Fortran IV program to calculate Z-transfer functions for the calculation of transient heat transfer through walls and roofs

Mitalas, G. P.; Arseneault, J. G.

Publisher's version / Version de l'éditeur:
https://doi.org/10.4224/20328318
DBR Computer Program, 1972-06

HA

National Research Council Canada
Conseil national de recherches Canada

10/34

DIVISION OF BUILDING RESEARCH

FORTRAN IV PROGRAM TO CALCULATE z-TRANSFER FUNCTIONS FOR THE CALCULATION OF TRANSIENT HEAT TRANSFER THROUGH WALLS AND ROOFS

BY

G.P. MITALAS AND J.G. ARSENEAULT

DBR COMPUTER PROGRAM NO. 33

OTTAWA

JUNE 1972

CP 33

This publication series has been initiated by the Division of Building Research as a convenience in the listing and exchange of computer programs which are developed in the course of its work. Programs submitted by the Division to user groups and available elsewhere will be listed in the series as well as those of less general interest which will be available only from the Division. A list of all programs in the series will be made available on request. Copies of the program tapes or cards are also available for some of the programs in this series.

NATIONAL RESEARCH COUNCIL OF CANADA
DIVISION OF BUILDING RESEARCH

FORTRAN IV PROGRAM TO CALCULATE z-TRANSFER FUNCTIONS FOR THE CALCULATION OF TRANSIENT HEAT TRANSFER THROUGH WALLS AND ROOFS

by

G. P. Mitalas and J. G. Arseneault

Computer Program No. 33
of the
Division of Building Research

OTTAWA
June 1972

Fortran IV Program to Calculate z-Transfer Functions for the Calculation of Transient Heat Transfer through Walls and Roofs

G. P. Mitalas and J. G. Arseneault[1]

Division of Building Research
National Research Council of Canada
Ottawa

The heat transmission matrix for a wall or roof has elements A, B, C and D; i.e.,

$$\begin{vmatrix} \theta_o \\ Q_o \end{vmatrix} = \begin{vmatrix} A & B \\ C & D \end{vmatrix} \cdot \begin{vmatrix} \theta_i \\ Q_i \end{vmatrix}$$

where θ = Laplace transform of surface temperature, and

Q = Laplace transform of surface flux

The elements A, B, C and D are functions of the thermal properties, thickness and position of materials in the wall. When the boundary conditions are of the first kind (i.e. when θ_o and θ_i are given), the fluxes are given by

$$\begin{vmatrix} Q_o \\ Q_i \end{vmatrix} = \frac{1}{B} \begin{vmatrix} D\,, & -1 \\ 1\,, & -A \end{vmatrix} \cdot \begin{vmatrix} \theta_o \\ \theta_i \end{vmatrix} \quad ;$$

and when boundary conditions are of the second kind, the equations invert to

$$\begin{vmatrix} \theta_o \\ \theta_i \end{vmatrix} = \frac{1}{C} \begin{vmatrix} A\,, & -1 \\ 1\,, & -D \end{vmatrix} \cdot \begin{vmatrix} Q_o \\ Q_i \end{vmatrix}$$

The program presented in this paper evaluates the coefficients of a set of z-transfer functions that are equivalent to the Laplace transfer functions D/B, 1/B, A/B, A/C, 1/C and D/C. These z-transfer functions relate to the z-transforms of the surface temperatures and heat fluxes in the same way as their counterpart Laplace transfer functions relate to the expressions above.

[1] Research Officer and Computer Systems Programmer, respectively.

The program will evaluate z-transfer functions that are exact for either a unit step input, a ramp type input or a periodic input with specified harmonic components. The user can choose, therefore, the option that best suits a particular problem.

Key Words: Frequency response, roofs, transient heat conduction, walls, z-transforms.

The heat transmission matrix for a wall or roof has elements A, B, C and D, i.e.,

$$\begin{vmatrix} \theta_o \\ Q_o \end{vmatrix} = \begin{vmatrix} A & B \\ C & D \end{vmatrix} \cdot \begin{vmatrix} \theta_i \\ Q_i \end{vmatrix}$$

where θ = Laplace transform of surface temperature

Q = Laplace transform of surface flux

The elements A, B, C and D are functions of the Laplace parameter, s, and of the thermal properties, thickness and position of materials in the wall. When the boundary conditions are of the first kind (i.e. when θ_o and θ_i are given), the fluxes are given by

$$\begin{vmatrix} Q_o \\ Q_i \end{vmatrix} = \frac{1}{B} \begin{vmatrix} D\,, & -1 \\ 1\,, & -A \end{vmatrix} \cdot \begin{vmatrix} \theta_o \\ \theta_i \end{vmatrix} ;$$

and when boundary conditions are of the second kind, the equations invert to

$$\begin{vmatrix} \theta_o \\ \theta_i \end{vmatrix} = \frac{1}{C} \begin{vmatrix} A\,, & -1 \\ 1\,, & -D \end{vmatrix} \cdot \begin{vmatrix} Q_o \\ Q_i \end{vmatrix}$$

The program presented in this paper evaluates the coefficients of a set of z-transfer functions that are equivalent to the Laplace transfer functions D/B, 1/B, A/B, A/C, 1/C and D/C. These z-transfer functions relate to the z-transforms of the surface temperatures and heat fluxes in the same way as their counterpart Laplace transfer functions relate to the expressions above.

The program will evaluate z-transfer functions that are exact for either a unit step input, a ramp type input or a periodic input with specified harmonic components. The user can choose, therefore, the option that best suits a particular problem.

1. Calculations of z-Transfer Functions [Ref. 1] [2]

The z-transfer functions for a multilayer wall can be calculated in two ways:

<u>Method 1</u> consists of choosing either a step or a ramp input function, I(z), and evaluating the output, 0(z), that corresponds to $1/s$ or $1/s^2$ times one of the Laplace transfer functions. The related z-transfer function is determined from 0(z)/I(z).

---

[2] The literature reference is at the end of the main text of this paper.

Method 2 involves solving a set of simultaneous linear algebraic equations to obtain the coefficients of a z-transfer function whose frequency response matches the exact frequency response of the multilayer slab at certain selected frequencies.

The z-transfer function corresponding to any one of the Laplace transfer functions can be expressed as the ratio of two finite polynomials, N(z)/D(z). The denominator, D(z), is the same for all the transfer functions that have a common denominator in their Laplace transfer function equivalents, and is the same for Method 1 and Method 2. The procedure for finding the coefficients of the denominator polynomial involves two steps.

(1) Determination of the poles of the associated Laplace transfer function:

i.e., find β_n, the roots of B = 0;

or γ_n, the roots of C = 0;

The elements of the transmission matrix for a wall have an infinite set of roots, which lie along the negative real axis in the s-plane. The position of the roots depends on the dimensions and thermal properties of all the layers, and cannot be expressed in any simple way. The necessary poles can be found numerically, however, by evaluating the functions B or C for a sequence of negative real values of s. This program evaluates the roots of B between zero and $-30/\Delta$, and the roots of C between zero and $-450/\Delta$, where Δ is the specified sampling interval of the z-transform.

(2) The evaluation of the product:

$$D(z) = \Pi\,(1 - e^{-\beta_n \Delta} z^{-1})$$

when the parent Laplace transfer function has the element B in the denominator, or

$$D(z) = (1 - z^{-1})\ \Pi\,(1 - e^{-\gamma_n \Delta} z^{-1})$$

when Laplace transfer function has C in the denominator.

Methods 1 and 2 differ only in the way the numerator polynomial is determined. Method 1 requires the evaluation of the time function that corresponds to $1/s$ (step input) or to $1/s^2$ (ramp input) times the appropriate Laplace transfer function, for $t = \Delta, 2\Delta, 3\Delta, \ldots\ldots$ The coefficients of 0(z) are evaluated by finding the residues of the Laplace transfer function at the previously determined poles.

The numerator N(z) is then evaluated using the expression

$$N(z) = \frac{D(z) \quad . \quad 0(z)}{I(z)}$$

where $$I(z) = \frac{1}{1 - z^{-1}} \quad \text{for a step input}$$

or $$I(z) = \frac{\Delta}{z(1 - z^{-1})^2} \quad \text{for a ramp input.}$$

Method 2 requires the evaluation of the Laplace transfer function of the wall at $s = i\omega_n$, and the calculation of the denominator D(z) at $z = e^{i\omega_n \Delta}$, where ω_n is the angular velocity at which the z-transfer function is to match the exact frequency response. This gives a pair of equations for each value of ω_n (i.e. real and imaginary parts are equated separately) except at $\omega_n = 0.0$ (i.e. steady state) where only the real part of the equation exists. The resulting set of equations for a series of values of ω_n can be expressed in matrix form, viz.:

$$\begin{bmatrix} 1 & 1 & 1 & & 1 \\ 1 & \mathrm{Cos}\,\omega_1\Delta & \mathrm{Cos}\,2\omega_1\Delta & \cdots & \mathrm{Cos}\,J\omega_1\Delta \\ 0 & \mathrm{Sin}\,\omega_1\Delta & \mathrm{Sin}\,2\omega_1\Delta & \cdots & \mathrm{Sin}\,J\omega_1\Delta \\ 1 & \mathrm{Cos}\,\omega_2\Delta & \mathrm{Cos}\,2\omega_2\Delta & \cdots & \mathrm{Cos}\,J\omega_2\Delta \\ 0 & \mathrm{Sin}\,\omega_2\Delta & \mathrm{Sin}\,2\omega_2\Delta & \cdots & \mathrm{Sin}\,J\omega_2\Delta \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ a_J \end{bmatrix} = \begin{bmatrix} X(0) \\ X(\omega_1) \\ Y(\omega_1) \\ X(\omega_2) \\ Y(\omega_2) \\ \cdot \\ \cdot \\ \cdot \end{bmatrix}$$

where the a's are the unknown coefficients of the N(z) polynomial and $X(\omega_n)$ and $Y(\omega_n)$ are real and imaginary parts of the product of the Laplace transfer function and denominator D(z) evaluated at $s = i\omega_n$ and $z = e^{i\omega_n \Delta}$ respectively. The solution of this matrix equation gives the unknown coefficients. It should be noted that in setting up this matrix, $\omega_n > \frac{2\pi}{3\Delta}$ should not be used since higher frequencies than this tend to give poorer results.

2. General Description of the Program

This Fortran IV program is designed for an IBM-360 computer with line printer. Appendix A consists of the coding sheets (A-1 to A-20), a sample of output (A-21 to A-25), and the flow diagrams (A-26 to A-31) for this program.

The program can handle slabs that are comprised of no more than 20 layers of homogeneous material and no more than 100 significant poles. The poles are evaluated to 10^{-14} precision, and the limit for the numerator and denominator series of the z-transform is set at 10^{-7}. At least one of the layers of the composite slab must have significant heat-storage capacity.

The program is designed to operate continuously; i.e., after the z-transforms for one wall have been completed, the program automatically reads the data for the following calculation. The program terminates when $\Delta = 0.0$ is read.

2.1 Input

Card 1	Sampling time interval Δ Format: (F 10.3)
Card 2 and 3	Description of the slab for title purpose only. Format: (80 A1)
Cards 4 to I	Groups of cards giving thermal properties, thickness, and description of the layers. Whenever applicable, the first card of the group contains values of thickness of layer, thermal conductivity, density,

specific heat, and
resistance of radiation path.

Otherwise, the first card contains the thermal resistance of a layer that has negligible heat storage capacity.
Format: (5F 10.4)
The second, third.... or more cards of the group can be used for the description of the layer if an integer is inserted in Column One.
Format: (30 A1)

Card I + 1 — Blank card to terminate the above input of thermal properties and their descriptions.

Card I + 2 — Code number, ICASE, and the number of frequencies, NW, to be fitted when Method 2 is to be used (see Table 1).
Format: (I1, I1)

Card I + 3 — This card is read only when ICASE = 2 or 5. It specifies the periods of the harmonics to be used in frequency response calculations.

Table 1. Code Number ICASE

Input Function / Boundary Condition	Method 1		Method 2
	Square Pulse	Triangle Pulse	Group of Harmonics
First Kind	Invalid Combination	1	2
Second Kind	3	4	5

3. Reference

[1] Stephenson, D.G. and Mitalas, G.P., Calculation of heat conduction transfer functions for multilayer slabs. Submitted to ASHRAE for presentation January 1971.

4. Acknowledgement

The authors gratefully acknowledge the many helpful suggestions made by Dr. D.G. Stephenson.

This paper is a contribution of the Division of Building Research, National Research Council of Canada, and is published with the approval of the Director of the Division.

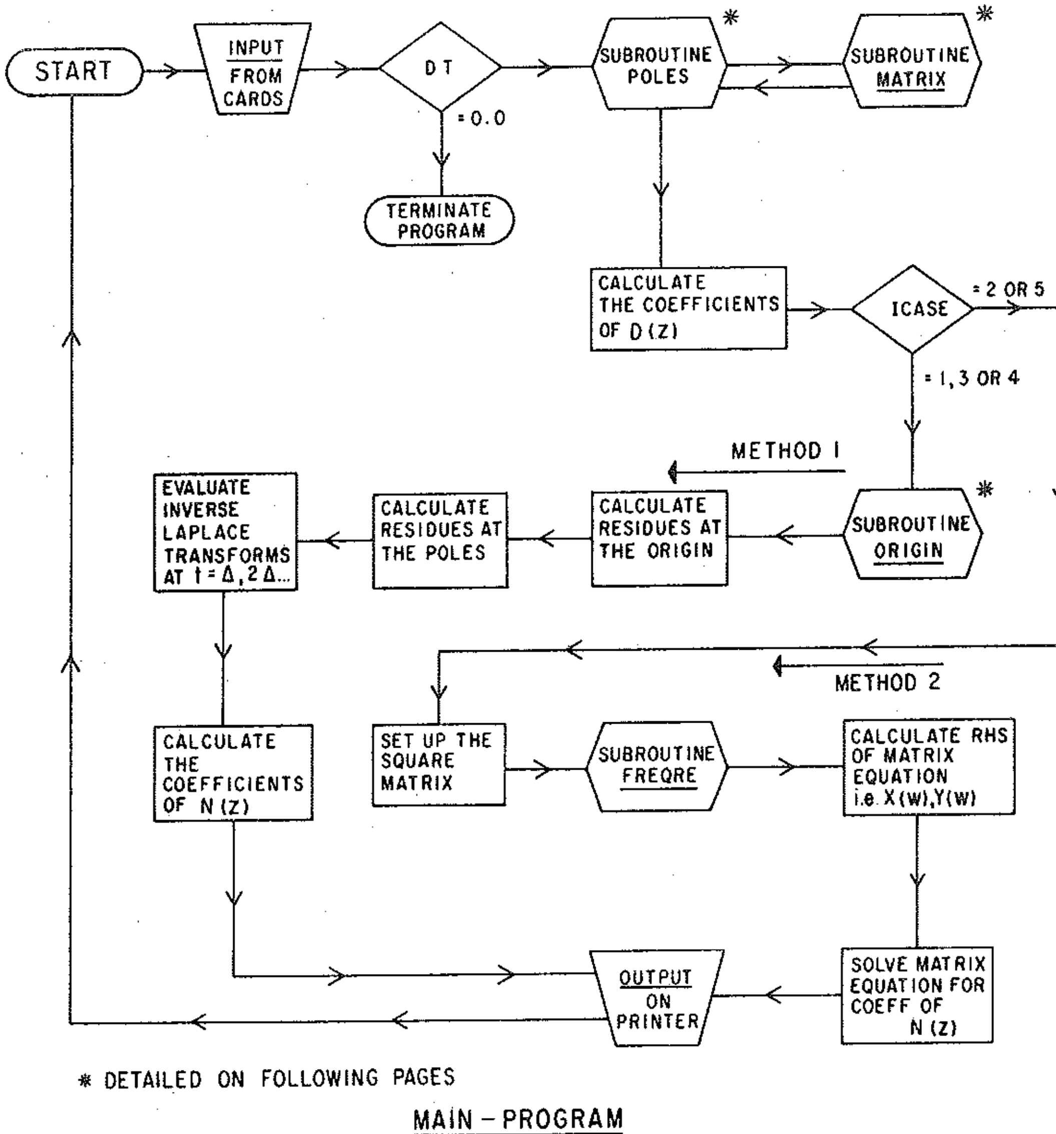
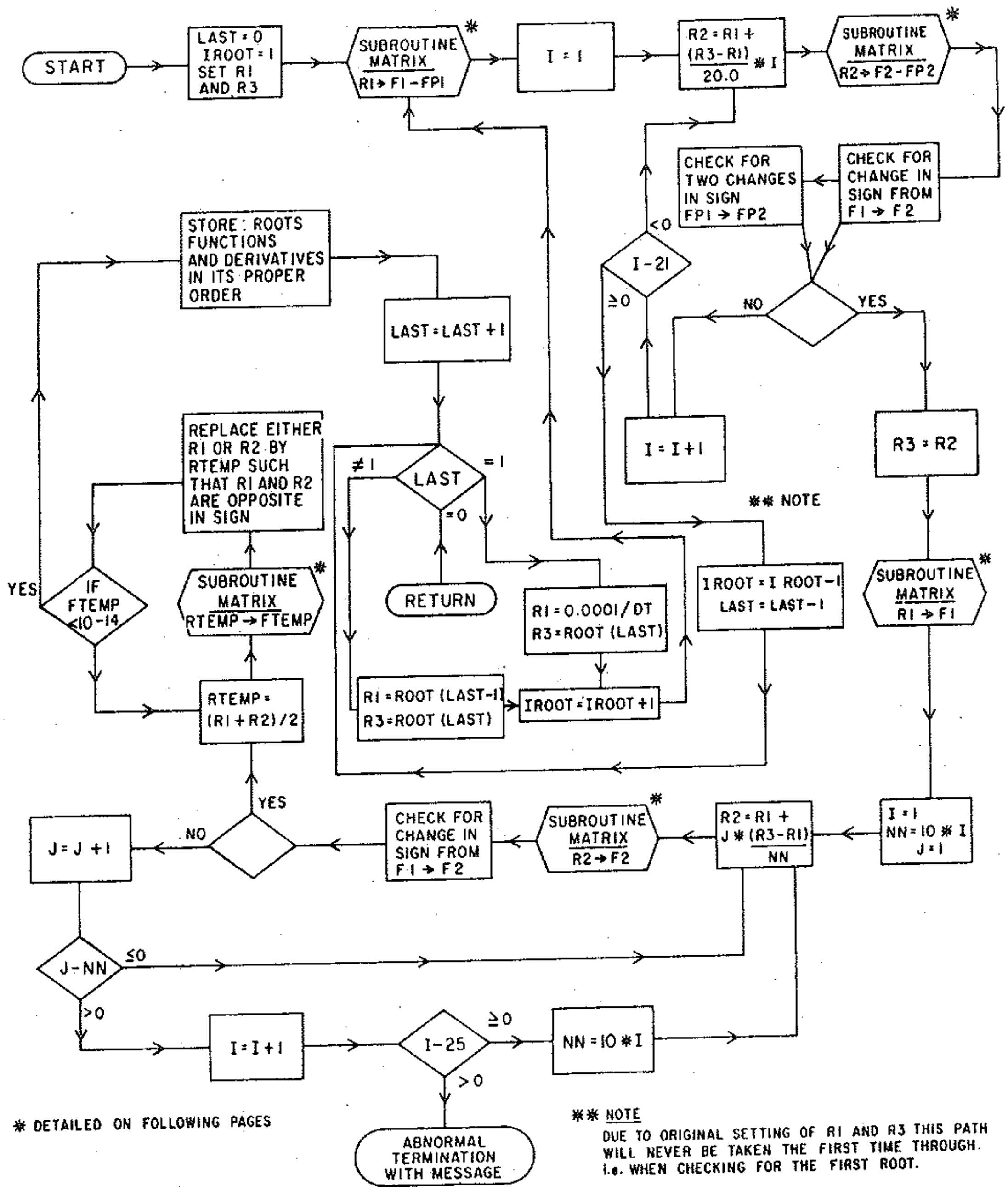
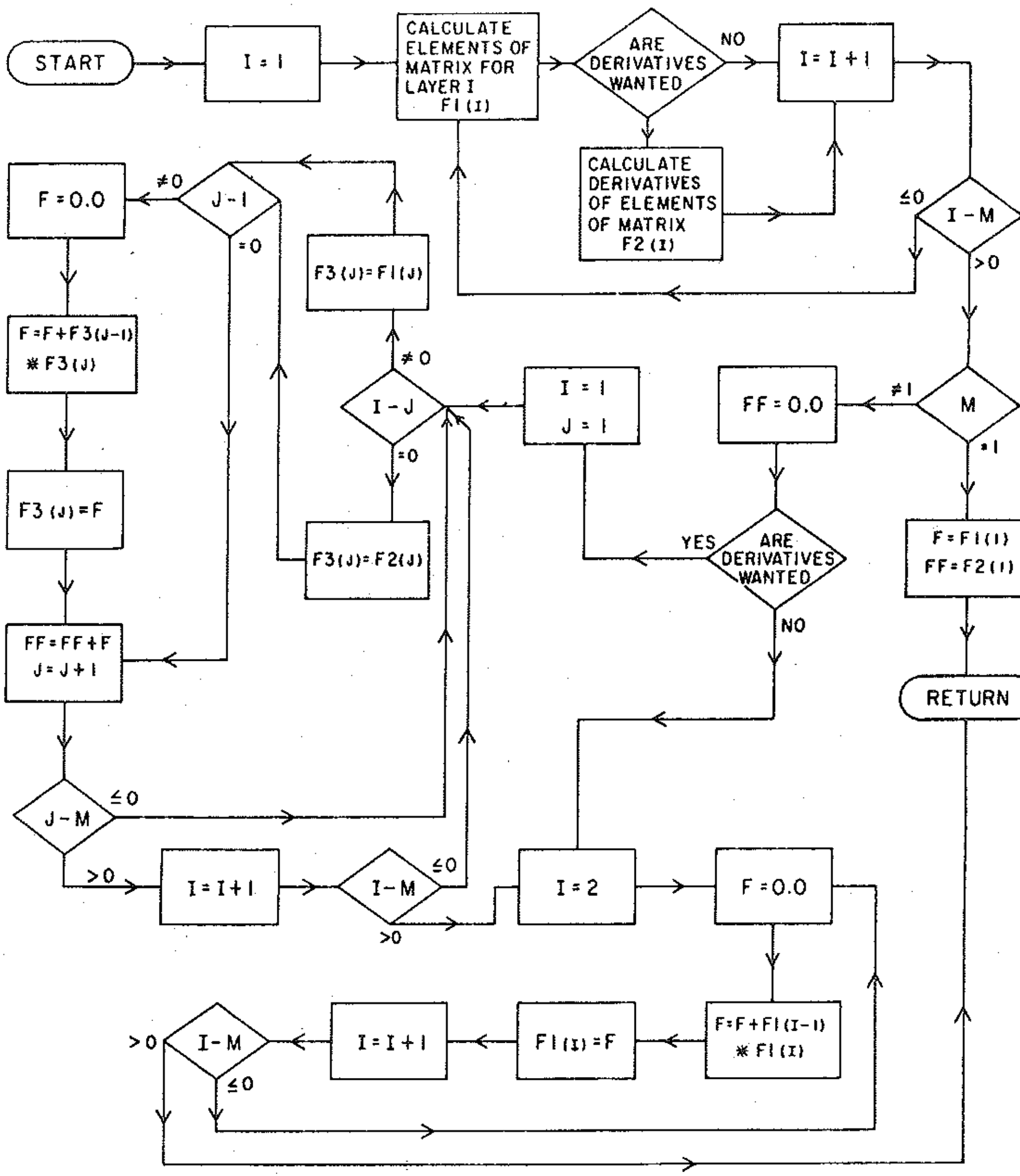


Figure A-1. Flow diagram for z-transfer function calculation program.



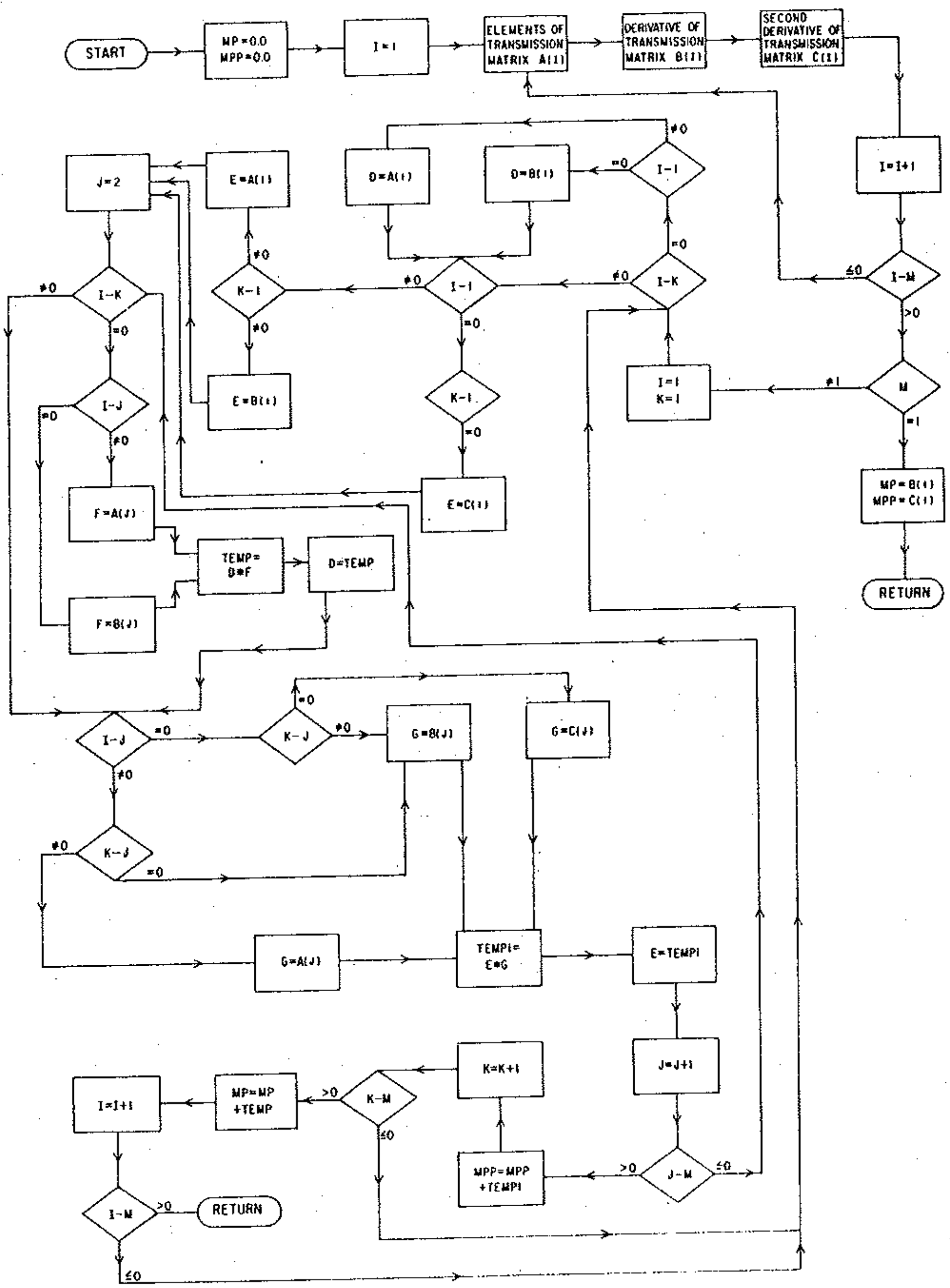
SUBROUTINE – POLES

Figure A-2. Flow diagram for determination of poles of Laplace transfer function and calculation of the values of A, B, C, D, and their derivatives at these poles.



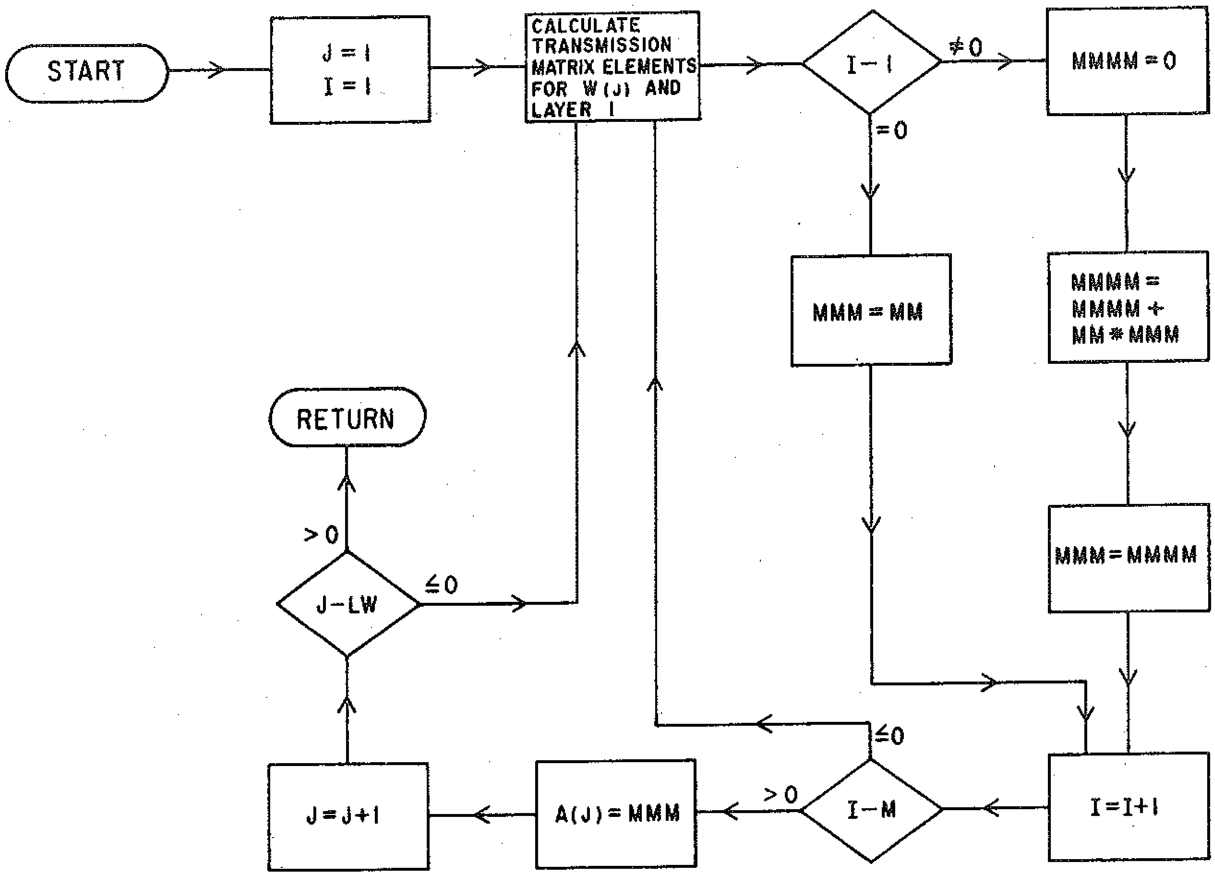
SUBROUTINE - MATRIX

Figure A-3. Flow diagram for the evaluation of A, B, C, D, and their derivatives for any real negative value of s.



SUBRROUTINE – ORIGIN

Figure A-4. Flow diagram for the evaluation of first and second derivative of A, B, C and D at $s = o$.



SUBROUTINE - FREQUE

Figure A-5. Flow diagram for the evaluation of A, B, C and D for pure imaginary arguments.

```
    FORTRAN PROGRAM TO EVALUATE Z-TRANSFORMS FOR CALCULATION OF
TRANSIENT HEAT TRANSFERS THROUGH WALLS AND ROOFS.

    THIS PROGRAM WILL DERIVE THE Z-TRANSFER FUNCTIONS FOR TWO
TYPES OF BOUNDARY CONDITIONS AND THE FORM OF BOUNDARY PARAMETERS
MUST BE SPECIFIED.

BOUNDARY CONDITIONS:
    OF THE FIRST KIND; (TEMPERATURE GIVEN FOR BOTH SURFACES).
A) RAMP INPUT                ICASE=1
B) FREQUENCY RESPONSE        ICASE=2

    OF THE SECOND KIND; (FLUX GIVEN FOR BOTH SURFACES).
A) STEP INPUT                ICASE=3
B) RAMP INPUT                ICASE=4
C) FREQUENCY RESPONSE        ICASE=5

    INPUT TO PROGRAM:
CARD(1)       DT  (F10.3)     DT=SAMPLING TIME INTERVAL.
CARD(2)*
       * DESCRIPTION OF THE SLAB FOR TITLE PURPOSE ONLY (80A1).
CARD(3)*

CARD(4)   *
  "       *
CARD(I+3)* XL(I),XK(I),D(I),SH(I),RES(I),(TEXT(I,J),J=1,30) WHERE
  "       * I INDICATES THE I'TH LAYER OF THE SLAB (5F10.4,30A1).
  "       *
CARD(M+3)*
    WHERE XL=THICKNESS OF LAYER.
          XK=THERMAL CONDUCTIVITY.
           D=DENSITY.
          SH=SPECIFIC HEAT.
         RES=RESISTANCE OF RADIATION PATH WHENEVER APPLICABLE
             OR THERMAL RESISTANCE OF LAYER WHEN THERE IS
             NEGLIGIBLE HEAT STORAGE.
        TEXT=DESCRIPTION OF LAYER, A SECOND CARD AND SO ON CAN BE
             USED BY INSERTING ANY INTEGER IN COLUMN ONE(1).
           M=NUMBER OF LAYERS THE SLAB IS COMPOSED OF.
CARD(M+4)    BLANK CARD TO STOP ABOVE INPUT.
CARD(M+5)    ICASE,NW (I1,I1)
             WHERE:  NW=NUMBER OF FREQUENCIES TO BE USED WHEN
                   FREQUENCY RESPONSE IS INVOLVED.
CARD(M+6)      W(2),W(3)...............W(NW+1)  (8F10.4)
               ABOVE CARD ONLY READ WHEN FREQUENCY RESPONSE IS
               INVOLVED. (ICASE=2 OR ICASE=5) W(1) IS SET TO 0.0
               W(I)'S ARE THE PERIODS.

NOMENCLATURE:
    RR=THICKNESS/THERMAL CONDUCTIVITY (XL/XK)
       OR THERMAL RESISTANCE OF LAYER WHEN THERE IS
       NEGLIGIBLE HEAT STORAGE. THEN RES=0.0
    BETA*BETA=XL*XL*D*SH/XK
    CO=THERMAL CONDUCTANCE AND USED AS 1/C'AT THE POLE FOR ICASE=4.
```

```
            AFTER A RUN IS COMPLETED CONTROL RETURNS TO READING CARD(1)
            THEREFORE A BLANK CARD IN THIS LOCATION TERMINATES THE PROGRAM.


      DOUBLE PRECISION RR(20),BETA(20),XL(20),XK(20),D(20),SH(20),
     9RES(20),ROOT(100),FUNC(100,4),DER(100,4),DN(100,3),MP(2,2),
     8MPP(2,2),POL1(100),POL2(100),POL3(100),POL4(100),POL5(100),
     7POL6(100),MX(11,11),X(11),Y(11),Z(11),TEM1(2),TEM2(2),TEM3(2),
     6DT,C0,C1X,C1Y,C1Z,C2X,C2Y,C2Z,T,ARG1,ARG2,PREC,DET,TEST,W(6)
      COMPLEX*16 A(6,2,2),TEMP,TEMP1,TEMP2,TEMP3
      INTEGER CARD,PRINT,TEXT(20,30),TEXT1(2,80)
      EQUIVALENCE (TEMP1,TEM1(1)),(TEMP2,TEM2(1)),(TEMP3,TEM3(1))
      CARD=1
      PRINT=3
      PREC=0.0
      W(1)=0.0
 1000 READ(CARD,1) DT
    1 FORMAT(   F10.3)
      IF(DT.EQ.0.0) GO TO 2000
      READ(CARD,2) ((TEXT1(I,J),J=1,80),I=1,2)
    2 FORMAT(   80A1)
      WRITE(PRINT,3) ((TEXT1(I,J),J=1,80),I=1,2)
    3 FORMAT(1H1,26X,80A1/27X,80A1)
      WRITE(PRINT,4)
    4 FORMAT(1H0,'LAYER',1X,'THICKNESS',4X,'CONDUCTIVITY',8X,'DENSITY',
     18X,'SP HEAT',11X,'RESISTANCE')
      WRITE(PRINT,5)
    5 FORMAT(2X,'    ',5X,'  ',4X,'                    ',3X,'               '
     1,2X,'                ',2X,'                    ',4X,'DESCRIPTION
     2OF LAYER'/)
      C0=0.0
      I=0
      DO 10 N=1,40
      I=I+1
      M=I
      READ(CARD,6) XL(I),XK(I),D(I),SH(I),RES(I),(TEXT(I,J),J=1,30)
    6 FORMAT(   5F10.4,30A1)
      IF(RES(I).EQ.0.0.AND.SH(I).EQ.0.0.AND.XL(I).NE.0.0) GO TO 20
      IF(RES(I).EQ.0.0.AND.XL(I).EQ.0.0) GO TO 30
      IF(XL(I).NE.0.0) GO TO 40
      RR(I)=RES(I)
      BETA(I)=0.0
      GO TO 50
   40 RR(I)=XL(I)/XK(I)
      BETA(I)=XL(I)*DSQRT(D(I)*SH(I)/XK(I))
      C0=C0+RES(I)
   50 C0=C0+RR(I)
      WRITE(PRINT,7) I,XL(I),XK(I),D(I),SH(I),RES(I),(TEXT(I,J),J=1,30)
    7 FORMAT(1X,I3,F10.4,F15.4,F18.4,F15.4,F20.4,13X,30A1)
      GO TO 10
   20 WRITE(PRINT,8) (TEXT(I,J),J=1,30)
    8 FORMAT(95X,30A1)
      I=I-1
   10 CONTINUE
```

```
      M=21
   30 M=M-1
      DO 60 I=1,M
      IF(XL(I).NE.0.0) GO TO 60
      RES(I)=0.0
   60 CONTINUE
      C0=1.0/C0
      WRITE(PRINT,1)
      WRITE(PRINT,1)
      WRITE(PRINT,9) C0
    9 FORMAT(30X,'THERMAL CONDUCTANCE,   U=',F6.3,2X,'
     1     ')
      WRITE(PRINT,1)
      WRITE(PRINT,11) DT
   11 FORMAT(39X,'SAMPLING TIME INTERVAL,  DT=',F9.3,'    ')
      WRITE(PRINT,1)
      READ(CARD,12) ICASE,NW
   12 FORMAT(    I1,I1)
      NW=NW+1
      MW=NW*2-1
      IF(ICASE.NE.2.AND.ICASE.NE.5) GO TO 65
      READ(CARD,13) (W(I),I=2,NW)
   13 FORMAT(   8F10.4)
      DO 61 I=2,NW
   61 W(I)=2.0*3.14159265/W(I)
   65 IF(ICASE.LT.3) GO TO 70
      IX=2
      JX=1
      GO TO 80
   70 IX=1
      JX=2
   80 CALL POLES(RR,BETA,RES,ROOT,DER,FUNC,M,IX,JX,DT,IROOT,ICASE)
      DO 90 I=1,100
      POL1(I)=0.0
      POL2(I)=0.0
      POL3(I)=0.0
      POL4(I)=0.0
      POL5(I)=0.0
   90 POL6(I)=0.0
      MMM=1
      NNN=0
      POL1(1)=1.0
      DO 100 I=1,IROOT
      POL2(1)=1.0
      POL2(2)=-DEXP(-ROOT(I)*DT)
      IF(DABS(POL2(2)).LT.1.0D-16)GO TO 110
      CALL POLYM(POL1,POL2,NNN,MMM)
  100 CONTINUE
  110 IF(ICASE.LT.3) GO TO 120
      POL2(1)=1.0
      POL2(2)=-1.0
      CALL POLYM(POL1,POL2,NNN,MMM)
  120 ID=NNN+1
      IF(ICASE.EQ.2.OR.ICASE.EQ.5) GO TO 130
```

```
      CALL ORIGIN(RR,BETA,RES,M,MP,MPP)
      GO TO(140,140,150,150,150),ICASE
  140 C1Y=-C0*C0*MP(1,2)
      C1X=C1Y+MP(2,2)*C0
      C1Z=C1Y+MP(1,1)*C0
      GO TO 160
  150 C0=1.0/MP(2,1)
      C1Y=-C0*C0*MPP(2,1)/2.0
      C1X=MP(2,2)*C0+C1Y
      C1Z=MP(1,1)*C0+C1Y
  160 C2X=0.0
      C2Y=0.0
      C2Z=0.0
      DO 170 I=1,IROOT
      IF(ICASE.GT.1) GO TO 180
      DN(I,2)=1.0/ROOT(I)/ROOT(I)/DER(I,2)
      GO TO 200
  180 IF(ICASE.EQ.4) GO TO 190
      DN(I,2)=-1.0/ROOT(I)/DER(I,3)
      GO TO 200
  190 DN(I,2)=1.0/ROOT(I)/ROOT(I)/DER(I,3)
  200 DN(I,1)=DN(I,2)*FUNC(I,4)
      DN(I,3)=DN(I,2)*FUNC(I,1)
      IF(ICASE.NE.4) GO TO 170
      C2X=C2X-DN(I,1)
      C2Y=C2Y-DN(I,2)
      C2Z=C2Z-DN(I,3)
  170 CONTINUE
      DO 210 I=1,ID
  210 POL2(I)=POL1(I)
      IF(ICASE.EQ.3) GO TO 220
      POL3(1)=1.0/DT
      POL3(2)=-2.0/DT
      POL3(3)=1.0/DT
      MMM=2
      GO TO 235
  220 POL3(1)=1.0
      POL3(2)=-1.0
      MMM=1
  235 CALL POLYM(POL2,POL3,NNN,MMM)
      POL3(1)=0.0
      POL3(2)=0.0
      POL3(3)=0.0
      DO 230 I=1,100
      II=I
      T=I*DT
      DO 240 J=1,IROOT
      IF(ROOT(J)*T.GE.40.0) GO TO 250
      POL3(I)=POL3(I)+DEXP(-ROOT(J)*T)*DN(J,1)
      POL4(I)=POL4(I)+DEXP(-ROOT(J)*T)*DN(J,2)
      POL5(I)=POL5(I)+DEXP(-ROOT(J)*T)*DN(J,3)
      IF(J.LE.10) GO TO 240
      IF(DABS(DEXP(-ROOT(J)*T)*DN(J,2)).LT.1.0D-16) GO TO 250
  240 CONTINUE
```

```
 250 IF(ICASE.EQ.4) GO TO 260
     POL3(I)=POL3(I)+C0*T+C1X
     POL4(I)=POL4(I)+C0*T+C1Y
     POL5(I)=POL5(I)+C0*T+C1Z
     GO TO 270
 260 POL3(I)=POL3(I)+C0*T*T /2.0+C1X*T+C2X
     POL4(I)=POL4(I)+C0*T*T /2.0+C1Y*T+C2Y
     POL5(I)=POL5(I)+C0*T*T /2.0+C1Z*T+C2Z
 270 IF(I.LE.10) GO TO 230
     IF(DABS(POL4(I)).LT.1.0D-16) GO TO 280
 230 CONTINUE
 280 MMM=II-1
     IF(ICASE.LT.3) GO TO 281
     DO 282 I=1,99
     II=101-I
     POL3(II)=POL3(II-1)
     POL4(II)=POL4(II-1)
 282 POL5(II)=POL5(II-1)
     POL3(1)=0.0
     IF(XL(M).EQ.0.0) POL3(1)=RR(M)
     POL4(1)=0.0
     POL5(1)=0.0
     IF(XL(1).EQ.0.0) POL5(1)=RR(1)
     MMM=MMM+1
     IF(ICASE.EQ.4) POL3(1)=0.0
     IF(ICASE.EQ.4) POL5(1)=0.0
 281 NN=NNN+1
     DO 290 I=1,NN
 290 POL6(I)=POL2(I)
     CALL POLYM(POL6,POL3,NNN,MMM)
     NN1=NNN+1
     DO 300 I=1,NN1
 300 POL3(I)=POL6(I)
     DO 310 I=1,NN
 310 POL6(I)=POL2(I)
     NNN=NN-1
     CALL POLYM(POL6,POL4,NNN,MMM)
     NN2=NNN+1
     DO 320 I=1,NN2
 320 POL4(I)=POL6(I)
     DO 330 I=1,NN
 330 POL6(I)=POL2(I)
     NNN=NN-1
     CALL POLYM(POL6,POL5,NNN,MMM)
     NN3=NNN+1
     DO 340 I=1,NN3
 340 POL5(I)=POL6(I)
     GO TO 350
 130 DO 360 I=1,MW,2
     DO 360 J=1,MW
     IF(I.EQ.1) GO TO 370
     K=(I+1)/2
     MX(I,J)=DSIN((J-1)*DT*W(K))
     GO TO 360
```

```
370 MX(I,J)=1.0
360 CONTINUE
    LW=MW-1
    DO 380 I=2,LW,2
    DO 380 J=1,MW
    K=I/2+1
380 MX(I,J)=DCOS((J-1)*DT*W(K))
381 CALL SCLVD(MX,11,MW,MW,PREC,DET,TEST)
    CALL FREQRE (BR,BETA,RES,XL,XK,D,SH,M,W,A,NW)
    DO 390 I=1,MW,2
    IF(I.EQ.1) GO TO 390
    K=(I+1)/2
    ARG1=0.0
    ARG2=0.0
    DO 410 J=1,ID
    ARG1=ARG1+POL1(J)*DCOS((J-1)*DT*W(K))
410 ARG2=ARG2-POL1(J)*DSIN((J-1)*DT*W(K))
    TEMP=DCMPLX(ARG1,ARG2)
    IF(ICASE.EQ.2) GO TO 420
    TEMP1=TEMP*A(K,2,2)/A(K,2,1)
    TEMP2=TEMP/A(K,2,1)
    TEMP3=TEMP*A(K,1,1)/A(K,2,1)
    GO TO 430
420 TEMP1=TEMP*A(K,2,2)/A(K,1,2)
    TEMP2=TEMP/A(K,1,2)
    TEMP3=TEMP*A(K,1,1)/A(K,1,2)
430 X(I-1)=TEM1(1)
    X(I)=-TEM1(2)
    Y(I-1)=TEM2(1)
    Y(I)=-TEM2(2)
    Z(I-1)=TEM3(1)
    Z(I)=-TEM3(2)
390 CONTINUE
    IF(ICASE.EQ.2) GO TO 391
    X(1)=0.0
    DO 392 I=1,M
392 X(1)=X(1)+SH(I)*D(I)*XL(I)
    X(1)=-DT/X(1)
    Y(1)=0.0
    DO 394 I=1,ID
394 Y(1)=Y(1)+I*POL1(I+1)
    X(1)=X(1)*Y(1)
    Y(1)=X(1)
    Z(1)=X(1)
    GO TO 393
391 ARG1=0.0
    DO 440 J=1,ID
440 ARG1=ARG1+POL1(J)
    TEMP=DCMPLX(ARG1,0.0D+01)
    IF(ICASE.EQ.2) GO TO 450
    TEMP1=TEMP*A(1,2,2)/A(1,2,1)
    TEMP2=TEMP/A(1,2,1)
    TEMP3=TEMP*A(1,1,1)/A(1,2,1)
    GO TO 460
```

```
  450 TEMP1=TEMP*A(1,2,2)/A(1,1,2)
      TEMP2=TEMP/A(1,1,2)
      TEMP3=TEMP*A(1,1,1)/A(1,1,2)
  460 X(1)=TEM1(1)
      Y(1)=TEM2(1)
      Z(1)=TEM3(1)
  393 DO 470 I=1,MW
      POL3(I)=0.0
      POL4(I)=0.0
  470 POL5(I)=0.0
      DO 480 I=1,MW
      DO 480 J=1,MW
      POL3(I)=POL3(I)+MX(I,J)*X(J)
      POL4(I)=POL4(I)+MX(I,J)*Y(J)
  480 POL5(I)=POL5(I)+MX(I,J)*Z(J)
      NN1=MW
      NN2=MW
      NN3=MW
      IF(ICASE.NE.5) GO TO 350
      NN1=NN1+1
      NN2=NN2+1
      NN3=NN3+1
      DO 481 I=1,14
      J=16-I
      POL3(J)=POL3(J-1)
      POL4(J)=POL4(J-1)
  481 POL5(J)=POL5(J-1)
      POL3(1)=0.0
      POL4(1)=0.0
      POL5(1)=0.0
  350 IF(ICASE.EQ.3) GO TO 490
      IF(ICASE.EQ.2.OR.ICASE.EQ.5) GO TO 500
      WRITE(PRINT,14)
   14 FORMAT(44X,'COEFFICIENTS FOR RAMP INPUT')
      GO TO 510
  490 WRITE(PRINT,15)
   15 FORMAT(44X,'COEFFICIENTS FOR STEP INPUT')
      GO TO 510
  500 WRITE(PRINT,16)
   16 FORMAT(44X,'COEFFICIENTS BY FREQUENCY RESPONSE')
      DO 501 I=2,NW
  501 W(I)=2.0*3.14159265/W(I)
      WRITE(PRINT,22) (W(I),I=2,NW)
   22 FORMAT('0',20X,'PERIODS',8F10.1)
  510 WRITE(PRINT,1)
      IF(ICASE.GT.2) GO TO 520
      WRITE(PRINT,17)
   17 FORMAT(14X,'J',18X,'D/B',19X,'1/B',19X,'A/B',17X,'D(Z)')
      GO TO 530
  520 WRITE(PRINT,18)
   18 FORMAT(14X,'J',18X,'D/C',19X,'1/C',19X,'A/C',17X,'D(Z)')
  530 NN=MIN0(NN1,NN2,NN3)
      N=MAX0(NN,ID)
      DO 540 I=1,N
```

```
      J=I-1
      IF(I.LE.ID.AND.I.LE.NN) GO TO 550
      IF(NN.LT.ID) GO TO 570
      GO TO 560
  570 WRITE(PRINT,19) J,POL1(I)
   19 FORMAT(9X,I6,68X,F20.6)
      GO TO 540
  560 WRITE(PRINT,21) J,POL3(I),POL4(I),POL5(I)
   21 FORMAT(9X,I6,F24.6,F22.6,F22.6,F20.6)
      GO TO 540
  550 WRITE(PRINT,21) J,POL3(I),POL4(I),POL5(I),POL1(I)
  540 CONTINUE
      GO TO 1000
 2000 WRITE(PRINT,3)
      CALL EXIT
      STOP
      END
```

```
C      SUBROUTINE POLES(RR,BETA,RES,ROOT,DER,FUNC,M,II,IJ,DT,IROOT)
C
C          SUBROUTINE TO CALCULATE THE ROOTS OF THE HEAT TRANSFER MATRIX
C      AND WILL STORE THE VALUE OF THE FUNCTIONS AND THE FIRST DERIVATIVE
C      AT THE ROOTS.
C
C          THE MAXIMUM NUMBER OF ROOTS THAT CAN BE OBTAINED IS SET
C      AT ONE HUNDRED(100)
C
C          THIS METHOD WILL FIRST FIND A ROOT BETWEEN 30.0/DT AND
C      100.0/DT, BEING ASSUMED THAT A ROOT EXIST IN THIS INTERVAL. THIS
C      ROOT IS ALSO LARGE ENOUGH TO GIVE SUFFICIENT ACCURACY TO EVALUATE
C      THE RESPONSE FACTORS.
C          THE METHOD CHECKS THE INTERVAL BETWEEN THE ORIGIN AND THIS
C      FIRST ROOT AND WHEN ANOTHER ROOT IS FOUND THE INTERVAL NEXT TO BE
C      CHECKED BECOMES THE INTERVAL BETWEEN THIS NEW ROOT AND THE NEXT
C      LARGEST ROOT AND SO ON. WHEN NO ROOT EXIST IN AN INTERVAL THE NEXT
C      SMALLEST INTERVAL IS SELECTED AND SO ON WORKING TOWARDS THE ORIGIN
C      UNTIL ALL ROOTS ARE FOUND.
C          TO CHECK FOR A ROOT THE METHOD SUBDIVIDES THE INTERVALS IN
C      RELATIVELY LARGE SEGMENTS AND CHECKS FOR BOTH A CHANGE IN SIGN OF
C      THE FUNCTION AND FOR TWO CHANGES IN DIRECTION OF THE SLOPE OF THE
C      FUNCTION. IF A ROOT EXIST, BY MAKING THESE TWO CHECKS, IT IS
C      INDICATED SO IN A RELATIVELY SHORT TIME. ONCE IT IS INDICATED THAT
C      A ROOT DOES EXIST IN A CERTAIN SEGMENT OF AN INTERVAL, THIS
C      SEGMENT IS FURTHER SUBDIVIDED AND USING A SIMILAR ROUTINE AS ABOVE
C      EXCEPT CHECKING FOR A CHANGE IN SIGN OF THE FUNCTION ONLY. IF ON
C      THE FIRST PASS A CHANGE IN SIGN IS NOT FOUND THE SEGMENT IS FURTHER
C      SUBDIVIDED INTO EVEN SMALLER PARTS UNTIL A CHANGE IN SIGN DOES
C      OCCUR. ONCE A CHANGE IN SIGN OCCURS THE ROOT IS ARRIVED AT BY
C      SPLITTING THIS INTERVAL SUCCESSIVELY IN HALF USING THE NEW SEGMENT
C      WITH FUNCTION VALUE OF OPPOSITE SIGN UNTIL A ROOT IS REACHED
C      WITHIN AN ACCURACY OF 10-14.
C          THE SPLITTING OF THE SEGMENTS TO ARRIVE AT A ROOT IS USED
C      BECAUSE A RELATIVELY CONSTANT NUMBER OF ITERATIONS ARE REQUIRED
C      TO OBTAIN THE ACCURACY WANTED. IN THE CASE OF THE REGULA FALSI
C      METHOD IT WAS FOUND THAT THE NUMBER OF ITERATIONS VARIED FROM AS
C      LOW AS FIVE (5) TO MORE THAN THREE HUNDRED (300) ITERATIONS. IN THE
C      LONG RUN IT WAS FOUND THAT THE SPLITTING OF THE POINTS REQUIRED
C      LESS RUNNING TIME.
C
C      NOMENCLATURE:
C          RR=THICKNESS/THERMAL CONDUCTIVITY (XL/XK)
C             OR THERMAL RESISTANCE OF LAYER WHEN THERE IS
C             NEGLIGIBLE HEAT STORAGE
C          BETA*BETA=XL*XL*D*SH/XK
C                    WHERE      D=DENSITY.
C                               SH=SPECIFIC HEAT.
C          RES=RESISTANCE OF RADIATION PATH WHENEVER APPLICABLE.
C
C          ROOT=CONTAINS THE ROOTS OF THE HEAT TRANSFER FUNCTIONS
C               ON RETURN.
C          DER=CONTAINS THE DERIVATIVE OF THE HEAT TRANSFER FUNCTIONS
```

```
               C           AT THE ROOTS ON RETURN.
               C        FUNC=CONTAINS THE VALUE OF THE HEAT TRANSFER FUNCTIONS
               C             AT THE ROOTS ON RETURN.
               C
               C        M=NUMBER OF LAYERS THE SLAB IS COMPOSED OF.
               C
               C        II AND IJ ARE THE ROW AND COLUMN SUBSCRIPTS OF THE ELEMENT OF
               C        THE MATRIX FOR WHICH THE ROOT IS FOUND.
               C        II=1*
               C            * BOUNDARY CONDITION OF THE FIRST KIND.
               C        IJ=2*
               C
               C        II=2*
               C            * BOUNDARY CONDITION OF THE SECOND KIND.
               C        IJ=1*
               C
               C        DT=TIME INTERVAL OF SAMPLING
               C
 0001                SUBROUTINE POLES(RR,BETA,RES,ROOT,DER,FUNC,M,II,IJ,DT,IROOT,ICASE)
 0002                DOUBLE PRECISION RR(20),BETA(20),RES(20),ROOT(100),DER(100,4),
                    1FUNC(100,4),F(2,2),FF(2,2),R1,R2,R3,F1,F2,F3,FP1,FP2,FP3,RTEMP,
                    2FTEMP,DT
 0003                DO 10 I=1,100
 0004                DO 20 J=1,4
 0005                FUNC(I,J)=0.0
 0006             20 DER(I,J)=0.0
 0007             10 ROOT(I)=0.0
 0008                LAST=0
 0009                R1=30.0/DT
 0010                R3=100.0/DT
 0011                IF(ICASE.EQ.4) R1=450.0/DT
 0012                IF(ICASE.EQ.4) R3=700.0/DT
 0013                DO 30 IROOT=1,100
 0014                IF(IROOT.EQ.1) GO TO 40
               C     FOLLOWING IS THE ROUTINE CHECKING FOR A CHANGE IN THE SIGN OF THE
               C     FUNCTION AND ALSO FOR TWO(2) CHANGES IN DIRECTION OF THE SLOPE, TO
               C     FIND WHETHER A ROOT EXIST OR NOT.
 0015            300 CALL MATRIX(RR,BETA,RES,R1,M,F,FF,2)
 0016                F1=F(II,IJ)
 0017                FP1=FF(II,IJ)
 0018                IC=0
 0019                DO 50 I=1,20
 0020                R2=R1+(R3-R1)/20.0*I
 0021                CALL MATRIX(RR,BETA,RES,R2,M,F,FF,2)
 0022                F2=F(II,IJ)
 0023                FP2=FF(II,IJ)
 0024                IF(F1.GT.0.0) GO TO 60
 0025                IF(F2.LE.0.0) GO TO 70
 0026                GO TO 80
 0027             60 IF(F2.LE.0.0) GO TO 80
 0028             70 IF(FP1.GT.0.0) GO TO 90
 0029                IF(FP2.GT.0.0) GO TO 100
 0030                GO TO 110
```

```
   90 IF(FP2.GT.0.0) GO TO 110
  100 IC=IC+1
      IF(IC.EQ.2) GO TO 80
  110 F1=F2
      FP1=FP2
   50 CONTINUE
      IROOT=IROOT-1
      LAST=LAST-1
  120 IF(LAST.EQ.0) GO TO 130
      IF(LAST.NE.1) GO TO 125
      R1=0.0001/DT
      GO TO 140
  125 R1=ROOT(LAST-1)
  140 R3=ROOT(LAST)
      R1=R1+0.00001/DT
      R3=R3-0.00001/DT
      GO TO 30
   80 R3=R2
   40 N=1
      CALL MATRIX(RR,BETA,RES,R1,M,F,FF,1)
      F1=F(II,IJ)
      DO 150 I=N,25
      NN=10*I
      DO 160 J=1,NN
      R2=R1+J*(R3-R1)/NN
      CALL MATRIX(RR,BETA,RES,R2,M,F,FF,1)
      F2= F(II,IJ)
      IF(F1.GT.0.0) GO TO 170
      IF(F2.LE.0.0) GO TO 160
      GO TO 190
  170 IF(F2.LE.0.0) GO TO 190
      GO TO 160
  190 RTEMP=(R1+R2)/2.0
      CALL MATRIX(RR,BETA,RES,RTEMP,M,F,FF,1)
      FTEMP=F(II,IJ)
      IF(FTEMP.EQ.0.0) GO TO 200
      IF(FTEMP.GT.0.0) GO TO 210
      IF(F1.GT.0.0) GO TO 220
      F1=FTEMP
      R1=RTEMP
      GO TO 230
  220 F2=FTEMP
      R2=RTEMP
      GO TO 230
  210 IF(F1.GT.0.0) GO TO 215
      F2=FTEMP
      R2=RTEMP
      GO TO 230
  215 F1=FTEMP
      R1=RTEMP
  230 IF( DABS((R1-R2)/R1)-1.0D-14 .GT.0.0) GO TO 190
  200 CALL MATRIX(RR,BETA,RES,R2,M,F,FF,2)
      GO TO 240
```

```
  160 CONTINUE
  150 CONTINUE
      WRITE(3,1)
    1 FORMAT('OUNABLE TO FIND A ROOT AFTER INDICATION THAT A ROOT EXISTE
     1D')
      CALL EXIT
  240 DO 250 I=1,IROOT
      J=I+1
      IF(R2.GT.ROOT(I)) GO TO 250
      J=I+1
      GO TO 260
  250 CONTINUE
  260 LAST=LAST+1
      IF(IROOT.EQ.1) GO TO 270
      JJ=IROOT+1
      DO 280 I=J,IROOT
      JJ=JJ-1
      ROOT(JJ)=ROOT(JJ-1)
      DO 280 K=1,4
      DER(JJ,K)=DER(JJ-1,K)
  280 FUNC(JJ,K)=FUNC(JJ-1,K)
  270 J=J-1
      ROOT(J)=R2
      DO 290 K=1,4
      KX=(K+1)/2
      KY=K/KX
      DER(J,K)=FF(KX,KY)
  290 FUNC(J,K)=F(KX,KY)
      GO TO 120
   30 CONTINUE
  130 RETURN
      END
```

```
0001          SUBROUTINE MATRIX(RR,BETA,RES,W,M,F,FF,ICONT)
        C
        C         SUBROUTINE TO CALCULATE THE HEAT TRANSFER MATRIX FOR A SLAB,
        C     AND THE DERIVATIVE OF THIS MATRIX.
        C
        C     IF ICONT=1 THE ROUTINE CALCULATES HEAT TRANSFER MATRIX ONLY.
        C     IF ICONT=2 THE ROUTINE CALCULATES HEAT TRANSFER MATRIX AND ITS
        C                DERIVATIVE.
        C
        C     NOMENCLATURE:
        C         RR=THICKNESS/THERMAL CONDUCTIVITY (XL/XK)
        C            OR THERMAL RESISTANCE OF LAYER WHEN THERE IS
        C            NEGLIGIBLE HEAT STORAGE.
        C         BETA*BETA=XL*XL*D*SH/K.
        C                   WHERE D=DENSITY.
        C                         SH=SPECIFIC HEAT.
        C         RES=RESISTANCE OF RADIATION PATH WHENEVER APPLICABLE.
        C
        C         W=VALUES ALONG THE AXIS FOR WHICH THE MATRIX OR THE MATRIX
        C           AND DERIVATIVES ARE FOUND.
        C         M=NUMBER OF LAYERS THE SLAB IS COMPOSED OF.
        C         F=CONTAINS THE VALUE OF THE HEAT TRANSFER MATRIX ON RETURN
        C         FF=CONTAINS THE VALUE OF THE DERIVATIVE ON RETURN.
        C
0002          DOUBLE PRECISION RR(20),BETA(20),F1(20,2,2),F2(20,2,2),F3(20,2,2),
             1F(2,2),FF(2,2),RES(20),P,R,ALPHA,SQ,W,TEMP,TEMP1
0003          DO 10 I=1,M
0004          P=DSQRT(W)*BETA(I)
0005          R=RR(I)
0006          ALPHA=BETA(I)
0007          SQ=DSQRT(W)
0008          IF(P.NE.0.0) GO TO 20
        C     ELEMENTS OF THE MATRIX FOR LAYER I WHERE THERE IS NEGLIGIBLE
        C     HEAT STORAGE.
0009          F1(I,1,1)=1.0
0010          F1(I,1,2)=R
0011          F1(I,2,1)=0.0
0012          F1(I,2,2)=1.0
0013          IF(ICONT.EQ.1) GO TO 10
        C     DERIVATIVES OF THE ELEMENTS OF THE MATRIX FOR LAYER I WHERE THERE
        C     IS NEGLIGIBLE HEAT STORAGE.
0014          F2(I,1,1)=0.0
0015          F2(I,1,2)=0.0
0016          F2(I,2,1)=0.0
0017          F2(I,2,2)=0.0
0018          GO TO 10
        C     ELEMENTS OF THE MATRIX FOR LAYER I FOR HEAT TRANSFER BY CONDUCTION
        C     ONLY.
0019       20 F1(I,1,1)=DCOS(P)
0020          F1(I,1,2)=R/P*DSIN(P)
0021          F1(I,2,1)=-P/R*DSIN(P)
0022          F1(I,2,2)=F1(I,1,1)
0023          IF(ICONT.EQ.1) GO TO 30
```

```
            C     DERIVATIVES OF THE ELEMENTS OF THE MATRIX FOR LAYER I FOR HEAT
            C     TRANSFER BY CONDUCTION ONLY.
0024              F2(I,1,1)=ALPHA*DSIN(ALPHA*SQ)/2.0/SQ
0025              F2(I,1,2)=-R*DCOS(ALPHA*SQ)/2.0/W+R*DSIN(ALPHA*SQ)/ALPHA/2.0/SQ/SQ
                 1/SQ
0026              F2(I,2,1)=ALPHA*ALPHA*DCOS(ALPHA*SQ)/2.0/R+DSIN(ALPHA*SQ)/2.0/SQ*
                 1ALPHA/R
0027              F2(I,2,2)=F2(I,1,1)
0028           30 IF(RES(I).EQ.0.0) GO TO 10
            C     ELEMENTS OF THE MATRIX FOR LAYER I WHERE THERE IS HEAT TRANSFER BY
            C     CONDUCTION AND THERMAL RADIATION USING CONDUCTION PART FROM ABOVE.
0029              TEMP=1.0/(F1(I,1,2)+RES(I))
0030              F1(I,2,1)=(F1(I,2,1)*RES(I)+2.0*F1(I,1,1)-2.0)*TEMP
0031              F1(I,1,1)=(F1(I,1,1)*RES(I)+F1(I,1,2))*TEMP
0032              F1(I,2,2)=F1(I,1,1)
0033              F1(I,1,2)=F1(I,1,2)*RES(I)*TEMP
0034              IF(ICONT.EQ.1) GO TO 10
            C     DERIVATIVES OF THE ELEMENTS OF THE MATRIX FOR LAYER I WHERE THERE
            C     IS HEAT TRANSFER BY CONDUCTION AND THERMAL RADIATION USING
            C     CONDUCTION PART FROM ABOVE
0035              TEMP1=F2(I,1,2)*TEMP
0036              F2(I,2,1)=(F2(I,2,1)*RES(I)+2.0*F2(I,1,1))*TEMP-F1(I,2,1)*TEMP1
0037              F2(I,1,1)=(F2(I,1,1)*RES(I)+F2(I,1,2))*TEMP-F1(I,1,1)*TEMP1
0038              F2(I,2,2)=F2(I,1,1)
0039              F2(I,1,2)=F2(I,1,2)*RES(I)*TEMP-F1(I,1,2)*TEMP1
0040           10 CONTINUE
            C     RETURN IF ONLY ONE LAYER INVOLVED.
0041              IF((M-1).NE.0) GO TO 50
0042              DO 40 K=1,2
0043              DO 40 L=1,2
0044              IF(ICONT.EQ.1) GO TO 40
0045              FF(K,L)=F2(1,K,L)
0046           40 F(K,L)=F1(1,K,L)
0047              RETURN
0048           50 DO 60 K=1,2
0049              DO 60 L=1,2
0050           60 FF(K,L)=0.0
0051              IF(ICONT.EQ.1) GO TO 150
            C     FOLLOWING IS THE ROUTINE TO COMBINE INDIVIDUAL DERIVATIVES OF THE
            C     HEAT TRANSFER MATRICES TO GET THE OVERALL DERIVATIVE.
0052              DO 140 I=1,M
0053              DO 120 J=1,M
0054              DO 80 K=1,2
0055              DO 80 L=1,2
0056              IF((I-J).EQ.0) GO TO 70
0057              F3(J,K,L)=F1(J,K,L)
0058              GO TO 80
0059           70 F3(J,K,L)=F2(J,K,L)
0060           80 CONTINUE
0061              IF((J-1).EQ.0) GO TO 120
0062              DO 90 K=1,2
0063              DO 90 L=1,2
0064           90 F(K,L)=0.0
```

```
      DO 100 K=1,2
      DO 100 L=1,2
      DO 100 N=1,2
  100 F(K,L)=F(K,L)+F3(J-1,K,N)*F3(J,N,L)
      DO 110 L=1,2
      DO 110 K=1,2
  110 F3(J,K,L)=F(K,L)
  120 CONTINUE
      DO 130 K=1,2
      DO 130 L=1,2
  130 FF(K,L)=FF(K,L)+F(K,L)
  140 CONTINUE
C     FOLLOWING IS THE ROUTINE TO COMBINE INDIVIDUAL HEAT TRANSFER
C     MATRIX TO GET THE OVERALL HEAT TRANSFER MATRIX.
  150 DO 190 I=2,M
      DO 160 K=1,2
      DO 160 L=1,2
  160 F(K,L)=0.0
      DO 170 K=1,2
      DO 170 L=1,2
      DO 170 N=1,2
  170 F(K,L)=F(K,L)+F1(I-1,K,N)*F1(I,N,L)
      DO 180 K=1,2
      DO 180 L=1,2
  180 F1(I,K,L)=F(K,L)
  190 CONTINUE
      RETURN
      END
```

```
0001                 SUBROUTINE ORIGIN(RR,BETA,RES,M,MP,MPP)
              C
              C      SUBROUTINE TO CALCULATE THE RESIDUES AT THE POLES OF THE
              C      Z-TRANSFER FUNCTIONS. (FIRST AND SECOND DERIVATIVES)
              C
              C      NOMENCLATURE:
              C          RR=THICKNESS/THERMAL CONDUCTIVITY (XL/XK).
              C             OR THERMAL RESISTANCE OF LAYER WHEN THERE IS
              C             NEGLIGIBLE HEAT STORAGE.
              C          BETA*BETA=XL*XL*D*SH/XK.
              C                   WHERE D=DENSITY.
              C                         SH=SPECIFIC HEAT.
              C          RES=RESISTANCE OF RADIATION PATH WHENEVER APPLICABLE.
              C
              C          M=NUMBER OF LAYER THE SLAB IS COMPOSED OF.
              C
              C          MP=CONTAINS THE VALUE OF THE FIRST DERIVATIVE AT THE
              C             POLES ON RETURN.
              C          MPP=CONTAINS THE VALUE OF THE SECOND DERIVATIVE AT THE POLES
              C              ON RETURN
              C
0002                 DOUBLE PRECISION RR(20),BETA(20),RES(20),MP(2,2),MPP(2,2),
                    1A(20,2,2),B(20,2,2),C(20,2,2),D(2,2),E(2,2),F(2,2),G(2,2),
                    2TEMP(2,2),TEMP1(2,2),P,R
0003                 DO 10 I=1,2
0004                 DO 10 J=1,2
0005                 MP(I,J)= 0.0
0006              10 MPP(I,J)=0.0
0007                 DO 40 I=1,M
0008                 P=BETA(I)*BETA(I)
0009                 R=RR(I)
              C      ELEMENTS OF THE MATRIX AT THE POLE FOR LAYER I, FOR CONDUCTION
              C      OR NEGLIGIBLE HEAT TRANSFER.
0010                 A(I,1,1)=1.0
0011                 A(I,1,2)=R
0012                 A(I,2,1)=0.0
0013                 A(I,2,2)=1.0
0014                 IF(RES(I).EQ.0.0) GO TO 20
              C      ELEMENTS OF THE MATRIX AT THE POLE FOR LAYER I, WHERE THERE IS
              C      HEAT TRANSFER BY CONDUCTION AND THERMAL RADIATION.
0015                 A(I,1,2)=R*RES(I)/(R+RES(I))
              C      FIRST DERIVATIVE OF THE ELEMENTS OF THE MATRIX AT THE POLE
              C      FOR LAYER I, FOR CONDUCTION OR NEGLIGIBLE HEAT STORAGE.
0016              20 B(I,1,1)=P/2.0
0017                 B(I,1,2)=R*P/6.0
0018                 B(I,2,1)=P/R
0019                 B(I,2,2)=P/2.0
0020                 IF(RES(I).EQ.0.0) GO TO 30
              C      FIRST DERIVATIVE OF THE ELEMENTS OF THE MATRIX AT THE POLE
              C      FOR LAYER I, WHERE THERE IS HEAT TRANSFER BY CONDUCTION AND
              C      THERMAL RADIATION.
0021                 B(I,1,1)=RES(I)*P/2.0/(R+RES(I))
0022                 B(I,1,2)=(1.0-R/(R+RES(I)))*RES(I)*R*P/6.0/(R+RES(I))
```

```
0023          B(I,2,1)=(RES(I)*P/R+P)/(R+RES(I))
0024          B(I,2,2)=B(I,1,1)
        C     SECOND DERIVATIVE OF THE ELEMENTS OF THE MATRIX AT THE POLE FOR
        C     LAYER I, FOR CONDUCTION OR NEGLIGIBLE HEAT STORAGE.
0025       30 C(I,1,1)=P*P/12.0
0026          C(I,1,2)=P*P*R/60.0
0027          C(I,2,1)=P*P/3.0/R
0028          C(I,2,2)=C(I,1,1)
0029       40 CONTINUE
        C     RETURN IF ONLY ONE LAYER INVOLVED.
0030          IF((M-1).NE.0) GO TO 60
0031          DO 50 I=1,2
0032          DO 50 J=1,2
0033          MP(I,J)=B(1,I,J)
0034       50 MPP(I,J)=C(1,I,J)
0035          RETURN
        C     FOLLOWING IS THE ROUTINE TO CALCULATE THE FIRST AND SECOND
        C     DERIVATIVE OF THE HEAT TRANSFER MATRIX AT THE POLES FOR A
        C     MULTILAYER SLAB.
0036       60 DO 280 I=1,M
0037          DO 260 K=1,M
0038          IF(I.NE.K) GO TO 80
0039          IF(I.NE.1) GO TO 70
0040          D(1,1)=B(1,1,1)
0041          D(1,2)=B(1,1,2)
0042          D(2,1)=B(1,2,1)
0043          D(2,2)=B(1,2,2)
0044          GO TO 80
0045       70 D(1,1)=A(1,1,1)
0046          D(1,2)=A(1,1,2)
0047          D(2,1)=A(1,2,1)
0048          D(2,2)=A(1,2,2)
0049       80 IF(I.NE.1) GO TO 90
0050          IF(K.NE.1) GO TO 100
0051          E(1,1)=C(1,1,1)
0052          E(1,2)=C(1,1,2)
0053          E(2,1)=C(1,2,1)
0054          E(2,2)=C(1,2,2)
0055          GO TO 120
0056       90 IF(K.NE.1) GO TO 110
0057      100 E(1,1)=B(1,1,1)
0058          E(1,2)=B(1,1,2)
0059          E(2,1)=B(1,2,1)
0060          E(2,2)=B(1,2,2)
0061          GO TO 120
0062      110 E(1,1)=A(1,1,1)
0063          E(1,2)=A(1,1,2)
0064          E(2,1)=A(1,2,1)
0065          E(2,2)=A(1,2,2)
0066      120 DO 240 J=2,M
0067          IF(I.NE.K) GO TO 170
0068          IF(I.EQ.J) GO TO 130
0069          F(1,1)=A(J,1,1)
```

```
      F(1,2)=A(J,1,2)
      F(2,1)=A(J,2,1)
      F(2,2)=A(J,2,2)
      GO TO 140
  130 F(1,1)=B(J,1,1)
      F(1,2)=B(J,1,2)
      F(2,1)=B(J,2,1)
      F(2,2)=B(J,2,2)
  140 DO 150 L=1,2
      DO 150 LL=1,2
      TEMP(L,LL)=0.0
      DO 150 LLL=1,2
  150 TEMP(L,LL)=TEMP(L,LL)+D(L,LLL)*F(LLL,LL)
      DO 160 L=1,2
      DO 160 LL=1,2
  160 D(L,LL)=TEMP(L,LL)
  170 IF(I.EQ.J) GO TO 180
      IF(K.EQ.J) GO TO 190
      G(1,1)=A(J,1,1)
      G(1,2)=A(J,1,2)
      G(2,1)=A(J,2,1)
      G(2,2)=A(J,2,2)
      GO TO 210
  180 IF(K.EQ.J) GO TO 200
  190 G(1,1)=B(J,1,1)
      G(1,2)=B(J,1,2)
      G(2,1)=B(J,2,1)
      G(2,2)=B(J,2,2)
      GO TO 210
  200 G(1,1)=C(J,1,1)
      G(1,2)=C(J,1,2)
      G(2,1)=C(J,2,1)
      G(2,2)=C(J,2,2)
  210 DO 220 L=1,2
      DO 220 LL=1,2
      TEMP1(L,LL)=0.0
      DO 220 LLL=1,2
  220 TEMP1(L,LL)=TEMP1(L,LL)+E(L,LLL)*G(LLL,LL)
      DO 230 L=1,2
      DO 230 LL=1,2
  230 E(L,LL)=TEMP1(L,LL)
  240 CONTINUE
      DO 250 L=1,2
      DO 250 LL=1,2
  250 MPP(L,LL)=MPP(L,LL)+TEMP1(L,LL)
  260 CONTINUE
      DO 270 L=1,2
      DO 270 LL=1,2
  270 MP(L,LL)=MP(L,LL)+TEMP(L,LL)
  280 CONTINUE
      RETURN
      END
```

```
0001          SUBROUTINE FREQRE (RR,BETA,RES,XL,XK,D,SH,M,W,A,LW)
        C
        C         THIS SUBROUTINE CALCULATES THE FUNCTIONS OF THE HEAT
        C     TRANSFER MATRIX WHEN S=IW  WHERE I=SQRT(-1.0)
        C
        C     NOMENCLATURE:
        C         RR=THICKNESS/THERMAL CONDUCTIVITY (XL/XK)
        C            OR THERMAL RESISTANCE OF LAYER WHEN THERE IS
        C            NEGLIGIBLE HEAT STORAGE.
        C         BETA*BETA=XL*XL*D*SH/XK
        C                   WHERE D=DENSITY
        C                         SH=SPECIFIC HEAT
        C         RES=RESISTANCE OF RADIATION PATH WHENEVER APPLICABLE.
        C
        C         M=NUMBER OF LAYER THE SLAB IS COMPOSED
        C
        C         W=ARRAY CONTAINING THE FREQUENCIES AT WHICH THE FUNCTIONS
        C           ARE EVALUATED
        C
        C         A=CONTAINS THE VALUES OF THE FUNCTIONS AT S=IW FOR THE
        C           VARIOUS FREQUENCIES ON RETURN FROM THE SUBROUTINE.
        C
0002          DOUBLE PRECISION RR(20),BETA(20),RES(20),XL(20),XK(20),D(20),
             1SH(20),AA,BB,CC,DD,EE,FF,P,R,ALPHA,PHI,PI,ARG1,ARG2,TEMP,W(6)
0003          COMPLEX*16 A(6,2,2),MM(2,2),MMM(2,2),MMMM(2,2)
0004          PI=3.14159265
0005          DO 60 J=1,LW
0006          DO 10 I=1,M
0007          R=RR(I)
0008          IF(W(J).NE.0.0.AND.XL(I).NE.0.0) GO TO 5
0009          MM(1,1)=(1.0D0,0.0D0)
0010          ARG2=0.0
0011          MM(1,2)=DCMPLX(R,ARG2)
0012          MM(2,1)=(0.0D0,0.0D0)
0013          GO TO 6
0014        5 P=2.0*PI/W(J)
0015          ALPHA=XK(I)/D(I)/SH(I)
0016          PHI=DSQRT(PI*XL(I)*XL(I)/ALPHA/P)
0017          AA=DSIN(PHI)
0018          BB=DCOS(PHI)
0019          CC=DEXP(PHI)
0020          DD=DEXP(-PHI)
0021          EE=(CC-DD)/2.0
0022          FF=(CC+DD)/2.0
0023          ARG1=FF*BB
0024          ARG2=EE*AA
0025          MM(1,1)=DCMPLX(ARG1,ARG2)
0026          ARG1=RR(I)*(FF*AA+EE*BB)/2.0/PHI
0027          ARG2=RR(I)*(FF*AA-EE*BB)/2.0/PHI
0028          MM(1,2)=DCMPLX(ARG1,ARG2)
0029          TEMP=2.0*ARG1*PHI*PHI/RR(I)/RR(I)
0030          ARG1=-ARG2*2.0*PHI*PHI/RR(I)/RR(I)
0031          ARG2=TEMP
```

```
      MM(2,1)=DCMPLX(ARG1,ARG2)
    6 MM(2,2)=MM(1,1)
      IF(RES(I).EQ.0.0) GO TO 7
      MM(2,1)=(MM(2,1)*RES(I)+2.0*MM(1,1)-2.0)/(MM(1,2)+RES(I))
      MM(1,1)=(MM(1,1)*RES(I)+MM(1,2))/(MM(1,2)+RES(I))
      MM(1,2)=(MM(1,2)*RES(I))/(MM(1,2)+RES(I))
      MM(2,2)=MM(1,1)
    7 IF(I.EQ.1) GO TO 20
      DO 30 K=1,2
      DO 30 L=1,2
      MMMM(K,L)=(0.0,0.0)
      DO 30 N=1,2
   30 MMMM(K,L)=MMMM(K,L)+MM(N,L)*MMM(K,N)
      DO 40 K=1,2
      DO 40 L=1,2
   40 MMM(K,L)=MMMM(K,L)
      GO TO 10
   20 DO 50 K=1,2
      DO 50 L=1,2
   50 MMM(K,L)=MM(K,L)
   10 CONTINUE
      DO 70 K=1,2
      DO 70 L=1,2
   70 A(J,K,L)=MMM(K,L)
   60 CONTINUE
      RETURN
      END
```

```
            SUBROUTINE POLYM(A,B,N,M)
      C
      C     THIS SUBROUTINE FINDS THE PRODUCT OF TWO POLYNOMIALS OF ORDER
      C     N AND M RESPECTIVELY.(N=ORDER OF NEW POLYNOMIAL A ON RETURN).
      C
            DOUBLE PRECISION A(100),B(100),C(100)
            K=N+M
            K=MINO(K,99)
            KK=K+1
            DO 10 I=1,100
         10 C(I)=0.0
            DO 25 I=1,KK
            KKK=I
            L=N+1
            L=MINO(I,L)
            J=I-M
            J=MAXO(1,J)
            DO 20 NN=J,L
            INN=I-NN+1
         20 C(I)=C(I)+A(NN)*B(INN)
            IF(I.LT.5)GO TO 25
            IF(DABS(C(I)).LT.0.0000001) GO TO 35
         25 CONTINUE
         35 KK=KKK
            DO 30 I=1,100
         30 A(I)=C(I)
            N=KK-1
            RETURN
            END
```

Samples of Input and Output

Note: Input and output are in consistent units. These examples are in the International System of Units (SI).

NUMERICAL DATA FOR EXAMPLE WALL.
SLAB COMPONENTS.

LAYER	THICKNESS	CONDUCTIVITY	DENSITY	SP HEAT	RESISTANCE	DESCRIPTION OF LAYER
1	0.0	0.0	0.0	0.0	0.1500	AIR INSIDE SURFACE
2	0.1000	0.7300	1600.0000	920.0000	0.0	BRICK
3	0.1000	1.3300	2000.0000	920.0000	0.0	CONCRETE MEDIUM WEIGHT
4	0.0	0.0	0.0	0.0	0.0600	AIR OUTSIDE SURFACE

THERMAL CONDUCTANCE, U= 2.369

SAMPLING TIME INTERVAL, DT= 3600.000

COEFFICIENTS FOR RAMP INPUT

J	D/B	1/B	A/B	D(Z)
0	5.133114	0.000852	11.093835	1.000000
1	-7.781125	0.049375	-17.670871	-1.337286
2	3.105002	0.118010	7.503862	0.451402
3	-0.255996	0.034871	-0.733343	-0.028029
4	0.003320	0.001198	0.010846	0.000168
5	-0.000005	0.000003	-0.000020	-0.000000
6	0.000000	0.000000	0.000000	

NUMERICAL DATA FOR EXAMPLE WALL.
SLAB COMPONENTS.

LAYER	THICKNESS	CONDUCTIVITY	DENSITY	SP HEAT	RESISTANCE	DESCRIPTION OF LAYER
1	0.0	0.0	0.0	0.0	0.1500	AIR INSIDE SURFACE
2	0.1000	0.7300	1600.0000	920.0000	0.0	BRICK
3	0.1000	1.3300	2000.0000	920.0000	0.0	CONCRETE MEDIUM WEIGHT
4	0.0	0.0	0.0	0.0	0.0600	AIR OUTSIDE SURFACE

THERMAL CONDUCTANCE, U= 2.369

SAMPLING TIME INTERVAL, DT= 3600.000

COEFFICIENTS BY FREQUENCY RESPONSE

PERIODS 28800.0 17280.0 12240.0 86400.0

J	D/B	1/B	A/B	D(Z)
0	5.342817	-0.001245	11.780631	1.000000
1	-8.615756	0.038669	-20.437882	-1.337286
2	4.556544	0.146869	12.380368	0.451402
3	-1.843089	0.015254	-6.121494	-0.028029
4	1.353690	0.008909	4.611540	0.000168
5	-0.965779	-0.006880	-3.288532	-0.000000
6	0.553785	0.004025	1.885314	
7	-0.226359	-0.001641	-0.770602	
8	0.048455	0.000348	0.164965	

NUMERICAL DATA FOR EXAMPLE WALL.
SLAB COMPONENTS.

LAYER	THICKNESS	CONDUCTIVITY	DENSITY	SP HEAT	RESISTANCE	DESCRIPTION OF LAYER
1	0.0000	0.0000	0.0000	0.0000	0.1500	AIR INSIDE SURFACE
2	0.1000	0.7300	1600.0000	920.0000	0.0000	BRICK
3	0.1000	1.3300	2000.0000	920.0000	0.0000	CONCRETE MEDIUM WEIGHT
4	0.0000	0.0000	0.0000	0.0000	0.0600	AIR OUTSIDE SURFACE

THERMAL CONDUCTANCE, U= 2.369

SAMPLING TIME INTERVAL, DT= 3600.000

COEFFICIENTS FOR STEP INPUT

J	D/C	1/C	A/C	D(Z)
0	0.060000	0.000000	0.150000	1.000000
1	-0.060151	0.000081	-0.193362	-1.724442
2	-0.008252	0.001780	0.034077	0.799418
3	0.013181	0.001778	0.014576	-0.075676
4	-0.000982	0.000162	-0.001499	0.000700
5	0.000007	0.000001	0.000010	-0.000000
6	-0.000000	0.000000	-0.000000	0.000000

NUMERICAL DATA FOR EXAMPLE WALL.
SLAB COMPONENTS.

LAYER	THICKNESS	CONDUCTIVITY	DENSITY	SP HEAT	RESISTANCE	DESCRIPTION OF LAYER
1	0.0000	0.0000	0.0000	0.0000	0.1500	AIR INSIDE SURFACE
2	0.1000	0.7300	1600.0000	920.0000	0.0000	BRICK
3	0.1000	1.3300	2000.0000	920.0000	0.0000	CONCRETE MEDIUM WEIGHT
4	0.0000	0.0000	0.0000	0.0000	0.0600	AIR OUTSIDE SURFACE

THERMAL CONDUCTANCE, U= 2.369

SAMPLING TIME INTERVAL, DT= 3600.000

COEFFICIENTS BY FREQUENCY RESPONSE

PERIODS 43200.0 21600.0 14400.0 10800.0 32400.0

J	D/C	1/C	A/C	D(Z)
0	0.000000	0.000000	0.000000	1.000000
1	0.084316	-0.000017	0.186706	-1.724442
2	-0.109552	0.000544	-0.268202	0.799418
3	0.009106	0.002665	0.059890	-0.075676
4	0.042625	0.000553	0.059664	0.000700
5	-0.043529	0.000127	-0.065753	-0.000000
6	0.038455	-0.000136	0.058134	0.000000
7	-0.030540	0.000117	-0.046171	
8	0.020960	-0.000083	0.031689	
9	-0.011693	0.000047	-0.017679	
10	0.004671	-0.000019	0.007061	
11	-0.001017	0.000004	-0.001538	

NUMERICAL DATA FOR EXAMPLE WALL.
SLAB COMPONENTS.

LAYER	THICKNESS	CONDUCTIVITY	DENSITY	SP HEAT	RESISTANCE	DESCRIPTION OF LAYER
1	0.0000	0.0000	0.0000	0.0000	0.1500	AIR INSIDE SURFACE
2	0.1000	0.7300	1600.0000	920.0000	0.0000	BRICK
3	0.1000	1.3300	2000.0000	920.0000	0.0000	CONCRETE MEDIUM WEIGHT
4	0.0000	0.0000	0.0000	0.0000	0.0600	AIR OUTSIDE SURFACE

THERMAL CONDUCTANCE, U= 2.369

SAMPLING TIME INTERVAL, DT= 3600.000

COEFFICIENTS FOR RAMP INPUT

J	D/C	1/C	A/C	D(Z)
0	0.000000	0.000000	0.000000	1.000000
1	0.088859	0.000011	0.193541	-1.724442
2	-0.129085	0.000749	-0.297781	0.799418
3	0.045849	0.002187	0.115608	-0.075676
4	-0.001759	0.000818	-0.007503	0.000700
5	-0.000061	0.000038	-0.000064	-0.000000
6	0.000000	0.000000	0.000000	0.000000
7	-0.000000	0.000000	-0.000000	

This paper was presented at the
First Symposium on the Use of Computers for
Environmental Engineering Related to Buildings
held at the U.S. National Bureau of Standards
30 Nov. to 2 Dec. 1970.