

A DIGITAL COMPUTER PROGRAM TO CALCULATE
A RESIDENTIAL COOLING LOAD)

by

DANA MILAN SMERCHEK
B.S., Kansas State University, 1973

A THESIS

submitted in partial fulfillment of the
requirements for the degree

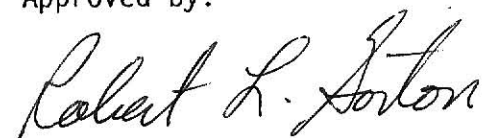
MASTER OF SCIENCE

Department of Mechanical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1975

Approved by:


Major Professor

LD
2668
T4
1975
S63
c.2

ii

TABLE OF CONTENTS

Chapter	Page
I. INTRODUCTION	1
Problem	1
Purpose	1
II. RESIDENTIAL COOLING LOAD	5
Conduction	5
Solar Radiation Heat Gains	6
Infiltration Heat Gains	6
Internal Heat Gains	7
Load Summary	7
III. PROGRAM CONSTRUCTION	
Input Data	8
Solar Heat Gains	14
Sol-Air Temperatures	16
Shading	16
Wall Heat Gains	21
Ceiling Heat Gains	23
Window Heat Gains	24
Door Heat Gains	25
Infiltration Loads	26
Internal Loads	27
Cooling Load Calculations	29
IV. RESULTS AND DISCUSSION	31
V. SUMMARY	51
BIBLIOGRAPHY	54
APPENDIX A	55
APPENDIX B	63
APPENDIX C	68
APPENDIX D	79
APPENDIX E	149

LIST OF TABLES

Table	Page
1. Peak Temperature Minus the Percentage of Daily Range	9
2. Orientation Numbers	13
3. Hourly Cooling Loads (BTU/hr) and Daily Cooling Load Totals (BTU/day)	46
4. Description of Data Cards	76
5. Cooling Load for Case IV by TETD Method	150

LIST OF FIGURES

Figure	Page
1. I Shaped House	11
2. L Shaped House	12
3. T Shaped House	18
4. U Shaped House	19
5. X Shaped House	20
6. Two Double Hung Windows	26
7. Two Awning Windows	26
8. Daily Outside Temperature Profile	32
9. Front View of Test House	33
10. Back View of Test House	34
11. First Floor Test House	35
12. Second Floor Test House	36
13. Cooling Load Profile Case One	40
14. Cooling Load Profile Case Two	41
15. Cooling Load Profile Case Three	42
16. Cooling Load Profile Case Four	43
17. Cooling Load Profile Case Five	44
18. Cooling Load Profile Case Six	45
19. Cooling Load Profiles for all Six Cases	47
20. House Used for Example One	56
21. Edge View of Wall	56
22. House and Shading for Example Two	57
23. Edge View Example Two	58
24. Example Two Areas X and Y	59

Figure	Page
25. Shading Example Three	60
26. View of Window in Example Four	61
27. Window Enlargement	61
28. Ceiling Cross Section	64
29. Top View of House for Sample Data	69
30. Side View of Walls 3 and 4	70
31. Side View of Walls 2 and 6	71
32. Side View of Wall 5	72
33. Side View of Wall 4	73
34. Data Cards 1-13	74
35. Data Cards 14-27	75

CHAPTER I

INTRODUCTION

Problem

In the past, home designers have generally sized residential air conditioning units by a peak load calculation or by a number of "rule of thumb" methods. Any of these methods were sufficiently accurate for the size of units sold and for the cost and availability of energy at the time. With the long-building energy crisis coming into the forefront, these methods are not adequate for future use. It is necessary for the home designer to look at a twenty-four hour cooling load profile so that construction or design of the home can be modified to minimize the total air conditioning load. This twenty-four hour cooling load profile can be obtained from a digital computer program that allows the designer to vary wall, window, and ceiling constructions; width of roof overhand; and the orientation of the home with respect to its lot.

A number of large air conditioning companies have already created such computer programs to calculate twenty-four hour cooling load profiles which use local five year weather tapes as their input information. As a result of using weather tapes, the cost of these programs is prohibitive except for calculating loads on large office buildings where the initial and operating costs are extremely large. Since the initial and operating costs for a single residential dwelling are relatively small as compared to those of a large office building, the cost of calculating a twenty-four hour cooling load for a residential home should be correspondingly low.

Purpose

The purpose of this thesis was to write and check by use on a typical

residence, a transient cooling load program for residential type construction. Architects and home designers have been asked more and more recently to minimize a home's air conditioning load with a limited amount of money to be spent on building materials. As a result, this program is aimed at calculating a twenty-four hour cooling load profile, which allows testing building material options.

With the ability of the home designer to look at a twenty-four hour cooling load, the decision on the size of the air conditioning unit is greatly aided. Larger than necessary units need no longer be installed just to make sure that the home is adequately cooled. The wasteful days of unit size overkill can be over. The home owner will no longer be forced to blindly have a larger than necessary unit installed that will run up initial and operating costs.

The designer can use this program for much more than just sizing the final air conditioning unit installed. It can be used to show that by spending more money on building materials such as insulation, storm windows, or brick veneering, that the peak load can be lowered and the overall total load can be spread out more over the entire day. Due to spreading out the load, the cooling load at five o'clock in the morning might be raised, and the peak cooling load at three o'clock in the afternoon might be lowered. This would allow a smaller unit to be installed and operated longer at peak capacity. These construction material changes are the types of changes owners of existing homes might possibly want to make to cut their monthly air conditioning costs. Many home owners are very interested in how they can spend a small sum of money on their home and cut their overall cooling expenses the most. The information from this program can help existing home owners as well as the future home owner.

In addition to construction material changes, the new home designers are interested in physical design changes of the house that may lower the total cooling load even further. Most important of these possible changes are the width of roof overhand and the angle of orientation of the house as it sets on its lot. It is obvious that widening the overhangs can decrease the air conditioning load, but there will be a point at which the designer must recognize that it costs more to build a larger overhang than the overhang will save in cooling load. Not so obvious is the angle of orientation that produces the minimum load. The shadowing of a surface that occurs due to adjacent parts of the house is not always easy to visualize, but it is easily computed in this program. To find the best angle is a trial and error type process. The designer in many cases may not have the chance to use this capability because the house may need to set a certain direction on the lot.

The commercial programs, Trace (Trane Air Conditioning Company) and Ecube (American Gas Association), are two examples of very sophisticated programs. They are very accurate due to the weather data information used as input for each specific location. For large buildings the accuracy of this weather information is very important. A small error in weather information on a large office building could mean the difference between the building being under air conditioned to having an oversized unit installed. The weather information, taken from five-year weather tapes to obtain this accuracy, is unfortunately very expensive to run on the computer. There is nothing wrong with this for large office buildings, for a small mistake in the cooling load could cost much more than the cost of one computer run. On the other hand, an expensive program like this would be unwarranted for a small residential home. Some commercial programs are not applicable to small buildings or residential homes. The special shading routine this program

contains for residential homes gives this program the ability to calculate a twenty-four hour cooling load profile for residential homes.

CHAPTER II

RESIDENTIAL COOLING LOAD

Before a cooling load for a residential home can be calculated, the individual components must be known. These are:

1. Conduction heat gains through solid surfaces.
2. Solar radiation through transparent surfaces.
3. Infiltration heat gains.
4. Internal heat gains.

Conduction

The conduction heat gains are contributed from the following four surfaces: exterior walls, windows, doors and the ceiling. The heat gain contributed by floors is not considered. Floors made of a concrete slab or with an unimproved basement below, could have a negative temperature differential across it which would indicate a heat loss rather than a heat gain. If this temperature differential is not negative, then its magnitude will be small; therefore, the floor heat gain will have a small influence on the final cooling load of the house.

Delays are seen in the interior space heat gain from the walls and roof due to the heat capacity of the heavy materials used in their construction. Even the air in the attic stores heat and can cause a delay. The magnitude and length of delay in the heat gain through walls and roofs can be changed by altering the amount of insulation or by adding a brick facing. Heavier materials such as brick facings should reduce the magnitudinal difference between the daily peak and minimum cooling loads. This could give the heat gain from these surfaces a much smoother profile. Windows and doors are constructed of lighter materials, so interior heat gains through them are

considered instantaneous. These conduction heat gains are computed as a function of four variables:

1. The intensity of the solar heat gain at the outside of the surface.
2. The outside dry bulb temperature.
3. The inside dry bulb temperature.
4. The thermal characteristics of the construction materials.

Solar Radiation Heat Gains

Each glass window in the home allows a contribution to the total solar radiation heat gain through the transparent surfaces. The solar radiation heat gain is a function of the intensity of the solar heat gain at the surface and the radiation transmitting properties of the window. These are seen as instantaneous interior space heat gains. The solar radiation heat gain can be reduced by using storm windows or draperies over the window.

Infiltration Heat Gains

The air leakage around each door and window in the home is combined to obtain the infiltration load. The infiltration load consists of two parts, a sensible and a latent. Depending on how humid the climate is, the latent infiltration load can range from a negligible amount to an amount equaling that of the sensible infiltration. For this reason, both sensible and latent infiltration must be considered. Infiltration load is a function of:

1. Outside dry bulb temperature.
2. Inside dry bulb temperature.
3. Outside humidity ratio.
4. Inside humidity ratio.
5. Width of the crack.
6. Length of the crack.
7. Speed of the wind outside.

Two ways of reducing this load are to use storm sashes and weatherstripping on all doors and windows.

Internal Heat Gains

The internal heat gain components are the people load (sensible and latent heat gain from each occupant) and the heat released by appliances and lights being operated in the home. This component is very difficult to predict for all families. To illustrate this difficulty, consider the people load. This load is a function of the number of family members, their ages and their activity level.

Load Summary

All of the cooling load components, except the internal heat gains, are functions of the environmental conditions exterior and interior to the house and the thermal characteristic of the construction materials. The internal loads will need a schedule for each load desired. To calculate the cooling load components by any method, a set of common variables must be specified. These include:

1. A description of the environment exterior and interior to the house.
2. A physical description of the house.
3. A listing of thermal characteristics for the construction materials.
4. A listing of window characteristics.

CHAPTER III

PROGRAM CONSTRUCTION

For the home designer who is going to use this program, a more detailed discussion of the program is needed. With a thorough understanding of the construction and assumptions used in the program, the user will be able to use this program to its best advantage. A copy of this program in its entirety is contained in the appendix.

The program is made up of many detailed subroutines, each having a very specific purpose in the program. Although the program needs all of these subroutines to calculate a final cooling load, the heart of this program is the response factor method of heat gains and cooling loads. (1)

Input Data

The input data is very important, for the output will be no better than the input that is supplied to the program. To run the program, all of the necessary input data must be supplied to the program. The required weather input is:

1. Month of year.
2. Latitude location of the house.
3. A twenty-four hour outside temperature profile.
4. Inside design temperature.

The outside temperature profile can be obtained from the ASHRAE Handbook of Fundamentals. (1) The handbook presents peak temperatures and daily temperature ranges with a 95% or 99% confidence level for most large cities in the United States. With a peak temperature and daily temperature range, the following table is used to calculate the outside temperature profile.

TABLE I
PEAK TEMPERATURE MINUS
THE PERCENTAGE OF DAILY RANGE

Time, hr	%	Time, hr	%	Time, hr	%
1	87	9	71	17	10
2	92	10	56	18	21
3	96	11	39	19	34
4	99	12	23	20	47
5	100	13	11	21	58
6	98	14	3	22	22
7	93	15	0	23	76
8	84	16	3	24	82

The solar conditions are calculated for the particular month and latitude location of the home supplied to the program. This information enables the program to calculate outside environmental conditions the house is exposed to.

The physical dimensions and description of the house are also required.

These include:

1. The number of walls in the house.
2. The angle of house-sun orientation.
3. The orientation of each wall.
4. The dimensions of the walls, ceiling, roof and windows and doors.
5. The construction materials of each wall and the ceiling.
6. The number and construction materials of windows and doors on each wall.
7. The total connected lighting load in the house.
8. The number of family members.

The number of exterior walls is the first descriptive information the program requires. An exterior wall is defined as a continuous exterior surface that maintains a constant wall azimuth. The "I" shaped house shown in Figure 1 has four exterior walls and the "L" shaped house in Figure 2 has six exterior walls. Numbers are used to identify the different walls. Any exterior wall can be designated as the number one wall. Starting with this wall on a floor plan, the remaining walls are numbered in a clockwise manner. If the number one wall is not facing directly south, then it must be a finite angle to the west or east of south. This finite angle is the angle of house-sun orientation. A positive angle is assumed for orientations to the west and a negative angle for orientations to the east. To minimize the angle of house-sun orientation, the number one wall should be selected so that its wall azimuth is as close to due south as possible. The wall azimuth of wall number one will be equal to the angle of house-sun orientation. The remaining wall azimuths are also required. These are supplied to the program as orientation numbers for each wall. The orientation numbers are assigned to the walls by the following method. The home is first rotated by the additive inverse angle of house-sun orientation. This leaves wall number one facing directly south. The walls of the home are now assigned an orientation number by the following table:

**THIS BOOK
CONTAINS
NUMEROUS PAGES
WITH DIAGRAMS
THAT ARE CROOKED
COMPARED TO THE
REST OF THE
INFORMATION ON
THE PAGE.**

**THIS IS AS
RECEIVED FROM
CUSTOMER.**

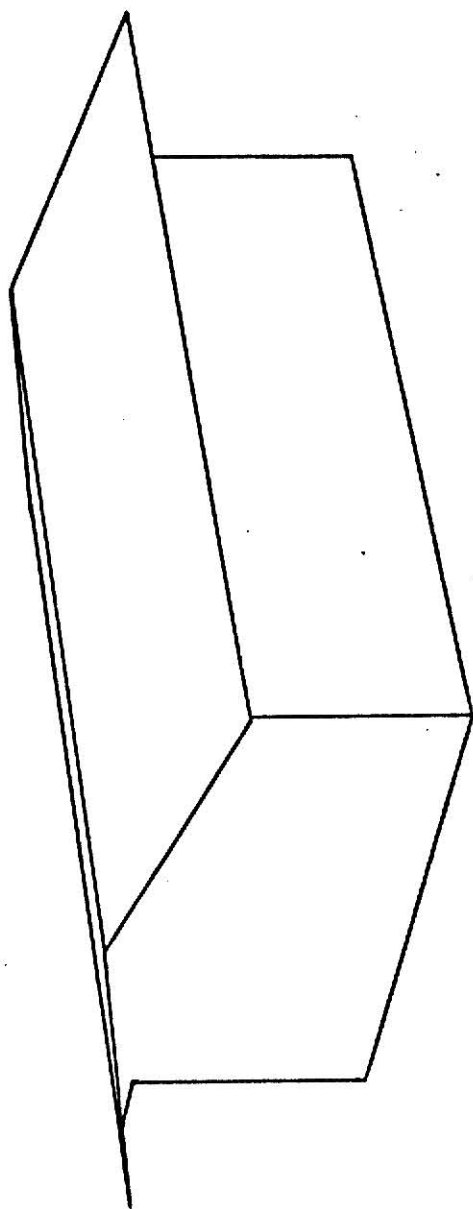


FIGURE 1. I SHAPED HOUSE

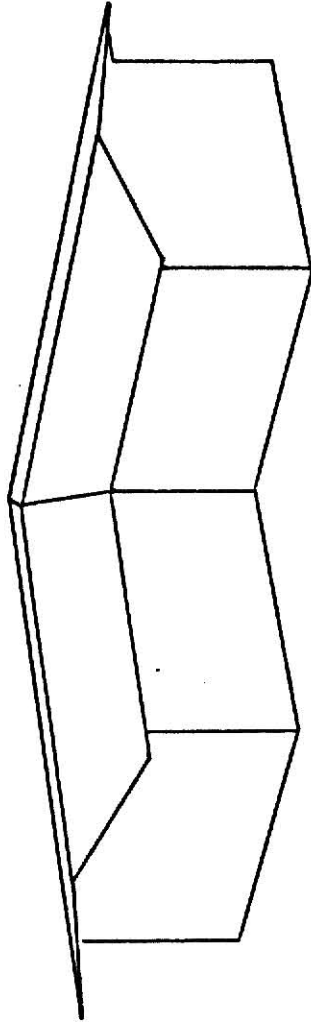


FIGURE 2. L SHAPED HOUSE

TABLE II
ORIENTATION NUMBERS

Wall Facing	Assigned Orientation Number
South	1.0
West	2.0
North	3.0
East	4.0

The wall azimuths are calculated by the following equation:

$$\psi = 90.0 (\text{ORN} - 1.0) + \text{AA}$$

Where,

ψ - Wall azimuth, degree.

ORN - Wall orientation number.

AA - Angle of house-sun orientation, degree.

For each wall, a series of common dimensions must be supplied. These dimensions are:

Wall length

Wall height

Wall tilt angle

Wall area

Roof slope

Roof area

Roof overhang

Roof running width

Roof length

For each window and door on each wall, another series of dimensions are supplied to the program. They are:

Height below window

Height of window

Height above window

Length to the right of the window on the exterior wall

Width of window

Length to the left of the window on the exterior wall

Window area

To complete the description of the home, the type of building material used to construct the walls, windows, doors, and ceiling are supplied to the program. Rather than supplying a list of materials for each wall, different combinations of building materials for walls and ceilings have been stored in the program and identified by construction types. Different window and door types were also stored in the program. For each wall, window, door and the ceiling, a construction type number is supplied to the program.

The total amount of lights supplied to the program is calculated by adding up the individual wattage of every light in the house and multiplying it by 3.413 BTU/hr-watt.

Solar Heat Gains

The first values that are calculated by the program are solar heat gain factors for each exterior surface of the house at each hour of the day. (1,5) A solar heat gain is defined as the total radiant heat admission through a one-eighth inch thick double strength glass due to solar radiation. The solar heat gains are a function of the time of day, the latitude, month of the year, the wall azimuth, and the wall tilt angle. The solar heat gain

(I_t) is made up of three components: the direct solar radiation (I_d), the diffuse sky radiation (I_{ds}), and the solar radiation reflected from surrounding surfaces (I_r). The direct solar radiation is the radiation due to the sun shining directly on a surface and it varies with the angle of incidence the sun rays make with the normal to the surface under consideration. The diffuse sky radiation is the indirect radiation a surface will receive from the sky. The solar radiation reflected from surrounding surfaces is the quantity of radiation received from surfaces around the house such as sidewalks, grass lawns and other homes. These components can be expressed as the following expressions:

$$I_d = I_{DN} \cdot \cos \theta$$

$$I_{ds} = C \cdot I_{DN} \cdot (1.0 - F_{SG})$$

$$I_r = R_g \cdot F_{SG} \cdot (C + \sin(\beta)) \cdot I_{DN}$$

Where,

I_{DN} - Direct normal radiation, BTU/hr-ft².

R_g - Ground reflectance.

β - Solar altitude, rad.

C - Dimensionless constant.

θ - Angle of incidence, rad.

F_{SG} - Shape factor.

These components are combined by the following equation: $I_t = I_d + I_{ds} + I_r$ and adjusted by the use of shading. It should be noted that a surface in the shade has a much smaller solar heat gain than a sunlit surface and that before sunrise and after sunset the solar heat gain is zero. The calculated solar heat gain quantity is used in computing heat gains through windows and in the calculation of sol-air temperatures.

Sol-air Temperatures

The sol-air temperature (1,5) is used in calculating the heat gain through the walls, roof, and doors. The sol-air temperature is an effective temperature defined as:

$$q/A = \alpha I_t + h_o (t_o - t_s) - \epsilon \Delta R = h_o (t_e - t_s)$$

or:

$$t_e = t_o + \alpha I_t / h_o - \epsilon \Delta R / h_o$$

Where,

t_s - Surface temperature, F.

t_e - Sol-air temperature, F.

t_o - Outside temperature, F.

q/A - Heat flux into a sunlit surface, BTU/hr-ft².

α - Absorption of the surface for solar radiation.

I_t - Solar heat gain, BTU/hr-ft².

ϵ - Hemispherical emittance of the surface.

ΔR - The difference between the long wave radiation incident on the surface from the sky and surroundings, and the radiation emitted by a black body at outdoor air temperature, Btu/hr-ft².

h_o - Coefficient of heat transfer by radiation and convection at the outer surface, Btu/hr-ft² F.

The term $\epsilon \Delta R / h_o$ is assumed equal to 7.0 F for horizontal surfaces or surfaces that are non-vertical, such as roofs. For vertical surfaces this term is zero. The sol-air temperature for walls is simply the outside dry bulb temperature during the night time or when the solar heat gain is zero.

Shading

To adjust the sol-air temperature or solar heat gain it is important that accurate shading information is known about the desired exterior surface. Little has been done with residential type shading problems

because it is easy to get into many different and unique shading problems with a few different home designs. For this reason, it is very difficult to write a generalized shading routine to handle all types of homes. This program tries to handle only homes that are constructed with the same number of floors throughout, eliminating the so-called split level home. A sample of some of the homes that could be run are shown in Figures 1 through 5. These homes are shaped like the letters E, I, L, T, U, or X, but the hip ridges of their roofs must always meet at the same level. It is also important that each leg or extension from the home must depart at a right angle. Unlike other parts of the program, which are outlined in the ASHRAE Handbook, the shading subroutine is a unique, original subroutine. Appendix A gives examples of calculations involved in the shading subroutine. The shading subroutine calculates the solar azimuths and solar altitudes of the sun's rays and calculates shading on a surface due to adjacent parts of the house.

There are three different parts that are considered. They are:

1. The roof overhang.
2. Perpendicular walls.
3. The roofs over perpendicular walls.

In the first instance, the shading on a surface is due to the overhang directly over it. The shading on a surface due to a perpendicular wall is the amount of surface that is shaded by that wall. The third part is similar to the second part. The only difference is that the shading occurs due to the roof rather than the wall. Shadowing from external objects such as other homes or trees are not handled in this subroutine.

The information from the shading subroutine is used to calculate adjusted sol-air temperatures and adjusted solar heat gains. This is done

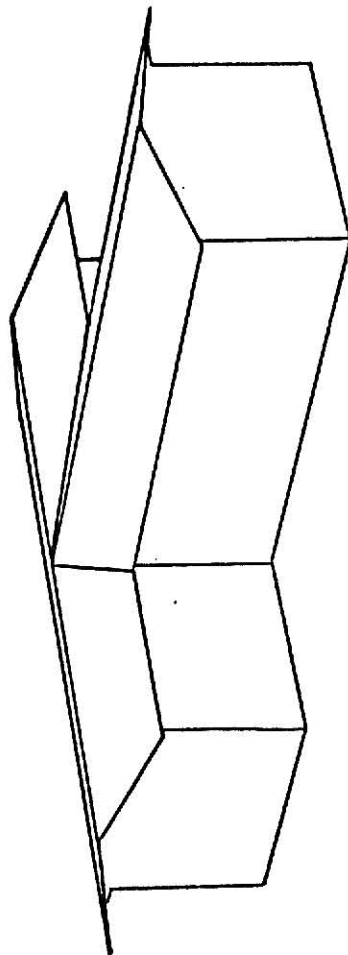


FIGURE 3. T SHAPED HOUSE

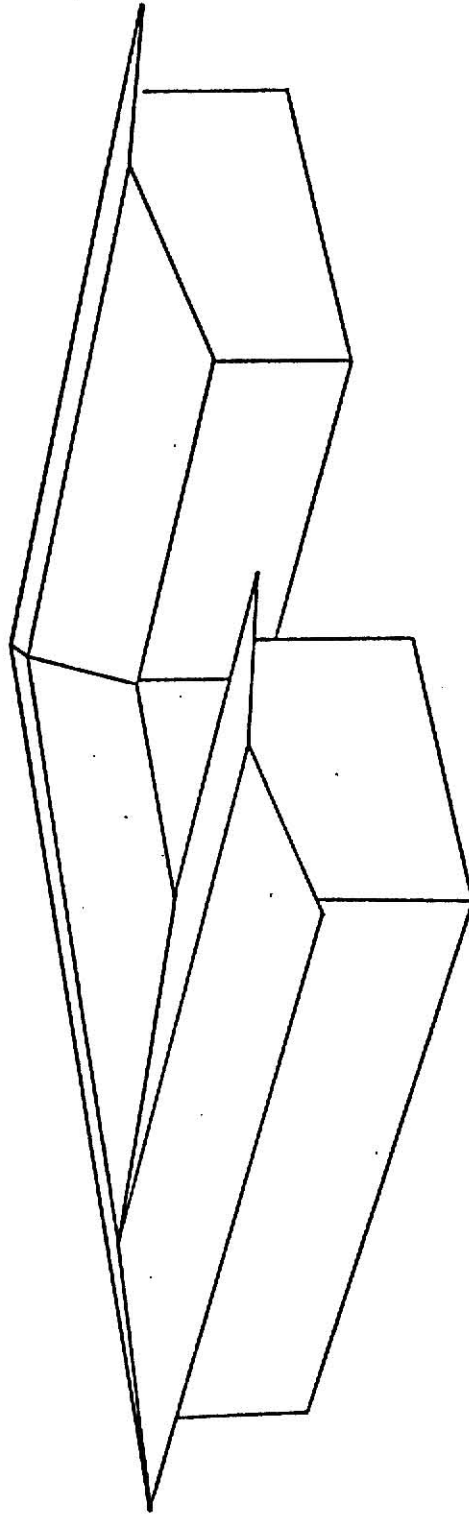


FIGURE 4. U SHAPED HOUSE

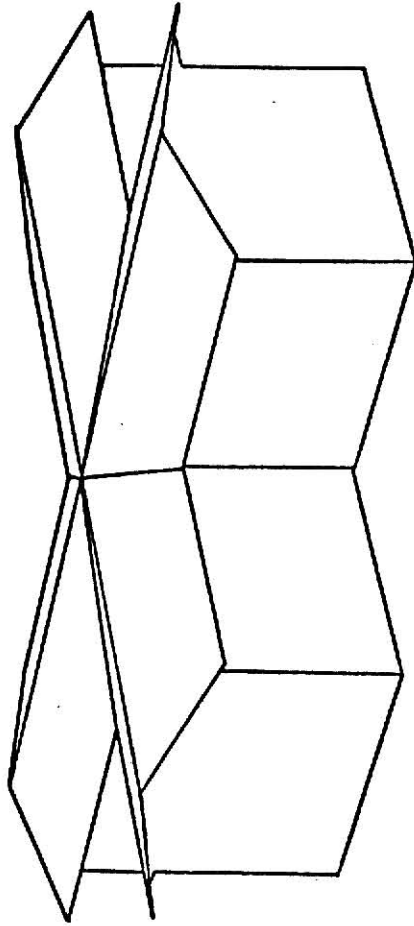


FIGURE 5. X SHAPED HOUSE

by a weighted average between the sunlit sol-air temperature and the shaded sol-air temperature to obtain an adjusted sol-air temperature. The following algebraic equation expresses this more clearly:

$$T_e \text{ adjusted} = (T_e \text{ sunlit} \cdot B) + T_e \text{ shaded} \cdot (1.0 - B)$$

Where,

T_e - Sol-air temperature, F.

B - Decimal fraction sunlit.

A similar expression is used to calculate the adjusted solar heat gains for windows.

Wall Heat Gain

One of the largest heat gain components is the conduction heat gain through the walls; therefore, it is very important that this quantity is calculated as closely as possible. Walls are usually made of heavy materials that have a high heat capacity and with considerable insulation; therefore, outside surface temperatures from a number of hours earlier may affect the interior heat gain at a particular hour. The response factor method (1,4,2) uses a series combination of past and present adjusted sol-air temperatures along with a second series of past heat gains to calculate the present heat gain. This method is a very good approximation of wall reaction to the adjusted sol-air temperatures. The basic equation that is used in the problem is expressed as follows:

$$q/A_{P,T} = \sum_{n=0} b_n \cdot (t_{b,T-n\Delta}) - \sum_{n=1} d_n \cdot (q/A_{P,T-n\Delta}) - t_{rc} \cdot \sum_{n=0} c_n$$

Where,

$q/A_{P,T}$ - Heat flux through wall P at time $T = 0$, BTU/hr-ft².

$t_{b,T-n\Delta}$ - Adjusted sol-air temperature, F, at time $(T - n)$ hours.

t_{rc} - Constant interior space temperature, F.

b_n - Sol-air temperature series coefficients, BTU/hr-ft².

c_n - Constant, BTU/hr-ft² F.

d_n - Heat flux series coefficients.

Note that the b_n 's, c_n 's and d_n 's are required for each different type of wall.

The q/A 's must all be initialized equal to zero before beginning any calculations with the response factor method. At the end of the first twenty-four hour calculation, the q/A should still be growing in magnitude. For the q/A 's to converge, the last few q/A 's at the end of the first twenty-four hours are needed to get the second twenty-four hour calculations started. After about three twenty-four hour periods, these q/A 's converge. To check for convergence, the following relationship is checked:

$$\left\{ U_{\text{wall}} \cdot \left[\sum_{n=1}^{24} t_{\text{en}} - t_{\text{inside}} \right] - \left[\sum_{n=1}^{24} q/A_n \right] \right\} \leq \epsilon$$

ϵ - Arbitrary convergence criteria ($\epsilon \leq .02$ in this thesis).

U_{wall} - Overall heat transfer coefficient, BTU/hr-ft² F.

t_e - Adjusted sol-air temperatures, F.

t_{inside} - Inside design temperature, F.

q/A - Calculated heat fluxes, BTU/hr-ft².

If this relationship is not satisfied, then a fourth or fifth twenty-four hour period will be calculated.

The subroutine was programmed to contain b_n 's, c_n 's, and d_n 's for six different types of walls. These wall types were:

1. An uninsulated frame wall.
2. A frame wall with one inch of insulation.
3. A frame wall with two inches of insulation.

4. A frame wall with three inches of insulation.
5. A brick veneered frame wall with no insulation.
6. A brick veneered frame wall with three inches of insulation.

The program is not limited to just these six walls. If a different type of wall is wished to be used, the b's, c's, and d's can be calculated according to the fortran program (2) written by G.P. Mitalas.

Ceiling Heat Gain

A similar response factor method was used to calculate the heat gains through the ceiling except that instead of using sol-air temperature in the response factor method, the attic temperatures were used. This generated the need to calculate the attic air temperature profile. The following two simultaneous differential equations describe the attic temperature:

$$A_R \cdot h_o \cdot (T_{eAR} - T_{roof}) = M_R \cdot C_R \cdot dT_{roof}/dt + A_R \cdot h_A \cdot (T_{roof} - T_{attic})$$

and

$$A_R \cdot h_o \cdot (T_{roof} - T_{attic}) = M_{attic} \cdot C_{air} \cdot dT_{attic}/dt + \dot{q}_{ceiling}$$

Where,

A_R - Area of the roof, ft^2

h_o - Outside convection coefficient, $BTU/hr-ft^2 \cdot F$

T_{eAR} - Average sol-air temperature on the roof, F

T_{roof} - Temperature of the roof, F

M_R - Mass of the roof, lbs

C_R - Heat capacity of the roof, $BTU/lb \cdot F$

t - Time, hrs

h_a - Inside convection coefficient, $BTU/hr-ft^2 \cdot F$

T_{attic} - Temperature of the attic, F

M_{attic} - Mass of the air in the attic, lbs

C_{air} - Specific heat of air, BTU/lbs- F

$\dot{q}_{\text{ceiling}}$ - Total heat gain through the ceiling, BTU/hr.

The heat gain through the ceiling is calculated by the response factor method once an attic temperature profile has been calculated. The two equations above can be solved for T_{roof} and T_{attic} at any time based on the values of the time interval before; therefore, these calculations need to be done for five twenty-four hour periods. After five twenty-four periods, the calculations used to obtain the T_{roof} and T_{attic} profiles will have converged. For a further description of the attic temperature calculations, see the appendix.

Window Heat Gain

The heat gains from the windows have two components, a solar component and a conduction component. The solar heat gain is calculated by the product of the window area times the shading coefficient times the adjusted solar heat gains. This is expressed by the following equation:

$$Q_{\text{solar}} = A_w \cdot SC \cdot SHG_{\text{adj.}}$$

where,

A_w - Area of the window, ft^2 .

SC - Shading coefficient.

SHG_{adj} - Adjusted solar heat gain, BTU/hr-ft^2 .

The shading coefficient is a function of the glass and the drapery type; therefore, this information is stored in the program for different types of glass and drapery combinations. The thermal load on the windows is handled as a simple one dimensional heat conduction problem where an overall heat transfer coefficient and a temperature difference across the glass is used to calculate the instantaneous load. The equation used to describe this is:

$$Q_{\text{thermal}} = A_w \cdot U_{\text{window}} \cdot (T_{\text{outside}} - T_{\text{inside}})$$

where,

A_w - Window area, ft^2 .

U_{window} - Overall heat transfer coefficient for the window, $\text{BTU/hr-ft}^2\text{F}$.

T_{outside} - Outside temperature, F.

T_{inside} - Inside temperature, F.

Two types of window constructions were written into the program. The first type is a single pane glass with no draperies or blinds. The second type is insulated glass with a 1/2 inch air space and white opaque roller shades. To simplify the input, just these two types were written into the present version of the program.

Door Heat Gain

The door heat gains were handled similarly to the windows, except the solar load was included in the adjusted sol-air temperature. The adjusted sol-air temperature was used in the following equation to calculate the door heat gains:

$$Q_{\text{door}} = A_{\text{door}} \cdot U_{\text{door}} \cdot (T_e - T_{\text{inside}})$$

where,

Q_{door} - Heat flux through the door, BTU/hr .

A_{door} - Area of the door, ft^2 .

U_{door} - Overall heat transfer coefficient, $\text{BTU/hr-ft}^2\text{F}$.

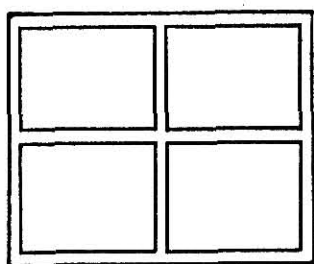
T_e - Adjusted sol-air temperature, F.

T_{inside} - Inside design temperature F.

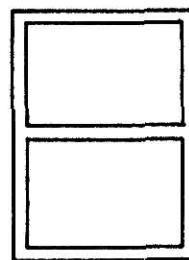
An instantaneous heat conduction calculation was made on the doors and windows because of their light construction.

Infiltration Loads

Infiltration was a difficult load to predict because it depends on so many variables. Some of these are wind velocity, width of crack, flow per foot of crack, and the temperature of the outside air. The best estimate of the infiltration load is based on data that is presented in the ASHRAE Handbook for a great many different types of windows. (1) The infiltration subroutine contains flow information for double hung, awning, and sliding glass as well as different combinations of each type. Two examples of these combinations are two double hung windows side by side as in Figure 6, or two awning type windows one on top of the other as in Figure 7.



**FIGURE 6. TWO
DOUBLEHUNG WINDOWS**



**FIGURE 7. TWO
AWNING WINDOWS**

In addition to this information, there is included flow information for each type of window for each season of the year. With the type and combination of windows and with the window dimensions, the length of crack can be calculated.

There are two components to be calculated to get the total infiltration, the sensible and the latent. Both of these components are calculated in the infiltration subroutine.

The sensible infiltration is calculated by the following equation that makes use of the above stored values:

$$I_{\text{sens}} = C \cdot L \cdot Q \cdot (T_{\text{outside}} - T_{\text{inside}})$$

where,

I_{sens} - Sensible infiltration per window, BTU/hr.

C - Constant, BTU/ft³.

L - Length of crack, ft.

Q - Air flow through the crack per foot, ft³/hr-ft.

T_{outside} - Outside temperature, F.

T_{inside} - Inside temperature, F.

The latent infiltration is calculated by the following equation:

$$I_{\text{lat}} = C_1 \cdot L \cdot Q \cdot (W_o - W_I)$$

where,

C_1 - Constant, BTU/ft³.

L - Length of crack, ft.

Q - Air flow through the crack per foot, ft³/hr-ft.

I_{lat} - Latent infiltration per window, BTU/hr

W_o - Outside humidity ratio.

W_I - Inside humidity ratio.

There are two different rates of crack flow written into the subroutine: a weatherstripped flow for use with the insulated glass and roller shades, and a nonweatherstripped flow for use with the uninsulated single pane glass with no draperies. The outside humidity ratio was considered a constant for the entire day. The total infiltration for the entire home at each hour is obtained by adding up the infiltration from each window and door in the house and dividing this by two. This total is divided by two because half of the house is infiltrating while the other half is exfiltrating.

Internal Loads

The last heat gain to be calculated is the internal load that occurs in a home. This load consists of lights, appliances, and people. Since

occurrence of these components is dependent upon the human factor, they are extremely hard to predict. Because of this, the appliance loads were neglected in the current version of the program. The load schedules for the lights and people were estimated in the program.

The single input to the people schedule was chosen as the number of members in the family. This information was used in the following schedule to calculate a sensible and latent load. For the first six hours of the day, the entire family is sleeping at a metabolic rate of 200 BTU/(hr)(prs) sensible and 100 BTU/(hr)(prs) latent. At seven o'clock in the morning everyone in the family gets up and gets ready for the day. This means that they are taking showers, getting dressed, and preparing breakfast. Since this is a faster pace, their metabolic rates were chosen as 250 BTU/(hr)(prs) sensible and 200 BTU/(hr)(prs) latent. During the day from eight o'clock a.m. until four o'clock p.m., everyone is gone except the homemaker who normally works at a metabolic rate of 250 BTU/(hr)(prs) sensible and 200 BTU/(hr)(prs) latent. Starting at five o'clock p.m. the rest of the family comes home and until ten o'clock p.m. their activity will remain high or at a rate of 250 BTU/(hr)(prs) sensible and 200 BTU/(hr)(prs) latent. The last two hours of the day the entire family is asleep at the same rates used for the first six hours of the day. Again, this is just a representative schedule for use in this program; the program can easily accept more accurate representations if they are known.

The light load was calculated on a similar schedule. It was assumed that no lights would be needed until one hour before sunset in the evening. The program determines the hour of sunset and one hour before this the lights are turned on and left on until ten o'clock p.m. Because most people are trying to keep unnecessary lights turned off in an effort to

save money, it is assumed that the amount of lights on at any one time will never be the maximum amount of light installed in the house. The lighting load chosen was one-third of the total light capacity during the evening.

Cooling Load Calculation

After all heat gain components have been computed, all that remains is to delay certain heat gains that appear as delayed cooling loads and to sum up the instantaneous cooling loads. The heat gains from the walls, ceiling, windows, and doors are loads that do not appear as cooling loads instantaneously. These heat gains, before becoming cooling loads, must actually raise the temperature of the air in the room. The amount of time it takes for these heat gains to become a cooling load depends on the home construction. The actual quantitative cooling load is calculated by a response factor method that uses past heat gain and cooling load information to calculate a present cooling load. (1;4) The following equation from the ASHRAE Handbook of Fundamentals demonstrates the procedure for calculation of loads for walls, ceiling, doors and windows:

$$Q_T = \text{Factor} \cdot \sum_{i=0} (v_i \cdot q_{e,T-i\Delta}) - \sum_{i=1} (w_i \cdot Q_{T-i\Delta})$$

where,

Factor - A construction correction.

Q_T - The cooling load at time $T - 0$, hr.

$q_{e,T-i\Delta}$ - Heat gain at time $T - i$, hr.

v_i and w_i - Response factor coefficients.

The total heat gains from the windows, walls, doors, and ceiling are totalled each hour to obtain a fixed profile for this equation. The v 's and w 's have been calculated for light construction, typical frame home,

but to make these response factors more accurate for the home under consideration the correction factor serves as a fine adjustment. The Q_T 's are the resulting cooling loads for the twenty-four hour period.

After making these calculations for three or four twenty-four periods, these cooling load values will have converged. The rest of the heat gains, including lights, people sensible and latent, and infiltration sensible and latent are added to the cooling load from the walls, windows, doors and ceiling to obtain the total cooling load for each hour of the day for the entire house. These heat gains were considered instantaneous because they change the air temperature immediately when they occur.

For a more detailed explanation of the calculations and of the program flow, a commented copy of the program is contained in the appendix.

CHAPTER IV

RESULTS AND DISCUSSION

In the preceding chapters the cooling load program was developed. In this chapter program capability is demonstrated by consideration of results from a "test" home.

The test house was located on the 40th north latitude. August 21st was chosen as the test date. The inside design temperature was chosen as 75 F and the outside dry bulb temperature profile used is shown in Figure 8. The inside humidity ratio was 0.0108 pounds of moisture per pound of dry air and the outside humidity ratio was 0.0158 pounds of moisture per pound of dry air. All of the previous information except the inside and outside humidity ratio was supplied as input data to the program. These were written as fixed values in the infiltration subroutine. The house used as the test home is depicted in Figures 9 through 12. The house was a two story house with a foundation size of 40'0 x 27'4 and a total living area of 2,109 square feet. As seen in the figures, the front of the house faced south. The figures also illustrate the location of the windows and doors on the house. The dimensions of the windows and doors were supplied to the computer as data. The variables left to be manipulated in the program were: the construction materials (of the walls, ceiling, windows and doors), width of overhang and angle of house-sun orientation. These variables were changed in each case to demonstrate the ability of the program. The following is a description of each of the six cases treated.

Case I

Wall construction: Uninsulated frame wall

Ceiling construction: Uninsulated ceiling

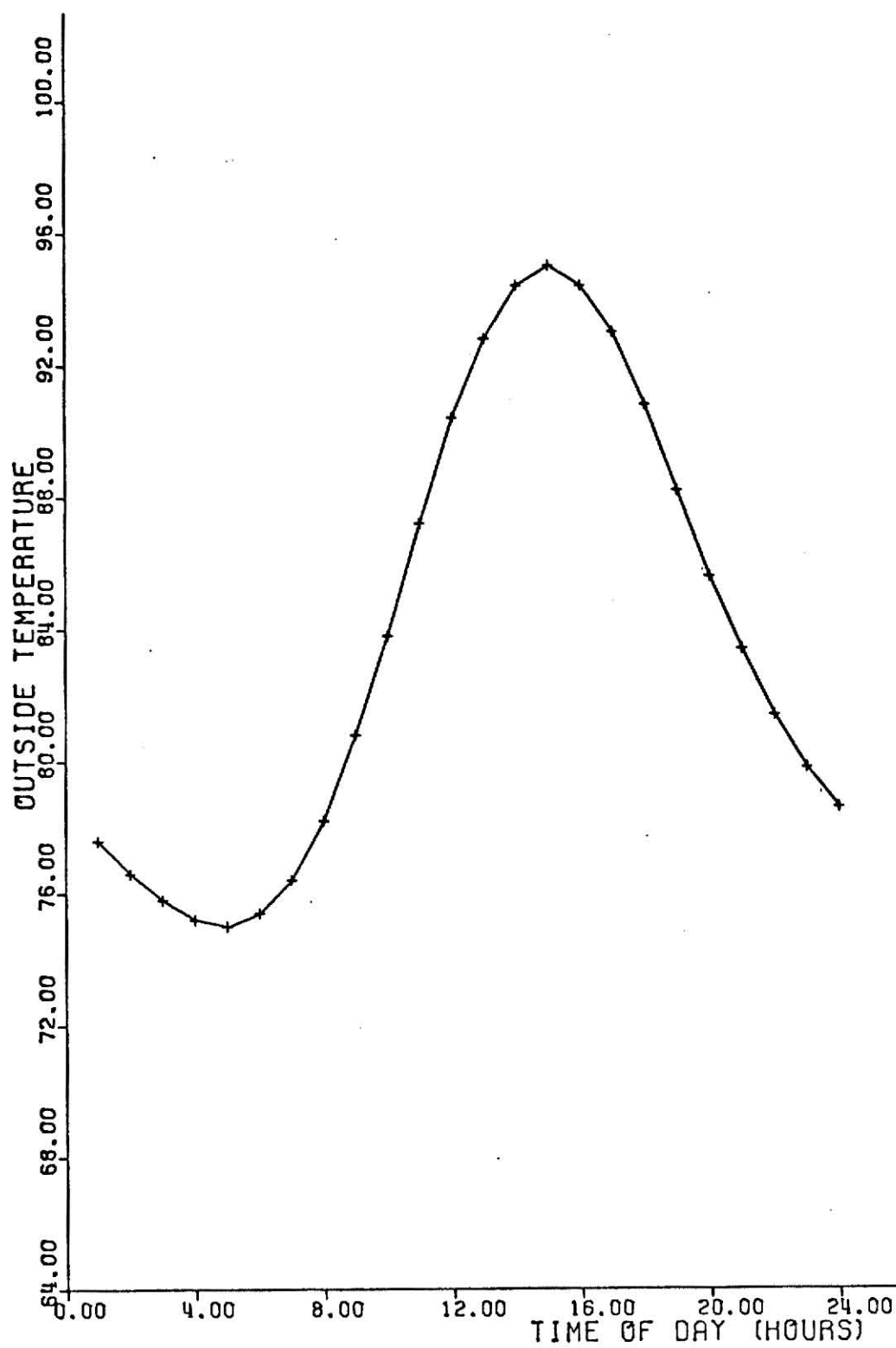


FIGURE 8. DAILY OUTSIDE TEMPERATURE PROFILE

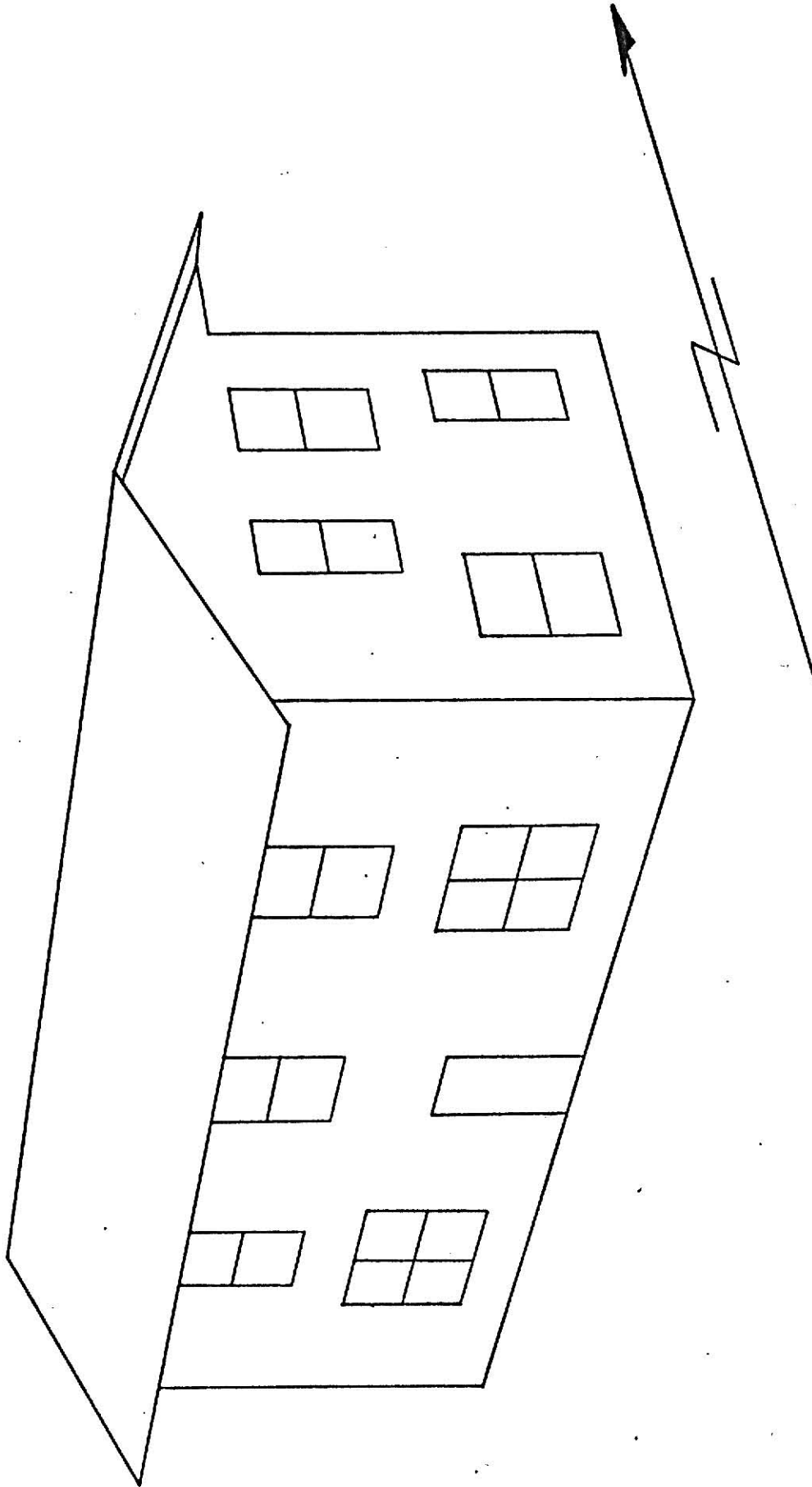


FIGURE 9. FRONT VIEW OF TEST HOUSE

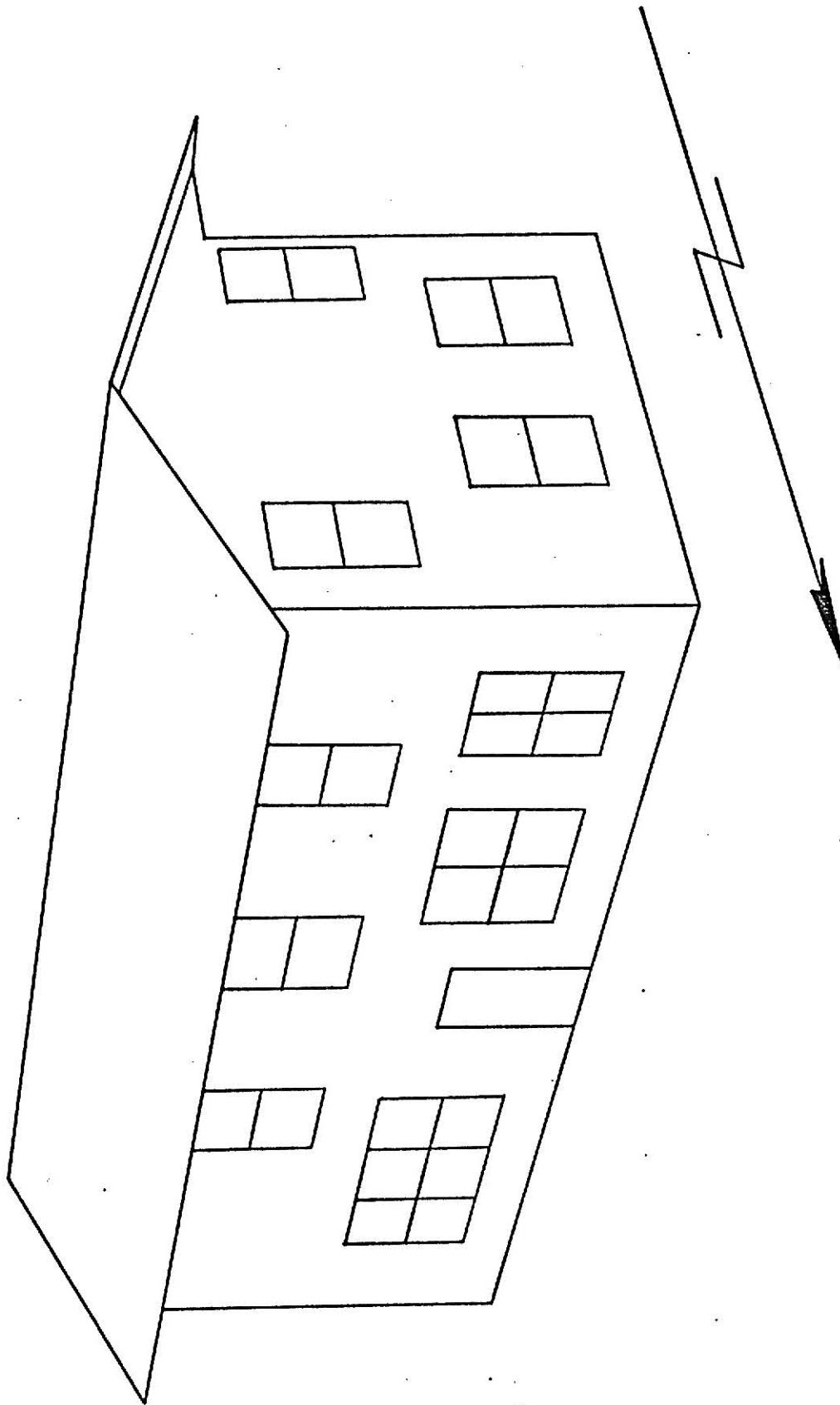


FIGURE 10. BACK VIEW OF TEST HOUSE

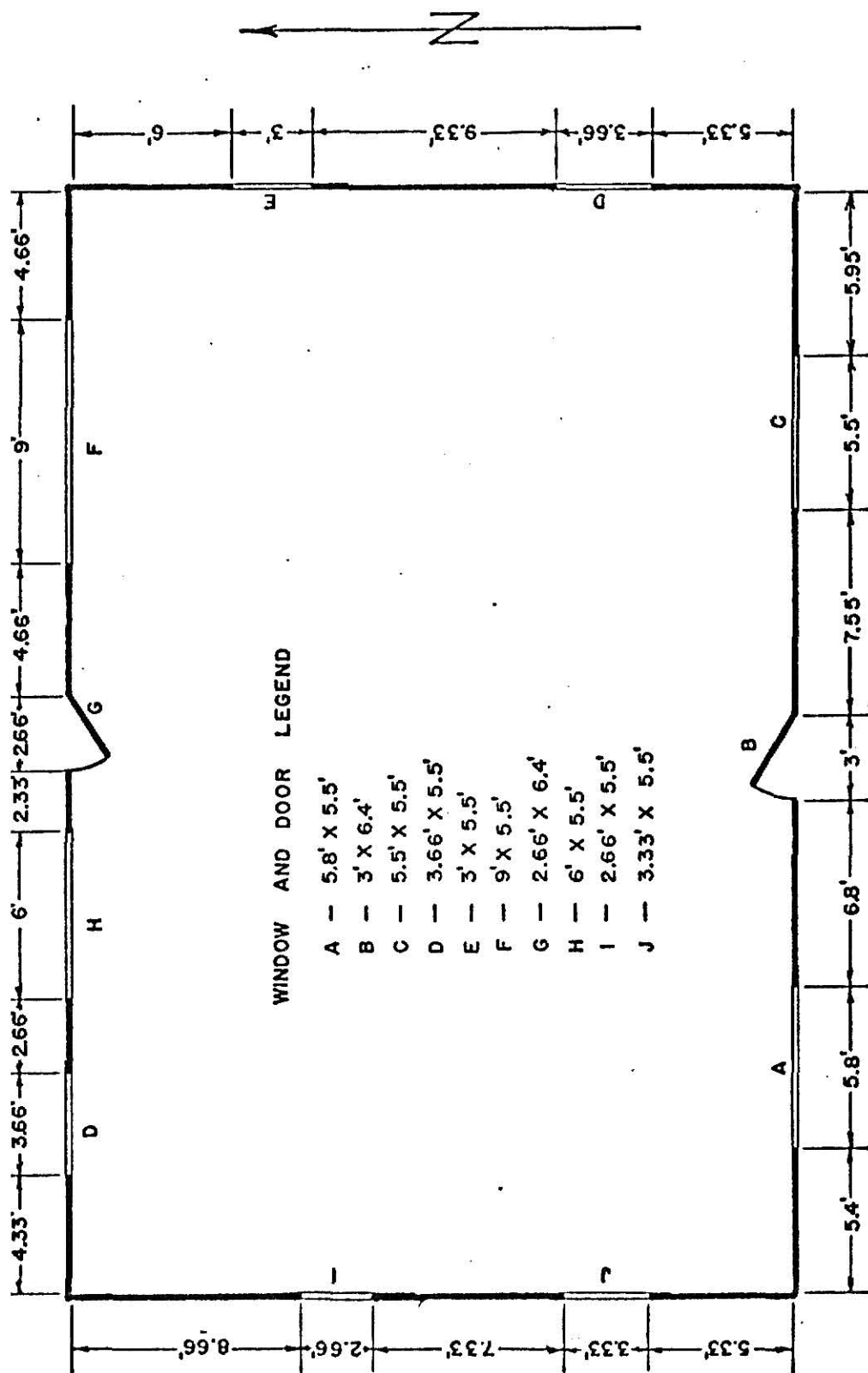


FIGURE II. FIRST FLOOR TEST HOUSE

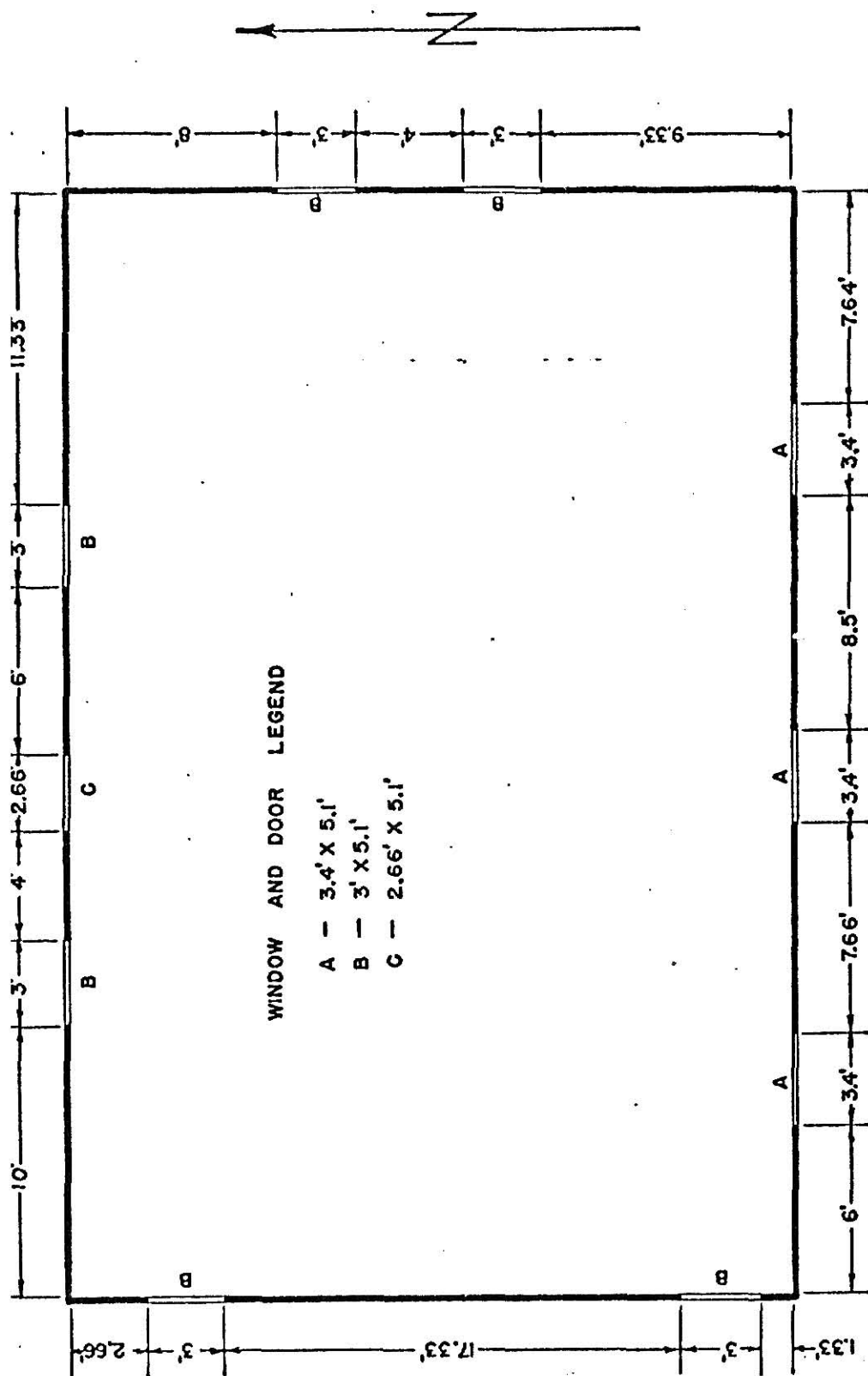


FIGURE 12. SECOND FLOOR TEST HOUSE

Window construction: Single sheet, unweatherstripped, no internal shading

Door construction: One and one-half inches of wood

Orientation: Facing south

Width of overhang: Three feet

Purpose: To establish a base load for comparison in subsequent cases.

Case II

Wall construction: Frame wall, one inch of insulation

Ceiling construction: Three inches of insulation

Window construction: Single sheet, unweatherstripped, no internal shading

Door construction: One and one-half inches of wood

Orientation: Facing south

Width of overhang: Three feet

Purpose: To illustrate the load reduction with the addition in insulation.

Case III

Wall construction: Frame wall, one inch of insulation

Ceiling construction: Three inches of insulation

Window construction: Insulated double pane, weatherstripped, white opaque roller shades

Door construction: One and one-half inches of wood

Orientation: Facing south

Width of overhang: Three feet

Purpose: To illustrate the reduction in load due to the window change

Case IV

Wall construction: Three inches of insulation, frame wall
Ceiling construction: Six inches of insulation
Window construction: Insulated double pane, weatherstripped,
white opaque roller shades
Door construction: One and one-half inches of wood
Orientation: Facing south
Width of overhang: Three feet
Purpose: To illustrate the load with this added insulation
and for comparison to the next two cases

Case V

Wall construction: Three inches of insulation, frame wall
Ceiling construction: Six inches of insulation
Window construction: Insulated double pane, weatherstripped,
white opaque roller shades
Door construction: One and one-half inches of wood
Orientation: Facing south
Width of overhang: One and one-half feet
Purpose: To illustrate the ability of the shading subroutine

Case VI

Wall construction: Three inches of insulation, frame wall
Ceiling construction: Six inches of insulation
Window construction: Insulated double pane, weatherstripped,
white opaque roller shades
Door construction: One and one-half inches of wood
Orientation: Facing west
Width of overhang: Three feet
Purpose: To illustrate the ability to rotate the house

The results of the load calculation for the six different cases are presented in graphical form in Figures 13 through 19. The numerical values on which the graphs are based are given in Table 3. The computer time on the IBM 370-158 to run these six cases was 29 seconds.

Case one was chosen as the base for the first four cases. The house chosen for this case was a structure with the lowest thermal resistance that the program was capable of, which implies the highest possible cooling load. Cases two, three and four were compared to this base load to demonstrate the load reduction of each case.

The cooling load profile in Figure 13 has a dip in it at 12 o'clock noon. A dip occurs when the sun azimuth aligns with a wall azimuth. In this case, at 12 o'clock noon the sun azimuth is aligned with the azimuth of the wall facing south. At 11 o'clock in the morning or at 1 o'clock in the afternoon the sun strikes two walls, but at 12 o'clock noon the sun can only strike the south wall. Since only one wall is sunlit at 12 o'clock, this will result in a slight dip in the cooling load. Two other dips occur. One at 7 o'clock in the morning and the other at 5 o'clock in the afternoon. These two dips were not as prominent as the one at 12 o'clock noon for case one. All three dips are equally significant in case four as depicted in Figure 16.

The cooling load for case one peaked at 76,327 BTU/hr at 3 o'clock in the afternoon. The daily cooling load was 911,565 BTU/day.

Case two was chosen to show the reduction in load caused by adding one inch of insulation to the walls and three inches of insulation to the ceiling. The peak cooling load was 49,328 BTU/hr at 3 o'clock in the afternoon, with a daily load of 601,979 BTU/day. This was a reduction of 35.4% in the peak load and 34% in the daily cooling load.

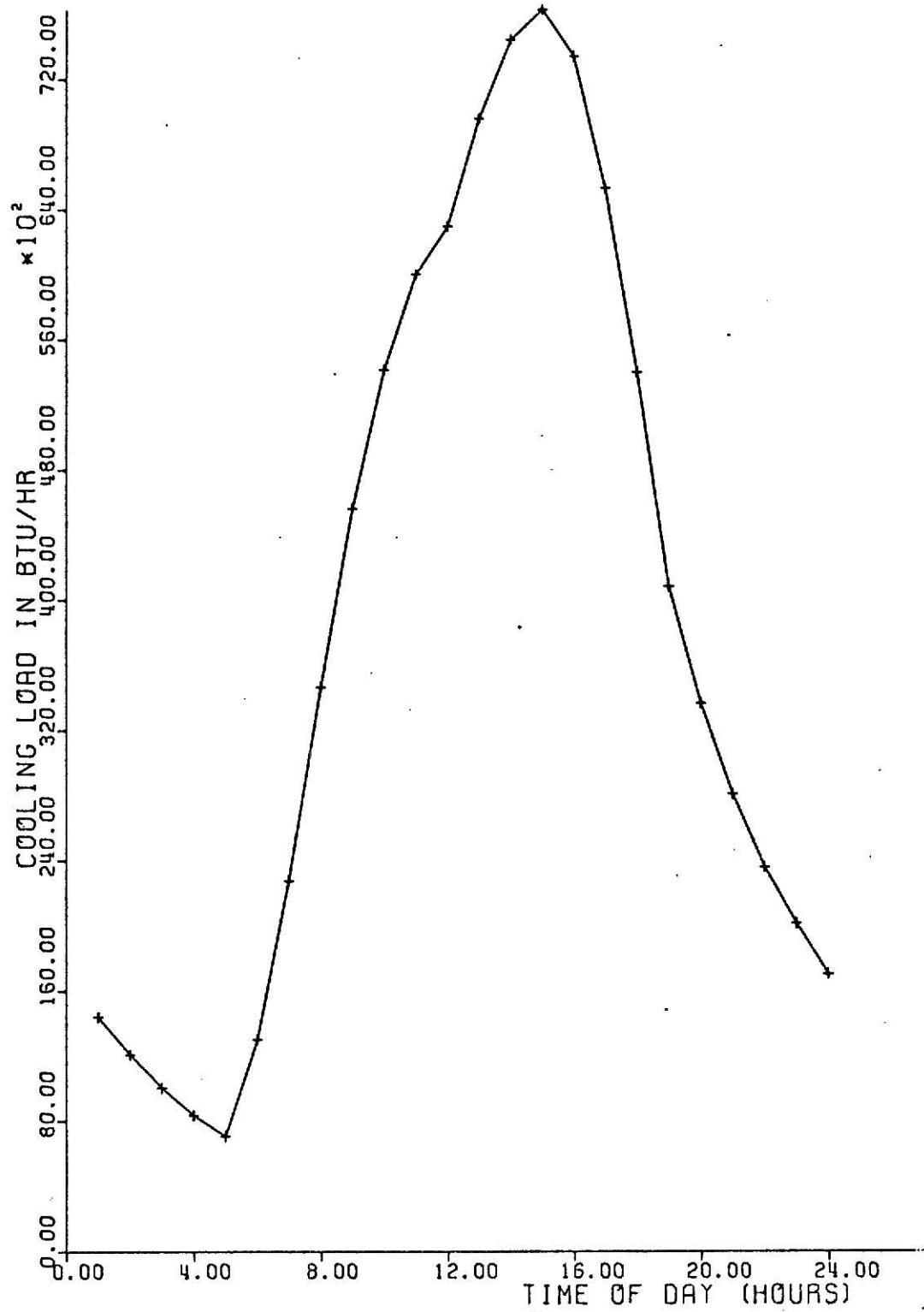


FIGURE 13. COOLING LOAD PROFILE CASE ONE

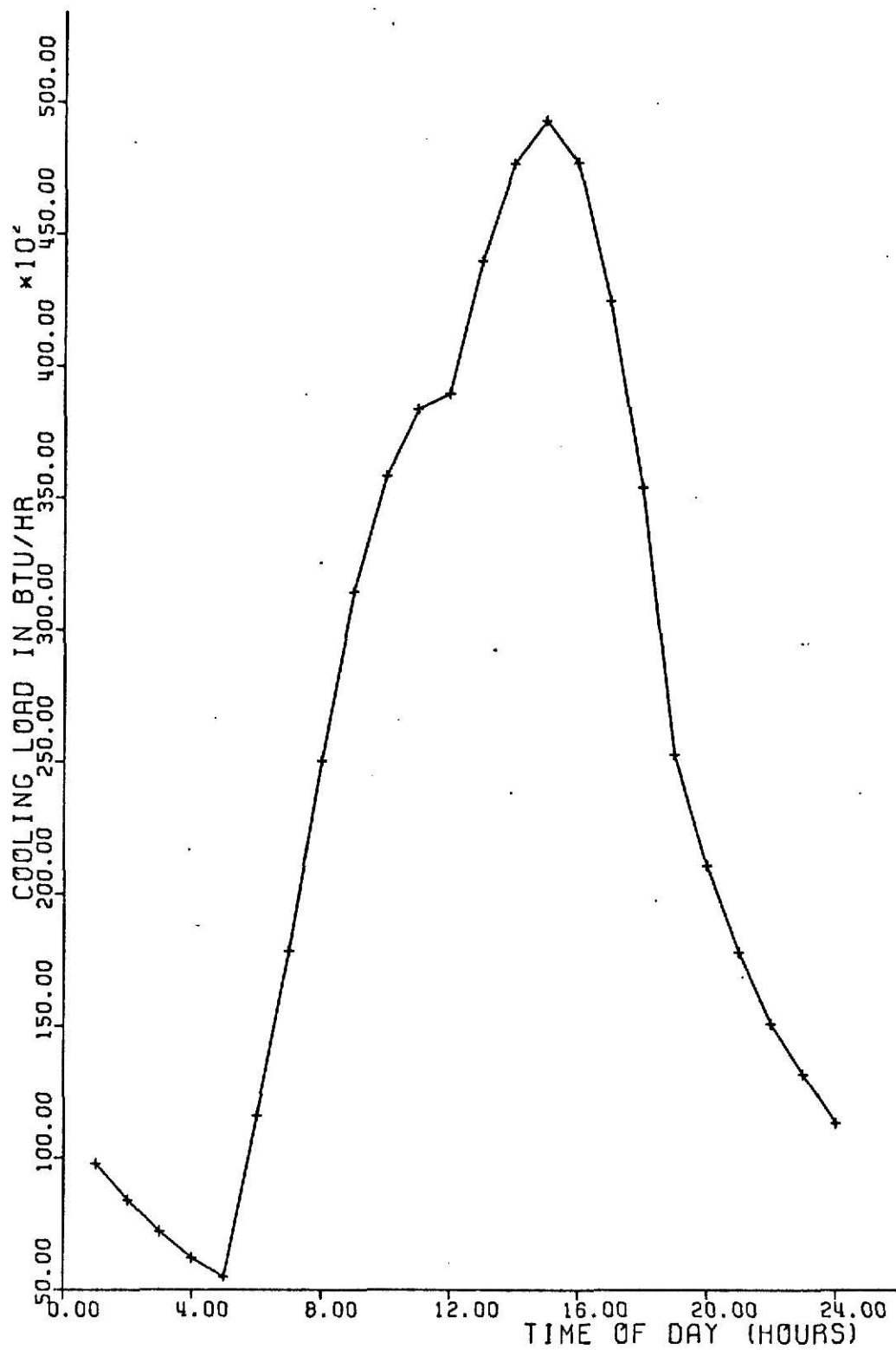


FIGURE 14. COOLING LOAD PROFILE CASE TWO

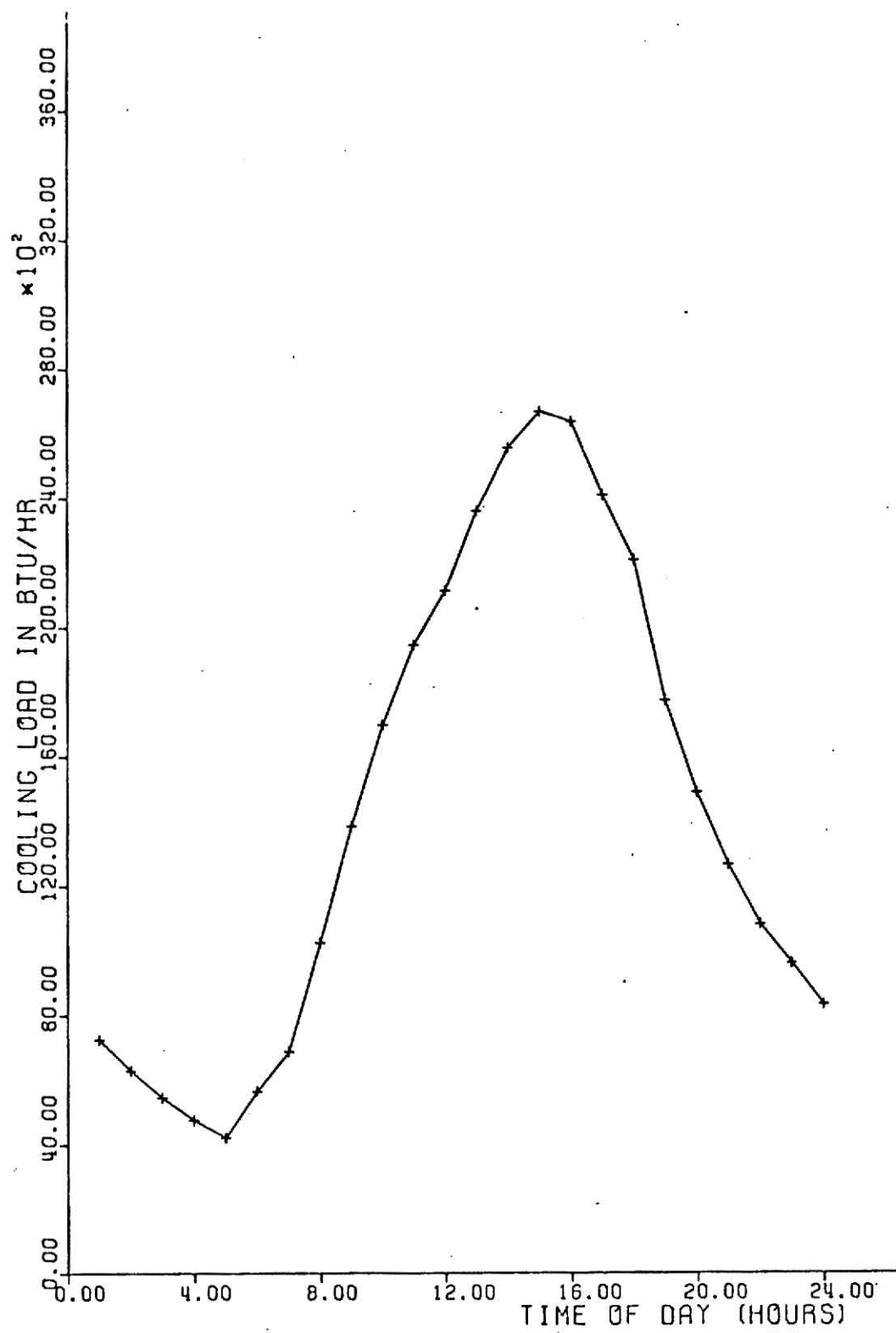


FIGURE 15. COOLING LOAD PROFILE CASE THREE

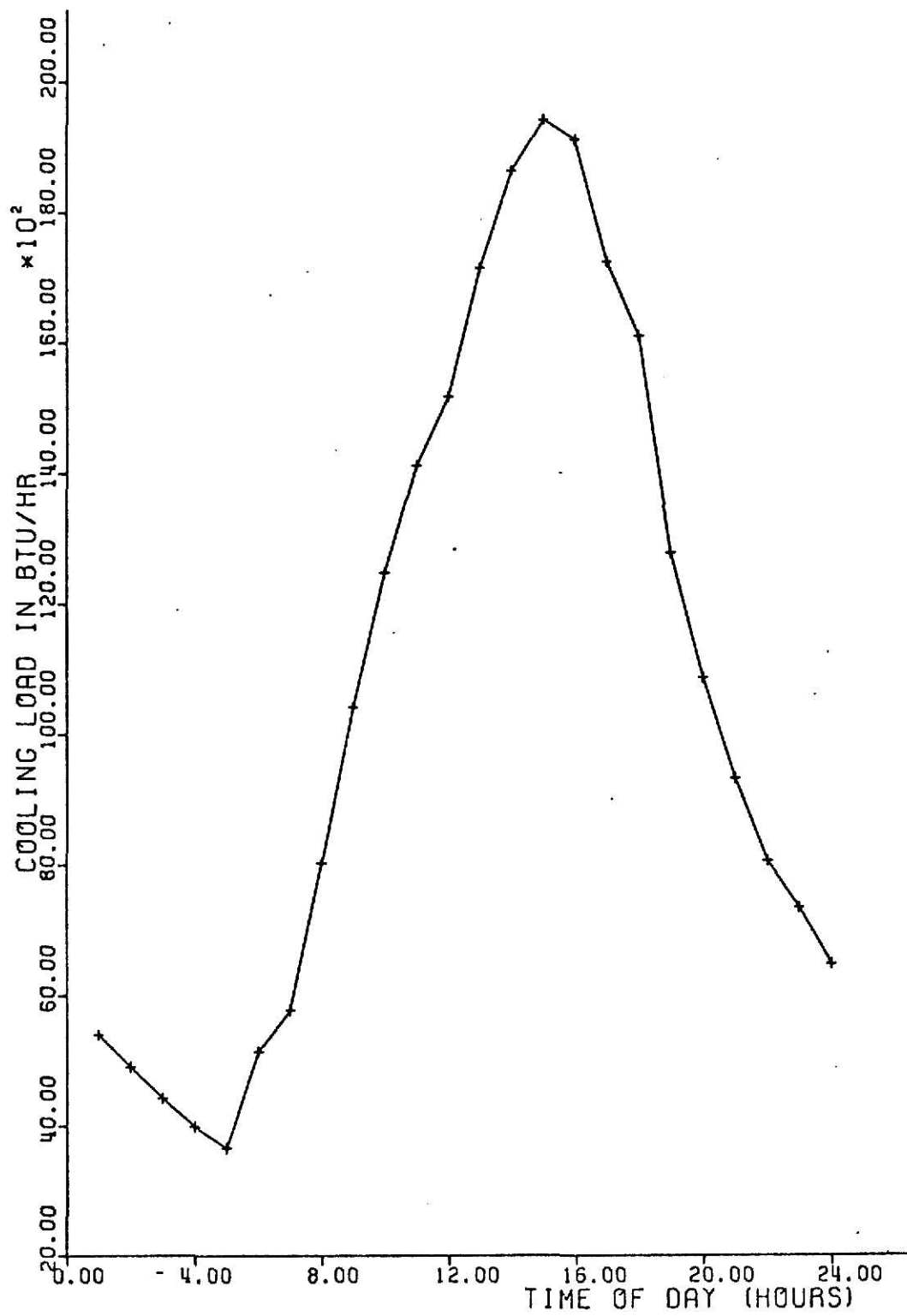


FIGURE 16. COOLING LOAD PROFILE CASE FOUR

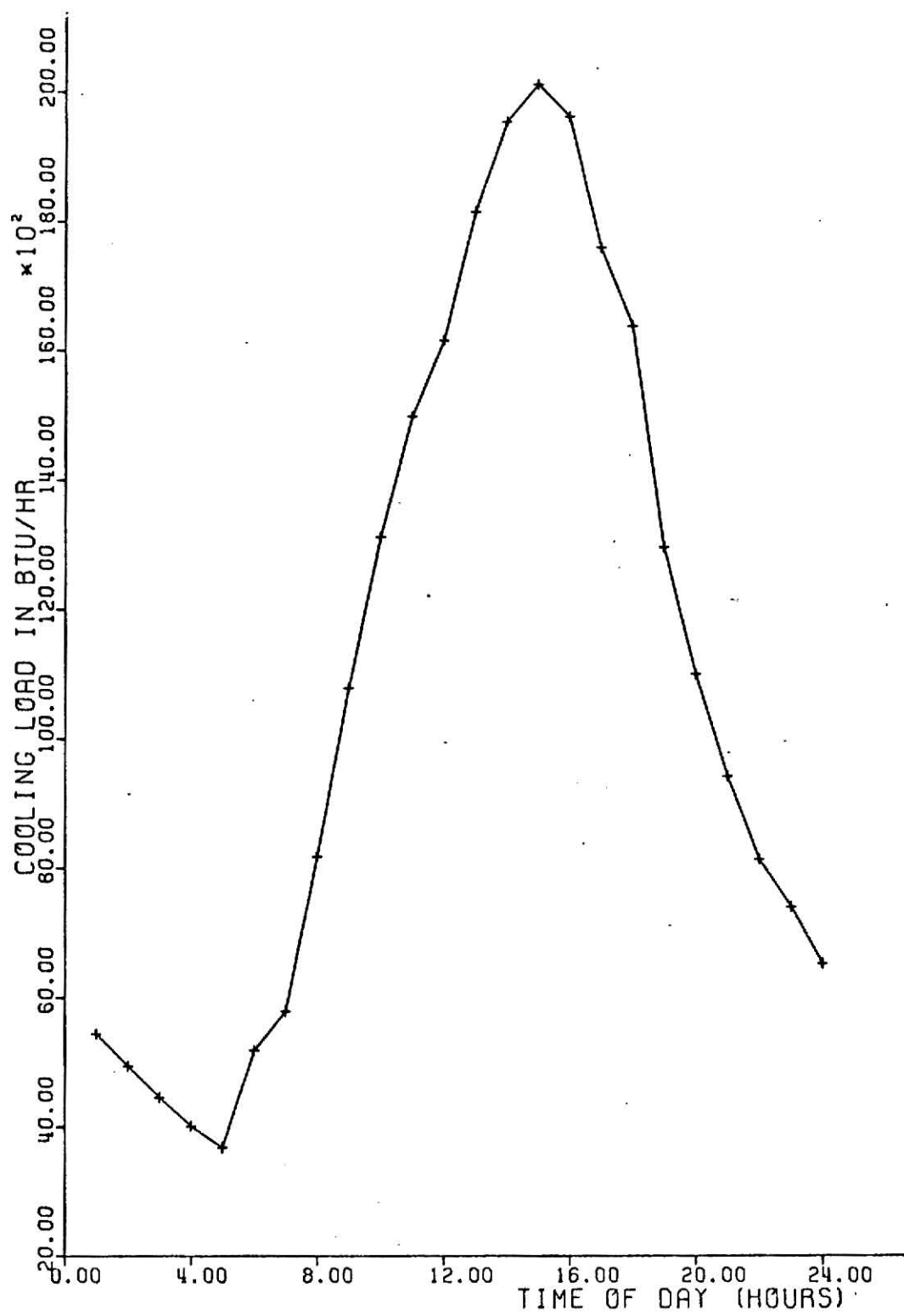


FIGURE 17. COOLING LOAD PROFILE CASE FIVE

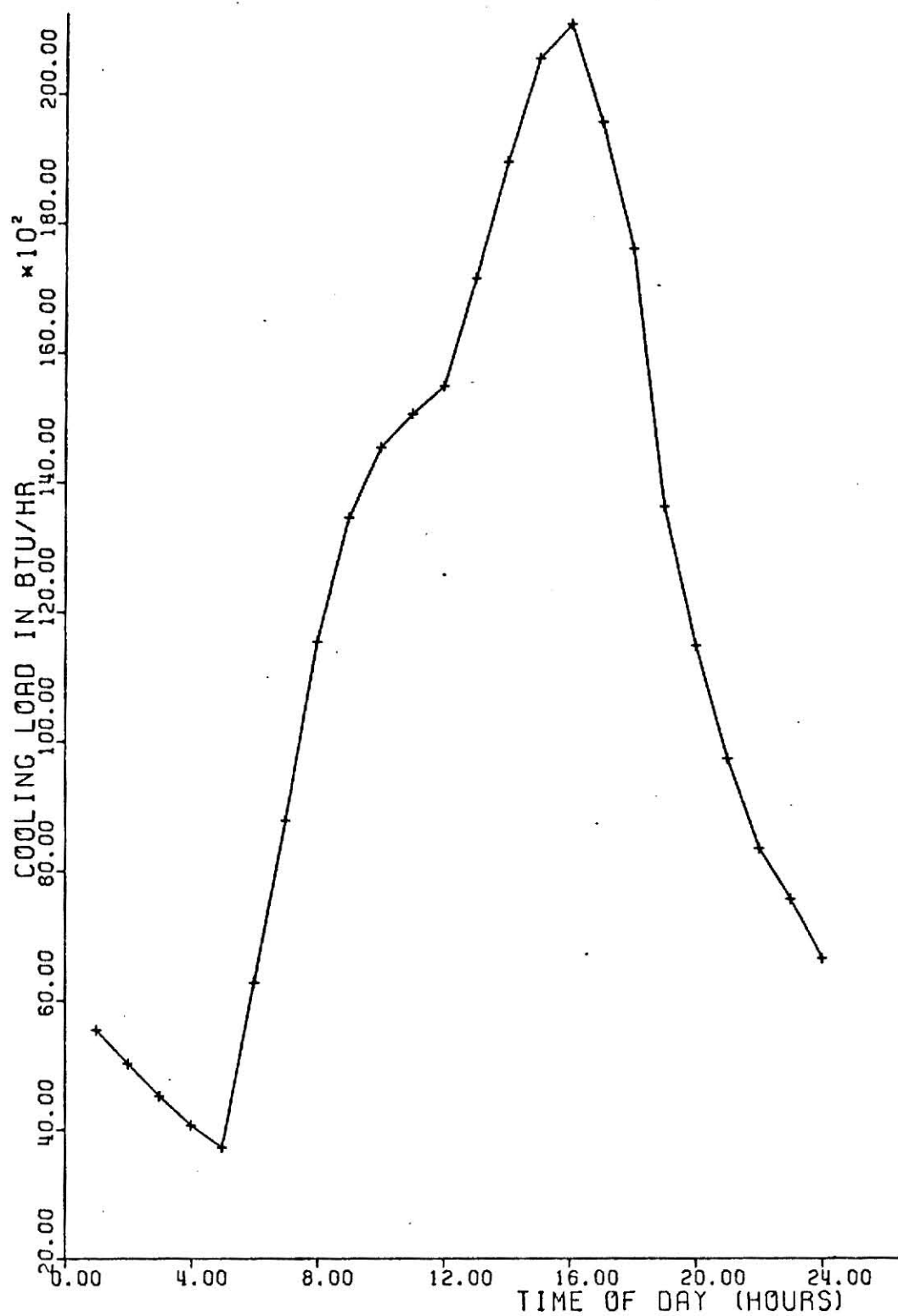


FIGURE 18. COOLING LOAD PROFILE CASE SIX

TABLE 3

HOURLY COOLING LOADS (BTU/HR) AND DAILY COOLING LOAD TOTALS (BTU/DAY)						

TIME	CASES					
OF						
DAY	1	2	3	4	5	6
*****	*****	*****	*****	*****	*****	*****
1	14427.8	9789.0	7249.2	5402.8	5445.0	5543.9
2	12100.2	8407.5	6294.9	4907.1	4941.6	5022.3
3	10070.0	7220.0	5463.3	4424.8	4453.0	4519.0
4	8355.6	6221.6	4755.0	3986.8	4009.9	4063.8
5	7087.8	5494.6	4224.4	3659.6	3678.5	3722.7
6	13042.3	11601.7	5673.5	5144.7	5185.9	6280.6
7	22743.0	17854.3	6883.9	5777.7	5796.8	8779.2
8	34658.2	25039.3	10246.4	8031.5	8176.2	11542.8
9	45634.5	31435.3	13854.3	10421.8	10784.2	13465.4
10	54161.7	35838.4	16988.2	12481.6	13127.4	14547.8
11	60029.7	38388.0	19474.7	14121.3	14985.2	15070.5
12	62984.8	38977.7	21148.5	15175.1	16162.8	15505.2
13	69601.6	43986.0	23612.8	17151.9	18154.2	17169.9
14	74460.3	47692.3	25570.0	18640.0	19544.9	18974.5
15	76327.6	49328.0	26688.2	19419.5	20126.6	20580.1
16	73440.3	47698.8	26383.9	19104.5	19626.3	21110.7
17	65340.1	42472.2	24100.7	17250.5	17599.1	19600.0
18	54022.2	35407.9	22101.1	16110.3	16391.2	17641.9
19	40851.3	25283.7	17747.2	12786.9	12972.4	13648.4
20	33696.8	21083.2	14904.6	10869.6	11005.3	11493.3
21	28111.7	17811.4	12670.6	9328.1	9430.8	9745.3
22	23596.1	15103.8	10822.7	8062.1	8142.5	8355.4
23	20166.7	13190.7	9611.1	7348.0	7412.3	7570.3
24	17048.7	11341.6	8328.0	6479.0	6531.0	6655.3
*****	*****	*****	*****	*****	*****	*****
DAILY TOTALS	911565.1	601979.1	343124.1	256260.9	263806.7	280607.8

**THE FOLLOWING
PAGES ARE BADLY
SPECKLED DUE TO
BEING POOR
QUALITY
PHOTOCOPIES.**

**THIS IS AS
RECEIVED FROM
CUSTOMER.**

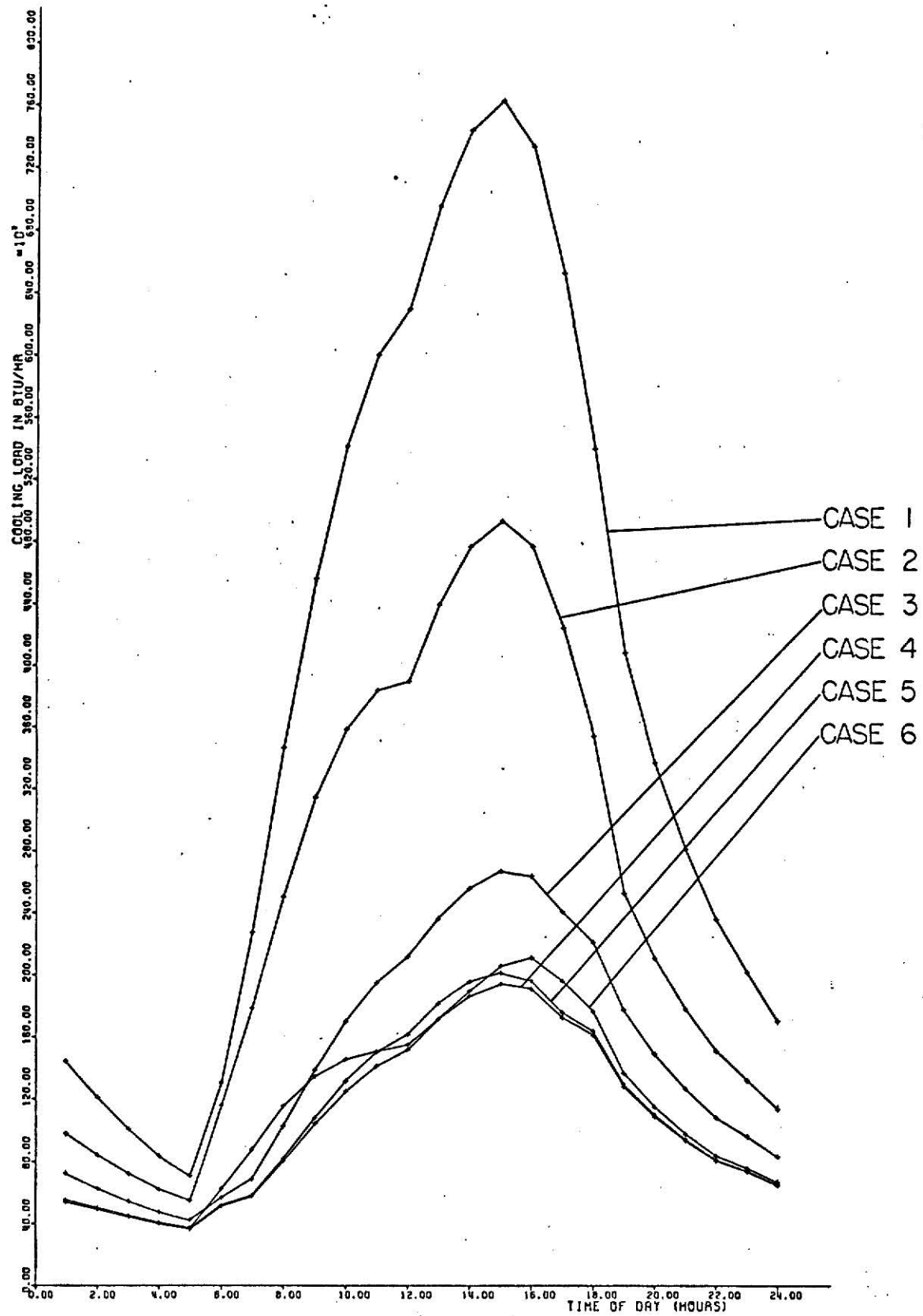


FIGURE 19. COOLING LOAD PROFILES FOR ALL SIX CASES

Case three peaked at 26,688 BTU/hr at 3 o'clock in the afternoon. This reduced the peak load by 29.7%. The daily cooling load was 343,124 BTU/day or a reduction in the base daily load of 28.4%. This case was chosen to show the effect weatherstripping and adding storm windows to windows can have on the cooling load.

The cooling load in case four was calculated for two reasons: to show the percent load reduction due to adding two inches of insulation and to establish a second base load for the last two cases. As shown in Table 3, case four peaked at 19,419 BTU/hr, a reduction of only 9.5% of the first base load. The daily cooling load was 256,260 BTU/day or a reduction of 9.5% also. This shows that the first insulation added to the house reduces the base cooling load more than additional insulation. The construction materials used in this case are very similar to those used in most Kansas homes today; therefore, this case was chosen to serve as a second base case.

Since all three dips are more distinct in case four than the others, the cause of the dips should be a predominant load. In case four the conduction loads from the walls, the ceiling, and the windows were reduced. The infiltration, latent and sensible, was also reduced. The load that has been effected the least is the radiation through transparent surfaces; therefore, this apparently is the cooling load component that is causing the dips in the cooling load profiles.

The first four cases were chosen to demonstrate the program's ability to calculate cooling loads for various constructions, while the last two cases were chosen to demonstrate the shading routine.

Case five was constructed of the same materials as in case four except the width of the overhangs were reduced by 50%. Case five's peak cooling load increased by 3.6% over the base case four. The total daily

cooling load increased by 2.9%. Figure 19 illustrates the similarity in cooling load profiles for cases four and five. A change in the overhang width appears to cause only a magnitudinal shift.

When the front of the house was rotated to the west, more than a magnitudinal shift was shown as depicted in Figure 19. For case six the cooling load peaked at 21,110.7 BTU/hr and at 4 o'clock in the afternoon, one hour later than the first five cases. This was an increase of 8.7% in the peak load over case four. The daily load increased by 9.5%. The hour shift occurred because of the larger western exposure later in the day.

After obtaining a cooling load profile, home designers must be able to analyze the results and draw conclusions from it. They must be able to recognize when a cooling load calculation may be too high or too low and make the appropriate modifications.

The first example of this is if the house was located on a high hill overlooking a large lake. The structure would probably have a higher infiltration load than those used in the program due to the higher winds and higher outside humidity ratio. Program adjustment would be required to reflect this situation.

Another example is a home located in a metropolitan area with a hazy atmosphere. This would lower the solar load.

A third example is an attic with some type of forced ventilation. Since the program considered the attic as enclosed and not ventilated, a forced ventilation would change the attic temperature and, therefore, change the ceiling load.

The calculated cooling load is based on an inside design temperature remaining at 75 F. Since thermostats activate with a rise in inside temperature, the cooling load will drop with that rise. The home designer will

need to consider each individual design and decide which load minimizing changes will save money.

CHAPTER V

SUMMARY

Home designers have used several different methods to calculate cooling loads for residential homes. Most of these methods give the designer a peak load only, and tell nothing about the cooling load at any other time of the day. With a twenty-four hour cooling profile, the total cooling load per day can be calculated. This is very helpful in selecting building materials, width of overhang, the orientation of the house and in estimating cooling costs.

The digital computer was used in calculating the twenty-four hour cooling loads for a residential structure. The program was written to calculate cooling loads for as many home types and structural variations as possible. However, the following limitations exist: there must be the same number of floors throughout the floor plan; all extensions from the main house are required to be at right angles; all of the roof hip ridges are required to meet at the same level. The program can calculate cooling profiles for L, T, U, or X shaped homes.

The shape of the home and its dimensions, the type of wall, ceiling, window and door construction are needed as input data. This information completely defines the home for the program calculation. The outside environmental information required in the program is solar heat gain calculated for different months of the year, the angle of house-sun orientation, and the supplied outside temperature profile.

The program with this input data has the capability of accommodating various construction materials, overhang widths, and angles of house-sun orientation. With this information, detailed shading calculations are

made to calculate what percentage of each surface is sunlit. This shading information is used to calculate adjusted sol-air temperatures and adjusted solar heat gains.

Heat gains from the walls and ceilings are calculated using the response factor method outlined in the ASHRAE Handbook of Fundamentals. The response factor method uses a time series combination of past sol-air temperatures and past heat gains to calculate the present heat gain.

Infiltration was programmed by the crack flow method for different types of windows; therefore, the infiltration rate is fixed by the type and size of window.

For the windows and doors, a sensible and latent infiltration load is calculated by the program.

The internal loads calculated were due to lights and people. Both were estimated from reasonable schedules for lighting and family activity.

Heat gains from the walls, ceiling, doors and windows are changed to cooling loads by another response factor method outlined in the ASHRAE Handbook of Fundamentals. These cooling loads are combined with the infiltration and the internal loads to obtain the total cooling load at each hour.

Six cases were run to demonstrate the program's ability to calculate the cooling load profiles for different situations. The first four cases used the same basic floor plan, but changed the construction of the walls, ceiling, doors and windows. The first case was an uninsulated home with single pane, unweatherstripped windows. As the program progressed to case four, there was progressively added more insulation and weatherstripped windows. Case four had six inches of insulation in the ceiling, three

inches of insulation in the walls, insulated glass and weatherstripped windows and doors. In the first four cases the building materials only were varied to demonstrate their effect on cooling load. In case five the building materials were left the same as in case four, but the width of the overhang was cut in half to demonstrate the effect shading would have on the cooling load. Again, in the sixth case, the building materials of case four were used along with three foot overhangs, but the home was rotated 90° from facing south to facing west. This demonstrated the program's ability to calculate the cooling load at any other orientation. The results from cases one through six can be observed in Figures 13 through 18 and in Table 3.

It is hoped this program will find practical use. The responsibility of using this program rests with the home designer. To save money on initial and monthly air conditioning costs, the home owner must demand that the home designer seek the information this cooling load program can provide. The program can be an aid in making the economic and energy decisions which must be made for future single-family housing units.

BIBLIOGRAPHY

1. ASHRAE Handbook of Fundamentals, American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc., 345 East 47th Street, New York, N. Y., 1972.
2. Mitalas, G. P., and J. G. Arseneault, Fortran IV Program to Calculate Transfer Functions for the Calculation of Transient Heat Transfer Through Walls and Roofs, National Research Council of Canada, Division of Building Research, Ottawa, Canada, 1972.
3. Mitalas, G. P., and D. G. Stephenson, "Room Thermal Response Factors," ASHRAE Transactions, Vol. 73, Part I, 1967.
4. Stephenson, D. G., and G. P. Mitalas, "Cooling Load Calculations by Thermal Response Factor Method," ASHRAE Transactions, Vol. 73, Part I, 1967.
5. Threlkeld, James L., Thermal Environmental Engineering, Prentice-Hall, Englewood Cliffs, New Jersey, 1970.

APPENDIX A

SHADING EXAMPLES

The following examples will not and are not intended to show how each shading case is calculated in the program, but to demonstrate the general theory used to calculate a few representative examples. The examples are:

- a. The shading on a wall from the overhang extending out over it.
- b. The wall shading from the overhang extending out over it in combination with the overhang from an adjacent wall that extends out perpendicular from the original wall.
- c. The wall shading that occurs with the roof overhanging this wall in combination with the shading from a roof's hip ridge that extends out perpendicular from the wall.
- d. The window shading from a roof overhang above it in combination with the overhang from a perpendicular wall.

There are several common variables that are required for every case. Some of these are supplied to the computer as data and others are calculated. Some of these calculated values are the solar altitude, the solar azimuth, and the shadow-line angle which is the solar altitude perpendicular to the wall. The variables that are supplied to the computer will be identified when they are used in the examples.

The first example is a plane wall with the roof overhanging the edge of the wall. Figure 20 shows the type of house and the shading example.

Figure 21 shows where the shadow-line angle (Ω) is located. Variables O and H , the width of overhang and the height of the wall, are values supplied to the computer as data. To calculate the percent sunlit area, the following equations are used:

$$\text{UNSHA} = 1.0 - (O \cdot \tan \Omega)/H \quad \text{if } \tan \Omega = \tan \beta / \cos \gamma.$$

Then,

$$\text{UNSHA} = 1.0 - (O \cdot \tan \beta)/(H \cdot \cos \gamma) \quad \text{and } \gamma = \theta - \psi$$

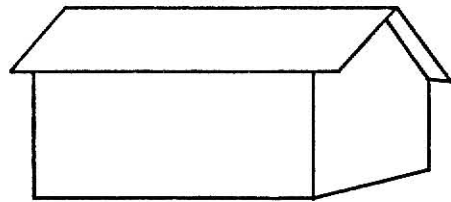


FIGURE 20. HOUSE USED FOR EXAMPLE ONE

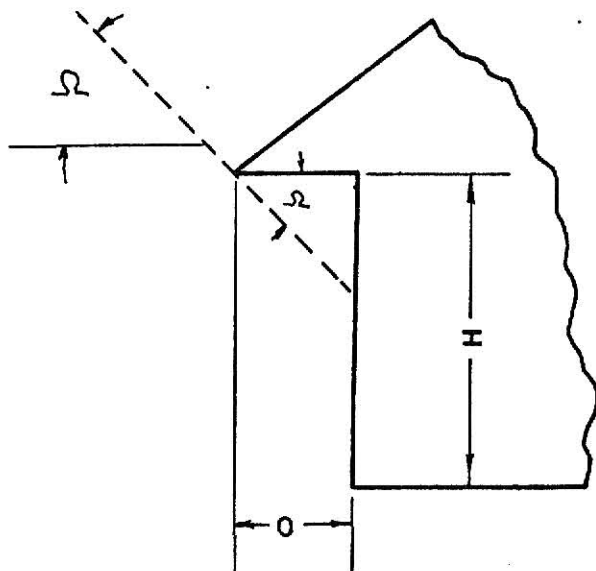


FIGURE 21. EDGE VIEW OF WALL

where,

UNSHA - The percent of the wall area sunlit.

O - Width of overhang, ft.

H - Height of wall, ft.

β - Solar altitude, rad.

γ - Wall-solar azimuth, rad.

$\emptyset$ - Solar azimuth, rad.

ψ - Wall azimuth, rad.

If the absolute value of γ should become larger than ninety degrees, the wall is shaded entirely. This is true for wall, window and door shading.

Example two is similar in many respects, but the wall extending out perpendicular to it causes a diagonal shadow across the wall as illustrated in Figure 22.

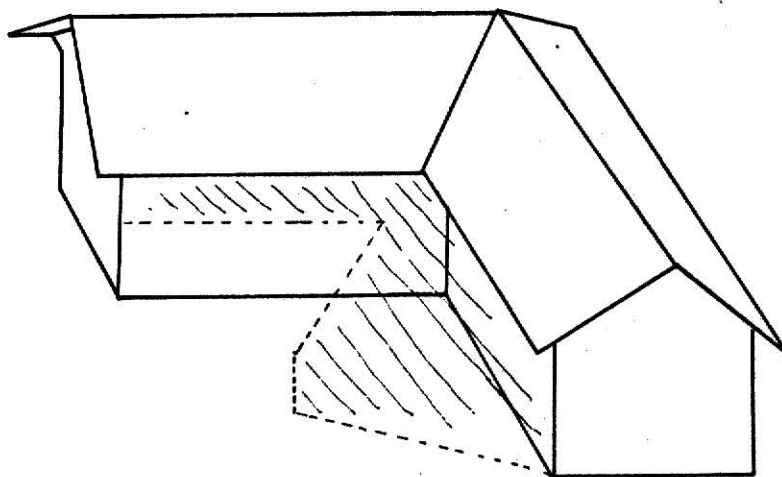


FIGURE 22. HOUSE AND SHADING FOR
EXAMPLE TWO

Figure 23 shows a working view of the house for computational purposes.

The notation will need to be explained before continuing. The wall that is being shaded will be subscripted with an n and the wall counter-clockwise looking at the floor plan from above will be subscripted $n - 1$.

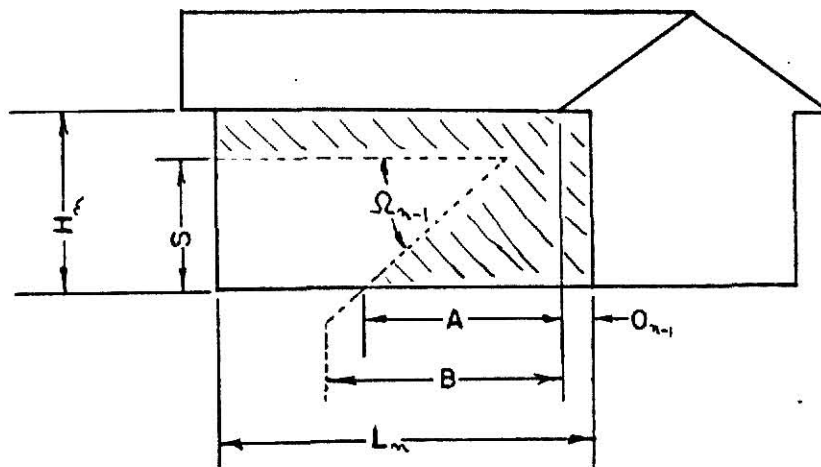


FIGURE 23. EDGE VIEW EXAMPLE TWO

Likewise, a wall clockwise from the n th wall will be subscripted $n + 1$.

This particular example can only occur when γ is less than zero and greater than -90 , and if Ω_{n-1} is greater than the slope of the roof on wall $n - 1$.

To calculate the percent of the wall area sunlit, divide the sunlit area into two areas, X and Y , as shown in Figure 24. Then these areas can be calculated with the following equations:

$$X = (L_n - A - O_{n-1}) \cdot S$$

$$S = \text{UNSHA} \cdot H_n$$

or

$$X = (L_n - A - O_{n-1}) \cdot \text{UNSHA} \cdot H_n$$

and

$$Y = S \cdot S / (2 \cdot \tan \Omega_{n-1}) = (\text{UNSHA}^2 \cdot H_n^2) / (2 \cdot \tan \Omega_{n-1})$$

$$\text{WAREA} = (X + Y) / (L_n \cdot H_n)$$

where,

L_n - Length of wall n , ft.

O_{n-1} - Overhang from the wall, ft.

UNSHA - Percent of wall_n sunlit.

H_n - Height of wall_n, ft.

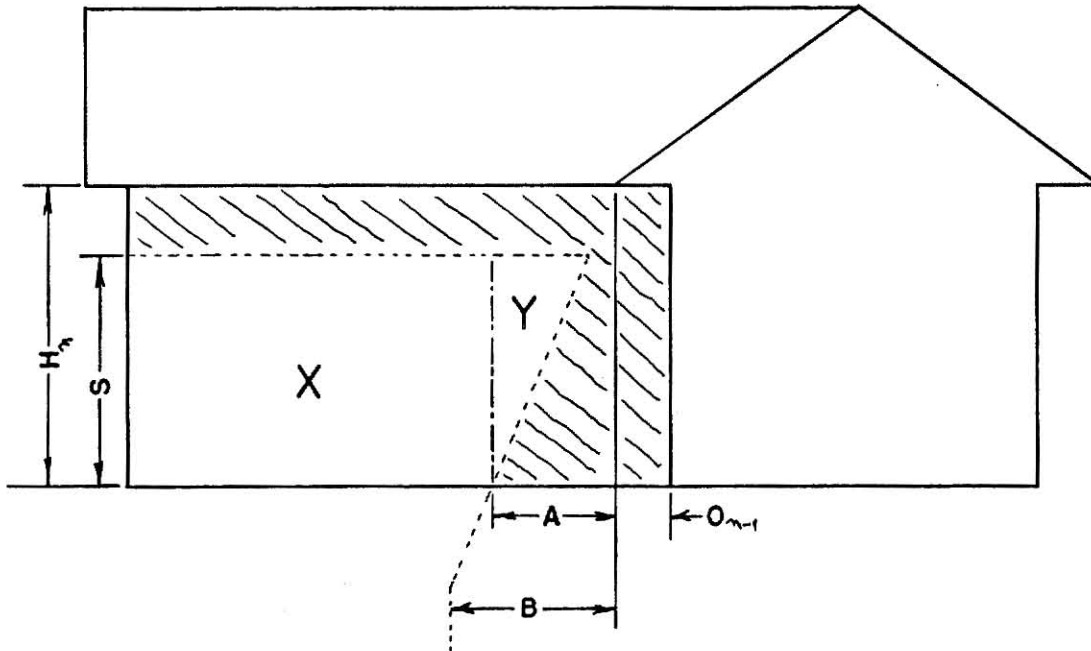


FIGURE 24. EXAMPLE TWO AREAS X AND Y

The values of H , L , and O are supplied to the computer as data. Check statements are used to determine if the home even has this type of shape. When a building is found that is shaped this way, it is then checked to see if the sun is in a position to give the wall this type of partial shading. If so, then it is checked to see if the quantity A or B from the Figure 23 is larger. If B is larger than A and A is smaller than $L_n - O_{n-1}$, the previous example has been found.

Example three is similar to example two except, O_{n-1} has fallen below the slope of the roof on wall_{n-1}. A working drawing of this problem is provided in Figure 25.

Again the sunlit area is divided into two shades, X_1 and Y_1 . X_1 and Y_1 are calculated as follows:

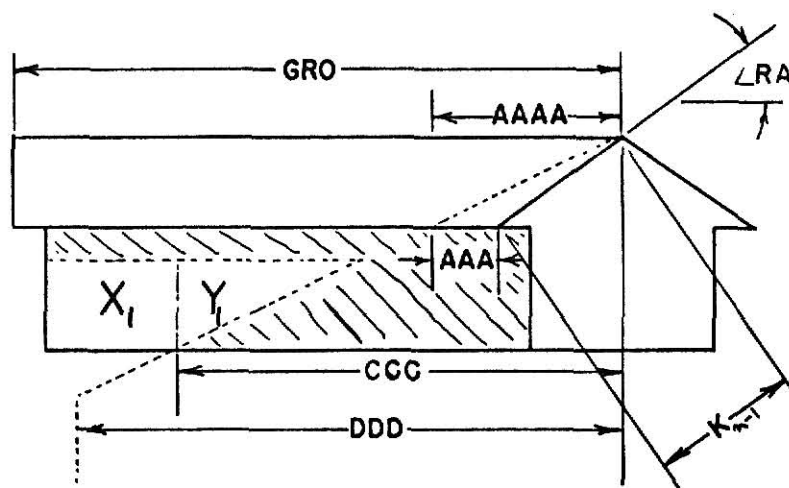


FIGURE 25. SHADING EXAMPLE THREE

$$X_1 = (GRO - c c c) (UNSHA \cdot H_n)$$

where,

$$GRO = K_{n-1} \cdot \cos (< RA) + L_{n-1} - O_{n-1}$$

$$c c c = (K_{n-1} \cdot \sin (< RA) + H_{n-1}) / \tan(\Omega_{n-1})$$

$$Y_1 = (UNSHA \cdot L_n) / (2 \cdot \tan(\Omega_{n-1})) \cdot (UNSHA \cdot L_n) = UNSHA^2 \cdot L_n^2 / (2 \cdot \tan(\Omega_{n-1}))$$

$$WAREA = (X_1 + Y_1) / (L_n \cdot H_n).$$

The same variables as in example two, plus K and $< RA$, were supplied to the computer as input data. Similar check statements are used to find this example as those used in example two.

Example four is different from the first three examples to provide an example of window shading. Many of the same concepts are used in this example that were used in the first three examples. Figure 26 illustrates shading example four. With the aid of Figure 27, the percent area sunlit of the window is calculated more easily. For this case the sunlit area is divided into two areas, X_2 and Y_2 . The areas of X_2 and Y_2 are calculated as follows:

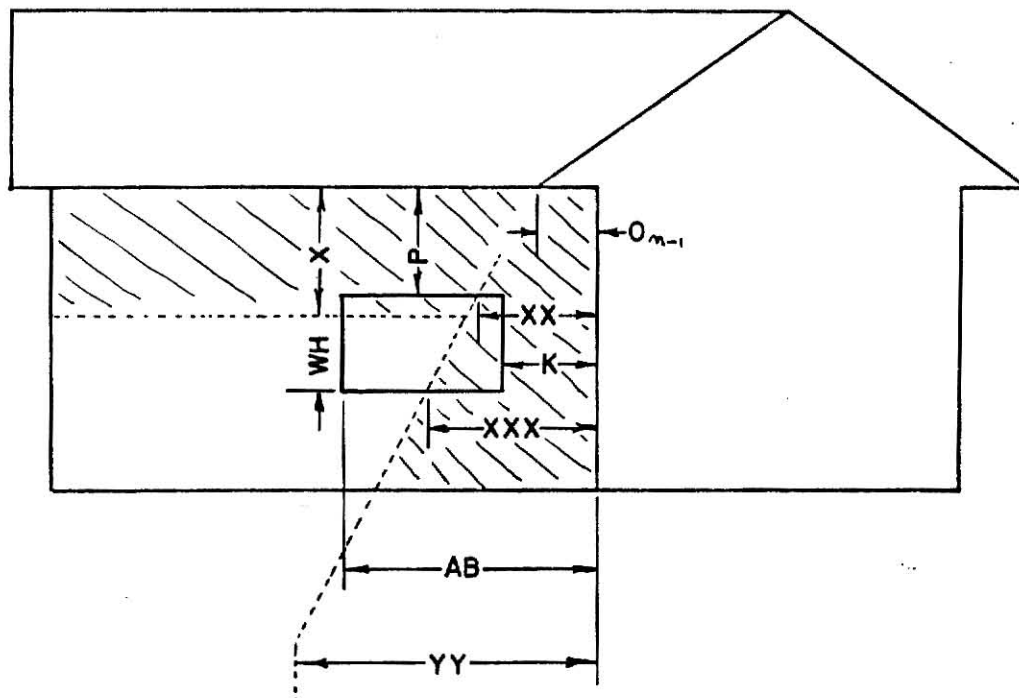


FIGURE 26. VIEW OF WINDOW IN EXAMPLE FOUR

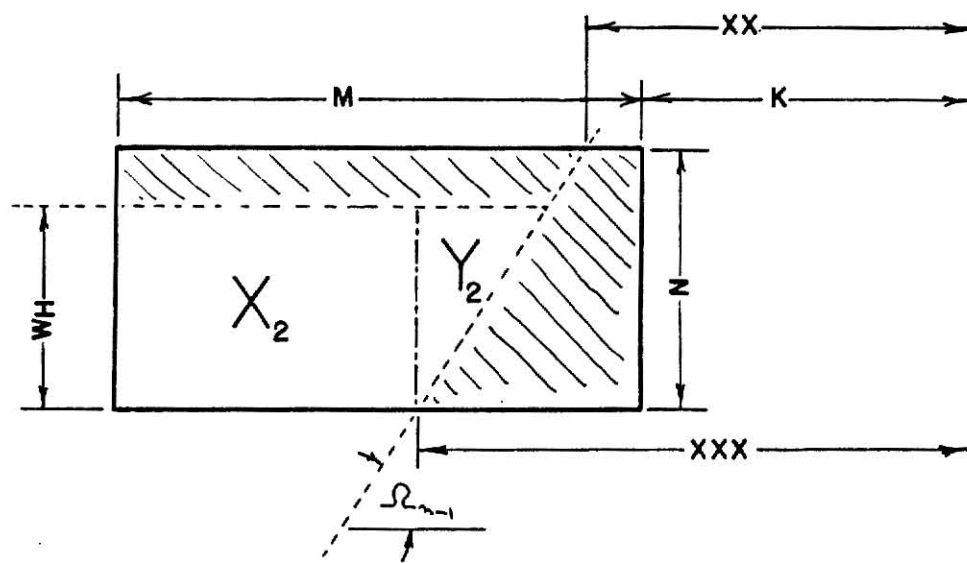


FIGURE 27. WINDOW ENLARGEMENT

$$X_2 = (AB - XX) \cdot WH$$

$$Y_2 = WH \cdot (WH / (2 \cdot \tan \Omega_{n-1})) = (WH^2) / (2 \cdot \tan \Omega_{n-1})$$

where,

$$XX = O_{n-1} + P / \tan \Omega_{n-1}, \text{ ft.}$$

$$XXX = O_{n-1} + (P + N) / \tan \Omega_{n-1}, \text{ ft.}$$

$$AB = K + M, \text{ ft.}$$

$$WH = P + N - O_n / \tan \Omega_n, \text{ ft.}$$

and

$$WAREA = (X_2 + Y_2) / (N \cdot M).$$

WAREA - Percent area of windows sunlit.

For this example P, N, M, K, and O are supplied to the computer as data for wall_n and wall_{n-1}. All other windows and door types are calculated on a similar basis.

When the program is required to find what percent of a surface is shaded, it must first find what type of shading can exist on that surface and then determine where the shadow line or lines cross it. In these examples, the preliminary steps were bypassed and that information was completely defined. The computer in case four calculated XX and XXX and determined if the shadow line even passes across the window. After the case has been identified by the program, then the equation to calculate the percent sunlit area is selected for that particular case.

APPENDIX B

THE ATTIC AIR TEMPERATURE CALCULATIONS

The heat gain through the roof and ceiling constitutes a large percentage of the total load; therefore, it is an important load to consider. As the program was being written a number of problems arose concerning the calculation of these heat gains. Some of these problems were: which sol-air temperatures should be used on the roof; how was the mixing of air in the attic occurring due to some roofs in the sun light and others in the shade; and which response factors should be used and how would these be calculated. A further explanation of these is needed.

The first problem concerns the difference in sol-air temperatures of the roofs. The sol-air temperatures on a roof can range in magnitude by as much as 50 F at one hour. The second problem is a function of the first problem. What is the mean attic temperature if the sol-air temperature of one roof is at 150 F and a second roof is at 95 F? There must be some kind of mixing of the attic air occurring. The third problem again builds on the previous problems.

It was thought at first that response factor coefficients could be calculated for a ceiling with an attic space above it and do a heat gain analysis through each roof. This satisfies problem one, but aggravates the second one. The solution implied that the attic air temperature under each roof was completely independent of the attic air temperature under any other roof. This eliminated the response factor method to obtain heat gains from sol-air temperatures on the roof for the ceiling. At this point energy balances were written in an attempt to calculate an attic temperature profile. The following derivation will show how these

equations were derived and how they are used to calculate a heat gain profile through the ceiling.

To start with, a positive direction for heat transfer is assumed or, equivalently the sol-air temperature on the roof is higher than the attic temperature and the attic temperature is higher than the inside design temperature. If these magnitudes are not correct, the sign of the heat flux will take care of the discrepancy. In Figure 28, a cross sectional view of an attic and the energy terms involved are shown.

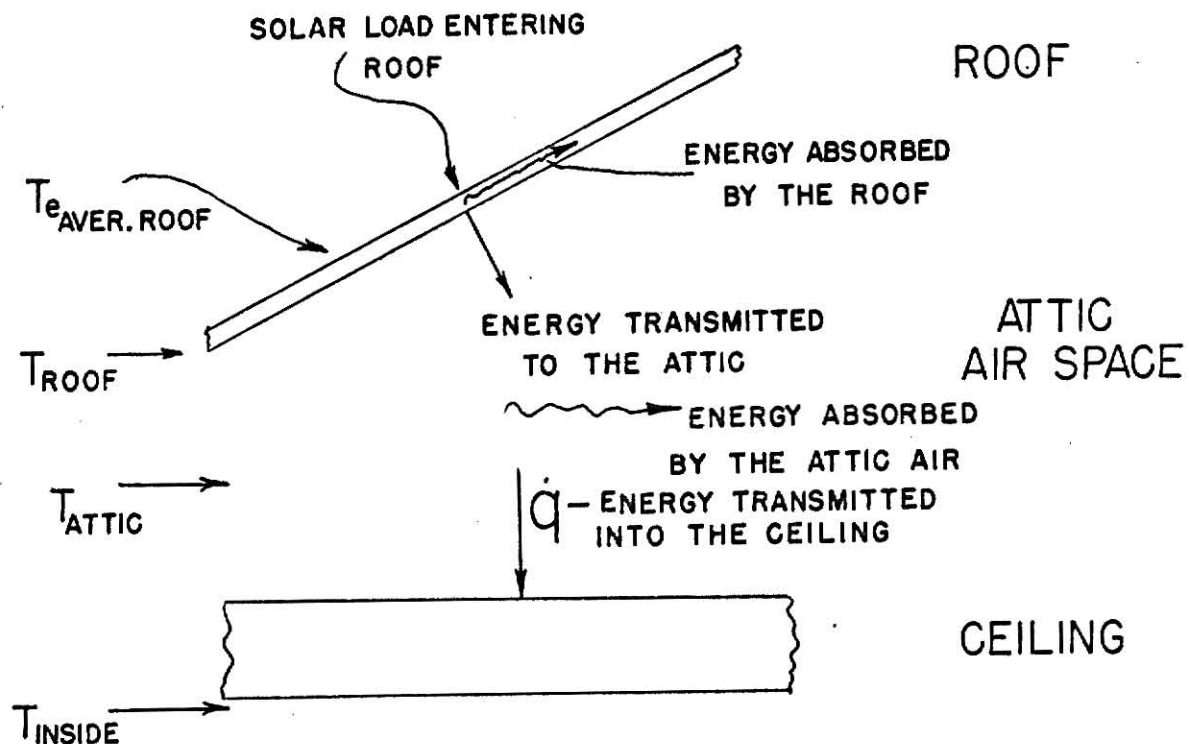


FIGURE 28. CEILING CROSS SECTION

The energy terms can be combined as the solar load entering the roof is equal to the energy absorbed by the roof plus the energy transmitted to the attic for the first equation, and the energy transmitted to the attic is equal to the energy absorbed by the attic air plus the energy transmitted into the ceiling for the second equation. The energy rates transmitted into the ceiling are generated from a response factor method; therefore,

the heat flux into the ceiling does not need to be broken down any further.

These expressions are written as:

$$A_R h_o (T_{e \text{ aver. roof}} - T_{\text{roof}}) = M_R C_R dT_{\text{roof}}/dt + A_R h_a (T_{\text{roof}} - T_{\text{attic}})$$

$$A_R h_a (T_{\text{roof}} - T_{\text{attic}}) = M_{\text{attic}} C_{\text{attic}} dT_{\text{attic}}/dt + q_{\text{ceiling}}$$

where,

$$T_{e \text{ aver. roof}} = \frac{\sum_{n=1} A_{R_n} \cdot T_{e \text{ roof}_n}}{\sum_{n=1} A_{R_n}}$$

A_R - Area of the roof, ft².

h_o - Outside convection coefficient, BTU/hr-ft²-F.

$T_{e \text{ aver. roof}}$ - Area weighted sol-air temperature on the roof, F.

T_R - Temperature of the roof materials, F.

M_R - Mass of the roof, lbs.

C_R - Heat capacity of the roof, BTU/lbs - F.

t - Time, hrs.

h_a - Inside attic convection coefficient, BTU/hr-ft²-F.

T_{attic} - Attic air temperature, F.

M_{attic} - Mass of attic air, lbs.

C_{attic} - Specific heat of air, BTU/lbs-F.

q_{ceiling} - Heat gain through the ceiling, BTU/hr.

Since these equations are not programmable in this form, they need to be transformed into equations that are programmable. First, dT/dt is written as $(T^* - T)/\Delta t$ where T^* is some temperature Δt time in the future before T occurs. To make this more general it is written for any time increment in hours as $(T^* - T)/\frac{1}{n}$ or $n(T^* - T)$. The time interval between calculations is $1/n$ hours. The above equations are written as:

$$A_R h_o (T_{e \text{ aver. roof}} - T_{\text{roof}}) = n M_R C_R (T^*_{\text{roof}} - T_{\text{roof}}) + A_R h_a (T_{\text{roof}} - T_{\text{attic}})$$

and

$$A_R h_a (T_{\text{roof}} - T_{\text{attic}}) = n M_{\text{attic}} C_{\text{attic}} (T^*_{\text{attic}} - T_{\text{attic}}) + q_{\text{ceiling}}$$

where,

T^* - is the future temperature by $1/n$ hours.

There first equation is solved for T^*_{roof} and the second equation for T^*_{attic} . These will yield the following equations:

$$T^*_{\text{roof}} = T_{\text{roof}} + (A_R h_o (T_{\text{e aver. roof}} - T_{\text{roof}}) - A_R h_a (T_{\text{roof}} - T_{\text{attic}})) / (n M_r C_r)$$

and

$$T^*_{\text{attic}} = T_{\text{attic}} + (A_R h_a (T_{\text{roof}} - T_{\text{attic}}) - q_{\text{ceiling}}) / (n M_{\text{attic}} C_{\text{attic}})$$

where,

$$h_o = 4.0 \text{ BTU/hr-ft}^2 \text{ F.}$$

$$h_a = 0.4 \text{ BTU/hr-ft}^2 \text{ F.}$$

Since the approximation for dT/dt may not be convergent for large one hour time intervals, the size of n may need to be changed. By trying different values it was found for residential roof masses that n needs to be at least four or five for convergence to occur; therefore, n was chosen as ten to insure convergence. Since n was chosen as ten, an average sol-air temperature profile for the roofs was needed that gave sol-air temperatures with a time interval of six minutes between them, or ten per hour.

A linear interpolation between the hour average sol-air temperatures was used. To get these calculations started, an attic temperature of 85 F was chosen, 75 F was used as the inside design temperature and 100 F was chosen as the roof temperature. These values were used for the last six hours of the day and they were used to calculate the q for the last six hours of the day by the equation:

$$\dot{q}_{\text{ceiling}} = A_{\text{ceiling}} U_{\text{ceiling}} (85 \text{ F} - 75 \text{ F}).$$

During the first hour that T_{roof}^* and T_{attic}^* were calculated, the values for T_{roof} , T_{attic} , and $T_{\text{e aver. roof}}$ from one time interval before were used and the initial $\dot{q}$ was used. At the end of the first hour a new value of $\dot{q}$ was calculated by the response factor equation. This new $\dot{q}$ was used for the second hour along with T_{roof} , T_{attic} and $T_{\text{e aver. roof}}$ for the first time interval before to calculate T_{roof}^* and T_{attic}^* . At the end of the second hour another new $\dot{q}$ was calculated to be used during the third hour. This procedure was repeated every hour for five twenty-four hour periods to insure the convergence of a solution for the attic temperature and $\dot{q}$ profiles. When this procedure has been completed by the program, in addition to the attic temperature profile being calculated, the ceiling heat gains have also been calculated.

APPENDIX C

For the program user a further explanation of the input to this program is needed. The following discussion will show a sample input to the program including drawings of the house, the data cards necessary to run the program, and a discussion of the information on these cards.

The home used in the sample data is fully described in Figures 29-33. The necessary data is punched on cards shown in Figures 34 and 35. In Table 4 is a tabular description of each card is found.

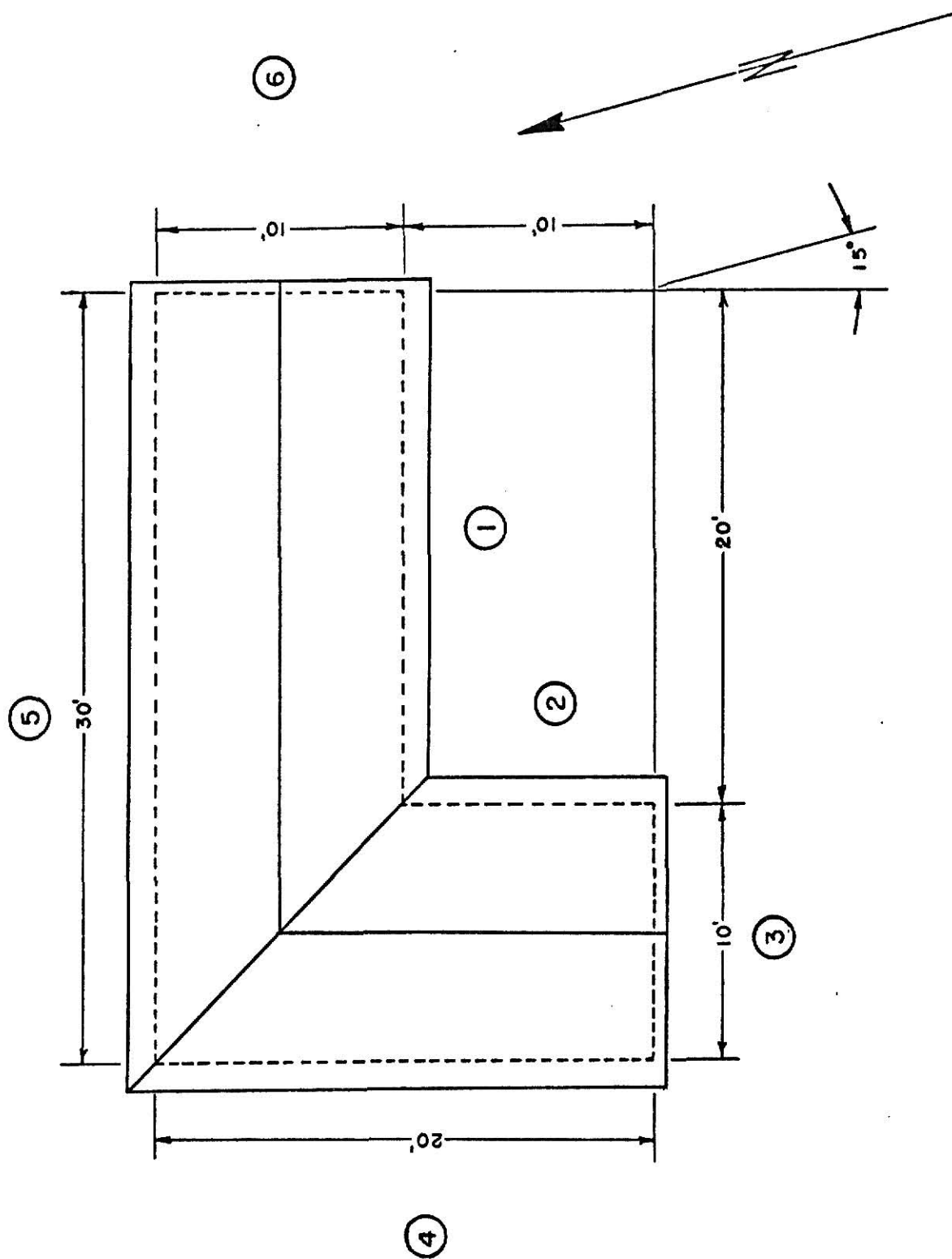


FIGURE 29. TOP VIEW OF HOUSE FOR SAMPLE DATA

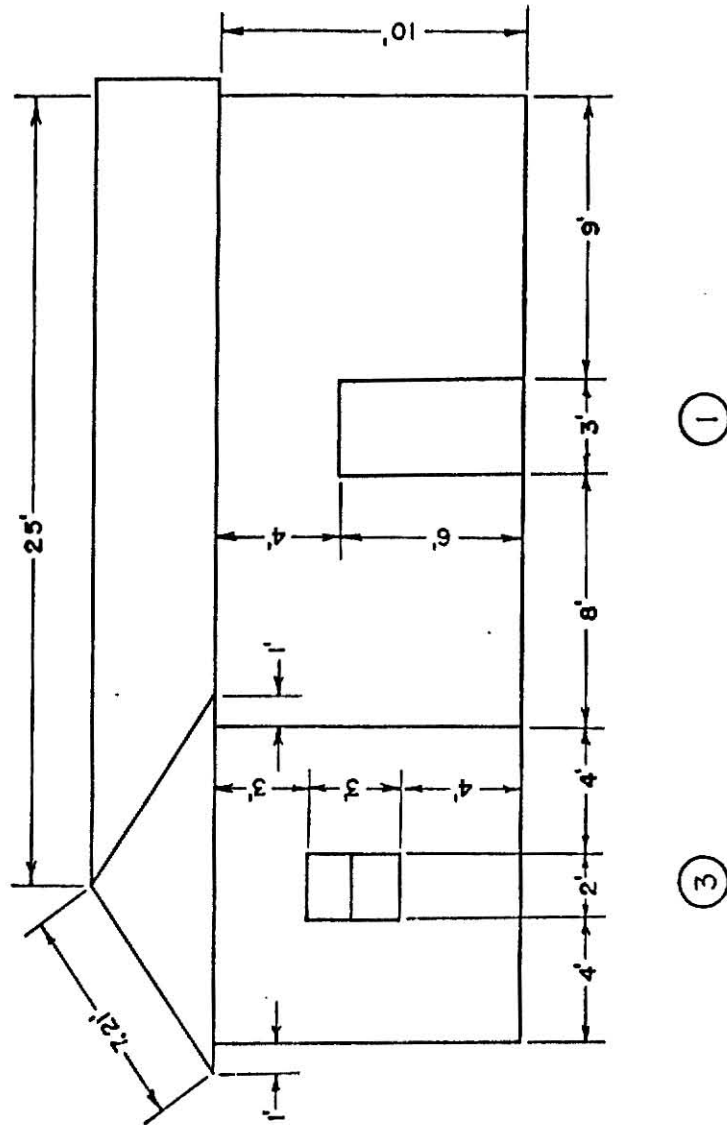


FIGURE 30. SIDE VIEW OF WALLS ③ AND ④

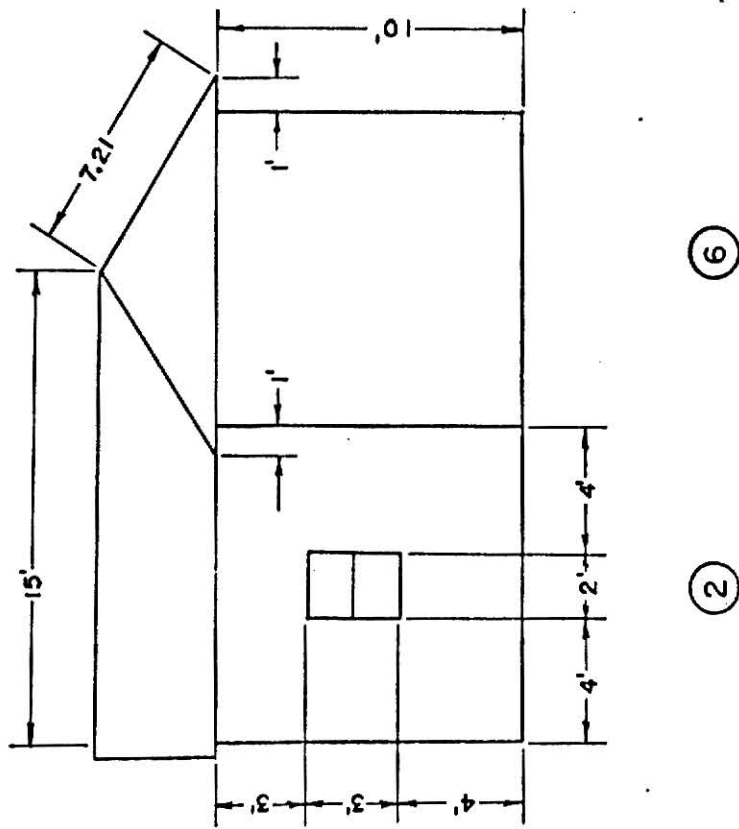


FIGURE 31. SIDE VIEW OF WALLS ② AND ⑥

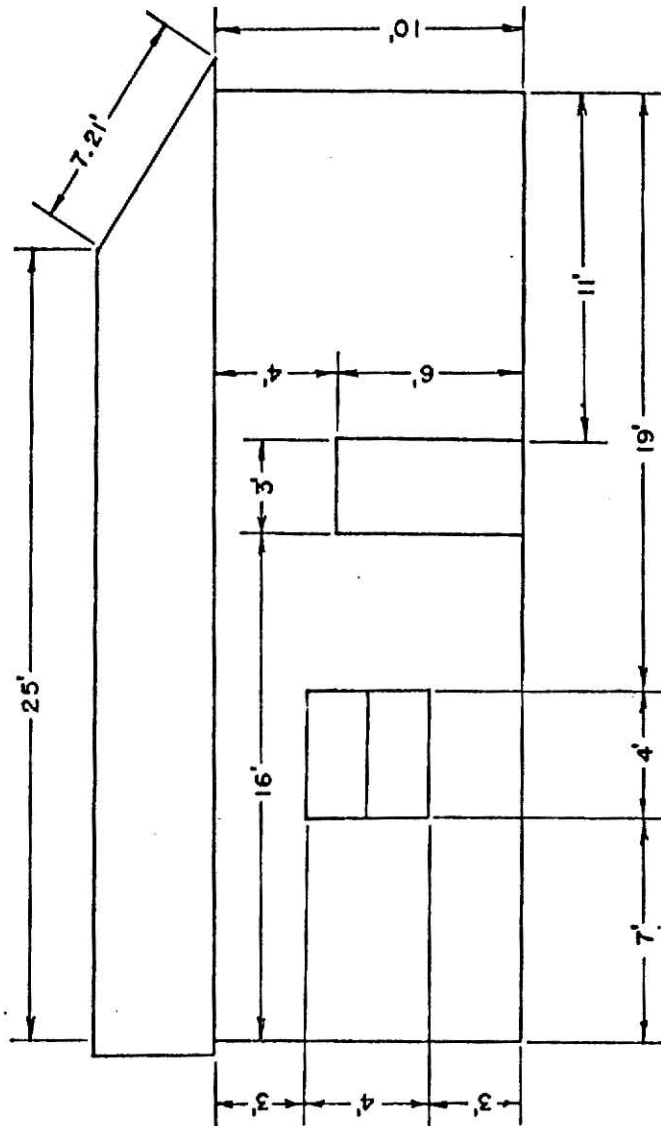
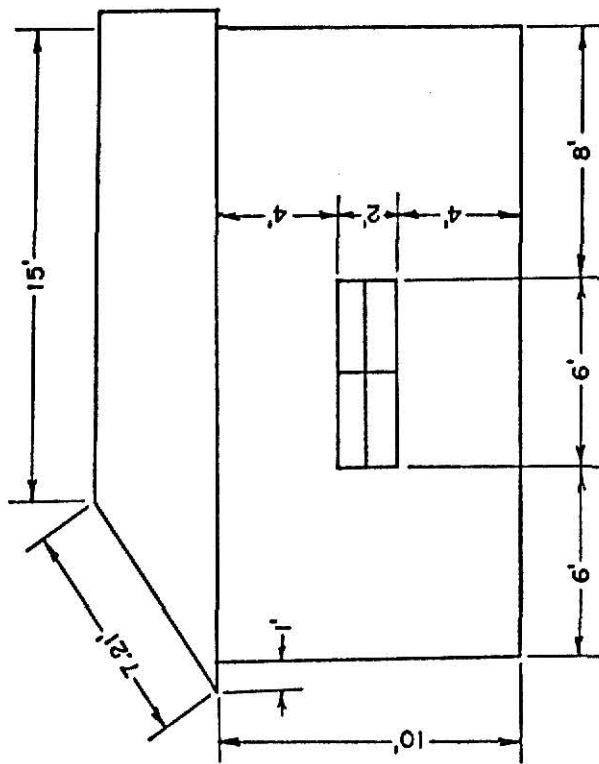


FIGURE '32. SIDE VIEW OF WALL (5)



④

FIGURE 33. SIDE VIEW OF WALL ④

ILLEGIBLE DOCUMENT

**THE FOLLOWING
DOCUMENT(S) IS OF
POOR LEGIBILITY IN
THE ORIGINAL**

**THIS IS THE BEST
COPY AVAILABLE**

TIME OF DAY (HOURS)

Literal Field For XA(J)

COOLING LOAD IN BTU/HR

Literal Field For YB(J)

Literal Field For ZC(J)

No. of Subroutines Called

1st Sub. Called 2nd Sub. Called 3rd Sub. Called 4th Sub. Called 5th Sub. Called 6th Sub. Called 7th Sub. Called 8th Sub. Called

9th Sub. Called 10th Sub. Called 11th Sub. Called 12th Sub. Called

No. of Walls Angle of Orientation Month of Year Latitude

1.0	.66	20.0	10.0	9999.00	200.00	158.62	1.0	7.21	25.0
4.0	.66	10.0	10.0	9999.00	100.00	86.52	1.0	7.21	15.0
1.0	0.0	10.0	10.0	9999.00	100.00	0.0	0.0	0.0	0.0
2.0	.66	20.0	10.0	9999.00	200.00	129.78	1.0	7.21	15.0
3.0	.66	30.0	10.0	9999.00	300.00	201.88	1.0	7.21	25.0
4.0	0.0	10.0	10.0	9999.00	100.00	0.0	0.0	0.0	0.0

Wall Orientation Number Slope of Roof Length of Wall Height of Wall Wall Slope Wall Area Roof Area Width of Roof Overhang Width of Roof Length of Hipridge

FIGURE 34. DATA CARDS 1-13

[illegible]

FIGURE 35. DATA CARDS 14-27

TABLE 4
DESCRIPTION OF DATA CARDS

Card No.	Description
1.	This is a literal field of length 40 spaces. This field is used in the Cal Comp Plotter Subroutines (Horizontal Axis). Anything can be put in the first 40 spaces, this card can even be left blank.
2.	This is a literal field of length 40 spaces. This field is used in the Cal Comp Plotter Subroutines (Vertical Axis). Anything can be put in the first 40 spaces, this card can even be left blank.
3.	This is a literal field of length 40 spaces. This field is used in the Cal Comp Plotter Subroutines (Legend). Anything can be put in the first 40 spaces, this card can even be left blank.
4.	This is the number of subroutines called. To make this program more versatile each of the subroutines are called by an identification number. A listing of these numbers is contained in the listing of the main program along with the minimum subroutines necessary to obtain each cooling load. The main program uses a series of logical IF statements to call the next subroutine.
5 & 6.	These two cards are a list of subroutines in the order they are called. The number of subroutines called determines how many cards will be used. Note 8 subroutines are listed on each card. This example calls 12 subroutines, 10 could have been called and a cooling load still obtained if subroutines (3) (Shafra) and (7) (Vallo) would not have been called. These two subroutines' sole function is to print out intermediate data.
7.	This is the first data card read in subroutine reval. The first value read is the number of walls on the house. The second value is the house-sun orientation angle (positive-clockwise-negative-counter clockwise). The third value is the month of the year. The fourth value is the latitude location of this home.
8,9,10, 11,12 & 13.	These are wall data cards. Each card contains the information for one wall. There will be a card for each wall. On each card the first value is the wall orientation number.

The second value is the slope of the roof. The third value is the horizontal length of the wall and the fourth value is the wall height. The fifth value is the wall slope. Note for a vertical wall the slope is a very large number, 9999.0 has been used in this case. This version of the program cannot handle anything except vertical walls. The sixth value is the total area of the walls including any windows or doors on that wall. The triangular area on walls (3) and (6) that have an attic behind them are not included in this wall area. The seventh value is the surface area of the roof. The eighth value is the width of overhang on each wall. For a wall such as (3) or (6) that does not have a horizontal overhang, then the width is considered to be zero. The ninth value to be read is the actual running width of the roof. The tenth value is the horizontal length of the hip ridge on the roof above this particular wall.

14. This card contains the number of windows on each wall. The first number is the number of windows on wall (1), the second number is the number of windows on wall (2), and etc.

15. This card contains the number of doors on each wall. The first number indicates the number of doors on the first wall, the second number indicates the number of doors on the second wall, and etc.

16,17,18,
19,20 & 21.

These six cards contain the information on the window and doors. One card contains information on one window or door and a card is used for each window and door. The order that windows and doors are read in is very important. The first cards read are the windows on the first wall, immediately following these are the doors on the first wall. Next the windows on the second wall are read, followed by the doors on this wall. This order is demonstrated in the sample input data.

Card (16) is for the door on wall (1).

Card (17) is for the window on wall (2).

Card (18) is for the window on wall (3).

Card (19) is for the window on wall (4).

Card (20) is for the window on wall (5).

Card (21) is for the door on wall (5).

The first value on these cards read is the vertical distance below the window or door. The second value is the height of the window or door. The third value is the vertical distance above the window or door. The fourth value is the horizontal distance to the right of the window or door. The fifth value is the width of the window or door. The sixth value is the length to the left of the window or door. The seventh value is the type of window or door. A listing of the type of windows and doors can be found in the subroutine AINFIL.

The eighth value is the area of the window or door.

- 22. This card has the inside design temperature.
- 23 & 24. These two cards contain the outside temperature profile. The first card contains the first 16 temperatures and the second card contains the last 8 temperatures.
- 25. This card contains the wall construction of each wall. A listing of wall constructions can be found in the subroutine load.
- 26. This card contains ceiling and attic information. The first value reads the ceiling type. The second value is the ceiling area. The third value is the volume of attic air. A listing of ceiling types can be found in the subroutine load.
- 27. The last card contains two values. The first one is the number of members in the family. The second value is the total light capacity in BTU/hr of the house.

APPENDIX D

The following is a complete, commented listing of the entire cooling load program for residential structures.

C THIS PROGRAM CALCULATES A TWENTY-FOUR HOUR COOLING LOAD PROFILE
 C FOR RESIDENTIAL HOMES.
 C THE FIRST THREE DATA CARDS ARE USED AS LITERAL FIELDS TO SUPPLY
 C THE AXIS LABELS TO THE CALCOMP PLOTTER. THE NEXT READ STATEMENT IN
 C THE MAIN PROGRAM IS THE NUMBER OF SUBROUTINES CALLED. TO MAKE THE
 C PROGRAM MORE VERSITAL THE MAIN PROGRAM WAS WRITTEN TO INITIALIZE
 C CERTAIN VALUES NECESSARY AND TO LOGICALLY CALL THE SUBROUTINE
 C NEEDED. THIS MEANS MORE THAN ONE HOME CAN BE RUN AT A TIME AND THE
 C AMCUNT OF INFORMATION PRINTED OUT ABOUT EACH HOME CAN BE VARIED.
 C THERE IS A MINIMUM SEQUENCE THAT MUST BE CALLED TO OBTAIN A
 C COOLING LOAD PROFILE. THIS MINIMAL SEQUENCE IS LISTED BELOW.
 C REVAL
 C SHADE
 C SOLTEM
 C LOAD
 C DWLCAD
 C AINFIL
 C AINTER
 C TOHEAT
 C COOLSU
 C TOCOOL
 C THESE SUBROUTINES ARE NUMBERED ACCORDING TO THE NUMBER ON THEIR
 C CALL STATEMENT STARTING WITH STATEMENT 1. STATEMENTS 1 THROUGH 12
 C ARE ACTUALLY SUBROUTINES. STATEMENTS 13 THROUGH 17 ARE EXECUTABLE
 C OPTIONS THAT CAN BE CALLED. THE FOLLOWING DESCRIPTION WILL
 C DESCRIBE WHAT THESE DO.
 C STATEMENT 13: READS PREVIOUSLY STORED DATA OFF OF THE DISK
 C STATEMENT 14: WRITES THE LISTED DATA OUT ON TO THE DISK AFTER ALL
 C CALCULATIONS HAVE BEEN MADE.
 C STATEMENT 15: THIS STATEMENT WILL RESULT IN THE COOLING LOAD
 C PROFILE BEING WRITTEN OUT.
 C STATEMENT 16 & 17: THESE ARE CALCOMP PLOTTER STATEMENTS USED TO
 C PLOT THE COOLING LOAD DATA. THE SUBROUTINES THESE CALL WERE
 C PROVIDED BY DR. HUGH WALKER.
 C ALL OTHER READ AND WRITE STATEMENTS ARE CONTAINED IN THE
 C SUBROUTINES. MUCH OF THIS PROGRAMS PROCEEDURES ARE OUTLINED IN THE
 C ASHRAE HANDBOOK OF FUNDAMENTALS, 1972. EACH OF THE SUBROUTINES ARE
 C COMMENTED AT THE BEGINING AND THROUGHOUT.
 C COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHFK,NN,NNN,N1,JCHEK
 C COMMON/AIRC/ALITE(24,15),BLITE(24,15),NUMC(15)
 C COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),I1I
 C COMMON/WALPRO/B0,B1,B2,B3,B4,B5,B6,BB0,BB1,BB2,BB3,BB4,BB5,BB6
 C COMMON/WALPRI/D1,D2,D3,D4,D5,D6,DD1,DD2,DD3,DD4,DD5,DD6,SC,SSC
 C COMMON/AIRC1/NOWIN(15),NCDQOR(15),WINDOW(15,10,8),NUMB(15),KODE
 C COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
 C COMMON/AIPC6/WTM(30,15),SRTM(15),ARTM(15)
 C COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLCD(15)
 C COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)

```

COMMON/AIRC9/UD(15,10),QDOCR(24,15,10),QWIND(24,15,10)
COMMON/AIR10/ALIGHT(24),PEOPLS(24),PEOPL(24),MEMBER,TOTLIG
COMMON/AC1/SACCOF(15,10)
COMMON/AIR11/TOHOU(24),TOLAT(24),TCWINS(24,15),TOWALR(24,15)
COMMON/AIR12/TOWLAT(24,15),TOWALL(24,15),FACTOR,COLO(28)
COMMON/AIR13/COOL(28),CT(24),TOHC(24)
COMMON/AIR14/SEC,CRFL(15,10),CRALEN(15,10),ALFIN(15,10,24)
COMMON/AIR20/SINFIL(15,10,24),ALFIL(15,10,24)
COMMON/AC21/TA(30),NOCE,NOWA(15),CEAR,VCLAIR,TR(30),WT(30)
COMMON/AIR3/QLOT(30,15),RQLOT(30)
COMMON/ARCON/XNUB(24),SUB(100)
DIMENSION XA(10),YB(10),ZC(10),XNU1(24,10),CT1(24,10)
105 FORMAT(' THE VALUES GIVEN FOR THE SUBROUTINE DESIRED IS GREATER T
      1HAN 14 AND IS EQUAL TO',I5)
503 FORMAT(' THE FOLLOWING VALUES OF CT(HOUR) IS THE 24 HOUR COOLING
      1LOAD FOR THE HOUSE')
504 FORMAT(' CT(',I2,')=',F9.1)
505 FORMAT(I10)
506 FORMAT(8F10.1)
507 FORMAT(10A4)
      KA10=0
C      KZ IS USED IN THE PLOTTER SUBROUTINES AND MUST BE SET TO ZERO AT
C      THE BEGINING OF THE PROGRAM.
      KZ=0
      IDISK=10
      KODE=0
      REWINDIDISK
      DO 200 JJ=1,24
      XNUB(JJ)=JJ
200 CONTINUE
      READ(5,507)(XA(J),J=1,10)
      READ(5,507)(YB(J),J=1,10)
      READ(5,507)(ZC(J),J=1,10)
C      NSUB IS THE NUMBER OF SUBROUTINES THAT WILL BE CALLED.
      READ(5,505) NSUB
C      ARRAY SUB(N) CONTAINS THE SEQUENCE OF SUBROUTINES THAT ARE TO BE
C      CALLED.
      READ(5,506)(SUB(N),N=1,NSUB)
      DO 38 K=1,NSUB
      J=SUB(K)
C      THE FOLLOWING IF STATEMENTS SELECT WHICH SUBROUTINE WILL BE THE
C      K TH SUBROUTINE TO BE CALLED.
      IF(J.EQ.1) GO TO 1
      IF(J.EQ.2) GO TO 2
      IF(J.EQ.3) GO TO 3
      IF(J.EQ.4) GO TO 4
      IF(J.EQ.5) GO TO 5
      IF(J.EQ.6) GO TO 6

```

```

IF(J.EQ.7) GO TO 7
IF(J.EQ.8) GO TO 8
IF(J.EQ.9) GO TO 9
IF(J.EQ.10) GO TO 10
IF(J.EQ.11) GO TO 11
IF(J.EQ.12) GO TO 12
IF(J.EQ.13) GO TO 13
IF(J.EQ.14) GO TO 14
IF(J.EQ.15) GO TO 15
IF(J.EQ.16) GO TO 16
IF(J.EQ.17) GO TO 17
WRITE(6,105) J
GO TO 52
C   THE FOLLOWING IS THE LIST OF SUBROUTINES.
C   1 CALL REVAL
C   IF KODE IS FOUND EQUAL TO 1 A BAD VALUE WAS FOUND AND EXECUTION OF
C   THE PROGRAM IS STOPPED IMMEDIATELY.
IF(KODE.EQ.1) GO TO 52
GO TO 38
2 CALL SHADE
GO TO 38
3 CALL SHAFRA
GO TO 38
4 CALL SCLTEM
GO TO 38
5 CALL LCAD
IF(KODE.EQ.1) GO TO 52
GO TO 38
6 CALL DWLCAD
IF(KODE.EQ.1) GO TO 52
GO TO 38
7 CALL VALLO
GO TO 38
8 CALL AINFIL
IF(KODE.EQ.1) GO TO 52
GO TO 38
9 CALL AINTER
GO TO 38
10 CALL TCHEAT
GO TO 38
11 CALL CCCLSU
GO TO 38
12 CALL TOCOOL
GO TO 38
13 REWINDISK
READ(IDISKIN,AA,ALAT,M,DIME2,N1,ALITE,RLITE,NOWIN,NODOOR,WINDOW,NU
1MB,TEMP,SOLAVW,DTM,RTM,WTM,SRTM,ARTM,QLQD,PQLQD,SQLOD,QDOOP,QWIND,
1QLQD,RQLQD,CT,WLITE,DLITE,NUMC,NNN,KALO,CT1,XNU1,XA,YB,ZC

```

```

      GO TO 38
14  REWINDDISK
      WRITE(IDISK)N,AA,ALAT,M,DIME2,N1,ALITE,BLITE,NOWIN,NODOOR,WINDOW,N
      UMB,TEMP,SOLAVW,DTM,RTM,WTM,SRTM,ARTM,QLOD,RQLQD,SQLOD,QDOOR,QWIND
      1,QLOT,RQLCT,CT,WLITE,DLITE,NUMC,NNN,KALO,CT1,XNU1,XA,YB,ZC
      GO TO 38
15  WRITE(6,503)
      KALO=KALO+1
      DO 120 ITIM=1,24
      CT1(ITIM,KALO)=CT(ITIM)
      XNU1(ITIM,KALO)=ITIM
120  WRITE(6,504) ITIM,CT(ITIM)
      GO TO 38
16  IF(KZ.NE.0) GO TO 201
      CALL STARTP
      KZ=KZ+1
201  CALL TITLE('COOLING LOAD PROFILE',20)
      CALL CALCCM(XNUB,CT,XA,YB,24,40,40)
      CALL MOVEON
      GO TO 38
17  IF(KZ.NE.0) GO TO 301
      CALL STARTP
      KZ=KZ+1
301  CALL TITLE('COOLING LOAD PROFILE',20)
      DO 302 IZZ=1,KALO
      DO 303 ITT=1,24
      CT(ITT)=CT1(ITT,IZZ)
303  CONTINUE
      CALL CVRLAY(XNUB,CT,ZC(IZZ),4,24,XA,YB,40,40)
302  CCNTINUE
      CALL MOVECN
      GO TO 38
38  CCNTINUE
52  CCNTINUE
      IF(KZ.NE.1) GO TO 202
      CALL STOPP
202  STOP
      END

```

```

SUBROUTINE REVAL
C   THIS SUBROUTINE CONTAINS ALL OTHER READ STATEMENTS CONTAINED IN
C   THE PROGRAM OR ALL THE READ STATEMENTS THAT PERTAIN TO THE COOLING
C   LOAD CALCULATIONS. THE VARIABLES USED IN THE SUBROUTINE ARE :
C   ANN=THE NUMBER OF WALL AROUND THE OUTSIDE OF THE HOME.
C   AA=THE ANGLE OF ORIENTATION WALL ONE IS FROM SOUTH: CLOCKWISE
C   (POSITIVE): COUNTER-CLOCKWISE(NEGATIVE)
C   AM=M=MONTH (1-12) OR (JAN. -DEC.) RESPECTIVELY
C   ALAT=LATITUDE
C   N1=N+1 THIS IS USED BECAUSE THE NUMBER OF WALLS IS CONTAINED IN
C   THE ARRAY'S SUBSCRIPT(2 TO N+1). THIS IS DONE TO HAVE CONTINUITY
C   ALL THE WAY AROUND THE HOUSE WHEN THE PROGRAM EXAMINES THE ANGLE
C   BETWEEN TWO WALL'S WALL AZIMUTHS.
C   NUMB-AND NUMC-ARE ARRAYS USED TO STORE CONSECUTIVE NUMBERS THAT
C   ARE USED AS TABLE HEADINGS FOR WALLS AND WINDOWS.
C   DIME2(I,J)=THE MATRIX THAT STORES WALL AND ROOF DIMENSIONS FOR
C   EACH WALL AND ROOF.
C   NOWIN(J)=THE NUMBER OF WINDOWS ON EACH WALL.
C   NODOOR(J)=THE NUMBER OF DOORS ON EACH WALL.
C   WINDOW(J,I,JJ)=THE MATRIX THAT STORES WINDOW AND DOOR DIMENSIONS
C   FOR EACH WINDOW AND DOOR ON EACH WALL.
C   ROOMT=INSIDE ROOM TEMPERATURE
C   TEMP(J)=OUTSIDE TEMPERATURE PROFILE
C   NOWA(N)=THE CONSTRUCTION TYPE OF EACH WALL.
C   NOCE=THE CONSTRUCTION TYPE OF THE CEILING USED.
C   CEAR=CEILING AREA (SQUARE FEET).
C   VOLAIR=VOLUME OF ATTIC AIR (CUBIC FEET).
C   MEMBER=TOTAL NUMBER OF FAMILY MEMBERS.
C   TOTLIG=THE TOTAL LIGHTS CONNECTED IN THE HOUSE.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHECK,NN,NNN,N1,JCHECK
COMMON/AIPC/ALITE(24,15),BLITE(24,15),NUMC(15)
COMMON/AIRC1/NOWIN(15),NODOOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIRC10/ALIGHT(24),PEOPLS(24),PEOPL(24),MEMBER,TOTLIG
COMMON/AC21/TA(30),NOCE,NOWA(15),CEAR,VOLAIR,TR(30),WT(30)
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
102 FORMAT(15I5)
503 FORMAT(I8,F12.2)
103 FORMAT(I10,F10.2,F10.2)
131 FORMAT(10F7.2)
137 FORMAT(4F10.2)
142 FORMAT(1X,15I2)
400 FORMAT(16(1X,F4.1)/8(1X,F4.1))
106 FORMAT(' THE TEMPERATURE IN THE HOUSE IS OUTSIDE THE ALLOWABLE RA
NGE (50F-100F) , PLEASE CHECK TO MAKE SURE YOU HAVE ENTERED '//THE
ICORRECT TEMPERATURES')
104 FORMAT(' THE OUTSIDE TEMPERATURE AT TIME ',I3,' IS OUTSIDE THE A

```



```

11LLCABLE TEMPERATURE RANGE (-40F-120F), PLEASE CHECK TO MAKE '/*' SUR
1E YOU HAVE ENTERED THE CORRECT TEMPERATURES')
  READ(5,137) ANN,AA,AM,ALAT
  NN=ANN
  P=AM
  N1=NN+1
  ANN=NN+2
  DO 30 I=2,14
    NUMC(I)=I-1
    NUMB(I)=I-1
    NUMB(1)=NUMB(N1)
    NUMB(NNN)=NUMB(2)
    READ(5,131)((DIME2(I,J),I=1,10),J=2,N1)
    READ(5,142)(NOWIN(J),J=2,N1)
    READ(5,142)(NCDOOR(J),J=2,N1)
    DO 10 J=2,N1
      K=NOWIN(J)+NCDOOR(J)
      IF(K.EQ.0) GO TO 10
      DO 11 I=1,K
        READ(5,131)(WINDOW(J,I,JJ),JJ=1,8)
      CONTINUE
      READ(5,400) ROOMT
      READ(5,400)(TEMP(J),J=1,24)
      READ(5,102) (NOWA(N),N=2,N1)
      READ(5,103) NOCE,CEAR,VOLAIR
      READ(5,503) MEMBER,TCTLIG
      ALAT=3.14159*ALAT/180.
      THE FOLLOWING ARE CHECK STATEMENTS TO MAKE SURE REASONABLE
      TEMPERATURES ARE READ IN BEFORE ANYMORE CALCULATIONS ARE MADE.
      IF(ROOMT.GT.100..OR.ROOMT.LT.50.) GO TO 50
      DO 60 ITIM=1,24
        IF(TEMP(ITIM).GT.120..OR.TEMP(ITIM).LT.-40.) GO TO 51
      CONTINUE
      GO TO 40
    WRITE(6,106)
    KODE=1
    RETURN
  51 WRITE(6,104) ITIM
    KODE=1
    RETURN
  40 CONTINUE
    RETURN
  END

```

```

SUBROUTINE SHADE
C THIS SUBROUTINE LOADS THE PERCENT AREA SUNLIT OF WALLS, ROOFS,
C WINDOWS, AND DOORS IN TO THE MATRICES ALITE, BLITE, WLITE, AND
C DLITE. THESE MATRICES ARE USED IN THE SUBROUTINE SOLTEMP TO
C CALCULATE ADJUSTED SCL-AIR TEMPERATURES. THESE PERCENT AREAS
C SUNLIT ARE CALCULATED BY THE FUNCTION WAREA.
C DIME2(I,J)=THE MATRIX THAT CONTAINS WALL AND ROOF DIMENSIONS FOR
C EACH WALL.
C ALITE(ITIM,N)=THE MATRIX THAT CONTAINS PERCENT SUNLIT AREAS FOR
C WALLS.
C BLITE(ITIM,N)=THE MATRIX THAT CONTAINS PERCENT SUNLIT AREAS FOR
C ROOFS.
C WLITE(ITIM,III)=THE MATRIX THAT CONTAINS PERCENT SUNLIT AREAS FOR
C WINDOWS.
C DLITE(ITIM,N,MM)=THE MATRIX THAT CONTAINS PERCENT SUNLIT AREAS
C FOR DOORS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),ACDOOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
DO 200 J=1,10
DIME2(J,1)=DIME2(J,NN+1)
200 DIME2(J,NN+2)=DIME2(J,2)
ICHEK=1
C IN FUNCTION WAREA(X) ICHEK MUST EQUAL 1 TO GET THE PERCENT AREA OF
C A WALL SUNLIT
DO 34 ITIM=1,24
DO 35 N=2,N1
35 ALITE(ITIM,N)=WAREA(X)
34 CONTINUE
ICHEK=2
C IN FUNCTION WAREA(X) ICHEK MUST EQUAL 2 TO GET THE PERCENT AREA OF
C A ROOF SUNLIT.
DO 44 ITIM=1,24
DO 45 N=2,N1
45 BLITE(ITIM,N)=WAREA(X)
44 CONTINUE
ICHEK=3
C IN FUNCTION WAREA(X) ICHEK MUST EQUAL 3 TO GET THE PERCENT AREA OF
C A WINDOW OR DOOR SUNLIT.
C THE FOLLOWING PROCEEDURE IS USED OVER AND OVER AS A METHOD TO MAKE
C CALCULATIONS ON EVERY WINDOW AND DOOR IN A HOUSE.
C KK- IS THE NUMBER OF WINDOWS ON WALL N.
C KK5- IS THE NUMBER OF DOORS ON WALL N.
C IF EITHER KK OR KK5 IS ZERO IT WILL SKIP OVER THESE CALCULATIONS
C FOR WALL N.
DO 50 ITIM=1,24
DO 51 N=2,N1

```

```
      KK=NOWIN(N)  
      KK5=NODDOR(N)  
      IF(KK.EQ.0) GO TO 54  
      DO 52 III=1, KK  
52  WLTE(ITIM,N,III)=WAREA(X)  
54  IF(KK5.EQ.0) GO TO 51  
      DO 53 MM=1, KK5  
      III=MM+KK  
53  CLTE(ITIM,N,MM)=WAREA(X)  
51  CONTINUE  
50  CCNTINUE  
      RETURN  
      END
```

```

FUNCTION WARFA(X)
C   THE PURPOSE OF THIS FUNCTION IS TO CALCULATE THE PERCENT SUNLIT
C   AREA ON ANY SURFACE ON A HOUSE. THESE AREAS INCLUDE ROOFS, WALLS
C   WINDOWS, AND DOORS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15),NUMC(15)
COMMON/AIRC1/NOWIN(15),NODDOOR(15),WINDOW(15,10,8),NUMR(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
C   THE FIRST STEP IN ANY CALCULATION IS TO DETERMINE THE MONTH AND
C   CALCULATE THE SOLAR ALTITUDE AND THE SOLAR AZIMUTH OF THE SUN AT
C   TIME ITIM.
IF(M.EQ.8) GO TO 5
IF(M.EQ.7) GO TO 6
IF(M.EQ.12) GO TO 3
IF(M.EQ.1) GO TO 4
5  A=351.0
   B=0.201
   C=0.122
   DEG=0.214675
   GO TO 7
6  A=344.0
   B=0.207
   C=0.136
   DEG=0.359537
   GO TO 7
3  A=391.
   B=0.142
   C=0.057
   DEG=-0.40927
   GO TO 7
4  A=390.
   B=0.142
   C=0.058
   DEG=-0.34907
   GO TO 7
7  AL=ITIM
   AH=(((AL-12.0)*60.0)*0.25)
   AH1=AH-0.005
   AH=3.14159*AH/180.
   AH1=3.14159*AH1/180.
   SIGH1=(90.0*(DIME2(1,N-1)-1.0))+AA
   SIGH2=(90.0*(DIME2(1,N)-1.0))+AA
   SIGH3=(90.0*(DIME2(1,N+1)-1.0))+AA
   SIGH4=SIGH2
   IF(SIGH4.GT.360.) SIGH4=SIGH4-360.
   IF(SIGH4.GT.180.) SIGH4=SIGH4-360.
   BETA=ARSIN((COS(ALAT)*COS(DEG)*COS(AH))+((SIN(ALAT)*SIN(DEG))))
   BET=ARSIN((COS(ALAT)*COS(DEG)*COS(AH1))+((SIN(ALAT)*SIN(DEG))))

```

```

      IF(BETA.LE.0.0) GO TO 8
      GO TO 9
8     WAREA=0.0
      RETURN
9     PHI=ARSIN(COS(DEG)*SIN(AH)/COS(BETA))
      PHI1=ARSIN(COS(DEG)*SIN(AH1)/COS(BET1))
      IF(PHI1.LT.PHI) GO TO 10
      IF(AH) 11,11,12
11    PHI=-3.14159-PHI
      GO TO 10
C     PHI IS THE SOLAR AZIMUTH AND BETA IS THE SOLAR ALTITUDE.
12    PHI=3.14159-PHI
10    SIGH4=3.14159*SIGH4/180.
      IF(PHI.LT.0.0.AND.SIGH4.GT.3.14159+PHI) SIGH4=SIGH4-6.28318
      IF(PHI.GT.0.0.AND.SIGH4.LT.PHI-3.14159) SIGH4=SIGH4+6.28318
C     GAMMA IS THE WALL-SOLAR AZIMUTH.
      GAMMA=PHI-SIGH4
      IF(GAMMA.GT.1.56905.AND.GAMMA.LT.1.588249) GAMMA=1.56905
      IF(GAMMA.LT.-1.56905.AND.GAMMA.GT.-1.588249) GAMMA=-1.56905
C     IF ICHEK IS 2 THE CALCULATION IS FOR ROOFS.
      IF(ICHEK.EQ.2) GO TO 50
C     IF ICHEK IS 3 THE CALCULATION IS FOR WINDOWS AND DOORS.
      IF(ICHEK.EQ.3) GO TO 139
      IF(ABS(GAMMA).GT.1.57079) GO TO 14
C     THE NEXT TWO STATEMENTS SHOW CASES WHEN THE WALL IS IN THE SHADE,
C     FIRST ONE IS SHADOWING DUE TO THE WALL FACING THE WRONG DIRECTION
C     THE SECOND ONE SHOWS OVERHANGS SHADOWING THE ENTIRE WALL.
      IF(DIME2(8,N)*TAN(BETA)/(DIME2(4,N)*COS(GAMMA)).GT.1.0) GO TO 14
      GO TO 13
14    UNSHA=0.0
      WAREA=UNSHA
      RETURN
C     *****
C     *****
C     WALLS
C     WALLS
13    UNSHA=(1.-(DIME2(8,N)*TAN(BETA)/(DIME2(4,N)*COS(GAMMA))))
      DIF1=SIGH1-SIGH2
      DIF2=SIGH2-SIGH3
C     THIS CHECK IS FOR U TYPE WALL SHADING.
      IF(DIF1.EQ.90..AND.DIF2.EQ.90.) GO TO 39
C     THIS CHECKS FOR COUNTER-CLOCKWISE L SHAPES.
49    IF(DIF1.NE.90.) GO TO 15
C     IF THE HOUSE IS A COUNTER-CLOCKWISE L, GAMMA MUST BE NEGATIVE FOR
C     SIDE WALL SHADING TO OCCUR.
      IF(GAMMA.GE.0.0) GO TO 19
      SIGX1=SIGH1-180.
      IF(SIGX1.GT.180.) SIGX1=SIGX1-360.

```

```

SIGX1=3.14159*SIGX1/180.
IF(PHI.LT.0.0.AND.SIGX1.GT.PHI+3.14159) SIGX1=SIGX1-6.28318
GAMX1=PHI-SIGX1
BETA1=ATAN(TAN(BETA1)/COS(GAMX1))
IF(BETA1.GT.1.56905.AND.BETA1.LT.1.588249) BETA1=1.56905
GAMM=-GAMMA
IF(TAN(BETA1).LT.DIME2(2,N-1)) GO TO 16
GRO=DIME2(9,N-1)*COS(ATAN(DIME2(2,N-1)))+DIME2(3,N)-DIME2(8,N-1)
A=DIME2(4,N-1)/TAN(BETA1)
B=DIME2(3,N-1)*TAN(GAMM)
WALL=DIME2(3,N)
IF(A+DIME2(8,N-1).GT.B) GO TO 30
IF(A+DIME2(8,N-1).GT.WALL) GO TO 31
WAREA=((DIME2(3,N)-A-DIME2(8,N-1))+(UNSHA*DIME2(4,N))/(2.0*TAN(BET
1A1)))*UNSHA*DIME2(4,N)/DIME2(6,N)
RETURN
30 IF(B.GT.WALL) GO TO 31
IF(TAN(BETA1)*(B-DIME2(8,N-1)).LT.(1.0-UNSHA)*DIME2(4,N)) GO TO 33
WAREA=(UNSHA*DIME2(4,N)*(DIME2(3,N)-B)+(((TAN(BETA1)*(B-DIME2(8,N-
11))-DIME2(4,N)*(1.0-UNSHA))*2)/(2.0*TAN(BETA1))))/DIME2(6,N)
RETURN
33 WAREA=(DIME2(3,N)-B)*DIME2(4,N)*UNSHA/DIME2(6,N)
RETURN
31 AB=A+DIME2(8,N-1)-DIME2(3,N)
IF(AB+TAN(BETA1).GT.UNSHA*DIME2(4,N)) GO TO 32
WAREA=((DIME2(4,N)*UNSHA-AB*TAN(BETA1))*2)/(2.0*TAN(BETA1)*DIME2(
16,N))
RETURN
32 WAREA=0.0
RETURN
16 GRO=DIME2(9,N-1)*COS(ATAN(DIME2(2,N-1)))+DIME2(3,N)-DIME2(8,N-1)
CCC=(DIME2(9,N-1)*SIN(ATAN(DIME2(2,N-1)))+DIME2(4,N-1))/TAN(BETA1)
DDD=COS(ATAN(DIME2(2,N-1)))*DIME2(9,N-1)-DIME2(8,N-1)+DIME2(3,N-1)
1*TAN(GAMM)
IF(DDD.GT.CCC) GO TO 35
IF(DDD.GT.GRO) GO TO 36
AAA=(DIME2(9,N-1)*SIN(ATAN(DIME2(2,N-1))))/TAN(BETA1)
AAAA=AAA-COS(ATAN(DIME2(2,N-1)))*DIME2(9,N-1)+DIME2(8,N-1)
IF(((1.0-UNSHA)*DIME2(4,N).GT.(DDD*TAN(BETA1)-SIN(ATAN(DIME2(2,N-1)
1)*DIME2(9,N-1)))) GO TO 37
WAREA=((UNSHA*DIME2(4,N)*(GRO-DDD))+((DDD*TAN(BETA1)-SIN(ATAN(DIME
12(2,N-1))) *DIME2(9,N-1)-(1.0-UNSHA)*DIME2(4,N))*2)/(2.0*TAN(BETA1
1)))/DIME2(6,N)
RETURN
37 WAREA=UNSHA*DIME2(4,N)*(GRO-DDD)/DIME2(6,N)
RETURN
35 IF(CCC.GT.GRO) GO TO 36
WAREA=(UNSHA*(GRO-CCC)*DIME2(4,N)+((UNSHA*DIME2(4,N))*2)/(2.0*TAN

```

```

1(BETA1))/DIME2(6,N)
RETURN
36 IF((CCC-GRO)*TAN(BETA1).GT.UNSHA*DIME2(4,N)) GO TO 32
WAREA=((UNSHA*DIME2(4,N)-(CCC-GRO)*TAN(BETA1))**2)/(2.0*TAN(BETA1)
1*DIME2(6,N))
RETURN
C THIS CHECKS IF THE HOUSE HAS CLOCKWISE L SHADING.
15 IF(DIF2.NE.90.) GO TO 19
C IF THE HOUSE HAS CLOCKWISE L SHADING THEN GAMMA MUST BE POSITIVE.
C IF THE WALL HAS NO SIDE SHADING THEN THE ONLY SHADING THAT IS
C NEEDED IS THE SHADING FROM OVERHANGS.
IF(GAMMA.LT.0.0) GO TO 19
SIGX3=SIGH3+180.
IF(SIGX3.GT.180.) SIGX3=SIGX3-360.
SIGX3=3.14159*SIGX3/180.
IF(PHI.GT.0.0.AND.SIGX3.LT.PHI-3.14159) SIGX3=SIGX3+6.28318
GAMX3=PHI-SIGX3
BETA3=ATAN(TAN(BETA)/COS(GAMX3))
IF(BETA3.GT.1.56905.AND.BETA3.LT.1.588249) BETA3=1.56905
IF(TAN(BETA3).LT.DIME2(2,N+1)) GO TO 17
GRO=DIME2(9,N+1)*COS(ATAN(DIME2(2,N+1)))+DIME2(3,N)-DIME2(8,N+1)
A=DIME2(4,N+1)/TAN(BETA3)
WALL=DIME2(3,N)
B=DIME2(3,N+1)*TAN(GAMMA)
IF(A+DIME2(8,N+1).GT.B) GO TO 40
IF(A+DIME2(8,N+1).GT.WALL) GO TO 41
WAREA=((DIME2(3,N)-A-DIME2(8,N+1))+((UNSHA*DIME2(4,N))/(2.0*TAN(BET
1A3))))*UNSHA*DIME2(4,N)/DIME2(6,N)
RETURN
40 IF(B.GT.WALL) GO TO 41
IF(TAN(BETA3)*(B-DIME2(8,N+1)).LT.(1.0-UNSHA)*DIME2(4,N)) GO TO 43
WAREA=(UNSHA*DIME2(4,N)*(DIME2(3,N)-B)+(((TAN(BETA3)*(B-DIME2(8,N+
1)))-DIME2(4,N)*(1.0-UNSHA))**2)/(2.0*TAN(BETA3))))/DIME2(6,N)
RETURN
43 WAREA=(DIME2(3,N)-B)*DIME2(4,N)*UNSHA/DIME2(6,N)
RETURN
41 AB=A+DIME2(8,N+1)-DIME2(3,N)
IF(AB+TAN(BETA3).GT.UNSHA*DIME2(4,N)) GO TO 32
WAREA=((DIME2(4,N)*UNSHA-AB*TAN(BETA3))**2)/(2.0*TAN(BETA3)*DIME2(
16,N))
RETURN
17 GRO=DIME2(9,N+1)*COS(ATAN(DIME2(2,N+1)))+DIME2(3,N)-DIME2(8,N+1)
CCC=(DIME2(9,N+1)*SIN(ATAN(DIME2(2,N+1)))+DIME2(4,N+1))/TAN(BETA3)
DDD=COS(ATAN(DIME2(2,N+1)))*DIME2(9,N+1)-DIME2(8,N+1)+DIME2(3,N+1)
1*TAN(GAMMA)
IF(DDD.GT.CCC) GO TO 45
IF(DDD.GT.GRO) GO TO 46
AAA=(DIME2(9,N+1)*SIN(ATAN(DIME2(2,N+1))))/TAN(BETA3)

```

```

      AAAA=AAA-COS(ATAN(DIME2(2,N+1)))*DIME2(9,N+1)+DIME2(8,N+1)
      IF((1.0-UNSHA)*DIME2(4,N).GT.(DDD*TAN(BETA3)-SIN(ATAN(DIME2(2,N+1)
1)*DIME2(9,N+1)))) GO TO 47
      WAREA=((UNSHA*DIME2(4,N)*(GRO-DDD))+((DDD*TAN(BETA3)-SIN(ATAN(DIME
12(2,N+1)))*DIME2(9,N+1)-(1.0-UNSHA)*DIME2(4,N)**2)/(2.0*TAN(BETA3
1)))/DIME2(6,N)
      RETURN
47  WAREA=UNSHA*DIME2(4,N)*(GRO-DDD)/DIME2(6,N)
      RETURN
45  IF(CCC.GT.GRO) GO TO 46
      WAREA=(UNSHA*(GRO-CCC)*DIME2(4,N)+((UNSHA*DIME2(4,N)**2)/(2.0*TAN
1(BETA3)))/DIME2(6,N)
      RETURN
46  IF((CCC-GRO)*TAN(BETA3).GT.UNSHA*DIME2(4,N)) GO TO 32
      WAREA=((UNSHA*DIME2(4,N)-(CCC-GRO)*TAN(BETA3)**2)/(2.0*TAN(BETA3)
1*DIME2(6,N))
      RETURN
39  IF(GAMMA) 49,19,15
19  WAREA=UNSHA
      RETURN
C *****
C *****
C *****
C *****
50  DIF1=SIGH1-SIGH2
      DIF2=SIGH2-SIGH3
C  THIS IS A CHECK STATEMENT FOR A U SHAPED ROOF.
      IF(DIF1.EQ.90..AND.DIF2.EQ.90.) GO TO 56
C  THIS IS A CHECK STATEMENT FOR AN I SHAPED ROOF AND WALL.
      IF(DIF1.NE.90..AND.DIF2.NE.90.) GO TO 55
C  THIS CHECKS IF THE ROOF HAS COUNTER-CLOCKWISE L SHADING.
      IF(DIF1.NE.90.) GO TO 54
      SIGX1=SIGH1-180.
      IF(SIGX1.GT.180.) SIGX1=SIGX1-360.
      SIGX1=3.14159*SIGX1/180.
      IF(PHI.LT.0.0.AND.SIGX1.GT.3.14159+PHI) SIGX1=SIGX1-6.28318
      GAMX1=PHI-SIGX1
      IF(GAMX1.LE.-1.57079) GO TO 52
      BETA1=ATAN(TAN(BETA)/COS(GAMX1))
      IF(BETA1.GT.1.56905.AND.BETA1.LT.1.588249) BETA1=1.56905
      GAMM=-GAMMA
      EEE=TAN(GAMM)*(DIME2(10,N-1)-DIME2(9,N)*COS(ATAN(DIME2(2,N))))
      FFF=DIME2(9,N-1)*SIN(ATAN(DIME2(2,N-1)))/TAN(BETA1)
      IF(ABS(GAMMA).GE.1.57079) GO TO 59
      IF(GAMMA.GE.0.0.AND.GAMMA.LT.1.57079) GO TO 58
      IF(EEE+DIME2(9,N-1)*COS(ATAN(DIME2(2,N-1))).GT.FFF) GO TO 61
      IF(EEE+DIME2(9,N-1)*COS(ATAN(DIME2(2,N-1))).GT.DIME2(10,N)) GO TO
162

```



```

        WAREA=((DIME2(10,N)-EEE)*DIME2(9,N)+((EEE)**2)*TAN(BETA1)/(2.0*SIN
1(ATAN(DIME2(2,N)))))/DIME2(7,N)
        RETURN
52  IF(TAN(BETA)/ABS(COS(GAMMA)).GT.DIME2(2,N)) GO TO 58
        WAREA=0.0
        RETURN
62  WAREA=(DIME2(9,N)-((FFF-DIME2(10,N))*TAN(BETA1)/SIN(ATAN(DIME2(2,N
1)))))*DIME2(10,N)/(2.0*DIME2(7,N))
        RETURN
59  IF(TAN(BETA)/ABS(COS(GAMMA)).GT.DIME2(2,N)) GO TO 60
        WAREA=0.0
        RETURN
58  WAREA=1.0
        RETURN
60  IF(GAMMA.GT.0.0) GO TO 58
        IF(DIME2(2,N-1).GT.TAN(BETA1)) GO TO 61
        GO TO 58
61  IF(FFF.GT.DIME2(10,N)) GO TO 62
        IF(TAN(BETA1).GT.DIME2(2,N-1)) GO TO 58
        WAREA=((DIME2(10,N)-(FFF/2.0))*DIME2(9,N))/DIME2(7,N)
        RETURN
C  THIS STATEMENT CHECKS IF THE ROOF HAS CLOCKWISE L ROOF SHADING.
54  IF(DIF2.NE.90.) GO TO 55
        SIGX3=SIGF3+180.
        IF(SIGX3.GT.180.) SIGX3=SIGX3-360.
        SIGX3=3.14159*SIGX3/180.
        IF(PHI.GT.0.0.AND.SIGX3.LT.PHI-3.14159) SIGX3=SIGX3+6.28318
        GAMX3=PHI-SIGX3
        IF(GAMX3.GE.1.57079) GO TO 63
        BETA3=ATAN(TAN(BETA)/COS(GAMX3))
        IF(BETA3.GT.1.56905.AND.BETA3.LT.1.588249) BETA3=1.56905
        EEE=TAN(GAMMA)*(DIME2(10,N+1)-DIME2(9,N)*COS(ATAN(DIME2(2,N))))
        FFF=DIME2(9,N+1)*SIN(ATAN(DIME2(2,N+1)))/TAN(BETA3)
        IF(ABS(GAMMA).GE.1.57079) GO TO 69
        IF(GAMMA.LE.0.0.AND.GAMMA.GT.-1.57079) GO TO 68
        IF(EEE+DIME2(9,N+1)*COS(ATAN(DIME2(2,N+1))).GT.FFF) GO TO 71
        IF(EEE+DIME2(9,N+1)*COS(ATAN(DIME2(2,N+1))).GT.DIME2(10,N)) GO TO
172
        WAREA=((DIME2(10,N)-EEE)*DIME2(9,N)+((EEE)**2)*TAN(BETA3)/(2.0*SIN
1(ATAN(DIME2(2,N)))))/DIME2(7,N)
        RETURN
63  IF(TAN(BETA)/ABS(COS(GAMMA)).GT.DIME2(2,N)) GO TO 58
        WAREA=0.0
        RETURN
72  WAREA=(DIME2(9,N)-((FFF-DIME2(10,N))*TAN(BETA3)/SIN(ATAN(DIME2(2,N
1)))))*DIME2(10,N)/(2.0*DIME2(7,N))
        RETURN
69  IF(TAN(BETA)/ABS(COS(GAMMA)).GT.DIME2(2,N)) GO TO 70

```

```

        WAREA=0.0
        RETURN
68  WAREA=1.0
    RETURN
70  IF(GAMMA.LT.0.0) GO TO 68
    IF(DIME2(2,N+1).GT.TAN(BETA3)) GO TO 71
    WAREA=1.0
    RETURN
71  IF(FFF.GT.DIME2(10,N)) GO TO 72
    IF(TAN(BETA3).GT.DIME2(2,N+1)) GO TO 59
    WAREA=((DIME2(10,N)-(FFF/2.0))*DIME2(9,N))/DIME2(7,N)
    RETURN
C   THESE STATEMENTS ARE USED IF THERE IS NO SIDE SHADING OR THE ROOF
C   IS AN 'I' TYPE ROOF.
55  IF(DIME2(7,N).EQ.0.0) GO TO 57
    SIGMA=ATAN(DIME2(2,N))
    THETA=ARCCS((COS(BETA)*COS(GAMMA)*SIN(SIGMA))+(SIN(BETA)*COS(SIGMA
1)))
    IF(COS(THETA).LT.0.0) GOTO 57
    WAREA=1.0
    RETURN
57  WAREA=0.0
    RETURN
C   THE FOLLOWING ARE CALCULATIONS FOR A U SHAPED ROOF. THE SIGN OF
C   GAMMA TELLS THE PROGRAM WHICH DIRECTION THE SHADING WILL BE,
C   COUNTER-CLOCKWISE OR CLOCKWISE.
56  IF(GAMMA) 80,81,82
81  WAREA=1.0
    RETURN
C   THIS STATEMENT IS FOR COUNTER-CLOCKWISE SHADING FOR U SHAPED ROOFS
80  SIGX1=SIGH1-180.
    IF(SIGX1.GT.180.) SIGX1=SIGX1-360.
    SIGX1=3.14159*SIGX1/180.
    IF(PHI.LT.0.0.AND.SIGX1.GT.3.14159+PHI) SIGX1=SIGX1-6.28318
    GAMX1=PHI-SIGX1
    BETAL=ATAN(TAN(BETA)/COS(GAMX1))
    IF(BETAL.GT.1.56905.AND.BETAL.LT.1.588249) BETAL=1.56905
    GAMM=-GAMMA
    EEE=TAN(GAMM)*(DIME2(10,N-1)-DIME2(9,N)*COS(ATAN(DIME2(2,N))))
    FFF=DIME2(9,N-1)*SIN(ATAN(DIME2(2,N-1)))/TAN(BETAL)
    IF(ABS(GAMMA).GE.1.57079) GO TO 79
    IF(EEE+DIME2(9,N-1)*COS(ATAN(DIME2(2,N-1))).GT.FFF) GO TO 91
    IF(EEE.GT.DIME2(10,N)-DIME2(9,N-1)*COS(ATAN(DIME2(2,N-1)))-DIME2(9
1,N+1)*COS(ATAN(DIME2(2,N+1)))) GO TO 92
    WAREA=((COS(ATAN(DIME2(2,N+1)))*DIME2(9,N)/2.0)+(DIME2(10,N)-COS(A
1TAN(DIME2(2,N+1)))-COS(ATAN(DIME2(2,N-1)))-EEE)*DIME2(9,N)+((EEE+C
1OS(ATAN(DIME2(2,N-1))))*2)*TAN(BETAL)/(2.0*SIN(ATAN(DIME2(2,N))))
1)/DIME2(7,N)

```

```

      RETURN
79  IF(TAN(BETA)/ABS(COS(GAMMA)).GT.DIME2(2,N)) GO TO 90
      WAREA=0.0
      RETURN
91  IF(FFF.GT.DIME2(10,N)-COS(ATAN(DIME2(2,N+1)))) GO TO 92
      IF(TAN(BETA1).GT.DIME2(2,N-1)) GO TO 58
      WAREA=((COS(ATAN(DIME2(2,N+1))))*DIME2(9,N)/2.0)+((DIME2(10,N)-COS(
1  ATAN(DIME2(2,N+1)))-FFF)*DIME2(9,N))+FFF*DIME2(9,N)/2.0)/DIME2(7,N
1)
      RETURN
92  AK=DIME2(10,N)-COS(ATAN(DIME2(2,N+1)))
      WAREA=SIN(ATAN(DIME2(2,N)))*AK*AK*TAN(BETA1)/(2.0*DIME2(7,N))
      RETURN
90  IF(DIME2(2,N-1).GT.TAN(BETA1)) GO TO 91
      GO TO 58
C   THIS STATEMENT IS FOR CLOCKWISE SHADING ON A U SHAPED ROOF.
82  SIGX3=SIGH3+180.
      IF(SIGX3.GT.180.) SIGX3=SIGX3-360.
      SIGX3=3.14159*SIGX3/180.
      IF(PHI.GT.0.0.AND.SIGX3.LT.PHI-3.14159) SIGX3=SIGX3+6.28318
      GAMX3=PHI-SIGX3
      BETA3=ATAN(TAN(BETA)/COS(GAMX3))
      IF(BETA3.GT.1.56905.AND.BETA3.LT.1.588249) BETA3=1.56905
      EEE=TAN(GAMMA)*(DIME2(10,N+1)-DIME2(9,N)*COS(ATAN(DIME2(2,N))))
      FFF=DIME2(9,N+1)*SIN(ATAN(DIME2(2,N+1)))/TAN(BETA3)
      IF(ABS(GAMMA).GE.1.57079) GO TO 109
      IF(EEE+DIME2(9,N+1)*COS(ATAN(DIME2(2,N+1))).GT.FFF) GO TO 111
      IF(EEE.GT.DIME2(10,N)-DIME2(9,N+1)*COS(ATAN(DIME2(2,N+1)))-DIME2(9
1,N-1)*COS(ATAN(DIME2(2,N-1)))) GO TO 112
      WAREA=((COS(ATAN(DIME2(2,N-1))))*DIME2(9,N)/2.0)+(DIME2(10,N)-COS(A
1TAN(DIME2(2,N-1)))-COS(ATAN(DIME2(2,N+1)))-EEE)*DIME2(9,N)+((EEE+C
1OS(ATAN(DIME2(2,N+1))))*2)*TAN(BETA3)/(2.0*SIN(ATAN(DIME2(2,N))))
1)/DIME2(7,N)
      RETURN
109 IF(TAN(BETA)/ABS(COS(GAMMA)).GT.DIME2(2,N)) GO TO 110
      WAREA=0.0
      RETURN
111 IF(FFF.GT.DIME2(10,N)-COS(ATAN(DIME2(2,N-1)))) GO TO 112
      IF(TAN(BETA3).GT.DIME2(2,N+1)) GO TO 58
      WAREA=((COS(ATAN(DIME2(2,N-1))))*DIME2(9,N)/2.0)+((DIME2(10,N)-COS(
1ATAN(DIME2(2,N-1)))-FFF)*DIME2(9,N))+FFF*DIME2(9,N)/2.0)/DIME2(7,N
1)
      RETURN
112 AK=DIME2(10,N)-COS(ATAN(DIME2(2,N-1)))
      WAREA=SIN(ATAN(DIME2(2,N)))*AK*AK*TAN(BETA3)/(2.0*DIME2(7,N))
      RETURN
110 IF(DIME2(2,N+1).GT.TAN(BETA3)) GO TO 111
      GO TO 58

```

```

C *****
C *****
C WIDOWS AND DOORS
C WIDOWS AND DOORS
C THIS IS THE START OF THE WINDOW AND DOOR SHADING CALCULATIONS.
139 IF(ABS(GAMMA).GT.1.57079) GO TO 14
    I=III
    X=(TAN(BETA)/COS(GAMMA))*DIME2(8,N)
    IF(X.LT.WINDOW(N,I,3)) GO TO 130
    IF(X.GT.WINDCW(N,I,3)+WINDCW(N,I,2)) GO TO 14
121 WH=WINDOW(N,I,3)+WINDOW(N,I,2)-X
    GO TO 140
130 WH=WINDCW(N,I,2)
140 DIF1=SIGH1-SIGH2
    DIF2=SIGH2-SIGH3
C THIS STATEMENT CHECKS FOR A U SHAPED HOUSE.
    IF(DIF1.EQ.90..AND.DIF2.EQ.90.) GO TO 138
C THIS STATEMENT CHECKS FOR AN 'I' SHAPED HOUSE. NO SIDE SHADING.
    IF(DIF1.NE.90..AND.DIF2.NE.90.) GO TO 141
C THIS STATEMENT CHECKS TO FIND OUT IF THE HOUSE IS A COUNTER-
C CLOCKWISE L.
120 IF(DIF1.NE.90.) GO TO 200
C THIS CHECKS TO FIND OUT IF GAMMA IS MINUS FOR COUNTER- CLOCKWISE
C SHADING.
    IF(GAMMA.GE.0.0) GO TO 141
    SIGX1=SIGH1-180.
    IF(SIGX1.GT.180.) SIGX1=SIGX1-360.
    SIGX1=3.14159*SIGX1/180.
    IF(PHI.LT.0.0.AND.SIGX1.GT.3.14159+PHI) SIGX1=SIGX1-6.28318
    GAMX1=PHI-SIGX1
    BETA1=ATAN(TAN(BETA)/COS(GAMX1))
    IF(BETA1.GT.1.56905.AND.BETA1.LT.1.588249) BETA1=1.56905
    GAMM=-GAMMA
    IF(TAN(BETA1).LT.DIME2(2,N-1)) GO TO 160
    XXX=DIME2(8,N-1)+(WINDOW(N,I,3)+WINDOW(N,I,2))/TAN(BETA1)
    XX=DIME2(8,N-1)+WINDCW(N,I,3)/TAN(BETA1)
    AB=WINDCW(N,I,4)+WINDCW(N,I,5)
    YY=DIME2(3,N-1)*TAN(GAMM)
400 IF(XX.GT.AB) GO TO 150
    IF(XX.GT.WINDOW(N,I,4)) GO TO 142
    IF(XXX.LT.WINDOW(N,I,4)) GO TO 141
    IF(XXX.GT.AB) GO TO 129
    IF(YY.GT.XXX) GO TO 123
    IF(YY.LT.WINDOW(N,I,4)) GO TO 141
    ZZ=(XXX-YY)*TAN(BETA1)
    WW=(XXX-WINDCW(N,I,4))*TAN(BETA1)
    IF(WH.LT.ZZ) GO TO 150
    IF(WH.GT.WW) GO TO 126

```

```

        WAREA=((AB-YY)*WH+((WH-ZZ)**2)/(2.0*TAN(BETA1)))/WINDOW(N,I,8)
        RETURN
126 WAREA=((AB-YY)*WH+((2.0*WH-ZZ-WW)*(YY-WINDOW(N,I,4)))/2.0)/WINDOW(
    N,I,8)
    RETURN
123 IF(WH.GT.(XXX-WINDOW(N,I,4))*TAN(BETA1)) GO TO 124
    WAREA=((WH*(AB-XXX))+((WH**2)/(2.0*TAN(BETA1)))/WINDOW(N,I,8)
    RETURN
124 WAREA=(WH*(AB-XXX)+((WH-(XXX-WINDOW(N,I,4))*TAN(BETA1)/2.0)*(XXX-WI
    NDO(N,I,4)))/WINDOW(N,I,8)
    RETURN
129 UU=(XXX-AB)*TAN(BETA1)
    VV=(XXX-WINDOW(N,I,4))*TAN(BETA1)
    IF(WH.LT.UU) GO TO 127
    IF(WH.GT.VV) GO TO 128
    IF(YY.GT.AB) GO TO 133
    IF(YY.LT.WINDOW(N,I,4)) GO TO 141
    ZZZ=XXX-(WH/TAN(BETA1))
    IF(YY.LT.ZZZ) GO TO 150
    ZZ=(XXX-YY)*TAN(BETA1)
    WAREA=((AB-YY)*WH+((WH-ZZ)**2)/(2.0*TAN(BETA1)))/WINDOW(N,I,8)
    RETURN
133 WAREA=((WH-UU)**2)/(2.0*TAN(BETA1)*WINDOW(N,I,8))
    RETURN
127 IF(YY.GT.AB) GO TO 14
    IF(YY.LT.WINDOW(N,I,4)) GO TO 141
    GO TO 150
128 IF(YY.LT.WINDOW(N,I,4)) GO TO 141
    IF(YY.GT.AB) GO TO 132
    ZZ=(XXX-YY)*TAN(BETA1)
    GO TO 126
132 WAREA=(2.*WH-UU-VV)*WINDOW(N,I,5)/(2.*WINDOW(N,I,8))
    RETURN
150 IF(YY.GT.AB) GO TO 14
    IF(YY.LT.WINDOW(N,I,4)) GO TO 141
152 WAREA=(WH/WINDOW(N,I,2))*(1.0-((YY-WINDOW(N,I,4))/WINDOW(N,I,5)))
    RETURN
153 IF(YY.GT.WINDOW(N,I,4)) GO TO 152
    GO TO 141
142 IF(XXX.LT.AB) GO TO 143
    IF(YY.LT.AB) GO TO 145
    WAREA=((AB-XX)**2)*TAN(BETA1)/(2.0*WINDOW(N,I,8))
    RETURN
144 WAREA=((AB-XXX)+WH/(2.0*TAN(BETA1)))*WH/WINDOW(N,I,8)
    RETURN
143 IF(YY.GT.XXX) GO TO 144
    IF(YY.LT.XX) GO TO 153
147 IF(WH.GT.(XXX-YY)*TAN(BETA1)) GO TO 146

```

```

        WAREA=(AB-YY)*WH/WINDOW(N,I,8)
        RETURN
146 WAREA=((AB-YY)*WH+((WH-(XXX-YY)*TAN(BETA1))**2)/(2.0*TAN(BETA1)))/
    WINDOW(N,I,8)
    RETURN
145 IF(YY.LT.XX) GO TO 150
    GO TO 147
C     THIS STATEMENT IS USED IF NO SIDE SHADING OCCURS ONLY OVERHANG
C     SHADING.
141 WAREA=WH/WINDOW(N,I,2)
    RETURN
138 IF(GAMMA) 120,141,200
160 XX=(SIN(ATAN(DIME2(2,N-1)))*DIME2(9,N-1)+WINDOW(N,I,3))/TAN(BETA1)
    1-CCS(ATAN(DIME2(2,N-1)))*DIME2(9,N-1)+DIME2(8,N-1)
    XXX=(SIN(ATAN(DIME2(2,N-1)))*DIME2(9,N-1)+WINDOW(N,I,3)+WINDOW(N,I
    1,2))/TAN(BETA1)-CCS(ATAN(DIME2(2,N-1)))*DIME2(9,N-1)+DIME2(8,N-1)
    AB=WINDOW(N,I,4)+WINDOW(N,I,5)
    YY=DIME2(3,N-1)*TAN(GAMMA)
    GO TO 400
C     THIS STATEMENT CHECKS FOR CLOCKWISE L SHAPED HOUSES.
200 IF(DIF2.NE.90.) GO TO 141
C     THIS STATEMENT CHECKS TO SEE IF GAMMA IS POSITIVE FOR CLOCKWISE
C     SHADING TO OCCUR.
    IF(GAMMA.LT.0.0) GO TO 141
    SIGX3=SIGH3+180.
    IF(SIGX3.GT.180.) SIGX3=SIGX3-360.
    SIGX3=3.14159*SIGX3/180.
    IF(PHI.GT.0.0.AND.SIGX3.LT.PHI-3.14159) SIGX3=SIGX3+6.28318
    GAMX3=PHI-SIGX3
    BETA3=ATAN(TAN(BETA)/COS(GAMX3))
    IF(BETA3.GT.1.56905.AND.BETA3.LT.1.588249) BETA3=1.56905
    IF(TAN(BETA3).LT.DIME2(2,N+1)) GO TO 260
    XXX=DIME2(8,N+1)+(WINDOW(N,I,3)+WINDOW(N,I,2))/TAN(BETA3)
    XX=DIME2(8,N+1)+WINDOW(N,I,3)/TAN(BETA3)
    AB=WINDOW(N,I,6)+WINDOW(N,I,5)
    YY=DIME2(3,N+1)*TAN(GAMMA)
401 IF(XX.GT.AB) GO TO 250
    IF(XX.GT.WINDOW(N,I,6)) GO TO 242
    IF(XXX.LT.WINDOW(N,I,6)) GO TO 141
    IF(XXX.GT.AB) GO TO 229
    IF(YY.GT.XXX) GO TO 223
    IF(YY.LT.WINDOW(N,I,6)) GO TO 141
    ZZ=(XXX-YY)*TAN(BETA3)
    WW=(XXX-WINDOW(N,I,6))*TAN(BETA3)
    IF(WH.LT.ZZ) GO TO 250
    IF(WH.GT.WW) GO TO 226
    WAREA=((AB-YY)*WH+((WH-ZZ)**2)/(2.0*TAN(BETA3)))/WINDOW(N,I,8)
    RETURN

```

```

226 WAREA=((AB-YY)*WH+((2.0*WH-ZZ-WH)*(YY-WINDOW(N,I,6)))/2.)/WINDOW(N
1,I,8)
RETURN
223 IF(WH.GT.(XXX-WINDOW(N,I,6))*TAN(BETA3)) GO TO 224
WAREA=((WH*(AB-XXX))+[WH**2]/(2.*TAN(BETA3)))/WINDOW(N,I,8)
RETURN
224 WAREA=(WH*(AB-XXX)+(WH-(XXX-WINDOW(N,I,6))*TAN(BETA3)/2.)*(XXX-WIN
1DOW(N,I,6)))/WINDOW(N,I,8)
RETURN
229 UU=(XXX-AB)*TAN(BETA3)
VV=(XXX-WINDOW(N,I,6))*TAN(BETA3)
IF(WH.LT.UU) GO TO 227
IF(WH.GT.VV) GO TO 228
IF(YY.GT.AB) GO TO 233
IF(YY.LT.WINDOW(N,I,6)) GO TO 141
ZZ=XXX-(WH/TAN(BETA3))
IF(YY.LT.ZZ) GO TO 250
ZZ=(XXX-YY)*TAN(BETA3)
WAREA=((AB-YY)*WH+((WH-ZZ)**2)/(2.*TAN(BETA3)))/WINDOW(N,I,8)
RETURN
233 WAREA=((WH-LU)**2)/(2.*TAN(BETA3)*WINDOW(N,I,8))
RETURN
227 IF(YY.GT.AB) GO TO 14
IF(YY.LT.WINDOW(N,I,6)) GO TO 141
GO TO 250
228 IF(YY.LT.WINDOW(N,I,6)) GO TO 141
IF(YY.GT.AB) GO TO 232
ZZ=(XXX-YY)*TAN(BETA3)
GO TO 226
232 WAREA=(2.*WH-UU-VV)*WINDOW(N,I,5)/(2.*WINDOW(N,I,8))
RETURN
250 IF(YY.GT.AB) GO TO 14
IF(YY.LT.WINDOW(N,I,6)) GO TO 141
252 WAREA=(WH/WINDOW(N,I,2))*(1.-((YY-WINDOW(N,I,6))/WINDOW(N,I,5)))
RETURN
253 IF(YY.GT.WINDOW(N,I,6)) GO TO 252
GO TO 141
242 IF(XXX.LT.AB) GO TO 243
IF(YY.LT.AB) GO TO 245
WAREA=((AB-XX)**2)*TAN(BETA3)/(2.*WINDOW(N,I,8))
RETURN
244 WAREA=((AB-XXX)+WH/(2.*TAN(BETA3)))*WH/WINDOW(N,I,8)
RETURN
243 IF(YY.GT.XXX) GO TO 244
IF(YY.LT.XX) GO TO 253
247 IF(WH.GT.(XXX-YY)*TAN(BETA3)) GO TO 246
WAREA=(AB-YY)*WH/WINDOW(N,I,8)
RETURN

```

```

246 WAREA=((AB-YY)*WH+((WH-(XXX-YY)*TAN(BETA3))**2)/(2.*TAN(BETA3)))/W
      WINDOW(N,I,8)
      RETURN
245 IF(YY.LT.XX) GO TO 250
      GO TO 247
260 XX=(SIN(ATAN(DIME2(2,N+1)))*DIME2(9,N+1)+WINDOW(N,I,3))/TAN(BETA3)
      1-COS(ATAN(DIME2(2,N+1)))*DIME2(9,N+1)+DIME2(8,N+1)
      XXX=(SIN(ATAN(DIME2(2,N+1)))*DIME2(9,N+1)+WINDOW(N,I,3)+WINDOW(N,I
      1,2))/TAN(BETA3)-COS(ATAN(DIME2(2,N+1)))*DIME2(9,N+1)+DIME2(8,N+1)
      AB=WINDOW(N,I,5)+WINDOW(N,I,6)
      YY=DIME2(3,N+1)*TAN(GAMMA)
      GO TO 401
      END

```



```

SUBROUTINE SHAFRA
C THIS SUBROUTINE IS AN OPTICNAL SUBROUTINE THAT CAN BE CALLED BUT
C DOES NOT HAVE TO BE. THIS SUBROUTINE'S ONLY FUNCTION IS TO PRINT
C OUT INFORMATION. THIS INFORMATION INCLUDES WALL DIMENSIONS
C CONTAINED IN THE MATRIX DIME2(N,J), THE WINDOW AND DOOR DIMENSIONS
C IN THE MATRIX WINDOW(I,J,K), AND THE PERCENT AREA SUNLIT FOR WALLS
C , ROOFS, WINDOWS, AND DOORS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15),NUMC(15)
COMMON/AIRC1/NOWIN(15),NCDGOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
132 FORMAT(' WALL NUMBER ',14I7)
201 FORMAT(' WALL I.D. NUMBER ',14F7.2)
202 FORMAT(' ROOF SLOPE ',14F7.2)
203 FORMAT(' WALL LENGTH ',14F7.2)
204 FORMAT(' WALL HEIGHTH ',14F7.2)
205 FORMAT(' WALL TILT ',14F7.2)
206 FORMAT(' WALL AREA ',14F7.2)
207 FORMAT(' ROOF AREA ',14F7.2)
208 FORMAT(' ROOF OVERHANG ',14F7.2)
209 FORMAT(' ROOF WIDTH ',14F7.2)
210 FORMAT(' ROOF LENGTH ',14F7.2)
301 FORMAT(' LEN BEL WINDOW ',14F7.2)
302 FORMAT(' WINDOW HEIGHTH ',14F7.2)
303 FORMAT(' LEN ABOVE WINDOW ',14F7.2)
304 FORMAT(' LEN R. OF WINDOW ',14F7.2)
305 FORMAT(' WIDTH OF WINDOW ',14F7.2)
306 FORMAT(' LEN L. OF WINDOW ',14F7.2)
307 FORMAT(' TYPE OF WINDOW ',14F7.2)
308 FORMAT(' WINDOW AREA ',14F7.2)
134 FORMAT(18,15F7.3)
135 FORMAT(3X,'TIME',1X,15I7)
136 FORMAT('1',29X,'THE HOUSE HAS BEEN ROTATED',F6.1,'DEGREES CLOCKWIS
1E')
138 FORMAT(30X,'THIS IS THE ',12,' MONTH')
139 FORMAT(1X,'THESE ARE THE SUNLITE FRACTIONS OF THE WALL AREAS')
140 FORMAT(1X,'THESE ARE THE DIMENSIONS FOR THE HOUSE UNDER EVALUATION
1')
141 FORMAT(1X,'THESE ARE THE SUNLITE FRACTION OF THE ROOF AREAS')
143 FORMAT(///' THESE ARE THE DIMENSIONS THAT DESCRIBE THE WINDOWS')
144 FORMAT(' FOR WALL NUMBER',13)
145 FORMAT(' WINDOW OR DOOR NO.',10I7)
146 FORMAT(///' THE FOLLOWING ARE THE DECIMAL FRACTIONS THAT THE WINDO
1WS AND DOORS ARE SUNLITE')
147 FORMAT(///' WINDOWS')
148 FORMAT(///' DOORS')
WRITE(6,138) M
WRITE(6,140)

```

```

WRITE(6,132)(NUMB(I),I=1,NNN)
WRITE(6,201)(DIME2(1,J),J=1,NNN)
WRITE(6,202)(DIME2(2,J),J=1,NNN)
WRITE(6,203)(DIME2(3,J),J=1,NNN)
WRITE(6,204)(DIME2(4,J),J=1,NNN)
WRITE(6,205)(DIME2(5,J),J=1,NNN)
WRITE(6,206)(DIME2(6,J),J=1,NNN)
WRITE(6,207)(DIME2(7,J),J=1,NNN)
WRITE(6,208)(DIME2(8,J),J=1,NNN)
WRITE(6,209)(DIME2(9,J),J=1,NNN)
WRITE(6,210)(DIME2(10,J),J=1,NNN)
WRITE(6,143)
DO 20 J=2,N1
K=NOWIN(J)+NODCCR(J)
IF(K.EQ.0) GO TO 20
K1=K+1
L=J-1
WRITE(6,144) L
WRITE(6,145)(NUMC(I),I=2,K1)
WRITE(6,301)(WINDOW(J,I,1),I=1,K)
WRITE(6,302)(WINDOW(J,I,2),I=1,K)
WRITE(6,303)(WINDOW(J,I,3),I=1,K)
WRITE(6,304)(WINDOW(J,I,4),I=1,K)
WRITE(6,305)(WINDOW(J,I,5),I=1,K)
WRITE(6,306)(WINDOW(J,I,6),I=1,K)
WRITE(6,307)(WINDOW(J,I,7),I=1,K)
WRITE(6,308)(WINDOW(J,I,8),I=1,K)
20 CONTINUE
WRITE(6,136) AA
WRITE(6,139)
WRITE(6,135)(NUMB(J),J=2,N1)
DO 50 ITIM=1,24
50 WRITE(6,134)ITIM,(ALITE(ITIM,I),I=2,N1)
WRITE(6,141)
WRITE(6,135)(NUMB(J),J=2,N1)
DO 60 ITIP=1,24
60 WRITE(6,134)ITIP,(BLITE(ITIP,I),I=2,N1)
WRITE(6,146)
DO 61 J=2,N1
K1=NOWIN(J)
K11=K1+1
K2=NODCCR(J)
K3=K2+K1
IF(K3.EQ.0) GO TO 61
L=J-1
WRITE(6,144) L
K22=K2+1
IF(K1.EQ.0) GO TO 62

```

```
WRITE(6,147)
WRITE(6,135)(NUMC(J1),J1=2,K11)
DO 63 ITIM=1,24
63 WRITE(6,134)ITIM,(WRITE(ITIM,J,I5),I5=1,K1)
62 IF(K2.EQ.0) GO TO 61
WRITE(6,148)
WRITE(6,135)(NUMC(J1),J1=2,K22)
DO 64 ITIM=1,24
64 WRITE(6,134) ITIM,(DLITE(ITIM,J,I5),I5=1,K2)
61 CONTINUE
RETURN
END
```

```

SUBROUTINE SOLTEM
C THIS SUBROUTINE CALCULATES ADJUSTED SOL-AIR TEMPERATURES. IT CALLS
C A FUNCTION NAMED SCLAIR(X) WHICH CALCULATES SOLAIR TEMPERATURES.
C SOME OF THE VARIABLES USED ARE:
C ALITE-BLITE-WLITE-DLITE-THESE MATRICES CONTAIN WHAT PERCENT AREA
C IS SUNLIT OF THE SURFACE.
C TEMP-THIS ARRAY CONTAINS THE OUTSIDE TEMPERATURE PROFILE.
C SOLAVW-THIS MATRIX CCNTAINS THE ADJUSTED SOLAR HEAT GAINS ON THE
C WINDOWS.
C RTM-ARE THE ADJUSTED SOL-AIR TEMPERATURES FOR THE WALLS.
C WTM-ARE THE ADJUSTED SOL-AIR TEMPERATURES FOR THE ROOFS.
C DTM-ARE THE ADJUSTED SOL-AIR TEMPERATURES FOR THE DOORS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),NODCOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC6/WTM(30,15),SRTM(15),ARTM(15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLOD(15)
COMMON/AIRC8/SJLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QDOOR(24,15,10),QWIND(24,15,10)
COMMON/AC1/SADCOF(15,10)
COMMON/AIRC3/QLOT(30,15),RQLOT(30)
C THESE VALUES ARE INITIALIZED TO ZERO AT THIS POINT BECAUSE THEY
C ARE USED AS SUMMATION VARIABLES LATER.
DO 490 N=2,N1
  ARTM(N)=0.0
  SRTM(N)=0.0
  AQLOD(N)=0.0
DO 490 ITIM=1,30
490 QLOD(ITIM,N)=0.0
DO 11 N=2,N1
DO 10 ITIN=7,30
  ITIM=ITIN-6
  T=TEMP(ITIM)
C ICHEK=1 FOR WALLS IN SOLGAN(X) AND SCLAIR(X).
  ICHEK=1
C JCHEK=1 FOR SUNLIT CASE OF SOLGAN(X).
  JCHEK=1
  RTM(ITIN,N)=T+ALITE(ITIM,N)*(SOLAIR(X)-T)
C ICHEK=2 FOR ROOFS IN SOLGAN(X) AND SCLAIR(X).
  ICHEK=2
C WTM(ITIN,N) START ITIN IN POSITION 7 BECAUSE THE RESPONSE FACTER
C METHOD NEEDS THE LAST SIX VALUES OF THE DAY IN THE FIRST SIX
C LOCATIONS.
  WTM(ITIN,N)=T+BLITE(ITIM,N)*(SOLAIR(X)-T)
  SRTM(N)=SRTM(N)+RTM(ITIN,N)
10 CONTINUE

```

```

C      THE NEXT STATEMENT CALCULATES THE AVERAGE SOLAIR TEMPERATURE ON
C      EACH WALL DURING THE DAY.
      ARTM(N)=SRTM(N)/24,0
11  CONTINUE
      DO 12 N=2,N1
        DO 12 K=1,6
          RTM(K,N)=RTM(K+24,N)
12  WTM(K,N)=WTM(K+24,N)
      DO 20 ITIM=1,24
        DO 20 N=2,N1
          JCHEK=1
          ICHEK=1
          SS=SOLGAN(X)
C      JCHEK=2 FOR SHADED CASE OF SOLGAN(X).
          JCHEK=2
          ST=SOLGAN(X)
          KK=NOWIN(N)
          KK5=NCDCCR(N)
          IF(KK.EQ.0) GO TO 54
          DO 52 III=1,KK
            WS=WLITE(ITIM,N,III)
52  SOLAVW(ITIM,N,III)=SS*WS+(1.0-WS)*ST
54  IF(KK5.EQ.0) GO TO 20
          DO 53 MM=1,KK5
            III=MM+KK
            T=TEMP(ITIM)
            ICHEK=1
            JCHEK=1
53  DTM(ITIM,N,MM)=T+DLITE(ITIM,N,MM)*(SOLAIR(X)-T)
20  CONTINUE
      RETURN
      END

```

```

C      FUNCTION SOLGAN(X)
C      THIS FUNCTION CALCULATES SOLAR HEAT GAINS FROM A METHOD OUTLINED
C      IN THE 1972 ASHRAE HANDBOOK OF FUNDAMENTALS
C      LIST OF VARIABLES:
C          M-MONTH OF THE YEAR (1-12)
C          SIGMA-WALL TILT ANGLE
C          AH-THE NUMBER OF MINUTES AFTER NOON , MINUTES BEFORE
C      NOON ARE NEGATIVE.
C          PSI-WALL AZIMUTH.
C          AI-WALL I.D. NUMBER.
C          BB-SOLAR ALTITUDE
C          PHI-SOLAR AZIMUTH.
C          THETA-ANGLE OF INCIDENCE
C          AIDN-DIRECT NORMAL SOLAR RADIATION
C          FSG-SHAPE FACTOR.
C      COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
C      AI=DIME2(1,N)
C      IF(M.EQ.8) GO TO 5
C      IF(M.EQ.7) GO TO 6
C      IF(M.EQ.12) GO TO 3
C      IF(M.EQ.1) GO TO 4
C      SOLGAN=-1.0
C      RETURN
3  A=391.
   B=0.142
   C=0.057
   DEG=-0.40927
   GO TO 10
4  A=390.
   B=0.142
   C=0.058
   DEG=-0.34907
   GO TO 10
5  A=351.0
   B=0.201
   C=0.122
   DEG=0.214675
   GO TO 10
6  A=344.0
   B=0.207
   C=0.136
   DEG=0.359537
   GO TO 10
C      WHEN ICHK IS EQUAL TO 1 THEN THE SOLGAN IS CALCULATED FOR WALLS.
10 IF(ICHEK.EQ.1) GO TO 7
   GO TO 8
7  SIGMA=ATAN(DIME2(5,N))
   GO TO 11

```

```

8  SIGMA=ATAN(DIME2(2,N))
11 AL=ITIM
C  THIS IS THE CALCULATION FOR THE HOUR ANGLE
  AH=((AL-12.0)*60.0)*0.25)
C  AH1 IS THE HOUR ANGLE A SMALL FINITE AMOUNT BEFORE AH.
C  THIS IS USED TO CALCULATE PHI SMALLER THAN -90. DEGREES AND LARGER
C  THAN 90. DEGREES.
  AH1=AH-.005
  PSI=(90.0*(AL-1.0))+AA
  IF(PSI.GT.360.) PSI=PSI-360.
  IF(PSI.GT.180.0) PSI=PSI-360.0
  IF(PSI.LT.-180.0) PSI=360.0-PSI
  AH=3.14159*AH/180.0
  AH1=3.14159*AH1/180.
  BBB=(COS(ALAT)*COS(DEG)*COS(AH))+(SIN(ALAT)*SIN(DEG))
  BBB1=(COS(ALAT)*COS(DEG)*COS(AH1))+(SIN(ALAT)*SIN(DEG))
  BB=ARSIN(BBB)
  BB1=ARSIN(BBB1)
  IF(BB.LE.0.0) GO TO 15
  GO TO 14
15 SOLGAN=0.0
  RETURN
14 PHI=ARSIN(COS(DEG)*SIN(AH)/COS(BB))
  PHI1=ARSIN(COS(DEG)*SIN(AH1)/COS(BB1))
C  WITH THE USE OF AH1 ANOTHER SOLAR ALTITUDE AND SOLAR AZIMUTH CAN
C  BE CALCULATED . IF THE EARLIER PHI (AH1) IS NOT SMALLER THAN
C  LATER PHI(AH), THEN PHI(AH) SHOULD BE EITHER LESS THAN -90. OR
C  GREATER THAN 90.. SPECIAL TREATMENT MUST BE TAKEN FOR THESE CASES
C  BECAUSE THE PRINCIPLE VALUES ARE THE VALUES CALCULATED FOR PHI.
C  THIS CHECK STATEMENT FINDS THESE CASES AND CORRECTS THE VALUES FOR
C  PHI.
  IF(PHI1.LT.PHI) GO TO 30
  IF(AH) 12,12,20
12 PHI=-3.14159-PHI
  GO TO 30
20 PHI=3.14159-PHI
30 PSI=PSI*3.14159/180.0
  IF(PHI.LT.0.0.AND.PSI-4.71239.GT.PHI) PSI=PSI-6.2831
  IF(PHI.GT.0.0.AND.PSI+4.71239.LT.PHI) PSI=PSI+6.2831
  GAMMA=PHI-PSI
  IF(ABS(GAMMA).GT.1.57079.AND.ICHEK.EQ.1) GO TO 25
  IF(JCHEK.EQ.2) GO TO 25
C  IF THE VALUE OF THETA IS GREATER THAN 90 DEGREES, THEN THE SURFACE
C  IS IN THE SHADE OR THAT THE COS (THETA) IS EQUAL TO ZERO.
  THETA=ARCOS((COS(BB)*COS(GAMMA)*SIN(SIGMA))+(SIN(BB)*COS(SIGMA)))
  IF(COS(THETA).LT.0.0) THETA=1.57079
  GO TO 26
25 THETA=1.57079

```

```
26 AIDN=A/EXP(B/SIN(BB))  
   FSG=(1.0-COS(SIGMA))/2.0  
   SOLGAN=(AIDN*COS(THETA))+(C*AIDN*(1.0-FSG))+(0.1*FSG*AIDN*(C+SIN(B  
18)))  
   RETURN  
   END
```



```

FUNCTION SOLAIR(X)
C   THIS FUNCTION CALCULATES THE SOL-AIR TEMPERATURE FOR SUNLIT AND
C   SHADED SURFACES. IT ALSO CALCULATES SOL-AIR TEMPERATURES ON ANY
C   VERTICAL SURFACE SUCH AS A WALL, WINDOW, CR DOOR AND SOL-AIRS ON
C   NON-VERTICAL SURFACES WHEN ICHEK=2.
C   THESE CALCULATIONS ARE OUTLINED IN THE ASHRAE GUIDE 1972.
C   THE ARGUMENT FOR THIS FUNCTION IS NOTHING MORE THAN A DUMMY
C   VARIABLE. ALL OF THE NEEDED VALUES ARE CONTAINED IN THE COMMON
C   STATEMENTS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),NODOOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
COMMON/WALPRO/B0,B1,B2,B3,B4,B5,B6,BB0,BB1,BB2,BB3,BB4,BB5,BB6
COMMON/WALPRI/D1,D2,D3,D4,D5,D6,DD1,DD2,DD3,DD4,DD5,DD6,SC,SSC
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC6/WTM(30,15),SRTM(15),ARTM(15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLOD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QDOOR(24,15,10),QWIND(24,15,10)
COMMON/AC1/SACCCF(15,10)
COMMON/AIRC3/QLDT(30,15),RQLET(30)
IF(ICHEK.EQ.1) GO TO 25
IF(ICHEK.EQ.2) GO TO 26
30  ALPHA=0.22
    SOLAIR=T+ALPHA*1.15*SOLGAN(X)-A
    RETURN
C   FOR VERTICAL SURFACES A=0.0
25  A=0.0
    GO TO 30
C   FOR NON-VERTICAL SURFACES A=7.0
26  A=7.0
    GO TO 30
END

```

```

SUBROUTINE LOAD
C THIS SUBROUTINE CALCULATES CEILING AND WALL HEAT GAIN PROFILES
C FOR A TWENTY-FOUR HOUR PERIOD.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
COMMON/AIRC1/NOWIN(15),NOCCGR(15),WINDOW(15,10,8),NUM8(15),KODE
COMMON/WALPRO/B0,B1,B2,B3,B4,B5,B6,BB0,BB1,BB2,BB3,BB4,BB5,BB6
COMMON/WALPR1/D1,D2,D3,D4,D5,D6,DD1,DD2,DD3,DD4,DD5,DD6,SC,SSC
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC6/WTM(30,15),SRTM(15),ARTM(15)
COMMON/AIRC7/CLCD(30,15),RQLCD(30),AQLCD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AC21/TA(30),NOCE,NOWA(15),CEAR,VCLAIR,TR(30),WT(30)
COMMON/AIR3/QLCT(30,15),RQLCT(30)
COMMON/AIR12/TOWLAT(24,15),TOWALL(24,15),FACTOR,COLO(28)
COMMON/AIR13/COCL(28),CT(24),TOHO(24)
100 FORMAT(' SUBROUTINE LOAD INTERATED ',I3,' TIME')
101 FORMAT(10X,F8.3,10X,F8.3,3X,I3,3X,I3)
102 FORMAT(' THE NUMBER OF THE ROOF IS EQUAL TO ',I3,' WHICH IS AN UN
1ACCEPTABLE VALUE')
103 FORMAT(' THE VALUE FOR WALL ',I3,' IS',I3,' WHICH IS UNACCEPTABLE
1')
104 FORMAT(1X,I3,3X,'TIME',6X,'TOUTSIDE AV.SOLAIR',4X,'ROOF TEMP',4X,
1'ATTIC TEMP',5X,'Q CEILING')
105 FORMAT(8X,I2,8X,F6.2,4X,F6.2,7X,F7.2,7X,F7.2,6X,F9.3)
C NOCE=1 REPRESENTS AN UNINSULATED CEILING WITH 1/2 INCH OF PLASTER
C BOARD.
IF(NOCE.EQ.1) GO TO 301
C NOCE=2 REPRESENTS A CEILING WITH 1/2 INCH PLASTER BOARD AND 3
C INCHES OF INSULATION.
IF(NOCE.EQ.2) GO TO 302
C NOCE=3 REPRESENTS A CEILING WITH 6 INCHES OF INSULATION AND 1/2
C INCH PLASTER BOARD.
IF(NOCE.EQ.3) GO TO 303
WRITE(6,102) NOCE
KODE=1
RETURN
301 UCF=0.681
GO TO 49
302 BB0=0.031995
BB1=0.038336
BB2=0.000896
BB3=0.0
BB4=0.0
BB5=0.0
BB6=0.0
DD1=-0.183051
DD2=0.000025

```

```

      CD3=0.0
      DD4=0.0
      CD5=0.0
      DD6=0.0
      SSC=0.071228
      UCE=0.087
      GO TO 49
303  RB0=0.005714
      BB1=0.022629
      BB2=0.004812
      BB3=0.000035
      BB4=0.0
      BB5=0.0
      BB6=0.0
      DD1=-0.304337
      DD2=0.016886
      DD3=-0.000001
      DD4=0.0
      DD5=0.0
      DD6=0.0
      SSC=0.033189
      UCE=0.047
      GO TO 49
49  CONTINUE
C   THIS IS THE WEIGHT OF THE ROOF PER SQUARE FOOT OF ROOF.
      ABC=4.166667
C   CPR IS THE SPECIFIC HEAT OF THE ROOF MATERIALS PER POUND.
      CPR=0.33
      ROARE=C.0
      Q=UCE*10.
      FA=0.4
      HO=4.0
C   THE TOTAL ROOF AREA IS SUMMED IN THE FOLLOWING LOOP AND THE TOTAL
C   IS CONTAINED IN THE VARIABLE ROARE
      DO 50 N=2,N1
      ROARE=ROARE+DIME2(7,N)
50  CONTINUE
      RCMAS=ROARE*ABC
      AMAS=0.075*VCLAIR
C   THE FOLLOWING LOOP (51) CALCULATES AN AREA WEIGHTED AVERAGE OF THE
C   SOL-AIR TEMPERATURES ON ALL OF THE ROOFS.
      DO 51 ITIM=7,30
      W=0.0
      DO 52 N=2,N1
      W=W+WTM(ITIM,N)*DIME2(7,N)
52  CONTINUE
      WT(ITIM)=W/ROARE
51  CONTINUE

```

```

C      LOOP (66) INITIALIZES THE FIRST GUESS FOR THE CEILING LOAD
C      CALCULATIONS.
      DO 66 ITIM=1,6
        WT(ITIM)=WT(ITIM+24)
        TA(ITIM)=85.0
        TR(ITIM)=100.
        RQLOD(ITIM)=Q
66      CONTINUE
C      KK1 IS THE NUMBER OF TIMES THE ATTIC AND ROOF TEMPERATURES ARE
C      CALCULATED EACH HOUR.
      KK1=10
      IJ=0
13      CONTINUE
      IJ=IJ+1
      WRITE(6,104) IJ
      DO 11 ITIM=7,30
        SOL=WT(ITIM-1)
        DIF=WT(ITIM)-SOL
        TEMM=TA(ITIM-1)
        TEMR=TR(ITIM-1)
C      FOR CASE ONE THE 'U'- VALUE IS SO LARGE THAT THE Q (HEAT FLUX)
C      MUST BE RECALCULATED FOR EACH TIME INTERVAL, IF NOT , TOO MUCH
C      HEAT WILL BE REMOVED IN AN HOURS TIME.
        IF(NOC.EQ.1) GO TO 203
        GO TO 204
203      DO 205 II=1, KK1
        AII=II
        TEMM=TEMM+(ROARE*HA*(TEMR-TEMM)-UCE*CEAR*(TEMM-ROGMT))/(AMAS*0.24*
1      KK1)
        AKK1=KK1
        WF=SOL+DIF*AII/AKK1
        TEMR=TEMR+(HO*(WF-TEMR)-HA*(TEMR-TEMM))*ROARE/(ROMAS*CPR*KK1)
205      CONTINUE
        GO TO 206
204      DO 63 II=1, KK1
        AII=II
        TEMM=TEMM+(ROARE*HA*(TEMR-TEMM)-RQLOD(ITIM-1)*CEAR)/(AMAS*0.24*KK1
1      )
        WF=SOL+DIF*AII/AKK1
        TEMR=TEMR+(HO*(WF-TEMR)-HA*(TEMR-TEMM))*ROARE/(ROMAS*CPR*KK1)
63      CONTINUE
206      TR(ITIM)=TEMR
        TA(ITIM)=TEMM
C      SINCE CEILING TYPE 1 IS VERY LIGHT IT TRANSMITTS HEAT
C      INSTANTANEOUSLY; THEREFORE, THIS CEILING HAS NO RESPONSE FACTORS.
C      THIS STATEMENT CHECKS TO FIND OUT IF THIS SPECIAL CEILING IS BEING
C      USED.
        IF(NOC.EQ.1) GO TO 201

```

```

      RQLQD(ITIM)=RQLCD(XX)
      GO TO 202
201 RQLCD(ITIM)=UCE*(TA(ITIM)-RCQMT)
202 ITI=ITIM-6
      WRITE(6,105) ITI,TEMP(ITI),WT(ITIM),TR(ITIM),TA(ITIM),RQLQD(ITIM)
11 CONTINUE
      IF(IJ.GT.4) GO TO 12
C     LOOP.(14) TAKES THE LAST 6 VALUES OF THE CALCULATIONS AND PLACES
C     THEM IN THE FIRST 6 LOCATIONS SO THAT THE NEXT SET OF CALCULATIONS
C     WILL HAVE THE BEST VALUES TO WORK WITH.
      DO 14 ITIM=1,6
      TA(ITIM)=TA(ITIM+24)
      TR(ITIM)=TR(ITIM+24)
      RQLQD(ITIM)=RQLCD(ITIM+24)
14 CONTINUE
      GO TO 13
12 CONTINUE
      DO 70 ITIM=6,30
      RQLCT(ITIM)=RQLQD(ITIM)*CEAR
70 CONTINUE
      K=0
      DO 493 N=2,N1
      JCHEK=0
      K=NOWA(N)
C     FOR NOWA(N)=1 THE WALL IS AN UNINSULATED FRAME WALL.
      IF(K.EQ.1) GO TO 401
C     FOR NOWA(N)=2 THE WALL IS A FRAME WALL WITH 1" OF INSULATION.
      IF(K.EQ.2) GO TO 402
C     FOR NOWA(N)=3 THE WALL IS A FRAME WALL WITH 2" OF INSULATION.
      IF(K.EQ.3) GO TO 403
C     FOR NOWA(N)=4 THE WALL IS A FRAME WALL WITH 3" OF INSULATION.
      IF(K.EQ.4) GO TO 404
C     FOR NOWA(N)=5 THE WALL IS A BRICK VENEERED FRAME WALL WITH NO
C     INSULATION.
      IF(K.EQ.5) GO TO 405
C     FOR NOWA(N)=6 THE WALL IS A BRICK VENEERED FRAME WALL WITH 3" OF
C     INSULATION.
      IF(K.EQ.6) GO TO 406
      L=N-1
      WRITE(6,103) L,K
      KODE=1
      RETURN
401 B0=0.07874
      B1=0.18185
      B2=0.02157
      B3=0.00001
      B4=0.0
      B5=0.0

```

```
B6=0.0
C1=-0.37759
D2=0.02246
C3=0.0
C4=0.0
D5=0.0
C6=0.0
SC=0.28217
U(N)=0.424
GO TO 494
402 B0=0.02069
    B1=0.06369
    B2=0.01131
    B3=0.00001
    B4=0.0
    B5=0.0
    B6=0.0
    C1=-0.53187
    D2=0.06834
    C3=0.0
    D4=0.0
    D5=0.0
    C6=0.0
    SC=0.09570
    U(N)=0.178
    GO TO 494
403 B0=0.00984
    B1=0.03810
    B2=0.00869
    B3=0.00003
    B4=0.0
    B5=0.0
    B6=0.0
    C1=-0.57344
    D2=0.08074
    C3=0.0
    D4=0.0
    D5=0.0
    D6=0.0
    SC=0.05666
    U(N)=0.112
    GO TO 494
404 B0=0.00509
    B1=0.02644
    B2=0.00838
    B3=0.00010
    B4=0.0
    B5=0.0
```

```

      B6=0.0
      D1=-0.59602
      D2=0.08757
      D3=-0.00002
      D4=0.0
      C5=0.0
      C6=0.0
      SC=0.04001
      U(N)=0.081
      GO TO 494
405  E0=0.00037
      B1=0.00823
      B2=0.00983
      B3=0.00125
      B4=0.00001
      B5=0.0
      B6=0.0
      D1=-1.03045
      C2=0.20108
      D3=-0.00726
      C4=0.0
      C5=0.0
      D6=0.0
      SC=0.01969
      U(N)=0.121
      GO TO 494
406  B0=0.00002
      B1=0.00147
      B2=0.00414
      B3=0.00155
      B4=0.00008
      B5=0.0
      B6=0.0
      D1=-1.18099
      C2=0.34494
      D3=-0.03182
      D4=0.00080
      D5=0.0
      C6=0.0
      SC=0.007262
      U(N)=0.055
      GO TO 494
494  SQLOD(N)=0.0
      DO 501 ITIM=7,30
      QLOD(ITIM,N)=CLCAC(X)
      SQLOD(N)=SQLOD(N)+QLOD(ITIM,N)
501  CONTINUE
      JCHEK=JCHEK+1

```

```

      AQLOD(N)=SQLOD(N)/24.0
C     IF THE RESPNCE FACTOR CALCULATIONS NEED TO INTERATE MORE THAN 4
C     TIMES IT HAS ALREADY CCNVERGED AND THE TWC VALUES WILL NEVER GET
C     TO WITH IN 0.02 OF EACH OTHER. IT SHOULD BE NOTED THAT THEY WILL
C     BE CLOSE OR WITHIN 0.4 OF EACH OTHER.
      IF(JCHEK.GT.4) GO TO 10
C     THE FOLLOWING TWO STATEMENTS CALCULATE VALUES FOR A CHECK
C     STATEMENT TO SEE IF THE RESPNCE FACTOR CALCULATIONS HAVE
C     CCNVERGED.
      Q1=U(N)*(ARIM(N)-ROOMT)
      Q2=AQLOD(N)
      WRITE(6,101) Q1,Q2,JCHEK,N
      IF(ABS(Q1-Q2).GT.0.02) GO TO 503
10    WRITE(6,100) JCHEK
      GO TO 493
503   CONTINUE
      DO 502 ITIM=1,6
C     THE FOLLOWING STATEMENT MAKES THE QLOD MATRIX OR THE WALL HEAT
C     GAINS CONTINUOUS FROM ONE CYCLE TO ANOTHER CYCLE.
      QLOD(ITIM,N)=QLOD(ITIM+24,N)
502   CONTINUE
      GO TO 494
493   CCNTINUE
C     THE FOLLOWING LOOP (53) SUMS UP THE WINDOW AND DOOR AREAS ON EACH
C     WALL. THIS TOTAL AREA IS B . THIS B IS SUBTRACTED FROM THE TOTAL
C     WALL AREA BEFORE BEING MULTIPLIED BY THE HEAT GAINS PER SQUARE
C     FOOT, QLOD(ITIM,N).
      DO 53 ITIM=7,30
      DO 53 N=2,N1
      B=0.0
      KK=NOWIN(N)
      KK5=NODOOR(N)
      IF(KK.EQ.0) GO TO 44
      DO 42 III=1,KK
42    B=B+WINDOW(N,III,8)
44    IF(KK5.EQ.0) GC TO 53
      DO 43 MM=1,KK5
      III=MM+KK
43    B=B+WINDOW(N,III,8)
53    QLOT(ITIM,N)=QLOD(ITIM,N)*(DIME2(6,N)-B)
      RETURN
      END

```



```

C      FUNCTION ROLOD(XX)
C      THIS FUNCTION CALCULATES THE HEAT GAIN THROUGH THE CEILING USING
C      THE RESPONSE FACTOR EQUATION AND A CONSTANT INSIDE TEMPERATURE.
C      BBO,BB1,BB2, AND ETC. ARE THE RESPONSE FACTOR FOR THE ATTIC
C      TEMPERATURES. DD1,DD2,DD3, AND ETC. ARE THE RESPONSE FACTOR FOR
C      THE HEAT GAINS THROUGH THE CEILING. THE VALUES ARE ASSIGNED TO
C      THESE VARIABLES IN THE LOAD SUBROUTINE FOR EACH UNIQUE WALL TYPE.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/WALPRO/B0,B1,B2,B3,B4,B5,B6,BBO,BB1,BB2,BB3,BB4,BB5,BB6
COMMON/WALPR1/D1,D2,D3,D4,D5,D6,DD1,DD2,DD3,DD4,DD5,DD6,SC,SSC
COMMON/AIRC6/WTM(30,15),SRTM(15),ARTM(15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLCD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIR3/QLOT(30,15),RQLOT(30)
COMMON/AC21/TA(30),NCCE,NOWA(15),CEAR,VOLAIR,TR(30),WT(30)
X=BBO*TA(ITIM)+BB1*TA(ITIM-1)+BB2*TA(ITIM-2)+BB3*TA(ITIM-3)+BB4*TA
1(ITIM-4)+BB5*TA(ITIM-5)+BB6*TA(ITIM-6)
Y=DD1*RQLOD(ITIM-1)+DD2*RQLOD(ITIM-2)+DD3*RQLOD(ITIM-3)+DD4*RQLOD(
1ITIM-4)+DD5*RQLOD(ITIM-5)+DD6*RQLOD(ITIM-6)
Z=ROOMT*SSC
ROLOD=X-Y-Z
RETURN
END

```

```

FUNCTION QLCAD(XX)
C THIS FUNCTION IS VERY SIMILAR TO ROLOD(X), THE ONLY DIFFERENCE IS
C THAT IT CALCULATES WALL GAINS INSTEAD OF CEILING HEAT GAINS.
C THIS SUBROUTINE CALCULATES WINDOW AND DOOR HEAT GAINS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,ANN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),RLITE(24,15)
COMMON/AIRC1/NOWIN(15),NODOOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),CLITE(24,15,5),III
COMMON/WALPRO/B0,B1,B2,B3,B4,B5,B6,BR0,BR1,BR2,BR3,BR4,BR5,BR6
COMMON/WALPRI/D1,D2,D3,D4,D5,D6,DD1,DD2,DD3,DD4,DD5,DD6,SC,SSC
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC6/WTM(30,15),SRTM(15),ARTM(15)
COMMON/AIRC7/QLGD(30,15),RQLGD(30),AQLGD(15)
COMMON/AIRC8/SQLGD(15),UCE,T,RCONT,UW(15,10)
COMMON/AIRC9/UD(15,10),QDOOR(24,15,10),QWIND(24,15,10)
COMMON/AC1/SACCCF(15,10)
COMMON/AIR3/QLCT(30,15),RQLCT(30)
X=B0*RTM(ITIM,N)+B1*RTM(ITIM-1,N)+B2*RTM(ITIM-2,N)+B3*RTM(ITIM-3,N
1)+B4*RTM(ITIM-4,N)+B5*RTM(ITIM-5,N)+B6*RTM(ITIM-6,N)
Y=D1*QLGD(ITIM-1,N)+D2*QLGD(ITIM-2,N)+D3*QLGD(ITIM-3,N)+D4*QLGD(IT
1IM-4,N)+D5*QLGD(ITIM-5,N)+D6*QLGD(ITIM-6,N)
Z=RCONT*SC
QLOAD=X-Y-Z
RETURN
END

```

```

SUBROUTINE CWLOAD
C SUBROUTINE CWLOAD CALCULATES HEAT GAINS FOR DOORS AND WINDOWS. ALL
C OF THESE TYPES OF CALCULATIONS ARE INSTANTANEOUS BECAUSE OF THE
C LIGHT CONSTRUCTION MATERIALS USED.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),NCDCOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC6/WTM(30,15),SRTM(15),ARTM(15)
COMMON/AIRC7/QLQD(30,15),RQLQD(30),AQLQD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),CDOOR(24,15,10),QWIND(24,15,10)
COMMON/ACL/SACCOF(15,10)
COMMON/AIRC3/CLCT(30,15),RQLOT(30)
C LOPP(25) MAKES SURE THE WINDOW AND DOOR LOAD IS CALCULATED ON ALL
C THE WALLS AND IT GIVES THE MACHINE ALL THE NEEDED INFORMATION
C TO CALCULATE HEAT GAINS TO THE APPROPRIATE MATRICES.
DO 25 N=2,N1
KK=NOWIN(N)
KK5=NODOOR(N)
IF(KK.EQ.0) GO TO 24
DO 22 III=1,KK
LKK=WINDOW(N,III,7)
10 FORMAT(' THERE IS NO VALUES FOR WINDOW ',I3,' ON WALL',I3)
C WINDOW TYPES(LKK=1,3,5,11-16,31-33) ARE DOUBLE PANE INSULATED
C GLASS WITH WHITE OPAQUE ROLLER SHADES.
C WINDOW TYPES(LKK=2,4,6,21-26,41-43) ARE SINGLE PANE
C UNWEATHERSTRIPPED , NO INTERNAL SHADING.
IF(LKK.EQ.1) GO TO 101
IF(LKK.EQ.2) GO TO 102
IF(LKK.EQ.3) GO TO 101
IF(LKK.EQ.4) GO TO 102
IF(LKK.EQ.5) GO TO 101
IF(LKK.EQ.6) GO TO 102
IF(LKK.GE.11.AND.LKK.LE.16) GO TO 101
IF(LKK.GE.21.AND.LKK.LE.26) GO TO 102
IF(LKK.GE.31.AND.LKK.LE.33) GO TO 101
IF(LKK.GE.41.AND.LKK.LE.43) GO TO 102
C IF LKK IS NOT SATISFIED BY ANY OF THE ABOVE IF STATEMENTS THEN THE
C COMPUTER PRINTS OUT WHICH WINDOW ON WHICH WALL IS NOT A LISTED
C WINDOW TYPE. THE PROGRAM THEN RETURNS TO THE MAIN PROGRAM AND
C TERMINATES.
WRITE(6,10) III,N
KODE=1
RETURN
C THIS IS A DOUBLE PANE INSULATED WINDOW WITH WHITE ROLLER SHADES.
101 UW(N,III)=0.56

```

```

      SADCOF(N,III)=0.25
      GO TO 22
C     THIS IS A SINGLE PANE WINDOW WITH NO INTERNAL SHADING.
102  UW(N,III)=1.06
      SADCOF(N,III)=1.0
      GO TO 22
22  CCNTINUE
24  IF(KK5.EQ.0) GO TO 25
C     THIS LOOP (23) LOADS THE U-VALUES FOR DOORS IN THE APPROPRIATE
C     MATRIX NOTE THAT ONLY ONE U-VALUE IS USED FOR ALL DOORS BECAUSE OF
C     THERE SIMILAR CCNSTRUCTION. IF OTHER DOOR TYPES ARE REQUESTED
C     MORE CAN BE ACDED.
      DO 23 MM=1,KK5
      III=MM+KK
      LKK=WINDOW(N,III,7)
      IF(LKK.GE.51.AND.LKK.LE.55) GO TO 500
C     THIS IS A 1 AND 1/2 INCH WOOD DOOR AN OUTSIDE FILM RESISTANCE OF
C     0.25 AND AN INSIDE FILM OF 0.61.
500  UD(N,MM)=0.355
      GO TO 23
23  CCNTINUE
25  CONTINUE
C     DOUBLE LOOP (40) CALCULATES HEAT GAINS FOR THE WINDOWS AND DOORS
C     FROM THE INFORMATION GIVEN ABOVE.
      DO 40 N=2,N1
      DO 40 ITIM=1,24
      KK=NOWIN(N)
      KK5=NODOOR(N)
      IF(KK.EQ.0) GO TO 44
      DO 42 III=1,KK
C     QINST IS THE THERMAL LOAD THROUGH THE WINDOWS.
      QINST=WINDOW(N,III,8)*UW(N,III)*(TEMP(ITIM)-ROOMT)
C     QRAD IS THE RADIANT LOAD THROUGH THE WINDCWS.
      QRAD=WINDOW(N,III,8)*SADCOF(N,III)*SOLAVW(ITIM,N,III)
C     QWIND IS THE TOTAL HEAT GAIN THROUGH THE WINDOWS.
42  QWIND(ITIM,N,III)=QINST+QRAD
44  IF(KK5.EQ.0) GO TO 40
      DO 43 MM=1,KK5
      III=MM+KK
C     QDCOR IS THE HEAT GAIN THROUGH THE DOOR USING SOL-AIR TEMPERATURES
C     ON THE DOOR.
43  QDCOR(ITIM,N,MM)=WINDOW(N,III,8)*UD(N,MM)*(DTM(ITIM,N,MM)-ROOMT)
40  CONTINUE
      RETURN
      END

```

```

SUBROUTINE VALLO
C   THIS SUBROUTINES FUNTION IS TO PRINT OUT VALUES OF INTERMEDIATE
C   CALCULATIONS, SUCH AS SOL-AIR TEMPERATURES AND HEAT GAINS FROM
C   WALLS, WINDOWS, AND DOORS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15),NUMC(15)
COMMON/AIRC1/NOWIN(15),ACDOOR(15),WINDCH(15,10,8),NUMB(15),KODE
COMMON/AIRC2/WLITE(24,15,10),DLITE(24,15,5),III
COMMON/WALPRO/B0,B1,B2,B3,B4,B5,B6,BB0,BB1,BB2,BB3,BB4,BB5,BB6
COMMON/WALPRI/D1,D2,D3,D4,D5,D6,DD1,DD2,DD3,DD4,DD5,DD6,SC,SSC
COMMON/AIRC5/TEMP(24),SCLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC6/WTM(30,15),SRTM(15),ARTM(15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLOD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QDOOR(24,15,10),QWIND(24,15,10)
COMMON/AC1/SACCOF(15,10)
COMMON/AIRC3/CLOT(30,15),RQLOT(30)
401 FORMAT(' THESE ARE THE HEAT FLOW RATES IN BTU/HR*FT*FT* FOR EACH W
      1 ALL AT EVERY HOUR OF THE DAY')
402 FORMAT(3X,'TIME',1X,15I7)
403 FORMAT(18,15F7.3)
404 FORMAT(' THESE ARE THE HEAT FLOW RATES IN BTU/HR*FT*FT FOR EACH R
      1 OOF')
405 FORMAT(' THESE ARE THE TOTAL HEAT FLOWS IN BTU/HR FOR EACH WALL E
      1 VERY HOUR')
406 FORMAT(' THESE ARE THE TOTAL HEAT FLOWS IN BTU/HR FOR EACH ROOF E
      1 VERY HOUR')
407 FORMAT(18,15F7.1)
408 FORMAT(' TOTAL HEAT FLOW FOR EACH WINDOW ON WALL ',I3)
409 FORMAT(' TOTAL HEAT FLCW FOR EACH DOOR ON WALL ',I3)
410 FORMAT(' THESE ARE THE SOLAIR TEMPERATURES FOR THE WALLS')
411 FORMAT(' THESE ARE THE SOLAIR TEMPERTURES FOR THE ROOFS')
412 FORMAT(' THESE ARE THE AVERAGE SOLAR HEAT GAINS FOR THE WINDOWS A
      1 ND THE SOL-AIR TEMPERTURES FOR THE DOORS')
413 FORMAT(' AVERAGE SOLAR HEAT GAINS FOR THE WINDOWS ON WALL ',I3)
414 FORMAT(' SOL-AIR TEMPERATURES FOR THE DOORS ON WALL ',I3)
      WRITE(6,410)
      WRITE(6,402)(NUMB(I),I=2,N1)
      DO 30 ITIM=1,30
      ITIN=ITIM-6
C   THE FOLLOWING ARE THE SOL-AIR TEMPERATURES OF THE WALLS.
      30 WRITE(6,407) ITIN,(RTM(ITIM,I),I=2,N1)
      WRITE(6,411)
      WRITE(6,402)(NUMB(I),I=2,N1)
      DO 31 ITIM=1,30
      ITIN=ITIM-6
C   THE FOLLOWING ARE THE SOL-AIR TEMPERATURES ON THE ROOFS.
      31 WRITE(6,407) ITIN,(WTM(ITIM,I),I=2,N1)

```

```

WRITE(6,412)
DO 32 J=2,N1
K1=NOWIN(J)
K11=K1+1
K2=NODCCR(J)
K3=K2+K1
IF(K3.EQ.0) GO TO 32
L=J-1
K22=K2+1
IF(K1.EQ.0) GO TO 34
WRITE(6,413) L
WRITE(6,402)(NUMC(I),I=2,K11)
DO 35 ITIM=1,24
C THE FOLLOWING ARE THE SOLAR HEAT GAINS ON EACH WINDOW.
35 WRITE(6,407) ITIM,(SOLAVW(ITIM,J,I5),I5=1,K1)
34 IF(K2.EQ.0) GO TO 32
WRITE(6,414) L
WRITE(6,402)(NUMC(I),I=2,K22)
DO 36 ITIM=1,24
C THE FOLLOWING ARE THE SOL-AIR TEMPERATURES ON THE DOORS.
36 WRITE(6,407) ITIM,(DTM(ITIM,J,I5),I5=1,K2)
32 CONTINUE
WRITE(6,401)
WRITE(6,402)(NUMB(I),I=2,N1)
DO 50 ITIM=7,30
ITIN=ITIM-6
C THESE ARE THE HEAT GAIN RATES FOR EACH WALL.
50 WRITE(6,403) ITIN,(QLOD(ITIM,I),I=2,N1)
WRITE(6,405)
WRITE(6,402)(NUMB(I),I=2,N1)
DO 54 ITIM=7,30
ITIN=ITIM-6
C THESE ARE THE WRITE STATEMENTS FOR THE TOTAL HEAT GAINS ON EACH
C WALL.
54 WRITE(6,407) ITIN,(QLCT(ITIM,I),I=2,N1)
DO 61 J=2,N1
K1=NOWIN(J)
K11=K1+1
K2=NODCCR(J)
K3=K2+K1
IF(K3.EQ.0) GO TO 61
L=J-1
K22=K2+1
IF(K1.EQ.0) GO TO 62
WRITE(6,408) L
WRITE(6,402)(NUMC(I),I=2,K11)
DO 63 ITIM=1,24
C THESE ARE THE WRITE STATEMENTS FOR THE TOTAL HEAT GAIN ON EACH

```

```
C    WINDOW.  
63  WRITE(6,407) ITIM,{QWIND(ITIM,J,I5),I5=1,K1}  
62  IF(K2.EQ.0) GO TO 61  
    WRITE(6,409) L  
    WRITE(6,402){NUMC(I),I=2,K22}  
    DO 64 ITIM=1,24  
C    THESE ARE WRITE STATEMENTS FOR THE TOTAL HEAT GAIN ON EACH DOOR.  
64  WRITE(6,407) ITIM,{QDOOR(ITIM,J,I5),I5=1,K2}  
61  CCNTINUE  
    RETURN  
    END
```

```

SUBROUTINE AINFIL
C   THIS SUBROUTINE CALCULATES THE SENSIBLE AND LATENT INFILTRATION
C   ON ALL WINDOWS AND DOORS LKK=WINDOW(N,III,7) IS THE TYPE OF
C   WINDOW THAT THE III' TH WINDOW IS ON WALL N.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),NCDCOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC7/QLCD(30,15),RQLCD(30),AQLCD(15)
COMMON/AIRC8/SQLCD(15),UCE,T,RODMT,LW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QDOCR(24,15,10),QWIND(24,15,10)
COMMON/AIRC3/QLCT(30,15),RQLCT(30)
COMMON/AIRC14/SEC,CRFL(15,10),CRALEN(15,10),ALFIN(15,10,24)
COMMON/AIRC20/SINFIL(15,10,24),ALFIL(15,10,24)
800 FORMAT(18,' IS THE VALUE FOR THE ',15,' WINDOW ON WALL',15)
801 FORMAT(18,' IS THE VALUE FOR THE ',15,' DOOR ON WALL',15)
DO 10 N=2,N1
KK=NOWIN(N)
KK5=NCDCOR(N)
IF(KK.EQ.0) GO TO 14
DO 12 III=1,KK
C   THIS IS THE WINDOW TYPE NUMBER TO FIND OUT WHAT TYPE THEY ARE. THE
C   TYPE ARE EXPLAINED UNDER STATEMENT 20 OR THE WINTER CHECK
C   STATEMENTS.
LKK=WINDOW(N,III,7)
C   THIS STATEMENT CHECKS FOR WINTER TIME.
IF(M.LE.2.OR.M.GE.12) GO TO 20
C   THIS STATEMENT CHECKS FOR SPRING TIME
IF(M.GE.3.AND.M.LE.5) GO TO 21
C   THIS STATEMENT CHECKS FOR FALL TIME
IF(M.GE.9.AND.M.LE.11) GO TO 21
C   THIS STATEMENT CHECKS FOR SUMMER TIME.
IF(M.GE.6.AND.M.LE.8) GO TO 22
C   WINTER TIME SELECTION STATEMENTS
C   WINTER TIME SELECTION STATEMENTS
20 WI=0.0046
WO=0.0046
C   WINDOW 1 IS AN AVERAGE FIT SINGLE DOUBLE HUNG WINDOW
C   WEATHERSTRIPPED.
IF(LKK.EQ.1) GO TO 51
C   WINDOW 2 IS AN AVERAGE FIT SINGLE DOUBLE HUNG WINDOW
C   UNWEATHERSTRIPPED.
IF(LKK.EQ.2) GO TO 52
C   WINDOW 3 TWO AVERAGE FIT DOUBLE HUNG WINDOWS, SIDE BY SIDE
C   WEATHERSTRIPPED.
IF(LKK.EQ.3) GO TO 53
C   WINDOW 4 TWO AVERAGE FIT DOUBLE HUNG WINDOWS, SIDE BY SIDE
C   UNWEATHERSTRIPPED.

```



```

IF(LKK.EQ.4) GO TO 54
C WINDOW 5 THREE AVERAGE FIT DOUBLE HUNG WINDOWS SIDE BY SIDE
C WEATHERSTRIPPED.
IF(LKK.EQ.5) GO TO 55
C WINDOW 6 THREE AVERAGE FIT DOUBLE HUNG WINDOWS SIDE BY SIDE
C UNWEATHERSTRIPPED.
IF(LKK.EQ.6) GO TO 56
C WINDOW 11 ONE AWNING TYPE WINDOW WEATHERSTRIPPED.
IF(LKK.EQ.11) GO TO 61
C WINDOW 12 TWO AWNING TYPE WINDOWS SIDE BY SIDE WEATHERSTRIPPED .
IF(LKK.EQ.12) GO TO 62
C WINDOW 13 FOUR AWNING TYPE WINDOWS TWO ON TOP OF TWO
C WEATHERSTRIPPED.
IF(LKK.EQ.13) GO TO 63
C WINDOW 14 TWO AWNING TYPE WINDOWSS ONE ON TOP OF THE OTHER ,
C WEATHERSTRIPPED.
IF(LKK.EQ.14) GO TO 64
C WINDOW 15 THREE AWNING TYPE WINDOWS SIDE BY SIDE
C WEATHERSTRIPPED.
IF(LKK.EQ.15) GO TO 65
C WINDOW 21 IS THE SAME AS 11 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.21) GO TO 71
C WINDOW 16 IS THREE AWNING TYPE WINDOWS ONE ON TOP OF THE OTHER
C WEATHERSTRIPPED.
IF(LKK.EQ.16) GO TO 66
C WINDOW 22 IS THE SAME AS 12 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.22) GO TO 72
C WINDOW 23 IS THE SAME AS 13 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.23) GO TO 73
C WINDOW 24 IS THE SAME AS 14 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.24) GO TO 74
C WINDOW 25 IS THE SAME AS 15 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.25) GO TO 75
C WINDOW 26 IS THE SAME AS 16 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.26) GO TO 76
C WINDOW 31 A SINGLE ( DOUBLE PANE) SLIDING WINDOW WEATHERSTRIPPED.
IF(LKK.EQ.31) GO TO 81
C WINDOW 32 A DOUBLE ( TRIPLE PANE) SLIDING WINDOW WEATHERSTRIPPED.
IF(LKK.EQ.32) GO TO 82
C WINDOW 33 A SLIDING GLASS DOOR WEATHERSTRIPPED.
IF(LKK.EQ.33) GO TO 93
C WINDOW 41 IS THE SAME AS 31 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.41) GO TO 91
C WINDOW 42 IS THE SAME AS 32 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.42) GO TO 92
C WINDOW 43 IS THE SAME AS 33 ONLY UNWEATHERSTRIPPED.
IF(LKK.EQ.43) GO TO 93
WRITE(6,800) LKK,III,N

```

```

      KODE=1
      RETURN
C     SPRING TIME SELECTION STATEMENTS
C     SPRING TIME SELECTION STATEMENTS
C     FALL TIME SELECTION STATEMENTS
C     FALL TIME SELECTION STATEMENTS
21  WI=0.0077
    WO=0.0077
      IF(LKK.EQ.1) GO TO 151
      IF(LKK.EQ.2) GO TO 152
      IF(LKK.EQ.3) GO TO 153
      IF(LKK.EQ.4) GO TO 154
      IF(LKK.EQ.5) GO TO 155
      IF(LKK.EQ.6) GO TO 156
      IF(LKK.EQ.11) GO TO 161
      IF(LKK.EQ.12) GO TO 162
      IF(LKK.EQ.13) GO TO 163
      IF(LKK.EQ.14) GO TO 164
      IF(LKK.EQ.15) GO TO 165
      IF(LKK.EQ.16) GO TO 166
      IF(LKK.EQ.21) GO TO 171
      IF(LKK.EQ.22) GO TO 172
      IF(LKK.EQ.23) GO TO 173
      IF(LKK.EQ.24) GO TO 174
      IF(LKK.EQ.25) GO TO 175
      IF(LKK.EQ.26) GO TO 176
      IF(LKK.EQ.31) GO TO 181
      IF(LKK.EQ.32) GO TO 182
      IF(LKK.EQ.33) GO TO 183
      IF(LKK.EQ.41) GO TO 191
      IF(LKK.EQ.42) GO TO 192
      IF(LKK.EQ.43) GO TO 193
      WRITE(6,800) LKK,III,N
      KODE=1
      RETURN
C     SUMMER TIME SELECTION STATEMENTS
C     SUMMER TIME SELECTION STATEMENTS
22  WI=0.0108
    WO=0.0158
      IF(LKK.EQ.1) GO TO 251
      IF(LKK.EQ.2) GO TO 252
      IF(LKK.EQ.3) GO TO 253
      IF(LKK.EQ.4) GO TO 254
      IF(LKK.EQ.5) GO TO 255
      IF(LKK.EQ.6) GO TO 256
      IF(LKK.EQ.11) GO TO 261
      IF(LKK.EQ.12) GO TO 262
      IF(LKK.EQ.13) GO TO 263

```

```

IF(LKK.EQ.14) GO TO 264
IF(LKK.EQ.15) GO TO 265
IF(LKK.EQ.16) GO TO 266
IF(LKK.EQ.21) GO TO 271
IF(LKK.EQ.22) GO TO 272
IF(LKK.EQ.23) GO TO 273
IF(LKK.EQ.24) GO TO 274
IF(LKK.EQ.25) GO TO 275
IF(LKK.EQ.26) GO TO 276
IF(LKK.EQ.31) GO TO 281
IF(LKK.EQ.32) GO TO 282
IF(LKK.EQ.33) GO TO 283
IF(LKK.EQ.41) GO TO 291
IF(LKK.EQ.42) GO TO 292
IF(LKK.EQ.43) GO TO 293
WRITE(6,800) LKK,III,N
KODE=1
RETURN
C   THE FOLLOWING ARE EQUATIONS TO CALCULATE CRACK LENGTHS AND VALUES
C   FOR INFILTRATION FLOW RATES PER FOOT OF CRACK LENGTH FOR WINDOWS
C   ONLY.
71  CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDOW(N,III,5))
    CRFL(N,III)=39.0
    GO TO 12
171 CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDOW(N,III,5))
    CRFL(N,III)=21.0
    GO TO 12
271 CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDOW(N,III,5))
    CRFL(N,III)=7.0
    GO TO 12
72  CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=39.0
    GO TO 12
172 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=21.0
    GO TO 12
272 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=7.0
    GO TO 12
73  CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=39.0
    GO TO 12
173 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=21.0
    GO TO 12
273 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=7.0
    GO TO 12

```

```

74 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
   CRFL(N,III)=39.0
   GO TO 12
174 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
274 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
   CRFL(N,III)=7.0
   GO TO 12
75  CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=39.0
   GO TO 12
175 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
275 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=7.0
   GO TO 12
76  CRALEN(N,III)=2.0*WINDOW(N,III,2)+4.0*WINDOW(N,III,5)
   CRFL(N,III)=39.0
   GO TO 12
176 CRALEN(N,III)=2.0*WINDOW(N,III,2)+4.0*WINDOW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
276 CRALEN(N,III)=2.0*WINDOW(N,III,2)+4.0*WINDOW(N,III,5)
   CRFL(N,III)=7.0
   GO TO 12
91  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=39.0
   GO TO 12
191 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
291 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=7.0
   GO TO 12
92  CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=39.0
   GO TO 12
192 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
292 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=7.0
   GO TO 12
93  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=39.0
   GO TO 12

```

```

193 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
    CRFL(N,III)=21.0
    GO TO 12
293 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
    CRFL(N,III)=7.0
    GO TO 12
51  CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
52  CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=39.0
    GO TO 12
53  CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
54  CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
    CRFL(N,III)=39.0
    GO TO 12
55  CRALEN(N,III)=4.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
56  CRALEN(N,III)=4.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
    CRFL(N,III)=39.0
    GO TO 12
61  CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDCW(N,III,5))
    CRFL(N,III)=24.0
    GO TO 12
62  CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
63  CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
64  CRALEN(N,III)=2.0*WINDCW(N,III,2)+3.0*WINDCW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
65  CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
66  CRALEN(N,III)=2.0*WINDOW(N,III,2)+4.0*WINDCW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
81  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12
82  CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
    CRFL(N,III)=24.0
    GO TO 12

```

```

83 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=34.0
   GO TO 12
151 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
152 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
153 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
154 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
155 CRALEN(N,III)=4.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
156 CRALEN(N,III)=4.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
   CRFL(N,III)=21.0
   GO TO 12
161 CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDCW(N,III,5))
   CRFL(N,III)=13.0
   GO TO 12
162 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
163 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
164 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
165 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
166 CRALEN(N,III)=2.0*WINDOW(N,III,2)+4.0*WINDOW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
181 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
182 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=13.0
   GO TO 12
183 CRALEN(N,III)=3.0*WINDCW(N,III,2)+2.0*WINDOW(N,III,5)
   CRFL(N,III)=19.0
   GO TO 12

```

```

251 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
252 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=7.0
    GO TO 12
253 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
254 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
    CRFL(N,III)=7.0
    GO TO 12
255 CRALEN(N,III)=4.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
256 CRALEN(N,III)=4.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=7.0
    GO TO 12
261 CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDOW(N,III,5))
    CRFL(N,III)=4.0
    GO TO 12
262 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
263 CRALEN(N,III)=3.0*WINDOW(N,III,2)+3.0*WINDOW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
264 CRALEN(N,III)=2.0*WINDOW(N,III,2)+3.0*WINDCW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
265 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
266 CRALEN(N,III)=2.0*WINDOW(N,III,2)+4.0*WINDCW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
281 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
282 CRALEN(N,III)=4.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
    CRFL(N,III)=4.0
    GO TO 12
283 CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDCW(N,III,5)
    CRFL(N,III)=6.0
    GO TO 12
12 CONTINUE
14 IF(KK5.EQ.0) GO TO 10
C   THE SAME-THING THAT WAS DONE FOR WINDOWS ABOVE IS NOW REDONE FOR

```

```

C      DOORS.  THE DOOR TYPES ARE LISTED UNDER THE WINTER TYPE STATEMENT
DO 13 MM=1, KK5
      III=MM+KK
C      LKD = WINDOW (N, III, 7) IS THE DOOR TYPE NUMBER.
      LKD=WINDOW(N, III, 7)
C      FIRST THE TIME OF YEAR IS FOUND IN THE NEXT FOUR IF STATEMENTS.
      IF(M.LE.2.OR.M.GE.12) GO TO 520
      IF(M.GE.3.AND.M.LE.5) GO TO 521
      IF(M.GE.9.AND.M.LE.11) GO TO 521
      IF(M.GE.6.AND.M.LE.8) GO TO 522
520  WI=0.0046
      WO=0.0046
C      SINGLE WEATHERSTRIPPED DOOR.
      IF(LKD.EQ.51) GO TO 551
C      DOUBLE WEATHERSTRIPPED DOOR
      IF(LKD.EQ.52) GO TO 552
C      SINGLE UNWEATHERSTRIPPED DOORS.
      IF(LKD.EQ.53) GO TO 553
C      DOUBLE UNWEATHERSTRIPPED DOORS.
      IF(LKD.EQ.54) GO TO 554
C      GARAGE DOORS.
      IF(LKD.EQ.55) GO TO 555
      WRITE(6,801) LKD, III, N
      KODE=1
      RETURN
521  WI=0.0077
      WO=0.0077
      IF(LKD.EQ.51) GO TO 651
      IF(LKD.EQ.52) GO TO 652
      IF(LKD.EQ.53) GO TO 653
      IF(LKD.EQ.54) GO TO 654
      IF(LKD.EQ.55) GO TO 655
      WRITE(6,801) LKD, III, N
      KODE=1
      RETURN
522  WI=0.0108
      WO=0.0158
      IF(LKD.EQ.51) GO TO 751
      IF(LKD.EQ.52) GO TO 752
      IF(LKD.EQ.53) GO TO 753
      IF(LKD.EQ.54) GO TO 754
      IF(LKD.EQ.55) GO TO 755
      WRITE(6,801) LKD, III, N
      KODE=1
      RETURN
C      THE FOLLOWING EQUATIONS CALCULATE CRACK LENGTHS AND PROVIDE FLOW
C      RATES PER FOOT OF CRACK FOR DOORS.
551  CRALEN(N, III)=2.0*WINDOW(N, III, 2)+2.0*WINDOW(N, III, 5)

```



```

        CRFL(N,III)=34.0
        GO TO 13
552  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=34.0
        GO TO 13
553  CRALEN(N,III)=2.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=34.0
        GO TO 13
554  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=34.0
        GO TO 13
555  CRALEN(N,III)=2.0*WINDOW(N,III,2)+(SEC+1.0)*WINDOW(N,III,5)
        CRFL(N,III)=68.0
        GO TO 13
651  CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDOW(N,III,5))
        CRFL(N,III)=19.0
        GO TO 13
652  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=19.0
        GO TO 13
653  CRALEN(N,III)=2.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=19.0
        GO TO 13
654  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=19.0
        GO TO 13
655  CRALEN(N,III)=2.0*WINDOW(N,III,2)+(SEC+1.0)*WINDOW(N,III,5)
        CRFL(N,III)=38.0
        GO TO 13
751  CRALEN(N,III)=2.0*(WINDOW(N,III,2)+WINDOW(N,III,5))
        CRFL(N,III)=6.0
        GO TO 13
752  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=6.0
        GO TO 13
753  CRALEN(N,III)=2.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=6.0
        GO TO 13
754  CRALEN(N,III)=3.0*WINDOW(N,III,2)+2.0*WINDOW(N,III,5)
        CRFL(N,III)=6.0
        GO TO 13
755  CRALEN(N,III)=2.0*WINDOW(N,III,2)+(SEC+1.0)*WINDOW(N,III,5)
        CRFL(N,III)=12.0
        GO TO 13
13  CONTINUE
10  CONTINUE
DO 15 N=2,N1
CO 15 IT=1,24

```

```

      KK=NOWIN(N)
      KK5=NCDOCR(N)
      IF(KK.EQ.0) GO TO 17
      DO 16 III=1, KK
C      THIS EQUATION CALCULATES LATENT INFILTRATION(ALFIL(N,III,ITIM))
C      FOR WINDOWS.
      ALFIL(N,III,ITIM)=CRALEN(N,III)*CRFL(N,III)*79.5*(WO-WI)
C      THIS EQUATION CALCULATES SENSIBLE INFILTRATION (SINFIL(N,III,ITIM)
C      ) FOR WINDOWS.
16  SINFIL(N,III,ITIM)=CRALEN(N,III)*CRFL(N,III)*0.018*(TEMP(ITIM)-ROO
      1MT)
17  IF(KK5.EQ.0) GO TO 15
      DO 18 MM=1, KK5
      III=MM+KK
C      THIS EQUATION CALCULATES LATENT INFILTRATION FOR DOORS.
      ALFIL(N,III,ITIM)=CRALEN(N,III)*CRFL(N,III)*79.5*(WC-WI)
C      THIS EQUATION CALCULATES SENSIBLE INFILTRATION FOR DOORS.
18  SINFIL(N,III,ITIM)=CRALEN(N,III)*CRFL(N,III)*0.018*(TEMP(ITIM)-ROO
      1MT)
15  CONTINUE
      RETURN
      END

```

```

SUBROUTINE AINTER
C   THIS SUBROUTINE ESTIMATES THE INTERIOR LOADS DUE TO PEOPLE INSIDE
C   THE HOUSE AND THE LOAD DUE TO THE LIGHTS IN THE HOUSE. BOTH OF
C   THESE LOADS ARE ESTIMATED BY CRUDE CALCULATIONS BECAUSE OF THE
C   UNPREDICTABILITY OF THESE LOADS. THE TWO VALUES INPUTED ARE MEMBER
C   , THE TOTAL NUMBER OF FAMILY MEMBERS LIVING IN THE HOUSE AND
C   TOTLIG , WHICH IS THE TOTAL LIGHT CAPACITY IN THE HOUSE IN BTU/HR
C   CALCULATED FROM THE WATT RATING OF THE LIGHTS.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),ACOCER(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLOD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QDOOR(24,15,10),QWIND(24,15,10)
COMMON/AIRC3/QLOT(30,15),RQLCT(30)
COMMON/AIRC10/ALIGHT(24),PEOPLS(24),PEOPL(24),MEMBER,TOTLIG
COMMON/AIRC11/TOHous(24),TOLAT(24),TCWINS(24,15),TCWALR(24,15)
COMMON/AIRC12/TOWLAT(24,15),TOWALL(24,15),FACTOR,COLO(28)
COMMON/AIRC13/COOL(28),CT(24),TCHC(24)
C   THE FOLLOWING ARRAYS ARE INITIALIZED TO ZERO IN LOOP 10. THE
C   VALUES ARE ALIGHT(ITIM)- THE LIGHT LOAD PER HOUR AT TIME ITIM.
C   PEOPL- THE LATENT HEAT GAIN FROM PEOPLE IN THE HOUSE AT EACH
C   HOUR. PEOPLS- THE SENSIBLE HEAT GAIN FROM PEOPLE IN THE HOUSE AT
C   EACH HOUR.
DO 10 ITIM=1,24
  ALIGHT(ITIM)=0.0
  PEOPLS(ITIM)=0.0
10 PEOPL(ITIM)=0.0
C   THIS LOOP(11) LOADS PEOPLE LOADS WITH THE PEOPLE ASLEEP FROM 1 AM
C   TO 6 AM , FOR BOTH LATENT AND SENSIBLE HEAT GAINS.
DO 11 ITIM=1,6
  PEOPLS(ITIM)=MEMBER*200.
11 PEOPL(ITIM)=MEMBER*100.
C   AT 7 AM EVERYONE IS MORE ACTIVE ; THEREFORE THIS IS A SPECIAL
C   TIME. HEAT GAINS ARE CALCULATED FOR THIS SINGLE HOUR .
  PEOPLS(7)=250.*MEMBER
  PEOPL(7)=MEMBER*200.
C   FROM 8 AM TO 4 PM ALL THE CHILDREN ARE AT SCHOOL , THE FATHER IS
C   AT WORK , AND THE HOMEMAKER IS LEFT AT HOME ALONG AT A MODERATE
C   ACTIVITY RATE FOR BOTH SENSIBLE AND LATENT HEAT GAINS.
DO 12 ITIM=8,16
  PEOPLS(ITIM)=250.
12 PEOPL(ITIM)=200.
C   LOOP(13) CALCULATES THE PEOPLE LOAD WITH EVERYONE IN THE FAMILY AT
C   HOME AT A MODERATE ACTIVITY LEVEL.
DO 13 ITIM=17,22
  PEOPLS(ITIM)=MEMBER*250.0

```

```

13 PEOPLL(ITIM)=MEMBER*200.0
C LOOP(14) CALCULATES THE PEOPLE LOADS WITH EVERYONE AT HOME AND
C ASLEEP FOR THE LAST TWO HOURS OF THE DAY.
DO 14 ITIM=23,24
  PECPLS(ITIM)=MEMBER*200.
14 PEOPLL(ITIM)=MEMBER*100.
C THE NEXT STATEMENT STARTS THE CALCULATIONS FOR THE LIGHT LOADS.
DO 15 ITIM=15,21
C THE IF STATEMENTS BELOW CHECK WHICH MONTH THE CALCULATIONS ARE
C BEING MADE FOR.
  IF(M.EC.8) GO TO 5
  IF(M.EQ.7) GO TO 6
  IF(M.EQ.12) GO TO 3
  IF(M.EC.1) GO TO 4
3 A=391.
  B=0.142
  C=0.057
  DEG=-0.40927
  GO TO 100
4 A=390.
  B=0.142
  C=0.058
  DEG=-0.34907
  GO TO 100
5 A=351.0
  B=0.201
  C=0.122
  DEG=0.214675
  GO TO 100
6 A=344.0
  B=0.207
  C=0.136
  DEG=0.359537
  GO TO 100
100 AL=ITIM
C AH- IS THE NUMBER OF MINUTES AFTER 12 O'CLOCK NOON.
  AH=(((AL-12.0)*60.0)*0.25)
  AH=3.14159*AH/180.0
  BBB=(COS(ALAT)*COS(DEG)*COS(AH))+((SIN(ALAT)*SIN(DEG))
  BB=ARSIN(BBB)
  IF(BB.LE.0.0) GO TO 16
15 CONTINUE
C ITIM- WILL BE THE FIRST HOUR AFTER SUNSET , INN WILL BE THE LAST
C HOUR OF SUNSHINE DURING THE DAY.
16 INN=ITIM-1
  DO 17 ITIM=INN,22
17 ALIGHT(ITIM)=.33*TOTLIG
  RETURN

```

END

```

SUBROUTINE TOHEAT
C THIS SUBROUTINE SUMS UP THE HEAT GAINS AT EACH HOUR FOR THE WALLS,
C CEILING, WINDOWS, AND DOORS. THIS SUM IS PLACED IN THE ARRAY
C TOHO(ITIM) FOR EACH HOUR.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),RLITE(24,15)
COMMON/AIRC1/NCWIN(15),NCDOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLCD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,RCOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QDOR(24,15,10),QWIND(24,15,10)
COMMON/AIRC3/QLCT(30,15),RQLCT(30)
COMMON/AIRC10/ALIGHT(24),PEPLS(24),PEPLL(24),MEMBER,TOTLIC
COMMON/AIRC11/TOHDS(24),TOLAT(24),TCWINS(24,15),TOWALR(24,15)
COMMON/AIRC12/TOWLAT(24,15),TOWALL(24,15),FACTOR,COLO(28)
COMMON/AIRC13/COOL(28),CT(24),TCHC(24)
COMMON/AIRC20/SINFIL(15,10,24),ALFIL(15,10,24)
105 FORMAT(' THIS IS THE TCWALR MATRIX AT THE END OF SUBROUTINE TOHE
1AT')
106 FORMAT(3X,'TIME',1X,15I7)
107 FORMAT(18,15F7.1)
C LOOP(1) INITIALIZES ALL THE SUMMATION ARRAYS USED IN THE
C SUBROUTINE
DO 1 ITIM=1,24
C TOHDS(ITIM) IS USED TO SUM ALL THE HEAT GAINS FOR THE ENTIRE
C HOUSE FOR EACH HOUR.
TOHDS(ITIM)=0.0
C TOLAT(ITIM) IS THE TOTAL LATENT LOAD AT EACH HOUR (PEOPLE AND
C INFILTRATION).
TOLAT(ITIM)=0.0
DO 1 N=2,N1
C TOWALL(ITIM,N) IS THE TOTAL ON EACH WALL AT EVERY HOUR OF THE DAY
C FOR LATENT INFILTRATION, SENSIBLE INFILTRATION AND THE HEAT GAINS
C FROM THE WALLS, WINDOWS AND DOORS.
TOWALL(ITIM,N)=0.0
C TCWINS(ITIM,N) IS THE TOTAL SENSIBLE INFILTRATION ON WALL N AT
C TIME ITIM.
TCWINS(ITIM,N)=0.0
C TOWLAT(ITIM,N) IS THE TOTAL LATENT INFILTRATION ON WALL N AT TIME
C ITIM.
TOWLAT(ITIM,N)=0.0
1 CONTINUE
C LOOP(2) SUMS UP HEAT GAINS ON EACH WALL FOR EACH HOUR.
DO 2 N=2,N1
DO 2 ITIM=1,24
ITIN=ITIM+6
TCWALR(ITIM,N)=QLCT(ITIN,N)
KK=NCWIN(N)

```

```

      KK5=NCDOOR(N)
      IF(KK.EQ.0) GO TO 10
      DO 3 III=1, KK
C      SUMMATION FOR WINDOWS.
      TOWLAT(ITIM,N)=TOWLAT(ITIM,N)+ALFIL(N,III,ITIM)
      TOWINS(ITIM,N)=TOWINS(ITIM,N)+SINFIL(N,III,ITIM)
      TOWALR(ITIM,N)=TOWALR(ITIM,N)+QWIND(ITIM,N,III)
3     CONTINUE
10    IF(KK5.EQ.0) GO TO 2
      DO 4 MM=1, KK5
C      SUMMATION FOR DOORS.
      III=MM+KK
      TOWLAT(ITIM,N)=TOWLAT(ITIM,N)+ALFIL(N,III,ITIM)
      TOWINS(ITIM,N)=TOWINS(ITIM,N)+SINFIL(N,III,ITIM)
      TOWALR(ITIM,N)=TOWALR(ITIM,N)+QDOOR(ITIM,N,MM)
4     CONTINUE
2     CONTINUE
C      LOOP(5) TAKES THE WALL TOTALS FROM LOOP(2) AND SUMS THESE UP FOR
C      HOURLY TOTALS FOR THE ENTIRE HOUSE.
      DO 5 ITIM=1,24
      ITIN=ITIM+6
      TOHO(ITIM)=RQLOT(ITIN)
      TOHOUS(ITIM)=ALIGHT(ITIM)+PEOPLS(ITIM)+PEOPLL(ITIM)
      TOLAT(ITIM)=PEOPLL(ITIM)
      DO 5 N=2,N1
C      TOHO(ITIM) ARE THE HOURLY HEAT GAINS THAT ARE DELAYED IN
C      SUBROUTINE COCLSU.
      TCHO(ITIM)=TOHO(ITIM)+TOWALR(ITIM,N)
      TOWALL(ITIM,N)=TOWLAT(ITIM,N)+TOWINS(ITIM,N)+TOWALR(ITIM,N)
      TOHOUS(ITIM)=TOHOUS(ITIM)+TOWALL(ITIM,N)
      TOLAT(ITIM)=TOLAT(ITIM)+TOWLAT(ITIM,N)
5     CONTINUE
      WRITE(6,105)
      WRITE(6,106)(NUMB(I),I=2,N1)
      DO 88 ITIM=1,24
C      TOWALR(ITIM,N) CAN GIVE THE USER AN IDEA OF HOW THE HEAT GAIN FOR
C      WALLS, WINDOWS, AND DOORS ARE DISTRIBUTED.
88    WRITE(6,107) ITIM,(TOWALR(ITIM,N),N=2,N1)
      RETURN
      END

```

```

SUBROUTINE COOLSU
C THIS SUBROUTINE CHANGES HEAT GAINS (TOHO(ITIM)) INTO COOLING LOADS
C (COOL(ITIM)). THEY ARE CHANGED BY A RESPONSE FACTOR EQUATION, THIS
C EQUATION IS WRITTEN AS A FUNCTION(COLOAD(X)). THE RESPONSE FACTORS
C ARE ADJUSTED BY THE USE OF A CORRECTION FACTOR THAT CORRECTS THESE
C VALUES TO THE TYPE OF CONSTRUCTION THAT IS BEING USED. THE VALUES
C IN TOHO(ITIM) MUST BE PLACED IN ANOTHER ARRAY CALLED COLO(ITIM+4).
C THIS IS DONE BECAUSE THE RESPONSE FACTOR EQUATION NEEDS THE LAST 4
C VALUES OF THE DAY TO CALCULATE THE FIRST HOUR OF THE DAY.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHECK,NN,NNN,N1,JCHECK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),NODOOR(15),WINDCW(15,10,8),NUMB(15),KODE
COMMON/AIRC5/TEMP(24),SCLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLCO(15)
COMMON/AIRC8/SQLCO(15),UCE,T,ROOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QDOOR(24,15,10),QWIND(24,15,10)
COMMON/AC21/TA(30),NOCE,NOWA(15),CEAR,VOLAIR,TR(30),WT(30)
COMMON/AIRC3/QLCT(30,15),RQLCT(30)
COMMON/AIRC10/ALIGHT(24),PEOPLS(24),PEOPLL(24),MEMBER,TOTLIG
COMMON/AIRC11/TOHOUS(24),TOLAT(24),TCWINS(24,15),TOWALR(24,15)
COMMON/AIRC12/TOWLAT(24,15),TOWALL(24,15),FACTOR,COLO(28)
COMMON/AIRC13/COGL(28),CT(24),TOHO(24)
C THE PURPOSE OF THIS SUBROUTINE IS TO TURN HEAT GAINS INTO COOLING LOADS
109 FORMAT(3X,'TIME',1X,15I7)
110 FORMAT(18,15F7.1)
111 FORMAT(15F7.1)
C THIS LOOP(1) LOADS THE TOHO(ITIM) VALUES INTO COLO(ITIM+4).
DO 1 ITIM=1,24
ITIN=ITIM+4
1 COLO(ITIN)=TOHO(ITIM)
C THIS LOOP(2) PLACES THE LAST 4 VALUES OF THE DAY IN FRONT OF THE
C PREVIOUS VALUES.
DO 2 ITIM=1,4
2 COLO(ITIM)=COLO(ITIM+24)
DO 3 ITIM=1,28
C COOL(ITIM) MUST BE GIVEN AN INITIAL VALUE.
3 COOL(ITIM)=0.0
C=0.0
A=CEAR*UCE
C LOOP(20) SUMS UP AN AREA WEIGHTED U-FACTOR FOR THE ENTIRE HOUSE.
DO 20 N=2,N1
C=C+DIME2(N,3)
A=A+U(N)*DIME2(N,6)
KK=NOWIN(N)
KK5=NODOOR(N)
IF(KK.EQ.0) GO TO 5
DO 4 III=1,KK
4 A=A+UW(N,III)*WINDOW(N,III,2)*WINDOW(N,III,5)

```



```

5 IF(KK5.EQ.0) GO TO 20
DO 6 MM=1, KK5
  III=MM+KK
6 A=A+UD(N,MM)*WINDOW(N,III,2)*WINDOW(N,III,5)
20 CONTINUE
  AKT=A/C
  FACTOR=-0.0171*AKT+1.0
  WRITE(6,111) FACTOR
C   LOOP(12) MAKES THE COOLING LOAD CALCULATIONS FOR THREE DAYS.
DO 12 J=1,3
C   LOOP(10) CALLS CCLCAD(X) FOR A TWENTY-FOUR HOUR PERIOD.
DO 10 ITIM=5,28
10 COOL(ITIM)=CCLCAD(X)
C   LOOP(11) GIVES THESE CALCULATIONS CONTINUITY FROM ONE DAY TO
C   ANOTHER.
DO 11 ITIM=1,4
11 COOL(ITIM)=COOL(ITIM+24)
12 CONTINUE
  RETURN
  END

```

```

FUNCTION COLOAD(Z)
C   THIS FUNCTION IS THE RESPONSE FACTOR EQUATION THAT CALCULATES
C   COOLING LOADS FROM HEAT GAINS. THIS METHOD CAN BE FOUND IN THE
C   ASHRAE GUIDE, 1972.
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),NCDCCR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC5/TEMP(24),SOLAVW(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC7/QLCC(30,15),RQLOD(30),AQLOD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,ROOPT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),ODDOR(24,15,10),OWIND(24,15,10)
COMMON/AIRC3/QLCT(30,15),RQLCT(30)
COMMON/AIRC10/ALIGHT(24),PECPLS(24),PEOPLL(24),MEMBER,TOTLIG
COMMON/AIRC11/TOFOUS(24),TOLAT(24),TOWINS(24,15),TOWALR(24,15)
COMMON/AIRC12/TOWLAT(24,15),TOWALL(24,15),FACTOR,COLO(28)
COMMON/AIRC13/COOL(28),CT(24),TOHC(24)
X=FACTOR*(0.6982*CCLO(ITIM)-1.2017*COLO(ITIM-1)+0.6617*COLO(ITIM-2
1)-0.1150*COLO(ITIM-3))
Y=1.8260*COOL(ITIM-1)-1.0697*COOL(ITIM-2)+0.2005*COOL(ITIM-3)
COLOAD=X+Y
RETURN
END

```

```

SUBROUTINE TOCOOL
COMMON/AIRC/N,AA,ITIM,ALAT,M,DIME2(10,15),ICHEK,NN,NNN,N1,JCHEK
COMMON/AIRC/ALITE(24,15),BLITE(24,15)
COMMON/AIRC1/NOWIN(15),NODOOR(15),WINDOW(15,10,8),NUMB(15),KODE
COMMON/AIRC5/TEMP(24),SOLAVH(24,15,10),DTM(24,15,10),RTM(30,15)
COMMON/AIRC7/QLOD(30,15),RQLOD(30),AQLOD(15)
COMMON/AIRC8/SQLOD(15),UCE,T,RCOMT,UW(15,10),U(15)
COMMON/AIRC9/UD(15,10),QCONCR(24,15,10),QWIND(24,15,10)
COMMON/AIRC3/QLOT(30,15),RQLOT(30)
COMMON/AIRC10/ALIGHT(24),PECPLS(24),PEOPLL(24),MEMBER,TOTLIG
COMMON/AIRC11/TOHOUS(24),TOLAT(24),TCWINS(24,15),TCWALR(24,15)
COMMON/AIRC12/TOWLAT(24,15),TOWALL(24,15),FACTOR,COLO(28)
COMMON/AIRC13/COOL(28),CT(24),TCHC(24)
C THE PURPOSE OF THIS SUBROUTINE IS TO SUM THE COOLING LOADS FOR THE ENTIRE
C HOUSE
  DO 1 ITIM=1,24
    ITIN=ITIM+4
    CT(ITIM)=ALIGHT(ITIM)+PECPLS(ITIM)+TOLAT(ITIM)+COOL(ITIN)
    DO 1 N=2,N1
      1 CT(ITIM)=CT(ITIM)+TOWINS(ITIM,N)
C CT(ITIM) IS THE TOTAL COOLING LOAD FOR THE HOUSE AT EACH HOUR
  RETURN
  END

```

C THE FOLLOWING ARE THE DATA CARDS FOR THE SIX CASES USED IN THE
C THESIS.

TIME OF DAY (HOURS)									
COOLING LOAD IN BTU/HR									
1	2	3	4	5	6	7	8	9	10
1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0		
9.0	10.0	11.0	12.0	15.0	16.0	1.0	2.0		
3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0		
11.0	12.0	15.0	16.0	1.0	2.0	3.0	4.0		
5.0	6.0	7.0	8.0	9.0	10.0	11.0	12.0		
15.0	16.0	1.0	2.0	3.0	4.0	5.0	6.0		
7.0	8.0	9.0	10.0	11.0	12.0	15.0	16.0		
1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0		
9.0	10.0	11.0	12.0	15.0	16.0	1.0	2.0		
3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0		
11.0	12.0	15.0	16.0	14.0					
4.0	0.0	8.0	40.0	9999.00	680.00	698.6	3.0	17.45	46.0
1.0	0.44	40.0	17.0	9999.00	464.66	0.0	0.0	0.0	0.0
2.0	0.0	27.33	17.0	9999.00	464.66	0.0	0.0	0.0	0.0
3.0	0.44	40.0	17.0	9999.00	680.00	698.6	3.0	17.45	46.00
4.0	0.0	27.33	17.00	9999.00	464.66	0.0	0.0	0.0	0.0
5	4	6	4						
1	0	1	0						
2.4	5.5	9.1	29.8	5.8	5.4	4.0	31.9		
2.4	5.5	9.1	5.95	5.5	31.55	4.0	30.25		
11.3	5.1	0.6	30.6	3.4	6.0	2.0	17.34		
11.3	5.1	0.6	19.54	3.4	17.06	2.0	17.34		
11.3	5.1	0.6	7.64	3.4	28.96	2.0	17.34		
0.0	6.4	9.6	19.0	3.0	18.0	53.0	19.2		
2.4	5.5	9.1	5.33	3.33	18.66	2.0	14.66		
2.4	5.5	9.1	16.00	2.66	8.66	2.0	18.33		
11.3	5.1	0.6	21.66	3.0	2.66	2.0	15.33		
11.3	5.1	0.6	1.33	3.0	23.00	2.0	15.33		
2.4	5.5	9.1	26.33	9.0	4.66	6.0	49.5		
2.4	5.5	9.1	10.66	6.0	23.33	4.0	33.0		
2.4	5.5	9.1	4.33	3.66	32.00	4.0	20.17		
11.3	5.1	0.6	25.66	3.0	11.33	2.0	15.3		
11.3	5.1	0.6	17.0	2.66	20.33	2.0	13.6		
11.3	5.1	0.6	10.0	3.0	27.00	2.0	15.3		
0.0	6.4	9.6	19.0	2.66	18.33	53.0	17.07		
2.4	5.5	9.1	16.0	2.66	8.66	2.0	14.67		
2.4	5.5	9.1	5.33	3.33	18.66	2.0	18.33		
11.3	5.1	0.6	20.66	3.00	2.66	2.0	15.3		
11.3	5.1	0.6	1.33	3.00	23.0	2.0	15.3		
75.0									
77.6	76.6	75.8	75.2	75.0	75.4	76.4	78.2	80.8	83.8
87.2	90.4	92.8	94.4	95.0	94.4				
93.0	90.8	88.2	85.6	83.4	81.4	79.8	78.6		

11.3	5.1	0.6	7.64	3.4	28.96	1.0	17.34
0.0	6.4	9.6	19.0	3.0	18.0	51.0	19.2
2.4	5.5	9.1	5.33	3.33	18.66	1.0	14.66
2.4	5.5	9.1	16.00	2.66	8.66	1.0	18.33
11.3	5.1	0.6	21.66	3.0	2.66	1.0	15.33
11.3	5.1	0.6	1.33	3.0	23.00	1.0	15.33
2.4	5.5	9.1	26.33	9.0	4.66	5.0	49.5
2.4	5.5	9.1	10.66	6.0	23.33	3.0	33.0
2.4	5.5	9.1	4.33	3.66	32.00	3.0	20.17
11.3	5.1	0.6	25.66	3.0	11.33	1.0	15.3
11.3	5.1	0.6	17.0	2.66	20.33	1.0	13.6
11.3	5.1	0.6	10.0	3.0	27.00	1.0	15.3
0.0	6.4	9.6	19.0	2.66	18.33	51.0	17.07
2.4	5.5	9.1	16.0	2.66	8.66	1.0	14.67
2.4	5.5	9.1	5.33	3.33	18.66	1.0	18.33
11.3	5.1	0.6	20.66	3.00	2.66	1.0	15.3
11.3	5.1	0.6	1.33	3.00	23.0	1.0	15.3
75.0							
77.6	76.6	75.8	75.2	75.0	75.4	76.4	78.2
93.0	90.8	89.2	85.6	83.4	81.4	79.8	78.6
2	2	2	2				
		21093.3	4480.0				
6		4436.5					
4.0		0.0	8.0	40.0			
1.0	0.44	40.0	17.0	9999.0	680.0	698.6	3.0
2.0	0.0	27.33	17.0	9999.00	464.66	0.0	0.0
3.0	0.44	40.	17.0	9999.00	680.00	698.6	3.0
4.0	0.0	27.33	17.00	9999.00	464.66	0.0	0.0
5	4	6	4				
1	0	1	0				
2.4	5.5	9.1	29.8	5.8	5.4	3.0	31.9
2.4	5.5	9.1	5.95	5.5	31.55	3.0	30.25
11.3	5.1	0.6	30.6	3.4	6.0	1.0	17.34
11.3	5.1	0.6	19.54	3.4	17.06	1.0	17.34
11.3	5.1	0.6	7.64	3.4	28.96	1.0	17.34
0.0	6.4	9.6	19.0	3.0	18.0	51.0	19.2
2.4	5.5	9.1	5.33	3.33	18.66	1.0	14.66
2.4	5.5	9.1	16.00	2.66	8.66	1.0	18.33
11.3	5.1	0.6	21.66	3.0	2.66	1.0	15.33
11.3	5.1	0.6	1.33	3.0	23.00	1.0	15.33
2.4	5.5	9.1	26.33	9.0	4.66	5.0	49.5
2.4	5.5	9.1	10.66	6.0	23.33	3.0	33.0
2.4	5.5	9.1	4.33	3.66	32.00	3.0	20.17
11.3	5.1	0.6	25.66	3.0	11.33	1.0	15.3
11.3	5.1	0.6	17.0	2.66	20.33	1.0	13.6
11.3	5.1	0.6	10.0	3.0	27.00	1.0	15.3
0.0	6.4	9.6	19.0	2.66	18.33	51.0	17.07
2.4	5.5	9.1	16.0	2.66	8.66	1.0	14.67

2.4	5.5	9.1	5.33	3.33	18.66	1.0	18.33								
11.3	5.1	0.6	20.66	3.00	2.66	1.0	15.3								
11.3	5.1	0.6	1.33	3.00	23.0	1.0	15.3								
75.0															
77.6	76.6	75.8	75.2	75.0	75.4	76.4	78.2	80.8	83.8	87.2	90.4	92.8	94.4	95.0	94.4
93.0	90.8	88.2	85.6	83.4	81.4	79.8	78.6								
4	4	4	4												
		31093.3		4480.0											
6		4436.5													
4.0		0.0		8.0		40.0									
1.0	0.44	40.0	17.0	9999.0	680.0	698.6		1.5	17.45	46.0					
2.0	0.0	27.33	17.0	9999.00	464.66	0.0		0.0	0.0	0.0					
3.0	0.44	40.	17.0	9999.00	680.00	698.6		1.5	17.45	46.00					
4.0	0.0	27.33	17.00	9999.00	464.66	0.0		0.0	0.0	0.0					
5	4	6	4												
1	0	1	0												
2.4	5.5	9.1	29.8	5.8	5.4	3.0	31.9								
2.4	5.5	9.1	5.95	5.5	31.55	3.0	30.25								
11.3	5.1	0.6	30.6	3.4	6.0	1.0	17.34								
11.3	5.1	0.6	19.54	3.4	17.06	1.0	17.34								
11.3	5.1	0.6	7.64	3.4	28.96	1.0	17.34								
0.0	6.4	9.6	19.0	3.0	18.0	51.0	19.2								
2.4	5.5	9.1	5.33	3.33	18.66	1.0	14.66								
2.4	5.5	9.1	16.00	2.66	8.66	1.0	18.33								
11.3	5.1	0.6	21.66	3.0	2.66	1.0	15.33								
11.3	5.1	0.6	1.33	3.0	23.00	1.0	15.33								
2.4	5.5	9.1	26.33	9.0	4.66	5.0	49.5								
2.4	5.5	9.1	10.66	6.0	23.33	3.0	33.0								
2.4	5.5	9.1	4.33	3.66	32.00	3.0	20.17								
11.3	5.1	0.6	25.66	3.0	11.33	1.0	15.3								
11.3	5.1	0.6	17.0	2.66	20.33	1.0	13.6								
11.3	5.1	0.6	10.0	3.0	27.00	1.0	15.3								
0.0	6.4	9.6	19.0	2.66	18.33	51.0	17.07								
2.4	5.5	9.1	16.0	2.66	8.66	1.0	14.67								
2.4	5.5	9.1	5.33	3.33	18.66	1.0	18.33								
11.3	5.1	0.6	20.66	3.00	2.66	1.0	15.3								
11.3	5.1	0.6	1.33	3.00	23.0	1.0	15.3								
75.0															
77.6	76.6	75.8	75.2	75.0	75.4	76.4	78.2	80.8	83.8	87.2	90.4	92.8	94.4	95.0	94.4
93.0	90.8	88.2	85.6	83.4	81.4	79.8	78.6								
4	4	4	4												
		31093.3		4480.0											
6		4436.5													
4.0		90.0		8.0		40.0									
1.0	0.44	40.0	17.0	9999.0	680.0	698.6		3.0	17.45	46.0					
2.0	0.0	27.33	17.0	9999.00	464.66	0.0		0.0	0.0	0.0					
3.0	0.44	40.	17.0	9999.00	680.00	698.6		3.0	17.45	46.00					
4.0	0.0	27.33	17.00	9999.00	464.66	0.0		0.0	0.0	0.0					

5	4	6	4
1	0	1	0
2.4	5.5	9.1	29.8
2.4	5.5	9.1	5.95
11.3	5.1	0.6	30.6
11.3	5.1	0.6	19.54
11.3	5.1	0.6	7.64
0.0	6.4	9.6	19.0
2.4	5.5	9.1	5.33
2.4	5.5	9.1	16.00
11.3	5.1	0.6	21.66
11.3	5.1	0.6	1.33
2.4	5.5	9.1	26.33
2.4	5.5	9.1	10.66
2.4	5.5	9.1	4.33
11.3	5.1	0.6	25.66
11.3	5.1	0.6	17.0
11.3	5.1	0.6	10.0
0.0	6.4	9.6	19.0
2.4	5.5	9.1	16.0
2.4	5.5	9.1	5.33
11.3	5.1	0.6	20.66
11.3	5.1	0.6	1.33
75.0			
77.6	76.6	75.8	75.2
93.0	90.8	88.2	85.6
4	4	4	4
31093.3			4480.0
6			4436.5

APPENDIX E

The purpose of this appendix is to make and compare a hand calculated cooling load to the computer cooling load. The hand calculations were made by the TETD method described in the ASHRAE Handbook of Fundamentals. These loads were calculated at three times; 2:00 PM, 4:00 PM, and 6:00 PM for Case IV. Table 5 shows the total hand calculations, the breakdown of these calculations and the comparable computer calculated loads.

The conduction loads were calculated by a combination of three hour average TETD's and instantaneous TETD's to gain the time lag characteristic of a frame wall.

The cooling load through the windows used a combination of three hour solar heat gain averages and instantaneous solar heat gains.

The infiltration, both latent and sensible, was calculated by the length of crack method.

The results of the hand calculations followed the same general profile as the computer load but were a little over 10% larger in magnitude.

TABLE 5 COOLING LOAD FOR CASE IV
BY TETD METHOD

		2:00	4:00	6:00
Conduction Cooling Loads	South Wall	1394.7	1399.1	1186.6
	West Wall	916.0	1247.3	1500.7
	North Wall	798.4	865.3	877.8
	East Wall	1020.1	919.3	864.1
	Roof	5555.5	5818.2	4721.6
Window Cooling Loads	South Windows	3650.2	2425.0	1412.3
	West Windows	2511.7	3864.0	2615.6
	North Windows	2806.1	2539.6	1835.7
	East Windows	1224.4	1093.0	633.0
Infiltration Cooling Load	South Sensible	218.3	218.3	177.8
	South Latent	248.6	248.6	248.6
	West Sensible	109.5	109.5	89.2
	West Latent	124.7	124.7	124.7
	North Sensible	272.1	272.1	221.6
	North Latent	309.7	309.7	309.7
	East Sensible	109.5	109.5	89.2
	East Latent	124.7	124.7	124.7

Internal Cooling Load	People	450.0	450.0	1800.0
Total Cooling Load		21085.7	21379.5	18140.1
Cooling Load From Program		18640.0	19104.5	16110.3

ACKNOWLEDGEMENTS

The author wishes to extend his deepest appreciation to Dr. Robert L. Gorton for his time, wisdom and encouragement given very generously throughout this thesis.

The assistance, guidance and Cal Comp Plotter subroutine from Dr. Hugh S. Walker are gratefully acknowledged.

Appreciation is also extended to Dr. Everett E. Haft and Dr. John E. Kipp for serving as committee members.

A special thanks goes to Mr. G.P. Mitalas for his work related to thermal response factors which this thesis was based on.

A special recognition goes to my wife, Donna, who gave her emotional and economic support during this study.

The author also acknowledges the Mechanical Engineering Department for the financial aid it provided.

A DIGITAL COMPUTER PROGRAM TO CALCULATE
A RESIDENTIAL COOLING LOAD

by

DANA MILAN SMERCHEK
B.S., Kansas State University, 1973

AN ABSTRACT OF A THESIS

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Mechanical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1975

Name: Dana M. Smerchek

Date of Degree: May 16, 1975

Institution: Kansas State University

Location: Manhattan, Kansas

Title of Study: A DIGITAL COMPUTER PROGRAM TO CALCULATE A RESIDENTIAL
COOLING LOAD

Pages in Study:

Candidate for Degree of Master of Science

Major Field: Mechanical Engineering

Home designers have used several different methods to calculate cooling loads for residential homes. The purpose of this thesis was to write and check by use on a typical residence, a transient cooling load program for residential type construction.

This digital computer program should aid the home designer in selecting construction materials, width of roof overhangs, and the orientation of the house on its lot.

The input data necessary is the environmental conditions, the shade of the home and its dimensions. However, the following home limitations exist: there must be the same number of floors throughout the floor plan; all extensions from the main house are required to be at right angles; all of the roof hip ridges are required to meet at the same level.

Conduction heat gains were calculated for walls, ceiling, doors and windows. The heat gains for walls and ceilings were calculated by the response factor method. The remaining conduction heat gains were calculated by instantaneous heat gain calculations.

Infiltration loads sensible and latent were calculated by the length of crack method.

The internal loads were calculated by estimated load schedules for lights and people.

These heat gains were summed at each hour and converted to cooling loads by a response factor method.

A rectangular two story house was used as the test home. Six different test cases were used to demonstrate the program's abilities. The first four cases varied the construction materials to demonstrate the program's ability to calculate cooling loads for various constructions. The fifth case changed the width of overhang and the sixth case rotated the house from facing south to west. These last two cases demonstrated the program's ability to calculate different shading cases.

It is hoped that this program will aid the home designer in making the economic and energy decisions which must be made for future single-family housing units.

VITA

Dana Milan Smerchek

Candidate for the Degree

Master of Science

Thesis: A DIGITAL COMPUTER PROGRAM TO CALCULATE A RESIDENTIAL COOLING LOAD

Major Field: Mechanical Engineering

Biographical:

Personal Data: Born in Topeka, Kansas, February 9, 1951; the son of the late Milan W. and Kathleen W. Smerchek.

Education: Graduated from Washburn Rural High School in 1969; received a Bachelor of Science degree in Mechanical Engineering from Kansas State University in December, 1973; completed requirements for the Master of Science degree in May, 1975.

Professional Experience: Employed during the summer of 1972 by Western Electric, Lee's Summit, Missouri, as a Summer Engineer in their Plastics Laboratory; during the summer of 1973 was employed by Amoco Oil Company's Research Center, Whiting, Indiana, in their automotive laboratory.