



Calhoun: The NPS Institutional Archive

Theses and Dissertations

Thesis Collection

1991

Numerical field model simulation of full-scale fire tests in a closed spherical/cylindrical vessel using advanced computer graphics techniques.

McCarthy, Timothy G.

Monterey, California. Naval Postgraduate School



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>



NAVAL POSTGRADUATE SCHOOL Monterey, California



THESIS

NUMERICAL FIELD MODEL SIMULATION OF
FULL-SCALE FIRE TESTS IN A CLOSED
SPHERICAL/CYLINDRICAL VESSEL USING
ADVANCED COMPUTER GRAPHICS TECHNIQUES

by

Timothy G. McCarthy

SEPTEMBER 1991

Thesis Advisor:

M.D. Kelleher

Approved for public release: Distribution is unlimited

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS			
a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release: Distribution is unlimited			
b. DECLASSIFICATION/DOWNGRADING SCHEDULE						
PERFORMING ORGANIZATION REPORT NUMBER(S)			5. MONITORING ORGANIZATION REPORT NUMBER(S)			
a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School		6b. OFFICE SYMBOL (If applicable) ME	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School			
c. ADDRESS (City, State and ZIP Code) Monterey, CA 93943-5000			7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000			
a. NAME OF FUNDING/SPONSORING ORGANIZATION		8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER			
c. ADDRESS (City, State, and ZIP Code)			10. SOURCE OF FUNDING NUMBER			
			PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
1. TITLE (Include Security Classification) NUMERICAL FIELD MODEL SIMULATION OF FULL-SCALE FIRE TESTS IN A CLOSED SPHERICAL/CYLINDRICAL VESSEL USING ADVANCED COMPUTER GRAPHICS TECHNIQUES						
2. PERSONAL AUTHORS TIMOTHY G. McCARTHY						
3a. TYPE OF REPORT Master's Thesis		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Year, Month, Day) SEPTEMBER 1991		15. PAGE COUNT 157
6. SUPPLEMENTARY NOTATION The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government						
7. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block numbers)			
FIELD	GROUP	SUB-GROUP	field model, fire simulation, fire modeling, numerical fire model, fires in enclosed vessels, enclosed fires			
9. ABSTRACT (Continue on reverse if necessary and identify by block numbers) Personnel and equipment casualties, caused by shipboard fires have adversely affected overall readiness of the U.S. Navy for centuries. Understanding the phenomena of fire in enclosed spaces, such as those found on surface ships and submarines, will greatly enhance the Navy's ability to combat or prevent them. This computer model was developed for use in conjunction with Fire-1, an experimental fire chamber test facility at the Naval Research Laboratory in Washington, D.C. It is a three-dimensional finite difference model which includes the phenomena of conduction, turbulence, global pressure correction, surface radiation and strong buoyancy flows. Given specific data on heat release, it predicts velocities, temperatures, pressures, densities and viscosities throughout its geometry. It has been reasonably validated by comparison with experiments in Fire-1. Advanced graphics techniques, such as color contouring and three-dimensional vector field plotting, have been applied to make output data more informative. This model, if easily modified to more specific geometries, may become a useful tool for naval architects in the design of the fire safe ship.						
10. DISTRIBUTION/AVAILABILITY OF ABSTRACT XX UNCLASSIFIED/UNLIMITED _ SAME AS RPT _ DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION unclassified			
2a. NAME OF RESPONSIBLE INDIVIDUAL M.D. Kelleher			22b. TELEPHONE (Include Area Code) (408) 646-2530		22c. OFFICE SYMBOL ME/Kk	

Approved for public release: Distribution is unlimited

Numerical Field Model Simulation of Full-Scale Fire Tests
in a Closed Spherical/Cylindrical Vessel
Using Advanced Computer Graphics Techniques

by

Timothy G. McCarthy
Lieutenant, United States Navy
B.S.M.E., University of Rochester, Rochester, N.Y., 1984

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE
IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

SEPTEMBER 1991



ABSTRACT

Personnel and equipment casualties, caused by shipboard fires have adversely affected overall readiness of the U.S. Navy for centuries. Understanding the phenomena of fire in enclosed spaces, such as those found on surface ships and submarines, will greatly enhance the Navy's ability to combat or prevent them. This computer model was developed for use in conjunction with Fire-1, an experimental fire chamber test facility at the Naval Research Laboratory in Washington, D.C. It is a three-dimensional finite difference model which includes the phenomena of conduction, turbulence, global pressure correction, surface radiation and strong buoyancy flows. Given specific data on heat release, it predicts velocities, temperatures, pressures, densities and viscosities throughout its geometry. It has been reasonably validated by comparison with experiments in Fire-1. Advanced graphics techniques, such as color contouring and three-dimensional vector field plotting, have been applied to make output data more informative. This model, if easily modified to more specific geometries, may become a useful tool for naval architects in the design of the fire safe ship.

TABLE OF CONTENTS

I.	INTRODUCTION	1
	A. BACKGROUND	1
	B. COMPUTER MODELING	3
	C. FIRE-1 TEST FACILITY	5
	D. THE COMPUTER PROGRAM	8
II.	DESCRIPTION OF NUMERICAL MODEL	11
	A. GOVERNING EQUATIONS	11
	B. INITIAL AND BOUNDARY CONDITIONS	13
	1. Initial Conditions	13
	2. Boundary Conditions	13
	C. MODELS OF PHYSICAL PHENOMENA	14
	1. Wall Conduction Model	14
	2. Turbulence Model	15
	3. Surface Radiation Model	17
III.	FINITE VOLUME CALCULATIONS	18
	A. INTRODUCTION	18
	B. CONTROL VOLUME ANALYSIS	20
	C. INTEGRATION OF THE CONSERVATION EQUATIONS	23
	D. DISCRETIZATION OF THE CONTINUITY EQUATION	24
	E. DISCRETIZATION OF THE ENERGY EQUATION	27
	F. DISCRETIZATION OF THE MOMENTUM EQUATION	39
	G. PRESSURE CORRECTION	47
	1. Global Pressure Correction	47
	2. Local Pressure Correction	49
IV.	NUMERICAL PROCESS	52
	A. INTRODUCTION	52
	B. SOLUTION PROCESS	55
	C. GRAPHICAL ANALYSIS	57
V.	CONCLUSIONS AND RECOMMENDATIONS	65
	A. CONCLUSIONS	65
	B. RECOMMENDATIONS	65
	APPENDIX	67

LIST OF REFERENCES 142
INITIAL DISTRIBUTION LIST 146

LIST OF TABLES

TABLE 4.1	MODEL PARAMETERS	52
TABLE 4.2	ADDITIONAL MODEL PARAMETERS	55

LIST OF FIGURES

Figure 1.1	Drawing of Fire-1 Test Vessel.	6
Figure 1.2	Sensor Locations of Fire-1 Test Vessel.	7
Figure 3.1	Basic Spherical Cell.	21
Figure 3.2	Basic Cylindrical Cell.	21
Figure 3.3	Two Dimensional Cell.	22
Figure 3.4	Two Dimensional Staggered Cell.	22
Figure 3.5	One Dimensional Quadratic Interpolation Scheme.	29
Figure 3.6	Calculation Cells for a Uniform Rectangular Grid.	33
Figure 4.1	Front View of Computer Model (YZ-Plane)	53
Figure 4.2	Side View of Computer Model (XY-Plane)	54
Figure 4.3	Temperature Profiles at 30 Seconds	59
Figure 4.4	Velocity Profile at 30 Seconds	60
Figure 4.5	Temperature Profile at 60 Seconds	61
Figure 4.6	Velocity Profile at 60 Seconds	62
Figure 4.7	Temperature Profile at 90 Seconds	63
Figure 4.8	Velocity Profile at 90 Seconds	64

LIST OF SYMBOLS AND ABBREVIATIONS

A	Area
A	Finite Difference Coefficients
ARU_	Source Term Variable
AU_	Source Term Variable
C_	Coefficients for Control Volume
C_M	Coefficients for Control Volume
C_P	Coefficients for Control Volume
COND_1	Coefficients for Control Volume
C _{pm}	Mean Isobaric Heat Capacity
CURV	Curvature Terms
CURVN	Orthogonal Curvature Terms
F _{A₁-A_j}	View Factor for Radiation Emitted by Surface i and Incident Upon Surface j
G	Gravitational Acceleration
G	Mass Flux Rate
G	Term Used in Radiation Model
g	Curvilinear Base Vector
g ₁	Scaling Term
g _{1j}	Covariant Metric Tensor
g ⁱ¹	Contravariant Metric Tensor
H	Mixing Length Parameter
h	Scale Factor
h	Convective Heat Transfer Coefficient
H	Enthalpy
J	Total Heat Flux
K	Adjustable Constant
k	Thermal Conductivity
M	Momentum Flux
m	Rate of Change
n	Normal Direction Toward the Vessel Center
P	Pressure
Pr	Prandtl Number
Pr _t	Turbulent Prandtl Number

q	Heat Flux
q_r	Thermal Radiation Energy
R	Universal Gas Constant
$R_$	Source Term Variable
$RR_$	Source Term Variable
Ri	Richardson Number
r	Distance Between Two Surfaces
S_f	Source Term
S_{hs}	Heat Source
S_{mp}	Mass Source Term
T	Temperature
t	Time
u	Velocity
V	Volume
VIS	Local Viscosity
X	Length in X-Direction (In QUICK Scheme)

GREEK LETTERS

β	Angles Formed by Radiation Surface Normals
χ	Term Used in Radiation Model
δ_{ij}	Kronecker Delta
ϵ	Emissivity
ϕ	Dissipation Function
μ	Dynamic Viscosity
θ	Directions, θ , r , and ϕ or Z
ρ	Fluid Density
σ	Stress
σ	Stefan-Boltzmann Constant
ψ	Term Used in Radiation Model

SUBSCRIPTS

B	Control Volume to the Back
b	Back Control Volume Face
E	Control Volume to the East
EQ	Equilibrium

e East Control Volume Face
 eff Effective
 F Control Volume to the Front
 f Front Control Volume Face
 g Global
 N Control Volume to the North
 n North Control Volume Face
 o Reference
 p Present Cell
 R Reference
 S Control Volume to the South
 s South Control Volume Face
 s Vessel Wall
 W Control Volume to the West
 w West Control Volume Face
 ,i derivative with respect to i
 ,t derivative with respect to time

SUPERSCRIPTS

n Future Value
 n-1 Present Value
 * Estimated Value
 * Ventilation Values
 ' Correction
 ^ Prior Value

I. INTRODUCTION

A. BACKGROUND

Annually, the effects of fires on Naval forces are particularly devastating. Ships may be removed from service for repairs which incur costs that may run into the tens of millions of dollars. Personnel casualties, ship down time, equipment repair and replacement all result in a loss of overall readiness of our fleets. The prevention of shipboard fires is of the utmost importance to today's Navy. The understanding of the phenomena of fire, especially in the enclosed spaces found aboard ship, is the first step toward its control and prevention.

The study of fire propagation requires the combined knowledge of fluid dynamics, mass and heat transfer, and combustion. Research into the mechanics of fire and prediction of its behavior will aid engineers in reducing the probability of its ignition and propagation.

There are a number of ways to conduct this research. The most obvious is experimental. But, fires aboard ships are very complex. Often they are in enclosed airtight spaces which allow pressures to build. These spaces may be full of electronic equipment, flammables or toxic substances. Their accessibility may be extremely limited, hampering efforts to

combat fires. An experiment that can accurately account for all these complexities becomes very expensive.

At the Naval Research Laboratory in Washington, D.C., the Navy has built Fire-1, a large pressure vessel designed to model fires aboard submarines, or closed compartments and tanks found on surface ships. It allows fires to be studied under the unique conditions experienced in shipboard fires.

Another method for conducting fire research is the use of a computer model. As computers get faster and can allow for large amounts of data storage, researchers are able to thoroughly model fire phenomena and predict future behavior without the continuous expensive full scale testing of Fire-1. Fires may be modeled by the numerical solution of the governing equations. These models are then verified by the existing data from experiments. With an accurate computer model, several options are available. More complex geometries may be incorporated for specific areas of interest. Entire models of ships may someday be developed to show areas of susceptibility in design. Effects of firefighting methods may be accurately predicted. The savings in running computer codes versus full scale testing are considerable.

Also, now that a high speed VAXSTATION 3100 SPX/RJ19 Model 38 workstation may be dedicated to this particular simulation, computing costs may be minimized. The current code requires approximately 1.0 hours of VAXSTATION CPU per second of fire time.

B. COMPUTER MODELING

Field modeling uses difference forms of the conservation equations of mass, momentum, energy and species. These are used to calculate temperature, velocity, pressure, viscosity and density at specific points in the volume of interest. This volume, being the compartment studied, is broken down to finite volume elements. The conservation equations are solved at this level for discrete time steps from a known initial condition. Additional models of physical effects such as radiation, turbulence, and wall conduction are included to increase the simulation's validity. This method requires large amounts of computer memory and high speed processors.

Much research has been done previously and has provided the basis for this thesis. At the University of Notre Dame [Refs. 1 and 2] work has been conducted involving aircraft cabin fires using a two dimensional finite difference field model which predicts velocity, temperature and smoke concentration inside the passenger area of an aircraft. Nicolette *et al.* [Ref. 3] developed a two dimensional model of transient cooling by natural convection. It utilized a fully transient, semi-implicit upwind differencing scheme and global pressure correction that was verified experimentally.

More recent [Refs. 4 through 12] studies have developed numerical solutions for three dimensional rectangular enclosures in which non-linear partial differential equations were solved by finite difference methods. Models for three

dimensional cylindrical coordinate buoyant flows [Refs. 13 through 19] have also been developed, and deal mainly with horizontal annuli with differential temperatures specified at inner and outer cylindrical walls. Smutek et al. [Ref. 18] studied buoyant flows in horizontal cylinders with differentially heated ends at low Rayleigh numbers ($74 \leq Ra \leq 18700$). Yang et al. [Ref. 19] conducted a similar study but with high Rayleigh numbers ($10^4 \leq Ra \leq 10^7$).

Studies have also been done on methods for decoupling the pressure terms from the Navier-Stokes Equation. The stream function-vorticity formulation has been used [Refs. 13 through 18] to calculate natural convection in various geometries. There are problems with this method such as instability at high Rayleigh numbers. Yang et al. [Ref. 19] address this problem and suggest using a primitive variable formulation when using arbitrary orthogonal coordinates.

Natural convection in spherical annuli was studied by Ozoë [Ref. 20] utilizing velocity-vector formulation. Field models involving prediction of fires in enclosures have been studied by Baum and Rehm [Refs. 21 through 24]. These include time dependent Boussinesq equations to simulate three dimensional buoyant convection and smoke aerosol coagulation. Field models involving three dimensional enclosures and employing the Boussinesq approximation, were studied by Bagnaro et al. [Ref. 25] and by Markatos and Pericleous [Ref. 26].

In this thesis, the numerical method developed by Yang et al. [Ref. 19] using primitive variable finite difference discretization in generalized orthogonal coordinates is employed. This method can handle complex geometries and has the numerical stability characteristic of primitive variable formulation.

C. FIRE-1 TEST FACILITY

An experimental test facility called Fire-1, has been constructed at the Naval Research Laboratory to study the behavior of fires in enclosed spaces found on submarines and surface ships. Since the computer code presented in this thesis models the geometry of Fire 1, this section contains a brief description of that facility. More information may be obtained from Alexander et al. [Ref. 27]. Figure 1.1 shows the basic layout. Fire-1 is a cylindrical pressure vessel with hemispherical endcaps. It is constructed of 3/8-inch ASTM 295 Grade C steel and can withstand internal pressures up to 89.7 psi and temperatures of 450°F. Its total length is 46.6 feet long. The cylinder and endcap radii are both 9.6 feet. Rupture discs are placed at each endcap to prevent failure due to overpressurization.

Figure 1.2 shows the instrumentation layout. An array of chromel-alumel thermocouples with ceramic insulation and stainless steel jackets, are placed near each endcap. Additional thermocouples are placed on the chamber walls, both

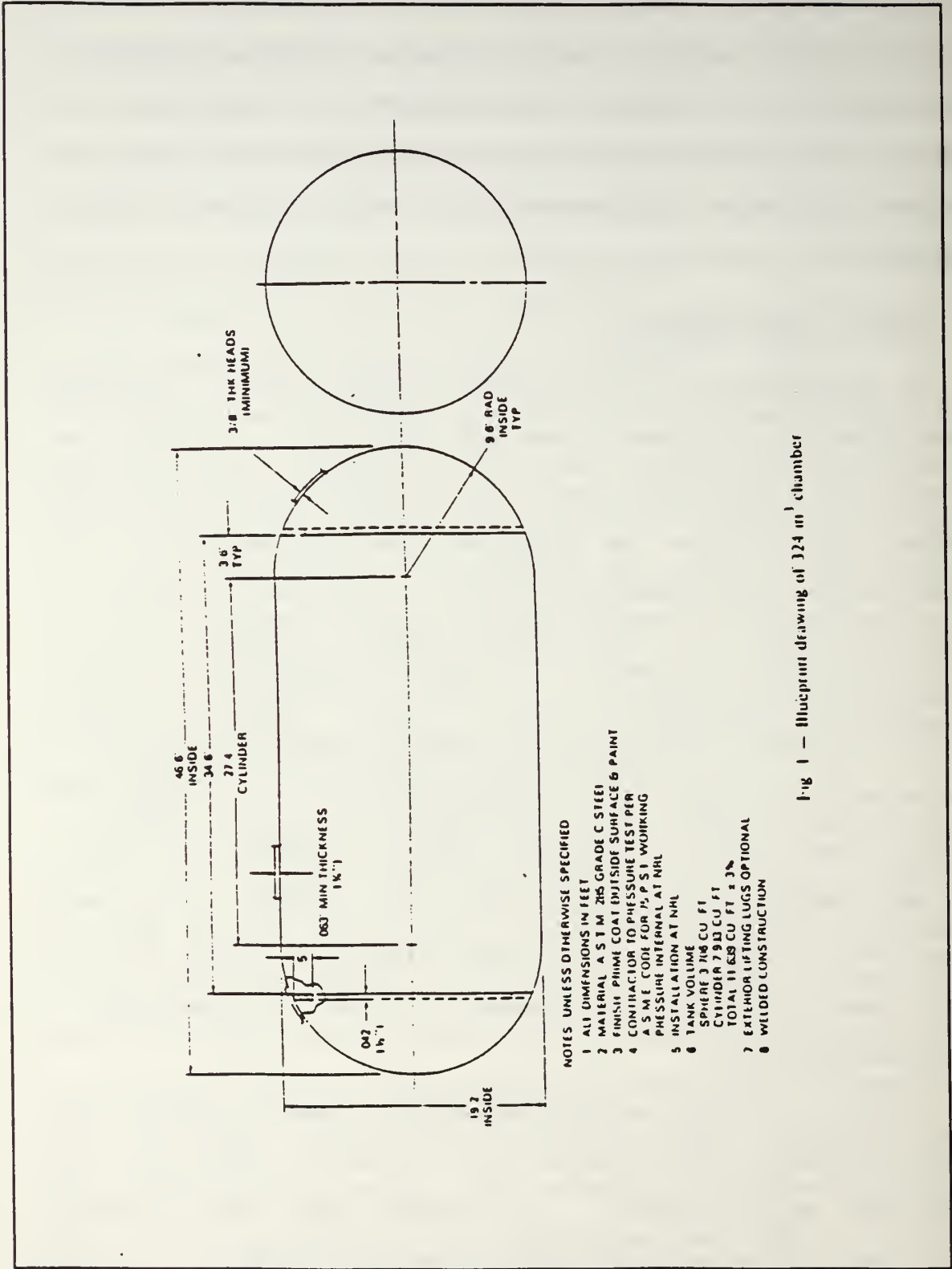


Fig 1 - Blueprint drawing of 324 m³ chamber

Figure 1.1 Drawing of Fire-1 Test Vessel.

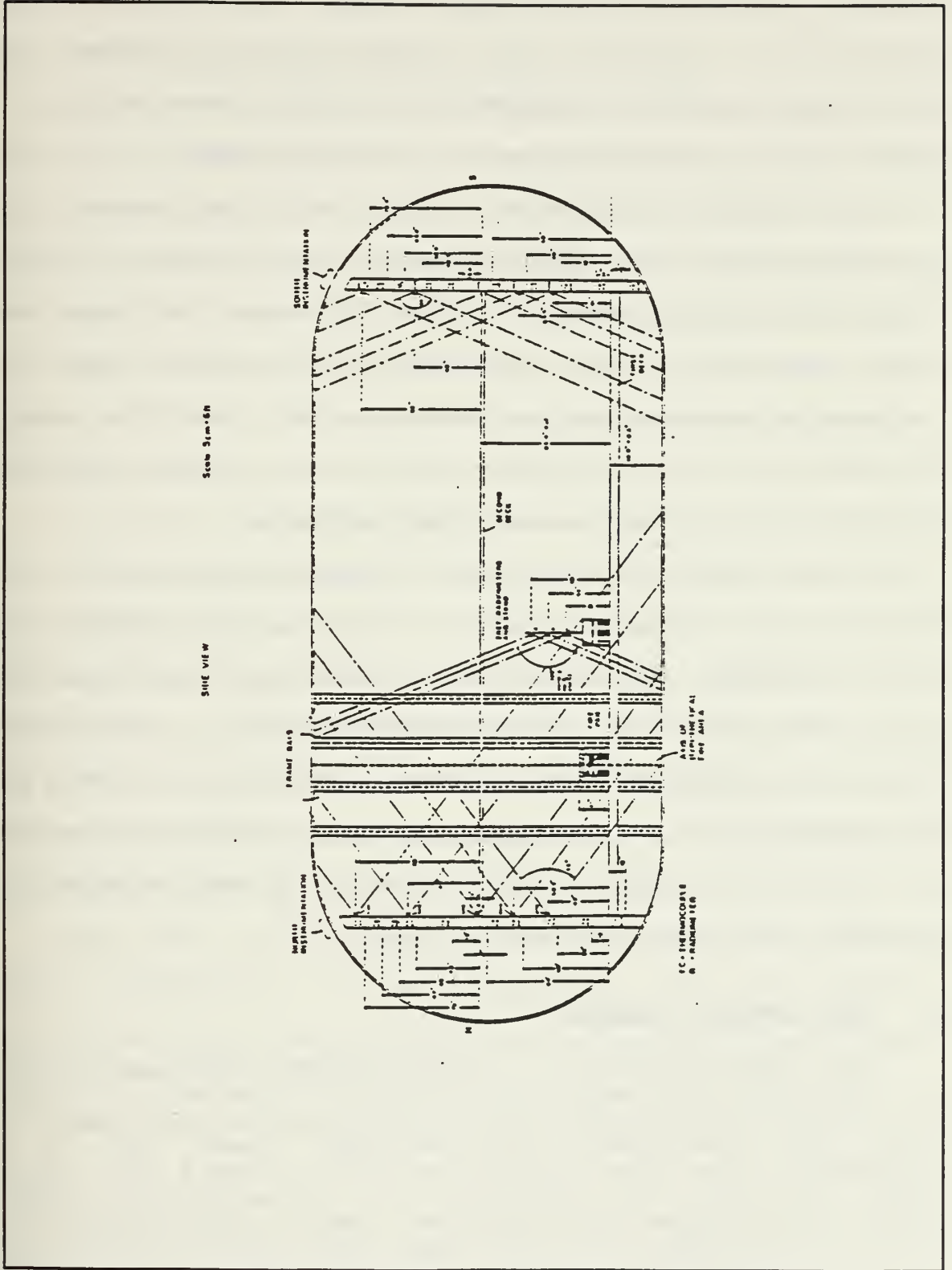


Figure 1.2 Sensor Locations of Fire-1 Test Vessel.

inside and out, to monitor inside and outside wall temperatures. A specific test might call for placement of extra thermocouples or radiometers at various other locations. These are arranged as required by the experiment.

Burn rate data is obtained using round, tapered edge fire pans of various cross sectional areas, and a constant level, liquid fuel supply system. To date, this data has been the least accurate in the experiment. The system and its calibration are described by Alexander *et al.* [Ref. 27]. Smoke concentration can be measured using video cameras, particle analysis and obscuration with laser detectors.

To more completely represent shipboard compartments, the facility has a number of features. First is the installation of two removable decks, one at the midheight, the other at three feet above the bottom. Either grated or solid deck plating is used depending on desired configuration. Second is the installation of a nitrogen pressurization system used as an extinguishing agent. Its performance is being tested for possible use combatting actual fires.

D. THE COMPUTER PROGRAM

This computer model is a joint project undertaken by the Naval Postgraduate School and the University of Notre Dame. It represents a low cost alternative to full scale test using Fire-1. With proper modifications, used in conjunction with

Fire-1, it will test effectiveness of damage-control systems and evaluate new ship designs.

In the work by Nies [Ref. 28], the code was based on a rectangular geometry with the volume identical to Fire-1. This was a three dimensional, finite volume model using primitive variables. Turbulence, wall conduction, and a global pressure correction factor were also included. Due to the unreliability of burn rate data, Nies [Ref. 28] devised a scheme for computing a heat release rate by using experimental pressure curves as input.

The actual geometry of Fire-1 was employed by Raycraft [Ref. 29]. Using its spherical/cylindrical coordinate system and detailed formulation of radiation surface view factors, global pressure correction, conduction and turbulence, the code created an extremely viable model for use with Fire-1. There were the continued problems with simulating the heat release data which were partially resolved by numerically fitting experimental burn rate data available.

Houck [Ref. 31] included a model which simulated internal forced circulation. It was compared to data run without circulation and it was concluded that circulation had minimal effects on the overall velocity and temperature profiles.

In this thesis, advanced three dimensional and color graphics techniques are used to present data generated using the previously developed codes. Using the VAXSTATION 3100 SPX and the software CA-DISSPLA [Ref. 31] the data is presented in

a more informative fashion. Color graphics are used to present isotherm profiles and three dimensional vector fields will represent velocity profiles.

II. DESCRIPTION OF NUMERICAL MODEL

A. GOVERNING EQUATIONS

The model is based on the system of conservation equations which govern the behavior of fluid flow and heat transfer in gases. These equations are in differential form and are presented in generalized curvilinear coordinates using standard tensor notation. Nies [Ref. 28] based his model on rectangular geometry using Cartesian coordinates. Raycraft [Ref. 29] refined the model to describe the exact geometry of Fire-1 and included surface radiation. Houck [Ref. 30] described the transformation to curvilinear coordinates, used by Yang et al. [Ref. 19], in detail and the following forms of the governing equations are obtained.

The equation of continuity is:

$$\rho_t + \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left\{ \sqrt{g} \rho \frac{u^i}{h_i} \right\} = 0 \quad (2.1)$$

The energy equation becomes:

$$\begin{aligned} & (\rho C_{pm} T)_t + \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left\{ \sqrt{g} \rho C_{pm} u^i \frac{T}{h_i} \right\} \\ & = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left\{ \sqrt{g} \frac{k_{eff} T_{,i}}{h_i^2} \right\} + S_f \end{aligned} \quad (2.2)$$

where the source term, S_f is:

$$S_f = \mu \Phi + P \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left\{ \sqrt{g} \frac{u^i}{h_i} \right\} + S_{hs} \quad (2.3)$$

and the dissipation term is:

$$\begin{aligned} \Phi = & 2 \left\{ \left(\frac{u^i}{h_i} \right)_{,j} \right\}^2 \delta_{ij} \\ & + \left\{ \left(\frac{u^i}{h_i} \right)_{,j} (1 - \delta_{ij}) \right\}^2 - \frac{2}{3} \left\{ \left(\frac{u^i}{h_i} \right)_{,i} \right\}^2 \end{aligned} \quad (2.4)$$

S_{hs} is the heat source term which is zero everywhere except nodes at the fire's location and δ_{ij} is the Kronecker Delta.

The momentum equation becomes:

$$\begin{aligned} & (\rho u^i)_{,t} + \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left\{ \sqrt{g} \frac{u^i u^j}{h_j} \right\} \\ & = -\frac{P_{,i}}{h_i} + \rho G^i + \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^j} \left\{ \frac{\sqrt{g} \sigma_i^j}{h_j} \right\} \\ & - \frac{1}{h_i h_j} \frac{\partial h_i}{\partial \theta^j} (\rho u^i u^j - \sigma_i^j) + \frac{1}{h_i h_j} \frac{\partial h_j}{\partial \theta^i} (\rho u^j u^i - \sigma_i^j) \end{aligned} \quad (2.5)$$

where the stress tensor is:

$$\begin{aligned} \sigma_i^j = & \mu_{eff} \left\{ \frac{h_j}{h_i} \frac{\partial}{\partial \theta^i} \left(\frac{u^j}{h_j} \right) \right. \\ & \left. + \frac{h_i}{h_j} \frac{\partial}{\partial \theta^j} \left(\frac{u^i}{h_i} \right) + \frac{\delta_{ij}}{h_i h_j} \frac{\partial q_{ll}}{\partial \theta^m} \left(\sqrt{g} \frac{u^m}{h_m} \right) \right\} \end{aligned} \quad (2.6)$$

Effective conductivity k_{eff} and dynamic viscosity μ_{eff} include both laminar and turbulent terms. Additional terms found in

the momentum equation are due to coriolis and centrifugal effects.

The equations of state remain unchanged through coordinate transformations and are given as:

$$P = \rho RT \quad (2.7)$$

$$\mathcal{H} = C_{pm} (T - T_R) \quad (2.8)$$

B. INITIAL AND BOUNDARY CONDITIONS

In order to solve this system of differential equations, boundary and initial conditions must be determined and applied.

1. Initial Conditions

The initial conditions for the model are determined from conditions present just prior to ignition in Fire-1. The air inside is totally at rest. The temperature is equal to ambient temperature and is assumed uniform throughout. Therefore, in the model, the entire velocity field is set to zero and the non-dimensional temperature field is set to 1.0 which corresponds to ambient temperature. Pressure and density distributions are at static equilibrium.

2. Boundary Conditions

Since the vessel wall is a solid boundary which is nonporous, the velocities, both normal and tangential to the wall, are zero. Mass flux across the wall is also zero. The temperature of the wall is equal to the temperature of the

fluid at the interface. Conservation of energy must also be met at the interface. The following three equations summarize wall boundary conditions:

$$u_{surf}^i = 0 \quad (2.9)$$

$$T_{fluid} = T_{solid} \quad (2.10)$$

$$q_r - k_f \frac{\partial T}{\partial n} \Big|_f = -k_s \frac{\partial T}{\partial n} \Big|_{solid} \quad (2.11)$$

where q_r is the heat flux arriving at the solid/fluid interface and n is the normal direction of the surface into the enclosure. There is conduction through the wall and convection from outer surface to ambient temperature.

Due to singularities occurring at $r=0$ in cylindrical/spherical coordinates, special care must be taken at the origin. Yang et al. [Ref. 19:pp. 167-168] discuss methods for addressing this problem. In this model, two consecutive control volumes are placed at $r=0$ and continuity is applied.

C. MODELS OF PHYSICAL PHENOMENA

1. Wall Conduction Model

This model calculates heat loss from the vessel through the walls to the environment. It assumes one dimension, unsteady heat flow and constant convective heat

transfer coefficient at the wall's exterior. The energy equation is:

$$(\rho_s C_{ps} T)_t = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} (\sqrt{g} k_s T_{,j} g^{ij}) + S \quad (2.12)$$

2. Turbulence Model

The turbulence model is a simple algebraic method used to predict mean flow quantities for incompressible boundary layer flows. Developed by Nee and Liu [Ref. 33], the model determines the effective viscosity in recirculating buoyant flows with large variations in turbulence levels. The equation, transformed into generalized curvilinear coordinates, is:

$$\frac{\mu_{eff}}{\mu_o} = 1 + \frac{\left(\frac{\ell}{H}\right)^2 \sqrt{\left(\frac{1}{H_j} \frac{\partial u^i}{\partial \theta^j}\right) (1 - \delta_i^j)}}{2 + \frac{Ri}{Pr_t}} \quad (2.13)$$

where ℓ/H is a non-dimensional mixing length parameter given as:

$$\frac{\ell}{H} = K \left\{ \frac{\sqrt{u^i u^i}}{\sqrt{\sum_{ij} \left(\frac{1}{h_j} \frac{\partial u^i}{\partial \theta^i} \right)^2}} + \frac{\sqrt{\sum_{ij} \left(\frac{1}{h_j} \frac{\partial u^i}{\partial \theta^j} \right)^2}}{\sqrt{\sum_{ij} \left(\frac{1}{h_i h_j} \frac{\partial^2 u^i}{\partial \theta^i \partial \theta^j} \right)^2}} \right\} \quad (2.14)$$

K is an adjustable constant and the Richardson Number, Ri, is given as:

$$Ri = \frac{H}{u_i^2} \frac{\left(\frac{\partial T}{\partial n} \right) \bar{n} \cdot \bar{g}}{\left[\left(\frac{\partial u^1}{\partial n} \right) \bar{n} \cdot \bar{g} \right]^2 + \left[\left(\frac{\partial u^2}{\partial n} \right) \bar{n} \cdot \bar{g} \right]^2 + \left[\left(\frac{\partial u^3}{\partial n} \right) \bar{n} \cdot \bar{g} \right]^2} \quad (2.15)$$

\bar{n} is a unit vector in the opposite direction of gravity.

Pr_t is the turbulent Prandtl number which is also used to compute the effective conductivity.

$$k_{eff} = \frac{1}{Pr} + \frac{1}{Pr_t} \frac{\mu_{eff}}{\mu_o} \quad (2.16)$$

Pr is the molecular Prandtl number.

3. Surface Radiation Model

Raycraft [Ref. 29, pp. 24-44] describes this model in detail. Summarizing, the radiation model considers only surface radiation. Smoke and gases are considered transparent. Inside the model, walls and flame areas are treated as surfaces. Each surface is considered to be gray and diffuse. Sparrow and Cess [Ref. 34] discuss the net radiosity method upon which this model is based.

Net rate of heat loss per unit area is given as:

$$\frac{Q_i}{A_i} = \sum_{j=1}^N G_{ij} \sigma T_j^4 \quad (2.17)$$

where

$$G_{ij} = \frac{\epsilon_i}{1 - \epsilon_i} (\delta_{ij} - \Psi_{ij}) \quad (2.18)$$

$$\Psi_{ij} = \chi_{ij}^{-1}$$

$$\chi_{ij} = \frac{\delta_{ij} - (1 - \epsilon_i) F_{A_i-A_j}}{\epsilon_i} \quad (2.19)$$

$F_{A_i-A_j}$ is the view factor of radiation emitted by surface i onto surface j . The general equation is given by

$$F_{A_i-A_j} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos \beta_1 \cos \beta_j dA_i dA_j}{\pi r^2} \quad (2.20)$$

III. FINITE VOLUME CALCULATIONS

A. INTRODUCTION

The numerical model's independent variables are time and three space coordinates. Dependent variables consist of the three dimensional components of velocity, temperature, pressure and density. These six unknowns require six equations for solution. They are the continuity equation (Eq. (2.1)), the three momentum equations (Eq. (2.5)), the energy equation (Eq. (2.2)), and the equations of state (Eq. (2.7) and (2.8)). Doria [Ref. 35] discretized these equations in a method similar to this particular model based on the generalized form presented by Patanker [Ref. 36]. Doria applied the conservation equations in integral form to each control volume creating a set of finite difference equations which would lead to a solution.

Each control volume, or cell, surrounds a nodal point where one value of each property is constant throughout. The center nodal point determines pressure density and temperature. The grid determining velocities are staggered by one-half a cell length. Patanker [Ref. 36:pp. 115-120] describes how this alleviates two problems: the pressure differential between the two adjacent nodes, which ultimately determines the velocity at the node in question, is based on

a length which is half as long as in the unstaggered cell (this reduces error by one half); second, stability is gained by this stagger which precludes unrealistic, wavy oscillatory velocity fields, since the difference of adjacent velocities are used to satisfy continuity.

Since primitive variables are used versus the stream function, the pressure term coupling between equations must be handled specially. An iterative procedure estimates pressure and then pressure is corrected to ensure continuity is satisfied for each cell. A local pressure correction is discussed by both Patanker [Ref. 36:pp. 120-128] and Doria [Ref. 35:pp. 26-32]. A global pressure correction is included in the model to handle net energy changes and is described by Nicolette, et al. [Ref. 3].

The finite difference equations are solved iteratively. Non-linear problems like fluid flow are difficult to force convergence to final solution. Many schemes have been developed to obtain the flow problem solution. Each method has its problems and instabilities. This model employs the Quadratic Upstream Interpolation for Convective Kinematics, or QUICK, developed by Leonard [Ref. 37]. QUICK estimates values and gradients of transport variables at the faces of the cells. It has the accuracy of central finite difference schemes and the stability of convective diffusion terms found in upwind differencing. Yang [Ref. 12] applied the QUICK scheme to coupled momentum energy and pressure equation

solutions for three-dimensional flow in tilted rectangular enclosures.

In this chapter, the governing equations will be applied to the specialized control volumes of the model. They will be put in integral form and discretized according to the QUICK scheme. Pressure correction from iteration will also be applied.

B. CONTROL VOLUME ANALYSIS

At the center of each elemental control volume, or cell, lies the grid point of interest. At this point, the model determines the unknown values of the dependent variables. Denoting this grid point as $P(i, j, k)$ we define its neighbors as: East $(i+1, j, k)$, West $(i-1, j, k)$, North $(i, j+1, k)$, South $(i, j-1, k)$, Front $(i, j, k+1)$, and Back $(i, j, k-1)$. The boundaries around P are designated by lower case letters $e, w, n, s, f,$ and b . Typical spherical and cylindrical cells are shown in Figures 3.1 and 3.2 respectively.

Figure 3.3 shows the basic two dimensional cell used to determine pressure, density and temperature. In contrast, Figure 3.4 shows the staggered grid used to determine velocities. The velocity u_i^1 is located on the west face; u_i^2 is located on the south face and u_i^3 is located on the back face (not shown). The superscripts on the velocities designate

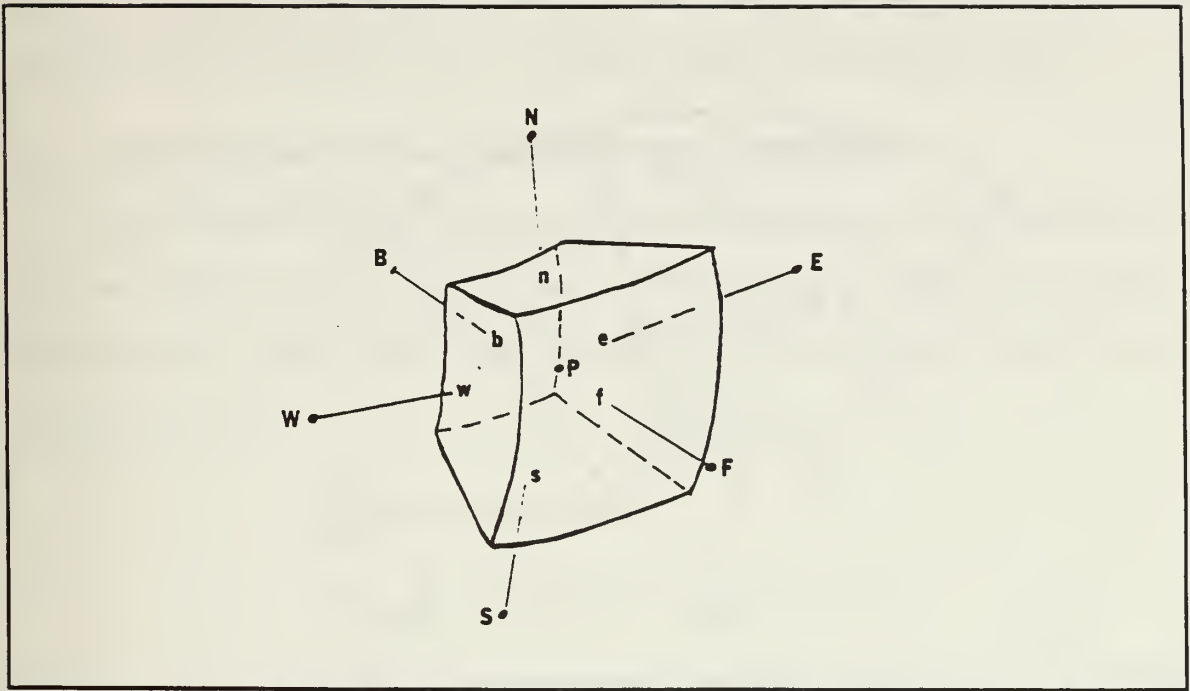


Figure 3.1 Basic Spherical Cell.

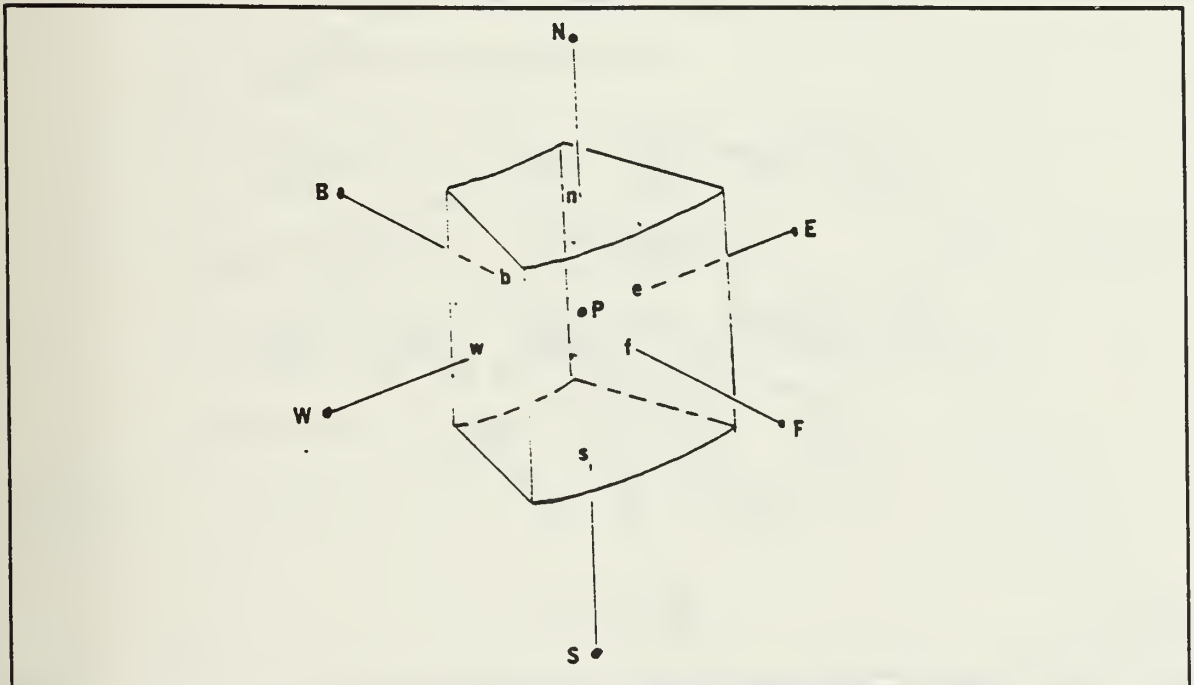


Figure 3.2 Basic Cylindrical Cell.

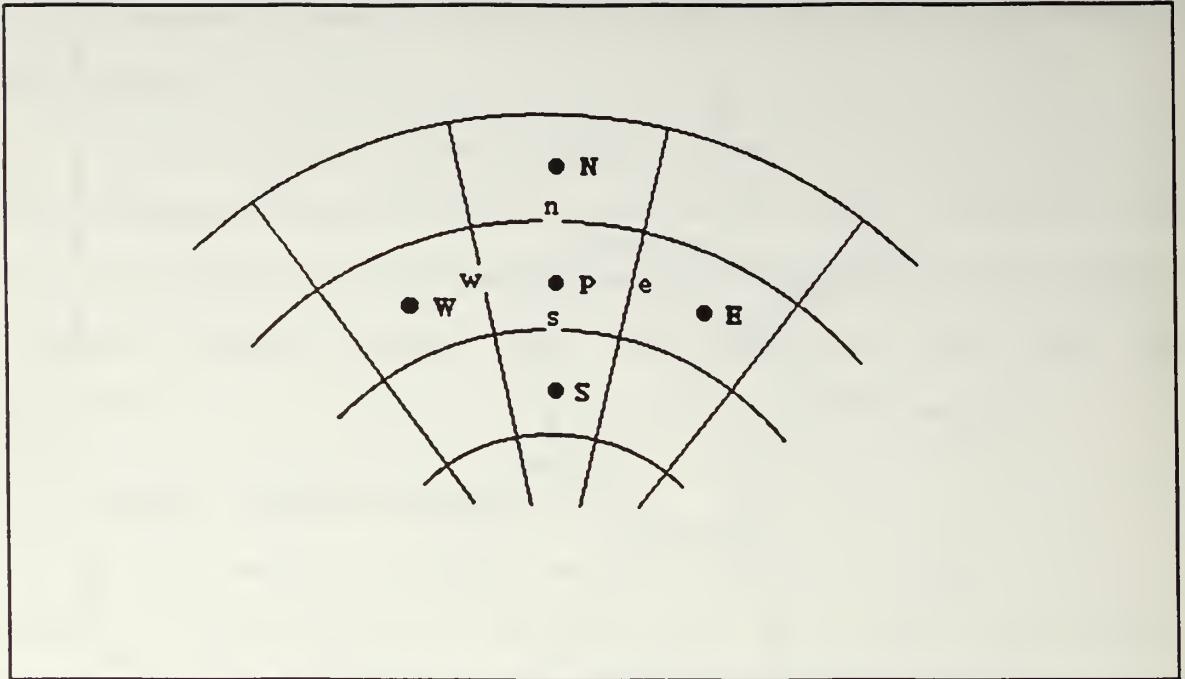


Figure 3.3 Two Dimensional Cell.

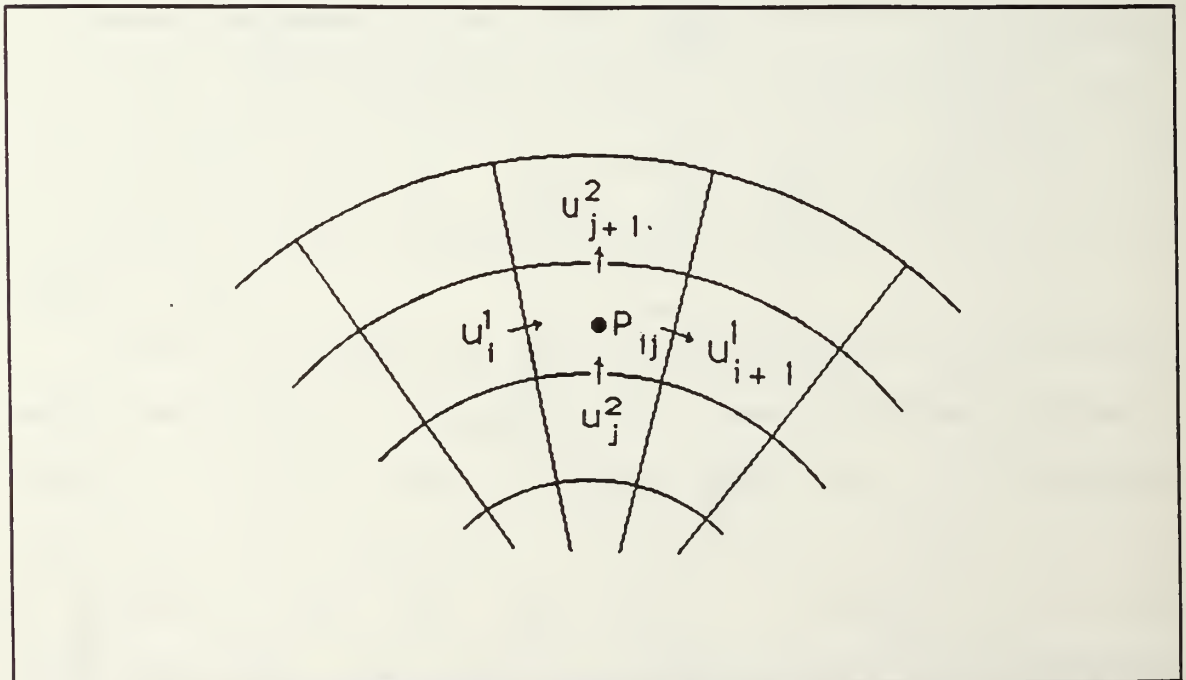


Figure 3.4 Two Dimensional Staggered Cell.

coordinate direction. These velocities are staggered in location by one-half cell length from the primary cell.

C. INTEGRATION OF THE CONSERVATION EQUATIONS

The conservation equations are integrated over each cell volume. From this point, they can be discretized into finite difference equations. The integral form of the continuity equation is:

$$\begin{aligned}
 & \int \frac{\partial \rho}{\partial t} h_1 h_2 h_3 \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & + \int \left[\frac{\partial}{\partial \theta^1} (\rho u^1 h_2 h_3) + \frac{\partial}{\partial \theta^2} (\rho u^2 h_3 h_1) \right. \\
 & \left. + \frac{\partial}{\partial \theta^3} (\rho u^3 h_1 h_2) \right] \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & = 0
 \end{aligned} \tag{3.1}$$

The energy equation becomes:

$$\begin{aligned}
 & \int \frac{\partial (\rho C_{pm} T)}{\partial t} h_1 h_2 h_3 \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & + \int \left[\frac{\partial}{\partial \theta^1} (\rho C_{pm} u^1 T h_2 h_3) + \frac{\partial}{\partial \theta^2} (\rho C_{pm} u^2 T h_1 h_3) \right. \\
 & \left. + \frac{\partial}{\partial \theta^3} (\rho C_{pm} u^3 T h_1 h_2) \right] \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & - \int \left[\frac{\partial}{\partial \theta^1} (q^1 h_2 h_3) + \frac{\partial}{\partial \theta^2} (q^2 h_1 h_3) + \frac{\partial}{\partial \theta^3} (q^3 h_1 h_2) \right] \\
 & \cdot \partial \theta^1 \partial \theta^2 \partial \theta^3 + \int S h_1 h_2 h_3 \partial \theta^1 \partial \theta^2 \partial \theta^3
 \end{aligned} \tag{3.2}$$

where:

$$q^1 = \frac{-k}{h_1} \frac{\partial T}{\partial \theta^1} \tag{3.3}$$

The momentum equations become:

$$\begin{aligned}
 & \int \frac{\partial}{\partial t} (\rho u^i) h_1 h_2 h_3 \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & + \int \frac{\partial}{\partial \theta^j} \left[\left(\frac{h_1 h_2 h_3}{h_j} \right) \rho u^i u^j \right] \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & = \int \frac{-\partial}{\partial \theta^i} \left(P \frac{h_1 h_2 h_3}{h_i} \right) \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & + \int \rho G_i h_1 h_2 h_3 \partial \theta^1 \partial \theta^2 \partial \theta^3 \tag{3.4} \\
 & + \int \frac{\partial}{\partial \theta^j} \left(\sigma^{ij} \frac{h_1 h_2 h_3}{h_i h_j} \right) \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & - \int \frac{h_1 h_2 h_3}{h_i h_j} \left[\frac{\partial h_i}{\partial \theta^j} (\rho u^j u^i - \sigma^{ij}) \right] \partial \theta^1 \partial \theta^2 \partial \theta^3 \\
 & + \int \frac{h_1 h_2 h_3}{h_j h_i} \left[\frac{\partial h_j}{\partial \theta^i} (\rho u^j u^i - \sigma^{ij}) \right] \partial \theta^1 \partial \theta^2 \partial \theta^3
 \end{aligned}$$

D. DISCRETIZATION OF THE CONTINUITY EQUATION

To provide maximum stability and accuracy for the model, three finite differencing schemes are utilized. Forward differencing is used for time dependence, central differencing is used for diffusion terms and the QUICK algorithm is used for the convective terms.

In forward differencing the future value of the time dependent variable is predicted from its previous value plus an additional term derived from the previously known slope m multiplied by the time step Δt . For example the new value for

density ρ^n is calculated using the old value ρ^{n-1} plus the extra term:

$$\rho^n = \rho^{n-1} + m\Delta t \quad (3.5)$$

The integrand in the continuity equation (3.1) becomes:

$$\frac{\partial \rho}{\partial t} dV = \frac{\rho^n - \rho^{n-1}}{\Delta t} h_1 h_2 h_3 \Delta\theta^1 \Delta\theta^2 \Delta\theta^3 = \frac{\rho^n - \rho^{n-1}}{\Delta t} \Delta V \quad (3.6)$$

Evaluating the integral, Equation (3.1) becomes:

$$\begin{aligned} & (\rho^n - \rho^{n-1}) \frac{\Delta V}{\Delta t} + [\rho u^1 h_2 h_3 d\theta^2 d\theta^3]_e - [\rho u^1 h_2 h_3 d\theta^2 d\theta^3]_w \\ & + [\rho u^2 h_1 h_3 d\theta^1 d\theta^3]_n - [\rho u^2 h_1 h_3 d\theta^1 d\theta^3]_s \\ & + [\rho u^3 h_1 h_2 d\theta^1 d\theta^2]_f - [\rho u^3 h_1 h_2 d\theta^1 d\theta^2]_b = 0 \end{aligned} \quad (3.7)$$

The mass flux, G , must be calculated at each face:

$$G_e = (\rho u^1)_e = u_e^1 \left[\frac{\rho_P (h_1 \Delta\theta^1)_{i+1} + \rho_E (h_1 \Delta\theta^1)_i}{(h_1 \Delta\theta^1)_{i+1} + (h_1 \Delta\theta^1)_i} \right] \quad (3.8)$$

$$G_w = (\rho u^1)_w = u_w^1 \left[\frac{\rho_P (h_1 \Delta\theta^1)_{i-1} + \rho_W (h_1 \Delta\theta^1)_i}{(h_1 \Delta\theta^1)_{i-1} + (h_1 \Delta\theta^1)_i} \right] \quad (3.9)$$

$$G_n = (\rho u^2)_n = u_n^2 \left[\frac{\rho_P (h_2 \Delta\theta^2)_{j+1} + \rho_N (h_2 \Delta\theta^2)_j}{(h_2 \Delta\theta^2)_{j+1} + (h_2 \Delta\theta^2)_j} \right] \quad (3.10)$$

$$G_s = (\rho u^2)_s = u_s^2 \left[\frac{\rho_P (h_2 \Delta\theta^2)_{j-1} + \rho_S (h_2 \Delta\theta^2)_j}{(h_2 \Delta\theta^2)_{j-1} + (h_2 \Delta\theta^2)_j} \right] \quad (3.11)$$

$$G_f = (\rho u^3)_f = u_f^3 \left[\frac{\rho_p (h_3 \Delta \theta^3)_{k+1} + \rho_F (h_3 \Delta \theta^3)_k}{(h_3 \Delta \theta^3)_{k+1} + (h_3 \Delta \theta^3)_k} \right] \quad (3.12)$$

$$G_b = (\rho u^3)_b = u_b^3 \left[\frac{\rho_p (h_3 \Delta \theta^3)_{k-1} + \rho_B (h_3 \Delta \theta^3)_k}{(h_3 \Delta \theta^3)_{k-1} + (h_3 \Delta \theta^3)_k} \right] \quad (3.13)$$

The areas of the faces of the cell are given as:

$$A_{e,w} = (h_2 \Delta \theta^2 h_3 \Delta \theta^3)_{e,w} \quad (3.14)$$

$$A_{n,s} = (h_1 \Delta \theta^1 h_3 \Delta \theta^3)_{n,s} \quad (3.15)$$

$$A_{f,b} = (h_1 \Delta \theta^1 h_2 \Delta \theta^2)_{f,b} \quad (3.16)$$

In final finite difference form the continuity equation becomes:

$$\frac{(\rho^n - \rho^{n-1}) \Delta V}{\Delta t} + G_e - G_w + G_n - G_s + G_f - G_b = S_{mp} \quad (3.17)$$

S_{mp} is the mass source term. As this residual approaches zero, the solution approach the exact solution. Iterations occur until S_{mp} reaches a specific, extremely small, cut off value.

E. DISCRETIZATION OF THE ENERGY EQUATION

Integrating over the control volume, the energy equation becomes:

$$\begin{aligned}
 & [(\rho C_{pm} T)^n - (\rho C_{pm} T)^{n-1}] \frac{\Delta V}{\Delta t} + G_e (C_{pm} T)_e A_e \\
 & - G_w (C_{pm} T)_w A_w + G_n (C_{pm} T)_n A_n - G_s (C_{pm} T)_s A_s + \\
 & G_f (C_{pm} T)_f A_f - G_b (C_{pm} T)_b A_b \\
 & = k_e A_e \left(\frac{\partial T}{h_1 \partial \theta^1} \right)_e - k_w A_w \left(\frac{\partial T}{h_1 \partial \theta^1} \right)_w \\
 & + k_n A_n \left(\frac{\partial T}{h_2 \partial \theta^2} \right)_n - k_s A_s \left(\frac{\partial T}{h_2 \partial \theta^2} \right)_s \\
 & + k_f A_f \left(\frac{\partial T}{h_3 \partial \theta^3} \right)_f - k_b A_b \left(\frac{\partial T}{h_3 \partial \theta^3} \right)_b + S_f \Delta V
 \end{aligned} \tag{3.18}$$

where S_f is the source term including dissipation, radiation, pressure work and heat sources. The total heat flux, J , resulting from convection and conduction is:

$$J_{e,w}^1 = \left[(\rho C_{pm} u^1 T) - k_{eff} \frac{\partial T}{h_1 \partial \theta^1} \right]_{e,w} \tag{3.19}$$

$$J_{n,s}^2 = \left[(\rho C_{pm} u^2 T) - k_{eff} \frac{\partial T}{h_2 \partial \theta^2} \right]_{n,s} \tag{3.20}$$

$$J_{f,b}^3 = \left[(\rho C_{pm} u^3 T) - k_{eff} \frac{\partial T}{h_3 \partial \theta^3} \right]_{f,b} \tag{3.21}$$

The final finite difference form of the energy equation becomes:

$$\begin{aligned}
 & [(\rho C_{pm} T)^n - (\rho C_{pm} T)^{n-1}] \frac{\Delta V}{\Delta t} + J_e^1 A_e \\
 & - J_w^1 A_w + J_n^2 A_n - J_s^2 A_s + J_f^3 A_f - J_b^3 A_b = S_f \Delta V
 \end{aligned}
 \tag{3.22}$$

The term $(\rho u^i C_{pm} T)$ in the flux equations give rise to difficulties since C_{pm} , ρ and T are evaluated at the nodal point instead of the surface of the cell. Thus, fluxes are determined from values of ρ , T , and C_{pm} at P and its neighbors.

The QUICK Scheme is used to determine accurate values of the dependent variables at the control volume surfaces with stable properties. QUICK couples the stability of upwind differencing with the accuracy of central differencing. It is achieved by using a parabolic polynomial interpolation to fit the control volume at three consecutive nodal points. Two nodes are located on either side of the surface and one is located upstream. Yang [Ref. 12:pp. 77-89] discusses QUICK for one, two and three dimensions. Houck [Ref. 30:pp. 37-50] and Raycraft [Ref. 29:pp. 63-74] used the QUICK scheme for the energy equations and that method is repeated here.

Figure 3.5 from Raycraft [Ref. 29:pp. 64] shows the one dimensional scheme for the quadratic interpolation of a non-uniform grid.

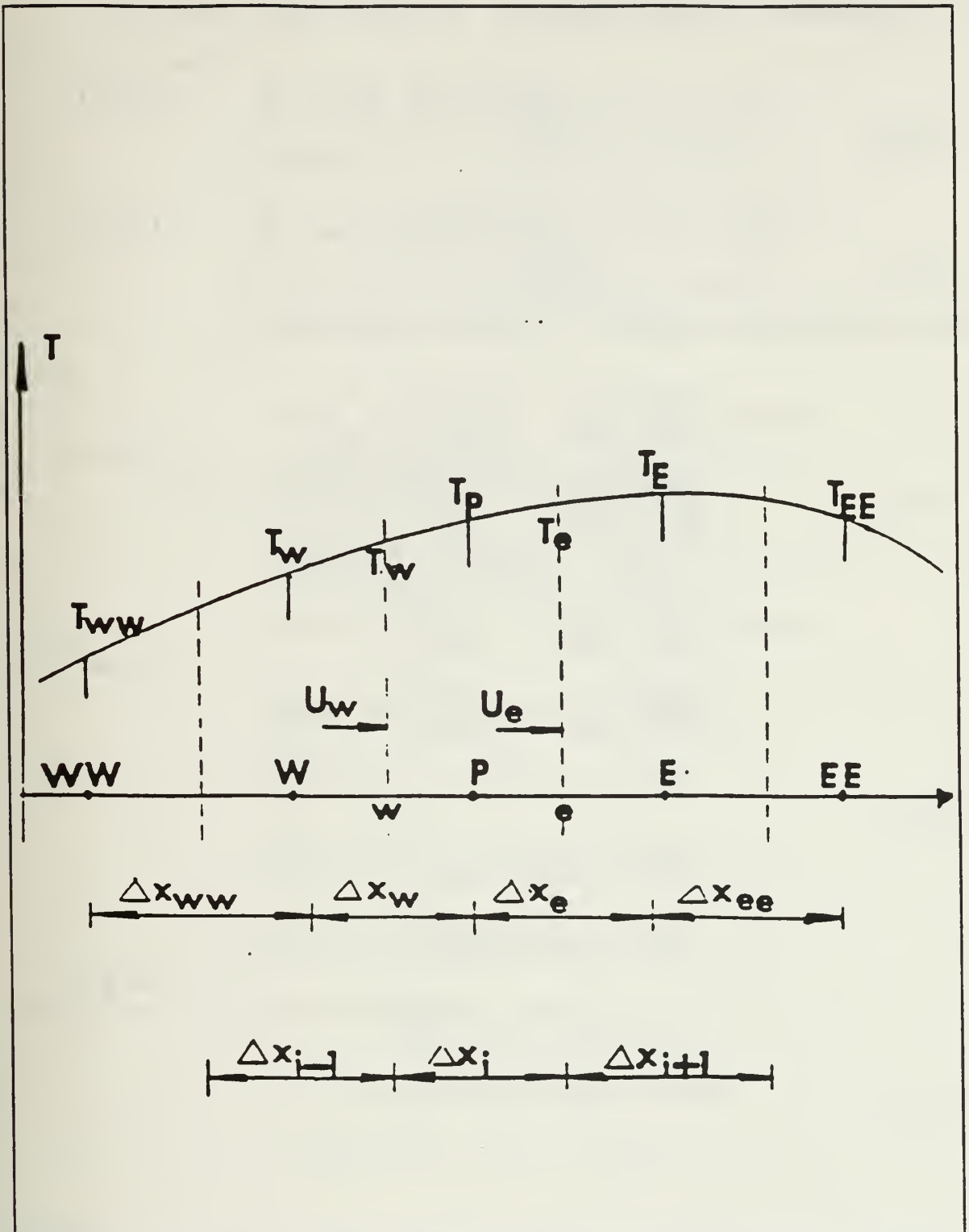


Figure 3.5 One Dimensional Quadratic Interpolation Scheme.

It is given by the equations

$$(\rho C_{pm} u T)_e = G_e C_{pm} \left[\left(\frac{T_p + T_E}{2} \right) - \frac{1}{8} \text{curv}_e \right] \quad (3.23)$$

$$(\rho C_{pm} v T)_w = G_w C_{pm} \left[\left(\frac{T_p + T_w}{2} \right) - \frac{1}{8} \text{curv}_w \right] \quad (3.24)$$

where the upstream weighted curvature terms are:

$$\begin{aligned} \text{curv}_e &= \frac{\Delta X_e^2}{\Delta X_i} \left[\frac{T_E - T_p}{\Delta X_e} - \frac{T_p - T_w}{\Delta X_w} \right] \quad \text{if } G_e > 0 \\ &= \frac{\Delta X_e^2}{\Delta X_{i+1}} \left[\frac{T_{EE} - T_e}{\Delta X_{ee}} - \frac{T_E - T_p}{\Delta X_e} \right] \quad \text{if } G_e < 0 \end{aligned} \quad (3.25)$$

$$\begin{aligned} \text{curv}_w &= \frac{\Delta X_w^2}{\Delta X_{i+1}} \left[\frac{T_p - T_w}{\Delta X_w} - \frac{T_w - T_{ww}}{\Delta X_{ww}} \right] \quad \text{if } G_w > 0 \\ &= \frac{\Delta X_w^2}{\Delta X_i} \left[\frac{T_E - T_p}{\Delta X_e} - \frac{T_p - T_w}{\Delta X_w} \right] \quad \text{if } G_w < 0 \end{aligned} \quad (3.26)$$

and

$$\begin{aligned} \Delta X_e &= \frac{1}{2} (\Delta X_i + \Delta X_{i+1}) \\ \Delta X_w &= \frac{1}{2} (\Delta X_i + \Delta X_{i-1}) \\ \Delta X_{ee} &= \frac{1}{2} (\Delta X_{i+1} + \Delta X_{i+2}) \\ \Delta X_{ww} &= \frac{1}{2} (\Delta X_{i-1} + \Delta X_{i-2}) \end{aligned} \quad (3.27)$$

In generalized orthogonal coordinates the convective flux terms become:

$$(\rho C_{pm} u^1 T)_e = G_e C_{pm} \left(\frac{T_p + T_E}{2} - \frac{1}{8} \text{curvn}_e \right) \quad (3.28)$$

$$(\rho C_{pm} u^2 T)_w = G_w C_{pm} \left(\frac{T_p + T_w}{2} - \frac{1}{8} \text{curvn}_w \right) \quad (3.29)$$

where

$$\begin{aligned} \text{curvn}_e &= \frac{(h_1 \Delta \theta^1)_e^2}{(h_1 \Delta \theta^1)_i} \left[\frac{T_E - T_p}{(h_1 \Delta \theta^1)_e} - \frac{T_p - T_w}{(h_1 \Delta \theta^1)_w} \right] \text{ if } G_e > 0 \\ &= \frac{(h_1 \Delta \theta^1)_e^2}{(h_1 \Delta \theta^1)_{i+1}} \left[\frac{T_{EE} - T_E}{(h_1 \Delta \theta^1)_{ee}} - \frac{T_E - T_p}{(h_1 \Delta \theta^1)_e} \right] \text{ if } G_e < 0 \end{aligned} \quad (3.31)$$

and

$$\begin{aligned} (h_1 \Delta \theta^1)_e &= \frac{1}{2} [(h_1 \Delta \theta^1)_i + (h_1 \Delta \theta^1)_{i+1}] \\ (h_1 \Delta \theta^1)_w &= \frac{1}{2} [(h_1 \Delta \theta^1)_i + (h_1 \Delta \theta^1)_{i-1}] \\ (h_1 \Delta \theta^1)_{ee} &= \frac{1}{2} [(h_1 \Delta \theta^1)_{i+1} + (h_1 \Delta \theta^1)_{i+2}] \\ (h_1 \Delta \theta^1)_{ww} &= \frac{1}{2} [(h_1 \Delta \theta^1)_{i-1} + (h_1 \Delta \theta^1)_{i-2}] \end{aligned} \quad (3.32)$$

Equation (3.22) now becomes:

$$\begin{aligned} & [(\rho C_{pm} T)^n - (\rho C_{pm} T)^{n-1}] h_1 \frac{\Delta V}{\Delta t} \\ &= A_E T_E + A_W T_W - A_P T_P + S (h_1 \Delta \theta^1) \end{aligned} \quad (3.33)$$

T_{ee} and T_{ww} are included in the source term using a semi-implicit tri-diagonal solution procedure. For a uniform grid, the other coefficients are:

$$A_E = \frac{C_{pm_e} (-7G_e + 3|G_e|)}{16} + C_{pm_w} (-G_w + |G_w|) + \frac{k_e}{h_1 \Delta \theta^1} \quad (3.34)$$

$$A_W = \frac{C_{pm_w} (9G_w + 3|G_w|)}{16} + C_{pm_e} (G_e + |G_e|) + \frac{k_w}{h_1 \Delta \theta^1} \quad (3.35)$$

$$A_P = \frac{9}{16} (G_w C_{pm_w} - G_e C_{pm_e}) + 3 (|G_w| C_{pm_w} + G_e) + \frac{k_w + k_e}{h_1 \Delta \theta^1} \quad (3.36)$$

$$S_P = Sh_1 \Delta \theta^1 - C_{pm_e} (|G_e| - G_e) T_{EE} - C_{pm_w} (|G_w| + G_w) T_{WW} \quad (3.37)$$

As mentioned before, Yang [Ref. 12:pp. 82-89] extended the QUICK algorithm to three dimensions. The three dimensional algorithm for generalized orthogonal coordinate system is described below.

As in the one dimensional case, the average temperature of the control volume is determined by interpolation of its neighbors in three directions. For illustration, Figure 3.6 from Raycraft [Ref. 29:pp. 68] shows a simpler uniform rectangular grid. The actual grid is similar except that its cylindrical/spherical geometry is more difficult to show. Yang [Ref. 12] describes how curvature terms are calculated for each of the temperatures and substituted into the convection

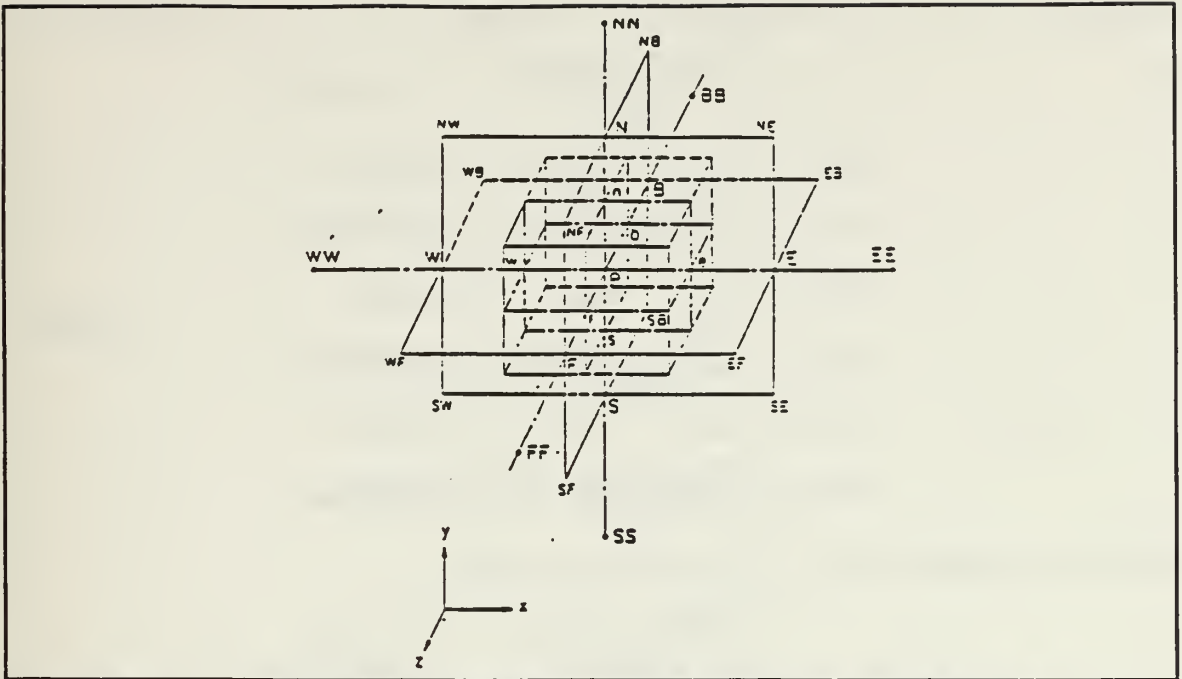


Figure 3.6 Calculation Cells for a Uniform Rectangular Grid.

terms of the energy equation. The new energy equation becomes

$$\begin{aligned}
 & [A_P^T + (\rho C_{pm})^{n-1}] \frac{\Delta V}{\Delta t} T_P \\
 & = A_E^T T_E + A_W^T T_W + A_N^T T_N + A_S^T T_S + A_F^T T_F + A_B^T T_B + S_U^T
 \end{aligned}
 \tag{3.38}$$

where the additional source term S_U^T is:

$$S_U^T = (\rho C_{pm})^{n-1} \frac{\Delta V}{\Delta t} - A_{EER} + A_{WNR} + A_{NNR} + A_{SSR} + A_{FFR} + A_{BBR}
 \tag{3.39}$$

The following terms are part of Equation (3.38). All values are for point (i, j, k) unless specified elsewhere. For = =

example, u_{1jk}^1 is designated u_1^1 whereas, $u_{i+1, j, k}$ is specified u_{i+1} .

$$\begin{aligned}
 CN &= G_n \cdot u_{j+1}^2 \cdot (h_3 \Delta \theta^3)_n \cdot (h_1 \Delta \theta^1)_n \\
 CS &= G_s \cdot u_j^2 \cdot (h_3 \Delta \theta^3)_s \cdot (h_1 \Delta \theta^1)_s \\
 CE &= G_e \cdot u_{i+1}^1 \cdot (h_3 \Delta \theta^3)_e \cdot (h_2 \Delta \theta^2)_e \\
 CW &= G_w \cdot u_i^1 \cdot (h_3 \Delta \theta^3)_w \cdot (h_2 \Delta \theta^2)_w \\
 CF &= G_f \cdot u_{k+1}^3 \cdot (h_1 \Delta \theta^1)_f \cdot (h_2 \Delta \theta^2)_f \\
 CB &= G_b \cdot u_k^3 \cdot (h_1 \Delta \theta^1)_b \cdot (h_2 \Delta \theta^2)_b
 \end{aligned} \tag{3.40}$$

Thermal conductivity is expressed as:

$$\begin{aligned}
 k_n &= \left[\frac{(k_j \cdot (h_2 \Delta \theta^2)_j)^{-1} + (k_{j+1} \cdot (h_2 \Delta \theta^2)_{j+1})^{-1}}{(h_2 \Delta \theta^2)_j + (h_2 \Delta \theta^2)_{j+1}} \right]^{-1} \\
 k_s &= \left[\frac{(k_j \cdot (h_2 \Delta \theta^2)_j)^{-1} + (k_{j-1} \cdot (h_2 \Delta \theta^2)_{j-1})^{-1}}{(h_2 \Delta \theta^2)_j + (h_2 \Delta \theta^2)_{j-1}} \right]^{-1} \\
 k_e &= \left[\frac{(k_i \cdot (h_1 \Delta \theta^1)_i)^{-1} + (k_{i+1} \cdot (h_1 \Delta \theta^1)_{i+1})^{-1}}{(h_1 \Delta \theta^1)_i + (h_1 \Delta \theta^1)_{i+1}} \right]^{-1} \\
 k_w &= \left[\frac{(k_i \cdot (h_1 \Delta \theta^1)_i)^{-1} + (k_{i-1} \cdot (h_1 \Delta \theta^1)_{i-1})^{-1}}{(h_1 \Delta \theta^1)_i + (h_1 \Delta \theta^1)_{i-1}} \right]^{-1} \\
 k_f &= \left[\frac{(k_k \cdot (h_3 \Delta \theta^3)_k)^{-1} + (k_{k+1} \cdot (h_3 \Delta \theta^3)_{k+1})^{-1}}{(h_3 \Delta \theta^3)_k + (h_3 \Delta \theta^3)_{k+1}} \right]^{-1} \\
 k_b &= \left[\frac{(k_k \cdot (h_3 \Delta \theta^3)_k)^{-1} + (k_{k-1} \cdot (h_3 \Delta \theta^3)_{k-1})^{-1}}{(h_3 \Delta \theta^3)_k + (h_3 \Delta \theta^3)_{k-1}} \right]^{-1}
 \end{aligned} \tag{3.41}$$

$$CONDN1 = k_n \cdot \left[\frac{h_3 \Delta \theta^3 \cdot h_1 \Delta \theta^1}{h_2 \Delta \theta^2} \right]_n$$

$$CONDS1 = k_s \cdot \left[\frac{h_3 \Delta \theta^3 \cdot h_1 \Delta \theta^1}{h_2 \Delta \theta^2} \right]_s$$

$$CONDE1 = k_e \cdot \left[\frac{h_3 \Delta \theta^3 \cdot h_2 \Delta \theta^2}{h_1 \Delta \theta^1} \right]_e$$

(3.42)

$$CONDW1 = k_w \cdot \left[\frac{h_3 \Delta \theta^3 \cdot h_2 \Delta \theta^2}{h_1 \Delta \theta^1} \right]_w$$

$$CONDF1 = k_f \cdot \left[\frac{h_1 \Delta \theta^1 \cdot h_2 \Delta \theta^2}{h_3 \Delta \theta^3} \right]_f$$

$$CONDB1 = k_b \cdot \left[\frac{h_1 \Delta \theta^1 \cdot h_2 \Delta \theta^2}{h_3 \Delta \theta^3} \right]_b$$

$$CEP = \frac{|CE| + CE}{16} \cdot \frac{(h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_i}$$

$$CEM = \frac{|CE| - CE}{16} \cdot \frac{(h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_{i+1}}$$

$$CWP = \frac{|CW| + CW}{16} \cdot \frac{(h_1 \Delta \theta^1)_w}{(h_1 \Delta \theta^1)_{i-1}}$$

$$CWM = \frac{|CW| - CW}{16} \cdot \frac{(h_1 \Delta \theta^1)_w}{(h_1 \Delta \theta^1)_i}$$

$$CNP = \frac{|CN| + CN}{16} \cdot \frac{(h_2 \Delta \theta^2)_n}{(h_2 \Delta \theta^2)_j}$$

$$CNM = \frac{|CN| - CN}{16} \cdot \frac{(h_2 \Delta \theta^2)_n}{(h_2 \Delta \theta^2)_{j+1}}$$

$$CSP = \frac{|CS| + CS}{16} \cdot \frac{(h_2 \Delta \theta^2)_s}{(h_2 \Delta \theta^2)_{j-1}}$$

$$CSM = \frac{|CS| - CS}{16} \cdot \frac{(h_2 \Delta \theta^2)_s}{(h_2 \Delta \theta^2)_j}$$

$$CFP = \frac{|CF| + CF}{16} \cdot \frac{(h_3 \Delta \theta^3)_f}{(h_3 \Delta \theta^3)_k}$$

$$CFM = \frac{|CF| - CF}{16} \cdot \frac{(h_3 \Delta \theta^3)_f}{(h_3 \Delta \theta^3)_{k+1}}$$

$$CBP = \frac{|CB| + CB}{16} \cdot \frac{(h_3 \Delta \theta^3)_b}{(h_3 \Delta \theta^3)_{k-i}}$$

$$CBM = \frac{|CB| - CB}{16} \cdot \frac{(h_3 \Delta \theta^3)_b}{(h_3 \Delta \theta^3)_k}$$

(3.43)

$$\begin{aligned}
A_{EE}^T &= \frac{-CEM \cdot (h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_{ee}} \\
A_{WW}^T &= \frac{-CWP \cdot (h_1 \Delta \theta^1)_w}{(h_1 \Delta \theta^1)_{ww}} \\
A_{NN}^T &= \frac{-CNM \cdot (h_2 \Delta \theta^2)_n}{(h_2 \Delta \theta^2)_{nn}} \\
A_{SS}^T &= \frac{-CSP \cdot (h_2 \Delta \theta^2)_s}{(h_2 \Delta \theta^2)_{ss}} \\
A_{FF}^T &= \frac{-CFM \cdot (h_3 \Delta \theta^3)_f}{(h_3 \Delta \theta^3)_{ff}} \\
A_{BB}^T &= \frac{-CBP \cdot (h_3 \Delta \theta^3)_b}{(h_3 \Delta \theta^3)_{bb}}
\end{aligned}
\tag{3.44}$$

Final coefficients for the source term are:

$$\begin{aligned}
A_{EER} &= A_{EE}^T \cdot T_{1+2} \cdot C_{pm_{1+1}} \\
A_{NWR} &= A_{WW}^T \cdot T_{1-2} \cdot C_{pm_{1-2}} \\
A_{NNR} &= A_{NN}^T \cdot T_{j+2} \cdot C_{pm_{j+2}} \\
A_{SSR} &= A_{SS}^T \cdot T_{j-2} \cdot C_{pm_{j-2}} \\
A_{FFR} &= A_{FF}^T \cdot T_{k+2} \cdot C_{pm_{k+2}} \\
A_{BBR} &= A_{BB}^T \cdot T_{k-2} \cdot C_{pm_{k-2}}
\end{aligned}
\tag{3.45}$$

Intermediate coefficients are:

$$A_{EI} = -\frac{1}{2} \cdot CE + CEP + CEM$$

$$\cdot \left[1 + \frac{(h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_{ee}} \right] + CWM \cdot \left[\frac{(h_1 \Delta \theta^1)_w}{(h_1 \Delta \theta^1)_e} \right] \quad (3.46)$$

$$A_{WI} = \frac{1}{2} \cdot CW + CWM + CWP$$

$$\cdot \left[1 + \frac{(h_1 \Delta \theta^1)_w}{(h_1 \Delta \theta^1)_{ww}} \right] + CEP \cdot \left[\frac{(h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_w} \right] \quad (3.47)$$

$$A_{NI} = -\frac{1}{2} \cdot CN + CNP + CNM$$

$$\cdot \left[1 + \frac{(h_2 \Delta \theta^2)_n}{(h_2 \Delta \theta^2)_{nn}} \right] + CEP \cdot \left[\frac{(h_2 \Delta \theta^2)_s}{(h_2 \Delta \theta^2)_n} \right] \quad (3.48)$$

$$A_{SI} = \frac{1}{2} \cdot CS + CSM + CSP$$

$$\cdot \left[1 + \frac{(h_2 \Delta \theta^2)_s}{(h_2 \Delta \theta^2)_{ss}} \right] + CNP \cdot \left[\frac{(h_2 \Delta \theta^2)_n}{(h_2 \Delta \theta^2)_s} \right] \quad (3.49)$$

$$A_{FI} = -\frac{1}{2} \cdot CF + CFP + CFM$$

$$\cdot \left[1 + \frac{(h_3 \Delta \theta^3)_f}{(h_3 \Delta \theta^3)_{ff}} \right] + CBM \cdot \left[\frac{(h_3 \Delta \theta^3)_b}{(h_3 \Delta \theta^3)_f} \right] \quad (3.50)$$

$$A_{BI} = \frac{1}{2} \cdot CB + CBM + CBP$$

$$\cdot \left[1 + \frac{(h_3 \Delta \theta^3)_b}{(h_3 \Delta \theta^3)_{bb}} \right] + CFP \cdot \left[\frac{(h_3 \Delta \theta^3)_f}{(h_3 \Delta \theta^3)_b} \right] \quad (3.51)$$

Final coefficients are:

$$\begin{aligned}
 A_E^T &= A_{EI} \cdot C_{pm_e} + CONDE1 \\
 A_W^T &= A_{WI} \cdot C_{pm_w} + CONDW1 \\
 A_N^T &= A_{NI} \cdot C_{pm_n} + CONDN1 \\
 A_S^T &= A_{SI} \cdot C_{pm_s} + CONDS1 \\
 A_F^T &= A_{FI} \cdot C_{pm_f} + CONDF1 \\
 A_B^T &= A_{BI} \cdot C_{pm_b} + CONDB1
 \end{aligned}
 \tag{3.52}$$

and:

$$\begin{aligned}
 A_p^T &= C_{p_{m,p}} \cdot (A_E^T + A_W^T + A_N^T + A_S^T + A_F^T + A_B^T \\
 &+ A_{EE}^T + A_{WW}^T + A_{NN}^T + A_{SS}^T + A_{FF}^T + A_{BB}^T) + CONDE1 \\
 &+ CONDW1 + CONDN1 + CONDS1 + CONDF1 + CONDB1
 \end{aligned}
 \tag{3.53}$$

F. DISCRETIZATION OF THE MOMENTUM EQUATION

The integrated momentum equation is:

$$\begin{aligned}
 (\rho u^i)_t V + M_e^{i1} A_e - M_w^{i1} A_w + M_n^{i2} A_n \\
 - M_s^{i2} A_s + M_f^{i3} A_f - M_b^{i3} A_b = S^i
 \end{aligned}
 \tag{3.54}$$

where A_i are the face areas of the staggered cell given by Equations (3.14 - 3.16). M^{ij} is the momentum flux in the θ^{ij} direction due to velocity u^i convection and to diffusion, and is given by:

$$M^{ij} = (\rho u^i u^j - \sigma_i^j)
 \tag{3.55}$$

Included in the source term S^i are pressure gradient, body, coriolis and centrifugal forces. The source term for velocity u^i is:

$$S^i = -P_e A_e + P_w A_w + \rho G^1 \Delta V \quad (3.56)$$

$$- M_p^{i2} (A_n - A_s) - M_p^{i3} (A_f - A_b) + (M_p^{22} + M_p^{33}) (A_e + A_w)$$

Yang, et al. [Ref. 19:pp. 11-13] describe a "stress flux formation" as it applies to a curvilinear coordinate system. Stresses are evaluated from previous information and the source is given in the current information. The momentum flux is:

$$M^{ij} = \hat{M}^{ij} + (\sigma_i^j - \sigma_j^i) \quad (3.57)$$

where:

$$\sigma_i^j = \frac{\mu}{\left[h_j \left(\frac{\partial u^i}{\partial \theta^j} \right) \right]} \quad (3.58)$$

$$\hat{M}^{ij} = \rho u^i u^j - \sigma_i^j \quad (3.59)$$

The momentum equation for velocity u^i is now:

$$(\rho u)_t + \hat{M}_e^{11} A_e - \hat{M}_w^{11} A_w + \hat{M}_n^{12} A_n - \hat{M}_s^{12} A_s + \hat{M}_f^{13} A_f + \hat{M}_b^{13} A_b = \hat{S} \quad (3.60)$$

where:

$$\hat{S} = S - (\sigma_1^i - \sigma_i^1)_e A_e + (\sigma_1^i - \sigma_i^1)_w A_w - (\sigma_1^2 - \sigma_2^1)_n A_n \quad (3.61)$$

$$+ (\sigma_1^2 - \sigma_2^1)_s A_s - (\sigma_1^3 - \sigma_3^1)_f A_f - (\sigma_1^3 - \sigma_3^1)_b A_b$$

The momentum equation for θ^1 takes a form similar to the energy equation

$$\left(A_p^{u^1} + \rho^{n-1} \frac{\Delta V}{\Delta t} \right) u_p^2 = A_e^{u^1} u_e^1 + A_w^{u^1} u_w^1 + A_n^{u^1} u_n^1 + A_s^{u^1} u_s^1 + A_f^{u^1} u_f^1 + A_b^{u^1} u_b^1 + S^{u^1} u^1 \quad (3.62)$$

Again we must obtain final coefficients. Introducing intermediate mass flow rate per unit area:

$$\begin{aligned} G_{ne} &= u_{j+1}^2 \left\{ \frac{[\rho_{j+1} (h_2 \Delta \theta^2)_j + \rho_j (h_2 \Delta \theta^2)_{j+1}]}{(h_2 \Delta \theta^2)_j + (h_2 \Delta \theta^2)_{j+1}} \right\} \\ G_{nw} &= u_{i-1, j+1}^2 \left\{ \frac{[\rho_{i-1, j+1} (h_2 \Delta \theta^2)_j + \rho_{i-1} (h_2 \Delta \theta^2)_{j+1}]}{(h_2 \Delta \theta^2)_j + (h_2 \Delta \theta^2)_{j+1}} \right\} \\ G_{se} &= u^2 \left\{ \frac{[\rho_{j-1} (h_2 \Delta \theta^2)_j + \rho_j (h_2 \Delta \theta^2)_{j-1}]}{(h_2 \Delta \theta^2)_j + (h_2 \Delta \theta^2)_{j-1}} \right\} \\ G_{sw} &= u_{i-1}^2 \left\{ \frac{[\rho_{i-1, j-1} (h_2 \Delta \theta^2)_j + \rho_{i-1} (h_2 \Delta \theta^2)_{j-1}]}{(h_2 \Delta \theta^2)_j + (h_2 \Delta \theta^2)_{j-1}} \right\} \\ G_e &= u_{i+1}^1 \left\{ \frac{[\rho_{i+1} (h_1 \Delta \theta^1)_e + \rho_i (h_1 \Delta \theta^1)_{ee}]}{(h_1 \Delta \theta^1)_e + (h_1 \Delta \theta^1)_{ee}} \right\} \end{aligned} \quad (3.63a)$$

$$\begin{aligned}
G_p &= U^i \left\{ \frac{[\rho_{i-1} (h_1 \Delta \theta^1)_e + \rho_i (h_1 \Delta \theta^1)_w]}{(h_1 \Delta \theta^1)_e + h_1 \Delta \theta^1_w} \right\} \\
G_w &= U_{i-1}^i \left\{ \frac{[\rho_{i-2} (h_1 \Delta \theta^1)_w + \rho_{i-1} (h_1 \Delta \theta^1)_{ww}]}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_{ww}} \right\} \\
G_{fe} &= U_{k+1}^3 \left\{ \frac{[\rho_{k+1} (h_3 \Delta \theta^3)_k + \rho_k (h_3 \Delta \theta^3)_{k+1}]}{(h_3 \Delta \theta^3)_k + (h_3 \Delta \theta^3)_{k+1}} \right\} \\
G_{fw} &= U_{i-1, k+1}^3 \left\{ \frac{[\rho_{i-1, k+1} (h_3 \Delta \theta^3)_k + \rho_{i-1} (h_3 \Delta \theta^3)_{k+1}]}{(h_3 \Delta \theta^3)_k + (h_3 \Delta \theta^3)_{k+1}} \right\} \\
G_{be} &= U^3 \left\{ \frac{[\rho_{k-1} (h_3 \Delta \theta^3)_k + \rho_k (h_3 \Delta \theta^3)_{k-1}]}{(h_3 \Delta \theta^3)_k + (h_3 \Delta \theta^3)_{k-1}} \right\} \\
G_{bw} &= U_{i-1}^3 \left\{ \frac{[\rho_{i-1, k-1} (h_3 \Delta \theta^3)_k + \rho_{i-1} (h_3 \Delta \theta^3)_{k-1}]}{(h_3 \Delta \theta^3)_k + (h_3 \Delta \theta^3)_{k-1}} \right\}
\end{aligned} \tag{3.63b}$$

Final mass flow rates are:

$$\begin{aligned}
CE &= \frac{1}{2} (G_e + G_p) \cdot (h_2 \Delta \theta^2)_e \cdot (h_3 \Delta \theta^3)_e \\
CW &= \frac{1}{2} (G_p + G_w) \cdot (h_2 \Delta \theta^2)_w \cdot (h_3 \Delta \theta^3)_w \\
CN &= (h_1 \Delta \theta^1)_n \cdot (h_3 \Delta \theta^3)_n \cdot \left\{ \frac{[G_{ne} (h_1 \Delta \theta^1)_w + G_{nw} (h_1 \Delta \theta^1)_e]}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e} \right\} \\
CS &= (h_1 \Delta \theta^1)_s \cdot (h_3 \Delta \theta^3)_s \cdot \left\{ \frac{[G_{se} (h_1 \Delta \theta^1)_w + G_{sw} (h_1 \Delta \theta^1)_e]}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e} \right\} \\
CF &= (h_1 \Delta \theta^1)_f \cdot (h_2 \Delta \theta^2)_f \cdot \left\{ \frac{[G_{fe} (h_1 \Delta \theta^1)_w + G_{fw} (h_1 \Delta \theta^1)_e]}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e} \right\} \\
CB &= (h_1 \Delta \theta^1)_b \cdot (h_2 \Delta \theta^2)_b \cdot \left\{ \frac{[G_{be} (h_1 \Delta \theta^1)_w + G_{bw} (h_1 \Delta \theta^1)_e]}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e} \right\}
\end{aligned} \tag{3.64}$$

The local viscosity is:

$$\begin{aligned}
 VIS_e &= VIS \\
 VIS_w &= VIS_{i-1} \\
 VIS_n &= \frac{(VIS_{j+1} + VIS + VIS_{i-1, j+1} + VIS_{i-1})}{4} \\
 VIS_s &= \frac{(VIS_{j-1} + VIS + VIS_{i-1, j-1} + VIS_{i-1})}{4} \\
 VIS_f &= \frac{(VIS_{k+1} + VIS + VIS_{i-1, k+1} + VIS_{i-1})}{4} \\
 VIS_b &= \frac{(VIS_{k-1} + VIS + VIS_{i-1, k-1} + VIS_{i-1})}{4}
 \end{aligned} \tag{3.65}$$

$$\begin{aligned}
 VISN1 &= VIS_n \cdot \left[\frac{(h_3 \Delta \theta^3) (h_1 \Delta \theta^1)}{(h_2 \Delta \theta^2)} \right]_n \\
 VISS1 &= VIS_s \cdot \left[\frac{(h_3 \Delta \theta^3) (h_1 \Delta \theta^1)}{(h_2 \Delta \theta^2)} \right]_s \\
 VISE1 &= VIS_e \cdot \left[\frac{(h_2 \Delta \theta^2) (h_3 \Delta \theta^3)}{(h_1 \Delta \theta^1)} \right]_e \\
 VISW1 &= VIS_w \cdot \left[\frac{(h_2 \Delta \theta^2) (h_3 \Delta \theta^3)}{(h_1 \Delta \theta^1)} \right]_w \\
 VISF1 &= VIS_f \cdot \left[\frac{(h_1 \Delta \theta^1) (h_2 \Delta \theta^2)}{(h_3 \Delta \theta^3)} \right]_f \\
 VISB1 &= VIS_b \cdot \left[\frac{(h_1 \Delta \theta^1) (h_2 \Delta \theta^2)}{(h_3 \Delta \theta^3)} \right]_b
 \end{aligned} \tag{3.66}$$

The momentum equation coefficients are:

$$\begin{aligned}
 A_{EER} &= A_{EE}^u \cdot u_{i+1}^i \\
 A_{WWR} &= A_{WW}^u \cdot u_{i-2}^i \\
 A_{NNR} &= A_{NN}^u \cdot u_{j+2}^i \\
 A_{SSR} &= A_{SS}^u \cdot u_{j-2}^1 \\
 A_{FFR} &= A_{FF}^u \cdot u_{k+2}^1 \\
 A_{BBR} &= A_{BB}^u \cdot u_{k-2}^1
 \end{aligned}
 \tag{3.67}$$

As with the energy equation, the value of the final coefficients are:

$$\begin{aligned}
 A_E^u &= A_{EI} + VISE1 \\
 A_W^u &= A_{WI} + VISW1 \\
 A_N^u &= A_{NI} + VISN1 \\
 A_S^u &= A_{SI} + VISS1 \\
 A_F^u &= A_{FI} + VISF1 \\
 A_B^u &= A_{BI} + VISB1
 \end{aligned}
 \tag{3.68}$$

and

$$\begin{aligned}
 A_p^u &= A_E^u + A_W^u + A_N^u + A_S^u + A_F^u + A_B^u \\
 &\quad + A_{EE}^u + A_{WW}^u + A_{NN}^u + A_{SS}^u + A_{FF}^u + A_{BB}^u
 \end{aligned}
 \tag{3.69}$$

The final source term is given as

$$\begin{aligned}
 S_u^u = & \frac{[\rho (h_1 \Delta \theta^1)_w + \rho_{i-1} (h_1 \Delta \theta^1)_e]}{[(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e]} \cdot \frac{\Delta V}{\Delta t} \cdot u^1 \\
 & + (h_2 \Delta \theta^2)_j (h_3 \Delta \theta^3)_k (P_{i-1} - P_i) + A_{EER} + A_{WWR} + A_{NNR} \\
 & + A_{SSR} + A_{FFR} + A_{BBR} + RE - RW + RN - RS \\
 & + RF - RB + RRY + RRZ - RRX - BUOY \\
 & \cdot \{ \sin [ZC(K)] \cdot (\rho - \rho_{eq}) \cdot (h_1 \Delta \theta^1)_w \\
 & \cdot \cos [XC(I)] \} + \{ (\rho_{i-1} - \rho_{eq,i-1}) (h_1 \Delta \theta^1)_e \\
 & \cdot \cos [XC(I-1)] \} / [(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e] \Delta V
 \end{aligned} \tag{3.70}$$

where XC and ZC represent the center of the cell. The remainder of the terms are explained below.

$$\begin{aligned}
 RE &= (h_2 \Delta \theta^2)_e (h_3 \Delta \theta^3)_e \cdot \left[\frac{\sigma^{11} - (u_{i+1}^1 - u_i^1) \cdot VIS_e}{(h_1 \Delta \theta^1)_e} \right] \\
 RW &= (h_2 \Delta \theta^2)_w (h_3 \Delta \theta^3)_w \cdot \left[\frac{\sigma_{i-1}^{11} - (u^1 - u_{i-1}^1) \cdot VIS_w}{(h_1 \Delta \theta^1)_w} \right] \\
 RN &= (h_1 \Delta \theta^1)_n (h_3 \Delta \theta^3)_n \cdot \left[\frac{\sigma_{j+1}^{12} - (u_{j+1}^1 - u_j^1) \cdot VIS_n}{(h_2 \Delta \theta^2)_n} \right] \\
 RS &= (h_1 \Delta \theta^1)_s (h_3 \Delta \theta^3)_s \cdot \left[\frac{\sigma^{12} - (u^1 - u_{j-1}^1) \cdot VIS_s}{(h_2 \Delta \theta^2)_s} \right] \\
 RF &= (h_1 \Delta \theta^1)_f (h_2 \Delta \theta^2)_f \cdot \left[\frac{\sigma_{k+1}^{13} - (u_{k+1}^1 - u_k^1) \cdot VIS_f}{(h_3 \Delta \theta^3)_f} \right] \\
 RB &= (h_1 \Delta \theta^1)_b (h_2 \Delta \theta^2)_b \cdot \left[\frac{\sigma^{13} - (u^1 - u_{k-1}^1) \cdot VIS_b}{(h_1 \Delta \theta^1)_b} \right]
 \end{aligned} \tag{3.71}$$

$$\begin{aligned}\bar{\sigma}^{12} &= \frac{1}{2} (\sigma_{j+1}^{i2} + \sigma_j^{i2}) \\ \bar{\sigma}^{13} &= \frac{1}{2} (\sigma_{k+1}^{i3} + \sigma_k^{i3}) \\ \bar{\sigma}^{22} &= \frac{\sigma^{22} (h_1 \Delta \theta^1)_w + \sigma_{i-1}^{22} (h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e} \\ \bar{\sigma}^{33} &= \frac{\sigma^{33} (h_1 \Delta \theta^1)_w + \sigma_{i-1}^{33} (h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e}\end{aligned}\tag{3.72}$$

$$AU1 = u^1$$

$$\begin{aligned}AU2 &= \left\{ \frac{u_{j+1}^2 (h_2 \Delta \theta^2)_j + u_j^2 (h_2 \Delta \theta^2)_j}{2 (h_2 \Delta \theta^2)_j} \right\} (h_1 \Delta \theta^1)_w \\ &+ \left\{ \frac{u_{i-1, j+1}^2 (h_2 \Delta \theta^2)_j + u_{i-1}^2 (h_2 \Delta \theta^2)_j}{2 (h_2 \Delta \theta^2)_j} \right\} (h_1 \Delta \theta^1)_e \\ &/ [(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e]\end{aligned}\tag{3.73}$$

$$\begin{aligned}AU3 &= \left\{ \frac{u_{k+1}^3 (h_3 \Delta \theta^3)_k + u_k^3 (h_3 \Delta \theta^3)_k}{2 (h_3 \Delta \theta^3)_k} \right\} (h_1 \Delta \theta^1)_w \\ &+ \frac{[u_{i-1, k+1}^3 (h_3 \Delta \theta^3)_k + u_{i-1}^3 (h_3 \Delta \theta^3)_k]}{2 (H_3 \Delta \theta^3)} (h_1 \Delta \theta^1)_e \\ &/ [(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e]\end{aligned}$$

$$AR = \frac{\rho (h_1 \Delta \theta^1)_w + \rho_{i-1} (h_1 \Delta \theta^1)_e}{(h_1 \Delta \theta^1)_w + (h_1 \Delta \theta^1)_e}$$

$$ARU12 = AR \cdot AU1 \cdot AU2\tag{3.74}$$

$$ARU13 = AR \cdot AU1 \cdot AU3$$

$$ARU22 = AR \cdot AU2 \cdot AU2$$

$$ARU33 = AR \cdot AU3 \cdot AU3$$

$$\begin{aligned}
RRY &= (\bar{\sigma}^{12} - ARU12) (h_3 \Delta \theta^3)_k [(h_1 \Delta \theta^1)_n - (h_1 \Delta \theta^1)_s] \\
RRZ &= (\bar{\sigma}^{13} - ARU13) (h_2 \Delta \theta^2)_j [(h_2 \Delta \theta^1)_f - (h_1 \Delta \theta^1)_b] \\
RRX &= (\bar{\sigma}^{22} - AUR22) (h_3 \Delta \theta^3)_k [(h_2 \Delta \theta^2)_e - (h_2 \Delta \theta^2)_w] \\
&\quad + (\bar{\sigma}^{33} - AUR33) (h_2 \Delta \theta^2)_j [(h_3 \Delta \theta^3)_e - (h_3 \Delta \theta^3)_w]
\end{aligned}
\tag{3.75}$$

Similarly, momentum equations for the other two directions may be obtained but are omitted for brevity.

G. PRESSURE CORRECTION

In the finite difference scheme, energy and momentum equations are used to solve for temperature and velocities. The equation of state and continuity are used to solve for density and pressure. Doria [Ref. 35] states that pressure is only weakly coupled to the equation of state. Therefore, updated temperatures and pressures determine density in the equation of state and continuity is used to correct pressure across each cell.

As discussed earlier, a disadvantage of using primitive variables is the difficulty in calculating pressure. Two corrections must be applied. First, a global pressure correction accounts for changes in net energy of the closed system. Second, a local pressure correction accounts for pressure changes causing the velocity field.

1. Global Pressure Correction

Nicolette, et al. [Ref. 3] developed a correction scheme for a two dimensional square enclosure. Raycraft [Ref.

30] modified it to fit the geometry of Fire-1. In a constant mass and volume system, the overall pressure depends on the addition or removal of energy. In such a system, the sum of all the cells' computed density times its volume is equal to a constant total mass. At any time during a run the mass must equal the total mass at equilibrium. Summing over N cells:

$$\sum \rho_i^n (\Delta V)_i = \sum \rho_{EQ_i} (\Delta V) \quad (3.76)$$

where n is the nth time step and the EQ subscript indicates the equilibrium point. Assuming that air is an ideal gas, its density is a function of temperature and pressure only. The actual values of both consist of the estimate and the global correction:

$$P = P^* + P'_g \quad (3.77)$$

$$T = T^* + T'_g \quad (3.78)$$

where P* and T* are the estimates and P_g' and T_g' are the global corrections using the ideal gas law and Equation (3.76). The global pressure correction becomes

$$P'_g = \frac{\sum P_{EQ} \left(\frac{\Delta V}{T_i} - \frac{\Delta V}{T^*} \right) - \sum \left(P^* \frac{\Delta V}{T^*} \right)}{\sum \frac{\Delta V}{T^*}} \quad (3.79)$$

Mass is conserved for each cell when an accurate final pressure is obtained.

2. Local Pressure Correction

Patanker [Ref. 36:pp. 120-126] and Doria [Ref. 36:pp. 26-32] developed a procedure for obtaining the local pressure correction. As in the global correction scheme, a pressure field is estimated from the previous time step. Velocities are calculated according to this pressure distribution and the law of continuity is applied to each cell. If the residual mass term S_{mp} approaches zero, then the estimated pressure field is satisfactory. If not, a local correction is calculated and applied to the original estimate. The new pressure field is used to compute a corrected velocity field and the residual mass S_{mp} is rechecked. The process repeats itself until S_{mp} is an acceptably small value. As in the global correction, the actual local pressure is:

$$P = P^* + P' \quad (3.80)$$

where P^* is again the estimate, usually the pressure of the preceding iteration, and P' is the local correction. Putting this correction in typical finite difference form:

$$A_p P_p' = A_E P_E' + A_W P_W' + A_N P_N' + A_S P_S' + A_F P_F' + A_B P_B' - S_{mp} \Delta V \quad (3.81)$$

where:

$$\begin{aligned}
 A_E &= \frac{\rho_e \cdot [(h_2 \Delta \theta^2) (h_3 \Delta \theta^3)]_e^2}{\left[A_{p_{e1}}^{u1} + \rho_e \frac{\Delta V}{\Delta t} \right]} \\
 A_W &= \frac{\rho_w \cdot [(h_2 \Delta \theta^2) (h_3 \Delta \theta^3)]_w^2}{\left[A_p^{u1} + \rho_w \frac{\Delta V}{\Delta t} \right]} \\
 A_N &= \frac{\rho_n \cdot [(h_1 \Delta \theta^1) (h_3 \Delta \theta^3)]_n^2}{\left[A_{p_{n1}}^{u2} + \rho_n \frac{\Delta V}{\Delta t} \right]} \\
 A_S &= \frac{\rho_s \cdot [(h_1 \Delta \theta^1) (h_3 \Delta \theta^3)]_s^2}{\left[A_p^{u2} + \rho_s \frac{\Delta V}{\Delta t} \right]} \\
 A_F &= \frac{\rho_f \cdot [(h_1 \Delta \theta^1) (h_2 \Delta \theta^2)]_f^2}{\left[A_{p_{f1}}^{u3} + \rho_f \frac{\Delta V}{\Delta t} \right]} \\
 A_B &= \frac{\rho_b \cdot [(h_1 \Delta \theta^1) (h_2 \Delta \theta^2)]_b^2}{\left[A_p^{u3} + \rho_b \frac{\Delta V}{\Delta t} \right]} \\
 A_p &= A_E + A_W + A_N + A_S + A_F + A_B
 \end{aligned} \tag{3.82}$$

Corrected velocities are:

$$\begin{aligned}
 u^1 &= u^{1*} + u^{1'} \\
 u^2 &= u^{2*} + u^{2'} \\
 u^3 &= u^{3*} + u^{3'}
 \end{aligned} \tag{3.83}$$

where:

$$\begin{aligned}u^{1'} &= \frac{(P_p + P_w) (h_2 \Delta \theta^2) (h_3 \Delta \theta^3)}{\left(A_p^{u^1} + \rho_w \frac{\Delta V}{\Delta t} \right)} \\u^{2'} &= \frac{(P_p + P_s) (h_1 \Delta \theta^1) (h_3 \Delta \theta^3)}{\left(A_p^{u^2} + \rho_s \frac{\Delta V}{\Delta t} \right)} \\u^{3'} &= \frac{(P_p + P_b) (h_1 \Delta \theta^1) (h_2 \Delta \theta^2)}{\left(A_p^{u^3} + \rho_b \frac{\Delta V}{\Delta t} \right)}\end{aligned}\tag{3.84}$$

Again S_{mp} is computed using continuity. If the residual mass is within a satisfactory range, the calculation is finished. If not, another iteration takes place.

IV. NUMERICAL PROCESS

A. INTRODUCTION

Temperature, velocity, pressure and density fields are produced by the code. Input parameters are initial conditions, fuel heat release rate, fire location, geometry and material characteristics such as fluid properties, wall properties and the external heat transfer coefficient. These are listed in Table 4.1.

TABLE 4.1 MODEL PARAMETERS

WALL CHARACTERISTICS	
Material	ASTM 285 Grade C Steel
Thickness	3/8 inch
Specific Heat	0.1 BTU/ (lbm•F)
Thermal Conductivity	25 BTU/ (hr•ft•F)
Density	487 lbm/ft ³
External Heat Transfer Coefficient	15.0 BTU/ (hr•ft ² •F)
FIRE CHARACTERISTICS	
Burn Rate	Function provided in program
Initial Temperature	35.6°C
Initial Pressure	1.0 ATM
Location of Fire	Center of Fire-1 23.1 ft. from each endccap 3.21 ft. from bottom

Figures 4.1 and 4.2 show the spherical/cylindrical grid used by the model. Endcaps are spherical with θ , R , and ϕ

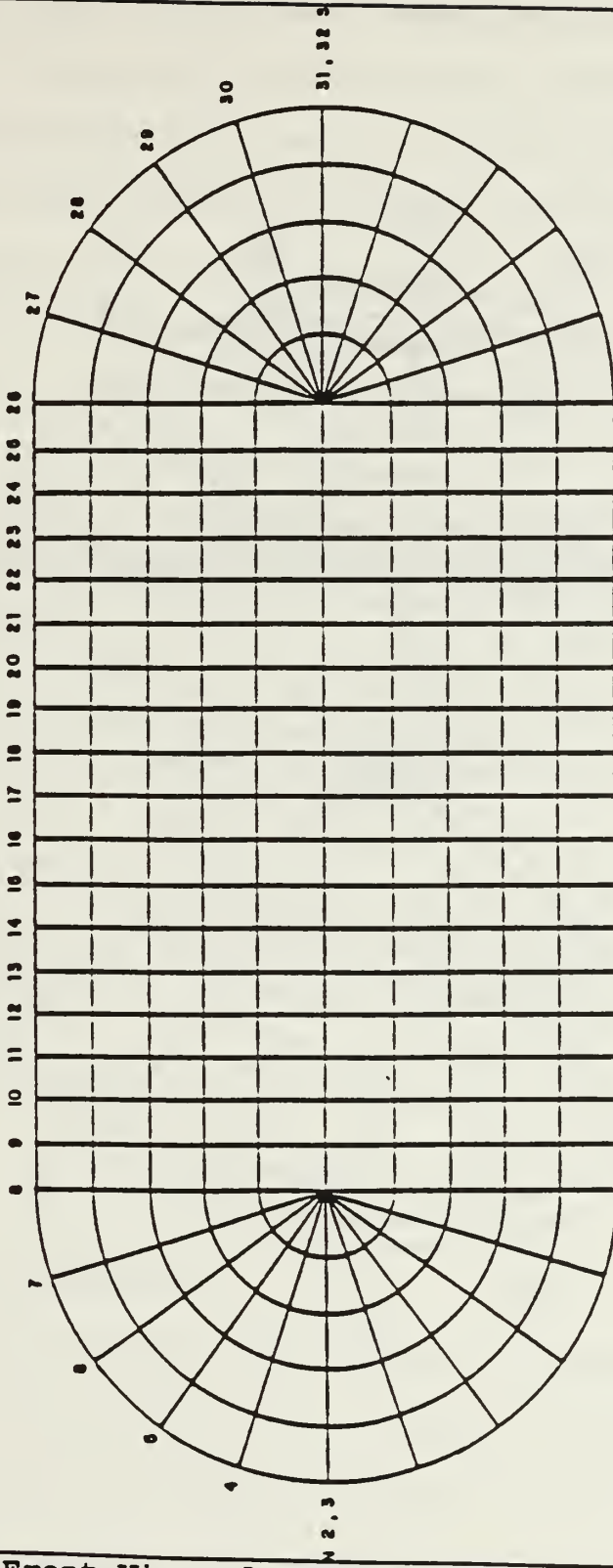


Figure 4.1 Front View of Computer Model (YZ-Plane)

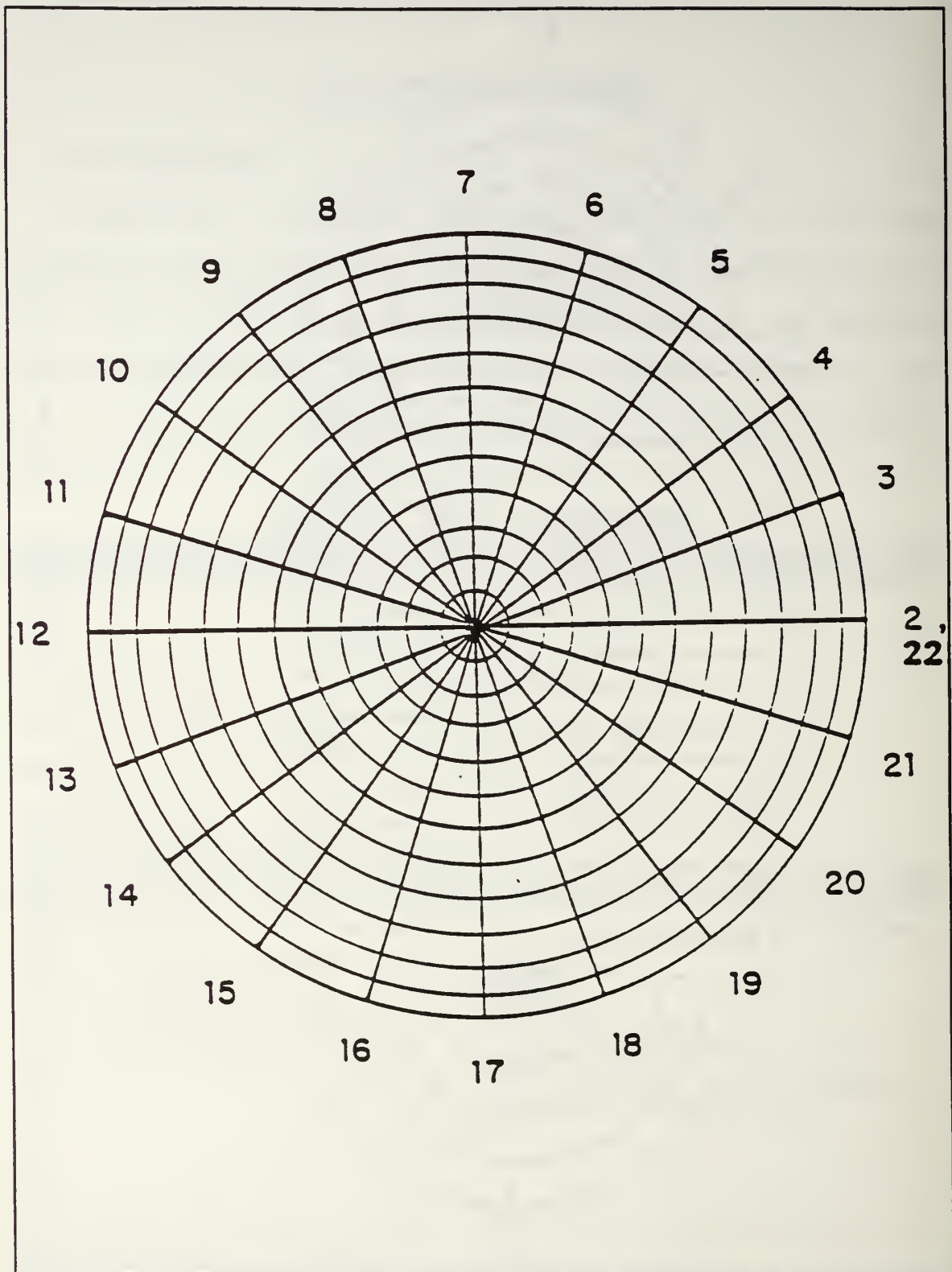


Figure 4.2 Side View of Computer Model (XY-Plane)

directions. The cylindrical midsection have θ , R and Z directions. There are 14 cells in the R direction, one at R=0 for avoiding singularity and one used as the vessel wall. There are 20 cells oriented clockwise in the θ direction. Each endcap has six cells in the ϕ direction with a cell again at zero to avoid singularity. The midsection has 18 cells in the Z direction (ϕ is used for simplicity). Table 4.2 gives information on grid parameters.

TABLE 4.2 ADDITIONAL MODEL PARAMETERS

GRID

Number of interior cells	6,720
Number of wall cells	560
Number of wall radiation zones	560
Number of fire radiation zones	19
Number of cells in R direction	14
Number of cells in θ direction	20
Number of cells in ϕ direction (per endcap)	6
Number of cells in Z direction (midsection)	18
Time step	0.0288 sec
VAXSTATION 3100 CPU time (1 CPU hour)	0.8-1.0 sec Fire Time

B. SOLUTION PROCESS

The model contains two separate programs. The first authored by Raycraft [Ref. 29] calculates the view factors for surface radiation. It produces a matrix of view factors. It is

used only once and its values are stored for use whenever called by the second program.

As described by Nies [Ref. 27], Raycraft [Ref. 29] and Houck [Ref. 30], the main program uses finite difference methods described previously to establish temperature, velocity, pressure and density fields. Initial parameters and the view factors are first read into the program. Geometry of the grid is then calculated and the fields are set to initial conditions. Next, effective viscosity is computed in subroutine CALVIS. Every two time steps, surface radiation flux is recalculated in subroutine RADHT. Subroutines CALT, GLOBE, CALU, CALX, CALW and CALP calculate temperature, the global pressure correction, the velocities and the local pressure correction. Using the corrected velocities, continuity is applied to each cell. If the residual mass RESORM is greater than 10.0 the program is unstable and stops. A smaller time step must be used. If RESORM is greater than a set tolerance level then the program iterates solution by recalculating velocities and pressures. Iterations continue until 1) RESORM is below tolerance levels, solution is reached and the program proceeds to next time step; 2) the maximum number of iterations is reached, or 3) RESORM is greater than 10.0 and the program is stopped.

C. GRAPHICAL ANALYSIS

The use of CA-DISSPLA™ [Ref. 31] has posed some difficult problems. The output from the computer model is in the spherical/cylindrical coordinate system created to simulate Fire-1. This output must be converted to cartesian coordinates in order to be manipulated by CA-DISSPLA™. Even after the conversion is completed the resulting irregularly spaced grid must be interpolated into a regularly spaced grid.

After some experimentation with grid interpolation schemes, a group of CA-DISSPLA™ subroutines are used to create a regularly spaced matrix. These subroutines interpolate values from a set number of neighbors. Care must be taken in choosing a grid size to ensure distortion of the field values does not occur and to ensure that the software will not zero data points with few close neighbors. A relatively course grid has been chosen (50 x 50 x 100) for graphics output. New data points created outside the enclosure have been set to ambient values to minimize distortion at the boundaries.

The VAXSTATION 3100 has proven to be an excellent machine. It has good numerical speed coupled with very sharp graphics capabilities. Future research of this numerical model has been greatly enhanced by the incorporation of this workstation.

The following figures are temperature and velocity profiles for times of 30, 60, and 90 seconds. They are two dimensional images of three dimensional phenomena. Each figure, whether temperature or velocity, presents an axial view (XY-plane) of the tank at the midplane and a longitudinal view (YZ-plane), again at the midplane.

Raycraft [Ref. 29] and Houck [Ref. 30] detailed the validation of the code against experimental data of Fire-1. They also discussed such phenomena as the fire plume, pressure effect, temperature stratification, and velocity fields. Much of their analysis will not be repeated here. Instead, the effects of local numerical perturbations will be discussed.

Raycraft [Ref. 29] observed remarkable symmetry in temperature and velocity profiles throughout the entire trial. Houck [Ref. 30] also observed the expected asymmetry, after implementing forced ventilation in two locations. In this thesis, these ventilation equations are not removed. The additive velocities were simply set to zero. As seen in Figures 4.3 to 4.8, a marked asymmetry similar to that observed in Houck [Ref. 30], has developed and is readily observed in both temperature and velocity profiles. This is despite the fact that the effects of ventilation have been supposedly negated. After the millions of calculations done by the computer to provide solutions, terms in the ventilation equations which are set to zero have greatly effected the entire field.

Color graphics have greatly enhanced ability to observe phenomena in the temperature fields. Temperatures can be quickly determined using the color legend, as Figures 4.3, 4.5, and 4.7 show. These figures have been printed on a Tektronics 4693D color printer and exhibit excellent clarity and resolution.

Three dimensional vector field representation of the velocity fields, Figures 4.4, 4.6 and 4.8 can only be represented in two dimensional form for this geometry. Results become confusing if three dimensions are shown.

TEMPERATURE PROFILE: 30 SEC

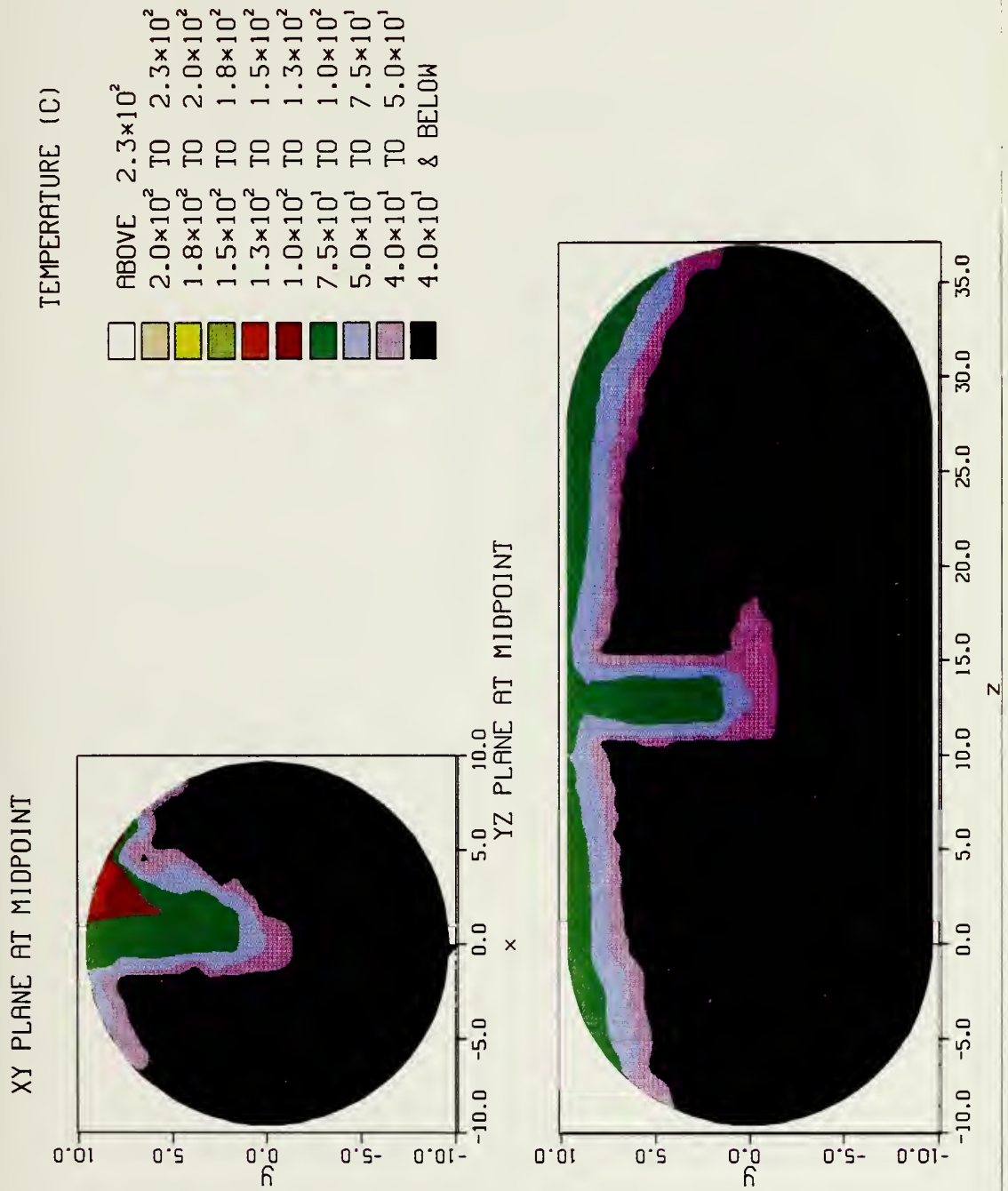


Figure 4.3 Temperature Profiles at 30 Seconds

VELOCITY PROFILE 30 SEC

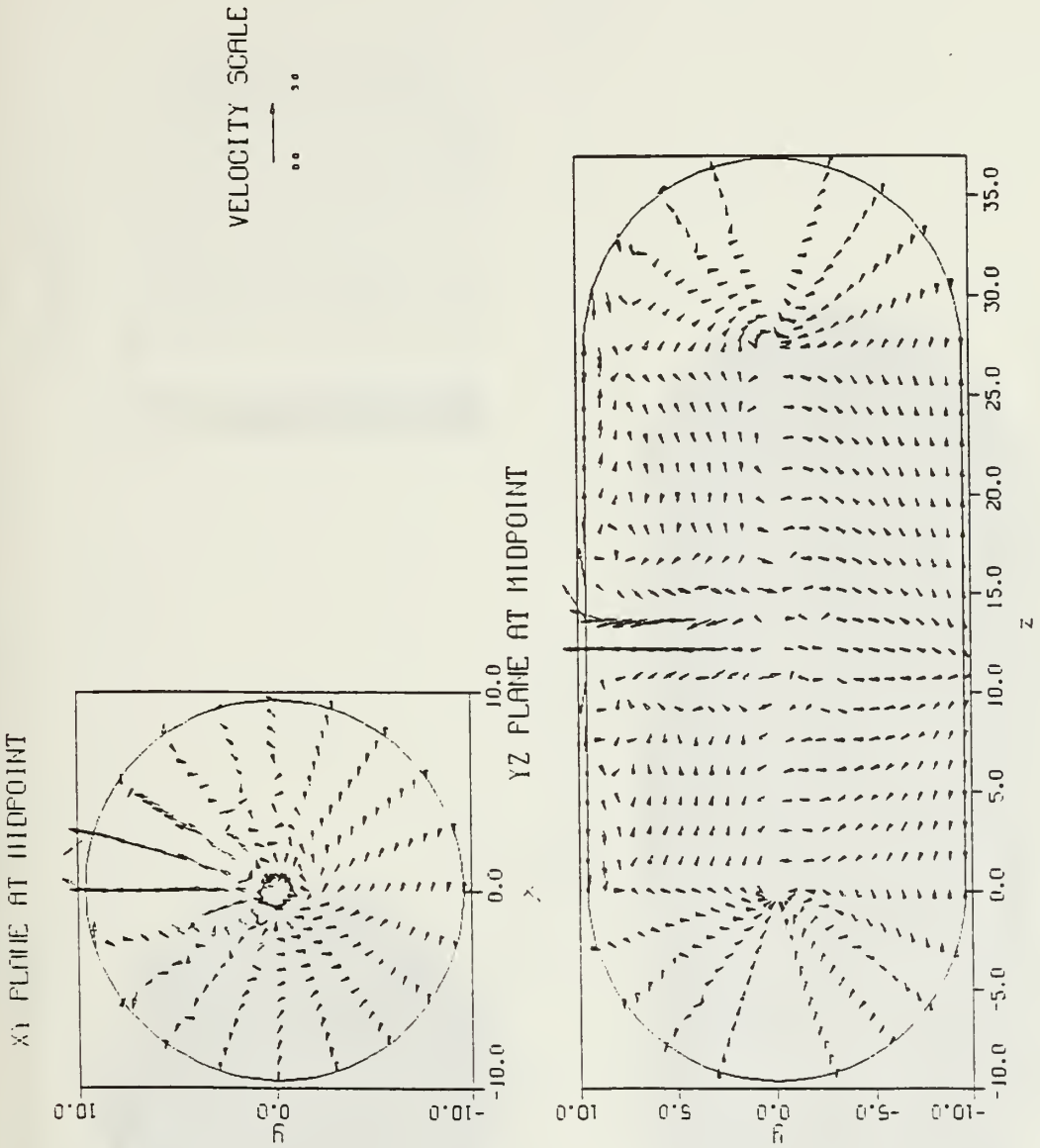


Figure 4.4 Velocity Profile at 30 Seconds

TEMPERATURE PROFILE: 60 SEC

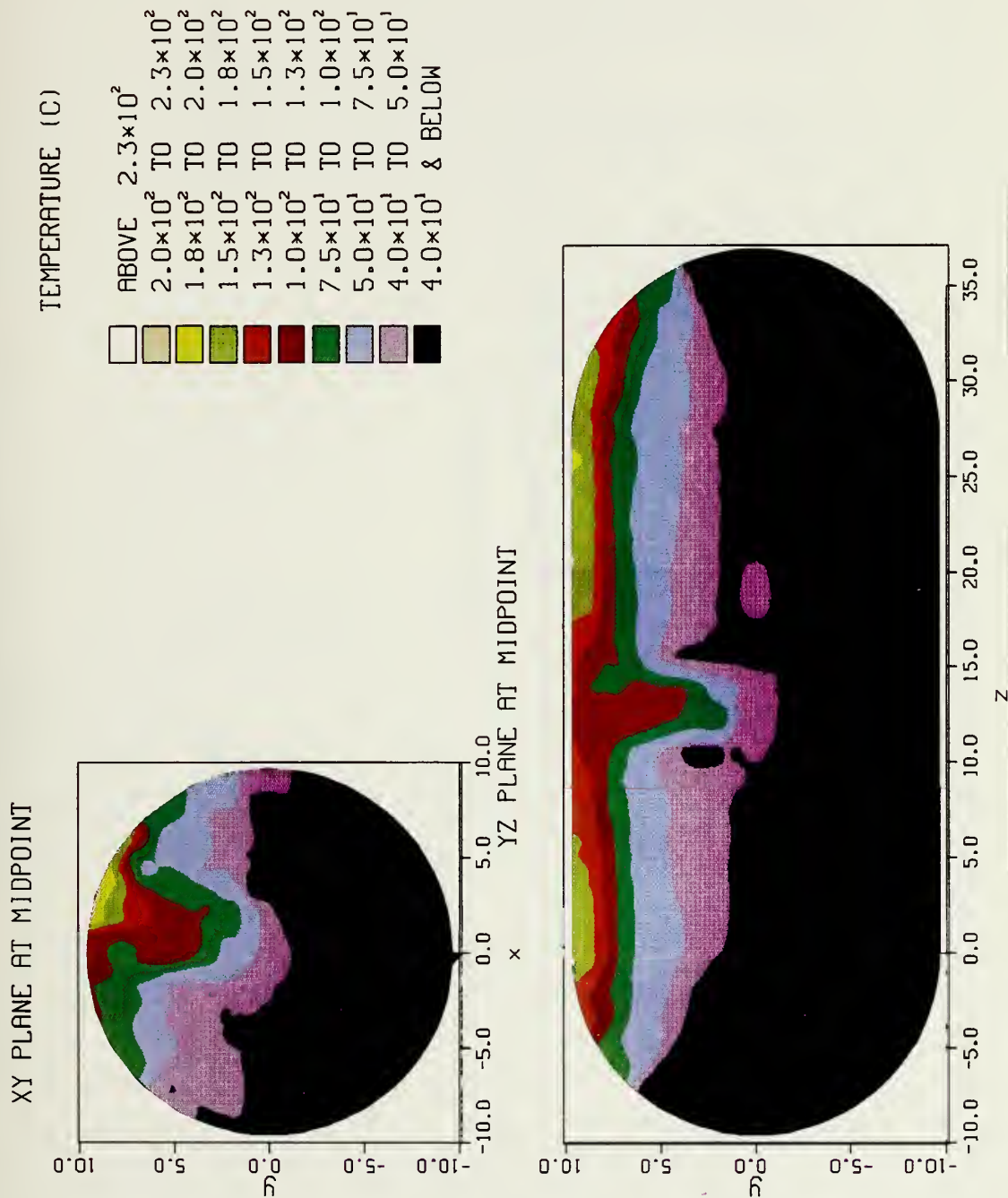


Figure 4.5 Temperature Profiles at 60 Seconds

VELOCITY PROFILE 60 SEC

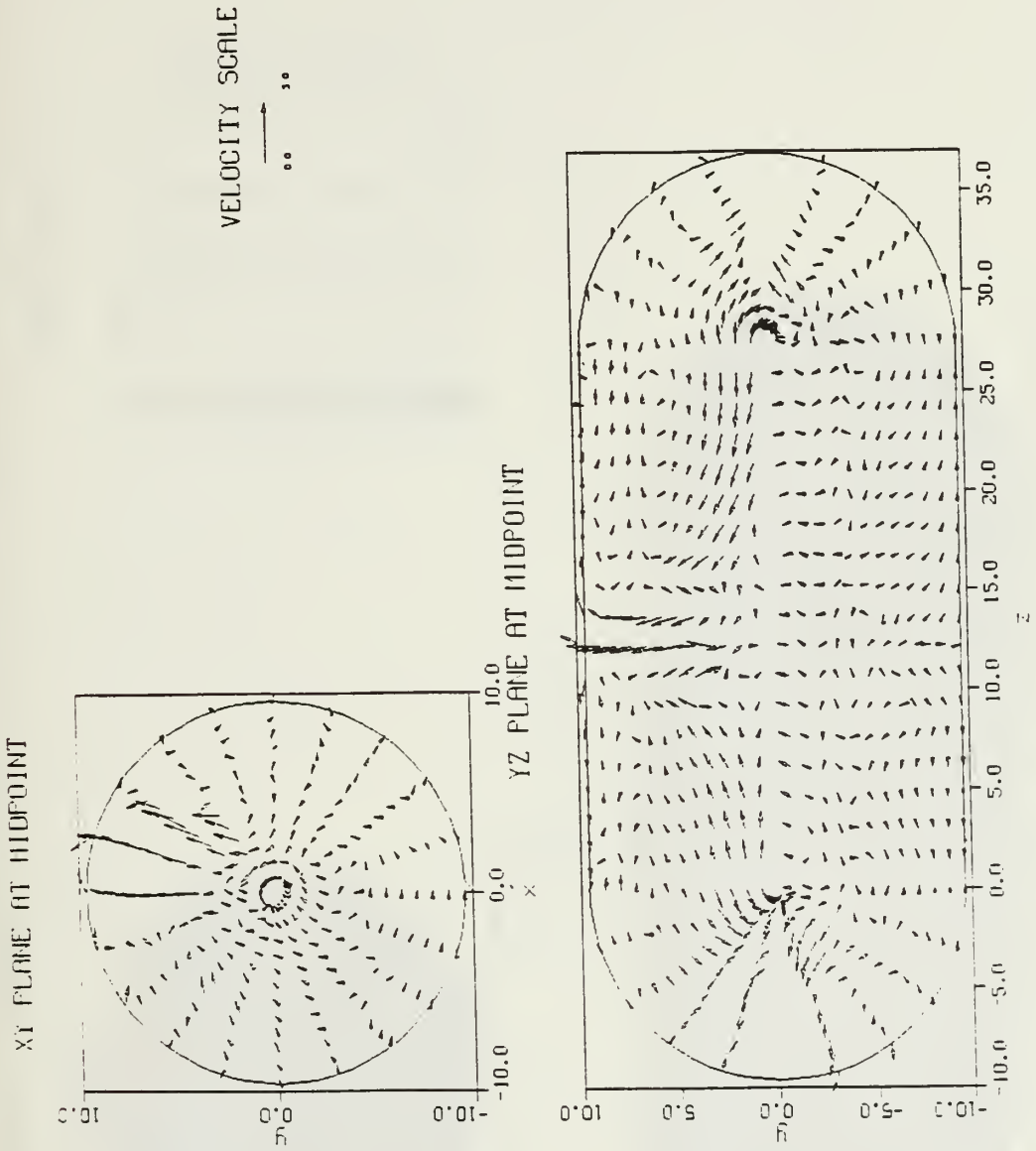


Figure 4.6 Velocity Profile at 60 Seconds

TEMPERATURE PROFILE: 90 SEC

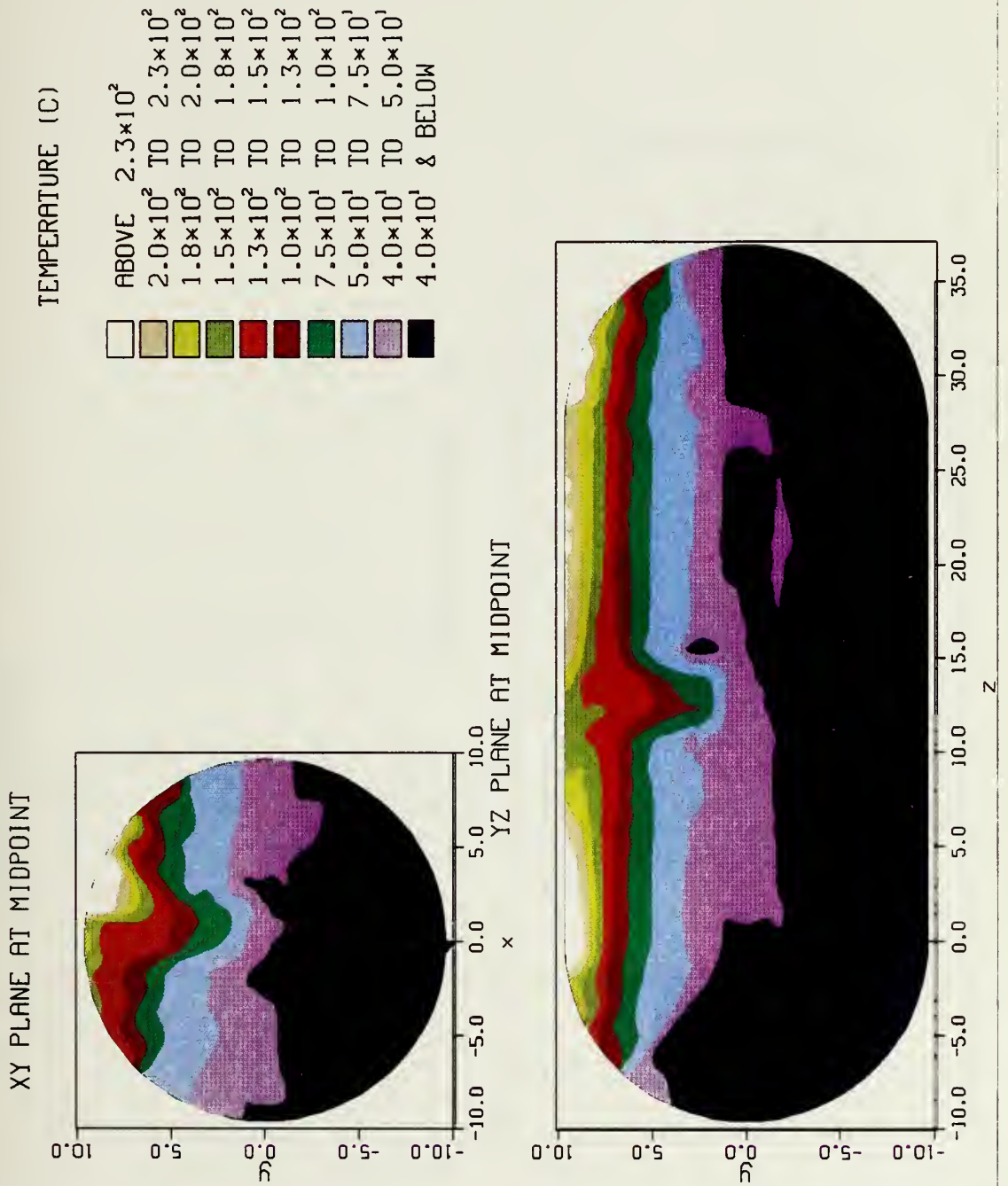


Figure 4.7 Temperature Profiles at 90 Seconds

VELOCITY PROFILE 90 SEC

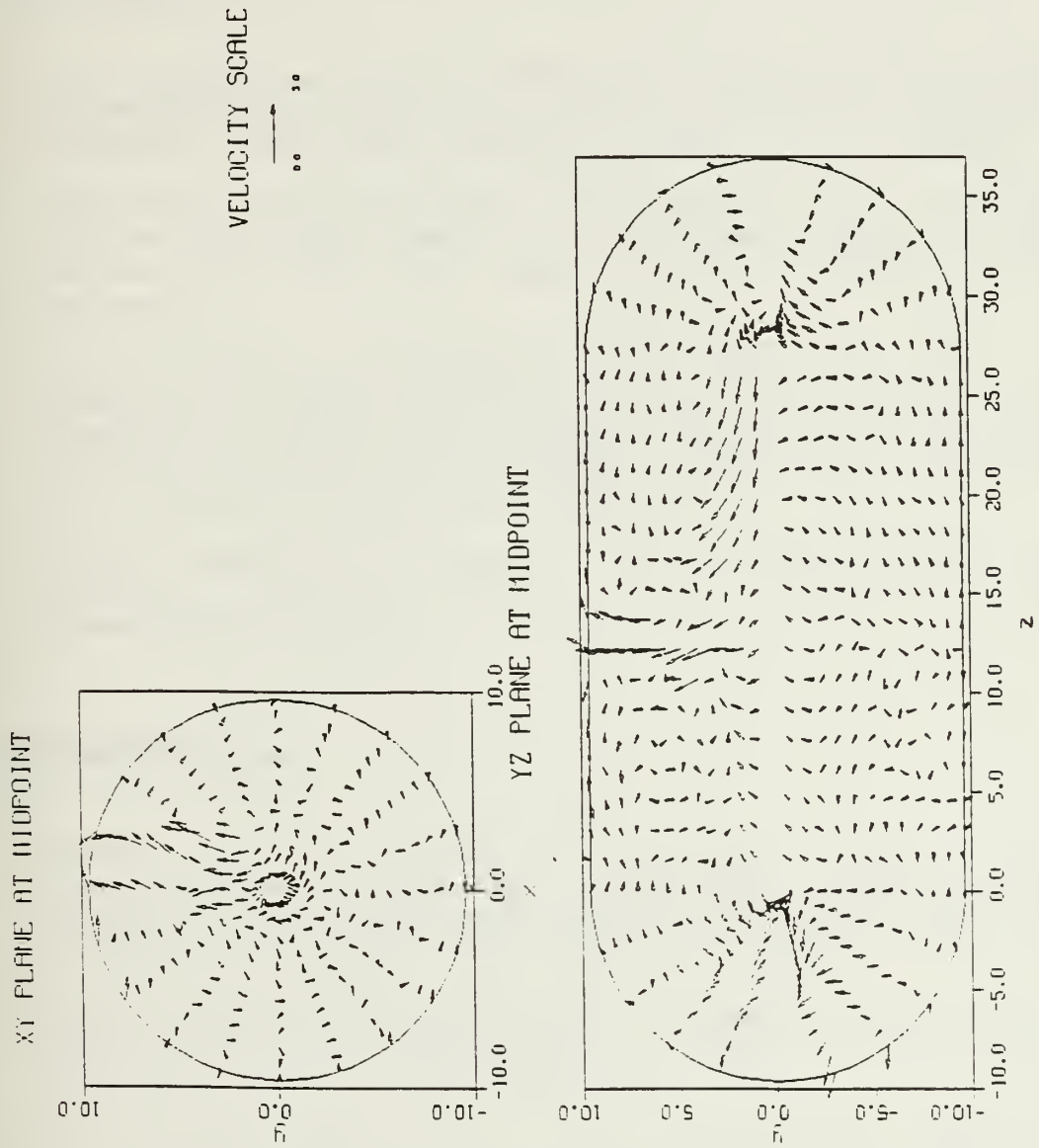


Figure 4.8 Velocity Profile at 90 Seconds

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

1. The acquisition of the VAXSTATION 3100 SPX XRJ19 Model 38 workstation with its blend of numerical speed and graphics clarity has greatly enhanced the research.
2. The ventilation equations incorporated into the model in the previous thesis have a great effect on the entire field even when their output velocities are set to zero.
3. Color graphics have provided an excellent means for presenting temperature profile data. Coupled with the Tektronics 4693 color print, CA-DISSPLA™ Graphics Software provides researchers with an excellent tool for displaying scalar data fields.
4. Three dimensional vector fields are difficult to present, ambiguous, and must be reduced to two dimensional images.

B. RECOMMENDATIONS

1. Removal of the ventilation equations is required to regain symmetry observed in previous research. These equations are effecting the entire field although their additive velocities have been set to zero.
2. More sophisticated physical models need to be formulated and incorporated, such as turbulence, gaseous radiation and combustion.
3. Streakline analysis in three dimensions should be conducted to show the path taken of an individual fluid particle as it leaves the flame area. This method may reveal more of the fluid dynamics than current representations of velocity vector fields.
4. The ultimate goal of this project is to develop a model which can predict behavior of fire in shipboard

situations, for example, changing the geometry to fit machinery spaces and berthing compartments. this will offer designers a valuable tool for the construction of safer ships and submarines.

APPENDIX

```

C ***** 00000100
C ** ** 00000200
C ** THREE-DIMENSIONAL NUMERICAL SIMULATION ** 00000300
C ** OF A FIRE SPREAD INSIDE A NAVY STORAGE TANK ** 00000400
C ** ** 00000500
C ** DEVELOPED BY : ** 00000600
C ** H.Q. YANG AND K.T. YANG ** 00000700
C ** ** 00000800
C ** DEPARTMENT OF AEROSPACE & MECHANICAL ENGINEERING ** 00000900
C ** UNIVERSITY OF NOTRE DAME ** 00001000
C ** NOTRE DAME, INDIANA, 46556 ** 00001100
C ** ** 00001200
C ** DEC. 1986 ** 00001300
C ** ** 00001400
C ***** 00001500
C ***** 00001600
C COMMON/R4/XC (93), YC (93), ZC (93), XS (93), YS (93), ZS (93),
& DXXC (93), DYXC (93), DZZC (93), DXXS (93), DYYS (93), DZZS (93) 00001700
COMMON/BL1/DX, DY, DZ, VOL, DTIME, VOLDT, THOT, TCOOL, P1, Q, QR 00001800
COMMON/BL7/NI, NIP1, NIM1, NJ, NJP1, NJM1, NK, NKP1, NKM1 00001900
& , NIP2, NJP2, NKP2, NA, NAP1, NAM1, NB, NBP1, NBM1, KRUN, NCHIP, NJRA, NWRP 00002000
COMMON/BL12/ NWRITE, NTAPE, NTMAX0, NTREAL, TIME, SORSUM, ITER 00002100
COMMON/BL14/HCOEF, TINF, CNT, ABTURB, BTURB, VISL, VISMAX, QCORRT, PM1, PM2 00002200
COMMON/BL16/ CONST1, CONST2, CONST3, CONST4, CONST6, NT, UO, H, UGRT, BUOY, 00002300
& CPO, PRT, CONDO, VIS0, RHOO, HR, TR, TA, DTEMP, TWRITE, TTAPE, TMAX, GC, RAIRO 00002400
COMMON/BL20/SIG11 (22, 16, 32), SIG12 (22, 16, 32), SIG22 (22, 16, 32) 00002500
& , SIG13 (22, 16, 32), SIG23 (22, 16, 32), SIG33 (22, 16, 32) 00002600
COMMON/BL22/ICHPB (10), NCHPI (10), JCHPB (10), NCHPJ (10), KCHPB (10),
& NCHPK (10), TCHP (10), CPS (10), CONS (10), WFAN (10) 00002700
COMMON/BL31/ TOD (22, 16, 32), ROD (22, 16, 32), POD (22, 16, 32) 00002800
& , COD (22, 16, 32), UOD (22, 16, 32), VOD (22, 16, 32), WOD (22, 16, 32) 00002900
COMMON/BL32/ T (22, 16, 32), R (22, 16, 32), P (22, 16, 32) 00003000
& , C (22, 16, 32), U (22, 16, 32), V (22, 16, 32), W (22, 16, 32) 00003100
COMMON/BL33/ TPD (22, 16, 32), RPD (22, 16, 32), PPD (22, 16, 32) 00003200
& , CPD (22, 16, 32), UPD (22, 16, 32), VPD (22, 16, 32), WPD (22, 16, 32) 00003300
COMMON/BL34/ HEIGHT (22, 16, 32), REQ (22, 16, 32),
& SMP (22, 16, 32), SMPP (22, 16, 32), PP (22, 16, 32),
& DU (22, 16, 32), DV (22, 16, 32), DW (22, 16, 32) 00003400
COMMON/BL36/AP (22, 16, 32), AE (22, 16, 32), AW (22, 16, 32), AN (22, 16, 32),
& AS (22, 16, 32), AF (22, 16, 32), AB (22, 16, 32),
& SP (22, 16, 32), SU (22, 16, 32), RI (22, 16, 32) 00003500
COMMON/BL37/ VIS (22, 16, 32), COND (22, 16, 32), NOD (22, 16, 32), RWALL (579) 00003600
& , CPM (22, 16, 32), HSZ (3, 2), NHSZ (22, 16, 32), RESORM (93) 00003700
COMMON/BL38/NTHCO, CX (12), CY (12), CZ (12), NTH (12, 3), TCOUP (12) 00003800
COMMON/BL39/ALEW, PCURVE, CONSRA, PCURM1, PSOUTH, QCORR, PERROR 00003900
DIMENSION VFMXC (579, 579), T4WALL (579) 00004000
DATA X, ILEFT, SORMAX, XTIME, ITMAX/20, 400000, 0.40, 0.0, 4/ 00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
C *** UO : REFERENCE VELOCITY (FT/SEC), 1 FT/SEC 00005200
C *** RHOO : REFERENCE AIR DENSITY (LBM/FT**3) 00005300
C *** H : REFERENCE LENGTH (FT) 00005400
C *** TA : REFERENCE TEMPERATURE (R) 00005500
C *** TINIT : INITIAL TEMPERATURE (O) 00005600
C *** GC : GRAVITATIONAL CONSTANT 00005700
C *** RAITR : GAS CONSTANT; 53.34 00005800
C *** CONST1 : RA*UO**2/GC 00005900
C *** CONST3 : INVERSE OF TA 00006000
C *** CONST4 : REFERENCE LENGTH (CM) 00006100

```

```

C *** CONST6 : REFERENCE VELOCITY (CM/S) 00006200
C *** CONSRA : TA**3/(RA*CP*JO*H*H) 00006300
C *** NTRWR : NTREAL/NWRITE*NWRITE 00006400
C *** NTRWA : NTREAL/NWALT*NWALT 00006500
C *** HCONV : HEAT TRANSFER COEFFICIENT ON THE AMBIENT (BTU/H.FT**2K) 00006600
C 00006700
C 00006800
C 00006900
C *** RAD,H: RADIUS OF THE CYLINDRICAL AND SPHERICAL SECTIONS 00007000
C CYL : LENGTH OF THE CYLINDRICAL SECTION OF THE TANK 00007100
C *** NI : TOTAL NUMBER CELLS IN THETA-DIRECTION 00007200
C NJ : R-DIRECTION 00007300
C NK : Z AND PHI-DIRECTIONS 00007400
C NA : FIRST NUMBER Z-DIRECTION, ALONG THE CYLINDER AXIS 00007500
C NB : LAST NUMBER Z-DIRECTION, ALONG THE CYLINDER AXIS 00007600
C *** HSZ(1,1),HSZ(1,2) FIRST AND LAST COORDIANTE OF HEAT SOURCE 00007700
C IN X-DIRECTION (IN DIMENSIONLESS FORM) 00007800
C HSZ(2,1),HSZ(2,2) FIRST AND LAST COORDIANTE OF HEAT SOURCE 00007900
C IN Y-DIRECTION (IN DIMENSIONLESS FORM) 00008000
C HSZ(3,1),HSZ(3,2) FIRST AND LAST COORDIANTE OF HEAT SOURCE 00008100
C IN Z-DIRECTION (IN DIMENSIONLESS FORM) 00008200
C 00008300
C *** ICHPB() : STARTING NODAL NUMBER FOR SOLID IN THETA-DIRECTION 00008400
C JCHPB() : R-DIRECTION 00008500
C KCHPB() : Z OR PHI-DIRECTION 00008600
C *** NCHPI() : NUMBER OF NODALS FOR SOLID IN THETA-DIRECTION 00008700
C NCHPJ() : R-DIRECTION 00008800
C NCHPK() : Z,PHI-DIRECTION 00008900
C ***** 00009000
C open(21,file='input.dat',status='old')
C ***** 00009200
C INPUT DATA & 00009300
C ***** 00009400
C write(6,*) 'calling input'
C CALL INPUT 00009500
C 00009600
C ***** 00009700
C GENERATE GRID SYSTEM & 00009800
C ***** 00009900
C write(6,*) 'calling grid'
C CALL GRID 0010000
C 0010100
C ***** 0010200
C *** READ VIEW FACTOR INVERSE MATRIX * 0010300
C ***** 0010400
C open(11,file='view.dat',status='old')
C do 225 i=1,579
C do 225 j=1,579
C 225 read(11,*) vfmxc(i,j)
C CLOSE (11) 0010900
C 0011000
C ***** 0011100
C INITIALIZE THE WHOLE FIELD & 0011200
C ***** 0011300
C write(6,*) 'calling init'
C CALL INIT 0011400
C 0011500
C ***** 0011600
C START CALCULATION & 0011700
C ***** 0011800
C 0011900
C 0012000
C 0012100
C 0012200
C 300 CONTINUE 0012300
C 0012400
C 0012500
C 0012600

```

```

C ***      NTMAX0 HAS THE MEANING AS "NTREAL" WHEN IT IS READ FROM      00012700
C          DISK OR TAPE.                                               00012800
                                                                    00012900
                                                                    00013000
                                                                    00013100
                                                                    00013200
                                                                    00013300
IF(XTIME .GT. TMAX) GO TO 303
NTREAL=NT+NTMAX0
TIME=TIME+DTIME
XTIME=TIME*H/UC
nxtime=jint(xtime)
ntwrit=jint(twrite)
write(6,*) 'time in seconds=',xtime                                00013400
write(6,*) 'int time=',nxtime
write(6,*) 'int time for writing=',ntwrit
                                                                    00013500
C#####                                                                    00013600
C  CALCULATE THE TRANSIENT HEAT INPUT &                               00013700
C  NOTE IF 1 IN PARENTHESIS, THE BURN RATE IS CALCULATED &          00013800
C  BY THE PRESSURE CURVE. IF EQUAL TO TWO, THE BURN RATE &         00013900
C  CURVE IS EITHER GIVEN OR ESTIMATED &                            00014000
C#####                                                                    00014100
      write(6,*) 'calling calq'
      CALL CALQ(2)
                                                                    00014200
C ***      START CALCULATION                                           00014300
                                                                    00014400
      ITER=0                                                           00014500
      JTERM=0                                                           00014600
      JJTERM=0                                                         00014700
                                                                    00014800
                                                                    00014900
C  DEFINE THE UPDATED TPD(I,J,K), CPD(I,J,K),RPD(I,J,K)              00015000
C  CPD(I,J,K) AND VPD(I,J,K) FOR THE USE OF CALVIS AND SU(I,J,K)    00015100
                                                                    00015200
      DO 48 K=1,NKP1
      DO 48 J=1,NJP1
      DO 48 I=1,NIP1
      TPD(I,J,K)=T(I,J,K)
      CPD(I,J,K)=C(I,J,K)
      RPD(I,J,K)=R(I,J,K)
      CPD(I,J,K)=U(I,J,K)
      VPD(I,J,K)=V(I,J,K)
      WPD(I,J,K)=W(I,J,K)
      48 CONTINUE
      29 CONTINUE
      JTERM=JTERM+1
      301 CONTINUE
                                                                    00015300
                                                                    00015400
                                                                    00015500
                                                                    00015600
                                                                    00015700
                                                                    00015800
                                                                    00015900
                                                                    00016000
                                                                    00016100
                                                                    00016200
                                                                    00016300
                                                                    00016400
                                                                    00016500
                                                                    00016600
                                                                    00016700
C#####                                                                    00016800
C  CALCULATE THE RADIATION HEAT FLUX AT EVERY NRAD TIME STEPS &    00016900
C#####                                                                    00017000
      NRAD = 2
      IF (MOD(NT,NRAD).NE.0) GOTO 4000
      CALL RADHT(T4WALL,VFMXC)
      4000 CONTINUE
                                                                    00017100
                                                                    00017200
                                                                    00017300
                                                                    00017400
                                                                    00017500
                                                                    00017600
                                                                    00017700
C#####                                                                    00017800
C  CALCULATE THE TEMPERATURE *
C#####                                                                    00017900
      write(6,*) 'calling calt'
      CALL CALT
                                                                    00018000
                                                                    00018100
C#####                                                                    00018200
C  CALCULATE THE SMOKE CONCENTRATION &
C#####                                                                    00018300
      write(6,*) 'calling calc'
      CALL CALC
                                                                    00018400
                                                                    00018500
                                                                    00018600
                                                                    00018700
      DO 2000 J=1,NJP1

```

```

DO 2000 I=1,NIP1
DO 2000 K=1,NKP1
IF(T(I,J,K).LT.TCOOL) T(I,J,K)=TCOOL
2000 CONTINUE
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C GLOBLE PRESSURE CORRECTION FOR ENCLOSED TANK AIR %
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
write(6,*) 'calling globe'
CALL GLOBE
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C CALCULATE THE TURBULENT VISCOSITY AND CONDUCTIVITY @
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
write(6,*) 'calling calvis'
CALL CALVIS
C*****
C CALCULATE THE DENSITY *
C*****
DO 100 J=1,NJP1
DO 100 I=1,NIP1
DO 100 K=1,NKP1
IF (NOD(I,J,K).EQ.1) GOTO 100
AAAA=BUOY*UGRT*HEIGHT(I,J,K)
R(I,J,K)=(UGRT*P(I,J,K)+(1./EXP(AAAA)))/T(I,J,K)
100 CONTINUE
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C CORRECT CONDUCTIVITY OF THE SOLID $
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
IF (NCHIP.EQ.0) GOTO 410
write(6,*) 'calling solcon'
CALL SOLCON
410 CONTINUE
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C START PRESSURE CORRECTION ITERATIVE LOOP, IT IS THE MAJOR %
C PART OF THE ERROR CONTROL ROUTINE %
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ITER=ITER-1
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C CALCULATE THE VELOCITY U,V,AND W @
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
write(6,*) 'calling velocities'
00023000
CALL CALU
CALL STRESS
C *****
CALL CALV
CALL STRESS
C *****
CALL CALW
write(6,*) 'wfan(1)=',wfan(1)
CALL STRESS
C *****
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C CALCULATE THE PRESSURE AND STRESS &
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
write(6,*) 'calling calp'
CALL CALP
CALL STRESS
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C IF SOURCE TERM IS LARGER THAN 10.0, STOP PROGRAM %

```

```

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  IF (RESORM(ITER).GT.10.0) GOTO 2020

  IF(RESORM(ITER) .LE. SORMAX) GO TO 49
  IF(ITER .EQ. 1) GO TO 302
  ITERM1=ITER-1
  IF(RESORM(ITER) .LE. RESORM(ITERM1)) GO TO 302
  GO TO 304
302 IF(JTERM .GE. 2) GO TO 37
  SOURCE=RESORM(ITER)
  GO TO 39
 37 IF(RESORM(ITER) .LE. SOURCE) GO TO 38
  GO TO 304
 38 SOURCE=RESORM(ITER)
 39 CONTINUE
C   WRITE(6,95) ITER,RESORM(ITER),SORSUM
 95 FORMAT(53X,' ITER=', I2,2X,' SOURCE=', F9.6,2X,' SORMUP=', F9.6)
  DO 23 K=1,NKP1
  DO 23 J=1,NJP1
  DO 23 I=1,NIP1
  TPD(I,J,K)=T(I,J,K)
  CPD(I,J,K)=C(I,J,K)
  RPD(I,J,K)=R(I,J,K)
  UPD(I,J,K)=U(I,J,K)
  VPD(I,J,K)=V(I,J,K)
  WPD(I,J,K)=W(I,J,K)
  PPD(I,J,K)=P(I,J,K)
 23 CONTINUE
  JJTERM=0
  IF(ITER .EQ. ITMAX) GO TO 49
  IF(JJTERM .EQ. 2) GO TO 35
  IF(ITER .EQ. 4) GO TO 29
 35 CONTINUE
  IF(JJTERM .EQ. 3) GO TO 58
  IF(ITER .EQ. 7) GO TO 29
 58 CONTINUE
  JJTERM=0
  GO TO 301
304 CONTINUE
  JJTERM=JJTERM+1
C   IF(JJTERM .EQ. 1) WRITE(6,95) ITER,RESORM(ITER),SORSUM
  IF(JJTERM .EQ. 1) GO TO 41
  IF(JJTERM .EQ. 2 .AND. JJTERM .EQ. 1 .AND. ITER .NE. 5) GO TO 41
  GO TO 32
 41 CONTINUE
  DO 40 K=1,NKP1
  DO 40 J=1,NJP1
  DO 40 I=1,NIP1
  R(I,J,K)=RPD(I,J,K)
  U(I,J,K)=UPD(I,J,K)
  V(I,J,K)=VPD(I,J,K)
  W(I,J,K)=WPD(I,J,K)
  P(I,J,K)=PPD(I,J,K)
 40 CONTINUE
  IF(ITER .EQ. ITMAX) GO TO 49
  GO TO 29
 82 CONTINUE
  DO 43 K=1,NKP1
  DO 43 J=1,NJP1
  DO 43 I=1,NIP1
  T(I,J,K)=TPD(I,J,K)
  C(I,J,K)=CPD(I,J,K)
  R(I,J,K)=RPD(I,J,K)
  U(I,J,K)=UPD(I,J,K)
  V(I,J,K)=VPD(I,J,K)
  W(I,J,K)=WPD(I,J,K)
  P(I,J,K)=PPD(I,J,K)

```

```

00025000
00025100
00025200
00025300
00025400
00025500
00025600
00025700
00025800
00025900
00026000
00026100
00026200
00026300
00026400
00026500
00026600
00026700
00026800
00026900
00027000
00027100
00027200
00027300
00027400
00027500
00027600
00027700
00027800
00027900
00028000
00028100
00028200
00028300
00028400
00028500
00028600
00028700
00028800
00028900
00029000
00029100
00029200
00029300
00029400
00029500
00029600
00029700
00029800
00029900
00030000
00030100
00030200
00030300
00030400
00030500
00030600
00030700
00030800
00030900
00031000
00031100
00031200
00031300
00031400
00031500
00031600
00031700

```

```

43 CONTINUE                                00031800
IF(ITER.EQ.ITMAX) GO TO 49                 00031900
IF((JTERM.EQ.3.AND.ITER.NE.8).OR.JJTERM.EQ.2) GO TO 49 00032000
GO TO 301                                  00032100
49 CONTINUE                                00032200
                                           00032300
      ITERS=ITERS+ITER                    00032400
C#####                                00032500
C   GO TO THE PRESSURE TRACKING SUBROUTINE ,PRINT OUT # 00032600
C   RESULTS IF AT THE RIGHT TIME INTERVAL #          00032700
C#####                                00032800
      write(6,*) 'calling ptrack'
                                           00032900
      CALL PTRACK
      IF (MOD(ntreal,NWRP).EQ.0) CALL OUT(1)          00033000
                                           00033100
                                           00033200
C#####                                00033300
C   FIND TEMPERATURES AT THERMOCOUPLE POINTS AND PRINT OUT % 00033400
C   IF AT THE RIGHT TIME INTERVAL %                00033500
C#####                                00033600
      if (nthco.eq.0) goto 2422
      CALL TCP
      IF (MOD(NTREAL,NWRP).EQ.0) CALL OUT(2)          00033800
2422 CONTINUE                                00033900
      IF (MOD(nxtime,ntwrit).EQ.0) CALL OUT(3)       00034000
00034100
C   IF(NTREAL.EQ.NTREAL/NWRITE*NWRITE) CALL OUT(3) 00034200
505 CONTINUE                                00034300
      IF((XTIME-DTIME*H/U0).GE.TMAX) GO TO 277      00034400
                                           00034500
C *** #####                                00034600
C   CALL TLEFT(IT)                                00034700
C 123 FORMAT(' TLEFT = ',I10)                   00034800
C   ITO=IT                                         00034900
C   IF(IT.LT.ITLEFT) CALL OUT(3)                 00035000
C *** #####                                00035100
                                           00035200
                                           00035300
C ***   RESET THE OLD TIME VALUES TOD, ROD, UOD, VOD AND POD. 00035400
                                           00035500
      DO 305 K=1,NKP1                          00035600
      DO 305 J=1,NJP1                          00035700
      DO 305 I=1,NIP1                          00035800
      TOD(I,J,K)=T(I,J,K)                    00035900
      COD(I,J,K)=C(I,J,K)                    00036000
      ROD(I,J,K)=R(I,J,K)                    00036100
      UOD(I,J,K)=U(I,J,K)                    00036200
      VOD(I,J,K)=V(I,J,K)                    00036300
      WOD(I,J,K)=W(I,J,K)                    00036400
      POD(I,J,K)=P(I,J,K)                    00036500
305 CONTINUE                                00036600
                                           00036700
C #####                                00036800
C   THIS WRITING IS FOR PLOTTINGS                00036900
C #####                                00037000
C   IF(NTREAL.NE.NTREAL/NTAPE*NTAPE)GOTO 522    00037100
C   IWRITE=10                                    00037200
C   WRITE(9,*)                                  00037300
C   & TIME,NTREAL,T,R,U,V,W,P,CPM,COND,VIS,QRNET,ITERT,QCORRE,PM1,PM2, 00037400
C   & H,TA,CC,CONDO,VISCO,RHOO,NI,NJ,NK,NIP1,NJP1,NKP1,NIX1,NJX1,NKX1, 00037500
C   & XC,YC,ZC,XS,YS,ZS,DXXC,DYYC,DZZC,DXXS,DYYS,DZSZ 00037600
C   WRITE(6,*) 'THE TIME WHEN THE DATA WAS STORED ON DISK IS:', 00037700
C   & XTIME                                     00037800
C #####                                00037900
                                           00038000
                                           00038100
                                           00038200
522 CONTINUE                                00038300

```

```

C *** ***** 00038400
C CALL TLEFT(IT) 00038500
C IF(IT.LT.ITLEFT) GO TO 166 00038600
C *** ***** 00038700
C TIMREM IS USED TO CALCULATE THE CPU TIME REMAINING AT NPS 00038800
C 00038900
c IF (TIMREM(0.).LE.80.) GOTO 166 00039100
do 222 k=1,nkpl
do 222 i=1,nipl
do 222 j=1,njpl
WRITE(9,555) t(i,j,k),u(i,j,k),v(i,j,k),w(i,j,k)
write(9,555) p(i,j,k),cpm(i,j,k),cond(i,j,k),vis(i,j,k)
222 continue
write(9,556) time,qr,qcorrt,pml,pm2,xxxxx
write(9,556) n,ta,u0,cond0,vis0,rho0
write(9,557) ntreal,ni,nj,nk,nipl,njpl,nkpl,niml,njml,nkml,iter
write(9,556) xc,yc,zc,xs,ys,zs
write(9,556) dxxc,dyyc,dzxc,dxxs,dyys,dzss
555 format(4(3x,e12.4))
556 format(6(1x,e10.3))
557 format(11i4)
REWIND 9 00039200
GO TO 300 00039300
303 CONTINUE 00039400
277 CONTINUE 00039500
WRITE(6,1111) 00039600
1111 FORMAT(2X,'***** THE MAXIMUM TIME HAS BEEN REACHED *****',I8) 00039700
c GO TO 172 00039800
C *** ***** 00039900
c 166 IF(NTREAL .NE. NTREAL/NTAPE*NTAPE) then 00040000
c234567 00040100
do 223 k=1,nkpl
do 223 i=1,nipl
do 223 j=1,njpl
WRITE(9,555) t(i,j,k),u(i,j,k),v(i,j,k),w(i,j,k)
write(9,555) p(i,j,k),cpm(i,j,k),cond(i,j,k),vis(i,j,k)
223 continue
write(9,556) time,qr,qcorrt,pml,pm2,xxxxx
write(9,556) n,ta,u0,cond0,vis0,rho0
write(9,557) ntreal,ni,nj,nk,nipl,njpl,nkpl,niml,njml,nkml,iter
write(9,556) xc,yc,zc,xs,ys,zs
write(9,556) cxxc,cyyc,azxc,cxxs,cyyys,dzss
REWIND 9
C *** ***** 00040700
GOTO 172 00040800
2020 CONTINUE 00040900
WRITE (6,*) ' RESIDUAL MASS IS LARGER THAN 10.0, PROGRAM STOPS' 00041000
172 CONTINUE 00041100
STOP 00041200
END 00041300
00041400
00041500
00041600
00041700
C ***** 00041800
SUBROUTINE INPUT 00041900
***** 00042000
THIS SUBROUTINE SETS UP REQUIRED VALUES TO BEGIN THE PROGRAM. *00042100
VARIABLES ARE: *00042200
* KRUN = WHEN EQUAL TO ONE,READ FROM THE *00042300
* RESTART DISK, ELSE FROM THE JCL *00042400
* NCHIP = NUMBER OF SOLID PIECES *00042500
* NWRP = NUMBER OF TIME STEPS TO WRITE ON THE *00042600
* PAPER *00042700

```



```

*      NTHCO      =      NUMBER OF THERMOCOUPLES TO PRINT OUT      *00042800
*      TMAX       =      MAXIMUM TIME ALLOWED (REAL)                *00042900
*      TWRITE     =      SECONDS IN REAL TIME TO PRINT THE        *00043000
*                  P,V,T FIELDS ON PAPER                          *00043100
*      TTAPE      =      TIME INTERVAL TO WRITE ON THE TAPE        *00043200
*      DTIME      =      TIME STEP (DIMENSIONLESS)                 *00043300
*      HSZ        =      HEAT SOURCE SIZE, USED TO CALCULATE      *00043400
*                  THE VOLUME OF THE FIRE CELL                    *00043500
*      ICHPB      =      FIRST SOLID NODE IN THETA DIRECTION       *00043600
*      JCHPB      =      FIRST SOLID NODE IN R DIRECTION           *00043700
*      KCHPB      =      FIRST SOLID NODE IN PHI DIRECTION        *00043800
*      NCHPI      =      NUMBER OF NODES IN THETA DIRECTION       *00043900
*      NCHPJ      =      NUMBER OF NODES IN R DIRECTION            *00044000
*      NCHPK      =      NUMBER OF NODES IN PHI DIRECTION         *00044100
*      CX,CY,CZ   =      THERMOCOUPLE POSITIONS IN THETA,R,PHI    *00044200
*****
*00044300
*00044400
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
&      DXXC(93),DYXC(93),DZXC(93),DXXS(93),DYYS(93),DZZS(93)    *00044500
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR          *00044700
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1              *00044800
&      ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP *00044900
COMMON/BL12/ NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER       *00045000
COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PM1,PM2 *00045100
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,UO,H,UGRT,BUOY,*00045200
&      CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO *00045300
COMMON/BL20/SIG11(22,16,32),SIG12(22,16,32),SIG22(22,16,32)  *00045400
&      ,SIG13(22,16,32),SIG23(22,16,32),SIG33(22,16,32)      *00045500
COMMON/BL22/ICHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10), *00045600
&      NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10)          *00045700
COMMON/BL31/ TOD(22,16,32),ROD(22,16,32),POD(22,16,32)       *00045800
&      ,COD(22,16,32),UOD(22,16,32),VOD(22,16,32),WOD(22,16,32) *00045900
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32)             *00046000
&      ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32)     *00046100
COMMON/BL33/ TPD(22,16,32),RPD(22,16,32),PPD(22,16,32)      *00046200
&      ,CPD(22,16,32),UPD(22,16,32),VPD(22,16,32),WPD(22,16,32) *00046300
COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32),                 *00046400
&      SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),            *00046500
&      DV(22,16,32),DW(22,16,32)                               *00046600
COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32), *00046700
&      AS(22,16,32),AF(22,16,32),AB(22,16,32),              *00046800
&      SP(22,16,32),SU(22,16,32),RI(22,16,32)                *00046900
COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579) *00047000
&      ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)    *00047100
COMMON/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12)   *00047200
*00047300
*00047400
C #1. READ IN DATA TO INDICATE EITHER KRUN=0 OR 1             *00047500
      READ(21,*) KRUN,NCHIP,NWRP,NTHCO                          *00047600
*00047700
C #2. READ IN DATA SET 1 - 6 DATA                             *00047800
      READ(21,*) TMAX,TWRITE,TTAPE,DTIME                       *00047900
*00048000
C #3. READ IN DATA FOR HEAT SOURCE                             *00048100
*00048200
      READ(21,*) HSZ(1,1),HSZ(1,2),HSZ(2,1),HSZ(2,2),HSZ(3,1),HSZ(3,2) *00048300
      WRITE(6,20) HSZ(1,1),HSZ(1,2),HSZ(2,1),HSZ(2,2),HSZ(3,1),HSZ(3,2) *00048400
20 FORMAT (/,20X,'HEAT SOURCE LOCATION IS IN THE VOLUME (NON-DIME', *00048500
&      'NSIONAL WITH RESPECT TO RADIUS)',                       *00048600
&      /,5X,'FROM ',F8.4,' TO ',F8.4,' IN X-DIRECTION',        *00048700
&      /,5X,'FROM ',F8.4,' TO ',F8.4,' IN Y-DIRECTION',        *00048800
&      /,5X,'FROM ',F8.4,' TO ',F8.4,' IN Z-DIRECTION',/)      *00048900
*00049000
*00049100
*00049200
C #4. READ IN DECK DATA                                       *00049300
*00049400
      IF (NCHIP.EQ.0) GOTO 16                                    *00049500
      PRINT *                                                    *00049600

```

```

PRINT *, ' THE REGION BOUNDED BY SOLID'
DO 19 N=1,NCHIP
READ (21,*) ICHPB(N),NCHPI(N),JCHPB(N),NCHPJ(N),KCHPB(N),
& NCHPK(N),TCHP(N),CPS(N),CONS(N),WFAN(N)
WRITE (6,10) N, ICHPB(N),NCHPI(N),JCHPB(N),NCHPJ(N),KCHPB(N),
& NCHPK(N),TCHP(N),CPS(N),WFAN(N),CONS(N)
10 FORMAT (2X,'N= ',I2,' ICHPB= ',I2,' NCHPI= ',I2,' JCHPB= ',I2,
& ' NCHPJ= ',I2,' KCHPB= ',I2,' NCHPK= ',I2,' TCHP= ',F8.5,
& ' CPS= ',F8.5,/, ' WFAN = ',F12.5,' CONS= ',F12.5,/)
19 CONTINUE
16 CONTINUE
write(6,*) 'nchip=',nchip

if(nthco.eq.0) goto 119
C #5. INPUT THERMOCOUPLE COORDINATE
C IN TERMS OF X(THETA), Y(RADIUS),Z(PHI)

PRINT *
PRINT *, ' THERMOCOUPLE POSITION IN TERMS OF THETA, R, PHI'
PRINT *
DO 110 I=1,NTHCO
READ (21,*) CX(I),CY(I),CZ(I)
WRITE (6,*) I, CX(I),CY(I),CZ(I)
110 CONTINUE
119 continue
RETURN
END

C
C *****
SUBROUTINE INIT
C *****
* THIS SUBROUTINE INITIALIZES THE FIELD AND CONSTANTS WITH RESPECT
* TO INITIAL START OR RESTARTING CAPABILITY.
* VARIABLES ARE :
* TIME = DIMENSIONLESS TIME
* UO = CHARACTERISTIC VELOCITY (1 FT/SEC)
* H = CHARACTERISTIC LENGTH (RADIUS(9.6FT))
* TR = TEMP IN DEGREES KELVIN
* TA = TEMP IN DEGREES RANKINE
* VISO = REFERENCE VISCOSITY (NONDIM)
* VISL = MINIMUM VISCOSITY (NONDIM)
* VISMAX = MAXIMUM VISCOSITY (NONDIM)
* HR = RADIUS IN CM
* CONDO = REFERENCE CONDUCTIVITY
* CO = INITIAL SMOKE CONCENTRATION
* NJRA = POINT OF RADIATION IN J DIRECTION
* LOCATED ON THE INNER SOLID BOUNDARY
* HCONV = HEAT TRANSFER COEFFICIENT
* HCOEF = DIMENSIONLESS HEAT TRANSFER COEF
* CONST1 = USED TO NONDIMENSIONALIZE PRESSURE
* RHOO = REFERENCE DENSITY
* GC = GRAVITY CONSTANT
* BUOY = BUOYANCY FORCE CONSTANT
* UGRT = PERFECT GAS LAW NONDIMENSIONAL CONSTANT
* CPO = REFERENCE SPECIFIC HEAT
* NWRITE/ = NONDIMENSIONAL FORMS OF TWRITE AND
* NTAPE = TTAPE
* MATRICES OF THE FORM
* _OD = DIMENSIONLESS PARAMETER AT OLD TIME
* = DIMENSIONLESS PARAMETER
* _PD = UPDATED DIMENSIONLESS PARAMETER
* WHERE THE PARAMETERS ARE
* U,V,W = VELOCITY IN THETA, R, PHI DIRECTION
* T,P,C = TEMP, PRESSURE, AND SMOKE CONCENTRATION

```

```

*      DU, DV, DZ      =      USED IN PRESSURE CORRECTION SUBROUTINE *00056300
*      PP              =      CORRECTED PRESSURE (P') *00056400
*      SU              =      SOURCE TERM *00056500
*      SP              =      TERM AT P NODAL POINT FOR BOUNDARY *00056600
*                      =      CONDITIONS *00056700
*      AP              =      COEFFICIENT AT NODAL POINT *00056800
*      AE, AW, AN      =      COEFFICIENTS AT PTS EAST, WEST, NORTH, *00056900
*      AS, AF, AB      =      SOUTH, FRONT, AND BACK *00057000
*      SMP              =      RESIDUAL MASS SUMMATION OF NODAL POINT *00057100
*      SMPP            =      LENGTH SCALE FOR TURBULENCE *00057200
*      CPM              =      MEAN SPECIFIC HEAT *00057300
*      VIS              =      VISCOSITY *00057400
*      COND             =      CONDUCTIVITY MATRIX *00057500
*      NHSZ             =      WHEN THIS VALUE EQUALS ZERO, THERE IS *00057600
*                      =      NO HEAT SOURCE LOCATED AT THE NODE *00057700
*      NOD              =      IF EQUAL TO ZERO, LIQUID *00057800
*                      =      IF EQUAL TO ONE, SOLID *00057900
*      _3, _E          =      BEGINNING AND ENDING NODAL POINT FOR *00058000
*                      =      THE SOLID IN I, J, K *00058100
*      REQ              =      DENSITY AT EQUILIBRIUM *00058200
*      NIP1             =      NODAL POINT IN I PLUS 1 (OTHERS SIMILAR) *00058300
*      XC, YC, ZC      =      THETA, R, PHI LOCATION OF NODAL POINT OF *00058400
*                      =      A CENTER CELL *00058500
*      DXXC, DYXC      =      LENGTH AROUND THE CENTER CELL *00058600
*      DZZC            =      *00058700
*      XS, YS, ZS      =      THETA, R, PHI LOCATION OF NODAL POINT OF *00058800
*                      =      A STAGGERED CELL *00058900
*      DXXS, DYXS      =      LENGTH AROUND THE STAGGERED CELL *00059000
*      DZZS            =      *00059100
*      CX, CY, CZ      =      LOCATION OF THERMOCOUPLE IN THETA, R, PHI *00059200
*****00059300
COMMON/R4/XC(93), YC(93), ZC(93), XS(93), YS(93), ZS(93),
&      DXXC(93), DYXC(93), DZZC(93), DXXS(93), DYXS(93), DZZS(93) 00059400
COMMON/BL1/DX, DY, DZ, VOL, DTIME, VOLDT, THOT, TCOOL, PI, Q, QR 00059600
COMMON/BL7/NI, NIP1, NIM1, NJ, NJP1, NJM1, NK, NKP1, NKMI 00059700
&      NIP2, NJP2, NKP2, NA, NAP1, NAM1, NB, NBPI, NBM1, KRUN, NCHIP, NJRA, NWRP 00059800
COMMON/BL12/ NWRITE, NTAPE, NTMAX0, NTREAL, TIME, SORSUM, ITER 00059900
COMMON/BL14/HCOEF, TINF, CNT, ABTURB, BTURB, VISL, VISMAX, QCORRT, PM1, PM200060000
COMMON/BL16/ CONST1, CONST2, CONST3, CONST4, CONST6, NT, UC, H, UGRT, BUOY, 00060100
&      CPO, PRT, CONDO, VISC, RHOO, HR, TR, TA, DTEMP, TWRITE, TTAPE, TMAX, GC, RAIRO0060200
COMMON/BL20/SIG11(22,16,32), SIG12(22,16,32), SIG22(22,16,32) 00060300
&      , SIG13(22,16,32), SIG23(22,16,32), SIG33(22,16,32) 00060400
COMMON/BL22/ ICHPB(10), NCHPI(10), JCHPB(10), NCHPJ(10), KCHPB(10), 00060500
&      NCHPK(10), TCHP(10), CPS(10), CONS(10), WFAN(10) 00060600
COMMON/BL31/ TOD(22,16,32), ROD(22,16,32), POD(22,16,32) 00060700
&      , COD(22,16,32), UOD(22,16,32), VOD(22,16,32), WOD(22,16,32) 00060800
COMMON/BL32/ T(22,16,32), R(22,16,32), P(22,16,32) 00060900
&      , C(22,16,32), U(22,16,32), V(22,16,32), W(22,16,32) 00061000
COMMON/BL33/ TPD(22,16,32), RPD(22,16,32), PPD(22,16,32) 00061100
&      , CPD(22,16,32), UPD(22,16,32), VPD(22,16,32), WPD(22,16,32) 00061200
COMMON/BL34/ HEIGHT(22,16,32), REQ(22,16,32), 00061300
&      SMP(22,16,32), SMPP(22,16,32), PP(22,16,32), 00061400
&      DU(22,16,32), DV(22,16,32), DW(22,16,32) 00061500
COMMON/BL36/AP(22,16,32), AE(22,16,32), AW(22,16,32), AN(22,16,32), 00061600
&      AS(22,16,32), AF(22,16,32), AB(22,16,32), 00061700
&      SP(22,16,32), SU(22,16,32), RI(22,16,32) 00061800
COMMON/BL37/ VIS(22,16,32), COND(22,16,32), NOD(22,16,32), RWALL(579) 00061900
&      , CPM(22,16,32), HSZ(3,2), NHSZ(22,16,32), RESORM(93) 00062000
COMMON/BL38/NTHCO, CX(12), CY(12), CZ(12), NTH(12,3), TCOUP(12) 00062100
COMMON/BL39/ALEW, PCURVE, CONSRA, PCURM1, PSOUTH, QCORR, PEROR 00062200
DATA GRAV/32.177 00062300
00062400
C *** INTRODUCE GIVEN PARAMETERS 00062500
TIME=C. 00062600
TR=TA/1.8 00062700
H=9.6 00062800
VISO=VISC/UC/H 00062900
00063000

```

```

VISL=VISO                                00063100
VISMAX=400.*VISL                          00063200
HR=H*30.48                                00063300
CONDC=VISO/PRT                             00063400
PI=4.*ATAN(1.)                             00063500
ALEW = 1.0                                 00063600
NJRA=15                                    00063700
                                           00063800
C THE HEAT TRANSFER COEFFICIENT IS IN BTU/HR/FT**2/F 00063900
HCONV=15.0                                 00064000
HCOEF=HCONV/(3600.*CPO*RHO0*U0)           00064100
CO = 0.0                                    00064200
                                           00064300
                                           00064400
CONST1=RHO0*U0*U0/(GC*14.696*144.)        00064500
CONST3=1.8/TA                              00064600
CONST4=H*30.48                             00064700
CONST6=U0*30.48                            00064800
NTMAXC=0                                    00064900
                                           00065000
BUOY=GRAV*H/(U0*U0)                        00065100
UGRT=UC*U0/(GC*RAIR*TA)                   00065200
TCOOL=1.0                                   00065300
CONSRA=TA*TA*TA/(RHO0*CPO*U0*3600.)*1.714E-9 00065400
                                           00065500
WRITE(6,200) TR,CONDO,VISO,CPO,HR,DTIME,HCONV 00065600
200 FORMAT(5X,'THE REFERENCE TEMPERATURE AND THERMAL PROPERTIES',/, 00065700
& /,5X,'T = ',F10.4,'K', CONDO = ',E12.6, 00065800
& /,5X,'VISO = ',E12.6,' CPO = ',E12.6, 00065900
& /,5X,'RADIUS = ',E12.6,' CM', 00066000
& /,5X,'DTIME = ',E12.6, 00066100
& /,5X,'HCONV = ',E12.6,/) 00066200
                                           00066300
NWRITE=jint(TWRITE*U0/DTIME/H)
00066400
NTAPE=jint(TTAPE*U0/DTIME/H)
00066500
C *** PRINT OUT INPUT INFORMATION 00066600
                                           00066700
WRITE(6,61) (STAR,I=1,90),KRUN,TMAX,TWRITE,TTAPE,NWRP 00066800
61 FORMAT(///,90A1,/,5X,'KRUN = ',I2,/,5X, 00066900
& 'TMAX = ',F8.3,' SECONDS',/5X,'TWRITE = ',F8.3, 00067000
& ' SECONDS',/,5X,'TTAPE = ',F8.3,' SECONDS', 00067100
& /,5X,' NUMBER INTERVALS OF WRITING ON PAPER ', I5,/) 00067200
C *** INITIALIZE VARIABLE FIELD 00067300
                                           00067400
DO 220 J=1,NJPI 00067500
DO 220 I=1,NIP1 00067600
DO 220 K=1,NKPI 00067700
ROD(I,J,K)=1. 00067800
R(I,J,K)=1. 00067900
RPD(I,J,K)=1. 00068000
UOD(I,J,K)=0. 00068100
U(I,J,K)=0. 00068200
UPD(I,J,K)=0. 00068300
VOD(I,J,K)=0. 00068400
V(I,J,K)=0. 00068500
VPD(I,J,K)=0. 00068600
W(I,J,K)=0. 00068700
WPD(I,J,K)=0. 00068800
XOD(I,J,K)=0. 00068900
X(I,J,K)=0. 00069000
XPD(I,J,K)=0. 00069100
Y(I,J,K)=0. 00069200
YPD(I,J,K)=0. 00069300
Z(I,J,K)=0. 00069400
ZPD(I,J,K)=0. 00069500
DW(I,J,K)=0. 00069600

```

SU(I,J,K)=0.	00069700
SP(I,J,K)=0.	00069800
PP(I,J,K)=0.	00069900
AP(I,J,K)=0.	00070000
AW(I,J,K)=0.	00070100
AE(I,J,K)=0.	00070200
AN(I,J,K)=0.	00070300
AS(I,J,K)=0.	00070400
AF(I,J,K)=0.	00070500
AB(I,J,K)=0.	00070600
SMP(I,J,K)=0.	00070700
SMPP(I,J,K)=0.	00070800
VIS(I,J,K)=VISL	00070900
COND(I,J,K)=CONDO	00071000
CPM(I,J,K)=1.0E0	00071100
TOD(I,J,K)=1.0E0	00071200
T(I,J,K)=TOD(I,J,K)	00071300
TPD(I,J,K)=TOD(I,J,K)	00071400
COD(I,J,K)=CO	00071500
C(I,J,K)=COD(I,J,K)	00071600
CPD(I,J,K)=COD(I,J,K)	00071700
NHSZ(I,J,K)=0	00071800
NOD(I,J,K)=0	00071900
220 CONTINUE	00072000
	00072100
	00072200
C *** DETERMINE THE POSITION OF HEAT SOURCE	00072300
	00072400
DO 300 I=2,NI	00072500
DO 300 J=2,NJ	00072600
	00072700
	00072800
C CHANGE TO RECTANGULAR COORDINATES	00072900
XX=YC(J)*COS(XC(I))	00073000
YY=YC(J)*SIN(XC(I))	00073100
	00073200
C CHECK TO SEE IF IN HS CONTROL VOLUME, IF SO SET NHSZ=1	00073300
IF (XX.LT.HSZ(1,1).OR.XX.GT.HSZ(1,2)) GOTO 310	00073400
IF (YY.LT.HSZ(2,1).OR.YY.GT.HSZ(2,2)) GOTO 310	00073500
NHSZ(I,J,16)=1	00073600
NHSZ(I,J,17)=1	00073700
315 FORMAT (2X,10(4X,I4,2X,I4))	00073800
GOTO 300	00073900
310 CONTINUE	00074000
300 CONTINUE	00074100
	00074200
	00074300
C *** DEFINE THERMAL PROPERTIES OF DECK AND SOLID	00074400
	00074500
IF (NCHIP.EQ.0) GOTO 410	00074600
DO 402 N=1,NCHIP	00074700
IB=ICHPB(N)	00074800
IE=IB+NCHPI(N)-1	00074900
JB=JCHPB(N)	00075000
JE=JB+NCHPJ(N)-1	00075100
KB=KCHPB(N)	00075200
KE=KB+NCHPK(N)-1	00075300
DO 405 I=IB,IE-1	00075400
DO 405 J=JB,JE-1	00075500
DO 405 K=KB,KE-1	00075600
COND(I,J,K)=CONDC*CONS(N)	00075700
CPM(I,J,K)=CPC*CPS(N)	00075800
NOD(I,J,K)=1	00075900
405 CONTINUE	00076000
402 CONTINUE	00076100
410 CONTINUE	00076200
	00076300
	00076400

C *** FOR CONTINUING RUN, READ DATA FROM TAPE OR DISK	00076500
IF(KRUN .EQ. 1) GO TO 9997	00076600
GO TO 15	00076700
9997 DO 222 K=1,NKPL	00076800
DO 222 I=1,NIP1	00076900
DO 222 J=1,NJPL	
READ(9,555) T(I,J,K),U(I,J,K),V(I,J,K),W(I,J,K)	
READ(9,555) P(I,J,K),CPM(I,J,K),COND(I,J,K),VIS(I,J,K)	
222 CONTINUE	
READ(9,556) TIME,QR,QCORRT,PM1,PM2,XXXXX	
READ(9,556) XXN,XXTA,XXU0,XXCOND0,XXVIS0,XXRHO0	
READ(9,557) NREAL,NI,NJ,NK,NIP1,NJPL,NKPL,NIM1,NJML,NKML,ITER	
READ(9,556) XC,YC,ZC,XS,YS,ZS	
READ(9,556) DXXC,DYYC,DZZC,DXXS,DYYS,DZZS	
555 FORMAT(4(3X,E12.4))	
556 FORMAT(6(1X,E10.3))	
557 FORMAT(11I4)	
REWIND 9	
WRITE(6,*)NTMAX0	00077800
15 CONTINUE	00077900
	00078000
	00078100
C *** DEFINE HEIGHT OF NODE POINTS AND COMPUTE HYDROSTATIC	00078200
EQUILIBRIUM DENSITY REQ(I,J,K)	00078300
	00078400
	00078500
DO 13 K=1,NKPL	00078600
DO 13 I=1,NIP1	00078700
DO 13 J=1,NJPL	00078800
DHY=YC(J)*SIN(XC(I))*SIN(ZC(K))	00078900
HEIGHT(I,J,K)=DHY	00079000
13 CONTINUE	00079100
C	00079200
	00079300
DO 229 J=1,NJPL	00079400
DO 229 I=1,NIP1	00079500
DO 229 K=1,NKPL	00079600
AAAA=-BUOY*UGRT*HEIGHT(I,J,K)	00079700
REQ(I,J,K)=EXP(AAAA)	00079800
IF(KRUN .NE. 0) GO TO 229	00079900
RPD(I,J,K)=REQ(I,J,K)/TPD(I,J,K)	00080000
ROD(I,J,K)=RPD(I,J,K)	00080100
R(I,J,K)=RPD(I,J,K)	00080200
229 CONTINUE	00080300
	00080400
C *** INITIALIZE U,V,T,R,P FIELD	00080500
	00080600
DO 210 K=1,NKPL	00080700
DO 210 J=1,NJPL	00080800
DO 210 I=1,NIP1	00080900
T(I,J,K)=TOD(I,J,K)	00081000
C(I,J,K)=COD(I,J,K)	00081100
R(I,J,K)=ROD(I,J,K)	00081200
U(I,J,K)=UOD(I,J,K)	00081300
V(I,J,K)=VOD(I,J,K)	00081400
W(I,J,K)=WOD(I,J,K)	00081500
P(I,J,K)=POD(I,J,K)	00081600
210 CONTINUE	00081700
	00081800
C *** FOLLOWING IS FOR DETERMINING THE THERMOCOUPLE POSITIONS	00081900
	00082000
DO 5000 N=1,NTHCO	00082100
DO 5001 I=1,NIP1	00082200
IF(XC(I).LT.CX(N).AND.XC(I-1).GE.CX(N)) GOTO 5002	00082300
5001 CONTINUE	00082400
5002 II=I	00082500

```

DO 5003 J=1,NJP1                                00082600
IF (YC(J).LT.CY(N).AND.YC(J+1).GE.CY(N)) GOTO 5004 00082700
5003 CONTINUE                                     00082800
5004 JJ=J                                         00082900
                                                    00083000
DO 5005 K=1,NKP1                                00083100
IF (ZC(K).LT.CZ(N).AND.ZC(K+1).GE.CZ(N)) GOTO 5006 00083200
5005 CONTINUE                                     00083300
5006 KK=K                                         00083400
NTH(N,1)=II                                       00083500
NTH(N,2)=JJ                                       00083600
NTH(N,3)=KK                                       00083700
5000 CONTINUE                                     00083800
                                                    00083900
RETURN                                            00084000
END                                                00084100
                                                    00084200
                                                    00084300
C _____ 00084400
C *** 00084500
SUBROUTINE CALVIS                                00084600
                                                    00084700
C *** 00084800
* THIS SUBROUTINE CALCULATES THE TURBULENT VISCOSITY AND UPDATES* 00084900
* THE VISCOSITY MATRIX * 00085000
***** 00085100
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93), 00085300
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93) 00085400
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1 00085500
& NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP 00085600
COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PML,PM2 00085700
COMMON/BL16/CONST1,CONST2,CONST3,CONST4,CONST6,NT,UC,H,UGRT,BOUY, 00085800
& CPO,PRT,CONDO,VISO,RH00,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR 00085900
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32) 00086000
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32) 00086100
COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32), 00086200
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32), 00086300
& DU(22,16,32),DV(22,16,32),DW(22,16,32) 00086400
COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32), 00086500
& AS(22,16,32),AF(22,16,32),AB(22,16,32), 00086600
& SP(22,16,32),SU(22,16,32),RI(22,16,32) 00086700
COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579) 00086800
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93) 00086900
                                                    00087000
                                                    00087100
C *** CALCULATE LOCAL SHEAR AND VISCOSITY VIS(I,J,K) 00087200
C 00087300
C *** SPECIFY LOCAL TURBULENT LENGTH SCALES SMPP(I,J,K) 00087400
                                                    00087500
DO 611 K=2,NK 00087600
KP2=K+2 00087700
KP1=K-1 00087800
KM1=K-1 00087900
KM2=K-2 00088000
DO 611 J=2,NJ 00088100
JP2=J+2 00088200
JP1=J+1 00088300
JM1=J-1 00088400
JM2=J-2 00088500
DO 611 I=2,NI 00088600
IP2=I-2 00088700
IP1=I-1 00088800
IM1=I-1 00088900
IM2=I-2 00089000
IF (I.EQ.2) IM2=NIM1 00089100
IF (I.EQ.NI) IP2=3 00089200
IF (NOD(I,J,K).EQ.1) GOTO 611 00089300

```

C	CENTRAL LENGTH OF THE SCALE CONTROL VOLUME	00089400
	DXP1=XL(IP1,J,K,0,0)	00089500
	DXI =XL(I ,J,K,0,0)	00089600
	DXM1=XL(IM1,J,K,0,0)	00089700
	DYP1=YL(I,JP1,K,0,0)	00089800
	DYJ =YL(I,J ,K,0,0)	00089900
	DYM1=YL(I,JM1,K,0,0)	00090000
	DZP1=ZL(I,J,KP1,0,0)	00090100
	DZK =ZL(I,J,K ,0,0)	00090200
	DZM1=ZL(I,J,KM1,0,0)	00090300
		00090400
		00090500
		00090600
		00090700
		00090800
CC	IF (J.EQ.2) DYS=DYS/2.	00090900
CC	IF (K.EQ.2) DZB=DZB/2.	00091000
	IF (J.NE.NJ) GOTO 101	00091100
	JP2=JP1	00091200
	DYN=DYN/2.	00091300
101	IF (K.NE.NK) GOTO 102	00091400
	KP2=KP1	00091500
	DZF=DZF/2.	00091600
102	CONTINUE	00091700
		00091800
C ***	CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR T	00091900
		00092000
	DXE =XL(IP1,J,K,0,1)	00092100
	DXW =XL(I ,J,K,0,1)	00092200
		00092300
	DYN =YL(I,JP1,K,0,2)	00092400
	DYS =YL(I,J ,K,0,2)	00092500
		00092600
	DZF =ZL(I,J,KP1,0,3)	00092700
	DZB =ZL(I,J,K ,0,3)	00092800
		00092900
C ***	CACULATE DV/DX,D2V/DX2,DU/DX,D2U/DX2,DW/DX AND D2W/DX2	00093000
		00093100
		00093200
	DUDX=(U(IP1,J,K)-U(I,J,K))/DXI	00093300
	DUDXW=0.5*(U(IP1,J,K)-U(IM1,J,K))/DXW	00093400
	DUDXE=0.5*(U(IP2,J,K)-U(I ,J,K))/DXE	00093500
	D2UDX2=(DUDXE-DUDXW)/DXI	00093600
		00093700
		00093800
	DVDXW=0.5*(V(I,JP1,K)-V(I,J,K)-V(IM1,JP1,K)-V(IM1,J,K))/DXW	00093900
	DVDXE=0.5*(V(IP1,JP1,K)-V(IP1,J,K)-V(I,JP1,K)-V(I,J,K))/DXE	00094000
	DVDX=0.5*(DVDXE+DVDXW)	00094100
	D2VDX2= (DVDXE-DVDXW)/DXI	00094200
		00094300
		00094400
	DWDXW=0.5*(W(I,J,KP1)-W(I,J,K)-W(IM1,J,KP1)-W(IM1,J,K))/DXW	00094500
	DWDXE=0.5*(W(IP1,J,KP1)+W(IP1,J,K)-W(I,J,KP1)-W(I,J,K))/DXE	00094600
	DWDX=0.5*(DWDXE-DWDXW)	00094700
	D2WDX2= (DWDXE-DWDXW)/DXI	00094800
		00094900
		00095000
602	CONTINUE	00095100
		00095200
C ***	CALCULATE DU/DY,D2U/DY2,DV/DY,D2V/DY2,DW/DY AND D2W/DY2	00095300
		00095400
		00095500
	DVDY=(V(I,JP1,K)-V(I,J,K))/DYJ	00095600
	DVDYS=0.5*(V(I,JP1,K)-V(I,JM1,K))/DYS	00095700
	DVDYN=0.5*(V(I,JP2,K)-V(I,J ,K))/DYN	00095800
	D2VDY2=(DVDYN-DVDYS)/DYJ	00095900
		00096000
		00096100


```

DUDYS=C.5*(U(IP1,J,K)+U(I,J,K)-U(IP1,JM1,K)-U(I,JM1,K))/DYS
DUDYN=C.5*(U(IP1,JP1,K)+U(I,JP1,K)-U(IP1,J,K)-U(I,J,K))/DYN
DUDY=C.5*(DUDYN+DUDYS)
D2UDY2=(DUDYN-DUDYS)/DYJ
00096200
00096300
00096400
00096500
00096600
00096700
DWDYS=C.5*(W(I,J,KP1)+W(I,J,K)-W(I,JM1,KP1)-W(I,JM1,K))/DYS
DWDYN=C.5*(W(I,JP1,KP1)+W(I,JP1,K)-W(I,J,KP1)-W(I,J,K))/DYN
DWDY=C.5*(DWDYN+DWDYS)
D2WDY2=(DWDYN-DWDYS)/DYJ
00096800
00096900
00097000
00097100
00097200
00097300
00097400
00097500
606 CONTINUE
00097600
00097700
C *** CALCULATE DU/DZ, D2U/DZ2, DV/DZ, D2V/DZ2, DW/DZ AND D2W/DZ2
00097800
00097900
DWDZ=(W(I,J,KP1)-W(I,J,K))/DZK
DWDZF=C.5*(W(I,J,KP2)-W(I,J,K))/DZF
DWDZB=C.5*(W(I,J,KP1)-W(I,J,KM1))/DZB
D2WDZ2=(DWDZF-DWDZB)/DZK
00098000
00098100
00098200
00098300
DVDZB=C.5*(V(I,JP1,K)+V(I,J,K)-V(I,JP1,KM1)-V(I,J,KM1))/DZB
DVDZF=C.5*(V(I,JP1,KP1)+V(I,J,KP1)-V(I,JP1,K)-V(I,J,K))/DZF
DVDZ=C.5*(DVDZF+DVDZB)
D2VDZ2=(DVDZF-DVDZB)/DZK
00098400
00098500
00098600
00098700
00098800
00098900
DUDZB=C.5*(U(IP1,J,K)+U(I,J,K)-U(IP1,J,KM1)-U(I,J,KM1))/DZB
DUDZF=C.5*(U(IP1,J,KP1)+U(I,J,KP1)-U(IP1,J,K)-U(I,J,K))/DZF
DUDZ=C.5*(DUDZF+DUDZB)
D2UDZ2=(DUDZF-DUDZB)/DZK
00099000
00099100
00099200
00099300
00099400
DRDX=((R(IP1,J,K)-REQ(IP1,J,K))-R(IM1,J,K)-REQ(IM1,J,K))/
& (DXE-DXW)
00099500
00099600
DRDY=((R(I,JP1,K)-REQ(I,JP1,K))-R(I,JM1,K)-REQ(I,JM1,K))/
& (DYN-DYS)
00099700
00099800
DRDZ=((R(I,J,KP1)-REQ(I,J,KP1))-R(I,J,KM1)-REQ(I,J,KM1))/
& (DZF-DZB)
00099900
01000000
DRDGA=SIN(ZC(K))*SIN(XC(I))*DRDY+COS(XC(I))*DRDX
& -COS(ZC(K))*DRDZ
01000100
01000200
01000300
C *** CALCULATE RICHARDSON NUMBER
01000400
01000500
STRAIN=DUDY**2+DVDX**2+DWDX**2+DVDZ**2+DWDY**2+DUDZ**2
01000600
DDO2=SQRT(DUDY*DUDY+DUDX*DUDX+DUDZ*DUDZ+DVDY*DVDY+DVDX*DVDX+
01000700
& DVDZ*DVDZ+DWDX*DWDX+DWDY*DWDY+DWDZ*DWDZ)
01000800
IF(DDO2.EQ.0.)GO TO 600
01000900
01010000
01010100
C *** CALCULATE TURBULENT LENGTH SCALE SMPP(I,J)
01010200
01010300
SMPP123=SQRT(((U(IP1,J,K)+U(I,J,K))*0.5)**2+((V(I,JP1,K)-V(I,J,K))*
& 0.5)**2+((W(I,J,KP1)+W(I,J,K))*0.5)**2)/DDO2
01010400
SMPP12=DDO2/SQRT(D2UDX2*D2UDX2+D2UDY2*D2UDY2
01010500
& -D2UDZ2*D2UDZ2+D2VDX2*D2VDX2+D2VDY2*D2VDY2+D2VDZ2*D2VDZ2+
01010600
& D2WDZ2*D2WDZ2+D2WDX2*D2WDX2+D2WDY2*D2WDY2)
01010700
SMPP(I,J,K)=CNT*(SMPP123+SMPP12)*.5
01010800
RI(I,J,K)=-BUOY*DRDGA/(R(I,J,K)*STRAIN)
01010900
ABRIPR=ABTURB+RI(I,J,K)/PRT
01012000
IF(ABRIPR.LT.0.)GO TO 600
01020100
IF(ABRIPR.EQ.0.)GO TO 613
01020200
GO TO 610
01020300
600 VIS(I,J,K)=VISL
01020400
GO TO 611
01020500
613 VIS(I,J,K)=VISMAL
01020600
GO TO 611
01020700
610 VIS(I,J,K)=VISL+R(I,J,K)*SMPP(I,J,K)*SMPP(I,J,K)*SQRT(STRAIN)/
01020800
& (BTURB*ABRIPR)
01020900

```

```

IF (VIS(I,J,K) .GT. VISMAL) VIS(I,J,K)=VISMAL
611 CONTINUE

DO 110 I=1,NIP1
DO 110 J=1,NJP1
VIS(I,J,NKP1)=VIS(I,J,NK)
VIS(I,J,1)=VIS(I,J,2)
110 CONTINUE

DO 120 J=1,NJP1
DO 120 K=1,NKP1
VIS(NIP1,J,K)=VIS(2,J,K)
VIS(1,J,K)=VIS(NI,J,K)
120 CONTINUE

DO 130 K=1,NKP1
DO 130 I=1,NIP1
VIS(I,NJP1,K)=VIS(I,NJ,K)
VIS(I,2,K)=VIS(I,3,K)
VIS(I,1,K)=VIS(I,2,K)
130 CONTINUE

DO 135 K=1,16
KK=NKP1-K
DO 135 I=1,NIP1
DO 135 J=1,NJP1
VIS(I,J,KK)=VIS(I,J,K)
135 CONTINUE

DO 140 I=1,NIP1
DO 140 J=1,NJP1
DO 140 K=1,NKP1
IF (MOD(I,J,K).EQ.1) GOTO 140
COND(I,J,K)=VIS(I,J,K)/PRT
140 CONTINUE

RETURN
END

```

	00103000
	00103100
	00103200
	00103300
	00103400
	00103500
	00103600
	00103700
	00103800
	00103900
	00104000
	00104100
	00104200
	00104300
	00104400
	00104500
	00104600
	00104700
	00104800
	00104810
	00104900
	00105000
	00105100
	00105110
	00105120
	00105130
	00105140
	00105150
	00105160
	00105170
	00105200
	00105300
	00105400
	00105500
	00105600
	00105700
	00105800
	00105900
	00106000
	00106100
	00106200
	00106300
	00106400
	00106500
	00106600
	00106700
	00106800
	00106900
	00107000
	00107100
	00107200
	00107300
	00107400
	00107500
	00107600
	00107700
	00107800
	00107900
	00108000
	00108100
	00108200
	00108300
	00108400
	00108500
	00108600
	00108700
	00108800
	00108900

C
C
C

SUBROUTINE CALT

```

COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZXC(93),DXXS(93),DYYS(93),DZZS(93)
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,P1,Q,QR
COMMON/BL7/NT,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP
COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAL,QCORRT,PM1,PM200107400
COMMON/BL16/CONST1,CONST2,CONST3,CONST4,CONST6,NT,UC,H,UGRT,BUOY,00107500
& CPC,PRT,CONDC,VISO,RH00,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIROO107600
COMMON/BL22/ICHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10),
& NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10)
COMMON/BL31/TOD(22,16,32),ROD(22,16,32),POD(22,16,32)
& ,COD(22,16,32),UOD(22,16,32),VOD(22,16,32),WOD(22,16,32)
COMMON/BL32/T(22,16,32),R(22,16,32),P(22,16,32)
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32)
COMMON/BL33/TPD(22,16,32),RPD(22,16,32),PPD(22,16,32)
& ,CPD(22,16,32),UPD(22,16,32),VPD(22,16,32),WPD(22,16,32)
COMMON/BL34/HEIGHT(22,16,32),REQ(22,16,32),
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),
& DU(22,16,32),DV(22,16,32),DW(22,16,32)
COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32),
& AS(22,16,32),AF(22,16,32),AB(22,16,32),

```

&	SP (22, 16, 32), SU (22, 16, 32), RI (22, 16, 32)	00109000
	COMMON/BL37/VIS (22, 16, 32), COND (22, 16, 32), NOD (22, 16, 32), RWALL (579)	00109100
&	, CPM (22, 16, 32), HSZ (3, 2), NHSZ (22, 16, 32), RESORM (93)	00109200
		00109300
C ***	CALCULATE COEFFICIENTS	00109400
	DO 100 K=2, NK	00109500
	KP2=K+2	00109600
	KP1=K+1	00109700
	KM1=K-1	00109800
	KM2=K-2	00109900
	DO 100 J=2, NJ	00110000
	JP2=J+2	00110100
	JP1=J+1	00110200
	JM1=J-1	00110300
	JM2=J-2	00110400
	DO 100 I=2, NI	00110500
	IP2=I+2	00110600
	IP1=I+1	00110700
	IM1=I-1	00110800
	IM2=I-2	00110900
	IF (I.EQ.2) IM2=NIM1	00111000
	IF (I.EQ.NI) IP2=3	00111100
		00111200
		00111300
C	CENTRAL LENGTH OF THE TEMPERTURE CONTROL VOLUME	00111400
	DXP1=XL (IP1, J, K, 0, 0)	00111500
	DXI =XL (I, J, K, 0, 0)	00111600
	DXM1=XL (IM1, J, K, 0, 0)	00111700
		00111800
	DYP1=YL (I, JP1, K, 0, 0)	00111900
	DYJ =YL (I, J, K, 0, 0)	00112000
	DYM1=YL (I, JM1, K, 0, 0)	00112100
		00112200
	DZP1=ZL (I, J, KP1, 0, 0)	00112300
	DZK =ZL (I, J, K, 0, 0)	00112400
	DZM1=ZL (I, J, KM1, 0, 0)	00112500
		00112600
		00112700
C ***	SURFACE LENGTH OF THE CONTROL VOLUME	00112800
	DXN=XL (I, JP1, K, 0, 2)	00112900
	DXS=XL (I, J, K, 0, 2)	00113000
	DXF=XL (I, J, KP1, 0, 3)	00113100
	DXB=XL (I, J, K, 0, 3)	00113200
		00113300
	DYF=YL (I, J, KP1, 0, 3)	00113400
	DYB=YL (I, J, K, 0, 3)	00113500
	DYE=YL (IP1, J, K, 0, 1)	00113600
	DYW=YL (I, J, K, 0, 1)	00113700
		00113800
		00113900
	DZE=ZL (IP1, J, K, 0, 1)	00114000
	DZW=ZL (I, J, K, 0, 1)	00114100
	DZN=ZL (I, JP1, K, 0, 2)	00114200
	DZS=ZL (I, J, K, 0, 2)	00114300
		00114400
C ***	CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR T	00114500
	DXEE=XL (IP2, J, K, 0, 1)	00114600
	DXE =XL (IP1, J, K, 0, 1)	00114700
	DXW =XL (I, J, K, 0, 1)	00114800
	DXWW=XL (IM1, J, K, 0, 1)	00114900
		00115000
		00115100
	DYNN=YL (I, JP2, K, 0, 2)	00115200
	DYN =YL (I, JP1, K, 0, 2)	00115300
	DYS =YL (I, J, K, 0, 2)	00115400
	DYSS=YL (I, JM1, K, 0, 2)	00115500
		00115600
	DZFF=ZL (I, J, KP2, 0, 3)	00115700

DZF = ZL(I, J, KP1, 0, 3)
 DZB = ZL(I, J, K, 0, 3)
 DZBB = ZL(I, J, KM1, 0, 3)

00115800
 00115900
 00116000
 00116100
 00116200
 00116300
 00116400
 00116500
 00116600
 00116700
 00116800
 00116900
 00117000
 00117100
 00117200
 00117300
 00117400
 00117500
 00117600
 00117700
 00117800
 00117900
 00118000
 00118100
 00118200
 00118300
 00118400
 00118500
 00118600
 00118700
 00118800
 00118900
 00119000
 00119100
 00119200
 00119300
 00119400
 00119500
 00119600
 00119700
 00119800
 00119900
 00120000
 00120100
 00120200
 00120300
 00120400
 00120500
 00120600
 00120700
 00120800
 00120900
 00123110
 00123120
 00123130
 00123140
 00123150
 00123160
 00123170
 00123180
 00123190
 00123191
 00123192
 00123193
 00123194
 00123195
 00123196
 00123197

C *** DEFINE THE AREA OF THE CONTROL VOLUME

DXYF = DXF * DYF
 DXYB = DXB * DYB
 DYZE = DYE * DZE
 DYZW = DYW * DZW
 DZKN = DZN * DXN
 DZXS = DZS * DXS

VOL = DXI * DYJ * DZK
 VOLDT = VOL / DTIME

ZXOYN = DZKN / DYN
 ZXOYS = DZXS / DYS
 XYOZF = DXYF / DZF
 XYOZB = DXYB / DZB
 YZOXE = DYZE / DXE
 YZOXW = DYZW / DXW

GN = (R(I, J, K) * DYP1 + R(I, JP1, K) * DYJ) / (DYP1 + DYJ)
 GS = (R(I, J, K) * DYM1 + R(I, JM1, K) * DYJ) / (DYM1 + DYJ)
 GE = (R(I, J, K) * DXP1 + R(IP1, J, K) * DXI) / (DXP1 + DXI)
 GW = (R(I, J, K) * DXM1 + R(IM1, J, K) * DXI) / (DXM1 + DXI)
 GF = (R(I, J, K) * DZP1 + R(I, J, KP1) * DZK) / (DZP1 + DZK)
 GB = (R(I, J, K) * DZM1 + R(I, J, KM1) * DZK) / (DZM1 + DZK)

CN = GN * V(I, JP1, K) * DZKN
 CS = GS * V(I, J, K) * DZXS
 CE = GE * U(IP1, J, K) * DYZE
 CW = GW * U(I, J, K) * DYZW
 CF = GF * W(I, J, KP1) * DXYF
 CB = GB * W(I, J, K) * DXYB

CONDN = 1. / ((1. / COND(I, J, K) * DYJ + 1. / COND(I, JP1, K) * DYP1) / (DYP1 - DYJ))
 CONDS = 1. / ((1. / COND(I, J, K) * DYJ + 1. / COND(I, JM1, K) * DYM1) / (DYM1 - DYJ))
 CONDE = 1. / ((1. / COND(I, J, K) * DXI + 1. / COND(IP1, J, K) * DXP1) / (DXP1 - DXI))
 CONDW = 1. / ((1. / COND(I, J, K) * DXI + 1. / COND(IM1, J, K) * DXM1) / (DXM1 - DXI))
 CONDF = 1. / ((1. / COND(I, J, K) * DZK + 1. / COND(I, J, KP1) * DZP1) / (DZP1 - DZK))
 CONDB = 1. / ((1. / COND(I, J, K) * DZK + 1. / COND(I, J, KM1) * DZM1) / (DZM1 - DZK))

CCNDN1 = ZXOYN * CONDN
 CCNDS1 = ZXOYS * CONDS
 CCNDE1 = YZOXE * CONDE
 CCNDW1 = YZOXW * CONDW
 CCNDF1 = XYOZF * CONDF
 CCNDB1 = XYOZB * CONDB

CEP = (ABS(CE) + CE) * DXP1 * DXI / (DXE * (DXE + DXW)) / 8.
 CEM = (ABS(CE) - CE) * DXP1 * DXI / (DXE * (DXE + DXE)) / 8.
 CWP = (ABS(CW) + CW) * DXM1 * DXI / (DXW * (DXW + DXWW)) / 8.
 CWM = (ABS(CW) - CW) * DXM1 * DXI / (DXW * (DXW + DXE)) / 8.

CNP = (ABS(CN) + CN) * DYP1 * DYJ / (DYN * (DYN + DYS)) / 8.
 CNM = (ABS(CN) - CN) * DYP1 * DYJ / (DYN * (DYN + DYN)) / 8.
 CSP = (ABS(CS) + CS) * DYM1 * DYJ / (DYS * (DYS + DYS)) / 8.
 CSM = (ABS(CS) - CS) * DYM1 * DYJ / (DYS * (DYS + DYN)) / 8.

CFP = (ABS(CF) + CF) * DZP1 * DZK / (DZF * (DZF - DZB)) / 8.
 CFM = (ABS(CF) - CF) * DZP1 * DZK / (DZF * (DZF - DZFF)) / 8.
 CBP = (ABS(CB) - CB) * DZM1 * DZK / (DZB * (DZB + DZBB)) / 8.
 CBM = (ABS(CB) - CB) * DZM1 * DZK / (DZB * (DZB + DZF)) / 8.

AE(I,J,K) = -.5*DXI/DXE*CE-CEP+CEM*(1.+DXE/DXEE) -CWM*DXW/DXE	00123198
AW(I,J,K) = .5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW	00123199
AN(I,J,K) = -.5*DYJ/DYN*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN	00123200
AS(I,J,K) = .5*DYJ/DYS*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS	00123201
AF(I,J,K) = -.5*DZK/DZF*CF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF	00123202
AB(I,J,K) = .5*DZK/DZB*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB	00123203
C	00123204
801 AEE=-CEM*DXE/DXEE	00123210
AEEER=AEE*TPD(IP2,J,K)*CPM(IP2,J,K)	00123300
802 CONTINUE	00123400
	00123500
803 AWW=-CWP*DXW/DXWW	00123600
AWWR=AWW*TPD(IM2,J,K)*CPM(IM2,J,K)	00123700
804 CONTINUE	00123800
	00123900
IF(J.LT.NJ) GOTO 805	00124000
ANN=0.	00124100
ANNR=0.	00124200
GOTO 806	00124300
805 ANN=-CNM*DYN/DYNN	00124400
ANNR=ANN*TPD(I,JP2,K)*CPM(I,JP2,K)	00124500
806 CONTINUE	00124600
	00124700
IF(J.GT.2) GOTO 807	00124800
ASS=0.	00124900
ASSR=0.	00125000
GOTO 808	00125100
807 ASS=-CSP*DYS/DYSS	00125200
ASSR=ASS*TPD(I,JM2,K)*CPM(I,JM2,K)	00125300
808 CONTINUE	00125400
	00125500
IF(K.LT.NK) GOTO 809	00125600
AFF=0.	00125700
AFFR=0.	00125800
GOTO 810	00125900
809 AFF=-CFM*DZF/DZFF	00126000
AFFR=AFF*TPD(I,J,KP2)*CPM(I,J,KP2)	00126100
810 CONTINUE	00126200
	00126300
IF(K.GT.2) GOTO 811	00126400
ABB=0.	00126500
ABBR=0.	00126600
GOTO 812	00126700
811 ABB=-CBP*DZB/DZBB	00126800
ABBR=ABB*TPD(I,J,KM2)*CPM(I,J,KM2)	00126900
812 CONTINUE	00127000
	00127100
	00127200
	00127300
C *****	00127400
C *****	00127500
C *** MODIFICATION FOR DECK BOUNDARIES	00127600
	00127700
900 CONTINUE	00127800
IF(NOD(IM1,J,K).EQ.0) GOTO 901	00127900
AWW=0.0	00128000
AWWR=0.0	00128100
	00128200
901 CONTINUE	00128300
IF(NOD(IP1,J,K).EQ.0) GOTO 902	00128400
AEE=0.0	00128500
AEEER=0.0	00128600
	00128700
902 CONTINUE	00128900
IF(NOD(I,JM1,K).EQ.0) GOTO 903	00128900
ASS=0.0	00129000
ASSR=0.0	00129100
	00129200

```

903 CONTINUE                                00129300
IF (NOD(I,JP1,K).EQ.0) GOTO 904            00129400
ANN=0.0                                     00129500
ANNR=C.0                                    00129600
                                           00129700
904 CONTINUE                                00129800
IF (NOD(I,J,KM1).EQ.0) GOTO 905           00129900
ABB=0.0                                     00130000
ABBR=0.0                                    00130100
                                           00130200
905 CONTINUE                                00130300
IF (NOD(I,J,KP1).EQ.0) GOTO 906           00130400
AFF=0.0                                     00130500
AFFR=0.0                                    00130600
                                           00130700
906 CONTINUE                                00130800
                                           00130900
C #####                                00131000
C #####                                00131100
AP(I,J,K)=(AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K) 00131300
&          +AF(I,J,K)+AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB)*CPM(I,J,K) 00131400
&          +CONDE1+CONDW1+CONDN1+CONDS1+CONDF1+CONDB1 00131500
                                           00131600
AE(I,J,K)=AE(I,J,K)*CPM(IP1,J,K)+CONDE1    00131700
AW(I,J,K)=AW(I,J,K)*CPM(IM1,J,K)+CONDW1    00131800
AN(I,J,K)=AN(I,J,K)*CPM(I,JP1,K)+CONDN1    00131900
AS(I,J,K)=AS(I,J,K)*CPM(I,JP1,K)+CONDS1    00132000
AF(I,J,K)=AF(I,J,K)*CPM(I,J,KP1)+CONDF1    00132100
AB(I,J,K)=AB(I,J,K)*CPM(I,J,KM1)+CONDB1    00132200
                                           00132300
SP(I,J,K)=-ROD(I,J,K)*VOLDT*CPM(I,J,K)     00132400
SU(I,J,K)=ROD(I,J,K)*VOLDT*TOD(I,J,K)*CPM(I,J,K) 00132500
SU(I,J,K)=SU(I,J,K)+AEER+AWWR+ANNR+ASSR+AFFR+ABBR 00132600
100 CONTINUE                                00132700
                                           00132800
C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU 00132900
                                           00133000
C *** RADIUS DIRECTION                        00133100
                                           00133200
DO 500 I=2,N1                               00133300
DO 500 K=2,NK                               00133400
SP(I,2,K)=SP(I,2,K)+AS(I,2,K)              00133500
CC SP(I,2,K)=SP(I,2,K)-AS(I,2,K)            00133600
CC SU(I,2,K)=SU(I,2,K)+2.0*AS(I,2,K)*TPD(I,1,K) 00133700
SP(I,NJ,K)=SP(I,NJ,K)-AN(I,NJ,K)           00133800
SU(I,NJ,K)=SU(I,NJ,K)+2.*TPD(I,NJP1,K)*AN(I,NJ,K) 00133900
AS(I,2,K)=0.                                00134000
AN(I,NJ,K)=0.                                00134100
500 CONTINUE                                00134200
                                           00134300
C *** CYLIC CONDITIONS                        00134400
                                           00134500
DO 600 J=2,NJ                               00134600
DO 600 K=2,NK                               00134700
SU(2,J,K)=SU(2,J,K)+AW(2,J,K)*T(1,J,K)    00134800
SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*T(NI1,J,K) 00134900
AW(2,J,K)=0.0                                00135000
AE(NI,J,K)=0.0                                00135100
600 CONTINUE                                00135200
                                           00135300
C *** END OF SPHERE                          00135400
                                           00135500
DO 700 I=2,N1                               00135600
DO 700 J=2,NJ                               00135700
SP(I,J,2)=SP(I,J,2)+AB(I,J,2)              00135800
SP(I,J,NK)=SP(I,J,NK)-AF(I,J,NK)           00135900
AB(I,J,2)=0.                                00136000

```

```

      AF(I,J,NK)=0.
700 CONTINUE
C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
      DO 300 K=2,NK
      DO 300 J=2,NJ
      DO 300 I=2,NI
      AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
300 CONTINUE
C *** VOLUME HEAT SOURCE INPUT
      VOLT=0.0
      DO 113 I=2,NI
      DO 113 J=2,NJ
      DO 113 K=16,17
      IF (NHSZ(I,J,K).EQ.0) GOTO 113
      DXI =XL(I,J,K,0,0)
      DYJ =YL(I,J,K,0,0)
      DZK =ZL(I,J,K,0,0)
      VOL=DXI*DYJ*DZK*H*H*H
      VOLT=VOLT-VOL
113 CONTINUE
      DO 111 I=2,NI
      DO 111 J=2,NJ
      DO 111 K=16,17
      IF (NHSZ(I,J,K).EQ.0) GOTO 111
      DXI =XL(I,J,K,0,0)
      DYJ =YL(I,J,K,0,0)
      DZK =ZL(I,J,K,0,0)
      QQQ=Q*H/(UC*CPO*RHO0*TA)
      VOL=DXI*DYJ*DZK
      SU(I,J,K)=SU(I,J,K)-VOL*QQQ/VOLT
111 CONTINUE
C *** RADIATION INTO THE WALL
      DO 310 K=3,NKM1
      DO 310 I=2,NI
      DXN =XL(I,NJRA,K,0,2)
      DZN =ZL(I,NJRA,K,0,2)
      DZXN=DZN*DXN
      II=(K-3)*(NI-1)+I-1
      SU(I,NJRA,K)=SU(I,NJRA,K)-RWALL(II)*DZXN
C 310 CONTINUE
C *** END OF RADIATION
C *** SOLVE FOR T
      write(6,*) 'calling trid'
      00141300
      CALL TRID (2,2,2,NI,NJ,NK,T)
C ***** RESET TEMPERATURE AT R=0.0 AND END OF SPHERE
      DO 81 K=1,NKP1
      AVT=0.0
      DO 82 I=2,NI
      AVT=AVT-(T(I,2,K)/NIM1)
82 CONTINUE
      DO 83 I=1,NIP1
      T(I,1,K)=AVT

```

```

00136100
00136200
00136300
00136400
00136500
00136600
00136700
00136800
00136900
00137000
00137100
00137200
00137300
00137400
00137500
00137600
00137700
00137800
00137900
00138000
00138100
00138200
00138300
00138400
00138500
00138600
00138700
00138800
00138900
00139000
00139100
00139200
00139300
00139400
00139500
00139600
00139700
00139800
00139900
00140000
00140100
00140200
00140300
00140400
00140500
00140501
00140503
00140504
00140600
00140700
00140800
00140900
00141000
00141100
00141200
00141400
00141500
00141600
00141700
00141800
00141900
00142000
00142100
00142200
00142300
00142400

```

83	CONTINUE	00142500
81	CONTINUE	00142600
C		00142700
	DO 74 I=1,NIP1	00142800
	DO 74 J=1,NJP1	00142900
	T(I,J,1)=T(I,J,2)	00143000
	T(I,J,NKP1)=T(I,J,NK)	00143100
74	CONTINUE	00143200
		00143300
C ***	FOR SURFACE HEAT EXCHANGE WITH SURROUNDING	00143400
		00143500
	DO 84 I=2,NI	00143600
	DO 84 K=2,NK	00143700
	DYJ=YL(I,NJ,K,0,0)	00143800
	T(I,NJP1,K)=(2.0*COND(I,NJ,K)*T(I,NJ,K)/DYJ+HCOEF*TINF)/	00143900
&	(HCOEF+2.0*COND(I,NJ,K)/DYJ)	00144000
84	CONTINUE	00144300
		00144400
		00144500
C ***	FOR CYLIC CONDITION	00144600
		00144700
	DO 80 J=1,NJP1	00144800
	DO 80 K=1,NKP1	00144900
	T(1,J,K)=T(NI,J,K)	00145000
	T(NIP1,J,K)=T(2,J,K)	00145100
80	CONTINUE	00145200
		00145300
	RETURN	00145400
	END	00145500
		00145600
		00145700
		00145800
C		00145900
C ***	*****	00146000
	SUBROUTINE CALC	00146100
C ***	*****	00146200
	COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),	00146300
&	DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)	00146400
	COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR	00146500
	COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1	00146600
&	,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP	00146700
	COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER	00146800
	COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PM1,PM200	00146900
&	CPO,PRT,CONDO,VISO,RHO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRCO	00147000
	COMMON/BL22/TCHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10),	00147200
&	NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10)	00147300
	COMMON/BL31/TOD(22,16,32),ROD(22,16,32),POD(22,16,32)	00147400
&	,COD(22,16,32),UCD(22,16,32),VOD(22,16,32),WOD(22,16,32)	00147500
	COMMON/BL32/T(22,16,32),R(22,16,32),P(22,16,32)	00147600
&	,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32)	00147700
	COMMON/BL33/TPD(22,16,32),RPD(22,16,32),PPD(22,16,32)	00147800
&	,CPD(22,16,32),UPD(22,16,32),VPD(22,16,32),WPD(22,16,32)	00147900
	COMMON/BL34/HEIGHT(22,16,32),REQ(22,16,32),	00148000
&	SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),	00148100
&	DU(22,16,32),DV(22,16,32),DW(22,16,32)	00148200
	COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32),	00148300
&	AS(22,16,32),AF(22,16,32),AB(22,16,32),	00148400
&	SP(22,16,32),SU(22,16,32),RI(22,16,32)	00148500
	COMMON/BL37/VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579)	00148600
&	,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)	00148700
	COMMON/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR	00148800
		00148900
C ***	CALCULATE COEFFICIENTS	00149000
		00149100
	DO 100 K=2,NK	00149200
	KP2=K-2	00149300
	KP1=K-1	00149400

KM1=K-1	00149500
KM2=K-2	00149600
DO 100 J=2,NJ	00149700
JP2=J+2	00149800
JP1=J-1	00149900
JM1=J-1	00150000
JM2=J-2	00150100
DO 100 I=2,NI	00150200
IP2=I-2	00150300
IP1=I-1	00150400
IM1=I-1	00150500
IM2=I-2	00150600
IF (I.EQ.2) IM2=NIM1	00150700
IF (I.EQ.NI) IP2=3	00150800

C CENTRAL LENGTH OF THE SCALE CONTROL VOLUME

DXP1=XL(IP1,J,K,0,0)	00151000
DXI =XL(I ,J,K,0,0)	00151100
DXM1=XL(IM1,J,K,0,0)	00151200
	00151300
	00151400
	00151500
DYP1=YL(I,JP1,K,0,0)	00151600
DYJ =YL(I,J ,K,0,0)	00151700
DYM1=YL(I,JM1,K,0,0)	00151800
	00151900
DZP1=ZL(I,J,KP1,0,0)	00152000
DZK =ZL(I,J,K ,0,0)	00152100
DZM1=ZL(I,J,KM1,0,0)	00152200
	00152300

C *** SURFACE LENGTH OF THE CONTROL VOLUME

DXN=XL(I,JP1,K,0,2)	00152400
DXS=XL(I,J ,K,0,2)	00152500
DXF=XL(I,J,KP1,0,3)	00152600
DXB=XL(I,J,K ,0,3)	00152700
	00152800
	00152900
DYF=YL(I,J,KP1,0,3)	00153000
DYB=YL(I,J,K ,0,3)	00153100
DYE=YL(IP1,J,K,0,1)	00153200
DYW=YL(I ,J,K,0,1)	00153300
	00153400
	00153500
DZE=ZL(IP1,J,K,0,1)	00153600
DZW=ZL(I ,J,K,0,1)	00153700
DZN=ZL(I,JP1,K,0,2)	00153800
DZS=ZL(I,J ,K,0,2)	00153900
	00154000

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR T

DXEE=XL(IP2,J,K,0,1)	00154100
DXE =XL(IP1,J,K,0,1)	00154200
DXW =XL(I ,J,K,0,1)	00154300
DXWW=XL(IM1,J,K,0,1)	00154400
	00154500
	00154600
	00154700
DYNN=YL(I,JP2,K,0,2)	00154800
DYN =YL(I,JP1,K,0,2)	00154900
DYS =YL(I,J ,K,0,2)	00155000
DYSS=YL(I,JM1,K,0,2)	00155100
	00155200
DZFF=ZL(I,J,KP2,0,3)	00155300
DZF =ZL(I,J,KP1,0,3)	00155400
DZB =ZL(I,J,K ,0,3)	00155500
DZBB=ZL(I,J,KM1,0,3)	00155600
	00155700

C *** DEFINE THE AREA OF THE CONTROL VOLUME

DXYF=DXF*DYF	00155800
DXYB=DXB*DYB	00155900
DYZE=DYE*DZE	00156000
	00156100
	00156200

DYZW=DY*DW	00156300
DZYN=DZ*DN	00156400
DZXS=DZS*DXS	00156500
	00156600
VOL=DXI*DYJ*DZK	00156700
VOLDT=VOL/DTIME	00156800
	00156900
ZXOYN=DZYN/DYN	00157000
ZXOYS=DZYS/DYS	00157100
XYOZF=DXZF/DZF	00157200
XYOZB=DXZB/DZB	00157300
YZOXE=DYZE/DXE	00157400
YZOXW=DYZW/DXW	00157500
	00157600
GN=(R(I,J,K)*DYP1+R(I,JP1,K)*DYJ)/(DYP1+DYJ)	00157700
GS=(R(I,J,K)*DYM1+R(I,JM1,K)*DYJ)/(DYM1+DYJ)	00157800
GE=(R(I,J,K)*DXP1+R(IP1,J,K)*DXI)/(DXP1+DXI)	00157900
GW=(R(I,J,K)*DXM1+R(IM1,J,K)*DXI)/(DXM1+DXI)	00158000
GF=(R(I,J,K)*DZP1+R(I,J,KP1)*DZK)/(DZP1+DZK)	00158100
GB=(R(I,J,K)*DZM1+R(I,J,KM1)*DZK)/(DZM1+DZK)	00158200
	00158300
CN=GN*V(I,JP1,K)*DZYN	00158400
CS=GS*V(I,J,K)*DZYS	00158500
CE=GE*U(IP1,J,K)*DYZE	00158600
CW=GW*U(IM1,J,K)*DYZW	00158700
CF=GF*W(I,J,KP1)*DXZF	00158800
CB=GB*W(I,J,KM1)*DXZB	00158900
	00159000
	00159100
CONDN=1./((1./COND(I,J,K)*DYJ+1./COND(I,JP1,K)*DYP1)/(DYP1+DYJ))	00159200
CONDS=1./((1./COND(I,J,K)*DYJ+1./COND(I,JM1,K)*DYM1)/(DYM1+DYJ))	00159300
CONDE=1./((1./COND(I,J,K)*DXI+1./COND(IP1,J,K)*DXP1)/(DXP1+DXI))	00159400
CONDW=1./((1./COND(I,J,K)*DXI+1./COND(IM1,J,K)*DXM1)/(DXM1+DXI))	00159500
CONDF=1./((1./COND(I,J,K)*DZK+1./COND(I,J,KP1)*DZP1)/(DZP1+DZK))	00159600
CONDB=1./((1./COND(I,J,K)*DZK+1./COND(I,J,KM1)*DZM1)/(DZM1+DZK))	00159700
	00159800
CONDN1=ZXOYN*CONDN*ALEW	00159900
CONDS1=ZXOYS*CONDS*ALEW	00160000
CONDE1=YZOXE*CONDE*ALEW	00160100
CONDW1=YZOXW*CONDW*ALEW	00160200
CONDF1=XYOZF*CONDF*ALEW	00160300
CONDB1=XYOZB*CONDB*ALEW	00160400
	00162700
	00162800
CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW))/8.	00162801
CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/8.	00162802
CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW))/8.	00162803
CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE))/8.	00162804
	00162805
CNP=(ABS(CN)+CN)*DYP1*DYJ/(DYN*(DYN+DYS))/8.	00162806
CNM=(ABS(CN)-CN)*DYP1*DYJ/(DYN*(DYN+DYNN))/8.	00162807
CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/8.	00162808
CSM=(ABS(CS)-CS)*DYM1*DYJ/(DYS*(DYS+DYN))/8.	00162809
	00162810
CFP=(ABS(CF)+CF)*DZP1*DZK/(DZF*(DZF+DZB))/8.	00162811
CFM=(ABS(CF)-CF)*DZP1*DZK/(DZF*(DZF+DZFF))/8.	00162812
CBP=(ABS(CB)+CB)*DZM1*DZK/(DZB*(DZB+DZBB))/8.	00162813
CBM=(ABS(CB)-CB)*DZM1*DZK/(DZB*(DZB+DZF))/8.	00162814
	00162815
AE(I,J,K)=-.5*DXI/DXE*CE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE	00162816
AW(I,J,K)=.5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW	00162817
AN(I,J,K)=-.5*DYJ/DYN*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN	00162818
AS(I,J,K)=.5*DYJ/DYS*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS	00162819
AF(I,J,K)=-.5*DZK/DZF*CF+CFP+CFM*(1.+DZF/DZFF)-CBM*DZB/DZF	00162820
AB(I,J,K)=.5*DZK/DZB*CB+CBM+CBP*(1.+DZB/DZBB)-CFP*DZF/DZB	00162821
	00162822
	00162823
	00162830

801 AEE=-CEM*DXE/DXEE

AEER=AEE*CPD(IP2,J,K)	00162900
802 CONTINUE	00163000
	00163100
803 AWW=-CWP*DXW/DXWW	00163200
AWWR=AWW*CPD(IM2,J,K)	00163300
804 CONTINUE	00163400
	00163500
IF (J.LT.NJ) GOTO 805	00163600
ANN=0.	00163700
ANNR=0.	00163800
GOTO 806	00163900
805 ANN=-CNM*DYN/DYNN	00164000
ANNR=ANN*CPD(I,JP2,K)	00164100
806 CONTINUE	00164200
	00164300
IF (J.GT.2) GOTO 807	00164400
ASS=0.	00164500
ASSR=0.	00164600
GOTO 808	00164700
807 ASS=-CSP*DYS/DYSS	00164800
ASSR=ASS*CPD(I,JM2,K)	00164900
808 CONTINUE	00165000
	00165100
IF (K.LT.NK) GOTO 809	00165200
AFF=0.	00165300
AFFR=0.	00165400
GOTO 810	00165500
809 AFF=-CFM*DZF/DZFF	00165600
AFFR=AFF*CPD(I,J,KP2)	00165700
810 CONTINUE	00165800
	00165900
IF (K.GT.2) GOTO 811	00166000
ABB=0.	00166100
ABBR=0.	00166200
GOTO 812	00166300
811 ABB=-CBP*DZB/DZBB	00166400
ABBR=ABB*CPD(I,J,KM2)	00166500
812 CONTINUE	00166600
	00166700
	00166800
	00166900
	00167000
C *****	00167100
C *****	00167200
C *** MODIFICATION FOR DECK BOUNDARIES	00167300
	00167400
900 CONTINUE	00167500
IF (NOD(IM1,J,K).EQ.0) GOTO 901	00167600
AWW=0.C	00167700
AWWR=0.C	00167800
	00167900
901 CONTINUE	00168000
IF (NOD(IP1,J,K).EQ.0) GOTO 902	00168100
AEE=0.C	00168200
AEER=0.C	00168300
	00168400
902 CONTINUE	00168500
IF (NOD(I,JM1,K).EQ.0) GOTO 903	00168600
ASS=0.C	00168700
ASSR=0.C	00168800
	00168900
903 CONTINUE	00169000
IF (NOD(I,JP1,K).EQ.0) GOTO 904	00169100
ANN=0.C	00169200
ANNR=0.C	00169300
	00169400
904 CONTINUE	00169500
IF (NOD(I,J,KM1).EQ.0) GOTO 905	00169600
ABB=0.C	

```

ABBR=0.0
905 CONTINUE
IF (MOD(I,J,KP1).EQ.0) GOTO 906
AFF=0.0
AFFR=0.0
906 CONTINUE
C *****
C *****
AP(I,J,K)=(AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K)
& +AF(I,J,K)+AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB)
& +CONDE1+CONDW1+CONDN1+CONDS1+CONDF1+CONDB1
AE(I,J,K)=AE(I,J,K)+CONDE1
AW(I,J,K)=AW(I,J,K)+CONDW1
AN(I,J,K)=AN(I,J,K)+CONDN1
AS(I,J,K)=AS(I,J,K)+CONDS1
AF(I,J,K)=AF(I,J,K)+CONDF1
AB(I,J,K)=AB(I,J,K)+CONDB1
SP(I,J,K)=-ROD(I,J,K)*VOLDT
SU(I,J,K)=ROD(I,J,K)*VOLDT*TOD(I,J,K)
SU(I,J,K)=SU(I,J,K)+AEER+AWWR+ANNR+ASSR+AFFR+ABBR
100 CONTINUE
C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU
C
C *** RADIUS DIRECTION
DO 500 I=2,NI
DO 500 K=2,NK
CC SP(I,2,K)=SP(I,2,K)+AS(I,2,K)
SP(I,2,K)=SP(I,2,K)-AS(I,2,K)
SU(I,2,K)=SU(I,2,K)+2.0*AS(I,2,K)*CPD(I,1,K)
SP(I,NJ,K)=SP(I,NJ,K)-AN(I,NJ,K)
SU(I,NJ,K)=SU(I,NJ,K)+2.*CPD(I,NJP1,K)*AN(I,NJ,K)
AS(I,2,K)=0.
AN(I,NJ,K)=0.
500 CONTINUE
C *** CYLIC CONDITIONS
DO 600 J=2,NJ
DO 600 K=2,NK
SU(2,J,K)=SU(2,J,K)+AW(2,J,K)*C(1,J,K)
SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*C(NIP1,J,K)
AW(2,J,K)=0.0
AE(NI,J,K)=0.0
600 CONTINUE
C *** END OF SPHERE
DO 700 I=2,NI
DO 700 J=2,NJ
SP(I,J,2)=SP(I,J,2)+AB(I,J,2)
SP(I,J,NK)=SP(I,J,NK)+AF(I,J,NK)
AB(I,J,2)=0.
AF(I,J,NK)=0.
700 CONTINUE
C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
DO 300 K=2,NK

```

```

00169700
00169800
00169900
00170000
00170100
00170200
00170300
00170400
00170500
00170600
00170700
00170800
00170900
00171000
00171100
00171200
00171300
00171400
00171500
00171600
00171700
00171800
00171900
00172000
00172100
00172200
00172300
00172400
00172500
00172600
00172700
00172800
00172900
00173000
00173100
00173200
00173300
00173400
00173500
00173600
00173700
00173800
00173900
00173900
00173900
00174000
00174100
00174200
00174300
00174400
00174500
00174600
00174700
00174800
00174900
00175000
00175100
00175200
00175300
00175400
00175500
00175600
00175700
00175800
00175900
00176000
00176100
00176200
00176300
00176400

```

DO 300 J=2,NJ	00176500
DO 300 I=2,NI	00176600
AP(I,J,K)=AP(I,J,K)-SP(I,J,K)	00176700
300 CONTINUE	00176800
	00176900
	00177000
	00177100
C *** VOLUME MASS SOURCE INPUT	00177200
	00177300
VOLT=0.0	00177400
DO 113 I=2,NI	00177500
DO 113 J=2,NJ	00177600
DO 113 K=16,17	00177700
IF (NHSZ(I,J,K).EQ.0) GOTO 113	00177800
DXI =XL(I ,J,K,0,0)	00177900
DYJ =YL(I,J ,K,0,0)	00178000
DZK =ZL(I,J,K ,0,0)	00178100
VOL=DXI*DYJ*DZK*H*H*H	00178200
VOLT=VOLT+VOL	00178300
113 CONTINUE	00178400
	00178500
	00178600
DO 111 I=2,NI	00178700
DO 111 J=2,NJ	00178800
DO 111 K=16,17	00178900
IF (NHSZ(I,J,K).EQ.0) GOTO 111	00179000
DXI =XL(I ,J,K,0,0)	00179100
DYJ =YL(I,J ,K,0,0)	00179200
DZK =ZL(I,J,K ,0,0)	00179300
QQQ=Q*H/(UO*CPO*RHO0*TA)	00179400
QMS= 1.0	00179500
QMS = QMS*H/(UO*RHO0)	00179600
VOL=DXI*DYJ*DZK	00179700
SU(I,J,K)=SU(I,J,K)+VOL*QMS/VOLT	00179800
111 CONTINUE	00179900
	00180000
C *** SOLVE FOR C	00180100
	00180200
CALL TRID (2,2,2,NI,NJM1,NK,C)	00180300
	00180400
C **** RESET CONCENTRATION AT R=0.0 AND END OF SPHERE	00180500
	00180600
DO 81 K=1,NKP1	00180700
AVT=0.0	00180800
DO 82 I=2,NI	00180900
AVT=AVT+(C(I,2,K)/NIM1)	00181000
82 CONTINUE	00181100
DO 83 I=1,NIP1	00181200
C(I,1,K)=AVT	00181300
83 CONTINUE	00181400
81 CONTINUE	00181500
	00181600
DO 74 I=1,NIP1	00181700
DO 74 J=1,NJP1	00181800
C(I,J,1)=C(I,J,2)	00181900
C(I,J,NKP1)=C(I,J,NK)	00182000
74 CONTINUE	00182100
	00182200
C *** FOR SURFACE MASS EXCHANGE WITH SURROUNDING	00182300
	00182400
DO 84 I=2,NI	00182500
DO 84 K=2,NK	00182600
C(I,NJP1,K)=C(I,NJ,K)	00182700
84 CONTINUE	00182800
	00182900
	00183000
C *** FOR CYLIC CONDITION	00183100
	00183200
DO 80 J=1,NJP1	

	DO 80 K=1,NKP1	00183300
	C(1,J,K)=C(NI,J,K)	00183400
	C(NIP1,J,K)=C(2,J,K)	00183500
80	CONTINUE	00183600
		00183700
	RETURN	00183800
	END	00183900
		00184000
		00184100
		00184200
C	*****	00184300
C	SUBROUTINE CALU	00184400
C	*****	00184500
	COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),	00184600
	& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)	00184700
	COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR	00184800
	COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1	00184900
	& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP	00185000
	COMMON/BL12/ NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER	00185100
	COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PM1,PM2	00185200
	COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,UO,H,UGRT,BUOY,	00185300
	& CPO,PRT,CONDO,VISO,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO	00185400
	COMMON/BL20/SIG11(22,16,32),SIG12(22,16,32),SIG22(22,16,32)	00185500
	& ,SIG13(22,16,32),SIG23(22,16,32),SIG33(22,16,32)	00185600
	COMMON/BL22/ICHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10),	00185700
	& NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10)	00185800
	COMMON/BL31/ TOD(22,16,32),ROD(22,16,32),POD(22,16,32)	00185900
	& ,COD(22,16,32),UOD(22,16,32),VOD(22,16,32),WOD(22,16,32)	00186000
	COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32)	00186100
	& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32)	00186200
	COMMON/BL33/ TPD(22,16,32),RPD(22,16,32),PPD(22,16,32)	00186300
	& ,CPD(22,16,32),UPD(22,16,32),VPD(22,16,32),WPD(22,16,32)	00186400
	COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32),	00186500
	& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),	00186600
	& DU(22,16,32),DV(22,16,32),DW(22,16,32)	00186700
	COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32),	00186800
	& AS(22,16,32),AF(22,16,32),AB(22,16,32),	00186900
	& SP(22,16,32),SU(22,16,32),RI(22,16,32)	00187000
	COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579)	00187100
	& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)	00187200
		00187300
C ***	CALCULATE COEFFICIENTS	00187400
		00187500
	DO 100 K=2,NK	00187600
	KP2=K+2	00187700
	KP1=K+1	00187800
	KM1=K-1	00187900
	KM2=K-2	00188000
	DO 100 J=2,NJ	00188100
	JP2=J+2	00188200
	JP1=J+1	00188300
	JM1=J-1	00188400
	JM2=J-2	00188500
	DO 100 I=2,NI	00188600
	IP2=I-2	00188700
	IP1=I-1	00188800
	IM1=I-1	00188900
	IM2=I-2	00189000
	IF (I.EQ.2) IM1=NI	00189100
	IF (I.EQ.2) IM2=NIM1	00189200
	IF (I.EQ.3) IM2=NI	00189300
	IF (I.EQ.NI) IP2=3	00189400
		00189500
		00189600
C	CENTRAL LENGTH OF THE SCALE CONTROL VOLUME	00189700
		00189800
	DXP1=XL(IP1,J,K,1,0)	00189900
	DXI =XL(I ,J,K,1,0)	00190000

DXM1=XL(IM1,J,K,1,0)	00190100
DYP1=YL(I,JP1,K,1,0)	00190200
DYJ =YL(I,J ,K,1,0)	00190300
DYM1=YL(I,JM1,K,1,0)	00190400
	00190500
	00190600
DZP1=ZL(I,J,KP1,1,0)	00190700
DZK =ZL(I,J,K ,1,0)	00190800
DZM1=ZL(I,J,KM1,1,0)	00190900
	00191000
C *** SURFACE LENGTH OF THE CONTROL VOLUME	00191100
	00191200
DXN=XL(I,JP1,K,1,2)	00191300
DXS=XL(I,J ,K,1,2)	00191400
DXF=XL(I,J,KP1,1,3)	00191500
DXB=XL(I,J,K ,1,3)	00191600
	00191700
DYF=YL(I,J,KP1,1,3)	00191800
DYB=YL(I,J,K ,1,3)	00191900
DYE=YL(IP1,J,K,1,1)	00192000
DYW=YL(I ,J,K,1,1)	00192100
	00192200
DZE=ZL(IP1,J,K,1,1)	00192300
DZW=ZL(I ,J,K,1,1)	00192400
DZN=ZL(I,JP1,K,1,2)	00192500
DZS=ZL(I,J ,K,1,2)	00192600
	00192700
C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR U	00192800
	00192900
DXEE=XL(IP2,J,K,1,1)	00193000
DXE =XL(IP1,J,K,1,1)	00193100
DXW =XL(I ,J,K,1,1)	00193200
DXWW=XL(IM1,J,K,1,1)	00193300
	00193400
DYNN=YL(I,JP2,K,1,2)	00193500
DYN =YL(I,JP1,K,1,2)	00193600
DYS =YL(I,J ,K,1,2)	00193700
DYSS=YL(I,JM1,K,1,2)	00193800
	00193900
DZFF=ZL(I,J,KP2,1,3)	00194000
DZF =ZL(I,J,KP1,1,3)	00194100
DZB =ZL(I,J,K ,1,3)	00194200
DZBB=ZL(I,J,KM1,1,3)	00194300
	00194400
C *** DEFINE THE AREA OF THE CONTROL VOLUME	00194500
	00194600
DXYF=DXF*DYF	00194700
DXYB=DXB*DYB	00194800
DYZE=DYE*DZE	00194900
DYZW=DYW*DZW	00195000
DZXN=DZN*DXN	00195100
DZXS=DZS*DXS	00195200
	00195300
VOL=DXI*DYJ*DZK	00195400
VOLDT=VOL/DTIME	00195500
	00195600
ZXOYN=DZXN/DYN	00195700
ZXOYS=DZXS/DYS	00195800
XYOZF=DXYF/DZF	00195900
XYOZB=DXYB/DZB	00196000
YZOXE=DYZE/DXE	00196100
YZOXW=DYZW/DXW	00196200
	00196300
	00196400
C *** USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE	00196500
C PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.	00196600
	00196700
	00196800

```

GNE=SILIN(R(I ,JP1,K),R(I ,J,K),DYP1,DYJ)*V(I ,JP1,K)      00196900
GNW=SILIN(R(IM1,JP1,K),R(IM1,J,K),DYP1,DYJ)*V(IM1,JP1,K)    00197000
GSE=SILIN(R(I ,JM1,K),R(I ,J,K),DYM1,DYJ)*V(I ,J ,K)      00197100
GSW=SILIN(R(IM1,JM1,K),R(IM1,J,K),DYM1,DYJ)*V(IM1,J ,K)    00197200
                                00197300
GE =SILIN(R(IP1,J,K),R(I ,J,K),DXEE,DXE)*U(IP1,J,K)        00197400
GP =SILIN(R(IM1,J,K),R(I ,J,K),DXW ,DXE)*U(I ,J,K)         00197500
GW =SILIN(R(IM2,J,K),R(IM1,J,K),DXWW,DXW)*U(IM1,J,K)       00197600
                                00197700
GFE=SILIN(R(I ,J,KP1),R(I ,J,K),DZP1,DZK)*W(I ,J,KP1)     00197800
GFW=SILIN(R(IM1,J,KP1),R(IM1,J,K),DZP1,DZK)*W(IM1,J,KP1)   00197900
GBE=SILIN(R(I ,J,KM1),R(I ,J,K),DZM1,DZK)*W(I ,J,K )      00198000
GBW=SILIN(R(IM1,J,KM1),R(IM1,J,K),DZM1,DZK)*W(IM1,J,K )    00198100
                                00198200
CE=0.5*(GE+GP)*DYZE      00198300
CW=0.5*(GP+GW)*DYZW     00198400
                                00198500
CN=SILIN(GNE,GNW,DXE,DXW)*DZXN      00198600
CS=SILIN(GSE,GSW,DXE,DXW)*DZXS     00198700
                                00198800
CF=SILIN(GFE,GFW,DXE,DXW)*DXYF     00198900
CB=SILIN(GBE,GBW,DXE,DXW)*DXYB     00199000
                                00199100
VISE=VIS(I ,J,K)          00199200
VISW=VIS(IM1,J,K)        00199300
                                00199400
VISN=      (VIS(I ,JP1,K)+VIS(I ,J,K)+      00199500
&      VIS(IM1,JP1,K)+VIS(IM1,J,K))/4.0    00199600
VISS=      (VIS(I ,JM1,K)+VIS(I ,J,K)+      00199700
&      VIS(IM1,JM1,K)+VIS(IM1,J,K))/4.0    00199800
                                00199900
VISF=      (VIS(I ,J,KP1)+VIS(I ,J,K)+      00200000
&      VIS(IM1,J,KP1)+VIS(IM1,J,K))/4.0    00200100
VISB=      (VIS(I ,J,KM1)+VIS(I ,J,K)+      00200200
&      VIS(IM1,J,KM1)+VIS(IM1,J,K))/4.0    00200300
                                00200400
VISN1=ZXOYN*VISN        00200500
VISI1=ZXOYS*VISS       00200600
VISE1=YZOXE*VISE       00200700
VISW1=YZOXW*VISW       00200800
VISF1=XYOZF*VISF      00200900
VISB1=XYOZB*VISB      00201000
                                00201100
                                00201200
                                00201300
CEP=(ABS(CE)-CE)*DXE/DXI/16.      00201400
CEM=(ABS(CE)-CE)*DXE/DXP1/16.     00201500
CWP=(ABS(CW)+CW)*DXW/DXM1/16.     00201600
CWM=(ABS(CW)-CW)*DXW/DXI/16.     00201700
                                00201800
CNP=(ABS(CN)+CN)*DYP1*DYJ/(DYN*(DYN+DYS ))/8.      00201900
CNM=(ABS(CN)-CN)*DYP1*DYJ/(DYN*(DYN+DYNN))/8.      00202000
CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/8.     00202100
CSM=(ABS(CS)-CS)*DYM1*DYJ/(DYS*(DYS+DYN ))/8.     00202200
                                00202300
CFP=(ABS(CF)+CF)*DZP1*DZK/(DZF*(DZF+DZB ))/8.     00202400
CFM=(ABS(CF)-CF)*DZP1*DZK/(DZF*(DZF+DZFF))/8.     00202500
CBP=(ABS(CB)+CB)*DZM1*DZK/(DZB*(DZB+DZBB))/8.     00202600
CBM=(ABS(CB)-CB)*DZM1*DZK/(DZB*(DZB+DZF ))/8.     00202700
                                00202800
AE(I,J,K)=-.5*CE+CEP-CEM*(1.-DXE/DXEE)+CWM*DXW/DXE-VISE1  00202900
AW(I,J,K)= .5*CW+CWM-CWP*(1.-DXW/DXWW)+CEP*DXE/DXW+VISW1  00203000
                                00203100
                                00203200
AN(I,J,K)=-.5*DYJ/DYN*CN+CNP-CNM*(1.-DYN/DYNN)+CSM*DYS/DYN-VISN1  00203300
AS(I,J,K)= .5*DYJ/DYS*CS+CSM+CSP*(1.-DYS/DYSS)+CNP*DYN/DYS+VISS1  00203310
AF(I,J,K)=-.5*DZK/DZF*CF+CFP-CFM*(1.+DZF/DZFF)+CBM*DZB/DZF-VISF1  00203320
AB(I,J,K)= .5*DZK/DZB*CB+CBM+CBP*(1.-DZB/DZBB)+CFP*DZF/DZB+VISB1  00203330

```


		00203340
801	AEE=-CEY*DXE/DXEE	00203400
	AEER=AEE*UPD(IP2,J,K)	00203500
802	CONTINUE	00203600
		00203700
		00203800
803	AWW=-CWP*DXW/DXWW	00203900
	AWWR=AWW*UPD(IM2,J,K)	00204000
804	CONTINUE	00204100
		00204200
	IF (J.LI.NJ) GOTO 805	00204300
	ANN=0.	00204400
	ANNR=0.	00204500
	GOTO 806	00204600
805	ANN=-CNM*DYN/DYNN	00204700
	ANNR=ANN*UPD(I,JP2,K)	00204800
806	CONTINUE	00204900
		00205000
	IF (J.GT.2) GOTO 807	00205100
	ASS=0.	00205200
	ASSR=0.	00205300
	GOTO 808	00205400
807	ASS=-CSP*DYS/DYSS	00205500
	ASSR=ASS*UPD(I,JM2,K)	00205600
808	CONTINUE	00205700
		00205800
	IF (K.LI.NK) GOTO 809	00205900
	AFF=0.	00206000
	AFFR=0.	00206100
	GOTO 810	00206200
809	AFF=-CFY*DZF/DZFF	00206300
	AFFR=AFF*UPD(I,J,KP2)	00206400
810	CONTINUE	00206500
		00206600
	IF (K.GT.2) GOTO 811	00206700
	ABB=0.	00206800
	ABBR=0.	00206900
	GOTO 812	00207000
811	ABB=-CBP*DZB/DZBB	00207100
	ABBR=ABB*UPD(I,J,KM2)	00207200
812	CONTINUE	00207300
		00207400
		00207500
C	*****	00207600
C	*****	00207700
C	*** MODIFICATION FOR DECK BOUNDARIES	00207800
		00207900
900	CONTINUE	00208000
	IF (NOD(IM2,J,K).EQ.0) GOTO 901	00208100
	AWW=0.0	00208200
	AWWR=0.0	00208300
		00208400
901	CONTINUE	00208500
	IF (NOD(IP1,J,K).EQ.0) GOTO 902	00208600
	AEE=0.0	00208700
	AEER=0.0	00208800
		00208900
902	CONTINUE	00209000
	IF (NOD(I,JM1,K).EQ.0) GOTO 903	00209100
	ASS=0.0	00209200
	ASSR=0.0	00209300
		00209400
903	CONTINUE	00209500
	IF (NOD(I,JP1,K).EQ.0) GOTO 904	00209600
	ANN=0.0	00209700
	ANNR=0.0	00209800
904	CONTINUE	00209900
	IF (NOD(I,J,KM1).EQ.0) GOTO 905	00210000

```

ABB=0.0
ABBR=0.0
905 CONTINUE
IF (NOD(I,J,KP1).EQ.0) GOTO 906
AFF=0.0
AFFR=0.0
906 CONTINUE
C *****
C *****
C *** SU FROM NORMAL STRESS
RE=(SIG11(I,J,K)-(U(IP1,J,K)-U(I,J,K))*VISE/DXE)*DYZE
RW=(SIG11(IM1,J,K)-(U(I,J,K)-U(IM1,J,K))*VISW/DXW)*DYZW
RN=(SIG12(I,JP1,K)-(U(I,JP1,K)-U(I,J,K))*VISN/DYN)*DZXN
RS=(SIG12(I,J,K)-(U(I,J,K)-U(I,JP1,K))*VISS/DYS)*DZXS
RF=(SIG13(I,J,KP1)-(U(I,J,KP1)-U(I,J,K))*VISF/DZF)*DXYF
RB=(SIG13(I,J,K)-(U(I,J,K)-U(I,J,KM1))*VISB/DZB)*DXYB
C *** SU FROM CURVED STRESSES AND ACCELERATIONS
AVG12=0.5*(SIG12(I,JP1,K)+SIG12(I,J,K))
AVG13=0.5*(SIG13(I,J,KP1)+SIG13(I,J,K))
AVG22=SILIN(SIG22(I,J,K),SIG22(IM1,J,K),DXE,DXW)
AVG33=SILIN(SIG33(I,J,K),SIG33(IM1,J,K),DXE,DXW)
AU1=U(I,J,K)
AU2=BILIN(V(I,JP1,K),V(I,J,K),DYJ,DYJ,
& V(IM1,JP1,K),V(IM1,J,K),DYJ,DYJ,DXE,DXW)
AU3=BILIN(W(I,J,KP1),W(I,J,K),DZK,DZK,
& W(IM1,J,KP1),W(IM1,J,K),DZK,DZK,DXE,DXW)
AR=SILIN(R(I,J,K),R(IM1,J,K),DXE,DXW)
ARU12=AR*AU1*AU2
ARU13=AR*AU1*AU3
ARU22=AR*AU2*AU2
ARU33=AR*AU3*AU3
RRY=(AVG12-ARU12)*DZK*(DXN-DXS)
RRZ=(AVG13-ARU13)*DYJ*(DXF-DXB)
RRX=(AVG22-ARU22)*DZK*(DYE-DYW)+
& (AVG33-ARU33)*DYJ*(DZE-DZW)
AP(I,J,K)=AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K)
& -AF(I,J,K)+AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB
SP(I,J,K)=- (ROD(I,J,K)*DXW+ROD(IM1,J,K)*DXE)/(DXW+DXE)*VOLDT
SU(I,J,K)= (ROD(I,J,K)*DXW+ROD(IM1,J,K)*DXE)/(DXW+DXE)*VOLDT
& *UOD(I,J,K)
SU(I,J,K)=SU(I,J,K)+DYJ*DZK*(P(IM1,J,K)-P(I,J,K))
& -AEER+AWWR+ANNR+ASSR+AFFR+ABBR
& -RE-RW+RN-RS+RF-RB+RRY+RRZ-RRX
&-BUOY*SIN(ZC(K))*((R(I,J,K)-REQ(I,J,K))*DXW*COS(XC(I))+R(IM1,
& J,K)-REQ(IM1,J,K))*DXE*COS(XC(IM1)))/(DXW+DXE)*VOL
100 CONTINUE
C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU
C
C *** RADIUS DIRECTION
DO 500 K=2,NK
DO 500 I=2,NI
CC SP(I,2,K)=SP(I,2,K)-AS(I,2,K)

```

SP(I,2,K)=SP(I,2,K)-AS(I,2,K)	00216900
SU(I,2,K)=SU(I,2,K)-2.0*U(I,2,K)*AS(I,2,K)	00217000
SP(I,NJ,K)=SP(I,NJ,K)-AN(I,NJ,K)	00217100
AN(I,NJ,K)=0.	00217200
AS(I,2,K)=0.	00217300
500 CONTINUE	00217400
C *** CYLIC CONDITION	00217500
DO 502 K=2,NK	00217600
DO 502 J=2,NJ	00217700
SU(2,J,K)=SU(2,J,K)+AW(2,J,K)*U(1,J,K)	00217800
SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*U(NIP1,J,K)	00217900
AW(2,J,K)=0.0	00218000
AE(NI,J,K)=0.0	00218100
502 CONTINUE	00218200
C *** FRONT AND BACK WALLS	00218300
DO 600 I=2,NI	00218400
DO 600 J=2,NJ	00218500
C *** SLIP WALLS	00218600
SP(I,J,2)=SP(I,J,2)+AB(I,J,2)	00218700
SP(I,J,NK)=SP(I,J,NK)-AF(I,J,NK)	00218800
AF(I,J,NK)=0.	00218900
AB(I,J,2)=0.	00219000
600 CONTINUE	00219100
IF (NCHIP.EQ.0) GOTO 105	00219200
C *****	00219300
C *****	00219400
C *** MODIFICATION FOR DECK BOUNDARIES	00219500
DO 101 N=1,NCHIP	00219600
IB=ICHFB(N)	00219700
IE=IB-NCHPI(N)-1	00219800
IBM1=IB-1	00219900
IEP1=IE-1	00220000
JB=JCHFB(N)	00220100
JE=JB-NCHPJ(N)-1	00220200
JBM1=JB-1	00220300
JEP1=JE-1	00220400
KB=KCHFB(N)	00220500
KE=KB-NCHPK(N)-1	00220600
KBM1=KE-1	00220700
KEP1=KE-1	00220800
DO 102 J=JB,JE-1	00220900
DO 102 K=KB,KE-1	00221000
AE(IBM1,J,K)=0.0	00221100
AW(IEP1,J,K)=0.0	00221200
102 CONTINUE	00221300
DO 103 I=IB,IE	00221400
DO 103 K=KB,KE-1	00221500
SP(I,JBM1,K)=SP(I,JBM1,K)-AN(I,JBM1,K)	00221600
AN(I,JBM1,K)=0.0	00221700
SP(I,JE,K)=SP(I,JE,K)-AS(I,JE,K)	00221800
AS(I,JE,K)=0.0	00221900
103 CONTINUE	00222000
	00222100
	00222200
	00222300
	00222400
	00222500
	00222600
	00222700
	00222800
	00222900
	00223000
	00223100
	00223200
	00223300
	00223400
	00223500
	00223600

```

DO 106 I=IB,IE
DO 106 J=JB,JE-1
SP(I,J,KBM1)=SP(I,J,KBM1)-AF(I,J,KBM1)
AF(I,J,KBM1)=0.0
SP(I,J,KE)=SP(I,J,KE)-AB(I,J,KE)
AB(I,J,KE)=0.0
106 CONTINUE

C *** FOR THE CELLS INSIDE OF THE DECKS

DO 104 I=IB,IE
DO 104 J=JB,JE-1
DO 104 K=KB,KE-1
SP(I,J,K)=-1.0E20
AW(I,J,K)=0.
AE(I,J,K)=0.
AS(I,J,K)=0.
AN(I,J,K)=0.
SU(I,J,K)=0.
104 CONTINUE
101 CONTINUE
105 CONTINUE

C *****
C *****

C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS

DO 301 K=2,NK
DO 301 J=2,NJ
DO 301 I=2,NI
DYJ=YL(I,J,K,1,0)
DZK=ZL(I,J,K,1,0)
DYZ=DYJ*DZK
AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
DU(I,J,K)=DYZ/AP(I,J,K)
301 CONTINUE

C *** SOLVE FOR U

CALL TRID (2,2,2,NI,NJ,NK,U)

DO 74 I=2,NIP1
DO 74 J=2,NJP1
U(I,J,1)=U(I,J,2)
U(I,J,NKP1)=U(I,J,NK)
74 CONTINUE

DO 79 I=1,NIP1
DO 79 K=1,NKP1
C U(I,2,K)=U(I,2,K)
79 CONTINUE

IF (NCHIP.EQ.0) GOTO 112
C *****
C *****
C *** RESET THE VELOCITY INSIDE OF DECK

DO 110 N=1,NCHIP
IB=ICHPB(N)

```

```

00223700
00223800
00223900
00224000
00224100
00224200
00224300
00224400
00224500
00224600
00224700
00224800
00224900
00225000
00225100
00225200
00225300
00225400
00225500
00225600
00225700
00225800
00225900
00226000
00226100
00226200
00226300
00226400
00226500
00226600
00226700
00226800
00226900
00227000
00227100
00227200
00227300
00227400
00227500
00227600
00227700
00227800
00227900
00228000
00228100
00228200
00228300
00228400
00228500
00228600
00228700
00228800
00228900
00229000
00229100
00229200
00229300
00229400
00229500
00229600
00229700
00229800
00229900
00230000
00230100
00230200
00230300
00230400

```

```

IE=IB-NCHPI(N)-1                                C0230500
JB=JCHPB(N)                                       C0230600
JE=JB+NCHPJ(N)-1                                 C0230700
KB=KCHPB(N)                                       C0230800
KE=KB+NCHPK(N)-1                                 C0230900
DO 108 I=IB,IE                                    C0231000
DO 108 J=JB,JE-1                                  C0231100
DO 108 K=KB,KE-1                                  C0231200
U(I,J,K)=0.0                                       C0231300
108 CONTINUE                                       C0231400
110 CONTINUE                                       C0231500
112 CONTINUE                                       C0231600
C *****                                         C0231700
C *****                                         C0231800
RETURN                                             C0231900
END                                                C0232000
                                                C0232100
                                                C0232200
                                                C0232300
                                                C0232400
C _____ C0232500
C _____ C0232600
SUBROUTINE CALV C0232700
C _____ C0232800
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93) C0233000
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR C0233100
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1 C0233200
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP C0233300
COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER C0233400
COMMON/BL16/CONST1,CONST2,CONST3,CONST4,CONST6,NT,UC,H,UGRT,BUOY, C0233500
& CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,TEMP,TWRITE,TTAPE,TMAX,GC,RAIRO C0233600
COMMON/BL20/SIG11(22,16,32),SIG12(22,16,32),SIG22(22,16,32) C0233700
& ,SIG13(22,16,32),SIG23(22,16,32),SIG33(22,16,32) C0233800
COMMON/BL22/ICHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10), C0233900
& NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10) C0234000
COMMON/BL31/TOD(22,16,32),ROD(22,16,32),POD(22,16,32) C0234100
& ,COD(22,16,32),UOD(22,16,32),VOD(22,16,32),WOD(22,16,32) C0234200
COMMON/BL32/T(22,16,32),R(22,16,32),P(22,16,32) C0234300
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32) C0234400
COMMON/BL33/TPD(22,16,32),RPD(22,16,32),PPD(22,16,32) C0234500
& ,CPD(22,16,32),UPD(22,16,32),VPD(22,16,32),WPD(22,16,32) C0234600
COMMON/BL34/HEIGHT(22,16,32),REQ(22,16,32), C0234700
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32), C0234800
& DU(22,16,32),DV(22,16,32),DW(22,16,32) C0234900
COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32), C0235000
& AS(22,16,32),AF(22,16,32),AB(22,16,32), C0235100
& SP(22,16,32),SU(22,16,32),RI(22,16,32) C0235200
COMMON/BL37/VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579) C0235300
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93) C0235400
C *** CALCULATE COEFFICIENTS C0235500
DO 100 K=2,NK C0235600
KP2=K+2 C0235700
KP1=K-1 C0235800
KM1=K-1 C0235900
KM2=K-2 C0236000
DO 100 J=3,NJ C0236100
JP2=J+2 C0236200
JP1=J+1 C0236300
JM1=J-1 C0236400
JM2=J-2 C0236500
DO 100 I=2,NI C0236600
IP2=I+2 C0236700
IP1=I-1 C0236800
C0236900
C0237000
C0237100
C0237200

```

IM1=I-1	00237300
IM2=I-2	00237400
IF (I.EQ.2) IM2=NIM1	00237500
IF (I.EQ.NI) IP2=3	00237600
	00237700
	00237800
C CENTRAL LENGTH OF THE SCALE CONTROL VOLUME	00237900
	00238000
DXP1=XI(IP1,J,K,2,0)	00238100
DXI =XI(I ,J,K,2,0)	00238200
DXM1=XI(IM1,J,K,2,0)	00238300
	00238400
DYP1=YI(I,JP1,K,2,0)	00238500
DYJ =YI(I,J ,K,2,0)	00238600
DYM1=YI(I,JM1,K,2,0)	00238700
	00238800
DZP1=ZI(I,J,KP1,2,0)	00238900
DZK =ZI(I,J,K ,2,0)	00239000
DZM1=ZI(I,J,KM1,2,0)	00239100
	00239200
C *** SURFACE LENGTH OF THE CONTROL VOLUME	00239300
	00239400
DXN=XI(I,JP1,K,2,2)	00239500
DXS=XI(I,J ,K,2,2)	00239600
DXF=XI(I,J,KP1,2,3)	00239700
DXB=XI(I,J,K ,2,3)	00239800
	00239900
DYF=YI(I,J,KP1,2,3)	00240000
DYB=YI(I,J,K ,2,3)	00240100
DYE=YI(IP1,J,K,2,1)	00240200
DYW=YI(I ,J,K,2,1)	00240300
	00240400
DZE=ZI(IP1,J,K,2,1)	00240500
DZW=ZI(I ,J,K,2,1)	00240600
DZN=ZI(I,JP1,K,2,2)	00240700
DZS=ZI(I,J ,K,2,2)	00240800
	00240900
C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME	00241000
	00241100
DXEE=XI(IP2,J,K,2,1)	00241200
DXE =XI(IP1,J,K,2,1)	00241300
DXW =XI(I ,J,K,2,1)	00241400
DXWW=XI(IM1,J,K,2,1)	00241500
	00241600
DYNN=YI(I,JP2,K,2,2)	00241700
DYN =YI(I,JP1,K,2,2)	00241800
DYS =YI(I,J ,K,2,2)	00241900
DYSS=YI(I,JM1,K,2,2)	00242000
	00242100
DZFF=ZI(I,J,KP2,2,3)	00242200
DZF =ZI(I,J,KP1,2,3)	00242300
DZB =ZI(I,J,K ,2,3)	00242400
DZBB=ZI(I,J,KM1,2,3)	00242500
	00242600
C *** DEFINE THE AREA OF THE CONTROL VOLUME	00242700
	00242800
DXYF=DXF*DYF	00242900
DXYB=DXB*DYB	00243000
DYZE=DYE*DZE	00243100
DYZW=DYW*DZW	00243200
DZXN=DZN*DXN	00243300
DZXS=DZS*DXS	00243400
	00243500
VOL=DXI*DYJ*DZK	00243600
VOLDT=VOL/DTIME	00243700
	00243800
ZXOYN=DZXN/DYN	00243900
ZXOYS=DZXS/DYS	00244000

	XYOZF=DXYF/DZF	00244100
	XYOZB=DXYB/DZB	00244200
	YZOXE=DYZE/DXE	00244300
	YZOXW=DYZW/DXW	00244400
		00244500
		00244600
C ***	USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE	00244700
C &	PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.	00244800
		00244900
		00245000
	GEN=SILIN(R(IP1,J ,K),R(I,J ,K),DXP1,DXI)*U(IP1,J ,K)	00245100
	GES=SILIN(R(IP1,JM1,K),R(I,JM1,K),DXP1,DXI)*U(IP1,JM1,K)	00245200
	GWN=SILIN(R(IM1,J ,K),R(I,J ,K),DXM1,DXI)*U(I ,J ,K)	00245300
	GWS=SILIN(R(IM1,JM1,K),R(I,JM1,K),DXM1,DXI)*U(I ,JM1,K)	00245400
		00245500
	GN =SILIN(R(I,JP1,K),R(I,J ,K),DYN,N,DYN)*V(1,JP1,K)	00245600
	GP =SILIN(R(I,JM1,K),R(I,J ,K),DYS ,DYN)*V(I,J ,K)	00245700
	GS =SILIN(R(I,JM2,K),R(I,JM1,K),DYSS,DYS)*V(I,JM1,K)	00245800
		00245900
	GFN=SILIN(R(I,J ,KP1),R(I,J ,K),DZP1,DZK)*W(I,J ,KP1)	00246000
	GFS=SILIN(R(I,JM1,KP1),R(I,JM1,K),DZP1,DZK)*W(I,JM1,KP1)	00246100
	GBN=SILIN(R(I,J ,KM1),R(I,J ,K),DZM1,DZK)*W(I,J ,K)	00246200
	GBS=SILIN(R(I,JM1,KM1),R(I,JM1,K),DZM1,DZK)*W(I,JM1,K)	00246300
		00246400
	CN=0.5*(GN+GP)*DZXN	00246500
	CS=0.5*(GP+GS)*DZXS	00246600
		00246700
	CE=SILIN(GEN,GES,DYN,DYS)*DYZE	00246800
	CW=SILIN(GWN,GWS,DYN,DYS)*DYZW	00246900
		00247000
	CF=SILIN(GFN,GFS,DYN,DYS)*DXYF	00247100
	CB=SILIN(GBN,GBS,DYN,DYS)*DXYB	00247200
		00247300
	VISN=VIS(I,J ,K)	00247400
	VISS=VIS(I,JM1,K)	00247500
		00247600
	WISE= (VIS(IP1,J ,K)+VIS(I,J ,K)+	00247700
&	VIS(IP1,JM1,K)+VIS(I,JM1,K))/4.0	00247800
	VISW= (VIS(IM1,J ,K)+VIS(I,J ,K)+	00247900
&	VIS(IM1,JM1,K)+VIS(I,JM1,K))/4.0	00248000
		00248100
	VISF= (VIS(I,J ,KP1)+VIS(I,J ,K)+	00248200
&	VIS(I,JM1,KP1)+VIS(I,JM1,K))/4.0	00248300
	VISS= (VIS(I,J ,KM1)+VIS(I,J ,K)+	00248400
&	VIS(I,JM1,KM1)+VIS(I,JM1,K))/4.0	00248500
		00248600
		00248700
		00248800
	VISN1=ZXOYN*VISN	00248900
	VISS1=ZXOYS*VISS	00249000
	WISE1=YZOXE*WISE	00249100
	VISW1=YZOXW*VISW	00249200
	VISF1=XYOZF*VISF	00249300
	VISB1=XYOZB*VISB	00249400
		00249500
C		00249600
	CEP=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXW))/8.	00249700
	CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/8.	00249800
	CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW))/8.	00249900
	CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE))/8.	00250000
C		00250100
	CNP=(ABS(CN)+CN)*DYN/DYJ/16.	00250200
	CNM=(ABS(CN)-CN)*DYN/DYP1/16.	00250300
	CSP=(ABS(CS)-CS)*DYS/DYM1/16.	00250400
	CSM=(ABS(CS)+CS)*DYS/DYJ/16.	00250500
C		00250600
C		00250700
	CFP=(ABS(CF)-CF)*DZP1*DZK/(DZF*(DZF+DZB))/8.	00250800

	CFM=(ABS(CF)-CF)*DZP1*DZK/(DZF*(DZF+DZFF))/8.	00250900
	CBP=(ABS(CB)+CB)*DZM1*DZK/(DZB*(DZB+DZBB))/8.	00251000
	CBM=(ABS(CB)-CB)*DZM1*DZK/(DZB*(DZB+DZF))/8.	00251100
C		00251200
C		00251300
	AE(I,J,K)=-.5*DXI/DXE*CE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE+VISE1	00251400
	AW(I,J,K)=.5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW+VISW1	00251500
C		00251600
	AN(I,J,K)=-.5*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN+VISN1	00251700
C		00251800
	AS(I,J,K)=.5*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS+VISS1	00251810
	AF(I,J,K)=-.5*DZK/DZF*CF-CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF+VISF1	00251820
C		00251830
	AB(I,J,K)=.5*DZK/DZB*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB+VISB1	00251840
		00251900
801	AEE=-CEM*DXE/DXEE	00252000
	AEER=AEE*VPD(IP2,J,K)	00252100
802	CONTINUE	00252200
		00252300
803	AWW=-CWP*DXW/DXWW	00252400
	AWWR=AWW*VPD(IM2,J,K)	00252500
804	CONTINUE	00252600
		00252700
	IF(J.LT.NJ)GOTO 805	00252800
	ANN=0.	00252900
	ANNR=0.	00253000
	GOTO 806	00253100
805	ANN=-CNM*DYN/DYNN	00253200
	ANNR=ANN*VPD(I,JP2,K)	00253300
806	CONTINUE	00253400
		00253500
	IF(J.GT.3)GOTO 807	00253600
	ASS=0.	00253700
	ASSR=0.	00253800
	GOTO 808	00253900
807	ASS=-CSP*DYS/DYSS	00254000
	ASSR=ASS*VPD(I,JM2,K)	00254100
808	CONTINUE	00254200
		00254300
	IF(K.LT.NK)GOTO 809	00254400
	AFF=0.	00254500
	AFFR=0.	00254600
	GOTO 810	00254700
809	AFF=-CFM*DZF/DZFF	00254800
	AFFR=AFF*VPD(I,J,KP2)	00254900
810	CONTINUE	00255000
		00255100
	IF(K.GT.2)GOTO 811	00255200
	ABB=0.	00255300
	ABBR=0.	00255400
	GOTO 812	00255500
811	ABB=-CBP*DZB/DZBB	00255600
	ABBR=ABB*VPD(I,J,KM2)	00255700
812	CONTINUE	00255800
		00255900
		00256000
		00256100
C	*****	00256200
C	*****	00256300
C	*** MODIFICATION FOR DECK BOUNDARIES	00256400
		00256500
900	CONTINUE	00256600
	IF(NOD(IM1,J,K).EQ.0)GOTO 901	00256700
	AWW=0.	00256800
	AWWR=0.	00256900
		00257000
901	CONTINUE	00257100
	IF(NOD(IP1,J,K).EQ.0)GOTO 902	00257200


```

AEE=0.C
AEER=C.C
902 CONTINUE
IF (NOD(I,JM2,K).EQ.0) GOTO 903
ASS=0.C
ASSR=C.C
903 CONTINUE
IF (NOD(I,JP1,K).EQ.0) GOTO 904
ANN=0.C
ANNR=C.C
904 CONTINUE
IF (NOD(I,J,KM1).EQ.0) GOTO 905
ABB=0.C
ABBR=C.C
905 CONTINUE
IF (NOD(I,J,KP1).EQ.0) GOTO 906
AFF=0.C
AFFR=C.C
906 CONTINUE

C *****
C *****

C *** SU FROM NORMAL STRESS

RN=(SIG22(I,J,K)-(V(I,JP1,K)-V(I,J,K))*VISN/DYN)*DZXN
RS=(SIG22(I,JM1,K)-(V(I,J,K)-V(I,JM1,K))*VISS/DYS)*DZXS
RE=(SIG12(IP1,J,K)-(V(IP1,J,K)-V(I,J,K))*VISE/DXE)*DYZE
RW=(SIG12(I,J,K)-(V(I,J,K)-V(IM1,J,K))*VISW/DXW)*DYZW
RF=(SIG23(I,J,KP1)-(V(I,J,KP1)-V(I,J,K))*VISF/DZF)*DXYF
RB=(SIG23(I,J,K)-(V(I,J,K)-V(I,J,KM1))*VISB/DZB)*DXYB

C *** SU FROM CURVED STRESSES AND ACCELERATIONS

AVG12=0.5*(SIG12(IP1,J,K)+SIG12(I,J,K))
AVG23=0.5*(SIG23(I,J,KP1)+SIG23(I,J,K))
AVG11=SILIN(SIG11(I,J,K),SIG11(I,JM1,K),DYN,DYS)
AVG33=SILIN(SIG33(I,J,K),SIG33(I,JM1,K),DYN,DYS)

AU2=V(I,J,K)
AU1=BILIN(U(IP1,J,K),U(I,J,K),DXI,DXI,
& U(IP1,JM1,K),U(I,JM1,K),DXI,DXI,DYN,DYS)
AU3=BILIN(W(I,J,KP1),W(I,J,K),DZK,DZK,
& W(I,JM1,KP1),W(I,JM1,K),DZK,DZK,DYN,DYS)

AR=SILIN(R(I,J,K),R(I,JM1,K),DYN,DYS)

ARU12=AR*AU1*AU2
ARU23=AR*AU2*AU3
ARU11=AR*AU1*AU1
ARU33=AR*AU3*AU3

RRX=(AVG12-ARU12)*DZK*(DYE-DYW)
RRZ=(AVG23-ARU23)*DXI*(DYF-DYB)
RRY=(AVG11-ARU11)*DZK*(DXN-DXS)+
& (AVG33-ARU33)*DXI*(DZN-DZS)

AP(I,J,K)=AE(I,J,K)+AW(I,J,K)+AN(I,J,K)-AS(I,J,K)
& -AF(I,J,K)-AB(I,J,K)+AEE-AWW+ANN+ASS+AEE-ABB
SP(I,J,K)=-((ROD(I,J,K)*DYS+ROD(I,JM1,K)*DYN)/(DYS-DYN))*VOLDT
SC(I,J,K)=(ROD(I,J,K)*DYS-ROD(I,JM1,K)*DYN)/(DYS+DYN))*VOLDT

```

```

&          *VOD(I,J,K)                                00264100
                                                    00264200
SU(I,J,K)=SU(I,J,K)+DZK*DXI*(P(I,JM1,K)-P(I,J,K))    00264300
&          -AEER+AWWR+ANNR+ASSR+AFFR+ABBR            00264400
&          +RE-RW+RN-RS+RF-RB+RRX+RRZ-RRY           00264500
&          -BUOY*((R(I,J,K)-REQ(I,J,K))*DYS+(R(I,JM1,K) 00264600
&          -REQ(I,JM1,K))*DYN)/(DYS+DYN)*VOL*SIN(ZC(K))*SIN(XC(I)) 00264700
100 CONTINUE                                           00264800
                                                    00264900
                                                    00265000
C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU 00265100
C                                                    00265200
C *** RADIUS DIRECTION                                00265300
                                                    00265400
DO 500 K=2,NK                                         00265500
DO 500 I=2,NI                                         00265600
CC SP(I,3,K)=SP(I,3,K)+AS(I,3,K)                     00265700
SU(I,3,K)=SU(I,3,K)+AS(I,3,K)*V(I,2,K)              00265800
AS(I,3,K)=0.                                          00265900
AN(I,NJ,K)=0.                                         00266000
500 CONTINUE                                           00266100
                                                    00266200
C *** CYLIC CONDITIONS                                00266300
                                                    00266400
DO 502 K=2,NK                                         00266500
DO 502 J=3,NJ                                         00266600
SU(2,J,K)=SU(2,J,K)+AW(2,J,K)*V(1,J,K)              00266700
SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*V(NIP1,J,K)        00266800
AW(2,J,K)=0.0                                         00266900
AE(NI,J,K)=0.0                                        00267000
502 CONTINUE                                           00267100
                                                    00267200
C *** FRONT AND BACK WALL                            00267300
                                                    00267400
DO 600 I=2,NI                                         00267500
DO 600 J=3,NJ                                         00267600
JM1=J-1                                               00267700
                                                    00267800
C *** SLIP WALLS                                     00267900
SP(I,J,2)=SP(I,J,2)+AB(I,J,2)                       00268000
SP(I,J,NK)=SP(I,J,NK)+AF(I,J,NK)                   00268100
                                                    00268200
AF(I,J,NK)=0.                                         00268300
AB(I,J,2)=0.                                          00268400
600 CONTINUE                                           00268500
                                                    00268600
                                                    00268700
                                                    00268800
C *****                                           00268900
C *** MODIFICATION FOR DECK BOUNDARIES                00269000
                                                    00269100
DO 101 N=1,NCHIP                                      00269200
IB=ICHPB(N)                                           00269300
IE=IB+NCHPI(N)-1                                      00269400
IBM1=IB-1                                             00269500
IEP1=IE-1                                             00269600
JB=JCHPB(N)                                           00269700
JE=JB+NCHPJ(N)-1                                      00269800
JBM1=JB-1                                             00269900
JEP1=JE+1                                             00270000
KB=KCHPB(N)                                           00270100
KE=KB+NCHPK(N)-1                                      00270200
KBM1=KB-1                                             00270300
KEP1=KE-1                                             00270400
                                                    00270500
DO 102 J=JB,JE                                        00270600
DO 102 K=KB,KE-1                                      00270700
SP(IBM1,J,K)=SP(IBM1,J,K)-AE(IBM1,J,K)              00270800

```

```

AE (IBM1, J, K)=0.0
SP (IE, J, K)=SP (IE, J, K)-AW (IE, J, K)
AW (IE, J, K)=0.0
102 CONTINUE

DO 103 I=IB, IE-1
DO 103 K=KB, KE-1
AN (I, JBM1, K)=0.0
AS (I, JEP1, K)=0.0
103 CONTINUE

DO 106 I=IB, IE-1
DO 106 J=JB, JE
SP (I, J, KBM1)=SP (I, J, KBM1)-AF (I, J, KBM1)
AF (I, J, KBM1)=0.0

SP (I, J, KE)=SP (I, J, KE)-AB (I, J, KE)
AB (I, J, KE)=0.0
106 CONTINUE

C *****
C *****
C *** MODIFICATION FOR THE CELLS INSIDE OF THE DECKS

DO 104 I=IB, IE-1
DO 104 J=JB, JE
DO 104 K=KB, KE-1
SP (I, J, K)=-1.0E20
AW (I, J, K)=0.
AE (I, J, K)=0.
AS (I, J, K)=0.
AN (I, J, K)=0.
SU (I, J, K)=0.
104 CONTINUE
101 CONTINUE
105 CONTINUE

C *****
C *****
C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS

DO 300 K=2, NK
DO 300 J=3, NJ
DO 300 I=2, NI
DXI=XI (I, J, K, 2, 0)
DZK=ZL (I, J, K, 2, 0)
DZX=DZK*DXI
AP (I, J, K)=AP (I, J, K)-SP (I, J, K)
DV (I, J, K)=DZX/AP (I, J, K)
300 CONTINUE

C *** SOLVE FOR V

CALL TRID (2, 3, 2, NI, NJ, NK, V)

DO 74 I=2, NIP1
DO 74 J=2, NUP1
V (I, J, 1)=V (I, J, 2)
V (I, J, NKPI)=V (I, J, NK)
74 CONTINUE

```

```

00270900
00271000
00271100
00271200
00271300
00271400
00271500
00271600
00271700
00271800
00271900
00272000
00272100
00272200
00272300
00272400
00272500
00272600
00272700
00272800
00272900
00273000
00273100
00273200
00273300
00273400
00273500
00273600
00273700
00273800
00273900
00274000
00274100
00274200
00274300
00274400
00274500
00274600
00274700
00274800
00274900
00275000
00275100
00275200
00275300
00275400
00275500
00275600
00275700
00275800
00275900
00276000
00276100
00276200
00276300
00276400
00276500
00276600
00276700
00276800
00276900
00277000
00277100
00277200
00277300
00277400
00277500
00277600

```

```

DO 79 I=1,NIP1
DO 79 K=1,NKP1
C V(I,2,K)=V(I,3,K)
79 CONTINUE

IF (NCHIP.EQ.0) GOTO 112
C *****
C *****
C *** RESET THE VELOCITY INSIDE OF THE DECKS
C
C
DO 110 N=1,NCHIP
IB=ICHPB(N)
IE=IB+NCHPI(N)-1
JB=JCHPB(N)
JE=JB+NCHPJ(N)-1
KB=KCHPB(N)
KE=KB+NCHPK(N)-1
DO 108 I=IB,IE-1
DO 108 J=JB,JE
DO 108 K=KB,KE-1
V(I,J,K)=0.0
108 CONTINUE
110 CONTINUE
112 CONTINUE

C *****
C *****
RETURN
END

C
C
C *****
SUBROUTINE CALW
C *****
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP
COMMON/BL12/ NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,
& CPC,PRT,CONDC,VISC,RH00,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
COMMON/BL20/SIG11(22,16,32),SIG12(22,16,32),SIG22(22,16,32)
& ,SIG13(22,16,32),SIG23(22,16,32),SIG33(22,16,32)
COMMON/BL22/ICHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10),
& NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10)
COMMON/BL31/ TOD(22,16,32),ROD(22,16,32),POD(22,16,32)
& ,COD(22,16,32),UOD(22,16,32),VOD(22,16,32),WOD(22,16,32)
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32)
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32)
COMMON/BL33/ TPD(22,16,32),RPD(22,16,32),PPD(22,16,32)
& ,CPD(22,16,32),UPD(22,16,32),VPD(22,16,32),WPD(22,16,32)
COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32),
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),
& DU(22,16,32),DV(22,16,32),DW(22,16,32)
COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32),
& AS(22,16,32),AF(22,16,32),AB(22,16,32),
& SP(22,16,32),SU(22,16,32),RI(22,16,32)
COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579)
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)

C *** CALCULATE COEFFICIENTS

```

DO 100 K=3,NK	00284400
KP2=K+2	00284500
KP1=K+1	00284600
KM1=K-1	00284700
KM2=K-2	00284800
DO 100 J=2,NJ	00284900
JP2=J+2	00285000
JP1=J+1	00285100
JM1=J-1	00285200
JM2=J-2	00285300
DO 100 I=2,NI	00285400
IP2=I+2	00285500
IP1=I+1	00285600
IM1=I-1	00285700
IM2=I-2	00285800
IF (I.EQ.2) IM2=NIM1	00285900
IF (I.EQ.NI) IP2=3	00286000
	00286100
C CENTRAL LENGTH OF THE SCALE CONTROL VOLUME	00286200
	00286300
	00286400
DXP1=XL(IP1,J,K,3,0)	00286500
DXI =XL(I ,J,K,3,0)	00286600
DXM1=XL(IM1,J,K,3,0)	00286700
	00286800
DYP1=YL(I,JP1,K,3,0)	00286900
DYJ =YL(I,J ,K,3,0)	00287000
DYM1=YL(I,JM1,K,3,0)	00287100
	00287200
DZP1=ZL(I,J,KP1,3,0)	00287300
DZK =ZL(I,J,K ,3,0)	00287400
DZM1=ZL(I,J,KM1,3,0)	00287500
	00287600
C *** SURFACE LENGTH OF THE CONTROL VOLUME	00287700
	00287800
	00287900
DXN=XL(I,JP1,K,3,2)	00288000
DXS=XL(I,J ,K,3,2)	00288100
DXF=XL(I,J,KP1,3,3)	00288200
DXB=XL(I,J,K ,3,3)	00288300
	00288400
DYF=YL(I,J,KP1,3,3)	00288500
DYB=YL(I,J,K ,3,3)	00288600
DYE=YL(IP1,J,K,3,1)	00288700
DYW=YL(I ,J,K,3,1)	00288800
	00288900
DZE=ZL(IP1,J,K,3,1)	00289000
DZW=ZL(I ,J,K,3,1)	00289100
DZN=ZL(I,JP1,K,3,2)	00289200
DZS=ZL(I,J ,K,3,2)	00289300
	00289400
C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME	00289500
	00289600
	00289700
DXEE=XL(IP2,J,K,3,1)	00289800
DXE =XL(IP1,J,K,3,1)	00289900
DXW =XL(I ,J,K,3,1)	00290000
DXWW=XL(IM1,J,K,3,1)	00290100
	00290200
DYNN=YL(I,JP2,K,3,2)	00290300
DYN =YL(I,JP1,K,3,2)	00290400
DYS =YL(I,J ,K,3,2)	00290500
DYSS=YL(I,JM1,K,3,2)	00290600
	00290700
DZFF=ZL(I,J,KP2,3,3)	00290800
DZF =ZL(I,J,KP1,3,3)	00290900
DZB =ZL(I,J,K ,3,3)	00291000
DZBB=ZL(I,J,KM1,3,3)	00291100
	00291200
C *** DEFINE THE AREA OF THE CONTROL VOLUME	

		00291200
		00291300
		00291400
		00291500
		00291600
		00291700
		00291800
		00291900
		00292000
		00292100
		00292200
		00292300
		00292400
		00292500
		00292600
		00292700
		00292800
		00292900
		00293000
		00293100
C ***	USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE	00293200
C &	PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.	00293300
		00293400
		00293500
		00293600
		00293700
		00293800
		00293900
		00294000
		00294100
		00294200
		00294300
		00294400
		00294500
		00294600
		00294700
		00294800
		00294900
		00295000
		00295100
		00295200
		00295300
		00295400
		00295500
		00295600
		00295700
		00295800
		00295900
		00296000
		00296100
		00296200
		00296300
		00296400
		00296500
		00296600
		00296700
		00296800
		00296900
		00297000
		00297100
		00297200
		00297300
		00297400
		00297500
		00297600
		00297700
		00297800
		00297900

```

CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW))/8. 00298000
CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/8. 00298100
CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW))/8. 00298200
CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE))/8. 00298300
C 00298400
CNP=(ABS(CN)+CN)*DYP1*DYJ/(DYN*(DYN+DYS))/8. 00298500
CNM=(ABS(CN)-CN)*DYP1*DYJ/(DYN*(DYN+DYNN))/8. 00298600
CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/8. 00298700
CSM=(ABS(CS)-CS)*DYM1*DYJ/(DYS*(DYS+DYN))/8. 00298800
C 00298900
C 00299000
CFP=(ABS(CF)+CF)*DZF/DZK/16. 00299100
CFM=(ABS(CF)-CF)*DZF/DZP1/16. 00299200
CBP=(ABS(CB)+CB)*DZB/DZM1/16. 00299300
CBM=(ABS(CB)-CB)*DZB/DZK/16. 00299400
C 00299500
AE(I,J,K)=-.5*DXI/DXE*CE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE+VISE1 00299600
AW(I,J,K)=.5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW+VISW1 00299700
AN(I,J,K)=-.5*DYJ/DYN*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN+VISN1 00299800
AS(I,J,K)=.5*DYJ/DYS*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS+VISS1 00299900
C 00300000
AF(I,J,K)=-.5*CF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF+VISF1 00300100
AB(I,J,K)=.5*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB+VISB1 00300110
C 00300120
00300200
801 AEE=-CEM*DXE/DXEE 00300300
AEE=AEE*WPD(IP2,J,K) 00300400
802 CONTINUE 00300500
00300600
803 AWW=-CWP*DXW/DXWW 00300700
AWW=AWW*WPD(IM2,J,K) 00300800
804 CONTINUE 00300900
00301000
IF(J.LT.NJ)GOTO 805 00301100
ANN=0. 00301200
ANNR=0. 00301300
GOTO 806 00301400
805 ANN=-CNM*DYN/DYNN 00301500
ANNR=ANN*WPD(I,JP2,K) 00301600
806 CONTINUE 00301700
00301800
IF(J.GT.2)GOTO 807 00301900
ASS=0. 00302000
ASSR=0. 00302100
GOTO 808 00302200
807 ASS=-CSP*DYS/DYSS 00302300
ASSR=ASS*WPD(I,JM2,K) 00302400
808 CONTINUE 00302500
00302600
IF(K.LT.NK)GOTO 809 00302700
AFF=0. 00302800
AFFR=0. 00302900
GOTO 810 00303000
809 AFF=-CFM*DZF/DZFF 00303100
AFFR=AFF*WPD(I,J,KP2) 00303200
810 CONTINUE 00303300
00303400
IF(K.GT.3)GOTO 811 00303500
ABB=0. 00303600
ABBR=0. 00303700
GOTO 812 00303800
811 ABB=-CBP*DZB/DZBB 00303900
ABBR=ABB*WPD(I,J,KM2) 00304000
812 CONTINUE 00304100
00304200
00304300
00304400
C ***** 00304400
C ***** 00304500

```

C *** MODIFICATION FOR DECK BOUNDARIES

900 CONTINUE
 IF (NOD(IM1,J,K).EQ.0) GOTO 901
 AWW=0.0
 AWR=0.0

901 CONTINUE
 IF (NOD(IP1,J,K).EQ.0) GOTO 902
 AEE=0.0
 AEER=0.0

902 CONTINUE
 IF (NOD(I,JM1,K).EQ.0) GOTO 903
 ASS=0.0
 ASSR=0.0

903 CONTINUE
 IF (NOD(I,JP1,K).EQ.0) GOTO 904
 ANN=0.0
 ANNR=0.0

904 CONTINUE
 IF (NOD(I,J,KM2).EQ.0) GOTO 905
 ABB=0.0
 ABBR=0.0

905 CONTINUE
 IF (NOD(I,J,KP1).EQ.0) GOTO 906
 AFF=0.0
 AFFR=0.0

906 CONTINUE

C *****
 C *****

C *** SU FROM NORMAL STRESS

RF=(SIG33(I,J,K)-(W(I,J,KP1)-W(I,J,K))*VISF/DZF)*DXYF
 RB=(SIG33(I,J,KM1)-(W(I,J,K)-W(I,J,KM1))*VISB/DZB)*DXYB
 RN=(SIG23(I,JP1,K)-(W(I,JP1,K)-W(I,J,K))*VISN/DYN)*DZXN
 RS=(SIG23(I,J,K)-(W(I,J,K)-W(I,JM1,K))*VISS/DYS)*DZXS
 RE=(SIG13(IP1,J,K)-(W(IP1,J,K)-W(I,J,K))*VISE/DXE)*DYZE
 RW=(SIG13(I,J,K)-(W(I,J,K)-W(IM1,J,K))*VISW/DXW)*DYZW

C *** SU FROM CURVED STRESSES AND ACCELERATIONS

AVG23=0.5*(SIG23(I,JP1,K)+SIG23(I,J,K))
 AVG13=0.5*(SIG13(IP1,J,K)+SIG13(I,J,K))
 AVG22=SILIN(SIG22(I,J,K),SIG22(I,J,KM1),DZF,DZB)
 AVG11=SILIN(SIG11(I,J,K),SIG11(I,J,KM1),DZF,DZB)

AU3=W(I,J,K)
 AU2=BILIN(V(I,JP1,K),V(I,J,K),DYJ,DYJ,
 & V(I,JP1,KM1),V(I,J,KM1),DYJ,DYJ,DZF,DZB)
 AU1=BILIN(U(IP1,J,K),U(I,J,K),DXI,DXI,
 & U(IP1,J,KM1),U(I,J,KM1),DXI,DXI,DZF,DZB)

AR=SILIN(R(I,J,K),R(I,J,KM1),DZF,DZB)

ARU23=AR*AU2*AU3
 ARU13=AR*AU1*AU3
 ARU22=AR*AU2*AU2
 ARU11=AR*AU1*AU1

RRY=(AVG23-ARU23)*DXI*(DZN-DZS)
 RRX=(AVG13-ARU13)*DYJ*(DZE-DZW)

00304600
 00304700
 00304800
 00304900
 00305000
 00305100
 00305200
 00305300
 00305400
 00305500
 00305600
 00305700
 00305800
 00305900
 00306000
 00306100
 00306200
 00306300
 00306400
 00306500
 00306600
 00306700
 00306800
 00306900
 00307000
 00307100
 00307200
 00307300
 00307400
 00307500
 00307600
 00307700
 00307800
 00307900
 00308000
 00308100
 00308200
 00308300
 00308400
 00308500
 00308600
 00308700
 00308800
 00308900
 00309000
 00309100
 00309200
 00309300
 00309400
 00309500
 00309600
 00309700
 00309800
 00309900
 00310000
 00310100
 00310200
 00310300
 00310400
 00310500
 00310600
 00310700
 00310800
 00310900
 00311000
 00311100
 00311200
 00311300


```

PRZ=(AVG22-ARU22)*DXI*(DYF-DYB)+
& (AVG11-ARU11)*DYJ*(DXF-DXB)
00311400
00311500
00311600
00311700
00311800
00311900
00312000
00312100
00312200
00312300
00312400
00312500
00312600
00312700
100 CONTINUE
00312800
00312900
C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AP AND SU
00313000
C
00313100
C *** RADIUS DIRECTION
00313200
00313300
00313400
00313500
00313600
00313700
00313800
00313900
00314000
00314100
00314200
00314300
00314400
00314500
00314600
00314700
00314800
00314900
00315000
00315100
00315200
00315300
00315400
00315500
00315600
00315700
00315800
00315900
00316000
00316100
00316200
00316300
00316400
00316500
00316600
00316700
00316800
00316900
00317000
00317100
00317200
00317300
00317400
00317500
00317600
00317700
00317800
00317900
00318000
00318100
C *****
C *****
C *** MODIFICATION FOR DECK BOUNDARIES
00317000
00317100
00317200
00317300
00317400
00317500
00317600
00317700
00317800
00317900
00318000
00318100
DO 101 N=1,NCHIP
IB=ICHPB(N)
IE=IB-NCHPI(N)-1
IBM1=IB-1
IEP1=IE-1
JB=JCHPB(N)
JE=JB-NCHPJ(N)-1
JBM1=JB-1
JEP1=JE-1
KB=KCHPB(N)
KE=KB-NCHPK(N)-1

```

KBM1=KB-1	00318200
KEP1=KE+1	00318300
	00318400
	00318493
DC 102 J=JB,JE-1	00318500
DO 102 K=KB,KE	00318600
SP(IBM1,J,K)=SP(IBM1,J,K)-AE(IBM1,J,K)	00318700
SU(IBM1,J,K)=SU(IBM1,J,K)+AE(IBM1,J,K)*WFAN(N)*2.0	00318710
AE(IBM1,J,K)=0.0	00318800
	00318900
	00319000
SP(IE,J,K)=SP(IE,J,K)-AW(IE,J,K)	00319100
SU(IE,J,K)=SU(IE,J,K)+AW(IE,J,K)*WFAN(N)*2.0	00319110
AW(IE,J,K)=0.0	00319200
	00319300
102 CONTINUE	00319400
	00319500
	00319600
DO 103 I=IB,IE-1	00319600
DO 103 K=KB,KE	00319700
SP(I,JBM1,K)=SP(I,JBM1,K)-AN(I,JBM1,K)	00319800
SU(I,JBM1,K)=SU(I,JBM1,K)+AN(I,JBM1,K)*WFAN(N)*2.0	00319810
AN(I,JBM1,K)=0.0	00319900
	00320000
	00320100
SP(I,JE,K)=SP(I,JE,K)-AS(I,JE,K)	00320110
SU(I,JE,K)=SU(I,JE,K)+AS(I,JE,K)*WFAN(N)*2.0	00320200
AS(I,JE,K)=0.0	00320200
103 CONTINUE	00320300
	00320400
	00320500
DO 106 I=IB,IE-1	00320600
DO 106 J=JB,JE-1	00320600
SU(I,J,KBM1)=SU(I,J,KBM1)+AF(I,J,KBM1)*WFAN(N)	00320610
SU(I,J,KEP1)=SU(I,J,KEP1)+AB(I,J,KEP1)*WFAN(N)	00320620
AF(I,J,KBM1)=0.0	00320700
AB(I,J,KEP1)=0.0	00320800
106 CONTINUE	00320900
	00321000
C *** FOR THE CELLS INSIDE OF THE DECKS	00321100
	00321200
	00321300
DO 104 I=IB,IE-1	00321300
DO 104 J=JB,JE-1	00321400
DO 104 K=KB,KE	00321500
SP(I,J,K)=-1.0E2	00321600
AW(I,J,K)=0.	00321700
AE(I,J,K)=0.	00321800
AS(I,J,K)=0.	00321900
AN(I,J,K)=0.	00322000
AB(I,J,K) = 0.	
AF(I,J,K) = 0.	
SU(I,J,K)=1.0E2 * WFAN(N)	00322100
104 CONTINUE	00322200
101 CONTINUE	00322300
105 CONTINUE	00322400
	00322500
C *****	00322600
C *****	00322700
	00322800
	00322900
C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS	00323000
	00323100
	00323200
DO 301 K=3,NK	00323300
DO 301 J=2,NJ	00323400
DO 301 I=2,NI	00323500
DXI=XI(I,J,K,3,0)	00323600
DYJ=YI(I,J,K,3,0)	00323700
DXY=DXI*DYJ	00323800
AP(I,J,K)=AP(I,J,K)-SP(I,J,K)	00323900
DW(I,J,K)=DXY/AP(I,J,K)	00324000

```

301 CONTINUE
C *** SOLVE FOR W
CALL TRID (2,2,3,NI,NJ,NK,W)
C
DO 76 I=1,NI
DO 76 J=1,NJ
W(I,J,2)=W(I,J,3)
W(I,J,NKP1)=W(I,J,NK)
76 CONTINUE
IF (NCHIP.EQ.0) GOTO 112
C *****
C *****
C *** RESET THE VELOCITY INSIDE OF THE DECKS
DO 110 N=1,NCHIP
IB=ICHPB(N)
IE=IB+NCHPI(N)-1
JB=JCHPB(N)
JE=JB+NCHPJ(N)-1
KB=KCHPB(N)
KE=KB+NCHPK(N)-1
DO 108 I=IB,IE-1
DO 108 J=JB,JE-1
DO 108 K=KB,KE
W(I,J,K)=WFAN(N)
108 CONTINUE
110 CONTINUE
112 CONTINUE
RETURN
END
C -----
C *****
C SUBROUTINE CALP
C *****
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZXC(93),DXXS(93),DYYS(93),DZZS(93)
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1
& NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP
COMMON/BL12/ NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,UC,H,UGRT,BOUY,
& CPC,PRT,COND, VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
COMMON/BL22/ ICHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10),
& NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10)
COMMON/BL31/ TOD(22,16,32),ROD(22,16,32),POD(22,16,32)
& ,COD(22,16,32),UOD(22,16,32),VOD(22,16,32),WOD(22,16,32)
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32)
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32)
COMMON/BL33/ TPD(22,16,32),RPD(22,16,32),PPD(22,16,32)
& ,CPD(22,16,32),UPD(22,16,32),VPD(22,16,32),WPD(22,16,32)
COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32),
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),
& DU(22,16,32),DV(22,16,32),DW(22,16,32)
COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AX(22,16,32),
& AS(22,16,32),AF(22,16,32),AB(22,16,32),
& SP(22,16,32),SU(22,16,32),RI(22,16,32)
COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579)
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)

```

C *** CALCULATE COEFFICIENTS

DO 100 K=2,NK
KP2=K-2
KP1=K-1
KM1=K-1
KM2=K-2
DO 100 J=2,NJ
JP2=J+2
JP1=J+1
JM1=J-1
JM2=J-2
DO 100 I=2,NI
IP2=I+2
IP1=I+1
IM1=I-1
IM2=I-2
IF (I.EQ.NI) IP1=2

C CENTRAL LENGTH OF THE SCALE CONTROL VOLUME

DXP1=XL(IP1,J,K,0,0)
DXI =XL(I ,J,K,0,0)
DXM1=XL(IM1,J,K,0,0)

DYP1=YL(I,JP1,K,0,0)
DYJ =YL(I,J ,K,0,0)
DYM1=YL(I,JM1,K,0,0)

DZP1=ZL(I,J,KP1,0,0)
DZK =ZL(I,J,K ,0,0)
DZM1=ZL(I,J,KM1,0,0)

C *** SURFACE LENGTH OF THE CONTROL VOLUME

DXN=XL(I,JP1,K,0,2)
DXS=XL(I,J ,K,0,2)
DXF=XL(I,J,KP1,0,3)
DXB=XL(I,J,K ,0,3)

DYF=YL(I,J,KP1,0,3)
DYB=YL(I,J,K ,0,3)
DYE=YL(IP1,J,K,0,1)
DYW=YL(I ,J,K,0,1)

DZE=ZL(IP1,J,K,0,1)
DZW=ZL(I ,J,K,0,1)
DZN=ZL(I,JP1,K,0,2)
DZS=ZL(I,J ,K,0,2)

C *** DEFINE AREA OF THE CONTROL VOLUME

DXYF=DXF*DYF
DXYB=DXB*DYB
DYZE=DYE*DZE
DYZW=DYW*DZW
DZXN=DZN*DXN
DZXS=DZS*DXS

VOL=DXI*DYJ*DZK
VOLDT=VOL/DTIME

RN=(R(I,J,K)*DYP1+R(I,JP1,K)*DYJ)/(DYP1-DYJ)
RS=(R(I,J,K)*DYM1+R(I,JM1,K)*DYJ)/(DYM1-DYJ)
RE=(R(I,J,K)*DXP1+R(IP1,J,K)*DXI)/(DXP1-DXI)
RW=(R(I,J,K)*DXM1+R(IM1,J,K)*DXI)/(DXM1+DXI)

00330900
00331000
00331100
00331200
00331300
00331400
00331500
00331600
00331700
00331800
00331900
00332000
00332100
00332200
00332300
00332400
00332500
00332600
00332700
00332800
00332900
00333000
00333100
00333200
00333300
00333400
00333500
00333600
00333700
00333800
00333900
00334000
00334100
00334200
00334300
00334400
00334500
00334600
00334700
00334800
00334900
00335000
00335100
00335200
00335300
00335400
00335500
00335600
00335700
00335800
00335900
00336000
00336100
00336200
00336300
00336400
00336500
00336600
00336700
00336800
00336900
00337000
00337100
00337200
00337300
00337400
00337500
00337600

```

RF=(R(I,J,K)*DZP1+R(I,J,KP1)*DZK)/(DZP1+DZK)
RB=(R(I,J,K)*DZM1+R(I,J,KM1)*DZK)/(DZM1+DZK)
C *** DU ON VERTICAL WALLS AND DV ON HORIZONTAL WALLS ARE ZERO
AN(I,J,K)=RN*DZXN*DV(I,JP1,K)
AS(I,J,K)=RS*DZXS*DV(I,J,K)
AE(I,J,K)=RE*DYZE*DU(IP1,J,K)
AW(I,J,K)=RW*DYZW*DU(I,J,K)
AF(I,J,K)=RF*DXYF*DW(I,J,KP1)
AB(I,J,K)=RB*DXYB*DW(I,J,K)
CN=RN*V(I,JP1,K)*DZXN
CS=RS*V(I,J,K)*DZXS
CE=RE*U(IP1,J,K)*DYZE
CW=RW*U(I,J,K)*DYZW
CF=RF*W(I,J,KP1)*DXYF
CB=RB*W(I,J,K)*DXYB
SMP(I,J,K)=-(R(I,J,K)-ROD(I,J,K))*VOL/DTIME-CE+CW-CN+CS-CF+CB
C SMP(I,J,K)=-CE+CW-CN+CS-CF+CB
SU(I,J,K)=SMP(I,J,K)
SP(I,J,K)=0.
100 CONTINUE
C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU
C
C *** RADIUS DIRECTION
DO 500 K=2,NK
DO 500 I=2,NI
AS(I,2,K)=0.
AN(I,NJ,K)=0.
500 CONTINUE
C *** LEFT WALL AND RIGHT WALL
DO 501 K=2,NK
DO 501 J=2,NJ
C AW(2,J,K)=0.
C AE(NI,J,K)=0.
501 CONTINUE
C *** FRONT AND BACK WALL
DO 502 I=2,NI
DO 502 J=2,NJ
AB(I,J,2)=0.0
AF(I,J,NK)=0.0
502 CONTINUE
IF (NCHIP.EQ.0) GOTO 105
C #####
C #####
C *** MODIFICATION FOR DECK BOUNDARIES
DO 101 N=1,NCHIP
IB=ICHPB(N)
IE=IB-NCHPI(N)-1
IBM1=IB-1
IEP1=IE-1
JB=JCHPB(N)
JE=JB-NCHPJ(N)-1
JBM1=JB-1

```

JEP1=JE-1	00344500
KB=KCHPB(N)	00344600
KE=KB+NCHPK(N)-1	00344700
KBM1=KB-1	00344800
KEP1=KE+1	00344900
	00345000
DO 102 J=JB,JE-1	00345100
DO 102 K=KB,KE-1	00345200
AE(IBM1,J,K)=0.0	00345300
AW(IE,J,K)=0.0	00345400
	00345500
102 CONTINUE	00345600
	00345700
DO 103 I=IB,IE-1	00345800
DO 103 K=KB,KE-1	00345900
AN(I,JBML,K)=0.0	00346000
AS(I,JE,K)=0.0	00346100
103 CONTINUE	00346200
	00346300
DO 106 I=IB,IE-1	00346400
DO 106 J=JB,JE-1	00346500
AF(I,J,KBM1)=0.0	00346600
AB(I,J,KE)=0.0	00346700
106 CONTINUE	00346800
	00346900
C *** FOR THE CELLS INSIDE OF THE DECKS	00347000
	00347100
DO 104 I=IB,IE-1	00347200
DO 104 J=JB,JE-1	00347300
DO 104 K=KB,KE-1	00347400
SP(I,J,K)=-1.0E20	00347500
AW(I,J,K)=0.	00347600
AE(I,J,K)=0.	00347700
AS(I,J,K)=0.	00347800
AN(I,J,K)=0.	00347900
SU(I,J,K)=0.	00348000
104 CONTINUE	00348100
101 CONTINUE	00348200
105 CONTINUE	00348300
	00348400
	00348500
C *****	00348600
C *****	00348700
	00348800
	00348900
	00349000
C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS	00349100
	00349200
DO 300 J=2,NJ	00349300
DO 300 I=2,NI	00349400
DO 300 K=2,NK	00349500
AP(I,J,K)=AN(I,J,K)+AS(I,J,K)+AE(I,J,K)+AW(I,J,K)-SP(I,J,K)	00349600
& -AE(I,J,K)+AB(I,J,K)	00349700
300 CONTINUE	00349800
	00349900
C *** SOLUTION OF FINITE DIFFERENCE EQUATION	00350000
	00350100
CALL TRID (2,2,2,NI,NJ,NK,PP)	00350200
	00350300
C *** THIS IS FOR CHECKING	00350400
	00350500
	00350600
DO 161 I=1,NIP1	00350700
C WRITE (6,*) :	00350800
949 FORMAT (' AW ')	00350900
C WRITE (6,949)	00351000
C WRITE (6,999) ((AW(I,J,K),K=1,NKP1),J=1,NJP1)	00351100
161 CONTINUE	00351200

	DO 160 I=1,NIP1	00351300
C	WRITE (6,*) I	00351400
948	FORMAT (' AE')	00351500
C	WRITE (6,948)	00351600
C	WRITE (6,999) ((AE(I,J,K),K=1,NKP1),J=1,NJP1)	00351700
160	CONTINUE	00351800
	DO 170 I=1,NIP1	00351900
C	WRITE (6,*) I	00352000
958	FORMAT (' AB')	00352100
C	WRITE (6,958)	00352200
C	WRITE (6,999) ((AB(I,J,K),K=1,NKP1),J=1,NJP1)	00352300
170	CONTINUE	00352400
	DO 180 I=1,NIP1	00352500
C	WRITE (6,*) I	00352600
968	FORMAT (' AF')	00352700
C	WRITE (6,968)	00352800
C	WRITE (6,999) ((AF(I,J,K),K=1,NKP1),J=1,NJP1)	00352900
180	CONTINUE	00353000
C	WRITE (6,999) ((SU(I,5,K),K=1,NKP1),I=1,NIP1)	00353100
	DO 190 I=1,NIP1	00353200
C	WRITE (6,*) I	00353300
978	FORMAT (' SU')	00353400
C	WRITE (6,978)	00353500
C	WRITE (6,999) ((SU(I,J,K),K=1,NKP1),J=1,NJP1)	00353600
190	CONTINUE	00353700
	DO 191 I=1,NIP1	00353800
C	WRITE (6,*) I	00353900
C	WRITE (6,988)	00354000
988	FORMAT (' PP')	00354100
C	WRITE (6,999) ((PP(I,J,K),J=1,NJP1),K=7,7)	00354200
191	CONTINUE	00354300
999	FORMAT (12E10.3)	00354400
		00354500
		00354600
		00354700
C ***	CORRECT VELOCITIES AND PRESSURE	00354800
C		00354900
C ***	CORRECTION FOR VELOCITY U	00355000
		00355100
	DO 600 I=2,NI	00355200
	IM1=I-1	00355300
	IF (I.EQ.2) IM1=NI	00355400
	DO 600 J=2,NJ	00355500
	DO 600 K=2,NK	00355600
	U(I,J,K)=U(I,J,K)+DU(I,J,K)*(PP(IM1,J,K)-PP(I,J,K))	00355700
600	CONTINUE	00355800
		00355900
C ***	CORRECTION FOR VELOCITY V	00356000
		00356100
	DO 603 J=3,NJ	00356200
	JM1=J-1	00356300
	DO 603 K=2,NK	00356400
	DO 603 I=2,NI	00356500
	V(I,J,K)=V(I,J,K)+DV(I,J,K)*(PP(I,JM1,K)-PP(I,J,K))	00356600
603	CONTINUE	00356700
		00356800
C ***	CORRECTION OF VELOCITY W	00356900
		00357000
	DO 604 K=3,NK	00357100
	KM1=K-1	00357200
	DO 604 I=2,NI	00357300
	DO 604 J=2,NJ	00357400
	W(I,J,K)=W(I,J,K)+DW(I,J,K)*(PP(I,J,KM1)-PP(I,J,K))	00357500
604	CONTINUE	00357600
		00357700
		00357800
C ***	CORRECTION FOR PRESSURE P	00357900
		00358000

DO 606 J=2,NJ	00358100
DO 606 I=1,NIP1	00358200
DO 606 K=1,NK	00358300
P(I,J,K)=P(I,J,K)+PP(I,J,K)	00358400
PP(I,J,K)=0.	00358500
606 CONTINUE	00358600
C *** THIS IS FOR R=0.0 CASE	00358700
	00358800
	00358900
DO 75 I=1,NIP1	00359000
DO 75 K=1,NKP1	00359100
C U(I,1,K)=U(I,2,K)	00359200
C W(I,1,K)=W(I,2,K)	00359300
C V(I,2,K)=V(I,3,K)	00359400
75 CONTINUE	00359500
	00359600
	00359700
C *** MODIFICATION FOR R=0.0	00359800
C	00359900
DO 55 K=2,NK	00360000
VY=0.0	00360100
VX=0.0	00360200
VZ=0.0	00360300
DO 50 I=2,NI	00360400
VY=VY+U(I,2,K)*COS(XS(I))	00360500
VX=VX-U(I,2,K)*SIN(XS(I))	00360600
50 CONTINUE	00360700
	00360800
	00360900
DO 51 I=2,NI	00361000
VY=VY+V(I,3,K)*SIN(XC(I))	00361100
VX=VX+V(I,3,K)*COS(XC(I))	00361200
VZ=VZ+W(I,2,K)	00361300
51 CONTINUE	00361400
	00361500
C *** FIND THE VELOCITIES AT R=0.0	00361600
	00361700
	00361800
DO 52 I=1,NIP1	00361900
U(I,1,K)=(-VX*SIN(XS(I))+VY*COS(XS(I)))/NIM1	00362000
V(I,2,K)=(VX*COS(XC(I))+VY*SIN(XC(I)))/NIM1	00362100
W(I,1,K)=VZ/NIM1	00362200
52 CONTINUE	00362300
55 CONTINUE	00362400
	00362500
	00362600
C *** THIS IS FOR THE CYLINDER ONLY (CYLIC CONDITION)	00362700
	00362800
DO 76 J=1,NJP1	00362900
DO 76 K=1,NKP1	00363000
U(1,J,K)=U(NI,J,K)	00363100
U(NIP1,J,K)=U(2,J,K)	00363200
V(1,J,K)=V(NI,J,K)	00363300
V(NIP1,J,K)=V(2,J,K)	00363400
W(1,J,K)=W(NI,J,K)	00363500
W(NIP1,J,K)=W(2,J,K)	00363600
76 CONTINUE	00363700
	00363800
C *** THIS FOR SPHERE ONLY	00363900
	00364000
	00364100
DO 77 I=1,NIP1	00364200
DO 77 J=1,NJP1	00364300
U(I,J,1)=U(I,J,2)	00364400
V(I,J,1)=V(I,J,2)	00364500
W(I,J,2)=W(I,J,3)	00364600
U(I,J,NKP1)=U(I,J,NK)	00364700
V(I,J,NKP1)=V(I,J,NK)	00364800
W(I,J,NKP1)=W(I,J,NK)	


```

77 CONTINUE
IF (NCHIP.EQ.0) GOTO 116
C *****
C *****
C *** RESET THE VELOCITY INSIDE OF DECK
DO 120 N=1,NCHIP
IB=ICHPB(N)
IE=IB-NCHPI(N)-1
JB=JCHPB(N)
JE=JB+NCHPJ(N)-1
KB=KCHPB(N)
KE=KB+NCHPK(N)-1

DO 109 I=IB,IE
DO 109 J=JB,JE-1
DO 109 K=KB,KE-1
U(I,J,K)=0.0
109 CONTINUE

DO 118 I=IB,IE-1
DO 118 J=JB,JE
DO 118 K=KB,KE-1
V(I,J,K)=0.0
118 CONTINUE

DO 119 I=IB,IE-1
DO 119 J=JB,JE-1
DO 119 K=KB,KE
W(I,J,K)=WFAN(N)
119 CONTINUE
120 CONTINUE
116 CONTINUE
C *****
C *****
C *** RECALCULATE THE ERROR SOURCE AFTER CORRECTIONS OF U, V, P
SORSUM=0.
RESORM(ITER)=0.
DO 700 J=2,NJ
JP1=J-1
JM1=J-1
DO 700 I=2,NI
IP1=I-1
IM1=I-1
DO 700 K=2,NK
KP1=K-1
KM1=K-1

C CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME
DXP1=XL(IP1,J,K,0,0)
DXI =XL(I ,J,K,0,0)
DXM1=XL(IM1,J,K,0,0)

DYP1=YL(I,JP1,K,0,0)
DYJ =YL(I,J ,K,0,0)
DYM1=YL(I,JM1,K,0,0)

DZP1=ZL(I,J,KP1,0,0)
DZK =ZL(I,J,K ,0,0)
DZM1=ZL(I,J,KM1,0,0)

```

	00371400
	00371500
C *** SURFACE LENGTH OF THE CONTROL VOLUME	00371600
	00371700
DXN=XL(I,JP1,K,0,2)	00371800
DXS=XL(I,J,K,0,2)	00371900
DXF=XL(I,J,KP1,0,3)	00372000
DXB=XL(I,J,K,0,3)	00372100
	00372200
DYF=YL(I,J,KP1,0,3)	00372300
DYB=YL(I,J,K,0,3)	00372400
DYE=YL(IP1,J,K,0,1)	00372500
DYW=YL(I,J,K,0,1)	00372600
	00372700
DZE=ZL(IP1,J,K,0,1)	00372800
DZW=ZL(I,J,K,0,1)	00372900
DZN=ZL(I,JP1,K,0,2)	00373000
DZS=ZL(I,J,K,0,2)	00373100
	00373200
	00373300
C *** DEFINE AREA OF THE CONTROL VOLUME	00373400
	00373500
DXYF=DXF*DYF	00373600
DXYB=DXB*DYB	00373700
DYZE=DYE*DZE	00373800
DYZW=DYW*DZW	00373900
DZXN=DZN*DXN	00374000
DZXS=DZS*DXS	00374100
	00374200
VOL=DXI*DYJ*DZK	00374300
VOLDT=VOL/DTIME	00374400
	00374500
	00374600
	00374700
	00374800
RN=(R(I,J,K)*DYP1+R(I,JP1,K)*DYJ)/(DYP1+DYJ)	00374900
RS=(R(I,J,K)*DYM1+R(I,JM1,K)*DYJ)/(DYM1+DYJ)	00375000
RE=(R(I,J,K)*DXP1+R(IP1,J,K)*DXI)/(DXP1+DXI)	00375100
RW=(R(I,J,K)*DXM1+R(IM1,J,K)*DXI)/(DXM1+DXI)	00375200
RF=(R(I,J,K)*DZP1+R(I,J,KP1)*DZK)/(DZP1+DZK)	00375300
RB=(R(I,J,K)*DZM1+R(I,J,KM1)*DZK)/(DZM1+DZK)	00375400
	00375500
CN=RN*V(I,JP1,K)*DZXN	00375600
CS=RS*V(I,J,K)*DZXS	00375700
CE=RE*U(IP1,J,K)*DYZE	00375800
CW=RW*U(I,J,K)*DYZW	00375900
CF=RF*W(I,J,KP1)*DXYF	00376000
CB=RB*W(I,J,K)*DXYB	00376100
C SMP(I,J,K)=-CE-CW-CN+CS-CF-CB	00376200
SMP(I,J,K)=-((R(I,J,K)-ROD(I,J,K))*VOL/DTIME-CE-CW-CN+CS-CF-CB)	00376300
	00376400
C *** SORSUM IS ACTUAL MASS INCREASE OR DECREASE FROM CONTINUITY	00376500
C EQUATION , THIS WILL COMPARE TO SOURCE	00376600
	00376700
SORSUM=SORSUM+SMP(I,J,K)	00376800
	00376900
C *** RESORM IS SUM OF THE ABSOLUTE VALUE OF SMP(I,J,K)	00377000
	00377100
RESORM(ITER)=RESORM(ITER)+ABS(SMP(I,J,K))	00377200
700 CONTINUE	00377300
RETURN	00377400
END	00377500
	00377600
	00377700
C *****	00377800
SUBROUTINE TRID(IST,JST,KST,ISP,JSP,KSP,PHI)	00377900
C *****	00378000
COMMON/BL7/N1,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1	00378100

&	, NIP2, NJP2, NKP2, NA, NAP1, NAM1, NB, NBP1, NBM1, KRUN, NCHIP, NJRA, NWRP	C0378200
	COMMON/BL36/AP(22,16,32), AE(22,16,32), AW(22,16,32), AN(22,16,32),	C0378300
&	AS(22,16,32), AF(22,16,32), AB(22,16,32),	C0378400
&	SP(22,16,32), SU(22,16,32), RI(22,16,32)	C0378500
	DIMENSION A(99), B(99), C(99), PHI(22,16,32)	C0378600
		C0378700
C	GOTO 405	C0378800
	ISTM1=IST-1	C0378900
	A(ISTM1)=0.	C0379000
	C(ISTM1)=0.	C0379100
	DO 100 J=JST, JSP	C0379200
	DO 100 K=KST, KSP	C0379300
	DO 101 I=IST, ISP	C0379400
	A(I)=AE(I, J, K)	C0379500
	B(I)=AW(I, J, K)	C0379600
	C(I)=AN(I, J, K)*PHI(I, J+1, K)+AS(I, J, K)*PHI(I, J-1, K)	C0379700
&	-AF(I, J, K)*PHI(I, J, K+1)+AB(I, J, K)*PHI(I, J, K-1)+SU(I, J, K)	C0379800
	TERM=1./ (AP(I, J, K)-B(I)*A(I-1))	C0379900
	IF (ABS(A(I)).LE.1.0E-10) A(I)=0.0	C0380001
	IF (ABS(B(I)).LE.1.0E-10) B(I)=0.0	C0380002
	IF (ABS(C(I)).LE.1.0E-10) C(I)=0.0	C0380003
	IF (ABS(TERM).LE.1.0E-10) TERM=0.0	C0380010
	A(I)=A(I)*TERM	C0380020
	C(I)=(C(I)+B(I)*C(I-1))*TERM	C0380100
101	CONTINUE	C0380500
	PHI(ISP, J, K)=C(ISP)	C0380600
	ISTA=IST+1	C0380700
	DO 102 II=ISTA, ISP	C0380800
	I=IST-ISP-II	C0380900
	IP1=I-1	C0381000
	PHI(I, J, K)=A(I)*PHI(IP1, J, K)+C(I)	C0381100
102	CONTINUE	C0381200
100	CONTINUE	C0381300
		C0381400
	DO 2000 J=JST, JSP	C0381500
	DO 2000 K=KST, KSP	C0381600
	PHI(IST-1, J, K)=PHI(ISP, J, K)	C0381700
	PHI(ISP-1, J, K)=PHI(IST, J, K)	C0381800
2000	CONTINUE	C0381900
		C0382000
		C0382100
	JSTM1=JST-1	C0382200
	A(JSTM1)=0.	C0382300
	C(JSTM1)=0.	C0382400
	DO 200 K=KST, KSP	C0382500
	DO 200 I=IST, ISP	C0382600
	DO 201 J=JST, JSP	C0382700
	A(J)=AN(I, J, K)	C0382800
	B(J)=AS(I, J, K)	C0382900
	C(J)=AE(I, J, K)*PHI(I-1, J, K)+AW(I, J, K)*PHI(I-1, J, K)	C0383000
&	-AF(I, J, K)*PHI(I, J, K+1)+AB(I, J, K)*PHI(I, J, K-1)+SU(I, J, K)	C0383100
	TERM=1./ (AP(I, J, K)-B(J)*A(J-1))	C0383200
	IF (ABS(A(J)).LE.1.0E-10) A(J)=0.0	C0383210
	IF (ABS(B(J)).LE.1.0E-10) B(J)=0.0	C0383220
	IF (ABS(C(J)).LE.1.0E-10) C(J)=0.0	C0383230
	IF (ABS(TERM).LE.1.0E-10) TERM=0.0	C0383240
	A(J)=A(J)*TERM	C0383300
	C(J)=(C(J)+B(J)*C(J-1))*TERM	C0383400
201	CONTINUE	C0383800
	PHI(I, JSP, K)=C(JSP)	C0383900
	JJSTA=JST-1	C0384000
	DO 202 JJ=JJSTA, JSP	C0384100
	J=JST-JSP-JJ	C0384200
	JJ1=J-1	C0384300
	PHI(I, J, K)=A(J)*PHI(I, JJ1, K)+C(J)	C0384400
202	CONTINUE	C0384500
200	CONTINUE	C0384600
		C0384700

	DO 2001 J=JST, JSP	00384800
	DO 2001 K=KST, KSP	00384900
	PHI(IST-1, J, K)=PHI(ISP, J, K)	00385000
	PHI(ISP+1, J, K)=PHI(IST, J, K)	00385100
2001	CONTINUE	00385200
		00385300
	KSTM1=KST-1	00385400
	A(KSTM1)=0.	00385500
	C(KSTM1)=0.	00385600
	DO 300 I=IST, ISP	00385700
	DO 300 J=JST, JSP	00385800
	DO 301 K=KST, KSP	00385900
	A(K)=AF(I, J, K)	00386000
	B(K)=AB(I, J, K)	00386100
	C(K)=AE(I, J, K)*PHI(I+1, J, K)+AW(I, J, K)*PHI(I-1, J, K)	00386200
	& -AN(I, J, K)*PHI(I, J+1, K)+AS(I, J, K)*PHI(I, J-1, K)+SU(I, J, K)	00386300
	TERM=1./(AP(I, J, K)-B(K)*A(K-1))	00386400
	IF (ABS(A(K)).LE.1.0E-10) A(K)=0.0	00386500
	IF (ABS(B(K)).LE.1.0E-10) B(K)=0.0	00386510
	IF (ABS(C(K)).LE.1.0E-10) C(K)=0.0	00386520
	IF (ABS(TERM).LE.1.0E-10) TERM=0.0	00386530
	A(K)=A(K)*TERM	00386540
	C(K)=(C(K)+B(K)*C(K-1))*TERM	00386600
301	CONTINUE	00386700
	PHI(I, J, KSP)=C(KSP)	00387100
	KSTA=KST+1	00387200
	DO 302 KK=KSTA, KSP	00387300
	K=KST+KSP-KK	00387400
	KP1=K+1	00387500
	PHI(I, J, K)=A(K)*PHI(I, J, KP1)+C(K)	00387600
302	CONTINUE	00387700
300	CONTINUE	00387800
		00387900
		00388000
	DO 2002 J=JST, JSP	00388100
	DO 2002 K=KST, KSP	00388200
	PHI(IST-1, J, K)=PHI(ISP, J, K)	00388300
	PHI(ISP-1, J, K)=PHI(IST, J, K)	00388400
2002	CONTINUE	00388500
		00388600
		00388700
	GOTO 700	00388800
		00388900
4405	CONTINUE	00388900
405	KSP1=KSP-1	00389000
	B(KSP1)=0.	00389100
	C(KSP1)=0.	00389200
	DO 600 II=IST, ISP	00389300
	I=IST+ISP-II	00389400
	DO 600 JJ=JST, JSP	00389500
	J=JST+JSP-JJ	00389600
	DO 601 KK=KST, KSP	00389700
	K=KSP-KST-KK	00389800
	KP1=K-1	00389900
	A(K)=AF(I, J, K)	00390000
	B(K)=AB(I, J, K)	00390100
	C(K)=AE(I, J, K)*PHI(I+1, J, K)+AW(I, J, K)*PHI(I-1, J, K)+AN(I, J, K)*	00390200
	& PHI(I, J+1, K)+AS(I, J, K)*PHI(I, J-1, K)+SU(I, J, K)	00390300
	TERM=1./(AP(I, J, K)-A(K)*B(K+1))	00390400
	B(K)=B(K)*TERM	00390500
	C(K)=(C(K)+A(K)*C(K+1))*TERM	00390600
	IF (ABS(A(K)).LE.1.0E-10) A(K)=0.0	00390700
	IF (ABS(B(K)).LE.1.0E-10) B(K)=0.0	00390800
	IF (ABS(C(K)).LE.1.0E-10) C(K)=0.0	00390900
601	CONTINUE	00391000
	PHI(I, J, KST)=C(KST)	00391100
	KSTP1=KST+1	00391200
	DO 602 K=KSTP1, KSP	00391300
		00391400

	PHI(I,J,K)=B(K)*PHI(I,J,K-1)+C(K)	00391500
602	CONTINUE	00391600
600	CONTINUE	00391700
		00391800
	DO 2003 J=JST,JSP	00391900
	DO 2003 K=KST,KSP	00392000
	PHI(IST-1,J,K)=PHI(ISP,J,K)	00392100
	PHI(ISP+1,J,K)=PHI(IST,J,K)	00392200
2003	CONTINUE	00392300
		00392400
		00392500
	JSP1=JSP+1	00392600
	B(JSP1)=0.	00392700
	C(JSP1)=0.	00392800
	DO 500 KK=KST,KSP	00392900
	K=KST+KSP-KK	00393000
	DO 500 II=IST,ISP	00393100
	I=IST+ISP-II	00393200
	DO 501 JJ=JST,JSP	00393300
	J=JSP+JST-JJ	00393400
	JP1=J+1	00393500
	A(J)=AN(I,J,K)	00393600
	B(J)=AS(I,J,K)	00393700
	C(J)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)+AF(I,J,K)*	00393800
	& PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)	00393900
	TERM=1./(AP(I,J,K)-A(J)*B(J+1))	00394000
	B(J)=B(J)*TERM	00394100
	C(J)=(C(J)+A(J)*C(J+1))*TERM	00394200
	IF (ABS(A(J)).LE.1.0E-10) A(J)=0.0	00394300
	IF (ABS(B(J)).LE.1.0E-10) B(J)=0.0	00394400
	IF (ABS(C(J)).LE.1.0E-10) C(J)=0.0	00394500
501	CONTINUE	00394600
	PHI(I,JST,K)=C(JST)	00394700
	JSTP1=JST+1	00394800
	DO 502 J=JSTP1,JSP	00394900
	PHI(I,J,K)=B(J)*PHI(I,J-1,K)+C(J)	00395000
502	CONTINUE	00395100
500	CONTINUE	00395200
		00395300
	DO 2004 J=JST,JSP	00395400
	DO 2004 K=KST,KSP	00395500
	PHI(IST-1,J,K)=PHI(ISP,J,K)	00395600
	PHI(ISP-1,J,K)=PHI(IST,J,K)	00395700
2004	CONTINUE	00395800
		00395900
		00396000
	ISP1=ISP-1	00396100
	B(ISP1)=0.	00396200
	C(ISP1)=0.	00396300
	DO 400 JJ=JST,JSP	00396400
	J=JST+JSP-JJ	00396500
	DO 400 KK=KST,KSP	00396600
	K=KST+KSP-KK	00396700
	DO 401 II=IST,ISP	00396800
	I=ISP+IST-II	00396900
	IP1=I-1	00397000
	A(I)=AE(I,J,K)	00397100
	B(I)=AW(I,J,K)	00397200
	C(I)=AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+AF(I,J,K)*	00397300
	& PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)	00397400
	TERM=1./(AP(I,J,K)-A(I)*B(I-1))	00397500
	B(I)=B(I)*TERM	00397600
	C(I)=(C(I)-A(I)*C(I+1))*TERM	00397700
	IF (ABS(A(I)).LE.1.0E-10) A(I)=0.0	00397800
	IF (ABS(B(I)).LE.1.0E-10) B(I)=0.0	00397900
	IF (ABS(C(I)).LE.1.0E-10) C(I)=0.0	00398000
401	CONTINUE	00398100
	PHI(IST,J,K)=C(IST)	00398200

```

ISTP1=IST+1
DO 402 I=ISTP1,ISP
PHI(I,J,K)=B(I)*PHI(I-1,J,K)+C(I)
402 CONTINUE
400 CONTINUE

DO 2005 J=JST,JSP
DO 2005 K=KST,KSP
PHI(IST-1,J,K)=PHI(ISP,J,K)
PHI(IST+1,J,K)=PHI(IST,J,K)
2005 CONTINUE

700 CONTINUE
RETURN
END

C *****
BLOCK DATA
C *****

COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP
COMMON/BL12/ NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PM1,PM2
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,UO,H,UGRT,BUOY,
& CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO
DATA NIP2,NIP1,NI,NIM1/23,22,21,20/
DATA NJP2,NJP1,NJ,NJM1/17,16,15,14/
DATA NKP2,NKP1,NK,NKM1/33,32,31,30/
DATA NAP1,NA,NAM1,NBP1,NB,NBM1/9,8,7,27,26,25/
DATA UO,TA,PRT,RHOO,CPO,VISO,NTMAX0/
& 1.0,555.86,1.0,0.0714,0.24,1.56E-4,0/
DATA TINF,CNT,ABTURB,BTURB/1.0,0.2,2.0,1.0/
DATA GC,RAIR/32.17,53.34/
DATA QCORRT,PM1/1.0,0.9/
END

C *****
SUBROUTINE GRID
C *****

COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP

C *** REGENERATION OF GRID

PI=4.*ATAN(1.)
DX=1.0/FLOAT(NIM1)
C DY=1./FLOAT(NJM1-2)
DY=1./FLOAT(NJM1-1)
DZ=PI/FLOAT(NKM1-NB+NA-2)

DO 19 I=1,NIP2
XS(I)=(I-2)*DX*2.0*PI
19 CONTINUE

C XS(1)=-DX*2.0*PI
C XS(2)=0.0
C XS(3)=0.001*2.0*PI
C DO 19 I=4,13
C XS(I)=(I-3)*DX*2.0*PI
C 19 CONTINUE

```

C		00405100
C	XS(14)=XS(13)	00405200
C	XS(13)=XS(14)-0.01*2.0*PI	00405300
C	DO 18 I=15,NIP1	00405400
C	XS(I)=XS(14)+(I-14)*DX*2.0*PI	00405500
C	18 CONTINUE	00405600
C	XS(NIP2)=XS(NIP1)+XS(3)	00405700
		00405800
		00405900
	YS(1)=0.000	00406000
	YS(2)=0.025	00406100
C	YS(3)=0.05	00406200
	DO 3 J=3,NJ	00406300
	YS(J)=(J-2)*DY	00406400
3	CONTINUE	00406500
	YS(NJP1)=YS(NJ)	00406600
	YS(NJ)=YS(NJP1)-3./8./12./9.6	00406700
	YS(NJP2)=YS(NJP1)+3./8./12./9.6	00406800
		00406900
CC	DO 3 J=4,NJP2	00407000
CC	YS(J)=(J-3)*DY	00407100
CC	3 CONTINUE	00407200
	DO 4 I=1,NIP1	00407300
	IP1=I+1	00407400
	DXXC(I)=XS(IP1)-XS(I)	00407500
4	CONTINUE	00407600
		00407700
	DXXC(NIP2)=DXXC(NIP1)	00407800
	DO 5 I=2,NIP2	00407900
	IM1=I-1	00408000
	DXXS(I)=.5*(DXXC(I)+DXXC(IM1))	00408100
5	CONTINUE	00408200
	DXXS(1)=DXXS(2)	00408300
		00408400
	DO 7 J=1,NJP1	00408500
	JP1=J+1	00408600
	DYYC(J)=YS(JP1)-YS(J)	00408700
7	CONTINUE	00408800
		00408900
	DYYC(NJP2)=DYYC(NJP1)	00409000
	DO 8 J=2,NJP2	00409100
	JM1=J-1	00409200
	DYYS(J)=.5*(DYYC(J)+DYYC(JM1))	00409300
8	CONTINUE	00409400
	DYYS(1)=DYYS(2)	00409500
		00409600
	DO 20 I=1,NIP2	00409700
	XC(I)=XS(I)+DXXC(I)/2.0	00409800
20	CONTINUE	00409900
		00410000
	DO 21 J=1,NJP2	00410100
	YC(J)=YS(J)+DYYC(J)/2.0	00410200
21	CONTINUE	00410300
		00410400
		00410500
	DO 9 K=4,NA	00410600
	ZS(K)=(K-3)*DZ	00410700
9	CONTINUE	00410800
		00410900
	DO 30 K=NBP1,NK	00411000
	ZS(K)=ZS(NA)+(K-NB)*DZ	00411100
30	CONTINUE	00411200
		00411300
	DO 31 K=NAP1,NB	00411400
	ZS(K)=PI/2.	00411500
31	CONTINUE	00411600
		00411700
	ZS(1)=0.0	00411800

	ZS(2)=0.05	00411900
	ZS(3)=0.10	00412000
C	ZS(NKP1)=ZS(NKM1)	00412100
C	ZS(NK)=ZS(NKP1)-0.05	00412200
C	ZS(NKM1)=ZS(NKP1)-0.10	00412300
C	ZS(NKP2)=ZS(NKP1)+0.05	00412400
		00412500
	ZS(NKP2)=ZS(NK)	00412600
	ZS(NKP1)=ZS(NKP2)-0.05	00412700
	ZS(NK)=ZS(NKP2)-0.10	00412800
		00412900
		00413000
	DO 10 K=1,NKP1	00413100
	IF (K.GE.NA.AND.K.LT.NB) GOTO 10	00413200
	KP1=K+1	00413300
	DZZC(K)=ZS(KP1)-ZS(K)	00413400
10	CONTINUE	00413500
		00413600
	DO 32 K=NA,NBM1	00413700
	DZZC(K)=2.854/(NB-NA)	00413800
32	CONTINUE	00413900
		00414000
	DZZC(NKP2)=DZZC(NKP1)	00414100
		00414200
	DO 11 K=2,NKP2	00414300
C	IF (K.EQ.NA.OR.K.EQ.NB) GOTO 11	00414400
	KM1=K-1	00414500
	DZZS(K)=.5*(DZZC(K)+DZZC(KM1))	00414600
11	CONTINUE	00414700
		00414800
	DZZS(1)=DZZS(2)	00414900
	DO 22 K=1,NKP2	00415000
	IF (K.GE.NA.AND.K.LT.NB) GOTO 22	00415100
	ZC(K)=ZS(K)+DZZC(K)/2.0	00415200
22	CONTINUE	00415300
		00415400
	DO 33 K=NA,NBM1	00415500
	ZC(K)=PI/2.	00415600
33	CONTINUE	00415700
		00415800
	IF (YS(1).LT.0.0) YS(1)=0.0	00415900
	IF (YC(1).LT.0.0) YC(1)=0.0	00416000
	PRINT *	00416100
	PRINT *, ' INPUT COORDINATE OF THE TANK IN THE ORDER OF '	00416200
	PRINT *, ' XS YS ZS XC YC',	00416300
	& ' ZC DXXS DYYS DZZS DXXC	00416400
	& , 'DYXC DZZC'	00416500
	DO 12 I=1,NKP2	00416600
C	WRITE(6,102) I,XS(I),YS(I),ZS(I),XC(I),YC(I),ZC(I),	00416700
C	& DXXS(I),DYYS(I),DZZS(I),DXXC(I),DYXC(I),DZZC(I)	00416800
C	102 FORMAT(2X,I4,12(2X,F8.5))	00416900
C	12 CONTINUE	00417000
		00417100
	RETURN	00417200
	END	00417300
		00417400
		00417500
		00417600
C	*****	00417700
C	FUNCTION XL(I,J,K,M,N)	00417800
C	*****	00417900
C	*****	00418000
C	WHEN M OR N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE*	00418100
C	HALF CELL (STAGGERED CELL) *	00418200
C	WHEN M OR N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*	00418300
C	HALF CELL (STAGGERED CELL) *	00418400
C	WHEN M OR N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*	00418500
C	HALF CELL (STAGGERED CELL) *	00418600


```

C      WHEN X = N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE* 00418700
C      WHOLE CELL * 00418800
C      WHEN X = N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE* 00418900
C      WHOLE CELL * 00419000
C      WHEN X = N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE* 00419100
C      WHOLE CELL * 00419200
C ***** 00419300
C 00419400
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93) 00419500
X1=XC(I) 00419700
X2=YC(J) 00419800
X3=ZC(K) 00419900
DXL=DXXC(I) 00420000
IF(M.EQ.N) GOTO 100 00420100
00420200
IF(M.EQ.1.OR.N.EQ.1) X1=XS(I) 00420300
IF(M.EQ.1.OR.N.EQ.1) DXL=DXXS(I) 00420400
IF(M.EQ.2.OR.N.EQ.2) X2=YS(J) 00420500
IF(M.EQ.3.OR.N.EQ.3) X3=ZS(K) 00420600
GOTO 1000 00420700
100 IF(M.EQ.1) X1=XC(I-1) 00420800
IF(M.EQ.1) DXL=DXXC(I-1) 00420900
IF(M.EQ.2) X2=YC(J-1) 00421000
IF(M.EQ.3) X3=ZC(K-1) 00421100
1000 CONTINUE 00421200
XL=X2*SIN(X3)*DXL 00421300
RETURN 00421400
END 00421500
00421600
00421700
00421800
C ***** 00421900
C FUNCTION YL(I,J,K,M,N) 00422000
C ***** 00422100
C ***** 00422200
C      WHEN M OR N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE* 00422300
C      HALF CELL (STAGGERED CELL) * 00422400
C      WHEN M OR N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE* 00422500
C      HALF CELL (STAGGERED CELL) * 00422600
C      WHEN M OR N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE* 00422700
C      HALF CELL (STAGGERED CELL) * 00422800
C      WHEN M = N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE* 00422900
C      WHOLE CELL * 00423000
C      WHEN M = N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE* 00423100
C      WHOLE CELL * 00423200
C      WHEN M = N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE* 00423300
C      WHOLE CELL * 00423400
C ***** 00423500
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93) 00423600
X1=XC(I) 00423700
X2=YC(J) 00423800
X3=ZC(K) 00423900
DYL=DYXC(J) 00424000
IF(M.EQ.N) GOTO 100 00424100
00424200
IF(M.EQ.2.OR.N.EQ.2) X2=YS(J) 00424300
IF(M.EQ.2.OR.N.EQ.2) DYL=DYYS(J) 00424400
IF(M.EQ.1.OR.N.EQ.1) X1=XS(I) 00424500
IF(M.EQ.3.OR.N.EQ.3) X3=ZS(K) 00424600
GOTO 1000 00424700
100 IF(M.EQ.2) X2=YC(J-1) 00424800
IF(M.EQ.2) DYL=DYXC(J-1) 00424900
IF(M.EQ.1) X1=XC(I-1) 00425000
IF(M.EQ.3) X3=ZC(K-1) 00425100
1000 CONTINUE 00425200
YL=1.00*DYL 00425300
RETURN 00425400

```

```

END
00425500
00425600
00425700
00425800
00425900
00426000
00426100
C *****
C FUNCTION ZL(I,J,K,M,N)
C *****
C *****
C WHEN M OR N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE*
C HALF CELL (STAGGERED CELL) *
C *****
C WHEN M OR N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C HALF CELL (STAGGERED CELL) *
C *****
C WHEN M OR N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C HALF CELL (STAGGERED CELL) *
C *****
C WHEN M = N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE*
C WHOLE CELL *
C *****
C WHEN M = N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C WHOLE CELL *
C *****
C WHEN M = N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C WHOLE CELL *
C *****
C *****
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP
X1=XC(I)
X2=YC(J)
X3=ZC(K)
DZL=DZZC(K)
IF(M.EQ.N) GOTO 100
IF(M.EQ.2.OR.N.EQ.2) X2=YS(J)
IF(M.EQ.1.OR.N.EQ.1) X1=XS(I)
IF(M.EQ.3.OR.N.EQ.3) GOTO 200
GOTO 1000
200 CONTINUE
IF(K.EQ.NA.OR.K.EQ.NB) GOTO 2000
X3=ZS(K)
DZL=DZZS(K)
GOTO 1000
100 IF(M.EQ.3) X3=ZC(K-1)
IF(M.EQ.3) DZL=DZZC(K-1)
IF(M.EQ.2) X2=YC(J-1)
IF(M.EQ.1) X1=XC(I-1)
1000 CONTINUE
ZL=X2*DZL
GOTO 300
2000 CONTINUE
DZL1=DZZC(K-1)
DZL2=DZZC(K)
IF(K.EQ.NB) DZL1=DZZC(K)
IF(K.EQ.NB) DZL2=DZZC(K-1)
ZL=(X2*DZL1+DZL2)/2.
300 CONTINUE
RETURN
END
C *****
C FUNCTION SILIN(V1,V2,D1,D2)
C *****
C IF(D1.EQ.0.C.AND.D2.EQ.0.0) D1=0.1
C IF(D1.EQ.0.0.AND.D2.EQ.0.0) D2=0.1
C SILIN=(V1*D2-V2*D1)/(D1+D2)
C RETURN
C END
00431000
00431100
00431200
00431300
00431400
00431500
00431600
00431700
00431800
00431900
00432000
00432100
00432200

```

		00432300
C	***** FUNCTION BILIN (V1, V2, D1, D2, V3, V4, D3, D4, D5, D6) *****	00432400
C	V12= (V1*D2+V2*D1) / (D1+D2) V34= (V3*D4+V4*D3) / (D3+D4) BILIN=(V12*D6+V34*D5) / (D5+D6) END	00432500 00432600 00432700 00432800 00432900 00433000 00433100 00433200
C	***** SUBROUTINE STRESS *****	00433300 00433400 00433500
C	COMMON/R4/XC (93), YC (93), ZC (93), XS (93), YS (93), ZS (93), & DXXC (93), DYXC (93), DZXC (93), DXXS (93), DYYS (93), DZZS (93) COMMON/BL1/DX, DY, DZ, VOL, DTIME, VOLDT, THOT, TCOOL, PI, Q, QR COMMON/BL7/NI, NIP1, NIM1, NJ, NJP1, NJM1, NK, NKP1, NKM1 & , NIP2, NJP2, NKP2, NA, NAP1, NAM1, NB, NBP1, NBM1, KRUN, NCHIP, NJRA, NWRP COMMON/BL20/SIG11 (22, 16, 32), SIG12 (22, 16, 32), SIG22 (22, 16, 32) & , SIG13 (22, 16, 32), SIG23 (22, 16, 32), SIG33 (22, 16, 32) COMMON/BL22/ICHPB (10), NCHPI (10), JCHPB (10), NCHPJ (10), KCHPB (10), & NCHPK (10), TCHP (10), CPS (10), CONS (10), WFAN (10) COMMON/BL32/ T (22, 16, 32), R (22, 16, 32), P (22, 16, 32) & , C (22, 16, 32), U (22, 16, 32), V (22, 16, 32), W (22, 16, 32) COMMON/BL37/ VIS (22, 16, 32), COND (22, 16, 32), NOD (22, 16, 32), RWALL (579) & , CPM (22, 16, 32), HSZ (3, 2), NHSZ (22, 16, 32), RESORM (93)	00433600 00433700 00433800 00433900 00434000 00434100 00434200 00434300 00434400 00434500 00434600 00434700 00434800 00434900 00435000
	DO 100 K=2, NK KP2=K+2 KP1=K+1 KM1=K-1 KM2=K-2 DO 100 J=2, NJ JP2=J+2 JP1=J+1 JM1=J-1 JM2=J-2 DO 100 I=2, NI IP2=I+2 IP1=I+1 IM1=I-1 IM2=I-2	00435100 00435200 00435300 00435400 00435500 00435600 00435700 00435800 00435900 00436000 00436100 00436200 00436300 00436400 00436500 00436600
C	CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME	00436700 00436800
	DXP1=XL (IP1, J, K, 0, 0) DXI =XL (I, J, K, 0, 0) DXM1=XL (IM1, J, K, 0, 0)	00436900 00437000 00437100 00437200
	DYP1=YL (I, JP1, K, 0, 0) DYJ =YL (I, J, K, 0, 0) DYM1=YL (I, JM1, K, 0, 0)	00437300 00437400 00437500 00437600
	DZP1=ZL (I, J, KP1, 0, 0) DZK =ZL (I, J, K, 0, 0) DZM1=ZL (I, J, KM1, 0, 0)	00437700 00437800 00437900 00438000
C ***	SURFACE LENGTH OF THE CONTROL VOLUME	00438100 00438200
	DXN=XL (I, JP1, K, 0, 2) DXS=XL (I, J, K, 0, 2) DXF=XL (I, J, KP1, 0, 3) DXB=XL (I, J, K, 0, 3)	00438300 00438400 00438500 00438600 00438700
	DYF=YL (I, J, KP1, 0, 3) DYB=YL (I, J, K, 0, 3) DYE=YL (IP1, J, K, 0, 1)	00438800 00438900 00439000

DYW=YL(I, J, K, 0, 1)	00439100
DZE=ZL(IP1, J, K, 0, 1)	00439200
DZW=ZL(I, J, K, 0, 1)	00439300
DZN=ZL(I, JP1, K, 0, 2)	00439400
DZS=ZL(I, J, K, 0, 2)	00439500
C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR T	00439600
DXEE=XL(IP2, J, K, 0, 1)	00439700
DXE =XL(IP1, J, K, 0, 1)	00439800
DXW =XL(I, J, K, 0, 1)	00439900
DXWW=XL(IM1, J, K, 0, 1)	00440000
DYNN=YL(I, JP2, K, 0, 2)	00440100
DYN =YL(I, JP1, K, 0, 2)	00440200
DYS =YL(I, J, K, 0, 2)	00440300
DYSS=YL(I, JM1, K, 0, 2)	00440400
DZFF=ZL(I, J, KP2, 0, 3)	00440500
DZF =ZL(I, J, KP1, 0, 3)	00440600
DZB =ZL(I, J, K, 0, 3)	00440700
DZBB=ZL(I, J, KM1, 0, 3)	00440800
UBAR=0.5*(U(IP1, J, K)+U(I, J, K))	00440900
VBAR=0.5*(V(I, JP1, K)+V(I, J, K))	00441000
WBAR=0.5*(W(I, J, KP1)+W(I, J, K))	00441100
DXY=DXI*DYJ	00441200
DYZ=DYJ*DZK	00441300
DZX=DZK*DXI	00441400
SIG11(I, J, K)=2.*VIS(I, J, K)*((U(IP1, J, K)-U(I, J, K))/DXI	00441500
& +VBAR*(DXN-DXS)/DXY	00441600
& +WBAR*(DXF-DXB)/DZX)	00441700
SIG22(I, J, K)=2.*VIS(I, J, K)*((V(I, JP1, K)-V(I, J, K))/DYJ	00441800
& -WBAR*(DYF-DYB)/DYZ	00441900
& +UBAR*(DYE-DYW)/DXY)	00442000
SIG33(I, J, K)=2.*VIS(I, J, K)*((W(I, J, KP1)-W(I, J, K))/DZK	00442100
& -UBAR*(DZE-DZW)/DZX	00442200
& -VBAR*(DZN-DZS)/DYZ)	00442300
100 CONTINUE	00442400
DO 200 K=2, NKP1	00442500
KP2=K-2	00442600
KP1=K-1	00442700
KM1=K-1	00442800
KM2=K-2	00442900
DO 200 J=2, NJP1	00443000
JP2=J-2	00443100
JP1=J-1	00443200
JM1=J-1	00443300
JM2=J-2	00443400
DO 200 I=2, NIP1	00443500
IP2=I-2	00443600
IP1=I-1	00443700
IX1=I-1	00443800
IX2=I-2	00443900
C **** FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CONTROL	00444000
C VOLUME FOR SIG12	00444100
C IF (J.EQ.2) GOTO 300	00444200
DXN=XL(I, J, K, 1, 0)	00444300
DXS=XL(I, JM1, K, 1, 0)	00444400
	00444500
	00444600
	00444700
	00444800
	00444900
	00445000
	00445100
	00445200
	00445300
	00445400
	00445500
	00445600
	00445700
	00445800

DYE=YL(I , J, K, 2, 0)	00445900
DYW=YL(IM1, J, K, 2, 0)	00446000
DXI=XL(I , J, K, 1, 2)	00446100
DYJ=YL(I , J, K, 2, 1)	00446200
	00446300
DYN=YL(I, J , K, 1, 0)	00446400
DYS=YL(I, JM1, K, 1, 0)	00446500
DXE=XL(I , J, K, 2, 0)	00446600
DXW=XL(IM1, J, K, 2, 0)	00446700
	00446800
UBAR=SILIN(U(I, J, K), U(I, JM1, K), DYN, DYS)	00446900
VBAR=SILIN(V(I, J, K), V(IM1, J, K), DXE, DXW)	00447000
	00447100
VIS12=BILIN(VIS(I , J, K), VIS(I , JM1, K), DYN, DYS,	00447200
& VIS(IM1, J, K), VIS(IM1, JM1, K), DYN, DYS, DXE, DXW)	00447300
	00447400
SIG12(I, J, K)= VIS12*((V(I, J, K)-V(IM1, J, K))/DXI	00447500
& -VBAR*(DYE-DYW)/(DXI*DYJ))	00447600
SIG12(I, J, K)=SIG12(I, J, K)+VIS12*((U(I, J, K)-U(I, JM1, K))/DYJ	00447700
& -UBAR*(DXN-DXS)/(DXI*DYJ))	00447800
300 CONTINUE	00447900
	00448000
C **** FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CONTROL	00448100
C VOLUME FOR SIG13	00448200
	00448300
DXF=XL(I, J, K , 1, 0)	00448400
DXB=XL(I, J, KM1, 1, 0)	00448500
DZE=ZL(I , J, K, 3, 0)	00448600
DZW=ZL(IM1, J, K, 3, 0)	00448700
DXI=XL(I , J, K, 1, 3)	00448800
DZK=ZL(I , J, K, 3, 1)	00448900
	00449000
DZF=ZL(I, J, K , 1, 0)	00449100
DZB=ZL(I, J, KM1, 1, 0)	00449200
DXE=XL(I , J, K, 3, 0)	00449300
DXW=XL(IM1, J, K, 3, 0)	00449400
	00449500
IF (DZF.EQ.0.0.OR.DZB.EQ.0.0.OR.DZE.EQ.0.0.OR.DZW.EQ.0.0)	00449600
& WRITE (6, *) I, J, K, DZF, DZB, DZE, DZW	00449700
UBAR=SILIN(U(I, J, K), U(I, J, KM1), DZF, DZB)	00449800
WBAR=SILIN(W(I, J, K), W(IM1, J, K), DXE, DXW)	00449900
	00450000
VIS13=BILIN(VIS(I , J, K), VIS(I , J, KM1), DZF, DZB,	00450100
& VIS(IM1, J, K), VIS(IM1, J, KM1), DZF, DZB, DXE, DXW)	00450200
	00450300
SIG13(I, J, K)= VIS13*((W(I, J, K)-W(IM1, J, K))/DXI	00450400
& -WBAR*(DZE-DZW)/(DXI*DZK))	00450500
SIG13(I, J, K)=SIG13(I, J, K)+VIS13*((U(I, J, K)-U(I, J, KM1))/DZK	00450600
& -UBAR*(DXF-DXB)/(DXI*DZK))	00450700
	00450800
	00450900
C **** FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CONTROL	00451000
C VOLUME FOR SIG23	00451100
	00451200
DZN=ZL(I, J , K, 3, 0)	00451300
DZS=ZL(I, JM1, K, 3, 0)	00451400
DYF=YL(I, J, K , 2, 0)	00451500
DYB=YL(I, J, KM1, 2, 0)	00451600
DZK=ZL(I, J, K, 3, 2)	00451700
DYJ=YL(I, J, K, 2, 3)	00451800
	00451900
DYN=YL(I, J , K, 3, 0)	00452000
DYS=YL(I, JM1, K, 3, 0)	00452100
DZF=ZL(I, J, K , 2, 0)	00452200
DZB=ZL(I, J, KM1, 2, 0)	00452300
	00452400
WBAR=SILIN(W(I, J, K), W(I, JM1, K), DYN, DYS)	00452500
VBAR=SILIN(V(I, J, K), V(I, J, KM1), DZF, DZB)	00452600

```

VIS23=BILIN(VIS(I ,J,K),VIS(I,JM1,K ),DYN,DYS,
& VIS(I,J,KM1),VIS(I,JM1,KM1),DYN,DYS, DZF,DZB)
SIG23(I,J,K)= VIS23*((V(I,J,K)-V(I,J,KM1))/DZK
& -VBAR*(DYF-DYB)/(DZK*DYJ))
SIG23(I,J,K)=SIG23(I,J,K)+VIS23*((W(I,J,K)-W(I,JM1,K))/DYJ
& -WBAR*(DZN-DZS)/(DZK*DYJ))
200 CONTINUE
DO 110 I=1,NIP1
DO 110 J=1,NJP1
C WRITE (6,998) I,J,SIG11(I,J,5),SIG12(I,J,5),SIG13(I,J,5),
C & SIG22(I,J,5),SIG23(I,J,5),SIG33(I,J,5)
998 FORMAT (2X,I4,1X,I4,6(1X,E11.4))
110 CONTINUE
RETURN
END
C
***
*****
SUBROUTINE CALQ(LL)
*****
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1
COMMON/BL12/NWRITE,NTAPE,NTMAXO,NTREAL,TIME,SORSUM,ITER
COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PM1,PM2
COMMON/BL16/CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,
& CP0,PRT,CONDO,VISC,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO
COMMON/BL34/HEIGHT(22,16,32),REQ(22,16,32),
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),
& DU(22,16,32),DV(22,16,32),DW(22,16,32)
COMMON/BL37/VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579)
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)
COMMON/BL39/ALEW,PCURVE,CONGRA,PCURM1,PSOUTH,QCORR,PERROR
C *** IN MANY OF THE FOLLOWING LINES A TEMPORARY CORRECTION FOR
C * ADJUSTING QQ TO AGREE WITH THE PRESSURE HAS BEEN APPLIED.
XTIME=TIME*H/U0
VOLT=0.0
DO 113 I=2,NI
DO 113 J=2,NJ
DO 113 K=16,17
IF (NHSZ(I,J,K).EQ.0) GOTO 113
DXI=XI(I,J,K,0,0)
DYJ=YI(I,J,K,0,0)
DZK=ZI(I,J,K,0,0)
VOL=DXI*DYJ*DZK*H*H*H
VOLT=VOLT+VOL
113 CONTINUE
QRVOL=0.
DO 70 I=561,579
QRVOL=QRVOL+RWALL(I)*1./12.*0.2*PI
70 CONTINUE
C
QR=QRVOL/VOLT*U0*CP0*RHOO*TA/H
IF (XTIME.LT.23.1) THEN
PCURVE=9.789522E-5*XTIME**2-2.388310E-6*XTIME**3-
& REQ(10,9,16)
& CPDT =9.789522E-5*XTIME**2-2.388310E-6*XTIME**2*3
ELSE
PCURVE=0.0052+.81264E-3*XTIME-.22604E-5*XTIME**2-.27262E-8*XTIME**00457300
00452700
00452800
00452900
00453000
00453100
00453200
00453300
00453400
00453500
00453600
00453700
00453800
00453900
00454000
00454100
00454200
00454300
00454400
00454500
00454600
00454700
00454800
00454900
00455000
00455100
00455200
00455210
00455300
00455400
00455500
00455600
00455700
00455800
00455900
00455910
00455920
00456000
00456100
00456200
00456300
00456400
00456500
00456510
00456520
00456530
00456540
00456550
00456560
00456570
00456580
00456590
00456591
00456592
00456593
00456594
00456595
00456596
00456597
00456598
00456599
00456600
00456700
00456800
00456900
00457000
00457100
00457200
00457300

```

```

&      3-.115621E-11*XTIME**4+REQ(10,9,16)                                00457400
DPDT=.81264E-3-.22604E-5*XTIME*2+.27262E-8*XTIME**2                    00457500
&      2*3.0-.115621E-11*XTIME**3*4                                      00457600
      ENDIF                                                                00457700
      IF ( LL .EQ. 1) THEN                                                00457710
      QQ=1.0E8*DPDT                                                       00457800
      Q=QQ*3.4134/60./60.                                                00457900
65  CONTINUE                                                                00458000
      Q=Q*QCORRT-QR                                                       00458100
                                                                              00458200
      ELSE                                                                  00458300
C  THIS USES A CURVE FIT THROUGH THE BURNRATE DATA GIVEN BY NRL        00458400
      QCORRT=0.0                                                           00458410
      QCORR=0.0                                                           00458420
      ITEST = 0                                                           00458500
      BURNR1= 5.4576748 +0.18815346*XTIME-.20153996E-03*XTIME**2        00458600
      BURNR2= -1.3116787 + .33158595*XTIME-.7342952E-03*XTIME**2        00458700
&      +.50945510E-06*XTIME**3                                           00458800
      IF (XTIME .LT. 100) THEN                                           00458900
      BURNR= BURNR2 + 1.3117-.013117*XTIME                               00459000
      ELSE                                                                  00459100
      BURNR = BURNR2                                                       00459200
      ENDIF                                                                00459300
      IF(XTIME .LE. 300) GO TO 60                                         00459400
      IF(BURNR2 .LT. BURNR1) THEN                                         00459500
      BURNR = (BURNR1 + BURNR2) / 2                                       00459600
      GO TO 60                                                             00459700
      ELSE                                                                  00459800
      IF ( XTIME .LT. 600.0) GO TO 60                                     00459900
      IF (ITEST .EQ. 0) THEN                                             00460000
      BURNR3 = BURNR2                                                     00460100
      ITEST = 1                                                           00460200
      ENDIF                                                                00460300
      BURNR = BURNR3                                                      00460400
      ENDIF                                                                00460500
60  Q = BURNR*2.2046*9612./3600.-QR                                       00460600
CC  THIS GIVES Q IN BTU/SEC                                              00460700
                                                                              00460800
      ENDIF                                                                00460900
      Q=59.313+0.7195*XTIME-0.1139E-2*XTIME**2-0.3367E-5*XTIME**3      00460910
      Q=Q*3412/3600                                                       00460920
      RETURN                                                                00461000
      END                                                                    00461100
                                                                              00461200
                                                                              00461300
                                                                              00461400
                                                                              00461500
C  -----                                                                00461600
***  SUBROUTINE RADHT(T4WALL,VFMXC)                                       00461700
***  -----                                                                00461800
      COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1                 00461900
&      ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP    00462000
      COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,CO,H,UGRT,BUOY, 00462100
&      CPO,PRT,CONDC,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR 00462200
      COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32)                 00462300
&      ,C(22,16,32),J(22,16,32),V(22,16,32),W(22,16,32)              00462400
      COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579) 00462500
&      ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)              00462600
      COMMON/BL39/ALEW,PCURVE,CONSR,A,PCURM1,P,SOUTH,QCORR,PERROR        00462700
                                                                              00462800
                                                                              00462900
      DIMENSION VFMXC(579,579),T4WALL(579)                               00463000
      DO 4010 K=3,NKM1                                                    00463100
      DO 4010 I=2,NI                                                       00463200
      II=(K-3)*(NI-I)-I-1                                                00463300
      T4WALL(II)=CONSR*A*T(I,NJRA,K)*T(I,NJRA,K)*T(I,NJRA,K)*T(I,NJRA,K) 00463400
4010 CONTINUE                                                            00463500
                                                                              00463600

```

```

C RADIATION FROM THE FIRE TO THE WALL                                00463700
DO 4011 J=3,9                                                       00463800
JJ=561+9-J                                                         00463900
AVT=0.25*(T(16,J,16)+T(17,J,16)+T(16,J,17)+T(17,J,17))         00464000
T4WALL(JJ)=CONGRA*AVT*AVT*AVT*AVT                                  00464100
4011 CONTINUE                                                       00464200
C                                                                      00464300
DO 4012 J=3,14                                                      00464400
JJ=568+J-3                                                          00464500
AVT=0.25*(T(6,J,16)+T(7,J,16)+T(6,J,17)+T(7,J,17))             00464600
T4WALL(JJ)=CONGRA*AVT*AVT*AVT*AVT                                  00464700
4012 CONTINUE                                                       00464800
C                                                                      00464900
DO 4020 I=1,579                                                      00465000
RWALL(I)=0.0                                                        00465100
DO 4020 J=1,579                                                      00465200
RWALL(I)=RWALL(I)+VFMXC(I,J)*T4WALL(J)                            00465300
4020 CONTINUE                                                       00465400
RETURN                                                                00465500
END                                                                    00465600
                                                                      00465700
                                                                      00465800
                                                                      00465900
                                                                      00466000
C                                                                      00466100
*** ***** 00466200
SUBROUTINE GLOBE                                                     00466300
*** ***** 00466400
* THIS SUBROUTINE CALCULATES THE GLOBAL PRESSURE CORRECTION,      *00466500
* WHEREBY THE PRESSURE MATRIX IS UPDATED.                          *00466600
* VARIABLES USED ARE:                                             *00466700
*      SUMT      =      SUM OF TEMPERATURES                       *00466800
*      SUMPT     =      SUM OF PRESSURE OVER TEMPERATURE          *00466900
*      SUMPET    =      SUM OF EQUILIBRIUM PRESSURE OVER TEMP    *00467000
*      UGRT     =      CONSTANT                                    *00467100
*      PCORR    =      PRESSURE CORRECTION                        *00467200
***** 00467300
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1                 00467400
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP 00467500
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,UO,H,UGRT,BUOY 00467600
& CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO0467700
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32)                00467800
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32)             00467900
COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32),                      00468000
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),                    00468100
& DU(22,16,32),DV(22,16,32),DW(22,16,32)                        00468200
COMMON/BL37/ VLS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579)00468300
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)            00468400
                                                                      00468500
SUMT=0.                                                              00468600
SUMPT=0.                                                             00468700
SUMPET=0.                                                            00468800
DO 370 I=2,NI                                                        00468900
DO 370 J=2,NJ                                                        00469000
DO 370 K=2,NK                                                        00469100
IF (NOD(I,J,K).EQ.1) GOTO 370                                       00469200
DXI=XL(I,J,K,0,0,0)                                                 00469300
DYJ=YL(I,J,K,0,0,0)                                                 00469400
DZK=ZL(I,J,K,0,0,0)                                                 00469500
VOL=DXI*DYJ*DZK                                                    00469600
SUMT=SUMT-1./T(I,J,K)*VOL                                           00469700
SUMPT=SUMPT+P(I,J,K)/T(I,J,K)*VOL                                    00469800
SUMPET=SUMPET-REQ(I,J,K)*(1./1.0-1./T(I,J,K))*VOL                 00469900
370 CONTINUE                                                         00470000
SUMPET=SUMPET/UGRT                                                  00470100
PCORR=(SUMPET-SUMPT)/SUMT                                           00470200
PCORRN=PCORR                                                         00470300
                                                                      00470400

```



```

DO 371 I=1,NIP1                                00470500
DO 371 J=1,NJP1                                00470600
DO 371 K=1,NKP1                                00470700
P(I,J,K)=P(I,J,K)+PCORRN                       00470800
371 CONTINUE                                    00470900
                                                00471000
RETURN                                           00471100
END                                               00471200
                                                00471300
                                                00471400
                                                00471500

```

```

C
*** *****
SUBROUTINE SOLCON                               00471700
*** *****
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1 00472000
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP 00472100
COMMON/BL12/ NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER 00472200
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,UO,H,UGRT,BUOY, 00472300
& CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO0472400
COMMON/BL22/ICHPB(10),NCHPI(10),JCHPB(10),NCHPJ(10),KCHPB(10), 00472500
& NCHPK(10),TCHP(10),CPS(10),CONS(10),WFAN(10) 00472600
COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579) 00472700
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93) 00472800
                                                00472900
DO 402 N=1,NCHIP                                00473000
IB=ICHPB(N)                                     00473100
IE=IB-NCHPI(N)-1                               00473200
JB=JCHPB(N)                                     00473300
JE=JB-NCHPJ(N)-1                               00473400
KB=KCHPB(N)                                     00473500
KE=KB-NCHPK(N)-1                               00473600
DO 405 I=IB,IE-1                               00473700
DO 405 J=JB,JE-1                               00473800
DO 405 K=KB,KE-1                               00473900
COND(I,J,K)=CONDO*CONS(N)                     00474000
CPM(I,J,K)=CPS(N)                              00474100
NOD(I,J,K)=1                                   00474200
IF (.EQ.NJ) COND(I,NJP1,K)=COND(I,NJ,K)       00474300
IF (1.EQ.2) COND(1,J,K)=COND(2,J,K)          00474400
IF (1.EQ.NI) COND(NIP1,J,K)=COND(NI,J,K)      00474500
IF (1.EQ.2.AND.J.EQ.NJ) COND(1,J+1,K)=COND(2,J,K) 00474600
IF (1.EQ.NI.AND.J.EQ.NJ) COND(NIP1,J+1,K)=COND(NI,J,K) 00474700
IF (J.EQ.NJ) CPM(I,NJP1,K)=CPM(I,NJ,K)       00474800
IF (1.EQ.2) CPM(1,J,K)=CPM(2,J,K)            00474900
IF (1.EQ.NI) CPM(NIP1,J,K)=CPM(NI,J,K)       00475000
IF (1.EQ.2.AND.J.EQ.NJ) CPM(1,J+1,K)=CPM(2,J,K) 00475100
IF (1.EQ.NI.AND.J.EQ.NJ) CPM(NIP1,J+1,K)=CPM(NI,J,K) 00475200
405 CONTINUE                                    00475300
402 CONTINUE                                    00475400
RETURN                                           00475500
END                                               00475600
                                                00475700
                                                00475800
                                                00475900

```

```

C
*** *****
SUBROUTINE PTRACK                               00476100
*** *****
COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PM1,PM200476300
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,UO,H,UGRT,BUOY,00476400
& CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO0476500
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32) 00476600
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32) 00476700
COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32), 00476800
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32), 00476900
& DU(22,16,32),DV(22,16,32),DW(22,16,32) 00477000
COMMON/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR 00477100
                                                00477200

```

```

CC ** THE FOLLOWING PRESSURE TEST IS A TEMPORARY MEASURE TO MODIFY THE 00477300
CC HEAT INPUT TO FORCE THE CALCULATED PRESSURE TO AGREE WITH THE 00477400
CC EXPERIMENTAL PRESSURE. IT WILL BE USED UNTIL ACCURATE HEAT INPUT 00477500
CC ** IS RECEIVED. 00477600
CC 00477700
PSOUTH=P(10,9,16)*CONST1+REQ(10,9,16) 00477800
PERROR=(PCURVE-PSOUTH)/PCURVE 00477900
QCORR=1.0+PERROR-(PSOUTH-PM1)/PCURVE 00478000
QCORR=1.0+PERROR-(PSOUTH-PM1)/PCURVE+(PSOUTH-PM1)/(PCURVE-PCURM1)* 00478100
& (PCURVE-PM1)/PCURVE 00478200
QCORRT=QCORRT*QCORR 00478300
PCURM1=PCURVE 00478400
PM1=PSOUTH 00478500
C 00478600
RETURN 00478700
END 00478800
00478900
00479000
00479100
C 00479200
*** 00479300
SUBROUTINE TCP 00479400
*** 00479500
00479600
00479700
* THIS SUBROUTINE CALCULATES THE TEMPERATURE AT THE TERMOCOUPLE *00479800
* POSITIONS. *00479900
*****00480000
COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93), 00480100
& DXXC(93),DYXC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93) 00480200
COMMON/BL16/CONST1,CONST2,CONST3,CONST4,CONST6,NT,JO,H,UGRT,BUOY, 00480300
& CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO0480400
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32) 00480500
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32) 00480600
COMMON/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12) 00480700
00480800
00480900
DO 5100 N=1,NTHCO 00481000
II=NTH(N,1) 00481100
JJ=NTH(N,2) 00481200
KK=NTH(N,3) 00481300
VOL=ABS((XC(II+1)-XC(II))*(YC(JJ+1)-YC(JJ))*(ZC(KK+1)-ZC(KK))) 00481400
TCOUP(N)=0. 00481500
DO 5101 I=II,II+1 00481600
III=II-III-1-I 00481700
DO 5102 J=JJ,JJ+1 00481800
JJJ=JJ+JJ+1-J 00481900
DO 5103 K=KK,KK+1 00482000
KKK=KK-KK+1-K 00482100
WVOL=ABS((XC(I)-CX(N))*(YC(J)-CY(N))*(ZC(K)-CZ(N)))/VOL 00482200
TCOUP(N)=TCOUP(N)+WVOL*T(III,JJJ,KKK) 00482300
5101 CONTINUE 00482400
TCOUP(N)=TCOUP(N)*TR-273.18 00482500
00482600
5100 CONTINUE 00482700
00482800
RETURN 00482900
END 00483000
00483100
00483200
00483300
C 00483400
*** 00483500
SUBROUTINE OUT(NN) 00483600
*** 00483700
COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR 00483800
COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1 00483900
& ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP 00484000

```

```

COMMON/BL12/ NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER 00484100
COMMON/BL14/HCOEF,TINF,CNT,ABTURB,BTURB,VISL,VISMAX,QCORRT,PM1,PM200484200
COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,00484300
& CPO,PRT,CONDO,VISO,RHOO,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIRO0484400
COMMON/BL32/ T(22,16,32),R(22,16,32),P(22,16,32) 00484500
& ,C(22,16,32),U(22,16,32),V(22,16,32),W(22,16,32) 00484600
COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32), 00484700
& SMP(22,16,32),SMPP(22,16,32),PP(22,16,32), 00484800
& DU(22,16,32),DV(22,16,32),DW(22,16,32) 00484900
COMMON/BL36/AP(22,16,32),AE(22,16,32),AW(22,16,32),AN(22,16,32), 00484910
& AS(22,16,32),AF(22,16,32),AB(22,16,32), 00484920
& SP(22,16,32),SU(22,16,32),RI(22,16,32) 00484930
COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOD(22,16,32),RWALL(579)00485000
& ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93) 00485100
COMMON/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12) 00485200
COMMON/BL39/ALEW,PCURVE,CONSR,PCURM1,PSOUTH,QCORR,PERROR 00485300
XTIME=TIME*H/UO 00485400
nnn=jnint(xtime)
nnx=nnn+1
IF( NN .EQ. 1 ) THEN 00485500
C 00485600
QRR=60.*60./3.412/1000.*QR 00485610
WRITE(6,500) XTIME,NTREAL,TIME,ITER,RESORM(ITER),SORSUM,QRR 00485700
500 FORMAT(1X,'TIME=',F7.3,' S',1X,'NTREAL=',I9,1X, 00485800
& 'TIME=',F7.2,' <O>',1X,'ITER=',I2,1X,'SOURCE=', 00485900
& F9.6,1X,'SORSUM=',F9.6,1X,' QR(KW) = ',F10.4) 00486000
C 00486100
QKW = ((60.*60.)/(3.412*1000.))* Q 00486200
PRINT * 00486300
PRINT *, ' PCURVE PSOUTH PERROR Q00486400
&CRR QCORRT Q(KW) ' 00486500
PRINT *, PCURVE,PSOUTH,PERROR,QCORR,QCORRT,QKW 00486600
PRINT * 00486700
C 00486800
ELSE IF( NN .EQ. 2 ) THEN 00486900
PRINT * 00487000
PRINT *, ' TEMPERATURES AT THERMOCOUPLE POSITION IN (C)' 00487100
WRITE (6,*) (TCOUP(N),N=1,NTHCO) 00487200
PRINT * 00487300
PRINT * 00487400
00487500
ELSE 00487600
C write(nnn,*)' time=',xtime,'seconds'
C write(nnn,*)' node# u v w'
C write(nnx,*)' time=',xtime,'seconds'
C write(nnx,*)' node# temperature pressure'
DO 502 L=1, nkpl 00487800
K=L 00487900
DO 502 M=1, NIP1 00488000
I=M 00488100
WRITE(6,504) I,K 00488200
504 FORMAT(/,2X,' I=',I2,5X,' K=',I2,/,10X,' T NOD',3X,'R(GM/C.C.)',2X, 00488300
& ' U(CM/SEC)',2X,' V(CM/SEC)',2X,' W(CM/SEC)', ' P (ATM)',5X,' SMP',5X, 00488400
& ' VIS(SEC/CM-CM)',3X,' COND(SEC/CM-CM)', ' XSMP',/) 00488500
513 DO 503 J=1,NJP1 00488600
C XTEMP=T(I,J,K)/CONST3-273.16 00488700
XTEMP=T(I,J,K) 00488800
C XR=R(I,J,K)*RHOO/2.2048 *1000.*(0.0328)**3 00488900
XR=R(I,J,K) 00489000
C XU=U(I,J,K)*CONST6 00489100
C XV=V(I,J,K)*CONST6 00489200
C XW=W(I,J,K)*CONST6 00489300
C XP=(P(I,J,K)*CONST1-REQ(I,J,K)*PINT) 00489400
XP=P(I,J,K) 00489500
XU=U(I,J,K) 00489600
XV=V(I,J,K) 00489700
XW=W(I,J,K-1) 00489800
CC XVIS=VIS(I,J,K)*RHOC*CPO*H*UO*1.48814 00489900

```

```

CC      XCOND=COND(I,J,K)*RHO0*CPO*H*UO*1.48814      00490000
        XVIS=VIS(I,J,K)/VISO      00490100
        XCOND=COND(I,J,K)/VISO      00490200
        XSMP=RI(I,J,K)      00490300
        DDYY=1./FLOAT(NJM1-2)      00490400
        PE =SQRT(U(I,J,K)**2+V(I,J,K)**2+W(I,J,K)**2)*DDYY/COND(I,J,K)      00490500
555     WRITE(nnn,555) i,j,k,xu,xv,xw      00490600
        format('node(',3i3,')',3e12.4)
        write(nnx,556) i,j,k,xtemp,xp
556     format('node(',3i3,')',2(2x,e12.4))
503     CONTINUE      00490900
502     CONTINUE      00491000
        WRITE(6,*) 'THE TIME WHEN THE DATA WAS STORED ON DISK IS:',
& XTIME
        close(nnn)
        close(nnx)      00487700
        ENDIF      00491100
        RETURN      00491200
        END      00491300

```

LIST OF REFERENCES

1. Yang, K.T., J.R. Lloyd, A.M. Kanury, and K. Satoh, "Modeling of Turbulent Buoyant Flows in Aircraft Cabins," *Combustion Science and Technology*, vol. 39, pp. 107-118, 1984.
2. Kou, H.S., K.T. Yang, and J.R. Lloyd, "Turbulent Buoyant Flow and Pressure Variations Around an Aircraft Fuselage in a Cross Wind Near the Ground," *Fire Safety Science--Proceedings of the First International Symposium*, pp. 173-184, 1986.
3. Nicolette, V.F., K.T. Yang, and J.R. Lloyd, "Transient Cooling by Natural Convection in a Two-Dimensional Square Enclosure," *Intl. J. Heat Transfer*, vol. 28, no. 9, pp. 1721-1732, 1985.
4. Aziz, K., and J.D. Hellums, "Numerical Solution of the Three-Dimensional Equation of Motion for Laminar Natural Convection," *Physics of Fluid*, vol. 10, pp. 314-324, 1967.
5. Mallinson, G.D., and G. De Vahl Davis, "Three-Dimensional Numerical Analysis of Transient Natural Convection in a Box a Numerical Study," *J. Fluid Mech.*, vol. 83, pp. 1-31, 1977.
6. Chan, A.M.C., and S. Banerjee, "Three-Dimensional Numerical Analysis of Transient Natural Convection in Rectangular Enclosure," *Int. J. Heat Mass Transfer*, vol. 26, pp. 1427-1438, 1983.
7. Ozeo, H., K. Fujii, N. Lior, and S.W. Churchill, "Long Rolls Generated by Natural Convection in an Inclined, Rectangular Enclosure," *Int. J. Heat Mass Transfer*, vol. 26, pp. 1427-1438, 1983.
8. Morrison, G.L., and V.G. Tran, "Laminar Flow Structure in Vertical Free Convection Cavities," *Int. J. Heat Mass Transfer*, vol. 21, pp. 203-213, 1978.
9. Yang, H.Q., K.T. Yang, and J.R. Lloyd, "Flow Transition in Laminar Flow in a Three-Dimensional Tilted Rectangular Enclosure," *Heat Transfer*, vol. 4, pp. 1495-1500, 1986.

10. Yang, H.Q., K.T. Yang, and J.R. Lloyd, "Laminar Natural Convection Flow Transition in Tilted Three-Dimensional Longitudinal Rectangular Enclosures," *Int. J. Heat Mass Transfer*, vol. 30, no. 8., pp. 1637,1644, 1987.
11. Yang, H.Q., K.T. Yang, and J.R. Lloyd, "Three Dimensional Buoyant Bimodal Flow Transition in Tilted Enclosures," *Int. J. Heat and Fluid Flow*, 1988.
12. Yang, H.Q., *Laminar Buoyant Flow Transitions in Three-Dimensional Tilted Rectangular Enclosures*, Ph.D. Thesis, University of Notre Dame, South Bend, IN, 1987.
13. Ozeo, H., T. Shibata, and S.W. Churchill, "Natural Convection in an Inclined Circular Cylinder Annulus Heated and Cooled in its End Plates," *Int. J. Heat Mass Transfer*, vol. 24, pp. 727-737, 1981.
14. Takata, Y., K. Fukuda, S. Hasegawa, H. Shimomura, and K. Sanokawa, "Three-Dimensional Natural Convection in a Porous Medium Enclosed in a Vertical Curved Annulus," *Numerical Heat Transfer*, pp. 29-39, 1983.
15. Takata, H., K. Iwashige, K. Fukuda, and S. Hasegawa, "Three-Dimensional Natural Convection in an Inclined Cylindrical Annulus," *Int. J. Heat Mass Transfer*, vol. 27, pp. 747-754, 1984.
16. Rao, Y., Y. Miki, K. Fukuda, and Y. Takata, "Flow Patterns of Natural Convection in Horizontal Cylindrical Annuli," *Int. J. Heat Mass Transfer*, vol. 28, pp. 705-714, 1985.
17. Fusegi, T., and B. Farouk, "A Three-Dimensional Study of Natural Convection in the Annulus Between Horizontal Concentric Cylinders," *Proc. 8th Int. Heat Transfer Conf.*, San Francisco, CA, vol. 4, pp. 1575-1580, 1986.
18. Smutek, C., P. Bontoux, B. Roux, G.H. Schiroky, A.C. Hurford, F. Rosenberger, and G. De Vahl Davis, "Three Dimensional Convection in Horizontal Cylinders: Numerical Solutions and Comparison with Experimental and Analytical Results," *Numerical Heat Transfer*, vol. 8, pp. 613-631, 1985.
19. Yang, H.Q., K.T. Yang, and J.R. Lloyd, "A Numerical Study of Three-Dimensional Laminar Natural Convection in a Horizontal Cylinder with Differentially Heated End Walls at High Rayleigh Numbers," *Proceedings of the Symposium on Heat and Mass Transfer Honoring Professor B.T. Chao*, University of Illinois, Urbana, IL, pp. 153-195, 1987.

20. Ozeo, H., K. Fujii, T. Shibata and S.W. Churchill, "Three-Dimensional Numerical Analysis of Natural Convection in a Spherical Annulus," *Numerical Heat Transfer*, vol. 8, pp. 383-406, 1985.
21. Baum, H.R., and R.G. Rehm, "Calculations of Three Dimensional Buoyant Plumes in Enclosures," *Combustion Science and Technology*, vol. 40, pp. 55-77.
22. Baum, H.R., and R.G. Rehm, "Natural Computation of Large Scale Fire-Induced Flows," paper presented at the Eighth Int. Conf. on Numerical Methods in Fluid Dynamics, Aachen, West Germany, 28 June-2 July 1982.
23. Baum, H.R. and R.G. Rehm, "Computation of Fire Induced Flow and Smoke Coagulation," Nineteenth Symposium, Int. of Combustion, The Combustion Institute, Pittsburgh, PA, pp. 921-931, 1982.
24. Rehm, R.G., and H.R. Baum, "The Equations of Motion for Thermally Driven, Buoyant Flows," *J. of Research of the National Bureau of Standards*, vol. 83, no. 3, pp. 297-308, May-June 1978.
25. Bagnaro, M., M. Laouisset, and F.C. Lockwood, "Field Model Prediction of Some Room Fires: Steady and Transient," *Fire Dynamics and Heat Transfer*, ASME HTD, vol. 25, ASME, New York, NY, pp. 107-114, 1983.
26. Markatos, N.C., and K.a. Pericleous, "An Investigation of Three Dimensional Fires in Enclosures," *Fire Dynamics and Heat Transfer*, ASME HTD, vol. 25, ASME, New York, NY, pp. 115-124, 1983.
27. Alexander, J.I., H.J. St. Aubin, J.P. Stone, T.T. Street, and F.W. Williams, "Large-Scale Pressurizable Fire Test Facility -- Fire-1," NRL Report 864, Naval Research Laboratory, Washington, D.C., December 1982.
28. Nies, G.F., *Numerical Field Model Simulation of Full Scale Tests in a Closed Vessel*, Master's and Mechanical Engineer's Thesis, Naval Postgraduate School, Monterey, CA, December 1986.
29. Raycraft, J.K., *Numerical Field Model Simulation of Full Scale Fire Tests in a Closed Spherical/Cylindrical Vessel*, Master's and Mechanical Engineer's Thesis, Naval Postgraduate School, Monterey, CA, December 1987.

30. Houck, R.R., *Numerical Field Model Simulation of Full Scale Fire Tests in a Closed Spherical/Cylindrical Vessel with Internal Ventilation*, Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1988.
31. CA-DISSPLA™ Computer Associates International Inc., Garden City, NY, 1988.
32. Eringen, A.C., *Mechanics of Continua*, John Wiley & Sons, Inc., New York, NY, 1967.
33. Department of Aerospace and Mechanical Engineering, University of Notre Dame, South Bend, IN, Technical Report TR-79002-78-2, "An Algebraic Turbulence Model for Buoyant Recirculating Flow," by V.W. Nee and V.K. Liu, 1978.
34. Sparrow, E.M., and R.D. Cess, *Radiation Heat Transfer*, Hemisphere Publishing Corporation, Washington, D.C., 1978.
35. Department of Aerospace and Mechanical Engineering, University of Notre Dame, South Bend, IN, Technical Report, TR-37191-74-4, "A Numerical Model for the Prediction of Two Dimensional Unsteady Flows of Multicomponent Gases with Strong Buoyancy Effect and Recirculation," by Michael L. Doria, November 1974.
36. Patanker, S.V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Company, New York, NY, 1980.
37. Leonard, S.P., "A Convectively Stable, Third-Order Accurate Finite-Difference Method for Steady Two-Dimensional Flow and Heat Transfer," *Numerical Properties and Methodologies in Heat Transfer*, ed. T.M. Shih, Hemisphere Publishing Corp., Washington, D.C., pp. 211-226, 1983.

INITIAL DISTRIBUTION LIST

	No. of Copies
1. Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2. Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3. Dr. Patricia A. Tatum Code 6183 Naval Research Laboratory Washington, D.C. 20375	2
4. Professor K.T. Yang Department of Aerospace and Mechanical Engineering University of Notre Dame Notre Dame, IN 46556	2
5. Professor M.D. Kelleher, Code ME/Kk Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 939443-5004	2
6. Chairman, Department of Mechanical Engineering Code ME Naval Postgraduate School Monterey, CA 93943-5004	1
7. Naval Engineering Curricular Officer, Code 34 Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 93943-5000	1
8. Mr. James Gagorik Code 211, ONT Balston Towers #1 800 N. Quincy Arlington, VA 22217	2
9. LT Timothy McCarthy FSC 473 Box 8-28 FPO AP 96349-1400	1

Thesis
M16652
c.1

McCarthy

Numerical field model
simulation of full-scale
fire tests in a closed
spherical/cylindrical
vessel using advanced
computer graphics tech-
niques.

Thesis
M16652
c.1

McCarthy

Numerical field model
simulation of full-scale
fire tests in a closed
spherical/cylindrical
vessel using advanced
computer graphics tech-
niques.

DUDLEY KNOX LIBRARY



3 2768 00018374 3