

## Construction of a low-Reynolds-number $k$ - $\epsilon$ model

by

**Qingyan Chen**

*Swiss Federal Institute of Technology Zurich  
Energy Systems Laboratory  
ETH-Zentrum, CH-8092 Zurich (Switzerland)*

Date submitted : May, 1989  
Computer : CDC Cyber 180/855  
Operating system : NOS/VE 1.4.1  
PHOENICS version : 1.4.5  
Date of issue of the code : 11 February, 1988

### ABSTRACT

A low-Reynolds-number  $k$ - $\epsilon$  model of turbulence has been constructed in PHOENICS-84. For the predictions of buoyant-affected flow in buildings, the Boussinesq approximation is used. Therefore, the buoyancy production terms in the  $k$  and  $\epsilon$  equations should be included in the computer code.

The model has been applied for the predictions of natural convection in a large-scale air-filled cavity. The computed results are in rather good agreement with the measurements. It has also been found that the setting of boundary conditions has a very significant influence on the computed results.

## **CONTENTS**

### **1. INTRODUCTION**

- 1.1 The Problem Considered**
- 1.2 Objectives of the Work**

### **2. OUTLINE OF THE LOW-REYNOLDS-NUMBER $k-\epsilon$ MODEL**

- 2.1 The Lam and Bremhorst's  $k-\epsilon$  Model**
- 2.2 The Buoyancy Production terms in the  $k$  and  $\epsilon$  Equations**
- 2.3 Boundary Conditions**

### **3. PHOENICS SETTINGS**

- 3.1 SATELLITE**
- 3.2 GROUND**

### **4. PRESENTATION OF THE RESULTS**

- 4.1 Typical Output**
- 4.2 Convergence**
- 4.2 Computer Storage and Time**

### **5. DISCUSSION OF THE RESULTS**

### **6. CONCLUSIONS**

### **7. ACKNOWLEDGMENTS**

### **8. REFERENCES**

### **9. NON-STANDARD NOMENCLATURE**

### **APPENDIX A**

- A.1 The Q1 file**
- A.2 The GROUND file**

## 1. INTRODUCTION

### 1.1 The Problem Considered

There are numerous practical flow problems related to building, which can be solved profitably by numerical techniques. The  $k$ - $\epsilon$  model of turbulence (Launder and Spalding, 1974) is the most widely used model for practical flow applications in buildings. The set of model equations is suitable for high-Reynolds-number flow. For wall flow, where local Reynolds numbers are considerably low, the equations are normally used in conjunction with empirical wall function formulae. The success of this method depends on the "universality" of the turbulent structure near the wall. When disagreements are found between measurements and predictions, it is difficult to judge whether the weakness of the method lies in the basic model equations or in the wall function formulae.

The logarithmic wall function formulae (Launder and Spalding 1974) are derived for fully developed, zero-pressure-gradient boundary layers. They are not suitable for the boundary layers of air flow in a room. Hence, the wall function can hardly present good results for the air flow near-walls in a room.

On the other hand, air flows in rooms in many cases include natural or mixed convection and the overall turbulence Reynolds numbers ( $R_t$ ) are rather small. The high-Reynolds-number  $k$ - $\epsilon$  model of turbulence with the logarithmic wall functions may not be suitable for both near the wall and far away from it. Therefore, it is necessary to apply an appropriate low-Reynolds-number model of turbulence in the computations of indoor air flow patterns.

In addition to the low-Reynolds-number property, buoyancy is a main factor in natural and mixed convection flow problems of buildings. It is also necessary to study the influence of the buoyancy production terms in the equations of the kinetic energy of turbulence ( $k$ ) and the dissipation rate of turbulence energy ( $\epsilon$ ).

## 1.2 Objectives of the Work

The objectives of the work are:

- to construct a low-Reynolds-number  $k$ - $\epsilon$  model with the buoyancy production terms in the  $k$  and  $\epsilon$  equations in the PHOENICS-84 code;
- to apply the model for the prediction of natural convection in a large-scale air-filled cavity and to compare the computed results with experimental data.

## 2. OUTLINE OF THE LOW-REYNOLDS-NUMBER MODEL

Over the past few years, many suggestions have been made for the extension of turbulence closure models to enable their use at low-Reynolds-numbers and to describe the flow close to a solid wall. Most of the extended models incorporate either a wall damping effect or a direct effect of molecular viscosity, or both, on the empirical constants and functions in the turbulence-transport equations devised originally for the high-Reynolds-number, fully turbulent flows remote from walls. In the absence of reliable turbulence data in the immediate vicinity of a wall or at low-Reynolds-numbers, these modifications have been based largely upon comparisons between calculations and experiments in terms of global parameters. According to the review on turbulence models for near-wall and low-Reynolds-number flows by Patel et al. (1985), the model developed by Lam and Bremhorst (1981) is one of them. It is based on the  $k$ - $\epsilon$  model and modifies the direct effect of molecular viscosity. An advantage of the model is that it employs a transport equation for the total dissipation rate, and that the form of the  $k$  and  $\epsilon$  source terms is simpler than found with other low-Reynolds-number versions. This model gave rather good results as reported by Patel et al. (1985). Therefore, it is selected in the present research.

## 2.1 The Lam and Bremhorst's Model

In the model of Lam and Bremhorst, the transport equations of the kinetic energy ( $k$ ) and the dissipation rate of turbulence energy ( $\epsilon$ ) are determined from:

$$\frac{Dk}{Dt} = \frac{\partial}{\partial x_j} \left[ \left( \frac{v_t}{\sigma_k} + v_l \right) \frac{\partial k}{\partial x_j} \right] + v_t \left( \frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i} \right) \frac{\partial V_i}{\partial x_j} - \epsilon \quad (1)$$

$$\frac{D\epsilon}{Dt} = \frac{\partial}{\partial x_j} \left[ \left( \frac{v_t}{\sigma_\epsilon} + v_l \right) \frac{\partial \epsilon}{\partial x_j} \right] + C_1 f_1 v_t \frac{\epsilon}{k} \left( \frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i} \right) \frac{\partial V_i}{\partial x_j} - C_2 f_2 \frac{\epsilon^2}{k} \quad (2)$$

The time-averaged flow field can be determined through the eddy viscosity given by:

$$v_t = C_\mu f_\mu k^2 / \epsilon \quad (3)$$

In fact, these equations are a general form of those given by Launder and Spalding (1974) in which the functions  $f_1$ ,  $f_2$ , and  $f_\mu$  are all assumed to be identically unity. This assumption cannot be valid within a laminar sublayer or in low-Reynolds-number flows. The functions  $f_1$ ,  $f_2$ , and  $f_\mu$  are given by the following equations:

$$f_\mu = (1 - e^{-A_\mu R_k})^2 \left( 1 + \frac{A_t}{R_t} \right) \quad (4)$$

$$f_1 = 1 + \left( \frac{A_{C1}}{f_\mu} \right)^3 \quad (5)$$

$$f_2 = 1 - e^{-R_t^2} \quad (6)$$

where the turbulence model constants are  $A_\mu = 0.0165$ ,  $A_t = 20.5$  and  $A_{C1} = 0.05$ , and  $R_k = \frac{k^{1/2} y}{v_l}$  and  $R_t = \frac{k^2}{v_l \epsilon}$  are turbulence

Reynolds numbers. The  $y$  in  $R_k$  is the distance from the cell centre to the nearest wall in the computer code.

## 2.2 The Buoyancy Production Terms in the $k$ Equations

Since the temperature difference in room air is relatively small, it is a common practice to use Boussinesq approximation. This approximation takes air density as constant and considers buoyancy influence on air movement in the momentum equation via the following term:

$$-\beta(T - T_0)g_i \quad (7)$$

Since the  $k$  and  $\epsilon$  equations are derived from the momentum equation, the buoyancy term in the momentum equation is then changed into buoyancy production terms in the  $k$  and  $\epsilon$  equations. The buoyancy production term for the  $k$  equation is approximated as:

$$S_k = \beta \frac{v_t}{\sigma_H} \frac{\partial(T - T_0)}{\partial x_i} g_i \quad (8)$$

$S_k$  is an additional source term on the right side of Equations (1). The source term for the  $\epsilon$  equation is:

$$S_\epsilon = C_3 \frac{\epsilon}{k} S_k \quad (9)$$

where  $C_3=1.44$  (Nielsen et al. 1979).

## 2.3 Boundary Conditions

When a low-Reynolds-number  $k$ - $\epsilon$  model is applied, no wall function formulae are required because the model is valid for the whole flow domain. However, the boundary conditions for  $k$  and  $\epsilon$  must be specified. We have set several boundary conditions

including those given by Lam and Bremhorst (1981) for  $k$  and  $\epsilon$ , but only the following two methods generate converged results.

### Method 1

According to the recommendation by Parry (1989), a zero value is assigned at the wall for the  $k$  equation and

$$\epsilon = \frac{2 v_1 k}{y^2} \quad (10)$$

for the  $\epsilon$  equation for the grid cell close to the wall. This boundary condition is equivalent to assuming a zero-gradient condition. This is because an expression for the near-wall variation of  $k$  may be derived by expanding the fluctuating velocities in a Taylor series about the wall and time averaging. The resulting expression for  $k$  may then be differentiated twice to provide an analytical relationship for

$$\epsilon = v_1 \frac{\partial^2 k}{\partial y^2} \quad (11)$$

If a zero-gradient condition is assumed for  $\epsilon$ , it can then be shown that by retaining only the first term in the analytical expression for  $k$ , there results Equation (10) for the grid point close to the wall.

### Method 2

The wall function formulae are still used in this method but the distance between the wall and the first grid can be very small. This indicates that the first grid points will be located inside the viscous layer. In this circumstance, the second grid points may still be in the inner region of the boundary layer and the low-Reynolds-number model will work well for those points. In the inside viscous layer, the wall function formulae present linear velocity and temperature distributions and they can be accepted for air flow in a room. This is different from the application of the high-

Reynolds-number model with the wall function formulae in which the first grid must be sufficiently remote from the wall.

### 3. PHOENICS SETTINGS

The PHOENICS-84 computer code developed by Rosten and Spalding (1987) was extended to include the turbulence model formulations described above.

The methodology for performing the numerical calculations uses under-relaxation and false time step factors for obtaining a convergent solution. The calculations were carried out using an up-wind differencing scheme and "staggered grids". A non-uniform grid is used to produce a very fine grid near the boundary and a coarse grid in the central region of the flow field. The grid for the two-dimensional case as shown in Figure 1 is 50x50.

The validation of the low-Reynolds-number  $k-\epsilon$  model for the natural convection in the large-scale air-filled cavity is carried out by the application of the experimental results from Cheesewright et al. (1986). The cavity was designed with a height of 2.5 m and a width of 0.5 m as shown in Figure 1. The top wall and the bottom wall were insulated to be adiabatic. The temperature difference between the hot wall and the cold wall was 45.8 K. The cold wall was cooled by the ambient air of the laboratory and its surface temperature is assumed to be 20.0°C. The corresponding Rayleigh number is  $5 \times 10^{10}$  and is based on the cavity height. The air velocities and their fluctuations were measured by a laser-Doppler anemometer system.

The problem is specified by a Q1 file and the list of which is given in Appendix A.1. The low-Reynolds-number  $k-\epsilon$  model with the buoyancy production terms in the  $k$  and  $\epsilon$  equations is built in subroutine GROUND with some auxiliary subroutines as detailed in Appendix A.2. However, they will be briefly explained in the following subsections.



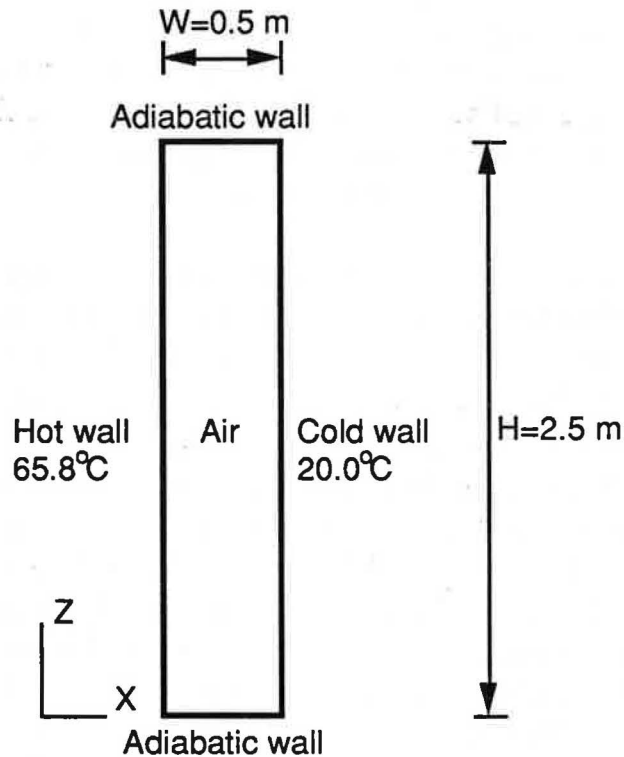


Figure 1. The sketch of the large scale cavity

### 3.1 SATELLITE

#### The Q1 file

GROUP 1. Run title and other preliminaries

This group provides the run title and the problem concerned. Non-standard variables of input data are also defined in this group.

GROUP 3. X-direction grid specification

50 grid cells are used for the top or bottom wall of the cavity. A non-uniform grid is used to produce a very fine grid near the boundary and a coarse grid in the central region of the flow field.

GROUP 5. Z-direction grid specification

This group is the same as Group 3 but for the vertical hot or cold wall. In addition, the dimension is different.

GROUP 7. Variables stored, solved & named

The variables solved are pressure (P1), the velocity components in x- and z-directions (U1 and W1) and the enthalpy (H1) as well as the turbulence energy (KE) and dissipation rate of turbulence energy (EP) which are given in the comment block. The pressure is solved by the whole-field method.

The comment block presents the corresponding statements as those for TURMOD(KEMODL) as shown in the manual pp. 2.42 (Rosten and Spalding 1987). This block is for the low-Reynolds-number k- $\epsilon$  model. The  $f_\mu$ ,  $f_1$ ,  $f_2$  and  $R_t$  are stored in C11, C12, C13 and C14 respectively. These functions and the turbulent kinematic viscosity (ENUT) are calculated in Group 9, Section 5 of subroutine GROUND. The values of the laminar kinematic viscosity (ENUL) is set in Group 9, Section 6 of subroutine GROUND. RSG3, RSG4, RSG5 are the overall length in x-, y- and z- directions which are used for the calculation of the smallest distance from a cell centre to the walls. The low-Reynolds-number model is activated by the PATCH and COVAL statements. The type name is PHASEM and the coefficients and values in the COVAL statements must be GRND10. GENT=T activates the the inclusion of all spatial derivatives of velocity and the OUTPUT statements specify the field printout of  $f_\mu$ ,  $f_1$ ,  $f_2$  and  $R_t$ .

GROUP 8. Terms (in differential equations) & devices

DIFCUT=0.0 activates the "upwind-interpolation" scheme.

GROUP 9. Properties of the medium (or media)

A constant density is assigned for RHO1.

GROUP 11. Initialization of variable or porosity fields

Uniform initial values are employed for the variables solved.

GROUP 13. Boundary conditions and special sources

This group provides two different kinds of boundary conditions as discussed in the preceding section. The one with method 1 is activated.

Method 1 uses the boundary conditions recommended by Parry (1989). For EP, the patch name must begin with "LBB..." and the coefficient is FIXVAL. The patch type is not very important because FIXVAL is a very large value. However, a PHASEM will speed up the

convergence because of the assumption of a zero-gradient boundary condition.

Method 2 uses the standard wall functions. If they are applied for the low-Reynolds-number  $k$ - $\epsilon$  model, the distance of the first grid should be as small as possible so that the first grid will be located in the viscous sublayer.

The fixed pressure block gives a reference value for the pressure field. This is necessary for natural convection problems.

In the present study, Boussinesq approximation is employed. GRND3 selects the buoyancy-force option as indicated in the manual pp. 1.9 (Rosten and Spalding 1987). The buoyancy source in the momentum equation is activated by

```
PATCH(BUOY,PHASEM,1,NX,1,NY,1,NZ,1,1)
COVAL(BUOY,W1,FIXFLU,GRND3)
```

The buoyancy production terms in the  $k$  and  $\epsilon$  equations are activated in the final block of this group.

#### GROUP 15. Termination of sweeps

The sweep number is 2000.

#### GROUP 17. Under-relaxation devices

The linear model of under-relaxation is only applied for P1 with a value of 0.5. The false-time-step model is used for the rest variables solved. The values are set to 0.3 except 0.1 for W1.

#### GROUP 18. Limits on variables or increments to them

This group sets the minimum and maximum values allowed to the variable solved.

#### GROUP 19. Data communicated by satellite to GROUND

In addition to the RSG3, RSG4 and RSG5 in Group 7, RSG1 and RSG2 are used for Boussinesq approximation as shown in the manual pp. 1.9 (Rosten and Spalding 1987). It should be noted that the negative sign for RSG2 is necessary to give a correct buoyancy-force. Perhaps there is a mistake in the PHOENICS code.

RG(1) is the volume expansivity coefficient and RG(2) the constant-pressure specific heat. They are used for the computation of the buoyancy production terms in the  $k$  and  $\epsilon$  equations in GROUND.

#### GROUP 22. Spot-value print-out

The spot-value is chosen at the first grid of the middle-height of the hot wall for monitoring the convergence solution.

#### GROUP 23. Field print-out and plot control

This group sets the printout format.

### 3.2 GROUND

#### Subroutine GROUND

##### Preliminaries

At first a number of the arrays are declared in the section of user's dimension. Those arrays are employed for storing the coordinate values of cell centres and for temporary storage of the buoyancy production source in the  $k$  equation.

In Group 1, storages are provided for the coordinate values of cell centres in  $x$ -,  $y$ - and  $z$ -directions. The special EARTH arrays, EASP5, EASP6, EASP7, EASP8, EASP9 and EASP10 are also activated. EASP8, EASP9 and EASP10 are always activated in GREX2 so that no MAKE statements are required.

##### The Low-Reynolds-number $k$ - $\epsilon$ model

For the low-Reynolds-number  $k$ - $\epsilon$  model,  $f_\mu$ ,  $f_1$ ,  $f_2$  and  $v_t$  are calculated and stored in Section 5 of Group 9.  $v_1$  is stored in Section 6 of Group 9. The auxiliary subroutine GETCOOR is used to get and store the coordinate values of the cell centres. Then the source terms of  $k$  and  $\epsilon$  for the low-Reynolds-number  $k$ - $\epsilon$  model are activated by the following statements in Group 7 of the Q1 file:

```
PATCH(LOWRE, PHASEM, 1, NX, 1, NY, 1, NZ, 1, LSTEP) .
COVAL(LOWRE, KE, GRND10, GRND10)
```

**COVAL(LOWRE, EP, GRND10, GRND10)**

The coefficients for  $k$  and  $\epsilon$  are set in Section 11 of Group 13 and the values are in Section 22 of Group 13. The setting of the coefficients and values is done in a similar way as that for the high-Reynolds-number model in subroutine GXKESO.

The buoyancy production terms for the  $k$  and  $\epsilon$  equations

The buoyancy production terms of the  $k$  and  $\epsilon$  equations are activated by the following statements in Group 13 of the Q1 file:

```
PATCH(KEEPBUOY, PHASEM, 1, NX, 1, NY, 1, NZ, 1, LSTEP)
COVAL(KEEPBUOY, KE, GRND, GRND)
COVAL(KEEPBOUY, EP, GRND, GRND)
```

In Section 1 of Group 13, the coefficient for  $k$  is set to  $(1.0 \times 10^{-11} v_t)$  and that for  $\epsilon$  is to  $(1.0 \times 10^{-11} \frac{k}{\epsilon})$ .  $(1.0 \times 10^{11} S_k/v_t)$  is given as the value for  $k$  and  $(1.0 \times 10^{11} C_3 S_k)$  for  $\epsilon$ . The  $S_k/v_t$  is calculated by subroutine BUOYKE and stored in the F array through subroutine VARTOF. Three function subroutines, DPHIDY, DPHIDZ and DPHIDX, are called by BUOYKE to calculate  $\partial H/\partial x$ ,  $\partial H/\partial y$  and  $\partial H/\partial z$ .

The boundary conditions

The boundary conditions for method 1 are set in Section 20 of Group 13 where value equals GRND8 which sets  $\epsilon$  to be  $\frac{2 v_1 k}{y^2}$ .

#### **4. PRESENTATION OF THE RESULTS**

The computations have been performed for the two kinds of boundary conditions. As an example, only the typical results by method 2 are presented in this section.

#### 4.1 Typical Output

FLOW FIELD AT ITHYD= 1, ISWEEP=2000, ISTEP= 1

XZPR IY= 1

... ..

FIELD VALUES OF W1

... ..

IX= 25	2.865E-03	9.407E-03	1.372E-02	5.134E-04	-1.023E-03
IX= 24	-9.152E-05	2.837E-03	6.952E-03	-2.349E-03	-3.086E-03
IX= 23	-7.877E-04	-8.048E-04	2.634E-03	-7.394E-04	-6.125E-03
IX= 22	2.198E-04	-1.378E-03	2.474E-04	1.947E-03	-3.404E-03
IX= 21	4.750E-04	-3.836E-04	-4.097E-04	7.115E-03	2.279E-02
IX= 20	4.058E-04	-1.251E-04	-3.058E-04	2.053E-02	6.597E-02
IX= 19	3.306E-04	7.331E-06	-1.731E-04	2.746E-02	7.783E-02
IX= 18	2.380E-04	1.218E-04	5.477E-05	3.597E-02	9.082E-02
IX= 17	1.565E-04	2.253E-04	4.140E-04	4.627E-02	1.050E-01
IX= 16	1.623E-04	3.699E-04	9.719E-04	5.861E-02	1.204E-01
IX= 15	4.634E-04	7.734E-04	1.868E-03	7.340E-02	1.372E-01
IX= 14	1.592E-03	2.104E-03	3.441E-03	9.135E-02	1.556E-01
IX= 13	4.800E-03	6.052E-03	6.672E-03	1.137E-01	1.757E-01
IX= 12	1.279E-02	1.639E-02	1.528E-02	1.428E-01	1.981E-01
IX= 11	3.086E-02	4.043E-02	4.389E-02	1.821E-01	2.234E-01
IX= 10	5.700E-02	7.534E-02	9.752E-02	2.190E-01	2.439E-01
IX= 9	8.370E-02	1.099E-01	1.470E-01	2.439E-01	2.570E-01
IX= 8	1.203E-01	1.562E-01	2.084E-01	2.728E-01	2.715E-01
IX= 7	1.689E-01	2.156E-01	2.797E-01	3.067E-01	2.878E-01
IX= 6	2.305E-01	2.870E-01	3.567E-01	3.471E-01	3.071E-01
IX= 5	3.027E-01	3.642E-01	4.309E-01	3.963E-01	3.310E-01
IX= 4	3.748E-01	4.316E-01	4.864E-01	4.538E-01	3.629E-01
IX= 3	4.195E-01	4.591E-01	4.949E-01	4.791E-01	3.930E-01
IX= 2	3.875E-01	3.996E-01	4.150E-01	4.031E-01	3.461E-01
IX= 1	2.098E-01	2.005E-01	2.020E-01	1.940E-01	1.681E-01
IZ=	21	22	23	24	25

#### 4.2 Convergence

\*\*\*\*\*

SPOT VALUES VS. SWEEP (/ITHYD IF PARAB)

IXMON= 1 IYMON= .1 IZMON= 25

## TABULATION OF ABSCISSA AND ORDINATES...

ISWP	P1	U1	W1	KE	EP
4.000E+02	3.173E+00	2.846E-04	1.710E-01	9.645E-03	3.578E-01
8.000E+02	3.147E+00	2.873E-04	1.685E-01	9.503E-03	3.499E-01
1.200E+03	3.146E+00	2.873E-04	1.682E-01	9.486E-03	3.489E-01
1.600E+03	3.146E+00	2.872E-04	1.681E-01	9.483E-03	3.488E-01
2.000E+03	3.146E+00	2.872E-04	1.681E-01	9.483E-03	3.488E-01
ISWP	H1	C11	C12	C13	C14
4.000E+02	6.387E+04	1.947E-02	1.794E+01	1.000E+00	1.537E+01
8.000E+02	6.387E+04	1.928E-02	1.845E+01	1.000E+00	1.525E+01
1.200E+03	6.388E+04	1.925E-02	1.852E+01	1.000E+00	1.524E+01
1.600E+03	6.388E+04	1.925E-02	1.853E+01	1.000E+00	1.524E+01
2.000E+03	6.388E+04	1.925E-02	1.853E+01	1.000E+00	1.524E+01

... ..

\*\*\*\*\*

## RESIDUALS VS. SWEEP (/ITHYD IF PARAB)

## TABULATION OF ABSCISSA AND ORDINATES...

ISWP	P1	U1	W1	KE	EP
4.000E+02	4.586E+02	2.632E+02	1.279E+03	1.706E+03	2.113E+01
8.000E+02	6.127E+01	1.101E+01	8.751E+01	1.719E+03	9.451E-01
1.200E+03	4.965E+00	8.084E-01	8.221E+00	1.735E+03	8.063E-02
1.600E+03	4.837E-01	8.146E-02	9.664E-01	1.737E+03	8.620E-03
2.000E+03	6.080E-02	1.119E-02	1.475E-01	1.737E+03	1.275E-03
ISWP	H1				
4.000E+02	4.615E+07				
8.000E+02	4.406E+06				
1.200E+03	4.352E+05				
1.600E+03	5.345E+04				
2.000E+03	8.405E+03				

The solution is very converged. Although the absolute residuals are considerable large at sweep number of 800, the monitored spot values are the same as those at sweep number of 2000.

## 4.2 Computer Storage and Time

The storage is 48893. A 2000 sweeps run takes about 10000 seconds CPU on the CDC Cyber 180/855 computer.

## 5. DISCUSSION OF THE RESULTS

Figure 2 shows the comparison between the measured and computed velocity profiles of vertical component ( $w_1$ ) at the middle height ( $z=1.25$  m). The computed results by Method 2 are in rather good agreement with the measurements. But the results obtained with Method 1 is not acceptable although it is a converged solution. The computed velocity profile by the standard built-in high-Reynolds-number model is also illustrated for

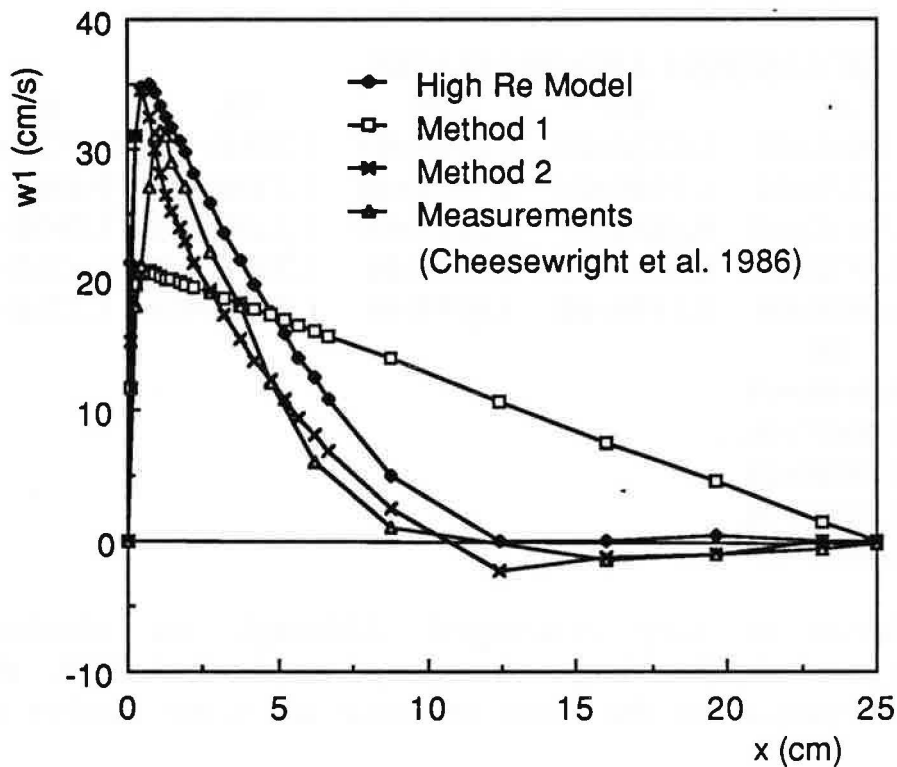


Figure 2. Profiles of velocity component  $w_1$  in the mid-height



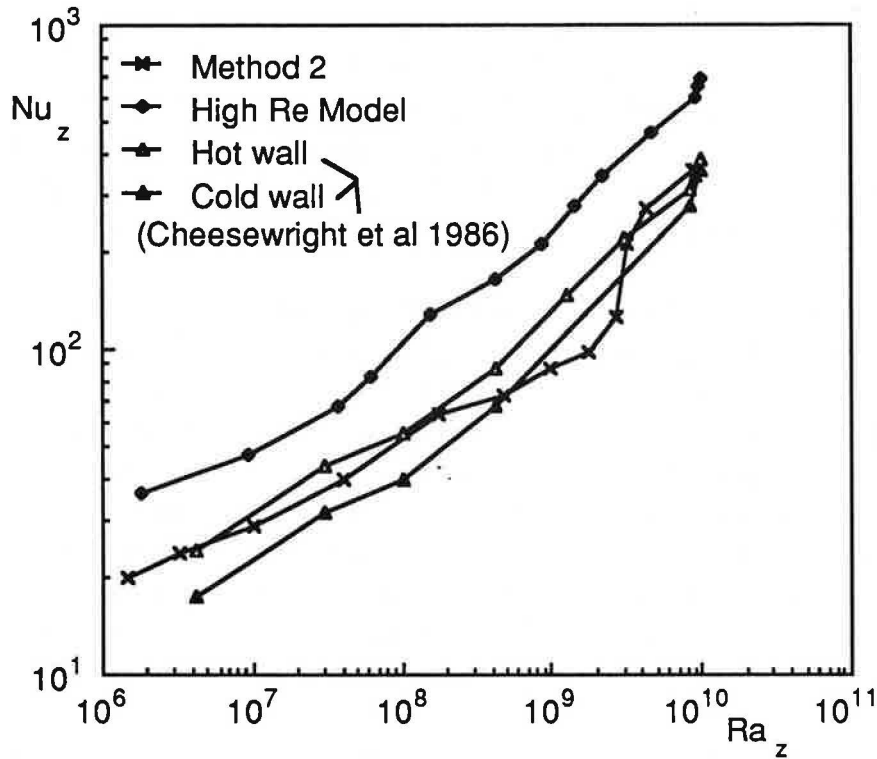


Figure 3. Local heat transfer data

comparison. In the computation with the high-Reynolds-number model, the grid distribution is the same as that for the low-Reynolds-number model. It is clear that Method 2 presents a slight better result for this case. Chen et al. (1990) applied this model for a water filled cavity and found that the low-Reynolds-number model is much better than the high-Reynolds-number model.

The local heat transfer data of the computations and the experiment are illustrated in Figure 3. The local Rayleigh number and the local Nusselt number are defined as:

$$Ra_z = \frac{g\beta(T_h - T_o)z^3}{\nu\alpha} \quad (12)$$

$$Nu_z = - \frac{z (\partial T / \partial x)_{x=0}}{T_h - T_o} \quad (13)$$

where  $T_h - T_o$  is the local wall to core temperature difference. The computed results by method 2 are in good agreement with the data from the experiment. Because of insufficient thermal insulation of the top wall, asymmetric data for the heat transfer through the hot and cold walls of the cavity are found. Therefore in Figure 3 the experimental curves for both the walls are given.

The values of Nusselt number computed by the high-Reynolds-number model are too high. Chen (1988) reported that the high-Reynolds-number model with the wall functions presents a too-low heat flux if the first grid is located outside the viscous sublayer of boundary. In this study, a too-high heat flux has been obtained when the first grid is in the viscous sublayer. This implies that the convective heat transfer computed by the high-Reynolds-number model depends on the first grid location. Therefore, this model is not suitable for natural convection in a cavity from the viewpoint of the heat transfer in boundary.

In the experiment (Cheesewright et al. 1986), the velocity fluctuations at the middle height of the cavity are provided. However, no detailed information is given on the definition of the velocity fluctuations. If we assume that  $k$  is the square of the velocity fluctuation, a comparison of  $k$  profiles in the middle height of the cavity between the computations and the measurements can be done as illustrated in Figure 4. The computed results with the buoyancy production terms in the  $k$  and  $\epsilon$  equations is in better agreement with the measurements in the mid-height section but with a little too high peak value in near wall region.

## 6. CONCLUSIONS

The low-Reynolds-number  $k$ - $\epsilon$  model of turbulence developed by Lam and Bremhorst (1981) is constructed in PHOENICS-84 together with the buoyancy production terms in the  $k$  and  $\epsilon$  equations.

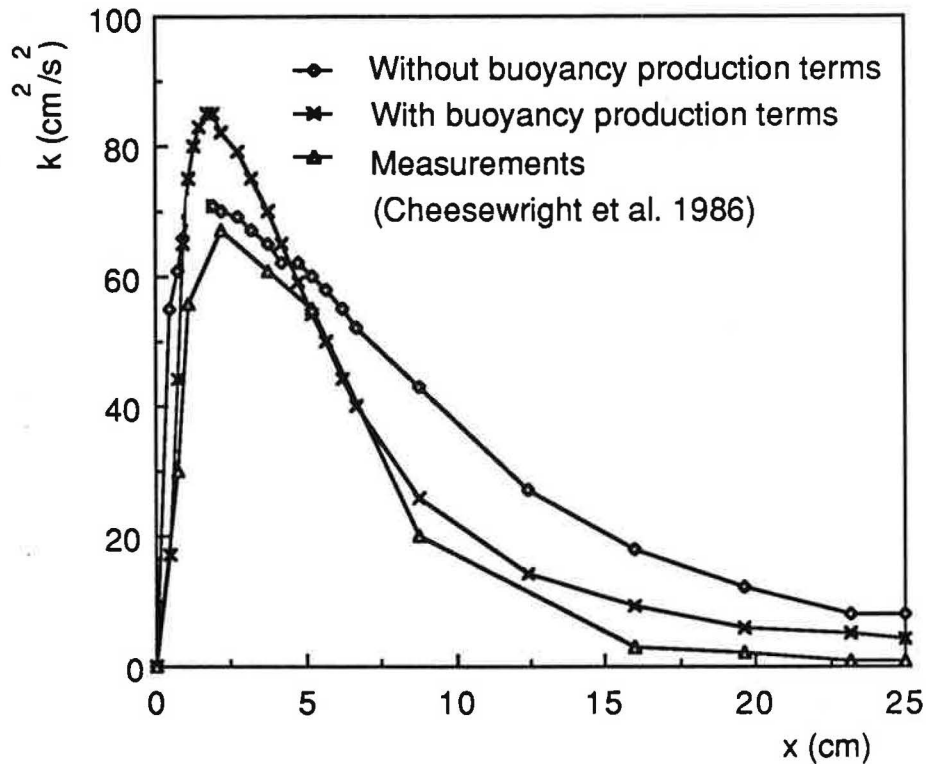


Figure 4. Profiles of turbulence energy in the mid-height

The model has been applied for the predictions of natural convection in a large-scale air-filled cavity. The boundary conditions are set with the following two different kinds of methods:

- Method one is recommended by Parry (1989) and
- Method two uses the standard wall-function formulae but the distance between the first grid and the wall should be very small.

The computed results with Method two are in rather good agreement with the measurements but with Method one deviate from the experimental data. The setting of boundary conditions has a very significant influence in the computed results. The velocity profile computed by Method two is better than that by the standard high-Reynolds-number model.

Compared with the experimental data, the computations by Method two give a correct relationship between local Rayleigh number and local Nusselt number. The standard high-Reynolds-number model may generate a grid-dependent value of heat transfer. Hence the model is not suitable for natural convection in a cavity.

The buoyancy production terms are important for the computations of kinetic energy of turbulence.

## 7. ACKNOWLEDGMENTS

This investigation was financially supported by the Swiss Federal Office of Energy (BEW). The author is indebted to Mr. C. Schönenberger at ETH-Zentrum for his preliminary work. Our thanks are also due to Mr. J.D. Parry of CHAM Ltd, London for the help in the computer coding details.

## 8. REFERENCES

- Cheesewright, R.; King, K.J. and Ziai, S. 1986. "Experimental data for the validation of computer codes for the prediction of two-dimensional buoyant cavity flows." *Significant Questions in Buoyancy Affected Enclosure or Cavity Flows*. ed: Humphrey, J.A.C et al. pp. 75-81. New York: ASME.
- Chen, Q. 1988. *Indoor Airflow, Air Quality and Energy Consumption of Buildings*, Ph.D. thesis, Delft University of Technology, The Netherlands.
- Chen, Q.; Moser, A. and Huber, A. 1990. "Prediction of buoyant, turbulent flow by a low-Reynolds-number k- $\epsilon$  model", *ASHRAE Transactions*, Vol. 96, Part 1.

- Lam, C.K.G. and Bremhorst, K. 1981 "A modified form of the k- $\epsilon$  model for predicting wall turbulence." *Journal of Fluids Engineering*. ASME Transactions. Vol. 103, pp. 456-460.
- Launder, B.E. and Spalding, D.B. 1974. "The numerical computation of turbulent flows." *Computer Methods in Applied Mechanics and Engineering*. Vol. 3, pp. 269-289.
- Nielsen, P.V.; Restivo, A. and Whitelaw, J.H. 1979. "Buoyancy-affected flows in ventilated rooms." *Numerical Heat Transfer*, Vol. 2, pp. 115-127.
- Parry, J.D. 1989. *Private Communications*, CHAM Ltd, London.
- Patel, W.C., Rodi, W., and Scheuerer G. 1985, "Turbulence models for near-wall and low Reynolds number flows: a review." *AAIA Journal*, Vol. 23, No. 9, pp. 1308-1319.
- Rosten, H.I. and Spalding, D.B. 1987. *The PHOENICS Reference Manual*, Technical Report 200, Version 1.4, CHAM Ltd, London.

## 9. NON-STANDARD NOMENCLATURE

Symbol	Meaning
$A_\mu, A_t, A_{C1}$	= turbulence model constants
$f_1, f_2, f_\mu$	= turbulence model functions
$R_k$	= turbulence Reynolds number $\frac{k^{1/2}y}{\nu_1}$
$R_t$	= turbulence Reynolds number $\frac{k^2}{\nu_1\epsilon}$
$S_k, S_\epsilon$	= source terms in k and $\epsilon$ equations due to buoyancy
$T_h, T_o$	= temperatures at the hot wall and cavity core
y	= distance from a cell centre to the nearest wall
$\beta$	= expansion coefficient

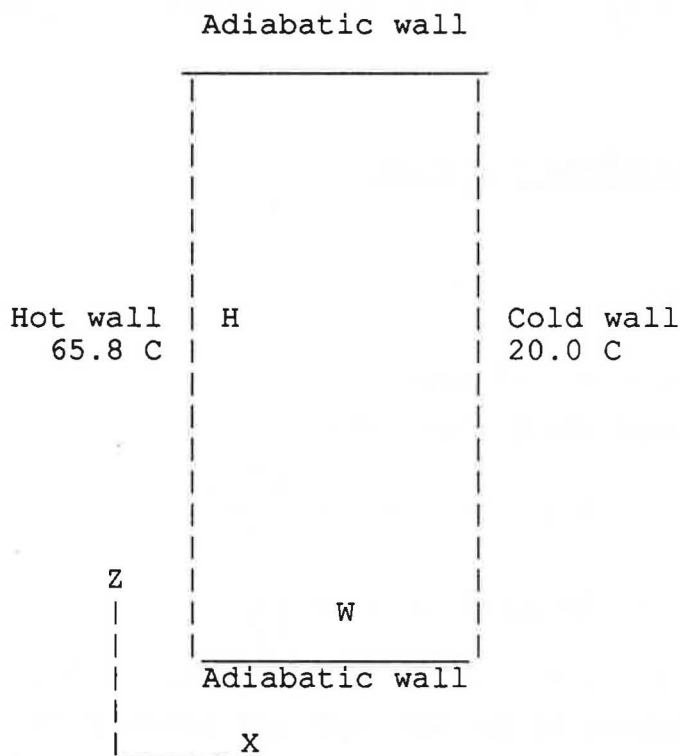
**APPENDIX A****A.1 The O1 file**

```
TALK=F;RUN( 1, 1);VDU=HP
  GROUP 1. Run title and other preliminaries
TEXT(Low Re k-e model & Buoyancy production.
```

```
### General comments begin ###
```

This is a two dimensional natural convection problem.  
This case is steady and incompressible and uses Boussinesq approximation.

The cavity dimension is  $H=2.5$  m in height and  $W=0.5$  m in width. The temperature difference between the hot and cold wall surfaces is 45.8 K and the cold surface temperature is assumed to be 20.0 C. The top and the bottom walls are adiabatic.



The sketch of the cavity

```
### General comments end ###
```

```

** Input data:
REAL(TCOLD, THOT, TDIFF, CP, RHO, BETA, MU, HEIGHT, WIDTH)
** Cold wall surface temperature: TCOLD
TCOLD=20.0
** Temperature difference between the cold and hot
wall surfaces: TDIFF
TDIFF=45.8
** Hot wall surface temperature: THOT
THOT=TCOLD+TDIFF
** Constant-pressure specific heat: CP
CP=1.008E3
** Air density: RHO
RHO=1.112
** Air expansion coefficient: BETA ..
BETA=3.2E-3
** Kinematic viscosity: MU
MU=1.692E-5
** Cavity height: HEIGHT
HEIGHT=2.5
** Cavity width: WIDTH
WIDTH=0.5
GROUP 2. Transience; time-step specification
GROUP 3. X-direction grid specification
NX=50
XFRAC(1)=-10; XFRAC(2)=0.002; XFRAC(3)=10; XFRAC(4)=0.005
XFRAC(5)=10; XFRAC(6)=0.036; XFRAC(7)=10; XFRAC(8)=0.005
XFRAC(9)=10; XFRAC(10)=0.002
GROUP 4. Y-direction grid specification
GROUP 5. Z-direction grid specification
NZ=50
ZFRAC(1)=-10; ZFRAC(2)=0.01; ZFRAC(3)=10; ZFRAC(4)=0.025
ZFRAC(5)=10; ZFRAC(6)=0.18; ZFRAC(7)=10; ZFRAC(8)=0.025
ZFRAC(9)=10; ZFRAC(10)=0.01
GROUP 6. Body-fitted coordinates or grid distortion
GROUP 7. Variables stored, solved & named
SOLVE(P1,U1,W1,H1);SOLUTN(P1,Y,Y,Y,N,N,N)

===== Comments begin:

** For the low-Reynolds-number k-e model, the following
block should be used.
SOLVE(KE,EP)
C11 stores fmu; C12 stores fl; C13 stores f2 & C14 stores
Rt
STORE(ENUT,C11,C12,C13,C14)
ENUT=GRND; EL1=GRND4; KELIN=0; ENUL=GRND; ENULA=MU
RSG3, RSG4, RSG5 is the overall length in x, y and z
directions.

```

```

RSG3=WIDTH; RSG4=1.0; RSG5=HEIGHT
PATCH (LOWRE, PHASEM, 1, NX, 1, NY, 1, NZ, 1, LSTEP)
COVAL (LOWRE, KE, GRND10, GRND10)
COVAL (LOWRE, EP, GRND10, GRND10)
GENK=T
OUTPUT (C11, Y, N, N, N, Y, Y); OUTPUT (C12, Y, N, N, N, Y, Y)
OUTPUT (C13, Y, N, N, N, Y, Y); OUTPUT (C14, Y, N, N, N, Y, Y)

```

=====  
===== Comments end.

```

GROUP 8. Terms (in differential equations) & devices
DIFCUT=0.0
GROUP 9. Properties of the medium (or media)
RHO1=RHO; PRNDTL(H1)=0.69; PRT(H1)=0.9
GROUP 10. Inter-phase-transfer processes and properties
GROUP 11. Initialization of variable or porosity fields
** Initial values
FIINIT(U1)=0.0; FIINIT(W1)=0.0
FIINIT(KE)=1.E-4; FIINIT(EP)=1.E-4
FIINIT(H1)=(TCOLD+THOT)*CP/2.0
GROUP 12. Convection and diffusion adjustments
GROUP 13. Boundary conditions and special sources

```

=====  
===== Comments begin:

### Method 1 ###

This block uses the boundary conditions recommended by John D. Parry of CHAM. The boundary conditions must be used together with the low-Reynolds-number k-e model. For EP, the patch name must begin with "LBB..." and the patch type is not very important because the COefficient is FIXVAL which is a very large value. However, a PHASEM will speed up the convergence.

```

** Hot wall (west)
PATCH (HOT, WWALL, 1, 1, 1, NY, 1, NZ, 1, 1)
COVAL (HOT, W1, 1.0, 0.0)
COVAL (HOT, KE, 1.0, 0.0)
COVAL (HOT, H1, 1.0/PRNDTL(H1), THOT*CP)
PATCH (LBBHOT, PHASEM, 1, 1, 1, NY, 1, NZ, 1, 1)
COVAL (LBBHOT, EP, FIXVAL, GRND8)
** Cold wall (east)
PATCH (COLD, EWALL, NX, NX, 1, NY, 1, NZ, 1, 1)
COVAL (COLD, W1, 1.0, 0.0)
COVAL (COLD, KE, 1.0, 0.0)
COVAL (COLD, H1, 1.0/PRNDTL(H1), TCOLD*CP)
PATCH (LBBCOL, PHASEM, NX, NX, 1, NY, 1, NZ, 1, 1)
COVAL (LBBCOL, EP, FIXVAL, GRND8)
** Bottom wall (low)

```



```

PATCH(FLOOR,LWALL,1,NX,1,NY,1,1,1)
COVAL(FLOOR,U1,1.0,0.0)
COVAL(FLOOR,KE,1.0,0.0)
PATCH(LBBFLR,PHASEM,1,NX,1,NY,1,1,1)
COVAL(LBBFLR,EP,FIXVAL,GRND8)
  ** Top wall (high)
PATCH(CEILING,HWALL,1,NX,1,NY,NZ,NZ,1,1)
COVAL(CEILING,U1,1.0,0.0)
COVAL(CEILING,KE,1.0,0.0)
PATCH(LBBCEI,PHASEM,1,NX,1,NY,NZ,NZ,1,1)
COVAL(LBBCEI,EP,FIXVAL,GRND8)

```

```

===== Comments end.

```

```

===== Comments begin:

```

```

### Method 2 ###

```

This block uses the standard wall functions. If they are applied for the low-Reynolds-number k-e model, the distance of the first grid should be as small as possible so that the first grid will be located in the viscous sublayer.

```

  ** Hot wall (west)
PATCH(HOT,WWALL,1,1,1,NY,1,NZ,1,1)
COVAL(HOT,W1,GRND2,0.0)
COVAL(HOT,KE,GRND2,GRND2);COVAL(HOT,EP,GRND2,GRND2)
COVAL(HOT,H1,GRND2,THOT*CP)
  ** Cold wall (east)
PATCH(COLD,EWALL,NX,NX,1,NY,1,NZ,1,1)
COVAL(COLD,W1,GRND2,0.0)
COVAL(COLD,KE,GRND2,GRND2);COVAL(COLD,EP,GRND2,GRND2)
COVAL(COLD,H1,GRND2,TCOLD*CP)
  ** Bottom wall (low)
PATCH(FLOOR,LWALL,1,NX,1,NY,1,1,1)
COVAL(FLOOR,U1,GRND2,0.0)
COVAL(FLOOR,KE,GRND2,GRND2);COVAL(FLOOR,EP,GRND2,GRND2)
  ** Top wall (high)
PATCH(CEILING,HWALL,1,NX,1,NY,NZ,NZ,1,1)
COVAL(CEILING,U1,GRND2,0.0)
COVAL(CEILING,KE,GRND2,GRND2)
COVAL(CEILING,EP,GRND2,GRND2)

```

```

===== Comments end.

```

```

  ** Fixed pressure
  - gives a reference value for the pressure field:
PATCH(PFIXED,LOW,3,3,1,NY,1,1,1)
COVAL(PFIXED,P1,FIXVAL,0.0)

```

```

COVAL (PFIXED, W1, 0.0, 0.0)
COVAL (PFIXED, H1, 0.0, TCOLD*CP)
  ** Buoyancy source in the momentum equation
    (Boussinesq approximation):
PATCH (BUOY, PHASEM, 1, NX, 1, NY, 1, NZ, 1, 1)
COVAL (BUOY, W1, FIXFLU, GRND3)

```

```

===== Comments begin:

```

```

  ** For the buoyancy production terms in the KE and EP
    equations:
PATCH (KEEPBUOY, PHASEM, 1, NX, 1, NY, 1, NZ, 1, 1)
COVAL (KEEPBUOY, KE, GRND, GRND)
COVAL (KEEPBUOY, EP, GRND, GRND)

```

```

===== Comments end.

```

```

  GROUP 14. Downstream pressure for PARAB=.TRUE.
  GROUP 15. Termination of sweeps
LSWEEP=2000
  GROUP 16. Termination of iterations
  GROUP 17. Under-relaxation devices
RELAX (P1, LINRLX, 0.5); RELAX (U1, FALSDT, 0.3)
RELAX (V1, FALSDT, 0.3); RELAX (W1, FALSDT, 0.1)
RELAX (KE, FALSDT, 0.3); RELAX (EP, FALSDT, 0.3)
RELAX (H1, FALSDT, 0.3)
  GROUP 18. Limits on variables or increments to them
VARMIN (U1)=-1.0;      VARMAX (U1)=1.0
VARMIN (W1)=-1.0;      VARMAX (W1)=1.0
VARMIN (KE)=1.0E-10
VARMIN (EP)=1.0E-10
VARMIN (H1)=TCOLD*CP; VARMAX (H1)=THOT*CP
  GROUP 19. Data communicated by satellite to GROUND
  ** For Boussinesq approximation
RSG2=-BETA/CP; RSG1=RSG2*(TCOLD+THOT)*CP/2.0; RSG10=-9.81
  ** For the buoyancy production terms in the k and e
    equations
RG (1)=BETA; RG (2)=CP
  GROUP 20. Preliminary print-out
  GROUP 21. Print-out of variables
  GROUP 22. Spot-value print-out
IXMON=1; IYMON=1; IZMON=NZ/2; TSTSWP=50
  GROUP 23. Field print-out and plot control
ITABL=3; IPLTL=LSWEEP; NUMCLS=10; XZPR=T; NPRINT=LSWEEP
  GROUP 24. Dumps for restarts
STOP

```

**A.2 The GROUND File****Subroutine GROUND**

```

C*****
C
C      SUBROUTINE GROUND
C
C      ... ..
C
C 2   User dimensions own arrays here, for example:
C     DIMENSION UUH(10,10),UUC(10,10),UUX(10,10),UUZ(10)
C=====C
C
C     Modified on 24 April, 1989.
C
C     This modification is for storing (PHI) cell centre
C     coordinate values.
C
C     The common block below has been used by the following
C     subroutines:
C     PROGRAM MAIN; GROUND; GETCOOR, DPHIDX, DPHIDY, DPHIDZ,
C     BUOYKE, VARTOF.
C     WARNING: The corresponding arrays in the above
C     subroutines must have the same dimensions!!!
C
C     XCOOR, YCOOR, and ZCOOR store the cell centre
C     coordinate values.
C     BKEEP is used for setting of boundary conditions of KE
C     or EP.
C     LOADED is a logical variable used to control that the
C     cell centre coordinate values are only stored once in
C     the run;
C     BUKESO stores Sk/EMUT - Buoyancy production for KE.
C
C     COMMON/BUOYANT/XCOOR,YCOOR,ZCOOR,BKEEP,LOADED,BUKESO
C     DIMENSION XCOOR(100),YCOOR(100),ZCOOR(100)
C     &,BKEEP(100,100), BUKESO(100,100)
C     LOGICAL LOADED
C     COMMON/NEARW/C1MOD(100,100),C2MOD(100,100)
C=====C
C
C 3   User places his data statements here, for example:
C     DATA NXDIM,NYDIM/10,10/
C
C     ... ..

```

```

C
C*****
C
C--- GROUP 1. Run title and other preliminaries
C
C      1 GO TO (1001,1002),ISC
C=====C
C
C      Modified on 24 April, 1989.
C
C      This modification is to store the cell centre
C      coordinate values in the F-array
C
C      1001 CALL SUB2(IX, 1, IY, 1)
C
C      The MAKE(...) command prepares the storage place.
C
C      LXXG : holds the x-coordinate values of the cell
C             centres
C      LYYG : holds the y-coordinate values of the cell
C             centres
C      LZZG : holds the z-coordinate values of the cell
C             centres
C
C      IF (NX .GT. 1) CALL MAKE(LXXG)
C      IF (NY .GT. 1) CALL MAKE(LXYG)
C      IF (NZ .GT. 1) CALL MAKE(LZZG)
C
C      EASP6 is used for boundary conditions for KE or EP
C      when the low-Reynolds-number model is applied.
C      EASP7 stores the buoyancy terms for the calculation of
C      KE and EP
C      EASP7 is equivalent to BUKESO.
C      EASP5, EASP8, EASP9, EASP10 are used for the low-
C      Reynolds-number cases and they are always available
C      (see GREX2)
C
C      CALL MAKE(EASP5)
C      CALL MAKE(EASP6)
C      CALL MAKE(EASP7)
C=====C
C      RETURN
C      1002 CONTINUE
C      RETURN
C
C      ... ..
C
C      95 CONTINUE
C      * ----- SECTION 5 -----
C      For ENUT.LE.GRND--- reference turbulent kinematic
C      viscosity, for which the block-location index is

```

```

C   AUX(VIST) .
C-----C
C
C   MODIFICATION : 24 April, 1989.
C
C   This modification is for Lam-Bremhorst low-Reynolds-
C   number turbulence model.
C
C   The following build-in subroutines are used:
C   - FN0(Y,X)           ->  Y = X
C   - FN1(Y,A)           ->  Y = A
C   - FN2(Y,X,A,B)       ->  Y = A + B * X
C   - FN22(Y,A)          ->  Y = AMAX1(Y,A)
C   - FN23(Y,A)          ->  Y = AMIN1(Y,A)
C   - FN25(Y,A)          ->  Y = A * Y
C   - FN26(Y,X)          ->  Y = Y * X
C   - FN27(Y,X)          ->  Y = Y / X
C   - FN28(Y,X,A)        ->  Y = A / X
C   - FN31(Y,X1,X2,A,B1,B2) ->  Y = A*(X1**B1)*(X2 ** B2)
C   - FN33(Y,A)          ->  Y = Y + A
C   - FN36(Y,X,A,B)      ->  Y = Y + A * exp(B * X)
C   - FN50(Y,I)          ->  Y = Y ** I
C   - FN56(Y,X1,X2,X3,A) ->  Y = A * X1 * X2 / X3
C   - FN61(Y,X,A)        ->  Y = A * X ** 3
C
C   Get the coordinate values of the cell centres:
C
C       IF (.NOT.LOADED) CALL GETCOOR
C
C   1. Calculation of the smallest distance from a cell centre
C   to the walls. These distances were directly stored in
C   the F-array (address EASP10).
C
C   The maximum coordinates of x, y, and z are given in
C   RSG3, RSG4 and RSG5.
C
C       KK = LOF(EASP10)
C       DO 911 IX = 1, NX
C       DO 911 IY = 1, NY
C           KK = KK + 1
C           WDIST = 1.E10
C
C           IF (NX.EQ.1) GO TO 912
C           WDIST = AMIN1(WDIST, XCOOR(IX))
C           WDIST = AMIN1(WDIST, RSG3 - XCOOR(IX))
C
C       912 IF (NY.EQ.1) GO TO 913
C           WDIST = AMIN1(WDIST, YCOOR(IY))
C           WDIST = AMIN1(WDIST, RSG4 - YCOOR(IY))
C
C       913 IF (NZ.EQ.1) GO TO 914

```

```

      WDIST = AMIN1(WDIST, ZCOOR(IZ))
      WDIST = AMIN1(WDIST, RSG5 - ZCOOR(IZ))
C
914   F(KK) = WDIST
911   CONTINUE
C
C 2. Calculation of the factor fmu
C
      CALL FN31(EASP8, KE, EASP10, 1.0, 0.5, 1.0)
      CALL FN27(EASP8, AUX(VISL))
C Avoid exponent underflow ...
      CALL FN23(EASP8, 1.E3)
      CALL FN1(EASP9, 1.0)
      CALL FN36(EASP9, EASP8, -1.0, -0.0165)
      CALL FN50(EASP9, 2)
C Find Rt
C Avoid negative epsilon ...
      CALL FN22(EP, 1.E-10)
      CALL FN56(EASP10, KE, KE, EP, 1.0)
      CALL FN27(EASP10, AUX(VISL))
C Store Rt
      CALL FN0(C14, EASP10)
      CALL FN28(EASP8, EASP10, 20.5)
      CALL FN33(EASP8, 1.0)
      CALL FN26(EASP8, EASP9)
C Avoid fmu greater than 1.0 ...
      CALL FN23(EASP8, 1.0)
C Store fmu for plotting ...
      CALL FN0(C11, EASP8)
C Store fmu in EASP5 for further uses
      CALL FN0(EASP5, EASP8)
C
C 3. Calculation of ENUT
C (Prandtl-Kolmogorov formula - modified)
C
      CALL FN22(KE, 1.E-10)
      CALL FN31(AUX(VIST), KE, EP, CMUCD, 2.0, -1.0)
      CALL FN26(AUX(VIST), EASP8)
C
C 4. Calculation of f1
C
      CALL FN28(EASP9, EASP8, 0.05)
      CALL FN61(EASP8, EASP9, 1.0)
      CALL FN33(EASP8, 1.0)
C
C Store f1 for plotting...
C
      CALL FN0(C12, EASP8)
C
C Calculate C1*f1
C

```

```

      CALL FN25(EASP8,C1E)
C
C 5. Calculation of f2
C
      CALL FN50(EASP10, 2)
C Avoid exponent underflow ...
      CALL FN23(EASP10, 1.E2)
      CALL FN1(EASP9, 1.0)
      CALL FN36(EASP9, EASP10, -1.0, -1.0)
C
C Store f2 for plotting ...
      CALL FN0(C13, EASP9)
C Calculate C2*f2
      CALL FN25(EASP9,C2E)
C
C 6. Storage of C1*f1 and C2*f2 in C1MOD and C2MOD
C
      K1 = LOF(EASP8)
      K2 = LOF(EASP9)
      DO 918 IX = 1, NX
      DO 918 IY = 1, NY
          K1 = K1 + 1
          K2 = K2 + 1
          C1MOD(IY, IX) = F(K1)
          C2MOD(IY, IX) = F(K2)
918 CONTINUE
C
C=====C
      RETURN
96 CONTINUE
C * ----- SECTION 6 -----
C For ENUL.LE.GRND--- reference laminar kinematic
C viscosity for which the block-location index is
C AUX(VISL).
C=====C
C
C MODIFICATION : 24 April, 1989.
C
C Store ENULA in VISL for the application in Section 9.5
C (above)
C
C FN1(Y,A) -> Y = A
      CALL FN1(AUX(VISL), ENULA)
C
C=====C
      RETURN
      . . . . .
C--- GROUP 13. Boundary conditions and special sources
C

```

```

13 CONTINUE
  GO TO (130,131,132,133,134,135,136,137,138,139,1310,
11311,1312,1313,1314,1315,1316,1317,1318,1319,
21320,1321),ISC
130 CONTINUE
C----- SECTION 1 ----- coefficient = GRND
C=====C
C
C Modification on 24 April, 1989.
C
C This section inserts the COefficients for KE and EP
C due to the buoyancy production.
C
C Coefficient for KE (INDVAR = 12) is 1.0E-11*EMUT
C
C FN2(Y, X, A, B) <--> Y = A + B*X
C IF(INDVAR.EQ.12)
C 1 CALL FN2(CO, AUX(VIST), 0.0, 1.0E-11)
C
C Coefficient for EP (INDVAR = 13) is 1.0E-11*KE/EP
C FN31(Y, X1, X2, A, B1, B2) <--> Y=A*(X1**B1)*(X2**B2)
C IF(INDVAR.EQ.13) CALL
C 1FN31(CO,AUX(VIST),AUX(LEN1), (1.0E-11*CD/CMU),1.0,-2.0)
C
C=====C
C RETURN
131 CONTINUE
... ..
1310 CONTINUE
C----- SECTION 11 ----- coefficient = GRND10
C=====C
C
C MODIFICATION : 24 April, 1989
C
C This block is used for the low-Reynolds-number model.
C We use the similar way as that for the high-Reynolds-
C number model in subroutine GXKESO.
C
C 1. COefficient of KE in the low-Reynolds-number model.
C
C IF (INDVAR.NE.12) GO TO 1350
C CALL FN31(CO, AUX(VIST), AUX(LEN1), CD/CMU, 1.0, -2.0)
C CALL FN27(CO, EASP5)
C RETURN
C
C 2. COefficient of EP in the low-Reynolds-number model.
C
C 1350 IF (INDVAR.NE.13) RETURN
C

```



```

C      Put C2*f2 from C2MOD to the F array
C
      K2 = L0F(EASP9)
      DO 1351 IX = 1, NX
          DO 1351 IY = 1, NY
              K2 = K2 + 1
              F(K2) = C2MOD(IY, IX)
1351      CONTINUE
C
      CALL FN26(EASP9,AUX(VIST))
      CALL FN31(CO, EASP9, AUX(LEN1), CD/CMU, 1.0, -2.0)
      CALL FN27(CO, EASP5)
C
C=====C
      RETURN
1311 CONTINUE
C----- SECTION 12 ----- value = GRND
C=====C
C
C Modified on 24 April, 1989.
C
C      This section inserts the VALues for KE and EP due to
C      buoyancy productions.
C
C      Value for KE (INDVAR = 12) is 1.0E11*Sk/EMUT
C
C      IF(INDVAR.NE.12) GOTO 13110
C
C      BUOYKE writes the Sk/ENUT term for the k-e model
C      in the array BUKESO.
C
C      CALL BUOYKE
C
C      VARTOF(LB) stores the content of the NX*NY-Array
C      BUKESO in the F-Array. LB is the block-location index
C      within F.
C
C      CALL VARTOF(EASP7)
C
C      FN2(Y, X, A, B) <--> Y = A + B*X
C
C      CALL FN2(VAL, EASP7, 0.0, 1.0E11)
      RETURN
C
13110 IF(INDVAR.NE.13) RETURN
C
C      Value for EP (INDVAR =13) is 1.0E11*C3*Sk
C
C      CALL VARTOF(EASP7)
C
C      FN31(Y, X1, X2, A, B1, B2) <--> Y=A*(X1**B1)*(X2**B2)

```

```

C     C3E is a further model constant for buoyant turbulent
C     flows
C
C     C3E = 1.44
C
C     FN21(Y, X1, X2, A, B) <--> Y = A + B * X1 * X2
C     CALL FN21(VAL, AUX(VIST), EASP7, 0.0, 1.0E11*C3E)
C
C=====C
C     RETURN
C     1312 CONTINUE
C
C     ... ..
C
C     1319 CONTINUE
C----- SECTION 20 ----- value = GRND8
C=====C
C
C MODIFICATION : 24 April, 1989.
C
C This block sets the boundary conditions for the low-
C Reynolds-number models. The first one set EP values for
C the grid cell close to a wall which is recommended by
C John D Parry of CHAM (Method 1).
C
C For the boundary conditions of EP.
C
C     ### Method 1 ###
C
C - EP value at the grid cell close to the wall.
C
C In the Q1 file, this is called by a patch name equal to
C "LBB...". The patch type can be any type. The
C COefficient is FIXVAL. However, a PHASEM is better
C for convergence.
C
C     IF(NPATCH(1:3).NE.'LBB') RETURN
C
C Get coordinate values of the cell centres
C     IF(.NOT.LOADED) CALL GETCOOR
C
C     IF(INDVAR.NE.13) RETURN
C
C Calculate the distance between the first grid and the wall
C -- y and store in EASP6
C     K6 = LOF(EASP6)
C     DO 13190 IX=1, NX
C     DO 13190 IY=1, NY
C         BKEEP(IY,IX)=0.0
C
C     IF(NX.EQ.1) GOTO 13191

```

```

      IF (IX.EQ. 1) BKEEP (IY, IX) = XCOOR(1)
      IF (IX.EQ. NX) BKEEP (IY, IX) = RSG3 - XCOOR (NX)
C
13191  IF (NY.EQ.1) GOTO 13192
      IF (IY.EQ. 1) BKEEP (IY, IX) = YCOOR (1)
      IF (IY.EQ. NY) BKEEP (IY, IX) = RSG4 - YCOOR (NY)
C
13192  IF (NZ.EQ.1) GOTO 13193
      IF (IZ.EQ. 1) BKEEP (IY, IX) = ZCOOR (1)
      IF (IZ.EQ. NZ) BKEEP (IY, IX) = RSG5 - ZCOOR (NZ)
C
13193  K6 = K6 + 1
      F(K6) = BKEEP (IY, IX)
13190  CONTINUE
C
C Set the value for EP
C FN56(Y, X1, X2, X3, A) <--> Y = A*X1*X2/X3
C
      CALL FN56 (VAL, AUX (VISL), KE, EASP6, 2.0)
C
C FN27(Y, X) <--> Y = Y/X
C
      CALL FN27 (VAL, EASP6)
      RETURN
C=====C
1320  CONTINUE
C----- SECTION 21 ----- value = GRND9
      RETURN
1321  CONTINUE
C----- SECTION 22 ----- value = GRND10
C=====C
C
C MODIFICATION : 24 April, 1989.
C
C This block is also used for the low-Reynolds-number
C model. We use the similar way as that for the high-
C Reynolds-number model in subroutine GXKESO.
C
C
C 1. VALUE of KE of the low-Reynolds-number model.
C
      IF (INDVAR.NE.12) GO TO 1352
      CALL FN31 (VAL, LGEN1, AUX (LEN1), CMU/CD, 1.0, 2.0)
      CALL FN26 (VAL, EASP5)
      RETURN
C
C
C 2. VALUE of EP of the low-Reynolds-number model.
C
1352  IF (INDVAR.NE.13) RETURN
C

```

```
C Put C1*f1 and C2*f2 to the F array
```

```
C
```

```

K1 = LOF(EASP8)
K2 = LOF(EASP9)
DO 1353 IX = 1, NX
DO 1353 IY = 1, NY
    K1 = K1 + 1
    K2 = K2 + 1
    F(K1) = C1MOD(IY, IX)
    F(K2) = C2MOD(IY, IX)

```

```
1353 CONTINUE
```

```
C
```

```

CALL FN22(EASP9, 1.E-10)
CALL FN27(EASP8, EASP9)
CALL FN26(EASP8, AUX(VIST))
CALL FN21(VAL, EASP8, LGEN1, 0.0, 1.0)
CALL FN26(VAL, EASP5)

```

```
C
```

```
C=====C
RETURN
```

```
... ..
```

### Other auxiliary subroutines

```
C=====C
```

```
C- Subroutine GETCOOR
```

```
C
```

```
C Modified on 24 April, 1989.
```

```
C
```

```
C GETCOOR loads the cell centre coordinates (PHI grids)
C in the arrays XCOOR, YCOOR and ZCOOR.
```

```
C
```

```
    SUBROUTINE GETCOOR
```

```
*COPY SATEAR
```

```
*COPY GRDLOC
```

```
*COPY GRDEAR
```

```
C
```

```

COMMON/BUOYANT/XCOOR, YCOOR, ZCOOR, BKEEP, LOADED, BUKESO
DIMENSION XCOOR(100), YCOOR(100), ZCOOR(100)
&, BKEEP(100,100), BUKESO(100,100)
LOGICAL LOADED

```

```
C
```

```
    IF (NX.EQ.1) GO TO 100
```

```
C
```

```

DO 10 II=1, NX
    LFXG = LOF(LXXG) + II
    XCOOR(II) = F(LFXG)

```

```
10
```

```
C
```

```
100 IF (NY.EQ.1) GO TO 200
```

```

C
DO 20 JJ=1,NY
  LFYG = LOF(LYYG) + JJ
20   YCOOR(JJ) = F(LFYG)
C
200  IF (NZ.EQ.1) GO TO 300
C
DO 30 KK=1,NZ
  LFZG = LOF(LZZG) + KK
30   ZCOOR(KK) = F(LFZG)
C
C Coordinates loaded ...
300  LOADED = .TRUE.
C
RETURN
END

```

```

C=====C
C- Function DPHIDX
C
C Modified on 24 April, 1989.
C
C DPHIDX calculates the local derivative of a scalar
C property in x-direction when II=1 and the local
C derivative of the square root of a scalar property
C when II=2.
C
FUNCTION DPHIDX(INDEXV,II)
C
*COPY SATEAR
*COPY GRDLOC
*COPY GRDEAR
C
COMMON/BUOYANT/XCOOR,YCOOR,ZCOOR,BKEEP,LOADED,BUKESO
DIMENSION XCOOR(100),YCOOR(100),ZCOOR(100)
&,BKEEP(100,100),BUKESO(100,100)
LOGICAL LOADED
C
IF (.NOT.LOADED) CALL GETCOOR
C
IF (IX.EQ.NX) GOTO 1
IF (IX.EQ. 1) GOTO 2
C
C Cell not at border
LFPHE = LOF(INDEXV) + IY + NY*IX
LFPHW = LOF(INDEXV) + IY + NY*(IX-2)
IF (II.EQ.1) DPHIDX = (F(LFPHE) - F(LFPHW))
& / (2*(XCOOR(IX+1) - XCOOR(IX-1)))
IF (II.EQ.2) DPHIDX = (SQRT(F(LFPHE)) - SQRT(F(LFPHW)))
& / (2*(XCOOR(IX+1) - XCOOR(IX-1)))
RETURN

```

```

C
C Cell at EAST border
  1 LFPHI = LOF(INDEXV) + IY + NY*(IX-1)
    LFPHW = LOF(INDEXV) + IY + NY*(IX-2)
    IF(II.EQ.1) DPHIDX = (F(LFPHI) - F(LFPHW))
      & / (XCOOR(IX) - XCOOR(IX-1))
    IF(II.EQ.2) DPHIDX = (SQRT(F(LFPHI)) - SQRT(F(LFPHW)))
      & / (XCOOR(IX) - XCOOR(IX-1))
    RETURN

```

```

C
C Cell at WEST border
  2 LFPHE = LOF(INDEXV) + IY + NY*IX
    LFPHI = LOF(INDEXV) + IY + NY*(IX-1)
    IF(II.EQ.1) DPHIDX = (F(LFPHE) - F(LFPHI))
      & / (XCOOR(IX+1) - XCOOR(IX))
    IF(II.EQ.2) DPHIDX = (SQRT(F(LFPHE)) - SQRT(F(LFPHI)))
      & / (XCOOR(IX+1) - XCOOR(IX))
    RETURN

```

```

C
  END

```

```

C=====C

```

```

C- Function DPHIDY

```

```

C
C   Modified on 24 April, 1989.

```

```

C
C   DPHIDY calculates the local derivative of a scalar
C   property in y-direction when II=1 and the local
C   derivative of the square root of a scalar property
C   when II=2.

```

```

C
C   FUNCTION DPHIDY(INDEXV, II)

```

```

C
C   *COPY SATEAR
C   *COPY GRDLOC
C   *COPY GRDEAR

```

```

C
C   COMMON/BUOYANT/XCOOR, YCOOR, ZCOOR, BKEEP, LOADED, BUKESO
C   DIMENSION XCOOR(100), YCOOR(100), ZCOOR(100)
C   &, BKEEP(100,100), BUKESO(100,100)
C   LOGICAL LOADED

```

```

C
C   IF (.NOT.LOADED) CALL GETCOOR

```

```

C
C   IF (IY.EQ.NY) GOTO 1
C   IF (IY.EQ. 1) GOTO 2

```

```

C
C Cell not at border
  LFPHN = LOF(INDEXV) + (IY+1) + NY*(IX-1)
  LFPHS = LOF(INDEXV) + (IY-1) + NY*(IX-1)
  IF(II.EQ.1) DPHIDY = (F(LFPHN) - F(LFPHS))

```

```

& / (2*(YCOOR(IY+1) - YCOOR(IY-1)))
IF(II.EQ.2) DPHIDY = (SQRT(F(LFPHN)) - SQRT(F(LFPHS)))
& / (2*(YCOOR(IY+1) - YCOOR(IY-1)))
RETURN
C
C Cell at NORTH border
1 LFPHI = LOF(INDEXV) + IY + NY*(IX-1)
LFPHS = LOF(INDEXV) + (IY-1) + NY*(IX-1)
IF(II.EQ.1) DPHIDY = (F(LFPHI) - F(LFPHS))
& / (YCOOR(IY) - YCOOR(IY-1))
IF(II.EQ.2) DPHIDY = (SQRT(F(LFPHI)) - SQRT(F(LFPHS)))
& / (YCOOR(IY) - YCOOR(IY-1))
RETURN
C
C Cell at SOUTH border
2 LFPHN = LOF(INDEXV) + (IY+1) + NY*(IX-1)
LFPHI = LOF(INDEXV) + IY + NY*(IX-1)
IF(II.EQ.1) DPHIDY = (F(LFPHN) - F(LFPHI))
& / (YCOOR(IY+1) - YCOOR(IY))
IF(II.EQ.2) DPHIDY = (SQRT(F(LFPHN)) - SQRT(F(LFPHI)))
& / (YCOOR(IY+1) - YCOOR(IY))
RETURN
C
END

```

=====C

C- Function DPHIDZ

C

Modified on 24 April, 1989.

C

DPHIDZ calculates the local derivative of a scalar property in z-direction when II=1 and the local derivative of the square root of a scalar property when II=2.

C

FUNCTION DPHIDZ(INDEXV, II)

C

\*COPY SATEAR

\*COPY GRDLOC

\*COPY GRDEAR

C

```

COMMON/BUOYANT/XCOOR, YCOOR, ZCOOR, BKEEP, LOADED, BUKESO
DIMENSION XCOOR(100), YCOOR(100), ZCOOR(100)
&, BKEEP(100,100), BUKESO(100,100)
LOGICAL LOADED

```

C

IF (.NOT.LOADED) CALL GETCOOR

C

IF (IZ.EQ.NZ) GOTO 1

IF (IZ.EQ. 1) GOTO 2

C

```

C Cell not at border
  LFPHH = LOF(HIGH(INDEXV)) + IY + NY*(IX-1)
  LFPHL = LOF(LOW(INDEXV)) + IY + NY*(IX-1)
  IF(II.EQ.1) DPHIDZ = (F(LFPHH) - F(LFPHL))
    & / (2*(ZCOOR(IZ+1) - ZCOOR(IZ-1)))
  IF(II.EQ.2) DPHIDZ = (SQRT(F(LFPHH)) - SQRT(F(LFPHL)))
    & / (2*(ZCOOR(IZ+1) - ZCOOR(IZ-1)))
  RETURN

C
C Cell at HIGH border
  1 LFPHI = LOF(INDEXV) + IY + NY*(IX-1)
  LFPHL = LOF(LOW(INDEXV)) + IY + NY*(IX-1)
  IF(II.EQ.1) DPHIDZ = (F(LFPHI) - F(LFPHL))
    & / (ZCOOR(IZ) - ZCOOR(IZ-1))
  IF(II.EQ.2) DPHIDZ = (SQRT(F(LFPHI)) - SQRT(F(LFPHL)))
    & / (ZCOOR(IZ) - ZCOOR(IZ-1))
  RETURN

C
C Cell at LOW border
  2 LFPHH = LOF(HIGH(INDEXV)) + IY + NY*(IX-1)
  LFPHI = LOF(INDEXV) + IY + NY*(IX-1)
  IF(II.EQ.1) DPHIDZ = (F(LFPHH) - F(LFPHI))
    & / (ZCOOR(IZ+1) - ZCOOR(IZ))
  IF(II.EQ.2) DPHIDZ = (SQRT(F(LFPHH)) - SQRT(F(LFPHI)))
    & / (ZCOOR(IZ+1) - ZCOOR(IZ))
  RETURN

C
  END

```

```

C=====C
C- Subroutine BUOYKE
C
C   Modified on 24 April, 1989.
C
C   BUOYKE calculates the source term Sk/EMUT due to
C   buoyancy. It is used for the KE and EP equations.
C
C   SUBROUTINE BUOYKE
C
C   *COPY SATEAR
C   *COPY GRDLOC
C   *COPY GRDEAR
C   Set dimensions of data-for-GROUND arrays here.
C   WARNING: the corresponding arrays in the MAIN program
C   of the satellite and EARTH must have the same
C   dimensions.
C   COMMON/LGRND/LG(20)/IGRND/IG(20)/RGRND/RG(100)
C   &/CGRND/CG(10)
C   LOGICAL LG
C   CHARACTER*4 CG
C
C

```



```

COMMON/BUOYANT/XCOOR, YCOOR, ZCOOR, BKEEP, LOADED, BUKESO
DIMENSION XCOOR(100), YCOOR(100), ZCOOR(100)
&, BKEEP(100,100), BUKESO(100,100)
LOGICAL LOADED
C
DO 1 IX = 1, NX
DO 2 IY = 1, NY
C
      BUKESO(IY, IX) = 0.0
      IF (NX.EQ.1) GOTO 10
      BUKESO(IY, IX) = BUKESO(IY, IX) + RSG8 * DPHIDX(H1,1)
C
10      IF (NY.EQ.1) GOTO 20
      BUKESO(IY, IX) = BUKESO(IY, IX) + RSG9 * DPHIDY(H1,1)
C
20      IF (NZ.EQ.1) GOTO 30
      BUKESO(IY, IX) = BUKESO(IY, IX) + RSG10 * DPHIDZ(H1,1)
C
30      CONTINUE
C RG(1) = BETA
C RG(2) = CP
C
      BUKESO(IY, IX) = (RG(1) / (RG(2) * PRT(H1)))
& * BUKESO(IY, IX)
C
2      CONTINUE
1      CONTINUE
C
      RETURN
      END

```

```

C=====C
C- Subroutine VARTOF
C
C   Modified on 24 April, 1989.
C
C   VARTOF loads the content of BUKESO in the F-array.
C
C   SUBROUTINE VARTOF(LB)
C
C *COPY SATEAR
C *COPY GRDLOC
C *COPY GRDEAR
C
C   COMMON/BUOYANT/XCOOR, YCOOR, ZCOOR, BKEEP, LOADED, BUKESO
C   DIMENSION XCOOR(100), YCOOR(100), ZCOOR(100)
C   &, BKEEP(100,100), BUKESO(100,100)
C   LOGICAL LOADED
C
C   KK = LOF(LB)
C   DO 10 II = 1, NX

```

```
DO 10 JJ = 1, NY
  KK = KK + 1
10   F(KK) = BUKESO(JJ, II)
C
RETURN
END
```