SUNCODE-PC™

A PROGRAM USER'S MANUAL

Program and Original Manual Written By:
Terry Wheeling and Larry Palmiter

SUNCODE-PC™ Manual Written By:
Mary Jane DeLaHunt

Published and distributed by Ecotope, Inc., 2812 East Madison, Seattle, WA, 98102, (206) 322-3753. For information, address Richard Beckerman, Ecotope, Inc., at the above address.

PROGRAM USERS MANUAL

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. GENERAL INFORMATION

Section page

Section 1-1. PROGRAM SUMMARY

A. OVERVIEW

SUNCODE-PC is a general purpose thermal analysis program for residential and small commercial buildings. It is the microcomputer version of SERI/RES, a mainframe program written for the Solar Energy Research Institute by Larry Palmiter and Terry Wheeling of Ecotope, Inc., Seattle, WA. SERI/RES has been extensively tested by SERI and others using measured building data, and has also been compared with such mainframe programs as DOE 2 and BLAST. It has proven itself in the United States and over 14 foreign countries for over three years. SUNCODE-PC is functionally equivalent to SERI/RES; in addition, a large number of bugs have been corrected, making SUNCODE an updated and enhanced version of SERI/RES.

The method of analysis used in the program is simulation. A thermal model of the building is created by the user. It is translated into mathematical form by the program. The mathematical equations are then solved repeatedly at time intervals of one hour or less for the period of simulation, usually one year. The mathematical representation of the building is a thermal network with non-linear, temperature dependent controls. The mathematical solution technique uses a combination of forward finite differences, Jacobian iteration, and constrained optimization.

The program has an interactive editor for creating building descriptions. While creating a building description the user is continuously prompted with headers that provide the names and units for each data entry. This allows for rapid and error free input. The editor also checks the validity of the input and reports errors as soon as possible. It also provides facilities for storing and referencing several types of building description files.

B. PURPOSE

The purpose of the program is to make accurate thermal analysis of residential and small commercial buildings accessible to those lacking special technical training. In keeping with this goal, the authors have tried to make the program as "user friendly" as possible. Many of these features also benefit experienced analysts.

Until recently, both peak power requirements and annual energy requirements for residences were estimated by rule of thumb. While the defects of such rules were well known, the low cost of energy did not justify further efforts. The cost of heating and cooling a building over its useful life now justifies more accurate analytic methods. Also, increasing interest in quantifying the benefits of passive solar construction has mandated the use of computer-based methods of analysis.

While many computer-based thermal analysis programs are available, most of them are designed primarily for the analysis of large commercial buildings. Many of the assumptions and simplifications used in these programs make them unsuitable or cumbersome for residential and small commercial work. Programs with a residential focus developed at the universities and national laboratories have remained undocumented and tend to require a specialized technical background on the part of the user. At the same time, manual design tools, such as the monthly correlation methods developed at Los Alamos Scientific Laboratory and the University of Wisconsin, are unable to provide answers to many common "what if" questions about building performance. The advantage of simulation is that nearly all such questions may be answered, in principle.

Our purpose has been to provide a design and analysis tool that is easily mastered by a person without special technical training, that gives answers to most of the commonly encountered design questions, that has broad applicability to all types of residential and small commercial buildings, that minimizes the time and effort of data entry, and that provides results with a high degree of accuracy at relatively low cost.

C. INTENDED USE

This program is intended for use by architects, engineers, consultants, building researchers, building code officials, utility analysts, and others interested in the thermal performance of residential and small commercial buildings.

The program is meant to have enough generality to accurately model almost all residential buildings and most small commercial buildings. It is particularly suitable for the analysis of various types of passive solar buildings. Special provisions are made for the analysis of attached sunspaces, thermostatically controlled fans, rockbin thermal storage, and vented Trombe walls.

The program allows the user great flexibility in choosing the level of detail to be used in modeling a building. This allows the program to be used in a quick and cursory way to evaluate general options and, at a later stage, to perform detailed analysis of the final design.

D. APPLICABILITY

Like all thermal analysis programs, this one has many limitations. Some of the limitations stem from deliberate choice, while others were imposed by necessary compromises. Perhaps the major limitation is the lack of detailed treatment of equipment. This is primarily a building loads program. It is designed to simulate the dynamic performance of the building in great detail and report the amounts of energy and power that the heating and cooling equipment must supply in order to maintain comfort conditions. No attempt has been made to simulate the actual performance of particular heating and cooling units. This must be contrasted with programs for the analysis of large commercial buildings, where a majority of the effort is expended on equipment simulation and the building is treated somewhat cursorily. Such an approach is necessary for these buildings because the most pressing question frequently is which types and combinations of equipment will provide the best comfort or the lowest operating costs. In many cases, the building itself is a "given" in the analysis.

In contrast, the energy and power requirements for small buildings are dominated by the performance of the building. This is particularly true for structures designed to maximize solar benefits. The inclusion of performance characteristics for the many kinds of residential equipment would have greatly increased the size and complexity of the program. In addition, the required operating curves for residential equipment are not generally available. Also, in residential work, there is frequently no choice about equipment (for example, whether to use a gas furnace or electric resistance heat), or the choice is made on non-energetic grounds. For these, and other reasons, the authors have chosen to restrict the program to the modeling of building loads. It is not intended for situations where equipment choice is the primary interest.

The present version of the program is not suitable for the accurate analysis of earth-sheltered or underground buildings, although reasonable approximations might be obtained by a sophisticated user. No special provisions have been made to allow accurate analysis of air-envelope or double-shell buildings, although, again, a sophisticated user may obtain reasonable results with the constructs provided. In both cases this would generally require analysis of experimental data to provide the values for the parameters which the program uses. It is beyond the scope of this manual to discuss such techniques.

The program does not model the long-wave or infra-red radiation transfer between surfaces in a room. This simplification is not likely to have any significant effect. It also does not model the actual distribution of incoming solar radiation to the various surfaces in a room, although the user has some freedom to allocate the solar gains among the surfaces. In some solar structures this may be an important effect, and it may prove difficult for the provided user allocation to properly model the situation. Further work with programs that have this capability may clarify those situations where it is important.

The shading of external surfaces and windows is not treated in full generality. The program allows for overhangs of infinite length, with finite depth and vertical offset. It provides for sidefins with infinite length, and finite height and depth. Both of these elements may shade a surface of any tilt or orientation. A provision is made for shading caused by faraway objects that obstruct the skyline. These capabilities will be adequate in most cases, but in a particular case may not meet the user's need. No provision is made for specular or diffuse reflectors.

In addition to the items discussed above, a number of simplifying assumptions and approximations have been made throughout the program. Such simplifications are necessary for two reasons. First, the cost of a first principles analysis would be truly prohibitive, even at today's reduced computer charges. More importantly, it can be shown that in most cases the accuracy of the approximate analysis is adequate for practical purposes. Thus, although the program may answer most questions accurately most of the time, it may not answer a particular question that depends upon the details of the modeling of some building component.

E. USING THIS PROGRAM

This program is written as two independent FORTRAN programs. Each program serves a different purpose. The first program is called the "EDIT module". This program provides an interactive environment designed to assist the user in entering a building description onto the computer. It does most of the file manipulation and serves as a one-point shopping center for the user to interact with the program. It also does extensive error checking to assure that the user's input is consistent. The second program, called the "LOADS module", performs the actual simulation.

Use of the program has been simplified as much as possible. All specifications for the weather files and the exact output desired, including file names, are contained in the building description. A typical simulation requires five steps.

1) Complete the necessary input data forms using the building plans and specifications.

2) Access the EDIT module and create a building description file.

3) Use the EDIT module's compile command (#7) to check for any errors in the building description and create an intermediate file.

4) Run the LOADS module. It will ask for the name of the intermediate file, perform the simulation, and produce the output files.

5) Print the output files or display them on the terminal using the system facilities or the utility program VIEW.COM.

F. DATA FILES

The program uses a number of files:

1. Header Files

   These files contain all of the headers for the building descriptions. There are two of them, one for English units, and one for metric units. These are fixed files, providing a convenient way to store information that would otherwise add greatly to the length of the program.

2. Weather Data Files

   These are the weather files available on disk. The program can read ASCII or binary weather data.

The remaining files are created by the program following the user's instructions. With one exception, the management of these files is the responsibility of the user. These files are discussed in greater detail in Chapter Three, but briefly, they are:

3. Building Files

   These contain a partial or complete building description in ASCII format, allowing them to be displayed on the terminal or printed.

4. Library Files

   These contain the same information as the bilding files but are in a format that can only be read by the EDITS module. They are used as "libraries" of frequently used building components.

5. Intermediate File

   This file contains the machine readable numeric form of the building description. It may include a building file and portions of several library files. It is created by the EDIT module, read by the LOADS module, and automatically deleted at the completion of the simulation.

6. Output Files

These files contain the simulation results. They are produced by the LOADS module and remain on disk until the user disposes of them.

The diagram below shows the general structure of SUNCODE-PC, including the relation of the two modules to the various files.

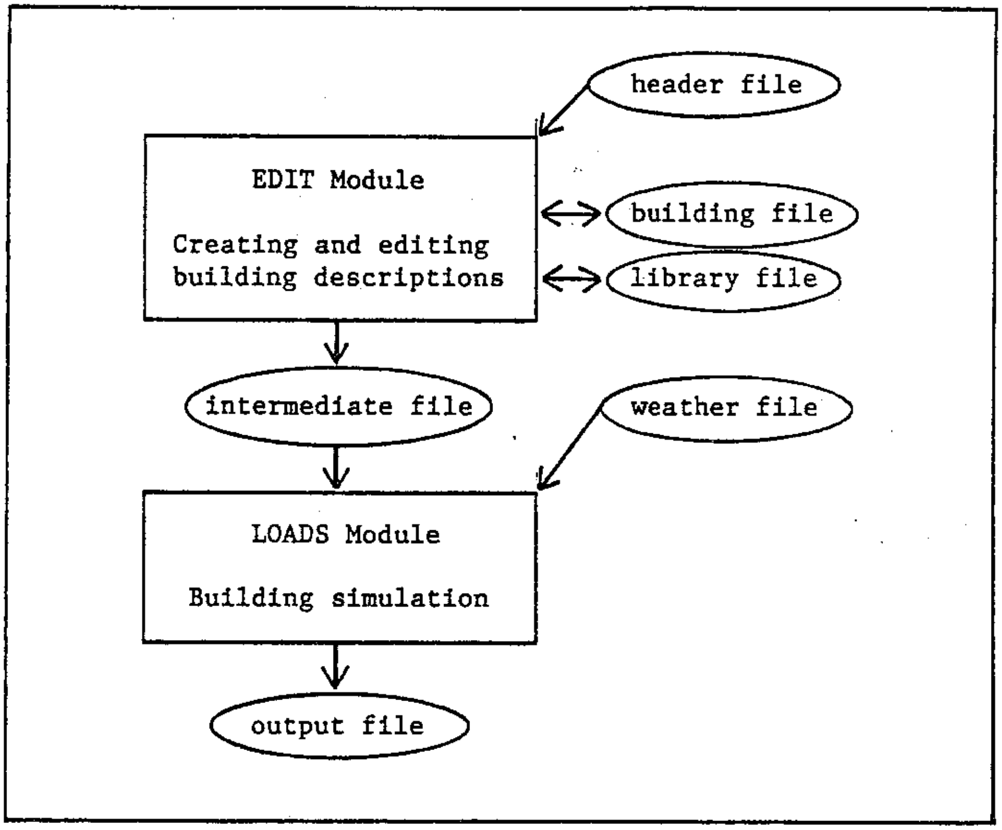


Fig. 1-1. Structure of SUNCODE-PC

Section 1-2. USER MANUAL SUMMARY

A. ORGANIZATION OF THE MANUAL

The first chapter of the manual, GENERAL INFORMATION, provides a concise overview of SUNCODE-PC and gives instructions on installation and operaton of the program, including such information as acceptable weather file formats.

The second chapter, THERMAL MODELING, is meant to serve several functions. The first function is to introduce the user to the descriptive constructs or language that this program uses to describe buildings. The second function is to provide guidance on how to model buildings. The third function is to show where each of the parameters that characterize the building are located in the input data sections. This chapter contains all of the information necessary to allow the user to prepare a building input form.

The third chapter, CREATING AND EDITING BUILDING DESCRIPTIONS, contains detailed instructions on the mechanics of using the editor program to create a building description file, make library files, and create the intermediate files.

The fourth chapter, THE VIEW PROGRAM, contains instructions on use of the utility program VIEW.COM. This program can be used to view or graph SUNCODE output, or to create output files which may be read into other programs (e.g., spreadsheets).

The fifth chapter, SOFTWARE MODIFICATION, presents technical information useful for modifying the program's source code.

The sixth chapter, TECHNICAL ALGORITHMS, presents the algorithms used by the LOADS program in mathematical form.

In addition to these chapters are several important appendices. The first, INPUT DATA SECTIONS, contains the headers, units, and a brief description of all input parameters. It also contains a table of unit abbreviations and conversions for the input parameters.

The second appendix, OUTPUT DATA SECTIONS, contains the headers, units, and a brief description of all output data sections.

The third appendix, INPUT DATA FORMS, contains blank input data forms that may be reproduced for use with the program.

The fourth appendix, OPTIONAL LOADS PROGRAM OUTPUT FILE, contains information on the unformatted output option.

B. SUGGESTED USE OF MANUAL

The beginning user should first read Chapter One in its entirety. Then read the chapter on Thermal Modeling, referring to the Input Data Sections in the appendix as the parameters of each data section are discussed. The user may also wish to review the preliminary stages of each of the examples in that chapter. The user should then try to complete the input forms for a very simple building. If this is difficult, review the thermal modeling chapter and try again. A little practice should make this a straightforward task.

After the user is comfortable with preparing the input forms, read the Editing chapter. After reading it through once, try using the editor to enter a simple building description into the computer. Experimentation with the various commands of the editor will allow the user to quickly master the various features. After completing a building description, use the command #7 (compile) to check the input for errors and, if desired, complete the simulation. Simulations with a short duration, perhaps several days, will provide useful feedback more quickly.

The program has been designed in such a way that the experienced user will rarely need to refer to the manual. Such a user will find that an occasional reference to the Input Data Sections appendix will be adequate during the preparation of the input forms. Once the editor has been used a number of times, the built-in prompts should be adequate feedback. Many users will find it possible to simply create the building descriptions directly with the editor, although such composing at the terminal is not recommended.

SECTION 1-3 INSTALLING SUNCODE-PC

This section contains the information necessary to install SUNCODE-PC on your computer.

A.SYSTEM CONFIGURATION

To run SUNCODE-PC 5.71 or previous a version, the following system configuration is required:

256K memory
math coprocessor
DOS 3.3 or later
ANSI.SYS device driver

To run SUNCODE-PC version 6.0 or later, the following system configuration is required:

2MB memory
386SX or better
DOS 3.3 or later
ANSI.SYS device driver

B.INSTALLING SUNCODE-PC

To install SUNCODE-PC, follow these steps:

1. Make a working copy of the SUNCODE distribution disks and any weather disks you received. File the original disks in a safe location.

2. Copy all files from disk 1 to a subdirectory of your hard disk. Disk 2 contains files relevant only to programmers and is not generally useful, so you do not need to copy the files from it. If you are going to run SUNCODE-PC from a floppy, disk 1 contains the files you need. You will need a second floppy drive to store output files.

3. Set up the ANSI.SYS device driver, if it is not already set up. The ANSI.SYS driver allows the cursor keys on the keyboard to be configured. The EDIT program will not run correctly unless the ANSI.SYS driver is installed when the computer is booted. To check whether ANSI.SYS is installed, look at the file CONFIG.SYS in the root directory on your default disk drive. This file must contain the command:

```
device=ansi.sys.
```

If the command is not present use the DOS EDIT program or Windows NOTEPAD to add it to the file.
Note that the file ANSI.SYS need not be on the same drive or in the same directory as the file CONFIG.SYS. You may use drive designations or directory pathnames in the command, for example: device=C:\bin\ansi.sys. Once the change is made you must reboot the computer.

4. If you would like to work with English unit inputs, copy the file SUNCODE.ENG to SUNCODE.H. If you would prefer Metric units, copy the file SUNCODE.MET to SUNCODE.H. This sets the SUNCODE-PC headers to the correct units convention.

SECTION 1-4 RUNNING SUNCODE-PC

This section contains instructions on how to run Suncode-PC. A sample problem is used as an example and additional useful information about SUNCODE operation is also presented.

A.RUNNING SUNCODE

SUNCODE consists of 2 DOS programs, EDITS.EXE and LOADS.EXE. Each is run from the DOS operating system prompt, or if you are using WINDOWS or some other GUI interface, you can run the programs from within the interface just as you would any other DOS program. EDITS helps to create and edit building descriptions, and serves as a pre-processor. The pre-processing function is referred to as compiling. To create, edit, or compile building descriptions, type or select EDIT.BAT at the system prompt. This batch file sets up and runs EDITS.EXE. Once the building description has been created, compiled, and saved, exit EDITS. The simulation can now be run by typing or selecting LOADS at the system prompt, and when asked for the input filename, entering the name of the compiled building description. Output will be written to the file you specified during compilation. For more detailed information, read through the following section on running the sample problem.

B.RUNNING THE SAMPLE PROBLEM

The sample problem serves several purposes. First, it illustrates most (but not all) features of SUNCODE. Second, the specified output format illustrates the flexibility of output allowed by the program. Third, running the sample program will help familiarize you with SUNCODE-PC modules and commands. Finally, the comprehensive output provides an excellent benchmark to ensure proper operation of the program and to benchmark execution time.

The files relevant to the sample problem are listed below:

SAMPLE.BLG	Sample building description (English Units)
SAMPLE.OUT	Output from sample problem (English Units)
METRIC.BLG	Sample building description (Metric Units)
METRIC.OUT	Output from sample problem (Metric Units)
DILLON.BIN	Weather file (binary format) specified in sample building description

The building description files and weather file should be in your SUNCODE directory if you followed the installation instructions, or on disk 1 if you are running SUNCODE-PC from the floppy drive. If you are using DOS, specify the Suncode disk or directory to be the current DOS directory by typing CD <drive>:<directory> (CD c:\suncode, for example). You are ready to start. The following paragraphs will walk you through setting up and running the sample problem. If you have set up SUNCODE-PC with english units then follow the instructions exactly. If you have set up SUNCODE-PC with metric units, substitute METRIC for SAMPLE in the instructions.

Running EDIT

If you are at the DOS prompt, type EDIT and carriage return (<return>). If you are using WINDOWS or another GUI run or select the EDIT.BAT program in the same way you run DOS programs. The SUNCODE EDITS Master Menu should display on the screen. Select command #2 (Read Building File) by typing the command number (2) and <return>.

The program will prompt:

```
ENTER FILE NAME (RETURN TO CANCEL) [    .BLG]    FILE:
```

Enter:

> SAMPLE <return> (or METRIC<return> if you specified metric units during setup).

No extension need be specified since .BLG is a default extension. Drive and directory information can be specfied, however SUNCODE-PC has a 12 character limit to filenames which greatly limits length of any path name.

The building description file label will be displayed and when you are asked to verify that the file is the correct one. Type, Y <return>. The program will indicate that it is loading the building file into the EDITS workfile. When it is done, you will be asked to press <return> to continue. The Master Menu will then be redisplayed.

The sample building does not need to be created or edited. Select command #7 (Compile Intermediate File) and <return>. This command transforms the building file into a file that the LOADS module can read. It is a necessary step in the simulation process. The program will prompt:

```
ENTER LIBRARY FILENAMES (CARRIAGE RETURN FOR NONE)
```

Type <return>. It will then prompt:

```
ENTER INTERMEDIATE FILENAME [sample.INT]
```

Sample.INT is displayed as the default filename. SUNCODE-PC combines the name of the input building description with the intermediate file default extension to get this default name. To us the default, type<return>. If you wish, you may change the name by typing a new name. If you do this, specify a ".INT" extension, or no extension at all, which will cause SUNCODE-PC to use the default extension. The next prompt will be:

```
ENTER SUMMARY OUTPUT FILENAME [sample.OUT]
```

The filename you see will be that of the intermediate file with the default ".OUT" extension added. If you wish, you may change the name by typing a new one, otherwise, press <return> to accept the default. Once the output filename is entered, the program will check the building description for errors. If there are no errors then the user will be asked whether the minimum number of "time steps per hour" should be used. The minimum value is calculated based upon the node placement of the building description. For the sample problem the minimum number is four. The user can press <return> to use the minimum value, or enter a greater number. For this sample just press <return>.

When you return to the Master Menu select command #1 (Terminate the Program) and <return> to quit the EDIT module. You will be asked whether you want to save the contents of the work file. Since we read it from disk and have made no changes, answer no. You will return to the operating system.

Running LOADS

At the DOS prompt type: LOADS <return>. Again, if you are using WINDOWS or another GUI run LOADS.EXE in the same way you run other DOS programs. The SUNCODE-PC copyright will be displayed and you will be prompted with:

```
ENTER INTERMEDIATE FILE [.INT]
```

Enter the name of the intermediate file you created in EDITS The name need not include the extension and can be no longer than 12 characters

The output file was specified during the compilation process and does not need to be repeated here. LOADS will create this file automatically. Weather files specified in the building description will also be opened. The LOADS module will then begin the simulation. You will know that the simulation is proceeding when you begin to see

```
 1 ......
```

on the screen. The 1 stands for January; the dots appear on the screen as a day's worth of simulations are begun. When the program has completed it you will be returned to the operating system.

Comparison Timings - Sample.blg	
Hardware(version)	Seconds
IBM PC/XT,8087	75.0
IBM AT,8287	40.0
VAX 780	9.0
25 MHZ 386 (v5.71)	5.5
DEC System 20	4.0
25 MHZ 386 (v6.0)	3.2
VAX 8600	1.5
33 MHZ 486 (v5.71)	1.5
33 MHZ 486 (v6.0)	1.2
66 MHZ Pentium(v5.71)	0.4
66 MHZ Pentium(v6.0)	0.3

The sample problem output you've produced may be compared with the SAMPLE.OUT file on disk 2 (if you ran the metric building then compare to the metric.out file). This can be done by reading the files into a text editor or by printing them. Results should be identical. However, you need not produce a full years's output; a month's worth of output is probably sufficient to verify correct operation of the program.

Modifying the Building Description

To practice modifying a building description in EDITS, we will change the run length from 1 year to 1 month. Return to EDITS and once again, select command #2 and read the sample building description into memory.

When the Master Menu has reappeared, select command #6 (Display/Edit Building). The Display/Edit Menu will appear. This is a good time to read Section 3-3 (Screen Editing) for information on displaying and editing building files.

To specify a simulation duration of one month rather than one year, edit the RUNS section of the building description. START should remain JAN 1., STOP should be changed to JAN 31. This change tells the LOADS program to read only January's weather data and produce output results available for that month. To make the change, do the following:

1. Choose building section #2 (RUNS) and <return>.
2. Type E (for edit) and <return>.
3. At the double arrow prompt (>>), type the following line:
   ,,,,,,JAN 31. <return>
4. If the STOP date is not JAN 31, repeat steps 2 and 3. If you get an error message, repeat step 3.
5. When you have made the change, type Q (for quit) and <return>. Then return to the master menu.

C.BUILDING DESCRIPTION HEADER FILES

EDITS uses the file SUNCODE.H as a header file. Information in this file is used as the header information displayed in EDITS, and during error checking and compilation. The default SUNCODE.H file is initially configured for English units. To change to metric units, SUNCODE.H must be replaced with a different header file. The file SUNCODE.MET contains the Metric units header file; SUNCODE.ENG contains the English units header file. To change units, copy the desired file to SUNCODE.H.

The header files can be edited to create custom header files. This may be desireable if users of the program speak a language other than english. To create custom headers, edit a copy of a current header file using a text editor in over write mode. It is important that the overall line length of lines in this file do not change (currently 80 characters). Each line starts with six characters, numbers or blanks, which should be left alone. The seventh character of each line is where changes can start. The last line (END OF FILE) of the header file should not be changed.

The first line, (starting at the seventh character) contains the default building description label which is used when you first save a file from the editor. Maximum length for the label is 70 characters.

After the first line, there is a data for each section of input. Each section has a section label, header information, and default values spread over several lines. The section label or name is the first line of each section. It can be up to 14 characters long and must be unique compared with other section names. Remember, the first 6 characters of each line should not be changed. The two parameters LINES and HEAD are not used so do not change those entries.

The remaining lines in each section contain the header information, data type information and default values. Each of the header lines starts for an asterisk in the 7th column and is used strictly to describe the input to the user. There can be a maximum of 5 lines per section, and a maximum of 74 characters per line including the asterisk. Any of the data (after the 6th column) can be changed on these lines.

Following the header labels is a row with data type information, and lastly a row with default values. The lengths and precision of *numeric* and *scheduled* input can be modified by changing the mask pattern (XXX.X) on the data type row. Be careful that the data masks are always seperated by spaces, and be sure to not change the length of any character inputs (AAA).

Header files are not always compatible between versions. You should generally always use header files that come with the version of SUNCODE you are running. If you have created a custom header file which you wish to use with a new version of SUNCODE, compare the original header files to see if there are any differences. Edit your header file to reflect these differences and then carefully test the file. Incompatibilities arrise from changes in the lengths of character variables, which cause shifts in the memory location of variables.

D.FILENAMES

Filenames are specified in the documentation as <filename.extension>. Default extensions are provided for many types of files. These files are specified as <filename>.EXTENSION. The following default extensions are provided:

.BLG	ASCII building files
.LIB	Library building files
.INT	Intermediate files
.OUT	Formatted output files
.UNF	Unformatted output files
.DAT	ASCII weather data files
.BIN	Binary weather data files

Filenames can be a maximum of 10 characters in length in versions previous to 5.71, and 12 characters after. Drive designations and MS-DOS 2.0 pathnames are acceptable and are included in the 10 or 12 character limit. The default extensions are not included.

E.CONFIGURATION OF CURSOR KEYS

The EDIT.BAT file uses the ANSI.SYS driver to configure the cursor keys for the EDITS program. Cursor Up will move to the previous entry line, Cursor Down will move to the next data section, Home will activate the Add command for data entry, and End will do the Quit command. All of these commands (as well as all other commands) can also be typed as before.

If EDITS.EXE is run directly (i.e., if the command EDITS is entered rather than EDIT) or if the ANSI.SYS driver is not installed, then the cursor keys will not be configured. In this case, move the cursor up with caret (^) and return, and move the cursor down with return.

F.UTILITY PROGRAMS

Three utility programs are included on the SUNCODE disks:

VIEW.COM -- This enables the user to quickly access and view SUNCODE formatted output files. This program is capable of displaying output files on the screen (allowing the user to move easily from one section of output to another), producing simple graphics, and producing ASCII files for "importing" into other programs (particularly spreadsheets) for more extensive graphics and analysis. The menus of VIEW are explained in Chapter 4.

TMY2BIN.EXE -- This is a FORTRAN program that converts ASCII weather data to binary files. Binary files use less than half the space of compressed ASCII files, and they are over twelve times smaller than original TMY weather files. The program can read ASCII integer files in two formats:

Original TMY format	(SUNCODE Data Type 1)
Compressed ASCII format	(SUNCODE Data Type 4).

The binary file produced is a SUNCODE Data Type 2 file. This program is discussed in further detail in Section H (WEATHER) below.

CHECKINT.EXE -- This enables the user to examine the contents of intermediate files created by the EDIT module and is mainly useful to programmers.

G.BATCH RUNS

When many buildings need to be modelled, it can be tedious to run EDITS and LOADS for each one. The Suncode-PC Loads program has the ability to run multiple simulations. Suncode-PC also comes with a complete set of DOS batch files to implement batch processing of the EDITS compile function, and to streamline the LOADS batch capabilities.

The native LOADS batch abilities can be accessed by creating a text file named SUNCODE.Q which contains the names of all the intermediate files to be run (compiled output from the EDIT module), one filename per line. LOADS automatically checks for SUNCODE.Q, and if it is present, asks whether to run the files specified there. If the user reponds yes, then Suncode-PC automatically reads SUNCODE.Q, and runs each building listed. SUNCODE.Q must be on the same disk drive as the LOADS program.

Suncode-PC also comes with a complete set of DOS batch files to implement batch processing of the EDITS compile function, and to streamline the LOADS batch capabilities. These require DOS 5 or higher. Other brands or versions of DOS may require modifications to these batch files. If you make modifications that you would like to share please send them to Ecotope.

Batch processing of the EDITS compile function is convienent when a large number of variations are being explored. It is cumbersome to move back and forth between the EDITS Display/Edit Menu and the Master Menu, for each change. It is far easier to create, debug and save a building description in EDITS, and then exit EDITS and use a text editor to create the variations. Each variation is saved to a unique file and then the batch compile command used to compile the descriptions. Any text editor or word processor that can read and write ASCII text files can be used to edit building description files. Line oriented editors used by programmers with block and column moves provide the most functionality. The DOS or Windows editors will also work.

Two DOS batch files are included to facilitate batch compilation: compile.bat and compblgs.bat. COMPILE.BAT is set up to compile a single building description. The building description filename should be specified on the command line without the ".blg" extension (i.e. COMPILE sample). Compile.bat uses the DOS pipeline feature to feed the appropriate key strokes into the EDITS program to compile the building. COMPBLGS.BAT is set up to compile one or more building descriptions. The building description filename, which can include DOS wild card characters, should be specified on the command line. Again, without the extension (i.e. COMPBLGS sam*). COMPILE.BAT is called for each building description file matching the file specification.

If there are errors in the building description, COMPILE.BAT will not feed EDITS the correct keystrokes and the user will be left in EDITS. The appropriate changes can then be made and compiled, and upon exiting EDITS, the process will continue. For this reason it is best, before making several variations on a theme, to first compile the base building description. This will insure there are no major errors. Additionally, since COMPILE.BAT accepts the minimum number of time steps, even if this is several hundred, it is a good idea to verify that the minimum number of time steps required by the base description is acceptable from a computation time stand point. If it this is unacceptably high, modifications can be done to the single base building description.

A DOS batch file has also been included to simplify batch operation of LOADS. RUNINTS.BAT is set up to run one or more building descriptions. The intermediate filename, which can include DOS wild card characters, should be specified on the command line. All filenames matching the file specification are automatically put in a SUNCODE.Q file and LOADS is run. An additional batch file, RUNALL.BAT, is set up to compile and run one or more building descriptions. It combines the functions of COMPBLGS.BAT and RUNINTS.BAT.

H.WEATHER

The weather data required by SUNCODE consists of hourly data for the following five variables: direct normal solar radiation, total horizontal radiation, ambient temperature, dew point temperature and wind speed. Data may be in English or metric units. The weather file to be used is specified in the EDIT module as part of the building description, and is read by the LOADS module during the simulation. Four weather data formats can be used in this version of SUNCODE:

DATA TYPE 1 -- Original NOAA TMY weather data file format (size: 1095K)Data is in metric units, and is stored in integer format in 132 character records. When the LOADS module reads in weather data from this type of file, columns not containing the required variables are skipped. Field widths and units for each of the relevant variables are shown in Table 1-1. The LOADS format statement for reading this type of file is: (24X, F4.0, 26X, F4.0, 45X, F4.1, F4.1, 3X, F4.1, 14X)

DATA TYPE 2 -- Binary weather format (size: 86K) The program TMY2BIN.EXE (supplied with SUNCODE) converts ASCII files (Data Types 1 or 4) to binary sixteen bit two's complement integers (Microsoft FORTRAN INTEGER*2). The input ASCII file may be in English or metric units. LOADS then reads the binary file, converting the integers to real numbers by dividing by 10 where appropriate (if English units, all integer values are divided by 10; if metric units, all integer values except radiation are divided by 10). The weather files supplied with SUNCODE are in this format.

DATA TYPE 3 -- Free format data The LOADS program is also able to read weather files that are not in integer format. data in free format files should be stored as real numbers with 5 values per record in the order specified in Table 1-1. Values should be separated by one or more spaces. The data should be stored in the actual units to be used in the LOADS module. For example if LOADS is to use degrees F for temperature, then temperature in degrees F must be stored in the weather file. Either English or metric units may be used.

DATA TYPE 4 -- Compressed ASCII format (size: 205K) This type of file contains only the five relevant variables from the TMY data. The data is stored in integer format and may be in English or metric units. Field widths and units for each variable are shown in Table 1-1. The LOADS format statements for reading this type of file are: (2F4.1, 2F5.1, F4.1) for English units; (2F4.0, 2F5.1, F4.1) for metric units.

If you are creating your own weather files, you have several format options:

1. Create a free format file (Data Type 3) using real numbers and actual units.
2. Create a compressed ASCII file (Data Type 4) using integer values and units as shown in Table 1-1. This file can be read directly into LOADS or it can be converted to binary form with the TMY2BIN program.
3. Change the format statement in TMY2BIN to read your weather file. This is a simple task, but requires a FORTRAN compiler. Remember that LOADS requires that the variables be in integer format and in the order and units shown in Table 1-1.

Table 1-1 Weather File Formats

	Data Type 1		Data Type 4	
	Original TMY		Compressed ASCII	
Variable	Units	Field Width	Units	Field Width
1.Direct Normal Radiation	KJ/M2	4	BTU*10/FT2 KJ/M2	4
2.Total Horiz. Radiation	KJ/M2	4	BTU*10/FT2 KJ/M2	4
3.Ambient Temperature	C DEGREES*10	4	F DEGREES*10 C DEGREES*10	5
4.Dew Point Temperature	C DEGREES*10	4	F DEGREES*10 C DEGREES*10	5
5.Wind Speed	M*10/SEC	4	MILES*10/HR	4

NOTE: If the sample problem and/or Ecotope supplied weather files ran without errors, but you are having trouble with weather files you created yourself, the following tips may be useful. Editing or tranporting a weather file may leave garbage characters at the beginning or end of the file. This is particularly likely to occur if the file has been downloaded from a mainframe computer. The presence of garbage may cause the LOADS program or the TMY2BIN program to abort, or it may produce erroneous output (the ambient summary section of the output is a good place to check results). If you are having problems, use a text editor that shows control characters and carriage returns (e.g. WordStar or PCWrite) to look for extraneous characters or carriage returns (simply typing the file will not reveal control characters). FORTRAN will not have problems with <control Z>, which it interprets as an end of file flag, but may have difficulty with any other control characters or extra carriage returns at the beginning of the file. The text editor may be used to edit the weather file, and that should remove the problem.

I.BUILDING COMPONENT LIMITS

SUNCODE-PC limits the number of building components that may be entered in each data section of the building description. To facilitate development of libraries of components, the total allowable number of entries for a data section is larger in some cases than the maximum number of components that can be used in any one run. The editor also limits the total number of data lines which can be entered in a single building description file, and the number of nodes which can be used in a single run. The limits for each building component are shown in Table 1-2. These limits should be sufficient to model most buildings. However, they may be modified by the sophisticated user with access to a FORTRAN compiler to accomodate specific situations. Instructions for doing this can be found in Chapter Five.

Table 1-2 Building Component Limits

	Suncode-PC Version			
Data Section	5.71 and before		6.0 and later	
	Referenced	Entered	Referenced	Entered
Runs	10	10	10	10
Zones	10	10	50	50
Interzone	20	20	20	20
Windows	20	20	100	100
Walls	40	40	200	200
Trombes	5	5	5	5
Fans	5	5	5	5
Rockbins	5	5	5	5
Surface.Types	10	65	40	395
HVAC.Types	10	65	10	395
Trombe.Types	5	65	5	395
Wall.Types	20	65	20	395
Mass.Types	20	65	20	395
PCM.Types	20	65	20	395
Glazing.Types	10	65	10	395
Bin.Types	5	65	5	395
Fan.Types	5	65	5	395
Overhang.Types	5	65	10	395
Sidefin.Types	5	65	10	395
Profile.Types	5	65	5	395
Outputs	10	10	10	10
Schedules	40	65	100	395
Seasons	20	65	20	395
Parameters	5	65	5	395
Stations	10	65	10	395
Total Data Lines	500	500	750	750
Total Nodes	200	200	600	600

J.REPORTING SOFTWARE BUGS or MODIFICATIONS

User's encountering problems which appear to be software bugs should report them to Ecotope. User's who make modifications to the program or create auxilliary tools are also urged to report them to Ecotope. Please submit a written description of the problem, along with the date, time and size of the edits.exe, loads.exe and suncode.h files you are using. If any of the files have been modified please describe the modifications. A disk should be included with building descriptions and output demonstrating the problem, and if there is space, the edits.exe, loads.exe, suncode.h and weather files.

Send the above materials to:

Suncode Bug Report
ECOTOPE, INC.
2812 East Madison Street
Seattle WA 98112
USA

email: ecotope@eskimo.com

Table 1-2 Building Component Limits

DATA SECTION	Maximum number of entries used in any one run	Maximum number of entries that may be entered
Runs	10	10
Zones	10	10 20
Interzone	20	20
Windows	20	20
Walls	40	40
Trombes	5	5
Fans	5	5
Rockbins	5	5
Surface.Types	10	65
HVAC.Types	10	65
Trombe.Types	5	65
Wall.Types	20	65
Mass.Types	20	65
PCM.Types	20	65
Glazing.Types	10	65
Bin.Types	5	65
Fan.Types	5	65
Overhang.Types	5	65
Sidefin.Types	5	65
Profile.Types	5	65
Outputs	10	10
Schedules	40 (schedules)	65 (lines)
Seasons	20	65
Parameters	5	65
Stations	10	65

CHAPTER 2. THERMAL MODELING OF BUILDINGS

Section 2-1. GENERAL

The thermal behavior of buildings depends in a complex way upon many interrelated factors. The engineering analysis of such a complex dynamic system is always a compromise between accuracy and cost. Using a greater level of detail generally improves the accuracy of the results and entails greater costs. The choice of an appropriate level of detail is an important aspect of practical analytic work.

Just as the authors of a program must strive for an appropriate level of detail in developing the equations in the program, the user must also represent the building with an appropriate level of detail. All thermal programs for buildings allow the user great flexibility in the level of detail with which the building is described. Choosing the appropriate level of detail requires a certain amount of engineering judgment--that is, knowledge of what is important and what is not in the solution of a particular problem. It is important to note that simulation of a building using different levels of detail is one of the best ways to develop such engineering judgement.

Within the limits of capability of a given program, the required detail in the building description depends primarily on the nature of the desired output. If the desired output is the annual heating load of a modestly insulated building located in a cold climate, a greatly simplified building description will provide quite accurate results. However, a much more detailed description will be necessary if the desired outputs are accurate hourly zone air temperature profiles of a multizone structure with thermostatically controlled fans between certain zones.

Thus, in order to use a building analysis program, the user must create a thermal model of the building using the constructs of the program. The program receives the user's thermal model and internally converts it to a mathematical form suitable for solution of the problem. Experience has shown that most of the differences in results obtained in the use of different programs to analyze the same building can be traced to differences in the user-created thermal model rather than differences due to the internal solution techniques.

This chapter presents the basic descriptive constructs provided by this program for developing the thermal model of the building. It should also serve as a helpful tutorial in the choice of a thermal model appropriate to the user's purpose. Some guidance is provided for levels of detail in modeling. It is not intended to be a text on thermal modeling, and, of necessity, much interesting and useful information must be excluded.

Section 2-2. THE BUILDING DESCRIPTION

This program is organized around the major thermal components or heat flow paths of a structure. The fundamental concept is that of a thermal zone. A zone is either a room, or group of rooms that operate at the same temperature. The zone temperature of an internal zone is a conductance-weighted average of surface temperatures. There are two special zones, one called AMBIENT and one called GROUND. The AMBIENT zone is outdoor air temperature, while GROUND is a user defined temperature. Conceptually, a building is represented as one or more zones with thermal communication between one another and the outdoor temperature and solar radiation.

The most common paths of thermal communication are walls, windows, and infiltration. Other paths of thermal communication include fan-forced convection, special storage elements such as rockbins, and special types of walls such as Trombe walls and walls made of phase change materials. In addition, the user must provide equipment specifications and schedules and details of the components of the major heat flow elements.

For instance, a simple building could be represented as shown in Figure 2-1. A single zone is connected by four walls. There is also an infiltration heat flow path. This simple conceptual model of the building is then developed into a building description for use by the program by a process of specifying the major features (that is, one zone and four walls), and then providing the necessary details for each major element. For instance, after describing the walls, one would first specify the layers that comprise each wall and then the properties of the materials that comprise each layer.

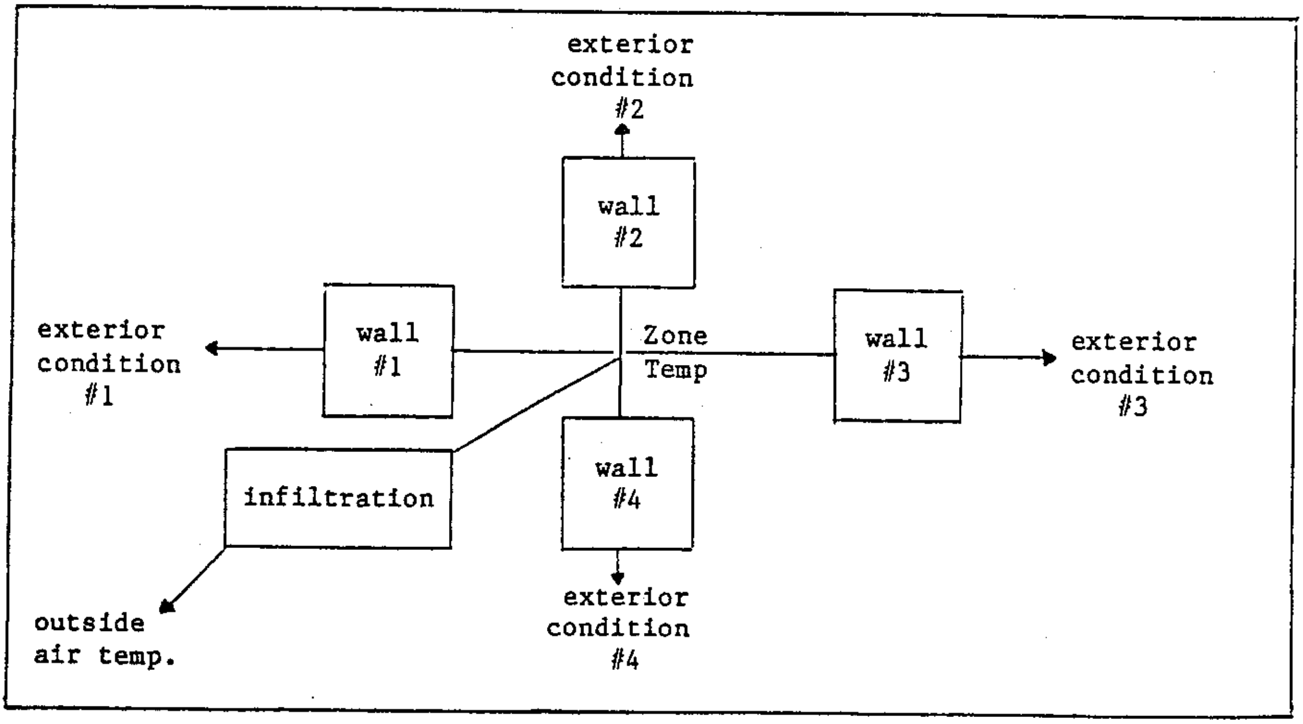


Fig. 2-1 A Simple Building

A building description is composed of several data sections, each of which may contain one or more lines of data values. Each section is a group of parameters describing a particular thermal component of the building. A complete list of the names of all data sections, with a brief description of the information contained in each, is given on the next page. Note that some sections require at least one parameter value to be entered, while others may be skipped entirely if they are not relevant to the problem in hand. The Input Data Section tables in the appendix contain a complete description of all data sections. Refer to the tables throughout the discussion of thermal modeling.

A workable procedure for defining a building is the seven-step process below. Given the flexibility of the program and the diversity of applications, users will tend to develop their own style eventually.

STEP 1 Specify the location and duration of the run(s) by entering the parameters of the RUNS section.

STEP 2 Define the zones of the building by entering the parameters of the ZONES section.

STEP 3 Specify the heat flow paths between zones by entering parameters in the INTERZONE, WALLS, FANS,and ROCKBINS sections.

STEP 4 Define the orientation, size, and shading of the exterior surfaces of the building by entering the parameters of the SURFACES section.

STEP 5 Specify the elements which compose the surfaces by entering the parameters for the WINDOWS, WALLS, and TROMBE.WALLS sections.

STEP 6 Define the components for each of the component sections referenced in the Primary sections above, or incorporate them from other files.

STEP 7 Enter the parameters for the OUTPUT and/or PARAMETERS sections, or incorporate them from a library file.

Guidelines to consider for each of these steps are discussed below.

Table 2-1 List of Data Sections

DATA SECTION NAME	INFORMATION CONTENT	CATEGORY	APPEND. REF.NO.
RUNS	location and duration of simulation run	Primary-Req.	1
ZONES	defines building zones	Primary-Req.	2
INTERZONE	heat flow and solar transfer between zones	Primary-Opt.	3
WINDOWS	location and size of windows	Primary-Opt.	4
WALLS	location, size, and type of walls	Primary-Opt.	5
TROMBE.WALLS	location and size of Trombe walls	Primary-Opt.	6
FANS	location and type of fans	Primary-Opt.	7
ROCKBINS	location and type of rockbins	Primary-Opt.	8
SURFACES	orientation and size of exterior surfaces	Component	9
HVAC.TYPES	properties of HVAC equipment	Component	10
TROMBE.TYPES	detail description of Trombe wall types	Component	11
WALL.TYPES	single or multi-layered wall types	Component	12
MASS.TYPES	wall material properties	Component	13
PCM.TYPES	phase change material properties	Component	14
GLAZING.TYPES	glazing material properties	Component	15
BIN.TYPES	detail description of rockbin types	Component	16
FAN.TYPES	detail description of fan types	Component	17
OVERHANG.TYPES	dimensions of overhangs	Component	18
SIDEFIN.TYPES	dimensions of sidefins	Component	19
SKYLINE.TYPES	specification of skyline shading	Component	20
OUTPUT	output definition	Primary-Opt.	21
SCHEDULES	multi-season/24-hour operation schedules	Component	22
SEASONS	duration of seasons of year	Component	23
PARAMETERS	detailed simulation run parameters	Component	24
STATIONS	definition of weather data files	Component	25

Section 2-3. RUNS

A. GENERAL

Parameters related to the building location are entered in the RUNS data section. The weather data to be used for a run is entered under STATION NAME (the corresponding weather file must then be described in the STATIONS data section). In addition, the user enters the simulation START date and STOP date for each run.

Note that more than one run can be specified at one time. For instance, the same building can be simulated in several locations by specifying several STATION NAMES.

B. GROUND REFLECTANCE

When short-wave solar radiation strikes the ground, it is reflected diffusely. The fraction reflected may vary from about .1 for extremely dark surfaces to .7 or more for freshly fallen snow. This effect is modeled using the GROUND REFLECTANCE value(s). The user may enter either a single value to be used for the entire run or the name of a schedule of monthly values. Typically, a constant value of .2 or .3 is used.

C. GROUND TEMPERATURE

This program provides for the use of a GROUND TEMPERATURE node. It is used in a similar fashion as the ambient air temperature node. Walls, rockbins, and zone conductance coefficients may be connected to the ground node by use of the keyword GROUND. The ground temperature is either a constant annual value or a schedule of values.

The temperature of the earth at depths of ten feet or more is typically a few degrees above the mean annual air temperature for the location. Three-dimensional heat transfer occurs through building surfaces in contact with the earth. Although there are a number of two-dimensional computer programs that approximate the three-dimensional case, none of the currently available building simulation programs have this capability.

The GROUND node is provided as a crude approximation to allow some assessment of earth heat transfer. The most important effect is the two or three month delay in the response of thick layers of earth to changes in ambient air temperature. Thus, in the early winter the ground at basement floor level is still warm, while during the summer the earth absorbs heat from the building.

The GROUND node is really only a user-defined temperature that is capable of being scheduled. When using the GROUND node, the surface heat transfer coefficient on the ground side of the wall should be specified as the reciprocal of the desired pure resistance between the last mass layer and the ground node. If no resistance is desired, the coefficient may be set to its maximum value: 99.999.

D. OTHER

A skyline profile may be referenced by entering the name of a SKYLINE profile. This must be defined in the SKYLINE.TYPES data section. For further discussion, see the section on solar radiation in this chapter.

The column PARAMETER TYPE contains convergence criteria and other run control parameters. The default value is <NONE>. This causes the program to use hard-coded values for the various convergence criteria used in the numerical solution. In nearly all cases the default values will be used; however, for the unusual case the user can create new convergence criteria by entering a PARAMETER TYPE and defining it in the PARAMETER.TYPES data section.

Section 2-4. ZONES

A. USING MULTIPLE ZONES

The program allows the user to model a building as either a single zone or multiple zones. The decision as to whether multiple zones are necessary depends primarily on the specification of heating, venting, and cooling setpoints. If two zones are to be operated at different temperatures during parts of the year or one of the zones (perhaps a sunspace) is uncontrolled, the use of multiple zones is necessary. Attics, basements, and crawl-spaces may also be modeled as additional uncontrolled zones. A little experimentation on the part of the user will soon reveal those cases in which a more complex multi-zone description is desirable.

The user must enter the names of the ZONES and the names of their HVAC TYPES in the ZONES data section. See the section in this chapter on "Equipment" for details.

B. INFILTRATION RATE

Heat gain or loss due to wind- and temperature-induced infiltration of outdoor air is a major element of the overall heat transfer in a typical residence. Infiltration effects are handled in two different ways. The simplest method is to specify a constant or scheduled rate of air flow expressed as air changes per hour for each zone. A second level of detail provides for a calculated rate based on inside to outside temperature difference and wind speed during each hour.

The user must enter a FLOOR AREA and CEILING HEIGHT for each zone. These are multiplied to obtain the zone volume upon which the air changes per hour are based. A numeric constant or the name of a user-defined schedule is entered under INFILTRATION RATE.

C. INTERNAL GAINS

Internal gains are an important factor in residential thermal modeling. The user enters either a constant rate or the name of a user-defined schedule under INTERNAL GAINS. Recent studies indicate that for typical residences the use of constant rates gives satisfactory accuracy for annual heating and cooling loads.

D. LATENT GAINS

Latent heat is the heat required to evaporate or condense water vapor in a zone. It is of primary importance for air conditioning calculations where the condensation of the vapor on the coils of the air conditioner creates an additional load on the equipment. A typical value is 450 BTU per hour. This corresponds to the evaporation of about ten pounds of water per day. The user enters a constant value or the name of a schedule under LATENT GAINS.

Section 2-5. CONDUCTION

Perhaps the simplest heat transfer mechanism in buildings is the gain or loss of heat by conductance through walls, ceilings, etc. For convenience, all building elements separating zones from each other and from AMBIENT and GROUND are referred to as walls. The program provides three different ways to describe conductance.

1) By the use of a steady state heat transfer coefficient,
2) By use of walls that are pure resistances,
3) By use of walls that have one or more layers of materials with heat capacity.

Three methods are provided rather than one so that the user has maximum flexibility in the choice of the level of detail for modeling wall elements. Thus, if the heat capacity of a given wall is judged to be inessential to the problem, it may be ignored. This approach also minimizes the labor of preparing a building description for the program.

A. USE OF CONDUCTANCE COEFFICIENTS

Whenever the user wishes to ignore the thermal capacity of the wall, solar effects on the inside and outside of the wall, and the exterior and interior surface temperatures of the wall, the product of the wall area and U-value is entered as a CONDUCTANCE COEFFICIENT in the INTERZONE data section. These coefficients may also be used to model estimated convective transfers between zones. Where conductances are specified between two building zones or from a zone to AMBIENT or GROUND, they must include all walls or paths of heat transfer not accounted for in the WALLS data section.

B. WALLS

The second and third levels of detail require use of the WALLS data section. One side of the wall, called the FRONT/INTERIOR side, must face a user-defined ZONE. The other, called the BACK/EXTERIOR side, may face either a ZONE, an EXTERIOR SURFACE, or one of the keywords, AMBIENT or GROUND. If AMBIENT or GROUND is specified, there are no solar effects on the exterior side. A wall may have the same zone specified on both sides so that it is wholly contained within the zone.

The user must enter the names of the INTERIOR ZONE and the EXTERIOR SURFACE of the wall, and the wall AREA. In addition, there is a SURFACE COEFFICIENT and a SOLAR COEFFICIENT. The SURFACE COEFFICIENT is the combined radiation and convection heat transfer coefficient. A typical value for interior vertical surfaces is 1.46 BTU per hour per square foot per degree Farenheit. For exterior surfaces, a typical value is about 4 BTU per hour per square foot per degree Farenheit. A table of surface conductances can be found in the ASHRAE Handbook of Fundamentals.

The SOLAR COEFFICIENT for an interior surface is the fraction of the total solar available in the zone that falls upon the given surface. For an exterior surface the SOLAR COEFFICIENT is the absorptivity of the surface for short-wave solar radiation.

C. USE OF R-VALUE WALLS

The second level of detail provided for wall descriptions is the use of an R-value together with the area of the wall. When represented in this way the program calculates heat flow through the wall and the interior and exterior surface temperatures. The program also calculates all solar effects on such walls.

To treat a wall as an R-value, enter R-n under WALL TYPE in the WALLS section. Individual layers of a wall may be treated as R-values by entering R-n for the appropriate layer in the WALL.TYPES data section. The value of n is limited only by the width of the entry code AAAAAAA in the data section header. Detailed output is available summarizing all of the factors relating to the performance of such walls.

D. USE OF CAPACITY WALLS

The third level of detail allows for the description of the wall as composed of one or more layers of material. Each of these layers may consist of either an R-value or a specified material described by its thickness, specific heat, density, and conductivity. In this way, walls of almost arbitrary complexity may be treated. Additionally, if the walls are part of an exterior surface and the user wishes to determine the effects of solar energy on the wall, he must specify an azimuth, an absorptance, and parameters for shading.

Since the program uses a thermal network model, nodes (each representing a thin slice of material) must be specified in materials with heat capacity. A variety of types of capacity layer are available. These include:

1) Single node with internal resistance,

   Used for thin layers of solid material or, in some cases, for thick layers where accurate surface temperatures are unimportant.

2) Single node without internal resistance,

   Used for water walls or drums where convective stirring effectively eliminates internal resistance; i.e., the entire thickness is at the same temperature as the surface.

3) Single node phase change,

   Allows the user to model thin layers of phase change material Latent heat of fusion must be specified.

4) Multi-node with internal resistance.

   Allows the user to specify any number of nodes within any given layer of material, so that the temperature at any point within the material can be modeled to any desired degree of accuracy.

The execution time (and cost) of a simulation depends in a linear way upon the number of mass nodes used. Therefore, the user must exercise good engineering judgment in the selection of the level of detail in modeling capacity elements. The following guidelines may be useful.

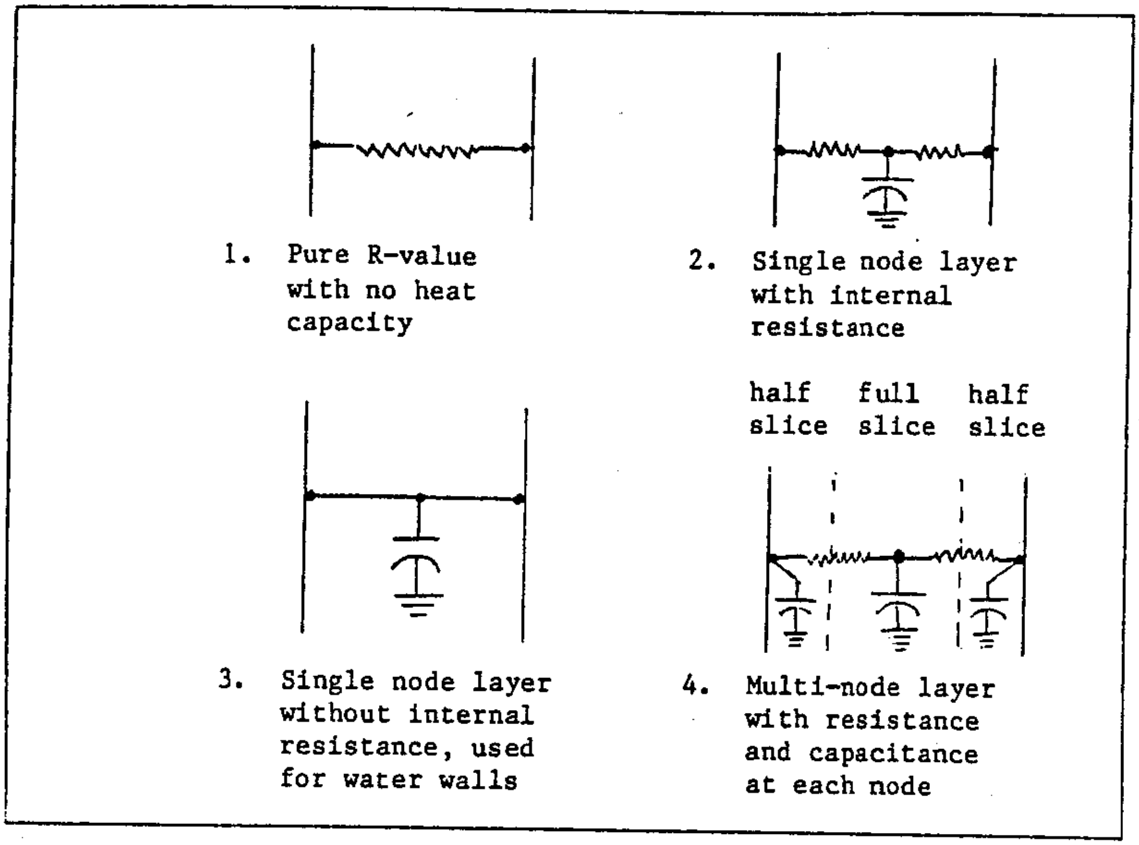
1) For walls experiencing large temperature variations at the surface (i.e., Trombe walls), a node spacing of about two inches will give accurate results. If the temperature variations and their dynamic effects are small, a larger spacing will be adequate.

2) Walls two or more feet thick may be modeled with surface layers spaced at two inches and an internal layer at four to six inch spacing without loss of accuracy. Earth berms and similar very thick walls may have nodes one foot or more apart in the interior.

3) Walls less than four inches thick can generally be modeled with a single node.

4) It is generally most effective to treat the interior layers of walls in detail and the outer layers in less detail. For instance, in a conventional 2 x 4 stud construction, it will be adequate to provide a mass node only for the interior wallboard or plaster.

5) Where there is a dominant thermal capacitance such as a concrete floor in a zone, the capacitance of other elements such as wallboard and furniture may be safely ignored.



1. Pure R-value with no heat capacity

2. Single node layer with internal resistance

3. Single node layer without internal resistance, used for water walls

4. Multi-node layer with resistance and capacitance at each node

Fig. 2-2. Types of Layers Available for Making Walls

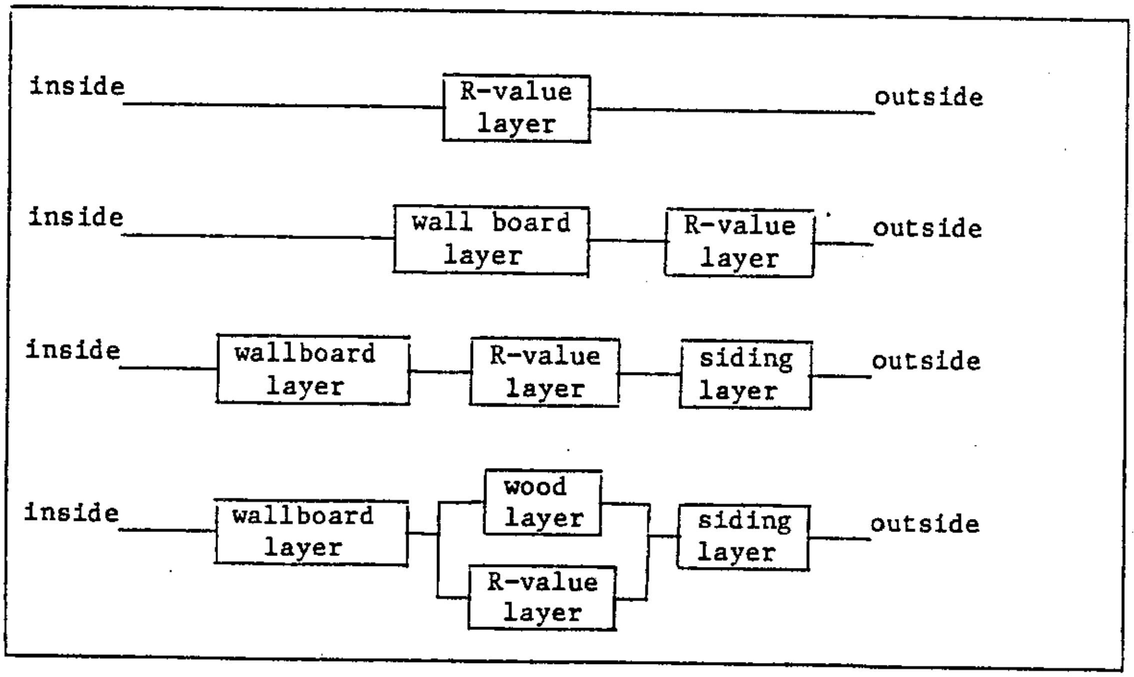


Fig. 2-3 Modeling a Wood-Frame Wall with Different Levels of Detail

Walls may be constructed of multiple layers, each with capacity. The name of each LAYER is entered in the WALL.TYPES data section with LAYER # 1 representing the layer closest to the user-specified FRONT/ INTERIOR ZONE. The parameters CONDUCTIVITY, DENSITY, SPECIFIC HEAT, THICKNESS, and number of NODES for each LAYER are entered in the MASS. TYPES data section.

E. TROMBE WALLS

Trombe walls are a special case of capacity walls, allowing thermocirculation of warm air through vents into the interior zone. They must be defined as belonging to an EXTERIOR SURFACE. The names of the INTERIOR ZONE, the EXTERIOR SURFACE, and the TROMBE TYPE; the interior SURFACE COEFFICIENT, and the interior SOLAR COEFFICIENT are entered in the TROMBE.WALL data section. If shading is specified, the additional parameters HEIGHT, LENGTH, HORIZONTAL LOCATION, and VERTICAL LOCATION also entered.

In the TROMBE.TYPES data section, the user enters the name of the WALL TYPE, the name of the GLAZING TYPE, the exterior SURFACE COEFFICIENT the EXTERIOR ABSORPTIVITY, and parameters controlling the thermocirculation. The HEIGHT BETWEEN VENTS is the vertical distance separating the centerlines of the upper and lower rows of vents. The VENT AREA RATIO is the ratio of one row of vents to the total area of the Trombe wall. The VENT COEFFICIENT is a number between zero and one that multiplies the volumetric flow rate. This accounts for the flow resistance due to the vents. A value used in other studies is .8; however, there is still controversy over the true magnitude of thermocirculation.

Section 2-6. SOLAR GAINS

Solar gains on exterior walls and through windows have an important impact on heating requirements. This program treats solar effects in detail.

A. EXTERIOR SURFACES

The user may define and name EXTERIOR SURFACES. Subsequently, walls or windows or both are defined as belonging to an exterior surface. The underlying logic is to minimize the geometric input required from the user. For each exterior surface, the user must enter the COMPASS AZIMUTH, the TILT, the HEIGHT, the LENGTH, and, optionally, the names of an OVERHANG TYPE, and LEFT and RIGHT SIDEFIN types in the SURFACES data section.

B. WINDOWS

Each window is defined as belonging to an exterior surface and facing an interior zone. The names of the EXTERIOR SURFACE,the INTERIOR ZONE, and the GLAZING TYPE are entered in the WINDOWS data section. The user must also enter a HEIGHT and LENGTH for each window, as these are used to calculate the window area. The HORIZONTAL and VERTICAL LOCATION parameters locate each window with respect to the exterior surface to which it belongs. They are optional, as they are only used when shading is specified on the surface.

Each window is composed of one or more layers of partially transparent material. The material properties and thickness are specified by the user. The program accounts for all multiple reflection and absorptance within and between the glazing layers. The user must enter the GLAZING U-VALUE, SHADING COEFFICIENT, EXTINCTION COEFFICIENT, INDEX OF REFRACTION, THICKNESS OF LAYER, and NUMBER OF LAYERS in the GLAZING TYPES data section. The U-VALUE is the steady-state conductance taken from inside air temperature to outside air temperature. Thus, it includes the internal and external combined surface coefficients. It may be scheduled.

The SHADING COEFFICIENT allows the user to model the effects of curtains, venetian blinds, and various types of external shading devices. The SHADING COEFFICIENT multiplies the solar heat gain. This is defined as the sum of the transmitted shortwave radiation and the inward-flowing fraction of the solar radiation absorbed in the glazing layers.

The SHADING COEFFICIENT concept used in this program is similar to that found in the ASHRAE Handbook of Fundamentals; however, the numerical values are not the same. The Handbook contains extensive tables of shading coefficients defined as the ratio of the solar heat gain through a given glazing assembly to that of a standardized single glazing. The SHADING COEFFICIENT used in this program is the ratio of the solar heat gain through the given assembly to the solar heat gain through the glazing layers alone. Thus, the ASHRAE values must be corrected before use in this program.

If the assembly contains a single layer of 1/8 inch glass, the ASHRAE values are used unaltered. The ASHRAE shading coefficients for glazing assemblies with multiple glazing layers or with a single layer whose properties differ from those of 1/8 inch glass must be corrected by dividing them by the ASHRAE shading coefficients for just the multiple glazings or the single layer.

This parameter can also be scheduled, for instance, to allow for solar control during the summer. It is intended that a small library of GLAZING TYPES will be adequate for practical work and so eliminate the need to specify these details.

C. SHADING

Several types of shading devices are provided. Each exterior surface may have a horizontal overhang. The user must specify the distance the overhang projects from the surface and the vertical distance above the top of the surface at which the overhang is located. See the OVERHANG.TYPES data section in the appendices for further details. Direct radiation is set to zero if the sun is obstructed by the overhang.

D. SIDEFINS

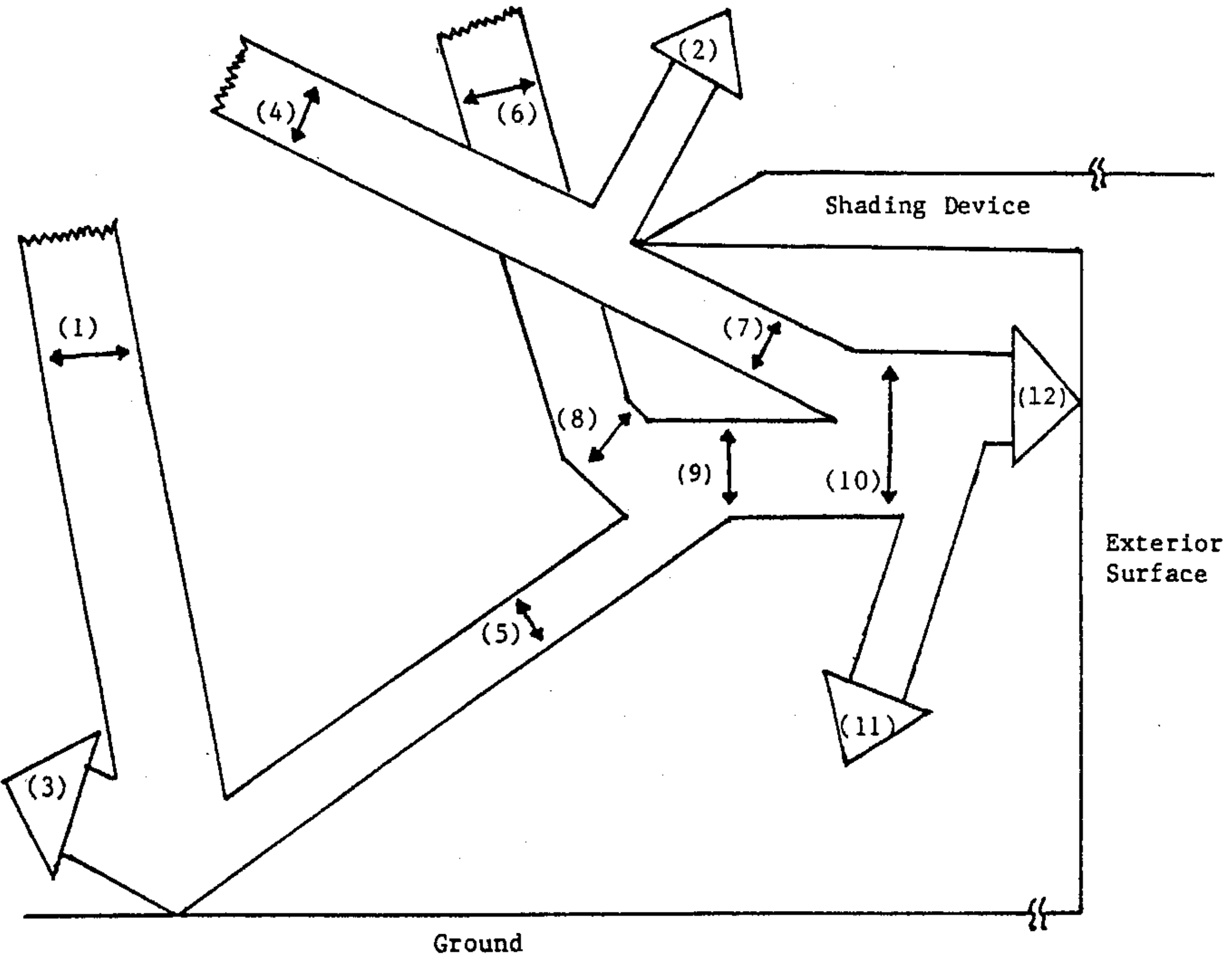
In a similar way, the user may specify left or right sidefins or both. The sidefins affect only the direct radiation. The side fins are assumed to project normal to the shaded surface. See the SIDEFIN.TYPES data section for further details on sidefins.

E. SKYLINE PROFILES

The program can model the effect of trees, buildings or other nearby objects in solar availability at the simulated building site. SKYLINE PROFILES are specified in the RUNS data section and defined in the SKYLINE.TYPES data section. Shading due to skyline obstructions is considered before all other shading effects and transmitted radiation values are calculated.

F. EXTERNAL DISTRIBUTION OF SOLAR RADIATION

The diagram below summarizes the exterior distribution of solar radiation.



(1) Total Horizontal Radiation after Skyline Shading.
(2) Direct Solar Lost by Overhang Shading. Calculated by program.
(3) Ground Scattering. Calculated by program.
(4) Direct Solar on Tilted Surface after Skyline Shading (Unshaded by Overhang). Calculated by program.
(5) Ground Diffuse on Tilted Surface. Calculated by program. May also be reduced by sidefin shading (not shown).
(6) Sky Diffuse on Tilted Surface (Unshaded). Calculated by program.
(7) Shaded Direct Solar on Tilted Surface. Calculated by program.
(8) Shaded Sky Diffuse Solar on Tilted Surface. Calculated by program.
(9) Total Shaded Diffuse on Tilted Surface. Calculated as (5)+(8).
(10) Total Incident Solar. See Fig. 2-5 for further effects on glazing systems.
(11) Reflection Losses. See (12) below.
(12) Solar Absorbed. Specified by user for each exterior wall as a fraction of Total Incident Solar (10).

Fig. 2-4 Illustration of Exterior Distribution of Solar Radiation

G. INTERNAL DISTRIBUTION OF SOLAR

The following parameters affect the magnitude and distribution of solar radiation in each zone.

1) SHADING COEFFICIENT for each window in the zone

2) SOLAR TRANSFER and REVERSE TRANSFER in the INTERZONE data section

3) FRACTION TO AIR and FRACTION LOST in the ZONES data section

Each window may have a user specified SHADING COEFFICIENT that multiplies solar heat gain through that window. The solar heat gain has two components: the short-wave solar transmitted through the window and the inward-flowing fraction of the solar radiation absorbed in the glazing layers. The inward-flowing absorbed radiation goes directly to the zone air temperature node.

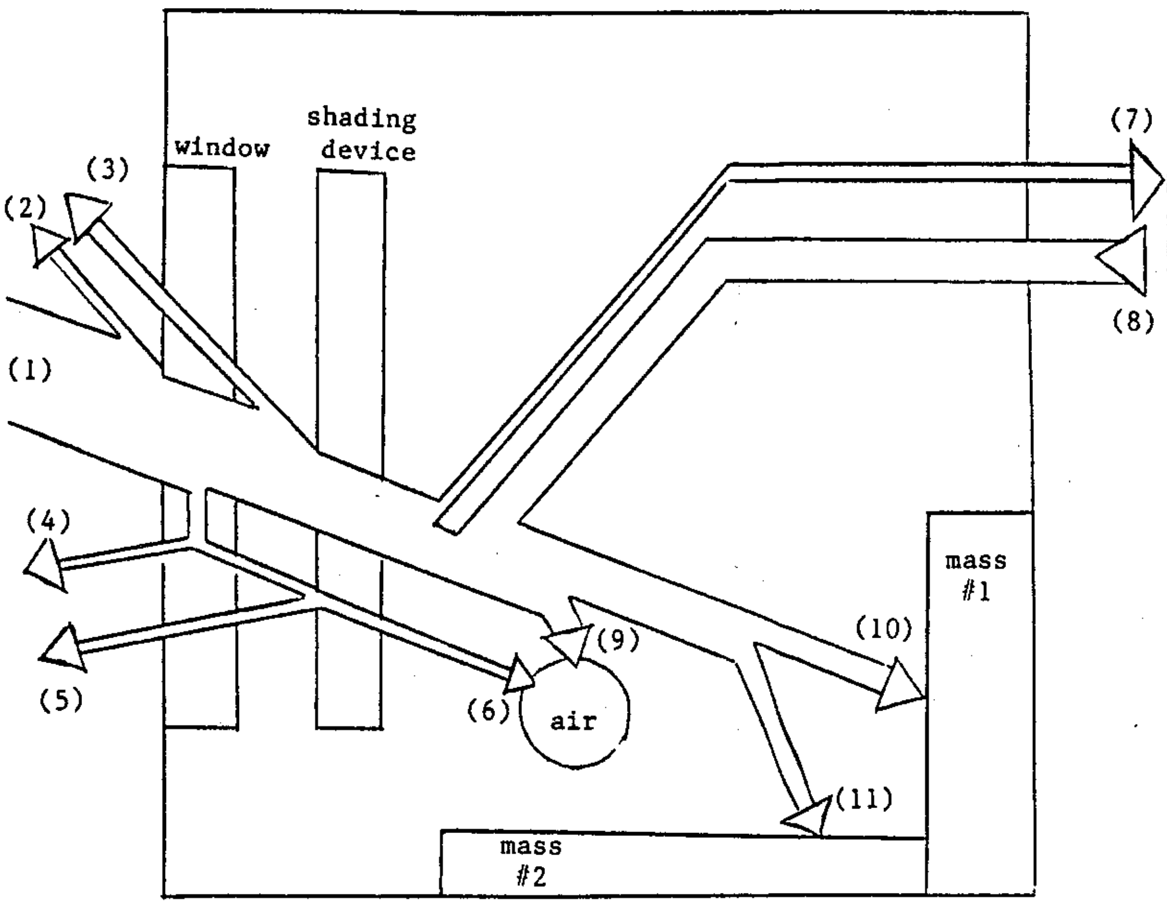
The sum over all windows of the transmitted short-wave multiplied by the SHADING COEFFICIENTS is then multiplied by a FRACTION LOST factor. This factor accounts for the short-wave radiation reflected back through the glazings and lost. The FRACTION LOST can be thought of as the effective cavity absorptance of the zone. Typical values range from .05 to .10.

The remaining short-wave radiation is then distributed by several mechanisms. If there are multiple zones, any fraction of radiation entering one zone may be passed to another and vice versa. The user enters these fractions under SOLAR TRANSFER and REVERSE TRANSFER in the INTERZONE data section. This allows for a crude treatment of the presence of transparent surfaces between zones.

After accounting for interzone transfers, in each zone a fraction of the remaining available short-wave radiation may be put immediately into the zone air temperature node. This is the FRACTION TO AIR in the ZONES data section. This allows the user to account for that portion of the radiation absorbed by non-massive objects and converted more or less instantly into heat. Typical values range from .10 to .25.

The remaining short-wave radiation, after all of the above factors are applied, may be distributed onto the walls in two ways. The first method simply allows the program to distribute the radiation over all walls in a given zone in proportion to their areas. If more detail is desired, the user may associate a fraction absorbed with each surface. This allows the user control over the relative amounts of radiation received by each wall. The fraction absorbed by each wall is entered as the SOLAR COEFFICIENT for each wall. For each zone, the sum of all the SOLAR COEFFICIENTS for wall surfaces facing that zone, plus the FRACTION LOST, plus the FRACTION TO AIR must equal one.

The diagram on the next page shows the internal distribution of radiation.



(1) Total Solar Incident on window.
(2) Reflection Losses from glazing layers.
(3) Reflection Losses from shading device.
(4) Outward flowing portion of solar absorbed in glass.
(5) Loss of inward flowing portion of solar absorbed due to shading device.
(6) Net inward flowing portion of solar absorbed in glass goes to air node.
(7) Shortwave solar loss to other zones.
(8) Shortwave solar gain from other zones.
(9) Shortwave to air node.
(10) Shortwave to mass #1.
(11) Shortwave to mass #2

Fig. 2-5 Illustration of Interior Distribution of Solar Radiation

Section 2-7. EQUIPMENT

A. GENERAL

When zone temperatures fall above or below the comfort range, equipment must be brought into operation to maintain comfort if possible. The program does not model the operation of equipment. This would require too much detailed description of the performance characteristics of many different types and models of equipment.

The basic strategy is to define adequate equipment performance and then, in the simulation, calculate the results of such equipment operation. For instance, a fan moves heat from a warm zone to a cooler one. Some minimum temperature differential must exist to justify the operation of the fan (i.e., the value of the heat moved should exceed the value of the energy required by the fan to move it).

Each zone may have cooling and heating setpoints. Adequate operation of the fan is then defined as movement of as much heat as necessary to maintain the heating setpoint in the cooler zone without causing the heating equipment to come on in the warmer zone, subject to the constraints of a minimum temperature differential and a maximum rate of delivery.

Equipment operation follows a sequence: Fans are operated first, then rockbins, then venting, and, finally, any remaining loads are satisfied, if possible, by the heating and cooling equipment. By this strategy the program calculates the loads on the various types of equipment without going into the details of particular types of equipment and their associated control systems. Thus, the program evaluates equipment loads, but not the input energy required to satisfy those loads by some particular set of equipment and controls.

B. HVAC

1. Heater

The heater provides heat to a zone as necessary to maintain its heating setpoint. If the user has specified a maximum capacity for the heater that is too low at a given moment, heat is added at the maximum rate, and the resulting zone temperature is calculated. The heater deals only with sensible heat and does not include any latent heat effects. The HEATING SETPOINT and HEATING CAPACITY are entered in the HVAC.TYPES data section. "No heating" is selected by leaving HEATING SETPOINT at its default value, <NONE>.

2. Venter

The venter provides for thermostatically controlled exchange of zone air with outdoor air. The intent is to model two phenomena. First, the venter can be seen as an economizer cycle for an air conditioning system whereby cooling is achieved by forced ventilation with cooler outside air without activation of the cooling coils, or equivalently, a separate thermostatically controlled fan may be used.

The second phenomenon is action by the occupant to increase natural ventilation, for instance by opening doors and windows. The venter removes heat to maintain a venting setpoint subject to its maximum capacity whenever the outdoor air is cooler than the indoor air.

The VENTING SETPOINT and VENTING CAPACITY (in air changes per hour) are entered in the HVAC.TYPES data section. As for heating, "no venting" is selected by the default VENTING SETPOINT value of <NONE>.

3. Cooler

The cooler removes heat from a zone as necessary to maintain the cooling setpoint subject to its maximum capacity. As with the heater, if the capacity is inadequate, it is operated at the maximum rate and the resulting zone conditions calculated. The cooler is thermostatically controlled and does not respond to latent loads.

The COOLING SETPOINT and COOLING CAPACITY are entered in the HVAC. TYPES data section. "No cooling" is selected by the default COOLING SETPOINT value of <NONE>.

4. Latent Heat

The program has limited capabilities for handling latent heat effects. Latent calculations are made in a similar fashion to those for a variable volume air system. The cooler is controlled by a dry-bulb thermostat. The sensible cooling load determines the rate of air flow through the cooler. The air is cooled to a user specified COOLER COIL TEMPERATURE and any resulting de-humidification of the zone air is calculated. The humidity ratio and relative humidity of the zone air are updated hourly.

The COOLER COIL temperature is entered in the HVAC.TYPES data section. A typical value is 55 F.. Detailed output regarding latent effects is available in the LATENT HEAT section of the ZONE SUMMARY.

C. FANS

The user may specify one or more fans between zones. Since fans are assumed to be uni-directional, the zones may be uniquely labeled as a SOURCE ZONE (the warmer one) and a SINK ZONE (the cooler one). A SINK ZONE may be connected to only one SOURCE ZONE by fan; however, a SOURCE ZONE may supply several SINK ZONES. The operation of a fan may be disabled for one user-defined SEASON of the year.

The names of the SOURCE and SINK ZONES, the name of the FAN TYPE, and the name of the OFF SEASON are entered in the FANS data section.

The MAXIMUM CAPACITY (volumetric flow rate), and the MINIMUM TEMPERATURE DIFFERENTIAL are entered in the FAN.TYPES data section.

Detailed output on fan performance is available in the FAN SUMMARY output block.

D. ROCKBINS

The rockbin model used in the program is the infinite NTU model developed at the University of Wisconsin by Pat Hughes and others. It is nearly identical to the rockbin module in TRNSYS 10. Rockbins are one-way flow devices in any given operating mode. The SOURCE ZONE provides the inlet air during the charge cycle, while the SINK ZONE receives the outlet air during the discharge cycle. A single zone may be specified as both the SOURCE ZONE and the SINK ZONE for a rockbin.

A rockbin loses or gains heat passively to one user-defined zone and to the predefined zones AMBIENT and GROUND. The names of the SOURCE ZONE, the SINK ZONE, the ROCKBIN TYPE, the ZONE FOR PASSIVE LOSSES, and the values for the passive conductances are entered in the ROCKBINS data section.

The user may specify either of two types of air flow control. In the first type, air flow is always in the same direction; that is, the inlet is always at the same physical end of the rockbin.

The second type has reversing flow; that is, the direction of air flow in the charge mode is opposite to that of the discharge mode. The second type allows for maximum advantage from stratification of temperature within the rockbin and generally provides superior performance.

The user must specify the type of flow control, the volumetric HEAT CAPACITY of the rockbin, the AXIAL CONDUCTANCE of the rockbin, the LENGTH and CROSS-SECTIONAL AREA of the rockbin, the names of the FAN TYPES for the charge and discharge fans, and the name of a user-defined CHARGE OFF SEASON. These parameters are entered in the BIN.TYPES data section. The charge and discharge fans may be of different types.

Detailed output on rockbin performance is available in the ROCKBIN SUMMARY output block.

E. FAN AND ROCKBIN CONTROL STRATEGY

Fans, including rockbin charge and discharge fans, are modeled as a thermostatically controlled conductance between zones or between a zone and a rockbin.

Each fan has an ideal controller. An ideal controller is one that delivers the maximum amount of heat from the source to the sink while obeying the four constraints discussed below. This requires the controller to be able to cycle the fan on and off at an arbitrarily high rate during a time increment. Equivalently, the fan controller can also be seen as being able to select that fan speed between zero and the fan's specified MAXIMUM CAPACITY, which maximizes fan performance.

The term duty cycle refers to, in the first viewpoint, the fraction of the time increment in which the fan is on and, in the second viewpoint, the fraction of full capacity at which the fan is operated.

Each fan operates at no more than its specified MAXIMUM CAPACITY; that is, its duty cycle cannot be greater than unity. This is referred to as the maximum capacity constraint.

Each fan controller has a MINIMUM TEMPERATURE DIFFERENCE between a SOURCE ZONE and a SINK ZONE. For the fan to operate, the temperature in the SOURCE ZONE must be greater than the temperature of the SINK ZONE plus the specified minimum temperature difference. This is referred to as the minimum temperature difference constraint.

In addition, the fan controller interacts with the SOURCE and SINK ZONE setpoints in the following ways. The setpoints in each zone must satisfy the inequality:

$$\text{HEATING SETPOINT} \leq \text{VENTING SETPOINT} \leq \text{COOLING SETPOINT}$$

Note that not all setpoints need be specified. For instance, a zone may have venting and cooling, but not heating. But all defined setpoints must obey the above inequalities.

Then, if a HEATING SETPOINT is specified for the SOURCE ZONE, the operation of the fan will not lower the temperature of the SOURCE ZONE below its HEATING SETPOINT. This is referred to as the maximum energy available constraint.

In addition, the fan will not raise the temperature of the SINK ZONE above the lowest setpoint specified (if any). That is, if the SINK ZONE has a HEATING SETPOINT specified, the fan will not raise the SINK ZONE temperature above the HEATING SETPOINT. If the SINK ZONE does not have a HEATING SETPOINT, but does have a VENTING SETPOINT specified, then the fan will not raise the SINK ZONE temperature above the VENTING SETPOINT. In the same way the fan will not raise the SINK ZONE temperature above the COOLING SETPOINT, if one is defined. This is referred to as the maximum energy needed constraint.

If no thermostat setpoints are specified, the fans will operate so as to deliver the maximum energy from the SOURCE ZONE to the SINK ZONE, subject to the minimum temperature difference and maximum capacity constraints.

Note that these interactions with HVAC thermostats cause the fans to operate primarily as a heating device for the SINK ZONE (subject to the constraint of not causing heating in the SOURCE ZONE), rather than as a cooling device for the SOURCE ZONE.

Rockbins may be either charging (receiving energy from the SOURCE ZONE) or discharging (delivering energy to the SINK ZONE) during a time increment, but not both. When conditions are such that either could occur, the rockbin will charge (charge priority).

Subject to the four constraints defined above, each fan or rockbin will be operated to deliver the maximum energy possible, with one exception. The energy delivered from a rockbin will be limited to the maximum capacity of the heater, and the temperature of the SINK ZONE will be the same as it would have been had the rockbin not operated. In particular, note that a rockbin SINK ZONE must be heated (i.e., a heating setpoint must be specified) for the rockbin to discharge.

The assumption of ideal control is formulated as a constrained optimization problem. The four constraints, combined with the restrictions discussed below, result in a uniquely determined duty cycle for each time increment.

F. FAN AND ROCKBIN PLACEMENT RESTRICTIONS

There are several restrictions on the placement of fans and rockbins. There are two reasons for these restrictions.

The first is the complexity of creating a consistent logic within the program for handling such situations. For instance, should a living room be kept at its cooling setpoint by heat delivered from a sunspace so that excess heat can be moved from the living room to the bedroom?

A second and related reason is the difficulty in the real-world situation of devising adequate control of such arrangements, combined with the fact that need for such complex fan arrangements may indicate that effort would be better invested in improving the building design. Generally, the restrictions are intended to avoid the situations diagrammed below.

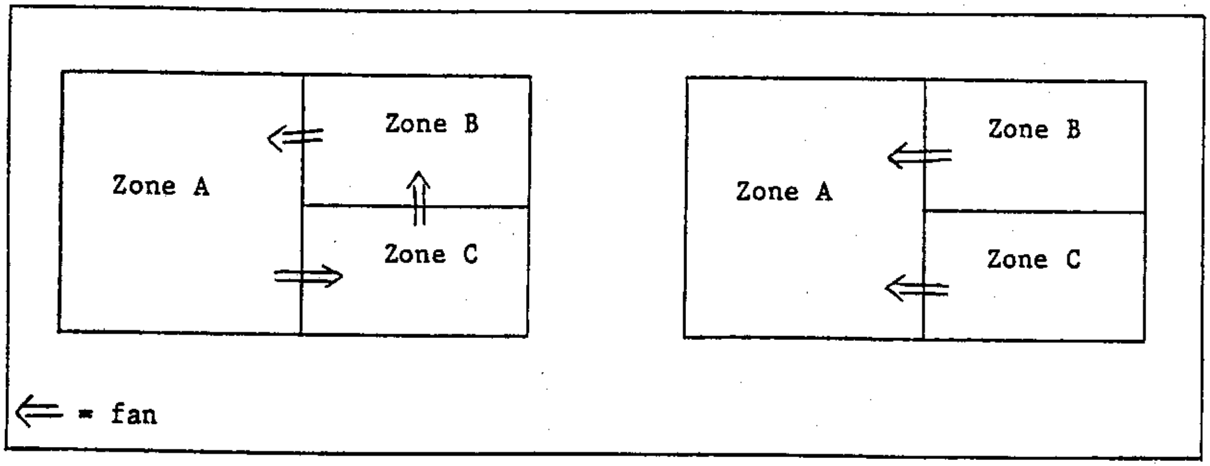


Fig. 2-6. Illegal Placements of Fans

Specifically, the restrictions are:

1) A zone may be the SINK ZONE for at most one fan;

2) A zone may be the SINK ZONE for at most one rockbin.

3) A zone that is the SINK ZONE for a fan may be connected to a rockbin only if it is:

    a) Both the SOURCE ZONE and the SINK ZONE for the rockbin, or

    b) The SINK ZONE for the rockbin, and the SOURCE ZONE for the fan is also the SOURCE ZONE for the rockbin;

4) A zone may not be the SOURCE ZONE for one fan and the SINK ZONE for another fan;

5) A zone that is the SOURCE ZONE for a fan may be the SINK ZONE for a rockbin, only if it is also the SOURCE ZONE for that rockbin.

Note that the restrictions allow a zone to be the SOURCE ZONE for several fans and rockbins, but limit a zone to be the SINK ZONE for at most one fan and/or rockbin.

Examples of allowable fan and rockbin placements are shown below.

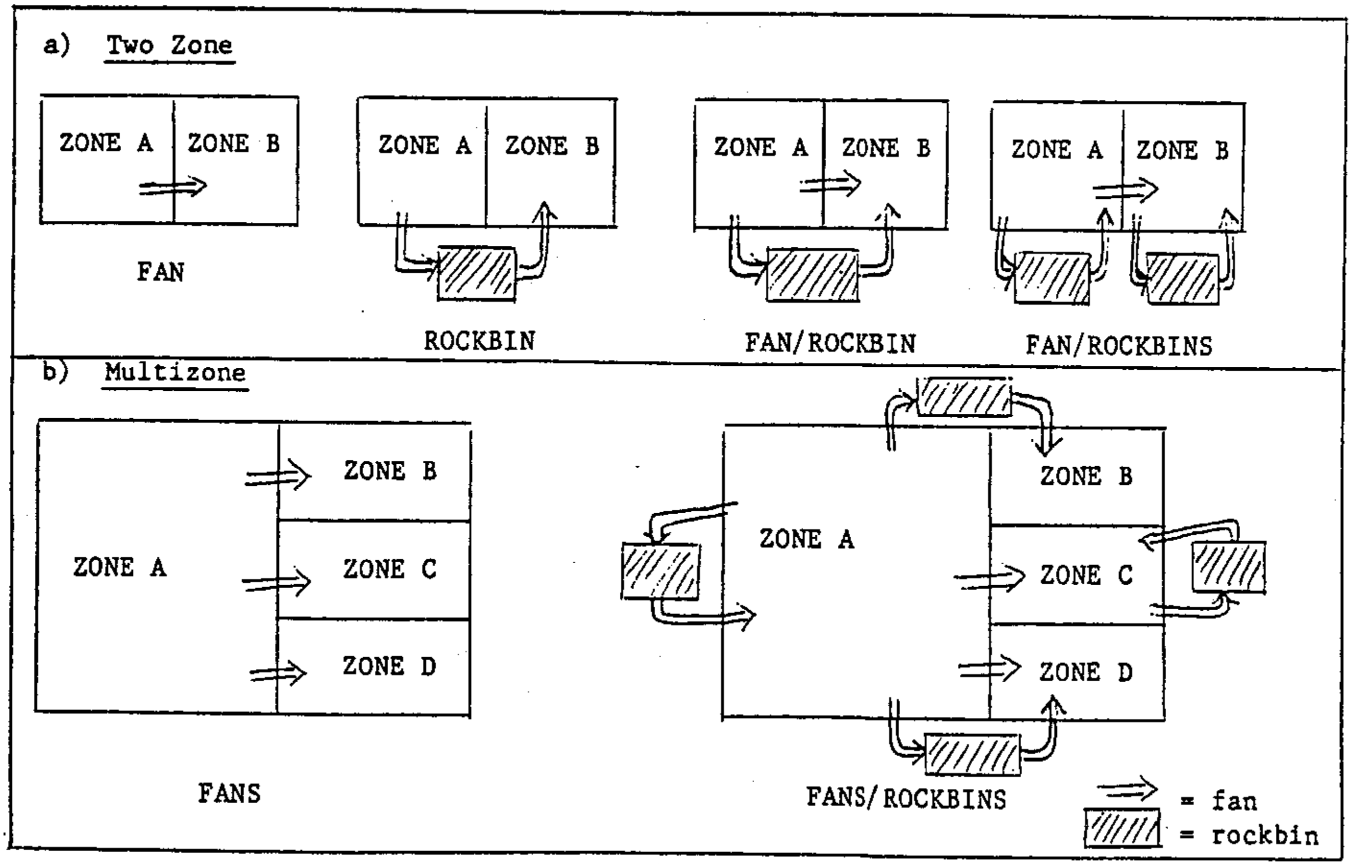


Fig. 2-7. Examples of Allowable Fan and Rockbin Placements

Section 2-8. SCHEDULES

Some of the input variables may be scheduled. That is, they can be assigned different values for different hours during the day or different values for different SEASONS of the year. Parameters that may be scheduled have an entry code of the form SSS.SSS in the data section header. They are:

GROUND REFLECTANCE
INFILTRATION RATE
LATENT GAIN
REVERSE TRANSFER
SHADING COEFFICIENT
VENTING SETPOINT
GROUND TEMPERATURE
INTERNAL GAIN
SOLAR TRANSFER
GLAZING U-VALUE
HEATING SETPOINT
COOLING SETPOINT

When a numeric value is specified for a parameter which may be scheduled, then that value will be used for the entire simulation run.

When the name of a SCHEDULE in the SCHEDULES section is specified instead of the numeric value; the set of data values and the times for which they apply, contained within the SCHEDULE entry, are used for that parameter.

The schedule assigns to the parameter the same data value for a given hour over a contiguous set of days (a SEASON). The user must define SEASONS by entering the START and STOP dates in the SEASONS data section.

The hourly values for the parameter may be specified as applying over a contiguous range of hours, or as applying to a single hour. Up to four values, each referring to a range or a single hour, may be specified in each data line. Multiple data lines may be entered for each schedule. They must be entered contiguously.

The first data line for each schedule entry must have the schedule name and season name specified. Subsequent data lines of the <u>same</u> schedule need not have the schedule name specified. In the same way, multiple lines referring to the same season need not have the season name repeated. Each new SCHEDULE or SEASON must start with a new schedule or season name.

The following rules apply for specifying the hours over which a parameter value applies:

1) Hours are specified as 1 (1:00 a.m.) through 24 (midnight).

2) Each parameter value must be preceded by a start hour. The value is used starting at that hour and continuing up to, but not including, the start hour for the next parameter value.

3) The last parameter value specified for a schedule is used for all remaining hours of the day.

Section 2-9. OUTPUT

The method of specifying output is designed to allow the user the maximum amount of flexibility regarding the level of detail for various components of the building. The possible outputs are given in detail in the Output Data Sections appendix. The output is divided into blocks organized around the major thermal components of the building. The complete building description used for the simulation is echoed in the output.

The output specification section will have one or more lines of parameter values entered. Each line will specify the block of output data desired, the time step for the output, the season for that output desired, the units (English or Metric) desired, the element of a given type for output desired (for instance, the third wall), the page or Output Data Section in the block desired, and the type of format for the output file.

For the convenience of the user, a standard output specification is defined and will automatically be produced if no values are entered in the Output Data Section. This output will consist of monthly and run length summaries for the AMBIENT and BUILDING output data blocks. All output pages for these blocks will be produced in the units in which the building description was entered. The format will include headers describing each variable and its units.

The first parameter in the output specification section is the OUTPUT TYPE. This refers to the blocks of output desired. The possible values are ALL, AMBIENT, BUILDING, ZONES, WINDOWS, WALLS, SURFACES, FANS, ROCKBINS, and TROMBES. The default value ALL will cause all applicable output blocks to be produced.

The second parameter TIME PERIOD defines the time step for the desired output. The third parameter specifies desired units. The fourth parameter specifies the user-defined season for desired output. The fifth parameter selects one of a number of elements, such as walls, for desired output. The fifth parameter selects which of several output sections for that element desired, and the sixth parameter selects either human or machine readable output.

As an example, suppose the user desired to run the building for a full year with monthly summaries for the AMBIENT, BUILDING, and ZONES output blocks. In addition, hourly output is desired for the zone and outdoor temperatures for the days January 15 and July 15. The user would define two seasons say, Jan15 and Jun15, which each start and stop on the appropriate day. Then in the output data section of the input description the user would enter the following:

OUTPUT TYPE	TIME PERIOD	UNITS	OUTPUT SEASON	BUILDING ELEMENT	OUTPUT SECTION	FORMAT?
AMBIENT	M		<ALL>	<ALL>	<ALL>	Y
BUILDING	M		<ALL>	<ALL>	<ALL>	Y
ZONES	M		<ALL>	<ALL>	<ALL>	Y
AMBIENT	H		JAN15	<ALL>	1	Y
AMBIENT	H		JUL15	<ALL>	1	Y
ZONES	H		JAN15	<ALL>	6	Y
ZONES	H		JUL15	<ALL>	6	Y

Output summaries will be created in the file in the order in which they are specified.

CHAPTER 3. CREATING AND EDITING BUILDING DESCRIPTIONS

Section 3-1. INTRODUCTION

The EDIT module is one of two Fortran modules that constitute the SUNCODE program. It is used to create, display and edit building descriptions. The editor allows rapid interactive entry and editing of the many parameters required for a particular simulation. The user is prompted with headings that give the format and appropriate units for each of the parameters entered.

In order to catch as many input errors as possible, the editor also performs all of the numeric operations necessary to transform the input data into the values actually used in the LOADS module. For instance, in describing a wall, the user enters parameters like thickness, number of nodes, and material properties. The editor module transforms these into the actual coefficients for temperature and radiation for each node. If errors are found, the program will report them.

The EDIT module is entered by typing EDIT and carriage return at the operating system prompt. As EDIT is initiated, it states whether it is using English or Metric (System International -- S.I.) units for the building description. The Master Menu is then displayed.

Section 3-2. THE MASTER MENU

The master menu, shown in Figure 3-1, controls all operations while in the EDIT module. It consists primarily of file management commands, with the notable exception of #6, the Display/Edit command, which is the screen editor. Each command is discussed briefly below.

```
SUNCODE              MASTER MENU

1. TERMINATE THE PROGRAM
2. READ BUILDING FILE
3. WRITE BUILDING FILE
4. READ LIBRARY FORMAT FILE
5. WRITE LIBRARY FORMAT FILE
6. DISPLAY/EDIT BUILDING
7. COMPILE INTERMEDIATE FILE

SELECT YOUR CHOICE #
```

Figure 3-1. The Master Menu

1. Terminate the Program

This quits the EDIT module and returns you to the operating system. You are given an opportunity to save your workfile as a building file, if desired.

2. Read Building File

This reads an existing building file into the workfile. You are prompted for a building file name. No filename extension need be specified; the extension .BLG is assumed. Filenames can be a maximum of 10 characters; drive designations and directory paths are acceptable. The program displays the label of the file you've specified, and requests confirmation that the file has been correctly identified. It then loads the file into the EDIT workfile.

3. Write Building File

This saves the workfile by writing it into a building file. You are prompted for a building file name. No filename extension need be specified; the extension .BLG is assumed. Filenames can be a maximum of 10 characters; drive designations and directory paths are acceptable. If a file of the same name already exists, the program will ask permission to write over the existing file, and if necessary, will prompt for a new filename.

4. Read Library Format File

This reads an existing library file into the workfile. Library files are not complete building descriptions; they consist of certain input sections which are used frequently without alteration. They are discussed further in Section 3-5. You are prompted for a library file name. No filename extension need be specified; the extension .LIB is assumed. Filenames can be a maximum of 10 characters; drive designations and directory paths are acceptable.

5. Write Library Format File
   This saves the workfile by writing it into a library file. Library files are not complete building descriptions; they consist of certain input sections which are used frequently without alteration. They are discussed further in Section 3-5. You are prompted for a library file name. No filename extension need be specified; the extension .LIB is assumed. Filenames can be a maximum of 10 characters; drive designations and directory paths are acceptable. If a file of the same name already exists, the program will report an error and prompt for a new filename.

6. Display/Edit Building
   This allows actual display and editing of the contents of the workfile. All parameter input and editing is completed as interactive screen entry under this command. The screen entry mode is discussed in detail in Section 3-3.

7. Compile Intermediate File
   This transforms the building file into the numeric file actually used by the LOADS module for simulation. The transformation process includes extensive error checking. If incorrect or undefined parameters are identified, they are reported to the user. Section 3-6 discusses this subject in further detail.

Section 3-3. SCREEN EDITING

All parameter input and editing is completed as interactive screen entry from the Display/Edit command (#6 in the Master Menu). First the desired building description data section is selected, then screen entry mode is initiated and used to enter new data or edit existing information.

A. DISPLAY/EDIT MENU

Once the Display/Edit command has been chosen, the Display/Edit menu, a list of all possible building description data sections, is displayed on the screen (see Figure 3-2). The user selects a building data section to view.

```
         SUNCODE                       DISPLAY/EDIT MENU

1 RETURN TO MASTER MENU

DISPLAY/EDIT ONE OF THE FOLLOWING (ARROW POINTS TO LAST DISPLAYED DATA)

=>2 RUNS           3 ZONES             4 INTERZONE          5 WINDOWS
  6 WALLS          7 TROMBE.WALLS      8 FANS               9 ROCKBINS
 10 SURFACES      11 HVAC.TYPES       12 TROMBE.TYPES      13 WALL.TYPES
 14 MASS.TYPES    15 PCM.TYPES        16 GLAZING.TYPES     17 BIN.TYPES
 18 FAN.TYPES     19 OVERHANG.TYPES   20 SIDEFIN.TYPES     21 SKYLINE.TYPES
 22 OUTPUTS       23 SCHEDULES        24 SEASONS           25 PARAMETERS
 26 STATIONS

SELECT YOUR CHOICE #
```

Figure 3-2. The Display/Edit Menu

Once a data section is selected, that section is displayed on the screen, and the editor automatically enters screen entry mode.

B. DATA STRUCTURE

Each building description file has a file label used for identification and one or more data sections. Each data section consists of a name, a heading, and one or more lines of data entry.

A data section heading contains titles for each parameter and a code that explains the length and type of data value expected for each parameter. The codes are:

AAAAAA - denotes a name with maximum length given by the number of A's

XXX.XX - denotes a number with the position of the decimal point fixed,

SSS.SS - denotes either a number or a name, used only for schedules.

Above each code appear the units for that data entry. English units or System International (S.I.) units are allowed. These are selected before entering the EDIT module. Each building description must use the same units throughout.

A complete list of all Input Data Sections with headers, English and SI units, default values, and explanations of the meanings of each data value can be found in the appendix. This appendix should be used whenever entering a building description or filling out an input data form.

C. SCREEN ENTRY MODE — EDITING COMMANDS

The screen entry mode is initiated automatically once a building section is being displayed. The screen editor has commands that provide for:

Adding new data lines to the building description,
Modifying existing data lines,
Deleting existing data lines,
Entering additional data lines,
Moving to other data sections to be edited, and
Returning to the Display/Edit menu.

When in screen entry mode, a prompt showing the allowed commands is displayed (see Figures 3-3 and 3-4). Note that the allowed commands differ slightly if the building section already contains some data. The command line is followed by the heading lines for the selected data section, lines of data (if previously entered) and a single arrow prompt (>). The user must respond to the prompt with a one letter command and carriage return, using any one of the commands shown at the top of the display. Any response other than one of the indicated command letters will be ignored.

```
COMMAND: Quit, Back, Fore, Add AND RETURN
```

Figure 3-3. Screen Entry Commands - No Existing Data in Building Section

```
COMMAND: Quit, Back, Fore, Add, Copy, Insert, Kill, Edit AND RETURN
```

Figure 3-4. Screen Entry Commands -- Data Already Existing in Building Section

Entering new data lines — the ADD command

To initially enter new data lines in the building section, use the ADD command. Type A and carriage return. Alternatively, strike the Home key. After the ADD command is entered, the prompt line at the top of the screen is changed, and now will indicate that you should enter the data values in the new data line. A double arrow prompt (>>) is then displayed.

Enter a line of data values, one value for each parameter shown in the heading line. The data values must be either:

1. an unsigned real number, or

2. a name (defined as a string of characters excluding commas and spaces, with the first character a letter A to Z).

Type each of the data values in the order shown by the heading lines. Separate the data values with a single comma and/or one or more blank characters.

As a user convenience, most of the parameters have default values. See the Input appendix for a list of all entries and defaults. If you wish to use the default value for the data entry, type two commas or a left-arrow (<). Note that any following characters for this entry are ignored, so that you can enter <NONE> or <ADEQ> and the correct default value (even if different from what was typed) is placed in the line. In some cases, you may want to default all of the values at the right end of a data entry line. To do so, simply enter carriage return after the last none-default entry. Some name entries (for example, ZONE NAME) have no default values; in these cases a user entry is mandatory. Failure to make a mandatory entry will result in an error message when the description is processed in the final stages of editing.

End the line of data values with a carriage return. You need not be concerned with directly positioning the new data values under their corresponding column headings. For example, the decimal point in the user entry need not be aligned with that in the heading. The program will automatically align all user entries, with numbers located by the position of the decimal point and with names left-justified. The total length of each data value, however, is fixed; entries that are too long will result in an error message.

If an error is detected, as when a number is entered in the position where a name is expected, or vice versa, or a number is outside the bounds indicated by a heading format, then a message appears indicating the type of error encountered and showing the entry where it was found. If an error message appears, the line is not added to the file. Retype the line with the appropriate correction.

When no errors are detected in the data line being entered, the line is rewritten on the display in its proper format and the double-arrow prompt (>>) is repeated. Another data line may now be added. When all new data lines have been added, type only carriage return in response to the double arrow prompt (>>). The display will again indicate the command options available, and the single arrow prompt (>) will be displayed.

When data lines already exist in the data section, commands other than ADD may be used to modify or delete existing lines, or add new lines.

Modifying an existing data line -- the EDIT command

To modify an existing data line, move the single arrow prompt (>) to the line to be altered, using the up and down arrow keys. Alternatively, the caret (^) and return keys may be used. When the prompt is on the desired line, type E for EDIT and carriage return. The prompt line at the top of the screen is changed, as with the ADD command, to indicate that the new data values for the line are to be entered. The double arrow prompt (>>) is displayed.

Enter a line of new data values, separating the values by commas or blanks, and terminate the line with a carriage return. Data values to be left unchanged can be indicated by two adjacent commas. A data value may be set to its default value by entering a left-arrow (<) in the corresponding position.

If an error is detected in the new data values, the action is the same as described above. That is, the type and position of the error are reported and the line must be retyped. If no error is detected, the modified line is displayed, and the single-arrow prompt (>) is repeated.

Deleting an existing data line -- the KILL command

To delete an existing line, again position the prompt to the desired line using the up and down arrow keys. Then type K for KILL and carriage return; the existing line will be removed from the file and the display.

Entering additional lines -- the ADD, INSERT and COPY commands

After having initially ADDed data lines to the section, more new lines may be entered. In response to the single arrow (>) prompt, type A for ADD (or strike the Home key), C for COPY or I for INSERT, and return.

The ADD command will place the new lines immediately below the line that the single arrow prompt is on. The action is identical to that described above for initially adding data lines.

The INSERT command will perform similarly, except that the new data line will be placed immediately above the line the prompt is on. In general, the order in which data lines occur is not significant. Schedules are the only exception.

If several lines with rather similar data values are being entered, the COPY command may be convenient. The COPY command is similar to the combined actions of the ADD and EDIT commands. That is, a new data line, which is a duplicate of the data line that the single arrow prompt (>) is on, is added to the section and the user may then edit the new line by specifying only those data values which are to be different. Hence data values which are entered as two adjacent commas in response to the double arrow prompt (>>) are set to the corresponding value in the line immediately above (i.e. in the line that the single arrow prompt (>) is on).

Moving to other data sections -- the FORE and BACK commands

You may move to other data sections within the building file. To move forward to the next section, type F for FORWARD and return, or type the Pg Dn key. To move back to the preceeding data section, type B for BACK and return, or type the Pg Up key. When the single arrow prompt is on the first data line in a section, and the up arrow is entered, the previous data section is selected. When the prompt is on the last data line, and the down arrow or return is entered, the next data section is selected. In some cases, more data lines may exist than can fit on the screen at one time. Here, the BACK (Pg Up) and FORE (Pg Dn) commands select the previous and next screen of data lines respectively.

Returning to the DISPLAY/EDIT menu -- the QUIT command

When editing is completed on the data section(s) originally selected, you may return to the DISPLAY/EDIT menu by typing Q for Quit or by typing the End key. From the DISPLAY/EDIT menu, any other data section may be selected for editing, or the user can return to the Master Menu.

Section 3-4. EXTERNAL EDITOR

Files prepared with another text editor may be read by the editing program and put into the standard file format. These files will be called "external" files. The first line is always interpreted as a label, not a section name. Each following line in an external file is recognized as one of three types by the first character in the line.

If the first character is an asterisk, the line is seen as a comment and/or header and is ignored.

If the first character is a space, the line is assumed to be a data entry line for the last defined data section; that is, each data entry line must be indented at least one space. As in interactive input, each parameter in the line should be separated by a comma and/or one or more spaces. Two adjacent commas or a left arrow (<) indicate a default value.

If the first character in the line is any character other than asterisk or space, the line is assumed to define a data section. The line should contain only the full data section name (e.g. WINDOWS). All data lines following will be assumed to be in this data section, until the next data section name is given.

In addition, the last line of the file must contain the following (starting in the first column, with no abbreviations):

```
END OF FILE
```

The file must be named <filename.BLG>. It should be read into the EDITS module workfile with command #2 (Read Building File). On input, each data entry will be checked for proper data parameters. Any error will cause the line to be ignored. Remember that the data values must be specified in exactly the same order as given in the standard headers.

Section 3-5. FILE HANDLING

The EDIT module accesses several type of disk files that store user building descriptions and control information for the program. Information that defines the format of a building description is contained in one file, SUNCODE.H. User building descriptions may be stored in two different types of files, each with its own format and purpose.

A. BUILDING DESCRIPTION FILES

Copies of prepared building description files may be saved from EDITS using the Master Menu command #3 (Write Building File). This produces a file of ASCII characters in a format that can be typed or printed by the operating system utility programs. An existing building description file can be accessed by reading it into EDITS with the Master Menu command #2 (Read Building File). Building description files are named <filename>.BLG.
NOTE: When leaving the EDIT module (command #1, Terminate Program), the user is offered a final opportunity to save work into a building description file.

B. LIBRARY FILES

Building description files will tend to contain some information that varies from simulation to simulation (i.e., the general layout of zones, surface orientations, etc.) and some information that tends to be the same for many simulations (i.e., wall material and glazing specifications, output formats, etc.). To streamline the input process, information that remains constant for a variety of simulations can be stored as library files. Library files can then be easily incorporated into future building descriptions; the data they contain need not be re-entered. Information such as the GLAZING.TYPES or WALL.TYPES data sections (any data section except OUTPUT, RUNS, ZONES, INTERZONE, WALLS, WINDOWS, TROMBE.WALLS, FANS or ROCKBINS) may be stored in this manner.

Library files are saved using the Master Menu command #5 (Write Library Format File). They are saved in a special format and cannot be accessed outside of the EDIT module. They can be read into the workfile with Master Menu command #4 (Read Library Format File). Library files are named <filename>.LIB.

The EDIT module uses library files to supplement the building description file during compiling. For any data entry referenced in the building description but not defined there, each of the library files will be searched in the order in which they were specified (see section 3-6), until the entry definition is found. The information recovered from the library files will be included in the run's summary output and is used by the simulation module, but it will not be copied into the building file.

C. INTERMEDIATE FILES

The Master Menu command #7 (Compile Intermediate File) creates a copy of the building description, this time in numeric format (with the coefficients defining the nodal network appended) for direct use by the LOADS module. This file is called <filename>.INT, and it is written in a format which is generally not accessible to the user. It is deleted automatically after the LOADS module has completed the simulation run.

Section 3-6. COMPILING THE BUILDING DESCRIPTION FILE

When you have completed all data entries and are ready to start a simulation run, you must execute the numeric processor section of the editor module. This section does additional error-checking on your building description, and translates the description into the numeric form required by the LOADS module.

If no errors are found, a temporary disk file is created that is read by the LOADS module and automatically deleted when the simulation run has been completed.

The preprocessor module is executed from the Master Menu with command #7 (Compile Intermediate File). The program responds with:

```
ENTER NAMES OF LIBRARY FILES (CARRIAGE RETURN FOR NONE)
```

If some of the component-level specifications for your run are contained in one or more libraries, enter the names of all the files to be used, each separated by a space or comma. If all the specifications for your run are contained in the current building description, just type carriage return.

The program will then request the filenames for the temporary file to be used as input to the loads program (<filename>.INT), and for the file to be used for summary output (<filename>.OUT).

The preprocessor then makes a two pass review of the contents of the building description. In the first pass, all references to component-level entries are checked to make sure that the referenced sections are present. As each library file is opened, the message:

```
SEARCHING <library filename>
```

is displayed. Data entries from library files are reported in the echo of inputs contained in the run summary output from the LOADS module, but will not be copied into the current building description file.

If any errors are discovered, the program displays the data line containing the error along with a description of the type of error, or a list of undefined components. The program than asks whether you wish to continue checking the remaining building description or abort the preprocessor. At the end of the first pass, if any errors were discovered, the preprocessor returns to the Master Menu. You may then edit the building description in the usual way.

If no errors are found in the first pass, the second pass checks for errors in the values specified for numeric parameters and the placement of fans and rockbins. If errors are found, they are reported and the program returns to the Master Menu. This pass will also calculate the thermal network node coefficients required by the loads program. It will ask for the number of time steps per hour to be used in the simulation with:

```
MINIMUM NUMBER OF STEPS PER HOUR = xx
DO YOU WISH TO USE THIS VALUE? (YES, NO, SUMMARY)
```

If you respond YES, the reported value is used. If you respond NO, a new value (which must be greater than the reported value) is requested. If you respond SUM, the minimum number of time steps required by each wall, Trombe wall, and rockbin in the building description is reported. Each is indicated by its line number in the data section in which it occurs.

Upon successful completion of the second pass, the program writes the temporary file of numeric values for use by the LOADS module, and returns to the Master Menu.

Section 3-7. ERROR MESSAGES

A. DATA ENTRY ERRORS

ERROR: IMPROPER RESPONSE. TRY AGAIN.
Message from MENU-MODE when menu selection number given by user is not one of the defined responses displayed on the screen.

ERROR: DATA SECTION NOT RECOGNIZED.

ERROR: DATA ENTRY NOT FOUND IN WORKING FILE.

ERROR: DATA ENTRY NOT FOUND IN ACTIVE SECTION.

ERROR: DATA ENTRY NOT FOUND IN SPECIFIED SECTION.

ERROR: TOO MANY DATA PARAMETERS IN LINE.

ERROR: DATA PARAMETER TOO LONG.

ERROR: DATA PARAMETER WRONG TYPE.

ERROR: ILLEGAL CHARACTER IN NUMERIC PARAMETER.

ERROR: INTEGER PORTION OF NUMERIC PARAMETER TOO BIG.

ERROR: DECIMAL PORTION OF NUMERIC PARAMETER TOO LONG.

ERROR: NO INFORMATION IN HELP FILE.

ERROR: HELP KEYWORD NOT FOUND.

ERROR ON FILE OPEN. RESPECIFY FILE.
Message given when user specified file in READ command is not found.

ERROR ON FILE OPEN. RESPECIFY FILE
Work file specified by user in OPEN command was not found.

ERROR: NO ACTIVE FILE.
Message given when user enters a command that requires a work file to be present, but none exists.

ERROR: COMMAND REQUIRES DATA VALUES.

ERROR: DATA SECTION FULL, NO MORE LINES MAY BE ADDED.

ERROR: DATA SECTION EMPTY.
Message given when user attempts to KILL an entry in an empty section.

ERROR ON MASTER HEADER FILE OPEN. COMMAND ABORTED.

ERROR: MASTER HEADER FILE OPEN WRONG TYPE. COMMAND ABORTED.

ERROR ON WORKING FILE OPEN. COMMAND ABORTED.

ERROR: TOO MANY DATA ENTRIES.
Message given when file being READ or merged (GET) contains too many data entries in a data section.

ERROR: LINE xx OF INPUT FILE IGNORED.
Message given when a data value error is encountered during a READ or GET operation.

ERROR: LINES xx TO xx OF INPUT FILE IGNORED.
Message given when an error during a READ or GET causes multiple lines to be ignored.

ERROR: DATA SECTION KEYWORD xxxxxxxxxx NOT RECOGNIZED.
Message given during a READ or GET when a line of the file begins with a character other than space or asterisk, and the line cannot be interpreted as a valid data section name.

B. PASS 1 OF PREPROCESSOR ERRORS

WARNING: ERROR ON OPEN OF LIBRARY FILE xxxxxxxxx.
Library file specified by user was not found.

ERROR: STACK OVERFLOW.
User building description contains too many name references.

ERROR: ZONE REFERENCED IN xxxxxxxx SECTION NOT DEFINED.

ERROR: THE FOLLOWING xxxxxxxxxx ARE NOT DEFINED:

ERROR: AT LEAST ONE ENTRY IN xxxxxxxxxxx SECTION IS REQUIRED.

ERROR: REQUIRED PARAMETER NOT GIVEN IN xxxxxxxxxx SECTION.
The line in error is echoed above, with a pointer indicating the position of the missing parameter.

ERROR: ILLEGAL DATE ENTRY IN xxxxxxxxxSECTION.

ERROR: SEASON NAME REQUIRED IN FIRST LINE OF SCHEDULE DEFINITION.

ERROR: HOUR OF DAY INDEX FOR SCHEDULE MUST BE FROM 1 TO 24.

C. PASS 2 of PREPROCESSOR ERRORS

ERROR: USE OF TROMBE WALLS CREATES TOO MANY ZONES.

ERROR: USE OF TROMBE WALLS CREATES TOO MANY WINDOWS.

ERROR: USE OF TROMBE WALLS CREATES TOO MANY WALLS.

ERROR: TOO MANY MASS NODES DEFINED.

ERROR: CONDUCTIVITY OF LAYER IN WALL xx =0.

ERROR: DENSITY OF LAYER IN WALL xx =0.

ERROR: SPECIFIC HEAT OF LAYER IN WALL xx =0.

ERROR: THICKNESS OF LAYER IN WALL xx =0.

ERROR: BOTH FRONT AND BACK SURFACE COEFFICIENTS IN WALL xx =0.

ERROR: TWO ADJACENT INFINITE CONDUCTIVITY LAYERS IN WALL xx.
Note: This error is only caused if one of the adjacent layers with infinite conductivity is a phase change material.

ERROR: ILLEGAL PLACEMENT OF FAN AND/OR ROCKBIN IN ZONE xx.

ERROR: SOLAR DISTRIBUTION IN ZONE xx DOES NOT ADD TO 1.0.

ERROR: VOLUME OR HEAT CAPACITY OF ROCKBIN xx =0.

CHAPTER 4. THE VIEW PROGRAM

Section 4-1. INTRODUCTION

The VIEW program enables the user to quickly access and view SUNCODE formatted output files. These files consist of the full building description, an output summary, and simulation results. They are sometimes very large; examining them by printing or typing them out may be a lengthy and inconvenient process. The VIEW program is capable of displaying output files on the screen (allowing the user to move easily from one section of output to another), producing simple graphics, and producing ASCII files for "importing" into other programs (particularly spreadsheets) for more extensive graphics and analysis.

As shown in Figure 4-1, the VIEW program consists of 5 menus. The main menu provides file handling capabilities, the macro menu facilitates use of macros, and the remaining three menus provide graphing capabilities. Each menu and its commands will be discussed in turn.

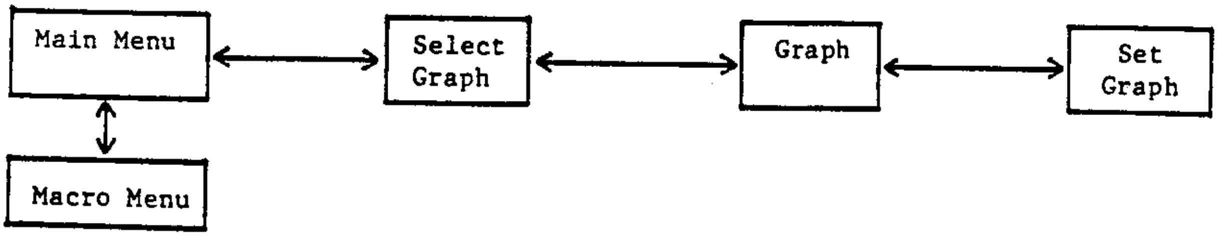


Figure 4-1. Menu Structure of VIEW

A. ENTERING VIEW

The VIEW program is entered by typing VIEW at the operating system prompt. The program will prompt for the name of a formatted SUNCODE output file. Enter the name of the file you wish to view (drive designations and directory paths are accepted). Once an acceptable filename has been entered, the main menu is displayed.

Section 4-2. THE MENUS

A. MAIN MENU

The Main Menu is shown in Figure 4-2. It provides file handling capabilities, allows examination of the specified output file, and enables the user to write output to an ASCII file.

```
Q)uit    I)nput    S)ummary    O)utput    G)raph    F)ile    M)acro    W)rite
```

Figure 4-2. Main Menu

Quit - Terminates VIEW and returns the user to the operating system.

Input - Allows the user to view the first section of the output file, the SUNCODE building description file used to produce the specified output. The data is viewed as it was entered, one section at a time. The user can scroll through the input by using the up arrow and the down arrow to move one line at a time, and Pg Dn and Pg Up to move to the next or the preceeding section of input.

Summary - Displays the building summary statistics given in the output file. The information given includes building steady state heat loss rates and thermal capacity.

Output - Allows the user to view specified sections of the SUNCODE output file. Use the up arrow and the down arrow to scroll up or down one line at a time, and Pg Up and Pg Dn to move to the next or the preceeding section of output.
This command is used to specify the output to be written to a LOTUS® file (see Write below) or graphed (see Graph below). The W)rite and G)raph commands are not active unless O)utput has already been used to specify the output section of interest.

File - Allows the user to choose a new SUNCODE output file to view.

Macro - Moves the user to the Macro menu.

Write - Writes the output section displayed on the screen to a LOTUS 1-2-3® import file. This command works only after the O)utput command has been used to specify a portion of output to write. The output is written into a file <filename>.PRN, which can be read into LOTUS.

Graph - Moves the user to the Select Graph menu, the first of three graphing menus. This command works only after the O)utput command has been used to specify a portion of output to graph.

LOTUS 1-2-3® is the registered trademark of the Lotus Development Corporation

B. SELECT GRAPH MENU

The Select Graph menu is shown in Figure 4-3. The specific output data to be graphed is selected within this menu.

```
Q)uit    S)elect    C)ancel    V)iew
```

Figure 4-3. Select Graph Menu

Quit - Returns the user to the Main menu.

Select - Enables the user to select which columns of output data are to be graphed. The program prompts with a graph number, and asks what column of data is to be plotted on that graph. It also asks for a label for the graph. If graph parameters have already been set, the program asks for a graph position. Only one column of data can be pictured on a single graph.
The S)elect command can be invoked repeatedly to assign data to up to 10 graphs.
This command must be used to select data for graphing before the V)iew command is invoked.

Cancel - Cancels all current selection made under the S)elect command. Once selections have been cancelled, no graphing is possible until new data for graphing is selected.

View - Moves the user to the Graph menu, the second of the three graphing menus. This command works only if S)elect has already been used to assign data to graphs.

C. GRAPH MENU

The Graph menu is used to actually make the desired plots of data. It is shown in Figure 4-4.

```
Q)uit    G)raph    S)et    E)dit
```

Figure 4-4. Graph Menu

Quit - Returns the user to the Select Graph menu.

Graph - Shows the graph(s) of the data. The graph will appear in default mode (a single line graph) unless specific graph parameters have previously been specified using the S)et command.

Edit - Allows editing of the graph displayed on the screen. This command is effective only after G)raph has been used to produce a graph on the screen. The cursor can be moved using the arrow keys; characters may be erased by spacing over them. To exit from edit mode, type <Control Q>.

Set - Moves the user to the Set Graph menu, the final of the three graphing menus. This command should be used to set graph parameters before the G)raph command is initially used.

D.SET GRAPH MENU

The Set Graph Menu is shown in Figure 4-5. It is used to set specific graph parameters such as screen position, axis scale, etc. Most of these settings are optional. However, if two or more graphs are to be displayed simultaneously, N)umber and P)osi must be set.

Q)uit N)umber P)osi Y)axis T)ype F)lip

Figure 4-5. Set Graph Menu

Quit - Returns the user to the Graph menu.

Number - Sets the number of graphs to be viewed simultaneously. The number of rows of graphs and number of columns of graphs must be specified (see Figure 4-6). A maximum of 4 rows and 4 columns may be specified. Essentially, this command divides the screen into blocks. It must be invoked before the P)osi command is used.

Posi - Positions each graph within the blocks set up under N)umber. The user is prompted for graph number, row position and column position. To position several graphs on a single screen, invoke P)osi repeatedly (see Figure 4-6).

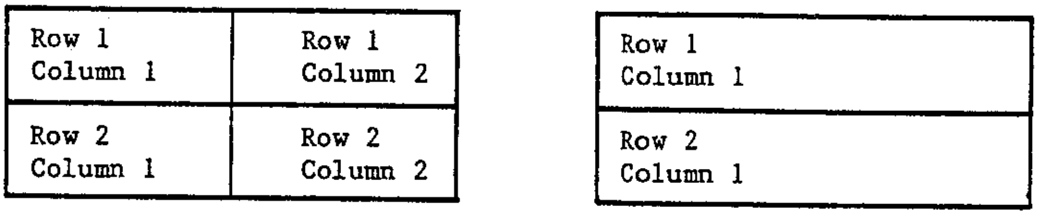


Figure 4-6. Positioning Several Graphs on a Screen

Yaxis - Changes the scale of the Yaxis. The user is prompted for graph number, maximum value and minimum value, and is offered the option to trim the plot to graph size.

Type - Changes the type of graph. The user is prompted for graph number and is asked to choose line, point or bar graph. The default is line graph.

Flip - Inverts the graph. The user is prompted for graph number.

E. MACRO MENU

The Macro menu is shown in Figure 4-7. It enables the user to store a sequence of commands in a disk file for ease in repeating them. These command sequences are called macros.

```
Q)uit    R)ead    W)rite    P)ause
```

Figure 4-7. Macro Menu

Quit - Returns user to the Main menu.

Read - Reads an existing macro file and executes the file. The program prompts for the name of a macro file. When the file is located, all commands stored in it are carried out. An internal timer within the program determines how long each screen will be displayed. This timer can be adjusted in the source code. While the macro is operating, no keystrokes are necessary; the program will automatically move to the next screen. When the macro is finished, the Main menu will be displayed.

Write - Begins and ends the recording of keystokes as a macro file. W)rite typed initially opens a macro file; the user will be prompted for a filename. All keystrokes typed thereafter will be recorded in the macro file. Any command in the VIEW program may be recorded. When W)rite is typed again, the macro file is closed and written to the disk. The Macro menu will then be displayed.

Pause - Slows down screen changes within macros. Pause can be invoked only while a macro file is being recorded. It causes the screen display to linger until: a) the maximum length of pause time is over, or b) the user enters any two keystrokes.

NOTE: The macro capability is very sensitive to changes in parameters within the VIEW program between the time the macro is created and the time it is executed. Macros should therefore be created and executed with consistency. For example, if the macro is created just after the user entered VIEW, it should be run just after the user enters VIEW.

Section 4-3. OTHER NOTES

The VIEW program was written and compiled using the 8087 version of Borland International's TURBOPASCAL® Version 2.0. It can be recompiled using TURBOPASCAL® Version 2.0 or higher. The files VIEW.PAS and SUNGRAF.PAS are necessary to recompile the program.

CHAPTER 5. SOFTWARE MODIFICATION

Section 5-1. INTRODUCTION

This chapter provides an outline of the program structure and information to aid in the modification of the program source code. Source code has been included with SUNCODE-PC to facilitate modification of the program to suit individual purposes. The authors take an ongoing interest in the program, and would like to be notified of modifications and identified bugs. Bug reports are particularly useful when enough supplemental information is provided to allow replication of the problem. Please note that the authors cannot support modified versions of the program.

Only users with a complete knowledge of FORTRAN should consider modifying the source code. The program has a long history, starting as a FORTRAN 66 mainframe program designed for teletype terminals. There is a great deal of orphaned code, and many orphaned variables. Modules using character arrays often interface with routines using integer arrays for the same data.

Section 5-2. SOURCE AND OBJECT CODE

SUNCODE consists of two main executable files, EDITS.EXE and LOADS.EXE. Source codes for each of these programs is supplied. Tables 5-1 and 5-2 list the source code files for the two programs and the subroutines they contain. The overall layout of the programs is shown in Figures 5-1 and 5-2.

SUNCODE 5.7 has been compiled using the large capacity version of the Microsoft FORTRAN Compiler 5.1. This version of the compiler and the resulting executable programs require MSDOS 3.0 or higher. We used the large memory model and floating point library (coprocessor required).

Two command files for the Microsoft NMAKE utility are included on the distribution disk (.MAK extension). These automate the compilation commands used to create SUNCODE, and should be consulted.

Section 5-3. MAXIMUM NUMBER OF INPUT ELEMENTS

For some applications, the maximum number of building elements (e.g., zones, walls, etc.) may need to be increased. Within the EDITS and LOADS modules, the arrays that store building description inputs, internally calculated values, and run output values are defined as having fixed (constant) sizes. To alter the maximum number of entries within any data section, the declared sizes of these internal arrays must be changed.

The sizes of the relevant arrays are specified in PARAMETER statements in SUNCOMM. This module is included in all other modules where array specifications are required. One array size PARAMETER is specified for each data section. The PARAMETER value is the maximum number of elements allowed in a run. For component type sections, i.e., WALL.TYPES, this is the maximum number of data entries which can be referenced.

The editor uses the MAXLI data structure to determine the number of entries that the user is allowed. This is located in EDATAA.FOR. For component sections, this should be changed to the parameter value entered in SUNCOMM. For component type sections, this should be the number of entries allowed to be edited, which is often more than the number allowed to be referenced in a building description.

An additional PARAMETER defines the maximum number of mass nodes allowed. Table 5-2 gives the name of the PARAMETER for each data section and its declared value in SUNCODE-PC Version 5.7.

Generally, the maximum size for any component or component type is 65. If any parameter is increased to be larger than 65, many code changes within the editor would be required to adequately increase the size of buffers and pointer arrays.

NOTE: If any aspects of SUNCOMM.FOR are changed, ALL source files for both LOADS and EDITS must be recompiled (the CHECKINT program must also be recompiled, if it is used).

Table 5-2 Source Code and Subroutines in the EDITS Modules and Utilities

EXECUTABLE FILE	SOURCE CODE FILES	SUBROUTINES
EDITS.EXE	EMAIN.FOR	----
	ECMPLR.FOR	CMPLR
	EEDITR.FOR	EDITR, LINKS, DCODE, FCHAR, FORMR, LISTR, FLINE
	EPASS2.FOR	EQMT2, SUNF2, TROM2, SCALE
	EFOPEN.FOR	FOPEN, FCLOS, NCODE
	EEDATA.FOR	----
	EEDATB.FOR	----
	EPASS1.FOR	PASS1, ERRS1, XLATR, PUSHR, POPPR, DATES
	EFILER.FOR	FSECT, MAKER, READR, WRITR
	ENUMBS.FOR	----
	ECOEF2.FOR	COEF2, STEP2, ERRS2
	ESCHDS.FOR	SCHDS
	EMENUS.FOR	MENUS, WMENU, ERASE, GETIN, ERROR
	EWORKR.FOR	FREAD, FWRIT, WFOPN, WFINP WFOUT, RSTOR, WFCLS
	EEDATC.FOR	----
	ESEDIT.FOR	SEDIT
	EPCOMD.FOR	PCOMD
	NEWCOMM.FOR SUNCOMM.FOR SUNMETA.FOR	Included in many of the source code modules above.
TMY2BIN.EXE	TMY2BIN.FOR	
CHECKINT.EXE	CHECKINT.FOR	
VIEW.COM	VIEW.PAS SUNGRAF.PAS	

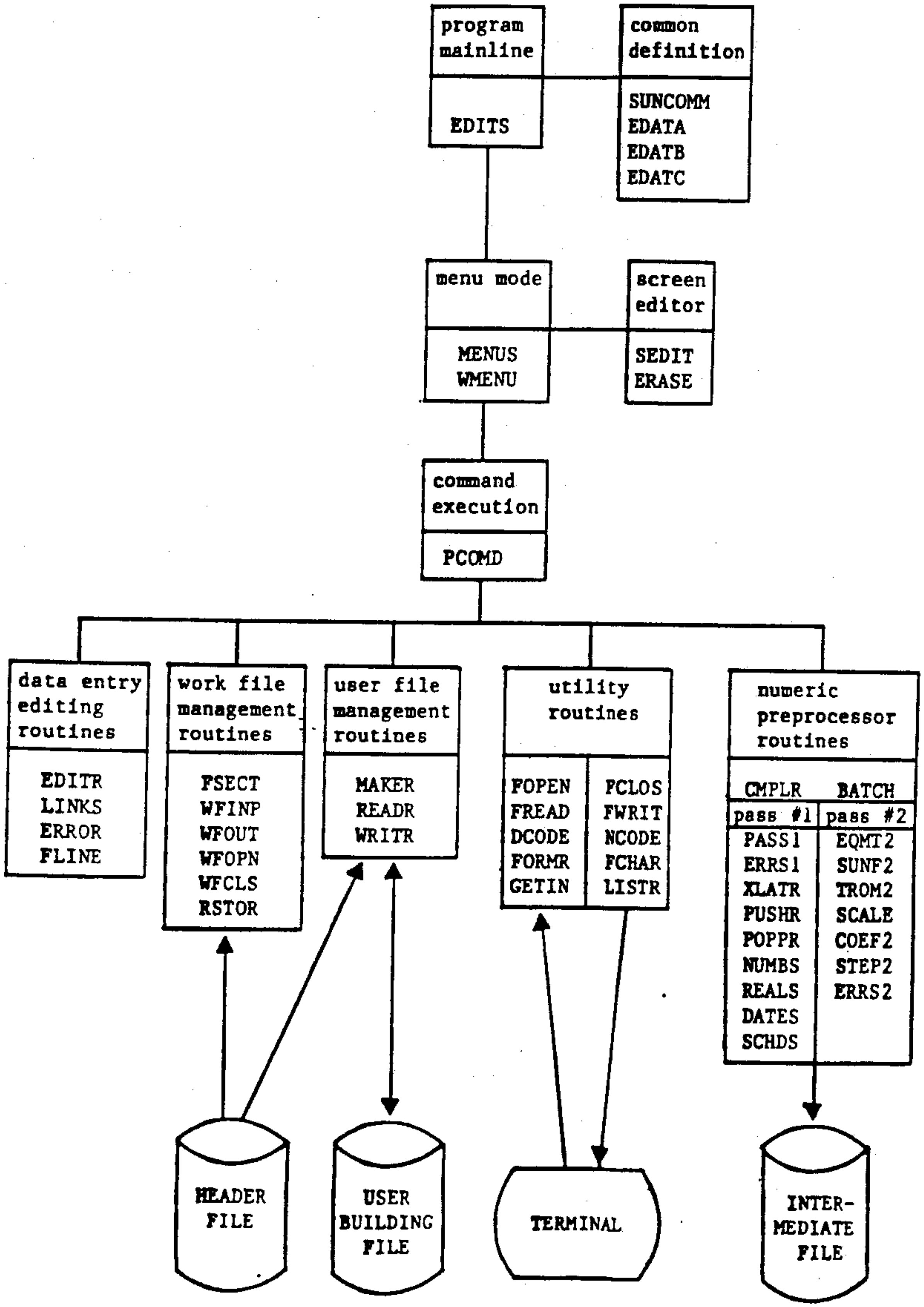


Fig. 5-1. General Organization of EDITS Program Modules.

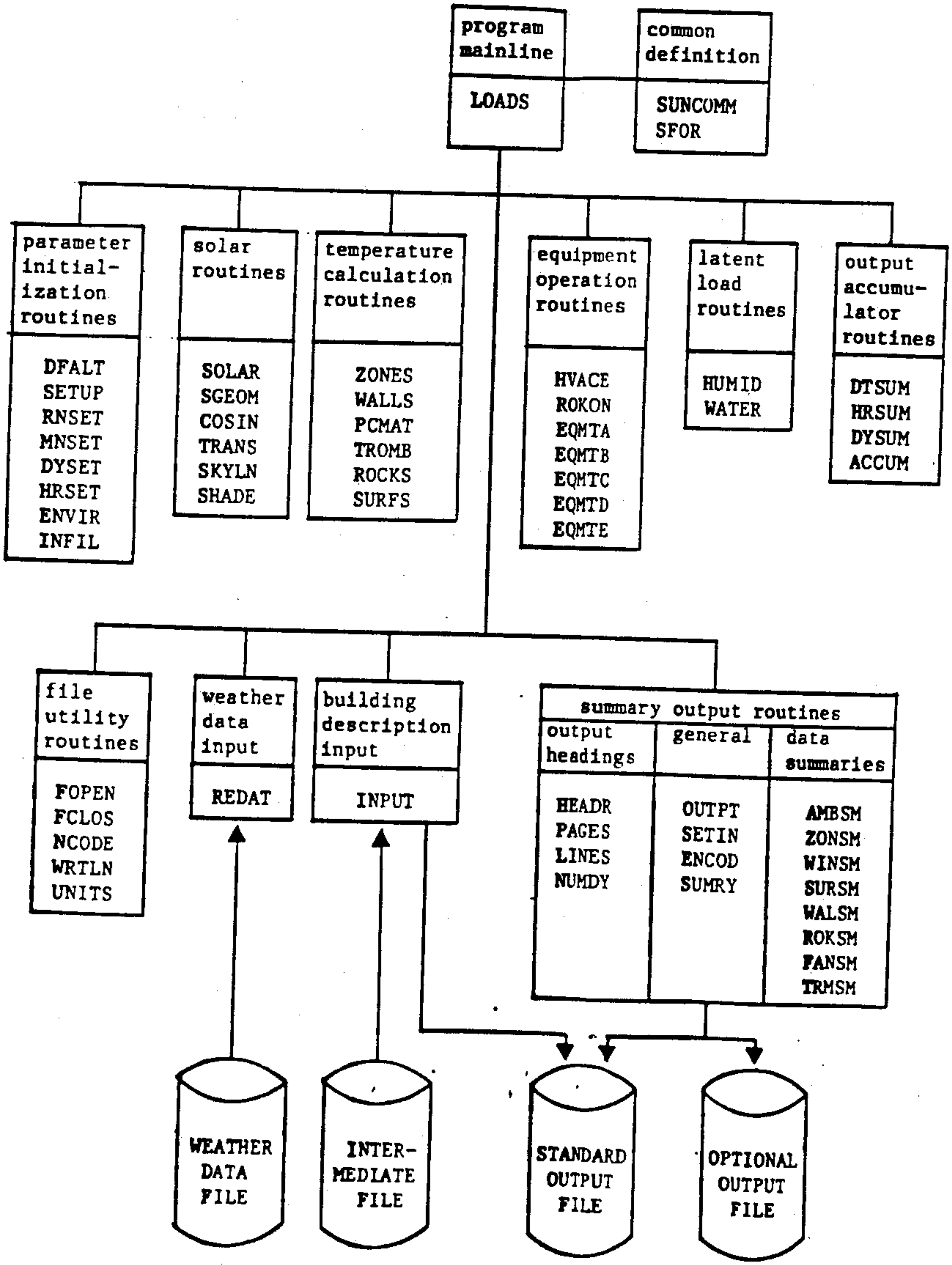


Fig. 5-2. General Organization of LOADS Program Modules.

B. COMPILING AND LINKING INFORMATION

Compiling

The mainline of each program (EMAIN.FOR for EDITS and LOADS.FOR for LOADS) was compiled using Version 3.3 of the Microsoft FORTRAN® Compiler. This version of the compiler requires MS-DOS 2.0 or greater. In order to compile the LOADS.FOR module, PAS2 of the compiler must be modified. This can be accomplished with the following command:

```
EXEMOD PAS2 /STACK A000
```

(The compiler's EXEMOD command increases the available stack size).

Source code for all other modules was compiled using Version 3.2 of the Microsoft FORTRAN® Compiler. This code could be recompiled using Version 3.3 with no adverse effects on program operation.

Linking

Object code was linked with the Microsoft FORTRAN® Version 3.3 linker, using the 8087.LIB.

The two files L.LNK and E.LNK (for LOADS and EDITS respectively) supply all the necessary information to the linker:

list of object files,
name of .EXE program to be created,
list of libraries to be used.

These files are supplied on Diskette 4. They may be called directly by the linker with the DOS command:

```
LINK < L.LNK
```

After linking, the LOADS.EXE and EDITS.EXE files were reduced in size using the Microsoft EXEPACK capability (Version 3.3 only).

If You Don't Have DOS 2.0 or Greater

All of SUNCODE may be compiled and linked using Version 3.2 of the Microsoft FORTRAN® compiler. This version does not require DOS 2.0. However, the resultant LOADS and EDITS programs will alter the time and date stamps on the input weather, building and header files each time they are run.

Section 5-3. MAXIMUM NUMBER OF INPUT ELEMENTS

For some applications, the maximum number of building elements (e.g. zones, walls, etc.) allowed for any simulation run may need to be increased. Within the EDIT and LOADS modules, the arrays that store building description inputs, internally calculated values, and run output values are defined as having fixed (constant) sizes. To alter the maximum number of entries within any data section, the declared sizes of these internal arrays must be changed.

The sizes of the relevant arrays are specified in PARAMETER statements in the common block module SUNCOMM.FOR. This module is then included in all other modules that access the common block. One array size PARAMETER is specified for each data section. The PARAMETER is defined as the maximum number of data entries allowed in that section. An additional PARAMETER defines the maximum number of mass nodes allowed. Table 5-2 gives the name of the PARAMETER for each data section and its declared value in SUNCODE-PC Version 5.4.

The arrays affected by the PARAMETERS listed in Table 5-2 are defined to be contained within even larger arrays (by EQUIVALENCE statements in the relevant program modules). Hence, altering the sizes of the individual "subarrays" will require that the size of the larger array also be changed. These "superarray" sizes are also specified by PARAMETER statements in the SUNCOMM.FOR module. The following arrays are affected:

MAXWORK	maximum number of lines to be stored in a workfile
MAXO	total number of real output values for daily, monthly and annual output levels
NNUMB	number of data sections used by LOADS
NINTS	total number of integer inputs in the intermediate file (including text strings)
NRELS	total number of real inputs in the intermediate file

Table 5-3 gives the factor by which these parameters must be increased (decreased) for a given increase (decrease) in one of the parameters controlling an individual data section. For example, if the parameter MWAL is increased from 40 to 50, then NINTS should be increased by (50 - 40)*6 = 60 (where the value 6 is taken from the NINTS multiplier column in Table 5-2), NRELS should be increased by (50 - 40)*19 = 190, and MAXO should be increased by (50 - 40)*11 = 110.

NOTE: If any array sizes are changed, ALL source files for both the LOADS and the EDITS programs must be recompiled (the CHECKINT program must also be recompiled, if it is used).

Table 5-3. PARAMETER Values Defining Maximum Number of Elements Allowed

Input Data Section	PARAMETER Name	Value in V5.4	Multipliers			No. of Values in V5.4		
			NINTS	NRELS	MAXO	INTGRS	REALS	OUTPUT
RUNS	MRUN	10	23	2	0	230	20	0
ZONES	MZON	10	7	7	26	70	70	260
INTERZONE	MINT	20	2	3	0	40	60	0
WINDOWS	MWIN	20	3	4	18	60	80	360
WALLS	MWAL	40	6	19	11	240	760	440
TROMBEWALLS	MTRM	5	3	6	3	15	30	15
FANS	MFAN	5	4	0	5	20	0	25
ROCKBINS	MROK	5	4	9	12	20	45	60
SURFACE.TYPES	MSUR	10	3	4	4	30	40	40
HVAC.TYPES	MHVC	10	0	7	0	0	70	0
TROMBE.TYPES	MTWT	5	3	5	0	15	25	0
WALL.TYPES	MWLT	20	6	6	0	120	120	0
MASS.TYPES	MMAS	20	1	7	0	20	140	0
PCM.TYPES	MPCM	20	0	0	0	0	0	0
GLAZING.TYPES	MGLZ	10	0	6	0	0	60	0
BIN.TYPES	MBIN	5	4	4	0	20	20	0
FAN.TYPES	MFNT	5	0	2	0	0	10	0
OVERHANG.TYPES	MOHT	5	0	4	0	0	20	0
SIDEFIN.TYPES	MSFT	5	0	4	0	0	20	0
PROFILE.TYPES	MPRO	5	0	11	0	0	55	0
OUTPUTS	MOUT	10	7	0	0	70	0	0
SCHEDULES	MSCH	40	2	24	0	80	960	0
SEASONS	MSEA	20	5	0	0	100	0	0
PARAMETERS	MPAR	5	4	7	0	20	35	0
STATIONS	MSIT	10	26	3	0	260	30	0
Total number of mass nodes	MNOD	200	1	4	0	200	800	0
	NINTS					1630		
	NRELS						3470	
	MAXO							1220

CHAPTER 6. TECHNICAL ALGORITHMS

Section 6-1. INTRODUCTION

This chapter presents a discussion of the technical algorithms which are used by the LOADS program. It contains separate sections describing a) the handling of solar energy inputs to the building, b) the calculation of temperatures for zones, walls and rockbins, c) the operation of the controllers for heating, venting, and cooling equipment, and for fans between zones or connected to rockbins, and d) the treatment of humidity levels within the building.

A building is conceptualized as one or more zones. Each zone has independent solar inputs and independent heating, cooling, and ventilation equipment and controls. Each zone may also contain a rockbin. Zones may be connected by walls or pure thermal conductances. In addition, thermostatically controlled fans may connect zones.

The major simplification in the conceptual model of zones is the use of a single zone temperature node. The program does not allow for direct radiation heat transfer between walls of a zone with separate calculation of convective heat transfer to the zone air. Instead, the zone is represented by a single temperature node. All heat transfer paths are connected to this central node. Walls are connected by a constant heat transfer coefficient to the central zone node. This heat transfer coefficient includes both convective and radiative heat transfer. This simplification avoids the calculation of radiation view factors (which would also require a three-dimensional building description in the input) and the solution of a radiosity matrix at each time step.

The central zone temperature is not really the air temperature. It is a conductance-weighted average of the all temperatures which affect the zone. In the simple case where there are no pure resistances or fans in the zone, the zone temperature is a weighted average of the surface temperatures. In this case it is, in effect, a form of mean radiant temperature. In some circumstances, the central node temperature may differ significantly from the true "air" temperature. Reference [1] contains an extensive discussion of the impact of the central node assumption. It is shown that, with proper calculation of the combined surface coefficients, the resulting error in temperature is typically comparable to that produced by differences in radiation transfer resulting from the detailed modeling of furniture in the zone. For convenience, the central zone temperature is referred to as the zone "air" temperature in this program.

Walls are coupled to the zone air node by constant coefficients which include the combined effects of convection and radiation heat transfer. New node temperatures are determined in each wall independently, using explicit finite differences. Equipment operation is controlled on the basis of the zone air temperature node. However, only equipment loads are calculated, no attempt has been made to model the actual performance of the equipment.

The general order of calculations within the LOADS program is illustrated in Figure 6-1. The "psuedo-code" fragment presented in the Figure shows only the heart of the calculations, those which are performed for each hour of the simulation run. First, solar intensity values read from the weather data file are used to calculate the amount of solar energy received on exterior surfaces and transmitted into the building. These solar intensity values are held constant for each time step within the hour. Air infiltration rates are set separately for each zone, at a user-specified constant value or as a function of current wind speed and the difference between ambient air temperature and the zone air temperature at the end of the previous hour. These infiltration rates are also held constant over the hour.

For each time step within the hour, new node temperatures are first calculated for each mass wall defined. These are based on the temperature of the node on either side at the end of the previous timestep (i.e., the "old" temperature). For nodes at the surface of the wall, the old air temperature is used in the calculation. Next, the program accumulates the energy flows from each zone air node to those elements having temperatures that are considered "fixed". These include the ambient and ground nodes, mass walls, and passive energy flows to rockbins.

Two levels of iteration may be used in the calculation of new zone air temperatures, depending on the building configuration. Iteration is used in the calculation of new zone air temperatures whenever direct energy flow paths (non-mass walls or loss coefficients) are defined between any interior zones. In this case, energy flows between zones are first calculated based on old zone air temperatures. Then, the equipment controllers for heating, venting, cooling (HVAC), and fans are used to set equipment operation and calculate new zone air temperatures. This process is repeated until the new air temperature for each zone differs from the previous value by less then a user-specified constant. When iteration is not required, the equipment controllers calculate the new air temperature directly.

Nested within the zone air temperature iteration, is a second level of iteration which is used when Trombe walls with natural convection air flow are present. In this case, the rate of air flow through the Trombe is calculated as a function of the most recent temperatures for the zone air node and the Trombe air gap node. A new air gap temperature is then calculated based on the old zone air temperature and the air flow rate. If required, this calculation of air flow rate and air gap temperature is iterated, holding the zone air temperature constant.

After new air temperatures for all zones are determined, new wall surface temperatures are calculated. For mass walls, these are based on the new air and new wall node temperatures. When a mass wall has a node at the surface of the wall (i.e., no thermal resistance from the mass wall surface to the first node), the surface temperature is taken as the first node temperature. For massless walls, the surface temperatures are calculated as a function of the new air temperatures on either side of the wall.

Next, if one or more rockbins are defined, they are checked to determine whether all or a part of the zone's heating load can be supplied by the rockbin. After the operational state of the rockbin's charge and discharge fans are determined, new node temperatures for the rockbin are calculated. Finally, once at the end of each hour, new zone humidity ratios and cooler latent loads are calculated, if they have been requested by the user.

In the following discussion of algorithms, we have attempted to use relatively simple notation, rather than a completely rigorous mathematical format. The equations presented, and particularly the variable names used, are not intended to have a direct one-to-one correspondence with those in the software. A few special symbols are used in this chapter, including "min [...]" and "max [...]" representing, respectively, the minimum and maximum of the values contained within the brackets. Also, "+oo" and "-oo" are used to indicate plus and minus infinity.

```
For hour := 1 to 24 do

    begin
    calculate solar input variables [Sub. SOLAR, et. al.];
    distribute solar gains within zones [Sub. HRSET];
    calculate infiltration rates for each zone [Sub. INFIL];
    For timestep := 1 to number of timesteps do

        begin
        calculate new wall node temperatures [Sub. WALLS];
        accumulate fixed energy flows: to mass walls, passively to rockbins,
             and to ambient and ground [Sub. ZONES];
        Repeat

            For zone := 1 to number of zones do

                begin
                If zone has Trombe then

                    Repeat

                        calculate Trombe thermocirculation [Sub. TROMB];
                        calculate Trombe air gap temperature [Sub. TROMB];

                    Until air gap temperature converged;

                accumulate interzone energy flows [Sub. ZONES];
                end;

            set fan and rockbin charge fan operation [Sub. EQMTA, et. al.];
            set HVAC equipment operation [Sub. HVACE];
            calculate new zone air temperatures [Sub. HVACE];

        Until zone air temperatures converged;
        calculate wall surface temperatures [Sub. SURFS];
        set rockbin discharge operation [Sub. ROKON];
        calculate rockbin node temperatures [Sub. ROCKS];
        end;

    calculate zone humidity ratios and cooler latent loads [Sub. WATER];
    end.
```

Figure 6-1. General Order of LOADS Program Calculations

Section 6-2. SOLAR ALGORITHMS

A. DISCUSSION

The program provides for extensive treatment of the thermal effects of solar energy on buildings. The model processes two hourly solar intensity values from the weather data file, global horizontal intensity and direct normal intensity. These parameters are held constant at an hourly input value for each time step within the hour. The position of the sun is determined each sunlit hour by calculating its altitude and azimuth angles. Most of the solar geometry formulas used here are taken from Reference [5]. The declination formula is taken from Reference [2]. The global horizontal intensity is separated into a direct ("beam") component and a diffuse component. The horizontal direct component is calculated using the space angle between a vertical and the parallel solar rays ("zenith angle"). The diffuse component is then taken as the difference between the global and direct.

The user may define the elevation of the horizon as seen from the building site in each of several segments of the sky centered about due south. This has the effect of completely blocking the direct component on all orientations and parts of the building when the sun is below the horizon. In this case, the diffuse component of the horizontal intensity is not altered, but the diffuse component on the other orientations is reduced due to a smaller ground reflectance contribution.

The user may specify several exterior surfaces of the building for which the effects of solar energy are to be considered. A surface is defined by its dimensions, the angle of its tilt from horizontal, and the azimuth angle of an outward pointing ray which is normal to the surface. Each surface is composed of one or more window and/or wall elements. An overhang and left and right sidefins may be separately defined for each surface, which will shade the direct component of solar energy on the surface.

The intensity of the direct solar component on each exterior surface specified is determined (as for horizontal) by calculating the space angle between the solar rays and an outward pointing ray normal to the surface ("angle of incidence"). For the diffuse component, the view factors from the building surface to the skydome and to the ground in front of the surface are determined. This one-time calculation does not consider any skyline profile angles specified, nor any overhangs, sidefins, or other surfaces defined for the building. Sky diffuse radiation is assumed to be evenly distributed ("isotropic"). The ground is taken as an infinite horizontal plate with a uniform, user-specified reflectivity.

Overhangs and sidefins are assumed to be infinite in length, hence no edge effects are considered. The shading effects of an overhang and sidefin are calculated separately, and then adjusted to account for the area contained in both shadows. The extent of shading is first calculated for the entire surface, and then separately for each window contained in the surface. The difference, total shading minus the sum of shading on all windows, is then applied evenly to all wall elements contained in the surface.

For direct and diffuse solar energy incident on each window, the fraction transmitted through the window (termed "shortwave") and the thermal gain due to energy absorbed within the windows (termed "longwave") are separately calculated. This window model is taken from Reference [3]. It assumes that the window is composed of one or more layers of identical glazing. For solar energy absorbed in the window, a maximum of four layers are considered. Hence the longwave thermal gain to the building will be understated when more than four glazing layers are specified. The window model uses the calculated angle of incidence for direct solar, and a user-specified constant angle for the diffuse component. The transmitted and absorbed fractions for both direct and diffuse components are reduced by any shading coefficient specified for the window.

The inward flowing fraction of energy absorbed within the window, or longwave radiation, is used as a thermal gain to the central air temperature node for the zone in which the window is contained. All shortwave solar energy transmitted through the window is assumed to be evenly distributed within the zone; the internal geometry of the zone is not considered. However, the energy transmitted through windows may be apportioned by the user to building elements within the zone. First, a fraction of the energy transmitted through all windows in a given zone may be "redirected" to any other zone as a simplified accounting for the transparent elements between zones.

After all such interzone solar transfers are considered, the total shortwave solar energy available in each zone may be divided into a fraction which is removed from the zone (losses due to reflection out the windows), a fraction which is immediate thermal gain to the zone air temperature node (to account for energy striking lightweight surfaces), and a fraction which strikes each wall defined within the zone. For solar energy incident on any interior wall containing one or more mass nodes, the model automatically allocates a portion of the solar energy which is used as an immediate thermal gain to the zone air temperature node, with the remainder absorbed by the first mass node. In the case of solar energy incident on massless walls, the energy is divided into thermal gains to the zone air temperature nodes on either side of the wall.

B. SOLAR POSITION

Subroutine

[DYSET] $dec = 23.45 * PI * sin(2 * PI * (day + 284)/365)/180$

[DYSET] $hourcrit = arccos(sin(dec) * cos(lat)/(cos(dec) * sin(lat)))$

[SGEOM] $hourangle = (12.5 - hour) * PI/12$

[SGEOM] $cos(zenith) = sin(dec)*sin(lat) + cos(dec)*cos(lat)*cos(hourangle)$

[SGEOM] $altitude = arcsin(cos(zenith))$

[SGEOM] $azim = arcsin(cos(dec) * sin(hourangle)/cos(altitude))$

[SGEOM] $azimuth = azim$, if $|hourangle| \le hourcrit$

$= (PI-|azim|)*azim/|azim|$, if $|hourangle| > hourcrit$

where:

PI = 3.1415927
day = day number in year 1,2,...,365
lat = station latitude
dec = daily solar declination
hourcrit = hourangle when sun crosses the E-W line, i.e. the azimuth = ± 90 degrees
hour = hour number in day 1,2,...,24
cos(zenith) = cosine of solar zenith angle
altitude = solar altitude angle
azim = temporary calculation for azimuth
azimuth = solar azimuth angle

C. SKYLINE SHADING — Subroutine SKYLN

$skyshade = 0$, if $altitude \le horizon(i)$ for i such that
$S(i) - (R/2) < azimuth < S(i) + (R/2)$
$= 1$, else

where:

i = index value 1,2,...,11

S(i) = azimuth angle given in heading of SKYLINE.TYPES input data section for i=1,...,11

R = range between azimuth angles given in SKYLINE.TYPES = 20 degrees

horizon(i) = user input value for horizon elevation under SKYLINE.TYPES input data section for i = 1,...,11

skyshade = skyline shading multiplier

D. HORIZONTAL INTENSITY

Subroutine

[SOLAR] dirhor = min [TH , cos(zenith)*DN]

[SOLAR] difhor = TH - dirhor

where:

TH = total horizontal radiation, read from weather tape

DN = direct normal radiation, read from weather tape

dirhor = direct component of horizontal radiation

difhor = diffuse component of horizontal radiation

E. INTENSITY ON ARBITRARY SURFACE

[SETUP] skyview = .5 * (1 + cos(tilt))

[SETUP] grdview = .5 * (1 - cos(tilt))

[COSIN] cos(incidence) = cos(tilt)*[sin(dec)*sin(lat) + cos(dec)*cos(lat)*
cos(hourangle)] + sin(tilt)*[cos(dec)*sin(az)*sin(hourangle) +
cos(az)*(sin(lat)*cos(dec)*cos(hourangle) - sin(dec)*cos(lat))]

[SOLAR] direct = cos(incidence) * DN * skyshade

[SOLAR] diffuse = skyview*difhor + grdview*reflect*(difhor + dirhor*skyshade)

where:

skyview = view factor from surface to skydome

grdview = view factor from surface to ground

tilt = angle between horizontal and surface

az = azimuth angle of outward pointing ray normal to surface

cos(incidence) = cosine of angle of incidence of solar rays on surface

direct = direct component of radiation on surface

diffuse = diffuse component of radiation on surface

reflect = user-specified ground reflectance value

F. OVERHANG AND SIDEFIN SHADING -- Subroutine SHADE

$\mathrm{gamma} = \mathrm{azimuth} - \mathrm{surfaz}$

If $\mathrm{gamma} \neq 0$

then

$$\mathrm{Do} = \frac{((\mathrm{Oh} * \tan(\mathrm{altitude})/\cos(\mathrm{gamma})) - \mathrm{Ov})}{(\sin(\mathrm{tilt}) + \cos(\mathrm{tilt}) * \tan(\mathrm{altitude})/\cos(\mathrm{gamma}))}$$

else $\mathrm{Do} = \mathrm{Oh}/\cos(\mathrm{tilt})$, if $\mathrm{tilt} \neq 90$ degrees
$= 0$, if $\mathrm{tilt} = 90$ degrees

with Do constrained such that $0 \leq \mathrm{Do} \leq \mathrm{H}$

$$\mathrm{Ds} = -\mathrm{So} + [\mathrm{Sh}*\sin(\mathrm{gamma})/(\cos(\mathrm{gamma})*\sin(\mathrm{tilt}) + \tan(\mathrm{altitude})*\cos(\mathrm{tilt}))]$$

with Ds constrained such that $0 \leq \mathrm{Ds} \leq \mathrm{L}$

$\mathrm{Ao} = \mathrm{Do} * \mathrm{L}$

$\mathrm{As} = \mathrm{Ds} * \mathrm{H}$

$\mathrm{Aos} = \mathrm{Ao} + \mathrm{As} - \mathrm{Ds} * \mathrm{Do}$

$\mathrm{sunfr} = 1 - (\mathrm{Aos}/(\mathrm{L} * \mathrm{H}))$

where:

azimuth = solar azimuth angle
surfaz = surface azimuth angle
gamma = angle between the solar and surface azimuths
altitude = solar altitude angle
tilt = surface tilt from horizontal
Oh = horizontal projection of overhang
Ov = vertical distance from top of surface to overhang
Do = distance from top of surface to bottom of shadow cast on surface by overhang
Sh = height of sidefin above plane of surface
So = distance from edge of surface to sidefin
Ds = distance from edge of surface to edge of shadow cast on surface by sidefin
L = length of surface
H = height of surface
Ao = area shaded by overhang only
As = area shaded by sidefin only
Aos = total area shaded by overhang and sidefin
sunfr = fraction of surface not shaded by both overhang and sidefin

G. WINDOW TRANSMISSIVITY -- Subroutine TRANS

$$n = 2*\ layers - 1$$

$$r = \arcsin(\sin(alpha)/refraction)$$

$$path = layers * thick/\cos(r)$$

$$x = ((refraction-1)/(refraction+1))^2$$

$$u = (\sin(alpha-r)/\sin(alpha+r))^2$$

$$v = (\tan(alpha-r)/\tan(alpha+r))^2$$

$$\begin{aligned} y &= (1-x)/(1+n*x), \text{ if } alpha = 0 \\ &= .5*[(1-u)/(1+n*u) + (1-v)/(1+n*v)], \text{ if } alpha \neq 0 \end{aligned}$$

$$tau = y * e^{(-K*path)}$$

where:

alpha = angle of incidence of solar rays
layers = number of layers of identical material in glazing
thick = thickness of one layer of glazing material
refraction = index of refraction of glazing material
K = extinction coefficient of glazing material
tau = transmissivity of glazing assembly for solar rays striking at an angle of incidence = alpha

H. WINDOW ABSORPTIVITY -- Subroutine TRANS

$$\begin{aligned} w &= (1-x)/(1+x), \text{ if } alpha = 0 \\ &= .5*[(1-u)/(1+u) + (1-v)/(1+v)], \text{ if } alpha \neq 0 \end{aligned}$$

$$z = w * e^{(-K*path/layers)}$$

$$abs = \sum_{i=1}^{j} (Aij*z^{i-1})$$

where (in addition to the definitions in Section G):

abs = inward flowing fraction of solar energy absorbed in the glazing for rays striking at an angle of incidence = alpha

j = min [4,layers]

Aij = coefficients

i	j 1	2	3	4
1	0.23	0.17	0.13	0.11
2	-	0.63	0.47	0.39
3	-	-	0.76	0.62
4	-	-	-	0.83

I. WINDOW CALCULATIONS — Subroutine SOLAR

$$\text{dirtrans} = \text{direct} * \text{tau(angle)} * \text{SC} * \text{Ag} * \text{sunfr}_g$$

$$\text{diftrans} = \text{diffuse} * \text{tau(difang)} * \text{SC} * \text{Ag}$$

$$\text{dirabs} = \text{direct} * \text{abs(angle)} * \text{SC} * \text{Ag} * \text{sunfr}_g$$

$$\text{difabs} = \text{diffuse} * \text{abs(difang)} * \text{SC} * \text{Ag}$$

where:

angle = angle of incidence of solar rays on window
SC = user-specified window shading coefficient
difang = user-specified angle of incidence for diffuse solar
dirtrans = direct solar transmitted through window
diftrans = diffuse solar transmitted through window
dirabs = inward flowing fraction of direct solar absorbed by glazing
difabs = inward flowing fraction of diffuse solar absorbed by glazing
Ag = area of window
sunfr_g = fraction of window area not shaded by overhang/sidefin
tau(angle) = transmissivity of glazing at angle of incidence = angle
abs(angle) = absorptivity of glazing at angle of incidence = angle

J. EXTERIOR WALL SOLAR ABSORBED — Subroutine SOLAR

$$\text{ABSw} = \text{Fw} * [\text{As} * (\text{direct} * \text{sunfr}_s + \text{diffuse}) - \sum_g (\text{Ag} * (\text{direct} * \text{sunfr}_g + \text{diffuse}))] * [\text{Aw}/(\text{As} - \sum_g \text{Ag})]$$

where:

the summations are over all windows in the surface containing the wall
As = area of surface containing the wall
Ag = area of a window contained in surface
Aw = area of wall w
direct = direct component intensity
diffuse = diffuse component intensity
Fw = user-specified exterior wall absorptivity
ABSw = solar energy absorbed on exterior of wall w
sunfr_s = fraction of surface area not shaded by overhang or sidefin
sunfr_g = fraction of window area not shaded by overhang or sidefin

K. INTERIOR DISTRIBUTION OF SOLAR -- Subroutine HRSET

$$\mathrm{QTRANSi} = \sum_{g} (\mathrm{diftrans} + \mathrm{dirtrans})$$

$$\mathrm{QAVAILi} = \mathrm{QTRANSi} - \sum_{z} (\mathrm{Fij} * \mathrm{QTRANSj}) + \sum_{z} (\mathrm{Fji} * \mathrm{QTRANSj})$$

$$\mathrm{QLOSTi} = \mathrm{FLi} * \mathrm{QAVAILi}$$

$$\mathrm{QSOLWALi} = \mathrm{FWi} * (1.0/(\mathrm{Hi}*\mathrm{Ri} + 1.0) * \mathrm{QAVAILi}$$

$$\mathrm{QSOLZONi} = \mathrm{QAVAILi}*\mathrm{FAi} + \sum_{g} (\mathrm{difabs}+\mathrm{dirabs}) + \sum_{m}[\mathrm{FWi}*\mathrm{QAVAILi}*\mathrm{Hi}*\mathrm{Ri}/(\mathrm{Hi}*\mathrm{Ri}+1.0)]$$
$$+ \sum_{r} [\mathrm{FWi}*\mathrm{QAVAILi}*(\mathrm{R}*\mathrm{Hi}*\mathrm{Hj}+\mathrm{Hi})/(\mathrm{R}*\mathrm{Hi}*\mathrm{Hj}+\mathrm{Hi}+\mathrm{Hj}) + \mathrm{FWj}*\mathrm{QAVAILj}*\mathrm{Hi}/(\mathrm{R}*\mathrm{Hi}*\mathrm{Hj}+\mathrm{Hi}+\mathrm{Hj})]$$

where:

$\sum_{z}$ = sum over all zones in building

$\sum_{g}$ = sum over all windows in zone i

$\sum_{m}$ = sum over all massive walls in zone i

$\sum_{r}$ = sum over all massless walls between zone i and zone j

QTRANSi = total solar energy transmitted into zone i
Fij = user-specified solar transfer fraction from zone i to zone j
QAVAILi = total solar energy available in zone i
FLi = user-specified solar fraction lost for zone i
QLOSTi = solar energy lost from zone i
FWi = user-specified fraction of sun to wall on side connected to zone i
Hi = wall surface coefficient on side connected to zone i
Ri = thermal resistance from wall surface connected to zone i into first mass node
QSOLWALi = solar absorbed by mass wall on side connected to zone i
FAi = user-specified fraction of solar to air node in zone i
R = thermal resistance from surface to surface for massless wall
QSOLZONi = total solar energy input to air temperature node in zone i

Section 6-3. TEMPERATURE ALGORITHMS

A. ZONE AIR TEMPERATURE

The program uses a central air temperature node for each zone defined by the user. Zone air is assumed to be well-mixed with no temperature stratification. This zone air temperature is used as the primary control for all equipment operation.

The energy balance equation defining the zone air temperature, T, can be written as:

$$\begin{aligned} &Qwall + Qpass + Qtc + Qzone + Qwindow + Qamb + Qgrd + Qinf + Qsolzon \\ &\quad + Qappli + Qfan + Qrock + Qheat - Qvent - Qcool = 0 \end{aligned}$$

where:

Qwall = energy flow between zone and enclosing mass walls (including Trombe walls)

$$= \sum_{walls} (UAwall * (Twall - T))$$

where:

Twall = temperature of first mass node in wall

UAwall = constant conductance from zone to first mass node in wall

$$= Awall * H/(R * H + 1.0)$$

where:

Awall = area of wall

H = wall surface coefficient

R = thermal resistance from wall surface to first mass node

Qpass = passive (non-controlled) energy flow between zone and rockbins connected to it

$$= \sum_{rocks} (UApass * (Trock - T))$$

where:

UApass = user-specified rockbed passive conductance value

Trock = average of all rockbed node temperatures

Qtc = Trombe wall thermocirculation energy flow

$$= \sum_{\text{trombes}} (UAtc * (Tgap - T))$$

where:

Tgap = trombe wall air gap temperature
UAtc = thermocirculation equivalent conductance value (see below)

Qzone = energy flow between zones through explicitly defined interzone loss coefficients or through massless walls between zones

$$= \sum_{\text{losses}} (Lzone * (Tzone - T)) + \sum_{\substack{\text{massless}\\\text{walls}}} ((UWz * AWz) * (Tzone - T))$$

where:

Tzone = other zone air temperature
Lzone = user-specified interzone loss coefficient
AWz = area of wall
UWz = air to air conductance through massless wall

$$= Hf * Hb/(Hf + Hb + Hf*Hb*R)$$

where:

Hf = front side surface coefficient
Hb = back side surface coefficient
R = thermal resistance of wall from surface to surface

Qwindow = energy flow through windows

$$= \sum_{\text{windows}} (UAwin * (Tamb - T))$$

where:

UAwin = air to air conductance through window
Tamb = ambient air temperature

Qamb = energy flow to ambient air through explicitly defined loss coefficients to ambient or through massless walls between the zone and ambient

$$= \sum_{\text{losses}} (Lamb * (Tamb - T)) + \sum_{\substack{\text{massless}\\\text{walls}}} ((UWa * AWa) * (Tamb - T))$$

where:

Tamb = ambient air temperature
Lamb = user-specified loss coefficient to ambient
AWa = area of wall
UWa = air-to-air conductance through massless wall
= same form as UWz above

Qgrd = energy flow to user-specified "ground" node through explicitly defined loss coefficients to ground or through massless walls between zone and ground

$$= \sum_{\text{losses}} (Lgrd * (Tgrd - T)) + \sum_{\substack{\text{massless}\\\text{walls}}} ((UWg * AWg) * (Tamb - T))$$

where:

Tgrd = ground node temperature
Lgrd = user-specified loss coefficient to ground
AWg = area of wall
UWg = air to air conductance through massless wall
= same form as UWz above

Qinf = energy flow due to air infiltration

= UAinf * (Tamb - T)

where:

Tamb = ambient air temperature
UAinf = infiltration equivalent conductance value (see below)

Qsolzon = total solar gain to zone (see above)

Qappli = user-specified appliance gain

Qfan = energy flow between zones by fans

Qheat = heating energy supplied to zone

Qvent = energy removed from zone by venting

Qcool = energy removed from zone by cooling

The new air temperature, T, is calculated each time step by rewriting the energy balance to isolate T in each term, as:

$$T = \frac{\text{NUMhr} + \text{NUMdtf} + \text{NUMdti} + \text{Qequip}}{\text{DENhr} + \text{DENdtf} + \text{DENdti}}$$

where:

$$\text{NUMhr} = \sum_{\text{windows}} (\text{UAwin*Tamb}) + \sum_{\text{losses}} (\text{Lamb*Tamb}) + \sum_{\substack{\text{massless}\\ \text{walls}}} (\text{UWa*AWa*Tamb})$$

$$+ \sum_{\text{losses}} (\text{Lgrd*Tgrd}) + \sum_{\substack{\text{massless}\\ \text{walls}}} (\text{UWg*AWg*Tgrd}) + \text{UAinf*Tamb+Qsolzon+Qappli}$$

$$\text{DENhr} = \sum_{\text{windows}} \text{UAwin} + \sum_{\text{losses}} \text{Lamb} + \sum_{\substack{\text{massless}\\ \text{walls}}} (\text{UWa*AWa}) + \sum_{\text{losses}} \text{Lgrd} + \sum_{\substack{\text{massless}\\ \text{walls}}} (\text{UWg*AWg}) + \text{UAinf}$$

$$\text{NUMdtf} = \sum_{\text{walls}} \text{UAwall*Twall} + \sum_{\text{rocks}} \text{UApass*Trock}$$

$$\text{DENdtf} = \sum_{\text{walls}} \text{UAwall} + \sum_{\text{rocks}} \text{UApass}$$

$$\text{NUMdti} = \sum_{\text{trombes}} (\text{UAtc*Tgap}) + \sum_{\text{losses}} (\text{Lzone*Tzone}) + \sum_{\substack{\text{massless}\\ \text{walls}}} (\text{UWz*AWz*Tzone})$$

$$\text{DENdti} = \sum_{\text{trombes}} \text{UAtc} + \sum_{\text{losses}} \text{Lzone} + \sum_{\substack{\text{massless}\\ \text{walls}}} (\text{UWz*AWz})$$

$$\text{Qequip} = \text{Qfan} + \text{Qrock} + \text{Qheat} - \text{Qvent} - \text{Qcool}$$

Each of the terms in the numerator and the denominator of the zone air temperature equation is calculated by the model as early as possible. Thus, constants are set up by one-time calculations in the Subroutine SETUP and Subroutine RNSET, energy flows constant for an hour are calculated by Subroutine HRSET (NUMhr, DENhr, DENdtf), energy flows which vary each time step but do not involve iterations (NUMdtf) and those which may require iteration (NUMdti, DENdti) are set by Subroutine ZONES.

The numerator and the denominator of the zone air temperature equation are accumulated within Subroutine ZONES with Qequip set to zero. This "pseudotemperature", which would be the zone temperature if no equipment were to operate, is used as the primary control for equipment operation. The final zone temperature is determined by the equipment controller routined discussed in detail elsewhere.

When non-equipment energy flow paths (loss coefficients or massless walls) exist between any two interior zones, then all zone air temperatures are solved iteratively. In each step of the iteration, the values of NUMdti and DENdti are calculated using the previous set of zone air temperatures. Equipment is then operated to determine a new set of zone air temperatures.

Iteration continues until two successive calculations of each zone air temperature differ by at most a user-specified "convergence criteria", or until reaching the user-specified maximum number of iterations per time step. Note that as the zone air temperatures are being iterated, if Trombe wall thermocirculation is present, then a new Trombe wall air gap temperature will be iteratively calculated within each zone air temperature iteration (see below for details).

B. INFILTRATION

Energy flow for air infiltration is calculated separately for each zone in the building as an equivalent thermal conductance, UAinf, which multiplies the difference between zone air temperature and ambient air temperature. The infiltration air conductance value is calculated hourly for each zone by Subroutine INFIL (with one time precalculation of parameters by Subroutine RNSET), as:

$$\text{UAinf} = \text{VOL} * \text{Cair} * \text{Pair} * e^{a * \text{elev}} * \text{AC}$$

where:

VOL = zone air volume (ft^3 or m^3)

Cair = air specific heat = 0.24 BTU/lbF = 1.00418 kJ/kgC

Pair = air density at sea level = 0.075 lb/ft^3 = 1.201385 kg/m^3

elev = station elevation (ft or m)

a = coefficient derived from exponential curve fit

= $-3.71781196 * 10^{-5}$/ft or $-1.219755 * 10^{-4}$/m

AC = infiltration air change rate (air changes/hour)

The air change rate, AC, is taken either as a user specified constant (or scheduled) value, or for the case of hourly varying air change rate as:

$$\text{AC} = (\text{C1} + \text{C2} * \text{WIND} + \text{C3} * |\text{Tz} - \text{Tamb}|) * \text{mult}$$

where

WIND = windspeed (mi/hr or m/s)

Tz = zone air temperature at end of previous hour (F or C)

Tamb = ambient air temperature (F or C)

mult = user-specified multiplier factor

The three coefficients C1, C2 and C3 are taken from Reference [4], and are listed below for both sets of units:

UNITS SYSTEM	C1	C2	C3
English	0.252	0.0218	0.0084
Metric	0.252	0.0488	0.0151

C. TROMBE WALL THERMOCIRCULATION

A Trombe wall may be specified as a building element between any zone and ambient air. The Trombe wall consists of an outer glazing system (optionally with scheduled shutters), an inner wall, and an air space between. Vents may be specified within the wall (an equal area at the top and bottom of the wall) to model the exchange of air between the Trombe wall airspace and the zone to which the Trombe wall is connected ("thermocirculation"). The energy flow due to thermocirculation is calculated as an equivelant thermal conductance value which multiplies the difference between zone air temperature and air gap temperature. The formulation of UAtc is taken from Reference [5], where it is stated to be appropriate only when the vents are the dominant resistance to air flow through the Trombe wall.

Reference [6] indicates that the thermocirculation air movement calculated by this model correlates reasonably well with that produced by more detailed models. However, the magnitude of the thermocirculation is highly dependent on the user-specified "vent discharge coefficient." Reference [6] indicates that the values of this coefficient typically used in the past of approximately 0.8 may be too high by as much as a factor of 2, for the Trombe wall configurations studied.

The thermocirculation equivalent conductance value, UAtc, is taken to be proportional to the square root of the difference in temperature between the air gap node and the zone air node. It is calculated in Subroutine TROMB (with precalculation of constants in Subroutine RNSET) as

$$\mathrm{UAtc} = \mathrm{Ctc} * \mathrm{SQRT}((\mathrm{Tgap} - \mathrm{Tz})/(\mathrm{Tz} + \mathrm{TO}))$$

where:

SQRT = the square root function
Tgap = temperature of air gap node (F or C)
Tz = temperature of zone air node (F or C)
TO = conversion to absolute temperature = 459.69 F or 273.16 C

and the constant coefficient is calculated as:

$$\mathrm{Ctc} = 2 * 3600 * \mathrm{Cair} * \mathrm{Pair} * e^{a*\mathrm{elev}} * \mathrm{Vd} * \mathrm{Av} * \mathrm{Aw} * \mathrm{SQRT}(g * \mathrm{Hv})$$

where:

3600 = 3600 seconds/hour
Cair = air specific heat = 0.24 BTU/lbF = 1.00418 kJ/kgC
Pair = air density at sea level
elev = station elevation (ft or m)
a = coefficient derived from exponential curve fit
= $- 3.71781196 * 10^{-5}$/ft or $-1.219755 * 10^{-4}$/m

Vd = user-specified vent discharge coefficient
Av = ratio of area of one row of vents to area of wall
Aw = area of Trombe wall (ft^2 or m^2)
Hv = height between top and bottom row of vents (ft or m)
g = gravitational constant = 32.0 ft/sec/sec = 9.754 m/sec/sec

Thermocirculation is controlled in two ways. First, the model is formulated to prevent reverse thermosiphoning. That is, UAtc is set to zero whenever $Tgap < Tz$. Second, when the "prevent zone overheat" option is selected by the user, UAtc is set to zero whenever a venting setpoint, Vz, is defined for the zone and $Tz \geq Vz$, or whenever a cooling setpoint, Cz, is defined and $Tz \geq Cz$.

The Trombe wall air gap temperature is calculated in Subroutine TROMB similarly to that for a zone, as:

$$Tgap = \frac{NUMgap}{DENgap} = \frac{UAwin*Tamb + UAwall*Twall + UAtc*Tz + Qsolgap}{UAwin + UAwall + UAtc}$$

where:

UAwin = air to air thermal conductance through glazing
UAwall = thermal conductance from air gap node to first mass node in Trombe wall
Tamb = ambient air temperature
Twall = temperature of first mass node in Trombe wall
Qsolgap = inward flowing fraction of solar energy absorbed by Trombe glazing (see Solar Algorithms above)

When UAtc = 0, Tgap is set once per time step by the above equation. When UAtc > 0, the equation defining Tgap is iterated with Tz held fixed but with UAtc updated by the last calculation of Tgap. Iteration continues until two successive calculations of Tgap differ by at most a user-specified "convergence criteria," or until reaching the user-specified maximum number of iterations.

D. WALL CALCULATIONS

The thermal response of walls is calculated by approximating the wall construction with a thermal network. The network is then solved using the method known as explicit finite differences or Euler's method. This section develops the basic equations for all wall calculations. All building elements with heat capacity are referred to as walls, whether or not they separate rooms in the building. In addition, walls include pure resistance elements which separate zones.

The constant coefficients which define the nodal network layout are calculated within Subroutine COEF2 of the EDITS program. Additional pre-calculation of coefficients occurs in Subroutine SETUP of the LOADS program. Solar inputs to walls are set in Subroutine HRSET. New wall node temperatures are calculated in Subroutine WALLS, with new wall surface temperatures calculated in Subroutine SURFS (SUNCODE Version 1.3).

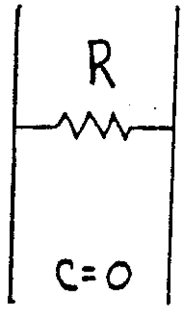
1. Nodal Layout

Each wall is connected to zone air temperatures on either side through user specified surface conductances. These conductances may be zero on one or the other, but not both sides of the wall. They are held constant for the entire run. In addition the wall can receive solar inputs on either or both sides. That part of the wall between surfaces is composed of one or more layers in series.

There are four possible types of layers. They are:

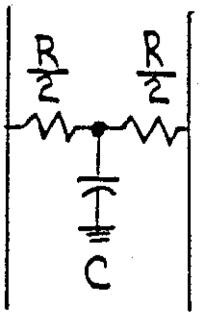
1) a pure thermal resistance,
2) a single node with internal thermal resistance (i.e., finite conductivity),
3) a single node without internal thermal resistance (i.e., a pure thermal capacitance layer or "infinite" conductivity), and
4) a multi-node layer with internal resistance.

These layer types and their thermal network diagrams are shown below. The symbols R and C represent, respectively, the total thermal resistance and the total thermal capacitance of the layer.



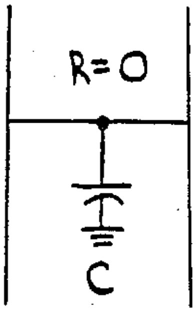
1.

Pure Resistance



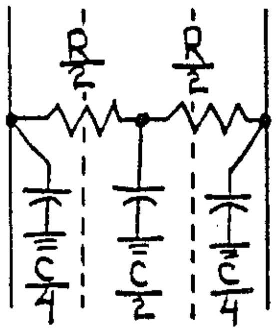
2.

One Node with Resistance



3.

One Node without Resistance



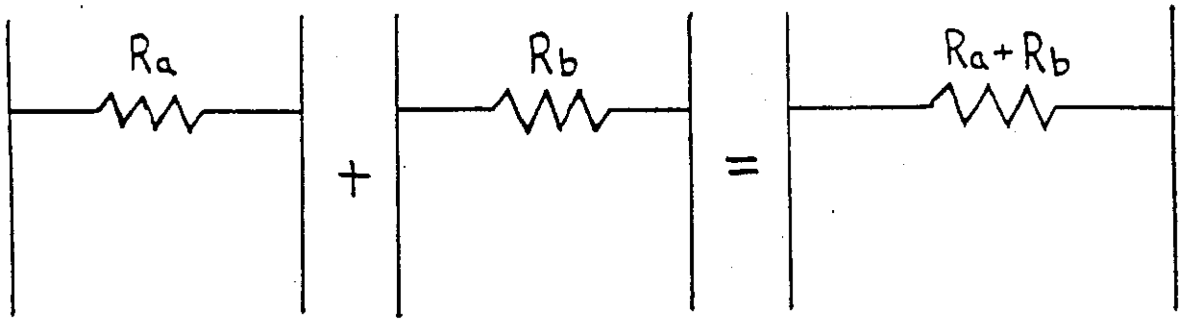
4.

Multi-node

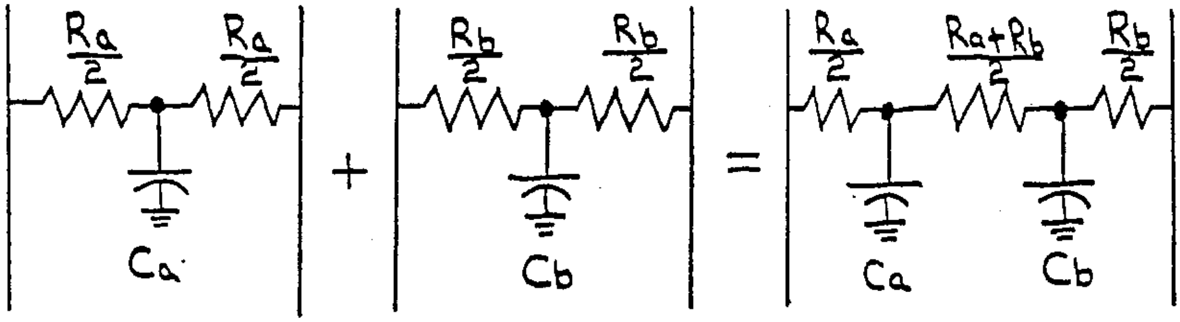
The way in which the various layers are combined to produce the thermal network for each wall are shown below. Each case shows how a layer of one type is combined with a layer of another type. A phase change material layer is required to have some thermal resistance specified on either side. The resistance would be provided by a surface coefficient, a pure thermal resistance layer, or a one node layer with finite conductivity. In particular, the program will not allow two consecutive phase-change layers. The series combination of layers is most easily phrased in terms of thermal resistances. We define the following symbols:

Ra = total thermal resistance of layer a,
Rb = total thermal resistance of layer b,
Ca = total thermal capacitance of layer a, and
Cb = total thermal capacitance of layer b.

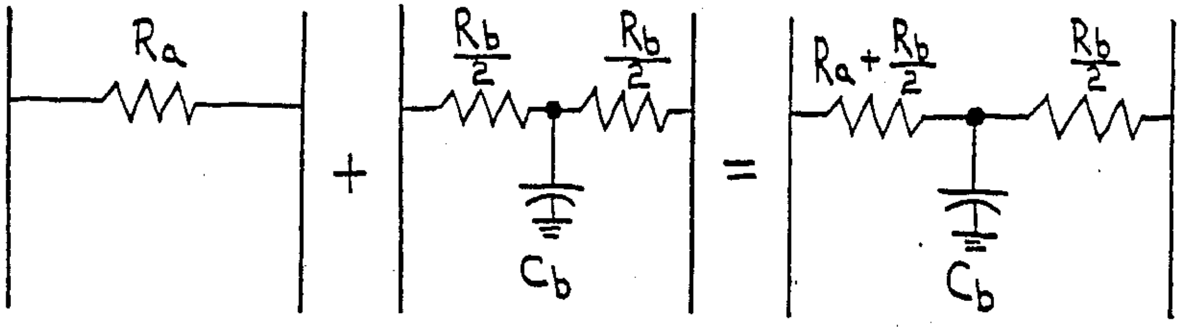
CASE I. Two pure resistance layers.



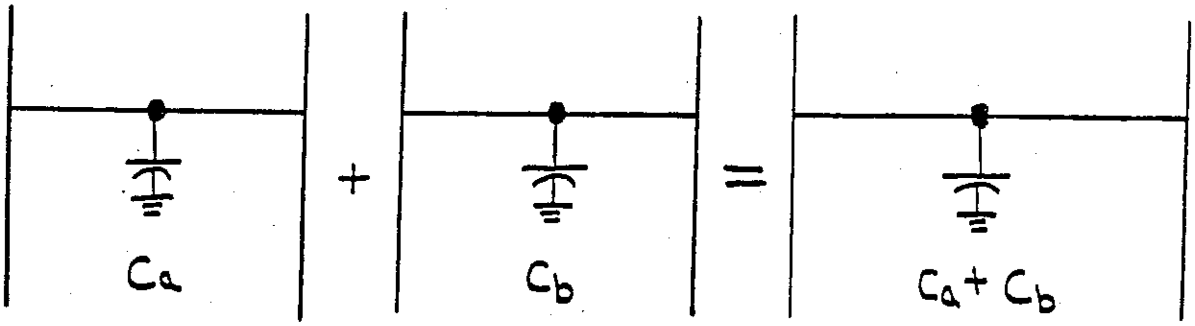
CASE II. Two single node layers.



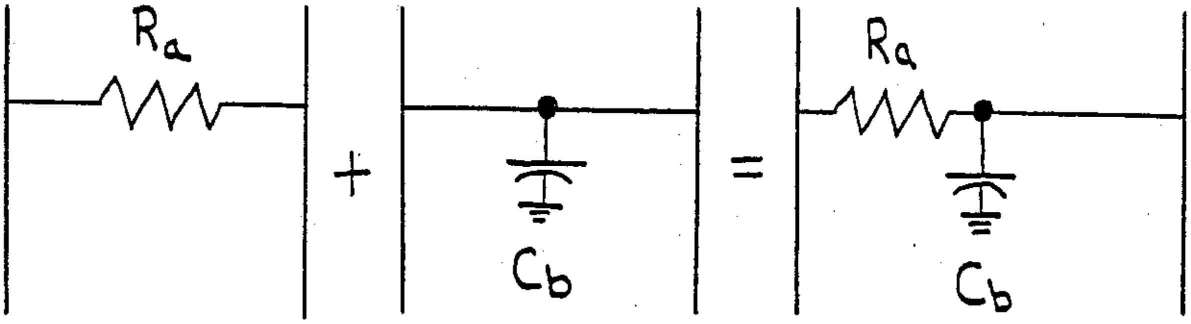
CASE III. Pure resistance layer and single node layer.



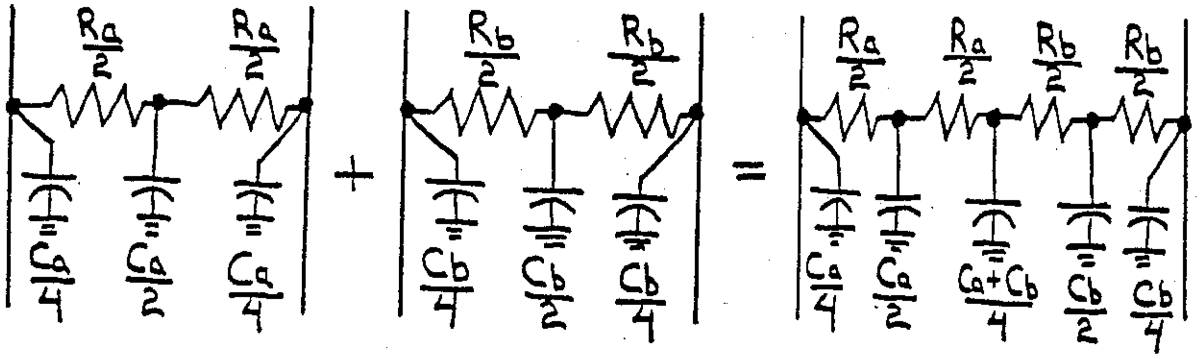
CASE IV. Two pure capacitance layers.



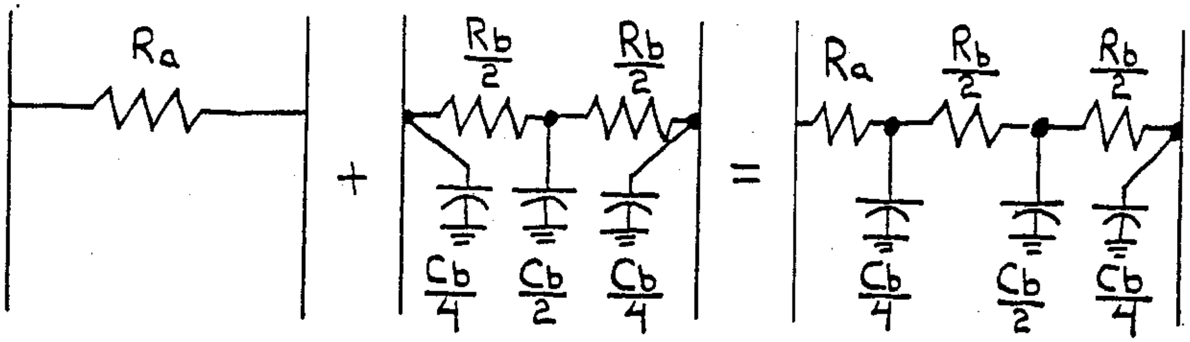
CASE V. Pure resistance layer and pure capacitance layer.



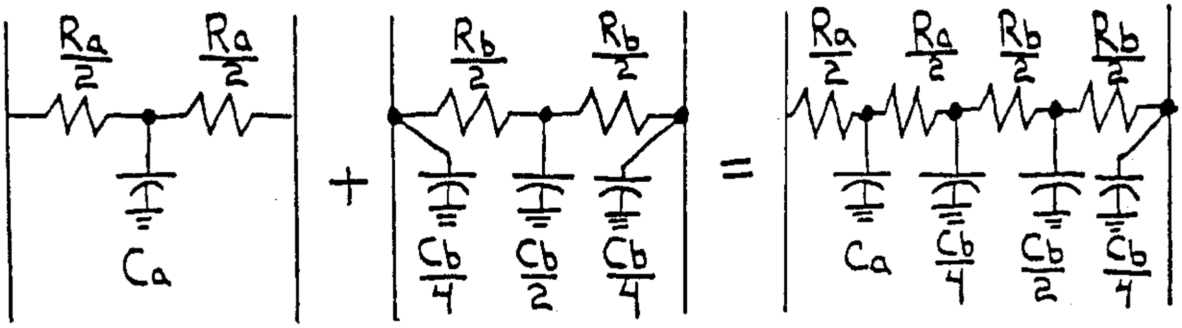
CASE VI. Two multi-node layers.



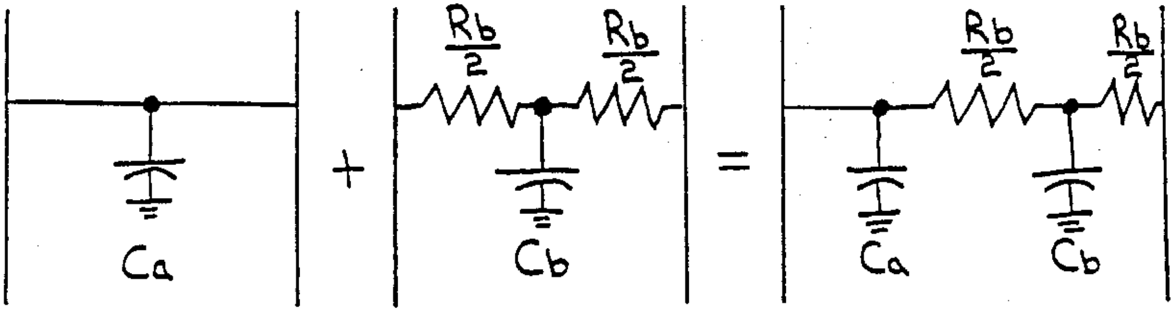
CASE VII. Pure resistance layer and multi-node layer.



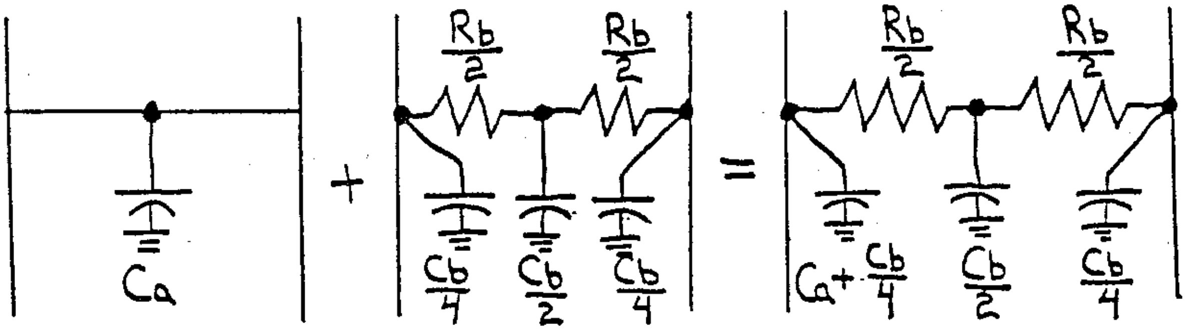
CASE VIII. Single node layer and multi-node layer.



CASE IX. Pure capacitance layer and single node layer.



CASE X. Pure capacitance layer and multi-node layer.



2. Interior Node Temperature Solution

The thermal network model contains only nodes with heat capacity in the interior of the wall. The governing equation for these internal nodes is derived from an instantaneous heat balance on the node. In words:

rate of heat storage = rate of heat gain from node on left + rate of heat gain from node on right.

Mathematically, this becomes:

$$C*(dT/dt) = HL*(TL - T) + HR*(TR - T)$$

where:

T = middle node temperature,
TL = left node temperature,
TR = right node temperature,
HL = thermal conductance to left node,
HR = thermal conductance to right node,
C = thermal capacitance of middle node, and
dT/dt = time derivative of middle node temperature.

The differential equations (one for each node) are solved by using explicit finite differences or Euler integration. The new temperature at the end of a time step is approximated as:

$$T' = T + D*(dT/dt)$$

where:

T' = new temperature of node at end of time step,
T = old node temperature,
D = length of time step.

This results in a set of independent equations for the new node temperatures, each of which has the form:

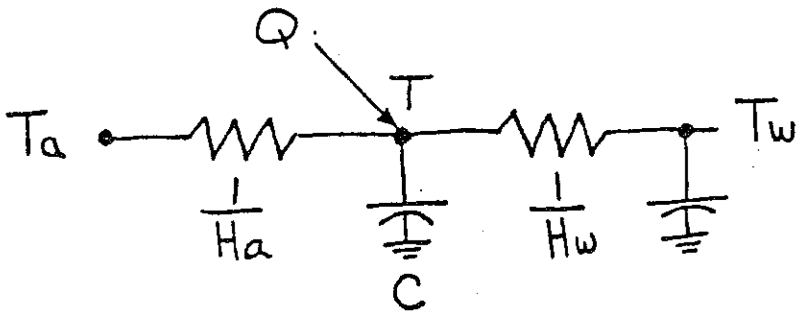
$$T' = (1 - D*HL/C - D*HR/C)*T + (D*HL/C)*TL + (D*HR/C)*TR$$

Both the second law of thermodynamics and mathematical stability of the explicit solution technique require that the first term in parentheses be non-negative. The EDITS program checks this in Subroutine COEF2 and tells the user the minimum number of time steps per hour which may be used. If the time step is unreasonably small, the user may reduce the number of nodes in those layers which required the small step. The main advantages of the explicit technique are simplicity, the fact that each node can be solved independently of the others, and the ability to handle non-linear boundary conditions.

3. Wall Surface Temperature Solution

The equations which define the conditions at the surfaces of the walls are more complex. The derivation of the equations for the first capacity node from the surface, the surface temperature, and the flow of heat and solar radiation at the surface depend on the type of layer which occurs at the surface.

CASE I Multi-node layer at surface.



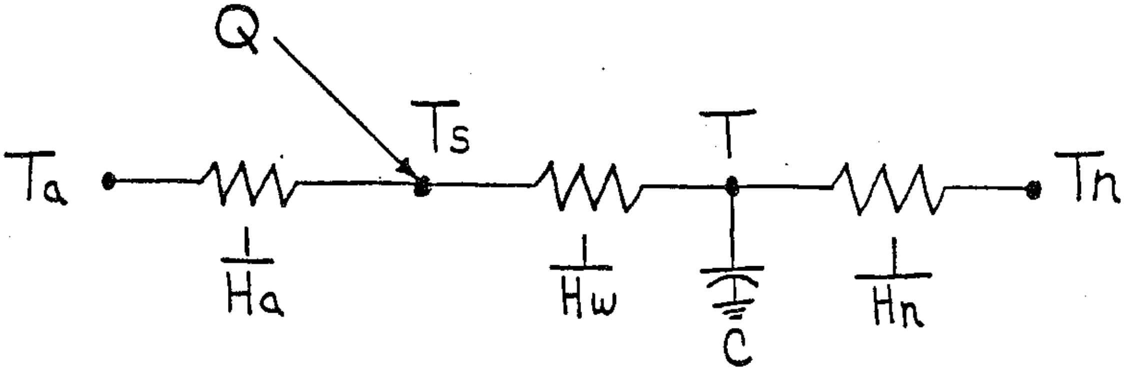
All of the solar is absorbed at the surface node. The node equation is:

T' = (1 - D*Ha/C - D*Hw/C)*T + (D*Ha/C)*Ta + (D*Hw/C)*Tw + (D/C)*Q

where:

T' = new node temperature at surface
Ta = old air temperature of zone at surface
Tw = old temperature of next node in from surface
Q = solar radiation absorbed at surface
Ha = conductance to air temperature node
Hw = conductance to next node in layer
C = thermal capacitance of surface node
D = length of time step

CASE II. Single node layer with internal resistance at surface.



This case is somewhat more complex since there is no node with capacity at the surface where the solar is absorbed. First, define the following:

H = Ha*Hw/(Ha + Hw)

where:

Ha = conductance from surface to air temperature node
Hw = conductance from surface to node in layer
H = conductance from air temperature to node in layer

Then, the solar radiation absorbed at the surface, Q, is broken into two parts, that which goes to the zone air node, Qa, and that which goes to the wall node, Qw, as follows:

Qa = (H/Hw)*Q
Qw = Q - Qa

The solar going to the air temperature node is passed to the ZONES subroutine where it is used in calculating the air temperature. The new node temperature of the layer is:

T' = (1 - D*H/C - D*Hn/C)*T + (D*H/C)*Ta + (D*Hn/C)*Tn + (D/C)*Qw

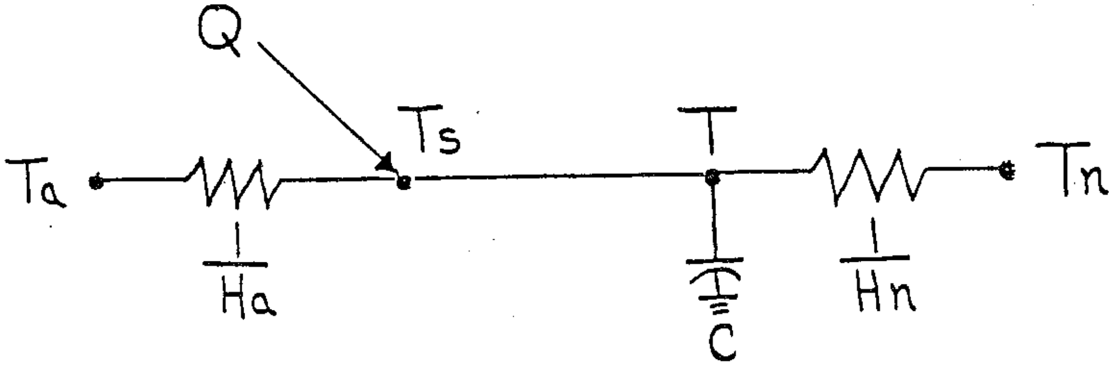
where:

Tn = temperature of next node in wall
Hn = conductance to next node in wall

The surface temperature (which is calculated in Subroutine SURFS after the zone air temperature and the wall node temperatures are updated) is:

Ts = (H/Hw)*Ta + (1 - H/Hw)*T' + (1/(Ha + Hw))*Q

CASE III. Single node layer without internal resistance at surface.



All solar energy is absorbed by the wall node. The equation for the new node temperature is:

$$T' = (1 - D*Ha/C - D*Hn/C)*T + (D*Ha/C)*Ta + (D*Hn/C)*Tn + (D/C)*Q$$

where:

Ha = conductance to air temperature node
Hn = conductance to next node in wall
Ta = air temperature
Tn = temperature of next node
Q = solar absorbed at node

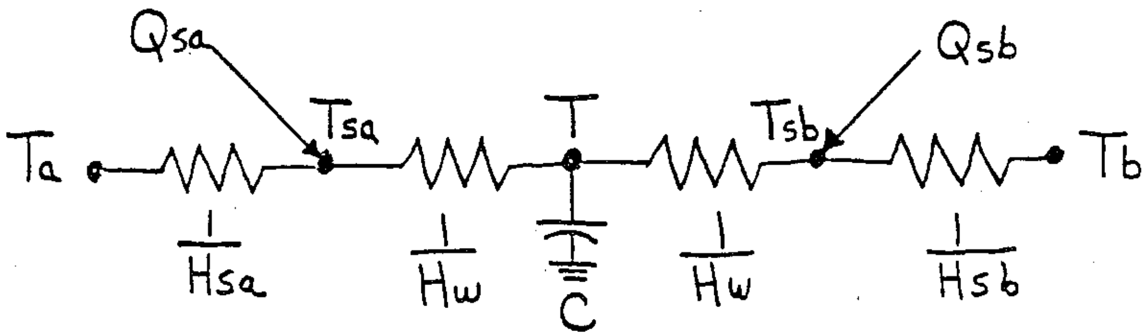
The surface temperature is given by:

$$Ts = T'$$

CASE IV. Pure resistance layer at the surface.

The equations are the same as CASE II setting Hw equal to the thermal conductance from the surface to the first capacity node in the wall. The case where the wall consists of only a pure resistance layer is treated later in this chapter.

CASE V. One node wall with internal resistance.



Walls which contain only a single capacity node are a special case since solar effects on both surfaces must be considered.

Let

$$Ha = Hsa*Hw/(Hsa + Hw)$$

$$Hb = Hsb*Hw/(Hsb + Hw)$$

where

Hsa = conductance from surface a to air node a
Hsb = conductance from surface b to air node b
Hw = conductance from center of layer to surface
Ha = overall conductance from layer node to air node Ta
Hb = overall conductance from layer node to air node Tb

Then

$$Qa = (Ha/Hw)*Qsa$$

$$Qb = (Hb/Hw)*Qsb$$

where

Qsa = solar radiation absorbed at surface a
Qsb = solar radiation absorbed at surface b
Qa = solar radiation flowing to air node Ta
Qb = solar radiation flowing to air node Tb

The equation for the new node temperature is:

$$T' = (1 - D*Ha/C - D*Hb/C)*T + (D*Ha/C)*Ta + (D*Hb/C)*Tb$$
$$+ (D/C)*(Qsa - Qa + Qsb - Qb)$$

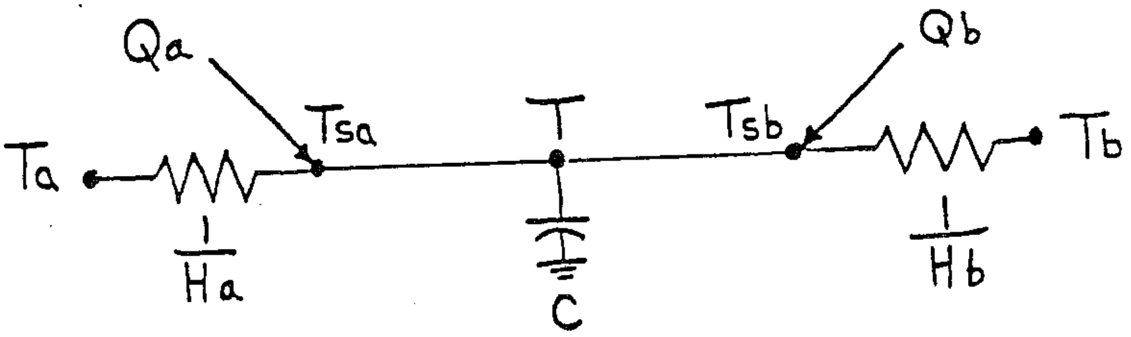
The surface temperature equations are:

$$Tsa = (Hsa/(Hsa + Hw))*Ta + (Hw/(Hsa + Hw))*T' + (1/(Hsa + Hw))$$

$$Tsb = (Hsb/(Hsb + Hw))*Tb + (Hw/(Hsb + Hw))*T' + (1/(Hsb + Hw)).$$

The order of calculation and subroutines used are the same as in CASE II.

CASE VI. One node wall without internal resistance.



This type of one node wall is used primarily to model water walls and other situations where the internal resistance is negligible. It is simpler than the previous case.

Solar radiation from both sides is absorbed entirely by the capacity node.

The new node temperature equation is:

T' = (1 - D*Ha/C - D*Hb/C)*T + (D*Ha/C)*Ta + (D*Hb/C)*Tb + (D/C)*(Qa + Qb)

where

Ha = conductance from surface to air temperature Ta
Hb = conductance from surface to air temperature Tb
Qa = solar radiation absorbed at surface a
Qb = solar radiation absorbed at surface b

The surface temperature equations are:

Tsa = T'
Tsb = T'

The order of calculation and the subroutines used are the same as CASE II.

4. Phase-change Material Layers.

Phase change layers involve special calculations. They have the same nodal equations as the pure capacitance layers, but differ in their ability to store heat without change of temperature. That is, the phase change material layer will behave as a pure capacitance layer until its temperature reaches the user-specified melting point. Then the temperature of the layer is held constant, while latent energy is stored up to the user-specified total latent storage capability of the layer. When the total latent storage capability is reached, then the layer again behaves as a pure capacitance layer. We define the following terms:

Tmelt = the melting point temperature of the phase-change material,
Qmelt = the latent heat of fusion of the material, and
Qstored = the latent heat stored in the layer.

Then we calculate the new temperature of the phase-change material as though it were an ordinary material (in Subroutine WALLS):

$$T' = (1 - D*HL/C - D*HR/C)*T + (D*HL/C)*TL + (D*HR/C)*TR + (D/C)*Q$$

where the symbols have the same meaning as in the previous equations. Let Qin be the total heat flow into the phase change layer during the time step, so that:

$$Qin = C*(T' - T)$$

Several cases for the calculation of the true new temperature, Tnew, and the Qstored term arise. The appropriate case and resulting values for Tnew and Qstored are selected in Subroutine PCMAT.

CASE I. $T < Tmelt$ and $T' < Tmelt$
or $T > Tmelt$ and $T' > Tmelt$
or $T' = Tmelt$

In each of these cases the material will not change phase, and we have:

$$Tnew = T'$$
and
$$Qstored = 0$$

CASE II. $T < Tmelt$ and $T' > Tmelt$

That is, the material is in the solid phase and is melting.
Let:

$$Qlatent = C*(T' - Tmelt).$$

If $Qlatent \le Qmelt$

then $Qstored = Qlatent$
and $Tnew = Tmelt$

otherwise

$Qstored = Qmelt$
and $Tnew = Tmelt + (Qlatent - Qmelt)/C.$

CASE III. $T > Tmelt$ and $T' < Tmelt$

That is, the material is in the liquid phase and is freezing.

Let:

$$Qlatent = C*(Tmelt - T').$$

If $Qlatent \leq Qmelt$

then $Qstored = Qmelt - Qlatent$
and $Tnew = Tmelt$

otherwise

$Qstored = 0$
and $Tnew = Tmelt + (Qmelt - Qlatent)/C.$

CASE IV. $T = Tmelt$ and $T' > Tmelt$

That is, the material is partially melted and is continuing to melt.

Let:

$$Qlatent = Qstored + C*(T' - T).$$

If $Qlatent \leq Qmelt$

then $Qstored' = Qlatent$
and $Tnew = Tmelt$

otherwise

$Qstored' = 0$
and $Tnew = Tmelt + (Qlatent - Qmelt)/C.$

CASE V. $T = Tmelt$ and $T' < Tmelt$

That is, the material is partially melted and is freezing.

Let:

$$Qlatent = Qstored + C*(T' - T).$$

If $Qlatent \geq 0$

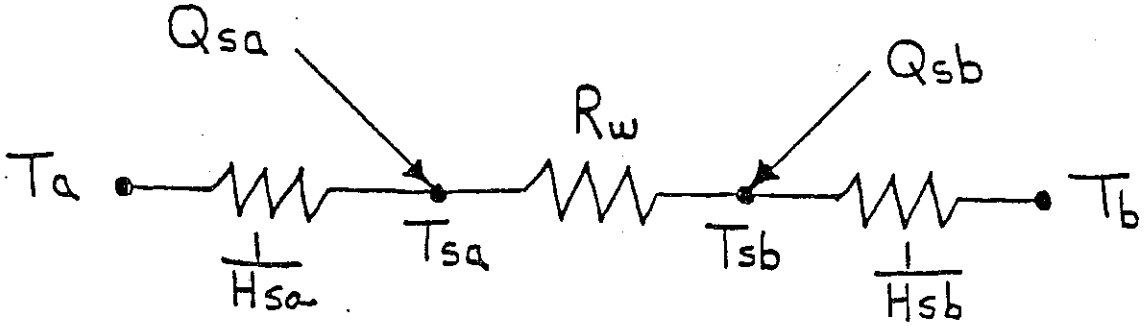
then $Qstored' = Qlatent$
and $Tnew = Tmelt$

otherwise

$Qstored' = 0$
and $Tnew = Tmelt + Qlatent/C.$

5. Pure Resistance Walls

These are another special case. The network is shown below.



where:

Ta = air temperature on side a
Tb = air temperature on side b
Tsa = surface temperature on side a
Tsb = surface temperature on side b
Hsa = surface conductance on side a
Hsb = surface conductance on side b
Rw = resistance of wall from surface to surface
Qsa = solar absorbed at surface a
Qsb = solar absorbed at surface b

Define the following terms:

Ha = Hsa + 1/Rw
Hb = Hsb + 1/Rw
J = Hsa + Hsb + Hsa*Hsb*Rw
K = 1/(Hsa*Hsb + ((Hsa + Hsb)/Rw))

The equations for solar heat flow to the air temperature nodes are:

Qa = (Hsa*Rw*Hb/J)*Qsa + (Hsa/J)*Qsb

Qb = (Hsb/J)*Qsa + (Hsb*Rw*Ha/J)*Qsb

The equations for the surface temperatures are:

Tsa = (Hsa*Hb*K)*Ta + (Hsb*K/Rw)*Tb + (Hb*K)*Qsa + (K/Rw)*Qsb

Tsb = (Hsa*K/Rw)*Ta + (Hsb*Ha*K)*Tb + (K/Rw)*Qsa + (Ha*K)*Qsb

E. ROCKBIN CALCULATIONS

The rockbin model used in this program is an adaptation of the infinite NTU model developed at the University of Wisconsin by Hughes, et al. [7,8]. It is therefore similar to the rockbin model in TRNSYS, Version 10. The rockbin is divided into five equal segments. It is assumed that the rock and air temperatures are identical in each segment (infinite NTU), and that there is no cross-sectional temperature gradient. The model allows for axial conductance, and passive losses to an internal zone, and the special zones AMBIENT and GROUND. Air flow in the rockbin is specified by the user as either uni-directional or bi-directional. If bi-directional flow is specified, the flow direction during the discharge cycle is opposite to that of the charge cycle. The fans which operate during the charge and discharge cycles may have different maximum flow rates and different minimum temperature differentials. The node used to determine the control logic for the rockbin is the outlet node. In any one mode of operation the air flows in a closed loop between the rockbin and the zone to which it is connected.

From the user inputs the following parameters are determined:

Vc = maximum volumetric flow rate for charge fan
Vd = maximum volumetric flow rate for discharge fan
P = density of air at altitude of location
C = heat capacity of rocks in one segment
Hz = passive conductance of one segment to internal zone
Ha = passive conductance of one segment to AMBIENT zone
Hg = passive conductance of one segment to GROUND zone
K = axial conductance of rockbin from one node to the next

Passive losses from the rockbin are assumed to be evenly spread over each segment of the rockbin.

Prior to actual operation of the rockbin, the fan/rockbin controller algorithm has already determined the direction of flow (if any) through the rockbin (in Subroutines EQMTA or ROKON), the zone to which the rockbin is connected, and the actual volumetric flow rate as a fraction of the maximum flow rate.

A heat balance on each node gives, in words:

$$\text{rate of storage} = \text{rate of gain from convection} + \text{rate of gain from axial conductance} + \text{rate of gain from passive conductance}$$

The capacitance flow rate, M, is calculated as

$$M = Ca * P * V$$

where

Ca = specific heat of air at constant pressure
V = volumetric air flow rate through rockbin
P = density of air at altitude of location

Assuming air flow from node 1 toward node 5, the equation for T(1), the inlet node temperature, is:

$$C*(dT(1)/dt) = M*(Tzone - T(1)) + K*(T(2) - Tin) + Hz*(Tpz - T(1)) + Ha*(Tamb - T(1)) + Hg*(Tgrd - T(1))$$

For each interior node temperature, Ti:

$$C*(dT(i)/dt) = M*(T(i-1) - T(i)) + K*(T(i-1) - T(i)) + K*(T(i+1) - T(i)) + Hz*(Tpz - T(i)) + Ha*(Tamb - T(i)) + Hg*(Tgrd - T(i))$$

For the outlet node temperature, T(5):

$$C*(dT(5)/dt) = M*(T(4) - T(5)) + K*(T(4) - T(5)) + Hz*(Tpz - T(5)) + Ha*(Tamb - T(5)) + Hg*(Tgrd - T(5))$$

where:

T(i) = temperature at node i
dT(i)/dt = time derivative of temperature at node i
Tzone = temperature of zone connected by air flow to rockbin
Tpz = temperature of internal zone for passive losses
Tamb = temperature of AMBIENT node
Tgrd = temperature of GROUND node

For air flow in the other direction, the node indices are reversed. These differential equations are solved by Euler integration (explicit finite differences) as:

$$T' = T + D*(dT/dt)$$

where:

T' = new temperature at node
T = old temperature at node
dT/dt = time derivative at node
D = length of time step

This results in the following equations for the new temperatures at each of the nodes, which are implemented in Subroutine ROCKS.

1) Inlet node

$$T'(1) = (1 - (D/C)*(M - K - Hz - Ha - Hg))*T(1) + (D*M/C)*Tzone + (D*K/C)*T(2) + (D*Hz/C)*Tpz + (D*Ha/C)*Tamb + (D*Hg/C)*Tgrd$$

2) Interior Nodes

$$\begin{aligned} T'(i) = {} & (1 - (D/C)*(M + 2*K + Hz + Ha + Hz))*T(i) + (D*M/C)*T(i-1) \\ & + (D*K/C)*T(i-1) + (D*K/C)*T(i+1) + (D*K/C)*T(i+1) \\ & + (D*Hz/C)*Tpz + (D*Ha/C)*Tamb + (D*Hg/C)*Tgrd \end{aligned}$$

3) Outlet Node

$$\begin{aligned} T'(5) = {} & (1 - (D/C)*(M - K - Hz - Ha - Hg))*T(5) + (D*M/C)*T(4) \\ & + (D*K/C)*T(4) + (D*Hz/C)*Tpz + (D*Ha/C)*Tamb + (D*Hg/C)*Tgrd \end{aligned}$$

As in the wall equations, the term which multiplies the old node temperature on the right side of each equation must be non-negative. The EDITS program checks for this in Subroutine COEF2. Typical rockbins will be stable with one hour time steps.

Once the node temperatures are updated, the heat stored in the rockbin is calculated as:

$$Qstored = C * \sum_{i=1}^{5} T(i)$$

Section 6-4. EQUIPMENT ALGORITHMS

A. HVAC CONTROLLER

This algorithm is implemented in Subroutine HVACE. The algorithm does not model the operation of HVAC equipment; rather it calculates only the heating, venting and cooling loads that each zone experiences. Heating and cooling are taken as a direct energy gain or loss, respectively. Venting is modeled as a controlled air exchange with ambient air. The algorithm uses separate heating, venting and cooling setpoints specified by the user, and optional maximum rates of HVAC equipment operation. At any time step, each zone may be in any one of three states: no HVAC equipment operation, heating only, or some combination of venting and cooling.

The algorithm solves the basic equation:

$$T = (N + QH - QV - QC)/D$$

for the unknown quantities:

T = resultant zone air temperature
QH = heating energy delivered to zone
QV = venting energy removed from zone
QC = cooling energy removed from zone

by using:

N = total energy flow to zone excluding HVAC equipment (i.e., the numerator of the zone air temperature defining equation)
D = total heat flow conductance given for the zone (i.e., the denominator of the zone air temperature defining equation)
Tamb = ambient air temperature
W = zone air temperature if no HVAC equipment operates
 = N / D
H = heating setpoint, taken as -oo if none defined
Hcap = heater maximum capacity, taken as +oo if "adequate"
V = venting setpoint, taken as +oo if none defined
Vcap = venter maximum capacity, taken as +oo if "adequate"
 = Vmax * Pair * Cair * VOL
  where Vmax = user input maximum venting air change rate
   Pair = air density;
   Cair = air heat capacity;
   VOL = zone air volume;
C = cooling setpoint, taken as +oo if none defined
Ccap = cooler maximum capacity, taken as +oo if "adequate"

The algorithm assumes that all specified setpoints obey the inequality:

$$H \leq V \leq C$$

If W < H, then the zone is in the heating mode. In this case, the solution is given as:

```
QH = min [ H*D-N , Hcap ]
T  = (N+QH)/D
QV = 0
QC = 0
```

When the zone is not in the heating mode, the algorithm checks for the operation of venting and/or cooling. This solution is more complex, since it depends on the relationship of the four temperatures W, Tamb, V and C. The full solution for each possible case is presented below. (In all cases, QH = 0.)

1) No venting or cooling

```
If  W < V < +oo  or  W < C < V = +oo  or  V < W < Tamb < C  or  V < W < C < Tamb

   then  T  = N / D
         QV = 0
         QC = 0
```

2) Venting only

```
If  Tamb < V < W < C

   then if  N - D * V < Vcap * (V - Tamb)

         then  T  = V
               QV = N - D * V
               QC = 0

         else  T  = (N - Vcap * Tamb)/(D + Vcap)
               QV = Vcap * (T - Tamb)
               QC = 0
```

3) Venting only

```
If  V < Tamb < W < C

   then if  Vcap = +oo

         then  T  = Tamb
               QV = N - D * Tamb
               QC = 0

         else  T  = (N + Vcap * Tamb)/(D + Vcap)
               QV = Vcap * (T - Tamb)
               QC = 0
```

4) Cooling only

```
If  $V < C < W < Tamb$  or  $C < W < V = +oo$

   then  $QV = 0$
         $QC = min [ N-D*C , Ccap ]$
         $T  = (N - QC) / D$
```

5) Both venting and cooling

```
If  $Tamb < V < C < W$

   then if  $N - D * V < Vcap * (V - Tamb)$

            then  $T  = V$
                  $QV = N - D * V$
                  $QC = 0$

            else if  $(N + Vcap * Tamb)/(D + Vcap) < C$

                     then  $T  = (N + Vcap * Tamb)/(D + Vcap)$
                           $QV = Vcap * (T - Tamb)$
                           $QC = 0$

                     else  $QC = min [ N+Vcap*Tamb-C*(D+Vcap) , Ccap ]$
                           $T  = (N + Vcap * Tamb - QC)/(D + Vcap)$
                           $QV = Vcap * (T - Tamb)$
```

6) Both venting and cooling.

```
If  $V < Tamb < C < W$

   then if  $Vcap = +oo$

            then  $T  = Tamb$
                  $QV = N - D * Tamb$
                  $QC = 0$

            else if  $(N + Vcap * Tamb)/(D + Vcap) < C$

                     then  $T  = (N + Vcap * Tamb)/(D + Vcap)$
                           $QV = Vcap * (T - Tamb)$
                           $QC = 0$

                     else  $QC = min [ N+Vcap*Tamb-C*(D+Vcap) , Ccap ]$
                           $T  = (N + Vcap * Tamb - QC)/(D + Vcap)$
                           $QV = Vcap * (T - Tamb)$
```

7) Both venting and cooling

```
If  V < C < Tamb < W

   then if  N - D * C < Ccap

           then  T  = C
                 QV = 0
                 QC = N - D * C

           else if  (N - Ccap)/D < Tamb

                   then  T  = ( N - Ccap ) / D
                         QV = 0
                         QC = Ccap

                   else if  Vcap = +oo

                           then  T  = Tamb
                                 QV = A - D * Tamb
                                 QC = 0

                           else  T  = (N + Vcap * Tamb - Ccap)/(D + Vcap)
                                 QV = Vcap * (T - Tamb)
                                 QC = Ccap
```

B. FAN/ROCKBIN CHARGE CONTROLLER

The fan controller algorithm calculates the energy moved from a source zone to a sink zone by any fan, and the energy delivered from a source zone to any rockbin when the rockbin is in its charging mode. Rockbin discharge control is handled separately, and is discussed in a following section. The algorithm also produces source and sink zone air temperatures (consistent with the derived fan operation) and HVAC energy flows, by using the HVAC controller algorithm discussed above.

Fans (including rockbin charge fans) are modeled as thermostatically controlled conductances between the source zone air temperature node and the sink zone air temperature node (or rockbin charge outlet node). The fan controller is assumed to be able to cycle the fan on and off at an arbitrarily high rate, or equivalently, to select any fan speed up to the fan's specified capacity. The controller is also assumed to be interconnected with the HVAC thermostat to provide consistent equipment operation (e.g. to avoid one device trying to heat a zone while another device is trying to cool it).

Four constraints are assumed to limit fan operation. Two of these constraints involve interaction with the HVAC thermostat setpoints. First, the "maximum energy available constraint" prevents any fan from operating whenever the fan source zone is in the heating mode. Second, the energy delivered by any fan is limited to avoid overheating the fan sink zone. This is referred to as the "maximum energy needed constraint" (undefined for rockbin charge fans). It prevents operation of the fan whenever the sink zone temperature is above the lowest of its defined HVAC setpoints. The third, the "minimum temperature difference constraint," prevents fan operation whenever the difference between source zone and sink temperatures is less than a user-specified minimum. Finally, the "maximum energy delivery constraint" limits fan operation to its user-specified maximum capacity. Subject to these four constraints, the fan controller algorithm maximizes the energy removed from the source zone by fans.

Specifically, the fan controller algorithm solves the system of equations:

$$\mathrm{Tsrc} = \frac{\mathrm{Nsrc} + \mathrm{QEsrc} - \sum_{i=1}^{N} \mathrm{QDi}}{\mathrm{Dsrc}}$$

$$\mathrm{Ti} = \frac{\mathrm{Ni} + \mathrm{QEi} + \mathrm{QDi}}{\mathrm{Di}} \qquad \text{for } i = 1,2,\ldots,\mathrm{NF}$$

$$\mathrm{QDi} = \mathrm{Ki} * \mathrm{UAFi} * (\mathrm{Tsrc} - \mathrm{Ti}) \qquad \text{for } i = 1,2,\ldots,\mathrm{N}$$

for the unknown quantities Tsrc, QEsrc, Ti, QDi, QEi and Ki by maximizing $\sum_{i=1}^{N} QDi$ subject to the condition that $QDi > 0$ only if all four of the following constraints are satisfied:

1) the maximum energy available constraint

   $$Tsrc \geq Hsrc$$

2) the maximum energy needed constraint (undefined for rockbins)

   $$Ti \leq Si \quad \text{where} \quad Si = \min [\ Hi\ ,\ Vi\ ,\ Ci\ ]$$

3) the minimum temperature difference constraint

   $$Tsrc \geq Ti + DTFi$$

4) the maximum energy delivery constraint

   $$0 \leq Ki \leq 1$$

where:

N = total number of fans, NF, plus number of rockbins, NR, connected to source
Tsrc = source zone air temperature
Ti = fan sink temperature = sink zone air temperature, for i = 1,...,NF
 or = rockbin charge outlet node temperature, i = 1,...,NR
QEsrc = source zone HVAC energy
QEi = sink zone HVAC energy
QDi = energy moved from source to sink by fan i
Ki = duty cycle for fan i
UAFi= user-specified maximum capacity for fan i
DTFi= user-specified minimum temperature difference for fan i
Hsrc= source zone heating setpoint
Hi = sink zone heating setpoint
Vi = sink zone venting setpoint
Ci = sink zone cooling setpoint
Ni = zone total energy flow, excluding HVAC and fans
 (i.e. the numerator of the zone air temperature defining equation)
Di = zone total conductance, excluding HVAC and fans
 (i.e. the denominator of the zone air temperature defining equation)

The following sections discuss the solution technique used by the fan controller algorithm and elaborate on details of its operation.

1. Multi-Fan Controller Algorithm

This section presents a general overview of the logic of the fan controller algorithm, with additional details contained in following sections. The algorithm separately considers each "fan network," comprised of a single source zone, all sink zones connected to the source by a fan, and the charge cycle of all rockbins connected to the source zone. It produces final air temperatures for each of the zones, HVAC equipment energy flows for each zone, and energy flows for each fan and the charge cycle of each rockbin. The logic is controlled by Subroutine EQMTA, and makes use of Subroutines EQMTB, EQMTC, EQMTD, EQMTE, and HVACE. Calculations discussed below are performed by Subroutine EQMTA, unless otherwise noted.

First we define the following quantities:

N = total number of fans and rockbins connected to the source zone

TMAX = source zone air temperature if only HVAC equipment (no fans) operates

= (Nsrc + QEsrc)/Dsrc

TMINi = minimum source zone air temperature required for fan i to run, i=1,...,N

= sink zone air temperature if only HVAC equipment (no fan) operates plus the user-specified minimum temperature difference for the fan

= DTF + (Ni + QEi)/Di

or = rockbin charge outlet node temperature plus the user-specified minimum temperature difference

= Tr + DTF

Assume that each TMINi is unique (the general case is discussed in a following section), and that they are ordered such that:

TMINi < TMINj for i=1,...,N and j=i+1

(The calculation and sorting of TMIN are performed in Subroutine EQMTB.)

QAVL(T) = energy that must be removed from the source zone to produce an air temperature of T

= Nsrc - Dsrc * T

Note that QAVL is a strictly decreasing function of T.

QDi(T) = energy which would be delivered by fan i, if it were the only fan and if the source zone were at the temperature T. Note that as discussed below, QD is a non-decreasing, piecewise linear function of T.

The controller algorithm operates by using the TMIN's as successively higher guesses for the final source zone air temperature Ti. That is, Ti = TMINi for i=1,...,N. Whenever TMINi > TMAX or i > N, then Ti is set to TMAX as the final temperature guess. Each increasing source zone temperature guess involves an increasing number of fans which will operate during the time step. For each guessed temperature, the energy which would be delivered by the fans, if the source zone were at the given temperature, is compared to the energy which must be removed from the source zone to produce that temperature. The algorithm proceeds until one of the four conditions given below is reached. (Note that at least one of the conditions is met whenever Ti = TMAX.)

CASE 1. $$\mathrm{QAVL}(Ti) = \sum_{j=1}^{i-1} \mathrm{QD}j(Ti)$$

That is, the energy removed by fans 1,...,i-1 at the source zone temperature Ti equals the energy needed to be removed to produce temperature Ti. In this case, the ith fan is assumed not to operate.

CASE 2. $$\mathrm{QAVL}(Ti) = \sum_{j=1}^{i} \mathrm{QD}j(Ti)$$

That is, the energy removed by fans 1,...,i at source zone temperature Ti equals the energy needed to produce temperature Ti.

CASE 3. $$\mathrm{QAVL}(Ti) < \sum_{j=1}^{i-1} \mathrm{QD}j(Ti)$$

That is, the energy removed by fans 1,...,i-1 at source zone temperature Ti is greater than the energy available in the source at temperature Ti.

CASE 4. $$\sum_{j=1}^{i-1} \mathrm{QD}j(Ti) < \mathrm{QAVL}(Ti) < \sum_{j=1}^{i} \mathrm{QD}j(Ti)$$

That is, there is sufficient energy available for fans 1,...,i-1 to operate, but that an additional constraint must be placed on fan i to obtain an energy balance.

In the first two cases, the source zone temperature is taken as Ti, and the energy moved by each fan j=1,...,i is taken as QDj(Ti). Once these values are established, the HVAC controller is used to calculate final air temperatures for each zone.

For Case 3 and Case 4, additional calculations are required to establish the source zone temperature and fan energy, prior to calculating the sink zone air temperatures by the HVAC controller. In Case 4, the source zone temperature is taken as Ti and each fan j=1,...,i-1 moves the energy given by QDj(Ti). The energy moved by fan i is calculated in Subroutine EQMTD as:

$$\mathrm{QDi(Ti)} = \mathrm{QAVL(Ti)} - \sum_{j=1}^{i-1} \mathrm{QDj(Ti)}$$

For Case 3, we know that fans 1,...,i-1 will operate, and that the source temperature is bounded above by Ti and below by the previous temperature guess. We then solve the source zone air temperature defining equation, in Subroutine EQMTC:

$$\mathrm{Tsrc} = \frac{\mathrm{Nsrc} - \sum_{j=1}^{i-1} \mathrm{QDj(Tsrc)}}{\mathrm{Dsrc}}$$

by using the linear nature of the fan energy delivery function QD (see below), giving:

$$\mathrm{Tsrc} = \frac{\mathrm{Nsrc} - \sum_{j=1}^{i-1} (\mathrm{Bj} + \mathrm{Aj} * \mathrm{Tsrc})}{\mathrm{Dsrc}}$$

Source zone venting and cooling energies do not appear explicitly in these equations since, as discussed below, their affects are accounted for in the summation of fan energies. Note that in solving the above equation for Tsrc, since the fan energy delivery function QD is piecewise linear, Subroutine EQMTC must first find a temperature interval which contains the value Tsrc and over which the coefficients Bj and Aj do not change. Once this interval is identified, and the appropriate Bj and Aj coefficients are selected, we have:

$$\mathrm{Tsrc} = \frac{\mathrm{Nsrc} - \sum_{j=1}^{i-1} \mathrm{Bj}}{\mathrm{Dsrc} + \sum_{j=1}^{i-1} \mathrm{Aj}}$$

Having established the source zone temperature, Tsrc, the fan delivery energies are taken as QDj(Tsrc) for fans j=1,...,i-1. Finally, as in the other cases, the HVAC controller is used to calculate zone HVAC energies and sink zone temperatures.

2. Interaction with Source Zone HVAC Equipment

The maximum energy available constraint for the fan controller algorithm prevents any fan from operating when the source zone is in the heating mode. However, source zone venting and/or cooling may operate in conjunction with one or more fans. If either venting or cooling in the source zone are found to be of adequate capacity to maintain the relevant setpoint, then fans which have a higher minimum "on-temperature," TMIN, will not operate.

The controller algorithm, in Subroutine EQMTB, adds "dummy" fans to the list of fans fed by the source zone to account for either venting or cooling equipment. Specifically, whenever venting equipment is defined with finite capacity, and TMAX is greater then the source zone venting setpoint, V, then a dummy fan is created having TMIN set to the maximum of V and the ambient air temperature, Tamb. Similarly, whenever cooling equipment is defined with finite capacity, and TMAX is greater then the source zone cooling setpoint, C, then a dummy fan is created having TMIN set to C.

3. Non-Unique Minimum Fan On Temperatures

In any time step it may happen that one or more sets of fans are found to have the same minimum source zone temperature for fan operation, TMIN. When this occurs, the algorithm evaluates the performance of fans at that temperature level only once by lumping all fans having the same TMIN. The primary complexity arises in the solution by the fan controller to Case 4 presented above. That is, when:

$$\sum_{j=1}^{i-1} \mathrm{QDj(Ti)} < \mathrm{QAVL(Ti)} < \sum_{j=1}^{i-1} \mathrm{QDj(Ti)} + \sum_{j=i}^{k} \mathrm{QDj(Ti)}$$

where: TMINi = TMINj for j=i,...,k.

This event is also handled by Subroutine EQMTD in the following manner. The source zone air temperature is taken as Ti, and the operation of fans i,...,k is limited to produce this temperature. The fans i,...,k are divided into two groups. The first group consists of all fans whose sink temperature is fixed. This includes all fan sink zones with adequate heating capacity, all rockbins, and the dummy source zone venting and cooling fans. The second group contains all other fans, those with inadequate heaters or no heater defined.

The fans in the second group are assumed not to operate, since any energy delivered to one of these sink zones will raise its air temperature, which in turn will raise its minimum fan on temperature above Ti. For the fans in the first group, an average duty cycle is calculated as:

$$f = Q / \sum (UAF*DTF)$$

where: $$Q = QAVL(Ti) - \sum_{j=1}^{i-1} QDj(Ti)$$

UAF = fan capacity
DTF = fan minimum temperature difference
and the summation is over all fans in the first group.

Then the energy delivered by each fan in the first group at this average duty cycle, f*UAF*DTF, is compared to the maximum energy needed, Qmax, for each fan sink zone. If for any fan j, Qmax < f*UAF*DTF, then QDj is set to Qmax, and the Q in the duty cycle equation above is decremented by Qmax. A new average duty cycle is calculated for all remaining fans in the first group. This process is repeated until $Qmax \geq f*UAF*DTF$ for all remaining fans. Then the energy delivered by each of these fans is taken as f*UAF*DTF.

3. The Fan Energy Delivery Function

In the full multi-fan problem, the energy delivered by any fan is a complex function of conditions existing in the source zone and conditions in all of the sink zones. A direct algebraic definition of the energy delivered by any fan for the general case is most difficult. However, if we assume that we know the source zone temperature which will result from the operation of all fans, then the energy delivered by any fan can be expressed as a non-decreasing, piecewise linear function of source zone temperature. Specifically, if Tsrc is the assumed source zone temperature, and QDi is the energy delivered by fan i, then:

$$QDi = Bi + Ai * Tsrc$$

The Bi and Ai values are chosen from three sets of coefficients corresponding to each of the three basic fan operation constraints: minimum temperature difference, maximum fan capacity, and maximum energy needed. For any given source zone temperature, Tsrc, we want to choose the coefficients corresponding to the constraint which is the most restrictive. That is, we want to choose Bij and Aij such that:

$$QDi = \min [\ Bij + Aij * Tsrc \quad \text{for } j=1,2,3\ ]$$

The form of each of the coefficients differs, depending on the type of fan sink involved. The values used for these coefficients are given in Table 6-1 for the different cases. The coefficients are calculated in Subroutine EQMTB, and are used by Subroutine EQMTE to select the appropriate set and to evaluate the fan energy delivery for a given source zone temperature.

The full derivation of these coefficients is not presented here, but involves relatively simple algebraic manipulation of the three following equations:

$$QD = UAF * (Tsrc - T)$$

$$DTF = Tsrc - T$$

$$S = (N + QD)/D$$

where: Tsrc = assumed source zone temperature
T = fan sink temperature
QD = energy delivered by fan
UAF = fan maximum capacity
DTF = fan minimum temperature difference
N = sink zone total energy flow, excluding HVAC and fans (i.e., the numerator of the zone temperature defining equation)
D = sink zone total heat flow conductance (i.e., the denominator of the zone temperature defining equation)
S = appropriate sink zone HVAC setpoint = min [H , V , C]

by first noting that the fan sink temperature, T, can be written in one of the following forms:

$$T = H$$

$$T = (N + Hcap + QD)/D$$

$$T = (N + QD)/D$$

$$T = Tr$$

where: H = sink zone heating setpoint
Hcap = sink zone heater maximum capacity
Tr = rockbin charge outlet node temperature

For example, in the case of a fan sink zone which is heated, but for which the heater alone is inadequate to maintain the zone's heating setpoint in the current timestep, the derivation proceeds as follows:

1) Minimum temperature difference constraint

$$DTF = Tsrc - T$$

$$= Tsrc - (N + Hcap + QD)/D$$

hence

$$D * DTF = D * Tsrc - (N + Hcap + QD)$$

and

$$QD = -(D * DTF + N + Hcap) + D * Tsrc$$

2) Maximum energy delivery constraint

$$QD = UAF * (Tsrc - T)$$

$$= UAF * (Tsrc - (N + Hcap + QD)/D)$$

rearranging gives

$$(D + UAF) * QD = UAF * (D * Tsrc - (N + Hcap))$$

and

$$QD = -UAF*(N + Hcap)/(D + UAF) \quad + \quad (UAF * D)/(D + UAF) * Tsrc$$

3) Maximum energy needed constraint

$$H = (N + QD)/D$$

or

$$QD = H * D - N$$

The coefficients for the other types of fan sinks are derived similarly.

Table 6-1. Fan Energy Delivery Function Coefficients

TYPE OF FAN SINK	TYPE OF CONSTRAINT	FAN ENERGY DELIVERY FUNCTION COEFFICIENTS B	A
zone heated with adequate heater	min delta T	UAf*DTf	0
	max fan cap	-UAf*H	UAf
	max needed	H*D-N	0
zone heated with inadequate heater	min delta T	-(D*DTf+N+Hcap)	D
	max fan cap	-UAf*(N+Hcap)/(D+UAf)	(D*UAf)/(D+UAf)
	max needed	H*D-N	0
zone unheated but vented or cooled	min delta T	-(D*DTf+N)	D
	max fan cap	-(UAf*N)/(D+UAf)	(D*UAf)/(D+UAf)
	max needed	min[V,C]*D-N	0
zone neither heated, vented, nor cooled	min delta T	-(D*DTf+N)	D
	max fan cap	-(UAf*N)/(D+UAf)	(D*UAf)/(D+UAf)
	max needed	undefined	undefined
rockbin charge cycle	min delta T	UAf*DTf	0
	max fan cap	-UAf*Tr	UAf
	max needed	undefined	undefined
dummy source zone venting fan	min delta T	Vcap*(max[V,Tamb] -Tamb)	0
	max fan cap	-Vcap*Tamb	Vcap
	max needed	undefined	undefined
dummy source zone cooling fan	min delta T	Ccap	0
	max fan cap	-oo*C	+oo
	max needed	Ccap	0

key:

N = sink zone total energy flow (numerator of zone temperature equation)
D = sink zone total conductance (denominator of zone temperature equation)
UAf = fan maximum capacity
DTf = user-specified fan minimum temperature difference
H = sink zone heating setpoint
Hcap = sink zone heater maximum capacity
V = sink zone venting setpoint
Vcap = sink zone venting capacity
C = sink zone cooling setpoint
Ccap = sink zone cooler maximum capacity
Tamb = ambient air temperature
Tr = rockbin charge outlet node temperature

C. ROCKBIN DISCHARGE CONTROLLER

In the rockbin discharge cycle, energy is removed from the rockbin and supplied to the rockbin sink zone. The controller for the discharge cycle is handled separately from that for the charge cycle by Subroutine ROKON. This controller is designed to only displace all or a part of any heating load experienced by the rockbin sink zone. Note that this implies that whenever the heating equipment maximum capacity is inadequate to maintain the heating setpoint in the rockbin sink zone, then the amount of energy removed from the rockbin will be limited to the heater maximum capacity.

The rockbin is designed to have charge priority, so that whenever the fan controller algorithm has placed a rockbin in the charging mode, then that rockbin cannot discharge. If all three of the following conditions are met:

1) the rockbin is not in the charging mode;

2) the heating load for the rockbin sink zone, QH, is greater than zero;

and 3) the rockbin discharge outlet node temperature, Td, is greater than the zone air temperature, Tz, plus the user-specified minimum temperature difference, DTF (that is, $Td > Tz + DTF$);

then the rockbin discharge rate, QR, is calculated as:

$$QR = \min [QH , UAF * (Td - Tz)]$$

where UAF is the rockbin discharge fan maximum capacity.

In this case, the sink zone air temperature remains unchanged, but the sink zone heating load is reduced by the amount QR.

Section 6-5. LATENT LOAD ALGORITHMS

A. COOLER LATENT LOAD

SUNCODE includes only a cursory treatment of the effects of moisture within a building. It estimates the mechanical cooling equipment "latent load," or the energy removed by condensation of water from the zone air being cooled, in Subroutine WATER. Cooler operation is totally controlled by the sensible cooling load calculated by the HVAC controller algorithm. The user-specified cooler maximum capacity also refers only to this sensible load. For the latent load calculations, the algorithm assumes that return air from the cooling unit is always held at a constant, user-specified temperature. It is assumed that the cooler is of adequate capacity to cool air to this temperature, and is capable of handling arbitrarily large air flow rates. In any hour, the total cooler latent load, QL, is taken as:

$$QL = Hevap * Fc * (Wz - Wc)$$

where: Hevap= heat of vaporization of water = 1061.0 BTU/lb = 2451.0 kJ/kg

Fc = air flow rate through cooler

Wz = humidity ratio of zone air

Wc = humidity ratio of cooler return air

Humidity ratio calculations are discussed below. The cooler air flow rate, Fc, is calculated as

$$Fc = QC / (Pair * Cair * (Tz - Tc))$$

where: QC = total sensible cooling, as determined by the HVAC controller

Pair = air density

Cair = air heat capacity

Tz = zone air temperature

Tc = user-specified cooler return air temperature

Note that the latent load is only calculated when the cooler is operating, else Fc = 0. (When $Tz < Tc$ and $QC > 0$, then Fc is taken as 0.)

The total latent load is split into component loads attributed to infiltration, venting, fan operation, and internal moisture release.

These are calculated as:

$$QLi = Hevap * Fi * (Wa - Wz)$$

$$QLv = Hevap * Fv * (Wa - Wz)$$

$$QLf = Hevap * Ff * (Wf - Wz)$$

$$QLo = QL - QLi - QLv - QLf$$

where QLi = latent load attributed to infiltration

QLv = latent load attributed to venting

QLf = latent load attributed to fan operation

QLo = latent load attributed to internal moisture release, and all other effects

Fi = infiltration air flow rate

Fv = venting air flow rate

Ff = fan air flow rate

Wa = ambient humidity ratio

Wz = zone humidity ratio

Wf = humidity ratio of other zone(s) connected by a fan. For fan source zones, this is a weighted average of the humidity ratios in all sink zones for which fans are operating.

Subroutine WATER also performs an approximate accounting of water flows between zones due to fan operation, water flow to ambient due to infiltration and venting, and internal moisture release as specified by the user. It calculates a new zone humidity ratio hourly, as a function of the previous hour's humidity ratio, and net water flows during the hour.

$$Wz = \frac{VOL*Pair*Wz + Fi*Wa + Fv*Wa + Ff*Wf + Fc*Wc + (LG/Hevap)}{VOL*Pair + Fi + Fv + Ff + Fc}$$

where VOL = zone air volume;

LG = user-specified latent gain.

The zone humidity ratio calculated above is constrained to that corresponding to 100% relative humidity, by using Function HUMID to calculate the humidity ratio of saturated air at the zone's air temperature. Note that if this occurs, some water is lost from the system. Given the zone's humidity ratio, Wz, and the humidity ratio of saturated air at the same temperature, Ws, the zone's relative humidity, Hrel, is calculated as:

$$Hrel = (Wz * (Ws + 0.622))/(Ws * (Wz + 0.622))$$

The humidity ratio of cooler return air, Wc, is taken as the minimum of that of saturated air at the cooler return temperature and the zone's humidity ratio. The humidity ratio of ambient air, Wa, is calculated using Function HUMID and ambient dewpoint temperature read from the weather data. In this case, HUMID is called from Subroutine ENVIR.

B. HUMIDITY RATIO OF SATURATED AIR

Function HUMID calculates the humidity ratio of saturated air at a specified temperature and air pressure. The algorithm is taken from Reference [9]. For any location, air pressure, P, is taken as a constant value given by:

$$P = A * e^{(B * ELEV)}$$

where: ELEV = station elevation
A = standard air pressure = 29.921 in Hg = 760 mm Hg
B = value from curve fit = $-3.71781196*10^{-5}$/ft = $-1.219755*10^{-4}$/m

The humidity ratio, W, is then calculated by the following. Let T be the air temperature in degrees Kelvin.

```
If  T ≥ 273.16  then  Z  = 373.16/T
                      P1 = -7.90298 * (Z - 1.0)
                      P2 = 5.02808 * ALOG10(Z)
                      Z1 = 11.344 * (1.0 - (1.0/Z))
                      P3 = -1.3816E-7 * (10.0 ** Z1 - 1.0)
                      Z2 = -3.49149 * (Z - 1.0)
                      P4 = 8.1328E-3 * (10.0 ** Z2 - 1.0)

                else  Z  = 273.16/T
                      P1 = -9.09718 * (Z - 1.0)
                      P2 = -3.56654 * ALOG10(Z)
                      P3 = 0.876793 * (1.0 - (1.0/Z))
                      P4 = ALOG10(0.0060273)

Let PVS = 29.921 * (10.0 ** (P1 + P2 + P3 + P4))

Then  W = (0.622 * PVS)/(P - PVS)         for P in English units of in Hg

  or  W = (0.622 * PVS)/(P/25.4 - PVS)    for P in metric units of mm Hg
```

where ALOG10 indicates base 10 logarithm, the symbol "**" indicates exponentiation, and numeric values are written in FORTRAN exponent form.

Section 6-6. REFERENCES

1. Muncie, R.W.R., Heat Transfer Calculations for Buildings. Applied Science Publishers, Ltd., Essex, England (1979).

2. Duffie, John A. and Wm. Beckman, Solar Energy Thermal Processes. Wiley-Interscience, New York (1974).

3. Willier, Austin, "Prediction of Performance of Solar Collectors," Applications of Solar Energy for Heating and Cooling of Buildings. ASHRAE GRP 170 (1977), pp. VIII 1-VIII 14.

4. U.S. Dept. of Energy, "Variable Infiltration Coefficients," Economic Analysis: Energy Performance Standards for New Buildings, DOE/CS-0129 (1980), pp. A16-A19. See also: DOE 2.1 Reference Manuals.

5. McFarland, Robert D., Pasole: A General Simulation Program for Passive Solar Energy, University of California, Los Alamos Laboratory, LA-7433-MS (1978).

6. Moore, John E., et al., "Thermal Storage Wall Model Development in DOE-2 Computer Program," Proceedings of the 1981 Annual Meeting of the American Section of the International Solar Energy Society, Inc. (1981).

7. Hughes, P.J., S.A. Klein and D.J. Close, "Packed Bed Thermal Storage Models for Solar Air Heating and Cooling Systems," Journal of Heat Transfer (May 1976), ASME, pp. 336-338.

8. Klein, S.A., et al., TRNSYS: A Transient System Simulation Program. Solar Energy Laboratory, University of Wisconsin, Madison (1979).

9. ASHRAE Task Group on Energy Requirements, Procedure for Determining Heating and Cooling Loads for Computerizing Energy Calculations. American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc., New York (1976).

INPUT DATA SECTIONS

CONTENTS

Table A-1. Abbreviations for Units

	Abbreviation	Units
energy	BTU	British thermal unit
	KBTU	1000 British thermal units
	MBTU	1×10^6 British thermal units
	KJ	kilojoules
	MJ	megajoules
	GJ	gigajoules
temperature	F	Fahrenheit degrees
	C	Celsius degrees
length	FT	feet
	M	meters
	MM	millimeters
area	SF	square feet
	SM	square meters
time	H	hour
	D	day
	M	month
volumetric flow	CFM	cubic feet per minute
	CM/H	cubic meters per hour
velocity	MPH	miles per hour
	M/S	meters per second
mass	LB	pounds mass
	KG	kilogram
volume	CF	cubic feet
	CM	cubic meters
	MCF	1×10^6 cubic feet
	MCM	1×10^6 cubic meters
relative volumetric flow	AC/H	air changes per hour
latitude and longitude	DEG	degrees

Table A-2. Unit Conversion Coefficients

English = Metric/COEFFICIENT

Metric = English * COEFFICIENT

	English Units	Metric Units	COEFFICIENT
velocity	MPH	M/S	0.44704
energy	MBTU	GJ	1.05435
energy/area	KBTU/SF	MJ/SF	11.34893
power	KBTU/H	KW	0.2928751
heat transfer	BTU/H-F-SF	W/C-SM	5.674466
volumetric flow	CFM	CM/H	1.6990106
volume	CF	CM	0.02831685
temperature difference	F DEGREES	C DEGREES	0.5555556
temperature	DEGREES F = (DEGREES C + 17.77)/0.5555556 DEGREES C = (DEGREES F - 32.0)*0.5555556		

From LOADS module LSMUP.FOR, subroutine UNITS.

Table A-3. Key for Input Data Sections

Abbreviation for name of parameter	SURF. COEF.
Entry code. Shows allowable magnitude of number. AAAA indicates alphabetic value. SS.SSS indicates either alphabetic or numeric entry.	XX.XXX
Abbreviation for units in English system.	BTU/F -SF-H
Default value (if any) in English system.	1.46
Abbreviation for units in SI system.	W/SM-C
Default value (if any) in SI units.	8.278

Table A-4. Data Section Information Content

DATA SECTION NAME	INFORMATION CONTENT	CATEGORY	PAGE
RUNS	location and duration of simulation run	Primary-Req.	1
ZONES	defines building zones	Primary-Req.	2
INTERZONE	heat flow and solar transfer between zones	Primary-Opt.	3
WINDOWS	location and size of windows	Primary-Opt.	4
WALLS	location, size, and type of walls	Primary-Opt.	5
TROMBE.WALLS	location and size of Trombe walls	Primary-Opt.	6
FANS	location and type of fans	Primary-Opt.	7
ROCKBINS	location and type of rockbins	Primary-Opt.	8
SURFACES	orientation and size of exterior surfaces	Component	9
HVAC.TYPES	properties of HVAC equipment	Component	10
TROMBE.TYPES	detail description of Trombe wall types	Component	11
WALL.TYPES	single or multi-layered wall types	Component	12
MASS.TYPES	wall material properties	Component	13
PCM.TYPES	phase change material properties	Component	14
GLAZING.TYPES	glazing material properties	Component	15
BIN.TYPES	detail description of rockbin types	Component	16
FAN.TYPES	detail description of fan types	Component	17
OVERHANG.TYPES	dimensions of overhangs	Component	18
SIDEFIN.TYPES	dimensions of sidefins	Component	19
SKYLINE.TYPES	specification of skyline shading	Component	20
OUTPUT	output definition	Primary-Opt.	21
SCHEDULES	multi-season/24-hour operation schedules	Component	22
SEASONS	duration of seasons of year	Component	23
PARAMETERS	detailed simulation run parameters	Component	24
STATIONS	definition of weather data files	Component	25

KEY: Primary --All data entries made in this section will be used in the simulation.
-Req. --At least one entry is required.
-Opt. --Data entries are optional.
Component --Only those data entries referenced in a Primary or another Component section will be used in the simulation.

RUNS

RUN LABEL	STATION NAME	GROUND REFL.	GROUND TEMP.	-START- MON	DAY	-STOP-- MON	DAY	SKYLINE PROFILE	PAR. TYPE
AAAAAAAAAAAAAAAA	AAAAAAAAAA	S.SSSS	SSS.SS	AAA	XX.	AAA	XX.	AAAAAA	AAAAAA
n/a	n/a	FRAC.	F	DATE		DATE		n/a	n/a
		0.3000	50.00	JAN	1.	DEC	31.	<NONE>	<NONE>
n/a	n/a	FRAC.	C	DATE		DATE		n/a	n/a
		0.3000	10.00	JAN	1.	DEC	31.	<NONE>	<NONE>

PARAMETER	DESCRIPTION
RUN LABEL	Label for simulation. Printed as header on each page of output, if output is formatted for printer.
STATION NAME	Name of station for weather data. (Must be defined in STATIONS section).
GROUND REFL.	Ground reflectance for solar radiation. This parameter may be scheduled.
GROUND TEMP.	Ground temperature. This parameter may be scheduled.
START	First day of simulation run. MON must be one of: JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC. DAY is day of month, 1 through 31.
STOP	Last day of simulation run. MON and DAY as above.
SKYLINE PROFILE	Name of skyline profile type. (Must be defined in SKYLINE.TYPES section).
PAR. TYPE	Name of set of run control parameters to be used for this run. Normally, the hard-coded default values should be used.

ZONES

ZONE NAME	HVAC TYPE	FLOOR AREA	HGT	INFIL. RATE	SOLAR TO AIR	SOLAR LOST	INTERNAL GAIN	LATENT GAIN
AAAAAAAAAA	AAAAAAAAAA	XXXXX.X	XX.X	SSS.SSS	X.XXX	X.XXX	SSSS.SSS	SSSS.SSS
n/a	n/a	SF	FT	AC/HR	FRAC.	FRAC.	KBTU/HR	KBTU/HR
	<NONE>	1.0	1.0	<VARYS>	0.000	0.000	0.000	0.000
n/a	n/a	SM	M	AC/HR	FRAC.	FRAC.	KW	KW
	<NONE>	1.0	1.0	<VARYS>	0.000	0.000	0.000	0.000

PARAMETER	DESCRIPTION
ZONE NAME	Name of zone, used for reference below.
HVAC TYPE	Name of type of heating, ventilation and cooling equipment for this zone (Must be defined in HVAC.TYPES section).
FLOOR AREA	Floor area of zone. Used with ceiling height to get volume.
HGT	Ceiling height of zone.
INFIL. RATE	Infiltration rate for this zone. Default entry of VARYS causes the infiltration rate to be calculated hourly as a function of wind speed and difference between zone air and ambient temperatures. This parameter can be scheduled.
SOLAR TO AIR	Fraction of Total Solar Available in zone which is input directly to air temperature node to account for lightweight surfaces, furniture, etc.
SOLAR LOST	Fraction of Total Solar Available in zone which is lost from system, to account for overall cavity absorptivity of zone.
INTERNAL GAIN	Rate of sensible heat gain from internal sources (appliances, occupants, etc.). This parameter can be scheduled.
LATENT GAIN	Rate of latent heat gain from internal sources. This parameter can be scheduled.

INTERZONE

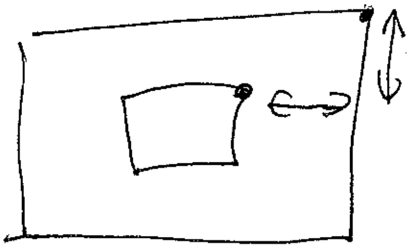
SOURCE ZONE	SINK ZONE	CONDUCTANCE COEF.	SOLAR TRANSFER	REVERSE TRANSFER
AAAAAAAAAA	AAAAAAAAAA	XXXXX.XX	S.SSSSS	S.SSSSS
n/a	n/a	BTU/F-H 0.00	FRAC. 0.00000	FRAC. 0.00000
n/a	n/a	W/C 0.00	FRAC. 0.00000	FRAC. 0.00000

PARAMETER	DESCRIPTION
SOURCE ZONE	Name of user-defined zone (must be defined in ZONES section above), for one side of conductance connection, or for the source of solar transfer and the sink for reverse solar transfer.
SINK ZONE	Name of user-defined zone (must be defined in ZONES section above), or one of the predefined zones: AMBIENT or GROUND. Zone is one side of conductance connection, or the sink for solar transfer, and the source of reverse solar transfer. Only CONDUCTANCE may be defined if SINK ZONE is AMBIENT or GROUND.
CONDUCTANCE COEF.	Thermal transfer rate between SOURCE ZONE and SINK ZONE.
SOLAR TRANSFER	Fraction of Total Transmitted Solar into SOURCE ZONE which is to be transferred to SINK ZONE, to account for glass doors between zones, etc. This parameter may be scheduled.
REVERSE TRANSFER	Fraction of Total Transmitted Solar into SINK ZONE which is to be transferred to SOURCE ZONE. This parameter may be scheduled.

WINDOWS

INTERIOR ZONE	EXTERIOR SURFACE	GLAZING TYPE	HEIGHT	LENGTH	---LOCATION---- HORZ.	VERT.
AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	XXXX.XX	XXXX.XX	XXXX.XX	XXXX.XX
n/a	n/a	n/a	FT 1.00	FT 1.00	FT 0.00	FT 0.00
n/a	n/a	n/a	M 1.00	M 1.00	M 0.00	M 0.00

PARAMETER	DESCRIPTION
INTERIOR	Name of user defined zone which contains window (must be be defined in ZONES section).
EXTERIOR SURFACE	Name of exterior surface in which WINDOW is located (must be defined in SURFACES section).
GLAZING TYPE	Name of type of glazing system in WINDOW (must be defined in GLAZING.TYPES section).
HEIGHT	Height of window. Used to obtain area of window.
LENGTH	Length of window. Used to obtain area of window.
LOCATION HORZ	Horizontal distance from upper right corner of WINDOW to upper right corner of EXTERIOR SURFACE, used for shading calculations. Left and right are defined as looking at the surface from exterior. This parameter is not required if shading by overhangs or sidefins is not being used.
LOCATION VERT	"Vertical" distance from upper right corner of WINDOW to upper right corner of EXTERIOR SURFACE, measured in the plane of the exterior surface. Left and right are defined as looking at the surface from exterior. This parameter is not required if shading by overhangs or sidefins is not being used.



WALLS

WALL TYPE	---FRONT/INTERIOR SIDE--			---BACK/EXTERIOR SIDE---			WALL AREA
	ZONE NAME	SURF COEF.	SOLAR COEF.	ZONE OR SURFACE	SURF COEF.	SOLAR COEF.	
AAAAAAAAAA	AAAAAAAAAA	XX.XXX	X.XXXX	AAAAAAAAAA	XX.XXX	X.XXXX	XXXXX.X
n/a	n/a	BTU/F -SF-H	FRAC.	n/a	BTU/F -SF-H	FRAC.	SF
		1.460	<AREA>		1.460	<AREA>	1.0
n/a	n/a	W/C -SM	FRAC.	n/a	W/C -SM	FRAC.	SM
		8.278	<AREA>		8.278	<AREA>	1.0

PARAMETER	DESCRIPTION
WALL TYPE	Name of type of wall (must be defined in WALL.TYPES section), or the predefined non-massive wall type specified by R-n, where n is the numeric value of the total heat resistance in the wall (e.g. R-20). The FRONT/ INTERIOR side of the wall corresponds to the first layer defined in a multi-layered wall.

FRONT/INTERIOR SIDE

ZONE	Name of user zone on "front" or interior side of wall (must be defined in ZONES section).
SURF COEF	Combined radiation and convection conductance at surface.
SOLAR COEF.	Fraction of Total Solar Available in FRONT/INTERIOR ZONE which is absorbed by wall.

BACK/EXTERIOR SIDE

ZONE OR SURFACE	Name of "space" on "back" or exterior side of wall. May be either a user zone defined in ZONES section; or one of the keywords AMBIENT and GROUND, for an exterior wall without solar on the exterior; or a surface defined in the SURFACES section, for an exterior wall with solar on the exterior. Walls completely contained within a zone are modeled by specifying the same ZONE on both surfaces.
SURF COEF	Combined radiation and convection conductance at surface.
SOLAR COEF.	If BACK/EXTERIOR "space" is a user zone, then this is the fraction of Total Available Solar in that zone which is absorbed on wall. Otherwise, this is the absorptivity of the exterior side of the wall (not used if keyword AMBIENT or GROUND is specified above).
WALL	Area of wall.

AREA
TROMBE.WALLS

INTERIOR ZONE	EXTERIOR SURFACE	TROMBE TYPE	SURF. COEF	SOLAR COEF.	HEIGHT	LENGTH	—LOCATION— HORZ	—LOCATION— VERT
AAAAAAAAAA	AAAAAAAAAA	AAAAAA	XX.XXX	X.XXXX	XX.XX	XXX.XX	XXX.XX	XXX.XX
n/a	n/a	n/a	BTU/F -SF-H	FRAC.	FT	FT	FT	FT
			1.460	<AREA>	1.00	1.00	0.00	0.00
n/a	n/a	n/a	W/C -SM	FRAC.	M	M	M	M
			8.278	<AREA>	1.00	1.00	0.00	0.00

PARAMETER	DESCRIPTION
INTERIOR ZONE	Name of user zone in which Trombe wall is located (must be defined in ZONES section).
EXTERIOR SURFACE	Name of exterior surface in which the Trombe wall is located (must be defined in SURFACES section).
TROMBE TYPE	Name of type of Trombe wall (must be defined in TROMBE.TYPES section).
SURF. COEF	Combined radiation and convection conductance for <u>interior</u> surface of Trombe wall.
SOLAR COEF.	Fraction of Total Solar Available in INTERIOR ZONE which is absorbed on the <u>interior</u> surface of Trombe wall.
HEIGHT	Height of Trombe wall.
LENGTH	Length of Trombe wall.
LOCATION HORZ.	Horizontal distance from upper right corner of Trombe wall to upper right corner of EXTERIOR SURFACE, used for shading calculations. Left and right are defined as looking at surface from exterior. This parameter is not required if shading by overhangs or sidefins is not being used.
LOCATION VERT.	"Vertical" distance from upper right corner of Trombe wall to upper right corner of EXTERIOR SURFACE, measured in the plane of EXTERIOR SURFACE. Left and right are defined as looking at surface from exterior. This parameter not required if shading by overhangs or sidefins is not being used.

FANS

SINK ZONE	SOURCE ZONE	FAN TYPE	OFF SEASON
AAAAAAAAAA	AAAAAAAAAA	AAAAAAAA	AAAAAAAA
n/a	n/a	n/a	n/a <NONE>
n/a	n/a	n/a	n/a <NONE>

PARAMETER	DESCRIPTION
SINK ZONE	Name of user zone (must be defined in ZONES section), to which the fan delivers warm air. See text for restrictions on the placement of fans.
SOURCE ZONE	Name of user zone (must be defined in ZONES section), from which the fan removes warm air. See text for restrictions on the placement of fans.
FAN TYPE	Name of fan type (must be defined in FAN.TYPES section).
OFF SEASON	Name of season during which fan is not operational (must be defined in SEASONS section).

ROCKBINS

SINK ZONE	SOURCE ZONE	ROCKBIN TYPE	ZONE FOR PASSIVE LOSSES	PASSIVE LOSS TO ZONE	PASSIVE LOSS TO AMBIENT	PASSIVE LOSS TO GROUND
AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	XXXX.XXX	XXXX.XXX	XXXX.XXX
n/a	n/a	n/a	n/a <NONE>	BTU/H-F 0.000	BTU/H-F 0.000	BTU/H-F 0.000
n/a	n/a	n/a	n/a <NONE>	W/C 0.000	W/C 0.000	W/C 0.000

PARAMETER	DESCRIPTION
SINK ZONE	Name of user-defined zone (must be defined in ZONES section above) to which rockbin discharges. *
SOURCE ZONE	Name of user-defined zone (must be defined in ZONES section above) from which rockbin is charged. *
ROCKBIN TYPE	Name of rockbin type (must be defined in BIN.TYPES section below).
ZONE FOR PASSIVE LOSSES	Name of user-defined zone (must be defined in ZONES section above) to which rockbin losses heat passively.
PASSIVE LOSS TO ZONE	Rate of heat loss from rockbin to zone named above.
PASSIVE LOSS TO AMBIENT	Rate of passive heat loss to AMBIENT temperature node.
PASSIVE LOSS TO GROUND	Rate of passive heat loss to GROUND temperature node.

---

* See text for restrictions on placement of rockbins.

SURFACES

EXTERIOR SURFACE	COMPASS AZIMUTH	TILT	HEIGHT	LENGTH	OVERHANG TYPE	LEFT SIDEFIN	RIGHT SIDEFIN
AAAAAAAAAA	XXX.X	XX.X	XXXX.XX	XXXX.XX	AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA
n/a	DEG 180.0	DEG 90.0	FT 1.0	FT 1.0	n/a <NONE>	n/a <NONE>	n/a <NONE>
n/a	DEG 180.0	DEG 90.0	M 1.0	M 1.0	n/a <NONE>	n/a <NONE>	n/a <NONE>

PARAMETER	DESCRIPTION
EXTERIOR SURFACE	Name of exterior surface, referenced in the WINDOW, WALL, AND TROMBE.WALL sections.
COMPASS AZIMUTH	Azimuth orientation of surface, with true North=0, East=90, South=180, and West=270 (values from 0 to 360 allowed).
TILT	Tilt of surface from horizontal, with horizontal=0, and vertical=90 (values from 0 to 90 allowed).
HEIGHT	Height of surface. Dimensions of surface are used for calculation of shading effects.
LENGTH	Length of surface. See description of HEIGHT above.
OVERHANG TYPE	Name of overhang for shading the surface (must be defined in OVERHANG.TYPES section)
LEFT SIDEFIN	Name of sidefin on left side of surface. (Must be defined in SIDEFIN.TYPES section below). Left and right are defined as looking at surface from exterior.
RIGHT SIDEFIN	Name of sidefin on right side of surface. (Must be defined in SIDEFIN.TYPES section below). Left and right are defined as looking at surface from exterior.

HVAC.TYPES

HVAC TYPE AAAAAAAAAA	HEATING SETPOINT SSS.SSS	VENTING SETPOINT SSS.SSS	COOLING SETPOINT SSS.SSS	HEATING CAPACITY XXXX.XXX	VENTING CAPACITY XXXXX.XX	COOLING CAPACITY XXXX.XXX	COOLER COIL XX.X
n/a	F <NONE>	F <NONE>	F <NONE>	KBTU/H <ADEQ>	AC/H <ADEQ>	KBTU/H <ADEQ>	F 55.0
n/a	C <NONE>	C <NONE>	C <NONE>	KW <ADEQ>	AC/H <ADEQ>	KW <ADEQ>	C 12.8

PARAMETER	DESCRIPTION
HVAC TYPE	Name of HVAC equipment type, used for reference by ZONES section.
HEATING SETPOINT	Heating thermostat setpoint; the temperature which heating equipment will attempt to maintain. If no HEATING SETPOINT is specified, zone has no heating. This parameter may be scheduled.
VENTING SETPOINT	Venting thermostat setpoint; the temperature at which venting is initiated and which ventilation equipment will attempt to maintain. If no VENTING SETPOINT is specified, zone has no venting. This parameter may be scheduled.
COOLING SETPOINT	Cooling thermostat setpoint; the temperature at which cooling is initiated and which cooling equipment will attempt to maintain. If no COOLING SETPOINT is specified, zone has no cooling. This parameter may be scheduled.
HEATING CAPACITY	Maximum capacity of heating equipment. Default signifies that heating equipment is of adequate capacity to meet all heating loads.
VENTING CAPACITY	Maximum capacity of ventilation equipment. Default signifies that ventilation is of adequate capacity to maintain the VENTING SETPOINT, whenever AMBIENT is cooler.
COOLING CAPACITY	Maximum capacity of cooling equipment. Default signifies that cooling is of adequate capacity to meet all cooling loads.
COOLER COIL	Temperature of cooling equipment coil. Used for calculations modeling the interaction of cooling equipment and latent loads.

TROMBE.TYPES

TROMBE TYPE	WALL TYPE	GLAZING TYPE	VENT OVER HEAT?	SURF COEF	WALL EXT. ABS.	VENT AREA RATIO	HEIGHT BETWEEN VENTS	VENT COEF
AAAAAA	AAAAAAAAAA	AAAAAAAAAA	A	XX.XXX	X.XXX	X.XXXX	XX.XXX	XX.XXX
n/a	n/a	n/a	n/a	BTU/F -SF-H	FRAC.	FRAC.	FT	FRAC.
			Y	1.460	1.000	0.0000	0.000	0.000
n/a	n/a	n/a	n/a	W/SM-C	FRAC.	FRAC.	M	FRAC.
			Y	8.278	1.000	0.0000	0.000	0.000

PARAMETER	DESCRIPTION
TROMBE TYPE	Name of Trombe wall type, used for reference by TROMBE.WALLS section.
WALL TYPE	Name of type of wall in the Trombe wall (must be defined in WALL.TYPES section). Use of predefined non-massive wall type specified as R-n is not allowed for Trombe walls.
GLAZING TYPE	Name of type of glazing system in the Trombe wall (must be defined in GLAZING.TYPES section below).
VENT OVER HEAT?	Allows vents to be closed when their operation would produce overheating in sink zone. Answer Y for yes if vents will cause overheating, or N for no if they do not.
SURF COEF	Combined radiation and convection conductance on <u>exterior</u> side of Trombe wall.
WALL EXT. ABS.	Absorptivity for solar radiation (transmitted through the glazing system) of the exterior surface of the wall.
VENT AREA RATIO	Area of one row of vents provided for thermocirculation divided by the total area of the Trombe wall.
HEIGHT BETWEEN VENTS	Distance from bottom row of vents to top row of vents.
VENT COEF	Dimensionless multiplier specifying the resistance to thermocirculation air flow caused by the vent openings. Values of 0.2 to 0.8 are recommended.

WALL.TYPES

WALL TYPE	LAYER # 1	LAYER # 2	LAYER # 3	LAYER # 4	LAYER # 5	LAYER # 6
AAAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA
n/a	n/a	n/a <NONE>	n/a <NONE>	n/a <NONE>	n/a <NONE>	n/a <NONE>
n/a	n/a	n/a <NONE>	n/a <NONE>	n/a <NONE>	n/a <NONE>	n/a <NONE>

PARAMETER	DESCRIPTION
WALL TYPE	Name of wall type, used for reference by TROMBE.TYPES, and WALLS sections. The first two characters of WALL TYPE should not be R-.
LAYER # n	Name of type of material composing each layer in the wall. May be name of a mass type defined in MASS.TYPES section, or name of a phase change material type defined in PCM.TYPES section below, or a non-massive layer specified as R-n where n is the total thermal resistance of the layer (e.g. R-20). LAYER # 1 corresponds to the front or interior side of the wall, the last layer specified corresponds to the back or exterior side of the wall.

MASS.TYPES

MASS TYPE	CONDUCTIVITY	DENSITY	SPECIFIC HEAT	THICKNESS	NODES
AAAAAAAAAA	X.XXXX	XXXX.XXX	X.XXXX	XX.XXXX	XX.
n/a	BTU/FT-F-H	LB/CF	BTU/LB-F	FT	n/a
	<N/A>	1.000	1.0000	1.0000	1.
n/a	W/M-C	KG/CM	KJ/KG-C	M	n/a
	<N/A>	1.000	1.0000	1.0000	1.

PARAMETER	DESCRIPTION
MASS TYPE	Name of type of material, used for reference by WALL.TYPES section. The first two letters of MASS TYPE should not be R-.
CONDUCTIVITY	Thermal conductivity of material. Water may be defined as infinite conductivity (default value). Any wall layer defined as being a MASS TYPE with infinite conductivity will have one node in the simulation.
DENSITY	Density of material.
SPECIFIC HEAT	Specific heat of material.
THICKNESS	Thickness of material.
NODES	Number of nodes to use in simulation of material. Note that for mass types with infinite conductivity (default value of conductivity), only one node may be specified.

PCM.TYPES

PCM TYPE	CONDUCTIVITY	DENSITY	SPECIFIC HEAT	THICKNESS	HEAT OF FUSION	MELTING POINT
AAAAAAAAAA	X.XXXX	XXXX.XXX	X.XXXX	XX.XXXX	XXXX.XX	XXX.X
n/a	BTU/FT-F-H	LB/CF	BTU/LB-F	FT	BTU/LB	DEG F
	<N/A>	1.000	1.0000	1.0000	1.00	80.0
n/a	W/M-C	KG/CM	KJ/KG-C	M	KJ/KG	DEG C
	<N/A>	1.000	1.0000	1.0000	1.00	26.7

PARAMETER	DESCRIPTION
PCM TYPE	Name of phase change material type, used for reference by WALL.TYPES section above. First two letters of name should not be R-.
CONDUCTIVITY	Conductivity of phase change material. The default value <N/A> specifies infinite conductivity.
DENSITY	Density of phase change material.
SPECIFIC HEAT	Specific heat of phase change material.
THICKNESS	Thickness of phase change material layer.
HEAT OF FUSION	Heat of fusion of phase change material.
MELTING POINT	Melting point temperature of phase change material.

Note that one mass node will be used to simulate each phase change material layer.

GLAZING.TYPES

GLAZING TYPE	GLAZING U VALUE	SHADING COEF.	EXTINCTION COEF.	INDEX OF REFRACTION	THICKNESS OF LAYER	NUMBER OF LAYERS
AAAAAAAAAA	SS.SSSSS	SS.SSSSS	X.XXXX	X.XXXX	X.XXXX	XX.
n/a	BTU/SF-F-H	FRAC.	1/IN	NONE	IN	n/a
	1.0	1.000	0.5000	1.5260	0.1250	1.
n/a	W/SM-C	FRAC.	1/MM	NONE	MM	n/a
	5.67	1.000	0.0197	1.5260	3.1750	1.

PARAMETER	DESCRIPTION
GLAZING TYPE	Name of type of glazing, used for reference by WINDOWS and TROMBE.TYPES sections.
GLAZING U VALUE	Heat transfer rate through glazing system (air to air). Note that the glazing system U-value may be scheduled to model the use of movable insulation. The maximum U-value consistent with accurate evaluation of the inner glazing surface temperature is 1.46 Btu/SF-F-H (8.284 W/SM-C).
SHADING COEF.	Shading coefficient of glazing system. A value of 1.0 gives maximum solar gain to interior; a value of 0.0 gives no solar gain to the interior. Note that the shading coefficient may be scheduled to model movable insulation or solar exclusion devices (for instance white curtains closed during summer).
EXTINCTION COEF.	Extinction coefficient of glazing material per unit thickness.
INDEX OF REFRACTION	Index of refraction of glazing material.
THICKNESS OF LAYER	Thickness of one layer of glazing material.
NUMBER OF LAYERS	Number of layers of glazing material in glazing system.

BIN.TYPES

ROCKBIN TYPE	BI-DIR	LENGTH	CROSS-SECTION	HEAT CAPACITY	AXIAL COND.	CHARGE FAN TYPE	DISCHARGE FAN TYPE	CHARGE OFF SEASON
AAAAAAAAAA	A	XXX.XX	XXX.XX	XXXX.XXX	XX.XXX	AAAAAAAA	AAAAAAAA	AAAAAAAA
n/a	Y/N	FT	SF	BTU/CF-F	BTU/F HR-FT	n/a	n/a	n/a
	Y	1.00	1.00	1.000	0.000			<NONE>
n/a	Y/N	M	SM	KJ/CM-C	W/M-C	n/a	n/a	n/a
	Y	1.00	1.00	1.000	0.000			<NONE>

PARAMETER	DESCRIPTION
ROCKBIN TYPE	Name of rockbin type, used for reference by ROCKBINS section above. See text for a discussion of rockbin operation and restrictions on placement.
BI-DIR	Indicator for bi-directional flow in rockbin. Respond Y for Yes to indicate rockbin has bi-directional flow. Respond N for No to indicate rockbin does not have bi-directional flow.
LENGTH	Length of rockbin.
CROSS-SECTION	Cross-sectional area of rockbin.
HEAT CAPACITY	Heat capacity per unit volume of rockbin.
AXIAL COND.	Effective 'conductance' of rockbin in axial direction. Allows rockbin to de-stratify during periods of no flow.
CHARGE FAN TYPE	Name of fan type used for charging (must be defined in FAN.TYPES section).
DISCHARGE FAN TYPE	Name of fan type used for discharging (must be defined in FAN.TYPES section).
CHARGE OFF SEASON	Name of season during which rockbin is not charged (must be defined in SEASONS section).

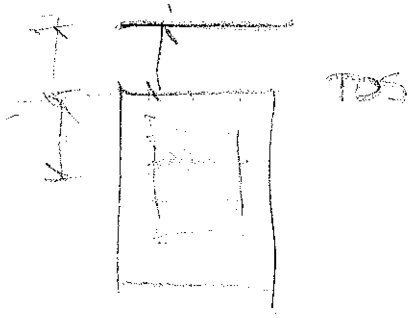
FAN.TYPES

FAN TYPE	FAN CAPACITY	MINIMUM TEMP. DIFF.
AAAAAAAA	XXXXX.X	XX.XX
n/a	CFM 1.0	F 0.00
n/a	CM/H 1.0	C 0.00

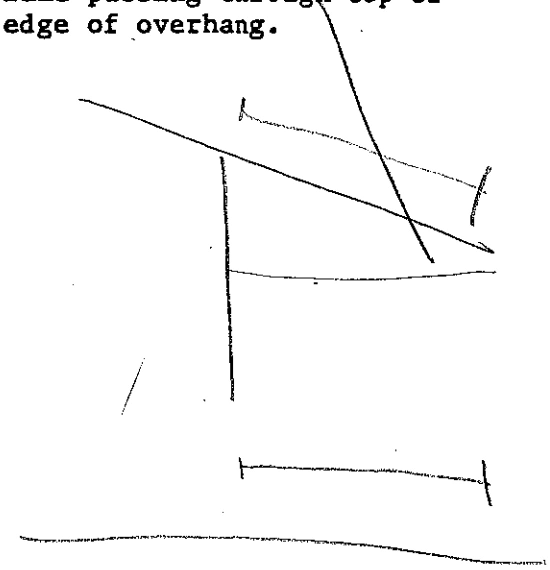
PARAMETER	DESCRIPTION
FAN TYPE	Name of fan type.
FAN CAPACITY	Maximum volumetric flow rate of fan.
MINIMUM TEMP. DIFF.	Minimum temperature difference which must exist before fan will turn on.

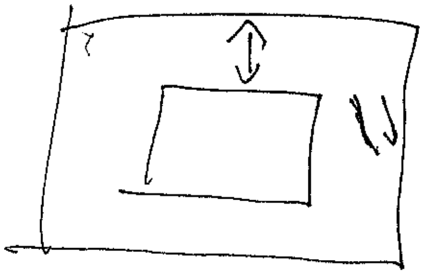
OVERHANG.TYPES

OVERHANG TYPE	VERTICAL OFFSET	HORIZONTAL PROJECTION
AAAAAAAAAA	XX.XX	XX.XX
n/a	FT 0.00	FT 0.00
n/a	M 0.00	M 0.00



PARAMETER	DESCRIPTION
OVERHANG TYPE	Name of overhang type, used for reference by SURFACES section. See text for the details of shading specification.
VERTICAL OFFSET	Distance from top of surface being shaded to overhang.
HORIZONTAL PROJECTION	Distance from vertical plane passing through top of surface being shaded to edge of overhang.





SIDEFIN.TYPES

SIDEFIN TYPE AAAAAAAAA	OFFSET TO SIDE XX.XX	LENGTH OF PROJECTION XX.XX
n/a	FT 0.00	FT 0.00
n/a	M 0.00	M 0.00

PARAMETER	DESCRIPTION
SIDEFIN TYPE	Name of sidefin, used for reference by SURFACES section. See text for a detailed discussion of shading specification.
OFFSET TO SIDE	Distance from side (left side for LEFT SIDEFIN, right side for RIGHT SIDEFIN) of surface being shaded to sidefin.
LENGTH OF PROJECTION	Distance from vertical plane passing through top of surface being shaded to farthest point on sidefin, measured normal to the vertical plane.

SKYLINE.TYPES

```
SKYLINE  --------- ALTITUDE ANGLE OF SKYLINE (DEGREES) --------
PROFILE  ---EAST------------------SOUTH-----------------WEST---
 TYPE    100   80   60   40   20    0   20   40   60   80  100
AAAAAA   XX.X XX.X XX.X XX.X XX.X XX.X XX.X XX.X XX.X XX.X XX.X

 n/a     ---------------------- degrees ----------------------
          0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
```

PARAMETER	DESCRIPTION
SKYLINE PROFILE TYPE	Name of skyline profile type, used for reference by RUNS section.
ALTITUDE ANGLE	Angle of skyline, given in degrees, for each of eleven segments (each 20 degrees wide) centered around due South. The first segment (indicated as EAST-100) refers to the segment ranging from 20 degrees North of East to due East. The middle segment (indicated as SOUTH-0) refers to the segment ranging from 10 degrees East of South to 10 degrees West of South. The last segment (indicated as WEST-100) refers to the segment ranging from due West to 20 degrees North of West.

OUTPUT

OUTPUT TYPE	TIME PERIOD	UNITS	OUTPUT SEASON	BUILDING ELEMENT	OUTPUT SECTION	FORMAT?
AAAAAAAA	A	A	AAAAAAAA	XXXX.	XXXX.	A
n/a	H/D/M	E/M	n/a			Y/N
ALL	M	E	<ALL>	<ALL>	<ALL>	Y

PARAMETER	DESCRIPTION
OUTPUT TYPE	One of the keywords: ALL, AMBIENT, BUILDING, ZONES, WINDOWS, WALLS, SURFACES, FANS, ROCKBINS, TROMBES. Designates output data sections which are desired. If no entries are made in the output section, the following hard-coded entry is assumed: AMBIENT M E/M <ALL> <ALL> <ALL> Y BUILDING M E/M <ALL> <ALL> <ALL> Y where E/M indicates that the units are the same as those used for the building input.

16
22
25
30
34

TIME PERIOD	Time step for output: hourly, daily, or monthly. Monthly output also produces run totals.
UNITS	Units to use for output: E for English units, M for metric units.
OUTPUT SEASON	User defined SEASON for which output will be made. The default value <ALL> produces output at the interval specified under TIME PERIOD for the duration of the run.
BUILDING ELEMENT	This parameter allows the selection of output for individual building elements of any OUTPUT TYPE, for instance, the third window is selected by entering WINDOWS under OUTPUT TYPE and 3 under BUILDING ELEMENT. The default value <ALL> produces output for all elements defined in the building description.
OUTPUT SECTION	This parameter allows selection of individual output data sections for a given OUTPUT TYPE and BUILDING ELEMENT. For instance, the user may produce output of the heating, venting, and cooling energy flows for one or more zones by entering ZONES under OUTPUT TYPE and 4 under OUTPUT SECTION. The default <ALL> produces output for all sections defined for the specified OUTPUT TYPE.
FORMAT?	Two types of output file can be produced. The first type is intended to be easily read. This type contains header lines which indicate the variables and their units. Selected by entering Y (the default value) for yes. The second type is intended to be used as input for other computer programs, and contains no header information. Selected by entering N for no.

SCHEDULES

SCHEDULE	SEASON	HR	VALUE	HR	VALUE	HR	VALUE	HR	VALUE
AAAAAAAA	AAAAAAAA	XX.	XXXX.XXX	XX.	XXXX.XXX	XX.	XXXX.XXX	XX.	XXXX.XXX
n/a	n/a	n/a	varies	n/a	varies	n/a	varies	n/a	varies
n/a	n/a	n/a	varies	n/a	varies	n/a	varies	n/a	varies

PARAMETER	DESCRIPTION
SCHEDULE NAME	Name of schedule, used for reference by one of the parameters which can be scheduled. More than one line may be specified for each schedule. The first line in any schedule specification must have the SCHEDULE NAME specified. The remaining lines, if immediately following, may repeat the SCHEDULE NAME or be blank for that parameter. Each new schedule must start with a new SCHEDULE NAME. See text for detailed discussion of use of schedules.
SEASON NAME	Name of user-defined season to use for schedule (must be defined in SEASONS section below). Default is same as for SCHEDULE NAME, the first line of a multi-line entry must specify the SEASON NAME, in following lines it is optional.
HR	First hour for which the following value is to be used. Hours are numbered 1 (1 a.m.) to 24 (midnight)
VALUE	Value used for hours including the preceding hour and up the following hour. The entry

HR	VALUE	HR
20	100	22

means the value 100 is used for the time period from 8 PM until 10 PM. The last VALUE given is assumed to apply for the rest of the day. Thus values which are the same for every hour of the day, but which vary by season, can be specified by giving just the value for the first hour of the day for each season.

Note that up to 4 sets of VALUE and HOUR pairs may be specified in each line. See EXAMPLES section on manual for some sample schedule definitions.

SEASONS

SEASON NAME	START DATE MON DAY	STOP DATE MON DAY	DAY OF WEEK [ALL/M-F/S-S]
AAAAAAAA	AAA XX.	AAA XX.	AAA
n/a	DATE	DATE	n/a ALL
n/a	DATE	DATE	n/a ALL

PARAMETER	DESCRIPTION
SEASON NAME	Name of SEASON for reference by SCHEDULES section.
START DATE	First day in season being defined. MON must be one of: JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC. DAY is day of month, 1 through 31.
STOP DATE	Last day in season being defined. MON and DAY as above.
DAY OF WEEK	Weekday and weekend schedules. ALL specifies all days. M-F specifies Monday through Friday. S-S specifies Saturday through Sunday. Default is ALL.

PARAMETERS

PAR. TYPE	INFIL. MULT.	MAX ZONE	ZONE CRIT.	MAX TROM	TROM CRIT.	INIT TEMP	H DD BASE	C DD BASE	WEEK DAY	DIF. ANG.
AAAAAA	XX.XX	XXX.	XX.XXX	XXX.	XX.XXX	XX.X	XX.X	XX.X	X.	XX.X
n/a	FRAC. 1.00	n/a 50.	F 0.100	n/a 50.	F 0.100	F 65.0	F 65.0	F 65.0	n/a 1.	DEG. 60.0
n/a	FRAC. 1.00	n/a 50.	C 0.050	n/a 50.	C 0.050	C 18.3	C 18.3	C 18.3	n/a 1.	DEG. 60.0

PARAMETER	DESCRIPTION
PAR. TYPE	Name of set of run control parameters, used for reference by RUNS section. Normally, the hard-coded default values for these parameters should be used, and then no entries need be made in this section.
INFIL. MULT.	Coefficient to multiply infiltration rate predicted by wind and temperature dependent formula, which is used whenever INFILTRATION RATE (ZONES section) is defaulted.
MAX ZONE	Maximum number of iterations to allow in calculation of zone air temperatures.
ZONE CRIT.	Criterion for defining convergence for zone air temperature calculations.
MAX TROM	Maximum number of iterations to allow in calculations of Trombe wall air gap temperatures with thermocirculation.
TROM CRIT.	Criterion for defining convergence for Trombe wall air gap temperature calculations.
INIT TEMP	Initial temperature to use for all mass and zone nodes.
H DD BASE	Base temperature to use for calculating heating degree days.
C DD BASE	Base temperature to use for calculating cooling degree days.
WEEK DAY	Day of week of Jan. 1st. Monday = 1, ... Sunday = 7. Used to adjust weekday and weekend distinction when using weather data for a particular calendar year.
DIF. ANG.	Angle of incidence at which transmissivity for diffuse radiation is calculated.

STATIONS

STATION NAME	LAT.	LONG.	ELEV.	FILENAME	DATA TYPE	UNITS	-START- MON DAY	-STOP-- MON DAY
AAAAAAAAAA	XX.XX	XXX.X	XXXXX.	AAAAAAAAAA	XX.	A	AAA XX.	AAA XX.
n/a	DEG	DEG <N/A>	FT 0.	n/a	n/a 1.	n/a E	DATE JAN 1.	DATE DEC 31.
n/a	DEG	DEG <N/A>	M 0.	n/a	n/a 1.	n/a E	DATE JAN 1.	DATE DEC 31.

PARAMETER	DESCRIPTION
STATION NAME	Name of station, used for reference by RUNS section.
LAT.	Latitude of station.
LONG.	Longitude of station.
ELEV.	Elevation above sea level of station.
FILENAME	Name of file which contains weather data for this station.
DATA TYPE	Index to distinguish different types of weather data files, depends on implementation.
UNITS	Units of data contained in specified file; one of keywords: E for English units, M for metric units.
START	Date of first day of data in weather file.
STOP	Date of last day of data in weather file.

OUTPUT DATA SECTIONS

CONTENTS

Table B-1. Abbreviations for Units

	Abbreviation	Units
energy	BTU	British thermal unit
	KBTU	1000 British thermal units
	MBTU	$1x10^6$ British thermal units
	KJ	kilojoules
	MJ	megajoules
	GJ	gigajoules
temperature	F	Fahrenheit degrees
	C	Celsius degrees
length	FT	feet
	M	meters
	MM	millimeters
area	SF	square feet
	SM	square meters
time	H	hour
	D	day
	M	month
volumetric flow	CFM	cubic feet per minute
	CM/H	cubic meters per hour
velocity	MPH	miles per hour
	M/S	meters per second
mass	LB	pounds mass
	KG	kilogram
volume	CF	cubic feet
	CM	cubic meters
	MCF	$1x10^6$ cubic feet
	MCM	$1x10^6$ cubic meters
relative volumetric flow	AC/H	air changes per hour
latitude and longitude	DEG	degrees

Table B-2. Unit Conversion Coefficients

English = Metric/COEFFICIENT   Metric = English * COEFFICIENT

	English Units	Metric Units	COEFFICIENT
velocity	MPH	M/S	0.44704
energy	MBTU	GJ	1.05435
energy/area	KBTU/SF	MJ/SF	11.34893
power	KBTU/H	KW	0.2928751
heat transfer	BTU/H-F-SF	W/C-SM	5.674466
volumetric flow	CFM	CM/H	1.6990106
volume	CF	CM	0.02831685
temperature difference	F DEGREES	C DEGREES	0.5555556
temperature	DEGREES F = (DEGREES C + 17.77)/0.5555556 DEGREES C = (DEGREES F - 32.0)*0.5555556		

From LOADS module LSMUP.FOR, subroutine UNITS.

```
BUILDING SUMMARY STATISTICS

BUILDING HEAT LOSS RATE:     AMBIENT =         BTU/H-F                   [W/C]
                                     =         BTU/H-F-(SF FLOOR AREA)   [W/C-SM]
                             GROUND  =         BTU/H-F                   [W/C]
                                     =         BTU/H-F-(SF FLOOR AREA)   [W/C-SM]

TOTAL BUILDING HEAT CAPACITY         =         BTU/F                     [KJ/C]

HEAT CAPACITY/AMBIENT LOSS RATIO     =         HOURS

ZONE ITERATION VIOLATION:    NUMBER  =         ERROR =

TROMBE ITERATION VIOLATION:  NUMBER  =         ERROR =
```

Variable	Description
BUILDING HEAT LOSS RATE	
AMBIENT	-Total steady-state heat loss rate for building calculated from inside air temperature to outside air temperature.
GROUND	-Total steady-state heat loss rate to ground node from inside air temperature.
TOTAL BUILDING HEAT CAPACITY	-Total heat capacity of building elements.
HEAT CAPACITY/ AMBIENT LOSS RATIO	-Ratio of heat capacity to AMBIENT loss rate.
ZONE ITERATION VIOLATION	-Reports the number of time increments for which the maximum number of iterations allowed was not adequate to achieve the specified convergence criteria for the zone air temperature, and the maximum temperature error due to non-convergence.
TROMBE ITERATION VIOLATION	-Same as above for the Trombe wall air gap temperature.

NOTE: The items on this page provide some miscellaneous information characterizing the building as a whole. In addition any failures of the numerical iteration algorithms for air temperatures are reported. The items on this page are not in the same format as the standard output sections.

AMBIENT SUMMARY (1 of 2)

```
----------------SOLAR RADIATION-----------  ---AMBIENT TEMPERATURE---

DIRECT   UNSHADED  DIRECT   DIFFUSE  TOTAL
NORMAL    HORIZ.   HORIZ.   HORIZ.   HORIZ.     MEAN   MIN   MAX   RANGE

KBTU/SF  KBTU/SF  KBTU/SF  KBTU/SF  KBTU/SF      F      F     F      F
 MJ/SM    MJ/SM    MJ/SM    MJ/SM    MJ/SM       C      C     C      C
_____________________________________________________________________
```

Variable	Description
SOLAR RADIATION	
DIRECT NORMAL	-Incident direct normal radiation.
UNSHADED HORIZ.	-Incident total horizontal before skyline shading.
DIRECT HORIZ.	-Incident direct horizontal radiation after skyline shading.
DIFFUSE HORIZ.	-Incident diffuse horizontal radiation after skyline shading.
TOTAL HORIZ.	-Incident total horizontal radiation after skyline shading.
AMBIENT TEMPERATURE	
MEAN	-Average hourly ambient temperature.
MIN	-Minimum hourly ambient temperature.
MAX	-Maximum hourly ambient temperature.
RANGE	-Average difference between daily maximum and minimum ambient temperatures.

AMBIENT SUMMARY (2 of 2)

```
---WIND SPEED---      -----HUMIDITY-----     MEAN      ----DEGREE DAYS-----
                                            GROUND
MEAN   MIN    MAX       DEWPOINT    RATIO     TEMP      HEATING      COOLING

MPH    MPH    MPH          F         NONE      F          FD           FD
M/S    M/S    M/S          C         NONE      C          CD           CD
```

Variable	Description
MEAN WIND SPEED	-Average hourly wind speed.
MIN WIND SPEED	-Minimum hourly wind speed.
MAX WIND SPEED	-Maximum hourly wind speed
DEWPOINT	-Average hourly dewpoint temperature.
HUMIDITY RATIO	-Average hourly humidity ratio.
MEAN GROUND TEMPERATURE	-Average hourly ground temperature.
HEATING DEGREE DAYS	-Degree days for heating.
COOLING DEGREE DAYS	-Degree days for cooling.

BUILDING SUMMARY (1 of 5)

```
----------------------------SOLAR RADIATION--------------------------

TRANS-     INTER-     INWARD       WINDOW      CAVITY      TOTAL
MITTED      ZONE      ABSORBED      LOSS        LOSS       GAIN

MBTU       MBTU        MBTU         MBTU        MBTU       MBTU
 GJ         GJ          GJ           GJ          GJ         GJ
---------------------------------------------------------------------
```

Variable	Description
TRANSMITTED	-Shortwave solar energy transmitted through all windows. Includes all transmitted.
INTERZONE	-Net transfer of shortwave with other zones through transparent walls.
INWARD ABSORBED	-Inward flowing portion of shortwave absorbed in glazing layers before shading coefficient.
WINDOW LOSS	-Reduction in solar heat gain due to user specified shading factors on windows.
CAVITY LOSS	-Reduction due to shortwave reflected back through windows specified by cavity absorptance.
TOTAL GAIN	-Total solar heat gain entering zone. This equals algebraic sum of the above variables.

BUILDING SUMMARY (2 of 5)

```
------------------------------HEAT FLOWS--------------------------------

WINDOWS   AMBIENT   GROUND   INFILTRATION   INTERZONE   INTERNAL GAINS

  MBTU      MBTU     MBTU        MBTU          MBTU          MBTU
   GJ        GJ       GJ          GJ            GJ            GJ
________________________________________________________________________
```

Variable	Description
WINDOWS	-Heat flow through all windows neglecting shortwave radiation effects.
AMBIENT	-Sum of all heat flows through walls and loss coefficients to ambient.
GROUND	-Sum of all heat flows through walls and loss coefficients to ground node.
INFILTRATION	-Sum of all heat flows due to infiltration.
INTERZONE	-Net heat flow to other zones through walls and loss coefficients.
INTERNAL GAINS	-Heat flow due to internal gains.

Note: All heat flows are reckoned positive inward. For accounting purposes the zone boundary is taken at the internal surfaces of walls and windows.

BUILDING SUMMARY (3 of 5)

```
-------------------------------HEAT FLOWS--------------------------------

ROCKBIN      ROCKBIN       FANS     TROMBE       INTERNAL       HEAT
ACTIVE       PASSIVE                WALLS        STORAGE        BALANCE

 MBTU         MBTU         MBTU     MBTU          MBTU           MBTU
  GJ           GJ           GJ       GJ            GJ             GJ

_________________________________________________________________________
```

Variable	Description
ROCKBIN ACTIVE	-Net controlled heat flow to zone from rockbins.
ROCKBIN PASSIVE	-Heat flow to zone due to passive losses from rockbin.
FANS	-Net heat flow to zone by fans.
TROMBE WALL	-Sum of heat flow by thermocirculation through Trombe wall vents and transfer from zone side of wall.
INTERNAL STORAGE	-Net heat flow from storage elements entirely within zone boundaries.
HEAT BALANCE	-Heat balance on zone over period of summary. This is equal to the sum of HEATFLOWS plus HEAT plus VENT plus COOL plus SOLAR GAIN.

Note: Heat flows reckoned positive inward. For accounting purposes, the zone boundary is taken at the interior surface of the walls and windows.

BUILDING SUMMARY (4 of 5)

```
------------------------------LATENT HEAT------------------------------------

INTERNAL    INFIL     FANS     VENT     TOTAL    ACLOAD     RH       HR

 MBTU       MBTU      MBTU     MBTU     MBTU      MBTU     NONE     NONE
  GJ         GJ        GJ       GJ       GJ        GJ      NONE     NONE

-----------------------------------------------------------------------------
```

Variable	Description
LATENT HEAT	
INTERNAL	-Internal latent loads; user specified.
INFIL	-Latent loads due to infiltration
FANS	-Latent loads due to moisture transfer by fans from other zones
VENT	-Latent loads due to venting.
TOTAL	-Total latent load.
ACLOAD	-Sum of sensible and latent cooling loads.
RH	-Mean hourly relative humidity in zone.
HR	-Mean hourly humidity ratio in zone.

Note: Relative humidity and humidity ratio are calculated each hour. The other latent output variables are calculated only for those time increments in which the cooling equipment operates. The latent summary is produced only when latent calculations are specified.

BUILDING SUMMARY (5 of 5)

```
-----EQUIP ENERGY-----     ----- MAXIMUM LOAD-----     SETPT FR.

HEAT     VENT     COOL     HEAT     VENT     COOL      HEAT   COOL

MBTU     MBTU     MBTU     KBTU/H   KBTU/H   KBTU/H    NONE   NONE
 GJ       GJ       GJ        KW       KW       KW      NONE   NONE
```

Variable	Description
EQUIP ENERGY HEAT VENT COOL	-Total sensible heat supplied or removed from conditioned space by heating, venting and cooling equipment.
MAXIMUM LOADS HEAT VENT COOL	-Maximum simultaneous hourly rate of sensible heat addition or removal required from equipment.
SETPT FR. HEAT COOL	-Fraction of hours air temperature is at the heating or cooling setpoints.

ZONE SUMMARY (1 of 6)

```
-----------------------------SOLAR RADIATION-------------------------

TRANS-     INTER-     INWARD       WINDOW       CAVITY       TOTAL
MITTED      ZONE      ABSORBED      LOSS         LOSS        GAIN

MBTU       MBTU        MBTU         MBTU         MBTU        MBTU
 GJ         GJ          GJ           GJ           GJ          GJ
____________________________________________________________________
```

Variable	Description
TRANSMITTED	-Shortwave solar energy transmitted through all windows. Includes all transmitted.
INTERZONE	-Net transfer of shortwave with other zones through transparent walls.
INWARD ABSORBED	-Inward flowing portion of shortwave absorbed in glazing layers before shading coefficient.
WINDOW LOSS	-Reduction in solar heat gain due to user specified shading factors on windows.
CAVITY LOSS	-Reduction due to shortwave reflected back through windows specified by cavity absorptance.
TOTAL GAIN	-Total solar heat gain entering zone. This equals algebraic sume of the above variables.

ZONE SUMMARY (2 of 6)

```
---------------------------------HEAT FLOWS----------------------------------

WINDOWS   AMBIENT   GROUND   INFILTRATION   INTERZONE   INTERNAL GAINS

  MBTU      MBTU     MBTU        MBTU          MBTU          MBTU
   GJ        GJ       GJ          GJ            GJ            GJ
______________________________________________________________________________
```

Variable	Description
WINDOWS	-Heat flow through all windows neglecting shortwave radiation effects.
AMBIENT	-Sum of all heat flows through walls and loss coefficients to ambient.
GROUND	-Sum of all heat flows through walls and loss coefficients to ground node.
INFILTRATION	-Sum of all heat flows due to infiltration.
INTERZONE	-Net heat flow to other zones through walls and loss coefficients.
INTERNAL GAINS	-Heat flow due to internal gains.

Note: All heat flows are reckoned positive inward. For accounting purposes the zone boundary is taken at the internal surfaces of walls and windows.

ZONE SUMMARY (3 of 6)

```
------------------------------HEAT FLOWS------------------------------

ROCKBIN      ROCKBIN                  TROMBE       INTERNAL       HEAT
ACTIVE       PASSIVE       FANS       WALLS        STORAGE        BALANCE

 MBTU         MBTU         MBTU       MBTU          MBTU           MBTU
  GJ           GJ           GJ         GJ            GJ             GJ
______________________________________________________________________
```

Variable	Description
ROCKBIN ACTIVE	-Net controlled heat flow to zone from rockbins.
ROCKBIN PASSIVE	-Heat flow to zone due to passive losses from rockbin.
FANS	-Net heat flow to zone by fans.
TROMBE WALL	-Sum of heat flow by thermocirculation through Trombe wall vents and transfer from zone side of wall.
INTERNAL STORAGE	-Net heat flow from storage elements entirely within zone boundaries.
HEAT BALANCE	-Heat balance on zone over period of summary. This is equal to the sum of HEATFLOWS plus HEAT plus VENT plus COOL plus SOLAR GAIN.

Note: Heat flows reckoned positive inward. For accounting purposes, the zone boundary is taken at the interior surface of the walls and windows.

ZONE SUMMARY (4 of 6)

```
------------------------------LATENT HEAT---------------------------------
INTERNAL    INFIL     FANS     VENT     TOTAL     ACLOAD     RH       HR

 MBTU       MBTU      MBTU     MBTU     MBTU      MBTU      NONE     NONE
  GJ         GJ        GJ       GJ       GJ        GJ       NONE     NONE
__________________________________________________________________________
```

Variable	Description
LATENT HEAT*	
INTERNAL	-Internal latent loads; user specified.
INFIL	-Latent loads due to infiltration
FANS	-Latent loads due to moisture transfer by fans from other zones
VENT	-Latent loads due to venting.
TOTAL	-Total latent load.
ACLOAD	-Sum of sensible and latent cooling loads.
RH	-Mean hourly relative humidity in zone.
HR	-Mean hourly humidity ratio in zone.

Note: Latent output variables are calculated only for those time increments in which the cooling equipment operates.

ZONE SUMMARY (5 of 6)

```
-----EQUIP ENERGY-----     ----- MAXIMUM LOAD-----   SETPT FR.

HEAT     VENT     COOL     HEAT     VENT     COOL     HEAT COOL

MBTU     MBTU     MBTU     KBTU/H   KBTU/H   KBTU/H   NONE NONE
 GJ       GJ       GJ        KW       KW       KW     NONE NONE
```

Variable	Description
EQUIP ENERGY HEAT VENT COOL	-Total sensible heat supplied or removed from conditioned space by heating, venting and cooling equipment.
MAXIMUM LOADS HEAT VENT COOL	-Maximum hourly rate of sensible heat addition or removal required from equipment.
SETPT FR. HEAT COOL	-Fraction of hours air temperature is at heating or cooling setpoints.

ZONE SUMMARY (6 of 6)

```
-FULL LOAD HOURS-   --MEAN DUTY CYCLE--       --AIR TEMPERATURE--

HEAT  VENT  COOL      HEAT  VENT  COOL        MEAN   MIN   MAX   RANGE

 H     H     H        NONE  NONE  NONE         F      F     F      F
 H     H     H        NONE  NONE  NONE         C      C     C      C
```

Variable	Description
FULL LOAD HOURS HEAT VENT COOL	-Total hours of operation of equipment at full capacity. May be used to determine operating energy for equipment. Undefined if equipment capacity is not specified.
MEAN DUTY CYCLE HEAT VENT COOL	-Average ratio for period of summary of the actual rate of energy delivery to the maximum capacity rate of delivery in each time increment in which the equipment operates. Undefined if equipment capacity is not entered.
AIR TEMPERATURE MEAN MIN MAX RANGE	-Mean, minimum, and maximum hourly zone air temperatures over period of summary. RANGE is the average temperature difference between daily maximums and minimums.

WINDOW SUMMARY (1 of 3)

```
---UNSHADED INCIDENT----     FRACTION SUNLIT      ---SHADED INCIDENT----
DIRECT   DIFFUSE   TOTAL     DIR  DIF  TOT        DIRECT  DIFFUSE  TOTAL

 MBTU     MBTU     MBTU      NONE NONE NONE        MBTU    MBTU    MBTU
  GJ       GJ       GJ       NONE NONE NONE         GJ      GJ      GJ
```

Variable	Description
UNSHADED INCIDENT DIRECT DIFFUSE TOTAL	-Incident solar radiation on glazing which would occur in the absence of shading effects. Shading here refers only to overhangs and sideins; skyline losses have already been removed.
FRACTION SUNLIT DIRECT DIFFUSE TOTAL	-Fraction of direct, diffuse, and total radiation incident on glazing after shading by overhangs and sidefins.
SHADED INCIDENT DIRECT DIFFUSE TOTAL	-Actual incident solar radiation.

WINDOW SUMMARY (2 of 3)

```
------TRANSMITTED-------     TRANSMISSIVITY      POTENTIAL     ACTUAL
DIRECT   DIFFUSE   TOTAL     DIR   DIF   TOT     HEAT GAIN    HEAT GAIN

 MBTU     MBTU     MBTU      NONE  NONE  NONE      MBTU         MBTU
  GJ       GJ       GJ       NONE  NONE  NONE      MBTU         MBTU
---------------------------------------------------------------------
```

Variable	Description
TRANSMITTED DIRECT DIFFUSE TOTAL	-Shortwave solar radiation transmitted through glazing.
TRNAMISSIVITY DIRECT DIFFUSE TOTAL	-Fraction of actual incident shortwave solar radiation transmitted through glazing.
POTENTIAL HEAT GAIN	-Transmitted shortwave solar radiation plus inward flowing portion of shortwave absorbed in glazing.
ACTUAL HEAT GAIN	-Potential solar heat gain adjusted by user-specified shading coefficient.

WINDOW SUMMARY (3 of 3)

```
HEAT LOSS      ----INNER GLASS TEMPERATURE---       LOSS          ADJUSTED
TO AMBIENT     MEAN     MIN     MAX     RANGE      U-VALUE        U-VALUE

  MBTU          F        F       F        F       BTU/SF-H-F     BTU/SF-H-F
   GJ           C        C       C        C        W/SM-C         W/SM-C
```

Variable	Description
HEAT LOSS TO AMBIENT	-Heat loss to ambient excluding all shortwave solar effects.
INNER GLASS TEMPERATURE MEAN MIN MAX RANGE	-Temperature of inner glazing layer. Mean hourly, maximum hourly, minimum hourly, and average daily range between minimum and maximum.
LOSS U-VALUE	-Effective U-value ignoring shortwave solar effects. Calculated as heat loss to ambient divided by mean zone to ambient temperature difference.
ADJUSTED U-VALUE	-Effective U-value adjusted for solar absorbed in glazing. Calculated as heat loss to ambient minus inward flowing absorbed radiation divided by mean zone to ambient temperature difference.

Variable	Description
UNSHADED INCIDENT DIRECT DIFFUSE TOTAL	-Solar radiation that would be incident on the surface if no shading was present. This is calculated after skyline losses are taken into account.
FRACTION SUNLIT DIRECT DIFFUSE TOTAL	-Fraction of the direct, diffuse, and total radiation that strikes the surface after all shading calculations.
SHADED INCIDENT DIRECT DIFFUSE TOTAL	-Solar radiation actually incident on the surface.

NOTE: All radiation values for exterior surfaces are expressed per square foot or per square meter.

```
                        WALL SUMMARY (1 of 1)


----SURFACE TEMPERATURES----   --SOLAR ABSORBED-- --HEAT FLOW--     SOL-
   FRT/INT        BCK/EXT       FRT/INT BCK/EXT   FRT/INT BCK/EXT   AIR
MEAN   RANGE   MEAN   RANGE                                         TEMP

  F      F       F      F        MBTU    MBTU      MBTU    MBTU      F
  C      C       C      C         GJ      GJ        GJ      GJ       C
```

Variable	Description
SURFACE TEMPERATURES	-Mean hourly temperatures and mean daily difference between maximum and minimum hourly temperatures for each side of wall.
MEAN RANGE	
SOLAR ABSORBED	-Total solar radiation absorbed at each surface of wall.
FRT/INT BCK/EXT	
HEAT FLOW	-Net heat flow at each side of wall taken positive inward. The sum of these equals the net storage of heat in the wall.
FRT/INT BCK/EXT	
SOL-AIR TEMP	-The mean sol-air temperature. Output only for exterior walls.

Note: Front and Back refer to the headings in the input for a given wall. "Front" must face a user-named zone; while "back" may face either a user-named zone, a user-defined exterior surface, or the predefined zones, "ambient" and "ground".

FAN SUMMARY (1 of 1)

HEAT FLOW	FULL LOAD HOURS	DUTY CYCLE	TEMPERATURE INLET	TEMPERATURE OUTLET
MBTU	H	NONE	F	F
GJ	H	NONE	C	C

Variable	Description
HEAT FLOW	-Total sensible heat flow through fan.
FULL LOAD HOURS	-Number of hours of actual operation of fan. May be used to estimate fan operating energy.
DUTY CYCLE	-Fraction of time fan operates averaged over all time increments in which fan operates.
TEMPERATURE INLET OUTLET	-Temperatures on inlet and outlet side averaged over all time increments in which fan operates.

ROCKBIN SUMMARY (1 of 2)

```
----HEAT FLOW----    ---PASSIVE LOSSES---      NET      ENERGY
CHARGE  DISCHARGE    AMBIENT ZONES GROUND     STORED    BALANCE

 MBTU     MBTU        MBTU    MBTU   MBTU      MBTU      MBTU
  GJ       GJ          GJ      GJ     GJ        GJ        GJ
```

Variable	Description
HEAT FLOW CHARGE DISCHARGE	-Heat flow into or out of rockbin during charge and discharge
PASSIVE LOSSES AMBIENT ZONES GROUND	-Total heat losses from rockbin to ambient, zones, and ground nodes.
NET STORED	-Net heat stored in rockbin during period of summary.
ENERGY BALANCE	-Energy balance on rockbin calculated as charge energy minus discharge energy minus passive losses minus net stored.

ROCKBIN SUMMARY (2 of 2)

```
--------------TEMPERATURES--------------  FULL LOAD HOURS   --DUTY CYCLE----
-----CHARGE------  ---DISCHARGE----  AVG  CHARGE DISCHARGE CHARGE DISCHARGE
OUTLET ZONE  DIFF  OUTLET ZONE DIFF  NODE
  F     F     F      F     F    F     F      H        H      NONE   NONE
  C     C     C      C     C    C     C      H        H      NONE   NONE
```

Variable	Description
TEMPERATURES	
CHARGE INLET	-Mean hourly temperature of node at charge end of rockbin.
CHARGE OUTLET	-Mean hourly temperature of node at discharge end of rockbin.
CHARGE ZONE	*-Mean hourly temperature of charge zone air.
CHARGE DIFF.	*-Mean temperature difference between charge zone air and inlet node of rockbin.
DISCHARGE ZONE	*-Mean hourly temperature of discharge zone air.
DISCHARGE DIFF.	*-Mean temperature difference between discharge zone air and outlet node of rockbin.
AVG BIN	-Average temperature of rockbin.
FULL LOAD HOURS CHARGE DISCHARGE	-Total full load hours of operation for charge and discharge fans. Can be used to calculate fan operating energy.
DUTY CYCLE CHARGE DISCHARGE	-Mean duty cycle for charge and discharge fans. Calculated as the ratio of actual operating time to length of time increment for each time increment in which the fan operates.

*Note: These are calculated only during time increments in which the rockbin operates.

TROMBE WALL SUMMARY (1 of 5)

```
---UNSHADED INCIDENT----   FRACTION SUNLIT    ---SHADED INCIDENT----
DIRECT  DIFFUSE  TOTAL     DIR  DIF  TOT      DIRECT  DIFFUSE  TOTAL

 MBTU    MBTU     MBTU     NONE NONE NONE      MBTU    MBTU     MBTU
  GJ      GJ       GJ      NONE NONE NONE       GJ      GJ       GJ
```

Variable	Description
UNSHADED INCIDENT DIRECT DIFFUSE TOTAL	-Incident solar radiation on glazing which would occur in the absence of shading effects. Shading here refers only to overhangs and sidefins; skyline losses have already been removed.
FRACTION SUNLIT DIRECT DIFFUSE TOTAL	-Fraction of direct, diffuse, and total radiation incident on glazing after shading by overhangs and sidefins.
SHADED INCIDENT DIRECT DIFFUSE TOTAL	-Actual incident solar radiation.

TROMBE WALL SUMMARY (2 of 5)

```
------TRANSMITTED-------     TRANSMISSIVITY    POTENTIAL    ACTUAL
DIRECT   DIFFUSE   TOTAL     DIR  DIF  TOT     HEAT GAIN    HEAT GAIN

 MBTU     MBTU     MBTU      NONE NONE NONE      MBTU         MBTU
  GJ       GJ       GJ       NONE NONE NONE      MBTU         MBTU
```

Variable	Description
TRANSMITTED DIRECT DIFFUSE TOTAL	-Shortwave solar radiation transmitted through glazing
TRNASMISSIVITY DIRECT DIFFUSE TOTAL	-Fraction of actual incident shortwave solar radiation transmitted through glazing.
POTENTIAL HEAT GAIN	-Transmitted shortwave solar radiation plus inward flowing portion of shortwave absorbed in glazing.
ACTUAL HEAT GAIN	-Potential solar heat gain adjusted by user-specified shading coefficient.

TROMBE WALL SUMMARY (3 of 5)

```
HEAT LOSS     -----INNER GLASS TEMPERATURE---          LOSS          ADJUSTED
TO AMBIENT     MEAN      MIN      MAX     RANGE       U-VALUE        U-VALUE

  MBTU          F         F        F        F       BTU/SF-H-F     BTU/SF-H-F
   GJ           C         C        C        C        W/SM-C          W/SM-C
```

Variable	Description
HEAT LOSS TO AMBIENT	-Heat loss to ambient excluding all shortwave solar effects.
INNER GLASS TEMPERATURE MEAN MIN MAX RANGE	-Temperature of inner glazing layer. Mean hourly, maximum hourly, minimum hourly, and average daily range between minimum and maximum.
LOSS U-VALUE	-Effective U-value ignoring shortwave solar effects. Calculated as heat loss to ambient divided by mean zone to ambient temperature difference.
ADJUSTED U-VALUE	-Effective U-value adjusted for solar absorbed in glazing. Calculated as heat loss to ambient minus inward flowing absorbed radiation divided by mean zone to ambient temperature difference.

TROMBE WALL SUMMARY (4 of 5)

```
----SURFACE TEMPERATURES----   --SOLAR ABSORBED--  --HEAT FLOW--
   FRONT            BACK
MEAN   RANGE    MEAN   RANGE      FRONT    BACK        FRONT  BACK

 F       F        F      F        MBTU     MBTU        MBTU   MBTU
 C       C        C      C         GJ       GJ          GJ     GJ
```

Variable	Description
SURFACE TEMPERATURES	-Mean hourly temperatures and mean daily difference between maximum and minimum hourly temperatures for each side of wall.
MEAN	
RANGE	
SOLAR ABSORBED	-Total solar radiation absorbed at each surface of wall
FRONT	
BACK	
HEAT FLOW	-Net heat flow at each side of wall taken positive inward. The sum of these equals the net storage of heat in the wall.
FRONT	
BACK	

Note: Front and Back refer to the headings in the input for a given wall. "Front" must face a user-named zone; while "back" may face either a user-named zone, a user-defined exterior surface, or the predefined zones, "ambient" and "ground".

TROMBE WALL SUMMARY (5 of 5)

```
---AIR GAP TEMPERATURE--     THERMO-    TOTAL     FLOW     HOURS    TOTAL
MEAN   MIN    MAX   RANGE    SIPHON     GAIN      RATE    OF FLOW   FLOW

 F      F      F      F       MBTU      MBTU      CFM        H       MCF
 C      C      C      C        GJ        GJ       CMH        H       MCM
```

Variable	Description
AIR GAP TEMPERATURE MEAN MIN MAX RANGE	-Air temperature in gap between outer face of Trombe wall and the inner glazing layer. Gives mean hourly, minimum hour, maximum hour, and mean daily range between maximum and minimum.
THERMOSIPHON	-Heat flow through thermocirculation of Trombe wall vents.
TOTAL GAIN	-Sum of thermocirculation transfer and transfer from zone side of Trombe wall.
FLOW RATE	-Average hourly air flow rate through vents for the hours in which flow occurs.
HOURS OF FLOW	-Number of hours in which thermocirculation occurs.
TOTAL FLOW	-Total volume of air moved by thermocirculation.

INPUT FORM ONE

RUNS

RUN LABEL	STATION NAME	GROUND REFL. [FRAC]	GROUND TEMP. [F]	-START- MON DAY [DATE]	-STOP- MON DAY [DATE]	SKYLINE PROFILE	PAR. TYPE
AAAAAAAAAAAAAAAA	AAAAAAAAAA	S.SSSS	SSS.SS	AAA XX.	AAA XX.	AAAAAA	AAAAAA

ZONES

ZONE NAME	HVAC TYPE	FLOOR AREA [SF]	HGT [FT]	INFIL. RATE [AC/H]	SOLAR TO AIR [FRAC]	SOLAR LOST [FRAC]	INTERNAL GAIN [KBTU/H]	LATENT GAIN [KBTU/H]
AAAAAAAAAA	AAAAAAAAAA	XXXXX.X	XX.X	SSS.SSS	X.XXX	X.XXX	SSSS.SSS	SSSS.SSS

INTERZONE

SOURCE ZONE	SINK ZONE OR AMBIENT OR GROUND	CONDUCTANCE COEF. [BTU/H-F]	SOLAR TRANSFER [FRAC]	REVERSE TRANSFER [FRAC]
AAAAAAAAAA	AAAAAAAAAA	XXXXX.XX	S.SSSSS	S.SSSSS

WINDOWS

INTERIOR ZONE	EXTERIOR SURFACE	GLAZING TYPE	HEIGHT	LENGTH	---LOCATION----	
			[FT]	[FT]	HORIZ [FT]	VERT [FT]
AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	XXXX.XX	XXXX.XX	XXXX.XX	XXXX.XX

INPUT FORM TWO

WALLS

	--FRONT/INTERIOR SIDE---			---BACK/EXTERIOR SIDE---			
WALL TYPE	ZONE NAME	SURF COEF [BTU/F -SF-H]	SOLAR COEF [FRAC]	ZONE OR SURFACE, AMBIENT, GROUND	SURF COEF [BTU/F -SF-H]	SOLAR COEF [FRAC]	WALL AREA [SF]
AAAAAAAAAA	AAAAAAAAAA	XX.XXX	X.XXXX	AAAAAAAAAA	XX.XXX	X.XXXX	XXXXX.X

TROMBE.WALLS

							--LOCATION--	
INTERIOR ZONE	EXTERIOR SURFACE	TROMBE TYPE	SURF COEF [BTU/F -SF-H]	SOLAR COEF [FRAC]	HEIGHT [FT]	LENGTH [FT]	HORIZ. [FT]	VERT. [FT]
AAAAAAAAAA	AAAAAAAAAA	AAAAAA	XX.XXX	X.XXXX	XX.XX	XXX.XX	XXX.XX	XXX.XX

FANS

SINK ZONE	SOURCE ZONE	FAN TYPE	OFF SEASON
AAAAAAAAAA	AAAAAAAAAA	AAAAAAAA	AAAAAAAA

ROCKBINS

SINK ZONE	SOURCE ZONE	ROCKBIN TYPE	ZONE FOR PASSIVE LOSSES	PASSIVE LOSS TO ZONE [BTU/H-F]	PASSIVE LOSS TO AMBIENT [BTU/H-F]	PASSIVE LOSS TO GROUND [BTU/H-F]
AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	XXXX.XXX	XXXX.XXX	XXXX.XXX

INPUT FORM THREE

SURFACES

EXTERIOR SURFACE	COMPASS AZIMUTH [DEG]	TILT [DEG]	HEIGHT [FT]	LENGTH [FT]	OVERHANG TYPE	LEFT SIDEFIN	RIGHT SIDEFIN
AAAAAAAAAA	XXX.X	XX.X	XXXX.XX	XXXX.XX	AAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA

HVAC.TYPES

HVAC TYPE	HEATING SETPOINT [F]	VENTING SETPOINT [F]	COOLING SETPOINT [F]	HEATING CAPACITY [KBTU/H]	VENTING CAPACITY [AC/H]	COOLING CAPACITY [KBTU/H]	COOLER COIL [F]
AAAAAAAAAA	SSS.SSS	SSS.SSS	SSS.SSS	XXXX.XXX	XXX.XX	XXXX.XXX	XX.X

TROMBE.TYPES

TROMBE TYPE	WALL TYPE	GLAZING TYPE	VENT OVER HEAT? [Y/N]	SURF COEF [BTU/F -SF-H]	WALL EXT. ABS. [FRAC]	VENT AREA RATIO [FRAC]	HEIGHT BETWEEN VENTS [FT]	VENT COEF [FRAC]
AAAAAA	AAAAAAAAAA	AAAAAAAAAA	A	XX.XXX	X.XXX	X.XXXX	XX.XXX	X.XXX

WALL.TYPES

WALL TYPE	LAYER # 1	LAYER # 2	LAYER # 3	LAYER # 4	LAYER # 5	LAYER # 6
AAAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA	AAAAAAAAA

INPUT FORM FOUR

MASS.TYPES

MASS TYPE	CONDUCTIVITY [BTU/FT-F-H]	DENSITY [LB/CF]	SPECIFIC HEAT [BTU/LB-F]	THICKNESS [FT]	NODES
AAAAAAAAAA	X.XXXX	XXXX.XXX	X.XXXX	XX.XXXX	XX.

PCM.TYPES

PCM TYPE	CONDUCTIVITY [BTU/FT-F-H]	DENSITY [LB/CF]	SPECIFIC HEAT [BTU/LB-F]	THICKNESS [FT]	HEAT OF FUSION [BTU/LB]	MELTING POINT [F]
AAAAAAAAAA	X.XXXX	XXXX.XXX	X.XXXX	XX.XXXX	XXXX.XX	XXX.X

GLAZING.TYPES

GLAZING TYPE	GLAZING U VALUE [BTU/F-SF-H]	SHADING COEF. [FRAC]	EXTINCTION COEF. [1/IN]	INDEX OF REFRACTION [NONE]	THICKNESS OF LAYER [IN]	NUMBER OF LAYERS
AAAAAAAAAA	SS.SSSSS	SS.SSSSS	X.XXXX	X.XXXX	X.XXXX	XX.

BIN.TYPES

ROCKBIN TYPE	BI-DIR [Y/N]	LENGTH [FT]	CROSS-SECTION AREA [SF]	HEAT CAPACITY [BTU/CF-F]	AXIAL COND. [BTU/F-HR-FT]	CHARGE FAN TYPE	DISCHARGE FAN TYPE	CHARGE OFF SEASON
AAAAAAAAAA	A	XXX.XX	XXX.XX	XXXX.XXX	XX.XXX	AAAAAAAA	AAAAAAAA	AAAAAAAA

INPUT FORM FIVE

FAN.TYPES

FAN TYPE	FAN CAPACITY [CFM]	MINIMUM TEMP. DIFF. [F]
AAAAAAAA	XXXXX.X	XX.XX

OVERHANG.TYPES

OVERHANG TYPE	VERTICAL OFFSET [FT]	HORIZONTAL PROJECTION [FT]
AAAAAAAAAA	XX.XX	XX.XX

SIDEFIN.TYPES

SIDEFIN TYPE	OFFSET TO SIDE [FT]	LENGTH OF PROJECTION [FT]
AAAAAAAAAA	XX.XX	XX.XX

SKYLINE.TYPES

SKYLINE	--------- ALTITUDE ANGLE OF SKYLINE [DEGREES] --------										
PROFILE	---EAST---										---WEST---
NAME	100	80	60	40	20	0	20	40	60	80	100
AAAAAA	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X	XX.X

INPUT FORM SIX

OUTPUT

OUTPUT TYPE	TIME PERIOD [H/D/M]	UNITS [E/M]	OUTPUT SEASON	BUILDING ELEMENT	OUTPUT SECTION	FORMAT? [Y/N]
AAAAAAAA	A	A	AAAAAAAA	XXXX.	XXXX.	A

SCHEDULES

SCHEDULE	SEASON	HR	VALUE	HR	VALUE	HR	VALUE	HR	VALUE
AAAAAAAA	AAAAAAAA	XX.	XXXX.XXX	XX.	XXXX.XXX	XX.	XXXX.XXX	XX.	XXXX.XXX

SEASONS

SEASON NAME	START DATE MON DAY [DATE]	STOP DATE MON DAY [DATE]	DAY OF WEEK [ALL/M-F/S-S]
AAAAAAAA	AAA XX.	AAA XX	AAA

INPUT FORM SEVEN

PARAMETERS

PAR. TYPE	INFIL. MULT. [FRAC]	MAX ZONE	ZONE CRIT [F]	MAX TROM	TROM CRIT [F]	INIT TEMP [F]	H DD BASE [F]	C DD BASE [F]	WEEK DAY	DIF. ANG. [DEG]
AAAAAA	XX.XX	XXX.	XX.XXX	XXX.	XX.XXX	XX.X	XX.X	XX.X	X.	XX.X

STATIONS

STATION NAME	LAT. [DEG]	LONG. [DEG]	ELEV. [FT]	FILENAME	DATA TYPE	UNITS [E/M]	-START-		-STOP--	
							MON	DAY	MON	DAY
AAAAAAAAAA	XX.XX	XXX.X	XXXXX.	AAAAAAAAAA	XX.	A	AAA	XX.	AAA	XX.

APPENDIX D: OPTIONAL LOADS PROGRAM OUTPUT FILE

This optional output file from the loads program is written as a formatted (ASCII), sequential access file. The contents and order of this file are determined by the user inputs under the OUTPUTS data section. The file is written as a set of logical records, where each logical record gives the numeric values corresponding to one of the lines of output that may be generated within the standard loads output file. Each logical record is comprised of two or more physical records. The first physical record contains information defining the type of output; the following physical records contain the numeric output values.

The first physical record of each logical record may be read by the following code fragment:

```
      READ (LU,900) (MONTH(I),I=1,3),IDAY,IHOUR,ITYPE,INDEX,IPAGE,IUNIT,NDATA
900   FORMAT (2X,3A1,2(1X,I2),5(1X,I3)
```

where

MONTH(I),I=1,3	gives	the month of output as one of the three-letter codes: JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC,TOT
IDAY	gives	the day of month (1 to 31) of output (written as blanks if monthly level output follows)
IHOUR	gives	the hour of day (1 to 24) of output (written as blanks if monthly level or daily level output follows)
ITYPE	gives	the type of output summary which follows, as: =1 AMBIENT SUMMARY =2 BUILDING SUMMARY =3 ZONES SUMMARY =4 WINDOW SUMMARY =5 SURFACES SUMMARY =6 WALL SUMMARY =7 FAN SUMMARY =8 ROCKBIN SUMMARY =9 TROMBE SUMMARY
INDEX	gives	the component index for which output follows (e.g. ITYPE=3, INDEX=2 for the second zone summary)
IPAGE	gives	the "page" number within each output type for which output follows (e.g. ITYPE=3, INDEX=2, IPAGE=4 for the latent heat page of the zones summary for the second zone)
IUNIT	gives	the units system of output as -1 for English and 1 for metric
NDATA	gives	the number of data values which follow on subsequent physical records (NOTE: IFIX((NDATA + 9)/10) gives the number of physical records of data which follow)

INPUT FORM SEVEN
(metric)

PARAMETERS

PAR. TYPE	INFIL. MULT. [FRAC]	MAX ZONE	ZONE CRIT [C]	MAX TROM	TROM CRIT [C]	INIT TEMP [C]	H DD BASE [C]	C DD BASE [C]	WEEK DAY	DIF. ANG. [DEG]
AAAAAA	XX.XX	XXX.	XX.XXX	XXX.	XX.XXX	XX.X	XX.X	XX.X	X.	XX.X

STATIONS

STATION NAME	LAT. [DEG]	LONG. [DEG]	ELEV. [M]	FILENAME	DATA TYPE	UNITS [E/M]	-START-		-STOP--	
							MON	DAY	MON	DAY
AAAAAAAAAA	XX.XX	XXX.X	XXXXX.	AAAAAAAAAA	XX.	A	AAA	XX.	AAA	XX.

Each following record in the logical record contains from 1 to 10 real values, and may be read by the following code fragment:

```
                  READ (LU,901) (DATA(I),I=1,10)
            901   FORMAT (10(1X,E12.6))
```

The order of values in the record and their units are exactly the same as for the corresponding output line in the standard loads program output file.