INSTYTUT FIZYKI JĄDROWEJ
INSTITUTE OF NUCLEAR PHYSICS
ИНСТИТУТ ЯДЕРНОЙ ФИЗИКИ

KRAKÓW

INP - RAPORT No 1180/PH

A SOLUTION OF THE THERMAL NEUTRON DIFFUSION EQUATION FOR A TWO-REGION CYLINDRICAL SYSTEM PROGRAM FOR ODRA-1305 COMPUTER

KRZYSZTOF DROZDOWICZ
URSZULA WOŹNICKA

KRAKÓW 1982

A SOLUTION OF THE THERMAL NEUTRON DIFFUSION EQUATION FOR A TWO-REGION CYLINDRICAL SYSTEM PROGRAM FOR ODRA-1305 COMPUTER

ROZWIĄZANIE RÓWNANIA DYFUZJI NEUTRONÓW TERMICZNYCH DLA DWUSTREFOWEGO UKŁADU CYLINDRYCZNEGO PROGRAM DLA E.M.C. ODRA-1305

РЕШЕНИЕ УРАВНЕНИЯ ДИФФУЗИИ ТЕПЛОВЫХ НЕЙТРОНОВ ДЛЯ ДВУХЗОННОЙ ЦИЛИНДРИЧЕСКОЙ СИСТЕМЫ ПРОГРАММА ДЛЯ ВЫЧИСЛИТЕЛЬНОЙ МАШИНЫ ODRA-1305

Krzysztof Drozdowicz, Urszula Woźnicka
Institute of Nuclear Physics

Kraków, May 1982

NAKŁADEM INSTYTUTU FIZYKI JĄDROWEJ W KRAKOWIE
UL. RADZIKOWSKIEGO 152

Kopię kserograficzną, druk i oprawę wykonano w IFJ Kraków

Wydanie I *Zam. 87/82* *Nakład 120 egz.*

ABSTRACT

The program in FORTRAN for the ODRA-1305 computer is described. The dependence of the decay constant of the thermal neutron flux upon the dimensions of the two-region concentric cylindrical system is the result of the program. The solution (with a constant neutron flux in the inner medium assumed) is generally obtained in the one-group diffusion approximation by the method of the perturbation calculation. However, the energy distribution of the thermal neutron flux and the diffusion cooling are taken into account. The program is written for the case when the outer medium is hydrogeneous. The listing of the program and an example of calculation results are included.

STRESZCZENIE

W pracy opisany jest program w FORTRANie dla e.m.c. ODRA-1305. Wynikiem programu jest zależność stałej zaniku strumienia neutronów termicznych od rozmiarów dwustrefowego układu cylindrycznego. Rozwiązanie (przy założeniu stałości strumienia neutronów w wewnętrznym ośrodku) uzyskane jest zasadniczo w jednogrupowym przybliżeniu dyfuzyjnym metodą rachunku zaburzeń. Jednakowoż uwzględnia się rozkład energetyczny strumienia neutronów termicznych oraz ochładzanie dyfuzyjne. Program napisano dla przypadku, gdy zewnętrzny ośrodek zawiera wodór. Na końcu załączony jest lieting programu oraz wyniki obliczeń dla przykładowego zestawu danych.

РЕЗЮМЕ

В статьи представлена программа на языке FORTRAN для вычислительной машины ODRA-1305. Окончательным результатом является зависимость декремента затухания потока тепловых нейтронов от размеров двухзонной цилиндрической системы. Решение (при основании постоянной величины потока нейтронов в внутренной среде) в основном получено в одногрупповой диффузионной аппроксимации методом возмущений. Однако, применено во внимание энергетическое распределение потока тепловых нейтронов и диффузионные охлождение. Программа написана для случая когда внешний модератор находится водородосодержащий. В последней части приложена листовка программы и комплект результатов в качестве примера.

1. INTRODUCTION

The report describes a computer program which gives a dependence of the decay constant λ_o^* of a thermal neutron flux upon dimensions of the two-region concentric cylindrical system. The dynamic diffusion equation for such a case has been solved by Woźnicka (1981) by means of the perturbation calculation. The theoretical decay constant λ_o^* has been obtained under assumpton of a constant thermal neutron flux $\phi_1^*(z,\rho)$ in the internal cylinder. The detailed calculation formulae with energy corrections (for hydrogenous moderators) have been given by Drozdowicz and Woźnicka (1982). Here some explanations of the calculation procedure are presented. The height H_1 and radius R_1 of the internal cylinder are fixed in the program (as input data) and the dimensions H_{2g}, R_{2g} of the external cylinder are variable. The program is written in the FORTRAN 1900 language (a version of FORTRAN 4).

2. INPUT DATA

# TEOR Name of the variable	The input data are read in the format F0.0 in one record:		
C	C_2	[cm^4/s]	- diffusion cooling coefficient of thermal neutrons in external moderator,
VSIGA	$\overline{v\Sigma}_{a2}$	[1/s]	- reciprocal of thermal neutron lifetime in external moderator,
DO	D_{o2}	[cm^2/s]	- pulsed diffusion coefficient of thermal neutrons in external moderator,
TETA	θ	[^oC]	- temperature,

R1	R_1	[cm]	- radius of internal cylinder,
H1	H_1	[cm]	- height of internal cylinder,
R2GPOCZ R2G	$R_{2g\,pocz}$	[cm]	- minimum value of radius R_{2g} of external cylinder,
H2GPOCZ H2G	$H_{2g\,pocz}$	[cm]	- minimum value of height H_{2g} of external cylinder,
R2GKON	$R_{2g\,kon}$	[cm]	- maximum value of radius R_{2g} of external cylinder,
H2GKON	$H_{2g\,kon}$	[cm]	- maximum value of height H_{2g} of external cylinder,
DELTAR	ΔR	[cm]	- step of R_{2g},
DELTAH	ΔH	[cm]	- step of H_{2g}.

3. RUN OF THE PROGRAM

The value of the decay constant λ_o^* is calculated for each value R_{2g} and H_{2g} in the two iterative loops. The external one is the result of the method of the perturbation calculation (Woźnicka 1981) and gives the sequential approximations $\lambda_o^{*(n)}$ of the λ_o^* value. Each consecutive n-order approximation $\lambda_o^{*(n)}$ is first treated in the internal loop. Such an approach follows from the interdependence between the decay constant $\lambda_o^{*(n)}$ being sought and all the other quantities being energy corrected: the average velocity of thermal neutrons, the average dynamic macroscopic cross-sections, the average diffusion coefficient, the dynamic extrapolated length etc. (cf. Drozdowicz 1981).

The whole procedure is repeated for the new values of R_{2g}, H_{2g}, for which the new decay constant λ_o^* is obtained.

3.1. MAIN PROGRAM: MASTER CYLINDER

The flow chart of the main program is given in Fig.1.

N — n - number of the sequential approximation of the perturbation calculation (index of the external loop),

DD — d_d - dynamic extrapolated length,

H2, R2 —
$$\left.\begin{aligned} H_2 &= H_{2g} + 2\,d_d \\ R_2 &= R_{2g} + d_d \end{aligned}\right\} \text{extrapolated dimensions.} \qquad (1)$$

Determination of the zero-order approximation (n = 0).

The internal loop starts with $d_d = 0$. The zero-order approximation of α is assumed as:

ALPA
$$\alpha^{(o)} = \frac{\pi}{H_2 - H_1} \quad . \qquad (2)$$

W — The value of $W^{(o)}$ is determined from the equation:

$$\frac{J_1(W^{(o)}R_1)}{Y_1(W^{(o)}R_1)} = \frac{J_o(W^{(o)}R_2)}{Y_o(W^{(o)}R_2)} \quad , \qquad (3)$$

[cf. Drozdowicz and Woźnicka (1982) Eq.(36) and (37)], where $J_k(x)$, $Y_k(x)$ are the Bessel functions of the first and second kind, respectively, of the k order. The graphic solution of Eq.(3) is presented in Figs 2 and 3. It is visible that the interval where the solution should be found numerically depends upon the value of the ratio R_2/R_1. The numerical solution of Eq.(3) is executed by the fol-

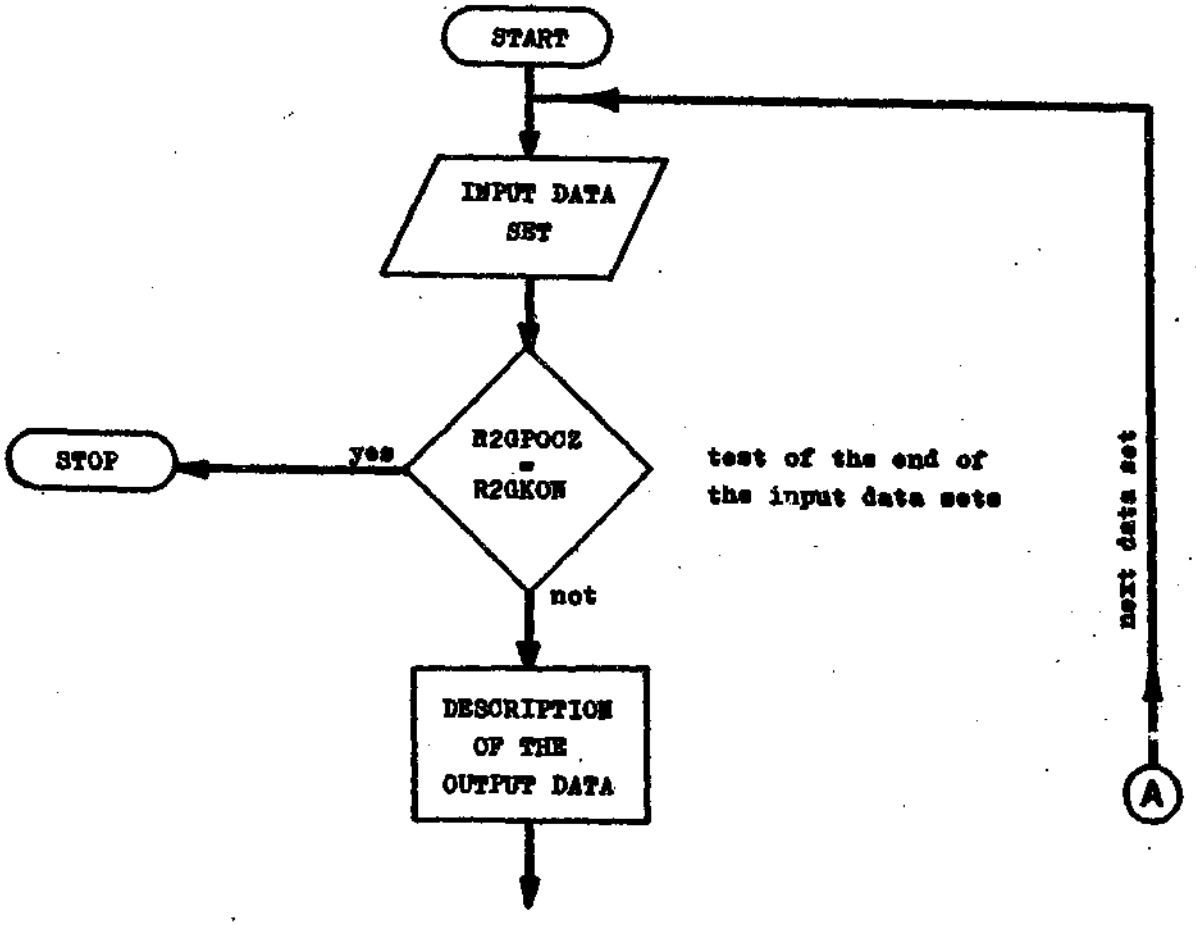


Fig.1. Flow chart of the program TEOR.

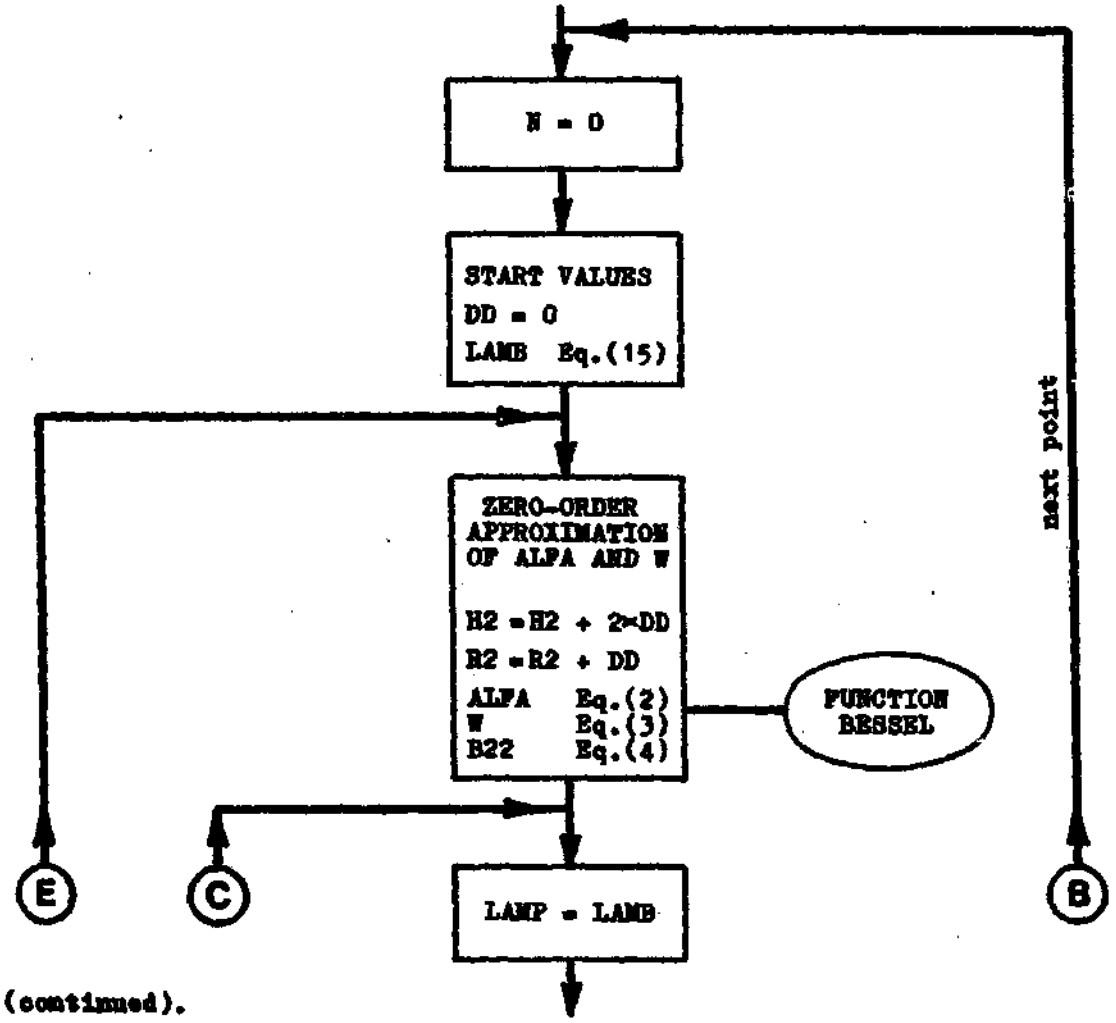


Fig.1 (continued).

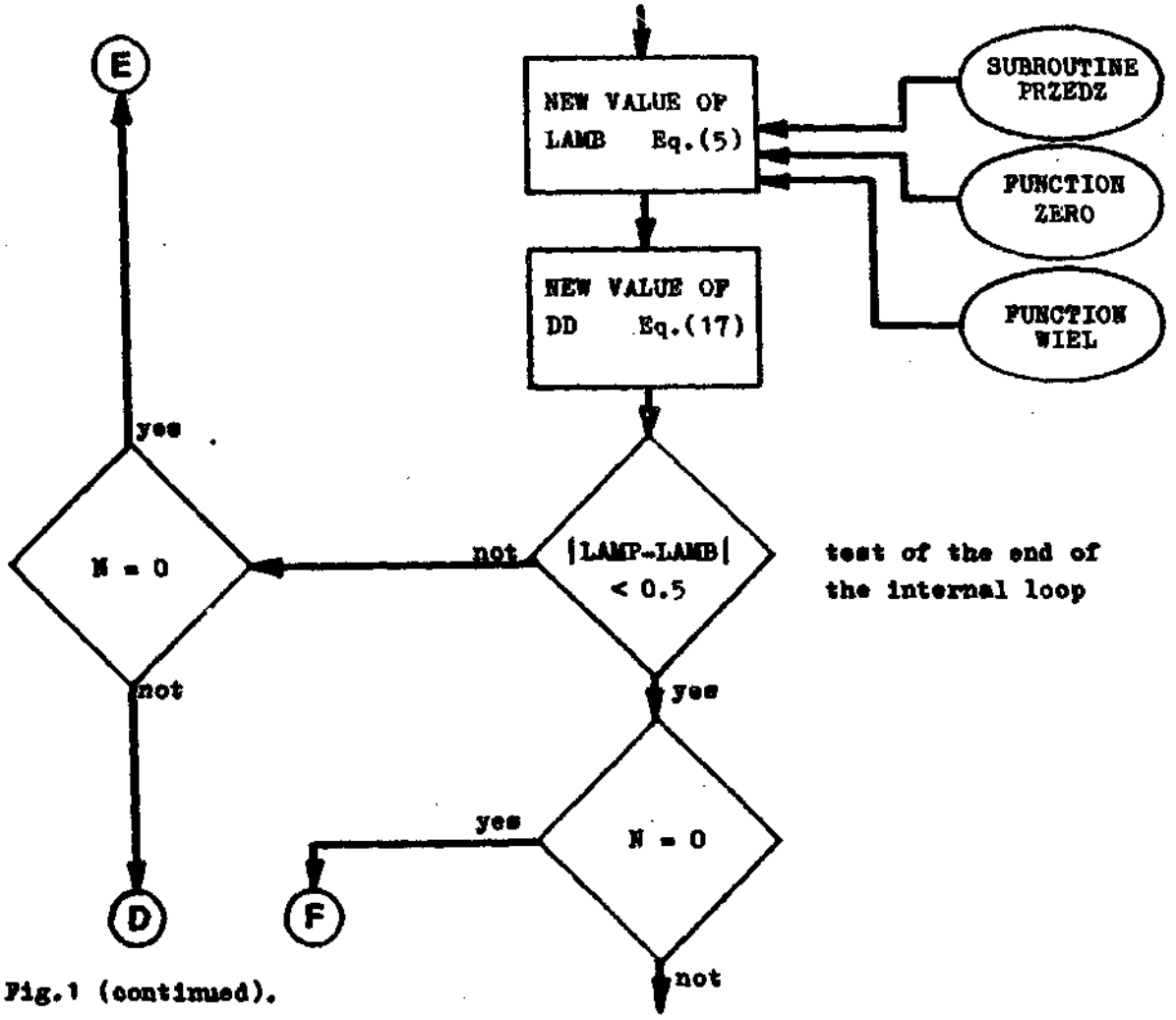


Fig.1 (continued).

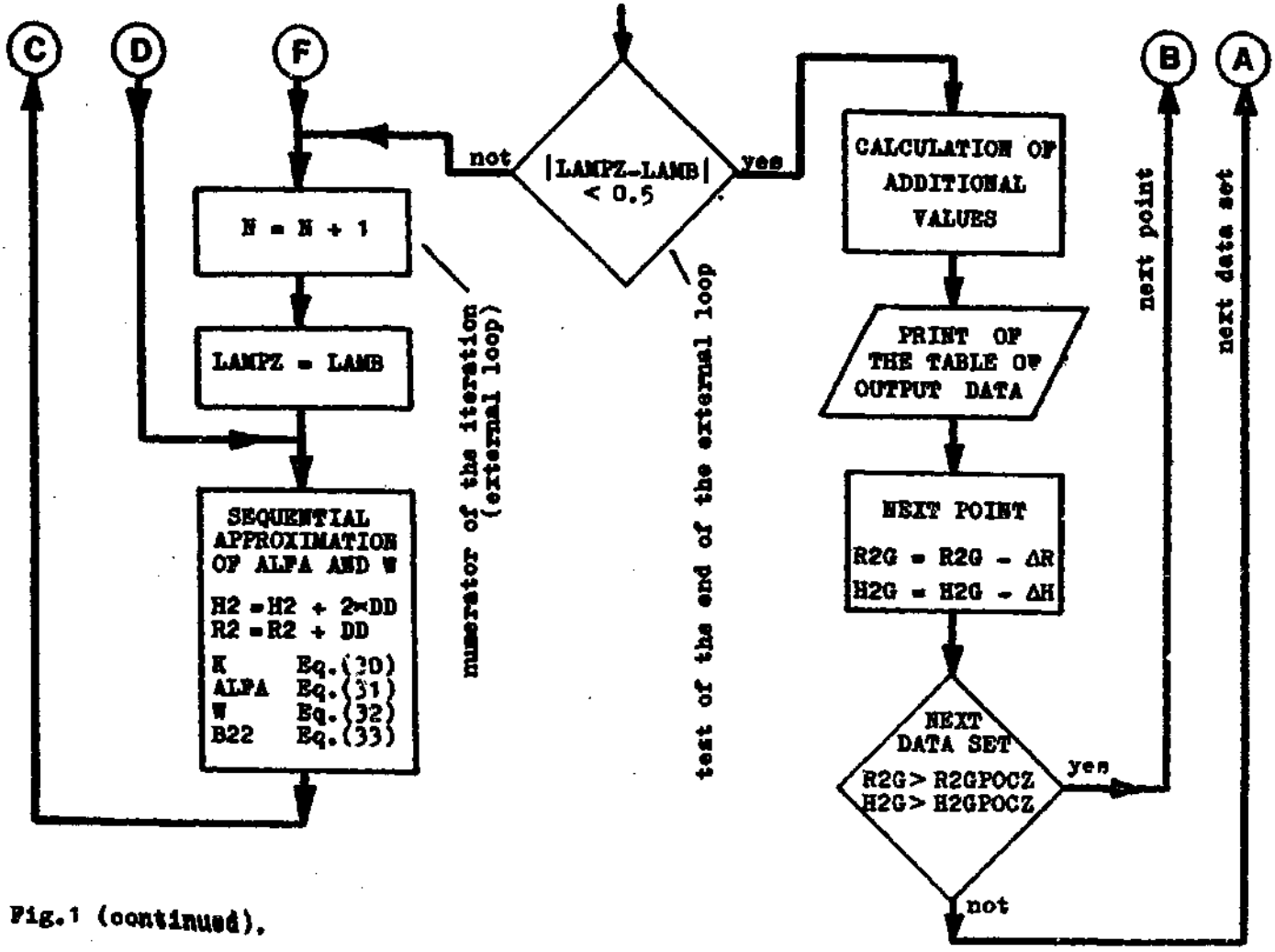
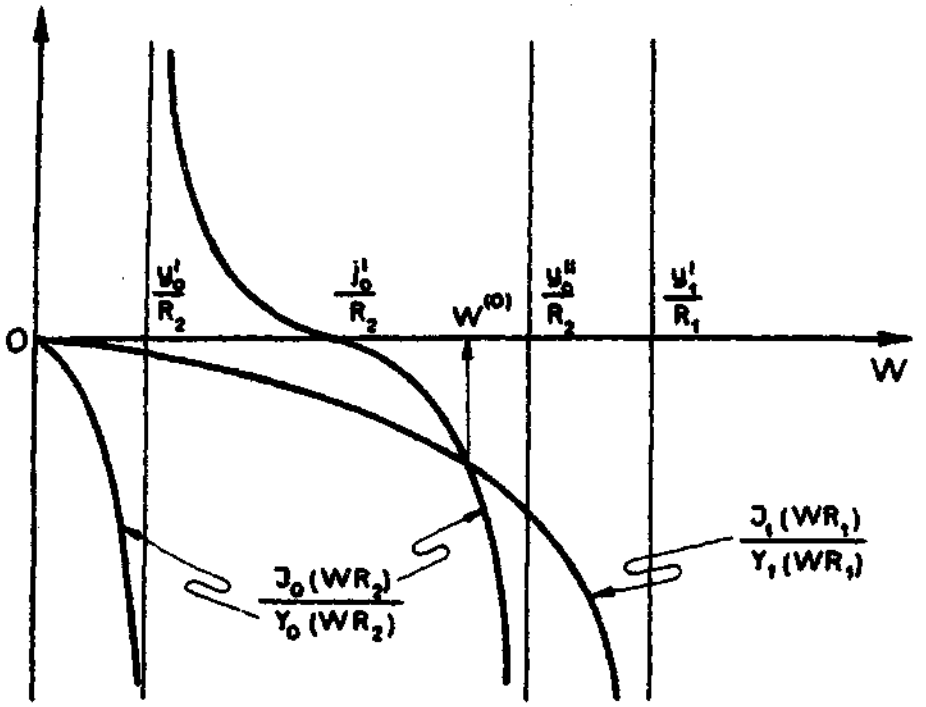


Fig.1 (continued).

CASE I: $\frac{R_2}{R_1} > \frac{y_0''}{y_1'}$



$j_0' \approx 2.405$ – FIRST ZERO OF $J_0(WR_2)$

$y_0' \approx 0.893$ – FIRST ZERO OF $Y_0(WR_2)$

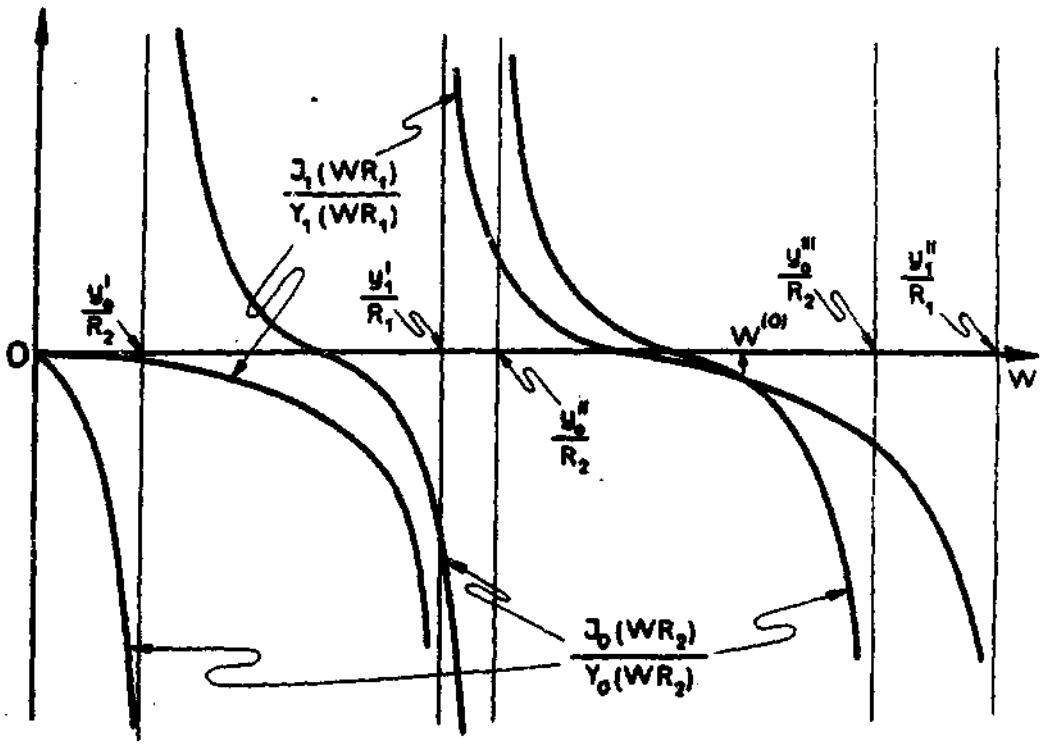
$y_0'' \approx 3.958$ – SECOND ZERO OF $Y_0(WR_2)$

$y_1' \approx 2.198$ – FIRST ZERO OF $Y_1(WR_1)$

FIG. 2. GRAPHIC SOLUTION OF EQUATION (3):

$$J_1(W^{(0)}R_1)/Y_1(W^{(0)}R_1) = J_0(W^{(0)}R_2)/Y_0(W^{(0)}R_2)$$

CASE II : $\frac{y_0'''}{y_1''} < \frac{R_0}{R_1} < \frac{y_0''}{y_1'}$



y_0' ≅ 0.893 – FIRST ZERO OF $Y_0(WR_2)$

y_1' ≅ 2.198 – FIRST ZERO OF $Y_1(WR_1)$

y_0'' ≅ 7.086 – THIRD ZERO OF $Y_0(WR_2)$

y_1'' ≅ 5.430 – SECOND ZERO OF $Y_1(WR_1)$

FIG. 3. GRAPHIC SOLUTION OF EQUATION (3)

$$J_1(W^{(0)}R_1) / Y_1(W^{(0)}R_1) = J_2(W^{(0)}R_2) / Y_2(W^{(0)}R_2)$$

BESSEL ZERO

lowing subprograms: FUNCTION BESSEL, SUBROUTINE PRZEDZ and FUNCTION ZERO. The decay constant is calculated from Eq.(31), reported by Drozdowicz and Woźnicka (1982). In the program it is rewritten as:

B22

$$(B_2^{(o)})^2 = (\alpha^{(o)})^2 + (W^{(o)})^2 \quad , \tag{4}$$

which is compared with:

$$(B_2^{(o)})^2 = -\frac{\bar{\Sigma}_{ad}}{\bar{D}_d}\cdot 2\left\{1 + \left[1 + 4\frac{\bar{\Sigma}_{ad}}{\bar{D}_d}\frac{(\overline{1/v})}{2} C_2\right]^{\frac{1}{2}}\right\}^{-1}, \tag{5}$$

where:

SIGAD

$$\bar{\Sigma}_{ad} = \bar{\Sigma}_a - \bar{\lambda}_o \quad , \tag{6}$$

SIGA

$$\bar{\Sigma}_a = (\overline{1/v})\cdot \overline{v\Sigma}_{a2} \quad , \tag{7}$$

DDK

$$\bar{D}_d = (\overline{1/v})\,\bar{v}\,\frac{1}{3(\bar{\Sigma}_D - \bar{\lambda}_o)} \quad , \tag{8}$$

SIGD

$$\bar{\Sigma}_D = \bar{v}/(3\,D_{o2}) \quad , \tag{9}$$

V1K

$$(\overline{1/v}) = \frac{\sqrt{\pi}}{2}\cdot\frac{1}{v_{oc}} \quad , \tag{10}$$

VK

$$\bar{v} = \frac{3}{4}\sqrt{\pi}\; v_{oc} \quad , \tag{11}$$

VOC

$$v_{oc} = v_o\left(1 - \frac{C_2}{D_{o2}}\cdot B_2^{(o)2}\right) \quad , \tag{12}$$

VO

$$v_o = 220000\,\sqrt{(\theta + 273{,}2)/293{,}6} \quad \text{[cm/s]} \quad , \tag{13}$$

LOK

$$\overline{\lambda}_o = (\overline{1/v})\,\lambda_o^{*(o)} \quad . \tag{14}$$

WIEL

The decay constant $\lambda_o^{*(o)}$ is calculated using the subprograms FUNCTION WIEL, SUBROUTINE PRZEDZ and FUNCTION ZERO. The start value of $\lambda_o^{*(o)}$ in the first execution of the internal loop is assumed to be equal to the approximated value of the decay constant for the homogenous cylindrical moderator:

LAMB

$$\lambda_{o\ start}^{*(o)} = \overline{v\Sigma}_{a2} + D_{o2}\left[\left(\frac{\pi}{H_{2g}}\right)^2 + \left(\frac{j_o}{R_{2g}}\right)^2\right] \quad , \tag{15}$$

JO

where $j_o = 2.405$. (16)

Next, the extrapolation length

DD

$$d_d = 2.28\,\overline{D}_d \tag{17}$$

is introduced into the dimensions R_2 and H_2 of the system [cf. Eq.(1)]. $\overline{D}_d$ is the average dynamic diffusion coefficient of thermal neutrons and it is λ_o^* - dependent. The extrapolated dimensions R_2 and H_2 [cf. Eqs (1), (17), (8) + (14)] are calculated using the value of the decay constant together with the other quantities obtained previously. The execution of the internal loop is now repeated [Eqs (2) + (14)], the new value of the decay constant is obtained and so on. The process ends when the difference between the two consecutive values of the decay constant $\lambda_o^{*(o)}$ is smaller than $0.5\ s^{-1}$ (it corresponds to the calculation accuracy equal from 0.01 % up to 0.001 % in the range of the decay constants considered). The zero-order approximation of the decay constant $\lambda_o^{*(o)}$, i.e. the first value of the external loop is then obtained.

Higher-order approximations (n = 1, 2, ...).

The values of the quantities in the n-order approximation are calculated using those obtained in the preceding (n-1)-order approximation (cf. Drozdowicz and Woźnicka 1982). Here the formulae are:

M

$$M^{(n)} = \frac{\cos[\alpha^{(n)}\,(h_2 - h_1) - \alpha_e h_1]}{2\,(\alpha^{(n)} + \alpha_e)} + \frac{\cos[\alpha^{(n)}\,(h_2 - h_1) + \alpha_e h_1]}{2\,(\alpha^{(n)} - \alpha_e)} \quad , \tag{18}$$

C4A

$$N^{(n)} = W^{(n)}\,R_1\,J_0(W_e R_1)\,T^{(n)} - W_2\,R_1\,J_1(W_e R_1) \quad , \tag{19}$$

S

$$S^{(n)} = \frac{J_0(W^{(n)} R_2)}{Y_0(W^{(n)} R_2)} \quad , \tag{20}$$

$$T^{(n)} = \frac{J_1(W^{(n)} R_1) - S^{(n)}\,Y_1(W^{(n)} R_1)}{J_0(W^{(n)} R_1) - S^{(n)}\,Y_0(W^{(n)} R_1)} \quad , \tag{21}$$

$$\alpha_e = \frac{\pi}{H_2} \quad , \tag{22}$$

$$W_e = \frac{J_0}{R_2} \quad , \tag{23}$$

$$h_1 = \frac{H_1}{2} \quad , \tag{24}$$

$$h_2 = \frac{H_2}{2} \quad , \tag{25}$$

C1

$$a_1^{(n)} = \frac{H_2 R_1}{W_e} \sin(\alpha_e h_1)\, J_1(W_e R_1) \quad , \tag{26}$$

C2

$$a_2^{(n)} = \frac{H_2 \sin(\alpha_e h_1)}{W_e^2 - (W^{(n-1)})^2}\, H^{(n-1)} \quad , \tag{27}$$

C3

$$a_3^{(n)} = \frac{\pi R_1}{W_e} \cdot \frac{J_1(W_e R_1)}{\sin[\alpha^{(n-1)}(h_1 - h_2)]}\, H^{(n-1)} \quad , \tag{28}$$

C4

$$a_4^{(n)} = \frac{\pi}{W_e^2 - (W^{(n-1)})^2} \cdot \frac{H^{(n-1)}\, H^{(n-1)}}{\sin[\alpha^{(n-1)}(h_1 - h_2)]} \tag{29}$$

K

$$K^{(n)} = 1 + \frac{a_1^{(n)}}{a_2^{(n)} + a_3^{(n)} + a_4^{(n)}} \quad , \tag{30}$$

ALFA

$$\alpha^{*(n)} = \alpha_e \sqrt{K^{(n)}} \quad , \tag{31}$$

W

$$W^{*(n)} = W_e \sqrt{K^{(n)}} \quad , \tag{32}$$

B22

$$(B_{d2}^{*(n)})^2 = (\alpha^{*(n)})^2 + (W^{*(n)})^2 \quad . \tag{33}$$

Next, the $\tilde{c}^{(n)}$ value is calculated as in the zero-order approximation from FUNCTION WIEL, SUBROUTINE PRZEDZ and FUNCTION ZERO, and the internal loop is executed as described above.

The sequential executions of the external loop give the sequential approximations $\lambda_o^{*(n)}$ of the decay constant λ_o^*. The loop is satisfied when the condition

$$| \lambda_o^{*(n-1)} - \lambda_o^{*(n)} | < 0.5\ s^{-1}$$

is fulfilled. An example is given in Fig.4, where the values of the decay constant $\lambda_0^{*(n)}$ obtained in the sequential approximations are depicted.

3.2. DESCRIPTION OF THE SUBPROGRAMS

SUBROUTINE PRZEDZ (F,E,A1,B1,A,B)

The subroutine determines the interval [A,B] where the zero of the function F will be found. The rest of the dummy arguments:

A, B
F

A1 - input parameter; lower limit of the investigated interval,

A1

B1 - input parameter; upper limit of the investigated interval,

B1

E - step of the investigation of the interval [A1,B1].

E

The subroutine determines the interval in which the sign of the function changes. The width of the output interval [A,B] is equal E.

FUNCTION ZERO (A,B,F,E)

ZERO
F
A, B
E

The subprogram FUNCTION ZERO calculates the zero of the function F in the interval [A,B] with the precision E. F is the name of any function. The zero of the function is determined as the point in the middle of the interval of the width E (the values of the function F have the opposite signs for the upper and lower limits of the interval).

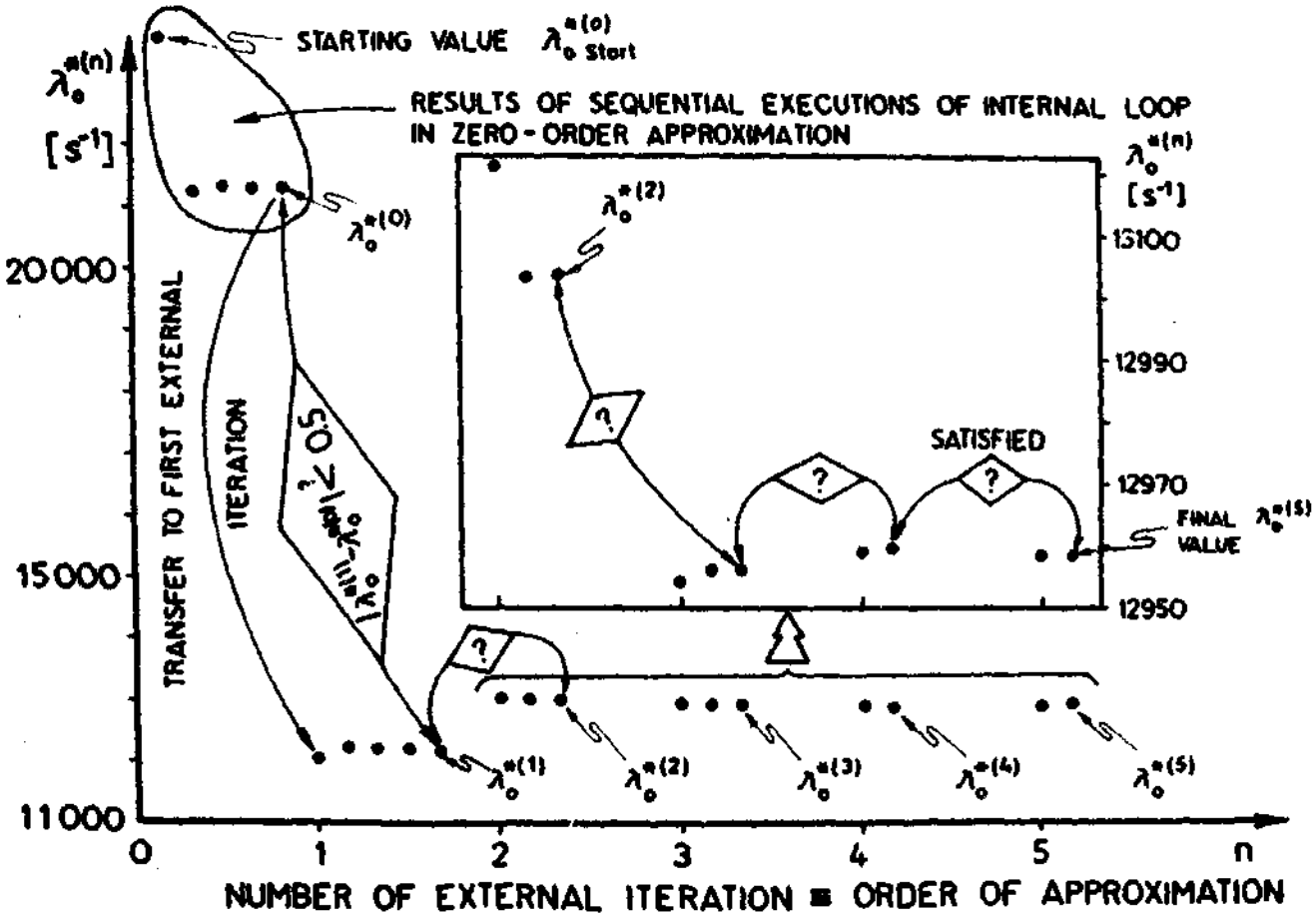


FIG. 4. EXAMPLE OF THE VALUES $\lambda_0^{*(n)}$ OBTAINED IN SEQUENTIAL APPROXIMATIONS

FUNCTION BESSEL (WO)

The subprogram FUNCTION BESSEL calculates the value of the expression:

BESSEL

$$\frac{J_1(w^{(o)}R_1)}{Y_1(w^{(o)}R_1)} - \frac{J_o(w^{(o)}R_2)}{Y_o(w^{(o)}R_2)} \qquad (34)$$

WO

for the real argument $w^{(o)}$. It utilizes the Bessel functions from the library of subprograms for the ODRA-1305 computer (ELWRO 1971).

FUNCTION WIEL (LAMBDA)

The subprogram FUNCTION WIEL calculates the value of the expression:

WIEL

$$(B_{d2}^{(n)})^2 + \frac{\bar{\Sigma}_{ad}}{D_d}\cdot 2\left\{1 + \left[1 + 4\frac{\bar{\Sigma}_{ad}}{D_d^2}(\overline{1/v})\,c_2\right]^{\frac{1}{2}}\right\}^{-1} \qquad (35)$$

LAMBDA

for the actual value of the decay constant $\lambda_o^{*(n)}$.

4. OUTPUT DATA

The output data are printed in the table which contains:

R2G — R_{2g} - radius of the external cylinder,

H2G — H_{2g} - height of the external cylinder,

H2G**-2 — $(1/H_{2g})^2$

R2DYN — $R_{2\,dyn} = R_{2g} + d_d$, (36)

H2DYN — $H_{2\,dyn} = H_{2g} + 2d_d$. (37)

R2STAT

$$R_{2\ stat} = R_{2g} + d_R \tag{38}$$

where

$$d_R = \frac{d}{1 + \frac{d}{R_{2g}}} , \tag{39}$$

$$d = 2.28\ \overline{(1/v)}\ D_{o2} , \tag{40}$$

H2STAT

$$H_{2\ stat} = H_{2g} + 2d , \tag{41}$$

PARGEOM

$$P_{geom} = \left(\frac{1}{H_{2g}}\right)^2 + \left(\frac{1}{R_{2g}}\right)^2 , \tag{42}$$

B**2STAT

$$B^2_{stat} = \left(\frac{J_o}{R_{2\ stat}}\right)^2 + \left(\frac{\pi}{H_{2\ stat}}\right)^2 \tag{43}$$

(i.e. buckling of the homogeneous cylinder),

LAMBDA

$\lambda_o^{*(n)}$ - the decay constant value from the last approximation.

Conditional print in the first column of the output table:

'B'

'B' - informs about the limitation of the value of buckling $(B_2^{(o)})^2$ in the zero-order approximation only. The maximum buckling is assumed:

BKON

$$(B_2^{(o)})^2_{max} = 0.3\,\frac{D_{o2}}{C_2} \tag{44}$$

and 'B' is printed if the value obtained in the standard run of the program is greater than $(B_2^{(o)})^2_{max}$.

'W'	'W'	- is printed if the solution of Eq.(3) does not exist in the interval from 0 up to the first asymptote of the function $J_0(W^{(o)}R_2)/Y_0(W^{(o)}R_2)$ (cf. Figs 2, 3). Then the maximum value $W^{(o)}_{max}$ is assumed:
B1		$$W^{(o)}_{max} = \frac{y_2}{R_2} \qquad (45)$$
		where $y_2 = 3.958$ is the second zero of the Bessel function $Y_0(x)$. It concerns the zero-order approximation only.

5. LISTING OF THE PROGRAM

```
      MASTER CYLINDER - THEORETICAL CURVE

      REAL J0,LAMP,LAMB,LAMPZ,M1,M2,M3,M4,M,K,LAMC
      EXTERNAL BESSEL,WIEL
      COMMON /THREE/ VK,V1K,SIGD,SIGA,C,B22,DDK
      COMMON /TWO/ R1,R2
      DATA PI,J0 /3.141592654,2.405/

C     SPECIFICATION OF THE INPUT DATA (ONE RECORD)

C     C       - DIFFUSION COOLING COEFFICIENT [CM**4/S]
C     VSIGA   - RECIPROCAL OF LIFETIME OF THERMAL NEUTRONS [1/S]
C     DU      - PULSED DIFFUSION COEFFICIENT OF THERMAL NEUTRONS
C     TETA    - TEMPERATURE [ C]                                  [CM**2/S]
C     R1      - RADIUS OF INTERNAL CYLINDER [CM]
C     H1      - HEIGHT OF INTERNAL CYLINDER [CM]
C     R2GPOCZ - MINIMUM VALUE OF RADIUS R2G OF EXTERNAL CYLINDER [CM]
C     H2GPOCZ - MINIMUM VALUE OF HEIGHT H2G OF EXTERNAL CYLINDER [CM]
C     R2GKON  - MAXIMUM VALUE OF RADIUS R2G OF EXTERNAL CYLINDER [CM]
C     H2GKON  - MAXIMUM VALUE OF HEIGHT H2G OF EXTERNAL CYLINDER [CM]
C     DELTAR  - STEP OF H2G [CM]
C     DELTAH  - STEP OF H2G [CM]
```

```
0031
0032          99    READ(2,100) C,VSIGA,DD,TETA,R1,H1,R2GPOCZ,R2GPOCZ,R2GKON,R2GKON,
0033               *DELTAR,DELTAH
0034          100   FORMAT(12F0.0)
0035                WRITE(3,200)
0036                IF(R2GPOCZ.EQ.R2GKON) STOP
0037                CONTINUE
0038
0039          C     DESCRIPTION OF THE OUTPUT DATA
0040                WRITE(3,200)
0041          200   FORMAT(1H1,///1X,120(1H*))
0042                WRITE(3,201)
0043          201   FORMAT(///30X,24HCYLINDER - THEORETICAL CURVE)
0044                WRITE(3,202)
0045          202   FORMAT(//25X,14HV=SIGMA A[1/S),5X,13HDD[(CM**2)/S] ,5X,12HC[(CM**4
0046               *)/S])
0047                WRITE(3,203) VSIGA,DD,C
0048          203   FORMAT(/ 9X,9HMODERATOR, 5X,F10.1,9X,F10.1,8X,F8.1//)
0049                WRITE(3,204) TETA,R1,R2GPOCZ,R2GKON,H1,H2GPOCZ,H2GKON
0050          204   FORMAT( 9X,6HTHETA=,F4.1,5H ST.///9X,4HR1G=,F5.2,3H CM,20X,4HR2G=,
0051               *F4.1,5H ... ,F4.1,3H CM/ 9X,4HH1G=,F5.2,3H CM,20X,4HH2G=,F4.1,5H ,
0052               *... ,F4.1,3H CM)
0053                WRITE(3,205)
0054          205   FORMAT(//1X,120(1H*)///)
0055                  WRITE(3,206)
0056          206   FORMAT(10X,108H -   R2G       H2G     H2G**-2     REDYN        H2DYN
0057               *   H2STAT       H2STAT      PARGEOM      B**2 STAT     LAMBDA//)
0058                WRITE(3,207)
0059          207   FORMAT(14X,4H[CM],5X,4H[CM],5X,8H[CM**-2],5X,4H[CM],5X,4H[C*),2(8X
0060               *,4H[CM]),2(12H   [CM**-2] ),9H    [1/S])
0061                WRITE(3,205)
0062
0063                H2G=H2GKON
0064                R2G=R2GKON
0065                LAMB=VSIGA+DD+((PI/H2G)**2+(JO/R2G)**2)
0066                POCZ=VSIGA+1.
0067
```

```
0068          9     M=0
0069                DD=0
0070          10    R2=R26+2*DD
0071                R2=R26+DD
0072
0073          C     ZERO ORDER APPROXIMATION OF ALFA AND W VALUES
0074                ALFA=PI/(R2-R1)
0075                A1=0.
0076                B1=3.958/R2
0077          11    CALL PRZEDZ(BESSEL,.005,A1,B1,A,B)
0078                IF(B.EQ.0.)GO TO 13
0079                W=ZERO(A,B,BESSEL,0.001)
0080          13    CONTINUE
0081                B=W
0082                W=61
0083
0084          12    B22=ALFA**2+W**2
0085                MB22=0
0086                IF(C.EQ.0) GO TO 20
0087                BKON=0.5*D0/C
0088                IF(B22.LE.BKON)GOTO 20
0089                B22=BKON
0090                MB22=1
0091          20    LAMP=LAMB
0092                V0=220000.*SQRT((TETA+273.2)/293.6)
0093                VOC=V0*(1.-C/D0*B22)
0094                VK=0.75*SQRT(PI)*VOC
0095                V1K=0.5*SQRT(PI)/VOC
0096                SIGD=VK/(3*D0)
0097                SIGA=V1K*VSIGA
0098                AKON=200000
0099                WRITE(5,104)N,POCZ
0100          104   FORMAT(1X,'N = ',I3,'POCZ = ',F8.1)
0101                CALL PRZEDZ(WIEL,1000.,POCZ, AKON, A,B)
0102                IF(B.NE.0.) GO TO 21
0103
```

```
0104          C     CONDITIONAL PRINT IF LAMB > AKON
0105                WRITE(3,102)
0106          102   FORMAT(15X,16HLAMBDA NOT FOUND)
0107                GOTO 60
0108            21  LAMB=ZERO(A,B,VIEL,0.5)
0109                DD=2.28*UDK
0110
0111          C     TEST OF THE FINISH OF THE INTERNAL ITERATION
0112                IF(ABS(LAMP-LAMB).LT.0.5 ) GO TO 22
0113                IF(N.EQ.0) GO TO 10
0114                GO TO 40
0115          22    V1KP=V1K
0116                IF(N.EQ.0) POCZ0 = LAMB*0.9
0117                IF(N.EQ.0) GO TO 50
0118
0119          C     TEST OF THE FINISH OF THE EXTERNAL ITERATION
0120                IF(ABS(LAMB-LAMPZ).GE.0.5 ) GO TO 30
0121                GO TO 50
0122
0123          30    N=N+1
0124                POCZ = LAMB*0.9
0125                IF(N.EQ.1) POCZ = LAMB*0.5
0126                IF(POCZ.LE.VSIGA) POCZ = VSIGA+1.
0127                LAMPZ=LAMB
0128
0129          C     FOLLOWING APPROXIMATION OF ALFA AND W VALUES
0130                WP=W
0131                DDP=DD
0132          40    R2=R2G+DD
0133                H2=H2G+2*DD
0134                W=COS(ALFA*(H2-H1)/2-.5*PI*H1/H2))/(2*(PI/H2+ALFA))
0135               **COS(ALFA*(H2-H1)/2+.5*PI*H1/H2))/(2*(ALFA-PI/H2))
0136                X=R1/R2*J0
0137                CALL F4J0V0(X,VA,VV)
0138                VY=WP*R2
0139                CALL F4J0V0(YY,VJ,VY)
0140                S=VJ/VY
```

```
0141                    XX=WP*R1
0142                    CALL F4J0Y0(XX,VJ0,VY0)
0143                    CALL F4J1Y1(XX,VJ1,VY1)
0144                    CALL F4J1Y1(X,VC,YC)
0145                    C4A=XX*V4*(VJ1-S*VY1)/(VJ0-S*VY0)-A*VC
0146                    C4B=1./((J0/R2)**2-WP**2)
0147                    C4C=PI/SIN(ALFA*0.5*(H1-H2))
0148                    C4=C4B*C4A *M*C4C
0149                    C3=C4C*VC*M*R1*R2/J0
0150                    C4=C4B*C4A*H2*SIN(0.5*PI*H1/H2)
0151                    C1=R1*P2*H2/J0*SIN(0.5*PI*H1/H2)*VC
0152                    K=1.+C1/(C2+C3+C4)
0153                    IF(K.GE.0.)GO TO 41
0154                    WRITE(3,103)
0155          103       FORMAT(17HCOEFFICIENT K < 0)
0156                    GOTO 60
0157          41        ALFA=(PI/H2)*SQRT(K)
0158                    W=(J0/R2)*SQRT(K)
0159
0160                    R22=ALFA**2+W**2
0161                    GO TO 20
0162
0163          C         CALCULATION OF THE ADDITIONAL VALUES
0164          50        DK=D0*V1KP
0165                    CW=1./(2*R2G)
0166                    DR=2.28*DY/(1.+2.28*DK*CW)
0167                    DH=2.28*DK
0168                    R2STAT=R2G+DR
0169                    H2STAT=H2G+DH
0170                    P6EOM=(1./H2G)**2+(1./H2G)**2
0171                    B22STAT=(J0/R2STAT)**2+(PI/H2STAT)**2
0172                    ODWR=1./H2G**2
0173
```

```
0174          C      PRINT OF THE OUTPUT DATA
0175                 IF(D.EQ.B1.AND.MM22.EQ.1)GOTO 510
0176                   IF(D.EQ.B1.OR.MM22.EQ.1)GOTO 510
0177                 WRITE(3,208)R2G,H2G,ODWR,R2,H2,R2STAT,H2STAT,PGEOM,B22STAT,LAMB
0178          208    FORMAT(10X,F9.2,3X,F5.2,3X,F7.5,3X,F6.3,3X,3(F9.3,3X),2(F7.4,3X ),
                    *F9.1)
0179                 GOTO 60
0180
0181          500    WRITE(3,209)R2G,H2G,ODWR,R2,H2,R2STAT,H2STAT,PGEOM,B22STAT,LAMB
0182          209    FORMAT(3X,1HW,3X,1HR,5X,F9.2,3X,F5.2,3X,F7.5,3X,F6.3,3X,3(F9.3,3X)
0183                 *,2(F9.4,3X),F9.1)
0184                 GOTO 60
0185          510    IF(D.EQ.B1)GOTO 520
0186                 WRITE(3,210)R2G,H2G,ODWR,R2,H2,R2STAT,H2STAT,PGEOM,B22STAT,LAMB
0187          210    FORMAT(5X,1HR,4X,       F9.2,3X,F5.2,3X,F7.5,3X,F6.3,3X,3(F9.3,3X)
0188                 *,2(F9.4,3X),F9.1)
0189                 GOTO 60
0190          520    WRITE(3,211)R2G,H2G,ODWR,R2,H2,R2STAT,H2STAT,PGEOM,B22STAT,LAMB
0191          211    FORMAT(5X,1HW,4X,       F9.2,3X,F5.2,3X,F7.5,3X,F6.3,3X,3(F9.3,3X)
0192                 *,2(F9.4,3X),F9.1)
0193
0194          C      NEXT POINT
0195          60     R2G=R2G-DELTAR
0196                 H2G=H2G-DELTAH
0197                 POCZ = POCZ0
0198                 IF(POCZ.LE.VSIGA) POCZ = VSIGA+1.
0199
0200          C      NEXT DATA
0201                 IF(R2G.LT.R2GPOCZ.OR.H2G.LT.H2GPOCZ) GOTO 99
0202                 GO TO 9
0203                 END
```

```
0204                FUNCTION BESSEL(WO)
0205
0206          C     CALCULATION OF THE VALUE OF THE EXPRESSION (34) FOR THE WO VALUE
0207          C     (CF. DESCRIPTION)
0208
0209                COMMON/TWO/R1,R2/ONE/M
0210                M=0
0211                X1=WO*R1
0212                X2=WO*R2
0213                CALL E4J1Y1(X1,VALUE,YVALUE)
0214                BESSEL=VALUE/YVALUE
0215                CALL E4J0Y0(X2,VALUE,YVALUE)
0216                BESSEL=BESSEL-VALUE/YVALUE
0217                RETURN
0218                END

0219                SUBROUTINE PRZEDZ(F,E,A1,B1,A,B)
0220
0221          C     DETERMINATION OF THE INTERVAL (A,B) WHERE THE ROOT
0222          C     OF THE FUNCTION F WILL BE FOUND
0223
0224          C     A1 - INPUT PARAMETER - LOWER LIMIT OF THIS INTERVAL
0225          C     B1 - INPUT PARAMETER - UPPER LIMIT OF THIS INTERVAL
0226          C     E  - STEP OF THE INVESTIGATION OF THE INTERVAL (A1,B1)
0227
0228                DIMENSION/ONE/M
0229                  A=A1
0230                B=A+E
0231          1     G=F(A)
0232                IF(M.EQ.1)GO TO 2
0233                IF(F(A)*G.LE.0.) RETURN
0234                A=B
0235                B=A+E
0236                IF(B.GT.B1)GO TO 2
0237                GO TO 1
0238          2     B=0.
0239                RETURN
0240                END
```

```
0241                FUNCTION ZERO(A,B,F,E)
0242
0243          C     CALCULATION OF THE ROOT OF THE FUNCTION F IN THE INTERVAL  (A,B)
0244          C     WITH THE PRECISION E
0245
0246                IF(F(A)*F(B).GT.0.)GO TO 4
0247                X1=A
0248                X2=B
0249          1     F1=F(X1)
0250                F2=F(X2)
0251                IF(F1.EQ.0.)GO TO 2
0252                IF(F2.EQ.0.)GO TO 3
0253                X=(X1+X2)/2.
0254                F0=F(X)
0255                IF(F0*F1.LE.0.)X2=X
0256                IF(F0*F2.LE.0.)X1=X
0257                IF(ABS(X1-X2).GE.E)GO TO 1
0258                ZERO=(X1+X2)/2.
0259                IF(ABS(F1-F2).GT.1.)GO TO 4
0260                RETURN
0261          2     ZERO=X1
0262                RETURN
0263          3     ZERO=X2
0264                RETURN
0265             4  CONTINUE
0266                ZERO=0.
0267                RETURN
0268                END
```

```
0269                FUNCTION VIEL(LAMBDA)
0270
0271          C     CALCULATION OF THE VALUE OF THE EXPRESSION (35)
0272          C     FOR THE LAMBDA VALUE (CF. DESCRIPTION)
0273
0274                REAL LAMBDA,LOK
0275                COMMON/THREE/  VK,V1K,SIGU,SIGA,C,B22,DOK/ONE/M
0276                LOK=V1K*LAMBDA
0277                DOK=V1K*VK*1./(3.*(SIGU-LOK))
0278                SIGAP=SIGA-LOK
0279                A=1.+4.*SIGAP*C*V1K/DOK**2
0280                IF(A.GE.0.)GO TO 1
0281                M=1
0282                WRITE(3,100)M
0283          100   FORMAT(2HM=,I2)
0284                WRITE(3,301)A,SIGAP,V1K,DOK,LAMBDA
0285          301   FORMAT(1X,2HA=,F10.4,6HSIGAP=,F10.4,6HV1K=,F10.4,6HDOK=,F10.4,7HLA
0286               CHBDA=,F10.4)
0287                RETURN
0288          1     VIEL=B22*SIGAP/DOK**2.*(1.+SQRT(A))**(-1)
0289                M=0
0290                RETURN
0291                END


0292                FINISH
```

CYLINDER - THEORETICAL CURVE

	V*SIGMA A(1/S)	DD((CM**2)/S)	C((CM**4)/S)
MODERATOR	4120.0	36945.0	7153.0

THETA=20.0 ST.C

R1G= 2.51 CM
H1G= 5.02 CM

R2G= 3.5 ... 7.0 CM
H2G= 7.0 ... 14.0 CM

	R2G (CM)	H2G (CM)	H2G**-2 (CM**-2)	R2DYN (CM)	H2DYN (CM)	R2STAT (CM)	H2STAT (CM)	PARGEOM (CM**-2)	B**2 STAT (CM**-2)	LAMBDA (1/S)
	5.00	10.00	0.01000	5.367	10.774	5.359	10.575	0.0500	0.4951	20455.3
	4.90	9.80	0.01041	5.290	10.561	5.281	10.175	0.0521	0.5065	21184.3
	4.80	9.60	0.01085	5.196	10.366	5.185	9.977	0.0543	0.5181	22197.4
	4.70	9.40	0.01132	5.098	10.190	5.085	9.750	0.0566	0.5280	23513.1
	4.60	9.20	0.01181	5.003	10.005	4.988	9.555	0.0591	0.5416	24519.6
	4.50	9.00	0.01235	4.908	9.810	4.891	9.357	0.0617	0.5556	25853.7
	4.40	8.80	0.01291	4.814	9.627	4.774	9.191	0.0646	0.5706	27322.3
	4.30	8.60	0.01352	4.720	9.441	4.678	8.995	0.0676	0.5863	28943.8
	4.20	8.40	0.01417	4.628	9.257	4.582	8.800	0.0709	0.6029	30750.1
	4.10	8.20	0.01487	4.538	9.073	4.487	8.606	0.0744	0.6205	32723.3
	4.00	8.00	0.01562	4.448	8.897	4.393	8.433	0.0781	0.6392	34927.8
	3.90	7.80	0.01644	4.361	8.724	4.299	8.221	0.0822	0.6590	37383.3
	3.80	7.60	0.01731	4.277	8.553	4.207	8.030	0.0866	0.6799	40124.4
B	3.70	7.40	0.01826	4.195	8.390	4.119	7.838	0.0913	0.7021	43180.1
B	3.60	7.20	0.01929	4.117	8.235	4.045	7.692	0.0963	0.7255	46656.4
B	3.50	7.00	0.02041	4.044	8.089	3.956	7.495	0.1020	0.7504	50520.7

REFERENCES

Drozdowicz K. and Woźnicka U. (1982), INP Report No 1179/AP

ELWRO (1971), "PORTRAN - library of subprograms for the ODRA computers of series 1300", Publ. ELWRO No 13044/5, Wrocław, Poland (in Polish)

Woźnicka U. (1981), J.Phys.D 14 1167