COMPUTER PROGRAM FOR THERMAL ANALYSIS OF TANK-MOUNTED MULTILAYER INSULATION

by Glenn R. Cowgill and William R. Johnson
Lewis Research Center
Cleveland, Ohio 44135

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • MAY 1971
An analysis was done on the space-hold thermal performance of various multilayer insulation configurations on an oblate spheroidal tank. The thermal code CINDA-3G (Chrysler Improved Numerical Differencing Analyzer for Third Generation Computers) was used for the thermal analysis, and several subroutines were written to tailor it for the needs of the problem. A portion of the tailoring is called the pre-preprocessor, which generates the data required as input to CINDA-3G. This pre-preprocessor and allied subroutines comprise a package to obtain a parametric thermal study with a minimum amount of input data. Parameters may be changed with a minimum of effort.
COMPUTER PROGRAM FOR THERMAL ANALYSIS OF
TANK-MOUNTED MULTILAYER INSULATION

by Glenn R. Cowgill and William R. Johnson

Lewis Research Center

SUMMARY

A computer program was developed to analyze the space-hold thermal performance of various multilayer insulation (MLI) configurations on the upper half of an oblate spheroidal liquid-hydrogen tank. The thermal model considered was an unshrouded tank within a hypothetical sun-oriented vehicle such that the payload was the only heat source in the system.

The thermal code CINDA-3G (Chrysler Improved Numerical Differencing Analyzer for Third Generation Computers) was chosen for the basic thermal analysis, and several subroutines were written to tailor it for the needs of the particular problem. A portion of the tailoring included a program called the pre-preprocessor, which generated the data required as input for the CINDA-3G program.

This pre-preprocessor and allied subroutines comprise a package to obtain a parametric thermal analysis of any multilayer covering on a spheroid. As a result, a minimum amount of input is required to specify the model geometry so that perturbations in the geometry configuration may be introduced with a minimum of effort.

INTRODUCTION

The long-term storage of liquid hydrogen in space can be effectively accomplished by using high-performance multilayer insulation (MLI) systems. A typical system would use a large number of highly reflective shields (aluminized polyester film) separated by low-thermal-conductivity spacers (e.g., silk netting).

A cross-sectional view of a sun-oriented vehicle with an unshrouded liquid-hydrogen tank covered with MLI is shown in figure 1. The payload intercepts all solar energy and thus represents the only high-temperature heat source in the vehicle. The direction of the anticipated heat flows could be as indicated by the arrows in figure 1. The only heat
being transferred into any of the tanks is through the MLI on the top half of the liquid-hydrogen tank. Therefore, this discussion will consider the thermal performance of the MLI on the top half of the liquid-hydrogen tank only.

This analysis will be general for a steady-state study of any oblate spheroidal tank and payload geometry. However, to simplify the discussion, the programs herein will be described with respect to a particular problem; that is, a steady-state analysis of a variable-thickness shield of MLI on the top half of an oblate spheroidal tank viewing a constant-temperature payload. A complete discussion and the results of this problem for the geometries considered may be found in reference 1.

A cross-sectional view of the variable-thickness MLI system on the top half of the liquid-hydrogen tank is shown in figure 2. The net heat flow by thermal radiation into the top surface of the MLI is transferred within the MLI by

1. Conduction in the lateral direction parallel to the shields, \( Q_{\text{lat}} \)
2. Combination of radiation and conduction in the normal direction perpendicular to the shields, \( Q_{\text{norm}} \)

For a steady-state condition to exist, the net thermal radiation received by the top surface of the MLI must be exactly equal to the total heat transferred through the MLI and into the liquid-hydrogen tank. The problem then becomes one of solving for the net heat flows after temperatures resulting from local heat balances throughout the MLI have been established.

There are a number of computer programs available which, when fed the proper input data to describe the model of a physical problem, will perform a thermal analysis and output the temperature distribution. When these programs are used, both building the mathematical model which describes the physical geometry and the preparation of the necessary input data are most tedious. To alleviate this difficulty, the computer subroutines described in this report were written to provide a ready means to generate this input data.

These subroutines are specifically designed to solve the problem described in reference 1, but can be generally applied to any tank in the shape of an oblate spheroid with layers of material covering the tank, or for a shell of varying thickness having the same general shape. These computer programs, then, generate the data required by CINDA-3G (ref. 2), which is the particular thermal analysis program chosen by the authors because of its versatility. The capability to readily effect small changes in geometry added to the versatility and afforded a substantial saving in time and effort.

The routines to be described herein are (1) a pre-preprocessor which takes the input data and converts it to the CINDA-3G data as prescribed by that program, (2) a view factor program which computes the geometric view factors for radiation in an enclosure, (3) an output routine which graphically displays the temperatures, thus making it easier to get an overall view of the temperature distribution, and (4) various other subroutines that are used for special operations by routines (1), (2), and (3).
ANALYSIS

Thermal Model

The thermal model for this problem is shown in figures 3 and 4. It consists of a 45° wedge of the MLI on top of an oblate spheroidal tank. This wedge is divided into a number of internal nodes.

For convenience, the geometry of all the nodes in a given column is assumed to be identical in the normal direction, having numerical values for the node width, node length, the lateral cross-sectional area (ALAT), and the normal area (ANORM) based on the dimensions on the top surface of the MLI. This should introduce no significant error when the maximum MLI thickness is small relative to the minor tank radius.

The node length for each segment (column) is described by specifying the angles \( \theta_1, \theta_2, \ldots, \theta_n \) (all \( \theta \)'s must be integers). The depth of all nodes is constant and is specified by two input variables: the number of shields in each node, NSPL, and the thickness of each shield, DELS. The product of NSPL and DELS then is the depth of each internal node. The number of nodes in each column is an input variable, NL, which is a function of the MLI thickness being simulated in that particular column. The program is set up to accept a maximum of 30 segments and up to 50 nodes in each column. For the particular problem under discussion, 10 segments were used with varying numbers of layers (see fig. 3). The computed performance of this wedge is then multiplied by 8 to obtain the results for the entire top half of the tank.

A constant-value boundary node represents the temperature of the tank and the tank surface is held at this constant temperature. The payload is simulated by a flat disk divided into concentric rings (maximum, 10 rings). Space is simulated by a constant-temperature node with temperature at absolute zero and an emissivity equal to unity (blackbody).

Thermal Radiation

An enclosure composed of the payload, the surface of the MLI, and an imaginary surface representing space is used to compute the net thermal radiation received by the top surface of the MLI. The enclosure formed with the various surfaces labeled is shown in figure 5. The enclosure used in this discussion is composed of the 10 tank surface areas, five payload surface areas, and an imaginary surface simulating space, for a total of 16 surfaces.

The payload surface represented by a flat disk is divided into concentric rings of approximately equal area (labeled \( A_{11} \) to \( A_{15} \) in fig. 5). Each of these rings is consid-
ered to be a separate surface of the enclosure, and as such, each ring may have a differ-
ent surface emissivity and a different constant temperature which is considered to be
an infinite heat source. This is then assumed to be a multiple gray surface enclosure
and simulates a radiosity network. A CINDA-3G subroutine, IRRAD1, is then used to
compute the net radiant heat flow rates to each surface of the enclosure. Subroutine
IRRAD1 uses a method described in reference 3 to obtain the net radiant heat flow to
each surface.

Computer Program - General

CINDA-3G is a two-pass or double-phase operation consisting of a preprocessor
phase and an execution phase. The user input data (description of the model) to
CINDA-3G are converted into FORTRAN IV language subroutines by the preprocessor.
These subroutines are then passed onto the system FORTRAN compiler for compilation
and execution for solution of the problem.

Since the user input data as described in reference 2 are for a fixed geometric
model, much extra labor would be necessary to change the input for a slightly modified
geometry. Thus, a set of subroutines were written to accept a general description of
the model and to generate the user input data so that the geometry can easily be varied
by small changes in input data. This set of author-written subroutines was added to the
CINDA-3G system as an additional phase of operation, making a three-pass or three-
phase operation. This additional phase, called the pre-preprocessor, is executed first
to generate all the necessary input data as required by the CINDA-3G preprocessor using
a minimum amount of punched card input data. The CINDA-3G data are described in ref-
ERENCE 2 (pp. 4.1 to 4.21). Control is then passed to the CINDA-3G preprocessor
(phase 2) to process this generated data and to generate the subroutines for the third
phase of the operation to perform the execution of the problem.

Because of this method of combining the three phases of operation and the use of
certain portions of the computer operating system, this overall program as described is
usable only on IBM 7094 II/7044 Direct Couple Systems running under the operating sys-
tem IBSYS version 13, with a FORTRAN IV compiler. However, the CINDA-3G program
is available for other computer configurations, for example UNIVAC 1108; and the rou-
tines written by the authors as described in this report could easily be modified to oper-
ate on these other computers since they are entirely written in FORTRAN IV.

Finite Difference Equations

The input to CINDA-3G is the description of the model. This description is given by
designating each incremental volume, called a node, by a number and then describing
the heat paths between nodes by specifying the node numbers of the two adjoining nodes
and a value which represents the conductance between these two nodes. Conductance is
the area of heat flow times the thermal conductivity divided by the conductor length.
Heat flow to boundaries is specified by supplying heat-transfer coefficients or radiation
constants as the conductance between the two nodes.

When a network (model) to the problem has been described, the solution is obtained
by solving a set of finite difference equations, which are an approximation to the partial
differential equations of the diffusion type

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T + S$$

where \(T\) is temperature, \(t\) is time, \(\alpha\) is diffusivity

$$\alpha = \frac{k}{C_p}$$

\(\nabla^2\) is the Laplacian operator in \(x, y, z\) coordinates, and \(S\) is a source term defined
as

$$S = \frac{q \alpha}{k}$$

where \(q\) is heat rate per unit volume and \(k\) is thermal conductivity.

Since we are interested in only the steady-state solution for this particular problem,
the time derivative is equal to zero, and equation (1) reduces to Poisson's equation:

$$\alpha \nabla^2 T + S = 0$$

Substituting equations (2) and (3) into equation (4) results in

$$\frac{k \nabla^2 T}{\rho C_p} + \frac{qk}{\rho C_p} = 0$$

Then since \(\rho C_p\) may be assumed to be a constant for steady-state computation, multiply
both sides of equation (5) by \(\rho C_p\) to get
\[ k \nabla^2 T + \dot{q} = k \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) + \dot{q} = 0 \]  

(6)

Then

\[ k \nabla^2 T + \dot{q} = \frac{k}{\Delta x} \left( \frac{\partial T}{\partial x^-} - \frac{\partial T}{\partial x^+} \right) + \frac{k}{\Delta y} \left( \frac{\partial T}{\partial y^-} - \frac{\partial T}{\partial y^+} \right) + \frac{k}{\Delta z} \left( \frac{\partial T}{\partial z^-} - \frac{\partial T}{\partial z^+} \right) + \dot{q} \]  

(7)

where the plus and minus signs in the first partial term indicate that they are taken on the negative or positive side, respectively, of the point under consideration. If three consecutive points (1, 0, and 2) ascending in the \( x \) direction are considered, the finite difference of the \( x \) portion of equation (7) is

\[ \frac{k}{\Delta x} \left( \frac{T_1 - T_0}{\Delta x^-} + \frac{T_2 - T_0}{\Delta x^+} \right) = \frac{k}{\Delta x} \left( \frac{T_1 - T_0}{\Delta x^-} + \frac{T_2 - T_0}{\Delta x^+} \right) \]  

(8)

applying this to equation (7) for \( x, y, \) and \( z \) for the points

1, 0, 2  in \( x \) direction
3, 0, 4  in \( y \) direction
5, 0, 6  in \( z \) direction

yields:

\[ k \nabla^2 T_0 + \dot{q} = \frac{k}{\Delta x} \left( \frac{T_1 - T_0}{\Delta x^-} + \frac{T_2 - T_0}{\Delta x^+} \right) + \frac{k}{\Delta y} \left( \frac{T_3 - T_0}{\Delta y^-} + \frac{T_4 - T_0}{\Delta y^+} \right) + \frac{k}{\Delta z} \left( \frac{T_5 - T_0}{\Delta z^-} + \frac{T_6 - T_0}{\Delta z^+} \right) + \dot{q} \]  

(9)
taking common denominator of volume \( V = (\Delta x)(\Delta y)(\Delta z) \) and \( A_x = (\Delta y)(\Delta z) \), \( A_y = (\Delta x)(\Delta z) \), \( A_z = (\Delta x)(\Delta y) \) and combining equation (9) with equation (6) to obtain

\[
\frac{kA_x}{\Delta x^-}(T_1 - T_0) + \frac{kA_x}{\Delta x^+}(T_2 - T_0) + \frac{kA_y}{\Delta y^-}(T_3 - T_0) + \frac{kA_y}{\Delta y^+}(T_4 - T_0) + \frac{kA_z}{\Delta z^-}(T_5 - T_0) + \frac{kA_z}{\Delta z^+}(T_6 - T_0) + q = 0
\]  

(10)

now let \( G_1 = kA_x/\Delta x^- \), \( G_2 = kA_x/\Delta x^+ \), \( G_3 = kA_y/\Delta y^- \), . . . , etc. Then

\[
G_1(T_1 - T_0) + G_2(T_2 - T_0) + G_3(T_3 - T_0) + G_4(T_4 - T_0) + G_5(T_5 - T_0) + G_6(T_6 - T_0) + \dot{q} = 0
\]  

(11)

\[
G_1T_1 + G_2T_2 + G_3T_3 + G_4T_4 + G_5T_5 + G_6T_6 + \dot{q} = T_0\left(G_1 + G_2 + G_3 + G_4 + G_5 + G_6\right)
\]  

(12)

or

\[
T_0 = \frac{G_1T_1 + G_2T_2 + G_3T_3 + G_4T_4 + G_5T_5 + G_6T_6 + \dot{q}}{G_1 + G_2 + G_3 + G_4 + G_5 + G_6}
\]  

(13)

or, in more general terms,

\[
T_0 = \frac{\sum G_i T_i + \dot{q}_0}{\sum G_i}
\]  

(14)

where \( i \) ranges over all neighboring nodes to \( T_0 \).
Since all the temperatures $T_i$ in equation (14) are not known, assume some initial temperatures and solve this equation for $T_0$; when this is done over all the nodes in the network, a new temperature distribution will be obtained. Iterate in this fashion with equation (14) over all the nodes in the network until the temperatures obtained do not change between two consecutive iterations by more than some prescribed value, called the convergence criterion. This is the method used by the execution routine CINDSS (see ref. 2).

**Geometric Radiation View Factors Between Surfaces on a Flat-Surfaced Payload and a Spheroidal Tank**

The geometric radiation view factor between any two surfaces $A_i$ and $A_j$ shown in the following sketch is defined by equation (15): 

$$A_i F_{i,j} = \int_{A_i} \int_{A_j} \frac{\cos \varphi_i \cos \varphi_j \, dA_i \, dA_j}{\pi R^2}$$  \hspace{1cm} (15)
where $F_{i,j}$ represents the portion of the total energy, either emitted or reflected by surface $i$, that is intercepted by surface $j$. As shown in the sketch, $\phi_i$ and $\phi_j$ are the angles between the respective normals to the surface elements $dA_i$ and $dA_j$ and the connecting line $R$.

The double integral in equation (15) is approximated by the following double summation in order to obtain a numerical solution on the digital computer:

$$A_i F_{i,j} = \sum_{A_i} \sum_{A_j} \frac{\cos \phi_i \cos \phi_j \Delta A_i \Delta A_j}{\pi R^2}$$  (16)

The radiosity subroutine used to calculate the net heat into each surface of the enclosure requires the following numerical input:

- $A_{NE}$ area of each surface in the enclosure
- $T_{NE}$ temperature of each surface in the enclosure
- $\epsilon_{NE}$ emissivity of each surface in the enclosure
- $A\times F$ area times geometric view factor for each interchange in the enclosure (see ref. 2, p. 6.6.2)

The procedure described in this section first determines the individual values of $A\times F$ for each of the tank surfaces viewing each of the payload surfaces. Once these primary values are determined, all other values of $A\times F$ are readily determined. The equations used to define the other values of $A\times F$ are discussed at the very end of this section.

The view factors, and subsequently the net $q$ into each of the surfaces in the radiosity enclosure, are computed for the entire top half of the tank and the entire payload. As described earlier, in the discussion of the model, the top node of each column of MLI is a 45° segment. These segments are extended completely around the tank to form a ring and the net $q$ is computed for each of these rings. These $q$'s are then divided by 8 to obtain the net $q$ into each node on top of the MLI. The method of computing the view factors described herein further breaks up each of these rings on the MLI and also breaks up the payload surfaces, described on page 3, into smaller rings. Then these small areas with their respective view factors are added together to obtain the $A\times F$ for each payload surface and each MLI surface, where each MLI surface is a ring as described previously.
The actual view factors are calculated as follows: The payload is arbitrarily divided into concentric rings each having a width of 0.1 foot (3.05 cm). Each ring with area $A_p$ is then further divided into incremental areas $\Delta A_p$. As shown in figure 6, the subscript $p$ will pertain to the payload surface, and the subscript $t$ will pertain to the tank surface. The number of incremental areas in each ring $N_J$ is varied so that the numerical value of $\Delta A_p$ remains approximately constant over the entire payload. Thus, for the payload ring shown in figure 6, $N_J$ is the nearest whole number determined by the following expression:

$$N_J = 20\pi R_p$$

The number of these rings of 0.1-foot (3.05-cm) width which are to be included in each of the individual payload surfaces is then readily determined.

The MLI surface on the tank is arbitrarily divided into 90 increments of $1^\circ$ each, measured from the vertical z-axis. As shown in figure 6, the incremental area extends in a ring completely around the tank. The location of this ring is specified by $\psi_m$, the angular distance from the vertical z-axis. Each ring with area $A_t$ is further divided into incremental areas $\Delta A_t$ of constant size over the entire tank surface. The number of individual rings to be included in each of the separate tank surfaces used in the enclosure is then readily determined once the angular size of each of the surfaces is specified.

The variable MLI thickness over any particular tank surface area is accounted for by reading in the number of nodes used at that particular location. The numerical values of $a$ and $b$ (the major and minor semi-axes of the spheroid formed by the surface of the MLI) are determined individually for each of the tank areas used.

$$a = X_r + \Delta S$$

$$b = Y_r + \Delta S$$

where $\Delta S$ is the thickness of the MLI and $X_r$ and $Y_r$ are the uninsulated tank dimensions. Thus for any particular surface area of the tank, the existing MLI thickness over that particular area is considered to exist over the entire tank surface.

Both the tank and the payload are symmetrical about their respective z-axes. Because of this symmetry and the fact that the z-axes of both coordinate systems coincide, the geometric view factor of any one element $\Delta A_t$ along a given tank ring to any given
complete payload ring is independent of position. Thus, equation (16) may be written in
the following manner:

\[ A_t F_t, p = A_t \sum_{A_p} \frac{\cos \varphi_t \cos \varphi_p \Delta A_p}{\pi R^2} \]  (19)

where \( A_t \) and \( A_p \) are the tank and payload ring areas, respectively. Thus, the view
factor \( F_t, p \) of any ring on the tank to any ring on the payload is the summation of the
view factors to all the incremental areas making up the payload ring.

To simplify the geometry, the typical element \( \Delta A_t \) on any given tank ring, selected
to view all elements \( \Delta A_p \) on any given payload ring, is located in the \( yz \) plane. In this
plane, the equation of the MLI surface is

\[ \frac{y^2}{a^2} + \frac{z^2}{b^2} = 1 \]  (20)

The following geometric relations are then valid:

\[
\begin{align*}
R_t &= ab \sqrt[1]{\frac{1}{b^2 \sin^2 \psi_m + a^2 \cos^2 \psi_m}} \\
y_t &= R_t \sin \psi_m \\
z_t &= R_t \cos \psi_m \\
\Delta h &= b - z_t
\end{align*}
\]  (21)

For each value of \( \psi_m \) used, the entire payload will be divided into incremental
areas \( \Delta A_p \). Thus, \( R_p \) will be a function of the payload ring being considered, and \( \gamma_n \)
will be a function of the location of \( \Delta A_p \). The following relations can be seen from fig-
ure 6:
\[ \begin{aligned}
& x_p = R_p \cos \gamma_n \\
& y_p = R_p \sin \gamma_n \\
& \Delta A_p = \frac{\pi}{N_J} \left[ \left( R_p + \frac{\Delta L}{2} \right)^2 - \left( R_p - \frac{\Delta L}{2} \right)^2 \right]
\end{aligned} \tag{22} 
\]

where \( N_J \) is determined by equation (17). Differentiating equation (20) gives the slope at any point on the curve:

\[ \frac{dz}{dy} = -\frac{b^2 y}{a^2 z} \tag{23} \]

The slope of a line normal to that point \( m_{N_1} \) can be expressed as

\[ \begin{aligned}
& m_{N_t} = -\frac{1}{\frac{dz}{dy}} = \frac{a^2 z_t}{b^2 y} = \frac{a^2 z_t}{b^2 y_t} \\
& \frac{dz}{dy} = -\frac{b^2 y}{a^2 z}
\end{aligned} \tag{24} \]

where \( z_t \) and \( y_t \) are the coordinates of the point shown in figure 6. The vector \( \vec{N}_t \) representing the normal from this point is

\[ \vec{N}_t = \hat{j} + m_{N_t} \hat{k} \tag{25} \]

and the unit vector \( \hat{N}_t \) becomes

\[ \hat{N}_t = \frac{\hat{j} + m_{N_t} \hat{k}}{\sqrt{1 + m_{N_t}^2}} \tag{26} \]

Since the payload surface is in the \( xy \) plane,

\[ \hat{N}_p = -\hat{k} \tag{27} \]
The vector $\vec{R}_{t,p}$ from $\Delta A_t$ to $\Delta A_p$ can be expressed as

$$\vec{R}_{t,p} = (x_p - x_t) \hat{i} + (y_p - y_t) \hat{j} + (z_p - z_t) \hat{k}$$

(28)

Since $x_t = 0$,

$$\vec{R}_{t,p} = x_p \hat{i} + (y_p - y_t) \hat{j} + (z_p - z_t) \hat{k}$$

(29)

$$R = \left| \sqrt{x_p^2 + (y_p - y_t)^2 + (z_p - z_t)^2} \right|$$

(30)

Thus the unit vector $\hat{R}_{t,p}$ becomes

$$\hat{R}_{t,p} = \frac{x_p \hat{i} + (y_p - y_t) \hat{j} + (z_p - z_t) \hat{k}}{R}$$

(31)

and

$$\hat{R}_{p,t} = -\hat{R}_{t,p}$$

(32)

The angles $\varphi_t$ and $\varphi_p$ can now be determined by forming scalar products:

$$\cos \varphi_t = \hat{N}_t \cdot \hat{R}_{t,p} = \frac{(y_p - y_t) + m_{N_t} (z_p - z_t)}{R \left| \sqrt{1 + m_{N_t}^2} \right|}$$

(33)

$$\cos \varphi_p = \hat{N}_p \cdot \hat{R}_{p,t} = \frac{z_p - z_t}{R}$$

The tank surface area $A$ of any ring is determined with the equation obtained from the following derivation:
\( dA = 2\pi y \, ds \)

\[ ds = \sqrt{(dy)^2 + (dz)^2} \]

\[ \frac{y^2}{a^2} + \frac{z^2}{b^2} = 1 \]

\[ y = a \sqrt{1 - \frac{z^2}{b^2}} \]

\[ dy = -az \, dz \]

\[ b^2 \, \sqrt{1 - \frac{z^2}{b^2}} \]
\[ \therefore \, dA = (2\pi) \left( a \sqrt{1 - \frac{z^2}{b^2}} \right) \left[ \sqrt{\frac{a^2 z^2}{b^4 \left( 1 - \frac{z^2}{b^2} \right)}} + 1 \right] \, dz \]

\[ \therefore \, dA = 2\pi a \left[ \sqrt{1 + \left( \frac{a^2}{b^4} - \frac{1}{b^2} \right) z^2} \right] \, dz \]

\[ \therefore \, A = 2\pi a \int_{z_i}^{z_{i+1}} \sqrt{1 + w^2 z^2} \, dz \]

where

\[ w^2 = \frac{a^2}{b^4} - \frac{1}{b^2} = \frac{a^2}{b^4} - \frac{b^2}{b^4} \]

Thus

\[ A_p = 2\pi a \left[ \frac{z}{2} \sqrt{1 + w^2 z^2} + \frac{1}{2w} \ln \left( zw + \sqrt{w^2 z^2 + 1} \right) \right]_{z_i}^{z_{i+1}} \]

(34)

Therefore all quantities required in equation (19) are determined.

To obtain the value of \[ A_{mF_{m,n}} \] between specific areas \( m \) and \( n \) is simply an appropriate summation of the results of equation (19):

\[ A_{mF_{m,n}} = \sum_{\text{All tank rings in the } m^{th} \text{ segment}} \sum_{\text{All payload rings in the } n^{th} \text{ segment}} A_{tF_{t,p}} \]

(35)
All other values of $A_{m}F_{m,n}$ can now be generated using the following relations:

1. Since no surface "sees" itself, $F_{m,m} = 0$ for any $m$.
2. Since no two surfaces on the tank "see" each other, $F_{m,n} = 0$ for all tank surfaces $m$ and $n$.
3. Since no two surfaces on the payload "see" each other, $F_{m,n} = 0$ for all payload surfaces $m$ and $n$.
4. By definition, $\sum_{n=1}^{\text{All surfaces}} F_{m,n} = 1$.
5. By reciprocity, $A_{m}F_{m,n} = A_{n}F_{n,m}$.

DESCRIPTION OF SUBROUTINES

As previously discussed, the program is divided into three phases: (1) the pre-preprocessor, which by using the input data described in appendix A of this report computes the input data needed to run a problem with CINDA-3G and generates a tape according to the formats given on pages 4.1 to 4.2 of reference 2; (2) the preprocessor, as described in reference 2; and (3) the execution phase, which actually computes the temperature distribution.

Pre-preprocessor Phase

The pre-preprocessor is composed of four subroutines which read data cards describing the geometry, compute the data required by the CINDA-3G system, and write this data onto a tape to be used in phase 2.

The subroutines in the pre-preprocessor are as follows:

<table>
<thead>
<tr>
<th>Subroutine name</th>
<th>FORTRAN call name</th>
<th>Brief description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREPRE</td>
<td>Main program</td>
<td>Reads the input data, computes the data required by CINDA-3G</td>
</tr>
<tr>
<td>VIEWFC</td>
<td>VIEWF</td>
<td>Computes the geometric view factors for the radiation interchange used in subroutine IRRAD1</td>
</tr>
<tr>
<td>RAYOUT</td>
<td>RITARY</td>
<td>Called by PREPRE; writes arrays of numbers onto the input tape as prescribed by CINDA-3G, p. 4.10 of ref. 2</td>
</tr>
<tr>
<td>COMDAT</td>
<td>BLOCK COMMON</td>
<td>Block common subroutine</td>
</tr>
</tbody>
</table>
Preprocessor Phase

The preprocessor is the second phase of the operation. A list of the subroutines and a description of each is given in reference 2 (pp. 8.1 to 8.3). The only variation is that the preprocessor reads the data from the tape that was prepared by the pre-preprocessor phase instead of from cards.

Execution Phase

The actual computing to determine the temperature distribution is done in the execution phase. In this execution phase the following subroutines are used: LINKO, EXECDK, VAR1DK, VAR2DK, OUTDK, which are generated by the preprocessor phase of CINDA-3G.

LINKO is the main program. Its only function is to call the INPUT routines to read in the CINDA input data and call EXECDK.

EXECDK is called the execution block of the program. It retains control until the problem is solved. Subroutines VAR1DK, VAR2DK, and OUTDK are called by EXECDK or by other subroutines called from EXECDK.

VAR1DK, VAR2DK, and OUTDK are the subroutines generated from the VARIABLES 1 block, VARIABLES 2 block, and OUTPUT block, respectively, that are described in reference 2. The order of calculation in the execution phase of the program is described, along with a flow chart, in reference 2 (pp. 3.12 and 3.13).

Several subroutines were written by the authors and incorporated into the execution phase in the user subroutine section. These subroutines may be easily modified or re-written by the user because they must be included into the program deck as FORTRAN source cards. The subroutines are

<table>
<thead>
<tr>
<th>Subroutine name</th>
<th>FORTRAN call name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNDCTR</td>
<td>GETG</td>
</tr>
<tr>
<td>OUTEND</td>
<td>ENDOUT</td>
</tr>
<tr>
<td>DISTRT</td>
<td>TDIS</td>
</tr>
<tr>
<td>GRAFIC</td>
<td>PICTR</td>
</tr>
<tr>
<td>PUNTEM</td>
<td>PCHT</td>
</tr>
</tbody>
</table>

Subroutine CNDCTR computes the value of all internal conduction connectors $G_{ij}$ between two nodes $i$ and $j$. 
\[ G_i = \frac{A_k}{l_{i,j}} \]  

where

\[ \begin{align*} 
A & \quad \text{area of heat flow} \\
 k & \quad \text{thermal conductivity} \\
l & \quad \text{distance from the center of node } i \text{ to center of node } j 
\end{align*} \]

If the thermal conductivity varies with temperature

\[ G_i = \frac{A}{l_{i} + l_{j}} \frac{k_{i}}{k_{j}} \]  

where \( k_{i} \) and \( k_{j} \) are temperature-varying thermal conductivities at nodes \( i \) and \( j \).

These conductors are computed just prior to each iteration of the network solution.

Subroutine CNDCTR as listed in this report was written to permit the use of very high thermal conductivity shields (pure aluminum as opposed to aluminized mylar) to increase the lateral thermal conductivity of some layers of the MLI. The nodes in these layers with high conductivity will be denoted as hi-k nodes. All other interior nodes with a lower conductivity will be denoted as standard nodes. It should be noted that this is conduction in the lateral direction only. The thermal conductivity for the standard nodes is determined by straight-line curve fit. Input variables CON1 and CON2 are the \( a \) and \( b \), respectively, for the line \( k = aT + b \). The data for the hi-k nodes are input in a table with variable name AKVST. The format for this table is found in appendix E. For the data used in the sample problem for both the standard nodes and the hi-k nodes see figure 8 of reference 1. The thermal conductivity for conduction in the normal direction \( k_{\text{effective}} \) is a combination of conduction and radiation (see ref. 1 for details). Subroutine CNDCTR also calls subroutine IRRAD1 to obtain the net \( Q \) into each surface of the radiosity enclosure.

Subroutine OUTEND calls subroutine GRAFIC and then computes and prints out the heating rates \( Q \) of the fuel tank to MLI, QNORM, and also the net heat into the MLI on the top surface of the MLI, QTS. QNORM and QTS are printed in tabular form to show the \( Q \) in or out of each segment on the tank. Further, these values are then multiplied by 8 to obtain the \( Q \) in and out for the entire tank. The names of these output variables are QN8 and QTS8. These printouts occur each time the print switch (NPRINT) calls for an output of the temperature distribution.
Subroutine TDIST reads the initial temperature distribution from tape unit SYSUT5 which was written in the pre-preprocessor.

Subroutine GRAFIC is a special subroutine which prints out the temperature distribution pictorially. Blocks are made, simulating the nodes, and are printed to simulate the shape of the shield. Inside each block is printed a node number and the temperature of that node. A node number with a minus sign in front denotes this node is a hi-k node, as described in the section on subroutine CNDCTR. If the top of the box is made up of dollar signs ($$$$$$) instead of asterisks (******) it denotes a surface with a high surface emissivity (see sample output listing on p. 31).

Subroutine PUNTEM punches cards with the current temperatures if either the number of iteration loops or the amount of computer time requested on the TCP card is exceeded.

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, December 2, 1970,
129-04.
# APPENDIX A

## COMPUTER PROGRAM SYMBOLS

<table>
<thead>
<tr>
<th>FORTRAN name</th>
<th>Corresponding mathematical symbol</th>
<th>Units in program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>( A_{\text{norm}} )</td>
<td>( \text{ft}^2 )</td>
<td>Conduction area in normal direction</td>
</tr>
<tr>
<td>AFA</td>
<td>( A_i \cdot F_{ij} )</td>
<td>( \text{ft}^2 )</td>
<td>Product of ( i )th surface area and the geometric view factor between ( i )th and ( j )th surfaces</td>
</tr>
<tr>
<td>AKVST</td>
<td>( k )</td>
<td>Btu/(hr)(ft)(( ^0 )F)</td>
<td>Thermal conductivity table</td>
</tr>
<tr>
<td>ARCLN</td>
<td>---</td>
<td>( \text{ft} )</td>
<td>Arc length</td>
</tr>
<tr>
<td>AREAL</td>
<td>( A_L )</td>
<td>( \text{ft}^2 )</td>
<td>Conduction area in lateral direction</td>
</tr>
<tr>
<td>BOTLIM</td>
<td>---</td>
<td>---</td>
<td>Integrand of equation (34) evaluated at lower limit ( z_i )</td>
</tr>
<tr>
<td>BTA</td>
<td>( \beta )</td>
<td>( \text{deg} )</td>
<td>Angle measured from minor axis to segment boundary on tank</td>
</tr>
<tr>
<td>C</td>
<td>---</td>
<td>---</td>
<td>Array of user constants to CINDA-3G</td>
</tr>
<tr>
<td>CINCON</td>
<td>---</td>
<td>---</td>
<td>Array of control constants to CINDA-3G</td>
</tr>
<tr>
<td>CONNAM</td>
<td>---</td>
<td>---</td>
<td>Array of control constant names</td>
</tr>
<tr>
<td>CON1</td>
<td>---</td>
<td>---</td>
<td>Constant = ( a ) \quad \text{a straight-line curve fit for} \quad k \quad \text{for lateral conduction;} \quad k = aT + b</td>
</tr>
<tr>
<td>CON2</td>
<td>---</td>
<td>---</td>
<td>Damping factor (CINDA control constant)</td>
</tr>
<tr>
<td>DAMPA</td>
<td>---</td>
<td>---</td>
<td>Incremental area on MLI for view factor computation</td>
</tr>
<tr>
<td>DAREA</td>
<td>( A_t )</td>
<td>( \text{ft}^2 )</td>
<td></td>
</tr>
</tbody>
</table>

20
<table>
<thead>
<tr>
<th>FORTRAN name</th>
<th>Corresponding mathematical symbol</th>
<th>Units in program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELDMP</td>
<td>---</td>
<td>---</td>
<td>Delta to add to change damping factor</td>
</tr>
<tr>
<td>DELS</td>
<td>---</td>
<td>ft</td>
<td>Thickness of each shield</td>
</tr>
<tr>
<td>DMLI</td>
<td>$\Delta S$</td>
<td>ft</td>
<td>Thickness of MLI at a given segment</td>
</tr>
<tr>
<td>DPTH</td>
<td>---</td>
<td>ft</td>
<td>Thickness (depth) of each node</td>
</tr>
<tr>
<td>EMISP</td>
<td>$\epsilon$</td>
<td>---</td>
<td>Emissivity of each surface of payload</td>
</tr>
<tr>
<td>EMISS</td>
<td>$\epsilon$</td>
<td>---</td>
<td>Emissivity of space</td>
</tr>
<tr>
<td>EMIST</td>
<td>$\epsilon$</td>
<td>---</td>
<td>Emissivity of each surface on MLI</td>
</tr>
<tr>
<td>G</td>
<td>G</td>
<td>Btu/(hr)$\left(^{0}\text{F}\right)$</td>
<td>Conductor value for heat transfer</td>
</tr>
<tr>
<td>H</td>
<td>h</td>
<td>ft</td>
<td>Tank-to-payload spacing</td>
</tr>
<tr>
<td>IEXECN</td>
<td>---</td>
<td>---</td>
<td>Name of CINDA execution routine to be used</td>
</tr>
<tr>
<td>LCK2</td>
<td>---</td>
<td>---</td>
<td>Tape unit SYSCK2 CINDA input written on</td>
</tr>
<tr>
<td>LUT3</td>
<td>---</td>
<td>---</td>
<td>Tape unit SYSUT3 used as temporary storage for FORTRAN subroutines to be used in execution phase</td>
</tr>
<tr>
<td>LUT5</td>
<td>---</td>
<td>---</td>
<td>Tape unit SYSUT5 initial temperatures stored on this unit, to be read in execution phase</td>
</tr>
<tr>
<td>MAXNL</td>
<td>---</td>
<td>---</td>
<td>Maximum number of nodes at any segment</td>
</tr>
<tr>
<td>NDAMP</td>
<td>---</td>
<td>---</td>
<td>Change damping factor every NDAMP iterations</td>
</tr>
<tr>
<td>NHIK</td>
<td>---</td>
<td>---</td>
<td>Number of high-conductivity layers</td>
</tr>
<tr>
<td>FORTRAN name</td>
<td>Corresponding mathematical symbol</td>
<td>Units in program</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>NIN</td>
<td>---</td>
<td>---</td>
<td>Tape unit, input unit</td>
</tr>
<tr>
<td>NL</td>
<td>---</td>
<td>---</td>
<td>Number of nodes in MLI in normal direction for each segment</td>
</tr>
<tr>
<td>NNODES</td>
<td>---</td>
<td>---</td>
<td>Total number of nodes in configuration</td>
</tr>
<tr>
<td>NNPAY</td>
<td>---</td>
<td>---</td>
<td>Number of nodes on payload</td>
</tr>
<tr>
<td>NNSEG</td>
<td>---</td>
<td>---</td>
<td>Number of segments on tank</td>
</tr>
<tr>
<td>NOFA</td>
<td>---</td>
<td>---</td>
<td>Number of entries in view factor matrix</td>
</tr>
<tr>
<td>NOUT</td>
<td>---</td>
<td>---</td>
<td>Tape unit, output unit</td>
</tr>
<tr>
<td>NPAY</td>
<td>---</td>
<td>---</td>
<td>Same as NNPAY above</td>
</tr>
<tr>
<td>NPRINT</td>
<td>---</td>
<td>---</td>
<td>Print temperatures every NPRINT iterations</td>
</tr>
<tr>
<td>NSEG</td>
<td>---</td>
<td>---</td>
<td>Same as NNSEG above</td>
</tr>
<tr>
<td>NSPL</td>
<td>---</td>
<td>---</td>
<td>Number of shields in each node</td>
</tr>
<tr>
<td>NSTMTS</td>
<td>---</td>
<td>---</td>
<td>Number of FORTRAN cards read in as user program to go into SUBROUTINES block of CINDA-3G</td>
</tr>
<tr>
<td>NSURF</td>
<td>---</td>
<td>---</td>
<td>Number of surfaces in radiation enclosure</td>
</tr>
<tr>
<td>NTAB</td>
<td>---</td>
<td>---</td>
<td>Total number of entries in K-vs-T table (AKVST)</td>
</tr>
<tr>
<td>NUMHIIK</td>
<td>---</td>
<td>---</td>
<td>Array containing layer numbers of high-conductivity layers</td>
</tr>
<tr>
<td>PSI</td>
<td>$\psi_m$</td>
<td>---</td>
<td>Angle from top of tank to $R_1$</td>
</tr>
<tr>
<td>QNORM</td>
<td>Q</td>
<td>Btu/hr</td>
<td>Heat flow from MLI into tank</td>
</tr>
<tr>
<td>FORTRAN name</td>
<td>Corresponding mathematical symbol</td>
<td>Units in program</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>QTS</td>
<td>Q</td>
<td>Btu/hr</td>
<td>Heat flow from enclosure into top of MLI</td>
</tr>
<tr>
<td>R1</td>
<td>R₁</td>
<td>ft</td>
<td>Distance from tank center to incremental surface i</td>
</tr>
<tr>
<td>R1X</td>
<td>R₁</td>
<td>ft</td>
<td>Distance from tank center to incremental surface i + 1</td>
</tr>
<tr>
<td>RP</td>
<td>Rₚ</td>
<td>ft</td>
<td>Radius of payload</td>
</tr>
<tr>
<td>SLAY</td>
<td>---</td>
<td>---</td>
<td>Number of nodes in MLI in normal direction at each 1º increment</td>
</tr>
<tr>
<td>SPACEE</td>
<td>ε</td>
<td>---</td>
<td>Emissivity of space</td>
</tr>
<tr>
<td>TCARDS</td>
<td>---</td>
<td>---</td>
<td>Switch: If equal to zero, use temperatures input with input data; then distribute initial temperatures throughout configuration with subroutine TDIST. If not equal to zero, read initial temperature distribution from cards.</td>
</tr>
<tr>
<td>THETA</td>
<td>θ</td>
<td>deg</td>
<td>Array of angles specifying angular segments on tank</td>
</tr>
<tr>
<td>THIKN</td>
<td>---</td>
<td>---</td>
<td>Same as DPTH above</td>
</tr>
<tr>
<td>TPAY</td>
<td>T</td>
<td>°R</td>
<td>Array temperatures of payload nodes</td>
</tr>
<tr>
<td>TSPACE</td>
<td>T</td>
<td>°R</td>
<td>Temperature of space node</td>
</tr>
<tr>
<td>TSURF</td>
<td>T</td>
<td>°R</td>
<td>Initial temperature on MLI surface</td>
</tr>
<tr>
<td>TWALL</td>
<td>T</td>
<td>°R</td>
<td>Temperature of tank wall</td>
</tr>
<tr>
<td>UPLIM</td>
<td>---</td>
<td>---</td>
<td>Integrand of equation (34) evaluated at upper limit</td>
</tr>
<tr>
<td>X</td>
<td>Xᵣ</td>
<td>ft</td>
<td>Major semi-axis of tank</td>
</tr>
<tr>
<td>FORTRAN name</td>
<td>Corresponding mathematical symbol</td>
<td>Units in program</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------</td>
<td>------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>XM1</td>
<td>$m_{N_1}$</td>
<td>--</td>
<td>Slope</td>
</tr>
<tr>
<td>Y</td>
<td>$Y_r$</td>
<td>ft</td>
<td>Minor semi-axis of tank</td>
</tr>
</tbody>
</table>
APPENDIX B

FLOW DIAGRAM

General program

START

Pre-preprocessor phase:
Read input data and generate data for pre-processor

Pre-processor phase

Subroutine VIEWFC
Subroutine RAYOUT

Execution phase

Subroutine DISTRT

Execution routine

Variables 1

Variables 2

Subroutine CNDCTR

Subroutine OUTEND

Subroutine GRAFIC

Output phase - tabular output

STOP

Pre-preprocessor

START

Read input data

Initialize

Generate node data block

Distribute temperatures linearly from cold to hot vertically

Write initial temperatures onto tape II

Compute areas for normal and lateral conduction

Compute radiation view factor in subroutine VIEWF

Generate conductor data block

Generate constants data block

Generate array data block

Generate execution, variables 1, variables 2, and output blocks

Write end of file on data tape

Write user subroutines onto data tape
APPENDIX C

DECK SETUP

The following is the deck setup to run this program at Lewis Research Center on the IBM 7094-7044 Direct Couple System using IBSYS version 13:

```
cc1 cc8 cc16
$ID JO USER NAME
$TCP TIME=XX, PAGES=YY
$SETUP 10 CIN3G
$IBJOB

[FORTRAN IV source or binary decks of pre-preprocessor as described in this report, (Includes subroutines PREPRE, VIEWFC, RAYOUT, and COMDAT.)]

$DATA

[FORTRAN IV source cards of user-written subroutines. For this problem these subroutines include CNDCTR, OUTEND, DISTRT, GRAFIC, and PUNTEM. Any other user-written subroutines may be added here.]

ENDX

(Input data as described in appendix E.)

$IBSYS

$REWIND SYSCK2

$SWITCH SYSIN1, SYSCK2
```
APPENDIX D

TAPE USAGE

Tape units must be made compatible between CINDA-3G and the computer system. A table of tape usage presently being used at Lewis Research Center only with the CINDA-3G program is included to aid in making this compatible with other systems.

<table>
<thead>
<tr>
<th>SYSUNI</th>
<th>FORTRAN number in CINDA-3G</th>
<th>Variable name in program</th>
<th>Lewis system number</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>UT3</td>
<td>2</td>
<td>LUT3</td>
<td>14</td>
<td>Copy of original problem data</td>
</tr>
<tr>
<td>UT4</td>
<td>3</td>
<td>LUT4</td>
<td>15</td>
<td>Parameter change data*</td>
</tr>
<tr>
<td>UT8</td>
<td>4</td>
<td>LUT1</td>
<td>3</td>
<td>Data number definitions</td>
</tr>
<tr>
<td>IN1</td>
<td>5</td>
<td>NIN</td>
<td>5</td>
<td>Input</td>
</tr>
<tr>
<td>OU1</td>
<td>6</td>
<td>NOUT</td>
<td>6</td>
<td>Output</td>
</tr>
<tr>
<td>UT2</td>
<td>8</td>
<td>LUT2</td>
<td>13</td>
<td>NA-NB pairs, data number definitions for parameter changes</td>
</tr>
<tr>
<td>CK1</td>
<td>9</td>
<td>-----</td>
<td>10</td>
<td>CINDA-3G master tape</td>
</tr>
<tr>
<td>CK2</td>
<td>10</td>
<td>-----</td>
<td>16</td>
<td>New master tape if updating, also used as problem recall data tape*</td>
</tr>
<tr>
<td>UT5</td>
<td>11</td>
<td>-----</td>
<td>7</td>
<td>Problem store data tape*</td>
</tr>
<tr>
<td>UT6</td>
<td>12</td>
<td>LBD3</td>
<td>1</td>
<td>Data tape (original problem and all parameter changes)</td>
</tr>
<tr>
<td>PP1(LB4)</td>
<td>13</td>
<td>-----</td>
<td>Calcomp</td>
<td>System plot tape</td>
</tr>
<tr>
<td>LB3</td>
<td>14</td>
<td>LB4P</td>
<td>9</td>
<td>Program tape (contains generated FORTRAN routines; LINKO, EXECTN, VARBL1, VARBL2, OUTCAL)</td>
</tr>
<tr>
<td>SYSUNI name</td>
<td>FORTRAN number in CINDA-3G</td>
<td>Variable name in program</td>
<td>Lewis system number</td>
<td>Function</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------</td>
<td>--------------------------</td>
<td>--------------------</td>
<td>----------</td>
</tr>
<tr>
<td>UT7</td>
<td>15</td>
<td>LUT7</td>
<td>2</td>
<td>Variables 1 calls generated from node and conductor data blocks*</td>
</tr>
<tr>
<td>UT9</td>
<td>16</td>
<td>-----</td>
<td>4</td>
<td>Internal file for reread</td>
</tr>
<tr>
<td>UT9</td>
<td>99</td>
<td>-----</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>LB2</td>
<td>17</td>
<td>-----</td>
<td>8</td>
<td>Overlay tape</td>
</tr>
</tbody>
</table>

The tapes marked by an asterisk need not be assigned if the particular options are not used. The STOREP option in CINDA-3G requires assigning and saving tapes on units LB3 and UT5. The RECALL option requires assigning and mounting these two tapes onto units LB3 and CK2, respectively.
APPENDIX E

DESCRIPTION AND FORMAT OF INPUT DATA TO PRE-PREPROCESSOR

Input Data Description

CARD 1 FORMAT( I4, 2I3, 7F10.5)
NSEG  NO. OF SEGMENTS  (MAXIMUM 30)
NPAY  NO. OF SURFACES, (SEGMENTS) ON PAYLOAD (MAXIMUM 10)
NSPL  NO. OF SHIELDS PER LAYER
DELS  THICKNESS OF EACH (SHIELD+SPACER)  FT.
TWALL  TEMPERATURE OF TANKWALL  DEG.R
TSURF  TEMPERATURE OF TOP SURFACE OF SHIELD  DEG.R
TS  TEMPERATURE OF SPACE  DEG.R
EMISS  EMISIVITY USED IN EQUATION FOR K(EFF), CONDUCTION IN NORMAL DIRECTION
SPACEE  EMISIVITY OF SPACE
RP  RADIUS OF PAYLOAD  FT.

CARD 2 FORMAT( 6F10.5, A6)
TCARDS  SWITCH TO READ INITIAL TEMPERATURES
IF TCARDS =O, CARDS WILL NOT BE READ. THE PROGRAM USES TWALL AND TSURF AND DISTRIBUTES THE TEMPERATURES BETWEEN THESE TWO VALUES AS THE INITIAL TEMPERATURE DISTRIBUTION.
IF TCARDS .GT. O THE INITIAL TEMPERATURE DISTRIBUTION IS READ FROM CARDS, (DEG. R). SEE CARDS 11 BELOW.
X  LENGTH OF SEMI-AXIS OF ELIPSE, (TANKWALL),ALONG X AXIS
Y  LENGTH OF SEMI-AXIS OF ELIPSE, (TANKWALL),ALONG Y AXIS
H  TANK TO PAYLOAD SPACING  FT.
CON1  CONSTANTS USED FOR STRAIGHT LINE CURVE FIT TO COMPUTE CON2  THERMAL CONDUCTIVITY FOR G VALUES FOR LATERAL CONDUCTION OF STANDARD NODES. K=CON1*T+CON2.
IEXECN  NAME OF EXECUTION ROUTINE  CINDSS OR CINDSL

CARD 3 FORMAT( 2110, F10.5)
NPRINT  DELTA PRINT, PRINT TEMP. DIST. EVERY NPRINT ITERATIONS
NDAMP  CHANG DAMPING FACTOR EVERY NDAMP ITERATIONS
DELDMP  DELTA TO ADD TO DAMPING FACTOR

CARD 4 FORMAT(I8,9F8.3)
GICNCON(I)  CINDA CONTROL CONSTANTS, AS REQUIRED BY EXECUTION ROUTINE BEING USED. CONSTANTS INPUT IN ORDER AS FOLLOWS ACCORDING TO FORMAT GIVEN ABOVE,
NLOOP, ARLXCA, DAMPA, DRLXCA, DAMPD, TIMEO, TIMEND, OUTPUT, DTIMEI, CSGFAC.

CARD 5 NTAB, (AKVST(I),I=1,NTAB)  FORMAT(I6,12F6.1/(6X,12F6.1))
TABLE OF TEMPERATURE VS. THERMAL CONDUCTIVITY FOR LATERAL CONDUCTION OF HIGH CONDUCTIVITY LAYERS.
NTAB= TOTAL NO. OF ENTRIES IN TABLE (TEMPS AND K'S)
DATA FORM NTAB TEMP1  K1 TEMP2  K2 --- TEMPN  KN
UNITS-- K BTU/HR-FT-DEG.R
T DEG.R

CARD 6 NL  NUMBER OF LAYERS IN EACH SEGMENT FORMAT ( NI2)
CARDS 7 THETAD ANGLE OF EACH SEGMENT, IN DEGREES, IF ALL SEGMENTS ARE EQUAL INPUT ONLY THE FIRST. FORMAT (N I2) THE SUM OF THE N ANGLES MUST EQUAL 90

CARDS 8 ON TWO CARDS (10F8.5)
EMISIP(I) EMISIVITY OF EACH PAYLOAD SEGMENT (MAXIMUM 10)
TPAY(I) TEMPERATURE OF EACH PAYLOAD SEGMENT (MAXIMUM 10)

CARDS 9 EMISST EMISSIVITY OF EACH SURFACE, RADIATION (8F10.5)

CARD 10 NHIK NO. OF HIGH CONDUCTIVITY LAYERS (30I2)
NUMHIK LAYER NUMBERS WHICH ARE HI COND.

CARDS 11 VNODES IF INITIAL TEMPS READ FROM CARDS - NO. OF NODES T(I) INITIAL TEMPERATURES, DEG. R (18,9F8.1/10F8.1)

---

Format of Input Data

---

NASA-C-3166 REV 9-45-350
APPENDIX F

INPUT DATA FOR SAMPLE PROBLEM

<table>
<thead>
<tr>
<th>SHIELDING STUDY CONFIGURATION NO. 24</th>
</tr>
</thead>
<tbody>
<tr>
<td>NODE NO.</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>125</td>
</tr>
<tr>
<td>9999</td>
</tr>
<tr>
<td>104</td>
</tr>
<tr>
<td>96</td>
</tr>
<tr>
<td>94</td>
</tr>
<tr>
<td>44</td>
</tr>
<tr>
<td>43</td>
</tr>
<tr>
<td>42</td>
</tr>
<tr>
<td>41</td>
</tr>
<tr>
<td>40</td>
</tr>
<tr>
<td>39</td>
</tr>
<tr>
<td>38</td>
</tr>
<tr>
<td>37</td>
</tr>
<tr>
<td>36</td>
</tr>
<tr>
<td>35</td>
</tr>
<tr>
<td>34</td>
</tr>
<tr>
<td>33</td>
</tr>
<tr>
<td>32</td>
</tr>
<tr>
<td>31</td>
</tr>
<tr>
<td>30</td>
</tr>
<tr>
<td>29</td>
</tr>
<tr>
<td>28</td>
</tr>
<tr>
<td>27</td>
</tr>
<tr>
<td>26</td>
</tr>
<tr>
<td>25</td>
</tr>
<tr>
<td>24</td>
</tr>
<tr>
<td>23</td>
</tr>
<tr>
<td>22</td>
</tr>
<tr>
<td>21</td>
</tr>
<tr>
<td>20</td>
</tr>
<tr>
<td>19</td>
</tr>
<tr>
<td>18</td>
</tr>
<tr>
<td>17</td>
</tr>
<tr>
<td>16</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>13</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>8</td>
</tr>
</tbody>
</table>

OUTPUT FOR SAMPLE PROBLEM

<table>
<thead>
<tr>
<th>SHIELDING STUDY CONFIGURATION NO. 24</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPACE</td>
</tr>
<tr>
<td>TEMP= 0.</td>
</tr>
<tr>
<td>EMISIV= 1.0000</td>
</tr>
</tbody>
</table>

ANGLE OF EACH SEGMENT (DEG.)

| ANGLE | 27.0 | 15.0 | 9.0 | 9.0 | 6.0 | 6.0 | 6.0 | 4.0 | 4.0 |

EMISIVITY OF EACH SEGMENT

| EMISIVITY | 0.0240 | 0.0240 | 0.0240 | 0.0240 | 0.0240 | 0.0240 | 0.0240 | 0.0240 | 0.0240 |

TANK TO PAYLOAD SPACING= 0.8175 FT.

<table>
<thead>
<tr>
<th>LAYER NO.</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
</tr>
<tr>
<td>23</td>
</tr>
<tr>
<td>22</td>
</tr>
<tr>
<td>21</td>
</tr>
</tbody>
</table>

31
<table>
<thead>
<tr>
<th>TANKWALL</th>
<th>1</th>
<th>19</th>
<th>37</th>
</tr>
</thead>
<tbody>
<tr>
<td>SURFACE</td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26</td>
<td>37</td>
</tr>
</tbody>
</table>

**TANKWALL**

- **NODE NO.** = 178
- **TEMP.** = 37.00

---

**Data Table**

<table>
<thead>
<tr>
<th>Layer</th>
<th>Temperature (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>37.00</td>
</tr>
<tr>
<td>2</td>
<td>37.00</td>
</tr>
<tr>
<td>3</td>
<td>37.00</td>
</tr>
<tr>
<td>4</td>
<td>37.00</td>
</tr>
<tr>
<td>5</td>
<td>37.00</td>
</tr>
<tr>
<td>6</td>
<td>37.00</td>
</tr>
<tr>
<td>7</td>
<td>37.00</td>
</tr>
<tr>
<td>8</td>
<td>37.00</td>
</tr>
<tr>
<td>9</td>
<td>37.00</td>
</tr>
<tr>
<td>10</td>
<td>37.00</td>
</tr>
<tr>
<td>11</td>
<td>37.00</td>
</tr>
<tr>
<td>12</td>
<td>37.00</td>
</tr>
<tr>
<td>13</td>
<td>37.00</td>
</tr>
<tr>
<td>14</td>
<td>37.00</td>
</tr>
<tr>
<td>15</td>
<td>37.00</td>
</tr>
<tr>
<td>16</td>
<td>37.00</td>
</tr>
<tr>
<td>17</td>
<td>37.00</td>
</tr>
<tr>
<td>18</td>
<td>37.00</td>
</tr>
<tr>
<td>19</td>
<td>37.00</td>
</tr>
<tr>
<td>20</td>
<td>37.00</td>
</tr>
</tbody>
</table>

---

**Legend**

- **TANKWALL**
- **NODE NO.**
- **TEMP.**
<table>
<thead>
<tr>
<th>NO. SEG</th>
<th>QTS</th>
<th>QNORM</th>
<th>QTS*8</th>
<th>QNORM*8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.06936889</td>
<td>0.00018959</td>
<td>0.55495113</td>
<td>0.00151672</td>
</tr>
<tr>
<td>2</td>
<td>0.10318061</td>
<td>0.00029260</td>
<td>0.82544488</td>
<td>0.00234082</td>
</tr>
<tr>
<td>3</td>
<td>0.09149027</td>
<td>0.00025234</td>
<td>0.73192215</td>
<td>0.00202030</td>
</tr>
<tr>
<td>4</td>
<td>0.04330576</td>
<td>0.00032500</td>
<td>0.36644613</td>
<td>0.00260004</td>
</tr>
<tr>
<td>5</td>
<td>0.04622486</td>
<td>0.00023741</td>
<td>0.36979891</td>
<td>0.00189930</td>
</tr>
<tr>
<td>6</td>
<td>0.00980636</td>
<td>0.00027727</td>
<td>0.07845092</td>
<td>0.00221822</td>
</tr>
<tr>
<td>7</td>
<td>-0.20052147</td>
<td>0.00028314</td>
<td>-1.60417175</td>
<td>0.00226517</td>
</tr>
<tr>
<td>8</td>
<td>-0.09447652</td>
<td>0.00018922</td>
<td>-0.75581217</td>
<td>0.00151382</td>
</tr>
<tr>
<td>9</td>
<td>-0.04718843</td>
<td>0.00019050</td>
<td>-0.37750745</td>
<td>0.00152403</td>
</tr>
<tr>
<td>10</td>
<td>-0.01882567</td>
<td>0.00018001</td>
<td>-0.15060541</td>
<td>0.00144006</td>
</tr>
</tbody>
</table>

SUMMATION 0.23647E-02 0.24173E-02 0.18917E-01 0.19339E-01
APPENDIX G
FORTRAN LISTING OF AUTHOR-WRITTEN SUBROUTINES

$IBFTC$ PREPRE

COMMON/BCD/ NBCD(60)
COMMON /TAPE/NIN,NOUT,LCK2
COMMON /RADIO/ NSEG,NSPAY,NSURF,X,Y,H,RP,THIKN,PIE,SLAY(90),
1 THETAD(30),A(41),AFA(1681)
DIMENSION ID(14),BTA(31),ERAD(31),THETA(30),F(30),IEST(50),NL(30)
DIMENSION ARCTN(30),AREAL(30),NODD(30),NUMHIC(30),C(36)
DIMENSION AVST(200), EMIST(41), TPAY(10), EMISP(10)
DIMENSION ISTMT(14), T(1545), CONNAM(10), CINCON(10)
DIMENSION IFMTI(7), IFMT4(4), IFMTH(5)
DIMENSION ZI(9),Z2(Z1),Z3(9),Z4(9),Z5(9),Z6(9),Z7(9),Z8(9)
DIMENSION Z9(Z1),Z10(Z1),Z11(9)
EQUIVALENCE (NNSEG,C), (NNPAY,C(2)), (NSURF,C(3)), (NMODES,C(4)),
X (MAXL,C(5)), (NPRINT,C6), (NDAMP,C(17)), (NNL,C(8)), (NREAL,C(9)),
X (NAREON,C(10)), (NARCTN,C(11)), (NTHETAC(12)), (NTEST,C(13)),
X (NKHIC,C(14)), (NEMISS,C(15)), (NAFAC(16)), (NSNT,C(17)),
X (SIGMA,C(21)), (OPTH,C(22)), (DELS,C(23)), (EMISS(C24)),
X (CON1,C25)), (CON2,C26)), (DELMPC(27)),
X (TABS,C(29)), (HIKCON,C(30)), (PSWITCH,C(31)), (RAD,C(32)),
X , (HH,C(33))
DATA IFMTI(1)/ 42H(121 F2.0/(10F8.5)) /
DATA IFMTH(1)/ 30H17X,17HBCD 5THERMA /,
DATA CONNAM(1)/60H 1NLOOPARLXCA 1MDAMP 1X /,
DATA M1, ONE, ONEM, TEN209 PI, TABS, SIGMA /,
DATA 1 Z1(1)/ 54H NL, NO. OF LAYERS IN EACH SEGMENT /,
2 Z2(1)/ 54H AREAL, AREA, LATERAL CONDUCTION, EACH SEGMENT (HORIZ) /,
3 Z3(1)/ 54H A, AREA, NORMAL CONDUCTION, EACH SEGMENT (VERT) /,
4 Z4(1)/ 54H ARCLN, ARC LENGTH, LENGTHS FOR LATERAL CONDUCTION /,
5 Z5(1)/ 54H THETA (DEG.), ANGLE OF EACH SEGMENT /,
6 Z6(1)/ 54H IEST, NEGATIVE NO. IS HI-COND. LAYER, LATERAL CONDCT/,
7 Z7(1)/ 54H AVST, K(BTU/HR.FT) VS T(1DEG R) FOR HI-K LAYERS /,
8 Z8(1)/ 54H EMIST, EMMISIVITY OF SHIELD, RADIATION. /,
9 Z9(1)/ 54H AF, AREA*VIEWFACTOR FOR RADIOSITY, FOR EACH SEGMENT /,
A Z10(1)/ 54H TABLE OF SURFACE NODE NUMBERS, SURFACE EACH SEGMENT /,
PIE=PI
P104= PI/4.
NIN= 5
NOUT= 6
LUT3=2
LCK2=10
LUT5=11
REWIND LCK2
REWIND LUT3
REWIND LUT5
DO 12 I=1,10
12 CINCON(I)=0.
C
C WRITE EXECUTE, ID, IBJOB CARDS. THEN ALL CINDA3G PREPROCESSOR

34
C CARDS ON TO TAPE SYSCK2 (9)
C WRITE(LCK2,2000)
C READ PRE-PREPROCESSOR DATA AND GENERATE CINDA DATA
C
I=1
1 READ(NIN,2018) ISTMT
 WRITE(LUT3) ISTMT
 IF(ISTMT(2) .EQ. IENDX) GO TO 7
 I= I+1
 GO TO 1
7 NSTMTS= I
 REWIND LUT3
 READ(VIN,1000) ID

C INPUT DATA DESCRIPTION
C CARD 1 FORMAT(I4, 2I3, 7F10.5)
 C NSEG NO. OF SEGMENTS (MAXIMUM 30)
 C NPAY NO. OF SURFACES, (SEGMENTS) ON PAYLOAD (MAXIMUM 10)
 C NSPL NO. OF SHIELDS PER LAYER
 C DELS THICKNESS OF EACH (SHIELD+SPACER) FT.
 C TWALL TEMPERATURE OF TANKWALL Deg.R
 C TSURF TEMPERATURE OF TOP SURFACE OF SHIELD Deg.R
 C TSPACE TEMPERATURE OF SPACE Deg.R
 C EMISS EMISIVITY USED IN EQUATION FOR K(EFF),
 CONDUCTION IN NORMAL DIRECTION
 C SPACEE EMISIVITY OF SPACE
 C RP RADIUS OF PAYLOAD FT.

C CARD 2 FORMAT(6F10.5, A6)
 C TCARDS SWITCH- TO READ INITIAL TEMPERATURES
 IF TCARDS =0, CARDS WILL NOT BE READ. THE PROGRAM
 USES TWALL AND TSURF AND DISTRIBUTES THE TEMPERATURES
 BETWEEN THESE TWO VALUES AS THE INITIAL TEMPERATURE
 DISTRIBUTION.
 IF TCARDS .GT. 0 THE INITIAL TEMPERATURE DISTRIBUTION
 IS READ FROM CARDS, (DEG. R). SEE CARDS 11 BELOW.
 X LENGTH OF SEMI-AXIS OF ELIPSE, (TANKWALL) ALONG X AXIS
 Y LENGTH OF SEMI-AXIS OF ELIPSE, (TANKWALL) ALONG Y AXIS
 H TANK TO PAYLOAD SPACING FT.
 C CON1 CONSTANTS USED FOR STRAIGHT LINE CURVE FIT TO COMPUTE
 C CON2 THERMAL CONDUCTIVITY FOR G VALUES FOR LATERAL
 C CONDUCTION OF STANDARD NODES. K=CON1*T+CON2.
 C IEEXECN NAME OF EXECUTION ROUTINE CINDSS OR CINDSL

C CARD 3 FORMAT(2I10, F10.5)
 C NPRINT DELTA PRINT, PRINT TEMP. DIST. EVERY NPRINT ITERATIONS
 C NDAMP CHANG DAMPING FACTOR EVERY NDAMP ITERATIONS
 C DDLMP DELTA TO ADD TO DAMPING FACTOR

C CARD 4 FORMAT(I8, 9F8.3)
 C CINDA CONTROL CONSTANTS, AS REQUIRED BY EXECUTION
 ROUTINE BEING USED. CONSTANTS INPUT IN ORDER AS
 FOLLOWS ACCORDING TO FORMAT GIVEN ABOVE,
 C NLOOP, ARXLCAA, DAMPA, DRLXCA, DAMPD, TIMED,
 C TIMEND, OUTPUT, DTMEI, CSGFAC.

C CARD 5 NTAB, (AKVST(I),I=1,NTAB) FORMAT(I6, 12F6.1/(6X, 12F6.1))
 C TABLE OF TEMPERATURE VS. THERMAL CONDUCTIVITY FOR LATERAL CONDUCTION OF HIGH CONDUCTIVITY LAYERS.
 C NTAB= TOTAL NO. OF ENTRIES IN TABLE (TEMPS AND K'S)
DATA FORM  NTAB  TEMP1  K1  TEMP2  K2 --- TEMPN  KN
UNITS--  K BTU/HR-FT-DEG.R
T  DEG.R

CARDS 6  YL  NUMBER OF LAYERS IN EACH SEGMENT  FORMAT (NI2)
CARDS 7  THETAD  ANGLE OF EACH SEGMENT, IN DEGREES, IF ALL SEGMENTS ARE EQUAL  INPUT ONLY THE FIRST.  FORMAT (IN 12)
THE SUM OF THE N ANGLES MUST EQUAL 90
CARDS 8  ON TWO CARDS  (10F8.5)
EMISP(I)  EMISIVITY OF EACH PAYLOAD SEGMENT  (MAXIMUM 10)
TPAY(I)  TEMPERATURE OF EACH PAYLOAD SEGMENT  (MAXIMUM 10)
CARDS 9  EMIST  EMISSIVITY OF EACH SURFACE, RADIATION  (8F10.5)
CARD 10  WHIK  NO. OF HIGH CONDUCTIVITY LAYERS  (30I2)
NUMHIK  LAYER NUMBERS WHICH ARE HI COND.
CARDS 11  WNODES IF INITIAL TEMPS READ FROM CARDS - NO. OF NODES
T(I)  INITIAL TEMPERATURES,  DEG. R  (18,9F8.1/10F8.1)

READ(YIN,1001)  NSEG,NPAY,NSPL,DELS,TWALL,TSURF,TSPACE,EMISS,
SPACE,RP,TCARDS,X,Y,H,CON1,CON2,IEEXECN,NPRINT,NDAMP,
2  DELOMP,CINCON
DPTH=DELS*FLOAT(NSPL)
THIKN= DPTH
HH=H .
NSEG=VSEG
NPAY= NPAY
NODE=3.0
READ(5,1003)  NTAB, (AKVST(I),I=1,NTAB)
HIKCON=AKVST(NTAB)
IF(NSEG.GT.30)  NSEG=30
IFMT1(2) = NBCDNSEG)
IFMT1(4) = NBCDNSEG)
READ(NIN,IFMT1) (NL(I),I=1,NSEG),(THETAD(I),I=1,NSEG),EMISP,TPAY
READ(NIN,1004) (EMIST(I),I=1,NSEG)
READ(NIN,1002)  NHIK, (NUMHIK(I),I=1,NHIK)
NSURF= NSEG + NPAY + 1
NSURFS=NSURF
NOFA= (NSURF*(NSURF+1)/2)+2
J=NSEG+1
MAXNL=)
DO 201 I=1,NSEG
201 IF(NL(I),GT. MAXNL) MAXNL=NL(I)
DO 200 I=J,NSURF
M=I-J+1
200 EMIST(I)= EMISP(M)
EMIST(NSURFS)=SPACEE
DO 2  I=1,50
2 ITESTM= I
IF(NHIK,LE. 0) GO TO 4
DO 3  I=1,NHIK
M= NUMHIK(I)
3 ITESM= -ITEST(M)
4 CONTINUE
IFMTH(4)=ISPS
IF(IEEXECN .EQ. ISL .OR. IEXECN .EQ. IMBK .OR. IEXECN .EQ. IBACK
1 .OR. IEXECN .EQ. IVARB) IFMTH(4)=ILPS
IF(THEHAD(2) .GT. 0.) GO TO 5
DO 6  I=2,NSEG
6 THEHAD(I)= THEHAD(1)
5 BTA(1) = 0.
9 BTA(I) = BTA(I-1) + THETA(I-1)
BTA(NSEG+1) = 90.0
WRITE(LCK2,1FMT)
WRITE(LCK2,2001)
NSEGPI = NSEG+1
7001 NNL = 1
7002 NREAL = NSEG+2
7003 NAREAL = NREAL*NSEGPI
7004 NARCLN = NAREAL+NSURF+1
7005 NTHETA = NARCLN+NSEGPI
7006 NITEST = NTHETA+NSEGPI
7007 NKhIC = NITEST+51
7008 NEmISS = NKHIC+NTAB+1
7009 NAFa = NEmISS+NSURF+1
7010 NSNt = NAFa+NOFA+1
C
C GENERATE NODE DATA
C NODE NO. FOR SPACE IS NNODES
C NODE NO. FOR FIRST PAYLOAD NODE IS (NNODES-NPAY)
C NODE NO. FOR TANKWALL IS (NNODES-NPAY-1)
C
C SHIELD NODES
C
WRITE(LCK2,2032)
101 NNODES = 0
C
C GENERATE INTERIOR NODES
C
DO 10 I=1,NSEG
10 NNODES = NNODES + NL(I)
NINSN = NNODES
C
ADD NO. OF SURFACE NODES TO NNODES, SURF NODES ONLY ALONG TANKWALL
C
NNODES = NNODES + NSEG
TEMP = TNODE
WRITE(LCK2,2003),NNODES,M1,TEMP,ONE,ONE,ONE,ONEM
C
C TANKWALL NODE
C
NNODES = NNODES-1
TEMP = TWALL
WRITE(LCK2,2004) NNODES,TEMP,ONE
C
C PAYLOAD NODES (BOUNDARY NODES)
C
DO 105 I=1,NPAY
NNODES = NNODES-1
TEMP = TPAY(I)
WRITE(LCK2,2004) NNODES,TEMP,ONE
105 CONTINUE
C
C NODE FOR SPACE (BOUNDARY NODE)
C
NNODES = NNODES-1
TEMP = TSPACE
WRITE(LCK2,2004) NNODES,TEMP,ONE
104 WRITE(LCK2,2007)
NNODES = IABS(NNODES)
IF(TCARDS.EQ.0.) GO TO 115
READ(NIN,1005) NNSPCH,(T(I),I=1,NNSPCH)
GO TO 110
115 CONTINUE
N=0
IF(TSRF) 207,207,208
C
C CONSTANT INITIAL TEMPS
C
207 TS=ABS(TSURF)
DELT=0.0
GO TO 205
C
C DISTRIBUTE TEMPS BETWEEN TWALL AND TSURF
C
208 DELT=(TSURF-TWALL)/FLOAT(MAXNL)
TS=TWALL
205 DO210 I=1,NSEG
NLAY=NL(I)+1
N=N+1
T(N)=TS
NLAY=NL(I+1)
N=N+1
T(N)=T(N-1)+DELT
211 CONTINUE
210 CONTINUE
DO 212 I=1,N
212 T(I)=T(I)-TAB
110 T(N_NODES)=TSPACE -TABS
M=N_NODES-NPAY-1
DO 213 I=1,NPAY
J=M+I
T(J)=T(J)-TABS
110 T(I)=T(I)-TAB
WRITE(LUT5) (T(I),I=1,N_NODES)
C
C COMPUTE AREAS FOR LATERAL (HORIZ), AND NORMAL (VERT), CONDUCTORS
C PUT AREAS INTO ARRAYS AREAL, AND AREAN AND OUTPUT IN ARRAY DATA AS
C ARRAY 2 AND ARRAY 3 RESPECTIVELY.
C ARRAY ARCLN, LENGTHS OF NODES, IS ARRAY 4.
C
80 CONTINUE
ASQ=X*X
BSQ=Y*Y
ASQBSQ=ASQ*BSQ
ERAD(I)=Y
K=0
DO 20 I=1,NSEG
ANG1=BTAN(I)
ANG2=BTAN(I+1)
THETA(I)=THETAD(I)*0.01745329
M=THETAD(I)
DLAY=NL(I)
DO 81 L=1,M
K=K+1
SLAY(K)=DLAY
81 ANG2=ANG2*0.01745329
SINAL=SIGN(ANG2)
COSAL=COS(ANG2)
SSQ=SINAL**2
CSQ=COSAL**2
ERAD(I+1)=SQRT(ASQBSQ/(BSQ*SSQ+ASQ*CSQ))
DMLI=DPTH*FLOAT(NL(I))
R1=ERAD(I)+DMLI
R2=ERAD(I+1)+DMLI
\[ \text{RAD} = \frac{(R_1 + R_2)}{2} \]
\[ \text{ARCLN}(I) = \text{RAD} \times \text{THETA}(I) \]
\[ \text{SMALLR} = R_2 \times \text{SINAL} \]
\[ \text{ENDw} = \text{SMALLR} \times \text{PI04} \]
\[ \text{AREAL}(I) = \text{ENDw} \times \text{DPTH} \]

20 CONTINUE
CALL VIEWF

C GENERATE CONDUCTOR DATA, ALL CONDUCTANCE VALUES WILL BE COMPUTED
C IN VARIABLES 1 (THEREFORE 1.0 WILL BE USED IN THIS ROUTINE FOR C.)
C
C WRITE(LCK2,2005)
C NORMAL CONDUCTORS IN SHIELD
C
NODA=1
NCOND=1
DO 11 I=1,NSEG
NLAY=NL(I)
NODB=NODA+1
WRITE(LCK2,2006)NCOND,NLAY,M1,NODA,M1,NODB,M1,ONE,ONE,ONE
NODA=NODA+NLAY+1
11 NCOND=NCOND+NLAY

C LATERAL CONDUCTORS IN SHIELD
C
NODA=2
DO 15 I=2,NSEG
NLAY=NL(I)
NLM1P2=NL(I-1)+1
NODB=NODA+NLM1P2
WRITE(LCK2,2006)NCOND,NLAY,M1,NODA,M1,NODB,M1,ONE,ONE,ONE
NODA=NODB
15 NCOND=NCOND+NLAY

C CONDUCTORS FROM EDGE OF SHIELD TO TANKWALL
C
NODA=0
NODB=NNODES-NPAY-1
NCOND=NCOND-1

144 DO 14 I=1,NSEG
NODA=NODA+1
NCOND=NCOND+1
WRITE(LCK2,2026)NCOND,NODA,NODB,TEN20
14 NODA=NODA+NL(I)

C COMPUTE TABLE OF SURFACE NODE NUMBERS
C
NODA=0
DO 16 I=1,NSEG
NODA= NODA+NL(I)+1
NSNOD(I)= NODA
16 CONTINUE
J=NSEG+1
JJ=NNODES-NPAY-J
DO 162 I=J,NSURF
162 NSNOD(I)=JJ+I
WRITE(LCK2,2007)

C GENERATE CONSTANTS DATA
C
WRITE(LCK2,2008) (I,C(I),I=1,20)
WRITE(LCK2,2024) (I,C(I),I=21,28)
WRITE(LCK2,2010) (I,C(I),I=29,36)
WRITE(LCK2,2022) (CONNAM(I),CINCON(I),I=1,10)
WRITE(LCK2,2007)
C
C GENERATE ARRAY DATA BLOCK
C TO OUTPUT ARRAY CALL SUBROUTINE RITARY(N,A,NO,NF,ZN)
C WHERE
C N=ARRAY NUMBER
C A=ARRAY TO BE PUT OUT
C NO= NUMBER OF ELEMENTS IN THE ARRAY
C NF= EITHER 1 OR 2 NF=1 IF ARRAY IS FLOAT
C NF=2 IF ARRAY IS INTEG
C ZN=BCD ARRAY WHICH BECOMES A TITLE FOR
C ARRAY N.
C
WRITE(LCK2,2009)
CALL RITARY(1,NL,NSEG,2,Z1)
CALL RITARY(2,AREAL,NSEG,1,Z2)
CALL RITARY(3,A,NSURF,1,Z3)
CALL RITARY(4,ARCLN,NSEG,1,Z4)
CALL RITARY(5,THETAD,NSEG,1,Z5)
CALL RITARY(6,TEST,50,2,Z6)
CALL RITARY(7,AKVST,NTAB,1,Z7)
CALL RITARY(8,EMIST,NSURF,1,Z8)
CALL RITARY(9,AFA,NOFA,1,Z9)
CALL RITARY(10,NSNOD,NSURF,2,Z10)
WRITE(LCK2,2007)
C
C GENERATE EXECUTION BLOCK
C
WRITE(LCK2,2012)
WRITE(LCK2,2020)
WRITE(LCK2,2033) IEXECN
WRITE(LCK2,2007)
C
C GENERATE VARIABLES 1 BLOCK
C
WRITE(LCK2,2015)
WRITE(LCK2,2007)
C
C GENERATE VARIABLES 2 BLOCK
C
WRITE(LCK2,2014)
WRITE(LCK2,2031)
WRITE(LCK2,2007)
C
C GENERATE OUTPUT CALLS
C
WRITE(LCK2,2016)
WRITE(LCK2,2029)
WRITE(LCK2,2028)
WRITE(LCK2,2025)
WRITE(LCK2,2030)
WRITE(LCK2,2007)
C
C WRITE SUPPLIED SUBROUTINES ONTO TAPE LCK2
C
WRITE(LCK2,2017)
DO 60 I=1,NSTMTS
READ(LUT3) ISTM
60 WRITE(LCK2,2018) ISTM
WRITE(LCK2,2019)
REWIND 2
C
40
WRITE SWITCH SYSIN1 TO SYSUT6, TO READ CINDA DATA BY CINDA

END FILE LCK2
WRITE(LCK2,2021)
END FILE LCK2
END FILE LCK2
END FILE LCK2
REWIND LCK2
WRITE(NOUT,5000)

5000 FORMAT(51H END OF PREPARE PGM. NOW SWITCH TO INPUT FOR PREPRO)
WRITE(6,50001) NSTMNTS

5001 FORMAT(45H NO. OF STATEMENTS IN SUBROUTINES LOADED = I5)
CALL EXIT

READ(NIN,1000) ID
1000 FORMAT(13A6,A2)
1001 FORMAT(14,I3,7F10.5/6F10.5,A6/2I10,F10.5/18,F9.8,3)
1002 FORMAT(30I2)
1003 FORMAT(16,F9.5)
1004 FORMAT(8F 10.5)
1005 FORMAT(18,9F8.1/(10F8.1))
1000 FORMAT(20HSEXECUTE IBJOB, 64X /)
1 34H$ID COWNILL PREPRO, 50X/ 6H$IBJOB ,78X /
2 27HI$EDIT SYSCK1,SCHF2, 57X / 13H$IBLDR PREPRO ,71X /
3 13H$IBLDR TAPDEF, 71X/ 13H$IBLDR FVIO, 71X/
1001 FORMAT(14,2I3,7F10.5/6F10.5,A6/2I10,F10.5/18,F9.8,3)
2 13H$IBLDR GENLNK, 71X /
3 13H$IBLDR GENLNK, 71X /
4 27HI$EDIT ALPHA,SYSLB2, 57X / 13H$IBLDR GENLNK, 71X /
5 27HI$EDIT ALPHA,SYSLB2, 57X / 13H$IBLDR GENLNK, 71X /
6 27HI$EDIT ALPHA,SYSLB2, 57X /
7 13H$IBLDR PACK43 , 71X / 13H$IBLDR ORMIN , 71X /
8 13H$IBLDR CODERD , 71X / 13H$IBLDR DATARD , 71X /
9 13H$IBLDR CINDA4 , 71X / 13H$IBLDR SEARCH, 71X /
A 27HI$EDITOR ALPHA,SYSLB2, 57X / 13H$IBLDR CINDA4, 71X /
B 13H$IBLDR LOCATE, 71X / 13H$IBLDR FIBLDR, 71X /
C 13H$IBLDR READTP, 71X / 13H$IBLDR GENLNK, 71X /
D 27HI$EDITOR ALPHA,SYSLB2, 57X / 13H$IBLDR CINDA4, 71X /
E 27HI$EDITOR ALPHA,SYSLB2, 57X / 13H$IBLDR CINDA4, 71X /
F 27HI$EDITOR ALPHA,SYSLB2, 57X /
G 13H$IBLDR SKIP , 71X / 13H$IBLDR SPLIT , 71X /
H 5H$DATA , 79X / 80X 1
2001 FORMAT(7X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2003 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2004 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2005 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2006 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2007 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2008 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2009 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2010 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2011 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2012 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2013 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2014 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2015 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2016 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2017 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2018 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2019 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2020 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
2021 FORMAT(17X,5HBCD 9,9A6,14X/7X,5HBCD 5,5A6,38X/7X,3HEND,70X )
SUBROUTINE VIEWF
C RADIATION VIEW FACTOR BETWEEN FLAT SURFACED PAYLOAD AND SPHEROIDAL
C TANK WITH VARYING MLI THICKNESS.
C INPUT REQUIRED
C SLAY - NO. OF NODES IN EACH 1 DEGREE SEGMENT
C RP - PAYLOAD RADIUS
C X AND Y = MAJOR AND MINOR RADII OF TANK (WITHOUT COVERING)
C NSEG - NO. OF SEGMENTS ON TANK
C NPAY - NO. OF RINGS ON PAYLOAD
C NSURF- TOTAL NO. OF SURFACES IN RADIATION ENCLOSURE
C H - TANK TO PAYLOAD SPACING
COMMON /RADIO/ NSEG, NPAY, NSURF, X, Y, H, RP, THIKN, PI, SLAY, THETA, A, AFA
COMMON /TAPE/ NIN, NOUT, LCK2
DIMENSION SLAY(90), A(41), AFA(1681), THETA(30)
DIMENSION SAREA(90), DAREA(90), SEG(100), DF(600), F(600), R(10)
DIMENSION SAF(90, 100), AF(41, 41), RVF(41, 41)
DATA Wl, W2, W3, W4, W5, W6, EYl, EY2, /6H A, 6H R, 6H DAREA,
1 6H SEGA, 6H RFV, 6H AF, 6H(I), 6H(I, J) /
TOP= 2.*PI
ONEO= 100.
HALF D= ONEO/2.
RPSQ= RP*RP
N= 10.*RP
PSI= -HALFD
DO 28 K= 1, 90
PSI= PSI+ ONEO
28 DMLI= SLAY(K)*THIKN
AX= X+ DMLI
BY= Y+ DMLI
AXSQ= AX*AX
BXSQ = BX * BX
SPSI = SIN(PSI)
CPSI = COS(PSI)
SSQ = SPSI * SPSI
CSQ = CPSI * CPSI
AXBX = AX * BX
R1 = AXBX * SQRT(1. / (BXSQ * SSQ + AXSQ * CSQ))
Y1 = R1 * SPSI
Z1 = R1 * CPSI
DH = BX - Z1
XMsq = (AXSQ * Z1) / (BXSQ * Y1)
WSQ = (AXSQ - BXSQ) / BXSQ / BXSQ
W = SQRT(WSQ)
HOW = 0.5 / W
SRQB = SQRT(1. + WSQ * BXSQ)
RJ = BX / 2. * SRQB
S = HOW * ALOG(BX * W + SRQB)
UPLIM = RJ * S
PSIX = PSI + HAllF0
SPX = SIN(PSIX)
CPX = COS(PSIX)
SPXSQ = SPX * SPX
CPXPSQ = CPX * CPX
RLX = AXBX * SQRT(1. / (BXSQ * SPXSQ + AXSQ * CPXPSQ))
ZLX = R1X * CPX
SRQZX = SQRT(WSQ * ZLX ** 2 + 1.)
T = ZLX / 2. * SRQZX
U = HOW * ALOG(ZLX * W + SRQZX)
BOTLIM = T + U
SAREA(K) = TOPI * AX * (UPLIM - BOTLIM)
IF(K.EQ.1) GO TO 12
DAREA(K) = SAREA(K) - SAREA(K-1)
GO TO 14
12 DAREA(K) = SAREA(K)
14 DO 28 I = 1, N
RING = I
RHO = 1. * (RING - 5)
SixRNG = 6. * RING
SEGAI(I) = PI * (1.1 * RING ** 2 - (1.1 * (RING - 1.1) ** 2)) / SIXRNG
NSGT = 5 * I
GAMA = 0.
DO 24 J = 1, NSGT
IF(J.EQ.1) GO TO 16
GAMA = GAMA + (TOPI / SIXRNG)
X2 = RHO * COS(GAMA)
Y2 = RHO * SIN(GAMA)
XA = X2
YB = Y2 - Y1
ZC = H - DMLI + DH
RV = SQRT(XA**2 + YB**2 + ZC**2)
SM = SQRT(1. + XMI**2)
COSP1 = (YB + XMI * ZC) / (RV * SM)
COSP2 = ZC / RV
IF(COSP1.LE.0.) GO TO 18
DF(J) = (COSP1 * COSP2 * SEGAI(I)) / (PI * RV * RV)
GO TO 20
18 DF(J) = 3.
20 IF(J.EQ.1) GO TO 22
FIJ = DF(J) + F(J-1)
GO TO 24
22 FIJ = DF(J)
24 CONTINUE
SAF(K, I) = DAREA(K) * F(NSGT)
28 CONTINUE
   XNPAY=NPAY
   DO 30 II=1,NPAY
   XI=II
   RR=SQRT((RPSQ*XI)/XNPAY)
   NDELR=10.*RR
   XNDLRR=NDELR
   R(II)=1.*XNDLRR
   JJ=II+NSEG
   IF(JJ.EQ.1) GO TO 29
   A(JJ)=PI*(R(II)**2-R(II-1)**2)
   GO TO 30
29   A(JJ)=PI*R(II)**2
30 CONTINUE
   DO 304 KK=1,NSEG
   IF(KK.EQ.1) GO TO 301
   NDEG1=NDEG2+1
   NDEG=T+THETA(KK)
   NDEG2=NDEG1+NDEG-1
   GO TO 302
301 NDEG1=1
   NDEG2=T+THETA(KK)
302 ASUM=0.
   DO 303 K=NDEG1,NDEG2
   ASUM=ASUM+DAREA(K)
303 CONTINUE
   A(KK)=ASUM
304 CONTINUE
   A(NSURF)= 122.211
   DO 40 KK=1,NSEG
   IF(KK.EQ.1) GO TO 32
   NDEG1=NDEG2+1
   NDEG=T+THETA(KK)
   NDEG2=NDEG1+NDEG-1
   GO TO 34
32 NDEG1=1
   NDEG2=T+THETA(KK)
34 DO 40 II=1,NPAY
   IF(II.EQ.1) GO TO 36
   NDR1=NDR2+1
   NDR2=10.*R(II)
   GO TO 37
36 NDR1=1
   NDR2=10.*R(II)
37 SUM=0.
   DO 38 K=NDEG1,NDEG2
   DO 38 I=NDR1,NDR2
   SUM=SUM+SAF(K,I)
38 CONTINUE
   JJ=II+NSEG
   AF(KK,JJ)=SUM
40 CONTINUE
   DO 42 I=1,NSEG
   DO 42 J=1,NSEG
42 AF(I,J)=0.
   NS1=NSEG+1
   NS2=NSURF+1
   DO 44 I=NS1,NS2
   DO 44 J=1,NS2
44 AF(I,J)=0.
   AF(NSURF,NSURF)=0.
   DO 48 I=1,NSEG
   SUM=0
DO 46 K=NS1,NS2
46 SUM=SUM+AF(I,K)
48 AF(I,NSURF)=A(I)-SUM
DO 52 I=NS1,NS2
SUM=0
DO 50 K=1,NSEG
50 SUM=SUM+AF(K,I)
52 AF(I,NSURF)=A(I)-SUM
DO 60 KK=NS1,NSURF
60 RVF(KK,JJ)=AF(KK,JJ)/A(KK)
AFA(1)=0.0
AFA(2)=0.0
I=2
DO 100 KK=1,NSEG
DO 100 JJ=1,NSURF
I=I+1
100 AFA(I)=AF(KK,JJ)
Z1=W1
Z2=EY1
WRITE(NOUT,1001) Z1,Z2,(A(I),I=1,NSURF)
Z1=W2
WRITE(NOUT,1001) Z1,Z2,(R(I),I=1,NPAY)
Z1=W3
WRITE(NOUT,1001) Z1,Z2,(DAREA(I),I=1,90)
Z1=W4
WRITE(NOUT,1001) Z1,Z2,(SEGA(I),I=1,N)
Z1=W5
Z2=EY2
WRITE(NOUT,1001) Z1,Z2
DO 200 KK=1,NSEG
WRITE(NOUT,1002) (RVF(KK,JJ),JJ=NS1,NSURF)
200 CONTINUE
Z1=W6
WRITE(NOUT,1001) Z1,Z2
DO 1100 KK=1,NSURF
WRITE(NOUT,1002) (AF(KK,JJ),JJ=KK,NSURF)
1100 CONTINUE
RETURN
1001 FORMAT(1HK,2A6/(1X,1P10E13.5))
1002 FORMAT(1X,1P10E13.5)
END

SUBROUTINE RITARY(NA,A,NJ,HEAD)
DIMENSION A(1), IFMT1(5), IFMT2(7)
DIMENSION HEAD(9)
COMMON/BCD/NBCD(60)
COMMON /TAPE/ NIN,NOUT,LCK2
DATA IFMT1(1) /30H(12X,4H,), 4X) / 7
DATA IFMT2(1) /42H(12X, X) / 7
DATA IFLOT1, IFLOT2, IFLOT3
1 / 6H(F12.7, 6HFI2.7, 6H,1H.), / 9
DATA IFIX1, IFIX2, IBLANK
1 / 6H(I12, 6H I12, 6H / 11
DATA IE1, IE2
1 / 6H(F12.7, 6HFI2.7, 6H,1H.), / 12
$IBFTC RAYOUT
I 1/6H(E12.5, 6HE12.5, /
NSEG= N
JJ= J
IFMT2(3)= IBLANK
IFMT2(4)= IBLANK
ITEM2= IBLLOT3
GO TO (10,11,12), JJ
12 IFMT1(2)= IE1
IFMT1(4)= IE2
IFMT2(5)= IE2
ITEM1= IE1
GO TO 21
10 IFMT1(2)= IBLLOT1
IFMT1(4)= IBLLOT2
IFMT2(5)= IBLLOT2
ITEM1= IBLLOT1
GO TO 21
11 IFMT1(2)= IFIX1
IFMT1(4)= IFIX2
IFMT2(5)= IFIX2
ITEM1= IFIX1
21 N1= 1
N2= 5
WRITE(LCK2,2011) NA,(HEAD(I),I=1,9)
31 IF (N2 .GE. NSEG) GO TO 40
WRITE (LCK2,IFMT1) (A(I),I=N1,N2)
N1= N1+5
N2= N2+5
GO TO 31
40 N1= NSEG-N1
N2= NSEG
IF(NN .LE. 0) GO TO 39
IFMT2(3)= ITEM1
IFMT2(4)= ITEM2
IFMT2(2)= NBCD(NN)
NBL= 36-13*NN
42 IFMT2(6)= NBCD(NBL)
WRITE(LCK2,IFMT2) (A(I),I=N1,N2)
WRITE (LCK2,2013)
RETURN
39 NN=1
IFMT2(2)= NBCD(NN)
NBL=56
GO TO 42
2011 FORMAT(11X,I2,3H $,9A6,10X)
2013 FORMAT(12X,3HEND,65X)
END

$IBFTC COMDAT

BLOCK DATA
COMMOV /BCD/ NBCD(60)
DATA NBCD(1) /360H 1 2 3 4 5 6 7 8 3
1 9 10 11 12 13 14 15 16 17 18 19 4
2 20 21 22 23 24 25 26 27 28 29 30 5
3 31 32 33 34 35 36 37 38 39 40 41 6
4 42 43 44 45 46 47 48 49 50 51 52 7
5 53 54 55 56 57 58 59 60 /
END
SUBROUTINE GETG
COMMON /KONST/ NSEG, NPAY, NSURF, NNODES, MAXNL, NPRINT, NDAMP, NNL,
  NAREAL, NAREAN, NARCLN, NTHETA, NITEST, NHIC, NEMISS, NFA, NST,
  NEXTRA(3),
3 SIGMA, DPTH, DELS, EMISS, CON1, CON2, DELDAMP, X, TABS, HIKON, NPSWCH, RAD, H
COMMON /COND/ G(1)
COMMON /TEMP/ T(1)
COMMON /SOURCE/ Q(1)
COMMON /ARRAY/ A(1)
COMMON /FIXCON/ KA(1)
COMMON /QCOM/ QTS(30), QNORM(30)
DIMENSION AREA(41), EMISIV(41), TQ(41), NTQ(1), NARY(1)
EQUIVALENCE (A, NARY), (TQ, NTQ)
EQUIVALENCE (NLOOP, KA(5)), (DAMA, KA(9)), (LOOPCT, KA(20))
EQUIVALENCE (ARLXCC, KA(30))
DATA NTIME /1/
NPSWCH = 0
IF(LOOPCT .GE. NLOOP) CALL PCHT(1)
IF(NPSWCH .GT. 0) GO TO 9998
IF(ALRLXCC .LT. .01) DAMPA = DAMPA + DELDAMP
IF(DAMP .GT. 1.0) DAMPA = 1.0
IF(MOD(LOOPCT, 100) .NE. 0) GO TO 32
CALL TIMLFT(TLEFT)
TLEFT = TLEFT / 3600.
WRITE(6, 9999) TLEFT, LOOPCT
IF(TLEFT .LT. 14400.0) CALL PCHT(2)
IF(NPSWCH .GT. 0) GO TO 9998
32 CONTINUE
35 NSPI = NSURF + 1
CALL ENDOUT
WRITE(6, 1001) LOOPCT, DAMPA, (TQ(I), I = 1, NSPI)
33 CONTINUE
NRAD = RAD
C GET G FOR ALL NORMAL (VERTICAL) CONNECTORS
C
AKPART = DELS * SIGMA / ((2. / EMISS) - 1.)
NODA = 0
NCOND = 0
N1 = 1
DO 11 I = 1, NSEG
M = NNL + 1
NLAY = NARY(M)
MA = NAREAN + 1
ADD = A(MA) / DPTH / 8.
N2 = N1 + NLAY - 1
10 DO 12 J = N1, N2
NODA = NODA + 1
NODB = NODA + 1
T1 = T(NODA) + TABS
T2 = T(NODB) + TABS
TPART = (T1 + T2) *(T1 + T2) * T2
AK = AKPART * TPART
G(J) = ADD * AK
12 CONTINUE
NT2 = NODA
41 N3 = NODB - NLAY
N4 = N3 + 1
G(N1) = G(N1) * 2.
QNORM(I) = G(N1) * (T(N4) - T(N3))
CONTINUE
N1 = N2 + 1
TSURF = T(NT2)
NODA = NODA + 1

COMPUTE G FOR LATERAL CONNECTORS

NODA = 1
NCOND = N2
DO 15 I = 2, NSEG
M = NNL + I
NLAY = NARY(M)
NLM1P2 = NARY(M - 1) + 1
L = NARCLN + I
EL1 = A(L - 1)/2.
EL2 = A(L)/2.
LL = NAREAL + I - 1
AREA = A(LLL)
DO 13 J = 1, NLAY
NODA = NODA + 1
NODB = YODA + NLM1P2
NCOND = NCOND + 1
T1 = T(NODA) + TABS
T2 = T(NODB) + TABS
M = NITEST + J
ITEST = NARY(MM)
IF (ITEST) 20, 19, 19
20
M = NKHIIC
IF (T1 .GT. 240.) GO TO 18
CALL LAGRAN(T1, AK1, A(MM), 7)
17
IF (T2 .GT. 240.) GO TO 16
CALL LAGRAN(T2, AK2, A(MM), 7)
GO TO 21
18
AK1 = HIKCON
GO TO 17
16
AK2 = HIKCON
GO TO 21
19
AK1 = CDN1 * T1 + CON2
AK2 = CDN1 * T2 + CON2
21
DEN = EL1/AK1 + EL2/AK2
G(NCOND) = AREA/DEN
13 CONTINUE
15 NODA = NODA + 1

COMPUTE SOURCE TERMS FOR NET Q, RADIOSITY, USING SUBR. IRRADI FOR INTERCHANGE BETWEEN FACE OF SHIELD AND PAYLOAD

GO TO (600, 601), NTIME
600
NTIME = 2
J = NSURF + 1
M = NAREAN - 1
K = NMISS - 1
DO 602 I = 1, J
MM = M + I
KK = K + I
SAREA(I) = A(MM)
EMISIV(I) = A(KK)
602 CONTINUE
NARY(NAFA + 1) = NSURF
A(NAFA + 2) = SIGMA
WRITE(5, 1000) (SAREA(I), I = 1, J)
WRITE(5, 1000) (EMISIV(I), I = 1, J)
601 CONTINUE
DO 500 I=1,NSURF
K=NSNT+I
J=NARY(K)
500 TQ(I+1)=T(J)
NTQ(I)=NSURF
CALL IRRADI(SAREA,EMISIV,A(NAFA),TQ)
DO 505 I=1,NSEG
K=NSNT+I
J=NARY(K)
Q(T(I))=Q(J)+TQ(I)/8.
Q(T(I))=Q(T(I))
505 CONTINUE
9998 RETURN
1000 FORMAT(1X,10G13.5)
1001 FORMAT(10HL LOOPCT= I6, 5X,6HDAMPA= F6.4, 1 /22H NET Q FROM RADIOSITY/(1X,10G13.5))
9999 FORMAT(12H TIME LEFT= F15.2, 10H LOOPCT= I8)
END
$IBFTC OUTEND
SUBROUTINE OUTEND
COMMON/TITLE/ HEAD(20)
COMMON /KONST/ K(1)
COMMON/FIXCON/ X(27),L,NP
COMMON/QCOM/ QTS(30),QNORM(30)
DIMENSION QTS8(30),QN8(30)
NSEG=K(1)
STS=0.
SN0=0.
CALL PICTR
DO 10 I=1,NSEG
QTS8(I)= QTS(I)*8.
QN8(I) = QNORM(I)*8.
STS=STS+QTS(I)
10 SN0=SN0+QNORM(I)
STS8=STS*8.
SN08=SN0*8.
WRITE(6,2000) (HEAD(I),I=1,20),(I,QTS(I),QNORM(I),
1 QTS8(I),QN8(I),I=1,NSEG)
WRITE(6,2001) STS,SN0,STS8,SN08
N=K(31)
L=0
NP=0
IF(N .EQ. 0) GO TO 4
GO TO (1,2),N
1 WRITE(6,2003)
GO TO 3
2 WRITE(6,2002)
3 WRITE(6,2004)
4 RETURN
2000 FORMAT(1H1,10X,20A6/9HL ND. SEG,8X,3HQT8,11X,5HQNORM,
1 24X,5HQT8*8,9X,7HQNORM*8 //
2 (18,4X,2F14.8,15X,2F14.8))
2001 FORMAT(12HLSUMMATION ,2G14.5,15X,2G14.5)
2002 FORMAT( 7HLNOTE.... THIS CASE STOPPED FOR TIME EXCEEDED, IT HAS NO
IT YET CONVERGED. )
2003 FORMAT(52HL..... THIS CASE STOPPED FOR ITERATION COUNT EXCEEDED. )
2004 FORMAT(53HK .... CARDS WERE PUNCHED TO RESTART FROM THIS POINT. )
END
$IBFTC DISTRT
SUBROUTINE TDIST
COMMON /KONST/ K(1)
COMMON /TEMP/ T(1)
LUT5=11
REWIND LUT5
NNODES=K(4)
12 READ (LUT5) (T(I),I=1,NNODES)
RETURN
END

$IBFTC GRAFIC
SUBROUTINE PICTR
COMMON /TITLE/ HEAD(20)
COMMON /TEMP/ T(1)
COMMON /ARRAY/ IA(1)
COMMON /KONST/ NSEG, NPAY, NSURF, NNODES, MAXNL, NPRINT, NDAMP, NNL,
1 NAREAL, NAREAN, NARCH, NTHETA, NITEST, NKHIC, NEMISS, NAFA, NSNT,
2 NEXTRA(3),
3 SIGMA, DPTH, DELS, EMISS, CON1, CON2, DEDM, X, TABS, HIKCON, PSWITCH, RAD, H
DIMENSION NODNO(15), TOUT(15), LINE(30), NBCD(50), IFMT(13)
DIMENSION NRENDN(15), NRENDT(15), IFMT(96), A(1)
EQUIVALENCE (IA, A)
DATA ILINE, IBLANK /6H**+***, 6H /
DATA IFMT(1) / 78H(2HP  8XvAl,30A4/2HP  A6~A2eAlr  (16*1X*A1)/2
1HP AbrA2rAlr  (F7.2rAl) /2
DATA XFMT1~11/576H(1Hlv5X.20A6/8X10(lH*~~(8H+*tt+**,2H+~1OX,1711H*)/8X
2918H*PAYLOAD NODES
9X
31H+  /BX,lOH+ NODE NOS, 18  r2H  *rlOX~lOH' NODE NO-*I5*2H */
48XvlH*+9X1
(AN 1
2H +e
10X. 1H* , 15x1  1H+/8X,  10H*
TEMPS=
5
F8.2 ,2H *10X,7H TEMP=,F8.2,2H *8X,10H+ EMISIV=,
6    F8.4,2H *10X,8H* EMISIV,F7.4,2H */8X,10H(1H*),
7****,2H**10X,17(1H*)/14HK ANGLE OF EACH/14H SEGMENT(DEG.)/10X,
8    F8.1/14HKEMISIVITY OF /14H EACH SEGMENT /10X,
1 / 6HSURFAC ,6HE
DO 5 I=1,NNODES
50
5 \[ T(I) = T(I) + T_{ABS} \]
\[ \text{MAXL} = \text{MAXNL} + 1 \]
\[ \text{NSPACE} = \text{NNODES} \]
\[ \text{NWALLZ} = \text{NSPACE} - \text{NPAY} - 1 \]
\[ \text{NCINIT} = 0 \]
\[ \text{NS1} = 1 \]
\[ \text{NS2} = \text{NSEG} \]
\[ \text{IF}(\text{NSEG} \geq 15) \text{ NS2} = 15 \]

8 \[ \text{NTH1} = \text{NTHETA} + \text{NS1} \]
\[ \text{NTH2} = \text{NTHETA} + \text{NS2} \]
\[ \text{NP1} = \text{NSPACE} - \text{NPAY} \]
\[ \text{NP2} = \text{NSPACE} - 1 \]
\[ \text{NET1} = \text{NEMISS} + \text{NS1} \]
\[ \text{NET2} = \text{NEMISS} + \text{NS2} \]
\[ \text{NEP1} = \text{NEP1} + \text{NSEG} \]
\[ \text{NEP2} = \text{NEP1} + \text{NPAY} - 1 \]
\[ \text{NEPSPACE} = \text{NEMISS} + \text{NSEG} + \text{NPAY} + 1 \]
\[ \text{IFMT1}(5) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(13) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(24) = \text{NBCD}(\text{NPAY} - 1) \]
\[ \text{IFMT1}(35) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(44) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(54) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(64) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(86) = \text{NBCD}(\text{NSEG}) \]
\[ \text{IFMT1}(95) = \text{NBCD}(\text{NSEG}) \]
\[ \text{WRITE}(6, 1004) \text{ HEA}(\text{J}, \text{J} = \text{NP1}, \text{NP2}), \text{NSPACE}, (T(I), I = \text{NP1}, \text{NP2}), T(\text{NSPACE}) \]

1 \[ \text{IFMT1}(5) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(13) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(24) = \text{NBCD}(\text{NPAY} - 1) \]
\[ \text{IFMT1}(35) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(44) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(54) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(64) = \text{NBCD}(\text{NPAY}) \]
\[ \text{IFMT1}(86) = \text{NBCD}(\text{NSEG}) \]
\[ \text{IFMT1}(95) = \text{NBCD}(\text{NSEG}) \]
\[ \text{WRITE}(6, 1004) \text{ HEA}(\text{J}, \text{J} = \text{NP1}, \text{NP2}), \text{NSPACE}, (T(I), I = \text{NP1}, \text{NP2}), T(\text{NSPACE}) \]

1 \[ \text{WRITE}(6, 2003) \]
\[ \text{DO} 10 I = 1, \text{MAXL} \]
\[ M = \text{MAXL} - I + 1 \]
\[ \text{IM} = M - 1 \]

9 \[ \text{DO} 11 \text{L} = 1, 15 \]
\[ \text{LINE}(\text{L}) = \text{IBLANK} \]
\[ \text{LINE}(\text{L} + 15) = \text{IBLANK} \]
\[ \text{NODNO}(\text{L}) = 0 \]

11 \[ \text{TOUT}(\text{L}) = 0.0 \]
\[ \text{NS1P1} = \text{NS1} - 1 \]
\[ \text{NC} = \text{NCINIT} \]
\[ \text{N} = 0 \]
\[ \text{ISNSW} = 1 \]

50 \[ \text{IF}(M \cdot \text{EQ.} 1) \text{ GO TO 12} \]
\[ \text{MM1} = M - 1 \]

52 \[ \text{MM} = \text{NITEST} + \text{MM1} \]
\[ \text{ITEST} = \text{IA}(\text{MM}) \]
\[ \text{IF}(\text{ITEST} \cdot \text{LT.} 0) \text{ ISNSW} = -1 \]

12 \[ \text{DO} 20 \text{J} = \text{NS1}, \text{NS2} \]
\[ \text{N2} = 2 \cdot (\text{J} - \text{NS1P1}) \]
\[ \text{NI} = \text{N2} - 1 \]
\[ \text{JJ} = \text{NNL} + \text{J} \]
\[ \text{JN} = \text{J} - \text{NS1} + 1 \]
\[ \text{NLAY} = \text{IA}(\text{JJ}) + 1 \]
\[ \text{IF}(\text{NLAY} \cdot \text{LT.} \text{ M}) \text{ GO TO 19} \]
\[ \text{N} = \text{N} + 1 \]

C

C \text{CHECK FOR TOP SURFACE TARGETING} 
C

26 \[ \text{IF}(\text{NLAY} \cdot \text{GT.} \text{ M}) \text{ GO TO 21} \]
C
C NLAY =M THIS IS A TOP SURFACE, CHECK TO SEE IF TARGETED
C
24 JJ=NEMISS+J
   EMISIV=A(JJ)
   IF(EMISIV .LT. 0.1) GO TO 21
   LINE(N1)=ITARG
   LINE(N2)=ITARG
   GO TO 27
C
C NL(JJ) = M THIS IS AN INTERIOR NODE OR TANKWALL SURFACE.
C
21 LINE(N1)= ILINE
   LINE(N2)= ILINE
27 ISIGN=1
   IF(I .LT. MAXL) GO TO 23
   NREND(N)=IBLANK
   NRENDT(N)=IBLANK
   LEND=IBLANK
   LLEN=ILINE
   NAM1=ITW1
   NAM2=ITW2
   NAM3=ISU1
   NAM4=ISU2
   GO TO 18
23 ISIGN=ISNSW
   LEND=ILINE
   LLEN=ILINE
   NAM1=NBCD(IN)
   NAM2=IBLANK
   NAM3=IBLANK
   NAM4=IBLANK
   NREND(N)=ILINE
   NRENDT(N)=ILINE
18 NN= NC+M
   NODNO(JJ)= NN*ISIGN
   TOUT(JY)=T(NN)
19 NC= NC+NLAY
20 CONTINUE
   IF(N .GT. 0) GO TO 22
   WRITE(6,2001)
   GO TO 10
22 IFMT( 6)= NBCD(N)
   IFMT(11)= NBCD(N)
   WRITE(6,IFMT)LLEN, ( LINE(L),L=1,30), NAM1,NAM2,LEND,
   1 ( NODNO(L),NREND(L),L=1,N), NAM3,NAM4,LEND,
   2 ( TOUT(L),NRENDT(L),L=1,N)
10 CONTINUE
   DO 31 I=1,3
31 LINE(I)=ILINE
   WRITE(6,2002) ( LINE(I),I=1,3), NWALL, T(NWALL), ( LINE(I),I=1,3)
   IF(NS2 .GE. NSEG) GO TO 40
   NS1= NS2+1
   NS2= NSEG
   NCINIT=NC
   GO TO 8
40 DO 41 I=1,NNODES
41 T(I)=T(I)-TABS
RETURN
2000 FORMAT(1H1, 5X,20A6/41X,17(1H*),10X,17(1H*)/)
   1 41X,1H*,15X,1H*,10X,1H*,15X,1H* / 
   2 41X, 9H* PAYLOAD,7X,1H*,10X,7H* SPACE,9X,1H* / 
   3 41X,10H* NODE NO.*,15,2H *,10X,10H* NODE NO.*,15,2H */
SUBROUTINE PCTE(N)
COMMON /TEMP/ T(I)
COMMON /FACON/ FC(I)
COMMON/KONST/ K(I)
C SET NP SWICH TO NON-ZERO TO INDICATE CARDS BEING PUNCHED
K(31)=N
NNODES=K(4)
FC(19)=10000.
WRITE(6,101)
PUNCH 100, NNODES, (T(I),I=1,NNODES)
100 FORMAT(18,9F8.1/(10F8.1))
101 FORMAT(36H1 TEMPERATURE CARDS BEING PUNCHED)
RETURN
END

$1BFTC PUNTEH

SUBROUTINE PCTE(N)
COMMON /TEMP/ T(I)
COMMON /FACON/ FC(I)
COMMON/KONST/ K(I)
C SET NP SWICH TO NON-ZERO TO INDICATE CARDS BEING PUNCHED
K(31)=N
NNODES=K(4)
FC(19)=10000.
WRITE(6,101)
PUNCH 100, NNODES, (T(I),I=1,NNODES)
100 FORMAT(18,9F8.1/(10F8.1))
101 FORMAT(36H1 TEMPERATURE CARDS BEING PUNCHED)
RETURN
END

$1BFTC PUNTEH

SUBROUTINE PCTE(N)
COMMON /TEMP/ T(I)
COMMON /FACON/ FC(I)
COMMON/KONST/ K(I)
C SET NP SWICH TO NON-ZERO TO INDICATE CARDS BEING PUNCHED
K(31)=N
NNODES=K(4)
FC(19)=10000.
WRITE(6,101)
PUNCH 100, NNODES, (T(I),I=1,NNODES)
100 FORMAT(18,9F8.1/(10F8.1))
101 FORMAT(36H1 TEMPERATURE CARDS BEING PUNCHED)
RETURN
END
APPENDIX H

SYMBOLS USED IN ANALYSIS

\( A \) area, \( \text{ft}^2; \text{m}^2 \)
\( \Delta A \) incremental area, \( \text{ft}^2; \text{m}^2 \)
\( A_x \) incremental area in \( x \) direction in finite difference equation, \( \text{ft}^2; \text{m}^2 \)
\( A_y \) incremental area in \( y \) direction in finite difference equation, \( \text{ft}^2; \text{m}^2 \)
\( A_z \) incremental area in \( z \) direction in finite difference equation, \( \text{ft}^2; \text{m}^2 \)
\( a \) major semi-axis of tank plus shield, \( \text{ft}; \text{m} \)
\( A_i F_{i,j} \) product of \( i^{th} \) surface area times geometric radiation view factor between \( i^{th} \) and \( j^{th} \) surface, \( \text{ft}^2; \text{m}^2 \)
\( b \) minor semi-axis of tank plus shield, \( \text{ft}; \text{m} \)
\( C_p \) specific heat, \( \text{Btu}/(\text{lb})(^\circ\text{F}); \text{J}/(\text{kg})(\text{K}) \)
\( F_{i,j} \) geometric radiation view factor - portion of total energy, either emitted or reflected diffusely from surface \( i \), that is intercepted by surface \( j \)
\( G \) conductor value for heat transfer, \( \text{Btu}/(\text{lb})(^\circ\text{F}); \text{J}/(\text{sec})(\text{K}) \)
\( h \) tank-to-payload spacing, \( \text{ft}; \text{m} \)
\( k \) thermal conductivity, \( \text{Btu}/(\text{hr})(\text{ft})(^\circ\text{F}); \text{J}/(\text{sec})(\text{m})(\text{K}) \)
\( \Delta L \) payload ring width, \( \text{ft}; \text{m} \)
\( m_{N_1} \) slope of normal \( N_1 \)
\( N \) normal to center of an incremental surface
\( N_E \) number of surfaces in radiation enclosure
\( N_J \) number of incremental areas in \( j^{th} \) payload ring
\( Q \) heat flow, \( \text{Btu}/\text{hr}; \text{J}/\text{sec} \)
\( q \) incremental heat flows, either across a plane of any segment or into (or out of) any surface segment, \( \text{Btu}/\text{hr}; \text{J}/\text{sec} \)
\( \dot{q} \) internal heat generation rate at each node, \( \text{Btu}/\text{hr}; \text{J}/\text{sec} \)
\( R \) distance between any two incremental surfaces, \( \text{ft}; \text{m} \)
\( R_p \) payload radius, \( \text{ft}; \text{m} \)
\( R_1 \) distance from tank center to incremental surface, \( \text{ft}; \text{m} \)
\( R_2 \) distance from payload center to incremental surface, ft; m
\( S \) source term, \( \dot{q} /k \), \((\text{ft}^3)(0^\circ \text{R})/\text{hr}; (\text{m}^3)(\text{K})/\text{sec}\)
\( \Delta S \) thickness of MLI, ft; m
\( T \) temperature, \( ^\circ \text{R} ; \text{K} \)
\( t \) time, sec
\( V \) volume, \( \text{ft}^3; \text{m}^3 \)
\( x, y, z \) coordinate system, point coordinates
\( x_r \) major semi-axis of tank, ft; m
\( \Delta x \) incremental distance in x direction
\( y_r \) minor semi-axis of tank, ft; m
\( \Delta y \) incremental distance in y direction, ft; m
\( \Delta z \) incremental distance in z direction, ft; m
\( \alpha \) thermal diffusivity, \( \text{ft}^2/\text{hr}; \text{m}^2/\text{sec} \)
\( \beta \) angle measured from minor axis to segment boundary on tank
\( \gamma \) location angle on payload
\( \epsilon \) emissivity
\( \theta \) angle on tank to specify segments
\( \rho \) density, \( \text{lb}/\text{ft}^3; \text{kg}/\text{m}^3 \)
\( \varphi \) angle between the normal to a surface and the vector \( R \) from that surface
\( \psi_m \) angle from top of tank to \( R_1 \)

Subscripts:
\( i \) index or counter
\( j \) index or counter
\( k \) index or counter
\( \text{lat} \) lateral
\( m, n \) reference or index
\( \text{norm} \) normal
\( \text{p} \) refers to payload surface increment
\( \text{t} \) refers to tank surface increment
Notation:

→ vector

^ unit vector

\( \nabla^2 \) Laplacian operator, \( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \)

\( \hat{i}, \hat{j}, \hat{k} \) unit vectors in x, y, z directions
REFERENCES


Solar radiation

Probable high-thermal-conductivity shield positions

Probable high-emissivity surface

Varying multilayer insulation (MLI) thickness

Adiabatic surface (typical)

Open support structure

Liquid oxygen tanks 163° R (91 K)

Figure 1. - Vehicle configuration.

Figure 2. - Cross-sectional view of model.
Figure 3. - 45° segment of the multi-layer insulation (MLI) divided into nodes.

Figure 4. - Cross-sectional view of several nodes.
"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—National Aeronautics and Space Act of 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546