PDR

EGG-CDD-5728

January 1982

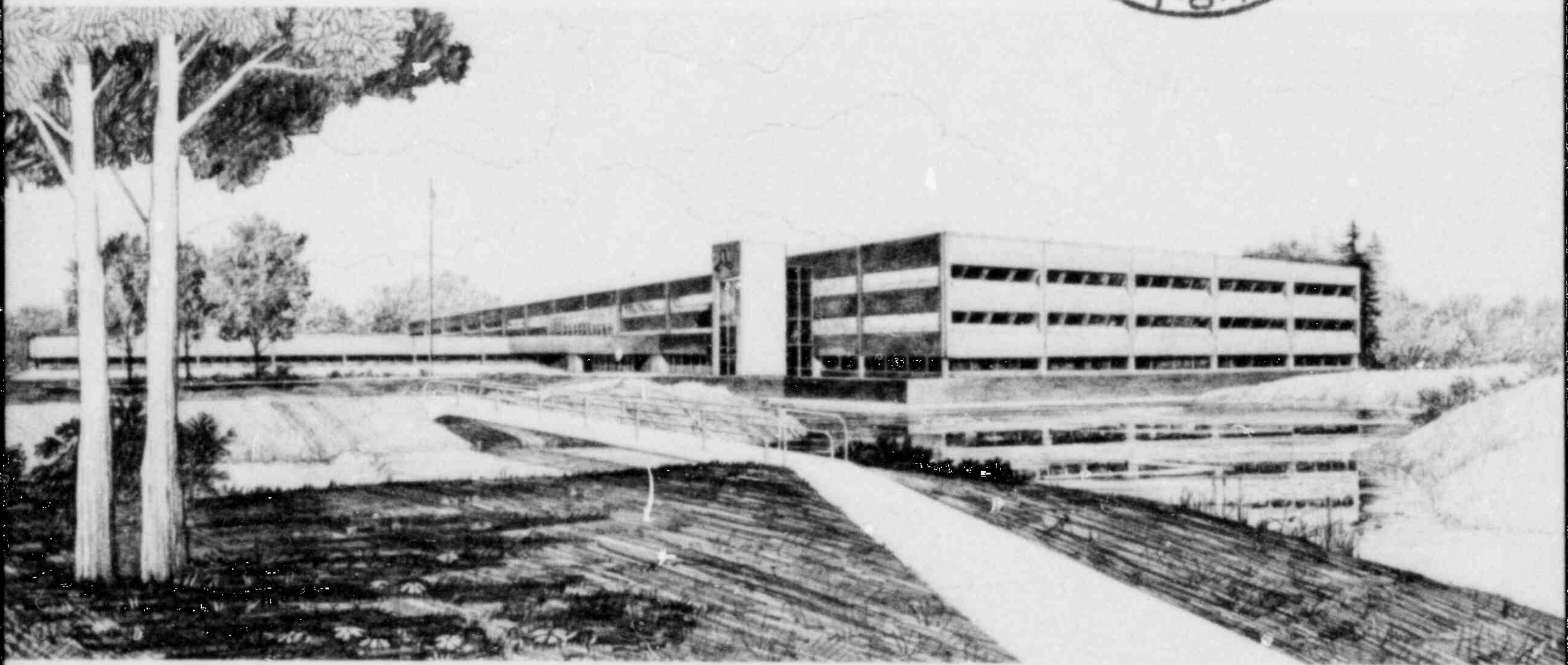
DEBRIS BED CHARACTERIZATION AND THERMAL BEHAVIOR MODELS FOR SCDAP

NRC Research and/or Technical Assistance Report

T. S. Hsieh

RECEIVED
MAR 02 1982
US NUCLEAR REGULATORY COMMISSION
DOCUMENT MANAGEMENT BR
TIDC

U.S. Department of Energy
Idaho Operations Office • Idaho National Engineering Laboratory



This is an informal report intended for use as a preliminary or working document

Prepared for
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555
Under DOE Contract No. DE-AC07-76ID01570
FIN. No. A6350

EG&G Idaho

EG&G Idaho, Inc

FORM EG&G-398
(Rev. 11-81)

INTERIM REPORT

Accession No. ______

Report No. EGG-CDD-5728

Contract Program or Project Title: Debris Bed Characterization and Thermal Behavior Models For SCDAP

Subject of this Document: Debris Bed Characterization and Thermal Behavior Models For SCDAP

Type of Document: Informal Report

Author(s): T. S. Hsieh

Date of Document: January 1982

Responsible NRC/DOE Individual and NRC/DOE Office or Division: G. P. Marino, NRC-RES

EG&G Idaho, Inc.
Idaho Falls, Idaho 83415

Prepared for the
U.S. Nuclear Regulatory Commission
Washington, D.C.
Under DOE Contract No. DE-AC07-76ID01570
NRC FIN No. A6360

INTERIM REPORT

EGG-CDD-5728
January 1982

DEBRIS BED CHARACTERIZATION AND THERMAL BEHAVIOR MODELS FOR SCDAP

T. S. Hsieh

Published January 1982

EG&G Idaho, Inc.
Idaho Falls, Idaho 83415

Prepared for the
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555
Under DOE Contract No. DE-AC07-76ID01570
FIN No. A6360

ACKNOWLEDGMENTS

The author gratefully acknowledges the support provided by G. H. Beers in programming the models described in this report.

CONTENTS

FIGURES

TABLES

DEBRIS BED CHARACTERIZATION AND THERMAL BEHAVIOR MODELS FOR SCDAP

1. INTRODUCTION

During a severe accident in a light water reactor (LWR), two general processes, either individually or in combination, are expected to result in disruption of the core. The first process is the progressive liquefaction of the core components and the subsequent redistribution and solidification of the liquefied materials. This process may result in a debris region of frozen masses. The second process is the extensive fragmentation of components along a coolant quench front due to rapid, quench-induced thermal shock of embritted components. This process may result in a debris region of loosely bound fragments.

The above two processes may drastically change the geometry of the core. For example, fragmentation of a 1-m length of the fuel rods in a pressurized water reactor fuel assembly into particles with an average diameter of 500 μm would increase the contact surface between fuel and coolant by a factor of 50. Also, coolant pressure drop across a 0.5-m thick porous debris bed with a coolant flow rate of 0.5 $kg/m^2 \cdot s$ and a specific surface area exposed to coolant of 8×10^5 m^{-1} would be a factor of 10^5 greater than that of an intact (rod-like) bundle. Obviously, the constitutive relations (loss coefficients, heat fluxes, etc.) for models normally used to analyze core thermal-hydraulic behavior must be modified to analyze disrupted core regions with such significantly different characteristics.

The Severe Core Damage Analysis Package (SCDAP) computer code[1] is being developed to analyze severe disruption of LWR cores. The initial version of SCDAP will analyze disruption of only a single fuel bundle. The models which have been developed for SCDAP to describe debris region characteristics and thermal behavior are described in this report. The models treat both conglomerate, or cohesive, debris beds and rubble debris beds which are formed by the liquefaction/redistribution/solidification and fragmentation processes, respectively. The models serve the following functions:

1. They define debris bed characteristics which include bed porosity, bed height, equivalent particle diameter, coolant pressure drop across the bed, and whether a rubble bed is in a packed or fluidized state.

2. They calculate debris bed and coolant temperature distributions and the state of bed coolability.

3. They describe propagation of a molting front within a debris bed.

Because of the paucity of data on debris beds for LWRs, the models described herein are considered preliminary in nature. The models wil be assessed when appropriate data become available and will be refined accordingly for incorporation in a latter version of SCDAP.

Section 2 describes the development of the debris region characterization and thermal behavior models. Section 3 describes the structure of the models and implementation of the models in SCDAP. Lastly, Section 4 presents results of the acceptance testing of the models.

2. MODEL DEVELOPMENT

The debris bed characterization and behavior models described herein are, in general, based on LWR and Liquid Metal Fast Breeder Reactor (LMFBR) experimental data, porous body thermal-hydraulic analysis, fluidized bed thermal-hydraulic analysis, and the Post Accident Heat Removal (PAHR) dryout heat flux analysis performed for LMFBRs. Because of the lack of an appropriate data base for developing models specifically for use in LWR severe accident analysis, the following assumptions were required during development of the models described herein. The assumptions are:

1. Debris bed is homogenous and consists of particles which can be effectively treated as spheres.

2. Within a debris bed, coolant flow is homogeneous with perfect mixing of the liquid and vapor phases.

3. Melting of material within a debris bed is an equilibrium process such that no superheating occurs until all of the debris has melted. The melting proceeds with infinite axial heat conduction.

Because the models assume homogenous flow conditions within a debris bed, some additional modeling considerations will be involved when the models are interfaced with the non-homogeneous bundle thermal-hydraulic models to be included in SCDAP.

The debris bed characterization models are described first. Then, the debris bed thermal models are described. Finally, a discussion of the range of applicability of the several correlations used for the model development is given.

2.1 Debris Bed Characterization

The models described in this section characterize both the cohesive and rubble debris beds for subsequent thermal analysis by the models described in Section 2.2.

2.1.1 Cohesive Debris Bed

Formation of a cohesive debris region is considered by two sets of models in SCDAP. Component liquefaction and liquefied material flow and solidification are analyzed with the LIQSOL models.[2] The debris transition and propagation models[1] consider the damage state for each component and determine when a bundle region has been sufficiently disrupted so that a cohesive or rubble debris region analysis is more appropriate than an intact bundle region analysis. When the criteria for transition to a cohesive debris region analysis are satisfied, the debris transition and propagation models calculate the following information:

1. Cohesive debris region height and elevation with respect to the fuel bundle
2. The masses of the individual constituents of the debris bed (UO_2, Zr, ZrO_2, etc.)
3. Surface area of debris exposed to coolant per unit volume of solid material in the region.

These data are used to characterize the cohesive debris bed for subsequent thermal behavior analysis.

Assuming that all available material in the cohesive debris region forms a homogeneous, porous body, the zero-porosity volume of the debris, V_c, is calculated as

$$V_c = \frac{M_{UO_2}}{\rho_{UO_2}} + \frac{M_{ZrO_2}}{\rho_{ZrO_2}} + \frac{M_{Zr}}{\rho_{Zr}} + \frac{M_{St}}{\rho_{St}} + \frac{M_{Ab}}{\rho_{Ab}} \qquad (1)$$

where

V_c = zero-porosity volume of cohesive debris bed (m^3)

M_X = mass of constituent X within the cohesive debris region (kg)

ρ_X = density of constituent X within the cohesive debris region (kg/m^3)

X = constituent material: UO_2, Zr, ZrO_2, St (structural material), and Ab (control rod material).

Using V_c, the porosity of the debris bed is calculated as

$$\varepsilon_c = 1 - \frac{V_c}{A_c H_c} \tag{2}$$

where

ε_c = porosity of cohesive debris bed

A_c = cross-sectional area of disrupted bundle region (m^2)

H_c = height of cohesive debris region (m).

Coolant pressure drop across a cohesive debris bed is calculated using Ergun's correlation based on a fixed-bed pressure drop analysis.[3,4] The correlation used is

$$\frac{\Delta P_c}{u_o H_c} = \alpha_c \mu + \beta_c G \tag{3}$$

where

ΔP_c = pressure drop across the bed (Pa)

u_o = superficial velocity of the coolant measured on an empty tube basis (m/s)

μ = coolant viscosity (kg/m•s)

G = coolant mass flux (kg/m^2•s)

$$\alpha_c = 150 \frac{(1 - \varepsilon_c)^2}{\varepsilon_c^3} \frac{1}{D_c^2}$$

$$\beta_c = 1.75 \frac{1 - \varepsilon_c}{\varepsilon_c^3} \frac{1}{D_c} \; .$$

The particle diameter in Equation (3) is usually obtained by direct measurement of particle sizes. However, for a cohesive debris bed an equivalent particle diameter, D_c, can be derived from the definition of the specific surface area and the assumption that the particles are spherical:

$$S_c = \frac{\text{surface area exposed to the fluid}}{\text{total volume of solid material}}$$

$$= \frac{\pi D_c^2}{\frac{\pi}{6} D_c^3} = \frac{6}{D_c}$$

or

$$D_c = \frac{6}{S_c} \tag{4}$$

where

S_c = surface area of debris exposed to coolant per unit volume of solid material in the region (m^{-1})

D_c = equivalent particle diameter (m).

D_c characterizes the diameter of the particles in the cohesive debris bed. This approach has been successfully applied to Equation (3) as described in Reference 5.

In Equation (3), α_c and β_c are constants characterizing the structure of the porous material; α_c is the viscous resistance coefficient, and β_c is the inertial resistance coefficient. A characteristic length, l_c, of the pore structure is calculated from the definition[6]

$$l_c = \frac{\beta_c}{\alpha_c} \tag{5}$$

where l_c is expressed in units of meters. The characteristic length is used to determine the volumetric heat transfer coefficient for interfacial transport of thermal energy between the solid and coolant. This will be discussed in Section 2.2.3.

2.1.2 Rubble Debris Bed

The debris transition and propagation models discussed in the previous section also consider formation of a rubble debris bed which may result during reflooding if sufficient component oxidation (embrittlement) has occurred prior to reflood. As in the case of cohesive debris, the debris transition and propagation models calculate data which are required to characterize a rubble debris bed for subsequent thermal analysis. These data are:

1. Rubble debris region height and elevation with respect to the fuel bundle

2. The masses of the individual constituents of the debris bed

3. The average particle size for the rubble debris bed.

Theoretical analyses of random packings of particles have been under development for the past several decades.[7,8] Debbas and Rampf[9] treated random packings of irregularly shaped particles with different size distributions by means of statistical methods. In a systematic assemblage of uniform spheres, the lossest (cubic) and highest possible packings (rhombohedral and face-centered cubic) correspond to porosities of 0.43 and 0.26, respectively. For the initial version of SCDAP, the models which describe the thermal-hydraulic behavior of a rubble debris bed are based on the assumption of no variation in particle size along the axial direction of the bed. This simplification allows use of data from the Post Accident Heat Removal and fuel melt experiments performed for the LMFBR[10] to derive a debris bed packing model for SCDAP. Measurements of fuel porosity, based on radiographs of a fuel particle bed that was heated to sodium boiling and then cooled, provided the following correlation for packed bed porosity, ϵ_r:[11]

$$\epsilon_r = 0.593 - 1.23 \times 10^{-4} B_F \qquad (6)$$

where

B_F = debris bed loading per unit area (kg/m^2).

The value of B_F is calculated with the expression

$$B_F = \frac{M_{UO_2} + M_{ZrO_2} + M_{Zr} + M_{St} + M_{Ab}}{A_r} \qquad (7)$$

where

A_r = cross-sectional area of the rubble bed (m^2).

It is noted that Equation (6) is valid for sodium coolant and bed loadings from 0 to 1000 kg/m^2. For water-UO_2, water-steel, water-lead, and aceton-steel systems, correlation between bed porosity and bed loading for particle packing can not be derived due to the scatter of experimental data.[10]

Since a rubble debris bed consists of loosely bound particles, the fragments may settle on the available horizontal space as a packed bed or be suspended in the upwardly flowing coolant as a fluidized bed, depending on the coolant flow rate. Furthermore, the debris particles may move out of the rubble debris region if the coolant velocity exceeds the terminal velocity of the particles.

For the packed bed configuration, the rubble debris bed pressure drop is calculated in the same manner as for the cohesive debris bed [Equation (3)]:

$$\frac{\Delta P_r}{u_o H_r} = \alpha_r \mu + \beta_r G \tag{8}$$

where

ΔP_r = coolant pressure drop across packed bed (Pa)

H_r = height of packed bed (m)

$\alpha_r = 150 \frac{(1 - \epsilon_r)^2}{\epsilon_r^3} \frac{1}{D_r^2}$

$\beta_r = 1.75 \frac{1 - \epsilon_r}{\epsilon_r^3} \frac{1}{D_r}$

D_r = average particle diameter (m) [input from debris transition and propagation models].

As was the case for a cohesive debris bed, a characteristic length of the pore structure in a rubble debris bed can be calculated as

$$l_r = \frac{\beta_r}{\alpha_r} \quad . \tag{9}$$

The frictional pressure loss (ΔP_r) in a packed bed of solid particles increases as the superficial velocity of the coolant flowing upwardly through the bed increases. When the fluid velocity reaches a value that induces an upward drag on the particles equal to the weight of the particles, the bed becomes weightless. This condition is defined to be minimum or incipient fluidization. Any further increase in the fluid velocity produces an upward motion of the particles.

The minimum fluidization velocity can be calculated by setting pressure drop equal to debris bed weight per unit cross-sectional area and solving for velocity, as was done in Reference 13. This yields

$$u_f = 9.2975 \times 10^{-3} \, D_r^{1.82} \, \frac{\rho_f^{-0.06} \, (\rho_s - \rho_f)^{0.94}}{\mu^{0.88}} \tag{10}$$

where

u_f = minimum fluidization velocity (m/s)

ρ_s = density of debris bed material (kg/m^3)

ρ_f = density of fluid (kg/m^3).

The density of the debris bed is calculated using the expression

$$\rho_s = \frac{M_{UO_2} + M_{ZrO_2} + M_{Zr} + M_{St} + M_{Ab}}{\dfrac{M_{UO_2}}{\rho_{UO_2}} + \dfrac{M_{ZrO_2}}{\rho_{ZrO_2}} + \dfrac{M_{Zr}}{\rho_{Zr}} + \dfrac{M_{St}}{\rho_{St}} + \dfrac{M_{Ab}}{\rho_{Ab}}} \quad . \tag{11}$$

The data which support the correlation given by Equation (10) cover a wide range of particle size, particle diameter, and coolant density, as reported in Reference (5). The correlation is valid only for Reynolds numbers[a] less than 7.57, and a correction factor is required for Reynolds numbers greater than 7.57. The correlation factor C_f is[4]

$$
\begin{aligned}
C_f &= 1, && \text{for } Re < 7.57 \\
C_f &= 1.364 - 0.18 \ln(Re), && \text{for } 7.57 \leq Re < 200 \\
C_f &= 0.214 + 39.4/Re, && \text{for } 200 \leq Re < 1000 \qquad (12) \\
C_f &= 0.254, && \text{for } Re \geq 1000.
\end{aligned}
$$

Using Equations (10) and (12), the minimum fluidization velocity, u_{mf}, can be expressed as

$$u_{mf} = C_f \cdot u_f \quad . \qquad (13)$$

This velocity is used to determine whether the debris particles will be in a packed bed configuration or in a fluidized state.

For a fluidized bed, the pressure drop across the bed remains constant with increasing flow rate; i.e.,

$$\Delta P_f = H_o (\rho_s - \rho_f) g \qquad (14)$$

where

ΔP_f = coolant pressure drop across fluidized rubble bed (Pa)

---

[a] Reynolds number is defined as GD_r/μ.

H_o = zero-porosity debris bed height (m)

$$= \left(\frac{M_{UO_2}}{\rho_{UO_2}} + \frac{M_{ZrO_2}}{\rho_{ZrO_2}} + \frac{M_{Zr}}{\rho_{Zr}} + \frac{M_{St}}{\rho_{St}} + \frac{M_{Ab}}{\rho_{Ab}}\right)\frac{1}{A_r}$$

g = gravitational acceleration constant (m/s^2)

= 9.8.

Porosity of a fluidized bed is calculated as[14]

$$\varepsilon_f = \left(\frac{Re}{Re_{mf}} \cdot \varepsilon_{mf}^n\right)^{\frac{1}{n}} \tag{15}$$

where

ε_f = porosity of fluidized bed

Re = Reynolds number of fluid flowing through fluidized bed

Re_{mf} = Reynolds number of fluid at minimum fluidization velocity

ε_{mf} = debris bed porosity at minimum fluidization velocity (Set equal to ε_r based on the assumption that bed expansion will not occur before the pressure drop has reached the buoyant weight per unit area of bed).

In logarithmic coordinates, n is the slope of the line which represents the relationship between Reynolds number and fluidized bed porosity. According to the analysis of Richardson and Zaki,[14]

n = 5.0, for $Re < 0.2$

$$n = \left(4.35 + 17.5 \frac{D_r}{D_t}\right) Re^{-0.03}, \qquad \text{for } 0.2 \leq Re < 1$$

$$n = \left(4.45 + 18 \frac{D_r}{D_t}\right) Re^{-0.1}, \qquad \text{for } 1 \leq Re < 200$$

$$n = 4.45\ Re^{-0.1}, \qquad \text{for } 200 \leq Re < 500$$

$$n = 2.39, \qquad \text{for } Re \geq 500 \qquad (16)$$

where

D_t = diameter of the debris bed (m) [input variable].

The fluidized bed height, L_f, is then calculated as

$$L_f = \frac{H_o}{1 - \varepsilon_f} \quad . \qquad (17)$$

The height, H_{rd}, of the rubble debris region is specified by the debris transition and propagation models in SCDAP. To prevent the fluidized bed height from exceeding H_{rd}, L_f and ε_f are adjusted as follows if L_f is calculated to be greater than H_{rd}:

$$L_f = H_{rd}$$

$$\varepsilon_f = 1 - \frac{H_o}{H_{rd}} \quad . \qquad (18)$$

2.2 Debris Bed and Coolant Thermal Models

Debris bed and coolant temperatures are determined by simultaneous solution of the following two differential equations:[15]

$$(1 - \varepsilon)\ \rho_p c_p \frac{\partial T_p}{\partial t} = (1 - \varepsilon)\ k_p \frac{\partial^2 T_p}{\partial z^2} - hv\ (T_p - T_c) + Q'''\ (1 - \varepsilon)$$

$$\varepsilon \rho_c c_c \frac{\partial T_c}{\partial t} = \varepsilon k_c \frac{\partial^2 T_c}{\partial z^2} + hv\,(T_p - T_c) - G\varepsilon c_c \frac{\partial T_c}{\partial z} \qquad (19)$$

where

ε = debris bed porosity

ρ_p, ρ_c = density of debris bed and coolant (kg/m^3)

c_p, c_c = specific heat of debris bed and coolant (J/kg•K)

k_p, k_c = thermal conductivity of debris bed and coolant (W/m•K)

T_p, T_c = debris bed and coolant temperature (K)

hv = volumetric heat transfer coefficient (W/m^3•K)

Q''' = volumetric heat generation rate (W/m^3)

z = axial coordinate (m).

Before solving Equation (19), local coolant conditions and regions of analysis, heat generation, heat transfer coefficient between solid and coolant, and debris bed dryout analysis will be discussed. These discussions will be followed by a description of the temperature solution and propagation of a melt front within a debris bed.

2.2.1 Local Coolant Condition and Regions of Analysis

A debris bed can be divided into the following three regions based on the coolant temperature distribution in the bed:

1. Subcooled region where coolant temperature is less than the coolant saturation temperature at the system pressure

2. Saturated region where coolant temperature is equal to the coolant saturation temperature

3. Superheated region where coolant temperature is greater than the coolant saturation temperature.

Here, coolant temperature is considered to represent the homogeneous temperature of the coolant with the assumption of perfect mixing of the liquid and vapor phases.

For a given coolant flow rate and debris bed volumetric heat generation rate, the height of the subcooled, saturated, and superheated regions can be calculated as follows:

$$L_{sc} = \frac{\rho_c V_c c_c}{Q'''(1 - \varepsilon)} (T_{sat} - T_{in})$$

$$L_{sa} = \frac{\rho_c V_c h_{fg}}{Q'''(1 - \varepsilon)}$$

$$L_{sp} = H_d - L_{sc} - L_{sa} \tag{20}$$

where

L_{sc} = subcooled region height (m)

L_{sa} = saturated region height (m)

L_{sp} = superheated region height (m)

V_c = coolant velocity (m/s)

T_{sat} = coolant saturation temperature at system pressure (K)

T_{in} = coolant inlet temperature (K)

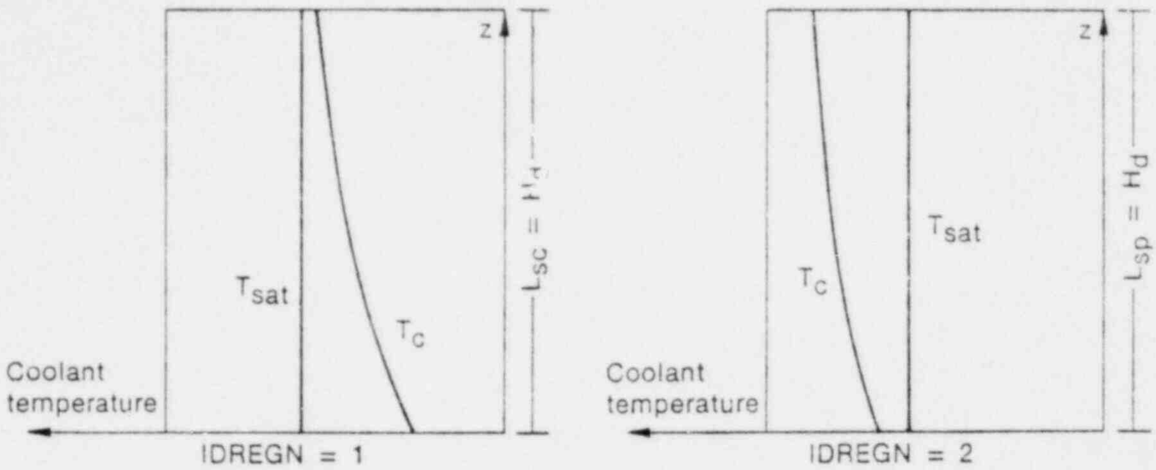
h_{fg} = latent heat of evaporation (J/kg)

H_d = debris bed height (m).

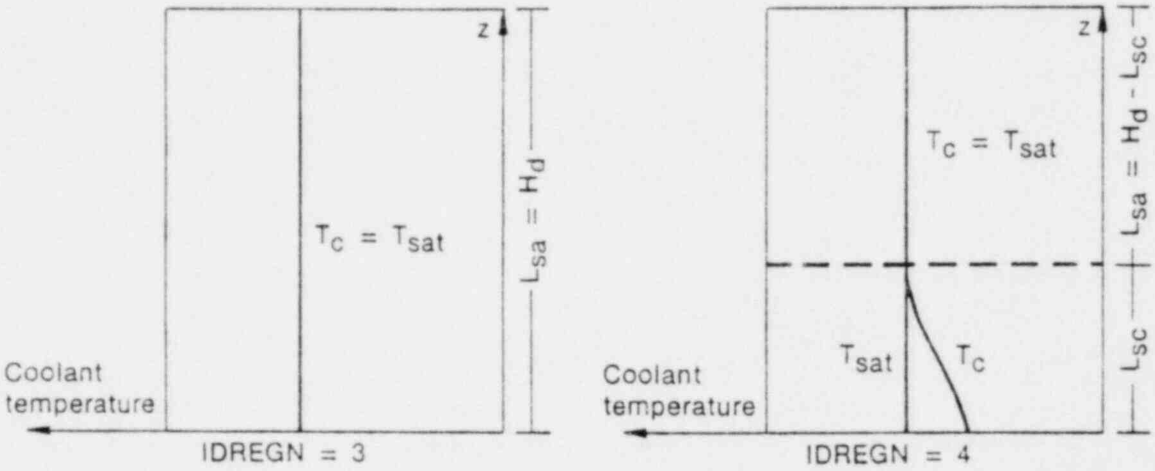
Figure 1 shows the possible combinations of regions for the temperature analysis. Also shown in the figure is the assigned debris identification number, IDREGN, for the different possible configurations. For each subcooled and superheated region, the region is divided into nd equally spaced nodes for the temperature analysis. For the saturated region, all of the heat generated within the bed is assumed to be consumed in evaporating saturated coolant (raising quality from 0 to 1). Coolant temperature remains at the saturation temperature, and the bed temperature remains the same as the previous nodal temperature. Thus, only three nodes are used for specifying the temperature of the coolant and debris bed in a saturated region. Based on the above discussion and with reference to Figure 1, the total number of nodes, NOND, is

$$\mathrm{NOND} = \begin{cases} \mathrm{nd}, & \text{for IDREGN} = 1 \\ \mathrm{nd}, & \text{for IDREGN} = 2 \\ 3, & \text{for IDREGN} = 3 \\ \mathrm{nd} + 2, & \text{for IDREGN} = 4 \\ \mathrm{nd} + 2, & \text{for IDREGN} = 5 \\ 2\mathrm{nd} + 1, & \text{for IDREGN} = 6 \end{cases} \quad . \tag{21}$$

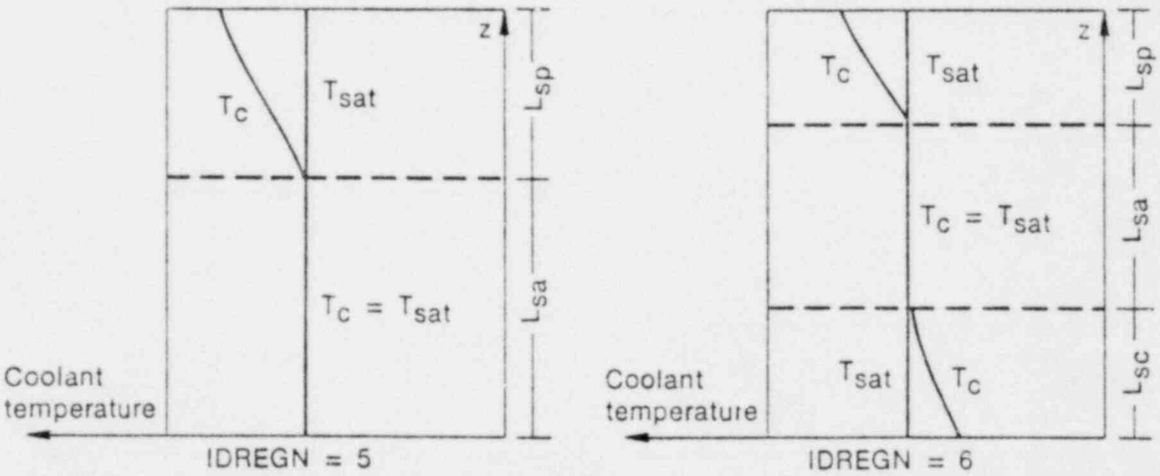
During the calculation of debris bed and coolant temperatures, heights of the different regions may change and/or different regions may be used due to changes in heat generation rate, coolant flow rate, etc. A model (REGMOD) has been developed to redefine the total number of nodes, node elevations, and coolant and debris bed temperatures corresponding to each



(a) Subcooled region (b) Superheated region



(c) Saturated region (d) Subcooled and saturated region



(e) Saturated and superheated region (f) Subcooled, saturated and superheated region

Figure 1. Possible regions to be analyzed in a debris bed.

node. The temperatures for each node are determined by REGMOD using interpolation of the previous nodal temperatures.

2.2.2 Heat Generation

The primary sources of heat generation within a debris bed will be decay heat (and possible fission heat for in-pile experiments) and chemical heat of reaction associated with Zircaloy oxidation at elevated temperatures. The heat source included in Equation (19) includes the above such heat sources. However, the magnitude of heat generation due to the different possible sources is an input to the debris characterization and behavior models described herein. Because decay heat and oxidation heat can be sensitive to the debris bed temperature, the heat source used in Equation (19) may need to reflect the bed temperature. The debris bed thermal models include an iteration scheme which can be used to obtain a consistant heat source and temperature solution.

2.2.3 Heat Transfer Coefficient

Because of the difficulties associated with assigning a heat-transfer surface area and temperature difference between the solid and fluid in a debris bed, the heat transfer coefficient between debris bed and coolant which appears in Equation (19) is based on the volume of the porous system. Using the characteristic length of the pore structure (l_c or l_r), the Nusselt and Reynold's numbers are defined as[16]

$$Nu = \frac{h_v l^2}{k_c}$$

$$Re = \frac{Gl}{\mu} \qquad (22)$$

where l equals l_c or l_r for a cohesive bed and rubble bed, respectively.

In the subcooled and saturated regions of either a cohesive debris bed or packed bed, hv is given as[17]

$$hv = \left[(7 - 10\varepsilon + 5\varepsilon^2)(1 + 0.7\,Re^{0.3}Pr^{\frac{1}{3}}) + (1.33 - 2.4\varepsilon + 1.2\varepsilon^2)\right.$$

$$\left. Re^{0.7}Pr^{\frac{1}{3}}\right]\frac{k_c}{l^2} \tag{23}$$

where

Pr = Prandtl number

= $\frac{c_c \mu}{k_c}$

ε = ε_c or ε_r for cohesive and rubble bed, respectively.

In the subcooled region of a fluidized rubble debris bed, hv is expressed as[18]

$$hv = 1.28 \times 10^5\,(Re\,F\varepsilon)^2\,Pr^{0.67}\left(\frac{\mu}{\mu_R}\right)^{0.83}\left(\frac{D_t}{D_r}\right)\left(\frac{\rho_s}{\rho_c}\right)^2\left(\frac{k_c}{l_r^2}\right) \tag{24}$$

where

$F\varepsilon$ = $[1 - 1.209(1 - \varepsilon_f)^{2/3}]^{-1}$

μ_R = coolant viscosity for water at 300 K.

For a saturated region of a fluidized rubble bed, hv is correlated as[19]

$$hv = 2.2 \times 10^5 \, Pr^{-2.13} \left[\left(\frac{T_{sat} c_l}{h_{fg}} \right) \left(\frac{\rho_l u^2 F \varepsilon^2}{h_{fg} \rho_v g} \right) \right]^{0.84} \left[\frac{D_r^2 (\rho_f - \rho_v)}{\sigma g} \right]^{0.5} \frac{k_l}{l_r^2} \tag{25}$$

where

σ = surface tension of coolant (kg/s^2)

= 59.3 for water at 373 K

ρ_l, ρ_v = density of liquid and vapor phases, respectively (kg/m^3).

In a superheated region for either a cohesive or rubble debris bed,[20]

$$hv = \left[\frac{1 \ (1. - \varepsilon)}{0.00377} \right]^{1.33} \frac{Re^{0.65} k_c}{l^2} \ . \tag{26}$$

2.2.4 Debris Bed Dryout

For a debris bed immersed in a pool of fluid with an insulated bottom or with a forced flow of fluid through the bottom of the bed, debris bed dryout is defined to occur when the vapor generation rate is sufficiently large to preclude an adequate flow of replenishing liquid. Such a situation might lead to a sustained temperature rise within the bed and subsequent melting of the debris bed material. Research on dryout in a rubble bed has been conducted by several experimenters[12,21-30] and has involved water, acetone, methanol, and sodium with steel, lead, sand, and urania.

Of the several dryout models available in the literature, the model developed by Lipinski (References 28-30) is one of the most recent. In addition, Lipinski's model considers bottom flooding in deep beds, as may occur in an LWR severe accident sequence. A model (DRYOUT) has been developed based on Lipinski's research. The model is described below.

Through algebraic manipulation of the one-dimensional conservation equations for two-phase, counter-current flow in a porous medium, Lipinski developed a simple quadratic equation for the debris bed heat flux (q) as a function of the effective saturation in the bed (Se). The equation is

$$\frac{1.75\,(1-\varepsilon)}{d\varepsilon^{3}\,h_{fg}^{2}}\left[\frac{1}{\rho_{v}(1-Se)^{3}} \pm \frac{1}{\rho_{l}Se^{3}}\right]q^{2} + \left[\frac{180\,(1-\varepsilon)^{2}}{d^{2}\varepsilon^{3}\,h_{fg}}\left(\frac{\mu_{v}}{\rho_{v}(1-Se)^{3}} + \frac{\mu_{l}}{\rho_{l}\,Se^{3}}\right) \mp \frac{3.5\,(1-\varepsilon)\,G}{d\varepsilon^{3}\rho_{l}Se^{3}h_{fg}}\right]q \pm \frac{1.75\,(1-\varepsilon)G^{2}}{d\varepsilon^{3}\rho_{l}Se^{3}} - \frac{180\,(1-\varepsilon)^{2}\mu_{l}G}{d^{2}\varepsilon^{3}\rho_{l}Se^{3}} - (\rho_{l}-\rho_{v})\,g = 0 \tag{27}$$

where

ε = debris bed porosity

d = debris particle diameter (m)

ρ_v, ρ_l = density of vapor and liquid phase of coolant (kg/m^3)

q = debris bed heat flux (W/m^2)

Se = bed effective saturation

μ_v, μ_l = viscosity of vapor and liquid phases of coolant ($kg/m^2 \cdot s$).

For $q > Gh_{fg}$, the upper arithmetic operation in Equation (27) is used. For $q < Gh_{fg}$, the lower arithmetic operation is used.

Heat fluxes predicted with Equation (27) are shown in Figure 2. The dryout heat flux is given by the maximas in the curves of heat flux versus the effective saturation of the bed. As the flow of liquid into the bottom of the bed increases, the maximum point on the heat flux versus effective saturation curves approaches an effective saturation of zero. Physically, this implies that the dryout heat flux (q_d) becomes equal to the heat required to vaporize all of the liquid entering the bottom of the bed ($q_d = Gh_{fg}$) and that the penetration of liquid from the top is precluded.

As discussed in Reference (31), the factors significant in calculating dryout heat flux are coolant properties and bed characteristics such as bed void fraction and particle diameter. As shown in Figure 3 (Figure 3 of Reference 31), dryout heat flux increases with bed void fraction and particle size for low or zero bottom flow rates. When the bottom flow rates become large enough that the effective saturation becomes zero at dryout, dryout heat fluxes become independent of bed void fraction and particle size.

The DRYOUT model numerically solves for the maximum heat flux (the dryout heat flux) predicted by Equation (27) given values of the dependent variables which describe the state of the debris bed. The calculated dryout heat flux is then compared with the local heat flux. If the local heat flux is greater than the dryout heat flux, the heat transfer coefficient given by Equation (26) is used in the subsequent temperature calculations. If not, the heat transfer coefficient as calculated with Equations (23), (24), or (25) is used in the temperature calculations.

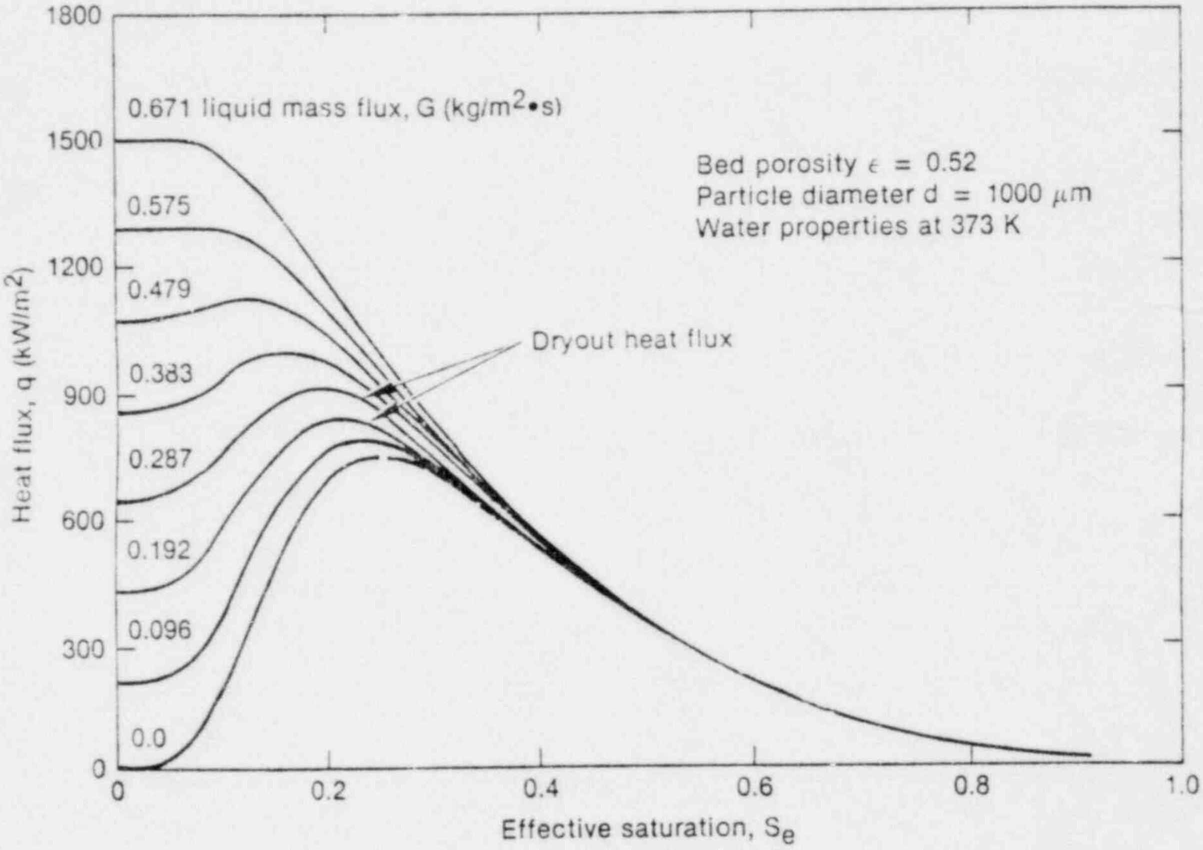


Figure 2. Dryout heat flux from Equation (27).

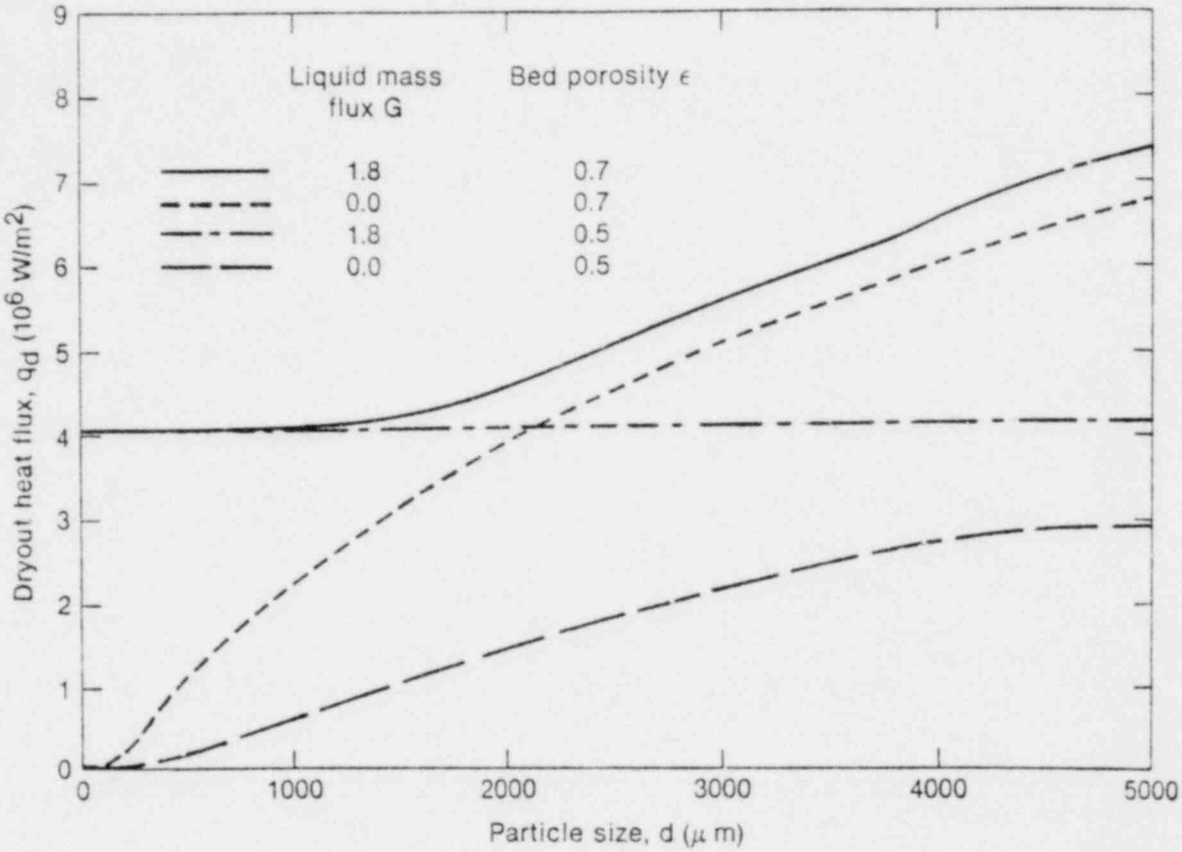


Figure 3. Dryout heat flux dependency upon coolant mass flux, bed void fraction, and particle size.

2.2.5 Temperature Solution

In the subcooled and superheated regions of a debris bed, nd equally-spaced nodes are used to solve Equation (19). The finite difference form of Equation (19) is:

$$\rho_{pn} c_{pn} \frac{T_{pn}^{m+1} - T_{pn}^{m}}{\Delta t} = \frac{k_{pn,n+1}\left(T_{pn+1}^{m+1/2} - T_{pn}^{m+1/2}\right) - k_{pn-1,n}\left(T_{pn}^{m+1/2} - T_{pn-1}^{m+1/2}\right)}{\Delta z^2}$$

$$- \frac{hv}{1 - \varepsilon}\left(T_{pn}^{m+1/2} - T_{cn}^{m+1/2}\right) + Q'''$$

$$\rho_{cn} c_{cn} \frac{T_{cn}^{m+1} - T_{cn}^{m}}{\Delta t} = \frac{k_{cn,n+1}\left(T_{cn+1}^{m+1/2} - T_{cn}^{m+1/2}\right) - k_{cn-1,n}\left(T_{cn}^{m+1/2} - T_{cn-1}^{m+1/2}\right)}{\Delta z^2}$$

$$+ \frac{hv}{\varepsilon}\left(T_{pn}^{m+1/2} - T_{cn}^{m+1/2}\right) - Gc_{cn} \frac{T_{cn+1}^{m+1/2} - T_{cn-1}^{m+1/2}}{2\Delta z} \qquad (28)$$

where

ρ_{pn}, ρ_{cn} = debris bed and coolant density at node n (kg/m^3)

c_{pn}, c_{cn} = debris bed and coolant specific heat (J/kg•K)

$k_{pn,n+1}$, $k_{cn,n+1}$ = debris bed and coolant thermal conductivity (W/m•K)

T_{pn}^{m}, T_{cn}^{m} = debris bed and coolant temperature at node n and at time step m (K)

T_{pn}^{m+1}, T_{cn}^{m+1} = debris bed and coolant temperature at node n and at time step m + 1 (K)

$T_{pn}^{m+1/2}$, $T_{cn}^{m+1/2}$ = $1/2\left(T_{pn}^{m+1} + T_{pn}^{m}\right)$ and $1/2\left(T_{cn}^{m+1} + T_{cn}^{m}\right)$, respectively (K)

Δz = space between spatial nodes (m)

Δt = time step (s).

The implicit finite difference form of Equation (28) with the appropriate boundary conditions is given below.

For the debris,

n=1

$$T_{p2}^{m+1}\left(-\frac{k_{p1,2}}{\Delta z^2}\right) + T_{p1}^{m+1}\left[\frac{\rho_{p1}c_{p1}}{\Delta t} + \frac{k_{p1,2}}{\Delta z^2} + \frac{hv}{2(1-\varepsilon)}\right] + T_{c1}^{m+1}\left[-\frac{hv}{2(1-\varepsilon)}\right]$$

$$= T_{p2}^{m}\left(-\frac{k_{p1,2}}{\Delta z^2}\right) + T_{p1}^{m}\left[\frac{\rho_{p1}c_{p1}}{\Delta t} + \frac{k_{p1,2}}{\Delta z^2} + \frac{hv}{2(1-\varepsilon)}\right] + T_{c1}^{m}\left[-\frac{hv}{2(1-\varepsilon)}\right]$$

$$+ Q''' + 2\cdot \mathrm{HTLBCD}/\Delta z \qquad (29)$$

n=n

$$T_{pn+1}^{m+1}\left(-\frac{k_{pn,n+1}}{2\Delta z^2}\right) + T_{pn}^{m+1}\left[\frac{\rho_{pn}c_{pn}}{\Delta t} + \frac{k_{pn,n+1}}{2\Delta z^2} + \frac{k_{pn-1,n}}{2\Delta z^2} + \frac{hv}{2(1-\varepsilon)}\right] + T_{pn-1}^{m+1}\left(-\frac{k_{pn-1,n}}{2\Delta z^2}\right)$$

$$+ T_{cn}^{m+1}\left[-\frac{hv}{2(1-\varepsilon)}\right] = T_{pn+1}^{m}\left(-\frac{k_{pn,n+1}}{2\Delta z^2}\right) + T_{pn}^{m}\left[\frac{\rho_{pn}c_{pn}}{\Delta t} + \frac{k_{pn,n+1}}{2\Delta z^2}\right.$$

$$\left.+ \frac{k_{pn-1,n}}{2\Delta z^2} + \frac{hv}{2(1-\varepsilon)}\right] + T_{pn-1}^{m}\left(-\frac{k_{pn-1,n}}{2\Delta z^2}\right) + T_{cn}^{m}\left[-\frac{hv}{2(1-\varepsilon)}\right] + Q''' \qquad (30)$$

n=nd=N

$$T^{m+1}_{pN}\left[\frac{\rho_{pN}c_{pN}}{\Delta t}+\frac{k_{pN-1,N}}{\Delta z^2}+\frac{hv}{2(1-\varepsilon)}\right]+T^{m+1}_{pN-1}\left(-\frac{k_{pN-1,N}}{\Delta z^2}\right)+T^{m+1}_{cN}\left[-\frac{hv}{2(1-\varepsilon)}\right]$$

$$=T^{m}_{pN}\left[\frac{\rho_{pN}c_{pN}}{\Delta t}+\frac{k_{pN-1,N}}{\Delta z^2}+\frac{hv}{2(1-\varepsilon)}\right]+T^{m}_{pN-1}\left(-\frac{k_{pN-1,N}}{\Delta z^2}\right)+T^{m}_{cN}\left[-\frac{hv}{2(1-\varepsilon)}\right]$$

$$+Q'''+\frac{2\cdot HTUBCD}{\Delta z} \qquad (31)$$

For the coolant,

n=1

$$T^{m+1}_{c2}\left(-\frac{k_{c1,2}}{\Delta z^2}+\frac{Gc_{c1}}{2\Delta z}\right)+T^{m+1}_{c1}\left(\frac{\rho_{c1}c_{c1}}{\Delta t}+\frac{k_{c1,2}}{\Delta z^2}+\frac{hv}{2\varepsilon}-\frac{Gc_{c1}}{2\Delta z}\right)+T^{m+1}_{p1}\left(-\frac{hv}{2\varepsilon}\right)$$

$$=T^{m}_{c2}\left(-\frac{k_{c1,2}}{\Delta z^2}+\frac{Gc_{c1}}{2\Delta z}\right)+T^{m}_{c1}\left(\frac{\rho_{c1}c_{c1}}{\Delta t}+\frac{k_{c1,2}}{\Delta z^2}+\frac{hv}{2\varepsilon}-\frac{Gc_{c1}}{2\Delta z}\right)$$

$$+T^{m}_{p1}\left(-\frac{hv}{2\varepsilon}\right)+\frac{2\cdot HTLBCC}{\Delta z} \qquad (32)$$

n=n

$$T^{m+1}_{cn+1}\left(-\frac{k_{cn,n+1}}{2\Delta z^2}+\frac{Gc_{cn}}{4\Delta z}\right)+T^{m+1}_{cn}\left(\frac{\rho_{cn}c_{cn}}{\Delta t}+\frac{k_{cn,n+1}}{2\Delta z^2}+\frac{k_{cn-1,n}}{2\Delta z^2}+\frac{hv}{2\varepsilon}\right)$$

$$+T^{m+1}_{cn-1}\left(-\frac{k_{cn-1,n}}{2\Delta z^2}-\frac{Gc_{cn}}{4\Delta z}\right)+T^{m+1}_{pn}\left(-\frac{hv}{2\varepsilon}\right)=T^{m}_{cn+1}\left(-\frac{k_{cn,n+1}}{2\Delta z^2}\right.$$

$$\left.+\frac{Gc_{cn}}{4\Delta z}\right)+T^{m}_{cn}\left(\frac{\rho_{cn}c_{cn}}{\Delta t}+\frac{k_{cn,n+1}}{2\Delta z^2}+\frac{k_{cn-1,n}}{2\Delta z^2}+\frac{hv}{2\varepsilon}\right)$$

$$+ T^{m}_{cn-1}\left(- \frac{k_{cn-1,n}}{2\Delta z^{2}} - \frac{Gc_{cn}}{4\Delta z}\right) + T^{m}_{pn}\left(- \frac{hv}{2\varepsilon}\right) \tag{33}$$

n=nd=N

$$T^{m+1}_{cN}\left(\frac{\rho_{cN}c_{cN}}{\Delta t} + \frac{k_{cN-1,N}}{\Delta z^{2}} + \frac{hv}{2\varepsilon} + \frac{Gc_{cN}}{2\Delta z}\right) + T^{m+1}_{cN-1}\left(- \frac{k_{cN-1,N}}{\Delta z^{2}} - \frac{Gc_{cN}}{2\Delta z}\right)$$

$$+ T^{m+1}_{pN}\left(- \frac{hv}{2\varepsilon}\right) = T^{m}_{cN}\left(\frac{\rho_{cN}C_{cN}}{\Delta t} + \frac{k_{cN-1,N}}{\Delta z^{2}} + \frac{hv}{2\varepsilon} + \frac{Gc_{cN}}{2\Delta z}\right) + T^{m}_{cN-1}$$

$$\left(- \frac{k_{cN-1,N}}{\Delta z^{2}} - \frac{Gc_{cN}}{2\Delta z}\right) + T^{m}_{pN}\left(- \frac{hv}{2\varepsilon}\right) + \frac{2 \cdot HTUBCC}{\Delta z} \tag{34}$$

where

HTLBCD = heat transfer into debris bed at lower boundary (W/m^2)

HTLBCC = heat transfer into coolant at lower boundary (W/m^2)

HTUBCD = heat transfer into debris bed at upper boundary (W/m^2)

HTUBCC = heat transfer into coolant at upper boundary (W/m^2).

The matrix form of Equations (29) through (34) is

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \cdot T = R \tag{35}$$

where

$$A_{11} = \begin{bmatrix} \left[\frac{\rho_{p1}c_{p1}}{\Delta t} + \frac{k_{p1,2}}{\Delta z^2} + \frac{hv}{2(1-\epsilon)}\right] & \left[-\frac{k_{p1,2}}{\Delta z^2}\right] & & \bigcirc \\ \left[-\frac{k_{p1,2}}{\Delta z^2}\right] & \ddots & \ddots & \\ & \ddots & \ddots & \left[-\frac{k_{pN-1,N}}{2\Delta z^2}\right] \\ \bigcirc & & \left[-\frac{k_{pN-1,N}}{\Delta z^2}\right] & \left[\frac{\rho_{pN}c_{pN}}{\Delta t} + \frac{k_{pN-1,N}}{\Delta z^2} + \frac{hv}{2(1-\epsilon)}\right] \end{bmatrix}$$

$$A_{21} = \begin{bmatrix} \left[-\frac{hv}{2\varepsilon}\right] & & \bigcirc \\ & \ddots & \\ \bigcirc & & \left[-\frac{hv}{2\varepsilon}\right] \end{bmatrix}$$

$$A_{12} = \begin{bmatrix} \left[-\frac{hv}{2(1-\varepsilon)}\right] & & \bigcirc \\ & \ddots & \\ \bigcirc & & \left[-\frac{hv}{2(1-\varepsilon)}\right] \end{bmatrix}$$

$$A_{22} = \begin{bmatrix}
\left[\frac{\rho_{c1}c_{c1}}{\Delta t} + \frac{k_{c1,2}}{\Delta z^2} + \frac{hv}{2\varepsilon} - \frac{Gc_{c1}}{2\Delta z}\right] & \left[-\frac{k_{c1,2}}{\Delta z^2} + \frac{Gc_{c1}}{2\Delta z}\right] & & \bigcirc \\
\left[\frac{k_{c1,2}}{2\Delta z^2} - \frac{Gc_{c2}}{4\Delta z}\right] & \ddots & \ddots & \\
 & \ddots & \ddots & \left[-\frac{k_{cN-1,N}}{2\Delta z^2} + \frac{Gc_{cN-1}}{4\Delta z}\right] \\
\bigcirc & & \left[-\frac{k_{cN-1,N}}{\Delta z^2} - \frac{Gc_{cN}}{2\Delta z}\right] & \left[\frac{\rho_{cN}c_{cN}}{\Delta t} + \frac{k_{cN-1,N}}{\Delta z^2} + \frac{hv}{2\varepsilon} + \frac{Gc_{cN}}{2\Delta z}\right]
\end{bmatrix}$$

$$T = \begin{bmatrix} T_{p1}^{m+1} \\ \vdots \\ T_{pN}^{m+1} \\ T_{c1}^{m+1} \\ \vdots \\ T_{cN}^{m+1} \end{bmatrix} \qquad R = \begin{bmatrix} R_1 \\ \vdots \\ R_N \\ R_{N+1} \\ \vdots \\ R_{2N} \end{bmatrix}$$

The large circles in matrices A_{11}, A_{12}, A_{21}, and A_{22} signify that the off-diagonal or off-tridiagonal elements are zero. Thus, there are 2N linear equations, the first N equations for the debris temperatures and the remaining N equations for the coolant temperatures. The equations are solved simultaneously.

The TEMPSR model solves the above matrix of equations to obtain the debris bed and coolant temperatures. For a saturated region, the coolant temperature is assumed to equal the saturation coolant temperature. The debris temperature is equal to the previous nodal temperature. Thus, calculations of coolant and debris bed temperatures are not needed for the saturation region.

2.2.6 Melting Front Propagation

The debris bed thermal models employ a simplified approach for describing propagation of a melt front within a debris bed. The following assumptions are used:

1. Axial heat conduction is infinite.

2. Super heating of the molten material does not occur until all of the debris bed has melted.

If the calculated debris bed temperature T_i at node i is greater than the melting temperature of the debris bed material, T_{mt}, the bed temperature at node i is set equal to the value of T_{mt}. The energy excess, $\rho[c_p(T_i - T_{mt}) - h_{sf}]\Delta V_i$, is assumed to increase the temperature of the debris material at node i-1. (ρ, c_p, h_{sf}, and ΔV_i are debris bed density, debris bed specific heat, latent heat of fusion for debris material, and debris bed volume associated with node i, respectively.) The debris material at node i-1 is first heated to the melting temperature. If the energy excess from node i is large enough, the debris material at node i-1 may be melted and the process may progress downward to node i-2, etc.

Propagation of a melt front, as described above, is analyzed by the FRAGTH and FROZTH models for the rubble and cohesive debris beds, respectively. Based on the progression of the melt front during any time step, the FRMELT and FGMELT models calculate the velocity of the front in the rubble and debris beds, respectively. Based on this velocity, the FRESTT and FGESTT models estimate the time when the melt front will reach the lower boundary of the rubble and cohesive debris beds, respectively.

2.3 Applicability Range of Correlations

Many of the equations described in Sections 2.1 and 2.2 have been derived empirically. Table 1 summarizes the data ranges of these empirical correlations. Also shown are the expected ranges of application of the correlations for SCDAP. For the most part, the ranges of the correlations span the expected range of application for SCDAP analyses. Planned assessment of SCDAP will demonstrate if exceeding the data ranges of the correlations is significant. Experimental data on debris bed behavior for LWR conditions will be obtained during the next several years. This data will provide a basis for removing deficiencies which exist in the present modeling.

TABLE 1. SUMMARY OF DATA RANGES AND EXPECTED RANGES FOR CORRELATIONS IN SECTION 2

Equation Number	Data Range	Expected Application Range
3,8	$0.3 < \epsilon < 0.8$ $0.2 < Re < 3500$ $1.5 \times 10^{-4} < D_p(m) < 2 \times 10^{-3}$	$0.2 < \epsilon < 1$ $0 < Re < 1.2 \times 10^5$ $1.9 \times 10^{-5} < D_p(m) < 3 \times 10^{-3}$
6	$0 < B_F(kg/m^2) < 1000$	$0 < B_F\ (kg/m^2) < 3.5 \times 10^4$
10,12	$0.3 < \epsilon < 0.8$ $0.001 < Re < 10^6$ $5.1 \times 10^{-5} < D_p(m) < 5.2 \times 10^{-3}$	$0.2 < \epsilon < 1$ $0 < Re < 1.2 \times 10^5$ $1.9 \times 10^{-5} < D_p(m) < 3 \times 10^{-3}$
15	$0.02 < Re < 1000$	$0 < Re < 1.2 \times 10^5$
23	$0 <$ Re[a] $< 10^2$ $0.35 < \epsilon < 1.0$	$0 <$ Re[a] $< 1.0 \times 10^2$ $0.2 < \epsilon < 1$
24,25	$1.0 \times 10^{-1} <$ Re[a] $< 2.0 \times 10^2$ $1.6 \times 10^{-3} < D_r(m) < 4.8 \times 10^{-3}$ $0.611 < \epsilon < 0.862$	$0 <$ Re[a] $< 1.0 \times 10^2$ $1.9 \times 10^{-5} < D_r(m) < 3 \times 10^{-3}$ $0.2 < \epsilon < 1$
26	$0.6 < \epsilon < 1$ $1.1 \times 10^{-2} <$ Re[a] $< 1.7 \times 10^{-1}$	$0.2 < \epsilon < 1$ $0 <$ Re[a] $< 1.0 \times 10^2$
27	$2 \times 10^{-4} < d(m) < 2 \times 10^{-2}$ $0 < \epsilon < 1$ $0 < Se < 1$	$1.9 \times 10^{-5} < d(m) < 3 \times 10^{-3}$ $0.2 < \epsilon < 1$ $0 < Se < 1$

---

[a] Reynolds number is defined in Equation (22).

3. DESCRIPTION AND IMPLEMENTATION OF COMPUTER MODELS

3.1 Implementation in SCDAP

The debris bed characterization and thermal behavior models described in this report will be employed by the DBUNDL subroutine in SCDAP. DBUNDL controls analysis of a disrupted fuel bundle. The functional decomposition of DBUNDL, shown in Table 2, illustrates the relationship of the subroutines (models) which are used to perform analysis of a disrupted bundle. The models described in Section 2 are contained in the subroutines DBFROZ and DBFRAG and the subroutines accessed by these subroutines. Subroutines DBFROZ and DBFRAG serve as the driver programs which direct the logic flow for the debris behavior models associated with the cohesive and rubble beds, respectively. The hierarchy of the subroutines called by DBFROZ and DBFRAG is shown in Tables 3 and 4, respectively. As shown by Tables 3 and 4, several of the subroutines are common to both DBFROZ and DBFRAG.

A driver program, DBDRIV, was developed to test DBFROZ and DBFRAG. In addition to calling DBFROZ and DBFRAG, the driver calls three other subroutines: DEBINP, DBTIME, and DBOUTD. The general purposes of DBTIME and DBOUTD are described in Table 2. The purpose of DEBINP is to provide input for DBFROZ and DBFRAG by simulating output from DBTRAN and DBREGN. Although DEBINP, DBTIME, and DBOUTD are routines that form part of the SCDAP functional decomposition, these same routines which have been developed for the driver, DBDRIV, are only temporary. However, they are comparable to what will be used by SCDAP and help define the final requirements for the SCDAP routines.

3.2 Description of Computer Models

FORTRAN listings of the subroutines included in DBFROZ and DBFRAG are given in Appendix A of this report. Also, FORTRAN listings of the driver program, DBDRIV, and the subroutines DEBINP, DBTIME, and DBOUTD are

TABLE 2. DBUNDL FUNCTIONAL DECOMPOSITION

Subroutine	Function
DBUNDL	Disrupted bundle analysis
├─ DBOUT	Disrupted bundle output
│ └─ DBOUTD	Debris region output
├─ DBTIME	Disrupted bundle time step
├─ DBANA	Disrupted bundle average behavior
├─ DBREGN	Region boundaries
├─ DBTRAN	Region boundary conditions
├─ DBNTAC	Intact bundle region analysis
├─ DBFRAG	Rubble debris analysis
│ ├─ FGTIME	Rubble debris time step
│ ├─ FGCHAR	Debris bed characteristics
│ ├─ FRAGTH	Debris thermally related behavior
│ ├─ FGMELT	Melting front propagation
│ └─ FGESTT	Boundary disruption time estimation
└─ DBFROZ	Cohesive debris analysis
├─ FRTIME	Cohesive debris time step
├─ FRCHAR	Debris bed characteristics
├─ FROZTH	Debris thermally related behavior
├─ FRMELT	Melting front propagation
└─ FRESTT	Boundary disruption time estimation

TABLE 3. DESCRIPTION OF DBFROZ

Subroutine	Function
DBFROZ	
FRTIME	Determines time steps for cohesive debris bed analysis
FRCHAR	Computes debris bed characteristics and debris hydraulic behavior
FROZTH	Calculates debris bed and coolant temperatures and thermally-related behavior
DRYOUT	Computes dryout heat flux of debris bed
USMNMX[a]	Finds largest value in a vector
REGMOD	Modifies region identification number and temperature distribution along debris bed
ICSCCU[a]	Interpolation subroutines
ICSEVU[a]	
TEMPSR	Solves for debris bed and coolant temperature
LSGECO[a]	Factors a real matrix by Gaussian elimination and estimates the condition of the matrix
LSGESL[a]	Solves the real system A*X=B
CHEMHT[b]	Computes Zr-steam reaction heat and H_2 release
NUCLHT[b]	Computes debris decay power
GASRLS[b]	Computes fission product release
FRMELT	Calculates movement of melt front in cohesive debris bed
FRESTT	Estimates time when melt front will reach lower boundary of cohesive debris bed

---

a. Utility or library subroutine.

b. Currently inactive subroutine.

TABLE 4. DESCRIPTION OF DBFRAG

Subroutine	Function
DBFRAG	
├─FGTIME	Determines time steps for rubble debris bed analysis
├─FGCHAR	Computes debris bed characteristics and debris hydraulic behavior
├─FRAGTH	Calculates debris bed and coolant temperatures and thermally-related behavior
│ ├─DRYOUT	Computes dryout heat flux of debris bed
│ │ └─USMNMX[a]	Finds largest value in a vector
│ ├─REGMOD	Modifies region identification number and temperature distribution along debris bed
│ │ ├─ICSCCU[a]	Interpolation subroutines
│ │ └─ICSEVU[a]	
│ ├─TEMPSR	Solves for debris bed and coolant temperature
│ │ ├─LSGECO[a]	Factors a real matrix by Gaussian elimination and estimates the condition of the matrix
│ │ └─LSGESL[a]	Solves the real system A*X=B
│ ├─CHEMHT[b]	Computes Zr-steam reaction heat and H_2 release
│ ├─NUCLHT[b]	Computes debris decay power
│ └─GASRLS[b]	Computes fission product release
├─FGMELT	Calculates movement of melt front in rubble debris bed
└─FGESTT	Estimates time when melt front will reach lower boundary of rubble debris bed

---

a. Utility or library subroutine.

b. Currently inactive subroutine.

given in Appendix A. The input and output variables for DBFROZ and DBFRAG and the subroutines called by DBFROZ and DBFRAG are listed in Tables 5 through 19.

TABLE 5. INPUT/OUTPUT DESCRIPTION FOR DBFROZ

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ABSMC	M_{Ab}	TOTAL MASS OF CONTROL ROD MATERIAL ACCUMULATED IN COHESIVE DEBRIS REGION	KG
INPUT	ALPHAC	(1)	EFFECTIVE ALPHA-ZR REACTION AREA	M2
INPUT	ALPHTC	(1)	EFFECTIVE ALPHA-ZR REACTION LAYER THICKNESS	M
INPUT	AREAR	A_c	TOTAL BUNDLE CROSS SECTIONAL AREA IN DEBRIS REGION	M2
INPUT	DELTT	Δt	TIME STEP	S
INPUT	ELVNC	(2)	ELEVATION OF POROUS BODY-FROM THE BOTTOM OF ROD BUNDLE TO THE BOTTOM OF POROUS BODY REGION	M
INPUT	HITEC	H_c	FROZEN DEBRIS REGION HEIGHT FROM DBREGN	M
INPUT	HTLBCC	HTLBCC	HEAT TRANSFER INTO COHESIVE DEBRIS COOLANT AT LOWER BOUNDARY	W/M2
INPUT	HTLBCD	HTLBCD	HEAT TRANSFER INTO COHESIVE DEBRIS BED TO LOWER BOUNDARY	W/M2
INPUT	HTUBCC	HTUBCC	HEAT TRANSFER INTO COHESIVE DEBRIS COOLANT AT UPPER BOUNDARY	W/M2
INPUT	HTUBCD	HTUBCD	HEAT TRANSFER INTO COHESIVE DEBRIS BED AT UPPER BOUNDARY	W/M2
INPUT	I	(2)	INDEX OF CURRENT TIME FROM TIME ARRAY	
INPUT	MAXL	(2)	MAXIMUM NUMBER OF NODES IN THE DEBRIS REGION WHICH COMBINED LIQUID, SATURATION, AND VAPOR REGION	
INPUT	ND	nd	NUMBER OF NODES TO BE USED IN LIQUID OR VAPOR REGION FOR DEBRIS REGION	
INPUT	NT	(2)	NUMBER OF TIME STEPS IN THE TIME ARRAY	

TABLE 5. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	STRMC	M_{St}	TOTAL MASS OF CORE STRUCTURAL MATERIALS ACCUMULATED IN COHESIVE DEBRIS BED	KG
INPUT	T	(2)	TIME STEP ARRAY	S
INPUT	UO2MC	M_{UO_2}	TOTAL MASS OF UO2 ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
INPUT	VINC	u_o	COOLANT VELOCITY AT BOTTOM OF COHESIVE DEBRIS REGION	M/S
INPUT	VINCOD	(2)	COOLANT VELOCITY AT BOTTOM OF COHESIVE DEBRIS REGION AT PREVIOUS TIME STEP	M/S
INPUT	XMASSC	(2)	TOTAL MASS OF DEBRIS ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
INPUT	ZRMC	M_{Zr}	TOTAL MASS OF ZR ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
INPUT	ZRO2AC	(1)	EFFECTIVE ZR-STEAM REACTION AREA	M2
INPUT	ZRO2MC	M_{ZrO_2}	TOTAL MASS OF ZRO2 ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
INPUT	ZRO2TC	(1)	EFFECTIVE ZRO2 REACTION LAYER THICKNESS	M
I/O	BEDTMP	T_p	DEBRIS BED TEMPERATURE CORRESPONDING TO AXIAL NODES	K
I/O	COLTMP	T_c	DEBRIS BED COOLANT TEMPERATURE CORRESPONDING TO AXIAL NODES	K
I/O	ELVAY	(2)	ELEVATION OF DEBRIS AXIAL NODES	M
I/O	FGRSC	(1)	FISSION GAS RELEASE DURING DELTT	MOLES
I/O	FISHC	(1)	FISSION DECAY HEAT GENERATION RATE	W/M3
I/O	IDREGN	IDREGN	DEBRIS REGION ID NUMBER OF LAST TIME STEP	

TABLE 5. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
I/O	NOND	NOND	NUMBER OF TOTAL NODES IN THE ANALYSIS	
I/O	OXDAC	(1)	ZR-STEAM HEAT GENERATION RATE	W/M3
I/O	VOLGSC	(1)	VOLATILE FISSION PRODUCTS RELEASE FURING DELTT	MOLE
OUTPUT	CURTIM	(2)	CURRENT TIME	S
OUTPUT	DOWNZ	(2)	LENGTH OF MOLTEN POOL PENETRATION IN COHESIVE DEBRIS	M
OUTPUT	EFFDIA	D_c	EFFECTIVE PARTICLE DIAMETER DERIVED FROM SPECIFIC SURFACE AREA	M
OUTPUT	ELVMC	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
OUTPUT	ESTOT	(2)	ESTIMATED TIME OF DISRUPTION	S
OUTPUT	HYDGC	(1)	HYDROGEN GENERATION DURING DELTT	MOLES
OUTPUT	IMT	(2)	COHESIVE DEBRIS REGION TYPE	
OUTPUT	KTERMC	(2)	FLAG TO INDICATE NORMALITY OF RUN	
OUTPUT	POROPC	ΔP	PRESSURE DROP ACROSS THE COHESIVE BED	PA
OUTPUT	POROSC	ε_c	AVERAGE COHESIVE BED POROSITY	
OUTPUT	RBDRPT	(2)	DISRUPTION FLAG	
OUTPUT	THMOTC	(2)	MOLTEN MATERIAL THICKNESS	M
OUTPUT	VELFTC	(2)	VELOCITY OF MOLTON POOL FLOWING DOWNWARD	M/S
OUTPUT	YCHRL	l_c	CHARACTERISTIC LENGTH OF FROZEN BED	M

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 6. INPUT/OUTPUT DESCRIPTION FOR DBFRAG

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ABSMR	M_{Ab}	TOTAL MASS OF CONTROL ROD MATERIAL ACCUMULATED IN RUBBLE DEBRIS BED	KG
INPUT	ALPHAR	(1)	EFFECTIVE ALPHA-ZR REACTION AREA	M2
INPUT	ALPHTR	(1)	EFFECTIVE ALPHA-ZR REACTION LAYER THICKNESS	M
INPUT	AREAR	A_r	TOTAL BUNDLE CROSS SECTIONAL AREA IN DEBRIS REGION	M2
INPUT	DELTT	Δt	TIME STEP	S
INPUT	ELVNR	(2)	ELEVATION OF RUBBLE BODY-FROM THE BOTTOM OF ROD BUNDLE TO THE BOTTOM OF RUBBLE BODY REGION	M
INPUT	HITRG	H_{rd}	FRAGMENTED DEBRIS REFION HEIGHT FROM DBREGN	M
INPUT	HTLBRC	HTLBCC	HEAT TRANSFER INTO RUBBLE DEBRIS COOLANT AT LOWER BOUNDARY	W/M2
INPUT	HTLBRD	HTLBCD	HEAT TRANSFER INTO RUBBLE DEBRIS BED AT LOWER BOUNDARY	W/M2
INPUT	HTUBRC	HTUBCC	HEAT TRANSFER INTO RUBBLE DEBRIS COOLANT AT UPPER BOUNDARY	W/M2
INPUT	HTUBRD	HTUBCD	HEAT TRANSFER INTO RUBBLE DEBRIS BED AT UPPER BOUNDARY	W/M2
INPUT	I	(2)	INDEX OF CURRENT TIME FROM TIME ARRAY	
INPUT	MAXL	(2)	MAXIMUM NUMBER OF NODES IN THE DEBRIS REGION WHICH COMBINED LIQUID, SATURATION,AND VAPOR REGION	
INPUT	ND	nd	NUMBER OF NODES TO BE USED IN LIQUID OR VAPOR REGION FOR DEBRIS REGION	
INPUT	NTIME	(2)	NUMBER OF TIME STEPS IN TIME ARRAY	

TABLE 6. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	SIZAVR	D_r	AVERAGE PARTICLE SIZE OVER ACCUMULATED MASS AND COMPONENTS IN RUBBLE DEBRIS BED	M
INPUT	STRMR	M_{St}	TOTAL MASS OF CORE STRUCTURE MATERIALS ACCUMULATED IN RUBBLE DEBRIS BED	KG
INPUT	T	(2)	TIME STEP ARRAY	S
INPUT	UO2MR	M_{UO_2}	TOTAL MASS OF UO2 ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG
INPUT	VINR	u_o	COOLANT INLET VELOCITY AT BOTTOM OF DEBRIS REGION	M/S
INPUT	VINROD	(2)	COOLANT INLET VELOCITY AT BOTTOM OF DEBRIS REGION AT PREVIOUS TIME STEP	M/S
INPUT	XMASSR	(2)	TOTAL MASS OF DEBRIS ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG
INPUT	ZRMR	M_{Zr}	TOTAL MASS OF ZIRCALOY ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG
INPUT	ZRO2MR	M_{ZrO_2}	TOTAL MASS OF ZRO2 ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG
INPUT	ZRO2TR	(1)	EFFECTIVE ZRO2 REACTION LAYER THICKNESS	M
I/O	BEDTMR	T_p	DEBRIS TEMPERATURE CORRESPONDING TO NONR NODES	K
I/O	COLTMR	T_c	COOLANT TEMPERATURE CORRESPONDING TO NONR NODES	K
I/O	ELVAR	(2)	ELEVATION OF AXIAL NODES IN DEBRIS	
I/O	ELVMR	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
I/O	FISHR	(1)	FISSION/DECAY HEAT GENERATION RATE	W/M3
I/O	IDREGR	IDREGN	REGION TYPE ID	

TABLE 6. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
I/O	NONR	NOND	NUMBER OF AXIAL NODES USED IN DEBRIS REGION ANALYSIS	
I/O	OXDAR	(1)	ZR-STEAM HEAT GENERATION RATE	W/M3
OUTPUT	CURTIM	(2)	CURRENT TIME	S
OUTPUT	DOWNR	(2)	LENGTH OF MOLTEN POOL PENETRATION	M
OUTPUT	ESTDT	(2)	ESTIMATED TIME OF DISRUPTION	S
OUTPUT	HITER	H_r, L_f	RUBBLE BED HEIGHT	M
OUTPUT	IRT	(2)	DEBRIS REGION TYPE	
OUTPUT	KFLUID	(2)	FLUIDIZATION FLAG, =0 PACKED BED, =1 FLUIDIZED BED	
OUTPUT	KTERM	(2)	ABNORMAL TERMINATION FLAG	
OUTPUT	PDROPR	ΔP	PRESSURE DROP ACROSS THE RUBBLE DEBRIS BED	PA
OUTPUT	POROSR	ε_r	AVERAGE RUBBLE BED POROSITY	
OUTPUT	PBDRPT	(2)	DISRUPTION FLAG	
OUTPUT	THMOTR	(2)	MOLTEN MATERIAL THICKNES	M
OUTPUT	VELMOT	(2)	VELOCITY OF MOLTEN POOL FLOWING DOWNWARD	M/S
OUTPUT	XCHRL	l_r	CHARACTERISTIC LENGTH OF THE PARTICLE IN RUBBLE DEBRIS BED	M

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 7. INPUT/OUTPUT DESCRIPTION FOR FRTIME

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	DELTIN	(2)	TIME STEP CALCULATED IN THE DISRUPTED BUNDLE TIME STEP SUBROUTINE	S
INPUT	NSTP	(2)	NUMBER OF STEPS IN WHICH DISRUPTED BUNDLE TIME STEP WILL BE BROKEN	
OUTPUT	DELTOT	(2)	TIME STEP TO BE USED IN DEBRIS ANALYSIS	S

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 8. INPUT/OUTPUT DESCRIPTION FOR FRCHAR

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ABSMC	M_{Ab}	TOTAL MASS OF CONTROL ROD MATERIAL ACCUMULATED IN COHESIVE DEBRIS REGION	KG
INPUT	AREAR	A_c	TOTAL BUNDLE CROSS SECTIONAL AREA IN DEBRIS REGION	M2
INPUT	DELTC	Δt	TIME STEP FOR COHESIVE DEBRIS REGION	S
INPUT	HITEC	H_c	FROZEN DEBRIS REGION HEIGHT FROM DBREGN	M
INPUT	ROABS	ρ_{Ab}	DENSITY OF CONTROL ROD MATERIAL FOR DEBRIS REGION	KG/M3
INPUT	ROCOL	ρ_c	COOLANT DENSITY AT BOTTOM OF DEBRIS	KG/M3
INPUT	ROSTR	ρ_{St}	DENSITY OF STRUCTURAL MATERIAL FOR DEBRIS REGION	KG/M3
INPUT	ROUO2	ρ_{UO_2}	UO2 DENSITY	KG/M3
INPUT	ROZR	ρ_{Zr}	ZR DENSITY	KG/M3
INPUT	ROZRO2	ρ_{ZrO_2}	ZRO2 DENSITY	KG/M3
INPUT	STRMC	M_{St}	TOTAL MASS OF CORE STRUCTURAL MATERIALS ACCUMULATED IN COHESIVE DEBRIS BED	KG
INPUT	UO2MC	M_{UO_2}	TOTAL MASS OF UO2 ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
INPUT	VINC	u_o	COOLANT VELOCITY AT BOTTOM OF COHESIVE DEBRIS REGION	M/S
INPUT	VINCOD	(2)	COOLANT VELOCITY AT BOTTOM OF COHESIVE DEBRIS REGION AT PREVIOUS TIME STEP	M/S
INPUT	XMASSC	(2)	TOTAL MASS OF DEBRIS ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
INPUT	XMUCOL	μ	VISCOSITY OF COOLANT AT BOTTOM OF DEBRIS	KG/SEC/M

TABLE 8. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ZRMC	M_{Zr}	TOTAL MASS OF ZR ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
INPUT	ZRO2MC	M_{ZrO_2}	TOTAL MASS OF ZRO2 ACCUMULATED FOR ALL COMPONENTS IN COHESIVE DEBRIS BED	KG
OUTPUT	EFFDIA	D_c	EFFECTIVE PARTICLE DIAMETER DERIVED FROM SPECIFIC SURFACE AREA	M
OUTPUT	KTERMC	(2)	FLAG TO INDICATE NORMALITY OF RUN	
OUTPUT	PDROPC	ΔP	PRESSURE DROP ACROSS THE COHESIVE BED	PA
OUTPUT	POROSC	ε_c	AVERAGE COHESIVE BED POROSITY	
OUTPUT	YCHRL	l_r	CHARACTERISTIC LENGTH OF FROZEN BED	M

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 9. INPUT/OUTPUT DESCRIPTION FOR FROZTH

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ALPHAC	(1)	EFFECTIVE ALPHA-ZR REACTION AREA	M2
INPUT	ALPHTC	(1)	EFFECTIVE ALPHA-ZR REACTION LAYER THICKNESS	M
INPUT	BEDTMP	T_p	DEBRIS BED TEMPERATURE CORRESPONDING TO AXIAL NODES	K
INPUT	COLTMP	T_c	DEBRIS BED COOLANT TEMPERATURE CORRESPONDING TO AXIAL NODES	K
INPUT	DELTT	Δt	TIME STEP	S
INPUT	EFFDIA	D_c	EFFECTIVE PARTICLE DIAMETER DERIVED FROM SPECIFIC SURFACE AREA	M
INPUT	ELVAY	(2)	ELEVATION OF DEBRIS AXIAL NODES	M
INPUT	ELVNC	(2)	ELEVATION OF POROUS BODY-FROM THE BOTTOM OF ROD BUNDLE TO THE BOTTOM OF POROUS BODY REGION	M
INPUT	FISHC	(1)	FISSION DECAY HEAT GENERATION RATE	W/M3
INPUT	HITEC	H_c	FROZEN DEBRIS REGION HEIGHT FROM DBREGN	M
INPUT	HTLBCC	HTLBCC	HEAT TRANSFER INTO COHESIVE DEBRIS COOLANT AT LOWER BOUNDARY	W/M2
INPUT	HTLBCD	HTLBCD	HEAT TRANSFER INTO COHESIVE DEBRIS BED TO LOWER BOUNDARY	W/M2
INPUT	HTUBCC	HTUBCC	HEAT TRANSFER INTO COHESIVE DEBRIS COOLANT AT UPPER BOUNDARY	W/M2
INPUT	HTUBCD	HTUBCD	HEAT TRANSFER INTO COHESIVE DEBRIS BED AT UPPER BOUNDARY	W/M2
INPUT	IDREGN	IDREGN	DEBRIS REGION ID NUMBER OF LAST TIME STEP	
INPUT	MAXL	(2)	MAXIMUM NUMBER OF NODES IN THE DEBRIS REGION WHICH COMBINED LIQUID, SATURATION,AND VAPOR REGION	

TABLE 9. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ND	nd	NUMBER OF NODES TO BE USED IN LIQUID OR VAPOR REGION FOR DEBRIS REGION	
INPUT	NOND	NOND	NUMBER OF TOTAL NODES IN THE ANALYSIS	
INPUT	OXDAC	(1)	ZR-STEAM HEAT GENERATION RATE	W/M3
INPUT	POROSC	ε_c	AVERAGE COHESIVE BED POROSITY	
INPUT	VINC	u_o	COOLANT VELOCITY AT BOTTOM OF COHESIVE DEBRIS REGION	M/S
INPUT	YCHRL	l_c	CHARACTERISTIC LENGTH OF FROZEN BED	M
INPUT	ZRO2AC	(1)	EFFECTIVE ZR-STEAM REACTION AREA	M2
INPUT	ZRO2TC	(1)	EFFECTIVE ZRO2 REACTION LAYER THICKNESS	M
OUTPUT	BEDTMP	T_p	DEBRIS BED TEMPERATURE CORRESPONDING TO AXIAL NODES	K
OUTPUT	COLTMP	T_c	DEBRIS BED COOLANT TEMPERATURE CORRESPONDING TO AXIAL NODES	K
OUTPUT	DOWNZ	(2)	LENGTH OF MOLTEN POOL PENETRATION IN COHESIVE DEBRIS	M
OUTPUT	ELVAY	(2)	ELEVATION OF DEBRIS AXIAL NODES	M
OUTPUT	ELVMC	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
OUTPUT	FGRSC	(1)	FISSION GAS RELEASE DURING DELTT	MOLES
OUTPUT	FISHC	(1)	FISSION DECAY HEAT GENERATION RATE	W/M3
OUTPUT	HYDGC	(1)	HYDROGEN GENERATION DURING DELTT	MOLES
OUTPUT	IDREGN	IDREGN	DEBRIS REGION ID NUMBER OF LAST TIME STEP	
OUTPUT	IMT	(2)	COHESIVE DEBRIS REGION TYPE	

TABLE 9. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
OUTPUT	NOND	NOND	NUMBER OF TOTAL NODES IN THE ANALYSIS	
OUTPUT	OXDAC	(1)	ZR-STEAM HEAT GENERATION RATE	W/M3
OUTPUT	THMOTC	(2)	MOLTEN MATERIAL THICKNESS	M
OUTPUT	VOLGSC	(1)	VOLATILE FISSION PRODUCTS RELEASE DURING DELTT	MOLE

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 10. INPUT/OUTPUT DESCRIPTION FOR FRMELT

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	DELTC	Δt	TIME STEP FOR COHESIVE DEBRIS REGION	S
INPUT	ELVAY	(2)	ELEVATION OF DEBRIS AXIAL NODES	M
INPUT	ELVMC	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
INPUT	IMT	(2)	COHESIVE DEBRIS REGION TYPE	
INPUT	NOND	NOND	NUMBER OF TOTAL NODES IN THE ANALYSIS	
OUTPUT	VELMTC	(2)	VELOCITY OF MOLTON POOL FLOWING DOWNWARD	M/S

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 11. INPUT/OUTPUT DESCRIPTION FOR FRESTT

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	CURTIM	(2)	CURRENT TIME	S
INPUT	ELVMC	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
INPUT	ELVNC	(2)	ELEVATION OF POROUS BODY-FROM THE BOTTOM OF ROD BUNDLE TO THE BOTTOM OF POROUS BODY REGION	M
INPUT	VELMTC	(2)	VELOCITY OF MOLTON POOL FLOWING DOWNWARD	M/S
OUTPUT	ESTDT	(2)	ESTIMATED TIME OF DISRUPTION	S

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 12. INPUT/OUTPUT DESCRIPTION FOR FGTIME

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	DELTIN	(2)	TIME STEP CALCULATED IN THE DISRUPTED BUNDLE TIME STEP SUBROUTINE	S
INPUT	NSTP	(2)	NUMBER OF STEPS IN WHICH DISRUPTED BUNDLE TIME STEP WILL BE BROKEN	
OUTPUT	DELTOT	(2)	TIME STEP TO BE USED IN DEBRIS ANALYSIS	S

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 13. INPUT/OUTPUT DESCRIPTION FOR FGCHAR

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ABSMR	M_{Ab}	TOTAL MASS OF CONTROL ROD MATERIAL ACCUMULATED IN RUBBLE DEBRIS BED	KG
INPUT	AREAR	A_r	TOTAL BUNDLE CROSS SECTIONAL AREA IN DEBRIS REGION	M2
INPUT	DELTR	Δt	TIME STEP FOR DEBRIS REGION	S
INPUT	HITRG	H_{rd}	FRAGMENTED DEBRIS REFION HEIGHT FROM DBREGN	M
INPUT	ROABS	ρ_{Ab}	DENSITY OF CONTROL ROD MATERIAL FOR DEBRIS REGION	KG/M3
INPUT	ROCOL	ρ_c	COOLANT DENSITY AT BOTTOM OF DEBRIS	KG/M3
INPUT	ROSTR	ρ_{St}	DENSITY OF STRUCTURAL MATERIAL FOR DEBRIS REGION	KG/M3
INPUT	ROUO2	ρ_{UO_2}	UO2 DENSITY	KG/M3
INPUT	ROZR	ρ_{Zr}	ZR DENSITY	KG/M3
INPUT	ROZRO2	ρ_{ZrO_2}	ZRO2 DENSITY	KG/M3
INPUT	SIZAVR	D_r	AVERAGE PARTICLE SIZE OVER ACCUMULATED MASS AND COMPONENTS IN RUBBLE DEBRIS BED	M
INPUT	STRMR	M_{St}	TOTAL MASS OF CORE STRUCTURE MATERIALS ACCUMULATED IN RUBBLE DEBRIS BED	KG
INPUT	UO2MR	M_{UO_2}	TOTAL MASS OF UO2 ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG
INPUT	VINR	u_o	COOLANT INLET VELOCITY AT BOTTOM OF DEBRIS REGION	M/S
INPUT	VINROD	(2)	COOLANT INLET VELOCITY AT BOTTOM OF DEBRIS REGION AT PREVIOUS TIME STEP	M/S
INPUT	XMASSR	(2)	TOTAL MASS OF DEBRIS ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG

TABLE 13. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	XMUCOL	μ	VISCOSITY OF COOLANT AT BOTTOM OF DEBRIS	KG/SEC/M
INPUT	ZRMR	M_{Zr}	TOTAL MASS OF ZIRCALOY ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG
INPUT	ZRO2MR	M_{ZrO_2}	TOTAL MASS OF ZRO2 ACCUMULATED FOR ALL COMPONENTS IN RUBBLE DEBRIS BED	KG
OUTPUT	HITER	L_f, H_r	RUBBLE BED HEIGHT	M
OUTPUT	KFLUID	(2)	FLUIDIZATION FLAG, =0 PACKED BED, =1 FLUIDIZED BED	
OUTPUT	KTERM	(2)	ABNORMAL TERMINATION FLAG	
OUTPUT	PORDPR	ΔP	PRESSURE DROP ACROSS THE RUBBLE DEBRIS BED	PA
OUTPUT	POROSR	ε_r	AVERAGE RUBBLE BED POROSITY	
OUTPUT	XCHRL	l_r	CHARACTERISTIC LENGTH OF THE PARTICLE IN RUBBLE DEBRIS BED	M

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 14. INPUT/OUTPUT DESCRIPTION FOR FRAGTH

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ALPHAR	(1)	EFFECTIVE ALPHA-ZR REACTION AREA	M2
INPUT	ALPHTR	(1)	EFFECTIVE ALPHA-ZR REACTION LAYER THICKNESS	M
INPUT	AREAR	A_r	TOTAL BUNDLE CROSS SECTIONAL AREA IN DEBRIS REGION	M2
INPUT	DELTR	Δt	TIME STEP FOR DEBRIS REGION	S
INPUT	ELVNR	(2)	ELEVATION OF RUBBLE BODY-FROM THE BOTTOM OF ROD BUNDLE TO THE BOTTOM OF RUBBLE BODY REGION	M
INPUT	HITER	H_r, L_f	RUBBLE BED HEIGHT	M
INPUT	HTLBRC	HTLBCC	HEAT TRANSFER INTO RUBBLE DEBRIS COOLANT AT LOWER BOUNDARY	W/M2
INPUT	HTLBRD	HTLBCD	HEAT TRANSFER INTO RUBBLE DEBRIS BED AT LOWER BOUNDARY	W/M2
INPUT	HTUBRC	HTUBCC	HEAT TRANSFER INTO RUBBLE DEBRIS COOLANT AT UPPER BOUNDARY	W/M2
INPUT	HTUBRD	HTUBCD	HEAT TRANSFER INTO RUBBLE DEBRIS BED AT UPPER BOUNDARY	W/M2
INPUT	KFLUID	(2)	FLUIDIZATION FLAG, =0 PACKED BED, =1 FLUIDIZED BED	
INPUT	MAXL	(2)	MAXIMUM NUMBER OF NODES IN THE DEBRIS REGION WHICH COMBINED LIQUID, SATURATION,AND VAPOR REGION	
INPUT	ND	nd	NUMBER OF NODES TO BE USED IN LIQUID OR VAPOR REGION FOR DEBRIS REGION	
INPUT	POROSR	ε_r	AVERAGE RUBBLE BED POROSITY	
INPUT	SIZAVR	D_r	AVERAGE PARTICLE SIZE OVER ACCUMULATED MASS AND COMPONENTS IN RUBBLE DEBRIS BED	M

TABLE 14. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	ZINR	u_o	COOLANT INLET VELOCITY AT BOTTOM OF DEBRIS REGION	M/S
INPUT	XCHRL	l_r	CHARACTERISTIC LENGTH OF THE PARTICLE IN RUBBLE DEBRIS BED	M
INPUT	ZRO2AR	(1)	EFFECTIVE ZR-STEAM REACTION AREA	M2
INPUT	ZRO2TR	(1)	EFFECTIVE ZRO2 REACTION LAYER THICKNESS	M
OUTPUT	BEDTMR	T_p	DEBRIS TEMPERATURE CORRESPONDING TO NONR NODES	
OUTPUT	COLTMR	T_c	COOLANT TEMPERATURE CORRESPONDING TO NONR NODES	
OUTPUT	DOWNR	(2)	LENGTH OF MOLTEN POOL PENETRATION	M
OUTPUT	ELVAR	(2)	ELEVATION OF AXIAL NODES IN DEBRIS	
OUTPUT	ELVMR	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
OUTPUT	FGRSR	(2)	FISSION GAS RELEASE (MOLES) DURING DELTR	
OUTPUT	FISHR	(1)	FISSION/DECAY HEAT GENERATION RATE	W/M3
OUTPUT	HYDGR	(1)	HYDROGEN GENERATION DURING DELTR	MOLES
OUTPUT	IDREGR	IDREGN	REGION TYPE ID	
OUTPUT	IRT	(2)	DEBRIS REGION TYPE	
OUTPUT	NONR	NOND	NUMBER OF AXIAL NODES USED IN DEBRIS REGION ANALYSIS	
OUTPUT	OXDAR	(1)	AR-STEAM HEAT GENERATION RATE	W/M3
OUTPUT	TESTBB	(2)	ESTIMATED TIME STEP FOR LOWER BOUNDARY BREAKTHROUGH AFTER CURRENT TIME	S
OUTPUT	THMOTR	(2)	MOLTEN MATERIAL THICKNES	M

TABLE 14. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
OUTPUT	VELMLT	(2)	VELOCITY OF MOLTEN POOL FLOWING DOWNWARD	M/S
OUTPUT	VOLGSR	(1)	VOLATILE FISSION PRODUCTS RELEASE DURING DELTR	MOLE

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 15. INPUT/OUTPUT DESCRIPTION FOR FGMELT

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	DELTR	Δt	TIME STEP FOR DEBRIS REGION	S
INPUT	ELVAR	(2)	ELEVATION OF AXIAL NODES IN DEBRIS	
INPUT	ELVMR	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
INPUT	IRT	(2)	DEBRIS REGION TYPE	
INPUT	NONR	NOND	NUMBER OF AXIAL NODES USED IN DEBRIS REGION ANALYSIS	
OUTPUT	VELMOT	(2)	VELOCITY OF MOLTEN POOL FLOWING DOWNWARD	M/S

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 16. INPUT/OUTPUT DESCRIPTION FOR FGESTT

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	CURTIM	(2)	CURRENT TIME	S
INPUT	ELVMR	(2)	MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY	M
INPUT	ELVNR	(2)	ELEVATION OF RUBBLE BODY-FROM THE BOTTOM OF ROD BUNDLE TO THE BOTTOM OF RUBBLE BODY REGION	M
INPUT	VELMOT	(2)	VELOCITY OF MOLTEN POOL FLOWING DOWNWARD	M/S
OUTPUT	ESTDT	(2)	ESTIMATED TIME OF DISRUPTION	S

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 17. INPUT/OUTPUT DESCRIPTION FOR DRYOUT

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	EFFDIA	d	EFFECTIVE PARTICLE DIAMETER DERIVED FROM SPECIFIC SURFACE AREA	M
INPUT	HFG	h_{fg}	LATENT HEAT OF VAPORIZATION	J/KG
INPUT	HITEC	H_c	FROZEN DEBRIS REGION HEIGHT FROM DBREGN	M
INPUT	PDROPC	ΔP	PRESSURE DROP ACROSS THE COHESIVE BED	PA
INPUT	POROSC	ε	AVERAGE COHESIVE BED POROSITY	
INPUT	QVOL	Q'''	VOLUMETRIC HEAT GENERATION RATE	W/M3
INPUT	ROCOL	ρ_l	COOLANT DENSITY AT BOTTOM OF DEBRIS	KG/M3
INPUT	ROVAP	ρ_v	VAPOR DENSITY	KG/M3
INPUT	SURTC	σ	SURFACE TENSION OF COOLANT	PA
INPUT	VINC	u_o	COOLANT VELOCITY AT BOTTOM OF COHESIVE DEBRIS REGION	M/S
INPUT	XMUCOL	μ_l	VISCOSITY OF COOLANT AT BOTTOM OF DEBRIS	KG/SEC/M
INPUT	XMUVAP	μ_v	VAPOR VISCOSITY	KG/SE/M
OUTPUT	QDYOUT	q_d	DRYOUT HEAT FLUX	W/M2

---

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 18. INPUT/OUTPUT DESCRIPTION FOR REGMOD

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	BEDTMP	T_p	DEBRIS BED TEMPERATURE CORRESPONDING TO AXIAL NODES	K
INPUT	COLTMP	T_c	DEBRIS BED COOLANT TEMPERATURE CORRESPONDING TO AXIAL NODES	K
INPUT	ELVAY	(2)	ELEVATION OF DEBRIS AXIAL NODES	M
INPUT	ELVNC	(2)	ELEVATION OF POROUS BODY-FROM THE BOTTOM OF ROD BUNDLE TO THE BOTTOM OF POROUS BODY REGION	M
INPUT	HITEC	H_c	FROZEN DEBRIS REGION HEIGHT FROM DBREGN	M
INPUT	IDR	(2)	REGION ID NUMBER CALCULATED IN FROZTH OR FRAGTH	
INPUT	IDREGN	IDREGN	DEBRIS REGION ID NUMBER OF LAST TIME STEP	
INPUT	MAXL	(2)	MAXIMUM NUMBER OF NODES IN THE DEBRIS REGION WHICH COMBINED LIQUID, SATURATION,AND VAPOR REGION	
INPUT	ND	nd	NUMBER OF NODES TO BE USED IN LIQUID OR VAPOR REGION FOR DEBRIS REGION	
INPUT	NOND	NOND	NUMBER OF TOTAL NODES IN THE ANALYSIS	
INPUT	TSAT	T_{sat}	COOLANT SATURATION TEMPERATURE	K
INPUT	XLLIQ	L_{sc}	LIQUID REGION LENGTH	M
INPUT	XLSAT	L_{sa}	SATURATION REGION LENGTH	M
INPUT	XLVAP	L_{sp}	VAPOR REGION LENGTH	M
OUTPUT	BEDTMP	T_p	DEBRIS BED TEMPERATURE CORRESPONDING TO AXIAL NODES	K
OUTPUT	COLTMP	T_c	DEBRIS BED COOLANT TEMPERATURE CORRESPONDING TO AXIAL NODES	K

TABLE 18. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
OUTPUT	ELVAY	(2)	ELEVATION OF DEBRIS AXIAL NODES	M

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

TABLE 19. INPUT/OUTPUT DESCRIPTION FOR TEMPSR

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
INPUT	DELTT	Δt	TIME STEP	S
INPUT	DELTZ	Δz	DISTANCE BETWEEN AXIAL NODE	M
INPUT	HTLBCC	HTLBCC	HEAT TRANSFER INTO COHESIVE DEBRIS COOLANT AT LOWER BOUNDARY	W/M2
INPUT	HTLBCD	HTUBCD	HEAT TRANSFER INTO COHESIVE DEBRIS BED TO LOWER BOUNDARY	W/M2
INPUT	HTUBCC	HTUBCC	HEAT TRANSFER INTO COHESIVE DEBRIS COOLANT AT UPPER BOUNDARY	W/M2
INPUT	HTUBCD	HTUBCD	HEAT TRANSFER INTO COHESIVE DEBRIS BED AT UPPER BOUNDARY	W/M2
INPUT	HVSC	Hv	HEAT TRANSFER COEFFICIENT BETWEEN BED AND COOLANT, VOLUMETRIC	W/K/M3
INPUT	IDYOT	(2)	INDICATOR OF COOLANT STATE	
INPUT	N	nd	NUMBER OF AXIAL NODES	
INPUT	N2	$2 \cdot nd$	2N FOR SOLVING COOLANT AND BED TEMPERATURE SIMULTANEOUSLY	
INPUT	POROSC	ϵ	AVERAGE COHESIVE BED POROSITY	
INPUT	QVOL	Q'''	VOLUMETRIC HEAT GENERATION RATE	W/M3
INPUT	TCOLCO	T_c	COOLANT TEMPERATURE AT N NODES AT PREVIOUS TIME STEP	K
INPUT	TEMPCO	T_p	BED TEMPERATURE AT N NODES AT PREVIOUS TIME STEP	K
INPUT	VINC	u_o	COOLANT VELOCITY AT BOTTOM OF COHESIVE DEBRIS REGION	M/S

TABLE 19. (CONTINUED)

USE	FORTRAN NAME	MATH SYMBOL	DESCRIPTION	UNITS
OUTPUT	TCOLC	T_c	COOLANT TEMPERATURE	K
OUTPUT	TEMPC	T_p	BED TEMPERATURE	K

(1) Variables that are currently inactive.

(2) Variables are not shown in Section 2.

4. RESULTS OF MODEL ACCEPTANCE TESTING

Testing of models is a necessary part of the process of code development. Deficiencies in logic flow, algebraic errors, and coding errors can be detected through model testing. For the debris characterization and thermal behavior models described in this report, model testing began with a line-by-line verification of all FORTRAN statements. Then, an extensive set of computer runs were performed using all modeling options to fully exploit the possible logic flow paths. Finally, calculational results were evaluated for reasonableness. This was done for each major subroutine. Results of the acceptance testing are discussed in the following sections.

4.1 FRCHAR

Subroutine FRCHAR calculates bed porosity, a characteristic particle length for heat transfer, and pressure drop across a cohesive debris bed. Comparisons of the pressure drop across a porous body calculated with FRCHAR and with hand-calculations using Ergun's correlation [Equation (3)] for various coolant velocities and debris porosities are shown in Table 20. These data indicate that the code predictions of FHCHAR agree well with Ergun's correlation and that Ergun's correlation is properly implemented in FRCHAR.

4.2 FGCHAR

Subroutine FGCHAR calculates bed height, minimal fluidization velocity, porosity, a characteristic particle length for heat transfer, and pressure drop for a rubble debris bed. The same procedure has been used to check the pressure drop calculations of FGCHAR as was used for subroutine FRCHAR. The results which are shown in Table 21 show excellent agreement. Predictions of minimum fluidization velocity, fluidized bed height, bed porosity, and pressure drop across the bed made using FGCHAR have been checked using the two examples given in Reference 5. Test conditions of these two examples are summarized in Table 22. Comparisons of the FGCHAR results with the two

TABLE 20. TEST RESULTS OF FRCHAR

Coolant Velocity (m/s)	Debris Bed Porosity	ΔP (Pa) FRCHAR	ΔP (Pa) Hand Calculation
0.01	0.37	7.775×10^2	7.775×10^2
0.02	0.37	2.254×10^3	2.254×10^3
0.03	0.37	4.430×10^3	4.430×10^3
0.04	0.37	7.305×10^3	7.305×10^3
0.05	0.37	1.088×10^4	1.088×10^4
0.06	0.37	1.515×10^4	1.515×10^4
0.07	0.37	2.012×10^4	2.013×10^4
0.08	0.37	2.580×10^4	2.580×10^4
0.09	0.37	3.217×10^4	3.217×10^4
0.10	0.37	3.924×10^4	3.924×10^4
0.14	0.37	7.451×10^4	7.451×10^4
0.01	0.104	4.329×10^4	4.328×10^4
0.02	0.104	1.182×10^5	1.181×10^5
0.03	0.104	2.246×10^5	2.246×10^5
0.04	0.104	3.627×10^5	3.625×10^5
0.05	0.104	5.323×10^5	5.321×10^5
0.06	0.104	7.335×10^5	7.333×10^5
0.07	0.104	9.663×10^5	9.660×10^5
0.08	0.104	1.231×10^6	1.230×10^5
0.09	0.104	1.527×10^6	1.526×10^6
0.10	0.104	1.854×10^6	1.854×10^6
0.14	0.104	3.480×10^5	3.479×10^6

TABLE 21. TEST RESULTS OF FRCHAR

Coolant Velocity (m/s)	Debris Bed Porosity	ΔP (Pa)	
		FRCHAR	Hand Calculation
0.001	0.37	4.629×10^{1}	4.629×10^{1}
0.002	0.37	9.957×10^{1}	9.957×10^{1}
0.003	0.37	1.598×10^{2}	1.599×10^{2}
0.004	0.37	2.271×10^{2}	2.271×10^{2}
0.005	0.37	3.014×10^{2}	3.014×10^{2}
0.006	0.37	3.826×10^{2}	3.826×10^{2}
0.007	0.37	4.709×10^{2}	4.709×10^{2}
0.008	0.37	5.661×10^{2}	5.661×10^{2}
0.009	0.37	6.683×10^{2}	6.683×10^{2}
0.010	0.37	7.775×10^{2}	7.775×10^{2}
0.015	0.37	1.428×10^{3}	1.429×10^{3}
0.020	0.37	2.254×10^{3}	2.254×10^{3}
0.0001	0.45	8.503×10^{1}	8.503×10^{1}
0.0002	0.45	1.703×10^{2}	1.703×10^{2}
0.0003	0.45	2.557×10^{2}	2.557×10^{2}
0.0004	0.45	3.414×10^{2}	3.414×10^{2}
0.0005	0.45	4.272×10^{2}	4.272×10^{2}
0.0006	0.45	5.133×10^{2}	5.133×10^{2}
0.0007	0.45	5.996×10^{2}	5.996×10^{2}
0.0008	0.45	6.860×10^{2}	6.860×10^{2}
0.0009	0.45	7.727×10^{2}	7.727×10^{2}
0.0010	0.45	8.596×10^{2}	8.596×10^{2}
0.0020	0.45	1.740×10^{3}	1.740×10^{3}
0.0030	0.45	2.641×10^{3}	2.641×10^{3}
0.0040	0.45	3.563×10^{3}	3.563×10^{3}

TABLE 22. TEST CONDITIONS OF THE TWO EXAMPLES IN REFERENCE 32

	Example 1	Example 2
Fixed bed height, m	0.7366	0.7112
Fixed bed porosity	0.45	0.37
Particle average diameter, m	7.493×10^{-4}	4.4196×10^{-3}
Particle density, kg/m^3	2.547×10^3	1.602×10^3
Coolant density, kg/m^3	9.98×10^2	9.98×10^2
Coolant velocity, m/s	1.259×10^{-2}	1.244×10^{-1}
Coolant viscosity, kg/m•s	1.30×10^{-3}	1.0×10^{-3}

examples are shown in Table 23. The results in Tables 21 and 23 demonstrate that FGCHAR does calculate rubble debris bed characteristics and hydraulic behavior reasonably well for both the packed and fluidized bed configurations, respectively.

4.3 TEMPSR

Debris bed and coolant temperatures are solved simultaneously in the TEMPSR subroutine. Steady state and transient temperature solutions by TEMPSR have been checked by comparison with analytical solutions. Calculated debris bed and coolant temperatures versus time are shown in Figure 4 for the case where the debris bed temperature is initially 2 K greater than that of the overlying coolant. With good heat transfer between debris and coolant (heat transfer coefficient greater than 10^{13} $W/m^3 \cdot K$) and no heat generation within the debris, the debris bed and coolant temperatures would be expected to reach an equilibrium temperature state very rapidly. This result is shown to occur in Figure 4 which indicates the TEMPSR calculations have been properly implemented. A second check case was made by assuming a continuous heating of the bed without any heat transfer within the porous system. For such conditions, debris bed temperature increases proportionally to the heating time, as shown in Figure 5. Agreement between the TEMPSR result and the analytical solution is excellent which indicates proper implementation within TEMPSR.

4.4 REGMOD

As discussed in Section 2.2.1, different regions may be used in analyzing debris bed and coolant temperatures, depending on coolant temperature, heat generation rate within the debris, and coolant flow rate. During the temperature calculations, the debris bed and coolant temperature distributions are modified by subroutine REGMOD to correspond to a new nodalization if the assigned region identification number is changed. The temperature distribution of the debris bed and coolant should be identical before and after calling REGMOD although the node number, elevation, and

TABLE 23. COMPARISON OF FGCHAR PREDICTIONS WITH TEXTBOOK SOLUTIONS

	Example 1		Example 2	
	FGCHAR	Textbook Solution	FGCHAR	Textbook Solution
Minimum fluidization velocity, m/s	4.353×10^{-3}	4.302×10^{-3}	2.118×10^{-2}	2.147×10^{-2}
Expanded bed height, m	1.02	1.01	1.99	1.99
Expanded bed porosity	0.60	0.60	0.775	0.775
Pressure drop across the bed, Pa	6.16×10^3	6.15×10^3	2.654×10^3	2.652×10^3

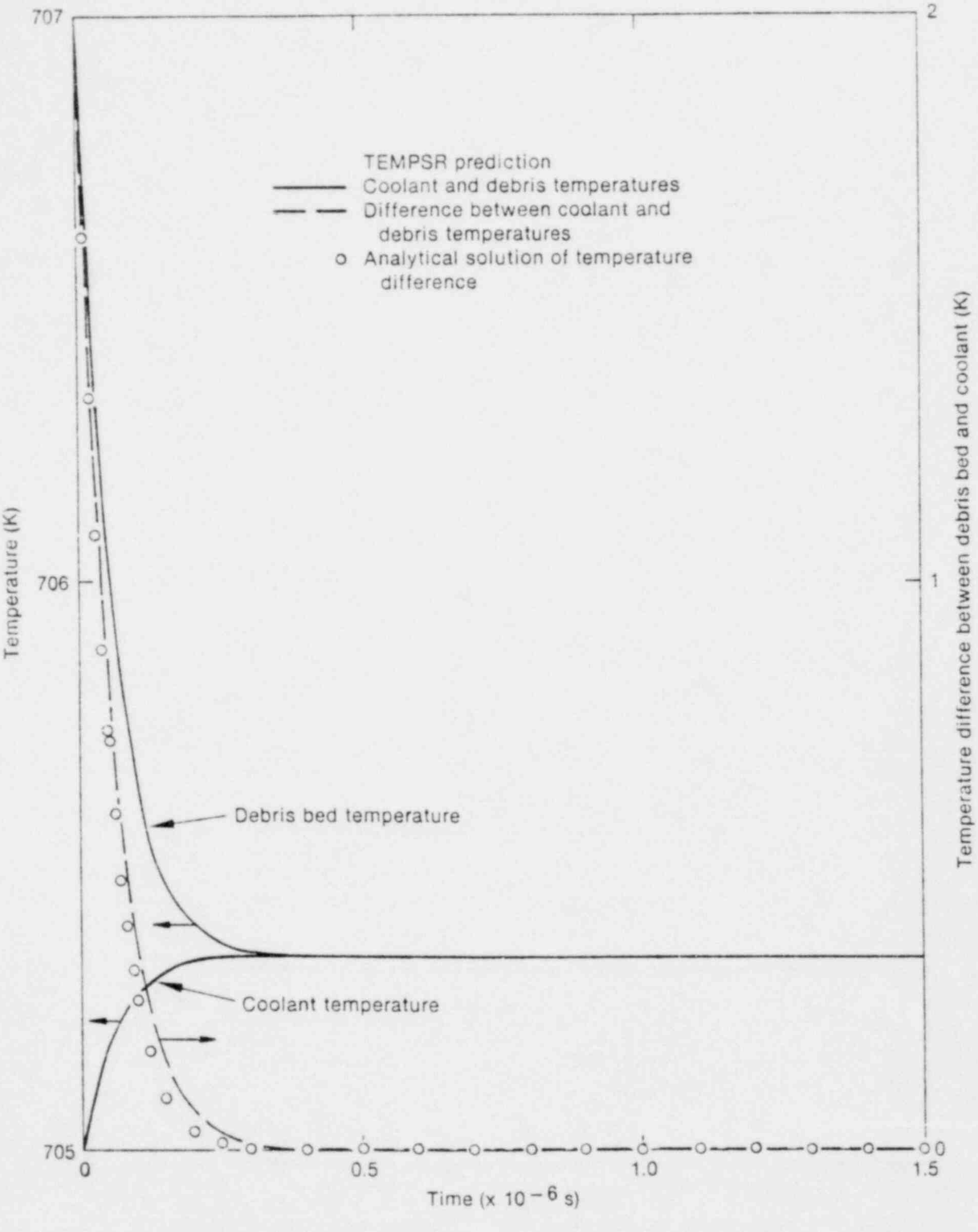


Figure 4. Comparison of TEMPSR prediction with analytical solution.

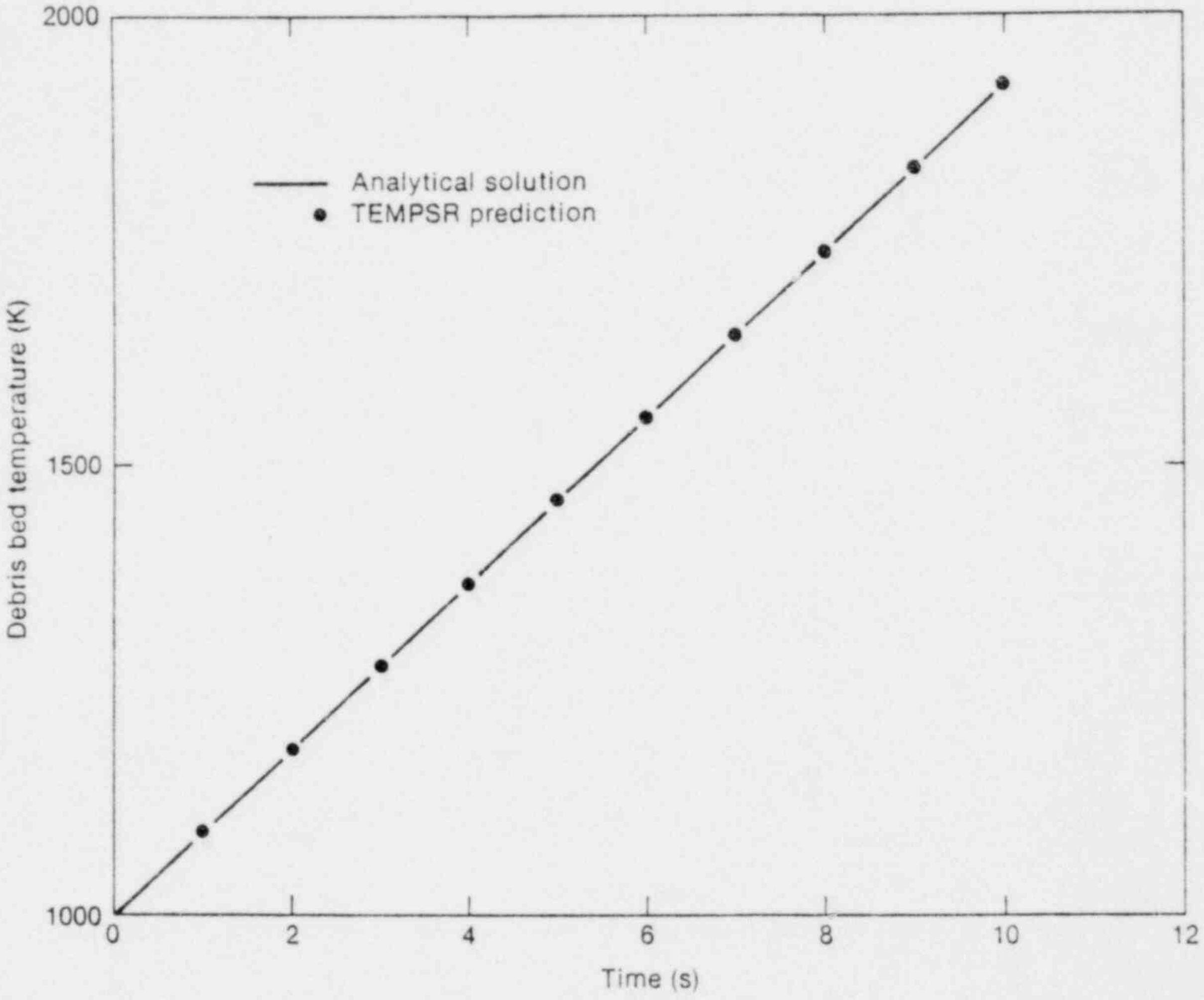


Figure 5. Comparison of TEMPSR prediction with analytical solution.

corresponding temperature may be changed. The interpolation capability of REGMOD for various transitions of region identification number are demonstrated by the results shown in Figures 6 and 7. The excellent agreement shown in Figures 6 and 7 indicates that the interpolation performed by REGMOD is correct.

4.5 FROZTH and FRAGTH

Subroutines FROZTH and FRAGTH compute debris and coolant temperatures and thermally related behavior (melting) of a cohesive and rubble debris bed, respectively. Calculated coolant and debris bed temperature distributions versus time are shown in Figure 8 for a debris bed with the following conditions:

1. Debris bed height of 0.[illegible] m
2. Constant volumetric heat generation rate of 3.38 x 10^3 W/m^3
3. Coolant velocity of 0.001 m/s
4. Bed porosity of 0.5208.

Initially the debris bed consists of subcooled, saturated, and superheated regions for the thermal calculations (i.e., IDREGN = 6). As heatup continues and the coolant temperatures exceed the saturation temperature, a superheated region with IDREGN = 2 is used for the thermal analysis.

The thermal response of a packed bed with the same characteristics as the cohesive bed discussed above is expected to be identical to that for the cohesive bed. This analysis was performed with the FRAGTH subroutine, and the results are shown in Figure 9. These results are identical to those shown in Figure 8 which indicates consistency between FROZTH and FRAGTH.

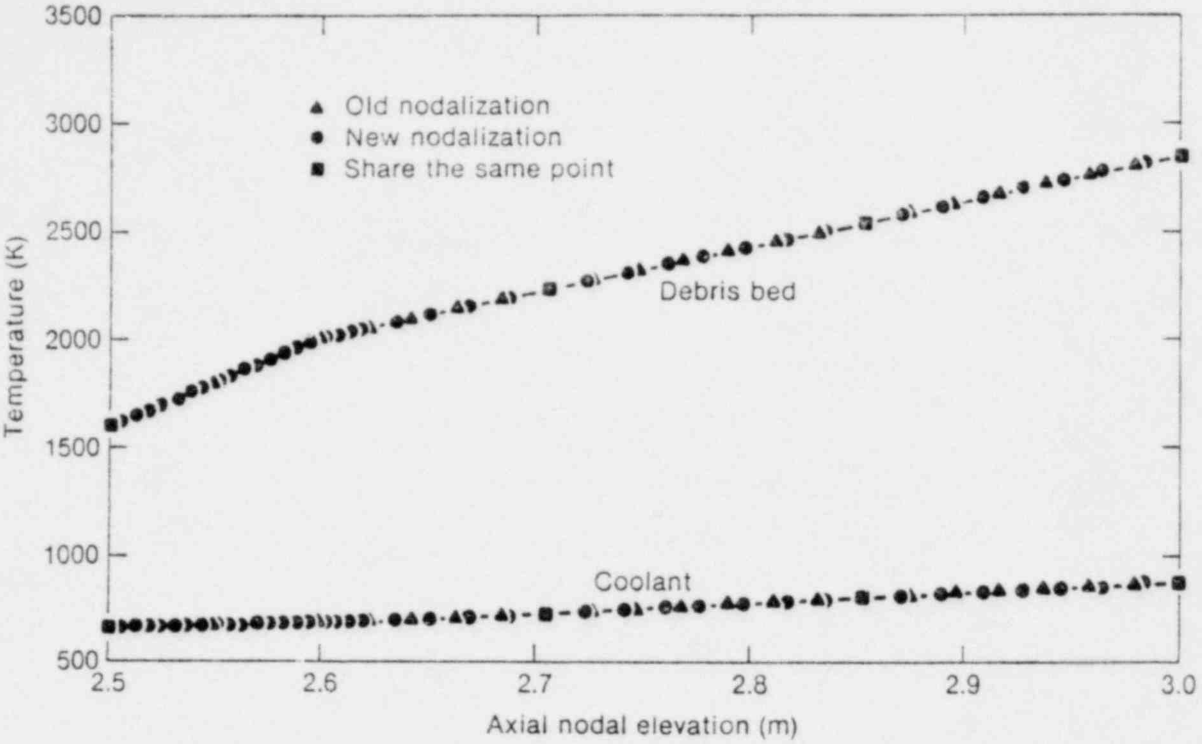


Figure 6. Interpolation of temperature distribution by REGMOD.

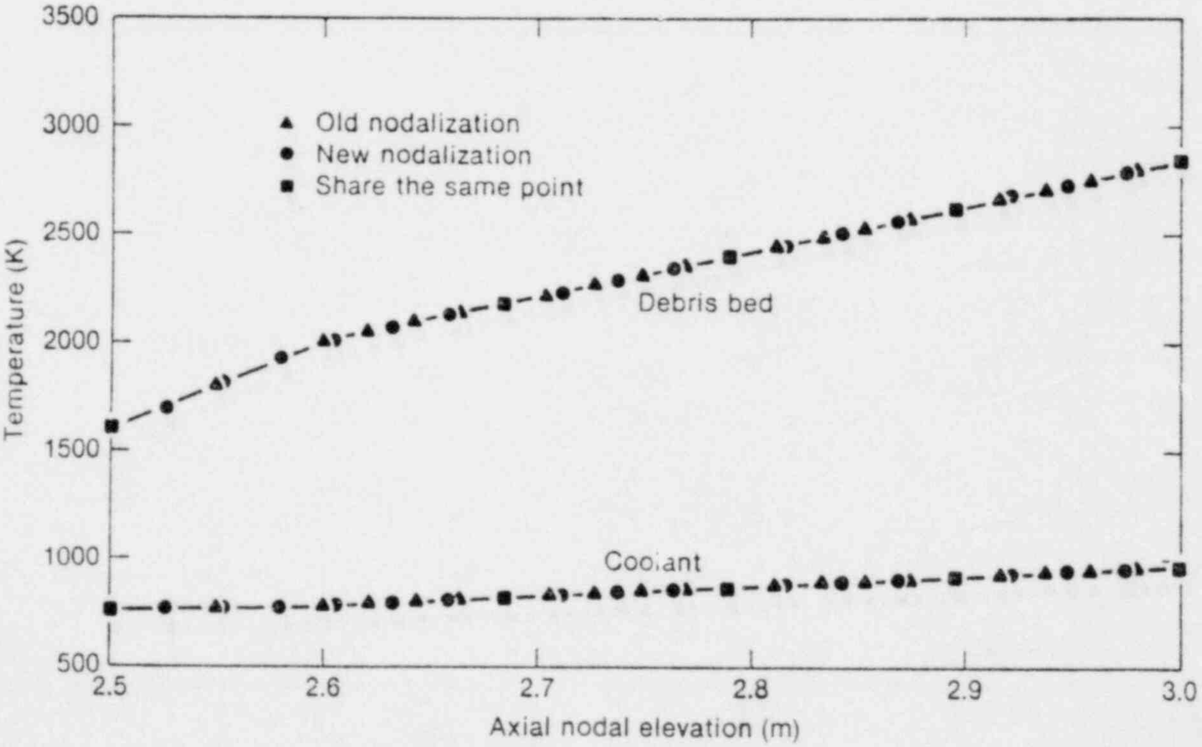


Figure 7. Interpolation of temperature distribution by REGMOD.

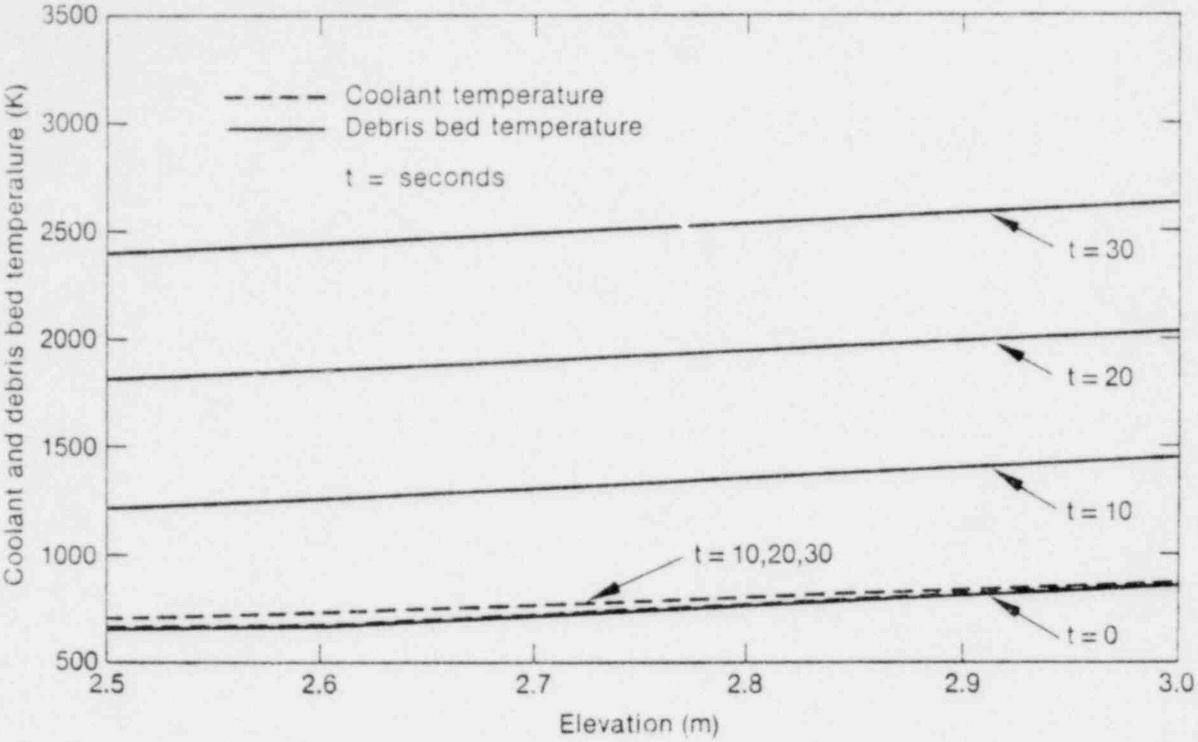


Figure 8. FROZTH calculation of debris bed and coolant temperature distribution.

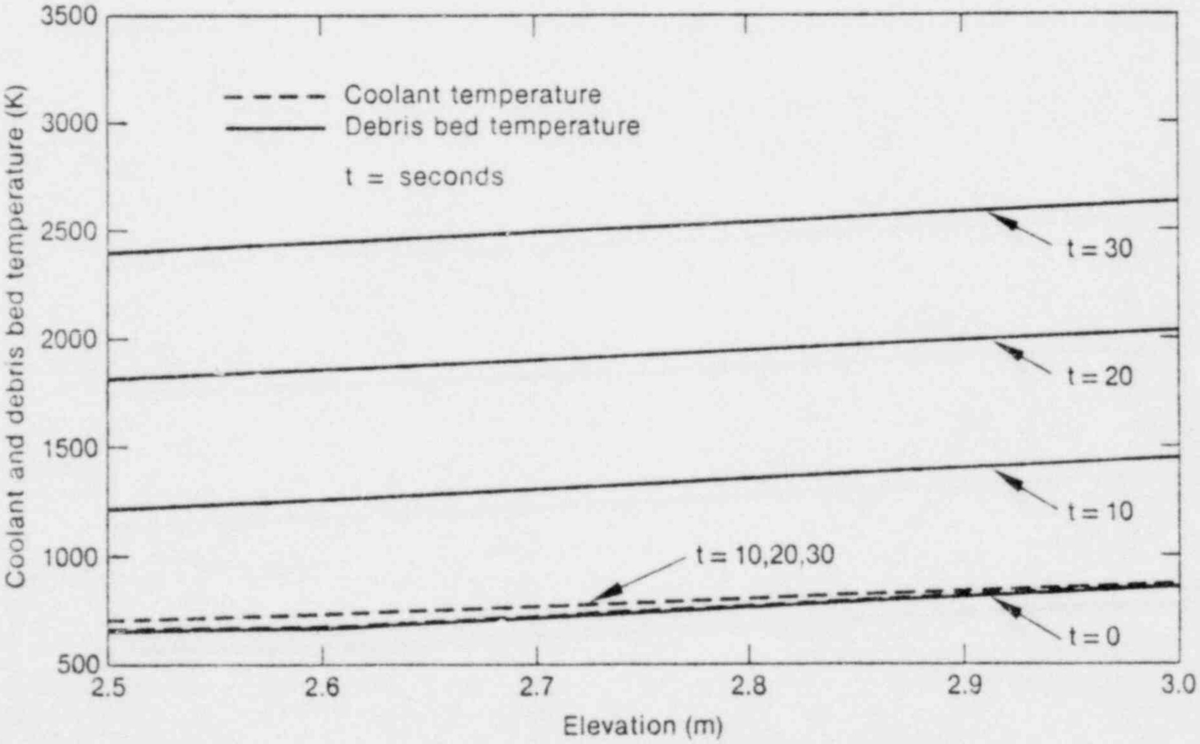


Figure 9. FRAGTH calculation of debris bed and coolant temperature distribution.

5. REFERENCES

1. C. M. Allison et al., *Severe Core Damage Analysis Package (SCDAP) Code Conceptual Design Report*, EGG-CDA-5397, April 1981.

2. L. J. Siefken, *Liquefaction-Flow-Solidification Model (LIQSOL) for SCDAP*, EGG-CDD-5708, December 1981.

3. S. Ergun, "Fluid Flow Through Packed Columns," *Chemical Engineering Progress, 48*, 1952, pp. 89-94.

4. M. Leva and M. Grummer, "Fluid Flow-Fluidization," *Ind-Eng. Chem. 44*, 1952, p. 69.

5. M. Leva, *Fluidization*, New York: McGraw-Hill Book Company, Inc., 1959, p. 87.

6. L. L. Vasiliev and V. A. Maiorov, "An Analytical Study of Resistance, Heat Transfer and Stability in Evaporative Cooling of a Porous Heat-Producing Element," *International Journal of Heat and Mass Transfer, 22*, 1979, pp. 301-307.

7. D. J. Admas et al., "Computations of Dense Random Packings of Hard Spheres," *Journal of Chemical Physics, 56*, 1972.

8. J. D. Bernal et al., "Coordination of Randomly Packed Spheres, "*Nature, Lond. 188*, 1960, pp. 910.

9. S. Debbas and H. Rumph, "On the Randomness of Beds Packed with Spheres or Irregular Shaped Particles," *Chemical Engineering Science, 21*, 1966, pp. 583-607.

10. E. L. Gluekler and L. Baker, Jr., "Post-Accident Heat Removal in LMFBR's," *Symposium on the Thermal and Hydraulic Aspects of Nuclear Reactor Safety, Vol. 2*, CONF-771120, November 1977, pp. 285-324.

11. J. D. Gabor et al., "Studies and Experiments on Heat Removal from Fuel Debris in Sodium," *Proceedings of the Fast Reactor Safety Meeting*, CONF-74041-P2, April 1974, p. 823.

12. V. K. Dhir and I. Catton, *Study of Dryout Heat Fluxes in Beds of Inductively Heated Particles*, NUREG-0262, June 1977.

13. M. Leva et al., "Fluid Flow Through Packed and Fluidized Systems," *U. S. Bureau of Mines Bulletin*, 504, 1951.

14. J. F. Richardson and W. N. Zaki, *Transactions of the Institute of Chemical Engineers, London, Vol. 32*, 1954, p. 35.

15. W. W. Marr and R. M. Crawford, "Porous, Heat-Generating Blockage in a Fuel Subassembly," Transactions of the American Nuclear Society, 15, 1975, p. 350.

16. M. M. El-Wakil, W. U. Choudhury and W. W. Marr, "Heat Transfer and Fluid Flow in Porous Fuel Elements and Thermal Shields," Nuclear Engineering and Design, 17, 1971, pp. 329-340.

17. D. J. Gunn, "Transfer of Heat or Mass to Particles in Fixed and Fluidized Beds," International Journal of Heat and Mass Transfer, 21, 1978, pp. 467-476.

18. F. M. Young and J. P. Holman, "Particle-to-Fluid Nucleate Boiling Heat Transfer in a Water-Fluidized System," I & EC Fundamentals, 7, 1968, pp. 561-567.

19. F. M. Young, Particle-to-Fluid Nucleate Boiling Heat Transfer in a Water Fluidized Medium, Ph.D. Dissertation, Southern Methodist University, 1967.

20. W. J. Choudhury and M. M. El-Wakil, "Heat Transfer and Flow Characteristics in Conductive Porous Media with Energy Generation," Proceedings of the Fourth International Heat Transfer Conference, Versailles, France, August 1970.

21. L. J. Baker et al., "Heat Removal from Molten Fuel Pools," Proceedings of International Meeting on Fast Reactor Safety and Related Physics, Chicago, October 1976.

22. V. K. Dhir and I. Catton, "Dryout Heat Fluxes in Very Deep Debris Beds," Transactions of the American Nuclear Society, 35, 1980, p. 358.

23. H. C. Hardee and R. H. Nilson, "Natural Convection in Porous Media with Heat Generation," Nuclear Science and Engineering, 63, 1977, p. 119.

24. D. Squarer and J. A. Peoples, "Dryout in Inducitively Heated Bed with and without Forced Flow," Transactions of the American Nuclear Society, 34, 1980, p. 535.

25. D. Squarer et al., "Dryout in Large Particle, Deep Debris Bed," Transactions of the American Nuclear Society, 38, 1981, p. 444.

26. J. B. Rivard, Postaccident Heat Removal: Debris Bed Experiments D-2 and D-3, NUREG-CR-0421, November 1978.

27. J. B. Rivard, Debris Bed Studies and Experiments at Sandia Laboratories, SAND78-0299, 1978.

28. R. J. Lipinski, "A Particle-Bed Drout Moded with Upward and Downward Boiling," Transactions of the American Nuclear Society, 35, 1980, p. 358.

29. R. J. Lipinski, "A One-Dimensional Particle Bed Dryout Model," *Transactions of the American Nuclear Society, 38*, 1981, p. 386.

30. R. J. Lipinski, "Debris Bed Modeling," *Advanced Reactor Safety Research Quarterly Report, January through March 1981* (to be published, 1981).

31. C. M. Allison and T-C. S. Hsieh, "Heat Transfer During Severe Core Dirsuptive Accidents in Light Water Reactors," to be published in *Proceedings of Seventh International Heat Transfer Conference*, Munich, September 6-10, 1982.

APPENDIX A
LISTINGS OF DEBRIS BEHAVIOR MODELS

APPENDIX A

LISTINGS OF DEBRIS BEHAVIOR MODELS

FORTRAN listings of the subroutines which are used to calculate debris bed characteristics and thermal behavior are given in Tables A-1 through A-15. Listings of DBDRIV, DEBINP, DBTIME, and DBOUTD are given in Tables A-16 through A-19.

TABLE A-1. LISTING OF DBFROZ

```
C
C
      SUBROUTINE  DBFROZ  ( XMASSC , ZRMC   , ZRO2MC , UO2MC  , STRMC  ,
     +                      ABSMC  , AREAR  , HITEC  , VINC   , VINCOD ,
     +                      ZRO2TC , ZRO2AC , ALPHTC , ALPHAC , ELVNC  ,
     +                      HTLBCD , HTLBCC , HTUBCD , HTUBCC , ND     ,
     +                      MAXL   , DELTT  , POROSC , POROPC ,
     +                      YCHRL  , KTERMC , EFFDIA , IDREGN , NOND   ,
     +                      ELVAY  , BEDTMP , COLTMP , THMOTC , ELVMC  ,
     +                      OXDAC  , FISHC  , HYDGC  , FGRSC  , VOLGSC ,
     +                      IMT    , DOWNZ  , VELTMC , T      ,
     +                      N      , I      , DISRUP , CURTIM , ESTDT  )
C
C
C SUBCODE NAME:  DBFROZ
C PURPOSE:  COHESIVE DEBRIS ANALYSIS DRIVER
C CALLING SUBROUTINES:  DBUNDL
C SUBROUTINES CALLED:  FRTIME , FRCHAR , FROZTH , FRESTT
C WORK PACKAGE:  15
C ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C LAST MODIFICATION DATE:  11/30/81
C
C
C
C
C
C DBFROZ HAS BEEN WRITTEN TO RESPOND TO THE NEEDS OF THE DBUNDL LOGIC.
C THERE ARE NO REGION DIMENSIONS SINCE THE ASSUMPTION IS THAT DBUNDL
C KEEP TRACK OF DIMENSIONING FOR *NUMREG*.  IN THE FINAL FORM,
C VARIABLES DIMENSIONED IN DBFROZ WILL BE IN ADJUSTABLE ARRAYS DEPENDING
C ON THE NUMBER OF RADIAL NODES.
C
C THE TIME *T* COMING INTO DBFROZ FROM DBUNDLE IS ASSUMED TO GO FROM
C T(I) TO T(I) + DELTA T.
C
C
C ------------------------------------------------------------------------
C
C                         RUBBLE DEBRIS ANALYSIS LOGIC [DBFRAG]
C                         -------------------------------------
C
C
C :COMPUTE TIME STEP, T(1)-T(N) [FGTIME]
C
C :I=1
C
C :DO WHILE: I.LT.N
C
C    :COMPUTE DEBRIS BED CHARACTERISTICS INCLUDING HYDRAULIC BEHAVIOR [FGCHAR]
C
C    :COMPUTE DEBRIS BED THERMALLY-RELATED BEHAVIOR  [FRAGTH]
C
C    :IF: DEBRIS BED MELTING OCCURS?
```

TABLE A-1. (CONTINUED)

```
C
C           :THEN: COMPUTE LIQUEFIED MATERIAL MOVEMENT  [FGMELT]
C
C       :ENDIF:
C
C       :IF: REGION BOUNDARY IS DISRUPTED?
C
C       :THEN:
C
C           :SET DISRUPTION =.A;'
C
C           :SET N TO I+1
C
C           :SET T(N) TO CURRENT TIME
C
C       :ELSE:
C
C           :ESTIMATE TIME FOR BOUNDARY DISRUPTION  [FGESTT]
C
C           :IF: ESTIMATED TIME IS .LT. T(I+2), :THEN: T(I+2)=EST. TIME
C
C       :ENDIF:
C
C   :ENDO:
C
C
C
C
C
C
C
C
      DIMENSION       ELVAY  ( 41 ) , BEDTMP ( 41 ) , COLTMP ( 41 ) ,
     +                IMT    ( 41 ) , ROMX   ( 40 ) , CPMX   ( 40 ) ,
     +                XMUX   ( 40 ) , XKMX   ( 40 ) , TC     ( 20 ) ,
     +                T      ( N )
C
C
      COMMON        ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +              XMUCD  , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +              SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +              ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +              XKMX   , XMUF3  , RODEB
C
C
      LOGICAL       DISRUP
C
C
C
C
      NSTPC = 10
C
C
```

```
C CALL FRTIME TO BREAKUP TIME STEP COMING FROM DBUNDL INTO SMALLER
C INTERVALS WHICH WILL DEPEND ON THE VALUE OF NSTP
C
      CALL  FRTIME  ( T (I) , DELTT , TC , NSTPC )
C
      J  = 1
C
   10 CONTINUE
C
C TEST TIME TO PREVENT ENTERING TIME FROM BEING ZERO
C
      IF  ( TC( J ) .EQ. 0.0 )   J = J + 1
C
      IF  ( J .LT. NSTPC )   THEN
C
         CURTIM = TC ( J )
         DELTC  = TC ( J + 1 ) - TC ( J )
C
C
C
      CALL  FRCHAR  ( XMASSC , ZRMC   , ZRO2MC , UO2MC  , STRMC  ,
     +                ABSMC  , AREAR  , HITEC  , VINC   , VINCOD ,
     +                DELTC  , POROSC , POROPC , YCHRL  , KTERMC ,
     +                EFFDIA )
C
C
C CALL FROZTH TO DETERMINE THE BEHAVIOR OF POROUS BED
C
      CALL  FROZTH  ( EFFDIA , POROSC , HITEC  , YCHRL  , VINC   ,
     +                ZRO2TC , ZRO2AC , ALPHTC , ALPHAC , ELVNC  ,
     +                HTLBCD , HTLBCC , HTUBCD , HTUBCC , ND     ,
     +                MAXL   , DELTC  , IDREGN , NOND   , ELVAY  ,
     +                BEDTMP , COLTMP , THMOTC , ELVMC  , OXDAC  ,
     +                FISHC  , HYDGC  , FGRSC  , VOLGSC , IMT    ,
     +                DOWNZ  , DISRUP )
C
C
C COMPUTE LIQUIFIED MATERIAL MOVEMENT IF DEBRIS BED MELTING OCCURS
C
         CALL  FRMELT  ( NOND   , IMT    , ELVMC  , ELVAY  , DELTC  ,
     +                   VELMTC )
C
C
C CHECK FOR LOWER REGION BOUNDARY BREAKTHROUGH
C
C IF BREAKTHROUGH OCCURS, RETURN TO DBUNDL WITH END OF TIME INTERVAL
C EQUAL TO CURRENT TIME.
C
         IF  ( DISRUP )   THEN
C
            NSTPC = J
C
         ELSE
```

TABLE A-1. (CONTINUED)

```
C
            CALL   FRESTT  ( ELVMC   , ELVNC   , VELMTC , CURTIM ,
    +                        ESTDT   )
C
C IF ESTIMATED TIME OF BREAKTHROUGH IS GREATER THAN CURRENT TIME
C PLUS ADJUSTED DELTA TIME, SET CURRENT TIME TO NEXT TIME INTERVAL
C
            IF  ( ESTDT .GT. TC( J + 1 ))   THEN
C
               CURTIM = TC( J + 1 )
C
            ELSE
C
C IF ESTIMATED TIME OF BREAKTHROUGH IS LESS THAN CURRENT TIME PLUS
C ADJUSTED DELTA T, SET CURRENT TIME TO ESTIMATED TIME AND NEXT TIME
C INTERVAL TO CURRENT TIME
C
               CURTIM = ESTDT
               TC( J + 1 ) = CURTIM
C
            ENDIF
C
C
C
         ENDIF
C
C WHEN J IS GREATER THAN NSTPC, CONTROL WILL RETURN TO DBUNDL WITH
C CTIMEC CONTAINING THE CURRENT TIME FOR CURRENT REGION
C
         J = J + 1
         GO  TO  10
C
      ENDIF
C
      RETURN
C
C
      END
```

TABLE A-2. LISTING OF FRTIME

```
C
C
      SUBROUTINE FRTIME  ( TIMIN , DELTT , TC , NSTP )
C
C  SUBCODE NAME: FRTIME
C  PURPOSE:  TO DEFINE THE COHESIVE DEBRIS TIME STEP
C  CALLING SUBROUTINES:  DBFROZ
C  SUBROUTINES CALLES:  NONE
C  WORKPACKAGE:  15
C  ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C  LAST MODIFICATION DATE:  11/30/81
C
C  INPUT VARIABLES                          DESCRIPTION
C
C     DELTIN           TIME STEP CALCULATED IN THE DISRUPTED BUNDLE TIME
C                      STEP SUBROUTINE DBTIME
C     NSTP             NUMBER OF STEPS IN WHICH TIME STEP FROM DBTIME WILL
C                      BE BROKEN
C
C  OUTPUT VARIABLES                         DESCRIPTION
C
C     DELTOT           TIME STEP TO BE USED IN COHESSIVE DEBRIS ANALYSIS WH
C                      IS DERIVED FROM DISRUPTED BUNDLE ANALYSIS TIME STEP
C
C
C
C
C
      DIMENSION       TC ( NSTP + 1 )
C
C
      STEPS = DELTT / FLOAT ( NSTP )
C
      DO  10  I = 1 , NSTP + 1
C
          TC ( I ) = TIMIN + FLOAT ( I + 1 ) * STEPS
C
   10 CONTINUE
C
      RETURN
      END
```

TABLE A-3. LISTING OF FRCHAR

```
C
C
C
      SUBROUTINE FRCHAR(XMASSC,ZRMC,ZRO2MC,UO2MC,STRMC,ABSMC,
     1 AREAR,HITEC,VINC,VINCOD,DELTC,
     2 POROSC,PDROPC,YCHRL,KTERMC,EFFDIA)
C
C       PURPOSE: TO DEFINE COHESIVE BED CHARACTERISTICS AND
C       CALCULATE HYDRAULIC BEHAVIOR
C
C       CALLING SUBROUTINE: DBFROZ
C
C       SUBROUTINE CALLED: NONE
C
C       ENGINEER/PROGRAMMER: STH
C
C       LAST DATE MODIFIED: 8/10/81
C
C       INPUT VARIABLES:
C
C       XMASSC        TOTAL MASS OF DEBRIS ACCUMULATED FOR ALL
C                     COMPONENTS(KG)
C
C       ZRMC          TOTAL MASS OF ZR ACCUMULATED FOR ALL
C                     COMPONENTS(KG)
C
C       ZRO2MC        TATAL MASS OF ZRO2 ACCUMULATED FOR ALL
C                     COMPONENTS(KG)
C
C       UO2MC         TOTAL MASS OF UO2 ACCUMULATED FOR ALL
C                     COMPONENTS(KG)
C
C       STRMC         TOTAL MASS OF CORE STRUCTURAL MATERIALS
C                     ACCUMULATED(KG)
C
C       ABSMC         TOTAL MASS OF CONTROL ROD MATERIAL
C                     ACCUMULATED(KG)
C
C       AREAR         TOTAL BUNDLE CROSS SECTIONAL AREA(M2)
C
C       HITEC         FROZEN DEBRIS REGION HEIGHT FROM DBREGN(M)
C
C       VINC          COOLANT VELOCITY AT INLET OF COHESIVE
C                     DEBRIS REGION(M/SEC)
C
C       VINCOD        COOLANT INLET VELOCITY AT PREVIOUS TIME STEP
C                     (M/SEC)
C
C       DELTC         TIME STEP(SEC)
C
C
C
C
```

```
C
C
C
C     OUTPUT VARIABLES:
C
C     POROSC      BED POROSITY
C
C     PDROPC      PRESSURE DROP ACROSS THE BED(PA)
C
C     YCHRL       CHARACTERISTIC LENGTH OF FROZEN BED, DERIVED
C                 FROM PRESSURE DROP EQUATION, M
C
C     KTERMC      FLAG, =0 NORMAL RUN, =1 ABNORMAL TERMINATION
C
C     EFFDIA      EFFECTIVE PARTICLE DIAMETER DERIVED FROM
C                 SPECIFIC SURFACE AREA(M)
C
C
C
C     AT THIS TIME, ALL MATERIALS PROPERTIES ARE ASSUMED
C     COMING THROUGH COMMON BLOCK, THESE INCLUDE:
C
C     ROZR        ZR DENSITY(KG/M3)
C
C     ROZRO2      ZRO2 DENSITY(KG/M3)
C
C     ROUO2       UO2 DENSITY(KG/M3)
C
C     ROSTR       DENSITY OF STRUCTURAL MATERIAL(KG/M3)
C
C     ROABS       DENSITY OF CONTRAL ROD MATERIAL(KG/M3)
C
C     ROCOL       COOLANT DENSITY(KG/M3)
C
C     XMUCOL      VISCOSITY OF COOLANT(KG/SEC/M)
C
C
C
      COMMON      ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +            XMUCD  , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +            SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +            ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +            XKMX   , XMUF3  , RODEB
C
C
C CALCULATE AVERAGE BED POROSITY
C
      POROSC = 1.0 - ( ZRMC / ROZR + ZRO2MC / ROZRO2 + UO2MC / ROUO2
     +             + STRMC / ROSTR + ABSMC / ROABS) / (AREAR * HITEC)
C
C CHECK POROSITY
```

TABLE A-3. (CONTINUED)

```
C
      IF ( POROSC .GT. 1.00 .OR. POROSC .LT. 0.00 )   THEN
          KTERMC = 1
          WRITE   ( 6 , 100 )
C
      ELSE
C
C  SET SPECIFIC SURFACE AREA OF POROUS BODY=SURFACE EXPOSED
C  TO THE FLUID PER UNIT VOLUME OF SOLID. FROM A. E.
C  SCHEIDEGGER, 3RD EDITION, P16, SPSUF=8.0E5(1/M)
C
          SPSUF = 8.0E+05
          EFFDIA = 6.0 / SPSUF
C
C  CALCULATE PRESSURE DROP ACROSS COHESIVE DEBRIS BED
C
          XITM1 = ( VINC - VINCOD ) * ROCOL / DELTC
          ALPHA = 150. * ( 1.0 - POROSC ) ** 2
     1            / (AMAX1(POROSC,1.0E-8)) ** 3 / (EFFDIA ** 2)
          BETA = 1.75 * ( 1.0 - POROSC )
     1            / (AMAX1(POROSC,1.0E-8)) ** 3 / EFFDIA
          PDROPC = XITM1 + (ALPHA * XMUCOL + BETA * ROCOL * VINC)
     +                     * VINC
          PDROPC = PDROPC * HITEC
          KTERMC = 0
          YCHRL = BETA / ( AMAX1(ALPHA , 1.0E-6) )
C
      ENDIF
C
  100 FORMAT(2X,78HMASS CALDULATION IS WRONG; BED POROSITY IS EITHER NEG
     1ATIVE OR GREATER THAN ONE)
C
      RETURN
      END
```

TABLE A-4. LISTING OF FROZTH

```
      SUBROUTINE  FROZTH  ( EFFDIA , POROSC , HITEC  , YCHRL  , VINC   ,
     +                      ZRO2TC , ZRO2AC , ALPHTC , ALPHAC , ELVNC  ,
     +                      HTLBCD , HTLBCC , HTUBCD , HTUBCC , ND     ,
     +                      MAXL   , DELTT  , IDREGN , NOND   , ELVAY  ,
     +                      BEDTMP , COLTMP , THMOTC , ELVMC  , OXDAC  ,
     +                      FISHC  , HYDGC  , FGRSC  , VOLGSC , IMT    ,
     +                      DOWNZ  , DISRUP )
C
C
C     PURPOSE: TO CALCULATE THERMALLY RELATED BEHAVIORS OF A POROUS
C     BED.
C
C     CALLING SUBROUTINE: DBFROZ
C
C     SUBROUTINE CALLED: TEMPSR
C
C     ENGINEER/PROGRAMMER:STH
C
C     LAST DATE MODIFIED: 8/10/81
C
C     INPUT VARIABLES:
C
C     EFFDIA    EFFECTIVE PARTICLE DIAMETER OF A POROUS BODY(M)
C
C     POROSC    DEBRIS BED POROSITY
C
C     HITEC     BED HEIGHT(M)
C
C     YCHRL     CHARACTERISTIC LENGTHOF POROUS BODY FROM DELT P, M
C
C     VINC      COOLANT INLET VELOCITY(M/SEC)
C
C     ZRO2TC    EFFECTIVE ZRO2 REACTION LAYER THICKNESS(M)
C
C     ZRO2AC    EFFECTIVE ZR-STEAM REACTION AREA(M2)
C
C     ALPHTC    EFFECTIVE ALPHA-ZR REACTION LAYER THICKNESS,M
C
C     ALPHAC    EFFECTIVE ALPHA-ZR REACTION AREA(M2)
C
C     ELVNC     ELEVATION OF POROUS BODY-FROM THE BOTTOM OF ROD BUNDLE
C               TO THE BOTTOM OF POROUS BODY REGION(M)
C
C     HTLBCD    HEAT TRANSFER INTO DEBRIS BED AT LOWER BOUNDARY,W/M2
C
C     HTLBCC    HEAT TRANSFER INTO COOLANT AT LOWER BOUNDARY,W/M2
C
C     HTUBCD    HEAT TRANSFER INTO DEBRIS BED AT UPPER BOUNDARY,W/M2
C
C     HTUBCC    HEAT TRANSFER INTO COOLANT AT UPPER BOUNDARY,W/M2
C
```

TABLE A-4. (CONTINUED)

```
C       ND          NUMBER OF NODES TO BE USED IN LIQUID OR VAPOR REGION
C
C       MAXL        MAXIMUM NUMBER OF NODES IN THE POROUS BODY WHICH
C                   COMBINED LIQUID, SATURATION AND VAPOR REGION,
C
C       DELTT       TIME STEP(S)
C
C
C
C
C
C
C       INPUT/OUTPUT VARIABLES:
C
C       IDREGN      REGION TYPE I.D. =1 VINC GT 0 ONE LIQUID REGION
C                                    =2 VINC GT 0 ONE VAPOR GEGION
C                                    =3 VINC GT 0 OND SAT. REGION
C                                    =4 VINC GT 0 TWO REG,L & S
C                                    =5 VINC GT 0 TWO REG,S & V
C                                    =6 VINC GT 0 THREE REG L&S&V
C                                    =7 VINC=0 ONE LIQUID REG
C                                    =8 VINC=0 ONE VAPOR REG
C
C       NOND        NUMBER OF NEDES USED IN POROUS BODY ANALYSIS,
C                   =ND   IDREGN=1,2,7,8
C                   =3    IDREGN=3
C                   =ND+2 IDREGN=4,5
C                   =2*ND+1 IDREGN=6
C
C       ELVAY       ELEVATION OF NODES IN DEBRIS, ELVAY(1)=ELVNC
C                   ELVAY(NOND)=ELVNC+HITEC
C
C       BEDTMP      DEBRIS TEMPERATURE CORRESPONDING TO NOND NODES
C
C       COLTMP      COOLANT TEMPERATURE CORRESPONDING TO NOND NODES
C
C       OXDAC       ZR-STEAM HEAT GENERATION RATE(W/M3)
C
C       FISHC       FISSION/DECAY HEAT GENERATION RATE(W/M3)
C
C       OUTPUT VARIABLES:
C
C       THMOTC      MOLTEN MATERIAL THICKNESS(M)
C
C       ELVMC       MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY,M
C
C       HYDGC       HYDROGEN GENERATION DURING DELTT(MOLES)
C
C       FGRSC       FISSION GAS RELEASE(MOLES) DURING DELTT
C
C       VOLGSC      VOLATILE FISSION PRODUCTS RELEASE DURING DELTT, MOLE
C
C       IMT         DEBRIS REGION TYPE, =0 SOLID REGION, =1 MOLTEN REGION
C                   TOTAL OF NOND VALUES
```

TABLE A-4. (CONTINUED)

```
C
C     DOWNZ     LENGTH OF MOLTEN POOL PENETRATION, M
C
C               AFTER CURRENT TIME, S
C     DISRUP    DISRUPTION FLAG
C
C
C
C
C     AT THIS TIME, ALL MATERIALS PROPERTIES ARE ASSUMED COMING
C     THROUGH COMMON BLOCK, THESE INCLUDE:
C     ROCOL     LIQUID COOLANT DENSITY(KG/M3)
C     ROVAP     STEAM DENSITY(KG/M3)
C     XMUCOL    LIQUID COOLANT VISCOSITY(KG/SEC/M)
C     XMUVAP    VAPOR VISCOSITY(KG/SEC/M)
C     HFG       LATENT HEAT OF VAPORIZATION(J/KG) OF COOLANT
C     SURTC     SURFACE TENSION OF COOLANT(PA)
C     CPCOL     LIQUID SPECIFIC HEAT(J/K/KG)
C     XKCOL     LIQUID COOLANT THERMAL CONDUCTIVITY)W/M/K)
C     XKVAP     VAPOR(STEAM) THERMAL CONDUCTIVITY(W/M/K)
C     CPDEB     SPECIFIC HEAT OF DEBRIS(J/K/KG)
C     ROZR      ZR DENSITY(KG/M3)
C     RCZRO2    ZRO2 DENSITY(KG/M3)
C     RCUC2     LO2 DENSITY(KG/M3)
C     ROSTR     DENSITY OF STRUCTURAL MATERIAL(KG/M3)
C     ROABS     DENSITY OF CONTROL ROD MATERIAL(KG/M3)
C     XMUDEB    VISCOSITY OF DEBRIS(KG/SEC/M)
C     XKDEB     THERMAL CONDUCTIVITY OF DEBRIS(W/M/K)
C     TSAT      SATURATION TEMPERATURE OF COOLANT OF FROZEN DEBRIS
C               ENVIRONMENT,K
C
```

TABLE A-4. (CONTINUED)

```
C       ROSAT       WATER DENSITY AT SATURATION(KG/M3)
C
C       TMELT       MELTING TEMPERATURE OF DEBRIS MATERIAL(K)
C
C       XLATC       LATENT HEAT OF FUSION FOR DEBRIS MATERIAL(J/KG)
C
C
C
C
        DIMENSION   ELVAY(41)  , BEDTMP(41) , COLTMP(41) , IMT(41)    ,
     +              BEDTMO(41) , COLTMO(41) , ROMX(40)   , CPMX(40)   ,
     +              XMUX(40)   , XKMX(40)   , TEMPBO(20) , TEMPB(20)  ,
     +              TCOLTO(20) , TCOLT(20)
C
C
        COMMON      ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +              XMUCOD , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +              SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +              ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +              XKMX   , XMUF3  , RODEB
C
C
        LOGICAL     DISRUP
C
C
        DATA    PI / 3.14159 /
C
C
C   CALCULATE VOLUMETRIC HEAT GENERATION RATE QVOL(W/M3)
C
        QVOL = OXDAC + FISHC
        IITMP = 0
C
        ELVOO = ELVMC
C
C
C   BASED ON QVOL,VINC, AND COLTMP CHECK THE REGION ID
C
 2020   CONTINUE
C
C
        XLLIQ = - 100.0
        XLSAT = - 100.0
        XLVAP = - 100.0
C
C   CALCULATE DRYOUT HEAT BASED ON LIPINSKI'S 1-D MODEL
C
        CALL DRYOUT(VINC,POROSC,HITEC,EFFDIA,PDROPC,QVOL,ROCOL,ROVAP,
     +XMUCOL,XMUVAP,HFG,SURTC,QDYOUT)
C
C
        IF  ( VINC .LE.  0.0 )   THEN
C
```

TABLE A-4. (CONTINUED)

```
C COMPARE QVOL TO QDYOUT
C
          IF  ( QVOL * HITEC * (1.0 - POROSC) .GE. QDYOUT )  THEN
              IDR = 8
C
          ELSE
              IDR = 7
C
C
          ENDIF
C
      ELSE
C
C REGION TYPE 1-6 LEFT TO BE IDENTIFIED
C
          IF  ( COLTMP(1) .GT. TSAT )    THEN
              IDR = 2
C
          ELSE
              IF  ( COLTMP(1) .EQ. TSAT )    THEN
                  IF  ( COLTMP(NOND) .EQ. TSAT )    THEN
                      IDR = 3
C
                  ELSE
C
C CALCULATE SATURATION REGION LENGTH
C
                      XLSAT = ROCOL * VINC * HFG / QVOL /
     +                            AMAX1 (( 1.0 - POROSC ) , 1.E-8 )
                      IDR = 5
                      XLVAP = HITEC - XLSAT
C
                      IF  ( QVOL * HITEC * ( 1.0 - POROSC )
     +                                  .LT.  QDYOUT )   IDR = 3
C
                      IF  ( XLSAT .GE. HITEC )    IDR = 3
                      IF  ( COLTMP(NOND) .LT. TSAT )   IDR = 3
C
                  ENDIF
C
              ELSE
C
C COLTMP(1)  .LT.  TSAT
C
                  IF  ( COLTMP(NOND) .LT. TSAT )    THEN
                      IDR = 1
C
                  ELSE
                      XLSAT = ROCOL * VINC * HFG / QVOL /
     +                        AMAX1 (( 1.0 - POROSC ) , 1.E-8 )
                      XLLIQ = XLSAT/HFG * CPCOL * (TSAT - COLTMP(1))
C
                      IF  ( COLTMP(NOND) .EQ. TSAT )   THEN
```

```
                          IDR = 4
                          XLSAT = HITEC - XLLIQ
C
                          IF  ( XLSAT .LE. 0 )    IDR = 1
C
                      ELSE
                          IDR = 6
                          XLVAP = HITEC - XLLIQ - XLSAT
                          IF  (QVOL * (HITEC - XLLIQ) * (1.0 - POROSC)
     +                                .LT.  QDYOUT )   THEN
                              IDR = 4
                              XLSAT = HITEC - XLLIQ
C
                          ELSE
                              IF ( (XLLIQ + XLSAT) .GE. HITEC )  THEN
                                  IDR = 4
                                  XLSAT = HITEC - XLLIQ
                                  IF  ( XLLIQ .GE. HITEC )   IDR = 1
C
                              ENDIF
C
                          ENDIF
C
                      ENDIF
C
                  ENDIF
C
              ENDIF
C
          ENDIF
C
      ENDIF
C
C DEBRIS REGION TYPE HAS BEEN IDENTIFIED,IDR=1-8,IF IDR AND
C IDREGN ARE NCT CONSISTANT, CALL SUBROUTINE REGMOD TO ADJUST
C TEMPERATURE OF BED AND COOLANT,NODES ELEVATION AND SET IDREGN
C EQUAL TC IDR
C
C
      CALL REGMOD(IDR,MAXL,ND,IDREGN,NOND,XLLIQ,XLSAT,XLVAP,
     +ELVNC,TSAT,HITEC,ELVAY,BEDTMP,COLTMP)
C
      N2 = ND * 2
C
      DO  50  I = 1 , NOND
C
          BEDTMO(I) = BEDTMP(I)
          COLTMC(I) = COLTMP(I)
C
   50 CONTINUE
C
C BASED ON COLTMP AND BEDTMP, ROMX(N2),CPMX(N2), XMUX(N2) AND
C XKMX(N2) ARE KNOWN ASSUME THEY ARE TEMPERATURE DEPENDENT,
```

TABLE A-4. (CONTINUED)

```
C FIRST N VALUES ARE FOR BED AND THE REST N VALUES ARE FOR
C COOLANT.
C
C NOW CALCULATE TEMPERATUR OF BED AND COOLANT FOR CURRENT TIME
C
C
      IF ( IDREGN .EQ. 1 .OR. IDREGN .EQ. 2 )   THEN
C
          IF  ( IDREGN .EQ. 1 )  IDYOT  = 0
           IF  ( IDREGN .EQ. 2 )  IDYOT  = 1
          VAR1 = 0.0
          VAR2 = 0.0
          VAR3 = 0.0
          VAR4 = 0.0
C
C         CALL MATPRO ( BEDTMO , COLTMO , ND , N2 , ROMX , CPMX ,
C                       XKMX , XMUX , IDREGN , IDYOT )
C
          DO  100  I = ND + 1 , N2
C
              VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
              VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
              VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
              VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
  100     CONTINUE
C
C
          XRENO = VAR1 * VINC * YCHRL / VAR4
          XPRD L =VAR2 * VAR4 / VAR3
C
          XNU = ( 7. - ( 10. - 5. * POROSC ) * PORCSC )  *
     +          ( 1. + .7 * XRENO ** .2 * XPRDL ** ( 1. / 3. )) +
     +          ( 1.33 - ( 2.4 - 1.2 * POROSC ) * POROSC ) * XRENO
     +            ** .7 * XPRDL ** ( 1. / 3. )
C
          HVSC = XNU * VAR3 / YCHRL ** 2
C
C
          DELTZ = HITEC / FLOAT( ND-1 )
C
       IF  ( IDYOT .EQ. 1 )   THEN
          HVSC = (YCHRL * ( 1.0 - POROSC ) / .00377) ** 1.33
     +              * XRENO ** 0.65 * VAR3 / YCHRL ** 2
C
       ENDIF
C
          CALL  TEMPSR  ( ND     , N2     , IDYOT  , BEDTMO , COLTMO ,
     +                    HVSC   , HTLBCD , HTLBCC , HTUBCD , HTUBCC ,
     +                    POROSC , VINC   , QVOL   , DELTT  , DELTZ  ,
     +                    BEDTMP , COLTMP )
C
          IF  ( IDREGN .EQ. 1 )   THEN
```

```
              DO  150  I = 1 , NOND
                  IF  ( COLTMP(I) .GE. TSAT)  COLTMP(I) = TSAT
  150         CONTINUE
C
          ELSE IF ( IDREGN .EQ. 2 )   THEN
              DO  160  I = 1 , NOND
                  IF  ( COLTMP(I) .GE. TSAT)  COLTMP(I) = TSAT
  160         CONTINUE
C
          ENDIF
C
C
      ELSE  IF  (IDREGN.EQ.7.OR.IDREGN.EQ.8)   THEN
          XNU = 7.0 - ( 10.0 - 5.0 * POROSC ) * POROSC
C
          IF ( IDREGN .EQ. 7 )   IDYOT = 0
          IF ( IDREGN .EQ. 8 )   IDYOT = 1
C
          VAR1 = 0.0
C
C          CALL MATPRO  ( BEDTMO , COLTMO , ... , ...
C
          DO  200  I = ND + 1 , N2
              VAR1 = VAR1 + XKMX(I) / FLOAT(ND)
  200     CONTINUE
C
          HVSC = XNU * VAR1 / YCHRL ** 2
          DELTZ = HITEC / FLOAT( ND - 1 )
C
          IF  ( IDYOT .EQ. 1 )   THEN
              HVSC = (YCHRL * ( 1.0 - POROSC) / .00377) ** 1.33
     +                * XRENO ** .65 * VAR3 / YCHRL ** 2
C
          ENDIF
C
          CALL TEMPSR  ( ND     , N2     , IDYOT  , BEDTMO , COLTMO ,
     +                   HVSC   , HTLBCD , HTLBCC , HTUBCD , HTUBCC ,
     +                   POROSC , VINC   , QVOL   , DELTT  , DELTZ  ,
     +                   BEDTMP , COLTMP )
C
C
          IF  ( IDREGN .EQ. 7 )   THEN
              DO  201  I = 1 , NOND
                  IF  ( COLTMP(I) .GE. TSAT )  COLTMP(I) = TSAT
  201         CONTINUE
C
          ENDIF
C
C
C IF IDREGN EQUALS TO 3, QVOL IS USED TO HEAT SATURATED WATER
C NO BED AND COOLANT TEMPERATURE CHANGES ASSUMED, I.E. NO
C CALCULATION OF TEMPERATURE IS NEEDED
C
```

```
    ELSE  IF  (IDREGN.EQ.4)   THEN
C
       IDYOT = 0
       DELTZ = XLLIQ / FLOAT( ND - 1 )
C
       VAR1 = 0.0
       VAR2 = 0.0
       VAR3 = 0.0
       VAR4 = 0.0
C
C
C      CALL  MATPRO ( BEDTMO , COLTMO , ... , ...
C
       DO  300  I = ND + 1 , N2
C
          VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
          VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
          VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
          VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
 300   CONTINUE
C
       XRENO = VAR1 * VINC * YCHRL / VAR4
       XPRDL = VAR2 * VAR4 / VAR3
C
       XNU = (7.0-(10.0-5.0*POROSC)*POROSC)*(1.0+0.7*XRENO**0.2
    +        *XPRDL**(1.0/3.0))+(1.33-(2.4-1.2*POROSC)*POROSC)*
    +        XRENO**0.7*XPRDL**(1.0/3.0)
C
       HVSC = XNU * VAR3 / YCHRL ** 2
C
       DO  400  I = 1 , ND
          TEMPBO(I) = BEDTMO(I)
          TCOLTO(I) = COLTMO(I)
 400   CONTINUE
C
C
       CALL TEMPSR  ( ND     , N2     , IDYOT  , TEMPBO , TCOLTO ,
    +                 HVSC   , HTLBCD , HTLBCC , HTUBCD , HTUBCC ,
    +                 POROSC , VINC   , QVOL   , DELTT  , DELTZ  ,
    +                 TEMPB  , TCOLT  )
C
C
       DO  500  I = 1 , ND
          BEDTMP(I) = TEMPB(I)
          COLTMP(I) = TCOLT(I)
          IF  ( COLTMP(I) .GE. TSAT )  COLTMP(I) = TSAT
 500   CONTINUE
C
       BEDTMP(ND+1) = BEDTMO(ND+1)
       BEDTMP(ND+2) = BEDTMO(ND+2)
       COLTMP(ND+1) = COLTMP(ND)
       COLTMP(ND+2) = COLTMP(ND)
C
```

TABLE A-4. (CONTINUED)

```
      ELSE  IF  ( IDREGN .EQ. 5 )    THEN
          IDYOT = 1
          DELTZ = XLVAP / FLOAT( ND - 1 )
C
          VAR1 = 0.0
          VAR2 = 0.0
          VAR3 = 0.0
          VAR4 = 0.0
C
          DO  600  I = ND + 1 , N2
C
              VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
              VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
              VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
              VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
  600     CONTINUE
C
          XRENO = VAR1 * VINC * YCHRL / VAR4
          XPRDL = VAR2 * VAR4 / VAR3
C
          XNU = (7.0-(10.0-5.0*POROSC)*POROSC)*(1.0+0.7*XRENO**0.2
     +          *XPRDL**(1.0/3.0))+(1.33-(2.4-1.2*POROSC)*POROSC)*
     +          XRENO**0.7*XPRDL**(1.0/3.0)
C
          HVSC = XNU * VAR3 / YCHRL ** 2
C
          DO  700  I = 1 , ND
              TEMPBO(I) = BEDTMO(I+2)
              TCOLTO(I) = COLTMO(I+2)
  700     CONTINUE
C
          IF  ( IDYOT .EQ. 1 )   THEN
              HVSC = (YCHRL * ( 1.0 - POROSC) / .00377 ) ** 1.33
     +                    * XRENO ** .65 * VAR3 / YCHRL ** 2
          ENDIF
C
          CALL  TEMPSR  ( ND     , N2     , IDYOT  , TEMPBO , TCOLTO ,
     +                    HVSC   , HTLBCD , HTLBCC , HTUBCD , HTUBCC ,
     +                    POROSC , VINC   , QVOL   , DELTT  , DELTZ  ,
     +                    TEMPB  , TCOLT  )
C
C
          DO  800   I = 1 , ND
              BEDTMP(I+2) = TEMPB(I)
              COLTMP(I+2) = TCOLT(I)
              IF  ( COLTMP(I+2) .LE. TSAT )  COLTMP(I+2) = TSAT
  800     CONTINUE
C
          COLTMP(1) = COLTMP(3)
          COLTMP(2) = COLTMP(3)
          BEDTMP(1) = BEDTMO(1)
          BEDTMP(2) = BEDTMO(2)
```

```
C
      ELSE  IF   ( IDREGN .EQ. 6 )   THEN
C
          IDYOT = 0
          DELTZ = XLLIQ / FLOAT( ND - 1 )
C
 1200     VAR1 = 0.0
          VAR2 = 0.0
          VAR3 = 0.0
          VAR4 = 0.0
C
C         CALL MATPRO  ( BEDTMO , COLTMO , ... , ...
C
C
          DO  900  I = ND + 1 , N2
C
              VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
              VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
              VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
              VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
  900     CONTINUE
C
          XRENO = VAR1 * VINC * YCHRL / VAR4
          XPRDL = VAR2 * VAR4 / VAR3
C
          XNU = (7.0-(10.0-5.0*POROSC)*POROSC)*(1.0+0.7*XRENO**0.2
     +          *XPRDL**(1.0/3.0))+(1.33-(2.4-1.2*POROSC)*POROSC)*
     +          XRENO ** 0.7 * XPRDL ** (1.0/3.0)
C
          HVSC = XNU * VAR3 / YCHRL ** 2
C
          IF ( IDYOT .EQ. 1 )   GO  TO  1300
C
          DO  1000  I = 1 , ND
              TEMPBO(I) = BEDTMO(I)
              TCOLTO(I) = COLTMO(I)
 1000     CONTINUE
C
          CALL  TEMPSR ( ND     , N2     , IDYOT  , TEMPBO , TCOLTO ,
     +                   HVSC   , HTLBCD , HTLBCC , HTUBCD , HTUBCC ,
     +                   POROSC , VINC   , QVOL   , DELTT  , DELTZ  ,
     +                   TEMPB  , TCOLT  )
C
C
          DO  1100  I = 1 , ND
              BEDTMP(I) = TEMPB(I)
              COLTMP(I) = TCOLT(I)
              IF  ( COLTMP(I) .GE. TSAT )  COLTMP(I) = TSAT
 1100     CONTINUE
C
          BEDTMP(ND+1) = BEDTMO(ND+1)
          COLTMP(ND+1) = COLTMP(ND)
```

TABLE A-4. (CONTINUED)

```
             IDYOT = 1
             DELTZ = XLVAP / FLOAT( ND - 1 )
C
             IF ( IDYOT .EQ. 1 )    GO  TO  1200
C
C        COOLANT PROPERTIES ARE VAPOR PORPERTIES NOW
C
             VAR1  = 0.0
             VAR3  = 0.0
C            CALL MATPRO ( BEDTMO , COLTMO , ... , ...
             DO 1310 I = ND+1 , N2
                 VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
                 VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
 1310        CONTINUE
C
C
 1300        CONTINUE
C
             IF ( IDYOT .EQ. 0 )    GO  TO  1600
C
             DO  1400  I = 1 , ND
                 TEMPBO(I) = BEDTMO( I+ND+1 )
                 TCOLTO(I) = COLTMO( I+ND+1 )
 1400        CONTINUE
C
             IF  ( IDYOT .EQ. 1 )    THEN
                 HVSC = (YCHRL * ( 1.0 - POROSC ) / .00377 ) ** 1.33
     +                  * XRENO ** .65 * VAR3 / YCHRL ** 2
             ENDIF
C
             CALL  TEMPSR  ( ND     , N2     , IDYOT  , TEMPBO , TCOLTO ,
     +                       HVSC   ; HTLBCD ; HTLBCC ; HTUBCD ; HTUBCC ;
     +                       POROSC , VINC   , QVOL   , DELTT  , DELTZ  ,
     +                       TEMPB  , TCOLT  )
C
C
             DO  1500  I = 1 , ND
                 BEDTMP( I+ND+1 ) = TEMPB(I)
                 COLTMP( I+ND+1 ) = TCOLT(I)
                 IF ( COLTMP(I + ND + 1) .LE. TSAT )  COLTMP(I+ND+1)=TSAT
 1500        CONTINUE
C
 1600        CONTINUE
C
         ENDIF
C
C        CHECK DEBRIS BED TEMPERATURE AND DEFINE MELTING REGION
C
         IMG=0
         ISG=0
         DOWNZ=0.0
C
         DO 2000 I=1,NOND-1
```

TABLE A-4. (CONTINUED)

```
C
          IF( (BEDTMP(I)+BEDTMP(I+1))/2.0 .GT. TMELT )   THEN
C
C     TMELT IS DEBRIS BED MELTING TEMPERATURE FROM MAPRO
C
              IMG = IMG + 1
              JUP = I + 1
              IMT(I) = 1
C
          ELSE
              ISG = ISG + 1
              IMT(I) = 0
C
          ENDIF
C
 2000 CONTINUE
C
C
      DO  2010  I = 1 , NOND - 1
C
          J = NOND - I
          DTMP1 = ( BEDTMP(J+1)+BEDTMP(J) ) / 2.0 - TMELT
C
          IF ( DTMP1 .LT. 0.0 )   THEN
              DTMP1 = 0.0
C
          ELSE
              DOWNZ=DOWNZ+DTMP1*(ELVAY(J+1)-ELVAY(J))*CPDEB/XLATC
              BEDTMP(J+1) = TMELT
              BEDTMP(J) = TMELT
C
          ENDIF
C
          IF  ( J .EQ. 1 )   GO  TO  2010
C
          IF ( IMT(J) .GT. IMT( J-1 ) )   THEN
C
C     MOLTEN MATERIALS MELTING THE LOWER NODE
C
              IF (DOWNZ .GT. (0.5*(ELVAY(J)-ELVAY(J-1))) )   THEN
                  BEDTMP(J) = TMELT
                  BEDTMP(J-1) = TMELT
                  IMT(J-1) = 1
                  DOWNZ = DOWNZ - ( ELVAY(J) - ELVAY(J-1) )
C
              ENDIF
C
          ENDIF
C
C
 2010 CONTINUE
      THMOTC = 0.0
C
```

TABLE A-4. (CONTINUED)

```
      IF ( IMT (1) .EQ. 1 )    THEN
          THMOTC = THMOTC + 0.5 * ( ELVAY (2) - ELVAY (1))
C
C  SET DISRUPTION FLAG WHEN DEBRIS REGION TYPE = 1, INDICATING MELTING
C
          DISRUP = .TRUE.
C
      ENDIF
C
      IF  ( IMT (NOND) .EQ. 1 )     THEN
          THMOTC = THMOTC + 0.5 * ( ELVAY (NOND) - ELVAY (NOND - 1))
C
      ENDIF
C
      DO   3020  I = 2 , NOND - 1
C
          IF   ( IMT (I) .EQ. 1 )    THEN
              THMOTC = THMOTC + 0.5 * ( ELVAY (I + 1) - ELVAY (I - 1))
C
          ENDIF
C
 3020 CONTINUE
C
C
C
C     NOW ADJUSTED BED TEMPERATURE AND MELTEN REGION HAS BEEN DEFINED
C
C     CALL CHEMHT(ZRO2TC,ZRO2AC,ALPHTC,ALPHAC,IDREGN,NOND,BEDTMP,
C    +     COLTMP,ELVAY,DELTT,OXDAC,HYDGC)
C     CALL NUCLHT(IDREGN,NOND,BEDTMP,ELVAY,DELTT,TOTIME,...ETC)
C
      QVOLN = OXDAC + FISHC
      IITMP = IITMP + 1
C
      IF ( IITMP .GT. 50 )    THEN
          WRITE     ( 6 , 2050 )
 2050 FORMAT(2X, 57HITERATION MORE THAN 50 TIMES NO CONVERGENCE IN TEMP.
     + CAL.)
          GO TO 2030
C
      ENDIF
C
      IF ( ABS((QVOLN-QVOL)/QVOLN) .GT. 0.05 )     THEN
          QVOL = QVOLN
          GO  TO  2020
C
      ENDIF
C
 2030 RETURN
      END
```

TABLE A-5. LISTING OF FRMELT

```
C
C
      SUBROUTINE  FRMELT  ( NOND   , IMT    , ELVMC  , ELVAY  , DELTC ,
     +                      VELMTC )
C
C
C SUBCODE NAME:  FRMELT
C PURGPOSE:  TO MEASURE THE LIQUID MATERIAL MOVEMENT WHEN DEBRIS BED
C             MELTING OCCURS.
C CALLING SUBROUTINES:  DBFROZ
C SUBROUTINES CALLED:  NONE
C WORK PACKAGE:  15
C ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C LAST MODIFICATION DATE:  11/30/81
C
C
C INPUT VARIABLES                       DESCRIPTION
C     NOND              NUMBER OF NODES USED IN POROUS BODY ANALYSIS
C     IMT               DEBRIS REGION TYPE: = 0, SOLID REGION
C                                           = 1, MOLTEN REGION
C     ELVMC             MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY
C     ELVAY             ELEVATION OF NODES IN DEBRIS
C     DELTC             TIME STEPS (S)
C
C OUTPUT VARIABLES                      DESCRIPTION
C     VELMTC            VELOCITY OF MOLTON POOL FLOWING DOWNWARD (M/S)
C
C
C
      DIMENSION     ELVAY( NOND ) , IMT( NOND )
C
C
C
C SAVE INCOMING ELEVATION OF MOLTON MATERIAL
C
      ELVMO = ELVMC
C
      DO  100  I = 1 , NOND
          IF  ( IMT(I) .EQ. 1 )   GO  TO  110
  100 CONTINUE
C
  110 IF  ( I .LE. NOND )   ELVMC = ELVAY(I)
C
      VELMTC = AMAX1( 0.0 , (ELVMO - ELVMC)) / DELTC
C
      RETURN
      END
```

TABLE A-6. LISTING OF FRESTT

```
C
C
      SUBROUTINE  FRESTT  ( ELVMC  , ELVNC  , VELMTC , CURTIM , ESTDT )
C
C
C  SUBCODE NAME:  FGESTT
C  PURPOSE:  TO ESTIMATE THE TIME OF DISRUPTION
C  CALLING SUBROUTINES:  DBFRAG
C  SUBROUTINES CALLED:  NONE
C  WORK PACKAGE:  15
C  ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C  LAST MODIFICATION DATE:  11/30/81
C
C
C  INPUT VARIABLES                   DESCRIPTION
C      ELVMC            MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY
C      ELVNC            ELEVATION OF RUBBLE BODY FROM THE BOTTOM OF ROD
C                       BUNDLE TO BOTTOM OF RUBBLE BODY REGION  (M)
C      VELMTC           VELOCITY OF MOLTEN POOL FLOWING DOWNWARD (M/S)
C      CURTIM           CURRENT TIME  (S)
C
C  OUTPUT VARIABLES                  DESCRIPTION
C      ESTDT            ESTIMATED TIME OF DISRUPTION  (S)
C
C
C
      TESTCC = ABS( ELVMC - ELVNC ) / AMAX1(1.0E-08 , VELMTC )
      ESTDT = CURTIM + TESTCC
C
      RETURN
      END
```

TABLE A-7. LISTING OF DBFRAG

```
C
C
      SUBROUTINE  DBFRAG  ( SIZAVR , XMASSR , ZRMR   , ZRO2MR , UO2MR  ,
     +                      STRMR  , ABSMR  , AREAR  , HITRG  , VINROD ,
     +                      VINR   , ZRO2TR , ZRO2AR , ALPHTR , ALPHAR ,
     +                      ELVNR  , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                      ND     , MAXL   , DELTT  , IDREGR , NONR   ,
     +                      IRT    , ELVAR  , COLTMR , BEDTMR , THMOTR ,
     +                      ELVMR  , OXDAR  , FISHR  , HITER  , POROSR ,
     +                      POROPR , XCHRL  , KFLUID , KTERM  , DOWNR  ,
     +                      VELMOT , T      , N      , I      ,
     +                      DISRUP , CURTIM , ESTDT  )
C
C
C
C  SUBCODE NAME:  DEFRAG
C  PURPOSE:  RUBBLE DEBRIS ANALYSIS DRIVER
C  CALLING SUBROUTINES:  DBUNDL
C  SUBROUTINES CALLED:  FGCHAR,FRAGTH
C  WORK PACKAGE:  15
C  ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C  LAST MODIFICATION DATE:  11/30/81
C
C
C
C
C
C
C
C
C  DBFRAG HAS BEEN WRITTEN TO RESPOND TO THE NEEDS OF THE DBUNDL LOGIC.
C  THERE ARE NO REGION DIMENSIONS SINCE THE ASSUMPTION IS THAT DBUNDL
C  KEEP TRACK OF DIMENSIONING FOR *NUMREG*.  IN THE FINAL FORM,
C  VARIABLES DIMENSIONED IN DBFRAG WILL BE IN ADJUSTABLE ARRAYS DEPENDING
C  ON THE NUMBER OF RADIAL NODES.
C
C  THE TIME *T* COMING INTO DBFRAG FROM DBUNDLE IS ASSUMED TO GO FROM
C  T(I) TO T(I) + DELTA T.
C
C
C
C
C  ----------------------------------------------------------------------
C
C
C                            COHESIVE DEBRIS ANALYSIS LOGIC [DBFROZ]
C                               ---------------------------------------
C
C
C
C  :COMPUTE TIME STEP, T(1)-T(N) [FRTIME]
C
C
C  :I=1
C
C
C  :DO WHILE: I.LT.N
C
C
C     :COMPUTE DEBRIS BED CHARACTERISTICS [FRCHAR]
C
C
C     :COMPUTE DEBRIS BED HYDRAULIC BEHAVIOR  [FRHYDR]
C
C
C     :COMPUTE DEBRIS BED THERMALLY-RELATED BEHAVIOR  [FRTHRM]
```

```
C
C     :IF: DEBRIS BED MELTING OCCURS?
C
C         :THEN: COMPUTE LIQUEFIED MATERIAL MOVEMENT
C
C     :ENDIF:
C
C     :IF: REGION BOUNDARY IS DISRUPTED?
C
C     :THEN:
C
C         :SET DISRUPTION FLAG
C
C         :SET N TO I+1
C
C         :SET T(N) TO CURRENT TIME
C
C      :ELSE:
C
C         :ESTIMATE TIME FOR BOUNDARY DISRUPTION  [FRESTT]
C
C         :IF: ESTIMATED TIME IS .LT. T(I+2), :THEN: T(I+2)=EST. TIME
C
C      :ENDIF:
C
C  :ENDO:
C
C  -------------------------------------------------------------------
C
C
      IMPLICIT  INTEGER  ( I - N )
C
C
      DIMENSION       ELVAR   ( 41 ) , BEDTMR ( 41 ) , COLTMR ( 41 ) ,
     +                IRT     ( 41 ) , ROMX   ( 40 ) , CPMX   ( 40 ) ,
     +                XMUX    ( 40 ) , XKMX   ( 40 ) , TR     ( 20 ) ,
     +                T       (  N )
C
C
      COMMON        ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +              XMUCD  , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +              SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +              ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +              XKMX   , XMUF3  , RODEB
C
C
      LOGICAL       DISRUP
C
C
C
      NSTPR = 10
C
```

```
C
C CALL FGTIME TO BREAKUP THE TIME STEP FROM DBUNDL INTO SMALLER
C INTERVALS BASED ON NSTPR.
C
      CALL  FGTIME  ( T ( I ) , DELTT , TR , NSTPR )
C
      NSTEP = NSTPR
      J = 1
C
   10 CONTINUE
C
C FIRST TEST PREVENTS TIME = 0 FROM BEING A STARTING POINT
C
      IF  ( TR( J ) .EQ. 0.0 )    J = J + 1
C
      IF  ( J .LE. NSTEP )    THEN
C
          CURTIM = TR ( J )
          DELTR  = TR ( J + 1 ) - TR ( J )
C
C
C MATERIAL PROPERTY ADJUSTMENT NECESSARY FOR FRAGMENTED CHARACTER-
C ISTIC SUBROUTINE *FGCHAR* DEPENDING ON THE RELATIONSHIP OF THE
C COOLANT TEMPERATURE AND THE SATURATION TEMPERATURE
C
      IF  ( COLTMR( 1 ) .GE. TSAT )    THEN
C
          ROCOLD = ROVAP
          XMUCD  = XMUVAP
C
      ELSE
C
          ROCOLD = ROCOL
          XMUCD  = XMUCOL
C
      ENDIF
C
C CALL FGCHAR TO DEFINE THE CHARACTERISTICS OF FRAGMENTED BUNDLE
C
          CALL  FGCHAR  ( SIZAVR , XMASSR , ZRMR   , ZRO2MR , UO2MR  ,
     +                    STRMR  , ABSMR  , AREAR  , HITRG  , VINR   ,
     +                    VINROD , DELTR  , HITER  , POROSR , POROPR ,
     +                    XCHRL  , KFLUID , KTERM  )
C
C
C CALL FRAGTH TO DESCRIBE THE BEHAVIOR OF THE FRAGMENTED BUNDLE
C
          CALL  FRAGTH  ( SIZAVR , POROSR , HITER  , XCHRL  , VINR   ,
     +                    ZRO2TR , ZRO2AR , ALPHTR , ALPHAR , ELVNR  ,
     +                    HTLBRD , HTLBRC , HTUBRD , HTUBRC , ND     ,
     +                    MAXL   , DELTR  , IDREGR , NONR   , AREAR  ,
     +                    KFLUID , ELVAR  , BEDTMR , COLTMR , THMOTR ,
     +                    ELVMR  , OXDAR  , FISHR  , HYDGR  , FGRSR  ,
```

TABLE A-7. (CONTINUED)

```
     +                    VOLGSR , IRT    , DOWNR  , DISRUP )
C
C
C COMPUTE LIQUIFIED MATERIAL MOVEMENT IF DEBRIS BED MELTING OCCURS
C
      CALL  FGMELT  ( NONR   , IRT    , ELVMR  , ELVAR  , DELTR  ,
     +                VELMOT )
C
C
C
C TEST FOR LOWER REGION BOUNDARY BREAKTHROUGH
C
C RETURN TO DBUNDL IF BREAKTHRU OCCURS, WITH END OF TIME INTERVAL
C EQUAL TO CURRENT TIME
C
          IF  ( DISRUP )   THEN
C
              NSTEP = J
C
C
          ELSE
C
C COMPUTE ESTIMATED TIME OF DISRUPTION
C
              CALL  FGESTT  ( ELVMR  , ELVNR  , VELMOT , CURTIM ,
     +                        ESTDT  )
C
C
C IF ESTIMATED TIME OF BREAKTHROUGH IS GREATER THAN CURRENT TIME PLUS
C ADJUSTED DELTA T, SET CURRENT TIME TO NEXT TIME INTERVAL.
C
              IF  ( ESTDT  .GT. TR( J + 1 ) )   THEN
C
                  CURTIM = TR( J + 1 )
C
              ELSE
C
C IF ESTIMATED TIME OF BREAKTHROUGH IS LESS THAN CURRENT TIME PLUS
C ADJUSTED DELTA T, SET CURRENT TIME TO ESTIMATED TIME AND NEXT TIME
C INTERVAL TO CURRENT TIME
C
                  CURTIM = ESTDT
                  TR ( J + 1 ) = CURTIM
C
              ENDIF
C
          ENDIF
C
C WHEN J IS GREATER THAN NSTPR, CONTROL WILL RETURN TO DBUNDL WITH
C CTIMER CONTAINING THE CURRENT TIME
C
           J = J + 1
           GO  TO  10
C
      ENDIF
C
C
      RETURN
C
      END
```

TABLE A-8. LISTING OF FGTIME

```
C
C
      SUBROUTINE  FGTIME  ( TIMIN , DELTT , TR , NSTP )
C
C  SUBCODE NAME:  FGTIME
C  PURPOSE:  TO DEFINE THE RUBBLE DEBRIS TIME STEP
C  CALLING SUBROUTINES:  DBFRAG
C  SUBROUTINES CALLED:  NONE
C  WORK PACKAGE:  15
C  ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C  LAST MODIFICATION DATE:  11/30/81
C
C  INPUT VARIABLES                            DESCRIPTION
C
C     DELTIN          TIME STEP COMING FROM DISRUPTED BUNDLE TIME STEP
C                     SUBROUTINE - DBTIME
C
C  OUTPUT VARIABLE                            DESCRIPTION
C
C     DELTCT          TIME STEP WHICH WILL BE USED IN THE RUBBLE DEBRIS
C                     ANALYSIS - DELTIN / NSTP
C     NSTP            NUMBER OF STEPS INTO WHICH DBTIME TIME STEP WILL
C                     BE BROKEN
C
C
C
       DIMENSION       TR ( NSTP + 1 )
C
C
      STEPS = DELTT / FLOAT ( NSTP )
C
      DO  10  I = 1 , NSTP + 1
C
          TR ( I ) = TIMIN + FLOAT ( I - 1 ) * STEPS
C
   10 CONTINUE
C
      RETURN
      END
```

TABLE A-9. LISTING OF FGCHAR

```
      SUBROUTINE  FGCHAR  ( SIZAVR , XMASSR , ZRMR   , ZRO2MR , UO2MR ,
     +                      STRMR  , ABSMR  , AREAR  , HITRG  , VINR  ,
     +                      VINROD , DELTR  , HITER  , PUROSR , PORUPR ,
     +                      XCHRL  , KFLUID , KTERM  )
C
C
C
C
C        PURPOSE: TO DEFINE RUBBLE BED CHARACTERISTICS AND HYDRAULIC
C
C        BEHAVIOR
C
C        CALLING SUBROUTINE:DEFRAG
C
C        SUBRCUTINE CALLED: NONE
C
C        ENGINEER/PROGRAMER:STH
C
C        LAST DATE MODIFIED:8/10/81
C
C        INPUT VARIABLES:
C
C        SIZAVR     AVERAGE PARTICLE SIZE OVER ACCUMULATED MASS AND
C                   COMPONENTS(M)
C
C        XMASSR     TOTAL MASS OF DEBRIS ACCUMULATED FOR ALL COMPONENTS(KG)
C
C        ZRMR       TOTAL MASS OF ZIRCALOY ACCUMULATED FOR ALL COMPONENTS
C                   (KG)
C
C        ZRO2MR     TOTAL MASS OF ZRO2 ACCUMULATED FOR ALL COMPONENTS(KG)
C
C        UO2MR      TOTAL MASS OF UO2 ACCUMULATED FOR ALL COMPONENTS(KG)
C
C        STRMR      TOTAL MASS OF CORE STRUCTURE MATERIALS ACCUMULATED(KG)
C
C        ABSMR      TOTAL MASS OF CONTROL ROD MATERIAL ACCUMULATED(KG)
C
C        AREAR      TOTAL BUNDLE CROSS SECTIONAL AREA(M2)
C
C        HITRG      FRAGMENMTED DEBRIS REGION HEIGHT FROM DBREGN(M)
C
C        VINR       COOLANT INLET VELOCITY(M/SEC)
C
C        VINROD     COOLANT INLET VELOCITY AT PREVIOUS TIMNE STEP,M/SEC
C
C        DELTR      TIME STEP(SEC)
C
C
C
C
```

TABLE A-9. (CONTINUED)

```
C
C
C
C     OUTPUT VARIABLES:
C     HITER    BED HEIGHT, IF PACKED BED, DELTA H=HITRG-HITER=A VOID
C              AT THE TOP OF PACKED BED(M)
C
C     POROSR   BED POROSITY, FOR MODO PCROSR IS AN AVERAGED VALUE OVER
C              THE BED
C
C     PDROPR   PRESSURE DROP ACCROSS THE BED(PA)
C
C     XCHRL    CHARACTERISTIC LENGTH OF THE PARTICLE, DERIVED FROM
C              PRESSURE DROP EQUATION(M)
C
C     KFLUID   FLUIDIZATION FLAG, =0 PACKED BED, =1 FLUIDIZED BED
C
C     KTERM    ABNORMAL TERMINATION FLAG,=0 NORMAL RUNS, =1 ABNORMAL
C              TERMINATION
C
C
C
C
C
C
C
C
C     AT THIS TIME, ALL MATERIAL PROPERTIES ARE ASSUMED COMING
C     THROUGH A COMMON BLOCK, THESE INCLUDE:
C
C     ROZR      ZR DENSITY(KG/M3)
C
C     ROZRO2    ZRO2 DENSITY( KG/M3)
C
C     ROUO2     UO2 DINSITY(KG/M3)
C
C     ROSTR     DENSITY OF STRUCTURAL MATERIAL(KG/M3)
C
C     ROABS     DENSITY OF CONTROL ROD MATERIAL(KG/M3)
C
C     ROCOL     COOLANT DENSITY(KG/M3)
C
C     XMUCOL    VISCOSITY OF COOLANT(KG/SEC/M)
C
C
C
C
```

TABLE A-9. (CONTINUED)

```
      COMMON      ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOL  ,
     +            XMUCOL , DMY1   , DMY2   , ROVAP  , XMUVAP , HFG    ,
     +            SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +            ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +            XKMX   , XMUF3  , RODEB
C
C
C
      PI  =  3.14159
C
C
C CALCULATE DEBRIS VOLUME WITH ZERO POROSITY
C
      TOTVOL = ZRMR / ROZR + ZRO2MR / ROZRO2 + UO2MR / ROUO2 + STRMR
     +             / ROSTR + ABSMR / ROABS
C
C CALCULATE ZERO-POROSITY BED HEIGHT(M)
C
      HITSOL = TOTVOL / AREAR
C
C CHECK DEBRIS REGION HEIGHT
C
      IF ( HITSOL .GT. HITRG )   THEN
         KTERM = 1
C
         WRITE  ( 6 , 100 )
C
      ELSE
C
C DEBRIS BED PACKING IS BASED ON LMFBR EXPERIMENTS
C
         POROSR = 0.593 - 1.23E-4 * XMASSR / AREAR
C
C CALCULATE PACKED BED HEIGHT
C
         HITER = HITSOL / ( 1.0 - POROSR )
C
         IF  ( HITER .GE. HITRG )   THEN
             HITER = HITRG
             POROSR = 1.0 - HITSOL / HITRG
C
         ENDIF
C
C CALCULATE MINIMUM FLUIDIZATION VELOCITY(M/SEC)
C
         ROBED = XMASSR / ( AREAR * HITSOL )
C
         GMFBC = 688.0 * ( SIZAVR * 39.37 ) ** 1.82 *
     +             ( ROCOL * 0.06243 * ( ROBED - ROCOL ) * 0.06243 )
     +              ** 0.94 * ( XMUCOL * 1E+03 ) ** (-0.88 )
     +              * 0.4536 / 3600.0 / ( 0.3048 ) ** 2
C
         RENOD1 = GMFBC * SIZAVR / XMUCOL
```

```
C
          IF  ( RENOD1 .LT. 7.5731 )   THEN
C
              CORFCT = 1.0
C
          ELSE IF ( RENOD1 .GE. 7.5731  .AND. RENOD1 .LT. 200.0 ) THEN
C
              CORFCT = 1.364135 -0.179864 * ALOG ( RENOD1 )
C
          ELSE  IF  ( RENOD1 .GE. 200.0  .AND. RENOD1 .LT. 1E+03 )  THEN
C
              CORFCT = 0.214413 + 39.3522 / RENOD1
C
          ELSE  IF  ( RENOD1 .GE. 1E+03 )   THEN
C
              CORFCT = 0.253764
C
          ENDIF
C
          RENOD2 = RENOD1 * CORFCT
          GMFBC  = GMFBC * CORFCT
          VELMF  = GMFBC / ROCOL
C
C
C CHECK PACKED OR FLUIDIZED BED
C
           IF  ( VINR .LE. VELMF )   THEN
               XITM1 = (VINR - VINROD) * ROCOL / DELTP
               ALPHA = 150.0 * (1.0 - POROSR) ** 2
    +                / (AMAX1(POROSR , 1.0E-8)) ** 3 / (SIZAVR ** 2 )
               BETA = 1.75 * (1.0 - POROSR)
    +                / (AMAX1(POROSR , 1.0E-8)) ** 3 / SIZAVR
C
C CALCULATE XCHRL, CHARACTERISTIC LENGTH OF PACKED PARTICLE
C
               XCHRL = BETA / ( AMAX1(ALPHA , 1.0E-8) )
C
C CALCULATE PRESSURE DROP, ERGUN'S EQUATION
C
               PDROPR = XITM1 + (ALPHA * XMUCOL + BETA * ROCOL
    +                         * VINR ) * VINR
               PDROPR = PDROPR * HITER
C
               KTERM = 0
               KFLUID = 0
C
           ELSE
               PDROPR = (XMASSR / AREAR - ROCOL * HITSOL) * 9.80665
C
C       ABOVE IS THE CALCULATION OF PRESSURE DROP ACROSS FLUIDIZED BED
C
C
C       IF RUBBLE BED IS FLUIDIZED, CALCULATE BED HEIGHT AND FLUIDIZED
C BED POROSITY ACCORDING TO LEVA'S FLUIDIZATION, 1959, P. 87.
```

```
C
C
            RENOD3 = SIZAVR * ROCOL * VINR / XMUCOL
            DIABND = ( AREAR * 4.0 / PI ) ** 0.5
C
            IF  ( RENOD3 .LT. 0.2 )   THEN
C
                SLOPC = 5.0
C
            ELSE IF ( RENOD3 .GE. 0.2  .AND. RENCD3 .LT. 1.0)  THEN
C
                SLOPC = ( 4.35 + 17.5 * SIZAVR / DIABND )
     +                * RENOD3 ** ( -0.03 )
C
            ELSE  IF  ( RENOD3 .GE. 1.0  .AND. RENCD3 .LT. 200.0) THEN
C
                SLOPC = (4.45 + 18.0 * SIZAVR / DIABND)
     +                * RENOD3 ** ( -C.1 )
C
            ELSE  IF  (RENOD3 .GE. 200.0  .AND. RENOD3 .LT. 500.) THEN
C
                SLOPC = 4.45 * RENOD3 ** ( -0.1 )
C
            ELSE  IF  ( RENOD3 .GE. 500.0 )   THEN
C
                SLOPC = 2.39
C
            ENDIF
C
            PORIMT = ( RENOD3 / RENOD2 * POROSR ** SLOPC )
     +                    ** ( 1.0 / SLCPC )
C
C
            WRITE ( 6 , 500 ) RENOD2 , RENOD3 , POROSR , SLOPC ,PORIMT
C
            PORIMT = AMIN1 ( PORIMT , 1.0 )
            HITER = HITER * ( 1.0 - POROSR ) /
     +                      AMAX1( 1.0E-08 , ( 1.0 - PORIMT ) )
            POROSR = PORIMT
C
            IF  ( HITER .GE. HITRG )   THEN
C
                HITER = HITRG
                POROSR = 1.0 - HITSOL / HITRG
C
            ENDIF
C
            ALPHA = 150.0 * (1.0 - POROSR) ** 2 / (AMAX1(POROSR ,
     +                1.0E-8)) ** 3 / (SIZAVR ** 2)
            BETA = 1.75 * (1.0 - POROSR) / (AMAX1(POROSR,1.0E-08))
     +                ** 3 / SIZAVR
            XCHRL = BETA / ( AMAX1(ALPHA , 1.0E-8) )
C
```

TABLE A-9. (CONTINUED)

```
              KFLUID = 1
              KTERM = 0
C
C
            ENDIF
C
      ENDIF
C
 100  FORMAT(2X,50HTOO MUCH MASS IN RUBBLE BED, CHECK FRAGMENTED MASS)
      WRITE(6,700)VELMF
 700  FORMAT(2X,6HVELMF=,E12.4)
C
C
  500 FORMAT  ( 2X , 7HRENOD2=,E12.5 , 7HRENOD3=,E12.5 , 7HPOROSR=,
     +          E12.5 , 6HSLOPC=,E12.5 , 7HPORIMT=,E12.5)
C
C
      RETURN
      END
```

TABLE A-10. LISTING OF FRAGTH

```
      SUBROUTINE  FRAGTH  ( SIZAVR , POROSR , HITER  , XCHRL  , VINR   ,
     +                      ZRO2TR , ZRO2AR , ALPHTR , ALPHAR , ELVNR  ,
     +                      HTLBRD , HTLBRC , HTUBRD , HTUBRC , ND     ,
     +                      MAXL   , DELTR  , IDREGR , NONR   , AREAR  ,
     +                      KFLUID , ELVAR  , BEDTMR , COLTMR , THMOTR ,
     +                      ELVMR  , OXDAR  , FISHR  , HYDGR  , FGRSR  ,
     +                      VOLGSR , IRT    , DOWNR  , DISRUP )
C
C
C
C     PURPOSE: TO CALCULATE THERMALLY RELATED BEHAVIORS OF A RUBBLE
C     BED.
C
C     CALLING SUBROUTINE: DBFROZ
C
C     SUBROUTINE CALLED: TEMPSR
C
C     ENGINEER/PROGRAMMER:  S. HSIEH
C
C     LAST DATE MODIFIED: 8/10/81
C
C     INPUT VARIABLES:
C
C     SIZAVR    AVERAGE PARTICLE DIAMETER OF A RUBBLE DEBRIS(M)
C
C     POROSR    DEBRIS BED POROSITY
C
C     HITER     BED HEIGHT(M)
C
C     XCHRL     CHARACTERISTIC LENGTHOF RUBBLE BODY FROM DELT P, M
C
C     VINR      COOLANT INLET VELOCITY(M/SEC), NOTE THAT VINR IS THE
C               SUPERFICIAL VELOCITY OF THE FLUID MEASURED ON AN
C               EMPTY TUBE BASIS. THE TRUE COOLANT VELOCITY IN THE
C               COOLANT CHANNEL VINCT(URE)=VINC/POROSR.
C
C     ZRO2TR    EFFECTIVE ZRO2 REACTION LAYER THICKNESS(M)
C
C     ZRO2AR    EFFECTIVE ZR-STEAM REACTION AREA(M2)
C
C     ALPHTR    EFFECTIVE ALPHA-ZR REACTION LAYER THICKNESS,M
C
C     ALPHAR    EFFECTIVE ALPHA-ZR REACTION AREA(M2)
C
C     ELVNR     ELEVATION OF RUBBLE BODY-FROM THE BOTTOM OF ROD BUNDLE
C               TO THE BOTTOM OF RUBBLE BODY REGION(M)
C
C
C     HTLBRD    HEAT TRANSFER INTO DEBRIS BED AT LOWER BOUNDARY, W/M2
C
C     HTLBRC    HEAT TRANSFER INTO DEBRIS COOLANT AT LOWER BOUNDARY,W/M2
C
C     HTUBRD    HEAT TRANSFER INTO DEBRIS BED AT UPPER BOUNDARY, W/M2
```

TABLE A-10. (CONTINUED)

```
C
C     HTUBRC     HEAT TRANSFER INTO DEBRIS COOLANT AT UPPER BOUNDARY,W/M2
C
C     ND         NUMBER OF NODES TO BE USED IN LIQUIC OR VAPOR REGION
C
C     MAXL       MAXIMUM NUMBER OF NODES IN THE RUBBLE BODY WHICH
C                COMBINED LIQUID, SATURATION AND VAPOR REGION,
C
C     DELTR      TIME STEP(S)
C
C     AREAR      TOTAL BUNDLE CROSS SECTIONAL AREA,M2
C
C     KFLUID     FLUIDIZATION FLAG, =0 PACKED BED, =1 FLUIDIZED
C                BEDF
C
C
C
C     INPUT/OUTPUT VARIABLES:
C
C     IDREGR     REGION TYPE I.D. =1 VINR GT 0 ONE LIQUID REGION
C                                 =2 VINR GT 0 ONE VAPOR GEGION
C                                 =3 VINR GT 0 OND SAT. REGION
C                                 =4 VINR GT 0 TWO REG,L & S
C                                 =5 VINR GT 0 TWO REG,S & V
C                                 =6 VINR GT 0 THREE REG L&S&V
C                                 =7 VINR=0 ONE LIQUID REG
C                                 =8 VINR=0 ONE VAPOR REG
C
C     NONR       NUMBER OF NEDES USED IN RUBBLE BODY ANALYSIS,
C                =ND  IDREGR=1,2,7,8
C                =3   IDREGR=3
C                =ND+2 IDREGR=4,5
C                =2*ND+1 IDREGR=6
C
C     ELVAR      ELEVATION OF NODES IN DEBRIS, ELVAR(1)=ELVNR
C                ELVAR(NONR)=ELVNR+HITER
C
C     BEDTMR     DEBRIS TEMPERATURE CORRESPONDING TO NONR NODES
C
C     COLTMR     COOLANT TEMPERATURE CORRESPONDING TO NONR NODES
C
C     OXDAR      ZR-STEAM HEAT GENERATION RATE(W/M3)
C
C     FISHR      FISSION/DECAY HEAT GENERATION RATE(W/M3)
C     ELVMR      MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY,M
C
C     OUTPUT VARIABLES:
C
C     THMOTR     MOLTEN MATERIAL THICKNESS(M)
C
C     HYDGR      HYDROGEN GENERATION DURING DELTR(MOLES)
```

```
C
C      FGRSR     FISSION GAS RELEASE(MOLES) DURING DELTR
C
C      VOLGSR    VOLATILE FISSION PRODUCTS RELEASE DURING DELTR, MOLE
C
C      IRT       DEBRIS REGION TYPE, =0 SOLID REGION, =1 MOLTEN REGION
C                TOTAL OF NONR VALUES
C
C      DOWNR     LENGTH OF MOLTEN POOL PENETRATION, M
C
C
C                AFTER CURRENT TIME, (S)
C
C      DISRUP    DISRUPTION FLAG
C
C
C
C
C      AT THIS TIME, ALL MATERIALS PROPERTIES ARE ASSUMED COMING
C      THROUGH COMMON BLOCK, THESE INCLUDE:
C
C      ROCOL     LIQUID COOLANT DENSITY(KG/M3)
C
C      ROVAP     STEAM DENSITY(KG/M3)
C
C      XMUCOL    LIQUID COOLANT VISCOSITY(KG/SEC/M)
C
C      XMUVAP    VAPOR VISCOSITY(KG/SEC/M)
C
C      HFG       LATENT HEAT OF VAPORIZATION(J/KG) OF COOLANT
C
C      SURTC     SURFACE TENSION OF COOLANT(PA)
C
C      CPCOL     LIQUID SPECIFIC HEAT(J/K/KG)
C
C      XKCOL     LIQUID COOLANT THERMAL CONDUCTIVITY)W/M/K)
C
C      XKVAP     VAPOR(STEAM) THERMAL CONDUCTIVITY(W/M/K)
C
C      CPDEB     SPECIFIC HEAT OF DEBRIS(J/K/KG)
C
C      ROZR      ZR DENSITY(KG/M3)
C
C      ROZRO2    ZRO2 DENSITY(KG/M3)
C
C      ROUO2     UO2 DENSITY(KG/M3)
C
C      ROSTR     DENSITY OF STRUCTURAL MATERIAL(KG/M3)
C
C      ROABS     DENSITY OF CONTROL ROD MATERIAL(KG/M3)
C
C      RODEB     DEBRIS BED DENSITY(POROSITY DEPENDENT)
C
```

```
C     XMUDEB    VISCOSITY OF DEBRIS(KG/SEC/M)
C
C     XKDEB     THERMAL CONDUCTIVITY OF DEBRIS(W/M/K)
C
C     TSATC     SATURATION TEMPERATURE OF COOLANT OF RUBBLE DEBRIS
C               ENVIRONMENT,K
C
C     ROSAT     WATER DENSITY AT SATURATION(KG/M3)
C
C     TMELT     MELTING TEMPERATURE OF DEBRIS MATERIAL(K)
C
C     XLATC     LATENT HEAT OF FUSION FOR DEBRIS MATERIAL(J/KG)
C
C
C
      DIMENSION   ELVAR(41)  , BEDTMR(41) , COLTMR(41) , IRT(41)    ,
     +            BEDTMO(41) , COLTMO(41) , ROMX(40)   , CPMX(40)   ,
     +            XMUX(40)   , XKMX(40)   , TEMPBO(20) , TEMPB(20)  ,
     +            TCOLTO(20) , TCOLT(20)
C
C
      COMMON      ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +            XMUCD  , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +            SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +            ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +            XKMX   , XMUF3  , RODEB
C
C
      LOGICAL     DISRUP
C
C
      DATA    PI / 3.14159 /
C
C CALCULATE VOLUMETRIC HEAT GENERATION RATE QVOL(W/M3)
C
      QVOL = OXDAR + FISHR
      IITMP = 0
      ELVMO = ELVMR
C
C
C BASED ON QVOL,VINR, AND COLTMR CHECK THE REGION ID
C
 2020 CONTINUE
C
      XLLIQ = -100.0
      XLSAT = -100.0
      XLVAP = -100.0
C
C
C CALCULATE DRYOUT HEAT FLUX, LIPINSKI'S MODEL
C
      CALL DRYOUT  (VINR,POROSR,HITER,SIZAVR,PDROPR,QVOL,ROCOL,ROVAP,
     +              XMUCOL,XMUVAP,HFG,SURTC,QDYOUT)
```

TABLE A-10. (CONTINUED)

```
C
      IF  ( VINR .LT. 0.0 )   THEN
C
C COMPARE QVOL TO QDYOUT
C
          IF ( QVOL * HITER * (1.0-POROSR)  .GE.  QDYOUT )  THEN
              IDR = 8
C
          ELSE
              IDR = 7
C
          ENDIF
C
      ELSE
C
C REGION TYPE 1-6 LEFT TO BE IDENTIFIED
C
          IF ( COLTMR(1) .GT. TSAT )   THEN
              IDR = 2
C
          ELSE
              IF  ( COLTMR(1) .EQ. TSAT )   THEN
C
                  IF  ( COLTMR(NONR) .EQ. TSAT )   THEN
                      IDR = 3
C
                  ELSE
C
C CALCULATE SATURATION REGION LENGTH
C
                      XLSAT = ROCOL * VINR * HFG / QVOL
     +                      / AMAX1 ( (1.0 - POROSR ) , 1.0E-08 )
                      IDR = 5
                      XLVAP = HITER - XLSAT
C
                      IF (XLSAT .GE. HITER)   IDR = 3
                      IF (COLTMR(NONR) .LT. TSAT)   IDR = 3
                      IF (QVOL*HITER*(1.0-POROSR) .LT. QDYOUT)
     +                                IDR = 3
C
                  ENDIF
C
              ELSE
C
C COLTMR(1)  .LT.   TSAT
C
                  IF (COLTMR(NONR) .LT. TSAT)   THEN
                      IDR = 1
C
                  ELSE
                      XLSAT = ROCOL * VINR * HFG / QVOL
     +                        / AMAX1((1.0 - POROSR) , 1.0E-8)
C
```

TABLE A-10. (CONTINUED)

```
                     IF ( XLSAT .GE. HITER )    XLSAT = HITER
C
                     XLLIQ = XLSAT / HFG * CPCOL*(TSAT-COLTMR(1))
C
                     IF  ( XLLIQ .GE. HITER )  XLLIQ = HITER
C
                     IF (COLTMR(NONR) .EQ. TSAT)    THEN
                         IDR = 4
                         XLSAT = HITER - XLLIQ
C
                         IF ( XLSAT .LE. 0.0 )     IDR = 1
C
                     ELSE
                         IDR = 6
                         XLVAP = AMAX1 ( 0.0,( HITER- XLLIQ - XLSAT ))
C
                         IF ( QVOL*(HITER-XLLIQ)*(1.0-POROSR)
     +                            .LT. QDYOUT )    THEN
C
                             IDR = 4
                             XLSAT = HITER - XLLIQ
C
                             IF  ( XLSAT .LE. 0.0 )   IDR = 1
C
C
                         ELSE  IF ((XLLIQ+XLSAT) .GE. HITER)  THEN
                             IDR  = 4
                             XLSAT = HITER -XLLIQ
C
C
                             IF  ( XLSAT .LE. 0.0 )  IDR = 1
C
                         ENDIF
C
                     ENDIF
C
                 ENDIF
C
              ENDIF
C
          ENDIF
C
      ENDIF
C
C     DEBRIS REGION TYPE HAS BEEN IDENTIFIED,IDR=1-8,IF IDR AND
C     IDREGR ARE NOT CONSISTANT, CALL SUBROUTINE REGMOD TO ADJUST
C     TEMPERATURE OF BED AND COOLANT,NODES ELEVATION AND SET IDREGR
C     EQUALS TO IDR
C
C
      IF  ( IDR .NE. IDREGR )    THEN
C
      CALL REGMOD  (IDR,MAXL,ND,IDREGR,NONR,XLLIQ,XLSAT,XLVAP,
```

TABLE A-10. (CONTINUED)

```
     +                             ELVNR,TSAT,HITER,ELVAR,BEDTMR,COLTMR)
C
      ENDIF
C
      N2 = ND * 2
C
      DO  50  I = 1 , NONR
C
          BEDTMO(I) = BEDTMR(I)
          COLTMO(I) = COLTMR(I)
C
   50 CONTINUE
C
C     BASED ON CCLTMR AND BEDTMR, ROMX(N2),CPMX(N2), XMUX(N2) AND
C     XKMX(N2) ARE KNOWN ASSUME THEY ARE TEMPERATURE DEPENDENT,
C     FIRST N VALUES ARE FOR BED AND THE REST N VALUES ARE FOR
C     COOLANT.
C
C     NOW CALCULATE TEMPERATUR OF BED AND COOLANT FOR CURRENT TIME
C
      IF ( IDREGR .EQ. 1 .OR. IDREGR .EQ. 2 )   THEN
C
          IF  ( IDREGR .EQ. 1 )   IDYOT = 0
          IF  ( IDREGR .EQ. 2 )   IDYOT = 1
C
          VAR1 = 0.0
          VAR2 = 0.0
          VAR3 = 0.0
          VAR4 = 0.0
C
C
C CALL MATPRO (BEDTMO,COLTMO,ND,N2,ROMX,CPMX,XKMX,XMUX,KDREGR,IDYOT)
C
C
C
          DO  100  I = ND + 1 , N2
C
              VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
              VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
              VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
              VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
  100     CONTINUE
C
          XRENO = VAR1 * VINR * XCHRL / VAR4
          XPRDL = VAR2 * VAR4 / VAR3
C
          XNU = (7.0 - (10.0 - 5.0*POROSR) * POROSR) * (1.0 + 0.7
     +        * XRENO ** 0.2 * XPRDL ** (1.0/3.0)) + (1.33-(2.4-1.2
     +        * POROSR) * POROSR) * XRENO ** 0.7 * XPRDL ** (1.0/3.0)
C
          HVSC = XNU * VAR3 / XCHRL ** 2
C
C
```

TABLE A-10. (CONTINUED)

```
         DELTZ = HITER / FLOAT(ND - 1)
C
         IF ( IDYOT .EQ. 0 .AND. KFLUID .EQ. 1 )   THEN
C
             IF ( POROSR .LT. 0.25 )    GO  TO  123
C
             FEBSN = ( 1.0 - (1.0 - POROSR) ** (2.0/3.0) * PI
     +             ** (1.0/3.0) * (0.75) ** (2.0/3.0)) ** (-1.0)
C
C XMUF3 IS WATER VISCOSITY AT 300 K FROM MATPRO
C
             DIABN = ( 4.0 * AREAR / PI) ** 0.5
             XNU = 1.28E-5*(XRENO*FEBSN)**2*XPRDL**(0.67)*(VAR4/
     +             XMUF3)**0.83*(DIABN/XCHRL)**0.5*(RODEB/VAR1)**2
             HVSC = XNU * VAR3 / XCHRL ** 2
C
         ENDIF
C
 123     CONTINUE
C
         IF ( IDYOT .EQ. 1 )   THEN
C
             HVSC = ( XCHRL * ( 1.0 - POROSR ) / .00377 ) ** 1.33 *
     +              XRENO ** .65 * VAR3 / XCHRL ** 2
C
         ENDIF
C
         CALL  TEMPSR  ( ND     , N2     , IDYOT  , BEDTMO , COLTMO ,
     +                   HVSC   , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                   POROSR , VINR   , QVOL   , DELTR  , DELTZ  ,
     +                   BEDTMR , COLTMR )
C
      ELSE IF   ( IDREGR .EQ. 7 .OR. IDREGR .EQ. 8 )    THEN
C
         XNU = 7.0 - ( 10.0 - 5.0 * POROSR ) * POROSR
C
         IF ( IDREGR .EQ. 7 )    IDYOT = 0
         IF ( IDREGR .EQ. 8 )    IDYOT = 1
C
         VAR1 = 0.0
C
C
C CALL MATPRO ( ...,...
C
C
         DO  200  I = ND + 1 , N2
C
             VAR1 = VAR1 + XKMX(I) / FLOAT(ND)
C
 200     CONTINUE
C
         HVSC = XNU * VAR1 / XCHRL ** 2
         DELTZ = HITER / FLOAT( ND - 1 )
C
```

TABLE A-10. (CONTINUED)

```
        IF ( IDYOT .EQ. 1 )    THEN
C
           HVSC = ( XCHRL * ( 1.0 - POROSR ) / .00377 ) ** 1.33 *
    +                  XRENO ** .65 * VAR3 / XCHRL ** 2
C
        ENDIF
C
C
        CALL  TEMPSR  ( ND     , N2     , IDYOT  , BEDTMO , COLTMO ,
    +                   HVSC   , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
    +                   POROSR , VINR   , ZVOL   , DELTR  , DELTZ ,
    +                   BEDTMR , COLTMR )
C
C
C    IF IDREGR EQUALS TO 3, QVOL IS USED TO HEAT SATURATED WATER
C    NO BED AND COOLANT TEMPERATURE CHANGES ASSUMED, I.E. NO
C    CALCULATION OF TEMPERATURE IS NEEDED
C
     ELSE IF  ( IDREGR .EQ. 4 )    THEN
        IDYOT = 0
        DELTZ = XLLIQ / FLOAT( ND - 1 )
C
        VAR1 = 0.0
        VAR2 = 0.0
        VAR3 = 0.0
        VAR4 = 0.0
C
C CALL MATPRO ( ... , ... , ...
C
C
C
        DO  300  I = ND + 1 , N2
C
           VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
           VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
           VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
           VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
 300    CONTINUE
C
        XRENO = VAR1 * VINR * XCHRL / VAR4
        XPROL = VAR2 * VAR4 / VAR3
C
        XNU = (7.0 - (10.0 - 5.0*POROSR)*POROSR) * (1.0 + 0.7
    +              * XRENO ** 0.2 *XPROL ** (1.0/3.0))+(1.33-(2.4-1.2
    +              * POROSR) * POROSR) * XRENO**0.7*XPROL**(1.0/3.0)
C
        HVSC = XNU * VAR3 / XCHRL ** 2
C
        IF  ( KFLUID .EQ. 1 )    THEN
C
           IF  ( POROSR .LT. 0.25 )    GO  TO  323
C
```

TABLE A-10. (CONTINUED)

```
                FEBSN = (1.0-(1.0 - POROSR) ** (2.0/3.0) * PI
     +                   ** (1.0/3.0) * (0.75) ** (2.0/3.0)) ** (-1.0)
                DIABN = ( 4.0 * AREAR / PI ) ** (0.5)
                XNU = 1.28E-5*(XRENO*FEBSN)**2*XPRDL**0.67*(VAR4 /
     +                XMUF3)**0.83*(DIABN/XCHRL)**0.5*(RODEB/VAR1)**2
                HVSC = XNU * VAR3 / XCHRL ** 2
C
           ENDIF
C
  323      CONTINUE
C
           DO  400  I = 1 , ND
C
                TEMPBO(I) = BEDTMO(I)
                TCOLTO(I) = COLTMO(I)
C
  400      CONTINUE
C
C
           CALL  TEMPSR  ( ND     , N2     , IDYOT  , TEMPBO , TCOLTO ,
     +                     HVSC   , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                     POROSR , VINR   , QVOL   , DELTR  , DELTZ  ,
     +                     TEMPB  , TCOLT  )
C
C
           DO  500  I = 1 , ND
C
                BEDTMR(I) = TEMPB(I)
                COLTMR(I) = TCOLT(I)
C
  500      CONTINUE
C
           BEDTMR(ND+1) = AMAX1( BEDTMO( ND + 1 ) , BEDTMR( ND ))
           BEDTMR(ND+2) = AMAX1( BEDTMO( ND + 1 ) , BEDTMR( ND ))
           COLTMR(ND+1) = COLTMR(ND)
           COLTMR(ND+2) = COLTMR(ND)
C
      ELSE IF(IDREGR.EQ.5)THEN
C
           IDYOT = 1
           DELTZ = XLVAP / FLOAT( ND - 1 )
C
           VAR1 = 0.0
           VAR2 = 0.0
           VAR3 = 0.0
           VAR4 = 0.0
C
C  CALL MATPRO ( ... , ... , ...
C
C
C
C
           DO  600  I = ND + 1 , N2
C
```

TABLE A-10. (CONTINUED)

```
                VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
                VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
                VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
                VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
  600       CONTINUE
C
            XRENO = VAR1 * VINR * XCHRL / VAR4
            XPRDL = VAR2 * VAR4 / VAR3
C
            XNU = (7.0 - (10.0 - 5.0*POROSR) * POROSR) * (1.0+0.7
     +           * XRENO ** 0.2 * XPRDL ** (1.0/3.0))+(1.33-(2.4-1.2
     +           * POROSR)* POROSR)* XRENO ** 0.7 * XPRDL**(1.0/3.0)
C
            HVSC = XNU * VAR3 / XCHRL ** 2
C
            DO  700  I = 1 , ND
C
                TEMPBO(I) = BEDTMO(I+2)
                TCOLTO(I) = COLTMO(I+2)
C
  700       CONTINUE
C
            IF  ( IDYOT .EQ. 1 )    THEN
C
                HVSC = (XCHRL * ( 1.0 - POROSR ) / .00377 ) ** 1.33 *
     +                 XRENO ** .65 * VAR3 / XCHRL ** 2
C
            ENDIF
C
C
            CALL  TEMPSR  ( ND     , N2     , IDYOT  , TEMPBD , TCOLTO ,
     +                      HVSC   , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                      POROSR , VINR   , QVOL   , DELTR  , DELTZ  ,
     +                      TEMPB  , TCOLT  )
C
C
            DO  800  I = 1 , ND
C
                BEDTMR(I+2) = TEMPB(I)
                COLTMR(I+2) = TCOLT(I)
C
  800       CONTINUE
C
            COLTMR(1) = COLTMR(3)
            COLTMR(2) = COLTMR(3)
C
            BEDTMR(1) = AMAX1( BEDTMO( 1 ) , BEDTMR( 3 ))
            BEDTMR(2) = AMAX1( BEDTMO( 2 ) , BEDTMR( 3 ))
C
        ELSE IF  ( IDREGR .EQ. 6 )    THEN
C
            IDYOT = 0
```

TABLE A-10. (CONTINUED)

```
          DELTZ = XLLIQ / FLOAT( ND - 1 )
C
 1200     VAR1 = 0.0
          VAR2 = 0.0
          VAR3 = 0.0
          VAR4 = 0.0
C
C
C CALL MATPRO ( ... , ... , ...
C
C
C
          DO  900  I = ND + 1 , N2
C
              VAR1 = VAR1 + ROMX(I) / FLOAT(ND)
              VAR2 = VAR2 + CPMX(I) / FLOAT(ND)
              VAR3 = VAR3 + XKMX(I) / FLOAT(ND)
              VAR4 = VAR4 + XMUX(I) / FLOAT(ND)
C
  900     CONTINUE
C
          XRENO = VAR1 * VINR * XCHRL / VAR4
          XPRDL = VAR2 * VAR4 / VAR3
          XNU = (7.0 - (10.0 - 5.0*POROSR) * POROSR)*(1.0+0.7
     +          * XRENO ** 0.2 * XPRDL**(1.0/3.0))+(1.33-(2.4-1.2
     +          * POROSR) * POROSR) * XRENO**0.7*XPRDL**(1.0/3.0)
C
          HVSC = XNU * VAR3 / XCHRL ** 2
C
          IF  ( IDYOT .EQ. 0 .AND. KFLUID .EQ. 1 )   THEN
C
              IF  ( POROSR .LT. 0.25 )    GO  TO  923
C
              FEBSN = ( 1.0 - (1.0 - POROSR) ** (2.0/3.0) * PI
     +               ** (1.0/3.0) * (0.75) ** (2.0/3.0)) ** (-1.0)
C
C XMUF3 IS THE WATER VISCOSITY AT 300 K FROM MATPRO
C
              DIABN = ( 4.0 * AREAR / PI) ** (0.5)
C
              XNU = 1.28E-5 * (XRENO * FEBSN) ** 2
     +              * XPRDL ** 0.67 * (VAR4 / XMUF3) ** 0.83 *
     +              (DIABN / XCHRL) ** 0.5 * (RODEB / VAR1 ) ** 2
C
              HVSC = XNU * VAR3 / XCHRL ** 2
C
          ENDIF
C
  923     CONTINUE
C
C
          DO  1000  I = 1 , ND
C
```

TABLE A-10. (CONTINUED)

```
            TEMPBO(I) = BEDTMO(I)
            TCOLTO(I) = COLTMO(I)
C
 1000     CONTINUE
C
          CALL  TEMPSR  ( ND     , N2     , IDYOT  , TEMPBO , TCOLTO ,
     +                    HVSC   , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                    POROSR , VINR   , QVOL   , DELTR  , DELTZ  ,
     +                    TEMPB  , TCOLT  )
C
C
          DO  1100  I = 1 , ND
C
            BEDTMR(I) = TEMPB(I)
            COLTMR(I) = TCOLT(I)
C
 1100     CONTINUE
C
          BEDTMR(ND+1) = AMAX1(BEDTMO( ND+1 ),BEDTMR( ND ))
          COLTMR(ND+1) = COLTMR(ND)
C
          IDYOT = 1
          DELTZ = XLVAP / FLOAT( ND - 1 )
C
C
C      COOLANT PROPERTIES ARE VAPOR PORPERTIES NOW
C
          VAR1 = 0.0
          VAR3 = 0.0
C
C
C CALL MATPRO ( ...,...,...
C
C
          DO  1310  I = ND + 1 , N2
C
            VAR1 = VAR1 + ROMX (I) / FLOAT ( ND )
            VAR3 = VAR3 + XKMX (I) / FLOAT ( ND )
C
 1310     CONTINUE
C
C
          IF  ( IDYOT .EQ. 0 )   GO  TO  1600
C
          DO  1400  I = 1 , ND
C
            TEMPBO(I) = BEDTMO ( I + ND + 1 )
            TCOLTO(I) = COLTMO ( I + ND + 1 )
C
 1400     CONTINUE
C
          IF  ( IDYOT .EQ. 1 )  THEN
C
```

TABLE A-10. (CONTINUED)

```
              HVSC = ( XCHRL * ( 1.0 - POROSR ) / .00377 ) ** 1.33 *
     +                     XRENO ** .65 * VAR3 / XCHRL ** 2
C
          ENDIF
C
C
          CALL  TEMPSR  ( ND     , N2     , IDYOT  , TEMPBO , TCOLTO ,
     +                    HVSC   , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                    POROSR , VINR   , QVOL   , DELTR  , DELTZ  ,
     +                    TEMPB  , TCOLT  )
C
C
          DO  1500  I = 1 , ND
C
              BEDTMR(I+ND+1) = TEMPB(I)
              COLTMR(I+ND+1) = TCOLT(I)
C
 1500     CONTINUE
C
 1600     CONTINUE
C
      ENDIF
C
C     CHECK DEBRIS BED TEMPERATURE AND DEFINE MELTING REGION
C
      IMG = 0
      ISG = 0
      DOWNR = 0.0
C
      DO  2000  I = 1 , NONR - 1
C
          IF ( (BEDTMR(I)+BEDTMR(I+1))/2.0 .GT. TMELT )  THEN
C
C     TMELT IS DEBRIS BED MELTING TEMPERATURE FROM MAPRO
C
              IMG = IMG + 1
              JUP = I + 1
              IRT(I) = 1
C
          ELSE
              ISG = ISG + 1
              IRT(I) = 0
C
          ENDIF
C
 2000 CONTINUE
C
      IRT( NONR ) = IRT( NONR - 1 )
C
      DO  2010  I = 1 , NONR - 1
C
          J = NONR - I
          DTMP1 = ( BEDTMR(J+1) + BEDTMR(J)) / 2.0 - TMELT
```

```
C
          IF  ( DTMP1 .LT. 0.0 )   THEN
              DTMP1 = 0.0
C
          ELSE
              DOWNR = DOWNR + DTMP1 * ( ELVAR(J+1)
     +                               - ELVAR (J) ) * CPDEB / XLATC
C
              BEDTMR(J+1) = TMELT
              BEDTMR(J) = TMELT
C
          ENDIF
C
          IF  ( J .EQ. 1 )   GO  TO  2010
C
C
          IF  ( IRT(J) .GT. IRT(J-1) )   THEN
C
C      MOLTEN MATERIALS MELTING THE LOWER NODE
C
              IF (DOWNR .GT. (0.5*(ELVAR(J)-ELVAR(J-1))) )  THEN
C
                  BEDTMR(J) = TMELT
                  BEDTMR(J-1) = TMELT
                  IRT(J-1) = 1
                  DOWNR = DOWNR - (ELVAR(J) - ELVAR(J-1))
C
              ENDIF
C
          ENDIF
C
 2010 CONTINUE
C
C
C
      THMOTR = 0.0
C
      IF ( IRT (1) .EQ. 1 )   THEN
          THMOTR = THMOTR + 0.5 * ( ELVAR (2) - ELVAR (1) )
C
C   SET DISRUPTION FLAG WHEN DEBRIS REGION TYPE = 1, INDICATING MELTING
C
          DISRUP = .TRUE.
C
      ENDIF
C
      IF  ( IRT ( NONR ) .EQ. 1 )   THEN
          THMOTR = THMOTR + 0.5 * ( ELVAR (NONR) - ELVAR ( NONR - 1 ))
C
      ENDIF
C
      DO  3020  I = 2 , NONR - 1
C
```

TABLE A-10. (CONTINUED)

```
          IF  ( IRT (I) .EQ. 1 )   THEN
              THMOTR = THMOTR + 0.5 * ( ELVAR (I + 1) - ELVAR (I - 1))
C
          ENDIF
C
 3020 CONTINUE
C
C
C
C
C NOW ADJUSTED BED TEMPERATURE AND MELTEN REGION HAS BEEN DEFINED
C
C     CALL CHEMHT(ZRO2TR,ZRO2AR,ALPHTR,ALPHAR,IDREGR,NONR,BEDTMR,
C    +        COLTMR,ELVAR,DELTR,OXDAR,HYDGR)
C
C     CALL NUCLHT(IDREGR,NONR,BEDTMR,ELVAR,DELTR,TOTIME,...ETC)
C
      QVOLN = OXDAR + FISHR
      IITMP = IITMP + 1
C
      IF  ( IITMP .GT. 50 )   THEN
C
          WRITE   ( 6 , 2050 )
 2050 FORMAT  (2X, 57HITERATION MORE THAN 50 TIMES NO CONVERGENCE IN TEM
     +P. CAL.)
C
          GO  TO  2030
C
      ENDIF
C
      IF  ( ABS((QVOLN-QVOL)/QVOLN) .GT. 0.05 )   THEN
          QVOL = QVOLN
          GO  TO  2020
C
      ENDIF
C
 2030 RETURN
      END
```

TABLE A-11. LISTING OF FGMELT

```
C
C
      SUBROUTINE  FGMELT  ( NONR   , IRT    , ELVMR  , ELVAR  , DELTR ,
     +                      VELMOT )
C
C
C SUBCODE NAME:  FGMELT
C PURPOSE:  TO MEASURE THE LIQUID MATERIAL MOVEMENT WHEN DEBRIS BED
C            MELTING OCCURS.
C CALLING SUBROUTINES:  DBFRAG
C SUBROUTINES CALLED:  NONE
C WORK PACKAGE:  15
C ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C LAST MODIFICATION DATE:  11/30/81
C
C
C
C INPUT VARIABLES                    DESCRIPTION
C     NONR             NUMBER OF NODES USED IN RUBBLE BODY ANALYSIS
C     IRT              DEBRIS REGION TYPE: = 0, SOLID REGION
C                                          = 1, MOLTEN REGION
C
C     ELVMR            MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY
C     ELVAR            ELEVATION OF NODES IN DEBRIS
C     DELTR            TIME STEPS (S)
C
C OUTPUT VARIABLES                   DESCRIPTION
C     VELMOT           VELOCITY OF MOLTON POOL FLOWING DOWNWARD  (M/S)
C
C
C
      DIMENSION     ELVAR( NONR ) , IRT( NONR )
C
C
C SAVE INCOMING ELEVATION OF MOLTON MATERIAL
C
      ELVMO = ELVMR
C
      DO  100  I = 1 , NONR
          IF  ( IRT(I) .EQ. 1 )   GO  TO  110
  100 CONTINUE
C
  110 IF  ( I .LE. NONR )   ELVMR = ELVAR(I)
C
      VELMOT = AMAX1( 0.0 , (ELVMO - ELVMR)) / DELTR
C
      RETURN
      END
```

TABLE A-12. LISTING OF FGESTT

```
C
C
      SUBROUTINE  FGESTT  ( ELVMR  , ELVNR  , VELMOT , CURTIM , ESTDT )
C
C
C SUBCODE NAME:  FGESTT
C PURPOSE:  TO ESTIMATE THE TIME OF DISRUPTION
C CALLING SUBROUTINES:  DBFRAG
C SUBROUTINES CALLED:  NONE
C WORK PACKAGE:  15
C ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C LAST MODIFICATION DATE:  11/30/81
C
C
C INPUT VARIABLES                       DESCRIPTION
C      ELVMR           MOLTEN CORE REGION ELEVATION AT LOWER BOUNDARY
C      ELVNR           ELEVATION OF RUBBLE BODY FROM THE BOTTOM OF ROD
C                      BUNDLE TO BOTTOM OF RUBBLE BODY REGION  (M)
C      VELMOT          VELOCITY OF MOLTEN POOL FLOWING DOWNWARD (M/S)
C      CURTIM          CURRENT TIME  (S)
C
C OUTPUT VARIABLES                      DESCRIPTION
C      ESTDT           ESTIMATED TIME OF DISRUPTION  (S)
C
C
C
C
      TESTBB = ABS( ELVMR - ELVNR ) / AMAX1(1.0E-08 , VELMOT )
      ESTDT = CURTIM + TESTBB
C
      RETURN
      END
```

TABLE A-13. LISTING OF DRYOUT

```
      SUBROUTINE DRYOUT(VINC,POROSC,HITEC,EFFDIA,PDROPC,QVOL,
     +ROCOL,ROVAP,XMUCOL,XMUVAP,HFG,SRUTC,QDYOUT)
C
C     PROPOSE: TO CALCULATE DEVRIS BED DRYOUT HEAT FLUX BASED ON
C     LIPINSKI'S 1-D MODEL
C
C     CALLING SUBROUTINES: FROZTH,FRAGTH
C
C     SUBROUTINE CALLED: USMNMX OF IMSL
C
C     ENGINEER/PROGRAMMER: STH
C
C     LAST DATE MODIFIED: 9/10/81
C
C     INPUT VARIABLES:
C
C     VINC       COOLANT INLET VELOCITY(M/SEC), NOTE VINC IS THE
C                SUPERFICIAL VELOCITY OF THE FLUID MEASURED ON AN
C                EMPTY TUBE BASIS. THE TRUE COOLANT VELOCITY VINCT(URE)
C                =VINC/POROSC.
C
C     POROSC     DEBRIS BED POROSITY
C
C     HITEC      BED HEIGHT(M)
C
C     EFFDIA     EFFECTIVE PARTICLE DIAMETER OF A POROUS BODY,M
C
C     PDROPC     PRESSURE DROP ACROSS THE BED, PA
C
C     ROCOL      LIQUID COOLANT DENSITY(KG/M3)
C
C     ROVAP      VAPOR DENSITY(KG/M3)
C
C     XMUCOL     LIQUID COOLANT VISCOSITY(KG/SEC/M)
C
C     XMUVAP     VAPOR VISCOSITY(KG/SEC/M)
C
C     HFG        LATENT HEAT OF VAPORIZATION(J/KG)
C
C     SURTC      SURFACE TENSION OF COOLANT(PA)
C
C     QVOL       VOLUMETRIC HEAT GENERATION RATE(W/M3)
C
C     OUTPUT VARIABLES:
C
C     QDYOUT     DRYOUT HEAT FLUX, W/M2
C
      DIMENSION       SEFF(100)   , QX3(100)
C
C
      QGUES=QVOL*HITEC*(1.0-POROSC)
C
C     VARYING SEFF BETWEEN 0-1.0 TO GET A DRYOUT HEAT FLUX
C
      DO  100  I = 1 , 100
C
          ITR = 0
          SEFF(I) = 0.01 * FLOAT(I)
  105     XITEM1 = 1.0 / ROCOL
          VAPEG = ROCOL * VINC * HFG
C
          IF ( QGUES .LT. VAPEG )   XITEM1 = ( -1.0) * XITEM1
C
C
          XITEMA = 1.75 * ( 1.0 - POROSC ) / EFFDIA /
     +  ( AMAX1(1.0E-8,POROSC) ) ** 3 / HFG ** 2 * (1.0 / ROVAP /
     +  ( AMAX1(1.0E-8,(1.0 - SEFF(I))) ) **3 * SEFF(I)**3+XITEM1)
```

```
C
        XITEM2 = 180. * ( 1.0 - POROSC ) ** 2 / EFFDIA ** 2
    +       / ( AMAX1( 1.0E-8   POROSC ))
    +       ** 3 / HFG * ( XMUVAP / ROVAP
    +       / ( AMAX1 ( 1.0E-8 , ( 1.0 - SEFF(I))) ) ** 3
    +       * SEFF(I) ** 3 + XMUCOL / ROCOL )
C
        XITEM3 = 3.5 * (1.0 - POROSC) * ROCOL * VINC / EFFDIA
    +       /( AMAX1( 1.0E-8 , POROSC ) ) ** 3 / HFG * XITEM1
C
        XITEMB = XITEM2 - XITEM3
C
        XITEM4 = ( -1.0) * (ROCOL-ROVAP) * 9.80665 * SEFF(I) ** 3
    +     - 180. * ( 1.0 - POROSC ) ** 2 * XMUCOL * ROCOL * VINC
    +     / EFFDIA ** 2 / (AMAX1(1.0E-8 , POROSC)) ** 3 / ROCOL
C
        XITEMC = XITEM4 + 1.75 * (1.0-POROSC)*(ROCOL*VINC) ** 2
    +     / EFFDIA / ( AMAX1(1.0E-8 , POROSC) ) ** 3 * XITEM1
C
C    NOW SOLVE THE EQUATION  A * Q ** 2 + B * Q + C = 0  FOR  Q
C
        XITEM5 = XITEMB ** 2 - 4.0 * XITEMA * XITEMC
C
        IF  ( XITEM5 .LT. 0.0 )   THEN
C
C           WRITE  ( 6 , 200 )   SEFF(I)
C 200       FORMAT  (2X,31HNO REAL SOLUTION FOR Q AT SEFF=,F10.3)
C
            QX3(I) = -1.0
C
C    QX3=-1.0 MEANS THERE IS NO REAL SOLUTION FOR Q
C    SET QX3(I)=MASS FLUX*HFG
C
            QX3(I) = VINC * ROCOL * HFG
            GO  TO  100
C
        ELSE
            QX1 = ( -XITEMB + XITEM5 ** 0.5 ) / 2.0 / XITEMA
            QX2 = ( -XITEMB - XITEM5 ** 0.5 ) / 2.0 / XITEMA
            QX3(I) = AMAX1( 0.0 , QX1 , QX2 )
C
        ENDIF
C
        IF  ( ITR .GT. 30 )   THEN
C
C    WRITE(6,600)
C 600 FORMAT(2X,69HQX3 VALUE OSCILLATES AROUND VAPEG,SET QX3 VALUE AT TH
C   +E VALUE OF VAPEG)
C
            QX3(I) = VAPEG
            QGUES = QX3(I)
C
            GO  TO  100
C
        ELSE
            IF ( XITEM1 .LT. 0.0 )    THEN
C
C    CHECK IF QX3 STILL .LT. VAPEG, IF NOT GO BACK TO ADJUST XITEM1
C
                IF ( QX3(I) .GE. VAPEG )    THEN
                    QGUES = QX3(I)
                    ITR = ITR + 1
                    GO  TO  105
C
                ENDIF
C
            ELSE
```

TABLE A-13. (CONTINUED)

```
C
C      CHECK IF QX3 STILL .GE. VAPEG, IF NOT GO BACK TO ADJUST XITEM1
C
                  IF  ( QX3(I) .LT. VAPEG )   THEN
                      QGUES = QX3(I)
                      ITR = ITR + 1
                      GO  TO  105
C
                  ENDIF
C
              ENDIF
C
          ENDIF
C
          QGUES = QX3(I)
C
  100 CONTINUE
C
C      FOR MODEL TESTING, PRINT ALL SEFF AND QX3, AND FIND MAXIMUM QX3
C      WHICH CORRESPONDING TO THE DRYOUT HEAT FLUX AT GIVEN SEFF
C
C      WRITE(6,300)(SEFF(I),I=1,100)
C      WRITE(6,400)(QX3(I),I=1,100)
C 300  FORMAT(5X,5HSEFF=,10(/10X,10F12.4))
C 400  FORMAT(5X,5HQX3= ,10(/10X,10E12.5))
C
C      USE IMSL UTILITY SUBROUTINE USMNMX TO FIND THE LARGEST VALUE OF
C      QX3(=QDYOUT). LATER ON A SIMPLE CODING MAY BE USED TO REPLACE
C      IMSL SUBROUTINE USMNMX TO AVOID THE UNAUTHORIZED TRANSFER OF IMSL.
C
      NTST = 100
      INC = 1
C
      CALL   USMNMX ( QX3 , NTST , INC , XMIN , QDYOUT )
C
      WRITE  ( 6 , 500 )   QDYOUT
  500 FORMAT  ( 2X , 17HDRYOUT HEAT FLUX=,  E13.6 )
C
      RETURN
      END
```

TABLE A-14. LISTING OF REGMOD

```
      SUBROUTINE REGMOD(IDR,MAXL,ND,IDREGN,NOND,XLLIQ,XLSAT,XLVAP,
     +ELVNC,TSAT,HITEC,ELVAY,BEDTMP,COLTMP)
C
C     PURPOSE: TO MODIFY NODE ELEVATION, TEMPERATURE AND REGION ID NUMBE
C
C     CALLING SUBROUTINE: FROZTH, FRAGTH
C
C     SUBROUTINE CALLED: NONE
C
C     ENGINEER/PROGRAMMER:STH
C
C     LAST DATE MODIFIED: 8/10/81
C
C     INPUT VARIABLES:
C
C     IDR         REGION ID NUMBER CALCULATED IN FROZTH OR FRAGTH
C
C     MAXL        MAXIMUM NUMBER OF NODES IN THE ANALYSIS
C
C     ND          NUMBER OF NODES IN LIQUID OR VAPOR REGION ANALYSIS
C
C     IDREGN      REGION ID NUMBER OF LAST TIME STEP
C
C     NOND        NUMBER OF TOTAL NODES IN THE ANALYSIS
C
C     XLLIQ       LIQUID REGION LENGTH, M
C
C     XLSAT       SATURATION REGION LENGTH, M
C
C     XLVAP       VAPOR REGION LENGTH, M
C
C     ELVNC       ELEVATION OF DEBRIS BED-FROM BOTTOM OF ROD
C                 BUNDLE TO THE BOTTOM OF DEBRIS BED, M
C
C     TSAT        COOLANT SATURATION TEMPERATURE, K
C
C     HITEC       DEBRIS BED HEIGHT, M
C
C     INPUT/OUTPUT VARIABLES:
C
C     ELVAY       ELEVATION OF NOND NODES, M
C
C     BEDTMP      BED TEMPERATURE CORRESPONDING TO NOND NODES, K
C
C     COLTMP      COOLANT TEMPERATURE CORRESPONDING TO NOND NODES, K
C
      DIMENSION ELVAY(41) , BEDTMP(41) , COLTMP(41) , BEDTO(41) ,
     +          COLTO(41) , ELVO(41) , COEF(40,3)
C
      DO  100  I = 1 , NOND
C
          BEDTO( I ) = BEDTMP( I )
          COLTO( I ) = COLTMP( I )
          ELVO( I ) = ELVAY( I )
C
  100 CONTINUE
C
      IF ( IDR .LE. 2 .OR. IDR .GE. 7 )   THEN
          NDR = ND
C
          DO  200  I = 1 , NDR
C
               ELVAY(I) = ELVNC + HITEC / FLOAT(NDR-1) * (I-1)
C
  200     CONTINUE
```

```
C
            IF  ( ABS( ELVAY( NDR ) - ELVO( NOND )) .LE. 1.0E-10 ) THEN
                 ELVAY( NDR )  = ELVO( NOND )
C
            ENDIF
C
            CALL  ICSCCU  ( ELVO , COLTO , NOND , COEF , 40 , IER )
C
C      WRITE  ( 6 , 1000 ) IER
C
            CALL  ICSEVU  ( ELVO , COLTO , NOND , COEF , 40 , ELVAY ,
     +                      COLTMP , NDR , IER )
C
C      WRITE  ( 6 , 1000 ) IER
C
            CALL ICSCCU  ( ELVO , BEDTO , NOND , COEF , 40 , IER )
C
C      WRITE  ( 6 , 1000 ) IER
C
            CALL ICSEVU  ( ELVO , BEDTO , NOND , COEF , 40 , ELVAY ,
     +                     BEDTMP , NDR , IER )
C
C      WRITE  ( 6 , 10000 ) IER
C
C
C
            NOND = NDR
            IDREGN = IDR
C
       ELSE IF  ( IDR .EQ. 3 )   THEN
            NDR = 3
C
            DO  300  I = 1 , NDR
C
                 ELVAY(I) = ELVNC + HITEC / 2.0 * (I-1)
                 COLTMP(I) = TSAT
C
  300       CONTINUE
C
            IF  ( ABS( ELVAY( NDR ) - ELVO( NOND )) .LE. 1.0E-10 ) THEN
C
                 ELVAY( NDR ) = ELVO( NOND )
C
            ENDIF
C
C
            CALL ICSCCU  ( ELVO , BEDTO , NOND , COEF , 40 , IER )
C
C      WRITE  ( 6 , 1000 )  IER
C
            CALL ICSEVU  ( ELVO , BEDTO , NOND , COEF , 40 , ELVAY ,
     +                     BEDTMP , NDR , IER )
C
C      WRITE  ( 6 , 1000 )  IER
C
C
            NUND = NDR
            IDREGN = IDR
C
       ELSE IF ( IDR .EQ. 4 )  THEN
            NDR = ND + 2
C
            DO  400  I = 1 , ND
C
                 ELVAY(I) = ELVNC + XLLIQ / FLOAT(ND-1) * (I-1)
                 COLTMP(I) = COLTO(1)+(TSAT-COLTO(1))/FLOAT(ND-1)*(I-1)
C
  400       CONTINUE
```

TABLE A-14. (CONTINUED)

```
C
          ELVAY(ND+1) = ELVNC + (HITEC + XLLIQ) / 2.0
          ELVAY(ND+2) = ELVNC + HITEC
C
          COLTMP(ND+1) = TSAT
          COLTMP(ND+2) = TSAT
C
          IF  ( ABS( ELVAY( NDR ) - ELVO( NOND )) .LE. 1.0E-10 ) THEN
C
               ELVAY( NDR ) = ELVO( NOND )
C
          ENDIF
C
          CALL  ICSCCU  ( ELVO , BEDTO , NOND , COEF , 40 , IER )
C
C     WRITE  ( 6 , 1000 )  IER
C
          CALL  ICSEVU  ( ELVO , BEDTO , NOND , COEF , 40 , ELVAY ,
     +                    BEDTMP , NDR , IER )
C
C     WRITE  ( 6 , 1000 )  IER
C
C
          NOND = NDR
          IDREGN = IDR
C
      ELSE  IF  ( IDR .EQ. 5 )    THEN
          NDR = ND + 2
C
C
          COLTMP(1) = TSAT
          COLTMP(2) = TSAT
C
          ELVAY(1) = ELVNC
          ELVAY(2) = ELVNC + XLSAT / 2.0
C
          DO  500  I = 3 , NDR
C
              COLTMP(I) = TSAT+(COLTO(NOND)-TSAT)/FLOAT(ND-1)*(I-3)
              ELVAY(I) = ELVNC+XLSAT+(HITEC-XLSAT)/FLOAT(ND-1)*(I-3)
C
  500     CONTINUE
C
          IF  ( ABS( ELVAY( NDR ) - ELVO( NOND )) .LE. 1.0E-10 )  THEN
C
              ELVAY( NDR ) = ELVO( NOND )
C
          ENDIF
C
          CALL  ICSCCU  ( ELVO , BEDTO , NOND , COEF , 40 , IER )
C
C      WRITE  ( 6 , 1000 )  IER
C
          CALL ICSEVU ( ELVO , BEDTO , NOND , COEF , 40 , ELVAY ,
     +                  BEDTMP , NDR , IER )
C
C      WRITE  ( 6 , 1000 )  IER
C
          NOND = NDR
          IDREGN = IDR
C
       ELSE  IF ( IDR .EQ. 6 )   THEN
C
C
```

TABLE A-14. (CONTINUED)

```
          NDR = 2 * ND + 1
C
          DO  800  I = 1 , ND
C
              COLTMP(I) = COLTO(1)+(TSAT-COLTO(1))/FLOAT(ND-1)*(I-1)
              ELVAY(I) = ELVNC + XLLIQ / FLOAT(ND - 1) * (I - 1)
C
  800     CONTINUE
C
          COLTMP(ND+1) = TSAT
          ELVAY(ND+1) = ELVNC + XLLIQ + XLSAT / 2.0
C
          DO  900  I = 1 , ND
C
              COLTMP(ND+1+I) = TSAT + (COLTO(NOND) - TSAT)
     +                  / FLOAT (ND - 1) * (I - 1)
              ELVAY(ND+1+I) = ELVNC+XLLIQ+XLSAT+(HITEC-XLLIQ-XLSAT)
     +                  / FLOAT (ND - 1) * (I - 1)
C
  900     CONTINUE
C
          IF  ( ABS( ELVAY( NDR ) - ELVO( NOND )) .LE. 1.0E-10 )  THEN
C
              ELVAY( NDR ) = ELVO( NOND )
C
          ENDIF
C
          CALL  ICSCCU  ( ELVO , BEDTO , NOND , COEF , 40 , IER )
C
C     WRITE  ( 6 , 910 )  (ELVO(I) , I = 1 , NOND )
C     WRITE  ( 6 , 920 )  (BEDTO(I) , I = 1 , NOND )
C
  910 FORMAT  ( 2X , 5HELVO=, 5(/2X , 10F11.6))
  920 FORMAT  ( 2X , 6HBEDTO=, 5(/2X , 10F11.3))
C
C     WRITE  ( 6 , 1000 )  IER
C
 1000 FORMAT  ( 2X , 4HIER= , I2 )
C
          CALL  ICSEVU ( ELVO , BEDTO , NOND , COEF , 40 , ELVAY ,
     +                   BEDTMP , NDR , IER )
C
C     WRITE  ( 6 , 930 )  ( ELVAY(I) , I = 1 , NDR )
C     WRITE  ( 6 , 940 )  ( BEDTMP (I) , I = 1 , NDR )
C
  930 FORMAT ( 2X , 6HELVAY= , 5(/2X , 10F11.6))
  940 FORMAT ( 2X , 7HBEDTMP= , 5(/2X , 10F11.3 ))
C
C     WRITE  ( 6 , 1000 )  IER
C
C
C
          NOND = NDR
          IDREGN = IDR
      ENDIF
      RETURN
      END
```

TABLE A-15. LISTING OF TEMPSR

```
      SUBROUTINE  TEMPSR  ( N      , N2     , IDYCT  , TEMPCO , TCOLCO ,
     +                      HVSC   , HTLBCD , HTLBCC , HTUBCD , HTUBCC ,
     +                      POROSC , VINC   , QVOL   , DELTT  , DELTZ  ,
     +                      TEMPC  , TCOLC  )
C
C
C
C     PURPOSE: TO CALCULATE COOLANT AND BED TEMPERATURE ALONG Z
C
C     CALLING SUBROUTINE:FROZTH
C
C     SUBROUTINE CALLED: MINERVA
C
C     ENGINEER/PROGRAMMER:STH
C
C     LAST DATE MODIFIED:8/10/81
C
C     INPUT VARIABLES:
C
C     N           NUMBER OF AXIAL NODES
C
C     N2          2*N FOR SOLVING COOLANT AND BED TEMPERATURE
C                 SIMULTANEOUSLY
C
C     IDYOT       INDICATER OF COOLANT STATE, =0 LIQUID, =1 VAPOR
C
C     TEMPCO      BED TEMPERATURES AT N NODES AT PREVIOUS TIME STEP(K)
C
C     TCOLCO      COOLANT TEMPERATURE AT N NODES AT PREVIOUS TIME STEP(K)
C
C     HVSC        HEAT TRANSFER COEFFICIENT BETWEEN BED AND COOLANT,
C                 VOLUMETRIC,W/K/M3
C
C     HTLBCD      HEAT TRANSFER INTO DEBRIS BED AT LOWER BOUNDARY,W/M2
C
C     HTLBCC      HEAT TRANSFER INTO DEBRIS COOLANT AT LOWER BOUNDARY,W/M2
C
C     HTUBCD      HEAT TRANSFER INTO DEBRIS BED AT UPPER BOUNDARY,W/M2
C
C     HTUBCC      HEAT TRANSFER INTO DEBRIS COOLANT AT UPPER BOUNDARY,W/M2
C
C     POROSC      DEBRIS BED POROSITY
C
C     VINC        COOLANT INLET VELOCITY, M/SEC, NOTE THAT VINC IS THE
C                 SUPERFICIAL VELOCITY OF THE FLUID MESURED ON AN
C                 EMPTY TUBE BASIS. IN TEMPSR CALCULATION, COOLANT
C                 VELOCITY IS THE TRUE VELOCITY IN COOLANT CHANNELS,
C                 THEREFORE, VINCT(URE)=VINC/POROSC. THIS ADJUSTMENT
C                 IS CALCULATED IN THE FIRST FORTRAN STATEMENT BELOW
C                 DIMENSION STATEMENT IN TEMPSR.
C
C     QVOL        VOLUMETRIC HEAT GENERATION RATE, W/M3(SOLID)
C
C     DELTT       TIME STEP(S)
C
C     DELTZ       DISTANCE BETWEEN AXIAL NODE,M
C
C
C
C
C
C
```

TABLE A-15. (CONTINUED)

```
C        OUTPUT VARIABLES:
C
C        TEMPC       BED TEMPERATURE, K
C
C        TCOLC       COOLANT TEMPERATURE, K
C
C
C
C
C
C
C
C        AT THIS TIME, ALL MATERIALS PROPERTIES ARE ASSUMED COMING THROUGH
C        COMMON BLOCK(S), THESE INCLUDE:
C
C        ROCOL       LIQUID COOLANT DENSITY, KG/M3
C
C        ROVAP       VAPOR DENSITY, KG/M3
C
C        CPCOL       LIQUID SPECIFIC HEAT, J/K/KG
C
C        CPVAP       VAPOR SPECIFIC HEAT, J/K/KG
C
C        XKCOL       LIQUID COOLANT THERMAL CONDUCTIVITY, W/M/K
C
C        XKVAP       VAPOR(STEAM) THERMAL CONDUCTIVITY, W/M/K
C
C        RODEB       DEBRIS BED DENSITY, KG/M3
C
C        CPDEB       DEBRIS BED SPECIFIC HEAT, J/K/KG
C
C        XKDEB       DEBRIS BED THERMAL CONDUCTIVITY, W/M/K
C
C        ASSUME MATERIALS PROPERTIES ARE TEMPERATURE DEPENDENT, THESE
C        INCLUDE: COOLANT DENSITY, SPECIFIC HEAT, THERMAL CONDUCTIVITY
C        AND DEBRIS BED DENSITY, SPECIFIC HEAT, AND THERMAL CONDUCTIVITY.
C        TEMPERATURES ARE BASED ON OLD TIME STEP VALUES
C        ASSUME ROMX(N2), CPMX(N2), XKMX(N2) ARE KNOWN, FIRST N NUMBERS
C        ARE FOR DEBRIS BED, THE FOLLOWING N VALUES ARE FOR COOLANT.
C
C
         COMMON      ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +               XMUCD  , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +               SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +               ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +               XKMX   , XMUF3  , RODEB
C
C
         DIMENSION  TEMPCO(20) , TCOLCO(20) , TEMPC(20)  , TCOLC(20)  ,
     +              XMX(40)    , BMX(40)    , ROMX(40)   , CPMX(40)   ,
     +              XKMX(40)   , Z(40)      , IPVT(40)   , DMY(19)    ,
     +              AMX(40,40)
C
C
C
C
C
C   ADJUST COOLANT VELOCITY FROM SUPERFICIAL VELOCITY TO THE TRUE
C   COOLANT VELOCITY IN COOLANT CHANNELS
C
         VINCT = VINC / AMAX1( 1.0E-8 , POROSC )
C
         DO  150  I = 1 , N2
C
         DO  100  J = 1 , N2
               AMX ( I,J ) = 0.0
  100      CONTINUE
```

```
C
  150 CONTINUE
C
      DIGUR = HVSC / 2.0 / ( 1.0 - POROSC )
C
      AMX(1,1) = ROMX(1) * CPMX(1) / DELTT + (XKMX(1) + XKMX(2))
     +         / 2.0 / DELTZ ** 2 + DIGUR
C
      AMX(1,2) = -( XKMX(1) + XKMX(2) ) / 2.0 / DELTZ ** 2
      AMX(1,N+1) = -DIGUR
      AMX(N,N-1) = -( XKMX(N) + XKMX(N-1) ) / 2.0 / DELTZ ** 2
C
      AMX(N,N) = ROMX(N) * CPMX(N) / DELTT + (XKMX(N) + XKMX(N-1))
     +   / 2.0 / DELTZ ** 2 + DIGUR
C
      AMX(N,N2) = -DIGUR
C
      DIGLL = HVSC / 2.0 / POROSC
C
      AMX(N+1,1) = -DIGLL
C
      AMX(N+1 , N+1) = ROMX(N + 1) * CPMX(N + 1) / DELTT
     +       + (XKMX(N + 1) + XKMX(N + 2)) / 2.0 / DELTZ ** 2 + DIGLL
     +       - ROMX(N + 1) * VINCT * CPMX(N + 1) / 2.0 / DELTZ
C
      AMX(N+1 , N+2) = -(XKMX(N + 1) + XKMX(N + 2)) / 2.0 / DELTZ ** 2
     +       + ROMX(N + 1) * VINCT * CPMX(N + 1) / 2.0 / DELTZ
C
      AMX(N2 , N2-1) = -(XKMX(N2 - 1) + XKMX(N2)) / 2.0
     +       / DELTZ ** 2 - ROMX(N2) * VINCT * CPMX(N2) / 2.0 / DELTZ
C
      AMX(N2 , N2) = ROMX(N2) * CPMX(N2) / DELTT
     +       + ( XKMX(N2) + XKMX (N2 - 1)) / 2.0 / DELTZ ** 2 + DIGLL
     +       + ROMX (N2) * CPMX(N2) * VINCT / 2.0 / DELTZ
C
      AMX(N2,N) = - DIGLL
C
      DO  200  I = 2 , N - 1
C
          XKUP = (XKMX(I + 1) + XKMX(I)) / 2.0 / 2.0 / DELTZ ** 2
          XKDN = (XKMX(I) + XKMX(I - 1)) / 2.0 / 2.0 / DELTZ ** 2
C
          AMX(I , I-1) = - XKDN
          AMX(I , I) = ROMX(I) * CPMX(I) / DELTT + XKUP + XKDN + DIGUR
          AMX(I , I+1) = - XKUP
          AMX(I , N+I) = - DIGUR
C
  200 CONTINUE
C
      DO  300  I = N + 2 , N2 - 1
C
          XKUP1 = (XKMX(I+1) + XKMX(I)) / 2.0 / 2.0 / DELTZ ** 2
          XKDN1 = (XKMX(I) + XKMX(I-1)) / 2.0 / 2.0 / DELTZ ** 2
C
          GCDZ = ROMX(I) * VINCT * CPMX(I) / 4.0 / DELTZ
C
          AMX(I , I-1) = - XKDN1 - GCDZ
          AMX(I , I) = ROMX(I) * CPMX(I) / DELTT + XKUP1 + XKDN1 + DIGLL
          AMX(I,I+1) = -XKUP1+GCDZ
          AMX(I , I-N) = - DIGLL
C
  300 CONTINUE
C
      DO  400  I = 1 , N
C
          XMX(I) = TEMPCO(I)
```

TABLE A-16. LISTING OF DBDRIV

```
C
  400 CONTINUE
C
      DO  500  I = N + 1 , N2
C
          XMX(I) = TCOLCO(I-N)
C
  500 CONTINUE
C
      DO  600  I = 1 , N2
          BMX(I) = 0.0
C
          DO  700  J = 1 , N2
              CST = AMX(I,J) * XMX(J)
              BMX(I) = BMX(I) + CST
  700     CONTINUE
C
  600 CONTINUE
C
      BMX(1) = BMX(1) + QVOL + 2.0 * HTLBCD / DELTZ
      BMX(N) = BMX(N) + QVOL + 2.0 * HTUBCD / DELTZ
      BMX(N+1) = BMX(N+1) + 2.0 * HTLBCC / DELTZ
      BMX(N2) = BMX(N2) + 2.0 * HTUBCC / DELTZ
C
      DO  800  I = 2 , N - 1
C
          BMX(I) = BMX(I) + QVOL
C
  800 CONTINUE
C
C     ADJUST TEMPSR FOR LOW QVOL AND LOW HTUBC AND HTLBC SITUATIONS.
C
      IF  ( QVOL  .LT.  1.0E-05
     +           .AND.  HTUBCD .LT. 1.0E-05
     +                .AND.  HTLBCD .LT. 1.0E-05 )   THEN
C
          DO  750  I = 1 , N2
C
              IF ( I .LE. N )    THEN
                  XADJ = DIGUR
                  AMX(I,I+N) = AMX(I,I+N) - DIGUR
C
              ELSE
                  XADJ = DIGLL
                  AMX(I,I-N) = AMX(I,I-N) - DIGLL
C
              ENDIF
C
              AMX(I,I) = AMX(I,I) + XADJ
              BMX(I) = ROMX(I) * CPMX(I) * XMX(I) / DELIT
C
  750     CONTINUE
C
      ENDIF
C
C     NOW SOLVE AMX(N2,N2)*XMX(N2)=BMX(N2) FOR XMX(N2)
C
      CALL  LSGECO   ( AMX , N2 , N2 , IPVT , RCOND , Z )
      CALL   LSGESL  ( AMX , N2 , N2 , IPVT , BMX , 0 )
C
      DO 900 I=1,N
C
          TEMPC(I) = BMX(I)
          TCOLC(I) = BMX(N+I)
C
  900 CONTINUE
C
      RETURN
      END
```

TABLE A-16. (CONTINUED)

```
*
      PROGRAM DBDRIV
C
C
C SUBCODE NAME:  DBDRIV
C PURPOSE:  DRIVER FOR DBFRAG, DBFROZ AND ASSOCIATED ROUTINES WHICH
C           SIMULATES DBUNDL IN TESTING AND INITIAL INTEGRATION PHASES
C CALLING SUBROUTINES:  NONE
C SUBROUTINES CALLED:  DEBINP , DBTIME , DBFRAG , DBFROZ , DBOUTD
C WORK PACKAGE:  15
C ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C LAST MODIFICATION DATE:  11/30/81
C
C
C IN DBUNDL SUBROUTINE, ALL OF THE FOLLOWING VARIABLES WILL BE IN
C ADJUSTABLE ARRAYS, WITH DIMENSIONS VARYING WITH THE NUMBER OF NODES
C AND DEBRIS REGIONS.
C
C
C
C
      IMPLICIT  INTEGER  ( I - N )
C
C
      DIMENSION       ELVAR  ( 41 ) , BEDTMR ( 41 ) , COLTMR ( 41 ) ,
     +                IRT    ( 41 ) , ELVAY  ( 41 ) , BEDTMP ( 41 ) ,
     +                COLTMP ( 41 ) , IMT    ( 41 ) , ROMX   ( 40 ) ,
     +                CPMX   ( 40 ) , XMUX   ( 40 ) , XKMX   ( 40 ) ,
     +                T      ( 20 ) , FRAG   ( 20 ) , FROZEN ( 20 ) ,
     +                RBDRPT ( 20 ) , ESTDT  ( 20 ) , CURTIM ( 20 )
C
C
      COMMON      ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +            XMUCD  , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +            SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +            ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +            XKMX   , XMUF3  , RODEB
C
C
      LOGICAL         FRAG , FROZEN , RBDRPT
C
C
C
C
C CALL *DEBINP* WHICH WILL DEFINE I/O VARIABLES NEEDED IN TESTING
C
      CALL  DEBINP
     +                   ( SIZAVR , XMASSR , ZRMR   , ZRO2MR , UO2MR  ,
     +                     STRMR  , ABSMR  , AREAR  , HITRG  , VINROD ,
     +                     VINR   , ZRO2TR , ZRO2AR , ALPHTR , ALPHAR ,
     +                     ELVNR  , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                     MAXL   , IDREGR , NONR   , IRT    , ELVAR  ,
     +                     COLTMR , BEDTMR , OXDAR  , FISHR  , ELVMR  ,
     +                     XMASSC , ZRMC   , ZRO2MC , UO2MC  , STRMC  ,
```

```
     +                    ABSMC  , HITEC  , VINC   , VINCOL , ZRO2TC ,
     +                    ZRO2AC , ALPHTC , ALPHAC , ELVNC  , HTLBCD ,
     +                    HTLBCC , HTUBCD , HTUBCC , IDREGN , NOND   ,
     +                    ELVAY  , BEDTMP , COLTMP , IMT    , OXDAC  ,
     +                    FISHC  , ELVMC  , ND     , N      , NUMREG ,
     +                    FRAG   , FROZEN , RBDRPT , CURTIM , T        )
C
C
C DETERMINE DISRUPTED TIME ARRAY
C
      CALL  DBTIME  ( T , N )
C
      I  = 1
C
   10 CONTINUE
C
C
C IF I IS LESS THAN N, THEN ITERATE AGAIN INCREASING TIME STEP
C
      IF  ( I .LE. N )   THEN
C
C DETERMINE CURRENT TIME STEP
C
          DELTT = T ( I + 1 ) - T ( I )
C
C ITERATE ON K UNTIL ALL REGION'S BEHAVIOR HAVE BEEN DESCRIBED
C
          DO  100  K = 1 , NUMREG
C
C COMPUTE FRAGMENTED DEBRIS BEHAVIOR FOR REGION K IF FRAG(K) = TRUE
C
C
              IF  ( FRAG (K) )   THEN
C
C
C MAKE THE CALL TO DBFRAG, THE DRIVER WHICH WILL HANDLE THE
C RUBBLE DEBRIS BEHAVIOR
C
C
                  CALL  DBFRAG
     +              ( SIZAVR , XMASSR , ZRMR   , ZRO2MR , UO2MR  ,
     +                STRMR  , ABSMR  , AREAR  , HITRG  , VINROD ,
     +                VINR   , ZRO2TR , ZRO2AR , ALPHTR , ALPHAR ,
     +                ELVNR  , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                ND     , MAXL   , DELTT  , IDREGR , NONR   ,
     +                IRT    , ELVAR  , COLTMR , BEDTMR ,THMOTR  ,
     +                ELVMR  , OXDAR  , FISHR  , HITER  , POROSR ,
     +                POROPR , XCHRL  , KFLUID , KTERM  , DOWNR  ,
     +                VELMOT , T      , N      , I      ,
     +                RBDRPT (K) , CURTIM (K) , ESTDT (K) )
C
C
      PRINT  * , ' '
```

TABLE A-16. (CONTINUED)

```
     +                          YCHRL  , EFFDIA , KTERMC , IDREGN , NOND   ,
     +                          ELVAY  , BEDTMP , COLTMP , IMT    , THMDTC ,
     +                          ELVMC  , DOWNZ  , VELMTC , CURTIM( K ) ,
     +                          DELTT  , FRAG( K ) , FROZEN( K ) )
C
  100       CONTINUE
C
C
            I = I + 1
C
            GO  TO  10
C
C
        ENDIF
C
C
        END
```

TABLE A-16. (CONTINUED)

```
      PRINT  * , 'CURTIM(K)=',CURTIM(K),'DELTT=',DELTT,'IDREGR=',IDREGR,
     + 'NONR=',NONR,'ELVAR=',ELVAR,'COLTMR ARRAY=',
     +     (COLTMR(NN),NN=1,NONR)
     + ,'BEDTMR ARRAY=',(BEDTMR(NN),NN=1,NONR),'THMOTR=' ,
     + THMOTR,'OXDAR=',OXDAR,'FISHR=',FISHR,'HITER=',
     + HITER,'POROSR=',POROSR,'PDROPR=',PDROPR,'XCHRL=',XCHRL,'KFLUID=',
     + KFLUID,'KTERM=',KTERM,'DOWNR=',DOWNR,'VELMOT=',VELMOT,
     + 'T(K)=',T(K),'T(K+1)=',T(K+1),'RBDRPT(K)=',RBDRPT(K) ,
     + 'ESTDT(K)=',ESTDT(K)
C
C
C
C COMPUTE FROZEN DEBRIS BEHAVIOR FOR REGION K IF FROZEN (K) = TRUE
C
C
               ELSE  IF  ( FROZEN (K) )   THEN
C
C
C
C CALL THE DRIVER DBFROZ WHICH WILL CONTROL THE FROZEN BEHAVIOR
C ROUTINES
C
C
                    CALL  DBFROZ
     +                 ( XMASSC , ZRMC   , ZRO2MC , UO2MC  , STRMC  ,
     +                   ABSMC  , AREAR  , HITEC  , VINC   , VINCOD ,
     +                   ZRO2TC , ZRO2AC , ALPHTC , ALPHAC , ELVNC  ,
     +                   HTLBCD , HTLBCC , HTUBCD , HTUBCC , ND     ,
     +                   MAXL   , DELTT  , POROSC , PDROPC ,
     +                   YCHRL  , KTERMC , EFFDIA , IDREGN , NOND   ,
     +                   ELVAY  , BEDTMP , COLTMP , THMOTC , ELVMC  ,
     +                   OXDAC  , FISHC  , HYDGC  , FGRSC  , VOLGSC ,
     +                   IMT    , DOWNZ  , VELMTC , T      ,
     +                   I      , N      , RBDRPT (K) , CURTIM (K) ,
     +                   ESTDT (K) )
C
C
C
C  WRITE OUT ALL VARIABLES LISTED IN THE ARGUMENTS)
C
       PRINT  * , DELTT , POROSC , PDROPC , YCHRL , KTERMC , EFFDIA ,
     +            IDREGN , NOND , ELVAY , BEDTMP , COLTMP , THMOTC ,
     +            ELVMC , OXDAC , FISHC , HYDGC , FGRSC , VOLGSC ,
     +            IMT , DOWNZ , VELMTC , T (I) , T (I+1) ,
     +            I , N , RBDRPT (K) , CURTIM (K) , ESTDT (K)
C
C
               ENDIF
C
               CALL   DBOUTD
     +                   ( HITER  , POROSR , PDROPR , XCHRL  , KFLUID ,
     +                     KTERM  , IDREGR , NONR   , ELVAR  , BEDTMR ,
     +                     COLTMR , IRT    , THMOTR , DOWNR  , VELMOT ,
     +                     ELVMR  , ESTDT(K) , HITEC  , POROSC , PDROPC ,
```

TABLE A-17. LISTING OF DEBINP

```
*
      SUBROUTINE DEBINP
     +                       ( SIZAVR , XMASSR , ZRMR   , ZRO2MR , UO2MR  ,
     +                         STRMR  , ABSMR  , AREAR  , HITRG  , VINROD ,
     +                         VINR   , ZRO2TR , ZRO2AR , ALPHTR , ALPHAR ,
     +                         ELVNR  , HTLBRD , HTLBRC , HTUBRD , HTUBRC ,
     +                         MAXL   , IDREGR , NONR   , IRT    , ELVAR  ,
     +                         COLTMR , BEDTMR , OXDAR  , FISHR  , ELVMR  ,
     +                         XMASSC , ZRMC   , ZRO2MC , UO2MC  , STRMC  ,
     +                         ABSMC  , HITEC  , VINC   , VINCOD , ZRO2TC ,
     +                         ZRO2AC , ALPHTC , ALPHAC , ELVNC  , HTLBCD ,
     +                         HTLBCC , HTUBCD , HTUBCC , IDREGN , NOND   ,
     +                         ELVAY  , BEDTMP , COLTMP , IMT    , OXDAC  ,
     +                         FISHC  , ELVMC  , ND     , N      , NUMREG ,
     +                         FRAG   , FROZEN , RBDRPT , CURTIM , T      )
C
C
C  SUBCODE NAME:  DEBINP
C  PURPOSE:  TEMPORARY INPUT SUBROUTINE FOR DBFRAG AND DBFROZ WHICH
C            SIMULATES OUTPUT FROM DBTRAN, DBREGN AND MATPRO ROUINES.
C  CALLING SUBROUTINES:  MODO
C  SUBROUTINES CALLED:  NONE
C  WORK PACKAGE:  15
C  ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C  LAST MODIFICATION DATE:  11/30/81
C
C
C
C  THIS SUBROUTINE INITIALIZES THE INPUT DATA NEEDED BY THE DEBRIS
C  BEHAVIOR MODEL DRIVERS *DBFRAG* AND *DBFROZ*.  IN SCDAP MODO, THIS
C  DATA WILL COME FROM INITIAL INPUT, DBTRAN, DBREGN, AND MATPRO SUB-
C  ROUTINES. IN THIS VERSION OF *DEBINP*, THE DATA IS THAT USED FOR
C  THE ACCEPTANCE TESTING FOR THE BEHAVIOR MODELS AND IS BASED ON TMI-2
C  FUEL BUNDLE GEOMETRY.
C
C
C
C  IN *DEBINP* ALL VARIABLES WITH THE EXCEPTION OF THE REGION FLAGS ARE
C  SINGALLY DIMENSIONED WITH RESPECT TO THE NUMBER OF DEBRIS REGIONS FOR
C  TESTING PURPOSES ONLY.
C
C
C
C
C  *****INPUT AND OUTPUT VARIABLES AND THEIR DESCRIPTIONS******
C
C  SEE THE DEBRIS BEHAVIOR MODELS FINAL DESIGN REPORT BY S. T. HSIEH FOR
C  COMPLETE VARIABLE DESCRIPTIONS.
C
C
C
C
      IMPLICIT  INTEGER  ( I - N )
C
C
      DIMENSION        ELVAR  ( 41 ) , BEDTMR ( 41 ) , COLTMR ( 41 ) ,
     +                 IRT    ( 41 ) , ELVAY  ( 41 ) , BEDTMP ( 41 ) ,
     +                 COLTMP ( 41 ) , IMT    ( 41 ) , ROMX   ( 40 ) ,
```

TABLE A-17. (CONTINUED)

```
     +              CPMX   ( 40 ) , XMUX   ( 40 ) , XKMX   ( 40 ) ,
     +              T      ( 20 ) , FRAG   ( 20 ) , FROZEN ( 20 ) ,
     +              RBDRPT ( 20 ) , CURTIM ( 20 )
C
C
      COMMON        ROZR   , ROZRO2 , ROUO2  , ROSTR  , ROABS  , ROCOLD ,
     +              XMUCD  , ROCOL  , XMUCOL , ROVAP  , XMUVAP , HFG    ,
     +              SURTC  , CPCOL  , XKCOL  , XKVAP  , CPDEB  , TSAT   ,
     +              ROSAT  , TMELT  , XLATC  , ROMX   , CPMX   , XMUX   ,
     +              XKMX   , XMUF3  , RODEB
C
C
      LOGICAL         FRAG , FROZEN , RBDRPT
C
C
C
C
C
C DEFINE THE NUMBER OF NODES, REGIONS, AND TIMESTEPS.
C
      ND     = 20
      NUMREG = 2
      N      = 10
C
C INITIALIZE REGION DEBRIS TYPE FLAGS
C
C
      FRAG ( 1 )   = .TRUE.
      FRAG ( 2 )   = .FALSE.
C
      FROZEN ( 1 ) = .FALSE.
      FROZEN ( 2 ) = .TRUE.
C
C
C INITIALIZE DISRUPTION FLAGS FOR ALL REGIONS
C
      DO  5  K = 1 , NUMREG
C
          RBDRPT( K ) = .FALSE.
C
    5 CONTINUE
C
C
C DEFINE MATERIAL PROPERTIES FOUND IN COMMON BLOCK
C
      ROZR   = 6.552E+03
      ROZRO2 = 5.82E+03
      ROUO2  = 1.097E+04
      ROSTR  = 8.0E+03
      ROABS  = 8.0E+03
      ROCOL  = 488.418
      XMUCOL = 8.68E-05
      ROVAP  = 160.20
      XMUVAP = 1.255E-05
```

```
      HFG     = 1.661E+06
      SURTC   = 54.3E-03
      CPCOL   = 3.683E+04
      XKCOL   = 3.209E-01
      XKVAP   = 5.193E-02
      CPDEB   = 580.0
      TSAT    = 700.0
      ROSAT   = 480.418
      TMELT   = 3150.0
      XLATC   = 2.74E+05
      XMUF3   = 8.62E-04
      RODEB   = 9.85E+03
C
C
      DO  20  I = 1 , ND
C
          ROMX (I)          = 9.85E+03
          ROMX ( I  +  ND ) = ROCOL
C
          CPMX ( I )        = CPDEB
          CPMX ( I  +  ND ) = CPCOL
C
          XMUX ( I )        = 4.24E-03
          XMUX ( I  +  ND ) = XMUCOL
C
          XKMX ( I )        = 1.904
          XKMX ( I  +  ND ) = XKCOL
C
   20 CONTINUE
C
C
C  DEFINE DATA THAT WILL BE INPUT BY DUMMY ARGUMENTS.  THIS GROUP
C  CONTAINS INPUT FOR FGCHAR, A FEW OF WHICH ARE ALSO INPUT TO FROZTH.
C
C
      SIZAVR = 3.0E-04
      XMASSR = 34.642
      ZRMR   = 5.268
      ZRO2MR = 0.375
      UO2MR  = 29.0
      STRMR  = 0.0
      ABSMR  = 0.0
      AREAR  = 3.664E-02
      HITRG  = 0.2
      VINR   = 0.10
      VINROD = 0.10
C
C  THIS GROUP OF VARIABLE DEFINITIONS CONTAINS INPUT FOR FRAGTH
C
      ZRO2TR = 0.5E-06
      ZRO2AR = 0.5E-10
      ALPHTR = 0.5E-06
      ALPHAR = 0.5E-10
```

TABLE A-17. (CONTINUED)

```
      ELVNR  = 2.5
      ELVMR  = ELVNR + HITRG
      HTLBRD = 0.0
      HTLBRC = 0.0
      HTUBRD = 0.0
      HTUBRC = 0.0
      MAXL   = 41
C
C THESE OUTPUT VARIABLES MUST ALSO BE INITIALIZED
C
      IDREGR = 1
      NONR   = 20
C
C
C INITIALIZE ELEMENTS 1 TO NONR OF THESE ARRAYS
C
      DIV    = FLOAT ( ND - 1 )
C
      DO  40  I = 1 , NONR
C
          FACTOR = ( I - 1 ) / DIV
C
          IRT ( I )    = 0
          ELVAR  ( I ) = ELVNR + 0.2 * FACTOR
          COLTMR ( I ) = 400.0 + 190.0 * FACTOR
          BEDTMR ( I ) = COLTMR ( I )
C
   40 CONTINUE
C
      OXDAR  = 1.69E+08
      FISHR  = 1.69E+08
C
C
C
C
C
C NOW SET UP THE INPUT VALUES FOR FROZEN BEHAVIOR
C
C
      HITEC  = 0.5
      VINC   = 0.001
      ZRO2TC = 0.5E-06
      ZRO2AC = 0.5E-10
      ALPHTC = 0.5E-06
      ALPHAC = 0.5E-10
      ELVNC  = 2.5
      ELVMC  = ELVNC + HITEC
      OXDAC  = 1.69E+08
      FISHC  = 1.69E+08
      HTLBCD = 0.0
      HTLBCC = 0.0
      HTUBCD = 0.0
      HTUBCC = 0.0
```

TABLE A-17. (CONTINUED)

```
      IDREGN = 5
      NCND   = 22
C
C INITIALIZE FIRST 2 ELEMENTS OF FOLLOWING OUTPUT ARRAYS
C
      IMT ( 1 )     = 0
      IMT ( 2 )     = 0
      ELVAY ( 1 )   = ELVNC
      ELVAY ( 2 )   = 2.55
      COLTMP ( 1 ) = 660.0
      COLTMP ( 2 ) = 670.0
      BEDTMP ( 1 ) = 660.0
      BEDTMP ( 2 ) = 670.0
C
C INITIALIZE ELEMENTS 3 THRU NOND OF THE ABOVE CUTPUT ARRAYS
C
C
      DO  60  I = 3 , NOND
C
          FACTOR = ( I - 3 ) / DIV
C
          IMT ( I )     = 0
          ELVAY ( I )   = 2.6 + 0.4 * FACTOR
          COLTMP ( I ) = 680.0 + 190.0 * FACTOR
          BEDTMP ( I ) = COLTMP ( I ) * FACTOR
C
   60 CONTINUE
C
C ASSIGN VALUES TO DATA IN FRCHAR ARGUMENT LIST
C
      XMASSC = 1.3122E-01 * 264.0
      ZRMC   = 1.99556E-02 * 264.0
      ZRO2MC = 1.419E-03 * 264.0
      UO2MC  = 1.09844E-01 * 264.0
      STRMC  = 0.0
      ABSMC  = 0.0
      AREAR  = 3.664E-02
      VINCOD = 5.05
C
      RETURN
      END
```

TABLE A-18. LISTING OF DBTIME

```
C
C
      SUBROUTINE  DBTIME   ( TIME  , N )
C
C THIS IS A DUMMY SUBROUTINE WHICH WILL CALCULATE THE TIME INTERVALS FOR
C THE DISRUPTED BUNDLE LOGIC.
C
C
      DIMENSION   TIME ( N + 1 )
C
C
      DELTAT = 10.0
C
      DO  20  I = 1 , N + 1
C
         TIME ( I ) = FLOAT ( I - 1 ) * DELTAT
C
   20 CONTINUE
C
C
      RETURN
      END
```

TABLE A-19. LISTING OF DBOUTD

```
      SUBROUTINE  DBOUTD
     +                  ( HITER  , POROSR , POROPR , XCHRL  , KFLUID ,
     +                    KTERM  , IDREGR , NONR   , ELVAR  , BEDTMR ,
     +                    COLTMR , IRT    , THMOTR , DOWNR  , VELMOT ,
     +                    ELVMR  , ESTOT  , HITEC  , POROSC , POROPC ,
     +                    YCHRL  , EFFDIA , KTERMC , IDREGN , NOND   ,
     +                    ELVAY  , BEDTMP , COLTMP , IMT    , THMOTC ,
     +                    ELVMC  , DOWNZ  , VELMTC , CURTIM ,
     +                    DELTT  , FRAG   , FROZEN )
C
C
C  SUBCODE NAME:  DBOUTD
C  PURPOSE:  TO PRINT THE DEBRIS BEHAVIOR ANALYSIS OUTPUT
C  CALLING SUBROUTINE:  DBOUT
C  SUBROUTINES CALLED:  NONE
C  WORK PACKAGE:  15
C  ENGINEER/PROGRAMMER:  S.T.HSIEH/G.H.BEERS
C  LAST MODIFICATION DATE:  11/30/81
C
C
C
C  THIS SUBROUTINE WRITES THE OUTPUT DATA FROM THE DEBRIS BEHAVIOR
C  MODELS *DBFRAG* AND *DBFROZ* TO TAPE6.
C
C  *DBOUTD* IS CURRENTLY IN THE FORMAT REQUIRED BY THE DRIVER ROUTINE
C  SIMULATING *DBUNDL* WHICH WAS WRITTEN TO PERFORM THE ACCEPTANCE
C  TESTING OF *DBFRAG* AND *DBFROZ*. THE OUTPUT WILL BE PRINTED
C  DEPENDING UPON THE VALUE OF THE DEBRIS REGION TYPE FLAG RBDRPT(K).
C
C
C
      DIMENSION       ELVAR  ( 41 ) , BEDTMR ( 41 ) , COLTMR ( 41 ) ,
     +                IRT    ( 41 ) , ELVAY  ( 41 ) , BEDTMP ( 41 ) ,
     +                COLTMP ( 41 ) , IMT    ( 41 )
C
C
      LOGICAL     FRAG , FROZEN
C
C
C
      WRITE  ( 6 , 900 )  ( '*' , I = 1 , 25 )
      WRITE  ( 6 , 1000 )  CURTIM , DELTT
C
      IF  ( FRAG )   THEN
C
          WRITE  ( 6 , 1010 )
          WRITE  ( 6 , 1020 )  HITER , POROSR , POROPR , XCHRL ,
     +                         KFLUID , KTERM
          WRITE  ( 6 , 1030 )
C
          WRITE  ( 6 , 1040 )  IDREGR , NONR ,
     +                         ( ELVAR( I ) , I = 1 , NONR )
          WRITE  ( 6 , 1050 )  'BED TEMPERATURE' ,
     +                         ( BEDTMR( I ) , I = 1 , NONR )
          WRITE  ( 6 , 1050 )  'COOLANT TEMPERATURE' ,
```

TABLE A-19. (CONTINUED)

```
     +                         ( COLTMR( I ) , I = 1 , NONR )
          WRITE  ( 6 , 1060 )  'DEBRIS REGION TYPE FOR NODES 1 TO NONR',
     +                         ( IRT( I ) , I = 1 , NONR )
          WRITE  ( 6 , 1070 )   THMOTR , DOWNR , VELMOT , ELVMR ,
     +                          ESTDT
C
C
      ELSE IF  ( FROZEN )   THEN
C
          WRITE  ( 6 , 2010 )
          WRITE  ( 6 , 2020 )   HITEC , POROSC , PDROPC , YCHRL ,
     +                          EFFDIA , KTERMC
          WRITE  ( 6 , 2030 )
          WRITE  ( 6 , 1040 )   IDREGN , NOND ,
     +                          ( ELVAY( I ) , I = 1 , NOND )
          WRITE  ( 6 , 1050 )   'BED TEMPERATURE' ,
     +                          ( BEDTMP( I ) , I = 1 , NOND )
          WRITE  ( 6 , 1050 )   'COOLANT TEMPERATURE' ,
     +                          ( COLTMP( I ) , I = 1 , NOND )
          WRITE  ( 6 , 1060 )   'DEBRIS REGION TYPE FOR NODES 1 , NOND',
     +                          ( IMT( I ) , I = 1 , NOND )
          WRITE  ( 6 , 1070 )   THMOTC , ELVMC , DOWNZ , VELMOT ,
     +                          ESTDT
C
      ENDIF
C
      RETURN
C
C
  900 FORMAT  ( // 2X , 25A )
 1000 FORMAT  ( // 2X , 'CURRENT TIME =' , F8.4
     +           / 2X , 'TIME STEP =' , 2X , F8.4   )
 1010 FORMAT  ( // 2X , 'OUTPUT FROM FRAGMENTED CHARACTERISTCS MODEL')
 1020 FORMAT  (  / 6X , 'BED HEIGHT =' , F10.3
     +           / 6X , 'BED POROSITY =' , E12.4
     +           / 6X , 'PRESSURE DROP ACROSS BED =' , E12.4
     +           / 6X , 'CHARACTERISTIC LENGTH OF PARTICAL =' , E12.4
     +           / 6X , 'FLUIDIZATION FLAG =' , I4
     +           / 6X , 'ABNORMAL TERMINATION FLAG =' , I4 )
C
 1030 FORMAT  ( // 2X , 'OUTPUT FROM FRAGMENTED DEBRIS BEHAVIOR MODEL')
 1040 FORMAT  (  / 6X , 'REGION TYPE ID =' , I4
     +           / 6X , 'NUMBER OF NODES =' , I6
     +           / 6X , 'ELEVATION OF NODES IN DEBRIS =',10(/3X,8F12.4))
 1050 FORMAT  (  / 6X , A , 10(/3X , 5F12.3 ))
 1060 FORMAT  (  / 6X , A , 10(/3X , 10I6 ))
 1070 FORMAT  ( // 6X , 'MOLTON MATERIAL THICKNESS =' , E15.4
     +           / 6X , 'LENGTH OF MOLTON MATIERAL THICKNESS =',F15.4
     +           / 6X , 'VELOCITY OF MOLTON POOL FLOWING DOWNWARD =' ,
     +                                                          E15.4
     +           / 6X , 'ELEVATION OF MOLTON MATERIAL =' , E15.4
     +           / 6X , 'ESTIMATED TIME OF DISRUPTION =' , F 12.4 )
 2010 FORMAT  ( // 2X , 'OUTPUT FROM FROZEN CHARACTERISTCS MODEL')
```

TABLE A-19. (CONTINUED)

```
 2020 FORMAT  (  / 6X , 'BED HEIGHT =' , F10.3
     +           / 6X , 'BED POROSITY =' , E12.4
     +           / 6X , 'PRESSURE DROP ACROSS BED =' , E12.4
     +           / 6X , 'CHARACTERISTIC LENGTH OF FROZEN BED =',E12.4
     +           / 6X , 'EFFECTIVE PARTICAL DIAMETER =' , E12.6
     +           / 6X , 'ABNORMAL TERMINATION FLAG =' , I4 )
C
 2030 FORMAT  ( // 2X , 'OUTPUT FROM FROZEN DEBRIS BEHAVIOR MODEL' )
C
C
      END
```