE GENERATED          FAKE  30
C  BY THE OPT=2 VERSION OF THE H-LEVEL COMPILER                         FAKE  40
C                                                                       FAKE  50
      COMMON/FAKE1/ SUM                                                 FAKE  60
      REAL*8 SUM                                                        FAKE  70
      RETURN                                                            FAKE  80
      END                                                               FAKE  90
```

```
AXEB     TITLE 'SOLVE AX=B WHERE A IS A COMPACTED LOWER TRI.MATRIX'
AXEB     START 0
* CALL AXEB(A,X,B,INDXA,N)
* A IS AN N BY N LOWER TRIANGULAR MATRIX. THE VECTOR A CONSISTS
* OF THE NONZERO ENTRIES IN A STORED SEQUENTIALLY BY ROWS.
* SOLVE LINEAR SYSTEM OF EQUATIONS.
***** IT IS ASSUMED THAT THE MAX. NO. OF NONZERO ENTRIES IN A ROW OF A
***** IS .LE. 52.   MOREOVER, THE DIAG. ENTRIES ARE NONZERO.
***** AT MOST 51 SUMMANDS IN SUML
A        EQU   1
X        EQU   2
B        EQU   3
INDXA    EQU   4
IND14    EQU   5
IINC     EQU   6
I        EQU   8
J        EQU   9
BADDR    EQU   10
         USING AXEB,15
         STM   14,12,12(13)
         LM    1,5,0(1)
         LA    IINC,8
         L     7,0(5)          N
         LA    0,16
         SLA   7,3             8N
         SR    7,IINC          8(N-1)
         SR    I,I
         LR    5,4
         LA    BADDR,POINT+14
         SR    A,IINC          A=ADD(A(K-1,K-1),K=1,....,N
LOOP     SDR   0,0             FORM SUML=SUM (J=1,....,I-1) A(I,J)*X(J)
*                              FOR I=1,....,N.
         L     11,4(IND14)
         B     0(BADDR,11)
**
**
         L     J,400(INDXA)    AT LEAST 51
         LD    6,408(A)
         MD    6,0(X,J)
         SDR   0,6
         L     J,392(INDXA)    AT LEAST 50
         LD    4,400(A)
         MD    4,0(X,J)
         SDR   0,4
         L     J,384(INDXA)    AT LEAST 49
         LD    2,392(A)
         MD    2,0(X,J)
```

```
SDR     0,2                                                  AIEB 480
L       J,376(INDIA)      AT LEAST 48                        AIEB 490
LD      6,384(A)                                             AIEB 500
MD      6,0(I,J)                                             AIEB 510
SDR     0,6                                                  AIEB 520
L       J,368(INDIA)      AT LEAST 47                        AIEB 530
LD      4,376(A)                                             AIEB 540
MD      4,0(I,J)                                             AIEB 550
SDR     0,4                                                  AIEB 560
L       J,360(INDIA)      AT LEAST 46                        AIEB 570
LD      2,368(A)                                             AIEB 580
MD      2,0(I,J)                                             AIEB 590
SDR     0,2                                                  AIEB 600
L       J,352(INDIA)      AT LEAST 45                        AIEB 610
LD      6,360(A)                                             AIEB 620
MD      6,0(I,J)                                             AIEB 630
SDR     0,6                                                  AIEB 640
L       J,344(INDIA)      AT LEAST 44                        AIEB 650
LD      4,352(A)                                             AIEB 660
MD      4,0(I,J)                                             AIEB 670
SDR     0,4                                                  AIEB 680
L       J,336(INDIA)      AT LEAST 43                        AIEB 690
LD      2,344(A)                                             AIEB 700
MD      2,0(I,J)                                             AIEB 710
SDR     0,2                                                  AIEB 720
L       J,328(INDIA)      AT LEAST 42                        AIEB 730
LD      6,336(A)                                             AIEB 740
MD      6,0(I,J)                                             AIEB 750
SDR     0,6                                                  AIEB 760
L       J,320(INDIA)      AT LEAST 41                        AIEB 770
LD      4,328(A)                                             AIEB 780
MD      4,0(I,J)                                             AIEB 790
SDR     0,4                                                  AIEB 800
L       J,312(INDIA)      AT LEAST 40                        AIEB 810
LD      2,320(A)                                             AIEB 820
MD      2,0(I,J)                                             AIEB 830
SDR     0,2                                                  AIEB 840
L       J,304(INDIA)      AT LEAST 39                        AIEB 850
LD      6,312(A)                                             AIEB 860
MD      6,0(I,J)                                             AIEB 870
SDR     0,6                                                  AIEB 880
L       J,296(INDIA)      AT LEAST 38                        AIEB 890
LD      4,304(A)                                             AIEB 900
MD      4,0(I,J)                                             AIEB 910
SDR     0,4                                                  AIEB 920
L       J,288(INDIA)      AT LEAST 37                        AIEB 930
LD      2,296(A)                                             AIEB 940
MD      2,0(I,J)                                             AIEB 950
SDR     0,2                                                  AIEB 960
L       J,280(INDIA)      AT LEAST 36                        AIEB 970
LD      6,288(A)                                             AIEB 980
MD      6,0(I,J)                                             AIEB 990
SDR     0,6                                                  AIE 1000
L       J,272(INDIA)      AT LEAST 35                        AIE 1010
LD      4,280(A)                                             AIE 1020
MD      4,0(I,J)                                             AIE 1030
```

```
SDR     0,4                                                                                AXE 1040
L       J,254(INDXA)      AT LEAST 34                                                      AXE 1050
LD      2,272(A)                                                                           AXE 1060
MD      2,0(X,J)                                                                           AXE 1070
SDR     0,2                                                                                AXE 1080
L       J,256(INDXA)      AT LEAST 33                                                      AXE 1090
LD      6,264(A)                                                                           AXE 1100
MD      6,0(X,J)                                                                           AXE 1110
SDR     0,6                                                                                AXE 1120
L       J,248(INDXA)      AT LEAST 32                                                      AXE 1130
LD      4,256(A)                                                                           AXE 1140
MD      4,0(X,J)                                                                           AXE 1150
SDR     0,4                                                                                AXE 1160
L       J,240(INDXA)      AT LEAST 31                                                      AXE 1170
LD      2,248(A)                                                                           AXE 1180
MD      2,0(X,J)                                                                           AXE 1190
SDR     0,2                                                                                AXE 1200
L       J,232(INDXA)      AT LEAST 30                                                      AXE 1210
LD      6,240(A)                                                                           AXE 1220
MD      6,0(X,J)                                                                           AXE 1230
SDR     0,6                                                                                AXE 1240
L       J,224(INDXA;      AT LEAST 29                                                      AXE 1250
LD      4,232(A)                                                                           AXE 1260
MD      4,0(X,J)                                                                           AXE 1270
SDR     0,4                                                                                AXE 1280
L       J,216(INDXA)      AT LEAST 28                                                      AXE 1290
LD      2,224(A)                                                                           AXE 1300
MD      2,0(X,J)                                                                           AXE 1310
SDR     0,2                                                                                AXE 1320
L       J,208(INDXA)      AT LEAST 27                                                      AXE 1330
LD      6,216(A)                                                                           AXE 1340
MD      6,0(X,J)                                                                           AXE 1350
SDR     0,6                                                                                AXE 1360
L       J,200(INDXA)      AT LEAST 26                                                      AXE 1370
LD      4,208(A)                                                                           AXE 1380
MD      4,0(X,J)                                                                           AXE 1390
SDR     0,4                                                                                AXE 1400
L       J,192(INDXA)      AT LEAST 25                                                      AXE 1410
LD      2,200(A)                                                                           AXE 1420
MD      2,0(X,J)                                                                           AXE 1430
SDR     0,2                                                                                AXE 1440
L       J,184(INDXA)      AT LEAST 24                                                      AXE 1450
LD      6,192(A)                                                                           AXE 1460
MD      6,0(X,J)                                                                           AXE 1470
SDR     0,6                                                                                AXE 1480
L       J,176(INDXA)      AT LEAST 23                                                      AXE 1490
LD      4,184(A)                                                                           AXE 1500
MD      4,0(X,J)                                                                           AXE 1510
SDR     0,4                                                                                AXE 1520
L       J,168(INDXA)      AT LEAST 22                                                      AXE 1530
LD      2,176(A)                                                                           AXE 1540
MD      2,0(X,J)                                                                           AXE 1550
SDR     0,2                                                                                AXE 1560
L       J,160(INDXA)      AT LEAST 21                                                      AXE 1570
LD      6,168(A)                                                                           AXE 1580
MD      6,0(X,J)                                                                           AXE 1590
```

```
SDR      0,6                                                               AXE 1600
L        J,152(INDXA)     AT LEAST 20                                      AXE 1610
LD       4,160(A)                                                          AXE 1620
MD       4,0(X,J)                                                          AXE 1630
SDR      0,4                                                               AXE 1640
L        J,144(INDXA)     AT LEAST 19                                      AXE 1650
LD       2,152(A)                                                          AXE 1660
MD       2,0(X,J)                                                          AXE 1670
SDR      0,2                                                               AXE 1680
L        J,136(INDXA)     AT LEAST 18                                      AXE 1690
LD       6,144(A)                                                          AXE 1700
MD       6,0(X,J)                                                          AXE 1710
SDR      0,6                                                               AXE 1720
L        J,128(INDXA)     AT LEAST 17 NONZERO OFF-DIAG. ENTRIES            AXE 1730
LD       4,136(A)                                                          AXE 1740
MD       4,0(X,J)                                                          AXE 1750
SDR      0,4                                                               AXE 1760
L        J,120(INDXA)     AT LEAST 16 NONZERO OFF-DIAG. ENTRIES            AXE 1770
LD       2,128(A)                                                          AXE 1780
MD       2,0(X,J)                                                          AXE 1790
SDR      0,2                                                               AXE 1800
L        J,112(INDXA)     AT LEAST 15 NONZERO OFF-DIAG. ENTRIES            AXE 1810
LD       6,120(A)                                                          AXE 1820
MD       6,0(X,J)                                                          AXE 1830
SDR      0,6                                                               AXE 1840
L        J,104(INDXA)     AT LEAST 14 NONZERO OFF-DIAG. ENTRIES            AXE 1850
LD       4,112(A)                                                          AXE 1860
MD       4,0(X,J)                                                          AXE 1870
SDR      0,4                                                               AXE 1880
L        J,096(INDXA)     AT LEAST 13 NONZERO OFF-DIAG. ENTRIES            AXE 1890
LD       2,104(A)                                                          AXE 1900
MD       2,0(X,J)                                                          AXE 1910
SDR      0,2                                                               AXE 1920
L        J,088(INDXA)     AT LEAST 12 NONZERO OFF-DIAG. ENTRIES            AXE 1930
LD       6,096(A)                                                          AXE 1940
MD       6,0(X,J)                                                          AXE 1950
SDR      0,6                                                               AXE 1960
L        J,080(INDXA)     AT LEAST 11 NONZERO OFF-DIAG. ENTRIES            AXE 1970
LD       4,088(A)                                                          AXE 1980
MD       4,0(X,J)                                                          AXE 1990
SDR      0,4                                                               AXE 2000
L        J,072(INDXA)     AT LEAST 10 NONZERO OFF-DIAG. ENTRIES            AXE 2010
LD       2,080(A)                                                          AXE 2020
MD       2,0(X,J)                                                          AXE 2030
SDR      0,2                                                               AXE 2040
L        J,64(INDXA)                                                       AXE 2050
LD       6,72(A)                                                           AXE 2060
MD       6,0(X,J)                                                          AXE 2070
SDR      0,6                                                               AXE 2080
L        J,56(INDXA)                                                       AXE 2090
LD       4,64(A)                                                           AXE 2100
MD       4,0(X,J)                                                          AXE 2110
SDR      0,4                                                               AXE 2120
L        J,48(INDXA)                                                       AXE 2130
LD       2,56(A)                                                           AXE 2140
MD       2,0(X,J)                                                          AXE 2150
```

```
        SDR    0,2                                                              AXE 2160
        L      J,40(INDXA)                                                      AXE 2170
        LD     6,48(A)                                                          AXE 2180
        MD     6,0(X,J)                                                         AXE 2190
        SDR    0,6                                                              AXE 2200
        L      J,32(INDXA)                                                      AXE 2210
        LD     4,40(A)                                                          AXE 2220
        MD     4,0(X,J)                                                         AXE 2230
        SDR    0,4                                                              AXE 2240
        L      J,24(INDXA)                                                      AXE 2250
        LD     2,32(A)                                                          AXE 2260
        MD     2,0(X,J)                                                         AXE 2270
        SDR    0,2                                                              AXE 2280
        L      J,16(INDXA)                                                      AXE 2290
        LD     6,24(A)                                                          AXE 2300
        MD     6,0(X,J)                                                         AXE 2310
        SDR    0,6                                                              AXE 2320
        L      J,8(INDXA)                                                       AXE 2330
        LD     4,16(A)                                                          AXE 2340
        MD     4,0(X,J)                                                         AXE 2350
        SDR    0,4                                                              AXE 2360
        L      J,0(INDXA)                                                       AXE 2370
        LD     2,8(A)                                                           AXE 2380
        MD     2,0(X,J)                                                         AXE 2390
        SDR    0,2                                                              AXE 2400
**                                                                              AXE 2410
POINT   L      11,12(IND14)     8*(NO.NONZERO ENTRIES IN ROW I OF A)            AXE 2420
        AR     A,11                                                             AXE 2430
        AR     INDXA,11                                                         AXE 2440
        AR     IND14,0                                                          AXE 2450
        AD     0,0(B,I)         COULD CHECK FOR LOSS OF LEADING FIGURES.        AXE 2460
        DD     0,0(A)           A(I,I)                                          AXE 2470
        STD    0,0(X,I)                                                         AXE 2480
        BXLE   I,IINC,LOOP                                                      AXE 2490
        LM     14,12,12(13)                                                     AXE 2500
        BR     14                                                               AXE 2510
        END                                                                     AXE 2520
```

```
YPAY     TITLE 'FORM YDOT = A*Y + X0'                                           YPAY   10
YPAY     START 0                                                                YPAY   20
* CALL YPAY(YDOT,A,Y,X0,INDXA,N)                                                YPAY   30
* FORM YDOT=A*Y+X0                                                              YPAY   40
* WHERE A IS AN N BY N MATRIX. HOWEVER, ONLY THE NONZERO ENTRIES IN A           YPAY   50
* ARE STORED IN THE VECTOR, A. STORAGE IN A IS BY ROWS                          YPAY   60
* VIEW INDXA AS AN INTEGER*4 VECTOR.                                            YPAY   70
* THE NONZERO ENTRY A(I) BELONGS IN COLUMN (INDXA(2I-1)+8)/8                    YPAY   80
* INDXA(4I-2) = -14*(NO. NONZERO ENTRIES IN ROW I OF MATRIX A).                 YPAY   90
* INDXA(4I)     =    8*(.                  ''                   ).          YPAY 100
* Y AND X0 ARE FULL VECTORS WITH N ELEMENTS                                 YPAY 110
* A ZERO ROW IN A IS OKAY.                                                  YPAY 120
*****                                                                       YPAY 130
* SET FOR AT MOST 52 NONZERO ENTRIES IN A ROW OF THE MATRIX A.              YPAY 140
** IT IS ASSUMED THAT THE MAXIMUM NO. OF NONZERO ENTRIES IN A ROW OF        YPAY 150
** A IS .LE. 52.                                                            YPAY 160
*****                                                                       YPAY 170
***** PROBABLY A REGULAR LOOP WOULD BE BETTER.                              YPAY 180
YDOT     EQU   1                                                            YPAY 190
A        EQU   2                                                            YPAY 200
Y        EQU   3                                                            YPAY 210
X0       EQU   4                                                            YPAY 220
INDXA    EQU   5                                                            YPAY 230
IND14    EQU   6                                                            YPAY 240
I        EQU   7                                                            YPAY 250
IINC     EQU   8                                                            YPAY 260
J        EQU   10                                                           YPAY 270
         USING YPAY,15                                                      YPAY 280
         STM   14,12,12(13)                                                 YPAY 290
         LM    1,6,0(1)                                                     YPAY 300
         LA    IINC,8                                                       YPAY 310
         SR    I,I            I=0                                           YPAY 320
         L     9,0(6)         N                                             YPAY 330
         SLA   9,3            8N                                            YPAY 340
         LA    10,16                                                        YPAY 350
         SR    9,8            8(N-1)                                        YPAY 360
         LR    6,5            USE 6 TO REFERENCE (4I) AND (4I-2)            YPAY 370
         LA    11,X0ADD                                                     YPAY 380
LOOP     LD    0,0(X0,I)      FORM YDOT(I) IN 0.    =X0(I)                  YPAY 390
         L     12,4(IND14)    -14*(NO. NONZERO ELE. IN ROW I OF A)          YPAY 400
         B     0(11,12)                                                     YPAY 410
*                                                                           YPAY 420
*                                                                           YPAY 430
         L     J,408(INDXA)   AT LEAST 52                                   YPAY 440
         LD    2,408(A)                                                     YPAY 450
         MD    2,0(J,Y)                                                     YPAY 460
         ADR   0,2                                                          YPAY 470
         L     J,400(INDXA)   AT LEAST 51                                   YPAY 480
         LD    6,400(A)                                                     YPAY 490
         MD    6,0(J,Y)                                                     YPAY 500
         ADR   0,6                                                          YPAY 510
         L     J,392(INDXA)   AT LEAST 50                                   YPAY 520
         LD    4,392(A)                                                     YPAY 530
         MD    4,0(J,Y)                                                     YPAY 540
         ADR   0,4                                                          YPAY 550
         L     J,384(INDXA)   AT LEAST 49                                   YPAY 560
         LD    2,384(A)                                                     YPAY 570
         MD    2,0(J,Y)                                                     YPAY 580
         ADR   0,2                                                          YPAY 590
         L     J,376(INDXA)   AT LEAST 48                                   YPAY 600
         LD    6,376(A)                                                     YPAY 610
         MD    6,0(J,Y)                                                     YPAY 620
         ADR   0,6                                                          YPAY 630
         L     J,368(INDXA)   AT LEAST 47                                   YPAY 640
         LD    4,368(A)                                                     YPAY 650
```

```
MD     4,0(J,Y)                                                  YPAY 660
ADR    0,4                                                       YPAY 670
L      J,360(INDXA)     AT LEAST 46                              YPAY 680
LD     2,360 (A)                                                 YPAY 690
MD     2,0(J,Y)                                                  YPAY 700
ADR    0,2                                                       YPAY 710
L      J,352(INDXA)     AT LEAST 45                              YPAY 720
LD     6,352 (A)                                                 YPAY 730
MD     6,0(J,Y)                                                  YPAY 740
ADR    0,6                                                       YPAY 750
L      J,344(INDXA)     AT LEAST 44                              YPAY 760
LD     4,344 (A)                                                 YPAY 770
MD     4,0(J,Y)                                                  YPAY 780
ADR    0,4                                                       YPAY 790
L      J,336(INDXA)     AT LEAST 43                              YPAY 800
LD     2,336 (A)                                                 YPAY 810
MD     2,0(J,Y)                                                  YPAY 820
ADR    0,2                                                       YPAY 830
L      J,328(INDXA)     AT LEAST 42                              YPAY 840
LD     6,328 (A)                                                 YPAY 850
MD     6,0(J,Y)                                                  YPAY 860
ADR    0,6                                                       YPAY 870
L      J,320(INDXA)     AT LEAST 41                              YPAY 880
LD     4,320 (A)                                                 YPAY 890
MD     4,0(J,Y)                                                  YPAY 900
ADR    0,4                                                       YPAY 910
L      J,312(INDXA)     AT LEAST 40                              YPAY 920
LD     2,312 (A)                                                 YPAY 930
MD     2,0(J,Y)                                                  YPAY 940
ADR    0,2                                                       YPAY 950
L      J,304(INDXA)     AT LEAST 39                              YPAY 960
LD     6,304 (A)                                                 YPAY 970
MD     6,0(J,Y)                                                  YPAY 980
ADR    0,6                                                       YPAY 990
L      J,296(INDXA)     AT LEAST 38                              YPA 1000
LD     4,296 (A)                                                 YPA 1010
MD     4,0(J,Y)                                                  YPA 1020
ADR    0,4                                                       YPA 1030
L      J,288(INDXA)     AT LEAST 37                              YPA 1040
LD     2,288 (A)                                                 YPA 1050
MD     2,0(J,Y)                                                  YPA 1060
ADR    0,2                                                       YPA 1070
L      J,280(INDXA)     AT LEAST 36                              YPA 1080
LD     6,280 (A)                                                 YPA 1090
MD     6,0(J,Y)                                                  YPA 1100
ADR    0,6                                                       YPA 1110
L      J,272(INDXA)     AT LEAST 35                              YPA 1120
LD     4,272 (A)                                                 YPA 1130
MD     4,0(J,Y)                                                  YPA 1140
ADR    0,4                                                       YPA 1150
L      J,264(INDXA)     AT LEAST 34                              YPA 1160
LD     2,264 (A)                                                 YPA 1170
MD     2,0(J,Y)                                                  YPA 1180
ADR    0,2                                                       YPA 1190
L      J,256(INDXA)     AT LEAST 33                              YPA 1200
LD     6,256 (A)                                                 YPA 1210
```

```
MD     6,0 (J,Y)                                                   YPA  1220
ADR    0,6                                                         YPA  1230
L      J,248 (INDXA)      AT LEAST 32                              YPA  1240
LD     4,248(A)                                                    YPA  1250
MD     4,0 (J,Y)                                                   YPA  1260
ADR    0,4                                                         YPA  1270
L      J,240 (INDXA)      AT LEAST 31                              YPA  1280
LD     2,240(A)                                                    YPA  1290
MD     2,0 (J,Y)                                                   YPA  1300
ADR    0,2                                                         YPA  1310
L      J,232 (INDXA)      AT LEAST 30                              YPA  1320
LD     6,232(A)                                                    YPA  1330
MD     6,0 (J,Y)                                                   YPA  1340
ADR    0,6                                                         YPA  1350
L      J,224 (INDXA)      AT LEAST 29                              YPA  1360
LD     4,224(A)                                                    YPA  1370
MD     4,0 (J,Y)                                                   YPA  1380
ADR    0,4                                                         YPA  1390
L      J,216 (INDXA)      AT LEAST 28                              YPA  1400
LD     2,216(A)                                                    YPA  1410
MD     2,0 (J,Y)                                                   YPA  1420
ADR    0,2                                                         YPA  1430
L      J,208 (INDXA)      AT LEAST 27                              YPA  1440
LD     6,208(A)                                                    YPA  1450
MD     6,0 (J,Y)                                                   YPA  1460
ADR    0,6                                                         YPA  1470
L      J,200 (INDXA)      AT LEAST 26                              YPA  1480
LD     4,200(A)                                                    YPA  1490
MD     4,0 (J,Y)                                                   YPA  1500
ADR    0,4                                                         YPA  1510
L      J,192 (INDXA)      AT LEAST 25                              YPA  1520
LD     2,192(A)                                                    YPA  1530
MD     2,0 (J,Y)                                                   YPA  1540
ADR    0,2                                                         YPA  1550
L      J,184 (INDXA)      AT LEAST 24                              YPA  1560
LD     6,184(A)                                                    YPA  1570
MD     6,0 (J,Y)                                                   YPA  1580
ADR    0,6                                                         YPA  1590
L      J,176 (INDXA)      AT LEAST 23                              YPA  1600
LD     4,176(A)                                                    YPA  1610
MD     4,0 (J,Y)                                                   YPA  1620
ADR    0,4                                                         YPA  1630
L      J,168 (INDXA)      AT LEAST 22                              YPA  1640
LD     2,168(A)                                                    YPA  1650
MD     2,0 (J,Y)                                                   YPA  1660
ADR    0,2                                                         YPA  1670
L      J,160 (INDXA)      AT LEAST 21                              YPA  1680
LD     6,160(A)                                                    YPA  1690
MD     6,0 (J,Y)                                                   YPA  1700
ADR    0,6                                                         YPA  1710
L      J,152 (INDXA)      AT LEAST 20                              YPA  1720
LD     4,152(A)                                                    YPA  1730
MD     4,0 (J,Y)                                                   YPA  1740
ADR    0,4                                                         YPA  1750
L      J,144 (INDXA)      AT LEAST 19                              YPA  1760
LD     2,144(A)                                                    YPA  1770
```

```
MD      2,0(J,Y)                                                  YPA 1780
ADR     0,2                                                       YPA 1790
L       J,136(INDXA)    AT LEAST 18                               YPA 1800
LD      6,136(A)                                                  YPA 1810
MD      6,0(Y,J)                                                  YPA 1820
ADR     0,6                                                       YPA 1830
L       J,128(INDXA)    AT LEAST 17                               YPA 1840
LD      4,128(A)                                                  YPA 1850
MD      4,0(Y,J)                                                  YPA 1860
ADR     0,4                                                       YPA 1870
L       J,120(INDXA)    AT LEAST 16                               YPA 1880
LD      2,120(A)                                                  YPA 1890
MD      2,0(Y,J)                                                  YPA 1900
ADR     0,2                                                       YPA 1910
L       J,112(INDXA)    AT LEAST 15                               YPA 1920
LD      6,112(A)                                                  YPA 1930
MD      6,0(Y,J)                                                  YPA 1940
ADR     0,6                                                       YPA 1950
L       J,104(INDXA)    AT LEAST 14                               YPA 1960
LD      4,104(A)                                                  YPA 1970
MD      4,0(Y,J)                                                  YPA 1980
ADR     0,4                                                       YPA 1990
L       J,96(INDXA)     AT LEAST 13                               YPA 2000
LD      2,96(A)                                                   YPA 2010
MD      2,0(Y,J)                                                  YPA 2020
ADR     0,2                                                       YPA 2030
L       J,88(INDXA)     AT LEAST 12                               YPA 2040
LD      6,88(A)                                                   YPA 2050
MD      6,0(Y,J)                                                  YPA 2060
ADR     0,6                                                       YPA 2070
L       J,80(INDXA)     AT LEAST 11                               YPA 2080
LD      4,80(A)                                                   YPA 2090
MD      4,0(Y,J)                                                  YPA 2100
ADR     0,4                                                       YPA 2110
L       J,72(INDXA)     AT LEAST 10 NONZERO ENTRIES IN ROW I      YPA 2120
LD      2,72(A)                                                   YPA 2130
MD      2,0(Y,J)                                                  YPA 2140
ADR     0,2                                                       YPA 2150
L       J,64(INDXA)                9                              YPA 2160
LD      6,64(A)                                                   YPA 2170
MD      6,0(Y,J)                                                  YPA 2180
ADR     0,6                                                       YPA 2190
L       J,56(INDXA)                8                              YPA 2200
LD      4,56(A)                                                   YPA 2210
MD      4,0(Y,J)                                                  YPA 2220
ADR     0,4                                                       YPA 2230
L       J,48(INDXA)                7                              YPA 2240
LD      2,48(A)                                                   YPA 2250
MD      2,0(Y,J)                                                  YPA 2260
ADR     0,2                                                       YPA 2270
L       J,40(INDXA)                                               YPA 2280
LD      6,40(A)                                                   YPA 2290
MD      6,0(Y,J)                                                  YPA 2300
ADR     0,6                                                       YPA 2310
L       J,32(INDXA)               5                               YPA 2320
LD      4,32(A)                                                   YPA 2330
```

```
         MD    4,0(Y,J)                                                YPA 2340
         ADR   0,4                                                     YPA 2350
         L     J,24(INDXA)         4                                   YPA 2360
         LD    2,24(A)                                                 YPA 2370
         MD    2,0(Y,J)                                                YPA 2380
         ADR   0,2                                                     YPA 2390
         L     J,16(INDXA)         3                                   YPA 2400
         LD    6,16(A)                                                 YPA 2410
         MD    6,0(Y,J)                                                YPA 2420
         ADR   0,6                                                     YPA 2430
         L     J,8(INDXA)          2                                   YPA 2440
         LD    4,8(A)                                                  YPA 2450
         MD    4,0(Y,J)                                                YPA 2460
         ADR   0,4                                                     YPA 2470
         L     J,00(INDXA)     AT LEAST 1 NONZERO ENTRY IN ROW I       YPA 2480
         LD    2,00(A)                                                 YPA 2490
         MD    2,00(Y,J)                                               YPA 2500
         ADR   0,2                                                     YPA 2510
*                                                                      YPA 2520
YOADD    L     12,12(IND14)    8*(NO. NONZERO ENTRIES IN ROW I OF A)   YPA 2530
         AR    A,12                                                    YPA 2540
         AR    INDXA,12                                                YPA 2550
         AR    IND14,0                                                 YPA 2560
         STD   0,0(YDOT,I)                                             YPA 2570
         BXLE  I,IINC,LOOP                                             YPA 2580
         LM    14,12,12(13)                                            YPA 2590
         BR    14                                                      YPA 2600
         END                                                           YPA 2610

IHC002I  STOP      0
```

APPENDIX B

SAMPLE INPUT AND OUTPUT

Included in this appendix are three examples of input and the corresponding output produced by the code. For each example we give the retention model, the input cards, and the output. The full printed output (unit 6 and unit 10) is given only for the first and third examples. Printout from unit 10 only is given for the second example.

Example 1

Radionuclide - CS-134

Half life - 2.05 years or 749. days

Mode of intake - inhalation (class D); see Fig. 2.1.

$f_1 = 0.95$

Source organs

1. Lungs

2. GIT

It is assumed that the total body retention is given by:

3. Total body: $R(t) = 0.13e^{-\frac{.693}{1.4}t} + 0.87e^{-\frac{.693}{135}t}$

Print cumulated activity (μCi-days) at these times:

2 da

7 da

30 da

60 da

180 da

1 yr

5 yrs

10 yrs

20 yrs

30 yrs

40 yrs

50 yrs

Below is an explanation of how Example 1 should be coded into card input. Follow Table 5.1, the description of the input data, and Table B.1, the data sheet, to fully understand the instructions.

Information Concerning the Differential Equations

Card 1 NØ: The number of differential equations.

12 Lungs
4 GIT
2 Total body
—
18

TLAST: The final value of T is 50 years or 18262.5 days. All times must be expressed in days.

H1 and EPS are set to zero (left blank) therefore taking default values.

Card 2 NPRT: We wish to print cumulated activity (μCi-days) at 12 different times; however, only 10 print intervals are allowable. Since some time periods are equally spaced, the time intervals may be considered as

$\underline{2.}$
$\underline{7.}$
$\underline{30.}$ + I•($\underline{30.}$) , where I=0,$\underline{1}$
$\underline{180.}$
$\underline{365.25}$
$\underline{1826.5}$
$\underline{3652.5}$ + I•($\underline{3652.5}$) , where I=0,1,2,3,$\underline{4}$

Thus there are 7 print intervals and TPRL(I), DTPR(I), NARPRT(I) are set equal to the underlined numbers as follows:

Table 8.1. CS-134: Intake by Inhalation

```
   18   16262.5
    7
        2.
        7.
       30.       38.    1
      100.
    366.28
   1026.26
    3628.5    3628.8    4
         1.       .3       .88       .26         1
CS-134          709.         8      114.
237.0007   2
      0.13      0.07     0.495     0.006
```

12 blank cards

	TPRL(I)	DTPR(I)	NARPRT(I)
Card 3	2.	0.	0
	7.	0.	0
.	30.	30.	1
.	180.	0.	0
.	365.25	0.	0
	1826.5	0.	0
Card 9	3652.5	3652.5	4

General Information for Chain of Radionuclides

Card 1 NISØ: 1 radionuclide in the chain
D3: .3
D4: .08 ——— deposition fractions
D5: .25
NØRG: 1 organ (excluding lungs and GIT)
IØPT: 0-inhalation
NØRAL: not applicable, set to 0

Card 2 TAG(1): CS-134
TR(1): 749. days
ICLASS(1): D
LAMAB(1,2): 6•0.95/(1-0.95) = 114.
P(1),NBRNCH(1),IBRNCH(1): not applicable, set to 0
ILAB: 0

Card 3 omitted because ILAB=0

Description of Retention

Card 1 omitted for first nuclide in chain

Card 2 ISØRS(1): 23
ØRGNM(1): T.BØDY
ICØM(1,1): 2

Card 3 AS(1,1): 0.13
AS(2,1): 0.87
TBS(1,1): 0.693/1.4=0.495
TBS(2,1): 0.693/135.=0.005

Other Tissues Transformation

Since no transformation is to be performed, 1 blank card must be present for each time the cumulated activities (μCi-days) are printed. Note this is 12, not 7.

Printout for Example 1 includes the following:

1. Summary of input concerning the differential equations.
2. Coefficient matrix printed by rows.
3. Nonzero elements of the coefficient matrix and the rows and columns where these elements appear.
4. The progress of the solution of the differential equations which includes the values of Y for each T.
5. Summary printout of input concerning the radionuclides and retention information for total body.
6. The remainder of the printout consists of cumulated activity for each organ at each time period. Printout for time periods 20 yrs., 30 yrs., 40 yrs., and 50 yrs. is omitted since the cumulated activities for these time periods are the same as those for 10 yrs. (3652.5 days).

Items 1-4 are output on unit 6; items 5 and 6 are output on unit 10.

OUTPUT FOR SAMPLE [illegible]

CS-[illegible] INTAKE BY INHALATION

```
NO. PRINT INTERVALS DESIRED = 7   NO. READ = 7
POINTS SELECTED TO PRINT INTERVAL = [illegible]   WHERE Q=0,1,....,N
        T                   DT                  N
 2.[illegible] 00   0.0                          [illegible]
 7.[illegible] 00   0.0                          0
 [illegible] 01   [illegible] 01       1
 1.[illegible] 02   0.0                          0
 [illegible] 02   0.0                          5
 1.[illegible] 03   0.0                          0
 [illegible] 03   [illegible] 03       [illegible]

NM= 14
NP=21
T0= 0.0
TLAST= 1.[illegible] 0[illegible]
H0= 1.[illegible]
EPS=1.00[illegible]-0[illegible]
```

```
MATRIX A PRINTED BY ROWS

 -0.6931467D 02  0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0           -0.6931467D 02  0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0           -0.6931467D 02  0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0           -0.3465740D 01  0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0           -0.1386294D 01  0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0           -0.1386294D 01
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.1386294D 01
 -0.1386294D 01  0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0           -0.3465740D 01  0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0           -0.3465740D 01  0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.6931467D 02  0.0            0.3465740D 01  0.0            0.0            0.0            0.0
  0.0            0.0            0.3465740D 01  0.3465740D 01 -0.2400000D 02  0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.2400000D 02 -0.1200000D 03  0.0            0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.6000000D 01 -0.1446150D 01  0.0
  0.0            0.0

  0.0            0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0            0.0            0.0            0.0            0.0            0.1446150D 01 -0.1000000D 01
  0.0            0.0

  0.9010910D 01  0.0            0.9010910D 01  0.0            0.1002180D 00  0.0            0.0            0.0
  0.1002180D 00  0.0            0.0            0.0            0.0            0.1482000D 02  0.0            0.0
 -0.4950000D 00  0.0

  0.6030300D 02  0.0            0.6030300D 02  0.0            0.1206000D 01  0.0            0.0            0.0
  0.1206000D 01  0.0            0.0            0.0            0.0            0.9918000D 02  0.0            0.0
  0.0           -0.5000000D-02
  ENTRIES OF A    ROW   COL
 -3.6931967D 02     1     1
 -3.6931967D 02     2     2
 -3.6931967D 02     3     3
 -0.3445746D 01     4     4
 -0.1146029D 01     5     5
  0.0               6     6
  0.0               7     7
 -0.1146292D 01     8     8
  0.1146292D 01     9     8
 -0.1146292D 01     9     9
  0.0              10     4
  0.0              10    10
  0.0              11     6
 -0.3445763D 01    11    11
  0.0              12     7
 -0.3445746D 01    12    12
  0.6931967D 02    13     2
  0.3445746D 01    13     4
  0.3445746D 01    13    11
  0.3445746D 01    13    12
 -0.2460903D 02    13    13
  0.2460903D 02    14    13
 -0.1840903D 01    14    14
  0.4330803D 01    15    14
 -0.1146815D 01    15    15
  0.1146815D 01    16    15
 -0.1136600D 01    16    16
 -0.9510418D 01    17     1
  0.4010518D 01    17     3
  0.1802199D 00    17     5
  0.1802199D 00    17     6
  0.0              17    13
  0.1642330D 02    17    14
  0.0              17    15
  0.0              17    16
 -0.4956002D 02    17    17
  0.6030380D 02    18     1
  0.6030340D 02    18     3
  0.1206060D 01    18     5
  0.1206060D 01    18     6
  0.0              18    13
  0.5919000D 02    18    14
  0.0              18    15
  0.0              18    16
 -0.5010000D-02    18    18


KFLAG = -1 FROM STIFF AT T =     0.0
 ERROR TEST FAILED WITH ABS(H) = HMIN

 H HAS BEEN REDUCED TO    0.99999999999999990D-00  AND STEP WILL BE RETRIED



T= 2.0000000000D 00    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15272208510D-03    1.35179077090D-01    0.0
  0.0                  3.37967492720D-02    2.75379577520D-02    0.0                  0.0                  0.0
  6.01410164210D-03    1.28320926590D-03    4.00913013610D-03    5.50633930450D-03    8.04292093230D-02    4.90773021130D-01

T= 7.0000000000D 00    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15300792900D-03    1.40160491160D-01    0.0
  0.0                  1.40811227990D-02    3.59950095310D-02    0.0                  0.0                  0.0
  6.01629132180D-03    1.28320836910D-03    4.16005540600D-03    7.67210000550D-03    1.56070338160D-01    3.51030094600D 03

T= 3.0000000000D 01    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15300792910D-03    1.40173260200D-01    0.0
  0.0                  3.40033150590D-02    3.60192699450D-02    0.0                  0.0                  0.0
  6.01629132170D-03    1.28320836800D-03    4.16006700930D-03    7.60051730070D-03    1.63071050190D-01    1.07070027900D 01

T= 6.0000000000D 01    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15300792910D-03    1.40173260240D-01    0.0
  0.0                  3.40033152610D-02    3.60192700670D-02    0.0                  0.0                  0.0
  6.01629132170D-03    1.28320836800D-03    4.16006700930D-03    7.60051738590D-03    1.63071555210D-01    2.71070300600D 01

T= 1.0000000000D 02    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15300792910D-03    1.40173260260D-01    0.0
  0.0                  3.40033150600D-02    3.60192701500D-02    0.0                  0.0                  0.0
  6.01629132170D-03    1.28320836800D-03    4.16006700930D-03    7.60051739690D-03    1.63071561110D-01    5.90317075130D 01

T= 3.6525000000D 02    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15300792910D-03    1.40173260260D-01    0.0
  0.0                  3.40033150600D-02    3.60192701500D-02    0.0                  0.0                  0.0
  6.01629132170D-03    1.28320836800D-03    4.16006700930D-03    7.60051739600D-03    1.63071561160D-01    8.00262111000D 01

T= 1.0262500000D 03    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15300792910D-03    1.40173260260D-01    0.0
  0.0                  3.40033150600D-02    3.60192701500D-02    0.0                  0.0                  0.0
  6.01629132170D-03    1.28320836800D-03    4.16006700930D-03    7.60051739600D-03    1.63071561160D-01    9.13359702330D 01

T= 3.6525000000D 03    Y FOLLOWS
  2.16401366930D-03    2.16401366930D-03    1.09603359240D-03    1.15300792910D-03    1.40173260260D-01    0.0
  0.0                  3.40033150600D-02    3.60192701500D-02    0.0                  0.0                  0.0
  6.01629132170D-03    1.28320836800D-03    4.16006700930D-03    7.60051739600D-03    1.63071561160D-01    9.13370107900D 01
```

```
T= 7.30500000000D 03    Y FOLLOWS
   2.16401366930D-03    2.16401366930D-03    1.09643359240D-03    1.15304792910D-03    1.44173266260D-01    0.0
   0.0                  3.60433150640D-02    3.60192701580D-02    0.0                  0.0                  0.0
   4.41629132170D-03    1.20324836900D-03    4.16046764930D-03    7.63051739680D-03    1.63071561160D-01    9.13378170410D 01

T= 1.09575000000D 04    Y FOLLOWS
   2.16401366930D-03    2.16401366930D-03    1.09643359240D-03    1.15304792910D-03    1.44173260260D-01    0.0
   0.0                  3.60433150640D-02    3.60192701580D-02    0.0                  0.0                  0.0
   4.41629132170D-03    1.20324836800D-03    4.16046764930D-03    7.60051739680D-03    1.63071561160D-01    9.13378164210D 01

T= 1.46100000000D 04    Y FOLLOWS
   2.16401366930D-03    2.16401366930D-03    1.09643359240D-03    1.15304792910D-03    1.44173260260D-01    0.0
   0.0                  3.60433150640D-02    3.60192701580D-02    0.0                  0.0                  0.0
   4.41629132170D-01    1.20324836900D-03    4.16046764930D-03    7.60051739680D-03    1.63071561160D-01    9.13378164570D 01

T= 1.82625000000D 04    Y FOLLOWS
   2.16401366930D-03    2.16401366930D-03    1.09643.59240D-03    1.15304792910D-03    1.44173260260D-01    0.0
   0.0                  3.60433150640D-02    3.60192701580D-02    0.0                  0.0                  0.0
   4.41629132179D-03    1.20324836900D-03    4.16046764930D-03    7.60051739680D-03    1.63071561160D-01    9.13378164590D 01

KFLAG =       0
```

```
PERCENT DEPOSITION
     N-P              T-B              P
   0.30             0.00             0.25
ISOTOPE    HALFLIFE    CLASS   LAMBD   BR.RATIO          RADNUC?     IBRNCH
CS-136     7.4982 02     D   1.1600 02 1.0000 00           0           0



ISOTOPE     ORGAN      A'S AND TB'S
CS-136
           T.BODY        1.3000-01   8.7000-01   4.9500-01   5.0000-33
```

```
T = 2.0000D 00

MICROCURIES-DAYS FROM INHALATION OF CS-136

LUNGS                                                            MICROCURIES ENTERING
                                                                    BLOOD        GI
ISOTOPE        N-P          T-B           P           LYMPH
CS-136     4.1280D-03  2.2492D-03  1.6897D-01  2.7530D-02        4.5157D-01  1.5399D-01


GI TRACT                                                         MICROCURIES ENTERING
                                                                    BLOOD
ISOTOPE     STOMACH        S.I.         U.L.I.       L.L.I.
CS-136     6.6161D-03  1.2932D-03  4.0481D-03  5.5063D-03        1.4629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-136     9.8320D-01
```

```
T = 7.0000D 00

MICROCURIES-DAYS FROM INHALATION OF CS-136

LUNGS                                                            MICROCURIES ENTERING
                                                                    BLOOD        GI
ISOTOPE        N-P          T-B           P           LYMPH
CS-136     4.3380D-03  2.2503D-03  1.8021D-01  3.5994D-02        4.7577D-01  1.5400D-01


GI TRACT                                                         MICROCURIES ENTERING
                                                                    BLOOD
ISOTOPE     STOMACH        S.I.         U.L.I.       L.L.I.
CS-136     6.6163D-03  1.2932D-03  4.1605D-03  7.6721D-03        1.4629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-136     3.6788D 00
```

```
T = 3.0600D 01

MICROCURIES-DAYS FROM INHALATION OF CS-134

LUNGS                                                            MICROCURIES ENTERING
                                                                    BLOOD       GI
ISOTOPE       N-P         T-B          P          LYMPH
CS-134     4.3280D-03  2.2503D-03  1.0622D-01  3.6019D-02         4.7500D-01  1.5400D-01


GI TRACT                                                         MICROCURIES ENTERING
                                                                    BLOOD
ISOTOPE     STOMACH       S.I.       U.L.I.      L.L.I.
CS-134     6.4163D-03  1.2832D-03  4.1665D-03  7.6005D-03         1.0629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-134     1.4870D 01
```

```
T = 6.3400D 01

MICROCURIES-DAYS FROM INHALATION OF CS-134

LUNGS                                                            MICROCURIES ENTERING
                                                                    BLOOD       GI
ISOTOPE       N-P         T-B          P          LYMPH
CS-134     4.3280D-03  2.2503D-03  1.0022D-01  3.6019D-02         4.7500D-01  1.5400D-01


GI TRACT                                                         MICROCURIES ENTERING
                                                                    BLOOD
ISOTOPE     STOMACH       S.I.       U.L.I.      L.L.I.
CS-134     6.4163D-03  1.2932D-03  4.1695D-03  7.6005D-03         1.0629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-134     2.7350E 01
```

```
T = 1.8000D 02

MICROCURIES-DAYS FROM INHALATION OF CS-134

LUNGS                                                            MICROCURIES ENTERING
                                                                    BLOOD       GI
ISOTOPE       N-P         T-B          P          LYMPH
CS-134     4.3280D-03  2.2503D-03  1.0022D-01  3.6019D-02         4.7500D-01  1.5400D-01


GI TRACT                                                         MICROCURIES ENTERING
                                                                    BLOOD
ISOTOPE     STOMACH       S.I.       U.L.I.      L.L.I.
CS-134      .4163D-03  1.2832D-03  4.1665D-03  7.6005D-03         1.0629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-134     5.9995D 01
```

```
T =  3.6525D 02

MICROCURIES-DAYS FROM INHALATION OF CS-134

LUNGS                                                                   MICROCURIES ENTERING
                                                                           BLOOD        GI
ISOTOPE       N-P          T-B           P           LYMPH
CS-134     4.3290D-03  2.2503D-03  1.8022D-01  3.6019D-02             4.7580D-01  1.5400D-01


GI TRACT                                                                MICROCURIES ENTERING
                                                                           BLOOD
ISOTOPE     STOMACH        S.I.        U.L.I.       L.L.I.
CS-134     6.4163D-03  1.2832D-03  4.1685D-03  7.6865D-03             1.4629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-134     9.0989D 01




T =  1.8263D 03

MICROCURIES-DAYS FROM INHALATION OF CS-134

LUNGS                                                                   MICROCURIES ENTERING
                                                                           BLOOD        GI
ISOTOPE       N-P          T-B           P           LYMPH
CS-134     4.3280D-03  2.2503D-03  1.8022D-01  3.6019D-02             4.7580D-01  1.5400D-01


GI TRACT                                                                MICROCURIES ENTERING
                                                                           BLOOD
ISOTOPE     STOMACH        S.I.        U.L.I.       L.L.I.
CS-134     6.4163D-03  1.2832D-03  4.1685D-03  7.6865D-03             1.4629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-134     9.1499D 01




T =  3.6525D 03

MICROCURIES-DAYS FROM INHALATION OF CS-134

LUNGS                                                                   MICROCURIES ENTERING
                                                                           BLOOD        GI
ISOTOPE       N-P          T-B           P           LYMPH
CS-134     4.3293D-03  2.2503D-03  1.8022D-01  3.6019D-02             4.7590D-01  1.5400D-01


GI TRACT                                                                MICROCURIES ENTERING
                                                                           BLOOD
ISOTOPE     STOMACH        S.I.        U.L.I.       L.L.I.
CS-134     6.4163D-03  1.2832D-03  4.1685D-03  7.6865D-03             1.4629D-01


OTHER ORGANS

ISOTOPE      T.BODY
CS-134     9.1501D 01
```

Example 2

Chain of Radionuclides -

$$\begin{tikzcd}
 & {}^{131}_{52}\mathrm{Te} \arrow[dr] & & {}^{131M}_{54}\mathrm{Xe} \arrow[dr] & \\
{}^{131M}_{52}\mathrm{Te} \arrow[ur, "18\%"] \arrow[rr, "82\%"'] & & {}^{131}_{53}\mathrm{I} \arrow[ur, "1.44\%"] \arrow[rr, "98.56\%"'] & & {}^{131}_{54}\mathrm{Xe}\ \text{(stable)}
\end{tikzcd}$$

Half lives -	Te-131M	30 hours
	Te-131	25 minutes
	I-131	8.06 days
	Xe-131M	11.8 days

Mode of intake - inhalation (class D - all radionuclides)

Tellurium as a parent of Iodine:

$f_1 = .25$

Source organs

1. Lungs

2. GIT

3. Cort bone: $R(t) = 0.0183e^{-\frac{.693}{9.2}t} + 0.01e^{-\frac{.693}{8000}t}$

4. Tra bone: $R(t) = 4.57 \times 10^{-3}e^{-\frac{.693}{9.2}t} + 0.01e^{-\frac{.693}{8000}t}$

5. Kidney: $R(t) = 1.42 \times 10^{-3}e^{-\frac{.693}{9.2}t} + 0.034e^{-\frac{.693}{23}t}$

6. Muscle: $R(t) = 0.128e^{-\frac{.693}{9.2}t} + 0.1e^{-\frac{.693}{17.7}t}$

7. Liver: $R(t) = 8.23 \times 10^{-3}e^{-\frac{.693}{9.2}t} + 0.055e^{-\frac{.693}{10.2}t}$

8. Thyroid: $R(t) = 9.14 \times 10^{-5}e^{-\frac{.693}{9.2}t}$

9. Other tissue: $R(t) = 0.159e^{-\frac{.693}{9.2}t}$

Te-131M and Te-131 do not recirculate to blood.

Iodine as a daughter of Tellurium:

$f_1 = .95$

Source organs

1. Lungs

2. GIT

3. Thyroid: $R(t) = -0.328e^{-\frac{.693}{0.243}t} + 0.016e^{-\frac{.693}{11.3}t} + 0.312e^{-\frac{.693}{117}t}$

4. Cort bone: $R(t) = 5.7 \times 10^{-2}e^{-\frac{.693}{0.243}t} - 2.78 \times 10^{-3}e^{-\frac{.693}{11.3}t}$

$+ 2.94 \times 10^{-3}e^{-\frac{.693}{117}t}$

5. Tra bone: $R(t) = 1.42 \times 10^{-2}e^{-\frac{.693}{0.243}t} - 6.96 \times 10^{-4}e^{-\frac{.693}{11.3}t}$

$+ 7.35 \times 10^{-4}e^{-\frac{.693}{117}t}$

6. Kidney: $R(t) = 4.42 \times 10^{-3}e^{-\frac{.693}{0.243}t} - 2.16 \times 10^{-4}e^{-\frac{.693}{11.3}t}$

$+ 2.28 \times 10^{-4}e^{-\frac{.693}{117}t}$

7. Muscle: $R(t) = 3.99 \times 10^{-1}e^{-\frac{.693}{0.243}t} - 1.95 \times 10^{-2}e^{-\frac{.693}{11.3}t}$

$+ 2.06 \times 10^{-2}e^{-\frac{.693}{117}t}$

8. Liver: $R(t) = 2.56 \times 10^{-2}e^{-\frac{.693}{0.243}t} - 1.25 \times 10^{-3}e^{-\frac{.693}{11.3}t}$

$+ 1.32 \times 10^{-3}e^{-\frac{.693}{117}t}$

9. Other tissue: $R(t) = 4.97 \times 10^{-1} e^{-\frac{.693}{0.243}t} - 2.43 \times 10^{-2} e^{-\frac{.693}{11.3}t}$

$$+ 2.56 \times 10^{-2} e^{-\frac{.693}{117}t}$$

I-131 recirculates to blood. Absorption takes place in the UL and LL with an f_1 of .95.

Xenon as a granddaughter of Tellurium:

$f_1 = .95$

Source organs

1. Lungs

2. GIT

3. Thyroid: $R(t) = 1.0e^{-\frac{.693}{.0833}t}$

4. Cort bone: $R(t) = 0.0571e^{-\frac{.693}{.0833}t}$

5. Tra bone: $R(t) = 0.0143e^{-\frac{.693}{.0833}t}$

6. Kidney: $R(t) = 4.43 \times 10^{-3} e^{-\frac{.693}{.0833}t}$

7. Muscle: $R(t) = 0.4e^{-\frac{.693}{.0833}t}$

8. Liver: $R(t) = 0.0257e^{-\frac{.693}{.0833}t}$

9. Other tissue: $R(t) = 0.498e^{-\frac{.693}{.0833}t}$

Xe-131 recirculates to blood.

Print cumulated activity (uCi-days) at the same times as Example 1.

Again we include a discussion of how Example 2 must be coded. Follow Table 5.1 and Table B.2 to fully understand the instructions.

Information Concerning the Differential Equations

Card 1 NØ: 4 x 12 = 48 Lungs
4 x 4 = 16 GIT
12 compartments for Te-131M (excluding GIT and Lungs)
12 compartments for Te-131 (excluding GIT and Lungs)
21 compartments for I-131 (excluding GIT and Lungs)
7 compartments for Xe-131M (excluding GIT and Lungs)
116

TLAST, H1, EPS same as Example 1.

Card 2...Card 9 same as Example 1.

General Information for Chain of Radionuclides

Card 1
NISØ: 4
D3: .3
D4: .08
D5: .25
NØRG: 7
IØPT: 0
NØRAL: not applicable, set to 0

Card 2
TAG(1): TE-131M
TR(1): 1.25
ICLASS(1): D
LAMAB(1,2): 2.
P(1),NBRNCH(1),IBRNCH(1): not applicable
ILAB: 0

Card 3 omitted because ILAB=0

Card 4
TAB(2): TE-131
TR(2): 0.01736
ICLASS(2): D
LAMAB(2,2): 2.
P(2): 0.18
NBRNCH(2): 2
IBRNCH(2): 1
ILAB: 0

Card 5 omitted because ILAB=0

Table B.2. TE-131M Chain: Intake by Inhalation

```
  116   10262.5
    7
        2.
        7.
       30.       30.    1
      100.
    365.25
   1826.25
    3652.5     3652.5   4
         4         .3      .06       .25         7
TE-131M          1.25        0        2.
TE-131        6.01736        0        2.       0.18 2 1
I-131            8.06        0      114.       0.82 2 2   1
        8.       114.      114.
XE-131M          11.8        0      114.     0.0144 1 1
15COR BONE 2
    0.0103       0.01  7.533E-2 8.663E-5
16TRA BONE 2
   4.57E-3       0.01  7.533E-2 8.663E-5
 6KIDNEYS  2
   1.42E-3      9.034  7.533E-2 3.013E-2
12MUSCLE   2
     8.128        0.1  7.533E-2 3.915E-2
 7LIVER    2
   8.23E-3      0.055  7.533E-2 6.794E-2
```

Table B.2. (cont'd)

```
12345678901234567890123456789012345678901234567890123456789012345678901234567890
22THYROID  1
   9.14E-5  7.533E-2
23OTH TISS 1
     0.159  7.533E-2
 0 0 0 0 0 0 0
15CBR BONE 2
    0.0103      0.01  7.533E-2  8.663E-5
16TRA BONE 2
    4.47E-3     0.01  7.533E-2  8.663E-5
19KIDNEYS  2
    1.42E-3    0.034  7.533E-2  3.013E-2
12MUSCLE   2
     0.128       0.1  7.533E-2  3.916E-2
17LIVER    2
    8.23E-3    0.066  7.533E-2  6.794E-2
22THYROID  1
   9.14E-5  7.533E-2
23OTH TISS 1
     0.159  7.533E-2
 1 1 1 1 1 1 1
15CBR BONE 3
   8.20E-2  -2.78E-3   2.94E-3     2.052  6.133E-2  5.923E-3
   1.42E-2  -4.80E-4   7.29E-4     2.052  6.133E-2  5.923E-3
16TRA BONE 3
19KIDNEYS  3
```

Table B.2. (cont'd)

```
   4.42E-3   -2.16E-4   2.28E-4       2.852    6.133E-2   5.923E-3
12MUSCLF   3
   3.99E-1   -1.05E-2   2.06E-2       2.852    6.133E-2   5.923E-3
 7LIVER    3
   2.50E-2   -1.25E-3   1.32E-3       2.852    6.133E-2   5.923E-3
22THYROID  3
    -0.328      0.016     0.312       2.852    6.133E-2   5.923E-3
23OTH TISS 3
   4.97E-1   -2.43E-2   2.56E-2       2.852    6.133E-2   5.923E-3
15COR BONE 1
    0.0857      0.319
16TRA BONE 1
    0.0143      0.319
 8KIDNEYS  1
   4.43E-3      0.319
12MUSCLE   1
       0.2      0.319
 7LIVER    1
    0.0267      0.319
22THYROID  1
 [illegible]    0.319
23OTH TISS 1
     0.455      0.319
[illegible]
```

Table 8.2. (cont'd)

```
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
 11516 612 722
```

<u>Card 6</u>	TAG(3): I-131 TR(3): 8.06 ICLASS(3): D LAMAB(3,2): 114. P(3): 0.82 NBRNCH(3): 2 IBRNCH(3): 2 ILAB: 1
<u>Card 7</u>	LAMAB(3,1) = 0 LAMAB(3,3) = 114. LAMAB(3,4) = 114.
<u>Card 8</u>	TAG(4): XE-131M TR(4): 11.8 ICLASS(4): D LAMAB(4,2): 114. P(4) = 0.0144 NBRNCH(4): 1 IBRNCH(4): 1 ILAB = 0

<u>Card 9</u> omitted because ILAB=0

<u>Description of Retention</u>

FØR TE-131M

<u>Card 1</u> omitted for first nuclide in the chain

<u>Cards 2 and 3</u> repeated for each organ

Cort bone

<u>Card 2</u>	ISØRS(1): 15 ØRGNM(1): CØR BØNE ICØM(1,1): 2	<u>Card 3</u>	AS(1,1): 0.0183 AS(2,1): 0.01 TBS(1,1): 7.533×10^{-2} TBS(2,1): 8.663×10^{-5}

Tra bone

<u>Card 2</u>	ISØRS(2): 16 ØRGNM(2): TRA BØNE ICØM(1,2): 2	<u>Card 3</u>	AS(1,2): 4.57×10^{-3} AS(2,2): 0.01 TBS(1,2): 7.533×10^{-2} TBS(2,2): 8.663×10^{-5}

Kidneys

Card 2	ISØRS(3):	6	Card 3	AS(1,3):	1.42×10^{-3}
	ØRGNM(3):	KIDNEYS		AS(2,3):	0.034
	ICØM(1,3):	2		TBS(1,3):	7.533×10^{-2}
				TBS(2,3):	3.013×10^{-2}

Muscle

Card 2	ISØRS(4):	12	Card 3	AS(1,4):	0.128
	ØRGNM(4):	MUSCLE		AS(2,4):	0.1
	ICØM(1,4):	2		TBS(1,4):	7.533×10^{-2}
				TBS(2,4):	3.915×10^{-2}

Liver

Card 2	ISØRS(5):	7	Card 3	AS(1,5):	8.23×10^{-3}
	ØRGNM(5):	LIVER		AS(2,5):	0.055
	ICØM(1,5):	2		TBS(1,5):	7.533×10^{-2}
				TBS(2,5):	6.794×10^{-2}

Thyroid

Card 2	ISØRS(6):	2	Card 3	AS(1,6):	9.14×10^{-5}
	ØRGNM(6):	THYRØID		TBS(1,6):	7.533×10^{-2}
	ICØM(1,6):	1			

Other tissue

Card 2	ISØRS(7):	23	Card 3	AS(1,7):	0.159
	ØRGNM(7):	ØTH TISS		TBS(1,7):	7.533×10^{-2}
	ICØM(1,7):	1			

FOR TE-131

Card 1	IS(1):	0
	IS(2):	0
	IS(3):	0
	IS(4):	0
	IS(5):	0
	IS(6):	0
	IS(7):	0

The remainder of the cards for TE-131 are the same as for TE-131M since the retention functions are the same.

FØR I-131

Card 1	IS(1):	1
	IS(2):	1
	IS(3):	1
	IS(4):	1
	IS(5):	1
	IS(6):	1
	IS(7):	1

Cards 2 and 3 are repeated for each organ

Cort bone

Card 2	ISØRS(1):	15	Card 3	AS(1,1):	5.70×10^{-2}
	ØRGNM(1):	CØR BØNE		AS(2,1):	-2.78×10^{-3}
	ICØM(3,1):	3		AS(3,1):	2.94×10^{-3}
				TBS(1,1):	2.852
				TBS(2,1):	6.133×10^{-2}
				TBS(3,1):	5.923×10^{-3}

Tra bone

Card 2	ISØRS(2):	16	Card 3	AS(1,2):	1.42×10^{-2}
	ØRGNM(2):	TRA BØNE		AS(2,2):	-6.96×10^{-4}
	ICØM(3,2):	3		AS(3,2):	7.35×10^{-4}
				TBS(1,2):	2.852
				TBS(2,2):	6.133×10^{-2}
				TBS(3,2):	5.923×10^{-3}

Kidneys

Card 2	ISØRS(3):	6	Card 3	AS(1,3):	4.42×10^{-3}
	ØRGNM(3):	KIDNEYS		AS(2,3):	-2.16×10^{-4}
	ICØM(3,3):	3		AS(3,3):	2.28×10^{-4}
				TBS(1,3):	2.852
				TBS(2,3):	6.133×10^{-2}
				TBS(3,3):	5.923×10^{-3}

Muscle

Card 2	ISØRS(4):	12	Card 3	AS(1,4):	3.99×10^{-1}
	ØRGNM(4):	MUSCLE		AS(2,4):	-1.95×10^{-2}
	ICØM(3,4):	3		AS(3,4):	2.05×10^{-2}
				TBS(1,4):	2.852
				TBS(2,4):	6.133×10^{-2}
				TBS(3,4):	5.923×10^{-3}

Liver

Card 2	ISØRS(5): 7	Card 3	AS(1,5): 2.56 x 10^{-2}
	ØRGNM(5): LIVER		AS(2,5): -1.25 x 10^{-3}
	ICØM(3,5): 3		AS(3,5): 1.32 x 10^{-3}
			TBS(1,5): 2.852
			TBS(2,5): 6.133 x 10^{-2}
			TBS(3,5): 5.923 x 10^{-3}

Thyroid

Card 2	ISØRS(6): 22	Card 3	AS(1,6): -0.328
	ØRGNM(6): THYRØID		AS(2,6): 0.016
	ICØM(3,6): 3		AS(3,6): 0.312
			TBS(1,6): 2.852
			TBS(2,6): 6.133 x 10^{-2}
			TBS(3,6): 5.923 x 10^{-3}

Other tissue

Card 2	ISØRS(7): 23	Card 3	AS(1,7): 4.97 x 10^{-1}
	ØRGNM(7): ØTH TISS		AS(2,7): -2.43 x 10^{-2}
	ICØM(3,7): 3		AS(3,7): 2.56 x 10^{-2}
			TBS(1,7): 2.852
			TBS(2,7): 6.133 x 10^{-2}
			TBS(3,7): 5.923 x 10^{-3}

FØR XE-131M

Card 1	IS(1): 1
	IS(2): 1
	IS(3): 1
	IS(4): 1
	IS(5): 1
	IS(6): 1
	IS(7): 1

Card 2 and 3 are repeated for each organ

Cort bone

Card 2	ISØRS(1): 15	Card 3	AS(1,1): 0.0571
	ØRGNM(1): CØR BØNE		TBS(1,1): 8.319
	ICØM(4,1): 1		

Tra bone

Card 2	ISØRS(2): 16	Card 3	AS(1,2): 0.0143
	ØRGNM(2): TRA BØNE		TBS(1,2): 8.319
	ICØM(4,2): 1		

Kidneys

Card 2	ISØRS(3):	6	Card 3	AS(1,3):	4.43 x 10^{-3}
	ØRGNM(3):	KIDNEYS		TBS(1,3):	8.319
	ICØM(4,3):	1			

Muscle

Card 2	ISØRS(4):	12	Card 3	AS(1,4):	0.4
	ØRGNM(4):	MUSCLE		TBS(1,4):	8.319
	ICØM(4,4):	1			

Liver

Card 2	ISØRS(5):	7	Card 3	AS(1,5):	0.0257
	ØRGNM(5):	LIVER		TBS(1,5):	8.319
	ICØM(4,5):	1			

Thyroid

Card 2	ISØRS(6):	22	Card 3	AS(1,6):	1.
	ØRGNM(6):	THYRØID		TBS(1,6):	8.319
	ICØM(4,6):	1			

Other tissue

Card 2	ISØRS(7):	23	Card 3	AS(1,7):	0.498
	ØRGNM(7):	ØTH TISS		TBS(1,7):	8.319
	ICØM(4,7):	1			

Other Tissues Transformation

Insert 12 cards all of the form

NØTØP:	1
IØTHER(1):	15
IØTHER(2):	16
IØTHER(3):	6
IØTHER(4):	12
IØTHER(5):	7
IØTHER(6):	22

The printout for Example 2 is given for the output to unit 10. Output to unit 6 is omitted because it is so lengthy. The cumulated activity (μCi-days) are omitted for T = 10 yrs., 20 yrs., 30 yrs., 40 yrs., and 50 yrs., since these are the same as those for T = 5 yrs., (1826.25 days). Negative cumulated activities represent machine round-off and are in effect 0.

```
                              OUTPUT FOR EXAMPLE .

                         TE-131M CHAIN.  INTAKE BY INHALATION


PERCENT DEPOSITION
    N-P            T-B            P
   0.30           0.08           0.25
ISOTOPE     HALFLIFE    CLASS   LAMAB   BR.RATIO         NBRNCH      IBBNCH
TE-131M     1.250D 00     D    2.000D 00 1.000D 00          0           0
TE-131      1.736D-02     D    2.000D 00 1.800D-01          2           1
I-131       8.040D 00     D    1.380D 02 8.230D-01          2           2
                      LAMAB-S 0.0
                    LAMAB-ULI 1.180D 02
                    LAMAB-LLI 1.140D 02
XE-131M     1.190D 01     D    1.180D 02 1.440D-02          1           1




ISOTOPE     ORGAN       A'S AND TB'S
TE-131M
            COR BONE      1.830D-02    1.000D-02    7.533D-02    9.663D-05
            TRA BONE      4.570D-03    1.000D-02    7.533D-02    9.663D-05
            KIDNEYS       1.420D-03    3.400D-02    7.533D-02    3.013D-02
            MUSCLE        1.280D-01    1.000D-01    7.533D-02    3.915D-02
            LIVER         6.230D-03    5.500D-02    7.533D-02    6.794D-02
            THYROID       9.140D-05    7.533D-02
            OTH TISS      1.590D-01    7.533D-02




ISOTOPE     ORGAN       A'S AND TB'S
TE-131
            COR BONE      1.830D-02    1.000D-02    7.533D-02    9.663D-05
            TRA BONE      4.570D-03    1.000D-02    7.533D-02    9.663D-05
            KIDNEYS       1.420D-03    3.400D-02    7.533D-02    3.013D-02
            MUSCLE        1.280D-01    1.000D-01    7.533D-02    3.915D-02
            LIVER         4.230D-03    5.500D-02    7.533D-02    6.794D-02
            THYROID       9.140D-05    7.533D-02
            OTH TISS      1.590D-01    7.533D-02




ISOTOPE     ORGAN       A'S AND TB'S
I-131
            COR BONE      5.700D-02  -2.780D-03    2.980D-03    2.852D 00    6.133D-02    5.923D-03
            TRA BONE      1.420D-02  -6.960D-06    7.350D-06    2.852D 00    6.133D-02    5.923D-03
            KIDNEYS       4.420D-03  -2.160D-06    2.280D-06    2.852D 00    6.133D-02    5.923D-03
            MUSCLE        1.990D-01  -1.950D-02    2.060D-02    2.852D 00    6.133D-02    5.923D-03
            LIVER         2.560D-02  -1.250D-03    1.320D-03    2.852D 00    6.133D-02    5.923D-03
            THYROID      -3.280D-01   1.600D-02    3.120D-01    2.852D 00    6.133D-02    5.923D-03
            OTH TISS      4.970D-01  -2.430D-02    2.560D-02    2.852D 00    6.133D-02    5.923D-03




ISOTOPE     ORGAN       A'S AND TB'S
XE-131M
            COR BONE      5.710D-02    8.319D 00
            TRA BONE      1.430D-02    8.319D 00
            KIDNEYS       4.430D-03    8.319D 00
            MUSCLE        4.000D-01    8.319D 00
            LIVER         2.570D-02    8.319D 00
            THYROID       1.000D 00    8.319D 00
            OTH TISS      4.980D-01    8.319D 00
```

T = [illegible]

MICROCURIES-DAYS FROM INHALATION OF TE-131M

LUNGS

ISOTOPE	N-P	T-B	P	LYMPH	MICROCURIES ENTERING BLOOD	MICROCURIES ENTERING GI
TE-131M	4.2537D-01	2.0928D-03	1.2616D-01	1.0559D-02	3.4704D-01	1.5226D-01
TE-131	2.9248D-04	1.4739D-04	2.1923D-02	3.0159D-03	4.7303D-02	9.0499D-03
I-131	4.7129D-04	4.4237D-06	6.3205D-03	1.6075D-03	9.5595D-03	1.7110D-04
XE-131M	5.7465D-11	1.0793D-10	2.0952D-06	4.9907D-07	8.2610D-06	2.1702D-09

GI TRACT

ISOTOPE	STOMACH	S.I.	U.L.I.	L.L.I.	MICROCURIES ENTERING BLOOD
TE-131M	6.2000D-03	1.7396D-02	4.2092D-02	4.6074D-02	3.4793D-02
TE-131	4.6119D-04	1.7369D-03	7.0091D-03	8.0842D-03	6.0690D-03
I-131	2.4297D-05	1.8044D-05	3.2000D-05	3.3473D-05	9.6131D-03
XE-131M	1.1952D-09	3.4408D-13	1.4277D-08	3.2270D-08	3.9225D-08

OTHER ORGANS

ISOTOPE	COR BONE	TRA BONE	KIDNEYS	MUSCLE	LIVER	THYROID	OTH TISS
TE-131M	1.2690D-02	4.7172D-03	1.4221D-02	1.0229D-01	2.0167D-02	4.3522D-05	7.0492D-02
TE-131	2.1205D-03	1.2175D-03	2.9399D-03	1.0537D-02	5.1641D-03	7.3027D-06	1.2773D-02
I-131	4.9926D-06	1.7164D-06	5.3631D-05	4.0240D-03	3.0949D-06	9.9100D-03	6.0059D-03
XE-131M	1.8773D-07	1.5994D-08	1.1151D-09	1.0040D-06	6.6689D-08	2.5171D-06	1.2535D-06

AFTER TRANSFORMATION

	COR BONE	TRA BONE	KIDNEYS	MUSCLE	LIVER	THYROID	T. BODY
TE-131M	4.7880D-03	4.6956D-03	1.5594D-02	4.5690D-02	2.4520D-02	9.3527D-08	1.4151D-01
TE-131	4.5520D-04	8.5116D-05	2.0263D-03	9.2007D-03	4.4447D-03	1.6402D-08	2.5642D-02
I-131	3.1032D-07	-5.5928D-07	1.7676D-08	1.3480D-06	-5.4160D-07	9.9073D-03	1.2057D-02
XE-131M	-6.4477D-11	4.6301D-11	6.6070D-12	2.8473D-10	-1.7315D-11	2.5164D-06	2.5164D-06

T = [illegible]

MICROCURIES-DAYS FROM INHALATION OF TE-131M

LUNGS

ISOTOPE	N-P	T-B	P	LYMPH	MICROCURIES ENTERING BLOOD	MICROCURIES ENTERING GI
TE-131M	4.2937D-01	2.1827D-03	1.2881D-01	1.9405D-02	3.9257D-01	1.5226D-01
TE-131	2.9248D-04	1.6782D-04	2.2608D-02	3.3515D-03	4.8310D-02	9.0499D-03
I-131	4.7129D-04	4.4105D-06	7.4776D-03	2.4838D-03	1.2050D-02	1.7112D-04
XE-131M	5.7465D-11	1.0446D-10	4.3723D-06	2.2831D-06	8.0199D-06	2.1750D-09

GI TRACT

ISOTOPE	STOMACH	S.I.	U.L.I.	L.L.I.	MICROCURIES ENTERING BLOOD
TE-131M	6.2000D-03	1.7397D-02	4.3679D-02	5.1632D-02	3.4793D-02
TE-131	4.5120D-04	1.7369D-03	7.9163D-03	9.4278D-03	6.0699D-03
I-131	2.9238D-05	1.6085D-05	3.3256D-05	3.9213D-05	1.0310D-02
XE-131M	1.0956D-09	3.4437D-13	1.5859D-08	5.8766D-08	3.9228D-08

OTHER ORGANS

ISOTOPE	COR BONE	TRA BONE	KIDNEYS	MUSCLE	LIVER	THYROID	OTH TISS
TE-131M	1.9703D-02	1.0568D-02	2.5296D-02	1.5414D-01	4.2660D-02	6.1076D-05	1.0625D-01
TE-131	3.5069D-03	1.9206D-03	4.5964D-03	2.8366D-02	7.7497D-03	1.1095D-05	1.9309D-02
I-131	1.1582D-05	2.0763D-06	9.9417D-05	8.0757D-03	5.1790D-06	6.8501D-02	1.0083D-02
XE-131M	5.5805D-07	1.3996D-07	4.3350D-08	3.9149D-06	2.5153D-07	9.7873D-06	4.8781D-06

AFTER TRANSFORMATION

	COR BONE	TRA BONE	KIDNEYS	MUSCLE	LIVER	THYROID	T. BODY
TE-131M	7.5553D-03	7.5215D-03	2.4351D-02	7.0623D-02	1.7176D-02	1.3605D-07	2.1329D-01
TE-131	1.3730D-03	1.1669D-03	4.4240D-03	1.2860D-02	6.7534D-03	2.4782D-08	3.0784D-02
I-131	2.0491D-06	-5.9366D-07	1.2920D-07	1.0977D-05	-4.7224D-07	6.0575D-02	2.0162D-02
XE-131M	-2.5880D-10	1.8001D-10	2.6023D-11	1.1246D-09	-6.7513D-11	9.7885D-06	9.7885D-06

```
T = 3.0000D 01

MICROCURIES-DAYS FROM INHALATION OF TE-131M

LUNGS                                                                  MICROCURIES ENTERING
                                                                          BLOOD        GI
ISOTOPE       N-P          T-B           P           LYMPH
TE-131M   4.23370-03  2.04270-03  1.28810-01  1.04020-02             3.92570-01  1.52260-01
TE-131    2.92440-08  1.47620-08  2.28080-02  3.35150-03             4.83100-02  9.04990-03
I-131     4.71290-06  4.43030-06  7.47060-03  2.04550-03             1.20540-02  1.71120-06
XE-131M   5.74450-11  1.24490-10  4.37770-06  2.29690-06             4.04200-06  2.17540-09


GI TRACT                                                               MICROCURIES ENTERING
                                                                          BLOOD
ISOTOPE    STOMACH        S.I.        U.L.I.       L.L.I.
TE-131M   6.20090-03  1.73970-02  4.30790-02  5.16360-02             3.47930-02
TE-131    9.51290-08  3.03490-03  7.91630-03  9.42450-03             6.06990-03
I-131     2.92400-05  1.60450-05  3.32540-05  3.92160-05             1.03190-02
XE-131M   1.79540-09  1.44130-10  1.58500-08  5.49700-08             3.92240-09


OTHER ORGANS

ISOTOPE     CCR BONE     TRA BONE     KIDNEYS      MUSCLE       LIVER       THYROID     OTH TISS
TE-131M   2.31230-02  1.04070-02  2.50170-02  1.58840-01  4.33460-02  6.20170-05  1.07890-01
TE-131    3.45420-03  1.96390-03  4.69150-03  2.00590-02  7.47440-03  1.12660-05  1.95990-02
I-131     1.44310-03  6.09550-06  1.27290-06  1.15020-02  7.37300-04  1.69410-01  1.42040-02
XE-131M   1.13430-06  2.99490-07  9.20100-09  4.36020-06  4.30430-07  2.09500-05  1.34330-05

AFTER TRANSFORMATION
            CCR BONE     TRA BONE     KIDNEYS      MUSCLE       LIVER       THYROID     T. BODY
TE-131M   7.74710-03  7.71270-03  2.08580-02  7.22090-02  3.77770-02  1.38540-07  2.16570-01
TE-131    1.40900-03  1.43190-03  4.51730-03  1.31210-02  6.06310-03  2.51640-08  1.91440-02
I-131     4.32530-06 -1.47770-07  2.44350-07  2.47310-05 -2.42400-07  1.49000-01  2.34420-02
XE-131M  -5.58130-10  3.95210-10  5.65410-11  2.46690-09 -1.44700-10  2.09440-05  2.09440-05
```

```
T = 4.0000D 01

MICROCURIES-DAYS FROM INHALATION OF TE-131M

LUNGS                                                                  MICROCURIES ENTERING
                                                                          BLOOD        GI
ISOTOPE       N-P          T-B           P           LYMPH
TE-131M   4.23370-03  2.04270-03  1.28810-01  1.04020-02             3.92570-01  1.52260-01
TE-131    2.92440-08  1.47620-08  2.28080-02  3.35150-03             4.83100-02  9.04990-03
I-131     4.71290-06  4.43030-06  7.47060-03  2.04550-03             1.20580-02  1.71120-06
XE-131M   5.74450-11  1.24490-10  4.37770-06  2.29690-06             8.36200-06  2.17540-09


GI TRACT                                                               MICROCURIES ENTERING
                                                                          BLOOD
ISOTOPE    STOMACH        S.I.        U.L.I.       L.L.I.
TE-131M   6.20090-03  1.73970-02  4.30790-02  5.16360-02             3.47930-02
TE-131    9.51290-08  3.03490-03  7.91630-03  9.42450-03             6.06990-03
I-131     2.92900-05  1.60450-05  3.32500-05  3.92160-05             1.03190-02
XE-131M   1.39540-09  1.44130-10  1.58500-08  5.49700-08             3.92200-09


OTHER ORGANS

ISOTOPE     CCR BONE     TRA BONE     KIDNEYS      MUSCLE       LIVER       THYROID     OTH TISS
TE-131M   2.01230-02  1.04070-02  2.44170-02  1.54040-01  4.33460-02  6.20170-05  1.07890-01
TE-131    3.65620-03  1.96390-03  4.69150-03  2.00590-02  7.07400-03  1.12660-05  1.95990-02
I-131     1.74720-03  4.35510-06  1.35300-06  1.22270-02  7.93910-04  1.82100-01  1.51920-02
XE-131M   1.29130-06  3.20490-07  9.94090-08  8.57590-06  5.76700-07  2.24400-05  1.11750-05

AFTER TRANSFORMATION
            CCR BONE     TRA BONE     KIDNEYS      MUSCLE       LIVER       THYROID     T. BODY
TE-131M   7.74710-03  7.71270-03  2.08580-02  7.22090-02  3.77770-02  1.38540-07  2.16570-01
TE-131    1.40800-03  1.40180-03  4.51730-03  1.31210-02  6.06310-03  2.51640-08  1.93040-02
I-131     8.50490-06 -1.50740-07  2.81240-07  2.85740-05 -2.78470-07  1.82190-01  3.04970-02
XE-131M  -5.93410-10  4.12510-10  6.06030-11  2.57650-09 -1.58950-10  2.24310-05  2.24310-05
```

```
T = [illegible]

MICROCURIES-DAYS FROM INHALATION OF TE-131M

LUNGS                                                             MICROCURIES ENTERING
                                                                    BLOOD        GI
ISOTOPE        N-P         T-B          P          LYMPH
TE-131M    4.29370D-01 2.60270D-01 1.20810D-01 1.86620D-02      3.92570D-01 1.52260D-01
TE-131     2.62480D-08 1.47620D-08 2.26080D-02 3.35150D-03      4.63100D-02 9.94990D-03
I-131      4.71290D-06 4.43100D-06 7.47060D-03 2.40550D-03      1.20560D-02 1.71120D-04
XE-131M    5.76650D-11 1.04480D-10 4.37770D-08 2.29490D-06      8.06200D-06 2.17500D-09


GI TRACT                                                          MICROCURIES ENTERING
                                                                    BLOOD
ISOTOPE     STOMACH       S.I.       U.L.I.      L.L.I.
TE-131M    6.22060D-01 1.73970D-02 6.38790D-02 5.16340D-02      3.47930D-02
TE-131     6.51200D-08 1.03690D-03 7.91630D-03 9.02650D-03      5.36990D-03
I-131      2.42900D-05 1.83050D-05 1.32540D-05 3.92160D-05      1.03190D-02
XE-131M    1.30560D-09 1.64100D-10 1.58500D-09 5.09700D-08      3.92200D-08


OTHER ORGANS

ISOTOPE      COR BONE    TRA BONE     KIDNEYS     MUSCLE       LIVER      THYROID     OTH TISS
TE-131M    2.01230D-02 1.08070D-02 2.50170D-02 1.50840D-01 8.33460D-02 6.20170D-05 1.67090D-01
TE-131     1.65620D-03 1.96390D-03 4.69150D-03 2.00590D-02 7.07460D-03 1.12660D-05 1.95990D-02
I-131      1.75510D-03 8.37490D-08 1.35950D-08 1.22830D-02 7.07400D-08 1.03050D-01 1.52610D-02
XE-131M    1.26710D-06 1.22350D-07 9.90660D-08 9.01670D-06 5.79320D-07 2.25420D-05 1.12260D-05

AFTER TRANSFORMATION
             COR BONE    TRA BONE     KIDNEYS     MUSCLE       LIVER      THYROID      T. BODY
TE-131M    7.74710D-01 7.71270D-01 2.40580D-02 7.22090D-02 3.77770D-02 1.38540D-07 2.16570D-01
TE-131     1.43800D-03 1.40180D-03 4.51730D-03 1.31210D-02 6.06310D-03 2.51620D-08 3.93440D-02
I-131      6.51470D-06 -1.56250D-07 2.42390D-07 2.06910D-05 -2.02280D-07 1.83040D-01 3.06350D-02
XE-131M   -5.96300D-10 8.16810D-10 6.08620D-11 2.50640D-09 -1.55670D-10 2.25350D-05 2.25350D-05
```

```
T = 3.65E+02 D

MICROCURIES-DAYS FROM INHALATION OF TE-131M

LUNGS                                                             MICROCURIES ENTERING
                                                                    BLOOD        GI
ISOTOPE        N-P         T-B          P          LYMPH
TE-131M    4.29370D-01 2.60270D-01 1.40810D-01 1.86620D-02      3.92570D-01 1.52260D-01
TE-131     2.62480D-08 1.47620D-08 2.26080D-02 3.35150D-03      4.63100D-02 9.94990D-03
I-131      4.71290D-06 4.43100D-06 7.47060D-03 2.40550D-03      1.20560D-02 1.71120D-04
XE-131M    5.76650D-11 1.04480D-10 4.37770D-08 2.29490D-06      9.04200D-06 2.17500D-09


GI TRACT                                                          MICROCURIES ENTERING
                                                                    BLOOD
ISOTOPE     STOMACH       S.I.       U.L.I.      L.L.I.
TE-131M    6.22060D-01 1.73970D-02 6.38790D-02 5.16340D-02      3.47930D-02
TE-131     6.51200D-08 1.03690D-03 7.91630D-03 9.02650D-03      6.06990D-03
I-131      2.42900D-05 1.83050D-05 1.32540D-05 3.92160D-05      1.03190D-02
XE-131M    1.30560D-09 1.64100D-10 1.58500D-09 5.09700D-08      3.92200D-08


OTHER ORGANS

ISOTOPE      COR BONE    TRA BONE     KIDNEYS     MUSCLE       LIVER      THYROID     OTH TISS
TE-131M    2.01230D-02 1.08070D-02 2.50170D-02 1.50840D-01 8.33460D-02 6.20170D-05 1.67090D-01
TE-131     1.65620D-03 1.96390D-03 4.69150D-03 2.00590D-02 7.07400D-03 1.12660D-05 1.95990D-02
I-131      1.75510D-03 8.37490D-08 1.35950D-08 1.22830D-02 7.07400D-08 1.03050D-01 1.52610D-02
XE-131M    1.26710D-06 1.22350D-07 9.90660D-08 9.01670D-06 5.79320D-07 2.25420D-05 1.12260D-05

AFTER TRANSFORMATION
             COR BONE    TRA BONE     KIDNEYS     MUSCLE       LIVER      THYROID      T. BODY
TE-131M    7.74710D-01 7.71270D-01 2.40580D-02 7.22090D-02 3.77770D-02 1.38540D-07 2.16570D-01
TE-131     1.43800D-03 1.40180D-03 4.51730D-03 1.31210D-02 6.06310D-03 2.51620D-08 3.93440D-02
I-131      6.51470D-06 -1.56250D-07 2.42390D-07 2.06910D-05 -2.02280D-07 1.83040D-01 3.06350D-02
XE-131M   -5.96300D-10 8.16810D-10 6.08620D-11 2.50640D-09 -1.55670D-10 2.25350D-05 2.25350D-05
```

```
T = 1.02E+05 [illegible]

MICROCURIES-DAYS FROM INHALATION OF TE-131M

LUNGS                                                                    MICROCURIES ENTERING
                                                                           BLOOD         GI
ISOTOPE        N-P          T-B           P          LYMPH
TE-131M    8.25972-03  2.28275-01  1.20018-01  1.04082-02           3.92572-01  1.52200-01
TE-131     2.42603-06  1.47629-04  2.20380-02  3.35190-03           4.83100-02  9.04990-03
I-131      4.71299-06  6.43089-06  7.47003-03  2.40558-03           1.20540-02  1.71125-04
XE-131M    5.70659-11  1.04448-12  6.37779-06  2.25698-06           8.04208-06  2.17580-09


GI TRACT                                                                 MICROCURIES ENTERING
                                                                           BLOOD
ISOTOPE      STOMACH        S.I.        U.L.I.      L.L.I.
TE-131M    6.23000-03  3.73072-02  4.34790-02  5.16360-02           3.47932-02
TE-131     2.51200-06  3.23040-03  7.91630-03  9.82059-03           6.06903-03
I-131      2.92048-05  1.93652-05  3.32503-05  3.52100-05           1.23193-02
XE-131M    1.70520-09  3.44130-10  1.50503-08  5.49702-10           3.42203-08


OTHER ORGANS

ISOTOPE     CORT BONE    TRA BONE     KIDNEYS      MUSCLE       LIVER       THYROID     OTH TISS
TE-131M    2.31230-02  1.20270 02  2.50170-02  1.50002-01  4.13440-02  4.23170-05  1.07893-01
TE-131     3.45622-03  1.96393-03  4.09159-03  2.00590-02  7.47000-03  3.12663-05  1.45950-02
I-131      1.79510-03  4.37690-06  1.35950-04  1.22030-02  7.67400-06  3.03652-01  1.52613-02
XE-131M    1.26717-06  3.22359-07  9.90603-08  9.61670-06  5.79328-07  2.25423-05  1.12202-05

AFTER TRANSFORMATION
            CORT BONE    TRA BONE     KIDNEYS      MUSCLE       LIVER       THYROID      T. BODY
TE-131M    7.74710-03  7.71270-03  2.00500-02  7.22048-02  3.77778-02  1.36542-07  2.16570-01
TE-131     1.45935-03  1.40195-03  4.51773-03  1.31210-02  6.96312-03  2.57622-06  3.43440-02
I-131      4.51562-06 -1.56285-07  2.07410-07  2.06580-05 -2.41870-07  1.33040-01  3.06350-02
XE-131M   -5.66383-09  4.74778-12  6.30560-11  2.50780-09 -1.55719-13  3.25153-05  2.25153-05
```

Example 3

Radionuclide - I-132

Half life - 2.38 hours

Mode of intake - ingestion

$f_1 = 0.95$

Source organs

1. GIT

2. Thyroid: $R(t) = -0.328e^{-\frac{.693}{0.24265}t} + 0.016e^{-\frac{.693}{11.3235}t}$

$$+ 0.312e^{-\frac{.693}{117.378}t}$$

3. Total body: $R(t) = 0.9973e^{-\frac{.693}{0.24265}t} - 0.04873e^{-\frac{.693}{11.3235}t}$

$$+ 0.05143e^{-\frac{.693}{117.378}t}$$

Print cumulated activity (μCi-days) for same times as Example 1.

A description of the input cards is given below. Follow Table 5.1 and Table B.3 to fully understand the instructions.

<u>Information Concerning the Differential Equations</u>

<u>Card 1</u> NØ: 4 GIT
6 Thyroid and Total body
—
10

TLAST, H1, EPS same as Example 1.

<u>Card 2...Card 9</u> same as Example 1.

Table B.3. I-130; Intake by Ingestion

```
   1.0    1.0202.5
    7
        2.
        7.
       29.       30.     1
      100.
    295.25
   1926.25
   3692.5    3692.5    4
         1                                        2      1      1
1.122        0.0002          114.
227.47E10  3
  -0.329     0.016     0.312    2.05697    6.12E-2   5.004E-3
227.009V   3
   0.0073  -0.00073  0.06103   2.05697    6.12E-2   5.004E-3

   12 blank cards
```

General Information for Chain of Radionuclides

Card 1
NISØ: 4
D3:
D4: } not applicable
D5:
NØRG: 2
IØPT: 1
NØRAL: 1

Card 2
TAG(1): I-132
TR(1): 0.0992
ICLASS(1): not applicable
LAMAB(1,2): 114.
P(1), NBRNCH(1), IBRNCH(1): not applicable
ILAB: 0

Card 3 omitted because ILAB=0.

Description of Retention

Card 1 omitted for first nuclide in the chain.

Card 2 and 3 repeated for each organ.

Thyroid

Card 2
ISØRS(1): 22
ØRGNM(1): THYRØID
ICØM(1,1): 3

Card 3
AS(1,1): -0.328
AS(2,1): 0.016
AS(3,1): 0.312
TBS(1,1): 2.85597
TBS(2,1): 6.12 x 10^{-2}
TBS(3,1): 5.904 x 10^{-3}

Total body

Card 2
ISØRS(2): 23
ØRGNM(2): T. BØDY
ICØM(1,2): 3

Card 3
AS(1,2): 0.9973
AS(2,2): -0.04873
AS(3,2): 0.05143
TBS(1,2): 2.85597
TBS(2,2): 6.12 x 10^{-2}
TBS(3,2): 5.904 x 10^{-3}

Other Tissues Transformation

Insert 12 blank cards.

Printout to both unit 6 and unit 10, is included for Example 3. Printout for cumulated activity (μCi-days) for T greater than 7 days has been omitted since there is no change in the cumulated activity after 7 days.

```
                                                        TEST EXAMPLE 3
                                                       STAGE BY INGESTION



NO. PRINT INTERVALS DESIRED = 7   NO. READ = 7
POINTS SELECTED IN PRINT INTERVAL = T+J*DT   WHERE J=0,.....,N
       T                   DT                  N
 2.0000000000000D 00   0.0                     0
 7.0000000000000D 00   0.0                     0
 1.0000000000000D 01   1.0000000000000D 01     1
 1.0600000000000D 02   0.0                     0
 4.6525000000000D 02   0.0                     0
 1.0262500000000D 03   0.0                     0
 1.6525000000000D 03   1.6525000000000D 03     6

NEQ  13
NPE=21
TO= 0.0
TLAST= 1.6262500000000D 03
HO= 1.0000000000000D-07
EPS=1.0000D-04
```

```
MATRIX A PRINTED BY ROWS

 -0.2461000 02   0.0            0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.2403670 02  -0.1206030 03   0.0            0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.6000000 01  -0.1006150 01   0.0            0.0            0.0            0.0            0.0
  0.0            0.0

  0.0            0.0            0.1006150 01  -0.1000000 01   0.0            0.0            0.0            0.0
  0.0            0.0

  0.0           -0.1790200 02   0.0            0.0           -0.2055970 01   0.0            0.0            0.0
  0.0            0.0

  0.0            0.1020000 01   0.0            0.0            0.0           -0.6120000-01   0.0            0.0
  0.0            0.0

  0.0            0.3556900 02   0.0            0.0            0.0            0.0            0.5900600-02   0.0
  0.0            0.0

  0.0            0.1116920 03   0.0            0.0            0.0            0.0            0.0           -0.2055970 01
  0.0            0.0

  0.0           -0.5555220 01   0.0            0.0            0.0            0.0            0.0            0.0
 -0.6120010-01   0.0

  0.0            0.5861020 01   0.0            0.0            0.0            0.0            0.0            0.0
  0.0           -0.5900200-02
 ENTRIES OF A  ROW  COL
-0.2400000D 02    1    1
 0.2400000D 02    2    1
-0.1200000D 03    2    2
 0.6000000D 01    3    2
-0.1846150D 01    3    3
 0.1846150D 01    4    3
-0.1600000D 01    4    4
 0.0              5    1
-0.3739200D 02    5    2
 0.0              5    3
 0.0              5    4
-0.2455970D 01    5    5
 0.0              6    1
 0.1024000D 01    6    2
 0.0              6    3
 0.0              6    4
-0.6120000D-01    6    6
 0.0              7    1
 0.3556600D 02    7    2
 0.0              7    3
 0.0              7    4
-0.5914000D-02    7    7
 0.0              8    1
 0.1113692D 03    8    2
 0.0              8    3
 0.0              8    4
-0.2455970D 01    8    8
 0.0              9    1
-0.5555220D 01    9    2
 0.0              9    3
 0.0              9    4
-0.6120000D-01    9    9
 0.0             10    1
 0.5963020D 03   10    2
 0.0             10    3
 0.0             10    4
-0.5904000D-02   10   10


KFLAG = -1 FROM STIFF AT T =     0.0
 ERROR TEST FAILED WITH ABS(H) = HMIN

 H HAS BEEN REDUCED TO    0.9999999999999999D-08  AND STEP WILL BE RETRIED



T= 2.0000000000D 00    Y FOLLOWS
  3.2271211637D-02   6.0991031996D-03   4.1426961506D-03   9.5752901920D-04  -2.3168726115D-02   1.5782969913D-03
  3.1020117279D-02   7.0845681934D-02  -4.8069077420D-03   5.1133780480D-03

T= 7.0000000000D 00    Y FOLLOWS
  3.2271211655D-02   6.0991032039D-03   4.1426979199D-03   9.5752377081D-04  -2.3168725646D-02   1.5783037215D-03
  3.1020279836D-02   7.0845680569D-02  -4.8069212716D-03   5.1133789696D-03

T= 3.0000000000D 01    Y FOLLOWS
  3.2271211629D-02   6.0991031977D-03   4.1426973679D-03   9.5751867389D-04  -2.3168726151D-02   1.5783007060D-03
  3.1020217228D-02   7.0845682085D-02  -4.8069120877D-03   5.1133646500D-03

T= 6.0000000000D 01    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 1.8000000000D 02    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 3.6525000000D 02    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 1.8262500000D 03    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 3.6525000000D 03    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 7.3050000000D 03    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 1.0957500000D 04    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 1.4610000000D 04    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

T= 1.8262500000D 04    Y FOLLOWS
  3.2271211629D-02   6.0991031973D-03   4.1426973680D-03   9.5751867507D-04  -2.3168726151D-02   1.5783007069D-03
  3.1020217223D-02   7.0845682085D-02  -4.8069120905D-03   5.1133646532D-03

KFLAG
```

```
ISOTOPE     HALFLIFE     CLASS   LAMBD   BR.RATIO         NBRNCH     IBRNCH
I-132       9.520D-02      D   1.130D 02 1.000D 00           0          0



ISOTOPE     ORGAN       A'S AND TB'S
I-132
          THYROID       -3.280D-01   1.600D-02   3.120D-01   2.856D 00   6.120D-02   5.908D-03
          T.BODY         9.973D-01  -8.873D-02   5.143D-02   2.856D 00   6.120D-02   5.908D-03




T = 2.0000D 00

MICROCURIE-DAYS FROM INGESTION OF I-132


GI TRACT                                                                 MICROCURIES ENTERING
                                                                           BLOOD
ISOTOPE     STOMACH        S.I.         U.L.I.        L.L.I.
I-132      3.2271D-02  6.0991D-03  8.1827D-03  9.5752D-04              6.9530D-01


OTHER ORGANS

ISOTOPE      THYROID      T.BODY
I-132      9.4299D-03  7.0752D-02




T = 7.0000D 00

MICROCURIE-DAYS FROM INGESTION OF I-132


GI TRACT                                                                 MICROCURIES ENTERING
                                                                           BLOOD
ISOTOPE     STOMACH        S.I.         U.L.I.        L.L.I.
I-132      3.2271D-02  6.0991D-03  8.1827D-03  9.5752D-04              6.9530D-01


OTHER ORGANS

ISOTOPE      THYROID      T.BODY
I-132      9.4299D-03  7.0752D-02
```

APPENDIX C

JOB CONTROL LANGUAGE STATEMENTS

The following Job Control Language (JCL) cards have been used to run TIMED on the IBM System/360 machines. After debugging we have avoided the costly compilation steps by obtaining hex decks and bypassing the compilation. If the user wishes to obtain the printout on unit 6, omit the DD card associated with it. If output to unit 7, 14, or 10 is to be suppressed, replace the DD parameters with DUMMY as was done in the case of unit 6.

```
//  Job card
//  EXEC  ASMFC,PARM.ASM='LOAD'
//ASM.SYSIN  DD  *

       AXED  source  deck

/*
//  EXEC  ASMFC,PARM.ASM='LOAD'
//ASM.SYSIN  DD  *

       YPAY  source  deck

/*
//  EXEC  FORTHCLG,PARM.FORT='XREF',
//  PARM.GO='SO=S1,SI=S0,EU=-1,DUMP=1',
//  REGION.GO=320K,
//FORT.SYSIN  DD  *

       FORTRAN  source  decks

/*
//GO.FT06F001  DD  DUMMY           SUPPRESS  OUTPUT  ON  UNIT  6
//GO.FT07F001  DD  SYSOUT=B          PUNCHED  OUTPUT.
//GO.FT14F001  DD  UNIT=SYSDA,DSN=&&UCIDAY,DISP=(NEW,PASS),
//  DCB=(RECFM=VBS,LRECL=150,BLKSIZE=1504),
//  SPACE=(1504,(10,10),RLSE)      DISK DATA SET
//GO.FT10F001  DD  SYSOUT=A,DCB=(RECFM=FA,BLKSIZE=133)
//GO.FT05F001  DD  *

     DATA DECK

/*
//
```