Radiation heat transfer analysis of a Czochralski furnace with a radiation shield

Frederick A. Merz

Follow this and additional works at: https://scholarworks.rit.edu/theses

Recommended Citation

This Thesis is brought to you for free and open access by RIT Scholar Works. It has been accepted for inclusion in Theses by an authorized administrator of RIT Scholar Works. For more information, please contact ritscholarworks@rit.edu.
RADIATION HEAT TRANSFER ANALYSIS
OF A CZOCHRALSKI FURNACE
WITH A RADIATION SHIELD

by
Frederick A. Merz

A Thesis Submitted
in
Partial Fulfillment
of the
Requirements for the Degree of
MASTER OF SCIENCE
in
Mechanical Engineering

Approved by:
Prof. Dr. Bhalchandra V. Karlekar
(Thesis Advisor)

Prof. Robert A. Ellson

Prof. Signature not legible

Prof. Dr. Bhalchandra V. Karlekar
(Department Head)

DEPARTMENT OF MECHANICAL ENGINEERING
COLLEGE OF ENGINEERING
ROCHESTER INSTITUTE OF TECHNOLOGY
ROCHESTER, NEW YORK
OCTOBER 1983
ACKNOWLEDGEMENTS

It is a pleasure for the author to express his sincere gratitude to Dr. Bhalchandra V. Karlekar for his able supervision and patient encouragement throughout this work.

The author is also thankful to Dr. W. E. Langlois for his useful help and suggestions. Appreciation also goes to Peggie Galbraith and Rose Zollo for their devotion in typing this paper.
TABLE OF CONTENTS

Abstract

Introduction

Nomenclature

The Crystal Growth in a Czochralski Furnace 1

History of Various Works 3

Statement of the Problem 5

Table 1 6a

Mathematical Formulation 7

Method of Solution 16

Svanberg Vorticity Parameter 18

Rotational Equation 24

Energy Equation 24

Radiant Energy Balance at the Melt Surface 27

Melt Surface Equilibrium 27

Radiation Network Equation 29

View Factors 34

Computer Program and Stability 40

Fluid, Temperature, and Radiation Profiles 47

Results 51

Discussions and Conclusions 54

References 55

Appendix 56

Naiver-Stokes Equation in Polar Coordinates

Rotational Velocity in Production

Svanberg Vorticity Equation

View Factors

Computer Program

Computer Results
### TABLE OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Layout of Czochralski Furnace</td>
<td>6</td>
</tr>
<tr>
<td>2 a</td>
<td>Axial Grid Cell</td>
<td>8</td>
</tr>
<tr>
<td>b</td>
<td>Off Axial Grid Cell</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>r+ and z- Grid Locations</td>
<td>21</td>
</tr>
<tr>
<td>4</td>
<td>Network of System without Shield</td>
<td>31</td>
</tr>
<tr>
<td>5</td>
<td>Network of System with Shield</td>
<td>32</td>
</tr>
<tr>
<td>6 a</td>
<td>Surface-to-shield Viewfactor</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>Shield Point Locations</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>Surface-to-shield Viewfactor</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>Furnace Views</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Instability Grid Locations</td>
<td>42</td>
</tr>
<tr>
<td>8</td>
<td>Instability Grid Values</td>
<td>43</td>
</tr>
<tr>
<td>9</td>
<td>Flow Streamlines</td>
<td>45</td>
</tr>
<tr>
<td>10</td>
<td>Temperature Isotherms (with Shield)</td>
<td>49</td>
</tr>
<tr>
<td>11</td>
<td>Surface Temperatures</td>
<td>50</td>
</tr>
</tbody>
</table>
This paper reports on the radiation heat transfer from the melt in a Czochralski crystal growing furnace in the presence of a radiation shield. The change of radiation heat transfer effects three basic areas, the cooling rate of the crystal, the bulk flow and the bulk temperature of the melt. The shield is installed above the melt to inhibit the heat transfer from the melt to the cooling crystal, allowing the crystal to grow 40% faster. Radiation is the primary mode of heat transfer in the furnace where the temperature to grow silicon crystals is around 1773°K.

The governing equations are the three-dimensional Navier Stokes equations in cylindrical coordinates for laminar flow, with the buoyancy term present, and the energy equation. The boundary conditions at the free surface of the melt are nonlinear due to the presence of the radiation. Instead of introducing the stream function to facilitate the solution, the Svanberg vorticity parameter, \( \omega/r \), was introduced. The resulting equations were solved by an iterative process to yield the velocity distribution in the melt. Then these equations were used to solve the unsteady state energy equation. The radiation viewfactors were calculated by using a program called CONFACII. The temperature values thus obtained were used for the buoyancy terms in the Navier Stokes equations.
INTRODUCTION

In the manufacturing of solid state electronic devices and in solar cells, silicon wafers are primarily used as the base material. The wafers are produced by slicing a solid silicon crystal grown in a Czochralski furnace. The crystals are grown in a low pressure, (20 torr) high temperature (1773°K) environment, so radiation is the primary method of heat transfer in the furnace. The objective of this work is to show the effect of introducing a radiation shield to enhance crystal growth. This paper will begin with a short description of the crystal growing process.
NOMENCLATURE

$C_v$ specific heat (J/kg·°K)

$g$ gravitational acceleration (cm/sec²)

$H$ melt depth (cm)

$p$ pressure (N/m²)

$r$ radial coordinate (cm)

$R_{cru}$ crucible radius (cm)

$R_{crys}$ crystal radius (cm)

$R_{i-w}$ spacial thermal resistance (1/cm²)

$R_i$ surface thermal resistance (1/cm²)

$T$ absolute temperature (°K)

$T_{cru}$ absolute temperature of the crucible (°K)

$T_{crys}$ absolute temperature of crystal (°K)

$u$ outward radial velocity (cm/sec)

$v$ azimuthal velocity (cm/sec)

$w$ upward axial velocity (cm/sec)

$z$ axial coordinate, measured from the crucible bottom (cm)

$S$ Svanberg vorticity 1/(cm·sec)

$\alpha$ volumetric expansion coefficient (1/°K)

$\epsilon$ emissivity

$\nu$ kinematic viscosity (m²/sec)

$\rho$ density (gm/cm³)

$\sigma$ Stefan-Boltzmann constant W/(°K⁴·m²)

$\psi$ Stokes streamfunction (cm³/sec)

$\omega$ vorticity, $\omega = \frac{\partial w}{\partial r} - \frac{\partial u}{\partial z}$ (1/sec)

$\Omega$ swirl, $r v$ (cm²/sec)

$\Omega_{cru}$ crucible rotation rate (cm²/sec)

$\Omega_{crys}$ crystal rotation rate (cm²/sec)
THE CRYSTAL GROWTH IN A CZOCHRALSKI FURNACE

This study was done to determine the effects of a radiation shield on the temperature and velocity profiles of the melt in a crucible and the effect it has on the radiation heat transfer in the furnace. This radiation shield is located above the molten surface in a furnace used to grow silicon crystals for integrated circuits or solar cells. The crystal growing process starts by melting pure silicon in a pure quartz crucible in an oxygen free, low pressure surroundings. Then, a perfect silicon crystal (either 1:0:0 or 1:1:1 orientation), called a seed, is lowered into the melt. The temperature of the melt is then lowered until the solid seed maintains equilibrium. The crucible and seed are continuously rotated in opposite directions, and the seed is then raised at a speed of 2.5 to 10 cm per hour for a 7.5 cm diameter crystal. The seed and crystal diameter is controlled by the temperature and pull speed; as the temperature is lowered or the pull speed is reduced the crystal will grow larger and conversely, the opposite conditions will reduce the diameter.

The crystal growing process consists of four steps, 1) melt the solid polysilicon, 2) grow the seed, 3) crystal growth, and 4) cool down to prevent cracking. The time required to produce a 5 kg crystal is about 12.5 hours. The breakdown of this time period into the four steps is 1.5 hours for melt down, 1 hour for seed growth, 9 hours for crystal growth and 1 hour for the crystal to cool.
The crystal growing time for a 10cm diameter crystal being pulled at 9 cm/hour is 72% of the total time. The goal of any crystal growing process is to produce as much material per unit time as possible. To achieve this goal, the growing time should be reduced. To reduce this time, the growth rate has to be increased.

The growth rate of the crystal is determined by the rate at which the silicon solidifies. The heat of fusion given off during this solidification is transported through the crystal and into the supercooled melt just under the crystal surface. To increase the pull speed, this heat has to be dissipated faster. This can be done in two ways, 1) the melt temperature can be lowered or, 2) the crystal surrounding temperature can be reduced. The method of reducing the melt temperature is to lower the heater output. This cools the crucible wall and then the melt. There is a point when the melt gets so cold that it will start to freeze at nucleation sites on the crucible wall. This is referred to as wall freeze. The wall freeze then starts to grow towards the center and if left to grow long enough, it will actually reach the crystal. To save the crystal, it has to be pulled out, thus producing smaller crystals.

The other method for increasing the pull speed is to allow the crystal to lose more heat to the surroundings. In a typical system, the surface of the melt radiates heat to the crystal thus reducing the rate of cooling. When a radiation shield is
introduced between the melt surface and the crystal, the heat transfer is reduced allowing the crystal to cool faster.

The goal of this thesis is to show that the melt surface will lose heat at a slower rate and the crystal will lose more heat by radiation with the installation of a radiation shield. This will then allow higher pull speeds and better yields of the crystallized material. The shield also results in another beneficial by-product, lower energy consumption. This is achieved by insulating the melt surface and thus requiring lower heater temperature to maintain the same melt temperature.

HISTORY OF VARIOUS WORKS

The first work of numerical solutions to flow and temperature profiles in a Czochralski furnace was done by Kobayski and Arizumi in 1975[1]. During the late 1970's, Langlois,[2,3,4,5] in his four papers also covered bulk flow parameter of melt in the crucible. These papers present several improvements. The vertical grid size in the numerical scheme was reduced at the top and bottom to show more detail and to reveal the reverse flow pattern in the melt just under the crystal. A direct method to solve the Naiver-Stokes equation by matrix manipulation was presented to reduce the computing time over the iterative method. The last improvement was the introduction of Svanberg vorticity parameter. This paper considers the radiation heat loss from the surface of the melt, instead of free radiation assumed in previous papers. Another
factor that was not considered in any paper so far is the influence of the heat of fusion from the crystal in the bulk flow patterns. This thesis follows the same practice.
STATEMENT OF THE PROBLEM

The crystal growing system consists of six elements, crystal, crucible, melt, furnace wall, top cover and shield as shown in Figure 1. The furnace (walls and top cover) is typically water-cooled and is sealed to prevent air leakage. An oxygen-free environment is provided by continuously purging argon at low pressure (20 torr). This reduces the build up of silicon monoxide that can cause problems as the crystal grows. The crucible is made from ultra pure fused quartz. The crystal is started by a seed of either 1:0:0 or 1:1:1 orientation. The melt is pure polysilicon doped with boron to achieve the proper resistivity for applications in integrated circuits or solar cells. The radiation shield, as far as the author knows, has only been used on an experimental basis. The material for the shield used was pure graphite that was pre-baked under vacuum to remove contaminations. The physical parameters used are listed in Table 1.

We wish to determine the effects of a radiation shield on the cooling rate of the crystal and the temperature of the molten silicon. This is done by comparing the heat loss from the crystal, with and without the radiation shield.
FIGURE 1 - LAYOUT OF CZOCHRALSKI FURNACE
TABLE 1

Melt density: 2.33 gm/cm$^3$
Specific heat: 9.75 $\times 10^4$ J/kg$^{-\circ}$K
Kinematic viscosity: 233 M$^2$/sec.
Thermal diffusivity: 1.4 cm$^2$/sec.
Volumetric expansion coeff: 1.41 $\times 10^{-6}$/°K

Emissivities:

Surface: 0.318  
Shield: 0.15  
Crystal: 0.2  
Wall: 0.40  
Furnace Top Place: 0.40

Shield angle: 30°
Crucible radius: 11.55 cm
Crystal radius: 3.75 cm
Crystal height: 20 cm
Melt depth: 10.91 cm
Crucible rotation rate: 1.47 rad/sec.
Crystal rotation rate: -2.31 rad/sec. (counter-rotation)
Crucible temperature: 1773.0 K
Crystal temperature: 1685.0 K
Wall temperature: 900°K
Furnace top plate temperature: 450°K
The equations used to solve this problem involved first casting them in differential form and then, in finite difference form. The finite difference form will consist of an iterative solution over a grid pattern containing two types of cells, axis and off axis elements, fig 2a, b. The four governing equations describing this problem are; 1) continuity equation, 2) three dimensional Navier Stokes equation, 3) conservation of energy equation with the free melt surface energy boundary condition described by radiation network equation, 4) Svanberg Vorticity equation. The following assumptions were made: (i) a steady state system; (ii) no-slip velocity condition; (iii) specified temperatures on all solid surfaces except the radiation shield; (iv) no heat generation due to the heat of fusion or by fluid friction and (v) temperature independent physical properties.

In the equations that follow, the independent variables are: T the temperature, u, w, and v respectively, the radial, axial, and rotational fluid velocities, p the pressure, ω vorticity parameter, and Ψ the Stream function. The constant parameters used are, \( k \) the thermal diffusivity, \( \rho \) the density, \( \nu \) the kinematic viscosity, g the gravitational acceleration and \( \alpha \) the volume expansion coefficient, and \( T_0 \) the average melt temperature.
FIGURE 2a – AXIAL GRID CELL

FIGURE 2b – OFF-AXIS GRID CELL
The seven dependent variables \((u, w, v, T, p, \omega, \psi)\) are functions of \(r, z,\) and time. They are independent with \(\Theta\) the angular direction. The rotational velocity is non-zero and the pressure term will not be solved for. Presented in functional form, the dependent variables are:

\[
\begin{align*}
  u &= u(r, z, t) \\
  w &= w(r, z, t) \\
  v &= v(r, z, t) \neq 0 \\
  T &= T(r, z, t) \\
  \omega &= (r, z, t) \\
  \psi &= (r, z, t)
\end{align*}
\]

The first governing equation is the continuity equation,

\[
\frac{1}{r} \frac{\partial}{\partial r} (ru) + \frac{\partial w}{\partial z} = 0 
\]

The conservation of momentum for the fluid is described by the well known Naiver-Stokes equation. The Naiver-Stokes equation in cylindrical coordinates are simplified using the Boussinesq approximation. The Boussinesq approximation assumes that compressibility may be neglected but the thermal expansion may not and this only modifies the gravitation force. With \(\frac{\partial}{\partial \Theta} = 0\), the Naiver Stokes equations become

\[
\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial r} + w \frac{\partial u}{\partial z} - \frac{v^2}{r} = \frac{-1}{\rho} \frac{\partial p}{\partial r} + \nu \left( \nabla^2 u - \frac{u}{r^2} \right)
\]
\[
\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial r} + w \frac{\partial w}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial z} + \nu \nabla^2 w + g \alpha (T - T_0) \\
\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial r} + w \frac{\partial v}{\partial z} + \frac{uv}{r} = \nu \nabla^2 v - \nu \frac{v}{r^2}
\]

where,

\[
\nabla^2 = \frac{\partial^2}{\partial r^2} + \left(\frac{1}{r}\right) \frac{\partial}{\partial r} + \frac{\partial^2}{\partial z^2}.
\]

The energy equation is

\[
\frac{\partial T}{\partial t} + \frac{1}{r} \frac{\partial}{\partial r} (ruT) + \frac{\partial}{\partial z} (wT) = \kappa \nabla^2 T.
\]

The five dependent variables, \( T, u, v, w, \) and \( p \), are coupled in the five equations. The fluid properties are assumed to be constant over the range of the dependent variables.

To solve equations 2 and 3, the radial and axial velocities are expressed in terms of the Stokes stream function, \( \psi \),

\[
u = \left(\frac{1}{r}\right) \partial \psi / \partial z \quad \text{and} \quad w = -\left(\frac{1}{r}\right) \partial \psi / \partial r.
\]

This definition satisfies the continuity equation, equation 1.

Introducing the vorticity parameter \( \omega \), defined by,

\[
\omega = \partial w / \partial r - \partial u / \partial z,
\]

\text{10}
The stream function and vorticity are related by,

\[
\frac{\partial}{\partial \theta} \left( \frac{1}{r} \frac{\partial \psi}{\partial \theta} \right) + \frac{1}{r} \frac{\partial^2 \psi}{\partial z^2} = -\omega, \]

The pressure term in the Naiver-Stokes equation was eliminated by cross differentiation of the radial and axial components yielding (See appendix I for derivation)

\[
\frac{\partial \omega}{\partial t} + \frac{\partial}{\partial r} (u\omega) + \frac{\partial}{\partial z} (w\omega) + \frac{2\nu}{r} \frac{\partial \psi}{\partial z} = \frac{\partial}{\partial r} \left( \frac{\partial T}{\partial r} \right) + \nu \Delta \omega - \nu \frac{\omega}{r^2} \]

The evaluation of \( \omega \) in equation was not done directly. A new variable called the Svanberg Vorticity term, \( S \), was introduced as proposed by Langious[4]. It is

\[
S = \omega/r \]

This variable is constant for an incompressible fluid where rotational symmetry is present. When the substitution for \( \omega \) is made in equation (9) and simplified it becomes

\[
\frac{\partial S}{\partial t} + \frac{1}{r} \frac{\partial}{\partial r} (r u S) + \frac{\partial}{\partial z} (w S) + \frac{\partial}{\partial z} \left( \frac{\Omega^2}{r^4} \right) \]

\[
\frac{\nu}{r} \frac{\partial T}{\partial r} + \frac{\nu}{r} \frac{\partial}{\partial r} \left[ \frac{1}{r} \frac{\partial}{\partial r} (r^2 S) \right] + \nu \frac{\partial^2 S}{\partial z^2} \]
where the angular momentum is defined by

\[ \Omega = rv \]

From this point onwards, we will use the set of equations 4, 5, 6, 7, 8, 9, 10, and 11 instead of equations 1, 2, 3, 4 and 5.

In order to formulate the boundary conditions the following assumptions were used.

1. The temperatures of the crystal, the crucible and the wall and the top cover of the furnace are uniform.
2. There is no heat transfer from the crucible to the melt, so the temperature of the melt touching the crucible and the crucible are the same.
3. The melt can be treated as an opaque material.
4. At the melt surface, radiation is the only mode of heat transfer.

The boundary conditions in differential form are as follows [1]:

AXIS, \( r = 0 \)

\[ \psi = 0, \omega = 0, v = 0, \frac{\partial T}{\partial r} = 0 \]  

CRUCIBLE BOTTOM, \( z = 0 \)

\[ \psi = 0, w = \frac{\partial u}{\partial z}, v = R_{cru} \Omega_{cru}, T = T_{cru} \]
CRUCIBLE WALL, \( r = R_{\text{cru}} \)
\[
\psi = 0, \quad w = \frac{\partial u}{\partial r}, \quad v = R_{\text{cru}} \quad \Omega_{\text{cru}}, \quad T = T_{\text{cru}}
\]

CRYSTAL FACE, \( z = H, \quad R \leq R_{\text{cry}} \)
\[
\psi = 0, \quad w = \frac{\partial u}{\partial r}, \quad v = R_{\text{crys}} \quad \Omega_{\text{crys}}, \quad T = T_{\text{crys}}
\]

FREE MELT SURFACE, \( z = H, \quad R_{\text{cry}} < r < R_{\text{cru}} \)
\[
\psi = 0, \quad w = 0, \quad \frac{\partial v}{\partial r} = 0, \quad \frac{\partial T}{\partial z} = \frac{\partial T}{\partial z_{\text{melt}}} - \frac{\partial T}{\partial z_{\text{rad}}}
\]

The problem as expressed in the differential form cannot be solved directly. The common approach to a solution is to express each equation in a time dependent difference form. These equations can then be iterated by marching along using small time steps. Since the equations are coupled, each equation must be processed at each time step and used for approximate values for the next step.

Equations 4, 5, 6, 7, 8, and 11 will now be recast in finite difference form.

The rotational velocity equation 4 becomes
\[
\frac{\Delta v_{i,j}}{\Delta t} + \frac{u_{i,j}}{v r_i} \frac{\nabla v_{i,j}}{v r_i} + \frac{w_{i,j}}{v z_i} + \frac{u_{i,j}}{r_i} \frac{\nabla v_{i,j}}{v r_i} + \frac{v_{i,j}}{r_i} \frac{v_{i,j}}{v r_i} = v \nabla^2 v_{i,j} - \frac{v_{i,j}}{r_i^2}
\]

The energy equation 5 becomes
\[
\frac{\Delta T_{i,j}}{\Delta t} + \frac{1}{r_i} \frac{\Delta}{\Delta r_i} (r_i u_{i,j} T_{i,j}) + \frac{\Delta}{\Delta z_j} (w_{i,j} T_{i,j}) = \kappa \nabla^2 T_{i,j}
\]
The radial and axial velocities in equation 6 are
\[ u_{i,j} = \left( \frac{1}{r_i} \right) \frac{\Delta \psi_{i,j}}{\Delta z_j}, \quad w_{i,j} = \left( \frac{-1}{r_i} \right) \frac{\Delta \psi_{i,j}}{\Delta r_i}. \]

The vorticity parameter, as defined by equation 7 becomes
\[ \omega_{i,j} = \frac{\Delta w_{i,j}}{\Delta r_i} - \frac{\Delta u_{i,j}}{\Delta z_j}. \]

The vorticity parameter and stream function are related, and equation 8 becomes
\[ \frac{\Delta}{\Delta r_i} \left( \frac{1}{r_i} \frac{\Delta \psi_{i,j}}{\Delta r_i} \right) + \frac{1}{r_i} \frac{\Delta^2 \psi_{i,j}}{\Delta z_j} = -\omega_{i,j}. \]

Equation 11 for the Svanberg vorticity parameter becomes
\[ \frac{\Delta S_{i,j}}{\Delta t} + \frac{1}{r_i} \frac{\Delta}{\Delta r_i} \left( r_i u_{i,j} T_{i,j} \right) + \frac{\Delta}{\Delta z_j} \left( w_{i,j} S_{i,j} \right) + \frac{\Delta}{\Delta z_j} \left( \frac{\Omega_{i,j}^2}{r_i} \right) \]
\[ = \frac{\alpha g}{r_i} \frac{\Delta T_{i,j}}{\Delta r_i} + \frac{v}{r_i} \frac{\Delta}{\Delta r_i} \left[ \frac{1}{r_i} \frac{\Delta}{\Delta r_i} \left( r_i^2 S_{i,j} \right) \right] + \frac{\Delta^2 S_{i,j}}{\Delta z_j}. \]
Before the finite difference equations can be solved, a grid pattern had to be developed. There are two different types of grid cells that model the fluid in the melt, on-axis and off-axis cells. This was required to handle terms of $1/r$ that would be undefined at the axis in the Svanberg equation. So, an entirely separate set of integral equations had to be developed as described in appendix 2. The cells bounding the top and bottom surface are only one half in height. This puts the grid coordinate on the boundary. The interior points in the energy and rotational equations were solved using central difference method and another set of equations were required to evaluate the boundary points using either forward or backward difference method. As can be seen several sets of equations were required to solve this problem. These methods were employed for each term in equations 13, 14, 15, 16, 17 and 18. These equations are coupled with each other, so they had to be solved as a set of equations rather than separate equations. The finite difference form of the boundary conditions is presented later in the section on the method of solution.
METHOD OF SOLUTION

A single solution can now be found by solving the difference form of the equations for each particular set of conditions. The basic approach was to assume initial values for the independent variables, \( t, u, v, w, \omega, \) and \( \psi \). Then a delta that represents a change in these variables would be calculated to improve the initial values. Now using the improved values, a new delta is calculated and this process was repeated until the delta change became very small and could be neglected. There were six sections to the solution as outlined below.

1. Dimension and initialize the parameters (or use values from last run)

2. Calculate new radial and axial velocity components

   Calculate:
   
   a. \( \Delta S \)
   b. \( S \) from \( \Delta S \)
   c. \( \omega \) from \( \Delta S \)
   d. \( \psi \) from \( \Delta \omega \), boundary conditions
   e. \( u, \omega \) from \( \Delta \psi \)

3. Calculate new temperature distribution

   Calculate:
   
   a. \( \Delta T \) conduction, Melt
3. (cont'd)
   b. $\Delta T$ conduction, Surface
   c. $\Delta T$ radiation, Surface
   l. Viewfactor
      a. grid locations (surface, wall, shield)
      b. Confac II
   d. $T_{\text{new}} = T_{\text{old}} + \Delta T/\Delta t$

4. Calculate new rotational velocity
   a. Boundary condition (rotation of crystal +
   b. Velocity profile (axial & radial)
   c. $v_{\text{new}} = v_{\text{old}} + \Delta v/\Delta t$

5. Check for the vorticity parameter (feedback)
   a. Calculated from velocities and averaged with vorticity
      calculated from S.

6. Check for steady-state to stop calculation.

The flow chart for the computer program is given in Appendix 4.
Now each of these steps will be discussed in more detail, starting with the initialization of the independent variables. This was done in two methods 1) initial guess or 2) use the values from the previous run. In the first case, \( S, \psi, \omega, u, \) and \( w \) were set to zero and estimated values for \( T \) and \( w \) were calculated. Estimated values for \( T \) and \( w \) were obtained by a proportional average of the fixed boundary conditions. In the second case, the values of all the independent variables were stored and used to initiate the next run.

Svanberg Vorticity Equation

The next step is to calculate the new values of axial and radial velocities via \( S \) and \( \psi \). The new value of \( S \) is obtained by calculating each term of equation 18 and summing, then multiplying by \( \Delta t \). Equation 18 can be symbolically written in the following form.

\[
\frac{\Delta S_{i,j}}{\Delta t} = (\Delta S_{i,j})_{ra} + (\Delta S_{i,j})_{za} + (\Delta S_{i,j})_{c} + (\Delta S_{i,j})_{b} + (\Delta S_{i,j})_{rv} + (\Delta S_{i,j})_{zv}
\]
The physical meaning of each of the terms in equation 19 can be related by the subscripts as follows: radial advection (ra), axial advection (za), coupling term (c), bouyancy term (b), radial viscous (rv), axial viscous (zv).

Each term in equation 19 is integrated over each cell to obtain S by the following scheme.

One of the terms will be expanded (for more detail see appendix 2).

\[ V \left( \frac{\Delta S_{i,j}}{\Delta t} \right) \text{ ra} = \int \int \int \left( \frac{1}{r} \frac{\partial}{\partial r} \left(ruS\right) \right) dV \]

\[ V = \pi \Delta z \left(r^2 - r'^2\right) = 2\pi \Delta r \Delta z \]

\[ V \left( \frac{\Delta S_{i,j}}{\Delta t} \right) = 2\pi \int \int \left[ruS\right]_{r_{i-}}^{r_{i+}} dz \]

\[ \left( \frac{\Delta S_{i,j}}{\Delta t} \right) = \frac{1}{r} \left( r_{i-} \langle u \rangle_{R_{i-}} - \langle S \rangle_{R_{i-}} - r_{i+} \langle u \rangle_{R_{i+}} + \langle S \rangle_{R_{i+}} \right) \]
The evaluation of the weighted average over each cell (Figure 2) designated by $\langle f \rangle_+$ where $f$ is a generic variable and is defined as:

$$\langle f \rangle_+ = \frac{1}{\Delta z} \int_{z-}^{z+} f(r_+, z) \, dz$$

$$\langle f \rangle_+ = \frac{1}{r \Delta r^+} \int_{r_i-}^{r_i+} rf(r, z) \, dr$$

$$\langle u \rangle_+ = \langle u_{r_i} \rangle_+ \langle f \rangle_{r_+} \langle u \rangle_{z+} = \langle w_{z+} \rangle \langle f_{z+} \rangle$$

where,

$$r_+ = r + \Delta r / 2 \quad z_+ = z + \Delta z / 2$$

For example

$$\langle f \rangle_{R+} = \frac{1}{2 \Delta z} (f_{r+, z} - f_{r+, z-}) \Delta z$$

To calculate

$$\langle f \rangle_{r+, z+} = r + \Delta r / 2, z + \Delta z / 2 \quad \text{(Point A in Figure 3)}$$

$$= (f_{i,j} + f_{i+1,j} + f_{i,j+1} + f_{i-,j+1}) / 4$$
\[
<f>_{r+z} = f_r + \Delta r/2, z - \Delta z/2 \text{ (Point B Figure 3)}
\]
\[= (f_{i,j} + f_{i+1,j} + f_{i,j-1} + f_{i-1,j-1})/4 \]

\[A \quad r+, \quad z+
\]
\[C \quad r+, \quad z\]
\[(i,j) \quad B \quad r+, \quad z-
\]

FIGURE 3
r+ and z- Grid Locations

Then returning to equation (27), equation for point C in Figure (3) becomes

\[
<f>_{r+} = (f_{i,j+1}, f_{i,j-1}, f_{i+1,j+1}, f_{i+1,j-1})/4
\]
or the average of the four exterior points rather than simply the average between the \((i,j)\) and \((i+1,j)\) terms.

Due to the fact that some terms of equation 18 have a \(1/r\) term, two types of grid cells were used as seen in Figure 2 to evaluate values at \(r = 0\). This required two complete sets of evaluations of the terms in equation 19.
Once $\frac{\Delta S}{\Delta t}$ has been calculated, the new value of $S$ is simply arrived at by equation

$$S_{\text{new}} = S_{\text{old}} + \left( \frac{\Delta S}{\Delta t} \right) \Delta t.$$ 

Using the new value of $S$, $\omega$ is then calculated by equation 10.

Equation 8 was then used to be solved for the $\Psi$ values for each point in the interior. This is done directly presented in [3] rather than iterative method. This method uses matrix manipulation, in the following simplified sequence.

1. Form a matrix of grid locations.
2. Calculate eigenvalues for the above matrix using the subroutine TQL2 ref. [9].
3. Transform axes to get diagonal matrix using the QL algorithm.
4. Solve transformed equation.
5. Re-transform solution for actual values.

A brief summary of the iterative method will be given here [2]. This solution of the stream function entails evaluating the equation (7). This is done by applying successive overrelaxation to obtain the relation between successive iterates $\Psi^n$ and $\Psi^{n+1}$.
where $\beta$ is a relaxation factor. As present in [3], the direct method is about 5 times faster than the iterative method and there is no possibility of false convergence.
ROTATIONAL EQUATION

Equation 13 was used to calculate the rotational velocity \( v \). An initial estimate of the rotational velocity was needed before the other velocities could be calculated. These estimates were obtained by iterating the static term, \([v]^2\) term of equation 3. To speed up convergence, a proportional average for the interior points was calculated from the boundary values. It turned out that a relatively good estimate of the rotational velocity was required to reduce instability in the main program. These estimated values were fed into the main program to obtain the radial and axial velocities. There was some difficulty in evaluating the dynamic terms (velocity coupled terms) next to the boundaries. To overcome this problem the dynamic terms next to the boundaries were set to zero so the rest of the problem could be solved. Updating the rotation velocities required two arrays, an old and a new. The new value was obtained by multiplying \( \Delta v/\Delta t \) by \( \Delta t \) and then adding to the old value.

ENERGY EQUATION

The energy equation, equation 14 was the third major equation to be solved. It is

\[
\frac{\partial T}{\partial t} + \frac{1}{r} \frac{\partial}{\partial r} (ruT) + \frac{\partial}{\partial z} (wT) = \kappa \nabla^2 T ,
\]
where $\kappa$ is the thermal diffusivity.

This equation was handled in the same manner as the rotational equation in terms of establishing an estimate of initial profile and updating the new profile. This equation was processed in the same manner as the rotational equation was. The initial estimate was a proportional average and the updating of the new profile was done in the same way. The velocity coupled terms were not set to zero as in the case of the rotational equations. The boundary conditions were more complicated, since the convection and radiation components were to be combined.

Once the interior values of $\Psi$ are obtained, the boundary conditions can be calculated. A summary of the boundary conditions adapted for the finite difference scheme is given below [2].

AXIS, $r = 0$

$$\Psi = 0, \quad \omega = 0, \quad v = 0, \quad \frac{\Delta T}{\Delta r} = 0$$  \hspace{1cm} 34a

CRUCIBLE BOTTOM, $z = 0$

$$\Psi = 0, \quad \omega = \frac{-2}{r(\Delta z)^2} \Psi (r, \Delta z), \quad v = R_{cru} \Omega_{cru}, \quad T = T_{cru}$$  \hspace{1cm} 34b

CRUCIBLE WALL, $r = R_{cru}$

$$\Psi = 0, \quad \omega = \frac{-2}{R_{cru}(\Delta r)^2} \Psi \left( R_{cru} - \Delta r, z \right), \quad v = R_{cru} \Omega_{cru}, \quad T = T_{cru}$$  \hspace{1cm} 34c
CRYSTAL FACE, \( z = H, R \leq R_{cry} \)

\[ \psi = 0, \quad \omega = \frac{-2}{r \, (\Delta z)^2} \, \psi (r, H-\Delta z), \quad v = \frac{\Omega}{R_{cry}} \, \Omega_{crys} \, T = T_{crys} \quad 34d \]

FREE MELT SURFACE, \( z = H, R_{cry} < r < R_{cru} \)

\[ \psi = 0, \quad \omega = 0, \quad \frac{\Delta v}{\Delta z} = 0, \quad \frac{\Delta T}{\Delta t} = \Delta T_{conv.} - \Delta T_{rad.} \quad 34e \]

The \( \Delta T_{radiation} \) term requires a careful scrutiny and is described in the following section.
RADIANT ENERGY BALANCE AT THE MELT SURFACE

There are three major considerations in the radiation boundary conditions at the melt surface, the net heat loss from the surface, the heat transfer from the cooling crystal, and the temperature profile of the melt surface. These were determined by considering a system with and without a radiation shield. To facilitate this evaluation, the problem was broken into three areas, surface equilibrium, radiation viewfactors and the radiation network equations.

Melt Surface Equilibrium

The melt surface elements are at thermal equilibrium when their change in internal energy is zero. The change in internal energy, $\Delta u$ in the finite difference form is

$$\frac{\Delta U}{\Delta t} = -\rho c_v V \frac{\Delta T_{i,j}}{\Delta t}.$$  

where $\rho$ is the density, $c_v$ is the specific heat, $V$ is the volume.

For equilibrium, $\Delta T_{i,j}/\Delta t$ must be equal to zero. This change in temperature of the surface element is zero when the convective heat transfer $Q_{\text{conv}}$, from the melt equals the radiation heat transfer $Q_{\text{rad}}$ to the surroundings.
The quantity $Q_{\text{conv}}$ in the absence of other mechanisms of heat transfer from a given surface element causes a temperature change of the element. The rate of this temperature change is designated by $(\Delta T/\Delta t)_{\text{conv}}$. In the presence of radiative heat transfer from the surface of the element, the temperature of the element changes at a rate $(\Delta T/\Delta t)_{\text{rad}}$. The difference between these two rates is responsible for the net change in the internal energy of the element. Thus rewriting equation 35

$$\frac{\Delta U}{\Delta t} = Q_{\text{conv}} - Q_{\text{rad}}$$

and then solving for

$$\frac{\Delta T_{i,i+1}}{\Delta t} = \frac{1}{\rho c V} (Q_{\text{conv}} - Q_{\text{rad}})$$

The convection term of equation 36 can be calculated by the energy equation, so to solve this equation, an expression for the radiation term must be developed.
Radiation Network Equations

To calculate the heat loss due to radiation, an electrical equivalent circuit of the radiation heat flow was constructed (see Figures 4 and 5). The wall, top plate, and the crystal are at known temperatures and the N surface element temperatures varied from iteration to iteration. The iterations were needed to solve the governing momentum equations until the heat transfer by radiation equaled the heat loss due to conduction. The general nodal equation has the form

\[
\frac{E_b - J}{R_i} + \frac{J_w - J_i}{R_{i-w}} + \frac{J_{sh} - J_i}{R_{i-s}} = 0
\]

where,

\[E_b = \sigma T_1^4\]

\[R_i = \varepsilon \frac{(1-\varepsilon_i)}{\varepsilon A_i}\]

\[R_{i-w} = 1/A_i F_{1-n}\]

\[\sigma = 5.67 \times 10^{-8} \text{ W/} (\text{K}^4 \text{ m}^2)\].

In the above equation, \(E_b\) is the blackbody emissive power, \(J\) is the radiosity, \(R_i\) is the surface resistance, \(R_{i-x}\) is the spacial resistance, \(\varepsilon\) is the emissivity, \(F_{1-n}\) is the radiation viewfactor, \(A\) is the surface area and \(\sigma\) is the Stefan-Boltzmann constant.

Equation 38 is then put in matrix form and solved. This enables the calculation of the heat loss due to radiation.
This equation is broken into 3 matrixes, A the resistance elements, J the radiosity unknowns and B the emissive power of the elements. The areas used for calculating the resistance values were of the complete element, for example, a ring for the surface elements, or the total surface area of the crystal. The radiation viewfactor's will be discussed in the next section. The system of equations in matrix form was:

\[
\begin{bmatrix}
A
\end{bmatrix}
\begin{bmatrix}
J
\end{bmatrix} =
\begin{bmatrix}
B
\end{bmatrix}
\]

\[
J = \begin{bmatrix}
J_1 \\
J_2 \\
\vdots \\
J_n \\
J_w \\
J_{\text{cry}} \\
J_{\text{ftp}} \\
J_{\text{sh}_\text{out}} \\
J_{\text{sh}_\text{in}} \\
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
E_{b1}/R_1 \\
E_{b2}/R_2 \\
\vdots \\
E_{bn}/R_n \\
E_{bw}/R_w \\
E_{b\text{cry}}/R_{\text{cry}} \\
E_{b\text{ftp}}/R_{\text{ftp}} \\
0 \\
0 \\
\end{bmatrix}
\]
$E_{ftp} = 0.084$

$R_{ftp} = 0.0016$

$J_{ftp} = 1.540$

$R_{ftp-w} = 0.0088$

$E_{w} = 0.084$

$R_{crys} = 5.207$

$R_{crys-ftp} = 0.0061$

$J_{crys} = 3.261$

$R_{ftp-i} = (0.0504 - 0.0884)$

$E_{bi} = (11.51 - 13.39)$

$R_{crys-i} = (0.0958 - 0.1971)$

$R_{crys-w} = 0.0281$

$J_{w} = 1.882$

$R_{w} = 0.0038$

$R_{i-w} = (0.1335 - 0.0726)$

$J_{i} = (5.058 - 5.660)$

$R_{i} = (0.0486 - 0.1155)$

UNITS

$R = \frac{1}{cm^2}$

$J = \frac{cal}{sec/in} \left( \frac{4.18 W}{cm} \right)$

FIGURE 4 - NETWORK SYSTEM WITHOUT SHIELD
\[ E = 0.084 \]

\[ J_{ftp} = 0.715 \]

\[ J_{cry} = 1.94 \]

\[ J_{sh_{out}} = 2.07 \]

\[ J_{sh_{in}} = 6.51 \]

\[ J_i = (7.82 - 7.7) \]

\[ R_{cry} = 0.017 \]

\[ R_{crys-ftp} = 0.0061 \]

\[ R_{crys-sh} = 0.0141 \]

\[ R_{ftp} = 0.0016 \]

\[ R_{ftp-sh_{out}} = 0.007 \]

\[ T_{sh} = 1334.0^\circ K \]

**UNITS**

\[ R = \frac{1}{\text{cm}^2} \]

\[ J-E = \frac{\text{cal}}{\text{sec} \cdot \text{in}} \left( \frac{4.18 \text{ W}}{\text{cm}} \right) \]

**FIGURE 5 - NETWORK SYSTEM WITH SHIELD**

32
The resistance matrix was 17 x 17 for the case with shield. This set of simultaneous equations were then solved for the radiosity values by using a subroutine called LEQ2S from the Statistical and Mathematics library. This routine allowed the inverse matrix of (A) to be saved for future calculations.

The next step was to calculate the radiant heat loss $Q_{\text{rad}}$ from the surface elements. Heat loss in the presence of radiation is:

$$Q_{\text{rad}} = \frac{(E_{bs} - J_i) A}{[(1-e_i/e_i)]}.$$  

Using equation 37 ($Q_{\text{conv}} = 0$) and equation 40

$$\frac{(\Delta T_{i,j})}{\Delta t} = \frac{Q_{\text{rad}}}{\rho c_v V} = \frac{(E_{bs} - J_i)}{[(1-e_i/e_i)] (\rho c_v \frac{\Delta z}{2})}.$$  

Equation 37 can be rewritten in the following form.

$$\frac{\Delta T_{i,j}}{\Delta t} = \left(\frac{\Delta T_{i,j}}{\Delta t}\right)_{\text{conv}} - \left(\frac{\Delta T}{\Delta t}\right)_{\text{rad}}.$$  

The new temperature at the surface becomes

$$T_{\text{new}} = T_{\text{old}} + \left(\frac{\Delta T_{i,j}}{\Delta t}\right) \Delta t.$$  

33
A steady state was assumed to have been reached when rate of change of temperature was less than .05°K/sec. This rate of change produced a change in the final temperature values of less than 1°K when allowed to run 200 iterations. The resistance values that make up the network matrix must be calculated and this requires the evaluation of the radiation viewfactors between the radiating surfaces.

View Factors

The radiation viewfactors between the free surface of the melt, crystal, wall, shield, and furnace top had to be obtained. A computer program, Confac II [7] was used to evaluate these viewfactors. Consider any two elemental areas A and B. In general area A may see all of area B or part of area B, or none of it. Which one of these three possibilities is applicable to a given pair of elemental surfaces, depends upon whether the rays connecting the edges of surfaces A and B are blocked by another surface such as the crystal.

Radiation viewfactor $F_{1-2} = \frac{\text{Radiant energy received by surface #2}}{\text{Radiant energy leaving surface #1}}$

The surface-to-shield viewfactors were calculated by determining the area "seen" by a surface element. This was done by constructing the surfaces and tangent planes with equation and then outlining the observed area with grid points.
**TOP VIEW**

**TYPICAL SURFACE ELEMENT**

**R_{cry}\**

**R_{wall}\**

**FIGURE 6 a - SURFACE-TO-SHIELD VIEWFACTOR**

**SHIELD POINT LOCATIONS**
FIGURE 6b - SURFACE-TO-SHIELD VIEWFACTOR

FURNACE VIEWS
1. Determine equation of shield surface

\[ x_1^2 + y_1^2 - (c_1 z + b)^2 = 0. \]

\[ c_1 = \text{slope of the shield cone} \]
\[ b = \text{intercept (} z = 0 \text{)} \]

2. Determine the equation of the surface tangent to the shield that passes through the surface element grid point.

\[ (x_2-x_1) \ 2x_1 + (y_2-y_1)2y_1 - (z_1-z_2) - (c_1 z_1 + b)c_1 = 0 \]

\[ (x_1,y_1) = \text{point on shield} \]
\[ (x_2,y_2) = \text{point on surface} \]

3. Solve equation (45) for \( x \),

\[ x_1 = (c_2^2 - y_1^2)^{1/2} \]
\[ c_2 = c_1 z_1 + r_{\text{shield}} \]

4. Substitute \( x \) into equation (46)

\[ y_1 = c_2^2 \left( \frac{C_2-C_3}{x_2} \right)^{1/2} \]
\[ C_3 = C_1 (c_1 z_1 + b) \]

5. Calculate \( X, Y \) at the top and bottom of the shield (points \( C, B \) in figure 6) for a given \( Z \).
6. The shield is defined by a set of points calculated between points A-B-C-D-A and using equal spacing determined by dividing the arc.

a. Calculate the angle \( \phi = \tan^{-1} \frac{y_1}{x_1} \)

b. Divide by the number of division desired (n)

c. Calculate the points by

\[
X = R_{\text{shield}} \sin (n-1) (\phi/n) \\
Y = R_{\text{shield}} \cos (n-1) (\phi/n)
\]

This procedure was repeated for each grid cell on the surface of the melt. The subroutine CONFAC II required the location of the points and an outline of the surface defined by the points listed counter clockwise. For the rest of the viewfactor calculations see appendix 3.

In figure 6, the area of the shield seen by element A is symmetric about any radial line. So, only half of the surface was described and the viewfactor was then doubled. For each viewfactor the program was written for the general case, so if the size or location of the system was different, the program could calculate new grid points and a new viewfactor. This feature allowed the evaluation of the effect of height and angle of the shield. The viewfactor had to be calculated for both cases, with and without the shield because many surface elements did not interact with one another when the shield was introduced. This
method was checked by calculating the viewfactor of an enclosed area. The sum was very close to 1, as it is supposed to be. The problem has been completely defined so the evaluation and problems encountered in running the program will be discussed next.
COMPUTER PROGRAM AND STABILITY

The computer used to solve this problem was a VAX Computer, located at the Rochester Institute of Technology and the program language was Fortran. The program (appendix 5) consisted of a main program and five subroutines (2 original and 3 pre-program). A detailed flow chart can be found in appendix 4. The main program computed the fluid velocity and temperature profiles. Two subroutines calculated the shapes of the elements for input to Confac II, a subroutine to evaluate the radiation shape factors, and evaluated the radiation heat transfer at the surface. The three pre-programs were TQL2 which solved the eigenfactor, LE01S which solved the system of simultaneous radiation network equations, and Confac II that calculated the radiation viewfactors. A relatively small number of grid points (400) was used for economizing on the storage space in the computer and the computation time.

A major problem in solving these types of fluid equations is stability. The other problem is amount of time required to determine where the results are going. The two problems are interrelated; the smaller the time step the more stable it becomes but longer time is required to determine whether the results are right or not.

There were four areas where stability was a problem. The initial estimates of the rotation and temperature profile was the
first area of concern. Poor initial values of these estimates cause the stream function to become unstable. To solve this problem, the initial values for the rotation and temperature were improved by a better proportioning scheme and more iterations. A second method was to store the previously run values for these parameters to start the next run.

The second source of instability was the time interval. After each iteration the maximum permissible time interval for the temperature and rotation equation were calculated. If the value of the time step negative, a small value such as .002 sec. was used for the time step. If the value was greater than zero and less than .02 second, it was used as is, and if it was greater than .02, it was set to .02. Larger values caused instability in the other equations.

The third source of instability was found in the method of evaluating some of the differencing equations. The first attempt to solve the radial viscous term, led to instability as can be seen in the following example.

\[ (\Delta S_{i,j})_{rv} = \frac{v}{r_i} \frac{\Delta r}{r_i} \left[ \left( \frac{1}{r} \frac{\partial}{\partial r} (r^2 S) \right)_{r_{i+}} - \left( \frac{1}{r} \frac{\partial}{\partial r} (r^2 S) \right)_{r_{i-}} \right] \]

The first step was to use the product rule on \( \frac{\partial}{\partial r} (r^2 S) \)

\[ \frac{\partial}{\partial r} (r^2 S)_{i+} = (2rS + r^2 \frac{\partial S}{\partial r})_{i+} \]

49

50
Refer to fig. 7 to get a visual picture of what is happening

\[
\begin{array}{ccc}
  & i-j, j+1 & \\
  i-1, j-1 & i, j & i+1, j \\
  i-1, j & i, j+1 & \\
  & i, j-1 & i+1, j-1
\end{array}
\]

Figure 7 - Instability Grid Locations

To evaluate the first term \( r_+ \)

\[
2rS \left|_{r_+} \right. = 2r_+ \left( S_{i+1, j} + S_{i, j} + S_{i+1, j-1} + S_{i, j-1} \right) / 4
\]

\[
r \frac{\partial S}{\partial r} \left|_{r_+} \right. = \frac{r_+}{2\Delta r} \left( S_{i+1, j+1} - S_{i, j+1} + S_{i+1, j-1} - S_{i, j-1} \right)
\]

Now when \( r \)-terms are calculated

\[
2rS \left|_{r_-} \right. = 2r_- \left( S_{i-1, j+1} + S_{i-1, j-1} + S_{i, j+1} + S_{i, j-1} \right) / 4
\]

\[
r \frac{\partial S}{\partial r} \left|_{r_-} \right. = \frac{r_-}{2\Delta r} \left( S_{i, j+1} - S_{i-1, j+1} + S_{i, j-1} - S_{i-1, j-1} \right)
\]

If the values of the grid points develop in a pattern like fig. 8, equation 49 will propagate this pattern. When examining the \( r^2 \) terms (much larger than the other terms), the four outer terms stay the same sign and when the \((i, j+1), (i, j-1)\) terms are subtracted they change sign. The result is the center grid point
that is being evaluated goes larger in the same sign as the outer points. This makes the next point get larger in the other direction. Thus, the equation goes unstable.

\[
\begin{array}{cccc}
+ & - & + & - \\
- & +B & -A & + \\
+ & - & + & - \\
\end{array}
\]

Figure 8 - Instability Grid Values

To overcome this problem, instead of using the \((i+1,j+1), (i-1,j-1)\) term for approximating \((i+1,j)\) or \((i,j+1), (i,j-1)\) for \((i,j)\) the \(2(i+1,j)\) or \(2i,j\) term should be averaged in. Equations 51b and 52b become

\[
(\Delta S_{i,j}) = \frac{\nu}{r_i} \left( \frac{1}{2} \left( S_{i+1,j+1} + S_{i+1,j-1} \right) - \left( S_{i-1,j+1} + S_{i-1,j-1} \right) \right)
\]

\[+ \frac{1}{2\Delta r} \left( r_+^2 \left( S_{i+1,j+1} + S_{i+1,j-1} \right) - r_-^2 \left( S_{i-1,j+1} + S_{i-1,j-1} \right) \right) - \left( r_+^2 + r_-^2 \right) \left( S_{i,j+1} + S_{i,j-1} \right) \]

where \(S_{r+,z+} = (S_{i,j+1} + S_{i+1,j-1})/2\)

\[S_{r+,z} = (S_{i,j} + S_{i+1,j})/2.\]
Now these equations involve

\[ S_{i,j+1}, S_{i,j-1}, S_{i+1,j}, S_{i-1,j} \]

Now if point A in fig. 8 is calculated, the plus terms are added to the minus terms before the difference is taken. This fact made the equation stable and this scheme was also used in evaluating the other terms to prevent instability.

Instability was also due to trying to satisfy the vorticity boundary conditions. A gradually decreasing grid size near the boundary was used in the beginning but this complicated the averaging schemes so it was dropped. With the smaller grid size, the reverse flow just under the crystal (dotted area in fig. 9), can be obtained. This does compare with Dr. Langlois' results [ref. 2].

Address the vorticity boundary conditions, a type of feedback was used. The problem was the flow patterns were well behaved but it took as many as 600 iterations. Then the flow (ψ) pattern got smaller until the values went negative and went unstable or unbounded. After studying the numbers and varying some of the values, a check on the vorticity term was calculated using the calculated values of velocities and the equation 7.
This value of $\omega$ was called Wcheck. Wcheck was subtracted from the $\omega$ calculated from $S$, as a measure of accuracy. The feedback was achieved by averaging and Wcheck and recalculating $Shi$ and the velocities. This mechanism slowed the rate of growth of flow field but kept the boundary values from going unstable. The accuracy of the interior points was very good, and the boundary point values was satisfactory.

All these methods to keep stability under control added more calculations per iteration. For example, while debugging the equation, a four minute cpu time was used because that was the maximum before it became a special job. The program took about $1\frac{1}{2}$ minutes to compile, which left 2 1/2 minutes for calculation. The amount of iteration dropped from around 150 to 300 iterations. This led to very long runs of 40 minutes which meant overnight service and sometimes required another 40 minutes (80 minutes total). This made it very difficult to evaluate the relative effectiveness of each of these methods but the feedback method seemed to have the most dramatic effect on controlling the velocity profile.

The method used to determine when a steady state was to monitor the (difference between successive values of $\Psi$). The difference or error was calculated on a absolute basis. The square root of sum of the difference divided by the sum of the absolute value of the sum and the time interval. This gave a good
indication when the values had reached an approximate solution. When this error value remained relatively constant, the program was stopped, and the fluid and temperature profiles were examined.

Fluid, Temperature, and Radiation Profiles

The results can be broken down into three areas, 1) flow profile, 2) temperature profile, and 3) effects of the radiation shield. The values for the various parameter used are listed in Table 1 (page 6a).

The flow profile can be seen in Figure 9. (The actual values are in appendix 6.) This is a plot of constant values of the stream function \( \psi \) or flow velocity in the radial and axial directions. The velocity was determined only to feed into the temperature equation so a finer grid was not used. The fine grid would have shown more detail of the flow such as the Taylor column effect or the reverse flow under the crystal as shown by a dash line ref.[5]. The rotational velocity was basically proportional to the radius times the crucible or crystal rotation. The influence of the reverse rotation of the crystal could be seen about three cells deep at which the flow reversed direction. This was the cause of the reverse flow pattern of the stream function.
The temperature profile as seen in Figure 10, shows grid points of equal temperature. The results compare to earlier papers [ref. 2, 4, 5], except for the effect of the radiation shield. The effects of the shield can be seen in Figure 11. The surface temperature is plotted as a function of the radius. The four curves represents a no shield, 45 degree shield, with and without compensation, and the one obtained from reference 3 for comparison. Note: Free radiation and a smaller grid size were used in reference 3.
FIGURE 10 - TEMPERATURE ISOTHERMS (WITH SHIELD)
FIGURE 11 - SURFACE TEMPERATURE
The effect due to a change in angle of the shield was small. The effect of height of the shield from the surface was also small. Three runs with different heights were made, 0, 1, 2 cm above the melt with the same slope angle, and there was less than a degree drop as the shield was raised.

The surface temperature plot (Figure 11) shows the difference between using a radiation shield and not. The initial dip in the plot with the shield is due to a gap between the shield and the crystal where free radiation occurs. The no shield curve shows a large temperature drop due to the loss of surface radiation. The curves demonstrate that the crucible temperature with the shield in place could be lowered and still maintain the melt temperature above the freeze point. The crucible wall temperature can be lowered by 60K and still maintain the same temperature around surface temperature around the crystal.

Results

There were two areas of change on the crystal growing process with the use of a radiation shield, 1) the cooling rate of the crystal, 2) the surface temperature of the melt. As the crystal grows, heat of fusion has to be dissipated and most of this heat is given off in the form of radiation from the crystal. So, the rate at which the heat leaves the crystal governs the rate of growth. The melt surface is the only part of the system which is at a high
temperature and therefore it acts as a heat source. This heat source heats the crystal and thus retards the cooling process for the crystal. The radiation shield reduces this effect of the melt surface by providing a barrier to heat transfer from the surface.

\[ Q_{\text{crys}} = \frac{E_{b\text{crys}} - J_{\text{crys}}}{R_{\text{crys}}} \]

This heat loss was calculated for the system with and without the shield and the percent increase in the rate of cooling is obtained by

\[ \% = \frac{Q_{\text{crys}} (w/o) - Q_{\text{crys}} (w)}{Q_{\text{crys}} (w)} \times 100 = 40\% \]

This percent is strongly influenced by the average crystal temperature, in this case 1400°K was used. The higher the temperature the lower the percentage and visa versa.

When the shield was introduced, the melt surface temperature increased as seen in Figure 11. This was not a desirable condition because this surface would have a higher potential to emit radiation. This meant that for a proper comparison of the heat losses from the crystal with and without shield, it was necessary to have the same radiant energy from the melt surface in both cases. To make the two surfaces equivalent, the sum of the temperatures to the fourth power were made equal.

\[ (\sum T_1^4)^{\text{without shield}} = (\sum T_1^4)^{\text{with shield}} \]
In order to ensure the above condition, the crucible temperature of the system with the shield was modified while leaving other parameters in Table 1 unchanged.

In order to satisfy equation 56, the temperature of the crucible had to be lowered by 60 K from the value that was used for the case without the shield. The radiation shield stabilized at 1334 K.

The velocity and temperature profiles (figures 9, 10) correlated well with those obtained by Langlois, [5] except for the reverse flow pattern under the crystal (figure 9). This small disagreement is due to the relatively coarse grid used. This inconsistency can be eliminated by employing a finer grid, as it was confirmed in a special run. However, this run entailed large amounts of computer time. Since the reverse flow pattern under the crystal does not affect the radiation heat transfer at the surface, a coarse grid was used to economize computer time. The lowering of the crucible temperature and increasing the rate of cooling of the crystal partially explains the experimental result by Lane [ref. 10] of 25% increase of crystal growth with a radiation shield.
DISCUSSION AND CONCLUSIONS

The important result was that less heat was transferred to the crystal from the melt with a radiation shield. This has two implications, faster cooling of the crystal producing higher growth rates and lower energy consumption (lower crucible temperature). The shield helps in two areas, near the crystal and the crucible wall. With the same wall temperature the melt temperature is higher in both areas. This will reduce the occurrence of wall freeze that stops crystal growth. Also, the wall temperature can be lowered to achieve higher growth rates without surface freeze. Some experimental work done by Lane [10] shows a 25% increase with a radiation shield. There are other areas that also can account for this increase that were could be investigated in the future. These areas are in the total energy picture of the growing process. As the shield is keeping the heat from leaving the surface, it is also reducing the direct radiation to the crystal. This will allow the crystal to cool faster thus, explaining some of the increased growth rate.

The Czochralski crystal growing process has three major components that are interrelated; the crystal, the melt and the furnace. Much work has been done in analyzing the melt and some work on the melt and solid crystal interface. This thesis evaluates the interactions between the melt and crystal to the furnace thru radiation heat transfer. Further work could entail
the entire system, involving the convection heat transfer of the melt, the heat of fusion at the crystal interface and the radiation heat given off from the melt and crystal.
REFERENCES


6 Patrick J. Roache, Computational Fluid Dynamics, (1972), Hermosa Publishers, Albequerque, N.M.


9 P. J. Harliman, Lecture Notes in Computer Science.

APPENDIX

1. Naiver-Stokes and Rotational Equations

2. Svanberg Vorticity

3. Addition Viewfactor Calculations

4. Flow Chart

5. Computer Program

6. Numerical Results
Naiver-Stokes equation in Polar Coordinates

\[
\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial r} + \frac{\nu}{r} \frac{\partial u}{\partial \theta} + w \frac{\partial u}{\partial z} - \frac{u^2}{r} = -\frac{1}{\rho} \frac{\partial p}{\partial r} \\
+ \nu \left( \nabla^2 u - \frac{u^2}{r^2} - \frac{2}{r^2} \frac{\partial u}{\partial \theta} \right)
\]

\[
\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial r} + \frac{\nu}{r} \frac{\partial w}{\partial \theta} + w \frac{\partial w}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial z} + \nu \nabla^2 w + g \alpha (T-T_0)
\]

\[
\frac{\partial \theta}{\partial t} + u \frac{\partial \theta}{\partial r} + \frac{\nu}{r} \frac{\partial \theta}{\partial \theta} + w \frac{\partial \theta}{\partial z} + \nu \frac{\partial \theta}{\partial r} = -\frac{1}{\rho r} \frac{\partial \nabla}{\partial \theta}
\\
+ \nu \left( \nabla^2 \theta + \frac{2}{r^2} \frac{\partial \theta}{\partial \theta} - \frac{\nu^2}{r^2} \right)
\]

where \( \nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{1}{r} \frac{\partial}{\partial r} + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2} + \frac{\partial^2}{\partial \theta^2} \)

by symmetry \( \frac{\partial}{\partial \theta} \) terms = 0

Canceling out the Pressure terms

Now \( \frac{\partial}{\partial z} \) (eq. 1)

\[
\frac{\partial}{\partial z} \left( \frac{\partial u}{\partial t} \right) + \frac{\partial u}{\partial z} \frac{\partial u}{\partial r} + u \frac{\partial^2 u}{\partial z \partial r} + \frac{\partial w}{\partial r} \frac{\partial u}{\partial z} \\
+ w \frac{\partial^2 u}{\partial z^2} - \frac{\partial}{\partial z} \left( \frac{u^2}{r} \right) = \frac{\partial p}{\partial r} \frac{\partial u}{\partial z} + \nu \frac{\partial}{\partial z} \left( \nabla^2 u - \frac{u^2}{r^2} \right)
\]

\[
(4)
\]
Now \( \frac{3}{2} (\text{eq. } 2) \)

\[
\left( \frac{\partial}{\partial r} \left( \frac{\partial w}{\partial z} \right) + \frac{\partial u}{\partial r} \frac{\partial w}{\partial r} + u \frac{\partial^2 w}{\partial r^2} + \right.
\]

\[
\frac{\partial w}{\partial r} \frac{\partial w}{\partial z} + w \frac{\partial^2 w}{\partial r \partial z} = -\frac{\partial p}{\partial r \partial z} + \frac{1}{\partial r} (u \nabla^2 w) + g \alpha \frac{2}{\partial r} (T - T_0)
\]

(5)

Subtract eq. (4) from eq. (5),

\[
\left( \frac{\partial}{\partial r} \left( \frac{\partial w}{\partial r} - \frac{\partial u}{\partial z} \right) + \frac{\partial u}{\partial r} \left( \frac{\partial w}{\partial r} - \frac{\partial u}{\partial z} \right) + \left( \frac{\partial w}{\partial z} \left( \frac{\partial w}{\partial r} - \frac{\partial u}{\partial z} \right) \right) \right.
\]

\[
+ u \frac{\partial^2 w}{\partial r^2} - u \frac{\partial^2 u}{\partial z \partial r} + w \frac{\partial^2 w}{\partial r \partial z} - w \frac{\partial^2 u}{\partial z^2} - \frac{\partial (\nabla^2 w)}{\partial z} = -\left( \frac{\partial p}{\partial r \partial z} \right) + \nabla^2 (\frac{\partial w}{\partial r} - \frac{\partial u}{\partial z}) - \nabla \left( \frac{\partial}{\partial z} \left( -\frac{u}{r^2} \right) \right) + \nabla \left( \frac{\partial}{\partial z} \left( -\frac{u}{r^2} \right) \right) + \nabla \left( \frac{1}{r^2} \frac{\partial w}{\partial z} \right) + g \alpha \frac{2}{\partial r} (T - T_0)
\]

(7)

Introducing the Naiver Stokes Stream Function

\[
\psi = \left( \frac{1}{r} \right) \frac{\partial w}{\partial z} , \quad w = -\frac{1}{r} \frac{\partial \psi}{\partial r}
\]

(8)

Now the vorticity parameter \( \omega \) is defined by

\[
\omega = \frac{\partial w}{\partial r} - \frac{\partial u}{\partial z}
\]

(9)

Substituting \( \omega \) for \( \frac{\partial w}{\partial r} - \frac{\partial u}{\partial z} + \text{Product Rule} \)

\[
\frac{\partial w}{\partial r} + \frac{\partial w}{\partial r} (\omega w) + \frac{\partial w}{\partial z} (\omega w) + \frac{2}{r} \frac{\partial w}{\partial z}
\]

\[
= \nabla^2 \omega + \frac{\omega w}{r^2}
\]
Rotational Velocity in Product Form

\[ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial r} + w \frac{\partial v}{\partial z} + \frac{uv}{r} = v \left( \nabla^2 v - \frac{v}{r^2} \right) \]

Add continuity equation:

\[ \frac{1}{r} \frac{\partial}{\partial r} (ru) + \frac{\partial w}{\partial z} = 0 \]  

or

\[ \frac{1}{r} \left( u \frac{\partial r}{\partial r} + r \frac{\partial u}{\partial r} \right) + \frac{\partial w}{\partial z} = 0 \]

Multiply both sides by \( v \)

or \( v \left( \frac{u}{r} + \frac{\partial u}{\partial r} + \frac{\partial w}{\partial z} \right) = 0 \times v \)

Now add to 2c equation 1

\[ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial r} + v \frac{\partial u}{\partial r} + w \frac{\partial v}{\partial z} + v \frac{\partial w}{\partial z} + \frac{uv}{r} + \frac{uv}{r} = v \left( \nabla^2 v - \frac{v}{r^2} \right) \]

Now product rule

\[ \frac{\partial v}{\partial t} + \frac{\partial}{\partial r} (uv) + \frac{\partial}{\partial z} (wv) + \frac{zu}{r} = v \left( \nabla^2 v - \frac{v}{r^2} \right) \]
APPENDIX 2

Svanberg Vorticity Equation

The expansion of the terms in the Svanberg Vorticity Equations was developed by W. E. Langlois [ref. 4]. A section of his work is presented here. The first section is the basic equations used to describe the terms in the development of the equation. The second section is the expansion of the equation for the off axis and on axis grid points.

2. CONSERVATIVE DIFFERENCING FOR THE ANGULAR MOMENTUM - OFF THE AXIS

Those grid cells which do not lie on the axis of symmetry are annular rings of rectangular section; a slice of a typical cell C is illustrated in Fig. 2. The center of a meridional section of C is the grid point $(r_i, z_j)$, with $r=(i-1)\Delta r$. The boundary $\partial C$ of C consists of the circular cylinders

$$R_{i+} = \{(r, z): r=r_{i+}, z_-<z<z_+\}$$

and the plane annuli

$$Z_\pm = \{(r, z): r_i^-<r<r_{i+}, z=z_\pm\}.$$ 

In these descriptions

$$r_{i+} = r_i + \Delta r/2 = (i-1 + \frac{1}{2})\Delta r$$

$$z_- = z_j - \Delta z_-/2,$$

$$z_+ = z_j + \Delta z_+/2.$$
The upward and downward \( z \)-increments may differ, because the study of Czochralski bulk flow is facilitated by using an axially stretched grid [3]. The height of the grid cell is

\[
\Delta z_j = (\Delta z_- + \Delta z_+)/2
\]

and its volume is

\[
V = \pi \Delta z_j \left( r_+^2 - r_-^2 \right) = 2\pi r_i \Delta r \Delta z_j .
\]  

(3)

We shall be concerned with the rate of change of angular momentum, and later of Svanberg vorticity, within \( C \). Rather than repeat mathematical displays, we shall define certain averages for a generic variable \( f \), which may represent \( \Omega \), \( S \), or any other quantity whose averages are of interest. We use \( f_{i,j} \) to denote the average of \( f \) over the grid cell, i.e.,

\[
V f_{i,j} = \int \int \int_C f \, dV = 2\pi \int_{r_{i-}}^{r_{i+}} \int_{z_-}^{z_+} f \, dz \, dr = 2\pi \int_{z_-}^{z_+} \int_{r_{i-}}^{r_{i+}} r f \, ddr \, dz .
\]  

(4)

At sufficiently high resolution, \( f_{i,j} \) may be regarded as an approximation to the value of \( f \) at the grid-point \((r_i, z_j)\), but it is seldom necessary to invoke this interpretation.

We also define averages over the bounding surfaces of \( C \):

\[
<f>_{R_{i+}} = \frac{1}{\Delta z_j} \int_{z_-}^{z_+} f(r_{i+}, z) \, dz ,
\]  

(5)

\[
<f>_{Z_\pm} = \frac{1}{r_i \Delta r} \int_{r_{i-}}^{r_{i+}} rf(r, z_{\pm}) \, dr .
\]

For the purpose of computing advective fluxes, we introduce special weighted averages \( <f>_{R_{i\pm}}^*, <f>_{Z_\pm}^* \) defined by the equations

\[
<w>_{R_{i\pm}} = <w>_{R_{i\pm}} , <f>_{Z_\pm}^* = <w>_{Z_\pm} <f>_{Z_\pm}^* .
\]  

(7)
3. CONSERVATIVE DIFFERENCING FOR THE ANGULAR MOMENTUM — THE AXIAL CORE

Those grid cells with \( i=1 \) are cylindrical volumes of radius \( R_{1+} = \Delta r/2 \), centered on the axis of symmetry, as shown in Fig. 3. The volume of the cell \( C_o \) centered at \((0,z_j)\) is therefore

\[
V_o = \pi(\Delta r)^2 \Delta z_j / 4
\]

and the average value \( f_{1,j} \) of the generic variable \( f \) over this cell is given by

\[
V_o f_{1,j} = \iiint_{C_o} f \, dv = 2\pi \int_0^{\Delta r/2} r \int_{z_-}^{z_+} f \, dz \, dr = 2\pi \int_{z_-}^{z_+} \int_0^{\Delta r/2} r f \, dr \, dz.
\]

The boundary \( \partial C_o \) of \( C_o \) consists of the cylinder \( R_{1+} \) and the disks

\[
z^0_{\pm} = \{(r,z): 0 < r < \Delta r/2, \ z = z_{\pm}\}.
\]

Averages and flux-weighted averages over these disks are defined by

\[
<f>_{z^0} = \frac{8}{(\Delta r)^2} \int_0^{\Delta r/2} r f(r,z_{\pm}) dr,
\]

\[
<wf>_{z^0} = <w>_{z^0} <f>_{z^0}^*.
\]
resemblance between the advective and viscous terms in (31) to their counterparts in (2) suggests in advance that this should be so.

Equation (31) has two distinctive terms, the coupling \((\partial/\partial z)(\Omega^2/r^4)\) and the bouyancy \((\alpha g/r)\partial T/\partial r\), but these present no special difficulty.

Thus our procedure will be to average (31) over \(C\), obtaining a prognostic equation for advancing \(S_{i,j}\). Extending the notation of Section 2 in an obvious fashion, we write

\[
\frac{\Delta S_{i,j}}{\Delta t} = \frac{(\Delta S_{i,j})_{ra}}{\Delta t} + \frac{(\Delta S_{i,j})_{za}}{\Delta t} + \frac{(\Delta S_{i,j})_{c}}{\Delta t} + \frac{(\Delta S_{i,j})_{b}}{\Delta t} + \frac{(\Delta S_{i,j})_{rv}}{\Delta t} + \frac{(\Delta S_{i,j})_{zv}}{\Delta t}.
\]

We now integrate (31) term by term, using either of the iterated forms in (21) as appropriate. First,

\[
y \frac{(\Delta S_{i,j})_{ra}}{\Delta t} = - \int_{C} \frac{1}{r} \frac{\partial}{\partial r} (ruS) dV = 2\pi \int_{z_+}^{z_-} [ruS]_{r_i^+}^{r_i^-} dz.
\]

With (3), (5) and (7), therefore,

\[
\frac{(\Delta S_{i,j})_{ra}}{\Delta t} = \frac{1}{r_i \Delta r} \left( r_j <u>_{R_i^-}^{R_i^+} - r_i <u>_{R_i^-}^{R_i^+} <S>_{r_i^-}^{r_i^+} \right). \tag{32}
\]

Next,

\[
y \frac{(\Delta S_{i,j})_{za}}{\Delta t} = - \int_{C} \frac{\partial}{\partial z} (wS) dV = 2\pi \int_{z_+}^{z_-} [wS]_{z_i^-}^{z_i^+} dr.
\]

Equations (3), (6) and (7) then yield

\[
\frac{(\Delta S_{i,j})_{za}}{\Delta t} = \frac{1}{\Delta z_j} \left( <w>_{z_+}^{z_+} - <w>_{z_-}^{z_-} - <S>_{z_+}^{z_+} - <S>_{z_-}^{z_-} \right).
\]
Applying one or the other of these procedures to each of the remaining terms in (31) yields

\[
\frac{(\Delta S_{i,j})_c}{\Delta t} = \frac{1}{\Delta z_j} \left[ \left\langle \left( \frac{\Omega}{r} \right)^2 \right\rangle_{z^-} - \left\langle \left( \frac{\Omega}{r} \right)^2 \right\rangle_{z^+} \right],
\]

(34)

\[
\frac{(\Delta S_{i,j})_b}{\Delta t} = \frac{\alpha g}{r_i \Delta r} \left( \left\langle \frac{1}{r} \right\rangle_{R_i^+} - \left\langle \frac{1}{r} \right\rangle_{R_i^-} \right),
\]

(35)

\[
\frac{(\Delta S_{i,j})_{rv}}{\Delta t} = \frac{\nu}{r_i \Delta r} \left[ \left\langle \frac{1}{r} \frac{\partial}{\partial r} (r^2 S) \right\rangle_{R_i^+} - \left\langle \frac{1}{r} \frac{\partial}{\partial r} (r^2 S) \right\rangle_{R_i^-} \right],
\]

(36)

\[
\frac{(\Delta S_{i,j})_{zv}}{\Delta t} = \frac{\nu}{\Delta z_j} \left( \left\langle \frac{\partial S}{\partial z} \right\rangle_{z^+} - \left\langle \frac{\partial S}{\partial z} \right\rangle_{z^-} \right).
\]

(37)

Each of the forms (32) through (37) is "conservative". That is, if \( \partial C \) lies entirely within the fluid and off the axis of symmetry, any increase in \( VS_{i,j} \) is exactly offset by a decrease in the total Svanberg vorticity of the neighboring grid cells. This applies even to the coupling and buoyancy terms: away from boundaries and the axis, these terms redistribute Svanberg vorticity rather than create it.

5. CONSERVATIVE DIFFERENCING FOR THE SVANBERG VORTICITY — THE AXIAL CORE

Using (18) to integrate (31) over the cylindrical volume \( C_0 \) forces us to deal with indeterminate forms, since one of the integrations begins at \( r=0 \). The orders of some relevant quantities as \( r \to 0 \) were established near the end of Section 2, but a few more are needed. With (14), (15), (29) and the definition of \( S \) as \( \omega/r \), we obtain

\[
S = S(0,z) + O(r)
\]
where
\[ S(0,z) = 2W''_0 - 3U'_0/\partial z. \]

When we integrate the viscous term in (31), it will be necessary to note that
\[ \frac{1}{r} \frac{\partial}{\partial r} (r^2 S) = 2S(0,z) + O(r). \]  \hspace{1cm} (39)

Since thermal diffusivity will not permit a conical singularity in the temperature field,
\[ T = T_0 + T''_0 r^2 + O(r^3). \]  \hspace{1cm} (40)

For any quantity \( f \) which does not vanish on the axis of symmetry, we define
\[ \langle f \rangle_0 = \frac{1}{\Delta z} \int_{z_-}^{z_+} f(0,z) \, dz, \]  \hspace{1cm} (41)
which makes available the approximation
\[ f_{1,j} \approx \langle f \rangle_0. \]

We now repeat the analysis of the preceding section, with appropriate changes to reflect that we are dealing with the cylindrical cell \( C_0 \) rather than the annular \( C \). First,
\[ V_o \frac{(\Delta S_{1,j}) \, ra}{\Delta t} = 2\pi \int_{z_-}^{z_+} [ruS]^0_{\Delta r/2} \, dz. \]  \hspace{1cm} (42)

With (5), (7), (17), (38) and (40), therefore,
\[ \frac{(\Delta S_{1,j}) \, ra}{\Delta t} = -\frac{4}{\Delta r} \langle u \rangle_{R_{1+}} \langle S \rangle^*_{R_{1+}}. \]
4. CONSERVATIVE DIFFERENCING FOR THE SVANBERG VORTICITY — OFF THE AXIS

If the vorticity $\omega$ is defined in terms of the meridional velocity components according to

$$\omega = \partial w/\partial r - \partial u/\partial z,$$

then $\omega$ satisfies the equation

$$\frac{\partial \omega}{\partial t} + \frac{1}{r} \frac{\partial}{\partial r} (ru \omega) + \frac{\partial}{\partial z} (w \omega) + \frac{2v}{r} \frac{\partial v}{\partial z} = \alpha \frac{\partial T}{\partial r} + \nu \nu^2 \omega - \nu \frac{\omega}{r^2};$$

(30)

the buoyancy term involves new symbols, viz., the volumetric expansion coefficient $\alpha$, the gravitational acceleration $g$, and the local temperature $T$. From (30) one may derive a prognostic equation governing the Svanberg vorticity $S=\omega/r$:

$$\frac{\partial S}{\partial t} + \frac{1}{r} \frac{\partial}{\partial r} (ruS) + \frac{\partial}{\partial z} (wS) + \frac{\partial}{\partial z} \left( \frac{\Omega^2}{r^4} \right)$$

$$= \frac{\alpha g}{r} \frac{\partial T}{\partial r} + \nu \frac{\partial}{\partial r} \left[ \frac{1}{r} \frac{\partial}{\partial r} (r^2 S) \right] + \nu \frac{\partial^2 S}{\partial z^2}. \quad (31)$$

In Section 2, we derived procedures for advancing $\Omega_{i,j}$ by consideration of angular momentum interchange between grid cells. Ultimately one must make estimates of various averages over the bounding surfaces of the cells, but the analysis itself dealt with finite quantities rather than infinitesimals. No analogous procedure applies here, because the Svanberg vorticity is defined differentially, i.e., locally. We must derive the differencing procedures from (31): there is no other route.

Fortunately - and this is the advantage of using $S$ rather than $\omega$ - Eq. (31) is of a form which is easily integrated over the grid-cell $C$. The
Next,

\[ V_0 \frac{(\Delta S_{1,j})_{za}}{\Delta t} = 2\pi \int_0^{\Delta r/2} r[wS]_{z+} dr. \]

Equations (17), (19) and (20) then yield

\[ \frac{(\Delta S_{1,j})_{za}}{\Delta t} = \frac{1}{\Delta z_j} \left( \begin{array}{c} \langle \omega \rangle_{z_0} \langle S \rangle_{z_0} * \langle \omega \rangle_{z_0} \langle S \rangle_{z_0} * \\ z_0^- \end{array} \right). \]

Applying this same calculation to the coupling term yields

\[ \frac{(\Delta S_{1,j})_c}{\Delta t} = \frac{1}{\Delta z_j} \left[ \frac{\langle \Omega \rangle_{z _0}}{r^2} - \frac{\langle \Omega \rangle_{r^2}}{z_0^0} \right], \]

which presents the problem of estimating \( \left\langle \left( \Omega / r^2 \right)^2 \right\rangle_{z_0^0}. \) These quantities, which are defined on grid-cell boundaries, must be estimated from analogous quantities defined for the cells themselves, viz.,

\[ \left\langle \left( \frac{\Omega}{r^2} \right)^2 \right\rangle_{1,j} = \frac{8}{(\Delta r)^2} \int_0^{\Delta r/2} r \left( \frac{\Omega(r,z_j)}{r^2} \right)^2 dr. \] (43)

For example, if centered estimates are used,

\[ \left\langle \left( \frac{\Omega}{r^2} \right)^2 \right\rangle_{z_0^0} = \frac{1}{2} \left[ \left\langle \left( \frac{\Omega}{r^2} \right)^2 \right\rangle_{1,j} + \left\langle \left( \frac{\Omega}{r^2} \right)^2 \right\rangle_{1,j+1} \right]. \]

It is consistent with (22) to take

\[ \left\langle \left( \frac{\Omega}{r^2} \right)^2 \right\rangle_{1,j} = \frac{64 \Omega^2_{1,j}}{(\Delta r)^4} = \frac{16 \Omega^2_{2,j}}{25(\Delta r)^4}. \]

The higher order procedure which led to (26) should be accompanied by a more complicated form here. Carrying out the integration (43) with the quadratic and quartic terms both retained in \( \Omega \) leads to

\[ \left\langle \left( \frac{\Omega}{r^2} \right)^2 \right\rangle_{1,j} = \frac{64 \Omega^2_{1,j}}{(\Delta r)^4} + \frac{8}{(9\Delta r)^4} (\Omega_{2,j-10\Omega_{1,j}})(11\Omega_{2,j-1163\Omega_{1,j}}). \]
Applying an integration similar to (42) to the buoyancy term in (31) yields

\[
\frac{(\Delta S_1)_b}{\Delta t} = \frac{8\alpha g}{(\Delta r)^2} \left[ <T>_{R_1+} - <T>_{0} \right].
\] (44)

The same technique, applied to the radial viscous term, involves the indeterminate form resolved by (39). We have

\[
\frac{(\Delta S_1)_r}{\Delta t} = \frac{8\nu}{(\Delta r)^2} \left\{ \left( \frac{1}{r} \frac{\partial}{\partial r} (r^2 \nu) \right)_{R_1+} - \frac{1}{\Delta z_j} \int_{z_-}^{z_+} \left( \lim_{r \to 0} \left[ \frac{1}{r} \frac{\partial}{\partial r} (r^2 \nu) \right] \right) dz \right\}.
\]

Using (39) and (41) to evaluate the second term, we obtain

\[
\frac{(\Delta S_1)_r}{\Delta t} = \frac{8\nu}{(\Delta r)^2} \left[ \left( \frac{1}{r} \frac{\partial}{\partial r} (r^2 \nu) \right)_{R_1+} - 2 <S>_{0} \right].
\] (45)

The first term, together with its counterpart in (36), represents an exchange of Svanberg vorticity between \( C_0 \) and its annular neighbor. As in Section 3, the factor of eight reflects the volume-ratio of these two cells.

The presence of the second term in (45) implies that, in a viscous fluid, Svanberg vorticity is created or destroyed at the axis of symmetry. If \( S \) were a physically conserved quantity, this would be a disturbing result because, as far as the three-dimensional flow field is concerned, \( x=y=0 \) is just one more vertical line. One would expect physically conserved quantities to enter the flow field only at boundaries, or perhaps where external force fields act as sources, as in (44).

The remaining term in (31) entails no difficulty. We obtain

\[
\frac{(\Delta S_1)_z}{\Delta t} = \frac{\nu}{\Delta z_j} \left( \frac{\partial S}{\partial z} \right)_{z_+}^{0} - \left( \frac{\partial S}{\partial z} \right)_{z_0}^{0}.
\]
APPENDIX 3

VIEWFACTORS

A sample of the viewfactor evaluation was given in the text, the surface-to-shield viewfactor. There were 6 more viewfactors, 1) surface-to-wall, 2) shield-to-crystal, 3) shield-to-wall, 4) shield-to-crystal, 5) shield-to-furnace top and 6) crystal-to-furnace top to be calculated. Once one viewfactor from a given source to one surface of a enclose volume, the other viewfactor was calculated by the

\[ F_{1-2} + F_{1-3} = 1 \]

or

\[ F_{1-3} = 1 - F_{1-2} \]

To verify the use of CONFAC II was done correct, the surface to shield and surface to wall viewfactor were calculated and equation 1 was verified. To obtain the viewfactor in the opposite direction, the reciprocity relations were used,

\[ A_1 F_{1-2} = A_2 F_{2-1} \]

The surface-to-wall viewfactor was calculated based on projecting a line from the surface to the shield and determining where, on the wall it hits. This is done for a point at the lower
and upper edge of the shield. To simplify the equation the z-coordinate was taken as the same as the shield value. The x and and y components were calculated in the following sequence.

1. Determine equation of the line that passes through the surface element and the tangent point on the shield line XY (fig. 1)

\[
y = m x + b \\
\text{where } m = \frac{y_2 - y_1}{x_2 - x_1} \\
b = y_1 - m x_1
\]

2. Determine the equation for the wall

\[
x^2 + y^2 = (\text{Rad}_{\text{wall}})^2
\]

3. Substitute equation 3 into 4 and solving for x using quadric equation.

4. Then substitute the x value into equation 3 to obtain y.

5. The wall is defined by a set of points calculated between points A-B-C-D-A and using equal spacing determined by dividing the arc.
FIGURE 1 - SURFACE-TO-WALL VIEWFACTOR

FURNACE VIEWS
a. Calculate the angle \( \theta = \tan^{-1} \frac{y_1}{x_1} \)
b. Divide by the number of division desired (n)
c. Calculate the points by
   \[
   X = R_{\text{shield}} \sin (n-1) \left( \frac{\theta}{n} \right)
   \]
   \[
   Y = R_{\text{shield}} \cos (n-1) \left( \frac{\theta}{n} \right)
   \]

The next viewfactor is the shield-to-wall value. This was calculated in a slightly different manner. In the first case, the viewfactor was calculated from a point to surface. In this case, the viewfactor was between two surfaces. The difference will be that an average value from several areas of the shield has to be determined as seen by the shaded area in fig. 1 and using the following procedure.

1. Calculate wall intersection. These are the points that are intersected by the tangent plane on the shield as seen in figure 2.
   a. equation of cone
   \[
   x^2 + y^2 - (C_1 z_1 - b)^2 = 0, \quad C_1 = \text{slope}
   \]
   b. Line on cone \( y = 0 \)
   \[
   x_1 = C_1 z_1 + b
   \]
   c. Z value for given X
   \[
   z = (x - b)/C_1
   \]
   d. Note that the value of \( x \) and \( z \) are the same on the wall as they are on the shield
   e. \( y = (R_{\text{wall}}^2 - x^2) \)
FIGURE 2   SHIELD-TO-WALL VIEWFACTOR
2. Pick value of \( N \) to divide the area of the shield in equal areas. See figure Xc
   
   a. Areas shield

   \[
   A = 2 \left( 1 + C_1^2 \right) \left( \frac{C_1}{2} y^2 + b y \right) y \bigg|_0^\text{Shield Height}
   \]

   b. Divide area by the number of and points to find section area

   \[
   A_{\text{sec}} = \frac{A}{N}
   \]

   c. 

   \[
   X_i = \left( \frac{\text{Sh}_{\text{out}} - \text{Sh}_{\text{in}}}{N} \right) + X_{i-1} \bigg|^{1/2}
   \]

   where \( X_i = \text{Sh}_{\text{in}} \)

   d. Once \( X_i \) are determined the center of the grid are calculated

   \[
   X_{i_{\text{center}}} = \frac{(X_N + X_{N-1})}{2}
   \]

3. The values of the wall points and shield points are sent to the Confac subroutine to calculate viewfactor.
The crystal-to-shield viewfactor was obtained by taking a point half way up the crystal and using this as the average viewfactor for the crystal as seen in figure 3, the same basic approach is taken. $\phi_1$ and $\phi_2$ are obtained by the geometry. Then the location of the shield that was "seen" by the element is then obtained.

Once $V.F_{su-sh}$, $V.F_{su-wa}$, $V.F_{sh-wa}$, $V.F_{cryr-sh}$, were calculated, their counter parts can also be calculated by

$$A_1 F_{1-2} = A_2 F_{2-1}$$

for example

$$V.F_{wa-cryr} = \frac{A_{cryr}}{A_{wa}} \cdot F_{cryr}$$

The last three viewfactors are obtained by equation 1. The values of the viewfactor obtained are listed in Appendix 6, Numerical Results.
CRYSTAL

CRUCIBLE WALL

SURFACE ELEMENTS

AREA "SEEN" BY SURFACE ELEMENT

FIGURE 3 CRYSTAL-TO-SURFACE VIEWFACTOR
APPENDIX 4

COMPUTER PROGRAM FLOW CHART

DIMENSION ALL ARRAYS

INITIALIZE MELT PARAMETERS

LOCATION OF GRID POINTS

INITIALIZE MELT VARIABLES
OR USE VALUES FROM PREVIOUS RUN

SET UP A MATRIX FOR SOLVING STREAM FUNCTION

EIGENVALUE SUBROUTINE

SOLVE FOR Δ S
ON AXIS
OFF AXIS

CALCULATE Ω

CALCULATE Ψ
CALCULATE VELOCITIES
AXIAL
RADIAL

YES
SECOND TIME THRU

NO
CALCULATE NEW $\omega$ (W CHECK) FROM VELOCITIES & AVERAGE

CALCULATE MELT TEMPERATURE

SOLVE FOR $\Delta S$

CALCULATE SURFACE TEMPERATURE

VIEWFACTORS

$\Delta T$ RADIATION

SIMULTANEOUS EQUATION

CALCULATE ROTATIONAL VELOCITY

CHECK STEADY STATE

NO

YES
PRINT OUTPUT
APPENDIX 5

COMPUTER PROGRAM
C RADIUS, Z AXIAL
C DIMENSION COORDINATE SYSTEM
DIMENSION R(20), Z(20)
DIMENSION DELZ(20), DELZPL(20), DELZMI(20), RPLUS(20), RMINUS(20)
C DIMENSIONALIZE TEMPERATURE T1=TEMP OLD T2=TEMP UPDATE
DIMENSION T1(20,20), T2(20,20)
C DIMENSIONALIZE VELOCITIES U RADIAL, V1,V2= ROTATION, W AXIAL
DIMENSION U(20,20), W(20,20)
C DIMENSIONALIZE STREAM FUNCTION SHI, VORTICITY WW, SVANBERG VORTICITY S
DIMENSION SHI(20,20), WW(20,20), SHIBAR(20,20)
DIMENSION S(20,20), DD(20,20)
C DIMENSION COEFFICIENTS OF SHI MATRIX UU,D,M,VV, E, F
DIMENSION UU(20,20), D(20), M(20), VV(20,20), E(20,20), F(20,20)
C DIMENSION TRANSFORMATION MATRICES P Q
DIMENSION P(18,18), Q(18,18)
DIMENSION DELT(20,20)
C DIMENSION OF CC MATRIX
DIMENSION X(18), Y(18), CC(18)
C DIMENSION OF W COMPONENTS
DIMENSION PARWR(20,20), PARUJ(20,20), FPRIM(20)
DIMENSION WCHECK(20,20), WERROR(20,20), ERMAY(3000)
DIMENSION DELRAD(20), T2NSH(20)
DIMENSION BPP(400), BPP2(400)
DIMENSION EB1(20), JS(20), EBS(19)
DIMENSION DELTI(3000), DELTIW(3000)
DIMENSION WR(120), EB(153), DDE(800), SN(20,20)
DIMENSION AR(20), BR(20), RBR(20)
DIMENSION ZAFF(20), ZAFFM(20)
DIMENSION SOA(20), S0P(20), S0M(20)
DIMENSION RFP(20), RFM(20)
DIMENSION OMEGA(20,20)
DIMENSION JW1(20), JW5(20), JW6(20)
DIMENSION ETT(20), RAR(19,19), RQOS(16,16), RIEFF(20)
INTEGER WKS, SIGMA, PF
REAL JS, P, Q, PINV, NHE, MD
REAL RPAR, PARR, PARRUT, PARTZ
REAL KV, JW1, JW5, JW6
ERMAX(1)=100,
M=0
M=1
II=16
II=20
JJ=20
JJM2=JJ=2
II=16
III=105
II=106
NS=104
JJ=6
C READ IN PREVIOUS VALUES
KSAV=1
DATSAV=1
NOSH=5
IF (KSAV,NE,1) GO TO 88
READ (III,700) ((T1(I,J),J=1,II),I=1,II)
READ (III,701) ((V1(I,J),J=1,II),I=1,II)
READ (III,701) ((Q(I,J),I=1,IIM2),J=1,JJM2)
READ (III,701) ((X(I),I=1,IIM2)
88 CONTINUE
  IF (DATAB,NE.1) GO TO 99
  READ (III,701) ((W(I,J),I=1,II),J=1,JJ)
  READ (III,701) ((U(I,J),I=1,II),J=1,JJ)
  READ (III,701) ((S(I,J),I=1,II),J=1,JJ)
  READ (III,701) ((W(I,J),I=1,II),J=1,JJ)
  IF (NOBS,NE.5) GO TO 588
  READ (NL,700) (T2NSH(I),I=1,II)
588 CONTINUE
  DO 99 I=1,II
  DO 99 J=1,JJ
  T2(I,J)=T1(I,J)
  V2(I,J)=V1(I,J)
99 CONTINUE
  IIM1=II-1
  JJM1=JJ-1
  LL=0.
  KK=1
  M=0.
  INT=25
C KP=NUMBER OF ITERATIONS
  KP=2000
  KPRINT=2
  NNN=0
  NN=2
  DELTIM=.08
C MELT DENSITY (GM/CM**3)
  RHO= 2.33
C SPECIFIC HEAT CP (CAL/GM-K)
  CP=0.233
C KINEMATIC VISCOSITY KV (CM**2/SEC)
  KV=.8
C THERMAL DIFFUSIVITY TD (CM**2/SEC)
  TD=0.125
  TD=1.93
C VOLUMETRIC EXPANSION COEFFICIENT VEC (1/K)
  VEC=0.0000141
C SURFACE EMISSIVITY EE
  EE=0.318
C SURFACE TENSION TEMPERATURE COEFFICIENT STTC (DYNES/(CM*K))
  STTC=0.43
C CRUCIPLE RADIUS CR (CM)
  CR= 11.55
C CRYSTAL RADIUS CRYR=3.75 (CM)
  CRYR=3.75
C CRYSTAL HEIGHT
  CRYH=20.0
C MELT DEPTH
  MD=10.91
C CRUCIPLE ROTATION RATE (RADIAN8/SEC)
  CRUROR=1.57
C CRYSTAL ROTATION RATE (RADIAN8/SEC)
  CRYROR=-2.31
C CRUCIPLE TEMPERATURE CRUTEMP (K)
  CRUTEM =1773.0
C CRYSTAL TEMPERATURE CRYTEMP (K)
13  DELZPL(J)=(Z(J+1)-Z(J))
    DELZMI(J+1)= (Z(J+1)-Z(J))
11  CONTINUE
    DELZ(1)=(Z(2)+Z(1))/2
    DELZ(JJ)=(Z(JJ+1)-Z(JJ))/2,
DO 135 J=2,JMJ
    ZAFP(J)=DELZMI(J)/DELZ(J)
    ZAFM(J)=DELZPL(J)/DELZ(J)
135  CONTINUE
    IF (KSAV,EQ.1) GO TO 35
C INITIALIZE MELT PARAMETERS S,SHI,W,V,OMEGA,T
    DO 35 I=1,II
    DO 35 J=1,JJ
    S(I,J)=0.0
    SHI(I,J)=0.0
    W(I,J)=0.0
    U(I,J)=0.0
    RA=(CR=R(I))/CR
    RB=Z(J)/MD
    PC=R(I)/CR
    RD=(MD=Z(J))/MD
    VI(I,J)=((RA+RB)*CRYROR*R(3)+(RC+RD)*CRURO*R(I))/2.
    VI(I,J)=VI(I,J)/(2*PI)
    TI(I,J)=((RA+RB)*CRYTEM+(RC+RD)*TRUETEM)/2,
    IF(J,NE,1) GO TO 40
    TI(I,J)=CRYTEM
    VI(I,J)=R(I)*CRYROR/(2*PI)
    GO TO 34
40  IF(J,NE, JJ) GO TO 44
    IF (I LE PP) GO TO 44
    TI(I,J)=CRYTEM
    VI(I,J)=R(I)*CRYROR/(2*PI)
    GO TO 34
44  IF(I,NE,1) GO TO 45
    VI(I,J)=0.0
    GO TO 34
45  IF (I, NE, II) GO TO 34
    TI(I,J)=CRUETEM
    VI(I,J)=CR*CRURO/(2*PI)
34  V2(I,J)=VI(I,J)
T2(I,J)=TI(I,J)
35  CONTINUE
C SET UP SYMMETRIC TRIDIAGONAL MATRIX CC
    DO 19 J=1,III
    X(J)=2.0/(1.0-1.0/(4.0+J**2))/DELR**2
    Y(J)=10.
    IF (J, EQ, II=2) GO TO 19
    Y(J+1)=-(1.0-(2.0+J+1)**2)*0.5/(DELR**2)
19  CONTINUE
    WRITE (JJJ,811)
    WRITE (JJJ,78) (R(I),I=1,II)
    WRITE (JJJ,812)
    WRITE (JJJ,74) (Z(J),J=1,JJ)
    WRITE (JJJ,72) (PLUS(I),I=1,II),(PMINUS(I),I=1,II)
    WRITE (JJJ,72) (DELZ(J),J=1,JJ), (DELZPL(J),J=1,JJ)
    WRITE (JJJ,74) (DELZMI(J),J=1,JJ)
    IF(KSAV,EQ.18) GO TO 155
C INITIALIZING Q MATRIX TO THE IDENTITY MATRIX
DO 15 I=1,IIM2
DO 15 J=1,IIM2
IF(T=J) 16,17,16
16 Q(I,J)=0.0
GO TO 15
17 Q(I,J)=1.0
15 CONTINUE
C CALCULATION OF P AND P=INVERSE MATRIX
CALL TOL2 (IIM2,IIM2,X,Y,O,IERR)
155 CONTINUE
DO 18 I=1,IIM2
DO 18 J=1,IIM2
P(I,J)=(( I**0.5)*Q(I,J))
18 CONTINUE
WRITE (JJJ,814)
WRITE (JJJ,222) (X(J),J=1,IIM2)
WRITE (JJJ,813)
WRITE (JJJ,223) (( Q(I,J),I=1,IIM2),J=1,IIM2)
WRITE (JJJ,802)
WRITE (JJJ,805) (( T1(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,804)
WRITE (JJJ,801)((V1 (I,J),I=1,II),J=1,JJ)
C COEFFICIENTS OF THE TRIDIAGONAL STREAM FUNCTION
DO 22 SIGMA=2,JJM1
D(SIGMA)=1./((DELZ(SIGMA)*DELZMI(SIGMA))
UU(SIGMA)=1./((DELZ(SIGMA)*DELZPL(SIGMA))
M(SIGMA)=2./((DELZPL(SIGMA)*DELZMI(SIGMA))
22 CONTINUE
C COEFFICIENTS OF GAUSSIAN REDUCTION
DO 23 SIGMA=2,JJM1
DO 23 L=1,IIM2
E(L,1)= 0.0
F(L,1)= 0.0
DD(L,SIGMA)=1./((X(L)+M(SIGMA)+D(SIGMA)*E(L,SIGMA=1))
E(L,SIGMA)= UU(SIGMA)*DD(L,SIGMA)
23 CONTINUE
IF (KSAV .EQ. 1) GO TO 123
IF(KK .EQ. 1) GO TO 112
C CALCULATION OF DELTA S2(I,J)
111 CONTINUE
DO 59 I=1,II
DO 59 J=1,II
DELS(I,J)=DELS(I,J)/DELTIM
59 CONTINUE
WRITE (JJJ,663) (SUMT2)
WRITE (JJJ,699) (DTMAX)
WRITE (JJJ,830) (KE, ERMAX(KE), KE=1, KK, INT)
WRITE (JJJ,803)
WRITE (JJJ,801) ((WW(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,910)
WRITE (JJJ,817) ((DELS(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,800)
WRITE (JJJ,827) ((SHI(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,810)
WRITE (JJJ,801) ((S(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,806)
WRITE (JJJ,801) (( U (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,808)
WRITE (JJJ,801) (( W (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,807)
WRITE (JJJ,801) ((WCHECK(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,809)
WRITE (JJJ,801) ((قدر I,J),I=1,II),J=1,JJ)
WRITE (JJJ,801)
WRITE (JJJ,827) ((DELT(I,J),I=1,II),J=1,JJ)
IF (KK,EQ,2) GO TO 117
WRITE (JJJ,982) ((I,DELRAD(I),DELT(I,JJ)),I=MPP,II)
WRITE (JJJ,498)
WRITE (JJJ,672) ((I,EB1(I),EB2(I),JS(I)),I=1,IMP2+5)
WRITE (JJJ,74) (SOA(J),J=1,JJ)
117 CONTINUE
WRITE (JJJ,805) (T2NSH(I),I=1,II)
WRITE (JJJ,802)
WRITE (JJJ,805) (I T2 (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,804)
WRITE (JJJ,827) ((V1(I,J),I=1,II),J=1,JJ)
IF (LCL,LE,100) GO TO 112
118 CONTINUE
C CALCULATION OF NEW DELTIME
DDE(I)=1000
DELTIM=15
PZ=1
DO 201 I=2,II+1
DO 201 J=2,J-J1
AA=2*TD*(1/RERL**2+1/DELTZ)**2
BB=1*8*(U(I+1,J)+1+U(I,J))*U(I+1,J)+U(I,J))= 4(R(I)+R(I+1))*(U(I,J)+U(I,J+1))/R(I)*DELTZ)
CI=1/4*(W(I,J+1)=W(I,J-1))/DELTZ(J)
PZ=PZ+1
DDE(PZ)=1/(AA-BB*CI)/2
IF(DDE(PZ),LT, DELTIM) DELTIM=DDE(PZ)
IF(DDE(PZ),NE,DELTIM) GO TO 201
IA=I
J=J
201 CONTINUE
M=II+1
DELTI(M)+DELTIM
C CALCULATION OF DELTIM FOR ROTATION
DDE(I)=10
PD=0
DELTIM=10
DO 301 I=2,II+1
DO 301 J=2,J II+1
DW=KV*(2/DELZ**2+2/DELZ)**2
EW=(U(I+1,J)-U(I,J))/4*DELTZ)
FW=(W(I,J+1)+(4*DELZPL(J))=W(I,J+1)/(4*DELZ(J))
GW=KV/R(I)
HW=2U(I,J)/R(I)
PD=PD+1
DDE(PD)=1/(2*(DW+EW+FW+GW+HW))
IF (DDE(PD),LT, DELTIM) DELTIM=DDE(PD)
IF (DDE(PD),NE,DELTIM) GO TO 301
IA=I
J=J
301 CONTINUE
MW=MW+1
DELTIM(MW)=DELTIM
IF(DELTIM(MW),LT,DELTIM(MI)) GO TO 302
RAD, FOR 9  19-JAN-1984 09:14  PAGE 7

DELTIM=DELTI(MI)
GO TO 303
302 DELTIM=DELTIW(MW)
303 CONTINUE
   IF(DELTIM.LT.0) DELTIM=0.03
   IF(DELTIM.GT.0.009) DELTIM=0.009
5113 CONTINUE
   DO 55 J=2,JJM1
      DO 55 I=1,II1
      SZP=(S(I,J+1)+S(I,J))/2
      SIZM=(S(I,J-1)+S(I,J))/2
      SIRP=(S(I+1,J)+S(I,J))/2
      SIRM=(S(I-1,J)+S(I,J))/2
      IF (I.EQ.1) GO TO 60
   C CALCULATION OF S (RADIAL ADVECTION)
      URP=1.0/(4.0*DELTZ(J))*SHI(I,J)+SHI(I+1,J)*SHI(I+1,J+1)=
               6SHI(I+1,J-I)
      URM= 1.0/(4.0*DELTZ(J))* (SHI(I,J)+SHI(I,J+1)+ SHI(I+1,J+1)=
               5SHI(I+1,J-I)
      SRMZP= (S(I,J+1)+ S(I,J+1))/2,
      SRMzn= (S(I,J-1)+ S(I,J-1))/2,
      SRPZP= (S(I+1,J)+ S(I+1,J))/2,
      SRezn= (S(I+1,J)+ S(I+1,J+1))/2,
      SRM= (SRMZP+SRMZN)/2*ZAFM(J)/2
      SRP= (SRPZP+SRMZN)/2*ZAFM(J)/2
      DELSRA=1./(R(I)*DELR)*URM*SRM=URP*SRP
   C
   C CALCULATION OF DELTA S (AXIAL ADVECTION)
      WZM=1.0/(4.0*R(I)*DELR)*((SHI(I,J+1)+SHI(I+1,J))*RFP(I)=
               7(SHI(I,J+1)+SHI(I,J+1))*RFP(I))
      WZP=1.0/(4.0*R(I)*DELR)*((SHI(I,J+1)+SHI(I+1,J))*RFP(I)=
               8(SHI(I,J+1)+SHI(I,J+1))*RFP(I))
      SZMRP=(S(I+1,J)+S(I,J+1))/2*RFP(I)
      SZMzn=(S(I+1,J)+S(I,J-1))/2*RFP(I)
      SZPRP=(S(I+1,J)+S(I+1,J))/2*RFP(I)
      SZA=(SZMRP+SZMZN)/2*ZAFM(J)
      SAP=(SZPRP+SZPRM)/2*ZAFM(J)
      DELSA=1./DELR(J)*(WZM*SZM=WZP*SZP)
   C
   C CALCULATION OF DELTA S (ROTATIONAL)
      OMP0=(OMEGA(I,J+1)+OMEGA(I+1,J)+OMEGA(I,J)+OMEGA(I+1,J))/4
      OMPN=(OMEGA(I,J+1)+OMEGA(I+1,J)+OMEGA(I,J)+OMEGA(I+1,J))/4
      OMP0=(OMEGA(I+1,J)+OMEGA(I,J)+OMEGA(I,J)+OMEGA(I+1,J))/4
      OMMN=(OMEGA(I+1,J)+OMEGA(I,J)+OMEGA(I,J)+OMEGA(I+1,J))/4
      OMP2=(OMPP**2/R(I)+R(I)**2)*4+OMP**2*(R(I)+R(I)**2)/2
      OMMN=(OMPP**2/R(I)+R(I)**2)*4+OMM**2*(R(I)+R(I)**2)/2
      DELSC=1./DELR(J)*(OM2M=OM2P)
   C
   C CALCULATION OF DELTA S (BOUYANCY)
      DELSB=VECG/(R(I)*DELR)*(T1(I,J+1)+T1(I,J+1)+
               4T1(I,J-1)=T1(I,J-1))/4
   C
   C CALCULATION OF DELTA S (VISCOUS RADIAL)
      PAR2P=((R(I+1)+R(I))/2)*2*(SPZP*SPR)/2=
               5((R(I+1)+R(I))/2)*2*(SPZP*SPR)/2
      PAR2M=((R(I+1)+R(I))/2)*2*(SPMZP*SPRM)/2=
               7((R(I+1)+R(I))/2)*2*(SPMZP*SPRN)/2
PPRP=1/RPLUS(I)*PAR2P/DELR
PPRP=1/RMINUS(I)*PAR2M/DELR
DELSRV=KV/(R(I)*PPRM)
PAP2P=((RI+1)**2*(SI+1,J+1)+2*(SI+1,J)+SI(I,J-1))/4=
2*(RI)**2*(SI+1,J)+SI(I,J-1))/4)/DELR
PAP2M=((RI)**2*(SI+1,J+1)+2*(SI+1,J)+SI(I,J-1))/4=
3*(RI)**2*(SI+1,J+1)+2*(SI+1,J)+SI(I,J-1))/4)/DELR
DELSRV=KV/(R(I)*DELR)*(PAP2P=PAP2M)

C CALCULATION OF DELTA S (VISCOUS AXIAL)
PAP2P=((SI+1,J+1)+2*(SI+1,J)+SI(I,J-1))/4=
4*(SI+1,J)+2*(SI+1,J)+SI(I,J-1))/4)/DELRPL(J)
PAP2M=((SI+1,J)+2*(SI+1,J)+SI(I,J-1))/4=
5*(SI+1,J)+2*(SI+1,J)+SI(I,J-1))/4)/DELRMI(J)
DELSRV=KV/DELRPL(J)*(PA2P=PAP2M)
GO TO 50

60 CONTINUE
PARUZP=U(I,J+1)=U(I,J))/DELR(J)
PARUZM=U(I,J)=U(I,J+1))/DELR(J)
PAR2WP=2*(W(I,J)+W(I,J+1))/2=2*(W(I,J)+
4*(I,J))/2)/2*DELR**2
PAR2W=((I,J)+W(I,J))/2=2*(W(I,J)+
5*(I,J))/2)/2*DELR**2
SOP(J)=2*PAR2WP=PARZUP
SOM(J)=2*PAR2WM=PARZUM
SOP(J)=SOP(J)+SOM(J)/2
SRP=(SI+1,J)+SI(I,J))/2
SR1=(SRP+SR1H)/2
SR2=(SR2P+SR2H)/2

C CALCULATION OF DELTA S (RADIAL ADVECTION) AXIAL CORE
URP=1.0/(4.0*DELR(J))*(SH1(I,J+1)+SH1(I,J+1)+SH1(I+1,J+1)=
6*SH1(I+1,J+1)
DELSRAM=4.0/(DELR*RPLUS(I))*(SOP(J)+SOM(J)+
78*(I+1,J+1)+S(I+1,J+1))/4*URP

C CALCULATION OF DELTA S (AXIAL ADVECTION) AXIAL CORE
WZP=1.0/(2.0*RPLUS(I)*DELR)*(SH1(I+1,J)+SH1(I,J+1)+
8*SH1(I+1,J)=SH1(I,J)
WZM=1.0/(2.0*RPLUS(I)*DELR)*(SH1(I+1,J+1)+SH1(I,J)+
7*SH1(I+1,J)=SH1(I,J))
DELSZA=1.0/DELR(J)*(WZP+SOM(J)+WZP+SOM(J))

C CALCULATION OF DELTA S (ROTATIONAL) AXIAL CORE
DELSA=1.0/DELR(J)*(16.0/(25.0*DELR**2))((R(2)*V1(2,J+1))**2=8*(R(2)*V1(2,J+1)**2)/2)

C CALCULATION OF DELTA S (BOUYANCY) AXIAL CORE
DELSB=8.0*VEC*/G*(DELR**2)*((T1(I+1,J+1)+T1(I,J+1)+
6+T1(I+1,J+1)+T1(I,J+1))/2.0)

C CALCULATION OF DELTA S (RADIAL VISCOUS) AXIAL CORE
SR1=(S(I+1,J+1)+S(I+1,J))/2
SR2=(S(I+1,J+1)+S(I+1,J))/2
SRP=(SR1+SR2)/2
PARSR=(SR2P+SR2P)/DELR
PARRP=1.0/RPLUS(I)*2.0*RPLUS(I)*SRP+RPLUS(I)**2*PARSR)
C
C CALCULATION OF DELTA S (AXIAL VISCOSITY AXIAL CORE)

DELSAV=(KV/DELZ(J))*((S(I,J+1)-S(I,J))+
5*SOP(J)+S0A(J))/((2*DELZ(J))*(S(I,J)+S(I,J+1)+
6*S0A(J)+S0M(J))/((2*DELZ(J)))

C
C CALCULATION OF DELTA II (EXTERNAL \)

C
C Nodal number scheme (corrected) for calculation of
C
C
C
C
C
31 SHIBAR(L,N)=E(L,N)*SHIBAR(L,N+1)+F(L,N)

C CALCULATE SHI
SUM3=0.0
ERR=0.0
DO 33 I=2,IIM1
DO 33 J=2, JJM1
SUM2=0.0
DO 36 K=1, IIM2
B=P(I-1,K)*SHIBAR(K,J)
36 SUM2=SUM2+B
EE=SHI(I,J)
SHI(I,J)=SUM2
ERR=ERR+(EE=SHI(I,J))**2
SUM3=SUM3+ABS(SHI(I,J))
33 CONTINUE
IF (WK.EQ.2) GO TO 5012
ERRMAX(KK)=ERR**0.5/SUM3/DELTIM
5012 CONTINUE
C CALCULATION OF U AND W VELOCITIES
DO 56 I=2, IIM1
DO 56 J=2, JJM1
U(I,1)=1.0/R(I)*(SHI(I+1,2)-SHI(I+1,1)+
6SHI(I-1,2)-SHI(I-1,1))/(2*DELZPL(1))
W(I,1)=1/R(I)*(SHI(I+1,1)-SHI(I,1)+SHI(I+1,2)-
7SHI(I-1,1))/(4*DELR)
W(I,J1)=1/R(I)*(SHI(I,J+1)-SHI(I+1,J)+
4SHI(I,J-1)-SHI(I,J))/(2*DELR)
W(I,J)=1/RPLUS(1)*(SHI(1,J+1)-SHI(1,J)+
5SHI(1,J-1)-SHI(1,J))/(2*DELR)
W(I,J1)=1/R(I)*(SHI(I,J+1)-SHI(I,1)+SHI(I,J+1)-
8SHI(I,1))/(4*DELR)
U(I,J)=1/R(I)*(SHI(I+1,J)-SHI(I,J)+
9SHI(I+1,J))/(4*DELR)
U(I,J1)=1/R(I)*(SHI(I+1,J)-SHI(I,1)+SHI(I+1,J1)-
10SHI(I,1))/(4*DELR)
U(I,J)=1/R(I)*(SHI(I+1,J)-SHI(I,J)+
11SHI(I+1,J))/(4*DELR)
U(I,J1)=1/R(I)*(SHI(I+1,J)-SHI(I,1)+SHI(I+1,J1)-
12SHI(I,1))/(4*DELR)
U(I,J)=1/R(I)*(SHI(I+1,J)-SHI(I,J)+
13SHI(I+1,J))/(4*DELR)
U(I,J1)=1/R(I)*(SHI(I+1,J)-SHI(I,1)+SHI(I+1,J1)-
14SHI(I,1))/(4*DELR)
U(I,J)=1.0/R(I)*(SHI(I,J)-SHI(I,J+1))/(DELZMI(JJ))
U(I,J1)=1.0/R(I)*(SHI(I,J)-SHI(I,J+1))/(DELZMI(JJ))
W(I,J)=1/R(I)*(SHI(I+1,J)-SHI(I,J))/(2*DELR)
W(I,J1)=1/R(I)*(SHI(I+1,J)-SHI(I,J))/(2*DELR)
W(I,J)=0
U(I,J)=0
W(I,1)=0
U(I,I)=0
56 CONTINUE
IF (WK.EQ.1) GO TO 512
DO 513 I=2, IIM1
DO 513 J=2, JJM1
SHI(I,2)=W(I,1)*R(I)*(2*DELZ(1))**2)**2/(-2)
SHI(I,1,J)=(W(I,J)*CR*DELR)**2)/(-2)
IF (I, GT, PP) GO TO 513
SHI(I,JJ-1)=(W(I,JJ)*R(I)*(2*DELZ(JJ))**2)/(-2)
513 CONTINUE
DO 623 I=2,II
DO 623 J=1,JJ
U(I,J) = I/R(I)*(SHI(I+1,J)-SHI(I-1,J)+SHI(I-1,J)-
5*SHI(I-1,J)/(2*DELP2(I))
W(I,J) = I/R(I)*(SHI(I,J+1)-SHI(I,J-1)+SHI(I-1,J)+SHI(I,J-1)-
4*SHI(I,J-1)/(2*DELR))
U(I,J) = I/R(I)*(SHI(I,2)-SHI(I,J))
U(I,J) = I/R(I)*(SHI(I,J+1)-SHI(I,J))
W(I,J) = I/R(I)*(SHI(I+1,J)-SHI(I,J))/2*DELP2(I)
GO TO 623
IF (I,GT,PP) GO TO 623
U(I,J) = I/R(I)*(SHI(I+1,J)+SHI(I,J))/2*DELP2(I)
623 CONTINUE
512 CONTINUE
C BOUNDARY CONDITION FOR W
DO 84 I=2,II
DO 84 J=1,JJ
W(I,J) = 0.1*(R(I)*1/(2*DELP2(I))**2)*SHI(I,J)
W(I,J) = 0.1*(R(I)*1/(2*DELP2(I))**2)*SHI(I,J)
GO TO 83
83 S(I,J) = W(I,J)/R(I)
84 CONTINUE
IF(WK,Eq,1) GO TO 511
IF(WK,Eq,2) GO TO 511
1010 CONTINUE
C BOUNDARY CONDITIONS FOR TEMPERATURE
C TEMPERATURE CALCULATION WITH RADIATION
IF (KK, Eq, 1) GO TO 39
RADCOE = 7.64 E -13
IF (PA, GT, 1) GO TO 85
CALL RADV (II, JJ, R, Z, JJ, PP, CR, CRYR, DELR, RSUR,
5RWALL, MPSH, RAR, RWO, RCYR, RSO, RSHIED)
85 CONTINUE
CALL DELTR (II, JJ, T2, RWALL, EB1, EB5, PA, EB, TSH, TEMP, NOSH,
6DELZ, RSUR, MPSH, DELRAD, JS, RAR, RWO, RCYR, RSO, RSHIED)
IMP = II - MPSH + 1
PA = PA + 1
39 CONTINUE
112 CONTINUE
C CALCULATE NEW TEMPERATURE DISTRIBUTION
DO 48 I=2,II
DO 48 J=1,JJ
81 IF (J, LE, JJ) GO TO 41
IF (I, LT, MPP) GO TO 48
C CALCULATION OF DELTA T FOR SURFACE
PARTR = (T(I-1,J)-T(I,J))/(2*DELR)
DELTR2 = (T(I+1,J)-2*T(I,J)+T(I,J))/(2*DELR)**2
DELZ2 = (T(I,J)-2*T(I,J)+T(I,J))/(2*DELR)**2
7DELP = 1/R(I)*PARTR+DELZ2
GO TO 713
C CALCULATE PARTIAL RUT W/T R
RUTP = (1/2*(R(I+1,J)+R(I,J)))**1/2*(U(I+1,J)+U(I,J))**1/2
81/2*(T(I+1,J)+T(I,J))
RUTH = (1/2*(R(I+1,J)+R(I,J)))**1/2*(U(I+1,J)+U(I,J))**1/2
41/2*(T1(I,J)+T1(I,J))
PARRUT=(RUTP=RUTM)/DELR
C CALCULATE PARTIAL W/T W/T Z
PTWZ=(3*T1(I,J)=4*T1(I,J)+T1(I,J=2))
PWZ=(3*W(I,J)=4*W(I,J=1)+W(I,J=2))
PARTZ=((W(I,J)+W(I,J=1))/2*PTWZ+(T1(I,J)+T1(I,J=1))/2
r*PWZ)/(2,0*DELMZ(J))
PARTZ=0,0
PARRUT=0,0
713 CONTINUE
C CALCULATE PARTIAL RUT W/T R
RUTP=(1/2*(R(I+1)+R(I)))*1/2*(U(I+1,J)+U(I+J))*
61/2*(T2(I+1,J)+T2(I,J))
RUTM=(1/2*(R(I+1)+R(I)))*1/2*(U(I+1,J)+U(I,J))*
T1/2*(T2(I+1,J)+T2(I,J))
PARRUT=(RUTP-RUTM)/DELR
C CALCULATE W/T W/T Z
WTP=W(I,J)*T2(I,J)
WTM=W(I,J=1)*T2(I,J=1)
WTP=W(I,J=2)*T2(I,J=2)
PARTZ=(3*WTP+4*WTO+WTM)/(2*DELMZ(J))
DELT(I,J)=TD*DELSQT=(1,0/R(I))*PARRUT=PARTZ
IF(KK,NE,1) GO TO 42
422 CONTINUE
TODEL=DELT(I,J)*DELTIM
GO TO 46
42 TODER=(DELT(I,J)=DELRAD(I))*DELTIM
46 T2(I,J)=T1(I,J)+TODER
GO TO 48
41 CONTINUE
C CALCULATION OF DELT T FOR CONDUCTION
PARTR=(T1(I+1,J)T1(I+1,J))/2*DELR
DELT2R= (T1(I+1,J)=2, T1(I,J) +T1(I,J)=1)/DELT2R
DELT2Z=T1(I,J=1)=T1(I,J)/DELTZ(J)=2-(T1(I,J)=
6T1(I,J=1)/DELMZ(J)=2
IF (I, EQ, 2) PARTR=PARTR=2
DELSQT=DELT2R+1/R(I)*PARTR+DELT2Z
C CALCULATE PARTIAL RUT W/T R
RUTP=(1,2,*(R(I+1)+R(I)))*1/2*(U(I+1,J)+U(I,J))*
51/2,*(T2(I+1,J)+T2(I,J))
RUTM=(1/2,*(R(I+1)+R(I)))*1/2,*(U(I+1,J)+U(I,J))*
T1,2,*(T2(I+1,J)+T2(I,J))
PARRUT=(RUTP-RUTM)/DELR
C CALCULATE PARTIAL W/T W/T Z
WTP=(1,2,*(W(I,J+1)+W(I,J)))*1/2*(T1(I,J+1)T1(I,J))
WTM=(1,2,*(W(I,J=1)+W(I,J))*1/2,*(T1(I,J=1)+T1(I,J))
PARTZ=WTP/DELTZ(J)=WTM/DELMZ(J)
DELT(I,J)=TD*DELSQT=1/R(I)*PARRUT=PARTZ
T2(I,J)=T1(I,J)+DELT(I,J)*DELTIM
IF(T2(I,J),GE,1773,) T2(I,J)=1773,0
48 CONTINUE
DTMAX=0,0
IF(K, EQ, 1) GO TO 5022
DO 5021 I=1,II
DO 5021 J=1,JJ
DTT=ABS(T1(I,J)=T2(I,J))
DTDD=ABS(DTT/DELTIM)
IF(DTDD,GT,DTMAX) DTMAX=DTDD
5021 CONTINUE
IF(DTMAX,GT,0.05) GO TO 5022
GO TO 100

5022 CONTINUE
IF (NOSH,NE,5) GO TO 624
SUMT2=0.0
ST2=0.0
DO 622 I=1,II
ST2=ST2*(I,J) + T2NSH(I)**4
IF (ST2, EQ, 0) GO TO 622
SUMT2=SUMT2+ST2/ABS(ST2)*(ABS(ST2)**0.25

622 CONTINUE
AVTSUM=SUMT2/I
DO 624 I=1,II
T2(II,1)=T2(II,1)-AVTSUM*DELTIM*T*0.004
IF (I, EQ, 1) GO TO 624
T2(1,1)=T2(1,1)-AVTSUM*DELTIM*T*0.004

624 CONTINUE
DO 47 I=1,II
DO 47 J=1, JJ
T1(I,J)=T2(2,J)
T2(I,J)=T1(I,J)
T1(I,J)=T2(I,J)

47 CONTINUE

C CALCULATION OF NEW ROTATIONAL VELOCITY
DO 57 I=2, IIM1
DO 57 J=2, JJ
C BOUNDARY FOR ROTATION
IF (JJ, GT, JJ) GO TO 91
IF (I, LT, PP) GO TO 57
V1 (I,J)= V2 (I,J-1)
V2 (I,J)= V1 (I,J)
GO TO 57

91 DELOR2=V1 (I+1,J-1)+2*V1 (I,J)+V1 (I-1,J))/*DELR**2
PARAR=V1 (I+1,J)+V1 (I-1,J))/*DELR/2,
PAROZ=V1 (I,J-1)+V1 (I,J))/*DELZ/(J-2),
DELORZ=((V1 (I,J-1)+V1 (I,J))/*DELZPL(J)=V1 (I,J)=

9V1 (I,J-1))/*DELZMI(J))/*DELZ(J)
DELTO2=DELORZ+1/R(I)*PARAR+DELOZ2
AA=KV*V1 (I,J)/R(I)**2
BD=2.*U(I,J)*V1 (I,J)/R(I)
OUP=1,1,2,*(V1(I+1,J)+V1(I,J))*1/2,*(U(I+1,J)+U(I,J)))
OUM=1,1,2,*(V1(I,J)+V1(I,J))*1/2,*(U(I,J)+U(I-1,J)))
PAROR=(OUP/OUM)/DELR
OPW=1,1,2,*(V1(I,J-1)+V1(I,J))*1/2,*(W(I,J+1)+W(I,J)))
OMP=1,1,2,*(V1(I,J-1)+V1(I,J))*1/2,*(W(I,J)+W(I,J)))
PAROWZ=(OPW/DELZPL(J)=OUM/DELZMI(J))
IF (I, EQ, 2) PAROR=0
IF (I, EQ, 2) PAROWZ=0
IF (J, EQ, 2) PAROWZ=0
IF (J, EQ, 2) PAROR=0
IF (J, EQ, JMJ1) PAROWZ=0
IF (J, EQ, JMJ1) PAROR=0

1522 CONTINUE
DELTO=(KV*DELTO2=(PAROR+PAROWZ+AA+BD))/*DELTIM
V2 (I,J)=V1 (I,J)+DELTO
V2(1,J)=V2(2,J)/4

57 CONTINUE
V2(1,1)=V2(2,1)/4
V2(1, JJ) = V2(2, JJ) / 4
C UPDATE V1 AND CALCULATE OMEGA
   DO 67 I = 1, II
   DO 67 J = 1, JJ
   IF (V2(I, J) < OT2) V2(I, J) = V2(6, 20)
   IF (V2(I, J) > V2(20, 10)) V2(I, J) = V2(20, 10)
   V1(I, J) = V2(I, J)
   IF (KK.EQ.1) GOTO 67
   OMEGA(I, J) = V1(I, J) * R(1)
   OMEGA(I, J) = OMEGA(2, J) / 10
67 CONTINUE
   IF (KSAV.EQ.1) GOTO 123
   MMM = MMM + 1
   IF (MMM.EQ.1) GOTO 114
   IF (MMM = 60) 115, 113, 116
114 DELTIM = DELTIM + 7
   GOTO 115
113 DELTIM = DELTIM / 7
116 CONTINUE
   LLL = LLL + 1
   IF (LLL.EQ.100) GOTO 117
   IF (LLL.EQ.100) GOTO 112
123 CONTINUE
   IF (ERMAX(KK, LT, 0, 00) GO TO 100
   KK = KK + 1
   IF (KK.EQ. KP) GOTO 100
   IF (KK .NE. NN) GOTO 118
   LLL = 100
   NN = NN + 1
   NN = NN * KP / KPRINT
   GOTO 111
511 CONTINUE
C CALCULATION OF W CHECK AND W ERROR
   DO 63 J = 1, JJ
   DO 63 I = 2, II
   FPRIM(J) = 1
   IF (I.EQ. II) GOTO 64
   C J=JJ
   IF (J .NE. JJ) GOTO 77
   PARU(I, J) = ((U(I, J) - U(I, J = 1)) / (2 * DELZ(J))
   PARWR(I, J) = ((W(I, J + 1) - W(I + 1, J)) / (2 * DELR))
   GOTO 66
77 IF (J .NE. 1) GOTO 65
   C J=II
   IF (J .NE. JJ) GOTO 77
   PARU(I, J) = ((U(I, J) - U(I, J + 1)) / 2 * DELZ(J))
   PARWR(I, J) = 0
   PARU(I, J) = ((U(I, J) - U(I, J)) / (2 * DELZ(J))
   GOTO 66
65 PARU(I, J) = ((W(I, J) - W(I, J + 1)) / (2 * DELR))
   PARU(I, J) = ((U(I, J + 1) - U(I, J)) / DELZ(J)) + (U(I, J) - U(I, J - 1))
   4 / DELZ(J)
   GOTO 66
   C I=II
   64 PARU(I, J) = ((W(I, J) - W(I, J)) / DELR
   IF (J .EQ. 1) GOTO 641
   IF (J .EQ. J) GOTO 642
   C J=II
   PARU(I, J) = ((U(I, J) - U(I, J)) / (2 * DELZ(J)))
   GOTO 66
641  PARU(I,J)=((U(I,J)=I-U(I,J))/2.0*DELZ(J))
GO TO 66
C J=J+1
642  PARU(I,J)=((U(I,J)=U(I,J)+1)/(2.0*DELZ(J)))
66  WCHECK(I,J)=PARU(I,J)+1.0/FPRIM(J)*PARU(I,J)
WW(I,J)=WW(I,J)+WCHECK(I,J)
ERROR(I,J)=WW (I,J)=WCHECK(I,J)
IF(I,NE,1) S(I,J)=WW(I,J)/R(I)
63  CONTINUE
IF(WK,NE,2) GO TO 1010
WK=2
GO TO 101
100  CONTINUE
DO 631 I=1,II
DO 631 J=1,JJ
DELS(I,J)=DELS(I,J)/DELTIM
IF(NOSH,NE,5) GO TO 631
T2MSH(I)=T2(I,JJ)
631  CONTINUE
MI=MW
WRITE (JJJ,698)
WRITE (JJJ,699) (DMAX)
WRITE (JJJ,824) ((DELTI(I),IA,JIA),I=1,MI INT)
WRITE (JJJ,824) ((DELTI(I),IA,JIA),I=1,MI INT)
WRITE (JJJ,830) (KE, ERMAX(KE), KE=1, KK, INT)
WRITE (JJJ,70)
WRITE (JJJ,828) HKTEMP
WRITE (JJJ,828) HKTEMP
WRITE (JJJ,802) ((MM(I,N),JM=1,II),N=1,JJ)
WRITE (JJJ,804) ((II(I,N),J=1,II),N=1,JJ)
WRITE (JJJ,827) (( V2 (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,70)
WRITE (JJJ,803) (MM(I,N),J=1,II),N=1,JJ)
WRITE (JJJ,800) (MM(I,N),J=1,II),N=1,JJ)
WRITE (JJJ,70)
WRITE (JJJ,806) ((U (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,808) ((W (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,70)
WRITE (JJJ,810) ((S (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,820) ((S (I,J),I=1,II),J=1,JJ)
WRITE (JJJ,910) ((DELT(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,817) ((DELT(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,807) (MM(I,N),J=1,II),N=1,JJ)
WRITE (JJJ,70)
WRITE (JJJ,809) ((WERROR(I,J),I=1,II),J=1,JJ)
WRITE (JJJ,809)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (JJJ,808)
WRITE (I,J,700) ((I,J),I=1,II,J=1,JJ)
WRITE (I,J,701) ((V1(I,J),I=1,II,J=1,JJ)
WRITE (I,J,701) ((O(I,J),I=1,II,M2),J=1,J1*2)
WRITE (I,J,701) (XI(I),I=1,II,M2)
WRITE (I,J,701) ((W(I,J),I=1,II),J=1,JJ)
WRITE (I,J,701) ((U(I,J),I=1,II,J=1,JJ)
WRITE (I,J,701) ((SHI(I,J),I=1,II),J=1,JJ)
WRITE (I,J,701) ((S(I,J),I=1,II,J=1,JJ)
WRITE (I,J,701) ((WM(I,J),I=1,II),J=1,JJ)
WRITE (NS,700) (T2NSH(I),I=1,II)

698 FORMAT ('I',2X,'SUM OF DELTA TEMP/DELTIME')
663 FORMAT ('0',2X,F10,3)
699 FORMAT ('0',3X,F10,5)
700 FORMAT (S(1X,F11,5))
701 FORMAT (S(1X,F11,7))
498 FORMAT ('0',3X,'I',10X,'EB8',11X,'EB1',13X,'J8')
487 FORMAT ('0',3X,'I',3X,'DELTA RADIATION',3X,'DELTA CONVECTION')
672 FORMAT ('0',1X,(I4,3(4X,F12,5)))
676 FORMAT ('0',1X,(E12,4))
982 FORMAT ('0',1X,(I4,3X,E12,4,4X,E12,4))
70 FORMAT ('I')
72 FORMAT ('0',10(F8,4,1X))
74 FORMAT ('0',10(F8,4,1X))
222 FORMAT ('0',18(F6,3,1X))
223 FORMAT ('0',18(F6,3,1X))
800 FORMAT ('0',50X,'STREAM FUNCTION SHI')
801 FORMAT ('0',1X,(I4,2F7,3,18F6,3))
817 FORMAT ('0',1X,(I4,2(F8,2),18(F6,2))
802 FORMAT ('0',50X,'TEMPERATURE DISTRIBUTION')
803 FORMAT ('0',50X,'VORTICITY PARAMETER W')
804 FORMAT ('0',50X,'ROTATIONAL COMPONENT V2')
805 FORMAT ('0',20(1X,F5,0))
806 FORMAT ('0',50X,'RADIAL VELOCITY COMPONENT U')
807 FORMAT ('0',50X,'WCHECK')
808 FORMAT ('0',50X,'AXIAL VELOCITY COMPONENT W')
809 FORMAT ('0',50X,'WERROR')
820 FORMAT ('0',50X,'DELTA')
810 FORMAT ('0',50X,'B')
811 FORMAT ('0',50X,'R(I)')
812 FORMAT ('0',50X,'Z(J)')
813 FORMAT ('0',50X,'Q(I,J) EIGEN VECTORS')
814 FORMAT ('0',50X,'X(J) EIGEN VALUES)
815 FORMAT ('0',50X,'P(I,J)')
816 FORMAT ('0',50X,'PINV (I,J)')
824 FORMAT ('0',4(I4,2F11,6,2X,('I3,'I3,'I3')
830 FORMAT ('0',7(F2,14,2F10,6))
910 FORMAT ('0',50X,'DELS')
827 FORMAT ('0',20(F6,2)
828 FORMAT ('0',1X,'SHIELD TEMPERATURE')
STOP
END

C

SUBROUTINE TOL7(NM,N,D,E,Z,IERR)
INTEGER I,J,K,L,M,N,II,NM,ML,IERR,L1
REAL D(N),E(N),Z(NM,N)
REAL SORT,ABS,SIGN
MACHEP= 1.0E-6
IERR = 0

END
IF (N .EQ. 1) GO TO 1001
  DO 100 I = 2, N
  E(I-1)=E(I)
  F=0.0
  B=0.0
  E(N)=0.0
  DO 240 L=1, N
  J=0
  H=MACHEP*(ABS(D(L))+ABS(E(L)))
  IF (B .LT. H) B=H
C LOOK FOR SMALL SUB- DIAGONAL ELEMENT
  DO 110 M=L, N
  IF (ABS(E(M)) .LE. B) GO TO 120
C CONTINUE
  110 IF (M .EQ. L) GO TO 220
  130 IF (J .EQ. 30) GO TO 1000
  J=J+1
C FORM SHIFT
  L=L+1
  G=D(L)
  P=(N(L)=G)/(2.0*E(L))
  R=SORT(P*P+1.0)
  D(L)=E(L)/(P+SIGN(R,P))
  H=G=D(L)
  DO 140 I=L, N
  140 D(I)=D(I)-H
  F=F+H
C QL TRANSFORMATION
  P=D(M)
  C=1.0
  S=0.0
  MML=M=L
C FOR I=M-1 STEP -1 UNTIL L DO
  DO 200 II=1, MML
  I=M-II
  G=C*E(I)
  H=C*P
  IF (ABS(P) .LT. ABS(E(I))) GO TO 150
  C=E(I)/P
  R=SORT(C*C+1.0)
  E(I+1)=S*P*R
  S=C/R
  C=1.0/R
  GO TO 160
  150 C=P/F(I)
  R=SORT(C*C+1)
  E(I+1)=S*E(I)*R
  S=1.0/R
  C=C*S
  160 P=C*D(I)=S*G
  D(I+1)=H+S*(C*G+S*D(I))
C FORM VECTOR
  DO 180 K=1, N
  H=Z(K, I+1)
  Z(K, I+1)=S*Z(K, I)+C*H
  Z(K, I)=C*Z(K, I)+S*H
  180 CONTINUE
C CONTINUE
  200 CONTINUE

E(L)= 8*P
D(L)=C*P
IF (ABS(E(L)) .GT. R) GO TO 130
220 D(L)=D(L)+F
240 CONTINUE

C ORDER EIGENVALUE AND EIGENVECTORS
DO 300 II=2,N
I=II-1
K=I
P=D(I)
DO 260 J=II,N
IF (D(J) .GE. P) GO TO 260
K=J
P=D(J)
260 CONTINUE
IF(K.EQ.0) GO TO 300
D(K)=D(I)
D(I)=P
DO 280 J=1,N
P=Z(J,I)
Z(J,I)=Z(J,K)
Z(J,K)=P
280 CONTINUE
300 CONTINUE
GO TO 1001
1000 IERR=L
1001 RETURN
END
SUBROUTINE DELTR (II,JJ,T2,RWALL,EB1,EBS,PA,EB,SHTEM,F,NOSH,
5DELZ,RSUR,MPHSH,MDELRA,JS,RAR,RNOS,RCRY,RSO,RSHED)
DIMENSION T2(20,20),BPP(400),JS(20)
DIMENSION EB1(20),EBS(19),DELZ(20)
DIMENSION RSUR(20),DELRA(20)
DIMENSION RAR(19,19),RNOS(16,16)
DIMENSION WAREA(400),REBS(1,17),EB(153),ICHNG(34),DET(17)
DIMENSION WR(120),RREBS(1,15),MICHNG(30),RDET(15)
REAL J8
AVCXY=1400.
EPSILN=0.318
CRUTEN=1773
SIGMAR=1.3555E+12
TWALL=500.
TSRUS=500.
RNO=2.33
CP=0.233
MMPHSH=II+MMPHSH+1
II=II+II+1
II=II+II+1
II=II+II+1
MPP=MMPHSH+1
MP2=MPP+1
IJOB=0
IF (PA.GT.2) IJOB=2
EBW=SIGMAR*(TWALL)**4
EBWTOT=(IMPSH*EB/PAWALL)
C STEFAN-BOLTZMANN CONSTANT SIGMAR (CAL/(SEC*K**3*C**2))
DO 800 II=MPPHSH,II
K=I+MPPHSH+1
EB1(K)=SIGMAR*(T2(I,JJ))**4
REBS(1,K)=EBS(K)/RSUR(K)
800 CONTINUE
IF(NOSH,NE,5) GO TO 900
EBS(IIMPSH+1)=SIGMAR*TWALL**4
REBS(1,IIMPSH+1)=EBS(IIMPSH+1)/RWALL
EBS(IIMPSH+2)=SIGMAR*VCRT**4
REBS(1,IIMPSH+2)=EBS(IIMPSH+2)/RCRY
EBS(IIMPSH+3)=SIGMAR*TSR**4
REBS(1,IIMPSH+3)=EBS(IIMPSH+3)/RSQ
EBS(IIMPSH+4)=0.0
REBS(1,IIMPSH+4)=0.0
EBS(IIMPSH+5)=0.0
REBS(1,IIMPSH+5)=0.0
DO 801 I=1,IIMPS5
EB(I)=EBS(I)
K=0
801 CONTINUE
IF(IJOB,EQ,2) GO TO 613
DO 614 J=1,IIMPS5
DO 614 I=1,IIMPS5
K=K+1
EB(K)=RAR(I,J)
IF (I,EQ,J) GO TO 613
614 CONTINUE
613 CONTINUE
N=IIMPS5
IB=IIMPS5
M=1
CALL LEQ18(EB,N,REBS,M,IB,IJOB,ICHNG,DDET,IER)
EPS=1.0E=4
617 CONTINUE
DO 612 I=1,IIMPS5
IF (I,GT,IIMPSH) GO TO 616
JS(I)=REBS(1,I)
GO TO 612
616 JS(IIMPSH+1)=REBS(I,IIMPSH+1)
JS(IIMPSH+2)=REBS(I,IIMPSH+2)
JS(IIMPSH+3)=REBS(I,IIMPSH+3)
JS(IIMPSH+4)=REBS(I,IIMPSH+4)
JS(IIMPSH+5)=REBS(I,IIMPSH+5)
612 CONTINUE
SHTEMP=((JS(IIMPSH+4)+JS(IIMPSH+5))/(2*SIGMAR))**.25
C EB IS REALLY JS( )
C CALCULATE DELT TEMP DUE TO RADIATION
DO 981 I=1,IIMPSH
K=I+MPSH-1
DELRAD(K)=(EB(I)*JS(I))/(RHO*CP*(1.0*EPSILS)/7EPSILS)*DELZ(JJ)
981 CONTINUE
C FREE RADIATION FROM ELEMENTS NOT COVER BY SHIELD
DO 611 I=MPP2,MPP
EBS(I+IIMPSH+7=MPP2)=SIGMAR*T2(I,JJ)**4
DELRAD(I)=(EBS(I+IIMPSH+7=MPP2)*EPSILS)/(RHO*CP*DELZ(JJ))
IF(NOSH,EQ,5) GO TO 746
DELRAD(MPPSH=2)=0.9*DELRAD(MPPSH)
DELRAD(MPPSH=1)=0.95*DELRAD(MPPSH)
GO TO 611
746 DELRAD(MPPSH=2)=1.1*DELRAD(MPPSH)
DELRAD(MPPSH=1)=1.05*DELRAD(MPPSH)
611 CONTINUE
GO TO 1000

900 CONTINUE

C CALCULATION OF DELRAD FOR NO SHIELD

DO 905 I=1,IMPSH
RREBS(I,I)=REBS(I,I)

905 CONTINUE

EBS(IMPSH+1)=SIGMAR*TWALL**4
RREBS(1,IMPSH+1)=EBS(IMPSH+1)/RWALL
EBS(IMPSH+2)=SIGMAR*AVCRYT**4
RREBS(1,IMPSH+2)=EBS(IMPSH+2)/RCRY
EBS(IMPSH+3)=SIGMAR*TSURG**4
RREBS(1,IMPSH+3)=EBS(IMPSH+3)/RSO
DO 901 I=1,IMPSH+3
EBI(I)=EBS(I)

901 CONTINUE

K=0
IF (IJOB,EQ,2) GO TO 712
DO 712 J=1,IMPS3
DO 713 I=1,IMPS3
K=K+1
WR(K)=RWDS(I,J)
IF (I,EQ,J) GO TO 712

713 CONTINUE

712 CONTINUE

N3=IMPSH+3
M3=1
I3=IMPSH+3
CALL LEQ18(WR,N3,RREBS,M3,I3,IJOB,MICHNG,RDET,IER)
DO 618 I=1,IMPSH+3
RREBS(I,1)=RREBS(I,1)

618 CONTINUE

GO TO 617

1000 CONTINUE

RETURN

END

SUBROUTINE RADVF (II,JJ,R,Z,JJJ,PP,CR,CYR,DER,RUR,
RWall,MPSh,RAR,RWDS,RCry,RSO,RSHIEL)
DIMENSION R(20),Z(20),BPP(400)
DIMENSION TANPTX(20,3),TANPTY(20,3),TANPTZ(20,3)
DIMENSION CTWLY(20),CTWLY(20),CTWLY(20)
DIMENSION WAINX(20,20),WAINY(20,20),WAINZ(20,20)
DIMENSION WALPTX(20,20),WALPTY(20,20),WALPTZ(20,20)
DIMENSION SMPTX(20,20),SMPTY(20,20),SMPTZ(20,20)
DIMENSION XABX(20),YABX(20),YABX(20),ZABX(20)
DIMENSION VFITWA(20),VFITSH(20)
DIMENSION XP(20),YP(20),ZP(20)
DIMENSION SMPTX(20),SMPTY(20),SMPTZ(20)
DIMENSION WAPTX(20),WAPTY(20),WAPTZ(20)
DIMENSION RLJUS(20),RMNUS(20)
DIMENSION AREAUS(20),BR(20),YY(20)
DIMENSION REE(20),RSUR(20),RITWA(20),RITSH(20),RITSW(20)
DIMENSION BP(20,20),BPP(20,20),BPP(400)
DIMENSION CISX(30),CISX(30),CISX(30)
DIMENSION CRISX(30),CRISY(30),CRISZ(30)
DIMENSION R19(19,19)
DIMENSION R16(16,16),RCRUS(40),RGOUS(40),RITW(20)
DIMENSION VFSUR(30),VFSUR(30),VFSURW(50),VFSUR(30)
DIMENSION CEX(30),CRY(30),CRZ(30),VFSUWA(50)
REAL PI
PI=3.141
II=20
JJ=20
IIIM1=II=1
CRHEIT=20.0
CR=11.55
CRYR=3.75
DELR=CR/(II=1)
PP=CRYR/DELR
DO 27 I=1,II
R(I)=DELRI(I)=1)
27 CONTINUE
C SMANG=SHIELD ANGLE
SHANGL=45.0
SHANG=SHANGL*(3.341/180.0)
C
C SHRIN = SHIELD RADIUS INNER
SHRIN= CRYR+1
C
C SHROUT = SHIELD RADIUS OUTER
SHROUT=CR
MPP=PP
MPPSH=(SHRIN/DELR+1.5)
IIIMPSH=II=MPPSH+1
C
C THAP1 = TANGENT PT ON SHIELD FROM SURFACE
MPPL1=MPP+1
DO 401 I=MPPSH,II
TANPTZ(I,1)=0.0
TANPTZ(I,2)=(CR=CRYR=2*DELR)*SIN(SHANG)
C1=(SHRIN-SHROUT)/(TANPTZ(I,1)=TANPTZ(I,2))
RQ=SHRIN=C1*TANPTZ(I,1)
DO 401 K=1,2
C2=(C1*TANPTZ(I,K)+RQ)
C3=(C1*TANPTZ(I,K)+RQ)*C1*0.5=TANPTZ(I,K)
TANPTY(I,K)=(C2**2=(C2**2=C3)/R(I))**2)**2=0.5
TANPTX(I,K)=(C2**2=TANPTY(I,K)**2)**2=0.5
401 CONTINUE
WRITE (JJJ,454)
WRITE (JJJ,455) SHANGL
402 WRITE (JJJ,490)
WRITE (JJJ,400)((TANPTX(I,K),TANPTY(I,K),TANPTZ(I,K)),K=1,2),
6I=MPPSH,II)
C
C WAIT = INTERSECTION PT ON WALL FROM SURFACE PT I
DO 450 I=MPPL1,II
YY(I)=0.0
DE=(TANPTY(I,1)=YY(I))/(TANPTX(I,1)=R(I))
BB=YY(I)=DE*R(I)
WALLR=CR
A=-DE**2+1
B=2*DE*BB
C=BB**2=WALLR**2
WAITXX(I,1)=(-B=(B**2=4.0*A*C)**0.5)/(2.0*A)
WAITYY(I,1)=DE*WAITXX(I,1)+BB
WAITZ(I,1)=0.0
K=2
RATSTW=WALLR/SHROUT
WAITXX(I,K)=TANPTX(I,2)*RATSTW
WAITYY(I,K)=TANPTY(I,2)*RATSTW
WAITZ(I,K)=TANPTZ(I,2)
450 CONTINUE
C  SHDPT  SHIELD POINTS FROM SURFACE PT I
DO 412 I=MPPSH,II
C  INCR  INCREMENT OF ARC
   INCR=2
   THA = TANPTY(I,1)/TANPTX(I,1)
   THAT1=ATAN(THA)
   DTHAT1=THAT1/INCR
   INCR1=INCR+1
   DO 410 M=1,INCR1
      SHDPTX(I,M)= SHRIN*COS((M-1)*DTHAT1)
      SHDPTY(I,M)= SHRIN*SIN((M-1)*DTHAT1)
      SHDPTZ(I,M)= TANPTZ(I,1)
   410 CONTINUE
   THA=TANPTY(I,2)/TANPTX(I,2)
   THASU=ATAN(THA!1)
   DTHASU=THASU/(3*INCR)
   INCR2= INCR+2
   INCTOT=INCR2+3*INCR
   DO 411 M=INCR2,INCTOT
      SHDPTX(I,M)= SHRUT*COS(THAT1=(M= INCR2)*DTHASU)
      SHDPTY(I,M)= SHRUT*SIN(THAT1=(M= INCR2)*DTHASU)
      SHDPTZ(I,M)= TANPTZ(I,2)
   411 CONTINUE
   N1=INCTOT+1
   SHDPTX(I, N1) = SHDPTX(I,1)
   SHDPTY(I, N1) = SHDPTY(I,1)
   SHDPTZ(I, N1) = SHDPTZ(I,1)
   412 CONTINUE
C  WALPT  WALL POINTS FROM SURFACE PT I
DO 465 I=MPPSH,II
   THATA2= ATAN(WAINTY(I,1)/WAINTX(I,1))
   DTHAT2=THATA2/(3*INCR)
   INCR1=3*INCR+1
   DO 470 M=1,INCR1
      WALPTX(I,M)= WALLR*COS((M-1)*DTHAT2)
      WALPTY(I,M)= WALLR*SIN((M-1)*DTHAT2)
      WALPTZ(I,M)= 0,0
   470 CONTINUE
   THAT2U=ATAN2(WAINTY(I,2), WAINTX(I,2))
   DTHAT2=THAT2U/(3, 0*INCR)
   INCR2=3*INCR+2
   NOFP=INCR2+3*INCR
   DO 478 M=INCR2,NOFP
      WALPTX(I,M)= WALLR*COS((NOFP=M)*DTHAT2)
      WALPTY(I,M)= WALLR*SIN((NOFP=M)*DTHAT2)
      WALPTZ(I,M)= TANPTZ(I,2)
   478 CONTINUE
   N2=NOFP+1
   WALPTX(I,N2)= WALPTX(I,1)
   WALPTY(I,N2)= WALPTY(I,1)
   WALPTZ(I,N2)= WALPTZ(I,1)
   465 CONTINUE
C  CTOWL  WALL POINTS FROM SURFACE CONE
   INCRSH=6
   INCSH1=INCRSH+1
   DO 460 I=1,INCSH1
   DELTSH=(SHRUT-SHRIN)/INCRSH
   CTOWLX(I)= SHRIN+(I-1)*DELTSH
   CTOWLX(I)= (WALLR**2=CTOWLX(I)**2)**0.5
CTOWLZ(I) = (CTOWLX(I)-R0)/C1

460 CONTINUE
INCW=3
INCP1=INCSh1
IF (CTOWLX(INCSh1),EQ,0,0) GO TO 463
THAWL=ATAN(CTOWLX(INCSh1)/CTOWLZ(INCSh1))
DTHAWL=THAWL/INCW
INCP1=INCW+INCSh1
IP1=INCSh1+1
DO 461 J=IP1,INCP1
CTOWLX(J)=WALLR*COS((INCWP1+J)*DTHAWL)
CTOWLX(J)=WALLR*SIN((INCWP1+J)*DTHAWL)
CTOWLZ(J)=TANPTZ(J,2)

461 CONTINUE

463 CONTINUE
THAWL=ATAN(CTOWLX(I)/CTOWLZ(I))
DTHAWL=THAWL/(INCW**2,0)
INCP2=INCP1+1
NPPIW=INCP2+INCW**2
DO 462 J=INCP2,NPPIW
CTOWLX(J)=WALLR*COS(J*INCWP2*DTHAWL)
CTOWLX(J)=WALLR*SIN(J*INCWP2*DTHAWL)
CTOWLZ(J)=TANPTZ(J,1)

462 CONTINUE
WRITE (JJJ,493)
WRITE (JJJ,400) (CTOWLX(I),CTOWLX(I),CTOWLZ(I)),I=1,NPPIW

C COORDINATES ON SHIELD = EQUAL AREA APART
INCSh=8
INCP1=INCSh+1
DO 480 N=2,INCSh
XASH(I)=SHRN
XASH(N)=(((SHRTU*2=SHRUI**2)/INCSh)+XASH(N=1)**2)*0.5
XASH(N=1)= (XASH(N)-XASH(N=1))/2,0+XASH(N=1)
YSH(N=1) =0,0
ZSH(N=1)=(XSH(N-1)=BO)/C1

480 CONTINUE
C VIEWFACTORS SURFACE TO SHIELD & WALL
DO 690 I=MPPSH,II
M=I=MPPSH+1
XP(M)=RI(I)
YP(M)=0,0
ZP(M)=0,0
MPTS=1
THET=0,0
DO 691 K=1,N1
SHPTX(K)=SHDPTX(I,K)
SHPTY(K)=SHDPTY(I,K)
SHPTZ(K)=SHDPTZ(I,K)

691 CONTINUE
CALL CONFCAX(XP,YP,ZP,MPTS,THET,SHPTX,SHPTY,SHPTZ,N1,VAFITS)
YFITS(I)=VAFITS#2,0
DO 692 K=1,N2
WAPTX(K)=WALPTX(I,K)
WAPTY(K)=WALPTY(I,K)
WAPTZ(K)=WALPTZ(I,K)

692 CONTINUE
CALL CONFCAX(XP,YP,ZP,MPTS,THET,WAPTX,WAPTY,WAPTZ,N2,VAFITW)
YFITSW(I)=VAFITW#2,0
THET1=ATAN(C1)
CALL CONFC(XSH,YSH,ZSH,INCSH,THET1,CTOWLX,CTOWLY,CTOWLZ,
6NPP1W,VFSTWA)

552 FORMAT('0',4X,F10.5)

C CALCULATE CRYSTAL TO SHIELD VF
CRX(1)=CRYR
CRY(1)=0.0
CRZ(1)=CRHEIT/2
CRLSHY=(SHRIN**2=CRYR**2)**0.5
CRUSHY=(CR**2=CRYR**2)**0.5
ARCI=ATAN(CRLSHY/CRYR)
ARCON=ATAN(CRUSHY/CRYR)
NOUT=6
WIN=3
DANCIN=ARCI/WIN
DO 800 I =1,NIN+1
CISX(I)=SHRIN*COS((I-1)*DANCIN)
CISY(I)=SHRIN*SIN((I-1)*DANCIN)
CISZ(I)=0.0
800 CONTINUE
DANC=ARCON/NOUT
DO 801 I=1,NOUT+1
CISX(I+WIN+1)=CR*COS((NOUT+1+I)*
7DANC)
CISY(I+WIN+1)=CR*SIN((NOUT+1+I)*
8DANC)
CISZ(I+WIN+1)=TANPTZ(I1,2)
801 CONTINUE
CISX(WIN+NOUT+2)=CISX(1)
CISY(WIN+NOUT+2)=CISY(1)
CISZ(WIN+NOUT+2)=CISZ(1)
NTP=WIN+NOUT+2
THETCR=0
INCRY=1
CALL CONFC (CRX,CRY,CRZ,INCR,THETCR,CISX,CISY,CISZ,
5NTOP, VFCRS)
VFCSRSH=2*VFCSR

C CALCULATION OF SURFACE TO CRYSTAL VIEWFACTOR
INCRY=3
DO 899 I=MPPSH,II
M=I-MPPSH+1
XP(M)=R(I)
YP(M)=0.0
ZP(M)=0.0
ANC=ASIN(CRYR/R(I))
DANC=ANC/INCR
DO 897 K=1,INCR+1
CRTSX(K)=CRYR*COS((K-1)*DANC)
CRTSY(K)=CRYR*SIN((K-1)*DANC)
CRTSZ(K)=0.0
897 CONTINUE
DO 898 K=INCR+2,2*INCR+2
CRTSX(K)=CRYR*COS((2*INCR+2+K)*DANC)
CRTSY(K)=CRYR*SIN((2*INCR+2+K)*DANC)
CRTSZ(K)=CRHEIT
898 CONTINUE
CRTSX(2*INCRY+3)=CRTSX(1)
CRTSY(2*INCRY+3)=CRTSY(1)
CRTSZ(2*INCRY+3)=CRTSZ(1)
N2=2*INCRY+3
CALL CONFCAC(XP,YP,ZP,MPTS,THET,CRSX,CRSBY,CRSZ,N2,VFSUC)
VFSUCR(I)=2*VFSUC

899 CONTINUE
WRITE(JJ,499)
WRITE (JJ,400) ((CRSX(I),CRSBY(I),CRSZ(I)),I=1,N2)
PI= 3.141
EPSILW=0.6
EPSILS=0.318
EPSISO=0.6
EPSICR=0.2
EPSISH=0.3
DO 900 I=MPPSII,II
RPLUS(I)=R(I)+DLR/2,
RMINUS(I)=R(I)+DLR/2,
AREASU(I)=PI* (RPLUS(I)**2-RMINUS(I)**2)
900 CONTINUE
AREASH=PI* (1.0+C1**2)**0.5*(C1*(TANPTZ(II,2)**2=8*TANPTZ(II,1)**2/2.0+B0*(TANPTZ(II,2)-TANPTZ(II,1))
AREAW=PI*WALLR*TANPTZ(II,2)
SRITSH=0.0
SRITWA=0.0
ARCRY=PI*CRYR*CRHEIT
AREASD=PI*CR**2
AWALL=AREAW
DO 600 I=1,IMPSS
M=1+MPPSH=1
RWAII = (1+EPSILW)/(EPSILW*A WALL)
RSUR(I)= (1+EPSILS)/(EPSILS*AREASU(M))
RITWA(I)= 1.0/(AREASU(M)*VFITSWA(M))
RITW(I)=RITWA(I)
RSTW = 1.0/(AREASH*VFSTW)
RITW = 1.0/(AREASU(M)*VFITSW(M))
SRITSH=SRITSH+1./RITSH(I)
SRITWA=SRITWA+1./RITWA(I)
600 CONTINUE
VFSSCR=ARCRY/AREASH*VFCSR
VFSSDH=1.0/VFSSCR
VFCSRSD=1.0/VFCSR
RSO=(1+EPSISO)/(EPSISO*AREASO)
RCRY=(1+EPSICR)/(EPSICR*ARCRY)
RCRH=1.0/(AREASH*VFSSCR)
RCRSD=1.0/(ARCRY*VFCSRSD)
RSOSH=1.0/(AREASH*VFSSDH)
RSHIDD=(1+EPSISH)/(EPSISH*AREASH)
WRITE (JJ,496)
WRITE (JJ,670) AREASH,AREAW,ARCRY,RSTWA,RWALL
WRITE (JJ,497)
WRITE (JJ,671) ((AREASU(I+MPPSH-1), RSUR(I), RITWA(I),
5RITSH(I)),I=1,IMPSS)
IMPSS=IMPSS+6
DD 700 J=1,IMPSS
DO 700 I=1,IMPSS
IF (J.GT.IMPSS) GO TO 702
IF (I.GT.IMPSS) GO TO 701
IF (I.NE. J) GO TO 701
RAP(I,J)=1./RSUR(J)+1./RITSH(J)+1./RITWA(J)
GO TO 700
701 CONTINUE
RAR(I,J)=0,0
RAR(IIMPSH+1,J)=1.0/RITWA(J)
RAR(IIMPSH+2,J)=0,0
RAR(IIMPSH+3,J)=0,0
RAR(IIMPSH+4,J)=0,0
RAR(IIMPSH+5,J)=1/RITBH(J)
RAR(IIMPSH+6,J)=0,0
GO TO 700

702 CONTINUE
IF (J,GT,IIMPSH+1) GO TO 703
IF (I,GT,IIMPSH) GO TO 704
RAR(I,J)=1/RITWA(I)
GO TO 700

704 CONTINUE
RAR(IIMPSH+1,J)=1/RSTWA+SRITWA+1/RWALL
RAR(IIMPSH+2,J)=0,0
RAR(IIMPSH+3,J)=0,0
RAR(IIMPSH+4,J)=0,0
RAR(IIMPSH+5,J)=1/R8TWA
RAR(IIMPSH+6,J)=0,0
GO TO 700

703 CONTINUE
IF (J,GT,IIMPSH+2) GO TO 705
IF (I,GT,IIMPSH) GO TO 706
RAR(I,J)=0,0
GO TO 700

706 CONTINUE
RAR(IIMPSH+1,J)=0,0
RAR(IIMPSH+2,J)=1/RCRY+1/RCRSH+1/RCR8D
RAR(IIMPSH+3,J)=1/RCR8D
RAR(IIMPSH+4,J)=1/RCRSH
RAR(IIMPSH+5,J)=0,0
RAR(IIMPSH+6,J)=0,0
GO TO 700

705 CONTINUE
IF(J,GT,IIMPSH+3) GO TO 707
IF(I,GT,IIMPSH+1) GO TO 708
RAR(I,J)=0,0

708 CONTINUE
RAR(IIMPSH+1,J)=0,0
RAR(IIMPSH+2,J)=1/RCR8D
RAR(IIMPSH+3,J)=1/R8OSH+1/R8D+1/RCR8D
RAR(IIMPSH+4,J)=1/R8OSH
RAR(IIMPSH+5,J)=0,0
RAR(IIMPSH+6,J)=0,0
GO TO 700

707 CONTINUE
IF (J,GT,IIMPSH+4) GO TO 721
IF (I,GT,IIMPSH) GO TO 722
RAR(I,J)=0,0
GO TO 700

722 RAR(IIMPSH+1,J)=0,0
RAR(IIMPSH+2,J)=1/RCRSH
RAR(IIMPSH+3,J)=1/R8OSH
RAR(IIMPSH+4,J)=1/RCSRSH+1/R8OSH+1/(2*RSHID)
RAR(IIMPSH+5,J)=1.0/(2*RSHID)
RAR(IIMPSH+6,J)=1/RSHID
GO TO 700

721 CONTINUE
IF(J,GT,1MPSH+5) GO TO 710
IF(I,GT,1MPSH) GO TO 709
RAR(I,J)=1.0/RITSH(I)
GO TO 700

709 RAR(I,1MPSH+1,J)=1.0/R8TWA
RAR(I,1MPSH+2,J)=0.0
RAR(I,1MPSH+3,J)=0.0
RAR(I,1MPSH+4,J)=1.0/(2*RSHIED)
RAR(I,1MPSH+5,J)=1.0/(2*RSHIED)+1/R8TWA*BRTSH
RAR(I,1MPSH+6,J)=1/RSHIED
GO TO 700

710 IF (I,GT,1MPSH) GO TO 711
RAR(I,J)=0.0
GO TO 700

711 CONTINUE
RAR(I,1MPSH+1,J)=0.0
RAR(I,1MPSH+2,J)=0.0
RAR(I,1MPSH+3,J)=0.0
RAR(I,1MPSH+4,J)=1.0
RAR(I,1MPSH+5,J)=1.0
RAR(I,1MPSH+6,J)=2.0

700 CONTINUE
SFCSU=0.0
ARESUR=0.0
SFITW=0.0
WRITE (JJJ,633)
WRITE (JJJ,678) ((RAR(I,J),I=1,1MPSH),J=1,1MPSH)
DO 715 M=MPPSH,II
VFSUWA(M)=VFITW(M)
VFCRSU(M)=AREASU(M)/ARCRY*VFSUCR(M)
SFCSU=SFCSU+VFCRSU(M)
SFITW=SFITW+VFSUWA(M)
ARESUR=ARESUR+AREASU(M)

715 CONTINUE
C BUILD RESISTANCE MATRIX FOR NO SHIELD
RWAL=RWALL
AVFCSU=SFCSU/1MPSH
AVFITW=SFITW/1MPSH
VFWASU=AWALL/ARESUR*AVFITW
VFCRWA=VFCRSU=AVFCSU
VFACR=AWALL/ARCRY*VFCRWA
VFCRSO=1-AVFCSU-VFCRWA
VFWASO=1-VFWASU-VFWACR
RCRSO=1/(ARCRY*VFCRSO)
RCRW=1/(AWALL*VFACR)
RSOW=1/(AWALL*VFWASO)
SR8OSU=0.0
SRITW=0.0
SRCRSU=0.0
DO 799 I=1,1MPSH
M=I+MPPSH=I
VFSUSO(M)=1-VFSUCR(M)=VFSUWA(M)
RSOSU(I)=1.0/AREASU(M)*VFSUSO(M)
RCRSU(I)=1.0/AREASU(M)*VFSUCR(M)
SRCRSU=SNCRSU+1.0/RCRSU(I)
SRITW=SRITW+1.0/RITW(I)
SR8OSU=SR8OSU+1.0/RSOSU(I)

799 CONTINUE
WRITE (JJJ,498)
WRITE (JJJ,661) RCR5O,RCRWA,RSOWA,RSO,RCRY,RCWAL,RSHIED,RCRSH,
3RSOSH
WRITE (JJJ,500)
WRITE (JJJ,682) ((I,RSOSU(I),RCR6U(I),RITW(I)),I=1,1,IMPSH)
WRITE (JJJ,501)
WRITE (JJJ,681) VFSHCR,VFSHSO,VFCSRO,VFCRSH,VFCRWA,VFWASO,VFWASU,
6VFSUA,AVFCSU
WRITE (JJJ,631)
WRITE (JJJ,511) ((I,VFITSH(I),VFITWA(I),VFSCUR(I)),I=MPPSH,II)
DO 802 J=1,IMPSH+3
DO 802 I=1,IMPSH+3
IF(J,GT,IMPSH) GO TO 803
IF(I,GT,IMPSH) GO TO 804
IF (I,NE,J) GO TO 804
RWOS(I,J)=1/RSUR(J)+1/RCRSU(J)+1/RITW(J)+1/RSOSU(J)
GO TO 802
804 CONTINUE
RWOS(I,J)=0,0
RWOS(IMPSH+1,J)=1/RITW(J)
RWOS(IMPSH+2,J)=1/RCR6U(J)
RWOS(IMPSH+3,J)=1/RSOSU(J)
GO TO 802
803 CONTINUE
IF(J,GT,IMPSH+1) GO TO 805
IF(I,GT,IMPSH) GO TO 806
RWOS(I,J)=1/RITW(I)
GO TO 802
806 RWOS(IMPSH+1,J)=SRITW+1/RWAL+1/RCRWA+1/RSOWA
RWOS(IMPSH+2,J)=1/RCRWA
RWOS(IMPSH+3,J)=1/RSOWA
GO TO 802
805 CONTINUE
IF(J,GT,IMPSH+2) GO TO 807
IF(I,GT,IMPSH) GO TO 808
RWOS(I,J)=1/RCRSU(I)
GO TO 802
808 CONTINUE
RWOS(IMPSH+1,J)=1/RCRWA
RWOS(IMPSH+2,J)=SRCRSU+1/RCR+1/RCRWA+1/RCR60
RWOS(IMPSH+3,J)=1/RCR60
GO TO 802
807 CONTINUE
IF(I,GT,IMPSH) GO TO 809
RWOS(I,J)=1/RSOSU(I)
GO TO 802
809 CONTINUE
RWOS(IMPSH+1,J)=1/RSOWA
RWOS(IMPSH+2,J)=1/RCR60
RWOS(IMPSH+3,J)=SRSOSU+1/RSO+1/RCR60+1/RSOWA
GO TO 802
802 CONTINUE
WRITE (JJJ,632)
WRITE (JJJ,680) ((RWOS(I,J),I=1,1,IMPSH+3),J=1,1,IMPSH+3)
551 FORMAT ('0',5X,'I4,9X,3F10,5,3X)
560 FORMAT ('0',5X,'I4,9X,6E10,5x)
571 FORMAT ('0',5X,'I4,9X,6F10,5,3X)
573 FORMAT ('0',5X,'I4,9X,6E10,5,3X)
574 FORMAT ('0',5X,'I4,9X,6F10,5,3X)
575 FORMAT ('0',5X,'I4,9X,6E10,5,3X)
576 FORMAT ('0',5X,'I4,9X,6F10,5,3X)
SUBROUTINE CONFACT(XP,YP,ZP,MPTS,THET,ARRAYX,ARRAYY,ARRAYZ,N,AVGVF)

XP,YP AND ZP = ARRAY OF POINTS ON VIEWING PLANE.
MPTS = NUMBER OF XP,YP, AND ZP POINTS.
THET = ANGLE IN RADIANS BETWEEN X-AXIS AND PLANE
CONTAINING XP,YP, AND ZP POINTS.
IF THET = 0.0, NO ROTATION IS MADE, (AND ONLY ONE ZP REQ'D)
IF THET NOT 0.0, ROTATION IS MADE ABOUT Z AXIS THEN
Y AND Z ARE INTERCHANGED, (MPTS VALUES OF ZP REQ'D)
ARRAYX, ARRAYY, AND ARRAYZ ARE VIEWED POLYGON.
N = NUMBER OF POINTS AROUND POLYGON + 1
AVGVF = VIEW FACTOR FROM XP,YP,ZP PLANE TO ARRAYX,ARRAYY,ARRAYZ
POLYGON.

REAL*8 SUM,PTVF
DIMENSION XPS(100),YPS(100)
DIMENSION XJSAV(200),YJSAV(200),ZJSAV(200)
DIMENSION XP(1),YP(1),ZP(1),ARRAYX(1),ARRAYY(1),ARRAYZ(1),X(200),Y(200),Z(200)
DATA PI,TWOPI/3,14159.6,283186/

CT=COS(THET)
ST=SIN(THET)
NODES=N+1
SUM=0.

C ROTATE IF THET GREATER THAN ZERO.
IF(THET,GT,0.0) GO TO 20
CT=COS(THET)
ST=SIN(THET)
NODES=N+1
SUM=0.
C
    ROTATE IF THET GREATER THAN ZERO.
    IF(THET,CT,0,0) GO TO 20
    DO 10 J=1,N
       XJSAV(J)=ARRAYX(J)
       YJSAV(J)=ARRAYY(J)
       ZJSAV(J)=ARRAYZ(J)
10   CONTINUE
    ZPS=ZP(I)
    DO 12 I=1,MPTS
       XPS(I)=XP(I)
       YPS(I)=YP(I)
12   CONTINUE
    GO TO 30
20   DO 22 J=1,N
       XJSAV(J)=ARRAYX(J)*CT+ARRAYY(J)*ST
       YJSAV(J)=ARRAYZ(J)
       ZJSAV(J)=ARRAYX(J)*ST+ARRAYY(J)*CT
22   CONTINUE
    ZPS=XP(I)*CT+YP(I)*ST
    DO 24 I=1,MPTS
       XPS(I)=XP(I)*CT+YP(I)*ST
       YPS(I)=ZP(I)
24   CONTINUE
30   CONTINUE
    DO 1 I=1,MPTS
    DO 2 J=1,N
       X(J)=XJSAV(J)=XPS(I)
       Y(J)=YJSAV(J)=YPS(I)
2   Z(J)=ZJSAV(J)=ZPS
    PTVF=0.D0
    DO 3 J=1,NODES
       X(J)=X(J)
       Y(J)=Y(J+1)
       Z(J)=Z(J+1)
    Z(J)=Z(J+1)
3   PROD=XJ*YJ+XJ*YJ
    VDOTP=XJ*YJ+YJ*YJ+ZJ*ZJ
    VN=SQRT((YJ*ZJ-ZJ*YJ)**2+(XJ*ZJ-XJ*ZJ)**2+PROD*PROD)
    IF(VN,EQ,0,0) WRITE(6,1000)XP,YP,ZP,THET,(ARRAYX(KK),ARRAYY(KK),
1   ARRAYZ(I),KK=1,N)
1000  FORMAT(10,2X,4F10.4/(2X,3F10.4))
    IF(VN,NEQ,0,0)STOP
    IF(VDOTP,NEQ,0,0) GO TO 6
    THETA=ATAN(VN/VDOTP)
    IF(THETA,LT,0,0) THETA=THETA+PI
    GO TO 7
6    THETA=PI/2.
7   CONTINUE
    COSGAM=PROD/VN
3   PTVF=PTVF+THETA*COSGAM
    PTVF=DABS(PTVF)/TWOPI
1   SUM=SUM+PTVF
    AVGVF=SUM/MPTS
C
APPENDIX 6

NUMERICAL RESULTS
### 1/RESISTANCE MAIN AX WITH SHIELD

<table>
<thead>
<tr>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
<th>R9</th>
<th>R10</th>
<th>R11</th>
<th>R12</th>
<th>R13</th>
<th>R14</th>
<th>R15</th>
<th>R16</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>26</td>
<td>35</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>25</td>
<td>40</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>24</td>
<td>44</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>23</td>
<td>51</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>55</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>62</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>66</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>72</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### I/RSRSU RCRW A R50W A KSU KCHY R.ALL KSHIELD KCRSm RSOSM

<table>
<thead>
<tr>
<th>I</th>
<th>I/RSRSU(I)</th>
<th>I/RCRSU(I)</th>
<th>I/KSH(I)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0608</td>
<td>0.02813</td>
<td>0.00483</td>
</tr>
<tr>
<td>0</td>
<td>0.00159</td>
<td>0.001698</td>
<td>0.00376</td>
</tr>
<tr>
<td>0</td>
<td>0.01049</td>
<td>0.01408</td>
<td>0.00698</td>
</tr>
</tbody>
</table>

### I/KSUSU(I) RCKSU(I) K1W(I)

<table>
<thead>
<tr>
<th>I</th>
<th>I/KSUSU(I)</th>
<th>I/RCKSU(I)</th>
<th>I/K1W(I)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.08840</td>
<td>0.19712</td>
<td>0.45743</td>
</tr>
<tr>
<td>2</td>
<td>0.13183</td>
<td>0.17219</td>
<td>0.13334</td>
</tr>
<tr>
<td>3</td>
<td>0.12730</td>
<td>0.15462</td>
<td>0.11247</td>
</tr>
<tr>
<td>4</td>
<td>0.11740</td>
<td>0.14146</td>
<td>0.10051</td>
</tr>
<tr>
<td>5</td>
<td>0.10579</td>
<td>0.13123</td>
<td>0.09272</td>
</tr>
<tr>
<td>6</td>
<td>0.10444</td>
<td>0.12305</td>
<td>0.08725</td>
</tr>
<tr>
<td>7</td>
<td>0.08413</td>
<td>0.11637</td>
<td>0.07519</td>
</tr>
<tr>
<td>8</td>
<td>0.07515</td>
<td>0.11083</td>
<td>0.06055</td>
</tr>
<tr>
<td>9</td>
<td>0.06748</td>
<td>0.10616</td>
<td>0.07755</td>
</tr>
<tr>
<td>10</td>
<td>0.06086</td>
<td>0.10219</td>
<td>0.07549</td>
</tr>
<tr>
<td>11</td>
<td>0.05524</td>
<td>0.09879</td>
<td>0.07375</td>
</tr>
<tr>
<td>12</td>
<td>0.05042</td>
<td>0.09583</td>
<td>0.07226</td>
</tr>
<tr>
<td>SURFACE ELEMENT</td>
<td>VFSUSH</td>
<td>VFSUMA</td>
<td>VFSUCr</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>9</td>
<td>0.86228</td>
<td>0.11772</td>
<td>0.27316</td>
</tr>
<tr>
<td>10</td>
<td>0.87166</td>
<td>0.35897</td>
<td>0.27797</td>
</tr>
<tr>
<td>11</td>
<td>0.86444</td>
<td>0.36300</td>
<td>0.27860</td>
</tr>
<tr>
<td>12</td>
<td>0.85962</td>
<td>0.38962</td>
<td>0.27683</td>
</tr>
<tr>
<td>13</td>
<td>0.85611</td>
<td>0.38714</td>
<td>0.27355</td>
</tr>
<tr>
<td>14</td>
<td>0.85343</td>
<td>0.37978</td>
<td>0.26949</td>
</tr>
<tr>
<td>15</td>
<td>0.85140</td>
<td>0.36987</td>
<td>0.26441</td>
</tr>
<tr>
<td>16</td>
<td>0.84956</td>
<td>0.35874</td>
<td>0.25912</td>
</tr>
<tr>
<td>17</td>
<td>0.84812</td>
<td>0.34719</td>
<td>0.25360</td>
</tr>
<tr>
<td>18</td>
<td>0.84689</td>
<td>0.33568</td>
<td>0.24795</td>
</tr>
<tr>
<td>19</td>
<td>0.84584</td>
<td>0.32449</td>
<td>0.24226</td>
</tr>
<tr>
<td>20</td>
<td>0.84492</td>
<td>0.31375</td>
<td>0.23658</td>
</tr>
<tr>
<td>Radial Velocity Component U</td>
<td>Axial Velocity Component W</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------------</td>
<td>----------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.002 0.004 0.006 0.007 0.008 0.009 0.010 0.010 0.010 0.010 0.009 0.008 0.007 0.006 0.005 0.004 0.003 0.002 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.005 0.009 0.013 0.016 0.019 0.021 0.023 0.023 0.022 0.020 0.018 0.015 0.012 0.009 0.006 0.003 0.002 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.007 0.013 0.019 0.024 0.028 0.031 0.033 0.034 0.033 0.032 0.029 0.026 0.022 0.018 0.013 0.009 0.005 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.006 0.012 0.018 0.023 0.027 0.031 0.032 0.032 0.030 0.028 0.025 0.021 0.017 0.012 0.008 0.004 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.006 0.012 0.018 0.022 0.026 0.029 0.030 0.030 0.028 0.026 0.024 0.020 0.016 0.012 0.008 0.004 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.005 0.010 0.015 0.019 0.022 0.025 0.026 0.027 0.025 0.021 0.018 0.014 0.011 0.007 0.004 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.004 0.009 0.013 0.017 0.022 0.023 0.024 0.024 0.022 0.021 0.019 0.016 0.013 0.010 0.007 0.004 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.004 0.007 0.010 0.013 0.016 0.017 0.017 0.016 0.014 0.012 0.011 0.010 0.009 0.008 0.006 0.004 0.002 0.001 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.002 0.005 0.007 0.009 0.011 0.012 0.013 0.013 0.011 0.011 0.010 0.009 0.008 0.006 0.005 0.004 0.003 0.002 0.001</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.001 0.002 0.003 0.004 0.005 0.006 0.006 0.006 0.006 0.006 0.006 0.006 0.006 0.005 0.005 0.004 0.003 0.002 0.001</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td>0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x</td>
<td>E_bS</td>
<td>E_bI</td>
<td>J_S</td>
</tr>
<tr>
<td>----</td>
<td>-------</td>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>1</td>
<td>11.59053</td>
<td>11.59053</td>
<td>7.82771</td>
</tr>
<tr>
<td>2</td>
<td>11.84485</td>
<td>11.84485</td>
<td>7.18841</td>
</tr>
<tr>
<td>3</td>
<td>12.06193</td>
<td>12.06193</td>
<td>7.19395</td>
</tr>
<tr>
<td>4</td>
<td>12.30141</td>
<td>12.30141</td>
<td>7.23952</td>
</tr>
<tr>
<td>5</td>
<td>12.50088</td>
<td>12.50088</td>
<td>7.30086</td>
</tr>
<tr>
<td>6</td>
<td>12.67760</td>
<td>12.67760</td>
<td>7.37016</td>
</tr>
<tr>
<td>7</td>
<td>12.92920</td>
<td>12.92920</td>
<td>7.44010</td>
</tr>
<tr>
<td>8</td>
<td>12.95441</td>
<td>12.95441</td>
<td>7.56888</td>
</tr>
<tr>
<td>9</td>
<td>13.03344</td>
<td>13.03344</td>
<td>7.58385</td>
</tr>
<tr>
<td>10</td>
<td>13.12933</td>
<td>13.12933</td>
<td>7.62463</td>
</tr>
<tr>
<td>11</td>
<td>13.17619</td>
<td>13.17619</td>
<td>7.67122</td>
</tr>
<tr>
<td>12</td>
<td>13.27143</td>
<td>13.27143</td>
<td>7.71840</td>
</tr>
<tr>
<td>13</td>
<td>0.04472</td>
<td>0.04472</td>
<td>2.78155</td>
</tr>
<tr>
<td>14</td>
<td>5.20729</td>
<td>5.20729</td>
<td>1.94099</td>
</tr>
<tr>
<td>15</td>
<td>0.04472</td>
<td>0.04472</td>
<td>0.71513</td>
</tr>
<tr>
<td>16</td>
<td>0.00000</td>
<td>0.00000</td>
<td>2.07275</td>
</tr>
<tr>
<td>17</td>
<td>0.00000</td>
<td>0.00000</td>
<td>6.51238</td>
</tr>
</tbody>
</table>

### DELT RADIATION

<table>
<thead>
<tr>
<th>x</th>
<th>DELT RADIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>0.1236E+02</td>
</tr>
<tr>
<td>8</td>
<td>0.1188E+02</td>
</tr>
<tr>
<td>9</td>
<td>0.1125E+02</td>
</tr>
<tr>
<td>10</td>
<td>0.1395E+02</td>
</tr>
<tr>
<td>11</td>
<td>0.1496E+02</td>
</tr>
<tr>
<td>12</td>
<td>0.1515E+02</td>
</tr>
<tr>
<td>13</td>
<td>0.1558E+02</td>
</tr>
<tr>
<td>14</td>
<td>0.1584E+02</td>
</tr>
<tr>
<td>15</td>
<td>0.1621E+02</td>
</tr>
<tr>
<td>16</td>
<td>0.1635E+02</td>
</tr>
<tr>
<td>17</td>
<td>0.1641E+02</td>
</tr>
<tr>
<td>18</td>
<td>0.1647E+02</td>
</tr>
<tr>
<td>19</td>
<td>0.1649E+02</td>
</tr>
<tr>
<td>20</td>
<td>0.1650E+02</td>
</tr>
</tbody>
</table>

### DELT CONVECTION

<table>
<thead>
<tr>
<th>x</th>
<th>DELT CONVECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>0.1236E+02</td>
</tr>
<tr>
<td>8</td>
<td>0.1188E+02</td>
</tr>
<tr>
<td>9</td>
<td>0.1125E+02</td>
</tr>
<tr>
<td>10</td>
<td>0.1395E+02</td>
</tr>
<tr>
<td>11</td>
<td>0.1496E+02</td>
</tr>
<tr>
<td>12</td>
<td>0.1515E+02</td>
</tr>
<tr>
<td>13</td>
<td>0.1558E+02</td>
</tr>
<tr>
<td>14</td>
<td>0.1584E+02</td>
</tr>
<tr>
<td>15</td>
<td>0.1621E+02</td>
</tr>
<tr>
<td>16</td>
<td>0.1635E+02</td>
</tr>
<tr>
<td>17</td>
<td>0.1641E+02</td>
</tr>
<tr>
<td>18</td>
<td>0.1647E+02</td>
</tr>
<tr>
<td>19</td>
<td>0.1649E+02</td>
</tr>
<tr>
<td>20</td>
<td>0.1650E+02</td>
</tr>
</tbody>
</table>