JAERI-M
5612

JFT－2aの平衡と局所不安定性・1
（一様電流分布の場合）

１９７４年3月

狐崎　晶雄・下村　安夫・前田　彦祐・吉川　允二

日本原子力研究所
Japan Atomic Energy Research Institute

ＪＦＴ－２ａの平衡と局所不安定性・１
（一様電流分布の場合）

日本原子力研究所東海研究所核融合研究室
狐崎　晶雄・下村　安夫・前田　彦祐・吉川　允二

（１９７４年２月受理）

高安定化磁場試験装置（ＪＦＴ－２ａあるいはＤＩＶＡ）の平衡量と局所不安定性の計算について述べる。本装置は，涙滴形断面トカマクの研究のみでなく，磁気リミターおよびダイバータの研究を目的としており，それらの設置を考慮して断面形状が選択された。ここに述べる計算プログラムは，解析式ψ（Ｒ，Ｚ）で与えられた平衡状態の平衡量と安定性を計算するものである。また，プラズマ・導体シェル間の真空領域を考慮するため，真空磁場の計算が容易に行なえるような形の平衡解を用いた。

Equilibrium and Localized Instability of JFT-2a. 1
(Flat Current Distribution)

Akio KITSUNEZAKI, Yasuo SHIMOMURA, Hikosuke MAEDA
and Masaji YOSHIKAWA

Thermonuclear Fusion Lab., Tokai, JAERI

(Received February 7, 1974)

Equilibrium and localized instability of the JFT-2a (DIVA) plasma have been studied numerically. JFT-2a is designed to examine the non-circular (tear drop) cross section tokamak and an axisymmetric divertor. Sectional form of the device is chosen to allow installation of the divertor and to enhance plasma stability. The computer code to calculate equilibrium values and examine localized instability criteria is described. Some results shown were obtained with the equilibrium solution of a special form in the plasma to apply a new method of calculating the vacuum magnetic field outside the plasma. Another computer code for vacuum magnetic field calculation and the results obtained with it are also described.

目 次 な し

1 はじめに

トカマク型プラズマ閉じ込め装置に関して現在までに得られた実験的成果のほとんどは円形断面をもつプラズマによって得られた。しかしながら，理論的検討によるとプラズマ断面の形を非円形にすると，涙滴形，長D形およびダブレット形等の場合には，円形断面の場合よりも高い電流密度まで安定性が保たれることが予想されている。[(1),(2)] この性質を生かすことができれば，小型で高温プラズマを保持する装置を作ることが可能であり，長D形およびダブレット形の実験はすでに開始されている。また，従来のトカマク実験では，プラズマを容器壁から離すために金属リミターが使用されているが，プラズマの温度が上るにつれてリミターの損傷およびリミターからの不純物の混入が大きな問題となり，現在すでに金属リミターの限界に近いと考えられている。この対策として，磁力線の形状によってリミター作用をさせる磁気リミターおよび不純物をとり除くダイバータが考案された。[(11)]

これらの設置の可能性，有効性の有無は将来の核融合炉の設計を大きく左右する。本研究室では現在可能な範囲内で涙滴形断面および軸対称磁気リミターあるいはダイバータの効果を実験的に研究するため，高安定化磁場試験装置（以下，JFT－2aと略記する）を設計，製作している。本装置の設計および計測のために各種の計算が行なわれているが，本論文では，精度の良い真空磁場の接続計算が容易に適用できる形のプラズマ平衡解の導出と，これを用いたプラズマ平衡量と局所不安定性の計算およびその結果について，また，プラズマ表面に接続する真空磁場の計算コードとその結果について述べる。

2 プラズマの平衡

2.1 平衡の方程式

軸対称トロイダル・プラズマの磁気流体的平衡は次の方程式で表わされる。

$$\frac{\partial^2 \Psi}{\partial R^2} - \frac{1}{R}\frac{\partial \Psi}{\partial R} + \frac{\partial^2 \Psi}{\partial Z^2} = \mu_0 \cdot R \cdot j_T \qquad (1)$$

ただし，Ψ は磁場の流れ関数（$2\pi\Psi$ が磁束を表わす），j_T はトロイダル方向の電流密度である。j_T はプラズマ圧力 P，トロイダル磁場 B_T と次の関係がある。

$$j_T = -R\frac{dP}{d\Psi} - \frac{1}{R} I \frac{dI}{d\Psi} \qquad (2)$$

$$I = R \cdot B_T \qquad (3)$$

$P(\Psi)$，$I(\Psi)$を定め，与えられた境界条件のもとで(1)式を解くことにより，トロイダル・プラズマの平衡状態が求められる。なお，磁場分布は Ψ を用いて次のように表わされる。

$$B_Z = \frac{1}{R}\frac{\partial \Psi}{\partial R} \qquad (4)$$

$$B_R = -\frac{1}{R}\frac{\partial\Psi}{\partial Z} \tag{5}$$

2.2 平 衡 解

平衡方程式(1)の解は，j_T の形により種々存在する。本論文に述べる平衡量計算のプログラムは，平衡の形を解析式で与えることにより，どのような平衡解にも応用できるが，実際に計算を行なうものは，以下に述べるものである。すなわち，$\frac{dP}{d\Psi}$，$I\frac{dI}{d\Psi}$ を定数とした場合を扱う。この形では，電流分布はRのみに依存する。（いわゆるflat current 分布）。この場合の解は，多くの形で得られているが，[(1)~(5)] 本論文では後述の精度のよい真空磁場の接続計算が容易に行なえるように，次の形の解を用いる。

$$\Psi_K = C_1R^2(R^2-4Z^2)+C_2R^2+C_3(R^2\ell nR-Z^2)+C_4-\frac{1}{2}aR^2Z^2 - \frac{1}{2}bZ^2 \tag{6}$$

ただし，$a=\mu_0\frac{dP}{d\Psi}$ ，　$b=\mu_0 I\frac{dI}{d\Psi}$

ここで，a, b, $C_1 \sim C_4$ は任意定数であるが，a，bは重要な物理量を表わすため，求めたい平衡状態に合わせて上記のように与える。$C_1 \sim C_4$は等Ψ曲線の形を与えるパラメータである。ここの計算では，プラズマの断面形状を特徴的に表わす4点で与え，この4点がプラズマ表面（Ψ＝0）上に存在するように$C_1 \sim C_4$ を求めている。

2.3 Ψ_K で表わされる形

任意定数$C_1 \sim C_4$の決定には，まず，a，bをプラズマ中心の半径R_0，プラズマ柱の平均半径$\bar{a}$，プラズマ電流値およびポロイダル・ベータ値から決める。

$$a = \mu_0 j\frac{\beta_{pi}}{R_0} \tag{7}$$

$$b = \mu_0 j\frac{R_0-\sqrt{R_0{}^2-\bar{a}^2}}{2}\cdot(1-\beta_{pi}) \tag{8}$$

ここで，$j=\frac{I_{pi}}{\pi\bar{a}^2}$

I_{pi}, β_{pi} ：プラズマ電流，ポロイダル・ベータの値の読込み値

次にプラズマ柱の中心（R_0, 0）を原点とする極座標表示（ρ, θ）を用いて，式

$$\rho^2(1-\varepsilon_2\cos 2\theta)-\frac{\varepsilon_3}{\bar{a}}\rho^3\cos 3\theta=\bar{a}^2 \tag{9}$$

ただし，ε_2 ：楕円変形率

ε_3 ：3角変形率

で表わされる曲線と，θ=0°，60°，120°，180°の直線との4交点を2.2に述べた4点として，$C_1 \sim C_4$ を求める。このようにして求めたΨ_K= 0の曲線は第1図に示すように4点

でしか(9)式の曲線と一致しないが，きわめて近い形になる。したがって，この方法でほぼ望む断面形状が得られる。またΨ_Kは(1)式の簡単な特解ではあるが，第2図，第3図に示すように，特徴的な各種の断面形状をもったプラズマ平衡状態を表わすことができる。

2.4 平衡量の計算

本プログラムは，$\Psi(R, Z)$，$\frac{\partial\Psi}{\partial R} = P(R, Z)$，$\frac{\partial\Psi}{\partial Z} = q(R, Z)$，および$\underline{P}(\Psi)$，$I^2(\Psi)$を解析式の形で与えて物理的に意味のある量の計算を行うものである。

(i) 等Ψ曲線の計算

R軸上でΨの極値となる点（磁気軸）を中心として角度等間隔で等Ψ点を求める。等Ψ線はオプションによりΨの値が一定間隔になるように，あるいは$\theta = 0$でのRが一定間隔になるように求める。（第4図）

(ii) 電流分布，磁場分布の計算

$$j_T(k, i) = R\frac{dP}{d\Psi} + \frac{1}{R}I\frac{dI}{d\Psi} \quad （トロイダル電流密度） \tag{10}$$

$$B_p(k, i) = \frac{1}{R}\sqrt{p^2 + q^2} \quad （ポロイダル磁場） \tag{11}$$

$$j_p(k, i) = -B_p \cdot \frac{dI}{d\Psi} \quad （ポロイダル電流密度） \tag{12}$$

$$B_T(k, i) = \frac{I}{R} \quad （トロイダル磁場） \tag{13}$$

(iii) 安全係数，シア，磁気井戸など

隣り合う等Ψ曲線の間を計算点を頂点とする3角形に分割して各3角形の面積を求め，これをもとに全断面積，全トロイダル電流，安全係数q，シアS，および磁気井戸Dを求める。

$$q(k) = \frac{\delta\phi(k)}{\delta\phi_p(k)} \tag{14}$$

$$S(k) = \Phi_T \cdot \frac{d}{d\phi}\left(\frac{1}{q}\right) \tag{15}$$

$$D(k) = \frac{V_1'(k) - V_1'(\mathrm{SURFACE})}{V_1'(k)} \tag{16}$$

ただし，$V'(k) = \frac{\delta V(k)}{\delta\phi(k)}$

$\delta V(k)$：第k番目の等Ψ曲面間の体積

$\delta\phi(k)$：第k番目の等Ψ曲面間のトロイダル磁束

$\delta\phi_p(k)$：第k番目の等Ψ曲面間のポロイダル磁束

Φ_T ：プラズマ断面中の全トロイダル磁束

$\frac{d}{d\phi}$ ：トロイダル磁束での微分

以上の結果を第5図，第6図に例示する。これらの結果から，涙滴型（$\varepsilon_2>0$，$\varepsilon_3<0$）および長D型（$\varepsilon_2<0$，$\varepsilon_3>0$）の場合には磁気井戸あるいはシアが円形に比べて大きく，安定性の増すことが予想される。なお，平均ポロイダル・ベータ値は次式によって求める。

$$\beta_p = \frac{\int -R\frac{dP}{d\varphi}\, dS}{\int j_T\, dS} \tag{17}$$

ただし，dSはプラズマ断面での積分を表わす。(17)式の値は，当初に読み込んだβ_{pi}とは異なるが，これに近い値となる。

3 安 定 性

前節までの計算をもとにして，局所不安定性に対するSolov′ev[14]の安定性係数を求めた。これには，2.4(iii)で使用した3角形を再び用いる。すなわち，第k番目の層の第j番目の3角形の重心（r_j，z_j）と面積S_jを用いて

$$U'(k) = \left(4\pi\sum_{j=1}^{M}\frac{S_j}{r_j}\right)/\triangle(k)$$

$$V_2'(k) = \left(4\pi\sum_{j=1}^{M} r_j S_j\right)/\triangle(k)$$

$$U''(k) = \left(U'(k+1) - U'(k)\right)/\triangle(k)$$

$$V''(k) = \left(V_2'(k+1) - V_2'(k)\right)/\triangle(k)$$

$$W(k) = \left(I_{k+1}\cdot U'(k+1) - I_k\cdot U'(k)\right)/\triangle(k)$$

$$SS(k) = -W(k)/\left(V'(k)\right)^3$$

$$Q1(k) = \left(4\pi\sum_{j=1}^{M}\frac{r_j}{p^2+q^2}\cdot S_j\right)/\triangle(k)$$

$$Q2(k) = \left(4\pi\sum_{j=1}^{M}\frac{1}{r_j(p^2+q^2)}\cdot S_j\right)/\triangle(k)$$

$$Q3(k) = \left(4\pi\sum_{j=1}^{M}\frac{r_j^3}{p^2+q^2}\cdot S_j\right)/\triangle(k)$$

$$Q4(k) = \left(4\pi\sum_{j=1}^{M} r_j^3 S_j\right)/\triangle(k)$$

$$Q5(k) = \left(4\pi \sum_{j=1}^{M} \frac{p^2+q^2}{r_j} \cdot S_j\right) / \triangle(k)$$

ただし，$\triangle(k) = \Psi(k+1) - \Psi(k)$

I，p，q は点（r_j，z_j）での値

これらを用いて，次の安定性係数を求める。

$$STL(k) = \frac{1}{4}(V_2'(k))^6 \cdot (SS(k))^2 - p'[I^2\{U''(k) \cdot Q1(k) - V''(k) \cdot Q2(k) + p'(Q2(k)Q3(k) - (Q1(k))^2)\} + U'(k)(I \cdot I' \cdot Q1(k) + p'Q3(k) - V''(k))] \quad (18)$$

安定性係数は等Ψ曲線間の層毎に求まり，プラズマ電流を増してゆくと，磁気軸の近傍から不安定になる。（注1） 第7図は変形パラメータε_2，ε_3を座標とする平面上の間隔 0.1 の網目点で安定に流せる最大電流を求め，これをもとにして$\varepsilon_2 - \varepsilon_3$ 平面上で補間（注2）した図である。ただし，$R_0 = 0.6\,m$，$\bar{a} = 0.1\,m$，$R = R_0$ でのトロイダル磁場10kG である。（注3）この結果によると，涙滴型断面および長D形断面の場合は，円形断面よりも多くの電流を流すことができ，その程度はβ_pの小さい程ど著しい。

以上の平衡量計算と不安定性計算のコードを付録1に示す。

（注1） 磁気軸を含む中心層は数値計算誤差が大きいため除外する。

（注2） CALCOMP GPCP（プロッタ用補間サブルーチン）

（注3） $q = 1$ は 83.3 kA に相当する。

4　JFT－2a の断面形状

JFT－2a は，非円形断面プラズマの安定性の検証と磁気リミターおよびダイバータ（軸対称）の検証を目的としており，形状の決定はこれらの目的に最も適したものを選択した。まず，円形トロイダルコイル内部の限られた空間で，プラズマとダイバータの空間を広くとれるように涙滴型とし，ダイバータ，真空容器の製作も考慮して，なるべく安定性の高い形状を選択した。（第8図） この形に沿って導体シェルを置けば，磁力線がシェル壁に沿い，プラズマの外側での磁気面（等Ψ曲面）の形が規定される。

Ψ_Kで計算したJFT－2a のシェル上での磁場分布を第9図に示す。シェル上での磁場分布は，磁気プローブによる測定からプラズマの状態を知る為に必要なものである。なお，β_pが大きいときは，$\Psi_K = 0$で表わされるプラズマ表面の形はシェルから外れるが，予想される実験値$\beta_p < 1$ の範囲内では，第10図のように最大 4mm である。また，次節に述べるように，真空領域を考慮した場合も，真空領域での等Ψ曲線はシェルから外れるが，$\beta_p < 1$ の範囲内では誤差は約2mmである。したがって，シェル上での磁場分布として，第9図の値を用いても大きな誤差は生じない。

5 真空磁場の計算

前節までは，ある等Ψ曲面に囲まれたトロイダル・プラズマ柱について調べたが，実際には，プラズマと導体シェルとの間には真空領域が存在する。真空領域での平衡状態（磁場分布）は，(1)式で右辺＝0とおいた方程式に従い，境界条件はプラズマ表面の磁場分布によって与えられる。プラズマ空間に接続する真空磁場を精度良く解くことは一般には困難であるが，(6)式の解でプラズマ表面磁場が与えられるときには，Harker[18] の方法を応用して求めることができる。[19],[20] この方法は楕円型偏微分方程式で記述される真空磁場のコーシー問題を解析接続法によって双曲型のコーシー問題に変換し，リーマン関数を求めることにより解の積分表示を得るものである。具体的には，まず，(6)式で与えられたプラズマ表面Ψ＝0の z＝0との交点 R_{p1}, R_{p2} を求め，$r_V=(R_{p1}+R_{p2})/2$, $a_V=(R_{p1}-R_{p2})/2$ を計算する。すると，真空領域の座標（T, V）と，（R, Z）座標は次のような関係にある。

$$\begin{cases} r_e \equiv r_V + a_V \cos(T+iV) \\ z_e \equiv f(r_e) \end{cases}$$

$$\begin{cases} R = \mathrm{Re}(r_e) - \mathrm{Im}(z_e) \\ Z = \mathrm{Im}(r_e) + \mathrm{Re}(z_e) \end{cases}$$

ただし，fは，$\Psi_K(R, Z)=0$ をzについて解いた式を表わし，V＝0がプラズマ境界をあらわす。このとき，座標（T, V）でのΨは次式で表わされる。

$$\Psi = \mathrm{Re}\int_0^V F\sqrt{\frac{R}{r_e}}\ \Psi_n\, dV$$

ただし，ここでΨ_nはプラズマ境界上での $\dfrac{\partial\Psi}{\partial V}$ を解析接続して得られる関数であり，また

$$F = 1+\sum_{n=1}^{\infty}\frac{\frac{3}{2}(\frac{3}{2}+1)\cdots(\frac{3}{2}+m-1)(-\frac{1}{2})(-\frac{1}{2}+1)\cdots(-\frac{1}{2}+m-1)}{(m!)^2}x^m$$

$$x = -\frac{(r_e-R)^2+(z_e-Z)^2}{4r_eR}$$

この方法で求めた真空磁場の例を第11図に示す。また，真空領域での等Ψ曲線がシェルに沿う計算例を第12図に示す。なお，真空磁場の計算プログラムを付録2に添える。

6 おわりに

JFT-2a装置の設計中に行なった計算の一部を述べた。本レポートの計算について，有益な助言を頂いた田中正俊氏，安積正史氏はじめ核融合研究室，理論グループの方々に感謝します。また，この研究に関し指導と激励をたまわった核融合研究室森室長に感謝いたします。

文　献

1) Jukes, J. D.　Nuclear Fusion 10, 131 (1970)

2) Laval, G. Pellat, R. Soule, J. L.　3rd International Symposium on Toroidal Plasma Confinement (Garching), 1973 Paper F2−I

3) Ohkawa, T.　Kakuyugo-Keokyu 22, 99 (1968)

4) Bazhanova, A. E. Shafranov, V. D.　MATT-TRANS-104 (1973)

5) Callen, J. D Dory, R. A.　ORNL-TM-3430 (1971)

6) Haas, F. A. Jukes, J. D.　CLN-P 301 (1972)

7) Gourdon, C. Touche, J.　EUR-CEA-FC 634 (1972)

8) Feneberg, W. Lackner, K.　IPP(Garching) III/7 6/118 (1973)

9) Grelot, P. Weisse, J.　6th European Conference on Controlled Fusion and Plasma Physic (Moscow) 1973 paper 1-4-2

10) Tanaka, M. Tuda, T.　JAERI-M 4532 (1971)

11) Yoshikawa, M.　JAERI-M 4494 (1972)

12) Kitsunezaki, A. Maeda, H. Shimomura, Y. Yoshikawa, M. 3rd International Symposium on Toroidal Plasma Confinement (Garching) 1973 paper G2

13) Yoshikawa, M. Shimomura, Y. Maeda, H. Kitsunezaki, A. 6th Furopean Conference on Controlled Fusion and Plasma Physics (Moscow) 1973 p173-176

14) Solov′ev, L. S.　JETP 26, 1167 (1968)

15) Solov′ev, L. S.　JETP 26, 400 (1968)

16) Mercier, C.　Nuclear Fusion 3, 89 (1963)

17) Marcier, C.　Nuclear Fusion 4, 213 (1964)

18) Harker, K. J.　J. Math. Phys. 4, 993 (1963)

19) 前田，狐崎，下村，吉川　日本物理学会年会 1973 5a−J−4

20) Maeda, H. Shimomura, Y. Kitsunezaki, A. Yoshikawa, M. to be published

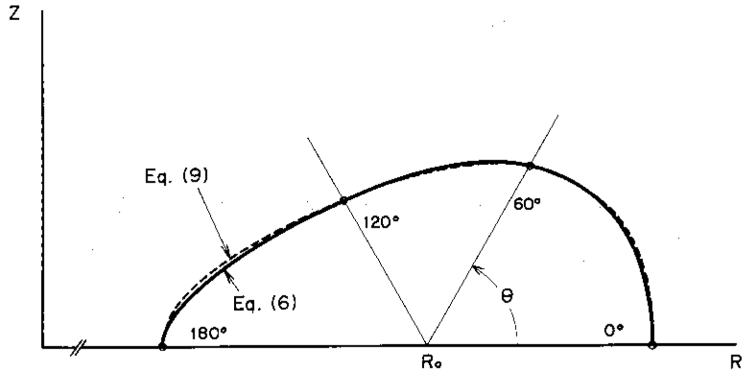


Fig. 1 Calculation of C_1—C_4 of Eq. (6)

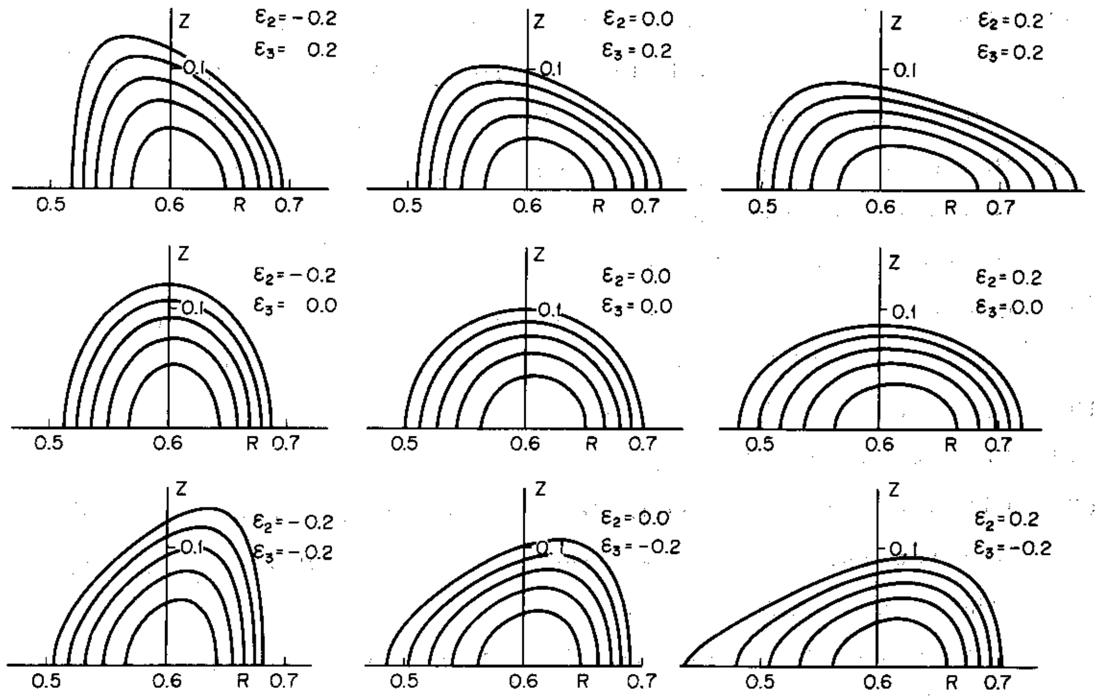


Fig. 2 Typical Shapes of Plasma Equilibrium expressed by Ψ_K (I)
($R_0 = 0.6$ m, $\bar{a} = 0.1$ m, $\beta_p = 1.0$)

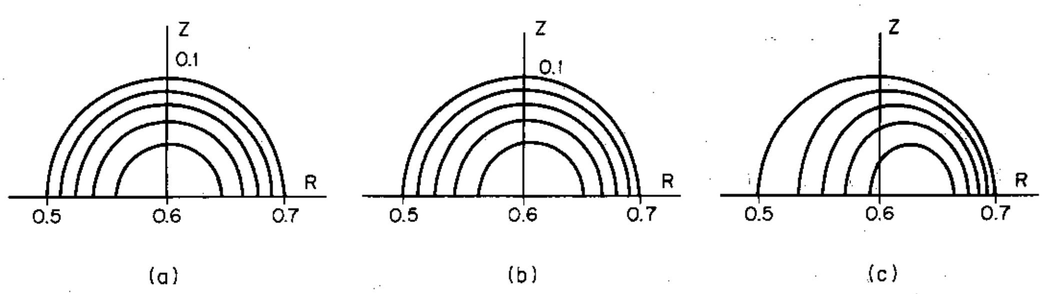


Fig. 3 Typical Shapes of Plasma Equilibrium expressed by ψ_K (II)
($R_0 = 0.6$ m, $\bar{a} = 0.1$ m, $\varepsilon_2 = 0.0$, $\varepsilon_3 = 0.0$)
(a) $\beta_p = 0.2$ (b) $\beta_p = 1.0$ (c) $\beta_p = 5.5$

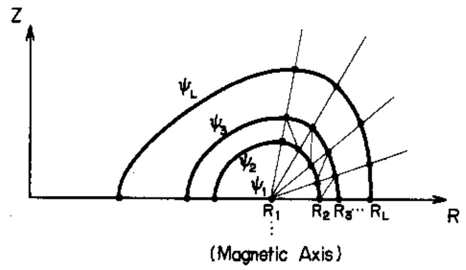


Fig. 4 Calculation of equi-ψ lines and nodes

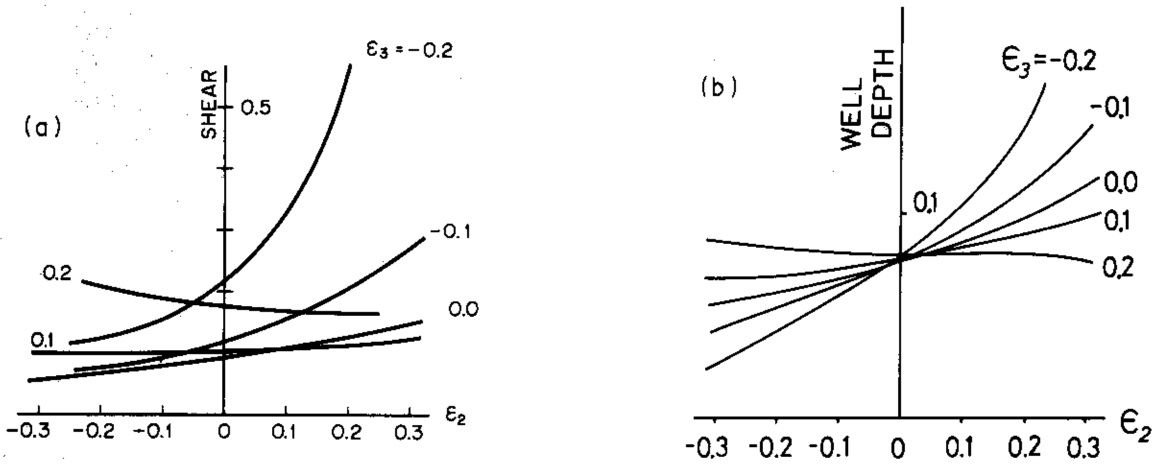


Fig. 5 (a) Shear and (b) Magnetic Well Depth
($R_0 = 0.6$ m, $\bar{a} = 0.1$ m, $\beta_p = 1.0$, $q = 1.0$, $B_{T0} = 1$ Wb/m^2)

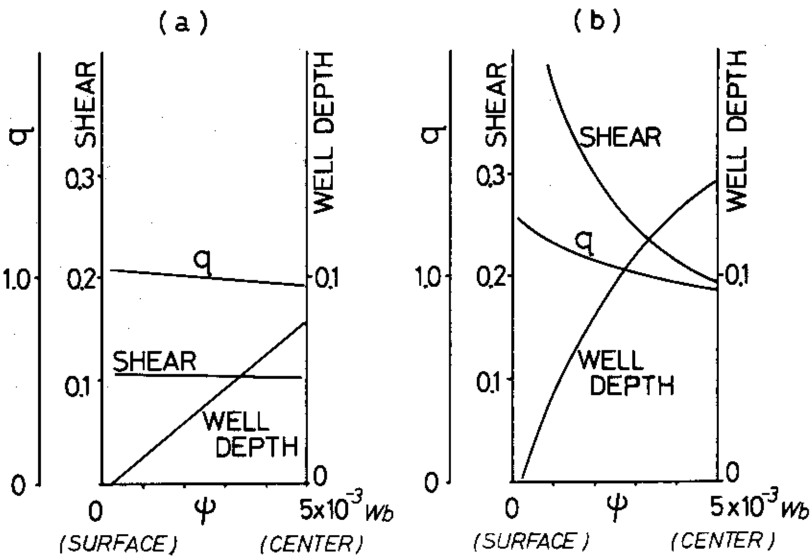


Fig. 6 Cross Sectional Profile of Shear, Well Depth, and q
($R_0 = 0.6$ m, $\bar{a} = 0.1$ m, $\beta_p = 1.0$, $q = 1.0$, $B_{T0} = 1$ Wb/m^2)
(a) $\varepsilon_2 = 0.0$, $\varepsilon_3 = 0.0$ (b) $\varepsilon_2 = 0.2$, $\varepsilon_3 = -0.2$

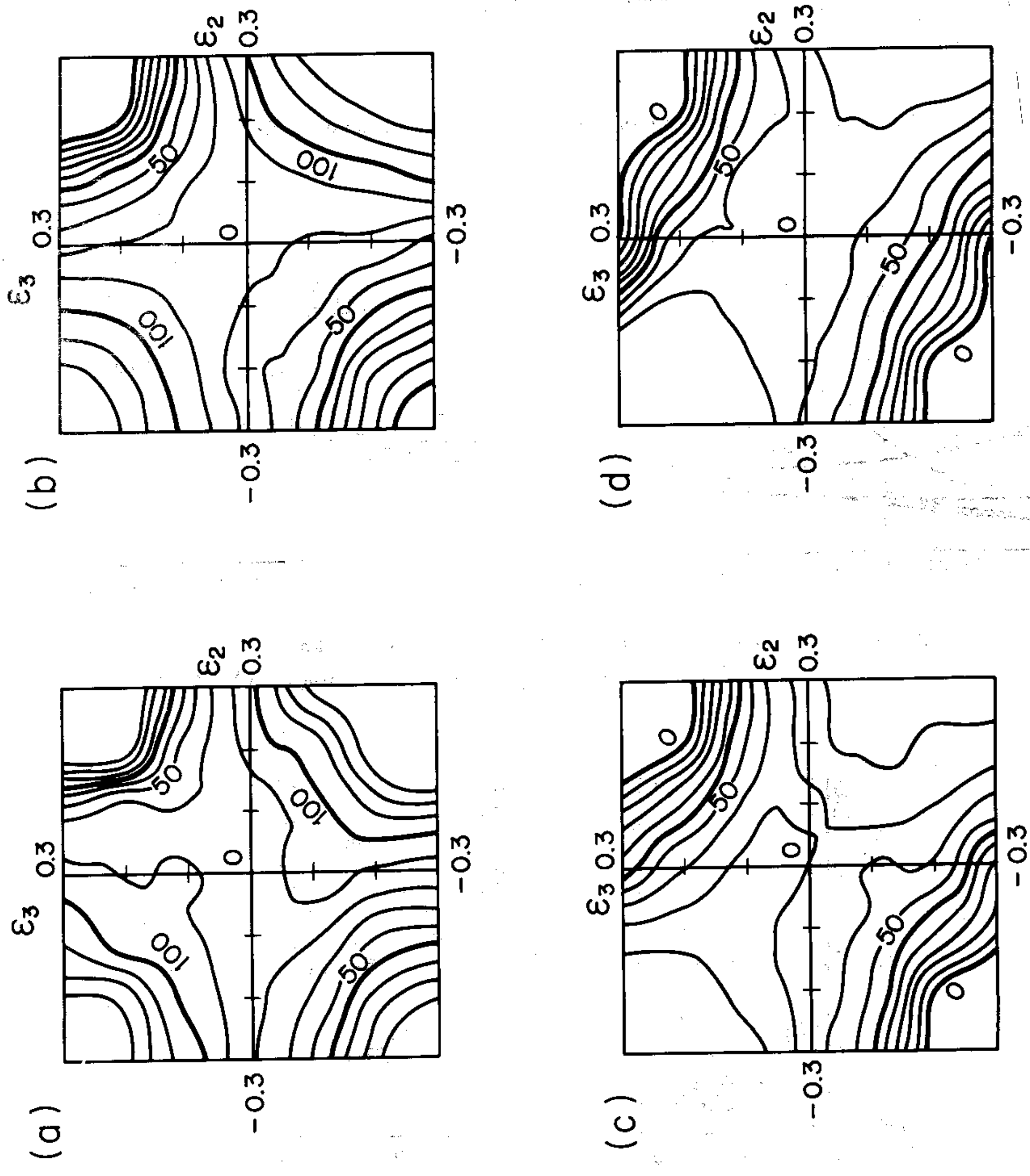


Fig. 7 Maximum Stable Currents (kA) (Localized Instability)
($R_0 = 0.6$ m, $\bar{a} = 0.1$ m, $B_{T_0} = 1$ Wb/m^2)
(a) $\beta_p = 0.2$ (b) $\beta_p = 0.5$ (c) $\beta_p = 1.0$ (d) $\beta_p = 1.5$

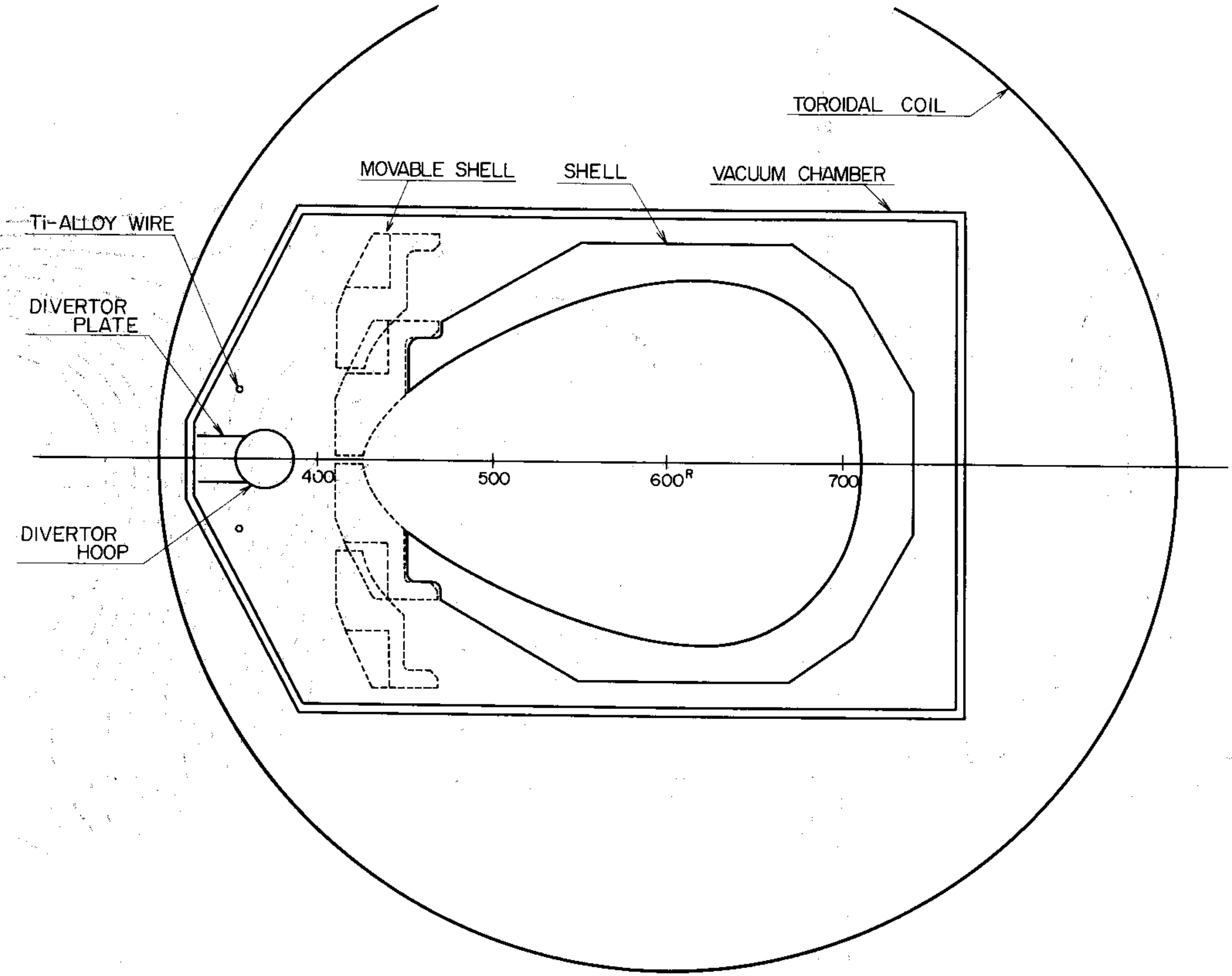


Fig. 8 Cross section of JFT−2a (DIVA)
($R_0 = 0.59$, $\overline{a} = 0.1143$, $\varepsilon_2 = 0.27$, $\varepsilon_3 = -0.17$; Eq. (9))

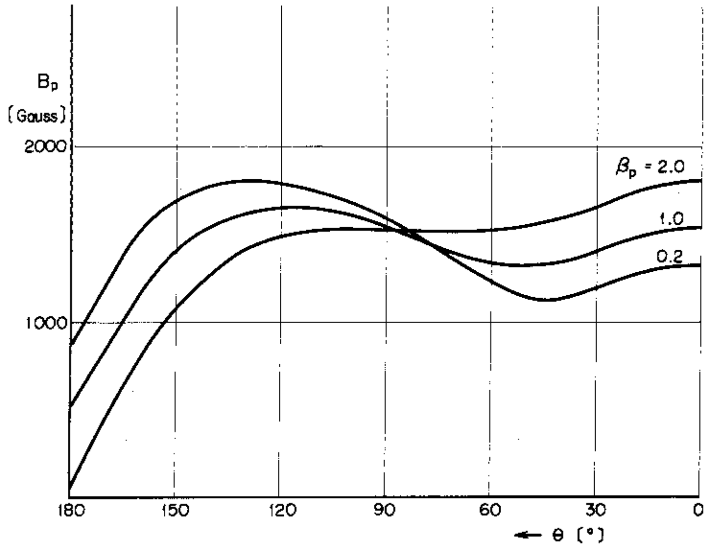


Fig. 9 Magnetic Field Distribution on the Shell Surface (JFT-2a) $I_p = 80\,kA\ (q \simeq 1.4)$

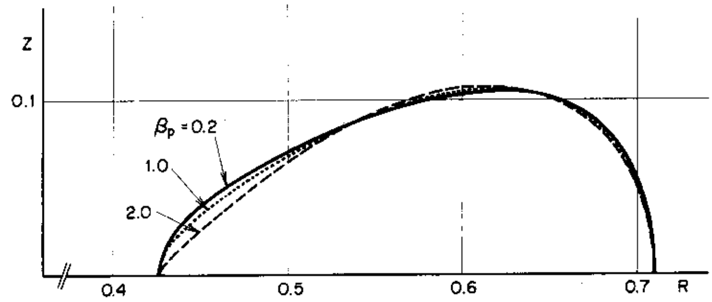


Fig. 10 Plasma Surface expressed by $\Psi_K = 0$

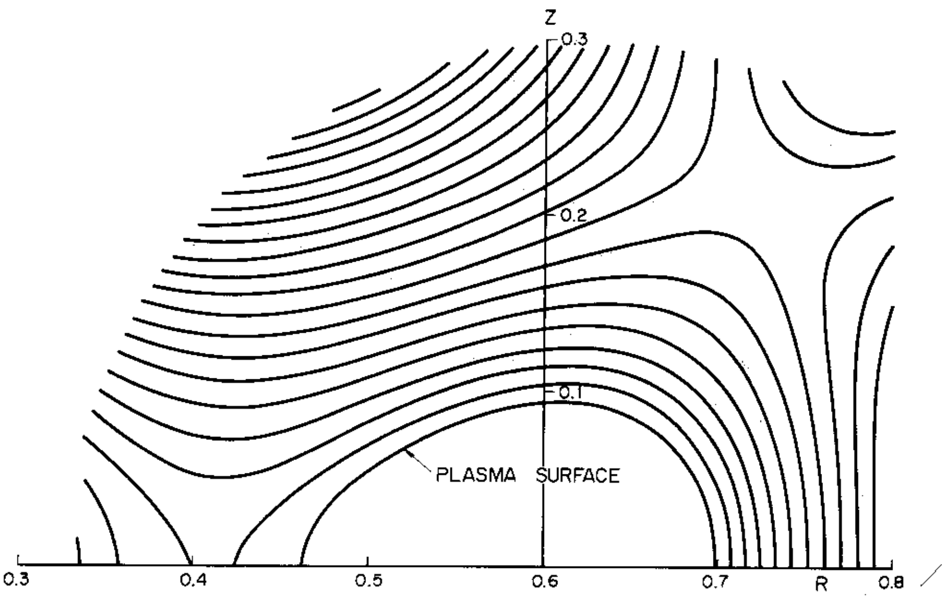


Fig. 11 Vacuum Magnetic Field outside the Plasma Column ($\beta_p = 1.0$)

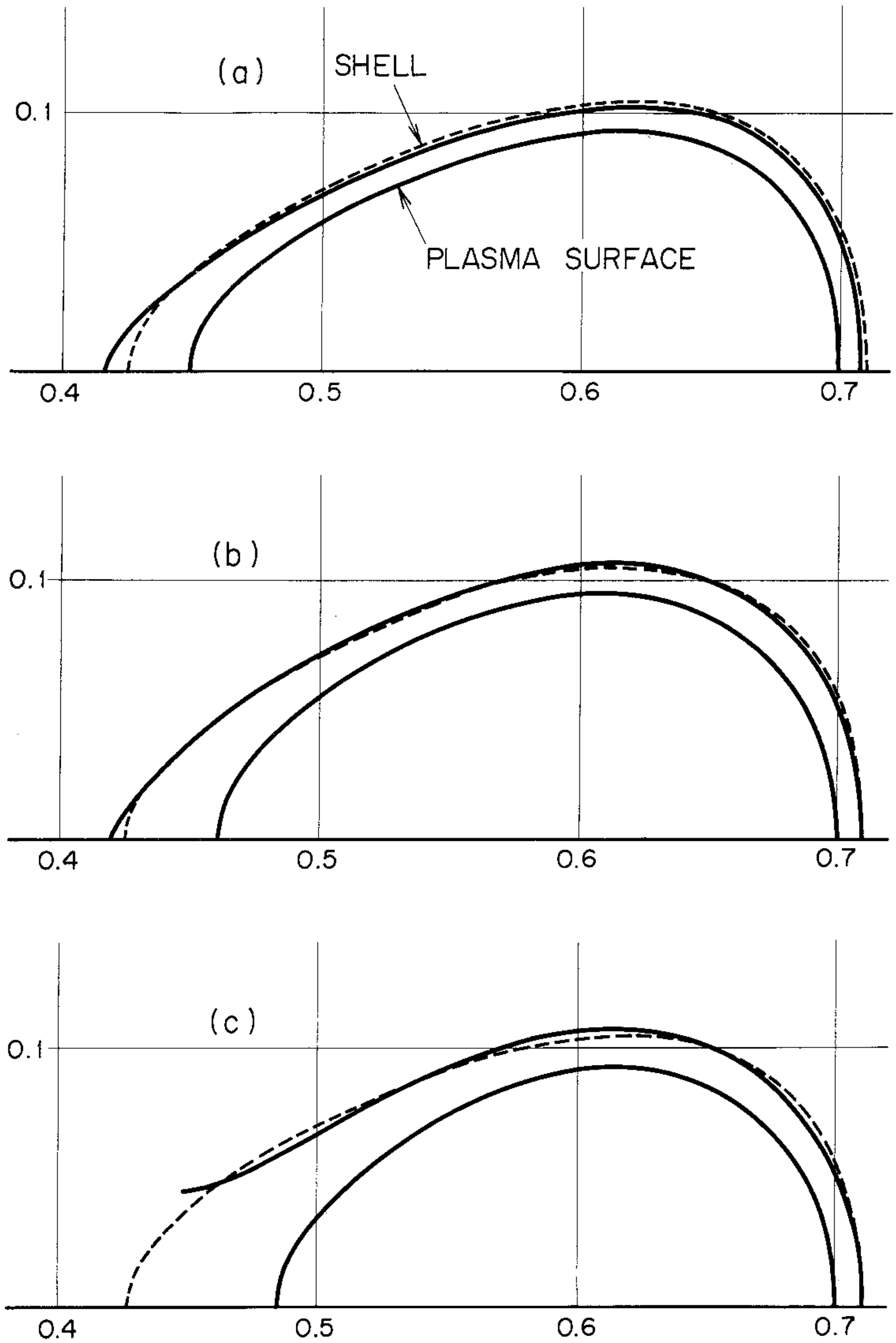


Fig. 12 Plasma Surface and Vacuum Magnetic Field Line (dotted line: shell surface)
(a) $\beta_p = 0.2$ (b) $\beta_p = 1.0$ (c) $\beta_p = 2.0$

付録 1 平衡量および不安定性の計算コード

```
CCCC      MAIN
      COMMON ARRAY(10000) <-------- (Node 数 (NPSi, NCHi) に応じて増減する
                                    (10000 の場合    49.1 k word)
      COMMON / COM10 / BTZERO
      COMMON / COM1 / RO,AA,ASP,ES2,ES3
      COMMON / REPEAT / LMN,COEFX
      COMMON / HAS / TTT,DCK
      COMMON / OPT / IZONE,IMESH,ICPY,IOPX
      COMMON / REGION / YMAX,YMIN,XMAX,XMIN
      COMMON / PSAT / EI(10)
      COMMON / RZ / RMAX,RMIN,ZMAX,RA
      COMMON / IO / IOIN,IOUT,IOI
      COMMON / LIMT / LIM1,LIM2,LIM3,EPS1,EPS2,EPS3
      COMMON / IPR / IPR1,IPR2,IPR3
      COMMON/BBBB / BETA,PAI,PO,P1,P2,TIO,VI1,VI2,BTO
      COMMON     / CONTR /  IPR(12)  ,LIM(6)    ,EPS(6)
      DIMENSION TITLE(9)
      DIMENSION R(5),Z(5)
      DIMENSION ZZ(5)
      EXTERNAL FUNCR,FUNCG
      PAI=3.14159265358979
      NCASE=0
C
      CALL INPT  ------------  DATAの読み込み
      CALL SSET  ------------  ARRAY(10000) の割り振り
C
    6 FORMAT(9A8)
  555 FORMAT(8E10.3)
  710 FORMAT(4I5)
  711 FORMAT(8F10.5)
 1002 FORMAT(6I12)
      READ(5,555) RO,AA,BTO
      ASP=RO/AA
      BTZERO=BTO
C --- TOTAL TOROIDAL CURRENT = CURRTT (KILO AMPERE)
      PO=0.
      P2=0.
      VI2=0.
      P1=1.0
      TIO=RO*BTO
      TIU=TIO**2
      READ(5,6)TITLE
      READ(5,1002) ICASE
      NCASE=0
 9999 CONTINUE
      NCASE=NCASE+1
      COEFX=1.0
      READ(5,711) ES2,ES3,AIP,BETAJ
      CURRTT=AIP*1.E3
      QS=AA*AA/RO*BTO/CURRTT*5.E6
      AK=2.*CURRTT/PAI/(AA   **2)
C     BETA = 2.0*MYU0* P-DASH
C     VI1 / BETA**** ACCORDING TO BAZHANOVA
C     CURRENT= MYU0/(2.*PAI) * I-MKS
      AK=AK*PAI*4.D-7
      BETA=AK*BETAJ/RO
      VI1=AK*0.5*(1.0-BETAJ)*(RO+ SQRT(RO**2-AA   **2))
      WRITE(6,560) NCASE,TITLE
      WRITE(6,604)
      WRITE(6,609) QS
      WRITE(6,610) CURRTT
      WRITE(6,611) BETAJ
      WRITE(6,561)RO,AA,BTO,BETA
      WRITE(6,605)
      WRITE(6,601) PO,P1,P2,TIO,VI1,VI2
      WRITE(6,824) ES2,ES3
  824 FORMAT(/10X,'ES2 = ',F10.5/10X,'ES3 = ',2F10.5/)
  560 FORMAT('1',///5X,'CASE NO. ',I3,5X,9A8//)
  604 FORMAT(///10X,'EQUATION'/15X,'L(PSI) + R**2 * EI(6) + EI(7) = 0.0'
     *//)
  609 FORMAT(10X,'INPUT Q',T30,1PD18.5)
  610 FORMAT(10X,'TOTAL TOROIDAL CURRENT',T30,1PD18.5)
  611 FORMAT(10X,'BETA-J (BAZHANOVA)',T30,F18.7/)
  600 FORMAT(10X,'NO. OF EQUI-PSI LINES',T30,I8)
  561 FORMAT(       10X,'RO',T30, 1PD18.5/10X,'AA',T30,D18.5/10X,'BTO',
     *   T30,D18.5/10X,'BETA',T30,D18.5//)
  605 FORMAT(5X,'PRESSURE=PO+P1*PSI+P2*PSI**2'/5X,'I**2=TIO+VI1*PSI+VI2*
     *PSI**2')
  601 FORMAT(10X,'PO',T30,1PD18.5/10X,'P1',T30,D18.5/10X,'P2',T30,
     *      D18.5/10X,'TIO',T30,D18.5/10X,'VI1',T30,D18.5/10X,'VI2',
     *      T30,D18.5//)
      ZRO=0.0
C     IF(LMN.EQ.0) GO TO 33
      DO 32 I=1,4  ------------  プラズマ表面 (ψ=0) の4点を求める
      TTT=(4-I)*PAI/3.0
      CALL HASAMI(ZRO,FUNCG,0.03,0.3,RRX,DAH,1.E-4)
      R(I)=RO+RRX*COS(TTT)
      Z(I)=RRX*SIN(TTT)
   32 CONTINUE
      R(5)=R(4)
      Z(5)=Z(4)
      CALL COSET(R,Z)  ------------  係数 (本文中(6)式 C1～C4) の決定
      WRITE(6,678)
      WRITE(6,679) (I,EI(I),I=1,7)
      WRITE(6,606)(I,R(I),Z(I),I=1,5)
  678 FORMAT(///10X,'PSI= EI(1)*R**2*(R**2-4.*Z**2) + EI(2)*(R**2*((R**2
     *-4.*Z**2)*DLOG(R)-3.*Z**2)+2.*Z**4/3.)'/
     *       T13,'EI(3)*R**2  +EI(4)*(R**2*DLOG(R)-Z**2)+EI(5)'/
     *        T13,'+EI(6)*0.5*R**2*Z**2 +EI(7)*0.5*Z**2'/)
  679 FORMAT(7(15X,'EI(',I2,')= ',1PD18.5/))
  602 FORMAT(///10X,'IZONE/IMESH/ICPY/IOPX',T30,4I6)
  603 FORMAT(10X,'YMAX/YMIN/XMAX/XMIN',T30,4F10.3)
C     MAGNETIC AXIS  ********************  ------------  磁気軸
      GOSA=1.E-4
      DDR=(R(5)-R(1))*0.2
      RR1=R(1)+DDR
      RR2=R(5)-DDR*0.5
      CALL HASAMI(ZRO,FUNCR,RR1,RR2,XT1,XT2,GOSA)
      WRITE(6,615) XT1,XT2
  615 FORMAT(/10X,'MAGNETIC AXIS',T30,2F15.7)
      RA=XT1
      IF( ABS(XT2-RO).LT. ABS(XT1-RO))RA=XT2
      IF(RA.EQ.0.0) GO TO 123
CCC*****************
C
      RMAX=R(5)
      CALL JOINT(TIO)  ------------  係数の受け渡し
   33 CONTINUE
      CALL FLUTE2(ARRAY)  ------------  平衡量, 局所不安定性の計算
C
      GO TO 129
  123   WRITE(6,607)
  607 FORMAT(/10X,'AXIS NOT EXIST'/)
  129 CONTINUE
  400 CONTINUE
```

```
      IF(NCASE.LT.ICASE) GO TO 9999
      STOP 77777
  905 FORMAT(/10X,'INPUT BOUNDARY'/5(15X,I9,3X,'R= ',F9.5,'  Z=',F9.5/))
      END
```

* SOURCE STATEMENT *

```
      SUBROUTINE COSET(X,Y)
      COMMON/BBBB / BETA,PAI,P0,P1,P2,TIO,VI1,VI2,BTO
      COMMON/PSAI / FI(10)
      DIMENSION X(1),Y(1),AA(5,6)
      FUNCA(R,Z)=R**2*(R**2-4.*Z**2)
      FUNCB(R,Z)=R**2*((R**2-4.*Z**2)*ALOG(R)-3.*Z**2)+2.*Z**4/3.
      FUNCC(R)=R**2
      FUNCD(R,Z)=R**2*ALOG(R)-Z**2
      FI(6)= 0.5*BETA
      FI(7)= 0.5*VI1
      FI(2)=0.0
C     DO 10 I=1,5
      DO 10 I=1,4
      J=I
      IF(I.EQ.4) J=5
C     AA(I,1)=FUNCA(X(I),Y(I))
      AA(I,1)=FUNCA(X(J),Y(J))
C     AA(I,2)=FUNCB(X(I),Y(I))
C     AA(I,3)=FUNCC(X(I))
      AA(I,2)=FUNCC(X(J))
C     AA(I,4)=FUNCD(X(I),Y(I))
      AA(I,3)=FUNCD(X(J),Y(J))
C     AA(I,5)=1.0
      AA(I,4)=1.0
C     AA(I,6)=(FI(6)*X(I)**2 + FI(7))*0.5*Y(I)**2
      AA(I,5)=(FI(6)*X(J)**2 + FI(7))*0.5*Y(J)**2
   10 CONTINUE
      EPS=1.0-5
C     CALL SWEEPD(AA,5,5,6,EPS,ILL)
      CALL SWEEPS(AA,5,4,5,EPS,ILL)
      IF(ILL.EQ.0) GO TO 100
      WRITE(6,101) ILL
  101 FORMAT(//10X,'ERROR IN SWEEPD ( COSET )',I10)
  100 CONTINUE
C     FI(1)=AA(1,6)
      FI(1)=AA(1,5)
C     FI(2)=AA(2,6)
C     FI(3)=AA(3,6)
      FI(3)=AA(2,5)
C     FI(4)=AA(4,6)
      FI(4)=AA(3,5)
C     FI(5)=AA(5,6)
      FI(5)=AA(4,5)
      RETURN
      END
```

← 係数（本文中 $C_1 \sim C_4$）の決定

* SOURCE STATEMENT *

```
      SUBROUTINE JOINT(TIO)
      COMMON / PSAI/ FI(10)
      COMMON / PSI / EI(10)
      COMMON / COM2C / P1,P2,B1,B2
      DO 10 I=1,7
   10 EI(I)=FI(I)
      EI(8)=TIO
      P1=EI(6)
      P2=0.0
      B1=2.0*EI(7)
      B2=0.0
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK CALPSI                                                              GGG
      SUBROUTINE CALPSI(CALR1,CALZ1,PHI1)                               GGG
      DOUBLE  CALR0,CALR1,CALR2,CALR4,CALZ1,CALZ2                       GGG
     1  ,CALZ4,PHI1,TEMP0, TEMP1                                        GGG
      COMMON / PSI / EI(10)                                             GGG
      CALR0=DLOG(CALR1)                                                 GGG
      CALR2=CALR1**2                                                    GGG
      CALR4=CALR2**2                                                    GGG
      CALZ2=CALZ1**2                                                    GGG
      CALZ4=CALZ2**2                                                    GGG
      TEMP0=EI(1)*CALR2*(CALR2-4.0*CALZ2)                               GGG
      IF(EI(2).EQ.0.0) GO TO 110                                        GGG
      TEMP0=TEMP0+EI(2)*(CALR2*((CALR2-4.0*CALZ2)*CALR0-3.0*CALZ2)      GGG
     *               +2./3.*CALZ4)                                      GGG
  110 TEMP1=EI(3)*CALR2+EI(4)*(CALR2*CALR0-CALZ2) + EI(5)               GGG
     *         - (EI(6)*CALR2+EI(7))*0.5*CALZ2                          GGG
      PHI1= TEMP0+TEMP1                                                 GGG
      RETURN                                                            GGG
      END                                                               GGG
```

← 関数 ψ(R, Z)

（他の平衡関数を計算するには CALPSI, FUNCP, FUNCQ, FUNCR, FUNCI, FNDI, FUNCT, FNDT を入れかえる。）

* SOURCE STATEMENT *

```
*DK FUNCP                                                               GGG
      DOUBLE PRECISION FUNCTION FUNCP(CALR1,CALZ1)                      GGG
      DOUBLE CALR0, CALR1, CALR2, CALZ1, CALZ2, TEMP                    GGG
      COMMON / PSI / EI(10)                                             GGG
      CALR0=DLOG(CALR1)                                                 GGG
      CALR2=CALR1**2                                                    GGG
      CALZ2=CALZ1**2                                                    GGG
      TEMP=4.0*EI(1)*(CALR2-2.0*CALZ2)+ 2.0*EI(3) - EI(6)*CALZ2         GGG
      TEMP=TEMP+ EI(4)*(2.0*CALR0+1.0) + EI(2)*(4.0*(CALR2-2.0*CALZ2)   GGG
     *          *CALR0+CALR2-10.0*CALZ2)                                GGG
      FUNCP=TEMP*CALR1                                                  GGG
      RETURN                                                            GGG
      END                                                               GGG
```

← $p = \frac{\partial \psi}{\partial R}$

* SOURCE STATEMENT *

```
*DK FUNCQ                                                               GGG
      DOUBLE PRECISION FUNCTION FUNCQ(CALR1,CALZ1)                      GGG
      DOUBLE CALR0, CALR1, CALZ1, CALR2, TEMP ,CALZ2                    GGG
      COMMON / PSI / EI(10)                                             GGG
      CALR2=CALR1**2                                                    GGG
      CALR0=DLOG(CALR1)                                                 GGG
      CALZ2=CALZ1**2                                                    GGG
      TEMP=EI(2)*(8./3.*CALZ2-2.0*CALR2*(4.0*CALR0-3.0))-8.*EI(1)*CALR2 GGG
     *          -2.0*EI(4) - EI(6)*CALR2-EI(7)                          GGG
      FUNCQ=CALZ1*TEMP                                                  GGG
      RETURN                                                            GGG
      END                                                               GGG
```

← $q = \frac{\partial \psi}{\partial Z}$

* SOURCE STATEMENT *

```
      FUNCTION FUNCR(R)                          <---- ψ (R, θ)
      COMMON / PSAI / EI(10)
      FUNCR=(4.*EI(1)+EI(2))*R**2+2.*EI(3)+EI(4)
     *        +(4.*EI(2)*R**2+2.*EI(4))*ALOG(R)
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK FUNCI
      DOUBLE PRECISION FUNCTION FUNCI(PSI0,PSI1)  <---- I(ψ)
      DOUBLE PSI0, PSI1, AI0 ,AI1
      COMMON / PSI / EI(10)
      COMMON / REPEAT / L,COEFX     <---- プラズマ電流の初期値との比（不安定性計算の際に使用）
      AI0=EI(8)
      AI1=EI(7)*2.0*COEFX
C     AI2=0.0
      FUNCI=DSQRT(AI0+AI1*PSI1*COEFX)
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK FNDI
      DOUBLE PRECISION FUNCTION FNDI(PSI0,PSI1)   <---- ∂I/∂ψ
      DOUBLE PSI0, PSI1, FUNCI ,AI1
      COMMON / PSI / EI(10)
      COMMON / REPEAT / L,COEFX
      AI1=EI(7)*2.0*COEFX
C     AI2=0.0
      FNDI=0.5*AI1/FUNCI(PSI0,PSI1)
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK FUNCT
      DOUBLE PRECISION FUNCTION FUNCT(PSI0,PSI1)  <---- P(ψ)
      DOUBLE  PSI0,PSI1,T0, T1
      COMMON / PSI / EI(10)
      COMMON / REPEAT / L,COEFX
      T1=EI(6) *COEFX
C     T2=0.0
      T0= -T1*PSI0
      FUNCT=T0+T1*PSI1
      FUNCT=FUNCT*COEFX
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK FNDT
      DOUBLE PRECISION FUNCTION FNDT(PSI0,PSI1)   <---- ∂P/∂ψ
      DOUBLE PSI0,PSI1,T1
      COMMON / PSI / EI(10)
      COMMON / REPEAT / L,COEFX
C     T2=0.0
      T1=EI(6)*COEFX
      FNDT=T1
      RETURN
      END
```

* SOURCE STATEMENT *

```
      FUNCTION FUNCG(R)                          <---- プラズマ境界の形 ( (1)式 )
      COMMON / COM1 / R0,AA,ASP,ES2,ES3
      COMMON / HAS / T,DAM
      FUNCG=R*R*((1.0-ES2*COS(2.*T))-R/AA*ES3*COS(3.*T))-AA*AA
      RETURN
      END
```

* SOURCE STATEMENT *

```
      SUBROUTINE HASAMI(A,FUNCX,XMIN,XMAX,RT1,RT2,EPS) <---- はさみうち法
      EPS=0.1*EPS                                           A = FUNCX(RT1)
      IND=0                                                 を解く
      RMIN=XMIN
      RMAX=XMAX
  170 IND=IND+1
  105 DELR=(RMAX-RMIN)/(4**IND)
      IF(DELR.LT.EPS) GO TO 300
      R2=RMIN
  110 R1=R2
      R2=R1+DELR
      IF(R2.GT.RMAX) GO TO 100
      TES1=FUNCX(R1)-A
      TES2=FUNCX(R2)-A
      D=TES1*TES2
      IF(D.LE.0.) GO TO 200
      GO TO 110
  200 RMIN=R1
      RMAX=R2
      IND=1
      GO TO 105
  300 RT1=R1
      GO TO 150
  100 IF(IND.LT.6) GO TO 170
      RT1=0.0
  150 RT2=0.0
      RETURN
      END
```

* SOURCE STATEMENT *

```
      BLOCK DATA
      COMMON / IO / IOIN,IOUT,IO1
      COMMON / CONST / PI,C2,C3,C4,C8,C16,C64,Y1,Y2,Y3,Y4,Y6,Y12
      DATA        IOIN/5/  ,IOUT/6/
     1                               ,PI/3.14159265358979/
     2,C2/0.5/  ,C3/0.333333333333333/    ,C4/0.25/ ,C8/0.125/
     3,C16/0.0625/         ,C64/0.015625/       ,Y1/1.0/  ,Y2/2.0/
     4,Y3/3.0/  ,Y4/4.0/  ,Y6/6.0/  ,Y12/12.0/
      END
```

* SOURCE STATEMENT *

```
*DK FLUTE
      SUBROUTINE FLUTE2(A)    ←――― 平衡量、不安定性の計算
      DIMENSION A(1)
      COMMON    / SEISUU / NPSI    ,NCHI     ,NDIE     ,NPSIM    ,NINT
     1,NPSIN    ,NTRI    ,NTRIM
      COMMON    / IO /    IOIN     ,IOUT     ,IO1
      COMMON    /LOCAT /  K01      ,K02      ,K03      ,K04      ,K05
     1,K06      ,K07      ,K08      ,K09      ,K10      ,K11      ,K12
     2,K13      ,K14      ,K15      ,K16      ,K17      ,K18      ,K19
     3,K20      ,K21      ,K22      ,K23      ,K24      ,K25      ,K26
     4,K27      ,K28      ,K29      ,K30      ,K31      ,K32      ,K33
      COMMON    / PLAS /   TITLE(18),IPAGE
      COMMON / PSI / EI(10)                                             GGG
      COMMON    / RZ /     RMAX     ,RMIN     ,ZMAX     ,RCEN
      COMMON    / CONTR /  IPR(12)  ,LIM(6)   ,EPS(6)
      DATA       IOIN/ 5 /,IOUT/ 6 /,IO1/ 1 / ,IPAGE/ 0 /
      DATA       NZONE/0/ ,MESH/0/
C 100 CONTINUE
      CALL SECOND(TIMEF)
C     CALL INPT    ---- CONTAINED IN MAIN ----
C     CALL SSET    ---- CONTAINED IN MAIN ----
      CALL STEP1(A(K01),A(K02),A(K03),NPSI,NCHI,EPS(1),IPR(1),LIM(1)   GGG
     *      ,IPR(8))                                                    GGG
      CALL STEP2(A(K01),A(K02),A(K03),A(K04),A(K05),A(K07),NCHI,NPSI
     1,NINT,EPS(2),IPR(2),LIM(1))
      CALL STEP3(A(K01),A(K02),A(K03),A(K04),A(K05),IPR(2),NCHI,NPSI)
      CALL STEP4(A(K01),A(K02),A(K03),A(K04),A(K05),A(K06),A(K05),A(K07)
     1,A(K08),NCHI,NPSI,NINT,EPS(2),EPS(3),LIM(1),LIM(3),IPR(2))
      CALL STEP5(A(K01),A(K02),A(K03),A(K04),A(K05),NCHI,NPSI,IPR(2)
     1,IPR(3))
      CALL STEP6(A(K02),A(K03),A(K04),A(K09),A(K10),NCHI,NPSI,NTRI
     1,IPR(5))
      CALL STEP7(A(K01),A(K03),A(K04),A(K09),A(K10),A(K11),A(K12),A(K13)
     1,A(K14),A(K15),A(K16),NCHI,NPSI,NTRI,IPR(5))
      CALL STEP8(A(K01),A(K04),A(K09),A(K10),A(K11),A(K12),A(K13),A(K14)
     1,A(K15),NPSI,NTRI,IPR(5))
      CALL STEP9(A(K01),A(K03),A(K17),A(K18),A(K19),NPSI,IPR(5))
      CALL STEP10(A(K03),A(K17),A(K18),A(K19),A(K20),A(K21),NPSI,IPR(5))
      CALL STEP11(A(K01),A(K02),A(K03),A(K17),A(K18),A(K19),A(K20)
     1,A(K21),A(K22),NPSI,NCHI,IPR(5))
      CALL STEP12(A(K01),A(K02),A(K03),A(K17),A(K18),A(K19),A(K20)
     1,A(K21),A(K22),A(K23),A(K24),A(K25),A(K26),A(K27),NCHI,NPSI
     2,IPR(6))
      IF(IPR(4).EQ.0) GO TO 110
      CALL STEP13(A(K03),A(K17),NPSI,IPR(4))
      CALL STEP15
  110 CONTINUE
      IF(IPR(7) .GT. 0) CALL STEP14(A(K01),A(K28),A(K29),A(K29),A(K30)
     1,A(K31),A(K32),A(K33),NCHI,NPSI,IPR(7))
      WRITE(IOUT,1020)
  100 CONTINUE
      RETURN                                                            GGGGGGGG
 1000 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3)
 1010 FORMAT(33H0 TOTAL CENTRAL PROCESSOR TIME IS,F10.3,4HSEC.)
 1020 FORMAT(26H0 THIS CASE ENDED NORMALLY/1X,135(1H#)/1X,135(1H#))
```

* SOURCE STATEMENT (FLUTE2)*

```
 1030 FORMAT(33H0 TOTAL CENTRAL PROCESSOR TIME IS,I3,4HMIN.,F10.3,4HSEC.
     1)
      END
```

* SOURCE STATEMENT *

```
*DK INPT
      SUBROUTINE INPT   ←――― 入力データの読み込み
      COMMON    / SEISUU /NPSI     ,NCHI     ,NDIE     ,NPSIM    ,NINT
     1,NPSIN    ,NTRI    ,NTRIM
      COMMON    / PLAS /   TITLE(18),IPAGE
      COMMON    / IO /     IOIN     ,IOUT     ,IO1
      COMMON / PSI / EI(10)                                             GGG
      COMMON    / PLOT /   NZONE    ,MESH     ,MAXR     ,MAXZ     ,INTER
     1,RMAXP    ,RMINP    ,ZMAXP    ,ZMINP
      COMMON    / CONTR / IPR(12)   ,LIM(6)   ,EPS(6)
      COMMON    / RZ /     RMAX     ,RMIN     ,ZMAX     ,RCEN
      DATA       BLANK/1H /
      READ(IOIN,1000) (TITLE(I),I=1,18)
      DO 110 I=1,18
      IF(TITLE(I) .NE. BLANK) GO TO 120
  110 CONTINUE
      CALL EXIT
  120 CONTINUE
  130 READ(IOIN,1010) ISEC
      IF(ISEC .EQ. 999) GO TO 190
      GO TO (140,150,160,170,180),ISEC
  140 READ(IOIN,1020) NPSI,NCHI,NDIE
      READ(IOIN,1020) (IPR(I),I=1,12)
      READ(IOIN,1030) RMAX,RMIN,ZMAX,RCEN
      READ(IOIN,1020) (LIM(I),I=1,6)
      READ(IOIN,1030) (EPS(I),I=1,6)
      GO TO 130
  150 READ(5,1030)(EI(I),I=1,7),TIO,RCEN                                GGG
      EI(8)=TIO                                                         GGG
      GO TO 130
  160 CONTINUE
      READ(IOIN,1020) NZONE,MESH,MAXR,MAXZ,INTER
      READ(IOIN,1030) RMAXP,RMINP,ZMAXP,ZMINP
      GO TO 130
  170 CONTINUE
  180 CONTINUE
      GO TO 130
  190 CONTINUE
      IPAGE = IPAGE+1
      WRITE(IOUT,1040) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1050)
      WRITE(IOUT,1060)
      WRITE(IOUT,1070) NPSI,NCHI,NDIE
      WRITE(IOUT,1080) (IPR(I),I=1,8)
      WRITE(IOUT,1090) (LIM(I),I=1,3)
      WRITE(IOUT,1100) RMAX,RMIN,ZMAX,RCEN
      WRITE(IOUT,1110) (EPS(I),I=1,3)
      WRITE(IOUT,1120)
      WRITE(IOUT,1130)
      IF(IPR(7) .EQ. 0) GO TO 200
      IPAGE = IPAGE+1
      WRITE(IOUT,1040) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1050)
      WRITE(IOUT,1150)
      WRITE(IOUT,1160) NZONE,MESH,MAXR,MAXZ,INTER
      WRITE(IOUT,1170) RMAXP,RMINP,ZMAXP,ZMINP
  200 CONTINUE
C     DELR = (RMAX-RCEN)/(NPSI-1)
C     RMAX = RMAX+DELR
      NPSIM=NPSI+1                                                      GGG
      NPSI = NPSI+1
```

```
57          NINT = LIM(2)
58          CALL RANDOM(IO1,DAM,1,1,0)
59          RETURN
60     1000 FORMAT(18A4)
61     1010 FORMAT(I3)
62     1020 FORMAT(12I6)
63     1030 FORMAT(6E12.5)                                                      GGG
64     1040 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3)
65     1050 FORMAT(11H0INPUT LIST)
66     1060 FORMAT(13H0 SECTION 001)
67     1070 FORMAT(
          1 30X,50HNPSI     NUMBER OF EQUI-PSI LINES                   ,I16
          2/30X,50HNCHI     NUMBER OF NODES ON EACH LINES              ,I16
          3/30X,50HNDIE     DELTA PSI <---- ゴ,μ'用 [illegible]          ,I16   )
68     1080 FORMAT(
          1 30X,50HIPR1     PRINT OPTION(PSI AND EQUI-PSI LINES)       ,I16
          2/30X,50HIPR2     PRINT OPTION(CHI,B-CHI,JACOBIAN,OTHERS)    ,I16
          3/30X,50HIPR3     PRINT OPTION(POLOIDAL CURRENT,B-TOLOIDAL)  ,I16
          4/30X,50HIPR4     PRINT OPTION(AREA,AEQ,ASP,JJ,JD)           ,I16
          5/30X,50HIPR5     PRINT OPTION(T,P,V2-DASH,S,D,W,SS,DD)      ,I16
          6/30X,50HIPR6     PRINT OPTION(STABILITY CONST.)             ,I16
          7/30X,50HI7R7     PRINT OPTION(PLOTTER)                      ,I16       GGG
          8/30X,50HIPR8     EQUI-PSI BY (1/0  =  PSI / R  )            ,I16  )    GGG
69     1090 FORMAT(
          1 30X,50HLIM1     ITERATION COUNT LIMIT FOR CALC. EQUI-PSI   ,I16
          2/30X,50HLIM2     ITERATION COUNT LIMIT FOR INTEG.           ,I16
          3/30X,50HLIM3     ITERATION COUNT LIMIT FOR DIF.             ,I16  )
70     1100 FORMAT(
          1 30X,50HRMAX     PLASMA OUTER RADIUS                        ,F16.5
          2/30X,50HRMIN     RR MINIMUM                                 ,F16.5
          3/30X,50HZMAX     ZZ MAXIMUM                                 ,F16.5
          4/30X,50HRCEN     PLASMA CENTER                              ,F16.5
          5)
71     1110 FORMAT(
          1 30X,50HEPS1     ITERATION CONVERGENCE CRITERION FOR EQ-PSI,1PE16.8
          2/30X,50HEPS2     ITERATION CONVERGENCE CRITERION FOR INTEG.,1PE16.8
          3/30X,50HEPS3     ITERATION CONVERGENCE CRITERION FOR DIF.  ,1PE16.8
          4)
72     1120 FORMAT(13H0 SECTION 002)
73     1130 FORMAT(
          * 30X,'PSI = A1 * R**2*(R**2-4.*Z**2)'                                GGG
          */30X,'     + A2 * (R**2*((R**2-4.*Z**2)*DLOG(R)-2./3.*Z**4)'         GGG
          */30X,'     + A3 * R**2'                                              GGG
          */30X,'     + A4 * (R**2*DLOG(R)-Z**2)'                               GGG
          */30X,'     + A5 '                                                    GGG
          */30X,'     - A6 * 0.5*R**2*Z**2'                                     GGG
          */30X,'     - A7 * 0.5*Z**2'/                                         GGG
          7/30X,52HI = SQRT(I0+I1*PSI+I2*PSI**2)
          */30X,52H                 I0 = TI0 = A8 = EI(8)                       GGG
          */30X,52H                 I1 = 2.0*EI(7) = 2.0*A8                     GGG
          1/30X,52H                 I2 = 0
          1/30X,52H        P        : PRESSURE                                  GGG
          5/30X,52HP = P0+P1*PS1+P2*PSI**2
          6/30X,52H                 P2 = 0
          7/30X,52H                 P1 = EI(6)                                  GGG
          8/30X,52H                 P0 = -P1*PSI(RP,0)                          GGG
          9/30X,  '  WHERE PSI(RP,0) : PSI AT PLASMA OUTER RADIUS'/)            GGG
74     1140 FORMAT(
          1 30X,21H         CONSTANT A1 =,1PE16.8
          2/30X,21H         CONSTANT A2 =,1PE16.8
          3/30X,21H         CONSTANT A3 =,1PE16.8
          4/30X,21H         CONSTANT A4 =,1PE16.8
          5/30X,21H         CONSTANT A5 =,1PE16.8
          6/30X,21H         CONSTANT A6 =,1PE16.8                               GGG
          7/30X,21H         CONSTANT A7 =,1PE16.8                               GGG
          8/30X,21H         CONSTANT A8 =,1PE16.8)                              GGG
75     1150 FORMAT(13H0 SECTION 003)
76     1160 FORMAT(
          1 30X,50HNZONE  NUMBER OF ZONES                               ,I16
          2/30X,50HMESH   NUMBER OF MESHES FOR EACH ZONE                ,I16
          3/30X,50HMAXR   OPTION TO R-MAX AND R-MIN                     ,I16
          4/30X,50HMAXZ   OPTION TO Z-MAX AND Z-MIN                     ,I16
          5/30X,50HINTER  OPTION TO INTERPOLATION(0=YES/1=NO)           ,I16)
77     1170 FORMAT(
          1 30X,50HRMAXP  R-MAXIMUM                                     ,F16.5
          2/30X,50HRMINP  R-MINIMUM                                     ,F16.5
          3/30X,50HZMAXP  Z-MAXIMUM                                     ,F16.5
          4/30X,50HZMINP  Z-MINIMUM                                     ,F16.5)
78          END
```

* SOURCE STATEMENT *

```
      *DK SSET
 1          SUBROUTINE SSET  <------  コア・メモリーの割りふり
 2          COMMON    /LOCAT /  K01     ,K02     ,K03     ,K04     ,K05
           1,K06      ,K07      ,K08     ,K09     ,K10     ,K11     ,K12
           2,K13      ,K14      ,K15     ,K16     ,K17     ,K18     ,K19
           3,K20      ,K21      ,K22     ,K23     ,K24     ,K25     ,K26
           4,K27      ,K28      ,K29     ,K30     ,K31     ,K32     ,K33
 3          COMMON    / CONTR /  IPR(12) ,LIM(6)  ,EPS(6)
 4          COMMON    / IO /     IOIN    ,IOUT    ,IO1
 5          COMMON    / SEISUU / NPSI    ,NCHI    ,NDIE    ,NPSIM   ,NINT
           1,NPSIN    ,NTRI     ,NTRIM
 6          COMMON    / PLOT /   NZONE   ,MESH    ,MAXR    ,MAXZ    ,INTER
           1,RMAXP    ,RMINP    ,ZMAXP   ,ZMINP
 7          DATA       IDBL/ 2 /
 8          NPC = NPSI*NCHI
 9          K01 = 1
10          K02 = K01+NPSI*IDBL
11          K03 = K02+NPC*IDBL
12          K04 = K03+NPC*IDBL
13          K05 = K04+NPC*IDBL
14          K06 = K05+NPC*IDBL
15          K07 = K05+MAX0(NINT*4,NPC*2)*IDBL
16          K08 = K07+NPC*IDBL
17          LAST = K08+NPC*IDBL
18          NPT = (NCHI-1)*NPSI*2
19          NTRI = (NCHI-1)*2
20          K09 = K04+NPT*IDBL
21          K10 = K09+NPT*IDBL
22          K11 = K10+NPT*IDBL
23          K12 = K11+NPSI*IDBL
24          K13 = K12+NPSI*IDBL
25          K14 = K13+NPSI*IDBL
26          K15 = K14+NPSI*IDBL
27          K16 = K15+NPSI*IDBL
28          LASU = K16+NPSI*IDBL
29          K17 = K03+NPSI*IDBL
30          K18 = K17+NPSI*IDBL
31          K19 = K18+NPSI*IDBL
32          K20 = K19+NPSI*IDBL
33          K21 = K20+NPSI*IDBL
34          K22 = K21+NPSI*IDBL
35          K23 = K22+NPSI*IDBL
36          K24 = K23+NPSI*IDBL
37          K25 = K24+NPSI*IDBL
38          K26 = K25+NPSI*IDBL
39          K27 = K26+NPSI*IDBL
40          LASV = K27+NPSI*IDBL
```

```
      LAST = MAX0(LAST,LASU,LASV)
      K28 = K01+NPC
      K29 = K28+NPC
      K30 = K29+NPC
      K31 = K30+NPC
      K32 = K31+NPSI
      K33 = K32+NPSI
      LASW = K33+101*NZONE*(MESH+1)
      IF(IPR(7) .GT. 0) LAST = MAX0(LAST,LASW)
      WRITE(IOUT,1000) LAST
 1000 FORMAT(23H0 THIS PROBLEM REQUIRED,I6,5HWORDS)
      RETURN
      END
```

* SOURCE STATEMENT *

等ψ曲線の計算

```
*DK STEP1
      SUBROUTINE STEP1(PSI,CALR,CALZ,KP,KC,EPS1,IPR1,LIM1,NOPT)              GGG
      DOUBLE      PSI       ,CALR      ,CALZ      ,PSI0      ,PSI1
     1,CALR1     ,CALR3     ,CALZ3   ,RZERO                                  GGG
      DOUBLE      THET      ,DELR      ,DTHET
      DOUBLE   DPSI
      DIMENSION  PSI(KP)    ,CALR(KC,KP)          ,CALZ(KC,KP)
      COMMON     / IO /      IOIN       ,IOUT     ,IO1
      COMMON     / PLAS /    TITLE(18),IPAGE
      COMMON     / RZ /     RRMAX      ,RMIN      ,ZMAX      ,RCEN           GGG
      COMMON/SEISUU/ NPSI, NCHI, NDIE, NPSIM, NINT
     *, NPSIN, NTRI, NTRIM
      DATA        PI/3.1415926535898/
      DATA        IDBL/ 2 /
      DATA        IPSI/4HPSI(/          ,PAL/1H)/ ,RR/2HRR/ ,ZZ/2HZZ/
      RZERO = RCEN
      EPS0 = 1.0F-5
      PAGE1 = 4HPAGE
 2000 FORMAT(//10X,'IFIX',I6/)
      CALL SECOND(TIMEF)
C * * CAL. OF PSI.
      CALL CLEAR(0.D0,CALR,NPSI*NCHI)
      CALL CLEAR(0.D0,CALZ,NPSI*NCHI)
      DELR=RRMAX-RCEN                                                        GGG
      DFLR=DELR/(NPSI-2)                                                     GGG
      CALR(1,1) = RZERO
      CALR(1,NPSI-1)=RRMAX                                                   GGG
      IF(NOPT.EQ.0) GO TO 122
      CALL CALPSI(CALR(1,1),CALZ(1,1),PSI(1))
      CALL CALPSI(CALR(1,NPSI-1),CALZ(1,1),PSI(NPSI-1))
      DPSI=(PSI(NPSI-1)-PSI(1))/(NPSI-2)
      DO 125 N=2,NPSI
  125 PSI(N)=PSI(N-1)+DPSI
      GO TO 123
  122 CONTINUE
      DAR=(RRMAX-RZERO)/(NPSI-2)
      DO 124 N=2,NPSI
  124 CALR(1,N)=CALR(1,N-1)+DAR
      DO 126 N=1,NPSI
  126 CALL CALPSI(CALR(1,N),CALZ(1,1) ,PSI(N))
  123 CONTINUE
      RMAX=RRMAX*1.2                                                         GGG
C * * CAL. OF EQUI PSI POINTS.
      CALL CLEAR(RZERO,CALR,NCHI)                                            GGG
      DTHET = PI/FLOAT(NCHI-1)
      DELR = DELR*0.25
      DO 200 N=2,NPSI
      PSI0 = PSI(N)
      PSI1 = PSI(N-1)
      DO 190 I=1,NCHI
      INDEX = 1
      THET = DTHET*FLOAT(I-1)
      IF(DABS(THET-PI*0.5) .LE. EPS0) INDEX = 2
      GO TO (130,140),INDEX
  130 CALR1 = CALR(I,N-1)
      GO TO 150
  140 CALR1 = CALZ(I,N-1)
  150 IF(CALR(I,N-1) .EQ. 0.0) GO TO 200
      CALL DIST(CALR1,THET,CALR3,CALZ3,PSI1,PSI0,DELR,LIM1,EPS1,IPR1,1)
      GO TO (160,170),INDEX
  160 CALR(I,N) = CALR3
      CALZ(I,N) = CALZ3
      GO TO 180
  170 CALR(I,N) = CALZ3
      CALZ(I,N) = CALR3
  180 IF(CALR(I,N).LE.RMIN .OR. CALR(I,N).GT. RMAX) GO TO 200
      IF(CALZ(I,N) .GE. ZMAX) GO TO 200
  190 CONTINUE
  200 CONTINUE
      IF(NASK(IPR1,1) .NE. -1) GO TO 250
      NN = 0
  210 NNN = NCHI+3
      IPAGE = IPAGE+1
C     CALL INTBCD(IPAGE,PAGE2,L)
C     WRITE(IOUT,1000) PAGE1,PAGE2
      WRITE(IOUT,1000) PAGE1,IPAGE                                           GGG
      WRITE(IOUT,1010) TITLE
      WRITE(IOUT,1020)
  220 N = NN+1
      NN = N+4
      IF(NN .GT. NPSI) NN = NPSI
      WRITE(IOUT,1030) (IPSI,L,PAL,PSI(L),L=N,NN)
      WRITE(IOUT,1060) (RR,ZZ,L=N,NN)
      DO 230 I=1,NCHI
  230 WRITE(IOUT,1040) I,(CALR(I,L),CALZ(I,L),L=N,NN)
      IF(NN .GE. NPSI) GO TO 240
      NNN = NNN+NCHI+3
      IF(NNN .GT. 60) GO TO 210
      GO TO 220
  240 CONTINUE
      IF(NASK(IPR1,2) .NE. -1) GO TO 250
      CALL SECOND(TIMEL)
      TIME0 = TIMEL-TIMEF
      WRITE(IOUT,1050) TIME0
  250 CONTINUE
      CALL RANDM(IO1,CALR,NPSI*NCHI*IDBL,1,1)
      CALL RANDM(IO1,CALZ,NPSI*NCHI*IDBL,2,1)
      RETURN
C1000 FORMAT(1H1,126X,A4,1X,A4)
 1000 FORMAT(1H1,126X,A4,1X,I4)                                              GGG
 1010 FORMAT(20X,18A4)
 1020 FORMAT(/14H EQUI-PSI LINE)
 1030 FORMAT(//5X,5(3X,A4,I2,A1,1PD13.5,2X))
 1040 FORMAT(1X,I5,5(1X,1P2D11.4,2X))                                        GGG
 1050 FORMAT(1X,' STEP1 CENTRAL PROCESSOR USED TIME',F10.3,4HSEC.)
 1060 FORMAT(5X,5(7X,A2,9X,A2,5X))
      END
```

* SOURCE STATEMENT *

```
*DK STEP2      χ (真空磁場の場合 ψ と直交系をなす座標.)の 計算準備
      SUBROUTINE STEP2(PSI,CALR,CALZ,F,DINT,ICAL,KCHI,KPSI,KINT,EPS2
     1,IPR2,LIM1)
      DOUBLE     PSI0      ,PSI1      ,TEMP1
      COMMON    / IO /     IOIN      ,IOUT      ,IO1
      COMMON/SEISUU/ NPSI, NCHI, NDIE, NPSIM, NINT
     *, NPSIN, NTRI, NTRIM
      COMMON     / PLAS / TITLE(18),IPAGE
      DOUBLE     PSI       ,CALR      ,CALZ      ,F         ,DINT
      DIMENSION  PSI(KPSI),CALR(KCHI,KPSI)      ,CALZ(KCHI,KPSI)
     1,F(KCHI,KPSI)        ,DINT(KINT,4)        ,ICAL(KPSI)
      DATA       IDBL/ 2 /
C * * F
      CALL SECOND(TIMEF)
      CALL CLEAR(0,D0,F,NPSI*NCHI)
      DO 130 N=1,NPSI
      DO 110 I=1,NCHI
      II = I
      IF(CALR(I,N)) 110,120,110
  110 ICAL(N) = NCHI
      GO TO 130
  120 ICAL(N) = II-1
  130 CONTINUE
      DO 140 N=2,NPSI
      PSI0 = PSI(N)
      PSI1 = PSI(N-1)
      NN = N
      IF(ICAL(N) .NE. NCHI) GO TO 150
      DO 140 I=2,NCHI
      CALL INTEG(TEMP1,I,N,CALR,CALZ,DINT(1,1),DINT(1,2),DINT(1,3)
     1,DINT(1,4),PSI0,PSI1,EPS2,LIM1,IPR2,NCHI,NPSI,KINT)
  140 F(I,N) = F(I-1,N)+TEMP1
      NPSIM = NPSI
      GO TO 160
  150 NPSIM = NN-1
  160 CONTINUE
      IF(NASK(IPR2,2) .NE. -1) GO TO 170
      CALL OUTPTM(F,NCHI,NPSI,0,0,-8,NCHI,NPSI,1,1HF,IPAGE)
      CALL SECOND(TIMEL)
      TIMEF = TIMEL-TIMEF
      WRITE(IOUT,1000) TIMEF
  170 CONTINUE
      CALL RANDM(IO1,F,NCHI*NPSI*IDBL,3,1)
      RETURN
 1000 FORMAT(35H0 STEP2 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

$$F = \int_0^l \frac{1}{R}\sqrt{p^2+g^2}\,ds$$

* SOURCE STATEMENT *

```
*DK FUNCF
      DOUBLE PRECISION FUNCTION FUNCF(CALR,CALZ)                       GGG
      DOUBLE PRECISION     FUNCI     ,FUNCP     ,FUNCQ
      DOUBLE     CALR      ,CALZ     ,P         ,Q
      P = FUNCP(CALR,CALZ)
      Q = FUNCQ(CALR,CALZ)
      FUNCF = DSQRT(P**2+Q**2)/CALR
      RETURN
      END
```

$$F = \frac{\sqrt{p^2+g^2}}{R}$$

* SOURCE STATEMENT *

```
*DK STEP3      χ, J, μ の計算
      SUBROUTINE STEP3(PSI,CALR,CALZ,F,VALU,IPR,KCHI,KPSI)
      DIMENSION  PSI(KPSI),CALR(KCHI,KPSI)      ,CALZ(KCHI,KPSI)
     1,F(KCHI,KPSI)        ,VALU(KCHI,KPSI)
      DOUBLE     PSI       ,CALR      ,CALZ      ,F         ,VALU
     1,F0        ,PIF       ,CALR1     ,CALZ1     ,TEMPO     ,PSII
      DOUBLE PRECISION     FUNCI     ,FUNCP     ,FUNCQ     ,FUNCF
     1,PSI0      ,PSI1
      COMMON     / IO /    IOUN      ,IOUT      ,IO1
      COMMON     / SEISUU /NPSI      ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI      ,NTRIM
      DATA       PI/3.1415926535898/,IDBL/2/
      CALL SECOND(TIMEF)
C     CHI
      CALL CLEAR(0,D0,VALU,NPSI*NCHI)
      DO 110 N=2,NPSIM
      F0 = F(NCHI,N)
      PIF = PI/F0
      DO 110 I=1,NCHI
  110 VALU(I,N) = PIF*F(I,N)
      CALL RANDM(IO1,VALU,NCHI*NPSI*IDBL,4,1)
      IF(NASK(IPR,1) .EQ. -1) CALL OUTP(VALU,NCHI,NPSIM,NCHI,NPSI,3
     1,3HCHI)
C     JACOBIAN
      CALL CLEAR(0,D0,VALU,NPSI*NCHI)
      DO 120 N=2,NPSIM
      F0 = F(NCHI,N)
      PIF = F0/PI
      DO 120 I=1,NCHI
      CALR1 = CALR(I,N)
      CALZ1 = CALZ(I,N)
      TEMPO = PIF*CALR1**2
  120 VALU(I,N) = TEMPO/(FUNCP(CALR1,CALZ)**2+FUNCQ(CALR1,CALZ1)**2)
      CALL RANDM(IO1,VALU,NCHI*NPSI*IDBL,5,1)
      IF(NASK(IPR,1) .EQ. -1) CALL OUTP(VALU,NCHI,NPSIM,NCHI,NPSI,8
     1,8HJACOBIAN)
C     MU
      PSI0 = PSI(NPSI-1)
      DO 130 N=2,NPSIM
      PSI1 = PSI(N)
      DO 130 I=1,NCHI
      PSII = FUNCI(PSI0,PSI1)
  130 VALU(I,N) = VALU(I,N)*PSII/CALR(I,N)**2
      CALL RANDM(IO1,VALU,NCHI*NPSI*IDBL,6,1)
      IF(NASK(IPR,1) .EQ. -1) CALL OUTP(VALU,NCHI,NPSIM,NCHI,NPSI,2,2HMU
     1)
      IF(NASK(IPR,2) .NE. -1) GO TO 140
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1000) TIME
  140 CONTINUE
 1000 FORMAT(35H0 STEP3 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      RETURN
      END
```

$$\chi = \frac{\pi \cdot F}{F_0}$$

$$J = \frac{F_0 \cdot R^2}{\pi\,(p^2+g^2)}$$

$$\mu = \frac{I \cdot J}{R^2}$$

* SOURCE STATEMENT *


```
*DK STEP4.    J', μ', Bx の計算
      SUBROUTINE STEP4(PSI,CALR,CALZ,VALU1,VALU2,VALU3,DINT,RDIF,ZDIF
     1,KCHI,KPSI,KINT,EPS2,EPS3,LIM1,LIM3,IPR)
      DIMENSION  PSI(KPSI),CALR(KCHI,KPSI)     ,CALZ(KCHI,KPSI)
     1,VALU1(KCHI,KPSI)    ,VALU2(KCHI,KPSI)    ,VALU3(KCHI,KPSI)
     2,DINT(KINT,4)        ,RDIF(KCHI,KPSI)     ,ZDIF(KCHI,KPSI)
      COMMON / PSI / EI(10)                                               GGG
      COMMON      / IO /     IOIN      ,IOUT      ,IO1
      COMMON      / SEISUU /NPSI       ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN      ,NTRI      ,NTRIM
      COMMON      / R2 /     RMAX      ,RMIN      ,ZMAX      ,RCEN
      COMMON      / PLAS /   TITLE(18),IPAGE
      COMMON / COM20 / P1,P2,B1,B2
      DOUBLE      PSI        ,CALR      ,CALZ      ,VALU1     ,VALU2     ,VALU3
     1,DINT       ,RDIF      ,ZDIF      ,DPSI      ,DELZ      ,PSI0      ,PSI1
     2,CALR0      ,CALZ0     ,P@        ,CALR3     ,CALZ3     ,CALR1     ,THETA
     3,TEMP1      ,DELR      ,RZERO                                       GGG
      DOUBLE PRECISION      FUNCI     ,FUNCP     ,FUNC@     ,FUNCF
      DATA        PI/3.1415926535898/,IDBL/2/
      CALL SECOND(TIMEF)
C     JACOBIAN-DASH
      RZERO=RCEN                                                          GGG
      CALL CLEAR(0.D0,RDIF,NCHI*NPSI)
      CALL CLEAR(0.D0,ZDIF,NCHI*NPSI)
      DPSI = (PSI(NPSI)-PSI(1))/(FLOAT(NPSI-1)*FLOAT(NDIE))
      NCHIM = NCHI/2+1
      DELZ = CALZ(NCHIM,NPSIM)/(NPSIM*NDIE)
      NCHIM = NCHI-1
      DO 110 N=2,NPSIM
      PSI1 = PSI(N)
      IF(N .LT. NPSI) PSI0 = PSI1+(PSI(N+1)-PSI1)/NDIE
      IF(N .EQ. NPSI) PSI0 = PSI1+DPSI
      DO 110 I=2,NCHIM
      CALR0 = CALR(I,N)
      CALZ0 = CALZ(I,N)
      P@ = FUNCP(CALR0,CALZ0)/FUNC@(CALR0,CALZ0)
      CALL DIFF(CALR0,CALZ0,CALR3,CALZ3,PSI0,PSI1,P@,DELZ,I,N,EPS3,LIM3
     1,IPR)
      RDIF(I,N) = CALR3
  110 ZDIF(I,N) = CALZ3
      CALL CLEAR(RZERO,RDIF,NCHI)                                         GGG
      DELR = (RMAX-RCEN)/(NPSI*NDIE*4.0)
      DO 140 N=2,NPSIM
      PSI1 = PSI(N)
      IF(N .LT. NPSI) PSI0 = PSI1+(PSI(N+1)-PSI1)/NDIE
      IF(N .EQ. NPSI) PSI0 = PSI1+DPSI
      DO 140 I=1,NCHI,NCHIM
      CALR1 = CALR(I,N)
      IF(I .GT. 1) GO TO 120
      THETA = 0.0
      GO TO 130
  120 THETA = PI
  130 CALL DIST(CALR1,THETA,CALR3,CALZ3,PSI1,PSI0,DELR,LIM3,0.1*EPS3,
     *        IPR,4)
      RDIF(I,N) = CALR3
  140 ZDIF(I,N) = CALZ3
      CALL RANDM(IO1,RDIF,NCHI*NPSI*IDBL,7,1)
      CALL RANDM(IO1,ZDIF,NCHI*NPSI*IDBL,8,1)
      DO 150 N=2,NPSIM
      NPM = N
      DO 150 I=1,NCHI
      IF(RDIF(I,N)) 160,160,150
  150 CONTINUE
      GO TO 170
  160 NPSIM = NPM-1
  170 CONTINUE
      CALL CLEAR(0.D0,VALU1,NCHI*NPSI)
      DO 180 N=2,NPSIM
      IF(N .LT. NPSI) PSI0 = PSI(N)+(PSI(N+1)-PSI(N))/NDIE
      IF(N .EQ. NPSI) PSI0 = PSI(N)+DPSI
      PSI1 = PSI(N-1)+(PSI(N)-PSI(N-1))/NDIE
      DO 180 I=2,NCHI
      CALL INTEG(TEMP1,I,N,RDIF,ZDIF,DINT(1,1),DINT(1,2),DINT(1,3)
     1,DINT(1,4),PSI0,PSI1,EPS2,LIM1,IPR,NCHI,NPSI,KINT)
  180 VALU1(I,N) = VALU1(I-1,N)+TEMP1
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,9,1)
      CALL CLEAR(0.D0,VALU2,NCHI*NPSI)
      IF(NASK(IPR,2) .NE. -1) GO TO 190
      CALL OUTPTM(RDIF,NCHI,NPSIM,0,0,-8,NCHI,NPSI,4,4HRDIF,IPAGE)
      CALL OUTPTM(ZDIF,NCHI,NPSIM,0,0,-8,NCHI,NPSI,4,4HZDIF,IPAGE)
      CALL OUTPTM(VALU1,NCHI,NPSIM,0,0,-8,NCHI,NPSI,6,6HF(DIF),IPAGE)
  190 CONTINUE
      CALL CLEAR(0.D0,VALU2,NCHI*NPSI)
      DO 200 N=2,NPSIM
      TEMP1 = VALU1(NCHI,N)/PI
      DO 200 I=1,NCHI
      CALR0 = RDIF(I,N)
      CALZ0 = ZDIF(I,N)
  200 VALU2(I,N) = TEMP1*CALR0**2/(FUNCP(CALR0,CALZ0)**2+FUNC@(CALR0
     1,CALZ0)**2)
      CALL RANDM(IO1,VALU2,NCHI*NPSI*IDBL,10,1)
      IF(NASK(IPR,2) .EQ. -1) CALL OUTPTM(VALU2,NCHI,NPSIM,0,0,-8,NCHI
     1,NPSI,9,9HJACOBIAN2,IPAGE)
      CALL CLEAR(0.D0,VALU1,NCHI*NPSI)
      CALL RANDM(IO1,VALU3,NCHI*NPSI*IDBL,5,2)
      DO 210 N=2,NPSIM
      TEMP1 = DPSI
      IF(N .LT. NPSI) TEMP1 = (PSI(N+1)-PSI(N))/NDIE
      DO 210 I=1,NCHI
  210 VALU1(I,N) = (VALU2(I,N)-VALU3(I,N))/TEMP1
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,11,1)
      IF(NASK(IPR,1) .EQ. -1) CALL OUTP(VALU1,NCHI,NPSIM,NCHI,NPSI,13
     1,13HJACOBIAN-DASH)
C     MU-DASH
      CALL CLEAR(0.D0,VALU1,NCHI*NPSI)
      PSI0 = PSI(NPSI-1)+DPSI
      DO 220 N=1,NPSIM
      TEMP1 = FUNCI(PSI0,PSI(N)+DPSI)
      DO 220 I=1,NCHI
  220 VALU1(I,N) = TEMP1*VALU2(I,N)/RDIF(I,N)**2
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,12,1)
      IF(NASK(IPR,2) .EQ. -1) CALL OUTPTM(VALU1,NCHI,NPSIM,0,0,-8,NCHI
     1,NPSI,3,3HMU2,IPAGE)
      CALL CLEAR(0.D0,VALU2,NCHI*NPSI)
      CALL  RANDM(IO1,VALU3,NCHI*NPSI*IDBL,6,2)
      DO 230 N=2,NPSIM
      TEMP1 = DPSI
      IF(N .LT. NPSI) TEMP1 = (PSI(N+1)-PSI(N))/NDIE
      DO 230 I=1,NCHI
  230 VALU2(I,N) = (VALU1(I,N)-VALU3(I,N))/TEMP1
      CALL RANDM(IO1,VALU2,NCHI*NPSI*IDBL,13,1)
      IF(NASK(IPR,1) .EQ. -1) CALL OUTP(VALU2,NCHI,NPSIM,NCHI,NPSI,7
     1,7HMU-DASH)
C     B-CHI
      CALL CLEAR(0.D0,VALU1,NCHI*NPSI)
      DO 240 N=2,NPSI
      DO 240 I=1,NCHI
```

```
      CALPO = CALR(I,N)
      CALZO = CALZ(I,N)
      VALU1(I,N) = DSQRT(FUNCP(CALRO,CALZO)**2+FUNCQ(CALRO,CALZO)**2)
  240 VALU1(I,N) = VALU1(I,N)/CALRO
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,14,1)
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,15,1)                       GGG
      IF(NASK(IPR,2) .EQ. -1) CALL OUTPTM(VALU1,NCHI,NPSI,0,0,-8,NCHI
     1,NPSI,12,12HB-CHI(CHECK),IPAGE)
C     TOROIDAL CURRENT
      CALL CLEAR(0.D0,VALU1,NCHI*NPSI)
      B=P1+2.0*P2*PSI(1)
      A=0.5*B1+B2*PSI(1)
      TEMP1= B*CALR(1,1)+A/CALR(1,1)
      DO 250 N=1,NCHI                                                 GGG
  250 VALU1(N,1)=TEMP1                                                GGG
      DO 260 N=2,NPSI                                                 GGG
      B=P1+2.0*P2*PSI(N)
      A=0.5*B1+B2*PSI(N)
      DO 260 I=1,NCHI                                                 GGG
  260 VALU1(I,N)= B*CALR(I,N)+A/CALR(I,N)
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,16,1)
      IF(NASK(IPR,1) .EQ. -1) CALL OUTP(VALU1,NCHI,NPSIM,NCHI,NPSI,16
     1,16HTOROIDAL CURRENT)
      IF(NASK(IPR,2) .NE. -1) GO TO 300
      CALL SECOND(TIMEL)
      TIME = TIMFL-TIMEF
      WRITE(IOUT,1000) TIME
  300 CONTINUE
 1000 FORMAT(35H0 STEP4 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK STEP5          jp, BT の計算
      SUBROUTINE STEP5(PSI,CALR,CALZ,VALU1,VALU2,KCHI,KPSI,IPR2,IPR3)
      DIMENSION  PSI(KPSI),CALR(KCHI,KPSI)     ,CALZ(KCHI,KPSI)
     1,VALU1(KCHI,KPSI)  ,VALU2(KCHI,KPSI)
      DOUBLE      PSI       ,CALR      ,CALZ      ,VALU1     ,VALU2     ,TEMP1
     1,TEMP2
      COMMON      / IO /    IOIN      ,IOUT      ,IO1
      COMMON      / SEISUU /NPSI      ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI      ,NTRIM                                    GGG
      DOUBLE PRECISION      FUNCI     ,FUNCP     ,FUNCQ     ,FUNCF     ,FNDI
      DATA        IDBL/ 2 /
      CALL SECOND(TIMEF)
C * * B-CHI
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,14,2)
      IF(NASK(IPR2,1) .EQ. -1) CALL OUTP(VALU1,NCHI,NPSIM,NCHI,NPSI,5
     1,5HB-CHI)
C * * POLOIDAL CURRENT
      CALL CLEAR(0.D0,VALU2,NCHI*NPSI)
      TEMP1 = PSI(NPSI-1)
      DO 100 N=1,NPSIM
      TEMP2 = PSI(N)
      DO 100 I=1,NCHI
  100 VALU2(I,N) = -VALU1(I,N)*FNDI(TEMP1,TEMP2)
      CALL RANDM(IO1,VALU2,NCHI*NPSI*IDBL,17,1)
      IF(NASK(IPR3,1) .EQ. -1) CALL OUTP(VALU2,NCHI,NPSIM,NCHI,NPSI,16
     1,16HPOLOIDAL CURRENT)
C * * B-TOROIDAL
      CALL CLEAR(0.D0,VALU1,NCHI*NPSI)
      PSI0 = PSI(NPSI-1)
      DO 110 N=1,NPSIM
      TEMP1 = FUNCI(PSI0,PSI(N))
      DO 110 I=1,NCHI
  110 VALU1(I,N) = TEMP1/CALR(I,N)
      CALL RANDM(IO1,VALU1,NCHI*NPSI*IDBL,18,1)
      IF(NASK(IPR3,1) .EQ. -1) CALL OUTP(VALU1,NCHI,NPSIM,NCHI,NPSI,10
     1,10HB-TOROIDAL)
      IF(NASK(IPR2,2) .NE. -1) GO TO 120
      CALL SECOND(TIMEL)
      TIME = TIMFL-TIMEF
      WRITE(IOUT,1000) TIME
  120 CONTINUE
 1000 FORMAT(35H0 STEP5 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK STEP6          3角形要素の重心,面積計算
      SUBROUTINE STEP6(CALR,CALZ,RG,ZG,SS,KCHI,KPSI,KTRI,IPR)
      DIMENSION  CALR(KCHI,KPSI)     ,CALZ(KCHI,KPSI)     ,RG(KTRI,KPSI)
     1,ZG(KTRI,KPSI)        ,SS(KTRI,KPSI)
      COMMON      / IO /    IOIN      ,IOUT      ,IO1
      COMMON      / SEISUU /NPSI      ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI      ,NTRIM
      COMMON      / PLAS /  TITLE(18),IPAGE
      DOUBLE PRECISION      CALR      ,CALZ      ,RG        ,ZG        ,SS
      DATA        IDBL/ 2 /
      CALL SECOND(TIMEF)
C * * TRIANGULAR FINITE ELEMENTS
      NPSIN = NPSIM-1
      DO 100 N=1,NPSIN
      M = 0
      NCHIM = NCHI-1
      DO 100 I=1,NCHIM
      M = M+1
      RG(M,N) = (CALR(I,N)+CALR(I,N+1)+CALR(I+1,N+1))/3.0
      ZG(M,N) = (CALZ(I,N)+CALZ(I,N+1)+CALZ(I+1,N+1))/3.0
      SS(M,N) = 0.5*(CALR(I,N+1)*CALZ(I+1,N+1)-CALR(I+1,N+1)*CALZ(I,N+1)
     1+CALR(I+1,N+1)*CALZ(I,N)-CALR(I,N)*CALZ(I+1,N+1)
     2+CALR(I,N)*CALZ(I,N+1)-CALR(I,N+1)*CALZ(I,N))
      M = M+1
      RG(M,N) = (CALR(I,N)+CALR(I+1,N)+CALR(I+1,N+1))/3.0
      ZG(M,N) = (CALZ(I,N)+CALZ(I+1,N)+CALZ(I+1,N+1))/3.0
      SS(M,N) = 0.5*(CALR(I+1,N+1)*CALZ(I+1,N)-CALR(I+1,N)*CALZ(I+1,N+1)
     1+CALR(I+1,N)*CALZ(I,N)-CALR(I,N)*CALZ(I+1,N)
     2+CALR(I,N)*CALZ(I+1,N+1)-CALR(I+1,N+1)*CALZ(I,N))
  100 CONTINUE
      NTRIM = M
      CALL RANDM(IO1,RG,NTRI*NPSI*IDBL,19,1)
      CALL RANDM(IO1,ZG,NTRI*NPSI*IDBL,20,1)
      CALL RANDM(IO1,SS,NTRI*NPSI*IDBL,21,1)
      IF(NASK(IPR,2) .NE. -1) GO TO 110
      CALL OUTPTM(RG,NTRI,NPSIN,0,0,-8,NTRI,NPSI,19,19HTRIANGULAR CENTER
     1 R,IPAGE)
      CALL OUTPTM(ZG,NTRI,NPSIN,0,0,-8,NTRI,NPSI,19,19HTRIANGULAR CENTER
     1 Z,IPAGE)
      CALL OUTPTM(SS,NTRI,NPSIN,0,0,-8,NTRI,NPSI,16,16HAREA OF TRIANGLE
     1,IPAGE)
      CALL SECOND(TIMEL)
      TIME = TIMFL-TIMEF
      WRITE(IOUT,1000) TIME
  110 CONTINUE
      RETURN
 1000 FORMAT(35H0 STEP6 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

* SOURCE STATEMENT *

```
*DK STEP7
      SUBROUTINE STEP7(PSI,TOLCU,RG,ZG,SS,AREA,JACB2,UDASH,VALUV,TOLOI
     1,POLOI,KCHI,KPSI,KTRI,IPR)
      DIMENSION  PSI(KPSI),TOLCU(KCHI,KPSI)     ,RG(KTRI,KPSI)
     1,ZG(KTRI,KPSI)         ,SS(KTRI,KPSI)       ,AREA(KPSI)
     2,JACB2(KPSI)           ,UDASH(KPSI)         ,VALUV(KPSI)
     3,TOLOI(KPSI)           ,POLOI(KPSI)
      COMMON     / IO /      IOIN      ,IOUT      ,IO1
      COMMON     / SEISUU /NPSI       ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI      ,NTRIM
      COMMON     / PLAS /   TITLE(18),IPAGE
      COMMON / COM20 / P1,P2,B1,B2
      DOUBLE PRECISION      PSI       ,TOLCU     ,RG        ,ZG        ,SS
     1,AREA      ,JACB2     ,UDASH     ,VALUV     ,TOLOI     ,POLOI     ,FUNCI
     2,TEMP1     ,PSI0      ,PSI1
      DATA       IDBL/2/   ,PI/3.1415926535898/
      CALL SECOND(TIMEF)
C * * AREA OF PLASMA CROSS SECTION
      CALL CLEAR(0.D0,AREA,NPSI)
      DO 100 N=1,NPSIN
      DO 100 M=1,NTRIM
  100 AREA(N) = AREA(N)+2.0*SS(M,N)
      CALL RANDM(IO1,AREA,NPSI*IDBL,22,1)
      IF(NASK(IPR,3) .NE. -1) GO TO 110
      WRITE(IOUT,1000)
      WRITE(IOUT,1010) (AREA(N),N=1,NPSIN)
  110 CONTINUE
C * * J2
      CALL RANDM(IO1,TOLCU,NCHI*NPSI*IDBL,16,2)
      CALL CLEAR(0.D0,JACB2,NPSI)
      NCHIM = NCHI-1
      PDCUR=0.0
      DO 130 N=1,NPSIN
      PSSS=0.5*(PSI(N)+PSI(N+1))
      M = 0
      DO 120 I=1,NCHIM
      M = M+1
      JACB2(N) = JACB2(N)+(TOLCU(I,N)+TOLCU(I,N+1)+TOLCU(I+1,N+1))/3.0
     1*SS(M,N)
      IF(N.LT.NPSIN) PDCUR=PDCUR+(P1+2.0*P2*PSSS)*RG(M,N)*SS(M,N)
      M = M+1
      JACB2(N) = JACB2(N)+(TOLCU(I,N)+TOLCU(I+1,N)+TOLCU(I+1,N+1))/3.0
     1*SS(M,N)
      IF(N.LT.NPSIN) PDCUR=PDCUR+(P1+2.0*P2*PSSS)*RG(M,N)*SS(M,N)
  120 CONTINUE
  130 JACB2(N) = 2.0*JACB2(N)
      TCUR=0.0
      DO 131 N=1,NPSIN-1
  131 TCUR=TCUR+JACB2(N)
      PDCUR=2.0*PDCUR
      BETA=PDCUR/TCUR
      WRITE(6,1200)BETA,PDCUR,TCUR
 1200 FORMAT(//10X,'** STEP 7 **',10X,'BETA POLOIDAL =',F10.3,10X,
     *     'P-DASH CURNT, T-CURNT =',1P2E12.3/)
      CALL RANDM(IO1,JACB2,NPSI*IDBL,23,1)
      IF(NASK(IPR,3) .NE. -1) GO TO 140
      WRITE(IOUT,1020)
      WRITE(IOUT,1010) (JACB2(N),N=1,NPSIN)
  140 CONTINUE
C * * U-DASH
      CALL CLEAR(0.D0,UDASH,NPSI)
      DO 160 N=1,NPSIN
      TEMP1 = 0.0
      DO 150 M=1,NTRIM
  150 TEMP1 = TEMP1+SS(M,N)/RG(M,N)
  160 UDASH(N) = 4.0*PI*TEMP1/(PSI(N+1)-PSI(N))
      CALL RANDM(IO1,UDASH,NPSI*IDBL,24,1)
      IF(NASK(IPR,3) .NE. -1) GO TO 170
      WRITE(IOUT,1030)
      WRITE(IOUT,1010) (UDASH(N),N=1,NPSIN)
  170 CONTINUE
C * * V
      CALL CLEAR(0.D0,VALUV,NPSI)
      DO 190 N=1,NPSIN
      TEMP1 = 0.0
      DO 180 M=1,NTRIM
  180 TEMP1 = TEMP1+RG(M,N)*SS(M,N)
  190 VALUV(N) = 4.0*PI*TEMP1                                            GGG
      CALL RANDM(IO1,VALUV,NPSI*IDBL,25,1)
      IF(NASK(IPR,3) .NE. -1) GO TO 200
      WRITE(IOUT,1040)
      WRITE(IOUT,1010) (VALUV(N),N=1,NPSIN)
  200 CONTINUE
C * * TOROIDAL MAGNETIC FLUX
      CALL CLEAR(0.D0,TOLOI,NPSI)
      PSI0 = PSI(NPSI-1)
      DO 220 N=1,NPSIN
      TEMP1 = 0.0
      DO 210 M=1,NTRIM
      CALL CALPSI(RG(M,N),ZG(M,N),PSI1)
  210 TEMP1 = TEMP1+FUNCI(PSI0,PSI1)/RG(M,N)*SS(M,N)
  220 TOLOI(N) = 2.0*TEMP1
      CALL RANDM(IO1,TOLOI,NPSI*IDBL,26,1)
      IF(NASK(IPR,3) .NE. -1) GO TO 230
      WRITE(IOUT,1050)
      WRITE(IOUT,1010) (TOLOI(N),N=1,NPSIN)
  230 CONTINUE
C * * POLOIDAL MAGNETIC FLUX
      CALL CLEAR(0.D0,POLOI,NPSI)
      DO 240 N=1,NPSIN
  240 POLOI(N) = 2.0*PI*(PSI(N+1)-PSI(N))
      CALL RANDM(IO1,POLOI,NPSI*IDBL,27,1)
      IF(NASK(IPR,3) .NE. -1) GO TO 250
      WRITE(IOUT,1060)
      WRITE(IOUT,1010) (POLOI(N),N=1,NPSIN)
  250 CONTINUE
C * * OUTPUT
      IF(NASK(IPR,2) .NE. -1) GO TO 290                                  GGG
      IPAGE = IPAGE+1
      WRITE(IOUT,1070) (TITLE(I),I=1,18),IPAGE
      IF(NASK(IPR,2) .EQ. -1) GO TO 270
      WRITE(IOUT,1080)
      DO 260 N=1,NPSIN
  260 WRITE(IOUT,1090) N,TOLOI(N),POLOI(N)
      GO TO 290
  270 WRITE(IOUT,1100)
      DO 280 N=1,NPSIN
  280 WRITE(IOUT,1090) N,TOLOI(N),POLOI(N),AREA(N),JACB2(N),UDASH(N)
     1,VALUV(N)
  290 CONTINUE
      IF(NASK(IPR,2) .NE. -1) GO TO 300
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1110) TIME
  300 CONTINUE
      RETURN
 1000 FORMAT(9H0 AREA(K))
 1010 FORMAT(6X,1P8D16.8)
```

```
 1020 FORMAT(7H0 J2(K))
 1030 FORMAT(11H0 U-DASH(K))
 1040 FORMAT(3H0 V)
 1050 FORMAT(24H0 TOROIDAL MAGNETIC FLUX)
 1060 FORMAT(24H0 POROIDAL MAGNETIC FLUX)
 1070 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3)
 1080 FORMAT(38H0               TOROIDAL          POROIDAL     /38H  PSI      MAGN
     1ETIC          MAGNETIC     /37H            FLUX                FLUX      )      GGG
 1090 FORMAT(1X,I5,2X,1P8D16.8)
 1100 FORMAT(35H0               TOROIDAL          POROIDAL     /102H  PSI      MAG
     1NETIC          MAGNETIC              AREA            J2              U-DA
     2SH               V          /38H               FLUX                FLUX      )
 1110 FORMAT(35H0 STEP7 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

• SOURCE STATEMENT •

```
*DK STEP8        Solov'ev の 不安定性係数 の 準備
      SUBROUTINE STEP8(PSI,RG,ZG,SS,Q1,Q2,Q3,Q4,Q5,KPSI,KTRI,IPR)
      DIMENSION  PSI(KPSI),RG(KTRI,KPSI)          ,ZG(KTRI,KPSI)
     1,SS(KTRI,KPSI)        ,Q1(KPSI)  ,Q2(KPSI)  ,Q3(KPSI)  ,Q4(KPSI)
     2,Q5(KPSI)
      COMMON       / IO /      IOIN       ,IOUT       ,IO1
      COMMON       / SEISUU /NPSI        ,NCHI       ,NDIE       ,NPSIM      ,NINT
     1,NPSIN       ,NTRI       ,NTRIM
      COMMON       / PLAS /   TITLE(18),IPAGE
      DOUBLE PRECISION     PSI        ,RG         ,ZG         ,SS         ,Q1
     1,Q2          ,Q3         ,Q4         ,Q5         ,TEMPO      ,TEMP1      ,FUNCP
     2,FUNCQ
      DOUBLE RA,ZA,SA,PA,QA                                                      GGG
      DATA         PI/3.1415926535898/,IDBL/2/
      CALL SECOND(TIMEF)
      CALL CLEAR(0.D0,Q1,KPSI)                                                   GGG
      CALL CLEAR(0.D0,Q2,KPSI)                                                   GGG
      CALL CLEAR(0.D0,Q3,KPSI)                                                   GGG
      CALL CLEAR(0.D0,Q4,KPSI)                                                   GGG
      CALL CLEAR(0.D0,Q5,KPSI)                                                   GGG
      DO 110 N=1,NPSIN
      TEMPO = 4.0*PI/(PSI(N+1)-PSI(N))
      DO 100 M=1,NTRIM
      RA=RG(M,N)                                                                 GGG
      ZA=ZG(M,N)                                                                 GGG
      SA=SS(M,N)                                                                 GGG
      PA=FUNCP(RA,ZA)                                                            GGG
      QA=FUNCQ(RA,ZA)                                                            GGG
      TEMP1=PA**2+QA**2                                                          GGG
      Q1(N)=Q1(N)+RA*SA/TEMP1                                                    GGG
      Q2(N)=Q2(N)+SA/RA/TEMP1                                                    GGG
      Q3(N)=Q3(N)+SA/TEMP1*RA**3                                                 GGG
      Q4(N)=Q4(N)+SA*RA**3                                                       GGG
  100 Q5(N)=Q5(N)+SA*TEMP1/RA                                                    GGG
      Q1(N) = TEMPO*Q1(N)
      Q2(N) = TEMPO*Q2(N)
      Q3(N) = TEMPO*Q3(N)
      Q4(N) = TEMPO*Q4(N)
  110 Q5(N) = TEMPO*Q5(N)
      CALL RANDM(IO1,Q1,NPSI*IDBL,28,1)
      CALL RANDM(IO1,Q2,NPSI*IDBL,29,1)
      CALL RANDM(IO1,Q3,NPSI*IDBL,30,1)
      CALL RANDM(IO1,Q4,NPSI*IDBL,31,1)
      CALL RANDM(IO1,Q5,NPSI*IDBL,32,1)
      IF(NASK(IPR,2) .NE. -1) GO TO 130
      IPAGE = IPAGE+1
      WRITE(IOUT,1000) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1010)
      DO 120 N=1,NPSIN
  120 WRITE(IOUT,1020) N,Q1(N),Q2(N),Q3(N),Q4(N),Q5(N)
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1030) TIME
  130 CONTINUE
 1000 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3)
 1010 FORMAT(86H0 PSI              Q1                    Q2                    Q3
     1          Q4                Q5             )
 1030 FORMAT(35H0 STEP8 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
 1020 FORMAT(1X,I5,1X,1P6D16.8)                                                  GGG
      RETURN
      END
```

• SOURCE STATEMENT •

```
*DK STEP9          V1' , V2'
      SUBROUTINE STEP9(PSI,VALUV,TOLOI,V1DASH,V2DASH,KPSI,IPR)
      DIMENSION  PSI(KPSI),VALUV(KPSI)          ,TOLOI(KPSI)
     1,V1DASH(KPSI)        ,V2DASH(KPSI)
      COMMON       / IO /      IOIN       ,IOUT       ,IO1
      COMMON       / SEISUU /NPSI        ,NCHI       ,NDIE       ,NPSIM      ,NINT
     1,NPSIN       ,NTRI       ,NTRIM
      COMMON       / PLAS /   TITLE(18),IPAGE
      DOUBLE PRECISION     PSI        ,VALUV      ,TOLOI      ,V1DASH
     1,V2DASH
      DATA         IDBL/2/
      CALL SECOND(TIMEF)
      CALL RANDM(IO1,VALUV,NPSI*IDBL,25,2)
      CALL RANDM(IO1,TOLOI,NPSI*IDBL,26,2)
      DO 100 N=1,NPSIN
      V1DASH(N) = VALUV(N)/(PSI(N+1)-PSI(N))
  100 V2DASH(N) = VALUV(N)/TOLOI(N)
      CALL RANDM(IO1,V1DASH,NPSI*IDBL,33,1)
      CALL RANDM(IO1,V2DASH,NPSI*IDBL,34,1)                                      GGG
      IF(NASK(IPR,2) .NE. -1) GO TO 140
      IF(NASK(IPR,2) .EQ. -1) GO TO 120
      IPAGE = IPAGE+1
      WRITE(IOUT,1000) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1010)
      DO 110 N=1,NPSIN
  110 WRITE(IOUT,1020) N,V2DASH(N)
      GO TO 140
  120 IPAGE = IPAGE+1
      WRITE(IOUT,1000) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1030)
      DO 130 N=1,NPSIN
  130 WRITE(IOUT,1020) N,V2DASH(N),V1DASH(N)
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1040) TIME
  140 CONTINUE
      RETURN
 1000 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3)
 1010 FORMAT(22H0 PSI    DV/(DPHAI-T)    )                                       GGG
 1020 FORMAT(1X,I5,1X,1P6D16.8)
 1030 FORMAT(38H0 PSI    DV/(DPHAI-T)              V2-DASH      )
 1040 FORMAT(35H0 STEP9 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

Shear, Well depth, g の計算

```
*DK STEP10
      SUBROUTINE STEP10(TOLOI,POLOI,V2DASH,SHEAR,DEPTH,STAB,KPSI,IPR)
      DIMENSION  TOLOI(KPSI)             ,POLOI(KPSI)          ,V2DASH(KPSI)
     1,SHEAR(KPSI)           ,DEPTH(KPSI)           ,STAB(KPSI)
      DOUBLE PRECISION     TOLOI     ,POLOI     ,V2DASH    ,SHEAR     ,DEPTH
     1,STAB
      COMMON / REPEAT / LMN,COEFX
      COMMON      / IO /     IOIN      ,IOUT      ,IO1
      COMMON      / SEISUU /NPSI      ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI     ,NTRIM
      COMMON      / PLAS /  TITLE(18),IPAGE
      DATA        IDBL/2/
      CALL SECOND(TIMEF)
      CALL RANDM(IO1,TOLOI,NPSI*IDBL,26,2)
      CALL RANDM(IO1,POLOI,NPSI*IDBL,27,2)
      CALL RANDM(IO1,V2DASH,NPSI*IDBL,34,2)
C * * SHEAR
      CALL CLEAR(0.D0,SHEAR,NPSI)
      NPSIO = NPSIN-1
      DO 100 N=1,NPSIO
  100 SHEAR(N) = 2.0*(POLOI(N+1)/TOLOI(N+1)-POLOI(N)/TOLOI(N))/
     1(TOLOI(N+1)+TOLOI(N))*COEFX
      TTOL=0.0
      DO 90 N=1,NPSIO
   90 TTOL=TTOL+TOLOI(N)
      DO 105 N=1,NPSIO
  105 SHEAR(N)=SHEAR(N)*TTOL
      CALL RANDM(IO1,SHEAR,NPSI*IDBL,35,1)
C * * WELL DEPTH
      DO 110 N=1,NPSIN
  110 DEPTH(N) = (V2DASH(N)-V2DASH(NPSIO))   /V2DASH(N)
      CALL RANDM(IO1,DEPTH,NPSI*IDBL,36,1)
C * * STABILITY FACTOR
      DO 120 N=1,NPSIN
  120 STAB(N) = TOLOI(N)/POLOI(N)/COEFX
      CALL RANDM(IO1,STAB,NPSI*IDBL,37,1)
C * * OUT PUT
      IF(NASK(IPR,1) .NE. -1) GO TO 140
      IPAGE = IPAGE+1
      WRITE(IOUT,1000) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1010)
      DO 130 N=1,NPSIN
  130 WRITE(IOUT,1020) N,SHEAR(N),DEPTH(N),STAB(N)
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1030) TIME
  140 CONTINUE
      RETURN
 1000 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3)
 1010 FORMAT(54H0 PSI          SHEAR           WELL DEPTH         STABILITY   /
     138X,16H      FACTOR      )
 1020 FORMAT(1X,I5,1X,1P6D16.8)                                                  GGG
 1030 FORMAT(36H0 STEP10 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

* SOURCE STATEMENT *

U″, V″ の計算

```
*DK STEP11
      SUBROUTINE STEP11(PSI,CALR,UDASH,V1DASH,UDASH2,VDASH2,WVALU,SHEAR
     1,WELL,KPSI,KCHI,IPR)
      DIMENSION  PSI(KPSI),UDASH(KPSI)             ,V1DASH(KPSI)
     1,CALR(KCHI,KPSI)      ,UDASH2(KPSI)          ,VDASH2(KPSI)
     2,WVALU(KPSI)          ,SHEAR(KPSI)           ,WELL(KPSI)
      COMMON      / SEISUU /NPSI      ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI     ,NTRIM
      COMMON      / PLAS /  TITLE(18),IPAGE
      COMMON      /IO /      IOIN      ,IOUT      ,IO1
      DOUBLE PRECISION     PSI       ,UDASH     ,V1DASH    ,VDASH2
     1,UDASH2    ,WVALU    ,SHEAR     ,WELL      ,PSIO      ,TEMPO
     2,CALR      ,PSI1     ,FNDT      ,TEMP1
     3,FUNCI     ,FNDI                                                      GGG
      DOUBLE CART1   ,CART2   ,PS1   ,PS2                                   GGG
      DATA        IDBL/2/
      CALL SECOND(TIMEF)
      CALL RANDM(IO1,UDASH,NPSI*IDBL,24,2)
      CALL RANDM(IO1,V1DASH,NPSI*IDBL,33,2)
      PSIO = PSI(NPSI-1)
      CALL CLEAR(0.D0,UDASH2,NPSI)
      CALL CLEAR(0.D0,VDASH2,NPSI)
      CALL CLEAR(0.D0,WVALU,NPSI)
      NPSIO = NPSIN-1
      DO 100 N=1,NPSIO
      CART1=0.5*(CALR(1,N)+CALR(1,N+1))                                     GGG
      CART2=0.5*(CALR(1,N+1)+CALR(1,N+2))                                   GGG
      CALL CALPSI(CART1,0.D0,PS1)                                           GGG
      CALL CALPSI(CART2,0.D0,PS2)                                           GGG
      TEMP1=PS2-PS1                                                         GGG
      UDASH2(N) = (UDASH(N+1)-UDASH(N))/TEMP1
      VDASH2(N) = (V1DASH(N+1)-V1DASH(N))/TEMP1
  100 WVALU(N) = (FUNCI(PSIO,PSI(N+1))*UDASH(N+1)-FUNCI(PSIO,PSI(N))
     1*UDASH(N))/TEMP1
      UDASH(1)=1.5*UDASH2(1)-0.5*UDASH2(2)                                  GGG
      DO 20 N=2,NPSIN                                                       GGG
   20 UDASH(N)=0.5*(UDASH2(N-1)+UDASH2(N))                                  GGG
      DO 22 N=1,NPSIN                                                       GGG
   22 UDASH2(N)=UDASH(N)                                                    GGG
      UDASH(1)=1.5*VDASH2(1)-0.5*VDASH2(2)                                  GGG
      DO 30 N=2,NPSIN                                                       GGG
   30 UDASH(N)=0.5*(VDASH2(N-1)+VDASH2(N))                                  GGG
      DO 32 N=1,NPSIN                                                       GGG
   32 VDASH2(N)=UDASH(N)                                                    GGG
      CALL RANDM(IO1,UDASH2,NPSI*IDBL,38,1)
      CALL RANDM(IO1,VDASH2,NPSI*IDBL,39,1)
      CALL RANDM(IO1,WVALU,NPSI*IDBL,40,1)
C * * SHEAR(SOL.)
      DO 110 N=1,NPSIN
  110 SHEAR(N) = -WVALU(N)/V1DASH(N)**3
      CALL RANDM(IO1,SHEAR,NPSI*IDBL,41,1)
C * * WELL(SOL.)
      PSIO = PSI(NPSI-1)
      DO 120 N=1,NPSIN
      TEMPO = (CALR(1,N)+CALR(1,N+1))*0.5
      CALL CALPSI(TEMPO,0.D0,PSI1)
  120 WELL(N) = FNDI(PSIO,PSI1)*SHEAR(N)-FNDT(PSIO,PSI1)*VDASH2(N)
     1/V1DASH(N)**3
      CALL RANDM(IO1,WELL,NPSI*IDBL,42,1)
      IF(NASK(IPR,1) .NE. -1) GO TO 160
      IPAGE = IPAGE+1
      IF(NASK(IPR,2) .EQ. -1) GO TO 140
      WRITE(IOUT,1000) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1010)
      DO 130 N=1,NPSIN
  130 WRITE(IOUT,1020) N,SHEAR(N),WELL(N)
      GO TO 160
  140 WRITE(IOUT,1030)
      DO 150 N=1,NPSIN
  150 WRITE(IOUT,1020) N,SHEAR(N),WELL(N),UDASH2(N),VDASH2(N),WVALU(N)
      CALL SECOND(TIMEL)
```

```
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1040) TIME
  160 CONTINUE
      RETURN
 1000 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3)
 1010 FORMAT(6H0 PSI ,32H    SHEAR(SOL.)        WELL(SOL.)     )
 1020 FORMAT(1X,I5,1X,1P6D16.8)
 1030 FORMAT(86H0 PSI       SHEAR(SOL.)        WELL(SOL.)           U-DASH2          GGG
     1    V-DASH2               W            )
 1040 FORMAT(36H0 STEP11 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

* SOURCE STATEMENT *

```
*DK STEP12        Solov'ev の不安定性係数
      SUBROUTINE STEP12(PSI,CALR,UD,Q1,Q2,Q3,Q4,Q5,V1D,UDD,VDD,SS,STL
     1,STC,KCHI,KPSI,IPR)
      DIMENSION  PSI(KPSI),UD(KPSI) ,Q1(KPSI) ,Q2(KPSI) ,Q3(KPSI)
     1,Q4(KPSI) ,Q5(KPSI) ,V1D(KPSI),UDD(KPSI),VDD(KPSI),CALR(KCHI,KPSI)
     2,STL(KPSI),STC(KPSI),SS(KPSI)
      COMMON / COM15 /      PS0,PS1,TY1,TY2,TY3,TY4,TY5,TY6,TY7
      COMMON / REPEAT / LMN,C ←―― Iが初期値の何倍かを表す.
      COMMON    / IO /       IOIN      ,IOUT      ,IO1
      COMMON    / PLAS /     TITLF(18),IPAGE
      COMMON    / RZ /       RMAX      ,RMIN      ,ZMAX      ,RCEN
      COMMON    / SEISUU / NPSI      ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI      ,NTRIM
      DOUBLE PRECISION    PSI       ,UD        ,Q1        ,Q2        ,Q3
     1,Q4        ,Q5        ,V1D       ,UDD       ,VDD       ,CALR      ,PSI0
     2,AI        ,AID       ,TD        ,TI        ,PSI1      ,TEMP1     ,TEMP2
     3,TEMP3     ,STL       ,STC       ,FUNCI     ,FNDI      ,FUNCT     ,FNDT
     4,SS
      DOUBLE PRECISION TEMP4,TEMP5,PS0,PS1,TY1,TY2,TY3,TY4,TY5,TY6,TY7
      DATA        IDBL/2/
      CALL SECOND(TIMEF)
      CALL RANDM(IO1,UD,NPSI*IDBL,24,2)
      CALL RANDM(IO1,Q1,NPSI*IDBL,28,2)
      CALL RANDM(IO1,Q2,NPSI*IDBL,29,2)
      CALL RANDM(IO1,Q3,NPSI*IDBL,30,2)
      CALL RANDM(IO1,Q4,NPSI*IDBL,31,2)
      CALL RANDM(IO1,Q5,NPSI*IDBL,32,2)
      CALL RANDM(IO1,V1D,NPSI*IDBL,33,2)
      CALL RANDM(IO1,UDD,NPSI*IDBL,38,2)
      CALL RANDM(IO1,VDD,NPSI*IDBL,39,2)
      CALL RANDM(IO1,SS,NPSI*IDBL,41,2)
C * * STABILITY CONSTANT
      PSI0 = PSI(NPSI-1)
      NPSI0 = NPSIN-1
      DO 100 N=1,NPSI0
      TEMP1 = (CALR(1,N)+CALR(1,N+1))*0.5
      CALL CALPSI(TEMP1,0.D0,PSI1)
      AI = FUNCI(PSI0,PSI1)
      AID = FNDI(PSI0,PSI1)
      TI = FUNCT(PSI0,PSI1)
      TD = FNDT(PSI0,PSI1)
      TEMP1 = 0.25*V1D(N)**6*SS(N)**2/(C**4)
      TEMP2 = UDD(N)*Q1(N)-VDD(N)*Q2(N)
      TEMP3=Q2(N)*Q3(N)-Q1(N)**2
      TEMP4=AI**2*(TEMP2+TD*TEMP3/C)/(C**5)
      TEMP5 = UD(N)*((AI*AID*Q1(N)+TD*Q3(N))/C-VDD(N))/(C**3)
      STL(N) = TEMP1-TD*(TEMP4+TEMP5)
      IF(N.NE.2) GO TO 200
      PS0=PSI0
      PS1=PSI1
      TY1=TEMP1
      TY2=TEMP2
      TY3=TEMP3
      TY4=UD(N)
      TY5=Q1(N)
      TY6=Q3(N)
      TY7=VDD(N)
  200 CONTINUE
      TEMP1 = SS(N)**2*(AI**2*(UD(N)*Q4(N)-V1D(N)**2)+Q4(N)*Q5(N)*C*C)
      TEMP2 = SS(N)*AI*(2.*TD+AI*AID*UD(N)/V1D(N)+AID/AI*Q5(N)/V1D(N))*C
      TEMP3 = TD*VDD(N)/V1D(N)**4*(AI**2*UD(N)+Q5(N)*C*C)*C
      TEMP1 = TEMP1+TEMP2+TEMP3-TD**2/V1D(N)**2*C*C
      TEMP2 = SS(N)**2*V1D(N)*Q4(N)+SS(N)*AI*(2.*VDD(N)/V1D(N)+AID/AI*C)
      TEMP3 = VDD(N)*((AI**2*UD(N)/C+Q5(N)*C)*VDD(N)/V1D(N)**2-TD)/V1D(N
     *)**3*C
      TEMP2 = 5./3.*TI*(TEMP2+TEMP3)
  100 STC(N) = TEMP1+TEMP2
      CALL RANDM(IO1,STL,NPSI*IDBL,43,1)
      CALL RANDM(IO1,STC,NPSI*IDBL,44,1)
      IF(NASK(IPR,1) .NE. -1) GO TO 120
      IPAGE = IPAGE+1
      WRITE(IOUT,1000) (TITLF(I),I=1,18),IPAGE
      WRITE(IOUT,1010)
      DO 110 N=1,NPSI0
  110 WRITE(IOUT,1020) N,STL(N),STC(N)
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1030) TIME
  120 CONTINUE
      RETURN
 1000 FORMAT(/// 19X,18A4,35X,4HPAGE,I3)
 1010 FORMAT(38H0              STABILITY          STABILITY   /38H  PSI        CON
     1ST.            CONST.      /6X,32H        (LOCAL)        (CONVECTIVE)   )
 1020 FORMAT(1X,I5,1X,1P6D16.8)                                                     GGG
 1030 FORMAT(36H0 STEP12 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

* SOURCE STATEMENT *

```
*DK STEP13      全断面積, 全電流 等の計算
      SUBROUTINE STEP13(AJ2,AA,KPSI,IPR)
      COMMON / COM17 / AJJ
      DIMENSION  AJ2(KPSI),AA(KPSI)
      COMMON    / IO /       IOIN      ,IOUT      ,IO1
      COMMON    / PLAS /     TITLF(18),IPAGE
      COMMON    / RZ /       RMAX      ,RMIN      ,ZMAX      ,RCEN
      COMMON    / SEISUU / NPSI      ,NCHI      ,NDIE      ,NPSIM     ,NINT
     1,NPSIN     ,NTRI      ,NTRIM
      DOUBLE PRECISION    AJ2       ,AA        ,AJJ       ,AREA      ,AEQ
     1,ASP       ,AJD
      DATA        PI/3.1415926535898/,IDBL/2/
      CALL SECOND(TIMEF)
      CALL RANDM(IO1,AA,NPSI*IDBL,22,2)
      CALL RANDM(IO1,AJ2,NPSI*IDBL,23,2)
      AJJ = 0.0
      AREA = 0.0
      DO 100 N=1,NPSIN-1                                                            GGG
      AJJ = AJJ+AJ2(N)
  100 AREA = AREA+AA(N)
      AEQ = DSQRT(AREA/PI)
      ASP = RCEN/AEQ
      AJD = AJJ/AREA
      IF(NASK(IPR,1) .NE. -1) GO TO 110
      IPAGE = IPAGE+1
```

```
      WRITE(IOUT,1000) (TITLE(I),I=1,18),IPAGE
      WRITE(IOUT,1010) AREA,AEQ,ASP,AJJ,AJD
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1020) TIME
  110 RETURN
 1000 FORMAT(////19X,18A4,35X,4HPAGE,I3)
 1010 FORMAT(
     1/30X,32HAREA OF PLASMA CROSS SECTION          ,1PD16.8
     2/30X,32HEQUIVALENT RADIUS                     ,1PD16.8
     3/30X,32HASPECT RATIO                          ,1PD16.8
     4/30X,32HTOTAL TOROIDAL CURRENT                ,1PD16.8
     5/30X,32HTOROIDAL CURRENT DENSITY              ,1PD16.8)
 1020 FORMAT(36H0 STEP13 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

* SOURCE STATEMENT *

```
*DK STEP14      等ψ曲線を 出力シート上に グラフ化して打出す
      SUBROUTINE STEP14(CALR,CALZ,VALU,XX,YY,ICAL,YYMX,I@,KCHI,KPSI,IPR)
      DOUBLE PRECISION    VALU
      DIMENSION  CALR(KCHI,KPSI)    ,CALZ(KCHI,KPSI)    ,VALU(KCHI,KPSI)
     1,XX(KCHI,KPSI)      ,YY(KCHI,KPSI)      ,ICAL(KPSI)
     2,I@(101,1),YYMX(KPSI)
      COMMON    / SEISUU / NPSI     ,NCHI     ,NDIE     ,NPSIM    ,NINT
     1,NPSIN    ,NTRI     ,NTRIM
      COMMON    / IO /    IOIN     ,IOUT     ,IO1
      COMMON    / PLAS /  TITLE(18),IPAGE
      COMMON    / PLOT /  NZONE    ,MESH     ,MAXR     ,MAXZ     ,INTER
     1,RMAXP    ,RMINP    ,ZMAXP    ,ZMINP
      DATA      IDBL/2/
      CALL SECOND(TIMEF)
      L=(NPSI-1)/10                                                     GGG
      IF(L.LT.1) L=1                                                    GGG
      CALL RANDM(IO1,VALU,NPSI*NCHI*IDBL,1,2)
      J=0                                                               GGG
      DO 100 N=L,NPSI-1,L                                               GGG
      J=J+1                                                             GGG
      DO 100 I=1,NCHI
  100 CALR(I,J)=VALU(I,N)
      CALL RANDM(IO1,VALU,NPSI*NCHI*IDBL,2,2)
      J=0                                                               GGG
      DO 110 N=L,NPSI-1,L                                               GGG
      J=J+1                                                             GGG
      DO 110 I=1,NCHI
  110 CALZ(I,J)=VALU(I,N)
      DO 140 N=1,J                                                      GGG
      DO 120 I=1,NCHI
      II = I
      IF(CALR(I,N) .EQ. 0.0) GO TO 130
  120 CONTINUE
      ICAL(N) = NCHI
      GO TO 140
  130 ICAL(N) = II-1
  140 CONTINUE
      DO 170 N=1,J                                                      GGG
      NN = ICAL(N)
      N1 = NN/2
      DO 150 I=1,N1
      II = NN-I+1
      TEMP = CALR(I,N)
      CALR(I,N) = CALR(II,N)
      CALR(II,N) = TEMP
      TEMP = CALZ(I,N)
      CALZ(I,N) = CALZ(II,N)
  150 CALZ(II,N) = TEMP
      NNN = NN+1
      IF(NNN .GT. NCHI) GO TO 170
      DO 160 I=NNN,NCHI
      CALR(I,N) = CALR(NN,N)
  160 CALZ(I,N) = CALZ(NN,N)
  170 CONTINUE
      IPAGE = IPAGE+1
      WRITE(IOUT,1000) (TITLE(I),I=1,18),IPAGE
      CALL JULY (J,1,NZONE,MESH,INTER,MAXZ,MAXR,ZMAXP,ZMINP,RMAXP       GGG
     1,RMINP,CALR,CALZ,XX,YY,I@,ICAL,YYMX,NPSI,1,NCHI)
      IF(NASK(IPR,2) .NE. -1) GO TO 180
      CALL SECOND(TIMEL)
      TIME = TIMEL-TIMEF
      WRITE(IOUT,1010) TIME
  180 RETURN
 1000 FORMAT(1H1,19X,18A4,35X,4HPAGE,I3/16H0 EQUI-PSI LINES)
 1010 FORMAT(36H0 STEP14 CENTRAL PROCESSOR USED TIME,F10.3,4HSEC.)
      END
```

* SOURCE STATEMENT *

```
      SUBROUTINE STEP15      初期値の何倍までプラズマ電流がとれるかを求める
      COMMON / COM1 / R0,AA,ASP,ES2,ES3
      COMMON / COM17 / AJJ
      COMMON / COM16 / BT0
      DOUBLE PRECISION        AJJ
      EXTERNAL TLS
      PAI=3.14159265358979
      AIP=AJJ/PAI*0.25E7
      @=2.*PAI*AA*AA*BT0/R0/AJJ
      CALL HASAMI(0.0,TLS,0.01,4.0,CRIT,DAM,0.002)
      AIPC=AIP*CRIT
      @C=@/CRIT
      WRITE(6,200)AIP,@,CRIT,AIPC,@C
  200 FORMAT(///10X,'SUBROUTINE STEP15 RESULT'/
     *//15X,'INITIAL CURRENT (COMP.)  IP(A)   ',F10.3
     * /15X,'INITIAL @       (COMP.)          ',F10.3
     * /15X,'CRITICAL RATIO                   ',F10.3
     * /15X,'CRITICAL CURRENT         IP(A)   ',F10.3
     * /15X,'CRITICAL @                       ',F10.3
     * /      )
      RETURN
      END
```

* SOURCE STATEMENT *

```
      FUNCTION TLS(X)        Solov'ev の局所不安定性係数
      COMMON / REPEAT / LMN,COEFX
      COMMON / COM15 /     PS0,PS1,TY1,TY2,TY3,TY4,TY5,TY6,TY7
      DOUBLE PRECISION TEMP4,TEMP5,PS0,PS1,TY1,TY2,TY3,TY4,TY5,TY6,TY7
      DOUBLE PRECISION AID,AI,TD,FUNCI,FNDI,FNDT
      COEFX=X  <-------------------- プラズマ電流の初期値との比
      AID=FNDI(PS0,PS1)
      AI=FUNCI(PS0,PS1)
      TD=FNDT(PS0,PS1)
      TEMP4=AI**2*(TY2*TD*TY3/X)/(X**5)
      TEMP5=TY4*((AI*AID*TY5+TD*TY6)/X-TY7)/(X**3)
      TLS=TY1-TD*(TEMP4+TEMP5)
      RETURN
      END
```

* SOURCE STATEMENT *

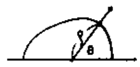
等ψ点を求める

```
      SUBROUTINE DIST(CALR1,THET,CALR3,CALZ3,PSI1,PSI0,DELR,LIM,EPS,
     *    IPR,N)
      DOUBLE CALR1,THET,CALR3,CALZ3,PSI0,PSI1,DELR
      COMMON / RZ / RMAX,RMIN,ZMAX,RCEN
      COMMON / HAS / TT,RZZZZ
      COMMON / COM1 / RD,AA,ASP,ES2,ES3
      EXTERNAL PPP
      HALFPI=1.5707963268
      RZZZZ=RCEN
      PSS=PSI0
      RAT=0.01*AA
      RMM= 5.0*AA
      TT=THET
      CALL HASAMI(PSS,PPP,RAT,RMM,RXX,DAM,EPS)
      IF(RXX.EQ.0.) GO TO 100
      CALR3=RCEN+RXX*COS(TT)
      CALZ3=RXX*SIN(TT)
      IF(DABS(THET-HALFPI).LE.1.E-5) GO TO 200
      RETURN
  200 RRRR=CALR3
      CALR3=CALZ3
      CALZ3=RRRR
      RETURN
  100 WRITE(6,1002)N,PSI0,THET
      RETURN
 1002 FORMAT(2X,'/DIST/ (',I2,') NO CALC.  PSI=',1PD12.5,'   ANGLE=',
     * 1PD12.5)
      END
```

* SOURCE STATEMENT *

角度θ(=T)が与えられたとき、ρの関数としてψを求める。

```
      FUNCTION PPP(R)
      DOUBLE RB,ZB,PHI
      COMMON / HAS / T,RX
      RB=RX+R*COS(T)
      ZB=R*SIN(T)
      CALL CALPSI(RB,ZB,PHI)
      PPP=PHI
      RETURN
      END
```



* SOURCE STATEMENT *

*DK INTEG　　X, J を求める際の数値積分

```
      SUBROUTINE INTEG(TEMP1,I,N,CALR,CALZ,RINT,ZINT,THET,CALF,PSI0,PSI1
     1,EPS,LIM,IPR,KCHI,KPSI,KIND)
C     INTEGRATION ROUTINE
      DIMENSION  CALR(KCHI,KPSI)     ,CALZ(KCHI,KPSI)     ,RINT(KIND)
     1,ZINT(KIND)            ,THET(KIND)           ,CALF(KIND)
      COMMON       / IO   /   IOIN      ,IOUT      ,IO1
      COMMON       / RZ /     RMAX      ,RMIN      ,ZMAX      ,RCEN
      DOUBLE        CALR      ,CALZ      ,RINT      ,ZINT      ,THET      ,CALF
     1,PSI0      ,PSI1      ,DTHET     ,CALR1     ,CALR3     ,CALZ3     ,TEMPO
     2,DTHETA    ,TEMP1     ,TEMP2     ,DELR
      DOUBLE PRECISION      FUNCI     ,FUNCP     ,FUNCB     ,FUNCF
      DATA          PI/3.14159265358 98/,HALFPI/1.57079632679491/
      DELR = DABS(CALR(I,N)-CALR(I,N-1))                                  GGG
      IF(DELR .LE. EPS) DELR = (CALR(1,2)-RCEN)                           GGG
      RINT(1) = CALR(I,N)
      RINT(5) = CALR(I-1,N)
      ZINT(1) = CALZ(I,N)
      ZINT(5) = CALZ(I-1,N)
      DO 107 L=1,5,4
      IF(RINT(L)-RCEN) 100,103,104
  100 IF(ZINT(L)) 101,101,102
  101 THET(L) = PI
      GO TO 107
  102 THET(L) = PI-DATAN(ZINT(L)/(RCEN-RINT(L)))
      GO TO 107
  103 THET(L) = HALFPI
      GO TO 107
  104 IF(ZINT(L)) 105,105,106
  105 THET(L) = 0.0
      GO TO 107
  106 THET(L) = DATAN(ZINT(L)/(RINT(L)-RCEN))
  107 CONTINUE
      DTHET = THET(1)-THET(5)
      DTHETA = DTHET*0.25
      IF(THET(1)-HALFPI) 108,108,109
  108 INDA = 2
      GO TO 110
  109 INDA = 1
  110 IF(THET(1) .GT. HALFPI .AND. THET(5) .LT. HALFPI) INDA = 3
      DO 117 L=2,4
      IND = 0
      THET(L) = THET(L-1)-DTHETA
      GO TO (111,112,113),INDA
  111 CALR1 = CALR(I-1,N-1)
      GO TO 114
  112 CALR1 = CALR(I,N-1)
      GO TO 114
  113 CALR1 = RCEN
  114 CONTINUE
      IF(DABS(THET(L)-HALFPI) .LE. 0.01*EPS) IND = 1
      IF(IND .EQ. 0) GO TO 115
      CALR1 = CALZ(I,N-1)
  115 CALL DIST(CALR1,THET(L),CALR3,CALZ3,PSI1,PSI0,DELR,LIM,     EPS
     1,IPR,2)
      IF(IND .EQ. 1) GO TO 116
      RINT(L) = CALR3
      ZINT(L) = CALZ3
      GO TO 117
  116 RINT(L) = CALZ3
      ZINT(L) = CALR3
  117 CONTINUE
      TEMPO = 0.0
      DO 118 L=1,5
  118 CALF(L) = FUNCF(RINT(L),ZINT(L))
      DO 119 L=1,4
  119 TEMPO = TEMPO+(CALF(L)+CALF(L+1))*DSQRT((RINT(L+1)-RINT(L))**2
     1+(ZINT(L+1)-ZINT(L))**2)
      LO1 = 5
  120 LO2 = LO1*2-1
      DO 121 L=1,LO2,2
      LL = LO2-L+1
      RINT(LL) = RINT(LO1)
      ZINT(LL) = ZINT(LO1)
      CALF(LL) = CALF(LO1)
      THET(LL) = THET(LO1)
  121 LO1 = LO1-1
      DTHETA = DTHET/(LO2-1)
      DO 128 L=2,LO2,2
      THET(L) = THET(L-1)-DTHETA
      GO TO( 122,123,124),INDA
  122 CALR1 = CALR(I-1,N-1)
```

```
      GO TO 125
  123 CALR1 = CALR(I,N-1)
      GO TO 125
  124 CALR1 = RCFN
  125 CONTINUE
      IF(DABS(THET(L)-HALFPI) .GT. 0.01*EPS) GO TO 126
      CALR1 = CALZ(I,N-1)
  126 CONTINUE
      CALL DIST(CALR1,THET(L),CALR3,CALZ3,PSI1,PSI0,DELR,LIM,     EPS
     1,IPR,3)
      IF(DABS(THET(L)-HALFPI) .LE. 0.01*EPS) GO TO 127
      RINT(L) = CALR3
      ZINT(L) = CALZ3
      GO TO 128
  127 RINT(L) = CALZ3
      ZINT(L) = CALR3
  128 CALF(L) = FUNCF(RINT(L),ZINT(L))
      LO3 = LO2-1
      TEMP1 = 0.0
      DO 129 L=1,LO3
  129 TEMP1 = TEMP1+(CALF(L)+CALF(L+1))*DSQRT((RINT(L+1)-RINT(L))**2
     1+(ZINT(L+1)-ZINT(L))**2)
      TEMP2 = DABS(TEMPO-TEMP1)
      IF(TEMP2/EPS-1.0) 132,132,130
  130 LO1 = LO2
      IF(LO1*2 .GT. KIND) GO TO 131
      TEMPO = TEMP1
      GO TO 120
  131 IF(NASK(IPR,2) .EQ. -1) WRITE(IOUT,1000) N,I
  132 TEMP1 = TEMP1*0.5
      RETURN
 1000 FORMAT(50H /INTEG/  THE ITERATION COUNT LIMIT EXCEED.   PSI=,I3
     1,7H ANGLE=,I3)
      END
```

* SOURCE STATEMENT *

```
*DK DIFF          ψ を求めるための刻幅ψ値を求める.
      SUBROUTINE DIFF(CALR0,CALZ0,CALR3,CALZ3,PSI0,PSI1,P0,DELZ,I,N,EPS
     1,LIMT,IPR)
      COMMON     / IO /     IOIN      ,IOUT      ,IO1
      DOUBLE      CALR0     ,CALR1     ,CALR2     ,CALR3     ,CALZ0     ,CALZ1
     1,CALZ2     ,CALZ3     ,ZETA1     ,ZETA2     ,ZETA3     ,PSI0      ,PSI1
     2,PSI2      ,PSI3      ,P0        ,DELZ
      ICNT = 1
      CALR1 = CALR0
      CALZ1 = CALZ0
      ZETA1 = PSI0-PSI1
  100 CALZ2 = CALZ1+DELZ
      CALR2 = P0*(CALZ2-CALZ0)+CALR0
      CALL CALPSI(CALR2,CALZ2,PSI2)
      ZETA2 = PSI0-PSI2
      IF(ZETA2 .EQ. 0.0) GO TO 101
      IF(DSIGN(1.0D0,ZETA1) .NE. DSIGN(1.0D0,ZETA2)) GO TO 101
      ZETA1 = ZETA2
      CALZ1 = CALZ2
      GO TO 100
  101 CALZ3 = (CALZ2-CALZ1)/(DABS(ZETA1)+DABS(ZETA2))*DABS(ZETA1)+CALZ1
      CALR3 = P0*(CALZ3-CALZ0)+CALR0
      CALL CALPSI(CALR3,CALZ3,PSI3)
      ZETA3 = PSI0-PSI3
      IF(DABS(ZETA3/PSI0) .LE. EPS) GO TO 104
      IF(ICNT .GE. LIMT) GO TO 103
      ICNT = ICNT+1
      IF(DSIGN(1.D0,ZETA1) .NE. DSIGN(1.D0,ZETA3)) GO TO 102
      ZETA1 = ZETA3
      CALZ1 = CALZ3
      GO TO 101
  102 ZETA2 = ZETA3
      CALZ2 = CALZ3
      GO TO 101
  103 IF(NASK(IPR,2) .EQ. -1) WRITE(IOUT,1000) N,I
  104 RETURN
 1000 FORMAT(48H /DIFF/  THE ITERATION COUNT LIMIT EXCEED.  PSI=,I3,3X
     1,6HANGLE=,I3)
      END
```

* SOURCE STATEMENT *

```
*DK NASK
      FUNCTION NASK(IBAS,ICNT)     プリント・オプションの判定            NASK   2
      DIMENSION  IND(3)
      IBASE=IBAS+1                                                      NASK   4
      DO 100 I=1,3                                                      NASK   5
  100 IND(I)=0                                                          NASK   6
      NASK=0                                                            NASK   7
      GO TO(101,102,103,104,105,106,107,108),IBASE                        GGG
  101 GO TO 109                                                         NASK   9
  102 IND(1)=1                                                          NASK  10
      GO TO 109                                                         NASK  11
  103 IND(2)=1                                                          NASK  12
      GO TO 109                                                         NASK  13
  104 IND(1)=1                                                          NASK  14
      IND(2)=1                                                          NASK  15
      GO TO 109                                                         NASK  16
  105 IND(3)=1                                                          NASK  17
      GO TO 109                                                         NASK  18
  106 IND(1)=1                                                          NASK  19
      IND(3)=1                                                          NASK  20
      GO TO 109                                                         NASK  21
  107 IND(2)=1                                                          NASK  22
      IND(3)=1                                                          NASK  23
      GO TO 109                                                         NASK  24
  108 IND(1)=1                                                          NASK  25
      IND(2)=1                                                          NASK  26
      IND(3)=1                                                          NASK  27
  109 IF(IND(ICNT).EQ.1) NASK=-1                                        NASK  28
      RETURN                                                            NASK  29
      END                                                               NASK  30
```

* SOURCE STATEMENT *

```
*DK CLEAR
      SUBROUTINE CLEAR(DATA,ARRAY,ICON)
      DIMENSION  ARRAY(ICON)
      DOUBLE      DATA      ,ARRAY          大きさ ICON の ARRAY すべてに DATA
      DO 100 I=1,ICON                       を入れる.
  100 ARRAY(I) = DATA                       ゼロ・クリアー等に使用する.
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK ERROR
      SUBROUTINE ERROR(NUM,HOL)
C * * ERROR ROUTINE
      DIMENSION HOL(1)
      COMMON     / IO /     IOIN     ,IOUT     ,IO1
      DIMENSION  CDC(2)
      DATA        IWORD/10/
      ICAL = NUM/IWORD
      IF(MOD(NUM,IWORD) .NE. 0) ICAL = ICAL+1
 1001 FORMAT(1H0)
      ENCODE(IWORD,1000,CDC) ICAL,IWORD
      WRITE(IOUT,1001)
 1000 FORMAT(1H(,I2,1HA/I2,1H))
      WRITE(IOUT,CDC) (HOL(I),I=1,ICAL)
      RETURN
      END

      SUBROUTINE RANDM(ITAPE,ARRAY,LENGTH,IPOS,IND)
C     I/O DEVICE SERVICE ROUTINE
C     ITAPE          UNIT NUMBER
C     ARRAY(LENGTH)  I/O DATA
C     IPOS           POSITION NUMBER
C     IND = 0        INITIAL SET
C     IND = 1        WRITE
C     IND = 2        READ
C
      COMMON/SEISUU/ NPSI, NCHI, NDIE, NPSIM, NINT
     *, NPSIN, NTRI, NTRIM
      DIMENSION  ARRAY(LENGTH)      ,IDSET(100)         ,IWD(11)
      DATA        IDSET/100*0/      ,MAXIO/0/
      IF(IND .EQ. 0) GO TO 103
      IF(IDSET(IPOS) .GT. 0) GO TO 102
      K=IPOS
   10 K=K-1
      IF(IDSET(K).GT.0) GO TO 20
      IF(K.GT.0) GO TO 10
      WRITE(6,2050)
 2050 FORMAT(/10X,'ERROR IN RANDM'/)
      STOP 9630
   20 IDSET(IPOS)=IDSET(K)
  102 J=IDSET(IPOS)                                             GGG
      IF(IND.EQ.2) GO TO 104                                    GGG
      DWRITE(ITAPE,J)(ARRAY(I),I=1,LENGTH)                      GGG
      IDSET(IPOS+1)=J
      RETURN                                                    GGG
  104 DREAD(ITAPE,J)(ARRAY(I),I=1,LENGTH)                       GGG
      RETURN
  103 CALL DFILE(ITAPE,256,J)
      IDSET(1)=1
      RETURN
      END
```

* SOURCE STATEMENT *

```
*DK OUTPTM
      SUBROUTINE OUTPTM(ARRAY,J1,I1,NCLM,NLOW,NMAX,J2,I2,ICAL,HOL,IPAGE)
C * * FLOATING ARRAY DATA PRINT SERVICE ROUTINE.
C     ARRAY(J2,I2)   OUTPUT DATA
C     J1
C     I1
C     NCLM           BIAS FOR COLUMNS
C     NLOW           BIAS FOR LOWS
C     NMAX           MAXIMUM NUMBER OF COLUMN
C     ICAL           NUMBER OF CHARACTER OF TITLE
C     HOL            TITLE
C     IPAGE          PAGE
C
      COMMON     / IO /     IOIN     ,IOUT     ,IO1
      COMMON     / PLAS /   TITLE(18),LPAGE                         GGG
      DOUBLE      ARRAY
      DIMENSION  ARRAY(J2,I2)      ,HOL(1)    ,CDC(3)
      DATA        CDC(1) /4H(/1X/   ,PAGE1/4HPAGE/      ,IWORD/ 4/  GGG
      ICAR = ICAL/IWORD
      IF(MOD(ICAL,IWORD) .NE. 0) ICAR = ICAR+1
C     ENCODE(IWORD,1000,CDC(2)) ICAR
C     ENCODE(IWORD,1001,CDC(3)) IWORD
      ICOUNT = 60
      IF(NMAX) 100,110,105
  100 II = 1
  101 JJ = 1
      III = II+NMAX-1
      IF(III .GT. I1) III = I1
      LL = II+NCLM
      LLL = III+NCLM
  102 JJJ = JJ+59
      IF(JJJ .GT. J1) JJJ = J1
      IF(ICOUNT+JJJ-JJ+3 .LE. 60) GO TO 103
      IPAGE = IPAGE+1
C     CALL INTBCD(IPAGE,PAGE2,L)
C     WRITE(IOUT,1002) TITLE,PAGE1,PAGE2
      WRITE(IOUT,1002) TITLE,PAGE1,IPAGE                           GGG
      WRITE(IOUT,1006)(HOL(I),I=1,ICAR)                            GGG
      ICOUNT = 3
  103 CONTINUE
      WRITE(IOUT,1003)
      WRITE(IOUT,1004) (L,L=LL,LLL)
      DO 104 J=JJ,JJJ
      JG = J+NLOW
  104 WRITE(IOUT,1005) JG,(ARRAY(J,I),I=II,III)
      ICOUNT = ICOUNT+JJJ-JJ+3
      JJ = JJJ+1
      IF(JJJ .LT. J1) GO TO 102
      II = III+1
      IF(III .LT. I1) GO TO 101
      GO TO 110
  105 JJ = 1
  106 II = 1
      JJJ = JJ+NMAX-1
      IF(JJJ .GT. J1) JJJ = J1
      LL = JJ+NCLM
      LLL = JJJ+NCLM
  107 III = II+59
      IF(III .GT. I1) III = I1
      IF(ICOUNT+III-II+3 .LE. 60) GO TO 108
      IPAGE = IPAGE+1
C     CALL INTBCD(IPAGE,PAGE2,L)
C     WRITE(IOUT,1002) TITLE,PAGE1,PAGE2
      WRITE(IOUT,1002) TITLE,PAGE1,IPAGE                           GGG
      WRITE(IOUT,1006)(HOL(I),I=1,ICAR)                            GGG
      ICOUNT = 3
  108 CONTINUE
      WRITE(IOUT,1003)
      WRITE(IOUT,1004) (L,L=LL,LLL)
      DO 109 I=II,III
      IG = I+NLOW
  109 WRITE(IOUT,1005) IG,(ARRAY(J,I),J=JJ,JJJ)
```

```
          ICOUNT = ICOUNT+III-II+3
          II = III+1
          IF(III .LT. I1) GO TO 107
          JJ = JJJ+1
          IF(JJJ .LT. J1) GO TO 106
      110 CONTINUE
          RETURN
     1000 FORMAT(1H,,I2)
     1001 FORMAT(1HA,I2,1H))
    C1002 FORMAT(1H1,19X,18A4,35X,A4,1X,A4)
     1002 FORMAT(1H1,19X,18A4,35X,A4,1X,I4)                          GGG
     1003 FORMAT(1H )
     1004 FORMAT(6X,8I16)
     1005 FORMAT(1X,I4,2X,1P8D16.8)                                  GGG
     1006 FORMAT(/1X,20A4)                                           GGG
          END
```

* SOURCE STATEMENT *

```
   *DK OUTP
          SUBROUTINE OUTP(ARRAY,I1,J1,I2,J2,ICAL,HOL)
    C * * FLOATING ARRAY DATA PRINT ROUTINE.
          DIMENSION ARRAY(I2,J2)        ,HOL(1)    ,CDC(3)
          COMMON     / IO /     IOIN      ,IOUT      ,IO1
          COMMON     / PLAS /  TITLE(18),IPAGE
          DOUBLE      ARRAY
          DATA        CDC(1)/4H(/1X/      ,PAGE1/4HPAGE/       ,IWORD/ 4/  GGG
         1,NMAX/8/   ,IPSI/3HPSI/
          ICAR = ICAL/IWORD
          IF(MOD(ICAL,IWORD) .NE. 0) ICAR = ICAR+1
    C     ENCODE(IWORD,1000,CDC(2)) ICAR
    C     ENCODE(IWORD,1001,CDC(3)) IWORD
          ICOUNT = 60
          JJ = 1
      100 II = 1
          JJJ = JJ+NMAX-1
          IF(JJJ .GT. J1) JJJ = J1
      101 III = II+59
          IF(III .GT. I1) III = I1
           IF(ICOUNT+III-II+3 .LE. 60) GO TO 102
          IPAGE = IPAGE+1
    C     CALL INTBCD(IPAGE,PAGE2,L)
    C     WRITE(IOUT,1002) (TITLE(I),I=1,18),PAGE1,PAGE2
          WRITE(IOUT,1002) (TITLE(I),I=1,18),PAGE1,IPAGE              GGG
          WRITE(IOUT,1005) (HOL(I),I=1,ICAR)                          GGG
          ICOUNT = 3
      102 CONTINUE
          WRITE(IOUT,1003) (IPSI,L,L=JJ,JJJ)
          DO 103 I=II,III
      103 WRITE(IOUT,1004) I,(ARRAY(I,J),J=JJ,JJJ)
          ICOUNT = ICOUNT+III-II+3
          II = III+1
          IF(III .LT. I1) GO TO 101
          JJ = JJJ+1
          IF(JJJ .LT. J1) GO TO 100
          RETURN
     1000 FORMAT(1H,,I2)
     1001 FORMAT(1HA,I2,1H))
    C1002 FORMAT(1H1,19X,18A4,35X,A4,1X,A4)
     1002 FORMAT(1H1,19X,18A4,35X,A4,1X,I4)                           GGG
     1003 FORMAT(6HONODE ,8(6X,A3,I2,5X))
     1004 FORMAT(1X,I4,2X,1P8D16.8)                                   GGG
     1005 FORMAT(/1X,20A4)                                            GGG
          END
```

* SOURCE STATEMENT *

```
          SUBROUTINE SECOND(TIME)
          TIME=0.
          RETURN
          END
```

* SOURCE STATEMENT *

```
   *DK JULY <------ 簡易プロッター・サブルーチン
          SUBROUTINE JULY(II,KK,IZO,IK,IP,IS,IR,YMAX,YMIN,XMAX,XMIN,X,Y,    THIRD  7
         *   XX,YY,IQ,JYNX,YYMAX,JI,JK,JN)                                   JULY   3
          DIMENSION YPR(6),XPR(50)                                          THIRD  8
          IK5=IK*IZO                                                        THIRD  9
          IK6=IK5+1                                                         THIRD 10
          CALL GRAD(II,KK,IZO,IK,IK5,IK6,IS,IR,YMAX,YMIN,XMAX,XMIN,         THIRD 11
         *   FACX,FACY,YPR,XPR,X,Y,IQ,JYNX,JI,JK,JN)                         JULY   8
          XMIN=XPR(1)                                                       JULY   9
          YMIN=YPR(1)                                                       JULY  10
          CALL FAC(II,KK,FACX,FACY,XMIN,YMIN,X,Y,XX,YY,JYNX,YYMAX,JI,JK,JN) JULY  11
          IF(IP.EQ.1)GO TO 10                                               JULY  12
          CALL AA(KK,II,IK5,IK6,YYMAX,XX,YY,IQ,JYNX,JN,JK,JI)               JULY  13
          GO TO 20                                                          JULY  14
       10 CALL BB(IK5,IK6,KK,II,XX,YY,IQ,JYNX,JN,JK,JI)                     JULY  15
       20 CALL PRINT(IK,IK6,YPR,XPR,IQ)                                     JULY  16
          RETURN                                                            JULY  17
          END                                                               JULY  18
```

* SOURCE STATEMENT *

```
   *DK GRAD
          SUBROUTINE GRAD(II,KK,IZO,IK,IK5,IK6,IS,IR,YMAX,YMIN,XMAX,XMIN,   THIRD 12
         *    FACX,FACY,YPR,XPR,X,Y,IQ,JYNX,JI,JK,JN)                        GRAD   3
          DIMENSION X(JN,JI),Y(JN,JK,JI),IQ(101,1),JYNX(1)                   GRAD   4
          DIMENSION YPR(6),XPR(50)                                          THIRD 13
          DATA IPOINT/1H./,    IBL/1H /                                     GRAD   6
          DO 10  I=1,IK6                                                    GRAD   7
          DO 10 J=1,101                                                     GRAD   8
       10 IQ(J,I)=IBL                                                       GRAD   9
          DO 1 I=1,IK6,IK                                                   GRAD  10
          DO 2 J=1,101                                                      GRAD  11
        2 IQ(J,I)=IPOINT                                                    GRAD  12
          IF(I.GE.IK6)GO TO 4                                               GRAD  13
          I1=I+1                                                            GRAD  14
          I2=I+IK-1                                                         GRAD  15
          DO 3 I3=I1,I2                                                     GRAD  16
          DO 3 J=1,101,20                                                   GRAD  17
        3 IQ(J,I3)=IPOINT                                                   GRAD  18
        1 CONTINUE                                                          GRAD  19
        4 IF(IS.EQ.1)GO TO 5                                                GRAD  20
          YMAX=-1.0E+20                                                     GRAD  21
          YMIN=1.0E+20                                                      GRAD  22
          DO 6 I=1,II                                                       GRAD  23
          NN=JYNX(I)                                                        GRAD  24
          DO 6 K=1,KK                                                       GRAD  25
          DO 6 N=1,NN                                                       GRAD  26
          IF(YMAX.LT.Y(N,K,I))YMAX=Y(N,K,I)                                 GRAD  27
          IF(YMIN.GT.Y(N,K,I))YMIN=Y(N,K,I)                                 GRAD  28
        6 CONTINUE                                                          GRAD  29
        5 IF(IR.EQ.1)GO TO 7                                                GRAD  30
```

```
      XMAX=-1.0E+20                                                     GRAD  31
      XMIN=1.0E+20                                                      GRAD  32
      DO 8 I=1,II                                                       GRAD  33
      NN=JYNX(I)                                                        GRAD  34
      DO 8 N=1,NN                                                       GRAD  35
      IF(XMAX.LT.X(N,I))XMAX=X(N,I)                                     GRAD  36
      IF(XMIN.GT.X(N,I))XMIN=X(N,I)                                     GRAD  37
    8 CONTINUE                                                          GRAD  38
    7 YH=(YMAX-YMIN)/5.                                                 GRAD  39
      XH=(XMAX-XMIN)/IZO                                                THIRD 14
      YPR(1)=YMIN                                                       GRAD  41
      YPR(6)=YMAX                                                       GRAD  42
      XPR(1)=XMIN                                                       GRAD  43
      IIZO=IZO+1                                                        THIRD 15
      XPR(IIZO)=XMAX                                                    THIRD 16
      DO 9 J=2,5                                                        GRAD  45
      XJ=J-1                                                            GRAD  46
      YPR(J)=YMIN+XJ*YH                                                 GRAD  47
    9 CONTINUE                                                          GRAD  49
      DO 11 J=2,IZO                                                     THIRD 17
      XJ=J-1                                                            THIRD 18
      XPR(J)=XMIN+XJ*XH                                                 THIRD 19
   11 CONTINUE                                                          THIRD 20
      FACY=(YMAX-YMIN)/100.                                             GRAD  50
      A5=FLOAT(IK5)                                                     GRAD  51
      FACX=(XMAX-XMIN)/A5                                               GRAD  52
      RETURN                                                            GRAD  53
      END                                                               GRAD  54
```

* SOURCE STATEMENT *

```
*DK FAC
      SUBROUTINE FAC(II,KK,   FACX,FACY,XMIN,YMIN,X,Y,XX,YY,JYNX,YYMAX, FAC    2
     *   JI,JK,JN)                                                      FAC    3
      DIMENSION X(JN,JI),Y(JN,JK,JI),XX(JN,JI),YY(JN,JK,JI)             FAC    4
     *   ,YYMAX(JK,JI) ,JYNX(1)                                         FAC    5
      DO 1 I=1,II                                                       FAC    6
      NN=JYNX(I)                                                        FAC    7
      DO 1 K=1,KK                                                       FAC    8
      DO 1 N=1,NN                                                       FAC    9
      XX(N,I)=(X(N,I)-XMIN)/FACX                                        FAC   10
      YY(N,K,I)=(Y(N,K,I)-YMIN)/FACY                                    FAC   11
    1 CONTINUE                                                          FAC   12
      DO 2 I=1,II                                                       FAC   13
      NN=JYNX(I)                                                        FAC   14
      DO 2 K=1,KK                                                       FAC   15
      YYMAX(K,I)=YY(1,K,I)                                              FAC   16
      DO 2 N=1,NN                                                       FAC   17
      IF(YYMAX(K,I).LT.YY(N,K,I))YYMAX(K,I)=YY(N,K,I)                   FAC   18
    2 CONTINUE                                                          FAC   19
      RETURN                                                            FAC   20
      END                                                               FAC   21
```

* SOURCE STATEMENT *

```
*DK BB
      SUBROUTINE BB(IK5,IK6,   KK,II,XX,YY,I@,JYNX,JN,JK,JI)            BB     2
      DIMENSION XX(JN,JI),YY(JN,JK,JI),I@(101,1)   ,JYNX(1)             BB     3
      DIMENSION IFOR(36)                                                FIRST136
      DATA(IFOR(I),I=1,36)/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0      FIRST137
     *  ,1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1HL                FIRST138
     *  ,1HM,1HN,1HO,1HP,1H@,1HR,1HS,1HT,1HU,1HV,1HW,1HX                FIRST139
     *  ,1HY,1HZ/                                                       FIRST140
      DO 1 I=1,II                                                       BB     6
      NN=JYNX(I)                                                        BB     7
      DO 1 K=1,KK                                                       BB     8
      DO 1 N=1,NN                                                       BB     9
      A5=FLOAT(IK5)                                                     FIRST141
      IF(XX(N,I).LT.0..OR.XX(N,I).GT.A5)GO TO 1                         BB    11
      IF(YY(N,K,I).LT.0..OR.YY(N,K,I).GT.100.)GO TO 1                   BB    12
      IY=IFIX(YY(N,K,I)+1.0)                                            FIRST142
      IX=IFIX(XX(N,I)+1.0)                                              FIRST143
      I@(IY,IX)=IFOR(I)                                                 BB    15
    1 CONTINUE                                                          BB    16
      RETURN                                                            BB    17
      END                                                               BB    18
```

* SOURCE STATEMENT *

```
*DK CC
      SUBROUTINE CC(X1,X2,X3,Y1,Y2,Y3,A,B,C)                            FIRST144
      IF(X1.EQ.X2)GO TO 1                                               FIRST145
      IF(X2.EQ.X3)GO TO 2                                               FIRST146
      IF(X3.EQ.X1)GO TO 2                                               FIRST147
      A=((Y1-Y2)/(X1-X2)-(Y2-Y3)/(X2-X3))/(X1-X3)                       FIRST148
      B=(Y1-Y2)/(X1-X2)-A*(X1+X2)                                       FIRST149
      C=Y1-A*X1**2-B*X1                                                 FIRST150
      GO TO 4                                                           FIRST151
    1 A=(Y1-Y3)/(X1-X3)                                                 FIRST152
      B=Y1-A*X1                                                         FIRST153
      C=0.0                                                             FIRST154
      GO TO 4                                                           FIRST155
    2 A=(Y1-Y2)/(X1-X2)                                                 FIRST156
      B=Y1-A*X1                                                         FIRST157
      C=0.0                                                             FIRST158
    4 RETURN                                                            FIRST159
      END                                                               FIRST160
```

* SOURCE STATEMENT *

```
*DK AA
      SUBROUTINE AA(   KK,II,IK5,IK6,YYMAX,XX,YY,I@,JYNX,JN,JK,JI)      FIRST161
      DIMENSION XX(JN,JI),YY(JN,JK,JI),I@(101,1),IFOR(36)               FIRST162
     *    ,YYMAX(JK,JI),JYNX(1)                                         FIRST163
      DATA(IFOR(I),I=1,36)/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0      FIRST164
     *  ,1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1HL                FIRST165
     *  ,1HM,1HN,1HO,1HP,1H@,1HR,1HS,1HT,1HU,1HV,1HW,1HX                FIRST166
     *  ,1HY,1HZ/                                                       FIRST167
      A5=FLOAT(IK5)                                                     FIRST168
      DO 15 I=1,II                                                      FIRST169
      NN=JYNX(I)                                                        FIRST170
      NN1=NN-1                                                          FIRST171
      NN2=NN-2                                                          FIRST172
      DO 16 K=1,KK                                                      FIRST173
      M1=0                                                              FIRST174
      DO 21 N=1,NN                                                      FIRST175
      IF(XX(N,I).LT.0) GO TO 21                                         FIRST176
      IF((YY(N,K,I).GE.0.0).AND.(YY(N,K,I).LE.100.0)) GO TO 26          FIRST177
   21 CONTINUE                                                          FIRST178
      GO TO 16                                                          FIRST179
   26 L1=N                                                              FIRST180
      IF(L1.EQ.NN)GO TO 16                                              FIRST181
      DO 3 N=L1,NN
      IF(YY(N,K,I).GT.100.0) GO TO 4
    3 CONTINUE
```

```
      GO TO 46
    4 CONTINUE
C                                          TYPE=1,2,3,4                    FIRST183
      DO 31 N=L1,NN                                                        FIRST184
      IF(XX(N,I).GT.A5)GO TO 11                                            FIRST185
      IF(YY(N,K,I).GT.100.0) GO TO 36                                      FIRST186
   31 CONTINUE                                                             FIRST187
      GO TO 16                                                             SECOND46
C                                          TYPE=2                          FIRST189
   11 N1=N                                                                 FIRST190
      GO TO 40                                                             FIRST191
C                                          TYPE=1                          FIRST192
   13 N1=N                                                                 FIRST193
      GO TO 40                                                             FIRST194
C                                          TYPE=1,3,4                      FIRST195
   36 IF(N.EQ.NN) GO TO 13                                                 FIRST196
      M1=N                                                                 FIRST197
      DO 37 N=M1,NN                                                        FIRST198
      IF(XX(N,I).GT.A5) GO TO 39                                           FIRST199
      IF(YY(N,K,I).LT.100.0) GO TO 38                                      FIRST200
   37 CONTINUE                                                             FIRST201
C                                          TYPE=1                          FIRST202
   39 IF(YY(N,K,I).LT.100.0)GO TO 500                                      FIRST203
      N1=M1                                                                FIRST204
      M1=0                                                                 FIRST205
      GO TO 40                                                             FIRST206
  500 K1=N                                                                 FIRST207
      N1=N                                                                 FIRST208
      GO TO 40                                                             FIRST209
C                                          TYPE=3,4                        FIRST210
   38 K1=N                                                                 FIRST211
      IF(N.NE.NN) GO TO 41                                                 FIRST212
      N1=NN                                                                FIRST213
      GO TO 40                                                             FIRST214
   41 DO 42 N=K1,NN                                                        FIRST215
      IF(XX(N,I).GE.A5) GO TO 13                                           FIRST216
      IF(YY(N,K,I).LT.0.0) GO TO 13                                        FIRST217
   42 CONTINUE                                                             FIRST218
   43 N1=NN                                                                FIRST219
      GO TO 40                                                             FIRST220
C                                          TYPE=5                          FIRST221
   46 DO 50 N=L1,NN                                                        FIRST222
      IF(XX(N,I).GE.A5) GO TO 13                                           FIRST223
      IF(YY(N,K,I).LT.0.0) GO TO 13                                        FIRST224
   50 CONTINUE                                                             FIRST225
      GO TO 43                                                             FIRST226
C                                          BEGIN INTERPOLATION             FIRST227
   40 IL1=XX(L1,I)+1                                                       FIRST228
      IM1=0                                                                FIRST229
      IF(M1.EQ.0) GO TO 51                                                 FIRST230
      IM1=XX(M1,I)+1                                                       FIRST231
      IK1=XX(K1-1,I)+1                                                     FIRST232
   51 IN1=XX(N1,I)+1                                                       FIRST233
      IL2=XX(L1+1,I)+1                                                     FIRST234
      IJK=0                                                                FIRST235
      L2=L1+1                                                              FIRST236
      DO 52 J=1,IK6                                                        FIRST237
      IF(J.GT.IL2) GO TO 53                                                FIRST238
      IF(IJK.EQ.1) GO TO 56                                                FIRST239
      IF(L1.EQ.1) GO TO 54                                                 FIRST240
      IF(L1.GT.NN1) CALL ERROR(52)                                         FIRST241
      X1=XX(L1-1,I)                                                        FIRST242
      X2=XX(L1,I)                                                          FIRST243
      X3=XX(L1+1,I)                                                        FIRST244
      Y1=YY(L1-1,K,I)                                                      FIRST245
      Y2=YY(L1,K,I)                                                        FIRST246
      Y3=YY(L1+1,K,I)                                                      FIRST247
      GO TO  55                                                            FIRST248
   54 X1=XX(1,I)                                                           FIRST249
      X2=XX(2,I)                                                           FIRST250
      X3=XX(3,I)                                                           FIRST251
      Y1=YY(1,K,I)                                                         FIRST252
      Y2=YY(2,K,I)                                                         FIRST253
      Y3=YY(3,K,I)                                                         FIRST254
   55 IF(X1.EQ.X2.AND.X2.EQ.X3.AND.X3.EQ.X1)GO TO 16                       FIRST255
      CALL CC(X1,X2,X3,Y1,Y2,Y3,A,B,C)                                     FIRST256
      IJK=1                                                                FIRST257
   56 IF((X1.EQ.X2).OR.(X2.EQ.X3).OR.(X3.EQ.X1))GO TO 80                   FIRST258
      CALL TERP(A,B,C,J-1,IS)                                              FIRST259
      GO TO 60                                                             FIRST260
   80 CALL TERP1(A,B,J-1,IS)                                               FIRST261
      GO TO 60                                                             FIRST262
   53 IF(IM1.EQ.0)GO TO 57                                                 FIRST263
      IF(J.GT.IM1)GO TO 61                                                 FIRST264
      AM1=J-1                                                              FIRST265
      DO 62 N=L2,M1                                                        FIRST266
      IF((XX(N,I).LE.AM1).AND.(XX(N+1,I).GE.AM1))GO TO 63                  FIRST267
   62 CONTINUE                                                             FIRST268
      GO TO 16                                                             SECOND47
   63 X1=XX(N,I)                                                           FIRST270
      X2=XX(N+1,I)                                                         FIRST271
      X3=XX(N+2,I)                                                         FIRST272
      Y1=YY(N,K,I)                                                         FIRST273
      Y2=YY(N+1,K,I)                                                       FIRST274
      Y3=YY(N+2,K,I)                                                       FIRST275
      IF(X1.EQ.X2.AND.X2.EQ.X3.AND.X3.EQ.X1)GO TO 16                       FIRST276
      CALL CC(X1,X2,X3,Y1,Y2,Y3,A,B,C)                                     FIRST277
      IF((X1.EQ.X2).OR.(X2.EQ.X3).OR.(X3.EQ.X1))GO TO 82                   FIRST278
      CALL TERP(A,B,C,J-1,IS)                                              FIRST279
      GO TO 60                                                             FIRST280
   82 CALL TERP1(A,B,J-1,IS)                                               FIRST281
      GO TO 60                                                             FIRST282
   61 IF(J.LE.IK1)GO TO 52                                                 FIRST283
      IP1=K1                                                               FIRST284
      GO TO 64                                                             FIRST285
   57 IP1=L1+2                                                             FIRST286
      IF(IP1.EQ.N1)IP1=IP1-1                                               FIRST287
   64 IF(J.GT.IN1)GO TO 70                                                 FIRST288
      AM1=J-1                                                              FIRST289
      DO 65 N=IP1,N1                                                       FIRST290
      IF((AM1.GE.XX(N-1,I)).AND.(AM1.LE.XX(N,I)))GO TO 66                  FIRST291
   65 CONTINUE                                                             FIRST292
      GO TO 16                                                             SECOND48
   66 IF(N.EQ.NN)GO TO 67                                                  FIRST294
      X1=XX(N-1,I)                                                         FIRST295
      X2=XX(N,I)                                                           FIRST296
      X3=XX(N+1,I)                                                         FIRST297
      Y1=YY(N-1,K,I)                                                       FIRST298
      Y2=YY(N,K,I)                                                         FIRST299
      Y3=YY(N+1,K,I)                                                       FIRST300
      GO TO 68                                                             FIRST301
   67 X1=XX(N-2,I)                                                         FIRST302
      X2=XX(N-1,I)                                                         FIRST303
      X3=XX(N,I)                                                           FIRST304
      Y1=YY(N-2,K,I)                                                       FIRST305
      Y2=YY(N-1,K,I)                                                       FIRST306
      Y3=YY(N,K,I)                                                         FIRST307
   68 IF(X1.EQ.X2.AND.X2.EQ.X3.AND.X3.EQ.X1)GO TO 16                       FIRST308
      CALL CC(X1,X2,X3,Y1,Y2,Y3,A,B,C)                                     FIRST309
      IF((X1.EQ.X2).OR.(X2.EQ.X3).OR.(X3.EQ.X1))GO TO 83                   FIRST310
```

```
          CALL TERP(A,B,C,J-1,IS)                                     FIRST311
          GO TO 60                                                    FIRST312
       83 CALL TERP1(A,B,J-1,IS)                                      FIRST313
       60 IF((IS.GT.101).OR.(IS.LT.1))GO TO 52                        FIRST314
          IB(IS,J)=IFOR(I)                                            FIRST315
          GO TO 52                                                    FIRST316
       70 IF((XX(N1,I).LT.0.0).OR.(XX(N1,I).GT.A5))GO TO 52           FIRST317
          IF((YY(N1*K*I).LT.0.0).OR.(YY(N1*K*I).GT.101.0))GO TO 52    FIRST318
          GO TO 67                                                    FIRST319
       52 CONTINUE                                                    FIRST320
       16 CONTINUE                                                    FIRST321
       15 CONTINUE                                                    FIRST322
          RETURN                                                      FIRST323
          END                                                         FIRST324
```

* SOURCE STATEMENT *

```
*DK TERP
          SUBROUTINE TERP(A,B,C,J,K)                                  TERP    2
          X=FLOAT(J)                                                  TERP    3
          Y=A*X*X+B*X+C                                               TERP    4
          K=IFIX(Y+1.0)                                               FIRST325
          RETURN                                                      TERP    6
          END                                                         TERP    7
```

* SOURCE STATEMENT *

```
*DK PRINT
          SUBROUTINE  PRINT(IK,IK6,YPR,XPR,IB)                        PRINT   2
          COMMON    / IO /      IOIN     ,IOUT      ,IO1
          DIMENSION YPR(6),XPR(50),IB(101,1)                          THIRD  21
          WRITE(IOUT,2)(YPR(I),I=1,6)                                 FIRST328
        2 FORMAT(12X,6(1PE10.3,10X))                                  PRINT   7
          WRITE(IOUT,3)                                               FIRST329
        3 FORMAT(13X,6(1H|,19X))                                      PRINT   9
          L=0                                                         PRINT  10
          DO 4 I=1,IK6                                                PRINT  11
          I1=I-1                                                      PRINT  12
          I2=MOD(I1,IK)                                               PRINT  13
          IF(I2.NE.0)GO TO 5                                          PRINT  14
          L=L+1                                                       PRINT  15
          XPP=XPR(L)                                                  PRINT  16
          WRITE(IOUT,7)XPP,(IB(J,I),J=1,101)                          FIRST330
        7 FORMAT(2X,1PE9.2,1H-,1X,101A1)                              PRINT  18
          GO TO 4                                                     PRINT  19
        5 WRITE(IOUT,8)(IB(J,I),J=1,101)                              FIRST331
        8 FORMAT(13X,101A1)                                           PRINT  21
        4 CONTINUE                                                    PRINT  22
          RETURN                                                      PRINT  23
          END                                                         PRINT  24
```

* SOURCE STATEMENT *

```
*DK TERP1
          SUBROUTINE TERP1(A,B,J,K)                                   TERP1   2
          X=FLOAT(J)                                                  TERP1   3
          Y=A*X+B                                                     TERP1   4
          K=IFIX(Y+1.0)                                               TERP1   5
          RETURN                                                      TERP1   6
          END                                                         TERP1   7
```

付録　2　　真空磁場の計算コード

```
CCCC    MAIN
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      COMMON / AEQ/ AEQ3
      COMMON / OPT / IZONE,IMESH,IOPY,IOPX
      COMMON / REGION / YMAX,YMIN,XMAX,XMIN
      COMMON / PSAI / EI(10)
      COMMON / RZ / RMAX,RMIN,ZMAX,RA
      COMMON / IO / IOIN,IOUT,IOI
      COMMON / LIMT / LIM1,LIM2,LIM3,EPS1,EPS2,EPS3
      COMMON / IPR / IPR1,IPR2,IPR3
      COMMON/BBBB / BETA,PAI,P0,P1,P2,TI0,VI1,VI2,BT0
      DIMENSION TITLE(9)
      DIMENSION R(5),Z(5)
      PAI=3.14159265358979
      NCASE=0
    6 FORMAT(9A8)
  100 FORMAT(8F10.0)
  710 FORMAT(4I5)
  500 CONTINUE
      READ(5,100,END=999) R0,AA,BT0,QS,BETAJ
      READ(5,100) (R(I),I=1,4)
      READ(5,100) (Z(I),I=1,4)
      AEQ3=AA
      R(5)=R(4)
      Z(5)=Z(4)
      P0=0.
      P2=0.
      VI2=0.
      P1=1.0
      READ(5,6)TITLE
C     TOTAL TOROIDAL CURRENT = CURRTT (KILO AMPERE)
      CURRTT=5.0D3*BT0*AEQ3**2/R0/QS
      TI0=R0*BT0
      TI0=TI0**2
      RP=R(5)
      CURRTT=CURRTT*1.0E3
      AK=2.*CURRTT/PAI/(AEQ3**2)
C     BETA = 2.0*MYU0* P=DASH
C     VI1 / BETA ....  ACCORDING TO BAZHANOVA
C     CURRENT= MYU0/(2.*PAI) * I=MKS
      AK=AK*PAI*4.D-7
      BETA=AK*BETAJ/R0
      VI1=AK*0.5*(1.0-BETAJ)*(R0+DSQRT(R0**2-AEQ3**2))
      WRITE(6,560) NCASE,TITLE
      WRITE(6,604)
      WRITE(6,610) CURRTT
      WRITE(6,611) BETAJ
      WRITE(6,561)R0,AA,BT0,BETA
      WRITE(6,605)
      WRITE(6,601) P0,P1,P2,TI0,VI1,VI2
  560 FORMAT('1',///5X,'CASE NO. ',I3,5X,9A8//)
  604 FORMAT(///10X,'EQUATION'/15X,'L(PSI) + R**2 * EI(6) + EI(7) = 0.0',
     *//)
  610 FORMAT(10X,'TOTAL TOROIDAL CURRENT',T30,1PD18.5)
  611 FORMAT(10X,'BETA-J (BAZHANOVA)',T30,F18.7/)
  600 FORMAT(10X,'NO. OF EQUI-PSI LINES',T30,I8)
  561 FORMAT(       10X,'R0',T30, 1PD18.5/10X,'AA',T30,D18.5/10X,'BT0',
     *  T30,D18.5/10X,'BETA',T30,D18.5//)
  605 FORMAT(5X,'PRESSURE=P0+P1*PSI+P1*PSI**2'/5X,'I**2=TI0+VI1*PSI+VI2*
     *PSI**2')
  601 FORMAT(10X,'P0',T30,1PD18.5/10X,'P1',T30,D18.5/10X,'P2',T30,
     *       D18.5/10X,'TI0',T30,D18.5/10X,'VI1',T30,D18.5/10X,'VI2',
     *       T30,D18.5//)
      CALL COSET (R,Z)
      WRITE(6,678)
      WRITE(6,679) (I,EI(I),I=1,7)
      WRITE(6,606)(I,R(I),Z(I),I=1,5)
  678 FORMAT(///10X,'PSI= EI(1)*R**2*(R**2-4.*Z**2) + EI(2)*(R**2*((R**2
     *-4.*Z**2)*DLOG(R)-3.*Z**2)+2.*Z**4/3.)'/
     *       T13,'EI(3)*R**2  +EI(4)*(R**2*DLOG(R)-Z**2)+EI(5)'/
     *        T13,' -EI(6)*0.5*R**2*Z**2 -EI(7)*0.5*Z**2'/)
  679 FORMAT(7(15X,'EI(',I2,')= ',1PD18.5/))
  602 FORMAT(///10X,'IZONE/IMESH/IOPY/IOPX',T30,4I6)
  603 FORMAT(10X,'YMAX/YMIN/XMAX/XMIN',T30,4F10.3)
C
      CALL LIFSON
      CALL VIDE
      GO TO 500
C
  606 FORMAT(/10X,'INPUT BOUNDARY'/5(15X,I3,3X,'R= ',F8.6,'  Z=',F8.6/))
  999 STOP
      END
```

（手書き注記：READ(5,100) (R(I),I=1,4) / READ(5,100) (Z(I),I=1,4) → プラズマ表面上の4点 → (6)式の係数を定める；CALL LIFSON ← 定数のうけ渡し；CALL VIDE ← 真空磁場の計算）

```
      SUBROUTINE COSET(X,Y)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      COMMON/BBBB / BETA,PAI,P0,P1,P2,TI0,VI1,VI2,BT0
      COMMON/PSAI / EI(10)
      DIMENSION X(1),Y(1),AA(5,6)
      FUNCA(R,Z)=R**2*(R**2-4.*Z**2)
      FUNCB(R,Z)=R**2*((R**2-4.*Z**2)*DLOG(R)-3.*Z**2)+2.*Z**4/3.
      FUNCC(R)=R**2
      FUNCD(R,Z)=R**2*DLOG(R)-Z**2
      EI(6)= 0.5*BETA
      EI(7)= 0.5*VI1
      EI(2)=0.0
      DO 10 I=1,4
      J=I
      IF(I.EQ.4) J=5
      AA(I,1)=FUNCA(X(J),Y(J))
      AA(I,2)=FUNCC(X(J))
      AA(I,3)=FUNCD(X(J),Y(J))
      AA(I,4)=1.0
      AA(I,5)=(EI(6)*X(J)**2 + EI(7))*0.5*Y(J)**2
   10 CONTINUE
      EPS=1.D-12
      CALL SWEEPD(AA,5,4,5,EPS,ILL)
      IF(ILL.EQ.0) GO TO 100
      WRITE(6,101) ILL
  101 FORMAT(//10X,'ERROR IN SWEEPD ( COSET )',I10)
  100 CONTINUE
      EI(1)=AA(1,5)
      EI(3)=AA(2,5)
      EI(4)=AA(3,5)
      EI(5)=AA(4,5)
      RETURN
      END
```

```
      SUBROUTINE LIFSON
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      COMMON / PSAI / EI(10)
```

```
      COMMON / INP  /  NOP(10),ALP,EPS,C,RP1,RCC,RM,NP,NTABT,MULT,NVC,
     *                 NV,NTESTR,NTESTT,NMAX,DELVO,EPSI,ERR
      COMMON / SIG / SIGMA,ETA
      C=EI(4)
      ALP=-2.0*EI(1)
      EPS=-(2.0+0.25*EI(6)/EI(1))
      ETA=1.0+0.5*EI(7)/EI(4)
      SIGMA=-2.0*(2.0*EI(1)+EI(3))/EI(4)
      RETURN
      END


*     MAIN         VACUUM MAGNETIC FIELD
      SUBROUTINE VIDE
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON / INP1 /  IOIN,IOUT,ICA,ICB
      COMMON / INP  /  NOP(10),ALP,EPS,C,RP1,RCC,RM,NP,NTABT,MULT,NVC,      GGG
     *                 NV,NTESTR,NTESTT,NMAX,DELVO,EPSI,ERR                 GGG
      DATA  IOIN /5/,  IOUT /6/,  IOA /1/,  IOB /2/
   10 CALL IMPT   <-- 真空磁場計算用の入力データ
      CALL ADEAM  <-- 真空磁場計算の主要サブルーチン
C     GO TO 10
      END


      SUBROUTINE IMPT
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON / SIG / SIGMA,ETA
      COMMON / INP  /  NOP(10),ALP,EPS,C,RP1,RCC,RM,NP,NTABT,MULT,NVC,
     *                 NV,NTESTR,NTESTT,NMAX,DELVO,EPSI,ERR
      COMMON / INP1 /  IOIN,IOUT,ICA,ICB
      COMMON / TITL /  NTITL(18)
      DATA   NBLNK      /4H    /
      READ(IOIN,1000)  (NTITL(I),I=1,18)
      DO 10  I=1,18
      IF(NTITL(I).NE.NBLNK) GO TO 20
   10 CONTINUE
      STOP 1111
   20 WRITE(IOUT,2100) (NTITL(I),I=1,18)
   30 READ(IOIN,1100)  NSECT
      KSECT=NSECT-20
      GO TO (50,60,70,80),KSECT
   50 READ(IOIN,1300) RP1,RCC,RM
      GO TO 30
   60 READ(IOIN,1400)  NP,NTABT,MULT,NVC,NV,NTESTR,NTESTT,NMAX
      GO TO 30
   70 READ(IOIN,1500)  DELVO,EPSI,ERR
      GO TO 30
   80 CONTINUE
      WRITE(IOUT,2300) RP1,RCC,RM
      WRITE(IOUT,2400) NP,NTABT,MULT,NVC,NV,NTESTR,NTESTT,NMAX
      WRITE(IOUT,2500) DELVO,EPSI,ERR
      RETURN
 1000 FORMAT(18A4)
 1100 FORMAT(I3)
 1200 FORMAT(10I3)
 1300 FORMAT(6D12.0)
 1400 FORMAT(8I3)
 1500 FORMAT(3D12.0)
 2100 FORMAT(1H1/18A4//)
 2300 FORMAT(1H0,5X,23H*** CONSTANT DATA INPUT                          /
     *10X 11HRP1       ...  1PD20.10 /10X 11HRCC       ...  1PD20.10/
     *10X 11HRM        ...  1PD20.10)
 2360 FORMAT(/10X 11HALPHA  ...  1PD20.10/
     * 10X 11HEPSIRON...  1PD20.10/10X 11HC         ...  1PD20.10/
     * 10X 11HSIGMA   ...  1PD20.10/10X 11HETA       ...  1PD20.10/)
 2361 FORMAT(10X 11HA1         ...  1PD20.10
     */10X,11HA3       ...  1PD20.10/10X 11HA4        ...  1PD20.10
     */10X 11HA5       ...  1PD20.10/10X 11HA6        ...  1PD20.10/
     * 10X 11HA7       ...  1PD20.10/)
 2400 FORMAT(1H0,9X  31HEQUI-PHI NO.                   NP     I5     /
     *           10X  31HT-DIRECT DIVIDE NO.            NTABT  I5     /
     *           10X  31HMULTI-FACTOR.                  MULT   I5     /
     *           10X  31HV-DIRECT MESH PTS.             NVC    I5     /
     *           10X  31HMESH PTS FOR INTEGRAL.         NV     I5     /
     *           10X  31HCHECK DATA FOR BJ,BJO.         NTESTR I5     /
     *           10X  31HCHECK DATA FOR BJ,BJO.         NTESTT I5     /
     *           10X  31HMAX COUNT FOR RIEMAN FU.       NMAX   I5     / )
 2500 FORMAT(1H0,9X  31HDATA FOR VMAX.                 DELTVC 1PD20.10 /
     *           10X  31HERROR FOR RIEMAN FUNC.         EPSI   1PD20.10 /
     *           10X  31HERROR FOR DELTA T.             ERR    1PD20.10  )
      END


      SUBROUTINE ADEAM
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION COMPLEX  PPHIV,CRCRE,CK,CE,CKE,SEKI,RE,ZE
      COMMON / SIG / SIGMA,ETA
      COMMON / INP1 /  IOIN,IOUT,ICA,ICB
      COMMON / INP  /  NOP(10),BLP,FPS,CC,RP ,RCC,RM,NP,NTABT,MULT,NVC,
     *                 NV,NTESTR,NTESTT,NMAX,DELVO,EPSI,ERR
      COMMON / PAE  /  PHO,ALP,EPS,C,A,RO
      COMMON / CM1  /  AA,BB
      COMMON / CM2  /  VC
      COMMON / CM4  /  RCR,RCI,RE,ZE,DUM(2)
      DATA   PAI /3.14159265358 98/
      ICTLA=0
      ALP=BLP
      EPS=FPS
      C=CC
      KK=1
      Y2=RP
      EBS=1.0D-13
      RH=-0.02
      SS=Y2-0.1
      EE=0.1
      PH0=0.0
      PH1=FUNC(Y2)
      PHO=-PH1
      CALL  RECFLS(SS,EE,RH,FUNC,EBS,Y1,ILL)
      WRITE(IOUT,2000) Y1,Y2
      A=0.5*(Y2-Y1)
      RO=0.5*(Y2+Y1)
      WRITE(IOUT,2100)PHO,ALP,EPS,C,A
      DELT=PAI/(NTABT*MULT)
      CALL   VMAX(VMA,PAI)
      WRITE(IOUT,2600)VMA
      DELVC= VMA/(NVC-1)
      NTT=NTABT*MULT+1
      DO 30 K=1,NVC        <-- (T,V)メッシュ系でのΨ計算
      VC=(K-1)*DELVC
```

6

```
      DELV=VC/(NV-1)
      WRITE(IOUT,2200)VC
      DO 20 J=1,NTT
      SEK=0.0
      T=DELT*(J-1)
      IF(J.EQ.1) T=ERR                                                  3333
      IF(J.EQ.NTT) T=PAI-ERR                                            3333
      IF(K.EQ.1) GO TO 1
      DO 10  I=1,NV
      V=DELV*(I-1)
      CALL  PHIV(PHIVR,PHIVI,T,V)
      PPHIV=DCMPLX(PHIVR,PHIVI)
      CALL  ARCKE(T,V,ABR,ABI)
      AA=ABR
      BB=ABI
      CALL  FUNCKE(CKER,CKEI)
      CKE=DCMPLX(CKER,CKEI)
      CALL  RCRE(T,V,RCRER,RCREI)
      CRCRE=DCMPLX(RCRER,RCREI)
      SEKI=CRCRE*CKE*PPHIV
      SEK2=DREAL(SEKI)
      SEK=SEK+SEK2*DELV
   10 CONTINUE
      GO TO 2
    1 CALL TVRZ(1,VC,RCR,RCI)
      SEK=0.0
    2 CONTINUE
      SEK=SEK*10000.0
      WRITE(1,301) RCR,RCI,SEK,ICTLA   ! ←(補間・作図(プロッター)サブルーチンへの
C     WRITE(6,201) RCR,RCI,SEK,ICTLA   !   データ転送(テープ)
   20 CONTINUE
   30 CONTINUE
      WRITE(1,302)                     ! ← 同上
      WRITE(6,202)
      RETURN
  201 FORMAT(2X,4HCNTL,1X,3F10.5,35X,I2,8X)
  301 FORMAT(   4HCNTL,1X,3F10.5,35X,I2,8X)
  202 FORMAT(2X,4HBEND,76X)
  302 FORMAT(   4HBEND,76X)
 2000 FORMAT(1H1///10X5HY1 =  1PD20.10,5X5HY2 =  1PD20.10 )
 2100 FORMAT(1H0,///   10X 31HPHO    ALP    EPS    C   A     = 1P5D16.4)
 2200 FORMAT(1H0,///10X  5HVC =  1PD20.5)
 2600 FORMAT(1H0,///10X  5HVMA=  1PD20.10  )
      END
```

```
      SUBROUTINE  REGFLS(SS,EE,RH,FUNC,EBS,Y1,ILL)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      ER=1.0D-13
 1000     X=SS
          F1=FUNC(X)
      IF(DABS(F1).LT.ER) GO TO 5000
 2000     X=X+RH
      IF(X.LT.EE) GO TO 6000
          F2=FUNC(X)
      IF(DABS(F2).LT.ER) GO TO 5000
      IF(F1*F2.LT.0.0) GO TO 3000
          F1=F2
      GO TO 2000
 3000 IF(DABS(RH).LE.EBS) GO TO 4000
          EE=X
          SS=X-RH
          RH=RH/10.0
      GO TO 1000
 4000     Y1=X-0.0*RH
          ILL=0
          RETURN
 5000     Y1=X
          ILL=0
      RETURN
 6000     ILL=3000
          Y1=X
          RETURN
          END
```

```
      SUBROUTINE  VMAX(VMA,PAI)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON / INP1 /  IOIN,IOUT,ICA,ICB                                GGG
      COMMON / INP  /  NOP(10),ALP,EPS,C,RP1,RCC,RM,NF,NTABT,MULT,NVC,
     *                 NV,NTESTR,NTESTT,NMAX,DELVO,EPSI,ERR
      COMMON / PAE  /  DUMM(5),RO
      RR=(RCC+RO)*(RCC-RO)
      MN=0
      DO 30 N=2,6                                                       GGG
      T=PAI*(N-1)/6.0
      IF(N.EQ.1)  T=ERR
      DO 10  L=1,1000
      V=DELVO*L
      CALL  TVRZ(T,V,R,Z)
      ALN=(R-RO)*(R-RO)+Z*Z
      IF(ALN.LE.RR) GO TO 10
      LMIN=L
      GO TO 20
   10 CONTINUE
      GO TO 40
   20 IF(LMIN.LT.MN) GO TO 30
      MN=LMIN
   30 CONTINUE
      VMA= DELVO*MN
      RETURN
   40 WRITE(IOUT,2000)
      STOP
 2000 FORMAT(1H0//10X 50H***** ERROR STOP ... NO VMAX AFTER 1000 ITERATI
     *ON )
      END
```

```
      SUBROUTINE PHIV(PHIVR,PHIVI,T,V)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON / PAE / PHO,ALP,EPS,C,A,RC
      COMMON / SIG / SIGMA,ETA
      COMMON/CM2/VC
      COMMON/CM4/RCR,RCI,RF,ZE,AB
      COMMON / COM  /  RET,ZET
      DOUBLE PRECISION COMPLEX RE,ZE,PHIR,PHIZ,PHIN,PIV,RET,ZET,SCALE,
     *                         TETA,AB,TETA2,RC,ZC
      TETA=DCMPLX(T,V)
      RE=RO+A*DCCOS(TETA)
      ZE=0.5*C+PHO+(ALP-0.5*C*SIGMA)*RE**2-0.5*RE**4*ALP+C*1.0*RE**2
     C*DCLOG(RE)
```

```
      ZE=ZE/(ALP*EPS*RE**2+C*ETA)
      ZE=DCSQRT(ZE)
      CALL TVRZ(T,VC,RCR,RCI)
      PHIR=2.0*ALP*RE-2.0*ALP*RE**3-2.0*ALP*EPS*RE*ZE**2+2.0*C*RE*DCLOG
     1     (RE) +C*(1.0-SIGMA)*RE
      PHIZ=-2.0*ALP*EPS*RE**2*ZE-2.0*C*ZE*ETA
      PHIN=PHIR**2+PHIZ**2
      PHIN=DCSQRT(PHIN)
      RET=-A*DCSIN(TETA)
      ZET=0.5*C*PHO*(ALP-0.5*C*SIGMA)*RE**2-0.5*ALP*RE**4*C*RE**2*
     C DCLOG(RE)
      ZET=ZET/(ALP*EPS*RE**2+C*ETA)
      ZET=DCSQRT(ZET)
        ZET=1.0/ZET
      ZET=ZET*(-ALP**2*EPS*RE**5 +C*ALP*(EPS-2.*ETA)*RE**3
     *      +(C*ETA*(2.*ALP-C*(     SIGMA-1.))-2.*ALP*EPS*(0.5*C*PHO))*RE
     *       +2.*C**2*ETA*RE*DCLOG(RE))
          ZET=ZET/(ALP*EPS*RE**2+C*ETA)**2
          ZET=0.5*ZET*(-A*DCSIN(TETA))
          PIV=PHIR*(-ZET)+PHIZ*RET
      PHIVR= PIV
      PHIVI=DIMAG( PIV)
      RETURN
      END
```

```
      SUBROUTINE  ARGKF(T,V,ABR,ABI)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION COMPLEX  RE,ZE,AB
      COMMON / CM4  /  RCR,RCI,RE,ZE,AB
      AB=(RE-RCR)**2+(ZE-RCI)**2
      AB=-AB
      AB=AB/(4.0*RE*RCR)
      ABR=AB
      ABI=DIMAG(AB)
      RETURN
      END
```

```
      SUBROUTINE  FUNCKF(CKER,CKEI)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION COMPLEX X,YK,AM,FN
      COMMON / INP1 /  IOIN,IOUT,IOA,IOB
      COMMON / INP  /  NOP(10),ALP,EPS,C,RP1,RCC,RM,NP,NTABT,MULT,NVC,        GGG
     *                 NV,NTESTR,NTESTT,NMAX,DELVO,EPSI,ERR                    GGG
      COMMON / CM1  /  AA,BB
      COMMON / CM3  /  D
      I=1
      N=5
      X=DCMPLX(AA,BB)
      AX=DCABS(X)
      YK=(0.0,0.0)
      AM=(1.0,0.0)
   10 DO 20  M=I,N
      B=M
      AY=0.5+B
      AZ=-1.5+B
      AM=AM*AY*AZ/(B*B)*X
      YK=YK+AM
   20 CONTINUE
      FN=YK+1.0
      AFN=DCABS(FN)
      DELTA=AX**(N+1)/((N+1)*(1.0-AX))
      DEL=DELTA/AFN
      IF((DEL/AFN).LT.EPSI) GO TO 30
      I=N+1
      N=N+5
      IF(N.GT.NMAX) GO TO 40
      GO TO 10
   30 CONTINUE
      CKER=DREAL(FN)
      CKEI=DIMAG(FN)
      RETURN
   40 WRITE(IOUT,2100) N
      STOP
 2100 FORMAT(1H0///10X  56H***** ERROR STOP ... RIEMAN FUNC NOT CONVERGE
     *NT AFTER   I8,2X 10HITERATION   )
      END
```

```
      SUBROUTINE  TVRZ(T,V,R,Z)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION COMPLEX  RE,ZE,TETA
      COMMON / SIG / SIGMA,ETA
      COMMON / PAE  /  PHO,ALP,EPS,C,A,RO
      TETA=DCMPLX(T,V)
      RE=RO+A*DCCOS(TETA)
      ZE=0.5*C+PHO+(ALP-0.5*C*SIGMA)*RE**2-0.5*ALP*RE**4+C*RE**2
     C*DCLOG(RE)
      ZE=ZE/(ALP*EPS*RE**2+C*ETA)
      ZE=DCSQRT(ZE)
      R=DREAL(RE)-DIMAG(ZE)
      Z=DIMAG(RE)+DREAL(ZE)
      RETURN
      END
```

```
      SUBROUTINE RCRE (T,V,RCRER,RCREI)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION COMPLEX RCE,RE,ZE
      COMMON/CM4/RCR,RCI,RE,ZE,DUM(2)
         RCE=RCR/RE
      RCE=DCSQRT(RCE)
      RCRER=RCE
      RCREI=DIMAG(RCE)
      RETURN
      END
```

```
      FUNCTION  FUNC(X)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON / SIG / SIGMA,ETA
      COMMON / PAE  /  PHO,ALP,EPS,C,A,RO
      FUNC=0.5*C+PHO+(ALP-0.5*C*SIGMA)*X**2-0.5*ALP*X**4+C*X**2
     C*DLOG(X)
      RETURN
      END
```