1988

Numerical field model simulation of full-scale fire tests in a closed spherical/cylindrical vessel with internal ventilation.

Houck, Richard Reid.

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

NUMERICAL FIELD MODEL SIMULATION OF FULL-SCALE FIRE TESTS IN A CLOSED SPHERICAL/CYLINDRICAL VESSEL WITH INTERNAL VENTILATION

by

Richard Reid Houck

September 1988

Thesis Co-Advisors: M. D. Kelleher
K. T. Yang

Jnclassified

ecurity Classification of this page

REPORT DOCUMENTATION PAGE

a Report Security Classification Unclassified	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report
2b Declassification/Downgrading Schedule	Approved for public release; distribution is unlimited.
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization: Naval Postgraduate School — 6b Office Symbol *(If Applicable)* 69	7a Name of Monitoring Organization: Naval Postgraduate School
6c Address *(city, state, and ZIP code)*: Monterey, CA 93943-5000	7b Address *(city, state, and ZIP code)*: Monterey, CA 93943-5000
8a Name of Funding/Sponsoring Organization — 8b Office Symbol *(If Applicable)*	9 Procurement Instrument Identification Number
8c Address *(city, state, and ZIP code)*	10 Source of Funding Numbers: Program Element Number / Project No / Task No / Work Unit Accession No

11 Title *(Include Security Classification)* Numerical Field Model Simulation of Full-Scale Fire Tests in a Closed Spherical/Cylindrical Vessel With Internal Ventilation

12 Personal Author(s) Richard R. Houck

13a Type of Report	13b Time Covered	14 Date of Report *(year, month,day)*	15 Page Count
Master's Thesis	From To	1988 September	232

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes

Field	Group	Subgroup

18 Subject Terms *(continue on reverse if necessary and identify by block number)* Field Model; Fire Simulation; Fire Modeling; Numerical Fire Model; Fires in Closed Vessels; Enclosed Fires

19 Abstract *(continue on reverse if necessary and identify by block number*

Shipboard fires have plagued mariners for centuries; they still cause significant damage and casualties each year. Improved fire prevention and control require a sound knowledge of the phenonema of fire. At the same time, a study of fires in enclosed pressure vessels has been undertaken by the Navy using FIRE–1, a large pressure vessel, to conduct full-scale experimental fires. A computer model is being developed to simulate the FIRE–1 tests. This three-dimensional finite difference model uses a cylindrical/spherical coordinate system and includes the effects of turbulence, surface and flame radiation, internal ventilation, global and local pressure corrections, strong buoyancy, and conjugate boundary conditions. Given a heat release rate, the model computes temperature, pressure, density and velocity fields for the entire vessel. This thesis presents the internal ventilation feature of the model and compares the numerical results to a nonventilated case. Additional features such as combustion and gaseous radiation are being incorporated to more accurately model real fires. When validated, this model will become a useful tool for evaluating fire prevention and control procedures and equipment.

20 Distribution/Availability of Abstract: [X] unclassified/unlimited [ ] same as report [ ] DTIC users	21 Abstract Security Classification: Unclassified
22a Name of Responsible Individual: Professor M. D. Kelleher	22b Telephone *(Include Area code)* (408) 696–2530 — 22c Office Symbol 69Kk

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted — security classification of this page

All other editions are obsolete — Unclassified

Numerical Field Model Simulation of Full-Scale Fire Tests in a Closed Spherical/Cylindrical Vessel With Internal Ventilation

by

Richard Reid Houck
Lieutenant Commander, U.S. Coast Guard
B.S., United States Coast Guard Academy, 1975
MBA, Pepperdine University, 1979

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1988

ABSTRACT

Shipboard fires have plagued mariners for centuries; they still cause significant damage and casualties each year. Improved fire prevention and control require a sound knowledge of the phenonema of fire. At the same time, a study of fires in enclosed pressure vessels has been undertaken by the Navy using FIRE-1, a large pressure vessel, to conduct full-scale experimental fires. A computer model is being developed to simulate the FIRE-1 tests. This three-dimensional finite difference model uses a cylindrical/spherical coordinate system and includes the effects of turbulence, surface and flame radiation, internal ventilation, global and local pressure corrections, strong buoyancy, and conjugate boundary conditions. Given a heat release rate, the model computes temperature, pressure, density and velocity fields for the entire vessel. This thesis presents the internal ventilation feature of the model and compares the numerical results to a nonventilated case. Additional features such as combustion and gaseous radiation are being incorporated to more accurately model real fires. When validated, this model will become a useful tool for evaluating fire prevention and control procedures and equipment.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

A	Area
A	Finite Difference Coefficients
ARU_	Source Term Variable (Eqn. 3.74)
AU_	Source Term Variable (Eqn. 3.73)
C_	Coefficients for Control Volume __ (Eqn. 3.40, 3.64)
C_M	Coefficients for Control Volume __ (Eqn. 3.43)
C_P	Coefficients for Control Volume __ (Eqn. 3.43)
COND_1	Coefficients for Control Volume __ (Eqn. 3.42)
C_{pm}	Mean Isobaric Heat Capacity
CURV	Curvature Term (Eqns. 3.25–3.26)
CURVN	Orthogonal Curvature Term (Eqns. 3.30–3.31)
F_{Ai-Aj}	View Factor for Radiation Emitted by Surface i and Incident upon Surface j (Eqn. 2.38)
G	Gravitational Acceleration
G	Mass Flux Rate (Eqns. 3.8–3.13)
G	Term Used in Radiation Model (Eqn. 2.35)
g	Curvilinear Base Vector
g_i	Scaling Term (Eqn. 2.8)
g_{ij}	Covariant Metric Tensor (Eqn. 2.16)
g^{ij}	Contravariant Metric Tensor (Eqn 2.17)
H	Mixing Length Parameter (Eqn. 2.31)
h	Scale Factor
h	Convective Heat Transfer Coefficient

J	Total Heat Flux (Eqn. 3.19–3.21)
K	Adjustable Constant (used in Eqn. 2.31)
k	Thermal Conductivity
M	Momentum Flux (Eqn. 3.55)
m	Rate of Change (Eqn. 3.5)
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 (Eqn. 3.71)
RR_	Source Term Variable (Eqn. 3.75)
Ri	Richardson Number (Eqn. 2.30)
r	Distance between Two Surfaces
S_f	Source Term (Eqn. 2.25)
S_{hs}	Heat Source
S_{mp}	Mass Source Term
T	Temperature
t	Time
u	Velocity
V	Volume
VIS	Local Viscosity (Eqn. 3.65)
X	Length in X-Direction (In QUICK Scheme)

GREEK LETTERS

β	Angles Formed by Radiation Surface Normals
χ	Term Used in Radiation Model (Eqn. 2.37)
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 (Eqn. 2.36)

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 (Eqns. 3.98–3.103)
′	Correction
^	Prior Value

I. INTRODUCTION

A. BACKGROUND

Fires aboard ships pose a great hazard to both personnel and materiel. Millions of dollars are spent annually on repairs of damage due to fires. Personnel casualties caused by fires cannot be measured in dollars and include both fatalities and severe injuries. Most personnel casualties result from toxic gas or smoke inhalation rather than contact with the fire. The prevention and control of shipboard fires is one of the Navy's and Coast Guard's greatest challenges in future ship design. The computer simulation of a shipboard fire presented in this thesis provides a tool which may be used to reduce the damage from shipboard fires.

In order to prevent fires and their associated casualties, it is necessary to better understand the basic phenomena of fire and smoke propagation within enclosed spaces. This requires knowledge of various physical phenomena: combustion, fluid mechanics, and heat and mass transfer. Extensive research using this basic knowledge is needed to predict the behavior of fires. With a better understanding of fires, ship designers and engineers can reduce the probability of ignition and propagation. New systems and procedures for fire control can be developed to reduce the losses should a fire start due to accident, equipment failure, or hostile action.

Shipboard fires have unique complexities not found in other fire scenarios. Access to a fire area is limited and spaces frequently contain electronic equipment, electrical power sources, machinery, combustibles, or toxic materials. Compartments are often closed, permitting pressure to build up in the space. Self-contained or recirculating ventilation systems present unusual fire scenarios. All of these complications must be considered in the study of shipboard fires; the model developed in this thesis has incorporated two of these complexities: pressure build-up and recirculating ventilation.

Shipboard fire research is currently being conducted by many organizations, including the Navy and the Coast Guard. Research includes both experimental work and computer modeling. Experimental work is limited due to its high cost. Scale models of fires do not predict the behavior of full-scale fires because of the complexity of the fire phenomena. It thus becomes necessary to conduct fire research with full-scale testing. At the Naval Research Laboratory in Washington, D. C., the U.S. Navy built FIRE–1, a large pressure vessel designed to simulate fires aboard submarines and surface ships. This unique test facility offers the researcher an opportunity to study a fire with the pressure building up in the vessel. This models a fire in a submarine or in a closed compartment on a surface ship.

Today's supercomputers, with their extremely rapid computational speed and massive storage capability, offer a researcher the option of computer modeling of fires. The systems of partial differential equations which govern the fire phenomena can now be solved numerically. The first models were simple, but current models are building on the older models, incorporating more phenomena and producing more accurate results. As each new submodel (such as a combustion or gas radiation model) is added, the quality of the numerical solutions improves. The models are being verified by comparison with actual fires, such as those conducted in FIRE-1.

When validated, computer models provide an excellent tool for the fire researcher. In experiments, each test must be repeated many times to verify the procedures, test facility, and data. The cost of these experiments becomes prohibitive. Experimental researchers must determine which test scenarios will produce the most meaningful results and how to design the data collection systems and procedures to monitor the most critical parameters. This is one aspect in which computer fire simulations become invaluable. By developing a code which accurately simulates a fire in FIRE-1, various fire scenarios can be modeled at a reasonable cost. The most interesting scenarios can then be investigated by experiments in FIRE-1.

Computer models may also be used in modeling fires which cannot be tested in full scale due to the size and geometry limitations of FIRE-1. An entire area of a ship might be modeled and the progress of the fire within and between compartments could be investigated. With such simulations, the spread of fire could be analyzed, and new methods can be evaluated to prevent the spread of fire from compartment to compartment. Additionally, the efficacy of fire extinguishing systems can be evaluated by introducing models of these systems into the fire model. All of these future uses require a validated code and the use of a large computer. While the cost of a computer model test is significantly less than a full-scale test, it still requires extensive computer time. The current code running on an IBM 3033 uses approximately 1.5 CPU hours per second of fire time. A supercomputer and vectorization could reduce this time by one or two orders of magnitude, but the number of model tests needed to fully validate the code still will require significant supercomputer resources.

B. COMPUTER MODELING

There are two basic procedure for modeling fires: field and zone modeling. Zone modeling involves dividing the fire area into control volumes or distinct regions [Ref. 1]. Each region contains a phenomena of particular interest, such as the base of the fire, fire plume, heating of the wall, ventilation inlet or outlet duct, etc. Mass and energy balances are conducted across the boundaries and interconnect all of the control volumes. This procedure provides information for the entire area, but the phenomena occurring within each control volume are not always understood.

Field modeling, also known as differential field equation modeling, divides the compartment into finite volume elements. The conservation equations in differential form are used to calculate the mass, momentum, energy, and smoke concentration at each time interval. The temperature, velocity, pressure, density, and smoke concentration are known in each volume element. Models for additional physical effects, such as turbulence, forced ventilation, and different geometry (such as equipment or decks) can be included in a field model to better simulate actual fires. Field modeling requires a large, fast computer with significantly more memory than zone modeling. The accuracy of the solution depends upon reducing the size of the control volumes; this increases the number of individual cells, the size of the problem, and the computing expense.

Much fire research has been conducted to provide a solid foundation for this thesis. Work performed at the University of Notre Dame [Refs. 2, 3] included a two-dimensional finite difference field model of aircraft fires. It predicted the movement of hot gases and smoke as well as temperature and smoke concentration levels in the seating area of an aircraft cabin. Additional work by Nicolette, et al. [Ref. 4] included the development of a two-dimensional model of transient cooling by natural convection. This model utilized a fully transient semi-implicit upwind differencing scheme with a global pressure correction. Experimental data showed good agreement with the numerical predictions.

Recent studies [Refs. 5 through 13] have developed numerical solutions for natural convection in three-dimensional rectangular enclosures using field modeling. They successfully solved nonlinear partial differential equations with a finite difference method. Models and studies involving three-dimensional cylindrical coordinate buoyant flows [Refs. 14 through 20] deal primarily with horizontal cylindrical annuli that have walls of different temperatures. Smutek, et al. [Ref. 19] studied convection in a horizontal cylinder with differentially heated ends at low Rayleigh numbers. Yang, et al. [Ref. 20] conducted a similar numerical study for high Rayleigh numbers.

The difficulty in calculating pressure has been addressed using methods that eliminate pressure from the governing equations. Stream function-vorticity methods [Refs. 14 through 19] have been used to solve natural convection problems in several geometries. The problems inherent in this method include instability at moderate to high Rayleigh numbers, difficulties in handling three-dimensional situations, and the lack of pressure information, which often is a parameter of interest. These problems are addressed by Yang, et al. [Ref. 20], who propose the use of primitive variables with an arbitrary orthogonal coordinate system.

Ozoe, et al. [Ref. 21] used a vorticity vector potential formulation and alternating-direction-implicit finite difference method to compute velocity and temperature fields for three-dimensional natural convection in a spherical annulus.

Baum and Rehm [Refs. 22 through 25] have developed several field models for prediction of fires. Their models use time-dependent inviscid Boussinesq equations to simulate three-dimensional buoyant convection and smoke aerosol coagulation. Field models have also been used to model room fires [Ref. 26] and fires in a general three-dimensional enclosure [Ref. 27].

The numerical method developed by Yang, et al. [Ref. 20] and used in this thesis is based upon the use of primitive variable finite difference discretization in generalized orthogonal coordinates. This method has the ability to handle complex geometries and the stability inherent in the primitive variable formulation.

C. FIRE–1, THE TEST FACILITY

To better understand the phenomena of fire inside a pressurized compartment, the Navy built an experimental pressure vessel for conducting test fires. This test facility is designated FIRE–1 and is located at the Naval Research Laboratory in Washington, D. C. A brief summary of FIRE–1 is contained here; a more detailed report is provided by Alexander, et al. [Ref. 28]. Figure 1.1 shows the basic layout of FIRE–1. It is a 46.6-foot-long cylindrical vessel with hemispherical ends, capable of pressures up to 89.7 psi at 450 F. The radius of both the cylinder and the end caps is 9.6 feet and the total enclosed volume is 11,639 cubic feet. The vessel is constructed of 3/8 inch ASTM 285 Grade C steel and contains rupture discs at each end to prevent overpressurization.

Instrumentation monitors various fire parameters, including pressures, temperatures, and smoke concentrations. Pressure transducers and bourdon tube gauges are located at the north and south ends of the vessel. Thermocouples and radiometers are installed as shown in Fig. 1.2. An array of ten thermocouples is located at each end of the tank. Each thermocouple is a chrome alumel wire of 0.2 mm diameter having ceramic insulation enclosed in 1 mm diameter Type 304 stainless steel jackets. Thermocouples are also located on the chamber wall to measure the inside and outside wall temperatures.

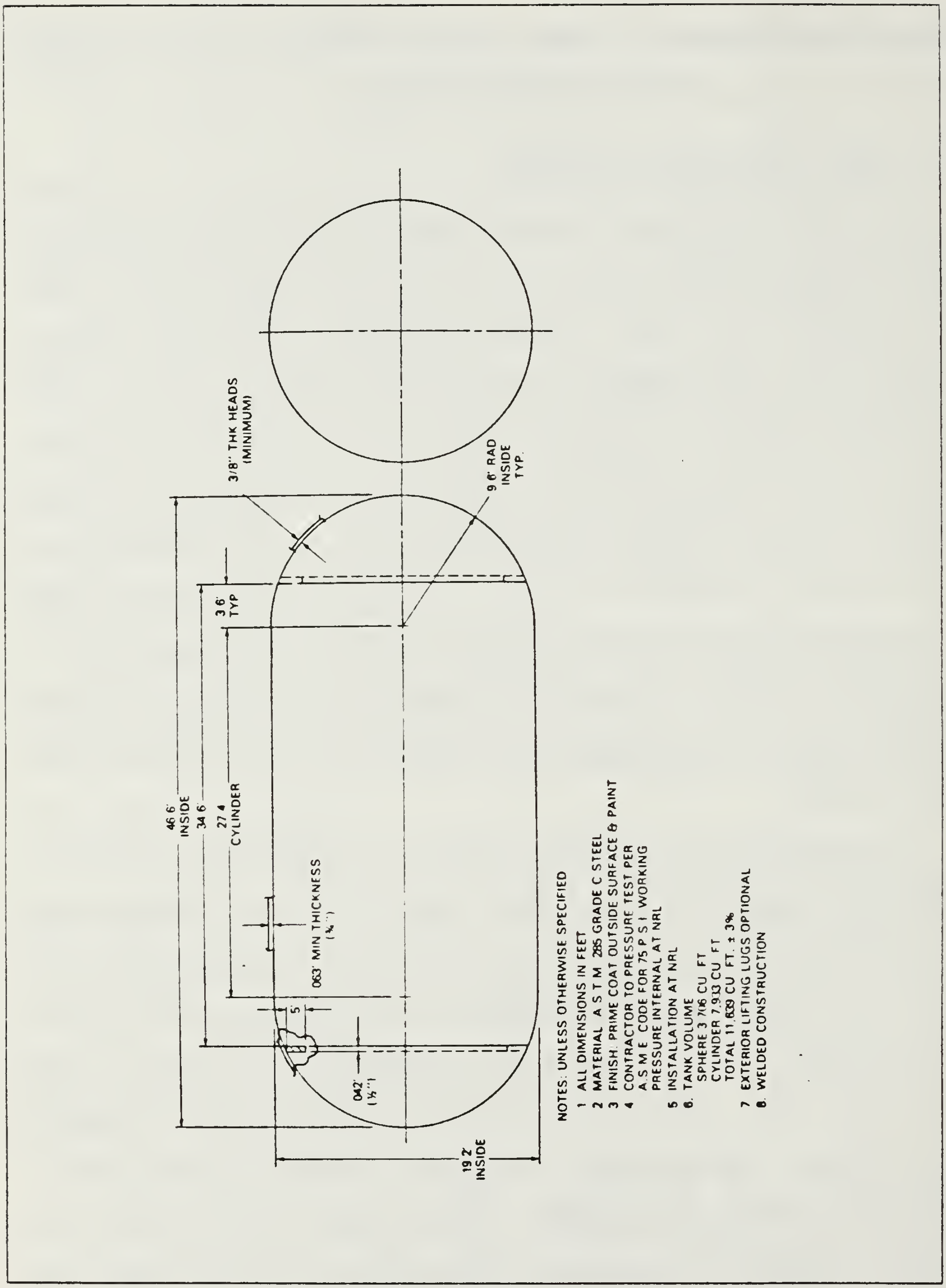

Figure 1-1. **Drawing of the FIRE–1 Test Vessel**

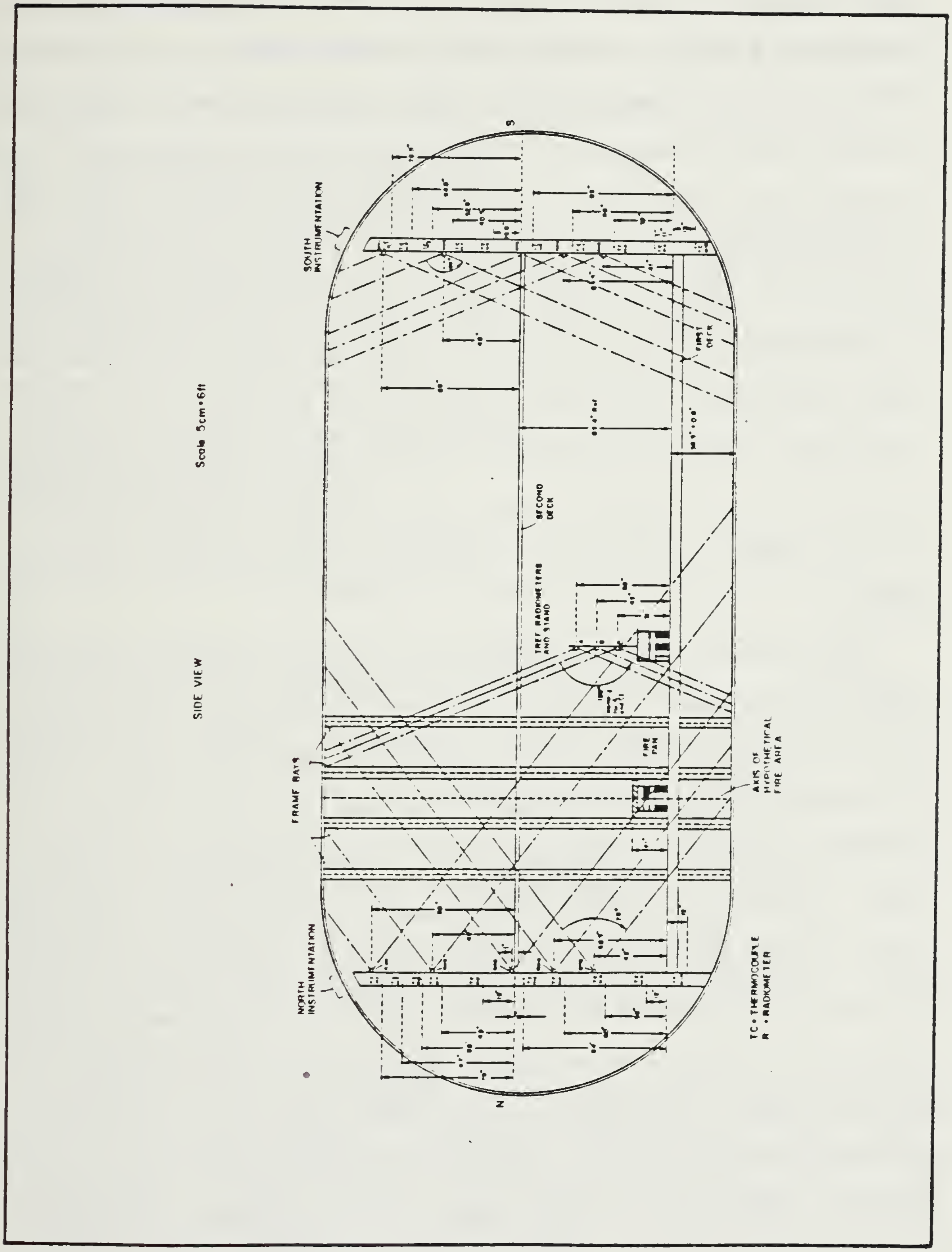

Figure 1-2. **Side View of FIRE-1 With Sensor Locations**

Additional thermocouples and radiometers are available for temporary installation at various locations as required for different tests. Smoke obscuration can be measured three ways: visual obscuration with video cameras, particle analysis, and obscuration with laser detectors. The fuel burn rate is determined with a round tapered-edge fire pan with various cross-sectional areas, provided with a constant-level fuel supply system. The operation and calibration is described by Alexander, et. al. [Ref 28]. To date, the burn rate data has not been accurate, so further experimentation is necessary to provide fuel burn rate. As discussed later, the lack of accurate burn rate data precludes complete verification of the computer code. In the interim, several methods of deducing burn rate have been developed for use in the computer model.

Three features permit modification of the tests to more accurately model the submarine or ship compartment being tested. First, there are two removable decks, one installed in the mid plane of the vessel and the other slightly over three feet above the bottom. Grated or solid deck plates can be installed to test various shipboard configurations. The decks have been incorporated in the computer model but have not yet been tested and verified. Second, a nitrogen pressurization system extinguishes the fire and can be used to evaluate its performance in an actual fire situation. Ten seconds after energizing the nitrogen system, the pressure in the vessel rises to two atmospheres and extinguishes the fire by reducing the partial pressure of oxygen to less than 10.5 percent. Third, there are two ventilation fans which can be installed to simulate the effects of internal ventilation. The ventilation system has been included in the computer model and is the subject of verification in Chapter 4 of this thesis.

D. FEATURES OF THE PROGRAM

The computer model was developed as a low-cost alternative to predict the spread of fire and smoke in enclosed spaces on naval vessels. Together with the FIRE-1 test facility, which can be used for validation of the computer code, it can be used to evaluate the effectiveness of damage control systems and new ship designs in the prevention and control of fires.

The computer model is a joint effort of the University of Notre Dame and the Naval Postgraduate School. The original work by Nies [Ref. 29] involved a model of a rectangular volume similar to FIRE-1. The model was a three-dimensional, finite difference model employing primitive variables. It included a global pressure correction, surface radiation, turbulence, and simple conduction at the walls. The unreliability of the burn rate data from FIRE-1 experiments caused a problem in validation of the computer model. To overcome this problem, a scheme for developing the burn rate based on the experimental pressure was developed; the procedure is describe by Nies [Ref. 29:pp. 61–63].

Raycraft [Ref. 30] developed a more sophisticated model which uses a spherical/cylindrical coordinate system to more accurately model FIRE-1. It also includes a more detailed formulation of surface radiation, global pressure correction, turbulence, and conduction. The problem with burn rate data persisted, and three trials were run to attempt to better simulate the burn rate. The conclusions were:

1. The pressure tracking case, Trial 1, provided a numerically generated heat release curve from other available sources. The pressure was forced to follow the experimental curve, causing large oscillations in the heat release and temperature data.

2. Trial 2 used a third-order polynomial fit of the experimental data provided by NRL. The pressure and temperature did not oscillate greatly, but the values obtained were very high. This indicated that experimental burn rate data was also too high. It was known at the onset that the heat release rate data could be off by some unknown scaling factor.

3. Of the three test cases examined, Trial 3 was a better representation of the fire in FIRE–1. This case combined the heat release rate levels obtained from Trial 1 with the third-order polynomial fit variation from Trial 2. The results were a realistic burn rate curve to use as input into the computer code. [Ref. 30]

The present code includes internal forced ventilation into the model. The effects of two fans blowing into the end caps of the vessel is investigated in this thesis using the burn rate curve discussed above in Conclusion 3. The results are compared with existing data of the fire model without ventilation.

E. THESIS OUTLINE

This thesis describes the numerical model, its derivation, and application. In Chapter 2, the governing equations, initial and boundary conditions, and the various submodels employed are discussed. Chapter 3 presents the derivation of the finite difference equations. The use of the control volume method in the spherical/cylindrical geometry is explained. The conservation equations are presented and integrated, finite difference equations are developed, and the pressure correction procedures are described. Chapter 4 presents the experimental data for the internal ventilation model and compares it with the nonventilated case. The conclusions and recommendations for future work are presented in Chapter 5. The appendix contains the code for the model.

II. NUMERICAL MODEL

A. GOVERNING EQUATIONS

1. Introduction

The governing differential equations used in the computer model are described in this section. They are initially presented for a Cartesian system and then transformed into a generalized curvilinear coordinate system using standard tensor notation. Several assumptions are made in the development of the governing equations. The fire is modeled as an unsteady volumetric heat source that is a third order polynomial in time, which resulted from previous work [Ref. 30]. The effects of combustion have not yet been incorporated into the code. Density varies in accordance with the perfect gas law.

Nies [Ref. 29] developed a computer code to model a fire in FIRE-1 using Cartesian coordinates as an initial approximation. Raycraft [Ref. 30] describes the code for the current spherical/cylindrical geometry which is summarized below.

2. General Equations

The governing equations include: conservation of mass (continuity), conservation of momentum, conservation of energy, and the equations of state. These are presented below in Cartesian coordinates and in standard tensor notations. The continuity equation is:

$$\rho,_t + (\rho u_i),_i = 0 \tag{2.1}$$

The momentum equation is given as:

$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} = -P_{,i} - \rho G_i + (\sigma_{ij})_{,j} \qquad (2.2)$$

and the energy equation is:

$$(\rho C_{pm} T)_{,t} + (\rho u_i C_{pm} T)_{,i} = (k T_{,j})_{,i} + \mu\Phi + P u_{i,i} \qquad (2.3)$$

The stress tensor is given as:

$$\sigma_{ij} = \mu_{eff}\left(u_{i,j} + u_{j,i} - \tfrac{2}{3}\delta_{ij} u_{k,k}\right) \qquad (2.4)$$

with δ_{ij} being the Kronecker delta, which equals the value of 1 when $i = j$ and equals the value of 0 when $i \neq j$. The dissipation function is:

$$\Phi = 2\left(u_{i,j}^2\right)\delta_{ij} + \left[u_{i,j}\left(1 - \delta_{i,j}\right)\right]^2 - \tfrac{2}{3}\left(u_{i,i}\right)^2 \qquad (2.5)$$

The equations of state are given as:

$$P = \rho R T \qquad (2.6)$$

$$h = C_{pm}(T - T_R) \qquad (2.7)$$

Since the computer model of FIRE-1 is in a combination of spherical and cylindrical coordinates, these equations must be transformed into a general curvilinear coordinate system (θ^1, θ^2, θ^3). Yang, et. al. [Ref. 20] outlines this process, using the rules established by Eringen [Ref.

31]. The generalized orthogonal coordinates are transformed as follows:

$$X_i \rightarrow \theta^i \tag{2.8}$$

with a scale factor, h_i, for curvilinear coordinates given as:

$$h_i = \sqrt{\vec{g}_i \cdot \vec{g}_i} = \sqrt{\left(\frac{\partial X_j}{\partial \theta^i}\right) \cdot \left(\frac{\partial X_j}{\partial \theta^i}\right)} \tag{2.9}$$

The scale factor is a component, therefore the summation rule does not apply to the subscript of h_i. Reference 31 gives the scale factors in cylindrical coordinates as:

$$h_1 = r = \theta^2 \tag{2.10}$$

$$h_2 = 1 \tag{2.11}$$

$$h_3 = 1 \tag{2.12}$$

In spherical coordinates, the scale factors are:

$$h_1 = r \sin \theta = \theta^2 \sin \theta^3 \tag{2.13}$$

$$h_2 = 1 \tag{2.14}$$

$$h_3 = r = \theta^2 \tag{2.15}$$

The covariant and contravariant metric tensors of orthogonal coordinates are given as:

$$g_{ij} = \vec{g}_i \cdot \vec{g}_j = \delta_{ij}\, h_i\, h_j \tag{2.16}$$

$$g^{ij} = \frac{\delta_{ij}}{h_i\, h_j} \tag{2.17}$$

The vector tangent to the u_i curve at P is given as:

$$u_i = \frac{g_{ij}\; u^{(j)}}{h_j} \tag{2.18}$$

and the velocity vector is given as:

$$u^i = \frac{u^{(i)}}{h_i} \tag{2.19}$$

In generalized orthogonal coordinates [Ref. 20], the continuity equation is:

$$\rho_t + \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left(\sqrt{g}\; \rho\; \frac{u^i}{h_i} \right) = 0 \tag{2.20}$$

and the energy equation becomes:

$$\left(\rho\, C_{pm}\, T\right)_t + \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left(\sqrt{g}\; \rho\; C_{pm}\; u^i\; \frac{T}{h_i} \right) = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left(\sqrt{g}\; \frac{k_{eff}\, T_{,i}}{h_i^2} \right) + S_f \tag{2.21}$$

with the momentum equation given as:

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

$$+ \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^j}\left(\frac{\sqrt{g}\, \sigma^j_i}{h_j}\right) - \frac{1}{h_i h_j} \frac{\partial h_i}{\partial \theta^j} \left(\rho u^i u^j - \sigma^j_i \right) + \tag{2.22}$$

$$+ \frac{1}{h_i h_j} \frac{\partial h_j}{\partial \theta^i} \left(\rho u^j u^i - \sigma^j_j \right)$$

The stress tensor is:

$$\sigma^j_i = \mu_{eff} \left[\frac{h_j}{h_i} \frac{\partial}{\partial \theta^i} \left(\frac{u^j}{h_j}\right) + \frac{h_i}{h_j} \frac{\partial}{\partial \theta^j} \left(\frac{u^i}{h_i}\right) + \frac{\delta_{ij}}{h_i h_j} \frac{\partial q_{ii}}{\partial \theta^m} \left(\sqrt{g}\, \frac{u^m}{h_m}\right) \right] \tag{2.23}$$

and the dissipation function is:

$$\Phi = 2\left[\left(\frac{u^i}{h_i}\right)_{,j}^2\right] \delta_{ij} + \left[\left(\frac{u^i}{h_i}\right)_{,j} (1 - \delta_{ij})\right]^2 - \frac{2}{3}\left[\left(\frac{u^i}{h_i}\right)_{,i}\right]^2 \tag{2.24}$$

The only difference between these equations and the cartesian coordinate equations is the additional terms in the momentum equation for Coriolis and centrifugal forces. In the energy equation, several terms have been lumped together in the source term:

$$S_f = \mu \Phi + P \frac{1}{\sqrt{g}} \frac{\partial}{\partial \theta^i} \left(\sqrt{g} \frac{u^i}{h_i} \right) + S_{hs} \tag{2.25}$$

The heat source term, S_{hs}, is nonzero only in the fire, since gas radiation effects have yet to be incorporated into the computer model. Furthermore, since the present study deals with turbulent flow, the conductivity, k_{eff}, and dynamic viscosity, μ_{eff}, are the effective quantities which include both the laminar and turbulent contributions.

B. INITIAL AND BOUNDARY CONDITIONS

In order to solve the governing equations, both initial and boundary conditions must be applied to the model.

1. Initial Conditions

The initial conditions of the model are the same as the conditions immediately prior to the ignition of the fire in FIRE-1. The air within the vessel is assumed to be totally at rest, so the entire velocity field is set equal to zero. The forced ventilation does not begin until the fire starts, so that the velocity field due to the forced ventilation builds as the fire starts to burn. The temperature of the field is uniform and equal to the ambient temperature, which corresponds to a nondimensional temperature of 1.0. Pressure and density distributions are due to the static equilibrium distribution inside the tank.

2. Boundary Conditions

The pressure vessel forms a solid wall around the entire area, so all velocities on the wall are zero; this satisfies the no-slip condition. Since there is no mass flux through the wall, all velocities normal to the wall are set equal to zero. Temperatures on the inside of the wall are equal to the temperature of the fluid immediately adjacent to the wall eliminating temperature discontinuities. The following equations describe these boundary conditions.

$$u^i = 0 \tag{2.26}$$

$$T_{surf} = T_{fluid} \tag{2.27}$$

Continuity of heat flux must be met at the walls.

$$q_r - k_f \frac{\partial T}{\partial n} = - k_s \frac{\partial T_s}{\partial n} \tag{2.28}$$

with n representing the normal direction towards the center of the vessel and q_r representing the thermal radiation energy. There is heat conduction through the walls and heat convection from the exterior walls to the environment at the ambient temperature.

Due to the cylindrical and spherical geometry, there is a singularity at a radius of zero which requires special treatment. Several different methods of correcting this problem are discussed by Yang, et al. [Ref. 20:pp. 167–168]. The method chosen for this model involves applying continuity to two consecutive radial control volumes placed in the vicinity of radius equal to zero. Of all the methods investigated, this was found to give the best representation of the flow and temperature flow fields.

The boundary conditions for the control volumes adjacent to the ventilation control volumes are discussed in Chapter 3.

C. PHYSICAL MODELS

1. Turbulence Model

An algebraic model is used to predict the average values of the dependent variables. More complicated models could be used, but the increase in computing time does not warrant their use. Nee and Liu [Ref. 32] developed a model that obtains the effective viscosity, μ_{eff}, in recirculating buoyant flows with large variations in turbulence levels. The equation, after being transformed to the generalized orthogonal coordinate system, is:

$$\frac{\mu_{eff}}{\mu_o} = 1 + \frac{\left(\frac{1}{H}\right)^2 \sqrt{\left(\frac{1}{h_j}\frac{\partial u^i}{\partial \theta^j}\right)^2 \left(1 - \delta_i^j\right)}}{2 + \frac{Ri}{Pr_t}} \tag{2.29}$$

where Pr_t is the turbulent Prandtl Number and the Richardson Number, Ri, is given as:

$$Ri = \frac{H}{u_i^2} \quad \frac{\left(\frac{\partial T}{\partial n}\right) \vec{n} \cdot \vec{g}}{\left[\left(\frac{\partial u^1}{\partial n}\right)\vec{n} \cdot \vec{g}\right]^2 + \left[\left(\frac{\partial u^2}{\partial n}\right)\vec{n} \cdot \vec{g}\right]^2 + \left[\left(\frac{\partial u^3}{\partial n}\right)\vec{n} \cdot \vec{g}\right]^2} \tag{2.30}$$

with $\vec{n}$ a unit vector in the direction opposite to gravity and 1/H the nondimensional mixing length parameter:

$$\frac{1}{H} = K \left\{ \frac{\sqrt{u^i u^i}}{\sqrt{\sum_{i,j}\left(\frac{1}{h_j}\frac{\partial u^i}{\partial \theta^j}\right)^2}} + \frac{\sqrt{\sum_{i,j}\left(\frac{1}{h_j}\frac{\partial u^i}{\partial \theta^j}\right)^2}}{\sqrt{\sum_{i,j}\left(\frac{1}{h_i h_j}\frac{\partial^2 u^i}{\partial \theta^i \partial \theta^j}\right)^2}} \right\} \qquad (2.31)$$

where K is an adjustable constant. The effective conductivity is defined by the following equation:

$$k_{eff} = \frac{1}{Pr} + \frac{1}{Pr_t}\frac{\mu_{eff}}{\mu_o} \qquad (2.32)$$

2. Conduction Model

As the fire progresses, the heat energy transferred to the environment becomes increasingly important. This requires a model for the heat conduction through the vessel walls. The energy transfer is treated as unsteady, one-dimensional heat conduction through the wall and convection with a constant heat transfer coefficient at the wall exterior. The energy equation in this case is:

$$(\rho_s C_{ps} T)_{,t} = \frac{1}{\sqrt{g}}\frac{\partial}{\partial \theta^i}\left(\sqrt{g}\, k_s T_{,j}\, g^{ij}\right) + S \qquad (2.33)$$

with $\rho_s C_{ps}$ being the heat capacitance of the wall and k_s being the conductivity of the wall.

3. Radiation Model

The radiation model is described in detail by Raycraft [Ref. 30:pp. 22–44] but is summarized below. The radiation model used is based on three assumptions. First, the model only considers surface radiation; this means that the gas and smoke inside the tank is considered to be transparent. Second, all surfaces are modeled as grey surfaces, with radiation diffusely distributed. Third, the tank walls and the flame of the fire are treated as surfaces. The radiation model is based on the net radiosity model discussed by Sparrow and Cess [Ref. 33]. The net rate of heat loss per unit area is given as:

$$\frac{Q_i}{A_i} = \sum_{j=1}^{N} G_{ij}\,\sigma\,T_j^4 \tag{2.34}$$

with the following definitions:

$$G_{ij} = \frac{\varepsilon_i}{1-\varepsilon_i}\,(\delta_{ij} - \Psi_{ij}) \tag{2.35}$$

$$\Psi_{ij} = \chi_{ij}^{-1} \tag{2.36}$$

$$\chi_{ij} = \frac{\delta_{ij} - (1-\varepsilon_i)\,F_{Ai-Aj}}{\varepsilon_i} \tag{2.37}$$

F_{Ai-Aj} is the view factor for the radiation emitted by the surface i and incident upon surface j. Generally, it is given as

$$F_{Ai-Aj} = \frac{1}{A_i}\int_{A_i}\int_{A_j} \frac{\cos\beta_i \cos\beta_j \, dA_i \, dA_j}{\pi r^2} \tag{2.38}$$

The view factor calculations are given in detail by Raycraft [Ref. 30:pp. 29 through 44].

4. Internal Ventilation Model

The internal ventilation model allows the user to set up forced internal ventilation in the field. This would normally represent outlets of the ship's ventilation system, but could also model ventilation due to damage (i.e., ruptured air lines or ventilation ducts) or damage control smoke removal equipment. The internal ventilation model defines a velocity in one or more control volumes.

III. FINITE DIFFERENCE EQUATIONS AND CALCULATIONS

A. INTRODUCTION

The numerical solution for the computer model has space and time as the independent variables, and velocity (in three directions), pressure, temperature, and density as the dependent variables. With six unknown dependent variables, six equations are needed to obtain a solution. The conservation of mass equation (Eqn. 2.20), conservation of energy equation (Eqn. 2.21), conservation of momentum equations (Eqn. 2.22), and the equation of state (Eqn. 2.6) are used. These equations are discretized in a method similar to that described by Doria [Ref. 34], based on the general discretization concept presented by Patankar [Ref. 35]. Doria divided the domain into separate control volumes and wrote conservation equations for each cell in an integral form. These integral equations became a set of finite difference equations which could be solved to provide a solution.

In the flow field, each cell is treated as a unit, with one value of each property reigning throughout the cell. The center of the cell determines the value of temperature, pressure and density. The velocity grids are staggered one-half cell away from the center. Patankar [Ref. 35:pp. 115–120] describes two problems which arise when the velocity cells are coincident with the basic cells. First, the velocity at the staggered cell center is calculated as a function of the pressure differential between the two adjacent nonstaggered cells. If the cells were not staggered, the velocity would be calculated based on the pressures of adjacent cells, which are twice as far away as in the staggered cell case. Second, staggered cells preclude unrealistic oscillating solutions.

Employment of primitive variables presents a problem with the coupling of the pressure term in different equations. Others have used the stream function to eliminate this coupling [Refs. 14–19] but in the present case, with the desire to determine the pressure, this method is inappropriate for the reasons cited in Chapter 1. In the computer code, an iterative procedure is used to estimate pressure. To ensure that the results are physically realistic, a numerical method must not violate the conservation properties. Patankar [Ref. 35:pp.120–126] and Doria [Ref. 34:pp.26–32] describe the method of satisfying conservation by correcting the estimated pressure to ensure that mass is conserved at every cell. In addition to the local pressure correction, a global pressure correction is included to account for the total energy change in the system, as described by Nicolette, et al. [Ref. 4].

In the finite difference method, differential elements are replaced by finite quantities in the integral form of the equations. Many methodologies have been developed for dealing with the differencing techniques and each has inherent features and problems. The QUICK methodology (Quadratic Upstream Interpolation for Convective Kinematics) developed by Leonard [Ref. 36] is used here for the convective terms. QUICK uses locally two-dimensional quadratic interpolation functions for estimating control volume face values and gradients of transported variables. It is third-order accurate and permits practical grid sizes. Yang [Ref. 13] employed QUICK in the coupled momentum, energy, and pressure equation solutions for three-dimensional flow in tilted rectangular enclosures.

B. CONTROL VOLUME

When defining the problem to be solved numerically, the flow field is divided up into finite elements, or cells that together make up the entire field. At the center of each cell is a grid point that is defined as the governing point of the cell. In discussing the grid points, the following nomenclature is used. The grid of interest is called P (I, J, K), with adjacent grids being defined 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 of the cell with grid point P are designated by lower case letters, or e, w, n, s, f, and b. Figures 3.1 and 3.2 shows typical cells in cylindrical and spherical coordinate systems.

As previously discussed, velocities are defined in a staggered grid system. To illustrate this, Figure 3.3 shows a two-dimensional cell; Figure 3.4 shows the location of the staggered velocities around the grid. The velocity, u^1_i, for the basic cell is located on the west face; u^2_J is on the south face; and u^3_k (not shown) is on the back face. In all cases, the staggered cell system is offset one-half cell from the primary cell system.

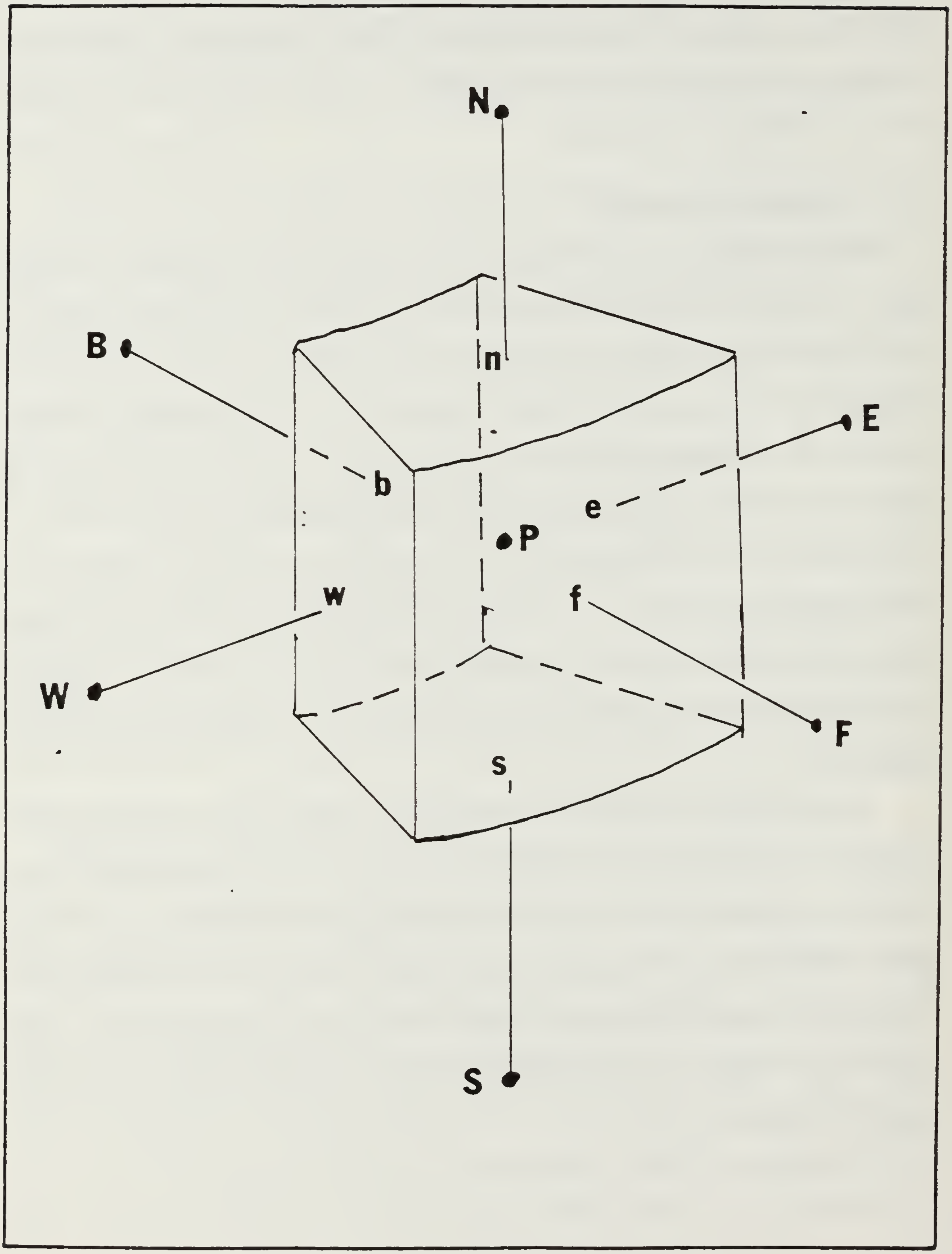

Figure 3-1. **Basic Cylindrical Cell**

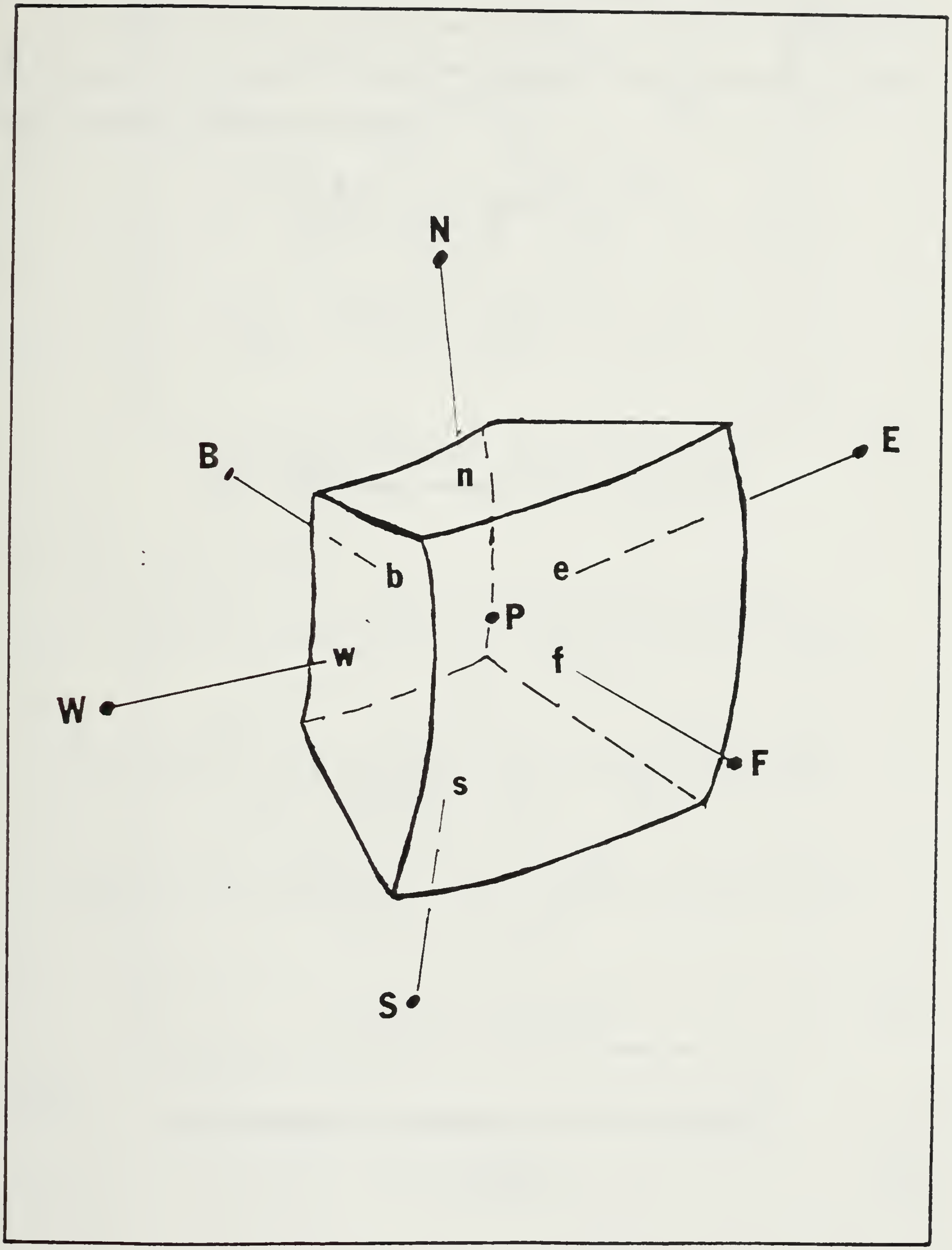

Figure 3-2. **Basic Spherical Cell**

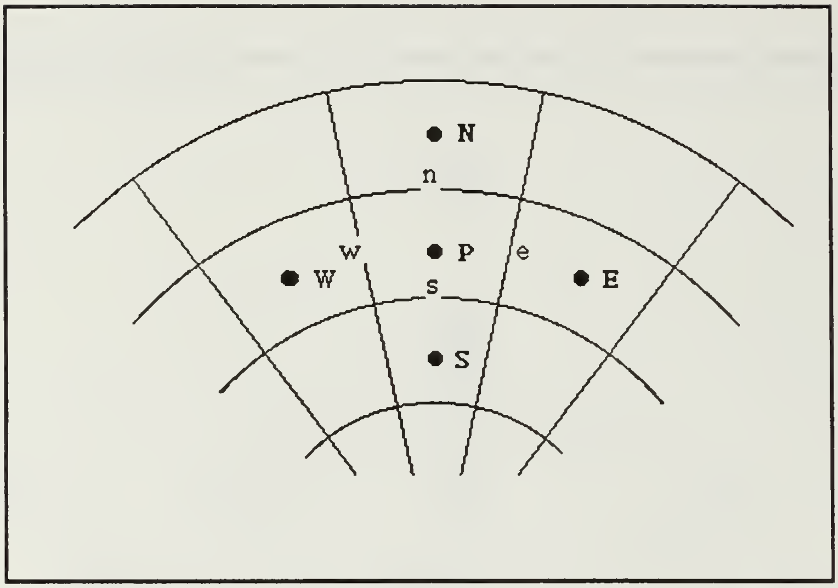

Figure 3-3. **Two-Dimensional Basic Cell**

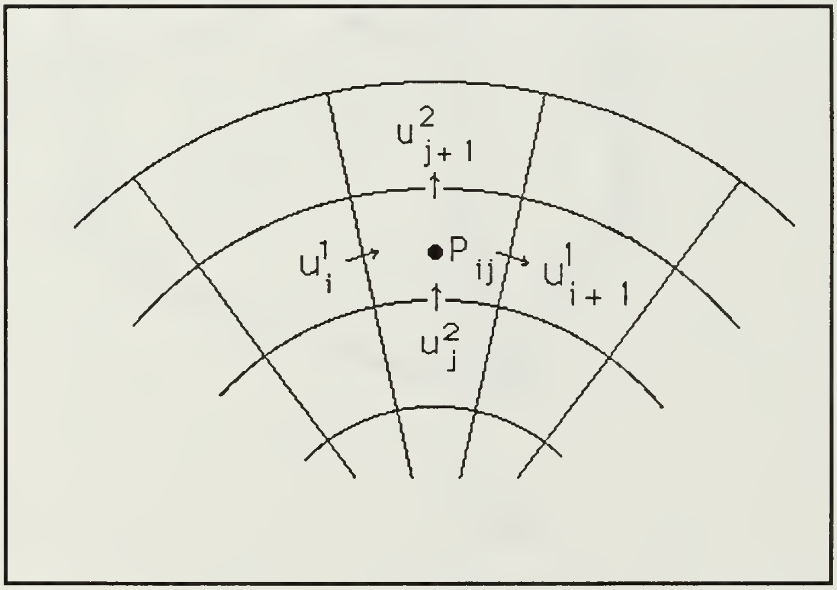

Figure 3-4. **Two-Dimensional Staggered Cell**

C. INTEGRATION OF CONSERVATION EQUATIONS

To discretize the conservation equations, it is first necessary to put them into an integral form by integrating over the volume of a cell. The continuity equation becomes:

$$\int \frac{\partial \rho}{\partial t} h_1 h_2 h_3 \, d\theta^1 \, d\theta^2 \, d\theta^3 +$$

$$\int \Big[\frac{\partial}{\partial \theta^1} (\rho u^1 h_2 h_3) + \frac{\partial}{\partial \theta^2} (\rho u^2 h_3 h_1) +$$

$$+ \frac{\partial}{\partial \theta^3} (\rho u^3 h_1 h_2) \quad d\theta^1 \, d\theta^2 \, d\theta^3 = 0 \tag{3.1}$$

and the energy equation is:

$$\int \frac{\partial (\rho C_{pm} T)}{\partial t} h_1 h_2 h_3 \, d\theta^1 \, d\theta^2 \, d\theta^3 + \int \Big[\frac{\partial}{\partial \theta^1} (\rho C_{pm} u^1 T h_2 h_3) +$$

$$\frac{\partial}{\partial \theta^2} (\rho C_{pm} u^2 T h_1 h_3) + \frac{\partial}{\partial \theta^3} (\rho C_{pm} u^3 T h_1 h_2) \Big] d\theta^1 \, d\theta^2 \, d\theta^3 \quad -$$

$$\int \Big[\frac{\partial}{\partial \theta^1} (q^1 h_2 h_3) + \frac{\partial}{\partial \theta^2} (q^2 h_1 h_3) + \frac{\partial}{\partial \theta^3} (q^3 h_1 h_2) \Big] d\theta^1 \, d\theta^2 \, d\theta +$$

$$\int S h_1 h_2 h_3 \, d\theta^1 \, d\theta^2 \, d\theta^3 \tag{3.2}$$

with:

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

The momentum equations are:

$$\int \frac{\partial}{\partial t} (\rho\, u^i)\, h_1\, h_2\, h_3\, d\theta^1\, d\theta^2\, d\theta^3 + \int \frac{\partial}{\partial \theta^j} \left[\left(\frac{h_1\, h_2\, h_3}{h_j} \right) \rho\, u^i\, u^j \right] d\theta^1\, d\theta^2\, d\theta^3$$

$$= \int - \frac{\partial}{\partial \theta^i} \left(P \frac{h_1\, h_2\, h_3}{h_i} \right) d\theta^1\, d\theta^2\, d\theta^3 + \int \rho\, G_i\, h_1\, h_2\, h_3\, d\theta^1\, d\theta^2\, d\theta^3$$

$$+ \int \frac{\partial}{\partial \theta^j} \left(\sigma^{ij} \frac{h_1\, h_2\, h_3}{h_i\, h_j} \right) d\theta^1\, d\theta^2\, d\theta^3$$

$$- \int \frac{h_1\, h_2\, h_3}{h_i\, h_j} \cdot \left[\frac{\partial h_i}{\partial \theta^j} (\rho\, u^j\, u^i - \sigma^{ij}) \right] d\theta^1\, d\theta^2\, d\theta^3$$

$$+ \int \frac{h_1\, h_2\, h_3}{h_j\, h_i} \cdot \left[\frac{\partial h_j}{\partial \theta^i} (\rho\, u^j\, u^j - \sigma^{jj}) \right] d\theta^1\, d\theta^2\, d\theta^3 \qquad (3.4)$$

D. CONTINUITY EQUATION

Once the governing equations have been integrated, the differential elements are replaced with finite quantities. Three separate differencing methods are used in the computer model: forward differencing for time, central differencing for the diffusion terms, and QUICK for the convection terms.

In forward differencing, the future value of a given parameter is found by adding its present value to the net change over a finite time. This change is described by the rate of change (slope) multiplied by the time step. For example,

$$\rho^n = \rho^{n-1} + m\, \Delta t \tag{3.5}$$

with ρ^{n-1} representing the present value of density, m is the rate of change, ρ^n is the future value, and Δt is the time step. Substituting this into the continuity equation (3.1) results in:

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

By evaluating the integral, the continuity equation becomes:

$$(\rho^n - \rho^{n-1})\frac{\Delta V}{\Delta t} + \left[\rho\, u^1\, h_2\, h_3\, d\theta^2\, d\theta^3\right]_e - \left[\rho\, u^1\, h_2\, h_3\, d\theta^2\, d\theta^3\right]_w$$

$$+ \left[\rho\, u^2\, h_1\, h_3\, d\theta^1\, d\theta^3\right]_n - \left[\rho\, u^2\, h_1\, h_3\, d\theta^1\, d\theta^3\right]_s + \tag{3.7}$$

$$+ \left[\rho\, u^3\, h_1\, h_2\, d\theta^1\, d\theta^2\right]_f - \left[\rho\, u^3\, h_1\, h_2\, d\theta^1\, d\theta^2\right]_b = 0$$

The mass flux rate, G, is evaluated at each of the six cell faces:

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

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

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

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

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

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

with the area of the faces given as:

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

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

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

In the finite difference format, the continuity equation becomes:

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

with S_{mp} defined as the mass source term. In an analytical solution, this mass source term is zero, but in numerical solutions it is a finite nonzero term. Through iteration, the numerical solution converges and the mass source term approaches zero. Instead of converging to zero, the source term is set equal to zero when it is less than or equal to 10^{-70}.

E. ENERGY EQUATION

The integrated energy equation is:

$$\left[(\rho C_{pm} T)^{n} - (\rho C_{pm} T)^{n-1}\right]\frac{\Delta V}{\Delta t} + G_e (C_{pm} T)_e A_e - G_w (C_{pm} T)_w A_w +$$

$$G_n (C_{pm} T)_n A_n - G_s (C_{pm} T)_s A_s + G_f (C_{pm} T)_f A_f - G_b (C_{pm} T)_b A_b =$$

$$= k_e A_e \left(\frac{\partial T}{h_1 \partial \theta^1}\right)_e - k_w A_w \left(\frac{\partial T}{h_1 \partial \theta^1}\right)_w + k_n A_n \left(\frac{\partial T}{h_2 \partial \theta^2}\right)_n -$$

$$- k_s A_s \left(\frac{\partial T}{h_2 \partial \theta^2}\right)_s - k_f A_f \left(\frac{\partial T}{h_3 \partial \theta^3}\right)_f + k_b A_b \left(\frac{\partial T}{h_3 \partial \theta^3}\right)_b + S_f \Delta V \qquad (3.18)$$

where all k's represent effective values. S_f is the source term and includes dissipation, radiation, pressure work, and all internal heat sources. J is the total heat flux resulting from convection and conduction.

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

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

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

These equations are the θ^1, θ^2, and θ^3 components of the total heat flux. The subscripts refer to the face to which they correspond. The term $(\rho\, C_{pm}\, u^1\, T)$ causes problems since u is evaluated at the cell surface, but all other values are evaluated at the cell center. Because of this, when using these equations, the fluxes must be expressed in terms of C_{pm}, ρ, and T at the point P and its neighbors. Substituting these equations into the integrated energy equation, the finite difference energy equation is:

$$\left[(\rho\, C_{pm}\, T)^{n} - (\rho\, C_{pm}\, T)^{n-1} \right] \frac{\Delta V}{\Delta t} + J^{1}_{e} A_{e} - J^{1}_{w} A_{w} +$$

$$+ J^{2}_{n} A_{n} - J^{2}_{s} A_{s} + J^{3}_{f} A_{f} - J^{3}_{b} A_{b} = S\ \Delta V \quad (3.22)$$

Of the many finite differencing methods, the QUICK scheme is used with the convective terms because it accurately predicts the dependent variable values at the control volume surfaces with stable properties. QUICK has the relative accuracy of the central differencing scheme coupled with the stability of an upwind scheme. It uses a parabolic polynomial interpolation to fit the control volume at three adjacent nodes. Yang [Ref.13:pp. 77–89] describes QUICK in one, two, and three dimensions. Raycraft [Ref. 30:pp. 63–74] developed the finite difference energy equations using the QUICK scheme. Since this method is used in the computer model, the derivation is repeated here.

The quadratic interpolation for a nonuniform grid is given as:

$$(\rho\, C_{pm}\, u\, T)_e = G_e\, C_{pm.e}\left[\left(\frac{T_P + T_e}{2}\right) - \frac{1}{8}\, CURV_e\right] \tag{3.23}$$

$$(\rho\, C_{pm}\, v\, T)_w = G_w\, C_{pm.w}\left[\left(\frac{T_P + T_w}{2}\right) - \frac{1}{8}\, CURV_w\right] \tag{3.24}$$

Figure 3.5 shows the one-dimensional scheme. The upstream weighted curvature terms CURV are:

$$CURV_e = \frac{\Delta X_e^2}{\Delta X_i}\left(\frac{T_E - T_P}{\Delta X_e} - \frac{T_P - T_W}{\Delta X_w}\right) \quad \text{if } G_e > 0$$

$$= \frac{\Delta X_e^2}{\Delta X_{i+1}}\left(\frac{T_{EE} - T_E}{\Delta X_{ee}} - \frac{T_E - T_P}{\Delta X_e}\right) \quad \text{if } G_e < 0 \tag{3.25}$$

$$CURV_w = \frac{\Delta X_w^2}{\Delta X_{i+1}}\left(\frac{T_P - T_W}{\Delta X_w} - \frac{T_W - T_{WW}}{\Delta X_{ww}}\right) \quad \text{if } G_w > 0$$

$$= \frac{\Delta X_w^2}{\Delta X_i}\left(\frac{T_E - T_P}{\Delta X_e} - \frac{T_P - T_W}{\Delta X_w}\right) \quad \text{if } G_w < 0 \tag{3.26}$$

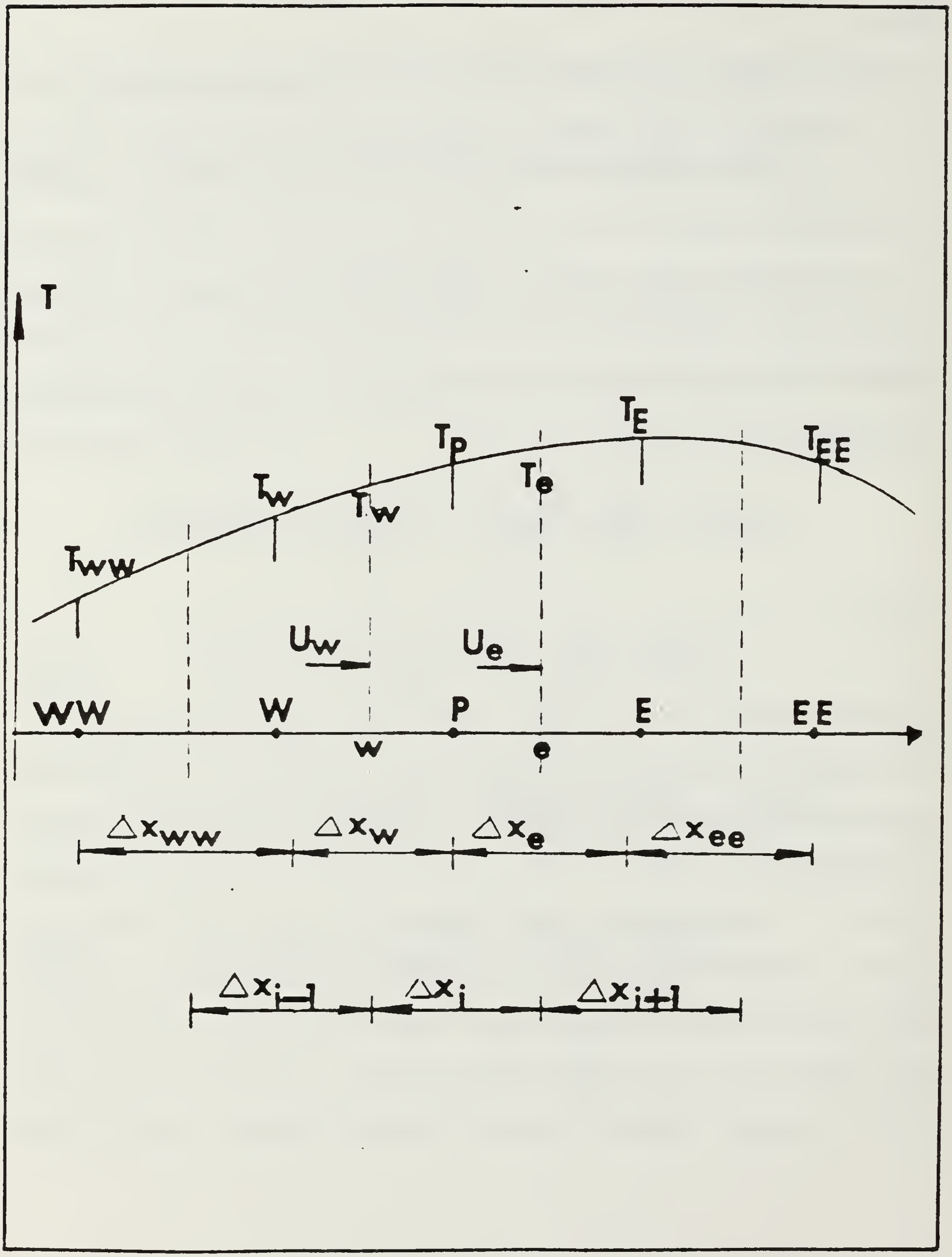

Figure 3-5. **One-Dimensional Quadratic Interpolation Scheme**

with

$$\Delta X_e = 0.5\,(\Delta X_i + \Delta X_{i+1})$$

$$\Delta X_w = 0.5\,(\Delta X_i + \Delta X_{i-1})$$

$$\Delta X_{ee} = 0.5\,(\Delta X_{i+1} + \Delta X_{i+2}) \tag{3.27}$$

$$\Delta X_{ww} = 0.5\,(\Delta X_{i-1} + \Delta X_{i-2})$$

In generalized orthogonal coordinates, the equations becomes:

$$\left(\rho\, C_{pm}\, u^1 T\right)_e = G_e\, C_{pm\,.e} \left(\frac{T_P + T_E}{2} - \frac{1}{8}\, CURVN_e \right) \tag{3.28}$$

$$\left(\rho\, C_{pm}\, u^2 T\right)_w = G_w\, C_{pm\,.w} \left(\frac{T_P + T_W}{2} - \frac{1}{8}\, CURVN_w \right) \tag{3.29}$$

with

$$CURVN_e = \frac{(h_1\, \Delta\theta^1)_e^2}{(h_1\, \Delta\theta^1)_i} \left(\frac{T_E - T_P}{(h_1\, \Delta\theta^1)_e} - \frac{T_P - T_W}{(h_1\, \Delta\theta^1)_w} \right) \quad \text{if } G_e > 0$$

$$= \frac{(h_1\, \Delta\theta^1)_e^2}{(h_1\, \Delta\theta^1)_{i+1}} \left(\frac{T_{EE} - T_E}{(h_1\, \Delta\theta^1)_{ee}} - \frac{T_E - T_P}{(h_1\, \Delta\theta^1)_e} \right) \quad \text{if } G_e < 0 \tag{3.30}$$

$$CURVN_w = \frac{(h_1\, \Delta\theta^1)_w^2}{(h_1\, \Delta\theta^1)_{i+1}} \left(\frac{T_P - T_W}{(h_1\, \Delta\theta^1)_w} - \frac{T_W - T_{WW}}{(h_1\, \Delta\theta^1)_{ww}} \right) \quad \text{if } G_w > 0$$

$$= \frac{(h_1\, \Delta\theta^1)_w^2}{(h_1\, \Delta\theta^1)_i} \left(\frac{T_E - T_P}{(h_1\, \Delta\theta^1)_e} - \frac{T_P - T_W}{(h_1\, \Delta\theta^1)_w} \right) \text{ if } G_w < 0 \qquad (3.31)$$

and

$$(h_1\, \Delta\theta^1)_e = 0.5 \left[(h_1 \Delta\theta^1)_i + (h_1 \Delta\theta^1)_{i+1} \right]$$

$$(h_1\, \Delta\theta^1)_w = 0.5 \left[(h_1 \Delta\theta^1)_i + (h_1 \Delta\theta^1)_{i-1} \right]$$

$$(h_1\, \Delta\theta^1)_{ee} = 0.5 \left[(h_1 \Delta\theta^1)_{i+1} + (h_1 \Delta\theta^1)_{i+2} \right] \qquad (3.32)$$

$$(h_1\, \Delta\theta^1)_{ww} = 0.5 \left[(h_1 \Delta\theta^1)_{i-1} + (h_1 \Delta\theta^1)_{i-2} \right]$$

The conventional finite difference form of Eqn. 3.22 for a one-dimension system is:

$$\left[(\rho\, C_{pm}\, T)^n - (\rho\, C_{pm}\, T)^{n-1} \right] h_1 \frac{\Delta V}{\Delta t} =$$

$$= A_E T_E + A_W T_W - A_P T_P + S(h_1\, \Delta\theta^1) \qquad (3.33)$$

Using a semi-implicit tri-diagonal solution procedure, both T_{EE} and T_{WW} are included in the source term. The remaining coefficients are:

$$A_E = \frac{C_{pm.e}(-7\, G_e + 3\, |G_e|)}{16} + C_{pm.w}(-G_w + |G_w|) + \frac{k_e}{h_1 \Delta\theta^1} \qquad (3.34)$$

$$A_W = \frac{C_{pm.w}(9\, G_w + 3\, |G_w|)}{16} + C_{pm.e}(G_e + |G_e|) + \frac{k_w}{h_1 \Delta\theta^1} \qquad (3.35)$$

$$A_p = \frac{9}{16}\left(G_w C_{pm.w} - G_e C_{pm.e}\right) + 3\left(|G_w| C_{pm.w} + |G_e|\right) + \frac{k_w + k_e}{h_1 \Delta\theta^1} \quad (3.36)$$

$$S_p = S\, h_1\, \Delta\theta^1 - C_{pm.e}\left(|G_e| - G_e\right) T_{EE} - C_{pm.w}\left(|G_w| + G_w\right) T_{WW} \quad (3.37)$$

The three-dimensional QUICK algorithm uses locally quadratic interpolation of temperature through each control volume. Figure 3.6 shows the calculation cell for a three-dimensional uniform rectangular grid. The cylindrical/spherical grid system used in the computer model is more complex, although conceptually the same. Yang [Ref. 13] discusses the evaluation of the curvilinear and temperature terms. Basically, curvature terms are calculated for each of the temperatures and substituted for the convective heat flux terms. Heat flux is calculated and substituted into Eqn. 3.22.

After separation of variables, the energy equation becomes:

$$\left[A_P^T + \left(\rho\, C_{pm.p}\right)^{n-1}\right] \frac{\Delta V}{\Delta t} T_P = A_E^T T_E + A_W^T T_W + A_N^T T_N + A_S^T T_S + A_F^T T_F + A_B^T T_B + S_u^T \quad (3.38)$$

with the additional source term,

$$S_u^T = \left(\rho\, C_{pm.p}\, T\right)^{n-1} \frac{\Delta V}{\Delta t} - A_{EER} + A_{WWR} + A_{NNR} + A_{SSR} + A_{FFR} + A_{BBR} \quad (3.39)$$

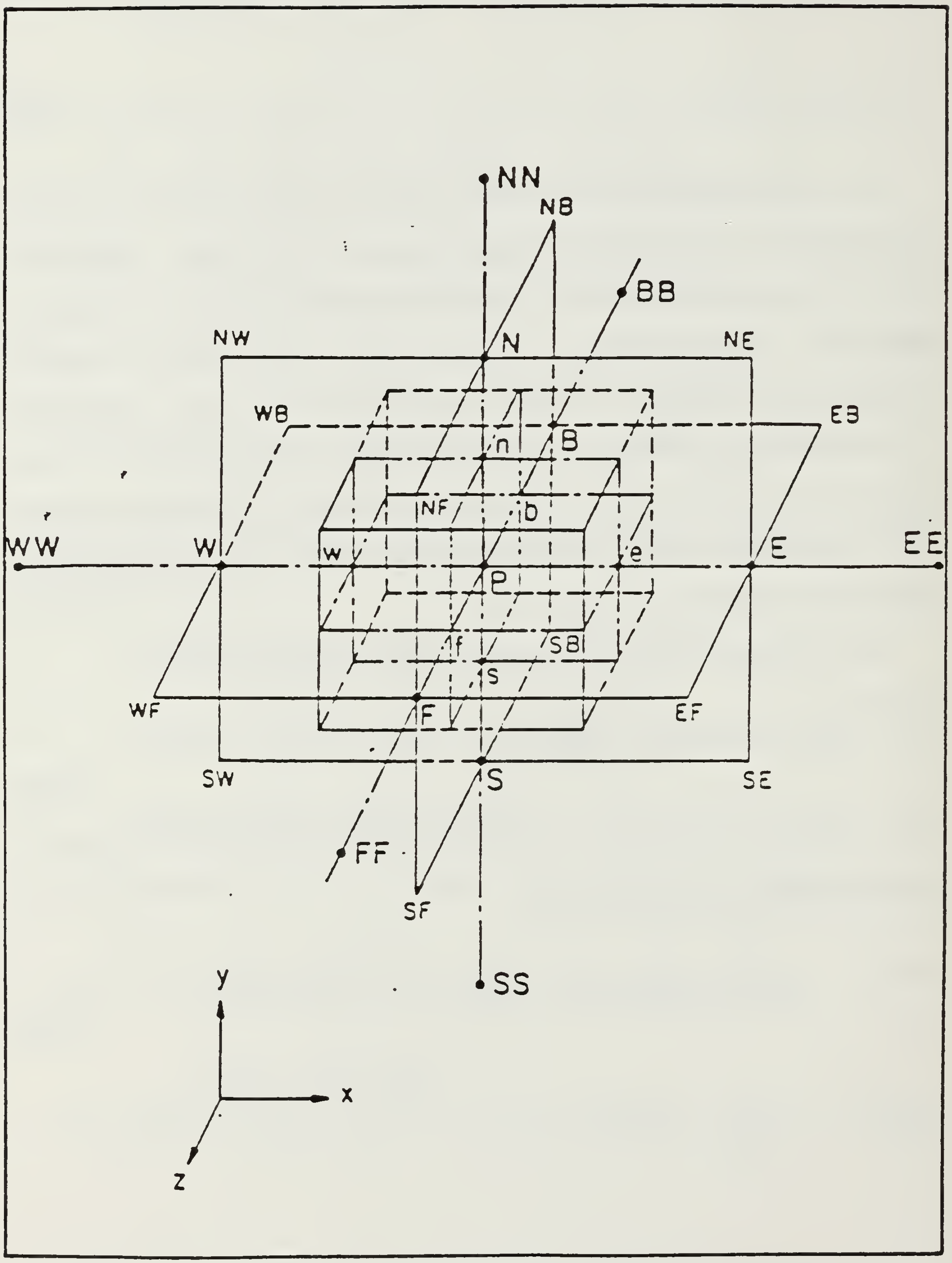

Figure 3-6. **Calculation Cell for a Uniform Rectangular Grid**

$$CN = G_n * u^2_{j+1} * \left(h_3 \Delta\theta^3\right)_n \left(h_1 \Delta\theta^1\right)_n$$

$$CS = G_s * u^2_j * \left(h_3 \Delta\theta^3\right)_s \left(h_1 \Delta\theta^1\right)_s$$

$$CE = G_e * u^1_{i+1} * \left(h_3 \Delta\theta^3\right)_e \left(h_2 \Delta\theta^2\right)_e \tag{3.40}$$

$$CW = G_w * u^1_i * \left(h_3 \Delta\theta^3\right)_w \left(h_2 \Delta\theta^2\right)_w$$

$$CF = G_f * u^3_{k+1} * \left(h_1 \Delta\theta^1\right)_f \left(h_2 \Delta\theta^2\right)_f$$

$$CB = G_b * u^3_k * \left(h_1 \Delta\theta^1\right)_b \left(h_2 \Delta\theta^2\right)_b$$

Thermal conductivity is:

$$k_n = \frac{1}{\dfrac{\dfrac{1}{k_j * \left(h_2\, \Delta\theta^2\right)_j} + \dfrac{1}{k_{j+1} * \left(h_2\, \Delta\theta^2\right)_{j+1}}}{\left(h_2\, \Delta\theta^2\right)_j + \left(h_2\, \Delta\theta^2\right)_{j+1}}} \tag{3.41}$$

$$k_s = \frac{1}{\dfrac{\dfrac{1}{k_j * \left(h_2\, \Delta\theta^2\right)_j} + \dfrac{1}{k_{j-1} * \left(h_2\, \Delta\theta^2\right)_{j-1}}}{\left(h_2\, \Delta\theta^2\right)_j + \left(h_2\, \Delta\theta^2\right)_{j-1}}}$$

$$k_e = \frac{1}{\dfrac{\dfrac{1}{k_i * \left(h_1\, \Delta\theta^1\right)_i} + \dfrac{1}{k_{i+1} * \left(h_1\, \Delta\theta^1\right)_{i+1}}}{\left(h_1\, \Delta\theta^1\right)_i + \left(h_1\, \Delta\theta^2\right)_{i+1}}}$$

$$k_w = \frac{1}{\dfrac{\dfrac{1}{k_i * \left(h_1\,\Delta\theta^1\right)_i} + \dfrac{1}{k_{i-1} * \left(h_1\,\Delta\theta^1\right)_{i-1}}}{\left(h_1\,\Delta\theta^1\right)_i + \left(h_1\,\Delta\theta^2\right)_{i-1}}}$$

$$k_f = \frac{1}{\dfrac{\dfrac{1}{k_k * \left(h_3\,\Delta\theta^3\right)_k} + \dfrac{1}{k_{k+1} * \left(h_3\,\Delta\theta^3\right)_{k+1}}}{\left(h_3\,\Delta\theta^3\right)_k + \left(h_3\,\Delta\theta^3\right)_{k+1}}}$$

$$k_b = \frac{1}{\dfrac{\dfrac{1}{k_k * \left(h_3\,\Delta\theta^3\right)_k} + \dfrac{1}{k_{k-1} * \left(h_3\,\Delta\theta^3\right)_{k-1}}}{\left(h_3\,\Delta\theta^3\right)_k + \left(h_3\,\Delta\theta^3\right)_{k-1}}}$$

$$CONDN1 = k_n * \left(\frac{h_3\,\Delta\theta^3 * h_1\,\Delta\theta^1}{h_2\,\Delta\theta^2}\right)_n$$

$$CONDS1 = k_s * \left(\frac{h_3\,\Delta\theta^3 * h_1\,\Delta\theta^1}{h_2\,\Delta\theta^2}\right)_s$$

$$CONDE1 = k_e * \left(\frac{h_3\,\Delta\theta^3 * h_2\,\Delta\theta^2}{h_1\,\Delta\theta^1}\right)_e \tag{3.42}$$

$$CONDW1 = k_w * \left(\frac{h_3\,\Delta\theta^3 * h_2\,\Delta\theta^2}{h_1\,\Delta\theta^1}\right)_w$$

$$\mathrm{CONDF\,1} = k_f * \left(\frac{h_1 \Delta\theta^1 * h_2 \Delta\theta^2}{h_3 \Delta\theta^3} \right)_f$$

$$\mathrm{CONDB\,1} = k_b * \left(\frac{h_1 \Delta\theta^1 * h_2 \Delta\theta^2}{h_3 \Delta\theta^3} \right)_b$$

In equations (3.41) and (3.42), all k's are the effective values.

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

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

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

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

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

$$\mathrm{CNM} = \frac{|CN| - CN}{16} \frac{(h_2 \Delta\theta^2)_n}{(h_2 \Delta\theta^2)_{j+1}} \qquad (3.43)$$

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

$$CSM = \frac{|CS| - CS}{16} \frac{\left(h_2\, \Delta\theta^2\right)_s}{\left(h_2\, \Delta\theta^2\right)_{j}}$$

$$CFP = \frac{|CF| + CF}{16} \frac{\left(h_3\, \Delta\theta^3\right)_f}{\left(h_3\, \Delta\theta^3\right)_{k}}$$

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

$$CBP = \frac{|CB| + CB}{16} \frac{\left(h_3\, \Delta\theta^3\right)_b}{\left(h_3\, \Delta\theta^3\right)_{k-1}}$$

$$CBM = \frac{|CB| + CB}{16} \frac{\left(h_3\, \Delta\theta^3\right)_b}{\left(h_3\, \Delta\theta^3\right)_{k}}$$

$$A^T_{EE} = \frac{-\,CEM * \left(h_1\, \Delta\theta^1\right)_e}{\left(h_1\, \Delta\theta^1\right)_{ee}}$$

$$A^T_{WW} = \frac{-\,CWP * \left(h_1\, \Delta\theta^1\right)_w}{\left(h_1\, \Delta\theta^1\right)_{ww}}$$

$$A^T_{NN} = \frac{-CNM * \left(h_2\, \Delta\theta^2\right)_n}{\left(h_2\, \Delta\theta^2\right)_{nn}}$$

$$A^T_{SS} = \frac{-CSP * \left(h_2\, \Delta\theta^2\right)_s}{\left(h_2\, \Delta\theta^2\right)_{ss}} \qquad (3.44)$$

$$A^T_{FF} = \frac{-CFM * \left(h_3\, \Delta\theta^3\right)_f}{\left(h_3\, \Delta\theta^3\right)_{ff}}$$

$$A^T_{BB} = \frac{-CBP * \left(h_3\, \Delta\theta^3\right)_b}{\left(h_3\, \Delta\theta^3\right)_{bb}}$$

$$A_{EER} = A^T_{EE} * T_{i+2} * C_{pm_{i+2}}$$

$$A_{WWR} = A^T_{WW} * T_{i-2} * C_{pm_{i-2}}$$

$$A_{NNR} = A^T_{NN} * T_{j+2} * C_{pm_{j+2}}$$

$$A_{SSR} = A^T_{SS} * T_{j-2} * C_{pm_{j-2}} \qquad (3.45)$$

$$A_{FFR} = A^T_{FF} * T_{k+2} * C_{pm_{k+2}}$$

$$A_{BBR} = A^T_{BB} * T_{k-2} * C_{pm_{k-2}}$$

The intermediate coefficients are :

$$A_{EI} = -0.5 * CE + CEP + CEM * \left[1 + \frac{\left(h_1 \, \Delta\theta^1\right)_e}{\left(h_1 \, \Delta\theta^1\right)_{ee}} \right] + \qquad (3.46)$$

$$+ CWM * \frac{\left(h_1 \, \Delta\theta^1\right)_w}{\left(h_1 \, \Delta\theta^1\right)_e}$$

$$A_{WI} = 0.5 * CW + CWM + CWP * \left[1 + \frac{\left(h_1 \, \Delta\theta^1\right)_w}{\left(h_1 \, \Delta\theta^1\right)_{ww}} \right] + \qquad (3.47)$$

$$+ CEP * \frac{\left(h_1 \, \Delta\theta^1\right)_e}{\left(h_1 \, \Delta\theta^1\right)_w}$$

$$A_{NI} = -0.5 * CN + CNP + CNM * \left[1 + \frac{\left(h_2 \, \Delta\theta^2\right)_n}{\left(h_2 \, \Delta\theta^2\right)_{nn}} \right] + \qquad (3.48)$$

$$+ CSM * \frac{\left(h_2 \, \Delta\theta^2\right)_s}{\left(h_2 \, \Delta\theta^2\right)_n}$$

$$A_{SI} = 0.5 * CS + CSM + CSP * \left[1 + \frac{\left(h_2 \, \Delta\theta^2\right)_s}{\left(h_2 \, \Delta\theta^2\right)_{ss}} \right] + \qquad (3.49)$$

$$+ CNP * \frac{\left(h_2 \, \Delta\theta^2\right)_n}{\left(h_2 \, \Delta\theta^2\right)_s}$$

$$A_{FI} = -0.5 * CF + CFP + CFM * \left[1 + \frac{\left(h_3\, \Delta\theta^3\right)_f}{\left(h_3\, \Delta\theta^3\right)_{ff}} \right] + $$

$$+ CBM * \frac{\left(h_3\, \Delta\theta^3\right)_b}{\left(h_3\, \Delta\theta^3\right)_f} \qquad (3.50)$$

$$A_{BI} = 0.5 * CB + CBM + CBP * \left[1 + \frac{\left(h_3\, \Delta\theta^3\right)_b}{\left(h_3\, \Delta\theta^3\right)_{bb}} \right] + $$

$$+ CFP * \frac{\left(h_3\, \Delta\theta^3\right)_f}{\left(h_3\, \Delta\theta^3\right)_b} \qquad (3.51)$$

The coefficients are:

$$A_E^T = A_{EI} * C_{pm.E} + CONDE1$$

$$A_W^T = A_{WI} * C_{pm.W} + CONDW1$$

$$A_N^T = A_{NI} * C_{pm.N} + CONDN1 \qquad (3.52)$$

$$A_S^T = A_{SI} * C_{pm.S} + CONDS1$$

$$A_F^T = A_{FI} * C_{pm.F} + CONDF1$$

$$A_B^T = A_{BI} * C_{pm.B} + CONDB1$$

A_P^T is the sum of all the values of A.

$$A_P^T = C_{pm.p} * \left(A_E^T + A_W^T + A_N^T + A_S^T + A_F^T + A_B^T + A_{EE}^T + A_{WW}^T + \right.$$

$$\left. + A_{NN}^T + A_{SS}^T + A_{FF}^T + A_{BB}^T\right) + CONDE1 + CONDW1 + \quad (3.53)$$

$$+ CONDN2 + CONDS1 + CONDF1 + CONDB1$$

F. MOMENTUM EQUATION

The integrated momentum equation is given as:

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

with A_i, the area of the staggered cell given by Eqns. 3.14 through 3.16. M^{ij} represents the total momentum flux in the θ^{ij} direction due to convection and diffusion for the u^i velocity component. M is evaluated at the face noted and is given by:

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

The source term includes body force, pressure gradient, centrifugal, and Coriolis forces and for u^1 is :

$$S^1 = -P_e A_e + P_w A_w + \rho G^1 \Delta V - M_p^{12} (A_n - A_s) -$$
$$- M_p^{13} (A_f - A_b) + (M_p^{22} + M_p^{33}) (A_e - A_w) \quad (3.56)$$

Yang et al. [Ref. 20: pp. 11-13] describes the concept of a "stress-flex formulation" as it applies to a curvilinear coordinate system.

Stresses are calculated from previous information and the source is given in the current iteration. The momentum flux is:

$$M^{ij} = \hat{M}^{ij} + \left(\hat{\sigma}_i^j - \sigma_i^j \right) \tag{3.57}$$

with

$$\hat{\sigma}_i^j = \frac{\mu}{\left[h_j \left(\frac{\partial u^i}{\partial \theta^j} \right) \right]} \tag{3.58}$$

$$\hat{M}^{ij} = \rho\, u^i u^j - \hat{\sigma}_i^j \tag{3.59}$$

The u^1 momentum equation becomes:

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

$$\hat{S} = S - \left(\hat{\sigma}_1^1 - \sigma_1^1 \right)_e A_e + \left(\hat{\sigma}_1^1 - \sigma_1^1 \right)_w A_w - \\ - \left(\hat{\sigma}_1^2 - \sigma_1^2 \right)_n A_n + \left(\hat{\sigma}_1^2 - \sigma_1^2 \right)_s A_s - \\ - \left(\hat{\sigma}_1^3 - \sigma_1^3 \right)_f A_f + \left(\hat{\sigma}_1^3 - \sigma_1^3 \right)_b A_b \tag{3.61}$$

The momentum equation for θ^i is given as:

$$\left(A_p^{u^1} + \rho^{n-1} \frac{\Delta V}{\Delta t}\right) u_p^1 = A_e^{u^1} u_e^1 + A_w^{u^1} u_w^1 +$$

$$+ A_n^{u^1} u_n^1 + A_s^{u^1} u_s^1 + A_f^{u^1} u_f^1 + A_b^{u^1} u_b^1 + S^{u^1} u^1 \qquad (3.62)$$

The intermediate mass flow rates per unit area are:

$$G_{ne} = u_{j+1}^2 \left\{ \frac{\left[\rho_{j+1}\left(h_2\,\Delta\theta^2\right)_j + \rho_j\left(h_2\,\Delta\theta^2\right)_{j+1}\right]}{\left(h_2\,\Delta\theta^2\right)_j + \left(h_2\,\Delta\theta^2\right)_{j+1}} \right\}$$

$$G_{nw} = u_{i-1,\,j+1}^2 \left\{ \frac{\left[\rho_{i-1,\,j+1}\left(h_2\,\Delta\theta^2\right)_j + \rho_{i-1}\left(h_2\,\Delta\theta^2\right)_{j+1}\right]}{\left(h_2\,\Delta\theta^2\right)_j + \left(h_2\,\Delta\theta^2\right)_{j+1}} \right\}$$

$$G_{se} = u^2 \left\{ \frac{\left[\rho_{j-1}\left(h_2\,\Delta\theta^2\right)_j + \rho_j\left(h_2\,\Delta\theta^2\right)_{j-1}\right]}{\left(h_2\,\Delta\theta^2\right)_j + \left(h_2\,\Delta\theta^2\right)_{j+1}} \right\} \qquad (3.63)$$

$$G_{sw} = u_{i-1}^2 \left\{ \frac{\left[\rho_{i-1,\,j-1}\left(h_2\,\Delta\theta^2\right)_j + \rho_{i-1}\left(h_2\,\Delta\theta^2\right)_{j-1}\right]}{\left(h_2\,\Delta\theta^2\right)_j + \left(h_2\,\Delta\theta^2\right)_{j-1}} \right\}$$

$$G_e = u_{i+1}^1 \left\{ \frac{\left[\rho_{i+1}\left(h_1\,\Delta\theta^1\right)_e + \rho_i\left(h_1\,\Delta\theta^1\right)_{ee}\right]}{\left(h_1\,\Delta\theta^1\right)_e + \left(h_1\,\Delta\theta^1\right)_{ee}} \right\}$$

$$G_p = u^1 \left\{ \frac{\left[\rho_{i-1}\left(h_1\,\Delta\theta^1\right)_e + \rho_i\left(h_1\,\Delta\theta^1\right)_w\right]}{\left(h_1\,\Delta\theta^1\right)_e + \left(h_1\,\Delta\theta^1\right)_w} \right\}$$

$$G_w = u_{i-1}^1 \left\{ \frac{\left[\rho_{i-2}\left(h_1\,\Delta\theta^1\right)_w + \rho_{i-1}\left(h_1\,\Delta\theta^1\right)_{ww}\right]}{\left(h_1\,\Delta\theta^1\right)_w + \left(h_1\,\Delta\theta^1\right)_{ww}} \right\}$$

$$G_{fe} = u^3_{k+1} \left\{ \frac{\left[\rho_{k+1} \left(h_3\,\Delta\theta^3\right)_k + \rho_k \left(h_3\,\Delta\theta^3\right)_{k+1}\right]}{\left(h_3\,\Delta\theta^3\right)_k + \left(h_3\,\Delta\theta^3\right)_{k+1}} \right\}$$

$$G_{fw} = u^3_{i-1,\,k+1} \left\{ \frac{\left[\rho_{i-1,\,k+1} \left(h_3\,\Delta\theta^3\right)_k + \rho_{i-1} \left(h_3\,\Delta\theta^3\right)_{k+1}\right]}{\left(h_3\,\Delta\theta^3\right)_k + \left(h_3\,\Delta\theta^3\right)_{k+1}} \right\}$$

$$G_{be} = u^3 \left\{ \frac{\left[\rho_{k-1} \left(h_3\,\Delta\theta^3\right)_k + \rho_k \left(h_3\,\Delta\theta^3\right)_{k-1}\right]}{\left(h_3\,\Delta\theta^3\right)_k + \left(h_3\,\Delta\theta^3\right)_{k-1}} \right\}$$

$$G_{bw} = u^3_{i-1} \left\{ \frac{\left[\rho_{i-1,\,k-1} \left(h_3\,\Delta\theta^3\right)_k + \rho_{i-1} \left(h_3\,\Delta\theta^3\right)_{k-1}\right]}{\left(h_3\,\Delta\theta^3\right)_k + \left(h_3\,\Delta\theta^3\right)_{k-1}} \right\}$$

The final mass flow rates for the control volume surfaces are:

$$CE = 0.5\,(G_e + G_p) * \left(h_2\,\Delta\theta^2\right)_e * \left(h_3\,\Delta\theta^3\right)_e$$

$$CW = 0.5\,(G_p + G_w) * \left(h_2\,\Delta\theta^2\right)_w * \left(h_3\,\Delta\theta^3\right)_w \qquad (3.64)$$

$$CN = \left(h_1\,\Delta\theta^1\right)_n \left(h_3\,\Delta\theta^3\right)_n \left\{ \frac{\left[G_{ne} \left(h_1\,\Delta\theta^1\right)_w + G_{nw} \left(h_1\,\Delta\theta^1\right)_e\right]}{\left[\left(h_1\,\Delta\theta^1\right)_w + \left(h_1\,\Delta\theta^1\right)_e\right]} \right\}$$

$$CS = \left(h_1\,\Delta\theta^1\right)_s \left(h_3\,\Delta\theta^3\right)_s \left\{ \frac{\left[G_{se} \left(h_1\,\Delta\theta^1\right)_w + G_{sw} \left(h_1\,\Delta\theta^1\right)_e\right]}{\left[\left(h_1\,\Delta\theta^1\right)_w + \left(h_1\,\Delta\theta^1\right)_e\right]} \right\}$$

$$CF = \left(h_1\,\Delta\theta^1\right)_f \left(h_2\,\Delta\theta^2\right)_f \left\{ \frac{\left[G_{fe} \left(h_1\,\Delta\theta^1\right)_w + G_{fw} \left(h_1\,\Delta\theta^1\right)_e\right]}{\left[\left(h_1\,\Delta\theta^1\right)_w + \left(h_1\,\Delta\theta^1\right)_e\right]} \right\}$$

$$CB = \left(h_1 \Delta\theta^1\right)_b \left(h_2 \Delta\theta^2\right)_b \left\{ \frac{\left[G_{be}\left(h_1 \Delta\theta^1\right)_w + G_{bw}\left(h_1 \Delta\theta^1\right)_e\right]}{\left[\left(h_1 \Delta\theta^1\right)_w + \left(h_1 \Delta\theta^1\right)_e\right]} \right\}$$

The local viscosity becomes:

$$VIS_e = VIS$$

$$VIS_w = VIS_{i-1}$$

$$VIS_n = \frac{(VIS_{j+1} + VIS + VIS_{i-1,\,j+1} + VIS_{i-1})}{4.0} \qquad (3.65)$$

$$VIS_s = \frac{(VIS_{j-1} + VIS + VIS_{i-1,\,j-1} + VIS_{i-1})}{4.0}$$

$$VIS_f = \frac{(VIS_{k+1} + VIS + VIS_{i-1,\,k+1} + VIS_{i-1})}{4.0}$$

$$VIS_b = \frac{(VIS_{k-1} + VIS + VIS_{i-1,\,k-1} + VIS_{i-1})}{4.0}$$

$$VISN1 = VIS_n * \left[\frac{\left(h_3 \Delta\theta^3\right)\left(h_1 \Delta\theta^1\right)}{h_2 \Delta\theta^2}\right]_n$$

$$VISS1 = VIS_s * \left[\frac{\left(h_3 \Delta\theta^3\right)\left(h_1 \Delta\theta^1\right)}{h_2 \Delta\theta^2}\right]_s$$

$$VISE1 = VIS_e * \left[\frac{\left(h_2 \Delta\theta^2\right)\left(h_3 \Delta\theta^3\right)}{h_1 \Delta\theta^1}\right]_e \qquad (3.66)$$

$$VISW1 = VIS_w * \left[\frac{\left(h_2\, \Delta\theta^2\right)\left(h_3\, \Delta\theta^3\right)}{h_1\, \Delta\theta^1} \right]_w$$

$$VISF1 = VIS_f * \left[\frac{\left(h_1\, \Delta\theta^1\right)\left(h_2\, \Delta\theta^2\right)}{h_3\, \Delta\theta^3} \right]_f$$

$$VISB1 = VIS_b * \left[\frac{\left(h_1\, \Delta\theta^1\right)\left(h_2\, \Delta\theta^2\right)}{h_3\, \Delta\theta^3} \right]_b$$

The coefficients for the momentum equations are:

$$A_{EER} = A^u_{EE} * u^1_{i+2}$$

$$A_{WWR} = A^u_{WW} * u^1_{i-2}$$

$$A_{NNR} = A^u_{NN} * u^1_{j+2} \qquad (3.67)$$

$$A_{SSR} = A^u_{SS} * u^1_{j-2}$$

$$A_{FFR} = A^u_{FF} * u^1_{k+2}$$

$$A_{BBR} = A^u_{BB} * u^1_{k-2}$$

The values of the coefficients A are given as:

$$A^u_E = A_{E1} + VISE1$$

$$A^u_W = A_{W1} + VISW1$$

$$A_N^u = A_{NI} + VISN1 \quad (3.68)$$

$$A_S^u = A_{SI} + VISS1$$

$$A_F^u = A_{FI} + VISF1$$

$$A_B^u = A_{BI} + VISB1$$

The value of $A^u{}_p$ is the summation of all of the values of A:

$$A_p^u = A_E^u + A_W^u + A_N^u + A_S^u + A_F^u + A_B^u + A_{EE}^u + A_{WW}^u +$$

$$+ A_{NN}^u + A_{SS}^u + A_{FF}^u + A_{BB}^u \quad (3.69)$$

The source term is given as:

$$S_u^u = \frac{\left[\rho\left(h_1\,\Delta\theta^1\right)_w + \rho_{i-1}\left(h_1\,\Delta\theta^1\right)_e\right]}{\left[\left(h_1\,\Delta\theta^1\right)_e + \left(h_1\,\Delta\theta^1\right)_w\right]} * \frac{\Delta V}{\Delta t} * u^1 +$$

$$+\left(h_2\,\Delta\theta^2\right)_j\left(h_3\,\Delta\theta^3\right)_k\left(P_{i-1} - P_i\right) + A_{EER} + A_{WWR} + A_{NNR} +$$

$$+ A_{SSR} + A_{FFR} + A_{BBR} + RE - RW + RN - RS = RF - RB +$$

$$+ RRY + RRZ - RRX - Buoy * \{\sin\left[ZC(K)\right] * (\rho - \rho_{EQ}) *$$

$$*\left(h_1\,\Delta\theta^1\right)_w * \cos\left[XC(I)\right]\Big\} + \Big\{\left(\rho_{i-1} - \rho_{EQ_{i-1}}\right)\left(h_1\,\Delta\theta^1\right)_e * \quad (3.70)$$

$$* \cos\left[XC(I-1)\right]\Big\} / \left[\left(h_1\,\Delta\theta^1\right)_w + \left(h_1\,\Delta\theta^1\right)_e\right]\Delta V$$

with XZ and ZC as the center of the basic cell. The additional parameters are given below.

$$RE = \left(h_2\,\Delta\theta^2\;h_3\,\Delta\theta^3\right)_e \left[\frac{\sigma^{11} - (u^1_{i+1} - u^1_i) * VIS_e}{\left(h_1\,\Delta\theta^1\right)_e}\right]$$

$$RW = \left(h_2\,\Delta\theta^2\;h_3\,\Delta\theta^3\right)_w \left[\frac{\sigma^{11}_{i-1} - (u^1 - u^1_{i-1}) * VIS_w}{\left(h_1\,\Delta\theta^1\right)_w}\right]$$

$$RN = \left(h_1\,\Delta\theta^1\;h_3\,\Delta\theta^3\right)_n \left[\frac{\sigma^{12}_{j+1} - (u^1_{j+1} - u^1_j) * VIS_n}{\left(h_2\,\Delta\theta^2\right)_n}\right] \tag{3.71}$$

$$RS = \left(h_1\,\Delta\theta^1\;h_3\,\Delta\theta^3\right)_s \left[\frac{\sigma^{12} - (u^1 - u^1_{j-1}) * VIS_s}{\left(h_3\,\Delta\theta^3\right)_s}\right]$$

$$RF = \left(h_1\,\Delta\theta^1\;h_2\,\Delta\theta^2\right)_f \left[\frac{\sigma^{13}_{k+1} - (u^1_{k+1} - u^1_k) * VIS_f}{\left(h_3\,\Delta\theta^3\right)_f}\right]$$

$$RB = \left(h_1\,\Delta\theta^1\;h_2\,\Delta\theta^2\right)_b \left[\frac{\sigma^{13} - (u^1 - u^1_{k-1}) * VIS_b}{\left(h_3\,\Delta\theta^3\right)_b}\right]$$

$$\bar{\sigma}^{12} = 0.5\left(\sigma^{12}_{j+1} + \sigma^{12}_j\right)$$

$$\bar{\sigma}^{13} = 0.5\left(\sigma^{13}_{k+1} + \sigma^{13}_k\right)$$

$$\bar{\sigma}^{22} = \frac{\sigma^{22}\left(h_1\,\Delta\theta^1\right)_w + \sigma^{22}_{i-1}\left(h_1\,\Delta\theta^1\right)_e}{\left(h_1\,\Delta\theta^1\right)_w + \left(h_1\,\Delta\theta^1\right)_e} \tag{3.72}$$

$$\bar{\sigma}^{33} = \frac{\sigma^{13}\left(h_1\,\Delta\theta^1\right)_w + \sigma^{33}_{i-1}\left(h_1\,\Delta\theta^1\right)_e}{\left(h_1\,\Delta\theta^1\right)_w + \left(h_1\,\Delta\theta^1\right)_e}$$

$$AU1 = u^1$$

$$AU2 = \left\{\left[\frac{u^2_{j+1}\left(h_2\,\Delta\theta^2\right)_j + u^2_j\left(h_2\,\Delta\theta^2\right)_j}{2\,\left(h_2\,\Delta\theta^2\right)_j}\right]\left(h_1\,\Delta\theta^1\right)_w\right.$$

$$\left.+\left[\frac{u^2_{i-1,\,j+1}\left(h_2\,\Delta\theta^2\right)_j + u^2_{i-1}\left(h_2\,\Delta\theta^2\right)_j}{2\,\left(h_2\,\Delta\theta^2\right)_j}\right]\left(h_1\,\Delta\theta^1\right)_e\right\} / \qquad (3.73)$$

$$/\left[\left(h_1\,\Delta\theta^1\right)_e + \left(h_1\,\Delta\theta^1\right)_w\right]$$

$$AU3 = \left\{\left[\frac{u^3_{k+1}\left(h_3\,\Delta\theta^3\right)_k + u^3_k\left(h_3\,\Delta\theta^3\right)_k}{2\,\left(h_3\,\Delta\theta^3\right)_k}\right]\left(h_1\,\Delta\theta^1\right)_w\right.$$

$$\left.+\left[\frac{u^3_{i-1,\,k+1}\left(h_3\,\Delta\theta^3\right)_k + u^3_{i-1}\left(h_3\,\Delta\theta^3\right)_k}{2\,\left(h_3\,\Delta\theta^3\right)_k}\right]\left(h_1\,\Delta\theta^1\right)_e\right\} /$$

$$/\left[\left(h_1\,\Delta\theta^1\right)_e + \left(h_1\,\Delta\theta^1\right)_w\right]$$

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

$$ARU12 = AR \ast AU1 \ast AU2$$

$$ARU13 = AR * AU1 * AU3 \tag{3.74}$$

$$ARU22 = AR * AU2 * AU2$$

$$ARU33 = AR * AU3 * AU3$$

$$RRY = (\bar{\sigma}^{12} - ARU12)\left(h_3 \Delta\theta^3\right)_k \left[\left(h_1 \Delta\theta^1\right)_n - \left(h_1 \Delta\theta^1\right)_s\right]$$

$$RRZ = (\bar{\sigma}^{13} - ARU13)\left(h_2 \Delta\theta^2\right)_j \left[\left(h_1 \Delta\theta^1\right)_f - \left(h_1 \Delta\theta^1\right)_b\right]$$

$$RRX = (\bar{\sigma}^{22} - ARU22)\left(h_3 \Delta\theta^3\right)_k \left[\left(h_2 \Delta\theta^2\right)_e - \left(h_2 \Delta\theta^2\right)_w\right] + $$

$$+ (\bar{\sigma}^{33} - ARU33)\left(h_2 \Delta\theta^2\right)_j \left[\left(h_3 \Delta\theta^3\right)_e - \left(h_3 \Delta\theta^3\right)_w\right] \tag{3.75}$$

The momentum equations in the other two directions can be similarly obtained.

G. PRESSURE CORRECTION

One difficulty encountered in employing primitive variables is the difficulty in calculating pressure. In a closed system, such as FIRE-1, there are two causes of changes in pressure. First, there are pressure changes throughout the field due to a net energy change in the system. To account for these changes, a global pressure correction is applied. Second, there are pressure changes locally which determine the velocity field. A local pressure correction is included to account for these changes.

1. Global Pressure Correction

A global pressure correction follows from the two-dimensional scheme developed by Nicolette, et al. [Ref. 4]. Overall pressure levels are increased or decreased depending upon whether energy is added or removed from the system. Since the volume and mass of the system are constant, the sum of the local density times the local volume will be constant, and equal to the equilibrium mass. Summing over all of the cells,

$$\sum \rho_i^n (\Delta V)_i = \sum \rho_{EQ,i} (\Delta V)_i \tag{3.76}$$

with n indicating any time and EQ indicating equilibrium.

Assuming a perfect gas, density is a function of pressure and temperature only, since volume is constant. The actual values of pressure and temperature at any time are the sum of an estimated value and the global correction.

$$P = P^* + P'_g \tag{3.77}$$

$$T = T^* + T'_g \tag{3.78}$$

with superscript * indicating the estimated value and superscript ′ indicating the global correction. By applying these two equations and the perfect gas law along with Eqn. 3.76, the global pressure correction becomes:

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

This correction is added to the estimated value from the previous time step, and iterated until a globally corrected pressure is obtained which conserves mass in every cell.

2. Local Pressure Correction

An iterative method involving the mass conservation equation is used to find the local pressure. Patankar [Ref. 35:pp. 120–126] and Doria [Ref. 34:pp. 26–32] describe the method for determining the local pressure correction. Initially, the pressure field is guessed or the previous pressure field is assumed. Then velocities are calculated based upon this assumed pressure distribution. Knowing the velocities, the mass source term, S_{mp} (also called residual mass), is calculated for each cell. The magnitude of the mass source term and the sum of the absolute values of every cell's residual mass serves as a check on the conservation of mass within each cell and through the entire flow field. If S_{mp} is close to zero, the guessed pressure field is satisfactory; if not, a local pressure correction is calculated and the process is repeated until S_{mp} is within the desired range. Once a satisfactory pressure field is found, the densities for the next time step can be found using the equation of state.

Similar to the global pressure correction, the actual pressure equals a guessed pressure (superscript *) plus the local pressure correction (superscript ′).

$$P = P^{*} + P'$$ (3.80)

The finite difference equation for the pressure correction takes on a form similar to the other finite difference conservation equations. The equation for P' is:

$$A_p P'_p = A_E P'_E + A_W P'_W + A_N P'_N + A_S P'_S + A_F P'_F + \\ + A_B P'_B - S_{mp} \Delta V$$ (3.81)

with

$$A_E = \frac{\rho_e * \left(h_2 \Delta\theta^2 h_3 \Delta\theta^3\right)_e^2}{\left(A_{p_{i+1}}^{u^1} + \rho_e \frac{\Delta V}{\Delta t}\right)}$$ (3.82)

$$A_W = \frac{\rho_w * \left(h_2 \Delta\theta^2 h_3 \Delta\theta^3\right)_w^2}{\left(A_{p}^{u^1} + \rho_w \frac{\Delta V}{\Delta t}\right)}$$ (3.83)

$$A_N = \frac{\rho_n * \left(h_1 \Delta\theta^1 h_3 \Delta\theta^3\right)_n^2}{\left(A_{p_{j+1}}^{u^2} + \rho_n \frac{\Delta V}{\Delta t}\right)}$$ (3.84)

$$A_S = \frac{\rho_s * \left(h_1 \Delta\theta^1 h_3 \Delta\theta^3\right)_s^2}{\left(A_{p}^{u^2} + \rho_s \frac{\Delta V}{\Delta t}\right)}$$ (3.85)

$$A_F = \frac{\rho_f * \left(h_1 \Delta\theta^1 h_2 \Delta\theta^2\right)_f^2}{\left(A_P^{u^3} + \rho_f \frac{\Delta V}{\Delta t}\right)} \quad (3.86)$$

$$A_B = \frac{\rho_b * \left(h_1 \Delta\theta^1 h_2 \Delta\theta^2\right)_b^2}{\left(A_P^{u^3} + \rho_b \frac{\Delta V}{\Delta t}\right)} \quad (3.87)$$

$$A_P = A_E + A_W + A_N + A_S + A_F + A_B \quad (3.88)$$

At the solid boundaries where the mass flux is zero, the coefficient A which corresponds to the boundary is equal to zero. When the final corrected pressure field has been calculated, new velocities are found from the following equations.

$$u^1 = u^{1*} + u^{1\prime} \quad (3.89)$$

$$u^2 = u^{2*} + u^{2\prime} \quad (3.90)$$

$$u^3 = u^{3*} + u^{3\prime} \quad (3.91)$$

with

$$u^{1\prime} = \frac{(P_P - P_W)\,(h_2 \Delta\theta^2 h_3 \Delta\theta^3)}{A_P^{u^1} + \rho_w \frac{\Delta V}{\Delta t}} \quad (3.92)$$

$$u^{2\prime} = \frac{(P_p - P_s)(h_1 \Delta\theta^1 h_3 \Delta\theta^3)}{A_p^{u^2} + \rho_s \frac{\Delta V}{\Delta t}} \tag{3.93}$$

$$u^{3\prime} = \frac{(P_p - P_b)(h_1 \Delta\theta^1 h_2 \Delta\theta^2)}{A_p^{u^3} + \rho_b \frac{\Delta V}{\Delta t}} \tag{3.94}$$

S_{mp} is then computed; if it is within the desired range, the calculation is complete. Otherwise a new P' is calculated and the procedure is repeated.

H. VENTILATION EQUATIONS

When forced ventilation is introduced, the velocity equation for the control volume containing the ventilation becomes:

$$A_p u_p = A_e u_e + A_w u_w + A_n u_n + A_s u_s + + A_f u_f + A_b u_b + S_u \tag{3.95}$$

with

$$A_p = 10^{20} \tag{3.96}$$

$$S_u = \text{specified velocity} * 10^{20} \tag{3.97}$$

this causes the velocity in the control volume to be equal to the desired values for ventilation, and not be affected by the upwind or other adjacent velocities.

The boundaries of the control volumes with specified velocity require special consideration. The equation for the downwind control volume becomes:

$$A_p u_p = A_e u_e + A_w u_w + A_n u_n + A_s u_s + + A_f u_f + A_b^{\bullet} u_b + S_u^{\bullet} \tag{3.98}$$

with the starred values defined as:

$$A_b^{\bullet} = 0.0 \tag{3.99}$$

$$S_u^{\bullet} = S_u + A_b u_b \tag{3.100}$$

This causes the ventilation to be the only effect from the upwind cell and represents a fixed velocity internal ventilation system. The equations for the adjacent control volumes whose boundaries are parallel to the flow must also change. For example, the equation for the control volume north of the specified ventilation control volume becomes

$$A_p u_p = A_e u_e + A_w u_w + A_n u_n + A_s^{\bullet} u_s + + A_f u_f + A_b u_b + S_u^{\bullet} \tag{3.101}$$

with

$$S_u^{\bullet} = S_u + 2\, u_s A_s \tag{3.102}$$

$$A_s^{\bullet} = 0.0 \tag{3.103}$$

This boundary equation makes the velocity in the entire volume constant, rather than varying between the staggered cell center and the boundary.

IV. EVALUATION OF NUMERICAL DATA

A. INTRODUCTION

The computer model presented here was designed to model fires in the experimental pressure vessel FIRE-1. The theory of the model has been given in previous chapters. This chapter will describe the modeling of a fire with internal ventilation in FIRE-1. Although such a fire test has yet to be experimentally run, this study will demonstrate the feature of internal ventilation in the computer model. This is one step to make the model more accurately represent real fires. The parameters used in the study will be presented in this chapter and the numerical solution process will be summarized. The effects of different time steps in the computation will also be discussed.

Two trials were conducted, one with internal ventilation and one without ventilation. A third trial was conducted using the ventilated case, but with different time steps for the iterations.

Pressure, temperature, and velocity fields are generated from the computer code. The temperature and velocity fields for various times will be discussed for both the ventilated and nonventilated cases. The global pressure and thermocouple temperatures will also be evaluated. The thermocouple temperatures correspond to the temperatures found at the location of the actual thermocouples in FIRE-1, in the north end cap (shown in Figure 4.1). Additionally, the global pressure

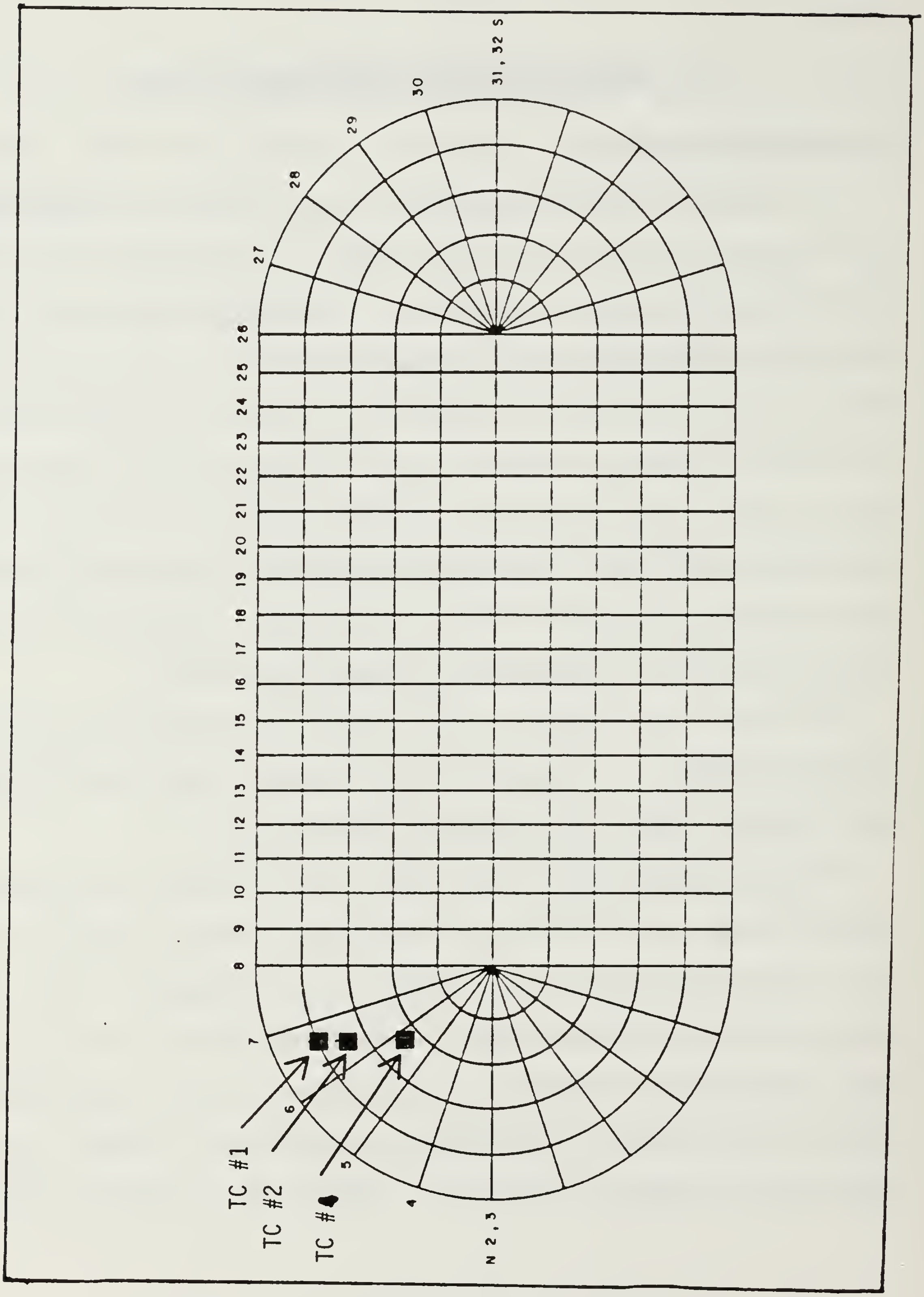

Figure 4-1. **Thermocouple Locations**

and one thermocouple temperature will be compared for the cases with different time steps.

B. NUMERICAL SOLUTION PARAMETERS

Various parameters are input into the numerical model in order to model a particular fire. These parameters include: initial conditions, fuel heat release rate, location of the fire, geometry of the enclosure, and physical characteristics of the enclosure, including heat transfer coefficient and fluid properties inside the enclosure. Other items could be added, depending upon the complexity of the model: decks, equipment, fire extinguishing systems, and combustion parameters. These are planned to be added to this model in the future. The location of sensors and the physical description of FIRE–1 is given in Chapter 1. The ventilation fan locations are shown in Figures 4.2 and 4.3. The material properties used in this thesis are listed in Table 4.1.

The numerical model of FIRE–1 uses a cylindrical/spherical coordinate system shown in Figures 4.2 and 4.3. The grid is spherical in the end caps, with θ, R, and ϕ directions, and cylindrical in the mid-section, with θ, R, and Z directions. There are 14 cells in the R direction; one cell represents the tank wall and another is in the vicinity of $R = 0$ and is used to avoid singularity at the origin. Each end cap has six ϕ cells; again, one cell is used to avoid singularity. The mid-section has 18 Z (or ϕ) cells and there are 20 cells in the θ direction oriented counterclockwise. Although a finer grid could be used to

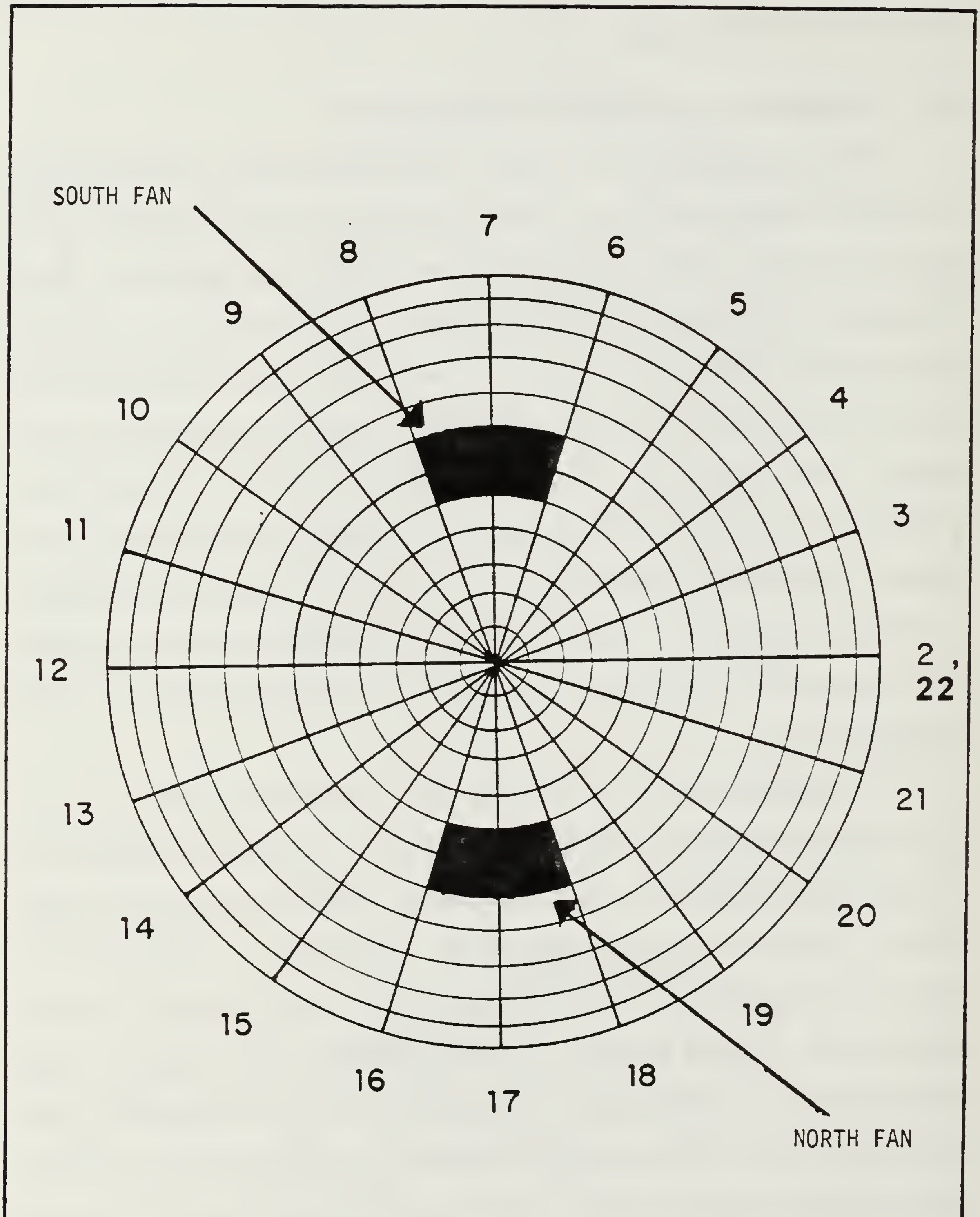

Figure 4-2. **Ventilation Location in Computer Model (End View)**

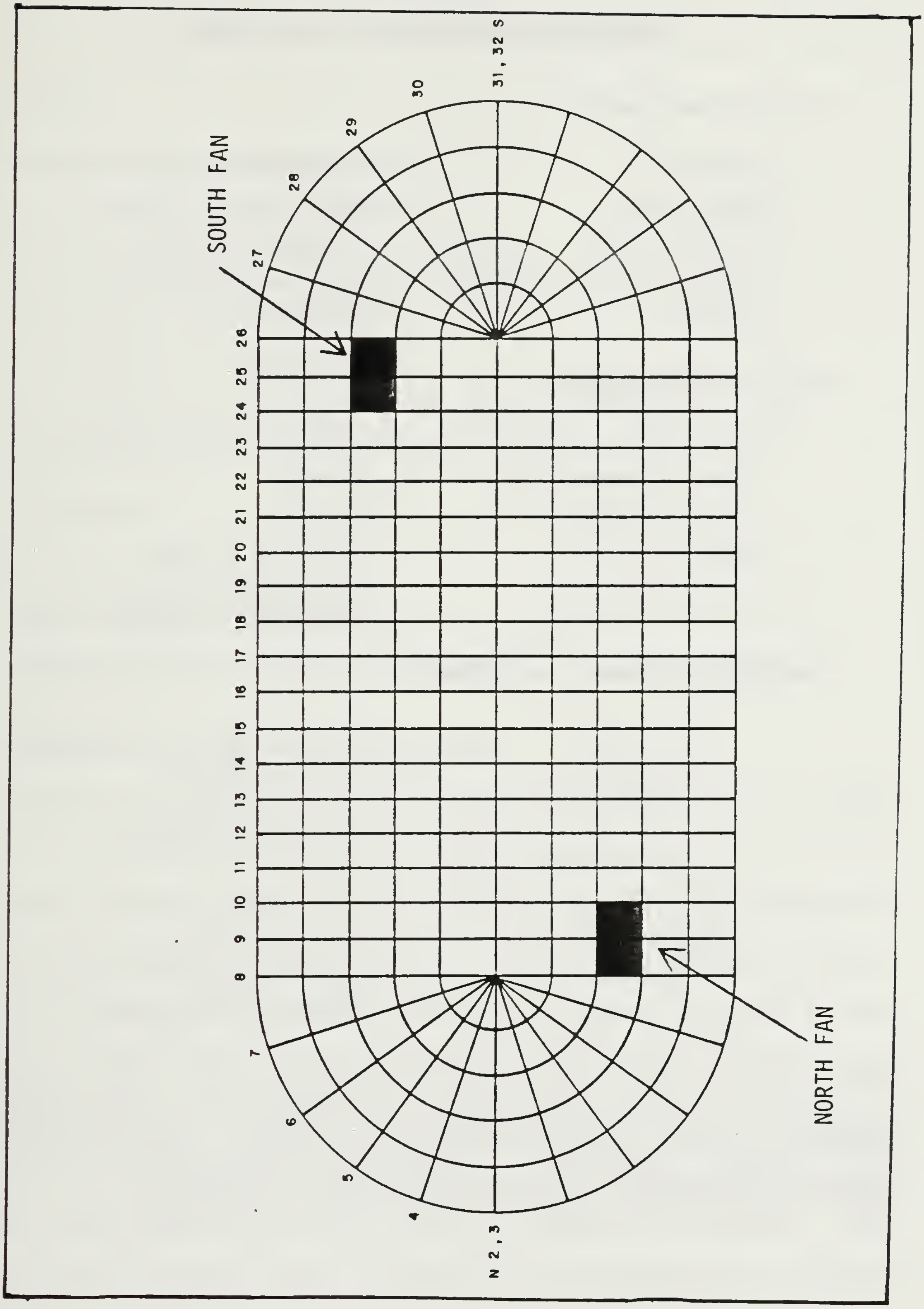

Figure 4-3. **Ventilation Location in Computer Model (Front View)**

TABLE 4.1

SPECIFIC 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
Fire Characteristics		
	Burn rate	A Given Function of Time
	Initial Temperature	35.6 C.
	Initial Pressure	1.0 Atm
	Location	Center of FIRE-1 23.1 ft from end 3.21 ft from bottom
Ventilation Characteristics		
	1. Velocity	3.18 ft/ sec
	Direction	South to North
	Location	11.1 ft from end 4.0 ft from bottom
	2. Velocity	3.18 ft/ sec
	Direction	North to South
	Location	35.5 ft from end 13.6 ft from bottom

give more accurate solutions, the limitations of the computer resources required that the grid not be enlarged. Table 4.2 presents additional information concerning the model parameters.

TABLE 4.2

GENERAL MODEL PARAMETERS

Grid	
Number of Interior Cells	6,720
Number of Tank Wall Cells	560
Number of Wall Radiation Zones	560
Number of Fire Radiation Zones	19
Cells in the θ Direction	20
Cells in the R Direction	14
Cells in the ϕ Direction (six in each end cap)	12
Cells in the Z direction (in the mid-section)	18
Time Step	
Varied	0.0192–0.0288 Sec
CPU Time (1 CPU hour)	0.6–0.8 sec fire time
External Heat Transfer Coefficient	15.0 Btu/ (hr ft^2 F)

C. NUMERICAL SOLUTION PROCESS

Two separate programs comprise this model; the first is a surface radiation preprocesser program which calculates the view factors. The main program is similar to that presented by Nies [Ref. 29:pp. 54–57] and Raycraft [Ref. 30:pp. 96–97]. The first part of the main program establishes the initial parameters and inputs the view factors. Then the effective viscosity is computed in Subroutine CALVIS. Every two time steps, the wall radiation flux is recalculated. Temperature, pressure and velocity are computed in subroutines using a semi-implicit technique which solves the finite difference equations. Subroutine CALT is then called to determine the temperatures, followed by the computation of the pressure and global pressure correction. Then the velocities and local pressure corrections are computed; the local pressure correction updates the velocities. With the corrected velocities, continuity is applied to each cell and the residual mass is found. The sum of the absolute value of every cell's residual mass is called the residual mass source, RESORM. The magnitude of RESORM indicates whether the pressure corrections are sufficient. If RESORM is too large, the program recalculates the velocities and pressures until RESORM comes within the desired range. If RESORM is greater than 10.0, the program stops because this only happens when there is a stability problem. If this occurs, the time step must be reduced and the program restarted using data from a previous step. To economize computer time, the temperature, global pressure, and density are only calculated every third iteration. The iterations continue until: (1) RESORM is below the predetermined value, (2) the maximum number of iterations has been reached, or (3) the CPU time presently available is insufficient to complete another iteration.

D. VENTILATION RESULTS

The numerical model was used to evaluate two fire scenarios: one included internal ventilation and the other did not. The specific parameters of the model were discussed previously. The validity of the ventilation model will be evaluated and the numerical results of the internal ventilation case will be compared to the nonventilated case.

A direct comparison can be made by looking at the spatial and temporal variations of the velocity and temperature fields. Although these fields are three-dimensional, they are presented in a two-dimensional form at three representative sections in the tank, shown in Figure 4.4. Section A is the mid-section front view, which cuts the vessel vertically along the axis (Y-Z plane). Section B is the mid-section end view from the south end, cutting the vessel through the middle of the vessel, perpendicular to the axis (X-Y plane). Section C is the section view at the base of the end cap from the south end, which is also cut perpendicular to the axis but at the intersection of the cylindrical and spherical portions of the tank (X-Y plane). The ventilated and nonventilated temperature and pressure fields for the times 30, 60, 90, 120 and 150 seconds are shown in Figures 4.5 through 4.35.

Many observations can be made in analyzing the field plots, but only the major phenomena will be discussed here. Raycraft, et al. [Ref. 38] discuss the results of the nonventilated computer model. In this thesis, discussion will be limited to comparisons of the two cases and some general comments. Particularly interesting phenomena include the flame plume, global velocity field, ventilation effects, temperature stratification, and the velocity field in a small region near the base of the flame plume during the beginning of the fire.

As can be seen in Figures 4.5 through 4.8, the flame plume is well formed early in the fire in both the nonventilated and ventilated cases

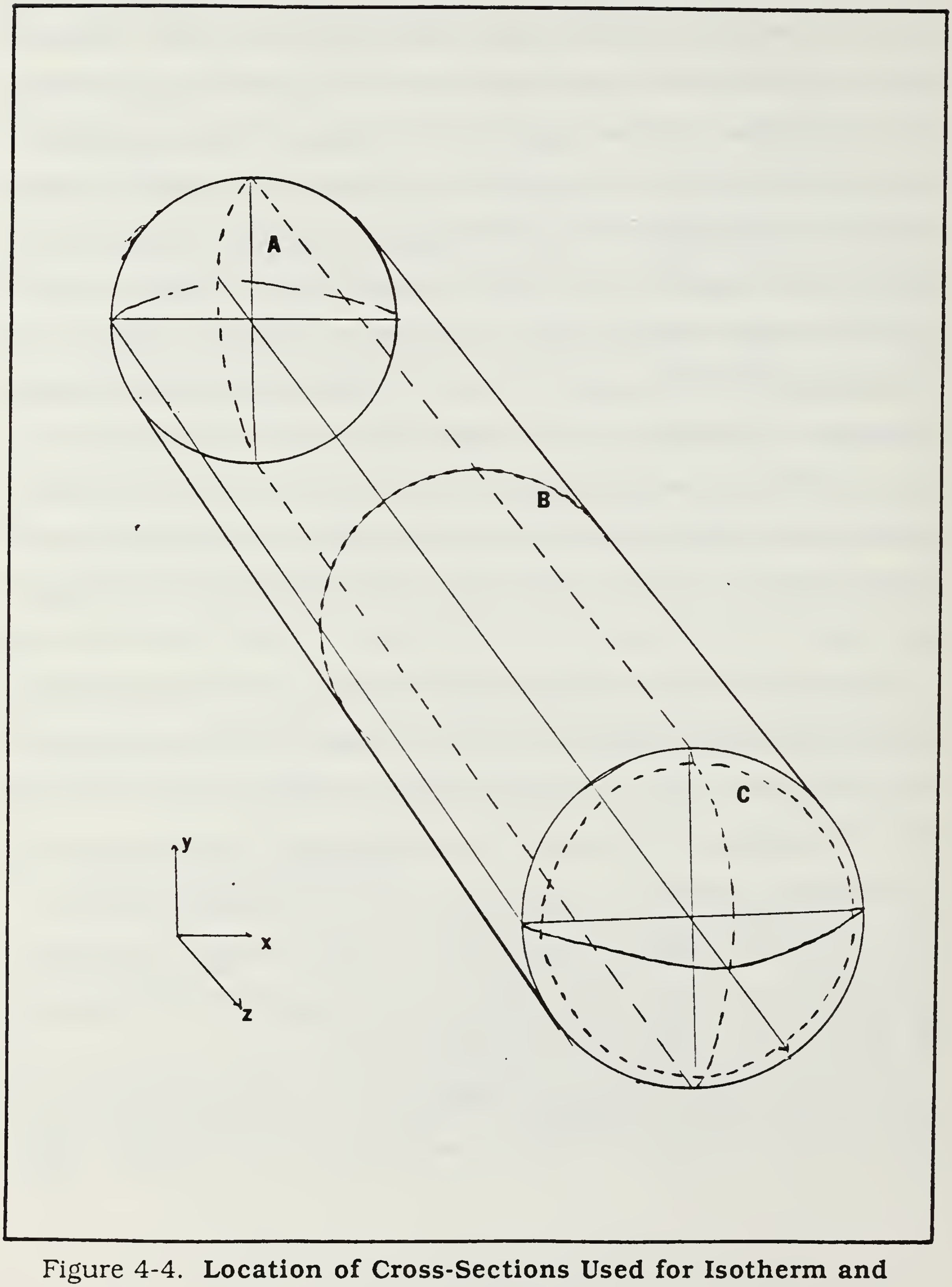

Figure 4-4. **Location of Cross-Sections Used for Isotherm and Velocity Field Plots**

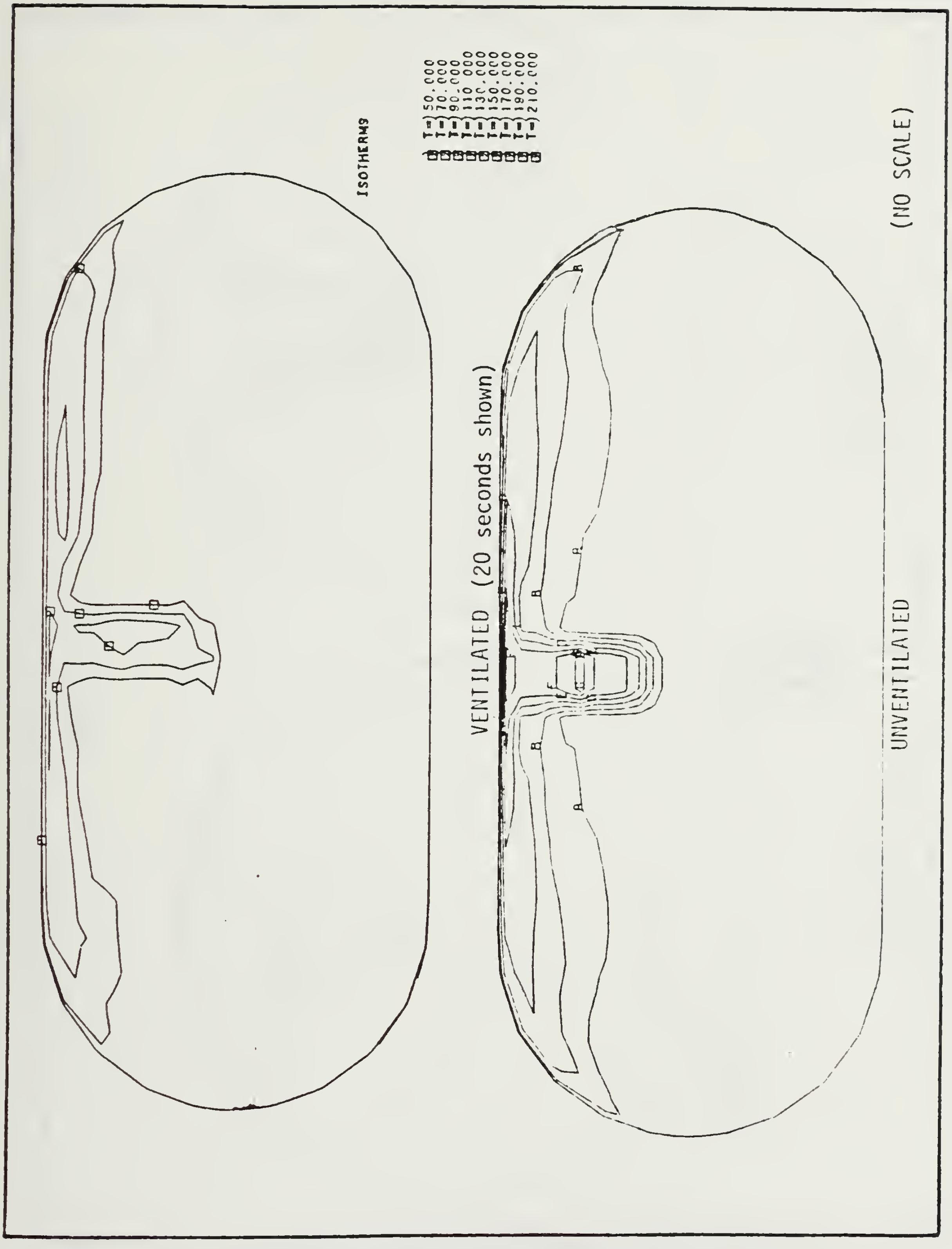

Figure 4-5. **Mid-Section Front Views of Isotherms at 30 Seconds**

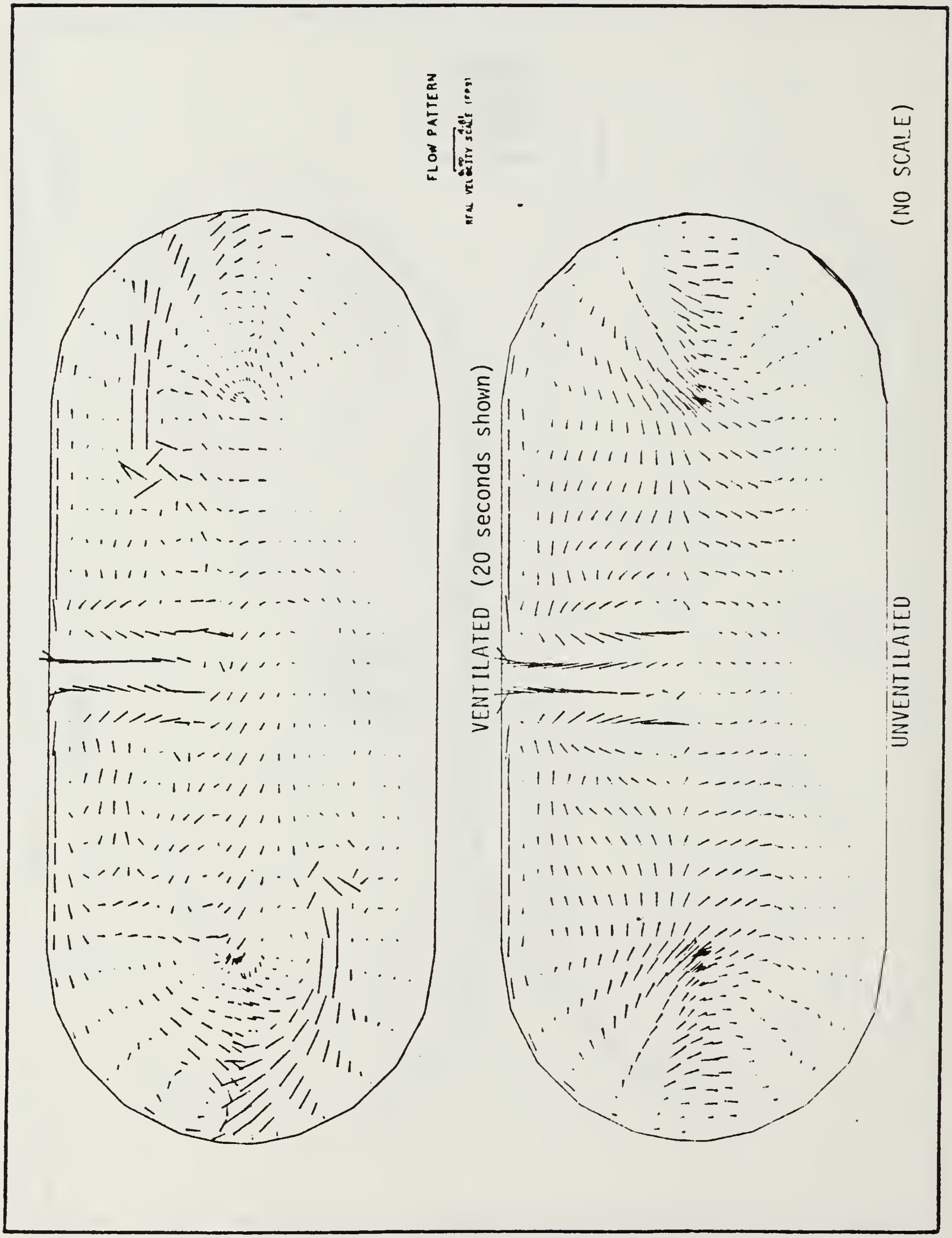

Figure 4-6. **Mid-Section Front Views of Velocity Field at 30 Seconds**

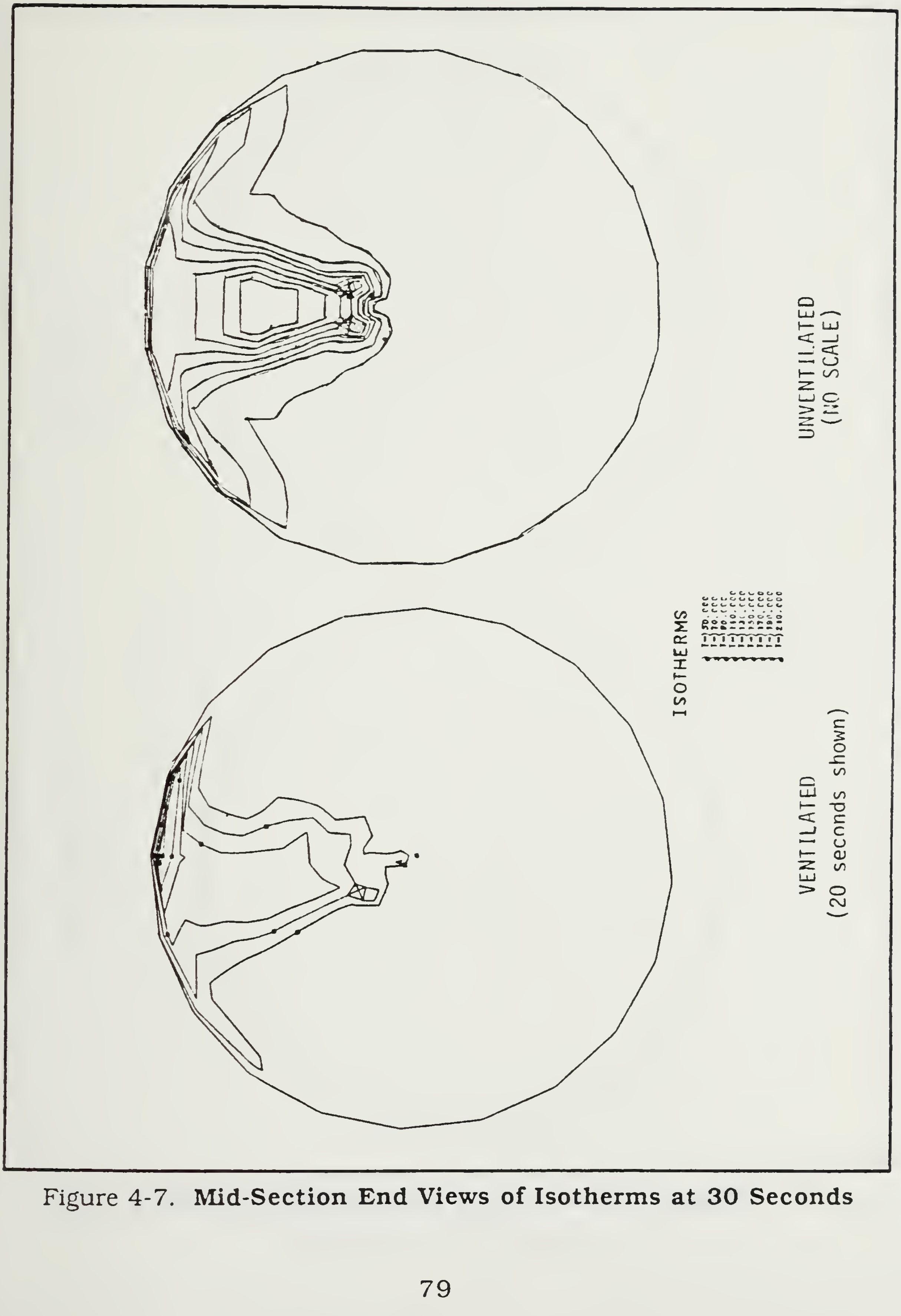

Figure 4-7. **Mid-Section End Views of Isotherms at 30 Seconds**

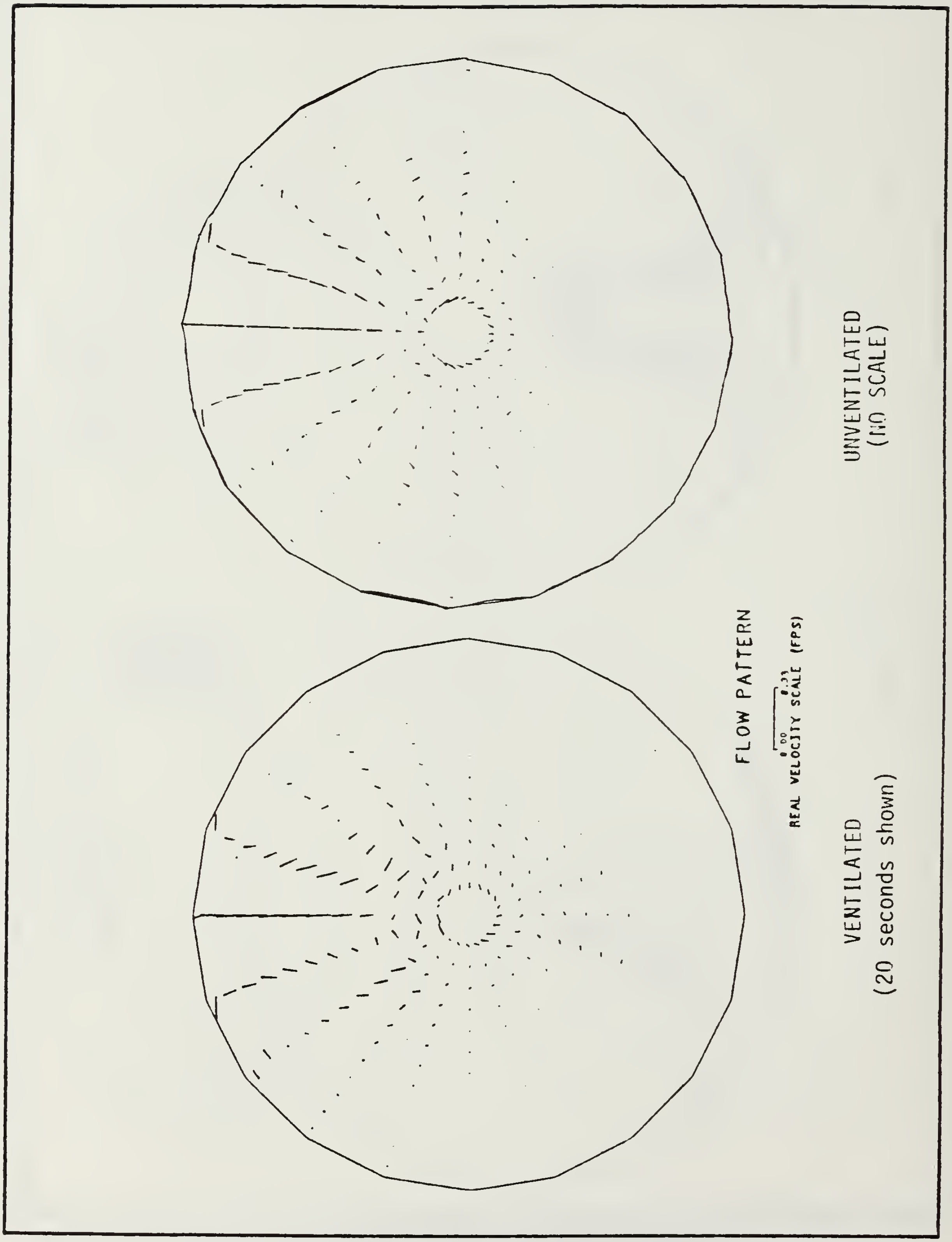

Figure 4-8. **Mid-Section End Views of Velocity Field at 30 Seconds**

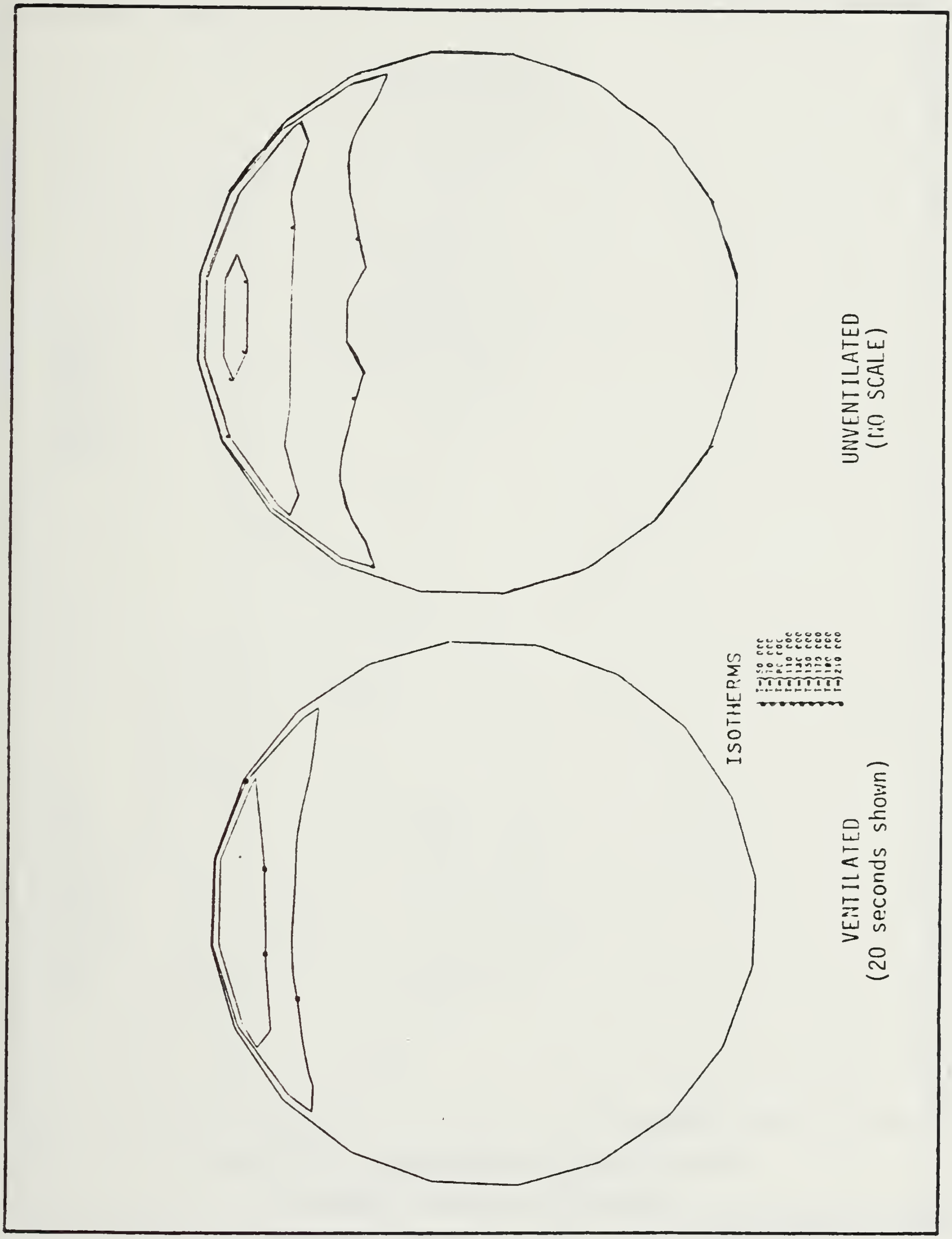

Figure 4-9. **Section View at Base of End Cap of Isotherms at 30 Seconds**

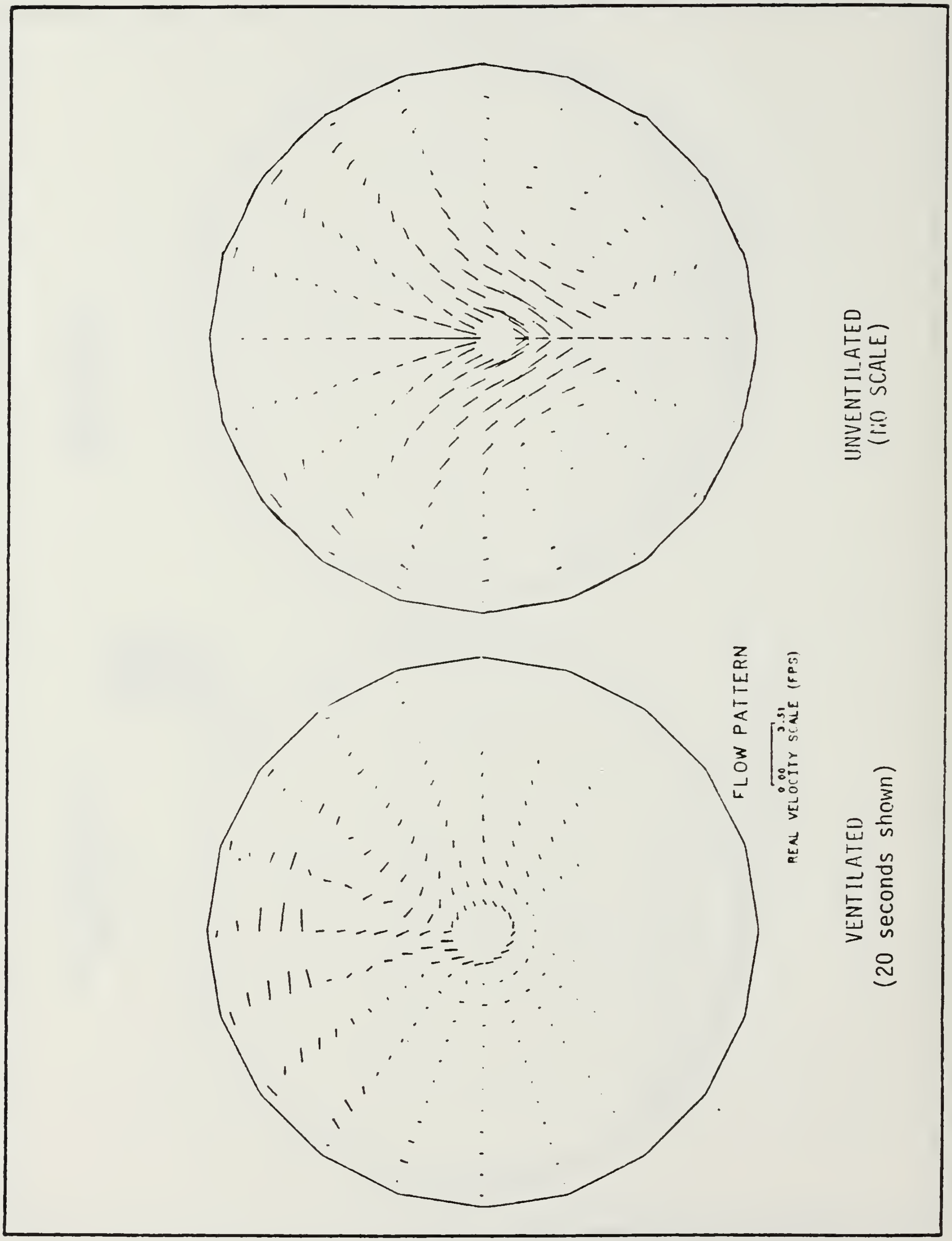

Figure 4-10. **Section View at Base of End Cap of Velocity Field at 30 Seconds**

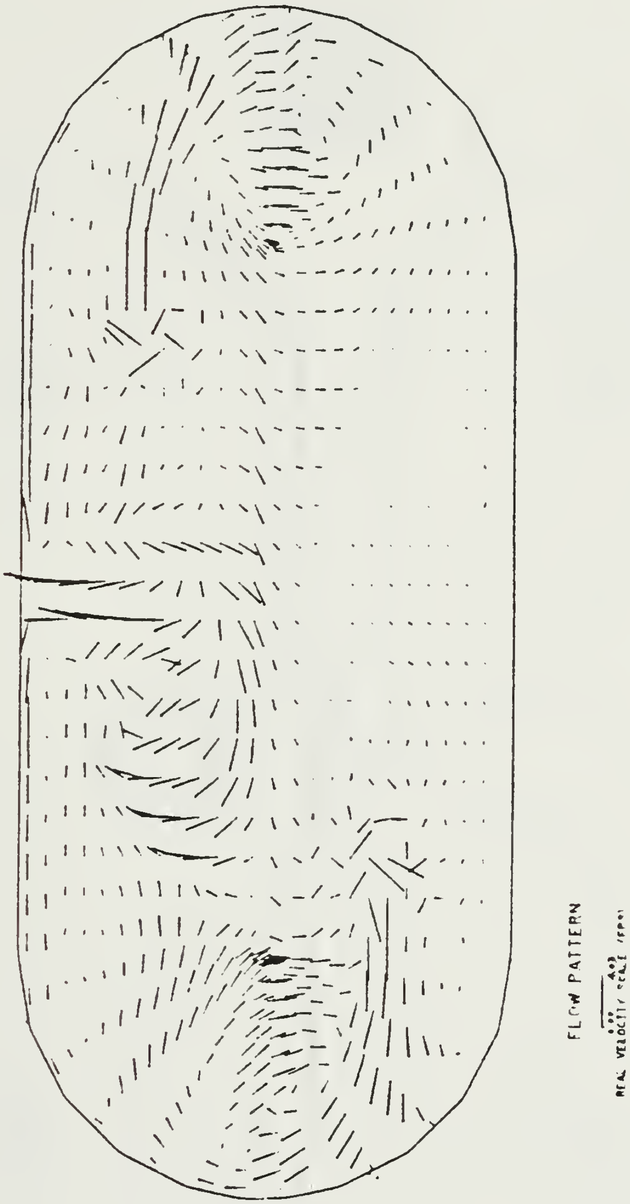

Figure 4-11. **Mid-Section Front View of Velocity Field at 40 Seconds**

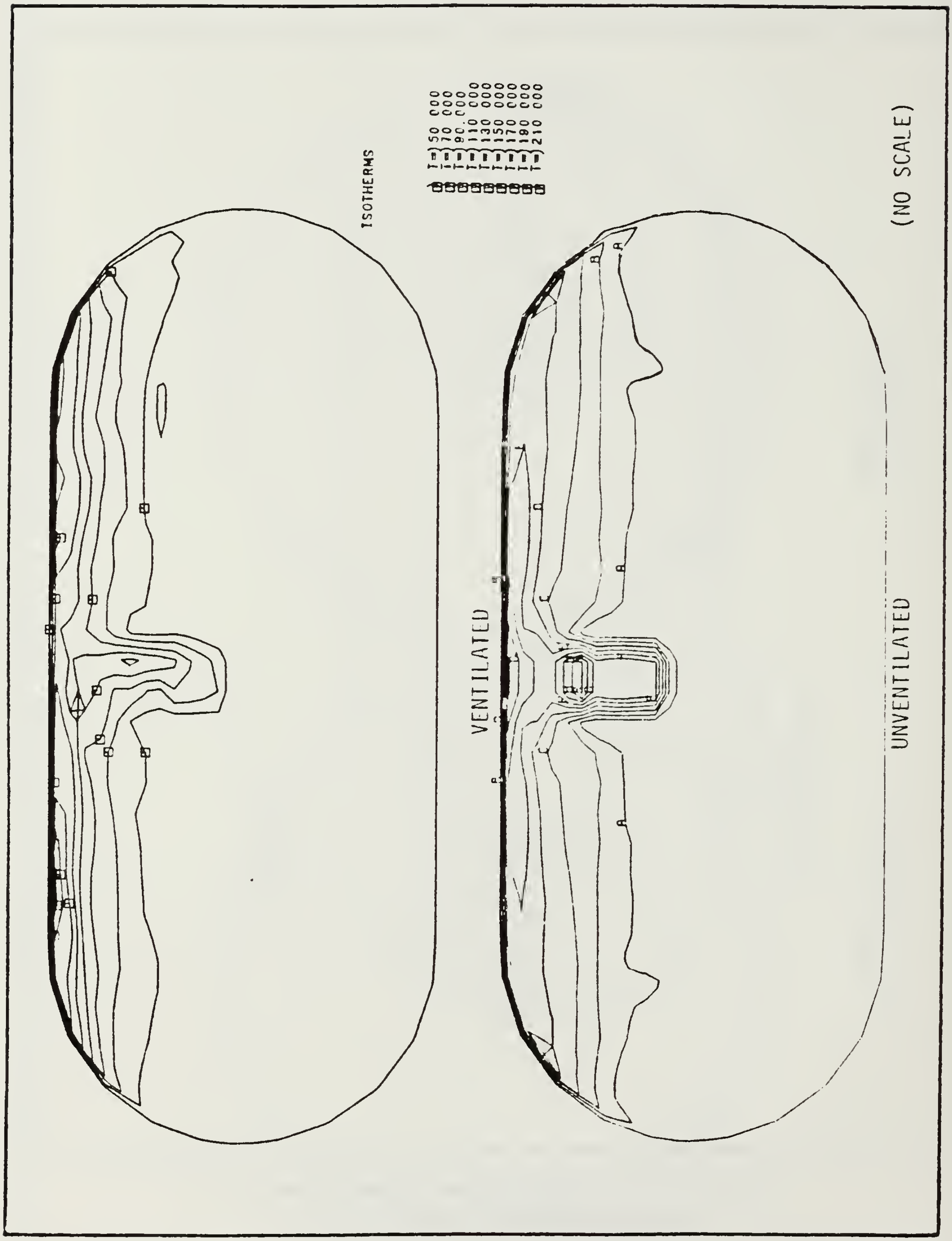

Figure 4-12. **Mid-Section Front Views of Isotherms at 60 Seconds**

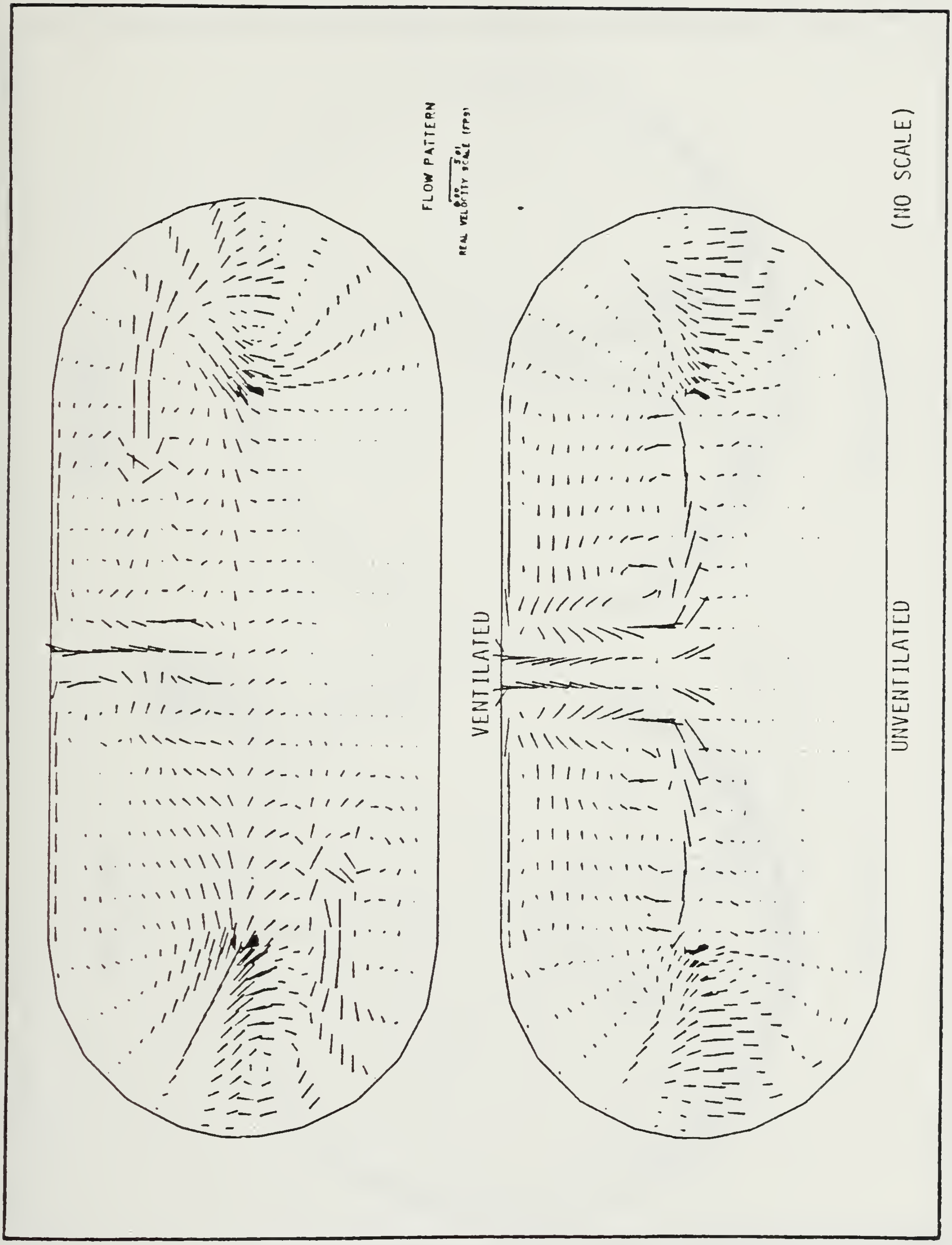

Figure 4-13. **Mid-Section Front Views of Velocity Field at 60 Seconds**

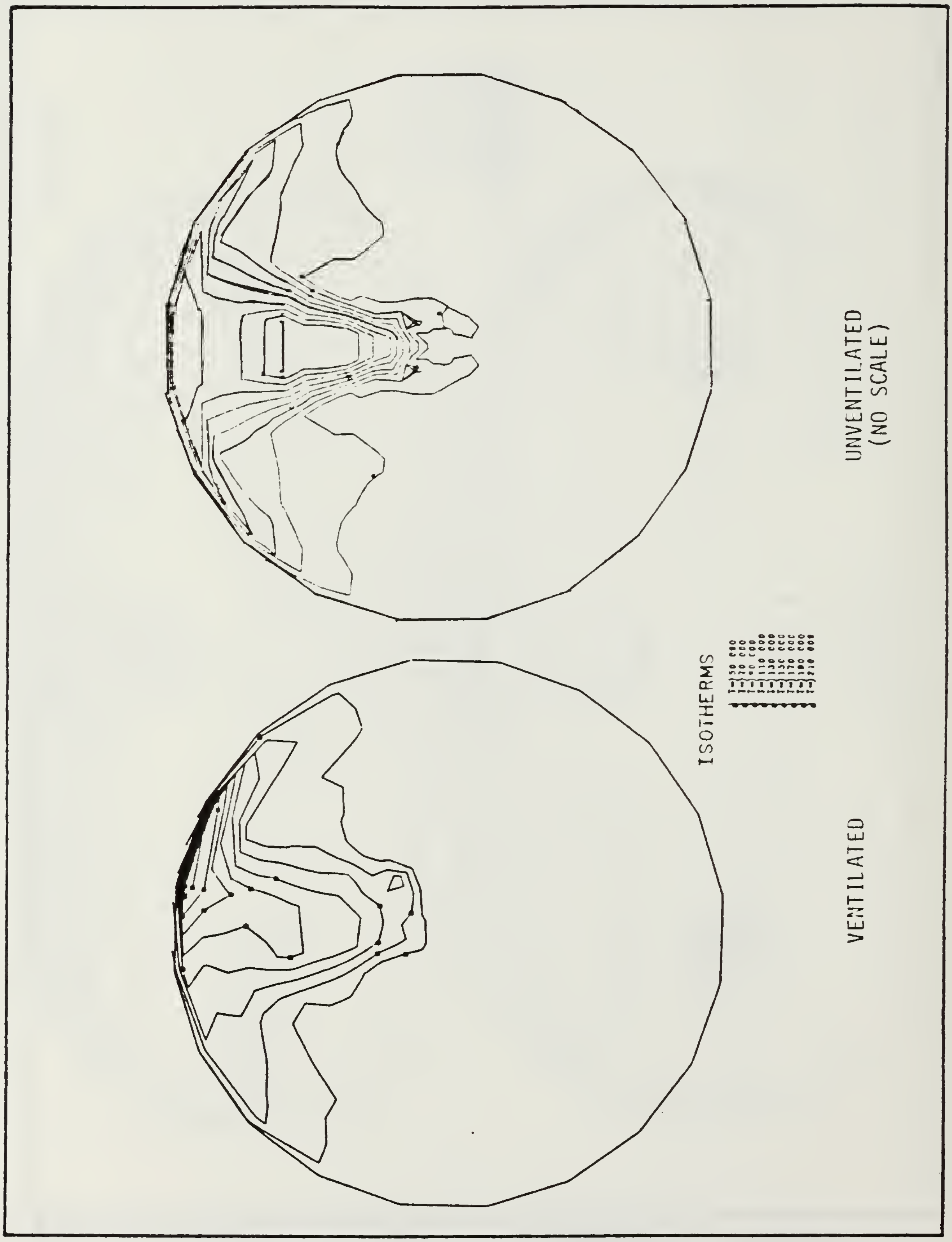

Figure 4-14. **Mid-Section End Views of Isotherms at 60 Seconds**

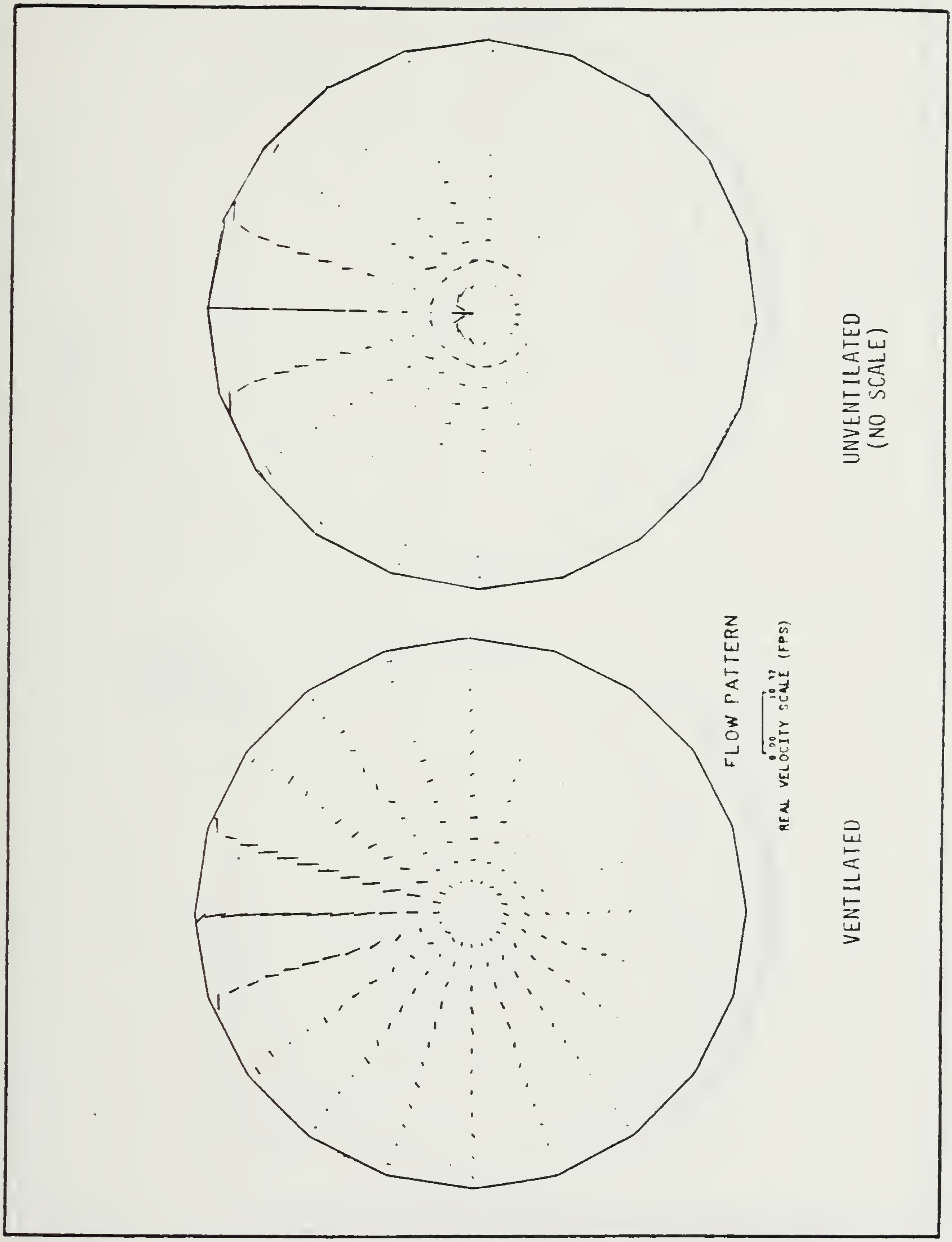

Figure 4-15. **Mid-Section End Views of Velocity Field at 60 Seconds**

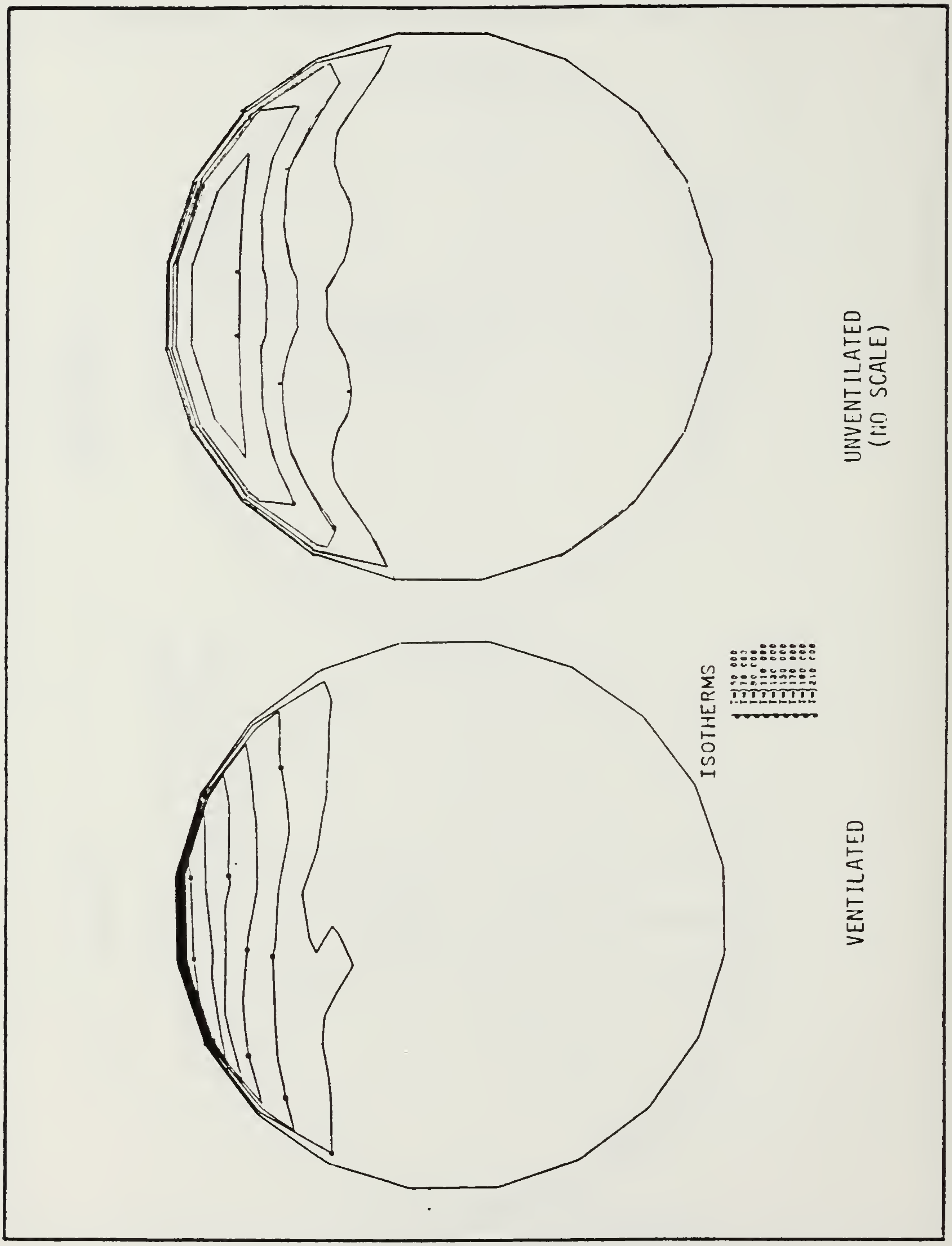

Figure 4-16. **Section View at Base of End Cap of Isotherms at 60 Seconds**

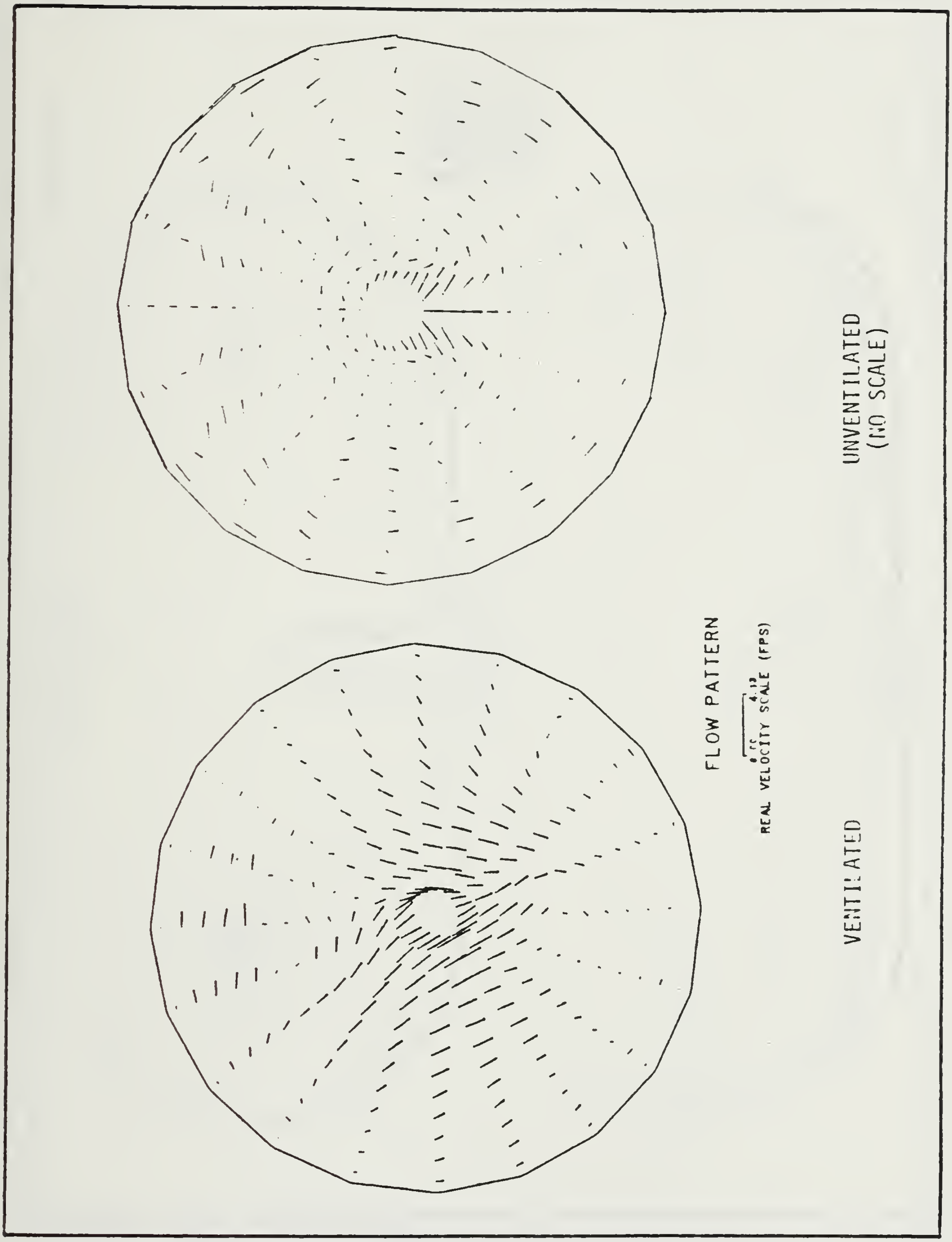

Figure 4-17. **Section View at Base of End Cap of Velocity Field at 60 Seconds**

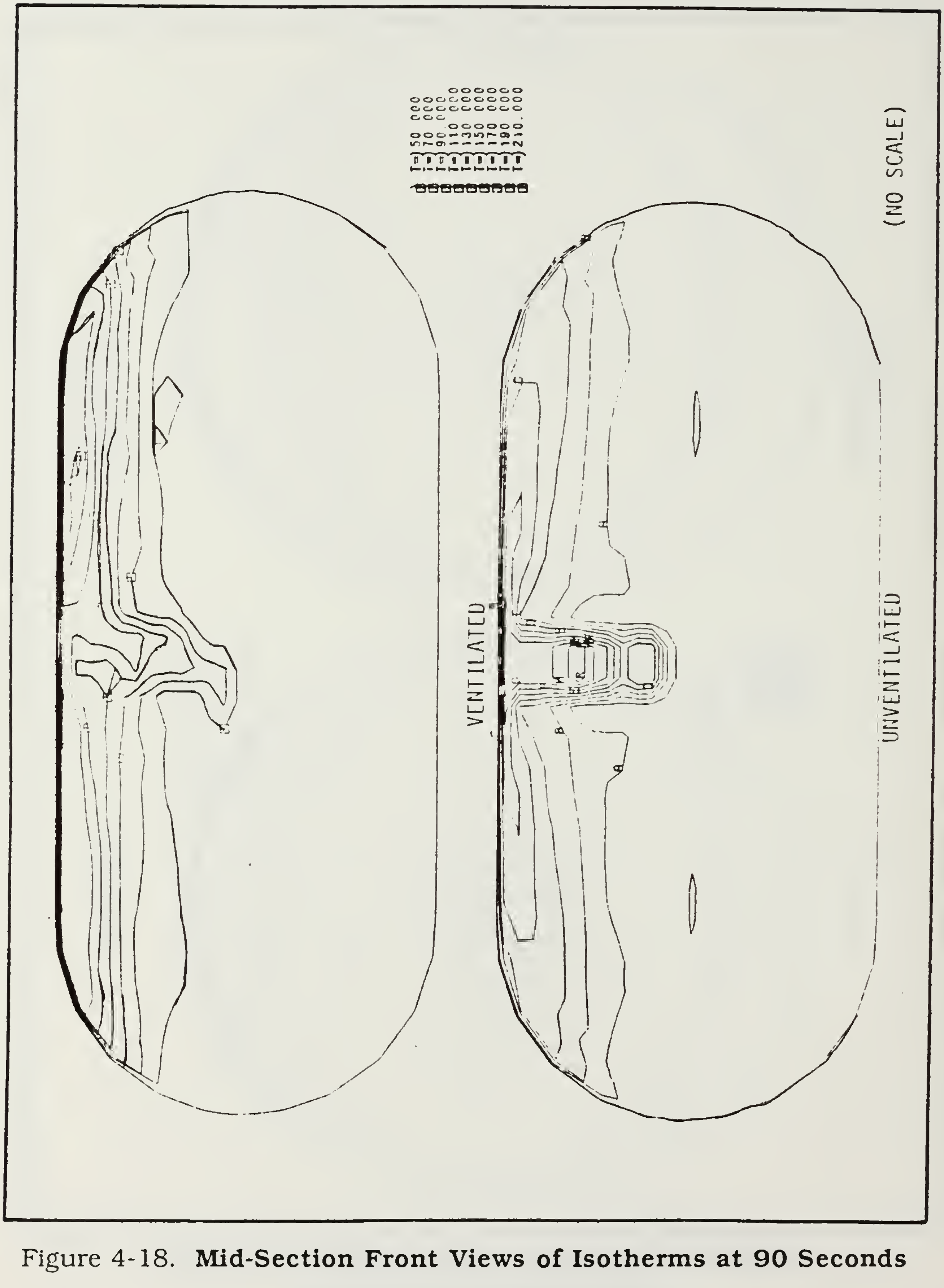

Figure 4-18. **Mid-Section Front Views of Isotherms at 90 Seconds**

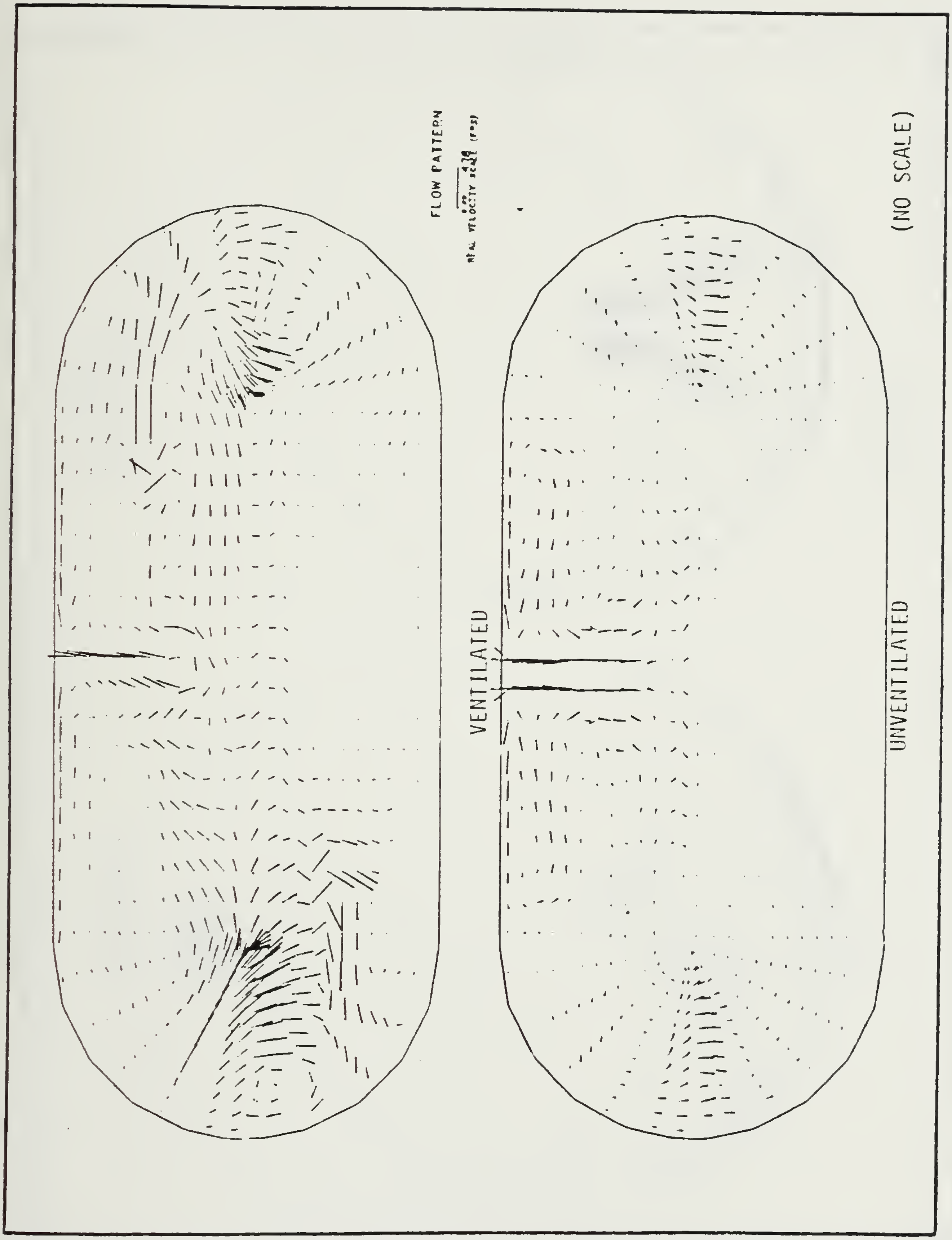

Figure 4-19. **Mid-Section Front Views of Velocity Field at 90 Seconds**

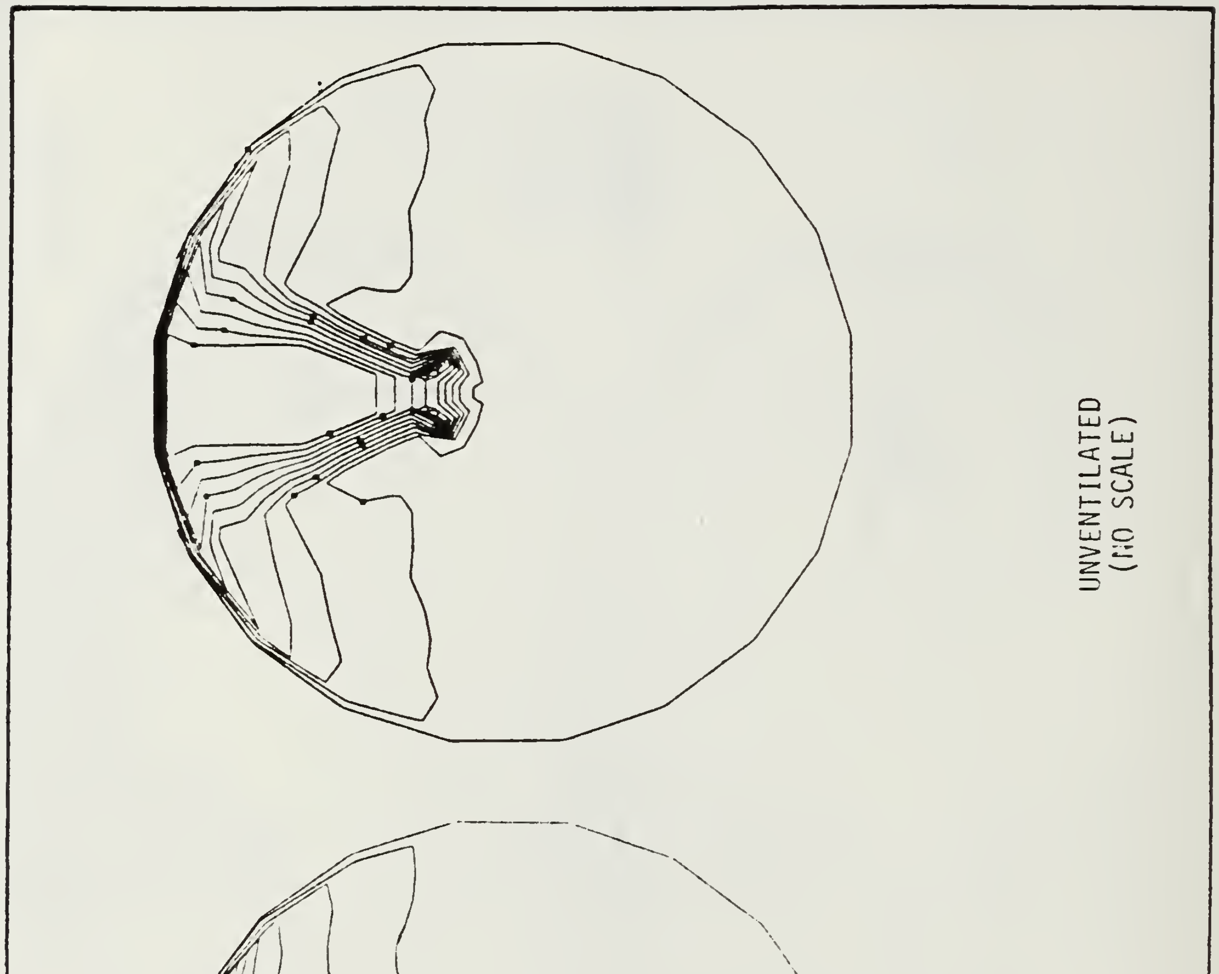

Figure 4-20. **Mid-Section End Views of Isotherms at 90 Seconds**

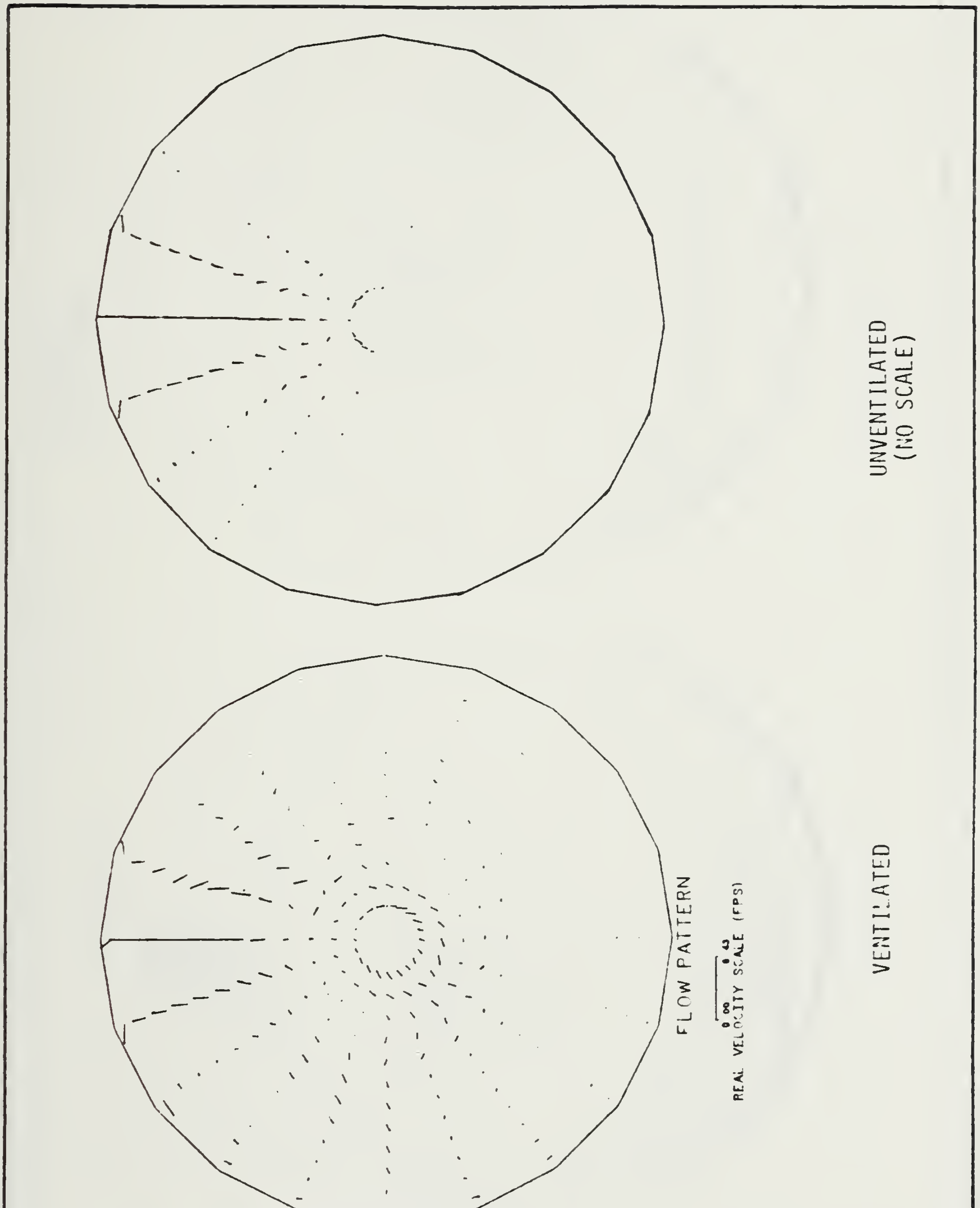

Figure 4-21. **Mid-Section End Views of Velocity Field at 90 Seconds**

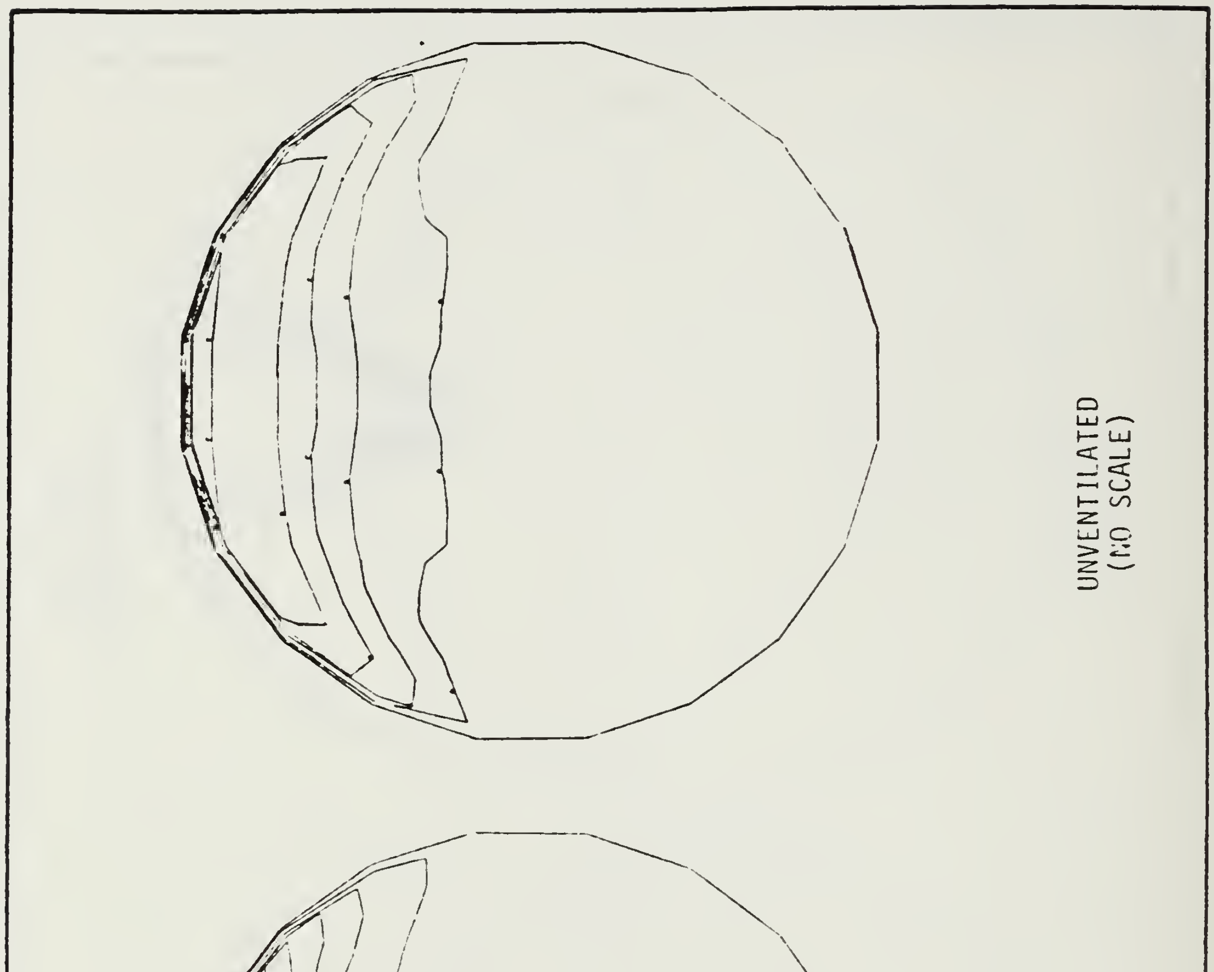

Figure 4-22. **Section View at Base of End Cap of Isotherms at 90 Seconds**

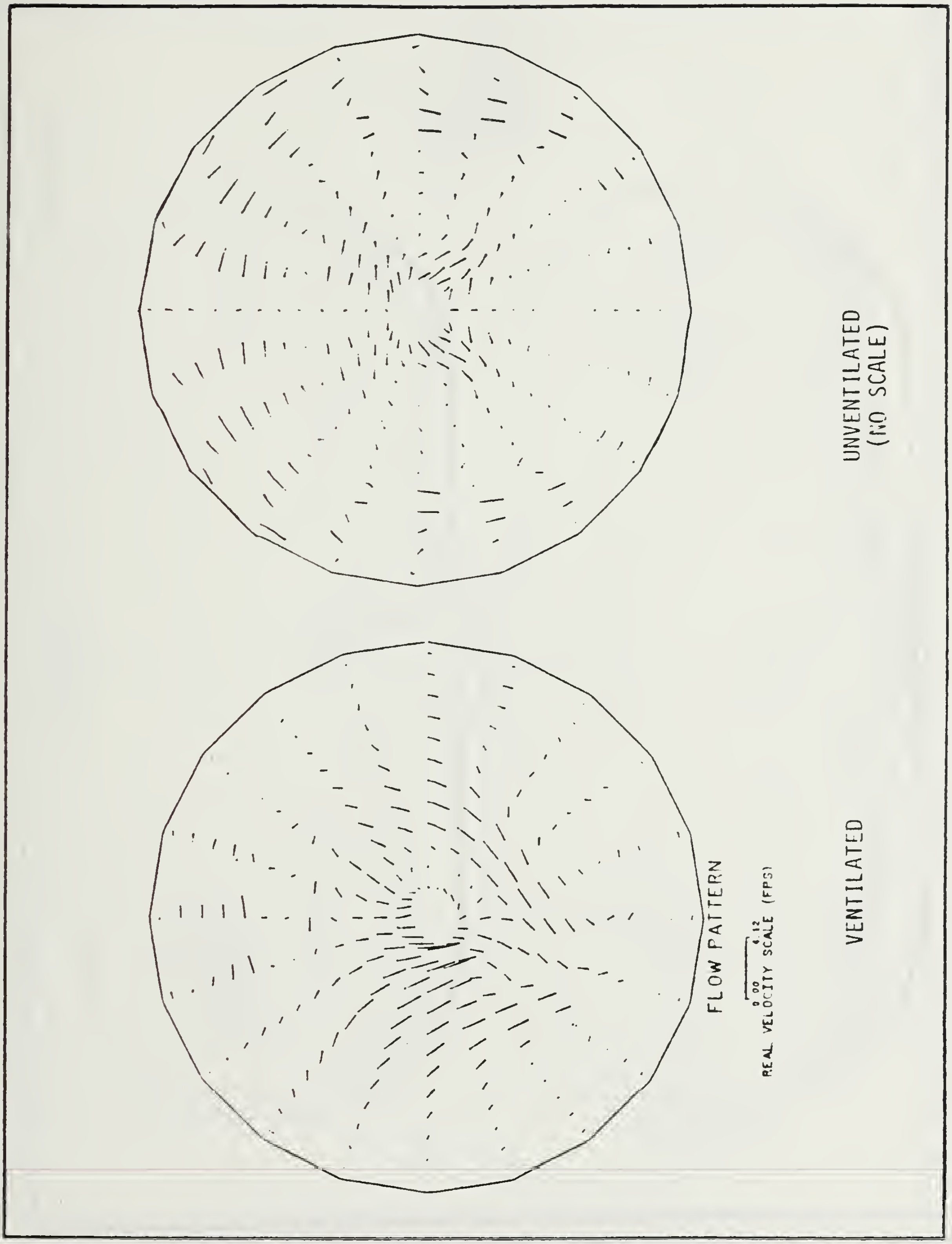

Figure 4-23. **Section View at Base of End Cap of Velocity Field at 90 Seconds**

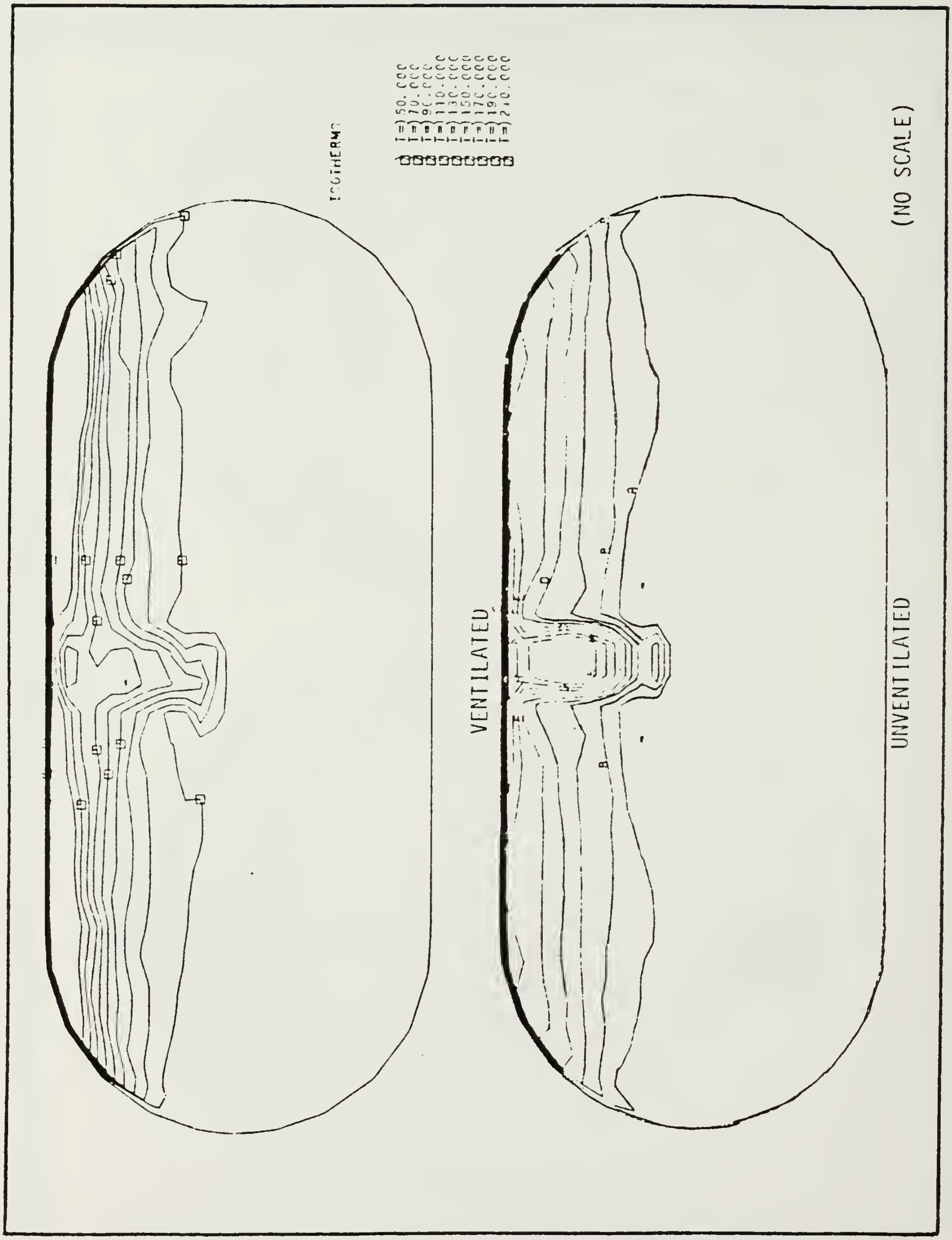

Figure 4-24. **Mid-Section Front Views of Isotherms at 120 Seconds**

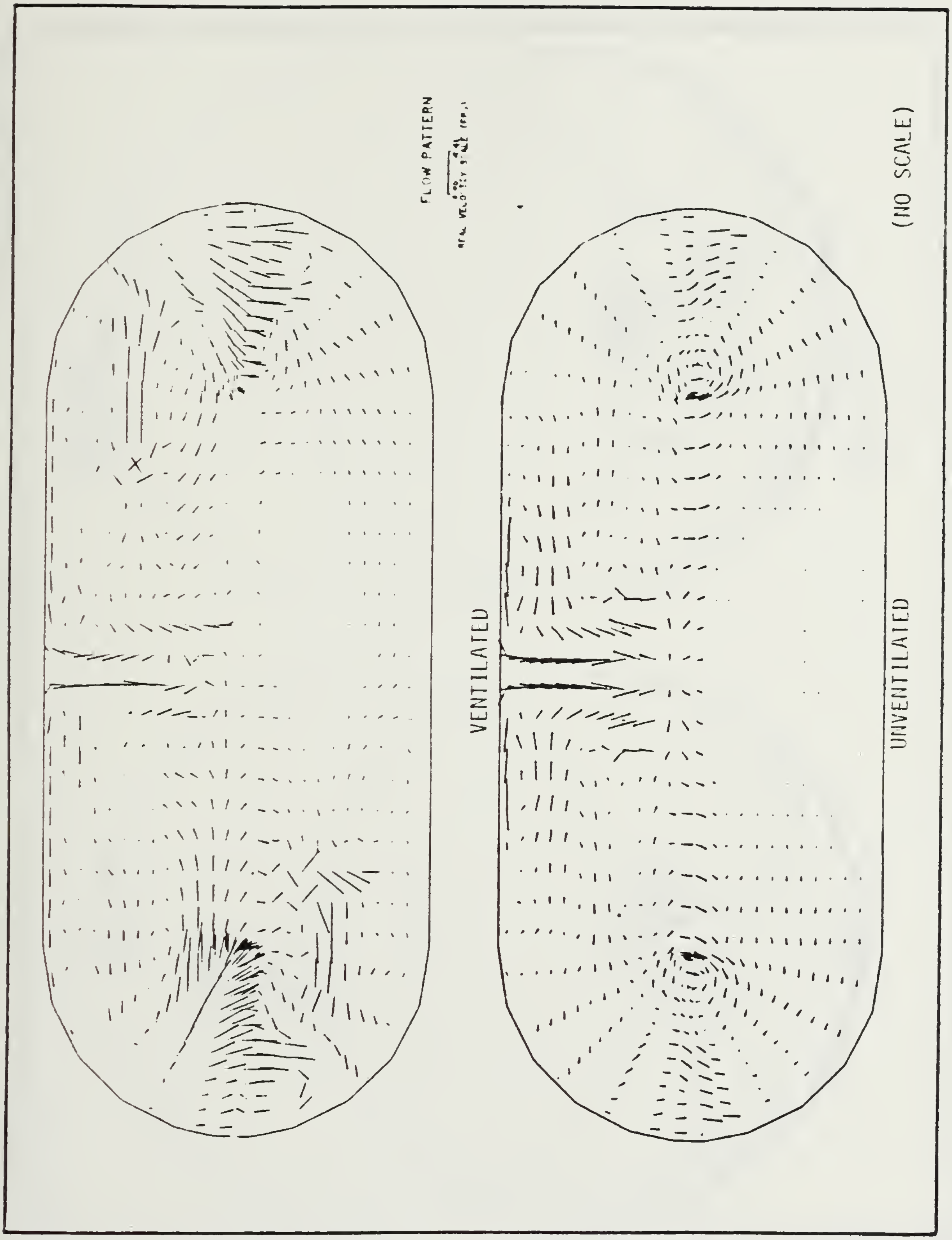

Figure 4-25. **Mid-Section Front Views of Velocity Field at 120 Seconds**

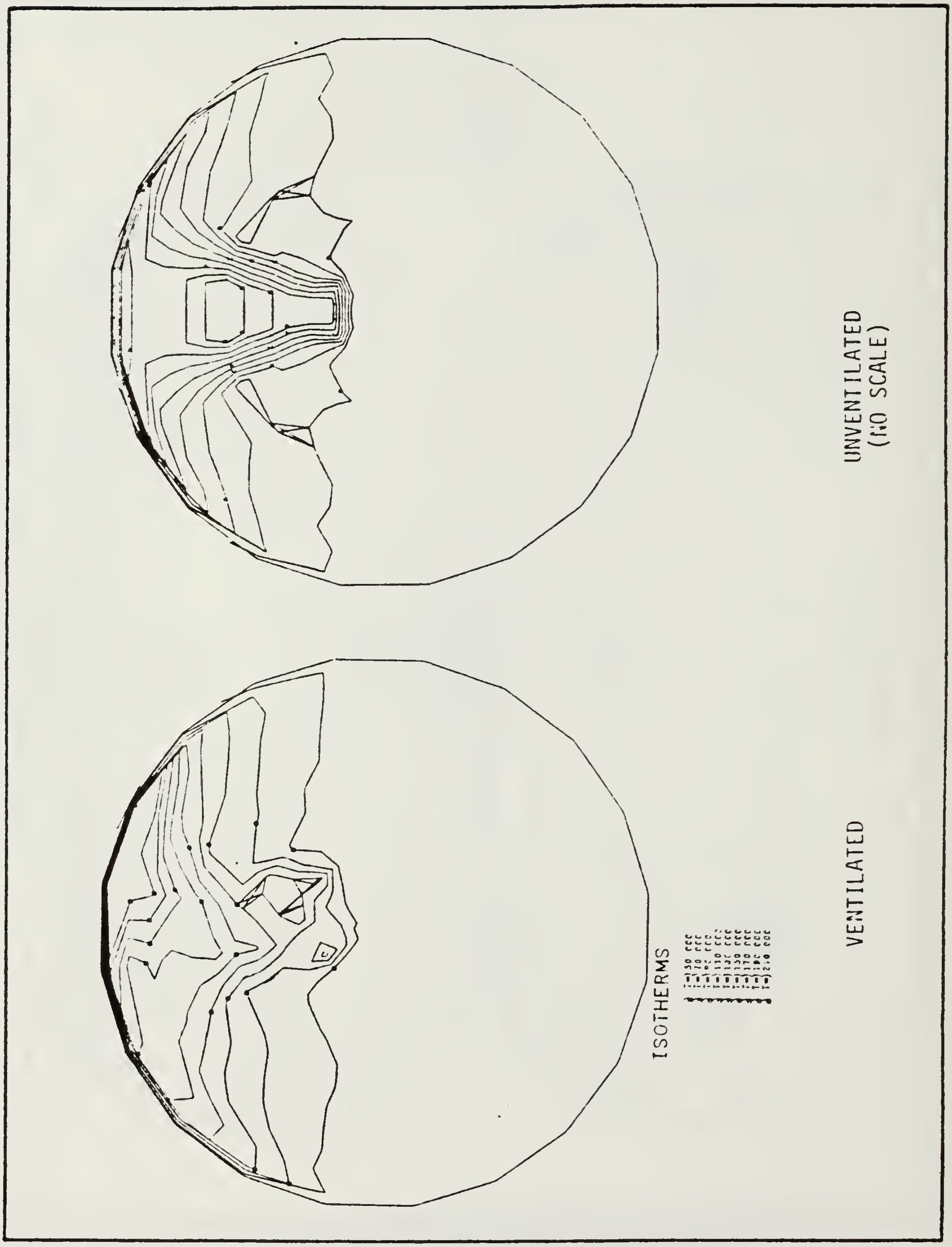

Figure 4-26. **Mid-Section End Views of Isotherms at 120 Seconds**

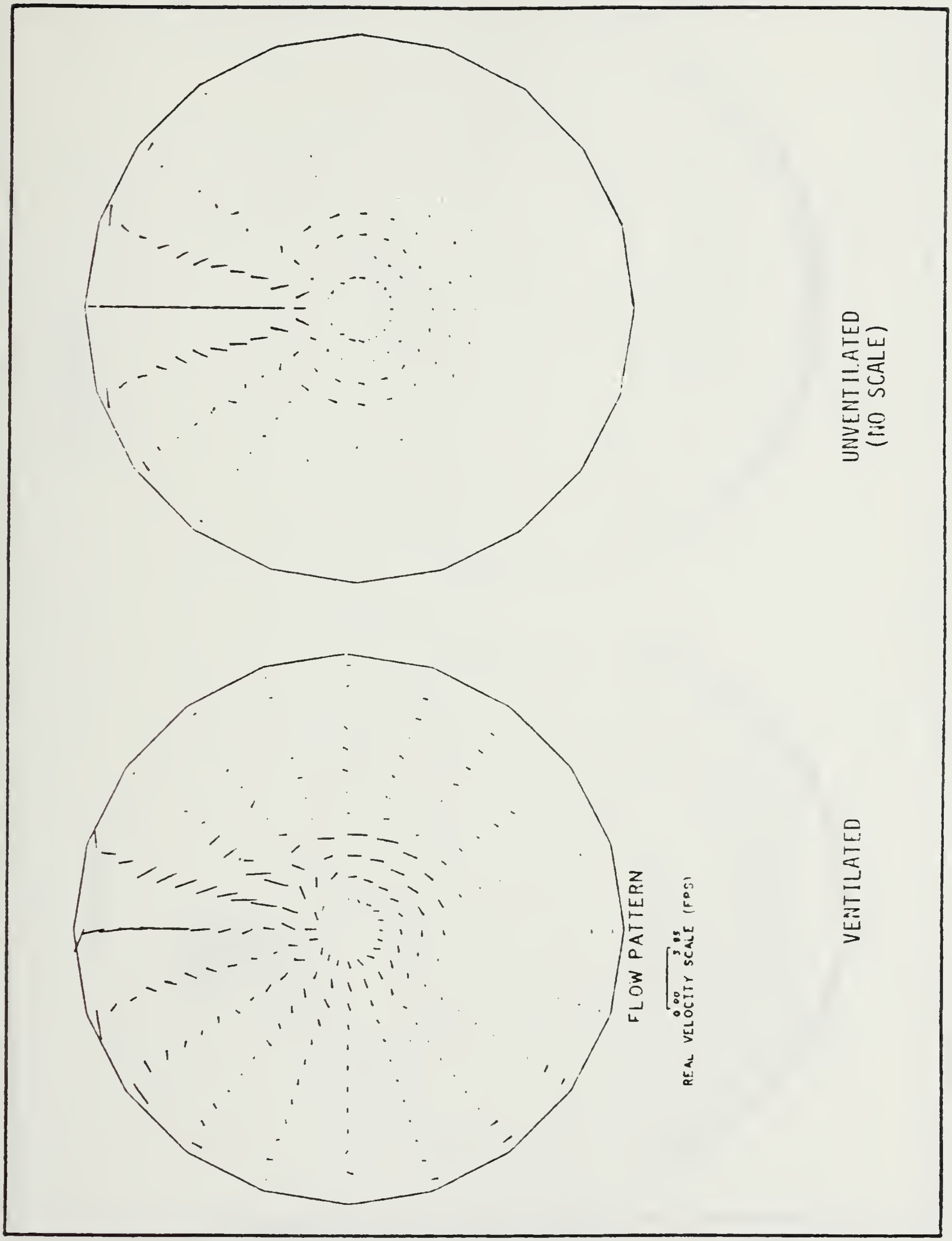

Figure 4-27. **Mid-Section End Views of Velocity Field at 120 Seconds**

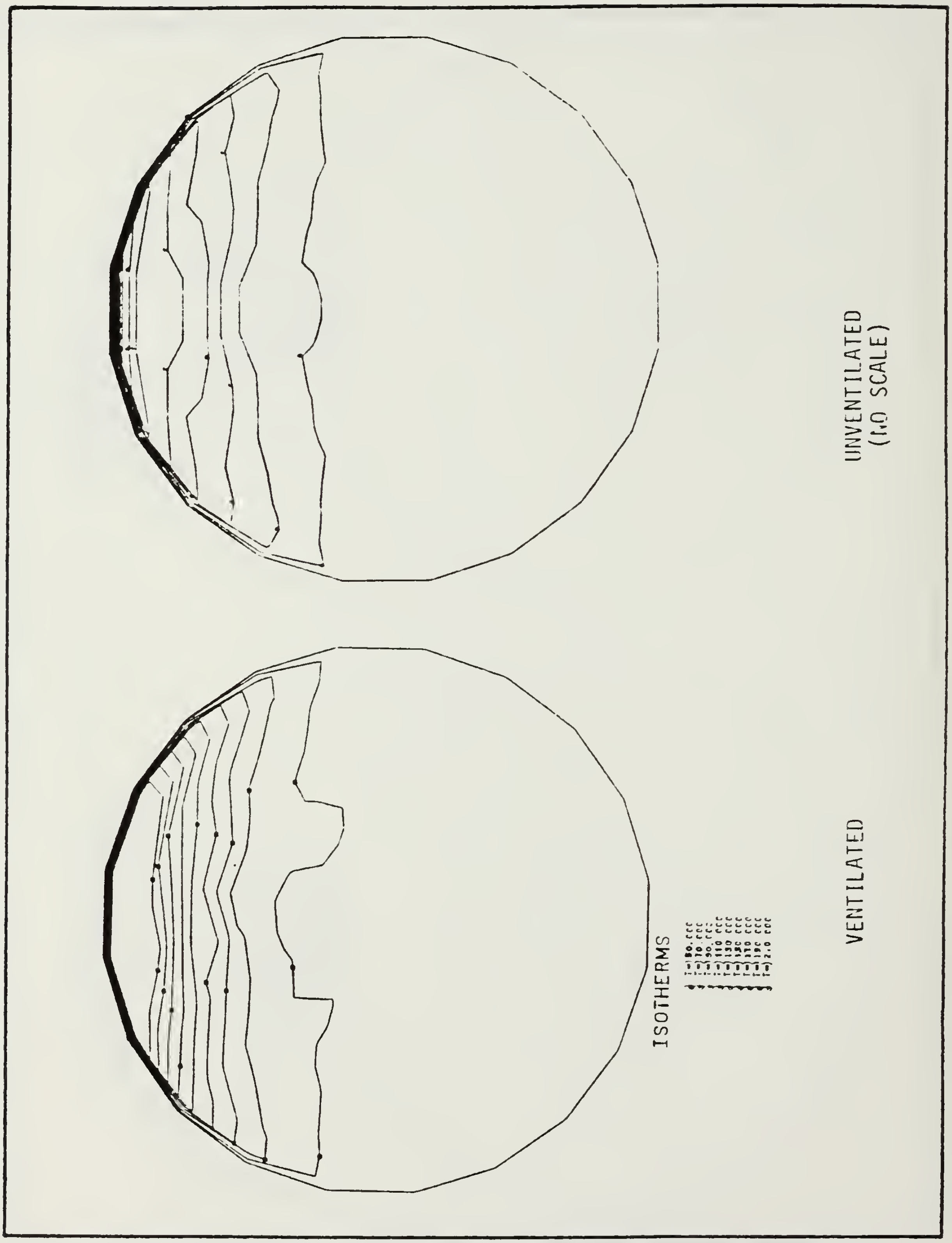

Figure 4-28. **Section View at Base of End Cap of Isotherms at 120 Seconds**

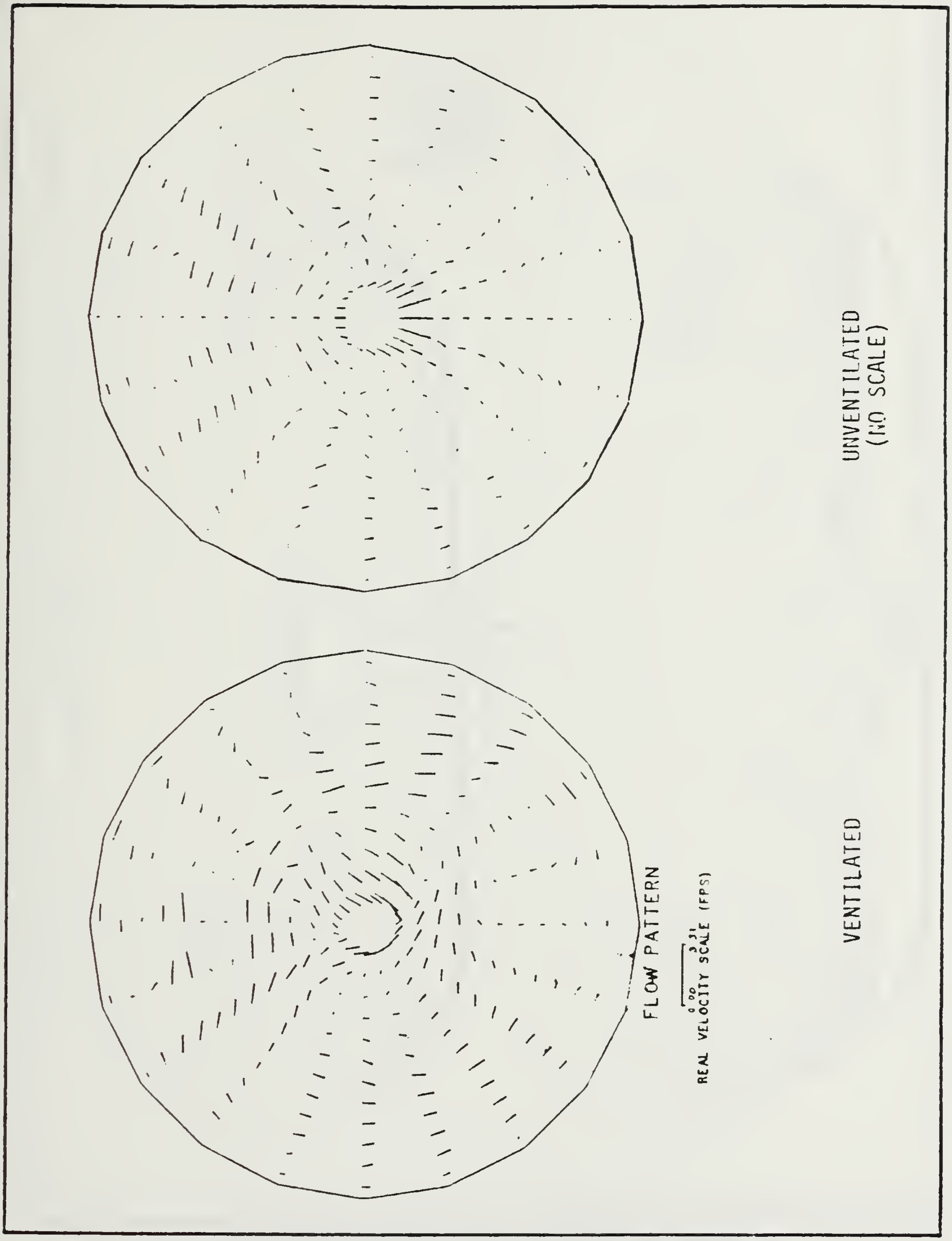

Figure 4-29. **Section View at Base of End Cap of Velocity Field at 120 Seconds**

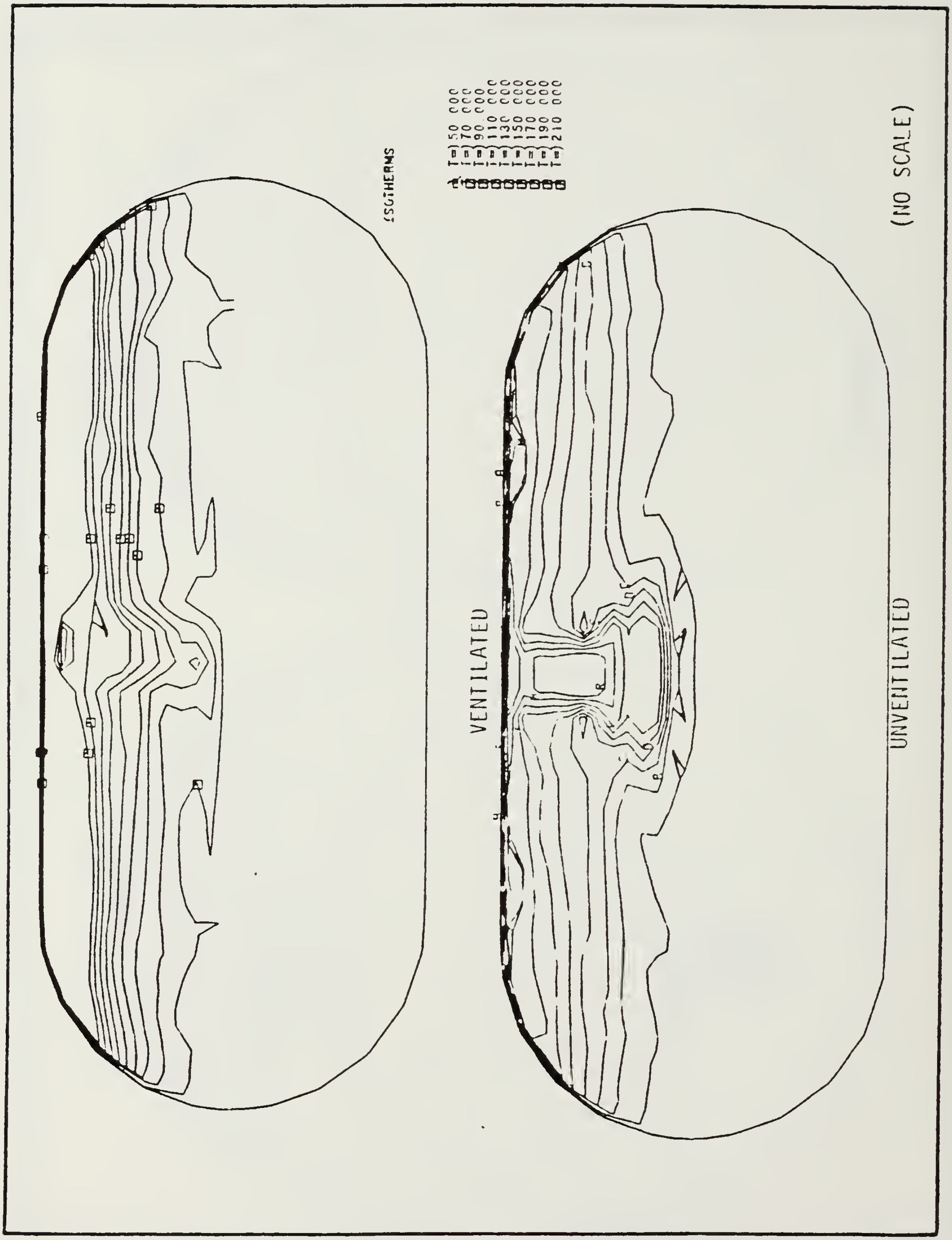

Figure 4-30. **Mid-Section Front Views of Isotherms at 150 Seconds**

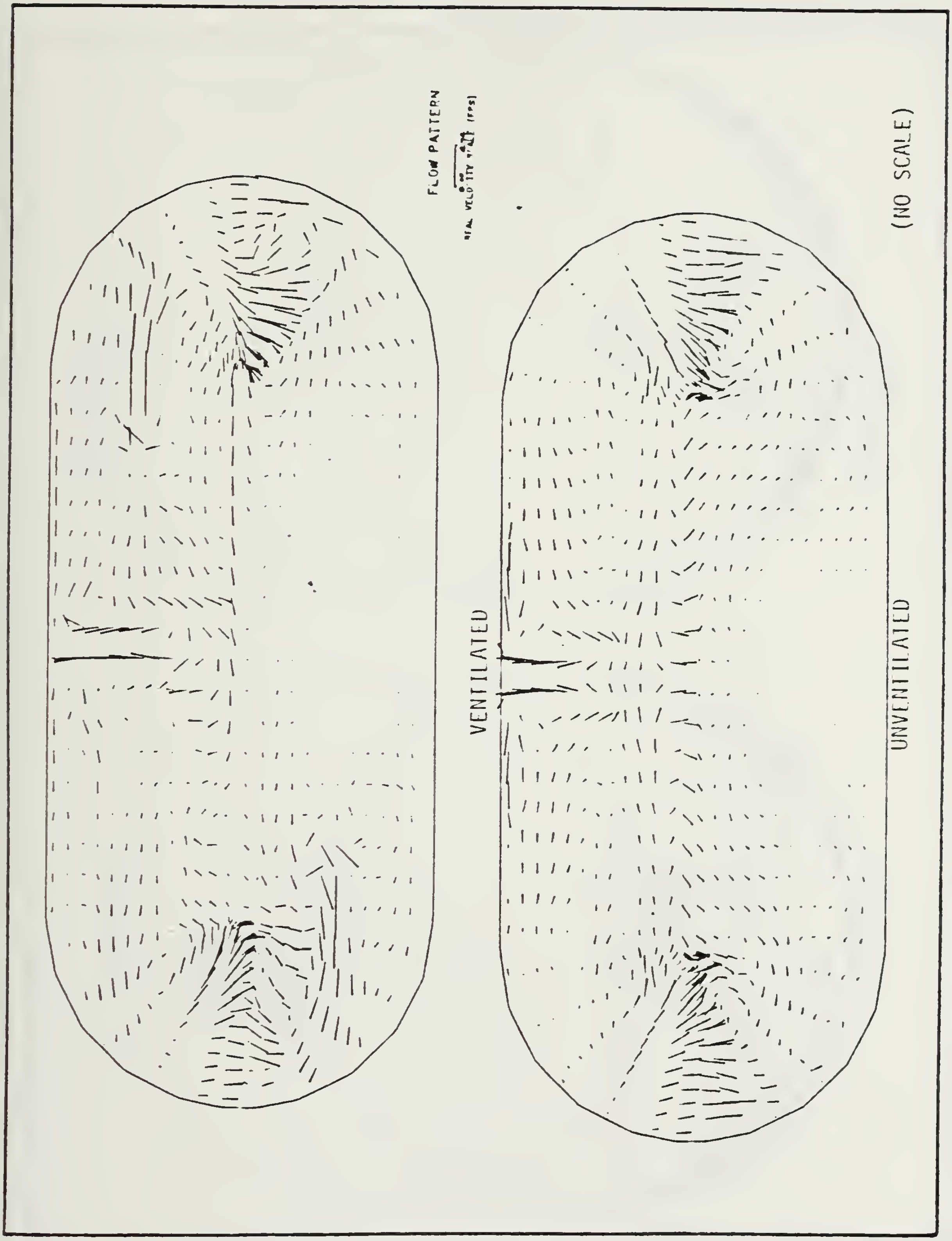

Figure 4-31. **Mid-Section Front Views of Velocity Field at 150 Seconds**

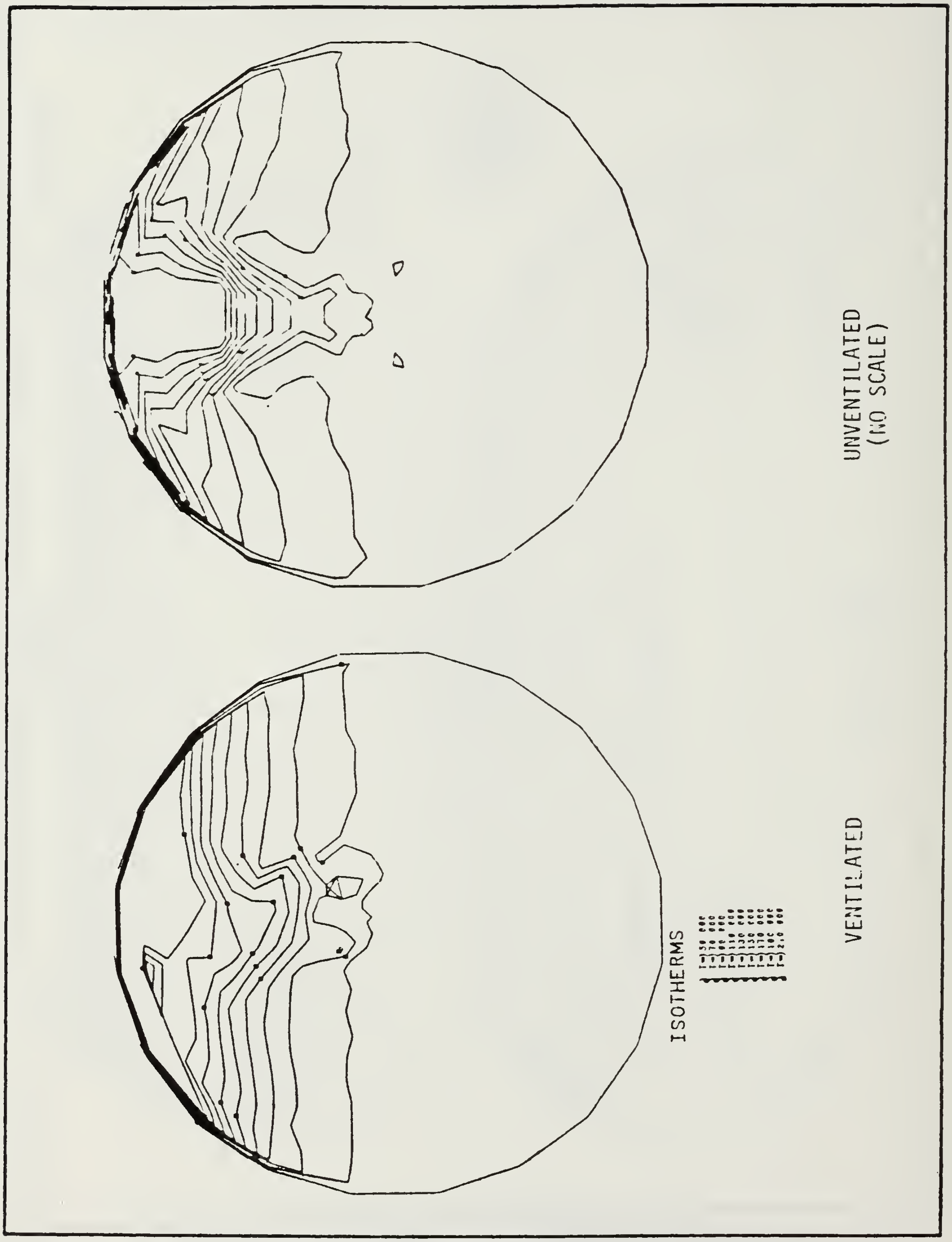

Figure 4-32. **Mid-Section End Views of Isotherms at 150 Seconds**

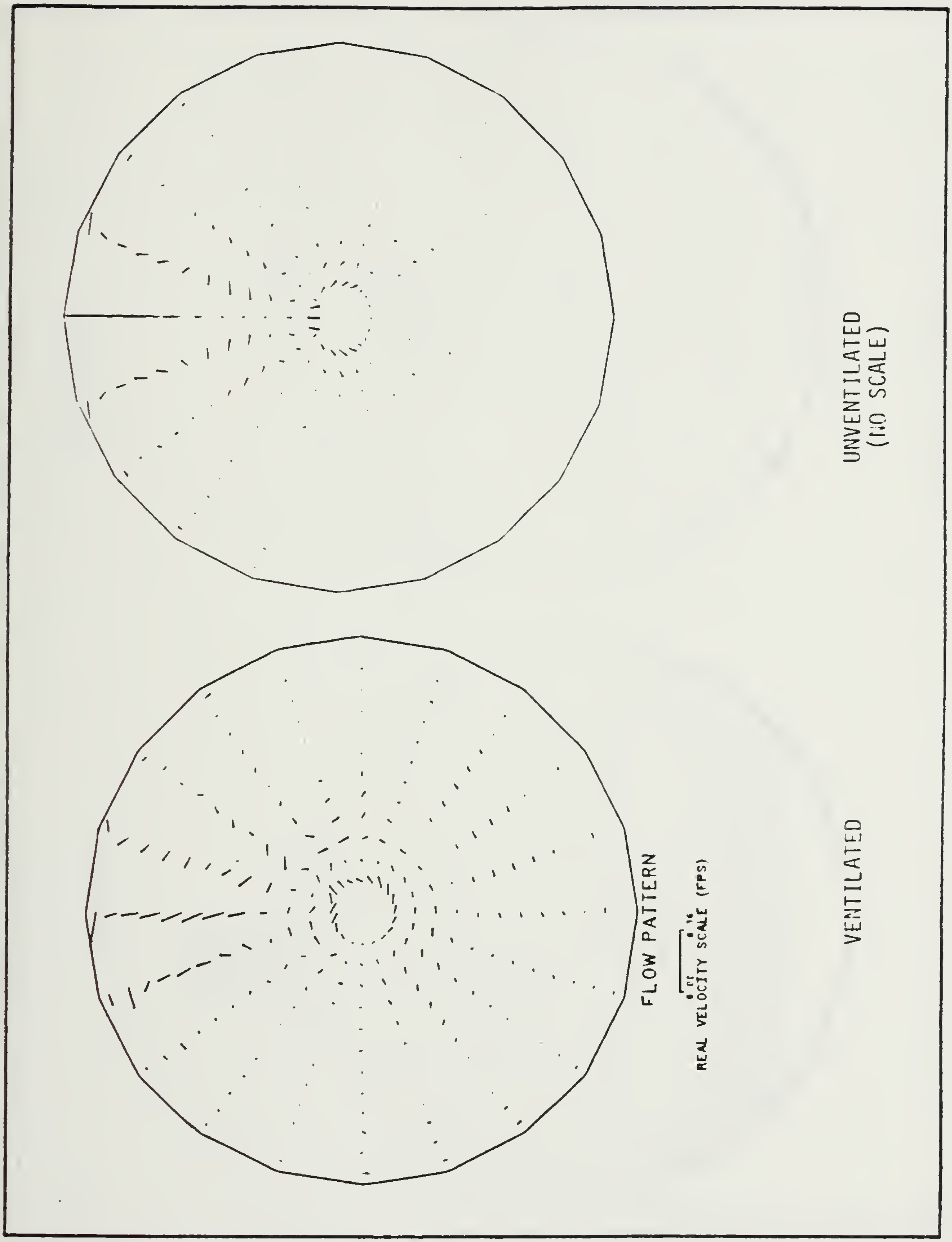

Figure 4-33. **Mid-Section End Views of Velocity Field at 150 Seconds**

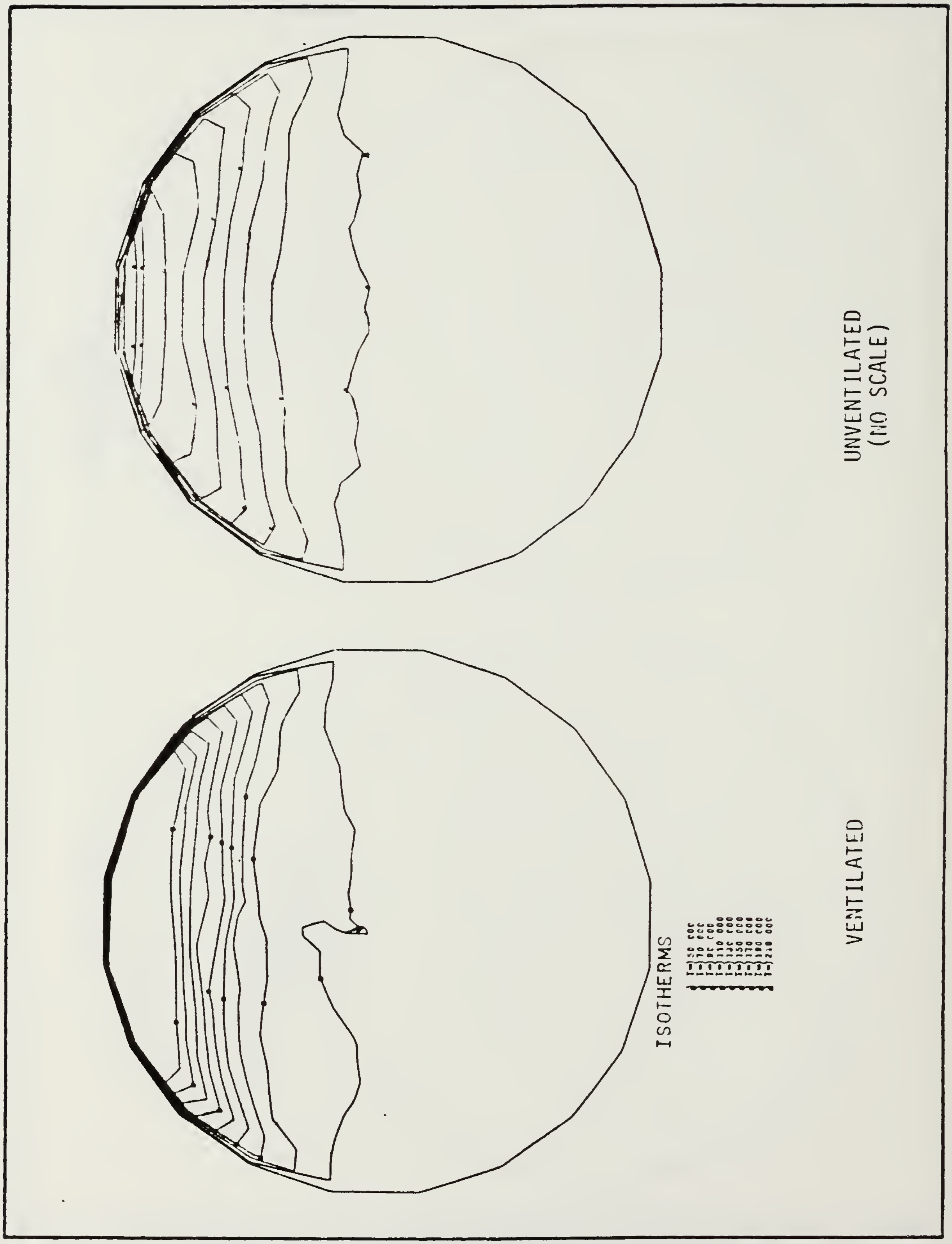

Figure 4-34. **Section View at Base of End Cap of Isotherms at 150 Seconds**

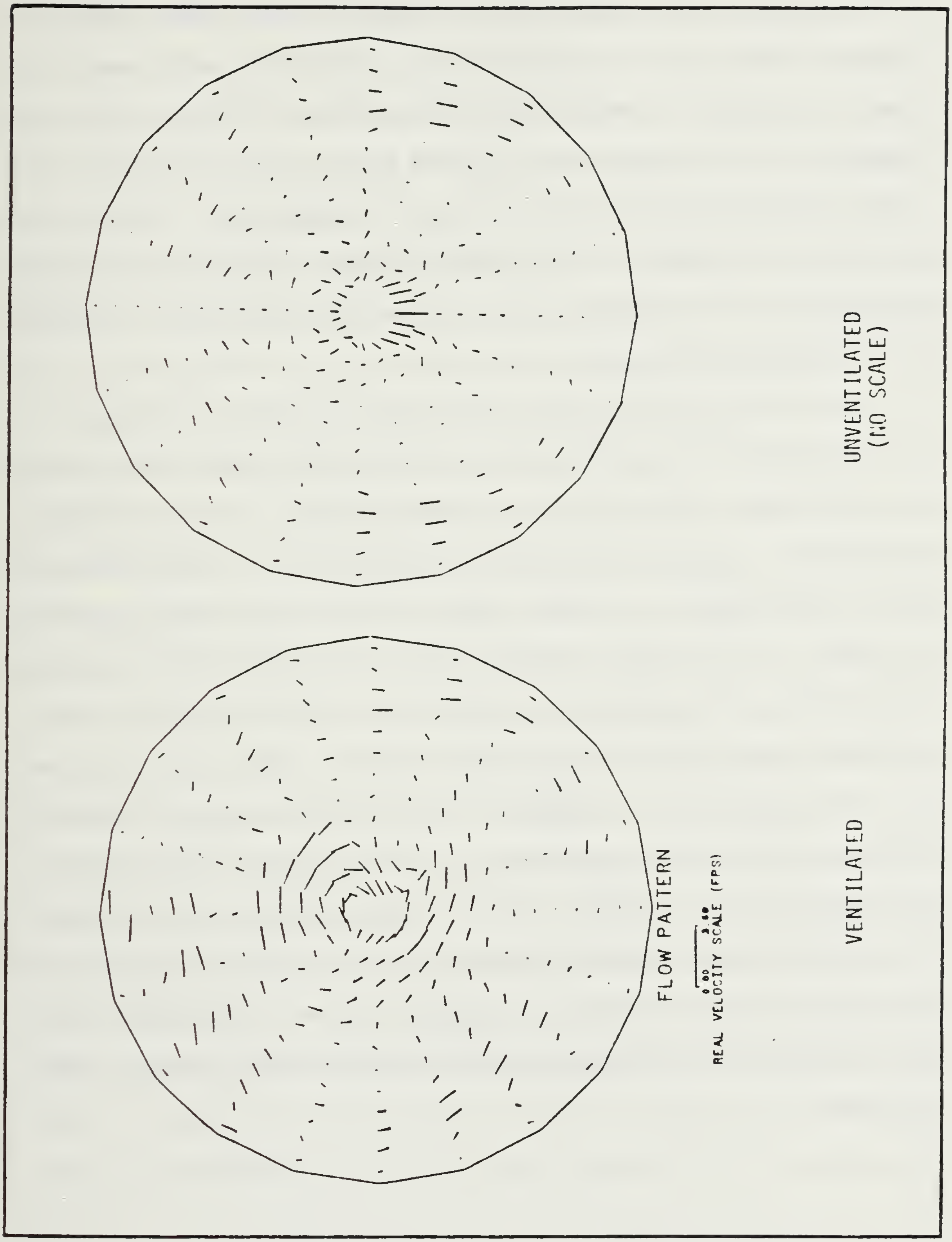

Figure 4-35. **Section View at Base of End Cap of Velocity Field at 150 Seconds**

and it dominates the local velocity field. As can be seen in Figures 4.12 through 4.33, the plume continues to dominate the field throughout the fire. The plume begins at the heat source and flows straight up until it reaches the ceiling, then it divides and flows towards either end of the vessel. In the local area of the fire, there is some entrainment of the field due to the plume flow. Due to the strength of the plume velocity, and the absence of any strong global circulation, the flame plume divides the velocity field in half, isolating the north and south regions.

The flow in the hot ceiling layer does not appear to have strong enough momentum to carry it into the lower half of the tank, even in the south end, where the fan augments the flow. Instead, the flow recirculates into the tank interior, resulting in a downward-biased flow. It then returns to the fire region in a somewhat spatially oscillatory path. As can be seen in Figures 4.8, 4.10, and the other end views of the velocity field, there is a spiral flow circulation pattern in the ventilated case. This creates a more stagnant region to the right of the vertical center line. Figures 4.7, 4.9, and the other end views of isotherms show higher temperatures in this stagnant region because the heated fluid is not being convectively transferred. It also makes the conductive heat transfer through the tank wall in the region more important, as the temperature is higher. In the nonventilated case, the flow fields and isotherms appears to be symmetric about the vertical plane.

As mentioned previously, the velocity of the fans is a constant 3.18 feet per second. This velocity is on the same order of magnitude as the flame plume, but since each fan is directed only toward the end caps, their impact on the global velocity field is not significant. The fan entrainment creates only a small local disturbance to the global flow pattern. The north fan outlet, in the lower region of the vessel, has little effect upon the global velocity since the global velocity in the region is very small, as seen in the nonventilated case. The fans effect the heat distribution locally, as discussed in the next paragraph.

Figure 4.5 shows a hot layer along the ceiling of the tank, with the temperature highly stratified in the upper region. The lower two-thirds of the tank are still near the initial temperature. This temperature distribution is exactly what the velocity field suggests, flow only in the upper third of the tank, and little flow in the bottom two-thirds. In Figures 4.12, 4.18, 4.24, and 4.30, the temperature stratification continues, but the heated fluid is slowly progressing toward the bottom of the tank. Even at 150 seconds, Figure 4.30 shows that the first isotherm, representing 15 degrees Centigrade above ambient, is only at the middle of the tank. The bottom half of the tank experiences very little temperature increase. In the ventilated case, the isotherms in the north end cap are higher than in the south. This can be attributed to the fans at either end which push up the heated fluid in the north end and push down the heated fluid in the south end. The effect is limited to a small region in the end cap because the fan velocity is relatively low and the flow is parallel to the isotherms.

Since flow is along the stratification, very little mixing of different temperature gases occurs except in the end caps, where flow is forced into a single region. Had the fans been oriented in a direction not parallel to the isotherms, one would expect the temperatures in the lower portion of the tank to be more affected.

One anomaly which appears in the ventilated case is the second circulation at the base of the flame plume on the north side seen in Figure 4.11. The flow in this region is flowing away from the flame plume until it turns upward as it hits the flow returning to the plume from the end caps. It is believed that this is a transient phenomena due to the interaction between the fan and flame plume entrainments. As can be seen in Figures 4.6 and 4.13, the phenomenon has disappeared. Additional data for a time of 45 seconds, not included herein, shows no indication of the second circulation pattern. The effects of this second circulation pattern can be easily seen in the temperature field in Figure 4.11.

Figures 4.36 through 4.39 present the data from the ventilated and nonventilated cases. Figure 4.36 shows that the global pressure in both cases is not very different. The differences can be attributed to two causes. First, the entire field is not at a thermodynamic equilibrium state, and the relationship between the global pressure and a field not in thermodynamic equilibrium is only an estimation. Any change to the field which would closer approach equilibrium, such as the mixing due to the fans, would affect the global pressure. Second,

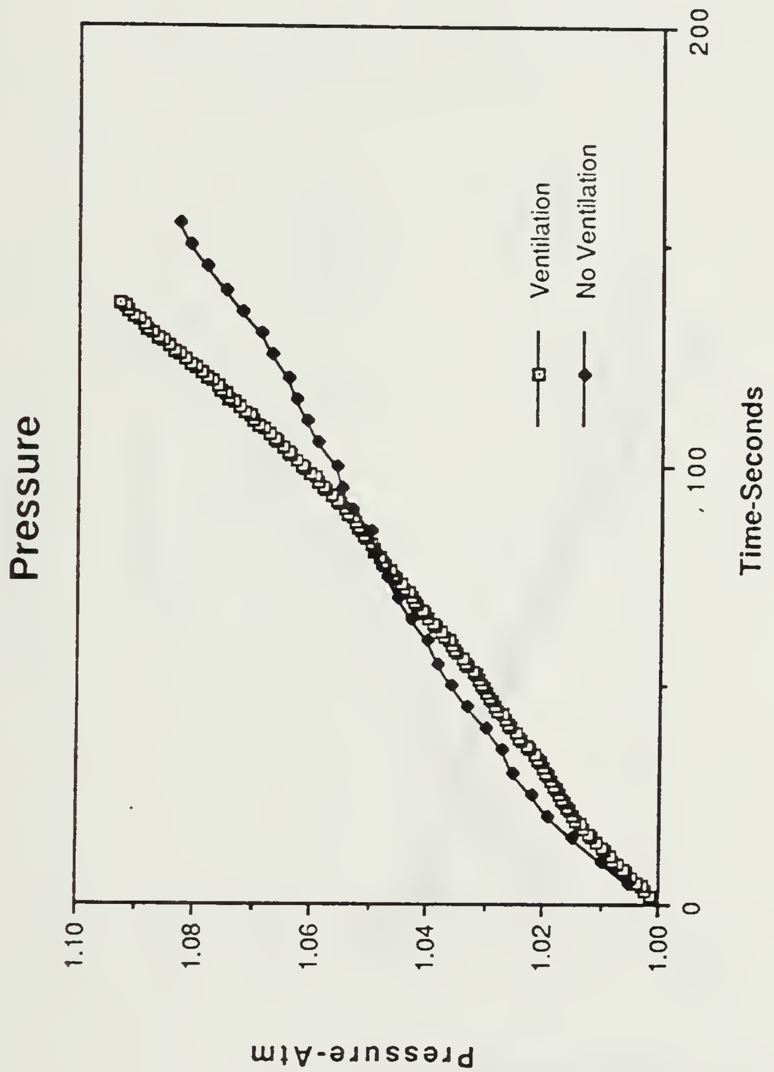

Figure 4-36. **Pressure Curves for the Ventilated and Nonventilated Cases**

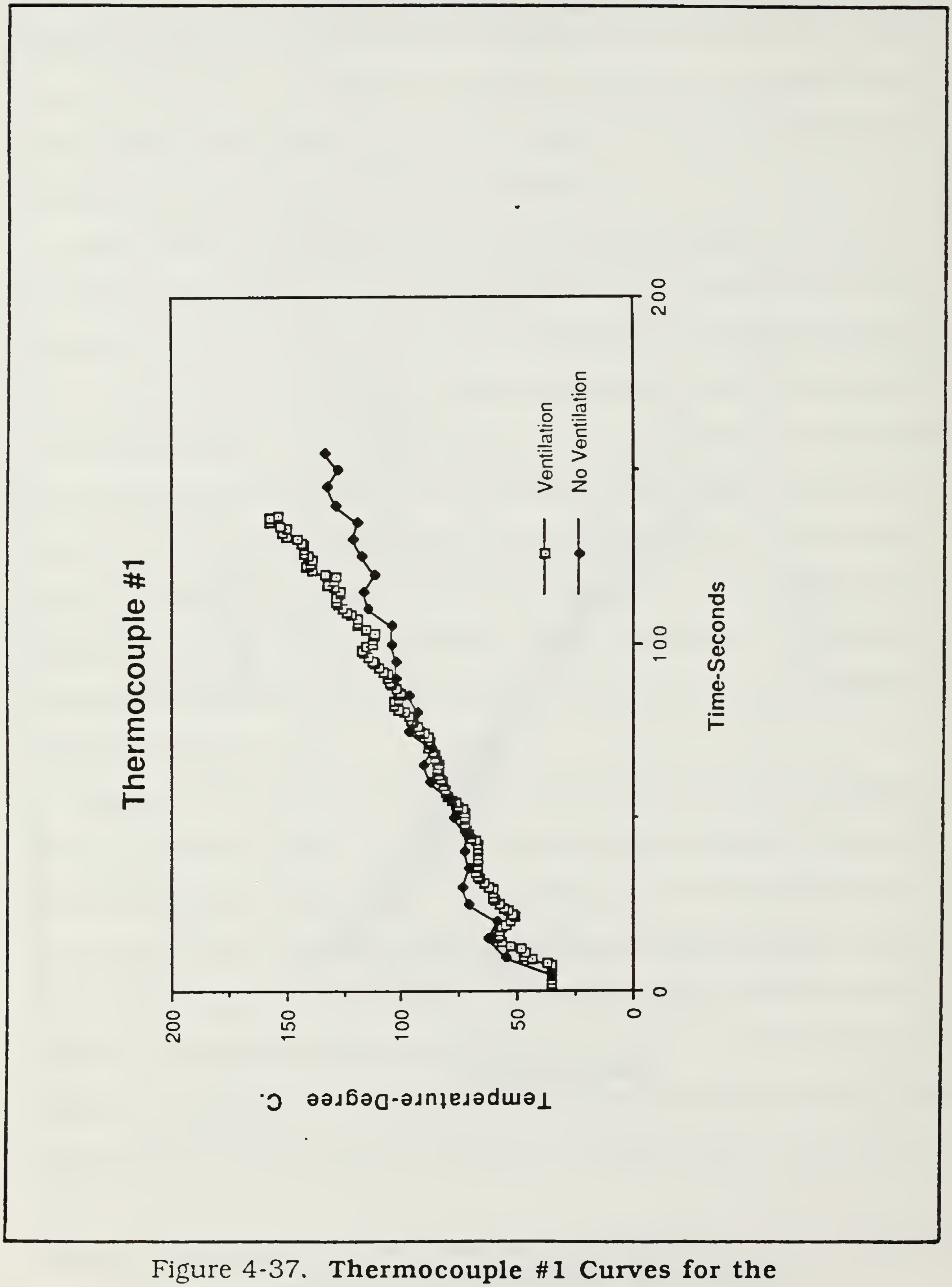

Figure 4-37. **Thermocouple #1 Curves for the Ventilated and Nonventilated Cases**

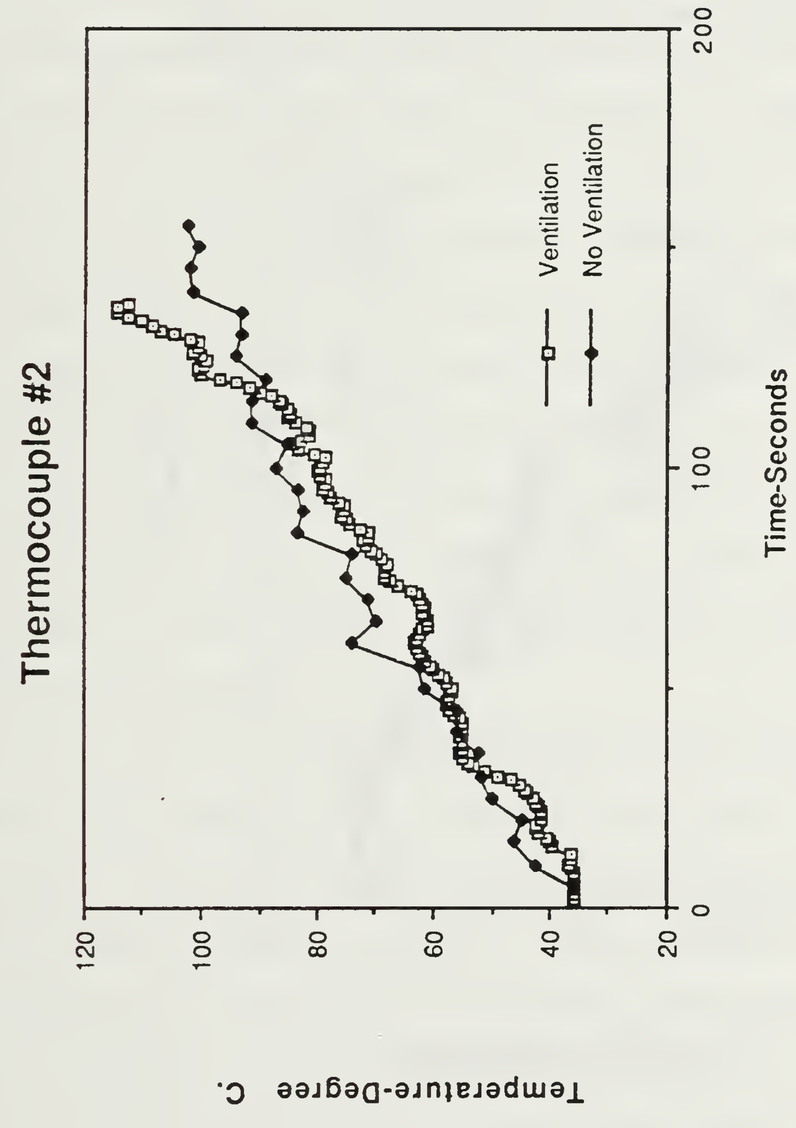

Figure 4-38. **Thermocouple #2 Curves for the Ventilated and Nonventilated Cases**

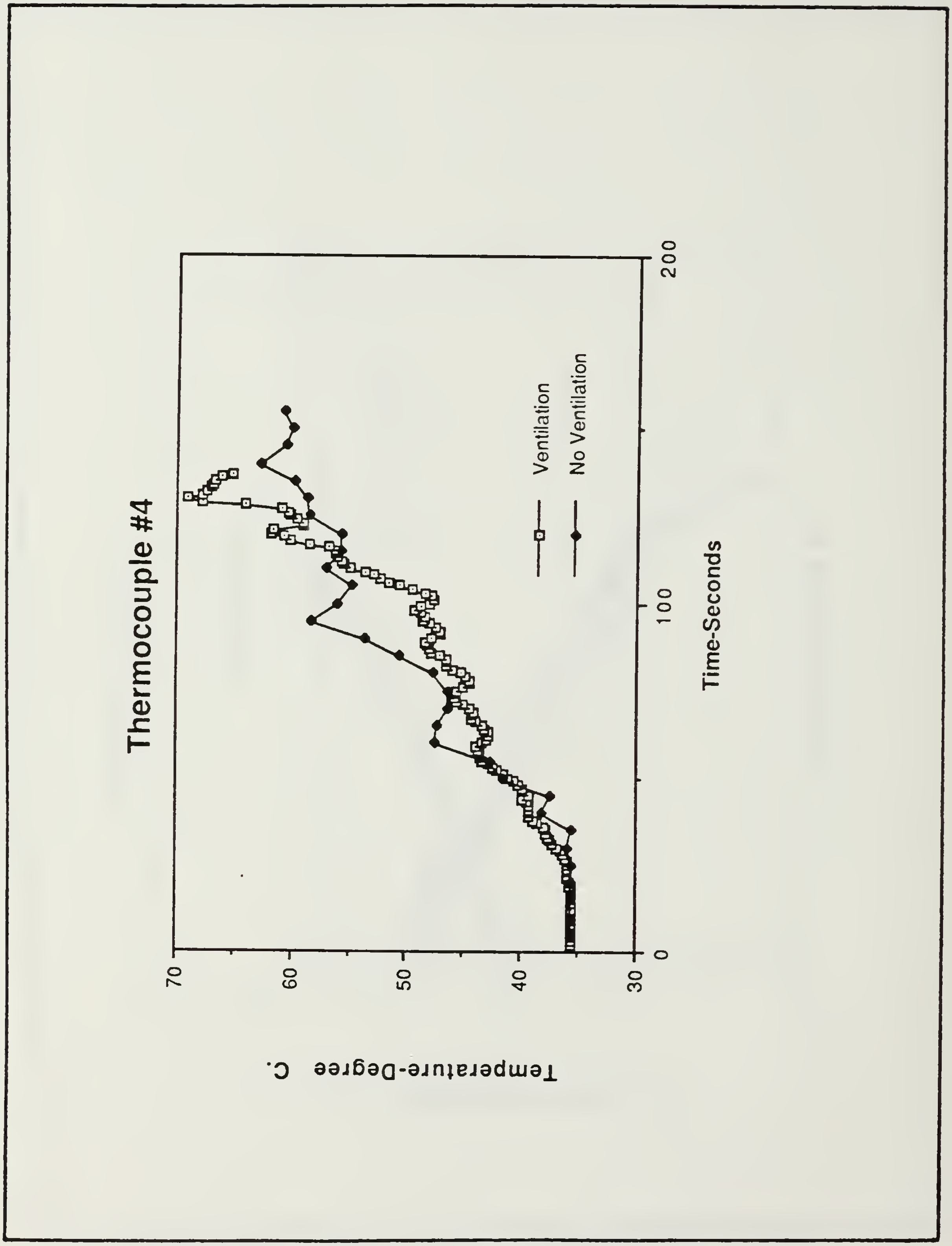

Figure 4-39. **Thermocouple #4 Curves for the Ventilated and Nonventilated Cases**

the fire is still in its first stages, and the entire field is rapidly changing. This dynamic situation, along with the approximations inherent in modeling, can also account for differences in the ventilated and nonventilated fields.

Figures 4.37 through 4.39 show the thermocouple temperatures versus time; the results are similar to the pressure, with the ventilated case increasing more slowly but then catching up to the nonventilated case, exceeding it at around 80 to 110 seconds. Since the thermocouples are in the north end cap, they are in the area in which the isotherms are pushed upward by the fan. This could explain why the temperatures are lower in the ventilated case. The temperatures exhibit some local fluctuations which could be the result of thermal instability associated with thermal plumes [Ref. 37]. In Figure 4.39 it appears that there are large oscillations, but the scale on the graph is smaller so that the temperature oscillation of all three thermocouples is in the same range. These oscillations appear in both the ventilated and nonventilated cases.

In most numerical models, the time step is an important factor. A small time step uses too much computer time, while too large a time step results in instability of the model. In this study, two trials were conducted with different time steps. In the first trial, a time step of 0.0288 seconds was used up to 40 seconds of fire time, and then the step was reduced to 0.0192. In the second trial, the beginning time step was 0.1152 seconds until 6 seconds of fire time, when the model

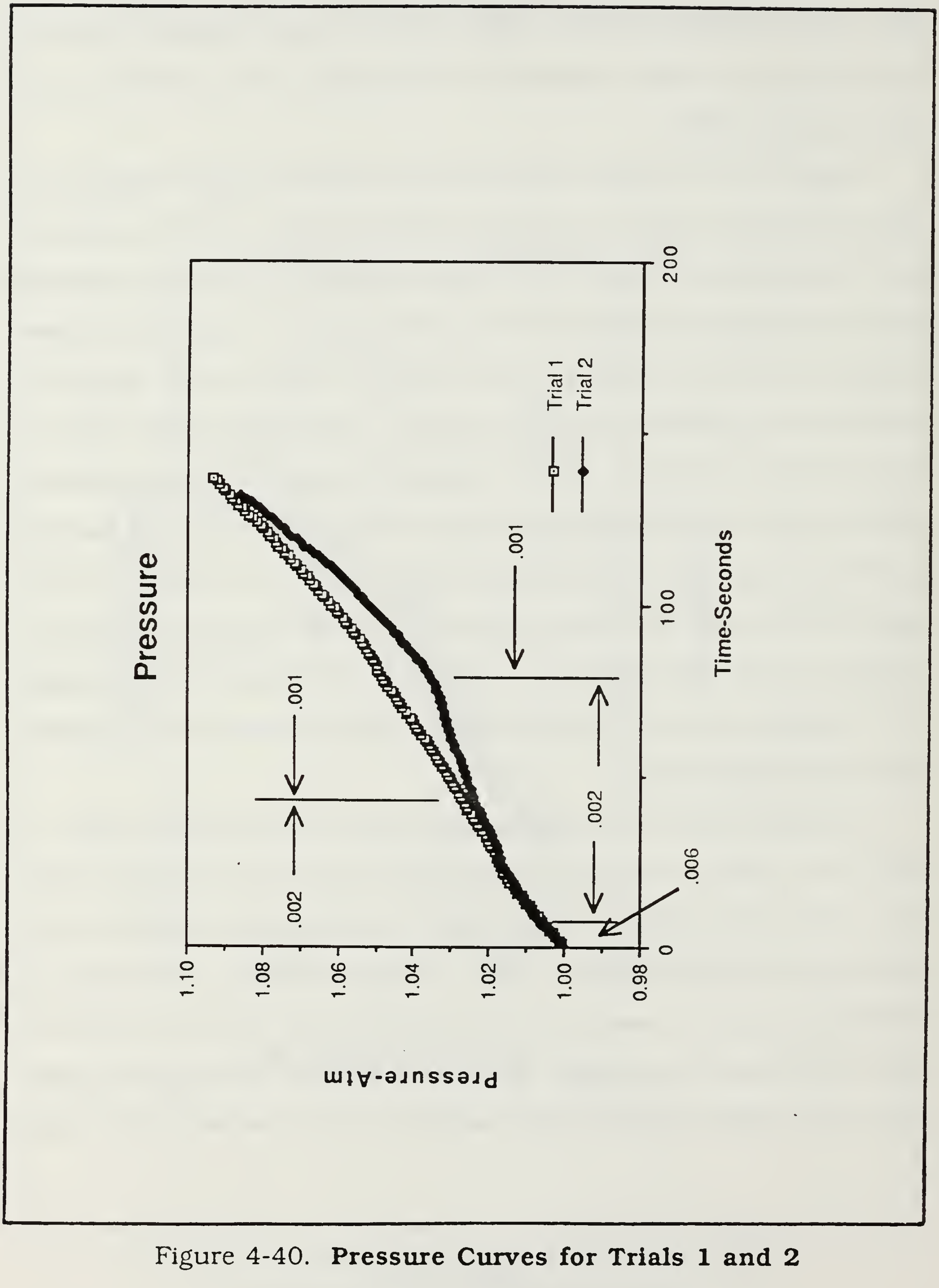

Figure 4-40. **Pressure Curves for Trials 1 and 2**

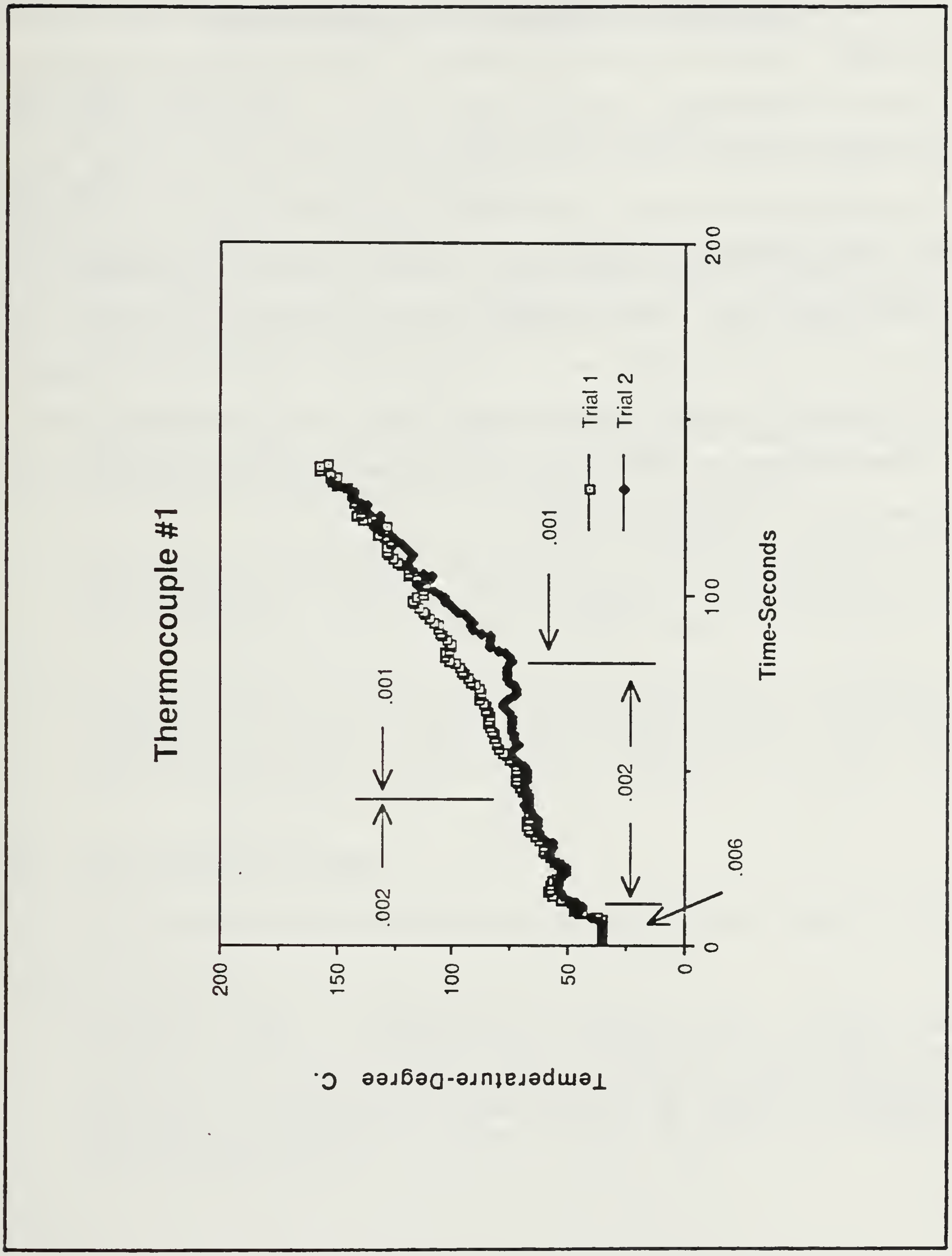

Figure 4-41. **Thermocouple #1 Curve for Trials 1 and 2**

became unstable. At that time, the time step was reduced to 0.0288 and further reduced to 0.0192 near 80 seconds, when it again became unstable. Figures 4.40 and 4.41 show the global pressure and temperature of thermocouple number 1 versus time for both trials. Note that the curves are coincident for the first 20 seconds, then diverge until approximately 90 seconds, when they begin to converge. At the end of the runs, both the pressure and temperature appear to become coincident once again. Since the only difference between these two runs was the time step difference, it is evident that time step does affect the transient results in this computer model. Also interesting is that it appears that solutions using different time steps would become the same after a long period of time.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

Several conclusions may be drawn from this simulation model of the FIRE-1 test facility with ventilation:

1. The ventilation model has been successfully incorporated into the numerical model of FIRE-1. The local velocity fields in the region of the fans exhibit a realistic behavior. The global effect of the fans is small due to the relatively low velocity and because the flow is parallel to the isotherms.

2. The global flow field exhibited appears realistic. The fire plume increases the gas velocity upward, resulting in a ceiling jet which is the dominant flow in the field. The flow recirculates within the field with minor variations caused by the ventilation.

3. The isotherms depict the concentration of hot gases in the top of the field. These hot gases stratify and slowly diffuse downward as time progresses. The isotherms are affected by the ventilation in the end cap region, where they are pushed upward or downward.

4. A small change in the time step makes a discernable difference in the transient solution. With different time steps, the transient solutions are different. When the time steps are the same, the previously diverging transient solutions appear to converge and become coincident.

B. RECOMMENDATIONS

The following recommendations are made for future work on the numerical model:

1. Additional FIRE-1 experiments are needed to better validate the numerical model. Accurate heat-release rate data must be obtained and included in the model, instead of using a synthesized rate. Additionally, sensors should be placed at different locations in the vessel to better validate the numerical results throughout the field.

2. Develop and incorporate additional models to simulate physical phenomena such as gaseous radiation and combustion.

3. Continue to expand and validate the model to include decks, equipment in the space, and fire-extinguishing systems.

4. Since the model uses an extensive amount of computer time, it is imperative that the numerical model be transferred to a super-computer or a dedicated mini-computer.

5. The ultimate goal of this project is to develop a computer model for predicting fire and smoke phenomena in shipboard situations. Completion of this goal will offer ship designers and engineers with a valuable tool to design and build safer ships and submarines.

APPENDIX

COMPUTER PROGRAM

```
C      ****************************************************************
C      **                                                            **
C      **         THREE-DIMENSIONAL NUMERICAL SIMULATION             **
C      **       OF A FIRE SPREAD INSIDE A NAVY STORAGE TANK          **
C      **                                                            **
C      **                     DEVELOPED BY :                         **
C      **               H.Q. YANG AND K.T. YANG                      **
C      **                                                            **
C      **    DEPARTMENT OF AEROSPACE & MECHANICAL ENGINEERING        **
C      **              UNIVERSITY OF NOTRE DAME                      **
C      **              NOTRE DAME, INDIANA, 46556                    **
C      **                                                            **
C      **                     DEC. 1986                              **
C      **                                                            **
C      ****************************************************************
C
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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/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,COND0,VIS0,RHO0,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)
      COMMON/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12)
      COMMON/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR
      DIMENSION VFMXC(579,579),T4WALL(579)
      DATA N,ITLEFT,SORMAX,XTIME,ITMAX/20,400000,0.40,0.0,4/
```

```
C *** U0     : REFERENCE VELOCITY (FT/SEC),1 FT/SEC
C *** RHOO   : REFERENCE AIR DENSITY (LBM/FT**3)
C *** H      : REFERENCE LENGTH (FT)
C *** TA     : REFERENCE TEMPERATURE (R)
C *** TINIT  : INITIAL TEMPERATURE (0)
C *** GC     : GRAVITATIONAL CONSTANT
C *** RAITR  : GAS CONSTANT; 53.34
C *** CONST1 : RA*U0**2/GC
C *** CONST3 : INVERSE OF TA
C *** CONST4 : REFERENCE LENGTH (CM)
C *** CONST6 : REFERENCE VELOCITY (CM/S)
C *** CONSRA : TA**3/(RA*CP*U0*H*H)
C *** NTRWR  : NTREAL/NWRITE*NWRITE
C *** NTRWA  : NTREAL/NWALT*NWALT
C *** HCONV  : HEAT TRANSFER COEFFICIENT ON THE AMBIENT (BTU/H.FT**2K)
C
C
C
C *** RAD,H: RADIUS OF THE CYLINDRICAL AND SPHERICAL SECTIONS
C     CYL  : LENGTH OF THE CYLINDRICAL SECTION OF THE TANK
C *** NI   : TOTAL NUMBER CELLS IN  THETA-DIRECTION
C     NJ   :                        R-DIRECTION
C     NK   :                        Z AND PHI-DIRECTIONS
C     NA   :           FIRST NUMBER  Z-DIRECTION, ALONG THE CYLINDER AXIS
C     NB   :           LAST  NUMBER  Z-DIRECTION, ALONG THE CYLINDER AXIS
C *** HSZ(1,1),HSZ(1,2)  FIRST AND LAST COORDIANTE OF HEAT SOURCE
C                             IN X-DIRECTION  (IN DIMENSIONLESS FORM)
C     HSZ(2,1),HSZ(2,2)  FIRST AND LAST COORDIANTE OF HEAT SOURCE
C                             IN Y-DIRECTION  (IN DIMENSIONLESS FORM)
C     HSZ(3,1),HSZ(3,2)  FIRST AND LAST COORDIANTE OF HEAT SOURCE
C                             IN Z-DIRECTION  (IN DIMENSIONLESS FORM)
C
C *** ICHPB( ) : STARTING NODAL NUMBER FOR SOLID IN THETA-DIRECTION
C     JCHPB( ) :                                        R-DIRECTION
C     KCHPB( ) :                                 Z OR PHI-DIRECTION
C *** NCHPI( ) : NUMBER OF NODALS  FOR SOLID IN THETA-DIRECTION
C     NCHPJ( ) :                                    R-DIRECTION
C     NCHPK( ) :                                Z,PHI-DIRECTION
C ***************************************************************

C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C        INPUT DATA                                             &
C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
      CALL INPUT

C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C        GENERATE GRID SYSTEM                                   &
C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
      CALL GRID

C ###############################################################
C ***   READ VIEW FACTOR INVERSE MATRIX                         #
C################################################################
  999 READ (11,END=998) VFMXC
      GOTO 999
```

```
  993 CONTINUE
      REWIND 11
      CLOSE (11)

C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C        INITIALIZE THE WHOLE FIELD                                &
C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
      CALL INIT

C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C         START CALCULATION                                        &
C &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

      NT=0
      NTIM=0

  300 CONTINUE

      NT=NT+1

C ***    NTMAXO HAS THE MEANING AS "NTREAL" WHEN IT IS READ FROM
C        DISK OR TAPE.

      IF(XTIME .GT. TMAX) GO TO 303
      NTREAL=NT+NTMAXO
      TIME=TIME+DTIME
      XTIME=TIME*H/U0


C&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C     CALCULATE THE TRANSIENT HEAT INPUT                           &
C     NOTE IF 1 IN PARENTHESIS, THE BURN RATE IS CALCULATED        &
C     BY THE PRESSURE CURVE.  IF EQUAL TO TWO, THE BURN RATE       &
C     CURVE IS EITHER GIVEN OR ESTIMATED                           &
C&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
        CALL CALQ(2)

C ***    START CALCULATION

      ITER=0
      JTERM=0
      JJTERM=0

C     DEFINE THE UPDATED TPD(I,J,K), CPD(I,J,K),RPD(I,J,K)
C     UPD(I,J,K) AND VPD(I,J,K) FOR THE USE OF CALVIS AND SU(I,J,K)

      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)
      UPD(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


C&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C    CALCULATE THE RADIATION HEAT FLUX AT EVERY NRAD TIME STEPS &
C&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

      NRAD = 2
      IF (MOD(NT,NRAD).NE.0) GOTO 4000
      CALL RADHT(T4WALL,VFMXC)
 4000 CONTINUE

C#################################################################
C     CALCULATE THE TEMPERATURE                                  #
C#################################################################
      CALL CALT

C&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C     CALCULATE THE SMOKE CONCENTRATION                          &
C&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
C     CALL CALC

      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%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
       CALL GLOBE

C@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     CALCULATE THE TURBULENT VISCOSITY AND CONDUCTIVITY         @
C@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      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
```

```
      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@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

      CALL CALU
CC    CALL STRESS
C *** **************
      CALL CALV
CC    CALL STRESS
C *** **************
      CALL CALW
CC    CALL STRESS
C *** **************

C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C     CALCULATE THE PRESSURE AND STRESS                            &
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

      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
      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(JTERM .EQ. 2) GO TO 35
      IF(ITER .EQ. 4) GO TO 29
   35 CONTINUE
      IF(JTERM .EQ. 3) GO TO 58
      IF(ITER .EQ. 7) GO TO 29
   58 CONTINUE
      JJTERM=0
      GO TO 301
  304 CONTINUE
      JJTERM=JJTERM+1
      IF(JJTERM .EQ. 1) WRITE(6,95) ITER,RESORM(ITER),SORSUM
      IF(JTERM .EQ. 1) GO TO 41
      IF(JTERM .EQ. 2 .AND. JJTERM .EQ. 1 .AND. ITER .NE. 5) GO TO 41
      GO TO 82
   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)
   43 CONTINUE
      IF(ITER .EQ. ITMAX) GO TO 49
      IF((JTERM .EQ. 3 .AND. ITER .NE. 8) .OR. JJTERM .EQ. 2) GO TO 49
      GO TO 301
   49 CONTINUE

      ITERT=ITERT+ITER
C#################################################################
C     GO TO THE PRESSURE TRACKING SUBROUTINE ,PRINT OUT          #
```

```
C     RESULTS IF AT THE RIGHT TIME INTERVAL                              #
C#####################################################################

      CALL PTRACK
      IF (MOD(NTREAL,NWRP).EQ.0) CALL OUT(1)

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C      FIND TEMPERATURES AT THERMOCOUPLE POINTS AND PRINT OUT    %
C      IF AT THE RIGHT TIME INTERVAL                             %
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

      CALL TCP
      IF (MOD(NTREAL,NWRP).EQ.0) CALL OUT(2)
 2422 CONTINUE
      IF (MOD(NTREAL,NWRITE).EQ.0) CALL OUT(3)
C     IF(NTREAL .EQ. NTREAL/NWRITE*NWRITE) CALL OUT(3)
  505 CONTINUE
      IF((XTIME+DTIME*H/U0) .GE. TMAX) GO TO 277

C *** ***********************************************************
C     CALL TLEFT(IT)
C 123 FORMAT(' ITLEFT = ',I10)
C     ITO=IT
C     IF(IT.LT.ITLEFT) CALL OUT(3)
C *** ***********************************************************


C ***    RESET THE OLD TIME VALUES  TOD, ROD, UOD, VOD AND POD.

      DO 305 K=1,NKP1
      DO 305 J=1,NJP1
      DO 305 I=1,NIP1
      TOD(I,J,K)=T(I,J,K)
      COD(I,J,K)=C(I,J,K)
      ROD(I,J,K)=R(I,J,K)
      UOD(I,J,K)=U(I,J,K)
      VOD(I,J,K)=V(I,J,K)
      WOD(I,J,K)=W(I,J,K)
      POD(I,J,K)=P(I,J,K)
  305 CONTINUE

C |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
C     THIS WRITING IS FOR PLOTTINGS                                  |
C|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
      IF(NTREAL .NE. NTREAL/NTAPE*NTAPE)GOTO 522
      IWRITE=10
      WRITE(IWRITE)
     & TIME,NTREAL,T,R,U,V,W,P,CPM,COND,VIS,QRNET,ITERT,QCORRT,PM1,PM2,
     & H,TA,U0,COND0,VIS0,RHO0,NI,NJ,NK,NIP1,NJP1,NKP1,NIM1,NJM1,NKM1,
     & XC,YC,ZC,XS,YS,ZS,DXXC,DYYC,DZZC,DXXS,DYYS,DZZS
      WRITE(6,*) 'THE TIME WHEN THE DATA WAS STORED ON TAPE IS:',
     & XTIME

C *** ***********************************************************
```

```
  522 CONTINUE

C *** ************************************************************
C     CALL TLEFT(IT)
C     IF(IT.LT.ITLEFT) GO TO 166
C *** ************************************************************
C TIMREM IS USED TO CALCULATE THE CPU TIME REMAINING AT NPS

      IF (TIMREM(0.).LE.80.) GOTO 166

      GO TO 300
  303 CONTINUE
  277 CONTINUE

      WRITE(6,1111)
 1111 FORMAT(2X,'****** THE MAXIMUM TIME HAS BEEN REACHED ******',I8)
      GO TO 172

C *** ***********************************************************
  166 IF(NTREAL .NE. NTREAL/NTAPE*NTAPE) WRITE(9)
     & TIME,NTREAL,T,R,U,V,W,P,CPM,COND,VIS,QRNET,ITERT,QCORRT,PM1,PM2,
     & H,TA,U0,COND0,VIS0,RHO0,NI,NJ,NK,NIP1,NJP1,NKP1,NIM1,NJM1,NKM1,
     & XC,YC,ZC,XS,YS,ZS,DXXC,DYYC,DZZC,DXXS,DYYS,DZZS
      REWIND 9
C *** ************************************************************

      GOTO 172
 2020 CONTINUE
      WRITE (6,*) '  RESIDUAL MASS IS LARGER THAN 10.0, PROGRAM STOPS'
  172 CONTINUE
      STOP
      END


C ______________________________________________________________________
* **********************************************************************
      SUBROUTINE INPUT
* **********************************************************************
*     THIS SUBROUTINE SETS UP REQUIRED VALUES TO BEGIN THE PROGRAM.    *
*     VARIABLES ARE:                                                   *
*               KRUN      =      WHEN EQUAL TO ONE,READ FROM THE       *
*                                RESTART DISK, ELSE FROM THE JCL       *
*               NCHIP     =      NUMBER OF SOLID PIECES                *
*               NWRP      =      NUMBER OF TIME STEPS TO WRITE ON THE  *
*                                PAPER                                 *
*               NTHCO     =      NUMBER OF THERMOCOUPLES TO PRINT OUT  *
*               TMAX      =      MAXIMUM TIME ALLOWED (REAL)           *
*               TWRITE    =      SECONDS IN REAL TIME TO PRINT THE     *
*                                P,V,T FIELDS ON PAPER                 *
*               TTAPE     =      TIME INTERVAL TO WRITE ON THE TAPE    *
*               DTIME     =      TIME STEP (DIMENSIONLESS)             *
*               HSZ       =      HEAT SOURCE SIZE, USED TO CALCULATE   *
*                                THE VOLUME OF THE FIRE CELL           *
*               ICHPB     =      FIRST SOLID NODE IN THETA DIRECTION   *
```

```
*              JCHPB     =      FIRST SOLID NODE IN R DIRECTION       *
*              KCHPB     =      FIRST SOLID NODE IN PHI DIRECTION     *
*              NCHPI     =      NUMBER OF NODES IN THETA DIRECTION    *
*              NCHPJ     =      NUMBER OF NODES IN R DIRECTION        *
*              NCHPK     =      NUMBER OF NODES IN PHI DIRECTION      *
*              CX,CY,CZ  =      THERMOCOUPLE POSITIONS IN THETA,R,PHI *
***********************************************************************

      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &         DXXC(93),DYYC(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/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,COND0,VIS0,RHO0,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)
      COMMON/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12)


C #1. READ IN DATA TO INDICATE EITHER KRUN=0 OR 1
      READ(5,*) KRUN,NCHIP,NWRP,NTHCO

C #2. READ IN DATA SET 1 - 6 DATA
      READ(5,*) TMAX,TWRITE,TTAPE,DTIME

C #3. READ IN DATA FOR HEAT SOURCE

      READ (5,*)  HSZ(1,1),HSZ(1,2),HSZ(2,1),HSZ(2,2),HSZ(3,1),HSZ(3,2)
      WRITE(6,20) HSZ(1,1),HSZ(1,2),HSZ(2,1),HSZ(2,2),HSZ(3,1),HSZ(3,2)
   20 FORMAT (/,20X,'HEAT SOURCE LOCATION IS IN THE VOLUME (NON-DIME',
     &   'NSIONAL WITH RESPECT TO RADIUS)',
     &     /,5X,'FROM  ',F8.4,'  TO  ',F8.4,'  IN X-DIRECTION',
     &     /,5X,'FROM  ',F8.4,'  TO  ',F8.4,'  IN Y-DIRECTION',
     &     /,5X,'FROM  ',F8.4,'  TO  ',F8.4,'  IN Z-DIRECTION',/)
```

```
C #4. READ IN DECK DATA

      IF (NCHIP.EQ.0) GOTO 16
      PRINT *
      PRINT *,'    THE REGION BOUNDED BY SOLID'
      DO 19 N=1,NCHIP
      READ (5,*) 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


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 (5,*) CX(I),CY(I),CZ(I)
      WRITE (6,*) I, CX(I),CY(I),CZ(I)
  110 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                       *
*          U0          =      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      *00054700
*          RHOO         =      REFERENCE DENSITY                       *00054800
*          GC           =      GRAVITY CONSTANT                        *00054900
*          BUOY         =      BUOYANCY FORCE CONSTANT                 *00055000
*          UGRT         =      PERFECT GAS LAW NONDIMENSIONAL CONSTANT*00055100
*          CPO          =      REFERENCE SPECIFIC HEAT                 *00055200
*          NWRITE/      =      NONDIMENSIONAL FORMS OF TWRITE AND      *00055300
*          NTAPE               TTAPE                                   *00055400
*     MATRICES OF THE FORM                                             *00055500
*          _OD          =      DIMENSIONLESS PARAMETER AT OLD TIME     *00055600
*          _            =      DIMENSIONLESS PARAMETER                 *00055700
*          _PD          =      UPDATED DIMENSIONLESS PARAMETER         *00055800
*     WHERE THE PARAMETERS ARE                                         *00055900
*          U,V,W        =      VELOCITY IN THETA, R , PHI DIRECTION    *00056000
*          T,P,C        =      TEMP, PRESSURE, AND SMOKE CONCENTRATION*00056100
*                                                                      *00056200
*          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           =      COEFICIENT AT NODAL POINT               *00056800
*          AE,AW,AN     =      COEFICIENTS 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
*          _B,_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,DYYC    =      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,DYYS    =      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),              00059400
     &          DXXC(93),DYYC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)  00059500
      COMMON/BL1/DX,DY,DZ,VOL,DTIME,VOLDT,THOT,TCOOL,PI,Q,QR            00059600
      COMMON/BL7/NI,NIP1,NIM1,NJ,NJP1,NJM1,NK,NKP1,NKM1                 00059700
     &  ,NIP2,NJP2,NKP2,NA,NAP1,NAM1,NB,NBP1,NBM1,KRUN,NCHIP,NJRA,NWRP  00059800
      COMMON/BL12/ NWRITE,NTAPE,NTMAXO,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,UO,H,UGRT,BUOY,00060100
```

```
     &  CP0,PRT,COND0,VIS0,RHO0,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/ T00(22,16,32),ROD(22,16,32),POD(22,16,32)
     &       ,C00(22,16,32),UOD(22,16,32),VOD(22,16,32),WOO(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/ TP0(22,16,32),RPD(22,16,32),PPD(22,16,32)
     &       ,CP0(22,16,32),UP0(22,16,32),VP0(22,16,32),WP0(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)
      COMMON/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12)
      COMMON/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR
      DATA GRAV/32.17/

C ***    INTROOUCE GIVEN PARAMETERS

      TIME=0.
      TR=TA/1.8
      H=9.6
      VIS0=VIS0/U0/H
      VISL=VIS0
      VISMAX=400.*VISL
      HR=H*30.48
      COND0=VIS0/PRT
      PI=4.*ATAN(1.)
      ALEW = 1.0
      NJRA=15

C  THE HEAT TRANSFER COEFFICIENT IS IN BTU/HR/FT**2/F
      HCONV=15.0
      HCOEF=HCONV/(3600.*CP0*RHO0*U0)
      CC = 0.0


      CONST1=RHO0*U0*U0/(GC*14.696*144.)
      CONST3=1.8/TA
      CONST4=H*30.48
      CONST6=U0*30.48
      NTMAX0=0

      BUOY=GRAV*H/(U0*U0)
      UGRT=U0*U0/(GC*RAIR*TA)
      TCOOL=1.0
      CONSRA=TA*TA*TA/(RHO0*CP0*U0*3600.)*1.714E-9

      WRITE(6,200) TR,COND0,VIS0,CP0,HR,DTIME,HCONV
```

```
  200 FORMAT(5X, 'THE REFRENCE TEMPERATURE AND THERMAL PROPERTIES',/,
     &    /,5X, 'T    = ',F10.4,'K,     CONDO = ',E12.6,
     &    /,5X,'VISO = ',E12.6,'    CPO   = ',E12.6,
     &    /,5X,'RADIUS = ',E12.6,'   CM',
     &    /,5X,'DTIME = ',E12.6,
     &    /,5X,'HCONV = ',E12.6,/)

      NWRITE=TWRITE*U0/DTIME/H
      NTAPE=TTAPE*U0/DTIME/H
C ***    PRINT OUT INPUT INFORMATION

      WRITE(6,61) (STAR,I=1,90),KRUN,TMAX,TWRITE,TTAPE,NWRP
   61 FORMAT(///,90A1,/,5X,'KRUN  =',I2,/,5X,
     & 'TMAX =',F8.3,' SECONDS',/5X,'TWRITE =',F8.3,
     & ' SECONDS',/,5X,'TTAPE  =',F8.3,' SECONDS',
     & /,5X,' NUMBER INTERVALS OF WRITING ON PAPER  ', I5,/)

C ***    INITIALIZE VARIABLE FIELD

      DO 220 J=1,NJP1
      DO 220 I=1,NIP1
      DO 220 K=1,NKP1
      ROD(I,J,K)=1.
      R(I,J,K)=1.
      RPD(I,J,K)=1.
      UOD(I,J,K)=0.
      U(I,J,K)=0.
      UPD(I,J,K)=0.
      VOD(I,J,K)=0.
      V(I,J,K)=0.
      VPD(I,J,K)=0.
      W(I,J,K)=0.
      WPD(I,J,K)=0.
      WOD(I,J,K)=0.
      POD(I,J,K)=0.
      P(I,J,K)=0.
      PPD(I,J,K)=0.
      DU(I,J,K)=0.
      DV(I,J,K)=0.
      DW(I,J,K)=0.
      SU(I,J,K)=0.
      SP(I,J,K)=0.
      PP(I,J,K)=0.
      AP(I,J,K)=0.
      AW(I,J,K)=0.
      AE(I,J,K)=0.
      AN(I,J,K)=0.
      AS(I,J,K)=0.
      AF(I,J,K)=0.
      AB(I,J,K)=0.
      SMP(I,J,K)=0.
      SMPP(I,J,K)=0.
      VIS(I,J,K)=VISL
      COND(I,J,K)=CONDO
      CPM(I,J,K)=1.0E0
```

```
      TOO(I,J,K)=1.0E0
      T(I,J,K)=TOO(I,J,K)
      TPD(I,J,K)=TOD(I,J,K)
      COO(I,J,K)=CO
      C(I,J,K)=COO(I,J,K)
      CPO(I,J,K)=COD(I,J,K)
      NHSZ(I,J,K)=0
      NOD(I,J,K)=0
  220 CONTINUE



C ***   DETERMINE THE POSITION OF HEAT SOURCE

      DO 300 I=2,NI
      DO 300 J=2,NJ

C CHANGE TO RECTANGULAR COORDINATES
      XX=YC(J)*COS(XC(I))
      YY=YC(J)*SIN(XC(I))

C CHECK TO SEE IF IN HS CONTROL VOLUME, IF SO SET NHSZ=1
      IF (XX.LT.HSZ(1,1).OR.XX.GT.HSZ(1,2)) GOTO 310
      IF (YY.LT.HSZ(2,1).OR.YY.GT.HSZ(2,2)) GOTO 310
      NHSZ(I,J,16)=1
      NHSZ(I,J,17)=1
  315 FORMAT (2X,10(4X,I4,2X,I4))
      GOTO 300
  310 CONTINUE
  300 CONTINUE


C ***     DEFINE THERMAL PROPERTIES OF DECK AND SOLID

      IF (NCHIP.EQ.0) GOTO 410
      DO 402 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 405 I=IB,IE-1
      DO 405 J=JB,JE-1
      DO 405 K=KB,KE-1
      CONO(I,J,K)=CONOO*CONS(N)
      CPM(I,J,K)=CPO*CPS(N)
      NOO(I,J,K)=1
  405 CONTINUE
  402 CONTINUE
  410 CONTINUE



C *** FOR CONTINUING RUN, READ DATA FROM TAPE OR DISK
```

```
      IF(KRUN .EQ. 1) GO TO 9997
      GO TO 15
 9997 READ(8,END=9998)
     & TIME,NTMAXO,TOD,ROD,UOD,VOD,WOD,POD,CPM,COND,VIS,QRNET,ITERT,QCOR
     &RT,PM1,PM2,XX,XX,XX,XX,XX,XX,NI,NJ,NK,NIP1,NJP1,NKP1,NIM1,NJM1
     & ,NKM1,XC,YC,ZC,XS,YS,ZS,DXXC,DYYC,DZZC,DXXS,DYYS,DZZS
      GO TO 9997
 9998 CONTINUE
      REWIND 8
      CLOSE (8)
      WRITE(6,*)NTMAXO
   15 CONTINUE


C ***     DEFINE HEIGHT OF NODE POINTS AND COMPUTE HYDROSTATIC
C         EQUILIBRIUM DENSITY REQ(I,J,K)


      DO 13 K=1,NKP1
      DO 13 I=1,NIP1
      DO 13 J=1,NJP1
      DHY=YC(J)*SIN(XC(I))*SIN(ZC(K))
      HEIGHT(I,J,K)=DHY
 13   CONTINUE
C
      DO 229 J=1,NJP1
      DO 229 I=1,NIP1
      DO 229 K=1,NKP1
      AAAA=-BUOY*UGRT*HEIGHT(I,J,K)
      REQ(I,J,K)=EXP(AAAA)
    . IF(KRUN .NE. 0) GO TO 229
      RPD(I,J,K)=REQ(I,J,K)/TPD(I,J,K)
      ROD(I,J,K)=RPD(I,J,K)
      R(I,J,K)=RPD(I,J,K)
  229 CONTINUE

C ***     INITIALIZE U,V,T,R,P FIELD

      DO 210 K=1,NKP1
      DO 210 J=1,NJP1
      DO 210 I=1,NIP1
      T(I,J,K)=TOD(I,J,K)
      C(I,J,K)=COD(I,J,K)
      R(I,J,K)=ROD(I,J,K)
      U(I,J,K)=UOD(I,J,K)
      V(I,J,K)=VOD(I,J,K)
      W(I,J,K)=WOD(I,J,K)
      P(I,J,K)=POD(I,J,K)
  210 CONTINUE

C ***   FOLLOWING IS FOR DETERMINING THE THERMOCOUPLE POSITIONS

      DO 5000 N=1,NTHCO
      DO 5001 I=1,NIP1
```

```
      IF (XC(I).LT.CX(N).AND.XC(I+1).GE.CX(N)) GOTO 5002
 5001 CONTINUE
 5002 II=I

      DO 5003 J=1,NJP1
      IF (YC(J).LT.CY(N).AND.YC(J+1).GE.CY(N)) GOTO 5004
 5003 CONTINUE
 5004 JJ=J

      DO 5005 K=1,NKP1
      IF (ZC(K).LT.CZ(N).AND.ZC(K+1).GE.CZ(N)) GOTO 5006
 5005 CONTINUE
 5006 KK=K
      NTH(N,1)=II
      NTH(N,2)=JJ
      NTH(N,3)=KK
 5000 CONTINUE


      RETURN
      END

C     ____________________________________________________________
C *** ************************************************************
      SUBROUTINE CALVIS

C *** *************************************************************
*     THIS SUBROUTINE CALCULATES THE TURBULENT VISCOSITY AND UPDATES*
*     THE VISCOSITY MATRIX                                          *
*******************************************************************

      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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
      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,COND0,VIS0,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      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/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 LOCAL SHEAR AND VISCOSITY VIS(I,J,K)
C
C ***    SPECIFY LOCAL TURBULENT LENGTH SCALES  SMPP(I,J,K)

      DO 611 K=2,NK
```

```
      KP2=K+2
      KP1=K+1
      KM1=K-1
      KM2=K-2
      DO 611 J=2,NJ
      JP2=J+2
      JP1=J+1
      JM1=J-1
      JM2=J-2
      DO 611 I=2,NI
      IP2=I+2
      IP1=I+1
      IM1=I-1
      IM2=I-2
      IF (I.EQ.2)  IM2=NIM1
      IF (I.EQ.NI) IP2=3
      IF (NOD(I,J,K).EQ.1) GOTO 611

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)

CC    IF (J.EQ.2) DYS=DYS/2.
CC    IF (K.EQ.2) DZB=DZB/2.
      IF (J.NE.NJ) GOTO 101
      JP2=JP1
      DYN=DYN/2.
  101 IF (K.NE.NK) GOTO 102
      KP2=KP1
      DZF=DZF/2.
  102 CONTINUE

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

      DXE =XL(IP1,J,K,0,1)
      DXW =XL(I  ,J,K,0,1)

      DYN =YL(I,JP1,K,0,2)
      DYS =YL(I,J  ,K,0,2)

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

C ***    CACULATE DV/DX,D2V/DX2,DU/DX,D2U/DX2,DW/DX AND D2W/DX2
```

```
      DUDX=(U(IP1,J,K)-U(I,J,K))/DXI
      DUDXW=0.5*(U(IP1,J,K)-U(IM1,J,K))/DXW
      DUDXE=0.5*(U(IP2,J,K)-U(I  ,J,K))/DXE
      D2UDX2=(DUDXE-DUDXW)/DXI


      DVDXW=0.5*(V(I,JP1,K)+V(I,J,K)-V(IM1,JP1,K)-V(IM1,J,K))/DXW
      DVDXE=0.5*(V(IP1,JP1,K)+V(IP1,J,K)-V(I,JP1,K)-V(I,J,K))/DXE
      DVDX=0.5*(DVDXE+DVDXW)
      D2VDX2=  (DVDXE-DVDXW)/DXI


      DWDXW=0.5*(W(I,J,KP1)+W(I,J,K)-W(IM1,J,KP1)-W(IM1,J,K))/DXW
      DWDXE=0.5*(W(IP1,J,KP1)+W(IP1,J,K)-W(I,J,KP1)-W(I,J,K))/DXE
      DWDX=0.5*(DWDXE+DWDXW)
      D2WDX2=  (DWDXE-DWDXW)/DXI


  602 CONTINUE

C ***    CALCULATE DU/DY,D2U/DY2,DV/DY,D2V/DY2,DW/DY AND D2W/DY2


      DVDY=(V(I,JP1,K)-V(I,J,K))/DYJ
      DVDYS=0.5*(V(I,JP1,K)-V(I,JM1,K))/DYS
      DVDYN=0.5*(V(I,JP2,K)-V(I,J  ,K))/DYN
      D2VDY2=(DVDYN-DVDYS)/DYJ


      DUDYS=0.5*(U(IP1,J,K)+U(I,J,K)-U(IP1,JM1,K)-U(I,JM1,K))/DYS
      DUDYN=0.5*(U(IP1,JP1,K)+U(I,JP1,K)-U(IP1,J,K)-U(I,J,K))/DYN
      DUDY=0.5*(DUDYN+DUDYS)
      D2UDY2=  (DUDYN-DUDYS)/DYJ


      DWDYS=0.5*(W(I,J,KP1)+W(I,J,K)-W(I,JM1,KP1)-W(I,JM1,K))/DYS
      DWDYN=0.5*(W(I,JP1,KP1)+W(I,JP1,K)-W(I,J,KP1)-W(I,J,K))/DYN
      DWDY=0.5*(DWDYN+DWDYS)
      D2WDY2=  (DWDYN-DWDYS)/DYJ

  606 CONTINUE

C ***    CALCULATE DU/DZ,D2U/DZ2,DV/DZ,D2V/DZ2,DW/DZ AND D2W/DZ2


      DWDZ=(W(I,J,KP1)-W(I,J,K))/DZK
      DWDZF=0.5*(W(I,J,KP2)-W(I,J,K   ))/DZF
      DWDZB=0.5*(W(I,J,KP1)-W(I,J,KM1))/DZB
      D2WDZ2=(DWDZF-DWDZB)/DZK


      DVDZB=0.5*(V(I,JP1,K)+V(I,J,K)-V(I,JP1,KM1)-V(I,J,KM1))/DZB
      DVDZF=0.5*(V(I,JP1,KP1)+V(I,J,KP1)-V(I,JP1,K)-V(I,J,K))/DZF
      DVDZ=0.5*(DVDZF+DVDZB)
```

```
      D2VDZ2=  (DVDZF-DVDZB)/DZK


      DUDZB=0.5*(U(IP1,J,K)+U(I,J,K)-U(IP1,J,KM1)-W(I,J,KM1))/DZB
      DUDZF=0.5*(U(IP1,J,KP1)+U(I,J,KP1)-U(IP1,J,K)-U(I,J,K))/DZF
      DUDZ=0.5*(DUDZF+DUDZB)
      D2UDZ2=  (DUDZF-DUDZB)/DZK

      DRDX=((R(IP1,J,K)-REQ(IP1,J,K))-(R(IM1,J,K)-REQ(IM1,J,K)))/
     &    (DXE+DXW)
      DRDY=((R(I,JP1,K)-REQ(I,JP1,K))-(R(I,JM1,K)-REQ(I,JM1,K)))/
     &    (DYN+DYS)
      DRDZ=((R(I,J,KP1)-REQ(I,J,KP1))-(R(I,J,KM1)-REQ(I,J,KM1)))/
     &    (DZF+DZB)
      DRDGA=SIN(ZC(K))*(SIN(XC(I))*DRDY+COS(XC(I))*DRDX)
     &     +COS(ZC(K))*DRDZ

C ***    CALCULATE RICHARDSON NUMBER

      STRAIN=DUDY**2+DVDX**2+DWDX**2+DVDZ**2+DWDY**2+DUDZ**2
      DDO2 = SQRT(DUDY*DUDY+DUDX*DUDX+DUDZ*DUDZ+DVDY*DVDY+DVDX*DVDX+
     & DVDZ*DVDZ+DWDX*DWDX+DWDY*DWDY+DWDZ*DWDZ)
      IF(DDO2.EQ.0.)GO TO 600

C ***    CALCULATE TURBULENT LENGTH SCALE SMPP(I,J)

      SMP123=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
      SMPP12=DDO2 /SQRT(D2UDX2*D2UDX2+D2UDY2*D2UDY2
     & +D2UDZ2*D2UDZ2+D2VDX2*D2VDX2+D2VDY2*D2VDY2+D2VDZ2*D2VDZ2+
     & D2WDZ2*D2WDZ2+D2WDX2*D2WDX2+D2WDY2*D2WDY2)
      SMPP(I,J,K)=CNT*(SMP123+SMPP12)*.5
      RI(I,J,K)=-BUOY*DRDGA/(R(I,J,K)*STRAIN)
      ABRIPR=ABTURB+RI(I,J,K)/PRT
      IF(ABRIPR .LT. 0.) GO TO 600
      IF(ABRIPR .EQ. 0.) GO TO 613
      GO TO 610
  600 VIS(I,J,K)=VISL
      GO TO 611
  613 VIS(I,J,K)=VISMAX
      GO TO 611
  610 VIS(I,J,K)=VISL+R(I,J,K)*SMPP(I,J,K)*SMPP(I,J,K)*SQRT(STRAIN)/
     &         (BTURB*ABRIPR)
      IF(VIS(I,J,K) .GT. VISMAX) VIS(I,J,K)=VISMAX
  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 (NOD(I,J,K).EQ.1) GOTO 140
      COND(I,J,K)=VIS(I,J,K)/PRT
  140 CONTINUE

      RETURN
      END



C ____________________________________________________________________
C *** ****************************************************************
      SUBROUTINE CALT
C *** ****************************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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/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,COND0,VIS0,RHO0,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),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=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.2)  IM2=NIM1
      IF (I.EQ.NI) IP2=3

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

      DXEE=XL(IP2,J,K,0,1)
      DXE =XL(IP1,J,K,0,1)
      DXW =XL(I  ,J,K,0,1)
      DXWW=XL(IM1,J,K,0,1)

      DYNN=YL(I,JP2,K,0,2)
      DYN =YL(I,JP1,K,0,2)
      DYS =YL(I,J  ,K,0,2)
      DYSS=YL(I,JM1,K,0,2)

      DZFF=ZL(I,J,KP2,0,3)
      DZF =ZL(I,J,KP1,0,3)
      DZB =ZL(I,J,K  ,0,3)
      DZBB=ZL(I,J,KM1,0,3)

C ***   DEFINE THE 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

      ZXOYN=DZXN/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)*DZXN
      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))

      CONDN1=ZXOYN*CONDN
      CONDS1=ZXOYS*CONDS
      CONDE1=YZOXE*CONDE
      CONDW1=YZOXW*CONDW
      CONDF1=XYOZF*CONDF
      CONDB1=XYOZB*CONDB


      CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW  ))/8.
      CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/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+DYNN))/8.
      CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/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
      AW(I,J,K)= .5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW
      AN(I,J,K)=-.5*DYJ/DYN*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN
      AS(I,J,K)= .5*DYJ/DYS*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS
      AF(I,J,K)=-.5*DZK/DZF*CF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF
      AB(I,J,K)= .5*DZK/DZB*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB
C
  801 AEE=-CEM*DXE/DXEE
      AEER=AEE*TPD(IP2,J,K)*CPM(IP2,J,K)
  802 CONTINUE

  803 AWW=-CWP*DXW/DXWW
      AWWR=AWW*TPD(IM2,J,K)*CPM(IM2,J,K)
  804 CONTINUE

      IF (J.LT.NJ) GOTO 805
      ANN=0.
      ANNR=0.
      GOTO 806
  805 ANN=-CNM*DYN/DYNN
      ANNR=ANN*TPD(I,JP2,K)*CPM(I,JP2,K)
  806 CONTINUE

      IF (J.GT.2) GOTO 807
      ASS=0.
      ASSR=0.
      GOTO 808
  807 ASS=-CSP*DYS/DYSS
```

```
      ASSR=ASS*TPD(I,JM2,K)*CPM(I,JM2,K)
  808 CONTINUE

      IF (K.LT.NK) GOTO 809
      AFF=0.
      AFFR=0.
      GOTO 810
  809 AFF=-CFM*DZF/DZFF
      AFFR=AFF*TPD(I,J,KP2)*CPM(I,J,KP2)
  810 CONTINUE

      IF (K.GT.2) GOTO 811
      ABB=0.
      ABBR=0.
      GOTO 812
  811 ABB=-CBP*DZB/DZBB
      ABBR=ABB*TPD(I,J,KM2)*CPM(I,J,KM2)
  812 CONTINUE



C ###################################################
C ###################################################
C *** MODIFICATION FOR DECK     BOUNDARIES

  900 CONTINUE
      IF (NOD(IM1,J,K).EQ.0) GOTO 901
      AWW=0.0
      AWWR=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,KM1).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 ###########################################################

      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)*CPM(I,J,K)
     &          +CONDE1+CONDW1+CONDN1+CONDS1+CONDF1+CONDB1

      AE(I,J,K)=AE(I,J,K)*CPM(IP1,J,K)+CONDE1
      AW(I,J,K)=AW(I,J,K)*CPM(IM1,J,K)+CONDW1
      AN(I,J,K)=AN(I,J,K)*CPM(I,JP1,K)+CONDN1
      AS(I,J,K)=AS(I,J,K)*CPM(I,JM1,K)+CONDS1
      AF(I,J,K)=AF(I,J,K)*CPM(I,J,KP1)+CONDF1
      AB(I,J,K)=AB(I,J,K)*CPM(I,J,KM1)+CONDB1

      SP(I,J,K)=-ROD(I,J,K)*VOLDT*CPM(I,J,K)
      SU(I,J,K)= ROD(I,J,K)*VOLDT*TOD(I,J,K)*CPM(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 ***    RADIUS DIRECTION

      DO 500 I=2,NI
      DO 500 K=2,NK
      SP(I,2,K)=SP(I,2,K)+AS(I,2,K)
CC    SP(I,2,K)=SP(I,2,K)-AS(I,2,K)
CC    SU(I,2,K)=SU(I,2,K)+2.0*AS(I,2,K)*TPD(I,1,K)
      SP(I,NJ,K)=SP(I,NJ,K)-AN(I,NJ,K)
      SU(I,NJ,K)=SU(I,NJ,K)+2.*TPD(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)*T(1   ,J,K)
      SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*T(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
      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/(U0*CP0*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
  310 CONTINUE

C ***   END OF RADIATION

C ***   SOLVE FOR T

      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
   83 CONTINUE
   81 CONTINUE
C
      DO 74 I=1,NIP1
      DO 74 J=1,NJP1
      T(I,J,1)=T(I,J,2)
      T(I,J,NKP1)=T(I,J,NK)
   74 CONTINUE

C ***    FOR SURFACE HEAT EXCHANGE WITH SURROUNDING

      DO 84 I=2,NI
      DO 84 K=2,NK
      DYJ=YL(I,NJ,K,0,0)
      T(I,NJP1,K)=(2.0*COND(I,NJ,K)*T(I,NJ,K)/DYJ+HCOEF*TINF)/
     &            (HCOEF+2.0*COND(I,NJ,K)/DYJ)
  84  CONTINUE


C ***        FOR CYLIC CONDITION

      DO 80 J=1,NJP1
      DO 80 K=1,NKP1
      T(1,J,K)=T(NI,J,K)
      T(NIP1,J,K)=T(2,J,K)
  80  CONTINUE

      RETURN
      END



C _______________________________________________________________
C *** ***********************************************************
      SUBROUTINE CALC
C *** ***********************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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,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,COND0,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/ TOO(22,16,32),ROO(22,16,32),POO(22,16,32)
     &  ,COO(22,16,32),UOO(22,16,32),VOO(22,16,32),WOO(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)
     &       ,CPO(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),NOO(22,16,32),RWALL(579)
     &       ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)
      COMMON/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR

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.2)  IM2=NIM1
      IF (I.EQ.NI) IP2=3

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

      DXEE=XL(IP2,J,K,0,1)
      DXE =XL(IP1,J,K,0,1)
      DXW =XL(I  ,J,K,0,1)
      DXWW=XL(IM1,J,K,0,1)

      DYNN=YL(I,JP2,K,0,2)
      DYN =YL(I,JP1,K,0,2)
      DYS =YL(I,J  ,K,0,2)
      DYSS=YL(I,JM1,K,0,2)

      DZFF=ZL(I,J,KP2,0,3)
      DZF =ZL(I,J,KP1,0,3)
      DZB =ZL(I,J,K  ,0,3)
      DZBB=ZL(I,J,KM1,0,3)

C ***   DEFINE THE 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

      ZXOYN=DZXN/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)*DZXN
      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))

      CONDN1=ZXOYN*CONDN*ALEW
      CONDS1=ZXOYS*CONDS*ALEW
      CONDE1=YZOXE*CONDE*ALEW
      CONDW1=YZOXW*CONDW*ALEW
      CONDF1=XYOZF*CONDF*ALEW
      CONDB1=XYOZB*CONDB*ALEW


      CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW  ))/8.
      CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/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+DYNN))/8.
      CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/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
      AW(I,J,K)= .5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW
      AN(I,J,K)=-.5*DYJ/DYN*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN
      AS(I,J,K)= .5*DYJ/DYS*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS
      AF(I,J,K)=-.5*DZK/DZF*CF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF
      AB(I,J,K)= .5*DZK/DZB*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB
                .

  801 AEE=-CEM*DXE/DXEE
      AEER=AEE*CPD(IP2,J,K)
  802 CONTINUE

  803 AWW=-CWP*DXW/DXWW
      AWWR=AWW*CPD(IM2,J,K)
  804 CONTINUE
```

```
      IF (J.LT.NJ) GOTO 805
      ANN=0.
      ANNR=0.
      GOTO 806
  805 ANN=-CNM*DYN/DYNN
      ANNR=ANN*CPD(I,JP2,K)
  806 CONTINUE

      IF (J.GT.2) GOTO 807
      ASS=0.
      ASSR=0.
      GOTO 808
  807 ASS=-CSP*DYS/DYSS
      ASSR=ASS*CPD(I,JM2,K)
  808 CONTINUE

      IF (K.LT.NK) GOTO 809
      AFF=0.
      AFFR=0.
      GOTO 810
  809 AFF=-CFM*DZF/DZFF
      AFFR=AFF*CPD(I,J,KP2)
  810 CONTINUE

      IF (K.GT.2) GOTO 811
      ABB=0.
      ABBR=0.
      GOTO 812
  811 ABB=-CBP*DZB/DZBB
      ABBR=ABB*CPD(I,J,KM2)
  812 CONTINUE



C ####################################################
C ####################################################
C *** MODIFICATION FOR DECK    BOUNDARIES

  900 CONTINUE
      IF (NOD(IM1,J,K).EQ.0) GOTO 901
      AWW=0.0
      AWWR=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,KM1).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 ##########################################################

      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
      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 MASS 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/(U0*CP0*RHO0*TA)
      QMS= 1.0
      QMS = QMS*H/(U0*RHO0)
      VOL=DXI*DYJ*DZK
      SU(I,J,K)=SU(I,J,K)+VOL*QMS/VOLT
  111 CONTINUE
```

```
C ***   SOLVE FOR C

      CALL TRID (2,2,2,NI,NJM1,NK,C)

C **** RESET CONCENTRATION AT R=0.0 AND END OF SPHERE

      DO 81 K=1,NKP1
      AVT=0.0
      DO 82 I=2,NI
      AVT=AVT+(C(I,2,K)/NIM1)
   82 CONTINUE
      DO 83 I=1,NIP1
      C(I,1,K)=AVT
   83 CONTINUE
   81 CONTINUE

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

C ***     FOR SURFACE MASS EXCHANGE WITH SURROUNDING

      DO 84 I=2,NI
      DO 84 K=2,NK
      C(I,NJP1,K)=C(I,NJ,K)
  84  CONTINUE


C ***         FOR CYLIC CONDITION

      DO 80 J=1,NJP1
      DO 80 K=1,NKP1
      C(1,J,K)=C(NI,J,K)
      C(NIP1,J,K)=C(2,J,K)
  80  CONTINUE

      RETURN
      END


C     ____________________________________________________________
C     ************************************************************
      SUBROUTINE CALU
C     ************************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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,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,COND0,VIS0,RHO0,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=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.2) IM1=NI
      IF (I.EQ.2) IM2=NIM1
      IF (I.EQ.3) IM2=NI
      IF (I.EQ.NI) IP2=3


C       CENTRAL LENGTH OF THE SCALE CONTROL VOLUME

      DXP1=XL(IP1,J,K,1,0)
      DXI =XL(I  ,J,K,1,0)
      DXM1=XL(IM1,J,K,1,0)

      DYP1=YL(I,JP1,K,1,0)
      DYJ =YL(I,J  ,K,1,0)
      DYM1=YL(I,JM1,K,1,0)

      DZP1=ZL(I,J,KP1,1,0)
      DZK =ZL(I,J,K  ,1,0)
      DZM1=ZL(I,J,KM1,1,0)
```

```
C ***    SURFACE LENGTH OF THE CONTROL VOLUME

      DXN=XL(I,JP1,K,1,2)
      DXS=XL(I,J  ,K,1,2)
      DXF=XL(I,J,KP1,1,3)
      DXB=XL(I,J,K  ,1,3)

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

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

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

      DXEE=XL(IP2,J,K,1,1)
      DXE =XL(IP1,J,K,1,1)
      DXW =XL(I  ,J,K,1,1)
      DXWW=XL(IM1,J,K,1,1)

      DYNN=YL(I,JP2,K,1,2)
      DYN =YL(I,JP1,K,1,2)
      DYS =YL(I,J  ,K,1,2)
      DYSS=YL(I,JM1,K,1,2)

      DZFF=ZL(I,J,KP2,1,3)
      DZF =ZL(I,J,KP1,1,3)
      DZB =ZL(I,J,K  ,1,3)
      DZBB=ZL(I,J,KM1,1,3)

C ***   DEFINE THE 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

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

```
C ***     USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE
C         PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.


      GNE=SILIN(R(I  ,JP1,K),R(I  ,J,K),DYP1,DYJ)*V(I  ,JP1,K)
      GNW=SILIN(R(IM1,JP1,K),R(IM1,J,K),DYP1,DYJ)*V(IM1,JP1,K)
      GSE=SILIN(R(I  ,JM1,K),R(I  ,J,K),DYM1,DYJ)*V(I  ,J  ,K)
      GSW=SILIN(R(IM1,JM1,K),R(IM1,J,K),DYM1,DYJ)*V(IM1,J  ,K)

      GE =SILIN(R(IP1,J,K),R(I  ,J,K),DXEE,DXE)*U(IP1,J,K)
      GP =SILIN(R(IM1,J,K),R(I  ,J,K),DXW ,DXE)*U(I  ,J,K)
      GW =SILIN(R(IM2,J,K),R(IM1,J,K),DXWW,DXW)*U(IM1,J,K)

      GFE=SILIN(R(I  ,J,KP1),R(I  ,J,K),DZP1,DZK)*W(I  ,J,KP1)
      GFW=SILIN(R(IM1,J,KP1),R(IM1,J,K),DZP1,DZK)*W(IM1,J,KP1)
      GBE=SILIN(R(I  ,J,KM1),R(I  ,J,K),DZM1,DZK)*W(I  ,J,K  )
      GBW=SILIN(R(IM1,J,KM1),R(IM1,J,K),DZM1,DZK)*W(IM1,J,K  )

      CE=0.5*(GE+GP)*DYZE
      CW=0.5*(GP+GW)*DYZW

      CN=SILIN(GNE,GNW,DXE,DXW)*DZXN
      CS=SILIN(GSE,GSW,DXE,DXW)*DZXS

      CF=SILIN(GFE,GFW,DXE,DXW)*DXYF
      CB=SILIN(GBE,GBW,DXE,DXW)*DXYB

      VISE=VIS(I  ,J,K)
      VISW=VIS(IM1,J,K)

      VISN=     (VIS(I  ,JP1,K)+VIS(I  ,J,K)+
     &           VIS(IM1,JP1,K)+VIS(IM1,J,K))/4.0
      VISS=     (VIS(I  ,JM1,K)+VIS(I  ,J,K)+
     &           VIS(IM1,JM1,K)+VIS(IM1,J,K))/4.0

      VISF=     (VIS(I  ,J,KP1)+VIS(I  ,J,K)+
     &           VIS(IM1,J,KP1)+VIS(IM1,J,K))/4.0
      VISB=     (VIS(I  ,J,KM1)+VIS(I  ,J,K)+
     &           VIS(IM1,J,KM1)+VIS(IM1,J,K))/4.0


      VISN1=ZXOYN*VISN
      VISS1=ZXOYS*VISS
      VISE1=YZOXE*VISE
      VISW1=YZOXW*VISW
      VISF1=XYOZF*VISF
      VISB1=XYOZB*VISB


      CEP=(ABS(CE)+CE)*DXE/DXI/16.
      CEM=(ABS(CE)-CE)*DXE/DXP1/16.
      CWP=(ABS(CW)+CW)*DXW/DXM1/16.
      CWM=(ABS(CW)-CW)*DXW/DXI/16.

      CNP=(ABS(CN)+CN)*DYP1*DYJ/(DYN*(DYN+DYS ))/8.
```

```
      CNM=(ABS(CN)-CN)*DYP1*DYJ/(DYN*(DYN+DYNN))/8.
      CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/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*CE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE+VISE1
      AW(I,J,K)= .5*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW+VISW1


      AN(I,J,K)=-.5*DYJ/DYN*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN+VISN1
      AS(I,J,K)= .5*DYJ/DYS*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS+VISS1
      AF(I,J,K)=-.5*DZK/DZF*CF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF+VISF1
      AB(I,J,K)= .5*DZK/DZB*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB+VISB1


  801 AEE=-CEM*DXE/DXEE
      AEER=AEE*UPD(IP2,J,K)
  802 CONTINUE

  803 AWW=-CWP*DXW/DXWW
      AWWR=AWW*UPD(IM2,J,K)
  804 CONTINUE

      IF (J.LT.NJ) GOTO 805
      ANN=0.
      ANNR=0.
      GOTO 806
  805 ANN=-CNM*DYN/DYNN
      ANNR=ANN*UPD(I,JP2,K)
  806 CONTINUE

      IF (J.GT.2) GOTO 807
      ASS=0.
      ASSR=0.
      GOTO 808
  807 ASS=-CSP*DYS/DYSS
      ASSR=ASS*UPD(I,JM2,K)
  808 CONTINUE

      IF (K.LT.NK) GOTO 809
      AFF=0.
      AFFR=0.
      GOTO 810
  809 AFF=-CFM*DZF/DZFF
      AFFR=AFF*UPD(I,J,KP2)
  810 CONTINUE

      IF (K.GT.2) GOTO 811
      ABB=0.
      ABBR=0.
      GOTO 812
```

```
  811 ABB=-CBP*DZB/DZBB
      ABBR=ABB*UPD(I,J,KM2)
  812 CONTINUE


C ##########################################################
C ##########################################################
C *** MODIFICATION FOR DECK    BOUNDARIES

  900 CONTINUE
      IF (NOD(IM2,J,K).EQ.0) GOTO 901
      AWW=0.0
      AWWR=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,KM1).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

      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,JM1,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)
      SU(I,2,K)=SU(I,2,K)+2.0*U(I,1,K)*AS(I,2,K)
      SP(I,NJ,K)=SP(I,NJ,K)-AN(I,NJ,K)
      AN(I,NJ,K)=0.
      AS(I,2,K)=0.
  500 CONTINUE

C ***    CYLIC CONDITION

      DO 502 K=2,NK
      DO 502 J=2,NJ
      SU(2 ,J,K)=SU(2 ,J,K)+AW(2 ,J,K)*U(1   ,J,K)
```

```
      SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*U(NIP1,J,K)
      AW(2 ,J,K)=0.0
      AE(NI,J,K)=0.0
  502 CONTINUE

C ***      FRONT AND BACK WALLS

      DO 600 I=2,NI
      DO 600 J=2,NJ

C ***        SLIP WALLS
      SP(I,J,2)=SP(I,J,2)+AB(I,J,2)
      SP(I,J,NK)=SP(I,J,NK)+AF(I,J,NK)

      AF(I,J,NK)=0.
      AB(I,J,2)=0.
 600  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
      KB=KCHPB(N)
      KE=KB+NCHPK(N)-1
      KBM1=KB-1
      KEP1=KE+1

      DO 102 J=JB,JE-1
      DO 102 K=KB,KE-1
      AE(IBM1,J,K)=0.0
      AW(IEP1,J,K)=0.0

  102 CONTINUE

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

      SP(I,JE,K)=SP(I,JE,K)-AS(I,JE,K)
      AS(I,JE,K)=0.0
  103 CONTINUE
```

```
      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
      00 79 K=1,NKP1
C     U(I,1,K)=U(I,2,K)
   79 CONTINUE


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

      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
      00 108 J=JB,JE-1
      DO 108 K=KB,KE-1
      U(I,J,K)=0.0
  108 CONTINUE
  110 CONTINUE
  112 CONTINUE
C ##################################################
C ###############################################################

      RETURN
      END



C     ________________________________________________________________
C     *****************************************************************
      SUBRCUTINE CALV
C     *****************************************************************

      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          OXXC(93),OYYC(93),OZZC(93),OXXS(93),DYYS(93),DZZS(93)
      COMMON/BL1/OX,DY,DZ,VOL,OTIME,VOLOT,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,NTMAXO,NTREAL,TIME,SORSUM,ITER
      COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,
     & CPO,PRT,CONOO,VISO,RHOO,HR,TR,TA,OTEMP,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)
     & ,COO(22,16,32),UOO(22,16,32),VOO(22,16,32),WOO(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),FPD(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=2,NK
      KP2=K+2
      KP1=K+1
      KM1=K-1
      KM2=K-2
      DO 100 J=3,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.2) IM2=NIM1
      IF (I.EQ.NI) IP2=3


C       CENTRAL LENGTH OF THE SCALE CONTROL VOLUME

      DXP1=XL(IP1,J,K,2,0)
      DXI =XL(I  ,J,K,2,0)
      DXM1=XL(IM1,J,K,2,0)

      DYP1=YL(I,JP1,K,2,0)
      DYJ =YL(I,J  ,K,2,0)
      DYM1=YL(I,JM1,K,2,0)

      DZP1=ZL(I,J,KP1,2,0)
      DZK =ZL(I,J,K  ,2,0)
      DZM1=ZL(I,J,KM1,2,0)

C ***    SURFACE LENGTH OF THE CONTROL VOLUME

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

      DYF=YL(I,J,KP1,2,3)
```

```
      DYB=YL(I,J,K  ,2,3)
      DYE=YL(IP1,J,K,2,1)
      DYW=YL(I  ,J,K,2,1)

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

C ***    CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME

      DXEE=XL(IP2,J,K,2,1)
      DXE =XL(IP1,J,K,2,1)
      DXW =XL(I  ,J,K,2,1)
      DXWW=XL(IM1,J,K,2,1)

      DYNN=YL(I,JP2,K,2,2)
      DYN =YL(I,JP1,K,2,2)
      DYS =YL(I,J  ,K,2,2)
      DYSS=YL(I,JM1,K,2,2)

      DZFF=ZL(I,J,KP2,2,3)
      DZF =ZL(I,J,KP1,2,3)
      DZB =ZL(I,J,K  ,2,3)
      DZBB=ZL(I,J,KM1,2,3)

C ***   DEFINE THE 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

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


C ***     USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE
C    &    PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.


      GEN=SILIN(R(IP1,J  ,K),R(I,J  ,K),DXP1,DXI)*U(IP1,J  ,K)
      GES=SILIN(R(IP1,JM1,K),R(I,JM1,K),DXP1,DXI)*U(IP1,JM1,K)
      GWN=SILIN(R(IM1,J  ,K),R(I,J  ,K),DXM1,DXI)*U(I  ,J  ,K)
      GWS=SILIN(R(IM1,JM1,K),R(I,JM1,K),DXM1,DXI)*U(I  ,JM1,K)
```

```
      GN =SILIN(R(I,JP1,K),R(I,J  ,K),DYNN,DYN)*V(I,JP1,K)
      GP =SILIN(R(I,JM1,K),R(I,J  ,K),DYS ,DYN)*V(I,J  ,K)
      GS =SILIN(R(I,JM2,K),R(I,JM1,K),DYSS,DYS)*V(I,JM1,K)

      GFN=SILIN(R(I,J  ,KP1),R(I,J  ,K),DZP1,DZK)*W(I,J  ,KP1)
      GFS=SILIN(R(I,JM1,KP1),R(I,JM1,K),DZP1,DZK)*W(I,JM1,KP1)
      GBN=SILIN(R(I,J  ,KM1),R(I,J  ,K),DZM1,DZK)*W(I,J  ,K  )
      GBS=SILIN(R(I,JM1,KM1),R(I,JM1,K),DZM1,DZK)*W(I,JM1,K  )

      CN=0.5*(GN+GP)*DZXN
      CS=0.5*(GP+GS)*DZXS

      CE=SILIN(GEN,GES,DYN,DYS)*DYZE
      CW=SILIN(GWN,GWS,DYN,DYS)*DYZW

      CF=SILIN(GFN,GFS,DYN,DYS)*DXYF
      CB=SILIN(GBN,GBS,DYN,DYS)*DXYB

      VISN=VIS(I,J  ,K)
      VISS=VIS(I,JM1,K)

      VISE=     (VIS(IP1,J  ,K)+VIS(I,J  ,K)+
     &           VIS(IP1,JM1,K)+VIS(I,JM1,K))/4.0
      VISW=     (VIS(IM1,J  ,K)+VIS(I,J  ,K)+
     &           VIS(IM1,JM1,K)+VIS(I,JM1,K))/4.0

      VISF=     (VIS(I,J  ,KP1)+VIS(I,J  ,K)+
     &           VIS(I,JM1,KP1)+VIS(I,JM1,K))/4.0
      VISB=     (VIS(I,J  ,KM1)+VIS(I,J  ,K)+
     &           VIS(I,JM1,KM1)+VIS(I,JM1,K))/4.0



      VISN1=ZXOYN*VISN
      VISS1=ZXOYS*VISS
      VISE1=YZOXE*VISE
      VISW1=YZOXW*VISW
      VISF1=XYOZF*VISF
      VISB1=XYOZB*VISB

C
      CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW ))/8.
      CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/8.
      CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW))/8.
      CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE ))/8.
C
      CNP=(ABS(CN)+CN)*DYN/DYJ/16.
      CNM=(ABS(CN)-CN)*DYN/DYP1/16.
      CSP=(ABS(CS)+CS)*DYS/DYM1/16.
      CSM=(ABS(CS)-CS)*DYS/DYJ/16.
C
C
      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.
C
C
      AE(I,J,K)=-.5*DXI/DXE*CE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE+VISE1
      AW(I,J,K)= .5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW+VISW1
C
      AN(I,J,K)=-.5*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN+VISN1
      AS(I,J,K)= .5*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS+VISS1
C
      AF(I,J,K)=-.5*DZK/DZF*CF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF+VISF1
      AB(I,J,K)= .5*DZK/DZB*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB+VISB1
C

  801 AEE=-CEM*DXE/DXEE
      AEER=AEE*VPD(IP2,J,K)
  802 CONTINUE

  803 AWW=-CWP*DXW/DXWW
      AWWR=AWW*VPD(IM2,J,K)
  804 CONTINUE

      IF (J.LT.NJ) GOTO 805
      ANN=0.
      ANNR=0.
      GOTO 806
  805 ANN=-CNM*DYN/DYNN
      ANNR=ANN*VPD(I,JP2,K)
  806 CONTINUE

      IF (J.GT.3) GOTO 807
      ASS=0.
      ASSR=0.
      GOTO 808
  807 ASS=-CSP*DYS/DYSS
      ASSR=ASS*VPD(I,JM2,K)
  808 CONTINUE

      IF (K.LT.NK) GOTO 809
      AFF=0.
      AFFR=0.
      GOTO 810
  809 AFF=-CFM*DZF/DZFF
      AFFR=AFF*VPD(I,J,KP2)
  810 CONTINUE

      IF (K.GT.2) GOTO 811
      ABB=0.
      ABBR=0.
      GOTO 812
  811 ABB=-CBP*DZB/DZBB
      ABBR=ABB*VPD(I,J,KM2)
  812 CONTINUE
```

```
C ##############################################################
C ###################################################
C *** MODIFICATION FOR DECK    BOUNDARIES

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

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

  902 CONTINUE
      IF (NOD(I,JM2,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,KM1).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

      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+AFF+ABB
      SP(I,J,K)=-(ROD(I,J,K)*DYS+ROD(I,JM1,K)*DYN)/(DYS+DYN)*VOLDT
      SU(I,J,K)= (ROD(I,J,K)*DYS+ROD(I,JM1,K)*DYN)/(DYS+DYN)*VOLDT
     &             *VOD(I,J,K)

      SU(I,J,K)=SU(I,J,K)+DZK*DXI*(P(I,JM1,K)-P(I,J,K))
     &          +AEER+AWWR+ANNR+ASSR+AFFR+ABBR
     &          +RE-RW+RN-RS+RF-RB+RRX+RRZ-RRY
     &   -BUOY*((R(I,J,K)-REQ(I,J,K))*DYS+(R(I,JM1,K)
     &      -REQ(I,JM1,K))*DYN)/(DYS+DYN)*VOL*SIN(ZC(K))*SIN(XC(I))
  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,3,K)=SP(I,3,K)+AS(I,3,K)
      SU(I,3,K)=SU(I,3,K)+AS(I,3,K)*V(I,2,K)
      AS(I,3,K)=0.
      AN(I,NJ,K)=0.
  500 CONTINUE

C ***    CYLIC CONDITIONS

      DO 502 K=2,NK
      DO 502 J=3,NJ
      SU(2 ,J,K)=SU(2 ,J,K)+AW(2 ,J,K)*V(1   ,J,K)
      SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*V(NIP1,J,K)
      AW(2 ,J,K)=0.0
      AE(NI,J,K)=0.0
  502 CONTINUE
```

```
C ***    FRONT AND BACK WALL

      DO 600 I=2,NI
      DO 600 J=3,NJ
      JM1=J-1

C ***          SLIP WALLS
      SP(I,J,2)=SP(I,J,2)+AB(I,J,2)
      SP(I,J,NK)=SP(I,J,NK)+AF(I,J,NK)

      AF(I,J,NK)=0.
      AB(I,J,2)=0.
  600 CONTINUE



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
      KB=KCHPB(N)
      KE=KB+NCHPK(N)-1
      KBM1=KB-1
      KEP1=KE+1

      DO 102 J=JB,JE
      DO 102 K=KB,KE-1
      SP(IBM1,J,K)=SP(IBM1,J,K)-AE(IBM1,J,K)
      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
C ***    ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS

      DO 300 K=2,NK
      DO 300 J=3,NJ
      DO 300 I=2,NI
      DXI=XL(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,NJP1
      V(I,J,1)=V(I,J,2)
      V(I,J,NKP1)=V(I,J,NK)
   74 CONTINUE
      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

      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     *******************************************************************
      SUBROUTINE CALW
C     *******************************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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,
     &  CP0,PRT,COND0,VIS0,RHO0,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
      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.2) IM2=NIM1
      IF (I.EQ.NI) IP2=3


C       CENTRAL LENGTH OF THE SCALE CONTROL VOLUME

      DXP1=XL(IP1,J,K,3,0)
      DXI =XL(I  ,J,K,3,0)
      DXM1=XL(IM1,J,K,3,0)

      DYP1=YL(I,JP1,K,3,0)
      DYJ =YL(I,J  ,K,3,0)
      DYM1=YL(I,JM1,K,3,0)

      DZP1=ZL(I,J,KP1,3,0)
      DZK =ZL(I,J,K  ,3,0)
      DZM1=ZL(I,J,KM1,3,0)

C ***    SURFACE LENGTH OF THE CONTROL VOLUME

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

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

      DZE=ZL(IP1,J,K,3,1)
      DZW=ZL(I  ,J,K,3,1)
      DZN=ZL(I,JP1,K,3,2)
```

```
      DZS=ZL(I,J  ,K,3,2)

C ***    CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME

      DXEE=XL(IP2,J,K,3,1)
      DXE =XL(IP1,J,K,3,1)
      DXW =XL(I  ,J,K,3,1)
      DXWW=XL(IM1,J,K,3,1)

      DYNN=YL(I,JP2,K,3,2)
      DYN =YL(I,JP1,K,3,2)
      DYS =YL(I,J  ,K,3,2)
      DYSS=YL(I,JM1,K,3,2)

      DZFF=ZL(I,J,KP2,3,3)
      DZF =ZL(I,J,KP1,3,3)
      DZB =ZL(I,J,K  ,3,3)
      DZBB=ZL(I,J,KM1,3,3)

C ***   DEFINE THE 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

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


C ***     USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE
C    &    PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.


      GNF=SILIN(R(I,JP1,K  ),R(I,J,K  ),DYP1,DYJ)*V(I,JP1,K  )
      GNB=SILIN(R(I,JP1,KM1),R(I,J,KM1),DYP1,DYJ)*V(I,JP1,KM1)
      GSF=SILIN(R(I,JM1,K  ),R(I,J,K  ),DYM1,DYJ)*V(I,J  ,K  )
      GSB=SILIN(R(I,JM1,KM1),R(I,J,KM1),DYM1,DYJ)*V(I,J  ,KM1)

      GF =SILIN(R(I,J,KP1),R(I,J,K  ),DZFF,DZF)*W(I,J,KP1)
      GP =SILIN(R(I,J,KM1),R(I,J,K  ),DZB ,DZF)*W(I,J,K  )
      GB =SILIN(R(I,J,KM2),R(I,J,KM1),DZBB,DZB)*W(I,J,KM1)

      GEF=SILIN(R(IP1,J,K  ),R(I,J,K  ),DXP1,DXI)*U(IP1,J,K  )
      GEB=SILIN(R(IP1,J,KM1),R(I,J,KM1),DXP1,DXI)*U(IP1,J,KM1)
      GWF=SILIN(R(IM1,J,K  ),R(I,J,K  ),DXM1,DXI)*U(I  ,J,K  )
```

```
      GWB=SILIN(R(IM1,J,KM1),R(I,J,KM1),DXM1,DXI)*U(I   ,J,KM1)

      CF=0.5*(GF+GP)*DXYF
      CB=0.5*(GP+GB)*DXYB

      CN=SILIN(GNF,GNB,DZF,DZB)*DZXN
      CS=SILIN(GSF,GSB,DZF,DZB)*DZXS

      CE=SILIN(GEF,GEB,DZF,DZB)*DYZE
      CW=SILIN(GWF,GWB,DZF,DZB)*DYZW

      VISF=VIS(I,J,K  )
      VISB=VIS(I,J,KM1)

      VISN=     (VIS(I,JP1,K  )+VIS(I,J,K  )+
     &           VIS(I,JP1,KM1)+VIS(I,J,KM1))/4.0
      VISS=     (VIS(I,JM1,K  )+VIS(I,J,K  )+
     &           VIS(I,JM1,KM1)+VIS(I,J,KM1))/4.0

      VISE=     (VIS(IP1,J,K  )+VIS(I,J,K  )+
     &           VIS(IP1,J,KM1)+VIS(I,J,KM1))/4.0
      VISW=     (VIS(IM1,J,K  )+VIS(I,J,K  )+
     &           VIS(IM1,J,KM1)+VIS(I,J,KM1))/4.0


      VISN1=ZXOYN*VISN
      VISS1=ZXOYS*VISS
      VISE1=YZOXE*VISE
      VISW1=YZOXW*VISW
      VISF1=XYOZF*VISF
      VISB1=XYOZB*VISB

C
      CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW  ))/8.
      CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/8.
      CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW))/8.
      CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE  ))/8.
C
      CNP=(ABS(CN)+CN)*DYP1*DYJ/(DYN*(DYN+DYS  ))/8.
      CNM=(ABS(CN)-CN)*DYP1*DYJ/(DYN*(DYN+DYNN))/8.
      CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/8.
      CSM=(ABS(CS)-CS)*DYM1*DYJ/(DYS*(DYS+DYN  ))/8.
C
C
      CFP=(ABS(CF)+CF)*DZF/DZK/16.
      CFM=(ABS(CF)-CF)*DZF/DZP1/16.
      CBP=(ABS(CB)+CB)*DZB/DZM1/16.
      CBM=(ABS(CB)-CB)*DZB/DZK/16.
C
      AE(I,J,K)=-.5*DXI/DXE*CE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE+VISE1
      AW(I,J,K)= .5*DXI/DXW*CW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW+VISW1
      AN(I,J,K)=-.5*DYJ/DYN*CN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN+VISN1
      AS(I,J,K)= .5*DYJ/DYS*CS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS+VISS1
C
      AF(I,J,K)=-.5*CF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF+VISF1
```

```
      AB(I,J,K)= .5*CB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB+VISD1
C

  801 AEE=-CEM*DXE/DXEE
      AEER=AEE*WPD(IP2,J,K)
  802 CONTINUE

  803 AWW=-CWP*DXW/DXWW
      AWWR=AWW*WPD(IM2,J,K)
  804 CONTINUE

      IF (J.LT.NJ) GOTO 805
      ANN=0.
      ANNR=0.
      GOTO 806
  805 ANN=-CNM*DYN/DYNN
      ANNR=ANN*WPD(I,JP2,K)
  806 CONTINUE

      IF (J.GT.2) GOTO 807
      ASS=0.
      ASSR=0.
      GOTO 808
  807 ASS=-CSP*DYS/DYSS
      ASSR=ASS*WPD(I,JM2,K)
  808 CONTINUE

      IF (K.LT.NK) GOTO 809
      AFF=0.
      AFFR=0.
      GOTO 810
  809 AFF=-CFM*DZF/DZFF
      AFFR=AFF*WPD(I,J,KP2)
  810 CONTINUE

      IF (K.GT.3) GOTO 811
      ABB=0.
      ABBR=0.
      GOTO 812
  811 ABB=-CBP*DZB/DZBB
      ABBR=ABB*WPD(I,J,KM2)
  812 CONTINUE


C ##############################################################
C ##############################################
C *** MODIFICATION FOR DECK    BOUNDARIES

  900 CONTINUE
      IF (NOD(IM1,J,K).EQ.0) GOTO 901
      AWW=0.0
      AWWR=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)
      RRZ=(AVG22-ARU22)*DXI*(DYF-DYB)+
     &   (AVG11-ARU11)*DYJ*(DXF-DXB)


      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)*DZB+ROD(I,J,KM1)*DZF)/(DZB+DZF)*VOLDT
      SU(I,J,K)= (ROD(I,J,K)*DZB+ROD(I,J,KM1)*DZF)/(DZB+DZF)*VOLDT
     &             *WOD(I,J,K)
      SU(I,J,K)=SU(I,J,K)+DXI*DYJ*(P(I,J,KM1)-P(I,J,K))
     &         +AEER+AWWR+ANNR+ASSR+AFFR+ABBR
     &         +RE-RW+RN-RS+RF-RB+RRY+RRX-RRZ
     & -BUOY*((R(I,J,K)-REQ(I,J,K))*DZB*COS(ZC(K))+(R(I,J,
     &  KM1)-REQ(I,J,KM1))*DZF*COS(ZC(KM1)))/(DZB+DZF)*VOL*SIN(XC(I))
 100  CONTINUE

C ***     TAKE CARE OF B.C. THRU AN,AS,AE,AW,AP AND SU
C
C ***     RADIUS DIRECTION

      DO 500 K=3,NK
      DO 500 I=2,NI
      KM1=K-1
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*W(I,1,K)*AS(I,2,K)
      SP(I,NJ,K)=SP(I,NJ,K)-AN(I,NJ,K)
      AS(I,2,K)=0.
      AN(I,NJ,K)=0.
 500  CONTINUE

C ***   CYLIC CONDITIONS

      DO 502 K=3,NK
      DO 502 J=2,NJ
      SU(2 ,J,K)=SU(2 ,J,K)+AW(2 ,J,K)*W(1    ,J,K)
      SU(NI,J,K)=SU(NI,J,K)+AE(NI,J,K)*W(NIP1,J,K)
      AW(2 ,J,K)=0.0
      AE(NI,J,K)=0.0
 502  CONTINUE

C ***     FRONT AND BACK WALL
      DO 600 I=2,NI
      DO 600 J=2,NJ
      SP(I,J,NK)=SP(I,J,NK)+AF(I,J,NK)
      SP(I,J,3 )=SP(I,J,3 )+AB(I,J,3 )
      AF(I,J,NK)=0.
      AB(I,J,3)=0.
  600 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
      KB=KCHPB(N)
      KE=KB+NCHPK(N)-1
      KBM1=KB-1
      KEP1=KE+1


      DO 102 J=JB,JE-1
      DO 102 K=KB,KE
      SP(IBM1,J,K)=SP(IBM1,J,K)-AE(IBM1,J,K)
      SU(IBM1,J,K)=SU(IBM1,J,K)+AE(IBM1,J,K)*WFAN(N)*2.0
      AE(IBM1,J,K)=0.0


      SP(IE,J,K)=SP(IE,J,K)-AW(IE,J,K)
      SU(IE,J,K)=SU(IE,J,K)+AW(IE,J,K)*WFAN(N)*2.0
      AW(IE,J,K)=0.0

  102 CONTINUE

      DO 103 I=IB,IE-1
      DO 103 K=KB,KE
      SP(I,JBM1,K)=SP(I,JBM1,K)-AN(I,JBM1,K)
      SU(I,JBM1,K)=SU(I,JBM1,K)+AN(I,JBM1,K)*WFAN(N)*2.0
      AN(I,JBM1,K)=0.0

      SP(I,JE,K)=SP(I,JE,K)-AS(I,JE,K)
      SU(I,JE,K)=SU(I,JE,K)+AS(I,JE,K)*WFAN(N)*2.0
      AS(I,JE,K)=0.0
  103 CONTINUE

      DO 106 I=IB,IE-1
      DO 106 J=JB,JE-1
      SU(I,J,KBM1)=SU(I,J,KBM1)+AF(I,J,KBM1)*WFAN(N)
      SU(I,J,KEP1)=SU(I,J,KEP1)+AB(I,J,KEP1)*WFAN(N)
      AF(I,J,KBM1)=0.0
      AB(I,J,KEP1)=0.0
  106 CONTINUE

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

```
      DO 104 I=IB,IE-1
      DO 104 J=JB,JE-1
      DO 104 K=KB,KE
      SP(I,J,K)=-1.0E2
      AW(I,J,K)=0.
      AE(I,J,K)=0.
      AS(I,J,K)=0.
      AN(I,J,K)=0.
      AB(I,J,K) = 0.
      AF(I,J,K) = 0.
      SU(I,J,K)=1.0E2 * WFAN(N)
  104 CONTINUE
  101 CONTINUE
  105 CONTINUE

C #######################################
C ########################################################

C ***    ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS


      DO 301 K=3,NK
      DO 301 J=2,NJ
      DO 301 I=2,NI
      DXI=XL(I,J,K,3,0)
      DYJ=YL(I,J,K,3,0)
      DXY=DXI*DYJ
      AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
      DW(I,J,K)=DXY/AP(I,J,K)
  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     ************************************************************
      SUBROUTINE CALP
C     ************************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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,NTMAXO,NTREAL,TIME,SORSUM,ITER
      COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,
     &  CP0,PRT,COND0,VIS0,RHO0,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),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=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)
      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 HORIZENTAL 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)-ROO(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
      KB=KCHPB(N)
      KE=KB+NCHPK(N)-1
      KBM1=KB-1
      KEP1=KE+1

      DO 102 J=JB,JE-1
      DO 102 K=KB,KE-1
      AE(IBM1,J,K)=0.0
      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,JE,K)=0.0
  103 CONTINUE

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

C *** FOR THE CELLS INSIDE OF THE DECKS

      DO 104 I=IB,IE-1
      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 300 J=2,NJ
      DO 300 I=2,NI
      DO 300 K=2,NK
      AP(I,J,K)=AN(I,J,K)+AS(I,J,K)+AE(I,J,K)+AW(I,J,K)-SP(I,J,K)
     &    +AF(I,J,K)+AB(I,J,K)
  300 CONTINUE

C ***    SOLUTION OF FINITE DIFFERENCE EQUATION

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

C *** THIS IS FOR CKECKING


      DO 161 I=1,NIP1
C     WRITE (6,*) I
 949  FORMAT ( '  AW ')
C     WRITE (6,949)
C     WRITE (6,999) ((AW(I,J,K),K=1,NKP1),J=1,NJP1)
 161  CONTINUE
      DO 160 I=1,NIP1
C     WRITE (6,*) I
 948  FORMAT ( '  AE ')
C     WRITE (6,948)
C     WRITE (6,999) ((AE(I,J,K),K=1,NKP1),J=1,NJP1)
 160  CONTINUE
      DO 170 I=1,NIP1
C     WRITE (6,*) I
 958  FORMAT ( '  AB ')
C     WRITE (6,958)
C     WRITE (6,999) ((AB(I,J,K),K=1,NKP1),J=1,NJP1)
 170  CONTINUE
      DO 180 I=1,NIP1
C     WRITE (6,*) I
 968  FORMAT ( '  AF ')
C     WRITE (6,968)
C     WRITE (6,999) ((AF(I,J,K),K=1,NKP1),J=1,NJP1)
 180  CONTINUE
C     WRITE (6,999) ((SU(I,5,K),K=1,NKP1),I=1,NIP1)
      DO 190 I=1,NIP1
C     WRITE (6,*) I
 978  FORMAT ( '  SU ')
C     WRITE (6,978)
C     WRITE (6,999) ((SU(I,J,K),K=1,NKP1),J=1,NJP1)
 190  CONTINUE
      DO 191 I=1,NIP1
C     WRITE (6,*) I
```

```
C     WRITE (6,988)
 988  FORMAT ( '  PP ')
C     WRITE (6,999) ((PP(I,J,K),J=1,NJP1),K=7,7)
 191  CONTINUE
 999  FORMAT (12E10.3)



C ***    CORRECT VELOCITIES AND PRESSURE
C
C ***    CORRECTION FOR VELOCITY U

      DO 600 I=2,NI
      IM1=I-1
      IF (I.EQ.2) IM1=NI
      DO 600 J=2,NJ
      DO 600 K=2,NK
      U(I,J,K)=U(I,J,K)+DU(I,J,K)*(PP(IM1,J,K)-PP(I,J,K))
  600 CONTINUE

C ***    CORRECTION FOR VELOCITY V

      DO 603 J=3,NJ
      JM1=J-1
      DO 603 K=2,NK
      DO 603 I=2,NI
      V(I,J,K)=V(I,J,K)+DV(I,J,K)*(PP(I,JM1,K)-PP(I,J,K))
  603 CONTINUE

C ***   CORRECTION OF VELOCITY W

      DO 604 K=3,NK
      KM1=K-1
      DO 604 I=2,NI
      DO 604 J=2,NJ
      W(I,J,K)=W(I,J,K)+DW(I,J,K)*(PP(I,J,KM1)-PP(I,J,K))
 604  CONTINUE


C ***    CORRECTION FOR PRESSURE P

      DO 606 J=2,NJ
      DO 606 I=1,NIP1
      DO 606 K=1,NK
      P(I,J,K)=P(I,J,K)+PP(I,J,K)
      PP(I,J,K)=0.
  606 CONTINUE

C *** THIS IS FOR R=0.0 CASE

      DO 75 I=1,NIP1
      DO 75 K=1,NKP1
C     U(I,1,K)=U(I,2,K)
C     W(I,1,K)=W(I,2,K)
C     V(I,2,K)=V(I,3,K)
```

```
   75  CONTINUE


C ***  MODIFICATION FOR R=0.0
C
       DO 55 K=2,NK
       VY=0.0
       VX=0.0
       VZ=0.0
       DO 50 I=2,NI
       VY=VY+U(I,2,K)*COS(XS(I))
       VX=VX-U(I,2,K)*SIN(XS(I))
    50 CONTINUE

       DO 51 I=2,NI
       VY=VY+V(I,3,K)*SIN(XC(I))
       VX=VX+V(I,3,K)*COS(XC(I))
       VZ=VZ+W(I,2,K)
    51 CONTINUE


C ***  FIND THE VELOCITIES AT R=0.0

       DO 52 I=1,NIP1
       U(I,1,K)=(-VX*SIN(XS(I))+VY*COS(XS(I)))/NIM1
       V(I,2,K)= (VX*COS(XC(I))+VY*SIN(XC(I)))/NIM1
       W(I,1,K)=VZ/NIM1
    52 CONTINUE
    55 CONTINUE



C ***   THIS IS FOR THE CYLINDER ONLY (CYLIC CONDITION)

       DO 76 J=1,NJP1
       DO 76 K=1,NKP1
       U(1,J,K)=U(NI,J,K)
       U(NIP1,J,K)=U(2,J,K)
       V(1,J,K)=V(NI,J,K)
       V(NIP1,J,K)=V(2,J,K)
       W(1,J,K)=W(NI,J,K)
       W(NIP1,J,K)=W(2,J,K)
   76  CONTINUE

C ***    THIS FOR SPHERE ONLY

       DO 77 I=1,NIP1
       DO 77 J=1,NJP1
       U(I,J,1)=U(I,J,2)
       V(I,J,1)=V(I,J,2)
       W(I,J,2)=W(I,J,3)
       U(I,J,NKP1)=U(I,J,NK)
       V(I,J,NKP1)=V(I,J,NK)
       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)


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)
      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)

      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
C     SMP(I,J,K)=-CE+CW-CN+CS-CF+CB
      SMP(I,J,K)=-(R(I,J,K)-ROD(I,J,K))*VOL/DTIME-CE+CW-CN+CS-CF+CB

C ***    SORSUM IS ACTUAL MASS INCREASE OR DECREASE FROM CONTINUITY
C        EQUATUON , THIS WILL COMPARE TO SOURCE

      SORSUM=SORSUM+SMP(I,J,K)

C ***    RESORM IS SUM OF THE ABSOLUTE VALUE OF SMP(I,J,K)

      RESORM(ITER)=RESORM(ITER)+ABS(SMP(I,J,K))
  700 CONTINUE
      RETURN
      END



C **********************************************************************
      SUBROUTINE TRID(IST,JST,KST,ISP,JSP,KSP,PHI)
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/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)
      DIMENSION A(99),B(99),C(99),PHI(22,16,32)

C     GOTO 405
      ISTM1=IST-1
      A(ISTM1)=0.
      C(ISTM1)=0.
      DO 100 J=JST,JSP
      DO 100 K=KST,KSP
      DO 101 I=IST,ISP
      A(I)=AE(I,J,K)
      B(I)=AW(I,J,K)
      C(I)=AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)
     &    +AF(I,J,K)*PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
      TERM=1./(AP(I,J,K)-B(I)*A(I-1))
      IF (ABS(A(I)).LE.1.0E-70) A(I)=0.0
      IF (ABS(B(I)).LE.1.0E-70) B(I)=0.0
      IF (ABS(C(I)).LE.1.0E-70) C(I)=0.0
      IF (ABS(TERM).LE.1.0E-70) TERM=0.0
      A(I)=A(I)*TERM
      C(I)=(C(I)+B(I)*C(I-1))*TERM
  101 CONTINUE
      PHI(ISP,J,K)=C(ISP)
      ISTA=IST+1
      DO 102 II=ISTA,ISP
      I=IST+ISP-II
      IP1=I+1
```

```
      PHI(I,J,K)=A(I)*PHI(IP1,J,K)+C(I)
  102 CONTINUE
  100 CONTINUE

      DO 2000 J=JST,JSP
      DO 2000 K=KST,KSP
      PHI(IST-1,J,K)=PHI(ISP,J,K)
      PHI(ISP+1,J,K)=PHI(IST,J,K)
 2000 CONTINUE


      JSTM1=JST-1
      A(JSTM1)=0.
      C(JSTM1)=0.
      DO 200 K=KST,KSP
      DO 200 I=IST,ISP
      DO 201 J=JST,JSP
      A(J)=AN(I,J,K)
      B(J)=AS(I,J,K)
      C(J)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)
     &    +AF(I,J,K)*PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
      TERM=1./(AP(I,J,K)-B(J)*A(J-1))
      IF (ABS(A(J)).LE.1.0E-70) A(J)=0.0
      IF (ABS(B(J)).LE.1.0E-70) B(J)=0.0
      IF (ABS(C(J)).LE.1.0E-70) C(J)=0.0
      IF (ABS(TERM).LE.1.0E-70) TERM=0.0
      A(J)=A(J)*TERM
      C(J)=(C(J)+B(J)*C(J-1))*TERM
  201 CONTINUE
      PHI(I,JSP,K)=C(JSP)
      JSTA=JST+1
      DO 202 JJ=JSTA,JSP
      J=JST+JSP-JJ
      JP1=J+1
      PHI(I,J,K)=A(J)*PHI(I,JP1,K)+C(J)
  202 CONTINUE
  200 CONTINUE

      DO 2001 J=JST,JSP
      DO 2001 K=KST,KSP
      PHI(IST-1,J,K)=PHI(ISP,J,K)
      PHI(ISP+1,J,K)=PHI(IST,J,K)
 2001 CONTINUE


      KSTM1=KST-1
      A(KSTM1)=0.
      C(KSTM1)=0.
      DO 300 I=IST,ISP
      DO 300 J=JST,JSP
      DO 301 K=KST,KSP
      A(K)=AF(I,J,K)
      B(K)=AB(I,J,K)
      C(K)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)
     &    +AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+SU(I,J,K)
```

```
      TERM=1./(AP(I,J,K)-B(K)*A(K-1))
      IF (ABS(A(K)).LE.1.0E-70) A(K)=0.0
      IF (ABS(B(K)).LE.1.0E-70) B(K)=0.0
      IF (ABS(C(K)).LE.1.0E-70) C(K)=0.0
      IF (ABS(TERM).LE.1.0E-70) TERM=0.0
      A(K)=A(K)*TERM
      C(K)=(C(K)+B(K)*C(K-1))*TERM
  301 CONTINUE
      PHI(I,J,KSP)=C(KSP)
      KSTA=KST+1
      DO 302 KK=KSTA,KSP
      K=KST+KSP-KK
      KP1=K+1
      PHI(I,J,K)=A(K)*PHI(I,J,KP1)+C(K)
  302 CONTINUE
  300 CONTINUE

      DO 2002 J=JST,JSP
      DO 2002 K=KST,KSP
      PHI(IST-1,J,K)=PHI(ISP,J,K)
      PHI(ISP+1,J,K)=PHI(IST,J,K)
 2002 CONTINUE


      GOTO 700

 4405 CONTINUE
  405 KSP1=KSP+1
      B(KSP1)=0.
      C(KSP1)=0.
      DO 600 II=IST,ISP
      I=IST+ISP-II
      DO 600 JJ=JST,JSP
      J=JST+JSP-JJ
      DO 601 KK=KST,KSP
      K=KSP+KST-KK
      KP1=K+1
      A(K)=AF(I,J,K)
      B(K)=AB(I,J,K)
      C(K)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)+AN(I,J,K)*
     &     PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+SU(I,J,K)
      TERM=1./(AP(I,J,K)-A(K)*B(K+1))
      B(K)=B(K)*TERM
      C(K)=(C(K)+A(K)*C(K+1))*TERM
      IF (ABS(A(K)).LE.1.0E-70) A(K)=0.0
      IF (ABS(B(K)).LE.1.0E-70) B(K)=0.0
      IF (ABS(C(K)).LE.1.0E-70) C(K)=0.0
  601 CONTINUE
      PHI(I,J,KST)=C(KST)
      KSTP1=KST+1
      DO 602 K=KSTP1,KSP
      PHI(I,J,K)=B(K)*PHI(I,J,K-1)+C(K)
  602 CONTINUE
  600 CONTINUE
```

```
      DO 2003 J=JST,JSP
      DO 2003 K=KST,KSP
      PHI(IST-1,J,K)=PHI(ISP,J,K)
      PHI(ISP+1,J,K)=PHI(IST,J,K)
 2003 CONTINUE


      JSP1=JSP+1
      B(JSP1)=0.
      C(JSP1)=0.
      DO 500 KK=KST,KSP
      K=KST+KSP-KK
      DO 500 II=IST,ISP
      I=IST+ISP-II
      DO 501 JJ=JST,JSP
      J=JSP+JST-JJ
      JP1=J+1
      A(J)=AN(I,J,K)
      B(J)=AS(I,J,K)
      C(J)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)+AF(I,J,K)*
     &     PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
      TERM=1./(AP(I,J,K)-A(J)*B(J+1))
      B(J)=B(J)*TERM
      C(J)=(C(J)+A(J)*C(J+1))*TERM
      IF (ABS(A(J)).LE.1.0E-70) A(J)=0.0
      IF (ABS(B(J)).LE.1.0E-70) B(J)=0.0
      IF (ABS(C(J)).LE.1.0E-70) C(J)=0.0
 501  CONTINUE
      PHI(I,JST,K)=C(JST)
      JSTP1=JST+1
      DO 502 J=JSTP1,JSP
      PHI(I,J,K)=B(J)*PHI(I,J-1,K)+C(J)
  502 CONTINUE
  500 CONTINUE

      DO 2004 J=JST,JSP
      DO 2004 K=KST,KSP
      PHI(IST-1,J,K)=PHI(ISP,J,K)
      PHI(ISP+1,J,K)=PHI(IST,J,K)
 2004 CONTINUE


      ISP1=ISP+1
      B(ISP1)=0.
      C(ISP1)=0.
      DO 400 JJ=JST,JSP
      J=JST+JSP-JJ
      DO 400 KK=KST,KSP
      K=KST+KSP-KK
      DO 401 II=IST,ISP
      I=ISP+IST-II
      IP1=I+1
      A(I)=AE(I,J,K)
      B(I)=AW(I,J,K)
      C(I)=AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+AF(I,J,K)*
```

```
     &     PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
      TERM=1./(AP(I,J,K)-A(I)*B(I+1))
      B(I)=B(I)*TERM
      C(I)=(C(I)+A(I)*C(I+1))*TERM
      IF (ABS(A(I)).LE.1.0E-70) A(I)=0.0
      IF (ABS(B(I)).LE.1.0E-70) B(I)=0.0
      IF (ABS(C(I)).LE.1.0E-70) C(I)=0.0
  401 CONTINUE
      PHI(IST,J,K)=C(IST)
      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(ISP+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,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,COND0,VISO,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      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 U0,TA,PRT,RHO0,CP0,VISO,NTMAXO/
     &     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),DYYC(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 ***   RENERATION 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)=0.01*2.0*PI
C     DO 19 I=4,13
C     XS(I)=(I-3)*DX*2.0*PI
C 19  CONTINUE
C
C     XS(14)=XS(13)
C     XS(13)=XS(14)-0.01*2.0*PI
C     DO 18 I=15,NIP1
C     XS(I)=XS(14)+(I-14)*DX*2.0*PI
C  18 CONTINUE
C     XS(NIP2)=XS(NIP1)+XS(3)


      YS(1)=0.000
      YS(2)=0.025
C     YS(3)=0.05
      DO 3 J=3,NJ
      YS(J)=(J-2)*DY
    3 CONTINUE
      YS(NJP1)=YS(NJ)
      YS(NJ  )=YS(NJP1)-3./8./12./9.6
      YS(NJP2)=YS(NJP1)+3./8./12./9.6

CC    DO 3 J=4,NJP2
CC    YS(J)=(J-3)*DY
CC 3  CONTINUE
      DO 4 I=1,NIP1
      IP1=I+1
      DXXC(I)=XS(IP1)-XS(I)
    4 CONTINUE

      DXXC(NIP2)=DXXC(NIP1)
      DO 5 I=2,NIP2
      IM1=I-1
      DXXS(I)=.5*(DXXC(I)+DXXC(IM1))
    5 CONTINUE
      DXXS(1)=DXXS(2)
```

```
      DO 7 J=1,NJP1
      JP1=J+1
      DYYC(J)=YS(JP1)-YS(J)
    7 CONTINUE

      DYYC(NJP2)=DYYC(NJP1)
      DO 8 J=2,NJP2
      JM1=J-1
      DYYS(J)=.5*(DYYC(J)+DYYC(JM1))
    8 CONTINUE
      DYYS(1)=DYYS(2)

      DO 20 I=1,NIP2
      XC(I)=XS(I)+DXXC(I)/2.0
   20 CONTINUE

      DO 21 J=1,NJP2
      YC(J)=YS(J)+DYYC(J)/2.0
   21 CONTINUE


      DO 9 K=4,NA
      ZS(K)=(K-3)*DZ
    9 CONTINUE

      DO 30 K=NBP1,NK
      ZS(K)=ZS(NA)+(K-NB)*DZ
   30 CONTINUE

      DO 31 K=NAP1,NB
      ZS(K)=PI/2.
   31 CONTINUE

      ZS(1)=0.0
      ZS(2)=0.05
      ZS(3)=0.10
C     ZS(NKP1)=ZS(NKM1)
C     ZS(NK)=ZS(NKP1)-0.05
C     ZS(NKM1)=ZS(NKP1)-0.10
C     ZS(NKP2)=ZS(NKP1)+0.05

      ZS(NKP2)=ZS(NK)
      ZS(NKP1)=ZS(NKP2)-0.05
      ZS(NK)=ZS(NKP2)-0.10


      DO 10 K=1,NKP1
      IF (K.GE.NA.AND.K.LT.NB) GOTO 10
      KP1=K+1
      DZZC(K)=ZS(KP1)-ZS(K)
   10 CONTINUE

      DO 32 K=NA,NBM1
      DZZC(K)=2.854/(NB-NA)
```

```
 32 CONTINUE

    DZZC(NKP2)=DZZC(NKP1)

    DO 11 K=2,NKP2
C   IF (K.EQ.NA.OR.K.EQ.NB) GOTO 11
    KM1=K-1
    DZZS(K)=.5*(DZZC(K)+DZZC(KM1))
 11 CONTINUE

    DZZS(1)=DZZS(2)
    DO 22 K=1,NKP2
    IF (K.GE.NA.AND.K.LT.NB) GOTO 22
    ZC(K)=ZS(K)+DZZC(K)/2.0
 22 CONTINUE

    DO 33 K=NA,NBM1
    ZC(K)=PI/2.
 33 CONTINUE

    IF (YS(1).LT.0.0) YS(1)=0.0
    IF (YC(1).LT.0.0) YC(1)=0.0
    PRINT *
    PRINT *,'     INPUT COORDINATE OF THE TANK IN THE ORDER OF '
    PRINT *,'    I       XS       YS        ZS        XC        YC',
   &    '        ZC        DXXS      DYYS      DZZS      DXXC      '
   & ,  'DYYC      DZZC'
    DO 12 I=1,NKP2
    WRITE(6,102) I,XS(I),YS(I),ZS(I),XC(I),YC(I),ZC(I),
   &             DXXS(I),DYYS(I),DZZS(I),DXXC(I),DYYC(I),DZZC(I)
102 FORMAT(2X,I4,12(2X,F8.5))
 12 CONTINUE

    RETURN
    END



C     **********************
      FUNCTION XL(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     WHEN M OR N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C                     HALF CELL (STAGGERED CELL)                *
C     WHEN M OR N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C                     HALF CELL (STAGGERED CELL)                *
C     WHEN M =  N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE*
C                     WHOLE CELL                                *
C     WHEN M =  N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C                     WHOLE CELL                                *
C     WHEN M =  N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C                     WHOLE CELL                                *
C*****************************************************************
```

```
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)
      X1=XC(I)
      X2=YC(J)
      X3=ZC(K)
      DXL=DXXC(I)
      IF(M.EQ.N) GOTO 100

      IF(M.EQ.1.OR.N.EQ.1) X1=XS(I)
      IF(M.EQ.1.OR.N.EQ.1) DXL=DXXS(I)
      IF(M.EQ.2.OR.N.EQ.2) X2=YS(J)
      IF(M.EQ.3.OR.N.EQ.3) X3=ZS(K)
      GOTO 1000
  100 IF(M.EQ.1) X1=XC(I-1)
      IF(M.EQ.1) DXL=DXXC(I-1)
      IF(M.EQ.2) X2=YC(J-1)
      IF(M.EQ.3) X3=ZC(K-1)
 1000 CONTINUE
      XL=X2*SIN(X3)*DXL
      RETURN
      END


C     **********************
      FUNCTION YL(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     WHEN M OR N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C                     HALF CELL (STAGGERED CELL)                *
C     WHEN M OR N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C                     HALF CELL (STAGGERED CELL)                *
C     WHEN M =  N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE*
C                     WHOLE CELL                                *
C     WHEN M =  N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C                     WHOLE CELL                                *
C     WHEN M =  N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C                     WHOLE CELL                                *
C***************************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)
      X1=XC(I)
      X2=YC(J)
      X3=ZC(K)
      DYL=DYYC(J)
      IF(M.EQ.N) GOTO 100

      IF(M.EQ.2.OR.N.EQ.2) X2=YS(J)
      IF(M.EQ.2.OR.N.EQ.2) DYL=DYYS(J)
      IF(M.EQ.1.OR.N.EQ.1) X1=XS(I)
      IF(M.EQ.3.OR.N.EQ.3) X3=ZS(K)
      GOTO 1000
  100 IF(M.EQ.2) X2=YC(J-1)
```

```
      IF(M.EQ.2) DYL=DYYC(J-1)
      IF(M.EQ.1) X1=XC(I-1)
      IF(M.EQ.3) X3=ZC(K-1)
 1000 CONTINUE
      YL=1.00*DYL
      RETURN
      END


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     WHEN M OR N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C                   HALF CELL (STAGGERED CELL)                 *
C     WHEN M OR N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C                   HALF CELL (STAGGEREO CELL)                 *
C     WHEN M =  N = 1 THEN SHIFT CELL IN THE NEG X DIRECTION ONE*
C                   WHOLE CELL                                 *
C     WHEN M =  N = 2 THEN SHIFT CELL IN THE NEG Y DIRECTION ONE*
C                   WHOLE CELL                                 *
C     WHEN M =  N = 3 THEN SHIFT CELL IN THE NEG Z DIRECTION ONE*
C                   WHOLE CELL                                 *
C*************************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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     *************************
      FUNCTION SILIN(V1,V2,D1,D2)
C     *************************
C     IF (D1.EQ.0.0.AND.D2.EQ.0.0) D1=0.1
C     IF (D1.EQ.0.0.AND.D2.EQ.0.0) D2=0.1
      SILIN=(V1*D2+V2*D1)/(D1+D2)
      RETURN
      END


C     ********************************************
      FUNCTION BILIN(V1,V2,D1,D2,V3,V4,D3,D4,D5,D6)
C     ********************************************
      V12=(V1*D2+V2*D1)/(D1+D2)
      V34=(V3*D4+V4*D3)/(D3+D4)
      BILIN=(V12*D6+V34*D5)/(D5+D6)
      END


C     *****************
      SUBROUTINE STRESS
C     *****************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(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/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)


      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

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)

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

      DXEE=XL(IP2,J,K,0,1)
      DXE =XL(IP1,J,K,0,1)
      DXW =XL(I  ,J,K,0,1)
      DXWW=XL(IM1,J,K,0,1)

      DYNN=YL(I,JP2,K,0,2)
      DYN =YL(I,JP1,K,0,2)
      DYS =YL(I,J  ,K,0,2)
      DYSS=YL(I,JM1,K,0,2)

      DZFF=ZL(I,J,KP2,0,3)
      DZF =ZL(I,J,KP1,0,3)
      DZB =ZL(I,J,K  ,0,3)
      DZBB=ZL(I,J,KM1,0,3)
```

```
      UBAR=0.5*(U(IP1,J,K)+U(I,J,K))
      VBAR=0.5*(V(I,JP1,K)+V(I,J,K))
      WBAR=0.5*(W(I,J,KP1)+W(I,J,K))

      DXY=DXI*DYJ
      DYZ=DYJ*DZK
      DZX=DZK*DXI

      SIG11(I,J,K)=2.*VIS(I,J,K)*((U(IP1,J,K)-U(I,J,K))/DXI
     &             +VBAR*(DXN-DXS)/DXY
     &             +WBAR*(DXF-DXB)/DZX)

      SIG22(I,J,K)=2.*VIS(I,J,K)*((V(I,JP1,K)-V(I,J,K))/DYJ
     &             +WBAR*(DYF-DYB)/DYZ
     &             +UBAR*(DYE-DYW)/DXY)

      SIG33(I,J,K)=2.*VIS(I,J,K)*((W(I,J,KP1)-W(I,J,K))/DZK
     &             +UBAR*(DZE-DZW)/DZX
     &             +VBAR*(DZN-DZS)/DYZ)
 100  CONTINUE

      DO 200 K=2,NKP1
      KP2=K+2
      KP1=K+1
      KM1=K-1
      KM2=K-2
      DO 200 J=2,NJP1
      JP2=J+2
      JP1=J+1
      JM1=J-1
      JM2=J-2
      DO 200 I=2,NIP1
      IP2=I+2
      IP1=I+1
      IM1=I-1
      IM2=I-2


C ****    FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CONTROL
C         VOLUME FOR SIG12

C     IF (J.EQ.2) GOTO 300
      DXN=XL(I,J  ,K,1,0)
      DXS=XL(I,JM1,K,1,0)
      DYE=YL(I  ,J,K,2,0)
      DYW=YL(IM1,J,K,2,0)
      DXI=XL(I  ,J,K,1,2)
      DYJ=YL(I  ,J,K,2,1)

      DYN=YL(I,J  ,K,1,0)
      DYS=YL(I,JM1,K,1,0)
      DXE=XL(I  ,J,K,2,0)
      DXW=XL(IM1,J,K,2,0)
```

```
      UBAR=SILIN(U(I,J,K),U(I,JM1,K),DYN,DYS)
      VBAR=SILIN(V(I,J,K),V(IM1,J,K),DXE,DXW)

      VIS12=BILIN(VIS(I  ,J,K),VIS(I  ,JM1,K),DYN,DYS,
     &            VIS(IM1,J,K),VIS(IM1,JM1,K),DYN,DYS, DXE,DXW)

      SIG12(I,J,K)=             VIS12*((V(I,J,K)-V(IM1,J,K))/DXI
     &                          -VBAR*(DYE-DYW)/(DXI*DYJ))
      SIG12(I,J,K)=SIG12(I,J,K)+VIS12*((U(I,J,K)-U(I,JM1,K))/DYJ
     &                          -UBAR*(DXN-DXS)/(DXI*DYJ))
  300 CONTINUE

C ****    FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CONTROL
C         VOLUME FOR SIG13

      DXF=XL(I,J,K  ,1,0)
      DXB=XL(I,J,KM1,1,0)
      DZE=ZL(I  ,J,K,3,0)
      DZW=ZL(IM1,J,K,3,0)
      DXI=XL(I  ,J,K,1,3)
      DZK=ZL(I  ,J,K,3,1)

      DZF=ZL(I,J,K  ,1,0)
      DZB=ZL(I,J,KM1,1,0)
      DXE=XL(I  ,J,K,3,0)
      DXW=XL(IM1,J,K,3,0)

      IF (DZF.EQ.0.0.OR.DZB.EQ.0.0.OR.DZE.EQ.0.0.OR.DZW.EQ.0.0)
     &   WRITE (6,*) I,J,K, DZF,DZB,DZE,DZW
      UBAR=SILIN(U(I,J,K),U(I,J,KM1),DZF,DZB)
      WBAR=SILIN(W(I,J,K),W(IM1,J,K),DXE,DXW)

      VIS13=BILIN(VIS(I  ,J,K),VIS(I  ,J,KM1),DZF,DZB,
     &            VIS(IM1,J,K),VIS(IM1,J,KM1),DZF,DZB, DXE,DXW)

      SIG13(I,J,K)=             VIS13*((W(I,J,K)-W(IM1,J,K))/DXI
     &                          -WBAR*(DZE-DZW)/(DXI*DZK))
      SIG13(I,J,K)=SIG13(I,J,K)+VIS13*((U(I,J,K)-U(I,J,KM1))/DZK
     &                          -UBAR*(DXF-DXB)/(DXI*DZK))


C ****    FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CONTROL
C         VOLUME FOR SIG23

      DZN=ZL(I,J  ,K,3,0)
      DZS=ZL(I,JM1,K,3,0)
      DYF=YL(I,J,K  ,2,0)
      DYB=YL(I,J,KM1,2,0)
      DZK=ZL(I,J,K,3,2)
      DYJ=YL(I,J,K,2,3)

      DYN=YL(I,J  ,K,3,0)
      DYS=YL(I,JM1,K,3,0)
      DZF=ZL(I,J,K  ,2,0)
      DZB=ZL(I,J,KM1,2,0)
```

```
      WBAR=SILIN(W(I,J,K),W(I,JM1,K),DYN,DYS)
      VBAR=SILIN(V(I,J,K),V(I,J,KM1),DZF,DZB)

      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,VOLOT,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,COND0,VIS0,RHO0,HR,TR,TA,OTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      COMMON/BL34/ HEIGHT(22,16,32),REQ(22,16,32),
     &      SMP(22,16,32),SMPP(22,16,32),PP(22,16,32),
     &    OU(22,16,32),OV(22,16,32),OW(22,16,32)
      COMMON/BL37/ VIS(22,16,32),COND(22,16,32),NOO(22,16,32),RWALL(579)
     &       ,CPM(22,16,32),HSZ(3,2),NHSZ(22,16,32),RESORM(93)
      COMMON/BL39/ALEW,PCURVE,CONSRA,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 =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

      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)
           DPDT  =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**
     &       3-.115621E-11*XTIME**4+REQ(10,9,16)
      DPDT=.81264E-3-.22604E-5*XTIME*2+.27262E-8*XTIME**
     &       2*3.0-.115621E-11*XTIME**3*4
         ENDIF
      IF ( LL .EQ. 1) THEN
      QQ=1.0E8*DPDT
      Q=QQ*3.4134/60./60.
   65 CONTINUE
      Q=Q*QCORRT-QR

      ELSE
C    THIS USES A CURVE FIT THROUGH THE BURNRATE DATA GIVEN BY NRL
        QCORRT=0.0
        QCORR=0.0
        ITEST = 0
        BURNR1= 5.4576748 +0.18815346*XTIME-.20153996E-03*XTIME**2
        BURNR2= -1.3116787 + .33158595*XTIME-.7342952E-03*XTIME**2
     &         +.50945510E-06*XTIME**3
        IF (XTIME .LT. 100) THEN
          BURNR= BURNR2 + 1.3117-.013117*XTIME
        ELSE
          BURNR = BURNR2
        ENDIF
        IF(XTIME .LE. 300) GO TO 60
        IF(BURNR2 .LT. BURNR1) THEN
            BURNR = (BURNR1 + BURNR2) / 2
            GO TO 60
        ELSE
            IF ( XTIME .LT. 600.0) GO TO 60
            IF (ITEST .EQ. 0) THEN
               BURNR3 = BURNR2
               ITEST = 1
            ENDIF
            BURNR = BURNR3
         ENDIF
   60    Q = BURNR*2.2046*9612./3600.-QR
CC   THIS GIVES Q IN BTU/SEC

      ENDIF
```

```
      Q=59.313+0.7195*XTIME-0.1139E-2*XTIME**2-0.3367E-5*XTIME**3
      Q=Q*3412/3600
      RETURN
      END



C     _________________________________________________________________
***   *****************************************************************
      SUBROUTINE RADHT(T4WALL,VFMXC)
***   *****************************************************************
      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/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,
     &  CP0,PRT,COND0,VIS0,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      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)
      COMMON/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR


      DIMENSION VFMXC(579,579),T4WALL(579)
      DO 4010 K=3,NKM1
      DO 4010 I=2,NI
      II=(K-3)*(NI-1)+I-1
      T4WALL(II)=CONSRA*T(I,NJRA,K)*T(I,NJRA,K)*T(I,NJRA,K)*T(I,NJRA,K)
 4010 CONTINUE

C RADIATION FROM THE FIRE TO THE WALL

      DO 4011 J=3,9
      JJ=561+9-J
      AVT=0.25*(T(16,J,16)+T(17,J,16)+T(16,J,17)+T(17,J,17))
      T4WALL(JJ)=CONSRA*AVT*AVT*AVT*AVT
 4011 CONTINUE
C
      DO 4012 J=3,14
      JJ=568+J-3
      AVT=0.25*(T(6,J,16)+T(7,J,16)+T(6,J,17)+T(7,J,17))
      T4WALL(JJ)=CONSRA*AVT*AVT*AVT*AVT
 4012 CONTINUE
C
      DO 4020 I=1,579
      RWALL(I)=0.0
      DO 4020 J=1,579
      RWALL(I)=RWALL(I)+VFMXC(I,J)*T4WALL(J)
 4020 CONTINUE
      RETURN
      END



C     _________________________________________________________________
***   *****************************************************************
```

```
      SUBROUTINE GLOBE
***   ******************************************************************
*     THIS SUBROUTINE CALCULATES THE GLOBAL PRESSURE CORRECTION,       *
*     WHEREBY THE PRESSURE MATRIX IS UPDATED.                          *
*     VARIABLES USED ARE:                                              *
*               SUMT        =     SUM OF TEMPERATURES                  *
*               SUMPT       =     SUM OF PRESSURE OVER TEMPERATURE     *
*               SUMPET      =     SUM OF EQUILIBRIUM PRESSURE OVER TEMP*
*               UGRT        =     CONSTANT                             *
*               PCORR       =     PRESSURE CORRECTION                  *
************************************************************************
      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/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,
     & CP0,PRT,COND0,VIS0,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      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/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)

      SUMT=0.
      SUMPT=0.
      SUMPET=0.
      DO 370 I=2,NI
      DO 370 J=2,NJ
      DO 370 K=2,NK
      IF (NOD(I,J,K).EQ.1) GOTO 370
      DXI=XL(I,J,K,0,0,0)
      DYJ=YL(I,J,K,0,0,0)
      DZK=ZL(I,J,K,0,0,0)
      VOL=DXI*DYJ*DZK
      SUMT=SUMT+1./T(I,J,K)*VOL
      SUMPT=SUMPT+P(I,J,K)/T(I,J,K)*VOL
      SUMPET=SUMPET+REQ(I,J,K)*(1./1.0-1./T(I,J,K))*VOL
  370 CONTINUE
      SUMPET=SUMPET/UGRT
      PCORR=(SUMPET-SUMPT)/SUMT
      PCORRN=PCORR

      DO 371 I=1,NIP1
      DO 371 J=1,NJP1
      DO 371 K=1,NKP1
      P(I,J,K)=P(I,J,K)+PCORRN
  371 CONTINUE

      RETURN
      END



C     _________________________________________________________________
***   *****************************************************************
```

```
      SUBROUTINE SOLCON
*** *******************************************************************
      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,
     &  CP0,PRT,COND0,VIS0,RHO0,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/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)

      DO 402 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 405 I=IB,IE-1
      DO 405 J=JB,JE-1
      DO 405 K=KB,KE-1
      COND(I,J,K)=COND0*CONS(N)
      CPM(I,J,K)=CPS(N)
      NOD(I,J,K)=1
      IF (J.EQ.NJ) COND(I,NJP1,K)=COND(I,NJ,K)
      IF (I.EQ.2)  COND(1,J,K)=COND(2,J,K)
      IF (I.EQ.NI) COND(NIP1,J,K)=COND(NI,J,K)
      IF (I.EQ.2.AND.J.EQ.NJ)  COND(1,J+1,K)=COND(2,J,K)
      IF (I.EQ.NI.AND.J.EQ.NJ) COND(NIP1,J+1,K)=COND(NI,J,K)
      IF (J.EQ.NJ) CPM(I,NJP1,K)=CPM(I,NJ,K)
      IF (I.EQ.2)  CPM(1,J,K)=CPM(2,J,K)
      IF (I.EQ.NI) CPM(NIP1,J,K)=CPM(NI,J,K)
      IF (I.EQ.2.AND.J.EQ.NJ)  CPM(1,J+1,K)=CPM(2,J,K)
      IF (I.EQ.NI.AND.J.EQ.NJ) CPM(NIP1,J+1,K)=CPM(NI,J,K)
  405 CONTINUE
  402 CONTINUE
       RETURN
       END


C     _________________________________________________________________
*** *******************************************************************
       SUBROUTINE PTRACK
*** *******************************************************************
      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,COND0,VIS0,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      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/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/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR

```

```
CC ** THE FOLLOWING PRESSURE TEST IS A TEMPORARY MEASURE TO MODIFY THE
CC    HEAT INPUT TO FORCE THE CALCULATED PRESSURE TO AGREE WITH THE
CC    EXPERIMENTAL PRESSURE.  IT WILL BE USED UNTIL ACCURATE HEAT INPUT
CC ** IS RECEIVED.
CC
      PSOUTH=P(10,9,16)*CONST1+REQ(10,9,16)
      PERROR=(PCURVE-PSOUTH)/PCURVE
      QCORR=1.0+PERROR-(PSOUTH-PM1)/PCURVE
      QCORR=1.0+PERROR-(PSOUTH-PM1)/PCURVE+(PSOUTH-PM1)/(PCURVE-PCURM1)*
     &    (PCURVE-PM1)/PCURVE
      QCORRT=QCORRT*QCORR
      PCURM1=PCURVE
      PM1=PSOUTH
C
      RETURN
      END



C     ________________________________________________________________
***   ****************************************************************
      SUBROUTINE TCP
***   ****************************************************************

**********************************************************************
*     THIS SUBROUTINE CALCULATES THE TEMPERATURE AT THE TERMOCOUPLE    *
*     POSITIONS.                                                       *
**********************************************************************
      COMMON/R4/XC(93),YC(93),ZC(93),XS(93),YS(93),ZS(93),
     &          DXXC(93),DYYC(93),DZZC(93),DXXS(93),DYYS(93),DZZS(93)
      COMMON/BL16/ CONST1,CONST2,CONST3,CONST4,CONST6,NT,U0,H,UGRT,BUOY,
     &  CP0,PRT,COND0,VIS0,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      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/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12)


      DO 5100 N=1,NTHCO
      II=NTH(N,1)
      JJ=NTH(N,2)
      KK=NTH(N,3)
      VOL=ABS((XC(II+1)-XC(II))*(YC(JJ+1)-YC(JJ))*(ZC(KK+1)-ZC(KK)))
      TCOUP(N)=0.
      DO 5101 I=II,II+1
      III=II+II+1-I
      DO 5101 J=JJ,JJ+1
      JJJ=JJ+JJ+1-J
      DO 5101 K=KK,KK+1
      KKK=KK+KK+1-K
      WVOL=ABS((XC(I)-CX(N))*(YC(J)-CY(N))*(ZC(K)-CZ(N)))/VOL
      TCOUP(N)=TCOUP(N)+WVOL*T(III,JJJ,KKK)
 5101 CONTINUE
      TCOUP(N)=TCOUP(N)*TR-273.18

 5100 CONTINUE
```

```
      RETURN
      END



C     _______________________________________________________________________
***   ***********************************************************************
      SUBROUTINE OUT(NN)
***   ***********************************************************************
      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,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,COND0,VIS0,RHO0,HR,TR,TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR
      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/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)
      COMMON/BL38/NTHCO,CX(12),CY(12),CZ(12),NTH(12,3),TCOUP(12)
      COMMON/BL39/ALEW,PCURVE,CONSRA,PCURM1,PSOUTH,QCORR,PERROR
      XTIME=TIME*H/U0
      IF( NN .EQ. 1) THEN
C
      QRR=60.*60./3.412/1000.*QR
      WRITE(6,500) XTIME,NTREAL,TIME,ITER,RESORM(ITER),SORSUM,QRR
  500 FORMAT(1X, 'TIME=',F7.3,' S',1X,'NTREAL=',I9,1X,
     & 'TIME=',F7.2,'<0>',1X,'ITER=',I2,1X,'SOURCE=',
     & F9.6,1X,'SORSUM=',F9.6,1X,' QR(KW) = ',F10.4)
C
      QKW = ((60.*60.)/(3.412*1000.))* Q
      PRINT *
      PRINT *, '  PCURVE           PSOUTH                 PERROR              Q
     &CRR             QCORRT              Q(KW) '
      PRINT *, PCURVE,PSOUTH,PERROR,QCORR,QCORRT,QKW
      PRINT *
C
      ELSE IF( NN .EQ. 2 ) THEN
      PRINT *
      PRINT *,'    TEMPERATURES  AT THERMOCOUPLE POSITION IN (C)'
      WRITE (6,*) (TCOUP(N),N=1,NTHCO)
      PRINT *
      PRINT *

      ELSE

      DO 502 L=25,25
```

```
      K=L
      DO 502 M=1,NIP1
      I=M
      WRITE(6,504) I,K
  504 FORMAT(/,2X,'I=',I2,5X,'K=',I2,/,10X,' T NOD',3X,'R(GM/C.C.)',2X,
     & 'U(CM/SEC)',2X,'V(CM/SEC)',2X,'W(CM/SEC)','P (ATM)',5X,'SMP',5X,
     & 'VIS(SEC/CM-CM)',3X,'COND(SEC/CM-CM)',' XSMP',/)
  513 DO 503 J=1,NJP1
C     XTEMP=T(I,J,K)/CONST3-273.16
      XTEMP=T(I,J,K)
C     XR=R(I,J,K)*RHO0/2.2048 *1000.*(0.0328)**3
      XR=R(I,J,K)
C     XU=U(I,J,K)*CONST6
C     XV=V(I,J,K)*CONST6
C     XW=W(I,J,K)*CONST6
C     XP=(P(I,J,K)*CONST1+REQ(I,J,K)*PINT)
      XP=P(I,J,K)
      XU=U(I,J,K)
      XV=V(I,J,K)
      XW=W(I,J,K+1)
CC    XVIS=VIS(I,J,K)*RHO0*CP0*H*U0*1.48814
CC    XCOND=COND(I,J,K)*RHO0*CP0*H*U0*1.48814
      XVIS=VIS(I,J,K)/VIS0
      XCOND=COND(I,J,K)/VIS0
      XSMP=RI(I,J,K)
      DDYY=1./FLOAT(NJM1-2)
      PE =SQRT(U(I,J,K)**2+V(I,J,K)**2+W(I,J,K)**2)*DDYY/COND(I,J,K)
      WRITE(6,511)J,XTEMP,XR,XU,XV,XW,XP,SMP(I,J,K),XVIS,XCOND,XSMP
  511 FORMAT(2X,'J=',I3,2X,F6.3,2X,F6.3,2X,F7.3,2X,F7.3,3X,F7.3,3X
     & ,F12.3,3X,F9.6,2X,F6.2,2X,F6.2,2X,F6.3)
  503 CONTINUE
  502 CONTINUE
      ENDIF
      RETURN
      END
```

LIST OF REFERENCES

1. Emmons, H. W., "Scientific Progress on Fire," *Ann. Rev. Fluid Mech.*, vol. 12, pp. 223–36, 1980.

2. 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.

3. 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.

4. 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.

5. 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.

6. 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.

7. Chan, A. M. C., and S. Banerjee, "Three-Dimensional Numerical Analysis of Transient Natural Convection in Rectangular Enclosure," *J. Heat Transfer*, vol. 101, pp. 114–119, 1979.

8. 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.

9. 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.

10. 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, 1495–1500, 1986.

11. 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.

12. Yang, H. Q., K. T. Yang, and J. R. Lloyd, "Three Dimensional Buoyant Bimodal Flow Transition in Tilted Enclosures," to appear in *Int. J. Heat and Fluid Flow*, 1988.

13. Yang, H. Q., *Laminar Buoyant Flow Transitions in Three-Dimensional Tilted Rectangular Enclosures*, Ph.D. Thesis, University of Notre Dame, South Bend, IN, 1987.

14. Ozoe, 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.

15. 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.

16. Takata, Y., 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.

17. 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.

18. 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.

19. 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.

20. 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.

21. Ozoe, 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.

22. Baum, H. R., and R. G. Rehm, "Calculations of Three Dimensional Buoyant Plumes in Enclosures," *Combustion Science and Technology*, vol. 40, pp. 55–77, 1984.

23. 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.

24. 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.

25. 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.

26. 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.

27. 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.

28. Alexander, J. I., H. J. St. Aubin, J. P. Stone, T. T. Street, and F. W. Williams, "Large-Scale Pressurizable Fire Test Facility—Fire I," NRL Report 864, Naval Research Laboratory, Washington, DC, December 1982.

29. 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.

30. 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.

31. Eringen, A. C., *Mechanics of Continua*, John Wiley & Sons, Inc., New York, NY, 1967.

32. 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.

33. Sparrow, E. M., and R. D. Cess, *Radiation Heat Transfer*, Hemisphere Publishing Corporation, Washington, DC, 1978.

34. 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.

35. Patankar, S. V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Company, New York, NY, 1980.

36. 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 Pub. Corp., Washington, DC, pp. 211–226, 1983.

37. Satoh, K., J. R. Lloyd, K. T. Yang, and A. M. Kanury, "A Numerical Finite Difference Study of the Oscillatory Behavior of Vertically Vented Compartments," *Numerical Properties and Methodologies in Heat Transfer*, ed. T. M. Shih, Hemisphere Pub. Corp., Washington, DC, pp. 517–528, 1983.

38. Raycraft, J., M. D. Kelleher, H. Q. Yang, and K. T. Yang, Naval Postgraduate Schol, Monterey, CA, NPS Report NPS-69-88-008, "Fire Spread in a Three-Dimensional Pressure Vessel with Radiation Exchange and Wall Heat Losses," August 1, 1988.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Commandant (G-PTE-1) U.S. Coast Guard Washington, DC 20590	2
4.	Dr. Patricia A. Tatem Code 6183 Naval Research Laboratory Washington, DC 20375	2
5.	Professor K. T. Yang Department of Aerospace and Mechanical Engineering University of Notre Dame Notre Dame, IN 46556	3
6.	Professor M.D. Kelleher, Code 69Kk Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 93943-5004	2
7.	Chairman, Department of Mechanical Engineering Code 69 Naval Postgraduate School Monterey, CA 93943-5004	2
8.	LCDR Richard Houck Resident Inspector Office 15708 23rd Avenue Seattle, WA 98166	2

9. CAPT Raymond Michelini 1
Code 56Y
Naval Sea Systems Command
Department of the Navy
Washington, DC 20362

10. Mr. James Gagorik 2
Code 211, ONT
Balston Towers #1
800 N. Quincy
Arlington, VA 22217

11. Dr. Robert Richards 3
U.S. Coast Guard
Research and Development Center
Avery Point
Groton, CT 06340-6096