

FUEL PERFORMANCE CODE "ACTIVE-II"

July, 1975

TOKAI WORKS

POWER REACTOR & NUCLEAR FUEL DEVELOPMENT CORPORATION

複製あるいは入手については、下記にお問い合わせください。

茨城県那珂郡東海村 〒 319-11

動力炉・核燃料開発事業団

東海事業所技術部研究管理課 ☎ 東海 (02928) 2-1111 内線 237

© Power Reactor and Nuclear Fuel Development Corporation — 1975

Enquiries about copyright and reproduction should be addressed to ;

Tokai Works, Power Reactor and Nuclear Fuel Development
Corporation,

Tokai, Ibaraki, Post No. 319-11, Japan.

July, 1975

Fuel Performance Code "ACTIVE-II"

Masami KATSURAGAWA*

Hiroaki KAWAMURA**

Abstract

The "ACTIVE-II" is a fuel performance analysis code for a thermal and mechanical analysis of cylindrical reactor fuel element under irradiation, as a function of the actual reactor operating history. The fuel element mechanical behavior is analyzed by a generalized plane-strain procedure using matrix, especially emphasizing fuel-clad mechanical interactions, fuel cracking behavior and fine analysis of stress and strain distributions of fuel and cladding. The code also includes models of mechanical behavior such as restructuring, swelling and creep of fuel, fission gas release, plutonium migration, and swelling and creep of cladding.

The ACTIVE-II code is written in FORTRAN-IV on IBM model 360/195. The output of the calculation is given not only in digital form but also in graphic display by plotter. Since the code is constructed by variable dimension method, number of meshes for the calculation is almost infinite as far as the computer can accept.

The code can be provided not only for sodium cooled fast breeder reactor fuels, but also for fuels of PWR, BWR and heavy water reactor like FUGEN constructing in Japan. The models, equations, materials' properties and entire program list of the ACTIVE-II code are written in this report.

The program of this code is available from Department of Nuclear Fuels Development, Power Reactor and Nuclear Fuel Development Corporation.

* Fuel Design Section, Plutonium Fuel Division, Tokai Works

** IBM Japan, Ltd.

Fuel Performance Code "ACTIVE-II"

実施責任者 青木利昌*

報告者 桂川正巳*

川村博章**

要 旨 ACTIVE-IIは、燃料棒の熱的・機械的挙動を原子炉の運転履歴を追って計算するコードである。円柱状の燃料要素は軸対称として扱われ、平面歪解析による方程式を行列式で解くもので、特に燃料と被覆管の機械的相互作用、燃料のクラックおよび燃料と被覆管内の応力一歪分布を詳細に計算することを重点にしている。

コードにはまた燃料の組織変化、燃料と被覆管のスエリングおよびクリープ、分裂性ガス放出、プルトニウムの拡散等の照射下での現象を考慮している。

プログラムはIBM 360/195用のFORTRAN-IVで書かれており、計算結果は数字とプロッターによる図面で出される。特にこのプログラムは可変ディメンジョン方式を採用しているため、計算するケースの大きさに制限はない。

対象とする燃料はナトリウム冷却高速炉、新型転換炉、加圧水型軽水炉および沸騰水型軽水炉の各種であり、オプションで選択できる。

本報告書ではプログラムのモデル、構成式、材料の物性値、およびプログラム・リストとその詳細な説明を記した。

プログラム入手御希望の場合には、核燃料開発本部宛御連絡下さい。

* 東海事業所プルトニウム燃料部設計課

** 日本アイ・ビー・エム株式会社

Contents

	Page
1. Introduction	1
2. Code Abstract	3
2.1 Model	3
2.2 Flow Diagram	3
3. Equations	9
3.1 Temperature distribution	9
3.2 Fuel restructuring	10
3.3 Plutonium migration	12
3.4 Plenum pressure and central void pressure	13
3.5 Creep deformation	14
3.6 Swelling	14
3.7 Mechanical analysis	15
3.8 Fuel cracking and healing	18
4. Solution Method	22
4.1 Program construction	22
4.2 Matrix solution	25
5. Material Properties	35
5.1 Material properties for sodium cooled fast breeder reactor fuels	35
5.2 Material properties for heavy water reactor fuels	43
5.3 Material properties for pressurized water reactor fuels	46
5.4 Material properties for boiling water reactor fuels ...	46
6. Input and Output Format	47
6.1 Core storage	47
6.2 Input parameter preparation	49
6.3 Input format	55
6.4 Output format	60

	Page
Acknowledgements	73
References	74
Appendix A. Subroutine Dictionary	75
Appendix B. Common Dictionary	129
Appendix C. Symbol Dictionary	145
Appendix D. Control Cards and Error Messages	159
Appendix E. Sample Input	163
Appendix F. Sample Output	165
Appendix G. Program List	189

1. Introduction

ACTIVE-II is a fuel performance analysis code for digital computer.

Under irradiation, fuel pin seems to be a living creature. Fuel cracks, creeps and swells. Fission gas bubbles move up to temperature gradient and form columnar grain region and central void. Fission product ingots are formed in central void, and drops during shutdown. Cladding creeps and swells, and so on. Thus, in a reactor, fuel pin is very vivid and "active".

This computer code analyzes the movements of fuel pin under irradiation, and reproduces these "active" movements following the actual reactor history. Therefore, this fuel performance code is named "ACTIVE".

The usage of this fuel pin performance code is very wide. That is, it can supply the input condition for transient, safety analysis codes of fuel pins. The code can be used to research the phenomena which is most sensitive to the fuel pin life. It may also evaluate the safety margin for a fuel pin design. Furthermore, fuel pin performance code can contribute to the operation manual of a reactor for more safety and economical usage of fuel.

There are many computer codes which calculate fuel pin performance; i.e. LIFE-I, FMODEL, CYGRO, BEHAVE, SATURN and so on. The authors had used LIFE-I code for a while, and compared the models with experimental results. But in order to compile some new models, and to shorten the computer processing time, ACTIVE-II was developed based on LIFE-I and other codes.

ACTIVE-II has some characteristics compared with these codes.

First, it can calculate fine structure of stress and strain distribution. By using variable dimension method, the user can choose as many meshes as he wishes.

Second, the code is made up by many modules. Therefore, it's easy to change any models in the code.

Third, ACTIVE-II can deal with any type of reactor fuel which has cylindrical shape. (i.e. fast breeder reactor fuel, heavy water reactor fuel, pressurized water reactor fuel and boiling water reactor fuel.)

Fourth, it can evaluate fuel cracking and healing which cause during reactor power change. Fuel cracking is evaluated until every stresses settle down below fracture stresses.

It is remarkable that the calculation time is minimized by adopting matrix method.

Although ACTIVE-II has many usages and characteristics, it is still under developing. There are many phenomena remaining unresolved which would be included in the code. These phenomena must be clarified through the verification of the code.

2. Code Abstract

2.1 Model

Assuming a cylindrical geometry, axi-symmetrical analysis is performed. As shown in Fig. 2.1, fuel column is divided into as many meshes as the user wishes by the plane perpendicular to the axis, and a plenum region is added at the top of the fuel column.

Fuel dimensions (i.e. clad outer diameter, clad inner diameter, fuel diameter and central hole diameter), pellet densities, O/M ratio, linear rod power and fast neutron flux are input for each axial region. These values are constant within each axial region, but have distributions for axial direction. Thermal and mass movements are not allowed between these axial regions.

Fuel central void is assumed to be separated from upper plenum during operation but to be connected during shutdown.

Radial cross section is shown in Fig. 2.2. Fuel is divided into three concentric zones corresponding to the regions with columnar grain growth region, equiaxed grain growth region and undisturbed region, and within these regions, fuel densities are assumed to be constant. These three regions and cladding region are divided into many submeshes for stress and strain analysis. For example, columnar, equiaxed, undisturbed and cladding regions are divided into 3, 3, 4, and 4 submeshes, respectively.

For these sub-meshes, temperature, materials' properties including creep and swelling, stress and strain (using plane-strain analysis) and fuel cracking are calculated.

For time intervals, quasi-steady state is assumed.

2.2 Flow diagram

Fig. 2.3 shows the flow diagram of ACTIVE-II. At the beginning of the MAIN program, dimensions for the program are decided according to the variable dimension method.

Capsule dimensions, material properties' options, relative power and fast flux distributions, and fuel densities are input as the initial value, and then, operation conditions are prepared assuming stepwise power change.

First, boundary movement and fuel densities are calculated using

the results of the former time step. Temperature distribution is calculated based on the coolant temperature assuming one dimensional calculation. Gap conductance between fuel and cladding is a function of gap width, and the thermal conductivity of fuel is a function of fuel density, temperature, and O/M ratio.

Using the equivalent stress of the former time step, the creep strains of fuel and cladding are calculated for each sub-region. If one of the increments of boundary movements, creep of fuel or creep of cladding during the time step is too much larger than the limit given by input, time interval is cut to half and the calculation is retried at the beginning of the temperature calculations.

Based on the temperature distribution, fuel restructuring, plutonium redistribution, fission gas release, plenum pressure, fuel burnup, fast neutron fluence and fuel and cladding swelling are also calculated.

Those results are used for the boundary conditions of stress and strain calculations, which are performed using matrix and gap condition iterations. That is, first open gap is assumed for the fuel - cladding interface and if the calculated results show no gap, the boundary conditions are reevaluated and the calculation is continued. After the calculation of stress and strain, stresses for radial, circumferential and axial directions of each sub-region are compared with the fracture stress, which depends on temperature, to check if the fuel cracking occurs. If one of the stresses is greater than the fracture stress, the code marks the fuel cracking at the region, and changes the Young's modulus of elasticity and Poisson's ratio of the region. Then the stress and strain are re-evaluated using these moderated properties, and the resulting stresses are compared with the fracture stresses.

After the convergence of the cracking routine or in the case of no cracking, healing conditions are checked and the whole calculated conditions are printed out before going to the next time step.

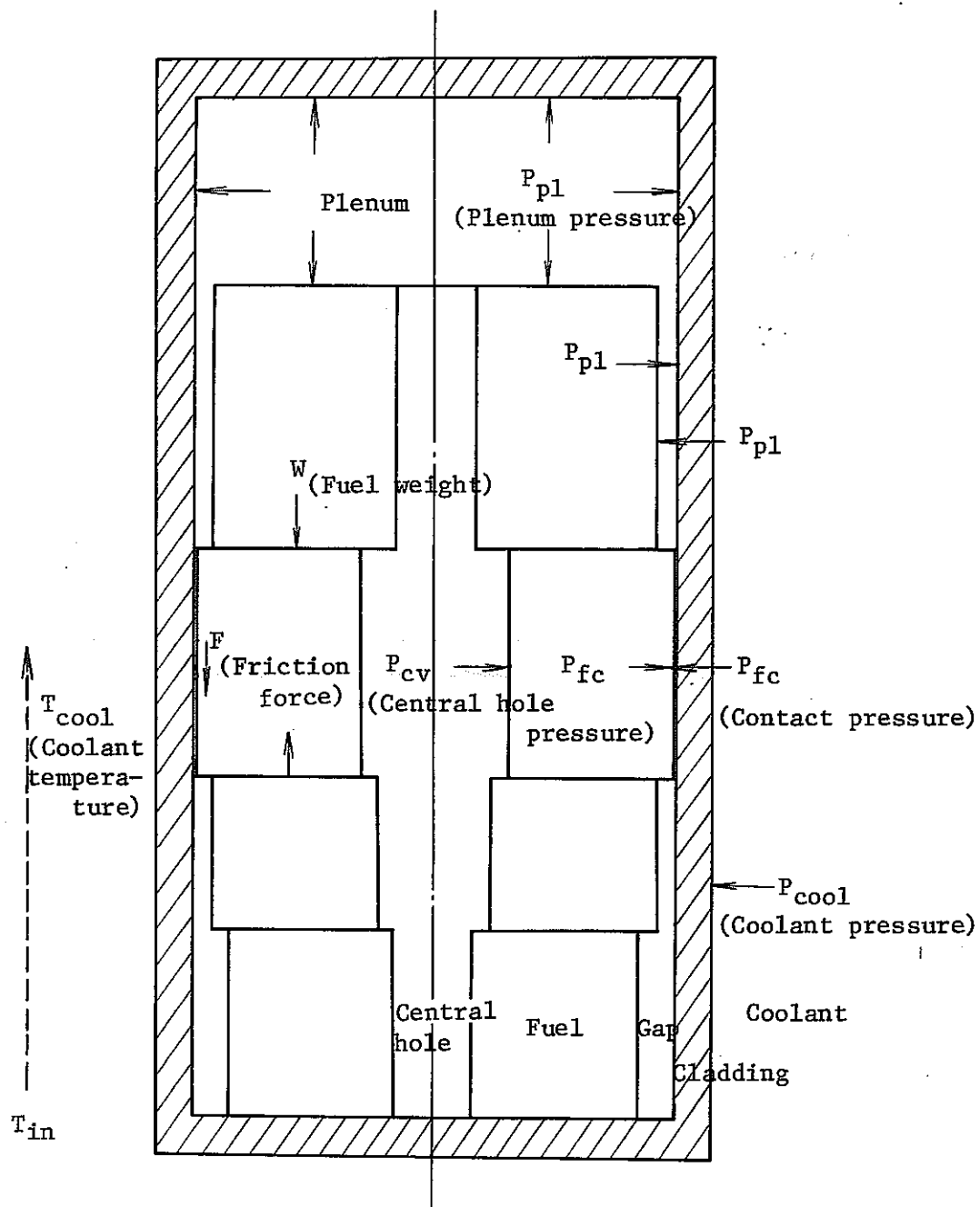


Fig. 2.1 Axial Cross Section of Fuel Pin

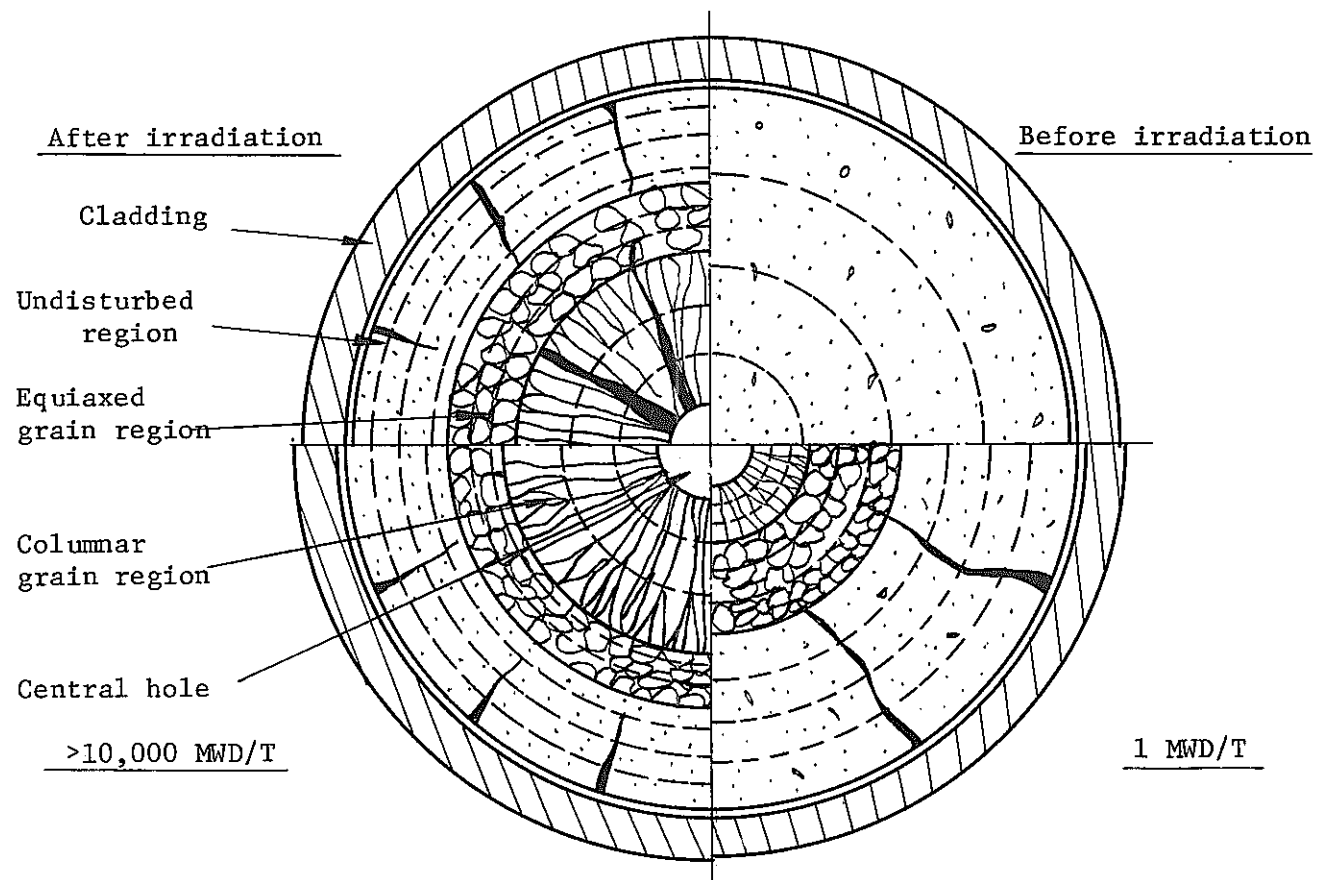


Fig. 2.2 Radial Cross Section of Fuel Pin

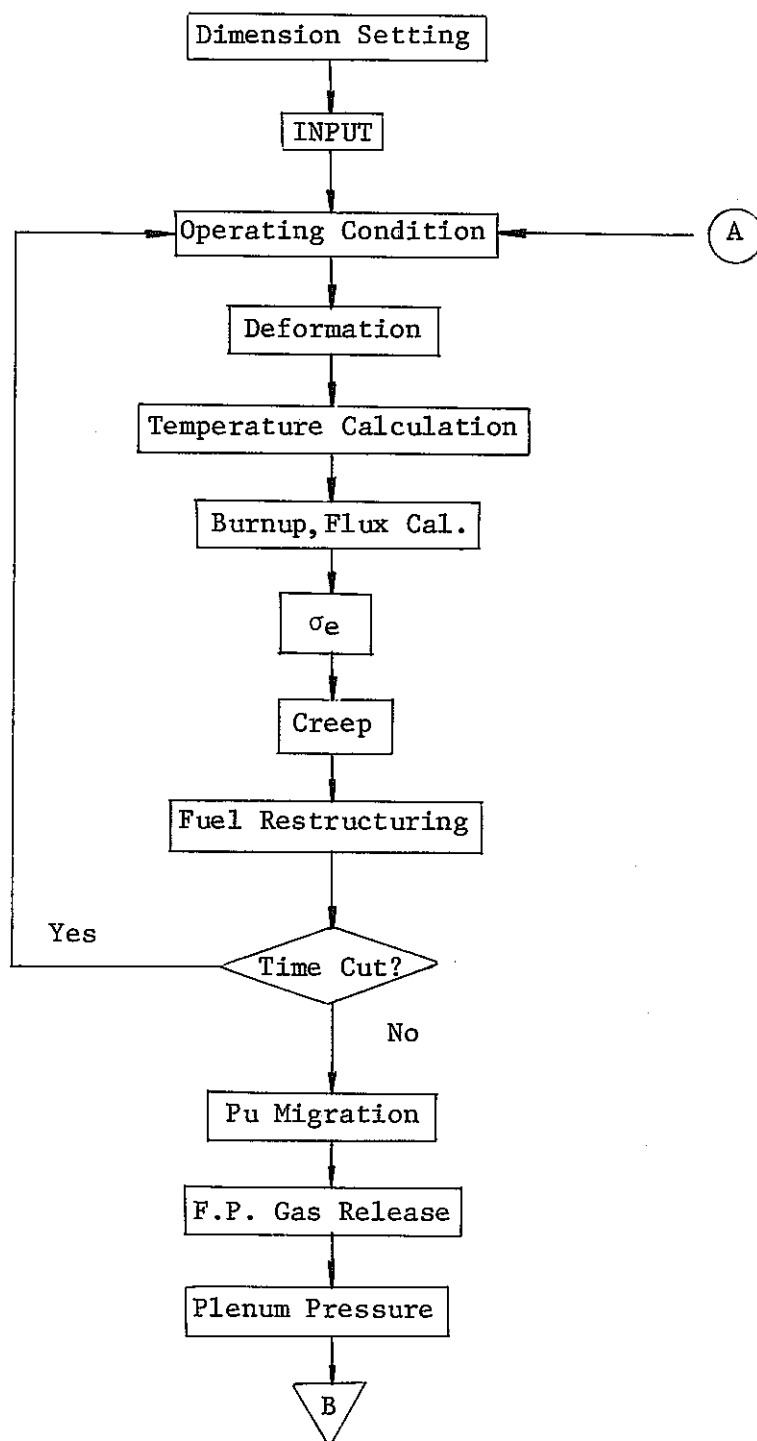


Fig. 2.3 Flow Diagram of ACTIVE-II

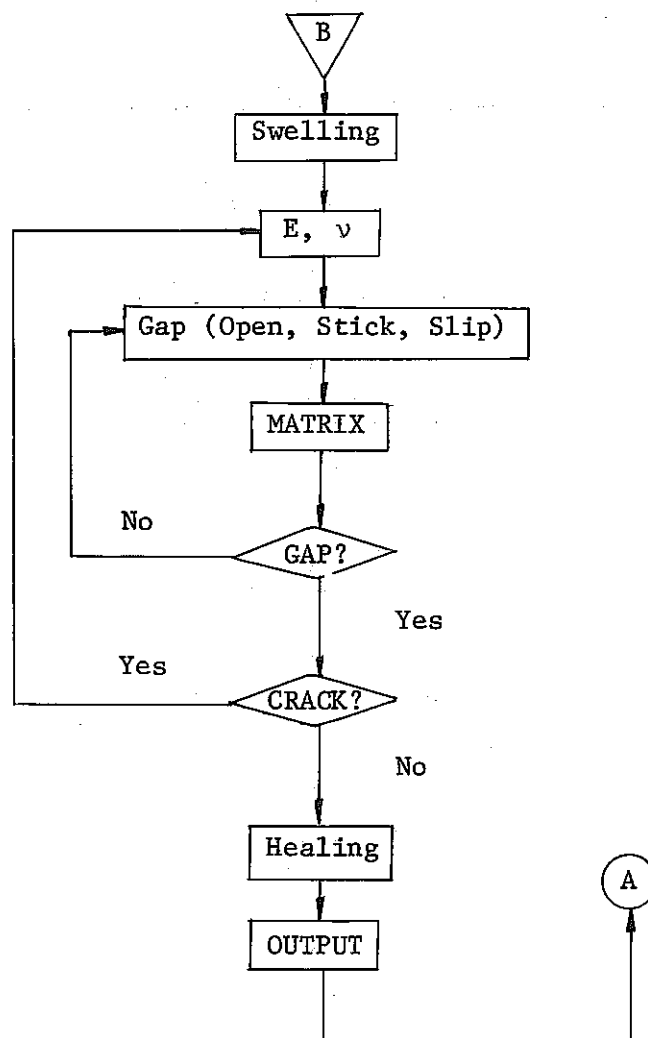


Fig. 2.3 Flow Diagram of ACTIVE-II (Continued)

3. Equations

3.1 Temperature Distribution

1) Coolant Temperature

Coolant temperature at the axial position (Z) is given by the following equation.

$$T_c(z) = T_c(0) + \frac{\int_0^z q(z) dz}{\rho C_p v s} \quad (1)$$

where, $T_c(z)$ = coolant temperature at the axial position (Z) ($^{\circ}\text{C}$),

$T_c(0)$ = coolant inlet temperature ($^{\circ}\text{C}$),

$q(z)$ = linear rod power (w/cm),

Z = axial distance from the bottom (cm),

C_p = coolant heat capacity (w·sec/g),

ρ = coolant density (g/cm³),

v = coolant velocity (cm/sec),

and s = coolant flow area (cm²).

2) Cladding Outer Surface Temperature

Cladding outer surface temperature is calculated based on coolant temperature at the same axial position.

$$T_{co} = T_c(z) + \frac{q}{2\pi r_{co} h_c} \quad (2)$$

where, T_{co} = cladding outer surface temperature ($^{\circ}\text{C}$),

$T_c(z)$ = coolant temperature ($^{\circ}\text{C}$),

q = linear rod power (w/cm),

r_{co} = cladding outer radius (cm),

and h_c = film coefficient (w/cm²· $^{\circ}\text{C}$)

3) Temperature Distribution in Cladding

$$T_c(i) = T_{co} + \frac{(q/\pi) \ln\{r_{co}/r_c(i)\}}{k_c + [k_c^2 + (a_2 q/\pi) \ln\{r_{co}/r_c(i)\}]^{1/2}} \quad (3)$$

where, $T_c(i)$ = cladding temperature at radius $r_c(i)$ ($^{\circ}\text{C}$),

k_c = thermal conductivity of cladding ($=a_1 + a_2 T$) (w/cm· $^{\circ}\text{C}$)

4) Fuel Surface Temperature

$$T_{fo} = T_{ci} + \frac{q}{2\pi r_g h_g} \quad (4)$$

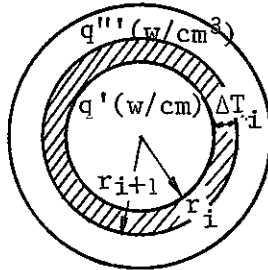
where, T_{fo} = fuel surface temperature ($^{\circ}\text{C}$),
 r_g = mean radius of fuel-clad gap (cm),
 T_{ci} = clad inner surface temperature ($^{\circ}\text{C}$),
 and h_g = gap conductance ($\text{W}/\text{cm}^2 \cdot ^{\circ}\text{C}$).

5) Temperature Distribution in Fuel

Fuel radius is divided into some meshes, and in each region power generation rate is assumed to be constant. Temperature rise across a region ($r_i \leq r \leq r_{i+1}$) is given by the equation

$$\Delta T_i = \frac{q'''}{4k_f} (r_{i+1}^2 - r_i^2) - \left(\frac{q'''}{2k_f} r_i^2 - \frac{q'}{2\pi k_f} \right) \ln \frac{r_{i+1}}{r_i}, \quad (5)$$

where q''' = heat generation rate in the region (W/cm^3),
 k_f = fuel thermal conductivity ($\text{W}/\text{cm} \cdot ^{\circ}\text{C}$),
 r_i = inner radius of the region (cm),
 r_{i+1} = outer radius of the region (cm),
 and q' = linear heat rating within the radius r_i (W/cm).



In these equations, materials' properties can be selected by options.

3.2 Fuel Restructuring

It is assumed that, during irradiation, fuel is divided into three regions (i.e. columnar grain growth region, equiaxed grain growth region and undisturbed region), and central hole opens at the center. These region boundaries are assumed to grow in a certain velocity.

1) Columnar Grain Growth Region

As columnar grain is believed to be formed by the movement of voids in the fuel, the velocity of the region boundary growth depends upon the velocity of void in the fuel. That is, the velocity is calculated by the following equation.

$$v = C_1 T_k^{-3/2} e^{-\frac{Q}{RT_k}} \frac{dT}{dr} \quad (\text{cm/sec}) \quad (6)$$

where C_1 = constant given by fine tuning of the code,
 T_k = temperature of columnar grain region outer boundary ($^{\circ}\text{K}$),
 Q = activation energy (cal/mole),
 $R = 1.987$ (cal/mole $\cdot^{\circ}\text{K}$)
 and $\frac{dT}{dr}$ = temperature gradient at columnar grain region outer boundary ($^{\circ}\text{C/cm}$).

2) Equiaxed Grain Growth Region

Assuming that equiaxed grain growth depends on the temperature only, the velocity of the region boundary is calculated by the following equation.

$$v = C_2 e^{-\frac{Q}{RT}} \quad (\text{cm/sec}), \quad (7)$$

where C_2 = constant calculated by fine tuning of the code,
 T = region outer boundary temperature ($^{\circ}\text{K}$),
 Q = activation energy (cal/mole),
 and $R = 1.987$ (cal/mole $^{\circ}\text{K}$).

3) Central Void

As the results of the restructuring described above, a certain central void is formed at the center of the fuel. The radius of the central hole is calculated by the following mass balance equation.

$$M = \pi \cdot \rho_{\text{col}} (r_{\text{col}}^2 - r_{\text{cv}}^2) + \pi \cdot \rho_{\text{eq}} (r_{\text{eq}}^2 - r_{\text{col}}^2) + \pi \cdot \rho_{\text{und}} (r_{\text{f}}^2 - r_{\text{eq}}^2) \quad (8)$$

where M = fuel weight per unit length (g/cm),

ρ_i = fuel density of region i (g/cm³),

r_i = outer radius of region i (cm),

and the suffix i means,

cv = central void,

col = columnar grain growth region,

eq = equiaxed grain growth region,

and f = fuel pellet.

4) Average Densities of the Regions

Within each regions, density is assumed to be independent on the position, and is calculated as the functions of the initial pellet density and burnup.

Density of columnar grain growth region

$$\rho_{col} = \rho_{col}^{\circ} + (0.99 \times \rho_{th} - \rho_{col}^{\circ}) \times \frac{B}{B_o}, \quad (9)$$

$$\rho_{col}^{\circ} = 0.767 \times \rho_o + 0.233 \times \rho_{th}, \quad (10)$$

where ρ_{th} = theoretical density (g/cm³),

ρ_o = initial pellet density (g/cm³),

B = burnup (a/o),

and B_o = constant calculated through fine tuning of the code (a/o).

Density of equiaxed grain growth region

$$\rho_{eq} = \rho_{eq}^{\circ} + (0.965 \times \rho_{th} - \rho_{eq}^{\circ}) \times \frac{B}{B_o} \quad (11)$$

$$\rho_{eq}^{\circ} = \rho_o \quad (12)$$

Density of undisturbed region

$$\rho_{und} = \rho_o \quad (13)$$

3.3 Plutonium Migration

Plutonium redistribution is calculated in a subroutine, but the plutonium generation and the feed back to the power distribution are neglected.

The fuel is divided into some concentric cylindrical shells of equal mass, and the calculation is based on the assumption that the plutonium migration is caused by thermal and chemical diffusion.

$$\rho \frac{dC^{Pu}}{dt} = -\nabla J^{Pu}, \quad (14)$$

where, ρ = fuel density,

C^{Pu} = plutonium concentration,

J^{Pu} = flux of plutonium,

$$= \rho \times D' \times W^{Pu} \times W^U \times \nabla T - \rho \times D \times \nabla W^{Pu},$$

D' = thermal diffusion coefficient,

D = diffusion coefficient of plutonium in the mixed oxide fuel,

W^{Pu} = w/o of plutonium,

and W^U = w/o of uranium.

3.4 Plenum Pressure and Central Void Pressure

Enclosed gas (usually helium gas), absorbed gas in the fuel (H_2O , N_2 etc.) and fission product gas are mixed and are filled in the plenum, fuel-clad gap and central void. Central void is assumed to be a pressure vessel separated from the plenum during operation, and to be connected after shut down.

Plenum pressure is calculated by the equation

$$P_{pl} = \frac{nRT_{pl}}{V_{pl} + V_{cv} + V_g} \quad (15)$$

where P_{pl} = plenum pressure (kg/cm^2),

n = total gas moles (moles),

R = Boltzmann constant (cm^3 -atm/mole $^{\circ}K$),

T_{pl} = plenum temperature ($^{\circ}K$),

V_{pl} = plenum volume (cm^3),

V_g = fuel-clad gap volume (cm^3),

and V_{cv} = central void volume (cm^3).

Central void pressure is calculated by the equation

$$P_{cv} = \left(n \times \frac{V_{cv}}{V_{pl} + V_{cv}} \right) \times \frac{R}{\frac{\sum_{i=1}^{NF} V_{cv}(i)}{T_{cv}(i)}} \quad (16)$$

where P = central void pressure (kg/cm^2),

NF = number of axial nodes,

$T_{cv}(i)$ = central void temperature of axial node-i ($^{\circ}\text{K}$),

and $V_{cv}(i)$ = central void volume of axial node-i ($^{\circ}\text{K}$),

3.5 Creep Deformation

Creep deformation is calculated based on the second creep rate of the material, and the creep rate functions include both thermal and irradiation enhanced creep.

$$\dot{\epsilon}_p = A\sigma_e^m e^{-\frac{Q}{RT}} + B\sigma_e^n \phi \quad (17)$$

where $\dot{\epsilon}_p$ = creep rate (1/hr),

Q = activation energy for thermal creep (cal/mole),

R = 1.987 (cal/mole $^{\circ}\text{K}$),

T = temperature ($^{\circ}\text{K}$),

ϕ = neutron flux (n/cm²·sec),

σ_e = equivalent stress

$$= \frac{1}{\sqrt{2}} [(\sigma_r - \sigma_{\theta})^2 + (\sigma_{\theta} - \sigma_z)^2 + (\sigma_z - \sigma_r)^2]^{1/2},$$

and A, B, m, n = constants.

3.6 Swelling

1) Fuel

Swelling of fuel is divided into three components, which are swelling by solid f.p., by gaseous f.p. and hot pressing.

$$\dot{\epsilon}_s = \dot{\epsilon}_{s.s.} + \dot{\epsilon}_{s.g.} + \dot{\epsilon}_{s.hp} \quad (18)$$

where $\dot{\epsilon}_s$ = total swelling rate (1/hr)

$\dot{\epsilon}_{s.s.}$ = swelling caused by solid fission products (1/hr),

$\dot{\epsilon}_{s.g.}$ = swelling caused by gaseous fission products (1/hr),

and $\dot{\epsilon}_{s.hp}$ = hot pressing components (1/hr).

The equations for these components will be given in the section of materials' properties.

2) Cladding

The swelling rate of cladding is chosen from some empirical equations which are given in the section of materials' properties.

3.7 Mechanical Analysis

Equations between stress and strain are given for each concentric cylinder regions based on plane-strain analysis.

Nomenclature used in the equations are as follows.

$\sigma_r, \sigma_\theta, \sigma_z$ = principal stresses for r, θ and z,

$\epsilon_r, \epsilon_\theta, \epsilon_z$ = total strains for r, θ and z directions,

$\epsilon_r^D, \epsilon_\theta^D, \epsilon_z^D$ = creep strains for r, θ and z directions,

$\epsilon_r^S, \epsilon_\theta^S, \epsilon_z^S$ = swelling strains for r, θ and z directions,

u = displacement of radial direction during time interval of Δt (excluding fuel restructure),

αT = thermal expansion,

E = Young's modulus of elasticity,

γ = Poisson's ratio,

P = pressure,

and Δ = increment during time interval of Δt .

1) Constitutive relations

Total strain is the sum of elastic strain, thermal expansion, creep strain and swelling strain.

$$\epsilon_r^T = \frac{1}{E} \{ \sigma_r - \nu(\sigma_\theta + \sigma_z) \} + \alpha T + \epsilon_r^D + \epsilon_r^S \quad (19)$$

$$\epsilon_\theta^T = \frac{1}{E} \{ \sigma_\theta - \nu(\sigma_z + \sigma_r) \} + \alpha T + \epsilon_\theta^D + \epsilon_\theta^S \quad (20)$$

$$\epsilon_z^T = \frac{1}{E} \{ \sigma_z - \nu(\sigma_r + \sigma_\theta) \} + \alpha T + \epsilon_z^D + \epsilon_z^S \quad (21)$$

2) Equilibrium

As axial symmetry is assumed, principal stresses are combined by the equation,

$$\frac{\partial \sigma_r}{\partial r} + \frac{\sigma_r - \sigma_\theta}{r} = 0 \quad (22)$$

3) Equation between strain and displacement

As a state of generalized plane strain is assumed, strain

increments in Δt are functions of increment of radial displacement in Δt as follows.

$$\Delta \epsilon_r^T = \frac{\partial u}{\partial r} \quad (23)$$

$$\Delta \epsilon_\theta^T = \frac{u}{r} \quad (24)$$

$$\Delta \epsilon_z^T = \text{constant} \quad (25)$$

4) Creep law

Prandtl-Reuss flow laws are used for creep strain calculation.

$$\Delta \epsilon_r^P = \frac{\Delta \epsilon_P}{2\sigma_e} (2\sigma_r - \sigma_\theta - \sigma_z), \quad (26)$$

$$\Delta \epsilon_\theta^P = \frac{\Delta \epsilon_P}{2\sigma_e} \times (2\sigma_\theta - \sigma_z - \sigma_r), \quad (27)$$

$$\Delta \epsilon_z^P = \frac{\Delta \epsilon_P}{2\sigma_e} \times (2\sigma_z - \sigma_r - \sigma_\theta), \quad (28)$$

where

$$\sigma_e = \frac{1}{\sqrt{2}} \{(\sigma_r - \sigma_\theta)^2 + (\sigma_\theta - \sigma_z)^2 + (\sigma_z - \sigma_r)^2\}^{1/2} \quad (29)$$

(equivalent stress),

and

$$\Delta \epsilon_P = f_S(\sigma_e)$$

(which is given by the equation (17)).

5) Boundary conditions

At the inner and outer boundary of each mesh, radial stress is equivalent to the contact pressure.

$$\sigma_r = -P_i \quad \text{at } r = r_i, \quad (30)$$

$$\sigma_r = -P_o \quad \text{at } r = r_o, \quad (31)$$

where i means inner boundary and o means outer boundary.

From the pressure balance between adjacent meshes,

$$P_{o,j-1} = P_{i,j}, \quad (32)$$

and from the continuity of the boundary,

$$U_{o,j-1} = U_{i,j}, \quad (33)$$

where j denotes mesh number.

At the central hole of fuel, contact pressure equals central hole gas pressure,

$$P_{i,1} = P_{cv}, \quad (34)$$

and at the outer surface of cladding, contact pressure equals coolant pressure,

$$P_{o,clad} = P_{cool}. \quad (35)$$

At the interface between fuel and cladding, two kinds of conditions are considered.

(i) When gap exists between fuel and clad, pressure of fuel outer surface and that of clad inner surface equal to plenum pressure.

$$P_{o,fuel} = P_{pl}, \quad (36)$$

$$P_{i,clad} = P_{pl}. \quad (37)$$

(See Fig. 3.1)

(ii) When fuel contacts with cladding, pressure of both fuel outer and cladding inner surface equal to contact pressure, and increments of radial displacement of both boundary are same.

$$P_{o,fuel} = P_{i,clad} = P_{fc} \quad (38)$$

$$U_{o,fuel} = U_{i,clad} \quad (39)$$

(See Fig. 3.2)

As the boundary conditions of axial direction, mean stresses of both fuel and cladding axial direction balance outside forces.

Friction force between fuel and cladding is also included in the boundary conditions of axial direction. (See Fig. 3.3)

(i) Fuel

$$\bar{\sigma}_{zf} = P_{pl} + \frac{\rho_f \cdot g \cdot V_f}{\pi(r_f^2 - r_c^2)} \times \frac{(L_f - Z)}{L_f} + \frac{F \times \Delta L_f}{\pi(r_f^2 - r_c^2)} \quad (40)$$

where $\bar{\sigma}_{zf}$ = mean stress of axial direction in fuel,
 ρ_f = mean density of fuel,
 g = gravity constant,
 V_f = fuel volume per unit axial length,
 r_f = fuel outer radius,
 r_c = fuel central void radius,
 L_f = total fuel column length,
 Z = distance from bottom of fuel,
 F = $2\mu\pi r_f P_{fc}$ (friction force),
 μ = friction factor,
 P_{fc} = contact pressure between fuel and cladding,
and ΔL_f = radial node length.

(ii) Cladding

$$\bar{\sigma}_{zc} = \frac{1}{\pi(r_{co}^2 - r_{ci}^2)} \times \{P_T + \pi(r_{co}^2 P_c - r_{ci}^2 P_{pl})\} + \rho_c \cdot g \cdot V_c \cdot (1 - \frac{Z}{L_c}) - F \times \Delta L_f \quad (41)$$

where $\bar{\sigma}_{zc}$ = mean stress of axial direction in cladding,
 ρ_c = density of cladding,
 V_c = cladding volume per unit axial length,
 r_{co} = cladding outer radius,
 r_{ci} = cladding inner radius,
and L_c = total cladding length.

6) Mean values in each region

Mean values in each region are given by the following equation.

$$\bar{A}(r) = \frac{\int r A(r) dr}{\int r dr} \quad (42)$$

3.8 Fuel Cracking and Healing

During the stress-strain calculation, ACTIVE-II checks whether fuel cracking would occur in any of the three principal directions. The cracking criterion is; if any of the principal stresses in a fuel region reach the fracture strength, the fuel cracks to the perpendicular direction to that principal stress in that region. The fracture stress used in the code is the data prepared by Canon et al.¹⁾.

$$\sigma_f = 15,000 + 3.7 \times T_k \quad (43)$$

where σ_f is the ultimate tensile stress in psi, and T_k is the temperature in °K. If crack occurs, the modifications mentioned below are applied to that region, and the stresses and strains are recalculated until none of the principal stresses in the fuel exceed the fracture stress.

The modification used in ACTIVE-II is the method proposed by Weeks et al. for LIFE-II²⁾ code. The stress component perpendicular to a crack vanishes in the region, and the elastic strain existing prior to cracking is then redistributed. For instance, when the stress of the axial direction σ_z exceeds the fracture stress σ_f , a crack occurs perpendicular to the axial direction. The axial elastic strain then decreases by σ_z/E , and the elastic strains in the other principal directions increase by $\nu\sigma_z/E$ each. As it is assumed that no volume change occurs by cracking, these changes in elastic strains are compensated by equal changes in plastic strains.

In ACTIVE-II code, plastic strains are divided into creep strains and swelling strains. Therefore the changes in plastic strains by creacking are described as follows.

Swelling strains for each direction

$$\Delta\epsilon^S = \sigma_z/9B \quad (44)$$

Creep strains

$$\Delta\epsilon_r^P = -\sigma_z/6G \quad (45)$$

$$\Delta\epsilon_\theta^P = -\sigma_z/6G \quad (46)$$

$$\Delta\epsilon_z^P = \sigma_z/3G \quad (47)$$

where B is the bulk modulus and G is the shear modulus.

The mechanical properties of the cracked fuel are modified as follows:

Young's modulus

$$E^C = E \times \left(-\frac{2}{3}\right)^L \quad (48)$$

Poisson's ratio

$$v^c = v \times \left(\frac{1}{2}\right)^L \quad (49)$$

Creep rate

$$\dot{\epsilon}_p^c = \dot{\epsilon}_p \times \left(\frac{3}{2}\right)^{L \times (m-1)} \quad (50)$$

where L is the number of cracks and m is the stress exponent for the thermal creep.

During steady state operation, crack healing is allowed. Cracks are assumed to heal under the following conditions.

- * The averaged stress in the region becomes compressive and exceeds a certain stress (7000kg/cm²).
- * The temperature in the region exceeds 1400°C, and the pressure is compressive and exceeds 70 kg/cm².
- * These conditions have continued more than 1 hour.

When the cracks heal, the mechanical properties of the region become those of uncracked fuel, but the plastic strains caused by cracking remain as before.

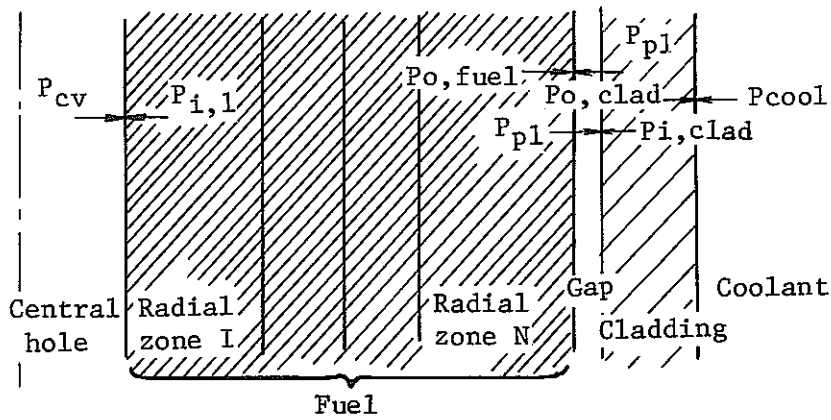


Fig. 3.1 Radial Force Balance (Gap Open)

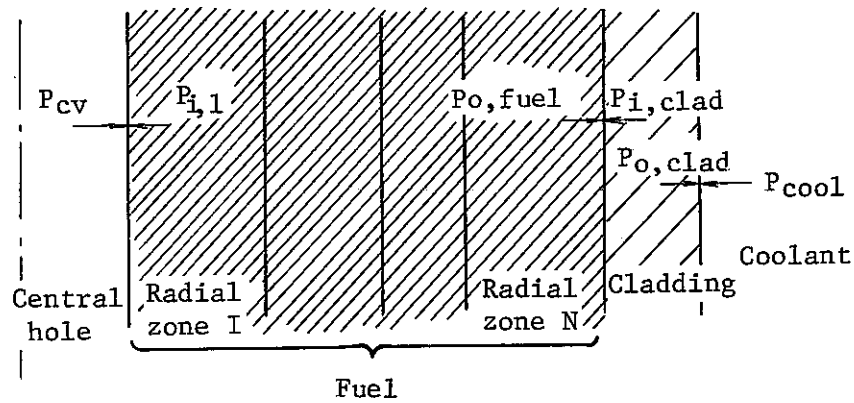


Fig. 3.2 Radial Force Balance (Gap Closed)

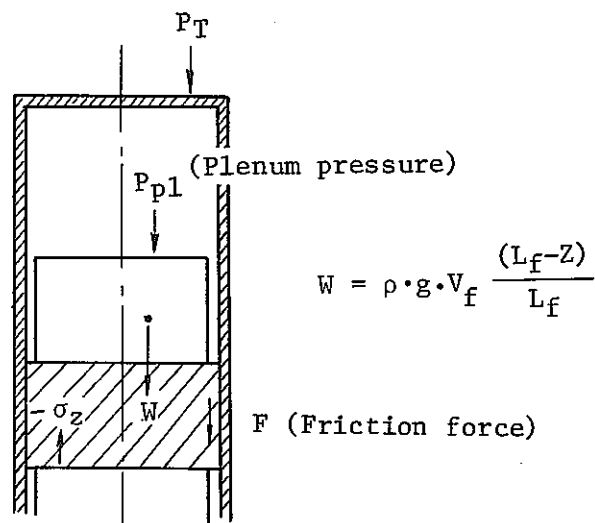


Fig. 3.3 Axial Force Balance

4. Solution Method

4.1 Program Construction

The code is written in FORTRAN-IV by using variable dimension method, in order to minimize the program size. Therefore, the MAIN routine is only used to decide the program size and the real calculation is done in subroutine ACTIVE which is called by MAIN ROUTINE.

1) MAIN program

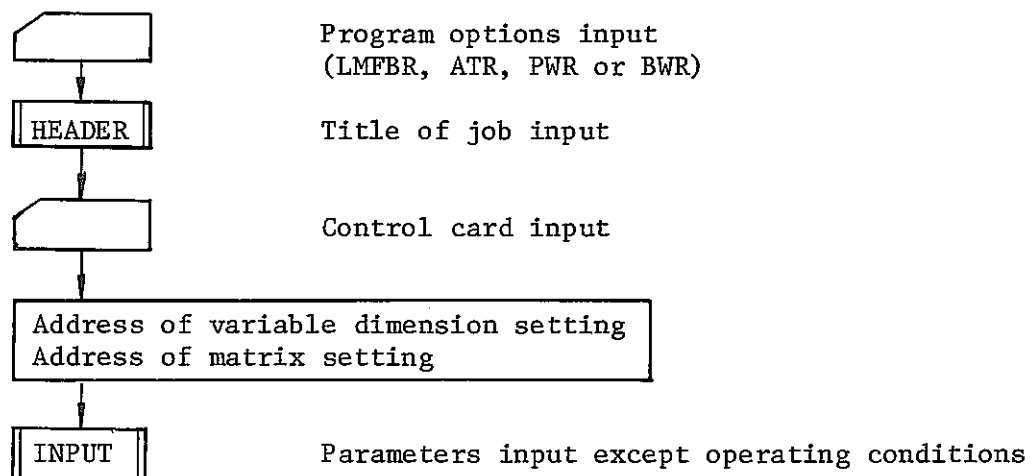
The program of MAIN is as follows.

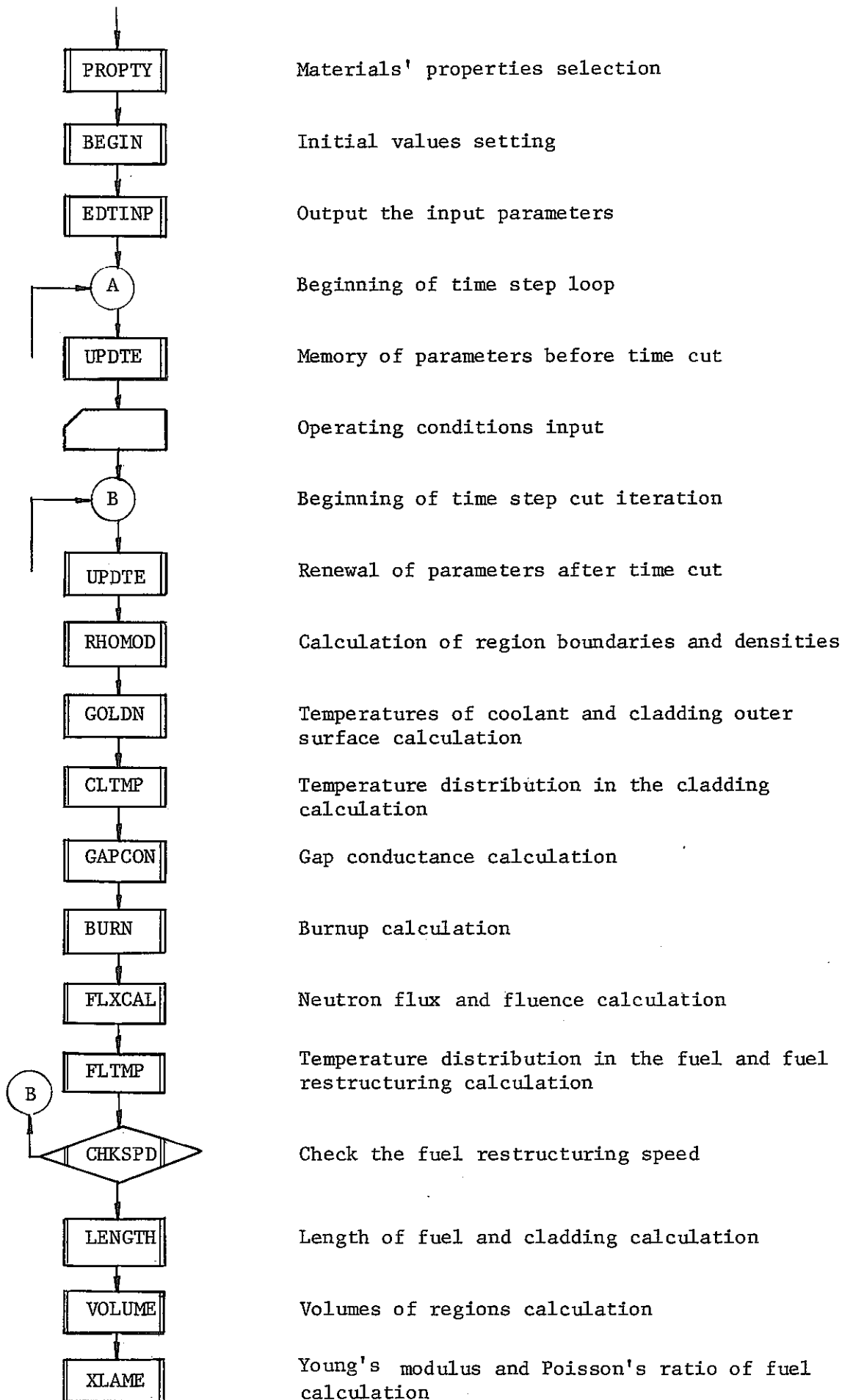
```
COMMON / DIM / IA,IAM
DIMENSION A(10000),AM(30000)
IA = 10000
IAM = 30000
CAL ACTIVE(A,AM)
STOP
END
```

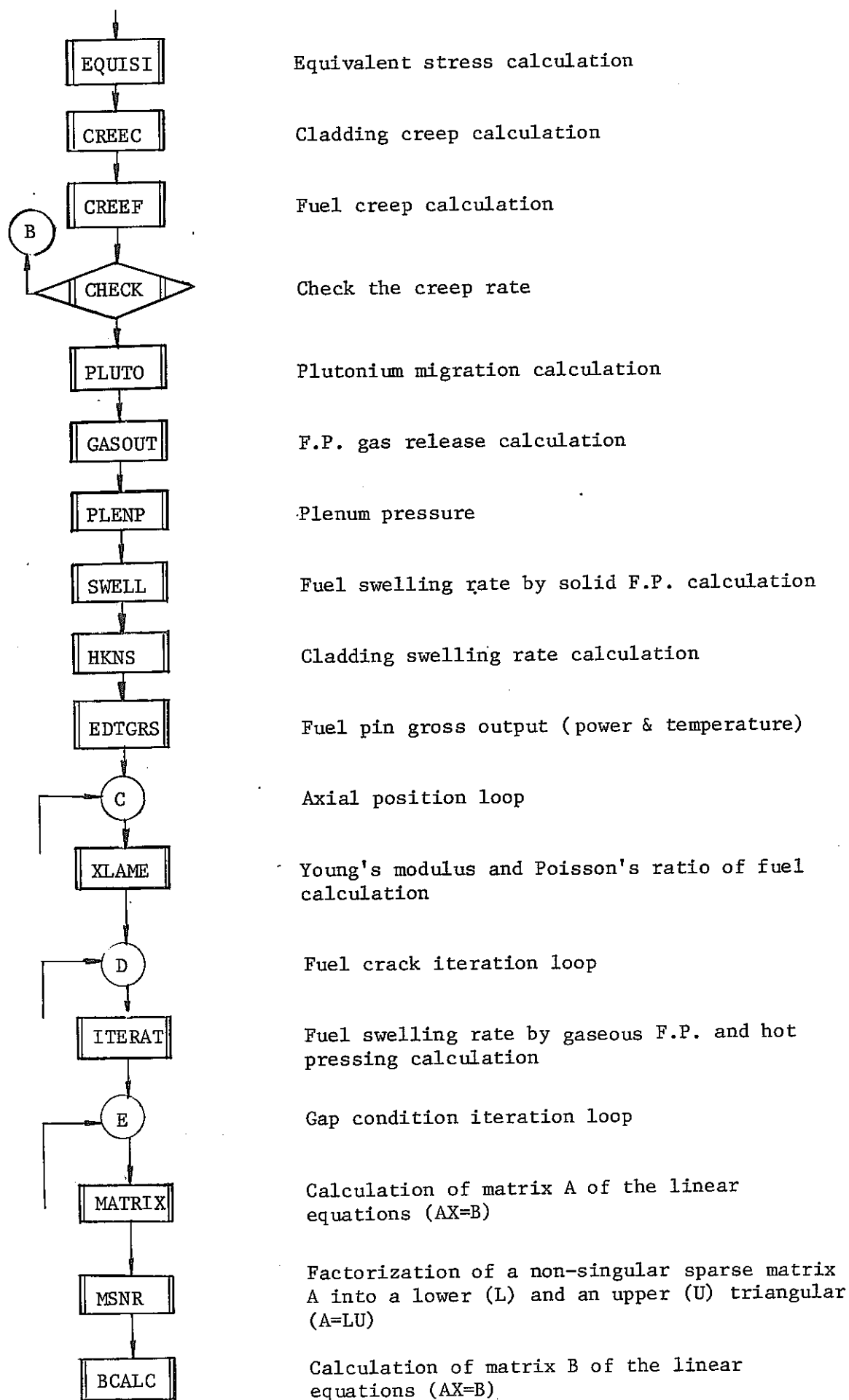
In this program, symbol A denotes the all symbols which have dimensions except the symbols used in matrix solution. IA is the number of As. AM denotes the symbols used in matrix solution and IAM is the number of AMs. See section 6.1 for details.

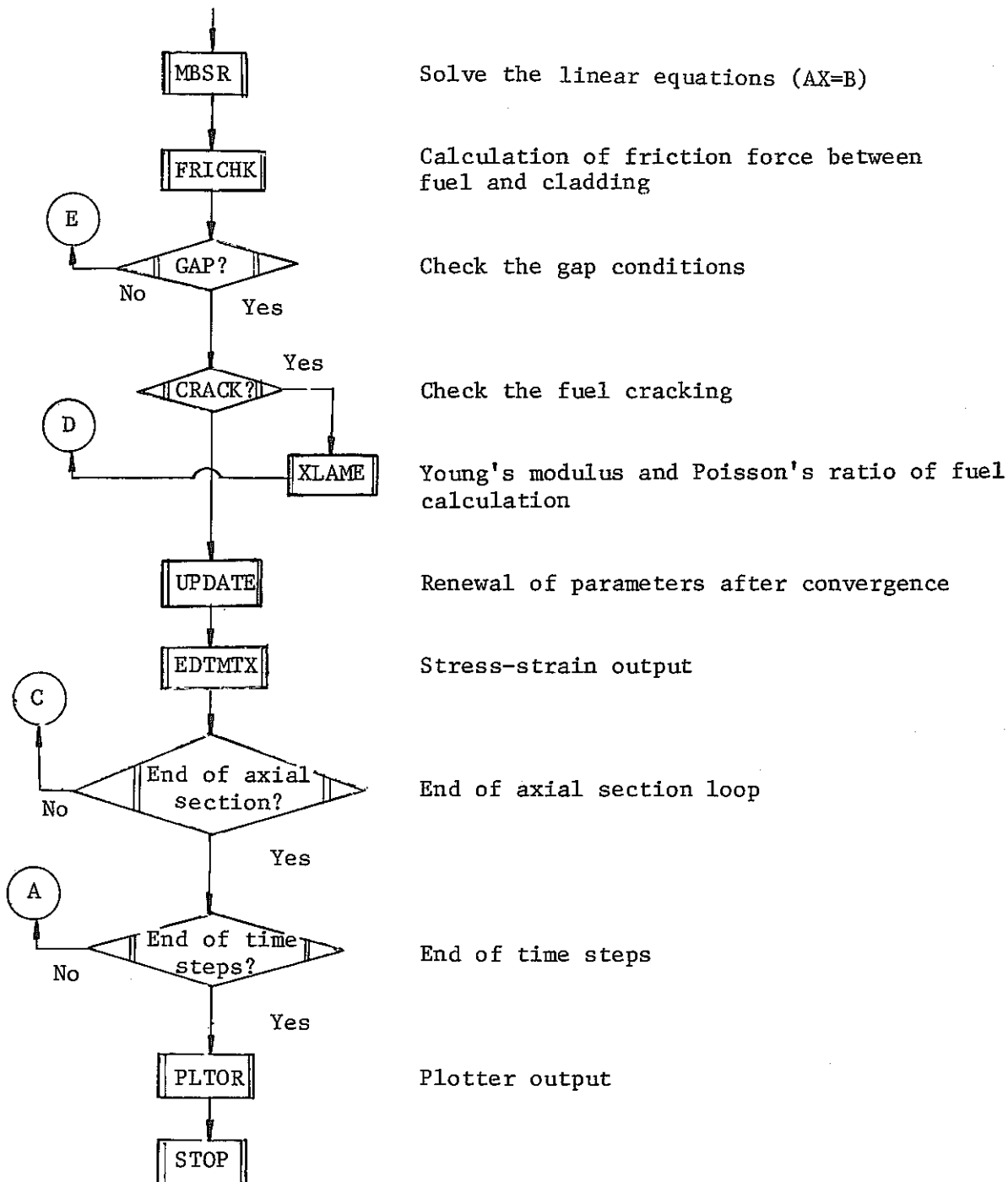
2) Subroutine ACTIVE

As mentioned above, the calculation is done in the subroutine ACTIVE. The flow of the subroutine and the subroutines called are as follows.









4.2 Matrix Solution

For the calculation by matrix method, the unknown factors and equations shown in section 3 are rearranged. Suffixes used hereinafter are as follows.

- NCR = Number of radial meshes in the columnar grain growth region
- NER = Number of radial meshes in the equiaxed grain growth region
- NUR = Number of radial meshes in the undisturbed region
- NCL = Number of radial meshes in the cladding region

M = NCR + NER + NUR
 = Number of radial meshes in the fuel region
 N = NCR + NER + NUR + NCL
 = Total number of radial meshes

1) Unknown factors

Radial total strains

$$\epsilon_{r_1}^T, \epsilon_{r_2}^T, \dots, \epsilon_{r_N}^T \quad [1 \sim N]$$

Circumferential total strains

$$\epsilon_{\theta_1}^T, \epsilon_{\theta_2}^T, \dots, \epsilon_{\theta_N}^T \quad [N+1 \sim 2N]$$

Radial stresses

$$\sigma_{r_1}, \sigma_{r_2}, \dots, \sigma_{r_N} \quad [2N+1 \sim 3N]$$

Circumferential stresses

$$\sigma_{\theta_1}, \sigma_{\theta_2}, \dots, \sigma_{\theta_N} \quad [3N+1 \sim 4N]$$

Axial stresses

$$\sigma_{z_1}, \sigma_{z_2}, \dots, \sigma_{z_N} \quad [4N+1 \sim 5N]$$

Radial creep strains

$$\epsilon_{r_1}^P, \epsilon_{r_2}^P, \dots, \epsilon_{r_N}^P \quad [5N+1 \sim 6N]$$

Circumferential creep strains

$$\epsilon_{\theta_1}^P, \epsilon_{\theta_2}^P, \dots, \epsilon_{\theta_N}^P \quad [6N+1 \sim 7N]$$

Axial creep strains

$$\epsilon_{z_1}^P, \epsilon_{z_2}^P, \dots, \epsilon_{z_N}^P \quad [7N+1 \sim 8N]$$

Average radial creep strains for fuel and cladding

$$\overline{\epsilon_{rf}^P}, \overline{\epsilon_{rc}^P} \quad [8N+1 \sim 8N+2]$$

Average circumferential creep strains for fuel and cladding

$$\overline{\epsilon_{\theta f}^P}, \overline{\epsilon_{\theta c}^P} \quad [8N+3 \sim 8N+4]$$

Average axial creep strains for fuel and cladding

$$\overline{\epsilon_{zf}^P}, \overline{\epsilon_{zc}^P} \quad [8N+5 \sim 8N+6]$$

Average axial total strains for fuel and cladding

$$\overline{\epsilon_{zf}^T}, \overline{\epsilon_{zc}^T} \quad [8N+7 \sim 8N+8]$$

Radial displacement

$$u_1, u_2, \dots, u_{N+2} \quad [8N+9 \sim 9N+10]$$

Radial direction contact pressures

$$P_1, P_2, \dots, P_{N+1} \quad [9N+11 \sim 10N+11]$$

Friction force between fuel and cladding

$$F \quad [10N+12]$$

2) Equations

The equations mentioned in section 3 are rearranged to gather the unknown factors at the left sides.

a) Constructive relations

Equations (19) and (20) are arranged as follows.

$$\epsilon_{rj}^T - \frac{1}{E_j} \sigma_{rj} + \frac{\nu_j}{E_j} \sigma_{\theta j} + \frac{\nu_j}{E_j} \sigma_{zj} - \epsilon_{rj}^P = \alpha T_j + \epsilon_{rj}^S$$

$$(j=1, N) \quad [1 \sim N]$$

$$\epsilon_{\theta j}^T + \frac{\nu_j}{E_j} \sigma_{rj} - \frac{1}{E_j} \sigma_{\theta j} + \frac{\nu_j}{E_j} \sigma_{zj} - \epsilon_{\theta j}^P = \alpha T_j + \epsilon_{\theta j}^S$$

$$(j=1, N) \quad [N+1 \sim 2N]$$

b) Creep law

The equations of (19), (20) and (21) are rewrote using the relation,

$$\Delta \epsilon^P = \epsilon_k^P - \epsilon_{ko}^P \quad (k=r, \theta, z).$$

$$\frac{\Delta \epsilon_P^j}{\sigma_{ej}} \sigma_{rj} - \frac{\Delta \epsilon_P^j}{2\sigma_{ej}} \sigma_{\theta j} - \frac{\Delta \epsilon_P^j}{2\sigma_{ej}} \sigma_{zj} - \epsilon_{rj}^P = -\epsilon_{roj}^P$$

$$(j=1, N) \quad [2N+1 \sim 3N]$$

$$-\frac{\Delta \epsilon_P^j}{2\sigma_{ej}} \sigma_{rj} + \frac{\Delta \epsilon_P^j}{\sigma_{ej}} \sigma_{\theta j} - \frac{\Delta \epsilon_P^j}{2\sigma_{ej}} \sigma_{zj} - \epsilon_{\theta j}^P = -\epsilon_{\theta oj}^P$$

$$(j=1, N) \quad [3N+1 \sim 4N]$$

$$-\frac{\Delta \epsilon_p^j}{2\sigma_{ej}} \sigma_{rj} - \frac{\Delta \epsilon_p^j}{2\sigma_{ej}} \sigma_{\theta j} + \frac{\Delta \epsilon_p^j}{\sigma_{ej}} \sigma_{zj} - \epsilon_{zj}^p = -\epsilon_{zoj}^p$$

(j=1,N) [4N+1 ~ 5N]

c) Averaging the creep strains

In order to get the mean values of creep strains in the fuel and cladding regions, equation (42) is used.

Mean radial creep strain in the fuel region is calculated by the equation

$$\overline{\epsilon_{rf}^p} = \frac{\int_{r_i}^{r_o} \epsilon_r^p r dr}{\int_{r_i}^{r_o} r dr}$$

Therefore,

$$(r_{o1}^2 - r_{i1}^2) \epsilon_{r1}^p + (r_{o2}^2 - r_{i2}^2) \epsilon_{r2}^p + \dots + (r_{oM}^2 - r_{iM}^2) \epsilon_{rM}^p - (r_{oM}^2 - r_{i1}^2) \overline{\epsilon_{rf}^p} = 0$$

[5N + 1]

Using the same equation,

$$(r_{o1}^2 - r_{i1}^2) \epsilon_{\theta 1}^p + \dots + (r_{oM}^2 - r_{iM}^2) \epsilon_{\theta M}^p - (r_{oM}^2 - r_{i1}^2) \overline{\epsilon_{\theta f}^p} = 0$$

[5N + 2]

$$(r_{o1}^2 - r_{i1}^2) \epsilon_{z1}^p + \dots + (r_{oM}^2 - r_{iM}^2) \epsilon_{zM}^p - (r_{oM}^2 - r_{i1}^2) \overline{\epsilon_{zf}^p} = 0$$

[5N + 3]

In the cladding region,

$$(r_{oM+1}^2 - r_{iM+1}^2) \epsilon_{rM+1}^p + \dots + (r_{oN}^2 - r_{iN}^2) \epsilon_{rN}^p - (r_{oN}^2 - r_{iM+1}^2) \overline{\epsilon_{rc}^p} = 0$$

[5N + 4]

$$(r_{oM+1}^2 - r_{iM+1}^2) \epsilon_{\theta M+1}^p + \dots + (r_{oN}^2 - r_{iN}^2) \epsilon_{\theta N}^p - (r_{oN}^2 - r_{iM+1}^2) \overline{\epsilon_{\theta c}^p} = 0$$

[5N + 5]

$$(r_{oM+1}^2 - r_{iM+1}^2) \epsilon_{zM+1}^P + \dots + (r_{oN}^2 - r_{iN}^2) \epsilon_{zN}^P - (r_{oN}^2 - r_{iM+1}^2) \overline{\epsilon_{zc}^P} = 0$$

$$[5N + 6]$$

d) Radial displacement

From the equations of (19), (20), (21), (23) and (24), stresses (σ_r , σ_θ , σ_z) and displacement ($u(r)$) are solved using the boundary conditions described by equations (30) and (31).

Then, the average displacement of each region is calculated using the mean radius of each region ($\bar{r}$).

$$\bar{r} = \frac{\int_{r_i}^{r_o} r \cdot r dr}{\int_{r_i}^{r_o} r dr} = \frac{2}{3} \cdot \frac{r_o^2 + r_o r_i + r_i^2}{r_o + r_i}$$

The displacement of the inner radius of a region is given by the equation;

$$\begin{aligned} u_i + P_i \times \frac{r_i^2}{2G(r_o^2 - r_i^2)} \left\{ -\frac{r_o^2}{r} - (1-2\nu)\bar{r} \right\} \\ + P_o \times \frac{r_o^2}{2G(r_o^2 - r_i^2)} \left\{ \frac{r_i^2}{r} + (1-2\nu)\bar{r} \right\} \\ + \overline{\epsilon_r^T} \times (\bar{r} - r_i) \\ + \overline{\epsilon_z^T} \times \nu \bar{r} \\ + \overline{\epsilon_r^P} \times \left\{ -\frac{1-2\nu}{2} \cdot \bar{r} - Y_1 + Y_2 \right\} \\ + \overline{\epsilon_\theta^P} \times \left\{ -\frac{1-2\nu}{2} \cdot \bar{r} + Y_1 - Y_2 \right\} \\ = \overline{\epsilon_{r_o}^T} \times \left\{ -r_i + \frac{\bar{r}}{2} - Y_1 + Y_2 \right\} \\ + \overline{\epsilon_{\theta o}^T} \times \left\{ -\frac{\bar{r}}{2} + Y_1 - Y_2 \right\} \\ + \overline{\alpha T} \times \{(1+\nu)\bar{r}\} \end{aligned}$$

$$- \overline{\epsilon_r^s} \times \left\{ -\frac{\overline{r}}{2} - Y_1 + Y_2 \right\}$$

$$- \overline{\epsilon_\theta^s} \times \left\{ -\frac{\overline{r}}{2} + Y_1 - Y_2 \right\}$$

$$- \overline{\epsilon_z^s} \times \{-v\overline{r}\}$$

[5N+7 ~ 6N+6]

where,

$$Y_1 \equiv \frac{1-2\nu}{1-\nu} \cdot \frac{\overline{r}}{2} \ln \frac{\overline{r}}{r_i}$$

$$Y_2 \equiv \frac{\{r_i^2 + (1-2\nu)\overline{r}^2\}r_o^2}{2(1-\nu)\overline{r}(r_o^2-r_i^2)} \ln \frac{r_o}{r_i}$$

The displacement of the outer radius of a region is given by the equation;

$$u_o + P_i \times \frac{r_i^2}{2G(r_o^2-r_i^2)} \left\{ -\frac{r_o^2}{r} - (1-2\nu) \frac{\overline{r}}{r} \right\}$$

$$+ P_o \times \frac{r_o^2}{2G(r_o^2-r_i^2)} \left\{ \frac{r_i^2}{r} + (1-2\nu) \frac{\overline{r}}{r} \right\}$$

$$+ \overline{\epsilon_r^T} \times (\overline{r} - r_o)$$

$$+ \overline{\epsilon_r^T} \times v\overline{r}$$

$$+ \overline{\epsilon_r^P} \times \left\{ -\frac{1-2\nu}{2} \frac{\overline{r}}{r} - Y_1 + Y_2 \right\}$$

$$+ \overline{\epsilon_\theta^P} \times \left\{ -\frac{1-2\nu}{2} \frac{\overline{r}}{r} + Y_1 - Y_2 \right\}$$

$$= \overline{\epsilon_{r_0}^T} \times \left\{ -r_o + \frac{\overline{r}}{2} - Y_1 + Y_2 \right\}$$

$$+ \overline{\epsilon_{\theta_0}^T} \times \left\{ -\frac{\overline{r}}{2} + Y_1 - Y_2 \right\}$$

$$+ \overline{\alpha T} \times \{(1+\nu) \overline{r}\}$$

$$- \overline{\epsilon_r^s} \times \left\{ -\frac{\overline{r}}{2} - Y_1 + Y_2 \right\}$$

$$-\overline{\epsilon}_{\theta}^s \times \left\{ -\frac{\overline{r}}{2} + Y_1 - Y_2 \right\}$$

$$-\overline{\epsilon}_z^s \times \{ -\nu \overline{r} \}$$

[6N+7 ~ 7N+6]

e) Boundary conditions

As shown in the section 3, boundary conditions of pressure are given as follows.

At the fuel central hole,

$$P_1 = P_{cv} . \quad [7N + 7]$$

At the cladding outer surface,

$$P_{N+1} = P_{cool} . \quad [7N + 8]$$

f) Axial displacement

According to the assumption of plane-strain analysis, axial displacements are independent of radius. Therefore, the axial displacement in the fuel region is given by averaging the displacements of all regions:

$$\begin{aligned} & (1+\nu)\overline{\epsilon}_{zf}^T + \nu\overline{\epsilon}_{rf}^p + \nu\overline{\epsilon}_{\theta}^p \\ & - \overline{\epsilon}_{zf}^p - \frac{\nu}{G} \cdot \frac{r_f^2}{r_f^2 - r_c^2} P_{fc} + \frac{\Delta L_f}{2G\pi(r_f^2 - r_c^2)} \cdot F \\ & = -\frac{1}{2G} [P_{p1} + \frac{\sigma_f g V_f}{\pi(r_f^2 - r_c^2)} (1 - \frac{Z}{L_f})] \\ & + \frac{\nu}{G} (1 - \frac{r_f^2}{r_f^2 - r_c^2}) P_{cv} + \frac{(1+\nu)}{r_f^2 - r_c^2} \sum_{j=1}^M (r_{oj}^2 - r_{ij}^2) \overline{\alpha T}_j \\ & + (1+\nu) \overline{\epsilon}_{zf}^s \end{aligned} \quad [7N + 9]$$

In the same manner, the axial displacement in the cladding region is given by the equation,

$$\begin{aligned} & (1+\nu) \overline{\epsilon}_{zc}^T + \nu\overline{\epsilon}_{rc}^p + \nu\overline{\epsilon}_{\theta c}^p - \overline{\epsilon}_{zc}^p \\ & + \frac{\nu}{G} \cdot \frac{r_{ci}^2}{r_{co}^2 - r_{ci}^2} \cdot P_{fc} - \frac{\Delta L}{2\pi G(r_{co}^2 - r_{ci}^2)} \cdot F \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{2\pi G(r_{co}^2 - r_{ci}^2)} \{ -P_T - \pi(r_{co}^2 P_c - r_{ci}^2 P_p) \\
 &\quad - \rho_c g V_c (1 - \frac{z}{L_f}) \} + \frac{(1 + \nu)}{(r_{co}^2 - r_{ci}^2)} \sum_j (r_{oj}^2 - r_{ij}^2) \overline{\alpha T_j} \\
 &\quad + \frac{\nu}{G} \cdot \frac{r_{co}^2}{r_{co}^2 - r_{ci}^2} P_c + (1 + \nu) \overline{\epsilon_{zc}^s} \quad [7N + 10]
 \end{aligned}$$

g) Radial stresses

Stresses calculated from equations (19), (20), (21), (23) and (24) are the functions of radius. Therefore, the mean stress of each region is given by averaging the stress function as follows.

$$\begin{aligned}
 \overline{\sigma_r} &= \frac{\int_{r_i}^{r_o} \sigma_r r dr}{\int_{r_i}^{r_o} r dr} \\
 \overline{\sigma_r} &+ \frac{r_o^2}{r_o^2 - r_i^2} \cdot (1 - \frac{2r_i \ln \frac{r_o}{r_i}}{r_o^2 - r_i^2}) \cdot P_o \\
 &- \frac{r_o^2}{r_o^2 - r_i^2} \cdot (1 - \frac{2r_o^2 \ln \frac{r_o}{r_i}}{r_o^2 - r_i^2}) \cdot P_i \\
 &+ \frac{G}{2(1-\nu)} \{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \} \cdot \overline{\epsilon_r^p} \\
 &- \frac{G}{2(1-\nu)} \{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \} \cdot \overline{\epsilon_\theta^p} \\
 &= \frac{G}{2(1-\nu)} \{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_i}{r_o})^2}{(r_o^2 - r_i^2)^2} \} \cdot \overline{\epsilon_{ro}^T} \\
 &- \frac{G}{2(1-\nu)} \{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \} \cdot \overline{\epsilon_{\theta o}^T} \\
 &- \frac{G}{2(1-\nu)} \{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \} \cdot \overline{\epsilon_r^s}
 \end{aligned}$$

$$+ \frac{G}{2(1-\nu)} \left\{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \right\} \cdot \bar{\epsilon}_\theta^s \quad [7N+11 \sim 8N+10]$$

h) Circumferential stresses

Circumferential stresses are averaged by the same method.

$$\begin{aligned} \bar{\sigma}_\theta &+ \frac{r_o^2}{r_o^2 - r_i^2} \cdot \left(1 + \frac{2r_i^2 \ln \frac{r_o}{r_i}}{r_o^2 - r_i^2} \right) \cdot P_o \\ &- \frac{r_i^2}{r_o^2 - r_i^2} \cdot \left(1 + \frac{2r_o^2 \ln \frac{r_o}{r_i}}{r_o^2 - r_i^2} \right) \cdot P_i \\ &- \frac{G}{2(1-\nu)} \left\{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \right\} \cdot \bar{\epsilon}_r^p \\ &+ \frac{G}{2(1-\nu)} \left\{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \right\} \cdot \bar{\epsilon}_\theta^p \\ &= - \frac{G}{2(1-\nu)} \left\{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \right\} \cdot \bar{\epsilon}_{ro}^T \\ &+ \frac{G}{2(1-\nu)} \left\{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \right\} \cdot \bar{\epsilon}_{\theta o}^T \\ &+ \frac{G}{2(1-\nu)} \left\{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \right\} \cdot \bar{\epsilon}_r^s \\ &- \frac{G}{2(1-\nu)} \left\{ 1 - \frac{4r_i^2 r_o^2 (\ln \frac{r_o}{r_i})^2}{(r_o^2 - r_i^2)^2} \right\} \cdot \bar{\epsilon}_\theta^s \quad [8N+11 \sim 9N+10] \end{aligned}$$

i) Axial stresses

By averaging the stress distribution $u(r)$ in each region;

$$\begin{aligned} \bar{\sigma}_z &+ 2\nu \cdot \frac{r_o^2}{r_o^2 - r_i^2} \cdot P_o - 2\nu \cdot \frac{r_i^2}{r_o^2 - r_i^2} \cdot P_i \\ &- 2G\nu \cdot \bar{\epsilon}_r^p - 2G\nu \cdot \bar{\epsilon}_\theta^p + 2G \cdot \bar{\epsilon}_z^p - 2G(1+\nu) \cdot \bar{\epsilon}_z^T \\ &= -2G(1+\nu) \cdot \bar{\alpha}^T - 2G(1+\nu) \cdot \bar{\epsilon}_z^T \quad [9N+11 \sim 10N+10] \end{aligned}$$

j) Boundary conditions between fuel and cladding

Boundary conditions between fuel and cladding are changed following the relationship between them.

(i) When gap exists

Contact pressure

$$P_{fc} = P_{pl} \quad [10N + 11] \quad (1)$$

Friction force

$$F = 0 \quad [10N + 12] \quad (1)$$

(ii) When gap does not exist (stick condition)

Strain

$$\bar{\epsilon}_{zf} = \bar{\epsilon}_{zc} \quad [10N + 11] \quad (2)$$

Displacement

$$u_{o,M} - u_{i,M+1} = r_{co} - r_f \quad [10N + 12] \quad (2)$$

(iii) When gap does not exist (slip condition)

At first, code calculates by the assumption of gap existence, and if the result shows no gap, code assumes the stick condition. Then if the friction force F , which is calculated by stick assumption, is greater than the value estimated by the contact pressure between fuel and cladding;

$$F > 2\pi\mu_s f_f P_{fc}$$

where μ_s is static friction constant, slip assumption is adopted.

Friction force

$$F = 2\pi\mu_d r_f P_{fc} \quad [10N + 11] \quad (3)$$

Displacement

$$u_{o,M} - u_{i,M+1} = r_{ci} - r_f \quad [10N + 12] \quad (3)$$

From these [10N + 12] equations, the [10N + 12] unknown factors are calculated using matrix method.

5. Material Properties

Material properties used in ACTIVE-II are defined in the subroutine PROPTY, except fuel thermal conductivity and clad-coolant heat conductance.

As the code ACTIVE-II can deal with not only sodium cooled fast breeder reactor fuels, but also the fuels of heavy water reactors, of pressurized water reactors and of boiling water reactors, one must select the appropriate properties for a certain reactor type. Hereinafter, the material properties are described for each reactor type, separately.

The units used in the code are standardized in CGS unit, but some of them have units in BTU or engineering units because of the data.

5.1 Material Properties for Sodium Cooled Fast Breeder Reactor Fuels

1) Coolant-cladding heat conductance³⁾

$$h_c = \frac{K_{Na}}{D_e} [5.0 + 0.025 \left(\frac{D_e \cdot W \cdot C_p}{A_f \cdot K_{Na}} \right)^{0.8}] \quad (51)$$

where, h_c = heat conductance (Btu/hr/ft²/°F),

K_{Na} = thermal conductivity of sodium,

$$= 54.306 - 1.878 \times 10^{-2}T + 2.0914 \times 10^{-6}T^2, \quad (52)$$

T = temperature of coolant (°F),

D_e = equivalent flow diameter (ft),

$$= \frac{4A_f}{P},$$

A_f = coolant flow area (ft²),

P = wet perimeter (ft),

W = sodium mass flow rate (lb_m/hr),

$$= \frac{L_f \times q_{av}}{\int_{T_{in}}^{T_{out}} C_p dT},$$

L_f = fuel column length,

q_{av} = average linear heat rating (Btu/in/hr),

and, C_p = heat capacity of sodium (Btu/lb_m/ft),

$$= 0.3457 - 0.792 \times 10^{-4}T + 3.41178 \times 10^{-8}T^2. \quad (53)$$

2) Fuel properties

Usually, fuels for fast breeder reactors are plutonium-uranium

mixed oxide, therefore properties of (20w/oPuO₂-80w/oUO₂) fuel are chosen for the representatives.

a) Thermal conductivity

Option (NOK) = 1

$$K_f = (-0.018 + \frac{54}{T} + 1.75 \times 10^{-12}T) \times \frac{1-P}{1+\beta P} \text{ (W/cm}^\circ\text{C)}$$

where, T = temperature (°K),

P = porosity,

and $\beta = 1$.

Sales' equation⁴⁾

Option (NOK) = 2

$$K_f = 0.011 + \frac{1}{(0.485-0.4465 \cdot D)T} \text{ (W/cm}^\circ\text{C)}$$

where, T = temperature (°C),

and D = ratio of density to the theoretical density,

= 0.99 for columnar grain region,

= 0.965 for equiaxed grain region.

Baily-Asamoto's equation⁵⁾

Option (NOK) = 3

$$K_f = \left\{ \frac{1}{(3.11+355.0 \cdot x) + 0.0272 \times T} + 5.39 \times 10^{-13} \times T^3 \right\} \\ \times \frac{\left(\frac{1-P}{1+\beta P} \right)}{\left(\frac{1-0.05}{1+\beta \times 0.05} \right)} \text{ (W/cm}^\circ\text{C)}$$

where, T = temperature (°K),

P = porosity,

X = 2-O/M (stoichiometry)

and $\beta = 1$.

Biancheria's equation⁶⁾

b) Thermal expansion rate⁷⁾

Thermal expansion rate is assumed to be the same for all three regions (i.e. columnar grain region, equiaxed grain region and

undisturbed region).

$$\alpha_f = 8.192 \times 10^{-6} + 2.430 \times 10^{-9} \times (T - 20) \quad (1/^{\circ}\text{C})$$

where, T = temperature ($^{\circ}\text{C}$).

c) Young's modulus of elasticity⁷⁾

$$E_f = 3.59 \times 10^7 \times (1 - 2.35 \times P) \left(1 - \frac{T}{22,200}\right) \quad (\text{psi})$$

where, T = temperature ($^{\circ}\text{F}$),

and P = porosity.

d) Poisson's ratio⁷⁾

$$\nu_f = \frac{0.317 \times (1 - 0.46 \times P) \times (22,200 - T) + 0.5 \times (T - 77)}{22,200 - 77}$$

where, T = temperature ($^{\circ}\text{F}$),

and P = porosity.

e) Shear modulus

$$G_f = \frac{E_f}{2(1 + \nu_f)}$$

f) Creep rate⁷⁾

$$\dot{\epsilon} = A_0 \times e^{-\frac{Q_0}{RT}} \times \left(1 + \frac{\psi}{\psi_0}\right) \times \sigma_e^m + A_1 \times e^{-\frac{Q_1}{RT}} / (d^2 + C \times \phi) \times \sigma^n \quad (1/\text{hr})$$

where, $A_0 = 1.83 \times 10^{-3}$,

$A_1 = 2.82 \times 10^7$,

$Q_0 = 140,000$ (cal/mole),

$Q_1 = 110,000$ (cal/mole),

ψ = fission density (fissions/sec/cc),

$\psi_0 = 3.0 \times 10^{12}$ (fissions/sec/cc),

σ_e = equivalent stress (psi),

$m = 4.5$,

$n = 1$,

$d = 10.0$ (μ) grain size,

$C = 6.0 \times 10^{-23}$,

ϕ = fast neutron flux (n/cm²/sec) (E>0.1 MeV),
and R = 1.1034 (cal/mole/°R).

g) Swelling rate

Fuel swelling is assumed to be caused by fission gases and solid fission products retained in each region.

Solid fission products⁷⁾

Swelling rate caused by solid fission products are calculated based on the volume of fission products per fission.

$$1.147 \times 10^{-23} \text{ (cm}^3\text{/fission)}$$

Gaseous fission products⁷⁾

In the undisturbed region, gaseous fission products are assumed to be incompressible, and to have constant volume.

$$39.15 \text{ (cm}^3\text{/mole)}$$

In the columnar grain region and equiaxed grain region, gaseous fission products are assumed to be controlled by ideal gas law as follows.

$$\left(\frac{\Delta V}{V}\right)_i^{\text{gas}} = \frac{n_i R \bar{T}_i}{V_i (P_i + P_i')}$$

where,

ΔV_i = volume change in the region i,

V_i = initial volume in the region i,

n_i = moles of fission gas retained in the region i,

$\bar{T}_i$ = mean temperature in the region i,

$P_i = \frac{1}{3} (\sigma_r^i + \sigma_\theta^i + \sigma_z^i)$ average stress in the region i,

P_i' = correction factor related to the surface tension of fission gas bubbles,

and R = gas constants (cal/g/mole/°K).

h) Hot pressing

$$\Delta e_i^{\text{hp}} = - \left[\frac{40 \Omega b_i D_{\text{oe}}^{-Q/RT}}{K T d^2} \right] \cdot P_i \cdot \left(1 - \frac{\rho_i}{\rho_{\text{th}}} \right) \Delta t$$

where , $\Delta e_i^{hp} = \left(\frac{\Delta V}{V}\right)_i^{hp}$
 $=$ volumetric strain due to hot pressing in the region i,
 $\rho_{th} = \rho_o / (1 - 3\alpha T)$
 $=$ theoretical density of fuel (g/cm³),
 $T_i =$ temperature (°K),
 $\alpha =$ thermal expansion rate (1/°C),
 $\Delta t =$ time interval (sec),
 $\Omega = 4.0 \times 10^{-23}$ (cm³)
 $=$ vacancy volume,
 $b_i = 4$
 $=$ stress concentration factor in the region i,
 $D_o = 6.8 \times 10^{-5}$ (cm²/sec)
 $=$ diffusion coefficient,
 $K = 1.38 \times 10^{-16}$ (erg/°K)
 $=$ Boltzmann's constant,
 $d = 10 \times 10^{-4}$
 $=$ average grain size,
 and $Q = 98.3$ (Kcal/g/mole)
 $=$ activation energy.

In the program, these factors are combined into two constants, C and QPK⁷⁾.

$$C = \frac{40\Omega b_i D_o}{K d^3}$$

$$= \frac{40 \times (4 \times 10^{-23}) \times 4 \times (6.8 \times 10^{-5})}{(1.38 \times 10^{-16}) \times (10^{-3})^2} \times 3600 \times (9.8 \times 10^5)$$

$$= 11.13 \times 10^6 \text{ (cm}^2 \cdot \text{°K/hr/Kg)}$$

$$QPK = \frac{Q}{R} = \frac{98300}{1.987}$$

$$= 49,600 \text{ (1/°K)}$$

i) Theoretical density

Theoretical density of pellet is given as a function of plutonium enrichment.

$$\rho_{th} = 10.96 + 0.5 \times C_{pu} \quad (\text{g/cm}^3)$$

where, C_{pu} = plutonium enrichment.

j) Fission gas release rate⁷⁾

Fission gas release rate is given as a function of temperature,

$$G_r(i) = \exp(-4.48 - \frac{8000}{T_i}) \quad (1/\text{hr})$$

where, T_i = temperature ($^{\circ}\text{K}$).

But if the swelling strain caused by fission gas exceeds 25%, gas release rate is accelerated by the following equation.

$$\begin{aligned} G_r^m(i) &= G_r(i) + [(\frac{\Delta V}{V})_{i\text{-gas}} - 0.25] \times 100 \\ &\quad \times [G_r(\text{columnar}) - G_r(i)] \end{aligned}$$

where, $G_r(\text{columnar})$ = gas release rate in the columnar grain region,

and $(\frac{\Delta V}{V})_{i\text{-gas}}$ = swelling strain caused by gas.

By the way, number of atoms per fission (n_a) and number of fissions per Kw-hr (n_f), which are used in the program, are as follows.

$$n_a = 0.27 \quad (\text{n/fission})$$

$$n_f = 1.116 \times 10^{17} \quad (\text{fissions/kw-hr})$$

3) Fuel-cladding interface

a) Gap conductance between fuel and cladding⁸⁾

Gap conductance is given by the function of gap width.

$$h_g = 1873 \times e^{-53.25 \times \Delta r} \quad (\text{Btu/ft}^2 \cdot \text{hr} \cdot ^{\circ}\text{F})$$

where, Δr = gap width (mm).

b) Friction factor between fuel and cladding⁷⁾

Friction factor is used to settle the gap conditions.

$$\text{Static friction factor} : \mu_s = 0.8$$

$$\text{Dynamic friction factor} : \mu_d = 0.6$$

4) Cladding properties

SUS 316 stainless steel (almost equivalent AISI-316 S.S.) is chosen for the representative of the cladding of fast breeder reactors.

a) Thermal conductivity

$$k_c = 0.130 + 1.36 \times 10^{-4} \times T \quad (\text{W/cm} \cdot ^\circ\text{C})$$

where, T = temperature ($^\circ\text{C}$).

b) Thermal expansion rate⁹⁾

$$\alpha_c = 17.0 \times 10^{-6} + 4.25 \times 10^{-9} \times T \quad (1/^\circ\text{C})$$

where, T = temperature ($^\circ\text{C}$).

c) Poisson's ratio⁷⁾

$$\nu_c = 0.3015 + 0.850 \times 10^{-4} \times T$$

where, T = temperature ($^\circ\text{C}$).

d) Young's modulus of elasticity¹⁰⁾

$$E = 2.0793 \times 10^6 - 8.8592 \times 10^2 \times T \quad (\text{Kg/cm}^2)$$

where, T = temperature ($^\circ\text{C}$)

3) Creep rate

$$\text{Option (NOC)} = -1^7)$$

$$\dot{\epsilon} = A \times e^{-\frac{Q}{RT}} \times \sigma^m + B \times \phi \times \sigma^n \quad (1/\text{hr})$$

where,

$$A = 2.7 \times 10^{-11},$$

$$Q = 95,000 \text{ (cal/mole)},$$

$$R = 1.1034 \text{ (cal/mole)},$$

$$T = (^\circ\text{R}),$$

$$m = 7,$$

$$B = C_e + \frac{A_e}{B_e} \times e^{\frac{-\phi_t}{B_e}},$$

$$A_e = 1.0 \times 10^{-8},$$

$$\begin{aligned} B_e &= 2.2 \times 10^{20}, \\ C_e &= 4.3 \times 10^{-30}, \\ \phi &= \text{neutron flux (n/cm}^2\text{/sec)}, \\ t &= \text{time (sec)}, \end{aligned}$$

and $n = 1$.

(Boltax, Biancheria & Guha's equation)

Option (NOC) = -27)

$$\dot{\epsilon} = A \times e^{-\frac{Q}{RT}} \times \sigma^m + B \times \phi \times \sigma^n \quad (1/\text{hr})$$

where,

$$\begin{aligned} A &= 2.7 \times 10^{-11}, \\ Q &= 95,000 \text{ (cal/mole)}, \\ R &= 1.1034 \text{ (cal/mole)}, \\ T &= \text{temperature (}^\circ\text{R)}, \\ m &= 7, \\ B &= C_e + \frac{A_e}{B_e} \times e^{-\frac{\phi t}{B_e}}, \\ A_e &= 1.0 \times 10^{-8} / (T/910), \\ B_e &= 2.2 \times 10^{20} \times (T/910), \\ C_e &= 4.3 \times 10^{-30} / (T/910), \\ \phi &= \text{neutron flux (n/cm}^2\text{/sec)}, \\ t &= \text{time (sec)}, \end{aligned}$$

and $n = 1$.

f) Swelling rate

Option (NOS) = -111)

$$\frac{\Delta V}{V} = [(T-40) \times 10^{-10}] (\phi t)^{2.05 - \frac{27}{\theta} + \frac{78}{\theta^2}} \exp(32.60 - \frac{5100}{T} - 0.015T) \quad (\%)$$

where,

$$\begin{aligned} \phi t &= \text{neutron fluence (n/cm}^2 \times 10^{-22}) \quad E > 0.1 \text{ MeV} \\ T &= (^\circ\text{K}) \end{aligned}$$

and $\theta = T-623$

for annealed 304 & 316 S.S

$$T = 370 \sim 500^\circ\text{C}$$

$$\phi t \leq 7 \times 10^{22} \text{ n/cm}^2$$

Option (NOH) ≤ -2

$$\frac{\Delta V}{V} = 9 \times 10^{-35} \times (\phi t)^{1.50} (4.028 - 3.712 \times 10^{-2} \times T + 1.0145 \times 10^{-4} \times T^2 - 7.879 \times 10^{-8} \times T^3)$$

where,

ϕt = neutron fluence (n/cm^2) $E > 0.1$ MeV

and T = temperature ($^{\circ}K$)

for 20% cold worked 316 S.S.

$$\phi t \leq 5 \times 10^{22} \text{ n/cm}^2$$

5.2 Material Properties for Heavy Water Reactor Fuels

Heavy Water Reactor means a heavy water moderated and boiling light water cooled reactor which is now being developed by PNC.

1) Coolant-cladding heat conductance

Heat conductance equation proposed by Jens and Lottes¹²⁾ is used.

$$\Delta T_{\text{film}} = 7.93 \times q^{1/4} \times \exp\left(-\frac{P}{63.2}\right) \quad (^{\circ}C)$$

where, q = surface heat flux (w/cm^2),

and P = coolant pressure.

Film coefficient is calculated using the temperature difference given by the above equation.

$$h_c = \frac{q}{2\pi r} \quad (w/cm^2/^{\circ}C)$$

where, r = clad outer radius (cm).

2) Fuel properties

Uranium dioxide or plutonium slightly enriched mixed oxide fuels are used for heavy water reactor. Therefore, the properties are the same as those used for uranium dioxide pellets.

a) Thermal conductivity

Option (NOK) = 3

Thermal conductivity of 95% T.D. fuel is,

$$k_f(0.05) = \frac{38.24}{T+129.4} + 4.79 \times 10^{-13} T^3 \quad (\text{W/cm} \cdot ^\circ\text{C})$$

where, $T = ^\circ\text{K}$

(Old Lyon's equation¹³⁾)

Modification for density is,

$$k_f(P) = \frac{1-P}{1+P(\alpha-1)} \times k_f(0)$$

where, $k_f(P)$ = thermal conductivity of porosity P fuel,

$\alpha = 1.5$,

$k_f(0)$ = thermal conductivity of 100% T.D. fuel,

and P = porosity.

(Biancheria's equation¹⁴⁾)

Option (NOK) = 4

Thermal conductivity of 95% T.D. fuel is,

$$k_f(0.05) = \frac{38.24}{T+129.4} + 6.1256 \times 10^{-13} T^3 \quad (\text{W/cm} \cdot ^\circ\text{C})$$

where, T = temperature ($^\circ\text{K}$).

(New Lyon's equation¹⁵⁾)

Modification for density is the same as option 3.

b) Thermal expansion rate

Thermal expansion rate is assumed to be the same for all three regions (i.e. columnar grain region, equiaxed grain region and undisturbed region).

$$\alpha_f = 9.61 \times 10^{-6} + 1.57 \times 10^{-9} \times T \quad (1/^\circ\text{C})$$

where, T = temperature ($^\circ\text{K}$).

c) Other properties of fuel

Other properties of fuel for heavy water reactor are the same as those of fast breeder reactor fuel.

3) Fuel-cladding interface

Properties related to fuel-cladding interface, such as gap con-

ductance and friction factor, are assumed to be the same as those of fast breeder reactor fuel.

4) Cladding properties

Zircaloy-2 is chosen for the representative of the cladding of advanced thermal reactors.

a) Thermal conductivity¹⁶⁾

$$k_c = 0.121 + 1.24 \times 10^{-4} \times T \quad (\text{W/cm} \cdot ^\circ\text{C})$$

where, $T = ^\circ\text{C}$.

b) Thermal expansion rate¹⁷⁾

$$\alpha_c = 5.675 \times 10^{-6} + 1.7 \times 10^{-9} \times T \quad (1/^\circ\text{C})$$

where, $T = ^\circ\text{C}$.

c) Poisson's ratio

$$\nu_c = 0.325 + 2.14 \times 10^{-4} \times T$$

where, $T = ^\circ\text{C}$.

d) Young's modulus of elasticity

$$E_c = 9.898 \times 10^5 - 9.265 \times 10^2 \times T \quad (\text{Kg/cm}^2)$$

where, $T = ^\circ\text{C}$.

e) Creep rate

$$\text{Option (NOC)} = -1$$

$$\dot{\epsilon} = 4 \times 10^{-27} \sigma_e \phi (T-160) \quad (1/\text{hr})$$

where,

$$T = ^\circ\text{C},$$

$$\phi = \text{fast neutron flux (m/cm}^2\text{/sec)},$$

$$E > 1 \text{ MeV}$$

and, $\sigma_e = \text{equivalent stress (psi)},$

Option (NOC) = -2

$$\dot{\epsilon} = 1.02 \times 10^{-13} \exp\left(\frac{14000}{RT}\right) \phi^{0.85} \sinh(1.63 \times 10^{-1} \sigma_e) \quad (1/\text{hr})$$

where,

$T = ^\circ\text{K}$,

$\sigma_e = \text{equivalent stress (Kg/mm}^2\text{)}$,

$\phi = \text{fast neutron flux (n/cm}^2\text{/sec)}$,

$E > 1 \text{ MeV}$

$R = 1.985 \text{ (cal/mole} \cdot ^\circ\text{K)}$,

and $Q = 14,000 \text{ (cal/mole)}$

$= \text{activation energy.}$

f) Swelling rate

Swelling rate is assumed to be zero in the code.

5.3 Material Properties for Pressurized Water Reactor Fuels

Material properties for pressurized water reactors are almost the same as those for heavy water reactor, expect coolant-cladding heat conductance.

1) Coolant-cladding heat conductance

Coolant-cladding heat conductance is calculated by Dittus-Boettler's equation¹⁸⁾ as follows.

$$N_u = 0.023 \times R_e^{0.8} \times P_r^{0.4}$$

where,

$N_u = \text{Nusselt number,}$

$R_e = \text{Reynolds number,}$

and $P_r = \text{Prandtle number,}$

5.4 Material Properties for Boiling Water Reactor Fuels

Material properties for boiling water reactors are assumed to be the same as those for heavy water reactors as mentioned before.

6. Input and Output Format

Using a sample calculation, core storage, input parameter preparation, input format, output format and plotter output are described in this chapter.

6.1 Core Storage

As mentioned in chapter 4, this code uses variable dimension method. Therefore, the total core storage size must be given in the MAIN routine according to the job size.

The required amount of storage space for a problem is determined with the following parameters.

NN = number of axial sections. (input)

NCR = number of meshes into which the columnar grain region is subdivided. (input)

NER = number of meshes into which the equiaxed grain region is subdivided. (input)

NUR = number of meshes into which the undisturbed region is subdivided. (input)

NFR = NCR + NER + NUR. (total number of meshes into which the fuel is subdivided.)

NCL = number of meshes into which the cladding is subdivided. (input)

NTR = NFR + NCL = NCR + NER + NUR + NCL.

NP = number of regions (concentric cylindrical shells) of equal mass into which the fuel is subdivided for plutonium migration. (input)

NBUF = storage size (in words) required for storing plotting informations. The plotting informations are written to logical unit every NBUF steps. (input)

If NBUF = 0, then the code sets NBUF to 50.

- 1) Core storage required for problem data, excluding those for matrix. (in words)

$$IA = 13NFR + NUR + NP + 1 + NN (41NTR + NFR + NUR + 8NP + 101) \\ + \begin{cases} 0 & \text{<without plotting>} \\ NBUF(3NP+19) & \text{<with plotting>} \end{cases}$$

The last term $NBUF(3NP+19)$, is added to IA when the plotting is required.

- 2) Core storage required for data only for matrix. (in words)

$$IMA = 1332NTR + 656$$

- 3) Core storage required for program (load module).

$$\text{Load module} = 240K \text{ bytes}$$

- 4) Region size in the REGION parameter used in the JOB card on IBM-360 or -370 series machines.

$$\text{REGION} = [240 + \frac{4(IA+IAM)}{1024}] \text{ K-bytes}$$

- 5) Example

$$NN=5, NCR=1, NER=1, NUR=1, NCL=1, NP=10, NBUF=50.$$

The storage requirement is obtained from the above equations as follows.

$$IA = 4241 \text{ words}$$

$$IAM = 5984 \text{ words}$$

$$\text{REGION} = 384 \text{ K (multiple of 128K)}$$

The typical MAIN program is written as below.

(See the section CONTROL CARDS)

```
//xxxx JOB (xxxx, ....) ....., REGION=384K
:
:
// EXEC FORTHCLG
//FORT.SYSIN DD *
COMMON/DIM/IA,IAM
DIMENSION A(4250), AM(5990)
IA=4250
IAM=5990
CALL ACTIVE (A, AM)
STOP
END
/*
:
:
```

6.2 Input Parameter Preparation

Since ACTIVE-II code calculates with a simplified model of fuel pin, it is very convenient to make a summary of irradiation data. As mentioned below, the irradiation data summary includes the information of fuel pin fabrication and irradiation conditions.

Table 6.1 shows the abstract of the irradiation, and the purpose of irradiation, irradiation abstracts and references are summarized.

Table 6.2 shows the fuel pin summary of before irradiation. Plenum volume of item 1-7 means the volume available for gas retaining, but excluding pellet-cladding gap volume of core region. The pellet-cladding gap volume of core region is automatically calculated and added to the plenum volume in ACTIVE-II. Adsorbed gas of item 1-10 includes moisture in the pellets, and coolant flow area per fuel pin of item 1-11 means the equivalent flow area around the fuel pin.

Table 6.3 shows the axial distributions of parameters before irradiation. Fuel pellet region is divided into some meshes along the axis, and the number and interval of meshes are optional. But the plenum region is allowed only at the top of the fuel pin, and its length must be calculated from the plenum volume equivalently (that is, equivalent length (L_{pl}) is calculated using the plenum volume (V_{pl}) of item 1-7 of Table 6.2 and the inner radius (r_{ci}) of cladding as follows).

$$L_{pl} = V_{pl} / (\pi \times r_{ci}^2)$$

In this sample, 4 axial meshes are required of 3 in fuel column region and 1 in plenum region. Rod power and fast flux distribution may be shown as relative values along the axial direction.

Table 6.4 shows the irradiation history. Integrated time means the time since the irradiation started, and the data are required for each time step where the irradiation conditions are changed. The conditions during the intervals are calculated interpolatedly in the code.

ACTIVE-II code usually needs the inlet and outlet temperatures of coolant, and calculates the coolant temperature distributions and coolant-cladding heat conductance using the properties of coolant and axial power distribution. But in the case of Dounrey Fast Reactor (DFR), in which the coolant is NaK and flows from top to bottom, or in the case where the temperature distributions of cladding outer surface is

calculated by other code, cladding outer surface temperatures are required instead of coolant outlet temperature.

Average linear heat rating and average fast flux mean the mean values calculated by simple averaging method along the axial meshes of fuel column.

Radii for radial profiles of rod power and fast flux mean the radii of gravity center of radial meshes which are divided using equal mass method, and the number of meshes should be the same used for plutonium migration calculation. Assuming that the density of fuel is constant, the radius of each gravity center of mesh is calculated as follows.

$$r_1 = \left(\frac{r_{fs}^2}{2N} + r_{cv}^2 \right)^{1/2}$$

$$r_{i+1} = \left(\frac{r_{fs}^2}{N} + r_i^2 \right)^{1/2} \quad (i=2,3,\dots,N)$$

where, r_i = radius of gravity center of mesh i , ($i=1,2,\dots, N$)
 r_{fs} = fuel outer radius,
 r_{cv} = radius of fuel central void or central hole,
 and N = total mesh number.

If the pellet is solid and has no central void, the equation becomes as follows.

$$r_i = \left(\frac{2i-1}{2N} \right)^{1/2} \times r_{fs}$$

For the input of ACTIVE-II, relative values are needed for the radii. The sample case shows the case that the pellet is solid and the number of meshes is 10.

Radial powers for each mesh points may be relative values, and the code normalize the power distribution as the mean value becomes unit. In this case, as EBR-II is a fast reactor, radial power distribution is flat.

Table 6.1 Irradiation Data Summary

Irradiation Data Summary		Date : July 7, 1975			
		Compiled: T. Kajiya			
		Approved: M. Katsuragawa			
•Title : G.E. Fl Group •Purpose : Investigation of mixed oxide fuel pin performance under fast neutron flux _____ _____ •S/A identification : XG01 •Fuel pin number : F1A •Irrad. duration : May 12, 1965 → May 22, 1965 •Reactor name : EBR-II •Maximum burnup : 1526 MWD/T_{OXIDE} •Max.neutron fluence : nvt (E>0.1 MeV) •Max.linear heat rate: 471.6 w/cm •Nominal values : - Core pellet : Composition 20w/oPuO₂-UO₂, Diameter 5.488 mm Density 95.3 %T.D - Blanket pellet : Composition _____, Diameter _____ mm Density %T.D - Cladding : Material 347 S.S. Inner dia. 5.588 mm, Outer dia. 6.350 mm - Pin length _____ mm, Total stack length _____ mm Lower blanket length _____ mm, Core length 361.19 mm, Upper blanket length _____ mm, Plenum length _____ mm.	•Irradiation position <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none; vertical-align: top;"> S/A position in the core </td> <td style="width: 50%; border: none; vertical-align: top;"> Pin position in the S/A </td> </tr> </table> •Notes - Material characteristics : _____ - Irradiation characteristics : _____ - Post irradiation: characteristics : _____ •Reference (1) S.A. Robin, et. al., Short-term Fast Flux (EBR-II) Irradiation of PuO₂-UO₂ Fuel Pins, GEAP-5570 (Oct., 1967) (2) _____ (3) _____ (4) _____ (5) _____			S/A position in the core	Pin position in the S/A
S/A position in the core	Pin position in the S/A				

Table 6.2 Fuel Pin Summary Before Irradiation

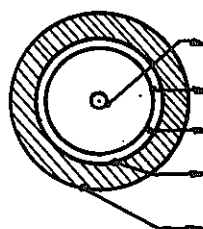
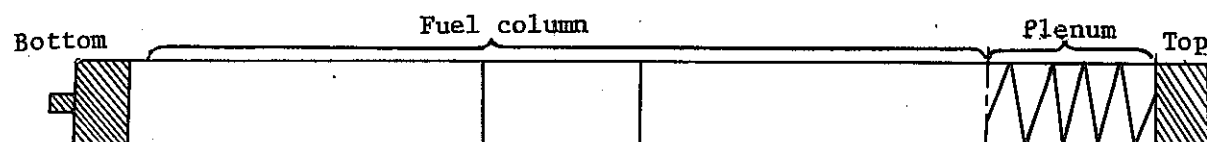
Before Irradiation - (1)**1. Fuel Pin Summary**

1-1 Total Fuel Pin Length		cm	
1-2 Fuel Stack Length	36.119	cm	
1-3 Material of Cladding/End Plug	347 S.S.	/	Maker/Lot No. /
			Cold Worked or Solution Treated %
1-4 Total Weight of Fuel Pin		g	
1-5 Total Weight of Fuel	22.51	g	
1-6 Spring Volume		cm ³	
1-7 Plenum Volume	7.371	cm ³	(Excluding pellet-cladding gap volume)
1-8 Spring Material/Spring Constnat			Kg/cm
1-9 Initial Internal Pressure At 20°C	1.033	Kg/cm ²	
1-10 Internal Gas			
- Total	0.407×10<sup>-3</sup>	mole	
- He	0.307×10<sup>-3</sup>	mole	
- Adsorbed	0.100×10<sup>-3</sup>	mole or	100 μl/g At °C
1-11 Coolant Flow Area/1 Fuel Pin	0.822	cm ²	
1-12 Av. Coolant Pressure	1.68	Kg/cm ²	
1-13 Density of Cladding	7.98	g/cm ³	

Table 6.3 Fuel Pin Detail

Before Irradiation - (2)

2. Fuel Pin Detail



		A	B	C	D	E	F	G	H	I	J	K
Node Length	(cm)	14.45	3.61	18.06								30.06
Fuel Inner Radius	(cm)											
Fuel Outer Radius	(cm)	0.2744	0.2744	0.2744								
Radial Gap Size	(cm)											
Clad Inner Radius	(cm)	0.2794	0.2794	0.2794								0.2794
Clad Outer Radius	(cm)	0.3175	0.3175	0.3175								0.3175
PuO ₂ Enrichment	(w/o)	0.2	0.2	0.2								
U-235 Enrichment	(w/o)	0.007	0.007	0.007								
Pu ²³⁹ /Pu ²⁴⁰ /Pu ²⁴¹ /Pu ²⁴²												
Fuel Density	(g/cm ³)	10.54	10.54	10.54								
Rod Power	(w/cm)	0.9792	1.0723	0.9485	(Relative value)							0.0
Fast Flux	(n/cm ² ·sec)	0.9792	1.0723	0.9485	(Relative value)							0.3
O/M Ratio		1.998	1.998	1.998								
Remarks												

Capsule I.R. (cm) 0.4254

Capsule O.R. (cm) 0.4762

Table 6.4 Irradiation History

PNCT841-75-17

Irradiation History - (1)

1. Reactor Operating Data (1) (Reactor EBR-II Position)

Date Year Mo.Day	Integrated Time		Coolant			Av.Linear Heat Rating (w/cm)	Av. Fast Flux (>0.1 MeV) (n/cm ² ·sec)	Clad Surface Temp. (°C)
	Days	Hours	Inlet Temp. °C	Outlet Temp. °C	Flow Rate g/sec			
	0.15		371.1	372.1		0.0		
	0.671		371.1	479.4		439.8	1.53×10 ¹⁵	
	2.00		371.1	479.4		439.8		
	2.125		371.1	372.1		0.0		
	2.50		371.1	372.1		0.0		
	2.921		371.1	479.4		439.8		
	6.35		371.1	479.4		439.8		
	6.643		371.1	419.2		195.5		
	6.80		371.1	419.2		195.5		
	7.142		371.1	479.4		439.8		
	9.50		371.1	479.4		439.8		
	10.0		371.1	372.1		0.0		

2. Radial Profiles of Rod Power and F. Flux

Radius	0.224	0.387	0.500	0.592	0.671	0.742	0.806	0.866	0.922	0.975
Fast Flux	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Power	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

6.3 Input Format

Table 6.5 shows the input format, and the sample calculation is shown in Appendix. The input explanation is as follows.

Card No.	Format	Symbol	Description
1	(2I5,3E10.5)	JFA JOUT EPSSPD EPSCRF EPSCRC	Option of reactor type 1 = fast breeder reactor 2 = heavy water reactor 3 = pressurized water reactor 4 = boiling water reactor Option for output style 0 = output only at the time step given in input 1 = output at the every time step computed in the code. Limitation of restructuring speed, and each region boundary movement should be less than EPSSPD (cm) in each time step. Default = 0.01 cm Limitation of fuel creep speed, and in each time step, fuel creep rate cal- culated by the equation (17) should be less than EPSCRF. Default = 0.05 Limitation of cladding creep speed, and in each time step, cladding creep rate calculated by equation (17) should be less than EPSCRC. Default = 0.01
2	(20A4)	TITLE	Title of the case, and can be continued as many cards as one wishes until the end card (*END). The title shown in the first card is written as the title of plotter output.
3	(16I5)	NN	Problem size and options. Number of axial meshes including plenum

Card No.	Format	Symbol	Description
			region.
		NCR	Number of radial meshes for stress-strain calculation in the columnar grain growth region.
		NER	Number of radial meshes for stress-strain calculation in the equiaxed grain growth region.
		NUR	Number of radial meshes for stress-strain calculation in the undisturbed region.
		NCL	Number of radial meshes for stress-strain calculation in the cladding.
		NP	Number of radial meshes for plutonium migration, and for radial power distribution.
		NOK	Option of fuel thermal conductivity. (See chapter 5; Material properties)
		NOS	Option of cladding swelling equation. (See chapter 5; Materials properties)
		NOC	Option of cladding creep equation. (See chapter 5; Materials properties)
		NPLOT	Option of plotter. 0 = plotter is not required. 1 = plotter output is required.
		NCRK	Option of fuel cracking. 0 = fuel cracking is not considered. 1 = fuel cracking routine becomes active.
		NSKIP	Option of temperature boundary. 0 = coolant outlet temperatures are input. 1 = cladding outer surface temperatures are input.
		NBUF	Number of time steps for plotter output. If JOUT of card No.1 is 0, NBUF equals the number of input time steps. Default = 50.

Using these numbers of meshes, the following symbols are defined.

$$NTR = NCR + NER + NUR + NCL$$

Total number of radial meshes

$$NFR = NCR + NER + NUR$$

Total number of radial meshes in the fuel region

Card No.	Format	Symbol	Description
4	(16I5)	IDLE	Input and plotter options. Options for the inputs of stress and strain conditions. 0 = input the initial conditions of stress and strain. 1 = the initial conditions of stress and strain are set to zero in the code, and the input cards from No.5 to No.18 are skipped.
		IPFUL	Number of axial position for plotter output concerned with temperature.
		IPSTN	Number of axial position for plotter output concerned with strain.
		IPSTS	Number of axial position for plotter output concerned with stress.
		IPCRK	Number of axial position for plotter output concerned with fuel cracking.
5	(5E15.8)	((EPR(I,J), I=1, NN), J=1, NTR)	Radial component of total strain at the initial condition. (At the axial position I and the radial position J.)
6	(5E15.8)	((EPC(I,J), I=1, NN), J=1, NTR)	Circumferential component of total strain at the initial condition.
7	(5E15.8)	((EPZ(I,J), I=1, NN), J=1, NTR)	Axial component of total strain at the initial condition.

Card No.	Format	Symbol	Description
8	(5E15.8)	((TPR(I,J), I=1, NN), J=1, NTR)	Radial component of accumulated creep strain at the initial condition.
9	(5E15.8)	((TPC(I,J), I=1, NN), J=1, NTR)	Circumferential component of accumulated creep strain at the initial condition.
10	(5E15.8)	((TPZ(I,J), I=1, NN), J=1, NTR)	Axial component of accumulated creep strain at the initial condition.
11	(5E15.8)	((TSR(I,J), I=1, NN), J=1, NTR)	Radial component of accumulated swelling strain at the initial condition.
12	(5E15.8)	((TSC(I,J), I=1, NN), J=1, NTR)	Circumferential component of accumulated swelling strain at the initial condition.
13	(5E15.8)	((TSZ(I,J), I=1, NN), J=1, NTR)	Axial component of accumulated swelling strain at the initial condition.
14	(5E15.8)	((SIGR(I,J), I=1, NN), J=1, NTR)	Radial component of averaged stress in the region (I,J) at the initial condition. (kg/cm ²)
15	(5E15.8)	((SIGC(I,J), I=1, NN), J=1, NTR)	Circumferential component of averaged stress in the region (I,J) at the initial condition. (kg/cm ²)
16	(5E15.8)	((SIGZ(I,J), I=1, NN), J=1, NTR)	Axial component of averaged stress in the region (I,J) at the initial condition.(kg/cm ²)

Card No.	Format	Symbol	Description
17	(5E15.8)	((ADISP(I,J), I=1, NN), J=1, NTR)	Axial displacement at the initial condition. (cm)
18	(5E15.8)	((RDISP(I,J), I=1, NN), J=1, NTR+2)	Radial displacement of the boundary at the initial condition. (cm)
19	(5E15.8)	(RVB(I), I=1, NN)	Fuel central hole radius at the initial condition. (cm)
20	(5E15.8)	(RCB(I), I=1, NN)	Radius of columnar grain growth region outer boundary at the initial condition. (cm)
21	(5E15.8)	(REB(I), I=1, NN)	Radius of equiaxed grain growth region outer boundary at the initial condition. (cm)
22	(5E15.8)	(RUB(I), I=1, NN)	Fuel outer radius at the initial condition. (cm)
23	(5E15.8)	(RCI(I), I=1, NN)	Cladding inner radius at the initial condition. (cm)
24	(5E15.8)	(RCO(I), I=1, NN)	Cladding outer radius at the initial condition. (cm)
25	(5E15.8)	(CLZO(I), I=1, NN)	Axial node length. (cm)

Card No.	Format	Symbol	Description
26	(5E15.8)	(QSFF(I), I=1, NN)	Axial distribution of fast neutron flux. (relative value)
27	(5E15.8)	(QSFP(I), I=1, NN)	Axial distribution of linear heat rating. (relative value)
28	(5E15.8)	(OTOM(I), I=1, NN)	Stoichiometry of each axial section.
29	(5E15.8)	(CPU(I, 1), I=1, NN-1)	Plutonium enrichment of each axial section.
30	(5E15.8)	(RHOP(I, 3), I=1, NN-1)	Pellet density of each axial section. (g/cm ³)
31	(5E15.8)	(F(1, L), L=1, NP)	Radial power distribution. (relative value)
32	(5E15.8)	RCPI RCPO AFNA	Capsule dimension and coolant flow area Capsule inner radius (cm) Capsule outer radius (cm) (If there is no capsule, PCPI>RCPO.) Coolant flow area per fuel pin. (cm ²)
33	(5E15.8)	(PG(I), I=1, 3)	Correction factor related to the surface tension of fission gas bubbles for the fuel swelling equation. (psi) (See chapter 5; Material properties)
34	(5E15.8)	PTOP PO	Outer force Axial force to the cladding (kg) Average coolant pressure at the initial condition. (kg/cm ²)

Card No.	Format	Symbol	Description
35	(5E15.8)	RHOC BURH1 BURH2	Cladding and fuel density parameter Density of cladding (g/cm^3) Correction factor related to the burnup effect on the columnar grain region density of equation (10). (a/o) Correction factor related to the burnup effect on the equiaxed grain region density of equation (10). (a/o)
36	(5E15.8)	CML(1) CML(2) DHR(1) DHR(2)	Fuel restructuring parameter Constant in the velocity equation of columnar grain region outer boundary movement of equation (6); C_1 (cm/sec) Constant in the velocity equation of equiaxed grain region outer boundary movement of equation (7); C_2 (cm/sec) Constant related to the activation energy of columnar grain region outer boundary movement of equation (6); (Q/R) ($^{\circ}\text{K}$) Constant related to the activation energy of equiaxed grain region outer boundary movement of equation (7); (Q/R) ($^{\circ}\text{K}$)
37	(5E15.8)	GASI TCON PDBU	Initial gas moles in the plenum including adsorbed gas. (moles) The ratio of coolant temperature rise along the fuel pin to the average linear heat rating. ($^{\circ}\text{C}\cdot\text{cm}/\text{w}$) (If the coolant outlet temperature is given in card No.38 or the cladding outer surface temperatures are given in card No.39, TCON=0.0) Depletion factor of linear heat rating as the burnup. (1 / a/o)

Card No.	Format	Symbol	Description
38	(E10.3, F5.1, 3E15.5, E10.5, I5)		Irradiation history data card. (This card and card No. 39, if necessary, should be repeated.) TIMEB Integrated time of irradiation (days). POI Average coolant pressure if changed. (kg/cm²) Default; POI is set to the same value as the one of the previous time step. POWAV Average linear heat rating of fuel pin. (w/cm) CFLX The ratio of the average fast neutron flux to the average linear heat rating. (n/cm²/sec)/(w/cm) TIN Coolant inlet temperature. (°C) TOUT Coolant outlet temperature. (°C) NPLOT C Option of plotter output for fuel crack pattern. 0 or blank = no plotter output 1 = plotter output at that time step.
39	(5E15.8)	(TOUTER(I), I=1, NN)	Cladding outer surface temperature distribution at the time step. (°C) (If ISKIP of card No.3 equals 0, this card should be omitted.)
40	E10.3	TIMEB	Option for the next case, and this card is used at the end of irradiation history. 0.0 = next case follows. /* = end of job.

Table 6.5 Input Format

INPUT DATA FORM

PAGE _____ OF _____

PNC841-75-17

JOB NUMBER _____										MAIN PROGRAM LABEL _____										KEYPUNCH RECORD _____										DATE _____										NO. OF CARDS _____									
PROBLEM _____										MODIFICATION _____										KEYPUNCHED BY _____																													
CODED BY _____										DATE _____										VERIFIED BY _____																													

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0												
01	J	F	A				J	O	U	T		E	P	S	S	P	D					E	P	S	C	R	F					E	P	S	C	R	C																																													
02	T	I	T	L	E																																																																													
03	-	-	-	-	-																																																																													
04	*END																																																																																	
05	N	N					N	C	R		N	E	R				N	U	R		N	C	L				N	P			N	O	K				N	O	S		N	O	C				N	P	L	O	T	N	C	R	K			N	S	K	I	P	N	B	U	F																
06	I	D	L	E			I	P	F	U	L	I	P	S	T	N		I	P	S	T	S	I	P	C	R	K																																																							
07	E	P	R	(	I	,	J	)																																																																										
08	E	P	C	(	I	,	J	)																																																																										
09	E	P	Z	(	I	,	J	)																																																																										
10	T	P	R	(	I	,	J	)																																																																										
11	T	P	C	(	I	,	J	)																																																																										
12	T	P	Z	(	I	,	J	)																																																																										
13	T	S	R	(	I	,	J	)																																																																										
14	T	S	C	(	I	,	J	)																																																																										
15	T	S	Z	(	I	,	J	)																																																																										
16	S	I	G	R	(	I	,	J	)																																																																									
17	S	I	G	C	(	I	,	J	)																																																																									
18	S	I	G	Z	(	I	,	J	)																																																																									
19	A	D	I	S	P	(	I	,	J	)																																																																								
20	R	D	I	S	P	(	I	,	J	)																																																																								

MEMO NO.

PAGE OF

PNC - PU

Table 6.5 Input Format (cOntinued)

INPUT DATA FORM

PAGE _____ OF _____

JOB NUMBER _____		MAIN PROGRAM LABEL _____		KEYPUNCH RECORD _____		DATE _____		NO. OF CARDS _____	
PROBLEM _____		MODIFICATION _____		KEYPUNCHED BY _____					
CODED BY _____		DATE _____		VERIFIED BY _____					
	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0
01	☐ RVB(I)	☐	☐	☐	☐	☐	☐	☐	☐
02	☐ RCB(I)	☐	☐	☐	☐	☐	☐	☐	☐
03	☐ REB(I)	☐	☐	☐	☐	☐	☐	☐	☐
04	☐ RUB(I)	☐	☐	☐	☐	☐	☐	☐	☐
05	☐ RCI(I)	☐	☐	☐	☐	☐	☐	☐	☐
06	☐ RCO(I)	☐	☐	☐	☐	☐	☐	☐	☐
07	☐ CLZO(I)	☐	☐	☐	☐	☐	☐	☐	☐
08	☐ QSFF(I)	☐	☐	☐	☐	☐	☐	☐	☐
09	☐ QSFP(I)	☐	☐	☐	☐	☐	☐	☐	☐
10	☐ OTOM(I)	☐	☐	☐	☐	☐	☐	☐	☐
11	☐ CPU(I,1)	☐	☐	☐	☐	☐	☐	☐	☐
12	☐ RHOP(I,3)	☐	☐	☐	☐	☐	☐	☐	☐
13	☐ F(1,L)	☐	☐	☐	☐	☐	☐	☐	☐
14	☐ RCPI	☐	☐ RCPO	☐	☐ AFNA	☐	☐	☐	☐
15	☐ PG(1)	☐	☐ PG(2)	☐	☐ PG(3)	☐	☐	☐	☐
16	☐ PTOP	☐	☐ PO	☐	☐	☐	☐	☐	☐
17	☐ RHOC	☐	☐ BURH1	☐	☐ BURH2	☐	☐	☐	☐
18	☐ CMI(1)	☐	☐ CMI(2)	☐	☐ DHR(1)	☐	☐ DHR(2)	☐	☐
19	☐ GASI	☐	☐ TCON	☐	☐ PDBU	☐	☐	☐	☐
20	☐ TIMEB	☐ POI	☐ POWAV	☐	☐ CFLX	☐	☐ TIN	☐	☐ TOUT
	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0

MEMO NO.

PAGE OF

PNC – PU

Table 6.5 Input Format (continued)

INPUT DATA FORM

PAGE OF

[illegible]

- 65 -

	MEMO NO.

PAGE OF

6.4 Output Format

The output consists of the list of input parameters and the calculated results at each time steps. The time steps to be output can be controlled by an option.

a) Input list

Page	Title	Contents
1	_____	Expression of the type of reactor.
2	_____	Title of the case
3	CONTROL NUMBER AND OPTION	Options adopted in the code by the input card No.3 and 4.
	CENTRAL VOID RADIUS	Central void radii of axial section 1 to NN. (cm)
	COLUMNAR GRAIN REGION OUTER BOUNDARY	Columnar grain growth region outer radius. (cm)
	EQUIAXED GRAIN REGION OUTER BOUNDARY	Equiaxed grain growth region outer radius. (cm)
	FUEL OUTER RADIUS	Fuel outer radius. (cm)
	CLAD INNER RADIUS	Cladding inner radius. (cm)
	CLAD OUTER RADIUS	Cladding outer radius. (cm)
	CPASULE INNER RADIUS	Capsule inner radius if there is. (cm)
	CAPSULE OUTER RADIUS	Capsule outer radius if there is. (cm)
	COOLANT FLOW AREA	Coolant flow area. (cm ²)
	FUEL AXIAL LENGTH	Fuel axial node length and the total fuel column length. (cm)
	CLAD AXIAL LENGTH	Cladding axial node length and the total cladding length. (cm)

Page	Title	Contents
	DENSITY OF FUEL (3 REGIONS) AND CLAD	Fuel density of columnar, equiaxed, and undisturbed regions and cladding density are shown for each axial node. (g/cm ³)
	O/M OF FUEL	Stoichiometry of each axial node.
	PLUTONIUM CONCENTRATION (WEIGHT-FRACTION)	Radial distribution of plutonium concentration of each axial node.
	AXIAL FAST FLUX (E.GE. 0.1 MEV) RATE	Axial fast flux distribution normalized as the mean value to be unit.
	AXIAL HEAT GENERATION RATE	Axial distribution of heat generation rate normalized in the fuel region.
	RADIAL HEAT GENERATION RATE	Radial heat generation rate normalized.
	INITIAL GAS IN PLENUM	Initial gas moles in the plenum including adsorbed gas. (moles)
	*** ACTUALLY USED CORE FOR ARRAY	The actually used core for array A(N) and input value. (words)
	*** ACTUALLY USED CORE FOR MATRIX	The actually used core for array AM(N) and input value. (words)
	*** RECOMMENDED CORE SIZE	The actually used core for the program shown in multiple of 128 K. (byte)

b) Output for each time step

Title	Contents
STEP NO.	Integrated time steps used in the code. (The output time steps are controlled by an option, JOUT of input card No.1)

Title		Contents
TIME		Integrated irradiation time from start up (hr)
POWAV		Average linear heat rating: Q (w/cm)
		$Q = \frac{Q_0}{(1+\epsilon_f^Z)}$ where Q_0 = average heat rating of input, and ϵ_f^Z = total axial strain of fuel column.
TOUT		Coolant outlet temperature (°C) (If it is not given in input, the code calculate the temperature.)
TIN		Coolant inlet temperature given in input (°C)
DELT		Time interval of the time step (hr)
CFLX		Ratio of fast neutron flux to linear heat rating given in input ((n/cm ² /sec)/(w/cm))
PO		Average coolant pressure (kg/cm ²)
GROSS OUTPUT NEXT		
		Integrated results of fuel pin conditions.
POWER		Linear heat rating of each axial section (w/cm)
FLUX		Average fast neutron flux of each axial section (n/cm ² /sec)
FLUENC		Average fast neutron fluence of each axial section (n/cm ²)
BURN UP		Average burn up of each axial section (a/o)
BURN UP		Average burn up of each axial section (MWD/T-fuel)
FUEL LENGTH		Axial node length of fuel column (cm)
CLAD LENGTH		Axial node length of cladding (cm)
THEO. DENS.		Theoretical density of fuel of each axial section (g/cm ³)
COOLANT TEMP		Coolant temperature distribution of each axial section (°C)
FILM H.		Coolant-cladding heat conductance of each axial section. (w/cm ² /°C)
GAP H.		Fuel-cladding gap conductance of each axial section (w/cm ² /°C)
DELT GAS REL.		Gas released during the time interval from each axial section (moles)

Title	Contents
TOTAL GAS IN PLENUM CUMULATIVE GAS RELEASED PERCENT OF TOTAL F.P. GAS PLENUM PRESSURE CENTRAL HOLE PRESSURE TOTAL FUEL LENGTH AND FL/FLO TOTAL CLAD LENGTH AND CL/CLO	Total gas in plenum including released fission gas and adsorbed gas. (moles) Cumulative fission gas released. (moles) Fission gas release rate. (%) Plenum pressure. (kg/cm ²) Pressure in the central void. (kg/cm ²) Total fuel column length and the percentage of length change. (cm) & (%) Total cladding length and the percentage of length change. (cm) & (%)
+++ PERFORMANCE RADIUS TEMP. AV. TEMP. ALPHA-T DENS MASS	Average performance of each radial region (i.e. columnar grain region, equiaxed grain region, undisturbed region of fuel, cladding and capsule region) is shown for each axial section. Radius of the boundary (cm) Temperature of the boundary (°C) Average temperature in the region (°C) Thermal expansion rate of the region Average density of the region (g/cm ³) Mass per unit length of the region (g/cm)
STRESS AND STRAIN OUTPUT NEXT RADIUS	Material properties, boundary conditions, stress and strain are shown for every radial meshes and axial sections which are used in stress-strain calculation. The output are separated into capsule, cladding and fuel regions. Radius of each boundary of radial mesh. (cm)

Title	Contents
DISPLACEMENT	Displacement of each boundary during the time interval. (cm)
TEMP.	Temperature of each boundary
PRESS.	Contact pressure at the boundaries.
ALPHA-T	Thermal expansion rate of each radial mesh.
POISSON RATIO	Average poisson ratio of each radial mesh.
YOUNG M.	Young modulus of elasticity of each radial mesh. (kg/cm ²)
NUMBER OF CRACK	Number of cracks in each radial mesh for radial, circumferential and axial directions.
STRESS	Stress in each radial mesh for radial, circumferential and axial directions. (kg/cm ²)
TOTAL STRAIN	Accumulative total strain in each radial mesh for radial, circumferential and axial directions.
CREEP STRAIN	Accumulative creep strain in each radial mesh for radial, circumferential and axial directions.
CREEP IN DELT	Creep strain increment in the time interval in each radial mesh for radial, circumferential and axial directions.
SWELLING STRAIN	Accumulative swelling strain in each radial mesh.
TOTAL	Total swelling strain.
SOLID COMPONENT	Swelling caused by solid fission products. (for fuel region only)
GAS COMPONENT	Swelling caused by gaseous fission products. (for fuel region only)
HOT PRESSING	Strains caused by hot pressing. (for fuel region only)

Title	Contents
PERCENT CHANGE IN ORIGINAL O.D. AVERAGE POISSON RATIO AVERAGE YOUNG MODULUS GAP SIZE FRICTION FORCE	Percent change of outer diameter vs original diameter. (%) Poisson's ratio averaged in fuel or cladding region. Young's modulus of elasticity averaged in fuel or cladding. (kg/cm ²) Radial gap size between fuel and cladding. (cm) Axial friction force between fuel and cladding. (kg/cm)
+++ PLUTONIUM MIGRATION RADIUS TEMP. PU CONC.	Plutonium radial distribution in each axial cross section. Boundary radius of mesh used in the plutonium migration calculation. (cm) Temperature of boundary of mesh. (°C) Plutonium concentration in each radial mesh.
CPU TIME	Accumulative computer time from the beginning of calculation. (sec)

3) Plotter output

ACTIVE-II has the routines of plotter output provided by Calcomp. It requires Calcomp 1136 drum type plotter, and the paper of 13 inches wide and about 10 meters long. The types of pens are black ink pen of 0.5 mm, red ink pen of 0.5 mm and green ink pen of 0.5 mm for No.1, 2 and 3 pen positions, respectively.

The estimated time for plotting is about 40 minutes.

Plotter output is controlled by the options given by input card. No.3 and No.4

The first figure is the title of the output and the date of code used is given.

The second figure is a set of fuel cracking patterns of cross sections perpendicular to and parallel to the axis. Red and green dotted

lines show the cracking positions. This output time position is controlled by NPLOTC given in input card No.38.

The third figure shows the gross performance of fuel pin as a function of time. Maximum fuel center temperature and maximum fuel surface temperature of fuel pin are shown by black solid lines, and red line shows the fission gas release rate and green line shows the plenum pressure as a function of time.

The 4th one shows the linear heat rating and burnup of the axial cross section specified by the option IPFUL given in input card No.4.

The 5th figure shows the maximum fuel center temperature as a function of time of the axial cross section specified by the option IPFUL given in input card No.4.

The 6th figure shows the radial temperature profiles of fuel temperature at each time step at the axial cross section specified by the option IPFUL given in input card No.4. In this figure, columnar grain region outer boundary and equiaxed grain region outer boundary are pointed by red and green marks respectively.

The 7th figure shows the fuel restructuring as a function of time. Equiaxed grain outer boundary, columnar grain outer boundary and central void radius are shown at the specified axial cross section by the option IPFUL given in the input card No.4.

The 8th figure shows the relationship between fuel-cladding gap and cladding hoop stress as a function of time. The axial cross section chosen for this figure is specified by the option IPSTS given in the input card No.4.

The 9th figure shows the total cladding circumferential strain and its creep and swelling components as a function of time. The axial cross section is specified by the option IPSTN given in the input card No.4.

The last one shows the plutonium redistribution calculated in subroutine PLUTO. Radial plutonium distribution for each time step is shown at the cross section specified by the option IPFUL given in the input card No.4

Acknowledgements

This ACTIVE-II code is not only the work of the authors but also the fruits of knowledge of many persons, to whom the authors would like to thank from the bottom of their hearts: P. G. Shewmon, B. R. T. Frost, R. W. Weeks, V. Z. Jankus, L. A. Neimark, J. D. B. Lambert, and W. F. Murphy of Materials Science Division of Argonne National Laboratory, U.S.A., A Watanabe of PNC and A. Doi of Hitachi Ltd. for their assistances for the author to learn actual fuel behavior and LIFE-I code, which is the mother of ACTIVE-II; T. Ohkubo of Tokyo Univ. and T. Kikuchi of JAERI for their consulting the modeling; H. Akutsu, T. Aoki, H. Mizuta, T. Ohtake, R. Yumoto, T. Kajiyama, K. Itagawa and other members of Plutonium Design Section of PNC, and M. Yomori of Japan IBM for their continued strong support and encouragement.

References

- 1) R. F. Canon, J. T. A. Roberts and R. J. Beals; Deformation of UO_2 at High Temperatures, J. Am. Ceram. Soc. 54(1971) 105.
- 2) V. Z. Jankus and R. W. Weeks; LIFE-II, a Computer Analysis of Fast Reactor Fuel-Element Behavior as a Function of Reactor Operating History, Nucl. Eng. and Design 18 (1972) 83-96.
- 3) R. A. Seban and T. Shimazaki; Heat Transfer to a Fluid Flowing Turbulently in a Smooth Pipe with Walls at Constant Temperature, ASME Paper 50-A-128 (1950).
- 4) C. W. Sayles; Trans. Am. Nucl. Soc. 10, 458 (1967).
- 5) W. E. Bailly; Nuclear Metallurgy Symposium, (1967).
- 6) A. Biancheria; WARD-4135-1, P.6 (Sep. 1969).
- 7) V. Z. Jankus & R. W. Weeks; LIFE-I, a FORTRAN-IV Computer Code for the Prediction of Fast-Reactor Fuel-Element Behavior, ANL-7736 (Nov. 1970).
- 8) BNWL-1345-1
- 9) Chromium-Nickel Stainless Steel Data Section I-Bulletin B, 'Mechanical and Physical Properties of the Austenitic Chromium-Nickel Stainless Steel at Elevated Temperature', International Nickel Co. Inc. (1963).
- 10) Steel for Nuclear Application, Manual by U.S. Steel, (May, 1967).
- 11) WARD-4135-11
- 12) W. H. Jens & P. A. Lottes; ANL-4627, (1951).
- 13) M. F. Lyons; UO_2 Pellet Thermal Conductivity from Irradiations with Central Melting, GEAP-4624.
- 14) A. Biancheria; Trans. Am. Nucl. Soc. 9, 15 (1966).
- 15) M. F. Lyons; DOCKET-50331
- 16) Nucleonics Handbook of Nuclear Research & Technology.
- 17) Engineering Information & Data (AMAX)
- 18) F. W. Dittus and L. M. K. Boelter; Univ. Calif., Pubs. Eng. 2, P.443, (1930).

APPENDIX A: SUBROUTINE DICTIONARY

Appendix A: Subroutine Dictionary

ACTIVE-II requires five kinds of subroutines, that is, 1) FORTRAN library subroutines, 2) IBM Tokyo data center service library subroutines, 3) IBM support subroutine (SL-MATH), 4) plotter subroutines, and 5) original subroutines written for ACTIVE-II

The explanation of the FORTRAN library subroutines are abridged, because they are fundamental for FORTRAN language. The full lists of the original subroutines written for ACTIVE-II are shown in Appendix D, and the explanations are described in this chapter. As other kinds of subroutines are proprietary of IBM, only the explanations of these are given here.

A.1 IBM Tokyo Data Center Service Library Subroutines

1) STIMER and TTIMER

The combinative use of STIMER with TTIMER enables the users to obtain the CPU time.

Example:

```

      .
      .
      .
CALL STIMER (ID) ..... ①
      .
      .
      .
CALL TTIMER (T) ..... ②
      .
      .
      .

```

where ID=2.

The argument T in the statement ② is the CPU time from the statement ① to the statement ②. (Floating type in unit of second)

2) DATE

The subroutine DATE returns the date of today in the format of month/day/year.

Example:

```

      REAL*8    DAY
      CALL DATE (DAY)
      WRITE(6,1) DAY
1  FORMAT(1H0, 'DATE', A8)

```

The argument DAY must be defined in double precision.

A.2 IBM Support Subroutines (SL-MATH)

SL-MATH (Subroutine Library-Mathematics)

SL-MATH provides powerful tools for the solution of many problems in industry, science and engineering. SL-MATH is a collection of subroutines dealing with matrix, algebra and numerical mathematics. The subroutines, computational building blocks written in FORTRAN-IV, operate on data that reside in main storage.

ACTIVE-II uses subroutine MSNR and MBSR in the SL-MATH to solve a stress-strain matrix, which is a non-symmetric sparse matrix.

1) Storing and processing of vectors for non-symmetric sparse matrices

A sparse N-by-N matrix A is described by three one-dimensional arrays: An array, also named A, which contains only the non-zero elements of the matrix A (stored row-wise), an array JA containing the corresponding column indices of the non-zero elements of A (stored row-wise), and an array IA containing pointers to the locations of the first element and corresponding column index of each row of A within the array A and JA, respectively. That is, A(K), for K=IA(I) is the first non-zero element of the i-th row of A, JA(K) being the corresponding column index.

The dimension of the array A (and the array JA, respectively) is (at least) the number of non-zero elements of the matrix A. The dimension of the array IA is (at least) N+1. IA(N+1) is the pointer to the (virtual) element of the array A immediately following the last location of a non-zero element, (to indicate the end of the last row).

Example:

$$\text{The 6-by-6 matrix } A = \begin{pmatrix} 7 & 0 & -3 & 0 & -1 & 0 \\ 2 & 8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -3 & 0 & 0 & 5 & 0 & 0 \\ 0 & -1 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & -2 & 0 & 6 \end{pmatrix}$$

with 12 non-zero elements is described by the arrays A, JA, IA as follows.

IA(1) = 1,	A(1)=7,	A(2)=-3,	A(3)=-1,
	JA(1)=1,	JA(2)= 3,	JA(3)= 5,
IA(2) = 4,	A(4)=2,	A(5)= 8,	
	JA(4)=1,	JA(5)= 2,	
IA(3) = 6,	A(6)=1,		
	JA(6)=3,		
IA(4) = 7,	A(7)=-3,	A(8)=5,	
	JA(7)= 1,	JA(8)=4,	
IA(5) = 9,	A(9)=-1,	A(10)=4,	
	JA(9)= 2,	JA(10)=5,	
IA(6) = 11,	A(11)=-2,	A(12)=6,	
	JA(11)=4,	JA(12)=6,	
IA(7) = 13.			

2) Subroutine MSNR

MSNR factorizes a given general non-singular sparse matrix A into a lower (L) and an upper (U) triangular factor ($A=L \cdot U$). Thereby, partial pivoting based on row-density is performed, i.e., the rows of A are chosen as pivot rows in the sequence of their lengths, and the updated elements with maximum absolute value are chosen as pivot elements (column choice).

Usage:

CALL MSNR (N,IA,JA,A,IBOT,IC,JC,IU,DI,C,IPR,IPC,IER,KA,AR)

- N. Given number of rows and columns to be processed for matrix A.
- IA. Given vector (dimension N+1) containing the pointers to the first element in each row of A in row-wise order.
- JA. Given vector (dimension NT=total number of non-zero elements of A) containing the column indices of A in row-wise order.
- A. Given vector (dimension NT) containing the values of the elements of A in row-wise order.
- IBOT. Given dimension of vectors JC and C (refer to the section Remarks).
- IC. Resultant vector (dimension N+1) containing the pointers to the first element in each row of L.
- JC. Resultant vector (dimension IBOT=total number of elements in the factors L and U, excluding the diagonals), containing the column indices of the elements of L and U in row-wise order. The column index of the last element in row I of L is followed by that of the first element in row I of U to the right of the diagonal.
- IU. Resultant vector (dimension N) containing the pointers to the first element in each row of U.
- DI. Resultant vector (dimension N) containing the values of the diagonal elements of L.

- C. Resultant vector (dimension IBOT) containing the numerical values of L and U in row-wise order, with the exception of the diagonal elements.
- IPR. Resultant vector (dimension N) determining the sequence of pivot rows.
- IPC. Resultant vector (dimension N) determining the sequence of pivot columns.
- IER. Resultant error indicator.
- KA. Auxiliary vector (dimension N) containing the column indices of the pivot row.
- AR. Auxiliary vector (dimension N) containing initially the values of the elements of row I of A and, finally, those of the rows I of L and U.

This vector must be defined in double precision.

Error Table:

Code	Explanation	Program Action	Comments
0	No error was detected.	The total number of multiplications and the number of built-ups are returned in KA(1) and KA(2), respectively.	
1000	Length of row I of A is zero.	The index I is returned in KA(1). Processing is discontinued.	
2000	Numerical value of the pivot element is zero.	The index of the last row processed correctly is returned in KA(1). KA(2) and KA(3) contain the pivot row and column last chosen. Processing is discontinued.	A is singular.
3000	Space of JC and C is exceeded.	The index of the last row processed correctly is returned in KA(i), and processing is discontinued.	IBOT must be increased.

Code	Explanation	Program Action	Comments
5000	Pivot element P at step I is approximately zero. ($-\text{FUZZ} \leq P \leq \text{FUZZ}$) (refer to the section Remarks).	Same as for code 2000.	Pivot element has a value below the limit.

Remarks:

The sequence (IPR) of pivot rows is determined by sorting the rows by their increasing lengths. At the i -th factorization step the row $\text{IRR}(I)$ of A is transformed into the i -th row of L and U by subtracting multiples of rows of U . After this process the element with the maximum absolute value in the remaining columns is chosen as pivot. Following this modified scheme of Gaussian elimination, the numerical values for all non-zero elements $U(i,j; i \neq j)$ and $L(i,j)$ are computed and stored row-wise as one matrix in the vector C without the diagonal elements, which are stored in vector DI . Only operations on non-zero operands are performed.

If $NT1$ is the number of built-ups in L and U , and $NT2$ is the total number of elements of A not located on the diagonal, $IBOT$ must be equal or greater than $NT1+NT2$ =total number of resultant element in JC or $C(L,U)$, respectively.

At the beginning of $MSNR$ there is a statement by which a variable $FUZZ$ is set to zero. Only pivot elements having an absolute value greater than $FUZZ$ will be accepted. The user may assign a small non-zero tolerance value to this variable by changing this statement.

3) Subroutine MBSR

$MBSR$ solves the system of linear equations $AX=B$ for a general non-singular sparse coefficient matrix A , given its right-hand sides (B) and its LU decomposition.

Usage:

CALL MBSR (N,IPR,IPC,IC,IU,JC,DI,C,B,X,Y)

- N. Given number of rows and columns to be processed for matrix A.
- IPR. Given permutation vector (dimension N) determining the sequence of pivot rows.
- IPC. Given permutation vector (dimension N) determining the sequence of pivot columns.
- IC. Given vector (dimension N+1) containing the pointers to the first element in each row of L.
- IU. Given vector (dimension N) containing the pointers to the first element in each row of U.
- JC. Given vector (dimension IBOT as used in the preceding factorization by MSNR), containing the column indices of the rows in L and U stored in row-wise order as one matrix C.
- DI. Given vector (dimension N) containing the diagonal elements of L.
- C. Given vector (dimension IBOT) containing row-wise numerical values of L and U without the diagonal elements.
- B. Given vector (dimension N) containing the right-hand sides of the system of linear equations.
- X. Resultant vector (dimension N) containing the solution of $AX=B$.
- Y. Auxiliary vector (dimension N) containing the intermediate results of $LY=B$.

Remarks:

Using the permutation vectors IPR and IPC of pivot rows and columns (determined by partial pivoting based on row density by MSNR), MBSR solves first the forward substitution $LY=B$. Then $UX=Y$ is solved (backward substitution), and the result is returned in X.

The method differs from the usual forward and backward substitution in that MBSR,

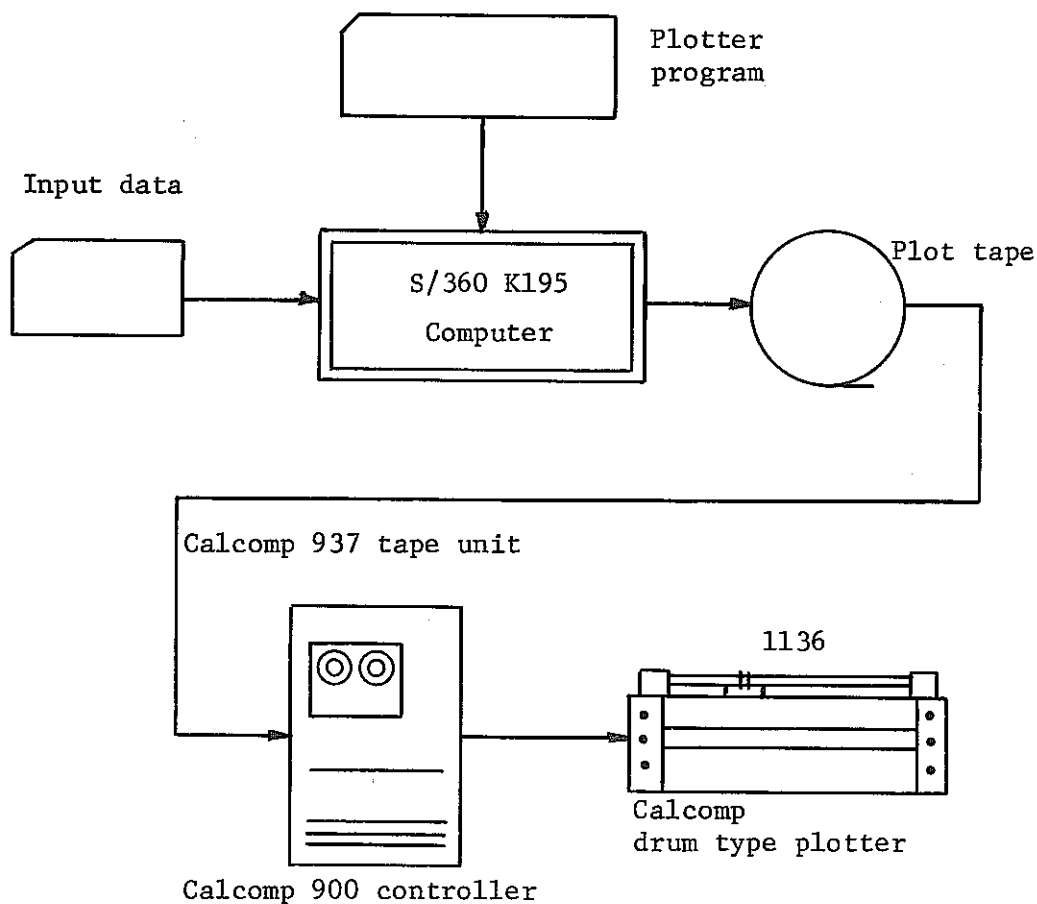
- (1) deals with rows and columns in permuted order, and
- (2) processes only the non-zero elements of the factors L and U.

A.3 Plotter Routines

Calcomp provides basic and functional subroutines, which is a set of closely related subroutines that generate output for controlling a digital graphic system. The programmer is not required to communicate with the hardware in its own data structure. Instead, he communicates with the set of subroutines in a manner such as: 'move the pen to specified coordinates'; 'place some characters at a certain location on the page'; 'draw a line through a series of points'.

ACTIVE-II uses nine plotter subroutines to plot several variables and transform the values of variables which are stored in the sequence of time history into visible forms.

Plotting system configuration (offline) is shown below.



1) CALL PLOTS(X,X) or CALL PLOTS(X,X,LDEV)

The PLOTS entry is used to initialize the PLOT subroutines. It must be called only once before any other call to plotter subroutine is given.

X is not used.

LDEV is the logical output device number.

If LDEV is not specified, it is set to 77.

2) CALL PLOT (X1,Y1,IPEN)

The PLOT entry is used primarily to move the pen in a straight line to a new position, with the pen either up or down during the movement.

X1,Y1 ; are the X, Y coordinates, in millimeters from the current reference point (origin), of the terminal position to which the pen is moved. An origin (where both X, Y equal zero) may be established anywhere on the plotting surface, as explained below for negative IPEN values.

IPEN = 2 ; the pen is down during movement, thus drawing a visible line.

IPEN = 3 ; the pen is up during movement.

IPEN = -2 ; A new origin is defined at the terminal position after or -3 the movement is completed as if IPEN were positive.

IPEN must be 999 in the last plotter call in a program. The effects are the same as if IPEN=-3, except that a 999 search record is written, and the output device is closed.

3) CALL NEWPEN (NPEN)

This entry enables program selection of any one of the three available pens. Pen No.1 is initially selected when the PLOTS entry is called.

NPEN ; specifies the number of the pen to be selected (1-3).

The old pen is raised, and the new pen is moved to the same physical location where the old pen was positioned.

4) CALL SYMBOL (X1, Y1, HEIGHT, IBCD, ANGLE, NC)

This entry is used to draw text such as titles, captions, and legends.

X1, Y1 ; are the coordinates, in millimeters, of the lower left-hand corner (before character rotation) of the first character to be produced. The pen is up while moving to this point. If X1 and/or Y1 equals 999.0, X1 and/or Y1 is assumed to be the current position.

HEIGHT ; is the height, in millimeters, of the character to be plotted.

IBCD ; is the text to be used as annotation. The character(s) must be left-justified and contiguous in: a single variable, an array, or a Hollerith literal.

ANGLE ; is the angle, in degrees from the X-axis, at which the annotation is to be plotted. ANGLE is positive, if counter-clockwise.

NC ; is the number of characters to be plotted from IBCD. If NC>0, the data must be left-justified in the first location of IBCD. If NC=0, one alphanumeric character is produced, using a single character which is right-justified in the first location of IBCD.

5) CALL SYMBOL(X1, Y1, HEIGHT, ICODE, ANGLE, IPEN)

This form of SYMBOL call is used to draw special centered symbols such as a box, octagon, triangle, etc., for plotting data points. X1, Y1, HEIGHT, and ANGLE are the same as described for the call in the above section (4). If the symbol to be produced is one of the centered symbols (i.e., if ICODE is less than 14), X1, Y1 represent the geometric center of the character produced.

ICODE ; is the integer equivalent of the desired symbol. Valid integers and their symbols are listed in the symb1 table of the Table A.1. If ICODE is 0 through 13, a centered symbol is produced.

IPEN =-1 ; the pen is up during movement, after which a single symbol is produced.

=-2 or less; the pen is down during movement, after which a single symbol is produced.

6) CALL NUMBER (X1, Y1, HEIGHT, FPN, ANGLE, NDEC)

This entry is used to draw a real variable (floating-point number). X1, Y1, HEIGHT, and ANGLE are the same as those arguments described for subroutine SYMBOL.

FPN ; is the location of the floating-point number that is to be plotted.

NDEC = 0 ; only the integer portion of the number and a decimal point are plotted after rounding.

=-1 ; only the integer portion of the number is plotted after rounding.

=1 through 11; specifies the number of digits to the right of the decimal point that are to be plotted.

7) CALL CIRCLE (X1, Y1, A1, A2, R1, R2, DS)

CIRCLE draws a circle, arc, or a spiral starting at a given point.

X1, Y1 ; are the coordinates of the starting point of the arc, in millimeters.

A1 ; is the angle for the start of the arc, in degrees.

A2 ; is the angle for the end of the arc, in degrees.

R1 ; is the radius at the start of the arc, in millimeters.

R2 ; is the radius at the end of the arc, in millimeters.

DS ; is a code used to specify the type of line desired.

If DS=0.0, a solid arc is drawn.

=0.5, a dashed arc is drawn.

8) CALL AROHD (X0, Y0, X1, Y1, S, W, ICODE)

AROHD draws an arrowhead at the end of a line segment.

X_0, Y_0 ; are the coordinates of the starting point of the line segment determining the direction of the arrowhead, in millimeters.

X_1, Y_1 ; are the coordinates of the tip of the arrowhead, in millimeters.

S ; is the length of the arrowhead, in millimeters.

W ; is the width of the arrowhead, in millimeters.

ICODE ; is a two-digit decimal code $10 \cdot I + J$ where:

I is 0 if no line is desired from X_0, Y_0 to the arrowhead at X_1, Y_1 .

1 if a line is desired.

2 if a line is desired and a second arrowhead pointing in the opposite direction is desired at X_0, Y_0 .

J is 1-7 specifying the type of arrowhead desired, as shown in Table A.1.

9) CALL DASHPT (X_1, Y_1, W)

DASHPT draws a dashed line from the present pen position to a specified point.

X_1, Y_1 ; are the coordinates of the point to which the dashed line is to be drawn, in millimeters.

W ; is the length of the dash and the space between dashes, in millimeters.

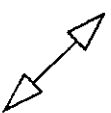
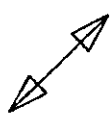
I - 0							
I - 1							
I - 2							
	J - 1	J - 2	J - 3	J - 4	J - 5	J - 6	J - 7

Table A.1 Symbols for plotter

CHARACTERS AVAILABLE IN THE IBM 360 SYMBOL ROUTINE

CODES NEXT TO EACH SYMBOL ARE: 1. INTEGER CODE USED IN SPECIAL SYMBOL CALL.
2. INTERNAL HEXADECIMAL CODE (EBCDIC MODULO 128)
SEE THE IBM 360 REFERENCE DATA CARD FOR CORRESPONDING PUNCHED CARD CODES.

0 00		16 10		32 20	}	48 30	Σ	64 40		80 50	&	96 60	-	112 70	0
1 01		17 11		33 21	{	49 31	$\div$	65 41	A	81 51	J	97 61	/	113 71	1
2 02		18 12	$\wedge$	34 22	μ	50 32	$\leq$	66 42	B	82 52	K	98 62	S	114 72	2
3 03	$+$	19 13	$\equiv$	35 23	π	51 33	$\geq$	67 43	C	83 53	L	99 63	T	115 73	3
4 04	$\times$	20 14	$\rightarrow$	36 24	ϕ	52 34	Δ	68 44	D	84 54	M	100 64	U	116 74	4
5 05		21 15		37 25	$\ominus$	53 35	$\square$	69 45	E	85 55	N	101 65	V	117 75	5
6 06	$\uparrow$	22 16	$\neq$	38 26	ψ	54 36	$\square$	70 46	F	86 56	O	102 66	W	118 76	6
7 07	$\times$	23 17	$\pm$	39 27	$\times$	55 37	$\backslash$	71 47	G	87 57	P	103 67	X	119 77	7
8 08	$\mathbb{Z}$	24 18	$-$	40 28	w	56 38	$\Uparrow$	72 48	H	88 58	Q	104 68	Y	120 78	8
9 09	$\mathbb{Y}$	25 19	$-$	41 29	λ	57 39	$\sqrt{}$	73 49	I	89 59	R	105 69	Z	121 79	9
10 0A		26 1A		42 2A	α	58 3A	$\dagger$	74 4A	Φ	90 5A	$\downarrow$	106 6A	∞	122 7A	$\circ$
11 0B	$\ast$	27 1B	$\int$	43 2B	δ	59 3B	$\ddagger$	75 4B	$\circ$	91 5B	\$	107 6B	,	123 7B	#
12 0C	$\times$	28 1C	$\supset$	44 2C	$\in$	60 3C	$\leftarrow$	76 4C	$<$	92 5C	$\times$	108 6C	$\%$	124 7C	$\odot$
13 0D		29 1D	$\vee$	45 2D	η	61 3D	$\times$	77 4D	(	93 5D	)	109 6D	$-$	125 7D	∇
14 0E	$\star$	30 1E	$\sim$	46 2E		62 3E	$\uparrow$	78 4E	$+$	94 5E	$\circ$	110 6E	$>$	126 7E	$=$
15 0F	$-$	31 1F	$\approx$	47 2F		63 3F	$\downarrow$	79 4F		95 5F	$-$	111 6F	$?$	127 7F	∇

NOTES: 1. THE FIRST 14 SYMBOLS IN THE TABLE ARE CENTERED SYMBOLS.
2. ANY INTEGER OR HEX CODE, OTHER THAN THOSE SHOWN, WILL BE CONSIDERED MODULO 128. THEREFORE EITHER OF TWO PUNCHED CARD CODES FROM THE 360 REFERENCE CARD MAY BE USED FOR EACH SYMBOL.

Table A.1 Symbols for plotter (continued)

A.4 Original Subroutines

Original subroutines written for ACTIVE-II are 59, and the whole lists are given in Appendix D.

The names of subroutines are as follows.

MAIN	ACTIVE	AREA
ATBFUN	AXIAL	AXISD
AXIST	BCALC	BCALCP
BEGIN	BURN	CHECK
CHKSPD	CLEAR	CLTMP
COEF	CORTMP	CRACK
CREEC	CREEF	EDTGRS
EDTINP	EDTMIX	EQUISI
FLCOND	FLTMP	FLXCAL
FRICHK	GAPCON	GASOUT
GOLDN	HEADER	HKNS
INPUT	IOSET	ITERAT
LENGTH	MATRX	MATRXP
OTMCAL	PLENP	PLOT1
PLOT2	PLOT3	PLOT4
PLOT5	PLTOR	PLUTO
POWCUT	PROPT	RHOMOD
SUBTIT	SWELL	TIMCUT
UPDATE	UPDATP	UPDTE
VOLUME	XLAME	

1) MAIN routine

MAIN routine must be programmed by the user when the ACTIVE-II is run. (See the section CORE STORAGE or CONTROL CARDS.)

Sample MAIN routine:

```
COMMON /DIM/ IA, IAM
DIMENSION A(5000), AM(10000)
IA=5000
IAM=10000
CALL ACTIVE (A,AM)
STOP
END
```

2) Subroutine ACTIVE (A,AM)

This subroutine ACTIVE controls the calls to all subroutines included in the ACTIVE-II code. The calling flow is described in chapter 4.

The time step interval is determined internally in the ACTIVE. The maximum changes in time step interval, average linear power, restructuring, fuel creep, and cladding creep allowed to take

place within a given time step are limited to specified values. If any of the computed changes exceed these limits, the time step interval is cut internally by this subroutine ACTIVE.

The limiting conditions are:

1. Time step interval must not exceeds the CX times DELTO.
Presently, CX=2.0, DELTO=preceding time interval (hrs).
(Refer to the subroutine TIMCUT.)
2. Average linear power change must not exceeds the EPSPOW.
Presently, EPSPOW=50.0 (w/cm).
(Refer to the subroutine TIMCUT.)
3. Restructuring velocity must not exceeds the EPSSPD.
EPSSPD is the input value. If zero, EPSSPD is set to 0.01 (cm/sec).
(Refer to the subroutine CHKSPD.)
4. Fuel creep increment must not exceeds the EPSCRF. EPSCRF is the input value. If zero, EPSCRF is set to 0.05.
(Refer to the subroutine CHECK.)
5. Cladding creep increment must not exceeds the EPSCRC. EPSCRC is the input value. If zero, EPSCRC is set to 0.01.
(Refer to the subroutine CHECK.)

If any of these limits are exceeded, the time step interval is halved, and the power and coolant temperature changes are adjusted accordingly.

Called from : MAIN (user written program)

Functions used : DATE, STIMER (library)

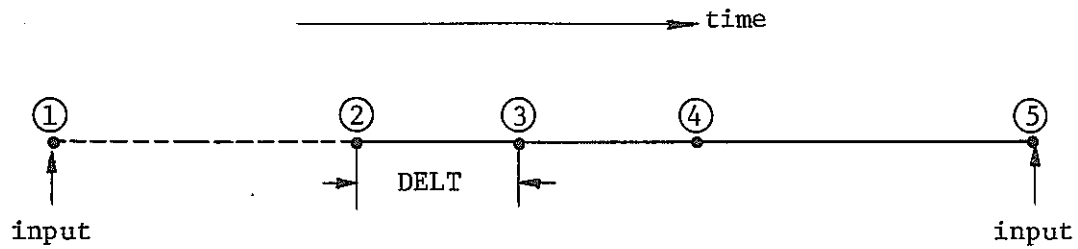
Commons required: ALWAYS, OTHERS, CONST, CAPSUL, IOFIL, FBRATR,
DIM, PLTOPT, DAY

Variables required:

A - used for all the variables except these for
matrix.

AM - used only for the variables related to the matrix.

Variables required:



Point of time

- ① Time specified on the preceding input card.
- ② Preceding step at which the calculation has just been done. (step number = IT-1)
- ③ Current time step at which the calculation is being done. (step number = IT)
- ④ Time once calculated but not accepted due to criteria for restructuring velocity and for the creep increment. (limitation 2.,4.,5.)
- ⑤ Time specified on the current input card.

Variables used at the above each point of time are as follows.

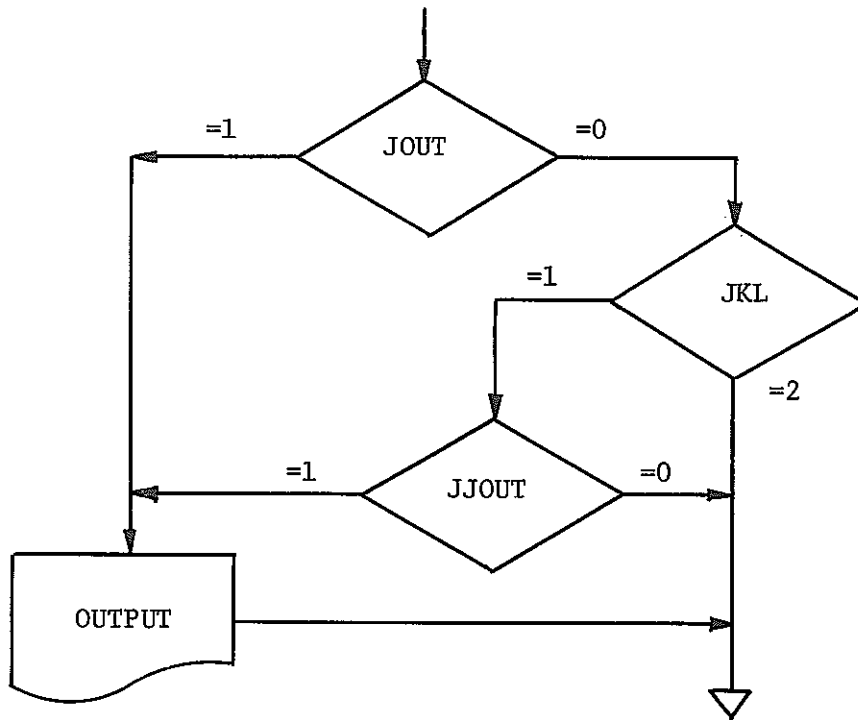
Point of time	②	③	④	⑤
time (hr)	TIP	TIM	YTIM	TIMO
average linear power (w/cm)	PPOWA	POWAV	YPOWAV	POWAO
coolant inlet temperature (°C)	PTIN	TIN	YTIN	TINO
coolant outlet temperature(°C)	PTOUT	TOUT	YTOUT	TOUTO
average external pressure on cladding (kg/cm ²)	PPO	PO	YPO	POO

The time step interval, DELT, is determined as (TIM-TIP).

IT - step counter for calculation. (step number)

ITPLOT - step counter for plotting.

Output timing is controlled by the input value, JOUT, and the internally determined values, JKL, JJOUT.



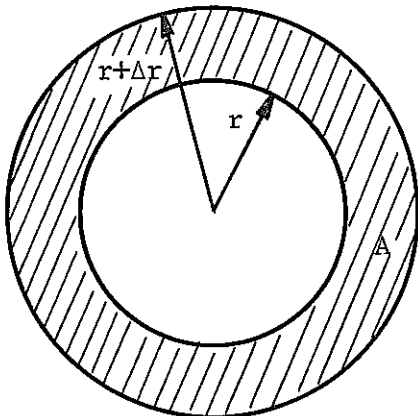
JKL - is set to 2, when the time step is cut internally due to the limiting conditions 1 or 2 as mentioned before. Otherwise, JKL is set to 1.

JJOUT - is set to 0, when the time step is cut due to the limiting conditions 3, 4 or 5 as mentioned before. Otherwise, JJOUT is set to 1.

JCR - is set to 1 or 2, when the fuel cracking or crack healing occurred, respectively. Otherwise, JCR is set to 0.
(Refer to the subroutine CRACK)

3) Function AREA (R, DR)

The AREA calculates the area between the circle with radius R and the concentric circle with radius R+DR.



$$A = (r+\Delta r)^2 - r^2 = \Delta r(2r+\Delta r)$$

Called from : CLTMP

Variables required:

R - radius r. (cm)

DR - Δr . (cm)

Variables changed :

AREA - $\Delta r(2r+\Delta r)$ (cm^2)

4) Function ATBFUN (T, DT, I)

This function calculates the thermal expansion in any of the three fuel regions, i.e., columnar grain region, equiaxed grain region, and undisturbed region.

The thermal expansion is computed by integrating the thermal coefficient of expansion from the room temperature to the temperature $T+DT/2$.

$\alpha(T)$ = $a_0 + a_1T$: thermal coefficient of expansion.
($^{\circ}\text{C}^{-1}$)

T : temperature ($^{\circ}\text{C}$)

Y = 20 $^{\circ}\text{C}$: room temperature

$$\begin{aligned} \int_Y^{T+\frac{\Delta T}{2}} \alpha(T) dT &= \int_Y^{T+\frac{\Delta T}{2}} (a_0+a_1T) dT = (a_0T + \frac{a_1}{2} T^2) \\ &+ (a_0+a_1T) \frac{\Delta T}{2} - (a_0Y + \frac{a_1}{2} Y^2) \end{aligned}$$

Called from : FLTMP

Commons required : OTHERS

Variables required:

T - temperature. ($^{\circ}\text{C}$)

DT - ΔT . ($^{\circ}\text{C}$)

Y - room temperature. (20 $^{\circ}\text{C}$)

I - = 1, indicator of columnar region.

= 2, indicator of equiaxed region.

= 3, indicator of undisturbed region.

ALFO - a_0 in the expression a_0+a_1T .

($^{\circ}\text{C}^{-1}$) (refer to the Common OTHERS)

ALF1 - a_1 in the expression a_0+a_1T .

($^{\circ}\text{C}^{-1}$) (refer to the Common OTHERS)

Variables changed:

$$ATBFUN = \int_Y^{T+\frac{\Delta T}{2}} \alpha(T) dT . \quad (-)$$

5) Subroutine AXIAL (X0, Y0, ID)

This subroutine AXIAL is used to plot an axial section number at the right-hand side of the graph.

Called from : PLOT4, PLTOR

Subroutines called: SYMBOL, NUMBER (library)

Variables required:

X0,Y0 - coordinates where the characters 'AXIAL SECTION='
are plotted. (mm)

ID - axial section number.

6) Subroutine AXISD (MX, XMN, DX)

The AXISD draws the latitudinal axis with regard to distance with graduations.

Called from : PLTOR

Subroutines called: PLOT, NEWPEN, NUMBER (library)

Variables required:

MX - number of graduations.

XMN - interval between graduations.

DX - scaling factor.

7) Subroutine AXIST (MX, XMN, DX)

The AXIST draws the latitudinal axis with regard to time with graduations.

Called from : PLTOR

Subroutines called: PLOT, NEWPEN, NUMBER (library)

Variables required:

MX - number of graduations.

XMN - interval between graduations.

DX - scaling factor.

- 8) Subroutine BCALC (B, ATBF, ATBCL, TPR, TPC, TPZ, PRS, RF, RC, P, BM, G, PB, GB, VOL, RH, PLPR, CVPR, PO, PTOP, DELT, TSR, TSC, TSZ, FLUX, TBCL, KFRIC, SWG, SWS, EHP, EPR, EPC, EPZ, MOUNT, IFRIC, NS, RB)

The BCALC provides the vector containing the right-hand sides of the stress-strain equations, i.e., stress-strain equations except for that of the plenum section.

Called from : ACTIVE
 Functions used : ALOG (library)
 Commons required : ALWAYS, OTHERS
 Variables required:

ATBF, ATBCL, TPR, TPC, TPZ, PRS, P, BM, G, PB, GB, VOL, RH, TSR, TSC, TSZ, FLUX, TBCL, KFRIC, SWG, SWS, EHP, EPR, EPC, EPZ, RB - refer to the Symbol Dictionary (Appendix C).
 RF - radii of the region boundaries in the fuel. (cm)
 RC - radii of the region boundaries in the cladding. (cm)
 PLPR - plenum pressure. (kg/cm²)
 CVPR - central void pressure. (kg/cm²)
 PO - average external pressure on cladding. (kg/cm²)
 PTOP - axial force on cladding. (kg)
 DELT - time step interval. (hr)
 MOUNT - switch used when the starting condition of the status of pellet-cladding gap is not accepted.
 IFRIC - indicator of status of pellet-cladding gap.
 NS - axial section number.

Variables changed:

B - vector containing the right-hand sides of the stress-strain linear equations except for those of the plenum section.

- 9) Subroutine BCALCP (B, ATBCL, TPR, TPC, TPZ, RC, P, G, PB, GB, VOL, RH, TBCL, FLUX, KFRIC, CRIF, DELT, TSR, TSC, TSZ, PLPR, PTOP, PO, EPR, EPC, EPZ, NS)

The BCALCP provides the vector containing the right-hand sides of the stress-strain equations only for the plenum section.

Called from : ACTIVE
 Functions used : ALOG (library)
 Commons required : ALWAYS, OTHERS

Variables required:

ATBCL, TPR, TPC, TPZ, P, G, PB, GB, VOL, RH, TBCL, FLUX,
 KFRIC, TSR, TSC, TSZ, EPR, EPC, EPZ, CRIF - refer to the
 Symbol Dictionary (Appendix C).

RC - radii of region boundaries in the cladding. (cm)
 DELT -- time step interval. (hr)
 PLPR - plenum pressure. (kg/cm)
 PTOP - axial force on cladding. (kg)
 PO - external pressure on cladding. (kg/cm²)
 NS - axial section number.

Variables changed :

B - vector containing the right-hand sides of the
 stress-strain linear equations for the plenum section.

10) Subroutine BEGIN (XM, ZFLX, RH, RHOP, CPU, LM, EHP, SWS, GS, GR,
 RVB, RCB, REB, RUB, QSF, QSFF, QSFP, POW, XCPI,
 XCPO, BU, HG, KFRIC, FLZO, FLZ, PRS, FLNC, CLZO,
 CLZ, FL, CL, DXM, SWG, RCI, RCO, RVBO, RCBO, REBO,
 RUBO, RCIO, RCOO, XMINT, RDIST, EPSEQ, SWS3, LMR,
 LMC, LMZ)

This subroutine is used for initialization of some variables
 before any of calculations is performed.

Called from : ACTIVE
 Commons required : ALWAYS, OTHERS, CONST, CAPSUL

Variables required:

RHOP, RVB, RCB, REB, RUB, QSFF, QSFP, FLZO, CLZO, RCI, RCO -
 refer to the Symbol Dictionary (Appendix C).

RCPI - capsule inner radius. (cm)
 RCPO - capsule outer radius. (cm)
 RHOC - cladding density. (g/cc)

PI - π

FA11, FA12, FA21, FA22, FA31, FA32 - refer to the Common CONST.

Variables changed (initialized):

XM, ZFLX, RH, LM, EHP, SWS, GS, GR, QSF, QSFF, POW,
 XCPI, XCPO, BU, HG, KFRIC, FLZ, PRS, FLNC, CLZ, DXM, SWG,
 RVBO, RCBO, REBO, RUBO, RCIO, RCOO, XMINT, RDIST, EPSEQ,
 SWS3, LMR, LMC, LMZ - refer to the Symbol Dictionary. (Appendix C)
 FL - total length of the fuel. (cm)
 CL - total length of the cladding. (cm)
 ALFO, ALF1, SMQ, SMQO - refer to the Common OTHERS.
 XEWTON - refer to the Common CONST.

11) Subroutine BURN (BU, POW, XM, EPZ, QSF, QSFP, DELT)

This is the subroutine which calculates the burn-up of the fuel. 2.246×10^{19} fissions per gram of the fuel are assumed to be equivalent to one atomic-percent burn-up. And 1 watt is equivalent to 3.1×10^{10} fissions per second. Using these constants and the linear power, the burn-up of the fuel is computed.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS

Variables required:

POW - average linear power during the time step. (w/cm)
 XM - mass. (g/cm)
 QSFP - axial power shape. (-)
 DELT - the time step interval. (hr)
 BUCNS - constant used for conversion from (w/cm) to
 (atomic-percent). ($3.1 \times 10^{10} \times 3600 / 2.246 \times 10^{19}$)
 PDBU - linear power decrement due to burn-up.
 (1/atomic-percent)

Variables changed :

BU - burn-up. (atomic-percent)
 QSF - axial power distribution factor. (g/cm)
 SMQ - summation of axial power shapes multiplied by mass.
 (g/cm)
 BUB - average burn-up of the fuel. (a/o)

12) Subroutine CHECK (DEQP, EF, EC, ISW)

The subroutine CHECK is used to check whether the equivalent creep-strain increment satisfies the given criterion or not. If the criterion is satisfied, the tentative time interval is accepted. If not, the time interval is halved in the subroutine ACTIVE.

Called from : ACTIVE

Commons required : ALWAYS

Variables required:

DEQP - equivalent creep-strain increment in the tentative time interval.

EF - criterion for DEQP in the fuel.

EC - criterion for DEQP in the cladding.

NN - total number of axial sections.

NFR - total number of radial meshes in the fuel.

NCL - total number of radial meshes in the cladding.

Variables changed :

ISW - indication of acceptance of the time step interval.
If ISW=1, smaller time interval is required. If
ISW=0, the tentative time step interval is accepted.

13) Subroutine CHKSPD (DXM, EPS, NN, ISW, DELT)

This subroutine is used to check whether the migration velocities of each region interfaces satisfy the given criterion or not. If the criterion is satisfied, the tentative time step interval is accepted. If not, the time interval is halved in the subroutine ACTIVE.

Called from : ACTIVE

Variables required:

DXM - migration velocity in the tentative time step interval. (cm/sec)

EPS - criterion for DXM. (cm)

NN - total number of axial sections.

DELT - tentatively decided time step interval. (hr)

Variables changed :

ISW - indication of acceptance of the time step interval.
 If ISW=1, smaller time step interval is required.
 If ISW=0, the tentative time step interval is accepted.

14) Subroutine CLEAR (IFRIC, JFRIC, NS)

This subroutine is used for setting the initial assumption of fuel-cladding gap condition. Fuel-cladding gap is assumed to be open.

Called from : ACTIVE

Variables required:

JFRIC - gap condition at the preceding time step.
 =0, open
 =1, stick
 =2, slip
 NS - axial section number.

Variables changed :

IFRIC - gap condition to be assumed at the current time step.
 =0, open
 =1, stick
 =2, slip

15) Subroutine CLTMP (POW, TO, RO, RI, R, T, TB, TBA, ATB, ATBA, RCLO, IT, RB)

The CLTMP is used to determine the temperature and the thermal expansion of the cladding and the capsule (if the fuel-element is capsulated).

First, determined are the radii of interest. The cladding or the capsule is equally divided into NCL meshes.

The temperature distribution, i.e., the temperature at each region boundary, of the cladding (or the capsule) is calculated using the given cladding (or the capsule) outer temperature. (See chapter 3.1-3) The cladding (or the capsule) outer temperature is determined in the subroutine GOLDN.

When the temperature distribution across the cladding (or the capsule) is known, the average temperature and average thermal expansion are determined.

Called from : ACTIVE
 Functions used : ALOG, SQRT (library), AREA
 Commons required : ALWAYS, OTHERS, CONST
 Variables required:

POW -	average linear power. (w/cm)	
TO -	temperature at the outside of the cladding or the capsule. (°C)	
RO -	outer radius of the cladding or the capsule. (cm)	
RI -	inner radius of the cladding or the capsule. (cm)	
CK1 -		} refer to the Common CONST described in the Common Dictionary (Appendix B).
CK2 -		
CA1 -		
CA2 -		
A -	same as XEWTON	
PI -	π	
IT -	time step number	

Variables changed :

R - radius of each region boundary. (cm)
 T - temperature at the radius R. (°C)
 TB - average temperature of the mesh. (°C)
 TBA - average temperature of the cladding or the capsule. (°C)
 ATB - average thermal expansion of the mesh. (-)
 ATBA - average thermal expansion of the cladding or the capsule. (-)
 RCLO,RB - refer to the Symbol Dictionary (Appendix C).
 RCL,TCL,ATCL - radius, temperature, and thermal expansion respectively used for the numerical integration by NEWTON-COTES-6.

16) Subroutine COEF (OTOM, N)

The numerical coefficients in the formula for fuel conductivity are calculated as a function of OTOM (fuel stoichiometry).

Called from : FLTMP

Commons required : ALWAYS, CONST, FBRATR

Variables required:

OTOM - O/M ratio (i.e., X in UOx) according to fuel stoichiometry.

N - axial section number.

JFA - option for reactor model.

=1, for FBR

=2, for ATR

=3, for PWR

=4, for BWR

NOK - option number for fuel conductivity formula.

Variables changed :

FKA1, FKB1, FKC1 -	} refer to the Common CONST described in the section Common Dictionary.
FKA2, FKB2, FKC2 -	
FKA3, FKB3, FKC3 -	

17) Subroutine CORTMP (TOUTER, TOUTO, PTOUR, ISW, TOUT)

The subroutine CORTMP is used for storing/retrieving the temperature at outside of the cladding. Storing/retrieving is needed when the time step interval is halved due to the criteria.

Called from : ACTIVE

Commons required : ALWAYS, IOFIL

Variables required/changed:

ISW - indicator of store/retrieval.

TOUT - coolant outlet temperature. (°C)

TOUTER, TOUTO, PTOUR - refer to the Symbol Dictionary.

18) Subroutine CRACK (SIGR, SIGC, SIGZ, BM, G, TBF, TPR, TPC, TPZ, TSR, TSC, TSZ, PRS, LM, LMR, LMC, LMZ, DELT, NS, JCR)

The CRACK computes whether fuel cracking would occur in any of the three principal directions.

A crack occurs when the stress component exceeds the ultimate tensile stress. The changes in elastic strain due to cracks are compensated by equal changes in plastic strain and volumetric expansion.

(See chapter 3.8)

The cracks are assumed to heal when the stresses in the region become compressive.

If the fuel cracking or crack healing occurs during the time step, the Young's modulus and the equations using it are re-computed.

Called from : ACTIVE

Commons required : ALWAYS, XCRAK

Variables required:

SIGR, SIGC, SIGZ, BM, G, TBF, PRS - refer to the Symbol Dictionary.

DELT - time step interval. (hr)

NS - axial section number.

THIC, HEAL1, HEAL2, SIGFO, SIGF1 - refer to the Common XCRAK.

Variables changed :

TPR, TPC, TPZ, TSR, TSC, TSZ, LM, LMR, LMC, LMZ - refer to the Symbol Dictionary.

JCR - =1, if any crack occurred during the time step.
=0, otherwise.

Significant internal variables:

SIGF - pellet fracture strength (kg/cm^2)
(ultimate tensile stress).

19) Subroutine CREEC (EQSIG, DEQP, TBC, FLNC, FLUX, DELT, N)

The equivalent creep-strain increments for the cladding regions are computed according to the creep law for cladding with associated constants.

Called from : ACTIVE

Functions used : EXP, SINH (Library)

Commons required : ALWAYS, OTHERS, XCREC, FBRATR

Variables required:

EQSIG, TBC, FLNC, FLUX - refer to the Symbol Dictionary.

DELT - time step interval. (hr)

N - axial section number.

A, Q, AE, BE, CE, FM, FN - refer to the Common XCREC.

NOC - option number for cladding creep formula. (input)
(See chapter 5 for option)

JFA - option number for reactor model. (Refer to the
Common FBRATR)

RKR - ideal gas constant. (cal/mol °R)

Variables changed :

DEQP - equivalent creep-strain increment.

- 20) Subroutine CREEF (EQSIG, DEQP, TBF, POW, FD, FLUX, FTHP, LM, DELT,
N, G, EPSEQ)

The subroutine CREEF calculates the equivalent creep-strain
increments for the fuel regions according to the creep law for
fuel with associated constants.

Called from : ACTIVE

Functions used : EXP (Library)

Commons required : ALWAYS, OTHERS, XCREF

Variables required:

EQSIG, TBF, POW, FD, FLUX, LM, G, EPSEQ - refer to the
Symbol Dictionary.

DELT - time step interval. (hr)

N - axial section number.

A0, A1, CFR, B, Q, Q1, FM, FN, GD - refer to the Common XCREF

Variables changed :

DEQP - equivalent creep-strain increment.

FTHP - thermal creep component.

- 21) Subroutine EDTGRS (IT, TIM, POWAV, TIN, TOUT, DELT, POW, FLUX, BU
TNA, XCPO, XCPI, TCPI, TCPO, RCO, RCI, TCLO,
TCL, FLZ, CLZ, FL, CL, OTOM, CPU, RHOT, RH,
XM, RVB, RCB, REB, RUB, RF, RC, HG, HC, TOGAS,
GR, TFR, TBFA, ATBFA, TBCLA, ATBLA, RCP, TCP,
TBCPA, ATBPA, FLNC, GS, PLPR, CVPR, FLO, CLO,
BUM, PO, FD, FDP)

All variables in the argument are edited and printed to the
printer before stress-strain matrix solving.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, CAPSUL, COMPLT, IOFIL

Variables required:

POW, FLUX, BU, TNA, XCPO, XCPI, TCPI, TCPO, RCO, RCI, TCLO,
TCL, FLZ, CLZ, OTOM, CPU, RHOT, RH, XM, RVB, RCB, REB,
RUB, HG, HC, GR, TFR, TBFA, ATBFA, TBCLA, ATBLA, RCP, TCP,
TBCPA, ATBPA, FLNC, GS, BUM, FD, FDP - refer to the Symbol
Dictionary.

IT - time step number.

TIM - time (hr)

POWAV - average linear power (w/cm)

TIN - coolant inlet temperature (°C)

TOUT - coolant outlet temperature. (°C)

DELT - time interval. (hr)

FL - total fuel length. (cm)

CL - total cladding length. (cm)

RF - boundary radius of each fuel mesh. (cm)

RC - boundary radius of each cladding mesh. (cm)

TOGAS - total gas in plenum. (mol)

PLPR - plenum pressure. (kg/cm²)

CVPR - central void pressure. (kg/cm²)

FLO - initial total fuel length. (cm)

CLO - initial total cladding length. (cm)

PO - average external pressure on cladding. (kg/cm²)

- 22) Subroutine EDTINP (RVB, RCB, REB, RUB, RCI, RCO, QSFF, QSFP, OTOM,
CPU, RHOP, F, XCPI, XCPO, FLZ, CLZ, FL, CL, PTOP,
PO)

Input variables are printed to the printer.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, CAPSUL, PLTOPT, IOFIL

Variables required:

RVB, RCB, REB, RUB, RCI, RCO, QSFF, QSFP, OTOM, CPU, RHOP,
F, XCPI, XCPO, FLZ, CLZ - refer to the Symbol Dictionary.

FL - total fuel length. (cm)

CL - total cladding length. (cm)
 PTOP - axial force on cladding. (kg)
 PO - average external pressure on cladding. (kg/cm²)

23) Subroutine EDTMTX (EPR, EPC, EPZ, SIGR, SIGC, SIGZ, TPR, TPC, TPZ, TSR, TSC, TSZ, RDIS, CRIF, RCP, TCP, TBCP, ATBCP, RCL, TCL, TBCL, ATBCL, YNG, POS, SHR, BM, RF, TF, TBF, ATBF, PRS, EHP, TPRF, TPRC, TPCF, TPCC, TPZF, TPZC, PRES, KFRIC, DPR, DPC, DPZ, DSR, DSC, DSZ, POSB, SHRB, SWG, SWS, N, RPM, TPM, GS, CPU, LM, LMR, LMC, LMZ, RCOO)

After the stress-strain matrix is solved, the stress, strain, displacement, and detailed informations are printed out.

Called from : ACTIVE

Functions used : TTIMER (Library)

Commons required : ALWAYS, XFRIC, CAPSUL, COMPLT, IOFIL

Variables required:

EPR, EPC, EPZ, SIGR, SIGC, SIGZ, TPR, TPC, TPZ, TSR, TSC, TSZ, RDIS, CRIF, RCP, TCP, TBCP, ATBCP, RCL, TCL, TBCL, ATBCL, BM, TBF, ATBF, PRS, EHP, KFRIC, SWG, SWS, GS, CPU, LM, LMR, LMC, LMZ, RCOO - refer to the Symbol Dictionary.

YNG - Young's modulus. (kg/cm²)

POS - Poisson's ratio. (-)

SHR - shear modulus. (kg/cm²)

RF - boundary radius of each fuel region. (cm)

TF - temperature at the boundary of each fuel region. (°C)

PRES - contact pressure at the boundary of each radial region. (kg/cm²)

DPR - radial creep strain increment during the time interval. (-)

DPC - circumferential creep strain increment. (-)

DPZ - axial creep strain increment. (-)

POSB - average Poisson's ratio. (-)

SHRB - average shear modulus. (kg/cm²)

RPM - same as RFRPM
 TPM - same as TFRPM } in the Symbol Dictionary.

N - axial section number.

24) Subroutine EQUISI (SK, SC, SZ, ES, PRS, N)

This subroutine is provided to calculate the equivalent stress, and the hydro-static pressure in the N-th axial section and all the radial regions, using the corresponding stresses.

Called from : ACTIVE

Functions used : SQRT (Library)

Commons required : ALWAYS

Variables required:

SR - the radial stress component. (kg/cm²)

SC - the circumferential stress component. (kg/cm²)

SZ - the axial stress component. (kg/cm²)

N - the axial section number.

NTR - the total number of radial regions in the fuel and cladding.

Variables changed :

ES - the equivalent stress. (kg/cm²)

PRS - the hydro-static pressure. (kg/cm²)

25) Subroutine FLCOND (XK, XK1, ZH, FP, FP95, T)

The thermal conductivity of each fuel region is calculated in this subroutine.

Called from : FLTMP

Commons required : ALWAYS, CONST

Variables required:

ZH - ρ/ρ_{th} , where ρ and ρ_{th} are the actual and theoretical densities of the fuel, respectively.

FP - $(1-P)/(1+\beta P)$, when P=porosity, and $\beta \approx 1$.

FP95 - FP for P=0.05, that is $FP95 = (1-0.05)/(1+0.05 \cdot \beta)$

T - temperature. (°C)

NOK - option number for fuel conductivity formula.

Variables changed :

XK - thermal conductivity of the fuel. (watt/cm.°C)

XK1 - temperature differential of XK.

- 26) Subroutine FLTMP (POW, TCL, RCI, RUB, REB, RCB, RVB, RH, DXM, XM, R, T, TB, ATB, TBA, ATBA, FD, RPM, TPM, TBPM, ATBPM, FDP, GDT, HG, F, OTOM, RHTHEO, DELT, RFO, IT, BU, RHOP, RB, EPR, EPC, EPZ, TPR, TPC, TPZ, TSR, TSC, TSZ, GS, SWG, SWS, EHP, WOLD, WNEW, XEPR, XEPC, XTPR, XTPC, XTPZ, XTSR, XTSC, XTSZ, XGS, XSWG, XSW, XEHP, XSW3, SWS3)

The temperature distribution and fuel restructuring are determined in the FLTMP.

The columnar grain region is equally divided into NCR meshes. The equiaxed grain region is equally divided into NER meshes, and for the undisturbed region, into NUR meshes.

The radial temperature distribution in the fuel is determined at the boundaries of NP regions of equal mass through integration of the one-dimensional heat-conduction equation. (See chapter 3.1-5)) The temperatures at the boundaries of NCR, NER, or NUR meshes are interpolated using those at the boundaries of NP regions of equal mass.

The temperature gradients at the interfaces of the NP regions of equal mass are stored for use in the subsequent calculation of plutonium migration.

After the boundary temperatures are determined, the average temperatures and thermal expansion of each region are computed.

The growth of the columnar grain region and the equiaxed grain region (i.e., fuel restructuring) are determined using the experimental formula. After the boundaries of the columnar and equiaxed regions have been determined, a final mass balance yields the radius of the central void

Some strains are modified considering the fuel restructuring.

Called from : ACTIVE
Subroutines called: COEF, FLCOND, OTMCAL
Functions used : ALOG, EXP (Library)

Variables required:

POW, TCL, RCI, RUB, REB, RCB, RVB, RH, XM, HG, F, BU, RHOP
 - refer to the Symbol Dictionary.
 RHTHEO- theoretical density. (g/cc) (same as RHOT in the
 Symbol Dictionary.
 CMI, DHR, PI, BURH1, BURH2 - refer to the Common OTHERS.
 IT - time step number.
 DELT - time interval. (hr)

Variables changed :

DXM, FD, TBPM, ATBPM, FDP, GDT, OTOM, RB, EPR, EPC, EPZ,
 TPR, TPC, TPZ, TSR, TSC, TSZ, GS, SWG, SWS, EHP, WOLD,
 WNEW, KEPR, XEPC, XTPR, XTPC, XTPZ, XTSR, XTSC, XTSZ, XGS,
 XSWG, XSWG, XEHP, XSWG3, SWS3 - refer to the Symbol Dictionary.

R-RFR

T-TFR

TB-TBF

ATB-ATBF

TBA-TBFA

ATBA-ATBFA

RPM-RFRPM

TPM-TFRPM

RFO-RFRO

in the Symbol Dictionary.

XML, CM - refer to the Common OTHERS.

27) Subroutine FLXCAL (POWAV, QSFF, ZFLX, FLNC, FLUX, DELT)

The FLXCAL computes the fast neutron flux and the fast
 neutron fluence.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS

Variables required:

POWAV - average linear power. (w/cm)
 QSFF, ZFLX - refer to the Symbol Dictionary.
 CFLX, SMQO, SMQ - refer to the Common OTHERS.
 DELT - time interval. (hr)

Variables changed :

FLUX, FLNC - refer to the Symbol Dictionary.

28) Subroutine FRICHK (RUB, RCI, DISP, PRES, FRIC, CRIF, KFRIC, KONV, IFRIC, NS)

The fuel-clad gap is first assumed to be open. The frictional force is zero, and the interface pressure equals to the plenum pressure. If the computed displacements of fuel and cladding indicate that the gap is closed, then, frictional force is checked. If the computed frictional force is larger than can be justified by the contact pressure and the specified static coefficient of friction, slip occurs at the fuel-clad interface. Otherwise, stick condition is assumed.

If the assumed condition conflicts with the computed one, the matrix is solved again with the modified condition,

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, XFRIC

Variables required:

RUB, RCI - refer to the Symbol Dictionary.

DISP - displacement computed through matrix solving. (cm)

PRES - contact pressure at the boundary of each region,
computed through matrix solving. (kg/cm^2)

FRIC - frictional force computed through matrix solving. (kg/cm)

NS - axial section number.

FST, GAP - refer to the Common XFRIC.

Variables changed :

CRIF, KFRIC - refer to the Symbol Dictionary.

KONV - convergence indicator.

IFRIC - fuel-clad gap indicator.

29) Subroutine GAPCON (RCI, RUB, HG)

The fuel-clad gap heat-transfer coefficient is computed as a function of gap size. This value is used in the calculation of fuel outer temperature.

Called from : ACTIVE

Function used : EXP (Library)

Common required : ALWAYS

Variables required:

RCI, RUB - refer to the Symbol Dictionary.

Variables changed :

HG - gap heat-transfer coefficient. (cal/hr·cm²·°F)

- 30) Subroutine GASOUT (RF, FLZ, POW, FD, TBF, VOL, EPR, EPC, EPZ, ATBF, SWS, GR, GS, GRT, DELT)

The subroutine GASOUT calculates the fractional gas-release rates, and gas-remaining rates for each of the fuel region. The release rate for any region is increased if the swelling of the region rises above a specified limit. (25% volume expansion)

The gas-release rate is multiplied by the amount of gas and by the time interval to determine the amount of gas released from, or that remaining in the region during the time step. The released gas is added to the previous total of released fission gas.

Called from : ACTIVE

Functions used : EXP (Library)

Commons required : ALWAYS, OTHERS, XGASO

Variables required:

POW, FD, TBF, VOL, EPR, EPC, EPZ, ATBF, SWS - refer to the Symbol Dictionary.

DELT - time interval. (hr)

AF, AN, CT, QR - refer to the Common XGASO.

A91 - refer to the Common OTHERS.

Variables changed :

GR - gas released. (mol)

GS - gas remaining. (mol)

GTR - total gas released. (mol)

- 31) Subroutine GOLDN (POWAV, QSF, TOUT, TIN, RCLO, POW, TCLO, RCPI, XCPO, AFNA, FL, HC, TNA, TCPO, TCPI, TOUTER, NSKIP, PO, FLZ)

The GOLDN computes the surface temperature at the outside of cladding. The coolant temperature at the middle of each axial section is determined by either Newton's interpolation method or linear interpolation between inlet and outlet temperature.

The linear power in each axial section is computed from the given average linear power for the fuel element. The outside temperature of the capsule (or cladding) is determined after the heat-transfer coefficient between the coolant and the fuel-element capsule (or cladding) is calculated.

Called from : ACTIVE

Functions used : ALOG, EXP (Library)

Commons required : ALWAYS, OTHERS, XGOLD, FBRATR, IOFIL

Variables required:

POWAV - average linear power for the fuel element. (w/cm)

QSF, TOUTER, FLZ - refer to the Symbol Dictionary.

TIN - coolant inlet temperature. ($^{\circ}\text{C}$)

TOUT - coolant outlet temperature. ($^{\circ}\text{C}$)

RCLO - outside radius of cladding. (cm)

RCPI - inside radius of capsule. (cm)

XCPO - outside radius of capsule. (cm)

AFNA - coolant flow area. (cm^2)

FL - fuel length. (cm)

NSKIP - =0, the coolant inlet, and outlet temperature is given (input).

=1, the cladding outer temperature is given (input).

PO - average external pressure on cladding. (kg/cm^2)

ASSO, ASS1, ANAO, ANA1, ANA2, CO, C1, C2 - refer to the Common XGOLD.

Variables changed :

POW, TCLO, HC, TNA, TCPO, TCPI - refer to the Symbol Dictionary.

32) Subroutine HEADER

The HEADER reads the title card until the *END card is encountered. The title card is printed soon after it is read before any other output is performed.

Called from : ACTIVE

Common required : ALWAYS, COMPLT, IOFIL

Variables changed :

TITLE - title

TITL - title for plotting.

33) Subroutine HKNS (TBCL, TSR, TSC, TSZ, FLX, DELT)

The HKNS uses the two formulae for the cladding swelling according to the option NOS to determine the cladding swelling.

Called from : ACTIVE

Functions used : ALOG, EXP (Library)

Commons required : ALWAYS, XHKNS, FBRATR

Variables required:

TBCL - average temperature of each region of the cladding.
(°C)

FLX - neutron flux. (n/sec·cm²)

DELT - time interval. (hr)

CS, BO, CO, C1, C2, C3 - refer to the Common XHKNS.

JFA - refer to the Common FBRATR.

Variables changed :

TSR, TSC, TSZ - refer to the Symbol Dictionary.

34) Subroutine INPUT (EPZ, TPR, TPC, TPZ, SIGR, SIGC, SIGZ, ADISP, RDISP, RVB, RCB, REB, RUB, RCI, RCO, QSFF, QSFP, F, EPR, EPC, TSR, TSC, TSZ, CPU, RHOP, OTOM, FL, CL, PTO, PO, EPSIN, CLZO)

The subroutine INPUT reads the input cards. (See Chapter 6.3)

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, CAPSUL, PLTOPT, IOFIL

Variables changed :

Refer to the Chapter 6.

35) Subroutine IOSET (IO)

The IOSET determines the file reference number for use of temporary output. Presently, these variables are not used.

Called from : ACTIVE

Commons required : ALWAYS

Variables changed :

IO - output file reference number.

- 36) Subroutine ITERAT (PRS, VOL, TBF, ATBF, SWG, EHP, RH, RF, GS, SWS, BM, EPR, EPC, EPZ, SWS3, SIGR, SIGC, SIGZ, RHTHEO, DELT, N, RHOP, BU, TSR, TSC, TSZ, XEHP, XSWG, PLPR, CVPR)

The ITERAT computes the fuel swelling due to gaseous fission products and the hot-pressing. The gaseous fission products in the undisturbed region are assumed to contribute to swelling in that region in a manner similar to solid fission products.

Called from : ACTIVE

Functions used : EXP (Library)

Commons required : ALWAYS, OTHERS, XITER

Variables required:

PRS, VOL, TBF, GS, SWS3, RHOP, BU - refer to the Symbol Dictionary.

RHTHEO - theoretical density. (g/cc)

CVPR - central void pressure. (kg/cm²)

DELT - time interval. (hr)

N - axial section number.

C,QRK - refer to the Common XITER.

PG, BURH1, BURH2 - refer to the Common OTHERS

Variables changed :

SWG, EHP, SWS, TSR, TSC, TSZ - refer to the Symbol Dictionary.

XEHP, XSWG - temporary use.

- 37) Subroutine LENGTH (FL, FLN, CL, CLN, FLZO, FLZ, CLZO, CLZ, EPZ)

The axial length of the fuel and the cladding is computed.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS

Variables required:

FLZO, CLZO, EPZ - refer to the Symbol Dictionary.

Variables changed :

FLZ,CLZ - refer to the Symbol Dictionary.

FL,FLN - total fuel length. (cm)

CL,CLN - total cladding length. (cm)

38) Subroutine MATRX (IA, JA, A, E, P, DEQP, EQSIG, RF, RC, G, PB, GB,
FLZ, KFRIC, MOUNT, IFRIC, NS, RB)

The MATRX determines the vector A (vector containing the values of the elements of the given matrix of interest in row-wise order), the vector IA (vector containing the pointers to the first element in each row of the matrix in row-wise order), and the JA (vector containing the column indices of the matrix in row-wise order) for use in the subsequent matrix solving by SL-MATH. (See appendix A.2)

When the first call to the subroutine MATRX during the time step, the initial assumption of the boundary condition for the fuel-cladding interface is that the gap is open.

After the matrix is solved by SL-MATH, and if the results represent that the gap is closed, the parts of A, IA, and JA related to the boundary condition are modified at the second or the third call to the MATRX.

Called from : ACTIVE

Functions used : ALOG (Library)

Commons required : ALWAYS, OTHERS

Variables required:

E, P, DEQP, EQSIG, G, PB, GB, FLZ, KFRIC, RB - refer to the
Symbol Dictionary.

RF - boundary radius of each fuel region.

RC - boundary radius of each cladding region.

MOUNT - count number that the MATRX is called.

IFRIC - indication of the fuel-cladding interface condition.

=0, the gap is open.

=1, stick condition.

=2, slip condition.

NS - axial section number.

Variables changed :

IA, JA, A - refer to the SL-MATH (See chapter A.2)

- 39) Subroutine MATRXP (IA, JA, A, E, P, DEQP, EQSIG, RC, G, PB, GB, CLZ, KFRIC, NS)

The subroutine MATRXP is used for the same purpose as the MATRX except that only the plenum section (i.e., only the cladding) is treated in this subroutine.

Called from : ACTIVE

Functions used : ALOG (Library)

Commons required : ALWAYS, OTHERS

Variables required:

E, P, DEQP, EQSIG, G, PB, GB, CLZ, KFRIC - refer to the Symbol Dictionary.

RC - boundary radius of each cladding region.

NS - axial section number.

Variables changed :

IA, JA, A - Same as those used in the MATRX, and refer to the SL-MATH.(See Chapter A.2)

- 40) Subroutine OTMCAL (OTOM, N)

The OTMCAL computes the burnup dependent O/M ratio. Presently, the values of OTOM are equal to the initial input value.

Called from : FLTMP

Commons required : ALWAYS

- 41) Subroutine PLENP (RVB, RCI, TFR, FLZ, GRT, FL, CL, TOUT, PLPR, CVPR, RUB, CLZ)

This subroutine is used to determine the central void pressure and the plenum pressure. The temperature of the gas in the central void and that of the gas in the plenum are volume averaged assuming the perfect gas law.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS

Variables required:

RVB, RCI, TFR, FLZ, RUB, CLZ - refer to the Symbol Dictionary.
 GRT - total gas released in the plenum. (mol)
 FL - total length of the fuel. (cm)
 CL - total length of the cladding. (cm)
 TOUT - coolant outlet temperature. (°C)
 RK - ideal gas constant. (kg·cm/mol·°K)

Variables changed :

PLPR - plenum pressure. (kg/cm²)
 CVPR - central void pressure. (kg/cm²)

- 42) Subroutine PLOT1 (XPOW, XRL, XTL, XT, XTLMX, XRLR, XTLR, XCPGP,
 XCPU, XBU, POW, RFRPM, TFRPM, RCB, REB, RUB, RCI,
 CPU, BU, PFR, TFR, IT, N)

This subroutine is used to store the information in the fuel region in preparation for plotting. The Unit conversion is also performed.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, XFRIC, PLTOPT

Variables required:

POW - average power. (w/cm)
 RFRPM - radii of concentric cylindrical shells of equal mass into which the fuel region is subdivided. (cm)
 TFRPM - temperature at the corresponding mesh point of RFRPM. (°C)
 RCB - radius of columnar and equiaxed zone interface. (cm)
 REB - radius of equiaxed and undisturbed interface. (cm)
 RUB - outer radius of the fuel. (cm)
 RCI - inner radius of the cladding. (cm)
 CPU - plutonium concentration in each concentric cylindrical shell.
 BU - burn-up. (atomic-percent)
 RFR - radii of concentric cylindrical shells of equal mesh interval into which the fuel region is subdivided. But each region, i.e., columnar, equiaxed, undisturbed region, is subdivided independently. (cm)

TFR - temperature at the corresponding mesh point of RFR.
(°C)

IT - counter for plotting.

N - axial section number to be plotted.

GAP - gap between fuel and cladding. (cm)

NCR - number of meshes into which the columnar region is subdivided.

NER - number of meshes into which the equiaxed region is subdivided.

NP1 - number of radial mesh points, including inner and outer boundaries, for plutonium migration.

Variables changed : all are used for plotting.

XPOW - same as POW at the specified axial section N. (w/cm)

XRL - same as RFRPM at the specified axial section N. (mm)

XTL - same as TFRPM at the specified axial section N.
(°C)

XT - surface temperature of the fuel at the specified axial section N. (°C)

XTLMX - maximum temperature in the fuel at the specified axial section N. (°C)

XRLR - same as RCB at the specified axial section N. (mm)

XTLR(IT,1) - temperature at the columnar and equiaxed boundary at the specified axial section N. (°C)

XTLR(IT,2) - temperature at the equiaxed and undisturbed boundary at the specified axial section N. (°C)

XCPGP - same as GAP. (mm)

XCPU - same as CPU at the specified axial section N.

XBU - same as BU at the specified axial section N.
(atomic-percent)

43) Subroutine PLOT2 (XSIGC, SIGC, VOL, IT, N)

The subroutine PLOT2 is used to store the average circumferential (hoop) stress in the cladding for plotting.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, PLTOPT

Variables required:

SIGC - circumferential stress component in the fuel and the

cladding. (kg/cm^2)

VOL - volume of the each region in the fuel and the cladding. (cm^3)

IT - counter for plotting.

N - axial section number to be plotted.

NCL - total number of radial meshes in the cladding.

NFR - total number of radial meshes in the fuel.

Variables changed : this is used for plotting.

XSIGC - volume averaged hoop stress in the cladding at the specified axial section N. (kg/mm^2)

- 44) Subroutine PLOT3 (XCSTN, XTSC, XTPC, RCO, RCOO, TSC, TPC, VOL, RDIS, IT, N)

This is used to store the strain in the cladding for plotting.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, PLTOPT

Variables required:

RCO - outer radius of the cladding. (cm)

RCOO - initial outer radius of the cladding. (cm)

TSC - circumferential swelling strain component in the fuel and the cladding.

TPC - circumferential creep strain.

VOL - volume. (cm^3)

RDIS - displacements of the radial region boundaries. (cm)

IT - counter for plotting.

N - axial section number to be plotted.

Variables changed : all are used for plotting.

XCSTN - $\Delta D/D$ of the cladding. (%)

XTSC - volume averaged circumferential swelling strain in the cladding. (%)

XTPC - volume averaged circumferential creep strain the cladding. (%)

- 45) Subroutine PLOT4 (RFRPM, RCB, REB, LMR, LMC, LMZ, IT, N)

The PLOT4 is used to plot the fuel cracking which would occur in any of the three principal directions at the specified axial section. The cracks in the radial direction are drawn, if exist, as

concentric circles in the circle representing the fuel pin. The concentric circle representing crack and the one representing region boundary are discriminated from each other by using a different colored pen.

The cracks in the circumferential direction are drawn, if exist, as arrowheads. These are also discriminated by using a different colored pen. The cracks in the axial direction are drawn, if exist, as lines in the rectangle which represents the fuel pin. The color is different from that of some boundaries.

Called from : ACTIVE

Functions used : SIN, COS (Library)

Subroutines called: PLOT, NEWPEN, SYMBOL, NUMBER, CIRCLE, AROHD, DASHPT (Library) AXIAL

Commons required : ALWAYS, OTHERS, PLTOPT, DAY

Variables required:

RFRPM - radii of concentric cylindrical shells of equal mass into which the fuel region is subdivided for plutonium migration. (cm)

RCB - radius of columnar and equiaxed region interface.(cm)

REB - radius of equiaxed and undisturbed region interface.(cm)

LMR - number of cracks in the radial direction.

LMC - number of cracks in the circumferential direction.

LMZ - number of cracks in the axial direction.

IT - time step counter.

N - axial section number to be plotted.

46) Subroutine PLOT5 (XTIME, XPLPR, XYP, ITP, IT, ITN, TIM, PLPR)

This subroutine is used to store the time step number and the real time to be plotted. The pressure in the plenum section and the gas release rate are also stored.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, PLTOPT, COMPLT

Variables required:

IT - counter for plotting.

ITN - time step counter.

TIM - time at the ITN time step. (hr)

PLPR - pressure in the plenum section. (kg/cm^2)
 YP - gas release rate from the fuel pin (from the whole axial sections). (%)

Variables changed : all are used for plotting.

ITP - same as IT.
 XTIME - same as TIM. (hr)
 XPLPR - same as PLPR. (kg/cm^2)
 XYP - same as YP. (%)

47) Subroutine PLTOR (XPOW, XRL, XTL, XT, XTLMX, XRLR, XTLR, XCPGAP, XCPU, XBU, XSIGC, XCSTN, XTSC, XTPC, XTIME, XPLPR, XYP, ITP, IT, ITN)

This is the plotting routine except for that of the fuel cracking.

Called from : ACTIVE

Subroutines called: PLOT, NEWPEN, SYMBOL, NUMBER (Library)
 SUBTIT, AXIAL, AXISD, AXIST

Commons required : ALWAYS, OTHERS, DAY, COMPLT, PLTOPT

Variables required: all are used for plotting.

XPOW - average linear power. (w/cm)
 XRL - radii of region boundaries in the fuel (mm)
 XTL - fuel temperature at the corresponding point of XRL. ($^{\circ}\text{C}$)
 XT - fuel pellet surface temperature. ($^{\circ}\text{C}$)
 XTLMX - maximum temperature in the fuel. ($^{\circ}\text{C}$)
 XRLR(1) - radius of columnar and equiaxed region boundary. (mm)
 XRLR(2) - radius of equiaxed and undisturbed region boundary. (mm)
 XTLR(1) - temperature at the point XRLR(1). ($^{\circ}\text{C}$)
 XTLR(2) - temperature at the point XRLR(2). ($^{\circ}\text{C}$)
 XCPGAP - clad-pellet gap. (mm)
 XCPU - plutonium concentration.
 XBU - burn-up (a/o)
 XSIGC - clad average hoop stress. (kg/mm^2)
 XCSTN - clad strain- $\Delta D/D$. (%)
 XTSC - average circumferential swelling strain in the cladding. (%)
 XTPC - average circumferential creep strain in the cladding. (%)

XTIME - time. (hr)
 XPLPR - plenum pressure. (kg/cm²)
 YYP - gas release rate. (%)
 ITP - time step number selected for plotting.
 IT - counter for plotting.
 ITN - time step counter.

- 48) Subroutine PLUTO (CP, RVB, RCB, REB, RP, TP, GDT, XM, RH, EPZ, DELT, D)

The PLUTO computes the radial migration of plutonium in the fuel by thermal and chemical diffusion.

The plutonium flux between each concentric region of equal mass is computed, and the result is used to compute the plutonium concentration in each region after the time interval.

Called from : ACTIVE
 Functions used : EXP (Library)
 Commons required : ALWAYS, OTHERS, IOFIL
 Variables required:

RVB, RCB, REB, GDT, XM, RH, EPZ - refer to the Symbol Dictionary.
 RP - radii of the boundaries of the NP equal mass radial regions. (cm)
 TP - temperature at the boundary of each equal mass region. (°C)
 DELT - time interval. (hr)

Variables changed :

CP - plutonium concentration in each equal mass region. (-)
 D - plutonium concentration change during the time interval. (1/sec)

- 49) Subroutine POWCUT (TIM, TIP, TIN, PTIN, TOUT, PTOUT, REPOW, PPOWA, PTOUT, TOUTER, DELTOU, DELTIM, DELIN, DELOUT, DELPOW, KUT, NSK, KUTCNT)

The change of the average linear power which is given from the input card is cut up according to the option KUT. In keeping with the power cut, the time and the coolant inlet, outlet temperatures

are also cut up.

Called from : ACTIVE

Commons required : ALWAYS

Variables required:

TIP - time of the previous time step. (hr)
 PTIN - coolant inlet temperature of the previous time step.
 (°C)
 PTOUT - coolant outlet temperature of the previous time step.
 (°C)
 PPOWA - average power of the previous time step. (w/cm)
 PTOUR - cladding outside temperature of the previous time
 step. (°C)
 TOUTER- cladding outside temperature. (°C)
 KUT - number to be cut up.
 NSK - =0, coolant inlet, outlet temperature is given.
 =1, cladding outside temperature is given.
 KUTCNT- counter used for cutting.

Variables changed :

TIM - time after cut up. (hr)
 TIN - coolant inlet temperature after cut up. (°C)
 TOUT - coolant outlet temperature after cut up. (°C)
 REPOW - average power. (w/cm)
 DELTOU- cladding outer temperature increment. (°C)
 DELTIM- time increment. (hr)
 DELTIN- coolant inlet temperature increment. (°C)
 DELOUT- coolant outlet temperature increment. (°C)
 DELPOW- power increment. (w/cm)

50) Subroutine PROPTY (CPU, RHOP, RHOT)

This subroutine prepares for the numerical constants associated with the properties.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, CONST, XCRAK, XCREC, XCREF,
 XFRIC, XGASO, XGOLD, XHKNS, XITER, XSWEL,
 XXLAM, FBRATR

Variables required:

CPU, RHOP - refer to the Symbol Dictionary.

Variables changed :

RHOT - refer to the Symbol Dictionary.

Variables included in the labelled common are referred to the Common Dictionary.

- 51) Subroutine RHOMOD (RH, RHOP, EPZ, RVB, RCB, REB, RUB, BU, XM, RHTHEO, RDIS, DXM, RVBO, RCBO, REBO, RUBO, RCI, RCO, XMINT, EPC, RDIST, EPR, XEPR, XEPC, R, WOLD, WNEW, IT, DELT)

The RHOMOD computes the boundary radius of each region, and the burnup dependent density of each radial region.

The minimum mass per unit length in columnar region is bounded by 10^{-3} (g/cm) to avoid numerical problems.

Called from : ACTIVE

Functions used : SQRT (Library)

Commons required : ALWAYS, OTHERS, IOFIL

Variables required:

RHOP, EPZ, BU, RDIS, DXM, RVBO, RCBO, REBO, RUBO, XMINT, EPC

EPR - refer to the Symbol Dictionary.

RHTHEO- theoretical density. (g/cc)

DELT - time interval. (hr)

BURH1, BURH2 - refer to the Common OTHERS.

Variables changed :

RH, RVB, RCB, REB, RUB, XM, RCI, RCO, RDIST - refer to the Symbol Dictionary.

- 52) Subroutine SUBTIT (TITLE)

This subroutine draws the title above each plotted picture.

Called from : PLTOR

Variables required:

TITLE - title to be plotted.

53) Subroutine SWELL (POW, SWS, FD, SWS3, DELT)

The SWELL computes the fuel swelling due to solid fission products by specifying the net solid volume created per fission and by multiplying by the number of fissions that have occurred in each fuel region.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, XSWEL

Variables required:

POW,FD- refer to the Symbol Dictionary.

DELT - time interval. (hr)

Variables changed :

SWS, SWS3 - refer to the Symbol Dictionary.

54) Subroutine TIMCUT (CX, DELTO, EPSPOW)

This subroutine decides the criterion for the time step interval itself and for the average power. The current time step interval must not exceed the value of CX times DELTO. The absolute value of the average power decrement or the increment must not exceed EPSPOW.

In the code ACTIVE-II, CX, DELTO, and EPSPOW are fixed as follows.

CX = 2.0

DELTO = 240.0 (hrs)

EPSPOW = 50.0 (w/cm)

Called from : ACTIVE

55) Subroutine UPDATE (SIGR, SIGC, SIGZ, SIGRO, SIGCO, SIGZO, EPR, EPC, EPZF, EPZC, EPRO, EPCO, EPZO, TPR, TPC, TPZ, TPRO, TPCO, TPZO, DSR, DSC, DSZ, TSRO, TSCO, TSZO, DISP, RDIS, N, EPSEQ)

After the stress-strain matrix equations are solved by using SL-MATH, the results are stored into the corresponding variables in the subroutine.

Called from : ACTIVE
 Functions used : SQRT (Library)
 Commons required : ALWAYS

Variables required:

SIGR - radial stress component. (kg/cm^2)
 SIGC - circumferential stress component. (kg/cm^2)
 SIGZ - axial stress component. (kg/cm^2)
 EPR - total radial strain. (-)
 EPC - total circumferential strain. (-)
 TPR - radial creep strain. (-)
 TPC - circumferential creep strain. (-)
 TPZ - axial creep strain. (-)
 EPZF - total axial strain of the fuel. (-)
 EPZC - total axial strain of the cladding. (-)
 DISP - displacement. (cm)

All variables listed above are the results through matrix solving.

N - axial section number.

Variables changed :

SIGRO-SIGR	}	in the Symbol Dictionary.
SIGCO-SIGC		
SIGZO-SIGZ		
EPRO-EPR		
EPCO-EPC		
EPZO-EPZ		

EPSEZ, RDIS - refer to the Symbol Dictionary.

56) Subroutine UPDATP (SIGR, SIGC, SIGZ, SIGRO, SIGCO, SIGZO, EPR, EPC, EPZC, EPRO, EPCO, EPZO, TPR, TPC, TPZ, TPRO, TPCO, TPZO, TSR, TSC, TSZ, TSRO, TSCO, TSZO, DISP, RDIS, N)

The UPDATP is used for the same purpose as the subroutine UPDATE except that the UPDATP treats only the plenum section.

Called from : ACTIVE
 Commons required : ALWAYS

Variables required:

SIGR, SIGC, SIGZ, EPR, EPC, TPR, TPC, TPZ, EPZC, DIST
 - same as mentioned in the UPDATE.
 N - axial section number.

Variables changed :

SIGRO, SIGCO, SIGZO, EPRO, EPCO, EPZO, RDIS
 - same as mentioned in the UPDATE.

- 57) Subroutine UPDTE (RVB, RCB, REB, RUB, RCI, RCO, BU, RDIST, DXM,
 FLNC, WOLD, YRVB, YRCB, YREB, YRUB, YRCI, YRCO,
 YBU, YRDIST, YDXM, YFLNC, YWOLD, ISW)

The subroutine UPDTE is used to store or retrieve the values of several variables to maintain the system. The subroutine ACTIVE decides the time step interval, using the given time histories. If the criterion is not satisfied, a computational control is returned to the starting point of time step loop after retrieving the values of the variables changed between the starting point and the criterion checking.

Called from : ACTIVE

Commons required : ALWAYS

Variables required:

ISW - a switch for controlling store and retrieval. If ISW=1, store procedure is selected. If ISW=2, retrieval procedure is selected.
 NN - the total number of axial sections, including a plenum section.
 NFR - the total number of radial regions in the fuel.

Variables changed :

RVB - central void radius. (cm)
 RCB - radius of columnar and equiaxed region interface.(cm)
 REB - radius of equiaxed and undisturbed region interface.(cm)
 RUB - outer radius of the fuel. (cm)
 RCI - inner radius of the cladding. (cm)
 RCO - outer radius of the cladding. (cm)
 BU - burn-up (atomic-percent).
 RDIST(N,1) - not used.
 (N,2) - accumulated migration of columnar and equiaxed

region interface caused by fuel restructuring.
(cm)

(N,3) - accumulated migration of equiaxed and undisturbed
region interface caused by fuel restructuring. (cm)

(N,4) - mass in unit length of the current time step.
(g/cm)

Where N means the axial section number.

DXM - migration velocity (rate of restructuring)
during the present time interval. (cm/sec)

FLNC - fluence. (n/cm²)

WOLD - mass in unit length. (g/cm)

YRVB, YRCB, YREB, YRUB, YRCI, YRCO, YBU, and YRDIST have
the same contents as those of corresponding variables
without the initial Y, except that the variables with
the initial Y have the starting value of the current
time step.

58) Subroutine VOLUME (VOL, RFR, RCL, FLZ, CLZ)

The VOLUME computes the volume of each concentric cylinder for
each axial section.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS

Variables required:

RFR, RCL, FLZ, CLZ - refer to the Symbol Dictionary.

Variables changed :

VOL - refer to the Symbol Dictionary.

59) Subroutine XLAME (TBC, TBF, E, P, G, RH, LM, BM, PB, GB, VOL, RHTHEO, N)

The XLAME computes the Poisson's ratio, Young's modulus, shear
modulus, and bulk modulus of each region for the fuel and the cladd-
ing. The Poisson's ratio and the Young's modulus are modified in
an attempt to represent the cracked material.

Called from : ACTIVE

Commons required : ALWAYS, OTHERS, XXLAM, FBRATR, IOFIL

Variables required:

TBF, RH, LM, VOL - refer to the Symbol Dictionary.

TBC - average temperature of each region in the cladding.
(°C)

RHTHEO- theoretical density. (g/cc)

N - axial section number.

EO, EP, TLG, ONU, ONUP, A1, A2, A3, A4 - refer to the Common
XXLAM.

Variables changed :

E, P. G, BM, PB, GB - refer to the Symbol Dictionary.

APPENDIX B: COMMON DICTIONARY

Appendix B: Common Dictionary

Some options, variables related to the dimensional size, variables which have fixed dimensions and some constants related to material properties are included in the following labelled commons.

B.1 Common list

Common Name	Explanation
ALWAYS	Variables related to the dimensional size and some options are included.
OTHERS	Some constants and variables having fixed dimensions are included.
CONST	Coefficients of thermal conductivity and of thermal expansion and of NEWTON-COTES-6 are included.
CAPSUL	Variables related to the capsule are included.
IOFIL	Definition of reference numbers of input/output files.
FBRATR	Reactor model (FBR, ATR, BWR, PWR) is specified.
DIM	Input sizes of variable dimensions are included.
PLTOPT	Options related to plotting are included.
DAY	Date is included.
COMPLT	Variables used for plotting are included.
XCRAK	Constants for fuel cracking are included.
XCREC	Constants for creep in the cladding are included.
XCREF	Constants for creep in the fuel are included.
XFRIC	Constants of friction are included.
XGASO	Constants for gas release are included.
XGOLD	Constants for coolant properties are included.
XHKNS	Constants for cladding swelling are included.
XITER	Constants for hot-pressing calculation are included.

Common Name	Explanation
XSWEL	Constant for solid volume per fission is included.
XXLAM	Constants for Poisson's ratio and Young's modulus are included.

Details are described in the section B.3.

B.2 Subroutines in which the commons are required.

The circles in the following table represent the correspondence between subroutines and commons.

Common Sub- routine	ALWAYS	CAPSUL	COMPLT	CONST	DAY	DIM	FBRATR	IOFIL	OTHERS	PLTOPT	XCRACK	XCREC	XCREP	XFRIC	XGASO	XGOLD	XHKNS	XITER	XSWEL	XXLAM
MAIN																				
ACTIVE	○	○		○	○	○	○	○	○	○										
AREA																				
ATBFUN									○											
AXIAL																				
AXISD																				
AXIST																				
BCALC	○								○											
BCALCP	○								○											
BEGIN	○	○		○					○											
BURN	○								○											
CHECK	○																			
CHKSPD																				
CLEAR																				
CLTMP	○			○					○											
COEF	○			○			○													
CORTMP	○							○												
CRACK	○										○									
CREEC	○						○		○			○								
CREEF	○								○				○							
EDTGRS	○	○	○					○	○											
EDTINP	○	○						○	○	○										

Common Sub- routine	ALWAYS	CAPSUL	COMPLT	CONST	DAY	DIM	FBRATR	IOFIL	OTHERS	PLTOPT	XCRAK	XCREC	XCREF	XFRIC	XGASO	XGOLD	XHKNS	XITER	XSWEL	XXLAM
EDTMTX	☐	☐	☐					☐						☐						
FLCOND	☐			☐																
FLTMP	☐			☐			☐	☐	☐											
FLXCAL	☐								☐											
FRICHK	☐								☐					☐						
GAPCON	☐																			
GASOUT	☐								☐						☐					
GOLDN	☐						☐	☐	☐							☐				
HEADER	☐		☐					☐												
NKNS	☐						☐										☐			
INPUT	☐	☐						☐	☐	☐										
IOSET	☐																			
ITERAT	☐								☐									☐		
LENGTH	☐								☐											
MATRX	☐							☐	☐					☐						
MATRXF	☐								☐											
OTMCAL	☐																			
PLENP	☐								☐											
PL0T1	☐								☐	☐				☐						
PLOT2	☐								☐	☐										
PLOT3	☐								☐	☐										
PLOT4	☐				☐				☐	☐										
PLOT5	☐		☐						☐	☐										
PLTOR	☐		☐		☐				☐	☐										
PLUTO	☐							☐	☐											
POWCUT	☐							☐												
PROPTY	☐			☐			☐		☐		☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
RHOMOD	☐							☐	☐											
SUBTIT																				
SWELL	☐								☐										☐	
TIMCUT																				
UPDATE	☐																			
UPDATP	☐																			
UPDTE	☐																			
VOLUME	☐								☐											
XLAME	☐						☐	☐	☐											☐

B.3 Definition of Variables in Common

* The column 'Subroutine' specifies the subroutine in which the value is set.

1) Common ALWAYS

Variable	Definition	Value	Subroutine*
NN	Total number of axial sections, including a plenum section	input	ACTIVE
NCR	Number of meshes into which the columnar grain region is subdivided.	input	ACTIVE
NER	Number of meshes into which the equiaxed grain region is subdivided.	input	ACTIVE
NUR	Number of meshes into which the undisturbed region is subdivided.	input	ACTIVE
NCL	Number of meshes into which the cladding is subdivided.	input	ACTIVE
NP	Number of regions, i.e., concentric cylindrical shells, of equal mass into which the fuel is subdivided for plutonium migration.	input	ACTIVE
NOK	Option number for fuel conductivity formula. (Refer to the section 5 Material Properties.)	input	ACTIVE
NOS	Option number for cladding swelling formula. (Refer to the section 5.)	input	ACTIVE
NOC	Option number for cladding creep formula. (Refer to the section 5.)	input	ACTIVE
NPLOT	Option for plotting. =1 for plotting. =0 for no plotting.	input	ACTIVE

Variable	Definition	Value	Subroutine*
NCRK	Option for cracking calculation. =1 for cracking calculation. =0 for no cracking calculation.	input	ACTIVE.
NTR	Total number of radial meshes in the fuel and cladding.	$NTR = NCR + NER + NUR + NCL$	ACTIVE
NFR	Total number of radial meshes in the fuel.	$NFR = NCR + NER + NUR$	ACTIVE
NTR1	$NTR + 1$	$NTR1 = NTR + 1$	ACTIVE
NTR2	$NTR + 2$	$NTR2 = NTR + 2$	ACTIVE
NNTR2	$NN \times (NTR + 2)$	$NNTR2 = NN \times (NTR + 2)$	ACTIVE
NNF	Total number of axial sections, excluding the plenum section.	$NNF = NN - 1$	ACTIVE
NCL1	Number of mesh points in the cladding.	$NCL1 = NCL + 1$	ACTIVE
NFR1	Number of mesh points in the fuel.	$NFR1 = NFR + 1$	ACTIVE
NP1	Number of mesh points for plutonium migration in the fuel.	$NP1 = NP + 1$	ACTIVE
IGEN	Number of columns (or rows) of the stress-strain matrix.	$IGEN = 12 + 10 \times NTR$	ACTIVE
IGEN1	$IGEN + 1$	$IGEN1 = IGEN + 1$	ACTIVE
NONZR	Number of non-zero elements in the matrix.	$NONZR = 24 + 56 \times NTR$	ACTIVE
IGENP	Number of columns (or rows) of the matrix for the plenum section.	$IGENP = 7 + 10 \times NCL$	ACTIVE
IGENP1	$IGENP + 1$	$IGENP1 = IGENP + 1$	ACTIVE
NONZRP	Number of non-zero elements in the matrix for the plenum section	$NONZRP = 12 + 56 \times NCL$	ACTIVE
N1C	} See the column 'Value'	NCL	ACTIVE
N2C		$2 \times NCL$	ACTIVE
N3C		$3 \times NCL$	ACTIVE

Variable	Definition	Value	Subroutine*
N4C		4×NCL	ACTIVE
N5C		5×NCL	ACTIVE
N6C		6×NCL	ACTIVE
N7C		7×NCL	ACTIVE
N8C		8×NCL	ACTIVE
N9C		9×NCL	ACTIVE
N10C		10×NCL	ACTIVE
N11C		11×NCL	ACTIVE
N12C		12×NCL	ACTIVE
N13C		13×NCL	ACTIVE
N14C		14×NCL	ACTIVE
N15C	See the column 'Value'	15×NCL	ACTIVE
N1		NTR	ACTIVE
N2		2×NTR	ACTIVE
N3		3×NTR	ACTIVE
N4		4×NTR	ACTIVE
N5		5×NTR	ACTIVE
N6		6×NTR	ACTIVE
N7		7×NTR	ACTIVE
N8		8×NTR	ACTIVE
N9		9×NTR	ACTIVE
N10		10×NTR	ACTIVE
N11		11×NTR	ACTIVE
N12		12×NTR	ACTIVE
N13		13×NTR	ACTIVE
N14		14×NTR	ACTIVE
N15		15×NTR	ACTIVE

2) Common OTHERS

Variable	Definition	Value	Subroutine
PG(3)	PG(1)=pressure due to surface tension of fission-gas bubbles for columnar grain region. PG(2)=pressure due to surface tension on-fission-gas bubbles for equiaxed grain region. PG(3)=pressure due to surface tension on fission-gas bubbles for undisturbed region. These are converted into the unit of kg/cm^2 in the subroutine PROPTY.	Input (psi.) PG(1)=3000., PG(2)=6000., PG(3)=1000000. are recommended.	INPUT
CML(2)	CML(1)=Constant C1 for restructuring velocity of columnar grain region outer boundary. CML(2)=Constant C1 for restructuring velocity of equiaxed grain region outer boundary. (See Eq. (6) and (7))	Input CML(1)= 1.92×10^7 (for FBR) CML(1)= 1.92×10^8 (for thermal reactor) CML(2)= 1.2×10^4 are recommended.	INPUT
DHR(2)	DHR(1)=Constant C2 for restructuring velocity of columnar grain region outer boundary. DHR(2)=Constant C2 for restructuring velocity of equiaxed grain region outer boundary. (See Eq. (6) and (7))	Input DHR(1)=64400. DHR(2)=44400. are the recommendations.	INPUT
ALFO(3)	Thermal coefficients of expansion for the columnar region (ALFO(1)), the equiaxed region (ALFO(2)), and the undisturbed region (ALFO(3)). (See section 5)	$(^\circ\text{C}^{-1})$	BEGIN and PROPTY
ALF1(3)	Thermal coefficients of expansion for the columnar region (ALF1(1)), the equiaxed region (ALF1(2)) and the undisturbed region (ALF1(3)).	$(^\circ\text{C}^{-2})$	BEGIN and PROPTY

Variable	Definition	Value	Subroutine
	(See section 5)		
XML(3)	XML(1)=0.0 XML(2)=mass in the columnar region. XML(3)=mass in the columnar and the equiaxed region.	Calculated (g/cm)	FLTMP
CM(2)	CM(1)=mass increment of columnar region during the time interval. CM(2)=mass increment of equiaxed region during the time interval.	Calculated (g/sec)	FLTMP
PI	π	3.1415927	PROPTY
PI2	2π	6.2831853	PROPTY
DP	Constant	1.0	PROPTY
GRAV	Constant for unit conversion	10^{-3}	PROPTY
SMQ	Summation of axial power shapes multiplied by mass.	Calculation (g/cm)	BEGIN and BURN
SMQO	Initial value of SMQ	Calc. (g/cm)	BEGIN
RHOC	Density of cladding	Input (g/cm ³)	INPUT
BURH1	Modifier for burn-up dependent density in the columnar region. (a/o) (See Eq. (9))	Input Recommended value: 20.0	INPUT
BURH2	Modifier for burn-up dependent density in the equiaxed zone. (a/o) (See Eq. (11))	Input recommended value: 20.0	INPUT
CFLX	Ratio of fast neutron flux to linear power. (n/cm ² /sec)/(watt/cm)	Input Default = 3.17×10^{12}	PROPTY or INPUT
CPOW	Conversion ratio from input power to linear power.	1.0	ACTIVE
BUCNS	Constant for unit conversion into (atomic-percent) from (w/cm).	$3.1 \times 10^{10} \times 3600 / 2.246 \times 10^{19}$	PROPTY
RK	Ideal gas constant	$8.31436 \times 1.0197 \times 10$ (kg·cm/mol·deg K)	PROPTY

Variable	Definition	Value	Subroutine
RKR	Ideal gas constant	8.31436/1.8/ 4.186 (cal/mol·deg R)	PROPTY
A91	Number of fissions/watt/hr.	1.116×10^{14}	PROPTY
GASI	Initial gas in the plenum section.	Input (moles)	INPUT
TCO	Constant of proportionality between the power and the outlet coolant temperature.	Input or calculated (degC/watt/cm)	INPUT or ACTIVE
PDBU	Decreasing rate of linear power due to burn-up.	Input (1/a/o)	INPUT
BUB	Average burn up of the fuel.	Calculated (atomic-percent)	BURN

3) Common CAPSUL

Variable	Definition	Value	Subroutine
RCPI	Inner radius of the capsule.	Inner (cm)	INPUT
RCPO	Outer radius of the capsule.	Input (cm)	INPUT
AFNA	Coolant flow area associated with the fuel element.	Input (cm ²)	INPUT

4) Common CONST

Variable	Definition	Value	Subroutine
FKA1 FKB1 FKC1	Numerical coefficients in the first (NOK=1) formula for fuel thermal conductivity. (See section 5)	Calculated	COEF
FKA2 FKB2 FKC2	Numerical coefficients in the second (NOK=2) formula for fuel conductivity. (See section 5)	Calculated	COEF
FKA3 FKB3 FKC3	Numerical coefficients in the third (NOK=3) formula for fuel conductivity. (See section 5)	Calculated	COEF
FA11 FA12	Coefficients of thermal expansion for the columnar region. (See section 5)	8.143×10^{-6} (FBR) 9.61×10^{-6} (ATR) 2.43×10^{-9} (FBR) 1.57×10^{-9} (ATR)	PROPTY
FA21 FA22	Coefficients of thermal expansion for the equiaxed region. (See section 5)	8.143×10^{-6} (FBR) 9.61×10^{-6} (ATR) 2.43×10^{-9} (FBR) 1.57×10^{-9} (ATR)	PROPTY
FA31 FA32	Coefficients of thermal expansion for the undisturbed region. (See section 5)	8.143×10^{-6} (FBR) 9.61×10^{-6} (ATR) 2.43×10^{-9} (FBR) 1.57×10^{-9} (ATR)	PROPTY
CK1 CK2	Constants for cladding thermal conductivity. (See section 5)	1.130 (FBR) 0.121 (ATR) 1.36×10^{-4} (FBR) 1.24×10^{-4} (ATR)	PROPTY
CA1 CA2	Constants for cladding thermal expansion. (See section 5)	17.0×10^{-6} (FBR) 5.67×10^{-6} (ATR) 4.25×10^{-9} (FBR) 1.7×10^{-9} (ATR)	PROPTY
XEWTON(7)	Coefficients for integration by NEWTON-COTES-6	XEWTON(1)=41/140 (2)=216/140 (3)=27/140 (4)=272/140 (5)=27/140 (6)=216/140 (7)=41/140	BEGIN

5) Common COMPLT

Variable	Definition	Value	Subroutine
YP	Gas release rate	Calc. (%)	EDTGRS
DDOD	$\Delta D/D$ of the cladding. ΔD -diameter change. D-initial diameter.	Calc. (%)	EDTGRS
TITLE(20)	Title identifying the problem.	Input.	HEADER

6) Common DAY

Variable	Definition	Value	Subroutine
DAYDAY	Time of date.	Month/day/year	DATE

7) Common DIM

Variable	Definition	Value	Subroutine
IA	Dimension size in words (4 bytes) for all the variables excluding the matrix solving. IA must be set at least as large as the storage space required for the problem. (See section 6)	Input	MAIN
IAM	Dimension size in words (4 bytes) for the variables for matrix solving. (See section 6)	Input	MAIN

8) Common FBRATR

Variable	Definition	Value	Subroutine
JFA	Identification of the reactor model to be calculated.	Input =1 for FBR =2 for ATR =3 for PWR =4 for BWR	ACTIVE

9) Common IOFIL

Variable	Definition	Value	Subroutine
IO5	Logical card input device number.	5	ACTIVE
IO6	Logical printer output device number.	6	ACTIVE
IO7	Logical punch output device number.	7	ACTIVE

10) Common PLTOPT

Variable	Definition	Value	Subroutine
IPFUL	Axial section number for plotting the information associated with the fuel.	Input	INPUT
IPSTN	Axial section number at which the cladding hoop strain is plotted.	Input	INPUT
IPSTS	Axial section number at which the cladding hoop stress is plotted.	Input	INPUT
IPCRK	Axial section number at which the fuel cracking is plotted.	Input	INPUT
NBUF	Storage (words) required for plotting. The plotting informations are written to output logical unit (77) every NBUF steps.	Input If 0, NBUF is set to 50 in ACTIVE.	INPUT
IPCNT	Switch controlling the plotting.	Calc.	ACTIVE

11) Common XRACK

Variable	Definition	Value	Subroutine
THLC	Critical temperature at which cracks are assumed to heal. (See section 3)	1400 °C	PROPTY
HEAL1	Critical compressive stress by which cracks are assumed to be healed, when the temperature is less than, or equal to THLC. (See section 3)	-7000.0 (kg/cm ²)	PROPTY

Variable	Definition	Value	Subroutine
HEAL2	Critical compressive stress by which cracks are assumed to be healed, when the temperature is greater than THLC. (See section 3)	-70.0 (kg/cm ²)	PROPTY
SIGF0	Numerical coefficient in the expression for σ_f ; fracture stress of pellet. (See section 3)	1000.0 (kg/cm ²)	PROPTY
SIGF1	Numerical coefficient in the expression for σ_f . (See section 3)	0.3 (kg/cm ² /°K)	PROPTY

12) Common XCREC

Variable	Definition	Value	Subroutine
CCA	} A Q Ae Be Ce m n in Eq. (17) for cladding creep rate. (See section 5)	2.7×10 ⁻¹¹	} PROPTY
QCC		95000(cal/mol)	
AE		10 ⁻⁸	
BE		2.2×10 ²⁰	
CE		4.3×10 ⁻³⁰	
FMC		7.0	
FNC		1.0	

13) Common XCREF

Variable	Definition		Value	Subroutine
CFAO	A_0	in Eq. (17) for creep rate for porous fuel. (See section 5)	1.83×10^{-3}	PROPTY
CFA1	A_1		1.07×10^6	
CFR	ϕ_0		3×10^{12} (fis/sec/cc)	
CFB	C		6×10^{-23}	
QCF	Q_0		14000 (cal/mol)	
QCF1	Q_1		11000 (cal/mol)	
FMF	m		4.5	
FNF	n		1.0	
GDF	d		10.0	

14) Common XFRIC

Variable	Definition	Value	Subroutine
FRICST	Static coefficient of friction. (See section 5)	0.8	PROPTY
FRICDY	Dynamic coefficient of friction. (See section 5)	0.6	PROPTY

15) Common XGASO

Variable	Definition	Value	Subroutine
AF	Number of gas atoms per fission.	0.246	PROPTY
AN	Avogadro's number.	6.02×10^{23}	PROPTY
CT	Constants for fission gas release rate. (See section 5)	4.48	PROPTY
QRG		8000	

16) Common XGOLD

Variable	Definition	Value	Subroutine
ASS0	Constants for thermal conductivity of capsule. (cal/hr/cm/°F)	75.9	PROPTY
ASS1		34.84×10^{-3}	
ANA0	Constants for thermal conductivity of sodium (cal/hr/cm/°F)	449	
ANA1		-0.155	
ANA2		17.29×10^{-6}	
AC0	Constants for thermal capacity of sodium. (cal/kg/°F)	192.053	
AC1		-440.061×10^{-4}	
AC2		1895.43×10^{-8}	

17) Common XHKNS

Variable	Definition	Value	Subroutine
HCS	Coefficients for cladding swelling. (See section 5)	10.0	PROPTY
CBO		9×10^{-35}	
HCO		4.028	
HC1		-3.712×10^{-2}	
HC2		1.0145×10^{-4}	
HC3		-7.879×10^{-8}	

18) Common XITER

Variable	Definition	Value	Subroutine
CI	Constants for hot-pressing calculation. (See section 5)	11.13×10^6	PROPTY
QRK		49600	

19) Common XSWEL

Variable	Definition	Value	Subroutine
SVPF	Net solid volume created per fission. (in ³)	0.7×10^{-24}	PROPTY

20) Common XXLAM

Variable	Definition	Value	Subroutine
EO	Constants in the formula for Young's modulus of fuel. (See section 5)	25.238×10^5	PROPTY
EP		-2.35	
TLG		22200	
ONU	Constants in the formula for Poisson's ratio at room temperature (25°C) (See section 5)	0.317	
ONUP		-0.46	
EA1	Constants in the formula for Young's modulus of cladding (See section 5)	2.0793×10^6 (FBR) 9.898×10^5 (ATR)	
EA2		-8.8592×10^2 (FBR) -9.265×10^2 (ATR)	
EA3	Constants in the formula for Poisson's ratio of cladding. (See section 5)	0.3015 (FBR) 0.325 (ATR)	
EA4		0.85×10^{-4} (FBR) 2.14×10^{-4} (ATR)	

APPENDIX C: SYMBOL DICTIONARY

Appendix C: Symbol Dictionary

The variable names listed in the following symbol table are not always identical to those used in each subroutine. But these are related to each other through the column 'Subroutine' in the symbol table and through the description of the corresponding subroutine in section 'SUBROUTINE LIST'.

The parenthesis in the column 'Symbol' represents the axial (left-side) and the radial (right-side) dimensional sizes of the symbol. The variables in the parenthesis are listed in the common ALWAYS in the section 'Common Dictionary'.

The variable in the bracket in the column 'Symbol' represents the location of the allocated symbol in the variable 'A'.

The column 'Subroutine' specifies the subroutine in which the symbol is referred.

Symbol	Definition	Subroutine	Unit
ADISP(NN,NTR) A[NADIS]	Initial axial displacements (in each of NN axial sections and for each of NTR regions).	INPUT	cm
ATBCL(NN,NCL) A[NATBCL]	Cladding thermal expansion averaged in the radial region.	BCALC,BCALCP, CLTMP,EDTMTX	—
ATBCP(NN,NCL) A[NATBCP]	Capsule thermal expansion averaged in the radial region.	CLTMP,EDTMTX	—
ATBF (NN,NFR) A[NATBF]	Fuel thermal expansion averaged in the radial region.	BCALC,EDTMTX, FLTMP,GASOUT, ITERAT	—
ATBFA (NN,3) A [NATBFA]	Fuel thermal expansion averaged over the each region. ATBFA(NN,1)-over the columnar region. ATBFA(NN,2)-over the equiaxed region. ATBFA(NN,3)-over the undisturbed region.	EDTGRS,FLTMP	—
ATBLA (NN) A[NATBLA]	Cladding thermal expansion averaged over the cladding.	CLTMP,EDTGRS	—

Symbol	Definition	Subroutine	Unit
ATBPA (NN) A[NATBPA]	Thermal expansion averaged over the capsule.	CLTMP,EDTGRS	—
ATBPM (NN,NP) A[NATBPM]	Thermal expansions averaged in the regions into which the fuel is subdivided for plutonium migration.	FLTMP	—
BM (NN,NTR) A[NBM]	Bulk modulus in the radial region of the fuel and the cladding.	BCALC,CRACK EDTMTX,ITERAT, XLAME	kg/cm ²
BU (NN) A[NBU]	Burn-up.	BEGIN,BURN,EDTGRS, FLTMP,ITERAT, PLOT1,RHOMOD, UPDTE	atomic- percent
BUM (NN) A[NBUM]	Burn-up.	EDTGRS	MWD/T
CLZ (NN) A[NCLZ]	Cladding length of each axial section.	BEGIN,EDTGRS, EDTINP,LENGTH, MATRXP,PLENP, VOLUME	cm
CLZO (NN) [NCLZO]	Initial cladding length.	EGIN,INPUT, LENGTH	cm
CPU (NN,NP) A[NCPU]	Plutonium concentration in the NP equal mass regions in the fuel.	BEGIN,EDTGRS, EDTINP,EDTMTX, INPUT,PLOT1,PLUTO, PROPTY	—
CPUD (NP) A[NCPUD]	Plutonium concentration change during the time interval.	PLUTO	1/sec
CRIF (NN) A[NCRIF]	Friction force between the fuel and the cladding.	BCALCP,EDTMTX, FRCHK	kg/cm
DELTOU (NN) A[NDELTA]	Outer temperature increment of the cladding (used for power cut).	POWCUT	°C
DEQP (NN,NTR) A[NDEQP]	Equivalent creep-strain increment of each region.	CHECK,CREEC, CREEF,MATRX, MATRXP	—
DXM (NN,3) A[NDXM]	Migration velocities of zone boundaries, i.e., inner	BEGIN,CHKSPD, FLTMP,RHOMOD, UPDTE	cm/sec

Symbol	Definition	Subroutine	Unit
	radius, columnar-equiaxed boundary, and equiaxed-undisturbed region boundary.		—
E (NN, NTR) A[NYNG]	Young's modulus in the fuel and cladding.	EDTMTX, MATRX, MATRXP, XLAME	kg/cm ²
EHP (NN, NTR) A[NEHP]	Hot-pressing of the fuel (i.e., compaction by decreasing the fuel porosity) contributing to swelling.	BCALC, BEGIN, EDTMTX, FLTMP, ITERAT.	—
EPC (NN, NTR) A[NEPC]	Total circumferential strain.	BCALC, BCALCP, EDTMTX, FLTMP, GASOUT, INPUT, ITERAT, RHOMOD, UPDATE, UPDATP	—
EPR (NN, NTR) A[NEPR]	Total radial strain.	BCALC, BCALCP, EDTMTX, FLTMP, GASOUT, INPUT, ITERAT, RHOMOD, UPDATE, UPDATP	—
EPSEQ(NN, NTR) A[NEPSEQ]	Equivalent creep-strain increment according to the Prandtl-Reuss flow laws.	BEGIN, CREEF, UPDATE	—
EPZ (NN, NTR) A[NEPZ]	Total axial strain.	BCALC, BCALCP, BURN, EDTMTX, FLTMP, GASOUT, INPUT, ITERAT, LENGTH, PLUTO, RHOMOD, UPDATE, UPDATP	—
EQSIG (NN, NTR) A[NEQSIG]	Equivalent stress.	CONVG, CREEC, CREEF, MATRX, MATRXP	kg/cm ²
F (NN, NP) A[NF]	Relative power density in radial shells of equal mass.	EDTINP, FLTMP, INPUT	—
FD (NN, NFR) A[NFD]	Fission density in the fuel.	CREEF, EDTGRS, FLTMP, GASOUT, SWELL	1/cm ²
FDPM (NN, NP) A[NFDPM]	Fission densities in the fuel regions into which the fuel is subdivided for plutonium migration.	EDTGRS, FLTMP	1/cm ²

Symbol	Definition	Subroutine	Unit
FLNC (NN) A[NFLNC]	Fast fluence.	BEGIN, CREEC, EDTGRS, FLXCAL, HKNS, UPDTE	n/cm
FLUX (NN) A[NFLUX]	Neutron flux.	BCALC, BCALCP, CREEC, CREEF, EDTGRS, FLXCAL	n/sec·cm ²
FLZ (NN) A[NFLZ]	Fuel length of each axial section.	BEGIN, EDTGRS, EDTINP, GASOUT, GOLDN, LENGTH, MATRX, PLENP, VOLUME	cm
FLZO (NN) A[NFLZO]	Initial fuel length.	BEGIN, LENGTH	cm
FTHP (NN, NTR) A[NFTHP]	Thermal creep component contributing to total creep of the fuel. (not used)	CREEF	—
G (NN, NTR) A[NSHR]	Shear modulus of each region of fuel and cladding.	BCALC, BCALCP, CRACK, CREEF, EDTMTX, MATRX, MATRXP, XLAME	kg/cm ²
GB (NN, 2) A[NSHRB]	Shear modulus averaged over the fuel or cladding.	BCALC, BCALCP, EDTMTX, MATRX, MATRXP, XLAME	kg/cm ²
GDT (NN, NP1) A[NGDT]	Temperature derivative at each region boundary. (dT/dr)	FLTMP, PLUTO	°C/cm
GR (NN) A[NGR]	Gas release rate from each axial section during the time step.	BEGIN, EDTGRS, GASOUT	mole
GS (NN, NFR) A[NGS]	Gas remaining in each region.	BEGIN, EDTGRS, EDTMTX, FLTMP, GASOUT, ITERAT	mole
HC (NN) A[NHC]	Heat transfer coefficient between the coolant and the fuel-element capsule (or the cladding if the element is unencapsulated).	EDTGRS, GOLDN	cal/hr· cm ² ·°F
HG (NN) A[NHG]	Gap heat-transfer coefficient between the fuel and the cladding.	BEGIN, EDTGRS, FLTMP, GAPCON	cal/hr· cm ² ·°F

Symbol	Definition	Subroutine	Unit
IO (NN) A[NIO]	Device reference number used for temporary data output. (not used)	IOSET	—
ITP (NBUF) A[NPIT]	Time step number selected for plotting.	PLOT5, PLTOR	—
KFRIC (NN) A[NKFRIC]	Gap status between the fuel and the cladding of each axial section. (0-gap open, 1-stick, 2-slip).	BCALC,BCALCP, BEGIN,CLEAR, EDTMX,FRCHK, MATRX,MATRXP	—
LM (NN,NTR) A[NLM]	Number of the cracks summed up in the three principal directions.	BEGIN,CRACK,CREEF, EDTMX,XLAME	—
LMC (NN,NTR) A[NLMC]	Number of cracks in the circumferential direction.	BEGIN,CRACK, EDTMX,PLOT4	—
LMR (NN,NTR) A[NLMR]	Number of cracks in the radial direction.	BEGIN,CRACK, EDTMX,PLOT4	—
LMZ (NN,NTR) A[NLMZ]	Number of cracks in the axial direction.	BEGIN,CRACK, EDTMX,PLOT4	—
OTOM (NN) A[NOTOM]	O/M ratio(i.e., X in UO _x) according to fuel stoichiometry.	EDTGRS,EDTINP, FLTMP,INPUT, OTMCAL	—
P (NN,NTR) A[NPOS]	Poisson's ratio of each region of the fuel and the cladding.	BCALC,BCALCP, EDTMX,MATRX, MATRXP,XLAME	—
PB (NN,2) A[NPOSB]	Poisson's ratio average over the fuel or the cladding.	BCALC,BCALCP, EDTMX,MATRX, MATRXP,XLAME	—
POW (NN) A[NPOW]	Linear power in each axial section	BEGIN,BURN,CLTMP, CREEF,EDTGRS, FLTMP,GASOUT, GOLDN,PLOT1,SWELL	watt/cm
PRS (NN,NTR) A[NPRS]	Hydro-static pressure.	BCALC,BEGIN,CRACK, EDTMX,ITERAT	kg/cm ²
PTOUR (NN) A[NPTOU]	Cladding outer surface temperature of the previous time step.	CORTMP,POWCUT	°C

Symbol	Definition	Subroutine	Unit
QSF (NN) A[NQSF]	Axial power distribution factor.	BEGIN,BURN,GOLDN	—
QSFF (NN) A[NQSFF]	Axial fast flux shape.	BEGIN,EDTINP, FLXCAL,INPUT	—
QSFP (NN) A[NQSFP]	Nonnormalized axial power shape.	BEGIN,BURN,EDTINP, INPUT	—
RB (NN,NTR) A[NRBAR]	Relative radius representing the each radial region.	BCALC,CLTMP,FLTMP, MATRX	cm
RCB (NN) A[MRCB]	Radius of columnar and equiaxed region interface.	BEGIN,EDTGRS, EDTIMP,FLTMP, INPUT,PLOT1,PLOT4, PLUTO,RHOMOD, UPDTE	cm
RCBO (NN) A[NRCBO]	Initial radius of columnar and equiaxed region interface.	BEGIN,RHOMOD	cm
RCI (NN) A[NRCI]	Inner radius of the cladding.	BEGIN,CLTMP, EDTGRS,EDTINP, FLTMP,FRCHK, GAPCON,INPUT, PLENP,PLOT1, RHOMOD,UPDTE	cm
RCIO (NN) A[NRCIO]	Initial inner radius of the cladding.	BEGIN	cm
RCL (NN,NCL1) A[NRCL]	Radii of the region boundaries in the cladding.	BCALC,BCALCP, CLTMP,EDTGRS, EDTMIX,MATRX, MATRXP,VOLUME	cm
RCLO (NN,NCL1) A[NRCLO]	Initial radii of the region boundaries in the cladding.	CLTMP	cm
RCO (NN) A[NRCO]	Outer radius of the cladding.	BEGIN,EDTGRS, EDTINP,GOLDN, INPUT,PLOT3, RHOMOD,UPDTE	cm
RCOO (NN) A[NRCOO]	Initial outer radius of the cladding	BEGIN,EDTMIX, PLOT3	cm
RCP (NN) A[NRCP]	Radii of the region boundaries in the capsule.	CLTMP,EDTMIX	cm
RDIS (NN,NTR2) A[NRDIS]	Radial displacements of the region boundaries in the fuel and the cladding.	EDTMIX,INPUT, PLOT3,RHOMOD, UPDATE,UPDATP	cm

Symbol	Definition	Subroutine	Unit
RDIST (NN,4) A[NRDIS]	Used for checking. (not used for calculations) RDIST(NN,1)-not used. RDIST(NN,2)-restructuring accumulated into columnar form. RDIST(NN,3)-restructuring accumulated into equiaxed form. RDIST(NN,4)-mass	BEGIN,RHOMOD,UPDTE	—
REB (NN) A[NREB]	Radius of equiaxed and undisturbed region interface.	BEGIN,EDTGRS, EDTINP,FLTMP,INPUT, PLOT1,PLOT4,PLUTO, RHOMOD,UPDTE	cm
REBO (NN) A[NREBO]	Initial radius of equiaxed and undisturbed region interface.	BEGIN,RHOMOD	cm
RFR (NN,NFR1) A[NRFR]	Radii of the region boundaries in the fuel.	BCALC,EDTGRS, EDTMTX,FLTMP, GASOUT,ITERAT, MATRX,PLOT1, RHOMOD,VOLUME	cm
RFRO (NN,NFR1) A[NRFR0]	Initial radii of the region boundaries in the fuel.	FLTMP	cm
RFRPM(NN,NP1) A[NRFRPM]	Radii of the region boundaries of the NP equal mass radial shells.	EDTMTX,FLTMP,PLOT1, PLOT4,PLUTO	cm
RH (NN,NTR) A[NRH]	Density of the fuel and the cladding.	BCALC,BCALCP,BEGIN, EDTGRS,FLTMP, ITERAT,PLUTO, RHOMOD,XLAME	g/cc
RHOP (NN,3) A[NRHOP]	Initial density of the three region in the fuel.	BEGIN,EDTINP, FLTMP,INPUT,ITERAT, PROPTY,RHOMOD	g/cc
RHOT (NN) A[NRHOT]	Theoretical density of the fuel.	EDTGRS,FLTMP, ITERAT,PROPTY, RHOMOD,XLAME	g/cc
RUB (NN) A[NRUB]	Outer radius of the fuel.	BEGIN,EDTGRS,EDTINP, FLTMP,FRICHK,GAPCON, INPUT,PLENP,PLOT1, RHOMOD,UPDTE	cm
RUBO (NN) A[NRUBO]	Initial outer radius of the fuel.	BEGIN,RHOMOD	cm

Symbol	Definition	Subroutine	Unit
RVB (NN) A[NRVB]	Central void radius.	BEGIN, EDTGRS, EDTINP, FLTMP, INPUT, PLENP, PLUTO, RHOMOD, UPDTE,	cm
RVBO (NN) A[NRVBO]	Initial central void radius.	BEGIN, RHOMOD	cm
SIGC(NN,NTR) A[NSIGC]	Hoop stress component of the fuel and the cladding.	CONVG, EDTMTX, INPUT, PLOT2, UPDATE, ITERAT, UPDATP	kg/cm ²
SIGR (NN,NTR) A[NSIGR]	Radial stress component.	CONVG, EDTMTX, INPUT, ITERAT, UPDATE, UPDATP	kg/cm ²
SIGZ (NN,NTR) A[NSIGZ]	Axial stress component.	CONVG, EDTMTX, INPUT, ITERAT, UPDATE, UPDATP	kg/cm ²
SWG (NN,NFR) A[NSWG]	Swelling rates due to gaseous fission products.	BCALC, BEGIN, EDTMTX, FLTMP, ITERAT	—
SWS (NN,NTR) A[NSWS]	Swelling rates due to solid fission products.	BCALC, BEGIN, EDTMTX, FLTMP, GASOUT, ITERAT, SWELL	—
SWS3 (NN,NUR) A[NSWS3]	Swelling rates due to the solid fission products in the NUR radial meshes of the undisturbed region.	BEGIN, ITERAT, SWELL	—
TBCL (NN,NCL) A[NTBCL]	Average temperature of each region in the cladding.	BCALC, BCALCP, CLTMP, CREEC, EDTMTX, HKNS, XLAME	°C
TBCLA (NN) A[NTBCLA]	Average temperature of the cladding.	CLTMP, EDTGRS	°C
TBCP (NN,NCL) A[NTBCP]	Average temperature of each region in the capsule.	CLTMP, EDTMTX	°C
TBCPA (NN) A[NTBCPA]	Average temperature of the capsule.	CLTMP, EDTGRS	°C
TBF (NN,NFR) A[NTBF]	Average temperature of each region in the fuel.	CRACK, CREEF, EDTMTX, FLTMP, GASOUT, ITERAT, XLAME	°C
TBFA (NN,3) A[NTBFA]	Average temperature of each region in the fuel.	EDTGRS, FLTMP	°C
TBPM (NN,NP) A[NTBFPM]	Average temperature of each equal mass shell in the fuel.	FLTMP	°C

Symbol	Definition	Subroutine	Unit
TCL (NN,NCL1) A[NTCL]	Temperature at the boundaries of the NCL radial meshes of the cladding.	CLTMP,EDTGRS, EDTMTX,FLTMP	°C
TCLO (NN) A[NTCLO]	Outer temperature of the cladding.	CLTMP,EDTGRS,GOLDN	°C
TCP (NN,NCL1) A[NTCP]	Temperature at the boundaries of the NCL radial meshes of the capsule.	CLTMP,EDTGRS, EDTMTX	°C
TCPI (NN) A[NTCPI]	Innter temperature of the capsule.	EDTGRS,GOLDN	°C
TCPO (NN) A[NTCPO]	Outer temperature of the capsule.	CLTMP,EDTGRS,GOLDN	°C
TFR (NN,NFR1) A[NTFR]	Temperatures at the boundaries of the radial regions of the fuel.	EDTGRS,EDTMTX, FLTMP,PLENP,PLOT1,	°C
TFRPM(NN,NP1) A[NTFRPM]	Temperatures at the boundaries of the radial NP equal mass shells in the fuel.	EDTMTX,FLTMP, PLOT1,PLUTO	°C
TNA (NN) A[NTNA]	Coolant temperature.	EDTGRS,GOLDN	°C
TOUTER (NN) A[NTOUR]	Outer surface temperature of the cladding.	CORTMP,GOLDN, POWCUT	°C
TOUTO (NN) A[NTOUO]	Outer surface temperature of the cladding of the previous time step.	CORTMP	°C
TPC (NN,NTR) A[NTPC]	Cumulative total circumferential creep strain.	BCALC,BCALCP,CRACK, EDTMTX,FLTMP,INPUT, PLOT3,UPDATE,UPDATP	—
TPR (NN,NTR) A[NTPR]	Cumulative total radial creep strain.	BCALC,BCALCP,CRACK, EDTMTX,FLTMP,INPUT, UPDATE,UPDATP	—
TPZ (NN,NTR) A[NTPZ]	Cumulative total axial creep strain.	BCALC,BCALCP,CRACK, EDTMTX,FLTMP,INPUT, UPDATE,UPDATP	—

Symbol	Definition	Subroutine	Unit
TSC (NN,NTR) A[NTSC]	Cumulative total circumferential swelling strain.	BCALC,BCALCP,CRACK,EDTMTX,FLTMP,HKNS,INPUT,ITERAT,PLOT3,UPDATE,UPDATP	—
TSR (NN,NTR) A[NTSR]	Cumulative total radial swelling strain.	BCALC,BCALCP,CRACK,EDTMTX,FLTMP,HKNS,INPUT,ITERAT,UPDATE,UPDATP	—
TSZ (NN,NTR) A[NTSZ]	Cumulative total axial swelling strain.	BCALC,BCALCP,CRACK,EDTMTX,FLTMP,HKNS,INPUT,ITERAT,UPDATE,UPDATP	—
VOL (NN,NTR) A[NVOL]	Volume of each region.	BCALC,BCALCP,GASOUT,ITERAT,PLOT2,PLOT3,VOLUME,XLAME	cm ³
WNEW (NFR) A[NWNEW]	Mass in the each radial region of the fuel.	FLTMP,RHOMOD	g/cm
WOLD (NN,NFR) A[NWOLD]	Mass in the each radial region of the fuel at the previous time step.	FLTMP,RHOMOD,UPDTE	g/cm
XBU (NBUF) A[NPBU]	Same as BU (used for plotting)	PLOT1, PLTOR	a/o
XCPGP (NBUF) A[NPCPGP]	Clad-pellet gap (used for plotting)	PLOT1, PLTOR	mm
XCPI (NN) A[NXCPI]	Capsule inner radius.	BEGIN,CLTMP,EDTGRS,EDTINP	cm
XCPO (NN) A[NXCPO]	Capsule outer radius.	BEGIN,CLTMP,EDTGRS,EDTINP	cm
XCPU (NBUF,NP) A[NPCPU]	Same as CPU(used for plotting)	PLOT1, PLTOR	—
XCSTN (NBUF) A[NPCSTN]	Diameter change ($\Delta D/D$) of the cladding. (used for plotting)	PLOT3, PLTOR	%
XEHP (NFR) A[NXEHP]	Used temporarily for modifying EHP considering the restructuring.	FLTMP,ITERAT	—

Symbol	Definition	Subroutine	Unit
XEPC (NFR) A[NXEPC]	Used temporarily for modifying EPC considering the restructuring.	FLTMP,RHOMOD	—
XEPR (NFR) A[NXEPR]	Used for modifying EPR.	FLTMP,RHOMOD	—
XGS (NFR) A[NXGS]	Used for modifying GS	FLTMP	mole
XM (NN,3) A[NXM]	Mass of each region in the fuel.	BEGIN,BURN,EDTGRS,FLTMP,PLUTQ,RHOMOD,	g/cm
XMINT (NN) A[NXMINT]	Initial mass of each axial section of the fuel.	BEGIN,RHOMOD	g/cm
XPLPR (NBUF) A[NPPLPR]	Plenum pressure (used for plotting)	PLOT5,PLTOR	kg/cm ²
XPOW (NBUF) A[NPPOW]	Same as POW(used for plotting)	PLOT1,PLTOR	watt/cm
XRL (NBUF,NP1) A[NPRL]	Same as RFRPM (used for plotting)	PLOT1,PLTOR	mm
XRLR (NBUF,2) A[NPRLR]	XRLR(NBUF,1)-same as RCB XRLR(NBUF,2)-same as REB (used for plotting)	PLOT1,PLTOR	mm
XSIGC (NBUF) A[NPSIGC]	Cladding average hoop stress. (used for plotting)	PLOT2,PLTOR	kg/mm ²
XSWG (NFR) A[NXSWG]	Used for modifying SWG considering the restructuring	FLTMP,ITERAT	—
XSWS (NFR) A[NXSWS]	Used for modifying SWS	FLTMP	—
SXWS3 (NUR) A[NXSWS3]	Used for modifying SWS3	FLTMP	—
XT (NBUF) A[NPT]	Fuel outer surface temperature.(used for plotting)	PLOT1,PLTOR	°C
XTIME (NBUF) A[NPTIME]	Time selected for plotting. (used for plotting)	PLOT5,PLTOR	hr

Symbol	Definition	Subroutine	Unit
XTL (NBUF,NP1) A[NPTL]	Same as TFRPM (used for plotting)	PLOT1,PLTOR	°C
XTLMX (NBUF) A[NPTLMX]	Maximum temperature in the fuel. (used for plotting)	PLOT1,PLTOR	°C
XTLR (NBUF,2) A[NPTLR]	Temperature at the radial point of radius XRLR. (used for plotting)	PLOT1,PLTOR	°C
XTPC (NFR) A[NXTPC]	Used for modifying TPC con- sidering the restructuring	FLTMP	—
XTPC (NBUF) A[NPTPC]	Cladding average circum- ferential creep strain. (used for plotting)	PLOT3,PLTOR	—
XTPR (NFR) A[NXTPR]	Used for modifying TPR.	FLTMP	—
XTPZ (NFR) A[NXTPZ]	Used for modifying TPZ.	FLTMP	—
XTSC (NFR) A[NXTSC]	Used for modifying TSC.	FLTMP	—
XTSC (NBUF) A[NPTSC]	Cladding average circum- ferential swelling strain. (used for plotting)	PLOT3,PLTOR	—
XTSR (NFR) A[NXTSR]	Used for modifying TSR.	FLTMP	—
XTSZ (NFR) A[NXTSZ]	Used for modifying TSZ.	FLTMP	—
XYP (NBUF) A[NPYP]	Gas release rate. (used for plotting)	PLOT5,PLTOR	%
YBU (NN) A[LBU]	BU } at the start of the	UPDTE	a/o
YDXM (NN,3) A[LDXM]	DXM } time step.	CHKSPD,UPDTE	cm/sec

Symbol	Definition	Subroutine	Unit
YFLNC (NN) A[LFLNC]	FLNC	UPDTE	n/cm ²
YRCB (NN) A[LRCB]	RCB	UPDTE	cm
YRCI (NN) A[LRCI]	RCI	UPDTE	cm
YRCO (NN) A[LRCO]	RCO	UPDTE	cm
YRDIST(NN,4) A[LRDIST]	RDIST	UPDTE	—
YREB (NN) A[LREB]	REB	UPDTE	cm
YRUB (NN) A[LRUB]	RUB	UPDTE	cm
YRVB (NN) A[LRVB]	RVB	UPDTE	cm
YTOUR (NN) A[LTOUR]	TOUTER	CORTMP	°C
YWOLD(NN,NFR) A[LWOLD]	WOLD	UPDTE	g/cm
ZFLX (NN) A[NZFLX]	ratio of Frenkel pair- generation rate to flux.	BEGIN,FLXCAL	—

APPENDIX D: CONTROL CARDS AND MESSAGES

Appendix D: Control Cards and Messages

D.1 Control Cards

A typical set of control cards used for the ACTIVE-II code on the IBM 360/195 at Tokyo Data Center is shown in Table D.1.

Included is a short FORTRAN program containing the variable 'A' and the variable 'AM'--these dimensions must be set at least as large as the core space required for the problem data (excluding the data for the matrix) and for the data only for the matrix, respectively. (See the section 6.1 Core Storage.) And the variable 'IA' and the variable 'IAM' must also be the same number as the dimensions of the 'A' and the 'AM', respectively.

Logical unit requirements are as follows.

Logical Number	Device Type	Use
5		System input device.
6		System output device.
7		System punch device.
77	Tape	Plotter output device

Logical unit requirements are described in the section I/O Unit Requirements.

On the stage of link-edit, four library data sets are required. SYS1.DCSLIB and SYS1.FORTLIB includes the functional subroutines such as SQRT, ABS, EXP, ALOG, DATE, STIMER, TTIMER, and so on, used in the ACTIVE-II.

SYS1.PLTLIB10 includes the subroutines used for plotting such as PLOTS, PLOT, SYMBOL, NUMBER, and so on.

FPPL.SLMATH includes the SL-MATH subroutines such as MSNR, MBSR, and so on.

```

//J1111AMK JOB (1111A,ACTIV),'KATSURAGAWA',TIME=1,REGION=384K
/*MAIN TYPE=(PLOT)
/*SETUP UNIT=PLOT,ID=(PLOT,RING,SAVE,NL)
/*JOBPARM LIN=20
//ACTIVE EXEC PLTFGCLG
//FORT.SYSIN DD *
*****
*****
***** SOURCE PROGRAM DECK *****
*****
*****
/*
//GO.SYSIN DD *
*****
*****
***** DATA CARD *****
*****
*****
/*

```

D.2 Error Stops

The following error stops have been programmed into ACTIVE-II.

Cause	Subroutine
1. Clad has hit capsule wall	GOLDN
2. '*END' card missing (input miss)	HEADER
3. Mass increment is too big	PLUTO
4. Mass in columnar zone is negative	RHOMOD
5. Clad temperature is too high	XLAME
6. Refer to the MSNR	MSNR(SL-MATH)

APPENDIX E: SAMPLE INPUT

Appendix E: Sample Input

Sample input is shown in Table E.1. The sample pin is F1A fuel pin, whose abstract is shown in Chapter 6.2. For the detail explanation, refer to Chapter 6.3.

Table E.1 Sample Input.

#-----1-----2-----3-----4-----5-----6-----7-----8												PAGE	1	
1	0	1	0									REC. NO.	1	
ACTIVE SENSITIVITY ANALYSIS												REC NO.	2	
*** G.E.F-1A (E8R-2) ***												REC NO.	3	
AXIAL MESH = 4												REC NO.	4	
RADIAL MESH												REC NO.	5	
COLUMNAR = 1												REC NO.	6	
EQUILAXED = 1												REC NO.	7	
UNDISTURBED = 1												REC NO.	8	
CLADDING = 1												REC NO.	9	
*END												REC NO.	10	
4	1	1	1	1	10	3	-2	-1	1	1	0	0	REC NO.	11
1	2	2	2	2									REC NO.	12
0.001		0.001		0.001		0.001		0.001					REC NO.	13
0.002		0.002		0.002		0.002		0.002					REC NO.	14
0.003		0.003		0.003		0.003		0.003					REC NO.	15
0.2744		0.2744		0.2744		0.2744		0.2744					REC NO.	16
0.2794		0.2794		0.2794		0.2794		0.2794					REC NO.	17
0.3175		0.3175		0.3175		0.3175		0.3175					REC NO.	18
14.45		3.61		18.06		30.06							REC NO.	19
0.9792		1.0723		0.9485		0.3							REC NO.	20
0.9792		1.0723		0.9485		0.0							REC NO.	21
1.998		1.998		1.998		1.998							REC NO.	22
0.2		0.2		0.2		0.2							REC NO.	23
10.54		10.54		10.54		10.54							REC NO.	24
1.0		1.0		1.0		1.0				1.0			REC NO.	25
1.0		1.0		1.0		1.0				1.0			REC NO.	26
0.4254		0.4762		0.822									REC NO.	27
3000.0		6000.0		1000000.0									REC NO.	28
0.0		1.68											REC NO.	29
7.98		20.0		20.0									REC NO.	30
	1.92E+07		1.2E+04	64400.0		44400.0							REC NO.	31
	0.407E-030.0		0.0										REC NO.	32
0.15		0.01				3.48E+12371.1		372.0					REC NO.	33
0.671		439.8				3.48E+12371.1		479.4		1			REC NO.	34
2.00		439.8				3.48E+12371.1		479.4					REC NO.	35
2.125		0.01				3.48E+12371.1		372.1					REC NO.	36
2.5		0.01				3.48E+12371.1		372.1					REC NO.	37
2.921		439.8				3.48E+12371.1		479.4					REC NO.	38
6.35		439.8				3.48E+12371.1		479.4					REC NO.	39
6.643		195.5				3.48E+12371.1		419.2					REC NO.	40
6.8		195.5				3.48E+12371.1		419.2					REC NO.	41
7.142		439.8				3.48E+12371.1		479.4					REC NO.	42
9.50		439.8				3.48E+12371.1		479.4					REC NO.	43
10.0		0.01				3.48E+12371.1		372.1					REC NO.	44
11.0	1.0	0.01				3.48E+1250.0		51.0					REC NO.	45
12.0	1.0	0.01				3.48E+1250.0		51.0		1			REC NO.	46
#-----1-----2-----3-----4-----5-----6-----7-----8														
TOTAL RECORDS												46		

APPENDIX F: SAMPLE OUTPUT

Appendix F: Sample Output

Sample output is shown here, calculated for F1A fuel pin as shown in Appendix E. For detail explanation of output, refer to Chapter 6.4, and 6.5.

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

===== ACTIVE - II - FRR =====

ACTIVE SENSITIVITY ANALYSIS			
*** G.E.F-1A (EBR-2) ***			
AXIAL MESH	=	4	
RADIAL MESH			
COLUMNAR	=	1	
EQUIAXED	=	1	
UNDISTURBED	=	1	
CLADDING	=	1	

===== INPUT VALUES OUTPUT NEXT =====

----- CONTROL NUMBER AND OPTION

NUMBER OF AXIAL SECTION NN = 4
 NUMBER OF MESH FOR COLUMNAR GRAIN REGION NCR = 1
 NUMBER OF MESH FOR EQUIAXED GRAIN REGION NER = 1
 NUMBER OF MESH FOR UNDISTURBED REGION NUR = 1
 NUMBER OF MESH FOR CLAD REGION NCL = 1
 NUMBER OF MESH FOR PLUTONIUM MIGRATION NP = 10
 FUEL THERMAL CONDUCTIVITY OPTION NOK = 3 (1/2/3 = SAYLES/BAILY-ASAMOTO/BIANCHERIA)
 CLAD SWELLING OPTION NOS = -2 (-1/-2 = ANNEALED 304,316S.S./20P.C.COLD WORK 316S.S.)
 CLAD CREEP OPTION NOC = -1 (-1/-2 = BOLTAX,BIANCHERIA,GUHA/MODIFIED)
 CRACK OPTION NCRK = 1 (0/1 = NO CRACK CALC./CRACK CALC.)
 PLOT OPTION NPLOT = 1 (0/1 = NO PLOT/PLOT)
 AXIAL SECTION NUMBER FOR FUEL = 2
 AXIAL SECTION NUMBER FOR STRAIN = 2
 AXIAL SECTION NUMBER FOR STRESS = 2
 AXIAL SECTION NUMBER FOR CRACK = 2
 STORAGE(WORDS) REQUIRED FOR PLOTTING STEP = 50

----- CENTRAL VOID RADIUS (CM) 1-NN

0.0010 0.0010 0.0010 0.0010

----- COLUMNAR GRAIN REGION OUTER BOUNDARY (CM) 1-NN

0.0020 0.0020 0.0020 0.0020

----- EQUIAXED GRAIN REGION OUTER BOUNDARY (CM) 1-NN

0.0030 0.0030 0.0030 0.0030

----- FUEL OUTER RADIUS (CM) 1-NN

0.2744 0.2744 0.2744 0.2744

----- CLAD INNER RADIUS (CM) 1-NN

0.2794 0.2794 0.2794 0.2794

----- CLAD OUTER RADIUS (CM) 1-NN

0.3175 0.3175 0.3175 0.3175

----- CAPSULE INNER RADIUS (CM) 1-NN

0.4254 0.4254 0.4254 0.4254

----- CAPSULE OUTER RADIUS (CM) 1-NN

0.4762 0.4762 0.4762 0.4762

----- COOLANT FLOW AREA = 0.82199997 (CM**2)

----- FUEL AXIAL LENGTH (CM) 1-NN ... TOTAL AXIAL LENGTH = 36.1200 (CM) ...

14.4500	3.6100	18.0600	0.0
---- CLAD AXIAL LENGTH (CM) 1-NN ... TOTAL AXIAL LENGTH = 66.1800 (CM) ...			
14.4500	3.6100	18.0600	30.0600
---- DENSITY OF FUEL (3 REGIONS) AND CLAD (G/CC)			
AXIAL SECTION = 1	10.6612	10.5400	7.9800
AXIAL SECTION = 2	10.6612	10.5400	7.9800
AXIAL SECTION = 3	10.6612	10.5400	7.9800
AXIAL SECTION = 4	0.0	0.0	7.9800
---- D/M OF FUEL PU-D(X) 1-NN			
1.9980	1.9980	1.9980	1.9980
---- PLUTONIUM CONCENTRATION (WEIGHT-FRACTION) 1-NP			
AXIAL SECTION = 1	0.2000	0.2000	0.2000
AXIAL SECTION = 2	0.2000	0.2000	0.2000
AXIAL SECTION = 3	0.2000	0.2000	0.2000
AXIAL SECTION = 4	0.2000	0.2000	0.2000
---- AXIAL FAST FLUX (E.GE. 0.1MEV) RATE 1-NN			
0.9792	1.0723	0.9485	0.3000
---- AXIAL HEAT GENERATION RATE 1-NN			
0.9792	1.0723	0.9485	0.0
---- RADIAL HEAT GENERATION RATE 1-NP			
AXIAL SECTION = 1	1.0000	1.0000	1.0000
AXIAL SECTION = 2	1.0000	1.0000	1.0000
AXIAL SECTION = 3	1.0000	1.0000	1.0000
AXIAL SECTION = 4	1.0000	1.0000	1.0000
---- INITIAL GAS IN PLENUM = 0.0004070 (MOLES)			
*** ACTUALLY USED CORE FOR ARRAY = 3897 WORDS (INPUT = 10000)			
*** ACTUALLY USED CORE FOR MATRIX = 5984 WORDS (INPUT = 20000)			
*** RECOMMENDED CORE SIZE = 256 K BYTE			

STEP NO. 1	TIME (HRS) =	3.600	POWAV (W/CM) =	0.010	TOUT (C) =	372.100	TIN (C) =	371.100	
	DELT (HRS) =	3.600	CFLX (N/W/CM) =	0.348E 13	PD (KG/CM**2) =	1.68			

===== GROSS OUTPUT NEXT =====									
+++ AXIAL DISTRIBUTION (N=1,NM) +++									
POWER(W/CM)	0.010	0.011	0.009	0.0					
FLUX(N/CM**2/SEC)	0.34076E 11	0.37316E 11	0.33008E 11	0.10440E 11					
FLUENCE(N/CM**2)	0.44163E 15	0.48362E 15	0.42778E 15	0.13530E 15					
BURN UP(A/O)	0.00000	0.00000	0.00000	0.0					
BURN UP(MWD/T)	0.0	0.0	0.0	0.0					
FUEL LENGTH(CM)	14.45000	3.61000	18.06000	0.0					
CLAD LENGTH(CM)	14.45000	3.61000	18.06000	30.06000					
THEO. DENS.(G/CC)	11.06000	11.06000	11.06000	0.0					
COOLANT TEMP.(C)	371.301	371.557	371.856	372.100					
FILM H.(W/CM**2/C)	3.33288	3.33231	3.33163	3.33109					
GAP H.(W/CM**2/C)	0.81487	0.81487	0.81487	0.81487					
DELT GAS REL.(MOLES)	0.0000000	0.0000000	0.0000000	0.0					
TOTAL GAS IN PLENUM	0.00040700 (MOLES)								
CUMULATIVE GAS RELEASED	0.0 (MOLES)								
PERCENT OF TOTAL F.P. GAS	0.0 (O/O)								
PLENUM PRESSURE	2.896 (KG/CM**2)								
CENTRAL HOLE PRESSURE	2.894 (KG/CM**2)								
TOTAL FUEL LENGTH AND FL/FLO	36.12000 (CM)								
TOTAL CLAD LENGTH AND CL/CLD	66.17999 (CM)								
+++ PERFORMANCE +++									
* AXIAL SECTION = 1 *									
	FUEL CENTER	COLUMNAR GR.	EQUITAEX GR.	FUEL OUTER	CLAD INNER	CLAD OUTER	CAPSULE INNER	CAPSULE OUTER	
RADIUS(CM)	0.00101	0.00200	0.00300	0.27440	0.27940	0.31750	0.42540	0.47620	
TEMP.(C)	371.325	371.325	371.325	371.311	371.304	371.303	371.302	371.302	
AV.TEMP(C)	371.325	371.325	371.318		371.303		371.302		
ALPHA-T(-)	0.00303	0.00303	0.00303		0.00626		0.00626		
DENS(G/CC)	10.66115	10.54000	10.54000		7.98000				
MASS(G/CM)	0.00010	0.00017	2.49291						
* AXIAL SECTION = 2 *									
	FUEL CENTER	COLUMNAR GR.	EQUITAEX GR.	FUEL OUTER	CLAD INNER	CLAD OUTER	CAPSULE INNER	CAPSULE OUTER	
RADIUS(CM)	0.00101	0.00200	0.00300	0.27440	0.27940	0.31750	0.42540	0.47620	
TEMP.(C)	371.585	371.585	371.585	371.568	371.561	371.560	371.559	371.558	
AV.TEMP(C)	371.585	371.585	371.576		371.560		371.558		
ALPHA-T(-)	0.00303	0.00303	0.00303		0.00627		0.00627		
DENS(G/CC)	10.66115	10.54000	10.54000		7.98000				
MASS(G/CM)	0.00010	0.00017	2.49291						
* AXIAL SECTION = 3 *									
	FUEL CENTER	COLUMNAR GR.	EQUITAEX GR.	FUEL OUTER	CLAD INNER	CLAD OUTER	CAPSULE INNER	CAPSULE OUTER	
RADIUS(CM)	0.00101	0.00200	0.00300	0.27440	0.27940	0.31750	0.42540	0.47620	
TEMP.(C)	371.880	371.880	371.880	371.866	371.859	371.858	371.858	371.857	
AV.TEMP(C)	371.880	371.880	371.873		371.858		371.857		
ALPHA-T(-)	0.00303	0.00303	0.00303		0.00627		0.00627		
DENS(G/CC)	10.66115	10.54000	10.54000		7.98000				
MASS(G/CM)	0.00010	0.00017	2.49291						
* AXIAL SECTION = 4 *									

	FUEL CENTER	COLUMNAR GR.	EQUIAXED GR.	FUEL OUTER	CLAD INNER	CLAD OUTER	CAPSULE INNER	CAPSULE OUTER
RADIUS(CM)	0.0	0.0	0.0	0.0	0.27940	0.31750	0.42540	0.47620
TEMP.(C)	0.0	0.0	0.0	0.0	372.100	372.100	372.100	372.100
AV. TEMP(C)	0.0	0.0	0.0	0.0	372.100	372.100	372.100	372.100
ALPHA-T(-)	0.0	0.0	0.0	0.0	0.00628	0.00628	0.00628	0.00628
DENS(G/CC)	0.0	0.0	0.0	0.0	7.98000	7.98000	7.98000	7.98000

===F= STRESS AND STRAIN OUTPUT NEXT =====

*** AXIAL SECTION = 1 ***

+++ CAPSULE (I=1,NCL) +++								
RADIUS(CM)	0.42540	0.47620						
TEMP.(C)	371.302	371.302						
ALPHA-T(-)	0.00626							
+++ CLAD (I=1,NCL) +++								
RADIUS(CM)	0.27940	0.31750						
DISPLACEMENT(CM)	0.00175	0.00199						
TEMP.(C)	371.304	371.303						
PRESS.(KG/CM**2)	0.28960E 01	0.16800E 01						
ALPHA-T(-)	0.00626							
POISSON RATIO(-)	0.33306							
YOUNG M.(KG/CM**2)	0.17504E 07							
STRESS(KG/CM**2)								
RADIAL	-0.22363E 01							
CIRCUMFERENTIAL	0.72244E 01							
AXIAL	0.20226E 01							
TOTAL STRAIN(-)								
RADIAL	0.62612E-02							
CIRCUMFERENTIAL	0.62684E-02							
AXIAL	0.62645E-02							
CREEP STRAIN(-)								
RADIAL	-0.59555E-15							
CIRCUMFERENTIAL	0.63651E-15							
AXIAL	-0.40928E-16							
CREEP IN DELT(-)								
RADIAL	-0.59555E-15							
CIRCUMFERENTIAL	0.63651E-15							
AXIAL	-0.40928E-16							
SWELLING STRAIN(-)								
TOTAL	0.55260E-15							

PERCENT CHANGE IN ORIGINAL O.D. = 0.0 (0/0)

AVERAGE POISSON RATIO(-) = 0.33306

AVERAGE YOUNG MODULUS(KG/CM**2) = 0.17504E 07

+++ FUEL (I=1,NFR) +++								
RADIUS(CM)	0.00101	0.00200	0.00300	0.27440				
DISPLACEMENT(CM)	0.00000	0.00001	0.00001	0.00089				
TEMP.(C)	371.325	371.325	371.325	371.311				
PRESS.(KG/CM**2)	0.28939E 01	0.29983E 01	0.29983E 01	0.28940E 01				
ALPHA-T(-)	0.00303	0.00303	0.00303	0.00303				
POISSON RATIO(-)	0.31705	0.31549	0.31549	0.31549				
YOUNG M.(KG/CM**2)	0.22370E 07	0.21741E 07	0.21741E 07	0.21741E 07				
NUMBER OF CRACK								
RADIAL	0	0	0	0				
CIRCUMFERENTIAL	0	0	0	0				

AXIAL	0	0	0									
STRESS(KG/CM**2)												
RADIAL	-0.29686E 01	-0.29986E 01	-0.28961E 01									
CIRCUMFERENTIAL	-0.30990E 01	-0.29998E 01	-0.28959E 01									
AXIAL	-0.34913E 01	-0.34155E 01	-0.32076E 01									
TOTAL STRAIN(-)												
RADIAL	0.30275E-02	0.30274E-02	0.30274E-02									
CIRCUMFERENTIAL	0.30274E-02	0.30274E-02	0.30274E-02									
AXIAL	0.30272E-02	0.30272E-02	0.30272E-02									
CREEP STRAIN(-)												
RADIAL	0.32659E-31	0.20904E-31	0.15566E-31									
CIRCUMFERENTIAL	0.13095E-31	0.20731E-31	0.15600E-31									
AXIAL	-0.45755E-31	-0.41635E-31	-0.31167E-31									
CREEP IN DELT(-)												
RADIAL	0.32659E-31	0.20904E-31	0.15566E-31									
CIRCUMFERENTIAL	0.13095E-31	0.20731E-31	0.15600E-31									
AXIAL	-0.45755E-31	-0.41635E-31	-0.31167E-31									
SWELLING STRAIN(-)												
TOTAL	0.65783E-09	0.36103E-09	0.15577E-09									
SOLID COMPONENT	0.19297E-09	0.19294E-09	0.46196E-09									
GAS COMPONENT	0.17805E-08	0.89015E-09	0.53401E-11									
HOT PRESSING	0.0	0.0	0.0									
PERCENT CHANGE IN ORIGINAL O.D. =	0.0	(0/0)										
AVERAGE POISSON RATIO(-)	=	0.31549										
AVERAGE YOUNG MODULUS(KG/CM**2) =	0.21741E 07											
GAP SIZE(CM)	=	0.00592										
FRICTION FORCE(KG/CM)	=	0.0	(OPEN)									
*** PLUTONIUM MIGRATION (I=1,NP) ***												
RADIUS(CM)	0.00101	0.08678	0.12272	0.15030	0.17355	0.19403	0.21255	0.22958	0.24543	0.26032	0.27440	
TEMP.(C)	371.325	371.324	371.323	371.321	371.320	371.318	371.317	371.315	371.314	371.312	371.311	
PU CONC.(-)	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	
*** AXIAL SECTION = 2 ***												
*** CAPSULE (I=1,NCL) ***												
RADIUS(CM)	0.42540	0.47620										
TEMP.(C)	371.559	371.558										
ALPHA-T(-)	0.00627											
*** CLAD (I=1,NCL) ***												
RADIUS(CM)	0.27940	0.31750										
DISPLACEMENT(CM)	0.00175	0.00199										
TEMP.(C)	371.561	371.560										
PRESS.(KG/CM**2)	0.28960E 01	0.16800E 01										
ALPHA-T(-)	0.00627											
POISSON RATIO(-)	0.33308											
YOUNG M.(KG/CM**2)	0.17501E 07											
STRESS(KG/CM**2)												
RADIAL	-0.22363E 01											
CIRCUMFERENTIAL	0.72244E 01											
AXIAL	0.20984E 01											
TOTAL STRAIN(-)												
RADIAL	0.62660E-02											

CIRCUMFERENTIAL	0.62732E-02			
AXIAL	0.62693E-02			
CREEP STRAIN(-)				
RADIAL	-0.65581E-15			
CIRCUMFERENTIAL	0.69342E-15			
AXIAL	-0.37613E-16			
CREEP IN DELT(-)				
RADIAL	-0.65581E-15			
CIRCUMFERENTIAL	0.69342E-15			
AXIAL	-0.37613E-16			
SWELLING STRAIN(-)				
TOTAL	0.63787E-15			
PERCENT CHANGE IN ORIGINAL O.D. =	0.0	(0/0)		
AVERAGE POISSON RATIO(-)	=	0.33308		
AVERAGE YOUNG MODULUS(KG/CM**2) =	0.17501E 07			
*** FUEL (I=1,NFR) ***				
RADIUS(CM)	0.00101	0.00200	0.00300	0.27440
DISPLACEMENT(CM)	0.00000	0.00001	0.00001	0.00083
TEMP.(C)	371.585	371.585	371.585	371.568
PRESS.(KG/CM**2)	0.28939E 01	0.30113E 01	0.30144E 01	0.28960E 01
ALPHA-T(-)	0.00303	0.00303	0.00303	0.00303
POISSON RATIO(-)	0.31705	0.31550	0.31550	0.31550
YOUNG M.(KG/CM**2)	0.22370E 07	0.21741E 07	0.21741E 07	
NUMBER OF CRACK				
RADIAL	0	0	0	
CIRCUMFERENTIAL	0	0	0	
AXIAL	0	0	0	
STRESS(KG/CM**2)				
RADIAL	-0.29779E 01	-0.30132E 01	-0.28961E 01	
CIRCUMFERENTIAL	-0.31246E 01	-0.30203E 01	-0.28958E 01	
AXIAL	-0.34251E 01	-0.33525E 01	-0.31100E 01	
TOTAL STRAIN(-)				
RADIAL	0.30298E-02	0.30298E-02	0.30297E-02	
CIRCUMFERENTIAL	0.30297E-02	0.30298E-02	0.30297E-02	
AXIAL	0.30296E-02	0.30296E-02	0.30296E-02	
CREEP STRAIN(-)				
RADIAL	0.29693E-31	0.17317E-31	0.10684E-31	
CIRCUMFERENTIAL	0.76894E-32	0.16254E-31	0.10723E-31	
AXIAL	-0.37383E-31	-0.33571E-31	-0.21407E-31	
CREEP IN DELT(-)				
RADIAL	0.29693E-31	0.17317E-31	0.10684E-31	
CIRCUMFERENTIAL	0.76894E-32	0.16254E-31	0.10723E-31	
AXIAL	-0.37383E-31	-0.33571E-31	-0.21407E-31	
SWELLING STRAIN(-)				
TOTAL	0.72064E-09	0.39549E-09	0.17058E-09	
SOLID COMPONENT	0.21132E-09	0.21129E-09	0.50588E-09	
GAS COMPONENT	0.19506E-08	0.97517E-09	0.58502E-11	
HOT PRESSING	0.0	0.0	0.0	
PERCENT CHANGE IN ORIGINAL O.D. =	0.0	(0/0)		
AVERAGE POISSON RATIO(-)	=	0.31550		
AVERAGE YOUNG MODULUS(KG/CM**2) =	0.21741E 07			
GAP SIZE(CM)	=	0.00592		
FRICTION FORCE(KG/CM)	=	0.0	(OPEN)	

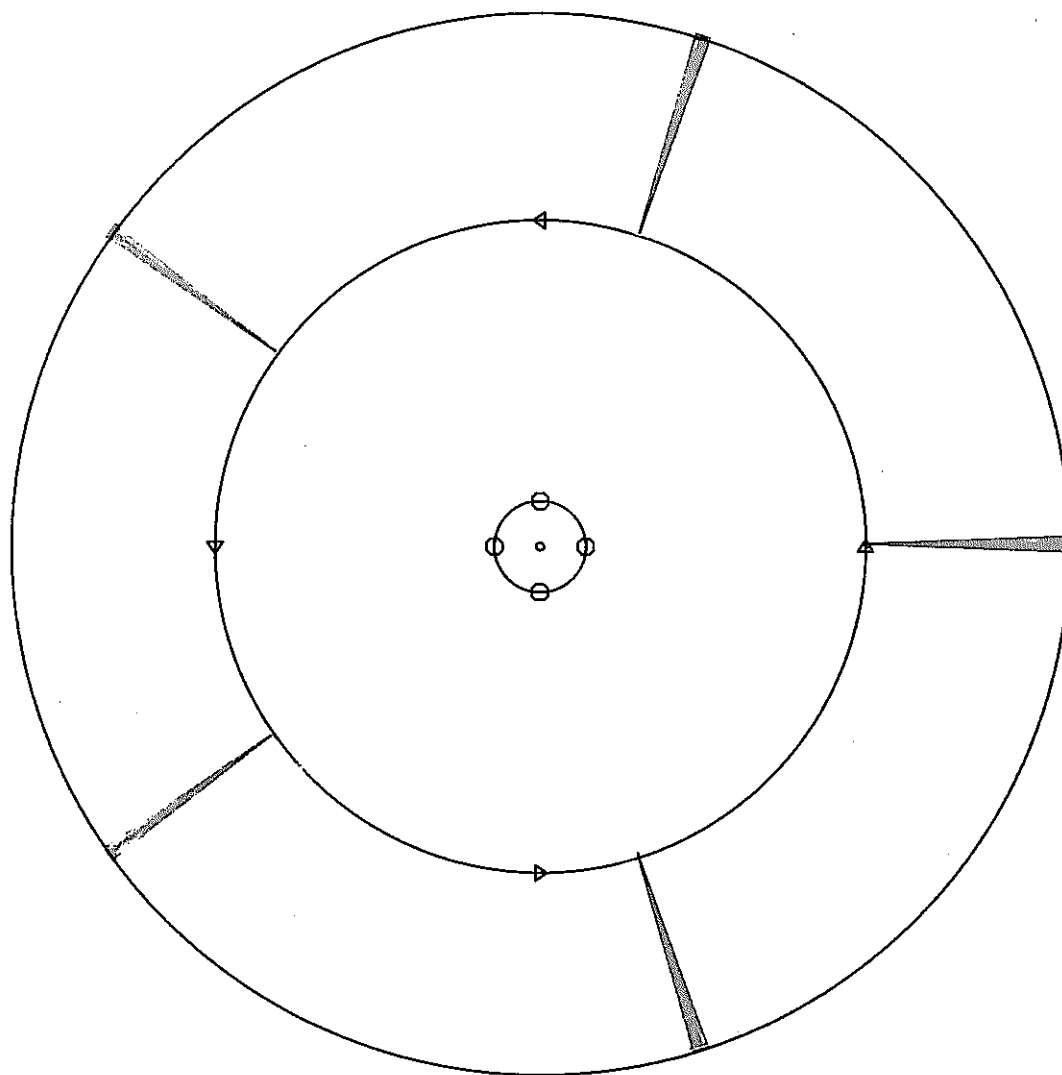
+++ PLUTONIUM MIGRATION (I=1,NP) +++											
RADIUS(CM)	0.00101	0.08678	0.12272	0.15030	0.17355	0.19403	0.21255	0.22958	0.24543	0.26032	0.27440
TEMP.(C)	371.585	371.583	371.582	371.580	371.578	371.576	371.575	371.573	371.571	371.570	371.568
PU CONC.(-)	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000	0.20000
*** AXIAL SECTION = 3 ***											
+++ CAPSULE (I=1,NCL) +++											
RADIUS(CM)	0.42540	0.47620									
TEMP.(C)	371.858	371.857									
ALPHA-T(-)	0.00627										
+++ CLAD (I=1,NCL) +++											
RADIUS(CM)	0.27940	0.31750									
DISPLACEMENT(CM)	0.00175	0.00199									
TEMP.(C)	371.859	371.858									
PRESS.(KG/CM**2)	0.28960E 01	0.16800E 01									
ALPHA-T(-)	0.00627										
POISSON RATIO(-)	0.33311										
YOUNG M.(KG/CM**2)	0.17499E 07										
STRESS(KG/CM**2)											
RADIAL	-0.22363E 01										
CIRCUMFERENTIAL	0.72244E 01										
AXIAL	0.21869E 01										
TOTAL STRAIN(-)											
RADIAL	0.62715E-02										
CIRCUMFERENTIAL	0.62787E-02										
AXIAL	0.62749E-02										
CREEP STRAIN(-)											
RADIAL	-0.58381E-15										
CIRCUMFERENTIAL	0.60964E-15										
AXIAL	-0.25828E-16										
CREEP IN DELT(-)											
RADIAL	-0.58381E-15										
CIRCUMFERENTIAL	0.60964E-15										
AXIAL	-0.25828E-16										
SWELLING STRAIN(-)											
TOTAL	0.53512E-15										
PERCENT CHANGE IN ORIGINAL O.D. =	0.0	(0/0)									
AVERAGE POISSON RATIO(-)	=	0.33311									
AVERAGE YOUNG MODULUS(KG/CM**2) =	0.17499E 07										
+++ FUEL (I=1,NFR) +++											
RADIUS(CM)	0.00101	0.00200	0.00300	0.27440							
DISPLACEMENT(CM)	0.00000	0.00001	0.00001	0.00083							
TEMP.(C)	371.880	371.880	371.880	371.866							
PRESS.(KG/CM**2)	0.28939E 01	0.29989E 01	0.30016E 01	0.28960E 01							
ALPHA-T(-)	0.00303	0.00303	0.00303	0.00303							
POISSON RATIO(-)	0.31705	0.31550	0.31550	0.31550							
YOUNG M.(KG/CM**2)	0.22369E 07	0.21740E 07	0.21740E 07	0.21740E 07							
NUMBER OF CRACK											
RADIAL	0	0	0	0							
CIRCUMFERENTIAL	0	0	0	0							
AXIAL	0	0	0	0							
STRESS(KG/CM**2)											

RADIAL	-0.29690E 01	-0.30006E 01	-0.28961E 01
CIRCUMFERENTIAL	-0.31001E 01	-0.30070E 01	-0.28959E 01
AXIAL	-0.32741E 01	-0.32059E 01	-0.29942E 01
TOTAL STRAIN(-)			
RADIAL	0.30325E-02	0.30324E-02	0.30324E-02
CIRCUMFERENTIAL	0.30324E-02	0.30324E-02	0.30324E-02
AXIAL	0.30323E-02	0.30323E-02	0.30323E-02
CREEP STRAIN(-)			
RADIAL	0.21816E-31	0.10586E-31	0.48960E-32
CIRCUMFERENTIAL	0.21448E-32	0.96273E-32	0.49309E-32
AXIAL	-0.23961E-31	-0.20214E-31	-0.98274E-32
CREEP IN DELT(-)			
RADIAL	0.21816E-31	0.10586E-31	0.48960E-32
CIRCUMFERENTIAL	0.21448E-32	0.96273E-32	0.49309E-32
AXIAL	-0.23961E-31	-0.20214E-31	-0.98274E-32
SWELLING STRAIN(-)			
TOTAL	0.63770E-09	0.34996E-09	0.15088E-09
SOLID COMPONENT	0.18692E-09	0.18689E-09	0.44748E-09
GAS COMPONENT	0.17262E-08	0.86298E-09	0.51771E-11
HOT PRESSING	0.0	0.0	0.0
PERCENT CHANGE IN ORIGINAL O.D. =	0.0	(0/0)	
AVERAGE POISSON RATIO(-)	=	0.31550	
AVERAGE YOUNG MODULUS(KG/CM**2) =	0.21740E 07		
GAP SIZE(CM)	=	0.00592	
FRICTION FORCE(KG/CM)	=	0.0	(OPEN)
+++ PLUTONIUM MIGRATION (I=1,NP) +++			
RADIUS(CM)	0.00101	0.08678	0.12272 0.15030 0.17355 0.19403 0.21255 0.22958 0.24543 0.26032 0.27440
TEMP.(C)	371.880	371.879	371.877 371.876 371.875 371.873 371.872 371.870 371.869 371.867 371.866
PU CONC.(-)	0.20000	0.20000	0.20000 0.20000 0.20000 0.20000 0.20000 0.20000 0.20000 0.20000 0.20000
*** AXIAL SECTION = 4 ***			
+++ CAPSULE (I=1,NCL) +++			
RADIUS(CM)	0.42540	0.47620	
TEMP.(C)	372.100	372.100	
ALPHA-T(-)	0.00628		
+++ CLAD (I=1,NCL) +++			
RADIUS(CM)	0.27940	0.31750	
DISPLACEMENT(CM)	0.00176	0.00199	
TEMP.(C)	372.100	372.100	
PRESS.(KG/CM**2)	0.28960E 01	0.16800E 01	
ALPHA-T(-)	0.00628		
POISSON RATIO(-)	0.33313		
YOUNG M.(KG/CM**2)	0.17496E 07		
STRESS(KG/CM**2)			
RADIAL	-0.22363E 01		
CIRCUMFERENTIAL	0.72244E 01		
AXIAL	0.23743E 01		
TOTAL STRAIN(-)			
RADIAL	0.62759E-02		
CIRCUMFERENTIAL	0.62831E-02		
AXIAL	0.62795E-02		

CREEP STRAIN(-)		
RADIAL	-0.18715E-15	
CIRCUMFERENTIAL	0.19033E-15	
AXIAL	-0.31848E-17	
CREEP IN DELT(-)		
RADIAL	-0.18715E-15	
CIRCUMFERENTIAL	0.19033E-15	
AXIAL	-0.31848E-17	
SWELLING STRAIN(-)		
TOTAL	0.95831E-16	
PERCENT CHANGE IN ORIGINAL O.D. =		
	0.0	(0/0)
AVERAGE POISSON RATIO(-) =		
	0.33313	
AVERAGE YOUNG MODULUS(KG/CM**2) =		
	0.17496E 07	
CPU TIME = 0.40 (SEC)		

** RUN DATE **
7/31/75

ACTIVE-II



--- CIRCUMF. CRACK

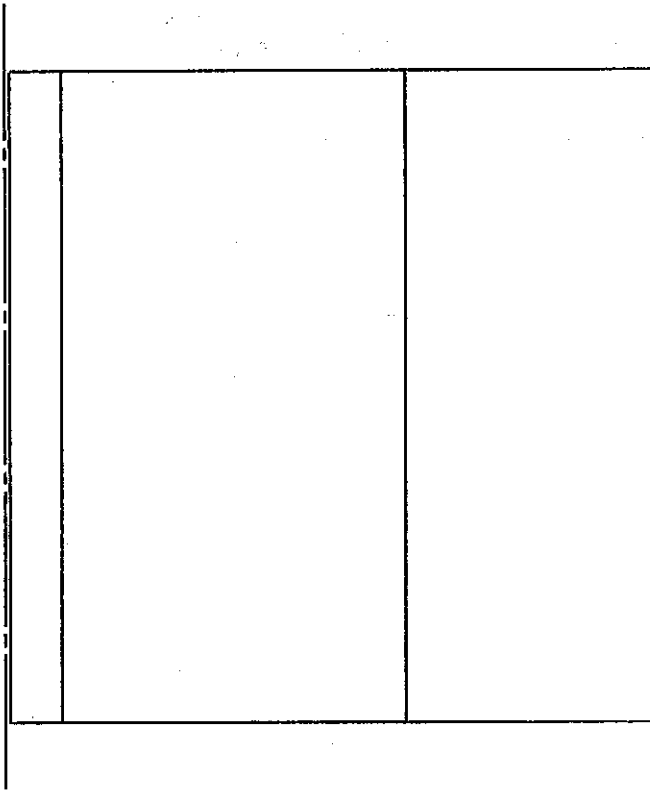
--- RADIAL CRACK

⊙ COLUMNAR BNDRY

△ EQUIAXED BNDRY

R-C-DIRECTION CRACK

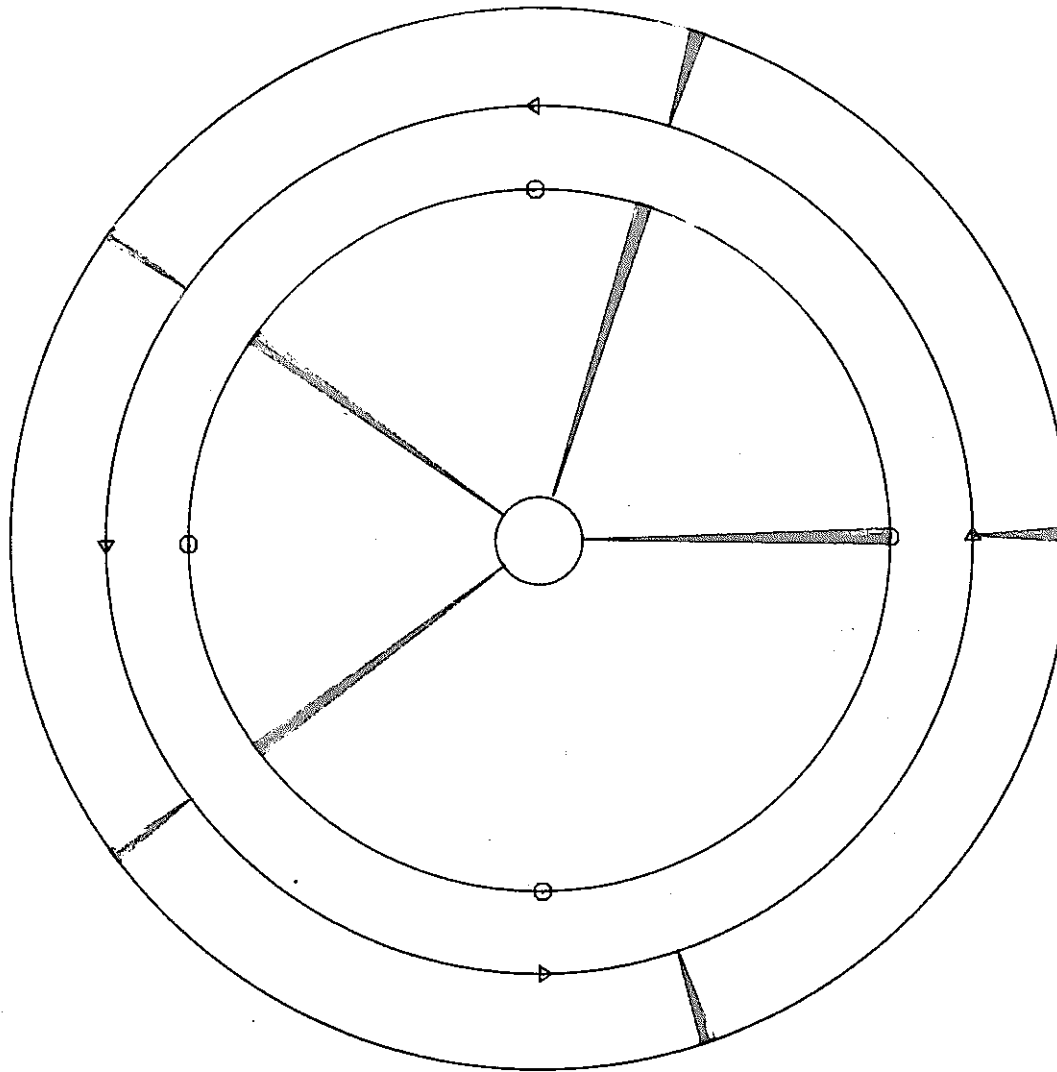
AXIAL SECTION = 2 STEP = 36



- AXIAL CRACK
- COLUMNAR GR. REGION
- EQUIAXED GR. REGION
- UNDISTURBED GR. REGION

Z-DIRECTION CRACK

AXIAL SECTION = 2 STEP = 36



--- CIRCUMF. CRACK

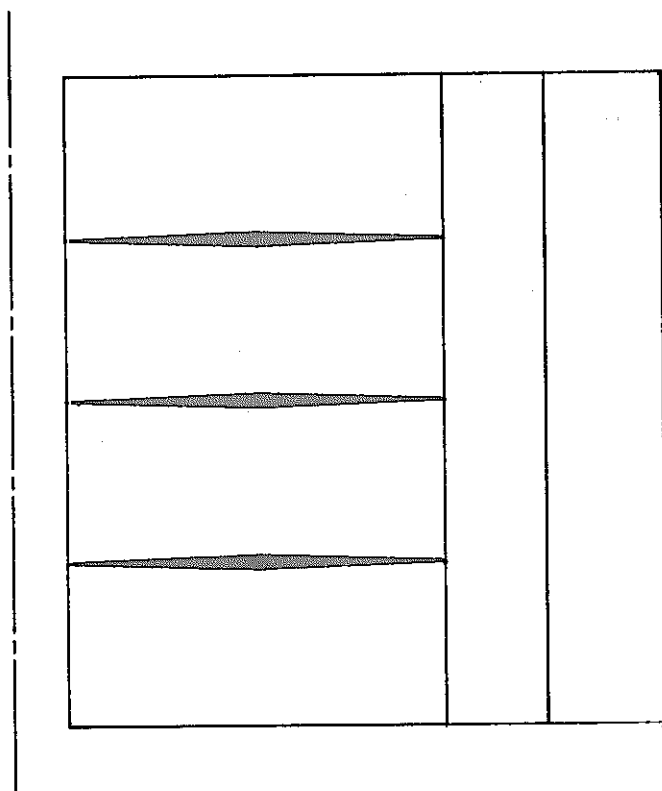
--- RADIAL CRACK

○ COLUMNAR BNDRY

△ EQUIAXED BNDRY

R-C-DIRECTION CRACK

AXIAL SECTION = 2 STEP - 122

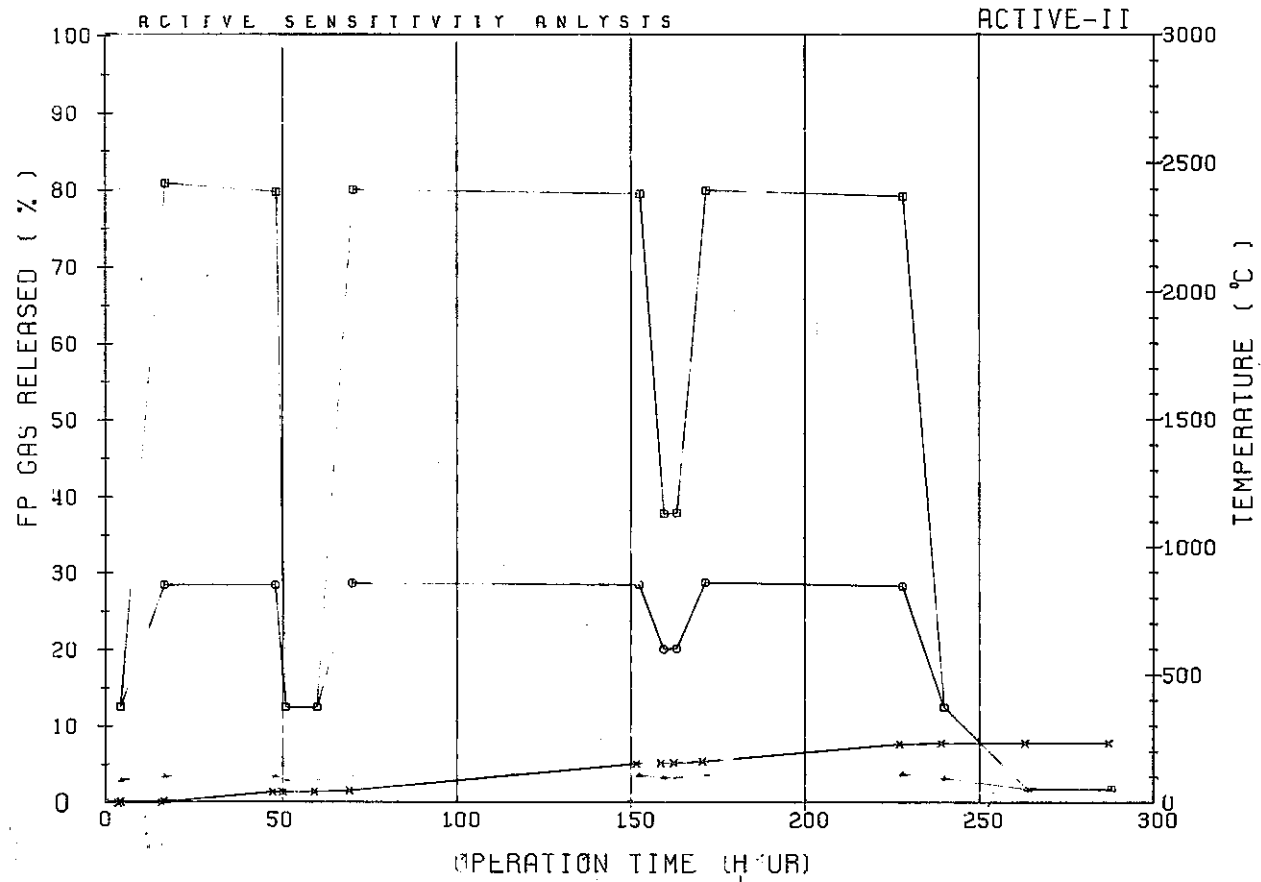


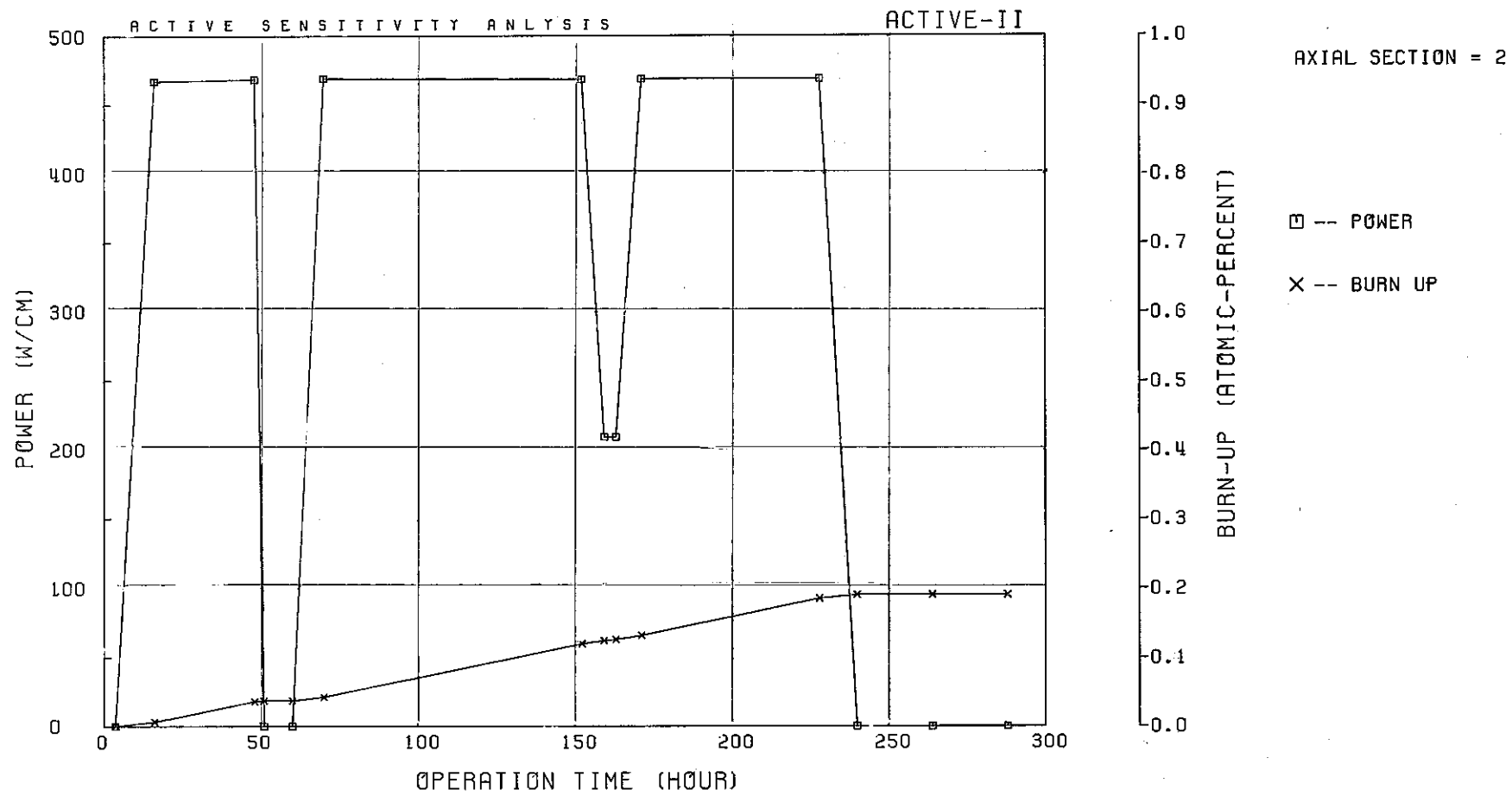
- AXIAL CRACK
- COLUMNAR GR. REGION
- EQUIAXED GR. REGION
- UNDISTURBED GR. REGION

Z-DIRECTION CRACK

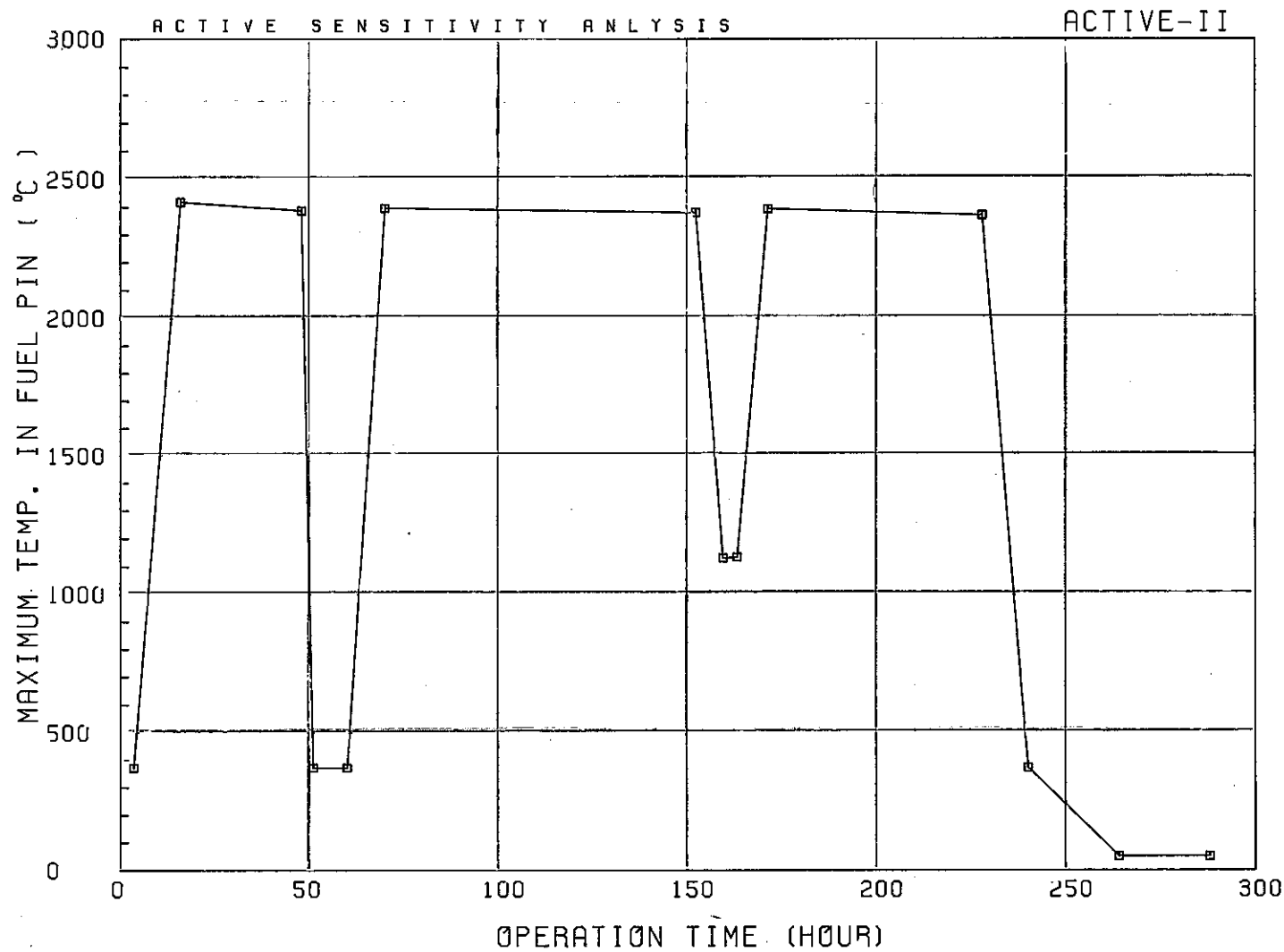
AXIAL SECTION = 2 STEP = 122

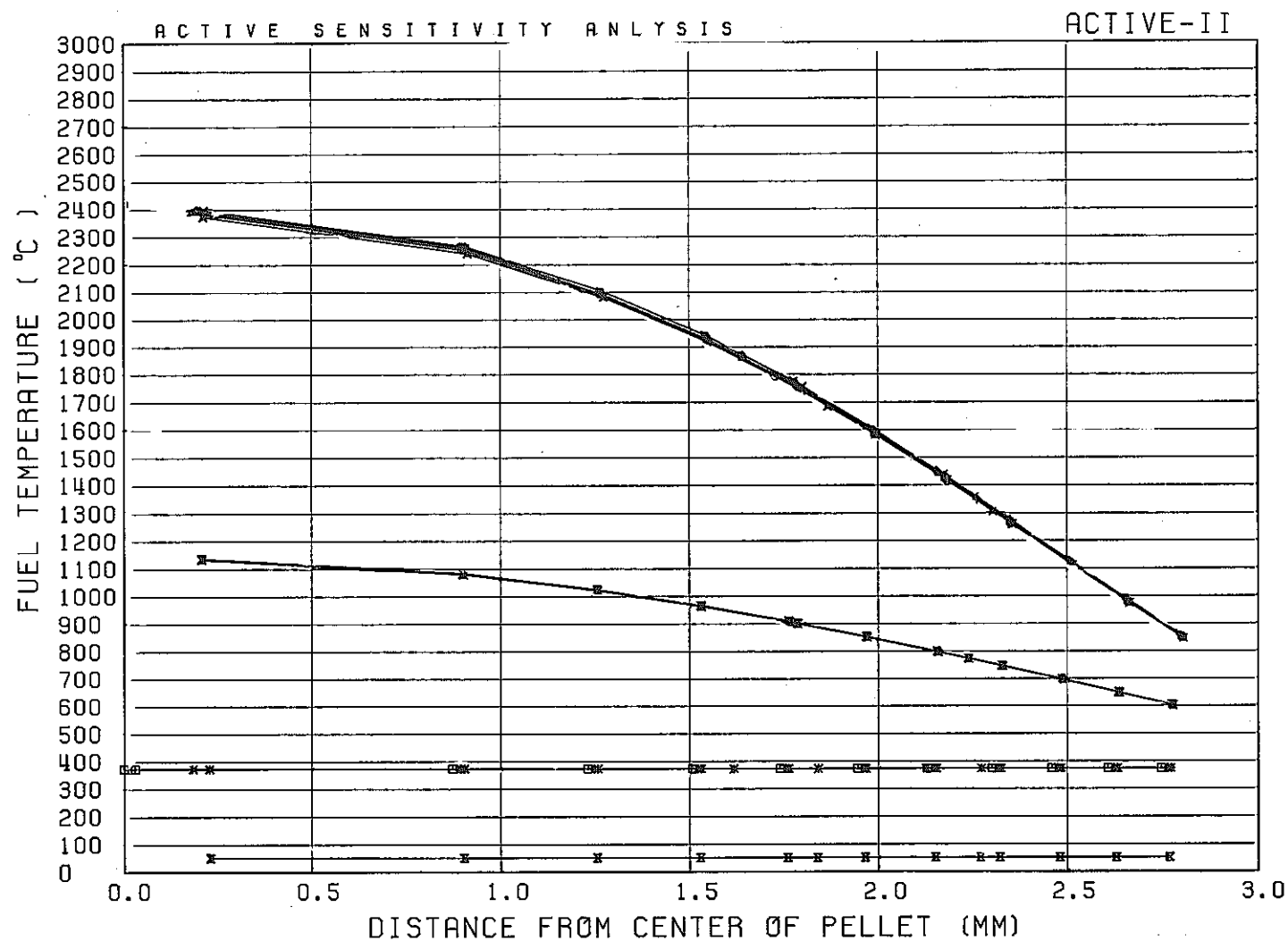
AXIAL SECTION - 2



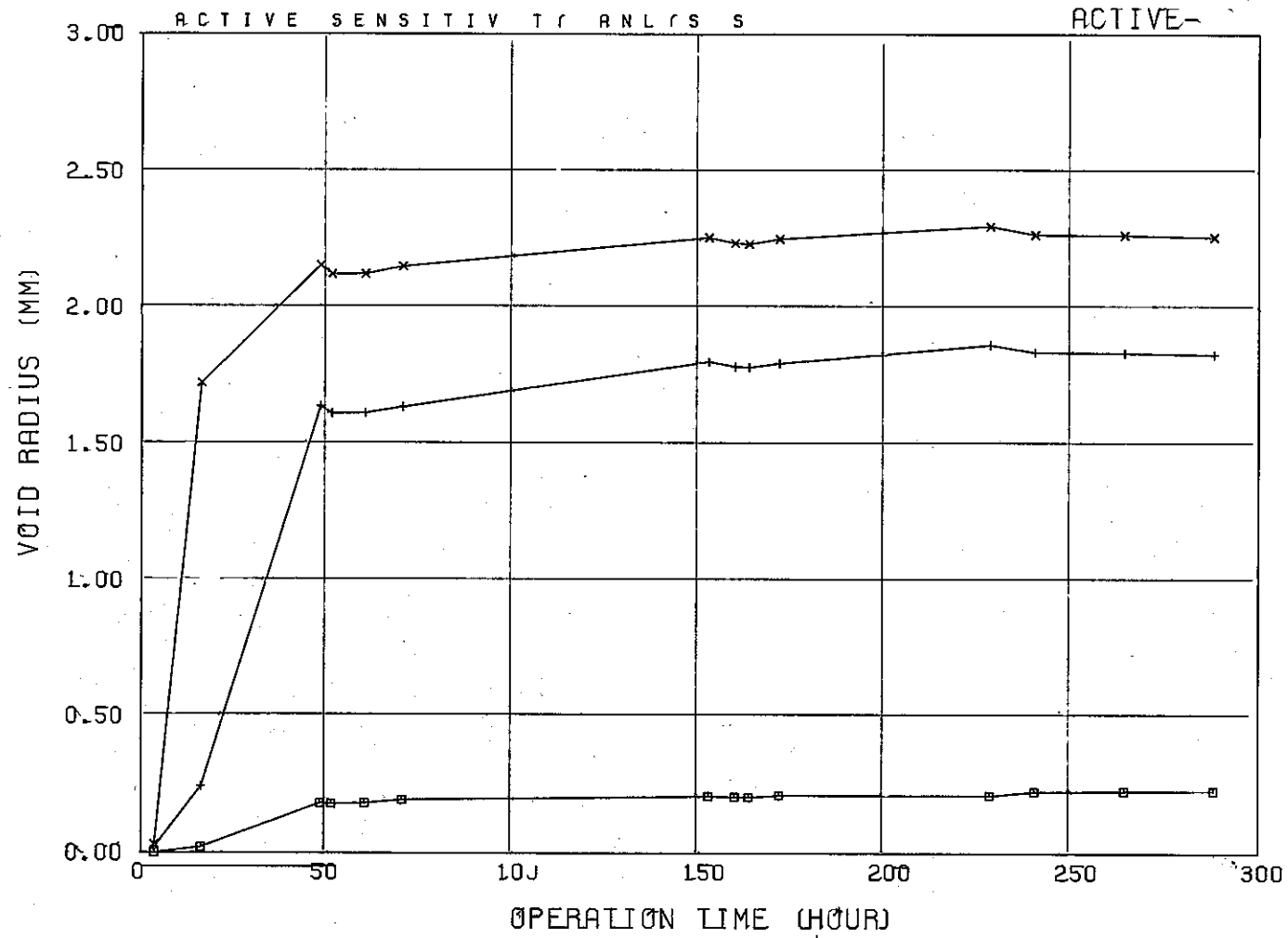


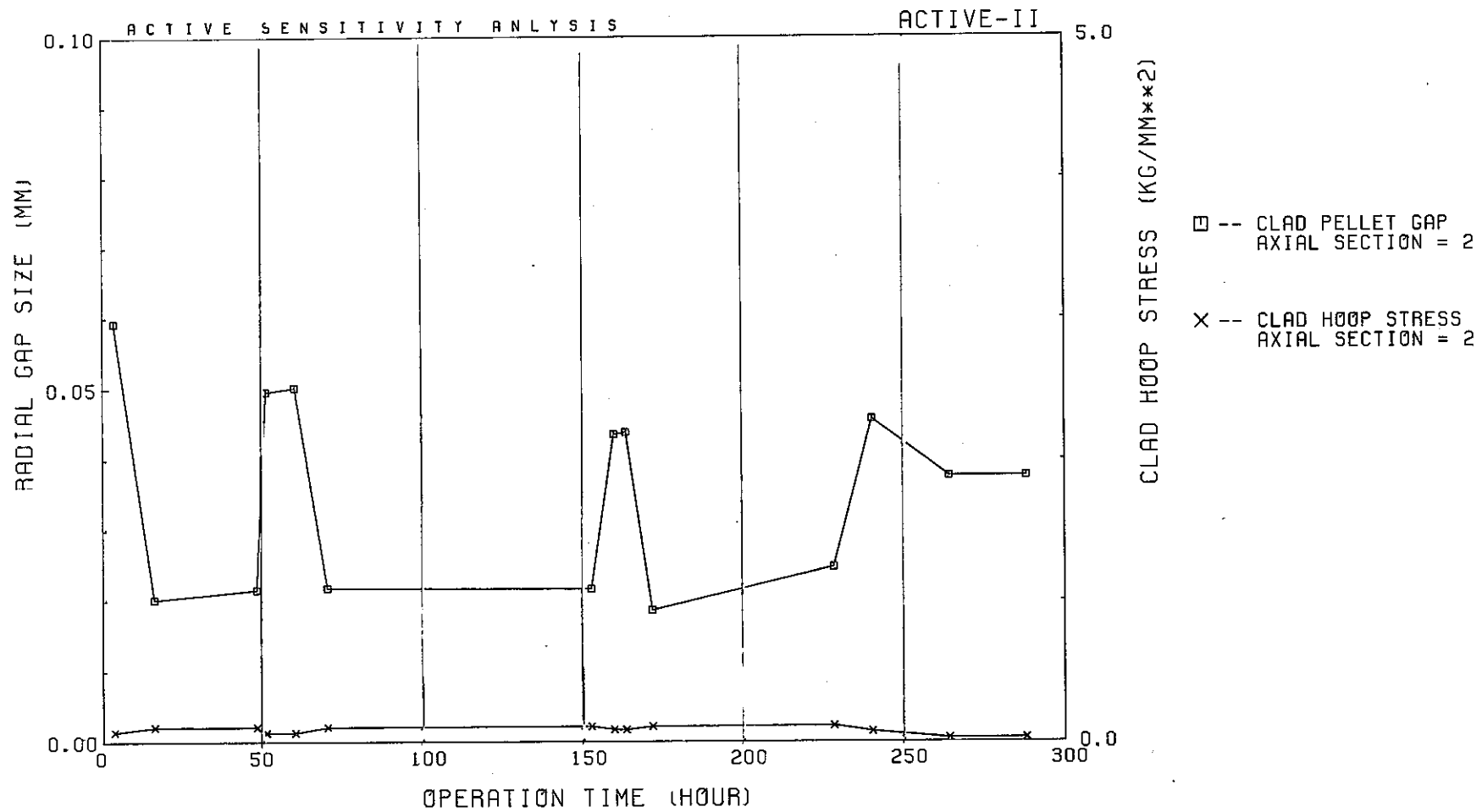
AXIAL SECTION = 2

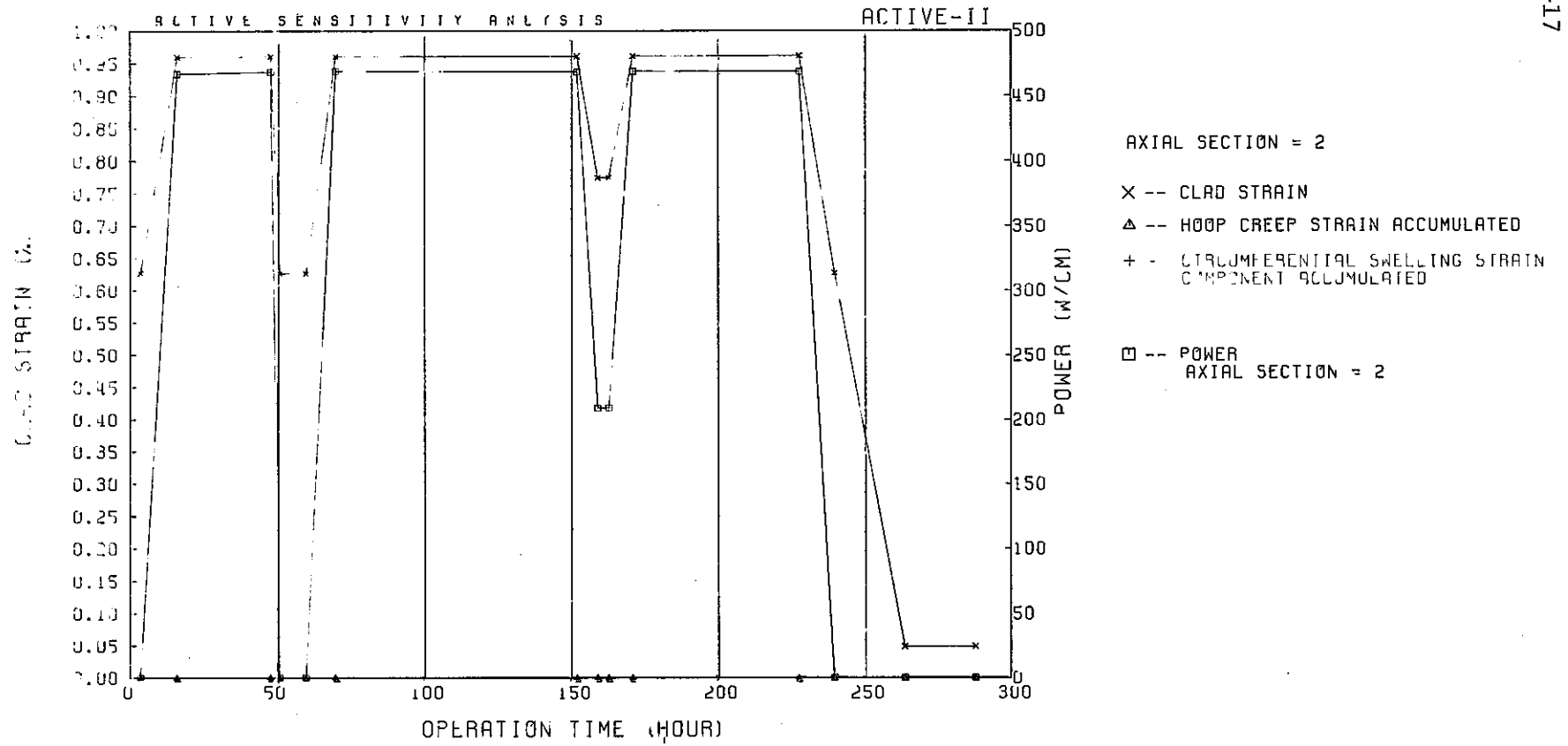


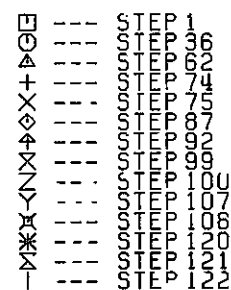


1	STEP	36	---	□
2	STEP	62	---	□
3	STEP	74	---	□
4	STEP	75	---	□
5	STEP	89	---	□
6	STEP	92	---	□
7	STEP	99	---	□
8	STEP	107	---	□
9	STEP	110	---	□
10	STEP	111	---	□
11	STEP	111	---	□
12	STEP	121	---	□
13	STEP	122	---	□
14	STEP	---	---	□
15	STEP	---	---	□
16	STEP	---	---	□
17	STEP	---	---	□
18	STEP	---	---	□
19	STEP	---	---	□
20	STEP	---	---	□
21	STEP	---	---	□
22	STEP	---	---	□
23	STEP	---	---	□
24	STEP	---	---	□
25	STEP	---	---	□
26	STEP	---	---	□
27	STEP	---	---	□
28	STEP	---	---	□
29	STEP	---	---	□
30	STEP	---	---	□
31	STEP	---	---	□
32	STEP	---	---	□
33	STEP	---	---	□
34	STEP	---	---	□
35	STEP	---	---	□
36	STEP	---	---	□
37	STEP	---	---	□
38	STEP	---	---	□
39	STEP	---	---	□
40	STEP	---	---	□
41	STEP	---	---	□
42	STEP	---	---	□
43	STEP	---	---	□
44	STEP	---	---	□
45	STEP	---	---	□
46	STEP	---	---	□
47	STEP	---	---	□
48	STEP	---	---	□
49	STEP	---	---	□
50	STEP	---	---	□
51	STEP	---	---	□
52	STEP	---	---	□
53	STEP	---	---	□
54	STEP	---	---	□
55	STEP	---	---	□
56	STEP	---	---	□
57	STEP	---	---	□
58	STEP	---	---	□
59	STEP	---	---	□
60	STEP	---	---	□
61	STEP	---	---	□
62	STEP	---	---	□
63	STEP	---	---	□
64	STEP	---	---	□
65	STEP	---	---	□
66	STEP	---	---	□
67	STEP	---	---	□
68	STEP	---	---	□
69	STEP	---	---	□
70	STEP	---	---	□
71	STEP	---	---	□
72	STEP	---	---	□
73	STEP	---	---	□
74	STEP	---	---	□
75	STEP	---	---	□
76	STEP	---	---	□
77	STEP	---	---	□
78	STEP	---	---	□
79	STEP	---	---	□
80	STEP	---	---	□
81	STEP	---	---	□
82	STEP	---	---	□
83	STEP	---	---	□
84	STEP	---	---	□
85	STEP	---	---	□
86	STEP	---	---	□
87	STEP	---	---	□
88	STEP	---	---	□
89	STEP	---	---	□
90	STEP	---	---	□
91	STEP	---	---	□
92	STEP	---	---	□
93	STEP	---	---	□
94	STEP	---	---	□
95	STEP	---	---	□
96	STEP	---	---	□
97	STEP	---	---	□
98	STEP	---	---	□
99	STEP	---	---	□
100	STEP	---	---	□
101	STEP	---	---	□
102	STEP	---	---	□
103	STEP	---	---	□
104	STEP	---	---	□
105	STEP	---	---	□
106	STEP	---	---	□
107	STEP	---	---	□
108	STEP	---	---	□
109	STEP	---	---	□
110	STEP	---	---	□
111	STEP	---	---	□
112	STEP	---	---	□
113	STEP	---	---	□
114	STEP	---	---	□
115	STEP	---	---	□
116	STEP	---	---	









APPENDIX G: PROGRAM LIST

Appendix G: Program List

The program is available from;
Department of Nuclear Fuels Development,
Power Reactor and Nuclear Fuel Development Corporation
1-9-13 Akasaka, Minato-ku, Tokyo, JAPAN

PNCT841-75-17

COMMON /DIM/ IA,IAM	00001300
DIMENSION A(4000),AM(6000)	00001400
IA=4000	00001500
IAM=6000	00001600
CALL ACTIVE(A,AM)	00001700
STOP	00001800
END	00001900

SUBROUTINE ACTIVE (A,AM)	00000100
C	00000200
C	00000300
C=====	00000400
C=====	00000500
C===== ACTIVE-II =====	00000600
C=====	00000700
C=====	00000800
C==	==00000900
C== A COMPUTER ANALYSIS OF FAST/THERMAL-REACTOR FUEL-ELEMENT BEHAVIOR	==00001000
C== AS A FUNCTION OF REACTOR OPERATING HISTORY .	==00001100
C==	==00001200
C=====	==00001300
C	00001400
C	00001500
REAL*8 DAYDAY	00001600
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NQS,NOC,NPLOT,NCRK,	00001700
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00001800
* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00001900
* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00002000
* N12C,N13C,N14C,N15C,	00002100
* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00002200
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00002300
* PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00002400
* CPDW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00002500
COMMON /CONST/ FKA1,FKB1,FKC1,FKA2,FKB2,FKC2,FKA3,FKB3,FKC3,	00002600
* FA11,FA12,FA21,FA22,FA31,FA32,	00002700
* CK1,CK2,CA1,CA2,XEWTN(7)	00002800
COMMON /CAPSUL/ RCPI,RCPO,AFNA	00002900
COMMON /IOFIL/ IO5,IO6,IO7	00003000
COMMON /FBRATR/ JFA	00003100
COMMON /DIM/ IA,IAM	00003200
COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00003300
COMMON /DAY/ DAYDAY	00003400
C	00003500
DIMENSION A(1) , AM(1)	00003600
C	00003700
IO5 = 5	00003800
IO6 = 6	00003900
IO7 = 7	00004000
ISPLIT = 0	00004100
C	00004200
C---- TO GET THE DATE	00004300
C	00004400
CALL DATE (DAYDAY)	00004500
C	00004600
6000 CONTINUE	00004700
C	00004800
C---- TIMER ZERO CLEAR	00004900
C	00005000
CALL STIMER(2)	00005100
C	00005200
C	00005300
C---- CARD 1	00005400
C	00005500
READ(IO5,9006,END=998) JFA,JOUT,EPSSPD,EPSCRF,EPSCRC	00005600
C	00005700
C	00005800

IF (EPSSPD.EQ.0.) EPSSPD=0.01	00005900
IF (EPSCRF.EQ.0.) EPSCRF=0.05	00006000
IF (EPSCRC.EQ.0.) EPSCRC=0.01	00006100
C	00006200
C JFA = 1 ... FBR	00006300
C JFA = 2 ... ATR	00006400
C JFA = 3 ... PWR	00006500
C JFA = 4 ... BWR	00006600
C	00006700
C JOUT = 0 ... OUTPUT EVERY STEP OF CARD INPUT	00006800
C JOUT = 1 ... OUTPUT EVERY STEP CALCULATED	00006900
C	00007000
C EPSSPD = LIMIT FOR RESTRUCTURING VELOCITY (CM/SEC)	00007100
C EPSCRF = LIMIT FOR FUEL CREEP STRAIN (-)	00007200
C EPSCRC = LIMIT FOR CLADDING CREEP STRAIN (-)	00007300
C	00007400
GO TO (10,20,30,40) , JFA	00007500
10 WRITE(106,9004)	00007600
GO TO 50	00007700
20 WRITE(106,9005)	00007800
GO TO 50	00007900
30 WRITE(106,9007)	00008000
GO TO 50	00008100
40 WRITE(106,9008)	00008200
50 CONTINUE	00008300
C	00008400
C---- CARD 2 ... TITLE CARD (CONTINUED TILL *END CARD)	00008500
C	00008600
CALL HEADER	00008700
C	00008800
C	00008900
C---- CARD 3 ... CONTROL CARD	00009000
C	00009100
C	00009200
READ(105,9000) NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00009300
* NSKIP,NBUF	00009400
C	00009500
C	00009600
C NN = NUMBER OF AXIAL SECTION	00009700
C NCR = NUMBER OF MESH FOR COLUMNAR GRAIN REGION	00009800
C NER = NUMBER OF MESH FOR EQUIAXED GRAIN REGION	00009900
C NUR = NUMBER OF MESH FOR UNDISTURBED REGION	00010000
C NCL = NUMBER OF MESH FOR CLAD REGION	00010100
C NP = NUMBER OF MESH FOR PLUTONIUM MIGRATION	00010200
C NOK = FUEL THERMAL CONDUCTIVITY OPTION	00010300
C 1/2/3 = SAYLES/BAILY-ASAMOTO/BIANCHERIA	00010400
C NOS = CLAD SWELLING OPTION	00010500
C -1/-2 = ANNEALED 304,316S.S./20P.C.COLD WORK 316S.S.	00010600
C NOC = CLAD CREEP OPTION	00010700
C -1/-2 = BOLTAX,BIANCHERIA,GUHA/MODIFIED	00010800
C NPLOT = PLOT OPTION	00010900
C 0/1 = NO PLOT/PLOT	00011000
C NCRK = 0 ... NO CRACK CALC. AND 2 SIGE LOOPS	00011100
C NCRK = 1 ... CRACK CALC. AND 2 SIGE LOOPS	00011200
C NSKIP = COOLANT TEMPERATURE INPUT OPTION	00011300
C 0/1 = NO INPUT/INPUT TEMPERATURE DISTRIBUTION	00011400
C NBUF = STORAGE(WORDS) REQUIRED FOR PLOTTER	00011500
C IF NBUF=0, THEN CODE SETS NBUF 50	00011600

C	00011700
C	00011800
C===== LOCATION SET FOR DIMENSION A =====	00011900
C	00012000
NFR = NCR+NER+NUR	00012100
NFR1 = NFR+1	00012200
NCL1 = NCL + 1	00012300
NTR= NFR+NCL	00012400
NTR1=NTR+1	00012500
NTR2 = NTR+2	00012600
NNTR = NN*NTR	00012700
NNTR2 = NN*NTR2	00012800
NNFR = NN*NFR	00012900
NNFR1 = NN*NFR1	00013000
NNCL = NN*NCL	00013100
NNCL1 = NN*NCL1	00013200
NP1 = NP + 1	00013300
NNP = NN*NP	00013400
NNP1 = NN*NP1	00013500
NNF = NN-1	00013600
N1 = NTR	00013700
N2 = NTR*2	00013800
N3 = NTR*3	00013900
N4 = NTR*4	00014000
N5 = NTR*5	00014100
N6 = NTR*6	00014200
N7 = NTR*7	00014300
N8 = NTR*8	00014400
N9 = NTR*9	00014500
N10 = NTR*10	00014600
N11 = NTR*11	00014700
N12 = NTR*12	00014800
N13 = NTR*13	00014900
N14 = NTR*14	00015000
N15 = NTR*15	00015100
N1C = NCL	00015200
N2C = NCL*2	00015300
N3C = NCL*3	00015400
N4C = NCL*4	00015500
N5C = NCL*5	00015600
N6C = NCL*6	00015700
N7C = NCL*7	00015800
N8C = NCL*8	00015900
N9C = NCL*9	00016000
N10C = NCL*10	00016100
N11C = NCL*11	00016200
N12C = NCL*12	00016300
N13C = NCL*13	00016400
N14C = NCL*14	00016500
N15C = NCL*15	00016600
C	00016700
NIO = 1	00016800
NEPR = NIO + NN	00016900
NEPC = NEPR + NNTR	00017000
NEPZ = NEPC + NNTR	00017100
NTPR = NEPZ + NNTR	00017200
NTPC = NTPR + NNTR	00017300
NTPZ = NTPC + NNTR	00017400

NTSR = NTPZ + NNTR	00017500
NTSC = NTSR + NNTR	00017600
NTSZ = NTSC + NNTR	00017700
NSIGR = NTSZ + NNTR	00017800
NSIGC = NSIGR + NNTR	00017900
NSIGZ = NSIGC + NNTR	00018000
NADIS = NSIGZ + NNTR	00018100
NRDIS = NADIS + NNTR	00018200
NRVB = NRDIS + NNTR2	00018300
NRCB = NRVB + NN	00018400
NREB = NRCB + NN	00018500
NRUB = NREB + NN	00018600
NRCI = NRUB + NN	00018700
NRCO = NRCI + NN	00018800
NQSFF = NRCO + NN	00018900
NQSFP = NQSFF + NN	00019000
NF = NQSFP + NN	00019100
NOTOM = NF + NNP	00019200
NDXM = NOTOM + NN	00019300
NXM = NDXM + 3*NN	00019400
NZFLX = NXM + 3*NN	00019500
NRH = NZFLX + NN	00019600
NRHOP = NRH + NNTR	00019700
NCPU = NRHOP + 3*NN	00019800
NLM = NCPU + NNP	00019900
NEHP = NLM + NNTR	00020000
NSWS = NEHP + NNTR	00020100
NGS = NSWS + NNTR	00020200
NGR = NGS + NNFR	00020300
NQSF = NGR + NN	00020400
NPOW = NQSF + NN	00020500
NXCPI = NPOW + NN	00020600
NXCPO = NXCPI + NN	00020700
NBU = NXCPO + NN	00020800
NBUM = NBU + NN	00020900
NHG = NBUM + NN	00021000
NTCLO = NHG + NN	00021100
NHC = NTCLO + NN	00021200
NTNA = NHC + NN	00021300
NTCPO = NTNA + NN	00021400
NTCPI = NTCPO + NN	00021500
NRCP = NTCPI + NN	00021600
NTCP = NRCP + NNCL1	00021700
NTBCP = NTCP + NNCL1	00021800
NTBCPA = NTBCP + NNCL	00021900
NATBCP = NTBCPA + NN	00022000
NATBPA = NATBCP + NNCL	00022100
NRCL = NATBPA + NN	00022200
NTCL = NRCL + NNCL1	00022300
NTBCL = NTCL + NNCL1	00022400
NTBCLA = NTBCL + NNCL	00022500
NATBCL = NTBCLA + NN	00022600
NATBLA = NATBCL + NNCL	00022700
NYNG = NATBLA + NN	00022800
NPOS = NYNG + NNTR	00022900
NPOSB = NPOS + NNTR	00023000
NSHR = NPOSB + 2*NN	00023100
NSHRB = NSHR + NNTR	00023200

NBM = NSHRB + 2*NN	00023300
NRFR = NBM + NNTR	00023400
NTER = NRFR + NNFR1	00023500
NTBF = NTER + NNFR1	00023600
NTBFA = NTBF + NNFR	00023700
NATBF = NTBFA + 3*NN	00023800
NATBFA = NATBF + NNFR	00023900
NRFRPM = NATBFA + 3*NN	00024000
NTERPM = NRFRPM + NNPI	00024100
NGDT = NTERPM + NNPI	00024200
NTBFPM = NGDT + NNPI	00024300
NATBPM = NTBFPM + NNP	00024400
NFDPM = NATBPM + NNP	00024500
NPRS = NFDPM + NNP	00024600
NFLZO = NPRS + NNTR	00024700
NFLZ = NFLZO + NN	00024800
NCLZO = NFLZ + NN	00024900
NCLZ = NCLZO + NN	00025000
NZL = NCLZ + NN	00025100
NFD = NZL + NN	00025200
NSWG = NFD + NNFR	00025300
NSWS3 = NSWG + NNFR	00025400
NFLNC = NSWS3 + NN*NUR	00025500
NFLUX = NFLNC + NN	00025600
NDEQP = NFLUX + NN	00025700
NEQSIG = NDEQP + NNTR	00025800
NFTHP = NEQSIG + NNTR	00025900
NVOL = NFTHP + NNTR	00026000
NKFRIC = NVOL + NNTR	00026100
NCRIF = NKFRIC + NN	00026200
NRHOT = NCRIF + NN	00026300
NRVBO = NRHOT + NN	00026400
NRCBO = NRVBO + NN	00026500
NREBO = NRCBO + NN	00026600
NRUBO = NREBO + NN	00026700
NRCIO = NRUBO + NN	00026800
NRCOO = NRCIO + NN	00026900
NRFRO = NRCOO + NN	00027000
NRCLO = NRFRO + NNFR1	00027100
NCPUD = NRCLO + NNCL1	00027200
NTOUR = NCPUD + NP	00027300
NTOUO = NTOUR + NN	00027400
NPTOU = NTOUO + NN	00027500
NDELT = NPTOU + NN	00027600
NXMINT = NDELT + NN	00027700
NRDIST = NXMINT + NN	00027800
NEPSEQ = NRDIST + 4*NN	00027900
NRBAR = NEPSEQ + NN*NNTR	00028000
NWOLD = NRBAR + NNTR	00028100
NWNEW = NWOLD + NNFR	00028200
NXEPR = NWNEW + NFR	00028300
NXEPC = NXEPR + NFR	00028400
NXTPR = NXEPC + NFR	00028500
NXTPC = NXTPR + NFR	00028600
NXTPZ = NXTPC + NFR	00028700
NXTSR = NXTPZ + NFR	00028800
NXTSC = NXTSR + NFR	00028900
NXTSZ = NXTSC + NFR	00029000

NXGS = NXTSZ + NFR	00029100
NXSWG = NXGS + NFR	00029200
NXSWS = NXSWG + NFR	00029300
NXSWS3 = XSWS + NFR	00029400
NXEHP = NXSWS3 + NUR	00029500
NLMR = NXEHP + NFR	00029600
NLMC = NLMR + NNTR	00029700
NLMZ = NLMC + NNTR	00029800
MEMO = NLMZ + NNTR	00029900
IPCNT = 0	00030000
IF (NPLT.EQ.0) GO TO 110	00030100
IF (NBUF.EQ.0) NBUF = 50	00030200
NPIT = NLMZ + NNTR	00030300
NPPOW = NPIT + NBUF	00030400
NPTIME = NPPOW + NBUF	00030500
NPPLPR = NPTIME + NBUF	00030600
NPYP = NPPLPR + NBUF	00030700
NPRL = NPYP + NBUF	00030800
NPTL = NPRL + NPI*NBUF	00030900
NPT = NPTL + NPI*NBUF	00031000
NPTLMX = NPT + NBUF	00031100
NPRLR = NPTLMX + NBUF	00031200
NPTLR = NPRLR + NBUF*2	00031300
NPCPGP = NPTLR + NBUF*2	00031400
NPCPU = NPCPGP + NBUF	00031500
NPBU = NPCPU + NP*NBUF	00031600
NPSIGC = NPBU + NBUF	00031700
NPSTN = NPSIGC + NBUF	00031800
NPTSC = NPSTN + NBUF	00031900
NPTPC = NPTSC + NBUF	00032000
MEMO = NPTPC + NBUF	00032100
110 CONTINUE	00032200
C	00032300
C	00032400
LRVB = MEMO	00032500
LRCB = LRVB + NN	00032600
LREB = LRCB + NN	00032700
LRUB = LREB + NN	00032800
LRCI = LRUB + NN	00032900
LRCO = LRCI + NN	00033000
LBU = LRCO + NN	00033100
LRDIST = LBU + NN	00033200
LDXM = LRDIST + 4*NN	00033300
LFLNC = LDXM + 3*NN	00033400
LWOLD = LFLNC + NN	00033500
LTOUR = LWOLD + NNFR	00033600
MEMO = LTOUR + NN	00033700
C	00033800
C	00033900
C===== LOCATION SET FOR MATRIX (DIMENSION AM)	00034000
C	00034100
IGEN = N10 + 12	00034200
NONZR = 56*NNTR + 24	00034300
C	00034400
C	00034500
IGUESS = 10	00034600
C	00034700
C	00034800

C	IBOT = NUMBER OF NON-ZERO ELEMENT AFTER L / U RESOLUTION	00034900
C		00035000
C		00035100
	IBOT = IGUESS * NONZR	00035200
C		00035300
	IGEN1 = IGEN + 1	00035400
	IGENP = 10*NCL + 7	00035500
	IGENP1 = IGENP + 1	00035600
	NONZRP = 54*NCL + 12	00035700
C		00035800
	MRES = 1	00035900
	MAUX = MRES + IGEN	00036000
	MIA = MAUX + IGEN	00036100
	MJA = MIA + IGEN1	00036200
	MA = MJA + NONZR	00036300
	MIC = MA + NONZR	00036400
	MJC = MIC + IGEN1	00036500
	MIU = MJC + IBOT	00036600
	MDI = MIU + IGEN	00036700
	MC = MDI + IGEN	00036800
	MIPR = MC + IBOT	00036900
	MIPC = MIPR + IGEN	00037000
	MKA = MIPC + IGEN	00037100
	MIER = MKA + IGEN	00037200
	MB = MIER + 5	00037300
	MEMOMX = MB + IGEN	00037400
C		00037500
C		00037600
	MEPR = MRES	00037700
	MEPC = MEPR + NTR	00037800
	MSIGR = MEPC + NTR	00037900
	MSIGC = MSIGR + NTR	00038000
	MSIGZ = MSIGC + NTR	00038100
	MTPR = MSIGZ + NTR	00038200
	MTPC = MTPR + NTR	00038300
	MTPZ = MTPC + NTR	00038400
	MTPRF = MTPZ + NTR	00038500
	MTPRC = MTPRF + 1	00038600
	MTPCF = MTPRC + 1	00038700
	MTPCC = MTPCF + 1	00038800
	MTPZF = MTPCC + 1	00038900
	MTPZC = MTPZF + 1	00039000
	MEPZF = MTPZC + 1	00039100
	MEPZC = MEPZF + 1	00039200
	MDISP = MEPZC + 1	00039300
	MPRES = MDISP + NTR2	00039400
	MFRIC = MPRES + NTR1	00039500
	MDSR = MSIGR	00039600
	MDSC = MSIGC	00039700
	MDSZ = MSIGZ	00039800
C		00039900
C	FOR PLENUM SECTION ONLY	00040000
C		00040100
	MEPRP = MRES	00040200
	MEPCP = MEPRP + NCL	00040300
	MSIGRP = MEPCP + NCL	00040400
	MSIGCP = MSIGRP + NCL	00040500
	MSIGZP = MSIGCP + NCL	00040600

MTPRP = MSIGZP + NCL	00040700
MTPCP = MTPRP + NCL	00040800
MTPZP = MTPCP + NCL	00040900
MTPRCP = MTPZP + NCL	00041000
MTPCCP = MTPRCP + 1	00041100
MTPZCP = MTPCCP + 1	00041200
MEPZCP = MTPZCP + 1	00041300
MDISPP = MEPZCP + 1	00041400
MPRESP = MDISPP + NCL1	00041500
MFRICP = MPRESP + NCL1	00041600
MDSRP = MSIGRP	00041700
MDSCP = MSIGCP	00041800
MDSZP = MSIGZP	00041900
C	00042000
C	00042100
C=====	00042200
C	00042300
C	00042400
C---- READ CARD 4 TO CARD 23	00042500
C	00042600
C	00042700
C	00042800
CALL INPUT (A(NEPZ),A(NTPR),A(NTPC),A(NTPZ),A(NSIGR),A(NSIGC),	00042900
* A(NSIGZ),A(NADIS),A(NRDIS),A(NRVB),A(NRCB),A(NREB),	00043000
* A(NRUB),A(NRCI),A(NRCO),A(NQSFF),A(NQSFP),A(NF),	00043100
* A(NEPR),A(NEPC),A(NTSR),A(NTSC),A(NTSZ),A(NCPU),	00043200
* A(NRHOP),A(NOTOM),FL,CL,PTOP,PO,EPSIN,A(NCLZO))	00043300
C	00043400
C---- MATERIAL PROPERTIES AND CONSTANTS	00043500
C	00043600
CALL PROPTY (A(NCPU),A(NRHOP),A(NRHOT))	00043700
C	00043800
C	00043900
C=====	00044000
C===== INITIAL SET =====	00044100
C=====	00044200
C	00044300
C	00044400
C	00044500
C---- SET I/O DEVICE FOR WORK FILE	00044600
C	00044700
CALL IOSET (A(NIO))	00044800
C	00044900
C---- INITIAL VALUES SET	00045000
C	00045100
CALL BEGIN (A(NXM),A(NZFLX),A(NRH),A(NRHOP),A(NCPU),	00045200
* A(NLM),A(NEHP),A(NSWS),A(NGS),A(NGR),A(NRVB),	00045300
* A(NRCB),A(NREB),A(NRUB),A(NQSF),A(NQSFF),A(NQSFP),	00045400
* A(NPOW),A(NXCPI),A(NXCPO),A(NBU),A(NHG),A(NKERIC),	00045500
* A(NFLZO),A(NFLZ),A(NPRS),A(NFLNC),	00045600
* A(NCLZO),A(NCLZ),FL,CL,A(NDXM),A(NSWG),A(NRCI),	00045700
* A(NRCO),A(NRVBO),A(NRCBO),A(NREBO),A(NRUBO),	00045800
* A(NRCIO),A(NRCOO),A(NXMINT),A(NRDIST),A(NEPSEQ),	00045900
* A(NSWS3),A(NLMR),A(NLMC),A(NLMZ))	00046000
C	00046100
C---- INPUT VALUES EDIT	00046200
C	00046300
CALL EDITNP (A(NRVB),A(NRCB),A(NREB),A(NRUB),A(NRCI),A(NRCO),	00046400

	*	A(NQSFF),A(NQSFP),A(NOTOM),A(NCPU),A(NRHOP),A(NF),	00046500
	*	A(NXCPI),A(NXCPO),A(NFLZ),A(NCLZ),FL,CL,PTOP,PO)	00046600
C			00046700
		MEMOC = (MEMO+MEMOMX+53000)*4/128000	00046800
		MEMOC = (MEMOC+1)*128	00046900
		WRITE(I06,9002) MEMO , IA , MEMOMX , IAM , MEMOC	00047000
C			00047100
		IT = 0	00047200
		ITPLOT = 0	00047300
		TIM = 0.	00047400
		TIMO = 0.	00047500
		TOGAS = GASI	00047600
		POWAO = 0.	00047700
		PPOWA = 0.	00047800
		PPD = 1.033	00047900
		PTOUT = 20.	00048000
		PTIN = 20.	00048100
		JKL = 1	00048200
		DELTO = 1.E+10	00048300
		CPOW = 1.0	00048400
		FLO = FL	00048500
		FLN = FL	00048600
		CLO = CL	00048700
		CLN = CL	00048800
		KJI = 0	00048900
		KJIK = 0	00049000
C			00049100
C			00049200
C			00049300
C		===== TIME STEP LOOP =====	00049400
C			00049500
C			00049600
C			00049700
		140 CONTINUE	00049800
C			00049900
		IT=IT+1	00050000
C			00050100
		CALL UPDTE (A(NRVB),A(NRCB),A(NREB),A(NRUB),A(NRCI),A(NRCO),	00050200
	*	A(NBU),A(NRDIST),A(NDXM),A(NFLNC),A(NWOLD),	00050300
	*	A(LRVB),A(LRCB),A(LREB),A(LRUB),A(LRCI),A(LRCO),	00050400
	*	A(LBU),A(LRDIST),A(LDXM),A(LFLNC),A(LWOLD),1)	00050500
		IF (KJI.EQ.0) GO TO 141	00050600
		TIP = TIM	00050700
		TIM = YTIM	00050800
		TOUT = YTOUT	00050900
		TIN = YTIN	00051000
		PO = YPO	00051100
		POWAV = YPOWAV	00051200
		IF (NSKIP.GT.0) CALL CORTMP (A(NTOUR),A(LTOUR),A(NPTOU),1,TOUT)	00051300
		DELT = TIM-TIP	00051400
		KJI = 0	00051500
		JJOUT = 1	00051600
		GO TO 390	00051700
		141 CONTINUE	00051800
		KJIK = 0	00051900
		IF (JKL-1) 160,160,150	00052000
150		TIP=TIM	00052100
		TIM=TIMO	00052200

	IF (NSKIP.GT.0) CALL CORTMP(A(NTOUR),A(NTOUO),A(NPTOU),1,TOUT)	00052300
	TOUT=TOUTO	00052400
	TIN=TINO	00052500
	PO = POO	00052600
	POWAV=POWAO	00052700
	JKL=1	00052800
	GO TO 360	00052900
160	CONTINUE	00053000
	TIP=TIM	00053100
C		00053200
C	=====	00053300
C		00053400
C		00053500
C		00053600
C	---- CARD 24 ... OPERATING HISTORY	00053700
C		00053800
C		00053900
	163 READ(IO5,9001,END=999) TIM,POI,REPOW,CF,TIN,TOUT,NPLOT	00054000
C		00054100
C		00054200
	TIM=TIM*24.0	00054300
	IF (TIM-TIMO) 170,180,180	00054400
170	IF(NPLOT.EQ.0) GO TO 6000	00054500
C		00054600
	CALL PLTOR (A(NPPOW),A(NPRL),A(NPTL),A(NPT),A(NPTLMX),	00054700
*	A(NPRLR),A(NPTLR),A(NPCPGP),A(NPCPU),A(NPBU),	00054800
*	A(NPSIGC),A(NPCSTN),A(NPTSC),A(NPTPC),	00054900
*	A(NPTIME),A(NPPLPR),A(NPYP),A(NPIT),ITPLOT,IT)	00055000
C		00055100
	GO TO 6000	00055200
180	CONTINUE	00055300
	IF (NSKIP.GT.0) CALL CORTMP(A(NTOUR),A(NTOUO),A(NPTOU),0,TOUT)	00055400
	IF(POI.GT.0.) PO=POI	00055500
	JJOUT = 1	00055600
	IF (CF) 210,210,200	00055700
200	CFLX=CF	00055800
210	CONTINUE	00055900
	POWAV=REPOW*CPW	00056000
	IF (TOUT) 230,230,240	00056100
230	TOUT=TIN+TCON*POWAV*SMQ/SMQO*FLN/FLO	00056200
	GO TO 250	00056300
240	TCON=(TOUT-TIN)/POWAV/SMQ*SMQO/FLN*FLO	00056400
250	IF (TIM-TIP-0.01) 260,270,270	00056500
260	TIM=TIP+0.01	00056600
270	IF (POWAV-0.01) 280,290,290	00056700
280	POWAV=0.01	00056800
290	IF (TOUT-TIN-1.) 300,310,310	00056900
300	TOUT=TIN+1.	00057000
C		00057100
310	TIMO = TIM	00057200
	TOUTO=TOUT	00057300
	POO=PO	00057400
	TINO=TIN	00057500
	POWAO = POWAV	00057600
	IF (NSKIP.GT.0) CALL CORTMP(A(NTOUR),A(NTOUO),A(NPTOU),2,TOUT)	00057700
C		00057800
C		00057900
360	DELT=TIM-TIP	00058000

C		00058100
C		00058200
C	CALL TIMCUT(CX,DELTO,EPSPOW)	00058300
C		00058400
C	---- TIME INTERVAL CHANGE CHECK	00058500
C		00058600
C	370 IF (DELT-CX*DELTO) 385,380,380	00058700
C		00058800
C	---- POWER CHANGE CHECK	00058900
C		00059000
C	385 IF (ABS(POWAV-PPOWA)-EPSPOW) 390,380,380	00059100
C		00059200
C		00059300
C	380 CONTINUE	00059400
C		00059500
C	---- TIME INTERVAL CUT	00059600
C		00059700
C	TIM=TIP+DELT/2.	00059800
C	POWAV=PPOWA+(POWAV-PPOWA)/2.	00059900
C	TOUT=PTOUT+(TOUT-PTOUT)/2.	00060000
C	PO=PPO+(PC-PPO)/2.	00060100
C	TIN=PTIN+(TIN-PTIN)/2.	00060200
C	IF (NSKIP.GT.0) CALL CORTMP(A(NTOUR),A(NTOUO),A(NPTOU),3,TOUT)	00060300
C	DELT=DELT/2.	00060400
C	JKL=2	00060500
C	GO TO 370	00060600
C		00060700
C	PREVIOUS POWAV PRESUMES NO BURNUP OF THE FUEL ELEMENT.	00060800
C		00060900
C	390 CONTINUE	00061000
C	YSMQ = SMQ/SMQO	00061100
C	391 CONTINUE	00061200
C	POWERC = POWAV	00061300
C	POWERC = COLD POWER	00061400
C	POWAV = POWAV*YSMQ	00061500
C		00061600
C	---- RESTRUCTURING VELOCITY CHECK	00061700
C		00061800
C	CALL CHKSPD (A(NDXM),EPSSPD,NN,ICLK,DELT)	00061900
C	IF (ICHK.GT.0) GO TO 417	00062000
C		00062100
C	---- REGION BOUNDARY MOVEMENT	00062200
C		00062300
C	CALL RHOMOD (A(NRH),A(NRHOP),A(NEPZ),A(NRVB),A(NRCB),	00062400
*	A(NREB),A(NRUB),A(NBU),A(NXM),A(NRHOT),A(NRDIS),	00062500
*	A(NDXM),A(NRVBO),A(NRCBO),A(NREBO),A(NRUBO),	00062600
*	A(NRCI),A(NRCO),A(NXMINT),A(NEPC),A(NRDIST),A(NEPR),	00062700
*	A(NXEPR),A(NXEPC),A(NRFR),A(NWOLD),A(NWNEW),IT,	00062800
*	DELT)	00062900
C		00063000
C	---- CLADDING OUTER SURFACE TEMPERATURE	00063100
C		00063200
C	CALL GOLDN (POWAV,A(NQSF),TOUT,TIN,A(NRCO),A(NPDW),A(NTCLO),	00063300
*	RCPI,RCPO,AFNA,FL,A(NHC),A(NTNA),A(NTCPO),A(NTCPI),	00063400
*	A(NTOUR),NSKIP,PO,A(NFLZ))	00063500
C		00063600
C	IF(RCPO-RCPI) 400,400,410	00063700
C		00063800

C----	CAPSULE TEMPERATURE DISTRIBUTION	00063900
C		00064000
	410 CALL CLTMP (A(NPOW),A(NTCPO),A(NXCPO),A(NXCPI),A(NRCP),A(NTCP),	00064100
	* A(NTBCP),A(NTBCPA),A(NATBCP),A(NATBPA),A(NRCLO),IT,	00064200
	* A(NRBAR))	00064300
C		00064400
C----	CLADDING TEMPERATURE DISTRIBUTION	00064500
C		00064600
	400 CALL CLTMP (A(NPOW),A(NTCLO),A(NRCO),A(NRCI),A(NRCL),A(NTCL),	00064700
	* A(NTBCL),A(NTBCLA),A(NATBCL),A(NATBLA),A(NRCLO),IT,	00064800
	* A(NRBAR))	00064900
C		00065000
C----	FUEL-CLADDING GAP CONDUCTANCE	00065100
C		00065200
	CALL GAPCON (A(NRCI),A(NRUB),A(NHG))	00065300
C		00065400
C----	FUEL BURN-UP	00065500
C		00065600
	CALL BURN (A(NBU),A(NPOW),A(NXM),A(NEPZ),A(NQSF),A(NQSEF),	00065700
	* DELT)	00065800
C		00065900
C----	FLUX , FLUENCE	00066000
C		00066100
	CALL FLXCAL (POWAV,A(NQSFF),A(NZFLX),A(NFLNC),A(NFLUX),DELT)	00066200
C		00066300
C----	FUEL TEMPERATURE DISTRIBUTION , RESTRUCTURING	00066400
C		00066500
	CALL FLTMP	00066600
	* (A(NPOW),A(NTCL),A(NRCI),A(NRUB),A(NREB),A(NRCB),A(NRVB),	00066700
	* A(NRH),A(NDXM),A(NXM),A(NRFR),A(NTFR),A(NTBF),A(NATBF),	00066800
	* A(NTBFA),A(NATBFA),A(NFD),A(NRFRPM),A(NTFRPM),A(NTBFPM),	00066900
	* A(NATBPM),A(NFDPM),A(NGDT),A(NHG),A(NF),A(NJTM),A(NRHOT),	00067000
	* DELT,A(NRFR),IT,A(NBU),A(NRHOP),A(NRBAR),	00067100
	* A(NEPR),A(NEPC),A(NEPZ),A(NTPR),A(NTPC),A(NTPZ),A(NTSR),	00067200
	* A(NTSC),A(NTSZ),A(NGS),A(NSWG),A(NSWS),A(NEHP),	00067300
	* A(NWOLD),A(NWNEW),A(NXEPR),A(NXEPC),A(NXTPR),A(NXTPC),	00067400
	* A(NXTPZ),A(NXTSR),A(NXTSC),A(NXTSZ),A(NXGS),A(NXSWG),	00067500
	* A(NXSWS),A(NXEHP),A(NXSWS3),A(NSWS3))	00067600
C		00067700
C----	AXIAL LENGTH	00067800
C		00067900
	CALL LENGTH (FL,FLN,CL,CLN,A(NFLZO),A(NFLZ),A(NCLZO),A(NCLZ),	00068000
	* A(NEPZ))	00068100
C		00068200
C----	VOLUME	00068300
C		00068400
	CALL VOLUME (A(NVOL),A(NRFR),A(NRCL),A(NFLZ),A(NCLZ))	00068500
C		00068600
	DO 415 NS=1,NN	00068700
C		00068800
C----	YOUNG'S MODULUS , POISSON'S RATIO	00068900
C		00069000
	CALL XLAME(A(NTBCL),A(NTBF),A(NYNG),A(NPOS),A(NSHR),A(NRH),	00069100
	* A(NLM),A(NBM),A(NPOSB),A(NSHRB),A(NVOL),A(NRHOT),	00069200
	* NS)	00069300
C		00069400
C----	EQUIVALENT STRESS	00069500
C		00069600

	CALL EQUISI (A(NSIGR),A(NSIGC),A(NSIGZ),A(NEQSIG),	00069700
	* A(NPRS),NS)	00069800
C		00069900
C----	CLADDING CREEP STRAIN	00070000
C		00070100
	CALL CREEC (A(NEQSIG),A(NDEQP),A(NTBCL),A(NFLNC),A(NFLUX),	00070200
	* DELT,NS)	00070300
C		00070400
C		00070500
	IF (NS.EQ.NN) GO TO 415	00070600
C		00070700
C----	FUEL CREEP STRAIN	00070800
C		00070900
	CALL CREEF (A(NEQSIG),A(NDEQP),A(NTBF),A(NPOW),A(NFD),A(NFLUX),	00071000
	* A(NFTHP),A(NLM),DELT,NS,A(NSHR),A(NEPSEQ))	00071100
	415 CONTINUE	00071200
C		00071300
C----	EQUIVALENT CREEP STRAIN INCREMENT CHECK	00071400
C		00071500
	CALL CHECK (A(NDEQP),EPSCRF,EPSCRC,ICLK)	00071600
	IF (ICHK.EQ.0) GO TO 425	00071700
	417 CONTINUE	00071800
	CALL UPDTE (A(NRVB),A(NRCB),A(NREB),A(NRUB),A(NRCI),A(NRCO),	00071900
	* A(NBU),A(NRDIST),A(NDXM),A(NFLNC),A(NWOLD),	00072000
	* A(LRVB),A(LRCB),A(LREB),A(LRUB),A(LRCI),A(LRCO),	00072100
	* A(LBU),A(LRDIST),A(LDXM),A(LFLNC),A(LWOLD),2)	00072200
	POWAV = POWERC	00072300
	IF (KJIK.NE.0) GO TO 420	00072400
	YTIM = TIM	00072500
	YTOUT = TOUT	00072600
	YTIN = TIN	00072700
	YPO = PO	00072800
	YPOWAV = POWAV	00072900
	IF (NSKIP.GT.0) CALL CORTMP(A(NTOUR),A(LTOUR),A(NPTOU),2,TOUT)	00073000
	KJIK = 1	00073100
	420 CONTINUE	00073200
	KJI = 1	00073300
	IF (JOUT.EQ.0) JJOUT=0	00073400
C		00073500
C----	TIME INTERVAL CUT	00073600
C		00073700
	TIM = (TIM+TIP) / 2.0	00073800
	POWAV = (POWAV+PPOWA) / 2.0	00073900
	TOUT = (TOUT+PTOUT) / 2.0	00074000
	TIN = (TIN+PTIN) / 2.0	00074100
	PO = (PO+PPO) / 2.0	00074200
	DELT = DELT / 2.0	00074300
	IF (NSKIP.GT.0) CALL CORTMP (A(NTOUR),A(LTOUR),A(NPTOU),3,TOUT)	00074400
	GO TO 391	00074500
	425 CONTINUE	00074600
C		00074700
C----	PLUTONIUM MIGRATION	00074800
C		00074900
	CALL PLUTO (A(NCPU),A(NRVB),A(NRCB),A(NREB),A(NRFRPM),A(NTFRPM),	00075000
	* A(NGDT),A(NXM),A(NRH),A(NEPZ),DELT,A(NCPUD))	00075100
C		00075200
C----	GAS RELEASE RATE	00075300
C		00075400

	CALL GASOUT (A(NRFR),A(NFLZ),A(NPOW),A(NFD),A(NTBF),A(NVOL),	00075500
*	A(NEPR),A(NEPC),A(NEPZ),A(NATBF),A(NSWS),A(NGR),	00075600
*	A(NGS),TOGAS,DELT)	00075700
C		00075800
C----	PLENUM PRESSURE	00075900
C		00076000
	CALL PLENP (A(NRVB),A(NRCI),A(NTFR),A(NFLZ),TOGAS,FL,CL,TOUT,	00076100
*	PLPR,CVPR,A(NRUB),A(NCLZ))	00076200
C		00076300
C----	CLADDING SWELLING STRAIN	00076400
C		00076500
	CALL HKNS (A(NTBCL),A(NTSR),A(NTSC),A(NTSZ),A(NFLUX),DELT)	00076600
C		00076700
C----	FUEL SWELLING STRAIN DUE TO SOLID FISSION PRODUCTS	00076800
C		00076900
	CALL SWELL (A(NPOW),A(NSWS),A(NFD),A(NSWS3),DELT)	00077000
C		00077100
	ISP = 0	00077200
C		00077300
	IF(JOUT.EQ.1) GO TO 440	00077400
	IF(JKL.NE.1) GO TO 450	00077500
	IF(JJOUT.LE.0) GO TO 450	00077600
C		00077700
	440 CONTINUE	00077800
C		00077900
C		00078000
	CALL EDTGRS (IT,TIM,POWAV,TIN,TOUT,DELT,A(NPOW),A(NFLUX),A(NBU),	00078100
*	A(NTNA),A(NXCPO),A(NXCPI),A(NTCPI),A(NTCPO),	00078200
*	A(NRCO),A(NRCI),A(NTCLO),A(NTCL),A(NFLZ),A(NCLZ),	00078300
*	FL,CL,A(NOTOM),A(NCPU),A(NRHOT),A(NRH),A(NXM),	00078400
*	A(NRVB),A(NRCB),A(NREB),A(NRUB),A(NRFR),	00078500
*	A(NRCL),A(NHG),A(NHC),TOGAS,A(NGR),A(NTFR),	00078600
*	A(NTBFA),A(NATBFA),A(NTBCLA),A(NATBLA),A(NRCP),	00078700
*	A(NTCP),A(NTBCPA),A(NATBPA),A(NFLNC),A(NGS),	00078800
*	PLPR,CVPR,FLO,CLO ,A(NBUM),PO,A(NFD),A(NFDPM))	00078900
	ISP = 1	00079000
	ITPLOT = ITPLOT + 1	00079100
C		00079200
	450 CONTINUE	00079300
C		00079400
C		00079500
C====	AXIAL SECTION LOOP GOING FROM LOWER TO UPPER SECTION =====	00079600
C		00079700
C		00079800
	DO 1000 NS=1,NN	00079900
	IF(NS.EQ.NN) GO TO 600	00080000
C		00080100
	JCR = 0	00080200
C		00080300
	490 CONTINUE	00080400
	MOUNT = 0	00080500
	KONV = 0	00080600
C	IFRIC = 0	00080700
	CALL CLEAR(IFRIC,A(NKFRIC),NS)	00080800
C		00080900
C	IFRIC = 0 ... GAP OPEN	00081000
C	IFRIC = 1 ... STICK	00081100
C	IFRIC = 2 ... SLIP	00081200

C		00081300
C	IF (JCR.GT.0) GO TO 500	00081400
C		00081500
C----	FUEL SWELLING STRAIN , HOT-PRESSING	00081600
C		00081700
	CALL ITERAT (A(NPRS),A(NVOL),A(NTBF),A(NATBF),A(NSWG),	00081800
*	A(NEHP),A(NRH),A(NRFR),A(NGS),A(NSWS),A(NBM),	00081900
*	A(NEPR),A(NEPC),A(NEPZ),A(NSWS3),A(NSIGR),A(NSIGC),	00082000
*	A(NSIGZ),A(NRHOT),DELT,NS,A(NRHOP),A(NBU),	00082100
*	A(NTSR),A(NTSC),A(NTSZ),A(NXEHP),A(NXSWG),	00082200
*	PLPR,CVPR)	00082300
C		00082400
C	500 CONTINUE	00082500
C		00082600
C----	DECIDE IA , JA , A ... ARGUMENT TO MSNR SUBROUTINE (SL/MATH)	00082700
C		00082800
	CALL MATRX (AM(MIA),AM(MJA),AM(MA),A(NYNG),A(NPOS),A(NDEQP),	00082900
*	A(NEQSIG),A(NRFR),A(NRCL),A(NSHR),A(NPOSB),A(NSHRB),	00083000
*	A(NFLZ),A(NKFRIC),MOUNT,IFRIC,NS,A(NRBAR))	00083100
C		00083200
C		00083300
C----	PIVOTING AND L / U RESOLUTION	00083400
C		00083500
	CALL MSNR (IGEN,AM(MIA),AM(MJA),AM(MA),IBOT,AM(MIC),AM(MJC),	00083600
*	AM(MIU),AM(MDI),AM(MC),AM(MIPR),AM(MIPC),AM(MIER),	00083700
*	AM(MKA),AM(MRES))	00083800
C		00083900
C		00084000
C----	B ARRAY CALCULATION OF AX=B	00084100
C		00084200
C		00084300
	CALL BCALC (AM(MB),A(NATBF),A(NATBCL),A(NTPR),A(NTPC),	00084400
*	A(NTPZ),A(NPRS),A(NRFR),A(NRCL),A(NPOS),A(NBM),	00084500
*	A(NSHR),A(NPOSB),A(NSHRB),A(NVOL),A(NRH),PLPR,CVPR,	00084600
*	PO,PTOP,DELT,A(NTSR),A(NTSC),A(NTSZ),A(NFLUX),	00084700
*	A(NTBCL),A(NKFRIC),A(NSWG),A(NSWS),A(NEHP),A(NEPR),	00084800
*	A(NEPC),A(NEPZ),MOUNT,IFRIC,NS,A(NRBAR))	00084900
C		00085000
C----	SOLVE AX=B	00085100
C		00085200
	CALL MBSR (IGEN,AM(MIPR),AM(MIPC),AM(MIC),AM(MIU),AM(MJC),	00085300
*	AM(MDI),AM(MC),AM(MB),AM(MRES),AM(MAUX))	00085400
C		00085500
C	AM (MRES) ARE FINAL RESULTS	00085600
C		00085700
C		00085800
C----	FUEL-CLADDING GAP ASSUMPTION CHECK	00085900
C		00086000
	CALL FRICHK (A(NRUB),A(NRCI),AM(MDISP),AM(MPRES),AM(MFRIC),	00086100
*	A(NCRIF),A(NKFRIC),KONV,IFRIC,NS)	00086200
C		00086300
	MOUNT = MOUNT + 1	00086400
C		00086500
	IF (KONV.EQ.0) GO TO 500	00086600
	IF (NCRK.LE.0) GO TO 550	00086700
C		00086800
C----	FUEL CRACKING	00086900
C		00087000

CALL CRACK	00087100
* (AM(MSIGH),AM(MSIGC),AM(MSIGZ),A(NBM),A(NSHR),A(NTBF),	00087200
* A(NTPR),A(NTPC),A(NTPZ),A(NTSR),A(NTSC),A(NTSZ),	00087300
* A(NPRS),A(NLM),A(NLMR),A(NLMC),A(NLMZ),DELT,NS,JCR)	00087400
IF (JCR.EQ.0) GO TO 550	00087500
C	00087600
C---- RE-COMPUTE YOUNG'S MODULUS , POISSON'S RATIO	00087700
C	00087800
CALL XLAME(A(NTBCL),A(NTBF),A(NYNG),A(NPOS),A(NSHR),A(NRH),	00087900
* A(NLM),A(NBM),A(NPOSB),A(NSHRB),A(NVOL),A(NRHOT),	00088000
* NS)	00088100
GO TO 490	00088200
550 CONTINUE	00088300
C	00088400
C	00088500
C	00088600
C IF CONVERGED , THEN UPDATE VALUES	00088700
C	00088800
C	00088900
CALL UPDATE (AM(MSIGH),AM(MSIGC),AM(MSIGZ),A(NSIGH),A(NSIGC),	00089000
* A(NSIGZ),AM(MEPR),AM(MEPC),AM(MEPZF),AM(MEPZC),	00089100
* A(NEPR),A(NEPC),A(NEPZ),AM(MTPR),AM(MTPC),AM(MTPZ),	00089200
* A(NTPR),A(NTPC),A(NTPZ),AM(MDSR),AM(MDSC),AM(MDSZ),	00089300
* A(NTSR),A(NTSC),A(NTSZ),AM(MDISP),A(NRDIS),NS,	00089400
* A(NEPSEQ))	00089500
C	00089600
IF(JOUT.EQ.1) GO TO 590	00089700
IF (JKL.NE.1) GO TO 650	00089800
IF(JJOUT.LE.0) GO TO 650	00089900
C	00090000
590 CONTINUE	00090100
C	00090200
C---- WRITE RESULTS OF EACH AXIAL SECTION	00090300
C	00090400
CALL EDTMTX (A(NEPR),A(NEPC),A(NEPZ),A(NSIGH),A(NSIGC),A(NSIGZ),	00090500
* A(NTPR),A(NTPC),A(NTPZ),A(NTSR),A(NTSC),A(NTSZ),	00090600
* A(NRDIS),A(NCRIF),A(NRCP),A(NTCP),A(NTBCP),	00090700
* A(NATBCP),A(NRCL),A(NTCL),A(NTBCL),A(NATBCL),	00090800
* A(NYNG),A(NPOS),A(NSHR),A(NBM),A(NRFR),A(NTFR),	00090900
* A(NTBF),A(NATBF),A(NPRS),A(NEHP),AM(MTPRF),	00091000
* AM(MTPRC),AM(MTPCF),AM(MTPCC),AM(MTPZF),AM(MTPZC),	00091100
* AM(MPRES),A(NKFRIC),AM(MTPR),AM(MTPC),	00091200
* AM(MTPZ),AM(MDSR),AM(MDSC),AM(MDSZ),A(NPOSB),	00091300
* A(NSHRB),A(NSWG),A(NSWS),NS,A(NRFRPM),A(NTFRPM),	00091400
* A(NGS),A(NCPU),A(NLM),A(NLMR),A(NLMC),A(NLMZ),	00091500
* A(NRCOO))	00091600
C	00091700
GO TO 650	00091800
C	00091900
600 CONTINUE	00092000
C	00092100
C---- FOLLOWING ARE FOR ONLY PLENUM SECTION	00092200
C	00092300
CALL MATRXP (AM(MIA),AM(MJA),AM(MA),A(NYNG),A(NPOS),A(NDEQP),	00092400
* A(NEQSIG),A(NRCL),A(NSHR),A(NPOSB),A(NSHRB),	00092500
* A(NCLZ),A(NKFRIC),NS)	00092600
C	00092700
CALL MSNR (IGENP,AM(MIA),AM(MJA),AM(MA),IBOT,AM(MIC),AM(MJC),	00092800

	*	AM(MIU),AM(MDI),AM(MC),AM(MIPR),AM(MIPC),AM(MIER),	00092900
	*	AM(MKA),AM(MRES))	00093000
C			00093100
		CALL BCALCP (AM(MB),A(NATBCL),A(NTPR),A(NTPC),A(NTPZ),A(NRCL),	00093200
	*	A(NPOS),A(NSHR),A(NPOSB),A(NSHRB),A(NVOL),A(NRH),	00093300
	*	A(NTBCL),A(NFLUX),A(NKFRIC),A(NCRIF),DELT,A(NTSR),	00093400
	*	A(NTSC),A(NTSZ),PLPR,PTOP,PO,A(NEPR),A(NEPC),	00093500
	*	A(NEPZ),NS)	00093600
		CALL MBSR (IGENP,AM(MIPR),AM(MIPC),AM(MIC),AM(MIU),AM(MJC),	00093700
	*	AM(MDI),AM(MC),AM(MB),AM(MRES),AM(MAUX))	00093800
		CALL UPDATP (AM(MSIGRP),AM(MSIGCP),AM(MSIGZP),A(NSIGR),A(NSIGC),	00093900
	*	A(NSIGZ),AM(MEPRP),AM(MEPCP),AM(MEPZCP),A(NEPR),	00094000
	*	A(NEPC),A(NEPZ),AM(MTPRP),AM(MTPCP),AM(MTPZP),	00094100
	*	A(NTPR),A(NTPC),A(NTPZ),AM(MDSRP),AM(MDSCP),	00094200
	*	AM(MDSZP),A(NTSR),A(NTSC),A(NTSZ),AM(MDISPP),	00094300
	*	A(NRDIS),NS)	00094400
C			00094500
		IF(JOUT.EQ.1) GO TO 640	00094600
		IF (JKL.NE.1) GO TO 650	00094700
		IF(JJOUT.LE.0) GO TO 650	00094800
C			00094900
	640	CONTINUE	00095000
		CALL EDTMTX (A(NEPR),A(NEPC),A(NEPZ),A(NSIGR),A(NSIGC),A(NSIGZ),	00095100
	*	A(NTPR),A(NTPC),A(NTPZ),A(NTSR),A(NTSC),A(NTSZ),	00095200
	*	A(NRDIS),A(NCRIF),A(NRCP),A(NTCP),A(NTBCP),	00095300
	*	A(NATBCP),A(NRCL),A(NTCL),A(NTBCL),A(NATBCL),	00095400
	*	A(NYNG),A(NPOS),A(NSHR),A(NBM),A(NRFR),A(NFR),	00095500
	*	A(NTBF),A(NATBF),A(NPRS),A(NEHP),AM(MTPRF),	00095600
	*	AM(MTPRCP),AM(MTPCF),AM(MTPCCP),AM(MTPZF),	00095700
	*	AM(MTPZCP),	00095800
	*	AM(MPRESP),	00095900
	*	A(NKFRIC),AM(MTPRP),AM(MTPCP),AM(MTPZP),AM(MDSRP),	00096000
	*	AM(MDSCP),AM(MDSZP),A(NPOSB),A(NSHRB),A(NSWG),	00096100
	*	A(NSWS),NS,A(NRFRPM),A(NFRPM),A(NGS),A(NCPU),	00096200
	*	A(NLM),A(NLMR),A(NLMC),A(NLMZ),A(NRCOD))	00096300
		IF(JOUT.LE.0) JJOUT=0	00096400
C			00096500
	650	CONTINUE	00096600
C			00096700
		IF (NPLT.EQ.0) GO TO 730	00096800
		IF (ISP.EQ.0) GO TO 730	00096900
C			00097000
			00097100
C			00097200
	C====	PREPARATION FOR PLOTTER =====	00097300
C			00097400
C			00097500
		IF (ISPLOT.EQ.0) CALL PLOTS(ACTIVE,ACTIVE,77)	00097600
C		77 IN ARGUMENTS MEANS LOGICAL UNIT NUMBER FOR PLOT	00097700
		IF (ISPLOT.EQ.0) CALL PLOT(0.,20.,-3)	00097800
		ISPLOT = 1	00097900
		IF (NS.NE.IPFUL) GO TO 710	00098000
C			00098100
	C----	FUEL	00098200
C			00098300
		CALL PLOT1 (A(NPPW),A(NPRL),A(NPTL),A(NPT),A(NPTLMX),	00098400
	*	A(NPRLR),A(NPTLR),A(NPCPGP),A(NPCPU),A(NPBU),	00098500
	*	A(NPOW),A(NRFRPM),A(NFRPM),A(NRCB),A(NREB),	00098600

	*	A(NRUB),A(NRCI),A(NCPU),A(NBU),A(NRFR),A(NTFR),	00098700
	*	ITPLOT,NS)	00098800
	710	CONTINUE	00098900
		IF (NS.NE.IPSTS) GO TO 720	00099000
	C		00099100
	C----	CLAD STRESS	00099200
	C		00099300
		CALL PLOT2 (A(NPSIGC),A(NSIGC),A(NVOL),ITPLOT,NS)	00099400
	720	CONTINUE	00099500
		IF (NS.NE.IPSTN) GO TO 730	00099600
	C		00099700
	C----	CLAD STRAIN	00099800
	C		00099900
		CALL PLOT3 (A(NPCSTN),A(NPTSC),A(NPTPC),A(NRCO),A(NRCOO),	00100000
	*	A(NTSC),A(NTPC),A(NVOL),A(NRDIS),ITPLOT,NS)	00100100
	730	CONTINUE	00100200
		IF (NCRK.LE.0) GO TO 1000	00100300
		IF (NPLOT.EQ.0) GO TO 1000	00100400
		IF (ISP.EQ.0) GO TO 1000	00100500
		IF (NS.NE.IPCRK) GO TO 1000	00100600
	C		00100700
		IF (ISPLOT.EQ.0) CALL PLOTS(ACTIVE,ACTIVE,77)	00100800
		IF (ISPLOT.EQ.0) CALL PLOT(0.,20.,-3)	00100900
		ISPLOT = 1	00101000
	C		00101100
	C----	FUEL CRACK	00101200
	C		00101300
		CALL PLOT4 (A(NRFRPM),A(NRCB),A(NREB),A(NLMR),A(NLMC),	00101400
	*	A(NLMZ),IT,NS)	00101500
		IPCNT = IPCNT + 1	00101600
	C		00101700
	1000	CONTINUE	00101800
	C		00101900
	C		00102000
	C		00102100
		IF (NPLOT.NE.0 .AND. ISP.EQ.1)	00102200
	*	CALL PLOT5 (A(NPTIME),A(NPPLPR),A(NPYP),A(NPIT),ITPLOT,IT,	00102300
	*	TIM,PLPR)	00102400
	C		00102500
	C		00102600
	C====	GOING TO NEXT TIME STEP =====	00102700
	C		00102800
	C		00102900
		PPOWA = POWERC	00103000
		PPO=PO	00103100
		PTIN = TIN	00103200
		PTOUT = TOUT	00103300
		IF (NSKIP.GT.0) CALL CORTMP(A(NTOUR),A(NTOUO),A(NPTOU),4,TOUT)	00103400
		IF (KJIK.NE.0) GO TO 140	00103500
		DELTO = DELT	00103600
	C		00103700
		GO TO 140	00103800
	C		00103900
	999	CONTINUE	00104000
		IF (NPLOT.EQ.0) GO TO 998	00104100
		CALL PLTOR (A(NPPOW),A(NPRL),A(NPTL),A(NPT),A(NPTLMX),	00104200
	*	A(NPRLR),A(NPTLR),A(NPCPGP),A(NPCPU),A(NPBU),	00104300
	*	A(NPSIGC),A(NPCSTN),A(NPTSC),A(NPTPC),	00104400

*	A(NPTIME),A(NPPLPR),A(NPYP),A(NPIT),ITPLOT,IT)	00104500
998	CONTINUE	00104600
	IF (ISPLIT.NE.0) CALL PLOT (0.,0.,999)	00104700
	WRITE(106,9003) DAYDAY	00104800
	RETURN	00104900
C		00105000
C	FORMAT	00105100
C		00105200
	9000 FORMAT(16I5)	00105300
	9001 FORMAT(E10.3,F5.1,3E15.5,E10.5,2I5)	00105400
	9002 FORMAT(/1H0,' *** ACTUALLY USED CORE FOR ARRAY =' ,I8,' WORDS',	00105500
	*5X,'(INPUT =' ,I8,')' /	00105600
	* 1H0,' *** ACTUALLY USED CORE FOR MATRIX =' ,I8,' WORDS',	00105700
	*5X,'(INPUT =' ,I8,')' /	00105800
	* 1H0,' *** RECOMMENDED CORE SIZE =' ,I4,' K BYTE')	00105900
	9003 FORMAT(1H1,' ***** JOB ENDED NORMALLY ***** ACTIVE-II *****',	00106000
	* 60X,'DATE',2X,A8)	00106100
	9004 FORMAT(1H1, 9(///// , 1H0, 35X, '===== A C T I V E - I I - F00106200	
	* B R ====='), /, 1H1)	00106300
	9005 FORMAT(1H1, 9(///// , 1H0, 35X, '===== A C T I V E - I I - A00106400	
	* T R ====='), /, 1H1)	00106500
	9006 FORMAT(2I5,10X,3E10.5)	00106600
	9007 FORMAT(1H1, 9(///// , 1H0, 35X, '===== A C T I V E - I I - P00106700	
	* W R ====='), /, 1H1)	00106800
	9008 FORMAT(1H1, 9(///// , 1H0, 35X, '===== A C T I V E - I I - B00106900	
	* W R ====='), /, 1H1)	00107000
C		00107100
	END	00107200

PNCT841-75-17

	FUNCTION AREA (R,DR)	00000100
C		00000200
C	AREA CALC. BETWEEN (R+DR) AND (R)	00000300
C		00000400
	AREA = DR*(2.*R+DR)	00000500
C		00000600
	RETURN	00000700
	END	00000800

	FUNCTION ATBFUN (T,DT,I)	00000900
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00001000
	* PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00001100
	* CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00001200
C		00001300
C	FUEL THERMAL EXPANSION CALC.	00001400
C		00001500
	Y = 20.	00001600
	ATBFUN = ALF0(I)*T+ALF1(I)*T*T/2.+(ALF0(I)+ALF1(I)*T)*DT/2.-	00001700
	* (ALF0(I)*Y+ALF1(I)*Y*Y/2.)	00001800
	RETURN	00001900
	END	00002000

PNCT841-75-17

SUBROUTINE AXIAL (X0,Y0,ID)	00002100
CALL SYMBOL(X0,Y0,4.,'AXIAL SECTION =',0.,15)	00002200
CALL NUMBER(X0+64.,Y0,4.,FLOAT(ID),0.,-1)	00002300
RETURN	00002400
END	00002500

SUBROUTINE AXISD(MX,XMN,DX)	00002600
DO 100 I=1,MX	00002700
X= XMN*(MX-I)	00002800
CALL NEWPEN(3)	00002900
IF(I.EQ.1.OR.I.EQ.MX) GO TO 200	00003000
CALL PLOT (X/DX,220.,3)	00003100
CALL PLOT (X/DX,0.,2)	00003200
200 CONTINUE	00003300
CALL NEWPEN(1)	00003400
CALL NUMBER (X/DX-4.,-7.0,4.0,X,0.,1)	00003500
100 CONTINUE	00003600
RETURN	00003700
END	00003800

SUBROUTINE AXIST(MX,XMN,DX)	00003900
DO 100 I=1,MX	00004000
J=MX-I+1	00004100
X = XMN*J	00004200
CALL NEWPEN(3)	00004300
IF(I.EQ.1) GO TO 200	00004400
CALL PLOT (J/DX,220.,3)	00004500
CALL PLOT (J/DX,0.0,2)	00004600
200 CONTINUE	00004700
CALL NEWPEN(1)	00004800
CALL NUMBER (J/DX-4.0,-7.0,4.0,X,0.,-1)	00004900
100 CONTINUE	00005000
CALL NUMBER (-2.0,-7.0,4.0,0.0,0.,-1)	00005100
RETURN	00005200
END	00005300

SUBROUTINE BCALC (B,ATBF,ATBCL,TPR,TPC,TPZ,PRS,RF,RC,P,BM,G,PB,	00005400
* GB,VOL,RH,PLPR,CVPR,PO,PTOP,DELT,TSR,TSC,TSZ,	00005500
* FLUX,TBCL,KFRIC,SWG,SWS,EHP,EPR,EPC,EPZ,	00005600
* MOUNT,IFRIC,NS,RB)	00005700
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00005800
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00005900
* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00006000
* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00006100
* N12C,N13C,N14C,N15C,	00006200
* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00006300
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00006400
* PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,	00006500
* CPOW,BUCNS,RK,RKR,A91,GAS1,TCON,PDBU,BUB	00006600
C	00006700
DIMENSION B(IGEN) , ATBF(NN,NFR) , ATBCL(NN,NCL) , TPR(NN,NTR) ,	00006800
* TPC(NN,NTR) , TPZ(NN,NTR) , PRS(NN,NTR) , RF(NN,NFR1) ,	00006900
* RC(NN,NCL1) , P(NN,NTR) , BM(NN,NTR) , G(NN,NTR) ,	00007000
* PB(NN,2) , GB(NN,2) , VOL(NN,NTR) , RH(NN,NTR) ,	00007100
* TSR(NN,NTR) , TSC(NN,NTR) , TSZ(NN,NTR) , FLUX(NN) ,	00007200
* TBCL(NN,NCL) , KFRIC(NN) , SWG(NN,NFR) ,	00007300
* SWS(NN,NTR) , EHP(NN,NTR) , EPR(NN,NTR) , EPC(NN,NTR) ,	00007400
* EPZ(NN,NTR) , RB(NN,NTR)	00007500
C	00007600
C RIGHT-HAND VECTORS B OF AX=B	00007700
C	00007800
IF (MOUNT) 10,10,3000	00007900
C	00008000
10 CONTINUE	00008100
C	00008200
DO 999 I=1,IGEN	00008300
999 B(I) = 0.	00008400
C	00008500
DO 1000 I=1,NFR	00008600
B(I) = ATBF(NS,I) + TSR(NS,I)	00008700
1000 CONTINUE	00008800
C	00008900
DO 1100 I=1,NCL	00009000
B(NFR+I) = ATBCL(NS,I) + TSR(NS,NFR+I)	00009100
1100 CONTINUE	00009200
C	00009300
DO 1200 I=1,NFR	00009400
B(N1+I) = ATBF(NS,I) + TSC(NS,I)	00009500
1200 CONTINUE	00009600
C	00009700
DO 1300 I=1,NCL	00009800
B(N1+NFR+I) = ATBCL(NS,I) + TSC(NS,NFR+I)	00009900
1300 CONTINUE	00010000
C	00010100
DO 1400 I=1,N1	00010200
B(N2+I) = -TPR(NS,I)	00010300
1400 CONTINUE	00010400
C	00010500
DO 1500 I=1,N1	00010600
B(N3+I) = -TPC(NS,I)	00010700
1500 CONTINUE	00010800
C	00010900
DO 1600 I=1,N1	00011000
B(N4+I) = -TPZ(NS,I)	00011100

1600 CONTINUE	00011200
C	00011300
C	00011400
DO 2000 I=1,NFR	00011500
RI = RF(NS,I)	00011600
RO = RF(NS,I+1)	00011700
RR = (RO+RI)*(RO-RI)	00011800
C	00011900
RA = RB(NS,I)	00012000
RA = RI	00012100
XY1 = (1.-2.*P(NS,I))*RA*ALOG(RA/RI) / (2.*(1.-P(NS,I)))	00012200
XY2 = (RI*RI+(1.-2.*P(NS,I))*RA*RA)*RO*RO*ALOG(RO/RI) /	00012300
* (2.*(1.-P(NS,I))*RA*RR)	00012400
OM = 1.-2.*RO*RO*ALOG(RO/RI) / RR	00012500
B(NS+6,I) = RA*(1.+P(NS,I))*ATBF(NS,I) + EPR(NS,I)*(-RI+RA/2.-	00012600
* XY1+XY2) + EPC(NS,I)*(-RA/2.+XY1-XY2)	00012700
* + TSR(NS,I)*(0.5*RA+XY1-XY2)	00012800
* + TSC(NS,I)*(0.5*RA-XY1+XY2)	00012900
* + TSZ(NS,I)*RA*P(NS,I)	00013000
2000 CONTINUE	00013100
C	00013200
DO 2100 I=1,NCL	00013300
RI = RC(NS,I)	00013400
RO = RC(NS,I+1)	00013500
RR = (RO+RI)*(RO-RI)	00013600
C	00013700
RA = RB(NS,NFR+I)	00013800
RA = RI	00013900
XY1 = (1.-2.*P(NS,NFR+I))*RA*ALOG(RA/RI) / (2.*(1.-P(NS,NFR+I)))	00014000
XY2 = (RI*RI+(1.-2.*P(NS,NFR+I))*RA*RA)*RO*RO*ALOG(RO/RI) /	00014100
* (2.*(1.-P(NS,NFR+I))*RA*RR)	00014200
OM = 1.-2.*RO*RO*ALOG(RO/RI) / RR	00014300
B(NS+6+NFR+I) = RA*(1.+P(NS,NFR+I))*ATBCL(NS,I) +	00014400
* EPR(NS,NFR+I)*(-RI+RA/2.-XY1+XY2)+EPC(NS,NFR+I)*(-RA/2.+XY1-XY2)	00014500
* + TSR(NS,NFR+I)*(0.5*RA+XY1-XY2)	00014600
* + TSC(NS,NFR+I)*(0.5*RA-XY1+XY2)	00014700
* + TSZ(NS,NFR+I)*RA*P(NS,NFR+I)	00014800
2100 CONTINUE	00014900
DO 2200 I=1,NFR	00015000
RI = RF(NS,I)	00015100
RO = RF(NS,I+1)	00015200
RR = (RO+RI)*(RO-RI)	00015300
C	00015400
RA = RB(NS,I)	00015500
RA = RO	00015600
XY1 = (1.-2.*P(NS,I))*RA*ALOG(RA/RI) / (2.*(1.-P(NS,I)))	00015700
XY2 = (RI*RI+(1.-2.*P(NS,I))*RA*RA)*RO*RO*ALOG(RO/RI) /	00015800
* (2.*(1.-P(NS,I))*RA*RR)	00015900
TH = 1.-2.*RI*RI*ALOG(RO/RI) / RR	00016000
B(NS+6,I) = RA*(1.+P(NS,I))*ATBF(NS,I) + EPR(NS,I)*(-RO+RA/2.-	00016100
* XY1+XY2) + EPC(NS,I)*(-RA/2.+XY1-XY2)	00016200
* + TSR(NS,I)*(0.5*RA+XY1-XY2)	00016300
* + TSC(NS,I)*(0.5*RA-XY1+XY2)	00016400
* + TSZ(NS,I)*RA*P(NS,I)	00016500
2200 CONTINUE	00016600
C	00016700
DO 2300 I=1,NCL	00016800
RI = RC(NS,I)	00016900
RO = RC(NS,I+1)	00017000
RR = (RO+RI)*(RO-RI)	00017100
C	00017200
RA = RB(NS,NFR+I)	00017300

RA = RO	00017000
XY1 = (1.-2.*P(NS,NFR+I))*RA*ALOG(RA/RI) / (2.*(1.-P(NS,NFR+I)))	00017100
XY2 = (RI*RI+(1.-2.*P(NS,NFR+I))*RA*RA)*RO*RO*ALOG(RO/RI) /	00017200
* (2.*(1.-P(NS,NFR+I))*RA*RR)	00017300
TH = 1.-2.*RI*RI*ALOG(RO/RI) / RR	00017400
B(N6+6+NFR+I) = RA*(1.+P(NS,NFR+I))*ATBCL(NS,I) + EPR(NS,NFR+I)*	00017500
* (-RO+RA/2.-XY1+XY2) + EPC(NS,NFR+I)*(-RA/2.+XY1-XY2)	00017600
* + TSR(NS,NFR+I)*(0.5*RA+XY1-XY2)	00017700
* + TSC(NS,NFR+I)*(0.5*RA-XY1+XY2)	00017800
* + TSZ(NS,NFR+I)*RA*P(NS,NFR+I)	00017900
2300 CONTINUE	00018000
C	00018100
B(N7+7) = CVPR	00018200
C	00018300
B(N7+8) = PO	00018400
C	00018500
SUMMA = 0.	00018600
SUMMB = 0.	00018700
DO 2400 N=NS,NNF	00018800
DO 2400 I=1,NFR	00018900
2400 SUMMA = SUMMA + RH(N,I)*VOL(N,I)*GRAV	00019000
C	00019100
DO 2450 I=1,NFR	00019200
2450 SUMMB = SUMMB + RH(NS,I)*VOL(NS,I)*GRAV / 2.0	00019300
SUMMAS = SUMMA-SUMMB	00019400
RO = RF(NS,NFR1)	00019500
RI = RF(NS,1)	00019600
RR = (RO+RI) * (RO-RI)	00019700
XG = GB(NS,1)	00019800
XP = PB(NS,1)	00019900
C	00020000
SUMRAT = 0.	00020100
SUMEPR = 0.	00020200
SUMTSZ = 0.	00020300
DO 2500 I=1,NFR	00020400
SUMRAT = SUMRAT + (RF(NS,I+1)+RF(NS,I))*(RF(NS,I+1)-RF(NS,I))*	00020500
* ATBF(NS,I)	00020600
SUMEPR = SUMEPR + (RF(NS,I+1)+RF(NS,I))*(RF(NS,I+1)-RF(NS,I))*	00020700
* EPR(NS,I)	00020800
SUMTSZ = SUMTSZ + (RF(NS,I+1)+RF(NS,I))*(RF(NS,I+1)-RF(NS,I))*	00020900
* TSZ(NS,I)	00021000
2500 CONTINUE	00021100
C	00021200
C	00021300
B(N7+9) = -(PLPR+SUMMAS/(PI*RR))/(2.*XG) + (1.+XP)*SUMRAT/	00021400
* RR - XP*RI*RI*CVPR/(XG*RR)	00021500
* + SUMTSZ*(1.+XP) / RR	00021600
C	00021700
SUMMA = 0.	00021800
SUMMB = 0.	00021900
C	00022000
DO 2600 N=NS,NN	00022100
DO 2600 I=1,NCL	00022200
2600 SUMMA = SUMMA + RH(N,NFR+I)*VOL(N,NFR+I)*GRAV	00022300
C	00022400
DO 2650 I=1,NCL	00022500
2650 SUMMB = SUMMB + RH(NS,NFR+I)*VOL(NS,NFR+I)*GRAV/2.0	00022600
C	00022700

	SUMMAS = SUMMA-SUMMB	00022800
C		00022900
	SUMRAT = 0.	00023000
	SUMEPR = 0.	00023100
	SUMTSZ = 0.	00023200
	DO 2700 I=1,NCL	00023300
	SUMRAT = SUMRAT + (RC(NS,I+1)**2-RC(NS,I)**2)*ATBCL(NS,I)	00023400
	SUMTSZ = SUMTSZ + (RC(NS,I+1)**2-RC(NS,I)**2)*TSZ(NS,NFR+I)	00023500
2700	SUMEPR = SUMEPR + (RC(NS,I+1)**2-RC(NS,I)**2)*EPR(NS,NFR+I)	00023600
C		00023700
	RO = RC(NS,NCL1)	00023800
	RI = RC(NS,1)	00023900
	RR = (RO+RI) * (RO-RI)	00024000
	XP = PB(NS,2)	00024100
	XG = GB(NS,2)	00024200
C		00024300
	B(N7+10) = (-PTOP-PI*(RO**2*PO-RI**2*PLPR)-SUMMAS)/(2.*PI*XG*RR)+00024400	
*	(1.+XP)*SUMRAT/RR + XP*RO*RO*PO / (XG*RR)	00024500
*	+ SUMTSZ*(1.+XP) / RR	00024600
C		00024700
	J = 1	00024800
	DO 2800 I=1,N1	00024900
	IF(I-NFR) 500,500,600	00025000
500	RO = RF(NS,I+1)	00025100
	RI = RF(NS,I)	00025200
	ATB = ATBF(NS,I)	00025300
	GO TO 700	00025400
600	RO = RC(NS,J+1)	00025500
	RI = RC(NS,J)	00025600
	ATB = ATBCL(NS,J)	00025700
	J = J+1	00025800
700	CONTINUE	00025900
	XP = P(NS,I)	00026000
	XG = G(NS,I)	00026100
	RR = (RO+RI) * (RO-RI)	00026200
	TH = 1.-2.*RI*RI*ALOG(RO/RI)/RR	00026300
	OM = 1.-2.*RO*RO*ALOG(RO/RI) / RR	00026400
	B(N7+10+I) =	00026500
*	XG*(EPR(NS,I)-EPC(NS,I)) * (TH+OM-TH*OM) / (2.*(1.-XP))	00026600
*	- XG*(OM+TH-TH*OM) / (2.*(1.-XP)) * TSR(NS,I)	00026700
*	- XG*(TH*OM- TH-OM) / (2.*(1.-XP)) * TSC(NS,I)	00026800
	B(N8+10+I) =	00026900
*	XG*(EPR(NS,I)-EPC(NS,I)) * (TH*OM-TH-OM) / (2.*(1.-XP))	00027000
*	- XG*(TH*OM-TH-OM) / (2.*(1.-XP)) * TSR(NS,I)	00027100
*	- XG*(OM+ TH-TH*OM) / (2.*(1.-XP)) * TSC(NS,I)	00027200
	B(N9+10+I) = -2.*XG*(1.+XP)*ATB	00027300
*	- 2.*XG*(1.+XP) * TSZ(NS,I)	00027400
2800	CONTINUE	00027500
C		00027600
C		00027700
C	VARIABLE CONDITIONS GAP OPEN/STICK/SLIP	00027800
C		00027900
3000	CONTINUE	00028000
C		00028100
	IF(IFRIC-1) 800,900,920	00028200
C		00028300
800	CONTINUE	00028400
C		00028500

C	OPEN	00028600
C		00028700
	B(N10+11) = PLPR	00028800
	B(N10+12) = 0.	00028900
C		00029000
	GO TO 990	00029100
C		00029200
	900 CONTINUE	00029300
C		00029400
C	STICK OR SLIP	00029500
C		00029600
	B(N10+11) = EPZ(NS,1)-EPZ(NS,NTR)	00029700
	GO TO 930	00029800
C		00029900
	920 CONTINUE	00030000
	B(N10+11) = 0.	00030100
C		00030200
	930 CONTINUE	00030300
	B(N10+12) = RC(NS,1)-RF(NS,NFR1)	00030400
C		00030500
	990 CONTINUE	00030600
C		00030700
	RETURN	00030800
	END	00030900

	SUBROUTINE BCALCP (B,ATBCL,TPR,TPC,TPZ,RC,P,G,PB,GB,VOL,RH,TBCL, 00031000
*	FLUX,KFRIC,CRIF,DELT,TSR,TSC,TSZ,PLPR,PTOP, 00031100
*	PO,EPR,EPC,EPZ,NS) 00031200
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK, 00031300
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1, 00031400
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP, 00031500
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C, 00031600
*	N12C,N13C,N14C,N15C, 00031700
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15 00031800
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2), 00031900
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX, 00032000
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB 00032100
	DIMENSION B(IGENP), ATBCL(NN,NCL), TPR(NN,NTR), TPC(NN,NTR), 00032200
*	TPZ(NN,NTR), RC(NN,NCL1), P(NN,NTR), G(NN,NTR), 00032300
*	PB(NN,2), GB(NN,2), VOL(NN,NTR), RH(NN,NTR), 00032400
*	TBCL(NN,NCL), FLUX(NN), KFRIC(NN), CRIF(NN), 00032500
*	TSR(NN,NTR), TSC(NN,NTR), TSZ(NN,NTR), EPR(NN,NTR), 00032600
*	EPC(NN,NTR), EPZ(NN,NTR) 00032700
C	00032800
C	00032900
C	RIGHT-HAND VECTORS B OF AX=B 00033000
C	00033100
C	FOR PLENUM SECTION ONLY 00033200
C	00033300
C	00033400
	DO 999 I=1,IGENP 00033500
	999 B(I) = 0. 00033600
C	00033700
	DO 1000 I=1,NCL 00033800
	B(I) = ATBCL(NS,I) + TSR(NS,NFR+I) 00033900
	1000 CONTINUE 00034000
C	00034100
	DO 1100 I=1,NCL 00034200
	B(N1C+I) = ATBCL(NS,I) + TSC(NS,NFR+I) 00034300
	1100 CONTINUE 00034400
C	00034500
	DO 1200 I=1,NCL 00034600
	B(N2C+I) = -TPR(NS,NFR+I) 00034700
	1200 CONTINUE 00034800
C	00034900
	DO 1300 I=1,NCL 00035000
	B(N3C+I) = -TPC(NS,NFR+I) 00035100
	1300 CONTINUE 00035200
C	00035300
	DO 1400 I=1,NCL 00035400
	B(N4C+I) = -TPZ(NS,NFR+I) 00035500
	1400 CONTINUE 00035600
C	00035700
	DO 1700 I=1,NCL 00035800
	XP = P(NS,NFR+I) 00035900
	RI = RC(NS,I) 00036000
	RO = RC(NS,I+1) 00036100
	RR = (RO+RI)*(RO-RI) 00036200
	OM = 1.-2.*RO*RO*ALOG(RO/RI) / RR 00036300
	B(N5C+3+I) = RC(NS,I)*(1.+P(NS,NFR+I))*ATBCL(NS,I) 00036400
	4 -RI*(OM*(EPR(NS,NFR+I)-EPC(NS,NFR+I)) / 2. + EPC(NS,NFR+I)) 00036500
*	- (-0.5*RI+RI*RO*RO*ALOG(RO/RI) / RR) * TSR(NS,NFR+I) 00036600
*	- (-0.5*RI-RI*RO*RO*ALOG(RO/RI) / RR) * TSC(NS,NFR+I) 00036700

* + RI*XP*TSZ(NS,NFR+I)		00036800
1700 CONTINUE		00036900
C		00037000
DO 1800 I=1,NCL		00037100
XP = P(NS,NFR+I)		00037200
RI = RC(NS,I)		00037300
RO = RC(NS,I+1)		00037400
RR = (RO+RI)*(RO-RI)		00037500
TH = 1.-2.*RI*RI*ALOG(RO/RI) / RR		00037600
B(N6C+3+I) = RC(NS,I+1)*(1.+P(NS,NFR+I))*ATBCL(NS,I)		00037700
* -RO*(TH*(EPR(NS,NFR+I)-EPC(NS,NFR+I))/2. + EPC(NS,NFR+I))		00037800
* - (-0.5*RO+RO*RI*RI*ALOG(RO/RI) / RR) * TSR(NS,NFR+I)		00037900
* - (-0.5*RO-RO*RI*RI*ALOG(RO/RI) / RR) * TSC(NS,NFR+I)		00038000
* + RO*XP*TSZ(NS,NFR+I)		00038100
1800 CONTINUE		00038200
C		00038300
B(N7C+4) = PLPR		00038400
C		00038500
B(N7C+5) = PO		00038600
C		00038700
CLAD OF PLENUM SECTION AXIAL DISPLACEMENT		00038800
SUMRAT = 0.		00038900
SUMMAS = 0.		00039000
SUMEPR = 0.		00039100
SUMTSZ = 0.		00039200
DO 1850 I=1,NCL		00039300
SUMMAS = SUMMAS + RH(NS,NFR+I)*VOL(NS,NFR+I)*GRAV / 2.0		00039400
SUMRAT = SUMRAT + (RC(NS,I+1)**2-RC(NS,I)**2)*ATBCL(NS,I)		00039500
SUMEPR = SUMEPR + (RC(NS,I+1)**2-RC(NS,I)**2)*EPR(NS,NFR+I)		00039600
SUMTSZ = SUMTSZ + (RC(NS,I+1)**2-RC(NS,I)**2)*TSZ(NS,NFR+I)		00039700
1850 CONTINUE		00039800
C		00039900
RO = RC(NS,NCL)		00040000
RI = RC(NS,1)		00040100
RR = (RO+RI) * (RO-RI)		00040200
XP = PB(NS,2)		00040300
XG = GB(NS,2)		00040400
C		00040500
B(N7C+6) = (-PTOP-PI*(RO**2*PO-RI**2*PLPR)-SUMMAS)/(2.*PI*XG*RR)+		00040600
* (1.+XP)*SUMRAT/RR + XP*RO*RO*PO / (XG*RR)		00040700
* + SUMTSZ*(1.+XP)/RR		00040800
C		00040900
C		00041000
C	STRESS	00041100
C		00041200
C		00041300
DO 1900 I=1,NCL		00041400
RO = RC(NS,I+1)		00041500
RI = RC(NS,I)		00041600
XP = P(NS,NFR+I)		00041700
XG = G(NS,NFR+I)		00041800
RR = (RO+RI) * (RO-RI)		00041900
ATB = ATBCL(NS,I)		00042000
TH = 1.-2.*RI*RI*ALOG(RO/RI) / RR		00042100
OM = 1.-2.*RO*RO*ALOG(RO/RI) / RR		00042200
C		00042300
B(N7C+6+I) =		00042400
* XG*(EPR(NS,NFR+I)-EPC(NS,NFR+I))*(TH+OM-TH*OM)/(2.*(1.-XP))		00042500

*	- XG*(OM+TH-TH*OM) / (2.*(1.-XP)) * TSR(NS,NFR+I)	00042600
*	- XG*(TH*OM- TH-OM) / (2.*(1.-XP)) * TSC(NS,NFR+I)	00042700
	B(N8C+6+I) =	00042800
*	XG*(EPR(NS,NFR+I)-EPC(NS,NFR+I))*(TH*OM-TH-OM)/(2.*(1.-XP))	00042900
*	- XG*(TH*OM-TH-OM) / (2.*(1.-XP)) * TSR(NS,NFR+I)	00043000
*	- XG*(OM+ TH-TH*OM) / (2.*(1.-XP)) * TSC(NS,NFR+I)	00043100
	B(N9C+6+I) = -2.*XG*(1.+XP)*ATB	00043200
*	- 2.*XG*(1.+XP) * TSZ(NS,NFR+I)	00043300
	1900 CONTINUE	00043400
C		00043500
	B(N10C+7) = 0.	00043600
C		00043700
C		00043800
C	FRICTION FORCE	00043900
C		00044000
C		00044100
C		00044200
	RETURN	00044300
	END	00044400

SUBROUTINE BEGIN (XM,ZFLX,RH,RHOP,CPU,LM,EHP,SWS,GS,GR,	00044500
* RVB,RCB,REB,RUB,QSF,QSFF,QSFP,POW,XCPI,XCPD,	00044600
4 BU,HG,KFRIC,FLZO,FLZ,PRS,FLNC,CLZO,CLZ,	00044700
* FL,CL,DXM,SWG,RCI,RCO,RVBO,RCBO,REBO,RUBO,	00044800
* RCIO,RCOO,XMINT,RDIST,EPSEQ,SWS3,LMR,LMC,LMZ)	00044900
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00045000
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00045100
* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00045200
* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00045300
* N12C,N13C,N14C,N15C,	00045400
* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00045500
COMMON /OTHERS/ PG(3),CMI(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00045600
* PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,	00045700
* CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00045800
COMMON /CONST/ FKA1,FKB1,FKC1,FKA2,FKB2,FKC2,FKA3,FKB3,FKC3,	00045900
* FA11,FA12,FA21,FA22,FA31,FA32,	00046000
* CK1,CK2,CA1,CA2,XEWTON(7)	00046100
COMMON /CAPSUL/ RCPI,RCPO,AFNA	00046200
DIMENSION XM(NN,3),ZFLX(NN),RH(NN,NTR),RHOP(NN,3),	00046300
* EHP(NN,NTR),SWS(NN,NTR),GS(NN,NFR),RVB(NN),	00046400
* RCB(NN),REB(NN),RUB(NN),QSF(NN),QSFF(NN),	00046500
* QSFP(NN),POW(NN),XCPI(NN),XCPD(NN),BU(NN),	00046600
* HG(NN),KFRIC(NN),FLZO(NN),FLZ(NN),	00046700
* SWG(NN,NFR),PRS(NN,NTR),FLNC(NN),CLZO(NN),	00046800
* CPU(NN,NP),LM(NN,NTR),CLZ(NN),GR(NN),DXM(NN,3),	00046900
* RCI(NN),RCO(NN),RVBO(NN),RCBO(NN),REBO(NN),	00047000
* RUBO(NN),RCIO(NN),RCOO(NN),XMINT(NN),	00047100
* RDIST(NN,4),EPSEQ(NN,NTR),SWS3(NN,NUR),	00047200
* LMR(NN,NTR),LMC(NN,NTR),LMZ(NN,NTR)	00047300
C	00047400
C	00047500
C INITIAL VALUES SET	00047600
C	00047700
C	00047800
FL = 0.0	00047900
CL = 0.0	00048000
DO 1000 N=1,NN	00048100
DO 3000 I=1,4	00048200
3000 RDIST(N,I)=0.	00048300
C	00048400
IF (N.EQ.NN) GO TO 100	00048500
C	00048600
ZFLX(N) = 1.053E-22/0.85	00048700
FLZO(N) = CLZO(N)	00048800
FLZ(N) = FLZO(N)	00048900
CLZ(N) = CLZO(N)	00049000
FL = FL+FLZO(N)	00049100
CL = CL+CLZO(N)	00049200
DXM(N,1) = 0.	00049300
DXM(N,2) = 0.	00049400
DXM(N,3) = 0.	00049500
KFRIC(N) = 0	00049600
DO 1010 K=1,NCR	00049700
RH(N,K)=RHOP(N,1)	00049800
1010 CONTINUE	00049900
XM(N,1) = PI *RHOP(N,1) * (RCB(N)-RVB(N)) * (RCB(N)+RVB(N))	00050000
KK=NCR+1	00050100
KKK=NCR+NER	00050200

DO 1020 K=KK, KKK	00050300
RH(N,K)=RHOP(N,2)	00050400
1020 CONTINUE	00050500
XM(N,2) = PI*RHOP(N,2)*(REB(N)-RCB(N))*(REB(N)+RCB(N))	00050600
KK = NCR+NER+1	00050700
KKK=NFR	00050800
DO 1030 K=KK, KKK	00050900
RH(N,K)=RHOP(N,3)	00051000
1030 CONTINUE	00051100
XM(N,3) = PI*RHOP(N,3)*(RUB(N)-REB(N))*(RUB(N)+REB(N))	00051200
C	00051300
100 CONTINUE	00051400
C	00051500
KK = NFR+1	00051600
KKK = NTR	00051700
DO 1035 K=KK, KKK	00051800
RH(N,K) = RHOC	00051900
1035 CONTINUE	00052000
C	00052100
RVBO(N) = RVB(N)	00052200
RCBO(N) = RCB(N)	00052300
REBO(N) = REB(N)	00052400
RUBO(N) = RUB(N)	00052500
RCIO(N) = RCI(N)	00052600
RCOO(N) = RCO(N)	00052700
C	00052800
DO 1040 K=1, NFR	00052900
SWG(N,K) = 0.	00053000
SWS(N,K) = 0.0	00053100
1040 GS(N,K) = 0.0	00053200
C	00053300
DO 1050 K=1, NUR	00053400
1050 SWS3(N,K) = 0.	00053500
DO 1000 K=1, NTR	00053600
LM(N,K) = 0	00053700
LMR(N,K) = 0	00053800
LMC(N,K) = 0	00053900
LMZ(N,K) = 0	00054000
EHP(N,K) = 0.	00054100
PRS(N,K) = 0.	00054200
EPSEQ(N,K) = 0.	00054300
1000 CONTINUE	00054400
C	00054500
DO 1400 I=1, NFR	00054600
1400 RH(NN,I) = 0.	00054700
DO 1500 I=1, NCL	00054800
1500 RH(NN,NFR+I) = RHOC	00054900
C	00055000
RHOP(NN,1) = 0.	00055100
RHOP(NN,2) = 0.	00055200
RHOP(NN,3) = 0.	00055300
GR(NN) = 0.	00055400
FLZO(NN) = 0.0	00055500
FLZ(NN) = FLZO(NN)	00055600
CLZ(NN) = CLZO(NN)	00055700
FL = FL+FLZO(NN)	00055800
CL = CL+CLZO(NN)	00055900
C	00056000

ALF0(1) = FA11	00056100
ALF1(1) = FA12	00056200
ALF0(2) = FA21	00056300
ALF1(2) = FA22	00056400
ALF0(3) = FA31	00056500
ALF1(3) = FA32	00056600
QSF(NN) = 0.	00056700
POW(NN) = 0.	00056800
SMQ = 0.	00056900
X = 0.	00057000
DO 2000 N=1,NNF	00057100
SM = 0.	00057200
DO 2010 K=1,3	00057300
2010 SM = SM + XM(N,K)	00057400
XMINT(N) = SM	00057500
QSF(N) = SM * QSFP(N)	00057600
SMQ = SMQ + QSF(N)	00057700
2000 X = X + QSFF(N)	00057800
DO 2020 N=1,NN	00057900
XCPO(N) = RCPO	00058000
XCPI(N) = RCPI	00058100
FLNC(N) = 0.	00058200
BU(N) = 0.0	00058300
HG(N) = 4.0	00058400
QSF(N) = QSF(N)*NNF / SMQ	00058500
2020 QSFF(N) = QSFF(N)*NNF / X	00058600
SMQ0 = SMQ	00058700
C	00058800
C===== INTEGRATION BY NEWTON-COTES-6	00058900
C	00059000
XEWTON(1) = 41./140.	00059100
XEWTON(2) = 216./140.	00059200
XEWTON(3) = 27./140.	00059300
XEWTON(4) = 272./140.	00059400
XEWTON(5) = 27./140.	00059500
XEWTON(6) = 216./140.	00059600
XEWTON(7) = 41./140.	00059700
C	00059800
RETURN	00059900
END	00060000

	SUBROUTINE BURN (BU,POW,XM,EPZ,QSF,QSFP,DELT)	00060100
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00060200
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00060300
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00060400
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00060500
*	N12C,N13C,N14C,N15C,	00060600
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00060700
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00060800
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00060900
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00061000
	DIMENSION BU(NN) , POW(NN) , XM(NN,3) , EPZ(NN,NTR) ,	00061100
*	QSF(NN) , QSFP(NN)	00061200
C		00061300
C		00061400
C	BURN UP CALCULATION	00061500
C		00061600
C		00061700
	SMQ = 0.	00061800
	BUB = 0.	00061900
C		00062000
	DO 1000 N=1,NNF	00062100
	SM = XM(N,1)+XM(N,2)+XM(N,3)	00062200
	BU(N) = BU(N) + BUCNS*POW(N)*DELT/SM	00062300
	BUB = BUB + BU(N)	00062400
	QSF(N) = SM*QSFP(N)*(1.-PDBU*BU(N))	00062500
	1000 SMQ = SMQ + QSF(N)	00062600
C		00062700
	BUB = BUB/NNF	00062800
C		00062900
C		00063000
	RETURN	00063100
	END	00063200

	SUBROUTINE CHECK (DEQP,EF,EC,ISW)	00115900
	DIMENSION DEQP(NN,NTR)	00116000
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00116100
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00116200
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00116300
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00116400
*	N12C,N13C,N14C,N15C,	00116500
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00116600
C		00116700
C	CREEP STRAIN INCREMENT CHECK	00116800
C		00116900
	ISW = 0	00117000
	DO 100 N=1,NN	00117100
	IF (N.EQ.NN) GO TO 250	00117200
	DO 200 J=1,NFR	00117300
	IF (DEQP(N,J).GT.EF) GO TO 400	00117400
	200 CONTINUE	00117500
C		00117600
	250 CONTINUE	00117700
	DO 300 J=1,NCL	00117800
	IF (DEQP(N,NFR+J).GT.EC) GO TO 400	00117900
	300 CONTINUE	00118000
C		00118100
	100 CONTINUE	00118200
	RETURN	00118300
C		00118400
	400 CONTINUE	00118500
	ISW = 1	00118600
	RETURN	00118700
	END	00118800

	SUBROUTINE CHKSPD (DXM, EPS, NN, ISW, DELT)	00118900
	DIMENSION DXM(NN,3)	00119000
C		00119100
C	RESTRUCTURING VELOCITY CHECK	00119200
C		00119300
	ISW = 0	00119400
	DO 100 N=1,NN	00119500
	IF (N.EQ.NN) GO TO 100	00119600
	DO 100 J=2,3	00119700
	X = DXM(N,J)*DELT*3600.0	00119800
	IF (X.GT.EPS) GO TO 200	00119900
100	CONTINUE	00120000
	RETURN	00120100
200	CONTINUE	00120200
	ISW = 1	00120300
	RETURN	00120400
	END	00120500

PNCT841-75-17

	SUBROUTINE CLEAR(IFRIC,JFRIC,NS)	00063300
	DIMENSION JFRIC(1)	00063400
C		00063500
C	ASSUME , GAP IS OPEN	00063600
C		00063700
C	IFRIC = JFRIC(NS)	00063800
	IFRIC = 0	00063900
	RETURN	00064000
	END	00064100

	SUBROUTINE CLTMP (POW,TO,RO,RI,R,T,TB,TBA,ATB,ATBA,RCLO,IT,RB)	00064200
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00064300
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00064400
*	IGEN,IGEN1,NONZR,	00064500
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00064600
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00064700
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00064800
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00064900
	COMMON /CONST/ FKA1,FKB1,FKC1,FKA2,FKB2,FKC2,FKA3,FKB3,FKC3,	00065000
*	FA11,FA12,FA21,FA22,FA31,FA32,	00065100
*	CK1,CK2,CA1,CA2,A(7)	00065200
	DIMENSION POW(NN) , TO(NN) , RO(NN) , RI(NN) , R(NN,NCL1) ,	00065300
*	T(NN,NCL1) , TB(NN,NCL) , TBA(NN) , ATB(NN,NCL) ,	00065400
*	ATBA(NN) , RCLO(NN,NCL1) , RB(NN,NTR)	00065500
C		00065600
	DIMENSION RCL(7) , TCL(7) , ATCL(7)	00065700
C		00065800
C		00065900
C	CLADDING/CAPSULE TEMPERATURE DISTRIBUTION CALC.	00066000
C		00066100
C		00066200
	Y = 20.0	00066300
	DO 1000 N=1,NN	00066400
	P = POW(N) / PI	00066500
	DR = (RO(N)-RI(N)) / NCL	00066600
	DO 1010 K=1,NCL1	00066700
1010	R(N,K) = RI(N) + DR*(K-1)	00066800
	IF(IT.NE.1) GO TO 1016	00066900
	DO 1015 K=1,NCL1	00067000
1015	RCLO(N,K) = R(N,K)	00067100
1016	CONTINUE	00067200
C		00067300
	DO 1017 K=1,NCL	00067400
	RB(N,NFR+K) = 2.*(R(N,K+1)*R(N,K+1)+R(N,K+1)*R(N,K)+R(N,K)*R(N,K))	00067500
*	/ (3.*(R(N,K+1)+R(N,K)))	00067600
1017	CONTINUE	00067700
C		00067800
	TDF = TO(N)	00067900
	AK = (CK1+CK2*TDF)	00068000
	AK2 = AK*AK	00068100
	DRR = (RO(N)-RI(N)) / 6.	00068200
	S = 0.	00068300
	S1 = 0.	00068400
	DO 1020 L=1,7	00068500
	RR = RI(N) + DRR*(L-1)	00068600
	RCL(L) = RR	00068700
	XL = ALOG(RO(N) / RR)	00068800
	TT = TDF + P*XL / (AK + SQRT(AK2+CK2*P*XL))	00068900
	TCL(L) = TT	00069000
C	TCL NOW IN DEG C	00069100
	ATCL(L) = CA1*TT + CA2*TT**2/2. - CA1*Y - CA2*Y**2/2.	00069200
	S1 = S1 + A(L)*RR*ATCL(L)	00069300
	S = S + A(L)*RR*TCL(L)	00069400
C		00069500
1020	CONTINUE	00069600
C		00069700
C		00069800
	IF (N.EQ.NN) GO TO 90	00069900

TBA(N) = S / (RO(N)+RI(N)) / 3.	00070000
GO TO 95	00070100
90 TBA(N) = TO(N)	00070200
95 CONTINUE	00070300
ATBA(N) = S1 / (RO(N)+RI(N)) / 3.	00070400
C	00070500
IF(NCL.NE.1) GO TO 100	00070600
C	00070700
C	00070800
C NCL = 1	00070900
C	00071000
T(N,1) = TCL(1)	00071100
T(N,2) = TO(N)	00071200
TB(N,1) = TBA(N)	00071300
ATB(N,1) = ATBA(N)	00071400
GO TO 1000	00071500
C	00071600
C	00071700
100 CONTINUE	00071800
C	00071900
C NCL . GE . 2	00072000
C	00072100
L = 2	00072200
T(N,1) = TCL(1)	00072300
S1 = RCL(1)	00072400
S2 = 0.	00072500
DO 1030 K=1,NCL	00072600
C	00072700
150 CONTINUE	00072800
IF (R(N,K+1)-RCL(L)) 300,300,200	00072900
200 CONTINUE	00073000
DRDR = RCL(L)-S1	00073100
ARA = AREA(S1,DRDR)	00073200
ARA = ARA*ATCL(L-1)	00073300
S2 = S2 + ARA	00073400
S1 = RCL(L)	00073500
L = L+1	00073600
IF (L.GT.7) GO TO 310	00073700
GO TO 150	00073800
300 CONTINUE	00073900
DRDR = DRR-(RCL(L)-R(N,K+1))	00074000
ARA = AREA(S1,DRDR)	00074100
ARA = ARA*ATCL(L-1)	00074200
S2 = S2 + ARA	00074300
310 CONTINUE	00074400
IF (L.GT.7) L=7	00074500
RAT = (RCL(L)-R(N,K+1)) / DRR	00074600
T(N,K+1) = TCL(L) + RAT*(TCL(L-1)-TCL(L))	00074700
TB(N,K) = (T(N,K)+T(N,K+1)) / 2.	00074800
ATB(N,K) = S2 / (DR*(2.*R(N, K)+DR))	00074900
S1 = R(N,K+1)	00075000
S2 = 0.0	00075100
C	00075200
1030 CONTINUE	00075300
C	00075400
T(N,NCL1) = TO(N)	00075500
C	00075600
1000 CONTINUE	00075700

PNCT841-75-17

C	00075800
RETURN	00075900
END	00076000

	SUBROUTINE COEF (OTOM,N)	00076100
C		00076200
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00076300
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00076400
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00076500
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00076600
*	N12C,N13C,N14C,N15C,	00076700
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00076800
	COMMON /CONST/ FKA1,FKB1,FKC1,FKA2,FKB2,FKC2,FKA3,FKB3,FKC3,	00076900
*	FA11,FA12,FA21,FA22,FA31,FA32,	00077000
*	CK1,CK2,CA1,CA2,XEWTON(7)	00077100
	COMMON /FBRATR/ JFA	00077200
C		00077300
	DIMENSION OTOM(NN)	00077400
C		00077500
C		00077600
C	COEFFICIENTS FOR FUEL THERMAL CONDUCTIVITY	00077700
C		00077800
C		00077900
	OTOMX = 2.00 - OTOM(N)	00078000
	IF(OTOMX) 100,100,200	00078100
100	OTOMX = -2.*OTOMX	00078200
200	DELKF = 5.0*OTOMX**2 - 3.7*OTOMX + 1.0	00078300
	FKA1 = -0.018*DELKF	00078400
	FKB1 = 54.0*DELKF	00078500
	FKC1 = 1.75E-12 * DELKF	00078600
	FKA2 = 0.011*DELKF	00078700
	FKB2 = 0.485 / DELKF	00078800
	FKC2 = 0.4465 / DELKF	00078900
	IF(JFA-2) 10,20,20	00079000
10	CONTINUE	00079100
	FKA3 = 3.11 + 355.0*OTOMX	00079200
	FKB3 = 0.0272	00079300
	FKC3 = 5.39E-13	00079400
	GO TO 30	00079500
20	CONTINUE	00079600
	FKA3 = 3.38	00079700
	FKB3 = 0.0262	00079800
	FKC3 = 4.79E-13	00079900
	IF(NOK.EQ.4) FKC3 = 6.1256E-13	00080000
30	CONTINUE	00080100
	RETURN	00080200
	END	00080300

	SUBROUTINE CORTMP (TOUTER,TOUTO,PTOUTR,ISW,TOUT)	00080400
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00080500
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00080600
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00080700
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00080800
*	N12C,N13C,N14C,N15C,	00080900
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00081000
	COMMON /IOFIL/ IO5,IO6,IO7	00081100
	DIMENSION TOUTER(NN) , TOUTO(NN) , PTOUTR(NN)	00081200
C		00081300
C	COOLANT TEMPERATURE ADJUSTMENT DUE TO TIME CUT	00081400
C		00081500
	IF(ISW.EQ.0) GO TO 50	00081600
	GO TO (10,20,30,40) , ISW	00081700
	10 DO 100 I=1,NN	00081800
	100 TOUTER(I) = TOUTO(I)	00081900
	GO TO 60	00082000
	20 DO 200 I=1,NN	00082100
	200 TOUTO(I) = TOUTER(I)	00082200
	GO TO 60	00082300
	30 DO 300 I=1,NN	00082400
	300 TOUTER(I) = PTOUTR(I)+(TOUTER(I)-PTOUTR(I)) / 2.0	00082500
	GO TO 60	00082600
	40 DO 400 I=1,NN	00082700
	400 PTOUTR(I) = TOUTER(I)	00082800
	GO TO 60	00082900
	50 READ(IO5,9000) (TOUTER(I),I=1,NN)	00083000
	TOUT = TOUTER(NN)	00083100
	60 RETURN	00083200
	9000 FORMAT(5E15.3)	00083300
	END	00083400

	SUBROUTINE CRACK	00083500
*	(SIGR,SIGC,SIGZ,BM,G,TBF,TPR,TPC,TPZ,TSR,TSC,TSZ,	00083600
*	PRS,LM,LMR,LMC,LMZ,DELT,NS,JCR)	00083700
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NDC,NPLOT,NCRK,	00083800
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00083900
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00084000
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00084100
*	N12C,N13C,N14C,N15C,	00084200
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00084300
	DIMENSION SIGR(NTR) , SIGC(NTR) , SIGZ(NTR) ,	00084400
*	BM(NN,NTR) , G(NN,NTR) , TBF(NN,NFR) , TPR(NN,NTR) ,	00084500
*	TPC(NN,NTR) , TPZ(NN,NTR) , TSR(NN,NTR) ,	00084600
*	TSC(NN,NTR) , TSZ(NN,NTR) , PRS(NN,NTR) ,	00084700
*	LM(NN,NTR) , LMR(NN,NTR) , LMC(NN,NTR) ,	00084800
*	LMZ(NN,NTR)	00084900
	COMMON /XCRAK/ THLC,HEAL1,HEAL2,SIGF0,SIGF1	00085000
C		00085100
C		00085200
C	CRACKING CHECK J.T.A ROBERTS AND B.J. WRONA	00085300
C		00085400
C		00085500
	JCR = 0	00085600
C		00085700
	IF (DELT.LE.0.001) RETURN	00085800
C		00085900
	DO 1000 I=1,NFR	00086000
	TS = TBF(NS,I) + 273.	00086100
C		00086200
	SIG = SIGR(I)	00086300
	L = 1	00086400
C		00086500
C	L = 1 RADIAL DIRECTION	00086600
C	L = 2 THETA DIRECTION	00086700
C	L = 3 AXIAL DIRECTION	00086800
C		00086900
	IF (SIG.GT.SIGC(I)) GO TO 10	00087000
	SIG = SIGC(I)	00087100
	L = 2	00087200
10	CONTINUE	00087300
	IF (SIG.GT.SIGZ(I)) GO TO 20	00087400
	SIG = SIGZ(I)	00087500
	L = 3	00087600
20	CONTINUE	00087700
C		00087800
C		00087900
C	CRACKING CHECK	00088000
C		00088100
	SIGF = SIGF0 + SIGF1*TS	00088200
C		00088300
C	SIGF MEANS CRITICAL STRESS OF BRITTLE-TO-DUCTILE TRANSITION	00088400
C		00088500
	IF (SIG.LT.SIGF) GO TO 500	00088600
C		00088700
	JCR = 1	00088800
	TSR(NS,I) = TSR(NS,I) + SIG/ (9.*BM(NS,I))	00088900
	TSC(NS,I) = TSR(NS,I)	00089000
	TSZ(NS,I) = TSR(NS,I)	00089100
	X = SIG / (6.*G(NS,I))	00089200

IF (L-2) 30,40,50	00089300
30 CONTINUE	00089400
LMR(NS,I) = LMR(NS,I) + 1	00089500
TPR(NS,I) = TPR(NS,I) + 2.*X	00089600
TPC(NS,I) = TPC(NS,I) - X	00089700
TPZ(NS,I) = TPZ(NS,I) - X	00089800
GO TO 60	00089900
C	00090000
40 CONTINUE	00090100
LMC(NS,I) = LMC(NS,I) + 1	00090200
TPC(NS,I) = TPC(NS,I) + 2.*X	00090300
TPR(NS,I) = TPR(NS,I) - X	00090400
TPZ(NS,I) = TPZ(NS,I) - X	00090500
GO TO 60	00090600
C	00090700
50 CONTINUE	00090800
LMZ(NS,I) = LMZ(NS,I) + 1	00090900
TPZ(NS,I) = TPZ(NS,I) + 2.*X	00091000
TPR(NS,I) = TPR(NS,I) - X	00091100
TPC(NS,I) = TPC(NS,I) - X	00091200
60 CONTINUE	00091300
C	00091400
LM(NS,I) = LMR(NS,I) + LMC(NS,I) + LMZ(NS,I)	00091500
C	00091600
GO TO 1000	00091700
C	00091800
C	00091900
C HEALING CHECK	00092000
C	00092100
C	00092200
500 CONTINUE	00092300
IF (LM(NS,I).EQ.0) GO TO 1000	00092400
HEAL = HEAL1	00092500
THL = THLC + 273.	00092600
IF (TS.GT.THLC) HEAL = HEAL2	00092700
IF (-(SIGR(I)+SIGC(I)+SIGZ(I))/3. .LE. -HEAL)	00092800
* GO TO 1000	00092900
LMR(NS,I) = 0	00093000
LMC(NS,I) = 0	00093100
LMZ(NS,I) = 0	00093200
LM(NS,I) = 0	00093300
C	00093400
JCR = 2	00093500
C	00093600
C	00093700
1000 CONTINUE	00093800
C	00093900
C	00094000
RETURN	00094100
END	00094200

	SUBROUTINE CREEC (EQSIG,DEQP,TBC,FLNC,FLUX,DELT,N)	00094300
C	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NCS,NOC,NPLOT,NCRK,	00094400
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NCS,NOC,NPLOT,NCRK,	00094500
*	NTR,NFR,NTRI,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00094600
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00094700
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00094800
*	N12C,N13C,N14C,N15C,	00094900
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00095000
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00095100
*	PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,	00095200
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00095300
	COMMON /XCRC/ A,Q,AE,BE,CE,FM,FN	00095400
	COMMON /FBRATR/ JFA	00095500
C		00095600
	DIMENSION EQSIG(NN,NTR) , DEQP(NN,NTR) , TBC(NN,NCL) ,	00095700
*	FLNC(NN) , FLUX(NN)	00095800
C		00095900
C		00096000
C	CLADDING CREEP STRAIN CALC.	00096100
C		00096200
C		00096300
	DO 1000 I=1,NCL	00096400
C		00096500
	J = NFR + I	00096600
	T = TBC(N,I)*1.8 + 32. + 460.	00096700
C	T NOW IN DEG R	00096800
	IF(EQSIG(N,J).LT.10.0) EQSIG(N,J)=10.0	00096900
C	PREVENTING ZERO DIVIDE	00097000
	ES = EQSIG(N,J) / 7.03E-02	00097100
C		00097200
	IF(JFA-2) 50,500,500	00097300
	50 CONTINUE	00097400
	IF(NOC+1) 100,200,200	00097500
	100 C = T/910.	00097600
	GO TO 300	00097700
	200 C = 1.0	00097800
	300 CONTINUE	00097900
C		00098000
	AEC = AE/C	00098100
	BEC = BE*C	00098200
	CEC = CE/C	00098300
C		00098400
	B = CEC + AEC/BEC * EXP(-FLNC(N)/BEC)	00098500
C		00098600
C		00098700
C	RKR IS IDEAL GAS CONSTANT IN CAL/MOLE/DEG R	00098800
C		00098900
	X1 = A * EXP(-Q/(RKR*T)) * ES**FM	00099000
	X2 = B * FLUX(N) * ES**FN	00099100
C		00099200
	DEP = (X1+X2) * DELT	00099300
	GO TO 900	00099400
	500 IF(NOC+1) 700,600,600	00099500
	600 DEP = 4.0E-27*ES*FLUX(N)*(TBC(N,I)-160.)*DELT	00099600
	GO TO 900	00099700
	700 TEMP = TBC(N,I)+273.	00099800
	ES = 1.63E-03*EQSIG(N,J)	00099900
	XFLX = FLUX(N)	00100000

PNCT841-75-17

DEP = 1.02E-13*EXP(-7053./TEMP)*XFLX**0.85*SINH(ES)*DELT	00100100
900 IF(DEP.LE.0.) DEP = 1.0E-30	00100200
DEQP(N,J) = DEP	00100300
C	00100400
1000 CONTINUE	00100500
C	00100600
C	00100700
RETURN	00100800
END	00100900

	SUBROUTINE CREEF (EQSIG,DEQP,TBF,POW,FD,FLUX,FTHP,LM,DELT,N,	00101000
*	G,EPSEQ)	00101100
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00101200
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00101300
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00101400
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00101500
*	N12C,N13C,N14C,N15C,	00101600
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00101700
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00101800
*	PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,	00101900
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00102000
	COMMON /XCREF/ AO,A1,CFR,B,Q,Q1,FM,FN,GD	00102100
C		00102200
	DIMENSION EQSIG(NN,NTR) , DEQP(NN,NTR) , TBF(NN,NFR) , POW(NN) ,	00102300
*	FD(NN,NFR) , FLUX(NN) , FTHP(NN,NTR) , LM(NN,NTR) ,	00102400
*	G(NN,NTR) , EPSEQ(NN,NTR)	00102500
C		00102600
C		00102700
C	FUEL CREEP STRAIN CALC.	00102800
C		00102900
C		00103000
	DO 1000 I=1,NFR	00103100
	FSRT = POW(N)*FD(N,I) * 3.1E+10	00103200
	TX = TBF(N,I)*1.8 + 32. + 460.	00103300
C	TX NOW IN DEG R	00103400
C		00103500
C	RKR IS IDEAL GAS CONSTANT IN CAL/MOLE/DEG R	00103600
C	FSRT IS NO. OF FISSIONS/ SEC / CC	00103700
C	TX NOW IN DEG R	00103800
C		00103900
	IF(EQSIG(N,I).LT.10.0) EQSIG(N,I)=10.0	00104000
C	PREVENTING ZERO DIVIDE	00104100
	IF (EPSEQ(N,I).GT.0.0) GO TO 100	00104200
	EPSEQ(N,I) = 1.E-30	00104300
	DEQP(N,I) = EPSEQ(N,I)	00104400
	GO TO 1000	00104500
100	CONTINUE	00104600
	ES = EQSIG(N,I) / 7.03E-02	00104700
	LML = LM(N,I)	00104800
C	A = AO*1.5**{LML*(FM-1.)}	00104900
	A = AO	00105000
	X0 = 0.0	00105100
	IT = 0	00105200
200	CONTINUE	00105300
	X1 = A*EXP(-Q/(RKR*TX))*(1.+FSRT/CFR)*ES**FM	00105400
C	X2 = A1*EXP(-Q1/(RKR*TX))/(GD**2) + B*FLUX(N)	00105500
	X2 = A1*EXP(-Q1/(RKR*TX))/(GD**2) + B*FSRT	00105600
	X2 = X2*ES**FN	00105700
	X3 = (FM*X1+FN*X2)*DELT	00105800
	X2 = (X1+X2)*DELT	00105900
	IF(X2.LE.0.) GO TO 500	00106000
	X4 = ABS(X2-X0)/X2	00106100
	X0 = X2	00106200
	X = ES	00106300
	ES = ES*(EPSEQ(N,I)-X2+X3) / (X3+ES/G(N,I)/3.)	00106400
	IF(ES.LT.0.) GO TO 500	00106500
	IT = IT + 1	00106600
	IF (IT.GT.100) GO TO 500	00106700

	IF (X4-0.0001) 300,200,200	00106800
300	IF (X4-0.0000001) 500,400,400	00106900
400	X5 = ABS(ES-X)	00107000
	IF (X5-0.5) 500,200,200	00107100
500	CONTINUE	00107200
	DEP = X2	00107300
	IF(DEP.LE.0.) DEP=1.E-30	00107400
	DEQP(N,I) = DEP	00107500
C		00107600
1000	CONTINUE	00107700
C		00107800
	RETURN	00107900
	END	00108000

	SUBROUTINE EDTGRS (IT,TIM,POWAV,TIN,TOUT,DELT,POW,FLUX,BU,TNA,	00000100
*	XCPO,XCPI,TCPI,TCPO,RCO,RCI,TCLO,TCL,FLZ,CLZ,	00000200
*	FL,CL,OTOM,CPU,RHOT,RH,XM,RVB,RCB,REB,	00000300
*	RUB,RF,RC,HG,HC,TOGAS,GR,TFR,TBFA,ATBFA,	00000400
*	TBCLA,ATBLA,RCP,TCP,TBCPA,ATBPA,FLNC,GS,	00000500
*	PLPR,CVPR,FLQ,CLO ,BUM,PO,FD,FOPM)	00000600
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NDC,NPLOT,NCRK,	00000700
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00000800
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00000900
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00001000
*	N12C,N13C,N14C,N15C,	00001100
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00001200
	COMMON /OTHERS/ PG(3),CMI(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00001300
*	PI,PI2,OP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,	00001400
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCQN,PDBU,BUB	00001500
	COMMON /CAPSUL/ RCPI,RCPO,AFNA	00001600
	COMMON /COMPLT/ YP,DDOD,TITLE(20)	00001700
	COMMON /IOFIL/ IO5,IO6,IO7	00001800
C		00001900
	DIMENSION POW(NN) , FLUX(NN) , BU(NN) , TNA(NN) , XCPO(NN) ,	00002000
*	XCPI(NN) , TCPI(NN) , TCPO(NN) , RCO(NN) , RCI(NN) ,	00002100
*	TCLO(NN) , TCL(NN,NCL1) , FLZ(NN) , CLZ(NN) ,	00002200
*	OTOM(NN) , RHOT(NN) , RH(NN,NTR) , XM(NN,3) ,	00002300
*	RVB(NN) , RCB(NN) , REB(NN) , CPU(NN,NP) ,	00002400
*	RUB(NN) , RF(NN,NFR1) , RC(NN,NCL1) , HG(NN) ,	00002500
*	HC(NN) , GR(NN) , TFR(NN,NFR1) , TBFA(NN,3) ,	00002600
*	ATBFA(NN,3) , TBCLA(NN) , ATBLA(NN) , RCP(NN,NCL1) ,	00002700
*	TCP(NN,NCL1) , TBCPA(NN) , ATBPA(NN) , FLNC(NN) ,	00002800
*	GS(NN,NFR) , BUM(NN) , FD(NN,NFR) , FDPM(NN,NP)	00002900
C		00003000
C		00003100
C	GROSS OUTPUT OF EACH REGION FOR EACH STEP	00003200
C		00003300
C		00003400
	DD 50 K=1,NN	00003500
50	BUM(K)=BU(K)*8386.0	00003600
	WRITE(IO6,9000) IT,TIM,POWAV,TOUT,TIN,DELT,CFLX,PD	00003700
	WRITE(IO6,9001)	00003800
	WRITE(IO6,9002) POW	00003900
	WRITE(IO6,9003) FLUX	00004000
	WRITE(IO6,9004) FLNC	00004100
	WRITE(IO6,9005) BU	00004200
	WRITE(IO6,9006) BUM	00004300
	WRITE(IO6,9007) FLZ	00004400
	WRITE(IO6,9008) CLZ	00004500
	WRITE(IO6,9009) RHOT	00004600
	WRITE(IO6,9010) TNA	00004700
	DD 100 K=1,NN	00004800
	HC(K) = HC(K) * 0.002093	00004900
100	HG(K) = HG(K) * 0.002093	00005000
	WRITE(IO6,9011) HC	00005100
	WRITE(IO6,9012) HG	00005200
	DD 200 K=1,NN	00005300
	HC(K) = HC(K) / 0.002093	00005400
200	HG(K) = HG(K) / 0.002093	00005500
	WRITE(IO6,9013) GR	00005600
	TGS = 0.	00005700
	DD 900 N=1,NN	00005800

	DO 900 K=1,NFR	00005900
	TGS = TGS + GS(N,K)	00006000
900	CONTINUE	00006100
	X = TOGAS - GASI	00006200
	YP = 100.*X/(TGS+X)	00006300
	WRITE(IO6,9014) TOGAS,X,YP	00006400
	WRITE(IO6,9015) PLPR,CVPR	00006500
	FDLBL = 100.*(FL-FLO)/FLO	00006600
	CDLBL = 100.*(CL-CLO)/CLO	00006700
	WRITE(IO6,9016) FL,FDLBL,CL,CDLBL	00006800
	WRITE(IO6,9101)	00006900
	DO 1000 N=1,NN	00007000
	K = N	00007100
	WRITE(IO6,9102) K	00007200
	IF(K.EQ.NN) GO TO 950	00007300
	IF(XCPI(K).GE.XCPO(K)) GO TO 910	00007400
	WRITE(IO6,9103) RVB(K),RCB(K),REB(K),RUB(K),RCI(K),RCO(K),XCPI(K),	00007500
	* XCPO(K)	00007600
	WRITE(IO6,9104) TFR(K,1),TFR(K,NCR+1),TFR(K,NCR+NER+1),TFR(K,NFR1)	00007700
	* ,TCL(K,1),TCL(K,NCL1),TCP(K,1),TCP(K,NCL1)	00007800
	WRITE(IO6,9105) TBFA(K,1),TBFA(K,2),TBFA(K,3),TBCLA(K),TBCPA(K)	00007900
	WRITE(IO6,9106) ATBFA(K,1),ATBFA(K,2),ATBFA(K,3),ATBLA(K),ATBPA(K)	00008000
	WRITE(IO6,9107) RH(K,1),RH(K,NCR+1),RH(K,NCR+NER+1),RH(K,NFR+1)	00008100
	WRITE(IO6,9108) XM(K,1),XM(K,2),XM(K,3)	00008200
	GO TO 920	00008300
910	WRITE(IO6,9103) RVB(K),RCB(K),REB(K),RUB(K),RCI(K),RCO(K)	00008400
	WRITE(IO6,9104) TFR(K,1),TFR(K,NCR+1),TFR(K,NCR+NER+1),TFR(K,NFR1)	00008500
	* ,TCL(K,1),TCL(K,NCL1)	00008600
	WRITE(IO6,9105) TBFA(K,1),TBFA(K,2),TBFA(K,3),TBCLA(K)	00008700
	WRITE(IO6,9106) ATBFA(K,1),ATBFA(K,2),ATBFA(K,3),ATBLA(K)	00008800
	WRITE(IO6,9107) RH(K,1),RH(K,NCR+1),RH(K,NCR+NER+1),RH(K,NFR+1)	00008900
	WRITE(IO6,9108) XM(K,1),XM(K,2),XM(K,3)	00009000
920	CONTINUE	00009100
	GO TO 1000	00009200
950	X=0.0	00009300
	IF(XCPI(K).GE.XCPO(K)) GO TO 960	00009400
	WRITE(IO6,9103) X,X,X,X,RCI(K),RCO(K),XCPI(K),XCPO(K)	00009500
	WRITE(IO6,9104) X,X,X,X,TCL(K,1),TCL(K,NCL1),TCP(K,1),TCP(K,NCL1)	00009600
	WRITE(IO6,9105) X,X,X,TBCLA(K),TBCPA(K)	00009700
	WRITE(IO6,9106) X,X,X,ATBLA(K),ATBPA(K)	00009800
	WRITE(IO6,9107) X,X,X,RH(K,NFR+1)	00009900
	GO TO 970	00010000
960	WRITE(IO6,9103) X,X,X,X,RCI(K),RCO(K)	00010100
	WRITE(IO6,9104) X,X,X,X,TCL(K,1),TCL(K,NCL1)	00010200
	WRITE(IO6,9105) X,X,X,TBCLA(K)	00010300
	WRITE(IO6,9106) X,X,X,ATBLA(K)	00010400
	WRITE(IO6,9107) X,X,X,RH(K,NFR+1)	00010500
970	CONTINUE	00010600
1000	CONTINUE	00010700
	WRITE(IO6,9050)	00010800
C		00010900
C		00011000
	RETURN	00011100
C		00011200
9000	FORMAT (1H1, / ' ' , 114(' *') , /	00011300
	* ' STEP NO.',I3,5X,'TIME (HRS) =',F11.3,7X,'POWAV (W/00011400	
	*CM) =',F11.3,5X,'TOUT (C) =',F8.3,2X,'TIN (C) =',F8.3, / 18X,'DELT00011500	
	* (HRS) =',F11.3,7X,'CFLX (N/W/CM) =',E11.3 ,4X,'PO (KG/CM**2) =' ,00011600	

```

*F7.2, / ' ',114('*) )                                00011700
9001 FORMAT( / 35X,'==== GROSS OUTPUT NEXT =====',//,1H,'+++ AXIA00011800
*L DISTRIBUTION (N=1,NN) +++', / )                      00011900
9002 FORMAT(1H,' POWER(W/CM) ',7F15.3,(/,1H,23X,7F15.3) )00012000
9003 FORMAT(1H,' FLUX(N/CM**2YSEC) ',7E15.5,(/,1H,25X,7E15.5) )00012100
9004 FORMAT(1H,' FLUENCE(N/CM**2) ',7E15.5,(/,1H,25X,7E15.5) )00012200
9005 FORMAT(1H,' BURN UP(A/O) ',7F15.5,(/,1H,25X,7F15.5) )00012300
9006 FORMAT(1H,' BURN UP(MWD/T) ',7F15.1,(/,1H,21X,7F15.1) )00012400
9007 FORMAT(1H,' FUEL LENGTH(CM) ',7F15.5,(/,1H,25X,7F15.5) )00012500
9008 FORMAT(1H,' CLAD LENGTH(CM) ',7F15.5,(/,1H,25X,7F15.5) )00012600
9009 FORMAT(1H,' THEO. DENS.(G/CC) ',7F15.5,(/,1H,25X,7F15.5) )00012700
9010 FORMAT(1H,' COOLANT TEMP.(C) ',7F15.3,(/,1H,23X,7F15.3) )00012800
9011 FORMAT(1H,' FILM H.(W/CM**2/C) ',7F15.5,(/,1H,25X,7F15.5) )00012900
9012 FORMAT(1H,' GAP H.(W/CM**2/C) ',7F15.5,(/,1H,25X,7F15.5) )00013000
9013 FORMAT(1H,' DELT GAS REL.(MOLES)',4X,7F15.7,(/,1H,27X,7F15.7) )00013100
9014 FORMAT( 1H,' TOTAL GAS IN PLENUM',16X,F15.8,' (MOLES)', / 00013200
* ,1H,' CUMULATIVE GAS RELEASED',12X,F15.8,' (MOLES)',00013300
*/ ,1H,' PERCENT OF TOTAL F.P. GAS',7X,F15.5,' (O/O)00013400
** ) 00013500
9015 FORMAT( 1H,' PLENUM PRESSURE',15X,F15.3,6X,'(KG/CM**2)', /00013600
* ,1H,' CENTRAL HOLE PRESSURE',9X,F15.3,6X,'(KG/CM**2)00013700
*)' ) 00013800
9016 FORMAT( 1H,' TOTAL FUEL LENGTH AND FL/FLO',4X,F15.5,' (C00013900
*M)',10X,F15.5,2X,'(O/O)', / 00014000
* ,1H,' TOTAL CLAD LENGTH AND CL/CLO',4X,F15.5,' (C00014100
*M)',10X,F15.5,2X,'(O/O)' ) 00014200
9101 FORMAT( 1H,' +++ PERFORMANCE +++' ) 00014300
9102 FORMAT( 1H,' * AXIAL SECTION = ',I2,' * ' / 14X,' FUEL C00014400
*ENTER COLUMNAR GR. EQUIAXED GR. FUEL OUTER CLAD INNER 00014500
* CLAD OUTER CAPSULE INNER CAPSULE OUTER' ) 00014600
9103 FORMAT( 1H,' RADIUS(CM)',8F15.5 ) 00014700
9104 FORMAT( 1H,' TEMP.(C)',8F15.3 ) 00014800
9105 FORMAT( 1H,' AV.TEMP(C)',5X,3F15.3,15X,F15.3,15X,F15.3 ) 00014900
9106 FORMAT( 1H,' ALPHA-T(-)',7X,3F15.5,15X,F15.5,15X,F15.5 ) 00015000
9107 FORMAT( 1H,' DENS(G/CC)',7X,3F15.5,15X,F15.5 ) 00015100
9108 FORMAT( 1H,' MASS(G/CM)',7X,3F15.5 ) 00015200
9050 FORMAT( 1H,' / , 35X,'==== STRESS AND STRAIN OUTPUT NEXT ====='00015300
*) 00015400
C 00015500
END 00015600

```

	SUBROUTINE EDTINP (RVB,RCB,REB,RUB,RCI,RCD,QSFF,QSFP,OTOM,CPU,	00015700
	* RHOP,F,XCPI,XCPO,FLZ,CLZ,FL,CL,PTOP,PO)	00015800
C		00015900
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00016000
	* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00016100
	* IGEN,IGEN1,NONZR,IGENP,IGENPI,NONZRP,	00016200
	* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00016300
	* N12C,N13C,N14C,N15C,	00016400
	* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00016500
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00016600
	* PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00016700
	* CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00016800
	COMMON /CAPSUL/ RCPI,RCPO,AFNA	00016900
	COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00017000
	COMMON /IOFIL/ IO5,IO6,IO7	00017100
C		00017200
	DIMENSION RVB(NN) , RCB(NN) , REB(NN) , RUB(NN) , RCI(NN) ,	00017300
	* RCO(NN) , QSFF(NN) , QSFP(NN) , OTOM(NN) , CPU(NN,NP) ,	00017400
	* RHOP(NN,3) , F(NN,NP) , XCPI(NN) , XCPO(NN) , FLZ(NN)	00017500
	* CLZ(NN)	00017600
C		00017700
C		00017800
C	OUTPUT OF INPUT VALUES	00017900
C		00018000
C		00018100
C		00018200
	WRITE(IO6,9000)	00018300
C		00018400
	WRITE(IO6,9001) NN,NCR,NER,NUR,NCL,NP,NOK	00018500
	WRITE(IO6,9022) NOS,NOC,NCRK,NPLOT,IPFUL,IPSTN,IPSTS,IPCRK,NBUF	00018600
C	RVB	00018700
	WRITE(IO6,9002) RVB	00018800
C	RCB	00018900
	WRITE(IO6,9003) RCB	00019000
C	REB	00019100
	WRITE(IO6,9004) REB	00019200
C	RUB	00019300
	WRITE(IO6,9005) RUB	00019400
C	RCI	00019500
	WRITE(IO6,9006) RCI	00019600
C	RCO	00019700
	WRITE(IO6,9007) RCO	00019800
C		00019900
	IF (RCPO-RCPI) 200,200,100	00020000
100	CONTINUE	00020100
	WRITE(IO6,9008) XCPI	00020200
	WRITE(IO6,9009) XCPO	00020300
	GO TO 300	00020400
C		00020500
200	CONTINUE	00020600
	WRITE(IO6,9010)	00020700
C		00020800
300	CONTINUE	00020900
	WRITE(IO6,9025) AFNA	00021000
	WRITE(IO6,9011) FL , FLZ	00021100
C		00021200
	WRITE(IO6,9012) CL , CLZ	00021300
C		00021400

WRITE(I06,9013)	00021500
C	00021600
DO 1000 I=1,NN	00021700
WRITE(I06,9014) I , (RHOP(I,J),J=1,3) , RHOC	00021800
1000 CONTINUE	00021900
C	00022000
WRITE(I06,9015) OTOM	00022100
WRITE(I06,9016)	00022200
DO 2000 I=1,NNF	00022300
WRITE(I06,9017) I , (CPU(I,J),J=1,NP)	00022400
2000 CONTINUE	00022500
C	00022600
WRITE(I06,9018) QSFF	00022700
WRITE(I06,9019) QSFP	00022800
WRITE(I06,9020)	00022900
C	00023000
DO 3000 I=1,NNF	00023100
WRITE(I06,9021) I , (F(I,J),J=1,NP)	00023200
3000 CONTINUE	00023300
C	00023400
C	00023500
WRITE(I06,9024) GAST	00023600
C	00023700
RETURN	00023800
C	00023900
C	00024000
9000 FORMAT (1H1 , // 25X,'==== INPUT VALUES OUTPUT NEXT =====',//)	00024100
9001 FORMAT (1H0,' ---- CONTROL NUMBER AND OPTION ' , //	00024200
* 10X,'NUMBER OF AXIAL SECTION NN =' , I3 , /	00024300
* 10X,'NUMBER OF MESH FOR COLUMNAR GRAIN REGION NCR =' ,	00024400
* I3 , /	00024500
* 10X,'NUMBER OF MESH FOR EQUIAXED GRAIN REGION NER =' ,	00024600
* I3 , /	00024700
* 10X,'NUMBER OF MESH FOR UNDISTURBED REGION NUR =' ,	00024800
* I3 , /	00024900
* 10X,'NUMBER OF MESH FOR CLAD REGION NCL =' , I3 , /	00025000
* 10X,'NUMBER OF MESH FOR PLUTONIUM MIGRATION NP =' , I3 , /	00025100
* 10X,'FUEL THERMAL CONDUCTIVITY OPTION NOK =' , I3 ,	00025200
* 5X,'(1/2/3 = SAYLES/BAILY-ASAMOTO/BIANCHERIA)')	00025300
9002 FORMAT (1H0,' ---- CENTRAL VOID RADIUS (CM) 1-NN ' , //	00025400
* (10X,9F12.4))	00025500
9003 FORMAT (1H0,' ---- COLUMNAR GRAIN REGION OUTER BOUNDARY (CM)	00025600
*1-NN ' , // (10X,9F12.4))	00025700
9004 FORMAT (1H0,' ---- EQUIAXED GRAIN REGION OUTER BOUNDARY (CM)	00025800
*1-NN ' , // (10X,9F12.4))	00025900
9005 FORMAT (1H0,' ---- FUEL OUTER RADIUS (CM) 1-NN ' , //	00026000
* (10X,9F12.4))	00026100
9006 FORMAT (1H0,' ---- CLAD INNER RADIUS (CM) 1-NN ' , //	00026200
* (10X,9F12.4))	00026300
9007 FORMAT (1H0,' ---- CLAD OUTER RADIUS (CM) 1-NN ' , //	00026400
* (10X,9F12.4))	00026500
9008 FORMAT (1H0,' ---- CAPSULE INNER RADIUS (CM) 1-NN ' , //	00026600
* (10X,9F12.4))	00026700
9009 FORMAT (1H0,' ---- CAPSULE OUTER RADIUS (CM) 1-NN ' , //	00026800
* (10X,9F12.4))	00026900
9010 FORMAT (1H0,' **** THIS FUEL IS UN-ENCAPSULATED **** ')	00027000
9011 FORMAT (1H0,' ---- FUEL AXIAL LENGTH (CM) 1-NN ... TOTAL	00027100
* AXIAL LENGTH =' , F12.4 , ' (CM) ...' , // (10X,9F12.4))	00027200

9012	FORMAT (1H0,' ---- CLAD AXIAL LENGTH (CM) 1-NN ... TOTAL	00027300
*	AXIAL LENGTH =' , F12.4 , ' (CM) ...' , // (10X,9F12.4))	00027400
9013	FORMAT (1H0,' ---- DENSITY OF FUEL (3 REGIONS) AND CLAD (G/CC	00027500
*	) ' /)	00027600
9014	FORMAT (1H ,10X,'AXIAL SECTION =' ,I3,5X,4F12.4)	00027700
9015	FORMAT (1H0,' ---- O/M OF FUEL PU-O(X) 1-NN ' , //	00027800
*	(10X,9F12.4))	00027900
9016	FORMAT (1H0,' ---- PLUTONIUM CONCENTRATION (WEIGHT-FRACTION)	00028000
*	1-NP ' /)	00028100
9017	FORMAT (1H ,10X,'AXIAL SECTION =' ,I3,5X,7F12.4, / (34X,7F12.4))	00028200
9018	FORMAT (1H0,' ---- AXIAL FAST FLUX (E .GE. 0.1MEV) RATE 1-NN	00028300
*	' , // (10X,9F12.4))	00028400
9019	FORMAT (1H0,' ---- AXIAL HEAT GENERATION RATE 1-NN ' , //	00028500
*	(10X,9F12.4))	00028600
9020	FORMAT (1H0,' ---- RADIAL HEAT GENERATION RATE 1-NP ' , /)	00028700
9021	FORMAT (1H ,10X,'AXIAL SECTION =' ,I3,5X,7F12.4, / (34X,7F12.4))	00028800
9022	FORMAT (10X,'CLAD SWELLING OPTION NOS =' ,I3,	00028900
*	5X,'(-1/-2 = ANNEALED 304,316S.S./20P.C.COLD WORK 316S.S.)	00029000
*	' , /	00029100
*	10X,'CLAD CREEP OPTION NCC =' ,I3,	00029200
*	5X,'(-1/-2 = BOLTAX,BIANCHERIA,GUHA/MODIFIED)' , /	00029300
*	10X,'CRACK OPTION NCRK =' ,I3 , 5X , '(0/1 = NO CRACK CALC	00029400
*	*,/CRACK CALC.)' , /	00029500
*	10X,'PLOT OPTION NPLT =' ,I3,5X,'(0/1 = NO PLOT/PLOT)' , /	00029600
*	20X,'AXIAL SECTION NUMBER FOR FUEL =' ,I3, /	00029700
*	20X,'AXIAL SECTION NUMBER FOR STRAIN =' ,I3 , /	00029800
*	20X,'AXIAL SECTION NUMBER FOR STRESS =' ,I3 , /	00029900
*	20X,'AXIAL SECTION NUMBER FOR CRACK =' ,I3, /	00030000
*	20X,'STORAGE(WORDS) REQUIRED FOR PLOTTING STEP =' ,I4, /)	00030100
9024	FORMAT(1H0,' ---- INITIAL GAS IN PLENUM =' ,F15.7,' (MOLES)')	00030200
9025	FORMAT(1H0,' ---- COOLANT FLOW AREA =' , F15.8 , ' (CM**2)')	00030300
C		00030400
END		00030500

	SUBROUTINE EDTMTX (EPR,EPC,EPZ,SIGR,SIGC,SIGZ,TPR,TPC,TPZ,TSR,	00030600
*	TSC,TSZ,RDIS,CRIF,RCP,TCP,TBCP,ATBCP,RCL,TCL,	00030700
*	TBCL,ATBCL,YNG,POS,SHR,BM,RF,TF,TBF,ATBF,PRS,	00030800
*	EHP,TPRF,TPRC,TPCF,TPCC,TPZF,TPZC,	00030900
*	PRES,KFRIC,DPR,DPC,DPZ,	00031000
*	DSR,DSC,DSZ,POSB,SHRB,SWG,SWS,N,RPM,TPM,GS,	00031100
*	CPU,LM,LMR,LMC,LMZ,RCOO)	00031200
C		00031300
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00031400
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00031500
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00031600
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00031700
*	N12C,N13C,N14C,N15C,	00031800
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00031900
	COMMON /XFRIC/ FST,FDY,GAP	00032000
	COMMON /CAPSUL/ RCPI,RCPO,AFNA	00032100
	COMMON /COMPLY/ YP,DDOD,TITLE(20)	00032200
	COMMON /IOFIL/ IO5,IO6,IO7	00032300
	DIMENSION EPR(NN,NTR) , EPC(NN,NTR) , EPZ(NN,NTR) , SIGR(NN,NTR) ,	00032400
*	SIGC(NN,NTR) , SIGZ(NN,NTR) , TPR(NN,NTR) , TPC(NN,NTR) ,	00032500
*	TPZ(NN,NTR) , TSR(NN,NTR) , TSC(NN,NTR) , TSZ(NN,NTR) ,	00032600
*	RDIS(NN,NTR2) , CRIF(NN) , RCP(NN,NCL1) , TCP(NN,NCL1) ,	00032700
*	TBCP(NN,NCL1) , ATBCP(NN,NCL1) , RCL(NN,NCL1) ,	00032800
*	TCL(NN,NCL1) , TBCL(NN,NCL1) , ATBCL(NN,NCL1) ,	00032900
*	YNG(NN,NTR) , POS(NN,NTR) , SHR(NN,NTR) , BM(NN,NTR) ,	00033000
*	RF(NN,NFR1) , TF(NN,NFR1) , TBF(NN,NFR) , ATBF(NN,NFR) ,	00033100
*	PRS(NN,NTR) , EHP(NN,NTR) , PRES(NTR1) , KFRIC(NN) ,	00033200
*	DPR(NTR) , DPC(NTR) , DPZ(NTR) , DSR(NTR) , DSC(NTR) ,	00033300
*	DSZ(NTR) , POSB(NN,2) , SHRB(NN,2) , SWG(NN,NFR) ,	00033400
*	SWS(NN,NTR) , RPM(NN,NP1) , TPM(NN,NP1) , GS(NN,NFR) ,	00033500
	DIMENSION CPU(NN,NP) , LM(NN,NTR) , LMR(NN,NTR) , LMC(NN,NTR) ,	00033600
*	LMZ(NN,NTR) , RCOO(NN)	00033700
C		00033800
C		00033900
C	DETAILED AND MATRIX RESULTANT OUTPUT	00034000
C		00034100
C		00034200
	WRITE(IO6,9000) N	00034300
	IF (RCPO-RCPI) 200,200,100	00034400
100	CONTINUE	00034500
C	CAPSULE OUTPUT	00034600
	WRITE(IO6,9110)	00034700
	WRITE(IO6,9210) (RCP(N,I),I=1,NCL1)	00034800
	WRITE(IO6,9220) (TCP(N,I),I=1,NCL1)	00034900
	WRITE(IO6,9310) (ATBCP(N,I),I=1,NCL1)	00035000
200	CONTINUE	00035100
C	CLAD OUTPUT	00035200
	WRITE(IO6,9120)	00035300
	WRITE(IO6,9210) (RCL(N,I),I=1,NCL1)	00035400
	WRITE(IO6,9240) (RDIS(N,I+1),I=NFR1,NTR1)	00035500
	WRITE(IO6,9220) (TCL(N,I),I=1,NCL1)	00035600
	J1 = NFR	00035700
	IF(N.EQ.NN) J1=0	00035800
	WRITE(IO6,9230) (PRES(J1+I),I=1,NCL1)	00035900
	WRITE(IO6,9310) (ATBCL(N,I),I=1,NCL1)	00036000
	JCOUNT = 0	00036100
400	IF(JCOUNT.GE.1) GO TO 500	00036200
	JJJ = 2	00036300

	KK = NFR1	00036400
	KKK = NTR	00036500
	KKKK = NCL	00036600
	GO TO 600	00036700
500	KK = 1	00036800
	KKK = NFR	00036900
	KKKK = NFR	00037000
	J1 = 0	00037100
	JJJ = 1	00037200
600	CONTINUE	00037300
	WRITE(IO6,9320) (POS(N,I),I=KK,KKK)	00037400
	WRITE(IO6,9330) (YNG(N,I),I=KK,KKK)	00037500
	IF (KK.GT.NFR) GO TO 650	00037600
	WRITE(IO6,9600)	00037700
	WRITE(IO6,9601) (LMR(N,I),I=KK,KKK)	00037800
	WRITE(IO6,9602) (LMC(N,I),I=KK,KKK)	00037900
	WRITE(IO6,9603) (LMZ(N,I),I=KK,KKK)	00038000
C		00038100
650	CONTINUE	00038200
	WRITE(IO6,9340)	00038300
	WRITE(IO6,9391) (SIGR(N,I),I=KK,KKK)	00038400
	WRITE(IO6,9392) (SIGC(N,I),I=KK,KKK)	00038500
	WRITE(IO6,9393) (SIGZ(N,I),I=KK,KKK)	00038600
	WRITE(IO6,9350)	00038700
	WRITE(IO6,9391) (EPR(N,I),I=KK,KKK)	00038800
	WRITE(IO6,9392) (EPC(N,I),I=KK,KKK)	00038900
	WRITE(IO6,9393) (EPZ(N,I),I=KK,KKK)	00039000
	WRITE(IO6,9360)	00039100
	WRITE(IO6,9391) (TPR(N,I),I=KK,KKK)	00039200
	WRITE(IO6,9392) (TPC(N,I),I=KK,KKK)	00039300
	WRITE(IO6,9393) (TPZ(N,I),I=KK,KKK)	00039400
	WRITE(IO6,9370)	00039500
	WRITE(IO6,9391) (DPR(J1+I),I=1,KKKK)	00039600
	WRITE(IO6,9392) (DPC(J1+I),I=1,KKKK)	00039700
	WRITE(IO6,9393) (DPZ(J1+I),I=1,KKKK)	00039800
	WRITE(IO6,9380)	00039900
	WRITE(IO6,9381) (TSR(N,I),I=KK,KKK)	00040000
	IF(JCOUNT.EQ.0) GO TO 700	00040100
	WRITE(IO6,9382) (SWS(N,I),I=1,NFR)	00040200
	WRITE(IO6,9383) (SWG(N,I),I=1,NFR)	00040300
	WRITE(IO6,9384) (EHP(N,I),I=1,NFR)	00040400
C		00040500
700	CONTINUE	00040600
C	WRITE(IO6,9385) (DSR(J1+I),I=1,KKKK)	00040700
	DDDD = (RCL(N,NCL1)-RCOO(N)) / RCOO(N) * 100.	00040800
	WRITE(IO6,9250) DDDD	00040900
	WRITE(IO6,9410) POSB(N,JJJ)	00041000
	YNGB = SHRB(N,JJJ)*2.*(1.+POSB(N,JJJ))	00041100
	WRITE(IO6,9420) YNGB	00041200
	IF(N.EQ.NN) GO TO 900	00041300
	IF(JCOUNT.GE.1) GO TO 800	00041400
	WRITE(IO6,9130)	00041500
	WRITE(IO6,9210) (RF(N,I),I=1,NFR1)	00041600
	WRITE(IO6,9240) (RDIS(N,I),I=1,NFR1)	00041700
	WRITE(IO6,9220) (TF(N,I),I=1,NFR1)	00041800
	WRITE(IO6,9230) (PRES(I),I=1,NFR1)	00041900
	WRITE(IO6,9310) (ATBF(N,I),I=1,NFR)	00042000
	JCOUNT = 1	00042100

	GO TO 400	00042200
800	WRITE(I06,9430) GAP	00042300
	IF (KFRIC(N)-1) 801,802,803	00042400
801	WRITE(I06,9700) CRIF(N)	00042500
	GO TO 804	00042600
802	WRITE(I06,9701) CRIF(N)	00042700
	GO TO 804	00042800
803	WRITE(I06,9702) CRIF(N)	00042900
804	CONTINUE	00043000
	WRITE(I06,9140)	00043100
	WRITE(I06,9510) (RPM(N,I),I=1,NP1)	00043200
	WRITE(I06,9520) (TPM(N,I),I=1,NP1)	00043300
	WRITE(I06,9530) (CPU(N,I),I=1,NP)	00043400
C		00043500
	RETURN	00043600
900	CONTINUE	00043700
C		00043800
C	TO GET THE CPU TIME ACCUMULATED (SEC)	00043900
C		00044000
	CALL TTIMER(TIMER)	00044100
C		00044200
	WRITE(I06,12321) TIMER	00044300
	RETURN	00044400
C		00044500
12321	FORMAT(1H0,'CPU TIME =',F7.2,' (SEC)')	00044600
9000	FORMAT(22H0*** AXIAL SECTION = , I2, 5H ***)	00044700
9110	FORMAT(29H0 *** CAPSULE (I=1,NCL) ***)	00044800
9120	FORMAT(29H0 *** CLAD (I=1,NCL) ***)	00044900
9130	FORMAT(29H0 *** FUEL (I=1,NFR) ***)	00045000
9140	FORMAT(40H0 *** PLUTONIUM MIGRATION (I=1,NP) ***)	00045100
9210	FORMAT(21H RADIUS(CM) , 7F15.5, (/, 1H ,20X, 7F15.5))	00045200
9220	FORMAT(19H TEMP.(C) , 7F15.3, (/, 1H ,18X, 7F15.3))	00045300
9230	FORMAT(21H PRESS.(KG/CM**2) , 7E15.5, (/, 1H ,20X, 7E15.5))	00045400
9240	FORMAT(21H DISPLACEMENT(CM) , 7F15.5, (/, 1H ,20X, 7F15.5))	00045500
9250	FORMAT(14X,'PERCENT CHANGE IN ORIGINAL O.D. =',F15.5,' (O/O)' /)	00045600
9310	FORMAT(27H ALPHA-T(-) , 7F15.5,(/, 1H ,26X,7F15.5))	00045700
9320	FORMAT(27H POISSON RATIO(-) , 7F15.5,(/, 1H ,26X,7F15.5))	00045800
9330	FORMAT(27H YOUNG M.(KG/CM**2) , 7E15.5,(/, 1H ,26X,7E15.5))	00045900
9340	FORMAT(23H STRESS(KG/CM**2))	00046000
9350	FORMAT(23H TOTAL STRAIN(-))	00046100
9360	FORMAT(23H CREEP STRAIN(-))	00046200
9370	FORMAT(23H CREEP IN DELT(-))	00046300
9380	FORMAT(23H SWELLING STRAIN(-))	00046400
9381	FORMAT(27H TOTAL , 7E15.5,(/, 1H ,26X,7E15.5))	00046500
9382	FORMAT(27H SOLID COMPONENT , 7E15.5,(/, 1H ,26X,7E15.5))	00046600
9383	FORMAT(27H GAS COMPONENT , 7E15.5,(/, 1H ,26X,7E15.5))	00046700
9384	FORMAT(27H HOT PRESSING , 7E15.5,(/, 1H ,26X,7E15.5))	00046800
9385	FORMAT(27H INCLEMENT IN DELT , 7E15.5,(/, 1H ,26X,7E15.5))	00046900
9391	FORMAT(27H RADIAL , 7E15.5,(/, 1H ,26X,7E15.5))	00047000
9392	FORMAT(27H CIRCUMFERENTIAL , 7E15.5,(/, 1H ,26X,7E15.5))	00047100
9393	FORMAT(27H AXIAL , 7E15.5,(/, 1H ,26X,7E15.5))	00047200
9410	FORMAT(37H AVERAGE POISSON RATIO(-) = , F15.5)	00047300
9420	FORMAT(37H0 AVERAGE YOUNG MODULUS(KG/CM**2) = , F15.5)	00047400
9430	FORMAT(37H0 GAP SIZE(CM) = , F15.5)	00047500
9440	FORMAT(37H FRICTION FORCE(KG/CM) = , F15.5)	00047600
9510	FORMAT(20H RADIUS(CM) , 11F10.5, (/, 1H , 19X, 11F10.5))	00047700
9520	FORMAT(18H TEMP.(C) , 11F10.3, (/, 1H , 17X, 11F10.3))	00047800
9530	FORMAT(15H PU CONC.(-) ,10X, 10F10.5, (/, 1H , 24X, 10F10.5))	00047900

9600	FORMAT(1H ,'	NUMBER OF CRACK ')		00048000
9601	FORMAT(27H	RADIAL	,I10,6I15,(/,1H ,21X,	00048100
	* 7I15))			00048200
9602	FORMAT(27H	CIRCUMFERENTIAL	,I10,6I15,(/,1H ,21X,	00048300
	* 7I15))			00048400
9603	FORMAT(27H	AXIAL	,I10,6I15,(/,1H ,21X,	00048500
	* 7I15))			00048600
9700	FORMAT(37H	FRICTION FORCE(KG/CM)	= , E15.5 ,	00048700
	* ' (OPEN)')			00048800
9701	FORMAT(37H	FRICTION FORCE(KG/CM)	= , E15.5 ,	00048900
	* ' (SLIP)')			00049000
9702	FORMAT(37H	FRICTION FORCE(KG/CM)	= , E15.5 ,	00049100
	* ' (STICK)')			00049200
	END			00049300

	SUBROUTINE EQUISI (SR,SC,SZ,ES,PRS,N)	00107300
	COMMON /ALWAYS/ NN,NCR,NER,NJR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00107400
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00107500
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00107600
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00107700
*	N12C,N13C,N14C,N15C,	00107800
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00107900
	DIMENSION SR(NN,NTR) , SC(NN,NTR) , SZ(NN,NTR) , ES(NN,NTR)	00108000
*	, PRS(NN,NTR)	00108100
C		00108200
C	EQUIVALENT STRESS CALC.	00108300
C		00108400
	DO 1000 I=1,NTR	00108500
	ES(N,I) = SQRT(((SR(N,I)-SC(N,I))**2 + (SR(N,I)-SZ(N,I))**2	00108600
*	+ (SC(N,I)-SZ(N,I))**2) / 2.0)	00108700
	PRS(N,I) = - (SR(N,I)+SC(N,I)+SZ(N,I)) / 3.0	00108800
1000	CONTINUE	00108900
C		00109000
	RETURN	00109100
	END	00109200

	SUBROUTINE FLCOND (XK,XK1,ZH,FP,FP95,T)	00049400
C		00049500
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00049600
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00049700
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00049800
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00049900
*	N12C,N13C,N14C,N15C,	00050000
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00050100
	COMMON /CONST/ FKA1,FKB1,FKC1,FKA2,FKB2,FKC2,FKA3,FKB3,FKC3,	00050200
*	FA11,FA12,FA21,FA22,FA31,FA32,	00050300
*	CK1,CK2,CA1,CA2,XEWTON(7)	00050400
C		00050500
C		00050600
C	FUEL THERMAL CONDUCTIVITY CALC.	00050700
C		00050800
C		00050900
	IF(NOK-2) 100,200,300	00051000
C		00051100
	100 Z = T+273.	00051200
	XK = (FKB1/Z + FKA1 + FKC1*Z**3) * FP	00051300
	XK1 = (-FKB1/Z**2 + 3.*FKC1*Z**2) * FP	00051400
	RETURN	00051500
C		00051600
	200 XK = FKA2 + 1./((FKB2-FKC2*ZH)*T)	00051700
	XK1 = -1./((FKB2-FKC2*ZH)*T*T)	00051800
	RETURN	00051900
C		00052000
	300 Z = T+273.	00052100
	A3 = FKA3*FP95/FP	00052200
	B3 = FKB3*FP95/FP	00052300
	C3 = FKC3*FP/FP95	00052400
C		00052500
C	BIANCHERIA OPTION	00052600
C		00052700
	XK = 1./((A3+B3*Z) + C3*Z**3	00052800
	XK1 = -B3/((A3+B3*Z)**2 + 3.*C3*Z**2	00052900
	RETURN	00053000
C	XK IN W/CM/DEG C	00053100
C	XK1 IN W/CM/DEG C /DEG C	00053200
C		00053300
	END	00053400

	SUBROUTINE FLTMP (POW,TCL,RCI,RUB,REB,RCB,RVB,RH,DXM,XM,R,T,	00053500
*	TB,ATB,TBA,ATBA,FD,RPM,TPM,TBPM,ATBPM,FDPM,GDT,	00053600
*	HG,F,OTOM,RHTHEO,DELT,RFO,IT,BU,RHOP,RB,	00053700
*	EPR,EPC,EPZ,TPR,TPC,TPZ,TSR,TSC,TSZ,GS,SWG,SWS,	00053800
*	EHP,WOLD,WNEW,XEPR,XEPC,XTPR,XTPC,XTPZ,XTSR,	00053900
*	XTSC,XTSZ,XGS,XSWG,XSWS,XEHP,XSWS3,SWS3)	00054000
C		00054100
	REAL*8 R1,R2,R3,R4,R1R,R2R,R3R,RHD1,RHD2,RHD3,XMD1,XMD2,XMD3	00054200
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLQT,NCRK,	00054300
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00054400
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00054500
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00054600
*	N12C,N13C,N14C,N15C,	00054700
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00054800
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00054900
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00055000
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00055100
	COMMON /CONST/ FKA1,FKB1,FKC1,FKA2,FKB2,FKC2,FKA3,FKB3,FKC3,	00055200
*	FA11,FA12,FA21,FA22,FA31,FA32,	00055300
*	CK1,CK2,CA1,CA2,XEWTON(7)	00055400
	COMMON /IOFIL/ IO5,IO6,IO7	00055500
	COMMON /FBRATR/ JFA	00055600
C		00055700
	REAL*8 XAR	00055800
	DIMENSION POW(NN) , TCL(NN,NCL1) , RCI(NN) , RUB(NN) ,	00055900
*	REB(NN) , RCB(NN) , RVB(NN) , RH(NN,NTR) , DXM(NN,3) ,	00056000
*	XM(NN,3) , R(NN,NFR1) , T(NN,NFR1) , TB(NN,NFR) ,	00056100
*	ATB(NN,NFR) , TBA(NN,3) , ATBA(NN,3) , FD(NN,NFR) ,	00056200
*	RPM(NN,NP1) , TPM(NN,NP1) , TBPM(NN,NP) , FDPM(NN,NP) ,	00056300
*	GDT(NN,NP1) , HG(NN) , ATBPM(NN,NP) , F(NN,NP) ,	00056400
*	OTOM(NN) , RHTHEO(NN) , RFO(NN,NFR1) , BU(NN) ,	00056500
*	RHOP(NN,3) , RB(NN,NTR)	00056600
	DIMENSION EPR(NN,NTR),EPC(NN,NTR),EPZ(NN,NTR),TPR(NN,NTR),	00056700
*	TPC(NN,NTR),TPZ(NN,NTR),TSR(NN,NTR),TSC(NN,NTR),	00056800
*	TSZ(NN,NTR),GS(NN,NFR),SWG(NN,NFR),SWS(NN,NTR),	00056900
*	EHP(NN,NTR),SWS3(NN,NUR)	00057000
	DIMENSION WOLD(NN,NFR),WNEW(NFR),XEPR(NFR),XEPC(NFR),XTPR(NFR),	00057100
*	XTPC(NFR),XTPZ(NFR),XTSR(NFR),XTSC(NFR),XTSZ(NFR),	00057200
*	XGS(NFR),XSWG(NFR),XSWS(NFR),XEHP(NFR),XSWS3(NUR)	00057300
C		00057400
C		00057500
C	FUEL TEMPERATURE DISTRIBUTION , RESTRUCTURING CALC.	00057600
C		00057700
C		00057800
	BT = 1.0	00057900
	IF(JFA.GE.2) BT = 0.5	00058000
C		00058100
	DO 1000 N=1,NNF	00058200
C		00058300
	P = POW(N)*8.6011E+02	00058400
C	P NOW IN CAL/CM/HR	00058500
	TO = TCL(N,1) + P/(HG(N)*2.*PI*RCI(N)) / 1.8	00058600
C	TO NOW IN DEG C	00058700
	W = POW(N)	00058800
C	W NOW IN WATTS/CM	00058900
C	RUB(N) NOW IN CM	00059000
C		00059100
	RHOT = RHTHEO(N)	00059200

C		00059300
	700 RO = RUB(N)	00059400
C		00059500
C	RO NOW IN CM	00059600
C		00059700
	R(N,NFR1) = RO	00059800
	T(N,NFR1) = TO	00059900
	RPM(N,NP1) = RO	00060000
	TPM(N,NP1) = TO	00060100
C		00060200
C	SUBDIVIDE EACH ZONE	00060300
C		00060400
	DR = (RCB(N)-RVB(N)) / NCR	00060500
C		00060600
	R(N,1) = RVB(N)	00060700
	DO 3100 I=2,NCR	00060800
	3100 R(N,I) = R(N,I-1) + DR	00060900
C		00061000
	DR = (REB(N)-RCB(N)) / NER	00061100
C		00061200
	K = NCR+2	00061300
	KK = NCR+NER	00061400
	R(N,NCR+1) = RCB(N)	00061500
	DO 3200 I=K,KK	00061600
	3200 R(N,I) = R(N,I-1) + DR	00061700
C		00061800
	DR = (RUB(N)-REB(N)) / NUR	00061900
C		00062000
	K = NCR+NER+2	00062100
	KK = NFR	00062200
	R(N,NCR+NER+1) = REB(N)	00062300
	DO 3300 I=K,KK	00062400
	3300 R(N,I) = R(N,I-1) + DR	00062500
C		00062600
	IF(IT.NE.1) GO TO 3330	00062700
C		00062800
	DO 3333 I=1,NFR1	00062900
	3333 RFO(N,I) = R(N,I)	00063000
	3330 CONTINUE	00063100
C		00063200
	DO 3400 I=1,NFR	00063300
	RB(N,I) = 2.*(R(N,I+1)*R(N,I+1)+R(N,I+1)*R(N,I)+R(N,I)*R(N,I)) /	00063400
	* (3.*(R(N,I+1)+R(N,I)))	00063500
	3400 CONTINUE	00063600
C		00063700
	XML(1) = 0.	00063800
	XML(2) = XM(N,1)	00063900
	XML(3) = XM(N,1)+XM(N,2)	00064000
	SM = XM(N,1)+XM(N,2)+XM(N,3)	00064100
	SF = 0.	00064200
C		00064300
	DO 1100 J=1,NP	00064400
	1100 SF = SF+F(N,J)	00064500
C		00064600
	NPNP = NP * 2	00064700
	DMO = SM/NP	00064800
C		00064900
	DMO = SM/NPNP	00065000
	XML = SM	

	RR = RO	00065100
	TT = TO	00065200
	WW = W	00065300
C		00065400
	I = 3	00065500
C		00065600
	NM = NP	00065700
	DDM = DMO	00065800
C		00065900
300	DM1 = XM(N,I)	00066000
	SAT = 0.	00066100
C		00066200
	GO TO (310,320,330) , I	00066300
310	J = 1	00066400
	P = 1. - (RHOP(N,I) + (0.99*RHOT - RHOP(N,I)) *BU(N)/BURH1)/RHOT	00066500
	GO TO 340	00066600
320	J = NCR+1	00066700
	P = 1. - (RHOP(N,I) + (0.965*RHOT - RHOP(N,I)) *BU(N)/BURH2)/RHOT	00066800
	GO TO 340	00066900
330	J = NCR+NER+1	00067000
	P = 1. - RHOP(N,I)/RHOT	00067100
340	CONTINUE	00067200
C		00067300
	FP = (1.-P)/(1.+BT*P)	00067400
	FP95 = (1.-0.05)/(1.+BT*0.05)	00067500
C		00067600
400	CONTINUE	00067700
	IF(DDM-DM1) 200,100,100	00067800
100	IF(I-2) 110,120,130	00067900
110	RI = RVB(N)	00068000
	GO TO 140	00068100
120	RI = RCB(N)	00068200
	GO TO 140	00068300
130	RI = REB(N)	00068400
140	DM = DM1	00068500
	ICHECK = 1	00068600
	GO TO 500	00068700
C		00068800
200	DM = DDM	00068900
	XAR = RR**2 - DM/(PT*RH(N,J))	00069000
	IF(XAR.LT.0.) XAR = 0.	00069100
	RI = DSQRT(XAR)	00069200
	ICHECK = 0	00069300
C		00069400
500	DR = RR - RI	00069500
	W1 = W*NP*F(N,NM)/(SF*SM)	00069600
	W3 = W1*RH(N,J)	00069700
	DW = W1*DM	00069800
	WW = WW - DW	00069900
C		00070000
C	FUEL THERMAL CONDUCTIVITY CALCULATION	00070100
C		00070200
C		00070300
C	OTOM MODIFY	00070400
C		00070500
	CALL OTMCAL (OTOM,N)	00070600
C		00070700
	CALL COEF (OTOM,N)	00070800

C		00070900
C	THERMAL CONDUCTIVITY ITERATION LOOP	00071000
C		00071100
	JTEMP = 0	00071200
	TTT = TT	00071300
	540 CONTINUE	00071400
C		00071500
	TCR = 1.+ALF0(I)*(TTT-20.)+ALF1(I)*(TTT*TTT-400.)/2.	00071600
C	TCR = 1.+ALF0(I)*(TT-20.)+ALF1(I)*(TT*TT-400.)/2.	00071700
	ZH = RH(N,J)/(RHOT*TCR)	00071800
C		00071900
	CALL FLCOND(XK,XK1,ZH,FP,FP95,TTT)	00072000
C	CALL FLCOND (XK,XK1,ZH,FP,FP95,TT)	00072100
C		00072200
	IF(WW.LE.0.) WW=0.0	00072300
C	IF(WW.LE.0.) GO TO 550	00072400
	IF(RI.LE.0.001) GO TO 550	00072500
	DT = W3*(RR**2-RI**2)/(4.*XK)-(W3*RI**2/(2.*XK)-WW/(PI2*XK))*	00072600
	* ALOG(RR/RI)	00072700
	GO TO 560	00072800
	550 DT = W3*RR**2/(4.*XK)	00072900
	560 CONTINUE	00073000
	IF(JTEMP.EQ.2) GO TO 570	00073100
	TTT = TT+DT/2.	00073200
	JTEMP = JTEMP + 1	00073300
	GO TO 540	00073400
	570 CONTINUE	00073500
C		00073600
C	END OF THERMAL CONDUCTIVITY LOOP	00073700
C		00073800
	TBPM(N,NM) = TT + DT/2.	00073900
	ATBPM(N,NM) = ATBFUN (TT,DT,I)	00074000
	SAT = SAT + ATBPM(N,NM)*DM	00074100
	FDPM(N,NM) = F(N,NM)*NP*RH(N,J) / (SM*SF)	00074200
C	FDPM NOW IN FISSIONS*CM/FISSIONS/CM**3	00074300
C		00074400
	XM1 = XM1 - DM	00074500
	RR = RI	00074600
	TT = TT + DT	00074700
C		00074800
	IF(ICHECK.EQ.0) GO TO 600	00074900
	ATBA(N,I) = SAT/XM(N,I)	00075000
	IF(RR.LE.0.) GO TO 825	00075100
	I = I - 1	00075200
	IF(I.GE.1) GO TO 831	00075300
	GO TO 829	00075400
	825 DO 826 JJ=1,I	00075500
	ATBA(N,JJ) = ATBA(N,I)	00075600
	826 CONTINUE	00075700
	829 CONTINUE	00075800
	DO 830 JJ=1,NM	00075900
	TPM(N,JJ) = TT	00076000
	RPM(N,JJ) = RR	00076100
	TBPM(N,JJ) = TBPM(N,NM)	00076200
	ATBPM(N,JJ) = ATBPM(N,NM)	00076300
	FDPM(N,JJ) = FDPM(N,NM)	00076400
	830 GDI(N,JJ) = -DT/DR	00076500
	GO TO 860	00076600

C		00076700
831	GRDT = DT/DR	00076800
	Y = 0.1*(TT-T0)/R0	00076900
	IF(GRDT-Y) 835,840,840	00077000
835	GRDT = Y	00077100
840	TK = TT+273.	00077200
	IF(GRDT.LE.0.) GRDT=1.E-06	00077300
C		00077400
	GO TO (841,842) , I	00077500
841	CONTINUE	00077600
	DRCB = CM1(I)*(TK**(-1.5)) * EXP(-DHR(I)/TK) * GRDT	00077700
	CM(I) = PI2*RR*RH(N,J)*DRCB	00077800
	GO TO 843	00077900
842	CONTINUE	00078000
	DREB = CM1(I)*EXP(-DHR(I)/TK)	00078100
	CM(I) = PI2*RR*RH(N,J)*DREB	00078200
843	CONTINUE	00078300
	DDM = DDM-DM	00078400
	IF(DDM.GT.0.) GO TO 850	00078500
	TPM(N,NM) = TT	00078600
	RPM(N,NM) = RR	00078700
	GDT(N,NM) = -DT/DR	00078800
	NM = NM-1	00078900
	IF(NM.GE.1) GO TO 300	00079000
	GO TO 7700	00079100
850	IF(I.GE.1) GO TO 300	00079200
	GO TO 860	00079300
600	TPM(N,NM) = TT	00079400
	RPM(N,NM) = RR	00079500
	GDT(N,NM) = -DT/DR	00079600
	IF(RR.LE.0.) GO TO 650	00079700
	NM = NM - 1	00079800
	DM1 = DM1 - DM	00079900
	DDM = DMO	00080000
	IF(NM.GE.1) GO TO 400	00080100
	GO TO 7700	00080200
650	DO 660 JJ=1,NM	00080300
	TPM(N,JJ) = TT	00080400
	RPM(N,JJ) = RR	00080500
	GDT(N,JJ) = -DT/DR	00080600
660	CONTINUE	00080700
7700	GRDT = DT/DR	00080800
	Y = 0.1*(TT-T0)/R0	00080900
	IF(GRDT-Y) 7835,7840,7840	00081000
7835	GRDT = Y	00081100
7840	TK = TT+273.	00081200
	IF(GRDT.LE.0.) GRDT=1.E-06	00081300
	IF(I-2) 790,720,710	00081400
710	DREB = CM1(2)*EXP(-DHR(2)/TK)	00081500
	CM(2) = PI2*RR*RH(N,NCR+1)*DREB	00081600
	ATBA(N,I)=SAT/(XM(N,I)-DM1)	00081700
	DM1=0.	00081800
	SAT=ATBPM(N,I-1)	00081900
	I=I-1	00082000
720	DRCB = CM1(1)*(TK**(-1.5))*EXP(-DHR(1)/TK)*GRDT	00082100
	CM(1) = PI2*RR*RH(N,1)*DRCB	00082200
	ATBA(N,I)=SAT/(XM(N,I)-DM1)	00082300
	DM1=0.	00082400

	SAT=ATBPM(N,I-1)	00082500
	I=I-1	00082600
790	CONTINUE	00082700
	ATBA(N,I)=SAT/(XM(N,I)-DM1)	00082800
C		00082900
C		00083000
860	CONTINUE	00083100
	T(N,1) = TT	00083200
	FD(N,1) = FDP(N,1)	00083300
C		00083400
C		00083500
C	INTERPOLATION	00083600
C		00083700
	IJ = 2	00083800
	DO 1400 K=2,NFR	00083900
1401	IF(R(N,K)-RPM(N,IJ)) 1403,1403,1402	00084000
1402	IJ = IJ + 1	00084100
	IF(IJ.GE.NPI) GO TO 1404	00084200
	GO TO 1401	00084300
1403	RAT = (RPM(N,IJ)-R(N,K)) / (RPM(N,IJ)-RPM(N,IJ-1))	00084400
	T(N,K) = TPM(N,IJ) + (TPM(N,IJ-1)-TPM(N,IJ)) * RAT	00084500
	FD(N,K) = FDP(N,IJ) + (FDP(N,IJ-1)-FDP(N,IJ)) * RAT	00084600
	GO TO 1400	00084700
1404	FD(N,K) = FDP(N,NP)	00084800
	T(N,K) = TPM(N,NP1) + (TPM(N,NP)-TPM(N,NP1)) * (RPM(N,NP1)-	00084900
	* R(N,K)) / (RPM(N,NP1)-RPM(N,NP))	00085000
	IJ = NP	00085100
1400	CONTINUE	00085200
C		00085300
	DO 1405 K=1,NFR	00085400
	TB(N,K) = (T(N,K)+T(N,K+1)) / 2.0	00085500
1405	CONTINUE	00085600
C		00085700
	DO 1500 IJ=1,3	00085800
	ST = 0.	00085900
	IF (IJ-2) 1501,1502,1503	00086000
1501	IJK1 = 1	00086100
	IJK2 = NCR	00086200
	GO TO 1504	00086300
1502	IJK1 = NCR + 1	00086400
	IJK2 = NCR + NER	00086500
	GO TO 1504	00086600
1503	IJK1 = NCR + NER + 1	00086700
	IJK2 = NFR	00086800
1504	CONTINUE	00086900
C		00087000
	DO 1550 IJK=IJK1,IJK2	00087100
	DDT = (T(N,IJK)-T(N,IJK+1))	00087200
	ATB(N,IJK) = ATBFUN(T(N,IJK+1),DDT,IJ)	00087300
1550	ST = ST + TB(N,IJK)	00087400
	TBA(N,IJ) = ST / (IJK2-IJK1+1)	00087500
1500	CONTINUE	00087600
C		00087700
C		00087800
	DXM(N,3) = -CM(2)	00087900
	DXM(N,2) = CM(2)-CM(1)	00088000
	DXM(N,1) = CM(1)	00088100
C		00088200

C		00088300
C		00088400
C	REGION MASS CALC. AND MODIFICATION	00088500
C		00088600
C		00088700
C		00088800
C	4100 CONTINUE	00088900
C		00089000
	DXM(N,1) = DXM(N,1) / (PI2*RH(N,1)*RVB(N))	00089100
	DXM(N,2) = DRCB	00089200
	DXM(N,3) = DREB	00089300
C		00089400
	DO 6100 I=1,NFR	00089500
6100	WNEW(I) = PI*RH(N,I)*(R(N,I+1)**2-R(N,I)**2)*(1.+EPZ(N,I))	00089600
	IF (IT.LE.1) GO TO 6900	00089700
	J = NFR	00089800
	I = NFR	00089900
	XEPR(J) = 0.	00090000
	XEPC(J) = 0.	00090100
	XTPR(J) = 0.	00090200
	XTPC(J) = 0.	00090300
	XTPZ(J) = 0.	00090400
	XTSR(J) = 0.	00090500
	XTSC(J) = 0.	00090600
	XTSZ(J) = 0.	00090700
	XGS(J) = 0.	00090800
	XSWG(J) = 0.	00090900
	XSWS(J) = 0.	00091000
	XEHP(J) = 0.	00091100
6200	CONTINUE	00091200
	W = WNEW(J)	00091300
6250	IF(WOLD(N,I)-W) 6400,6300,6300	00091400
6300	DW = W/WNEW(J)	00091500
	XEPR(J) = XEPR(J) + DW*EPR(N,I)	00091600
	XEPC(J) = XEPC(J) + DW*EPC(N,I)	00091700
	XTPR(J) = XTPR(J) + DW*TPR(N,I)	00091800
	XTPC(J) = XTPC(J) + DW*TPC(N,I)	00091900
	XTPZ(J) = XTPZ(J) + DW*TPZ(N,I)	00092000
	XTSR(J) = XTSR(J) + DW*TSR(N,I)	00092100
	XTSC(J) = XTSC(J) + DW*TSC(N,I)	00092200
	XTSZ(J) = XTSZ(J) + DW*TSZ(N,I)	00092300
	XGS(J) = XGS(J) + DW*GS(N,I)	00092400
	XSWG(J) = XSWG(J) + DW*SWG(N,I)	00092500
	XSWS(J) = XSWS(J) + DW*SWS(N,I)	00092600
	XEHP(J) = XEHP(J) + DW*EHP(N,I)	00092700
	WOLD(N,I) = WOLD(N,I)-W	00092800
	J = J-1	00092900
	IF(J.LE.0) GO TO 6850	00093000
	XEPR(J) = 0.	00093100
	XEPC(J) = 0.	00093200
	XTPR(J) = 0.	00093300
	XTPC(J) = 0.	00093400
	XTPZ(J) = 0.	00093500
	XTSR(J) = 0.	00093600
	XTSC(J) = 0.	00093700
	XTSZ(J) = 0.	00093800
	XGS(J) = 0.	00093900
	XSWG(J) = 0.	00094000

	XSWJ(J) = 0.	00094100
	XEHP(J) = 0.	00094200
	GO TO 6200	00094300
6400	DW = WOLD(N,I)/WNEW(J)	00094400
	XEPR(J) = XEPR(J) + DW*EPR(N,I)	00094500
	XEPC(J) = XEPC(J) + DW*EPC(N,I)	00094600
	XTPR(J) = XTPR(J) + DW*TPR(N,I)	00094700
	XTPC(J) = XTPC(J) + DW*TPC(N,I)	00094800
	XTPZ(J) = XTPZ(J) + DW*TPZ(N,I)	00094900
	XTSR(J) = XTSR(J) + DW*TSR(N,I)	00095000
	XTSC(J) = XTSC(J) + DW*TSC(N,I)	00095100
	XTSZ(J) = XTSZ(J) + DW*TSZ(N,I)	00095200
	XGS(J) = XGS(J) + DW*GS(N,I)	00095300
	XSWG(J) = XSWG(J) + DW*SWG(N,I)	00095400
	XSWJ(J) = XSWJ(J) + DW*SWS(N,I)	00095500
	XEHP(J) = XEHP(J) + DW*EHP(N,I)	00095600
	W = W - WOLD(N,I)	00095700
	I = I - 1	00095800
	IF(I.GE.1) GO TO 6250	00095900
	DW = W/WNEW(J)	00096000
	XEPR(J) = XEPR(J)+DW*EPR(N,I)	00096100
	XEPC(J) = XEPC(J)+DW*EPC(N,I)	00096200
	XTPR(J) = XTPR(J)+DW*TPR(N,I)	00096300
	XTPC(J) = XTPC(J)+DW*TPC(N,I)	00096400
	XTPZ(J) = XTPZ(J)+DW*TPZ(N,I)	00096500
	XTSR(J) = XTSR(J)+DW*TSR(N,I)	00096600
	XTSC(J) = XTSC(J)+DW*TSC(N,I)	00096700
	XTSZ(J) = XTSZ(J)+DW*TSZ(N,I)	00096800
	XGS(J) = XGS(J)+DW*GS(N,I)	00096900
	XSWG(J) = XSWG(J)+DW*SWG(N,I)	00097000
	XSWJ(J) = XSWJ(J)+DW*SWS(N,I)	00097100
	XEHP(J) = XEHP(J)+DW*EHP(N,I)	00097200
	J = J - 1	00097300
	IF(J.LE.0).GO TO 6850	00097400
	DO 6450 I = 1,J	00097500
	XEPR(I) = EPR(N,I)	00097600
	XEPC(I) = EPC(N,I)	00097700
	XTPR(I) = TPR(N,I)	00097800
	XTPC(I) = TPC(N,I)	00097900
	XTPZ(I) = TPZ(N,I)	00098000
	XTSR(I) = TSR(N,I)	00098100
	XTSC(I) = TSC(N,I)	00098200
	XTSZ(I) = TSZ(N,I)	00098300
	XGS(I) = GS(N,I)	00098400
	XSWG(I) = SWG(N,I)	00098500
	XSWJ(I) = SWS(N,I)	00098600
	XEHP(I) = EHP(N,I)	00098700
6450	CONTINUE	00098800
6850	DO 6550 I=1,NFR	00098900
	EPR(N,I) = XEPR(I)	00099000
	EPC(N,I) = XEPC(I)	00099100
	TPR(N,I) = XTPR(I)	00099200
	TPC(N,I) = XTPC(I)	00099300
	TPZ(N,I) = XTPZ(I)	00099400
C	TSR(N,I) = XTSR(I)	00099500
C	TSC(N,I) = XTSC(I)	00099600
C	TSZ(N,I) = XTSZ(I)	00099700
C	GS(N,I) = XGS(I)	00099800

C	SWG(N,I) = XSWG(I)	00099900
C	SWS(N,I) = XSWS(I)	00100000
C	EHP(N,I) = XEHP(I)	00100100
	6550 CONTINUE	00100200
	6900 DO 6950 I=1,NFR	00100300
	6950 WOLD(N,I) = WNEW(I)	00100400
	1000 CONTINUE	00100500
C		00100600
	4200 CONTINUE	00100700
C		00100800
	RETURN	00100900
	END	00101000

	SUBROUTINE FLXCAL (POWAV, QSFF, ZFLX, FLNC, FLUX, DELT)	00101100
	COMMON /ALWAYS/ NN, NCR, NER, NUR, NCL, NP, NOK, NOS, NOC, NPLOT, NCRK,	00101200
*	NTR, NFR, NTR1, NTR2, NNTR2, NNF, NCL1, NFR1, NP1,	00101300
*	IGEN, IGEN1, NONZR, IGENP, IGENP1, NONZRP,	00101400
*	N1C, N2C, N3C, N4C, N5C, N6C, N7C, N8C, N9C, N10C, N11C,	00101500
*	N12C, N13C, N14C, N15C,	00101600
*	N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, N11, N12, N13, N14, N15	00101700
	COMMON /OTHERS/ PG(3), CM1(2), DHR(2), ALFO(3), ALF1(3), XML(3), CM(2),	00101800
*	PI, PI2, DP, GRAV, SMQ, SMQO, RHOC, BURH1, BURH2, CFLX,	00101900
*	CPDW, BUCNS, RK, RKR, A91, GAS1, TCON, PDBU, BUB	00102000
C		00102100
	DIMENSION QSFF(NN) , ZFLX(NN) , FLNC(NN) , FLUX(NN)	00102200
C		00102300
C	FLUX , FLUENCE CALC.	00102400
C		00102500
	DO 1000 N=1, NN	00102600
	FLUX(N) = CFLX*QSFF(N)*POWAV*SMQO/SMQ	00102700
	FLNC(N) = FLNC(N) + FLUX(N)*DELT*3600.	00102800
1000	CONTINUE	00102900
C		00103000
	RETURN	00103100
	END	00103200

	SUBROUTINE FRICHK (RUB,RCI,DISP,PRES,FRIC,CRIF,KFRIC,KONV,IFRIC, 00103300
*	NS) 00103400
C	00103500
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK, 00103600
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1, 00103700
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP, 00103800
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C, 00103900
*	N12C,N13C,N14C,N15C, 00104000
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15 00104100
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2), 00104200
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX, 00104300
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,POBU,BUB 00104400
	COMMON /XFRIC/ FST,FDY,GAP 00104500
C	00104600
	DIMENSION RUB(NN) , RCI(NN) , DISP(NTR2) , PRES(NTR1) , 00104700
*	CRIF(NN) , KFRIC(NN) 00104800
C	00104900
C	00105000
C	00105100
C	FRICITION FORCE CHECK 00105200
C	00105300
C	00105400
	IF (IFRIC-1) 100,200,300 00105500
C	00105600
100	CONTINUE 00105700
C	00105800
	GAP = (RCI(NS)+DISP(NFR+2))-(RUB(NS)+DISP(NFR+1)) 00105900
C	00106000
	IF (GAP) 102,102,101 00106100
101	CONTINUE 00106200
C	00106300
C	GAP OPEN 00106400
C	00106500
	IFRIC = 0 00106600
	GO TO 300 00106700
C	00106800
102	CONTINUE 00106900
	GAP = 0. 00107000
	IFRIC = 2 00107100
C	IFRIC = 1 00107200
	KONV = 0 00107300
	RETURN 00107400
C	00107500
200	CONTINUE 00107600
C	00107700
	GAP = 0. 00107800
	GRIC = PI2*(RUB(NS)+DISP(NFR+1))*PRES(NFR+1)*FST 00107900
	IF(GRIC.LE.0.) GO TO 202 00108000
	IF (FRIC-GRIC) 201,201,202 00108100
C	00108200
201	CONTINUE 00108300
C	00108400
C	STICK 00108500
C	00108600
	IFRIC = 1 00108700
	GO TO 300 00108800
202	CONTINUE 00108900
C	00109000

PNCT841-75-17

C	SLIP	00109100
C		00109200
	IFRIC = 2	00109300
	KONV = 0	00109400
	RETURN	00109500
300	CONTINUE	00109600
	KONV = 1	00109700
	KFRIC(NS) = IFRIC	00109800
	CRIF(NS) = FRIC	00109900
C		00110000
	RETURN	00110100
	END	00110200

	SUBROUTINE GAPCON (RCI,RUB,HG)	00000100
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLDT,NCRK,	00000200
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00000300
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00000400
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00000500
*	N12C,N13C,N14C,N15C,	00000600
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00000700
	DIMENSION RCI(NN) , RUB(NN) , HG(NN)	00000800
C		00000900
C	FUEL-CLADDING GAP CONDUCTANCE CALC.	00001000
C		00001100
	DO 1000 N=1,NN	00001200
	RO = RCI(N)	00001300
	RI = RUB(N)	00001400
	X = RO-RI	00001500
	IF(X.LT.0.) X=0.0	00001600
C	HG(N) = 1873.*EXP(-135.2*X) / 144.	00001700
C	HG NOW IN BTU/HR/IN/IN/DEGF	00001800
C		00001900
C		00002000
	HG(N) = 508.05 * EXP(-53.23*X)	00002100
C		00002200
C	HG NOW IN CAL/HR/CM/CM/DEG F	00002300
C		00002400
	1000 CONTINUE	00002500
	RETURN	00002600
	END	00002700

	SUBROUTINE GASOUT (RF,FLZ,POW,FD,TBF,VOL,EPR,EPC,EPZ,ATBF,SWS,	00002800
	* GR,GS,GRT,DELT)	00002900
C		00003000
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLDT,NCRK,	00003100
	* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00003200
	* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00003300
	* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00003400
	* N12C,N13C,N14C,N15C,	00003500
	* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00003600
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00003700
	* PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00003800
	* CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00003900
	COMMON /XGASO/ AF,AN,CT,QR	00004000
C		00004100
	DIMENSION RF(NN,NFR1) , FLZ(NN) , POW(NN) , FD(NN,NFR) ,	00004200
	* TBF(NN,NFR) , VOL(NN,NTR) , EPR(NN,NTR) , EPC(NN,NTR) ,	00004300
	* EPZ(NN,NTR) , ATBF(NN,NFR) , SWS(NN,NTR) , GR(NN) ,	00004400
	* GS(NN,NFR)	00004500
C		00004600
C	GAS RELEASE RATE CALC.	00004700
C		00004800
C		00004900
C	AF IS ATOMS OF GAS PER FISSION	00005000
C	AN IS AVOGADRO'S NUMBER	00005100
C		00005200
C	FRPT IS FRACTION OF GAS RELEASED PER HOUR	00005300
C		00005400
C	GR(N) = GAS RELEASED FROM AXIAL SECTION N IN DELT	00005500
C	GS(N,I) = GAS REMAINING IN AXIAL SECTION N , RADIAL REGION I	00005600
C	GRT = TOTAL GAS RELEASED	00005700
C		00005800
	DO 1000 N=1,NNF	00005900
	DFN = A91*POW(N)*DELT	00006000
	GR(N) = 0.	00006100
C		00006200
	DO 1100 I=1,NFR	00006300
	T = TBF(N,I)*1.8+32.0 + 460.0	00006400
	FRPT = EXP(-CT-QR/T)	00006500
	DGS = AF*VOL(N,I)*FD(N,I)*DFN/AN	00006600
	G0 = GS(N,I)	00006700
	EG = EPR(N,I)+EPC(N,I)+EPZ(N,I)-3.*ATBF(N,I)-SWS(N,I)	00006800
	EHR = 0.	00006900
	IF(EG-0.25) 200,200,100	00007000
	100 EHR = (EG-0.25)*100.	00007100
	200 F = FRPT + EHR*0.0005	00007200
	X = F*DELT	00007300
	IF(X-0.1) 300,400,400	00007400
	300 Y = (1.-X/3.+X*X/12.)*X/2.	00007500
	Z = X-X*Y	00007600
	REL = G0*Z + DGS*Y	00007700
	GS(N,I) = G0+DGS-REL	00007800
	GO TO 1100	00007900
	400 IF(X-170.) 600,500,500	00008000
	500 GS(N,I) = DGS/X	00008100
	GO TO 700	00008200
	600 GS(N,I) = DGS/X + (G0-DGS/X)*EXP(-X)	00008300
	700 REL = G0 + DGS - GS(N,I)	00008400
	1100 GR(N) = GR(N) + REL	00008500

PNCT841-75-17

C		00008600
	GRT = GRT + GR(N)	00008700
C		00008800
	1000 CONTINUE	00008900
C		00009000
	RETURN	00009100
C		00009200
	END	00009300

	SUBROUTINE GOLDN (POWAV, QSF, TOUT, TIN, RCLO, POW, TCLO, RCPI,	00009400
*	XCPO, AFNA, FL, HC, TNA, TCPO, TCPI, TOUTER, NSKIP, PO,	00009500
*	FLZ)	00009600
	COMMON /ALWAYS/ NN, NCR, NER, NUR, NCL, NP, NOK, NOS, NOC, NPLOT, NCRK,	00009700
*	NTR, NFR, NTR1, NTR2, NNTR2, NNF, NCL1, NFR1, NP1,	00009800
*	IGEN, IGEN1, NONZR, IGENP, IGENP1, NONZRP,	00009900
*	N1C, N2C, N3C, N4C, N5C, N6C, N7C, N8C, N9C, N10C, N11C,	00010000
*	N12C, N13C, N14C, N15C,	00010100
*	N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, N11, N12, N13, N14, N15	00010200
	COMMON /OTHERS/ PG(3), CM1(2), DHR(2), ALF0(3), ALF1(3), XML(3), CM(2),	00010300
*	PI, PI2, DP, GRAV, SMQ, SMQ0, RHOC, BURH1, BURH2, CFLX,	00010400
*	CPOW, BUCNS, RK, RKR, A91, GAS1, TCON, PDBU, BUB	00010500
	COMMON /XGOLD/ ASSO, ASS1, ANAO, ANA1, ANA2, CO, C1, C2	00010600
	COMMON /FBRATR/ JFA	00010700
	COMMON /IOFIL/ IO5, IO6, IO7	00010800
	DIMENSION QSF(NN) , TCLO(NN) , POW(NN) , RCLO(NN) , HC(NN) ,	00010900
*	TNA(NN) , TCPO(NN) , TCPI(NN) , TOUTER(NN) , FLZ(NN)	00011000
C		00011100
C		00011200
C	CLADDING OUTER SURFACE TEMPERATURE CALC.	00011300
C		00011400
C		00011500
C	=====	00011600
C	CALCULATES CLAD OD TEMPERATURE FOR ENCAPSULATED (RCPI.LT.XCPO),	00011700
C	AND UNENCAPSULATED (RCPI.GE.XCPO) FUEL ELEMENTS.	00011800
C	TEMPERATURE DROPS IN NA ARE DETERMINED BY NEWTONIAN EXTRAPOLATION	00011900
C	(2-3 AND 8-9). ***	00012000
C	CAPSULE IS ASSUMED TO BE STAINLESS STEEL	00012100
C	=====	00012200
C		00012300
	AF=AFNA	00012400
	WB=POWAV*8.6011E+02	00012500
C		00012600
C	WB IS AVERAGE POWER IN CAL/HR/CM	00012700
C		00012800
	TINF = 1.8*TIN + 32.	00012900
	TOUTF = 1.8*TOUT + 32.	00013000
	B0=C0+(C1+C2*TINF)*TINF	00013100
	B1=C1+2.*C2*TINF	00013200
	B2=C2	00013300
	DTMP=TOUTF-TINF	00013400
	CPDT=(B0+(B1/2.+B2*DTMP/3.)*DTMP)*DTMP	00013500
	WDOT=WB*FL/CPDT	00013600
	JNX=0	00013700
	IF (XCPO-RCPI) 110,110,120	00013800
110	JNX=1	00013900
120	RCPO=XCPO	00014000
	IF(NSKIP) 121,121,122	00014100
121	CONTINUE	00014200
	TNA(NN)=TOUT	00014300
	TCPO(NN)=TOUT	00014400
	TCPI(NN)=TOUT	00014500
	TCLO(NN)=TOUT	00014600
	GO TO 123	00014700
122	CONTINUE	00014800
	TNA(NN) = TOUTER(NN)	00014900
	TCPO(NN) = TOUTER(NN)	00015000
	TCPI(NN) = TOUTER(NN)	00015100

	TCLO(NN) = TOUTER(NN)	00015200
123	CONTINUE	00015300
	POW(NN)=0.	00015400
	XNF=NNF	00015500
	S=0.	00015600
	SPW = 0.	00015700
C	=====	00015800
	DO 130 N=1,NNF	00015900
	SPW = SPW + QSF(N)	00016000
130	S=S+QSF(N)*FLZ(N)	00016100
	IF(NSKIP.GT.0) GO TO 135	00016200
	X=0.	00016300
	SW=0.	00016400
C	=====	00016500
C		00016600
135	CONTINUE	00016700
	DO 1000 N=1,NN	00016800
C		00016900
	IF (NSKIP.GT.0) TNA(N) = N*(TOUT-TIN) / NN + TIN	00017000
	IF(NSKIP.GT.0) GO TO 155	00017100
	SW=SW+QSF(N)*FLZ(N)	00017200
	C=CPDT*(SW-QSF(N)*FLZ(N)/2.)/S	00017300
140	CP=80+(B1+B2*X)*X	00017400
	X1=(C+(B1/2.+B2*X/1.5)*X*X)/CP	00017500
	EPS=ABS(X1-X)/X1	00017600
	X=X1	00017700
	IF (EPS-.00001) 150,140,140	00017800
150	T=TINF+X	00017900
	TNA(N)=(T-32.) / 1.8	00018000
	IF(JFA.GE.2) TNA(N) = N *(TOUT-TIN)/NN+TIN	00018100
C==	THIS IS NA TEMPERATURE AT AXIAL NODE N =====	00018200
155	CONTINUE	00018300
	W=WB*QSF(N)*XNF/SPW	00018400
	POW(N)=W / 8.6011E+02	00018500
C		00018600
C	POW(N) NOW IN WATT/CM	00018700
C		00018800
	IF (JNX) 170,170,160	00018900
160	RCPO=RCLO(N)	00019000
170	P=PI2*RCPO	00019100
	IF(NSKIP.GT.0) GO TO 195	00019200
	GO TO (2100,2200,2300,2200),JFA	00019300
2100	DE = 4.*AF/P	00019400
	CNST=4.*WDOT/P	00019500
	CNST1=W/RCPO/PI2	00019600
	TB=T	00019700
	Y=0.	00019800
180	XK=ANA0+(ANA1+ANA2*TB)*TB	00019900
	CP=C0+(C1+C2*TB)*TB	00020000
	PE=CNST*CP/XK	00020100
C==	THIS IS PECLET NO. =====	00020200
	XNU=5.0+0.025*(PE**0.8)	00020300
C==	THIS IS NUSSELT NO. =====	00020400
	HC(N) = XNU*XK/DE	00020500
C	HC NOW IN CAL/HR/CM**2/DEG F	00020600
	Y1=CNST1/HC(N)	00020700
	IF(Y1.LE.0.) GO TO 190	00020800
	TB=T+Y1/2.	00020900

	EPS=ABS(Y1-Y)/Y1	00021000
	Y=Y1	00021100
	IF (EPS-.0001) 190,180,180	00021200
190	T5=T+Y	00021300
	GO TO 196	00021400
2200	HFLUX = POW(N)/P	00021500
	IF(HFLUX.LE.0.) GO TO 2210	00021600
	DTFILM = 7.93*(HFLUX**0.25)*EXP(-PO/63.2)	00021700
C	DTFILM = HFLUX/1.1443	00021800
	T5 = (TNA(N)+DTFILM)*1.8+32.0	00021900
	HC(N) = HFLUX/DTFILM*1762.3*252.0/929.03	00022000
C	HC NOW IN CAL/HR/CM**2/DEG F	00022100
	GO TO 196	00022200
2210	HC(N) = 0.0	00022300
	T5 = TNA(N)*1.8+32.0	00022400
	GO TO 196	00022500
C		00022600
C	\$\$\$\$ FILM COEFFICIENT OF PWR (JFA = 3) \$\$\$	00022700
C	* * * (DITTUS - BOETLER EQ.) * * *	00022800
2300	CONTINUE	00022900
C	*** INTEGRATED HEAT CAPACITY IS IN KCAL/KG	00023000
	DTMP = TOUT - TIN	00023100
	CPDT = 2.623*DTMP - 0.01473*DTMP**2/2. + 3.528E-05*DTMP**3/3.	00023200
C	*** FLP IS IN M	00023300
	FLP = FL/100.0	00023400
C	*** WBP IS IN KCAL/HR/M	00023500
	WBP = WB/10.	00023600
C	*** AFNAP IS IN M**2	00023700
	AFNAP = AFNA/10000.0	00023800
C	*** GAMMAP IS IN KG/M**3	00023900
	GAMMAP = 860.8 + 1.042*TNA(N) - 0.00514*TNA(N)**2	00024000
C	*** WP IS IN M/HR	00024100
	WP = FLP*WBP/(CPDT*AFNAP*GAMMAP)	00024200
C	*** DEP IS IN M	00024300
	DEP = 4.*AFNA/(PI2*RCPO*100.)	00024400
C	*** ENUP IS IN M**2/HR	00024500
	ENUP =(0.4015E-06 - 1.791E-09*TNA(N) + 2.946E-12*TNA(N)**2)*3600.	00024600
C	*** REYNOL'S NUMBER	00024700
	REP = WP*DEP/ENUP	00024800
C	*** PRANDTLE'S NUMBER	00024900
	PRP = 5.491 - 0.0381*TNA(N) + 0.777E-04*TNA(N)**2	00025000
C	*** THERMAL CONDUCTIVITY IS IN KCAL/M/HR/C	00025100
	RAMDAP = 0.4847 + 1.596E-03*TNA(N) - 5.497E-06*TNA(N)**2	00025200
C	*** FILM COEFFICIENT IS IN KCAL/HR/M**2/C	00025300
	HCP = 0.023*RAMDAP/DEP*REP**0.8*PRP**0.4	00025400
	HC(N) = HCP/18.	00025500
C	*** HC IS IN CAL/HR/CM**2/F	00025600
	HCP = HCP*1.1622E-04	00025700
C	*** HCP IS IN W/CM**2/C	00025800
	T5 = (TNA(N)+POW(N)/(PI2*RCPO*HCP))*1.8 + 32.0	00025900
	GO TO 196	00026000
C		00026100
195	CONTINUE	00026200
	T5 = TOUTER(N)*1.8 + 32.0	00026300
	HC(N) = 0.0	00026400
196	CONTINUE	00026500
	TCPO(N)=(T5-32.) / 1.8	00026600
C==	THIS IS CAPSULE OD TEMPERATURE =====	00026700

	IF (JNX) 210,210,200	00026800
200	TCLO(N)=TCPO(N)	00026900
	GO TO 1000	00027000
210	W1=2.*W/PI2	00027100
	XL=ALOG(RCPO/RCPI)	00027200
	XKO=(ASSO+ASS1*T5)	00027300
	T4=T5+W1*XL/(XKO+SQRT(XKO*XKO+ASS1*W1*XL))	00027400
	TCPI(N)=(T4-32.) / 1.8	00027500
C==	THIS IS CAPSULE ID TEMPERATURE =====	00027600
	IF (RCPI-RCLO(N)) 220,220,230	00027700
220	WRITE (IO6,270) N	00027800
C		00027900
	STOP	00028000
C		00028100
230	C=W*ALOG(RCPI/RCLO(N))/PI2	00028200
	D0=ANA0+(ANA1+ANA2*T4)*T4	00028300
	D1=ANA1+2.*ANA2*T4	00028400
	D2=ANA2	00028500
	Y=0.	00028600
240	XK=D0+(D1+D2*Y)*Y	00028700
	Y1=(C+(D1/2.+D2*Y/1.5)*Y*Y)/XK	00028800
	IF (Y1.LE.0.) GO TO 250	00028900
	EPS=ABS(Y1-Y)/Y1	00029000
	Y=Y1	00029100
	IF (EPS-.0001) 250,240,240	00029200
250	TCLO(N)=(T4+Y-32.) / 1.8	00029300
C==	THIS IS CLAD OD TEMPERATURE =====	00029400
C		00029500
	1000 CONTINUE	00029600
C		00029700
C		00029800
	270 FORMAT(1H1,'----- SUBROUTINE GOLDN -----', / ,	00029900
	* 1H0,' FUEL-CLAD HAS HIT CAPSULE WALL IN SECTION',I2,'.'/,	00030000
	*2X,'EXECUTION CONTINUATION IS NONSENSE')	00030100
	RETURN	00030200
	END	00030300

	SUBROUTINE HEADER	00030400
C		00030500
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK	00030600
	COMMON /COMPLT/ YP,DDOD,TITL(20)	00030700
	COMMON /IOFIL/ IO5,IO6,IO7	00030800
	DIMENSION TITLE(20)	00030900
	DATA CEND/'*END'/	00031000
C		00031100
C	TITLE CARD READ TILL FOUND *END CARD	00031200
C		00031300
	ISW = 0	00031400
	100 READ(IO5,9000,END=300) TITLE	00031500
	IF (TITLE(1) . EQ . CEND) GO TO 200	00031600
	WRITE(IO6,9001) TITLE	00031700
	IF (ISW.NE.0) GO TO 100	00031800
	DO 101 I=1,20	00031900
	TITL(I) = TITLE(I)	00032000
	101 CONTINUE	00032100
	ISW = 1	00032200
	GO TO 100	00032300
C		00032400
	300 WRITE(IO6,9002)	00032500
	STOP	00032600
C		00032700
	200 CONTINUE	00032800
C		00032900
	RETURN	00033000
C		00033100
C	FORMAT	00033200
C		00033300
	9000 FORMAT(20A4)	00033400
	9001 FORMAT(1H0,20X,20A4)	00033500
	9002 FORMAT(1H1,'----- SUBROUTINE HEADER -----', / ,	00033600
	* 1H0,'*END CARD MISSING (*END CARD MEANS TITLE CARD END)')	00033700
C		00033800
	END	00033900

	SUBROUTINE HKNS (TBCL,TSR,TSC,TSZ,FLX,DELT)	00034000
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00034100
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00034200
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00034300
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00034400
*	N12C,N13C,N14C,N15C,	00034500
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00034600
C		00034700
	COMMON /XHKN5/ CS,B0,C0,C1,C2,C3	00034800
	COMMON /FBRATR/ JFA	00034900
	DIMENSION TBCL(NN,NCL) , TSR(NN,NTR) , TSC(NN,NTR) ,	00035000
*	TSZ(NN,NTR) , FLX(NN)	00035100
C		00035200
C		00035300
C	CLADDING SWELLING STRAIN CALC.	00035400
C		00035500
C		00035600
	DO 1000 N=1,NN	00035700
	FLUX = FLX(N)	00035800
C		00035900
	DO 2000 I=1,NCL	00036000
	TX = TBCL(N,I)	00036100
	SW = TSR(N,NFR+I)	00036200
C		00036300
	IF(JFA-2) 10,20,20	00036400
10	CONTINUE	00036500
	IF(NOS+2) 200,200,100	00036600
100	TK = TX + 273.	00036700
C	TX IN DEG C , TK NOW IN DEG K	00036800
	FX = FLUX/1.2E+22	00036900
	SITA = TK-623.+1.E-50	00037000
	IF(SITA.LT.CS) SITA = CS	00037100
	XN = 2.05-27./SITA+78./SITA**2	00037200
	SDH = (FX*3600.)**XN*((TK-40.)*1.E-10)*EXP(-0.015*TK-5100./TK+	00037300
*	32.6) / 300.	00037400
	GO TO 300	00037500
C		00037600
200	XN = 1.50	00037700
	TC = TX	00037800
C	TX IN DEG C , TC NOW IN DEG C	00037900
	X = ALOG(B0/300.) + XN*ALOG(FLUX*3600.)	00038000
	SDH = EXP(X) * (C0+TC*(C1+TC*(C2+TC*C3)))	00038100
	IF(SDH.LT.0.) SDH = 0.	00038200
300	CONTINUE	00038300
C		00038400
	TSR(N,NFR+I) = (SW**(1./XN) + SDH**(1./XN)*DELT)**XN	00038500
	GO TO 30	00038600
C		00038700
20	TSR(N,NFR+I) = SW	00038800
30	CONTINUE	00038900
	TSC(N,NFR+I) = TSR(N,NFR+I)	00039000
	TSZ(N,NFR+I) = TSR(N,NFR+I)	00039100
2000	CONTINUE	00039200
1000	CONTINUE	00039300
	RETURN	00039400
	END	00039500

	SUBROUTINE INPUT { EPZ,TPR,TPC,TPZ,SIGR,SIGC,SIGZ,ADISP,RDISP,	00039600
*	RVB,RCB,REB,RUB,RCI,RCO,QSFF,QSFP,F,EPR,EPC,	00039700
*	TSR,TSC,TSZ,CPU,RHOP,OTOM,FL,CL,PTOP,PO,EPSIN,	00039800
*	CLZO }	00039900
C		00040000
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00040100
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00040200
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00040300
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00040400
*	N12C,N13C,N14C,N15C,	00040500
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00040600
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00040700
*	PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,	00040800
*	CPDW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00040900
	COMMON /CAPSUL/ RCPI,RCPO,AFNA	00041000
	COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00041100
	COMMON /IOFIL/ IO5,IO6,IO7	00041200
	DIMENSION TPR(NN,NTR),TPC(NN,NTR),TPZ(NN,NTR),SIGR(NN,NTR),	00041300
*	SIGC(NN,NTR),SIGZ(NN,NTR),ADISP(NN,NTR),RDISP(NN,NTR2),	00041400
*	RVB(NN),RCB(NN),REB(NN),RUB(NN),RCI(NN),RCO(NN),	00041500
*	QSFF(NN),QSFP(NN),F(NN,NP),EPZ(NN,NTR),EPR(NN,NTR),	00041600
*	EPC(NN,NTR),TSR(NN,NTR),TSC(NN,NTR),TSZ(NN,NTR),	00041700
*	CPU(NN,NP),RHOP(NN,3),OTOM(NN),CLZO(NN)	00041800
C		00041900
C		00042000
C	READ CARD 4 TO CARD 23	00042100
C		00042200
C		00042300
C		00042400
C----	CARD 4 ... SKIP AND PLOT OPTION	00042500
C		00042600
	READ(IO5,9000) IDLE,IPFUL,IPSTN,IPSTS,IPCRK	00042700
C		00042800
C		00042900
C	IDLE = INITIAL VALUES SET OPTION	00043000
C	O/I = CARD INPUT/ZERO SET BY CODE	00043100
C	IPFUL = AXIAL SECTION NUMBER FOR FUEL PLOT	00043200
C	IPSTN = AXIAL SECTION NUMBER FOR STRAIN PLOT	00043300
C	IPSTS = AXIAL SECTION NUMBER FOR STRESS PLOT	00043400
C	IPCRK = AXIAL SECTION NUMBER FOR CRACK PLOT	00043500
C		00043600
	IF(IDLE) 100,100,200	00043700
100	CONTINUE	00043800
	READ(IO5,9001) EPR	00043900
	READ(IO5,9001) EPC	00044000
	READ(IO5,9001) EPZ	00044100
	READ(IO5,9001) TPR	00044200
	READ(IO5,9001) TPC	00044300
	READ(IO5,9001) TPZ	00044400
	READ(IO5,9001) TSR	00044500
	READ(IO5,9001) TSC	00044600
	READ(IO5,9001) TSZ	00044700
	READ(IO5,9001) SIGR	00044800
	READ(IO5,9001) SIGC	00044900
	READ(IO5,9001) SIGZ	00045000
	READ(IO5,9001) ADISP	00045100
	READ(IO5,9001) RDISP	00045200
	GO TO 300	00045300

C		00045400
	200 CONTINUE	00045500
C		00045600
	DO 1000 N=1,NN	00045700
	DO 1050 M=1,NTR	00045800
	EPR(N,M) = 0.0	00045900
	EPC(N,M) = 0.0	00046000
	EPZ(N,M) = 0.0	00046100
	TPR(N,M) = 0.0	00046200
	TPC(N,M) = 0.0	00046300
	TPZ(N,M) = 0.0	00046400
	TSR(N,M) = 0.0	00046500
	TSC(N,M) = 0.0	00046600
	TSZ(N,M) = 0.0	00046700
	SIGR(N,M) = 0.0	00046800
	SIGC(N,M) = 0.0	00046900
	SIGZ(N,M) = 0.0	00047000
1050	ADISP(N,M) = 0.0	00047100
	DO 1000 M=1,NTR2	00047200
	RDISP(N,M) = 0.0	00047300
1000	CONTINUE	00047400
C		00047500
	300 CONTINUE	00047600
C		00047700
C----	CARD 5 ... CENTRAL VOID RADIUS (CM) 1 - NN	00047800
	READ(I05,9001) RVB	00047900
C----	CARD 6 ... COLUMNAR GRAIN REGION BOUNDARY (CM) 1 - NN	00048000
	READ(I05,9001) RCB	00048100
C----	CARD 7 ... EQUIAXED GRAIN REGION BOUNDARY (CM) 1 - NN	00048200
	READ(I05,9001) REB	00048300
C----	CARD 8 ... FUEL OUTER RADIUS (CM) 1 - NN	00048400
	READ(I05,9001) RUB	00048500
C----	CARD 9 ... CLAD INNER RADIUS (CM) 1 - NN	00048600
	READ(I05,9001) RCI	00048700
C----	CARD 10 ... CLAD OUTER RADIUS (CM) 1 - NN	00048800
	READ(I05,9001) RCO	00048900
C----	CARD 11 ... CLAD AXIAL LENGTH (CM) 1 - NN	00049000
	READ(I05,9001) CLZO	00049100
C----	CARD 12 ... AXIAL FLUX RATE (-) 1 - NN	00049200
	READ(I05,9001) QSFF	00049300
C----	CARD 13 ... AXIAL POWER RATE (-) 1 - NN	00049400
	READ(I05,9001) QSFP	00049500
C----	CARD 14 ... Q / M (INPUT X OF PU-Q(X)) 1 - NNF	00049600
	READ(I05,9001) OTOM	00049700
C		00049800
C----	CARD 15 ... PULUTONIUM CONCENTRATION (WEIGHT RATE) 1 - NNF	00049900
	READ(I05,9001) (CPU(N,1),N=1,NNF)	00050000
C		00050100
C----	CARD 16 ... DENSITY OF UNDISTURBED REGION (G/CC) 1 - NNF	00050200
	READ(I05,9001) (RHOP(N,3),N=1,NNF)	00050300
C		00050400
C----	CARD 17 ... RADIAL HEAT GENERATION RATE (-) 1 - NP , 1 - NNF	00050500
	READ(I05,9001) (F(1,I),I=1,NP)	00050600
	IF(NNF.LE.1) GO TO 2001	00050700
	DO 2000 N=2,NNF	00050800
	DO 1999 I=1,NP	00050900
	F(N,I) = F(1,I)	00051000
1999	CONTINUE	00051100

2000 CONTINUE	00051200
2001 CONTINUE	00051300
C	00051400
C	00051500
C---- CARD 18 ... CAPSULE INNER RADIUS, OUTER RADIUS, COOLANT FLOW AREA	00051600
READ(I05,9001) RCPI,RCPO,AFNA	00051700
C---- CARD 19 ... SURFACE TENSION OF VOID IN FUEL 3 REGIONS	00051800
C FROM 1 TO 3 (KG/CM**2)	00051900
READ(I05,9001) (PG(I),I=1,3)	00052000
C---- CARD 20 ... AXIAL FORCE ON CLADDING (KG) ,	00052100
C COOLANT PRESSURE (KG/CM**2)	00052200
READ(I05,9001) PTOP,PO	00052300
C---- CARD 21 ... CLAD DENSITY (G/CC) , DENSITY MODIFIER BY BURN-UP	00052400
READ(I05,9001) RHOC,BURH1,BURH2	00052500
C---- CARD 22 ... CONSTANTS FOR BOUNDARY VELOCITY	00052600
READ(I05,9001) (CMI(L),L=1,2),(DHR(L),L=1,2)	00052700
C---- CARD 23 ... INITIAL GAS MOLE IN PLENUM (MOLE)	00052800
READ(I05,9001) GASI,TCON,PDBU	00052900
C	00053000
RETURN	00053100
C	00053200
C FORMAT	00053300
C	00053400
9001 FORMAT(5E15.8)	00053500
9000 FORMAT(16I5)	00053600
C	00053700
END	00053800

	SUBROUTINE IOSET(IO)	00053900
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00054000
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00054100
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00054200
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00054300
*	N12C,N13C,N14C,N15C,	00054400
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00054500
C		00054600
C	LOGICAL UNIT SET FOR WORK FILE	00054700
C		00054800
	DIMENSION IO(NN)	00054900
	DO 100 I=1,NN	00055000
	IO(I) = I+10	00055100
100	CONTINUE	00055200
	RETURN	00055300
	END	00055400

SUBROUTINE ITERAT (PRS,VOL,TBF,ATBF,SWG ,EHP,RH,RF,GS,SWS,	00055500
* BM,EPR,EPC,EPZ,SWS3,SIGR,SIGC,SIGZ,RHTHEO,	00055600
* DELT,N,RHOP,BU,TSR,TSC,TSZ,XEHP,XSWG ,	00055700
* PLPR,CVPR)	00055800
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00055900
* NTR,NFR,NTRI,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00056000
* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00056100
* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00056200
* N12C,N13C,N14C,N15C,	00056300
* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00056400
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00056500
* PI,PI2,DP,GRAY,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00056600
* CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00056700
COMMON /XITER/ C,QRK	00056800
DIMENSION PRS(NN,NTR) , VOL(NN,NTR) , TBF(NN,NFR) ,	00056900
* ATBF(NN,NFR) , SWG(NN,NFR) , EHP(NN,NTR) , RH(NN,NFR),	00057000
* RF(NN,NFR1) , GS(NN,NFR) , SWS(NN,NTR) , BM(NN,NTR) ,	00057100
* EPR(NN,NTR) , EPC(NN,NTR) , EPZ(NN,NTR) ,	00057200
* SWS3(NN,NUR) , SIGR(NN,NTR) , SIGC(NN,NTR) ,	00057300
* SIGZ(NN,NTR) , RHTHEO(NN) , RHOP(NN,3) , BU(NN) ,	00057400
* TSR(NN,NTR) , TSC(NN,NTR) , TSZ(NN,NTR) , XEHP(NFR) ,	00057500
* XSWG(NFR)	00057600
C	00057700
C	00057800
C FUEL SWELLING STRAIN , HOT-PRESSING CALC.	00057900
C	00058000
C	00058100
C RK IS IDEAL GAS CONSTANT IN KG*CM/MOLE/DEG K	00058200
C GS IS FISSION GAS IN MOLES CALCULATED IN GASOUT SUBROUTINE	00058300
C BM IS BULK MODULUS	00058400
C	00058500
C	00058600
C K = 1	00058700
C J = 1	00058800
C	00058900
C RHOT = RHTHEO(N)	00059000
C	00059100
C DO 1100 I=1,NFR	00059200
C T = TBF(N,I) + 273.	00059300
C	00059400
C	00059500
C IF (I.GT.NCR) K=2	00059600
C IF (I.GT.NCR+NER) K=3	00059700
C	00059800
C IF(K-2) 100,100,110	00059900
C	00060000
C SWELLING CAUSED FROM FISSION SOLID	00060100
C	00060200
C 110 SWS(N,I) = SWS3(N,J) + 2.389*GS(N,I)/VOL(N,I)*2.54*2.54*2.54	00060300
C J = J+1	00060400
C 100 CONTINUE	00060500
C	00060600
C	00060700
C SWELLING CAUSED FROM HOT PRESSING	00060800
C	00060900
C GO TO (210,220,230) , K	00061000
C 210 P = 1. - (RHOP(N,I) + (0.99*RHOT - RHOP(N,I)) *BU(N)/BURH1)/RHOT	00061100
C GO TO 240	00061200

220	P = 1. - (RHOP(N,I) +(0.965*RHOT - RHOP(N,I)) *BU(N)/BURH2)/RHOT	00061300
	GO TO 240	00061400
230	P = 1. - RHOP(N,I)/RHOT	00061500
240	CONTINUE	00061600
	XEHP(I) = EHP(N,I)	00061700
	XSWG(I) = SWG(N,I)	00061800
C	TEMP = -(C/T)*PRS(N,I)*(P + XEHP(I))*DELT*	00061900
C	TEMP = -(C/T)*PRS(N,I)*(P+ XSWG(I)+XEHP(I))*DELT*	00062000
C	* EXP(-QRK/T)	00062100
	TEMP = - C/T * PRS(N,I) * P * DELT * EXP(-QRK/T)	00062200
	IF(TEMP.GT.0.) TEMP=0.	00062300
	EHP(N,I) = XEHP(I) + TEMP	00062400
	IF(EHP(N,I).LT.-P) EHP(N,I)=-P	00062500
C		00062600
C		00062700
C	SWELLING CAUSED FROM FISSION GAS	00062800
C		00062900
C		00063000
	P = PG(K)	00063100
	TEMP = PRS(N,I)	00063200
	IF(TEMP.LT.0.0) TEMP=0.0	00063300
	PPP = P + TEMP	00063400
	IF(PPP.LE.CVPR) PPP = CVPR	00063500
	SWG(N,I) = GS(N,I)*RK*T / (VOL(N,I)*(PPP))	00063600
C		00063700
C	SWG(N,I) = GS(N,I)*RK*T / (VOL(N,I)*(P+TEMP))	00063800
C	SWG(N,I) = GS(N,I)*RK*T / (VOL(N,I)*P)	00063900
C		00064000
	TSR(N,I) = (SWS(N,I)+SWG(N,I)+EHP(N,I)) / 3.	00064100
	TSC(N,I) = TSR(N,I)	00064200
	TSZ(N,I) = TSR(N,I)	00064300
C		00064400
	1100 CONTINUE	00064500
C		00064600
	RETURN	00064700
	END	00064800

	SUBROUTINE LENGTH (FL,FLN,CL,CLN,FLZO,FLZ,CLZO,CLZ,EPZ)	00064900
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00065000
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00065100
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00065200
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00065300
*	N12C,N13C,N14C,N15C,	00065400
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00065500
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00065600
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00065700
*	CPDW,BUCNS,RK,RKR,A91,GAST,TCON,PDBU,BUB	00065800
C		00065900
	DIMENSION EPZ(NN,NTR) , FLZ(NN) , FLZO(NN) , CLZ(NN) ,	00066000
*	CLZO(NN)	00066100
C		00066200
C	AXIAL LENGTH CALC.	00066300
C		00066400
	FL = 0.	00066500
	CL = 0.	00066600
C		00066700
	DO 1000 N=1,NNF	00066800
	FLZ(N) = FLZO(N)*(1.+EPZ(N,1))	00066900
	FL = FL+FLZ(N)	00067000
	CLZ(N) = CLZO(N)*(1.+EPZ(N,NFR+1))	00067100
	CL = CL+CLZ(N)	00067200
C	CLZ(N) = FLZ(N)	00067300
	1000 CONTINUE	00067400
C		00067500
	CL = CL + CLZO(NN)*(1.+EPZ(NN,NFR+1))	00067600
	CLZ(NN) = CLZO(NN)*(1.+EPZ(NN,NFR+1))	00067700
	FLN = FL	00067800
	CLN = CL	00067900
C		00068000
C		00068100
	RETURN	00068200
	END	00068300

SUBROUTINE MATRX (IA,JA,A,E,P,DEQP,EQSIG,RF,RC,G,PB,GB,FLZ,	00000100
* KFRIC,MOUNT,IFRIC,NS,RB)	00000200
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00000300
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00000400
* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00000500
* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00000600
* N12C,N13C,N14C,N15C,	00000700
* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00000800
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00000900
* PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00001000
* CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00001100
COMMON /XFRIC/ FST,FDY	00001200
COMMON /IOFIL/ IO5,IO6,IO7	00001300
DIMENSION IA(IGEN1),JA(NONZR),A(NONZR),E(NN,NTR),	00001400
* P(NN,NTR),DEQP(NN,NTR),EQSIG(NN,NTR),	00001500
* RF(NN,NFR1),RC(NN,NCL1),G(NN,NTR),PB(NN,2),	00001600
* GB(NN,2),FLZ(NN),KFRIC(NN),RB(NN,NTR)	00001700
C	00001800
C	00001900
C ELEMENTS A OF AX=B, AND IA,JA FOR USE IN THE SUBSEQUENT	00002000
C CALL TO THE SL-MATH SUBROUTINES	00002100
C	00002200
C	00002300
IF (MOUNT) 20,20,10	00002400
C	00002500
10 J = 56*NTR + 20	00002600
GO TO 3000	00002700
C	00002800
20 CONTINUE	00002900
C	00003000
J = 0	00003100
K=1	00003200
C 1 - N	00003300
DO 1000 I=K,N1	00003400
XE = E(NS,I)	00003500
XP = P(NS,I)	00003600
J = J + 1	00003700
IA(I) = J	00003800
JA(J) = I	00003900
A(J) = 1.0	00004000
J = J + 1	00004100
JA(J) = I + N2	00004200
A(J) = -1./XE	00004300
J = J + 1	00004400
JA(J) = I + N3	00004500
A(J) = XP/XE	00004600
J = J + 1	00004700
JA(J) = I + N4	00004800
A(J) = XP/XE	00004900
J = J + 1	00005000
JA(J) = I + N5	00005100
A(J) = -1.0	00005200
1000 CONTINUE	00005300
C	00005400
C N+1 - 2N	00005500
C	00005600
DO 1100 I=K,N1	00005700
XE = E(NS,I)	00005800

XP = P(NS,I)	00005900
J = J + 1	00006000
IA(I+N1) = J	00006100
JA(J) = I + N1	00006200
A(J) = 1.0	00006300
J = J + 1	00006400
JA(J) = I + N2	00006500
A(J) = XP/XE	00006600
J = J + 1	00006700
JA(J) = I + N3	00006800
A(J) = -1./XE	00006900
J = J + 1	00007000
JA(J) = I + N4	00007100
A(J) = XP/XE	00007200
J = J + 1	00007300
JA(J) = I + N6	00007400
A(J) = -1.0	00007500
1100 CONTINUE	00007600
C	00007700
C 2N+1 - 3N	00007800
C	00007900
DO 1200 I=K,N1	00008000
XEP = DEQP(NS,I)	00008100
XES = EQSIG(NS,I)	00008200
J = J + 1	00008300
IA(I+N2) = J	00008400
JA(J) = I + N2	00008500
A(J) = XEP/XES	00008600
J = J + 1	00008700
JA(J) = I + N3	00008800
A(J) = -XEP/(2.*XES)	00008900
J = J + 1	00009000
JA(J) = I + N4	00009100
A(J) = -XEP/(2.*XES)	00009200
J = J + 1	00009300
JA(J) = I + N5	00009400
A(J) = -1.0	00009500
1200 CONTINUE	00009600
C	00009700
C 3N+1 - 4N	00009800
C	00009900
DO 1300 I=K,N1	00010000
XEP = DEQP(NS,I)	00010100
XES = EQSIG(NS,I)	00010200
J = J + 1	00010300
IA(I+N3) = J	00010400
JA(J) = I + N2	00010500
A(J) = -XEP/(2.*XES)	00010600
J = J + 1	00010700
JA(J) = I + N3	00010800
A(J) = XEP/XES	00010900
J = J + 1	00011000
JA(J) = I + N4	00011100
A(J) = -XEP/(2.*XES)	00011200
J = J + 1	00011300
JA(J) = I + N6	00011400
A(J) = -1.0	00011500
1300 CONTINUE	00011600

C		00011700
C	4N+1 - 5N	00011800
C		00011900
	DO 1400 I=K,N1	00012000
	XEP = DEQP(NS,I)	00012100
	XES = EQSIG(NS,I)	00012200
	J = J + 1	00012300
	IA(I+N4) = J	00012400
	JA(J) = I + N2	00012500
	A(J) = -XEP/(2.*XES)	00012600
	J = J + 1	00012700
	JA(J) = I + N3	00012800
	A(J) = -XEP/(2.*XES)	00012900
	J = J + 1	00013000
	JA(J) = I + N4	00013100
	A(J) = XEP/XES	00013200
	J = J + 1	00013300
	JA(J) = I + N7	00013400
	A(J) = -1.0	00013500
	1400 CONTINUE	00013600
C		00013700
C		00013800
C	5N+1 - 5N+3	00013900
C		00014000
	DO 1800 I=1,3	00014100
	J = J + 1	00014200
	IA(I+N5) = J	00014300
	DO 1850 IJ=1,NFR	00014400
	RO = RF(NS,IJ+1)	00014500
	RI = RF(NS,IJ)	00014600
	JA(J) = N5+(I-1)*NTR + IJ	00014700
	A(J) = (RO+RI)*(RO-RI)	00014800
	J = J + 1	00014900
	1850 CONTINUE	00015000
	JA(J) = N8 + 1 + 2*(I-1)	00015100
	RO = RF(NS,NFR1)	00015200
	RI = RF(NS,1)	00015300
	A(J) = -(RO+RI)*(RO-RI)	00015400
	1800 CONTINUE	00015500
C		00015600
C	5N+4 - 5N+6	00015700
C		00015800
	DO 1900 I=1,3	00015900
	J = J + 1	00016000
	IA(I+N5+3) = J	00016100
	DO 1950 IJ=1,NCL	00016200
	RO = RC(NS,IJ+1)	00016300
	RI = RC(NS,IJ)	00016400
	JA(J) = N5 + (I-1)*NTR + NFR + IJ	00016500
	A(J) = (RO+RI)*(RO-RI)	00016600
	J = J + 1	00016700
	1950 CONTINUE	00016800
	JA(J) = N8 + 2 + 2*(I-1)	00016900
	RO = RC(NS,NCL1)	00017000
	RI = RC(NS,1)	00017100
	A(J) = -(RO+RI)*(RO-RI)	00017200
	1900 CONTINUE	00017300
. C		00017400

C	5N+7 - 7N+6	00017500
C		00017600
	K=0	00017700
C	INNER DISPLACEMENT	00017800
	DO 2000 I=1,N1	00017900
	J = J + 1	00018000
	IA(I+N5+6) = J	00018100
	IF(I-NFR) 640,640,650	00018200
640	RD = RF(NS,I+1)	00018300
	RI = RF(NS,I)	00018400
	GO TO 660	00018500
650	K = K+1	00018600
	RD = RC(NS,K+1)	00018700
	RI = RC(NS,K)	00018800
660	CONTINUE	00018900
	XP = P(NS,I)	00019000
	XG = G(NS,I)	00019100
	RR =(RO+RI)*(RO-RI)	00019200
C	RA = RB(NS,I)	00019300
	RA = RI	00019400
	XY1 = (1.-2.*XP)*RA*ALOG(RA/RI) / (2.*(1.-XP))	00019500
	XY2 = RO*RO*(RI*RI+(1.-2.*XP)*RA*RA)*ALOG(RO/RI) /	00019600
	* (2.*RR*(1.-XP)*RA)	00019700
	JA(J) = I	00019800
	A(J) = RA-RI	00019900
	J = J+1	00020000
	JA(J) = I + N5	00020100
	A(J) = -RA*(1.-2.*XP)/2. -XY1+XY2	00020200
	J = J + 1	00020300
	JA(J) = I + N6	00020400
	A(J) = -RA*(1.-2.*XP)/2. +XY1-XY2	00020500
	J = J + 1	00020600
	A(J) = RA*XP	00020700
	IF(I-NFR) 670,670,680	00020800
670	JA(J) = N8 + 7	00020900
	J = J + 1	00021000
	JA(J) = N8 + 8 + I	00021100
	A(J) = 1.	00021200
	GO TO 690	00021300
680	JA(J) = N8 + 8	00021400
	J = J + 1	00021500
	JA(J) = N8 + 9 + I	00021600
	A(J) = 1.	00021700
690	CONTINUE	00021800
	J = J + 1	00021900
	JA(J) = N9 + 10 + I	00022000
	A(J) = -RI*RI*(RO*RO/RA+(1.-2.*XP)*RA) / (2.*XG*RR)	00022100
	J = J + 1	00022200
	JA(J) = N9 + 11 + I	00022300
	A(J) = RO*RO*(RI*RI/RA+(1.-2.*XP)*RA) / (2.*XG*RR)	00022400
2000	CONTINUE	00022500
C		00022600
C	OUTER DISPLACEMENT	00022700
C		00022800
	K=0	00022900
	DO 2100 I=1,N1	00023000
	J = J + 1	00023100
	IA(I+N6+6) = J	00023200

	IF(I-NFR) 700,700,710	00023300
700	RD = RF(NS,I+1)	00023400
	RI = RF(NS,I)	00023500
	GO TO 720	00023600
710	K = K + 1	00023700
	RD = RC(NS,K+1)	00023800
	RI = RC(NS,K)	00023900
720	CONTINUE	00024000
	XP = P(NS,I)	00024100
	XG = G(NS,I)	00024200
	RR = (RD+RI)*(RD-RI)	00024300
C	RA = RB(NS,I)	00024400
	RA = RD	00024500
	XY1 = (1.-2.*XP)*RA*ALOG(RA/RI) / (2.*(1.-XP))	00024600
	XY2 = RD*RD*(RI*RI+(1.-2.*XP)*RA*RA)*ALOG(RD/RI) /	00024700
	* (2.*RR*RA*(1.-XP))	00024800
	JA(J) = I	00024900
	A(J) = RA-RD	00025000
	J = J+1	00025100
	JA(J) = I + N5	00025200
	A(J) = -RA*(1.-2.*XP)/2. -XY1+XY2	00025300
	J = J + 1	00025400
	JA(J) = I + N6	00025500
	A(J) = -RA*(1.-2.*XP)/2. +XY1-XY2	00025600
	J = J + 1	00025700
	A(J) = RA*XP	00025800
	IF(I-NFR) 730,730,740	00025900
730	JA(J) = N8 + 7	00026000
	J = J + 1	00026100
	JA(J) = N8 + 9 + I	00026200
	A(J) = 1.	00026300
	GO TO 750	00026400
740	JA(J) = N8 + 8	00026500
	J = J + 1	00026600
	JA(J) = N8 + 10 + I	00026700
	A(J) = 1.	00026800
750	CONTINUE	00026900
	J = J + 1	00027000
	JA(J) = N9 + 10 + I	00027100
	A(J) = -RI*RI*(RD*RD/RA+(1.-2.*XP)*RA) / (2.*XG*RR)	00027200
	J = J + 1	00027300
	JA(J) = N9 + 11 + I	00027400
	A(J) = RD*RD*(RI*RI/RA+(1.-2.*XP)*RA) / (2.*XG*RR)	00027500
2100	CONTINUE	00027600
C		00027700
C	BOUNDARY CONDITION FOR PRESSURE	00027800
C	FUEL	00027900
	J = J + 1	00028000
	IA(N7+7) = J	00028100
	JA(J) = N9 + 11	00028200
	A(J) = 1.0	00028300
C		00028400
C		00028500
	J = J + 1	00028600
	IA(N7+8) = J	00028700
	JA(J) = N10 + 11	00028800
	A(J) = 1.	00028900
C		00029000

C	FUEL AXIAL DISPLACEMENT	00029100
C		00029200
	J = J + 1	00029300
	XGB = GB(NS,1)	00029400
	XPB = PB(NS,1)	00029500
	IA(N7+9) = J	00029600
	JA(J) = N8 + 1	00029700
	A(J) = XPB	00029800
	J = J + 1	00029900
	JA(J) = N8 + 3	00030000
	A(J) = XPB	00030100
	J = J + 1	00030200
	JA(J) = N8 + 5	00030300
	A(J) = -1.0	00030400
	J = J + 1	00030500
	JA(J) = N8 + 7	00030600
	A(J) = 1.+XPB	00030700
	J = J + 1	00030800
	RO = RF(NS,NFR1)	00030900
	RI = RF(NS,1)	00031000
	RR = (RO+RI)*(RO-RI)	00031100
	JA(J) = N9 + 11 + NFR	00031200
	A(J) = -XPB*RO*RO / (XGB*RR)	00031300
	J = J + 1	00031400
	DL = FLZ(NS)	00031500
	JA(J) = N10 + 12	00031600
C====	REMARK	00031700
	A(J) = DL / (2.*XGB*PI*RR)	00031800
C====	REMARK	00031900
C		00032000
C	CLAD AXIAL DISPLACEMENT	00032100
C		00032200
	J = J + 1	00032300
	IA(N7+10) = J	00032400
	RO = RC(NS,NCL1)	00032500
	RI = RC(NS,1)	00032600
	XPB = PB(NS,2)	00032700
	XGB = GB(NS,2)	00032800
	RR = (RO+RI)*(RO-RI)	00032900
	JA(J) = N8 + 2	00033000
	A(J) = XPB	00033100
	J = J + 1	00033200
	JA(J) = N8 + 4	00033300
	A(J) = XPB	00033400
	J = J + 1	00033500
	JA(J) = N8 + 6	00033600
	A(J) = -1.0	00033700
	J = J + 1	00033800
	JA(J) = N8 + 8	00033900
	A(J) = 1.+XPB	00034000
	J = J + 1	00034100
	JA(J) = N9 + 11 + NFR	00034200
	A(J) = XPB*RI*RI / (XGB*RR)	00034300
	J = J + 1	00034400
	JA(J) = N10 + 12	00034500
C====	REMARK	00034600
	A(J) = -DL / (2.*XGB*PI*RR)	00034700
C====	REMARK	00034800

C		00034900
C	STRESS === SIGR ===	00035000
C		00035100
	K = 0	00035200
	DO 2600 I=1,N1	00035300
	J = J + 1	00035400
	IA(N7+10+I) = J	00035500
	JA(J) = N2 + I	00035600
	A(J) = 1.0	00035700
	J = J + 1	00035800
	IF(I-NFR) 770,770,780	00035900
770	RO = RF(NS,I+1)	00036000
	RI = RF(NS,I)	00036100
	GO TO 790	00036200
780	K = K + 1	00036300
	RO = RC(NS,K+1)	00036400
	RI = RC(NS,K)	00036500
790	CONTINUE	00036600
	XP = P(NS,I)	00036700
	XG = G(NS,I)	00036800
	RR = (RO+RI) * (RO-RI)	00036900
	TH = 1.-2.*RI*RI*ALOG(RO/RI)/RR	00037000
	OM = 1.-2.*RO*RO*ALOG(RO/RI)/RR	00037100
	JA(J) = N5 + I	00037200
	A(J) = XG*((1.-TH)*OM + TH) / (2.*(1.-XP))	00037300
	J = J + 1	00037400
	JA(J) = N6 + I	00037500
	A(J) = XG*((TH-1.)*OM - TH) / (2.*(1.-XP))	00037600
	J = J + 1	00037700
	JA(J) = N9 + 10 + I	00037800
	A(J) = -RI*RI*OM / RR	00037900
	J = J + 1	00038000
	JA(J) = N9 + 11 + I	00038100
	A(J) = RO*RO*TH / RR	00038200
2600	CONTINUE	00038300
C		00038400
C	STRESS === SIGC ===	00038500
C		00038600
	K = 0	00038700
	DO 2700 I=1,N1	00038800
	J = J + 1	00038900
	IA(N8+10+I) = J	00039000
	JA(J) = N3 + I	00039100
	A(J) = 1.0	00039200
	J = J + 1	00039300
C		00039400
	IF(I-NFR) 830,830,840	00039500
830	RO = RF(NS,I+1)	00039600
	RI = RF(NS,I)	00039700
	GO TO 850	00039800
840	K = K + 1	00039900
	RO = RC(NS,K+1)	00040000
	RI = RC(NS,K)	00040100
850	CONTINUE	00040200
	XP = P(NS,I)	00040300
	XG = G(NS,I)	00040400
	RR = (RO+RI) * (RO-RI)	00040500
C		00040600

	TH = 1.-2.*RI*RI*ALOG(RO/RI)/RR	00040700
	OM = 1.-2.*RO*RO*ALOG(RO/RI)/RR	00040800
C		00040900
	JA(J) = N5 + I	00041000
	A(J) = XG*((OM-1.)*TH- OM) / (2.*(1.-XP))	00041100
	J = J + 1	00041200
	JA(J) = N6 + I	00041300
	A(J) = XG*((1.-OM)*TH+ OM) / (2.*(1.-XP))	00041400
	J = J + 1	00041500
	JA(J) = N9 + 10 + I	00041600
	A(J) = -RI*RI*(2.-OM)/RR	00041700
	J = J + 1	00041800
	JA(J) = N9 + 11 + I	00041900
	A(J) = RO*RO*(2.-TH) / RR	00042000
2700	CONTINUE	00042100
C		00042200
C	STRESS === SIGZ ===	00042300
C		00042400
	K = 0	00042500
	DO 2800 I=1,N1	00042600
	J = J + 1	00042700
	IA(N9+10+I) = J	00042800
	JA(J) = N4 + I	00042900
	A(J) = 1.0	00043000
	J = J + 1	00043100
C		00043200
	IF(I-NFR) 890,890,900	00043300
890	RO = RF(NS,I+1)	00043400
	RI = RF(NS,I)	00043500
	GO TO 910	00043600
900	K = K + 1	00043700
	RO = RC(NS,K+1)	00043800
	RI = RC(NS,K)	00043900
910	CONTINUE	00044000
	XP = P(NS,I)	00044100
	XG = G(NS,I)	00044200
	RR = (RO+RI)*(RO-RI)	00044300
C		00044400
	TH = 1.-2.*RI*RI*ALOG(RO/RI)/RR	00044500
	DM = 1.-2.*RO*RO*ALOG(RO/RI)/RR	00044600
C		00044700
	JA(J) = N5 + I	00044800
	A(J) = -2.*XG*XP	00044900
	J = J + 1	00045000
	JA(J) = N6 + I	00045100
	A(J) = -2.*XG*XP	00045200
	J = J + 1	00045300
	JA(J) = N7 + I	00045400
	A(J) = 2.*XG	00045500
	J = J + 1	00045600
	A(J) = -2.*XG*(1.+XP)	00045700
	IF(I-NFR) 920,920,930	00045800
920	JA(J) = N8 + 7	00045900
	GO TO 940	00046000
930	JA(J) = N8 + 8	00046100
940	CONTINUE	00046200
	J = J + 1	00046300
	JA(J) = N9 + 10 + I	00046400

A(J) = -2.*XP*RI*RI/RR	00046500
J = J + 1	00046600
JA(J) = N9 + 11 + I	00046700
A(J) = 2.*XP*RO*RO/RR	00046800
2800 CONTINUE	00046900
C	00047000
C	00047100
3000 CONTINUE	00047200
C	00047300
IF (IFRIC-1) 950,960,970	00047400
C	00047500
C	00047600
950 CONTINUE	00047700
C	00047800
C	00047900
C GAP OPEN	00048000
C	00048100
C	00048200
J = J + 1	00048300
IA(N10+11) = J	00048400
JA(J) = N9+11 + NFR	00048500
A(J) = 1.	00048600
J = J + 1	00048700
IA(N10+12) = J	00048800
JA(J) = N10 + 10	00048900
A(J) = 0.	00049000
J = J + 1	00049100
JA(J) = N10 + 11	00049200
A(J) = 0.	00049300
J = J + 1	00049400
JA(J) = N10 + 12	00049500
A(J) = 1.	00049600
C	00049700
GO TO 990	00049800
C	00049900
C	00050000
960 CONTINUE	00050100
C	00050200
C STICK ASSUMPTION	00050300
C	00050400
J = J + 1	00050500
IA(N10+11) = J	00050600
JA(J) = N8 + 7	00050700
A(J) = 1.	00050800
J = J + 1	00050900
JA(J) = N8 + 8	00051000
A(J) = -1.	00051100
C	00051200
GO TO 980	00051300
C	00051400
C	00051500
970 CONTINUE	00051600
C	00051700
C SLIP ASSUMPTION	00051800
C	00051900
J = J + 1	00052000
IA(N10+11) = J	00052100
JA(J) = N9 + 11 + NFR	00052200

A(J) = PI2*RF(NS,NFR1)*FDY	00052300
J = J + 1	00052400
JA(J) = N10 + 12	00052500
A(J) = -1.	00052600
C	00052700
980 CONTINUE	00052800
C	00052900
J = J + 1	00053000
IA(N10+12) = J	00053100
JA(J) = N8 + 9 + NFR	00053200
A(J) = 1.	00053300
J = J + 1	00053400
JA(J) = N8 + 10 + NFR	00053500
A(J) = -1.	00053600
990 CONTINUE	00053700
J = J+1	00053800
IA(N10+13) = J	00053900
C	00054000
C	00054100
RETURN	00054200
END	00054300

	SUBROUTINE MATRXP (IA,JA,A,E,P,DEQP,EQSIG,RC,G,PB,GB,CLZ,KFRIC,	00054400
	* NS)	00054500
C		00054600
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00054700
	* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00054800
	* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00054900
	* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00055000
	* N12C,N13C,N14C,N15C,	00055100
	* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00055200
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00055300
	* PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00055400
	* CPOW,BUCNS,RK,RKR,A91,GASI,TCON,POBU,BUB	00055500
C		00055600
	DIMENSION IA(IGENP1),JA(NONZRP),A(NONZRP),E(NN,NTR),	00055700
	* P(NN,NTR),DEQP(NN,NTR),EQSIG(NN,NTR),RC(NN,NCL1),	00055800
	* G(NN,NTR),PB(NN,2),GB(NN,2),CLZ(NN),KFRIC(NN)	00055900
C		00056000
C		00056100
C	ELEMENTS A OF AX=B, AND IA,JA FOR USE IN THE SUBSEQUENT	00056200
C	CALL TO THE SL-MATH SUBROUTINES	00056300
C		00056400
C	FOR PLENUM SECTION ONLY	00056500
C		00056600
C		00056700
	J = 0	00056800
C	I = N	00056900
	DO 1000 I=1,NCL	00057000
	XE = E(NS,NFR+I)	00057100
	XP = P(NS,NFR+I)	00057200
	J = J + 1	00057300
	IA(I) = J	00057400
	JA(J) = I	00057500
	A(J) = 1.0	00057600
	J = J + 1	00057700
	JA(J) = I + N2C	00057800
	A(J) = -1./XE	00057900
	J = J + 1	00058000
	JA(J) = I + N3C	00058100
	A(J) = XP/XE	00058200
	J = J + 1	00058300
	JA(J) = I + N4C	00058400
	A(J) = XP/XE	00058500
	J = J + 1	00058600
	JA(J) = I + N5C	00058700
	A(J) = -1.0	00058800
	1000 CONTINUE	00058900
C		00059000
C	N+1 - 2N	00059100
C		00059200
	DO 1100 I=1,NCL	00059300
	XE = E(NS,NFR+I)	00059400
	XP = P(NS,NFR+I)	00059500
	J = J + 1	00059600
	IA(N1C+I) = J	00059700
	JA(J) = I + N1C	00059800
	A(J) = 1.0	00059900
	J = J + 1	00060000
	JA(J) = I + N2C	00060100

A(J) = XP/XE	00060200
J = J + 1	00060300
JA(J) = I + N3C	00060400
A(J) = -1./XE	00060500
J = J + 1	00060600
JA(J) = I + N4C	00060700
A(J) = XP/XE	00060800
J = J + 1	00060900
JA(J) = I + N6C	00061000
A(J) = -1.0	00061100
1100 CONTINUE	00061200
C	00061300
C 2N+1 - 3N	00061400
C	00061500
DO 1200 I=1,NCL	00061600
XEP = DEQP(NS,NFR+I)	00061700
XES = EQSIG(NS,NFR+I)	00061800
J = J + 1	00061900
IA(N2C+I) = J	00062000
JA(J) = I + N2C	00062100
A(J) = XEP/XES	00062200
J = J + 1	00062300
JA(J) = I + N3C	00062400
A(J) = -XEP/(2.*XES)	00062500
J = J + 1	00062600
JA(J) = I + N4C	00062700
A(J) = -XEP/(2.*XES)	00062800
J = J + 1	00062900
JA(J) = I + N5C	00063000
A(J) = -1.0	00063100
1200 CONTINUE	00063200
C	00063300
C 3N+1 - 4N	00063400
C	00063500
DO 1300 I=1,NCL	00063600
XEP = DEQP(NS,NFR+I)	00063700
XES = EQSIG(NS,NFR+I)	00063800
J = J + 1	00063900
IA(I+N3C) = J	00064000
JA(J) = I + N2C	00064100
A(J) = -XEP/(2.*XES)	00064200
J = J + 1	00064300
JA(J) = I + N3C	00064400
A(J) = XEP/XES	00064500
J = J + 1	00064600
JA(J) = I + N4C	00064700
A(J) = -XEP/(2.*XES)	00064800
J = J + 1	00064900
JA(J) = I + N6C	00065000
A(J) = -1.0	00065100
1300 CONTINUE	00065200
C	00065300
C 4N+1 - 5N	00065400
C	00065500
DO 1400 I=1,NCL	00065600
XEP = DEQP(NS,NFR+I)	00065700
XES = EQSIG(NS,NFR+I)	00065800
J = J + 1	00065900

IA(I+N4C) = J	00066000
JA(J) = I + N2C	00066100
A(J) = -XEP/(2.*XES)	00066200
J = J + 1	00066300
JA(J) = I + N3C	00066400
A(J) = -XEP/(2.*XES)	00066500
J = J + 1	00066600
JA(J) = I + N4C	00066700
A(J) = XEP/XES	00066800
J = J + 1	00066900
JA(J) = I + N7C	00067000
A(J) = -1.0	00067100
1400 CONTINUE	00067200
C	00067300
C	00067400
C	00067500
C 5N+1 - 5N+3	00067600
C	00067700
DO 1800 I=1,3	00067800
J = J + 1	00067900
IA(I+N5C) = J	00068000
C	00068100
DO 1850 IJ=1,NCL	00068200
RO = RC(NS,IJ+1)	00068300
RI = RC(NS,IJ)	00068400
JA(J) = N5C + (I-1)*NCL + IJ	00068500
A(J) = (RO+RI)*(RO-RI)	00068600
J = J + 1	00068700
1850 CONTINUE	00068800
C	00068900
JA(J) = N8C + I	00069000
RO = RC(NS,NCL1)	00069100
RI = RC(NS,1)	00069200
A(J) = -(RO+RI)*(RO-RI)	00069300
1800 CONTINUE	00069400
C	00069500
C 5N+4 - 7N+3	00069600
C	00069700
C INNER DISPLACEMENT 5N+4 - 6N+3	00069800
C	00069900
DO 1900 I=1,NCL	00070000
J = J + 1	00070100
IA(I+N5C+3) = J	00070200
RO = RC(NS,I+1)	00070300
RI = RC(NS,I)	00070400
RR = (RO+RI)*(RO-RI)	00070500
XP = P(NS,NFR+I)	00070600
XG = G(NS,NFR+I)	00070700
JA(J) = I + N5C	00070800
A(J) = -RI*(1.-2.*XP)/2. + RI*RO*RO*ALOG(RO/RI) / RR	00070900
J = J + 1	00071000
JA(J) = I + N6C	00071100
A(J) = -RI*(1.-2.*XP)/2. - RI*RO*RO*ALOG(RO/RI) / RR	00071200
J = J + 1	00071300
JA(J) = N8C + 4	00071400
A(J) = RI*XP	00071500
J = J + 1	00071600
JA(J) = N8C + 4 + I	00071700

	A(J) = 1.0	00071800
	J = J + 1	00071900
	JA(J) = N9C + 5 + I	00072000
	A(J) = -RI*((1.-2.*XP)*RI*RI+RO*RO) / (2.*XG*RR)	00072100
	J = J + 1	00072200
	JA(J) = N9C + 6 + I	00072300
	A(J) = (1.-XP)*RI*RO*RO / (XG*RR)	00072400
	1900 CONTINUE	00072500
C		00072600
C	OUTER DISPLACEMENT 6N+4 - 7N+3	00072700
C		00072800
	DO 2000 I=1,NCL	00072900
	J = J + 1	00073000
	IA(I+N6C+3) = J	00073100
	RO = RC(NS,I+1)	00073200
	RI = RC(NS,I)	00073300
	RR = (RO+RI)*(RO-RI)	00073400
	XP = P(NS,NFR+I)	00073500
	XG = G(NS,NFR+I)	00073600
	JA(J) = I + N5C	00073700
	A(J) = -RO*(1.-2.*XP)/2. + RO*RI*RI*ALOG(RO/RI)/RR	00073800
	J = J + 1	00073900
	JA(J) = I + N6C	00074000
	A(J) = -RO*(1.-2.*XP)/2. - RO*RI*RI*ALOG(RO/RI)/RR	00074100
	J = J + 1	00074200
	JA(J) = N8C + 4	00074300
	A(J) = RO*XP	00074400
	J = J + 1	00074500
	JA(J) = N8C + 5 + I	00074600
	A(J) = 1.0	00074700
	J = J + 1	00074800
	JA(J) = N9C + 5 + I	00074900
	A(J) = -RO*RI*RI*(1.-XP) / (XG*RR)	00075000
	J = J + 1	00075100
	JA(J) = N9C + 6 + I	00075200
	A(J) = RO*(RI*RI+(1.-2.*XP)*RO*RO) / (2.*XG*RR)	00075300
	2000 CONTINUE	00075400
C		00075500
C	BOUNDARY CONDITION FOR PRESSURE	00075600
C		00075700
	J = J + 1	00075800
	IA(N7C+4) = J	00075900
	JA(J) = N9C + 6	00076000
	A(J) = 1.0	00076100
	J = J + 1	00076200
	IA(N7C+5) = J	00076300
	JA(J) = N10C + 6	00076400
	A(J) = 1.	00076500
C		00076600
C		00076700
C		00076800
C	CLAD AXIAL DISPLACEMENT 7N+6	00076900
C		00077000
	J = J + 1	00077100
	IA(N7C+6) = J	00077200
	RO = RC(NS,NCL1)	00077300
	RI = RC(NS,1)	00077400
	XPB = PB(NS,2)	00077500

	XGB = GB(NS,2)	00077600
	RR = (RO+RI)*(RO-RI)	00077700
	JA(J) = N8C + 1	00077800
	A(J) = XPB	00077900
	J = J + 1	00078000
	JA(J) = N8C + 2	00078100
	A(J) = XPB	00078200
	J = J + 1	00078300
	JA(J) = N8C + 3	00078400
	A(J) = -1.0	00078500
	J = J + 1	00078600
	JA(J) = N8C + 4	00078700
	A(J) = 1.+XPB	00078800
	J = J + 1	00078900
	JA(J) = N9C + 6	00079000
	A(J) = XPB*RI*RI / (XGB*PR)	00079100
	J = J + 1	00079200
	JA(J) = N10C + 7	00079300
C====	REMARK	00079400
	A(J) = -CLZ(NS) / (2.*PI*XGB*RR)	00079500
C====	REMARK	00079600
C		00079700
C	STRESS == SIGR == 7N+7 - 8N+6	00079800
C		00079900
	DO 2300 I=1,NCL	00080000
	J = J + 1	00080100
	IA(I+N7C+6) = J	00080200
	JA(J) = I + N2C	00080300
	A(J) = 1.0	00080400
	J = J + 1	00080500
	RO = RC(NS,I+1)	00080600
	RI = RC(NS,I)	00080700
	RR = (RO+RI)*(RO-RI)	00080800
	XP = P(NS,NFR+I)	00080900
	XG = G(NS,NFR+I)	00081000
	TH = 1.-2.*RI*RI*ALOG(RO/RI)/RR	00081100
	OM = 1.-2.*RO*RO*ALOG(RO/RI)/RR	00081200
	JA(J) = I + N5C	00081300
	A(J) = XG*((1.-TH)*OM + TH) / (2.*(1.-XP))	00081400
	J = J + 1	00081500
	JA(J) = I + N6C	00081600
	A(J) = XG*((TH-1.)*OM - TH) / (2.*(1.-XP))	00081700
	J = J + 1	00081800
	JA(J) = N9C + 5 + I	00081900
	A(J) = -RI*RI*OM / RR	00082000
	J = J + 1	00082100
	JA(J) = N9C + 6 + I	00082200
	A(J) = RO*RO*TH / RR	00082300
2300	CONTINUE	00082400
C		00082500
C	STRESS == SIGC == 8N+7 - 9N+6	00082600
C		00082700
	DO 2400 I=1,NCL	00082800
	J = J + 1	00082900
	IA(I+N8C+6) = J	00083000
	JA(J) = I + N3C	00083100
	A(J) = 1.0	00083200
	J = J + 1	00083300

C		00083400
	RO = RC(NS,I+1)	00083500
	RI = RC(NS,I)	00083600
	RR = (RO+RI)*(RO-RI)	00083700
	XP = P(NS,NFR+I)	00083800
	XG = G(NS,NFR+I)	00083900
	TH = 1.-2.*RI*RI*ALOG(RO/RI) / RR	00084000
	OM = 1.-2.*RO*RO*ALOG(RO/RI) / RR	00084100
C		00084200
	JA(J) = I + N5C	00084300
	A(J) = XG*((OM-1.)*TH- OM) / (2.*(1.-XP))	00084400
	J = J + 1	00084500
	JA(J) = I + N6C	00084600
	A(J) = XG*((1.-OM)*TH+ OM) / (2.*(1.-XP))	00084700
	J = J + 1	00084800
	JA(J) = N9C + 5 + I	00084900
	A(J) = -RI*RI*(2.-OM) / RR	00085000
	J = J + 1	00085100
	JA(J) = N9C + 6 + I	00085200
	A(J) = RO*RO*(2.-TH) / RR	00085300
2400	CONTINUE	00085400
C		00085500
C	STRESS == SIGZ == 9N+7 - 10N+6	00085600
C		00085700
	DO 2500 I=1,NCL	00085800
	J = J + 1	00085900
	IA(I+N9C+6) = J	00086000
	JA(J) = I + N4C	00086100
	A(J) = 1.0	00086200
	J = J + 1	00086300
	RO = RC(NS,I+1)	00086400
	RI = RC(NS,I)	00086500
	XP = P(NS,NFR+I)	00086600
	XG = G(NS,NFR+I)	00086700
	RR = (RO+RI)*(RO-RI)	00086800
C		00086900
	TH = 1.-2.*RI*RI*ALOG(RO/RI) / RR	00087000
	OM = 1.-2.*RO*RO*ALOG(RO/RI) / RR	00087100
C		00087200
	JA(J) = I + N5C	00087300
	A(J) = -2.*XG*XP	00087400
	J = J + 1	00087500
	JA(J) = I + N6C	00087600
	A(J) = -2.*XG*XP	00087700
	J = J + 1	00087800
	JA(J) = I + N7C	00087900
	A(J) = 2.*XG	00088000
	J = J + 1	00088100
	JA(J) = N8C + 4	00088200
	A(J) = -2.*XG*(1.+XP)	00088300
	J = J + 1	00088400
	JA(J) = N9C + 5 + I	00088500
	A(J) = -2.*XP*RI*RI / RR	00088600
	J = J + 1	00088700
	JA(J) = N9C + 6 + I	00088800
	A(J) = 2.*XP*RO*RO / RR	00088900
2500	CONTINUE	00089000
C		00089100

C	FRICITION FORCE	10N+7	00089200
C			00089300
	J = J + 1		00089400
	IA(N10C+7) = J		00089500
	JA(J) = N10C + 7		00089600
	A(J) = 1.0		00089700
C			00089800
	J = J + 1		00089900
	IA(N10C+8) = J		00090000
C			00090100
	RETURN		00090200
	END		00090300

	SUBROUTINE OTMCAL (OTOM,N)	00090400
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00090500
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00090600
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00090700
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00090800
*	N12C,N13C,N14C,N15C,	00090900
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00091000
	DIMENSION OTOM(NN)	00091100
C		00091200
C	OTOM (O/M) MODIFY /	00091300
C		00091400
	RETURN	00091500
	END	00091600

	SUBROUTINE PLENP (RVB,RCI,TFR,FLZ,GRT,FL,CL,TOUT,PLPR,CVPR,RUB,	00000100
*	CLZ)	00000200
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00000300
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00000400
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00000500
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00000600
*	N12C,N13C,N14C,N15C,	00000700
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00000800
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00000900
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00001000
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00001100
C		00001200
	DIMENSION RVB(NN) , RCI(NN) , TFR(NN,NFR1) , FLZ(NN) ,RUB(NN) ,	00001300
*	CLZ(NN)	00001400
C		00001500
C		00001600
C	PLENUM PRESSURE , AND CENTRAL VOID PRESSURE CALC.	00001700
C		00001800
C		00001900
	VT = 0.	00002000
	VTT = 0.	00002100
	TD = TOUT + 273.	00002200
C	VPL = PI*RCI(NN)*RCI(NN)*CLZ(NN)	00002300
	VPL = PI*RCI(NN)*RCI(NN)*(CL-FL)	00002400
C		00002500
	DO 1000 N=1,NNF	00002600
	VVO = PI*RVB(N)*RVB(N)*FLZ(N)	00002700
	VT = VT + VVO	00002800
	TX = TFR(N,1) + 273.	00002900
	VTT = VTT + VVO/TX	00003000
C		00003100
	VGAP = PI*(RCI(N)+RUB(N))*(RCI(N)-RUB(N)) * FLZ(N)	00003200
	VPL = VPL + VGAP	00003300
C		00003400
	1000 CONTINUE	00003500
C		00003600
C		00003700
	VTOT = VPL + VT	00003800
C		00003900
C	TAV = VTOT/(VPL/TD + VTT)	00004000
C		00004100
C	RK IS IDEAL GAS CONSTANT IN KG*CM/MOLE/DEG K	00004200
C		00004300
C	PLPR = GRT*RK*TAV / VTOT	00004400
C		00004500
	PLPR = GRT*RK*TD / VTOT	00004600
	CVPR = GRT*RK*VT / (VTOT*VTT)	00004700
C		00004800
C	PLPR,CVPR NOW IN KG/CM**2	00004900
C		00005000
C		00005100
	RETURN	00005200
	END	00005300

SUBROUTINE PLOT1 (XPOW,XRL,XTL,XT,XTLMX,XRLR,XTLR,XCPGP,XCPU,	00000100
* XBU,POW,RFRPM,TFRPM,RCB,REB,RUB,RCI,CPU,BU,	00000200
* RFR,TFR,IT,N)	00000300
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00000400
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1	00000500
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00000600
* PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX	00000700
COMMON /XFRIC/ FST,FDY,GAP	00000800
COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00000900
DIMENSION XPOW(NBUF) , XRL(NBUF,NP1) , XTL(NBUF,NP1) , XT(NBUF) ,	00001000
* XTLMX(NBUF) , XRLR(NBUF,2) , XTLR(NBUF,2) ,	00001100
* XCPGP(NBUF) , XCPU(NBUF,NP) , XBU(NBUF) , POW(NN) ,	00001200
* RFRPM(NN,NP1) , TFRPM(NN,NP1) , RCB(NN) , REB(NN) ,	00001300
* RUB(NN) , RCI(NN) , CPU(NN,NP) , BU(NN) ,	00001400
* RFR(NN,NFR1) , TFR(NN,NFR1)	00001500
C	00001600
C FUEL REGION PLOTTER PREPARATION	00001700
C	00001800
C	00001900
XPOW(IT) = POW(N)	00002000
XBU(IT) = BU(N)	00002100
XRLR(IT,1) = RCB(N)*10.	00002200
XRLR(IT,2) = REB(N)*10.	00002300
XCPGP(IT) = GAP * 10.0	00002400
XTLR(IT,1) = TFR(N,NCR+1)	00002500
XTLR(IT,2) = TFR(N,NCR+NER+1)	00002600
C	00002700
XTLMAX = 1.E-10	00002800
DO 100 I=1,NP1	00002900
XRL(IT,I) = RFRPM(N,I)*10.0	00003000
XTL(IT,I) = TFRPM(N,I)	00003100
IF (XTL(IT,I).GT.XTLMAX) XTLMAX=XTL(IT,I)	00003200
IF (I.EQ.NP1) GO TO 100	00003300
XCPU(IT,I) = CPU(N,I)	00003400
100 CONTINUE	00003500
XTLMX(IT) = XTLMAX	00003600
XT(IT) = XTL(IT,NP1)	00003700
RETURN	00003800
END	00003900

SUBROUTINE PLOT2 (XSIGC,SIGC,VOL,IT,N)	00004000
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00004100
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1	00004200
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00004300
* PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX	00004400
COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00004500
DIMENSION XSIGC(NBUF) , SIGC(NN,NTR) , VOL(NN,NTR)	00004600
C	00004700
C	00004800
C CLAD STRESS PLOTTER PREPARATION	00004900
C	00005000
C CLAD AVERAGE HOOP STRESS	00005100
C	00005200
X = 0.	00005300
Y = 0.	00005400
DO 100 I=1,NCL	00005500
Y = Y + VOL(N,NFR+I)	00005600
X = X + SIGC(N,NFR+I)*VOL(N,NFR+I)	00005700
100 CONTINUE	00005800
XSIGC(IT) = X*0.01 / Y	00005900
RETURN	00006000
END	00006100

SUBROUTINE PLOT3 (XCSTN,XTSC,XTPC,RCO,RCOO,TSC,TPC,VOL,RDIS,IT,N)		00006200
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,		00006300
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1	00006400
COMMON /OTHERS/ PG(3),CML(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),		00006500
*	PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX	00006600
COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT		00006700
DIMENSION XCSTN(NBUF), XTSC(NBUF), XTPC(NBUF), RCO(NN),		00006800
*	RCOO(NN), TSC(NN,NTR), TPC(NN,NTR), VOL(NN,NTR),	00006900
*	RDIS(NN,NTR2)	00007000
C		00007100
C		00007200
C	CLAD STRAIN PLOTTER PREPARATION	00007300
C		00007400
C		00007500
	XCSTN(IT) = (RCO(N)+RDIS(N,NTR2)-RCOO(N)) / RCOO(N) * 100.	00007600
	X = 0.	00007700
	Y = 0.	00007800
	Z = 0.	00007900
	DO 100 I=1,NCL	00008000
	Z = Z + VOL(N,NFR+I)	00008100
	X = X + TSC(N,NFR+I)*VOL(N,NFR+I)	00008200
	Y = Y + TPC(N,NFR+I)*VOL(N,NFR+I)	00008300
100	CONTINUE	00008400
	XTSC(IT) = X*100. / Z	00008500
	XTPC(IT) = Y*100. / Z	00008600
	RETURN	00008700
	END	00008800

SUBROUTINE PLOT4 (RFRPM,RCB,REB,LMR,LMC,LMZ,IT,N)	00008900
REAL*8 DAYDAY	00009000
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00009100
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1	00009200
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00009300
* P1,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX	00009400
COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00009500
COMMON /DAY/ DAYDAY	00009600
DIMENSION RFRPM(NN,NP1) , RCB(NN) , REB(NN) ,	00009700
* LMR(NN,NTR) , LMC(NN,NTR) , LMZ(NN,NTR)	00009800
C	00009900
C**** IPCRK = AXIAL SECTION FOR FUEL CRACK	00010000
C	00010100
C FUEL CRACK PLOTTER PREPARATION	00010200
C	00010300
C	00010400
C CRACK PLOT	00010500
C	00010600
C	00010700
C	00010800
IF (IPCNT.GT.0) GO TO 100	00010900
CALL SYMBOL(0.,200.,10.,'** RUN DATE **',0.,14)	00011000
CALL SYMBOL(30.,185.,10.,DAYDAY,0.,8)	00011100
CALL SYMBOL(40.,100.,10.,'ACTIVE-II',0.,9)	00011200
CALL PLOT(200.,0.,-3)	00011300
100 CONTINUE	00011400
C	00011500
C	00011600
DX = 100. / (RFRPM(N,NP1)*10.)	00011700
RADI = RFRPM(N,1)*10.*DX	00011800
DELR1 = (RCB(N)-RFRPM(N,1)) / NCR * 10.0	00011900
DELR2 = (REB(N)-RCB(N)) / NER * 10.0	00012000
DELR3 = (RFRPM(N,NP1)-REB(N)) / NUR * 10.0	00012100
CALL PLOT (150.,0.,-3)	00012200
CALL NEWPEN(1)	00012300
RAD = RADI	00012400
CALL CIRCLE (RAD,130.,0.,360.,RAD,RAD,0.0)	00012500
C	00012600
C	00012700
C RADIAL	00012800
C	00012900
C	00013000
DO 10040 K=1,3	00013100
IF (K-2) 10041,10042,10043	00013200
10041 KK = 1	00013300
KKK = NCR	00013400
DELR = DELR1	00013500
GO TO 10044	00013600
10042 KK = NCR+1	00013700
KKK = NCR+NER	00013800
DELR = DELR2	00013900
GO TO 10044	00014000
10043 KK = NCR+NER+1	00014100
KKK = NFR	00014200
DELR = DELR3	00014300
10044 CONTINUE	00014400
C	00014500
C	00014600

DO 10050 KL=KK, KKK	00014700
CALL NEWPEN(2)	00014800
LR = LMR (N, KL)	00014900
DLR = DELR / (LR+1)	00015000
IF (LR.EQ.0) GO TO 10051	00015100
C	00015200
DO 10060 L=1, LR	00015300
RAD = RAD + DLR*DX	00015400
CALL CIRCLE (RAD, 130., 0., 360., RAD, RAD, 0.5)	00015500
10060 CONTINUE	00015600
C	00015700
10051 CONTINUE	00015800
C	00015900
RAD = RAD + DLR*DX	00016000
CALL NEWPEN(1)	00016100
CALL CIRCLE (RAD, 130., 0., 360., RAD, RAD, 0.0)	00016200
10050 CONTINUE	00016300
C	00016400
IF (K.EQ.3) GO TO 10040	00016500
CALL SYMBOL (RAD, 130., 3., K, 0., -1)	00016600
CALL SYMBOL (0., 130.+RAD, 3., K, 90., -1)	00016700
CALL SYMBOL (-RAD, 130., 3., K, 180., -1)	00016800
CALL SYMBOL (0., 130.-RAD, 3., K, 270., -1)	00016900
C	00017000
10040 CONTINUE	00017100
C	00017200
CALL SYMBOL (-35., 15., 5., 'R-C-DIRECTION CRACK', 0., 19)	00017300
CALL AXIAL (-50., 0., IPCRK)	00017400
CALL SYMBOL (30., 0., 4., 'STEP =', 0., 6)	00017500
CALL NUMBER (58., 0., 4., FLOAT(IT), 0., -1)	00017600
C	00017700
C	00017800
C CIRCUMFERENTIAL	00017900
C	00018000
CALL NEWPEN(3)	00018100
C	00018200
LCMX = 0	00018300
DO 10070 K=1, NFR	00018400
IF (LMC(N, K).GT.LCMX) LCMX=LMC(N, K)	00018500
10070 CONTINUE	00018600
C	00018700
IF (LCMX.NE.0) ANG=2.*PI / LCMX	00018800
RAD = RAD	00018900
C	00019000
DO 10080 K=1, 3	00019100
IF (K-2) 10081, 10082, 10083	00019200
10081 KK = 1	00019300
KKK = NCR	00019400
DELR = DELR1	00019500
GO TO 10084	00019600
10082 KK = NCR+1	00019700
KKK = NCR+NER	00019800
DELR = DELR2	00019900
GO TO 10084	00020000
10083 KK = NCR+NER+1	00020100
KKK = NFR	00020200
DELR = DELR3	00020300
10084 CONTINUE	00020400

C		00020500
	DO 10090 L=KK,KKK	00020600
	RADO = RAD	00020700
	RAD = RAD + DELR*DX	00020800
	IF (LMC (N,L).EQ.0) GO TO 10090	00020900
	LC = LMC (N,L)	00021000
C		00021100
	DO 10100 LL=1,LC	00021200
	ANGL = ANG*(LL-1)	00021300
	X0 = RAD*COS(ANGL)	00021400
	Y0 = RAD*SIN(ANGL) + 130.	00021500
	X1 = RADO*COS(ANGL)	00021600
	Y1 = RADO*SIN(ANGL) + 130.	00021700
	CALL AROHD (X0,Y0,X1,Y1,RAD-RADO,3.0,6)	00021800
10100	CONTINUE	00021900
C		00022000
10090	CONTINUE	00022100
C		00022200
10080	CONTINUE	00022300
C		00022400
	CALL NEWPEN(2)	00022500
	CALL SYMBOL (150.,180.,4.,'---',0.,3)	00022600
	CALL SYMBOL (166.,180.,4.,'CIRCUMF. CRACK',0.,14)	00022700
	CALL NEWPEN(3)	00022800
	CALL SYMBOL (150.,160.,4.,'---',0.,3)	00022900
	CALL SYMBOL (166.,160.,4.,'RADIAL CRACK',0.,12)	00023000
	CALL NEWPEN(1)	00023100
	CALL SYMBOL (150.,140.,4.,1,0.,-1)	00023200
	CALL SYMBOL (166.,140.,4.,'COLUMNAR BNDRY',0.,14)	00023300
	CALL SYMBOL (150.,120.,4.,2,0.,-1)	00023400
	CALL SYMBOL (166.,120.,4.,'EQUIAXED BNDRY',0.,14)	00023500
	CALL PLOT (300.,0.,-3)	00023600
C		00023700
C		00023800
C	AXIAL	00023900
C		00024000
C		00024100
	LZMX = 0	00024200
	DO 10110 K=1,NFR	00024300
	IF (LMZ(N,K).GT.LZMX) LZMX=LMZ(N,K)	00024400
10110	CONTINUE	00024500
C		00024600
	DY = 100. / (LZMX+1)	00024700
	CALL NEWPEN(1)	00024800
C		00024900
	Y0 = 190.	00025000
	DO 10119 K=1,4	00025100
	CALL PLOT (0.,Y0,3)	00025200
	Y0 = Y0 - 20.4	00025300
	CALL PLOT (0.,Y0,2)	00025400
	Y0 = Y0 - 1.5	00025500
	CALL PLOT (0.,Y0,3)	00025600
	Y0 = Y0 - 1.5	00025700
	CALL PLOT (0.,Y0,2)	00025800
	Y0 = Y0 - 1.5	00025900
10119	CONTINUE	00026000
	CALL PLOT (0.,Y0,3)	00026100
	CALL PLOT (0.,70.,2)	00026200

C		00026300
	RAD = RAD1	00026400
	CALL PLOT (RAD,180.,3)	00026500
	CALL PLOT (RAD,80.,2)	00026600
C		00026700
	DO 10120 K=1,3	00026800
	IF (K-2) 10121,10122,10123	00026900
10121	KK = 1	00027000
	KKK = NCR	00027100
	DELR = DELR1	00027200
	GO TO 10124	00027300
10122	KK = NCR+1	00027400
	KKK = NCR+NER	00027500
	DELR = DELR2	00027600
	GO TO 10124	00027700
10123	KK = NCR+NER+1	00027800
	KKK = NCR	00027900
	DELR = DELR3	00028000
10124	CONTINUE	00028100
C		00028200
	DO 10130 L=KK,KKK	00028300
	CALL NEWPEN(1)	00028400
	RADO = RAD	00028500
	RAD = RAD + DELR*DX	00028600
	CALL PLOT (RAD,180.,3)	00028700
	IF (L.LT.KKK) GO TO 10125	00028800
	CALL PLOT (RAD,80.,2)	00028900
	GO TO 10126	00029000
10125	CALL DASHPT (RAD,80.,2.0*FLOAT(K))	00029100
10126	CONTINUE	00029200
	IF (LMZ (N,L).EQ.0) GO TO 10130	00029300
	LZ = LMZ (N,L)	00029400
	CALL NEWPEN(2)	00029500
C		00029600
	RADC = (RAD+RADO)*0.5	00029700
	DO 10140 LL=1,LZ	00029800
	Y = 180.-DY*LL	00029900
	CALL AROHD (RADC,Y,RADO,Y,RADC-RADO,2.,6)	00030000
	CALL AROHD (RADC,Y,RAD,Y,RAD-RADC,2.,6)	00030100
10140	CONTINUE	00030200
C		00030300
10130	CONTINUE	00030400
C		00030500
10120	CONTINUE	00030600
C		00030700
	CALL NEWPEN(1)	00030800
	CALL PLOT (RAD1,180.,3)	00030900
	CALL PLOT (RAD,180.,2)	00031000
	CALL PLOT (RAD1,80.,3)	00031100
	CALL PLOT (RAD,80.,2)	00031200
	CALL SYMBOL (0.,15.,5.,'Z-DIRECTION CRACK',0.,17)	00031300
	CALL AXIAL (0.,0.,IPCRK)	00031400
	CALL SYMBOL (80.,0.,4.,'STEP =',0.,6)	00031500
	CALL NUMBER (108.,0.,4.,FLOAT(IT),0.,-1)	00031600
C		00031700
C		00031800
C		00031900
C		00032000

CALL NEWPEN(2)	00032100
CALL SYMBOL (130.,170.,4.,'---',0.,3)	00032200
CALL SYMBOL (146.,170.,4.,'AXIAL CRACK',0.,11)	00032300
CALL NEWPEN(1)	00032400
CALL SYMBOL (130.,150.,2.,'-----',0.,6)	00032500
CALL SYMBOL (146.,150.,4.,'COLUMNAR GR. REGION',0.,19)	00032600
CALL SYMBOL (130.,130.,4.,'---',0.,3)	00032700
CALL SYMBOL (146.,130.,4.,'EQUITAXED GR. REGION',0.,19)	00032800
CALL SYMBOL (130.,110.,6.,'--',0.,2)	00032900
CALL SYMBOL (146.,110.,4.,'UNDISTURBED GR. REGION',0.,22)	00033000
CALL PLOT (240.,0.,-3)	00033100
C	00033200
RETURN	00033300
END	00033400

SUBROUTINE PLOT5 (XTIME,XPLPR,XYP,ITP,IT,ITN,TIM,PLPR)	00033500
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00033600
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1	00033700
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00033800
* PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX	00033900
COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00034000
COMMON /COMPLT/ YP,DOOD,TITLE(20)	00034100
DIMENSION XTIME(NBUF) , XPLPR(NBUF) , XYP(NBUF) , ITP(NBUF)	00034200
ITP(IT) = ITN	00034300
XTIME(IT) = TIM	00034400
XPLPR(IT) = PLPR	00034500
XYP(IT) = YP	00034600
RETURN	00034700
END	00034800

SUBROUTINE PLTOR (XPOW,XRL,XTL,XT,XTLMX,XRLR,XTLR,XCPGAP,XCPU,	00034900
* XBU,XSIGC,XCSTN,XTSC,XTPC,XTIME,XPLPR,XYP,	00035000
* ITP,IT,ITN)	00035100
REAL*8 DAYDAY	00035200
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00035300
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1	00035400
COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00035500
* PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX	00035600
COMMON /DAY/ DAYDAY	00035700
COMMON /COMPLT/ YP,DDOD,TITLE(20)	00035800
COMMON /PLTOPT/ IPFUL,IPSTS,IPSTN,IPCRK,NBUF,IPCNT	00035900
DIMENSION XPOW(NBUF), XRL(NBUF,NP1), XTL(NBUF,NP1),	00036000
* XT(NBUF), XTLMX(NBUF), XRLR(NBUF,2), XTLR(NBUF,2),	00036100
* XCPGAP(NBUF), XCPU(NBUF,NP), XBU(NBUF),	00036200
* XSIGC(NBUF), XCSTN(NBUF), XTSC(NBUF),	00036300
* XTPC(NBUF), XTIME(NBUF), XPLPR(NBUF),	00036400
* XYP(NBUF), ITP(NBUF)	00036500
C****	00036600
C**** ITP = TIME STEP NUMBER SELECTED	00036700
C**** XPOW = POWER (W/CM)	00036800
C**** XBU = BURN-UP (ATOMIC-PERCENT)	00036900
C**** XTL = FUEL TEMP (DEGC)	00037000
C**** XRL = FUEL RADIAL COORDINATE (MM)	00037100
C**** XTLMX = FUEL PIN MAXIMUM TEMP (DEGC)	00037200
C**** XPLPR = PLENUM PRESSURE (KG/CM**2)	00037300
C**** XYP = FISSION GAS RELEASED PERCENT ()	00037400
C**** XT = PELLET SURFACE TEMP (DEGC)	00037500
C**** XTIME = TIME (HRS)	00037600
C**** XSIGC = CLAD HOOP STRESS (KG/MM**2)	00037700
C**** XRLR(1) = COLUMNAR GRAIN REGION BOUNDARY (MM)	00037800
C**** XRLR(2) = EQUIAXED REGION BOUNDARY (MM)	00037900
C**** XCPGAP = CLAD-PELLET GAP (MM)	00038000
C**** XCSTN = CLAD STRAIN - DELTAD/D - PERCENT ()	00038100
C**** XCPU = PLUTONIUM CONCENTRATION	00038200
C**** XTSC = CIRCUMFERENTIAL SWELLING STRAIN COMPONENT ACCUMULATED	00038300
C**** XTPC = HOOP CREEP STRAIN ACCUMULATED	00038400
C**** IPFUL = AXIAL SECTION FOR FUEL AND CLAD-PELLET GAP	00038500
C**** IPSTS = AXIAL SECTION FOR CLAD HOOP STRESS	00038600
C**** IPSTN = AXIAL SECTION FOR CLAD STRAIN - PERCENT CHANGE	00038700
C**** IN ORIGINAL O.D.	00038800
C****	00038900
C**** PLOT PEN SELECTION N = 1 --- BLACK	00039000
C**** 2 --- RED	00039100
C**** 3 --- GREEN	00039200
DATA IPLT /0/	00039300
C**** IPLT = 1 ---SKIP PLOT AXIS	00039400
ITN = ITN - 1	00039500
XRL(1,1) = 0.	00039600
C****	00039700
IF (IPCNT.GT.0) GO TO 100	00039800
CALL SYMBOL(0.,200.,10.,'*** RUN DATE **',0.,14)	00039900
CALL SYMBOL(30.,185.,10.,DAYDAY,0.,8)	00040000
CALL SYMBOL(40.,100.,10.,'ACTIVE-II',0.,9)	00040100
CALL PLOT(100.,0.,-3)	00040200
100 CONTINUE	00040300
CALL PLOT (150.,0.,-3)	00040400
C****	00040500
FMPMAX = 1.E-10	00040600

	DO 1000 I=1,IT	00040700
	IF(XTLMX(I).GT.FMPMAX) FMPMAX=XTLMX(I)	00040800
1000	CONTINUE	00040900
	DO 1100 I=1,10	00041000
	MM =1000*I	00041100
	IF(FLOAT(MM).GT.FMPMAX) GO TO 1200	00041200
1100	CONTINUE	00041300
1200	CONTINUE	00041400
	MY = MM / 100	00041500
C****		00041600
	MN = 50000	00041700
	IF(XTIME(IT).LT.100000.) MN = 5000	00041800
	IF(XTIME(IT).LT.10000.) MN = 500	00041900
	IF(XTIME(IT).LT.1000.) MN = 50	00042000
	IF(XTIME(IT).LT.100.) MN = 5	00042100
	DO 1300 I=1,10000	00042200
	NMN = MN*I	00042300
	IF(FLOAT(NMN).GT.XTIME(IT)) GO TO 1400	00042400
1300	CONTINUE	00042500
1400	CONTINUE	00042600
	MX = NMN / MN	00042700
C****	WRITE AXIS	00042800
	DY = MY / 220.	00042900
	DX = MX / 300.	00043000
	CALL NEWPEN (3)	00043100
	CALL PLOT (-34.,0.,3)	00043200
	CALL PLOT (-34.,220.,2)	00043300
	CALL NEWPEN (2)	00043400
	CALL PLOT (0.,220.,3)	00043500
	CALL PLOT (0.,0.,2)	00043600
	CALL NEWPEN (1)	00043700
	CALL PLOT (300.,0.,2)	00043800
	CALL PLOT (300.,220.,2)	00043900
	CALL PLOT (0.,220.,2)	00044000
	DO 1500 I=1,MY	00044100
	J=MY-I+1	00044200
	IF(MOD(I,5).NE.1) GO TO 1600	00044300
	CALL PLOT (298.,J/DY,3)	00044400
	CALL PLOT (302.,J/DY,2)	00044500
	F = FLOAT(J*100)	00044600
	CALL NUMBER (303.,J/DY-2.0,4.,F,0.,-1)	00044700
	GO TO 1500	00044800
1600	CONTINUE	00044900
	CALL PLOT (299.0,J/DY,3)	00045000
	CALL PLOT (301.0,J/DY,2)	00045100
1500	CONTINUE	00045200
	CALL NUMBER (303.,-2.0,4.,0.,0.,-1)	00045300
	CALL SYMBOL (328.,80.0,5.,'TEMPERATURE (C)',90.,17)	00045400
	CALL NUMBER (324.,148.,2.,0.,90.,-1)	00045500
	DO 1700 I=1,MX	00045600
	CALL NEWPEN(3)	00045700
	J=MX-I+1	00045800
	IF(I.EQ.1) GO TO 1650	00045900
	CALL PLOT(J/DX,220.,3)	00046000
	CALL PLOT(J/DX,0.,2)	00046100
1650	CONTINUE	00046200
	F=FLOAT(MN*J)	00046300
	CALL NEWPEN(1)	00046400

	CALL NUMBER(J/DX-4.0,-7.0,4.0,F,0.,-1)	00046500
1700	CONTINUE	00046600
	CALL NUMBER(-2.0,-7.0,4.0,0.0,-1)	00046700
	CALL SYMBOL(100.0,-20.0,5.,'OPERATION TIME (HOUR)',0.,21)	00046800
	CALL NEWPEN(2)	00046900
	CALL NUMBER(-14.0,-2.5,5.0,0.0,-1)	00047000
	I=0	00047100
2000	I=I+1	00047200
	CALL PLOT(-1.0,I*11.0,3)	00047300
	CALL PLOT(1.0,I*11.0,2)	00047400
	I=I+1	00047500
	F=FLOAT(I*5)	00047600
	CALL PLOT(-2.0,FLOAT(I)*11.0,3)	00047700
	CALL PLOT(2.0,FLOAT(I)*11.0,2)	00047800
	CALL NUMBER (-15.0,I*11.0-2.0,4.,F,0.,-1)	00047900
	IF(I.LT.20) GO TO 2000	00048000
	CALL SYMBOL(-20.,80.,5.,'FP GAS RELEASED ()',90.,21)	00048100
	CALL SYMBOL (-20.,170.,5.,108,90.,-1)	00048200
	CALL NEWPEN(3)	00048300
	CALL PLOT(-36.0,0.,3)	00048400
	CALL PLOT(-32.0,0.,2)	00048500
	CALL NUMBER(-42.,-2.0,4.0,0.0,-1)	00048600
	XPLMX=1.E-20	00048700
	DO 2050 I=1,IT	00048800
	IF(XPLMX.LT.XPLPR(I)) XPLMX=XPLPR(I)	00048900
2050	CONTINUE	00049000
	DO 2060 I=1,100	00049100
	IF(XPLMX.GT.FLOAT(I*100)) GO TO 2060	00049200
	IPLN=I	00049300
	GO TO 2070	00049400
2060	CONTINUE	00049500
2070	CONTINUE	00049600
	I=0	00049700
2100	I=I+1	00049800
	CALL PLOT(-35.0,FLOAT(I)*11.0,3)	00049900
	CALL PLOT(-33.0,FLOAT(I)*11.0,2)	00050000
	I=I+1	00050100
	F = FLOAT(I*5*IPLN)	00050200
	CALL PLOT(-36.,FLOAT(I)*11.0,3)	00050300
	CALL PLOT(-32.,FLOAT(I)*11.0,2)	00050400
	CALL NUMBER(-46.,FLOAT(I)*11.0-2.0,4.,F,0.,-1)	00050500
	IF(I.LT.20) GO TO 2100	00050600
	CALL SYMBOL(-50.,80.,5.,'PLENUM PRESSURE (KG/CM**2)',90.,26)	00050700
	CALL NEWPEN(1)	00050800
	DX = NMN / 300.	00050900
	DY = MM / 220.	00051000
	ITM = IT-1	00051100
	DO 4000 I=1,ITM	00051200
	CALL SYMBOL (XTIME(I)/DX,XPLMX(I)/DY,2.0,0,0.,-1)	00051300
	CALL PLOT (XTIME(I)/DX,XPLMX(I)/DY,3)	00051400
	CALL PLOT (XTIME(I+1)/DX,XPLMX(I+1)/DY,2)	00051500
4000	CONTINUE	00051600
	CALL SYMBOL (XTIME(IT)/DX,XPLMX(IT)/DY,2.0,0,0.,-1)	00051700
	DO 4100 I=1,ITM	00051800
	CALL SYMBOL (XTIME(I)/DX,XT(I)/DY,2.0,1,0.,-1)	00051900
	CALL PLOT (XTIME(I)/DX,XT(I)/DY,3)	00052000
	CALL PLOT (XTIME(I+1)/DX,XT(I+1)/DY,2)	00052100
4100	CONTINUE	00052200

CALL SYMBOL (XTIME(IT)/DX,XPLMX(IT)/DY,2.0,1.0,-1)	00052300
CALL NEWPEN(3)	00052400
DY=50.0*IPLN / 110.0	00052500
DO 4200 I=1,ITM	00052600
CALL SYMBOL (XTIME(I)/DX,XPLPR(I)/DY,2.0,2.0,-1)	00052700
CALL PLOT (XTIME(I)/DX,XPLPR(I)/DY,3)	00052800
CALL PLOT (XTIME(I+1)/DX,XPLPR(I+1)/DY,2)	00052900
4200 CONTINUE	00053000
CALL SYMBOL (XTIME(IT)/DX,XPLPR(IT)/DY,2.0,2.0,-1)	00053100
CALL NEWPEN(2)	00053200
DY = 50./110.	00053300
DO 4300 I=1,ITM	00053400
CALL SYMBOL (XTIME(I)/DX,XYP(I)/DY,2.0,4.0,-1)	00053500
CALL PLOT (XTIME(I)/DX,XYP(I)/DY,3)	00053600
CALL PLOT (XTIME(I+1)/DX,XYP(I+1)/DY,2)	00053700
4300 CONTINUE	00053800
CALL SYMBOL (XTIME(IT)/DX,XYP(IT)/DY,2.0,4.0,-1)	00053900
CALL NEWPEN(1)	00054000
CALL SUBTIT(TITLE)	00054100
CALL AXIAL (340.,210.,IPFUL)	00054200
CALL SYMBOL (340.,160.,4.,0.0,-1)	00054300
CALL SYMBOL (342.,158.,4.,' -- MAXIMUM TEMP. IN FUEL',0.,25)	00054400
CALL SYMBOL (340.,154.,4.,1.0,-1)	00054500
CALL SYMBOL (342.,152.,4.,' -- PELLET SURFACE TEMP.',0.,24)	00054600
CALL NEWPEN(2)	00054700
CALL SYMBOL (340.,148.,4.,4.0,-1)	00054800
CALL SYMBOL (342.,146.,4.,' -- FP GAS RELEASED',0.,19)	00054900
CALL NEWPEN(3)	00055000
CALL SYMBOL (340.,142.,4.,2.0,-1)	00055100
CALL SYMBOL (342.,140.,4.,' -- PLENUM PRESSURE',0.,19)	00055200
C**** WRITE AXIS FOR FUEL TEMP	00055300
CALL NEWPEN(1)	00055400
CALL PLOT (520.,0.,-3)	00055500
CALL PLOT (300.,0.,2)	00055600
CALL PLOT(300.,220.,2)	00055700
CALL PLOT(0.,220.,2)	00055800
XRLMAX=1.E-10	00055900
DO 4400 I=1,IT	00056000
IF(XRL(I,NP1).GT.XRLMAX) XRLMAX=XRL(I,NP1)	00056100
4400 CONTINUE	00056200
DO 3000 I=1,20	00056300
X=0.5*(I-1)	00056400
IF(XRLMAX.LT.X) GO TO 3100	00056500
3000 CONTINUE	00056600
3100 CONTINUE	00056700
MXD = I	00056800
DXD = X/300.	00056900
CALL AXISD (MXD,0.5,DXD)	00057000
CALL SYMBOL (65.,-17.,5.,'DISTANCE FROM CENTER OF PELLET (MM)',	00057100
* 0.,35)	00057200
CALL PLOT(0.,0.,3)	00057300
CALL PLOT(0.,220.,2)	00057400
DY=FLOAT(MM)/220.	00057500
DO 3200 I=1,MY	00057600
CALL NEWPEN(1)	00057700
J=MY-I+1	00057800
Y= 100.*J	00057900
CALL NUMBER(-17.,Y/DY-2.,4.,Y,0.,-1)	00058000

	IF(I.EQ.1) GO TO 3200	00058100
	CALL NEWPEN(3)	00058200
	CALL PLOT(0.,Y/DY,3)	00058300
	CALL PLOT(300.,Y/DY,2)	00058400
3200	CONTINUE	00058500
	CALL NEWPEN(1)	00058600
	CALL NUMBER(-17.,-2.,4.,0.,0.,-1)	00058700
	CALL SYMBOL(-23.,70.0,5.,'FUEL TEMPERATURE (C)',90.,22)	00058800
	CALL NUMBER(-27.,163.0,2.0,0.,90.,-1)	00058900
	DO 3300 I=1,IT	00059000
	ICODE = I-1	00059100
	ISW1 = 0	00059200
	ISW2 = 0	00059300
	IF(ICODE.GE.13) ICODE=13	00059400
	CALL PLOT (XRL(I,1)/DXD,XTL(I,1)/DY,3)	00059500
	DO 3400 J=2,NP1	00059600
	CALL SYMBOL(XRL(I,J-1)/DXD,XTL(I,J-1)/DY,2.0,ICODE,0.,-1)	00059700
	IF(ISW1.EQ.1) GO TO 3410	00059800
	IF(XRL(I,J).LT.XRLR(I,1)) GO TO 3410	00059900
	CALL PLOT(XRLR(I,1)/DXD,XTLR(I,1)/DY,2)	00060000
	CALL NEWPEN(2)	00060100
	CALL SYMBOL(XRLR(I,1)/DXD,XTLR(I,1)/DY,2.,ICODE,0.,-1)	00060200
	CALL NEWPEN(1)	00060300
	ISW1 = 1	00060400
3410	CONTINUE	00060500
	IF(ISW2.EQ.1) GO TO 3420	00060600
	IF(XRL(I,J).LT.XRLR(I,2)) GO TO 3420	00060700
	CALL PLOT(XRLR(I,2)/DXD,XTLR(I,2)/DY,2)	00060800
	CALL NEWPEN(3)	00060900
	CALL SYMBOL(XRLR(I,2)/DXD,XTLR(I,2)/DY,2.,ICODE,0.,-1)	00061000
	CALL NEWPEN(1)	00061100
	ISW2 = 1	00061200
3420	CONTINUE	00061300
	CALL PLOT(XRL(I,J)/DXD,XTL(I,J)/DY,2)	00061400
3400	CONTINUE	00061500
	CALL SYMBOL(XRL(I,NP1)/DXD,XTL(I,NP1)/DY,2.0,ICODE,0.,-1)	00061600
3300	CONTINUE	00061700
	CALL NEWPEN(2)	00061800
	CALL SYMBOL(320.,190.,4.,'COLUMNAR BNDRY',0.,14)	00061900
	CALL NEWPEN(3)	00062000
	CALL SYMBOL(320.,180.,4.,'EQUIAXED BNDRY',0.,14)	00062100
	CALL NEWPEN(1)	00062200
	CALL AXIAL (320.,210.,IPFUL)	00062300
	DO 5000 I=1,IT	00062400
	IF(I.GE.15) GO TO 5010	00062500
	ITX = ITP(I)	00062600
	ICODE = I-1	00062700
	YY = 160.0-5.0*(I-1)	00062800
	CALL SYMBOL(320.,YY,4.0,ICODE,0.,-1)	00062900
	CALL SYMBOL(323.,YY-2.0,4.0,' --- STEP',0.,9)	00063000
	CALL NUMBER(360.,YY-2.0,4.,FLOAT(ITX),0.,-1)	00063100
5000	CONTINUE	00063200
	GO TO 5020	00063300
5010	CONTINUE	00063400
	CALL SYMBOL (369.,93.,4.,30,0.,-1)	00063500
C	CALL NUMBER (374.,93.,4.,FLOAT(IT),0.,-1)	00063600
5020	CONTINUE	00063700
	CALL SUBTIT(TITLE)	00063800

C****	WRITE AXIS FOR VOID RADIUS	00063900
	CALL NEWPEN(1)	00064000
	CALL PLOT(480.,0.,-3)	00064100
	CALL PLOT(300.,0.,2)	00064200
	CALL PLOT (300.,220.,2)	00064300
	CALL PLOT (0.,220.,2)	00064400
	CALL AXIST(MX,FLOAT(MN),MX/300.)	00064500
	CALL SYMBOL (100.,-20.,5.,'OPERATION TIME (HOUR)',0.,21)	00064600
	CALL PLOT(0.,0.,3)	00064700
	CALL PLOT(0.,220.,2)	00064800
	VMAX = 1.E-20	00064900
	DO 5100 I=1,IT	00065000
	IF(VMAX.LT.XRL(I,1)) VMAX=XRL(I,1)	00065100
5100	CONTINUE	00065200
	DO 5150 I=1,100	00065300
	XX = 0.1*(I-1)	00065400
	IF(VMAX.LT.XX) GO TO 5200	00065500
5150	CONTINUE	00065600
5200	CONTINUE	00065700
	MMM=(I-1)*10+1	00065800
	DY = X/220.	00065900
	DXT = MN / 300.	00066000
	DO 6500 I=1,MXD	00066100
	F=0.5*(MXD-I)	00066200
	CALL NEWPEN(1)	00066300
	CALL NUMBER(-20.,F/DY-2.0,4.0,F,0.,2)	00066400
	IF(I.EQ.1.OR.I.EQ.MXD) GO TO 6500	00066500
	CALL NEWPEN(3)	00066600
	CALL PLOT(0.,F/DY,3)	00066700
	CALL PLOT(300.,F/DY,2)	00066800
6500	CONTINUE	00066900
	CALL NEWPEN(1)	00067000
	CALL SYMBOL(-28.,80.,5.,'VOID RADIUS (MM)',90.,16)	00067100
	DO 6600 I=1,ITM	00067200
	CALL SYMBOL (XTIME(I)/DXT,XRL(I,1)/DY,2., 0 ,0.,-1)	00067300
	CALL PLOT (XTIME(I+1)/DXT,XRL(I+1,1)/DY,2)	00067400
6600	CONTINUE	00067500
	CALL SYMBOL (XTIME(IT)/DXT,XRL(IT,1)/DY,2., 0 ,0.,-1)	00067600
	DO 6610 I=1,ITM	00067700
	CALL SYMBOL(XTIME(I)/DXT,XRLR(I,1)/DY,2.,3,0.,-1)	00067800
	CALL PLOT(XTIME(I+1)/DXT,XRLR(I+1,1)/DY,2)	00067900
6610	CONTINUE	00068000
	CALL SYMBOL(XTIME(IT)/DXT,XRLR(IT,1)/DY,2.,3,0.,-1)	00068100
	DO 6620 I=1,ITM	00068200
	CALL SYMBOL(XTIME(I)/DXT,XRLR(I,2)/DY,2.,4,0.,-1)	00068300
	CALL PLOT(XTIME(I+1)/DXT,XRLR(I+1,2)/DY,2)	00068400
6620	CONTINUE	00068500
	CALL SYMBOL(XTIME(IT)/DXT,XRLR(IT,2)/DY,2.,4,0.,-1)	00068600
	CALL AXIAL(320.,210.,IPFUL)	00068700
	CALL SYMBOL (320.,180.,4.,0,0.,-1)	00068800
	CALL SYMBOL (323.,178.,4.,' --- VOID RADIUS',0.,16)	00068900
	CALL SYMBOL (320.,172.,4.,3,0.,-1)	00069000
	CALL SYMBOL (323.,170.,4.,' --- COLUMNAR BNDRY',0.,19)	00069100
	CALL SYMBOL (320.,164.,4.,4,0.,-1)	00069200
	CALL SYMBOL (323.,162.,4.,' --- EQUIAXED BNDRY',0.,19)	00069300
	CALL SUBTIT(TITLE)	00069400
C****	WRITE AXIS FOR MAXIMUM FUEL TEMP.	00069500
	CALL NEWPEN(1)	00069600

CALL PLOT(480.,0.,-3)	00069700
CALL PLOT(300.,0.,2)	00069800
CALL PLOT(300.,220.,2)	00069900
CALL PLOT(0.,220.,2)	00070000
CALL AXIST(MX,FLOAT(MN),MX/300.)	00070100
CALL SYMBOL(100.,-20.0,5.,'OPERATION TIME (HOUR)',0.,21)	00070200
CALL PLOT(0.,0.,3)	00070300
CALL PLOT(0.,220.,2)	00070400
DY = MY/220.	00070500
DO 6000 I=1,MY	00070600
J=MY-I+1	00070700
CALL NEWPEN(1)	00070800
IF(MOD(I,5).NE.1) GO TO 6100	00070900
F=FLOAT(J#100)	00071000
CALL NUMBER(-19.,J/DY-2.,4.,F,0.,-1)	00071100
IF(I.EQ.1) GO TO 6000	00071200
CALL NEWPEN(3)	00071300
CALL PLOT(0.0,J/DY,3)	00071400
CALL PLOT(300.,J/DY,2)	00071500
GO TO 6000	00071600
6100 CONTINUE	00071700
CALL PLOT(0.0,J/DY,3)	00071800
CALL PLOT(2.,J/DY,2)	00071900
6000 CONTINUE	00072000
CALL NEWPEN(1)	00072100
CALL NUMBER(-19.,-2.0,4.,0.,0.,-1)	00072200
CALL SYMBOL(-22.,40.0,5.,'MAXIMUM TEMP. IN FUEL PIN (C)',90.,	00072300
* 31)	00072400
CALL NUMBER(-26.,178.,2.,0.,90.,-1)	00072500
DY=MM/220.	00072600
DO 6200 I=1,ITM	00072700
CALL SYMBOL (XTIME(I)/DXT,XTLMX(I)/DY,2.0,IPLT,0.,-1)	00072800
CALL PLOT (XTIME(I+1)/DXT,XTLMX(I+1)/DY,2)	00072900
6200 CONTINUE	00073000
CALL SYMBOL (XTIME(IT)/DXT,XTLMX(IT)/DY,2.0,IPLT,0.,-1)	00073100
CALL AXIAL(320.,210.,IPFUL)	00073200
CALL SUBTIT(TITLE)	00073300
C**** WRITE AXIS FOR POWER	00073400
PMAX=1.E-20	00073500
DO 7000 I=1,IT	00073600
IF(PMAX.LT.XPOW(I)) PMAX=XPOW(I)	00073700
7000 CONTINUE	00073800
DO 7100 I=1,10000	00073900
MP=100*I	00074000
IF(FLOAT(MP).GT.PMAX) GO TO 7200	00074100
7100 CONTINUE	00074200
7200 CONTINUE	00074300
DYP = MP/220.	00074400
DY = DYP	00074500
MPY=MP/50	00074600
CALL NEWPEN(1)	00074700
CALL PLOT (480.,0.,-3)	00074800
CALL PLOT (300.,0.,2)	00074900
CALL PLOT(300.,220.,2)	00075000
CALL PLOT(0.,220.,2)	00075100
CALL AXIST (MX,FLOAT(MN),MX/300.)	00075200
CALL SYMBOL (100.,-20.,5.,'OPERATION TIME (HOUR)',0.,21)	00075300
CALL PLOT (0.,0.,3)	00075400

	CALL PLOT (0.,220.,2)	00075500
	DO 7300 I=1,MPY	00075600
	CALL NEWPEN(1)	00075700
	J=MPY-I+1	00075800
	F=J*50.	00075900
	IF(MOD(I,2).NE.1) GO TO 7400	00076000
	CALL NUMBER(-17.,F/DY-2.,4.,F,0.,-1)	00076100
	IF(I.EQ.1) GO TO 7300	00076200
	CALL NEWPEN(3)	00076300
	CALL PLOT (0.,F/DY,3)	00076400
	CALL PLOT (300.,F/DY,2)	00076500
	GO TO 7300	00076600
7400	CONTINUE	00076700
	CALL PLOT (2.,F/DY,3)	00076800
	CALL PLOT (0.,F/DY,2)	00076900
7300	CONTINUE	00077000
	CALL NEWPEN(1)	00077100
	CALL NUMBER (-17.,-2.,4.,0.,0.,-1)	00077200
	CALL SYMBOL (-23.,83.0,5., 'POWER (W/CM)',90.,12)	00077300
	CALL PLOT (0.,0.,3)	00077400
	DO 7500 I=1,IT	00077500
	CALL PLOT (XTIME(I)/DXT,XPOW(I)/DY,2)	00077600
	CALL SYMBOL (XTIME(I)/DXT,XPOW(I)/DY,2.0,IPLT,0.,-1)	00077700
7500	CONTINUE	00077800
	YBUMAX=20.0	00077900
	IF(XBU(IT).LT.10.) YBUMAX=10.	00078000
	IF(XBU(IT).LT.1.0) YBUMAX=1.0	00078100
	DY= YBUMAX/220.	00078200
	CALL PLOT (330.,0.,3)	00078300
	CALL PLOT (330.,220.,2)	00078400
	DO 7510 I=1,10	00078500
	F = YBUMAX-YBUMAX*(I-1)/10	00078600
	CALL PLOT (330.,F/DY,3)	00078700
	CALL PLOT (332.,F/DY,2)	00078800
	CALL NUMBER (334.,F/DY-2.,4.,F,0.,1)	00078900
7510	CONTINUE	00079000
	CALL PLOT(330.,0.,3)	00079100
	CALL PLOT(332.,0.,2)	00079200
	CALL NUMBER (334.,-2.,4.,0.0,0.,1)	00079300
	CALL SYMBOL (360.,60.,5., 'BURN-UP (ATOMIC-PERCENT)',90.,24)	00079400
	CALL PLOT (0.,0.,3)	00079500
	DO 7520 I=1,IT	00079600
	CALL PLOT (XTIME(I)/DXT,XBU(I)/DY,2)	00079700
	CALL SYMBOL (XTIME(I)/DXT,XBU(I)/DY,2.,4,0.,-1)	00079800
7520	CONTINUE	00079900
	CALL AXIAL (380.,210.,IPFUL)	00080000
	CALL SYMBOL (380.,160.,4.,0,0.,-1)	00080100
	CALL SYMBOL (382.,158.,4., ' -- POWER',0.,9)	00080200
	CALL SYMBOL (380.,140.,4.,4,0.,-1)	00080300
	CALL SYMBOL (382.,138.,4., ' -- BURN UP',0.,11)	00080400
	CALL SUBTIT(TITLE)	00080500
C****	CLAD-PELLET GAP AND CLAD HOOP STRESS	00080600
	GAPMAX=1.E-10	00080700
	SIGMAX=1.E-10	00080800
	SIGMIN = 0.	00080900
	DO 8000 I=1,IT	00081000
	IF(GAPMAX.LT.XCPGAP(I)) GAPMAX=XCPGAP(I)	00081100
	IF(SIGMAX.LT.XSIGC(I)) SIGMAX=XSIGC(I)	00081200

	IF (SIGMIN.GT.XSIGC(I)) SIGMIN=XSIGC(I)	00081300
8000	CONTINUE	00081400
	DO 8100 I=1,10000	00081500
	XG=0.05*I	00081600
	IF(XG.GT.GAPMAX) GO TO 8200	00081700
8100	CONTINUE	00081800
8200	CONTINUE	00081900
	DYG=XG/220.	00082000
	MGY=5*I	00082100
	DO 8300 I=1,10000	00082200
	XS=5.0*I	00082300
	IF(XS.GT.SIGMAX) GO TO 8400	00082400
8300	CONTINUE	00082500
8400	CONTINUE	00082600
	ISIGX = I	00082700
	DO 8410 I=1,10000	00082800
	XSMIN = -5.0*I	00082900
	IF (XSMIN.LT.SIGMIN) GO TO 8420	00083000
8410	CONTINUE	00083100
8420	CONTINUE	00083200
	ISIGN = I	00083300
	IF (SIGMIN.GE.0.) ISIGN = 0	00083400
	IF(SIGMIN.GE.0.) XSMIN=0.	00083500
	XS = XS-XSMIN	00083600
	I = ISIGX+ISIGN	00083700
	DYS=XS/220.	00083800
	MSY= I*5	00083900
	CALL NEWPEN(1)	00084000
	CALL PLOT (510.,0.,-3)	00084100
	CALL PLOT (300.,0.,2)	00084200
	CALL PLOT (300.,220.,2)	00084300
	CALL PLOT (0.,220.,2)	00084400
	CALL PLOT (0.,0.,2)	00084500
	DO 8500 I=1,MSY	00084600
	J=MSY-I+1	00084700
	JNUM = J-5*ISIGN	00084800
	IF(MOD(J,5).NE.0) GO TO 8450	00084900
	CALL PLOT (298.,1.0*J/DYS,3)	00085000
	CALL PLOT (300.,1.0*J/DYS,2)	00085100
	IF(JNUM.LT.0) CALL PLOT(302.,1.*J/DYS,3)	00085200
	IF(JNUM.LT.0) CALL PLOT(304.,1.*J/DYS,2)	00085300
	CALLNUMBER(305.,1.0*J/DYS-2.,4.,1.0*IABS(JNUM),0.,1)	00085400
	GO TO 8500	00085500
8450	CONTINUE	00085600
	CALL PLOT(299.,1.*J/DYS,3)	00085700
	CALL PLOT(300.,1.*J/DYS,2)	00085800
8500	CONTINUE	00085900
	IF(ISIGN.GT.0) CALL PLOT(302.,0.,3)	00086000
	IF(ISIGN.GT.0) CALL PLOT(304.,0.,2)	00086100
	CALL NUMBER(305.,-2.,4.,5.*ISIGN,0.,1)	00086200
	CALL SYMBOL(329.,80.,5.,'CLAD HOOP STRESS (KG/MM**2)',90.,27)	00086300
	CALL AXIST (MX,FLOAT(MN),MX/300.)	00086400
	CALL SYMBOL(100.,-20.,5.,'OPERATION TIME (HOUR)',0.,21)	00086500
	CALL NUMBER (-17.,-2.,4.,0.,0.,2)	00086600
	DO 8600 I=1,MGY	00086700
	IF(MOD(I,5).NE.0) GO TO 8700	00086800
	CALL NUMBER (-17.,0.01*I/DYG-2.,4.,0.01*I,0.,2)	00086900
	CALL PLOT (0.,0.01*I/DYG,3)	00087000

	CALL PLOT (2.,0.01*I/DYG,2)	00087100
	GO TO 8600	00087200
8700	CONTINUE	00087300
	CALL PLOT (0.,0.01*I/DYG,3)	00087400
	CALL PLOT (1.,0.01*I/DYG,2)	00087500
8600	CONTINUE	00087600
	CALL SYMBOL (-22.,80.,5., 'RADIAL GAP SIZE (MM)',90.,20)	00087700
	X0 = XTIME(1)/DXT	00087800
	Y0 = XCPGAP(1)/DYG	00087900
	CALL SYMBOL (X0,Y0,2.0,0,0.,-1)	00088000
	DO 8800 I=2,IT	00088100
	X0 = XTIME(I)/DXT	00088200
	Y0 = XCPGAP(I)/DYG	00088300
	CALL PLOT (X0,Y0,2)	00088400
	CALL SYMBOL (X0,Y0,2.,0,0.,-1)	00088500
8800	CONTINUE	00088600
	X0 = XTIME(1)/DXT	00088700
	Y0 = (XSIGC(1)-XSMIN)/DYS	00088800
	CALL SYMBOL (X0,Y0,2.,4,0.,-1)	00088900
	DO 8900 I=2,IT	00089000
	X0 = XTIME(I)/DXT	00089100
	Y0 = (XSIGC(I)-XSMIN)/DYS	00089200
	CALL PLOT (X0,Y0,2)	00089300
	CALL SYMBOL (X0,Y0,2.,4,0.,-1)	00089400
8900	CONTINUE	00089500
	CALL SUBTIT(TITLE)	00089600
	CALL SYMBOL(343.,160.,4.,0,0.,-1)	00089700
	CALL SYMBOL(345.,158.,4., ' -- CLAD PELLET GAP',0.,19)	00089800
	CALL AXIAL (361.,152.,IPFUL)	00089900
	CALL SYMBOL(343.,130.,4.,4,0.,-1)	00090000
	CALL SYMBOL(345.,128.,4., ' -- CLAD HOOP STRESS',0.,20)	00090100
	CALL AXIAL (361.,122.,IPSTS)	00090200
C****	CLAD STRAIN AND POWER	00090300
	STNMAX = 1.E-10	00090400
	STNMIN = 0.	00090500
	DO 9000 I=1,IT	00090600
	IF (STNMAX.LT.XCSTN(I)) STNMAX=XCSTN(I)	00090700
	IF (STNMIN.GT.XCSTN(I)) STNMIN=XCSTN(I)	00090800
	IF (STNMIN.GT.XTPC(I)) STNMIN=XTPC(I)	00090900
9000	CONTINUE	00091000
	STNXXX = STNMAX-STNMIN	00091100
	HABA = 5.0	00091200
	IF (STNXXX.LT.10.) HABA=0.50	00091300
	IF (STNXXX.LT.1.) HABA=0.05	00091400
	DO 9100 I=1,10000	00091500
	XS = HABA*I	00091600
	IF (XS.GT.STNMAX) GO TO 9200	00091700
9100	CONTINUE	00091800
9200	CONTINUE	00091900
	ISTNX = I	00092000
	DO 9210 I=1,10000	00092100
	XS = -HABA*I	00092200
	IF (XS.LT.STNMIN) GO TO 9220	00092300
9210	CONTINUE	00092400
9220	CONTINUE	00092500
	ISTNN = I	00092600
	IF (STNMIN.GE.0.) ISTNN=0	00092700
	I = ISTNX+ISTNN	00092800

	DY = I*HABA/220.	00092900
	MSY = I	00093000
	CALL NEWPEN(1)	00093100
	CALL PLOT(500.,0.,-3)	00093200
	CALL PLOT(300.,0.,2)	00093300
	CALL PLOT(300.,220.,2)	00093400
	CALL PLOT(0.,220.,2)	00093500
	CALL PLOT(0.,0.,2)	00093600
	DO 9300 I=1,MPY	00093700
	J = MPY-I+1	00093800
	F = J*50.	00093900
	CALL PLOT(298.,F/DYP,3)	00094000
	CALL PLOT(300.,F/DYP,2)	00094100
	CALL NUMBER(301.,F/DYP-2.,4.,F,0.,-1)	00094200
9300	CONTINUE	00094300
	CALL NUMBER(301.,-2.,4.,0.,0.,-1)	00094400
	CALL SYMBOL(320.,90.,5.,'POWER (W/CM)',90.,12)	00094500
	CALL AXIST(MX,FLOAT(MN),MX/300.)	00094600
	CALL SYMBOL(100.,-20.,5.,'OPERATION TIME (HOUR)',0.,21)	00094700
	YO = 0.	00094800
	IF(YO-HABA*ISTNN.LT.0.) CALL PLOT(-22.,YO/DY,3)	00094900
	IF(YO-HABA*ISTNN.LT.0.) CALL PLOT(-20.,YO/DY,2)	00095000
	CALL NUMBER(-19.,-2.,4.,ABS(YO-HABA*ISTNN),0.,2)	00095100
	DO 9400 I=1,MSY	00095200
	YO = YO+HABA	00095300
	IF(YO-HABA*ISTNN.LT.0.) CALL PLOT(-22.,YO/DY,3)	00095400
	IF(YO-HABA*ISTNN.LT.0.) CALL PLOT(-20.,YO/DY,2)	00095500
	CALL NUMBER(-19.,YO/DY-2.,4.,ABS(YO-HABA*ISTNN),0.,2)	00095600
	CALL PLOT(0.,YO/DY,3)	00095700
	CALL PLOT(2.,YO/DY,2)	00095800
9400	CONTINUE	00095900
	CALL SYMBOL(-34.,80.,5.,'CLAD STRAIN (I)',90.,15)	00096000
	CALL SYMBOL(XTIME(1)/DXT,XPOW(1)/DYP,2.,0.,0.,-1)	00096100
	DO 9500 I=2,IT	00096200
	XO = XTIME(I)/DXT	00096300
	YO = XPOW(I)/DYP	00096400
	CALL PLOT(XO,YO,2)	00096500
	CALL SYMBOL(XO,YO,2.,0.,0.,-1)	00096600
9500	CONTINUE	00096700
	XS =-ISTNN*HABA	00096800
	CALL SYMBOL(XTIME(1)/DXT,(XCSTN(1)-XS)/DY,2.,4.,0.,-1)	00096900
	DO 9600 I=2,IT	00097000
	XO = XTIME(I)/DXT	00097100
	YO = (XCSTN(I)-XS)/DY	00097200
	CALL PLOT(XO,YO,2)	00097300
	CALL SYMBOL(XO,YO,2.,4.,0.,-1)	00097400
9600	CONTINUE	00097500
	CALL NEWPEN(2)	00097600
	CALL SYMBOL(XTIME(1)/DXT,(XTPC(1)-XS)/DY,2.,2.,0.,-1)	00097700
	DO 9610 I=2,IT	00097800
	XO = XTIME(I)/DXT	00097900
	YO = (XTPC(I)-XS)/DY	00098000
	CALL PLOT(XO,YO,2)	00098100
	CALL SYMBOL(XO,YO,2.,2.,0.,-1)	00098200
9610	CONTINUE	00098300
	CALL NEWPEN(3)	00098400
	CALL SYMBOL(XTIME(1)/DXT,(XTSC(1)-XS)/DY,2.,3.,0.,-1)	00098500
	DO 9620 I=2,IT	00098600

	XO = XTIME(I)/DXT	00098700
	YO = (XTSC(I)-XS)/DY	00098800
	CALL PLOT (XO,YO,2)	00098900
	CALL SYMBOL (XO,YO,2.,3,0.,-1)	00099000
9620	CONTINUE	00099100
	CALL NEWPEN(1)	00099200
	CALL SUBTIT(TITLE)	00099300
	CALL AXIAL (340.,180.,IPSTN)	00099400
	CALL SYMBOL (340.,165.,4.,4,0.,-1)	00099500
	CALL SYMBOL (342.,163.,4.,' -- CLAD STRAIN',0.,15)	00099600
	CALL NEWPEN(2)	00099700
	CALL SYMBOL (340.,155.,4.,2,0.,-1)	00099800
	CALL SYMBOL (342.,153.,4.,' -- HOOP CREEP STRAIN ACCUMULATED',	00099900
	* 0.,33)	00100000
	CALL NEWPEN(3)	00100100
	CALL SYMBOL (340.,143.,4.,3,0.,-1)	00100200
	CALL SYMBOL (342.,141.,4.,' -- CIRCUMFERENTIAL SWELLING STRAIN',	00100300
	* 0.,35)	00100400
	CALL SYMBOL (358.,135.,4.,'COMPONENT ACCUMULATED',0.,21)	00100500
	CALL NEWPEN(1)	00100600
	CALL SYMBOL (340.,110.,4.,0,0.,-1)	00100700
	CALL SYMBOL (342.,108.,4.,' -- POWER',0.,9)	00100800
	CALL AXIAL (360.,102.,IPFUL)	00100900
C****	PLUTONIUM CONCENTRATION	00101000
	CALL PLOT (560.,0.,-3)	00101100
	CALL PLOT (300.,0.,2)	00101200
	CALL PLOT (300.,220.,2)	00101300
	CALL PLOT (0.,220.,2)	00101400
	CALL SUBTIT(TITLE)	00101500
	XCPUMX=1.E-10	00101600
	DO 9700 I=1,IT	00101700
	DO 9700 J=1,NP	00101800
	IF(XCPU(I,J).GT.XCPUMX) XCPUMX=XCPU(I,J)	00101900
9700	CONTINUE	00102000
	DO 9710 I=1,100	00102100
	X=0.05*I	00102200
	IF(X.GT.XCPUMX) GO TO 9720	00102300
9710	CONTINUE	00102400
9720	CONTINUE	00102500
	MY = I*5	00102600
	DY = X/220.	00102700
	CALL AXISD (MXD,0.5,DXD)	00102800
	CALL SYMBOL (65.,-17.,5.,'DISTANCE FROM CENTER OF PELLET (MM)',	00102900
	* 0.,35)	00103000
	CALL PLOT (0.,0.,3)	00103100
	CALL PLOT (0.,220.,2)	00103200
	DO 9730 I=1,MY	00103300
	CALL NEWPEN(1)	00103400
	J=MY-I+1	00103500
	Y=0.01*J	00103600
	IF(MOD(J,5).NE.0) GO TO 9740	00103700
	CALL NUMBER (-17.,Y/DY-2.,4.,Y,0.,2)	00103800
	IF(I.EQ.1) GO TO 9730	00103900
	CALL NEWPEN(3)	00104000
	CALL PLOT (0.,Y/DY,3)	00104100
	CALL PLOT (300.,Y/DY,2)	00104200
	GO TO 9730	00104300
9740	CONTINUE	00104400

	CALL PLOT (0.,Y/DY,3)	00104500
	CALL PLOT (2.,Y/DY,2)	00104600
9730	CONTINUE	00104700
	CALL NEWPEN(1)	00104800
	CALL NUMBER (-17.,-2.,4.,0.,0.,-1)	00104900
	CALL SYMBOL (-23.,70.,5.,'PLUTONIUM CONCENTRATION',90.,23)	00105000
	DO 9800 I=1,IT	00105100
	ICODE=I-1	00105200
	ISW1=0	00105300
	ISW2=0	00105400
	IF(ICODE.GE.13) ICODE=13	00105500
	X2 = (XRL(I,1)+ XRL(I,2))/2.0	00105600
	DO 9900 J=2,NP	00105700
	X1= X2	00105800
	X2= (XRL(I,J)+ XRL(I,J+1))/2.0	00105900
	CALL SYMBOL (X1/DXD,XCPU(I,J-1)/DY,2.0,ICODE,0.,-1)	00106000
	IF(ISW1.EQ.1) GO TO 9910	00106100
	IF(X2.LT.XRLR(I,1)) GO TO 9910	00106200
	Y = XCPU(I,J) + (XCPU(I,J-1)-XCPU(I,J))*(X2-XRLR(I,1))/(X2-X1)	00106300
	CALL PLOT (XRLR(I,1)/DXD,Y/DY,2)	00106400
	CALL NEWPEN(2)	00106500
	CALL SYMBOL(XRLR(I,1)/DXD,Y/DY,2.,ICODE,0.,-1)	00106600
	CALL NEWPEN(1)	00106700
	ISW1=1	00106800
9910	CONTINUE	00106900
	IF(ISW2.EQ.1) GO TO 9920	00107000
	IF(X2.LT.XRLR(I,2)) GO TO 9920	00107100
	Y = XCPU(I,J) + (XCPU(I,J-1)-XCPU(I,J))*(X2-XRLR(I,2))/(X2-X1)	00107200
	CALL PLOT (XRLR(I,2)/DXD,Y/DY,2)	00107300
	CALL NEWPEN(3)	00107400
	CALL SYMBOL(XRLR(I,2)/DXD,Y/DY,2.,ICODE,0.,-1)	00107500
	CALL NEWPEN(1)	00107600
	ISW2=1	00107700
9920	CONTINUE	00107800
	CALL PLOT (X2/DXD,XCPU(I,J)/DY,2)	00107900
9900	CONTINUE	00108000
	CALL SYMBOL (X2/DXD,XCPU(I,NP)/DY,2.,ICODE,0.,-1)	00108100
9800	CONTINUE	00108200
	CALL NEWPEN(1)	00108300
	CALL SYMBOL (320.,200.,4.,'PLUTONIUM CONCENTRATION',0.,23)	00108400
	CALL AXIAL (320.,193.,IPFUL)	00108500
	CALL NEWPEN(2)	00108600
	CALL SYMBOL(320.,180.,4.,'COLUMNAR BNDRY',0.,14)	00108700
	CALL NEWPEN(3)	00108800
	CALL SYMBOL(320.,170.,4.,'EQUIAXED BNDRY',0.,14)	00108900
	CALL NEWPEN(1)	00109000
	DO 10000 I=1,IT	00109100
	IF(I.GE.15) GO TO 10010	00109200
	ITX = ITP(I)	00109300
	ICODE = I-1	00109400
	YY=160.-5.*(I-1)	00109500
	CALL SYMBOL(320.,YY,4.0,ICODE,0.,-1)	00109600
	CALL SYMBOL(323.,YY-2.0,4.0,' --- STEP',0.,9)	00109700
	CALL NUMBER(360.,YY-2.0,4.,FLOAT(ITX),0.,-1)	00109800
10000	CONTINUE	00109900
	GO TO 10020	00110000
10010	CONTINUE	00110100
	CALL SYMBOL (369.,93.,4.,30,0.,-1)	00110200

PNCT841-75-17

C	CALL NUMBER (374.,93.,4.,FLOAT(IT),0.,-1)	00110300
10020	CONTINUE	00110400
C		00110500
	CALL PLOT(600.,0.,-3)	00110600
	RETURN	00110700
	END	00110800

	SUBROUTINE PLUTO (CP,RVB,RCB,REB,RP,TP,GDT,XM,RH,EPZ,DELT,D)	00005400
C		00005500
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00005600
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00005700
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00005800
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00005900
*	N12C,N13C,N14C,N15C,	00006000
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00006100
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00006200
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00006300
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00006400
	COMMON /IOFIL/ IO5,IO6,IO7	00006500
C		00006600
	DIMENSION CP(NN,NP) , RVB(NN) , RCB(NN) , REB(NN) , RP(NN,NP1) ,	00006700
*	TP(NN,NP1) , GDT(NN,NP1) , XM(NN,3) , RH(NN,NTR) ,	00006800
*	EPZ(NN,NTR) , D(NP)	00006900
C		00007000
C		00007100
C	PLUTONIUM MIGRATION CALC.	00007200
C		00007300
C		00007400
	DO 1000 N=1,NNF	00007500
C		00007600
	ISW = 0	00007700
	S = DELT*3600.	00007800
	CX = XM(N,1)+XM(N,2)+XM(N,3)	00007900
900	CONTINUE	00008000
	I = 3	00008100
	B = 0.	00008200
	X = 0.0	00008300
C		00008400
	DO 1100 J=1,NP	00008500
C		00008600
	K = NP-J+1	00008700
	R = RP(N,K)	00008800
	T = TP(N,K)	00008900
C		00009000
	600 IF(I-2) 100,200,300	00009100
C		00009200
	100 RR = RVB(N)	00009300
	II = 1	00009400
	GO TO 400	00009500
C		00009600
	200 RR = RCB(N)	00009700
	II = NCR+1	00009800
	GO TO 400	00009900
C		00010000
	300 RR = REB(N)	00010100
	II = NCR+NER+1	00010200
C		00010300
	400 CONTINUE	00010400
C		00010500
	IF(R-RR) 450,500,500	00010600
450	I=I-1	00010700
	IF (I.EQ.0) GO TO 500	00010800
	GO TO 600	00010900
	500 CONTINUE	00011000
C		00011100

	XY = -1.E+05/(1.9865*T)	00011200
C	U = 1.+EPZ(N,II)	00011300
	U = 1.0	00011400
	H = CX/U/NP	00011500
	IF(XY+60. .LT. 0. .OR. K .EQ. 1) GO TO 700	00011600
C		00011700
	V = CP(N,K)-CP(N,K-1)	00011800
	W = (1.-(CP(N,K)+CP(N,K-1))/2.)*(CP(N,K)+CP(N,K-1))/2.	00011900
	IF(V.GT.W) ISW=1	00012000
	A = PI2*RP(N,K)*U*0.046*EXP(XY)*(W*11.0*GDT(N,K)/T -	00012100
	* 2.*V/((RP(N,K+1)-RP(N,K-1)))) * RH(N,II)	00012200
	GO TO 800	00012300
C		00012400
	700 A = 0.	00012500
C		00012600
	800 D(K) = (A-B) / H	00012700
C		00012800
	Y = ABS(D(K)/CP(N,K))	00012900
	IF(Y.GT.X) X=Y	00013000
	B = A	00013100
C		00013200
	1100 CONTINUE	00013300
C		00013400
	IF(X) 950,950,960	00013500
	950 DO 1200 J=1,NP	00013600
	CP(N,J) = CP(N,J) + D(J)*S	00013700
	1200 CONTINUE	00013800
	GO TO 1400	00013900
	960 P = 0.01/X	00014000
	IF(P.GT.S) GO TO 950	00014100
	DO 1300 J=1,NP	00014200
	CP(N,J) = CP(N,J) + D(J)*P	00014300
	1300 CONTINUE	00014400
	S = S-P	00014500
	GO TO 900	00014600
C		00014700
	1400 CONTINUE	00014800
C		00014900
	IF(ISW.EQ.0) GO TO 1000	00015000
	WRITE(I06,9000) N,(CP (N,J),J=1,NP)	00015100
	9000 FORMAT(IH1,'----- SUBROUTINE PLUTO ----- MASS INCREMENT TOO BIG --	00015200
	*---',/IHO,I3,/(IHO,10E12.4))	00015300
	1000 CONTINUE	00015400
C		00015500
C		00015600
	RETURN	00015700
	END	00015800

	SUBROUTINE POWCUT (TIM,TIP,TIN,PTIN,TOUT,PTOUT,REPOW,PPOWA,PTOUR,00015900	
*	TOUTER,DELTOU,DELTIM,DELIN,DELOUT,DELPow,KUT,NSK,KUTCNT)	00016000
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NDS,NOC,NPLOT,NCRK,	00016100
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00016200
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00016300
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00016400
*	N12C,N13C,N14C,N15C,	00016500
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00016600
C		00016700
	DIMENSION TOUTER(NN) , DELTOU(NN) , PTOUR(NN)	00016800
C		00016900
C		00017000
	IF (KUTCNT.GT.1) GO TO 10	00017100
	DELTIM = (TIM-TIP) / KUT	00017200
	DELIN = (TIN-PTIN) / KUT	00017300
	DELPow = (REPOW-PPOWA) / KUT	00017400
	IF(NSK.GT.0) GO TO 100	00017500
	DELOUT = (TOUT-PTOUT) / KUT	00017600
	GO TO 200	00017700
	100 DO 150 I=1,NN	00017800
	150 DELTOU(I) = (TOUTER(I)-PTOUR(I)) / KUT	00017900
	200 CONTINUE	00018000
C		00018100
C		00018200
C		00018300
	10 CONTINUE	00018400
	TIM = TIP + DELTIM	00018500
	TIN = PTIN + DELIN	00018600
	IF(NSK.GT.0) GO TO 300	00018700
	TOUT = PTOUT + DELOUT	00018800
	GO TO 400	00018900
	300 TOUT = PTOUR(NN) + DELTOU(NN)	00019000
	400 CONTINUE	00019100
	REPOW = PPOWA + DELPow	00019200
	RETURN	00019300
	END	00019400

	SUBROUTINE PROPTY (CPU,RHOP,RHOT)	00019500
C	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00019600
		00019700
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00019800
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00019900
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00020000
*	N12C,N13C,N14C,N15C,	00020100
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00020200
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00020300
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00020400
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00020500
	COMMON /CONST/ FKA1,FKB1,FKC1,FKA2,FKB2,FKC2,FKA3,FKB3,FKC3,	00020600
*	FA11,FA12,FA21,FA22,FA31,FA32,	00020700
*	CK1,CK2,CA1,CA2,XEWTN(7)	00020800
	COMMON /XCRAK/ THLC,HEAL1,HEAL2,SIGF0,SIGF1	00020900
	COMMON /XCREC/ CCA,QCC,AE,BE,CE,FMC,FNC	00021000
	COMMON /XCREF/ CFA0,CFA1,CFR,CFB,QCF,QCF1,FMF,FNF,GDF	00021100
	COMMON /XFRIC/ FRICST,FRICDY	00021200
	COMMON /XGASO/ AF,AN,CT,QRG	00021300
	COMMON /XGOLD/ ASSO,ASS1,ANAO,ANA1,ANA2,ACO,AC1,AC2	00021400
	COMMON /XHKNS/ HCS,HBO,HCO,HC1,HC2,HC3	00021500
	COMMON /XITER/ CI,QRK	00021600
	COMMON /XSWEL/ SVPF	00021700
	COMMON /XXLAM/ EO,EP,TLG,ONU,ONUP,EA1,EA2,EA3,EA4	00021800
	COMMON /FBRATR/ JFA	00021900
C		00022000
	DIMENSION CPU(NN,NP) , RHOP(NN,3) , RHOT(NN)	00022100
C		00022200
C		00022300
C	MATERIAL PROPERTIES , AND CONSTANTS SET UP	00022400
C		00022500
C		00022600
C		00022700
C		00022800
C	/OTHERS/	00022900
C		00023000
C		00023100
	PI = 3.1415927	00023200
	PI2 = 6.2831853	00023300
	DP = 1.	00023400
	GRAV = 1.E-03	00023500
	CFLX = 1.041E+14 * 12.*2.54 / 1000.	00023600
C	CFLX NOW IN (N/CM**2) / (W/CM)	00023700
	BUCNS=(3.1E+10)*3600./(2.246E+19)	00023800
C		00023900
C	BURNUP IS 1. AFTER 2.*E+19 FISSIONS PER GRAM OF FUEL.	00024000
C		00024100
	A91 = 1.116E+14	00024200
C		00024300
C	UNIT = 3.1*10**10FISSIONS/WATT/SEC*3600SEC/HR = 11.16E+13 FISSIONS	00024400
C	/WATT/HR	00024500
C		00024600
	RK = 8.31436*1.0197*10.	00024700
	RKR = 8.31436/1.8/4.186	00024800
C		00024900
C	R = 8.31436 JOULE/MOLE/DEG K , 1 JOULE = 1.0197*10. KG*CM	00025000
C	RK IS IDEAL GAS CONSTANT IN KG*CM/MOLE/DEG K	00025100
C	RKR IS IDEAL GAS CONSTANT IN CAL/MOLE/DEG R	00025200

C		00025300
C		00025400
C	CONVERSION TO KG/CM**2 FROM PSI	00025500
C		00025600
C		00025700
	DD 500 I=1,3	00025800
	PG(I) = PG(I) * 7.03E-02	00025900
500	CONTINUE	00026000
C	PG NOW IN KG/CM**2	00026100
C		00026200
C	/CONST/	00026300
C		00026400
C		00026500
	IF(JFA-2) 470,480,480	00026600
470	CONTINUE	00026700
	FA11 = 8.143E-06	00026800
	FA12 = 2.430E-09	00026900
	FA21 = 8.143E-06	00027000
	FA22 = 2.430E-09	00027100
	FA31 = 8.143E-06	00027200
	FA32 = 2.430E-09	00027300
	GO TO 490	00027400
480	FA11 = 9.61E-06	00027500
	FA12 = 1.57E-09	00027600
	FA21 = 9.61E-06	00027700
	FA22 = 1.57E-09	00027800
	FA31 = 9.61E-06	00027900
	FA32 = 1.57E-09	00028000
490	CONTINUE	00028100
C		00028200
	IF(JFA-2) 510,520,520	00028300
510	CK1 = 0.130	00028400
	CK2 = 1.36E-04	00028500
	CA1 = 17.0E-06	00028600
	CA2 = 4.25E-09	00028700
	GO TO 530	00028800
520	CK1 = 0.121	00028900
	CK2 = 1.24E-04	00029000
	CA1 = 5.675E-06	00029100
	CA2 = 1.7E-09	00029200
530	CONTINUE	00029300
C		00029400
C	RHOT FOR UO2 = 10.96 G/CC FOR U-PUO2 = 11.06 (20WT.PC) ROBERTS	00029500
C		00029600
	DO 1000 N=1,NNF	00029700
	DO 1000 I=2,NP	00029800
	CPU(N,I) = CPU(N,1)	00029900
1000	CONTINUE	00030000
C		00030100
	DO 1500 N=1,NNF	00030200
	RHOT(N) = 10.96 + 0.5*CPU(N,1)	00030300
	RHOP(N,1) = 0.767*RHOP(N,3) + 0.233*RHOT(N)	00030400
	RHOP(N,2) = RHOP(N,3)	00030500
1500	CONTINUE	00030600
	RHOT(NN) = 0.	00030700
C		00030800
C		00030900
C		00031000

C		00031100
C	/XCRAK/	00031200
C	== SUBROUTINE CRACK ==	00031300
C		00031400
C	THLC = 1400.	00031500
C	HEAL1 = -7000.	00031600
C	HEAL2 = -70.	00031700
C	SIGF0 = 1000.	00031800
C	SIGF1 = 0.3	00031900
C		00032000
C		00032100
C		00032200
C		00032300
C		00032400
C	/XCREC/	00032500
C	== SUBROUTINE CREEC ==	00032600
C		00032700
C	CCA = 2.7E-11	00032800
C	QCC = 95000.	00032900
C	AE = 1.0E-08	00033000
C	BE = 2.2E+20	00033100
C	CE = 4.3E-30	00033200
C	FMC = 7.0	00033300
C	FNC = 1.0	00033400
C		00033500
C		00033600
C		00033700
C	/XCREF/	00033800
C	== SUBROUTINE CREEF ==	00033900
C		00034000
C	CFA0 = 1.83E-03	00034100
C	CFA1 = 2.82E+07	00034200
C	CFA1 = 1.07E+06	00034300
C	CFR = 3.0E+12	00034400
C	CFR IN FISSIONS/SEC/CC	00034500
C	CFB = 6.0E-23	00034600
C	QCF = 140000.	00034700
C	QCF1 = 110000.	00034800
C	FMF = 4.5	00034900
C	FNF = 1.0	00035000
C	GDF = 10.0	00035100
C		00035200
C		00035300
C	/XGASO/	00035400
C	== SUBROUTINE GASOUT ==	00035500
C		00035600
C	AF = 0.246	00035700
C		00035800
C	AF IS ATOMS OF GAS PER FISSION	00035900
C		00036000
C	AN = 6.02E+23	00036100
C		00036200
C	AN IS AVOGADRO'S NUMBER	00036300
C		00036400
C	CT = 4.48	00036500
C	QRG = 8000.	00036600
C		00036700
C		00036800

C	/XFRIC/	== SUBROUTINE FRCHK ==	00036900
C			00037000
C			00037100
C	FRICST = 0.8		00037200
C	FRICDY = 0.6		00037300
C	FRICDY = 0.		00037400
C			00037500
C			00037600
C	/XGOLD/	== SUBROUTINE GOLDN ==	00037700
C			00037800
C			00037900
	ASS0 = 9.18 * 252. / (12.*2.54)		00038000
	ASS1=4.214E-3 * 252. / (12.*2.54)		00038100
	ANAO=54.306 * 252. / (12.*2.54)		00038200
	ANA1=-0.01878 * 252. / (12.*2.54)		00038300
	ANA2=2.0914E-6 * 252. / (12.*2.54)		00038400
	AC0 = 0.345696 * 252. / 0.4536		00038500
	AC1 = -0.79211E-04 * 252. / 0.4536		00038600
	AC2 = 3.41178E-08 * 252. / 0.4536		00038700
C			00038800
C			00038900
C			00039000
C			00039100
C	/XHKNS/	== SUBROUTINE HKNS ==	00039200
C			00039300
C			00039400
	HCS = 10.0		00039500
	HBO = 9.0E-35		00039600
	HCO = 4.028		00039700
	HC1 = -3.712E-02		00039800
	HC2 = 1.0145E-04		00039900
	HC3 = -7.879E-08		00040000
C			00040100
C			00040200
C	/XITER/	== SUBROUTINE ITERAT ==	00040300
C			00040400
C			00040500
	CI = 11.13E+06		00040600
C	CI NOW IN DEG K *CM**2/HR/KG , 1 KG = 9.8E+05 DYNE		00040700
	QRK = 49600.		00040800
C			00040900
C			00041000
C	/XXLAM/	== SUBROUTINE XLAME ==	00041100
C			00041200
C			00041300
	EO = 3.59E+07 * 7.03E-02		00041400
	EP = -2.35		00041500
	TLG = 22200.		00041600
	ONU = 0.317		00041700
	ONUP = -0.46		00041800
C			00041900
	IF(JFA-2) 1910,1920,1920		00042000
1910	CONTINUE		00042100
	EA1 = 2.0793E+06		00042200
	EA2=-8.8592E+02		00042300
	EA3=0.3015		00042400
	EA4=0.850E-04		00042500
	GO TO 1930		00042600

1920	EA1=9.898E+05	00042700
	EA2 = -9.265E+02	00042800
	EA3 = 0.325	00042900
	EA4 = 2.14E-04	00043000
1930	CONTINUE	00043100
C		00043200
C	/XSWEL/ == SUBROUTINE SWELL ==	00043300
C		00043400
C		00043500
	SVPF = 0.70E-24	00043600
C		00043700
C	NET SOLID VOLUME CREATED PER FISSION ...	00043800
C		00043900
C		00044000
C		00044100
	RETURN	00044200
	END	00044300

	SUBROUTINE RHOMOD (RH,RHOP,EPZ,RVB,RCB,REB,RUB,BU,XM,RHTHEO,RDIS,00044400	
*	DXM,RVBO,RCBO,REBO,RUBO,RCI,RCO,XMINT,EPC,00044500	
*	RDIST,EPR,XEPR,XEPC,R,WOLD,WNEW,IT,DELT)00044600	
C		00044700
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,00044800	
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,00044900	
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,00045000	
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,00045100	
*	N12C,N13C,N14C,N15C,00045200	
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N1500045300	
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),00045400	
*	PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,00045500	
*	CPOW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB00045600	
	COMMON /IOFIL/ IO5,IO6,IO700045700	
	DIMENSION RH(NN,NTR) , RHOP(NN, 3) , EPZ(NN,NTR) ,00045800	
*	RVB(NN) , RCB(NN) , REB(NN) , RUB(NN) , BU(NN) ,00045900	
*	XM(NN,3) , RHTHEO(NN) , RDIS(NN,NTR2) , DXM(NN,3) ,00046000	
*	RVBO(NN) , RCBO(NN) , REBO(NN) , RUBO(NN) ,00046100	
*	RCI(NN) , RCO(NN) , XMINT(NN) , EPC(NN,NTR) ,00046200	
*	RDIST(NN,4) , EPR(NN,NTR)00046300	
	DIMENSION R(NN,NFR1) , XEPR(NFR) , XEPC(NFR) , WNEW(NFR) ,00046400	
*	WOLD(NN,NFR)00046500	
C		00046600
C		00046700
C	DETERMINATION OF EACH ZONE BOUNDARY00046800	
C		00046900
C		00047000
	DO 1000 N=1,NNF00047100	
C		00047200
	DXM(N,1) = DXM(N,1) * DELT * 3600.00047300	
	DXM(N,2) = DXM(N,2) * DELT * 3600.00047400	
	DXM(N,3) = DXM(N,3) * DELT * 3600.00047500	
	RDIST(N,3) = RDIST(N,3) + DXM(N,3)00047600	
	RDIST(N,2) = RDIST(N,2) + DXM(N,2)00047700	
	RVB(N) = RVB(N) + RDIS(N,1)00047800	
	RCB(N) = RCB(N) + RDIS(N,NCR+1) + DXM(N,2)00047900	
	REB(N) = REB(N) + RDIS(N,NCR+NER+1) + DXM(N,3)00048000	
	IF(RVB(N).LT.RVBO(N)) RVB(N)=RVBO(N)00048100	
	IF(RCB(N).LT.RCBO(N)) RCB(N)=RCBO(N)00048200	
	IF(REB(N).LT.REBO(N)) REB(N)=REBO(N)00048300	
	RUB(N) = RUB(N) + RDIS(N,NFR+1)00048400	
	RCI(N) = RCI(N) + RDIS(N,NFR+2)00048500	
	RCO(N) = RCO(N) + RDIS(N,NTR+2)00048600	
	IF(REB(N).GE.RUB(N)) REB(N) = RUB(N)-0.00100048700	
	IF(RCB(N).GE.REB(N)) RCB(N) = REB(N)-0.00100048800	
	RHOT = RHTHEO(N)00048900	
	SM = XMINT(N) / (1.+EPZ(N,1))00049000	
C		00049100
	DO 1100 I=1,300049200	
C		00049300
	IF(I-2) 100,200,30000049400	
100	KK=100049500	
	KKK=NCR00049600	
	RI=RVB(N)00049700	
	RO = RCB(N)00049800	
	RIO = RVBO(N)00049900	
	ROO = RCBO(N)00050000	
	COF = 1.00050100	

	RAT = 0.99	00050200
	BUR = BURH1	00050300
	GO TO 400	00050400
200	KK=NCR+1	00050500
	KKK=NCR+NER	00050600
	RI=RCB(N)	00050700
	RO = REB(N)	00050800
	RIO = RCBO(N)	00050900
	ROO = REBO(N)	00051000
	COF = 1.	00051100
	RAT = 0.965	00051200
	BUR = BURH2	00051300
	GO TO 400	00051400
300	KK=NCR+NER+1	00051500
	KKK=NFR	00051600
	RI=REB(N)	00051700
	RO = RUB(N)	00051800
	RIO = REBO(N)	00051900
	ROO = RUBO(N)	00052000
	COF = 0.	00052100
	RAT = 1.	00052200
	BUR = 1.0	00052300
400	CONTINUE	00052400
C		00052500
	ERAV = 0.	00052600
	ECAV = 0.	00052700
	IF(IT.LE.1) GO TO 501	00052800
	DO 500 K=KK,KKK	00052900
	ERAV = ERAV+EPR(N,K)	00053000
	ECAV = ECAV+EPC(N,K)	00053100
500	CONTINUE	00053200
	ERAV = ERAV/(KKK-KK+1)	00053300
	ECAV = ECAV/(KKK-KK+1)	00053400
501	CONTINUE	00053500
	CX = (1.+ERAV)*(1.+ECAV)	00053600
C	CX = 1.+ERAV+ECAV+EPZ(N,1)	00053700
C		00053800
C		00053900
C	BURN-UP DEPENDENT DENSITY CALC. AND AXIAL MODIFICATION	00054000
C		00054100
C		00054200
	DO 1200 K=KK,KKK	00054300
	RH(N,K) = RHOP(N,I) + COF*(RAT*RHOT-RHOP(N,I)) * BU(N)/BUR	00054400
	RH(N,K) = RH(N,K) / ((1.+EPZ(N,K)) * CX)	00054500
C	RH(N,K) = RH(N,K) / CX	00054600
1200	CONTINUE	00054700
C		00054800
	IF (I.EQ.1) GO TO 1100	00054900
	XM(N,I) = PI*(RO+RI)*(RO-RI)*RH(N,KK)	00055000
C		00055100
1100	CONTINUE	00055200
	ROIST(N,4) = 1.+EPZ(N,1)	00055300
	XM(N,1) = SM-XM(N,2)-XM(N,3)	00055400
C		00055500
C	REGION BOUNDARY UPDATE	00055600
C		00055700
C	****	00055800
	IF(XM(N,1).GE.0.0001) GO TO 1300	00055900

	XM(N,1) = 0.0001	00056000
1300	XAR = RCB(N)*RCB(N)-XM(N,1)/(PI*RH(N,1))	00056100
	IF(XAR.LE.0.) XAR = 0.0	00056200
	RVB(N) = SQRT(XAR)	00056300
	IF(RVB(N).GE.RVBO(N)) GO TO 1400	00056400
	RVB(N) = RVBO(N)	00056500
1400	XM(N,1) = PI*RH(N,1)*(RCB(N)*RCB(N)-RVB(N)*RVB(N))	00056600
	IF(XM(N,1).GT.0.) GO TO 1405	00056700
	WRITE(106,9000) N	00056800
	STOP	00056900
1405	CONTINUE	00057000
C		00057100
	SMD = XM(N,1)+XM(N,2)+XM(N,3)	00057200
C		00057300
	DO 1500 I=1,3	00057400
	IF(I-2) 1410,1420,1430	00057500
1410	KK = 1	00057600
	KKK = NCR	00057700
	GO TO 1440	00057800
1420	KK = NCR+1	00057900
	KKK = NCR+NER	00058000
	GO TO 1440	00058100
1430	KK = NCR+NER+1	00058200
	KKK = NFR	00058300
1440	CONTINUE	00058400
	DO 1450 K=KK,KKK	00058500
	RH(N,K) = RH(N,K)*SMD/SM	00058600
1450	CONTINUE	00058700
C		00058800
	XM(N,I) = XM(N,I)*SMD/SM	00058900
C		00059000
1500	CONTINUE	00059100
C	****	00059200
C		00059300
	RDIST(N,4) = SM	00059400
1000	CONTINUE	00059500
C		00059600
C	PLENUM	00059700
C		00059800
	RCI(NN) = RCI (NN) + RDIS(NN,NFR+2)	00059900
	RCD(NN) = RCD (NN) + RDIS(NN,NTR+2)	00060000
C		00060100
	RETURN	00060200
C		00060300
9000	FORMAT(1H1,'----- SUBROUTINE RHOMOD -----', / ,	00060400
	* 1H0,'MASS IN COLUMNAR ZONE OF AXIAL SECTION ',I2,' IS NEGATIVE')	00060500
C		00060600
	END	00060700

SUBROUTINE SUBTIT (TITLE)	00060800
DIMENSION TITLE(1)	00060900
CALL SYMBOL (3.0,222.0,3.0,TITLE,0.,80)	00061000
CALL SYMBOL (250.,222.,5.,'ACTIVE-II',0.,9)	00061100
RETURN	00061200
END	00061300

	SUBROUTINE SWELL (POW,SWS,FD,SWS3,DELT)	00061400
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLDT,NCRK,	00061500
*	NTR,NFR,NTR1,NTR2,NNTR2,MNF,NCL1,NFR1,NP1,	00061600
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00061700
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00061800
*	N12C,N13C,N14C,N15C,	00061900
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00062000
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00062100
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00062200
*	CPW,BUCNS,RK,RKR,A91,GAST,TCON,PDBU,BUB	00062300
	COMMON /XSWEL/ SVPF	00062400
	DIMENSION POW(NN) , SWS(NN,NTR) , FD(NN,NFR) , SWS3(NN,NUR)	00062500
C		00062600
C		00062700
C	FUEL SWELLING STRAIN DUE TO SOLID FISSION PRODUCTS	00062800
C		00062900
C		00063000
C	SVPF = NET SOLID VOLUME CREATED PER FISSION	00063100
C		00063200
	DO 1000 N=1,NNF	00063300
	DFN = A91*POW(N)*DELT	00063400
C		00063500
C	DFN IS THE NUMBER OF FISSIONS IN THE CROSS SECTION OF FUEL ROD	00063600
C	OF ONE CM IN HEIGHT	00063700
C		00063800
	J = 1	00063900
	DO 1100 I=1,NFR	00064000
	DSWS = SVPF*DFN*FD(N,I)*2.54*2.54*2.54	00064100
	IF(I.LE.NCR+NER) GO TO 100	00064200
	SWS3(N,J) = SWS3(N,J) + DSWS	00064300
	J = J+1	00064400
	GO TO 1100	00064500
100	CONTINUE	00064600
	SWS(N,I) = SWS(N,I) + DSWS	00064700
1100	CONTINUE	00064800
1000	CONTINUE	00064900
	RETURN	00065000
	END	00065100

PNCT841-75-17

SUBROUTINE TIMCUT(CX,DELTO,EPSPOW)	00120600
CX = 2.	00120700
DELTO = 240.0	00120800
EPSPOW = 50.0	00120900
RETURN	00121000
END	00121100

	SUBROUTINE UPDATE (SIGR,SIGC,SIGZ,SIGRO,SIGCO,SIGZO,EPR,EPC,	00065200
*	EPZF,EPZC,EPRO,EPCO,EPZO,TPR,TPC,TPZ,TPRO,	00065300
*	TPCO,TPZO,DSR,DSC,DSZ,TSRO,TSCO,TSZO,DISP,	00065400
*	RDIS,N,EPSEQ)	00065500
C		00065600
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00065700
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00065800
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00065900
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00066000
*	N12C,N13C,N14C,N15C,	00066100
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00066200
C		00066300
	DIMENSION SIGR(NTR) , SIGC(NTR) , SIGZ(NTR) , SIGRO(NN,NTR) ,	00066400
*	SIGCO(NN,NTR) , SIGZO(NN,NTR) , EPR(NTR) , EPC(NTR) ,	00066500
*	EPRO(NN,NTR) , EPCO(NN,NTR) , EPZO(NN,NTR) , TPR(NTR) ,	00066600
*	TPC(NTR) , TPZ(NTR) , TPRO(NN,NTR) , TPCO(NN,NTR) ,	00066700
*	TPZO(NN,NTR) , DSR(NTR) , DSC(NTR) , DSZ(NTR) ,	00066800
*	TSRO(NN,NTR) , TSCO(NN,NTR) , TSZO(NN,NTR) ,	00066900
*	DISP(NTR2) , RDIS(NN,NTR2) , EPSEQ(NN,NTR)	00067000
C		00067100
C		00067200
C		00067300
C	UPDATE SOME VALUES FOR NEXT TIME STEP	00067400
C		00067500
C		00067600
C		00067700
	DO 1000 I=1,NTR	00067800
	SIGRO(N,I) = SIGR(I)	00067900
	SIGCO(N,I) = SIGC(I)	00068000
	SIGZO(N,I) = SIGZ(I)	00068100
	EPRO(N,I) = EPR(I)	00068200
	EPCO(N,I) = EPC(I)	00068300
C		00068400
	XXX = (EPR(I)+EPC(I)+EPZF)/3.	00068500
	X1 = TPR(I) - TPRO(N,I)	00068600
	XX1 = EPR(I)-TPRO(N,I)-XXX	00068700
	TPRO(N,I) = TPR(I)	00068800
	TPR(I) = X1	00068900
	X2 = TPC(I) - TPCO(N,I)	00069000
	XX2 = EPC(I)-TPCO(N,I)-XXX	00069100
	TPCO(N,I) = TPC(I)	00069200
	TPC(I) = X2	00069300
	X3 = TPZ(I) - TPZO(N,I)	00069400
	XX3 = EPZF-TPZO(N,I)-XXX	00069500
	TPZO(N,I) = TPZ(I)	00069600
	TPZ(I) = X3	00069700
	XXX = 2.*((XX1-XX2)**2+(XX2-XX3)**2+(XX3-XX1)**2)	00069800
	EPSEQ(N,I) = SQRT(XXX)/3.0	00069900
C		00070000
	IF (I-NFR) 100,100,200	00070100
100	EPZO(N,I) = EPZF	00070200
	GO TO 1000	00070300
200	EPZO(N,I) = EPZC	00070400
C		00070500
	1000 CONTINUE	00070600
C		00070700
C		00070800
	DO 2000 I=1,NTR2	00070900

PNCT841-75-17

RDIS(N,I) = DISP(I)	00071000
2000 CONTINUE	00071100
C	00071200
C	00071300
RETURN	00071400
END	00071500

	SUBROUTINE UPDATP (SIGR,SIGC,SIGZ,SIGRO,SIGCO,SIGZO,EPR,EPC,	00071600
*	EPZC,EPRO,EPCO,EPZO,TPR,TPC,TPZ,TPRO,TPCO,	00071700
*	TPZO,TSR,TSC,TSZ,TSRO,TSCO,TSZO,DISP,RDIS,N)	00071800
C		00071900
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00072000
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00072100
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00072200
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00072300
*	N12C,N13C,N14C,N15C,	00072400
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00072500
C		00072600
	DIMENSION SIGR(NCL) , SIGC(NCL) , SIGZ(NCL) , SIGRO(NN,NTR) ,	00072700
*	SIGCO(NN,NTR) , SIGZO(NN,NTR) , EPR(NCL) , EPC(NCL) ,	00072800
*	EPRO(NN,NTR) , EPCO(NN,NTR) , EPZO(NN,NTR) , TPR(NCL) ,	00072900
*	TPC(NCL) , TPZ(NCL) , TPRO(NN,NTR) , TPCO(NN,NTR) ,	00073000
*	TPZO(NN,NTR) , TSR(NCL) , TSC(NCL) , TSZ(NCL) ,	00073100
*	TSRO(NN,NTR) , TSCO(NN,NTR) , TSZO(NN,NTR) ,	00073200
*	DISP(NCL1) , RDIS(NN,NTR2)	00073300
C		00073400
C		00073500
C		00073600
C	UPDATE SOME VALUES FOR NEXT TIME STEP --- PLENUM SECTION ONLY	00073700
C		00073800
C		00073900
C	TPR,TPC,TPZ,TSR,TSC,TSZ ARE USED TEMPORARY FOR INCREMENT OF	00074000
C	STRAIN RESPECTIVELY	00074100
C		00074200
C		00074300
	DO 1000 I=1,NCL	00074400
	K = NFR+I	00074500
	SIGRO(N,K) = SIGR(I)	00074600
	SIGCO(N,K) = SIGC(I)	00074700
	SIGZO(N,K) = SIGZ(I)	00074800
	EPRO(N,K) = EPR(I)	00074900
	EPCO(N,K) = EPC(I)	00075000
	XXX = TPR(I) - TPRO(N,K)	00075100
	TPRO(N,K) = TPR(I)	00075200
	TPR(I) = XXX	00075300
	XXX = TPC(I) - TPCO(N,K)	00075400
	TPCO(N,K) = TPC(I)	00075500
	TPC(I) = XXX	00075600
	XXX = TPZ(I) - TPZO(N,K)	00075700
	TPZO(N,K) = TPZ(I)	00075800
	TPZ(I) = XXX	00075900
	EPZO(N,K) = EPZC	00076000
	RDIS(N,K+1) = DISP(I)	00076100
C		00076200
	1000 CONTINUE	00076300
C		00076400
	RDIS(N,NTR2) = DISP(NCL1)	00076500
C		00076600
	RETURN	00076700
	END	00076800

SUBROUTINE UPDTE (RVB,RCB,REB,RUB,RCI,RCO,BU,RDIST,DXM,	00109300
* FLNC,WOLD,YRVB,YRCB,YREB,YRUB,YRCI,YRCO,	00109400
* YBU,YRDIST,YDXM,YFLNC,YWOLD,ISW)	00109500
DIMENSION RVB(NN),RCB(NN),REB(NN),RUB(NN),RCI(NN),RCO(NN),	00109600
* BU(NN),RDIST(NN,4),DXM(NN,3),FLNC(NN),WOLD(NN,NFR),	00109700
* YRVB(NN),YRCB(NN),YREB(NN),YRUB(NN),YRCI(NN),	00109800
* YRCO(NN),YBU(NN),YRDIST(NN,4),YDXM(NN,3),YFLNC(NN),	00109900
* YWOLD(NN,NFR)	00110000
COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00110100
* NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00110200
* IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00110300
* N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00110400
* N12C,N13C,N14C,N15C,	00110500
* N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00110600
C	00110700
C	00110800
GO TO (1,2) , ISW	00110900
C	00111000
1 CONTINUE	00111100
C	00111200
DO 100 N=1,NN	00111300
YRVB(N) = RVB(N)	00111400
YRCB(N) = RCB(N)	00111500
YREB(N) = REB(N)	00111600
YRUB(N) = RUB(N)	00111700
YRCI(N) = RCI(N)	00111800
YRCO(N) = RCO(N)	00111900
YBU(N) = BU(N)	00112000
YFLNC(N) = FLNC(N)	00112100
DO 110 J=1,4	00112200
YRDIST(N,J) = RDIST(N,J)	00112300
110 CONTINUE	00112400
DO 120 J=1,3	00112500
YDXM(N,J) = DXM(N,J)	00112600
120 CONTINUE	00112700
DO 130 J=1,NFR	00112800
YWOLD(N,J) = WOLD(N,J)	00112900
130 CONTINUE	00113000
C	00113100
100 CONTINUE	00113200
RETURN	00113300
C	00113400
2 CONTINUE	00113500
C	00113600
DO 200 N=1,NN	00113700
RVB(N) = YRVB(N)	00113800
RCB(N) = YRCB(N)	00113900
REB(N) = YREB(N)	00114000
RUB(N) = YRUB(N)	00114100
RCI(N) = YRCI(N)	00114200
RCO(N) = YRCO(N)	00114300
BU(N) = YBU(N)	00114400
FLNC(N) = YFLNC(N)	00114500
DO 210 J=1,4	00114600
RDIST(N,J) = YRDIST(N,J)	00114700
210 CONTINUE	00114800
DO 220 J=1,3	00114900
DXM(N,J) = YDXM(N,J)	00115000

220 CONTINUE	00115100
DO 230 J=1,NFR	00115200
WOLD(N,J) = YWOLD(N,J)	00115300
230 CONTINUE	00115400
200 CONTINUE	00115500
C	00115600
RETURN	00115700
END	00115800

	SUBROUTINE VOLUME (VOL,RFR,RCL,FLZ,CLZ)	00076900
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NCS,NOC,NPLOT,NCRK,	00077000
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00077100
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00077200
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00077300
*	N12C,N13C,N14C,N15C,	00077400
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00077500
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALFO(3),ALF1(3),XML(3),CM(2),	00077600
*	PI,PI2,DP,GRAV,SMQ,SMQO,RHOC,BURH1,BURH2,CFLX,	00077700
*	CPDW,BUCNS,RK,RKR,A91,GASI,TCON,PDBU,BUB	00077800
	DIMENSION VOL(NN,NTR) , RFR(NN,NFR1) , RCL(NN,NCL1) ,	00077900
*	FLZ(NN) , CLZ(NN)	00078000
C		00078100
C		00078200
C	VOLUME CALC.	00078300
C		00078400
C		00078500
	DO 1000 N=1,NNF	00078600
C		00078700
C	FUEL	00078800
C		00078900
	DO 1100 I=1,NFR	00079000
	VOL(N,I) = PI*(RFR(N,I+1)+RFR(N,I))*(RFR(N,I+1)-RFR(N,I))*FLZ(N)	00079100
1100	CONTINUE	00079200
C		00079300
C	CLAD	00079400
C		00079500
	DO 1200 I=1,NCL	00079600
	VOL(N,NFR+I) = PI*(RCL(N,I+1)+RCL(N,I))*(RCL(N,I+1)-RCL(N,I)) *	00079700
*	* CLZ(N)	00079800
1200	CONTINUE	00079900
C		00080000
C		00080100
1000	CONTINUE	00080200
C		00080300
C		00080400
C	PLENUM	00080500
C		00080600
	DO 2000 I=1,NCL	00080700
	VOL(NN,NFR+I) = PI*(RCL(NN,I+1)+RCL(NN,I))*(RCL(NN,I+1)-RCL(NN,I))	00080800
*	* CLZ(NN)	00080900
2000	CONTINUE	00081000
C		00081100
C		00081200
	RETURN	00081300
	END	00081400

	SUBROUTINE XLAME(TBC,TBF,E,P,G,RH,LM,BM,PB,GB,VOL,RHTHEO,N)	00081500
	COMMON /ALWAYS/ NN,NCR,NER,NUR,NCL,NP,NOK,NOS,NOC,NPLOT,NCRK,	00081600
*	NTR,NFR,NTR1,NTR2,NNTR2,NNF,NCL1,NFR1,NP1,	00081700
*	IGEN,IGEN1,NONZR,IGENP,IGENP1,NONZRP,	00081800
*	N1C,N2C,N3C,N4C,N5C,N6C,N7C,N8C,N9C,N10C,N11C,	00081900
*	N12C,N13C,N14C,N15C,	00082000
*	N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15	00082100
	COMMON /OTHERS/ PG(3),CM1(2),DHR(2),ALF0(3),ALF1(3),XML(3),CM(2),	00082200
*	PI,PI2,DP,GRAV,SMQ,SMQ0,RHOC,BURH1,BURH2,CFLX,	00082300
*	CPOW,BUCNS,RK,RKR,A91,GAS1,TCON,PDBU,BUB	00082400
	COMMON /XXLAM/ EO,EP,TLG,CNU,ONUP,A1,A2,A3,A4	00082500
	COMMON /FBRATR/ JFA	00082600
	COMMON /IOFIL/ IO5,IO6,IO7	00082700
	DIMENSION TBC(NN,NCL) , TBF(NN,NFR) , E(NN,NTR) , P(NN,NTR) ,	00082800
*	G(NN,NTR) , RH(NN,NTR) , LM(NN,NTR) , BM(NN,NTR) ,	00082900
*	PB(NN,2) , GB(NN,2) , VOL(NN,NTR) , RHTHEO(NN)	00083000
C		00083100
C		00083200
C	YOUNG'S MODULUS,POISSON'S RATIO,SHEAR MODULUS , AND BULK MODULUS	00083300
C		00083400
C		00083500
C		00083600
C	GIVEN VALUES OF EP,ONU,ONUP PERTAIN TO UD2 . GIVEN EO AND TLG	00083700
C	REFER TO (U.8 PU.2)02 .0. FOR UD2 EO=3.16 AND TLG=16300.	00083800
C		00083900
C		00084000
C		00084100
C	FUEL	00084200
C		00084300
C		00084400
C	IF (N.EQ.NN) GO TO 500	00084500
C		00084600
	RHOT = RHTHEO(N)	00084700
	J = 0	00084800
	K = 1	00084900
	KK = NCR	00085000
C		00085100
	SP = 0.	00085200
	SG = 0.	00085300
	SV = 0.	00085400
C		00085500
C		00085600
C	400 CONTINUE	00085700
C		00085800
	DO 1100 I=K,KK	00085900
	LML = LM(N,I)	00086000
	T = TBF(N,I)*1.8 + 32.0	00086100
	IF(T-TLG+100. .GT. 0.) T=TLG-100.	00086200
	PS = 1.-RH(N,I)/RHOT	00086300
	EX = EO*(1.+EP*PS)*(1.-T/TLG)	00086400
	PX = ONU*(1.+ONUP*PS)	00086500
	P(N,I) = (PX*(TLG-T)+0.5*(T-77.)) / (TLG-77.)	00086600
	P(N,I) = P(N,I) / (2.)*LML	00086700
	E(N,I) = EX / (1.5)*LML	00086800
	G(N,I) = E(N,I) / (2.+2.*P(N,I))	00086900
	IF(G(N,I).LT.10.0) G(N,I)=10.0	00087000
	BM(N,I) = E(N,I) / (3.*(1.-2.*P(N,I)))	00087100
C		00087200

	SV = SV + VOL(N,I)	00087300
	SP = SP + P(N,I)*VOL(N,I)	00087400
	SG = SG + G(N,I)*VOL(N,I)	00087500
C		00087600
	1100 CONTINUE	00087700
	J = J+1	00087800
C		00087900
	IF(J-2) 100,200,300	00088000
	100 K=KK+1	00088100
	KK = NCR+NER	00088200
	GO TO 400	00088300
	200 K=KK+1	00088400
	KK = NFR	00088500
	GO TO 400	00088600
	300 CONTINUE	00088700
C		00088800
	PB(N,1) = SP / SV	00088900
	GB(N,1) = SG / SV	00089000
C		00089100
	500 CONTINUE	00089200
C		00089300
C	CLAD	00089400
C		00089500
	K=NFR	00089600
C		00089700
	SP = 0.	00089800
	SG = 0.	00089900
	SV = 0.	00090000
C		00090100
	DO 1200 I=1,NCL	00090200
	K = K+1	00090300
	TC = TBC(N,I)	00090400
	P(N,K) = A3+A4*TC	00090500
	IF(TC+A1/A2.GT.0) GO TO 1111	00090600
	E(N,K) = A1+A2*TC	00090700
	G(N,K) = E(N,K)/(2.+2.*P(N,K))	00090800
	BM(N,K) = E(N,K) / (3.*(1.-2.*P(N,K)))	00090900
C		00091000
	SV = SV + VOL(N,K)	00091100
	SP = SP + P(N,K)*VOL(N,K)	00091200
	SG = SG + G(N,K)*VOL(N,K)	00091300
C		00091400
	1200 CONTINUE	00091500
	PB(N,2) = SP / SV	00091600
	GB(N,2) = SG / SV	00091700
C		00091800
C		00091900
C	E . G . BM NOW IN KG/CM**2	00092000
C		00092100
C		00092200
	1000 CONTINUE	00092300
C		00092400
	RETURN	00092500
	1111 WRITE(106,1112) N	00092600
	1112 FORMAT(1H1,'----- SUBROUTINE XLAME -----' , / ,	00092700
	* 1H0,'CLADDING TEMPERATURE OF AXIAL SECTION ',I2,' IS TOO HIGH')	00092800
	STOP	00092900
	END	00093000